

Out-of-equilibrium sampling as a protocol to unveil locally-smooth neural network configurations in non-convex optimization

Clarissa Lauditi

Supervisors
Prof. Riccardo Zecchina
Asst. Prof. Enrico M. Malatesta

A Dissertation
submitted to the Faculty of Graduate Studies
in partial fulfillment of the requirements
for the Degree of
Doctor of Philosophy

Graduate Program in Physics
Politecnico of Turin University
Turin, Italy

© Clarissa Lauditi, 2024

Contents

List of Figures vii

Introduction 4

I Preliminaries 5

1 Emerging behaviors in neural networks 7
- 1.1 Basics on self-/un-/supervised learning . . . 7
- 1.2 Fundamental questions in the theory of Deep Learning . . . 9
  - 1.2.1 The role of initialization . . . 12
  - 1.2.2 Avoiding glassy dynamics . . . 15
  - 1.2.3 Algorithmic hardness from a geometric perspective . . . 19

2 Biasing the measure into wide valleys 23
- 2.1 The instructive case of the binary Perceptron . . . 24
  - 2.1.1 Contradictory numerical results . . . 27
  - 2.1.2 Large deviation analysis . . . 28
- 2.2 Local entropy as an objective function: the robust ensemble . . . 29
  - 2.2.1 Focusing - BP . . . 30

II The benefits of robustness 33

3 Sampling atypical solutions in binary models 35
- 3.1 Storing $P \simeq \mathcal{O}(N)$ random patterns with a Perceptron . . . 35
  - 3.1.1 Equilibrium configurations . . . 37
  - 3.1.2 Characterizing the landscape in extensive neighborhoods . . . 41
- 3.2 Non-convex networks with multiple layers . . . 48
  - 3.2.1 The storage problem for a Tree Committee machine . . . 48
- 3.3 Numerics . . . 53

4 Atypical phase transition in the overparameterized regime 55
- 4.1 Model definition and learning setup . . . 56
  - 4.1.1 Gaussian Equivalence Principle . . . 57

4.2 Geometry of minima vs overparameterization . . . 58
4.2.1 Averaged free-entropy . . . 59
4.2.2 The generalization error . . . 63
4.2.3 Local entropy landscape of solutions . . . 68
4.3 Comparing solutions in deep architectures . . . 72
4.3.1 Overparameterized TCM . . . 72
4.3.2 Multi-layer NNs . . . 74

5 Linear mode connectivity in continuous non-convex loss landscape 79
5.1 The negative Perceptron problem . . . 80
5.2 Typical overlap between pairs of solutions . . . 82
5.2.1 Nested overlap structure . . . 84
5.3 Manifold spanned by y solutions . . . 85
5.3.1 Sampling with the same margin $k \geq \kappa_E$ . . . 87
5.3.2 Sampling with different margins . . . 90
5.3.3 Star-shaped space of solutions . . . 93
5.4 More features on the solution space . . . 94
5.4.1 Non-equilibrium solutions encountered along the path . . . 94
5.4.2 Anisotropy of the solution manifold . . . 95
5.5 Sampling bias and disconnection . . . 97

6 The rewards of a negative margin 101
6.1 The negative RFM . . . 102
6.2 Deeper models . . . 107
6.3 Generalization error along the geodesic . . . 109

III Neuroscience comes into play 113

7 Modeling dendritic non-linearities with a TCM 115
7.1 The single neuron model . . . 116
7.2 Memory-related properties . . . 118
7.2.1 Analytical estimation of the critical capacity . . . 118
7.2.2 Algorithmic capacity, learning speed and noise robustness . . . 123
7.3 Distribution of synaptic weights . . . 126
7.4 Generalization performance . . . 128

8 The random features Hopfield Model 131
8.1 Phase diagram: storage and learning regimes . . . 132
8.1.1 Specializing to one feature retrieval . . . 133
8.1.2 Retrieval of one pattern . . . 136
8.1.3 Merging the results . . . 138
8.2 Generalization in the RFHM . . . 139
8.2.1 Signal-to-noise analysis . . . 141

IV Conclusions 145

9 Conclusions and perspectives 147

Publications 153

Bibliography 155

Acronyms 167

A Details on the RFM 171
A.1 Average over the disorder: GEP . . . 171
A.2 Equilibrium configuration limits . . . 173
A.2.1 Storage case limit . . . 173
A.2.2 Overparameterization regime . . . 174
A.3 Bayesian generalization error . . . 174
A.3.1 Entropic - Energetic term . . . 175
A.4 Stability distribution . . . 177
A.5 Franz-Parisi potential . . . 179
A.5.1 RS - Ansatz . . . 182

B Details on the negative Perceptron 185
B.1 Typical overlap between pairs of solutions . . . 185
B.1.1 RS - Ansatz . . . 186
B.1.2 $R \to p$ limit . . . 187
B.2 Training error of the convex envelope . . . 188
B.2.1 General structure for the overlap matrix . . . 190
B.2.2 Specific case with different margins . . . 191
B.2.3 Convex subset in a probabilistic setting . . . 192
B.3 Stability distribution along the geodesic path . . . 193
B.4 Sampling bias of the Perceptron Algorithm . . . 195
B.5 Extrapolation to large N values . . . 196

C Details of the *biological* TCM 197
C.1 Free entropy and saddle points . . . 197
C.1.1 Effective order parameters for some non-linearities . . . 198
C.2 The critical capacity . . . 200
C.3 Distribution of synaptic weights . . . 202
C.3.1 Maximal storage limit . . . 203

D Details on the RF-Hopfield Model 205
D.1 Feature retrieval . . . 205
D.1.1 RS - Ansatz . . . 207
D.2 Pattern retrieval . . . 210
D.2.1 RS - Ansatz . . . 212
D.3 Numerical simulations . . . 213

D.3.1 Additional figures 214

List of Figures

1.1 Rectangular deep network with L layers and n_l neurons for each. Every weight matrix is a matrix $\boldsymbol{W}_\ell \in \mathbb{R}^{n_{l-1} \times n_l}$ and given an input $\boldsymbol{\xi}^\mu$, the network implements the function $f_{\boldsymbol{w}}(\boldsymbol{\xi}^\mu)$ until the output. 13

1.2 Train error as a function of the iteration number for a deep linear network with $L = 4$ layers and different values of the width w. In the NTK regime (*left*), the training loss shows typical linear convergence behaviour at large width; in the saddle-to-saddle regime (*right*), the dynamics is slower and the plateaus become longer as the width grows. Reprinted from [3]. 14

1.3 Train and test losses of the negative log-likelihood (or accuracy) as a function of $\log(t)$ for a ResNet-18 on CIFAR100 (*left*) with batch size B and learning rate α. On the *right*, a one-hidden-layer NN with 10 hidden neurons trained with MSE on CIFAR-10. The batch size is $B = 100$ and the learning rate is $\alpha = 0.1$. In both cases the training is carried out by SGD. Reprinted from [24]. 17

1.4 One-dimensional sketch of a loss landscape exhibiting OGP. Solutions are split into clusters, with diameter of each cluster smaller than the distance between any pair clusters. Reprinted from [80]. 20

2.1 *Left* panel: averaged local entropy of typical solutions as a function of the distance, for several values of α. *Right* panel: distance $d_{\min}$ for which the Franz-Parisi potential is zero. At the SAT/UNSAT transition line (vertical dashed) the distance coincides with the typical distance between solutions $d \simeq 0.218$. Reprinted from [88]. 27

2.2 Random walk on the solutions belonging to the same connected component. These configurations, despite being invisible to the replica computation, are present everywhere in the solution space of the binary Perceptron. Reprinted from [89]. 27

2.3 Local entropy of atypical solutions (red), i.e. subdominant wrt the flat measure, and typical solutions (blue) as a function of the distance d from a reference for different values of α. Reprinted from [46]. 29

2.4 (A): Portion of a BP factor graph for a replicated variable w_j with $y = 3$ replicas and a reference $w_j^\star$. The squares represents the interactions $\gamma w_j^\star w_j$. All the messages are assumed to be the same in corresponding edges. (B): Transformed graph once exploiting the symmetry to reduces the number of nodes, keeping only one representative per replicas. The hexagon is now the pseudo-self-interaction that depends on both γ and y. Reprinted from [45]. 30

3.1 *Left* panel: the cube represents the space of possible values of weights for a binary Perceptron model. The green shaded area is the convex space for the corresponding spherical problem. Here only two vertices of the cube lies inside the green region. *Middle* and *right* panels: space of solutions (green shaded area) in the spherical Perceptron with $\kappa = 0$ and $\kappa < 0$ respectively. Reprinted from [88]. 36

3.2 *Left*: the value $\kappa_{\max}$ at which the entropy vanishes, plotted as a function of α. This quantity goes to 0 at α_c. Reprinted from [90]. *Right*: RS entropy as a function of the margin κ for several values of α and $\kappa_{\max}$ as a function of α. 40

3.3 Normalized Hamming distance d between typical solutions as a function of their margin κ and for different α values. The lines become dashed when the entropy of solutions becomes negative, i.e. when $\kappa > \kappa_{\max}$. Reprinted from [90]. 41

3.4 Local entropy profiles (with $\kappa = 0$) of typical solutions at $\alpha = 0.5$ as a function of the distance, for various values of $\tilde{\kappa}$. The dashed line is the geometric upper bound obtained by counting all the configurations at $\alpha = 0$. The inset shows a detail of the three curves in the small d range with colors corresponding to the main plot. Points with error bars show the results of numerical experiments (10 samples at $N = 2001$) obtained with the focusing-BP algorithm. Reprinted from [90]. 45

3.5 Representation of a portion of the space of network configurations. The dots represent solutions (zero-error configurations) with different κ margins. Red arrows indicate four examples of typical solutions. Yellow arrows indicate three examples of the type of atypical solutions found around the typical ones with a larger margin. Reprinted from [90]. 46

3.6 Local entropy profiles (with $\kappa = 0$) of typical maximum margin solutions (left panel) and its derivative (right panel) as a function of the distance, for different values of α. For $\alpha = \alpha'_u \simeq 0.729$ the local entropy starts to be non-monotonic. Reprinted from [90]. 47

3.7 *ReLU* activation function: for $\alpha = \alpha'_u \simeq 0.773$ the local entropy starts to be non-monotonic. Reprinted from [90]. 52

3.8 *Sign* activation function: for $\alpha = \alpha'_u \simeq 0.558$ the local entropy starts to be non-monotonic. Reprinted from [90]. 52

3.9 Number of train errors (blue points, left y-axis) and mean overlap of pairs of solutions (red points, right y-axis) as a function of the training set size, for different architectures and datasets. The overlaps between solutions obtained with different random initializations is well below 1 even as the model is barely able to fit all of the patterns. The inset shows local energy profiles right below the algorithmic threshold, i.e. at the largest dataset size for which the error is still zero. Reprinted from [90]. 53

4.1 Entropy as a function of α for fixed $\alpha_T = 3$ and different values of the margin κ. Reprinted from [99]. 62

4.2 Entropy as a function of α_T for fixed $\alpha = 1$ and different values of the margin κ. Reprinted from [99]. 63

4.3 (*Left*) Generalization as a function of the overparameterization degree $1/\alpha$ in the log x-axis scale. (*Right*) Generalization error in the large overparameterization limit ($\alpha \to 0$) versus $\alpha_T = P/D$ for different values of the margin κ. The dotted lines represent the corresponding generalization error of the barycenter of typical solutions (see Sec. 4.2.2). Reprinted from [99]. 64

4.4 Train (left panel) and test error (right panel) of the SA algorithm as a function of $1/\alpha$. Different colors represent the result of the simulations with different values of D, while maintaining fixed $\alpha_T = 3$. The corresponding generalization error matches the replica theory result (dotted horizontal line) in the $\alpha \to 0$ limit. Reprinted from [99]. 67

4.5 Train (left panel) and test error (right panel) of zero-temperature Monte Carlo algorithm as a function of $1/\alpha$ for different values of margins. In the simulations we fixed $D = 201$ and $P = 603$ i.e. $\alpha_T = 3$, and we ran the algorithm for a fixed number of sweeps (200). We show also in dashed the analytical predictions coming from replica theory. Reprinted from [99]. 67

4.6 *Left*: local entropy (LE) profiles of solutions with margin $\kappa = 0$ and for some typical references having margins $\tilde{\kappa}$. Here $\alpha = 0.5$ and $\alpha_T = 8$. The maximum margin is $\kappa_{\max} \simeq 0.847$ and corresponds to the curve with the largest LE. The inset is a zoom of the entropies for small distances. *Right*: estimating the $\alpha_{\rm LE}$ transition by looking at the local entropy profiles of maximum margin references and its derivative with respect to the distance (inset). Here $\alpha_T = 2$ and the critical capacity is $\alpha_c \simeq 1.045$. Reprinted from [99]. 69

4.7 SAT/UNSAT interpolation threshold and Local Entropy transition versus α_T for the binary non-convex model of random features. For $\alpha_T \to 0$ we recover the critical capacity [22] and the local entropy transition [90] of a non-overparameterized binary perceptron trained on random patterns. In the inset we show the training error of SA, fBP and SBPI versus the degree of overparameterization $1/\alpha$ for $D = 201$ and $\alpha_T = 3$. Reprinted from [99]. 70

4.8 (*Left panel*) Generalization error as a function of the degree of overparameterization $1/\alpha$, for $\alpha_T = 3$. Vertical dashed lines denote the SAT-UNSAT transition α_c^{-1}, the local entropy transition $\alpha_{\rm LE}^{-1}$, and the $\kappa_{\rm opt}$ transition $(\alpha^*)^{-1}$. We show the generalization error of typical solutions with fixed margin $\kappa = 0$, 0.107, 1 and with maximum margin $\kappa_{\rm max}(\alpha, \alpha_T)$. The dashed turquoise curve is the error of the barycenter of typical solutions having zero margin, whereas the black line represents the generalization error of the "best" barycenter which was found by optimizing the margin. In the inset we show that this optimal margin $\kappa_{\rm opt}$ undergoes a transition when crossing α^*. We also show numerical results ($D = 201$) of two representative algorithms: SA (violet points) and fBP (blue points). Yellow points represent the barycenter of zero margin solutions as computed by using the BP estimation of the posterior distribution. (*Right panel*) Generalization error versus α_T in the large overparameterization regime ($\alpha = 0.05$, $D = 201$) for both theories and different optimization strategies. Reprinted from [99]. 71

4.9 Overparameterized continuous tree-like committee machine: tests performed with $D = 2001$, $\alpha_T = 5$ and $K = 9$. Dashed red line: mean overlap between two SGD solutions on the same sample, as a function of the number of parameters N. Solid lines: test error for BP and SGD. The vertical dashed grey line at $N = 3600$ ($\alpha^{-1} \approx 0.36$) denotes the algorithmic threshold below which the algorithms solve fewer than 50% of the samples. Inset: Local energy profiles, i.e. average train error as a function of the Euclidean distance from a solution. Dashed lines are measured at $N = 3996$ ($\alpha^{-1} \approx 0.4$), solid lines at $N \approx 13500$ ($\alpha^{-1} \approx 1.35$). Reprinted from [99]. 73

4.10 Train (full points) and test (empty points) error as a function of the network parameters N for a fully-connected network with 5 hidden layers. The network is trained using full batch gradient descent with ADAM optimization, with random orthogonal initialization (red curves) and adversarial initialization (blue curves). Mean overlap between solutions is far below 1 (dashed lines). Inset: local energy profiles for both algorithms at two different values of the overparameterization ($N \simeq 9 \cdot 10^3$ (full lines) and $N \simeq 2 \cdot 10^4$ (dashed lines)). Reprinted from [99]. . . . 76

4.11 (*Left*) Train (dashed) and test (solid) errors of solutions obtained with SGD and ADAM (gray and blue lines respectively) as a function of C. Inset: local energy as a function of the distance from solutions (gray: SGD with $C = 54$; blue: ADAM with $C = 64$). (*Right*) Overlap of a solution with its initial condition as a function of the epochs. The overlap is reported layer by layer and it refers to ADAM optimizer, a number of filters $C = 20$, and a learning rate $lr = 0.01$. Reprinted from [99]. 76

5.1 The zero temperature phase diagram of the continuous spherical perceptron showing the density of constraints α as a function of the margin $\kappa_{\scriptscriptstyle E}$ by which the problem is solved. For each $\kappa_{\scriptscriptstyle E} \leq 0$ the problem is non-convex and RSB transition effects appear. Reprinted from [123]. 81

5.2 Stability distributions related to a *typical* (left) and to an *atypical* (right) solutions at $\kappa_{\scriptscriptstyle E} = -0.7$. Notice how the threshold for the satisfiability coincide with the value Δ up to which the distribution is zero in the typical case, while $\Delta_{eff} = -0.5 > \Delta$ for robust solutions. . . . 82

5.3 As a function of k_2 from bottom to above: typical overlap q_1 of solutions having margin $k_1 = -0.7$ (dashed pink line); typical overlap p between two solutions having margin k_1 and k_2 (full blue line); typical overlap q_2 of solutions with margin k_2 (dashed black line). Here $\alpha = 3$. The dotted vertical line indicates the threshold κ_{dAT} for the RS stability. Reprinted from [119]. 84

5.4 Sketch of the projection of a straight path (*blue*) onto the geodesic (*red*) in the case of a one-dimensional path, i.e. $y = 2$. Reprinted from [119]. 85

5.5 (Left panel) Training error manifold spanned by $y = 3$ typical solutions where $k = \kappa_{\scriptscriptstyle E} = -0.5$ and $\alpha = 1$. (Right panel) Typical energy barrier between $y = 2$ solutions with $k = \kappa_{\scriptscriptstyle E} = -0.5$ and $\alpha = 1$ (corresponding to the edges of the triangle in panel a) compared with SA sampling at different sizes. Reprinted from [119]. 87

5.6 (Left panel) Training error at $\alpha = 1$ of the 2-dimensional simplex when $\kappa_2^\star < k < \kappa_3^\star$ and the energy related to the barycenter is strictly greater than zero. (Right panel) One dimensional projection of the barrier between solutions with increasingly large $k_{1,2}$. Extrapolation for large N related to SA predictions are also reported. Reprinted from [119]. . 90

5.7 (Left panel) Coalescence thresholds as a function of α for $\kappa_{\scriptscriptstyle E} = -0.5$. The dashed red curve stands for the dAT line. (Right panel) The barycenter energy of solutions at margin $\kappa_{\scriptscriptstyle E} = -0.5$ and $\alpha = 1$ as a function of k for different dimensions y of the ensemble. Reprinted from [119]. 90

5.8 (Left panel) 2-dimensional simplex involving the training error between 2 typical solutions (lower vertices, $\kappa_{\scriptscriptstyle E} = -0.5$) and a robust solution (top vertex, $k_2 = -0.1$) belonging to the kernel subset $\kappa_{\text{krn}} \sim -0.171$ for $\alpha = 1$. (Right panel) Training error between a typical solution (left-end point) at margin $k_1 = \kappa_{\scriptscriptstyle E} = -0.5$ and robust configurations at margins $k_2 = -0.4, -0.35, -0.3$. Since now $k_2 < \kappa_{\text{krn}}$ an energy barrier is always present. Reprinted from [119]. 92

5.9 Sketch (*left*) and zoom (*right*) of the star-shaped organization of the solution manifold. In both cases the blue shades corresponds to zero energy regions, while the red ones to UNSAT regions for the problem. Reprinted from [119]. 93

5.10 Coalescence threshold lines at $\kappa_E = -0.5$ (blue and cyan), and the κ_{krn} threshold (green line) as a function of α. In orange the dAT transition line, delimiting the RS-stable region; in red the RS estimate of the κ_{max}. Above the dAT line, dashed lines indicate the continuation of the (unstable) RS predictions. Points are numerical extrapolations from SA samples. Reprinted from [119]. 94
5.11 Stability distributions of solutions found by increasing γ along the path between a typical solution at $k_1 = -0.5$ towards a more robust solution at $k_2 = -0.1$ ($\alpha = 1$). The geodesic connecting the two solutions is at zero training error. Purple and red lines at the extremes of the curves represent the typical stability distribution for $\kappa_E = -0.5$ and $\kappa_E = -0.1$ respectively. Reprinted from [119]. 96
5.12 Typical overlap $\bar{p}$ between the barycenter of y independently-sampled solutions with margin κ_E and the maximum margin solution at $\kappa_{\max}(\alpha)$. The dashed curves refer to the $y = 3$ case while the full ones are for $y = \infty$. After the vertical dashed line ($\alpha_{\text{dAT}}(\kappa_E = 0)$) the predicted value of $\kappa_{\max}(\alpha)$ is not exact (RS is unstable). 97
5.13 Maximum training error along the geodesic path ($y = 2$) connecting the fBP solution with $\kappa_{\max}$ at a given α, with different solutions found by PA, SA and SGD. Non-zero energies along the path indicate disconnection in the solution space. Vertical dashed lines refer to dAT and LE transitions. The *inset* reports the stabilities associated with algorithms solutions. Red dashed line is the theoretical prediction for a typical solution with $\kappa_E = -0.5$. Reprinted from [119]. 98
5.14 (*Left*) Maximum barrier height along the geodesic path connecting two SGD solutions obtained with the cross-entropy loss as a function of α. For large system size (N), these solutions disconnect at the α_{LE} transition. The *inset* refers to the appearance of barriers between SGD and SA at α_{dAT}. (*Right*) How the energy barrier decreases for SGD as a function of epochs when initializing the algorithm to typical (i.e. SA) solutions. Reprinted from [119]. 98

6.1 Generalization error as a function of κ and with $\alpha = 5, \alpha_T = 50$ for different sampled solutions and numerical simulations. In *green* the typical test error of a solution of the error-counting loss at $\beta \to \infty$; SA with Hinge loss and $N = 1000$ is over that curve. In *orange* the generalization of a CE-loss solution with $\gamma = 3$ and norm fixed to 1; SGD with the same parameters and $N = 1000$ agrees with the theory (in dashed lin and purple dots the corresponding training error). In *red* the test error of the barycenter of typical solution at the given κ (BP with $N = 1000$ matches perfectly the theory). 104
6.2 Generalization errors as a function of κ for $\alpha = 5$ (*left*) and $\alpha = 0.5$ (*right*). For each subplot, the full curves correspond to the barycenter error, while the dashed lines to typical solutions of the Theta loss at that κ. Different values of the noise strength η are displayed. 106

6.3 (*left* panel) Train error as a function of κ for solution at different α sampled wit CE with fixed norm and $\gamma = 0.5$. (*right* panel) Test errors wrt to $\kappa' = 0$ of configurations at a given κ for different α values. Dashed lines refer to typical solutions, full curves to the barycenter of typical solutions, dotted-dashed to CE configurations. 107

6.4 Train (*left*) and test (*right*) errors for one-hidden multi-layer Perceptron with different H hidden neurons trained with $P = 10$K patterns of Fashion-MNIST with ADAM and CE loss for 100 epochs. Averages over 5 independent samples. L2 regularization $\lambda = 10^{-4}$. 107

6.5 Heatmap of train (*left*) and test (*right*) accuracy of the L2 norm Lagrange multiplier versus the margin κ for a MLP with one hidden layer ($H = 1000$) trained on two classes (bird, deer) of CIFAR10 with CE-loss and SAM [9]. 108

6.6 Training (*left*) and generalization (*right*) curves along the geodesic path connecting two solutions with margins $\kappa_1 = \kappa_2$ for $\alpha = 1$. The interpolation parameter is $\gamma \in [0, 1]$, and the predictions are derived using replica theory under the RS ansatz. 110

7.1 Modeling data: summation of single-pulse in the apical oblique dendrites of a CA1 pyramidal cell model shows a sigmoidally modulated within-branch summation (red circles) and linear between-branch summation (open green circles). Reprinted from [131]. 117

7.2 Representation of the single non-linear neuron model. The lower-left inset shows the functional form of the dendritic non-linearities à la Polsky; it consists of a combination of a ReLU activation and a sigmoid activation. 118

7.3 (*Top panel*) Pre-activation distributions of a given hidden neuron for different threshold values θ_d compared with the transfer function scale. *Inset*: critical capacities $\alpha_d(\theta_d)$ for different activation functions. The "identity" refers to the Perceptron case. (*Bottom panels*) Train errors for the TCM with fixed K hidden neurons using SGD (*left*-part) or LAL (*right*-part). Grey curves refer to the Perceptron case. 123

7.4 (*Top panels*) Speed of learning as a function of the epochs for the TCM compared with the Perceptron. Left-top subplots refer to LAL; right-top to SGD. Hyperparameters are specified in the legends. (*Bottom panels*) Average training errors and local energy profiles as a function of multiplicative Gaussian noise on the weights for the TCM with Polsky (orange lines) or the Perceptron case (blue lines). 125

7.5 Theoretical (*full black curve*) and experimentally observed distributions of weights for pyramidal cells in neocortex. Real data are taken from [142], and contain recording on about 1000 connected pairs (out of about 10000 tried). Coding levels for fitting are set to $f_{in} = f_{out} = 0.05$, while $\theta_d \sim 0.039$ such that $\overline{W} \sim 0.778$ and $\theta_s = 0.8$. 128

7.6 *Upper left*: weight distribution. *Upper right*: test error vs epochs (the curves end when the neuron has fitted the whole training set, i.e. the train error is zero). *Lower left*: robustness to input perturbations. *Lower right*: local energy. 129

8.1 (*left*) First part of the phase diagram (α_D, α) for the RFHM. The highlighted region refers to the phase where the features $\boldsymbol{f}_k$ are attractors. (*right*) Feature magnetization μ along a vertical cut of the phase diagram: increasing α the feature magnetization μ becomes different from zero with a first-order phase transition. The dashed line is the analytical prediction of the RS theory, while the markers are numerical experiments averaged over many samples for each value of α. The simulations are performed initializing the model to a feature $\mathbf{f}_k$, running the update rule (D.3), then measuring μ_k at convergence. Reprinted from [148]. 137

8.2 (*left*) Second part of the phase diagram (α_D, α) for the pattern recovery threshold. The critical capacity $\alpha_c(\alpha_D)$ decreases when the degree of correlation between patterns increases (α_D decreases accordingly). (*right*) Comparison with numerical results for the retrieval of one pattern. Each pixel represents the mean pattern magnetization for given values of α and α_D. The simulations are performed initializing the model to a pattern $\boldsymbol{\xi}_\nu$, running the update rule, then measuring m_ν at convergence. Reprinted from [148]. 138

8.3 Analysis of the discrepancy between simulations and theory in the case of the retrieval of one pattern. (*left*) If the temperature is low enough, the entropy becomes negative, signaling the incorrectness of the RS ansatz. As we lower α_D we see that the entropy becomes more negative, which is consistent with the RS solution progressively becoming a worse approximation of the numerical simulations. (*right*) Numerical check that the residual magnetizations goes to zero as $N \to \infty$ when the model is initialized in $\boldsymbol{\xi}_1$. Reprinted from [148]. 139

8.4 Completed phase diagram of the RFHM in the (α_D, α) plane. Three phases can be distinguished as we vary α and α_D: patterns are stable fixed points of the dynamics in the storage phase; features are stable fixed points in the learning phase; spin-glass dominated regime happens between the two. Reprinted from [148]. 140

8.5 (*left*) Magnetization as a function of α, for fixed α_D. Blue line is the magnetization μ of hidden features, which grows to 1 if α is high enough (*learning phase*). Orange line is the magnetization m^{train} of the train examples, which is $m^{\text{train}} \simeq 1$ for low α and drops when α increases, as expected from an associative memory (*storage phase*). Surprisingly, m^{train} grows to 1 again for high values of α. Near this transition, also test examples have $m^{\text{test}} = 1$, as shown by the red line (*generalization phase*). (*right*) Comparison between the phase diagram of a standard HM from [146] (*top*, temperature T vs α) and the phase diagram of the dense RFHM (*bottom*, α vs α_D). 141

8.6 (*left*) Feature magnetization μ (*dashed*) and test examples magnetization m^{test} (*solid*) as a function of α for different α_D. Dashed vertical lines are the analytical predictions from [148]. (*right*) Scaling with N of the maximum number of features D_{gen} for which we observe a generalization transition. Specifically, the maximum $D(N)$ at which 10 samples have $m^{\text{test}} > 0.9$. 142

A.1 Histograms of the distribution of stabilities for MCT0 solutions at $\alpha_T = 1.4$ (top left), $\alpha_T = 5$ (top right) and $\alpha_T = 10$ (bottom). The full line is the replica prediction of the distribution for typical solutions in the thermodynamic limit given by equation (A.33). Reprinted from [99]. . 179

B.1 (Left) Maximum fraction of errors along the geodesic path connecting two solutions obtained with PA at $\alpha = 1$ and $\kappa_E = 0.5$ using different learning rates at $N = 1000$. Dashed line is the reference barrier between typical solutions. (Right) Average overlap between PA solutions as a function of the learning rate. Horizontal dashed lines are q_{κ_E} and $q^\star_\infty$ at $\alpha = 1$. Reprinted from [119]. 195

B.2 Energy on the linear interpolation between two solutions with margin $k = \kappa_E = -0.5$ found by sampling with SA at various sizes and their extrapolations (full lines) at large N. The two colors refer to the case $\gamma = 0.5$ (red) and $\gamma = 0.2$ (blue). The horizontal dashed black lines represents the theoretical predictions. In the extrapolations we have fitted the last 5 points. Reprinted from [119]. 196

D.3.1 Comparison with numerical results for the retrieval of one feature (dashed theory). Note that, as we increase α, the finite-size effects become more pronounced. Simulations are performed initializing the model to a feature $\mathbf{f}_k$, running the update rule, then measuring μ_k at convergence for increasing values of N. From [148]. 214

D.3.2 Comparison with numerical results for the retrieval of one pattern (dashed theory). The discrepancy increases for smaller values of α_D. Simulations are performed initializing the model to a pattern $\boldsymbol{\xi}_\nu$, running the update rule, then measuring m_ν at convergence for increasing values of N. From [148]. 214

Introduction

In this Doctoral thesis we address the analytical and computational problem of learning with neural networks by making use of static approaches and bias techniques on sampling arising from the toolbox of statistical physics. The main topic, which is to clarify what prevents algorithms from finding solutions to certain non-convex optimization problems, will be devoted to a perspective all about the geometric characterization of solutions in the optimization landscape. The following work is divided into three parts.

- Part I: Preliminaries. The first part aims to introduce the reader to the midst of literature and debates about neural networks (NNs) emerging behaviours, clarifying that many of the variables chosen when training NNs can be studied in isolation to quantify their impact on the learning dynamics.
    - ◇ After formalizing the differences between supervised and unsupervised tasks (1.1), Sec. 1.2 provides an overview on some of the regimes for which supervised learning in a non-convex, high-dimensional optimization landscape can be facilitated: the role of initialization for the learning speed (1.2.1), the impact of overparameterization in locally convexifying the landscape (1.2.2), and a rigorous definition of algorithmic hardness from a geometrical perspective (1.2.3).
    - ◇ Chapter 2 stands with the literature supporting high local entropy solutions as the ones attractive for the non-equilibrium dynamics of local search algorithms, from which all the protocols derived in the following parts, and the conclusions drawn, take their inspiration. Section 2.1 reports the case study of a minimal neural network model for which the optimization is high-dimensional and non-convex: the Perceptron with binary weights. The goal is to explain, using approaches borrowed from statistical physics of disordered systems, how it is possible in the typical case to characterize learning properties of simple models and how, depending on the probability measure from which solutions are sampled, one does or does not derive meaningful theoretical predictions compared to algorithmic results. Section 2.1.2 proposes an adequate measure to sample from the landscape attractive solutions for the algorithm dynamics; starting from that, section 2.2 illustrates how it is possible to optimize solvers to target high local entropy (i.e. dense clusters) configurations due to the correlation between flatness and performance of the corresponding minimizer.

- Part II: the benefits of robustness. Once it has been clarified that the non-trivial landscape where the learning process occurs is characterized by minima associated with different performance, and specifically that flat solutions are not only dynamically attractive but also robust to perturbations and performant for generalization, the second part of the manuscript focuses on introducing protocols to observe the structural change of these relevant configurations as the complexity of the learning task increases.

  - ◇ In Chapter 3 and for two models of networks with binary weights, the Perceptron and the Tree Committee machine, minima on randomly generated supervised tasks are characterized as configurations with a certain level of robustness (i.e., margin) and the first results on hierarchical organization of clusters as solutions arising from the coalescence of robust configurations is derived (sec.s 3.1 and 3.2). Then, the analysis specializes on an analytical method, entirely based on local geometry of minimizers and independent from the specific algorithmic strategy, to identify the constraint density threshold (i.e. proportional to the complexity of the tasks and the density of patterns in the dataset) that causes clusters to break and prevents algorithms from converging (3.1.2).

  - ◇ Chapter 4 adapts all the techniques in 3 to a mismatched teacher-student problem where the correlation among patterns visible to the student comes from a low-dimensional manifold of data. In Sec. 4.1, which defines the models, all the assumptions, which allows for the use of the Gaussian Equivalence Principle (4.1.1), are introduced in order to study the typical case student performance. In following sections, the geometry of minima of the learning tasks are studied by inspecting the fallout of overparameterization, keeping track of possible phase transitions when changing the expressivity of the student compared to the teacher model. As expected, prediction performances of the student get better when increasing the overparameterization, and a clear relationship between the robustness of a minimizer and the associated generalization is reported (4.2.2). As in 3, it is possible to derive a threshold for algorithmic performance degradation with the disappearance of robust solutions from the landscape, showing how efficient solvers cannot overcome the conjectured hard phase (4.2.3). The chapter ends with a digression on deeper models, supporting evidences for which, despite the overparameterization of bigger architectures, the optimization landscape still remains rough and non-convex, being the scope that of finding minima with an average non-trivial overlap among them (4.3).

  - ◇ Chapter 5 reports the first analytical study on the energetic barriers along the geodesic path connecting minimizers sampled independently from the equilibrium measure with different degrees of robustness. The model is a non-convex one-layer NN with continuous weights, i.e. the spherical negative Perceptron (5.1), and the goal of studying the training error on the

convex envelope spanned by y different solutions is that of deriving algorithmic implications of simple connectivity properties (5.5) and to characterize the geometry of the space of minima (5.3) more in detail with respect to 3 and 4. The existence in theory of a large geodesically convex component for sufficiently high degree of overparameterization and among solutions with high margin (5.3) reveals how it is possible that convex optimization strategies find in practice linearly-mode-connected solutions, and give rise to a star-shaped organization of the zero-energy manifold, where for sufficiently low density of constraints, even dominant configurations of a flat probability measure are linearly connected to a convex cluster of minima.

- ◇ Chapter 6 refers to a work in preparation that uses hierarchical space organization of solutions with different margins to think of a practical application of averaging methods in non-convex optimization landscapes. The sections, each of them devoted to report evidences of how it is possible to play with the landscape structure to take advantage of negative margin solutions, explain how sometimes, given an optimization problem with some level of noise in the dataset, it is worthy to relax some constraints in the learning process, since averaging over under-performing model has the effect of targeting a regularized robust and well-performing barycenter in the end. Section 6.3 extends the analysis to a simple analytical proof that the generalization error along the geodesic between negative margin solutions shows a minimum in the middle of the path, which one has reason to think corresponds to an atypical robust solution.

- Part III: neuroscience comes into play. The present part of the manuscript talks about two simple examples of biological models to address questions in neuroscience through the use of optimization-like strategies. What is the memory capacity of a single neuron of a pyramidal cell (7)? And what if one can construct an associative memory model that, instead of simply storing prototypes i.i.d. patterns, plays with correlated inputs (8)?

  - ◇ Chapter 7 is a work in preparation that proposes use of a Tree Committee Machine with positive weights to model the activation scheme of an excitatory pyramidal cell that includes dendritic non-linearities. For the model, both theoretical critical capacity (7.2.1) and maximal algorithmic storage load (7.2.2) are derived, discussing how the simple addition of a non-linear activity layer makes the model faster to learn, more robust to noise, and with better prediction performances (7.4) with respect to its linear counterpart.

  - ◇ Chapter 8 is an extension of the Hopfield model with correlated patterns, i.e. the Random Feature Hopfield. Compared to the standard version, when trying to use the Hebb rule for the coupling synaptic matrix to store as prototypes not i.i.d. but correlated inputs, different phases appears in the usual phase diagram: a learning phase where the features generating

the inputs become attractive for the zero-temperature asynchronous dynamics (8.1), and a generalization phase (8.2) where mixtures of features let the model have new fixed point attractors related to patterns that it has never seen before.

- Part IV: conclusions. In this final Chapter we will discuss the results obtained and we will highlight several possible directions for future works.

Part I

Preliminaries

Chapter 1

Emerging behaviors in neural networks

The success of deep neural networks (DNNs), now far beyond data-driven research, is still a mystery to theoretical science. Indeed, the complexity of models presents an obstacle to interpretability and although statistical mechanics has provided valuable support and analytical methods to avoid handling them as "black boxes", heuristics still remain at the heart of the way neural networks are trained today. Trial and error processes are largely driven by empirical insights and still many of the fundamental questions remain open questions for the field. This chapter aims to provide insights into emerging network behaviours and related lines of research.

1.1 Basics on self-/un-/supervised learning

In the realm of machine learning self-/un-/supervised learning represent the primary paradigms that differ significantly in terms of their objectives, training data requirements and application domains.

- Image classification, for instance, is a classic example of *supervised learning*. Indeed, the main ingredient of the branch is the presence of labeled training data, wherein each input instance is paired with a corresponding output label. In that context, the network aims to learn a functional mapping $f : \{\boldsymbol{\xi}^\mu\}_{\mu=1}^P \to \{\sigma^\mu\}_{\mu=1}^P$ to correctly assign the right labels σ^μ with each input $\boldsymbol{\xi}^\mu$ of the P-dimensional training set $\mathcal{D}$. In the synthetic case, the finite data set $\mathcal{D}$ is built by sampling independently identically distributed (i.i.d.) examples from $\mathcal{P}$, the usually non-accessible distribution of data. Assuming that the feed-forward dynamics of a network with D layers and N_l neurons for each is specified by D weight matrices $\boldsymbol{W}^1, \dots, \boldsymbol{W}^D$ of the form $W^l \in \mathbb{R}^{N_l \times N_{l-1}}$, and a number D of activity neurons $\boldsymbol{x}^1, \dots, \boldsymbol{x}^D$ for each layer, we have

$$\boldsymbol{x}^l = \phi\left(\boldsymbol{W}^l \boldsymbol{x}^{l-1}\right) \quad \text{for } l = 1, \dots, D \tag{1.1}$$

  the network rule to process the activity pattern from one layer to the next. In Eq. (1.1), $\phi(x)$ is a generic non-linear function acting element-wise and we

identified $\boldsymbol{x}^0 \equiv \boldsymbol{\xi}$ and $x^D \equiv y$ the input and output respectively. To learn the map f, the game is to optimize the network weights by minimizing a well-suited loss function $\mathcal{L}(\sigma^\mu, y^\mu)$ that penalizes any discrepancy between the correct output σ^μ and the network prediction y^μ. The goal is to find the set of weights $\boldsymbol{w}$ to minimize the training error

$$\epsilon_t = \frac{1}{P}\sum_{\mu=1}^{P} \mathcal{L}(\sigma^\mu, y^\mu) \tag{1.2}$$

hoping that this is in principle sufficient to correctly classify the label of previously unseen examples, being able to generalize. The ability to generalize largely defines the performance of a network, as minimizing the generalization error

$$\epsilon_g = \langle \mathbb{E}_{\boldsymbol{\xi}^\star} \mathcal{L}(\sigma^\star, y^\star(\boldsymbol{w}, \boldsymbol{\xi}^\star)) \rangle \tag{1.3}$$

i.e. the average number of errors on a new example $(\boldsymbol{\xi}^\star, \sigma^\star)$ (denoting by $\langle\cdot\rangle$ the ensemble average), is evidence that the network has learned hidden features from the data, understanding the emerging differences between classes.

- *Unsupervised learning*, in contrast, deals with data that has not been labeled. The primary objective is to discover inherent structures or patterns within the data without the need of pre-classifying them, an expensive procedure if the amount of data is large, the expertise requirement gives a financial constraint (e.g. medical imaging, scientific research) or when data are continuously changing in dynamical environments and so previously accurate labels may become obsolete quickly.

  Let now $\mathcal{D} = \{\boldsymbol{\xi}^\mu\}_{\mu=1}^P$ be the usual dataset consisting of P samples with the empirical data distribution $q(\boldsymbol{\xi})$. Unsupervised learning strives to fine-tune the parameters $\boldsymbol{w}$ of a network of the type in Eq. (1.1) within a family of probability distributions $p(\boldsymbol{\xi}; \boldsymbol{w})$ so that they closely mimic the empirical distribution $q(\boldsymbol{\xi})$. In order to do that, one common approach is to maximize the log-likelihood of the dataset concerning the parameters $\boldsymbol{w}$

$$\mathcal{L}(\boldsymbol{w}) = \int d\boldsymbol{\xi}\, q(\boldsymbol{\xi}) \log p(\boldsymbol{\xi}; \boldsymbol{w}). \tag{1.4}$$

  This optimization aims to adapt p to allocate higher probabilities to the observed data points, subsequently narrowing the gap between it and $q(\boldsymbol{\xi})$. Inversely to classical problems in equilibrium statistical mechanics when one begins with a known Boltzmann distribution $p(\boldsymbol{\xi}; \boldsymbol{w})$ and from that computes macroscopic properties, here we sample $\boldsymbol{\xi}$ with the objective of discovering $p(\boldsymbol{\xi}; \boldsymbol{w})$. Once a well-fitting model $p(\boldsymbol{\xi}; \boldsymbol{w})$ is established, it opens the door for a wide spectrum of applications: to natural language processing (NLP) for topic modeling or word embedding (to convert words into vectors, capturing semantic meanings), to signal processing (e.g. separating a mixture of audio signals into its component sources), data clustering, or to image and video recognition for feature learning and object segmentation.

- *Self-supervised learning* is the subfield that sits at the intersection between supervised and unsupervised learning. In self-supervised learning, labels are automatically generated from the data itself, without the need for explicit external labels. In the context, one often considers data points $\{\boldsymbol{\xi}\}_{\mu=1}^{P}$ and a transformation $\mathcal{T}$ that generates a target $\sigma = \mathcal{T}(\boldsymbol{\xi})$. The goal during training is formalized as that of learning for the network the feature representation $f_{\boldsymbol{w}}(\boldsymbol{\xi})$ that predicts σ from $\boldsymbol{\xi}$. These representations are often task-agnostic and can be fine-tuned for various downstream tasks, which may or may not be related to the original transformations $\mathcal{T}$. So, while in unsupervised learning the goal is usually to learn the inherent structure of the data and in supervised learning models typically generalize within the scope of the task they were trained for (i.e. sampling unseen data points for testing generalization from the same probability distribution as the training set points), the feature representation as the product of self-supervised protocols can be used as a starting point for a spectrum of tasks that may require subsequent fine-tuning.

1.2 Fundamental questions in the theory of Deep Learning

Once the main branches of machine learning have been clarified, it should be clear that throughout the following discussion, the main focus will be on problems related to supervised learning, and to a large extent easily generalizable to the other branches. When dealing with neural networks (NNs) to solve a specific learning task, all the choices concerning the architecture of the network [1], its initialization [2, 3], the choice of the loss function to be optimized [4–6] and the optimization algorithm itself [7–9] are crucial to the final performance. It is of common knowledge that deep neural networks (DNNs) today with order of billions of parameters outperform their shallow cousins but a unified theoretical framework to justify this does not yet exist. All the steps taken in this direction are based on the following intuitions: deep nets generally implement higher complex functions over the input space compared to single-hidden-layer shallow networks with an equivalent number of neurons [10–12], thus leading to higher expressivity. In classification tasks wherein if $f : \mathbb{R}^N \to \{C_1, \dots, C_k\}$ is the function that maps a point in the N-dimensional space to one of the k discrete output classes $\{C_1, \dots, C_k\}$, the goal is to find through the network the decision boundary $f(\boldsymbol{\xi}; \boldsymbol{w})$ that can partitions the input space into regions (each corresponding to distinct classes), deep networks have the capability to disentangle the decision-making process for complex tasks [13]. This lies in the ability to transform curved decision boundaries in the input space into more straightforward linear decision boundaries in the hidden layers, supporting the idea of hierarchical learning of features [14] and promoting efficiency (linear classifiers are usually quicker to train) and generalization (a simpler decision boundary is often a sign that the model has captured the underlying data distribution well). But despite their expressive power, DNNs must be enabled to perform. Clearly, as anticipated, all the other choices concerning the cost function or

the initialization of the algorithms (just to name a few) have a non-trivial impact on the learning dynamics and can be studied in isolation to optimize performances. As an early example of this, initialization of the network parameters significantly influences the speed of the training dynamics, not only for vanishing/exploding gradient effects who can hamper the learning process [15–18], but also because an unfortunate initialization is related to improper positioning of the network in the loss landscape [3]. Indeed, the high dimension and multi-equilibrium properties of the landscape in which the optimization dynamics moves, make it full of different minima that can be labeled according to the associated network performance: good minima, bad minima, rare states and saddles. Setting aside the precise definition for each, it is sufficient to understand for now that what distinguishes a good minimum from others is the presence of a state where not only the function value (i.e. the loss) is low, but models that reside in these good minima are able to generalize on out-of-sample data. Good minima prevent overfitting to the training data, ensuring they focus on the core data distribution instead of its noise or peculiarities. Hence, it would be worthy for a good initialization to select a region in the landscape that both speeds up convergence and is intrinsically close to a well-performing minimizer. In this regard, there is a body of literature [3, 19–21] including those concerning deep linear networks (DLNs), that formalizes how, depending on the size of the initialization, there is a phase transition in which either the network lies inside a narrow valley that brings it quickly to a global minimum or near a saddle where the dynamics will be slower, since much of the training time will be wasted running away from initialization on the saddle. But how much time? In such high-dimensional non-convex landscapes, is not just the presence of saddles but the roughness itself is an obstacle for convergence, since an exponential number of local poorly generalizing minima [22, 23] can themselves potentially trap the dynamics for long times [24]. At least, this is what we have learnt from statistical physics by studying spin glass systems [25, 26], being the loss landscape of these models equally complex in nature [27], i.e. high-dimensional because of many particles (spins) interacting together and non-convex when considering disordered systems. Nevertheless, the trapping problem in bad minima was empirically solved through the evidence that as long as your network is *big* enough, a good initialization combined with gradient descent (GD) dynamics works well in the domain [28–30]. Why there are no issues with these strategies, originally designed for convex optimizations, and how overparameterizing the network (i.e. increasing the network dimension with respect to the data points) has the effect of smoothing the landscape without stumbling into overfitting, are other crucial questions in debating about DNNs efficiency. From one side, it seems that the implicit regularization of networks trained when using gradient-based methods leads to optimization of low-rank matrices of weights [31–34]. This can be seen as an implicit bias towards simpler, more structured solutions that offer good generalization and no-overfitting since they rely on few, more salient features; or as a *grokking* phenomenon, where networks engage in late-time rich feature learning after the training loss is already low because they operate in a simpler, more constrained manner in the early lazy training [35]. By analyzing the spectrum of the Hessian at convergence, a significant fraction of eigenvalues are close to zero [36, 37], implying that the effective dimensionality during GD optimization is lower

than the full parameter space [38]. It can only be explained by the presence of nearly convex regions in the loss landscape that seem to be attractive for gradient-based (e.g. SGD) dynamics. As an evidence of this, in [39, 40] the authors show that different SGD runs end up in the same basin of attraction by explicitly finding a path at zero training error connecting different minimizers [41].

So, it is not an heuristic that, despite the landscape per se should transition from being almost convex to being highly chaotic when the networks become sufficiently deep, the dynamics continue to converge. As we mentioned, the implicit regularization bias of SGD [42], but also batch normalization [18], the use of skip/residual connections [43] or dropout [44] are all things that promote flat minimizers in the landscape and prevent the transition to chaotic regimes. Indeed, an idea that will be the backbone of this entire manuscript, it is the geometry of neural minimizers that affects generalization properties. Meaning, it is the existence of highly surrounded mimima themselves that impacts the learning process, suggesting that combining energy minimization with flatness maximization during the training dynamics leads to highly performing protocols [45]. It is certainly intuitive that when thinking of modifying learning strategies in a given loss landscape, it is the entropy of good (and flat) minima that contributes to whether or not the strategy will be successful. In this regard, a broad spectrum of literature [8, 40, 45–49] point out how usually good minima in learning tasks are rare states, i.e. subdominant with respect to the uniform measure over all possible solutions for a given problem. This makes the task of finding them from scratch (without leveraging methods that inherently or explicitly favor flat regions) highly non-trivial. However, pruning methods [50–53] that involve iterative training and consistent model size reduction, have proven effective in detecting optimal minima in an agnostic way. The idea of pruning is indeed that of implementing regularization effects "by hand": the model is trained on a given tasks, at the end of the procedure the smallest weights are pruned, and the new pruned network is then rewound to be trained again and again, iteratively. The protocol, better known as *iterative magnitude pruning* (IMP) [54], it is different from cutting the weights without going to training, since in the latter the model will just reduce its expressivity by losing any benefit. What pruning techniques are instead doing is denoising, meaning they are progressively killing useless directions in the training, thus leaving just the weights that match with the data structure. It is somewhat like, because of many variables, what gradient-based methods are able to do given that they optimize low-rank matrices of weights.

Up to this point, it is evident there is a redundant interplay between data structure, network dimension, implicit dynamic bias and geometric properties of minimizers. Yet, the need to push for increasingly highly expressive models with control over minima local geometry, has led to use networks in regimes where, since the number of neurons far exceeds the sample size, the networks themselves become kernel machines [55], with a deterministic dynamics under certain conditions that is looking for one global minimum in a convex-like optimization setting. In fact, if one studies the dynamics of a network in this regime, it is easy to realize that the internal representations remains static (i.e. fixed at initialization) and the learning is lazy, implying that by the end of the dynamics, the network weights will have deviated only slightly from

their initial values [56, 57] and that the network is not actually exploiting any feature extraction capabilities from the data. Even when dealing with NNs in these regimes where the dynamics has *trivialized*, the functional approximation implemented by the network can change due to the sample size. In [58, 59] the authors show how, by increasing the sample size, the NN is going to fit first the top eigenvalues of the target functions and then the others, thus explaining why empirically for small sample sizes infinitely wide NNs seem to have a bias towards *smooth* functional approximation despite the huge amount of parameters.

A big step forward to bridge the gap in understanding between theoretical models, treated as kernel machines when the width is large, and practical applications in deep learning, where feature evolution remains the main ingredient [60], has been done in a series of work [61–63], where the authors develop path integral formulation of gradient flow dynamics or for biologically-plausible alternatives in infinite-width networks to track dynamical evolution of hidden layer activation distribution, of the neural tangent kernel (NTK), and output predictions as the rich feature learning goes on.

Having summarized some of the highly debated topics, it is clear that it is necessary to line up ideas and that there is still much to understand, since many of the variables involved are related by non-trivial relations. In the following sections, we will delve into the details of some aspects, among which those that are most relevant to support a geometric characterization of the landscape as a valid alternative to understand many of the insights on how the good minima form and why networks manage to converge to them, the main topic of this entire discussion.

1.2.1 The role of initialization

In order to give a more clearly defined intuition concerning the role of initialization in shaping the training dynamics, we will briefly review the results in [3]. We will see how, when playing with initialization size in large-width deep linear networks (DLNs), two different regimes can be distinguished: a neural tangent kernel (NTK) type of regime that is characterized by a loss landscape in which the dynamics converges fast to a global minimum, shifting little the weights from initialization [57, 64], and a saddle-to-saddle dynamics type of regime, where the dynamics is slow since the network moves close to a sequence of saddles. We will intentionally remain very generic in our treatment, since the goal is not to delve into the details of the theoretical proof of the results, but rather to provide an example of how the geometry of the landscape affects the learning speed. For the sake of clarity, let us consider a regression problem with a training set size of dimension P composed by a set of input patterns $\{\boldsymbol{\xi}\}_{\mu=1}^{P} = \{\xi_1, \dots, \xi_{n_0}\}_{\mu=1}^{P} \in \mathbb{R}^{n_0 \times P}$ and output $\{\boldsymbol{\sigma}\}_{\mu=1}^{P} = \{\sigma_1, \dots, \sigma_{n_L}\}_{\mu=1}^{P} \in \mathbb{R}^{n_L \times P}$, being n_0 and n_L the dimensions of the input and output respectively. So, if the network has $\ell = 0, \dots, L$ layers, each containing n_ℓ neurons, the connection weights will be matrices $\boldsymbol{W}_\ell \in \mathbb{R}^{n_{l-1} \times n_l}$ and the network function will map any input $\boldsymbol{\xi} \in \mathbb{R}^{n_0}$ to

$$f_{\boldsymbol{w}}(\boldsymbol{\xi}) = (\boldsymbol{W}_L \circ \sigma \circ \dots \circ \sigma \circ \boldsymbol{W}_1)(\boldsymbol{\xi}). \tag{1.5}$$

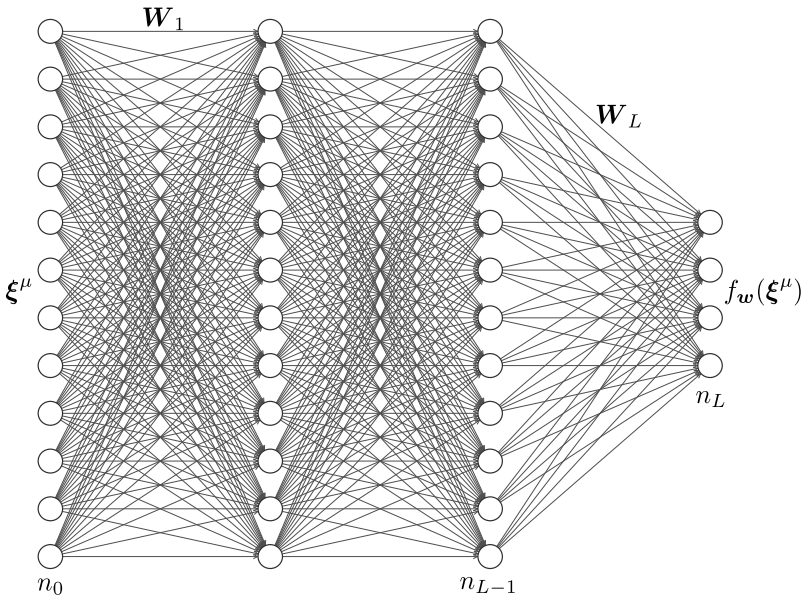


Figure 1.1: Rectangular deep network with L layers and n_l neurons for each. Every weight matrix is a matrix $\boldsymbol{W}_\ell \in \mathbb{R}^{n_{l-1}\times n_l}$ and given an input $\boldsymbol{\xi}^\mu$, the network implements the function $f_{\boldsymbol{w}}(\boldsymbol{\xi}^\mu)$ until the output.

The network will be rectangular if $n_0 = \ldots = n_{L-1} = w$, so when the number of hidden neurons in each layer is the same (see fig. 1.1), and we will consider the infinite width limit $w \to \infty$ for an infinitely wide neural networks with L layers and parameters $\theta = \{\boldsymbol{W}_1, \ldots, \boldsymbol{W}_L\}$. The mapping is linear $\boldsymbol{A}_{\boldsymbol{w}} = \boldsymbol{W}_L \ldots \boldsymbol{W}_1$ if the network is linear and so σ, which would represent a generic activation function, is the identity. Plus, if the loss function is the usual mean squared error loss

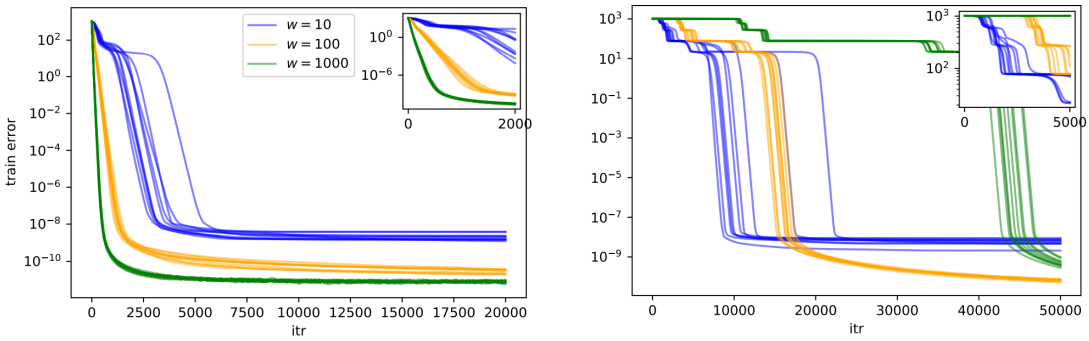
$$\mathcal{L} = \frac{1}{P}\sum_{\mu=1}^{P} (f_{\boldsymbol{w}}(\boldsymbol{\xi}^\mu) - \sigma^\mu)^2 \tag{1.6}$$

it will measure for each sample the average of the squares of the errors between the predicted and actual values. It is usually desirable that just picking a random initialization, a gradient descent dynamics is sufficient to optimize that loss by finding the global minimum. This is highly non-obvious since the landscape of NNs when the width is very large has a *non-generic* non-convexity: characterized by a manifold of global minima in high dimension (i.e. $\equiv$ #parameters $-$ #data points), a high-dimensional manifold of saddles, and non-strict saddles where the Hessian has no negative eigenvalues. Now, depending on how the network is initialized, you will see a phase transition in which for small initialization the network lies inside a flat region, while for large initialization the network starts from a narrow valley that brings it to a completely different dynamics. If we consider the linear network randomly initialized as $\theta_0 \sim \mathcal{N}(0, w^{-\gamma})$, where γ is the scale of the variance, then $\gamma \leq 1$ will represent the *large* initialization, since for very large w, the variance grows up for small γ; $\gamma \geq 1$ instead will be the *small* initialization, since the variance goes to zero very fast if γ is large. The two quantities of interest in understanding if, based on γ, the network will

be in a global minima dominated regime or in a saddle dominated regime, are both the distance d_{min} to the closest global minimum and the distance d_{saddle} to the closest saddle. It can be demonstrated that the following scaling laws hold in the $w \to \infty$ regime

- $\gamma \leq 1 : d_{\text{min}} \sim w^{-\frac{(1-\gamma)(1-L)}{2}} \quad \text{and} \quad d_{\text{saddle}} \sim w^{\frac{1-\gamma}{2}}$.
- $\gamma \geq 1 : d_{\text{min}} \sim 1 \quad \text{and} \quad d_{\text{saddle}} \sim w^{-\frac{\gamma-1}{2}}$

meaning that, for small γ the distance to the closest global minimum goes to zero as the width grows because of the negative exponent, whereas the distance to the saddle increases with width. On the other hand, if γ is large the distance to the closest global minimum remains roughly constant as the width grows up, but instead you will initialize closer and closer to a saddle. In support of this, as can be seen from



(a) $\gamma = 0.75$ (NTK) **(c)** $\gamma = 1.5$ (S-S)

Figure 1.2: Train error as a function of the iteration number for a deep linear network with $L = 4$ layers and different values of the width w. In the NTK regime (*left*), the training loss shows typical linear convergence behaviour at large width; in the saddle-to-saddle regime (*right*), the dynamics is slower and the plateaus become longer as the width grows. Reprinted from [3].

Fig. 1.2, for the same learning problem (i.e. with the same loss) but with different initializations, the dynamics will either be fast (i.e. the train error is a monotonic linearly decreasing function of the iteration number) or slow, starting with a plateau that corresponds to the time needed to escape the saddle. In [3] the authors also demonstrate how the first regime corresponds to the NTK regime under gradient flow dynamics, meaning the regime in which NNs behave like kernel machines with a fixed internal feature representation that depends on the asymptotic dynamics in the infinite width limit. Here, as we mentioned, the dynamics is deterministic and the evolution of the network function is described by the linear differential equation

$$\partial_t f_\theta(\boldsymbol{\xi}) \propto \frac{1}{P} \sum_{\mu=1}^{P} \boldsymbol{K}_{\text{NTK}}(\boldsymbol{\xi}, \boldsymbol{\xi}^\mu) \left(\sigma^\mu - f_\theta(\boldsymbol{\xi}^\mu)\right). \tag{1.7}$$

The expression involves the kernel $\boldsymbol{K}_{\mathrm{NTK}}(\boldsymbol{\xi}, \boldsymbol{\xi}^\mu) = \nabla_\theta f(\boldsymbol{\xi}; \theta_0) \cdot \nabla_\theta f(\boldsymbol{\xi}^\mu; \theta_0)$ which is the scalar product of the gradient of the output of the network at different training data points with respect to the parameters. It is simply a measure on how much the training point $\boldsymbol{\xi}^\mu$ is going to affect the functional map at another point $\boldsymbol{\xi}$. The reason why the NTK is deterministic (as you can see from the definition it only depends on the parameters at initialization θ_0), is that in the regime the network converges to a very close minimum, and because the dynamics ensures that the NN moves very little, there is a small change in the parameters. One can easily solve eq. (1.7) and get an expression for the network function at any time t.

The other extreme regime in which $\gamma \to \infty$ belongs to a vanishing initialization and precisely if $\gamma = \infty$, then looking at the GF dynamics tells you that nothing happens (i.e. the weights do not change), meaning: the closer you are to a saddle the longer it takes to escape. Instead, in all intermediate regime that correspond to small initialization, Fig. 1.2 shows you how after the first escape time (i.e. the number of iterations before escaping the first plateau), because the saddles end up being in an invariant region where the rank of the matrix representing the network is 1, the network will visit a sequence of saddles of increasing rank before reaching the global minimum. This gives reason to the multiple plateau in Fig. 1.2 and to a slower dynamics compared to the large initialization case.

1.2.2 Avoiding glassy dynamics

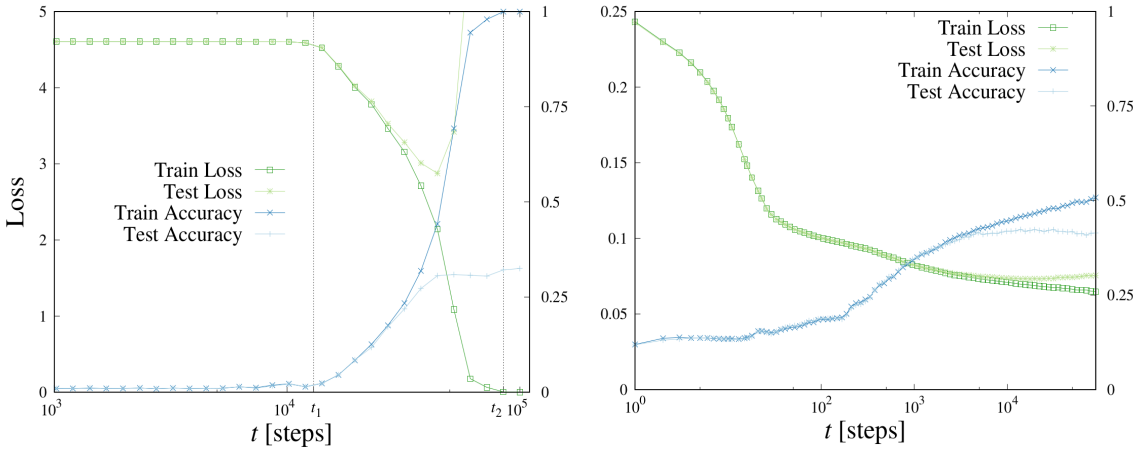
In proceeding to argue how even a single variable taken in isolation can have an impact on changing the learning landscape, the authors of [24] and [65] focused on formalizing similarities and differences in the dynamics of neural networks compared to spin glasses. In this brief section, we deal with some of their results concerning the role of the network dimension with respect to the number of data points.

When training a NN with standard protocols, the objective is to minimize a loss function that plays the role of a dataset-dependent energy, while the weights are the degrees of freedom for the problem. The randomness in the data represent the *quenched disorder* for the physics perspective, since this randomness do not evolve within the experiment timescale. Starting from a reasonable initialization (we discussed it in Sec. 1.2.1), the training is routinely performed using Stochastic Gradient Descent (SGD) for the majority of applications, where, apart from noise due to the random selection of mini-batches, the weights are constantly adjusted by moving to the direction in which the loss has the steepest rate of decrease. The protocol is then similar to what in physics is called a *quench*, which refers to a rapid cooling of a material from high-temperature state to low-temperature. This process is carried out so quickly that the system does not have enough time to reach thermal equilibrium at intermediate steps; instead, it becomes *frozen* into a disordered state that depends on the configuration it had in the high-temperature phase. In the same way, the final parameters of a NN trained with SGD depend both on the sequences of mini-batches used and on the initialization. So, it is perhaps intuitive how if a parallelism in models constructions, energy landscapes and dynamic protocols can be made between spin glasses and NNs, then it is worthy to understand whether or not they also share sim-

ilar properties. Precisely, spin glasses are known to display glassy dynamics after a quench, meaning the system gets stuck for long times in local minima [66–68]. Indeed, as the temperature decreases, the relaxation time (i.e. the time it takes for the spins to relax back to the state of equilibrium) increases dramatically, growing faster than an exponential function of the inverse of temperature. Plus, *aging* dynamics comes into play, meaning that the system behaviour at a given time does not just depend on the current state, but on its history, specifically how long it has been since it was quenched into a glassy state. As a consequence, spin glasses are usually not ergodic, not being able to decorrelate from their initial conditions.

In a similar way as it was done for spin glass systems, one can try to probe the training dynamics of NNs through the measurement of one and two-point correlation functions from which properties of the loss landscapes can be inferred. In studying the dynamics of different NN models, the authors of [24] find that, regarding the dimension of the network with respect to the data points, a phase transition can be distinguished between two regimes: an *easy* phase for overparameterized neural networks (more parameters than data points), where even if after an initial exploration of the energy/loss landscape the system displays glassy dynamics, it is able in the end to reach a well-performing global minimum contrarely to what would happen in spin glass where at most the system would get stuck in a local minimum at high energy. A *hard* phase corresponds to underparameterized NNs (more data point than parameters) in which instead the dynamics is purely glassy and the networks are not able to interpolate the dataset.

As anticipated in Sec. 1.2, many conjectures are put forward as to why DNNs with a huge number of parameters are able to converge quickly in a highly rough and non-convex landscape without overfitting: one is that there exist rare and wide minima that have large basins of attraction and can be reached without any substantial barrier crossing by the training dynamics [45]. The other instead suggests that, if the systems are sufficiently overparameterized, despite the landscape itself being non-convex, the dynamics move in a lower dimension exploring nearly connected components. Hence there are no spurious minima that can trap the dynamics in this regime [69, 70]. Even though it is reasonable that it is a combination of both factors that deflects dynamics from glassy behaviour, empirical analysis shows that overparameterization in NNs has the effect of stretching the rough landscape by making the dynamic exploration simpler. For instance, looking at Fig. 1.3 from [24], one can see both train and test loss (and accuracy for completeness) as a function of the logarithm of time (measured in units of iterations) for an overparameterized system (on the left) and an underparameterized one (on the right). It is clear how for the ResNet-18, the first dynamical step consists in exploring the landscape up to a time t_1 where the loss and accuracy stay roughly constant. Then, for $t_1 < t < t_2$ the loss starts decreasing quite linearly as a function of $log(t)$, that is the time during the dynamics where the mean-square displacement of weights $\Delta(t_w, t_w+t) = \frac{1}{N}\sum_{i=1}^{N}(w_i(t_w) - w_i(t_w+t))^2$ in a time interval t after the time laps t_w of the quench, develops a dependence on t_w, thus showing aging. Indeed, this time interval during training is the time where the longer is t_w, the longer it takes for the system to diffuse, even if the characteristic plateau in spin glasses that quantifies how much the system is frozen into a local minimum



ResNet-18 on CIFAR-100, $B = 64$, $\alpha = 0.01$.

Loss of the under-parametrized model.

Figure 1.3: Train and test losses of the negative log-likelihood (or accuracy) as a function of $\log(t)$ for a ResNet-18 on CIFAR100 (*left*) with batch size B and learning rate α. On the *right*, a one-hidden-layer NN with 10 hidden neurons trained with MSE on CIFAR-10. The batch size is $B = 100$ and the learning rate is $\alpha = 0.1$. In both cases the training is carried out by SGD. Reprinted from [24].

is not well reproduced for DNNs (see details in [24]). This goes in the direction of supporting that it is not the barrier crossing phenomenon that slows down learning, but instead it is the presence of an increasingly large number of flat directions that become available to the system during its descent. Finally, for $t > t_2$, the system reaches a phase where the loss is almost zero, suggesting that it is diffusing close to the bottom of the landscape. The underparameterized case has instead an unfortunate dynamics (Fig. 1.3, *right* panel), where the loss does not reach its minimum value because the system gets stuck asymptotically in a local minimum at high energy.

One can also imagine to study how the most relevant aspect of learning, meaning the generalization error on unseen samples behaves as a function of the network size as the authors did in [65], by making comparison between the *double-descent* in NNs with the *jamming* transition in glasses. In general, once taken a classification problem where a given network with N parameters $\boldsymbol{W}$ has to learn P input-output associations $\{\boldsymbol{\xi}^\mu, \sigma^\mu\}_{\mu=1}^P$ by minimizing a loss function $\mathcal{L}(\boldsymbol{W})$, it is possible to count the *unsatisfied constraints*, meaning the number of incorrectly classified patterns by looking at $\Delta^\mu \equiv 1 - \sigma^\mu f(\boldsymbol{\xi}^\mu; \boldsymbol{W})$ if the problem consists in binary classification, so $\sigma^\mu = \pm 1$ and it is only the sign of the network output $f(\boldsymbol{\xi}^\mu; \boldsymbol{W})$ that counts. Traditional statistical learning theory would suggest that, if the model complexity is not sufficient to capture the underlying data distribution, meaning the model has too few parameters $N < P$, then both training and test (generalization) errors will be high. As more parameters are added, the model starts to fit the training data better and the number of unsatisfied constraints $\Delta^\mu \to 0$. But after a certain point, the model should begin to overfit, since it will capture not only the true signal but also the noise in the training data, causing generalization error to rise even as training error

continues to decrease. However, once the number of parameters surpasses the number of training samples ($N > P$), something interesting happens. Instead of continuing to rise, the test error begins to decrease again, even if the number of parameters grows far beyond the number of training data points by defying classical bias-variance trade-off expectations. This gives rise to the U-shaped curve of the *double-descent* phenomenon first observed for DNNs in [20, 71], often due to implicit regularization effects during training dynamics that lower the effective dimension of parameters [44, 72], together with symmetries present in the network. It is also a confirmation that the model is so flexible that it can effectively average over many different function; that is, it does not commit too strongly to any single noise-dependent hypothesis, which could lead to overfitting.

In relation to spin glasses, much has been understood about double-descent, particularly the cusp in generalization before the second descent, by relating it to the *jamming* transition in the hard sphere packing problem [73–78]. Briefly, suppose we have spheres of radius R in positions $\{r_i\}$, for a total number of degrees of freedom N. We denote with $r_{ij} = ||r_i - r_j||$ the distance between particles i and j and define their overlap as $\Delta_{ij} = 2R - r_{ij}$. As a consequence, two particles will be in contact if $\Delta_{ij} > 0$. The goal here is to arrange the spheres within a given volume in a way that maximizes the packing density — the proportion of the volume that is occupied by the particles — while ensuring that the spheres do not overlap. It is intuitive to understand that, as the density of the packing increases (either by increasing the number of spheres in a fixed volume or decreasing the volume for a fixed number of spheres), the system reaches a point where there is no longer any room for the spheres to move without overlapping. This happens at the jamming point, where the system undergoes a phase transition from a liquid-like state, where spheres can move past each other, to a solid-like state, where they are locked in place relative to their neighbors. At this point the Hessian of the finite-range potential (which includes a repulsive interaction that becomes significant only when particles approach within a certain distance) displays many (almost) flat directions [79], suggesting that the system has become isostatic. That is, at the jamming transition the number of constraints (from the contacts between particles) matches the number of degrees of freedom. The system has no redundant constraints that would make it rigid and no deficit of constraints that would make it under-constrained, ensuring that small displacements of particles do not lead to significant energy changes, which is reflected in the Hessian as zero or near-zero eigenvalues. In the same way, if one considers the problem of learning as that of satisfying P constraints with N parameters (see above), it turns out that the equivalent of jamming is the interpolation point at which, by increasing the number of parameters N while keeping P fixed, the number of unsatisfied constraints suddenly drops to zero, i.e. the network starts fitting the dataset. Before jamming (large P or small N) the generalization error is increasing; at the transition it displays a cusp; after jamming (the overparameterized regime) the test error decreases monotonically. Plus, as in glasses, the Hessian is singular at the interpolation point, implying that really the network enters a phase (the overparameterized phase) where redundacy of parameters is present and suggesting that it is in this phase that the dynamics cannot get stuck in poor minima since there are not enough constraints

to form local minima [65].

1.2.3 Algorithmic hardness from a geometric perspective

Inference problems, in whose class also enters machine learning in the Bayesian framework, being the goal that of inferring the posterior probability distribution of the model parameters given the data, often boils down some computational problems. The canonical way in statistics to model computational difficulty is NP completeness, which collects all problems that are both NP and NP-hard. NP is the class of decision problems for which a given solution can be verified in a polynomial time and a problem is NP-hard if every problem in NP can be reduced to it in a polynomial time. NP completeness deals with the worst-case scenario, meaning it assumes that nature produces adversarial instances for a given problem and the goal is to argue that there is no algorithm that works for all instances if the problem is computationally hard. But because in statistics data and models are not necessarily worst case, contrarily the problems are often concerned with the typical, expected-case performance (those in machine learning), NP completeness turns out not to be well suited for inference problems who often deals with probabilities and uncertainties. One of the most intriguing developments in modern day statistics regarding computational aspects, is the presence of borderline trivial problems where even if the number of parameters is small, they are difficult to solve because of data and model sizes or the presence of various kind of structural restrictions like sparsity or discreteness. So, if we are in the situation where we cannot prove certain classes of problems are NP-hard but a gap exists between information theoretic vs computation, what makes these problems hard? In this brief section we will delve in some ideas in [80] which we report discursively and completely informally, concerning new approaches in statistics to attack algorithmic intractability that come from intuitions in statistical physics and known under the name of the *overlap gap property* (OGP). Having a theory that formalizes how optimization problems can be hard because of solution space topology changes, will lend strength to the whole treatment of this thesis, the well-trodden idea being by now to understand when the space of solutions prevents learning.

Suppose we have a generic minimization (maximization) problem with random input $\mathcal{X}$ and defined as $\min_{\theta\in\Theta}\mathcal{L}(\theta,\mathcal{X})$ where $\mathcal{L}$ is some objective function and Θ is the solution space over which we will make the choices. The informal definition of overlap gap property that conveys the hardness of the optimization goes as follows: the model exhibits OGP if there exists some positive value $\mu > 0$, which denotes the proximity to optimality, and two parameters ν_1 and ν_2 such that if we pick two solutions θ_1 and θ_2 which are μ-good

$$\mathcal{L}(\theta_{1/2},\mathcal{X}) \leq \min_{\theta\in\Theta}\mathcal{L}(\theta,\mathcal{X}) + \mu \tag{1.8}$$

in the sense that they have provided value which is not more than μ from optimality, then it is true that

$$||\theta_1 - \theta_2|| \in [0,\nu_1] \cup [\nu_2,\infty). \tag{1.9}$$

This means that the distance between these two solutions is either at most ν_1 or at least ν_2 with no value in between. Since the overlap (i.e. the inner product

between two solutions) is a proxy for the distance, here explained why Eq. (1.9) can be rephrased using that rather than the distance. To have a more intuitive insight about what this means, if we look at the pictorial representation of the optimization problem in Fig. 1.4, the horizontal line represents the multi-dimensional set of parameters (now σ), while the vertical axis corresponds to the objective function value. If we are trying

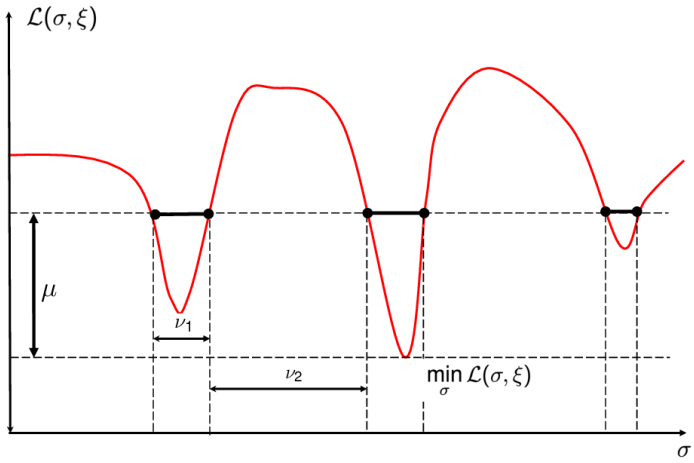


Figure 1.4: One-dimensional sketch of a loss landscape exhibiting OGP. Solutions are split into clusters, with diameter of each cluster smaller than the distance between any pair clusters. Reprinted from [80].

to minimize $\mathcal{L}$ and also looking at solutions that are μ-far from optimality, then once identified the optimal, it is easy to see that by choosing μ we are partitioning the entire solution space into clusters of μ-good solutions. OGP means the distance between any two solutions within a cluster is smaller than the distance between any two solutions from different clusters. Every diameter of the clusters is at most ν_1 but their distance is at least ν_2. The distance between clusters is indeed what makes OGP different from simple clustering, meaning the loss $\mathcal{L}$ could be non convex but if $\nu_1 > \nu_2$ then no OGP is present.

As an example of a learning paradigm that has a long-standing tradition in physics, the *symmetric binary perceptron* (SBP) with random data is exhibiting OGP [81, 82]. It was first introduced by Aubin, Perkins and Zdeborová in [83] and studied in [48] through the lens of statistical physics methods (we will discuss these in details in the next sections). Given a vector of binary weights $\boldsymbol{w} \in \{\pm 1\}^N$ (i.e. the Perceptron has just one layer of activity) and P inputs $\boldsymbol{\xi}^\mu \in \mathbb{R}^N$ that can be either normal or Rademacher, we say that the network satisfies the example if $|\xi^\mu \cdot w| \leq \kappa$, being κ a positive constant. So, leaving for the next chapters all the details about the formalization of a supervised learning task involving the Perceptron and its variations, we just stand with the constraint satisfaction problem (CSP) formalization: the binary vector $\boldsymbol{w}$ is called a solution for the problem if it satisfies

$$\frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i \xi_i^\mu \in I_\kappa \qquad \forall \mu \in [P] \tag{1.10}$$

being $I_\kappa \in [-\kappa, \kappa]$. In the proportional regime where $P = \alpha N$, it was first found in [84, 85] that if we call $\Sigma(\boldsymbol{\xi})$ the space of solutions, than for each choice of κ there exists a critical value of density of constraints $\alpha_c(\kappa)$ for which $\Sigma(\boldsymbol{\xi})$ is empty with high probability if $\alpha > \alpha_c(\kappa)$. The scaling for the critical value turns out to be $\alpha_c(\kappa) = \frac{\log 2}{\log P(|\boldsymbol{\xi}\cdot\boldsymbol{w}|<\kappa)}$, so inversely proportional to the probability that the absolute value of the Gaussian preactivation of the network is at most κ. Also something interesting happens if one looks at the rearrangement of solutions in the phase space: below $\alpha_c(\kappa)$ and for every density $\alpha > 0$, most of the solution space splits into singletons clusters that are distant $\mathcal{O}(N)$ from each other [22]. That is, an extreme form of clustering is going on here, where each cluster is a single dot but since not the entire solution space is isolated, there could potentially be tunnels connecting various singletons (weak clustering). One might ask whether or not in the positive density phase there exists a regime in α where clustering occurs but without exceptions (strong clustering) [48]. Indeed, the relevance of this is for algorithmic convergence as we will discuss in chapter 2, because if we would are with only point-like isolated solutions in the optimization landscape, then finding one would be like finding a needle in a haystack [45]. As the authors of [81] suggest, the answer is *yes*: there exists a positive density $\alpha_{\text{OGP}}(\kappa) < \alpha_c(\kappa)$ below which the problem is solvable, but above which algorithmic hardness prevents algorithms from finding solutions. Meaning, there is a gap between the satisfiability of the problem that goes on until $\alpha_c(\kappa)$ and the computational hardness, which allows the solution space to be heuristically explored until the OGP comes into play.

Chapter 2

Biasing the measure into wide valleys

In the previous chapters, we have clarified how the dynamics of stochastic algorithmic processes, such as learning in NNs, are sensitive to variations in hyperparameters that can drastically alter the complexity of the tasks. We introduced the static characterization of the optimization landscape geometry as a tool to investigate and formalize what prevents algorithms from converging. To date, the existence of a rigorous definition of computational hardness in the worst-case scenario, i.e. the Overlap Gap Property (OGP) [80], draws from a wide literature in statistical physics. It is particularly useful for the upcoming discussions, to make a summary of the most relevant achievements in the field for the context of neural networks. In general, learning dynamics in NNs is an example of an out-of-equilibrium process in complex systems. The underlying stochastic dynamics do not necessarily achieve states described by an equilibrium probability measure, as would be the case for ergodic systems. Moreover, sets of configurations that are quite atypical for certain classes of algorithms can become typical for others. This fact is not surprising per se in a general context, but it becomes significant and unexpected when sets of relevant configurations, typically inaccessible to a wide range of search algorithms, suddenly become extremely attractive to others. In this chapter, we discuss how this phenomenon manifests in learning with large-scale neural networks, beginning with a case study involving low precision synaptic weights: the binary Perceptron model [23]. The underlying idea, which forms a fundamental part of this thesis, is that in non-convex optimization problems, models can exhibit different phases, characterized by the arrangement of the solution space into one, many, or a few connected clusters. If efficient algorithms (those running in polynomial-time) seem to exist only when the system has so-called *unfrozen* clusters (or weak clustering in mathematical notation) with extensive connected regions of solutions where at most a sub-linear number of variables are fixed, then a contradiction arises whenever it is observed that the canonical Gibbs ensemble, shows that all solutions of a given optimization problem are *frozen*, yet algorithms are still able to solve it efficiently.

