Classifier-dependent feature selection via greedy methods

Greedy feature selection: Classifier-dependent feature selection via greedy methods

Fabiana Camattari[1,2†], Sabrina Guastavino[1,2*†], Francesco Marchetti[3], Michele Piana[1,2], Emma Perracchione[4]

[1]MIDA, Dipartimento di Matematica, Università di Genova, via Dodecaneso 35, Genova, 16145, Italy.
[2]Osservatorio Astrofisico di Torino, Istituto Nazionale di Astrofisica, via Osservatorio 20, Pino Torinese, Torino, 10025, Italy.
[3]Dipartimento di Matematica "Tullio Levi-Civita", Università di Padova, via Trieste 63, Padova, 35121, Italy.
[4]Dipartimento di Scienze Matematiche "Giuseppe Luigi Lagrange', Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino, 10129, Italy.

Contributing authors: camattari@dima.unige.it; guastavino@dima.unige.it; francesco.marchetti@unipd.it; piana@dima.unige.it; emma.perracchione@polito.it;
[†]These authors contributed equally to this work.

Abstract

The purpose of this study is to introduce a new approach to feature ranking for classification tasks, called in what follows greedy feature selection. In statistical learning, feature selection is usually realized by means of methods that are independent of the classifier applied to perform the prediction using that reduced number of features. Instead, greedy feature selection identifies the most important feature at each step and according to the selected classifier. In the paper, the benefits of such scheme are investigated theoretically in terms of model capacity indicators, such as the Vapnik-Chervonenkis (VC) dimension or the kernel alignment, and tested numerically by considering its application to the problem of predicting geo-effective manifestations of the active Sun.

Keywords: statistical learning, machine learning, classification, feature selection, greedy methods

1 Introduction

Greedy algorithms are currently mainly used to iteratively select a reduced and appropriate number of examples according to some error indicators, and hence to produce surrogate and sparse models [1–6]. The ambition of this paper is to analyze and extend greedy methods to work in the significantly more challenging case of feature reduction, i.e., as the computational core for feature-ranking schemes in the framework of classification issues.

The importance of this application follows from the fact that sparsity enhancement is a crucial issue for statistical learning procedures, which might be performed, e.g., via Fisher score [7], methods based on mutual information [8], Relief

and its variants [9]. Indeed, supervised learning models are usually trained on a reduced number of features, which are typically obtained by means of either Lasso regression [10] or variations of the classical Lasso (Group Lasso [11], Adaptive Lasso [12], Adaptive Poisson re-weighted Lasso [13] to mention a few) or linear Support Vector Machine (SVM) feature ranking [14]. However, at the state of the art, for a given classifier, these algorithms are often not able to actually capture all the corresponding most relevant features for that classifier. More specifically, in the case of Lasso and its generalizations [15], drawbacks in feature selection ability are shown when there exists dependence structures among covariates. Therefore, here we designed feature-based greedy methods that iteratively select the most important feature at each step in a classifier-dependent fashion. We point out that any classifier can be used in this scheme, which allows a totally model-dependent feature ranking process.

At a more theoretical level, this study investigated the effectiveness of the greedy scheme in terms of the Vapnik-Chervonenkis (VC) dimension [16], which is a complexity indicator common to any classifier, such as Feed-forward Neural Networks (FNNs), and it is related to the empirical risk [17]. As a particular instance, we further investigated how greedy methods behave for kernel-based classifiers, such as SVMs [18], and in doing so we considered a particular complexity score, known as kernel alignment. These theoretical findings have been used for a case study concerning the classification and prediction of severe geomagnetic events triggered by solar flares.

Solar flares [19] are the most explosive manifestations of the active Sun and the main trigger of space weather [20]. They may be followed by coronal mass ejections (CMEs) [21], which, in turn, may generate geomagnetic storms potentially impacting both space and on-earth technological assets [22]. Data-driven approaches forecasting these events leverage machine learning algorithms trained against historical archives containing physical features extracted from remote-sensing data such as solar images or time series of physical parameters acquired from in-situ instruments [23–27]. These archives systematically provide a huge amount of descriptors and it is currently well-established that this redundancy of information significantly hampers the prediction performances of the forecasters [28]. Our feature-based greedy scheme was applied in this context, in order to identify among the features the redundant ones and consequently to improve the classification performances.

The paper is organized as follows. Section 2 introduces our greedy feature selection scheme, which will be motivated thanks to the theoretical analysis in Subsections 2.1 and 2.2. Section 3 describes the application of greedy feature selection to both simulated and real datasets. Our conclusions are offered in Section 4.

2 Greedy feature ranking schemes

Feature reduction (or feature subset selection) techniques can be classified into filter, wrapper, and embedded methods: filter methods identify an optimal subset based on general patterns in data [29]; wrapper methods use a machine learning algorithm to search for the optimal subset by considering all possible feature combinations [30]; in embedded methods feature selection is integrated or built into the classifier algorithm [31]. This proposed greedy scheme falls in the class of wrapper feature subset selection methods, but unlike the classical approaches, such as recursive feature elimination (RFE) or recursive feature augmentation (RFA) [14] and forward step-wise selection [32], the proposed greedy method is fully model-dependent.

Given a set of examples depending on several features, greedy methods are frequently used to find an optimal subset of examples and, for such task, since they might be target-dependent, they have already been proved to be effective (see e.g. [4, 5, 33]). Here, instead of focusing on the examples, we drive our attention towards the problem of feature selection. To this aim, we considered a binary classification problem with training examples

$$\Xi = X \times Y = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}, \quad (1)$$

where $\boldsymbol{x}_i \in \Omega \subseteq \mathbb{R}^d$ and $y_i \in \mathbb{R}$. For the particular case of the binary classification setting, we fix $y_i \in \{-1, +1\}$.

In the machine learning framework, feature reduction is typically performed by means of linear models, and once the features are identified, non-linear methods like neural networks are applied to predict the given task. However, the fact that some specific feature could be useful for some classifier does not imply that the same feature is relevant for any classification model, and this is probably the main weakness of current feature reduction methods in this context. Conversely, our feature-based greedy method (see e.g. [34] for a general overview of greedy methods) will consist in iteratively selecting the most important feature at each step and in agreement with the considered classifier.

To reach this objective, as usually done, we split the initial dataset X into a training set, which consists of $\{(X^{(t)}, Y^{(t)})\}$, and a validation set made of $\{(X^{(v)}, Y^{(v)})\}$. Then, at the k-th greedy step, $k-1$ features have already been selected (without loosing generalities the first $k-1$) and we then train $d-k$ models $\mathcal{M}_p$ with $x_1, \dots, x_{k-1}, x_p$, $p = k, \dots, d$. Then, given an accuracy score μ (the largest the better), we select the k-th feature as

$$x_k = \arg \max_{p=k,\dots,d} \mu(\mathcal{M}_p(X^{(v)}), Y^{(v)}). \qquad (2)$$

We point out that any model can be used in (2), and this implies a totally target-dependent feature selection, which also accounts for the model used to predict a given task.