In the following, we will see how to overcome the static equilibrium description of learning. As anticipated in the preliminary chapter, the statistically relevant typical solutions under a uniform measure over all minimizers consist primarily of bad min-

ima, which in the unfortunate case are point-like isolated solutions in the landscape. Therefore, if it happens that algorithms in some problems with a typical landscape of this nature can find solutions when everything suggests that the task is hard from a statistical physics standpoint (not rigorous, not as the worst-case), it is reasonable to assume that the description itself is inappropriate. Throughout this discussion, to ensure relevance to algorithms in non-convex problems, we move away from the equilibrium description and we revisit and introduce the *local entropy* (LE) [45–48, 86]. LE is both an additional quantity to be included in the Gibbs measure, for an analytical description that can sample good and flat solutions (i.e. highly surrounded) of algorithms, and also an objective function that can be exploited to derive a class of optimization strategies that actively target dense clusters of solutions (if they exist).

2.1 The instructive case of the binary Perceptron

Introduced by Rosenblatt in 1962 [87], the Perceptron is the simplest example of a feed-forward neural network that acts as a simplified model of neuron. The network has no hidden layers and is parameterized by a vector of synaptic couplings that in the discrete formulation of the model is $\{w_i\}_{i=1}^N \in \{-1,+1\}^N$. Given an input pattern $\boldsymbol{\xi}^\mu$ from a synthetic dataset $\mathcal{D} = \{\boldsymbol{\xi}^\mu, \sigma^\mu\}_{\mu=1}^P$ in dimension P, each component of the input is considered an i.i.d. Rademacher variable while the binary outputs σ^μ can be either uncorrelated, as in the *random* classification scenario, or determined by a teacher device $\boldsymbol{w}^T$ that generates the rule to map each input to a label, as in the *generalization* case. In general, the output of the network is given by $y^\mu(\boldsymbol{\xi}^\mu; \boldsymbol{w}) = \text{sign}(\boldsymbol{w} \cdot \boldsymbol{\xi}^\mu)$. In a biological analogy, the label $\sigma^\mu = 1$ represents the excited state of a neuron, while $\sigma^\mu = -1$ the quiescent one. Despite its simplicity, the Perceptron offers a range of beneficial computational characteristics: it processes inputs simultaneously, can store an extensive amount of information, and its learning procedure can be implemented as a straightforward online process. Nevertheless, it also faces significant computational constraints, being limited to solve only tasks that involve linearly separable classifications.

Once the training set $\mathcal{D}$ is specified, one can borrow tools from statistical physics in order to study the typical learning behaviour of the model, extracted in the thermodynamic limit, where both $P, N \to \infty$ with a finite ratio $\alpha = P/N = \mathcal{O}(1)$ representing the storage load. Indeed, it is possible to explore the probability distribution imposed on the synaptic couplings $P_{\boldsymbol{w}}(\boldsymbol{w})$ and to follow how it changes when adding new constraints (i.e. patterns) to the task. Being interested in self-averaging quantities, whose probability distribution gets focused on typical values as the variance vanishes with $N \to \infty$, one can consider the *quenched* entropy density

$$S = \left\langle \log \int d\mu(\boldsymbol{w}) \prod_{\mu=1}^{P} \Theta\left(\sigma^\mu \frac{\boldsymbol{w} \cdot \boldsymbol{\xi}^\mu}{\sqrt{N}}\right) \right\rangle_{\mathcal{D}} = \left\langle \log \int d\mu(\boldsymbol{w})\, \mathbb{X}_{\mathcal{D}}(\boldsymbol{w}) \right\rangle_{\mathcal{D}} \tag{2.1}$$

which is a measure of the average of the logarithm of the partition function over the

disorder, represented by the dataset, whose randomness is supposed to not evolve during the experiment timescale. Indeed, the integral in the expression refers to the volume of solutions satisfying all patterns constraints and $d\mu(\boldsymbol{w})$ is instead the prior weight measure at each fixed disorder realization. To exploit traslational invariance and treat each component w_i homogeneously, it is possible to exchange the order between the average and the logarithm making use of the simple identity

$$\log(x) = \lim_{n\to 0^+} \frac{x^n - 1}{n}. \tag{2.2}$$

This gives rise to the *replica trick*, thanks to which, by simply considering an enlangerd system composed of $n \in \mathbb{N}$ independent and identical copies of the original one, the disorder average can be easily performed and in the end it introduces an interaction between the copies. During the computation, as it will be clear in all the analytical steps reported in the thesis, it is unfeasible to perform the action minimization over the full parametric space. So, the saddle point equations, whose solutions represents the order parameters values that dominate the contribution to the partition function in the thermodynamic limit and describe the macroscopic state of the system, are usually solved for a restricted subspace. Usually, it consists in an *ansatz* over the geometrical structure of the described phase space, that in the simplest case is the Replica Symmetric (RS) ansatz, for which the volume is supposed being characterized by a unique connected component, and so the virtual replicas are symmetric under permutation. Making use of this powerful method, and assuming the space of solutions is well described by the chosen ansatz, it is possible to explore and answer multiple non-trivial questions about the statics of the learning problem. It is important to stress that, in the current formulation, the learning process is treated as a constraint satisfaction problem (CSP) and that the Hamiltonian of the system relies on the cost function that one wishes to minimize. Expressing the quenched entropy as in Eq. (2.1), we took for granted the $\beta \to \infty$ limit, which corresponds to assigning a uniform weight over the global minima of the energy function.

Solving numerically saddle points, one can access the entropy value of solutions for a given constraint density α. It is intuitive to imagine that in the random (i.e. storage) case, while increasing α, one crosses a critical value α_c such that the typical distance between solutions goes to zero and accordingly the solution space shrinks to a single point. The corresponding entropy will be zero at this threshold, since for binary models we are counting solutions and not measuring volumes at zero temperature (as would be in the continuous counterpart). So, if it happened that the entropy is negative in certain α regimes, this would suggest that the ansatz on the geometric structure is wrong, forcing to consider solutions in which the Gibbs measure can be decomposed into mixtures of pure states (Replica Symmetry Breaking, RSB). The straightforward variation to RS is when n replicas can be grouped into $\frac{n}{m}$ blocks, containing m replicas each. The 1RSB ansatz assumes solutions organized in an exponential number of groups, well separated between each other, and for which ergodicity is broken. So, it differentiates between intra-cluster overlap q_1 and inter-cluster overlp q_0 between solutions.

It is not the purpose of the present chapter to delve into the detail of the analytical

solution of the model nor to derive the SAT/UNSAT transition α_c at which the entropy vanishes. For that, we refer to [23] and we suppose to known a priori the critical capacity α_c in the storage setting. Now, let's say we want to characterize the landscape geometry of solutions sampled from the Gibbs measure below α_c. The frozen-1RSB stable phase for this model [22] already suggests that the Gibbs measure is a pure state composed by an exponential number of point-like clusters, with internal overlap $q_1 = 1$ and inter-cluster overlap $q_0 = q_{RS}$. However, we can keep on asking questions: what does the neighborhood of these typical solutions looks like? Is it possible to find other solutions in close proximity? Remember that the overlap q_{RS} refers to an averaged quantity, not ruling out a priori that there are solutions with $q < q_{RS}$ for a different probability measure. The suitable quantity for getting insights about the local properties of the solution landscape is the Franz-Parisi potential

$$\phi_{FP}(s) = \frac{1}{N}\left\langle \frac{1}{Z}\int d\mu(\tilde{\boldsymbol{w}})\,\mathbb{X}_{\mathcal{D}}(\tilde{\boldsymbol{w}})\,\mathcal{S}_{\mathcal{D}}(\tilde{\boldsymbol{w}},s)\right\rangle_{\boldsymbol{\mathcal{D}}} \tag{2.3}$$

which is a typical *local entropy* (LE) density of solutions

$$\mathcal{S}_{\boldsymbol{\mathcal{D}}}(\tilde{\boldsymbol{w}},s) = \frac{1}{N}\ln\int d\mu(\boldsymbol{w})\mathbb{X}_{\boldsymbol{\mathcal{D}}}(\boldsymbol{w})\,\delta\left(Ns - \sum_{i=1}^{N}\tilde{w}_i w_i\right). \tag{2.4}$$

In Eq. (2.4), once choosing a reference configuration $\tilde{\boldsymbol{w}}$, its LE is just logarithm of the number of configurations that are solutions $\boldsymbol{w}$ at a given overlap s from $\tilde{\boldsymbol{w}}$. The potential, just imposes the reference to be itself a solution, given the characteristic function $\mathbb{X}_{\mathcal{D}}(\tilde{\boldsymbol{w}})$ which is the zero temperature limit of the Gibbs factor. Notice that, the overlap between solutions is related to a normalized Hamming distance with the relation $d = \frac{1-s}{2}$ and so everything can be expressed as a function over a distance measure. For any distance, the LE is bounded above by the *total* number of configurations at distance d. This value, that we call $\mathcal{S}_{\max}$, is attained at $\alpha = 0$, when the problem is left with no constraints.

A plot of $\phi_{FP}(d)$ as a function of d is shown in the left panel of Fig. 2.2 for different values of α and in the RS ansatz. It exists a neighborhood of distances $d \in [0, d_{\min})$ around the reference $\tilde{\boldsymbol{w}}$ for which the local entropy is negative, which is unphysical in that binary case. But as anticipated, even if this in principle suggests the need for a more accurate ansatz, the potential will be at most zero for that range of distances, not changing the general picture. So, in this case, one can demonstrate as in [23], that $\phi_{FP}(d=0) = 0$ but that for $d \to 0$ is

$$\frac{\partial\phi_{FP}}{\partial d} = \alpha C d^{-1} + \mathcal{O}(\ln d) \tag{2.5}$$

where C is a negative constant. This tells that $d_{\min} > 0$ for *any* $\alpha > 0$ and so that typical solutions are isolated, since if in Eq. (2.3) the overlap is normalized by N, one has to flip an extensive number of weights in order to find a solution $\boldsymbol{w}$ when starting from $\tilde{\boldsymbol{w}}$. In this frozen and golf-like course, no algorithm is expected to converge, and although nothing can be said except in the typical case, glassy landscape of this kind have long been known to be a major obstacle to optimization (see 1.2.2) [66, 68].

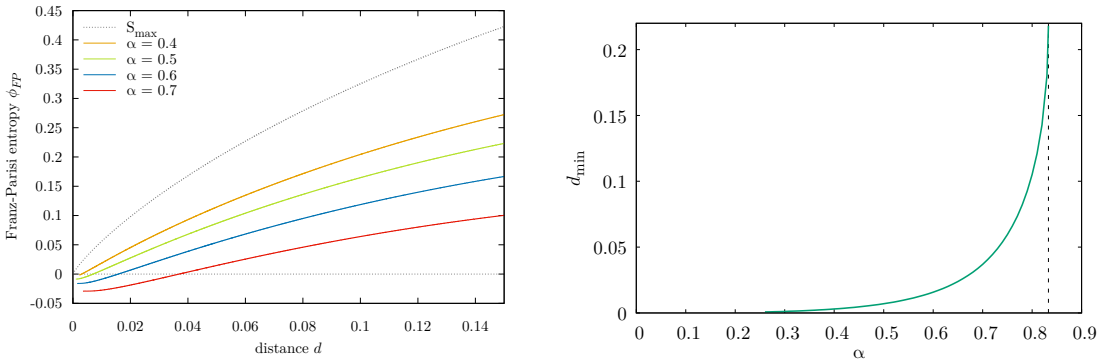


Figure 2.1: *Left* panel: averaged local entropy of typical solutions as a function of the distance, for several values of α. *Right* panel: distance $d_{\min}$ for which the Franz-Parisi potential is zero. At the SAT/UNSAT transition line (vertical dashed) the distance coincides with the typical distance between solutions $d \simeq 0.218$. Reprinted from [88].

2.1.1 Contradictory numerical results

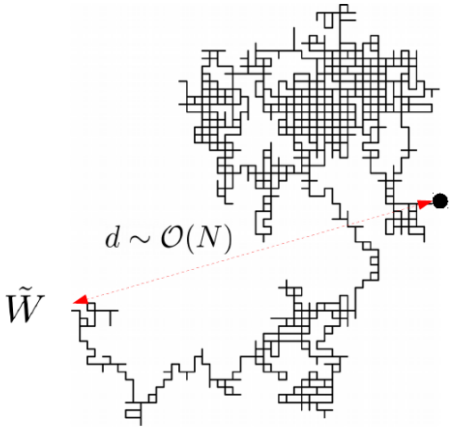


Figure 2.2: Random walk on the solutions belonging to the same connected component. These configurations, despite being invisible to the replica computation, are present everywhere in the solution space of the binary Perceptron. Reprinted from [89].

Is what happens in practice so catastrophic? Some numerical results [86] are in agreement with this picture. Simulated Annealing (SA) solving time for this problem is in fact known to diverge, even when the number of synapses N is not that large. However, despite challenging conditions, some heuristics that are variations of the Belief Propagation (BP) algorithm nearly saturates the SAT/UNSAT threshold [45]. With such contradictory results, it is crucial to identify what makes solutions discovered by efficient algorithms appealing for learning dynamics. One key characteristic from the theoretical analysis, i.e. geometrical isolation of typical solutions, is easily verifiable. Suppose, indeed, that accordingly to [86], we are able to target some solutions for the problem using a kind of a *smart* optimization strategy (we will not enter into the detail of the optimization protocol because is not relevant to the message to

be conveyed). In order to estimate the properties of the neighborhood of that solution, one can implement a random walk restricted to the set of solutions at zero energy, i.e. a zero temperature Monte Carlo (MCT0), during which it is possible to measure the quantities of interest [46]. To measure the radius of connected components, one approach is to let a random walk progress for an extended period and observe the maximum distance it reaches. However, due to the curse of dimensionality, a more efficient method is to bias the exploration outward. This can be achieved by arranging the indices in the Monte Carlo move proposal to favor the selection of synapses that retain the same value as the initial configuration $\tilde{\boldsymbol{w}}$. The method reveals that in the landscape there are connected structures that span across the phase space, reaching distances of the order $\mathcal{O}(N)$.

2.1.2 Large deviation analysis

In the present section we solve the mystery concerning how it is possible that the Boltzmann measure trace away clustered regions in the solution space, by the lens of a large deviation analysis [46]. Indeed, we define a biased probability measure towards solutions that have many other solutions nearby

$$P_{\mathcal{D}}(\tilde{\boldsymbol{w}}; y, d) = \frac{e^{yN\mathcal{S}_{\mathcal{D}}(\tilde{\boldsymbol{w}};d)}}{\int d\mu(\tilde{\boldsymbol{w}}')\, e^{yN\mathcal{S}_{\mathcal{D}}(\tilde{\boldsymbol{w}};d)}} \tag{2.6}$$

where y has the role of an inverse temperature that assigns a large statistical weight the more the solution have a high local entropy value. Indeed, in the $y \to \infty$ limit, we would expect to find the minimum of an energy-like function defined as $\mathcal{E}(\tilde{\boldsymbol{w}}; d) = -\frac{1}{N} \ln \mathcal{N}_{\mathcal{D}}(\tilde{\boldsymbol{w}}, d)$, whose average over the disorder gives the local entropy in Eq. 2.4. So, the LE can be obtained as the first derivative of a certain averaged free entropy density

$$\mathcal{F}(d, y) = -\frac{1}{Ny} \Big\langle \ln \left(\int d\mu(\tilde{\boldsymbol{w}})\, \mathbb{X}_{\mathcal{D}}(\tilde{\boldsymbol{w}})\, \mathcal{N}_{\mathcal{D}}(\tilde{\boldsymbol{w}}, d)^y \right) \Big\rangle_{\mathcal{D}} \tag{2.7}$$

that is what now one wishes to compute with the replica theory. Being not the purpose of this chapter to focus of any detail, we refer to [46, 89] for the step-by-step computation of (2.7).

The results for the random storage scenario are shown in Fig. 2.4. If the blue dotted curves refer, as before, to the local entropy of typical configurations, the theoretical red curves of the LE when sampling solutions with the biased measure (eq. (2.6)), are now greater than zero in a neighborhood of d for $\alpha < \alpha_c$. This signals the presence of dense clusters of solutions that shrink when increasing the constraints. Additionally, the curves for different capacities collapse onto each other when $d \to 0$, and precisely on the $\alpha = 0$ binomial curve, which means that the centers of these clusters are extremely dense. For large distances the local entropy tends to the entropy of typical solutions (according to the Boltzmann distribution) with a second-order phase transition. We also notice that there is a threshold value in the density, namely α_U, beyond which the curves stop being monotonic in d, signaling that beyond a threshold that differs from the SAT/UNSAT, the dense cluster is

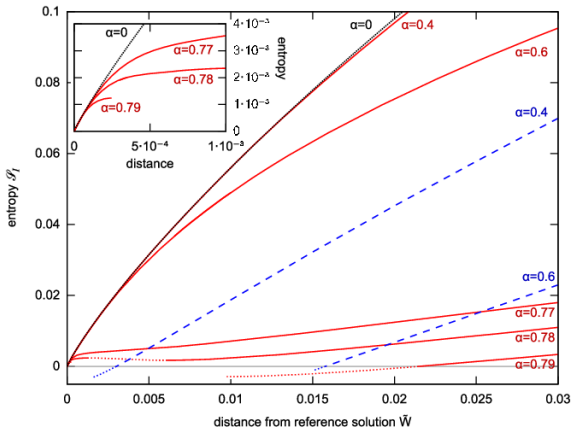


Figure 2.3: Local entropy of atypical solutions (red), i.e. subdominant wrt the flat measure, and typical solutions (blue) as a function of the distance d from a reference for different values of α. Reprinted from [46].

breaking up. We anticipate that, given a model, the threshold value for which this phenomenon happens, would be related in binary models to the algorithmic threshold hardness above which efficient solvers can no longer solve the task at zero energy configurations (chapter 3, 4).

2.2 Local entropy as an objective function: the robust ensemble

What if one would like to optimize local entropy in practical solvers without the need to rely on a difficult estimation of Gibbs-like integrals? [45] It is possible to generalize the biased measure expression in Eq. (2.6) for finite temperature and using a soft distance constraint. The resulting measure would read

$$\begin{aligned} P_{\mathcal{D}}(\tilde{\boldsymbol{w}}; y, \gamma, \beta) &= \frac{e^{yN\mathcal{S}_{\mathcal{D}}(\tilde{\boldsymbol{w}};d)}}{Z(y,\gamma,\beta)} = \frac{e^{y \ln \sum_{\boldsymbol{w}} e^{-\beta E(\boldsymbol{w}) - \gamma d(\boldsymbol{w},\tilde{\boldsymbol{w}})}}}{Z(y,\gamma,\beta)} \\ &= \frac{1}{Z(y,\gamma,\beta)} \left(\sum_{\boldsymbol{w}} e^{-\beta E(\boldsymbol{w}) - \gamma d(\boldsymbol{w},\tilde{\boldsymbol{w}})} \right)^y \end{aligned} \tag{2.8}$$

where if one assumes y to be an integer, Z will be the partition function of an enlarged system consisting in y replicas $\{\boldsymbol{w}^a\}$ of the original model, all coupled to a center $\tilde{\boldsymbol{w}}$, which is the reference configuration

$$\begin{aligned} Z(y,\gamma,\beta) &= \sum_{\tilde{\boldsymbol{w}}} \left(\sum_{\boldsymbol{w}} e^{-\beta E(\boldsymbol{w}) - \gamma d(\boldsymbol{w},\tilde{\boldsymbol{w}})} \right)^y \\ &= \sum_{\tilde{\boldsymbol{w}}} \sum_{\{\boldsymbol{w}^a\}} e^{-\beta \sum_{a=1}^y E(\boldsymbol{w}^a) - \gamma \sum_{a=1}^y d(\boldsymbol{w}^a,\tilde{\boldsymbol{w}})}. \end{aligned} \tag{2.9}$$

So, applying the method as a solver just consists in sampling replicas, adding an elastic constraint between them, and while annealing the temperature during the learning process in order to find the minimum, also a cooling in the γ parameter is needed. Intuitively, this will ensure the coupled system to dynamically find a region in the optimization landscape that is both low in energy and sufficiently flat to contain all of the replicas.

2.2.1 Focusing - BP

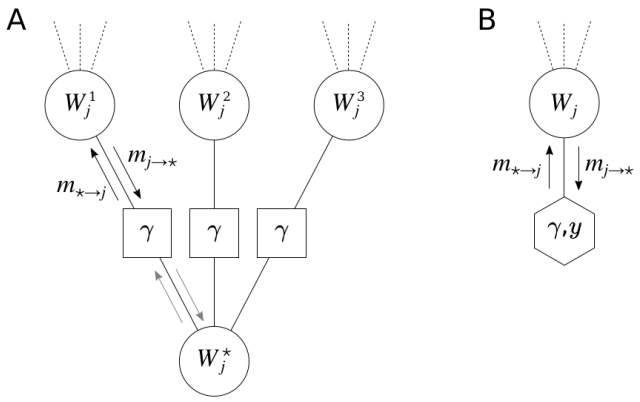


Figure 2.4: (A): Portion of a BP factor graph for a replicated variable w_j with $y = 3$ replicas and a reference $w_j^\star$. The squares represents the interactions $\gamma w_j^\star w_j$. All the messages are assumed to be the same in corresponding edges. (B): Transformed graph once exploiting the symmetry to reduces the number of nodes, keeping only one representative per replicas. The hexagon is now the pseudo-self-interaction that depends on both γ and y. Reprinted from [45].

As an example of this, which would be cited the most along the following discussions, let us briefly describe the replicated version of Belief Propagation, namely focusing BP [45], to seek maximally dense solutions in the landscape. Instead of focusing on the joint distribution across each replicated vector variable (necessary for a more accurate handling of correlations), one can simply replicate the original factor graph y times. In this setup, each replicated site j is linked to an additional variable $w_j^\star$, thereby introducing y interactions between all the w_j^a and $w_j^\star$. If we now make a symmetry assumption between the replicas, implying that each replica of the system would behave in exactly the same way and that same messages would be exchanged along the edges of the graph, regardless of the replica index, we can avoid the redundancy and work with a single system. This model is essentially the same as the standard factor graph, but with the difference that the elastic interaction between the replicas is now represented as a self-interaction at each variable site w_j. In this setup, each variable communicates with $y - 1$ identical copies of itself via an auxiliary variable, which can be eliminated from the equations. This structure of the factor node is depicted in Fig. 2.4. Consequently, at each iteration step t, each variable receives an

additional message of a specific form

$$m_{\star\to j}^{t+1} = \tanh\left((y-1)\tanh^{-1}\left(m_{j\to\star}^{t}\tanh\gamma\right)\right)\tanh\gamma \tag{2.10}$$

being $m_{j\to\star}^{t}$ the cavity magnetization derived from the standard factor graph, excluding the self-interaction at time t. Following this approximated transformation, the inverse temperature parameter y re-emerges as a continuous variable and is not restricted to integer values anymore.

Part II

The benefits of robustness

Chapter 3

Sampling atypical solutions in binary models

Investigating the geometric properties of different minimizers of the loss functions is crucial for gaining insights into the dynamics of learning and the ability of models to generalize. As it should be clear by now, numerous empirical studies have supported a consistent relationship between the flatness of a minimizer and the models' generalization capabilities [45–48], where the presence of high local entropy minima strongly depends on the number of the network parameters in relation to the number of constraints. In a more refined sense, the loss functions of neural networks appear to have wide flat regions of solutions, highly accessible and well-generalizing, as corroborated by empirical results across different datasets and randomized labels, suggesting a stable attribute of the networks. In [9], it was demonstrated how a procedure based on simultaneously minimizing the loss value and the loss sharpness, better known as Sharpness Aware Minimization (SAM), outperforms on a variety of benchmark datasets. More recently, the tendency towards wide basins opened up the perspective for ensemble methods and distributed training [40, 41], confirming how flourishing this line of research is.

In the present Chapter, we provide the first analytical insight on the geometric structure of these wide minimizers in binary models [90], characterizing them as a coalescence of minima with a high level of robustness. We show how these results allow to identify the regimes of learning in which they appear and, far more disruptive than that, the threshold that prevents algorithms from learning.

3.1 Storing $P \simeq \mathcal{O}(N)$ random patterns with a Perceptron

Let us start by defining the learning problem for a single layer neural network (i.e. the perceptron) as we did before, i.e. as that of learning the functional mapping $f : \mathbb{R}^N \to \pm 1$ that assigns a label $\sigma^\mu = \pm 1$ for each of the P patterns $\boldsymbol{\xi}^\mu \in \mathbb{R}^N$ in the dataset $\mathcal{D} = \{\boldsymbol{\xi}^\mu, \sigma^\mu\}_{\mu=1}^P$. In what follows we will focus on synthetic dataset, i.e. where the patterns are random i.i.d. N-dimensional Gaussian variables $\xi_i^\mu \sim \mathcal{N}(0,1),\ \forall i = 1, \dots, N$ while the labels σ^μ are chosen to be ± 1 with equal probability.

This setting aims to model the situation in which the rule that generates the label is noisy or the data are corrupted, so that there is no effective correlation between the input patterns and the target label. Therefore, one cannot speak of generalization or predictive properties in the context, but equally compelling is the question on how many patterns is the network able to store correctly, i.e. what is the maximum number of constraints P it is capable of satisfying. As clarified in previous sections, the perceptron serves as a fundamental building block for all the more complex DNNs, being a simplified model of neuron that learns to adjust a vector of weights $\boldsymbol{w}$ in order to correctly classify each neuronal signal (pattern) as excitatory (+1) or inhibitory (−1)

$$y^{\mu} = \text{sign}\left(\frac{1}{\sqrt{N}}\sum_{i=1}^{N} w_i\,\xi_i^{\mu}\right). \tag{3.1}$$

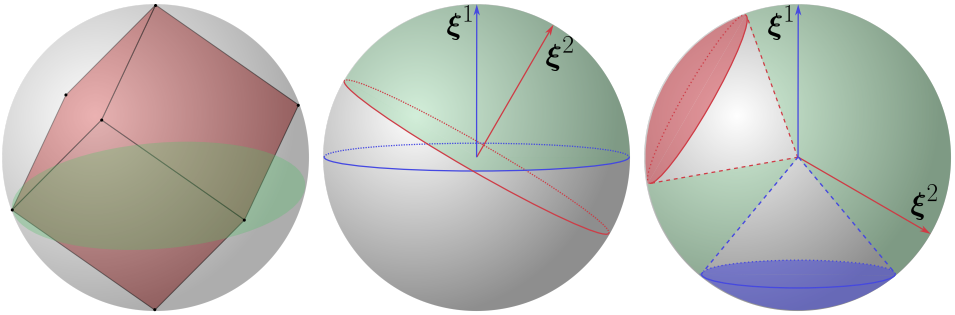


Figure 3.1: *Left* panel: the cube represents the space of possible values of weights for a binary Perceptron model. The green shaded area is the convex space for the corresponding spherical problem. Here only two vertices of the cube lies inside the green region. *Middle* and *right* panels: space of solutions (green shaded area) in the spherical Perceptron with $\kappa = 0$ and $\kappa < 0$ respectively. Reprinted from [88].

The optimization space depends on the tipe of weights, which in standard settings can live both in the hypersphere of radius $\sqrt{N}$ if the weight vector is constrained by $||\boldsymbol{w}||^2 = N$, or on the hypercube if the model is binary $w_i \in \pm 1$, being all the possible value of $\boldsymbol{w}$ lying on the vertices of the cube. Since it can only model linearly separable functions, it can only deal with classification problems in which an $N-1$ dimensional hyperplane exists that separates the two classes of labels. Learning the assignments, or finding the hyperplane, means to satisfy the set of constraints

$$\Delta^{\mu} = \frac{\sigma^{\mu}}{\sqrt{N}}\sum_{i} w_i\,\xi_i^{\mu} \geq 0 \qquad \forall \mu = 1, \ldots, P \tag{3.2}$$

where a positive stability Δ^{μ} for the example μ implies that the network output label y^{μ} is equal to the correct one σ^{μ}. In general, it may be not enough to simply fit the dataset and it may be necessary to do so in such a way that is robust against input perturbations, i.e. by imposing a margin $\kappa > 0$

$$\Delta^{\mu} = \frac{\sigma^{\mu}}{\sqrt{N}}\sum_{i} w_i\,\xi_i^{\mu} \geq \kappa \qquad \forall \mu = 1, \ldots, P \tag{3.3}$$

that ensures a solution for Eq. (3.3) to be also a solution for Eq. (3.2) (not vice versa), but with the advantage that within a certain level of pattern perturbation, which will depend on κ, the problem will still be satisfied. In practice, one can think of choosing a margin as choosing a ball of radius $\sim \kappa\,\mathcal{O}(\sqrt{N})$ such that the ball containing the nearest pattern to the hyperplane will still be to the right side of the decision boundary. Sometimes satisfying all constraints can be hard or even impossible, and one may decide to admit errors within a certain threshold by imposing a margin $\kappa < 0$ (see Fig. 3.1 for a pictorial representation). In models with continuous weights this will have an effect on the convexity of the problem, as we will see in Chapter 5. For now, let us restrict to the binary non-convex setting for which $\boldsymbol{w} \in \{-1, +1\}^N$ and $\kappa > 0$.

3.1.1 Equilibrium configurations

In retracing in detail the results in [90], we already take the statistical mechanics approach of looking for *self-averaging* quantities in the thermodynamic limit $N, P \to \infty$ for granted. Precisely, once it has been clarified that the flat measure over all configurations that are solutions with a given margin κ is proportional to

$$\mathbb{X}_{\boldsymbol{\xi}}(\boldsymbol{w}; \kappa) = \prod_{\mu=1}^{P} \Theta\left(\frac{\sigma^\mu}{\sqrt{N}} \sum_{i=1}^{N} w_i\, \xi_i^\mu - \kappa\right) \tag{3.4}$$

if we consider the error counting loss, being $\Theta(\cdot)$ the Heaviside theta function, it should be clear from Chapter 2 that the Gibbs entropy of the solutions can be obtained by averaging $\langle \ldots \rangle_{\boldsymbol{\xi},\sigma}$ over the disorder induced by the random realization of patterns and labels

$$\phi(\alpha, \kappa) = \lim_{N\to\infty} \frac{1}{N} \langle \ln Z \rangle_{\boldsymbol{\xi},\sigma} = \lim_{N\to\infty} \frac{1}{N} \langle \ln \sum_{\{w_i = \pm 1\}} \mathbb{X}_{\xi,\sigma}(\boldsymbol{w}; \kappa) \rangle_{\boldsymbol{\xi},\sigma} \tag{3.5}$$

being Z the partition function, which is the zero temperature limit of an equilibrium Gibbs measure, with the number of violated patterns as energy. The reason why we are interested in doing so, is that by exploring the entropy of different kind of solutions (i.e. with different margin κ), will allow us to conjecture about the amount of sharp and highly surrounded minimizers in the phase space.

Using the the replica trick $\phi = \lim_{N\to\infty} \lim_{n\to 0} \frac{\ln \langle Z^n \rangle_{\boldsymbol{\xi},\sigma} - 1}{nN}$, we again discover that the computation of Eq. (3.5) reduces in computing the replicated partition function of an enlarged system with $n \in \mathbb{N}$ identical copies of the original one

$$\begin{aligned} Z^n &= \int \prod_{a=1}^{N} d\mu(W^a) \prod_{\mu a} \Theta\left(\frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^a \xi_i^\mu - \kappa\right) \\ &= \int \prod_{a\mu} \frac{dv_\mu^a d\hat{v}_\mu^a}{2\pi} \prod_{\mu a} \Theta\left(v_\mu^a - \kappa\right) e^{-i v_\mu^a \hat{v}_\mu^a} \sum_{\{\boldsymbol{w}^a\}_{a=1}^n} e^{i \sum_{\mu,a} \hat{v}_\mu^a \frac{1}{\sqrt{N}} \sum_i w_i^a \xi_i^\mu}. \end{aligned} \tag{3.6}$$

and where it is the disorder average which introduces a coupling between the n copies. Here we have sent $\xi_i^\mu \to \xi_i^\mu \sigma^\mu$ without affecting the probability measure over the patterns $\boldsymbol{\xi}^\mu$, since we can simply set $\sigma^\mu = 1,\ \forall \mu = 1, \dots, P$ without loss of generality. Plus, we also introduced the auxiliary variable $v_\mu^a = \frac{1}{\sqrt{N}} \sum_{i=1}^N w_i^a \xi_i^\mu$ so that the pattern dependence is isolated in an exponential term when using the integral representation of the Dirac delta function

$$\delta(v) = \int \frac{d\hat{v}}{2\pi} e^{iv\hat{v}}. \tag{3.7}$$

We are now able to perform the disorder average, obtaining in the large N limit

$$\prod_\mu \left\langle \prod_i e^{i \frac{\xi_i^\mu}{\sqrt{N}} \sum_a w_i^a \hat{v}_\mu^a} \right\rangle_{\boldsymbol{\xi}^\mu} \simeq \prod_\mu e^{-\sum_{a<b} \hat{v}_\mu^a \hat{v}_\mu^b \left(\frac{1}{N} \sum_i w_i^a w_i^b \right) - \frac{1}{2} \sum_a \left(\hat{v}_\mu^a \right)^2}. \tag{3.8}$$

The expression reveals the order parameter for the model, being $q_{ab} \equiv \frac{1}{N} \sum_{i=1}^N w_i^a w_i^b$ the degree of overlap between two distinct replicas a and b. We can thus introduce an integral over the spectrum of potential values for this parameter by employing again a delta function. In the context, the conjugate parameter $\hat{q}_{ab}$ is required to be imaginary, thus allowing to make the transformation $\hat{q}_{ab} \to i\hat{q}_{ab}$. This step enables us to decouple across both the synaptic indices i, which range from 1 to N, and the pattern indices μ, which extend from 1 to $P = \alpha N$. By doing so, the subsequent formula for the replicated volume is easy to express

$$\langle Z^n \rangle_{\boldsymbol{\xi}} = \int \prod_{a<b} \frac{dq_{ab} d\hat{q}_{ab}}{(2\pi/N)} e^{NS(q_{ab}, \hat{q}_{ab})} \tag{3.9}$$

where we have defined

$$S(q_{ab}, \hat{q}_{ab}) = -\sum_{a<b} q_{ab}\hat{q}_{ab} + G_S(\hat{q}_{ab}) + \alpha G_E(q_{ab}) \tag{3.10a}$$

$$G_S = \ln \sum_{\{w^a\}} e^{\sum_{a<b} \hat{q}_{ab} w^a w^b} \tag{3.10b}$$

$$G_E = \ln \int \prod_a \frac{dv^a d\hat{v}^a}{2\pi} \prod_a \Theta(v^a - \kappa)\, e^{i \sum_a v^a \hat{v}^a - \frac{1}{2} \sum_a (\hat{v}^a)^2 - \sum_{a<b} q_{ab} \hat{v}^a \hat{v}^b} \tag{3.10c}$$

being the action $S(q_{ab}, \hat{q}_{ab})$ divided according to a physics analogy in interaction, entropic (G_S) and energetic (G_E) terms. Intuitively, the first is related to the interaction between different degrees of freedom in the system, which in the context have become the different replicas; the second is instead linked to the volume, associated with the configurational entropy of the model, which accounts for the number of ways it can be arranged microscopically without changing its macroscopic state; finally, the third is connected to the constraints. In order to move on with the replica calculation it is necessary to choose an Ansatz for the structure of the overlap matrix. As explained in 2, the simplest is the Replica Symmetric (RS) ansatz.

Replica-symmetric Ansatz

The RS ansatz, which assumes all replicas to be equivalent, means that any observable should be the same for all replicas, being the interactions between any pair identical. This greatly simplifies the mathematical treatment, since it means that the order parameter describing the overlap between replicas does not depend on which pair of replicas is considered

$$q_{ab} = \delta_{ab} + q(1 - \delta_{ab}) \tag{3.11a}$$
$$\hat{q}_{ab} = \hat{q}(1 - \delta_{ab}). \tag{3.11b}$$

We can therefore proceed by computing the entropic term using standard Hubbard-Stranotovich transformations and introducing the Gaussian measure $\int Dx = \int_{-\infty}^{+\infty} \frac{dx}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$ to obtain

$$\mathcal{G}_S \equiv \frac{\hat{q}}{2} + \lim_{n\to 0} \frac{G_S}{n} = \int Dx \ln 2\cosh(\sqrt{\hat{q}}x) \tag{3.12}$$

where we take $\mathcal{G}_S = \log G_S$ to recover the saddle point density for the entropy term in the $n \to 0$ limit. In the same way it is straightforward to recover the final expression for the energetic density term, which reads

$$\mathcal{G}_E \equiv \lim_{n\to 0} \frac{G_E}{n} = \int Dx \ln H\left(\frac{\kappa - \sqrt{q}x}{\sqrt{1-q}}\right) \tag{3.13}$$

having defined the complementary error function as $H(x) \equiv \frac{1}{2}\mathrm{erfc}\left(\frac{x}{\sqrt{2}}\right)$. Therefore, since the entropy is now given by

$$\phi_{RS} = \lim_{n\to 0} \frac{1}{n} \max_{\{q,\hat{q}\}} S_{RS}(q, \hat{q})\,, \tag{3.14}$$

the values of q and $\hat{q}$ that are computed by saddle point equations, i.e. by taking the derivatives of S with respect to both and setting them to zero, by neglecting $\mathcal{O}(n)$ terms the entropy expression is simply given by

$$\phi_{RS} = -\frac{\hat{q}}{2}(1 - q) + \mathcal{G}_S + \alpha\, \mathcal{G}_E. \tag{3.15}$$

Solving numerically saddle points one can study what is the satisfiability threshold for the problem as a function of κ (Fig. 3.2, *left*) and how the entropy of solutions changes when sampling with different margins κ (Fig. 3.2, *right*). Indeed, when choosing a margin κ, and by increasing the density of constraints $\alpha = P/N$, the network is supposed to learn more and more patterns as described in 2. In the large N limit, the model exhibits a sharp transition for satisfiability at the *critical capacity* $\alpha_c(\kappa)$ where $q \to 1$: for $\alpha < \alpha_c(\kappa)$ the probability that an instance of the problem with margin κ has a solution is 1, but it sharply drops to zero above this threshold [22]. The critical capacity is then the maximum density value α with non-vanishing entropy of solutions $\phi(\alpha_c(\kappa), \kappa) = 0$ (recent rigorous results prove the value for zero margin $\alpha_c(0) \simeq 0.833$ [91]). If, accordingly, we name $\kappa_{\max}(\alpha)$ the maximum margin as the

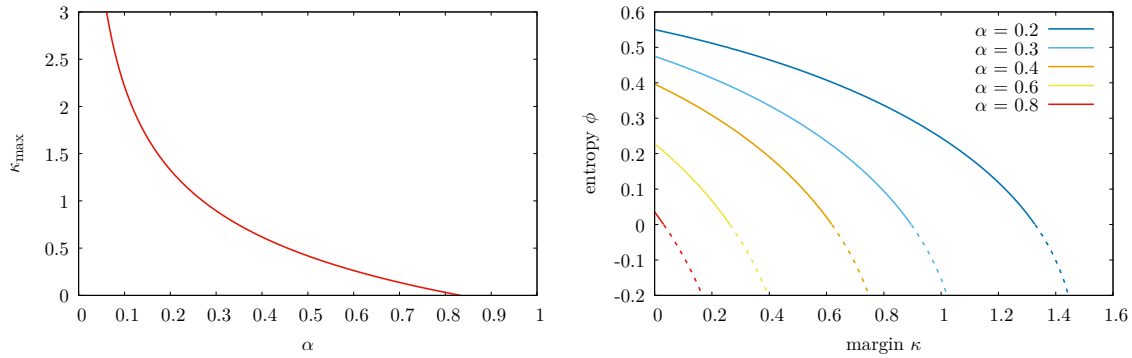


Figure 3.2: *Left*: the value $\kappa_{\max}$ at which the entropy vanishes, plotted as a function of α. This quantity goes to 0 at α_c. Reprinted from [90]. *Right*: RS entropy as a function of the margin κ for several values of α and $\kappa_{\max}$ as a function of α.

largest κ with non-vanishing entropy for a given α, then in Fig. 3.2 it is reported its value as predicted by the RS computation. By increasing the margin we are increasing the robustness requirement of the solution, and because of that, the critical capacity on the x-axis is a monotonically decreasing function of $\kappa_{\max}$. The critical capacity $\alpha_c(\kappa) \to 0$ when $\kappa \to \infty$ and it stands around $\alpha_c(0) \simeq 0.83$ as rigorous results confirm [85, 91].

Knowing now the thresholds for the satisfiability at each κ, we can verify what is the entropy of solutions ϕ_{RS} as a function of the margin. If in the plot of Fig. 3.2 we fix a value of α, the entropy appears to be a decreasing function of κ, that is: robust solutions with high margin are the least in the landscape. Indeed, at fixed α the greatest value of ϕ_{RS} is always for $\kappa = 0$ and the more one increases α, the more the phase space shrinks. It is important to stress that since when looking at different solutions we are not changing the measure as we did in Chapter 2 but only the margins, then it is still true for the space of solutions that an exponential number of them exists, it is the prefactor that changes according to κ: the $\kappa > 0$ solutions are exponentially less numerous (in N) compared to the $\kappa = 0$ ones and yet they have smaller mutual distance. The Hamming distance as a function of κ is again monotonically decreasing (Fig. 3.3) with the margin, and the lines change from solid to dashed when the entropy of solutions becomes negative, i.e. when $\kappa = \kappa_{\max}$ as defined above.

As explained in [22] and discussed in Chapter 2, since the space of typical solutions at $\kappa = 0$ is characterized by clusters with vanishing entropy, that is, from a point-like solution you have to flip an extensive number of weights ($\mathcal{O}(N)$) to reach another, one hope for algorithmic convergence is that robust solutions, condensing in regions of the landscape where they are closer than the typical solutions of the problem ($\kappa = 0$), are also those most highly surrounded. Indeed, being the margin of a solution a hard measure of stability with respect to local perturbations of the weights, this suggests that there is some kind of relationship between the flatness of a minimizer and the associated notion of margin. In order to explore how, below the critical capacity, solutions with high margin are arranged and how the structure of solution space

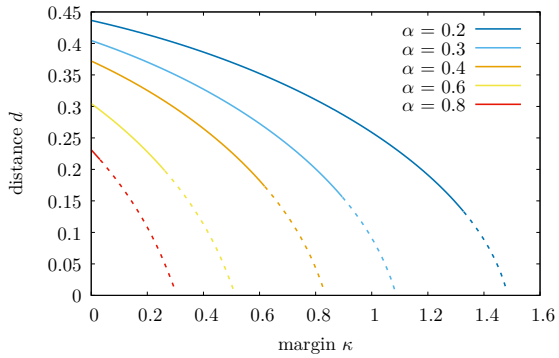


Figure 3.3: Normalized Hamming distance d between typical solutions as a function of their margin κ and for different α values. The lines become dashed when the entropy of solutions becomes negative, i.e. when $\kappa > \kappa_{\max}$. Reprinted from [90].

affects the performance of learning algorithms, one can begin by analyzing in which part of the phase space different solutions tend to be concentrated.

3.1.2 Characterizing the landscape in extensive neighborhoods

Similarly as we did in Chapter 2, one can define the *local entropy* of a configuration $\tilde{\boldsymbol{w}}$ and margin $\tilde{\kappa}$ as the logarithm of the number of configurations $\boldsymbol{w}$ that are solutions with margin κ at normalized Hamming distance d from the reference $\tilde{\boldsymbol{w}}$

$$\frac{1}{N}\ln \mathcal{N}_{\boldsymbol{\xi}}(\tilde{\boldsymbol{w}}, d, \kappa) = \frac{1}{N}\ln \sum_{\boldsymbol{w}} \mathbb{X}_{\boldsymbol{\xi}}(\boldsymbol{w}; \kappa)\, \delta\left(N(1-2d) - \sum_{i=1}^{N} \tilde{w}_i w_i\right). \tag{3.16}$$

Studying the local entropy profile as we vary the distance d allows to characterize the density of solutions (with given κ) in an extensive neighborhood of any given configuration. In the large N limit, because of self-averaging properties, the local entropy profile is fully captured by the average over the choices of the reference solution and the training set (i.e. a given disorder realization) by the Franz-Parisi potential

$$\phi_{FP}(t_1) = \frac{1}{N}\left\langle \frac{1}{Z}\sum_{\tilde{\boldsymbol{w}}} \mathbb{X}_{\xi}(\tilde{\boldsymbol{w}}; \tilde{\kappa}) \ln \mathcal{N}_{\xi}(\tilde{\boldsymbol{w}}, t_1, \kappa) \right\rangle_{\boldsymbol{\xi}} \tag{3.17}$$

where $t_1 = 1 - 2d$ is now the overlap with respect to the reference extracted from the Gibbs measure. In general, given a function $f(\{\boldsymbol{\xi}^\mu\}, \tilde{\boldsymbol{w}})$ the two averages can be computed easily as

$$\left\langle \left\langle f(\{\boldsymbol{\xi}^\mu\}, \tilde{\boldsymbol{w}}) \right\rangle_{\tilde{\boldsymbol{w}}} \right\rangle_{\boldsymbol{\xi}} = \left\langle \frac{\sum_{\{\tilde{\boldsymbol{w}}\}} \prod_\mu \Theta\left(\frac{1}{\sqrt{N}}\sum_i \tilde{w}_i\, \xi_i^\mu - \kappa\right)\, f(\{\boldsymbol{\xi}^\mu\}, \tilde{\boldsymbol{w}})}{\sum_{\{\tilde{\boldsymbol{w}}\}} \prod_\mu \Theta\left(\frac{1}{\sqrt{N}}\sum_i \tilde{w}_i\, \xi_i^\mu - \kappa\right)} \right\rangle_{\boldsymbol{\xi}} \tag{3.18}$$

where the ensemble average simplifies when using the replica trick to rewrite the expression: the denominator as the product of $n-1$ replicas since in the $n \to 0$ limit

$Z^{-1} = \lim_{n\to 0} Z^{n-1}$, assigning the replica index 1 to the expression in the numerator and the rest of indices from 2 to n to the others. If we use indices $a, b \in [n]$ for the replicas of the reference typical configuration, then

$$\left\langle f(\{\boldsymbol{\xi}^\mu\}, \tilde{\boldsymbol{w}})\right\rangle_{\tilde{\boldsymbol{w}}} = \lim_{n\to 0} \sum_{\{\tilde{\boldsymbol{w}}^a\}} \prod_{\mu a} \Theta\left(\frac{1}{\sqrt{N}} \sum_i \tilde{w}_i^a \, \xi_i^\mu - \kappa\right) f(\{\boldsymbol{\xi}^\mu\}, \tilde{\boldsymbol{w}}^{a=1}). \tag{3.19}$$

Plus, using $c, d \in [s]$ for the replicated $\boldsymbol{w}$, then since $\ln \mathcal{N}_\xi = \lim_{s\to 0} \partial_s \mathcal{N}_\xi^s$ we get

$$\begin{aligned}
\phi_{FP}(t_1) &= \lim_{\substack{n\to 0\\ s\to 0}} \partial_s \sum_{\{\tilde{\boldsymbol{w}}^a\}} \prod_a \left\langle \mathbb{X}_\xi(\tilde{\boldsymbol{w}}^a; \tilde{\kappa}) \, \mathcal{N}_\xi^s(\tilde{\boldsymbol{w}}^{a=1}, t_1)\right\rangle_{\boldsymbol{\xi}} = \\
&= \lim_{\substack{n\to 0\\ s\to 0}} \partial_s \int \prod_{\mu a} \frac{dv_\mu^a d\hat{v}_\mu^a}{2\pi} \, \Theta\left(v_\mu^a - \tilde{\kappa}\right) e^{-i v_\mu^a \hat{v}_\mu^a} \int \prod_{\mu c} \frac{du_\mu^c d\hat{u}_\mu^c}{2\pi} \, \Theta\left(u_\mu^c - \kappa\right) e^{-i u_\mu^c \hat{u}_\mu^c} \\
&\quad \times \sum_{\{\tilde{\boldsymbol{w}}^a\}} \sum_{\{\boldsymbol{w}^c\}} \prod_\mu \left\langle e^{\frac{i}{\sqrt{N}} \sum_i \xi_i^\mu \left(\sum_a \tilde{w}_i^a \hat{v}_\mu^a + \sum_c w_i^c \hat{u}_\mu^c\right)} \right\rangle_{\boldsymbol{\xi}^\mu} \prod_c \delta\left(N t_1 - \sum_{i=1}^N \tilde{w}_i^{a=1} w_i^c\right)
\end{aligned} \tag{3.20}$$

where two auxiliary variables have been introduced enforcing their definitions by Dirac delta functions

$$v_\mu^a = \frac{1}{\sqrt{N}} \sum_i \tilde{w}_i^a \xi_i^\mu \tag{3.21a}$$

$$u_\mu^c = \frac{1}{\sqrt{N}} \sum_i w_i^c \xi_i^\mu. \tag{3.21b}$$

Now the average over disorder can be done in the thermodynamic limit and it reads

$$\begin{aligned}
&\left\langle e^{\frac{i}{\sqrt{N}} \sum_i \xi_i^\mu \left(\sum_a \tilde{w}_i^a \hat{v}_\mu^a + \sum_c w_i^c \hat{u}_\mu^c\right)} \right\rangle_{\boldsymbol{\xi}^\mu} \simeq e^{-\frac{1}{2N} \sum_i \left(\sum_a \tilde{w}_i^a \hat{v}_\mu^a + \sum_c w_i^c \hat{u}_\mu^c\right)^2} \\
&= e^{-\frac{1}{2}\sum_{ab}\left(\frac{1}{N}\sum_i \tilde{w}_i^a \tilde{w}_i^b\right)\hat{v}_\mu^a \hat{v}_\mu^b - \frac{1}{2}\sum_{cd}\left(\frac{1}{N}\sum_i w_i^c w_i^d\right)\hat{u}_\mu^c \hat{u}_\mu^d - \sum_{ac}\left(\frac{1}{N}\sum_i \tilde{w}_i^a w_i^c\right)\hat{v}_\mu^a \hat{u}_\mu^c}.
\end{aligned} \tag{3.22}$$

Previous expression only depends on the first two moments of the variables (3.21). We can therefore define the order parameters

$$q_{ab} \equiv \left\langle v_\mu^a v_\mu^b \right\rangle_{\boldsymbol{\xi}^\mu} = \frac{1}{N} \sum_i \tilde{w}_i^a \tilde{w}_i^b \,, \tag{3.23a}$$

$$p_{cd} \equiv \left\langle u_c^\mu v_d^\mu \right\rangle_{\boldsymbol{\xi}^\mu} = \frac{1}{N} \sum_i w_i^c w_i^d \,, \tag{3.23b}$$

$$t_{ac} \equiv \left\langle v_\mu^a u_\mu^c \right\rangle_{\boldsymbol{\xi}^\mu} = \frac{1}{N} \sum_i \tilde{w}_i^a w_i^c \,. \tag{3.23c}$$

The first one is the overlap between two typical solutions of the binary Perceptron; the second represents instead the overlap between the coupled replicas; the third is

the overlap between one of the replicas $\boldsymbol{w}^c$ and the renference solution $\tilde{\boldsymbol{w}}^a$. Notice that, because of the delta function constraining the reference and $\boldsymbol{w}$, we have $t_{1c} = t_1$ for every c. By replacing the parameters definition, the Franz-Parisi entropy can be finally written as

$$\phi_{FP}(t_1) = \lim_{\substack{n\to 0\\ s\to 0}} \partial_s \int \prod_{a<b} \frac{dq_{ab}d\hat{q}_{ab}}{2\pi} \prod_{c<d} \frac{dp_{cd}d\hat{p}_{cd}}{2\pi} \prod_{c,\, a\neq 1} \frac{dt_{ac}d\hat{t}_{ac}}{2\pi} \prod_c \frac{d\hat{t}_{1c}}{2\pi}\, e^{NS(q_{ab},\hat{q}_{ab},p_{cd},\hat{p}_{cd},t_{cd},\hat{t}_{cd})} \tag{3.24}$$

where now the action reads

$$S = -\sum_{a<b} q_{ab}\hat{q}_{ab} - \sum_{c<d} p_{cd}\hat{p}_{cd} - \sum_{ac} t_{ac}\hat{t}_{ac} + G_S + \alpha\, G_E \tag{3.25a}$$

$$G_S = \ln \sum_{\{\tilde{w}^a\}} \sum_{\{w^c\}} e^{\sum_{a<b}\hat{q}_{ab}\tilde{w}^a\tilde{w}^b + \sum_{c<d}\hat{p}_{cd}w^cw^d + \sum_{ac}\hat{t}_{ac}\tilde{w}^aw^c} \tag{3.25b}$$

$$\begin{aligned} G_E = \ln \int \prod_a \frac{dv^a d\hat{v}^a}{2\pi} e^{iv_a\hat{v}_a} \int \frac{du^c d\hat{u}^c}{2\pi} e^{iu_c\hat{u}_c} \prod_a \Theta(v_a - \tilde{\kappa}) \prod_c \Theta(u_c - \kappa) \\ \times\, e^{-\frac{1}{2}\sum_{ab} q_{ab}\hat{v}^a\hat{v}^b - \frac{1}{2}\sum_{cd} p_{cd}\hat{u}^c\hat{u}^d - \sum_{ac} t_{ac}\hat{v}^a\hat{u}^c}. \end{aligned} \tag{3.25c}$$

Replica-symmetric Ansatz

In keeping with the computation as we did before, we proceed by applying the replica symmetry (RS) hypothesis within the structure of the order parameters. It is now necessary to differentiate between the overlap t_1 involving the replica $\tilde{\boldsymbol{w}}^{a=1}$ in the overlap constraint and the overlap t_0 for all the others of the reference

$$q_{ab} = \delta_{ab} + q(1 - \delta_{ab}) \tag{3.26a}$$

$$p_{cd} = \delta_{cd} + p(1 - \delta_{cd}) \tag{3.26b}$$

$$t_{ac} = t_1\, \delta_{a1} \,+\, t_0\, (1 - \delta_{a1}). \tag{3.26c}$$

Similarly for their conjugate quantities

$$\hat{q}_{ab} = \hat{q}(1 - \delta_{ab}) \tag{3.27a}$$

$$\hat{p}_{cd} = \hat{p}(1 - \delta_{cd}) \tag{3.27b}$$

$$\hat{t}_{ac} = \hat{t}_1\, \delta_{a1} \,+\, \hat{t}_0\, (1 - \delta_{a1}). \tag{3.27c}$$

In the entropic term, in order to be able to completely factorize over the indices $a = 1, \dots, n$ and $c = 1, \dots, s$ we have to reorganize the term

$$\hat{t}_0 \sum_a \tilde{w}^a \sum_c w^c = \frac{\hat{t}_0}{2} \left[\left(\sum_a \tilde{w}^a + \sum_c w^c \right)^2 - \left(\sum_a \tilde{w}^a \right)^2 - \left(\sum_c w^c \right)^2 \right] \tag{3.28}$$

and then to perform three Hubbard-Stratonovich transformations, for eliminating the squared sums with the Gaussian variables x, y, z. By factorizing across the reference

replicas indices, computing both $n, s \to 0$ limits and performing rotations when possible to get rid of some Gaussian integrals, the final expression for the entropic density reads

$$\mathcal{G}_S \equiv \frac{\hat{p}}{2} + \lim_{\substack{n\to 0\\ s\to 0}} \partial_s G_S = \int Dx \frac{\sum_{\tilde{w}=\pm 1} e^{\sqrt{\hat{q}}\tilde{w}x} \int Dy \ln 2\cosh\left(\sqrt{\hat{p} - \frac{\hat{t}_0^2}{\hat{q}}}y + \frac{\hat{t}_0}{\sqrt{\hat{q}}}x + (\hat{t}_1 - \hat{t}_0)\tilde{w}\right)}{2\cosh\left(\sqrt{\hat{q}}x\right)}. \tag{3.29}$$

Similarly, for the energetic term, once the term coupled to the parameters t_0 and t_1 has been reorganized, the $n \to 0$ limit computed and the preactivations of both the reference and the constrained replicas integrated, the expression simplifies in the $s \to 0$ limit

$$\mathcal{G}_E \equiv \lim_{\substack{n\to 0\\ s\to 0}} \partial_s G_E = \int Dx \frac{\displaystyle\int Dy\, H\left(\frac{\sqrt{\gamma}\left(\tilde{\kappa} - \sqrt{q}x\right) - (t_1 - t_0)y}{\sqrt{\gamma(1-q) - (t_1 - t_0)^2}}\right) \ln H\left(\frac{\kappa - \frac{t_0}{\sqrt{q}}x - \sqrt{\gamma}y}{\sqrt{1-p}}\right)}{H\left(\frac{\tilde{\kappa} - \sqrt{q}x}{\sqrt{1-q}}\right)} \tag{3.30}$$

having defined $\gamma \equiv p - \frac{t_0^2}{q}$ and being the approximation of the Franz-Parisi potential

$$\phi_{\text{FP}} = -\frac{\hat{p}}{2}(1-p) + t_0\hat{t}_0 - t_1\hat{t}_1 + \mathcal{G}_S + \alpha\mathcal{G}_E. \tag{3.31}$$

Within this ansatz, which assumes the energy landscape is smooth without any ruggedness or valleys and that the problem is homogeneously distributed, lacking of any form of hierarchical structure, it may occur that the entropy becomes negative for certain parameters. This suggests the ansatz is incorrect (i.e. it cannot be for a binary system) and that more sophisticated approaches such as replica symmetry breaking (RSB) ansatz are needed to accurately describe the organization of the solution space. This would allow to account for a multitude of pure states that are not simply related by symmetry transformations. However, at this level we rely on RS approximation, since the number of solutions we are counting is at most $\mathcal{N}_{\boldsymbol{\xi}} = 0$ when the entropy is negative [23].

We can compute the potential (3.31) solving saddle points for the order parameters (the derivatives are straightforward), and by setting the margin of the constrained solutions to be at $\kappa = 0$ we are simply changing the robustness $\tilde{\kappa}$ of the reference configuration. For any value of α for which the problem is SAT ($0 < \alpha < \alpha_c(\kappa)$) there are several phases depending on $\tilde{\kappa}$ if one solves the potential by varying the distance from the reference (Fig. 3.4):

- when $\tilde{\kappa} = 0$ is set solutions are isolated since the Franz-Parisi entropy is always negative in a neighborhood of $d = 0$ and we recover Huang and Kabashima results [23] concerning the *frozen*-1RSB structure of the space of solutions. Starting from a $\tilde{\kappa} = 0$ solution we have to move to a finite distance $d^\star$ to encounter the next one. Here $\phi_{\text{FP}}(d)$ has only one maximum with positive entropy at large distances (this maximum is present for all the values of $\tilde{\kappa}$) which is located at the typical distance d_{typ} between solutions with margin $\tilde{\kappa}$ and κ.

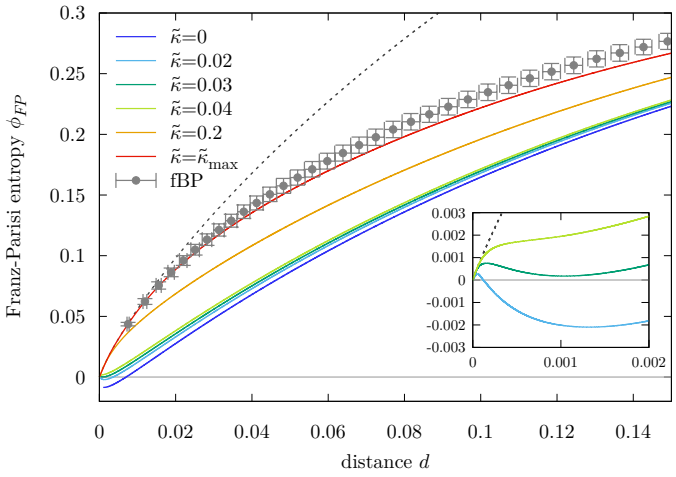


Figure 3.4: Local entropy profiles (with $\kappa = 0$) of typical solutions at $\alpha = 0.5$ as a function of the distance, for various values of $\tilde{\kappa}$. The dashed line is the geometric upper bound obtained by counting all the configurations at $\alpha = 0$. The inset shows a detail of the three curves in the small d range with colors corresponding to the main plot. Points with error bars show the results of numerical experiments (10 samples at $N = 2001$) obtained with the focusing-BP algorithm. Reprinted from [90].

- For $0 < \tilde{\kappa} < \tilde{\kappa}_{\max}$ instead there exists always a neighborhood of $d = 0$ where the average local entropy is positive, suggesting that typical solutions with non-zero margin are surrounded by an exponential number of solutions with zero margin. If one looks at the inset of Fig. 3.4, the local entropy is nearly indistinguishable from the total number of configurations at that distance (the dashed binomial curve obtained for the unconstrained system with $\alpha = 0$), meaning that up to a small, but still $\mathcal{O}(N)$ Hamming distance, almost all configurations around the reference are themselves solutions, i.e. $\tilde{\kappa} > 0$ references belong to dense clusters with characteristic size depending on $\tilde{\kappa}$.

- Again looking at the inset, there exists a threshold $\tilde{\kappa}_{\min}(\alpha) > 0$ such that if $0 < \tilde{\kappa} < \tilde{\kappa}_{\min}(\alpha)$ the local entropy is negative in an interval of distances $d \in [d_1, d_2]$ not containing the origin. For that, no solutions can be found in that spherical shell, causing low margin minimizers to belong to isolated clusters.

- On the contrary, by increasing the margin, another regime for $\tilde{\kappa}_u > 0$ can be found such that if $\tilde{\kappa}_{\min}(\alpha) < \tilde{\kappa} < \tilde{\kappa}_u$ the local entropy is positive at every distance but non monotonic. Indeed, $\phi_{FP}(d)$ develops a secondary maximum at short distances, implying that references with such margin are immersed in regions with finite characteristic size and poses them having an entropy barrier when $\phi_{FP}(d)$ changes concavity.

- Finally, for high margin solutions $\tilde{\kappa} > \tilde{\kappa}_u(\alpha)$ where $\tilde{\kappa}_u(\alpha) < \tilde{\kappa}_{\max}(\alpha)$ the local entropy is monotonic as a function of d, with only one maximum for large

distances. This implies that commonly found solutions with sufficiently large $\tilde{\kappa}$ are likely to be found within dense clusters that appear lacking of a distinct characteristic size, i.e. they could possibly span across vast scales—the high local entropy regions.

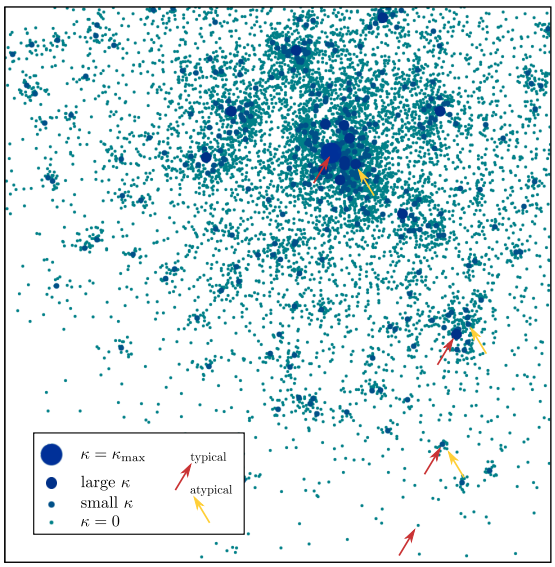


Figure 3.5: Representation of a portion of the space of network configurations. The dots represent solutions (zero-error configurations) with different κ margins. Red arrows indicate four examples of typical solutions. Yellow arrows indicate three examples of the type of atypical solutions found around the typical ones with a larger margin. Reprinted from [90].

The picture so far confirms that, although solutions at a given margin $\tilde{\kappa} > 0$ are isolated from solutions at the same margin by construction [23] (like solutions at $\kappa = 0$ are isolated from solutions at the same $\kappa = 0$), it is not true that they are absolutely isolated if within a distance $\mathcal{O}(N)$ there are solutions with margin $\kappa < \tilde{\kappa}$. Indeed, if one imagines choosing a sufficiently high margin value $\tilde{\kappa} > \kappa_u$ for the reference and spanning the entire range of $0 < \kappa < \tilde{\kappa}$, the spatial arrangement of solutions with different margins in the landscape will turn out to be that in Fig. 3.5. In the sketch, every dots represents solutions (zero-error configurations) at a given constrained density (α) and the larger the dots are, the higher is the associated margin. As clarified by analytical results, coalescence of minima with high margin results in dense regions of solutions over long distances. However, it is important to stress that when we are accounting for $\boldsymbol{w}(\kappa = 0)$ sampled from a probability distribution with a distance constraint, we are changing the measure from the equilibrium one (3.4) and so these solutions will be *atypical* wrt to (3.4). As the number of patterns increases, the solutions thin out, their margin gets smaller, and above some threshold in α the large, connected structure break up and eventually disappear.

Proceeding by analogy with both the transition from weak to strong clustering in Sec. 1.2.3 or the replica symmetry breaking in Sec. 1.2.2, it is not surprising to think that the threshold density at which the breaking occurs makes convergence difficult

for algorithms. Yet, some kind of correspondence between the wide and flat minima of Chapter 2 and the high-margin solutions remains to be defined. For this reason, we performed numerical experiments by sampling solutions found with the focusing-BP algorithm (see 2), which by design seeks maximally dense solutions. By measuring their average local entropy using Belief Propagation (BP), one can compare the fBP curve with that of the most robust solution at a given α (Fig. 3.4). The points agree with $\kappa_{\max}$ at short distances, while the profile overestimates the maximum margin at large distances. This is consistent with what one might expect: putting a bias on the measure to maximize local entropy (2), in fact, is not the same as sampling only solutions with a given margin value, and it is the first method that, while remaining general, gives a more precise estimate of how much a solution is surrounded. On the contrary, however, characterizing the landscape according to the robustness of different minimizers makes the approach more accessible and offers a new way to understand what links the stability of solutions towards perturbations of patterns with their generalization performance (Chapter 4).

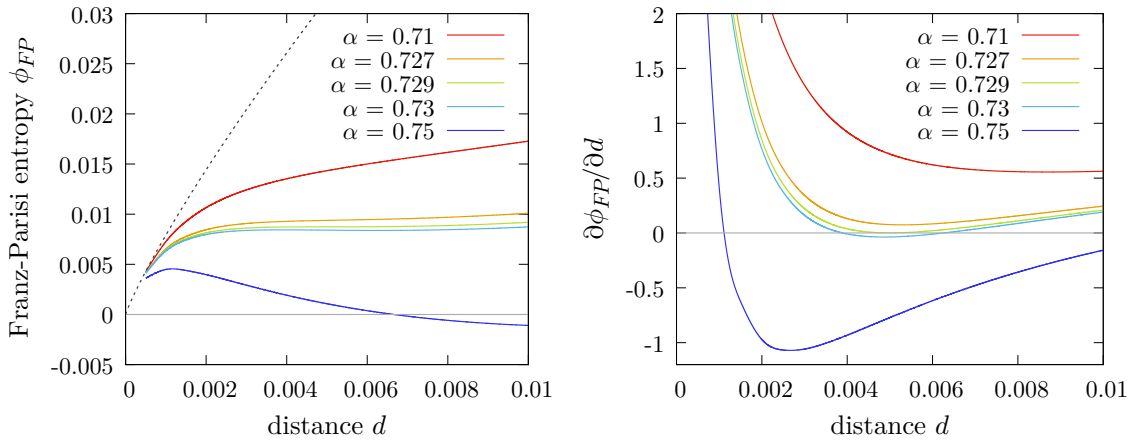


Figure 3.6: Local entropy profiles (with $\kappa = 0$) of typical maximum margin solutions (left panel) and its derivative (right panel) as a function of the distance, for different values of α. For $\alpha = \alpha'_u \simeq 0.729$ the local entropy starts to be non-monotonic. Reprinted from [90].

Let us define the value α'_u as the largest α for which the large-scale phase exists. It is characterized by the property $\tilde{\kappa}_u(\alpha'_u) = \tilde{\kappa}_{\max}(\alpha'_u)$, which is the density value for which the maximum margin curve is no longer monotonic, developing a maximum at small d. Indeed, if the curve is not monotonic for $\tilde{\kappa}_{\max}$, there is no hope for it to be so for any smaller value of margin, being the space left with "isolated balls". It is easy to access α'_u value since by looking at local entropy of $\kappa_{\max}$ for increasing α, one will encounter the threshold at which the derivative $\partial\phi_{FP}/\partial d$ with respect to the distance develops a new zero (Fig. 3.6). For binary Perceptron $\alpha'_u \simeq 0.729$ and the entropy becomes negative at larger $\alpha > \alpha'_u$ in a given range of distances, confirming that at that point the cluster has split into sparsified pieces with entropic barriers. From large deviation analysis instead (Chapter 2) the threshold is slightly bigger $\alpha_u \simeq 0.77$, thus confirming how the second method is an upper bound for the first. The relevance of this threshold for algorithmic convergence will be clarified in

the next chapter; for now, let's just focus on characterizing the solution space by sampling with different margins.

3.2 Non-convex networks with multiple layers

3.2.1 The storage problem for a Tree Committee machine

As evidence that robust solutions in a non-convex landscape are structured with the same hierarchy, that is, configurations with a high margin are encircled by solutions with smaller and smaller margins as the distance from them increases, we have repeated the same computations (i.e. the typical entropy of solutions and the computation of the local entropy around a typical solution) for a different architecture that stores random patterns: the tree committee machine (TCM), which is the simplest one hidden layer model of NN. A TCM is a form of a feedforward neural network that employs a hierarchical, tree-like structure. Each node in this tree is typically a Perceptron, giving rise to a "committee" of computational units that implement an early form of ensemble learning: the overall output is made by aggregating individual decisions of each committee. So, suppose we have K non-overlapping perceptrons with N/K weights each, and a second layer of *fixed* weights c_l, $l = 1, \dots, K$. On the output of every of the K perceptrons a point-wise activation function $g(\cdot)$ is also applied. For a given pattern $\boldsymbol{\xi}^\mu$, the output is

$$\sigma_{\text{out}} = \text{sign}\left(\frac{1}{\sqrt{K}}\sum_{l=1}^{K} c_l g(x_l)\right) \tag{3.32}$$

being x_l the preactivation of each perceptron l

$$x_l \equiv \sqrt{\frac{K}{N}}\sum_{i=1}^{N/K} w_{li}\,\xi_{li}^\mu \tag{3.33}$$

For synthetic dataset with Gaussian inputs ξ_i^μ and Rademacher outputs σ^μ, the computation can be carried out keeping generic the activation and the replicated partition function for the error-counting loss reads

$$\langle\langle \mathbb{Z}^n\rangle_{\boldsymbol{\xi}}\rangle_\sigma = \mathbb{E}_{\boldsymbol{\xi},\sigma}\int\prod_a d\mu(\boldsymbol{w}^a)\prod_{al}\delta\left(\frac{N}{K}-\sum_i (w_{li}^a)^2\right)\prod_{\mu a}\Theta\left(\frac{\sigma^\mu}{\sqrt{K}}\sum_l c_l g\left(x_l^a\right)-\kappa\right) \tag{3.34}$$

being $d\mu(\boldsymbol{w}) = \prod_{li} dw_{li}\left[\frac{1}{2}\delta(w_{li}-1)+\frac{1}{2}\delta(w_{li}+1)\right]$ the measure over binary weights, the normalization $\sum_i w_{li}^2 = N/K$ and κ the margin as usual. Once the average over the disorder induced by the random realization of inputs has been performed and the usual order parameters introduced, the Gibbs entropy becomes

$$\phi = -\frac{1}{2nK}\sum_{l=1}^{K}\sum_{a,b} q_{ab}^l \hat{q}_{ab}^l + G_S + \alpha\, G_E \tag{3.35}$$

with

$$G_S = \frac{1}{nK}\sum_l \ln \int \prod_a d\mu(w^a)\, e^{\frac{1}{2}\sum_{ab}\hat{q}^l_{ab}w^a w^b} \tag{3.36a}$$

$$G_E = \frac{1}{n}\mathbb{E}_\sigma \ln \int \prod_{la} \frac{du^a_l\, d\hat{u}^a_l}{2\pi} \prod_a \Theta\left(\frac{\sigma}{\sqrt{K}}\sum_l c_l\, g(u^a_l) - \kappa\right) e^{-i\sum_{al}\hat{u}^a_l u^a_l - \frac{1}{2}\sum_l\sum_{ab} q^l_{ab}\hat{u}^a_l\hat{u}^b_l} \tag{3.36b}$$

being now the terms depending on $q^l_{ab} = \frac{K}{N}\sum_i w^a_{li}w^b_{li}$ the overlap between subperceptrons l of different replicas a and b. It is important to stress that in the scenario, since the fields are non overlapping and each hidden neuron sees a portion of the pattern, there is no correlation between hidden units in the same architecture.

Replica-symmetric Ansatz

For all activation functions, we are interested in the problem in the large K limit where $\frac{K}{N} \to 0$, which allows to invoke central limit theorem and leads to crucial simplification of the expressions [92–95]. Indeed, the activations for each hidden neuron are Gaussian and the expression of G_E, which accounts for the fraction of solutions of the problem, will only depend on the first two moments of the activation function $g(\cdot)$ [96, 97].

Assuming solutions of the form $q^l_{ab} = \delta_{ab} + (1-\delta_{ab})q$ for all l, a, b in the large K limit and the same for the coniugated parameters, the expressions (3.36) become

$$\phi_{RS} = -\frac{\hat{q}}{2}(1-q) + \mathcal{G}_S + \alpha\,\mathcal{G}_E \tag{3.37a}$$

$$\mathcal{G}_S = \int Dx\ \ln 2\cosh\left(\sqrt{q}x\right) \tag{3.37b}$$

$$\mathcal{G}_E = \int Dx \ln H\left(\frac{\kappa - \sqrt{\Delta_q - \Delta_0}x}{\sqrt{\Delta_1 - \Delta_q}}\right). \tag{3.37c}$$

In the last identity we have defined the effective order parameters in the $K \to \infty$ limit

$$\Delta_a \equiv \Delta(a) \equiv \int Dx \left[\int Dy\, g\left(\sqrt{a}x + \sqrt{1-a}y\right)\right]^2 \tag{3.38}$$

which, as anticipated, gives rise of quantities proportional to the mean and variance of the activations. Solving saddle points to get typical values for the order parameters, the volume of the space of solutions at margin κ in the typical case turns out to be completely defined. As a consequence, to quantify what is the local entropy of each solution at a given margin, it only remains to compute the order parameters related to the configurations constrained at finite distance d or overlap t_1 from the reference. Indeed, the reference configuration is extracted as before from the uniform measure

at zero temperature

$$P_{\boldsymbol{\xi},\sigma}(\tilde{\boldsymbol{w}}) = \frac{\prod_\mu \Theta\left(\frac{\sigma^\mu}{\sqrt{K}}\sum_l c_l g\left(\sqrt{\frac{K}{N}}\sum_i \tilde{w}^a_{li}\xi^\mu_{li}\right)-\kappa\right)}{\int d\mu(\tilde{\boldsymbol{w}}')\prod_l \delta\left(\frac{N}{K}-\sum_i(\tilde{w}'_{li})^2\right)\prod_\mu \Theta\left(\frac{\sigma^\mu}{\sqrt{K}}\sum_l c_l g\left(\sqrt{\frac{K}{N}}\sum_i \tilde{w}'_{li}\xi^\mu_{li}\right)-\kappa\right)}. \tag{3.39}$$

Franz-Parisi potential

Given the probability distribution in Eq. (3.39) over the set of configurations satisfying all the constraints with a margin κ, with a little effort we can compute the local entropy (Franz-Parisi potential) of $\tilde{\boldsymbol{w}}$ averaged over possible realizations of patterns and labels as defined in Eq. (3.17)

$$\phi_{FP}(t_1) = \frac{1}{N}\lim_{s\to 0}\lim_{n\to 0}\frac{\partial}{\partial s}\left\langle \int d\mu(\tilde{w})\, P_{\boldsymbol{\xi},\sigma}(\tilde{w})\,\mathcal{N}^n_{\boldsymbol{\xi},\sigma}(\tilde{w}^{a=1},t_1)\right\rangle_{\boldsymbol{\xi},\sigma} = \frac{1}{N}\lim_{s\to 0}\lim_{n\to 0}\frac{\partial}{\partial s}\langle Z^{n,s}_{FP}\rangle_{\boldsymbol{\xi},\sigma}. \tag{3.40}$$

Using $a,b=1,\dots,n$ and $c,d=1,\dots,s$ at the two sets of replica indices, the order parameters

$$q^l_{ab} = \frac{K}{N}\sum_i \tilde{w}^a_{li}\tilde{w}^b_{li} \tag{3.41a}$$

$$p^l_{cd} = \frac{K}{N}\sum_i w^c_{li}w^d_{li} \tag{3.41b}$$

$$t^l_{ac} = \frac{K}{N}\sum_i \tilde{w}^a_{li}w^c_{li} \tag{3.41c}$$

refer, as in the volume computation, to the overlap between subperceptrons in different replicas since the weights of the second layer are fixed. Imposing the same RS ansatz as we did in Eq. (3.26), and collecting in the potential the usual contributions, one realizes that the entropic term does not change from that of the Perceptron in Eq. (3.29). This is because it represents the logarithm of the number of solutions $\boldsymbol{w}$ at overlap t_1 from $\tilde{\boldsymbol{w}}$ for $\alpha=0$, where there are no constraints induced by the training set. Indeed, the entropic density is affected only by the binary nature of weights in the large K limit

$$\mathcal{G}_S = \int Du\,\frac{\sum_{\tilde{W}=\pm 1} e^{\tilde{W}\sqrt{\tilde{q}}x}\int Dv \ln\left[2\cosh\left(\sqrt{\hat{p}-\frac{\hat{t}_0^2}{\hat{q}}}v+\frac{\hat{t}_0}{\sqrt{\hat{q}}}u+\left(\hat{t}_1-\hat{t}_0\right)\tilde{W}\right)\right]}{2\cosh\left(\sqrt{\tilde{q}}u\right)}. \tag{3.42}$$

The energetic term is instead more complex, being necessary to perform the $K \to \infty$ limit also

$$\begin{aligned} G_E =& \mathbb{E}_\sigma \ln \int \prod_l Dx_l\, Dz_l \left[\int \prod_l \frac{d\lambda_l\, d\hat{\lambda}_l}{2\pi} \Theta\left(\frac{\sigma}{\sqrt{K}} \sum_l c_l\, g(\lambda_l) \right) \prod_l e^{i\left(\lambda_l + \sqrt{q-t_0} x_l + \sqrt{t_0} z_l\right)\hat{\lambda}_l - \frac{1-q}{2}\hat{\lambda}^2} \right]^{n-1} \\ & \times \int \prod_l \frac{d\lambda_l\, d\hat{\lambda}_l}{2\pi} \int \prod_{lc} \frac{du_l^c\, d\hat{u}_l^c}{2\pi} \Theta\left(\frac{\sigma}{\sqrt{K}} \sum_l c_l g(\lambda_l) \right) \prod_c \Theta\left(\frac{\sigma}{\sqrt{K}} \sum_l d_l g(u_l^c) \right) \\ & \times \prod_l e^{-\frac{1-q}{2}\hat{\lambda}_l^2 + i\hat{\lambda}_l\left(\lambda_l + \sqrt{q-t_0} x_l + \sqrt{t_0} z_l + i(t_1 - t_0)\sum_c \hat{u}_l^c\right) - \frac{1-p}{2}\sum_c (\hat{u}_l^c)^2 - \frac{p-t_0}{2}\left(\sum_c \hat{u}_l^c\right)^2 + i\hat{u}_l^c (u_l^c + \sqrt{t_0} z_l} \end{aligned}$$

where we have defined

$$Z \equiv Z(\{x_l\}, \{z_l\}) = \int \prod_l D\lambda_l\, \Theta\left(\frac{\sigma}{\sqrt{K}} \sum_l c_l g(\sqrt{1-q}\lambda_l - \sqrt{q-t} x_l - \sqrt{t} z_l) \right). \tag{3.43}$$

Inserting Hubbard-Stratonovich transformations to decouple over c, performing change of variables and rotations to get rid of some integrals, and in the end performing the infinite K limit, the energetic term with effective order parameters Σ_0, Σ_1, D_0, D_1, Δ, Δ_3 reads

$$\mathcal{G}_E = \int Dx\, \frac{\displaystyle\int Dy\, H\left(\frac{\sqrt{\Gamma}\left(\tilde{\kappa} - \sqrt{\Sigma_0} x\right) - D_1 y}{\sqrt{\Gamma\Sigma_1 - D_1^2}} \right) \ln H\left(\frac{\kappa - \frac{D_0}{\sqrt{\Sigma_0}} x - \sqrt{\Gamma} y}{\sqrt{\Delta_3}} \right)}{H\left(\dfrac{\tilde{\kappa} - \sqrt{\Sigma_0} x}{\sqrt{\Sigma_1}} \right)} \tag{3.44}$$

where $\Gamma \equiv \Delta - \frac{D_0^2}{\Sigma_0}$. As before, order parameters depend on the choice of the activation function and are given by

$$\Delta = \int Dx \left(\int Du\, g\left(\sqrt{1-p} u + \sqrt{p} x \right) \right)^2 - \left(\int Dx\, g(x) \right)^2 \tag{3.45a}$$

$$\Delta_3 = \int Dx \left[g(x) - \left(\int Du\, g\left(\sqrt{1-p} u + \sqrt{p} x \right) \right)^2 \right] \tag{3.45b}$$

$$\Sigma_0 = \int Dx \left(\int D\lambda\, g(\sqrt{1-q}\lambda + \sqrt{q} x) \right)^2 - \left(\int Dx\, g(x) \right)^2 \tag{3.45c}$$

$$\Sigma_1 = \int Dx \left[g^2(x) - \left(\int D\lambda\, g(\sqrt{1-q}\lambda + \sqrt{q} x) \right)^2 \right] \tag{3.45d}$$

$$D_0 = \int Dx D\lambda\, g(\lambda) \left[g\left(t_0 \lambda + \sqrt{1-t_0^2} x \right) - g(x) \right] \tag{3.45e}$$

$$D_1 = \int Dx D\lambda\, g(\lambda) \left[g\left(t_1 \lambda + \sqrt{1-t_1^2} x \right) - g\left(t_0 \lambda + \sqrt{1-t_0^2} x \right) \right] \tag{3.45f}$$

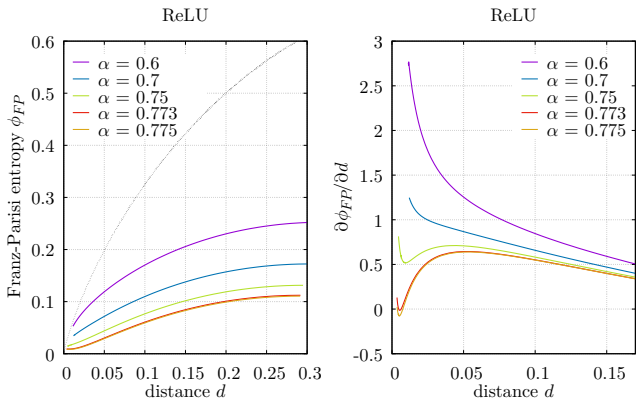


Figure 3.7: *ReLU* activation function: for $\alpha = \alpha'_u \simeq 0.773$ the local entropy starts to be non-monotonic. Reprinted from [90].

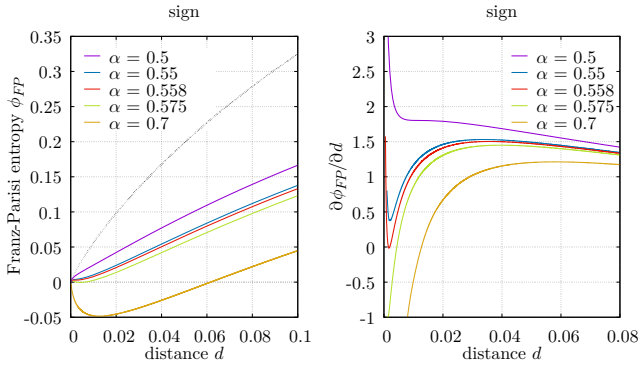


Figure 3.8: *Sign* activation function: for $\alpha = \alpha'_u \simeq 0.558$ the local entropy starts to be non-monotonic. Reprinted from [90].

Specializing to some standard activation functions such as Rectified Linear Unit (ReLU) or the sign, all the integrals in Eq. (3.45) can be solved analytically. Concerning the potential expression

$$\phi_{\text{FP}} = -\frac{\hat{p}}{2}(1-p) + t_0\hat{t}_0 - t_1\hat{t}_1 + \mathcal{G}_S + \alpha\mathcal{G}_E \tag{3.46}$$

it is now similar to the one obtained for the Perceptron (Eq. (3.31)). We show in Fig. 3.7 and in Fig. 3.8 the local entropy profiles (with $\kappa = 0$) of the maximum margin solutions for both $g(x) = \text{ReLU}(x) = \max(0, x)$ and $g(x) = \text{sign}(x)$ activations, and the corresponding plot of the derivative with respect to the Hamming distance. The behavior of the curves for increasing values of α is the same as the picture described for the simpler Perceptron, confirming that a stable attribute of non-convex landscapes is their composition of regions with closely connected solutions when $\alpha < \alpha'(u)$. The monotonic to non-monotonic transition is $\alpha'_u \simeq 0.773$ for the ReLU whereas it has a much lower value $\alpha'_u \simeq 0.558$ for the sign activation function.

3.3 Numerics

The picture so far aligns with experiments conducted on deep multi-layer perceptrons (MLPs) using various architectures and real-world datasets. Specifically, if we alter dataset sizes and, for each size, train a network multiple times with different random initial conditions, storing the distances between final configurations, we can built up a statistics concerning different minimizers and their arrangement at the bottom of the landscape. Notably, these configurations are consistently distant from each other. When the number of patterns is low enough, the training error is of course zero, until reaching an algorithmic threshold where the dataset size exceeds the network capacity for flawless memorization. Even at this threshold, distinct solutions remain significantly separate, suggesting the existence of multiple minimizers at the algorithmic threshold.

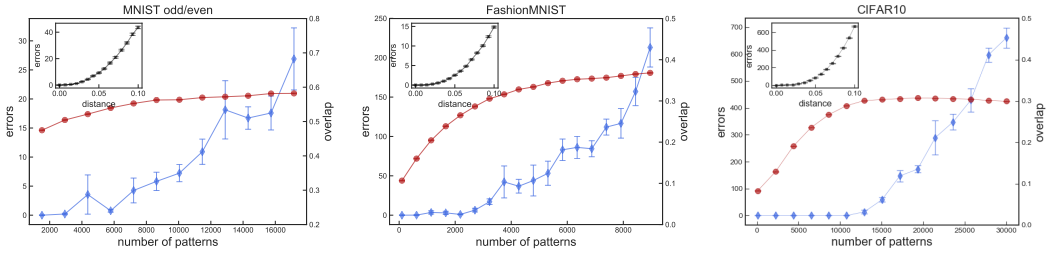


Figure 3.9: Number of train errors (blue points, left y-axis) and mean overlap of pairs of solutions (red points, right y-axis) as a function of the training set size, for different architectures and datasets. The overlaps between solutions obtained with different random initializations is well below 1 even as the model is barely able to fit all of the patterns. The inset shows local energy profiles right below the algorithmic threshold, i.e. at the largest dataset size for which the error is still zero. Reprinted from [90].

We refer to three distinct architectures: a two hidden layer MLP trained on MNIST with odd-even digits, the same architecture on FashionMNIST, and a three hidden layer MLP on CIFAR10. To train networks with binary weights, we use a standard BinaryNet approach [98], omitting biases and batch normalization, and employing cross-entropy loss with SGD optimization. If different solutions are obtained with different initializations, before computing the average overlap between solution pairs, it is necessary to account for network symmetries, such as neuron permutation within the same layer and sign reversal symmetry. To do so, one can match each solution pair, considering the sign of neuron weights, beginning from the lowest layer and adjusting subsequent layers to break these symmetries before computing overlaps (more details in 4.3.2 from [40]). Results, depicted in Fig. 3.9, are consistent across all tested architectures and datasets. They reveal the presence of numerous solutions at the algorithmic threshold, as evidenced by the average overlap consistently being well below 1. Plus, in scenarios where the number of parameters allows for perfect pattern fitting, solutions are not isolated, as indicated by their local energy profiles' typical convex shape (*inset* of each subplot), derived by perturbing a solution through random weight flipping, and determining the number of errors at various distances

proportionals to the strength of the perturbation. The curves are precisely obtained by altering the flip probability and averaging over several noise realizations, similarly to what has been done in [49]. As a consequence, if as in Fig. 3.9 there is a range of distances for which the error remains flat, this means that solutions found for each of the problems are surrounded at an extensive distance.

Chapter 4

Atypical phase transition in the over-parameterized regime

In order to understand the complexity of learning based on the minima's geometry, it is crucial to do more than just investigate the entropy linked to various configurations at the bottom of the landscape. It is also necessary to consider a learning context that incorporates specific fundamental features and allows one to characterize performances. So far, in discussing supervised learning, we have limited ourselves to the treatment of cases where the data in the training set are chosen in a worst-case manner. The inputs are indeed i.i.d. elements drawn from some probability distribution while the labels are random in themselves, without any correlation between patterns or even between each pattern and its associated label. This is highly unrealistic, as the contexts in which neural networks are most successful today are those where not only the data are labeled through a generative recipe that can be enforced by introducing some *teacher* structure, but the data themselves are characterized by complex correlations. Real-world data, especially in domains like image and speech recognition, are inherently structured, meaning that statistical properties of one part of the data can inform or predict properties of another part, a feature that NNs exploit for pattern recognition and prediction. As a consequence, despite existing in a high-dimensional space, data often reside on low-dimensional manifolds, that stand as the actual input spaces, or the *worlds* of our problems.

The random feature model (RFM) in machine learning, is a simple way to represent the idea that the model captures the structure in the data by assuming that the observable data points are generated from a low-dimensional latent space, which is then transformed into the observed high-dimensional space through some nonlinear process. The purpose of the present Chapter is to delve into the analysis of the minimizers' geometry for this paradigmatic model, which allows to inspect the fallout of overparameterization when the student has a more expressive architecture than the teacher. In practice, since we will assume the rule that generates the label from the patterns (i.e. the teacher) depends in a simple way from the latent representation, to which however the student network does not have access, we will be able to study what are the consequences of increasing the dimensionality of the student network wrt to the data dimension. In this setting we will be able to understand what is the impact for performance of the highly robust minimizers described in Chatpter 3, and to

distinguish among two phase transitions as we vary the overparameterization degree of the problem. The first will be the information theoretic (SAT/UNSAT transition) at which zero-error solutions appear and perfect fitting of the data becomes possible; the second instead goes under the name of *atypical* phase transition [99], which coincides with the sharp appearance of highly entropic and highly robust solutions in the phase space. In going trough the results of [99], we will shed light to the gap between information theoretic and computation as introduced in Sec. 1.2.3 but in the typical case, which in the context of binary networks is a purely geometrical bound.

4.1 Model definition and learning setup

Let us consider the problem of learning a dataset as the one described by the joint probability distribution of input-output pairs $\mathcal{D} = \{\boldsymbol{\xi}^\mu, y\mu\}_{\mu=1}^P$ where the rule that generates the labels y^μ according to the patterns can be described by a random, but fixed, neural networks with weights $\boldsymbol{w}^T \in \{-1,+1\}^D$ that we named the *teacher*

$$y^\mu = \text{sign}\left(\frac{1}{\sqrt{D}}\sum_{k=1}^{D} w_k^T \xi_k^\mu\right) \tag{4.1}$$

and that is a binary perceptron in dimension D for the present case. Having defined the generative recipe for the P patterns of the training set $\mathcal{D}$, the canonical teacher-student setup first introduced in [100], is completely defined by introducing a *student*, which represents the learning model whose objective is to recover the parameters of the teacher (i.e. to infer the rule) $\boldsymbol{w}^T$ by minimizing an error function $E(\boldsymbol{w}, \mathcal{D})$ where $\boldsymbol{w} \in \{-1,+1\}^N$ are the student parameters.

If we suppose the student network lies in a different $N-$dimensional space compared to the teacher where $N > D$ and that the problem is not matched, i.e. the student has in general a different architecture from Eq (4.1), we can image being in the situation in which the input patterns $\tilde{\boldsymbol{\xi}}$ it has access to are highly redundant, since it does not know about the low dimensional manifold of data in which $\boldsymbol{\xi}$ represents the coordinate of each point. Plus, it is of course not aware about the rule that generates the labels. If the student model is a RFM then the ouputs are obtained by

$$y_{\text{out}}^\mu \equiv \text{sign}\left(\frac{1}{\sqrt{N}}\sum_{i=1}^{N} w_i \tilde{\xi}_i^\mu\right) \tag{4.2}$$

where each pattern comes from the hidden dimension D and it is projected as a pattern $\tilde{\boldsymbol{\xi}}$ on a visible feature space of dimension N through

$$\tilde{\xi}_i^\mu = \sigma\left(\frac{1}{\sqrt{D}}\sum_{k=1}^{D} F_{ki}\xi_k^\mu\right). \tag{4.3}$$

In Eq. (4.3), if the matrix of random inputs is $\boldsymbol{\xi} \in \mathbb{R}^{P\times D}$, then the matrix of features is $F \in \mathbb{R}^{D\times N}$ with random elements to make the problem analytically tractable, while $\sigma(\cdot)$ is a non-linear function acting component-wise.

4.1.1 Gaussian Equivalence Principle

In the following we will suppose $\{\xi_k\}_{k=1}^D$ to be i.i.d. Gaussian random variables distributed as $\mathcal{N}(0,1)$ for each pattern and if we define $u_i,\ i=1,\dots,N$ as the N variables that are linear superpositions of ξ_k, then

$$u_i = \frac{1}{\sqrt{D}} \sum_k F_{ki}\xi_k \tag{4.4}$$

are also Gaussian variables with zero mean and a covariance matrix

$$U_{ij} = \mathbb{E}[u_i u_j] = \frac{1}{D} \sum_k F_{ki}F_{kj} \tag{4.5}$$

where, if in turn the elements of F are $\mathcal{O}(1)$ in the thermodynamic limit $N, D, P \to \infty$ (keeping $\alpha \equiv P/N$ and $\alpha_D \equiv D/N$ fixed), satisfyng

$$\frac{1}{D}\sum_{k=1}^{D} F_{ki}^2 = 1, \qquad \forall i \tag{4.6a}$$

$$\frac{1}{\sqrt{D}}\sum_{k=1}^{D} F_{ki}F_{kj} = \mathcal{O}(1) \qquad \forall i \neq j \tag{4.6b}$$

The second requirement suggests that two different sub-perceptrons in the student model are almost uncorrelated, so the covariances of the u_i variable scale in the limit as

$$\mathbb{E}[u_i^2] = 1; \quad \mathbb{E}[u_i u_j] = \mathcal{O}(\frac{1}{\sqrt{D}}). \tag{4.7}$$

We can also define the preactivations of teacher and student respectively as

$$v = \frac{1}{\sqrt{D}}\sum_{k=1}^{D} \xi_k w_k^T, \quad \lambda = \frac{1}{\sqrt{N}}\sum_{i=1}^{N} w_i\, \sigma(u_i). \tag{4.8}$$

In the simple case where $\tilde{\xi}_i = \sigma(u_i)$ are element-wise i.i.d. Gaussian variables if $\sigma(x) \equiv x$ is the identity, the central limit theorem (CLT) in the large N, D, P limits ensures joint Gaussianity for both (v, λ), but for generic $\sigma(x)$ the inputs will not be normally distributed, having a non-trivial Covariance matrix. However, the *Gaussian Equivalence Principle* (GEP) [101–105] ensures that, as long as it is true that

$$s_{k_1,\dots,k_s} \equiv \frac{1}{\sqrt{N}}\sum_{i=1}^{N} w_i \dots w_i F_{k_1 i} \dots F_{k_s i} \equiv \mathcal{O}(1), \quad \forall n, s \geq 1 \tag{4.9}$$

i.e. as long as the student weights w_i do not align "too much" with the weights of the generator F_{ki}, remaining $s_k \sim \mathcal{O}(1)$ in the limit, then the sums of weakly correlated variables in Eq. (4.8) will be themselves Gaussian with mean

$$\mathbb{E}[u] = 0, \quad \mathbb{E}[\lambda] = \frac{\mu_0}{\sqrt{N}}\sum_i w_i \tag{4.10}$$

and covariance

$$\mathbb{E}\left[\lambda_\mu^a \lambda_\mu^b\right] - \mathbb{E}\left[\lambda_\mu^a\right]\mathbb{E}\left[\lambda_\mu^b\right] = \mu_\star^2 \frac{1}{N}\sum_i w_i^a w_i^b + \mu_1^2 \frac{1}{D}\sum_{k=1}^D s_k^a s_k^b \tag{4.11a}$$

$$\mathbb{E}\left[v_\mu \lambda_\mu^a\right] = \mu_1 \frac{1}{D}\sum_{k=1}^D s_k^a w_k^T \tag{4.11b}$$

$$\mathbb{E}[v^2] = \frac{1}{D}\sum_{k=1}^D \left(w_k^T\right)^2 = 1 \tag{4.11c}$$

being the expectations taken over the random realization of patterns $\boldsymbol{\xi}$. The non linear function appears through the coefficients

$$\mu_0 = \langle \sigma(u) \rangle \tag{4.12a}$$
$$\mu_1 = \langle u\, \sigma(u) \rangle = \langle \sigma'(u) \rangle \tag{4.12b}$$
$$\mu_2 = \langle \sigma^2(u) \rangle \tag{4.12c}$$
$$\mu_\star^2 = \mu_2 - \mu_1^2 - \mu_0^2 \tag{4.12d}$$

where $\langle \psi(u) \rangle$ stands for the mean value of the function ψ when $u \sim \mathcal{N}(0,1)$ is a Gaussian variable. As a consequence, GEP posits that statistical behavior of the student model is equivalent to a Gaussian covariate model

$$\tilde{\xi}_i^\mu = \sigma\left(\frac{1}{\sqrt{D}}\sum_{k=1}^D F_{ki}\xi_k^\mu\right) = \mu_0 + \frac{\mu_1}{\sqrt{D}}\sum_{k=1}^D F_{ki}\xi_k^\mu + \mu_\star \eta_i^\mu \tag{4.13}$$

in which each projected pattern $\tilde{\boldsymbol{\xi}}^\mu$ is a *linear* combination of the patterns components ξ_k^μ plus Gaussian noise η, the analytical treatment of the learning problem becomes possible due to the simplification. In Eq. (4.13), the strength of the noise depends on the degree of non-linearity of the activation function σ [102, 103] (see Appendix A.1 for the detailed calculation of the first two moment of Eq. (4.8) via the replica computation).

4.2 Geometry of minima vs overparameterization

Having clarified that it is possible to address analytically the problem under certain conditions, all that remains is to define the loss function for the setting. Let us consider as loss the primitive that counts the number of misclassified patterns in the training set $\mathcal{D}$, whose stability Δ^μ

$$\Delta^\mu(\boldsymbol{w}) \equiv y^\mu \lambda^\mu(\boldsymbol{w}) \tag{4.14}$$

is the product of the pre-activation of the output unit $\lambda^\mu(\boldsymbol{w})$ of the student with the binary label of the pattern $y^\mu = \pm 1$. The loss function per pattern is then given by

$$\ell_{NE}(-\Delta^\mu(\boldsymbol{w}); \kappa) = \Theta(-\Delta^\mu(\boldsymbol{w}) + \kappa) \tag{4.15}$$

being $\Theta(\cdot)$ the usual Heaviside step function and κ the margin of the problem. In order to compute the averaged free entropy of the model $\phi = \lim_{N,D,P\to\infty} \frac{1}{N}\langle \ln Z\rangle_{\boldsymbol{\xi},F}$, which as in Chapter 3 it is necessary if one wants to explore different solutions in the phase space by biasing the measure with κ, we are interested in the large-size limit where the calculations can be performed by asymptotic methods while keeping fixed the densities

$$\alpha \equiv \frac{P}{N}, \quad \alpha_T \equiv \frac{P}{D}, \quad \alpha_D \equiv \frac{D}{N}. \tag{4.16}$$

All these quantities will play a role in determining in which part of the phase diagram of the model we will be: the first (α) represents the degree of overparameterization of the student in dimension N wrt the data points in dimension D - we will define the model as overparameterized as soon as $\alpha < 1$; the second (α_T) is instead the degree of information the teacher is giving to the student about the problem - the $\alpha_T \to 0$ limit will reduce to the storage case in which the labels in the training set will no longer be correlated with the patterns; finally, the third (α_D) stands for the degree of correlation of the patterns $\tilde{\boldsymbol{\xi}}$ (4.3) the student has access to - the extreme limit $\alpha_D \to \infty$ comes down to the case where the projected patterns are no longer correlated.

In general, at generic temperature $\beta = \frac{1}{T}$, the partition function of the model Z reads

$$Z(\beta) = \sum_{\boldsymbol{w}} e^{-\beta \sum_{\mu=1}^{P} \ell_{NE}(-\Delta^\mu(\boldsymbol{w});\kappa)} \tag{4.17}$$

which is the generating function in the variable $e^{-\beta}$ of the number of errors. Since we are only interested in sampling from the uniform measure configurations that are solutions at zero-error, we can consider the large β limit for which Z reduces on counting the global minima, i.e. the $\boldsymbol{w}$ for which all the patterns are being correctly classified with the required robustness

$$Z = \sum_{\boldsymbol{w}} \prod_{\mu} \Theta\left(\Delta^\mu(\boldsymbol{w}) - \kappa\right) \equiv \sum_{\boldsymbol{w}} \mathbb{X}_{\boldsymbol{\xi},F}(\boldsymbol{w};\kappa) \tag{4.18}$$

and the flat measure over all configurations being accordingly

$$P_{\boldsymbol{\xi},F}(\boldsymbol{w};\kappa) = \frac{\mathbb{X}_{\boldsymbol{\xi},F}(\boldsymbol{w};\kappa)}{Z}. \tag{4.19}$$

4.2.1 Averaged free-entropy

By enforcing the definitions of the pre-activations of the teacher and the student as in Eq. (4.8), the replicated action (remember the replica trick to compute ϕ) becomes

$$\mathbb{E}_{\{\boldsymbol{\xi}^\mu,F\}}[Z^n] = \int \prod_{ia} dw_i^a \, P_W(\boldsymbol{w}^a) \int \prod_{\mu} du^\mu \prod_{\mu a} d\lambda_a^\mu \, e^{-\beta \sum_{\mu=1}^{P} \sum_{a=1}^{n} \Theta\left(-\mathrm{sign}(u^\mu)\lambda_a^\mu + \kappa\right)}$$
$$\times \mathbb{E}_{\{\boldsymbol{\xi}^\mu,F\}} \left[\prod_{\mu} \delta\left(u^\mu - \frac{1}{\sqrt{D}} \sum_{k=1}^{D} w_k^T \xi_k^\mu \right) \prod_{\mu a} \delta\left(\lambda_a^\mu - \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^a \, \sigma\left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki} \xi_k^\mu \right) \right) \right] \tag{4.20}$$

where if the distributions of patterns and features are symmetric, we can perform the gauge transformation $\xi_k^\mu \to w_k^T \xi_k^\mu$, $F_{ki} \to w_k^T F_{ki}$, in order to consider $w_k^T = 1$, $\forall k$ without loss of generality, being $\mathbb{E}_{\{\boldsymbol{\xi}^\mu, F\}}$ the average over both the patterns (including the desired output and thus the teacher) and the features.

The average over the disorder of the $\{\boldsymbol{\xi}^\mu\}$ has already been computed by making use of GEP (Appendix A.1) where the multivariate Gaussian distribution of u^μ and λ_a^μ reads

$$P\left(u^\mu, \{\lambda_a^\mu\} \,|F_{ki}, w_i^a\right) = \frac{1}{\sqrt{2\pi \det \Sigma}} e^{-\frac{1}{2}\sum_{a,b=0}^{n} \left(\Upsilon_a^\mu - \rho_a\right)\left(\Sigma^{-1}\right)_{ab}\left(\Upsilon_b^\mu - \rho_b\right)} \tag{4.21}$$

having $\Upsilon_0^\mu \equiv u^\mu$, $\Upsilon_a^\mu \equiv \lambda_a^\mu$, $\forall a = 1, \dots, n$, the mean vector $\rho_0 = 0$ and $\rho_a = \frac{\mu_0}{\sqrt{N}} \sum_i w_i^a$ for $a = 1, \dots, n$. The covariance is instead

$$\Sigma \equiv \begin{pmatrix} 1 & M_a \\ M_a & Q_{ab} \end{pmatrix} \tag{4.22}$$

by introducing the order parameters as

$$M_a = \mu_1 \frac{1}{D} \sum_{k=1}^{D} \left(\frac{1}{\sqrt{N}} \sum_{i=1}^{N} F_{ki} w_i^a \right) w_k^T = \mu_1 \frac{1}{D} \sum_{k=1}^{D} s_k^a w_k^T \equiv \mu_1 r_a \tag{4.23a}$$

$$Q_{ab} = \mu_\star^2 \frac{1}{N} \sum_{i=1}^{N} w_i^a w_i^b + \mu_1^2 \frac{1}{D} \sum_{k=1}^{D} s_k^a s_k^b \equiv \mu_\star^2 q_{ab} + \mu_1^2 p_{ab}. \tag{4.23b}$$

In the definitions of Eq. (4.23) each of the order parameter appear in a form that is slightly different from the one of the usual teacher-student scenario [106]. Since the teacher and student live in spaces with different dimensions, the overlap M_a between them is now the projection of the student weights in the teacher space, while the overlap Q_{ab} has two contributions. The first one q_{ab} is the usual overlap between different replicas of the student, while the second term p_{ab} represents the overlap between the projection (in the teacher space) of the weights of two students with replica indexes a and b. By safely imposing definitions in Eq. (4.23), it becomes easy to perform the average over random Gaussian features. We get a term of the following form

$$\int \prod_{ka} \frac{ds_k^a d\hat{s}_k^a}{2\pi} e^{i\sum_{ka} s_k^a \hat{s}_k^a} \prod_{ki} \mathbb{E}_{F_{ki}} \left[e^{-i\frac{F_{ki}}{\sqrt{N}} \sum_a \hat{s}_k^a w_i^a} \right] = \int \prod_{ka} \frac{ds_k^a d\hat{s}_k^a}{2\pi} e^{i\sum_{ka} s_k^a \hat{s}_k^a - \frac{1}{2}\sum_{ab,k} \hat{s}_k^a \hat{s}_k^b \left(\frac{1}{N}\sum_i w_i^a w_i^b\right)}. \tag{4.24}$$

Now, denoting by $\overline{\cdots}$ the average over both patterns and random features, the final result reads

$$\overline{Z^n} = \int \prod_{a<b} \frac{dq_{ab} d\hat{q}_{ab}}{2\pi} \prod_{a \le b} \frac{dp_{ab} d\hat{p}_{ab}}{2\pi} \prod_a \frac{dr_a d\hat{r}_a}{2\pi} e^{N\phi} \tag{4.25}$$

where

$$\phi = -\sum_{a<b} q_{ab}\hat{q}_{ab} - \frac{\alpha_D}{2}\sum_{ab} p_{ab}\hat{p}_{ab} - \alpha_D \sum_a r_a \hat{r}_a + G_{SS} + \alpha_D G_{SE} + \alpha G_E \tag{4.26a}$$

$$G_{SS} = \ln \int \prod_a dw_a \, P_w(w_a) \, e^{\frac{1}{2}\sum_{a\neq b}\hat{q}_{ab} w_a w_b} \tag{4.26b}$$

$$G_{SE} = \ln \int \prod_a \frac{ds_a d\hat{s}_a}{2\pi} \, e^{i\sum_a s_a\hat{s}_a + \sum_a \hat{r}_a s_a + \frac{1}{2}\sum_{ab}\hat{p}_{ab}s_a s_b - \frac{1}{2}\sum_{ab} q_{ab}\hat{s}_a\hat{s}_b} \tag{4.26c}$$

$$G_E = \ln \int \prod_a \frac{d\lambda_a d\hat{\lambda}_a}{2\pi}\frac{du d\hat{u}}{2\pi} \, e^{iu\hat{u} + i\sum_a \lambda_a\hat{\lambda}_a - \beta\sum_a \Theta(-\mathrm{sign}(u)\lambda_a + \kappa) - \frac{\hat{u}^2}{2} - \frac{1}{2}\sum_{ab} Q_{ab}\hat{\lambda}_a\hat{\lambda}_b - \hat{u}\sum_a M_a\hat{\lambda}_a} \tag{4.26d}$$

and again we have separated in the Gibbs entropy expression different contributions. Notice that G_{SS} is the usual entropic contribution in a perceptron storing random patterns, whereas G_E is the usual energetic contribution in the teacher student setting. G_{SE} is a new term that we call "entropic-energetic" since it depends on both overlaps q_{ab} and conjugated ones $\hat{p}_{ab}$, $\hat{r}_a$. It can be computed analytically, since it contains only Gaussian integrals

$$G_{SE} = -\frac{1}{2}\ln\det\left(\mathbb{I} - q\hat{p}\right) + \frac{1}{2}\sum_{ab}\hat{r}_a\left[\left(\mathbb{I} - q\hat{p}\right)^{-1} q\right]_{ab}\hat{r}_b \,. \tag{4.27}$$

Replica-Symmetric Ansatz

We impose a Replica-Symmetric (RS) ansatz for the order parameters: $q_{ab} = \delta_{ab} + q\,(1-\delta_{ab})$, $\hat{q}_{ab} = \hat{q}\,(1-\delta_{ab})$; $p_{ab} = p_d\delta_{ab} + p\,(1-\delta_{ab})$, $\hat{p}_{ab} = -\hat{p}_d\delta_{ab} + \hat{p}\,(1-\delta_{ab})$ and $r_a = r$, $\hat{r}_a = \hat{r}$, obtaining

$$\mathcal{G}_{SS} \equiv \frac{\hat{q}}{2} + \lim_{n\to 0}\frac{G_{SS}}{n} = \int Dx \, \ln 2\cosh\left(\sqrt{\hat{q}}x\right) \tag{4.28a}$$

$$\mathcal{G}_{SE} \equiv \lim_{n\to 0}\frac{G_{SE}}{n} = -\frac{1}{2}\frac{q}{1-q} - \frac{1}{2}\ln\left[1 + (\hat{p}+\hat{p}_d)(1-q)\right] + \frac{1}{2}\frac{(\hat{p}+\hat{r}^2)(1-q) + \frac{q}{1-q}}{1 + (\hat{p}+\hat{p}_d)(1-q)} \tag{4.28b}$$

$$\mathcal{G}_E \equiv \lim_{n\to 0}\frac{G_E}{n} = 2\int Dx \, H\left(-\frac{Mx}{\sqrt{Q-M^2}}\right)\ln H_\beta\left(\frac{\kappa - \sqrt{Q}x}{\sqrt{Q_d - Q}}\right) \tag{4.28c}$$

in the $n \to 0$ limit. Here we have separated diagonal and off-diagonal contributions $M \equiv \mu_1 r$, $Q \equiv \mu_\star^2 q + \mu_1^2 p$, $Q_d \equiv \mu_\star^2 + \mu_1^2 p_d$, and defined

$$H\left(x\right) \equiv \frac{1}{2}\mathrm{Erfc}\left(\frac{x}{\sqrt{2}}\right) \,, \tag{4.29a}$$

$$H_\beta\left(x\right) \equiv e^{-\beta} + \left(1 - e^{-\beta}\right) H\left(x\right) \,. \tag{4.29b}$$

The free entropy of the system within this ansatz is

$$\phi_{RS} = -\frac{\hat{q}}{2}(1-q) + \frac{\alpha_D}{2}\left(p_d\hat{p}_d + p\hat{p}\right) - \alpha_D r\hat{r} + \mathcal{G}_{SS} + \alpha_D\mathcal{G}_{SE} + \alpha\mathcal{G}_E \,. \tag{4.30}$$

and eventually ϕ can be found using Laplace method that optimizes over eight order parameters

$$\phi = \max_{q,\hat{q},p,\hat{p},p_d,\hat{p}_d,r,\hat{r}} \ \phi_{RS} \tag{4.31}$$

according to the saddle point equations

$$\begin{aligned} q &= 1 - 2\frac{\partial \mathcal{G}_{SS}}{\partial \hat{q}}\,, & p &= -2\frac{\partial \mathcal{G}_{SE}}{\partial \hat{p}}\,, & p_d &= -2\frac{\partial \mathcal{G}_{SE}}{\partial \hat{p}_d}\,, & r &= \frac{\partial \mathcal{G}_{SE}}{\partial \hat{r}}\,, \\ \hat{q} &= -2\alpha_D\frac{\partial \mathcal{G}_{SE}}{\partial q} - 2\alpha\frac{\partial \mathcal{G}_E}{\partial q}\,, & \hat{p} &= -2\alpha_T\frac{\partial \mathcal{G}_E}{\partial p}\,, & \hat{p}_d &= -2\alpha_T\frac{\partial \mathcal{G}_E}{\partial p_d}\,, & \hat{r} &= \alpha_T\frac{\partial \mathcal{G}_E}{\partial r}. \end{aligned} \tag{4.32}$$

Following the same ideas of Chapter 3, we can verify that the entropy ϕ in Fig 4.1 is again a decreasing function of the margin κ for a given α, plus a first intuition about the configurations at zero-error can be gained if we look at the plot from right to left, while we are increasing the degree of overparameterization (i.e. decreasing α). Since the horizontal line at zero entropy highlights the first value in α for which solutions with a given κ starts to appear (remember that since the problem lives on the binary hypercube the entropy is bounded below to zero, signaling the RS ansatz fails when $\phi < 0$), it is clear how if the student is underparameterized ($\alpha > 1$), then the only solutions at the bottom of the landscapes appears to be the low-margin nearly isolated ones; instead, it is only when decreasing α that the high margin solutions begin to emerge.

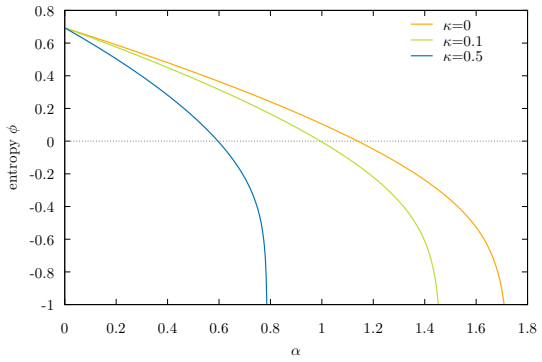


Figure 4.1: Entropy as a function of α for fixed $\alpha_T = 3$ and different values of the margin κ. Reprinted from [99].

As in the simple binary perceptron [22], the *interpolation threshold* or critical capacity $\alpha_c(0)$ is found by looking to the value of α for which the RS free entropy $\phi_{RS}(0)$ of the $\kappa = 0$ solutions vanishes or simultaneously $q \to 1$, which is the disappearing threshold of *all* solutions. We can also study the entropy for a fixed α as we increase the information degree α_T (Fig. 4.2). When the dimension of the teacher is much larger than the number of patterns in the training set $\alpha_T = \frac{P}{D} \ll 0$, the problem becomes that of a student seeing random patterns. Indeed in the $\alpha_T \to 0$ limit (see Appendix A.2.1) the free entropy simplifies as the one of the storage problem derived by Gardner [107, 108], regardless of the non-linearity $\sigma(\cdot)$ used.

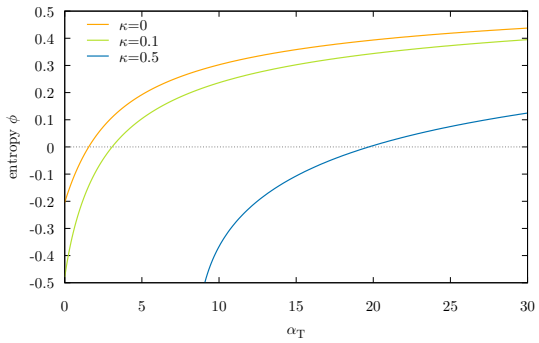


Figure 4.2: Entropy as a function of α_T for fixed $\alpha = 1$ and different values of the margin κ. Reprinted from [99].

Until now, we have indeed changed model and introduced correlations between data, but apart from the external parameters specifying the relation between the teacher and the student, the picture on the sparsity of the high-margin solutions compared to those with low-margin has remained the same. However, because the labels are now related to the patterns, from the solutions of the saddle point equations we can also compute what in general characterizes the performance associated to a given configuration: the generalization error.

4.2.2 The generalization error

Typical case

The generalization error ϵ_g, which is defined as the probability of wrongly classifying a new (unseen) pattern after the learning procedure, can be computed by averaging on the ensemble $\langle\cdot\rangle$ the error related to the new extracted pattern $\{\boldsymbol{\xi}^\star, y^\star\}$

$$\begin{aligned}\epsilon_g &\equiv \langle \mathbb{E}_{\boldsymbol{\xi}^\star} \Theta\left(-y^\star \hat{y}^\star\right)\rangle \\ &= \int du d\lambda\, \Theta\left(-u\lambda\right) \mathbb{E}_{\boldsymbol{\xi}^\star} \left\langle \delta\left(u - \frac{1}{\sqrt{D}}\sum_{k=1}^{D} w_k^T \xi_k^\star\right) \delta\left(\lambda - \frac{1}{\sqrt{N}}\sum_{i=1}^{N} w_i\, \sigma\left(\frac{1}{\sqrt{D}}\sum_{k=1}^{D} F_{ki}\xi_k^\star\right)\right)\right\rangle \\ &= \int \frac{du d\hat{u}}{2\pi}\frac{d\lambda d\hat{\lambda}}{2\pi}\, \Theta\left(-u\lambda\right) e^{iu\hat{u}+i\lambda\hat{\lambda}-\frac{\hat{u}^2}{2}-\frac{1}{2}Q_d\hat{\lambda}^2-M\hat{u}\hat{\lambda}} = 2\int_0^\infty Du\, H\left(-\frac{Mu}{\sqrt{Q_d - M^2}}\right).\end{aligned} \tag{4.33}$$

From a geometrical intuition we expect the generalization of a given student solution $\boldsymbol{w}$ to be related to its alignment with the teacher and indeed

$$\epsilon_g = \frac{1}{\pi}\arccos\left(\frac{M}{\sqrt{Q_d}}\right) \tag{4.34}$$

if we perform the last integral in (4.33). It is nothing but the standard formula of ϵ_g for the classical teacher-student problem, but written in terms of the "projected" overlap

with the teacher M and the "projected" norm of the weights Q_d in the D-dimensional space.

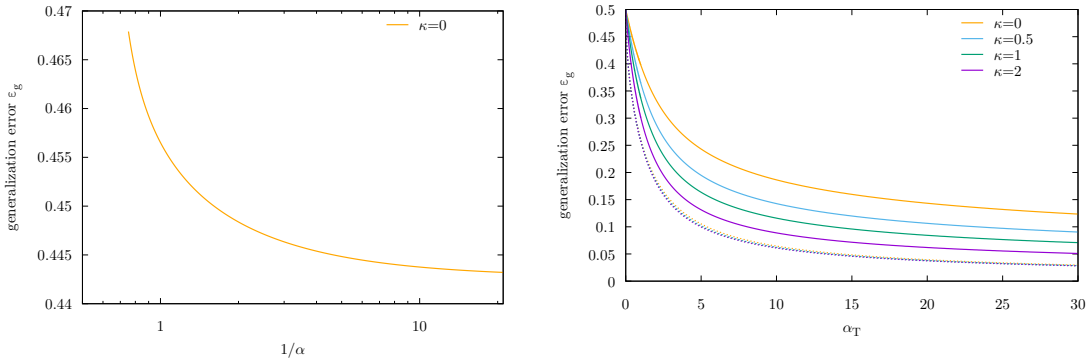


Figure 4.3: (*Left*) Generalization as a function of the overparameterization degree $1/\alpha$ in the log x-axis scale. (*Right*) Generalization error in the large overparameterization limit ($\alpha \to 0$) versus $\alpha_T = P/D$ for different values of the margin κ. The dotted lines represent the corresponding generalization error of the barycenter of typical solutions (see Sec. 4.2.2). Reprinted from [99].

Now, given fixed values of κ and α_T, it is possible to study how much the generalization error improves when increasing the overparameterization degree of the student model. In the left side of Fig. 4.3, ϵ_g as a function of $1/\alpha$ decreases until it reaches a plateau that can be evaluated by imposing the extreme infinite overparameterization limit $\alpha \to 0$. In this regime also $\alpha_D = \frac{\alpha}{\alpha_T}$ is vanishing, and therefore from saddle point equations one can conclude that $\hat{q} \to 0$ and consequently $q \to 0$, meaning that typical solutions are uncorrelated in the space of the students. However there is still information about the teacher, so the corresponding overlap in that space is not zero, but one can rid of conjugated parameters $\hat{p}$, $\hat{p}_d$ and $\hat{r}$ by expressing them in terms of the others. Here, the entropy is tending towards the binomial value, i.e. when there are no constraints present in the problem

$$\phi \simeq \ln 2 + \frac{\alpha}{\alpha_T}\delta\phi \tag{4.35}$$

and

$$\delta\phi = \frac{1}{2}\left(1 - p_d - \frac{r^2}{p_d - p}\right) + \mathcal{G}_{SE} + \alpha_T \mathcal{G}_E \tag{4.36a}$$

$$\mathcal{G}_{SE} = \frac{1}{2}\left(\frac{p}{p_d - p} + \ln(p_d - p)\right) \tag{4.36b}$$

$$\mathcal{G}_E = 2\int Dx\, H\left(-\frac{rx}{\sqrt{p - r^2}}\right) \ln H_\beta\left(\frac{\kappa - \sqrt{p}x}{\sqrt{\frac{\mu_\star^2}{\mu_1^2} + p_d - p}}\right). \tag{4.36c}$$

Now $\mathcal{G}_E$, apart for the dependence on $\sigma(\cdot)$, is identical to the energetic term of the classical teacher-student problem, while $\mathcal{G}_{SE}$ reduces to the entropic term of a spherical perceptron storing random patterns [106]. By solving the new simplified saddle

points, one can numerically compute the plateau of the generalization error in the limit, which is plotted as a function of α_T in Fig. 4.3 for different values of the margin. For every value of α, and precisely for $\alpha \to 0$, the generalization improves when sampling solutions with higher and higher margin. Indeed, the ability to generalize can be understood easily from the perspective of a robust minimizer: being high-margin solutions the ones surrounded by a flat region in the landscape (Chapter 3), this means that small changes in parameters do not significantly increase the loss, leading to models less sensitive to fluctuations. In turn, these flat regions suggest that the student has found a set of parameters that are not overly specialized to the training data; it has captured instead more general underlying patterns rather than storing specific training samples.

Bayesian case

In Fig. 4.3, the dotted lines refer to the generalization curves for $\kappa = \{0, 0.5\}$ of the Bayesian error. If in the typical case the generalization is defined as the expected error over the data distribution with the model parameters at their estimated value, the Bayesian framework treats instead the parameters as random variables with their own distribution. As a consequence, the error is indeed the expected generalization over both the data and model distributions. It is defined as the probability that the ensemble of students generalizes correctly with respect to the teacher, i.e. the probability that the average of the outputs of the students sampled from the posterior on a random new pattern has different sign than that given by the teacher

$$\epsilon_g^B = \mathbb{E}_{\boldsymbol{\xi}^\star} \overline{\Theta\left(-y^\star \langle \hat{y}^\star \rangle_{\boldsymbol{w}|\{\boldsymbol{\xi}^\mu\}}\right)}. \tag{4.37}$$

In the expression $\boldsymbol{\xi}^\star$, $y^\star$ are respectively a test pattern and its corresponding label (computed using equation (4.1)); $\hat{y}^\star$ is the output of the student given input $\boldsymbol{\xi}^\star$ as in (4.2); $\overline{\cdots}$ is the average over training patterns $\{\boldsymbol{\xi}^\mu\}$ and random features F, and finally $\langle\cdot\rangle_{\boldsymbol{w}|\{\boldsymbol{\xi}^\mu\}}$ is the average over the posterior distribution, namely the average over the probability distribution of the student weights given the training data

$$\langle \hat{y}^\star \rangle_{\boldsymbol{w}|\{\boldsymbol{\xi}^\mu\}} = \frac{1}{Z} \int \prod_i dw_i \, P_w(\boldsymbol{w}) \operatorname{sign}\left(\frac{1}{\sqrt{N}} \sum_i w_i \sigma\left(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k^\star \right) \right) \times e^{-\beta \sum_\mu \Theta\left[-\left(\frac{1}{\sqrt{D}} \sum_k \xi_k^\mu\right)\left(\frac{1}{\sqrt{N}} \sum_i w_i \sigma\left(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k^\mu\right)\right)\right]}. \tag{4.38}$$

If we use again $w_k^T = 1$ without loss of generality, we can start the computation by extracting the definitions $\hat{y}^\star$ using the integral representation of a Dirac delta function

$$\epsilon_g^B = \int \frac{dx d\hat{x}}{2\pi} e^{ix\hat{x}} \, \mathbb{E}_{\boldsymbol{\xi}^\star} \Theta\left[-\left(\frac{1}{\sqrt{D}} \sum_k \xi_k^\star \right) x \right] \overline{e^{-i\hat{x}\langle y^\star \rangle_{\boldsymbol{w}|\{\boldsymbol{\xi}^\mu\}}}}. \tag{4.39}$$

Next, by making use of the identity

$$e^{-i\hat{x}\langle y^\star \rangle_{\boldsymbol{w}|\{\boldsymbol{\xi}^\mu\}}} = \sum_{s=0}^{\infty} \frac{(-i\hat{x})^s}{s!} \langle y^\star \rangle^s_{\boldsymbol{w}|\{\boldsymbol{\xi}^\mu\}} \tag{4.40}$$

it is easy to perform the average over training patterns and random features by introducing replicas and using central limit theorem (detailed computation can be found in A.3). The final result for the expression becomes

$$\epsilon_g^B = \frac{1}{\pi} \arccos\left(\frac{M}{\sqrt{Q}}\right) \tag{4.41}$$

where $Q \equiv \mu_\star^2 q + \mu_1^2 p$ is now the projected overlap of weights belonging to different replicas. It is useful to notice that if we would like to compute the generalization error of the barycenter of typical solutions with a given margin κ, the formula above (4.37) will remain the same. In this context indeed, taking the average of the ensemble of the networks means finding a representative model that reflects the central tendency of the distribution, that is akin to computing the mean of the distribution over the network parameters. The only dependence on the margin is implicit in the order parameters q, p and r.

Comparing with numerical experiments

In order to corroborate analytical results about typical configurations with margin κ sampled from the Gibbs measure (4.19), we can exploit a spectrum of different optimization techniques, among which those that have the Gibbs distribution as the stationary probability measure: the zero-temperature Monte Carlo (MCT0) and the Simulated Annealing (SA) algorithms. If the most numerous solutions in the landscape are those with $\kappa = 0$, which, in analogy with Chapter 3, represents isolated points in a exponentially large phase space, it is well-known that local search strategies sampling from Gibbs do not work in this "golf-course". Indeed, both MCT0 and SA have troubles in finding solutions at large system size. Nevertheless, for finite-size effects, and in the highly overparameterized regime, those algorithms are able to find solutions and the statistical properties of those are in agreement with the predictions of replica theory. In Fig. 4.4 there are both train and test errors of the SA as a function of $1/\alpha$, where different colours represents different values of D while keeping α_T fixed. In order to increase D without changing the ratio $\frac{P}{D}$, the number of data points P and the student dimension N must increase proportionally, making it increasingly difficult to find a solution while we are approaching the thermodynamic limit. Indeed, the interpolation point at which zero-error solutions start to be found shifts more and more to the right in $1/\alpha$ as we increase D. Nonetheless, for a fixed system size, we can reach zero or sufficiently small training error if the model is overparameterized *enough*. The corresponding generalization error matches the replica theory result (dotted horizontal line) in the infinite overparameterized limit $\alpha \to 0$, confirming how good it is the agreement with the obtained simulations.

The same results holds for MCT0, for which in Fig. 4.5 we now show train and test error with fixed D and P but for different values of the margins as we vary $1/\alpha$. As we saw in Fig. 4.1 from a different perspective, by increasing the margin makes finding the solutions more difficult, since their entropy is decreasing accordingly. The question of whether it is also true that they appear later in the landscape as a function of $1/\alpha$ compared to the solutions at $\kappa = 0$, will be clarified in the next section. However

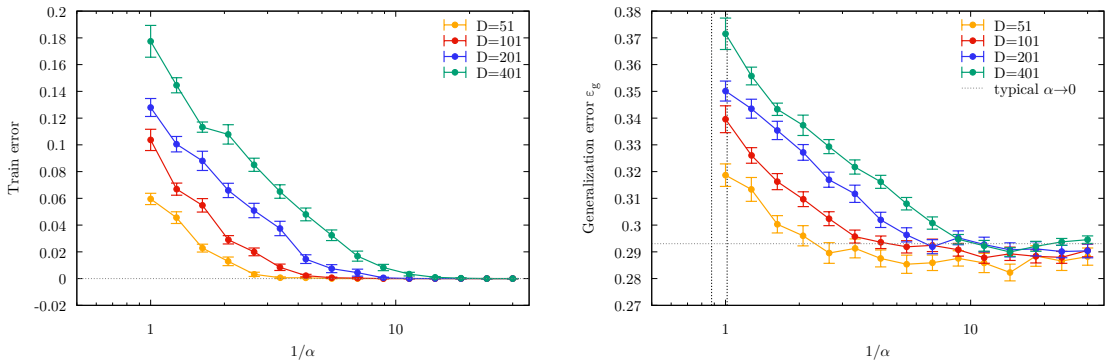


Figure 4.4: Train (left panel) and test error (right panel) of the SA algorithm as a function of $1/\alpha$. Different colors represent the result of the simulations with different values of D, while maintaining fixed $\alpha_T = 3$. The corresponding generalization error matches the replica theory result (dotted horizontal line) in the $\alpha \to 0$ limit. Reprinted from [99].

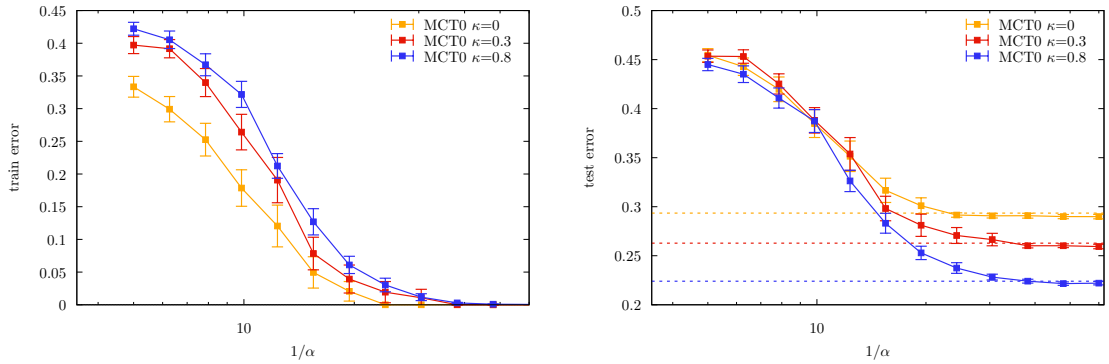


Figure 4.5: Train (left panel) and test error (right panel) of zero-temperature Monte Carlo algorithm as a function of $1/\alpha$ for different values of margins. In the simulations we fixed $D = 201$ and $P = 603$ i.e. $\alpha_T = 3$, and we ran the algorithm for a fixed number of sweeps (200). We show also in dashed the analytical predictions coming from replica theory. Reprinted from [99].

when solutions start to be accessible, increasing the margin increases the accuracy on the test set (*right* part of Fig. 4.5), with plateaus in the test that precisely correspond to the analytical predictions for $\alpha \to 0$.

The picture so far, concerning algorithms that are both not efficient in finding solutions in general, and highly performing solutions in particular, confirms that zero/low margin solutions are nearly isolated between each other, raising the question of where in landscape the highly surrounded and robust solutions are instead. In Chapter 3 we have clarified that maximum margin solutions, which represent a proxy for configurations at high local entropy, are the rarest in the landscape, being characterized by vanishing external entropy $\phi(\kappa_{\max}, \alpha) = 0$. Therefore, it is not surprising that neither of the two algorithms is able to find one of those flat/nearly flat configurations, since the entropic barrier they are subjected to is much stronger than their ability to escape *frozen* isolated configurations at the bottom of the landscape. Thus, it is

essential to shift the focus on those subdominant and flat configurations that we are now both able to describe (Chapter 3) and to sample using efficient search strategies (Chapter 2).

4.2.3 Local entropy landscape of solutions

To study the local entropy landscape around a typical solution with margin $\tilde{\kappa}$ we resort to the definition of the Franz-Parisi potential that we repeat here for clarity

$$\Phi_{FP}(t_1) = \frac{1}{Z}\int \prod_i d\tilde{w}_i \, P_w(\tilde{\boldsymbol{w}}) e^{-\tilde{\beta}\sum_\mu \ell_{NE}\left(-y^\mu \tilde{\lambda}^\mu;\tilde{\kappa}\right)} \ln \mathcal{N}(\tilde{\boldsymbol{w}}, t_1) \tag{4.42}$$

where $\mathcal{N}(\tilde{\boldsymbol{w}}, t_1)$ is the usual number of configurations $\boldsymbol{w}$ and margin κ extracted from the Gibbs measure with the number of error as energy (4.15) that have an overlap t_1 with the reference configuration $\tilde{\boldsymbol{w}}$

$$\mathcal{N}(\tilde{\boldsymbol{w}}, t_1) \equiv \int \prod_i dw_i \, P_w(\boldsymbol{w}) e^{-\beta\sum_\mu \ell_{NE}(-y^\mu\lambda^\mu;\kappa)} \delta\left(\sum_i w_i\tilde{w}_i - Nt_1\right) . \tag{4.43}$$

In (A.35), both $P_w(\tilde{\boldsymbol{w}})$ and $P_w(\boldsymbol{w})$ are the prior distributions for binary weights of the reference $\tilde{w}_i \in \{-1,+1\}^N$ and the constrained configuration $w_i \in \{-1,+1\}^N$. Making use of the techiques we have discussed already (details on the computation can be found in Appendix A.5), under the RS Ansatz the computation is involved but doable. The averaged local entropy is in the end

$$\phi_{FP} = -\frac{\hat{q}}{2}(1-q) + \frac{\alpha_D}{2}\left(p_d\hat{p}_d + p\hat{p}\right) - \alpha_D r\hat{r} + \mathcal{G}_{SS} + \alpha_D \mathcal{G}_{SE} + \alpha \mathcal{G}_E \tag{4.44}$$

with associated definitions that are reported in A.5, and where the order parameters q, $\hat{q}$, p_d, $\hat{p}_d$, p, $\hat{p}$, r, $\hat{r}$, k_1, $\hat{k}_1$, k_0, $\hat{k}_0$, $\hat{t}_1$, t_0 and $\hat{t}_0$ are found by solving the saddle point equations with ϕ_{FP} as free entropy.

As we did in Chapter 3, the bottom landscape characterization can proceed either way: by fixing a density of constraints (i.e. α) and a degree of correlation between input patterns (i.e. α_D), one can look at the averaged local entropy of solutions with increasingly large margins (Fig. 4.6, *left*); or fixing α_D and $\tilde{\kappa}$ one can study what happens at the $\tilde{\kappa}$ solutions when increasing α (Fig. 4.6, *right*). The overall picture has indeed remained the same as for the simple Perceptron. For low enough values of $\alpha < \alpha_c(\alpha_T)$ the space is full of a variety of solutions at $\tilde{\kappa}$ that are more or less surrounded by $\kappa = 0$ configurations. Zero-margin references are again isolated because of the negative entropy in a neighborhood of $d = 0$. This explains the poor performances of the SA and MCT0, both in terms of efficiency in finding a solution and in generalization, since they directly target the isolated singletons ($\tilde{\kappa} = 0$) with the highest Gibbs entropy. Even for small non-zero value of $\tilde{\kappa}$ the local entropy is negative or non-monotonic for some distances d, denoting the existence of small isolated clusters. Keeping increasing the reference margin one eventually reaches the threshold $\tilde{\kappa} = \kappa_u(\alpha)$ that separates the non-monotonic local entropy region with the

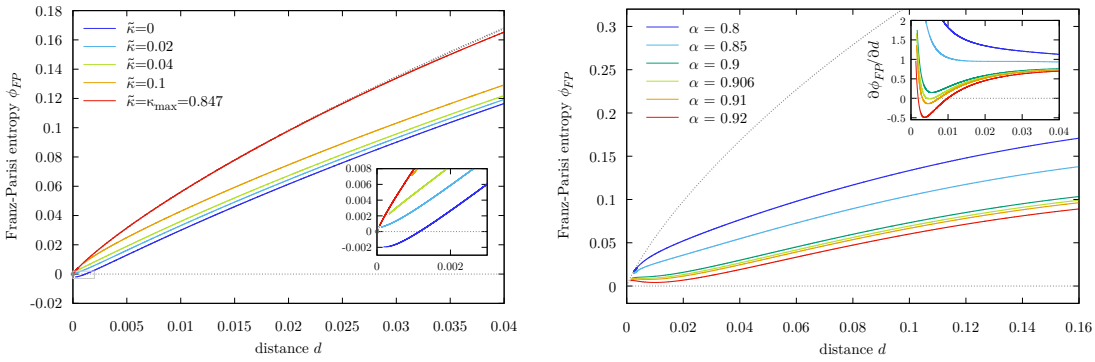


Figure 4.6: *Left*: local entropy (LE) profiles of solutions with margin $\kappa = 0$ and for some typical references having margins $\tilde{\kappa}$. Here $\alpha = 0.5$ and $\alpha_T = 8$. The maximum margin is $\kappa_{\max} \simeq 0.847$ and corresponds to the curve with the largest LE. The inset is a zoom of the entropies for small distances. *Right*: estimating the $\alpha_{\rm LE}$ transition by looking at the local entropy profiles of maximum margin references and its derivative with respect to the distance (inset). Here $\alpha_T = 2$ and the critical capacity is $\alpha_c \simeq 1.045$. Reprinted from [99].

monotonic one ($\tilde{\kappa} > \kappa_u(\alpha)$). Plus, it is still possible to define the rarest $\tilde{\kappa} = \kappa_{\max}$ solution in the landscape, which is the minimum at zero entropy $\phi_{RS}(\kappa_{\max}, \alpha, \alpha_D) = 0$ for each α and α_D.

Now, let's consider the problem from the second perspective. What degree of overparameterization is necessary (the minimum value of α) for the solutions at $\kappa_{\max}$ to appear in the landscape? Answering this question is straightforward if, for each α and α_D, one examines the local entropy of $\kappa_{\max}$ as we did in 3. When the student model is overparameterized and α is small, the solution at $\kappa_{\max}$ extends on a large scale, and the entropy is monotonic for every distance (Fig 4.6, *right* side, *blue* curve). By increasing α and reducing overparameterization, the solutions at $\kappa_{\max}$ increasingly shrinks (i.e. the cluster of solutions it belongs, decreases, and so does the local entropy), until, for $\alpha = \alpha_{LE}$, even what should be the flattest solution of the problem breaks down in multiple isolated clusters with finite size. The transition signalling that the local entropy (LE) has changed concavity happens then when

$$\kappa_u(\alpha_{LE}) = \kappa_{\max}(\alpha_{LE}) \tag{4.45}$$

which is, α_{LE} is the value of α after which not even the maximum margin solution has a monotonic local entropy profile (in the same way as we defined it for Chapter 3, reported here for clarity). In Fig. 4.6, *right* side, the derivative of the light-green curve develops a new zero for small distances at $\alpha = \alpha_{LE} = 0.906$ for $\alpha_T = 2$.

Does this perhaps mean that the problem will be UNSAT at $\alpha_T = 2$ for every $\alpha > \alpha_{LE}$? Obviously not, the crucial point of the discussion is to understand that while the critical capacity threshold $\alpha_c(\alpha_T)$ is associated with the interpolation point beyond which no solution will exist (not even those with zero margin) that can perfectly fit the dataset, the *local entropy transition* is instead linked to the idea (currently a conjecture) that when the flat solutions in the landscape break, even if there are still point-like solutions left, the problem will become computationally hard.

In reporting the phase diagram of the model in (α_T, α) plane, we can indeed clearly see what was just anticipated. Three regions can be distinguished as we increase the degree of overparameterization for every value of α_T.

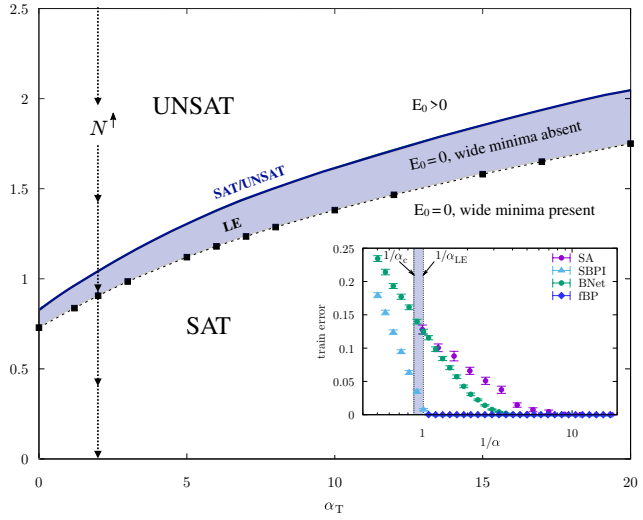


Figure 4.7: SAT/UNSAT interpolation threshold and Local Entropy transition versus α_T for the binary non-convex model of random features. For $\alpha_T \to 0$ we recover the critical capacity [22] and the local entropy transition [90] of a non-overparameterized binary perceptron trained on random patterns. In the inset we show the training error of SA, fBP and SBPI versus the degree of overparameterization $1/\alpha$ for $D = 201$ and $\alpha_T = 3$. Reprinted from [99].

- An UNSAT region when the value of the density of constraint exceeds the interpolation threshold: $\alpha > \alpha_c(\alpha_T)$. Since in this region there exists no configuration of weights that are able to fit the whole training set at zero error, the threshold is independent of the learning algorithm. It depends only on the properties of the training data and of the architecture. Indeed, when $\alpha_T \to 0$ and the problem reduces on storing uncorrelated patterns belonging to classes with uncorrelated labels from data (i.e. the storage case), the threshold coincides with the value $\alpha_c \sim 0.833$ from the binary Perceptron case. On the other hand, for $\alpha < \alpha_c(\alpha_T)$ we have a SAT region, so in principle the complexity of the student becomes sufficient to learn the dataset.

- For $\alpha_{LE}(\alpha_T) < \alpha < \alpha_c(\alpha_T)$, despite the existence of configurations of weights that fit all the training set, they are not easily accessible by algorithms because they are either isolated or belonging to minima will small characteristic size.

- For $\alpha \leq \alpha_{LE}(\alpha_T)$ highly entropic wide minima start to appear and despite being rare compared to the isolated solutions, they are accessible by simple, efficient algorithms. The threshold is thus an upper bound for the effectiveness of learning strategies.

Notice that, by moving from top to bottom of the diagram we are increasing the model size (N) and thus the overparameterization, remaining consistent the picture with the figures mentioned above (4.6). When the teacher signal is weak and α_T is low, we have to reach $\alpha < 1$ phase in order to see non-isolated solutions. Instead, the stronger the teacher signal and the larger the α_T, the more the α_{LE} regime with wide minima present increases.

Experimental checks

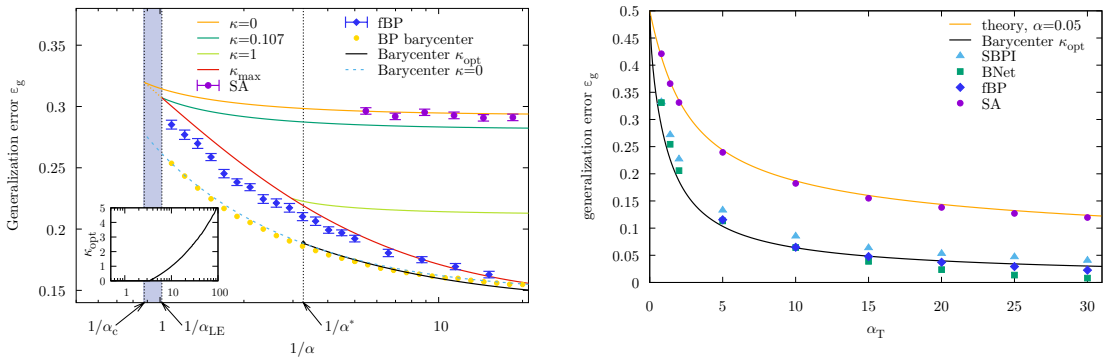


Figure 4.8: (*Left panel*) Generalization error as a function of the degree of overparameterization $1/\alpha$, for $\alpha_T = 3$. Vertical dashed lines denote the SAT-UNSAT transition α_c^{-1}, the local entropy transition $\alpha_{\rm LE}^{-1}$, and the $\kappa_{\rm opt}$ transition $(\alpha^*)^{-1}$. We show the generalization error of typical solutions with fixed margin $\kappa = 0$, 0.107, 1 and with maximum margin $\kappa_{\rm max}(\alpha, \alpha_T)$. The dashed turquoise curve is the error of the barycenter of typical solutions having zero margin, whereas the black line represents the generalization error of the "best" barycenter which was found by optimizing the margin. In the inset we show that this optimal margin $\kappa_{\rm opt}$ undergoes a transition when crossing α^*. We also show numerical results ($D = 201$) of two representative algorithms: SA (violet points) and fBP (blue points). Yellow points represent the barycenter of zero margin solutions as computed by using the BP estimation of the posterior distribution. (*Right panel*) Generalization error versus α_T in the large overparameterization regime ($\alpha = 0.05$, $D = 201$) for both theories and different optimization strategies. Reprinted from [99].

As an experimental check of the validity of the geometric conjecture as an upper bound for algorithmic hardness, we show in the inset of Fig. 4.7 the train error of four algorithms that are representative of a spectrum of sampling strategies for fixed value of α_T. On one extreme of the spectrum, as before, there is SA sampling from the equilibrium Gibbs distribution. It is indeed the worst in performance, since it is able to solve the problem only for value of overparameterization that are way far above SAT and LE thresholds. On the opposite end, we have fBP algorithm as defined in Chapter 2, which is the modified version of the message-passing BP algorithm designed to target local entropy regions. In perfect agreement with the theoretical results, which we stress to be independent of any sampling protocol, it stops finding solutions right at the threshold α_{LE}. Two more heuristic algorithms are specifically designed to work efficiently on binary architectures. One is Stochastic BP-inspired (SBPI)

algorithm [109], which can be regarded as a simple and fast approximate version of fBP, and indeed the performances are almost identical, degrading when the cluster of dense solutions breaks. The other is BinaryNet (BNet) [110], which is a modified version of SGD. Despite their individual and strategy-dependent performances, none of these algorithms can find solutions below $1/\alpha_{LE}$, confirming that the gap between information-theoretic and computational limits is a consequence of a spatial arrangement of solutions where the dominant factor is the distance (or overlap) between them (see 1.2.3).

Since the analytical protocol for targeting clustered solutions has proven efficient compared to algorithms, we can compare the associated performances. In Fig. 4.8, *left* side, the generalization error versus degree of overparameterization $1/\alpha$ always improves (i.e. for all the values of the margin) as the size of the student increases. When SA is able to find solutions, the corresponding generalization is compatible to the one obtained by typical zero-margin configurations. fBP instead behaves similarly to $\kappa_{\max}$ and tends to the generalization error of the barycenter of zero-margin configurations (i.e. the Bayesian error at $\kappa = 0$) in the large overparameterization regime. One can furthermore optimize the barycenter by looking for the value of κ_{opt} of the best performance (*inset*). Notice that, there exists a transition in $1/\alpha^\star$ above which κ_{opt} starts becoming greater than zero. Finally, as expected, since BP allows to derive marginal probabilities for each variable from the posterior distribution, it is in perfect agreement with the turquoise curve of the Bayesian error.

Test error as a function of α_T is again the same story: while SA remains attached to the typical solutions, SBPI, BNet and fBP that do not target or sample from Gibbs, perform much better, and according to the Bayesian error optimized on κ.

4.3 Comparing solutions in deep architectures

Since in the regimes of large overparametrization that we are considering, one might be tempted to think that the problem itself becomes convex and that, therefore, the space of minima concentrates in a single interpolation point when $\alpha \to \alpha_{LE}$, it is necessary to clarify in more realistic settings, which can only be tackled numerically, that this is not the case. Indeed, there are flat minima, but the problem does not become convex, as there is a non-trivial value ($\neq 1$) for the overlap of solutions independently sampled.

4.3.1 Overparameterized TCM

As a first example, let us study the case of an overparameterized TCM with K hidden units, trained on random patterns. The teacher and the patterns are generated in the same way as for Eq. (4.1), and in particular the device receives binary input $\tilde{\boldsymbol{\xi}}$ of length N, obtained by projecting the randomly-generated inputs $\boldsymbol{\xi}$ from dimension D through a random feature matrix F and a non-linearity $\sigma = \text{sign}(x)$ as in Eq. (4.3). If we divide the inputs in N/K groups, each of which is fed to one of the K hidden

units, the final output of the committee is then decided by majority voting as

$$y_{\text{out}} \equiv \text{sign}\left(\sum_{h=1}^{K} \text{sign}\left(\sqrt{\frac{K}{N}} \sum_{i=(h-1)\frac{N}{K}+1}^{h\frac{N}{K}} w_i \tilde{\xi}_i\right)\right). \tag{4.46}$$

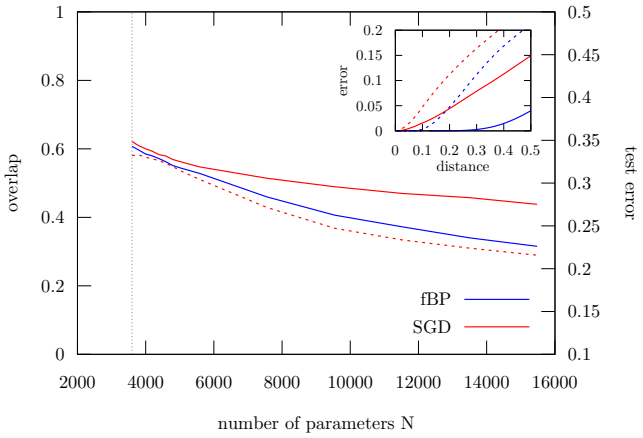


Figure 4.9: Overparameterized continuous tree-like committee machine: tests performed with $D = 2001$, $\alpha_T = 5$ and $K = 9$. Dashed red line: mean overlap between two SGD solutions on the same sample, as a function of the number of parameters N. Solid lines: test error for BP and SGD. The vertical dashed grey line at $N = 3600$ ($\alpha^{-1} \approx 0.36$) denotes the algorithmic threshold below which the algorithms solve fewer than 50% of the samples. Inset: Local energy profiles, i.e. average train error as a function of the Euclidean distance from a solution. Dashed lines are measured at $N = 3996$ ($\alpha^{-1} \approx 0.4$), solid lines at $N \approx 13500$ ($\alpha^{-1} \approx 1.35$). Reprinted from [99].

Beside the architecture, another major difference with the Perceptron case, is that we consider now $\boldsymbol{w}$ to be continuous, and since each unit is invariant to scaling because of the choice of the activation σ, we can fix the hidden unit norms to 1. For the model, we consider two learning strategies as reported in Fig. 4.9. The first one consists in an extension of fBP for continuous weights when leveraging over CLT in the N/K large limit. The second is instead SGD with $\tanh(\beta\Delta)$ as differentiable activation function. In that case, we move towards $\sigma = \text{sign}(\Delta)$ by scheduling β to increase and eventually diverge during training. In each case, even if the test error has strategy-dependent performance, for which SGD is slightly worse than fBP in this regard, it behaves, as expected, as a decreasing function of the number of network parameters N. Anyway, both algorithms fail to find a solution below $\alpha^{-1} \sim 0.36$, which we therefore take as a plausible estimate of the algorithmic threshold α_{LE}^{-1} where the phase of robust solutions presumably changes. To corroborate the idea that distinct clusters of solutions exist for $\alpha^{-1} > \alpha_{LE}^{-1}$, for which the problem remains non-convex, it is possible to study the average overlap of differently initialized solutions. At least, this can be done for SGD, since fBP is deterministic by nature. By measuring the overlap between two SGD solutions $\langle \frac{\boldsymbol{w}^a \cdot \boldsymbol{w}^b}{N} \rangle$ with cosine similarity, and averaging over multiple samples, we can see (Fig. 4.9) how it grows when reducing α^{-1} and,

accordingly, the whole volume of solutions. It does not tend to 1 as α goes to α_{LE}, dispelling any doubt that the problem has become trivial.

We can also measure the flatness of different minimizers found by the algorithms by looking for the *averaged local energy*. It is a measure of the average training loss profile around a minimum when randomly perturbing its weights and so when increasing the distance from the optimum. Given the weight configuration $w^\star$, we define $\delta E_{\text{train}}(w^\star, \eta)$ as the averaged error difference with respect to $E_{\text{train}}(w^\star)$ when perturbing $w^\star$ by a (multiplicative) noise proportional to a parameter η

$$\delta E_{\text{train}}(w^\star, \eta) = \mathbb{E}_z E_{\text{train}}(w^\star + \eta z \odot w^\star) - E_{\text{train}}(w^\star) \tag{4.47}$$

and where the symbol $\odot$ stands for the element-wise product with the normally distributed random variable $z \sim \mathcal{N}(0, I_N)$. Although local entropy and local energy metrics could potentially change without dependence on each other, it is plausible to assume that, in a landscapes with moderate irregularities, they would show a strong correlation. The first is indeed a measure of the size of the lowest valleys, whereas the second measures the average height. In the *inset* of Fig. 4.9, the local energy of two sets of value in α^{-1} are reported. As expected, close to the threshold the minima are generally sharper, growing more rapidly the error as a function of the distance from the minimum since the solution space is breaking down. Generally, for both algorithms, the profiles of the curves are flat for small distances, revealing that they are consistently biased towards flat regions.

4.3.2 Multi-layer NNs

When considering the space of configurations of standard multi-layer architectures, defining a metric requires taking into account all the *symmetries* present and grouping the classifiers into equivalence classes. In most common architectures there are two of them: one is continuous, a scale symmetry that permits the weights to be renormalized, and the other is discrete, a permutation symmetry which accounts for the interchangeability of unit labels in a hidden layer (or filter labels in convolutional layers). Specifically, the latter implies that two networks might seem substantially different in the parameter space despite being very similar, or even identical, in their functional output. This situation might arise, for instance, when the Euclidean distance between the parameters of two networks is considered, and one network is just a permutation of the units within the layers of the other. So, since the purpose of our numerical simulations here is to catch a qualitative insight of the non-convexity by making use of distance (or overlap) metrics, it is necessary to clarify the protocol to break the symmetries before comparing the networks. We will do that according to [40].

Breaking the symmetries

- Networks using ReLU activation functions and continuous parameters possess a scale invariance characteristic: the weights of the neurons in each hidden layer can be scaled by any positive real number (which can vary across neurons)

without affecting the network's output. This is due to the ReLU functional property since $\mathrm{ReLU}(a\Delta) \equiv a\mathrm{ReLU}(\Delta)$ for any positive scalar a. To counteract this symmetry, we adopt a normalization strategy that begins with the first layer and advances sequentially through the subsequent layers. For each hidden layer, starting from the first and continuing up to the penultimate layer $L-1$, we compute the norm $|w_k^\ell|$ of each hidden unit κ within that layer ℓ. Then, we scale the incoming weights, bias, and batch-norm parameters of the unit by $|w_k^\ell|^{-1}$ and its outgoing weights by $|w_k^\ell|$. This normalization ensures that every unit in the layer has a norm of 1, yet the network's overall function (and its loss) is preserved. The final layer lacks this symmetry; however, it can undergo a global rescaling by any positive factor, as the network output is unaffected due to the argmax operation used. In this way, by normalization, we can avoid issues concerning infinitely many parameter representations.

- Instead, to break permutational symmetry, for each layer ℓ ranging from 1 to $L-1$, we employ an algorithm that identifies the optimal permutation π of indices for the second network, chosen to maximize the cosine similarity between the units of the two networks. This permutation is applied to the corresponding incoming and outgoing weight indices of the second network layer ℓ, as well as to other related parameters such as biases and batch-norm parameters. To do that, any algorithm that can resolve the problem of minimum weight matching in bipartite graphs is sufficient. This method does not ensure the global minimization of the distance between two networks in the worst-case scenario, and the feasibility of such an objective from a computational standpoint is uncertain. Additionally, it may not account for biases when they are part of the network structure. Nonetheless, this approach assures that if $\boldsymbol{w}^a$ and $\boldsymbol{w}^b$ represent the same network but with shuffled labels of hidden units, they will be matched almost surely by the end of the process.

We refer to [40] for all the details on the implementations.

Results

We examined a straightforward, fully-connected, multi-layer Perceptron architecture with a fixed number of hidden layers but with varying width in order to increase the model complexity (N). The model task is to classify the MNIST dataset according the parity of the depicted digits, requiring so a binary classification. Using full-batch GD complemented by ADAM, we get different generalization plateaus and algorithmic thresholds according to the chosen initializations (Fig 4.10). Precisely, since the adversarial initialization introduces some label noise in the dataset, the cusp of the test error at the interpolation point is more pronounced that the one for the standard orthogonal initialization. When the model begins to align with the training data, there are multiple possible solutions, leading to a mean overlap among separate instances that is less than 1 once taken into account the symmetries. The inset indicates that these solutions maintain their robustness against noise perturbation near the algorithmic threshold. Indeed, the local energy is flat in that regime.

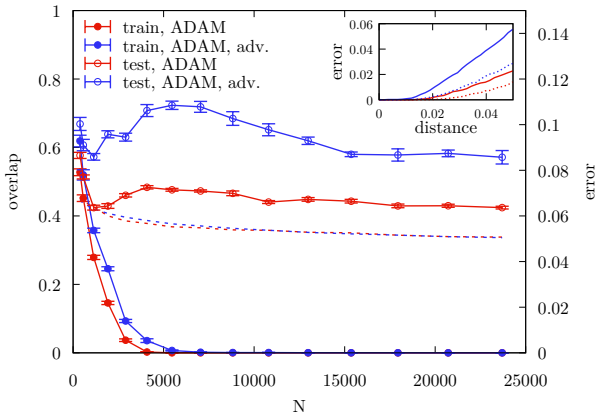


Figure 4.10: Train (full points) and test (empty points) error as a function of the network parameters N for a fully-connected network with 5 hidden layers. The network is trained using full batch gradient descent with ADAM optimization, with random orthogonal initialization (red curves) and adversarial initialization (blue curves). Mean overlap between solutions is far below 1 (dashed lines). Inset: local energy profiles for both algorithms at two different values of the overparameterization ($N \simeq 9 \cdot 10^3$ (full lines) and $N \simeq 2 \cdot 10^4$ (dashed lines)). Reprinted from [99].

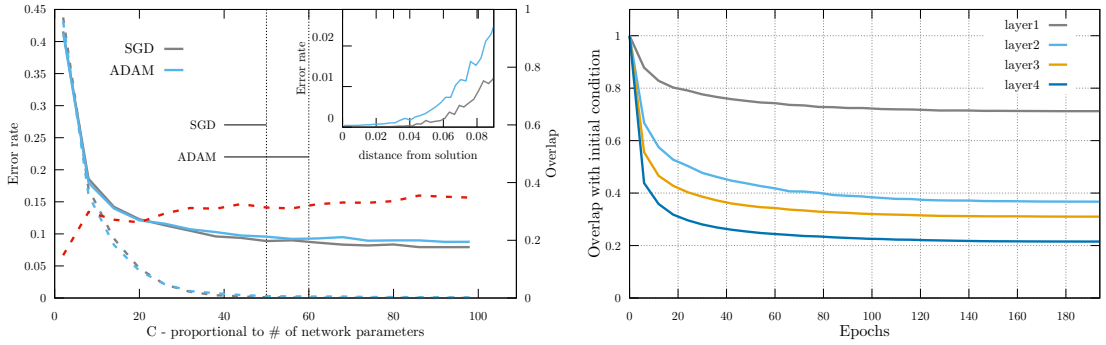


Figure 4.11: (*Left*) Train (dashed) and test (solid) errors of solutions obtained with SGD and ADAM (gray and blue lines respectively) as a function of C. Inset: local energy as a function of the distance from solutions (gray: SGD with $C = 54$; blue: ADAM with $C = 64$). (*Right*) Overlap of a solution with its initial condition as a function of the epochs. The overlap is reported layer by layer and it refers to ADAM optimizer, a number of filters $C = 20$, and a learning rate $lr = 0.01$. Reprinted from [99].

In our second analysis of deep network structures, we examine a five-layer neural network (NN) comprising of four convolutional layers followed by one fully connected layer [111]. The level of overparameterization in this model is controlled through a parameter C where each layer ℓ contains $2\ell \cdot C$ convolutional filters. We trained this convolutional neural network (CNN) on the CIFAR10 dataset for 200 epochs using two distinct learning algorithms: ADAM with momentum and SGD (details on the hyperparameters can be found in [99]). With real datasets like CIFAR10, achieving near-zero training error is feasible, but attaining exactly 0 error requires significantly

more computational effort. Therefore, we define solutions as configurations that misclassify no more than one pattern, with less than 0.0017% training error, and base our estimate of the algorithmic (LE) threshold on this criterion. Fig. 4.11 displays the training (dashed line) and test (solid line) errors for both ADAM and SGD optimizers. For ADAM and SGD, the model starts fitting the training data at $C = 60$ and $C = 50$, respectively, while the generalization error steadily decreases as the number of network parameters increases. Notably, these architectures operate in *lazy* training regimes, since the first layer changes little during training, while subsequent layers alter more significantly. However, the distance from the initial conditions to the trained configurations does not fall to zero as the number of epochs grow up (Fig. 4.11, *right*). Our findings, consistent with analytical results and parallel studies of other networks, reveal that, unlike the interpolation threshold in convex networks, overlaps do not always reach the maximum value of one, as shown by the red curve in Fig. 4.11. The solutions found by both SGD and ADAM display similar geometric characteristics, being the first derivative of the perturbed configurations at short distances from the minima pratically zero (*inset*).