In the following we investigate the effects of the proposed scheme in terms of VC dimension and for particular instances of kernel learning theory, while a stopping criterion for the algorithm is discussed later in view of the incoming analysis and trade-off remarks.

2.1 The VC dimension in the greedy framework

We consider the dataset (1), where we now suppose that $\Omega = \bigotimes_{k=1}^{d} \Omega^k$ with $\Omega^k = [a_k, b_k] \subset \mathbb{R}$. Given a classifying function $f : \Omega \longrightarrow Y$ we consider the *zero-one loss function*

$$c(\boldsymbol{x}, y, f) = \frac{1}{2}|f(\boldsymbol{x}) - y|,$$

which is 0 if $f(\boldsymbol{x}) = y$ and 1 otherwise. From this loss, we can define the *empirical risk*

$$\hat{e}(\Xi, f) = \frac{1}{n} \sum_{i=1}^{n} c(\boldsymbol{x}_i, y_i, f).$$

Assuming that Ξ is sampled from some fixed unknown probability distribution $p(\boldsymbol{x}, y)$ on $\Omega \times Y$, we note that the empirical risk is the empirical mean value of so-called *generalization risk*, i.e.:

$$e(f) = \int_{\Omega \times Y} c(\boldsymbol{x}, y, f) \, \mathrm{d}p(\boldsymbol{x}, y),$$

i.e., it is the mean value of c averaged over all possible test samples generated by $p(\boldsymbol{x}, y)$, and hence it represents the misclassification probability. However, minimizing the empirical risk does not necessarily correspond to a low generalization risk (refer, e.g., to [35, §5] or [36, §5 & §6]). Indeed, this might lead to poor generalization capability in the sense that statistical learning theory already proved that the generalization *capacity* of a given model is somehow inversely related to the empirical risk. Such general idea can be formalized in different ways, such as via the VC dimension. In order to define it, we need to introduce the concept of *shattering* in this context. Let $\Xi_1, \dots, \Xi_{2^n}$ be all the different datasets obtainable taking all possible configurations of labels assigned to the data. A class $\mathcal{F}$ shatters the set X if for every dataset Ξ_i, $i = 1, \dots, 2^n$, there exists a function $f : \Omega \longrightarrow Y$, $f \in \mathcal{F}$, such that $\hat{e}(\Xi_i, f) = 0$.

Definition 1 The VC dimension of a class $\mathcal{F}$ of classifying functions is the largest natural number s such that there exists a set X of s examples that can be shattered by $\mathcal{F}$. If such s does not exist, then the VC dimension is ∞.

Let us consider a class $\mathcal{F}$ of classifying functions on Ω whose VC dimension is $s < n$. Then, if $f \in \mathcal{F}$ and $\delta > 0$, the bound

$$e(f) \leq \hat{e}(\Xi, f) + C(s, n, \delta),$$

holds with probability $1-\delta$, where the so-called capacity term is

$$C(s,n,\delta)=\sqrt{\frac{1}{n}\left(s\left(\log\frac{2n}{s}+1\right)+\log\frac{4}{\delta}\right)}.$$

The generalization risk (and thus the test error) is bounded by the sum between the empirical risk (that is the training error) and the capacity term of the class, which is monotonically increasing with the VC dimension. If we choose a *poor* class, we get a low VC dimension but a possibly high empirical risk; this situation is usually called *underfitting*. On the other hand, by choosing a *rich* class we can obtain a very small empirical risk, but the VC dimension, and thus the capacity term, is likely to be large; this condition is called *overfitting*. In the following, our purpose is to study how the VC dimension evolves during the greedy steps. It is natural to guess that the capacity of a classifier increases if the information contained in an added feature is considered.

Definition 2 Let $\mathcal{F}$ be a class of binary classifying functions $f:\Omega\longrightarrow Y$. Letting $\boldsymbol{e}_k$ be the k-th cardinal basis vector, we define the k-blind class $\mathcal{F}^{(k)}$, $k\in\{1,\dots,d\}$, $\mathcal{F}^{(k)}\subseteq\mathcal{F}$ as the class of functions $f^{(k)}:\Omega\longrightarrow Y$ such that

$$f^{(k)}(\boldsymbol{x})=f^{(k)}(\boldsymbol{x}+\delta\boldsymbol{e}_k),$$

for any $\delta\in\mathbb{R}$ such that $\boldsymbol{x}+\delta\boldsymbol{e}_k\in\Omega$.

For example, consider the class of functions

$$\mathcal{F}_{W,\boldsymbol{b}}:=\{f:\Omega\longrightarrow Y\mid f(\boldsymbol{x})=\tilde{f}(W\boldsymbol{x}+\boldsymbol{b})\},\quad(3)$$

where W is a $r\times d$ matrix and $\boldsymbol{b}$ is a $r\times 1$ vector, $r\geq 1$. Many well-known classifiers are included in $\mathcal{F}_{W,\boldsymbol{b}}$, such as, neural networks and linear models. In this setting, classifiers in $\mathcal{F}^{(k)}_{W,\boldsymbol{b}}$ can be constructed by restricting to W and $\boldsymbol{b}$ such that $W_{:,k}=\boldsymbol{0}$, where $W_{:,k}$ is the k-th column of W, and $b_k=0$.

Remark 1 As $\mathcal{F}^{(k)}\subseteq\mathcal{F}$, the fact that that $\mathrm{VC}(\mathcal{F}^{(k)})\leq\mathrm{VC}(\mathcal{F})$, trivially follows.

In order to formally prove that by adding a feature in the greedy step the obtained classifier cannnot be less expressive (in terms of VC dimension) than the previous one, we introduce two maps:

- $\pi_k:\Omega\longrightarrow\bigotimes_{\substack{i=1\\i\neq k}}^{d}\Omega^i$, so that $\pi_k(\boldsymbol{x})=(x_1,\dots,x_{k-1},x_{k+1},\dots,x_d)$, which is a projection.
- $\iota_\alpha:\pi_k(\Omega)\longrightarrow\Omega$, $\alpha\in\Omega^k$, so that $\iota_\alpha(\boldsymbol{x})=(x_1,\dots,x_{k-1},\alpha,x_{k+1},\dots,x_d)$, which is injective.

Note that applying $\iota_\alpha\circ\pi_k$ to X has the effect of setting to α the k-th feature for all the examples.

Proposition 1 *X is shattered by $\mathcal{F}^{(k)}$ if and only if $\iota_\alpha(\pi_k(X))$ is shattered by $\mathcal{F}^{(k)}$.*

Proof Any classifier in $\mathcal{F}^{(k)}$ cannot rely on the k-th feature. Precisely, for each $\boldsymbol{x}_i\in X$ we can find $\delta_i\in\mathbb{R}$ so that $\boldsymbol{x}_i+\delta_i\boldsymbol{e}\in\iota_\alpha(\pi_k(X))$. Hence, it is equivalent for any function in $\mathcal{F}^{(k)}$ to shatter X and $\iota_\alpha(\pi_k(X))$. □

For any function $f^{(k)}\in\mathcal{F}^{(k)}$ and $\alpha\in\Omega^k$, we can define a classifier $g:\pi_k(\Omega)\longrightarrow Y$ such that $g(\boldsymbol{x})=f^{(k)}(\iota_\alpha(\boldsymbol{x}))$. Denoting by $\mathcal{G}$ the class consisting of such functions g, we achieve the following result.