Chapter 5

Linear mode connectivity in continuous non-convex loss landscape

Although a consistent picture has emerged from the previous chapters, namely that the more robust the solutions are, the closer they are in the phase space, a precise investigation into their connectivity must be carried forward in light of the empirical results on the *linear mode connectivity* (LMC) of solutions found by different optimizers that have a bias towards flat degenerate regions. Briefly speaking, LMC [40, 112–114] refers to the phenomenon where pairs of configurations can be connected by straight linear paths without a significant increase in the loss value once taken into account scaling and permutational symmetries. For simple architectures and tasks, studies on the value of the loss along the linear interpolation path between the initialization and a solution suggest a smooth loss landscape [115]. This is because there seems to exist a subspace where training can proceed by descending a single smooth slope without any barriers. However, the picture becomes more complicated when trying to characterize such a high-dimensional landscape with a one-dimensional estimate for complex models [116]. Indeed, even if there were a straight linear path between initialization and the optimum (which is not the case in deeper models), this would not provide any information on the actual dynamics of algorithms navigating a high-dimensional landscape. Anyway, results on LMC are not so much about the linear path from initialization but about the connectivity between different minimizers in the loss landscape. This certainly aligns with the direction taken so far: if there exist convex basins in the landscape where minima can be connected, then it is true that, even though in general the landscape may be rough and non-convex, there exist regions of simpler geometry (i.e., flat) where algorithms end up. Plus, the existence of LMC supports the Lottery Ticket Hypothesis (LTH). In a dense neural network that is randomly initialized, smaller subnetworks, known as *winning tickets* can potentially match or even exceed the original network performance in a similar timeframe. These winning tickets are likely to be situated within the subspaces that are linearly connected. Indeed, pruning methods employed in LTH can be viewed as strategies to isolate these linearly connected subspaces. The relationship highlights a potential mechanism by which LTH operates, underlining the significance of linear connectivity in neural network optimization.

In general, LMC is yet another clue to questioning about traditional beliefs in

high-dimensional non-convex optimization (see Section1.2.2), raising several questions in ensembling methods [117, 118] or merging models [114]. Can we leverage LMC to merge different models effectively, creating a hybrid model that combines the strengths of each? [114] Or, for ensemble methods which exploit the diversity among models to enhance generalization and robustness by requiring independent training processes, can one facilitate the construction of ensembles from neighboring high-quality solutions along the low-loss paths without necessitating separate training phases? Just to give a hint about the type of questions.

In the present Chapter, we will provide the first statistical mechanics study on the behavior of the training error on the *geodesic path* connecting two solutions as in [119] for the negative spherical Perceptron problem [106, 120, 121]. We will see how, together with the analytical characterization on the robustness of minimizers (Chapters 3, 4), LMC reveals about the complexity and geometry of the loss landscape, producing a complete characterization of the *shape* of the solution space.

5.1 The negative Perceptron problem

We denote by $\boldsymbol{W}$ the N-dimensional vector of weights of the Perceptron, that is normalized on the N-dimensional sphere of radius $\sqrt{N}$ such that $||\boldsymbol{W}||^2 = N$. Considering Rademacher inputs $\boldsymbol{\xi}$ and outputs σ where each component is i.i.d., we construct the random dataset $\mathcal{D}$ of $P = \alpha N$ pairs: $\{\boldsymbol{\xi}^\mu, \sigma^\mu\}_{\mu=1}^P$. The purpose is to characterize the space of solutions with a certain margin κ_E, that is the hard threshold for the zero energy condition. If that condition for satisfiability of all the constraints at large N and constant α reads

$$\sigma^\mu \frac{\boldsymbol{W} \cdot \boldsymbol{\xi}^\mu}{\sqrt{N}} - \kappa_E > 0 \tag{5.1}$$

and $\kappa_E < 0$, then since the linear separability of the dataset is no longer a necessary condition, the problem is non-convex. Being the simplest non-convex model of NNs, the negative Perceptron has recently gained attention both in physics [121–123] and in mathematics communities [124, 125]. For the first, different structural phase transitions have been characterized through the lens of replica theory. In the phase diagram (κ, α) where the density of constraints remains $\mathcal{O}(1)$ in the limit $N, P \to \infty$, by varying κ one encounters different non-ergodic Replica-Symmetry-Breaking (RSB) phases. At zero temperature, if one looks at Fig 5.1 and for any fixed value of κ_E, by increasing the density of constraints there exists as usual the critical value $\alpha_c(\kappa_E)$ that separates the SAT phase with an exponential number of solutions, from the UNSAT phase where it is no longer possible to satisfy all the constraints. Computed within the RS ansatz in *light blue* and refined using 1RSB approximation in *orange*, it is easy to notice how, as long as $\kappa_E > 0$ and the problem is convex, the two predictions coincide, being the landscape that of one ergodic connected component with the free energy that has only one minimum. Instead, as soon as κ_E becomes negative, the RS stops being stable towards RSB effects. If one studies the Hessian matrix

$$H_{ab;cd} = \frac{\partial^2 S_{RS}}{\partial q_{ab}\, \partial q_{cd}} \tag{5.2}$$

where S_{RS} is the ground-state entropy prediction within RS ansatz, the RS saddle point will be stable as long as every eigenvalue of the matrix is non-negative. When Replica-Symmetry breaks and vanishing eigenvalues appear, the function $q(x)$ describing the typical overlap between two independently sampled solutions will no longer be constant and the space of solutions becomes that of disconnected non-ergodic clusters. The *de Almeida-Thouless* (dAT) transition line ($\alpha_{\mathrm{dAT}}(\kappa_E)$) in Fig. 5.1 identifies the instability of the RS solution. Crossing $\alpha_{\mathrm{dAT}}(\kappa_E)$ for $\kappa_E > \kappa_{\mathrm{1RSB}}$, the dAT instability leads to a fullRSB phase, while for $\kappa_{\mathrm{RFOT}} < \kappa_E < \kappa_{\mathrm{1RSB}}$ the dAT line marks the point where the system transitions from being RS to a stable 1RSB phase. In the scenario where $\kappa_E < \kappa_{\mathrm{RFOT}}$, a sequence of three distinct transitions occurs as α increases. Observing the diagram from lower to the upper part, the system first enters a dynamical 1RSB phase when α is between α_{dyn} and α_{K}. In this phase, $q_1 > q_0$ and the Parisi block parameter is set to $m = 1$. Here, the subscripts 'dyn' and 'K' denote the *dynamical* and *Kauzmann* phase transitions, which are fundamental in the mean field theory of glasses. As α increases from α_{K} to α_{G}, the complexity of the clusters that dominate the Gibbs measure vanishes. The system then only retains 1RSB solutions with $m < 1$. Upon further increasing α, the 1RSB solution becomes unstable, leading to a *Gardner* transition that brings the system into a fullRSB phase. This series of transitions has been similarly pointed out in high-dimensional systems of hard spheres, being at the basis of the "Random First Order Theory" (RFOT) [75, 126, 127]. However, since it is not the purpose of this Chapter to delve into the details concerning the phase diagram, we refer to a more detailed discussion to the works [121, 123]. In what follows, we restrict to the properties of the ground state in the region below $\alpha_{\mathrm{dAT}}(\kappa_E)$ where the RS holds.

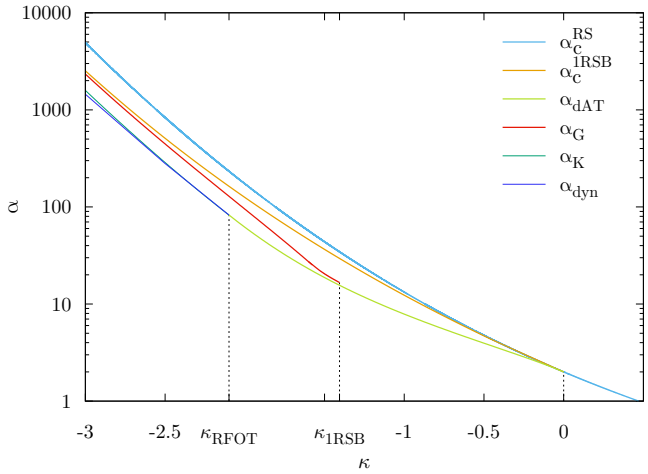


Figure 5.1: The zero temperature phase diagram of the continuous spherical perceptron showing the density of constraints α as a function of the margin κ_E by which the problem is solved. For each $\kappa_E \leq 0$ the problem is non-convex and RSB transition effects appear. Reprinted from [123].

5.2 Typical overlap between pairs of solutions

In the RS phase, the problem is SAT w.h.p. in the thermodynamic limit and the uniform probability density over the solutions reads

$$p_{\boldsymbol{\xi},\kappa_E}(\boldsymbol{W}) = \frac{1}{Z_{\boldsymbol{\xi},\kappa_E}}\delta(\|\boldsymbol{W}\|^2 - N)\prod_{\mu=1}^{P}\Theta(\boldsymbol{W}\cdot\boldsymbol{\xi}^\mu - \kappa_E\sqrt{N}) \tag{5.3}$$

having the partition function $Z_{\boldsymbol{\xi},\kappa_E}$ the role of a normalization factor. For entropic reason, if the problem is satisfied at a given α, it will be characterized by an exponential number solutions that we call *typical*, i.e. having a stability distribution with a hard constraint precisely at $\Delta^\mu \equiv \boldsymbol{W}\cdot\boldsymbol{\xi}^\mu = \kappa_E\sqrt{N}$ as described by the measure (5.3) and Fig. 5.2 (*left*). However, as we saw in chapters 3 and 4, this does not rule out the existence *atypical* solutions becoming statistically irrevelant for the measure (5.3) in the large N regime, because subdominant in number. Those satisfy higher margin constraints that the one of the problem (Fig. 5.2, *right*), or have in general a different stability profile with average greater than the typical. If the purpose is that of characterizing the geometry of the solution space, we cannot neglect in general solutions with $k > \kappa_E$, i.e. the ones highly surrounded and well-behaving in chapters 3 and 4.

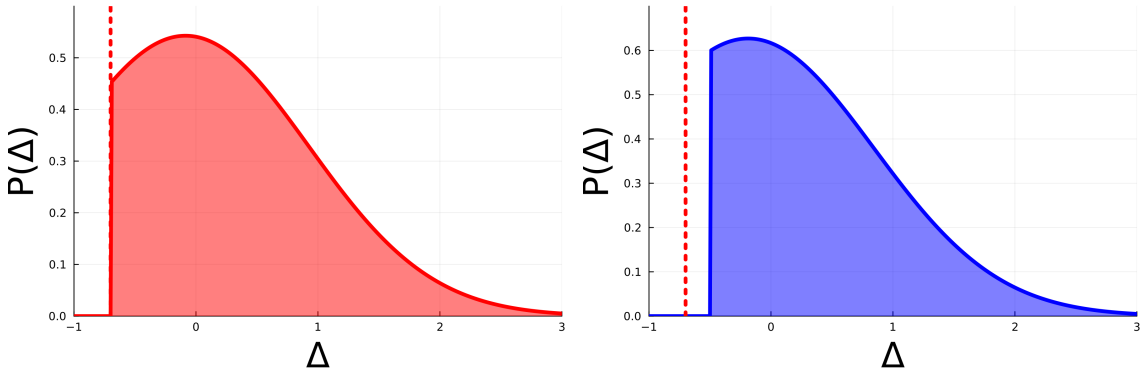


Figure 5.2: Stability distributions related to a *typical* (left) and to an *atypical* (right) solutions at $\kappa_E = -0.7$. Notice how the threshold for the satisfiability coincide with the value Δ up to which the distribution is zero in the typical case, while $\Delta_{eff} = -0.5 > \Delta$ for robust solutions.

Let us start by considering the free entropy landscape of solutions pairs independently sampled from the measure (5.3). If we label them as $\boldsymbol{W}^1$ and $\boldsymbol{W}^2$ with margins k_1 and k_2 both greater than κ_E, we can compute in general

$$\phi(k_1,k_2) = \frac{1}{N}\mathbb{E}_{\boldsymbol{\xi}}\log Z_{\boldsymbol{\xi}}(k_1,k_2) \tag{5.4}$$

where the average is over the patterns $\boldsymbol{\xi}$ because as usual we gauge-transform $\xi_i^\mu \to \sigma^\mu\,\xi_i^\mu$ to get rid of the label. Because in principle there is no distance constraint between the configurations, the partition function will factorize $Z_{\boldsymbol{\xi}}(k_1,k_2) = Z_{\boldsymbol{\xi}}(k_1)Z_{\boldsymbol{\xi}}(k_2)$.

Using the replica trick, the computation of

$$\langle Z^n \rangle = \mathbb{E}_{\boldsymbol{\xi}} \int \prod_{ar} d\mu(\boldsymbol{W}^{ar}) \prod_{\mu ar} \Theta\left(\frac{1}{\sqrt{N}} \sum_i W_i^{ar} \xi_i^\mu - k_r \right) \tag{5.5}$$

proceeds straightforward once the index $r = 1, 2$ running over the two solutions with margins k_r and the replica index $a = 1, \dots, n$ have been introduced. As expected, within the RS ansatz (details on the computation are reported in B.1), the overlap matrix can be divided in two contributions

$$q_{rr}^{ab} = \delta_{ab} + q_r(1 - \delta_{ab}) \quad \forall r = s, \tag{5.6a}$$

$$q_{rs}^{ab} = R\,\delta_{ab} + p\,(1 - \delta_{ab}) \quad \forall r \neq s. \tag{5.6b}$$

where it is necessary to differentiate the typical overlap q_r of each configuration from the overlap p between them, since in principle $k_1 \neq k_2$. Notice that, in our notation, p and R are respectively the overlaps between the two reference solutions within the same replicas or not. There is no reason to assume p and R to be different since the system is decoupled. However, since we are interested in the value of p, which is the observable that allows to characterize the landscape of pairwise solutions and that would not appear in the calculation if $p = R$, we use the trick of assuming them different and later perform the limit $R \to p$. The calculation follows standard steps and in the end the free entropy can be written as

$$\phi = \frac{1}{2} \sum_r q_r \hat{q}_r + \sum_r \hat{h}_r + p\,\hat{p} - R\,\hat{R} + \mathcal{G}_S + \alpha \mathcal{G}_E \tag{5.7}$$

and with

$$\mathcal{G}_S = \frac{1}{2} \ln \frac{4\pi^2}{(2\hat{h}_1 + \hat{q}_1)(2\hat{h}_2 + \hat{q}_2) - (\hat{p} - \hat{R})^2} + \frac{\hat{h}_2\,\hat{q}_1 + (\hat{h}_1 + \hat{q}_1)\,\hat{q}_2 + \hat{p}\,(\hat{R} - \hat{p})}{(2\hat{h}_1 + \hat{q}_1)(2\hat{h}_2 + \hat{q}_2) - (\hat{p} - \hat{R})^2} \tag{5.8a}$$

$$\begin{aligned} \mathcal{G}_E = \int \mathcal{D}z_1 \mathcal{D}z_2 \ln \int \mathcal{D}x\, H\left(\frac{k_1 + \sqrt{R - p}\,x + \sqrt{q_1}\,z_1}{\sqrt{1 - q_1 - R + p}} \right) \\ \times H\left(\frac{k_2 + \sqrt{R - p}\,x + \frac{p}{\sqrt{q_1}}\,z_1 + \sqrt{q_2 - \frac{p^2}{q_1}}\,z_2}{\sqrt{1 - q_2 - R + p}} \right) \end{aligned} \tag{5.8b}$$

where $\mathcal{D}x \equiv dx \frac{e^{-x^2/2}}{\sqrt{2\pi}}$ and $H(x) = \frac{1}{2}\mathrm{erfc}\left(\frac{x}{\sqrt{2}}\right)$ as usual. Again, by looking at the expression (5.7), it is easy to check that in the limit $R \to p$ the formula reduces the sum of the two entropies of single configuration. To actually compute the value of p we can resort to the saddle point equation $\hat{R} = \alpha \frac{\partial G_E}{\partial R}$ in the limit $R \to p$ (see Appendix B.1.2), which eventually returns the implicit equation

$$p = \alpha \sqrt{(1 - q_1)(1 - q_2)} \int \mathcal{D}z_1 \mathcal{D}z_2\, GH\left(\frac{k_1 + \sqrt{q_1}\,z_1}{\sqrt{1 - q_1}} \right) GH\left(\frac{k_2 + \frac{p}{\sqrt{q_1}}\,z_1 + \sqrt{q_2 - \frac{p^2}{q_1}}\,z_2}{\sqrt{1 - q_2}} \right). \tag{5.9}$$

It can be solved numerically by Newton methods once the saddle points of single configuration have been fixed since, as mentioned, the free entropy value is not affected by p.

5.2.1 Nested overlap structure

To understand the arrangement of solutions at different margins and specific α in the phase space, we can study how the overlap parameters q_1, q_2, p and their relation evolve as a functions of k_1 and k_2. As detailed for the phase diagram, since for the RS approximation the space of solution is a unique connected component, we can trust the results until $\kappa_{\mathrm{dAT}}(\alpha)$. Let us suppose to be in the situation for which $k_1 < k_2$ (Fig. 5.3). In the plane of the overlap as a function of k_2, then q_1 will be a constant, since it does not depend on the typical overlap of configuration $\boldsymbol{W}^2$. Instead, $q_2(k_2)$ will be an increasing function of k_2, since as we saw from Chapter 3, increasing the margin leads to sample solutions closer and closer together even if rarer. What is really interesting, however, is that for each value of $k_1 < k_2$, the overlap between the two solution is such that $q_1 < p < q_2$. This suggests that solutions with a given margin are arranged in an hierarchical fashion: the most robust solutions are gradually surrounded by solutions with smaller and smaller margin (a similar conclusion was highlighted in Chapter 3). Furthermore, as shown in Fig. 5.3 for fixed k_1 and α, $p\,(k_2; \alpha, k_1)$ behaves as a non-monotonic function of k_2. It initially increases up to a certain $k_2^\star(\alpha)$ and then it starts to decrease, showing that the space of these rearranged solutions is not isotropic with respect to high-margin solutions (it will be discussed better in Section 5.4.2).

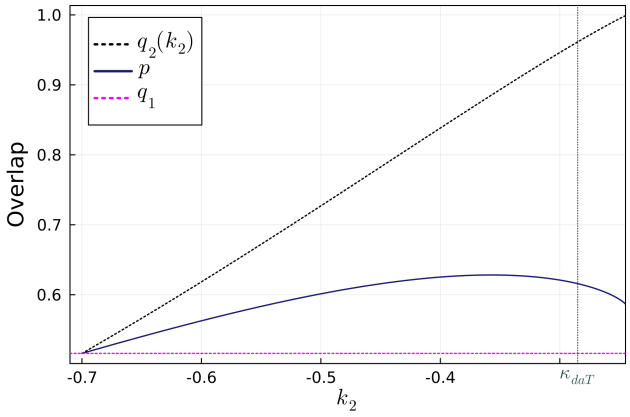


Figure 5.3: As a function of k_2 from bottom to above: typical overlap q_1 of solutions having margin $k_1 = -0.7$ (dashed pink line); typical overlap p between two solutions having margin k_1 and k_2 (full blue line); typical overlap q_2 of solutions with margin k_2 (dashed black line). Here $\alpha = 3$. The dotted vertical line indicates the threshold κ_{dAT} for the RS stability. Reprinted from [119].

5.3 Manifold spanned by y solutions

Having an estimate, albeit qualitative for now, of how solutions with different margins are spatially organized, we can move forward and characterize the asymptotic training error along the manifold spanned by y independently sampled solutions. Practically, if we have y solutions $\{\boldsymbol{W}^r\}_{r=1}^y$ with margins $\{k_r\}_{r=1}^y$ in general different, we can consider the projection onto the N-dimensional sphere of the $y-1-$simplex

$$\boldsymbol{W}_{\boldsymbol{\gamma}} = \frac{\sqrt{N}\sum_{r=1}^y \gamma_r \boldsymbol{W}^r}{\|\sum_{r=1}^y \gamma_r \boldsymbol{W}^r\|}, \tag{5.10}$$

which gives rise to a convex envelope since the vector of interpolation coefficients $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_y)$ is such that $\gamma_r \geq 0$ and $\sum_{r=1}^y \gamma_r = 1$. In the simple case where $y = 2$, interpolating between two normalized vectors produces a straight path between them. When projecting on the sphere the linear path, it becomes a geodesic path for $y = 2$ and geodesic polytopes of $y-1$ dimension for $y > 2$ (their edges are themselves geodesic polytopes of dimension $y-2$, Fig. 5.4).

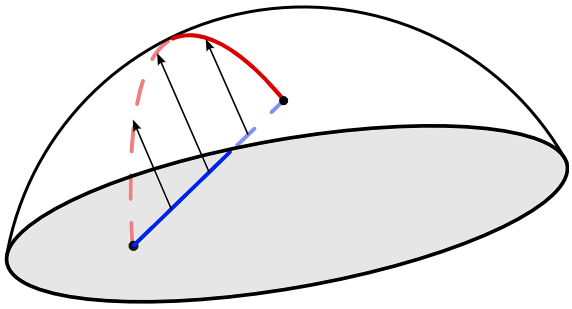


Figure 5.4: Sketch of the projection of a straight path (*blue*) onto the geodesic (*red*) in the case of a one-dimensional path, i.e. $y = 2$. Reprinted from [119].

By changing the margins $\{k_r\}_{r=1}^y$, and so changing the edges of the polytope, one can focus on different regions of the solution manifold. Setting κ_E the margin of the problem, studying the expected training error on the manifold means to evaluate

$$E_{\boldsymbol{\gamma}} = \lim_{N\to+\infty} \alpha\, \mathbb{E}_{\boldsymbol{\xi}} \left\langle \Theta\big(- \boldsymbol{W}_{\boldsymbol{\gamma}} \cdot \boldsymbol{\xi}^\mu + \kappa_E \sqrt{N}\big) \right\rangle_{k_1,\dots,k_y} \tag{5.11}$$

where the average $\langle \bullet \rangle_{k_1,\dots,k_y}$ is taken over the product measure of y Gibbs ensemble, each of the form (5.3), that is factorized

$$\langle \bullet \rangle_{k_1,\dots,k_y} = \frac{\int \prod_{r=1}^y d\mu(\boldsymbol{W}_r) \prod_{\mu r} \Theta\left(\frac{1}{\sqrt{N}} \sum_i W_{ri} \xi_i^\mu - k_r\right) \bullet}{\prod_r^y Z(k_r)}. \tag{5.12}$$

Our general case will be the one of sampling one solution with margin k_1 and $y-1$ solutions with margins k_2, both in principle such that: $k_1 \neq k_2 \neq \kappa_E$. With the replica trick the computation is standard before the ansatz, and

$$E_{\boldsymbol{\gamma}} = \lim_{N\to\infty} \lim_{n\to 0} \int \prod_{a<b,rs} \frac{dq_{rs}^{ab}\, d\hat{q}_{rs}^{ab}}{2\pi} \int \prod_{ar} \frac{d\hat{h}_r^a}{2\pi} e^{Nn\phi(q,\hat{q},\hat{h})} g(q) \tag{5.13}$$

where q_{rs}^{ab} is now a tensor and

$$g(q) = \frac{\int \prod_{ar} \frac{d\lambda_r^a d\hat{\lambda}_r^a}{2\pi} \prod_{ar} \Theta(\lambda_r^a - k_r)\, \Theta\left(-\frac{\sum_{r=1}^{y} \gamma_r \lambda_r^1}{\sqrt{\sum_{rs} \gamma_r \gamma_s q_{rs}^{a=b}}} + \kappa_E\right) e^{i \sum_{ar} \hat{\lambda}_r^a \lambda_r^a - \frac{1}{2} \sum_{abrs} q_{rs}^{ab} \hat{\lambda}_r^a \hat{\lambda}_s^b}}{\int \prod_{ar} \frac{d\lambda_r^a d\hat{\lambda}_r^a}{2\pi} \prod_{ar} \Theta(\lambda_r^a - k_r)\; e^{i \sum_{ar} \hat{\lambda}_r^a \lambda_r^a - \frac{1}{2} \sum_{abrs} q_{rs}^{ab} \hat{\lambda}_r^a \hat{\lambda}_s^b}}. \tag{5.14}$$

In the large N limit, the saddle points $\{q^\star, \hat{q}^\star, \hat{h}^\star\}$ for ϕ are not affected by $g(q)$ and so the small n limit will lead to

$$E_{\boldsymbol{\gamma}} = \lim_{n \to 0}\; g(q_\star). \tag{5.15}$$

RS - Ansatz

Since the tensor q_{rs}^{ab} is identified as before by two matrices contributions $q_{rs}^{a=b}$ and $q_{rs}^{a \neq b}$, then the RS structure reads

$$q_{rs}^{aa} = \delta_{rs} + (1 - \delta_{rs})\, p_{rs} \tag{5.16a}$$

$$q_{rs}^{a \neq b} \equiv t_{rs} = q_r \delta_{rs} + (1 - \delta_{rs})\, p_{rs} \tag{5.16b}$$

with the same notation as in (B.8), where q_r for $r \in [y]$ is the typical overlap of each solution at margin k_r, while p_{rs} is instead the overlap between solutions with margins k_r and k_s. Using Hubbard-Stranotovich transformations and by integrating when possible (details on the computation are reported in B.2.1), the final expression turns out to be

$$E_{\boldsymbol{\gamma}} = \Theta\left(\kappa_E c_{\boldsymbol{\gamma}} - \sum_r \gamma_r k_r\right) \int \prod_r \mathcal{D}z_r \frac{\int_{A_1}^{B^1} \mathcal{D}\lambda_1 \ldots \int_{A_{y-1}}^{B^{y-1}} \mathcal{D}\lambda_{y-1} \left(H(A_y) - H(B_y)\right)}{\prod_r H(A_r)} \tag{5.17}$$

and if

$$f_{\boldsymbol{\gamma}}(\kappa_E, \{k_r\}_r) \equiv \kappa_E c_{\boldsymbol{\gamma}} - \sum_r \gamma_r k_r \tag{5.18a}$$

$$I_{\boldsymbol{\gamma}}\left(\kappa_E, \{k_r\}_r\right) \equiv \int \prod_r \mathcal{D}z_r \frac{\int_{A_1}^{B^1} \mathcal{D}\lambda_1 \ldots \int_{A_{y-1}}^{B^{y-1}} \mathcal{D}\lambda_{y-1} \left(H(A_y) - H(B_y)\right)}{\prod_r H(A_r)} \tag{5.18b}$$

the formula reduces to the factorized product of two meaningful terms

$$E_{\boldsymbol{\gamma}} = \Theta\left(f_{\boldsymbol{\gamma}}(\kappa_E, \{k_r\}_r)\right) I_{\boldsymbol{\gamma}}\left(\kappa_E, \{k_r\}_r\right). \tag{5.19}$$

The first depends on a Heaviside function which is linear in the margins and where $c_{\boldsymbol{\gamma}} = \sqrt{\sum_{rs} q_{rs} \gamma_r \gamma_s}$ is somewhat proportional to the norm. The second is instead a non-negative function involving the product of $2y$ Gaussian integrals. So, since the error evals zero only if the argument $f_{\boldsymbol{\gamma}}(\kappa_E, \{k_r\}_r) < 0$, we end up with a simple condition for the geodesics to entirely lie on the zero error manifold.

5.3.1 Sampling with the same margin $k \geq \kappa_E$

For the sake of clarity, it is useful to divide the characterization in cases. We first consider the case where the vertices of the simplex are all sampled with the same margin $k_r = k \geq \kappa_E$. If it is the case, then $f_{\boldsymbol{\gamma}} = \kappa_E c_{\boldsymbol{\gamma}} - k$ since all solutions are statistically independent and the only order parameter is $q = \frac{1}{N}\mathbb{E}\langle \boldsymbol{W}^1 \cdot \boldsymbol{W}^2 \rangle_{k,k}$ the overlap between them, which coincides with the typical overlap of each within pairs of virtual replicas since $k_r = k$. As a consequence, the norm

$$c_{\boldsymbol{\gamma}}^2 = (1-q)\sum_r \gamma_r^2 + q \ \leq 1 \tag{5.20}$$

and the strict inequality holds at the vertices, while the norm has its minimum at the barycenter $\gamma_r = \frac{1}{y}$, where it is

$$c_{\text{barycenter}}^2 \quad = \quad (1-q)\,\frac{1}{y} + q. \tag{5.21}$$

Case $k = \kappa_E$

When $k = \kappa_E$, given that $c_{\boldsymbol{\gamma}} \leq 1$, the argument of the theta function in equation (5.19) which is $f_{\boldsymbol{\gamma}} = \kappa_E c_{\boldsymbol{\gamma}} - k$, is invariably positive for $\kappa_E < 0$. As a result, every point on the simplex retains some non-zero energy. For $\kappa_E > 0$, instead, the argument is consistently negative, meaning that at the threshold value $\kappa_E = 0$, $E_\gamma = 0\ \forall \gamma$ along the interpolation path between two solutions. Indeed, the problem is convex as soon as $\kappa_E \geq 0$ and so it is not surprising that typical solutions are linearly connected in that regime.

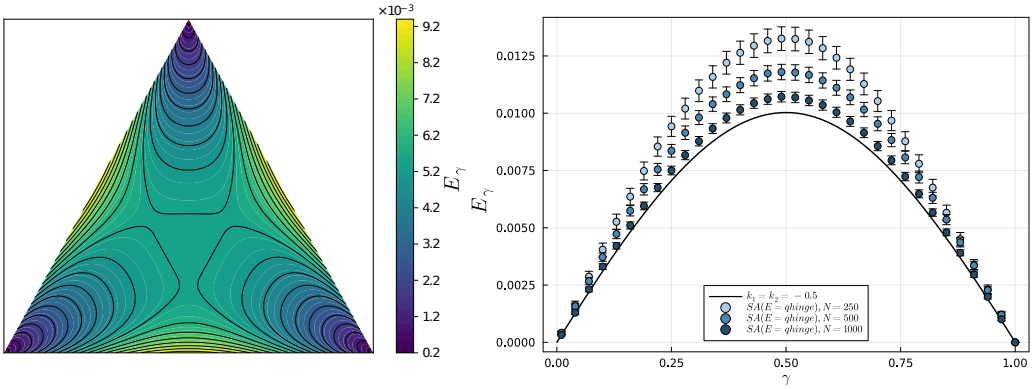


Figure 5.5: (Left panel) Training error manifold spanned by $y = 3$ typical solutions where $k = \kappa_E = -0.5$ and $\alpha = 1$. (Right panel) Typical energy barrier between $y = 2$ solutions with $k = \kappa_E = -0.5$ and $\alpha = 1$ (corresponding to the edges of the triangle in panel a) compared with SA sampling at different sizes. Reprinted from [119].

If one reduces to the $y = 3$ case for visualization purposes, having $E_\gamma > 0$ when $\kappa_E < 0$ means that the whole 2-dimensional simplex (the triangle in Fig. 5.5, *left* part) is such that not even the broken path passing through the barycenter of the

simplex lies at zero energy. As a consequence, typical solutions κ_E cannot be linearly connected without overcoming an energetic barrier. However, by inspecting the heatmap in Fig. 5.5, one realizes that the path with the lowest barrier passes through the barycenter, since the training error related to this point (obtained when imposing $\gamma_r = \frac{1}{y}$)

$$\overline{\boldsymbol{W}} = \frac{\sqrt{N}\sum_{r=1}^{y}\boldsymbol{W}^r}{\|\sum_{r=1}^{y}\boldsymbol{W}^r\|} \tag{5.22}$$

is lower that the maximum of the barrier along the edges. Projecting on one-dimension the energy barrier along the sides of the triangle, which is the error of the geodesic connecting $y = 2$ solutions (Fig. 5.5, *right* side), it is easy to notice that the energy is symmetric with respect to the barycenter, since by construction we are sampling the same kind of solutions. Plus, if one uses SA algorithm to sample solutions as uniformly as possible from the phase space with the quadratic Hinge loss, it is possible to impose the margin with the definition

$$\ell(\Delta_{\kappa_E}) \equiv \Delta_{\kappa_E}^2\Theta(-\Delta_{\kappa_E}) \tag{5.23}$$

where $\Delta_{\kappa_E}^{\mu} = y^{\mu}(w\cdot\xi^{\mu}) - \kappa_E\|w\|$. Optimizing the Hinge loss instead of the number of errors makes the sampling much faster as the SA dynamics benefits from the high-energy solutions organization, while the zero-energy manifold remains unaltered. Indeed, by increasing the model size N, the protocol predictions of the barrier aligns more and more with the theoretical results from replica theory (see how the error behaves as a function of $1/N$ in Appendix B.5).

Case $k > \kappa_E$

When $k > \kappa_E$, the condition $\Theta(\kappa_E c_{\boldsymbol{\gamma}} - k)$ needs to be discussed in three different regimes:

- for $0 \leq \kappa_E \leq k$ the whole convex envelope has zero training error since, as clarified before, the space of solutions is convex.
- For $\kappa_E < k < \kappa_y^{\star}(\alpha, \kappa_E) < 0$ an extended region in the proximity of each corner has zero training error, meaning that as soon as we sample *atypical* solutions, they turns out to be not only surrounded (Chapter 3, 4) but also connected for *small* distances to other solutions. It is not obvious to conclude that all solutions sampled when constructing the envelope and so all the configurations for each interpolation point, have a stability distribution of either the two form in Fig. 5.2. Indeed, we will see in Sec. 5.4.1, that along the polytope we are also sampling solutions with stabilities different from the equilibrium ones. In the case $y = 2$ we can extract the extension of the region at zero energy around each corner: the training error is zero if it is zero for $\gamma \in [0, \gamma_L]$ and $\gamma \in [\gamma_R, 1]$. By symmetry $\frac{1}{2} - \gamma_L = \gamma_R - \frac{1}{2}$ and

$$\gamma_{L,R} = \frac{1}{2} \pm \frac{1}{2}\sqrt{1 - 2\left(\frac{1 - \left(\frac{k}{\kappa_E}\right)^2}{1-q}\right)} \tag{5.24}$$

and one can find the condition for $\kappa_2^\star$ as the value of k in equation (5.24) for which the 1-dimensional simplex lies entirely at zero energy. By imposing $\gamma_L = \gamma_R$ the value of $\kappa_2^\star$ is found by solving the implicit equation

$$1 - \left(\frac{k}{\kappa_E}\right)^2 = \frac{1 - q(k)}{2} \tag{5.25}$$

for k, where we have explicited for convenience the dependence of the overlap q wrt the margin.

- This in turn suggest that, as long as the margin is large enough, (i.e. $k > \kappa_y^\star(\alpha, \kappa_E)$) we can find the whole simplex in dimension $y-1$ is at zero training energy, meaning that $\kappa_E c_{\boldsymbol{\gamma}} - k < 0$ for any value of $\boldsymbol{\gamma}$. Indeed, the previous argument to compute $\kappa_2^\star$ can be easily extended to generic y by reminding that if it has to be $\kappa_E c_\gamma - k < 0$ for all the interpolation points, then the inequality has to hold at fixed κ_E also for the value of c_γ for which it has its minimum, i.e. at the barycenter. When we increase the value of k, by symmetry, this is the last point to have non-zero training error. The condition for which the whole simplex has zero energy is then

$$k > \kappa_E c_{\text{barycenter}} = \kappa_E \sqrt{(1-q)\frac{1}{y} + q} \equiv \kappa_y^\star \tag{5.26}$$

which is consistent with (5.25) for $y = 2$. Since the overlap q increases as the margin k increases, then it easy to see that the thresholds are ordered in y

$$\kappa_2^\star < \kappa_3^\star < \cdots < \kappa_\infty^\star = \kappa_E \sqrt{q} \tag{5.27}$$

and so it will happen, as in Fig. 5.6, that given $\kappa_2^\star < \kappa_3^\star$, the energy barriers along the edges will go to zero faster than the energy at the center of the simplex.

We call these thresholds *coalescence thresholds*, and if $\kappa_2^\star(\kappa_E, \alpha)$ is the minimum value of k that ensures linear connectivity between solutions in the phase space of κ_E, then $\kappa_\infty^\star$ is the threshold above which the projected convex hull of the entire ensemble of solutions lies at zero energy. This stands for the geodesically convex component of the manifold of solutions, whose size can be bounded by the typical overlap q between solutions at $\kappa_\infty^\star$.

In the left panel of Fig. 5.6 we show the training error on the 2-dimensional simplex with the vertices having $\kappa_2^\star < k < \kappa_3^\star$. In that case, since the solutions are extracted with $k > \kappa_2^*$, the geodesic paths between them (i.e. the edges of the triangle) are at zero energy by definition; however, since $k < \kappa_3^\star$, the 2-simplex presents configurations at non-zero training error near and at its barycenter. The projection on 1-dimension of the geodesic path among two solutions with increasing margins is shown in the right panel. Increasing the margin has the effect to lower the barrier between *atypical* configurations (the dots represent as before the predictions of the energy using SA algorithm).

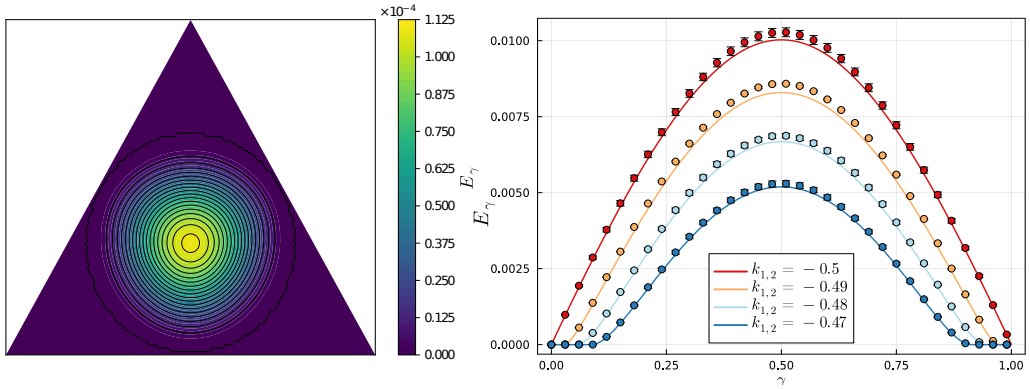


Figure 5.6: (Left panel) Training error at $\alpha = 1$ of the 2-dimensional simplex when $\kappa_2^\star < k < \kappa_3^\star$ and the energy related to the barycenter is strictly greater than zero. (Right panel) One dimensional projection of the barrier between solutions with increasingly large $k_{1,2}$. Extrapolation for large N related to SA predictions are also reported. Reprinted from [119].

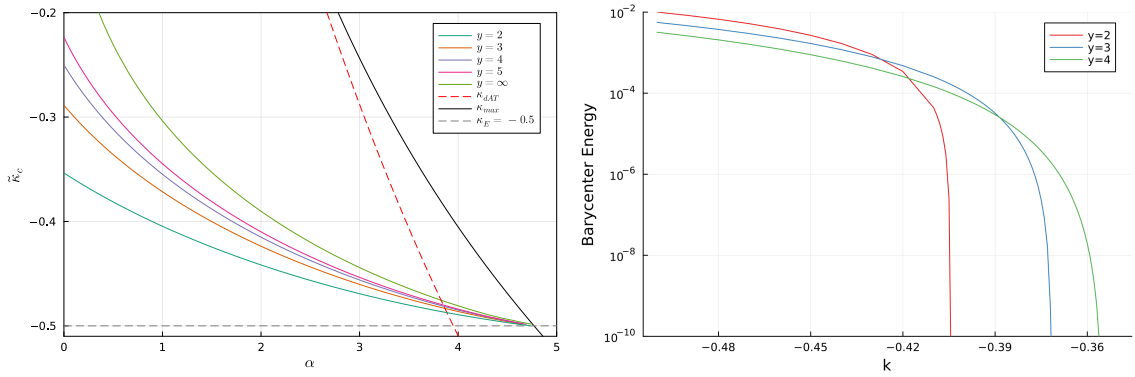


Figure 5.7: (Left panel) Coalescence thresholds as a function of α for $\kappa_E = -0.5$. The dashed red curve stands for the dAT line. (Right panel) The barycenter energy of solutions at margin $\kappa_E = -0.5$ and $\alpha = 1$ as a function of k for different dimensions y of the ensemble. Reprinted from [119].

To clarify the ascending order of the coalescence thresholds, one can also report their values as a function of the density of constraints (Fig. 5.7, *left*). When increasing α and decreasing accordingly the solution space, the distances on average become smaller and the thresholds for zero energy decrease, collapsing to the value κ_E when $\alpha = \alpha_c(\kappa_E)$. In the same way, looking at the barycenter, even if for sufficiently small values of k its energy related to s solutions has training error larger with respect to the barycenter of $s < y$ solutions, the energy goes to zero faster if y is lower (Fig. 5.7, *right*).

5.3.2 Sampling with different margins

We now focus on the connectivity of solutions with different margins. As already mentioned, if we suppose the first $y-1$ vertices to be sampled all with margin k_1 and

the last vertex with margin k_2, then the matrix t_{rs} in Eq. (5.16) will be of the type

$$q^{a\neq b} = t = \begin{pmatrix} q_1 & \dots & q_1 & p \\ \vdots & \ddots & \vdots & \vdots \\ q_1 & \dots & q_1 & p \\ p & \dots & p & q_2 \end{pmatrix} \tag{5.28}$$

with the usual notation. In this case, it is possible to perform some more integral wrt the definition (B.33), so that the expression *simplifies* as

$$E_{\boldsymbol{\gamma}} = \Theta\left(\kappa_E c_{\boldsymbol{\gamma}} - k_1 \sum_{r=1}^{y-1} \gamma_r - k_2 \gamma_y\right) \int \prod_r \mathcal{D}x_1\, \mathcal{D}x_2 \frac{\int_{A_1}^{B^1} \mathcal{D}\lambda_1 \dots \int_{A_{y-1}}^{B^{y-1}} \mathcal{D}\lambda_{y-1} \left(H(A_y) - H(B_y)\right)}{\prod_r H(A_r)} \tag{5.29}$$

with parameters

$$A_r = \begin{cases} \frac{k_r - \sqrt{q_1} x_1}{\sqrt{1-q}}\,, & r \in [y-1] \\ \frac{k_r - \frac{p}{\sqrt{q_1}} x_1 - \sqrt{q_2 - \frac{p^2}{q_1}}\, x_2}{\sqrt{1-q_2}}\,, & r = y \end{cases} \tag{5.30a}$$

$$B_l \equiv \begin{cases} \frac{\kappa_E c_{\boldsymbol{\gamma}} - k_1 \sum_{r=l+1}^{y-1} \gamma_r - k_2 \gamma_y - \sqrt{q} x_1 \sum_{r=1}^{l} \gamma_r - \sqrt{1-q} \sum_{r=1}^{l-1} \gamma_r \lambda_r}{\gamma_l \sqrt{1-q_l}}\,, & l \in [y-1] \\ \frac{\kappa_E c_{\boldsymbol{\gamma}} - \sqrt{q} x_1 \sum_{r=1}^{y-1} \gamma_r - \gamma_y \left[\frac{p}{\sqrt{q_1}} x_2 + \sqrt{q_2 - \frac{p^2}{q_1}}\, x_2\right] - \sqrt{1-q_1} \sum_{r=1}^{y-1} \gamma_r \lambda_r}{\gamma_y \sqrt{1-q_2}}\,, & l = y \end{cases} \tag{5.30b}$$

and the norm $c_{\boldsymbol{\gamma}}^2 = \sum_r \gamma_r^2 + 2\gamma_y p \sum_{r=1}^{y-1} \gamma_r + 2q_1 \sum_{r<s<y} \gamma_r \gamma_s$. The argument of the theta function in (B.40) can be inspected analytically in simple cases. We start from the case $y = 2$. In this case $c_{\boldsymbol{\gamma}} = \gamma^2 + (1-\gamma)^2 + 2p\gamma(1-\gamma)$ and the training error of a point on the geodesic, which is parameterized by $\gamma \in [0, 1]$, vanishes if

$$\kappa_E^2 \left[\gamma^2 + (1-\gamma)^2 + 2p\gamma(1-\gamma)\right] > k_1 \gamma^2 + k_2 (1-\gamma)^2 + 2k_1 k_2 \gamma (1-\gamma). \tag{5.31}$$

If for the present case it is true that $k_1 > \kappa_E$ and $k_2 > \kappa_E$, then an extended region between $\gamma \in [0, \gamma_L]$ and $\gamma \in [\gamma_R, 1]$ will have zero training error with

$$\gamma_{L,R} = \frac{k_2(k_2 - k_1) - \kappa_E^2 (1-p) \pm \kappa_E \sqrt{k_1^2 - 2k_1 k_2 p + k_2^2 - \kappa_E^2 (1-p^2)}}{(k_1 - k_2)^2 - 2\kappa_E^2 (1-p)}. \tag{5.32}$$

Supposing that k_1 is fixed, the expression (5.32) when $\gamma_L = \gamma_R$ reduces to a condition for the margin k_2 such that the two solutions $\boldsymbol{W}^1$ and $\boldsymbol{W}^2$ at overlap p will be geodesically connected at zero training error

$$\kappa_{\text{krn}}(k_1) = k_1 p - \sqrt{(1-p^2)(\kappa_E^2 - k_1^2)}. \tag{5.33}$$

Since when k_1 increases, p increases as well, $\kappa_{\text{krn}}(k_1)$ is a decreasing function of k_1 and so the maximum value of κ_{krn} is obtained when $k_1 = \kappa_E$

$$\kappa_{\text{krn}} = k_1 p. \tag{5.34}$$

What does the threshold mean? For every value of $k_2 \geq \kappa_{\text{krn}}$ all solutions with k_1 will be connected with the typical solutions of the problem ($k_1 = \kappa_E$). This allows to identify the *kernel* of the solution shape (as the subscript of the threshold suggests), which defines, along with the corresponding overlap of solutions at that margin, the subset of solutions that are "visible" - through geodesic paths - from any typical point of the manifold.

In the left-hand panel of Fig. 5.8, the training error is depicted for a simplex composed by $y = 3$ solutions. The scenario involves two vertices at the same margin of the problem $k_1 = \kappa_E$, and a third vertex (belonging to the top of the triangle), which is a robust solution with a margin $k_2 > \kappa_E$. Since in that configuration is $k_2 > \kappa_{\text{krn}}$, the top vertex is geodesically connected to the two base vertices of the simplex. However, as we mentioned, typical solutions are not connected by straight paths at zero-energy, so there is the usual barrier at the lower corner of the heatmap. If the red line marks the boundary between the zero-energy region (*above*) and the non-zero energy area (*below*), this also suggests that it is possible to connect the two typical solutions within broken paths which in the shortest case pass through the barycenter of the triangle.

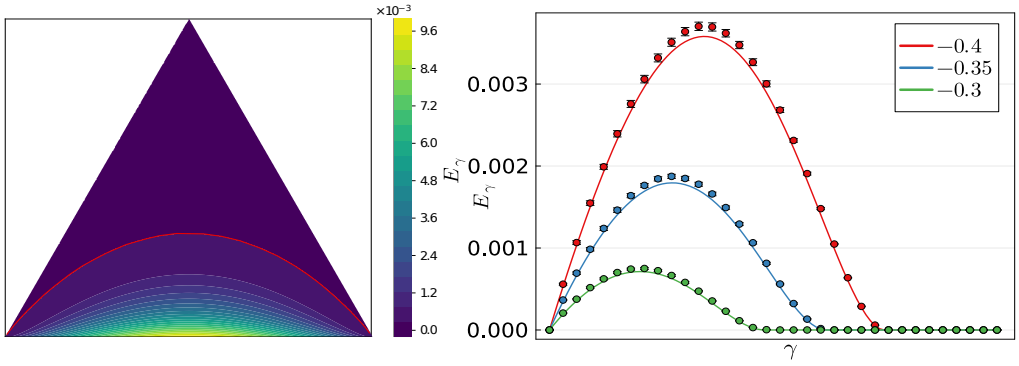


Figure 5.8: (Left panel) 2-dimensional simplex involving the training error between 2 typical solutions (lower vertices, $\kappa_E = -0.5$) and a robust solution (top vertex, $k_2 = -0.1$) belonging to the kernel subset $\kappa_{\text{krn}} \sim -0.171$ for $\alpha = 1$. (Right panel) Training error between a typical solution (left-end point) at margin $k_1 = \kappa_E = -0.5$ and robust configurations at margins $k_2 = -0.4, -0.35, -0.3$. Since now $k_2 < \kappa_{\text{krn}}$ an energy barrier is always present. Reprinted from [119].

As in all the previous cases, the straight path in one dimension is useful to notice that, as long as we increase the right-end margin k_2 while keeping fixed the left one at $k_1 = \kappa_E$, the barrier decreases when moving towards κ_{krn}. There is no more reason now to have a symmetric path with respect to the center. Indeed, as one might intuitively predict, the barrier is more shifted towards typical problem solutions (left), while atypical and robust solutions are always connected to a zero-error neighborhood.

5.3.3 Star-shaped space of solutions

Putting together all the pieces on the connectivity of same and different margin solutions with the hierarchical structure whereby the most robust solutions are the ones innermost in the manifold, it is possible to conclude that the solution space at low constrained density α is *star-shaped*. Even if at fixed κ_E, typical solutions of the problem are not connected by straight linear paths, almost all solutions are geodesically connected through zero energy paths with a subset of atypical high-margin solutions. This subset, which we named the *kernel* is nested into the core of a large geodesically convex component, that we characterized as the set of all configurations with margins $k > \kappa^\star_\infty(\alpha, \kappa_E)$. Notice that it is straightforward to conclude that the kernel lies inside the convex region since $q_{\rm krn} > q^\star_\infty$. A sketch on the phase space, together with a detailed zoom in of solutions with the corresponding margin, are reported in Fig. 5.9.

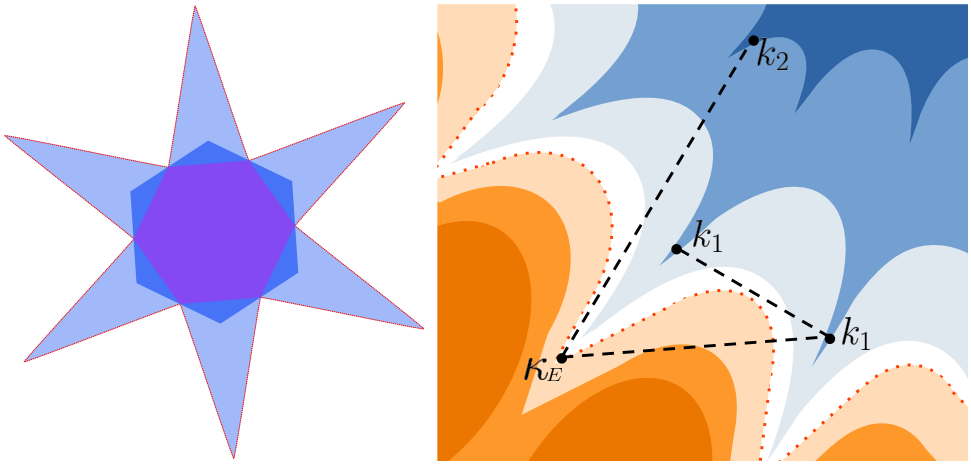


Figure 5.9: Sketch (*left*) and zoom (*right*) of the star-shaped organization of the solution manifold. In both cases the blue shades corresponds to zero energy regions, while the red ones to UNSAT regions for the problem. Reprinted from [119].

The simple 2-dimensional visualization of the solution manifold is represented in the left panel (Fig. 5.9). If the red edges bound the zero-energy region, one can symbolically image the typical solutions at κ_E to belong to the tips of the star. The large geodesically convex component is drawn in dark blue with inside, in purple, the kernel of solutions. Imagine looking at a zoom of the tip of the star: if the red dotted line represents again the border of the connected manifold of solutions, then configurations that satisfy margins larger than the one of the problem $\kappa_E < k_1 < k_2$ are organized in a nested structure (darker shades of blue). When a typical solution with margin κ_E is geodesically connected with a solution with margin k_1, an energy barrier (acrossing of the orange region) is observed. However, the k_1 solutions belong to the convex sub-manifold (the geodesic path falls within the white-blue region). Solutions with even higher margin k_2, located in the kernel, are connected to almost any other solution.

The phase diagram in the plane (κ, α) with all the new found transitions (Fig. 5.10)

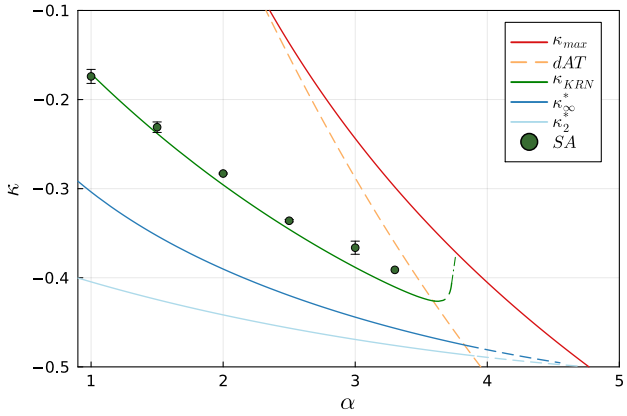


Figure 5.10: Coalescence threshold lines at $\kappa_E = -0.5$ (blue and cyan), and the κ_{krn} threshold (green line) as a function of α. In orange the dAT transition line, delimiting the RS-stable region; in red the RS estimate of the κ_{max}. Above the dAT line, dashed lines indicate the continuation of the (unstable) RS predictions. Points are numerical extrapolations from SA samples. Reprinted from [119].

is to be read as follows: above each transition curve related to a $\kappa_x(\alpha)$, the linear connection between x solutions occurs. The theoretical curves are reliable until dAT transition line, signaling that symmetry-breaking effects are coming into play when $\alpha(\kappa) > \alpha_{\text{dAT}}(\kappa)$ as the hierarchical structure for which $q_1 < p < q_2$ if $k_1 < k_2$ is no longer valid and κ_{krn} suddenly grows for $\alpha > \alpha_{\text{dAT}}$. Extrapolations points (in *green*) at large N for SA suggest that the $\kappa_{\text{krn}}(\alpha)$ estimate for κ_E solutions to be connected with the kernel subset, well approximates the replica theory analysis.

5.4 More features on the solution space

Until now, the protocol by which we have characterized the solution space involved sampling from the Gibbs equilibrium measure robust solutions or at the same margin of the problem and studying their connectivity. Since, as anticipated in section 5.4.1, the zero-error manifold also includes non-equilibrium solutions, which together with the equilibrium solutions at large k, form those regions in the landscape that are attractive to the dynamics of the algorithms, it is useful to characterize their stability. In addition, we mentioned that this star shaped set of solutions is anisotropic with respect to the hierarchical structure, so those solutions at κ_{max} for which the entropy goes to zero (definitions as in chapters 3, 4), do not coincide with the barycenter of typical configurations.

5.4.1 Non-equilibrium solutions encountered along the path

To understand which type of configurations are found along the geodesic path between two solutions, we derive an analytic expression for the stability distribution on the geodesic connecting two weights $\boldsymbol{W}_\gamma = \sqrt{N}\frac{\gamma \boldsymbol{W}^1 + (1-\gamma)\boldsymbol{W}^2}{\|\gamma \boldsymbol{W}^1 + (1-\gamma)\boldsymbol{W}^2\|}$, with margin k_1 and k_2,

i.e.

$$
\begin{aligned}
P(\Delta) &= \mathbb{E}_{\boldsymbol{\xi}} \left\langle \delta\Big(\Delta - \frac{\boldsymbol{W}_{\boldsymbol{\gamma}} \cdot \boldsymbol{\xi}^1}{\sqrt{N}}\Big) \right\rangle_{k_1,k_2} \\
&= \mathbb{E}_{\boldsymbol{\xi}} \frac{1}{Z_{\boldsymbol{\xi}}(k_1,k_2)} \int d\boldsymbol{W}^1 d\boldsymbol{W}^2 \, \mathbb{X}_{\boldsymbol{\xi}}(\boldsymbol{W}^1;k_1)\mathbb{X}_{\boldsymbol{\xi}}(\boldsymbol{W}^2;k_2)\, \delta\left(\Delta - \frac{\mathbf{W}_{\boldsymbol{\gamma}} \cdot \boldsymbol{\xi}^1}{\sqrt{N}}\right)
\end{aligned}
\tag{5.35}
$$

being $\mathbb{X}_{\boldsymbol{\xi}}(\boldsymbol{W};k)$ the characteristic function which is the zero temperature limit of the Gibbs factor with the number of error as energy for a solution $\boldsymbol{W}$ with margin k. By imposing an RS ansatz and moving on with the computation, in the end the expression reduces to $P(\Delta) = f_\gamma(\{(q_{rs}^{ab})^\star\}, \Delta)$, where $\{(q_{rs}^{ab})^\star\}$ are the overlap values which extremize the free entropy (details in B.3). It turns out that, while interpolating along the path between the two solutions, the threshold value up to which the distribution is zero (it has to exist since we are considering the error counting loss) is

$$
\Delta_{eff} = \frac{\gamma k_1 + (1-\gamma)k_2}{c_{\boldsymbol{\gamma}}} \tag{5.36}
$$

which at the path's extremes reverts to the thresholds of the two typical solutions with margins $\Delta_{eff}|\gamma = 0 = k_2$ and $\Delta eff|_{\gamma=1} = k_1$. For every γ within the range of $(0,1)$, instead, it can be seen as the effective margin of solutions along that path. In Fig. 5.11, the stability distribution for solutions along the geodesic is displayed for $k_2 = 0.1 > \kappa_{\text{krn}}$. Consistent with the definition of κ_{krn}, it is observed that if $k_2 > \kappa_{\text{krn}}$, then $k_1 \leq \Delta_{eff} \leq k_2$ for all $\gamma \in [0,1]$, suggesting that every configuration along the geodesic is a solution. While he presence of a sharp threshold separating zero from non-zero values of the stability distribution is reminiscent of the typical/atypical solutions case (Fig. 5.2), the functional form of the distribution itself beyond this threshold is qualitatively different from the one of typical solutions at a given margin value k (i.e. it does not follow a truncated Gaussian). This aligns with the anticipation: solutions along the geodesic do not align with the equilibrium measure at that k, hence their sampling is not under control.

5.4.2 Anisotropy of the solution manifold

To support the degree of anisotropy of the phase space around high margin solutions, one can consider the barycenter of an ensemble of κ_E-margin configurations and study its overlap with respect to the $\kappa_{\max}$ solution at that α. As in previous chapters, once the threshold for the satisfiability κ_E has been decided, all configurations with $\kappa_E \leq k \leq \kappa_{\max}(\alpha)$ will be solutions since $\kappa_{\max}(\alpha)$ represents the SAT/UNSAT transition line.

The barycenter of y independently-sampled replicas of the system at a given margin κ_E can be defined as

$$
\overline{\boldsymbol{W}}(\kappa_E) = \frac{\sum_{a=1}^{y} \boldsymbol{W}^a(\kappa_E)}{||\sum_{a=1}^{y} \boldsymbol{W}^a(\kappa_E)||} \tag{5.37}
$$

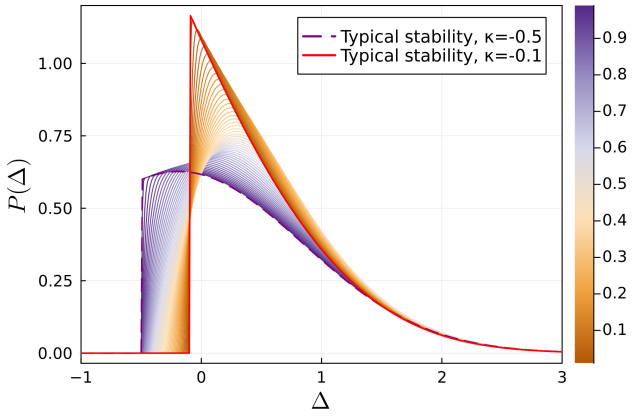


Figure 5.11: Stability distributions of solutions found by increasing γ along the path between a typical solution at $k_1 = -0.5$ towards a more robust solution at $k_2 = -0.1$ ($\alpha = 1$). The geodesic connecting the two solutions is at zero training error. Purple and red lines at the extremes of the curves represent the typical stability distribution for $\kappa_E = -0.5$ and $\kappa_E = -0.1$ respectively. Reprinted from [119].

where the projection on the sphere is necessary to ensure the vector $\overline{\boldsymbol{W}}(\kappa_E)$ is correctly normalized: $\|\overline{\boldsymbol{W}}(\kappa_E)\|^2 = 1$. Once having the overlap p between each of the y solutions and the $\kappa_{\max}$, the overlap with the barycenter is straightforward to define

$$\bar{p} = \frac{1}{N}\langle \overline{\boldsymbol{W}}(\kappa_E), \boldsymbol{W}(\kappa_{\max})\rangle = \frac{p\left(\kappa_E, \kappa_{\max}\right)}{\sqrt{\frac{1}{y} + \frac{y-1}{y}\, q(\kappa_E)}} \xrightarrow{y\to\infty} \frac{p\left(\kappa_E, \kappa_{\max}\right)}{\sqrt{q(\kappa_E)}} \tag{5.38}$$

and it is possible to study its value for different value of κ_E and y as a function of α (Fig. 5.12). When y is finite (*dashed* lines) and α is zero, so that there are no constraints to satisfy, since we are randomly sampling two vectors from the N-dimensional sphere, w.h.p. it is clear how $\bar{p}\left(\kappa_E, \kappa_{\max}; \alpha = 0\right) \simeq 0$ independently from the margin κ_E. In general, when y is small we do not expect the barycenter of y solutions to be a good proxy for the central region in which robust minimizers are located. Indeed, $\bar{p}\left(\kappa_E, \kappa_{\max}; \alpha\right) < 1$ for all the ranges of α and all values of κ_E, of course increasing with α since because the volume of solutions itself is decreasing with α, the distances are decreasing between solutions on average.

In the limit $y = \infty$, the barycenter coincides with $\kappa_{\max}$ for $\alpha = 0$ (i.e. the overlap is 1 independently from κ_E), but even when reducing to the convex problem (*blue* curve) where $\kappa_E = 0$, we have $\bar{p}\left(\kappa_E = 0, \kappa_{\max}; \alpha > 0\right) \neq 1$ when increasing α. The fact that the overlap $\bar{p}$ is different from one in this case indicates the anisotropy of the system with respect to the robust solution. That is, for the $\kappa_E = 0$ case, we have that the overlap $\bar{p}$ is equal to 1 only at the critical storage capacity $\alpha^c_{\mathrm{RS}} = 2$ bounding the SAT/UNSAT transition.

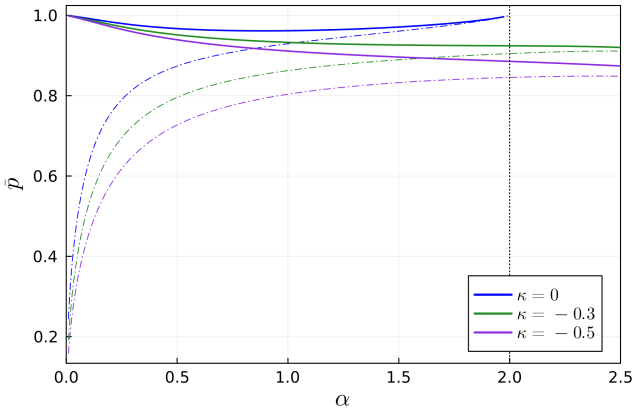


Figure 5.12: Typical overlap $\bar{p}$ between the barycenter of y independently-sampled solutions with margin κ_E and the maximum margin solution at $\kappa_{\max}(\alpha)$. The dashed curves refer to the $y = 3$ case while the full ones are for $y = \infty$. After the vertical dashed line ($\alpha_{\text{dAT}}(\kappa_E = 0)$) the predicted value of $\kappa_{\max}(\alpha)$ is not exact (RS is unstable).

5.5 Sampling bias and disconnection

We compare the properties of solutions found with different solvers on instances of the negative perceptron with $\kappa_E = -0.5$. Relating the dynamical bias of commonly used and highly performing protocols to the linear connectivity of different regions of the solution space is indeed crucial to understand why in practical applications LMC is an evidence despite the fact that the convex subspace does not cover the whole volume of solutions.

First of all, since we have an algorithm (chapter 2) to sample maximum margin solution, that is fBP, and since by construction $\boldsymbol{W}^1$ with $\kappa_{\max}$ belongs to the kernel subset, we can prove the star-sharpness by looking at experimental evidences on the straight path errors between $\boldsymbol{W}^1$ and minimizers sampled with different strategies. Studying how the landscape evolves when increasing the constraint density α and how precisely RSB transitions affect the linear connectivity, is the purpose of this section.

Typical solutions are approximated by carefully sampling through SA protocol (using Hinge loss as in Eq. (5.23)), whose predictions for energetic barriers (Fig. 5.5) and stabilities (*inset* of Fig. 5.13) are effective at large system size when the computational times to find the local minima are affordable. Non-zero energy barriers between SA and the fBP solutions seem to appear in close proximity of the dAT transition line (Fig. 5.13, *red* dots), where the space breaks up and it is no longer possible to rely on straight paths to move towards the star core by starting from the tips. SGD on the cross-entropy loss – the most common optimization objective for this class of problems – yields *robust* solutions [48] with higher average stability than typical, as shown in the inset of 5.13. Here, the disconnection transition wrt to the core is delayed compared to SA and seems to appear in close proximity of the LE transition for the model [123]. To complete the picture on one ends of the spectrum for which solutions *less robust* than typical can be sampled, the classic Perceptron Algorithm

(PA) for sufficient small learning rate, is instead able to sample solutions with a large mass of stabilities at threshold (*inset* of Fig. 5.13). Since they have smaller effective stability, the disconnection with $\kappa_{\max}$ is in this case anticipated before the dAT line. So it seems that the stronger the bias towards high-margin regions and later the disconnection in α of the convex or nearly convex subset to which robust solutions belong.

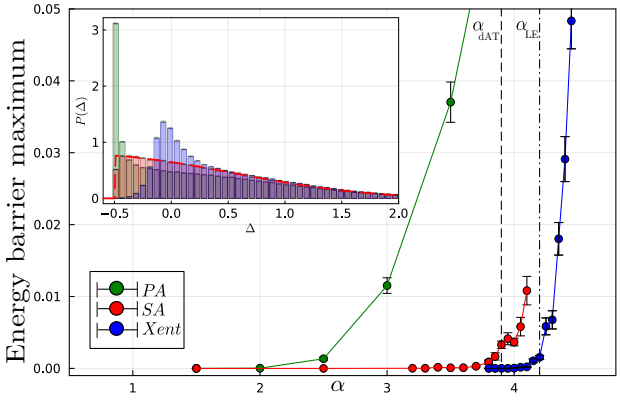


Figure 5.13: Maximum training error along the geodesic path ($y = 2$) connecting the fBP solution with $\kappa_{\max}$ at a given α, with different solutions found by PA, SA and SGD. Non-zero energies along the path indicate disconnection in the solution space. Vertical dashed lines refer to dAT and LE transitions. The *inset* reports the stabilities associated with algorithms solutions. Red dashed line is the theoretical prediction for a typical solution with $\kappa_E = -0.5$. Reprinted from [119].

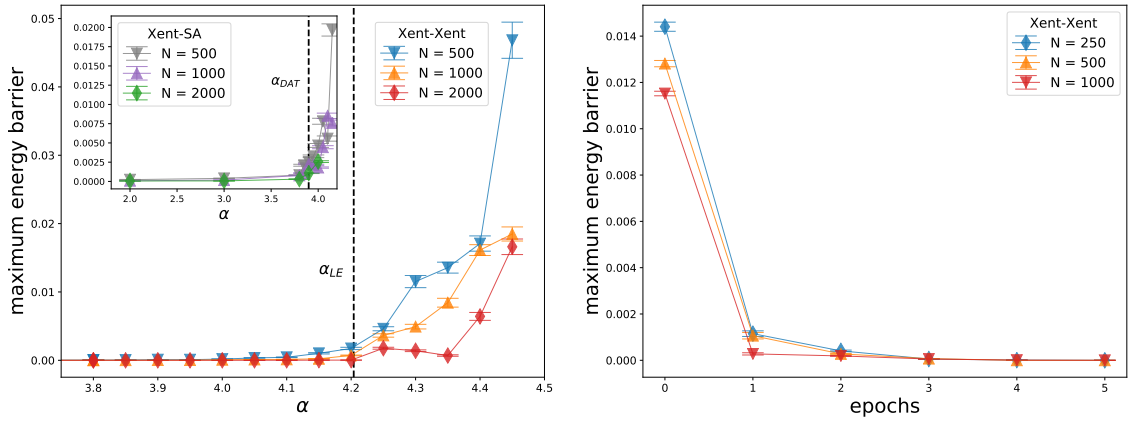


Figure 5.14: (*Left*) Maximum barrier height along the geodesic path connecting two SGD solutions obtained with the cross-entropy loss as a function of α. For large system size (N), these solutions disconnect at the α_{LE} transition. The *inset* refers to the appearance of barriers between SGD and SA at α_{dAT}. (*Right*) How the energy barrier decreases for SGD as a function of epochs when initializing the algorithm to typical (i.e. SA) solutions. Reprinted from [119].

It is also possible to analyze the inductive bias of SGD dynamics with the cross-entropy loss by evaluating the average maximum energy barrier's height between pairs of independent solutions (means are over different initial conditions). As in the main Fig. 5.14 (*left*), we observe that randomly initialized trajectories on the $N-$ sphere access the zero-energy manifold of solutions by entering directly in the geodesically convex component. SGD-SGD solutions are then linearly connected at zero error until α_{LE} (as it is for fBP-SGD pairs in Fig. 5.13). Instead, by looking at the energetic barrier among SGD-SA solutions (*inset*,Fig. 5.14), they start to disconnected at α_{dAT} line, which means that the disconnection starts from the outside and propagates towards the core of the star progressively when increasing α.

If we consider pairs of trajectories that begin within the zero-energy region, corresponding to typical solutions identified by SA, we realize (Fig. 5.14, *right*) that the initial energy barrier between these points, though initially non-zero, rapidly decreases to zero within a few epochs. So, SGD naturally drifts to the inner and convex region of the solution space. We do not have reasons to believe that this behaviour is specific to the problem in exam and, although more complex the metrics when dealing with deeper NNs, one could imagine to extend the characterization of kernel and convex subsets for more complex architectures. We refer to Appendix B.4 for numerical details concerning simulations.

Chapter 6

The rewards of a negative margin

The purpose of the present brief Chapter is to give an hint on how it is possible to play with the landscape structure with the advantage of a computational cost when one expects a hierarchical arrangement of solutions with margins as deeply described until now [90, 99, 119]. This is inspired by a work in preparation [128].

Suppose that you want to solve a classification problem where the rule that generates the classes relies on a teacher structure to which, of course, you do not have access. It might be the case that the teacher is noisy: either the data are blurred by a noise that is independent from the examples, the rule itself is noisy, the labels are corrupted, or simply the student gets a nonlinear combination of patterns from the teacher (which will be the case). If you are in that situation, then it will happen that, while trying to optimize over the training error, you realize that it is no longer possible to infer the rule since the training set is damaged. At least, this is true if you have a large training set and the density $\alpha > \alpha_c(\eta)$, that is the threshold above which there is no student that is capable to reproduce all classifications (given η the noise magnitude). Is that really a bad thing? Given that the purpose is to generalize, is it always necessary to reach zero training error in order to do so in an efficient way? In realistic scenarios with noisy data, we know today that the way NNs are trained is precisely to handle the noise and achieve generalization without perfectly fitting the training set. Common optimization strategies to tackle that issue concern regularization techniques, dropout or early stopping, all of them with the objective of keeping stable the prediction performances while avoiding overfitting. However, most of the times you need a validation set to understand when it happens that the performances of the network start to deteriorate as the model started to fit the noise. What if one would like to avoid wasting time with a validation set? Is there an alternative for training networks that comes from understanding the landscape geometry?

On the pages to come, we show how it is possible to enlarge the solution space by relaxing the constraints with a negative margin, both when the student (a RFM) sees by construction some sort of effective noise in the data, and in realistic scenarios where usually to obtain good generalization you have to optimize the network norm. We will see how, by increasing the solution space, one can leverage on ensembling methods that help finding the best generalization protocols.

6.1 The negative RFM

As an example of model with some simple structure in the data, which can be regarded as a structured noise, let us consider the mismatched classification problem as described in Chapter 4. Briefly, we have a teacher perceptron with weights $\boldsymbol{w}^{\boldsymbol{T}}$ in the D-dimensional space ($\kappa = 1, \dots, D$) that assigns to i.i.d. standard normal random input variables $\boldsymbol{\xi}^\mu$ (with $\mu = 1, \dots, P$) a label y^μ, via the rule

$$y^\mu = \text{sign}\left(\frac{1}{\sqrt{D}}\sum_{k=1}^{D} w_k^T \xi_k^\mu\right). \tag{6.1}$$

Now, if we can express the effective patterns the student has access to $\tilde{\boldsymbol{\xi}}^\mu \in \mathbb{R}^N$ as given by a Gaussian covariate model

$$\tilde{\xi}_i^\mu = \sigma\left(\frac{1}{\sqrt{D}}\sum_{k=1}^{D} F_{ki}\xi_k^\mu\right) = \mu_0 + \frac{\mu_1}{\sqrt{D}}\sum_{k=1}^{D} F_{ki}\xi_k^\mu + \mu_\star \eta_i^\mu \tag{6.2}$$

for which each $\tilde{\boldsymbol{\xi}}^\mu \in \mathbb{R}^N$ is a linear combination of the pattern components $\boldsymbol{\xi}^\mu \in \mathbb{R}^D$ plus some noise depending on the degree of non-linearity of the transfer function σ and where $\eta_i \sim \mathcal{N}(0,1)$ is an i.i.d. standard Gaussian random variable, then the student has the second layer of weights $\boldsymbol{w}$ in dimension N to perform the classification with a margin κ

$$\hat{y}^\mu = \text{sign}\left(\frac{1}{\sqrt{N}}\sum_{i} \tilde{\xi}_i^\mu w_i\right). \tag{6.3}$$

Studying the performance of that model, i.e. deriving training and generalization errors in the typical case as a function of $\alpha = \frac{P}{N}$ and $\alpha_D = \frac{D}{N}$ and for a generic loss function $\ell_{\mathcal{D}}(\boldsymbol{w}; \kappa)$, means to extremize the free energy expression $\phi = \lim_{N,P,D\to\infty} \frac{1}{N}\langle \ln Z\rangle_{\xi,F}$ in the thermodynamic limit, solving saddle point equations for the order parameters under certain ansatz over the solution structure. With respect to the derivation in Chapter 4, if one has not the error counting loss but a generic loss function, just changes the energetic term contribution in the Gibbs free energy (4.26), that it is now

$$\begin{aligned}\mathcal{G}_E &= \frac{1}{n}\ln\int\frac{du\,d\hat{u}}{2\pi}\int\prod_a \frac{d\lambda^a d\hat{\lambda}^a}{2\pi} e^{iu\hat{u}+i\sum_a \lambda^a\hat{\lambda}^a - \frac{(1+\eta^2)}{2}\hat{u}^2 - \frac{1}{2}\sum_{ab} Q^{ab}\hat{\lambda}^a\hat{\lambda}^b - \hat{u}\sum_a M^a\hat{\lambda}^a} e^{-\beta\sum_a \ell\left(\text{sign}(u)\lambda^a - \kappa||\boldsymbol{w}||\right)} \\ &= 2\int Dx\, H\Big(-\frac{M}{\sqrt{Q-M^2}}x\Big)\ln\int D\lambda\, e^{-\beta\ell(\sqrt{Q_d-Q}\lambda+\sqrt{Q}x-\kappa\sqrt{h})}\end{aligned} \tag{6.4}$$

where between the first and second lines we avoided standard steps concerning the replacement of the RS ansatz (see 4.2.1). Notice also that now the margin is rescaled with the norm $||\boldsymbol{w}||$ of the model, which is itself an order parameter that we call $h = \frac{1}{N}\sum_i (w_i^a)^2$, being $a \in [n]$ the replica index of the n copies of the system. It is

necessary to rescale the margin in this context, to be sure that, regardless of the norm, the problem we are trying to solve is in the κ margin instance, and is not being solved for an effective κ_{eff} that can be very different to the actual κ if the norm is either big or small. By proceeding in an agnostic way, the standard protocol to implicitly optimize over the prediction (i.e. test) error, is to at least reach zero training error, and this means to perform the $\beta \to \infty$ limit of the expression (6.4)

$$\mathcal{G}_E = 2\alpha \int Dx\, H\Big(-\frac{M}{\sqrt{Q_d - M^2}}x\Big) \max_y \Big[-\frac{y^2}{2} - \ell\Big(\sqrt{\delta Q}\, y + \sqrt{Q_d}\, x - \kappa\sqrt{h}\Big)\Big]. \quad (6.5)$$

The training error becomes accordingly

$$\begin{aligned}\epsilon_t &= \lim_{\beta\to\infty}\lim_{N\to\infty} \mathbb{E}_{\xi,F}\langle \Theta\left(-\text{sign}\Big(\frac{1}{\sqrt{D}}\sum_{k=1}^{D}\xi_k^1 w_k^T\Big)\Big(\frac{1}{\sqrt{N}}\sum_i \sigma\,\Big(\frac{1}{\sqrt{D}}\sum_k F_{ki}\xi_k^1\Big)w_i\Big) + \kappa||W||\right)\rangle \\ &= 2\int Dx\, H\Big(-\frac{M}{\sqrt{Q-M^2}}x\Big)\,\Theta\left(-\sqrt{\delta Q}\,\lambda^\star - \sqrt{Q_d}\,x + \kappa||W||\right)\end{aligned} \quad (6.6)$$

given $\lambda^\star = \arg\min_\lambda \left[\frac{y^2}{2} + \ell\Big(\sqrt{\delta Q}\, y + \sqrt{Q_d}\, x - \kappa\sqrt{h}\Big)\right]$ and the fact that the free energy ϕ (4.26) is not affected by the observable ϵ_t in the large N limit. Now, if we want to come back to the standard case with a loss counting the number of errors, it is just sufficient to replace $\ell_{NE}(x) = \Theta(x)$ in the expression (6.6) with the role of a loss per pattern, while in general we can inspect what happens to the space of solutions sampled with cross-entropy (CE) loss if we substitute

$$\ell_{CE}(x,\gamma) = \frac{1}{2\gamma}\log(1 + e^{-2\gamma x}). \quad (6.7)$$

Notice that, introducing a different loss function can dramatically change the nature of the optimization problem. Indeed, since $\mathcal{L}(\boldsymbol{w}) = \sum_{\mu=1}^{P} \ell_{CE}(y^\mu \hat{y}^\mu - \kappa||\boldsymbol{w}||;\gamma)$, which is the loss over the dataset $\mathcal{D} = \{\boldsymbol{\xi}^\mu, y^\mu\}^P$, is convex and the constraints are linear in the weights $\boldsymbol{w}$, then for $\kappa \geq 0$ the problem itself becomes convex, but it is not for $\kappa < 0$ as it is in 5.

In general, when using local search optimization strategies such as SGD, γ in the loss function (6.7) plays a crucial role in influencing the norm of the model. Indeed, since the parameter acts as a scaling factor for the alignment between the prediction and the actual label, it adjusts the steepness of the loss function with respect to that. A larger γ amplifies the effect of deviations from correct predictions, and it can effectively acts as a form of regularization. By heavily penalizing certain types of errors and low confidence, it prevents the model from fitting too closely the training set $\mathcal{D}$, penalizing large weights. However, if γ is too high, the model might become too conservative, potentially leading to underfitting (high bias). If it is too low, the model might not adequately learn the underlying patterns in the data (high variance). So, determining the optimal value of γ is a task-dependent and model-dependent procedure, which itself requires to monitor both the training and validation performance of the model to assess its generalization capabilities.

In this context, especially when the networks are large and require regularization to avoid overfitting, one might consider attempting an approach guided by the geometry of solutions in the landscape. Let's assume that the goal is to generalize on the task in question at $\kappa' = 0$. As anticipated, one might think of choosing an adequate γ, or equally to fix a γ and looking for an optimal norm, and studying the generalization performance over a new extracted pattern $\{\boldsymbol{\xi}^\star, y^\star\}$ (the derivation is the same as in Section 4.2.2)

$$\epsilon_g \equiv \langle \mathbb{E}_{\boldsymbol{\xi}^\star} \Theta\left(-y^\star \hat{y}^\star\right) \rangle \tag{6.8}$$

of a solution sampled with cross-entropy loss with a margin κ and from the usual measure

$$P_{\mathcal{D}}(\boldsymbol{w}; \kappa) = \lim_{\beta \to \infty} \frac{e^{-\beta \sum_\mu \ell_{CE}(y^\mu \hat{y}^\mu - \kappa ||\boldsymbol{w}||)}}{\int d\boldsymbol{w'}\, e^{-\beta \sum_\mu \ell_{CE}(y^\mu \hat{y}^\mu - \kappa ||\boldsymbol{w'}||)}}. \tag{6.9}$$

To avoid confusion, let us just clarify that we can always do that: sampling a solution of a given problem κ and evaluating its generalization performance with respect to a problem $\kappa' = 0$. Now the question is: is there an advantage to choose in principle a $\kappa' \neq \kappa$? Proceeding by analogy with chapters 3 and 4, it seems obvious that choosing a robust solution is advantageous for performance, and that, contrarely, relaxing the constraints of the problem at $\kappa' = 0$ and sampling a less robust solution with $\kappa < \kappa'$ (negative), is not convenient. In fact, in Fig. 6.1, we see that once $\gamma = 3$ is chosen, we can study the generalization error at $\kappa' = 0$, as well as the corresponding training error of the configuration at the κ of the instance from which we are sampling, across a whole span of κ.

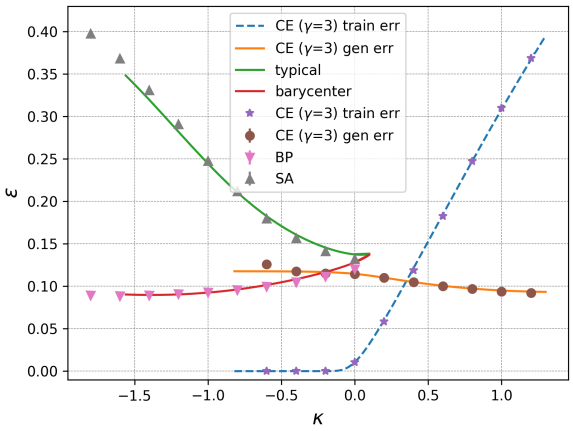


Figure 6.1: Generalization error as a function of κ and with $\alpha = 5, \alpha_T = 50$ for different sampled solutions and numerical simulations. In *green* the typical test error of a solution of the error-counting loss at $\beta \to \infty$; SA with Hinge loss and $N = 1000$ is over that curve. In *orange* the generalization of a CE-loss solution with $\gamma = 3$ and norm fixed to 1; SGD with the same parameters and $N = 1000$ agrees with the theory (in dashed lin and purple dots the corresponding training error). In *red* the test error of the barycenter of typical solution at the given κ (BP with $N = 1000$ matches perfectly the theory).

Take for instance a constraint density $\alpha = 5$ as in Fig. 6.1, where if we look at the corresponding training error for $\beta \to \infty$, the student is not able to solve the problem at $\kappa' = 0$ ($\epsilon_t(\kappa' = 0) \neq 0$). In this case, the generalization of the solution sampled with CE, and consequently the numerical data of SGD at a fixed norm and with the same γ (not having optimized on either parameter), tell us that to generalize a $\kappa' = 0$, the best possible approach is to sample robust solutions with margins $\kappa > \kappa'$. Indeed, the generalization error here is a decreasing function of the margin (as in Chapter 4), and it is not necessary to ensure fitting the training set to achieve good generalization error, consistent with what is seen in practical applications. Similarly, but with much worse performance, if we refer back to the typical case with the loss being the number of errors, and thus we sample a solution from

$$P_{\mathcal{D}}(\boldsymbol{w};\kappa) = \lim_{\beta\to\infty} \frac{e^{-\beta\sum_\mu \Theta(y^\mu\hat{y}^\mu - \kappa||\boldsymbol{w}||)}}{\int d\boldsymbol{w'}\, e^{-\beta\sum_\mu \Theta(y^\mu\hat{y}^\mu - \kappa||\boldsymbol{w'}||)}} \tag{6.10}$$

the green line in Fig. 6.1, and accordingly SA with squared Hinge loss

$$\ell_H(\boldsymbol{w};\kappa) \equiv (y^\mu\hat{y}^\mu - \kappa||\boldsymbol{w}||)^2\Theta(-y^\mu\hat{y}^\mu + \kappa||\boldsymbol{w}||) \tag{6.11}$$

both decrease with κ. They tend towards the random case $\epsilon_g \to 0.5$ when $\kappa \to -\infty$, which is equivalent to say that no constraints are left for the task ($\alpha \to 0$), and they stop at $\kappa_{\max}(\alpha)$, that is when, beyond the threshold, there are no longer solutions left in the phase space. Notice that we no longer need to report the training error when reducing to the Theta-loss case, since the measure at zero temperature reduces, up to a normalization factor, to the uniform characteristic function over all solutions for that given κ and α.

But what happens if, instead, we refer back to the nested organization of solutions with margin as in Chapter 5? Roughly speaking, in the context of linear interpolation paths, we saw that, when sampling solutions with margins $\kappa \geq \kappa_{\text{krn}}$, which was the threshold for the geodesically convex component in the problem of a negative Perceptron, the stability of solutions along the interpolation suggested that multiple solutions with higher margins with respect to κ (i.e. the extremes) are encountered along the path. In the same way, by extension, if the present goal is to generalize well at $\kappa' = 0$, one can imagine to relax some constraints and sample with the Theta loss a infinite number of solutions with $\kappa < \kappa' = 0$. Taking the barycenter of these independently sampled solutions, or studying the Bayesian test error at $\kappa' = 0$ as in (4.37)

$$\epsilon_g^B = \mathbb{E}_{\boldsymbol{\xi}^\star}\, \overline{\Theta\left(-y^\star\langle\hat{y}^\star\rangle_{\boldsymbol{w}|\{\boldsymbol{\xi}^\mu\}}\right)} \tag{6.12}$$

leads to the red curve in Fig. 6.1. The generalization error of the centroid has a minimum at a negative κ, which means that, by relaxing the constraints and thus increasing the volume of the solution space, with ensembling it is possible to find a resulting model that is robust and high-performing. Notice also that, as it should be, the point for which the centroid generalization reaches the ϵ_g of a typical solution is at $\kappa_{\max}(\alpha)$, since at that margin the entropy of the corresponding configuration is zero. Now, if we look at the minimum of the barycenter generalization compared to the

minimum of a CE solution, the centroid gains slightly in performance, but it is clear what we are doing: taking the barycenter of solutions with a negative margin has the effect of a regularization. The attribute is stable also towards noisy perturbations of the patterns in the training set. Indeed, in the case for which the input patterns of the teacher are blurred according to the distribution

$$P(\xi_k^{\mu'}) = \frac{1}{\sqrt{2\pi\eta^2}} e^{-\frac{(\xi_k^{\mu'} - \xi_k^{\mu})}{2\eta^2}} \tag{6.13}$$

where η characterizes the noise strength, the patterns accessible to the student are always the true ones but the corrupted rule (i.e. the noisy teacher) changes the labels in the dataset. In the context, the noise just changes the variance of the teacher preactivation and accordingly the energetic term in the free energy is now

$$\mathcal{G}_E = 2\alpha \int Dx\, H\Big(-\frac{M}{\sqrt{Q_d(1+\eta^2) - M^2}}x\Big) \max_y \Big[-\frac{y^2}{2} - \ell\Big(\sqrt{\delta Q}\, y + \sqrt{Q_d}\, x - \kappa\sqrt{h}\Big)\Big]. \tag{6.14}$$

Notice that, by increasing the noise strenght at a fixed α, α_T values, the barycenter error becomes worse in the span of κ. However, the optimum of ϵ_g^B shifts towards more negative κ values when increasing η (Fig. 6.2). Plus, decreasing α in the context (*right* panel) while keeping fixed α_T means to increase the overparameterization degree of the student. As a consequence, the landscape simplifies and the generalization of the barycenter at different η is flattened throughout the span of κ.

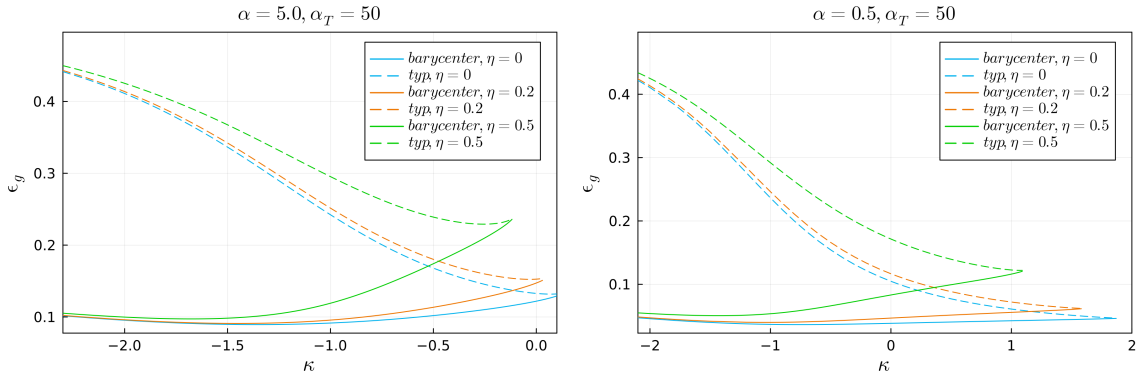


Figure 6.2: Generalization errors as a function of κ for $\alpha = 5$ (*left*) and $\alpha = 0.5$ (*right*). For each subplot, the full curves correspond to the barycenter error, while the dashed lines to typical solutions of the Theta loss at that κ. Different values of the noise strength η are displayed.

What if one would like to reach a comparable performance with CE, by fine-tuning for the optimal norm or by fixing the norm and fine-tuning for the optimal γ? As shown in Fig. 6.3, decreasing γ has per se the regularizing effect we were looking for, making now the generalization error nearly constant in all the range of κ. However, the minimum of the barycenter is not reached by any test error value of CE. Because of the flat shape of the test error with CE with a proper γ, in the

context of explaining performances as an optimal/nearly optimal local organization structure of the corresponding solution, we can look at the training error on the left side of Fig. 6.3: also in this case, one could potentially relax some constraints, having an advantage in computational cost, and look during training for the optimal value of γ of a solution with negative κ.

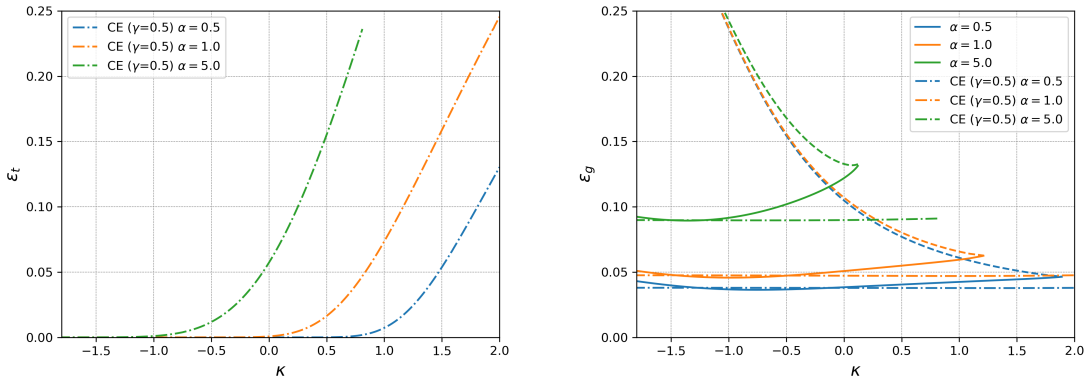


Figure 6.3: (*left* panel) Train error as a function of κ for solution at different α sampled wit CE with fixed norm and $\gamma = 0.5$. (*right* panel) Test errors wrt to $\kappa' = 0$ of configurations at a given κ for different α values. Dashed lines refer to typical solutions, full curves to the barycenter of typical solutions, dotted-dashed to CE configurations.

6.2 Deeper models

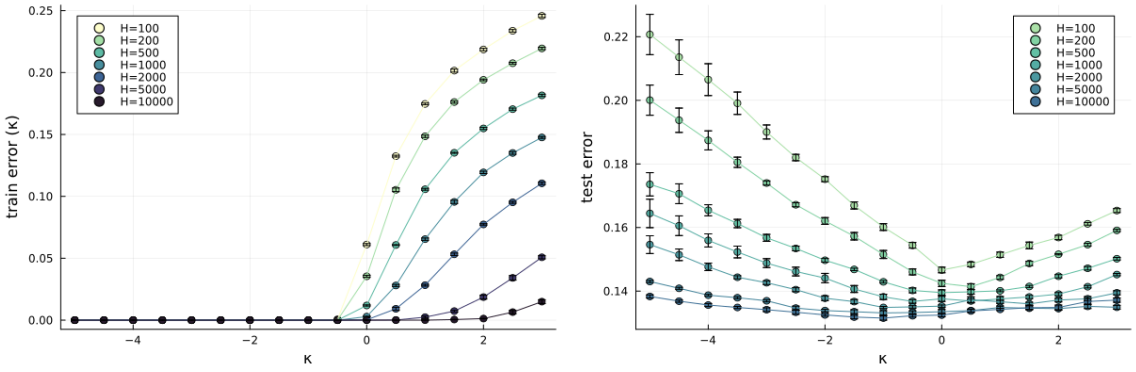


Figure 6.4: Train (*left*) and test (*right*) errors for one-hidden multi-layer Perceptron with different H hidden neurons trained with $P = 10$K patterns of Fashion-MNIST with ADAM and CE loss for 100 epochs. Averages over 5 independent samples. L2 regularization $\lambda = 10^{-4}$.

In deeper models, which can be studied numerically, it would be computationally expensive to obtain an estimate of a centroid of y solutions that approximates well

a high margin configurations. Consequently, one can just search for solutions using algorithms biased by construction towards robust regions, and can study the test error in a range of κ while increasing the overparametrization of the model. Notice that, since the notion of margin is ambiguous when referring to multi-layer architectures, we just use κ here in the loss definition and not as a threshold for intermediate layers.

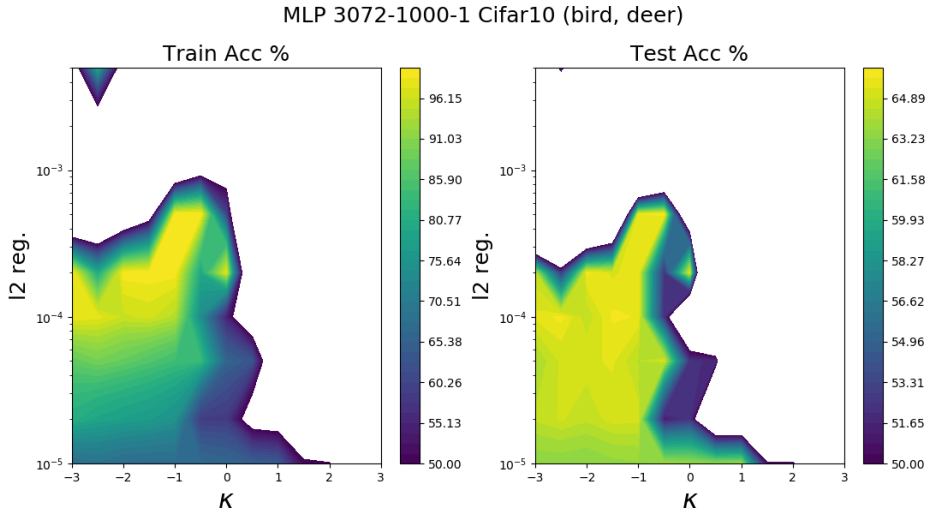


Figure 6.5: Heatmap of train (*left*) and test (*right*) accuracy of the L2 norm Lagrange multiplier versus the margin κ for a MLP with one hidden layer ($H = 1000$) trained on two classes (bird, deer) of CIFAR10 with CE-loss and SAM [9].

In Fig. 6.4 there are train and test errors for one hidden layer Perceptron with H hidden neurons, trained on Fashion-MNIST with an L2 regularization using SGD with ADAM. Increasing H means to increase accordingly the overparameterization. The purpose is again to generalize at $\kappa' = 0$ by sampling solutions with a given κ. There is no gain for the model if H is sufficiently small, since the optimal test error is precisely at $\kappa = 0$ and by increasing the margin $\kappa > 0$ leads to underfit on the training dataset (train error on the *left* of 6.4). Instead, increasing H has the effect of flattening the generalization both left and right with respect to $\kappa = 0$, as it is for 6.3. This happens until, for a certain value of H large, the curve develops a minimum at a negative κ, suggesting that really by relaxing the constraints and enlarging the landscape while trying to find flat solutions per se, could be a valid alternative to rethink the training of very wide models. In support of this, Fig. 6.5 shows the heatmap of L2 regularization versus the margin κ for the same model with $H = 1000$ trained on two classes of CIFAR10. Again, the optimal test accuracy evaluated at $\kappa' = 0$ is for some models sampled with non trivial values of negative κ and regularization degree.

6.3 Generalization error along the geodesic

To strengthen both insights regarding the fact that if one optimizes the norm of a model, it is possible *settle* for a less robust but equally performant configuration, or that without altering the norm, the averaging of inherently less robust models ultimately leads a robust and high-performing solution, one can study, with a formalism similar to Chapter 5, the generalization error along the geodesic of two configurations with negative $\kappa_{1,2}$ with respect to an instance of a problem at $\kappa_E = \kappa' = 0$ as in previous sections.

Let us start from the non-convex minimal model: the Perceptron of Chapter 5 with $\kappa_{1,2} < 0$. If, as mentioned, along the linear path between two solutions at $\kappa_{1,2} \leq \kappa_E$ one encounters atypical configurations that can or cannot be solutions according to the margin values, what is the generalization error on the interpolation? By studing the usual dataset of Gaussian inputs $\mathbf{x}^\mu \sim \mathcal{N}(0, \mathbb{I}_N)$ and labels given by a teacher $y^\mu = \text{sign}\left(\frac{\mathbf{w}^T\mathbf{x}^\mu}{\sqrt{N}}\right)$ which is itself a Perceptron, one can compute the generalization error of a student network defined as

$$\epsilon_g = \mathbb{E}_{\mathcal{D}}\Big\langle \Theta\Big(- y\frac{\gamma\mathbf{w}_1 + (1-\gamma)\mathbf{w}_2}{\sqrt{N}||\boldsymbol{w}||}\mathbf{x}\Big)\Big\rangle \tag{6.15}$$

where γ is the interpolation parameter which is $\gamma \in [0,1]$ as in Eq. (5.10), and the average is over the factorized measure $\langle . \rangle = \frac{1}{Z_1 Z_2}\int \prod_r d\mu(w_r)\, \prod_{\mu r}\Theta\Big(y^\mu \frac{\mathbf{W_r}\mathbf{x}^\mu}{\sqrt{N}} - k_r\Big)$. The order parameters consist, as in the usual setting after averaging over the disorder, on the overlap $q^{ab}_{rs} = \frac{1}{N}\sum_i w^a_r w^b_s$ between two of the virtual replicas $a, b \in \{n\}$ of the real solutions $r, s \in [y = 2]$ at the extremes of the path; plus, we have the alignment between teacher and student $m^a_r = \frac{1}{N}\sum_i w^T w^a_r$, being both the configurations normalized on the N-sphere. If, for an RS ansatz, one assumes (as in chapter 5) the overlap matrices structure of the kind

$$q^{ab}_{rr} = \delta_{ab} + q_r(1-\delta_{ab}) \quad \forall r = s \tag{6.16a}$$
$$q^{ab}_{rs} = R\,\delta_{ab} + p\,(1-\delta_{ab}) \quad \forall r \neq s \tag{6.16b}$$
$$m^a_r = m \tag{6.16c}$$

and then studies the saddle point equations in the $R \to p$ limit, it turns out it is possible to evaluate the generalization error at the obtained values, since Eq. (6.15) simplifies as

$$\epsilon_g = 2\int Dz D\lambda_2 \int_{A(\lambda_2,z)}^{\infty} D\lambda_1\, H\Big(\frac{\sum_r m_r\lambda_r}{\sqrt{1-p-\sum_r m_r^2}}\Big) \tag{6.17}$$

and with $A = \frac{\gamma\sqrt{p}z - (1-\gamma)(\sqrt{1-p}\lambda_2 - \sqrt{p}z)}{\gamma\sqrt{1-p}}$. It is straightforward to obtain the training error along the geodesic with respect to the instance at $\kappa_E = 0$, which is

$$\epsilon_t = \int \frac{du d\hat{u}}{2\pi}\prod_{ar}\frac{d\lambda^a_r d\hat{\lambda}^a_r}{2\pi}\prod_{ar}\Theta\Big(\text{sign}(u)\lambda^a_r - \kappa_r\Big)\Theta\Big(- u\frac{\gamma\lambda^1_1 + (1-\gamma)\lambda^1_2}{||W||}\Big) \tag{6.18}$$
$$\times\, e^{-\frac{1}{2}\sum_{ar,bs} q^{ab}_{rs}\hat{\lambda}^a_r\hat{\lambda}^b_s - \frac{\hat{u}^2}{2} - \sum_{ra}\hat{u}\hat{\lambda}^a_r m^a_r + i\sum_{ra}\lambda^a_r\hat{\lambda}^a_r + iu\hat{u}}$$

and leads with the same ansatz as (6.16) to

$$\epsilon_t = 2 \int Du \int Dx \int Dy_1 \frac{\Theta(u) \int_{\Gamma_1}^{\Gamma} D\lambda_1 \Theta\left(\frac{-(1-\gamma)\kappa_2 - \gamma\kappa_1}{\gamma}\right) \left[H(\Gamma_2) - H(\Delta_2)\right]}{H\left(\Gamma_1\right) H\left(\Gamma_2\right)} \tag{6.19}$$

being for completeness the definitions

$$\begin{aligned}
\Gamma &= \frac{\gamma(\sqrt{q_1 - m_1^2}\, x - m_1\, u) - (1-\gamma)\kappa_2}{\gamma\sqrt{1-q_1}} \\
\Gamma_1 &= \frac{\kappa_1 + \sqrt{q_1 - m_1^2}\, x - m_1\, u}{\sqrt{1-q_1}} \\
\Gamma_2 &= \frac{\kappa_2 + \frac{p - m_1 m_2}{\sqrt{q_1 - m_1^2}} x + \sqrt{\frac{(p-m_1m_2)(q_1 - p + m_1 m_2 - m_1^2) + (q_1 - m_1^2)(q_2 - p + m_1 m_2 - m_2^2)}{q_1 - m_1^2}}\, y_1 - m_2\, u}{\sqrt{1-q_2}} \\
\Delta_2 &= \Gamma_2 - \frac{\kappa_2}{\sqrt{1-q_2}} - \frac{\gamma(\sqrt{1-q_1}\,\lambda_1 - \sqrt{q_1 - m_1^2}\, x + m_1\, u)}{(1-\gamma)\sqrt{1-q_2}}.
\end{aligned} \tag{6.20}$$

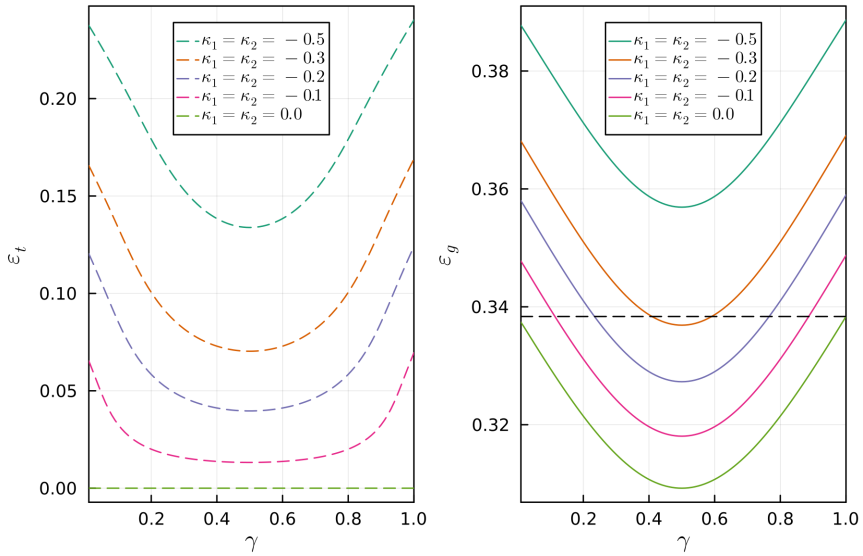


Figure 6.6: Training (*left*) and generalization (*right*) curves along the geodesic path connecting two solutions with margins $\kappa_1 = \kappa_2$ for $\alpha = 1$. The interpolation parameter is $\gamma \in [0, 1]$, and the predictions are derived using replica theory under the RS ansatz.

Looking at Fig. 6.6 something interesting happens: first, given that it is $\kappa' = \kappa_E = 0$, the training error (on the *left*) of solutions sampled with $\kappa_{1,2} = 0$ is zero along the entire path, while it is always greater than zero for each less robust solution for which it is $\kappa_{1,2} < 0$, increasing the more negative the margins are. It is useful to remember that the instance of the problem for $\kappa_E = 0$ is convex, and that is the reason why the *dashed-green* line is exactly zero for all the span of γ, as well as the reason for

which the minimum of ϵ_t is always at the barycenter of the path, given the hierarchical ordering of configurations with margins as described in Chapter 5. The generalization error behaves apparently the same way as ϵ_t, i.e. decreasing with $\kappa_{1,2}$ in an ordered fashion and, for any fixed values of $\kappa_{1,2}$, decreasing along the geodesic. However, if one compares the typical generalization error of a solution at the same margin of the problem (i.e. $\kappa = 0$, *dashed-black* line in ϵ_g), something non-trivial can be realized: even though negative-margin configurations are less robust, at energy greater tha zero, and underperforming per se, the generalization error encountered along the geodesic connecting them is better than that of a typical solution to the given problem, supporting all the results shown above about the efficiency of averaging methods. Probably, but this is a conjecture that has yet to be proved (we defer confirmation to the output of [128]), if one looked at the stability of the barycenter of negative solutions when they perform better than $\epsilon_g(\kappa = 0)$, one would understand that there are some violated patterns in the dataset, but that the mode of the distributions would be at an effective margin $\kappa_{eff} > 0$.

Part III

Neuroscience comes into play

Chapter 7

Modeling dendritic non-linearities with a TCM

The question on how individual neurons make computational signal processing is a key issue in neuroscience. The simplest scenario is that dendrities simply add together synaptic inputs in a linear way, effectively propagating a dot product of a vector of pre-synaptic activity with a vector of synaptic weights to the axon initial segment, where a thresholding operation is then applied to decide whether the neuron emits an action potential or not. From that perspective, neurons are analogous to simple Perceptrons, whose memory-related functions have been studied extensively [129, 130]. In this view, if one looks at both the statistics of the connectivity matrix of a simple Perceptron and the experimentally recorded distribution of weights in a pyramidal neocortex cell, it is natural to realize that for both the models (its modeling and the actual one), a relevant fraction of silent synapses is present when trying to maximize the information storage. This suggests that synaptic connectivity in cortex is optimized to store a large number of activity patterns in a robust fashion [130]. Even though there is a debate concerning the actual shape of the stored information, whether in the form of simple attractors that represents prototype of internal representations of stimuli, or in the form of sequences of activity, it is reasonable to agree that having high levels of silent synapses is beneficial to store patterns as fixed point attractors of the neural dynamics.

Some more recent experimental results regarding the integrative properties of pyramidal cells [131], emphasize how, instead of a global linear or sublinear summation of a stimuli, neurons enact a kind of nonlinear integration due to the presence of dendrites. It turns out that, regarding the spatial arrangement of dendrites, nearby inputs on the same branch are summed sigmoidally, while widely separated inputs (i.e. inputs to different branches) are summed linearly. One can intend these non-linearities from a biophysical perspective [132], due to various types of voltage-gated ionic currents, such as NMDA receptor mediated synaptic currents [133], calcium [134], or sodium currents [135]. These non-linearities could potentially transform neurons into devices that are analogous to multi-layer perceptrons [131, 136], or cascades of linear-nonlinear processes [137, 138]. For all the above, tree-like non-linear morphologic structures of dendrities combined with a second layer of connectivity which sends the signal to the central body of the cell, appears to be a good model to inspect

about computational capabilities of single neurons. Given that multi-layer networks are known to have more expressivity than single layer ones, what degree of dendritic non-linearities enhance the computational capabilities of neurons? Previous studies on tree committee machines from the machine learning/statistical physics community typically do not take into account constraints that are present in neurons, such as sign constraints on synaptic weights, and the specific dendritic non-linearities that are observed experimentally.

In the present Chapter, we set out to study a single neuron model with non-linear dendrites implementing experimentally observed non-linearities, and sign-constrained synapses. We will show how, despite the capacity of the model remains finite when the size of the hidden layer diverges, it can be increased arbitrarily depending on the shape of the non-linearity. In all cases, by playing with the thresholds for the firing rates, the non-linearity in exam increases the storage capacity wrt single one layer neurons. We will also confirm that the model is able to reproduce the experimental distribution of synaptic weights and that, when testing the model capabilities numerically on real datasets, the non-linearity not only tends to enhance the algorithmic capacity and to increase the training speed of the network with respect to one-layer models, but also it allows to find solutions that are more robust to input and synaptic perturbations and generalize better. Results are preliminary results of a work in preparation [139].

7.1 The single neuron model

Following a series of works [129, 130, 140], we focus on the study of excitatory neurons, while incorporating the action of inhibitory ones with an activation threshold, which is the value that the processed synaptic inputs via the dendritic branches has to overcome in order to fire an action potential. We do so since the inhibitory contribution is more stereotyped, given that its primary role is to regulate and balance the activity of the excitatory part, preventing over-excitation and maintaining the network stability. This relies on the choice of positive synaptic weights.

The neuron activity state, signifying the firing of an action potential, is represented here in binary labeled responses: $'1'$ signifies the neuron is in an *active* state, while $'0'$ indicates the *inactive* state. In practice, when the neuron receives some inputs from a collection of other excitatory neurons responses, that is modeled as a N-dimensional activity pattern (comprising of binary contributions of 0/1) with elements $\xi_i \in \{0, 1\}$, there is an associated *desider state* that it has to match ($\sigma \in \{0, 1\}$) in order to process the signal according to the correct prototyped activity. In the input pattern, each element identified by the index $i = 1, \dots, N$ reflects the state of the corresponding afferent neuron and the proportion of $'1'$ in these patterns is given by the input coding level f_{in}, while the count of $'1'$ in the labels is known as the output coding level f_{out}.

The weight of the excitatory synaptic connection with neuron i is denoted by $w_i \geq 0$. The neuron can learn a number $P = \alpha N$ of activity pattern and label pairs, $\{\xi^\mu, \sigma^\mu\}_{\mu=1}^{P}$, as from the machine learning perspective, by iteratively adjusting the synaptic weights to link a specific activity pattern of afferent neurons with a corresponding activity state for the model itself. Here, as mentioned, the purpose

is to study how dendritic non-linearities affect the memory capability of the model. To choose the actual transfer function for the dendritic integration, we refer to the experimentally measured data summation as in [131] for a rat neocortical pyramidal neuron. In [131], the authors made comparison of within-branch and between-branch summations by positioning two electrodes to be activated in either the two positions, then by keeping track of the resulting voltage traces. For the first, by positioning the electrodes near the same dendritic branch, the summation turns out to be sigmoidally modulated. Between-branch, instead, the summation is linear (Fig. 7.1).

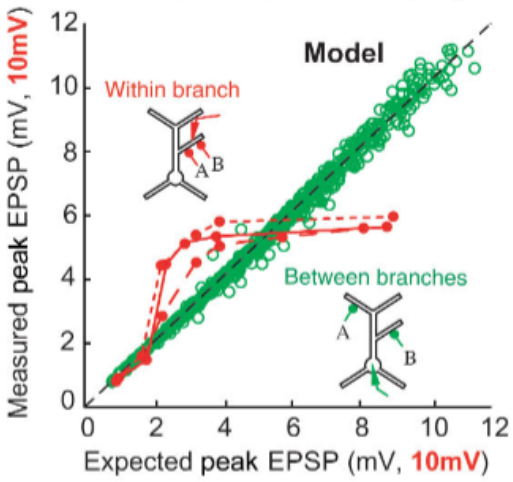


Figure 7.1: Modeling data: summation of single-pulse in the apical oblique dendrites of a CA1 pyramidal cell model shows a sigmoidally modulated within-branch summation (red circles) and linear between-branch summation (open green circles). Reprinted from [131].

Taking this evidence as a starting point, let us define the model precisely. It has K dendritic branches and a single output corresponding to its activity state. The mapping from the inputs $\vec{\xi}^{\mu} \in \{0,1\}^N$ to the hidden layer is realized by the K dendritic branches, each having N/K positive weights $w_{li} \geq 0$ (first layer), and then by summing them with a second layer. As we focus on excitatory connectivity matrices, the thresholds θ_d and θ_s, respectively acting on the dendritic branches and on the neuron soma, will stand for the inhibitory activity. For simplicity the threshold θ_d is equal for every dendritic branch. The output σ^{μ} of the neuron in correspondence with the input pattern $\vec{\xi}^{\mu}$ is obtained via the linear function that sums the contributions of the dendritic branches

$$\sigma^{\mu}_{\text{out}} = \Theta\left(\Delta^{\mu}\right) = \Theta\left(\frac{1}{\sqrt{K}}\left(\sum_{l=1}^{K} c_l\, g(\lambda^{\mu}_l) - K\theta_s\right)\right) \tag{7.1}$$

where λ^{μ}_l is the input of the l-th dendritic branch, i.e. $\lambda^{\mu}_l = \left(\sqrt{\frac{K}{N}}\sum_{i=1}^{N/K} w_{li}\xi^{\mu}_{li} - \sqrt{\frac{N}{K}}\theta_d\right)$, while w_{li} is the weight connecting the input entry i in dendritic branch l and c_l is the weight connecting the dendritic branch l to the output ($c_l = 1\ \forall l$); $g(x)$ stands for

the dendritic contribution to the signal processing. It is defined as

$$g(x) = \begin{cases} \max(0, x) & x < x_{\min} \\ 2(1 - x_{\min})\sigma\left(\gamma(x - x_{\min})\right) - 1 + 2x_{\min} & x \geq x_{\min} \end{cases} \tag{7.2}$$

where $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function. We will refer to the transfer function from now on with the name of *Polsky*, that is represented in the inset of Fig. 7.2 to match the one in Fig. 7.1. For all the following investigations we set $x_{\min} = 0.33$ and $\gamma = 15$ in order to quantitatively reproduce Fig. 7.1. The total transfer function of the neuron is then finally obtained by applying a Heaviside function on the output node (Fig. 7.2).

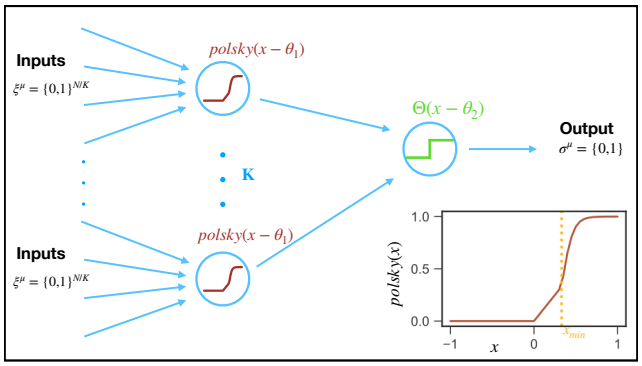


Figure 7.2: Representation of the single non-linear neuron model. The lower-left inset shows the functional form of the dendritic non-linearities à la Polsky; it consists of a combination of a ReLU activation and a sigmoid activation.

For analytical and numerical reasons, throughout the discussion we will focus on the large N and K limit $N, K \to \infty$ with $\frac{K}{N} \to 0$. Taking the synaptic weights of order 1 ($w_{li} \sim \mathcal{O}(1)$), the preactivations to the hidden neurons are of order $\sum_i w_{li}\xi_{li}^\mu \sim \mathcal{O}(\frac{N}{K})$. We therefore take $\theta_d \sim \mathcal{O}(1)$ and rescale it by $\sqrt{\frac{N}{K}}$ in Eq. (7.1). The same kind of reasoning applies to the second threshold, that if taken of order 1 has to be rescaled by $\frac{1}{\sqrt{K}}$. The choices are useful as they give finite preactivations variance (respectively on the dendritic branch and on the output) in the large N, K limits.

7.2 Memory-related properties

7.2.1 Analytical estimation of the critical capacity

Since the model we have defined is equivalent to a TCM with non-overlapping and positive receptives fields, and the task of storing i.i.d. binary patterns $\boldsymbol{\xi}^\mu \in \{0,1\}^N$ and labels $\sigma^\mu \in \{0,1\}$ is the usual supervised learning task, whose performance we have characterized so far in the typical case scenario using the replica method, we can do the same here to compute the maximum memory capacity of the model. Given

the training set $\{\boldsymbol{\xi}^\mu, \sigma^\mu\}_{\mu=1}^{P=\alpha N}$, the probability distribution of each component of a pattern is given by

$$P(\xi_{li}^\mu) = f_{\text{in}}\,\delta(\xi_{li}^\mu - 1) + (1 - f_{\text{in}})\,\delta(\xi_{li}^\mu) \tag{7.3}$$

being f_{in}, as mentioned, the input coding level. We consider a probability distribution of labels to be equal in form to (7.3) but we allow the possibility to have different coding level in the output f_{out}. In order to study the volume of synaptic weights that correctly solve the task, we resort to the averaged replicated partition function once fixed the measure over the synpatic weights

$$\int d\mu(w)\,\bullet \equiv \int_0^\infty \prod_{li} dw_{li}\,\bullet\,. \tag{7.4}$$

As usual, by defining appropriate order parameters, it is possible to conveniently study the problem in the large-N limit. Here we have

$$\sum_i w_{li}^a = \frac{N}{K}\overline{w} + \sqrt{\frac{N}{K}} M_l^a \tag{7.5a}$$

$$q_l^{ab} \equiv \frac{K}{N}\sum_i w_{li}^a w_{li}^b \tag{7.5b}$$

$$Q_l^a \equiv \frac{K}{N}\sum_i (w_{li}^a)^2 \tag{7.5c}$$

where Eq. (7.5a) represents the average synaptic weight that we express in two contributions: the first is an averaged *scaled* synaptic weight $\overline{w} = \frac{\theta_d}{f}$ justified by the fact that for each sub-perceptron of the first layer only $\frac{N}{K} f_{\text{in}}$ synapses contribute; the second, instead, is a $\sqrt{\frac{N}{K}}$ correction that is needed in order to fine tune the average synaptic weight relatively to the threshold θ_d. Notice also that, a part from the usual overlap q_l^{ab} of two replicas wrt the same branch l, we have here the averaged squared norm Q_l^a of a synaptic weight belonging to l. Enforcing these definitions and applying an RS ansatz, we are able to express the Gibbs free entropy in the large N, P limits. It is

$$\phi_{RS} = \frac{q\,\hat{q}}{2} - Q\,\hat{Q} - \hat{M}\frac{\theta_d}{f} + \mathcal{G}_S + \alpha\,\mathcal{G}_E \tag{7.6a}$$

$$\mathcal{G}_S = \ln\sqrt{\frac{2\pi}{\hat{q} - 2\hat{Q}}} + \frac{1}{2}\left(\frac{\hat{M}^2 + \hat{q}}{\hat{q} - 2\hat{Q}}\right) + \int Dz\,\ln H\left(-\frac{\hat{M} + \sqrt{\hat{q}}z}{\sqrt{\hat{q} - 2\hat{Q}}}\right) \tag{7.6b}$$

$$\mathcal{G}_E = \mathop{\mathbb{E}}_{\sigma} \int \prod_l Dt_l\,\ln\left[\int \prod_l D\lambda_l\,\Theta\left(\frac{\sigma}{\sqrt{K}}\left(\sum_l c_l\, g\left(\sqrt{f_{\text{in}}(1 - f_{\text{in}})(Q - q)}\lambda_l + a_l\right) - K\theta_s\right)\right)\right] \tag{7.6c}$$

being for convenience

$$a_l \equiv f_{\text{in}} M + \sqrt{f_{\text{in}}(1 - f_{\text{in}})q}\,t_l. \tag{7.7}$$

Large K limit

We focus on the $K \to \infty$ limit as we did in 3.2.1 ($\frac{K}{N} \to 0$), both because it is *biologically* realistic: the number of dendritic branches in neurons is typically large (in some cases even more than a hundred) and the number of synapses in each branch is typically large as well (ranging in the 10,000s), and also because the treatment simplifies thanks to CLT and Gaussian activations upstream to each dendritic branch. It is the energetic term (7.6c) that needs manipulations in the limit. Indeed, if we consider the square bracket contribution, by expanding the neuron activation up to the second order, it turns out we can integrate with respect to all the λ_l variables term by term and

$$\begin{aligned} I &= \int \frac{dh\, d\hat{h}}{2\pi} e^{-ih\hat{h}}\, \Theta\left[\sigma\left(h - \sqrt{K}\theta_o\right)\right] \int \prod_l D\lambda_l\, e^{\frac{i\hat{h}}{\sqrt{K}} \sum_l c_l\, g\left(a_l + \sqrt{f_{\text{in}}(1-f_{\text{in}})(Q-q)}\,\lambda_l\right)} \\ &\simeq \int Dh\, \Theta\left(\sigma(M^{(0)} + \sqrt{\Delta^{(0)}}h - \sqrt{K}\theta_s)\right) \end{aligned} \tag{7.8}$$

where we have introduced the variables:

$$M^{(0)} = \frac{1}{\sqrt{K}} \sum_l c_l \langle g \rangle_\lambda \tag{7.9a}$$

$$D^{(0)} = \frac{1}{K} \sum_l c_l^2 \left[\, \langle g^2 \rangle_\lambda - \langle g \rangle_\lambda^2 \,\right] \tag{7.9b}$$

and the notation

$$\langle g \rangle_\lambda = \int D\lambda\, g\left(f_{\text{in}} M + \sqrt{f_{\text{in}}(1-f_{\text{in}})q}\, t_l + \sqrt{f_{\text{in}}(1-f_{\text{in}})(Q-q)}\, \lambda\right). \tag{7.10}$$

One can use iteratively CLT for the K integrals over the variable t_l, getting

$$\mathcal{G}_E = \mathop{\mathbb{E}}_{\sigma} \int Dt \ln \int D\lambda\, \Theta\left[\sigma\left(M_0 + \sqrt{D_0}\, t + + \sqrt{D_1}\, \lambda - \sqrt{K}\theta_s\right)\right] \tag{7.11}$$

where now

$$M_0 = \frac{1}{\sqrt{K}} \sum_l c_l \langle\, \langle g \rangle_\lambda \rangle_t = m_c \langle\, \langle g \rangle_\lambda \rangle_t \tag{7.12a}$$

$$D_0 = \frac{1}{K} \sum_l c_l^2 \left[\langle\, \langle g \rangle_\lambda^2 \rangle_t - \langle\, \langle g \rangle_\lambda \rangle_t^2 \right] = w_c \left[\langle\, \langle g \rangle_\lambda^2 \,\rangle_t - \langle\, \langle g \rangle_\lambda \rangle_t^2 \right] \tag{7.12b}$$

$$D_1 = w_c \left[\langle\, \langle g^2 \rangle_\lambda \rangle_t - \langle\, \langle g \rangle_\lambda^2 \rangle_t \right]. \tag{7.12c}$$

Using the definition of the committee machine in which all weights in the second layer are s.t. $c_l = 1$, we have $m_c = \sqrt{K}$ and $w_c = 1$. In order to get a well-defined large K limit we have to impose (analogously to what we have done on the dendritic threshold θ_d) that the divergence induced by θ_s cancels with the one coming from M_0. We therefore impose that M scales, in the large K limit, as: $M = \overline{M} + \frac{\delta M}{\sqrt{K}}$.

Enforcing the scaling in (7.12a), and getting read of one Gaussian integral by making a rotation between variables, it turns out that $\overline{M}$ is fixed by the implicit relation

$$\theta_o = \int Dy\, g\left(\sqrt{f_{\text{in}}(1-f_{\text{in}})Q}\, y + f_{\text{in}}\overline{M}\right) \tag{7.13}$$

that involves the output threshold. Hence, the energetic term (7.11) becomes

$$\begin{aligned}\mathcal{G}_E =& \mathop{\mathbb{E}}_{\sigma} \int Dz\, \ln H\left(\frac{\sqrt{D_0}\, z - \sigma\Delta}{\sqrt{D_1}}\right) = \int Dt\, \left[f_{\text{out}} H\left(\frac{\sqrt{D_0}\, z - \Delta}{\sqrt{D_1}}\right) + \right. \\ &\left. + (1-f_{\text{out}}) H\left(\frac{\sqrt{D_0}\, z + \Delta}{\sqrt{D_1}}\right)\right]\end{aligned} \tag{7.14}$$

where the order parameters, strictly dependent on the choice of the activation function $g(x)$ (which we leave generic for now for completeness), are simplified as

$$\begin{aligned}\Delta &\equiv f_{\text{in}} M \int Dx\, g'\left(\sqrt{f_{\text{in}}(1-f_{\text{in}})Q}\, x + f_{\text{in}}\overline{M}\right) \\ D_0 &= \Delta_q - \Delta_0 \\ D_1 &= \Delta_Q - \Delta_q\end{aligned}$$

renaming δM by M with a slight abuse of notation. The effective order parameters are now, as in 3.2.1

$$\Delta_q = \int Dx \left[\int Dy\, g\left(\sqrt{f_{\text{in}}(1-f_{\text{in}})q}\, x + \sqrt{f_{\text{in}}(1-f_{\text{in}})(Q-q)}\, y + f_{\text{in}}\overline{M}\right)\right]^2 \tag{7.16}$$

Δ_Q and Δ_0, being obtained by simply substituting in the previous expression $q \to Q$ and $q \to 0$ respectively. Notice that the whole quenched entropy is in the end perfectly equivalent to the one found for the positive perceptron model as studied in [129, 130], but where the order parameters q and Q are replaced by $\Delta_q - \Delta_0$ and $\Delta_Q - \Delta_0$ respectively.

It is possible to compute the maximum storage capacity α_c, which is the maximum density of input patterns the neuron is capable to classify, by sending $q = Q - dq$ for $dq \to 0$, that is the regime in which the set of possible synaptic weights shrinks toward a single point. Scaling the order parameters accordingly (details of the limit can be found in C.2)

$$\begin{aligned}\hat{q} &\simeq \frac{C}{(Q-q)^2} \\ \hat{q} - 2\hat{Q} &\simeq \frac{A}{Q-q} \\ \hat{M} &\simeq \frac{B\sqrt{C}}{Q-q}\end{aligned}$$

and expanding when necessary, the saddle point equation for the critical capacity α_c reads in the end

$$\alpha_c = \frac{2\Gamma_1 H(B)}{\frac{d\Gamma_0}{dQ} - \frac{\Gamma_0}{\Gamma_1}\frac{d\Gamma_1}{dQ}}. \tag{7.18}$$

We expect that, for fixed coding levels f_{in} and f_{out}, playing with the threshold values θ_d and θ_o changes the scale of the activation $g(x)$ with respect to the pre-activations. Let us restrict to the Polsky case. In the limit $\frac{K}{N} \to 0$, the pre-activations before the dendritic integration are Gaussian, with mean μ and variance σ^2 that will depend on θ_d, f_{in} and on the weight distribution (uniform for simplicity) with mean $\bar{w} = \frac{\theta_d}{f_{\text{in}}}$ by construction. From simple arguments, one can derive the standard deviation of the distribution

$$\sigma = \sqrt{\mathbb{E}_{\boldsymbol{\xi}} \left(\sqrt{\frac{K}{N}} \sum_{i=1}^{N/K} W_{li} \xi_{li} \right)^2 - \left(\sqrt{\frac{K}{N}} \mathbb{E}_{\boldsymbol{\xi}} \sum_{i=1}^{N/K} W_{li} \xi_{li} \right)^2} = \sqrt{f_{\text{in}}(1 - f_{\text{in}}) \left(Q - M^2\right)} \tag{7.19}$$

which is independent of l since each dendritic branch has access to an independent portion of the input. In the $\alpha \to 0$ limit the saddle point equations in C.2 can be solved exactly, and the variance of the preactivation can be further simplified. We therefore get the standard deviation of the dendritic preactivation depends linearly on the dendritic inhibition threshold

$$\sigma = \theta_d \sqrt{\frac{1 - f_{\text{in}}}{f_{\text{in}}}}. \tag{7.20}$$

It is easy to check that this linear relation is satisfied even at finite α, or considering a different distribution over the weights at initialization.

As shown in Fig 7.3 and from Eq. (7.20), decreasing the first threshold θ_d while keeping $f_{\text{in}}, f_{\text{out}} = 0.5$ fixed, has the effect of peaking the variance of the pre-activations in the range for which the Polsky transfer function is linear. Therefore, the corresponding critical capacity α_c when $\theta_d \to 0$ (*red* curve) tends towards the critical capacity of a TCM with $g(x) \equiv x$, which is that of the positive Perceptron (*dashed-black* curve) [129, 130] (here α_c no longer depends of the first threshold θ_d, but just to the second one θ_s). On the opposite, if one increases the variance by increasing the threshold θ_d, the pre-activations move towards the active range of Polsky, where the non-linearity is more pronounced. Indeed, the critical capacity is an increasing function of θ_d, and precisely it diverges with the number of hidden neurons (i.e. $\alpha_c \propto \sqrt{\ln K}$) when $g(x) \to \Theta(x)$, as it is for the standard case with weights positive and negative [129, 130, 140] when considering an activation function that is not weakly differentiable [47, 97]. So, in theory, the memory capacity of this simple model of neuron is greater than the much smaller capacity of its linear counterpart when exploiting the non-linearity effect. For completeness, we also report the $g(x) \equiv \text{ReLU}(x)$ case in Fig 7.3, demonstrating that even for that the critical capacity can grow a lot, but more slowly than in the Polsky case. As mentioned, the estimation of the critical capacity here is based on the RS ansatz; since the model we are analyzing is non-convex, that ansatz is thought to be only an upper bound to the true result. In order to get more precise results on the critical capacity, one should resort to RSB effects [97, 123, 141].

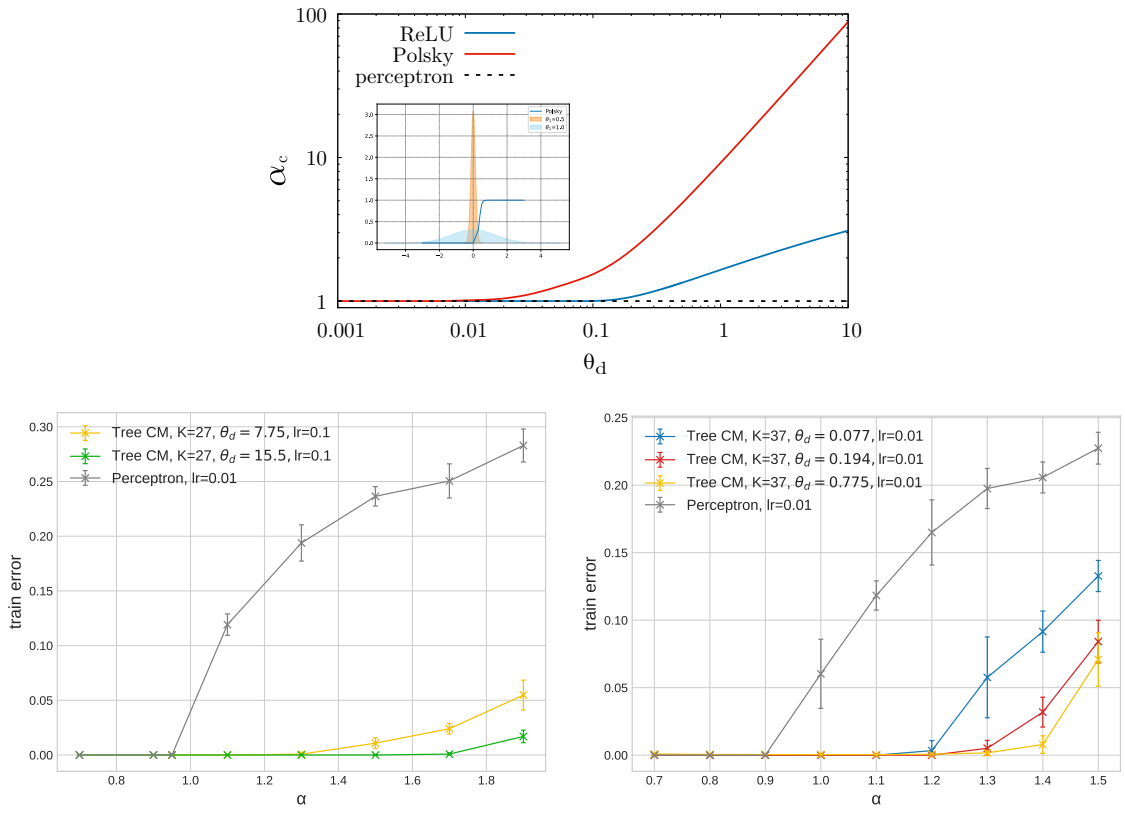


Figure 7.3: (*Top panel*) Pre-activation distributions of a given hidden neuron for different threshold values θ_d compared with the transfer function scale. *Inset*: critical capacities $\alpha_d(\theta_d)$ for different activation functions. The "identity" refers to the Perceptron case. (*Bottom panels*) Train errors for the TCM with fixed K hidden neurons using SGD (*left*-part) or LAL (*right*-part). Grey curves refer to the Perceptron case.

7.2.2 Algorithmic capacity, learning speed and noise robustness

Having a clue of the theoretical advantages of taking a non-linear two-layer NN as an activation scheme for a neuron, we can inspect and elaborate about what is the computational memory capacity in practice, or what are the advantages of a non-linearity for the speed of learning or robustness, by simulating the activity of this model. In a nutshell, let us derive some preliminary conclusions when taking into account algorithms inspired by machine learning practice: SGD and least-action learning (LAL). For both, since the weights are constrained to be positive, if they become negative during the training they are instantaneously reset to zero. For the first, SGD requires a differentiable objective to minimize, so we choose the cross-entropy (CE) loss

$$\mathcal{L}_{\rm CE}(w) = \sum_{\mu=1}^{P} f_{\gamma_{ce}}\left(\sigma_\mu \sigma^\mu_{\rm out}(w)\right) \tag{7.21}$$

where $f_{\gamma_{ce}}(x) = -\frac{x}{2} + \frac{1}{2\gamma_{ce}} \log\left(2\cosh(\gamma_{ce} x)\right)$ and the output pre-activation $\sigma^{\mu}_{\text{out}}$ is $o^{\mu} = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i \xi_i^{\mu}$ in the case of the perceptron and Eq. (7.1) in the case of the committee machine. The dynamics implemented is standard, a part for the sign constraints of the weights, and so it does not need clarifications. The second algorithm instead relies on a natural extension of the PA algorithm for two-layer NN. It goes as in Algorithm 1, where for each sample pattern, if the network gives an output that is different from the correct one, the prevision and so the error is fixed by trying to adjust the weights corresponding to the hidden unit that is the closest to the threshold θ_d; this is the hidden unit in the committee that makes the fewest mistakes. Some modifications of this algorithm are possible to enhance the biological plausibility, such as for instance randomly choosing a subset of dendritic branches to update. However, the 01-LAL algorithm is the one that empirically shows the highest capacity when choosing $g(x) = Polsky(x)$. In Fig. 7.3, the two bottom subplots refer to both the algorithmic capacities of a Perceptron and the TCM when trained with either of the algorithms: LAL on the left, SGD on the right.

Algorithm 1 01-LAL

for $t = 1, 2, \ldots$ **do**
  $\xi^{\mu}, \sigma^{\mu} \leftarrow$ sample pattern
  $\sigma^{\mu}_{\text{out}}(w, \xi^{\mu}; \theta_d, \theta_s) = \Theta$ (Eq.7.1)
  if $\sigma^{\mu}_{\text{out}}(w, \xi^{\mu}; \theta_d, \theta_s) \neq \sigma^{\mu}$ **then**
    $\eta^{\mu} = 2\sigma^{\mu} - 1$
    $\phi_k = \left\{\eta^{\mu}\left(\sum_i w_{ik}\xi^{\mu}_{ik} - \frac{N}{K}\theta_d\right) \text{ for } k \in 1, \ldots, K\right\}$
    $k_{\star} = \text{argmax}(\phi_k : \phi_k < 0)$
    $w_{ik_{\star}} \leftarrow w_{ik_{\star}} + \lambda \eta^{\mu} \xi^{\mu}_{ik_{\star}}$
    if $w_i < 0$ **then**
      $w_i \leftarrow 0$
    end if
  end if
end for

For the Perceptron case, the optimization problem is convex and so with a little effort one can make the empirical capacity to get very close to the critical capacity $\alpha_c \sim 1$, since the phase space is by definition a unique connected component and no effects of RSB are coming into play. For the committee, instead, the case is more involved. The problem is non convex due to the non-linearity and indeed, even if the algorithmic capacity benefits of the Polsky transfer function, for both LAL and SGD, α_{alg} for different ranges of thresholds is far from what one would expect theoretically. The highest values of α_{alg} is reached in both cases when the variance of the pre-activations makes the transfer function like an Heaviside function, and despite reaching a value $\alpha_{\text{alg}} > 1$ (precisely $\alpha_{\text{alg}} \sim 1.7$ for LAL and $\alpha_{\text{alg}} \sim 1.38$ for SGD with the best threshold choices), it is still very far away from α_c. Here, RSB effects and the corresponding changes in the geometrical structure of minimizers, are making the problem hard in some ranges of density and there is the usual gap between algorithmic convergence and the SAT/UNSAT threshold that we have discussed in detail. To be

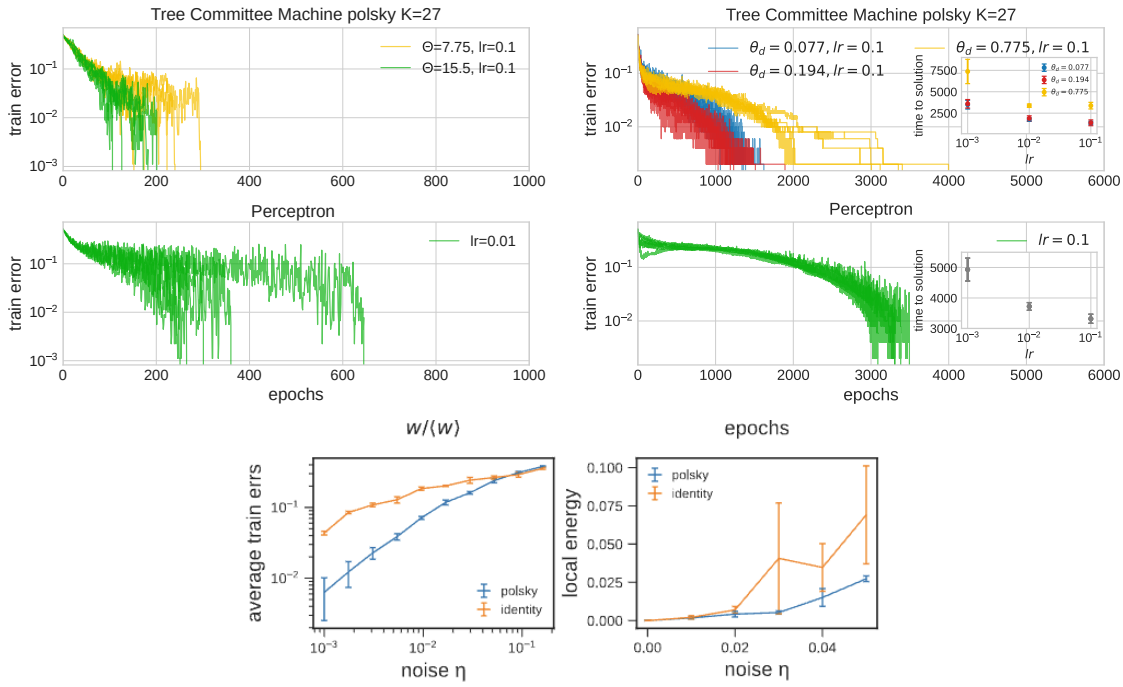


Figure 7.4: (*Top panels*) Speed of learning as a function of the epochs for the TCM compared with the Perceptron. Left-top subplots refer to LAL; right-top to SGD. Hyper-parameters are specified in the legends. (*Bottom panels*) Average training errors and local energy profiles as a function of multiplicative Gaussian noise on the weights for the TCM with Polsky (orange lines) or the Perceptron case (blue lines).

fair, in fact, the comparison between theory and experiments should be made once the actual memory threshold has been calculated, compared with computational hardness (chapters 3, 2) (we leave that for a later work). Therefore, we can say that, although the computational advantage of having non-smooth dendritic non-linearities is a fact, the memory capacity of the model remains finite even when K the hidden neurons numbers is large.

In any case, there is still something we can wonder about: what is the learning speeds when using a non-linear integration of the spike from the outside? Is the robustness to corruption of the prototype patterns better in the non-linear case? To answer these questions we can make comparisons between the TCM (denoted with *polsky* in figures 7.4) and the Perceptron (denoted with *identity* in figures 7.4) cases as we did until now. In Fig. 7.4 (*top* subplots) are reported the learning speed of the two models when trained with the same number of i.i.d. patterns with the two algorithms in exam. The TCM is always faster to learn the dataset, and in particular for LAL, learning is faster the more non-smooth the signal is (note that by construction, the data being unstructured per se and the task simple, the algorithm does not benefit in learning with more informative but less sharp signals and indeed it performs better with the transfer function becoming effectively binary). For SGD in contrast, the learning speed seems better for a non-trivial threshold value, so the pre-activations are in the active range of the Polsky and the gradient is more expressive. Of course,

in this case, the number of epochs necessary to reach zero training error is governed not only by the learning rate λ (*inset* of top right Fig. 7.4) but also by the cross-entropy loss parameter $\gamma_{\rm ce}$ that we keep fixed for simplicity for all the experiments. Being more manageable SGD, we also inspect the error robustness to a multiplicative Gaussian noise perturbation of the patterns, which is nothing that a different way to measure, at the end of the training procedure, the flatness of the minimizers in which the model ended up. In the *bottom* of Fig. 7.4 we report a result that is not trivial since the linear model is convex while the committee is not: the minima found by the nonlinear model are flatter, with a larger height of the basins of attraction, and therefore more stable to noise.

7.3 Distribution of synaptic weights

The distribution of synaptic weights is an observable that is considered to be informative of the way learning takes place in the brain. Interestingly, it can also be measurable in experiments. In [129], the theoretical distribution of the synaptic weights was computed for the one layer neuron model and compared with experiments with a fitting procedure, finding a good agreement. In particular, it was found that the distribution of synaptic weights of Purkinje cells strongly resemble the one of perceptrons learning random input-output associations in the critical capacity regime. In this regime the distribution of synaptic weights possess a large (more than 50%) fraction of vanishing or "silent" synapses corresponding to a delta function at zero weight, and it is a truncated Gaussian at positive weights. The fact that the connectivity matrix of weights is in general sparse strongly depends on the degree of robustness of the classification. Indeed, for the reliability of the classification implemented by the neuron model, one can impose a minimal size κ for the basins of attraction corresponding to the patterns in the training set, and this turns out to be equivalent of requiring

$$\Delta^{\mu}(\boldsymbol{w};\theta_d,\theta_s) > \kappa, \quad \forall \mu \in [P] \tag{7.22}$$

being the stability Δ^{μ} that in Eq. (7.1) with a reliability parameter (or margin) $\kappa > 0$. Hence, the larger the robustness imposed for the classification, the sparser will be the connectivity matrix of weights. As reported in [130], electrophysiological studies on pyramidal cells in neocortex, have consistently found a value for this connection probability of order 10% CITA, thus consistent with the theoretical results on the perceptron.

Here, we set out to study how the shape of the probability distribution of the weights changes in our non-linear neuron model. It is defined with respect to the weight of the first sub-perceptron W_{11} for simplicity as

$$P(W) = \left\langle \left\langle \frac{1}{Z} \int_0^{\infty} \prod_{li} dW_{li} \prod_{\mu=1}^{\alpha N} \Theta \left[\frac{\sigma^{\mu}}{\sqrt{K}} \left(\sum_{l=1}^{K} c_l \, g\left(\lambda_l\right) - K\theta s \right) - \kappa \right] \delta(W - W_{11}) \right\rangle \right\rangle_{\{\xi^{\mu},\sigma^{\mu}\}} \tag{7.23}$$

being Z the partition function of the solutions with margin κ. Here, applying the replica trick is standard as for the typical volume of solution computed in 7.2.1. Indeed, the entropic and energetic contributions to the free energy are the same as 7.6, and in the $n \to 0$ limit the computation simplifies as

$$P(W) = \lim_{n\to 0} \int_0^\infty \prod_{l,a} dW_l^a \, \delta(W - W_1) \, e^{\sum\limits_{a<b,l} \hat{q}_l^{ab} W_l^a W_l^b + \sum\limits_{a,l} \hat{Q}_l^a (W_l^a)^2 + \sum\limits_{a,l} \hat{M}_l^a W_l^a} \tag{7.24}$$

provided the order parameters satisfy the same saddle point equations as before. Under the RS ansatz the expression becomes

$$\begin{aligned} P(W) &= \Theta(W) \int Dz \, \frac{e^{-\frac{1}{2}(\hat{q}-2\hat{Q})W^2 + (\sqrt{\hat{q}}z + \hat{M})W}}{\int_0^\infty dW \, e^{-\frac{1}{2}(\hat{q}-2\hat{Q})W^2 + (\sqrt{\hat{q}}z + \hat{M})W}} \\ &= \Theta(W) \sqrt{\hat{q} - 2\hat{Q}} \, e^{-\frac{1}{2}(\hat{q}-2\hat{Q})W^2 + \hat{M}W} \int Dz \, e^{\sqrt{\hat{q}}Wz} \, \frac{G\left(-\frac{\sqrt{\hat{q}}z + \hat{M}}{\sqrt{\hat{q}-2\hat{Q}}}\right)}{H\left(-\frac{\sqrt{\hat{q}}z + \hat{M}}{\sqrt{\hat{q}-2\hat{Q}}}\right)}. \end{aligned} \tag{7.25}$$

Notice that the dependence of $P(W)$ on K and on the activation function is not explicit, but is concealed inside the order parameters that clearly depend on them through the saddle point they have to satisfy. Also, when $\alpha = 0$ the synaptic weight satisfies an exponential distribution

$$P(W) = \Theta(W) \frac{f_{\text{in}}}{\theta_d} e^{-\frac{f_{\text{in}}}{\theta_d} W} \tag{7.26}$$

since the model has fixed average of weights $\overline{W} = \frac{\theta_d}{f_{\text{in}}}$ by construction but not fixed norm. When one increases the number of constraints, for a fixed value of the somatic and dendritic thresholds, more and more synapses tend to assume low weight and the decay at large W becomes Gaussian. In the critical capacity limit a non-vanishing fraction of synapses become "silent", and the rest of the distribution is a Gaussian. Indeed, for $\alpha \to \alpha_c(\kappa)$, replacing the scaling of the order parameters as in Eq. 7.17, the $P(W)$ simplifies

$$\begin{aligned} P(W) = &\Theta(W) \, e^{-\frac{A}{2dq}W^2 + \frac{B\sqrt{C}}{dq}W} \sqrt{\frac{A}{dq}} \int Dz \, e^{\frac{\sqrt{C}}{dq}Wz} \left[G\left(\sqrt{\frac{C}{A}} \frac{B+z}{\sqrt{\hat{q}-2\hat{Q}}} \right) \Theta(B+z) \right. \\ &\left. - \sqrt{\frac{C}{Adq}} (B+z) \Theta(-B-z) \right] \end{aligned} \tag{7.27}$$

and using the identity

$$\int Dz \, e^{az} \, (z+b) \, \Theta(-b-z) = (a+b) \, e^{\frac{a^2}{2}} H(a+b) - e^{-ab} \, G(b) \tag{7.28}$$

we obtain

$$P(W) = H(-B)\,\delta(W) + \frac{1}{\sqrt{2\pi}W_\star} e^{-\frac{(W+BW_\star)^2}{2W_\star^2}} \Theta(W) \tag{7.29}$$

where $W_\star \equiv \frac{\sqrt{C}}{A}$. As showed in [129] in the one layer neuron model the synaptic weight distribution changes from being exponential to being a Gaussian plus a spike consisting to a fraction $H(B)$ of "silent" weights at the critical capacity. Indeed as constraints due to the training set are added, more and more synapses tend to assume low weight. This is the case also in our two layer neuron model. It is interesting to note that both the distribution of synaptic weight at finite α (7.25) and at critical capacity are in form exactly the same as the one derived in [129] for the one layer neuron model; the dependence on the non-linearity induced by the dendrites is actually implicit in the order parameters. The difference in the non-linear dendritic contribution case is that, is order to set the fraction of silent synapses to reproduce experiments, one just needs to module the inhibitory contribution to the soma θ_s to have $H(-B) \sim 90\%$, given that the mean weight is fixed by experiments and so there are θ_d, f_{in} and f_{out}. Instead, in the linear case, one had to explicitly impose a reliability parameter $\kappa > 0$ in the classification to obtain $H(-B)$ with the correct value. So, the two-layer model is intrinsically more robust, and this goes in the right direction, since explicitly tuning κ is not realistic in biological settings. If one compares the theoretical distribution (7.29) with data recorded in [142], one finds a good agreement as it is shown in Fig. 7.5.

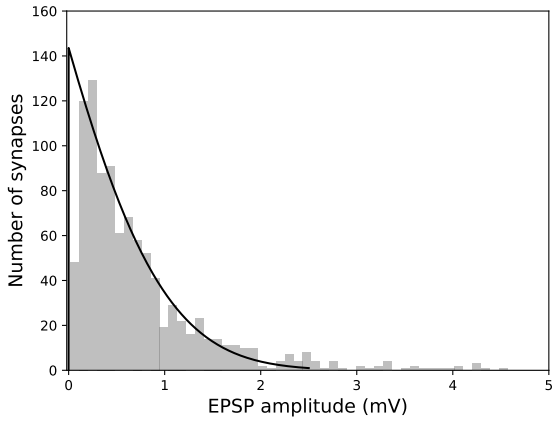


Figure 7.5: Theoretical (*full black curve*) and experimentally observed distributions of weights for pyramidal cells in neocortex. Real data are taken from [142], and contain recording on about 1000 connected pairs (out of about 10000 tried). Coding levels for fitting are set to $f_{in} = f_{out} = 0.05$, while $\theta_d \sim 0.039$ such that $\overline{W} \sim 0.778$ and $\theta_s = 0.8$.

7.4 Generalization performance

With the aim of studying the generalization properties of the neuron model in Eq.7.1, we perform binary classification tasks on the MNIST dataset. Notice that it is im-

portant to choose a division of the dataset that gives classes with similar coding level, otherwise, the classification task becomes trivial due to the natural correlation of the classification of the neuron with the coding level. The patterns of this real-world dataset are binarized in zeros and ones with a threshold given by the median of the single pattern. There is however an ambiguity on the possible choice for this binarization: we can assign to pixels lower than the median the value zero and to pixels higher than the median the value one, or viceversa (the two choices leading potentially to a different coding level of the patterns). It is possible to avoid both these problems by composing the two types of binarization in the same pattern (notice that this leads to doubling the size of the input and therefore the number of synaptic weights of the neuron). Furthermore, to preserve the locality of the inputs to the dendritic branch, each of them is provided with a different squared portion of the image (with both choices of the binarization combined) in such a way that the portions are non-overlapping between themselves.

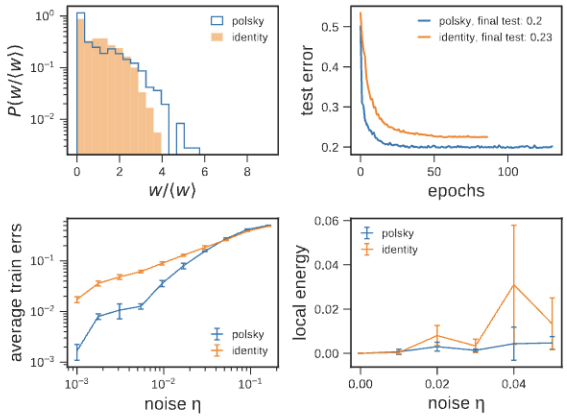


Figure 7.6: *Upper left*: weight distribution. *Upper right*: test error vs epochs (the curves end when the neuron has fitted the whole training set, i.e. the train error is zero). *Lower left*: robustness to input perturbations. *Lower right*: local energy.

Starting from the *bottom* subplots of Fig. 7.6, it is consistent with the storage case that the TCM is more robust (i.e. flatter local energy) than the Perceptron (again, *identity* in the legend) model. Plus, the generalization error is also better when integrating the dendritic contribution (*top-right* plot), since the flatter the solution, the better its prediction performances (see 2).

Although the model is a small step towards integrating some biological features by which statistical mechanics can simplify and take into isolation some computational features of biological and complex models, all the results on memory advantage, speed of learning, robustness to noise and generalization go towards the right direction. Still, much more can be done. Just as a clarification and to give a hint as to what are the possible implications after these first characterizations: first of all, in biological models N/K is generally not very large, which would require the effort of numerically integrating the saddle point equations for a finite value of K without taking advantage of CLT simplifications. This would probably also cure the gap between theoretical

and computational capacities, making the comparison more meaningful. Plus, future work could potentially concentrate on investigating the role of overparameterization and input sparsity in biological neural networks, or it would also be interesting to extend algorithms based on Belief Propagation [47, 143] to the kind of model studied here, in order to investigate their biological plausibility [86] and to make the analysis richer.

Chapter 8

The random features Hopfield Model

The Hopfield model (HM) [144] is a paradigmatic model of associative memory with a long-standing tradition in statistical physics, neuroscience and machine learning communities. It was originally created as a toy biological model, whose goal is to store P binary configurations, called *memories* or *patterns*, that represent the firing or non-firing states of N neurons. In the context, the prescribed memories $\boldsymbol{\xi}^\mu \in \{-1,+1\}^N$ for $\mu \in [P]$, are by definition successfully stored if the model dynamics has a fixed point very close to each pattern. The dynamics is that of a zero temperature greedy MCMC of an Ising spin system $s_i^{(t+1)} = \text{sign}\left(\sum_{j\neq i}^N J_{ij} s_j^{(t)}\right)$ where the synaptic connectivity matrix J is chosen to store the memories via the Hebb rule [145]. The rule $J_{ij} = \frac{1}{N}\sum_{\mu=1}^P \xi_{\mu i}\xi_{\mu j}$ summarizes one of the fundamental concepts in both neuroscience and machine learning, which is that neural connections are strengthened through simultaneous activation, at the basis of synaptic plasticity and unsupervised learning mechanisms. For the first, when a neuron s_i repeatedly and persistently takes part in firing another neuron s_j, then since the synaptic connection from s_i to s_j is strengthened, it suggests a mechanism in which learning involves understanding how different concepts are connected. From the second, this form of learning is unsupervised since it doesn't require external inputs or correction. The network self-organizes based on the input patterns it receives, in order to store them as basins of attraction. Given that the dynamics of the HM corresponds to minimize the energy function $\mathcal{H}(\boldsymbol{s}) = -\frac{1}{2}\sum_{i\neq j} J_{ij}s_i s_j$, it is possible to characterize the phase diagram of the associated statistical mechanics model using replica theory [146]. At zero temperature, the model can store up to $P = 0.138N$ random patterns, yielding an extensive capacity proportional to the number of neurons. In that retrieval ferromagnetic phase, the memories correspond to stable states dominating the Gibbs measure. But in general, if one represents the phase diagram in the temperature T versus loading density $\alpha = \frac{P}{N}$, it is easy to discover, that a part from the retrieval phase, whose capacity α_c decreases when increasing the temperature (it is $\alpha_c(T=0) \simeq 0.138$), there are in (α, T) both regions in which patterns still exist as a metastable states (but the equilibrium is dominated by a mixtures of the patterns) or when pure spin glass states come into play, erasing the retrieval states [147].

In the present Chapter, we will delve in the characterization of the phase diagram of an extension of the HM model inspired by the manifold hyphotesis in machine learning. As pointed out in previous chapters, we can address the limitation of i.i.d. unstructured data by proposing a generative model for the patterns where each pattern is produced by the linear combinations of a fixed vocabulary of what we call *features* weighted by pattern specific *coefficients*, followed by an element wise non-linearity. We will see how, by just changing the structure of data without modifying the Hebb rule, our associative memory model shows a rich phenomenology, where for some regions of parameters, features of the data become themselves stable fixed point attractors of the dynamics. We conjecture indeed that this simple variation, that we name Random-Feature Hopfield Model (RFHM) is able to *learn* from the data and thus to *generalize*, despite this being outside the scope with which the task is constructed. This refers to the works [148] and [149].

8.1 Phase diagram: storage and learning regimes

We consider structured patterns given by a linear projection and a latent vector composed with a non-linearity

$$\xi_{\nu i} = \sigma\left(\frac{1}{\sqrt{D}}\sum_{k=1}^{D} c_{\nu k} f_{ki}\right) \tag{8.1}$$

where $\sigma(\cdot)$ is a generic non-linear function, f_{ki} is called the matrix of *features* and $c_{\nu k}$ is the matrix of *coefficients* as for the standard projection in a RFM (chapter 4). The specific case we consider is the one of i.i.d. uniform binary features $f_{ki} = \pm 1$, i.i.d standard Gaussian coefficients $c_{\nu k}$, and σ equal to the sign function, so that it is still meaningful to consider the *magnetization* value

$$m_\nu = \frac{1}{N}\sum_{i=1}^{N} \xi_{\nu i} s_i,\ \nu \in [P] \tag{8.2}$$

which is the overlap between the νth embedded pattern and the state of the system with N nodes $s_i = \pm 1$ as binary spins. As anticipated, if the coupling matrix is defined through the set of P correlated patterns $\{\boldsymbol{\xi}_\nu\}_{\nu=1}^{P}$ via the Hebbian rule

$$J_{ij} = \frac{1}{N}\sum_{\nu=1}^{P} \xi_{\nu i}\xi_{\nu j}. \tag{8.3}$$

then nothing has changed wrt the standard HM, a part from Eq. (8.1). The goal of storing the $P = \alpha N$ patterns can be also seen by a denoising operation, since if the model is able to plant the patterns as attractor states for the dynamics, it does mean that it is capable of retrieving a given stored pattern starting from noisy version of it, because the dynamics lead it to fall into the corresponding basin. The critical capacity $\alpha_c(T = 0) \simeq 0.138$ of the standard HM means that the model is

able to retrieve all the patterns if $\alpha < \alpha_c$, while if $\alpha > \alpha_c$ the model undergoes a first order phase transition that leads it to *catastrophic forgetting*, i.e. where the dynamics fixed points are completely uncorrelated from the patterns. In our RFHM we are interested in understanding if there exists some regimes in which the features themselves $\mathbf{f}_k$ become attractors of the dynamics, and what happens accordingly to the patterns in the phases where the model has become able to learn the manifold that generates the data. In practice, we would like to see if in some regions of parameters is

$$\mu_k = \frac{1}{N}\sum_i f_{ki} s_i, \quad k \in [D] \tag{8.4}$$

the *feature magnetization* greater than zero for some k. As in 4, we have some $\mathcal{O}(1)$ external parameters to explore different phases

$$\alpha = \frac{P}{N}, \quad \alpha_D = \frac{D}{N}. \tag{8.5}$$

The first is the usual density of patterns, while by tuning the second we can switch between weakly and strongly correlated samples. In fact, in the $\alpha_D \to \infty$ limit we expect to recover the standard HM as the examples become uncorrelated (see Appendix D.2.1). In the thermodynamic limit $N \to \infty$ we choose the regime where P, D are proportional to N, and where the parameters (8.5) are related by the equation $\alpha = \alpha_D \alpha_T$, being $\alpha_T = P/D$. For the computation of the averaged free entropy $\phi = \lim_{N\to\infty} -\frac{1}{\beta N}\langle \ln Z\rangle_{c,f}$ via the replica method, we will make two ansatz on the order parameters (eq.s (8.2), (8.4)) for the structure of the solution: the case where the model retrieves one of the features, and the case where the model retrieves one of the examples.