Proposition 2 *$\iota_\alpha(\pi_k(X))$ is shattered by $\mathcal{F}^{(k)}$ if and only if $\pi_k(X)$ is shattered by $\mathcal{G}$.*

Proof Assume that there exists $f^{(k)}\in\mathcal{F}^{(k)}$ that shatters $\iota_\alpha(\pi_k(X))$. Note that the shattering does not rely on the k-th feature, which is constant, and therefore this is equivalent to shatter $\pi_k(\iota_\alpha(\pi_k(X)))=\pi_k(X)$ in a lower-dimensional space by means of a classifier g so that $f^{(k)}=g\circ\pi_k$. Finally, by defining $\boldsymbol{x}^{(k)}=\pi_k(\boldsymbol{x})$, $\boldsymbol{x}\in\iota_\alpha(\pi_k(X))$, we further obtain $\boldsymbol{x}=\iota_\alpha(\boldsymbol{x}^{(k)})$, and therefore $g(\boldsymbol{x}^{(k)})=f^{(k)}(\iota_\alpha(\boldsymbol{x}^{(k)}))$ for $\boldsymbol{x}^{(k)}\in\pi_k(X)$, which completes the proof. □

Corollary 1 *We have that* $\mathrm{VC}(\mathcal{G})\leq\mathrm{VC}(\mathcal{F})$.

Proof By putting together Propositions 1 and 2 we can affirm that X is shattered by $\mathcal{F}^{(k)}$ if and only if $\pi_k(X)$ is shattered by $\mathcal{G}$. Note that X and $\pi_k(X)$ have the same cardinality, and therefore $\mathrm{VC}(\mathcal{G})=\mathrm{VC}(\mathcal{F}^{(k)})$. We conclude the proof by virtue of Remark 1. □

The results in Corollary 1 formalizes the idea that by adding a feature in the greedy step the obtained classifier cannot be less expressive than the previous one. Nevertheless, in this greedy context we consider a sort of trade-off that deals with the VC dimension: precisely, a high VC-dimension allows the model to fit more complex patterns but may lead to overfitting. Hence, we will discuss later a robust stopping criteria for the greedy iterative rule. Now, as a particular case study, we consider SVM classifiers, which are probably the most frequently used ones. Further, being they based on kernels, other capability measures concerning such classifiers can be straightforwardly studied.

2.2 SVM in the greedy framework

Following the SVM literature, we drive our attention towards (strictly) positive definite kernels $\kappa : \Omega \times \Omega \longrightarrow \mathbb{R}$ that satisfy

$$\int_{\Omega} \kappa(\boldsymbol{x}, \boldsymbol{z}) v(\boldsymbol{x}) v(\boldsymbol{z}) d\boldsymbol{x} d\boldsymbol{z} \geq 0, \quad \forall v \in L_2(\Omega),$$

for $\boldsymbol{x}, \boldsymbol{z} \in \Omega$. Then, those kernels can be decomposed via the Mercer's Theorem as (see e.g. Theorem 2.2. [37] p. 107 or [38]):

$$\kappa(\boldsymbol{x}, \boldsymbol{z}) = \sum_{k \geq 0} \lambda_k \rho_k(\boldsymbol{x}) \rho_k(\boldsymbol{z}), \quad \boldsymbol{x}, \boldsymbol{z} \in \Omega,$$

where $\{\lambda_k\}_{k \geq 0}$ are the (non-negative) eigenvalues and $\{\rho_k\}_{k \geq 0}$ are the (L_2-orthonormal) eigenfunctions of the operator $T : L_2(\Omega) \longrightarrow L_2(\Omega)$, given by

$$T[v](\boldsymbol{x}) = \int_{\Omega} \kappa(\boldsymbol{x}, \boldsymbol{z}) v(\boldsymbol{z}) d\boldsymbol{z}.$$

Mercer's theorem provides an easy background for introducing feature maps and spaces. Indeed, for Mercer kernels we can interpret the series representation in terms of an inner product in the so-called *feature space* F, which is a Hilbert space. Indeed, we have that

$$\kappa(\boldsymbol{x}, \boldsymbol{z}) = \langle \Phi(\boldsymbol{x}), \Phi(\boldsymbol{z}) \rangle_F, \quad \boldsymbol{x}, \boldsymbol{z} \in \Omega, \tag{4}$$

where $\Phi : \Omega \longrightarrow F$ is a *feature map*. For a given kernel, the feature map and space are not unique. A possible solution is the one of taking the map $\Phi(\boldsymbol{x}) = \kappa(\cdot, \boldsymbol{x})$, which is linked to the characterization of F as a reproducing kernel Hilbert space; see [18, 39] for further details. Both in machine learning literature and in approximation theory, radial kernels are truly common. They are kernels for whom there exists a Radial Basis Function (RBF) $\varphi : \mathbb{R}_+ \longrightarrow \mathbb{R}$, where $\mathbb{R}_+ = [0, \infty)$, and (possibly) a shape parameter $\gamma > 0$ such that, for all $\boldsymbol{x}, \boldsymbol{z} \in \Omega$,

$$\kappa(\boldsymbol{x}, \boldsymbol{z}) = \kappa_\gamma(\boldsymbol{x}, \boldsymbol{z}) = \varphi_\gamma(||\boldsymbol{x} - \boldsymbol{z}||_2) = \varphi(r),$$

where $r = ||\boldsymbol{x} - \boldsymbol{z}||_2$. Among all radial kernels, we remark that the Gaussian one is given by

$$\kappa(\boldsymbol{x}, \boldsymbol{z}) = \kappa_\gamma(\boldsymbol{x}, \boldsymbol{z}) = \mathrm{e}^{-\gamma \|\boldsymbol{x} - \boldsymbol{z}\|_2^2} = \mathrm{e}^{-\gamma r^2}. \tag{5}$$

In the following, for simplicity, we omit the dependence on γ, which is also known as scale parameter in machine learning literature.