8.1.1 Specializing to one feature retrieval

Average over $c_{\nu k>1}$

For the model to retrieve one feature, means that some $\mu_1 = \mathcal{O}(1)$ while the others $\mu_k = \mathcal{O}(\frac{1}{\sqrt{N}})$ are vanishing in the thermodynamic limit. At the same time, it is also $m_\nu = \mathcal{O}(\frac{1}{\sqrt{N}})$ for each of the P patterns, so we are looking for a solution of the form

$$\boldsymbol{\mu} = (\mu, 0, ..., 0) \quad \mathbf{m} = (0, ..., 0). \tag{8.6}$$

when $N \to \infty$. The quenched average in the free energy definition comes from the disorder induced by the random realization of coefficients $c_{\nu k}$ and features f_{ki}, and computing that average means to compute

$$\langle Z^n\rangle = e^{-\frac{\beta}{2}Pn} \sum_{\{s_i^a\}} \int \prod_{\nu a} \frac{dm_\nu^a}{\sqrt{2\pi}} e^{\frac{\beta}{2}\sum_\nu \sum_a (m_\nu^a)^2} \left\langle \prod_{\nu a} \delta\left(m_\nu^a - \frac{1}{\sqrt{N}}\sum_i \sigma\left(\frac{1}{\sqrt{D}}\sum_k c_{\nu k} f_{ki}\right) s_i^a \right) \right\rangle_{c,f} \tag{8.7}$$

in the case where f_{1i} and s_i^a are correlated by hypothesis. Using the GEP [102–104] as we did in 4 under certain conditions of the feature matrix (see eq. (4.6)), we

can average over the coefficients $c_{\nu k>1}$, being $P(m_\nu^a) = \mathcal{N}(m_\nu^a; \bar{m}, Q)$ the probability distribution of the variable m_ν^a that is a multivariate Gaussian where the mean is now proportional to the non-vanishing feature magnetization $\mu_1^a = \frac{1}{N}\sum_i f_{1i} s_i^a$ and the second moment has instead remained unchanged with respect to the standard GEP formulation. In this case, the term $k = 1$ gives a contribution to the first moment since we can isolate in

$$\left\langle \left\langle \prod_{\nu a} \delta\left(m_\nu^a - \frac{1}{\sqrt{N}} \sum_i \sigma\left(\frac{1}{\sqrt{D}} \sum_{k>1} c_{\nu k} f_{ki} + \frac{1}{\sqrt{D}} c_{\nu 1} f_{1i} \right) s_i^a \right) \right\rangle_{c_{\nu k>1}} \right\rangle_{f_{ki}} \tag{8.8}$$

the term with positive overlap involving f_{1i} and s_i^a. Given the usual definitions of coefficients $\kappa_0 = \int Dz\, \sigma(z)$, $\kappa_1 = \int Dz\, z\sigma(z)$, $\kappa_2 = \int Dz\, \sigma^2(z)$, $\kappa_*^2 = \kappa_2 - \kappa_1^2 - \kappa_0^2$, where $Dz = \frac{e^{-z^2/2}}{\sqrt{2\pi}} dz$, the mean of m_μ^a is obtained by using the hypothesis of the theorem, i.e. off-diagonal features are almost uncorrelated

$$\overline{m}^a = \frac{\kappa_0}{\sqrt{N}} \sum_i s_i^a + \frac{\kappa_1\, c_{\nu 1}}{\sqrt{\alpha_D}} \mu_1^a \tag{8.9}$$

and if μ_1^a is of order $O\left(\frac{1}{\sqrt{N}}\right)$, we recover back the standard mean in the Gaussian Equivalence as exposed in [99, 102–104]. The second moment computation can be performed similarly, but now even if μ_1^a is of $\mathcal{O}(1)$, the new terms will always be sub-leading in the thermodynamic limit

$$\left\langle m_\nu^a m_\nu^b \right\rangle_c = \frac{1}{N} \sum_i s_i^a s_i^b \left[\kappa_2^i - (\kappa_0^i)^2 - (\kappa_1^i)^2 \right] + \frac{1}{N} \sum_{ij} s_i^a s_j^b \kappa_0^i \kappa_0^j + \frac{1}{N} \sum_{ij} s_i^a s_j^b \left(\frac{1}{D} \sum_{k>1} f_{ki} f_{kj} \right) \kappa_1^i \kappa_1^j \tag{8.10}$$

with new variables

$$\kappa_0^i \equiv \int Dv\, \sigma\left(v + \frac{1}{\sqrt{D}} c_{\nu 1} f_{1i} \right) = \kappa_0 + \frac{c_{\nu 1}}{\sqrt{D}} f_{1i}\, \kappa_1 \tag{8.11a}$$

$$\kappa_1^i \equiv \int Dv\, v\, \sigma\left(v + \frac{1}{\sqrt{D}} c_{\nu 1} f_{1i} \right) = \kappa_1 + \frac{c_{\nu 1}}{\sqrt{D}} f_{1i} \int Dv\, \sigma''(v) \tag{8.11b}$$

$$\kappa_2^i \equiv \int Dv\, \sigma^2\left(v + \frac{1}{\sqrt{D}} c_{\nu 1} f_{1i} \right) = \kappa_2 + \frac{2c_{\nu 1}}{\sqrt{D}} f_{1i} \int Dv\, \sigma(v)\sigma'(v) + \left(\frac{c_{\nu 1}^2}{D} \right) \int Dv\, (\sigma'(v))^2 \tag{8.11c}$$

and so a covariance matrix of the form

$$Q_{ab} = \left\langle m_\nu^a m_\nu^b \right\rangle_c - \left\langle m_\nu^a \right\rangle_c \left\langle m_\nu^b \right\rangle_c = \frac{\kappa_*^2}{N} \sum_i s_i^a s_i^b + \frac{\kappa_1^2}{D} \sum_{k>1} \mu_k^a \mu_k^b = \kappa_*^2\, q^{ab} + \kappa_1^2\, p^{ab}. \tag{8.12}$$

Free energy expression

The computation proceeds involving standard steps, and since we can integrate all the vanishing pattern and feature magnetizations, the expression simplifies. Plus,

separating order parameters contributions into diagonal and off-diagonal terms as in the RS ansatz, and by performing the zero temperature limit $\beta \to \infty$, the free energy turns out to be (details in Appendix D.1)

$$f = -\frac{\alpha}{2}\left(1 + \delta q\,\hat{q} + \delta p\,\delta\hat{p}_d + p_d\,\delta\hat{p}\right) + \frac{\alpha}{2}\left(\frac{\kappa_*^2 + \kappa_1^2\,p_d}{1 - \kappa_1^2\delta p - \kappa_*^2\delta q} + \frac{\delta\hat{p} + \delta q\,\delta\hat{p}_d}{1 - \alpha_T\,\delta q\,\delta\hat{p}}\right) + \frac{\alpha_T}{2}\mu^2\,\delta\hat{p} - \mu\,\hat{\mu} + \frac{1}{2}\int Dz\,\left(z\sqrt{\alpha\,\hat{q}} + \hat{\mu}\right)\left(2\,\Theta\left(z\sqrt{\alpha\,\hat{q}} + \hat{\mu}\right) - 1\right) \tag{8.13}$$

with scalings

$$q = 1 - \frac{\delta q}{\beta} \tag{8.14a}$$

$$p = p_d - \frac{\delta p}{\beta} \tag{8.14b}$$

$$\hat{p} = \beta\,\delta\hat{p}_d - \frac{1}{2}\delta\hat{p} \tag{8.14c}$$

$$\hat{p}_d = \beta\,\delta\hat{p}_d + \frac{1}{2}\delta\hat{p} \tag{8.14d}$$

that can be seen by inspection. Now, by extremizing the free energy expression, one can derive saddle points equations for all the order parameters and look if for some α and α_D there are regions in the diagram for which $\mu \sim 1$. In order to explain the first results in Fig. 8.1, let us proceed the following way: suppose that we have a fixed value α for the density of patterns. Given α, increasing α_D and thus moving horizontally in the phase diagram (Fig. 8.1), means to increase the size of the vocabulary. When α_D is *low* enough and so it is the vocabulary, if α the number of patterns that the model sees is sufficiently *big*, then it is intuitive to agree that given that it sees multiple times some non-linear combination of features, it is in the end able to retrieve the features as attractor states. When the features instead become too many, the equivalent of catastrophic forgetting happens and suddenly is no longer possible to retrieve a $\boldsymbol{f}_k$. It happens precisely at the standard Hopfield critical capcity $\alpha_D \simeq 0.138$ only if $\alpha = \infty$, that is because in the limit $P \gg N, D$, we have that the coupling matrix, becomes (up to a factor that can be reabsorbed in the temperature) the standard matrix of a HM where the patterns are replaced by the features. In general, we have

$$\tilde{J}_{ij} = \frac{1}{P}\sum_{\mu=1}^{P}\xi_i^\mu\xi_j^\mu \tag{8.15}$$

if we rescale the couplings by a P/N factor that can be absorbed in the temperature. For large number of examples P and a rotationally invariant distribution $P(\mathbf{c})$, the

RFHM couplings become

$$\tilde{J}_{ij} = \frac{1}{P}\sum_{\mu}\sigma\left(\frac{1}{\sqrt{D}}\mathbf{c}_\mu\cdot\mathbf{f}_i\right)\sigma\left(\frac{1}{\sqrt{D}}\mathbf{c}_\mu\cdot\mathbf{f}_j\right) \tag{8.16}$$

$$\approx \int dP(\mathbf{c})\,\sigma\left(\frac{1}{\sqrt{D}}\mathbf{c}\cdot\mathbf{f}_i\right)\sigma\left(\frac{1}{\sqrt{D}}\mathbf{c}\cdot\mathbf{f}_j\right) \tag{8.17}$$

$$= r\left(\frac{1}{D}\mathbf{f}_i\cdot\mathbf{f}_j\right) \tag{8.18}$$

where in the last line we used rotational invariance to express the coupling as a function of the scalar product among the two features. The function $r(z)$ depends on the ensemble considered and on the activation function. If $r(z)\approx az$ for small argument we recover the standard Hopfield model, up to a prefactor that can be again reabsorbed in T. In the case of Gaussian $\boldsymbol{c}$ and non-decreasing activation functions, we have

$$\tilde{J}_{ij} \approx \int \frac{du d\hat{u}}{(2\pi)^2}\,\sigma(u)\,\sigma(v)\exp\left\{-i\hat{u}u - i\hat{v}v - \frac{1}{2D}\hat{u}^2\|\mathbf{f}_i\|^2 - \frac{1}{2D}\hat{v}^2\|\mathbf{f}_j\|^2 - \frac{1}{D}\hat{u}\hat{v}\mathbf{f}_i\cdot\mathbf{f}_j\right\}$$
$$= \kappa_1^2\frac{1}{D}\mathbf{f}_i\cdot\mathbf{f}_j \tag{8.19}$$

considering independently distributed feature vectors, with $\|\mathbf{f}_i\|^2 = D$, $\|\mathbf{f}_j\|^2 = D$ and $\mathbf{f}_i\cdot\mathbf{f}_j = O(\sqrt{D})$. So if the matching between the RFHM and the standard HM is perfect for $\alpha=\infty$ and μ in that regime takes the role of the magnetization, then the limiting $\alpha_D \sim 0.138$. In general, for $\alpha > \alpha_c(\alpha_D)$, the feature magnetization becomes finite with a discontinuous jump as in the usual case (*right* side, Fig. 8.1). The critical point $\alpha_c(\alpha_D)$ rapidly increases when α_D increases, and it finally diverges in the limit (8.19). So, since if the dataset is large enough the model is able to retrieve the features, this resembles the feature extraction that deep neural networks and some shallow generative models perform [150–153], explaining the reason why we call this regime *learning* phase.

8.1.2 Retrieval of one pattern

For the second case, looking instead for a solution of the form

$$\boldsymbol{\mu} = (0,...,0), \quad \mathbf{m} = (m,0,...,0) \tag{8.20}$$

means to carefully apply the GEP only to the vanishing pattern magnetizations, leaving the terms involving m_1 as they are

$$\prod_{\nu>1,k}\left\langle\prod_a \delta\left(m_\nu^a - \frac{1}{\sqrt{N}}\sum_i \sigma\left(\frac{1}{\sqrt{D}}\sum_k c_{\nu k}f_{ki}\right)s_i^a\right)\right\rangle_{c_{\nu k}} = \prod_{\nu>1}\frac{1}{\sqrt{2\pi\det Q}}e^{\frac{1}{2}\sum_{ab}m_\nu^a\left[Q^{-1}\right]_{ab}m_\nu^b} \tag{8.21}$$

where now the only difference is that all the feature magnetizations defined in eq. (8.4) scale as $1/\sqrt{N}$.

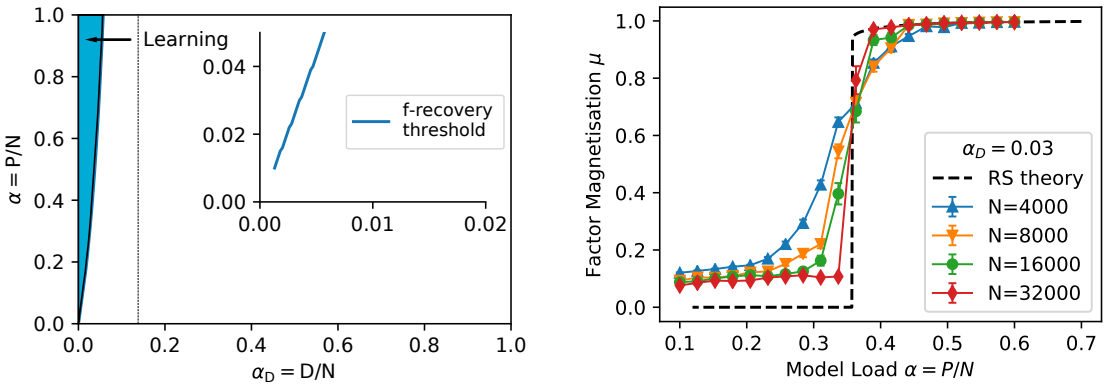


Figure 8.1: (*left*) First part of the phase diagram (α_D, α) for the RFHM. The highlighted region refers to the phase where the features $\boldsymbol{f}_k$ are attractors. (*right*) Feature magnetization μ along a vertical cut of the phase diagram: increasing α the feature magnetization μ becomes different from zero with a first-order phase transition. The dashed line is the analytical prediction of the RS theory, while the markers are numerical experiments averaged over many samples for each value of α. The simulations are performed initializing the model to a feature $\mathbf{f}_k$, running the update rule (D.3), then measuring μ_k at convergence. Reprinted from [148].

Free energy expression

After the usual integrations of pattern and feature magnetizations that we do not want to retrieve, imposing the RS ansatz and exploiting the limit $\beta \to \infty$, we get

$$f = \frac{1}{2}\left(m^2 + \alpha\,\delta p \delta\hat{p}_d + \alpha\,\delta q\left(\hat{q} - \frac{\delta\hat{p}_d}{1-\alpha_T\,\delta q\,\delta\hat{p}}\right)\right) + \frac{\alpha}{2}\delta\hat{p}\left(p_d - \frac{1}{1-\alpha_T\,\delta q\delta\hat{p}}\right)$$
$$- \int Dx \int Dv \left(m + x\sqrt{\alpha\,\hat{q}}\right)\left(\theta\left(m + x\sqrt{\alpha\,\hat{q}}\right) + \theta\left(v\right) - 1\right) \tag{8.22}$$

the free energy expression that we report for clarity (details in D.2). This time, the set of the order parameters do not include μ and $\hat{\mu}$, but it does include m (without the need for a conjugate variable $\hat{m}$). In that storage phase (Fig. 8.2), since the patterns that arrive to the model are correlated, the more one increases the level of correlation (and accordingly decreases α_D), the lower is the maximal storage capacity $\alpha_c(\alpha_D)$. Indeed, since all the patterns share a infinite size vocabulary, accomodating each of them as an isolated basin is no longer possible by construction, since each pair of $\boldsymbol{\xi}_\nu$ will have a non-trivial overlap. A useful limit to consider is $\alpha_D \to \infty$, where the correlation between patterns becomes vanishing and the RFHM reduces to the HM (Appendix D.2.1), with a pattern retrieval capacity $\alpha \sim 0.138$ at zero temperature. In Fig. 8.2, α is an increasing function of α_D, and accordingly the recovery threshold $\alpha \to 0$ when $\alpha_D \to 0$. The comparison between analytical results and numerical simulations (details in D.3.1), are reported in the right panel of Fig. 8.2. For all the range of α_D, numerics shows lower threshold than the theory, suggesting that strong

correlations might be responsible of a failure of the RS ansatz. In fact, in [146], the authors found that the correct ansatz at zero temperature is indeed the full-replica-symmetry-breaking one, letting the entropy to become more negative (Fig. 8.3, *left*), the smaller is the value of α_D.

One can also rule out a possible inconsistency of the solution ansatz by numerically check that the magnetizations $m_{\nu>1}$ and μ_k are effectively going to zero in the thermodynamic limit ($N \to \infty$) when initializing the model in $\boldsymbol{\xi}_1$ (Fig. 8.3, *right*).

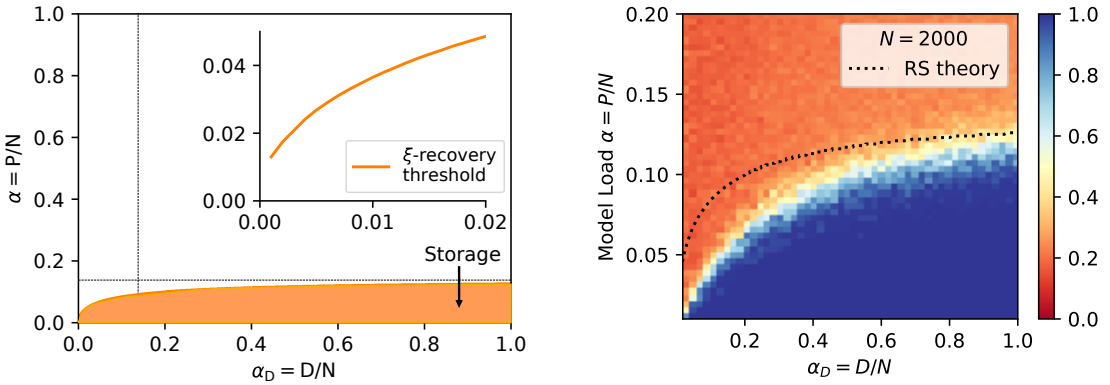


Figure 8.2: (*left*) Second part of the phase diagram (α_D, α) for the pattern recovery threshold. The critical capacity $\alpha_c(\alpha_D)$ decreases when the degree of correlation between patterns increases (α_D decreases accordingly). (*right*) Comparison with numerical results for the retrieval of one pattern. Each pixel represents the mean pattern magnetization for given values of α and α_D. The simulations are performed initializing the model to a pattern $\boldsymbol{\xi}_\nu$, running the update rule, then measuring m_ν at convergence. Reprinted from [148].

8.1.3 Merging the results

Summarizing the results, the phase diagram we have developed reveals two distinct transition lines that define three separate areas in the plane (α_D, α): a feature retrieval regime, identified by a non-zero feature magnetization solutions, the pattern retrieval region, for which we have non-zero pattern magnetization solutions, and an intermediate spin-glass region between the two (Fig. 8.4). The process we describe as the "learning transition" becomes apparent when one follows a vertical trajectory in the phase diagram, namely by fixing a value of α_D and progressively increasing α. Beginning from lower values of α, the model exhibits the storage of correlated patterns: although its capacity is less than the one for the standard uncorrelated case, the general behavior remains similar, marked by a maximum threshold for the number of storable patterns, beyond which the magnetization m_ν suddenly drops to zero. Notably, when α is equal or less than 0.138, further increasing the patterns unveils the new phase beyond the spin-glass state, where the model has become able to infer the underlying features present in the data.

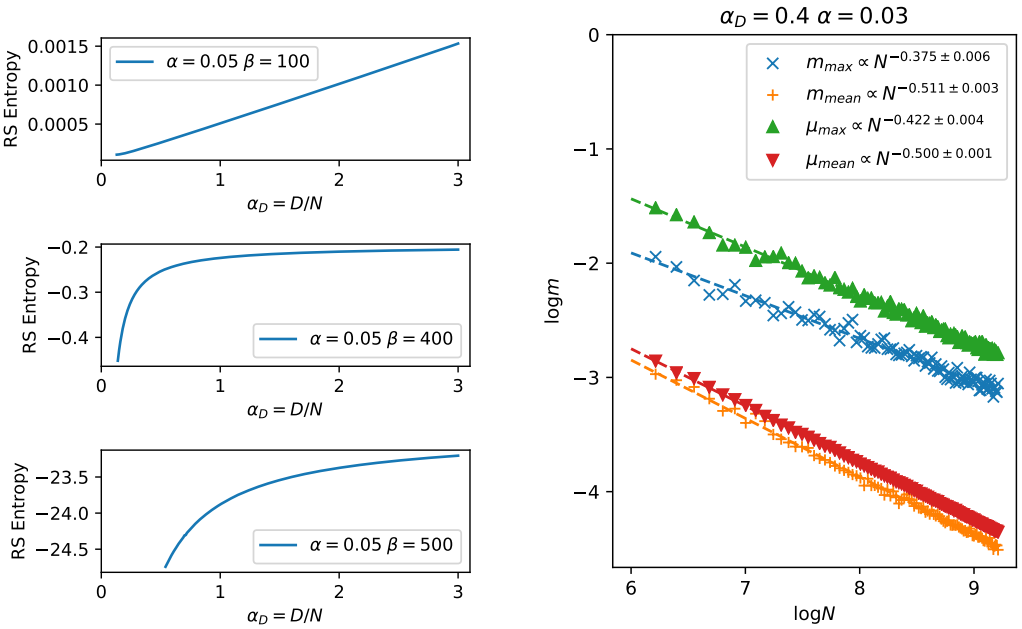


Figure 8.3: Analysis of the discrepancy between simulations and theory in the case of the retrieval of one pattern. (*left*) If the temperature is low enough, the entropy becomes negative, signaling the incorrectness of the RS ansatz. As we lower α_D we see that the entropy becomes more negative, which is consistent with the RS solution progressively becoming a worse approximation of the numerical simulations. (*right*) Numerical check that the residual magnetizations goes to zero as $N \to \infty$ when the model is initialized in $\boldsymbol{\xi}_1$. Reprinted from [148].

8.2 Generalization in the RFHM

Having clarified that it is possible to learn the features if the matrix $f \in \mathbb{R}^{D\times N}$ is dense and P is large but still $\alpha = \mathcal{O}(1)$, it is not trivial to answer the question: for a finite number of features, which is the most realistic assumption to have when considering a low-dimensional manifold of data, how many patterns P do I need with respect to the model dimension N, to be still able to extract $\boldsymbol{f}_k$ from $\boldsymbol{\xi}_\nu$? If we have a finite vocabulary c_ν with exactly L non-zero entries, in random locations and uniformly sampled in ± 1, then the latent vectors $c_\nu \in \mathbb{R}^D$ in that sparse case will be such that $L = \mathcal{O}(1)$ (the standard fully-dense case is $L = D$). GEP is no longer valid when the vocabulary is sparse, and so to answer the question we can use both numerical simulations and signal-to-noise analysis.

To clarify notation, given that the examples are correlated, it is useful to call *train example*, $\boldsymbol{\xi}^{\text{train}}$, any of the P examples that are used in the Hebb rule defining the couplings matrix J (Eq. (8.3)). Instead, we also call *test example*, $\boldsymbol{\xi}^{\text{test}}$, any linear combination (followed by the sign activation) of the features that is not used in J. Starting from high density of correlation, i.e. low value of α_D, we can increase the density of patterns α to inspect what happens to the magnetizations $m^{\text{train}} =$

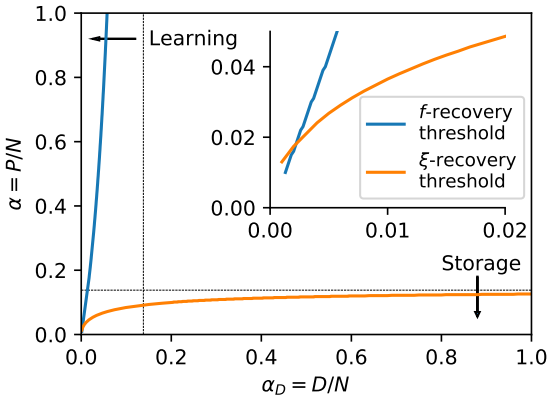


Figure 8.4: Completed phase diagram of the RFHM in the (α_D, α) plane. Three phases can be distinguished as we vary α and α_D: patterns are stable fixed points of the dynamics in the storage phase; features are stable fixed points in the learning phase; spin-glass dominated regime happens between the two. Reprinted from [148].

$\frac{1}{N}\sum_{i=1}^{N} \tilde{s}_i \xi_i^{\text{train}}$, $m^{\text{test}} = \frac{1}{N}\sum_{i=1}^{N} \tilde{s}_i \xi_i^{\text{test}}$, and $\mu = \frac{1}{N}\sum_{i=1}^{N} \tilde{s}_i f_i$ respectively. To do so, we can just initialize the model to the configuration $\boldsymbol{s}^{(0)}$ whose stability we want to check (we consider $\boldsymbol{s}^{(0)} = \boldsymbol{\xi}^{\text{train}}$, $\boldsymbol{s}^{(0)} = \boldsymbol{\xi}^{\text{test}}$, and $\boldsymbol{s}^{(0)} = \boldsymbol{f}_k$, with $\boldsymbol{f}_k$ any of the columns of f), and then we run the update rule

$$s_i^{(t+1)} = \text{sign}\left(\sum_{j\neq i}^{N} J_{ij} s_j^{(t)}\right) \tag{8.23}$$

until the model reaches a fixed point $\tilde{\boldsymbol{s}}$. Finally, we compute the magnetization as the normalized scalar product between the fixed point and the initial condition. What happens is summarized in Fig. 8.5 (*left* panel). For low α and α_D, the density of patterns is low and indeed the model is able to store $\boldsymbol{\xi}_\nu$ as attractor states; by increasing α, as it is for the standard HM, the magnetization m^{train} decreases and then suddenly drops to zero at the threshold α_c. In this regime, all three the magnetizations are vanishing, suggesting that we enter the spin-glass phase where the attractors are uncorrelated with either $\boldsymbol{\xi}^{\text{train}}, \boldsymbol{\xi}^{\text{test}}$ and $\boldsymbol{f}_k$. But as it is in the dense case, if one keeps increasing P the number of patterns and α accordingly, suddenly the features become stable minima of the free energy. Then, surprisingly, after the features have been learned, the model enters a phase in which the train examples are stable again (see Fig. 8.5, orange line). Or rather, it is physically implausible that a phase disappears for low α and reappears when α is large. They are in fact the test examples that become fixed points together with the train examples. We call this the *generalization phase*, as it resembles the behavior of inference models that perform well on previously unseen examples.

In Fig. 8.6 we show learning (dashed lines) and generalization transitions (solid lines) for different values of L and α_D. We observe that increasing either L or α_D results in shifting the generalization transition to higher α values. Interestingly, however, the learning transition shows a minimal (if any) dependence on L, since

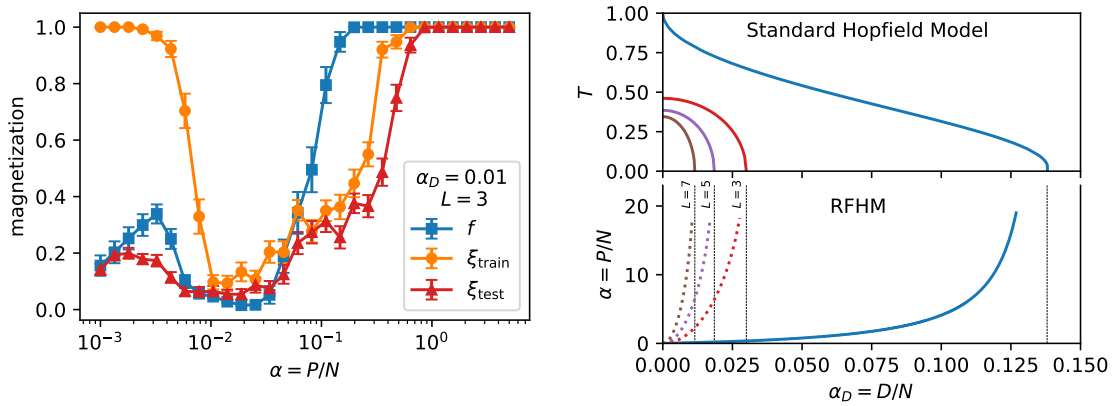


Figure 8.5: (*left*) Magnetization as a function of α, for fixed α_D. Blue line is the magnetization μ of hidden features, which grows to 1 if α is high enough (*learning phase*). Orange line is the magnetization m^{train} of the train examples, which is $m^{\text{train}} \simeq 1$ for low α and drops when α increases, as expected from an associative memory (*storage phase*). Surprisingly, m^{train} grows to 1 again for high values of α. Near this transition, also test examples have $m^{\text{test}} = 1$, as shown by the red line (*generalization phase*). (*right*) Comparison between the phase diagram of a standard HM from [146] (*top*, temperature T vs α) and the phase diagram of the dense RFHM (*bottom*, α vs α_D).

the vertical dashed line of the fully-dense case ($L = D$) coincides with the sparse case at any L. Further, in Fig. 8.6, we analyze the maximum number of features D_{gen} that coexists with a generalization transition: D_{gen} scales as $\mathcal{O}(N)$ if $L = \mathcal{O}(1)$, while as $\mathcal{O}(\sqrt{N})$ if $L = D$.

How to explain this phenomenon anyway? The conjecture is that the ability of the network to recognize previously unseen examples relies on the appearence of *spurious* feature combinations, analogous to how the classic Hopfield Network creates mixtures of training examples at low α values. Considering the RFHM in the limit $\alpha \to \infty$, it has been demonstrated already ([148]) that it converges to the classic model, with features having the role of examples. Therefore, proceeding by analogy as in [146], we can expect a transition at $\alpha_D \simeq 0.03$ where mixtures of $L = 3$ features become fixed points. Mixtures with larger values of L are likely to achieve stability at lower α values. We speculate that each of these transitions marks the beginning of the generalization lines in the α_D vs α plane. The expectation is that these lines will mimic the pattern of the learning transition, implying that with a finite α the generalization transition will shift towards lower α_D values. This aligns with the notion that combinations involving larger L necessitate a greater number of examples to become stable. It also corresponds with the observation that combinations with $L = D$ are only stable when $\alpha_D = 0$, as previously pointed in [146].

8.2.1 Signal-to-noise analysis

To support the phase diagram in Fig. 8.6, we can provide an intuitive signal-to-noise analysis, to both confirm that the learning transition occurs at finite α even if

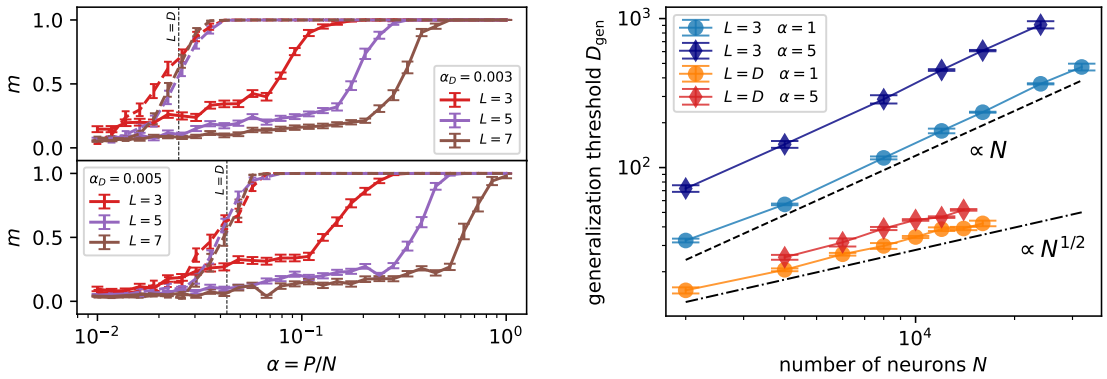


Figure 8.6: (*left*) Feature magnetization μ (*dashed*) and test examples magnetization m^{test} (*solid*) as a function of α for different α_D. Dashed vertical lines are the analytical predictions from [148]. (*right*) Scaling with N of the maximum number of features D_{gen} for which we observe a generalization transition. Specifically, the maximum $D(N)$ at which 10 samples have $m^{\text{test}} > 0.9$.

$L = \mathcal{O}(1)$ and that consequently, a mixture of features in that sparse case, become stable when $\alpha > 0$. First of all, we can write the local field of a single feature retrieval as

$$\begin{aligned} h_{k''i} &= \sum_{j=1}^{N} J_{ij} f_{k''j} \\ &= \frac{1}{L} \sum_{kk'} \left(\delta_{kk'} P \frac{L}{D} + (1 - \delta_{kk'}) O\left(\sqrt{P}\right) \frac{L^2}{D^2} \right) \left(\delta_{k'k''} + (1 - \delta_{k'k''}) O\left(\frac{1}{\sqrt{N}}\right) \right) f_{ki} \end{aligned} \tag{8.24}$$

where now the patterns in N-dimension are properly scaled with $\sqrt{L}$. As usual, we can isolate the signal term, and rescale accordingly the rest of noise

$$h_{k''i} = \frac{P}{D} \left[f_{k''i} + O\left(\sqrt{\frac{D}{N}}\right) + \frac{L}{D} O\left(\sqrt{\frac{D}{P}}\right) + \frac{L}{D} O\left(\sqrt{\frac{D^2}{PN}}\right) \right]. \tag{8.25}$$

When the signal can be recovered? Consistently with Fig. 8.6, at finite α and for $P, D, N \gg 1$ the transition seems to weakly depend on L, since all the noise terms that scale with L can be ignored. Instead, in the case of a mixture of features, if one defines $\chi_{\nu i} = \sum_{k''} c_{\nu k''} f_{k''i}$, where the coefficients at this level can be defined as either sparse or dense, the field reads

$$\sum_{j=1}^{N} J_{ij} \chi_{\nu j} = \sum_{k''} c_{\nu k''} \left[\frac{1}{N} \sum_{j} \sum_{\mu} \left(\frac{1}{\sqrt{L}} \sum_{k} c_{\mu k} f_{ki} \right) \left(\frac{1}{\sqrt{L}} \sum_{k'} c_{\mu k'} f_{k'i} \right) f_{k''j} \right] \tag{8.26}$$

$$= \frac{P}{D} \left[\chi_{\nu i} + O\left(\sqrt{\frac{LD}{N}}\right) + O\left(\sqrt{\frac{L^3}{DP}}\right) + O\left(\sqrt{\frac{L^3}{NP}}\right) \right]. \tag{8.27}$$

When $L = \mathcal{O}(1)$, as N, D approach infinity, the second and third noise terms vanish to zero. To keep a finite value in the first noise term, we need to set $D = \mathcal{O}(N)$. This means that the retrieval of sparse mixtures is feasible at a positive value of α. In contrast, retrieving dense mixtures at a finite α becomes unfeasible when $L = D = \mathcal{O}(N)$, as the signal would be overwhelmed by noise. It becomes possible instead when $L = D = O(N^{1/2})$, meaning $\alpha = 0$. In this particular regime, not only is the first cross-talk term finite, but the second and third noise terms are finite too. We recover so the scalings of the numerical results in Fig. 8.6.

The ability of an HM to store some odd non-linear combination of examples is not in general a new result [146, 154], but it gains a new perspective in light of this new model. Indeed, if the network is able to combine features hidden in the examples to produce new attractors, this means that it can also recognize all the samples generated with the same features. Even if, in general, one can retrieve attractors in the learning phase only having prior information on the features, meaning that the model will dynamically converge to some $\boldsymbol{f}_k$ only if initialized close to it [148], the generalization phase seems to overcome that issue. In fact, in that phase, the model is able to mix the correct features without any prior information about them.

Part IV

Conclusions

Chapter 9

Conclusions and perspectives

This thesis, whose goal is to demonstrate how in non-convex optimization models it is the existence of minima associated with different performances that impacts the learning process, provides an overview on the geometric characterization of solutions for simple models of neural networks and supervised classification tasks. The non-rigorous analytical methods used are those in statistical mechanics of disordered systems (i.e., the replica method), which provide a snapshot of the equilibrium potential states and offer insights into different features of the network, such as its capacity, the generalization ability, and the robustness of solutions. The purpose, which is to explain why in practice the learning dynamics in a landscape that is typically so high dimensional and non-convex [22, 23] converges to flat minima [8, 40, 45–49] (i.e., highly surrounded by other solutions) is addressed in Parts I and II with three examples of geometric landscape description [90, 99, 119]. Part I reports the first example of correlation between the existence of a subdominant number of solution clusters and their attractiveness for convergence of local convex optimization algorithms (e.g., SGD). Especially when looking at learning problems with constrained weights (i.e., binary), the entropic barriers given by a landscape where point-like isolated solutions are hard to find, are dynamically overcome by the existence of performing clusters with high local entropy values [45–48, 86]. The local entropy of a solution, which is a proportional measure of how much a configuration is surrounded at a given distance by other minima, is a possible key for improving iterative optimization strategies. Indeed, empirical evidence suggests that the flatness of a minimizer correlates with its performance in generalization. In practice, to avoid bad (sharp) minima both dynamically and at convergence, it is possible to replicate the model, adding an elastic interaction between all of the replicas, and then prooceed as usual with the optimization. Local entropy is both an observable to statically describe significant flat regions in the landscape and an objective if dynamically one wishes to improve the network performance.

Another measure related to flatness is the robustness of a solution, also called the margin. It refers to the distance between the decision boundary (like a hyperplane in linear classifiers) and the nearest data point from each class. A larger margin implies that the decision boundary is not just separating the classes but doing so with a significant buffer zone. Indeed, the margin is a measure of confidence in the classification: the farther the data points are from the boundary, the more confident

the classification. So, since for sufficiently smooth functional mappings, a flat region in the solution space can be regarded as a part of the loss landscape where small changes in the network parameters lead to relatively small changes in the loss, and the robustness refers to how well a network maintains its performance when subjected to perturbations, the connection between the two is significant. In the models of Chapters 3 and 4, this connection is formalized, since the flat regions of solutions are described as a coalescence of minima with a high level of robustness [90, 99]. But is the presence or absence of robust models always a good estimate of how dynamically simple or difficult it is to converge to a minimum? In models with binary weights (Chapters 3 and 4), since typical solutions of the problem are point-like as mentioned [22, 23, 82], characterized by *frozen*-1RSB configurations, i.e. extensively distant from each other but with an internal overlap $q_1 \simeq 1$, when clusters of atypical solutions break down by increasing the density of constraints to be satisfied, there is no hope for algorithms to find the remaining sharp solutions in those regimes [90]. In practice, the gap between information theoretic vs computation in the average case is well represented for binary models by the breakdown of robust solutions, happening at the *local entropy* transition for $\alpha > \alpha_{LE}$ (i.e., when the Franz Parisi potential displays a gap as a function of distance for the maximum margin ($\kappa_{\max}$) solution). On the opposite, for continuous non-convex models, such as the spherical negative perceptron (Chapter 5), behaviour of algorithms with the structural change of the landscape in α is different, since even beyond the disappearance of extended clusters of minima, the remaining solutions are shown to always be surrounded by a large number of other solutions at any distance, up to the information theoretic threshold α_c (i.e., SAT/UNSAT threshold in statistical mechanics jargon). This is because in the spherical case the low energy landscape is expected to have a full-RSB organization, with a distrubution of pairwise overlaps $P(q)$ without gaps, for which even typical solutions (that are the point-like ones in the binary setting) are expected to be accessible at every satisfiable constraint density.

If one would like to have an insight on how the threshold at which the learning problem becomes computationally hard changes from the typical (i.e., α_{LE}) to the worst case, the last of which would be the algorithmic hardness regime (α_{OGP}) [81], and apply the current approach of looking for the Franz Parisi entropy profile as a function of distance from the picked reference configuration, one could imagine to sample the reference solution with the highest local entropy at every distance, and look for the constraint density value α for which the entropy becomes zero in an interval of distances. Of course, this would still not be a rigorous approach, since zero entropy only means to have a sub-exponential number of solutions in a certain range, not precisely zero, but it might decrease the gap between α_{LE} and α_{OGP}. It is not the purpose of this manuscript to delve into any kind of result in the worst case and we will left this discussion for future works. An alternative description of the geometric arrangement of solutions in continuous models, when the local entropy transition α_{LE} does not represent a barrier to algorithmic convergence, and related connections between the landscape shape and the algorithmic bias towards locally convex regionsLMC [40, 112–114], is the linear mode connectivity analysis (5) between solutions sampled independently with different margins κ [119]. For that, not only

high-margin solutions tend to concentrate in nearby regions, but for sufficiently high κ, there exist linear paths at zero energy that can connect them, identifying a subset region of solutions that is convex in expectation. Low margin solutions are outside the convex subset, but there exist some linear paths that can connect the outer region to the high-margin one: indeed, the kernel region represents the collection of robust solutions from which all the minima of a given learning problem can be linearly reached. The organization of the solution space is here *star-shaped*, i.e. composed of a core of high-margin solutions, while at the tips of the star there are low-margin ones. The convex core is attractive for SGD, since if one initializes the dynamics at the star tips, which cover the majority of the volume of solutions, in few epochs SGD ends up diffusing in this high-margin solution subset.

The linear mode connectivity property in non-convex optimization tasks, coupled with the implicit regularization effect of SGD dynamics favoring low-rank weight matrices [31–34], its tendency to flat minima, and the effectiveness of pruning techniques in managing overparameterized neural network architectures [50–52], are all interrelated subjects. These phenomena collectively suggest that, despite the apparent complexity of the loss landscape, the underlying geometric structure of performing solutions is usually simple. This enhances generalization - simple flat models are less likely to overfit - and robustness - smoother predictions make the model robust to perturbations. In order to leverage on the regularization effect of high κ solutions, and playing with the matryoshka organization of the space space wrt the margin, one can also derive results on the advantage of averaging methods. Suppose to be in the case where the patterns to be classified contain some structured noise (6): if one allows the network to make some mistakes in the prediction within a certain level of *negative* margin, and then takes the barycenter over a multiple set of low performing models (i.e., with energy greater than zero), this will target a solution that is both robust and well-generalizing. This is not surprising, since averaging has the effect of tracing out the noise by promoting a regularized barycenter. Plus, it is not trivial that by averaging over configurations that are not local minima of a given learning problem, one ends up in a configuration that is both solution and generalizes better than the exponentially many local minima of that given problem. It might be also possible extending the purpose of averaging as an optimization protocol, in which enlarging the solution space is obtained by implementing a scheme where the back-propagation of error signals is selectively adjusted or halted for errors within that negative margin. This could mean setting the gradients to zero for errors within the margin, essentially treating them as correct classifications for the purpose of weight updates. We leave that as a purpose for future works.

Finally, part III goes beyond the purpose of characterizing and improving the computational performance of neural networks per se, and focuses on two modeling examples, with the aim of exploiting artificial networks to investigate some organic features concerning cellular activity, such as the role of dendritic non-linearity [139] and pattern recognition (or associative memory) in neuroscience [148, 149]. For the former, it is well known from recording experiments with stimulating electrodes, that dendrites of pyramidal cells contain voltage-gated channels that can amplify or modulate signals, enabling them to generate local spikes known as dendritic spikes [131].

This property makes dendrites to act somewhat independently of the neuron main action potential, contributing to the neuron overall output in a non-linear fashion. This could possibly transform neurons into devices that are analogous to multi-layer perceptrons [129, 130, 137], where the role of non-linearity shapes how the synaptic signal is processed before sending it to the axon initial segment. In chapter 7, we propose a one hidden layer model of NN with non-overlapping receptive fields, known as Tree Committee Machine (TCM), to mimic the non-linear contribution of dendrites as experimentally observed [131]. The model, which has positive weights to focus on excitatory cells, includes the contribution of inhibitory signals with the presence of thresholds for the firing rates at each dendritic branch and on the final neuronal output. Studying how the non-linearity affects the memory capacity of the model let us to conclude how this can promote consistent growth compared to the standard linear case. Plus, non-linearity enhances the speed of convergence on equilibrium states, solutions of prototype classification tasks, and the robustness of these solutions towards input perturbations. It is also biologically plausible that most of the neuron activity reflects a relevant fraction of silent synapses [130], as confirmed by fitting experimental data with the theoretical prediction of the TCM. A possible improvement for the model, towards an approach that takes advantage of theoretical networks to explain the interplay of different agents in emerging biological features, would consist in investigating on how astrocytes affect learning and memory of a brain-like neuronal model, where recurrent currents are at the basis of interaction among cells. Indeed, astrocytes, which are a type of glial cell, have been increasingly recognized in the modulation of synaptic transmission, the maintenance of the blood-brain barrier, the regulation of blood flow, and the modulation of neural activity and plasticity. Given some recent results that neuron-astrocyte models with recurrent connections can naturally implement the core computation performed by the Transformer block in AI [155], it would be useful to characterize their impact on storing efficiently information, which may also explain the success of Transformers across a diverse set of information domains and computational tasks [156, 157].

Another prototype model of information storage, which implements a basic form of associative memory, storing activity patterns as fixed points attractors of the dynamics, is the Hopfield model [154]. In its standard version, due to the nature of the energy function, the update rules, and the fact that patterns are i.i.d. [145], presenting the network with a partial or noisy version of a stored pattern will lead the network to evolve towards the complete, noiseless version of the stored pattern. This makes the model able to store approximately 0.14 times the number of neurons in the network for random patterns [146, 147]. In trying to make a step forward towards more realistic assumptions about structure in data, if one instead considers patterns coming from a generative recipe where each input data is a non-linear combination of a fixed vocabulary (i.e. features) with some coefficients (4), a model that we named Random-Feature Hopfield Model (RFHM) [148, 149], it turns out that some rich phenomenology emerges. First, when the vocabulary size is *low* enough, and if the number of data that the model sees is sufficiently *big*, the model will be able to *learn* the features as attractor states, even though it never explicitly saw them as input data [148]. Plus, when this happens, since the model was able to learn in some sense

the manifold structure of data, it also becomes capable of producing new fixed point attractors as new combination of features hidden in the examples [149]. However, all this happens in a regime where the size of the vocabulary should be $\mathcal{O}(N^{1/2})$ the dimension N of the network if one wishes to retrieve dense mixtures of features. It still remains an open question how to modify the model to work in a regime in which $D = \mathcal{O}(N)$, maybe extending the model with high-order interactions [153, 158].

Publications

[90] Carlo Baldassi, Clarissa Lauditi, Enrico M. Malatesta, Gabriele Perugini, and Riccardo Zecchina. *Unveiling the Structure of Wide Flat Minima in Neural Networks*. 2021. URL: https://link.aps.org/doi/10.1103/PhysRevLett.127.278301.

[99] Carlo Baldassi, Clarissa Lauditi, Enrico M. Malatesta, Rosalba Pacelli, Gabriele Perugini, and Riccardo Zecchina. *Learning through atypical phase transitions in overparameterized neural networks*. 2022. URL: https://link.aps.org/doi/10.1103/PhysRevE.106.014116.

[119] Brandon Livio Annesi, Clarissa Lauditi, Carlo Lucibello, Enrico M. Malatesta, Gabriele Perugini, Fabrizio Pittorino, and Luca Saglietti. *Star-Shaped Space of Solutions of the Spherical Negative Perceptron*. 2023. URL: https://link.aps.org/doi/10.1103/PhysRevLett.131.227301.

[128] Carlo Baldassi, Clarissa Lauditi, Enrico M. Malatesta, Gabriele Perugini, Luca Saglietti, and Riccardo Zecchina. *In preparation*. 2023.

[139] Carlo Baldassi, Nicolas Brunel, Clarissa Lauditi, Enrico M. Malatesta, Fabrizio Pittorino, and Riccardo Zecchina. *In preparation*. 2023.

[148] M. Negri, C. Lauditi, G. Perugini, C. Lucibello, and E. Malatesta. *Storage and Learning Phase Transitions in the Random-Features Hopfield Model*. 2023. URL: https://link.aps.org/doi/10.1103/PhysRevLett.131.257301.

[149] Matteo Negri, Clarissa Lauditi, Gabriele Perugini, Carlo Lucibello, and Enrico M. Malatesta. *Random Feature Hopfield Networks generalize retrieval to previously unseen examples*. 2023.

Bibliography

[1] Zhizhen Chi, Hongyang Li, Jingjing Wang, and Huchuan Lu. *On the importance of network architecture in training very deep neural networks*. 2016. DOI: 10.1109/ICSPCC.2016.7753635.

[2] Frederik Benzing, Simon Schug, Robert Meier, Johannes von Oswald, Yassir Akram, Nicolas Zucchet, Laurence Aitchison, and Angelika Steger. *Random initialisations performing above chance and how to find them*. 2022. arXiv: 2209.07509.

[3] Arthur Jacot, François Ged, Berfin Şimşek, Clément Hongler, and Franck Gabriel. *Saddle-to-Saddle Dynamics in Deep Linear Networks: Small Initialization Training, Symmetry, and Sparsity*. 2022. arXiv: 2106.15933.

[4] Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu. *Deeply-Supervised Nets*. 2015. URL: https://proceedings.mlr.press/v38/lee15a.html.

[5] Yichuan Tang. *Deep Learning using Linear Support Vector Machines*. 2015. arXiv: 1306.0239.

[6] Katarzyna Janocha and Wojciech Marian Czarnecki. *On Loss Functions for Deep Neural Networks in Classification*. 2017. arXiv: 1702.05659.

[7] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2015. URL: http://arxiv.org/abs/1412.6980.

[8] Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent Sagun, and Riccardo Zecchina. *Entropy-SGD: Biasing Gradient Descent Into Wide Valleys*. 2017. URL: https://openreview.net/forum?id=B1YfAfcgl.

[9] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. *Sharpness-Aware Minimization for Efficiently Improving Generalization*. 2021. URL: https://openreview.net/forum?id=6Tm1mposlrM.

[10] Guido Montúfar, Razvan Pascanu, Kyunghyun Cho, and Yoshua Bengio. *On the Number of Linear Regions of Deep Neural Networks*. 2014. arXiv: 1402.1869.

[11] Ronen Eldan and Ohad Shamir. *The Power of Depth for Feedforward Neural Networks*. 2016. URL: https://proceedings.mlr.press/v49/eldan16.html.

[12] Matus Telgarsky. *Benefits of depth in neural networks*. 2016. URL: https://proceedings.mlr.press/v49/telgarsky16.html.

[13] Yoshua Bengio, Aaron Courville, and Pascal Vincent. *Representation Learning: A Review and New Perspectives*. 2014. URL: https://api.semanticscholar.org/CorpusID:393948.

[14] Francesco Cagnetta, Leonardo Petrini, Umberto M. Tomasini, Alessandro Favero, and Matthieu Wyart. *How Deep Neural Networks Learn Compositional Data: The Random Hierarchy Model*. 2023. arXiv: 2307.02129.

[15] Y. Bengio, P. Simard, and P. Frasconi. *Learning long-term dependencies with gradient descent is difficult*. 1994. DOI: 10.1109/72.279181.

[16] Xavier Glorot and Yoshua Bengio. *Understanding the difficulty of training deep feedforward neural networks*. Chia Laguna Resort, Sardinia, Italy, 2010. URL: https://proceedings.mlr.press/v9/glorot10a.html.

[17] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. *On the difficulty of training recurrent neural networks*. Ed. by Sanjoy Dasgupta and David McAllester. Atlanta, Georgia, USA, 2013. URL: https://proceedings.mlr.press/v28/pascanu13.html.

[18] Sergey Ioffe and Christian Szegedy. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. 2015. arXiv: 1502.03167.

[19] Andrew M. Saxe, James L. McClelland, and Surya Ganguli. *Exact solutions to the nonlinear dynamics of learning in deep linear neural networks*. 2013. URL: https://api.semanticscholar.org/CorpusID:17272965.

[20] Madhu S. Advani and Andrew M. Saxe. *High-dimensional dynamics of generalization error in neural networks*. 2017. arXiv: 1710.03667.

[21] Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. *Implicit Regularization of Discrete Gradient Dynamics in Linear Neural Networks*. 2019. arXiv: 1904.13262.

[22] Werner Krauth and Marc Mézard. *Storage capacity of memory networks with binary couplings*. 1989. URL: https://hal.science/jpa-00211124.

[23] Haiping Huang and Yoshiyuki Kabashima. *Origin of the computational hardness for learning with binary synapses*. 2014.

[24] Marco Baity-Jesi, Levent Sagun, Mario Geiger, Stefano Spigler, Gerard Ben Arous, Chiara Cammarota, Yann LeCun, Matthieu Wyart, and Giulio Biroli. *Comparing Dynamics: Deep Neural Networks versus Glassy Systems*. 2018. URL: https://proceedings.mlr.press/v80/baity-jesi18a.html.

[25] Andrea Crisanti and H. J. Sommers. *The sphericalp-spin interaction spin glass model: the statics*. 1992. URL: https://api.semanticscholar.org/CorpusID:122030390.

[26] Giorgio Parisi. *Spin glasses and fragile glasses: Statics, dynamics, and complexity*. 2006. DOI: 10.1073/pnas.0601120103. URL: https://www.pnas.org/doi/abs/10.1073/pnas.0601120103.

[27] Anna Choromanska, MIkael Henaff, Michael Mathieu, Gerard Ben Arous, and Yann LeCun. *The Loss Surfaces of Multilayer Networks*. San Diego, California, USA, 2015. URL: https://proceedings.mlr.press/v38/choromanska15.html.

[28] Kenji Kawaguchi. "Deep Learning without Poor Local Minima". In: *Advances in Neural Information Processing Systems*. Ed. by D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett. Vol. 29. Curran Associates, Inc., 2016. URL: https://proceedings.neurips.cc/paper_files/paper/2016/file/f2fc990265c712c49d51a18a32b39f0c-Paper.pdf.

[29] Haihao Lu and Kenji Kawaguchi. *Depth Creates No Bad Local Minima*. 2017. arXiv: 1702.08580.

[30] Kenji Kawaguchi and Yoshua Bengio. *Depth with nonlinearity creates no bad local minima in ResNets*. 2019. URL: https://www.sciencedirect.com/science/article/pii/S0893608019301820.

[31] Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. *Implicit Regularization in Deep Matrix Factorization*. 2019. arXiv: 1905.13655.

[32] Zhiyuan Li, Yuping Luo, and Kaifeng Lyu. *Towards Resolving the Implicit Bias of Gradient Descent for Matrix Factorization: Greedy Low-Rank Learning*. 2021. arXiv: 2012.09839.

[33] Nadav Timor, Gal Vardi, and Ohad Shamir. *Implicit Regularization Towards Rank Minimization in ReLU Networks*. 2022. arXiv: 2201.12760.

[34] Tomer Galanti and Tomaso Poggio. *SGD Noise and Implicit Low-Rank Bias in Deep Neural Networks*. 2022.

[35] Tanishq Kumar, Blake Bordelon, Samuel J Gershman, and Cengiz Pehlevan. *Grokking as the Transition from Lazy to Rich Training Dynamics*. 2023.

[36] Levent Sagun, Leon Bottou, and Yann LeCun. *Eigenvalues of the Hessian in Deep Learning: Singularity and Beyond*. 2017. arXiv: 1611.07476.

[37] Levent Sagun, Utku Evci, V. Ugur Guney, Yann Dauphin, and Leon Bottou. *Empirical Analysis of the Hessian of Over-Parametrized Neural Networks*. 2018. arXiv: 1706.04454.

[38] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. *Visualizing the Loss Landscape of Neural Nets*. 2018. arXiv: 1712.09913.

[39] Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred A. Hamprecht. *Essentially No Barriers in Neural Network Energy Landscape*. 2019. arXiv: 1803.00885.

[40] Fabrizio Pittorino, Antonio Ferraro, Gabriele Perugini, Christoph Feinauer, Carlo Baldassi, and Riccardo Zecchina. *Deep networks on toroids: removing symmetries reveals the structure of flat regions in the landscape geometry*. 2022. URL: https://doi.org/10.1088%2F1742-5468%2Fac9832.

[41] Rahim Entezari, Hanie Sedghi, Olga Saukh, and Behnam Neyshabur. *The Role of Permutation Invariance in Linear Mode Connectivity of Neural Networks*. 2022. arXiv: 2110.06296.

[42] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*. 2014. URL: http://jmlr.org/papers/v15/srivastava14a.html.

[43] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. *Deep Residual Learning for Image Recognition*. 2016. DOI: 10.1109/CVPR.2016.90.

[44] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*. 2014. URL: http://jmlr.org/papers/v15/srivastava14a.html.

[45] Carlo Baldassi, Christian Borgs, Jennifer T. Chayes, Alessandro Ingrosso, Carlo Lucibello, Luca Saglietti, and Riccardo Zecchina. *Unreasonable effectiveness of learning neural networks: From accessible states and robust ensembles to basic algorithmic schemes*. 2016. URL: https://api.semanticscholar.org/CorpusID:1558782.

[46] Carlo Baldassi, Alessandro Ingrosso, Carlo Lucibello, Luca Saglietti, and Riccardo Zecchina. *Subdominant Dense Clusters Allow for Simple Learning and High Computational Performance in Neural Networks with Discrete Synapses*. 2015. URL: https://doi.org/10.1103%2Fphysrevlett.115.128101.

[47] Carlo Baldassi, Fabrizio Pittorino, and Riccardo Zecchina. *Shaping the learning landscape in neural networks around wide flat minima*. 2019. DOI: 10.1073/pnas.1908636117. URL: https://doi.org/10.1073%2Fpnas.1908636117.

[48] Carlo Baldassi, Enrico M Malatesta, Matteo Negri, and Riccardo Zecchina. *Wide flat minima and optimal generalization in classifying high-dimensional Gaussian mixtures*. 2020. URL: https://doi.org/10.1088%2F1742-5468%2Fabcd31.

[49] Fabrizio Pittorino, Carlo Lucibello, Christoph Feinauer, Gabriele Perugini, Carlo Baldassi, Elizaveta Demyanenko, and Riccardo Zecchina. *Entropic gradient descent algorithms and wide flat minima**. Dec. 2021. URL: http://dx.doi.org/10.1088/1742-5468/ac3ae8.

[50] Song Han, Jeff Pool, John Tran, and William J. Dally. *Learning both Weights and Connections for Efficient Neural Networks*. 2015. arXiv: 1506.02626.

[51] Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. *Pruning Convolutional Neural Networks for Resource Efficient Inference*. 2017. arXiv: 1611.06440.

[52] Jonathan Frankle and Michael Carbin. *The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks*. 2019. arXiv: 1803.03635.

[53] Viplove Arora, Daniele Irto, Sebastian Goldt, and Guido Sanguinetti. *Quantifying lottery tickets under label noise: accuracy, calibration, and complexity*. 2023. arXiv: 2306.12190.

[54] Franco Pellegrini and Giulio Biroli. *Neural Network Pruning Denoises the Features and Makes Local Connectivity Emerge in Visual Tasks*. 2022. URL: https://proceedings.mlr.press/v162/pellegrini22a.html.

[55] Arthur Jacot, Franck Gabriel, and Clément Hongler. *Neural Tangent Kernel: Convergence and Generalization in Neural Networks*. 2020. arXiv: 1806.07572.

[56] Alexander G. de G. Matthews, Mark Rowland, Jiri Hron, Richard E. Turner, and Zoubin Ghahramani. *Gaussian Process Behaviour in Wide Deep Neural Networks*. 2018. arXiv: 1804.11271.

[57] Jaehoon Lee, Yasaman Bahri, Roman Novak, Samuel S. Schoenholz, Jeffrey Pennington, and Jascha Sohl-Dickstein. *Deep Neural Networks as Gaussian Processes*. 2018. arXiv: 1711.00165.

[58] Blake Bordelon, Abdulkadir Canatar, and Cengiz Pehlevan. *Spectrum Dependent Learning Curves in Kernel Regression and Wide Neural Networks*. 2021. arXiv: 2002.02561.

[59] Abdulkadir Canatar, Blake Bordelon, and Cengiz Pehlevan. *Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks*. 2021. DOI: 10.1038/s41467-021-23103-1.

[60] Maria Refinetti, Sebastian Goldt, Florent Krzakala, and Lenka Zdeborova. "Classifying high-dimensional Gaussian mixtures: Where kernel methods fail and neural networks succeed". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 8936–8947. URL: https://proceedings.mlr.press/v139/refinetti21b.html.

[61] Blake Bordelon and Cengiz Pehlevan. *Self-consistent dynamical field theory of kernel evolution in wide neural networks**. 2023. URL: https://dx.doi.org/10.1088/1742-5468/ad01b0.

[62] Blake Bordelon and Cengiz Pehlevan. *The Influence of Learning Rule on Representation Dynamics in Wide Neural Networks*. 2023. arXiv: 2210.02157.

[63] Nikhil Vyas, Alexander Atanasov, Blake Bordelon, Depen Morwani, Sabarish Sainathan, and Cengiz Pehlevan. *Feature-Learning Networks Are Consistent Across Widths At Realistic Scales*. 2023. arXiv: 2305.18411.

[64] Arthur Jacot, Franck Gabriel, and Clement Hongler. *Neural Tangent Kernel: Convergence and Generalization in Neural Networks*. 2018.

[65] Mario Geiger, Stefano Spigler, Stéphane d'Ascoli, Levent Sagun, Marco Baity-Jesi, Giulio Biroli, and Matthieu Wyart. *Jamming transition as a paradigm to understand the loss landscape of deep neural networks*. 2019. URL: https://doi.org/10.1103%2Fphysreve.100.012115.

[66] Jean-Philippe Bouchaud, Leticia F. Cugliandolo, Jorge Kurchan, and Marc Mézard. *Out of equilibrium dynamics in spin-glasses and other glassy systems*. 1997. arXiv: cond-mat/9702070.

[67] Ludovic Berthier and Giulio Biroli. *Theoretical perspective on the glass transition and amorphous materials*. 2011. URL: https://link.aps.org/doi/10.1103/RevModPhys.83.587.

[68] Leticia F. Cugliandolo. *Dynamics of glassy systems*. 2002. arXiv: cond-mat/0210312.

[69] C. Daniel Freeman and Joan Bruna. *Topology and Geometry of Half-Rectified Network Optimization*. 2017. arXiv: 1611.01540.

[70] Daniel Soudry and Yair Carmon. *No bad local minima: Data independent training error guarantees for multilayer neural networks*. 2016. arXiv: 1605.08361.

[71] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. *Reconciling modern machine-learning practice and the classical bias–variance trade-off*. 2019. URL: https://doi.org/10.1073%2Fpnas.1903070116.

[72] Rich Caruana, Steve Lawrence, and C. Giles. *Overfitting in Neural Nets: Backpropagation, Conjugate Gradient, and Early Stopping*. 2000.

[73] Matthias Schmidt and Hartmut Löwen. *Phase diagram of hard spheres confined between two parallel plates*. 1997. URL: https://link.aps.org/doi/10.1103/PhysRevE.55.7228.

[74] Corey S. O'Hern, Leonardo E. Silbert, Andrea J. Liu, and Sidney R. Nagel. *Jamming at zero temperature and zero applied stress: The epitome of disorder*. 2003. URL: https://link.aps.org/doi/10.1103/PhysRevE.68.011306.

[75] Giorgio Parisi and Francesco Zamponi. *Mean-field theory of hard sphere glasses and jamming*. 2010. URL: https://doi.org/10.1103%2Frevmodphys.82.789.

[76] Jorge Kurchan, Giorgio Parisi, and Francesco Zamponi. *Exact theory of dense amorphous hard spheres in high dimension I. The free energy*. 2012. URL: https://dx.doi.org/10.1088/1742-5468/2012/10/P10012.

[77] Jorge Kurchan, Giorgio Parisi, Pierfrancesco Urbani, and Francesco Zamponi. *Exact Theory of Dense Amorphous Hard Spheres in High Dimension. II. The High Density Regime and the Gardner Transition*. May 2013. URL: http://dx.doi.org/10.1021/jp402235d.

[78] Patrick Charbonneau, Jorge Kurchan, Giorgio Parisi, Pierfrancesco Urbani, and Francesco Zamponi. *Exact theory of dense amorphous hard spheres in high dimension. III. The full RSB solution*. Oct. 2013. DOI: 10.1088/1742-5468/2014/10/P10009.

[79] Matthieu Wyart, Leonardo E. Silbert, Sidney R. Nagel, and Thomas A. Witten. *Effects of compression on the vibrational modes of marginally jammed solids*. 2005. URL: https://link.aps.org/doi/10.1103/PhysRevE.72.051306.

[80] David Gamarnik. *The overlap gap property: A topological barrier to optimizing over random structures*. 2021. URL: https://api.semanticscholar.org/CorpusID:238249739.

[81] David Gamarnik, Eren C. Kızıldağ, Will Perkins, and Changji Xu. *Algorithms and Barriers in the Symmetric Binary Perceptron Model*. 2022. arXiv: 2203.15667.

[82] Damien Barbier, Ahmed El Alaoui, Florent Krzakala, and Lenka Zdeborová. *On the Atypical Solutions of the Symmetric Binary Perceptron*. 2023. arXiv: 2310.02850.

[83] Benjamin Aubin, Will Perkins, and Lenka Zdeborova. *Storage capacity in symmetric binary perceptrons*. 2019. URL: https://doi.org/10.1088%2F1751-8121%2Fab227a.

[84] Emmanuel Abbe, Shuangping Li, and Allan Sly. *Proof of the Contiguity Conjecture and Lognormal Limit for the Symmetric Perceptron*. 2021. arXiv: 2102.13069.

[85] Will Perkins and Changji Xu. *Frozen 1-RSB structure of the symmetric Ising perceptron*. 2021. arXiv: 2102.05163.

[86] Carlo Baldassi, Alfredo Braunstein, Nicolas Brunel, and Riccardo Zecchina. *Efficient supervised learning in networks with binary synapses*. 2007. URL: https://www.pnas.org/content/104/26/11079.

[87] Christoph von der Malsburg. *Frank Rosenblatt: Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Jan. 1986. DOI: 10.1007/978-3-642-70911-1_20.

[88] Enrico M. Malatesta. *High-dimensional manifold of solutions in neural networks: insights from statistical physics*. 2023. arXiv: 2309.09240.

[89] Luca Saglietti. *Out of Equilibrium Statistical Physics of Learning*. 2020.

[90] Carlo Baldassi, Clarissa Lauditi, Enrico M. Malatesta, Gabriele Perugini, and Riccardo Zecchina. *Unveiling the Structure of Wide Flat Minima in Neural Networks*. 2021. URL: https://link.aps.org/doi/10.1103/PhysRevLett.127.278301.

[91] Jian Ding and Nike Sun. *Capacity lower bound for the Ising perceptron*. 2018. arXiv: 1809.07742.

[92] E. Barkai, D. Hansel, and I. Kanter. *Statistical mechanics of a multilayered neural network*. 1990. URL: https://link.aps.org/doi/10.1103/PhysRevLett.65.2312.

[93] E. Barkai, D. Hansel, and H. Sompolinsky. *Broken symmetries in multilayered perceptrons*. 1992. URL: https://link.aps.org/doi/10.1103/PhysRevA.45.4146.

[94] A. Engel, H. M. Köhler, F. Tschepke, H. Vollmayr, and A. Zippelius. *Storage capacity and learning algorithms for two-layer neural networks*. 1992. DOI: 10.1103/PhysRevA.45.7590. URL: https://link.aps.org/doi/10.1103/PhysRevA.45.7590.

[95] Rémi Monasson and Riccardo Zecchina. *Weight Space Structure and Internal Representations: A Direct Approach to Learning and Generalization in Multilayer Neural Networks*. 1995. URL: https://link.aps.org/doi/10.1103/PhysRevLett.75.2432.

[96] Carlo Baldassi, Enrico M. Malatesta, and Riccardo Zecchina. *Properties of the Geometry of Solutions and Capacity of Multilayer Neural Networks with Rectified Linear Unit Activations*. 2019. URL: https://link.aps.org/doi/10.1103/PhysRevLett.123.170602.

[97] Jacob A. Zavatone-Veth and Cengiz Pehlevan. *Activation function dependence of the storage capacity of treelike neural networks*. 2021. DOI: 10.1103/PhysRevE.103.L020301. URL: https://link.aps.org/doi/10.1103/PhysRevE.103.L020301.

[98] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. *Binarized Neural Networks*. 2016.

[99] Carlo Baldassi, Clarissa Lauditi, Enrico M. Malatesta, Rosalba Pacelli, Gabriele Perugini, and Riccardo Zecchina. *Learning through atypical phase transitions in overparameterized neural networks*. 2022. URL: https://link.aps.org/doi/10.1103/PhysRevE.106.014116.

[100] E Gardner and B Derrida. *Three unfinished works on the optimal storage capacity of networks*. 1989. URL: https://dx.doi.org/10.1088/0305-4470/22/12/004.

[101] Song Mei and Andrea Montanari. *The generalization error of random features regression: Precise asymptotics and double descent curve*. 2020. arXiv: 1908.05355.

[102] Sebastian Goldt, Marc Mé zard, Florent Krzakala, and Lenka Zdeborová. *Modeling the Influence of Data Structure on Learning in Neural Networks: The Hidden Manifold Model*. 2020. URL: https://doi.org/10.1103%2Fphysrevx.10.041044.

[103] Federica Gerace, Bruno Loureiro, Florent Krzakala, Marc Mé zard, and Lenka Zdeborová. *Generalisation error in learning with random features and the hidden manifold model**. 2021. URL: https://doi.org/10.1088%2F1742-5468%2Fac3ae6.

[104] Sebastian Goldt, Bruno Loureiro, Galen Reeves, Florent Krzakala, Marc Mézard, and Lenka Zdeborová. *The Gaussian equivalence of generative models for learning with shallow neural networks*. 2021. arXiv: 2006.14709.

[105] Hong Hu and Yue M. Lu. *Universality Laws for High-Dimensional Learning with Random Features*. 2022. arXiv: 2009.07669.

[106] A. Engel and C. Van den Broeck. *Statistical Mechanics of Learning*. Cambridge University Press, 2001. DOI: 10.1017/CBO9781139164542.

[107] E Gardner. *The space of interactions in neural network models*. 1988.

[108] E Gardner and B Derrida. *Optimal storage properties of neural network models*. 1988.

[109] Carlo Baldassi, Alfredo Braunstein, Nicolas Brunel, and Riccardo Zecchina. *Efficient supervised learning in networks with binary synapses*. 2007.

[110] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. *Binarized neural networks*. 2016. URL: https://papers.nips.cc/paper/2016/hash/d8330f857a17c53d217014ee776bfd50-Abstract.html.

[111] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. *Deep double descent: Where bigger models and more data hurt*. 2021.

[112] Jonathan Frankle, Gintare Karolina Dziugaite, Daniel M. Roy, and Michael Carbin. *Linear Mode Connectivity and the Lottery Ticket Hypothesis*. 2020. arXiv: 1912.05671.

[113] David Yunis, Kumar Kshitij Patel, Pedro Henrique Pamplona Savarese, Gal Vardi, Jonathan Frankle, Matthew Walter, Karen Livescu, and Michael Maire. *On Convexity and Linear Mode Connectivity in Neural Networks*. 2022. URL: https://openreview.net/forum?id=TZQ3PKL3fPr.

[114] Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. *Git Re-Basin: Merging Models modulo Permutation Symmetries*. 2023. arXiv: 2209.04836.

[115] Ian J. Goodfellow, Oriol Vinyals, and Andrew M. Saxe. *Qualitatively characterizing neural network optimization problems*. 2015. arXiv: 1412.6544.

[116] Jonathan Frankle. *Revisiting "Qualitatively Characterizing Neural Network Optimization Problems"*. 2020. arXiv: 2012.06898.

[117] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Dilan Gorur, Razvan Pascanu, and Hassan Ghasemzadeh. *Linear Mode Connectivity in Multitask and Continual Learning*. 2020. arXiv: 2010.04495.

[118] Gregory Benton, Wesley Maddox, Sanae Lotfi, and Andrew Gordon Gordon Wilson. *Loss Surface Simplexes for Mode Connecting Volumes and Fast Ensembling*. 2021. URL: https://proceedings.mlr.press/v139/benton21a.html.

[119] Brandon Livio Annesi, Clarissa Lauditi, Carlo Lucibello, Enrico M. Malatesta, Gabriele Perugini, Fabrizio Pittorino, and Luca Saglietti. *Star-Shaped Space of Solutions of the Spherical Negative Perceptron*. 2023. URL: https://link.aps.org/doi/10.1103/PhysRevLett.131.227301.

[120] E Gardner and B Derrida. *Optimal storage properties of neural network models*. 1988. URL: https://dx.doi.org/10.1088/0305-4470/21/1/031.

[121] Silvio Franz, Giorgio Parisi, Maksim Sevelev, Pierfrancesco Urbani, and Francesco Zamponi. *Universality of the SAT-UNSAT (jamming) threshold in non-convex continuous constraint satisfaction problems*. 2017. URL: https://scipost.org/10.21468/SciPostPhys.2.3.019.

[122] Silvio Franz and Giorgio Parisi. *The simplest model of jamming*. 2016. URL: https://dx.doi.org/10.1088/1751-8113/49/14/145001.

[123] Carlo Baldassi, Enrico M. Malatesta, Gabriele Perugini, and Riccardo Zecchina. *Typical and atypical solutions in non-convex neural networks with discrete and continuous weights*. 2023. arXiv: 2304.13871.

[124] Ahmed El Alaoui, Andrea Montanari, and Mark Sellke. *Shattering in Pure Spherical Spin Glasses*. 2023. arXiv: 2307.04659.

[125] Andrea Montanari, Yiqiao Zhong, and Kangjie Zhou. *Tractability from overparametrization: The example of the negative perceptron*. 2023. arXiv: 2110.15824.

[126] Ludovic Berthier and Giulio Biroli. *Theoretical perspective on the glass transition and amorphous materials*. 2011. DOI: 10.1103/revmodphys.83.587. URL: https://doi.org/10.1103%2Frevmodphys.83.587.

[127] T. R. Kirkpatrick and D. Thirumalai. *Colloquium: Random first order transition theory concepts in biology and physics*. 2015. URL: https://link.aps.org/doi/10.1103/RevModPhys.87.183.

[128] Carlo Baldassi, Clarissa Lauditi, Enrico M. Malatesta, Gabriele Perugini, Luca Saglietti, and Riccardo Zecchina. *In preparation*. 2023.

[129] Nicolas Brunel, Vincent Hakim, Philippe Isope, Jean-Pierre Nadal, and Boris Barbour. *Optimal information storage and the distribution of synaptic weights: perceptron versus Purkinje cell*. 2004.

[130] Nicolas Brunel. *Is cortical connectivity optimized for storing information?* 2016. DOI: 10.1038/nn.4286.

[131] Alon Poleg-Polsky, Bartlett Mel, and Jackie Schiller. *Polsky, A., Mel, B.W. Schiller, J. Computational subunits in thin dendrites of pyramidal cells. Nat. Neurosci. 7, 621-627*. July 2004. DOI: 10.1038/nn1253.

[132] Nelson Spruston, Greg Stuart, and Michael Häusser. *Principles of dendritic integration*. English. Mar. 2016.

[133] S. J. Martin, P. D. Grimwood, and R. G. M. Morris. *Synaptic Plasticity and Memory: An Evaluation of the Hypothesis*. 2000. URL: https://doi.org/10.1146/annurev.neuro.23.1.649.

[134] David Holcman, Zeev Schuss, and Eduard Korkotian. *Calcium Dynamics in Dendritic Spines and Spine Motility*. Aug. 2004. DOI: 10.1529/biophysj.103.035972.

[135] Jeremy Edgerton, Jesse Hanson, Cengiz Gunay, and Dieter Jaeger. *Dendritic Sodium Channels Regulate Network Integration in Globus Pallidus Neurons: A Modeling Study*. Nov. 2010. DOI: 10.1523/JNEUROSCI.2662-10.2010.

[136] Matthew E. Larkum, Thomas Nevian, Maya Sandler, Alon Polsky, and Jackie Schiller. *Synaptic Integration in Tuft Dendrites of Layer 5 Pyramidal Neurons: A New Unifying Principle*. 2009. URL: https://www.science.org/doi/abs/10.1126/science.1171958.

[137] Nicolas Brunel, Vincent Hakim, and Magnus JE Richardson. *Single neuron dynamics and computation*. 2014. URL: https://www.sciencedirect.com/science/article/pii/S0959438814000130.

[138] Balázs B Ujfalussy, Máté Lengyel, and Tiago Branco. *Global, multiplexed dendritic computations under in vivo-like conditions*. 2017. DOI: 10.1101/235259. URL: https://www.biorxiv.org/content/early/2017/12/16/235259.

[139] Carlo Baldassi, Nicolas Brunel, Clarissa Lauditi, Enrico M. Malatesta, Fabrizio Pittorino, and Riccardo Zecchina. *In preparation*. 2023.

[140] Claudia Clopath, Jean-Pierre Nadal, and Nicolas Brunel. *Storage of Correlated Patterns in Standard and Bistable Purkinje Cell Models*. Apr. 2012. URL: https://doi.org/10.1371/journal.pcbi.1002448.

[141] A. Engel, H. M. Köhler, F. Tschepke, H. Vollmayr, and A. Zippelius. *Storage capacity and learning algorithms for two-layer neural networks*. 1992. DOI: 10.1103/PhysRevA.45.7590. URL: https://link.aps.org/doi/10.1103/PhysRevA.45.7590.

[142] Sen Song, Per Jesper Sjöström, Markus Reigl, Sacha Nelson, and Dmitri B Chklovskii. *Highly Nonrandom Features of Synaptic Connectivity in Local Cortical Circuits*. Mar. 2005. DOI: 10.1371/journal.pbio.0030068.

[143] Carlo Lucibello, Fabrizio Pittorino, Gabriele Perugini, and Riccardo Zecchina. *Deep learning via message passing algorithms based on belief propagation*. 2021. arXiv: 2110.14583.

[144] John Hopfield. *Neural Networks and Physical Systems with Emergent Collective Computational Abilities*. May 1982. DOI: 10.1073/pnas.79.8.2554.

[145] D.O. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. A Wiley book in clinical psychology. Wiley, 1949. ISBN: 9780471367277. URL: https://books.google.it/books?id=dZ0eDiLTwuEC.

[146] Daniel J Amit, Hanoch Gutfreund, and H Sompolinsky. *Statistical mechanics of neural networks near saturation*. 1987. URL: https://www.sciencedirect.com/science/article/pii/0003491687900923.

[147] A. Crisanti, D. J. Amit, and H. Gutfreund. *Saturation level of the Hopfield model for neural network*. Aug. 1986. DOI: 10.1209/0295-5075/2/4/012.

[148] M. Negri, C. Lauditi, G. Perugini, C. Lucibello, and E. Malatesta. *Storage and Learning Phase Transitions in the Random-Features Hopfield Model*. 2023. URL: https://link.aps.org/doi/10.1103/PhysRevLett.131.257301.

[149] Matteo Negri, Clarissa Lauditi, Gabriele Perugini, Carlo Lucibello, and Enrico M. Malatesta. *Random Feature Hopfield Networks generalize retrieval to previously unseen examples*. 2023.

[150] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. *A fast learning algorithm for deep belief nets*. 2006.

[151] Geoffrey E Hinton and Ruslan R Salakhutdinov. *Reducing the dimensionality of data with neural networks*. 2006.

[152] Jérôme Tubiana and Rémi Monasson. *Emergence of compositional representations in restricted Boltzmann machines*. 2017.

[153] Dmitry Krotov and John J Hopfield. *Dense associative memory for pattern recognition*. 2016.

[154] Daniel J Amit, Hanoch Gutfreund, and Haim Sompolinsky. *Information storage in neural networks with low levels of activity*. 1987.

[155] Leo Kozachkov, Ksenia V. Kastanenka, and Dmitry Krotov. *Building transformers from neurons and astrocytes*. 2023. DOI: 10.1073/pnas.2219150120. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.2219150120.

[156] Riccardo Rende, Federica Gerace, Alessandro Laio, and Sebastian Goldt. *What does self-attention learn from Masked Language Modelling?* 2024. arXiv: 2304.07235.

[157] Itay Lavie, Guy Gur-Ari, and Zohar Ringel. *Towards Understanding Inductive Bias in Transformers: A View From Infinity*. 2024. arXiv: 2402.05173.

[158] Carlo Lucibello and Marc Mézard. *Exponential Capacity of Dense Associative Memories*. 2024. DOI: 10.1103/PhysRevLett.132.077301.