With radial kernel as well, SVMs can be used for classification purposes and several complexity indicators, such as the kernel alignment, can be studied in order to have a better understanding of the greedy strategy based on SVM, i.e., when the generic classifier in Equation (2) is an SVM function. The notion of kernel alignment was first introduced by [40] and later investigated in e.g. [41]. Other common complexity indicators related to the alignment can be found in [42]. Given two kernels κ_1 and $\kappa_2 : \Omega \times \Omega \longrightarrow \mathbb{R}^d$, the empirical alignment evaluates the similarity between the corresponding kernel matrices. It is given by

$$\mathsf{A}(X, \mathsf{K}_1, \mathsf{K}_2) = \frac{(\mathsf{K}_1, \mathsf{K}_2)_{\mathsf{F}}}{\sqrt{||\mathsf{K}_1||_{\mathsf{F}} ||\mathsf{K}_2||_{\mathsf{F}}}},$$

where $\mathsf{K}_1 := \mathsf{K}_1(X)$ and $\mathsf{K}_2 := \mathsf{K}_2(X)$ denote the Gram matrices for the kernels κ_1 and κ_2 on X, respectively and

$$(\mathsf{K}_1, \mathsf{K}_2)_{\mathsf{F}} = \sum_{i,j=1}^{n} \kappa_1(\boldsymbol{x}_i, \boldsymbol{x}_j) \kappa_2(\boldsymbol{x}_i, \boldsymbol{x}_j).$$

The alignment can be seen as a similarity score based on the cosine of the angle. For arbitrary matrices, this score ranges between -1 and 1.

For classification purposes we can define an ideal target matrix as $\mathsf{Y} = \boldsymbol{y}\boldsymbol{y}^\intercal$, where $\boldsymbol{y} = (y_1, \ldots, y_n)^\intercal$ is the vector of labels. Then the empirical alignment between the kernel matrix K

and the target matrix Y can be written as:

$$\mathsf{A}(X,\mathsf{K},\mathsf{Y}) = \frac{(\mathsf{K},\mathsf{Y})_{\mathsf{F}}}{\sqrt{||\mathsf{K}||_{\mathsf{F}}||\mathsf{Y}||_{\mathsf{F}}}} = \frac{(\mathsf{K},\mathsf{Y})_{\mathsf{F}}}{n\sqrt{||\mathsf{K}||_{\mathsf{F}}}}.$$

Such alignment with the target matrix is an indicator of the classification accuracy of a classifier. Indeed, to higher alignment scores correspond a separation of the data with a low bound on the generalization error [41].

We now prove the following result which will be helpful in understanding our greedy approach.

Theorem 3 *Given two kernels κ_1 and $\kappa_2 : \Omega \times \Omega \longrightarrow \mathbb{R}^d$, if $||\mathsf{K}_2||_{\mathsf{F}} \geq ||\mathsf{K}_1||_{\mathsf{F}}$ then $\mathsf{A}(X,\mathsf{K}_1,\mathsf{Y}) \leq \mathsf{A}(X,\mathsf{K}_2,\mathsf{Y})$.*

Proof By hypothesis we have that:

$$\mathsf{A}(X,\mathsf{K}_1,\mathsf{Y}) = \frac{(\mathsf{K}_1,\mathsf{Y})_{\mathsf{F}}}{n\sqrt{||\mathsf{K}_1||_{\mathsf{F}}}} \leq \frac{(\mathsf{K}1,\mathsf{Y})_{\mathsf{F}}}{n\sqrt{||\mathsf{K}_2||_{\mathsf{F}}}}.$$

Then, by adding and subtracting $(\mathsf{K}_2,\mathsf{Y})_{\mathsf{F}}$ at the numerator, and thanks to the linearity of the norm, we obtain:

$$\begin{aligned}\mathsf{A}(X,\mathsf{K}_1,\mathsf{Y}) &\leq \frac{(\mathsf{K}1,\mathsf{Y})_{\mathsf{F}}}{n\sqrt{||\mathsf{K}_2||_{\mathsf{F}}}} \\ &= \frac{(\mathsf{K}1-\mathsf{K}_2,\mathsf{Y}-\mathsf{Y})_{\mathsf{F}}}{n\sqrt{||\mathsf{K}_2||_{\mathsf{F}}}} + \frac{(\mathsf{K}_2,\mathsf{Y})_{\mathsf{F}}}{n\sqrt{||\mathsf{K}_2||_{\mathsf{F}}}} \\ &= \mathsf{A}(X,\mathsf{K}_2,\mathsf{Y}).\end{aligned}$$

□

Considering again Equation (2), let us denote by $X^{(k-1)}$ the dataset at the $(k-1)$ greedy step which already contains $k-1$ features and by $X^{(k)}$ the one that is constructed at the k-th step accordingly to our greedy rule. Then, as a corollary of the previous theorem, we have the following result.

Corollary 2 *If κ is a non-increasing radial kernel, then*

$$\mathsf{A}(X^{(k)},\mathsf{K}(X^{(k)}),\mathsf{Y}) \geq \mathsf{A}(X^{(k-1)},\mathsf{K}(X^{(k-1)}),\mathsf{Y}).$$

Proof Being $\varphi : \mathbb{R}_+ \longrightarrow \mathbb{R}$ non-increasing, for

$$\boldsymbol{x},\boldsymbol{z} \in \mathbb{R}^d,$$

we obtain

$$\begin{aligned}&\varphi\left(\|\boldsymbol{x}-\boldsymbol{z}\|_2\right) = \\ &= \varphi(\|(x_1,x_2,\ldots,x_k)-(z_1,z_2,\ldots,z_k)\|_2) \leq \\ &\leq \varphi(\|(x_1,x_2,\ldots,x_{k-1})-(z_1,z_2,\ldots,z_{k-1})\|_2),\end{aligned}$$

which in particular implies that

$$\mathsf{K}_{ij}(X^{(k-1)}) \geq \mathsf{K}_{ij}(X^{(k)}) \geq 0, \quad i,j = 1,\ldots,n.$$

Thus, we get

$$\|\mathsf{K}(X^{(k-1)})\|_{\mathsf{F}} \geq \|\mathsf{K}(X^{(k)})\|_{\mathsf{F}},$$

and hence

$$\mathsf{A}(X^{(k)},\mathsf{K}(X^{(k)}),\mathsf{Y}) \geq \mathsf{A}(X^{(k-1)},\mathsf{K}(X^{(k-1)}),\mathsf{Y}).$$

□

Note that this kind of feature augmentation strategy via greedy schemes shows some similarities with the so-called Variably Scaled Kernels (VSKs), first introduced in [43] and recently applied in the framework of inverse problems, see e.g. [44, 45]. Indeed, both approaches are based on adding features and both are again characterized by a trade-off between the model capacity, which can be characterized by the kernel alignment, and the model accuracy. To achieve a good trade-off between these two factors we need a stopping criteria for the iterative rule shown in (2).

2.3 Stopping criterion

In actual applications, the greedy iterative algorithm should select, at first, the most relevant features, and then, if no relevant features are available, any accuracy score should saturate. Among several scores μ, a robust one is the so-called True Skill Statistic (TSS) for its characteristic of being insensitive to class imbalance [46]. Precisely, letting TN, FP, FN, TP respectively the number of true negatives, false positives, false negatives and true positives, the TSS is defined by:

$$\begin{aligned}\mathrm{TSS}(\mathrm{TN},\mathrm{FP},\mathrm{FN},\mathrm{TP}) = &\ \mathrm{recall}(\mathrm{TN},\mathrm{FP},\mathrm{FN},\mathrm{TP}) \\ &+ \mathrm{specificity}(\mathrm{TN},\mathrm{FP},\mathrm{FN},\mathrm{TP}) - 1,\end{aligned}$$

where

$$\mathrm{recall}(\mathrm{TN},\mathrm{FP},\mathrm{FN},\mathrm{TP}) = \frac{\mathrm{TP}}{\mathrm{FN}+\mathrm{TP}}, \quad (6)$$

and

$$\mathrm{specificity}(\mathrm{TN},\mathrm{FP},\mathrm{FN},\mathrm{TP}) = \frac{\mathrm{TN}}{\mathrm{FP}+\mathrm{TN}}. \quad (7)$$

In order to introduce a stopping criteria, we need to point out that we construct a greedy feature ranking by considering, at each step, q splits of the dataset into training and validation sets. Specifically, at the k-th step of the greedy algorithm, each one of the $d-k$ datasets, composed by the $k-1$ selected features $x_1, \dots, x_{k-1}$ and the added one x_p $(p = k, \dots, d)$, is divided into training and validation sets, namely $\{(X_{p,h}^{(t)}, Y_{p,h}^{(t)})\}$ and $\{(X_{p,h}^{(v)}, Y_{p,h}^{(v)})\}$ respectively, for $h = 1, \dots, q$, and fixed p. Hence, once the models $\mathcal{M}_p$ $(p = k, \dots, d)$ have been trained the k-th feature is chosen so that:

$$x_k = \arg \max_{p=k,\dots,d} \frac{1}{q} \sum_{h=1}^{q} \mu(\mathcal{M}_p(X_{p,h}^{(v)}), Y_{p,h}^{(v)}), \quad (8)$$

where μ is the TSS score. Then, letting m_k be the average of the TSS scores computed on different splits at the k-th step and σ_k the associated standard deviation, we stop the greedy iteration at the k-th step if:

$$\frac{|m_{k+1} - m_k|}{\sqrt{(\sigma_{k+1}^2 + \sigma_k^2)}} < \tau, \quad (9)$$

and τ is a given threshold. By doing so, we stop the greedy algorithm when the added feature does not contribute to the accuracy score. In order to better understand this fact, we provide in the following a numerical experiment with synthetic data. Dealing with real data, we might stop the greedy iteration as shown in (9), but then select only the first k^* features, where k^* is

$$k^* = \arg \max_{j=1,\dots,k} m_j. \quad (10)$$

3 Numerical experiments

The first numerical experiment wants to numerically show the convergence of the greedy algorithm and the efficacy of the stopping rule. Then, we will show an application in the context of space weather, which aims to show how this general method is able to infer on the physical aspects of the problem.

3.1 Applications to a toy dataset

We first focused on the application of the nonlinear SVM greedy technique to a balanced simulated dataset constructed as follows: we considered the set $X = \{\boldsymbol{x}_i\}_{i=1}^n$ of $n = 1000$ random points in dimension $d = 15$ sampled from a uniform distribution over $[0, 1)$ and the set of corresponding function values $\{f_{\alpha,i} = f_\alpha(\boldsymbol{x}_i)\}_{i=1}^n$, where $f_\alpha : [0, 1)^d \longrightarrow \mathbb{R}$ is defined as

$$f_\alpha(\boldsymbol{x}) = e^{x_1^2} + e^{x_2} + 3x_3 + 2\cos(x_4 x_5) + 4x_6^2 + 10^\alpha \sum_{j=7}^{d} x_j. \quad (11)$$

and $\alpha \in \{-8, -6, -4, -2\}$. Each $f_{\alpha,i}$ was then labeled according to a threshold value to obtain the set of outputs $Y = \{y_i\}$, i.e., $y_i = 1$ if $f_{\alpha,i}$ is greater than the mean value attained by f_α, and $y_i = -1$ otherwise. From (11) we note that the first 6 features (i.e., x_j for $j = 1, \dots, 6$) are meaningful for classification purposes when α is lower than -4, while the contribution of the remaining ones is negligible. The classifier used in the following was an SVM model for which both the scale parameter of the Gaussian kernel and the bounding box are optimized via standard cross-validation. The results of using such a classifier into the greedy scheme are reported in Table 1. Such table contains the greedy ranking of the features x_j, $j = 1, \dots, d$, and the TSS values obtained at each step by averaging over 7 different validation sets. Letting $\tau = 9\text{e}-2$ be the threshold for the stopping criteria in (9), the greedy algorithm selected the features reported in Table 1, which are above the black solid line. As expected, the algorithm selected only the first six features (the most relevant ones) when α is small enough ($\alpha \leq -6$). Then, as soon as the remaining features become more meaningful the greedy selection takes into account more features. In this didactic example we report all the TSS values until the end, to emphasise the robustness of our procedure that correctly identified the most relevant features.

Table 1 Feature ranking for the greedy scheme on the dataset generated as in (11). The selected features are identified by the bold line in the table.

$\alpha = -8$		$\alpha = -6$		$\alpha = -4$		$\alpha = -2$	
x_j	TSS	x_j	TSS	x_j	TSS	x_j	TSS
x_1	0.204 ± 0.050	x_1	0.204 ± 0.050	x_1	0.198 ± 0.048	x_1	0.197 ± 0.034
x_6	0.550 ± 0.049	x_6	0.553 ± 0.050	x_6	0.558 ± 0.048	x_6	0.553 ± 0.044
x_3	0.798 ± 0.049	x_3	0.798 ± 0.049	x_3	0.800 ± 0.051	x_3	0.787 ± 0.041
x_2	0.930 ± 0.030	x_2	0.930 ± 0.030	x_2	0.933 ± 0.031	x_2	0.888 ± 0.017
x_4	0.939 ± 0.021	x_4	0.939 ± 0.021	x_4	0.939 ± 0.024	x_4	0.895 ± 0.025
x_5	0.954 ± 0.015	x_5	0.954 ± 0.015	x_5	0.961 ± 0.017	x_{13}	0.899 ± 0.024
x_{12}	0.953 ± 0.014	x_{12}	0.953 ± 0.014	x_{12}	0.953 ± 0.014	x_8	0.888 ± 0.034
x_{13}	0.953 ± 0.022	x_{13}	0.953 ± 0.022	x_9	0.948 ± 0.022	x_5	0.895 ± 0.036
x_9	0.946 ± 0.028	x_9	0.946 ± 0.028	x_{13}	0.948 ± 0.025	x_{14}	0.900 ± 0.035
x_{11}	0.928 ± 0.039	x_{11}	0.928 ± 0.039	x_{14}	0.929 ± 0.026	x_7	0.896 ± 0.039
x_{14}	0.932 ± 0.021	x_{14}	0.932 ± 0.021	x_{11}	0.920 ± 0.027	x_{10}	0.895 ± 0.039
x_7	0.914 ± 0.023	x_7	0.914 ± 0.023	x_7	0.911 ± 0.031	x_{12}	0.881 ± 0.036
x_{10}	0.889 ± 0.024	x_{10}	0.889 ± 0.024	x_8	0.890 ± 0.043	x_{11}	0.881 ± 0.046
x_8	0.883 ± 0.025	x_8	0.873 ± 0.041	x_{10}	0.878 ± 0.027	x_9	0.871 ± 0.028

3.2 Applications to solar physics: geo-effectiveness prediction

We now focus on a significant space weather application, i.e., the prediction of severe geo-effectiveness events based on the use of both remote sensing and in-situ data. More specifically, data-driven methods addressing this task typically utilizes features acquired by in-situ instruments at Lagrangian point L1 (i.e., the Lagrangian point between the Sun and the Earth) to forecast a significant increase of the SYM-H index, i.e., the expression of the geomagnetic disturbance at Earth [47].