Acronyms

BP Belief Propagation

CLT Central Limit Theorem

CSP Constraint Satisfaction Problem

fBP focusing Belief Propagation

GD Gradient Descent

GEP Gaussian Equivalence Principle

HM Hopfield Model

IMP Iterative magnitude pruning

LAL Least Action Algorithm

LE Local entropy

LMC Linear mode connectivity

LTH Lottery Ticket Hyphotesis

MCT0 Zero-Temperature Monte Carlo

MLP Multi-layer Perceptron

OGP Overlap Gap Property

PA Perceptron Algorithm

RFHM Random Feature Hopfield Model

RFM Random Feature Model

RS Replica - Symmetric

RSB Replica Symmetry Breaking

SA Simulated Annealing

SAT Satisfiable

SGD Stochastic Gradient Descent

1RSB One-Step Replica Symmetry Breaking

TCM Tree Committee Machine

UNSAT Unsatisfiable

Appendices

Appendix A

Details on the RFM

A.1 Average over the disorder: GEP

In order to compute the first two moments of the teacher and student preactivations as in the setting with a binary Perceptron teacher and a binary RFM as student, we refer to Sec. 4.1 for the formal definition of the learning problem. When σ is linear, the average can be easily computed by using the integral representation of the Dirac delta function as in the classical models without (random) feature projections. When σ is non-linear the computation of the average is more involved. We can, however, compute the moments of the variables u^μ and λ_a^μ as in Eq. (4.8)

$$v^\mu = \frac{1}{\sqrt{D}} \sum_{k=1}^{D} \xi_k^\mu w_k^T, \quad \lambda_a^\mu = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^a \, \sigma \left(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k \right) \tag{A.1}$$

since in the large N, D, P limit (with $\alpha \equiv P/N$ and $\alpha_D \equiv D/N$ fixed), these are the moments of a multivariate Gaussian random variable

$$P\left(u^\mu, \{\lambda_a^\mu\} \,|F_{ki}, w_i^a\right) = \frac{1}{\sqrt{2\pi \det \Sigma}} e^{-\frac{1}{2} \sum_{a,b=0}^{n} \left(\Upsilon_a^\mu - \rho_a\right) \left(\Sigma^{-1}\right)_{ab} \left(\Upsilon_b^\mu - \rho_b\right)} \tag{A.2}$$

according to the GEP. In the expression (A.2) we have defined $\Upsilon_0^\mu \equiv u^\mu$, $\Upsilon_a^\mu \equiv \lambda_a^\mu$, $\forall a = 1, \dots, n$, the mean vectors as ρ_0, ρ_a and the covariance Σ_{ab}. We will compute explicitly the first two moments of the variables u^μ and λ_a^μ as in [99], and we will refer to [102] for the computation of the fourth moment satisfying Wick's theorem. The mean of u^μ is trivial

$$\mathbb{E}_{\boldsymbol{\xi}} \left[u^\mu\right] = 0 \tag{A.3}$$

whereas that of λ_a^μ reads

$$\begin{aligned} \mathbb{E}_{\boldsymbol{\xi}^\mu} \left[\lambda_a^\mu\right] &= \int \prod_{i=1}^{N} \frac{dv_i^\mu d\hat{v}_i^\mu}{2\pi} e^{i \sum_i \hat{v}_i^\mu v_i^\mu} \left[\frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^a \sigma\left(v_i^\mu\right) \right] \prod_k \mathbb{E}_{\xi_k^\mu} e^{-i \frac{\xi_k^\mu}{\sqrt{D}} \left(\sum_i \hat{v}_i^\mu F_{ki}\right)} \\ &= \int \prod_{i=1}^{N} \frac{dv_i^\mu d\hat{v}_i^\mu}{2\pi} e^{i \sum_i \hat{v}_i^\mu v_i^\mu} \left[\frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^a \sigma\left(v_i^\mu\right) \right] e^{-\frac{1}{2} \sum_{ij} \left(\frac{1}{D} \sum_k F_{ki} F_{kj} \right) \hat{v}_i^\mu \hat{v}_j^\mu}. \end{aligned} \tag{A.4}$$

By making use of the assumptions on the feature matrix (4.6) and the definitions in (4.12), the expression simplifies as

$$\mathbb{E}_{\boldsymbol{\xi}^\mu}\left[\lambda_a^\mu\right] = \frac{1}{\sqrt{N}}\sum_{i=1}^{N} w_i^a \int \frac{dv_\mu d\hat{v}_\mu}{2\pi} e^{i\hat{v}_\mu v_\mu}\sigma\left(v_\mu\right) e^{-\frac{\hat{v}_\mu^2}{2}} = \mu_0 \frac{1}{\sqrt{N}}\sum_i w_i^a \,. \tag{A.5}$$

The second moment of u^μ is instead

$$\mathbb{E}_{\boldsymbol{\xi}^\mu}\left[u_\mu^2\right] = \frac{1}{D}\sum_{k=1}^{D}\left(w_k^T\right)^2 = 1 \tag{A.6}$$

because of the binary nature of weights $w_k^T \in \{-1,+1\}^D$, while that of λ_a^μ is more involved

$$\mathbb{E}_{\boldsymbol{\xi}^\mu}\left[\lambda_\mu^a\lambda_\mu^b\right] = \int \prod_{i=1}^{N}\frac{dv_i^\mu d\hat{v}_i^\mu}{2\pi} e^{i\sum_i \hat{v}_i^\mu v_i^\mu}\left[\frac{1}{N}\sum_{ij} w_i^a w_j^b \sigma\left(v_i^\mu\right)\sigma\left(v_j^\mu\right)\right] e^{-\frac{1}{2}\sum_{ij}\left(\frac{1}{D}\sum_k F_{ki}F_{kj}\right)\hat{v}_i^\mu\hat{v}_j^\mu} \,. \tag{A.7}$$

By splitting $i=j$ and $i\neq j$ contributions, because of the assumptions (4.6), $\frac{1}{D}\sum_k F_{ki}F_{kj}$ with $i\neq j$ is of order $1/\sqrt{D}$ and we can therefore expand the exponential

$$\begin{aligned} e^{i\hat{v}_i^\mu v_i^\mu + i\hat{v}_j^\mu v_j^\mu - \left(\frac{1}{D}\sum_k F_{ki}F_{kj}\right)\hat{v}_i^\mu\hat{v}_j^\mu} &\simeq e^{i\hat{v}_i^\mu v_i^\mu + i\hat{v}_j^\mu v_j^\mu}\left[1-\frac{1}{2}\left(\frac{1}{D}\sum_k F_{ki}F_{kj}\right)\hat{v}_i^\mu\hat{v}_j^\mu\right] \\ &= \left[1+\left(\frac{1}{D}\sum_k F_{ki}F_{kj}\right)\frac{d}{dv_i^\mu}\frac{d}{dv_j^\mu}\right] e^{i\hat{v}_i^\mu v_i^\mu + i\hat{v}_j^\mu v_j^\mu} \end{aligned} \tag{A.8}$$

so that it becomes easy to perform the integrals

$$\begin{aligned} \mathbb{E}_{\boldsymbol{\xi}^\mu}\left[\lambda_\mu^a\lambda_\mu^b\right] &= \mu_2\frac{1}{N}\sum_i w_i^a w_i^b + \frac{1}{N}\sum_{i\neq j} w_i^a w_j^b \int Dv_i^\mu Dv_j^\mu \sigma\left(v_i^\mu\right)\sigma\left(v_j^\mu\right)\left[1+\left(\frac{1}{D}\sum_k F_{ki}F_{kj}\right)v_i^\mu v_j^\mu\right] \\ &= \left(\mu_2-\mu_1^2-\mu_0^2\right)\frac{1}{N}\sum_i w_i^a w_i^b + \frac{\mu_0^2}{N}\sum_i w_i^a\sum_j w_j^b + \mu_1^2\frac{1}{D}\sum_{k=1}^{D} s_k^a s_k^b \end{aligned} \tag{A.9}$$

having defined the "projected" weights in the teacher space s_k^a as

$$s_k^a \equiv \frac{1}{\sqrt{N}}\sum_{i=1}^{N} F_{ki} w_i^a \,. \tag{A.10}$$

The covariance is therefore

$$\mathbb{E}_{\boldsymbol{\xi}^\mu}\left[\lambda_\mu^a\lambda_\mu^b\right] - \mathbb{E}_{\boldsymbol{\xi}^\mu}\left[\lambda_\mu^a\right]\mathbb{E}_{\boldsymbol{\xi}^\mu}\left[\lambda_\mu^b\right] = \mu_\star^2\frac{1}{N}\sum_i w_i^a w_i^b + \mu_1^2\frac{1}{D}\sum_{k=1}^{D} s_k^a s_k^b \,. \tag{A.11}$$

We can also define the projection of the teacher weights in the student space as

$$s_i^T \equiv \frac{1}{D}\sum_{k=1}^{D} F_{ki} w_k^T \,. \tag{A.12}$$

Using again assumptions (4.6), we get for the cross term

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\xi}^\mu}\left[u_\mu \lambda_\mu^a\right] &= \int \prod_i \frac{dv_i^\mu d\hat{v}_i^\mu}{2\pi}\frac{du_\mu d\hat{u}_\mu}{2\pi} e^{i\sum_i \hat{v}_i^\mu v_i^\mu + i\hat{u}_\mu u_\mu}\left[\frac{u_\mu}{\sqrt{N}}\sum_i w_i^a \sigma\left(v_i^\mu\right)\right] e^{-\frac{\hat{u}_\mu^2}{2}} \\
&\quad \times e^{-\frac{1}{2}\sum_{ij}\left(\frac{1}{D}\sum_k F_{ki}F_{kj}\right)\hat{v}_i^\mu \hat{v}_j^\mu - \hat{u}_\mu \sum_i \hat{v}_i^\mu s_i^T} \\
&= \frac{1}{\sqrt{N}}\sum_i w_i^a \int \frac{dv_\mu d\hat{v}_\mu}{2\pi}\frac{du_\mu d\hat{u}_\mu}{2\pi} e^{i\hat{v}_\mu v_\mu + i\hat{u}_\mu u_\mu}\left[u_\mu \sigma\left(v_\mu\right)\right] e^{-\frac{\hat{u}_\mu^2}{2} - \frac{\hat{v}_\mu^2}{2} - \hat{u}_\mu \hat{v}_\mu s_i^T} \\
&= \mu_1 \frac{1}{\sqrt{N}}\sum_i w_i^a s_i^T = \mu_1 \frac{1}{D}\sum_{k=1}^{D} s_k^a w_k^T \,.
\end{aligned} \tag{A.13}$$

The covariance of random variables u^μ and λ_a^μ can be then written as

$$\Sigma \equiv \begin{pmatrix} 1 & M_a \\ M_a & Q_{ab} \end{pmatrix} \tag{A.14}$$

where

$$M_a = \mu_1 \frac{1}{D}\sum_{k=1}^{D} s_k^a w_k^T \equiv \mu_1 r_a \tag{A.15a}$$

$$Q_{ab} = \mu_\star^2 \frac{1}{N}\sum_{i=1}^{N} w_i^a w_i^b + \mu_1^2 \frac{1}{D}\sum_{k=1}^{D} s_k^a s_k^b \equiv \mu_\star^2 q_{ab} + \mu_1^2 p_{ab} \tag{A.15b}$$

being r_a the overlap between the teacher and the projection of the student weights in the D dimension, q_{ab} the usual overlap between pairs of different replicas of the student, and p_{ab} the overlap between a pair of solutions projected in the teacher space.

A.2 Equilibrium configuration limits

A.2.1 Storage case limit

Performing the limit $\alpha_D \to 0$ once the RS approximation has been replaced in the free energy expression (see some standard steps of the computation in 4.2.1), means to find the simplification of the order parameters as

$$\begin{aligned}
q &= 1 - 2\frac{\partial \mathcal{G}_{SS}}{\partial \hat{q}}\,, \quad p = q\,, \quad p_d = 1\,, \quad r = 0\,, \\
\hat{q} &= -2\alpha \frac{\partial \mathcal{G}_E}{\partial q}\,, \quad \hat{p} = 0\,, \quad \hat{p}_d = 0\,, \quad \hat{r} = 0
\end{aligned} \tag{A.16}$$

therefore giving $Q = (\mu_\star^2 + \mu_1^2)\, q$, $Q_d = \mu_\star^2 + \mu_1^2$. As a consequence, the free energy in Eq. (4.30) reduces to the free entropy of a storage problem as derived by Gardner in [107, 108]. Notice that this limit is reached independently of the non-linearity used $\sigma(\cdot)$.

A.2.2 Overparameterization regime

A.3 Bayesian generalization error

The Bayesian generalization error, whose definition

$$\epsilon_g^B = \int \frac{dx d\hat{x}}{2\pi}\, e^{ix\hat{x}}\, \mathbb{E}_{\boldsymbol{\xi}^\star} \Theta\left[-\left(\frac{1}{\sqrt{D}} \sum_k \xi_k^\star \right) x \right] \overline{e^{-i\hat{x}\langle y^\star \rangle_{\boldsymbol{w}|\{\boldsymbol{\xi}^\mu\}}}} \tag{A.17}$$

can be reported from Eq. (4.37), it is easy to compute by starting from the average of the student output over the posterior distribution. Using $a,\, b \in [n]$ for $Z^{-1} = \lim_{n\to 0} Z^{n-1}$ and $l,\, m \in [s]$ for the replicated expectation in Eq. (4.25), the definition (4.38) simplifies as

$$\begin{aligned}
\langle y^\star \rangle^s_{\boldsymbol{w}|\{\boldsymbol{\xi}^\mu\}} &= \lim_{n\to 0} \Big[\int \prod_{ia} dW_i^a\, P_{\boldsymbol{W}}(W^a)\, \mathrm{sign}\Big(\frac{1}{\sqrt{N}} \sum_i W_i^1 \sigma\Big(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k^\star \Big) \Big) \\
&\quad \times e^{-\beta \sum_{\mu a} \Theta\left(-\mathrm{sign}(\frac{1}{\sqrt{D}} \sum_k \xi_k^\mu)(\frac{1}{\sqrt{N}} \sum_i W_i^a \sigma\left(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k^\mu \right)) + \kappa \right)} \Big]^s \\
&= \lim_{n\to 0} \int \prod_l \frac{dh_l d\hat{h}_l}{2\pi} e^{i \sum_l h_l \hat{h}_l} \prod_l \mathrm{sign}(h_l) \int \prod_{ial} dw_i^{la} P_w(\boldsymbol{w}^{la}) \\
&\quad \times e^{-\beta \sum_{\mu a} \Theta\left[-\mathrm{sign}\left(\frac{1}{\sqrt{D}} \sum_k \xi_k^\mu \right) \left(\frac{1}{\sqrt{N}} \sum_i w_i^{la} \sigma\left(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k^\mu \right) \right) \right] - \frac{i}{\sqrt{N}} \sum_l \hat{h}_l \sum_i w_i^{l1} \sigma\left(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k^\star \right)}.
\end{aligned} \tag{A.18}$$

The average over patterns and features is now straightforward to compute using central limit theorem (CLT)

$$\begin{aligned}
\overline{\langle \hat{y}^\star \rangle^s_{\mathbf{w}|\boldsymbol{\xi}^\mu}} &= \lim_{n\to 0} \mathbb{E}_{\mathcal{D}} \int \prod_l \frac{dh_l d\hat{h}_l}{2\pi} e^{i \sum_l h_l \hat{h}_l} \prod_l \mathrm{sign}(h_l) \int \prod_{ial} dW_i^{al}\, P_{\boldsymbol{W}}(W^{al}) \\
&\quad \times e^{-\beta \sum_{\mu al} \Theta\left(-\mathrm{sign}(\frac{1}{\sqrt{D}} \sum_k \xi_k^\mu)(\frac{1}{\sqrt{N}} \sum_i W_i^{al} \sigma\left(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k^\mu \right)) + \kappa \right) - i \sum_l \hat{h}_l \left(\frac{1}{\sqrt{N}} \sum_i W_i^{1l} \sigma\left(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k^\star \right) \right)} \\
&= \lim_{n\to 0} \mathbb{E}_F \int \prod_l \frac{dh_l d\hat{h}_l}{2\pi} e^{i \sum_l h_l \hat{h}_l} \prod_l \mathrm{sign}(h_l) \int \prod_{ial} dW_i^{al}\, P_{\boldsymbol{W}}(W^{al}) \int \prod_\mu \frac{du^\mu d\hat{u}^\mu}{2\pi} \int \prod_{\mu al} \frac{d\lambda_{al}^\mu d\hat{\lambda}_{al}^\mu}{2\pi} \\
&\quad \times e^{-\beta_{\mu al} \Theta\left(-\mathrm{sign}(u^\mu) \lambda_{al}^\mu + \kappa \right)} e^{i \sum_\mu u^\mu \hat{u}^\mu + i \sum_{\mu al} \lambda_{al}^\mu \hat{\lambda}_{al}^\mu - \frac{1}{2} \sum_\mu (\hat{u}^\mu)^2 - \frac{1}{2} \sum_{al,bm} Q_{lm}^{ab} \hat{\lambda}_{al}^\mu \hat{\lambda}_{bm}^\mu - \sum_{\mu al} M_l^a \hat{u}^\mu \hat{\lambda}_{al}^\mu} \\
&= \lim_{n\to 0} \int \prod_l \frac{dh_l d\hat{h}_l}{2\pi} e^{i \sum_l h_l \hat{h}_l - \frac{\mu_\star^2}{2} \sum_{q_{11}^{lm}} \hat{h}_l \hat{h}_m l m} \prod_l \mathrm{sign}(h_l) \int \prod_{\substack{a<b \\ lm}} \frac{dq_{ab}^{lm} d\hat{q}_{ab}^{lm}}{2\pi} \prod_{\substack{a\le b \\ lm}} \frac{dp_{ab}^{lm} d\hat{p}_{ab}^{lm}}{2\pi} \prod_{al} \frac{dr_a^l d\hat{r}_a^l}{2\pi} e^{N\phi'}
\end{aligned} \tag{A.19}$$

having defined

$$\phi' = -\sum_{a<b}\sum_{lm} q_{ab}^{lm}\hat{q}_{ab}^{lm} - \frac{\alpha_D}{2}\sum_{ab} p_{ab}^{lm}\hat{p}_{ab}^{lm} - \alpha_D \sum_a r_a^l \hat{r}_a^l + G_{SS} + \alpha_D\, G'_{SE} + \alpha\, G_E \quad \text{(A.20)}$$

and with

$$G_{SS} = \ln \int \prod_{al} dw_{la}\, P_w(w_{la})\, e^{\frac{1}{2}\sum_{a\neq b}\hat{q}_{ab}^{lm} w_{la} w_{mb}} \quad \text{(A.21a)}$$

$$G'_{SE} = \frac{1}{D}\sum_k \ln \int \prod_{al} \frac{ds_k^{la} d\hat{s}_k^{la}}{2\pi}\, e^{i\sum_{la} s_k^{la}\hat{s}_k^{la} + \sum_{la}\hat{r}_a^l s_k^{la} + \frac{1}{2}\sum_{ab}\sum_{lm}\hat{p}_{ab}^{lm} s_k^{la} s_k^{mb} - \frac{1}{2}\sum_{ab}\sum_{lm} q_{ab}^{lm}\hat{s}_k^{la}\hat{s}_k^{mb} - i\frac{\mu_1 \xi_k^\star}{\sqrt{D}}\sum_l \hat{h}_l s_k^{l1}} \quad \text{(A.21b)}$$

$$G_E = \ln \int \prod_{la} \frac{d\lambda_{la} d\hat{\lambda}_{la}}{2\pi}\frac{du d\hat{u}}{2\pi}\, e^{iu\hat{u} + i\sum_{la}\lambda_{la}\hat{\lambda}_{la} - \beta\sum_{la}\Theta(-u\lambda_{la}) - \frac{\hat{u}^2}{2} - \frac{1}{2}\sum_{ab}\sum_{lm} Q_{ab}^{lm}\hat{\lambda}_{la}\hat{\lambda}_{mb} - \hat{u}\sum_{la} M_a^l \hat{\lambda}_{la}} \quad \text{(A.21c)}$$

once the pre-activations definition have been extracted as usual.

A.3.1 Entropic - Energetic term

Apart for the different numbers of replicas, G_{SS} and G_E have the same expression as before (see equations (4.26)), while the entropic-energetic part has an additional term that depends on the test pattern $\boldsymbol{\xi}^\star$ and can therefore be isolated with a prime index

$$G'_{SE} = G_{SE} + \frac{1}{D}\delta G_{SE} \quad \text{(A.22)}$$

where G_{SE} is defined as in (4.26c) and

$$\delta G_{SE} \equiv \sum_k \ln \langle\langle e^{-\frac{\mu_1 \xi_k^\star}{\sqrt{D}}\sum_l \hat{h}_l s_k^{l1}} \rangle\rangle_k \quad \text{(A.23a)}$$

$$\langle\langle \bullet \rangle\rangle_k \equiv \frac{\displaystyle\int \prod_{al} \frac{ds_k^{la} d\hat{s}_k^{la}}{2\pi}\, e^{i\sum_{la} s_k^{la}\hat{s}_k^{la} + \sum_{la}\hat{r}_a^l s_k^{la} + \frac{1}{2}\sum_{ab}\sum_{lm}\hat{p}_{ab}^{lm} s_k^{la} s_k^{mb} - \frac{1}{2}\sum_{ab}\sum_{lm} q_{ab}^{lm}\hat{s}_k^{la}\hat{s}_k^{mb}} \bullet}{\displaystyle\int \prod_{al} \frac{ds_k^{la} d\hat{s}_k^{la}}{2\pi}\, e^{i\sum_{la} s_k^{la}\hat{s}_k^{la} + \sum_{la}\hat{r}_a^l s_k^{la} + \frac{1}{2}\sum_{ab}\sum_{lm}\hat{p}_{ab}^{lm} s_k^{la} s_k^{mb} - \frac{1}{2}\sum_{ab}\sum_{lm} q_{ab}^{lm}\hat{s}_k^{la}\hat{s}_k^{mb}}}. \quad \text{(A.23b)}$$

Given that at first order in D, G'_{SE} is equal to G_{SE}, we have the same saddle point equations for every ansatz over replicas, as it should be. Plus, we can expand for large D the equation (A.23a)

$$\delta G_{SE} = \ln \prod_k \left[1 - i\frac{\mu_1}{\sqrt{D}}\xi_k^\star \sum_l \langle\langle s_k^{l1}\rangle\rangle_k \hat{h}_l - \frac{\mu_1^2}{2D}(\xi_k^\star)^2 \sum_{lm} \langle\langle s_k^{l1} s_k^{m1}\rangle\rangle_k \hat{h}_l \hat{h}_m \right]. \quad \text{(A.24)}$$

As can be seen from Eq. (A.21), the measure $\langle\langle \bullet \rangle\rangle_k$ is related to the derivatives of G_{SE}; therefore, we can use the saddle point equations and substitute the complicated integral expression with corresponding order parameters

$$s_k^{la}{}_k = \frac{\partial G_{SE}}{\partial \hat{r}_a^l} = r_a^l \tag{A.25a}$$

$$s_k^{la}\, s_k^{mb}{}_k = \frac{\partial G_{SE}}{\partial \hat{p}_{ab}^{lm}} = p_{ab}^{lm}. \tag{A.25b}$$

Notice how the right-hand expressions do not depend on k anymore, since G_{SE} is factorized over this index. We obtain

$$\begin{aligned}\delta G_{SE} &= \ln \prod_k \left[1 - i\frac{\mu_1}{\sqrt{D}}\xi_k^\star \sum_l r_1^l \hat{h}_l - \frac{\mu_1^2}{2D}(\xi_k^\star)^2 \sum_{lm} p_{11}^{lm}\hat{h}_l\hat{h}_m\right] \\ &\simeq -i\mu_1\left(\frac{1}{\sqrt{D}}\sum_k \xi_k^\star\right)\sum_l r_1^l\hat{h}_l - \frac{\mu_1^2}{2}\left(\frac{1}{D}\sum_k(\xi_k^\star)^2\right)\sum_{lm}\left(p_{11}^{lm} - r_1^l r_1^m\right)\hat{h}_l\hat{h}_m.\end{aligned} \tag{A.26}$$

RS - Ansatz

Assuming a RS ansatz, we get

$$\begin{aligned}\delta G_{SE} \simeq -i\mu_1\left(\frac{1}{\sqrt{D}}\sum_k \xi_k^\star\right) r\sum_l \hat{h}_l - \frac{\mu_1^2}{2}\left(\frac{1}{D}\sum_k(\xi_k^\star)^2\right) \\ \times\left[(p-r^2)\left(\sum_l \hat{h}_l\right)^2 + (p_d - p)\sum_l \hat{h}_l^2\right]\end{aligned} \tag{A.27}$$

so that, coming back to (A.19) we have

$$\begin{aligned}\overline{\langle \hat{y}^\star\rangle_{\mathbf{w}|\boldsymbol{\xi}^\mu}^s} &= \int \prod_l \frac{dh_l d\hat{h}_l}{2\pi} e^{i\sum_l h_l\hat{h}_l - \frac{\mu_\star^2}{2}q\left(\sum_l \hat{h}_l\right)^2 - \frac{\mu_\star^2}{2}(1-q)\sum_l(\hat{h}_l)^2}\prod_l \mathrm{sign}(h_l) \\ &\times e^{-i\mu_1\left(\frac{1}{\sqrt{D}}\sum_k \xi_k^\star\right) r\sum_l \hat{h}_l - \frac{\mu_1^2}{2}\left(\frac{1}{D}\sum_k(\xi_k^\star)^2\right)\left[(p-r^2)\left(\sum_l \hat{h}_l\right)^2 + (p_d-p)\sum_l(\hat{h}_l)^2\right]} \\ &= \int Dx\left[\int Dh\,\mathrm{sign}\left(\sqrt{\mu_\star^2(1-q) + \mu_1^2\sigma_\xi^\star(p_d-p)}h + \mu_1 m_\xi^\star r + \sqrt{\left(\mu_\star^2 q + \mu_1^2\sigma_\xi^\star(p-r^2)\right)}x\right)\right]^s \\ &= \int Dx\,\mathrm{erf}\left(\frac{\mu_1 m_\xi^\star r + \sqrt{\left(\mu_\star^2 q + \mu_1^2\sigma_\xi^\star(p-r^2)\right)}x}{\sqrt{2\mu_\star^2(1-q) + 2\mu_1^2\sigma_\xi^\star(p_d-p)}}\right)^s\end{aligned} \tag{A.28}$$

having defined

$$m_\xi^\star \equiv \frac{1}{\sqrt{D}} \sum_k \xi_k^\star \,, \tag{A.29a}$$

$$\sigma_\xi^\star \equiv \frac{1}{D} \sum_k (\xi_k^\star)^2 \,. \tag{A.29b}$$

Inserting back this expression into Eq. (A.17), we finally find the expression for the Bayesian generalization error

$$\begin{aligned}
\epsilon_g^B &= \mathbb{E}_{\xi^\star} \int Dx\, \Theta\Bigg(- \operatorname{sign}\Big(\frac{1}{\sqrt{D}} \sum_k \xi_k^\star\Big) \operatorname{erf}\Bigg(\frac{\mu_1 m_\star r + \sqrt{\Big(\mu_\star^2 q + \mu_1^2 \sigma_\star (p - r^2)\Big)} x}{\sqrt{2\mu_\star^2(1-q) + 2\mu_1^2 \sigma_\star (p_d - p)}} \Bigg) \Bigg) \\
&= \int Dx \int \frac{du\, d\hat{u}}{2\pi}\, e^{i\hat{u}u} e^{\sum_k \ln \cos(\frac{\hat{u}}{\sqrt{D}})} \Theta\Bigg(- \operatorname{sign}(u) \operatorname{erf}\Bigg(\frac{\mu_1 u r + \sqrt{\Big(\mu_\star^2 q + \mu_1^2 (p - r^2)\Big)} x}{\sqrt{2\mu_\star^2(1-q) + 2\mu_1^2 (p_d - p)}} \Bigg) \Bigg) \\
&= 2 \int Dx \int Du\, \Theta(u)\, \Theta\Bigg(- \operatorname{erf}\Bigg(\frac{\mu_1 u r + \sqrt{\Big(\mu_\star^2 q + \mu_1^2 (p - r^2)\Big)} x}{\sqrt{2\mu_\star^2(1-q) + 2\mu_1^2 (p_d - p)}} \Bigg) \Bigg) \\
&= \frac{1}{\pi} \arccos\left(\frac{M}{\sqrt{Q}} \right).
\end{aligned} \tag{A.30}$$

Notice that, as mentioned in the main text, if we would like to compute the generalization error of the barycenter of typical solutions with a given margin κ, the formula above would remain the same, since the only dependence on the margin is implicit in the order parameters q, p and r.

The behaviour of the generalization error of the barycenter of typical solutions with vanishing and non-vanishing margins can be found in Fig. 4.3 when it is optimized as a function of κ. Indeed, for every value of α one can find the best performance by looking for the minimum Bayesian error at κ_{opt}. It turns out that, as soon as $\alpha > \alpha^\star$, $\kappa_{\text{opt}} = 0$ while is becomes larger than zero when increasing the overparameterization degree and $\alpha < \alpha^*$.

A.4 Stability distribution

The stability or *aligning field* to an input $\boldsymbol{\xi}^\mu$ and its corresponding label y^μ can be defined as

$$\Delta^\mu \equiv y^\mu \lambda^\mu = \frac{y^\mu}{\sqrt{N}} \sum_{i=1}^{N} w_i \sigma \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki} \xi_k^\mu \right) \tag{A.31}$$

which is greater than zero for each μ-pattern if the configuration $\boldsymbol{w}$ minimizes the error over the dataset $\mathcal{D}$. Notice that in the present case we reduce to $\kappa = 0$ for

simplicity. The stability distribution then reads

$$
\begin{aligned}
P(\Delta) &= \mathbb{E}_{\mathcal{D}}\langle \delta\Big(\Delta - \frac{y^1}{\sqrt{N}}\sum_i W_i \sigma\Big(\frac{1}{\sqrt{D}}\sum_k F_{ki}\xi_k^1\Big)\Big)\rangle \\
&= \mathbb{E}_{\mathcal{D}}\frac{1}{Z}\int d(W)\, P(W)\, e^{-\beta\sum_\mu \ell_{NE}\Big(\text{sign}\big(\frac{1}{\sqrt{D}}\sum_{k=1}^D \xi_k^\mu w_k^T\big)\big(\frac{1}{\sqrt{N}}\sum_i \sigma\,\big(\frac{1}{\sqrt{D}}\sum_k F_{ki}\xi_k^\mu\big) w_i^a\Big) - \kappa\Big)} \\
&\quad \times \delta\Big(\Big(\Delta - \frac{y^1}{\sqrt{N}}\sum_i W_i\sigma\Big(\frac{1}{\sqrt{D}}\sum_k F_{ki}\xi_k^1\Big)\Big)\Big)
\end{aligned}
\tag{A.32}
$$

and since if one replicates the system saddle points are not affected by the delta in the large N limit, one can first compute the saddle points equations and then replace them in the stability distribution

$$
\begin{aligned}
P(\Delta) &= \lim_{n\to 0}\int \frac{du\, d\hat{u}}{2\pi}\int\prod_a \frac{d\lambda^a d\hat{\lambda}^a}{2\pi} e^{i\hat{u}\left(u + iM\sum_a \hat{\lambda}^a\right) + i\sum_a \lambda^a\hat{\lambda}^a - \frac{\hat{u}^2}{2} - \frac{Q}{2}\left(\sum_a \hat{\lambda}^a\right)^2 - \frac{Q_d - Q}{2}\sum_a (\hat{\lambda}^a)^2} \\
&\quad \times e^{-\beta\sum_a \ell_{NE}(\text{sign}(u)\lambda^a - \kappa)}\,\delta\Big(\Delta - \text{sign}(u)\lambda_1\Big) \\
&= \frac{2\, e^{-\beta\ell_{NE}(\Delta-\kappa)}}{\sqrt{Q_d - Q}}\int Dx\, \frac{H\Big(-\frac{M}{\sqrt{Q(1+\eta^2)-M^2}}x\Big)\, G\Big(\frac{\Delta - \sqrt{Q}x}{\sqrt{Q_d - Q}}\Big)}{\int D\lambda\, e^{-\beta\ell_{NE}\Big(\left(\sqrt{Q_d-Q}\lambda + \sqrt{Q}x\right) - \kappa\Big)}} \\
&= \frac{2\, e^{-\beta\Theta(\kappa-\Delta)}}{\sqrt{Q_d - Q}}\int Dx\, H\Big(-\frac{M}{\sqrt{Q(1+\eta^2) - M^2}}x\Big)\,\frac{G\Big(\frac{\Delta - \sqrt{Q}x}{\sqrt{Q_d - Q}}\Big)}{H_\beta\Big(\frac{\kappa - \sqrt{Q}x}{\sqrt{Q_d - Q}}\Big)}
\end{aligned}
\tag{A.33}
$$

where in the last line the Θ loss expression has been made explicit and $G(x) \equiv \frac{e^{-x^2/2}}{\sqrt{2\pi}}$.

In Fig. A.1 we present the stability distribution for typical solutions across various values of α_T and compared with MCT0 simulations. As we mentioned in the main text (section 4.2.3), since MCT0 samples from Gibbs distribution, the agreement with replica theory is very good. For each of the cases there exists a sharp threshold value of stability Δ that exactly corresponds to the threshold imposed by the margin $\kappa = 0$. Notably, the maximum mass of the distribution also appears to be near the origin, especially for low values of α_T. This pattern aligns with observations made in previous studies about *sharp* solutions in both one-layer [106] and two-layer neural networks [47], where the maximum at the threshold means the kind of solutions we are describing have a low level or stability against pattern perturbations. In contrast, solutions described as *flat* or having high local entropy typically exhibit a low likelihood of possessing small stability. The trend indicates that focusing on regions with high local entropy promotes a soft margin.

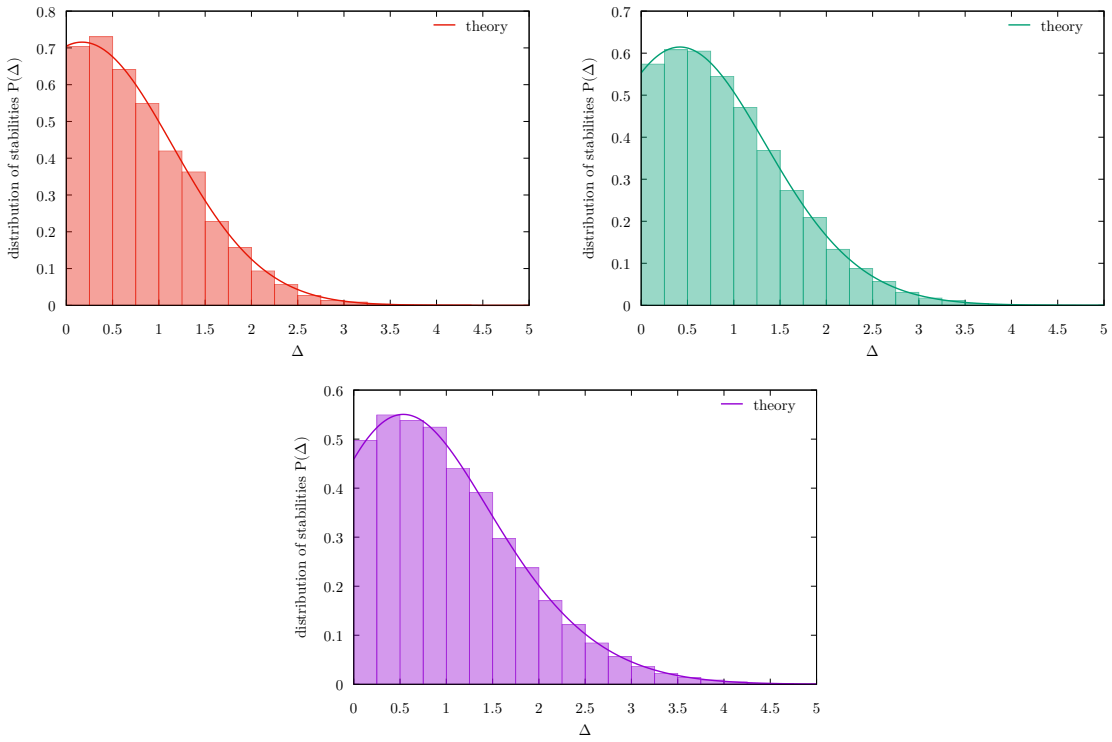


Figure A.1: Histograms of the distribution of stabilities for MCT0 solutions at $\alpha_T = 1.4$ (top left), $\alpha_T = 5$ (top right) and $\alpha_T = 10$ (bottom). The full line is the replica prediction of the distribution for typical solutions in the thermodynamic limit given by equation (A.33). Reprinted from [99].

A.5 Franz-Parisi potential

To access the typical value of the averaged local entropy of a RFM that is a solution of a supervised learning task with teacher $y^\mu = \text{sign}\left(\frac{1}{\sqrt{D}}\sum_k w_k^T \xi_k^\mu\right)$, we need to compute the expression

$$\Phi_{FP}(t_1) = \frac{1}{Z}\int \prod_i d\tilde{w}_i \, P_w(\tilde{\boldsymbol{w}}) e^{-\tilde{\beta}\sum_\mu \ell_{NE}\left(-y^\mu \frac{1}{\sqrt{N}}\sum_i \tilde{w}_i \, \sigma\left(\frac{1}{\sqrt{D}}\sum_k F_{ki}\xi_k^\mu\right);\tilde{\kappa}\right)} \ln \mathcal{N}(\tilde{\boldsymbol{w}}, t_1) \tag{A.34}$$

where the local entropy of the reference configuration $\tilde{\boldsymbol{w}}$ is proportional to

$$\mathcal{N}(\tilde{\boldsymbol{w}}, t_1) \equiv \int \prod_i dw_i \, P_w(\boldsymbol{w}) e^{-\beta\sum_\mu \ell_{NE}\left(-y^\mu \frac{1}{\sqrt{N}}\sum_i w_i \, \sigma\left(\frac{1}{\sqrt{D}}\sum_k F_{ki}\xi_k^\mu\right);\kappa\right)} \delta\left(\sum_i w_i \tilde{w}_i - N t_1\right). \tag{A.35}$$

By introducing $a = 1, \dots, n$ unconstrained replicas $\tilde{\boldsymbol{w}}^a$ and $c = 1, \dots, s$ constrained replicas $\boldsymbol{w}^c$, if we enforce the definitions

$$u^\mu = \frac{1}{\sqrt{D}} \sum_k w_k^T \xi_k^\mu \tag{A.36a}$$

$$\tilde{\lambda}_a^\mu = \frac{1}{\sqrt{N}} \sum_i \tilde{w}_i^a \, \sigma\left(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k^\mu \right) \tag{A.36b}$$

$$\lambda_c^\mu = \frac{1}{\sqrt{N}} \sum_i w_i^c \, \sigma\left(\frac{1}{\sqrt{D}} \sum_k F_{ki} \xi_k^\mu \right) \tag{A.36c}$$

for the pre-activation of the teacher, the the references and the constrained solutions, we can make use of GEP in order to average over the random realization of patterns $\boldsymbol{\xi}^\mu$ in the latent space. The distribution of random variables $P(u^\mu, \{\tilde{\lambda}_a^\mu\}, \{\lambda_c^\mu\})$ in the $N \to \infty$ limit is a multivariate Gaussian of dimension $1 + n + s$. Therefore, the average gives

$$\begin{aligned}
\phi_{FP}(t_1) = & \lim_{s\to 0} \lim_{n \to 0} \frac{\partial}{\partial s} \mathbb{E}_F \int \prod_{\mu ac} \frac{du^\mu \, d\hat{u}^\mu}{2\pi} \frac{d\tilde{\lambda}_a^\mu \, d\hat{\tilde{\lambda}}_a^\mu}{2\pi} \frac{d\lambda_c^\mu \, d\hat{\lambda}_c^\mu}{2\pi} \prod_\mu e^{iu^\mu \hat{u}^\mu + i \sum_a \tilde{\lambda}_a^\mu \hat{\tilde{\lambda}}_a^\mu + i \sum_c \lambda_c^\mu \hat{\lambda}_c^\mu} \\
& \times \int \prod_{ia} d\tilde{w}_i^a \prod_{ic} dw_i^c \prod_a P_w(\tilde{\boldsymbol{w}}) \prod_c P_w(\boldsymbol{w}) \prod_c \delta\left(\sum_i w_i \tilde{w}_i - N t_1 \right) \\
& \times \prod_\mu e^{-\tilde{\beta} \sum_a \ell_{NE}\left(-y^\mu \tilde{\lambda}_a^\mu; \tilde{\kappa}\right) - \beta \sum_c \ell_{NE}\left(-y^\mu \lambda_c^\mu; \kappa\right) - \frac{(\hat{u}^\mu)^2}{2} - \frac{1}{2} \sum_{ab} Q_{ab} \hat{\tilde{\lambda}}_a^\mu \hat{\tilde{\lambda}}_b^\mu - \frac{1}{2} \sum_{cd} Q_{cd} \hat{\lambda}_c^\mu \hat{\lambda}_d^\mu} \\
& \times e^{-\hat{u}^\mu \sum_a M_a \hat{\tilde{\lambda}}_a^\mu - \hat{u}^\mu \sum_c M_c \hat{\lambda}_c^\mu}
\end{aligned} \tag{A.37}$$

with covariance

$$\Sigma = \begin{pmatrix} \langle u^\mu u^\mu \rangle & \langle u^\mu \tilde{\lambda}_a^\mu \rangle & \langle u^\mu \lambda_c^\mu \rangle \\ \langle \tilde{\lambda}_a^\mu u^\mu \rangle & \langle \tilde{\lambda}_a^\mu \tilde{\lambda}_b^\mu \rangle & \langle \tilde{\lambda}_a^\mu \lambda_c^\mu \rangle \\ \langle \lambda_c^\mu u^\mu \rangle & \langle \lambda_c^\mu \tilde{\lambda}_a^\mu \rangle & \langle \lambda_c^\mu \lambda_d^\mu \rangle \end{pmatrix} = \begin{pmatrix} 1 & \tilde{M}_a & M_c \\ \tilde{M}_a & \tilde{Q}_{ab} & T_{ac} \\ M_c & T_{ac} & Q_{cd} \end{pmatrix}$$

and definitions

$$\langle u^\mu u^\mu \rangle = \frac{1}{D} \sum_{k,h=1}^{D} w_k^T w_h^T \langle \xi_k^\mu \xi_h^\mu \rangle = 1 \tag{A.38a}$$

$$\langle u^\mu \tilde{\lambda}_a^\mu \rangle = \frac{\mu_1}{D\sqrt{N}} \sum_{k=1}^{D} \sum_{i=1}^{N} w_k^T F_{ki} \tilde{w}_i^a = \frac{\mu_1}{D} \sum_k \tilde{s}_k^a w_k^T = \mu_1 \tilde{r}_a = \tilde{M}_a \tag{A.38b}$$

$$\langle u^\mu \lambda_c^\mu \rangle = \frac{\mu_1}{D\sqrt{N}} \sum_{k=1}^{D} \sum_{i=1}^{N} w_k^T F_{ki} w_i^c = \frac{\mu_1}{D} \sum_k s_k^c w_k^T = \mu_1 r_c = M_c \tag{A.39a}$$

$$\langle \tilde{\lambda}_a^\mu \tilde{\lambda}_b^\mu \rangle = \frac{\mu_1^2}{D} \sum_{k=1}^{D} \tilde{s}_k^a \tilde{s}_k^b \; + \; \frac{\mu_*^2}{N} \sum_{i=1}^{N} \tilde{w}_i^a \tilde{w}_i^b = \mu_1^2 \tilde{p}_{ab} \; + \; \mu_*^2 \tilde{q}_{ab} = \tilde{Q}_{ab} \tag{A.39b}$$

$$\langle \tilde{\lambda}_a^\mu \lambda_c^\mu \rangle = \frac{\mu_1^2}{D} \sum_{k=1}^{D} \tilde{s}_k^a s_k^c \; + \; \frac{\mu_*^2}{N} \sum_{i=1}^{N} \tilde{w}_i^a w_i^c = \mu_1^2 k_{ac} \; + \; \mu_*^2 t_{ac} = T_{ac} \tag{A.39c}$$

$$\langle \lambda_c^\mu \lambda_d^\mu \rangle = \frac{\mu_1^2}{D} \sum_{k=1}^{D} s_k^c s_k^d \; + \; \frac{\mu_*^2}{N} \sum_{i=1}^{N} w_i^c w_i^d = \mu_1^2 p_{cd} \; + \; \mu_*^2 q_{cd} = Q_{cd} \tag{A.39d}$$

Substituting all definitions by use of delta functions the expression simplifies

$$\begin{aligned} \phi_{FP} = & - \sum_{a<b} \tilde{q}_{ab} \hat{\tilde{q}}_{ab} - \frac{\alpha_D}{2} \sum_{ab} \tilde{p}_{ab} \hat{\tilde{p}}_{ab} - \alpha_D \sum_a \tilde{r}_a \hat{\tilde{r}}_a - \sum_{c<d} q_{cd} \hat{q}_{cd} - \frac{\alpha_D}{2} \sum_{cd} p_{cd} \hat{p}_{cd} \\ & - \alpha_D \sum_c r_c \hat{r}_c - \alpha_D \sum_{ac} k_{ac} \hat{k}_{ac} - \sum_{ac} t_{ac} \hat{t}_{ac} + G_{SS} + \alpha_D \, G_{SE} + \alpha \, G_E \end{aligned} \tag{A.40}$$

collecting entropic, energetic and entropic-energetic contributions as usual

$$G_{SS} = \ln \int \prod_a d\tilde{w}_a \, P_w(\tilde{w}_a) \int \prod_c dw_c \, P_w(w_c) \, e^{\frac{1}{2} \sum_{a \neq b} \hat{\tilde{q}}_{ab} \tilde{w}_a \tilde{w}_b - \frac{1}{2} \sum_{a \neq b} \hat{q}_{cd} w_c w_d + \sum_{ac} \hat{t}_{ac} \tilde{w}_a w_c} \tag{A.41a}$$

$$\begin{aligned} G_{SE} = \ln \int \prod_a \frac{d\tilde{s}_a d\hat{\tilde{s}}_a}{2\pi} \prod_c \frac{ds_c d\hat{s}_c}{2\pi} \, & e^{i \sum_a \tilde{s}_a \hat{\tilde{s}}_a + i \sum_c s_c \hat{s}_c + \sum_a \hat{\tilde{r}}_a \tilde{s}_a + \sum_c \hat{r}_c s_c + \frac{1}{2} \sum_{ab} \hat{\tilde{p}}_{ab} \tilde{s}_a \tilde{s}_b + \frac{1}{2} \sum_{cd} \hat{p}_{cd} s_c s_d} \\ & \times e^{-\frac{1}{2} \sum_{ab} \tilde{q}_{ab} \hat{\tilde{s}}_a \hat{\tilde{s}}_b - \frac{1}{2} \sum_{cd} q_{cd} \hat{s}_c \hat{s}_d + \sum_{ac} \hat{k}_{ac} \tilde{s}_a s_c - \sum_{ac} t_{ac} \hat{\tilde{s}}_a \hat{s}_c} \end{aligned} \tag{A.41b}$$

$$\begin{aligned} G_E = \ln \int \prod_a \frac{d\tilde{\lambda}_a d\hat{\tilde{\lambda}}_a}{2\pi} \prod_a \frac{d\lambda_c d\hat{\lambda}_c}{2\pi} \frac{du d\hat{u}}{2\pi} \, & e^{iu\hat{u} + i \sum_a \tilde{\lambda}_a \hat{\tilde{\lambda}}_a + i \sum_c \lambda_c \hat{\lambda}_c - \tilde{\beta} \sum_c \Theta\left(-\mathrm{sign}(u) \tilde{\lambda}_c + \tilde{\kappa}\right)} \\ & \times e^{-\beta \sum_a \Theta(-\mathrm{sign}(u)\lambda_a + \kappa) - \frac{\hat{u}^2}{2} - \frac{1}{2} \sum_{ab} \tilde{Q}_{ab} \hat{\tilde{\lambda}}_a \hat{\tilde{\lambda}}_b - \frac{1}{2} \sum_{cd} Q_{cd} \hat{\lambda}_c \hat{\lambda}_d - \hat{u} \sum_a \tilde{M}_a \hat{\tilde{\lambda}}_a - \hat{u} \sum_c M_c \hat{\lambda}_c - \sum_{ac} T_{ac} \hat{\tilde{\lambda}}_a \hat{\lambda}_c} . \end{aligned} \tag{A.41c}$$

Notice that, since the entropic-energetic term is Gaussian, it can be easily rewritten once the following quantities have been defined

$$\hat{\tilde{r}}_\alpha \equiv \left(\hat{\tilde{r}}_a, \hat{r}_c \right) \in \mathbb{R}^{n+s} \, , \tag{A.42a}$$

$$\overline{q}_{\alpha\beta} \equiv \begin{pmatrix} \tilde{q}_{ab} & t_{ac} \\ t_{ac} & q_{cd} \end{pmatrix} \in \mathbb{R}^{(n+s)\times(n+s)} \tag{A.42b}$$

$$\hat{\overline{p}}_{\alpha\beta} \equiv \begin{pmatrix} \hat{\tilde{p}}_{ab} & k_{ac} \\ k_{ac} & \hat{p}_{cd} \end{pmatrix} \in \mathbb{R}^{(n+s)\times(n+s)} \tag{A.42c}$$

so that the (A.41b) in terms of $\hat{\overline{r}}_\alpha$, $\overline{q}_{\alpha\beta}$ and $\hat{\overline{p}}_{\alpha\beta}$ becomes

$$G_{SE} = -\frac{1}{2}\ln\det\left(\mathbb{I} - \overline{q}\,\hat{\overline{p}}\right) + \frac{1}{2}\sum_{\alpha\beta}\hat{\overline{r}}_\alpha\left[\left(\mathbb{I} - \overline{q}\,\hat{\overline{p}}\right)^{-1}\overline{q}\right]_{\alpha\beta}\hat{\overline{r}}_\beta\,. \tag{A.43}$$

A.5.1 RS - Ansatz

By imposing the Replica-Symmetric Ansatz on the order parameters

$$\tilde{q}_{ab} = \delta_{ab} + \tilde{q}\,(1-\delta_{ab}) \qquad \hat{\tilde{q}}_{ab} = \hat{\tilde{q}}\,(1-\delta_{ab}) \tag{A.44a}$$
$$\tilde{p}_{ab} = \tilde{p}_d\,\delta_{ab} + \tilde{p}\,(1-\delta_{ab}) \qquad \hat{\tilde{p}}_{ab} = -\hat{\tilde{p}}_d\,\delta_{ab} + \hat{\tilde{p}}\,(1-\delta_{ab}) \tag{A.44b}$$
$$\tilde{r}_a = \tilde{r} \qquad \hat{\tilde{r}}_a = \hat{\tilde{r}} \tag{A.44c}$$
$$k_{ac} = k_1\,\delta_{a1} + k_0(1-\delta_{a1}) \qquad \hat{k}_{ac} = \hat{k}_1\,\delta_{a1} + \hat{k}_0(1-\delta_{a1}) \tag{A.44d}$$
$$t_{ac} = t_1\,\delta_{a1} + t_0(1-\delta_{a1}) \qquad \hat{t}_{ac} = \hat{t}_1\,\delta_{a1} + \hat{t}_0(1-\delta_{a1}) \tag{A.44e}$$

and the same for those of the constrained configurations, we can start be computing the entropic and energetic contributions.

Entropic - Entropic and Energetic terms

By decoupling the term $\sum_{ac}\hat{t}_{ac}\tilde{w}^a w^c$ according to the Ansatz

$$\sum_{ac}\hat{t}_{ac}\,\tilde{w}^a w^c = \frac{\hat{t}_0}{2}\left[(\sum_a \tilde{w}^a + \sum_c w^c)^2 - (\sum_a \tilde{w}^a)^2 - (\sum_c w^c)^2\right] + (\hat{t}_1 - \hat{t}_0)\,\tilde{w}^1\sum_c w^c \tag{A.45}$$

and the norms $\tilde{q}_{aa} = q_{cc} \equiv 1$ because of the prior on binary weights, the entropic term becomes

$$\begin{aligned}\mathcal{G}_{SS} &= \frac{\hat{q}}{2} + \lim_{s\to 0}\lim_{n\to 0}\partial_s G_{SS} \\ &= \int Dx \frac{\sum_{\tilde{w}=\pm 1} e^{\sqrt{\hat{\tilde{q}}}\tilde{w}x}\int Dy \ln 2\cosh\left(\sqrt{\hat{q} - \frac{\hat{t}_0^2}{\hat{\tilde{q}}}}\,y + \frac{\hat{t}_0}{\sqrt{\hat{\tilde{q}}}}x + (\hat{t}_1 - \hat{t}_0)\tilde{w}\right)}{2\cosh\left(\sqrt{\hat{\tilde{q}}}x\right)}.\end{aligned} \tag{A.46}$$

The energetic term is straightforward but more involved. It is however equal (apart for a redefinition of order parameters) to the energetic term that is obtained in a classic teacher-student setting

$$\mathcal{G}_E \equiv \lim_{\substack{n\to 0\\ s\to 0}}\partial_s G_E = 2\int DxDy\,\frac{H_{\tilde{\beta}}\left(u(x,y)\right)}{H_{\tilde{\beta}}\left(h(x)\right)}\int_{h(x)}^{\infty} Dz\,\ln H_\beta\left(v(x,y,z)\right) \tag{A.47}$$

being $H_\beta(x) \equiv e^{-\beta} + \left(1 - e^{-\beta}\right) H(x)$ with $H(x) \equiv \frac{1}{2}\mathrm{Erfc}\left(\frac{x}{\sqrt{2}}\right)$ and

$$u(x,y) \equiv \frac{\tilde{M}\sqrt{\Gamma}\,(by - ax) - M\sqrt{\tilde{Q}}y}{\sqrt{(\tilde{Q} - \tilde{M}^2)(\Gamma - M^2) - (T_0 - M\tilde{M})^2}} \tag{A.48a}$$

$$v(x,y,z) \equiv \frac{\kappa - \sqrt{\Gamma}(ay + bx) - \frac{T_1 - T_0}{\sqrt{\tilde{Q}_d - \tilde{Q}}}z}{\sqrt{Q_d - Q}} \tag{A.48b}$$

$$h(x) \equiv \frac{\tilde{\kappa} - \sqrt{\tilde{Q}}x}{\sqrt{\tilde{Q}_d - \tilde{Q}}} \tag{A.48c}$$

$$\Gamma \equiv Q - \frac{(T_1 - T_0)^2}{\tilde{Q}_d - \tilde{Q}} \tag{A.48d}$$

$$b \equiv \frac{T_0}{\sqrt{\tilde{Q}\Gamma}}, \qquad a \equiv \sqrt{1 - b^2}. \tag{A.48e}$$

Entropic - Energetic term

Following a series of algebraic manipulations, the entropic-energetic term reads

$$\begin{aligned}
\mathcal{G}_{SE} \equiv \lim_{\substack{n\to 0 \\ s\to 0}} \partial_s G_{SE} = &-\frac{1}{2}\ln\eta + \frac{1}{2\eta}\left[\left(\hat{p} + \hat{r}^2\right)(1-q) - (\hat{p}_d + \hat{p})\,q\right] \\
&+ \frac{1}{2\eta\tilde{\eta}}\left\{(1-q)(\hat{k}_1 - \hat{k}_0)^2\left[1 - q + \frac{1}{\tilde{\eta}}\left((1-\tilde{q})^2(\hat{\tilde{p}} + \hat{\tilde{r}}^2) + q\right)\right]\right\} \\
&+ \frac{1}{\eta\tilde{\eta}}\left\{(\hat{k}_1 - \hat{k}_0)\left[(1-q)(1-\tilde{q})(\hat{k}_0 + \hat{r}\hat{\tilde{r}}) + t_0\right]\right\} \\
&+ \frac{1}{2\eta\tilde{\eta}}\left\{(\hat{p}_d + \hat{p})(t_1 - t_0)^2\left[\hat{\tilde{p}}_d + \hat{\tilde{p}} - \frac{1}{\tilde{\eta}}\left(\hat{\tilde{p}} + \hat{\tilde{r}}^2 + (\hat{\tilde{p}}_d + \hat{\tilde{p}})^2\tilde{q}\right)\right]\right\} \\
&+ \frac{1}{\eta\tilde{\eta}}\left\{(t_1 - t_0)\left[(\hat{p}_d + \hat{p})\left(\hat{\tilde{p}}_d + \hat{\tilde{p}}\right)t_0 + \hat{k}_0 + \hat{r}\hat{\tilde{r}}\right]\right\} \\
&+ \frac{(\hat{k}_1 - \hat{k}_0)(t_1 - t_0)}{\eta\tilde{\eta}}\left[1 + \frac{1}{\tilde{\eta}}\left[(1-\tilde{q})\left(\hat{\tilde{p}} + \hat{\tilde{r}}^2\right) - (\hat{\tilde{p}}_d + \hat{\tilde{p}})\tilde{q}\right]\right]
\end{aligned} \tag{A.49}$$

where we have defined the quantities

$$\eta \equiv 1 + (\hat{p}_d + \hat{p})(1-q) \tag{A.50a}$$

$$\tilde{\eta} \equiv 1 + (\hat{\tilde{p}}_d + \hat{\tilde{p}})(1-\tilde{q}). \tag{A.50b}$$

Final expression of the potential

The typical free entropy of the system is in the end

$$\phi = -\frac{\hat{q}}{2}(1-q) + \frac{\alpha_D}{2}\left(p_d\hat{p}_d + p\hat{p}\right) - \alpha_D r\hat{r} + \mathcal{G}_{SS} + \alpha_D\,\mathcal{G}_{SE} + \alpha\,\mathcal{G}_E \tag{A.51}$$

where the order parameters q, p, p_d, r, $\hat{q}$, $\hat{p}$, $\hat{p}_d$, $\hat{r}$ are found by saddle point equations.

Appendix B

Details on the negative Perceptron

B.1 Typical overlap between pairs of solutions

In order to compute the typical overlap between pairs of solutions $\boldsymbol{W}^1$ and $\boldsymbol{W}^2$ independently sampled from (5.3) it is necessary to evaluate

$$p = \frac{1}{N}\mathbb{E}\langle \boldsymbol{W}^1 \cdot \boldsymbol{W}^2 \rangle_{k_1,k_2} \tag{B.1}$$

where $\langle \cdot \rangle_{k,k'}$ represents the average over the Cartesian product of densities (5.3) with the corresponding margins $k_{1,2}$, while $\mathbb{E}$ is the expectation over the disorder $\boldsymbol{\xi}$. The typical value of p in the thermodynamic limit concentrates and it can be accessed through the lens of replica theory as an order parameter assessed through a saddle point equation. The free entropy of the decoupled system is

$$\phi(k_1, k_2) = \frac{1}{N}\mathbb{E}_{\boldsymbol{\xi}} \log Z_{\boldsymbol{\xi}}(k_1, k_2) \tag{B.2}$$

and since it contains a factorized product $Z_{\boldsymbol{\xi}}(k_1, k_2) = Z_{\boldsymbol{\xi}}(k_1)Z_{\boldsymbol{\xi}}(k_2)$, being

$$Z_{\boldsymbol{\xi}}(k) \equiv \int \prod_{i=1}^{N} dW_i \, \delta\left(\sum_{i=1}^{N} W_i^2 - N\right) \prod_{\mu=1}^{P} \Theta\left(\frac{1}{\sqrt{N}}\boldsymbol{W} \cdot \boldsymbol{\xi}^\mu - k\right) \equiv \int d\mu(\boldsymbol{W}) \, \mathbb{X}_{\boldsymbol{\xi}}(\boldsymbol{W}; k) \tag{B.3}$$

the expression (B.2) is straightforward to simplify. It reduces on computing $\phi = \lim_{N\to\infty} \lim_{n\to 0} \frac{\langle Z^n \rangle_{\boldsymbol{\xi}} - 1}{nN}$ and so the average of a replicated partition function with the same disorder realization. Avoiding some standard steps, one can conveniently rewrite the function as

$$\langle Z^n \rangle = \int \prod_{a<b,rs} \frac{dq_{rs}^{ab} \, d\hat{q}_{rs}^{ab}}{2\pi} \int \prod_{ar} \frac{d\hat{h}_r^a}{2\pi} \, e^{Nn\phi(q,\hat{q},\hat{h})} \tag{B.4}$$

where appropriate order parameters have been defined

$$q_{rs}^{ab} = \frac{1}{N}\sum_i W_i^{ar} W_i^{bs}. \tag{B.5}$$

Notice that we have now two indices: $a \in [n]$ is for the virtual replicas of the system, while $r \in \{1, 2\}$ refers to the pairs of solutions whose overlap we are trying to characterize. In (B.4), the free energy ϕ now reads

$$\phi(q, \hat{q}, \hat{h}) = -\frac{1}{2n} \sum_{a<b,rs} q_{rs}^{ab} \hat{q}_{rs}^{ab} + \frac{1}{n} \sum_{ar} \hat{h}_r^a + \frac{1}{n} G_S + \frac{\alpha}{n} G_E \tag{B.6}$$

where

$$G_S = \ln \int \prod_{ar} dW^{ar} \, e^{\frac{1}{2} \sum_{a<b,rs} W^{ar} W^{bs} \hat{q}_{rs}^{ab} - \sum_{ar} (W^{ar})^2 \hat{h}_r^a} \tag{B.7a}$$

$$G_E = \ln \int \prod_{ar} \frac{d\lambda_r^a \, d\hat{\lambda}_r^a}{2\pi} \prod_{ar} e^{i\lambda_r^a \hat{\lambda}_r^a} \, \Theta\left(\lambda_r^a - k_r\right) \, e^{-\frac{1}{2} \sum_{a<b,rs} q_{rs}^{ab} \hat{\lambda}_r^a \hat{\lambda}_s^b}. \tag{B.7b}$$

B.1.1 RS - Ansatz

As mentioned in the main text, the RS ansatz can be imposed as follows

$$q_{rr}^{ab} = \delta_{ab} + q_r(1 - \delta_{ab}) \quad \forall r = s, \tag{B.8a}$$

$$q_{rs}^{ab} = R\,\delta_{ab} + p\,(1 - \delta_{ab}) \quad \forall r \neq s \tag{B.8b}$$

and q_r represents the typical overlap of a solution with k_r, while p and R are respectively the overlaps between the two reference solutions within the same or different replicas. There is no reason in principle to assume p and R to be different since the system is decoupled, but since we are looking for the value of p which would not appear in the calculation if $p = R$, we use the trick of assuming them different and later perform the limit $R \to p$. The calculation follows standard steps and in the end the free entropy can be written as

$$\phi = \frac{1}{2} \sum_r q_r \hat{q}_r + \sum_r \hat{h}_r + p\,\hat{p} - R\,\hat{R} + \mathcal{G}_S + \alpha \mathcal{G}_E \tag{B.9}$$

where

$$\mathcal{G}_S = \frac{1}{2} \ln \frac{4\pi^2}{(2\hat{h}_1 + \hat{q}_1)(2\hat{h}_2 + \hat{q}_2) - (\hat{p} - \hat{R})^2} + \frac{\hat{h}_2\,\hat{q}_1 + (\hat{h}_1 + \hat{q}_1)\,\hat{q}_2 + \hat{p}\,(\hat{R} - \hat{p})}{(2\hat{h}_1 + \hat{q}_1)(2\hat{h}_2 + \hat{q}_2) - (\hat{p} - \hat{R})^2} \tag{B.10a}$$

$$\mathcal{G}_E = \int \mathcal{D}z_1 \mathcal{D}z_2 \ln \int \mathcal{D}x \, H\left(\frac{k_1 + \sqrt{R - p}\,x + \sqrt{q_1}\,z_1}{\sqrt{1 - q_1 - R + p}}\right) \times H\left(\frac{k_2 + \sqrt{R - p}\,x + \frac{p}{\sqrt{q_1}}\,z_1 + \sqrt{q_2 - \frac{p^2}{q_1}}\,z_2}{\sqrt{1 - q_2 - R + p}}\right) \tag{B.10b}$$

and $\mathcal{D}x \equiv dx \, \frac{e^{-x^2/2}}{\sqrt{2\pi}}$, $H(x) = \frac{1}{2}\mathrm{erfc}\left(\frac{x}{\sqrt{2}}\right)$. By differentiating Eq.(B.9) with respect to the order parameters $q_1, q_2, p, R, \hat{q}_1, \hat{q}_2, \hat{h}_1, \hat{h}_2, \hat{p}, \hat{R}$, the saddle point equations take

the form of

$$
\begin{aligned}
&q_1 = -2\frac{\partial \mathcal{G}_S}{\partial \hat{q}_1}\,, \quad q_2 = -2\frac{\partial \mathcal{G}_S}{\partial \hat{q}_2}\,, \quad p = -\frac{\partial \mathcal{G}_S}{\partial \hat{p}}\,, \quad R = \frac{\partial \mathcal{G}_S}{\partial \hat{R}}\,,\\
&\hat{q}_1 = -2\alpha\frac{\partial \mathcal{G}_E}{\partial q_1}\,, \quad \hat{q}_2 = -2\alpha\frac{\partial \mathcal{G}_E}{\partial q_2}\,, \quad \hat{p} = -\alpha\frac{\partial \mathcal{G}_E}{\partial p}\,, \quad \hat{R} = \alpha\frac{\partial \mathcal{G}_E}{\partial R}\,, \quad 1+\frac{\partial \mathcal{G}_S}{\partial \hat{h}_1} = 0\,, \quad 1+\frac{\partial \mathcal{G}_S}{\partial \hat{h}_2} = 0.
\end{aligned}
\tag{B.11}
$$

and since in this simple case the coniugated order parameters can be solved as a function of the non-coniugated ones

$$
\begin{aligned}
&\hat{q}_1 = \frac{q_1}{(1-q_1)^2}\,, \qquad \hat{q}_2 = \frac{q_2}{(1-q_2)^2}\,, \qquad \hat{p} = \frac{p}{(1-q_1)(1-q_2)}\,,\\
&\hat{R} = \frac{p}{(1-q_1)(1-q_2)}\,, \quad \hat{h}_1 = \frac{1-2q_1}{2(1-q_1)^2}\,, \quad \hat{h}_2 = \frac{1-2q_2}{2(1-q_2)^2}.
\end{aligned}
\tag{B.12}
$$

we have reduced the number of saddle point equations to evaluate numerically. In the above expressions we took advante of $p = R$. Notice also that, if $p = R$ is directly replaced in eq. (B.6), he expression reduces to the sum of single entropy configuration.

B.1.2 $R \to p$ limit

To actually compute the value of p we resort to the saddle point equation $\hat{R} = \alpha\frac{\partial G_E}{\partial R}$ in the limit $R \to p$. Defining

$$
\begin{aligned}
\mathcal{Z} &\equiv \int \mathcal{D}x\, H\left(\frac{k_1 + \sqrt{R-p}\,x + \sqrt{q_1}\,z_1}{\sqrt{1-q_1-R+p}}\right) H\left(\frac{k_2 + \sqrt{R-p}\,x + \frac{p}{\sqrt{q_1}}\,z_1 + \sqrt{q_2 - \frac{p^2}{q_1}}\,z_2}{\sqrt{1-q_2-R+p}}\right)\\
&= \int \mathcal{D}x\, H(a+bx)H(c+dx)
\end{aligned}
\tag{B.13}
$$

and integrating by parts once the limit has been replaced in order to remove divergences, we have

$$
\frac{\partial G_E}{\partial R} = \frac{1}{\sqrt{(1-q_1)(1-q_2)}} \int \mathcal{D}z_1 \mathcal{D}z_2 \frac{G(a)G(c)}{\mathcal{Z}}
\tag{B.14}
$$

since when $R \to p$, then $b, d \to 0$ as

$$
\lim_{R\to p} \frac{b}{\sqrt{R-p}} = \frac{1}{\sqrt{1-q_1}}
\tag{B.15a}
$$

$$
\lim_{R\to p} \frac{d}{\sqrt{R-p}} = \frac{1}{\sqrt{1-q_2}}.
\tag{B.15b}
$$

Using equation (B.12) for $\hat{R}$ we therefore get p by solving the following implicit equation

$$
\begin{aligned}
p &= \alpha\sqrt{(1-q_1)(1-q_2)}\int \mathcal{D}z_1\mathcal{D}z_2\, GH(a)\, GH(c) \\
&= \alpha\sqrt{(1-q_1)(1-q_2)}\int \mathcal{D}z_1\mathcal{D}z_2\, GH\left(\frac{k_1+\sqrt{q_1}\,z_1}{\sqrt{1-q_1}}\right) GH\left(\frac{k_2+\frac{p}{\sqrt{q_1}}\,z_1+\sqrt{q_2-\frac{p^2}{q_1}}\,z_2}{\sqrt{1-q_2}}\right)
\end{aligned}
\tag{B.16}
$$

where $GH(x) \equiv \frac{G(x)}{H(x)}$.

The $\kappa_{\max}$ limit for the overlap of solution-pairs

One can imagine to specialize to the case where one of the two configuration has $k_2 = \kappa_{\max}$ and so it is a the threshold for satisfiability. In that case, by definition (see chapters 3, 4) the internal entropy of $\boldsymbol{W}^2$ vanishes and $q_2 \to 1$. The corresponding equation for p in that limit becomes

$$
\begin{aligned}
p =& \alpha\sqrt{(1-q_1)}\int \mathcal{D}z_1\mathcal{D}z_2\, GH\left(\frac{k_1+\sqrt{q_1}\,z_1}{\sqrt{1-q_1}}\right)\Theta\left(\kappa_{\max}+p\,z_1+\sqrt{1-\frac{p^2}{q_1}}\,z_2\right) \\
&\times\left(\kappa_{\max}+p\,z_1+\sqrt{1-\frac{p^2}{q_1}}\,z_2\right).
\end{aligned}
\tag{B.17}
$$

when one uses the identity

$$
GH(ax) \simeq ax\,\Theta(x)\,, \qquad \text{for } a\to\infty \tag{B.18}
$$

Integrating in z_2 we get the desired expression

$$
\begin{aligned}
p = \alpha\sqrt{(1-q_1)}\int \mathcal{D}z_1\, GH\left(\frac{k_1+\sqrt{q_1}\,z_1}{\sqrt{1-q_1}}\right) \\
\times\left[\left(\kappa_{\max}+\frac{p}{\sqrt{q_1}}\,z_1\right) H\left(-\frac{\kappa_{\max}+\frac{p}{\sqrt{q_1}}z_1}{\sqrt{1-\frac{p^2}{q_1}}}\right)+\sqrt{1-\frac{p^2}{q_1}}\,G\left(\frac{\kappa_{\max}+\frac{p}{\sqrt{q_1}}z_1}{\sqrt{1-\frac{p^2}{q_1}}}\right)\right].
\end{aligned}
\tag{B.19}
$$

B.2 Training error of the convex envelope

We refer to the general computation of the manifold spanned by y solutions sampled with different margins. Given the Perceptron problem as defined in the main text (5.1), we have a number y of solutions $\boldsymbol{W}^r$, $r \in [y]$ that are $N-$dimensional vectors normalized on the sphere with radius $\sqrt{N}$: $||\boldsymbol{W}^r||^2 = N$. The general case is the one

for which, given an instance of the problem at κ_E, we have one solution at margin $k_1 > \kappa_E$ and $y-1$ solutions at margins $k_2 < \kappa_E$, for which in principle $k_1 \neq k_2 \neq \kappa_E$. Constructing the convex envelope means to define the interpolating vector

$$\boldsymbol{W}_{\boldsymbol{\gamma}} = \sqrt{N}\frac{\sum_r \gamma_r \boldsymbol{W}^r}{\| \sum_r \gamma_r \boldsymbol{W}^r \|} \tag{B.20}$$

where $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_y)$ stands for a vector of interpolating coefficients such that $\gamma_r \geq 0$ and $\sum_{r=1}^{y} \gamma_r = 1$. Notice that in the expression (B.20), $\boldsymbol{W}_{\boldsymbol{\gamma}}$ is normalized on the sphere to ensure correspondence: $||\boldsymbol{W}_{\boldsymbol{\gamma}}||^2 = N$. We want to compute the corresponding expected asymptotic training error

$$E_{\boldsymbol{\gamma}} = \lim_{N\to+\infty} \mathbb{E}_{\boldsymbol{\xi}} \left\langle \Theta\big(- \boldsymbol{W}_{\boldsymbol{\gamma}} \cdot \boldsymbol{\xi}^\mu + \kappa_E \sqrt{N}\big) \right\rangle_{k_1,\dots,k_y}. \tag{B.21}$$

where the expectation $\langle \bullet \rangle$ is over the factorized measure

$$\langle \bullet \rangle \;=\; \frac{\int \prod_{r=1}^{y} d\mu(\boldsymbol{W}^r) \prod_{\mu r} \Theta\left(\frac{1}{\sqrt{N}} \sum_i W_i^r \xi_i^\mu - k_r\right) \bullet}{\prod_{r=1}^{y} Z(k_r)}. \tag{B.22}$$

Using the replica trick $Z^{-1} = \lim_{n\to 0} Z^{n-1}$ we have

$$\begin{aligned}
E_{\boldsymbol{\gamma}} &= \lim_{N\to\infty} \lim_{n\to 0} \mathbb{E}_{\boldsymbol{\xi}} \int \prod_{a=1}^{n} \prod_{r=1}^{y} d\mu(\boldsymbol{W}^{ar}) \prod_{\mu ar} \Theta\left(\frac{1}{\sqrt{N}} \sum_i W_i^{ar} \xi_i^\mu - k_r\right) \Theta\left(-\sum_i W_{,i}^1 \xi_i^1 + \kappa_E\right) \\
&= \lim_{N\to\infty} \lim_{n\to 0} \mathbb{E}_{\boldsymbol{\xi}} \int \prod_{ar} d\mu(\boldsymbol{W}^{ar}) \prod_{\mu ar} \frac{d\lambda_r^{\mu a} d\hat{\lambda}_r^{\mu a}}{2\pi} \prod_{\mu ar} \Theta(\lambda_r^{\mu a} - k_r) \; \Theta\left(-\frac{\sqrt{N}\sum_{r=1}^{y} \gamma_r \lambda_r^{11}}{\| \sum_r \gamma_r \boldsymbol{W}^{1r} \|} + \kappa_E\right) \\
&\quad \times e^{i \sum_{\mu ar} \hat{\lambda}_r^{\mu a} \lambda_r^{\mu a} - i \sum_{\mu a r i} \frac{1}{\sqrt{N}} \hat{\lambda}_r^{\mu a} W_{ri}^a \xi_i^\mu} \\
&= \lim_{N\to\infty} \lim_{n\to 0} \int \prod_{a<b,rs} \frac{dq_{rs}^{ab}\, d\hat{q}_{rs}^{ab}}{2\pi} \int \prod_{ar} \frac{d\hat{h}_r^a}{2\pi} \, e^{N n \phi(q,\hat{q},\hat{h})} g(q)
\end{aligned} \tag{B.23}$$

where $\phi(q, \hat{q}, \hat{h})$ and q_{rs}^{ab} have been defined respectively in equations (B.7) and (B.5). The function $g(q^\star)$, to be evaluated at the saddle point of ϕ since

$$E_{\boldsymbol{\gamma}} = \lim_{n\to 0} \; g(q_\star) \tag{B.24}$$

in the small n limit, reads instead

$$g(q) = \frac{\int \prod_{ar} \frac{d\lambda_r^a d\hat{\lambda}_r^a}{2\pi} \prod_{ar} \Theta(\lambda_r^a - k_r) \; \Theta\left(-\frac{\sum_{r=1}^{y} \gamma_r \lambda_r^1}{\sqrt{\sum_{rs} \gamma_r \gamma_s q_{rs}^{a=b}}} + \kappa_E\right) e^{i \sum_{ar} \hat{\lambda}_r^a \lambda_r^a - \frac{1}{2} \sum_{abrs} q_{rs}^{ab} \hat{\lambda}_r^a \hat{\lambda}_s^b}}{\int \prod_{ar} \frac{d\lambda_r^a d\hat{\lambda}_r^a}{2\pi} \prod_{ar} \Theta(\lambda_r^a - k_r) \; e^{i \sum_{ar} \hat{\lambda}_r^a \lambda_r^a - \frac{1}{2} \sum_{abrs} q_{rs}^{ab} \hat{\lambda}_r^a \hat{\lambda}_s^b}}. \tag{B.25}$$

B.2.1 General structure for the overlap matrix

In the RS ansatz the tensor q_{rs}^{ab} is identified by two matrices $q_{rs}^{a=b}$ and $q_{rs}^{a\neq b}$ including diagonal and off-diagonal contributions. They have the following structure

$$q_{rs}^{aa} = \delta_{rs} + (1-\delta_{rs})\, p_{rs} \tag{B.26a}$$

$$q_{rs}^{a\neq b} \equiv t_{rs} = q_r \delta_{rs} + (1-\delta_{rs})\, p_{rs}. \tag{B.26b}$$

As clarified in the main text, q_r, $r \in [y]$ is the typical overlap between pairs of virtual replicas having margin k_r, while p_{rs} is the usual effective overlap between a solution having margin k_r and another having margin k_s, with $r, s \in [y]$ and $r \neq s$. The computation proceeds straightforward once it has been realized that the term in (B.25), depending on q_{rs}^{ab} can be written as

$$\begin{aligned} -\frac{1}{2}\sum_{abrs} q_{rs}^{ab}\hat{\lambda}_r^a\hat{\lambda}_s^b &= -\frac{1}{2}\sum_r (1-q_r)\sum_a \left(\hat{\lambda}_r^a\right)^2 - \frac{1}{2}\sum_{rs} t_{rs}\sum_{ab}\hat{\lambda}_r^a\hat{\lambda}_s^b \\ &= -\frac{1}{2}\sum_r (1-q_r)\sum_a \left(\hat{\lambda}_r^a\right)^2 - \frac{1}{2}\sum_r \left(\sum_{as}\mathcal{T}_{rs}\hat{\lambda}_s^a\right)^2 \end{aligned} \tag{B.27}$$

being $\mathcal{T}_{rs}$ the *square root* matrix of t_{rs} as defined in (B.26b). Using a Hubbard-Stratonovich transformation and integrating over $\hat{\lambda}_r^a$, since the denominator tends to 1 when $n \to 0$, we get

$$\begin{aligned} E_{\boldsymbol{\gamma}} &= \lim_{n\to 0}\int\prod_r \mathcal{D}x_r \int\prod_{ar}\frac{d\lambda_r^a}{\sqrt{2\pi(1-q_r)}}\prod_{ar}\Theta(\lambda_r^a - k_r)\,\Theta\left(\kappa_E c_{\boldsymbol{\gamma}} - \sum_{r=1}^{y}\gamma_r\lambda_r^1\right) e^{-\frac{1}{2}\sum_{ar}\frac{(\lambda_r^a - \sum_s \mathcal{T}_{rs}x_s)^2}{1-q_r}} \\ &= \int\prod_r \mathcal{D}x_r \frac{\int\prod_r \mathcal{D}\lambda_r \prod_r \Theta(\sqrt{1-q_r}\lambda_r + \sum_s \mathcal{T}_{rs}x_s - k_r)\,\Theta\left(\kappa_E c_{\boldsymbol{\gamma}} - \sum_{r=1}^{y}\gamma_r\left(\sqrt{1-q_r}\lambda_r + \sum_s \mathcal{T}_{rs}x_s\right)\right)}{\prod_r H\left(\frac{k_r - \sum_s \mathcal{T}_{rs}x_s}{\sqrt{1-q_r}}\right)} \end{aligned} \tag{B.28}$$

where now the norm reads

$$c_{\boldsymbol{\gamma}} \equiv \sqrt{\sum_{rs}\gamma_r\gamma_s q_{rs}^{a=b}} = \sqrt{\sum_r \gamma_r^2 + \sum_{r\neq s} p_{rs}\gamma_r\gamma_s}. \tag{B.29}$$

Getting rid of Theta function, i.e. by expressing them as integration boundaries, the expression simplifies, and in the end

$$\begin{aligned} E_{\boldsymbol{\gamma}} &= \int\prod_r \mathcal{D}x_r \frac{\int\prod_r \mathcal{D}\lambda_r \prod_r \Theta(\lambda_r - A_r)\,\Theta\left(\frac{\kappa_E c_{\boldsymbol{\gamma}} - \sum_{rs}\gamma_r\mathcal{T}_{rs}x_s - \sum_{r=1}^{y-1}\sqrt{1-q_r}\gamma_r\lambda_r}{\gamma_y\sqrt{1-q_y}} - \lambda_y\right)}{\prod_r H(A_r)} \\ &= \int\prod_r \mathcal{D}x_r \frac{\int\prod_{r=1}^{y-2}\mathcal{D}\lambda_r \prod_{r=1}^{y-2}\Theta(\lambda_r - A_r)\,\Theta(B_{y-1} - A_{y-1})\int_{A_{y-1}}^{B_{y-1}}\mathcal{D}\lambda_{y-1}\int_{A_y}^{B_y}\mathcal{D}\lambda_y}{\prod_r H(A_r)} \\ &= \int\prod_r \mathcal{D}x_r\,\Theta(B_1 - A_1)\frac{\int_{A_1}^{B_1}\mathcal{D}\lambda_1\cdots\int_{A_y}^{B_y}\mathcal{D}\lambda_y}{\prod_r H(A_r)} \end{aligned} \tag{B.30}$$

where

$$A_r \equiv \frac{k_r - \sum_s \mathcal{T}_{rs} x_s}{\sqrt{1-q_r}}\,, \qquad r \in [y] \tag{B.31a}$$

$$B_l \equiv \frac{\kappa_E c_{\boldsymbol{\gamma}} - \sum_{r=l+1}^{y} \gamma_r k_r - \sum_{r=1}^{l} \gamma_r \sum_s \mathcal{T}_{rs} x_s - \sum_{r=1}^{l-1} \sqrt{1-q_r}\gamma_r \lambda_r}{\gamma_l \sqrt{1-q_l}}\,, \qquad l \in [y] \tag{B.31b}$$

Notably the term

$$\Theta\left(B_1 - A_1\right) = \Theta\left(\kappa_E c_{\boldsymbol{\gamma}} - \sum_r \gamma_r k_r\right) \tag{B.32}$$

does not depend on any of the x_r, $r \in [y]$ but just on a linear combination of the margins at the edges of the envelope. The innermost integral can be performed analytically, leading to the final expression

$$E_{\boldsymbol{\gamma}} = \Theta\left(\kappa_E c_{\boldsymbol{\gamma}} - \sum_r \gamma_r k_r\right) \int \prod_r \mathcal{D}z_r \frac{\int_{A_1}^{B^1} \mathcal{D}\lambda_1 ... \int_{A_{y-1}}^{B^{y-1}} \mathcal{D}\lambda_{y-1} \left(H(A_y) - H(B_y)\right)}{\prod_r H(A_r)}\,. \tag{B.33}$$

Therefore, the quantities $f_{\boldsymbol{\gamma}}$ and $I_{\boldsymbol{\gamma}}$ defined in the main text are

$$f_{\boldsymbol{\gamma}}(\kappa_E, \{k_r\}_r) \equiv \kappa_E c_{\boldsymbol{\gamma}} - \sum_r \gamma_r k_r \tag{B.34a}$$

$$I_{\boldsymbol{\gamma}}\left(\kappa_E, \{k_r\}_r\right) \equiv \int \prod_r \mathcal{D}z_r \frac{\int_{A_1}^{B^1} \mathcal{D}\lambda_1 ... \int_{A_{y-1}}^{B^{y-1}} \mathcal{D}\lambda_{y-1} \left(H(A_y) - H(B_y)\right)}{\prod_r H(A_r)}. \tag{B.34b}$$

B.2.2 Specific case with different margins

In the following case, where $y-1$ solutions have margin $k_1 > \kappa_E$, and the remaining one has margin $k_2 \neq k_1 > \kappa_E$, the overlap matrix in the RS approximation has the simple form

$$q^{a \neq b} = t = \begin{pmatrix} q_1 & \dots & q_1 & p \\ \vdots & \ddots & \vdots & \vdots \\ q_1 & \dots & q_1 & p \\ p & \dots & p & q_2 \end{pmatrix} \tag{B.35}$$

It is necessary to express the square root of the matrix t_{rs} in order to move on with the computation. It has the same form as t_{rs}

$$\mathcal{T} = \begin{pmatrix} \overline{q}_1 & \dots & \overline{q}_1 & \overline{p} \\ \vdots & \ddots & \vdots & \vdots \\ \overline{q}_1 & \dots & \overline{q}_1 & \overline{p} \\ \overline{p} & \dots & \overline{p} & \overline{q}_2 \end{pmatrix} \tag{B.36}$$

being the elements

$$\overline{q}_1 = \frac{(y-1)q_1 + \sqrt{(y-1)\left(q_1 q_2 - p^2\right)}}{(y-1)\mathcal{Z}} \tag{B.37a}$$

$$\overline{q}_2 = \frac{q_2 + \sqrt{(y-1)\left(q_1 q_2 - p^2\right)}}{\mathcal{Z}} \tag{B.37b}$$

$$\overline{p} = \frac{p}{\mathcal{Z}} \tag{B.37c}$$

$$\mathcal{Z} = q_2 + (y-1)q_1 - 2\sqrt{(y-1)\left(q_1 q_2 - p^2\right)}. \tag{B.37d}$$

As a consequence, the term $\sum_s \mathcal{T}_{rs} x_s$ in (B.33) can be written as

$$\sum_s \mathcal{T}_{rs} x_s = \begin{cases} \overline{q}_1 \sum_{s=1}^{y-1} x_s + \overline{p} x_y \,, & \text{if } r \in [y-1] \\ \overline{p} \sum_{s=1}^{y-1} x_s + \overline{q}_2 x_y \,, & \text{if } r = y \end{cases} \tag{B.38}$$

By performing some rotations over the x_r variables and using the identities

$$(y-1)\overline{q}_1^2 + \overline{p}^2 = q_1 \tag{B.39a}$$

$$\overline{p}((y-1)\overline{q}_1 + \overline{q}_2) = p \tag{B.39b}$$

$$\sqrt{y-1}\left(\overline{q}_1 \overline{q}_2 - \overline{p}^2\right) = \sqrt{q_1 q_2 - p^2} \tag{B.39c}$$

$$\tag{B.39d}$$

it is possible to perform $y-2$ of the y integrals over x_r. The simplified expression results

$$E_{\boldsymbol{\gamma}} = \Theta\left(\kappa_E c_{\boldsymbol{\gamma}} - k_1 \sum_{r=1}^{y-1} \gamma_r - k_2 \gamma_y\right) \int \prod_r \mathcal{D}x_1 \, \mathcal{D}x_2 \frac{\int_{A_1}^{B^1} \mathcal{D}\lambda_1 \ldots \int_{A_{y-1}}^{B^{y-1}} \mathcal{D}\lambda_{y-1} \left(H(A_y) - H(B_y)\right)}{\prod_r H(A_r)} \tag{B.40}$$

with $c_{\boldsymbol{\gamma}}^2 = \sum_r \gamma_r^2 + 2\gamma_y p \sum_{r=1}^{y-1} \gamma_r + 2q_1 \sum_{r<s<y} \gamma_r \gamma_s$ and

$$A_r = \begin{cases} \frac{k_r - \sqrt{q_1} x_1}{\sqrt{1-q}} \,, & r \in [y-1] \\ \frac{k_r - \frac{p}{\sqrt{q_1}} x_1 - \sqrt{q_2 - \frac{p^2}{q_1}} x_2}{\sqrt{1-q_2}} \,, & r = y \end{cases} \tag{B.41a}$$

$$B_l \equiv \begin{cases} \frac{\kappa_E c_{\boldsymbol{\gamma}} - k_1 \sum_{r=l+1}^{y-1} \gamma_r - k_2 \gamma_y - \sqrt{q} x_1 \sum_{r=1}^{l} \gamma_r - \sqrt{1-q} \sum_{r=1}^{l-1} \gamma_r \lambda_r}{\gamma_l \sqrt{1-q_l}} \,, & l \in [y-1] \\ \frac{\kappa_E c_{\boldsymbol{\gamma}} - \sqrt{q} x_1 \sum_{r=1}^{y-1} \gamma_r - \gamma_y \left[\frac{p}{\sqrt{q_1}} x_2 + \sqrt{q_2 - \frac{p^2}{q_1}} x_2\right] - \sqrt{1-q_1} \sum_{r=1}^{y-1} \gamma_r \lambda_r}{\gamma_y \sqrt{1-q_2}} \,, & l = y. \end{cases} \tag{B.41b}$$

B.2.3 Convex subset in a probabilistic setting

In the main text we have employed a notion of convexity in a probabilistic setting that we clarify here exactly as in [119]. Given a system size N with a density of constraints α and an instance determined by κ_E for a realization of patterns $\{\boldsymbol{\xi}^\mu\}_{\mu=1}^{\alpha N}$, we denote with $S_{\boldsymbol{\xi},\kappa_E}$ the solution set for the CSP. While, in general, the random

set $S_{\boldsymbol{\xi},\kappa_E}$ is with high-probability geodesically non-convex, if one span the simplex constructed by the convex envelope of y independently sampled solutions $\{\boldsymbol{W}^r\}_{r=1}^{y}$ with a margin $\kappa > \kappa_y^\star(\alpha, \kappa_E)$

$$A_y = \left\{ \boldsymbol{W}_{\boldsymbol{\gamma}} \,:\, \sum_r \gamma_r = 1,\ 0 \le \gamma_r \le 1\ \forall r, \boldsymbol{W}_{\boldsymbol{\gamma}} = \sqrt{N}\frac{\sum_r \gamma_r \boldsymbol{W}^r}{\|\sum_r \gamma_r \boldsymbol{W}^r\|} \right\}, \tag{B.42}$$

it will be with high probability entirely made of solutions

$$\lim_{N\to\infty} \mathbb{P}(A_y \subseteq S_{\boldsymbol{\xi},\kappa_E}) = 1. \tag{B.43}$$

Here the randomness is given by both the realization of $\boldsymbol{\xi}$ and the sampling of the $\boldsymbol{W}^r$. While we conjecture Eq. (B.43) to hold in the $\kappa > \kappa_y^\star(\alpha, \kappa_E)$ regime, we cannot establish it firmly with the only use of non-rigorous replica techniques. A much weaker notion of convexity in probability is instead implied by our (again non-rigorous) analysis. In fact, we have that

$$\lim_{N\to\infty} \mathbb{P}(\boldsymbol{W}_{\boldsymbol{\gamma}} \in S_{\boldsymbol{\xi},\kappa_E}) = 1 \qquad \forall \boldsymbol{\gamma}. \tag{B.44}$$

This result can be obtained by applying Markov's inequality to the energy of the interpolant configuration: $\mathbb{P}(E(\boldsymbol{W}_{\boldsymbol{\gamma}}) > \epsilon) \le \frac{\mathbb{E}E(\boldsymbol{W}_{\boldsymbol{\gamma}})}{\epsilon}$. The large limit of the expectation $\mathbb{E}E(\boldsymbol{W}_{\boldsymbol{\gamma}})$ is in fact the energy $E_{\boldsymbol{\gamma}}$ in (B.33). This result, specific to each point in $\boldsymbol{\gamma}$, can be further enhanced by considering the *total* energy, which is the integration of the energy across the simplex. One can show, indeed, that it tends towards zero in expectation as the system size N becomes large and k exceed $\kappa_y^\star(\alpha, \kappa_E)$. When it happens, this leads us to suppose that the maximum energy point on the simplex (which is the condition wrt which the threshold $\kappa_y^\star(\alpha, \kappa_E)$ is derived) also asymptotically approaches zero energy, because of the continuity of the expected energy argument, thereby supporting the robust notion of convexity as described in Eq. (B.43). However, a more precise characterization of this stochastic geometry and mathematical validation of our conjectures remain as open challenges.