3.2.1 The dataset and the models

The dataset we used consisted of a collection of solar wind, geomagnetic and energetic indices. In particular, it was composed by $N = 7888320$ examples and $d = 15$ features sampled at each minute starting from (1-st January 2005) to (31-st December 2019). Below we summarize the features we used:

1. B [nT], the magnetic field intensity, and $\mathrm{B_x}$, $\mathrm{B_y}$ and $\mathrm{B_z}$ [nT], its three coordinates.
2. V [Km/s], the velocity of the solar wind, and $\mathrm{V_x}$, $\mathrm{V_y}$ and $\mathrm{V_z}$ [Km/s], its three coordinates.
3. T, the proton temperature, and ρ, the proton density number [$\mathrm{cm^{-3}}$].
4. $\mathrm{E_k}$, $\mathrm{E_m}$, $\mathrm{E_t}$ the kinetic, magnetic and total energies.
5. $\mathrm{H_m}$, the magnetic helicity.
6. SYM-H [nT], a geomagnetic activity index that quantifies the level of geomagnetic disturbance.

The first ten features were acquired at the Lagrangian point L1 by in-situ instruments, the energies and the magnetic helicity being adimensional derived quantities, and the SYM-H being measured at Earth. The task considered in what follows consisted in identifying the most relevant features used to predict whereas a geo-effective event occurred, i.e., when the SYM-H was less than -50 nT (label 1), or not (label -1). The dataset at our disposal was highly unbalanced: the rate of positive events was about 2.5%. In order exploit our data analysis, we first need to fix the notation. We denote by $\tilde{X} = \{\tilde{\boldsymbol{x}}_i\}_{i=1}^N \subseteq \Omega$, where $\Omega \subseteq \mathbb{R}^d$, the set of input samples an by $\tilde{Y} = \{\tilde{y}_i\}_{i=1}^N$, with $\tilde{y}_i \in \{-1, 1\}$, the set of associated labels. The features denoted by $\tilde{x_j}$, $j = 1, \ldots, d$, represent respectively B, $\mathrm{B_x}$, $\mathrm{B_y}$, $\mathrm{B_z}$, V, $\mathrm{V_x}$, $\mathrm{V_y}$, $\mathrm{V_z}$, T, ρ, $\mathrm{E_k}$, $\mathrm{E_m}$, $\mathrm{E_t}$, $\mathrm{H_m}$ and the SYM-H.

The analysis was performed with data aggregated by hours, i.e., letting $m = 60$, $n = N/m$

and

$$\boldsymbol{x}_i = \frac{(\sum_{k=i}^{i+m} \tilde{\boldsymbol{x}_k})}{m},$$

we focused on $X = \{\boldsymbol{x}_i\}_{i=1}^n \subseteq \Omega$. Similarly, we defined the set of aggregated labels $Y = \{y_i\}_{i=1}^n$.

Given X and Y, the first step of our study consisted in using different feature selection approaches to rank the features accordingly to their relevance (see Subsection 3.2.2). After this step, we investigated how these results can be exploited to improve the prediction task (see Subsection 3.2.3). In doing so, we used both SVM and a Feed-forward Neural Network (FNN) in order to predict whether a geo-effective event occurred or not in the next hour. Specifically, the SVM algorithm was trained by performing a randomized and cross-validated search over the hyper-parameters of the model (the regularization parameter C and the kernel coefficient γ) taken from uniform distributions on $I_C = [0.1, 1000]$ and $I_\gamma = [0.001, 0.1]$ respectively. Instead, the FNN architecture was characterized by 7 hidden layers. The Rectified Linear Unit (ReLU) function was used to activate the hidden layers, the sigmoid activation function was applied to activate the output, and the binary cross-entropy was used as loss function. The model was trained over 200 epochs using the Adam optimizer with learning rate equal to 0.001, with a mini-batch size of 64. In order to prevent overfitting, an L^2 regularization constraint was set as 0.01 in the first two layers. Further, we used an early stopping strategy to select the best epoch with respect to the validation loss.

3.2.2 Greedy feature selection approaches

In order to apply efficiently our greedy strategy to both SVM and FNN, we first considered a subset X_p of the original dataset X with a reduced number of examples: we took $p = 3333$ examples. The so-constructed ranking was compared to a state-of-the-art method, i.e., the Lasso feature selection. Precisely, the active set of features returned by Lasso was composed by: B_x, B_y, B_z, V_y, V_z, T, ρ, E_k, E_m, E_t, H_m and the SYM-H. Note that neither V and B, which are physically meaningful for the considered task, were selected by cross-validated Lasso.

In Table 2 we report the results of the greedy feature ranking scheme by using SVM and FNN. In this table, the features are ordered accordingly to the greedy selection. In particular, the greedy iteration stopped with all the features reported in the table accordingly to (9), but the selected features were only the ones above the bold line, as in (10). We can note that, the features selected for both SVM and FNN are only a few, and this is due to the fact that greedy schemes are model-dependent and hence are able to truly capture the most significant ones. We further point out that in order to extract such features, we made use of a validation set and we did not considered any test set, since it was not at our disposal. Therefore, the greedy feature extraction is coherently based on the TSS computed on the validation set, and not on the test set. Nevertheless, we are now interested in understanding how the selected features work in the prediction (on tests sets) of the original task and with all examples.

Table 2 Feature rankings for the greedy schemes on the dataset used for the prediction of geo-effective solar storms.

Greedy ranking (SVM)		Greedy ranking (FNN)	
x_j	TSS	x_j	TSS
SYM-H	0.703 ± 0.179	SYM-H	0.936 ± 0.052
B_z	0.823 ± 0.121	B	0.943 ± 0.034
V	0.804 ± 0.115	E_t	0.958 ± 0.039
E_t	0.825 ± 0.176	V_x	0.934 ± 0.078
V_x	0.853 ± 0.147		
E_m	0.804 ± 0.184		
B	0.835 ± 0.115		

Interestingly, the features extracted as the most prominent ones are indeed those associated with physical processes involved in the transfer of energy from the CMEs to the Earth's magnetosphere and, thus, with the CME likelihood for inducing geomagnetic storms. B_z, i.e., a southward directed interplanetary magnetic field, is indeed required for magnetic reconnection with the Earth's magnetic field to occur, and thus for the energy carried by the solar wind and/or CMEs to be transferred to the Earth system. In addition, the bulk speed V, or equivalently the radial component of the flow velocity vector V_x, is directly related to the kinetic energy

of the solar wind. On the one hand, it is well known that particularly fast particle streams or solar transients can compress the magnetosphere on the sunward side. On the other hand, high levels of magnetic energy (quadratically proportional to the magnetic field intensity) can be converted into thermal energy that heats the Earth's atmosphere, expanding it. In both cases, it appears evident that the transfer of energy, either kinetic or magnetic or total, enabled by the magnetic reconnection between the interplanetary and terrestrial magnetic fields, disrupts the magnetosphere current system, thus causing geomagnetic disturbances. As a conclusion, the extracted features are the physical quantities with the higher expected predictive capability.

3.2.3 Prediction of geo-effective solar events with greedy-selected features

In order to numerically validate our greedy procedure we compared the performances of SVM and FNN trained with respectively: all features, the features returned by Lasso, and the greedily selected features. The comparison was performed by computing several scores (reported in Tables 3 and 4) and by averaging on different splits of the test set: in particular, we computed the TSS as reference score, the Heidke Skill Score (HSS) [48], precision, recall (see equation (6)), specificity (see equation (7)), F1 score (which is the harmonic mean of precision and recall), and balanced accuracy (which is the arithmetic mean between recall and specificity). We can observe that for the SVM-based prediction, when using the features extracted with the greedy procedure, we have a remarkable improvement of all accuracy scores. Further, although the performances of the FNN are essentially the same, independently of the feature selection scheme, we note that we were able to achieve the same accuracy scores with only a few features selected ad hoc (3 in this case). This points out again the fact that features extracted by methods, such as Lasso, might be redundant for the considered classifiers. This is even more evident when using the FNN algorithm, which achieved the same accuracy with only 3 greedily selected features. The improvement in terms of accuracy was remarkable only for SVM classifiers, which is known to be less robust then neural networks to *noise*, i.e., redundant information stored in redundant features.

4 Conclusions and future work

We introduced a novel class of feature reduction schemes, namely greedy feature selection algorithms. Their main advantage consists in the fact that they are able to identify the most relevant features for any given classifier. We studied their behaviour both analytically and numerically. Analytically, we could conclude that the models constructed in such a way cannot be less expressive than the standard ones (in terms of VC dimension or kernel alignment). Numerically, we showed their efficacy on a problem associated to the prediction of geo-effective events of the active Sun. As the activity of the Sun is cyclic, work in progress consists in using greedy schemes to study which features are relevant on either high or low activity periods. Finally, as there is a growing interest in physics-informed neural networks (PINN), we should investigate, both theoretically and numerically, which are the challenges that greedy methods could achieve in this context.

Acknowledgements

Fabiana Camattari and Emma Perracchione kindly acknowledge the support of the Fondazione Compagnia di San Paolo within the framework of the Artificial Intelligence Call for Proposals, AIxtreme project (ID Rol: 71708). Sabrina Guastavino was supported by the Programma Operativo Nazionale (PON) "Ricerca e Innovazione" 2014–2020. The research by Michele Piana was supported in part by the MIUR Excellence Department Project awarded to Dipartimento di Matematica, Università di Genova, CUP D33C23001110001. All authors are members of the Gruppo Nazionale per il Calcolo Scientifico - Istituto Nazionale di Alta Matematica (GNCS - INdAM).

References

[1] Dutta, S., Farthing, M.W., Perracchione, E., Savant, G., Putti, M.: A greedy non-intrusive reduced order model for shallow

Table 3 Average scores obtained with SVM using different subsets of features.

Metric	All features	LASSO selection	Greedy selection (SVM)
TSS	0.679 ± 0.055	0.677 ± 0.088	0.736 ± 0.051
HSS	0.731 ± 0.043	0.739 ± 0.040	0.808 ± 0.021
Precision	0.822 ± 0.117	0.840 ± 0.068	0.909 ± 0.043
Recall	0.683 ± 0.059	0.681 ± 0.090	0.738 ± 0.052
Specificity	0.995 ± 0.005	0.996 ± 0.002	0.998 ± 0.001
F1 score	0.737 ± 0.041	0.745 ± 0.039	0.812 ± 0.021
Balanced Accuracy	0.839 ± 0.027	0.839 ± 0.044	0.868 ± 0.026

Table 4 Average scores obtained with FNN using different subsets of features.

Metric	All features	LASSO selection	Greedy selection (SVM)
TSS	0.913 ± 0.054	0.917 ± 0.043	0.895 ± 0.054
HSS	0.685 ± 0.105	0.638 ± 0.119	0.669 ± 0.128
Precision	0.577 ± 0.153	0.519 ± 0.159	0.571 ± 0.176
Recall	0.935 ± 0.065	0.945 ± 0.056	0.919 ± 0.068
Specificity	0.978 ± 0.014	0.972 ± 0.017	0.976 ± 0.019
F1 score	0.695 ± 0.010	0.650 ± 0.114	0.680 ± 0.122
Balanced Accuracy	0.957 ± 0.027	0.959 ± 0.022	0.948 ± 0.027

water equations. J. Comput. Phys. **439**, 110378 (2021)

[2] De Marchi, S., Schaback, R., Wendland, H.: Near-optimal data-independent point locations for radial basis function interpolation. Adv. Comput. Math. **23**, 317–330 (2005)

[3] Santin, G., Haasdonk, B.: Convergence rate of the data-independent P-greedy algorithm in kernel-based approximation. Dolomites Res. Notes Approx. **10**(2), 68–78 (2017)

[4] Wenzel, T., Santin, G., Haasdonk, B.: A novel class of stabilized greedy kernel approximation algorithms: Convergence, stability and uniform point distribution. J. Approx. Theory **262**, 105508 (2021)

[5] Wenzel, T., Santin, G., Haasdonk, B.: Analysis of target data-dependent greedy kernel algorithms: Convergence rates for f-, f·P- and f/P-greedy. Constructive Approx. **57**(1), 45–74 (2023)

[6] Wirtz, D., Haasdonk, B.: A Vectorial Kernel Orthogonal Greedy Algorithm. Dolomites Res. Notes Approx. **6**, 83–100 (2013)

[7] Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley-interscience, New York (2012)

[8] Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans. Pattern Anal. Mach. Intell., 1226–1238 (2005)

[9] Robnik-Šikonja, M., Kononenko, I.: Theoretical and empirical analysis of ReliefF and RReliefF. Machine Learn. **53**(1-2), 23–69 (2003)

[10] Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc., Ser. B, Methodol. **50**(1), 267–288 (1996)

[11] Yuan, M., Lin, Y.: Model selection and estimation in regression with grouped variables. J. R. Stat. Soc. Series B: Stat. Meth. **68**(1),