B.3 Stability distribution along the geodesic path

The stability distribution along the geodesic path connecting two solutions $\boldsymbol{W}^1$ and $\boldsymbol{W}^2$ with margins $k_{1,2}$ means to evaluate

$$P(\Delta) = \mathbb{E}_{\boldsymbol{\xi}} \left\langle \delta\Big(\Delta - \frac{\boldsymbol{W}_{\boldsymbol{\gamma}} \cdot \boldsymbol{\xi}^1}{\sqrt{N}}\Big) \right\rangle_{k_1,k_2} \tag{B.45}$$

where now the vector of interpolation as in (B.20) is simply $\boldsymbol{W}_{\boldsymbol{\gamma}} = \sqrt{N}\frac{\gamma \boldsymbol{W}^1 + (1-\gamma)\boldsymbol{W}^2}{\|\gamma \boldsymbol{W}^1 + (1-\gamma)\boldsymbol{W}^2\|}$. By writing as usual $Z^{-1} = \lim_{n\to 0} Z^{n-1}$ and enforcing the definitions of preactivations

by using delta functions, we get

$$
\begin{aligned}
P(\Delta) &= \int \prod_{ar} \frac{d\lambda_r^a d\hat{\lambda}_r^a}{2\pi} \prod_{ar} \Theta(\lambda_r^a - \kappa_r) e^{i\sum_{ar} \lambda_r^a \hat{\lambda}_r^a - \frac{1}{2}\sum_{arbs} \hat{\lambda}_r^a q_{rs}^{ab} \hat{\lambda}_s^b} \, \delta\Big(\Delta - \frac{\gamma\lambda_1^1 + (1-\gamma)\lambda_2^1}{||W||}\Big) \\
&= \int \mathcal{D}z \prod_r \mathcal{D}y_r \int \prod_r \frac{d\lambda_r d\hat{\lambda}_r}{2\pi} e^{-\frac{1}{2}\sum_r (1-q_r)(\hat{\lambda}_r)^2 + i\sum_r \sqrt{q_r - p} y_r \hat{\lambda}_r + i\sqrt{p} z \sum_r \hat{\lambda}_r + i\sum_r \lambda_r \hat{\lambda}_r} \times \\
&\quad \left(\int \prod_r \frac{d\lambda_r d\hat{\lambda}_r}{2\pi} e^{-\frac{1}{2}\sum_r (1-R+p-q_r)(\hat{\lambda}_r)^2 + i\sum_r \sqrt{q_r - p} y_r \hat{\lambda}_r + i\sqrt{p} z \sum_r \hat{\lambda}_r} \times e^{i\sum_r \lambda_r \hat{\lambda}_r} \prod_r \Theta(\lambda_r - \kappa_r) \right)^{n-1} \\
&\quad \times \prod_r \Theta(\lambda_r - \kappa_r) \, \delta\Big(\Delta - \frac{\gamma(\sqrt{A_1}\lambda_1 - b_1) + (1-\gamma)\sqrt{A_2}\lambda_2 - b_2}{||W||}\Big) \\
&= \int \mathcal{D}z \prod_r \mathcal{D}y_r \frac{\int \frac{\mathcal{D}\lambda}{\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{||W||\Delta + \gamma b_1 - (1-\gamma)(\sqrt{A_2}\lambda_2 - b_2)}{\gamma\sqrt{A_1}}\right)^2} \Theta\left(\frac{||W||\Delta + \gamma b_1 - (1-\gamma)(\sqrt{A_2}\lambda_2 - b_2)}{\gamma} - b_1 - \kappa_1\right)}{\prod_r H\left(\frac{b_r + \kappa_r}{\sqrt{A_r}}\right)} \\
&\quad \times \Theta(\sqrt{A_2}\lambda - b_2 - \kappa_2) \frac{||W||}{\gamma\sqrt{A_1}}
\end{aligned}
\tag{B.46}
$$

where $A_r = 1 - q_r$ and $b_r = \sqrt{q_r - p} y_r + \sqrt{p} z$. Looking at the numerator of the last equality in eq. (B.46), we realize it can be expressed as

$$
\frac{||W||}{\gamma\sqrt{A_1}} \int_{\frac{b_2+\kappa_2}{\sqrt{A_2}}}^{\frac{||W||\Delta - \gamma\kappa_1 + (1-\gamma)b_2}{(1-\gamma)\sqrt{A_2}}} D\lambda \, \Theta\Big(\frac{||W||\Delta - \gamma\kappa_1 - (1-\gamma)\kappa_2}{(1-\gamma)\sqrt{A_2}}\Big) G\Big(\frac{||W||\Delta + \gamma b_1 - (1-\gamma)(\sqrt{A_2}\lambda - b_2)}{\gamma\sqrt{A_1}}\Big)
\tag{B.47}
$$

by making explicit the Heaviside functions as integral extremes. Handling the expressions a little more, finally we end up with the formula

$$
\begin{aligned}
P(\Delta) &= \sqrt{\frac{\Gamma}{2\pi(1-q_1)}} \frac{c_\gamma}{\gamma} \Theta\Big(c_\gamma \Delta - \gamma k_1 - (1-\gamma)k_2\Big) \times \\
&\quad \times \int \mathcal{D}z \mathcal{D}y \, e^{\frac{1}{2}\Xi^2 - \frac{(b_1(y)\gamma + b_2(z,y)(1-\gamma) + \Delta W)^2}{2(1-q_1)\gamma^2}} \frac{H\left(\frac{(b_2(z,y)+k_2)}{\sqrt{\Gamma(1-q_2)}} - \Xi\right) - H\left(\frac{(b_2(z,y)(1-\gamma) - \gamma k_1 + \Delta c_\gamma)}{\sqrt{\Gamma(1-q_2)}(1-\gamma)} - \Xi\right)}{\prod_r H\left(\frac{b_r + k_r}{\sqrt{1-q_r}}\right)}
\end{aligned}
\tag{B.48}
$$

where we have introduced the functions

$$
b_1(y) = \sqrt{q_1} y \tag{B.49a}
$$

$$
b_2(z,y) = \frac{p}{\sqrt{q_1}} y + \sqrt{q_2 - \frac{p^2}{q_1}} z \tag{B.49b}
$$

and the constants

$$\Gamma = \frac{(1-q_1)\gamma^2}{(1-q_1)\gamma^2 + (1-q_2)(\gamma-1)^2} \tag{B.50a}$$

$$\Xi = \frac{(1-\gamma)\sqrt{1-q_2}(\|W\|\Delta + \gamma b_1(y) + (1-\gamma)b_2(z,y))}{\gamma\sqrt{1-q_1}\sqrt{(1-\gamma)^2(1-q_2)+\gamma^2(1-q_1)}} \tag{B.50b}$$

$$c_\gamma = \sqrt{\gamma^2 + (1-\gamma)^2 + 2\gamma(1-\gamma)p}. \tag{B.50c}$$

B.4 Sampling bias of the Perceptron Algorithm

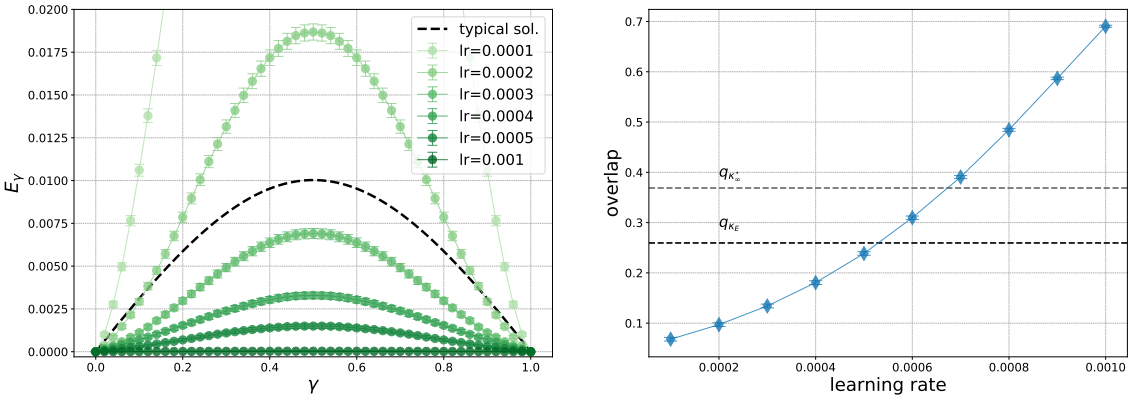


Figure B.1: (Left) Maximum fraction of errors along the geodesic path connecting two solutions obtained with PA at $\alpha = 1$ and $\kappa_E = 0.5$ using different learning rates at $N = 1000$. Dashed line is the reference barrier between typical solutions. (Right) Average overlap between PA solutions as a function of the learning rate. Horizontal dashed lines are q_{κ_E} and $q_\infty^\star$ at $\alpha = 1$. Reprinted from [119].

Here we provide some additional details on how the learning rate influences the sampling bias in PA. As previously discussed and noted in the main text, a sufficiently low learning rate allows the algorithm to identify atypical solutions with stability distributions centered around the problem's minimum margin. These solutions, typically emerging close to the initial point, exhibit extremely low similarity with each other, with much lower overlap than the one between pairs of problem's typical solutions, which numerically dominate the flat measure over all minimizers. In scenarios with low constraint densities, these solutions do not present barriers with to fBP solutions, but they do exhibit non-zero barriers among themselves at any constraint density (α). However, adjusting the learning rate in the optimization process can significantly change the nature of the solutions sampled by PA. As shown in the left panel of Fig.B.1, higher learning rates, which lead to faster optimization, result in solution pairs with progressively decreasing energy barriers. These barriers reach a point where they intersect with the standard energy barrier among equilibrium solutions (represented by the dashed black line). On the contrary, right section of Fig.B.1, the reciprocal overlap increases with the learning rate, up to values that are

larger than that of the equilibrium solutions and comparable with that of the overlap of solutions in the kernel regions with margin $\kappa_\infty^\star$.

B.5 Extrapolation to large N values

For the sake of clarity, all the comparisons in the main text between theoretical prediction of the energy barriers and numerical simulations of SA with Hinge loss refer to the numerical extrapolations of experiments in the large N limit. Increasing N, indeed, has the effect of lowering the gap between RS theory and simulations, since finite size effects comes into play at finite N. In Fig. B.2 we show, as a function of $1/N$, the behaviour of the training error on two points ($\gamma = 0.5, 0.2$) on the linear interpolation path between two typical solutions of the problem ($k_{1,2} = \kappa_E$). As it is clear from the plot, the large N extrapolations fall within the corresponding analytical predictions in horizontal lines.

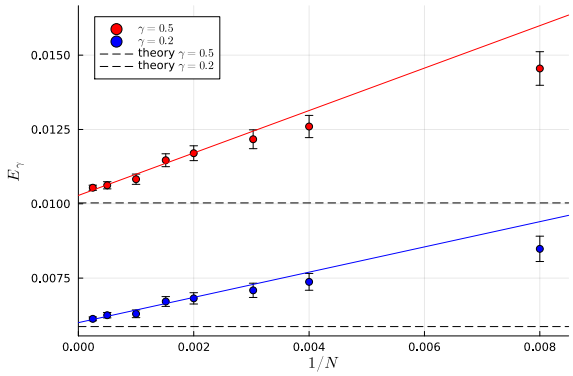


Figure B.2: Energy on the linear interpolation between two solutions with margin $k = \kappa_E = -0.5$ found by sampling with SA at various sizes and their extrapolations (full lines) at large N. The two colors refer to the case $\gamma = 0.5$ (red) and $\gamma = 0.2$ (blue). The horizontal dashed black lines represents the theoretical predictions. In the extrapolations we have fitted the last 5 points. Reprinted from [119].

Appendix C

Details of the *biological* TCM

In this chapter we refer to the replica computation under the RS ansatz for the dendritic model of a neuron. The computation is standard, and with the relevant steps given in the main text, the calculation of the averaged free entropy in the thermodynamic limit does not need much further explanation. Rather, we will focus on the critical capacity limit and the weight distribution expression.

C.1 Free entropy and saddle points

As reported in (C.1), the average free entropy of the model is

$$\phi = \lim_{N\to\infty} \frac{1}{N}\langle \ln Z\rangle_{\xi,\sigma} = \frac{q\hat{q}}{2} - Q\hat{Q} - \overline{W}\hat{M} + \mathcal{G}_S + \alpha\,\mathcal{G}_E \tag{C.1a}$$

$$\mathcal{G}_S = \ln\sqrt{\frac{2\pi}{\hat{q}-2\hat{Q}}} + \frac{1}{2}\left(\frac{\hat{M}^2+\hat{q}}{\hat{q}-2\hat{Q}}\right) + \int Dz\,\ln H\left(-\frac{\hat{M}+\sqrt{\hat{q}}z}{\sqrt{\hat{q}-2\hat{Q}}}\right) \tag{C.1b}$$

$$\mathcal{G}_E = \int Dt\left[f_{\text{out}} H\left(\frac{\sqrt{D_0}\,z-\Delta}{\sqrt{D_1}}\right) + (1-f_{\text{out}})H\left(\frac{\sqrt{D_0}\,z+\Delta}{\sqrt{D_1}}\right)\right] \tag{C.1c}$$

in the $N, K \to \infty$ limits and with $\frac{K}{N} \to 0$. We now need to compute the saddle point equations by differentiating (C.1a) with respect to the order parameters $Q, q, M, \hat{Q}, \hat{q}, \hat{M}$. The saddle point equations involving the entropic term (i.e. taking derivatives with respect to $\hat{q}$, $\hat{Q}$ and $\hat{M}$) are the same as in the case of the perceptron [129]

$$\overline{W} = \int Dz\,\frac{\int_0^\infty dW\, W e^{-(\hat{q}-2\hat{Q})\frac{W^2}{2}+(\hat{M}+\sqrt{\hat{q}}z)W}}{\int_0^\infty dW\, e^{-(\hat{q}-2\hat{Q})\frac{W^2}{2}+(\hat{M}+\sqrt{\hat{q}}z)W}} \tag{C.2a}$$

$$Q = \int Dz\,\frac{\int_0^\infty dW\, W^2 e^{-(\hat{q}-2\hat{Q})\frac{W^2}{2}+(\hat{M}+\sqrt{\hat{q}}z)W}}{\int_0^\infty dW\, e^{-(\hat{q}-2\hat{Q})\frac{W^2}{2}+(\hat{M}+\sqrt{\hat{q}}z)W}} \tag{C.2b}$$

$$q = \int Dz\left[\frac{\int_0^\infty dW\, W e^{-(\hat{q}-2\hat{Q})\frac{W^2}{2}+(\hat{M}+\sqrt{\hat{q}}z)W}}{\int_0^\infty dW\, e^{-(\hat{q}-2\hat{Q})\frac{W^2}{2}+(\hat{M}+\sqrt{\hat{q}}z)W}}\right]^2 . \tag{C.2c}$$

In order to express the remaining saddle point equations in a compact way, we can define the quantities

$$a_\sigma(z) = \frac{\sqrt{D_0}z - \sigma\Delta}{\sqrt{D_1}} = \sqrt{\frac{D_0}{D_1}}(z - \tau_\sigma)$$
$$\tau_\sigma = \frac{\sigma\Delta}{\sqrt{D_0}}\,.$$

Deriving (C.1a) with respect to M, Q and q lead respectively to

$$0 = \mathbb{E}_\sigma \sigma \int Dz\, \frac{G(a_\sigma(z))}{H(a_\sigma(z))} \tag{C.4a}$$
$$\hat{Q} = \frac{\alpha}{2}\mathbb{E}_\sigma \int Dz\, \frac{G(a_\sigma(z))}{H(a_\sigma(z))} \left[\frac{a_\sigma(z)}{D_1}\frac{dD_1}{dQ} - \frac{z}{\sqrt{D_0 D_1}}\frac{dD_0}{dQ}\right] \tag{C.4b}$$
$$\hat{q} = \alpha\mathbb{E}_\sigma \int Dz\, \frac{G(a_\sigma(z))}{H(a_\sigma(z))} \left[-\frac{a_\sigma(z)}{D_1}\frac{dD_1}{dq} + \frac{z}{\sqrt{D_0 D_1}}\frac{dD_0}{dq}\right] \tag{C.4c}$$

where we used the saddle point equation (C.4a) when performing the derivative in Q[1]. The six saddle points Eqs. (C.2a, C.2b, C.2c, C.4a, C.4b, C.4c), obtained in the large $K, N \to \infty$ limit with $N \gg K$, have to be numerically solved to obtain the values of the order parameters and represent the final result of our RS analysis.

Notice that imposing $g(x) = x$ we recover the previous saddle point expressions obtained for the simple linear neuron model [129, 130]. Plus, in the case $f_{\text{out}} = \frac{1}{2}$ saddle point equation (C.4a) gives $M = 0$; in this case therefore $\tau_\sigma = 0$.

C.1.1 Effective order parameters for some non-linearities

We report here the analytical expressions of the effective order parameters for several non-linearities of interest

- *Theta non-linearity*: $g(x) = \Theta(x)$.

  To evaluate the integrals it is useful to use the following identity

  $$\int Dz\, H^2\left(a + bz\right) = H\left(\frac{a}{\sqrt{1+b^2}}\right) - 2T\left(\frac{a}{\sqrt{1+b^2}}, \frac{1}{\sqrt{1+2b^2}}\right) \tag{C.5}$$

  where T is the Owen's T function defined as

  $$T(h, s) \equiv \frac{1}{2\pi}\int_0^s dx\, \frac{e^{-(1+x^2)\frac{h^2}{2}}}{1+x^2}\,, \tag{C.6}$$

  and has the following important properties

  $$T(h, s) \simeq \frac{G(h)s}{\sqrt{2\pi}} + O\left(s^2\right) \qquad \text{for } s \to 0 \tag{C.7a}$$
  $$T(h, 1) = \frac{1}{2}H(h)H(-h) \tag{C.7b}$$

[1] this is why no derivative with respect to Q of Δ compares in (C.4b)

where we remind that $G(x)$ is the Gaussian with mean zero and unit variance. Defining the quantity

$$M_\star = \frac{f_{\rm in}\overline{M}}{\sqrt{f_{\rm in}(1-f_{\rm in})Q}} \tag{C.8}$$

the effective order parameters for the theta non-linearity are

$$\Delta = M_\star G\left(-M_\star\right) \tag{C.9a}$$

$$\Delta_q = H(-M_\star) - 2T\left(M_\star, \sqrt{\frac{Q-q}{Q+q}}\right) \tag{C.9b}$$

and, in particular

$$\Delta_Q = H(-M_\star)\,, \tag{C.10a}$$

$$\Delta_0 = H(-M_\star)^2\,. \tag{C.10b}$$

$\overline{M}$ is instead fixed by the relation

$$\theta_o = H(-M_\star)\,. \tag{C.11}$$

- *ReLU non-linearity*: $g(x) = x\,\Theta(x)$. We have

$$\Delta = fMH\left(-\tilde{M}\right) \tag{C.12a}$$

$$\begin{aligned}\Delta_q = {} & f_{\rm in}(1-f_{\rm in})(Q-q)\sqrt{\frac{Q-q}{Q+q}}G^2\left(\frac{f_{\rm in}\overline{M}}{\sqrt{f_{\rm in}(1-f_{\rm in})(Q+q)}}\right) \\ & + f_{\rm in}\left((1-f_{\rm in})q + f_{\rm in}\overline{M}^2\right)\left[H\left(-M_\star\right) - 2T\left(-M_\star, \sqrt{\frac{Q-q}{Q+q}}\right)\right] \\ & + 2f_{\rm in}\overline{M}\sqrt{f_{\rm in}(1-f_{\rm in})Q}\,G\left(M_\star\right)H\left(-M_\star\sqrt{\frac{Q-q}{Q+q}}\right) \\ & + 2f_{\rm in}(1-f_{\rm in})q\sqrt{\frac{Q-q}{Q+q}}G\left(M_\star\right)G\left(M_\star\sqrt{\frac{Q-q}{Q+q}}\right)\end{aligned} \tag{C.12b, C.12c}$$

and in particular

$$\Delta_Q = f_{\rm in}\left((1-f_{\rm in})Q + f_{\rm in}\overline{M}^2\right)H\left(-M_\star\right) + + f_{\rm in}\overline{M}\sqrt{f_{\rm in}(1-f_{\rm in})Q}\,G\left(-M_\star\right)\,, \tag{C.13a}$$

$$\Delta_0 = \left[\sqrt{f_{\rm in}(1-f_{\rm in})Q}\,G\left(M_\star\right) + f_{\rm in}\overline{M}H\left(-M_\star\right)\right]^2 \tag{C.13b}$$

Note that the equation that implicitly fixes the value of $\overline{M}$ now reads

$$\theta_o = \sqrt{f_{\rm in}(1-f_{\rm in})Q}\,G\left(M_\star\right) + f_{\rm in}\overline{M}H\left(-M_\star\right) \tag{C.14}$$

- *Polsky non-linearity*: sadly, this activation contains a sigmoid part and we cannot do that part of the integration analytically.

C.2 The critical capacity

In this section we show how in our formalism, it is possible to compute the maximal number of inputs that the neuron is able to classify. We underline that this can be done for a generic form of dendritic non-linearity. In the critical capacity limit, the set of possible synaptic weights shrinks towards a single point and q tends to Q:

$$q = Q - dq. \tag{C.15}$$

Correspondingly, the other order parameters scale as

$$\hat{q}\,, \hat{Q} \sim \frac{C}{dq^2} \tag{C.16a}$$

$$\hat{q} - 2\hat{Q} \sim \frac{A}{dq} \tag{C.16b}$$

$$\hat{M} \sim \frac{B\sqrt{C}}{dq} \tag{C.16c}$$

Using the identities (where a is a positive constant)

$$\frac{\int_0^\infty dx\, x\, e^{-a\frac{x^2}{2}+bx}}{\int_0^\infty dx\, e^{-a\frac{x^2}{2}+bx}} = \frac{b}{a} + \frac{1}{\sqrt{a}} \frac{G\left(-\frac{b}{\sqrt{a}}\right)}{H\left(-\frac{b}{\sqrt{a}}\right)} \tag{C.17a}$$

$$\frac{\int_0^\infty dx\, x^2\, e^{-a\frac{x^2}{2}+bx}}{\int_0^\infty dx\, e^{-a\frac{x^2}{2}+bx}} = \frac{1}{a} + \frac{b^2}{a^2} + \frac{b}{a^{3/2}} \frac{G\left(-\frac{b}{\sqrt{a}}\right)}{H\left(-\frac{b}{\sqrt{a}}\right)} \tag{C.17b}$$

and the expansion

$$\frac{G(x)}{H(x)} \simeq x\theta(x)\,, \qquad \text{for } |x| \gg 1\,, \tag{C.18}$$

the saddle point equations (C.2) can be written as

$$\overline{W} = \frac{\sqrt{C}}{A}\left[G(B) - BH(B)\right] \tag{C.19a}$$

$$Q = \frac{C}{A^2}\left[\left(1+B^2\right)H(B) - BG(B)\right] \tag{C.19b}$$

$$A = H(B). \tag{C.19c}$$

More work is required to derive the asymptotic limit of equations (C.4). First of all, we need the expansion of the effective order parameters

$$D_0 = \Delta_q - \Delta_0 = \Delta_Q - \Delta_0 - D_1 \simeq \Gamma_0 - \Gamma_1 dq \tag{C.20a}$$

$$D_1 = \Delta_Q - \Delta_q = \Gamma_1 dq + O(dq^2) \tag{C.20b}$$

where we have defined

$$\Gamma_0 = \Delta_Q - \Delta_0 = \int Dx\, g^2\left(\sqrt{f_{\rm in}(1-f_{\rm in})Q}\, x + f_{\rm in}\overline{M}\right) - \left[\int Dy\, g\left(\sqrt{f_{\rm in}(1-f_{\rm in})Q}\, y + f_{\rm in}\overline{M}\right)\right]^2 \tag{C.21a}$$

$$\Gamma_1 = f_{\rm in}(1-f_{\rm in})\int Dz\, \left[g'\left(\sqrt{f_{\rm in}(1-f_{\rm in})Q}\, z + f_{\rm in}\overline{M}\right)\right]^2 . \tag{C.21b}$$

Now we subtract (C.4b) with (C.4c) getting

$$\begin{aligned}\hat{q} - 2\hat{Q} &= \alpha\mathbb{E}_\sigma \int Dz\, \frac{G(a_\sigma(z))}{H(a_\sigma(z))}\left[\frac{z}{\sqrt{D_0 D_1}}\left(\frac{dD_0}{dq} + \frac{dD_0}{dQ}\right) - \frac{a_\sigma(z)}{D_1}\left(\frac{dD_1}{dq} + \frac{dD_1}{dQ}\right)\right] \\ &= \alpha\mathbb{E}_\sigma \int Dz\, \frac{G(a_\sigma(z))}{H(a_\sigma(z))}\left[\frac{z}{\sqrt{D_0 D_1}}\frac{d\Gamma_0}{dQ} - \frac{a_\sigma(z)}{D_1}\frac{d\Gamma_1}{dQ}dq\right] .\end{aligned} \tag{C.22}$$

Similarly (C.4c) becomes

$$\hat{q} = \alpha\mathbb{E}_\sigma \int Dz\, \frac{G(a_\sigma(z))}{H(a_\sigma(z))}\left[-\frac{a_\sigma(z)}{D_1}\frac{dD_1}{dq}\right] \tag{C.23}$$

because the second term in (C.4c) is subleading in dq. Using the expansion (C.18) and the identities

$$\int Dz\, z^2\, \Theta(z-\tau_\sigma) = H(\tau_\sigma) + \tau_\sigma G(\tau_\sigma) \tag{C.24a}$$

$$\int Dz\, z\, \Theta(z-\tau_\sigma) = G(\tau_\sigma) \tag{C.24b}$$

we obtain the following saddle point equations

$$0 = \mathbb{E}_\sigma \sigma\left[G(\tau_\sigma) - \tau_\sigma H(\tau_\sigma)\right] \tag{C.25a}$$

$$A = \alpha_c\left[\frac{1}{\Gamma_1}\frac{d\Gamma_0}{dQ}\,\mathbb{E}_\sigma H(\tau_\sigma) - \frac{\Gamma_0}{\Gamma_1^2}\frac{d\Gamma_1}{dQ}\mathbb{E}_\sigma\left[(1+\tau_\sigma^2)H(\tau_\sigma) - \tau_\sigma G(\tau_\sigma)\right]\right] \tag{C.25b}$$

$$C = \frac{\alpha_c\Gamma_0}{\Gamma_1}\mathbb{E}_\sigma\left[(1+\tau_\sigma^2)H(\tau_\sigma) - \tau_\sigma G(\tau_\sigma)\right] , \tag{C.25c}$$

which involve the critical capacity as an unknown parameter to find. The full set of saddle point equations for the order parameters A, B, C, M, Q and for α_c are in the end

$$\overline{W} = \frac{\sqrt{C}}{A}\left[G(B) - BH(B)\right] \tag{C.26a}$$

$$Q = \frac{C}{A^2}\left[\left(1+B^2\right)H(B) - BG(B)\right] \tag{C.26b}$$

$$A = H(B) \tag{C.26c}$$

$$0 = \mathbb{E}_\sigma \sigma\left[G(\tau_\sigma) - \tau_\sigma H(\tau_\sigma)\right] \tag{C.26d}$$

$$A = \frac{\alpha_c}{\Gamma_1}\frac{d\Gamma_0}{dQ}\,\mathbb{E}_\sigma H(\tau_\sigma) - \frac{1}{\Gamma_1}\frac{d\Gamma_1}{dQ}C \tag{C.26e}$$

$$C = \frac{\alpha_c\Gamma_0}{\Gamma_1}\mathbb{E}_\sigma\left[(1+\tau_\sigma^2)H(\tau_\sigma) - \tau_\sigma G(\tau_\sigma)\right] . \tag{C.26f}$$

As we have anticipated before, in the case $f_{\text{out}} = 0.5$ the equations can be further simplified, since $B = 0$ and therefore $\tau_\sigma = 0$; the saddle point equations (C.26) then reduce to

$$\overline{W} = \sqrt{\frac{\alpha_c \Gamma_0}{2\Gamma_1}} \frac{1}{H(B)} \left[G(B) - BH(B) \right] \tag{C.27a}$$

$$Q = \frac{\alpha_c \Gamma_0}{2\Gamma_1} \frac{1}{H^2(B)} \left[\left(1 + B^2\right) H(B) - BG(B) \right] \tag{C.27b}$$

$$\alpha_c = \frac{2\Gamma_1 H(B)}{\frac{d\Gamma_0}{dQ} - \frac{\Gamma_0}{\Gamma_1}\frac{d\Gamma_1}{dQ}} \,. \tag{C.27c}$$

C.3 Distribution of synaptic weights

The distribution of synaptic weights is

$$P(W) = \left\langle \frac{1}{Z} \int_0^\infty \prod_{li} dW_{li} \prod_{\mu=1}^{\alpha N} \Theta \left[\frac{\sigma^\mu}{\sqrt{K}} \left(\sum_{l=1}^{K} c_l \, g \left(\sqrt{\frac{K}{N}} \sum_{i=1}^{N/K} W_{li} \xi_{li}^\mu - \sqrt{\frac{N}{K}} \theta_1 \right) - K\theta_2 \right) \right] \delta(W - W_{11}) \right\rangle_{\{\xi^\mu, \sigma^\mu\}} \tag{C.28}$$

with respect to the first weight of the first sub-perceptron W_{11} for simplicity. Repeating the same steps as usual, i.e. introducing n virtual copies of the system for $Z^{-1} = \lim_{n\to 0} Z^{n-1}$, performing the quenched average and the $N \to \infty$ limit, we have

$$\begin{aligned} P(W) = \lim_{n\to 0} \int \prod_{a<b,l} \frac{dq_l^{ab} \, d\hat{q}_l^{ab}}{2\pi K/N} \int \prod_{a,l} \frac{dQ_l^a \, d\hat{Q}_l^a}{2\pi K/N} \int \prod_{a,l} \frac{dM_l^a \, d\hat{M}_l^a}{2\pi\sqrt{K/N}} \, e^{-\frac{N}{K} \sum_{a<b,l} q_l^{ab} \hat{q}_l^{ab} - \frac{N}{K} \sum_{a,l} Q_l^a \hat{Q}_l^a - \frac{N}{K} \overline{W} \sum_{a,l} \hat{M}_l^a} \\ \times e^{N\alpha \, G_E(q_l^{ab}, Q_l^a, M_l^a) + (\frac{N}{K} - 1) \, G_S(\hat{q}_l^{ab}, \hat{Q}_l^a, \hat{M}_l^a)} \int_0^\infty \prod_{l,a} dW_l^a \, \delta(W - W_1) \, e^{\sum_{a<b,l} \hat{q}_l^{ab} W_l^a W_l^b + \sum_{a,l} \hat{Q}_l^a (W_l^a)^2 + \sum_{a,l} \hat{M}_l^a W_l^a} \end{aligned} \tag{C.29}$$

where the entropic and energetic terms are the same as in Eq. (C.1), while the order parameters are those defined in (7.5). In the limit $n \to 0$ therefore

$$P(W) = \lim_{n\to 0} \int_0^\infty \prod_{l,a} dW_l^a \, \delta(W - W_1) \, e^{\sum_{a<b,l} \hat{q}_l^{ab} W_l^a W_l^b + \sum_{a,l} \hat{Q}_l^a (W_l^a)^2 + \sum_{a,l} \hat{M}_l^a W_l^a} \tag{C.30}$$

provided that the order parameters satisfy the same saddle point equations as before. Under RS ansatz the expression (C.30) becomes

$$\begin{aligned} P(W) &= \Theta(W) \int Dz \, \frac{e^{-\frac{1}{2}(\hat{q} - 2\hat{Q})W^2 + (\sqrt{\hat{q}} z + \hat{M})W}}{\int dW \, e^{-\frac{1}{2}(\hat{q} - 2\hat{Q})W^2 + (\sqrt{\hat{q}} z + \hat{M})W}} \\ &= \Theta(W) \sqrt{\hat{q} - 2\hat{Q}} \, e^{-\frac{1}{2}(\hat{q} - 2\hat{Q})W^2 + \hat{M}W} \int Dz \, e^{\sqrt{\hat{q}} W z} \, \frac{G\left(-\frac{\sqrt{\hat{q}} z + \hat{M}}{\sqrt{\hat{q} - 2\hat{Q}}} \right)}{H\left(-\frac{\sqrt{\hat{q}} z + \hat{M}}{\sqrt{\hat{q} - 2\hat{Q}}} \right)} . \end{aligned} \tag{C.31}$$

Notice that the dependence of $P(W)$ on K and on the activation function is not explicit, but is concealed inside the order parameters through the saddle point equations they have to satisfy.

C.3.1 Maximal storage limit

In the critical capacity limit $\alpha \to \alpha_c$ the expression of the distribution greatly simplifies. Using scalings as in (C.16), we find

$$\begin{aligned} P(W) =& \Theta(W)\, e^{-\frac{A}{2dq}W^2+\frac{B\sqrt{C}}{dq}W} \sqrt{\frac{A}{dq}} \int Dz\, e^{\frac{\sqrt{C}}{dq}Wz} \\ &\times \left[G\left(\sqrt{\frac{C}{A}} \frac{B+z}{\sqrt{\hat{q}-2\hat{Q}}} \right) \Theta(B+z) - \sqrt{\frac{C}{Adq}} (B+z)\Theta(-B-z) \right] \end{aligned} \tag{C.32}$$

and with the identity

$$\int Dz\, e^{az}\, (z+b)\, \Theta(-b-z) = (a+b)\, e^{\frac{a^2}{2}} H(a+b) - e^{-ab}\, G(b) \tag{C.33}$$

we get the final expression

$$P(W) = H(-B)\, \delta(W) + \frac{1}{\sqrt{2\pi}W_\star} e^{-\frac{(W+BW_\star)^2}{2W_\star^2}} \Theta(W) \tag{C.34}$$

where $W_\star \equiv \frac{\sqrt{C}}{A}$. The right-hand-term of the distribution can be identifies as the positive portion of a normal distribution with mean $BW_\star$ and a standard deviation $W_\star$. The silent synapses instead are represented by the left-hand term that with a Dirac delta function generates a spike of zero-weight synapses. The fraction of silent synapses is $H(B)$.

Appendix D

Details on the RF-Hopfield Model

D.1 Feature retrieval

Starting from the definition of the replicated partition function as in Eq. 8.7, in the main text it is shown how is possible to perform the average over $c_{\nu k} > 1$ using GEP and taking into account $\mu_1^a = \frac{1}{N}\sum_i f_{1i} s_i^a$ the non-vanishing feature magnetization. Using CLT, if the distribution of m_ν^a is a multivariate Gaussian $P(m_\nu^a) = \mathcal{N}(m_\nu^a; \overline{m}, Q)$ with mean $\bar{m}^a = \frac{c_{\nu 1}}{\sqrt{\alpha_D}}\mu_1^a \kappa_1$ and Covariance (8.12), doing a simple shift $m_\nu^a \to m_\nu^a + \bar{m}^a$ and imposing the definitions of the order parameters in equations (8.4), (8.2) and (8.12), the partition function reads

$$\begin{aligned}\langle Z^n \rangle = e^{-\frac{\beta}{2}Pn} \sum_{\{s_i^a\}} \int \prod_{\nu a} \frac{dm_\nu^a}{\sqrt{2\pi}} \prod_{ab} \frac{dq^{ab} d\hat{q}^{ab}}{2\pi} \prod_{ab} \frac{dp^{ab} d\hat{p}^{ab}}{2\pi} \prod_{k>1,a} \frac{d\mu_k^a d\hat{\mu}_k^a}{2\pi} \prod_{1a} \frac{d\mu_1^a \hat{\mu}_1^a}{2\pi} \left\langle e^{\frac{\beta}{2}\sum_{\nu a}(m_\nu^a + \bar{m}^a)^2} \right\rangle_{c_{\nu 1}} \\ \times \left\langle \left(\prod_\nu \frac{1}{\sqrt{\det Q}} e^{-\frac{1}{2}\sum_{ab} m_\nu^a (Q^{-1})^{ab} m_\nu^b} \right) e^{-\alpha N \sum_{a<b} \hat{q}^{ab}\left(q^{ab} - \frac{1}{N}\sum_i s_i^a s_i^b\right) - \alpha N \sum_{a \le b} \hat{p}^{ab}\left(p^{ab} - \frac{1}{D}\sum_{k>1} \mu_k^a \mu_k^b\right)} \right. \\ \left. \times \prod_{k>1,a} e^{i\hat{\mu}_k^a\left(\mu_k^a - \frac{1}{\sqrt{N}}\sum_i f_{ki} s_i^a\right)} \prod_a e^{i\hat{\mu}_1^a\left(\mu_1^a - \frac{1}{N}\sum_i f_{1i} s_i^a\right)} \right\rangle_{f_{ki}}\end{aligned} \tag{D.1}$$

We can now average over $c_{\nu 1}$, supposing that it is a standard normal distribuited random variable. For each $\nu \in [P]$ we have

$$\begin{aligned}\left\langle e^{\frac{\beta}{2}\sum_a \left(m_\nu^a + c_{\nu 1}\frac{\mu_1^a \kappa_1}{\sqrt{\alpha_D}}\right)^2} \right\rangle_{c_{\nu 1}} &= e^{\frac{\beta}{2}\sum_a (m_\nu^a)^2} \left\langle e^{c_{\nu 1}\left(\frac{\beta\kappa_1}{\sqrt{\alpha_D}}\sum_a m_\nu^a \mu_1^a\right) + \frac{1}{2}c_{\nu 1}^2\left(\frac{\beta\kappa_1^2}{\alpha_D}\sum_a \left(\mu_1^a\right)^2\right)} \right\rangle_{c_{\nu 1}} \\ &= \frac{1}{\sqrt{1 - \beta\frac{\kappa_1^2}{\alpha_D}\sum_a \mu_{a1}^2}} e^{\frac{\beta}{2}\sum_a (m_\nu^a)^2 + \frac{\beta^2\kappa_1^2}{2C\alpha_D}\left(\sum_a m_\nu^a \mu_{a1}\right)^2}\end{aligned} \tag{D.2}$$

being

$$C = 1 - \beta\frac{\kappa_1^2}{\alpha_D}\sum_a \mu_{a1}^2. \tag{D.3}$$

It is also straightforward to integrate over the vanishing pattern magnetizations, having then

$$\begin{aligned}\langle Z^n\rangle = e^{-\frac{\beta}{2}Pn}\sum_{\{s_i^a\}}\int\prod_{ab}\frac{dq^{ab}d\hat{q}^{ab}}{2\pi}\prod_{ab}\frac{dp^{ab}d\hat{p}^{ab}}{2\pi}\prod_{k>1,a}\frac{d\mu_k^a\hat{\mu}_k^a}{2\pi}\prod_{1a}\frac{d\mu_1^a\hat{\mu}_1^a}{2\pi}\\ \times\Bigg\langle e^{-\frac{\alpha N}{2}\ln\left(1-\beta\frac{\kappa_1^2}{\alpha_D}\sum_a\mu_{a1}^2\right)-\frac{\alpha N}{2}\ln\det\left(\mathbb{I}-\beta Q-\frac{\beta^2}{C}\frac{\kappa_1^2}{\alpha_D}p_1Q\right)-\frac{\alpha N}{2}\sum_{a\neq b}\hat{q}^{ab}q^{ab}+\frac{\alpha}{2}\sum_{a\neq b}\hat{q}^{ab}\sum_i s_i^a s_i^b}\\ \times e^{-\frac{\alpha N}{2}\sum_{ab}\hat{p}^{ab}p^{ab}+\frac{\alpha_T}{2}\sum_{ab}\hat{p}^{ab}\sum_{k>1}\mu_k^a\mu_k^b}\\ \times e^{i\sum_{k>1,a}\hat{\mu}_k^a\mu_k^a-\frac{1}{\sqrt{N}}i\sum_{k>1,a}\hat{\mu}_k^a\sum_i f_{ki}s_i^a+i\sum_a\hat{\mu}_1^a\mu_1^a-\frac{1}{N}i\sum_a\hat{\mu}_1^a\sum_i f_{1i}s_i^a}\Bigg\rangle_{f_{ki}}\end{aligned} \tag{D.4}$$

and where we have defined a new variable

$$p_1^{ab}\equiv\mu_1^a\mu_1^b. \tag{D.5}$$

Notice that now the terms depending on the indices $k>1$ and $k=1$ are decoupled as it should be. Performing the integral over the feature magnetizations, which reads

$$Z_1=\int\prod_{k>1,a}\frac{d\mu_k^a d\hat{\mu}_k^a}{2\pi}\exp\left\{\frac{\alpha_T}{2}\sum_{ab}\hat{p}^{ab}\sum_{k>1}\mu_k^a\mu_k^b+i\sum_{k>1,a}\hat{\mu}_k^a\mu_k^a\right\}\left\langle\exp\left\{-\frac{1}{\sqrt{N}}i\sum_{ka}\hat{\mu}_k^a\sum_i f_{ki}s_i^a\right\}\right\rangle_{\{f_{ki}\}_{k>1}} \tag{D.6}$$

needs to compute the disorder average over f_{ki} with $k>1$ and $i\in[N]$

$$\left\langle\prod_{k>1,a}\exp\left\{-\frac{i}{\sqrt{N}}\hat{\mu}_k^a\sum_i f_{ki}s_i^a\right\}\right\rangle_{\{f_{ki}\}_{k>1}}=e^{-\frac{1}{2}\sum_{ab}\left(\sum_{k>1}\hat{\mu}_k^a\hat{\mu}_k^b\right)\left(\frac{1}{N}\sum_i s_i^a s_i^b\right)}. \tag{D.7}$$

The result would have been the same if $f_{ki}=\pm1$ with equal probability. We then recognize that the integral in equation (D.6) is factorized over $k>1$, leaving the opportunity to write $Z_1=e^{\alpha_D N\phi_1}$, with

$$\phi_1=\ln\int\prod_a\frac{d\mu^a d\hat{\mu}^a}{2\pi}\exp\left\{i\sum_a\hat{\mu}^a\mu^a-\frac{1}{2}\sum_{ab}q^{ab}\hat{\mu}^a\hat{\mu}^b+\frac{\alpha_T}{2}\sum_{ab}\hat{p}^{ab}\mu^a\mu^b\right\}.$$

Here, we also have replaced the definition of q^{ab} in equation (D.7) and we have included for simplicity the diagonal term $q^{aa}=1$ in the summation $\frac{1}{2}\sum_{ab}q^{ab}\hat{\mu}^a\hat{\mu}^b$. At this point we can integrate over the variables $\hat{\mu}^a,\mu^a$, getting

$$\phi_1=-\frac{1}{2}\ln\det\left(\mathbb{I}-\frac{\alpha}{\alpha_D}q\hat{p}\right). \tag{D.8}$$

Rearranging the terms and scaling $\hat{\mu}_1^a \to iN\hat{\mu}_1^a$, we get

$$\begin{aligned}\langle Z^n \rangle = e^{-\frac{\beta}{2}Pn} \int \prod_{ab} \frac{dq^{ab}d\hat{q}^{ab}}{2\pi} \prod_{ab} \frac{dp^{ab}d\hat{p}^{ab}}{2\pi} \prod_{1a} \frac{d\mu_1^a \hat{\mu}_1^a}{2\pi} \; e^{-\frac{\alpha N}{2}\sum_{a\neq b}\hat{q}^{ab}q^{ab} - \frac{\alpha N}{2}\sum_{ab}\hat{p}^{ab}p^{ab} - N\sum_a \hat{\mu}_1^a\mu_1^a} \\ \times e^{-\frac{\alpha N}{2}\ln\left(1-\beta\frac{\kappa_1^2}{\alpha_D}\sum_a \mu_{a1}^2\right) - \frac{\alpha N}{2}\ln\det\left(\mathbb{I}-\beta Q - \frac{\beta^2}{C}\frac{\kappa_1^2}{\alpha_D}p_1 Q\right) - \frac{\alpha_D N}{2}\ln\det\left(\mathbb{I}-\frac{\alpha}{\alpha_D}q\hat{p}\right)} \\ \times \left\langle \sum_{\{s_i^a\}} e^{\frac{\alpha}{2}\sum_{a\neq b}\hat{q}^{ab}\sum_i s_i^a s_i^b + \sum_a \hat{\mu}_1^a \sum_i f_{1i} s_i^a} \right\rangle_{f_{1i}} .\end{aligned} \tag{D.9}$$

D.1.1 RS - Ansatz

We impose an RS ansatz over all the order parameters

$$\mu_{k'}^a = \mu_{k'} \tag{D.10a}$$
$$\hat{\mu}_{k'}^a = \hat{\mu}_{k'} \tag{D.10b}$$
$$q^{ab} = \delta^{ab} + q(1-\delta^{ab}) \tag{D.10c}$$
$$\hat{q}^{ab} = \delta^{ab} + \hat{q}(1-\delta^{ab}) \tag{D.10d}$$
$$p^{ab} = p_{\mathrm{d}}\delta^{ab} + p(1-\delta^{ab}) \tag{D.10e}$$
$$\hat{p}^{ab} = \hat{p}_{\mathrm{d}}\delta^{ab} + \hat{p}(1-\delta^{ab}) \tag{D.10f}$$

and if for convenience we also define

$$Q_{ab} = Q_{\mathrm{d}}\delta_{ab} + Q\,(1-\delta_{ab}) \tag{D.11}$$

where we call Q_d and Q the diagonal and off-diagonal elements of the matrix Q respectively

$$\begin{aligned} Q_{\mathrm{d}} &= \kappa_*^2 + \kappa_1^2 p_{\mathrm{d}} \\ Q &= \kappa_*^2 q + \kappa_1^2 p \end{aligned}$$

then the ansatz allows to linearize the $s^a s^b$ in the term

$$\ln\left\langle e^{-\frac{1}{2}\alpha\hat{q}n}\sum_{\{s^a\}} e^{\frac{1}{2}\alpha\hat{q}\left(\sum_a s^a\right)^2 + f_1\sum_a \hat{\mu}_1 s^a}\right\rangle_{f_1} = -\frac{1}{2}\alpha\hat{q}n + n\left\langle \int Dz\, \ln\left[2\cosh\left(z\sqrt{\alpha\hat{q}} + \hat{\mu}_1 f_1\right)\right]\right\rangle_{f_1} . \tag{D.12}$$

To compute the determinants in equation (D.9) we can express the generic RS matrix as $X^{ab} = x_{\mathrm{d}}\delta_{ab} + x\,(1-\delta_{ab})$ and so the following holds

$$\ln\det X^{ab} = n\ln\left(x_{\mathrm{d}} - x\right) + n\frac{x}{x_{\mathrm{d}} - x} + O(n^2). \tag{D.13}$$

We therefore have

$$\begin{aligned} D_0 &= \ln\det\left(\mathbb{I} - \beta Q - \frac{\beta^2}{C}\frac{\kappa_1^2}{\alpha_D}p_1 Q\right) \\ &= n\ln\left[1-\beta\left(Q_{\mathrm{d}} - Q\right)\right] - n\frac{\beta Q + \frac{\beta^2\kappa_1^2}{\alpha_D}\mu_1^2\left(Q_{\mathrm{d}} - Q\right)}{1-\beta\left(Q_{\mathrm{d}} - Q\right)} + O(n^2) \end{aligned} \tag{D.14}$$

and

$$
\begin{aligned}
D_1 &= \ln\det\left(\mathbb{I} - \alpha_T\, q\hat{p}\right) \\
&= n\ln\left[1 - \alpha_T\left(\hat{p}_\mathrm{d} - \hat{p}\right)(1-q)\right] - n\frac{\alpha_T\left(\hat{p} + \hat{p}_\mathrm{d} q - 2q\hat{p}\right)}{1 - \alpha_T\left(\hat{p}_\mathrm{d} - \hat{p}\right)(1-q)} + O(n^2).
\end{aligned}
\tag{D.15}
$$

Free energy

In the end, the free energy expression reads

$$
\begin{aligned}
f^{\mathrm{RS}} =& \frac{\alpha}{2} - \frac{\alpha}{2}\beta\hat{q}\,(q-1) + \frac{\alpha}{2}\hat{p}_\mathrm{d} p_\mathrm{d} - \frac{\alpha}{2}\hat{p}p - \frac{\alpha}{2}\frac{\kappa_1^2}{\alpha_D}\mu^2 + \hat{\mu}\mu \\
&+ \frac{\alpha}{2\beta}\left[\ln\left[1 - \beta\left(Q_\mathrm{d} - Q\right)\right] - \frac{\beta Q + \beta^2\frac{\kappa_1^2}{\alpha_D}\mu_1^2\left(Q_\mathrm{d} - Q\right)}{1 - \beta\left(Q_\mathrm{d} - Q\right)}\right] \\
&+ \frac{\alpha_D}{2\beta}\left[\ln\left(1 - \alpha_T\beta\left(\hat{p}_\mathrm{d} - \hat{p}\right)(1-q)\right) - \frac{\alpha_T\beta\left(\hat{p} + \hat{p}_\mathrm{d} q - 2q\hat{p}\right)}{1 - \alpha_T\beta\left(\hat{p}_\mathrm{d} - \hat{p}\right)(1-q)}\right] \\
&- \frac{1}{\beta}\left\langle \int Dz\, \ln\left[2\cosh\left(\beta\left[z\sqrt{\alpha\hat{q}} + \hat{\mu}f\right]\right)\right]\right\rangle_f
\end{aligned}
\tag{D.16}
$$

where we have removed the index from the order parameters μ_1 and $\hat{\mu}_1$ and rescaled the order parameters with β to get rid of the factor in the f^{RS} definition. Taking the derivatives of the free energy in equation (D.16) we get the saddle point equations

$$q = \int Dz\, \tanh^2\left(\beta\left[z\sqrt{\alpha\hat{q}} + \hat{\mu}\right]\right) \tag{D.17a}$$

$$\mu = \int Dz\, \tanh\left(\beta\left[z\sqrt{\alpha\hat{q}} + \hat{\mu}\right]\right) \tag{D.17b}$$

$$\hat{q} = \frac{\kappa_*^2(\kappa_1^2 p + \kappa_*^2 q)}{(1 + \beta\kappa_1^2(p - p_\mathrm{d}) + \beta\kappa_*^2(q-1))^2} + \frac{\hat{p} + \frac{\alpha}{\alpha_D}\beta q(\hat{p} - \hat{p}_\mathrm{d})^2}{\beta(\frac{\alpha}{\alpha_D}\beta(q-1)(\hat{p} - \hat{p}_\mathrm{d}) - 1)^2} \tag{D.17c}$$

$$\hat{p} = \frac{\beta\kappa_1^2(\kappa_1^2 p + \kappa_*^2 q)}{(1 + \beta\kappa_1^2(p - p_\mathrm{d}) + \beta\kappa_*^2(q-1))^2} \tag{D.17d}$$

$$\hat{p}_\mathrm{d} = \frac{\kappa_1^2(1 + \beta\kappa_1^2(2p - p_\mathrm{d}) + \beta\kappa_*^2(2q-1))}{(1 + \beta\kappa_1^2(p - p_\mathrm{d}) + \beta\kappa_*^2(q-1))^2} \tag{D.17e}$$

$$p = \frac{1}{\alpha_D}\mu^2 + \frac{q + \frac{\alpha}{\alpha_D}\beta\hat{p}(q-1)^2}{(\frac{\alpha}{\alpha_D}\beta(q-1)(\hat{p} - \hat{p}_\mathrm{d}) - 1)^2} \tag{D.17f}$$

$$p_\mathrm{d} = \frac{1}{\alpha_D}\mu^2 + \frac{1 + \frac{\alpha}{\alpha_D}\beta(2\hat{p} - \hat{p}_\mathrm{d})(q-1)^2}{(\frac{\alpha}{\alpha_D}\beta(q-1)(\hat{p} - \hat{p}_\mathrm{d}) - 1)^2} \tag{D.17g}$$

$$\hat{\mu} = \frac{\alpha}{\alpha_D}(\hat{p}_\mathrm{d} - \hat{p})\mu. \tag{D.17h}$$

Limit $\beta \to \infty$

In the infinite β limit the order parameters scale, as it can be seen by inspection, as

$$q = 1 - \frac{\delta q}{\beta} \tag{D.18a}$$

$$p = p_d - \frac{\delta p}{\beta} \tag{D.18b}$$

$$\hat{p} = \beta\, \delta\hat{p}_d - \frac{1}{2}\delta\hat{p} \tag{D.18c}$$

$$\hat{p}_d = \beta\, \delta\hat{p}_d + \frac{1}{2}\delta\hat{p} \tag{D.18d}$$

so that the difference $\hat{p}_d - \hat{p} = \delta\hat{p}$ remains finite. Using the scalings, the new saddle point equations become

$$\delta q = \frac{d}{dx} H\left(-\frac{\hat{\mu}+x}{\sqrt{\alpha\hat{q}}}\right)\Big|_{x=0} = \frac{2}{\sqrt{\alpha\hat{q}}} G\left(-\frac{\hat{\mu}}{\sqrt{\alpha\hat{q}}}\right) \tag{D.19a}$$

$$\mu = 2H\left(-\frac{\hat{\mu}}{\sqrt{\alpha\hat{q}}}\right) - 1 \tag{D.19b}$$

$$\hat{q} = \frac{\kappa_\star^2(\kappa_1^2 p_\mathrm{d} + \kappa_\star^2)}{(1-\kappa_1^2\delta p - \kappa_\star^2\delta q)^2} + \frac{\delta\hat{p}_\mathrm{d} + \frac{\alpha}{\alpha_D}\delta\hat{p}^2}{(1 - \frac{\alpha}{\alpha_D}\,\delta q\,\delta\hat{p})^2} \tag{D.19c}$$

$$\delta\hat{p} = \frac{\kappa_1^2}{1-\kappa_1^2\delta p - \kappa_\star^2\delta q} = \frac{\kappa_1^2}{1-\delta Q} \tag{D.19d}$$

$$\delta\hat{p}_\mathrm{d} = \frac{\kappa_1^2(\kappa_1^2 p_\mathrm{d} + \kappa_*^2)}{(1-\kappa_1^2\delta p - \kappa_\star^2\delta q)^2} = \frac{\kappa_1^2\, Q_\mathrm{d}}{(1-\delta Q)^2} \tag{D.19e}$$

$$\delta p = \beta(p_d - p) = \frac{\delta q}{1 - \frac{\alpha}{\alpha_D}\,\delta q\,\delta\hat{p}} \tag{D.19f}$$

$$p_\mathrm{d} = \frac{1}{\alpha_D}\mu^2 + \frac{1 + \frac{\alpha}{\alpha_D}\delta q^2\delta\hat{p}_\mathrm{d}}{(1 - \frac{\alpha}{\alpha_D}\,\delta q\,\delta\hat{p})^2} \tag{D.19g}$$

$$\hat{\mu} = \frac{\alpha}{\alpha_D}\,\delta\hat{p}\,\mu \tag{D.19h}$$

where we used the identity $1 - \tanh^2(x) = \frac{d}{dx}\tanh(x)$ for the first equation. It is now easy, from the zero temperature expressions, to perform the $\alpha \to \infty$ limit that reduces to the standard Hopfield Model for the feature magnetization case. In this limit, the scaling are as follows

$$\delta q \to \frac{\alpha_D}{\alpha}\delta q \tag{D.20a}$$

$$\hat{q} \to \frac{\alpha}{\alpha_D}\hat{q} \tag{D.20b}$$

$$\delta p \to \frac{\alpha_D}{\alpha}\delta p \tag{D.20c}$$

$$\hat{\mu} \to \frac{\alpha}{\alpha_D}\hat{\mu}. \tag{D.20d}$$

Here the value of μ and $\hat{\mu}$ change in different phases: for low α_D and when retrieval is possible, then $\mu \to 1$ and $\hat{\mu} \to \infty$. It is therefore necessary to scale also μ with α. Simplifying the equations and by rescaling $\kappa_1^2 \delta q \to \delta q$ and $\hat{q} \to \kappa_1^4 \hat{q}$, we obtain the standard HM equations for the features

$$\delta q = \frac{2}{\sqrt{\alpha_D \hat{q}}} G\left(-\frac{\mu}{\sqrt{\alpha_D \hat{q}}}\right) \tag{D.21a}$$

$$\mu = 2H\left(-\frac{\mu}{\sqrt{\alpha_D \hat{q}}}\right) - 1 \tag{D.21b}$$

$$\hat{q} = \frac{1}{(1-\delta q)^2} \tag{D.21c}$$

as anticipated in the main text when performing the $P \to \infty$ limit (8.19).

D.2 Pattern retrieval

We have to start again from the replicated partition function

$$\langle Z^n \rangle = \sum_{\{s_i^a\}} \left\langle e^{\frac{\beta}{2N} \sum_\nu \left(\sum_i \sigma\left(\frac{1}{\sqrt{D}} \sum_k c_{\nu k} f_{ki} \right) s_i^a \right)^2} \right\rangle_{c,f} \tag{D.22}$$

and since we want only one magnetization with the pattern m_1^a (8.2) to be of order $\mathcal{O}(1)$ while the rest of order $\mathcal{O}(\frac{1}{\sqrt{N}})$ (see the ansatz (8.20)), we rescale properly the finite magnetization $m_1^a \to \sqrt{N} m_1^a$

$$\langle Z^n \rangle = \sum_{\{s_i^a\}} \int \prod_{\nu a} \frac{dm_\nu^a}{\sqrt{2\pi}} e^{\frac{\beta N}{2} \sum_a (m_1^a)^2 + \frac{\beta}{2} \sum_{\nu>1} \sum_a (m_\nu^a)^2} \left\langle \prod_a \delta\left(\sqrt{N} m_1^a - \frac{1}{\sqrt{N}} \sum_i \sigma\left(\frac{1}{\sqrt{D}} \sum_k c_{1k} f_{ki} \right) s_i^a \right) \right.$$
$$\left. \prod_{a,\nu>1} \delta\left(m_\nu^a - \frac{1}{\sqrt{N}} \sum_i \sigma\left(\frac{1}{\sqrt{D}} \sum_k c_{\nu k} f_{ki} \right) s_i^a \right) \right\rangle_{c,f} . \tag{D.23}$$

By computing the average over the $P-1$ patterns $c_{\nu k}$ with $\nu > 1$ using GEP, one can integrate the vanishing magnetizations in the thermodynamic limit, having

$$\langle Z^n \rangle = \sum_{\{s_i^a\}} \int \prod_{\nu a} \frac{dm^a}{\sqrt{2\pi}} \left\langle e^{-\frac{\beta N}{2} \sum_a (m^a)^2 + \beta \sum_{ia} m^a \sigma\left(\frac{1}{\sqrt{D}} \sum_k c_k f_{ki} \right) s_i^a - \frac{\alpha N}{2} \ln \det(\mathbb{I} - \beta Q)} \right\rangle_{c,f} \tag{D.24}$$

where we have removed the index "1" from c_{1k} and m_1^a for simplicity. Introducing the order parameters as in (8.4), (8.2) and (8.12), the average over the features means to

extract the argument from the non-linearity $\sigma(\cdot)$

$$\begin{aligned}\langle Z^n\rangle = \sum_{\{s_i^a\}} \int \prod_a \frac{dm^a}{2\pi} \prod_{a<b} \frac{dq^{ab}\, d\hat{q}^{ab}}{2\pi} \prod_{a\leq b} \frac{dp^{ab}\, d\hat{p}^{ab}}{2\pi} \prod_{ak} \frac{d\mu_k^a\, d\hat{\mu}_k^a}{2\pi} \prod_i \frac{dv_i d\hat{v}_i}{2\pi}\, e^{-\frac{N\alpha}{2}\sum_{a\neq b} q^{ab}\,\hat{q}^{ab}} \\ \times\, e^{-\frac{N\alpha}{2}\sum_{ab} p^{ab}\hat{p}^{ab} + i\sum_{ak}\mu_k^a\hat{\mu}_k^a + i\sum_i v_i\hat{v}_i - \frac{\beta N}{2}\sum_a (m^a)^2 - \frac{\alpha N}{2}\ln\det(\mathbb{I}-\beta Q) + \frac{\alpha}{2}\sum_{a\neq b}\hat{q}^{ab}\sum_i s_i^a s_i^b} \\ \times\, e^{\frac{\alpha_T}{2}\sum_{ab}\hat{p}^{ab}\sum_k \mu_k^a\mu_k^b + \beta\sum_{ia} m^a\sigma(v_i)s_i^a} \left\langle e^{-i\sum_{ki} f_{ki}\left(\frac{1}{\sqrt{N}}\sum_a \hat{\mu}_k^a s_i^a + \frac{1}{\sqrt{D}}\hat{v}_i c_k\right)}\right\rangle_{c,f}\end{aligned} \tag{D.25}$$

and since it factorizes over $k \in [D]$, it becomes trivial to perform the average over the feature magnetizations, leaving in the end

$$\begin{aligned}\langle Z^n\rangle = \sum_{\{s_i^a\}} \int \prod_a \frac{dm^a}{2\pi} \prod_{a<b} \frac{dq^{ab}\, d\hat{q}^{ab}}{2\pi} \prod_{a\leq b} \frac{dp^{ab}\, d\hat{p}^{ab}}{2\pi} \prod_i \frac{dv_i d\hat{v}_i}{2\pi} \prod_a \frac{dt^a d\hat{t}^a}{2\pi}\, e^{-\frac{N\alpha}{2}\sum_{a\neq b} q^{ab}\,\hat{q}^{ab}} \\ \times\, e^{-\frac{N\alpha}{2}\sum_{ab} p^{ab}\hat{p}^{ab} + i\sum_i v_i\hat{v}_i + iN\sum_a t^a\hat{t}^a - \frac{\beta N}{2}\sum_a (m^a)^2 - \frac{\alpha N}{2}\ln\det(\mathbb{I}-\beta Q) - \frac{\alpha_D N}{2}\ln\det(\mathbb{I}-\alpha_T\, q\hat{p})} \\ \times\, e^{\frac{\alpha}{2}\sum_{a\neq b}\hat{q}^{ab}\sum_i s_i^a s_i^b + \beta\sum_{ia} m^a\sigma(v_i)s_i^a - \frac{1}{2}\sum_i \hat{v}_i^2 - \frac{\alpha_T N}{2}\sum_{ab} t^a\left(\hat{p}^{-1}-\alpha_T\, q\right)^{-1}_{ab} t^b - i\sum_a \hat{t}^a\sum_i \hat{v}_i s_i^a}\end{aligned} \tag{D.26}$$

where

$$t^a = \frac{1}{N}\sum_i \hat{v}_i s_i^a. \tag{D.27}$$

Now we rearrange all the terms depending on $v, \hat{v}$ and on the spins

$$\sum_{\{s_i^a\}} \int \prod_i \frac{dv_i\, d\hat{v}_i}{2\pi}\, e^{-\frac{1}{2}\sum_i \hat{v}_i^2 + i\sum_i \hat{v}_i\left(v_i - \sum_a \hat{t}^a s_i^a\right) + \frac{\alpha}{2}\sum_{a\neq b}\hat{q}^{ab}\sum_i s_i^a s_i^b + \beta\sum_a m^a\sum_i \sigma(v_i)s_i^a} = e^{N\phi_2} \tag{D.28}$$

and integrating over the conjugated parameter we get

$$\phi_2 = \ln\int Dv \sum_{\{s^a\}} e^{-\frac{1}{2}\sum_{ab}\left(\alpha\,\hat{q}^{ab} - \hat{t}^a\hat{t}^b\right)s^a s^b + \sum_a s^a\left(v\,\hat{t}^a + \beta\,\sigma(v)\,m^a\right)}. \tag{D.29}$$

Scaling the order parameters with β: $\hat{q}^{ab} \to \beta^2\,\hat{q}^{ab}$, $\hat{t}^a \to \beta\,\hat{t}^a$, $\hat{p}^{ab} \to \beta\,\hat{p}^{ab}$ and recognizing that on the saddle point t^a is purely imaginary $\hat{t}^a \to i\hat{t}^a$, in the end

$$\begin{aligned}\langle Z^n\rangle = \int \prod_a \frac{dm^a}{2\pi} \prod_{a<b} \frac{dq^{ab}\, d\hat{q}^{ab}}{2\pi} \prod_{a\leq b} \frac{dp^{ab}\, d\hat{p}^{ab}}{2\pi} \prod_a \frac{dt^a d\hat{t}^a}{2\pi}\, e^{-\frac{N\alpha\beta^2}{2}\sum_{a\neq b} q^{ab}\,\hat{q}^{ab} - \frac{N\alpha\beta}{2}\sum_{ab} p^{ab}\hat{p}^{ab}} \\ \times\, e^{-N\beta\sum_a t^a\hat{t}^a - \frac{\beta N}{2}\sum_a (m^a)^2 - \frac{\alpha N}{2}\ln\det(\mathbb{I}-\beta Q) - \frac{\alpha_D N}{2}\ln\det(\mathbb{I}-\alpha_T\beta\, q\hat{p})} \\ \times\, e^{\frac{\alpha_T N}{2}\sum_{ab} t^a\left(\hat{p}^{-1}-\alpha_T\, q\right)^{-1}_{ab} t^b + N\phi_2}.\end{aligned} \tag{D.30}$$

D.2.1 RS - Ansatz

The free energy can be now easily expressed as a function of the RS order parameters as before (D.10)

$$
\begin{aligned}
f_{\mathrm{RS}} = & \frac{1}{2}m^2 - \frac{\beta\alpha}{2}\hat{q}(q-1) + \frac{\alpha}{2}p_d\hat{p}_d - \frac{\alpha}{2}p\hat{p} + t\hat{t} - \frac{\alpha_T}{2}\frac{t^2\,\hat{p}_d}{(1-\beta\alpha_T\,\hat{p}_d)} + \frac{\alpha_T}{2}\frac{t^2\,\hat{p}}{(1-\beta\alpha_T\,q\,\hat{p})} \\
& + \frac{\alpha}{2\beta}\left[\ln\left(1-\beta\left(Q_{\mathrm{d}}-Q\right)\right) - \frac{\beta Q}{1-\beta(Q_{\mathrm{d}}-Q)}\right] \\
& + \frac{\alpha}{2\beta\alpha_T}\left[\ln\left(1-\beta\alpha_T(\hat{p}_d-\hat{p})(1-q)\right) - \frac{\beta\alpha_T\left(\hat{p}+q\hat{p}_d-2q\hat{p}\right)}{1-\beta\alpha_T(\hat{p}_d-\hat{p})(1-q)}\right] \\
& - \frac{1}{\beta}\int Dx \int Dv\ \ln 2\cosh\left[\beta\left(v\,\hat{t}+\sigma(v)\,m+\sqrt{\alpha\hat{q}-\hat{t}^2}\,x\right)\right]
\end{aligned}
\tag{D.31}
$$

where we can separate the covariance matrix between diagonal and off-diagonal contributions

$$
Q = \kappa_*^2\,q + \kappa_1^2\,p \tag{D.32a}
$$

$$
Q_{\mathrm{d}} = \kappa_*^2 + \kappa_1^2\,p_d. \tag{D.32b}
$$

Limit $\beta \to \infty$

In the $\beta \to \infty$ limit, by choosing $\sigma(\cdot) = \mathrm{sign}(\cdot)$ as activation function, the scaling for the order parameters turn out to be

$$
q \to 1 - \frac{\delta q}{\beta}, \quad p = p_d - \frac{\delta p}{\beta}, \quad \hat{p} = \beta\delta\hat{p}_d - \frac{1}{2}\delta\hat{p}, \quad \hat{p}_d = \beta\delta\hat{p}_d + \frac{1}{2}\delta\hat{p}, \quad \hat{t} \to \frac{\delta\hat{t}}{\beta^2}
$$

from which on can derive how the free energy and so the saddle point equations become in the limit

$$
\begin{aligned}
\delta q &= \frac{2}{\sqrt{\alpha\,\hat{q}}}\,G\left(-\frac{m}{\sqrt{\alpha\,\hat{q}}}\right) \\
t &= \frac{2\,m}{\pi\sqrt{\alpha\,\hat{q}-\hat{t}^2}} \\
m &= 2H\left(-\frac{m}{\sqrt{\alpha\,\hat{q}}}\right) - 1 \\
p_d &= \frac{1+\frac{\alpha}{\alpha_D}\,\delta q^2\delta\hat{p}_d}{(1-\frac{\alpha}{\alpha_D}\delta q\,\delta\hat{p})^2} \\
\delta p &= \beta\left(p_d - p\right) = \frac{\delta q}{1-\frac{\alpha}{\alpha_D}\,\delta q\,\delta\hat{p}} \\
\hat{q} &= \frac{\kappa_\star^2(\kappa_1^2\,p_{\mathrm{d}}+\kappa_\star^2)}{(1-\kappa_1^2\delta p-\kappa_\star^2\delta q)^2} + \frac{\delta\hat{p}_{\mathrm{d}}+\frac{\alpha}{\alpha_D}\delta\hat{p}^2}{(1-\frac{\alpha}{\alpha_D}\,\delta q\,\delta\hat{p})^2}
\end{aligned}
$$

$$\delta\hat{t} = \delta q\, t$$
$$\delta\hat{p} = \frac{\kappa_1^2}{1 - \kappa_1^2\,\delta p - \kappa_*^2\,\delta q}$$
$$\delta\hat{p}_d = \frac{\kappa_1^2\,(\kappa_1^2\, p_d + \kappa_*^2)}{(1 - \kappa_1^2\,\delta p - \kappa_*^2\,\delta q)^2}.$$

Limit $\alpha_D \to \infty$

Taking the $\alpha_D \to \infty$ limit we should recover the standard Hopfield model. Indeed, the saddle point equations become

$$q = \int Dx\, \tanh^2\left[\beta\left(m + \sqrt{\alpha\,\hat{q}}\,x\right)\right] \tag{D.36a}$$
$$m = \int Dx\, \tanh\left[\beta\left(m + \sqrt{\alpha\,\hat{q}}\,x\right)\right] \tag{D.36b}$$
$$p \to q \tag{D.36c}$$
$$p_d \to 1 \tag{D.36d}$$
$$\hat{q} = \frac{\hat{p}}{\beta} + \frac{q\,\kappa_*^2(\kappa_1^2 + \kappa_*^2)}{\left[1 - \beta(1-q)(\kappa_1^2 + \kappa_*^2)\right]^2} = \frac{(\kappa_1^2 + \kappa_*^2)^2\, q}{\left[1 - \beta(1-q)(\kappa_1^2 + \kappa_*^2)\right]^2} \tag{D.36e}$$
$$\hat{p} = \frac{\beta\, q\,\kappa_1^2(\kappa_1^2 + \kappa_*^2)}{\left[1 - \beta(1-q)(\kappa_1^2 + \kappa_*^2)\right]^2} \tag{D.36f}$$
$$\hat{p}_d = \frac{\kappa_1^2\,(1 - \beta\,(1-2q)\,(\kappa_1^2 + \kappa_*^2))}{\left[1 - \beta(1-q)(\kappa_1^2 + \kappa_*^2)\right]^2} \tag{D.36g}$$
$$\hat{t} \to 0 \tag{D.36h}$$

where $\kappa_1^2 + \kappa_*^2 = 1$ for $\sigma(v) = \text{sign}(v)$.

D.3 Numerical simulations

In order to implement numerical simulations of the asynchronous update rule

$$s_i^{(t+1)} = \text{sign}\left(\sum_{j(\neq i)}^{N} J_{ij} s_j^{(t)}\right)$$

to evaluate the magnetization with a pattern or a feature, we initialize the spins to the attractor we would like to recover, and once found the fixed point of the dynamics, we compute the overlap with the initial condition.
The pseudo-code, which is trivial, is

Algorithm 2 Zero-temperature asynchronous dynamics

Require: coupling matrix J_{ij}, initial condition $\boldsymbol{\xi}_\nu$ (or $\mathbf{f}_k$)
  $\boldsymbol{s} \leftarrow \boldsymbol{\xi}_\nu$ (or $\boldsymbol{s} \leftarrow \mathbf{f}_k$) ▷ Initialize the spins to one of the examples (or feature)
  while true **do**
    for $i \in \pi([1, ..., N])$ **do** ▷ $\pi()$ is a random permutation
      $h_i \leftarrow \sum_{j(\neq i)}^{N} J_{ij} s_j$ ▷ Compute the local field
      $s_i \leftarrow \text{sign}(h_i)$ ▷ Update the spin
    end for
    if $s_i = \text{sign}(h_i)\ \forall i$ **then**
      break ▷ Stop at fixed point
    end if
  end while
  $m_\nu \leftarrow \frac{1}{N} \sum_{i=1}^{N} s_i \xi_{\nu i}$ (or $\mu_k \leftarrow \frac{1}{N} \sum_{i=1}^{N} s_i f_{ki}$) ▷ Final magnetization

D.3.1 Additional figures

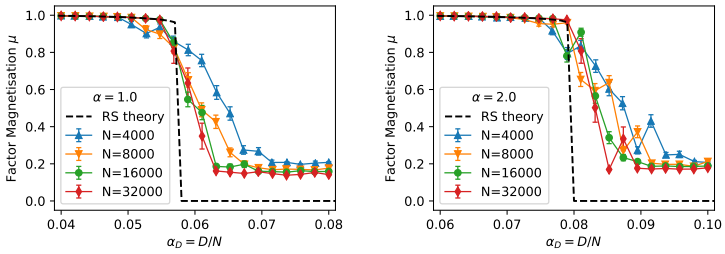


Figure D.3.1: Comparison with numerical results for the retrieval of one feature (dashed theory). Note that, as we increase α, the finite-size effects become more pronounced. Simulations are performed initializing the model to a feature $\mathbf{f}_k$, running the update rule, then measuring μ_k at convergence for increasing values of N. From [148].

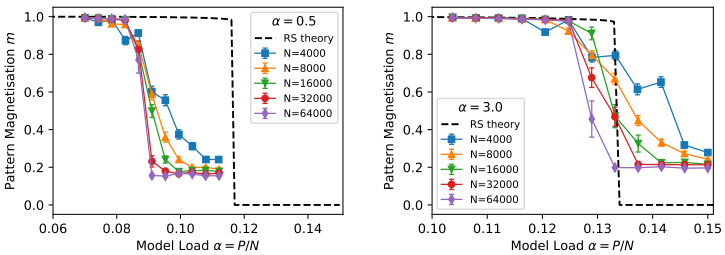


Figure D.3.2: Comparison with numerical results for the retrieval of one pattern (dashed theory). The discrepancy increases for smaller values of α_D. Simulations are performed initializing the model to a pattern $\boldsymbol{\xi}_\nu$, running the update rule, then measuring m_ν at convergence for increasing values of N. From [148].