49–67 (2006)

[12] Zou, H.: The adaptive lasso and its oracle properties. J. American Stat. Association **101**(476), 1418–1429 (2006)

[13] Guastavino, S., Benvenuto, F.: A consistent and numerically efficient variable selection method for sparse Poisson regression with applications to learning and signal recovery. Stat. Comput. **29**(3), 501–516 (2019)

[14] Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Machine Learn. **46**, 389–422 (2002)

[15] Freijeiro-González, L., Febrero-Bande, M., González-Manteiga, W.: A critical review of lasso and its derivatives for variable selection under dependence among covariates. Internat. Stat. Rev. **90**(1), 118–145 (2022)

[16] Vapnik, V.N., Chervonenkis, A.Y.: On the uniform convergence of relative frequencies of events to their probabilities. Theory Probab. Appl. **16**, 264–280 (1971)

[17] Bartlett, P.L., Mendelson, S.: Rademacher and gaussian complexities: risk bounds and structural results. J. Machine Learn. Res. **3**, 463–482 (2002)

[18] Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press, Cambridge (2004)

[19] Piana, M., Emslie, A.G., Massone, A.M., Dennis, B.R.: Hard X-ray Imaging of Solar Flares vol. 164. Springer, Berlin (2022)

[20] Schwenn, R.: Space weather: The solar perspective. Living reviews in solar physics **3**(1), 1–72 (2006)

[21] Kahler, S.: Solar flares and coronal mass ejections. Annual review of astronomy and astrophysics **30**(1), 113–141 (1992)

[22] Gonzalez, W., Joselyn, J.-A., Kamide, Y., Kroehl, H.W., Rostoker, G., Tsurutani, B.T., Vasyliunas, V.: What is a geomagnetic storm? Journal of Geophysical Research: Space Physics **99**(A4), 5771–5792 (1994)

[23] Bobra, M.G., Couvidat, S.: Solar flare prediction using sdo/hmi vector magnetic field data with a machine-learning algorithm. The Astrophysical Journal **798**(2), 135 (2015)

[24] Camporeale, E., Wing, S., Johnson, J.: Machine Learning Techniques for Space Weather. Elsevier, United States (2018)

[25] Florios, K., Kontogiannis, I., Park, S.-H., Guerra, J.A., Benvenuto, F., Bloomfield, D.S., Georgoulis, M.K.: Forecasting solar flares using magnetogram-based predictors and machine learning. Solar Physics **293**(2), 28 (2018)

[26] Guastavino, S., Candiani, V., Bemporad, A., Marchetti, F., Benvenuto, F., Massone, A.M., Mancuso, S., Susino, R., Telloni, D., Fineschi, S., Piana, M.: Physics-driven Machine Learning for the Prediction of Coronal Mass Ejections' Travel Times. The Astrophys. J. **954**(2), 151 (2023)

[27] Telloni, D., Lo Schiavo, M., Magli, E., Fineschi, S., Guastavino, S., Nicolini, G., Susino, R., Giordano, S., Amadori, F., Candiani, V., *et al.*: Prediction capability of geomagnetic events from solar wind data using neural networks. The Astrophys. J. **952**(2), 111 (2023)

[28] Campi, C., Benvenuto, F., Massone, A.M., Bloomfield, D.S., Georgoulis, M.K., Piana, M.: Feature ranking of active region source properties in solar flare forecasting and the uncompromised stochasticity of flare occurrence. The Astrophysical Journal **883**(2), 150 (2019)

[29] Bommert, A., Sun, X., Bischl, B., Rahnenführer, J., Lang, M.: Benchmark for filter methods for feature selection in high-dimensional classification data. Comput. Stat. & Data Anal. **143**, 106839 (2020)

[30] Bajer, D., Dudjak, M., Zorić, B.: Wrapper-based feature selection: how important is the wrapped classifier? In: 2020 International

Conference on Smart Systems and Technologies (SST), pp. 97–105 (2020). IEEE

[31] Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D., Saeed, J.: A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. J. Appl. Sci. and Tech. Trends **1**(2), 56–70 (2020)

[32] James, G., Witten, D., Hastie, T., Tibshirani, R., Taylor, J.: An Introduction to Statistical Learning with Applications in Python, pp. 233–235. Springer, Cham (2023)

[33] Wenzel, T., Marchetti, F., Perracchione, E.: Data-driven kernel designs for optimized greedy schemes: A machine learning perspective. SIAM J. Sci Comput. **46**(1), 101–126 (2024)

[34] Temlyakov, V.N.: Greedy approximation. Acta Numer. **17**, 235–409 (2008)

[35] Schölkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge, MA, USA (2002)

[36] Vapnik, V.N.: Statistical Learning Theory. Wiley, NY, USA (1998)

[37] Fasshauer, G.E.: Meshfree Approximations Methods with MATLAB. World scientific, Singapore (2007)

[38] Mercer, J.: Functions of positive and negative type and their connection with the theory of integral equations. Phil. Trans. Royal Society **209**, 415–446 (1909)

[39] Fasshauer, G.E., McCourt, M.: Kernel-based Approximation Methods Using MATLAB. World scientific, Singapore (2015)

[40] Cristianini, N., Shawe-Taylor, J., Elisseeff, A., Kandola, J.: On kernel-target alignment. In: Dietterich, T., Becker, S., Ghahramani, Z. (eds.) Advances in Neural Information Processing Systems, vol. 14. MIT Press, Cambridge (2001)

[41] Wang, T., Dongyan, Z., Shengfeng, T.: An overview of kernel alignment and its applications. Artificial Intell. Rev. **43**(2), 179–192 (2015)

[42] Donini, M., Aiolli, F.: Learning deep kernels in the space of dot product polynomials. Machine Learn. **106**, 1245–1269 (2017)

[43] Bozzini, M., Lenarduzzi, L., Rossini, M., Schaback, R.: Interpolation with variably scaled kernels. IMA J. Numer. Anal. **35**, 199–219 (2015)

[44] Perracchione, E., Camattari, F., Volpara, A., Massa, P., Massone, A.M., Piana, M.: Unbiased clean for stix in solar orbiter. The Astrophys. J. Suppl. Series **268**(2), 68 (2023)

[45] Perracchione, E., Massone, A.M., Piana, M.: Feature augmentation for the inversion of the Fourier transform with limited data. Inverse Probl. (2021)

[46] Bloomfield, D.S., Higgins, P.A., McAteer, R.T.J., Gallagher, P.T.: Toward reliable benchmarking of solar flare forecasting methods. The Astrophys. J. Letters **747**(2), 41 (2012)

[47] Wanliss, J.A., Showalter, K.M.: High-resolution global storm index: Dst versus sym-h. Journal of Geophysical Research: Space Physics **111**(A2) (2006)

[48] Heidke, P.: Berechnung des erfolges und der gute der windstarkevorhersagen im sturmwarnungsdienst. Geogr. Ann. **8**, 301–349 (1926)