POLITECNICO DI TORINO
Repository ISTITUZIONALE

Binary Adaptive Embeddings from Order Statistics of Random Projections

Original
Binary Adaptive Embeddings from Order Statistics of Random Projections / Valsesia, Diego; Magli, Enrico. - In: IEEE SIGNAL PROCESSING LETTERS. - ISSN 1070-9908. - 24:1(2017), pp. 111-115. [10.1109/LSP.2016.2639036]

Availability:
This version is available at: 11583/2671706 since: 2017-05-23T11:07:07Z

Publisher:
Institute of Electrical and Electronics Engineers Inc.

Published
DOI:10.1109/LSP.2016.2639036

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

12 October 2024

Binary adaptive embeddings from order statistics of random projections

Diego Valsesia, and Enrico Magli

***Abstract*—We use some of the largest order statistics of the random projections of a reference signal to construct a binary embedding that is adapted to signals correlated with such signal. The embedding is characterized from the analytical standpoint and shown to provide improved performance on tasks such as classification in a reduced-dimensionality space.**

***Keywords*—Binary Embeddings, Random projections**

I. Introduction

The ever-increasing amount of information that is produced in the age of Big Data calls for efficient techniques for storage and processing of a large number of high-dimensional signals. Compact representations can be obtained with different methods depending whether the particular task requires signal reconstruction (e.g., image and video compression for delivery and visualization) or the goal is to infer some information from the signals (e.g., in classification, regression, information retrieval, etc.). Embeddings provide compact representations of signals for the latter tasks. Formally, an embedding is a transformation that maps a set of signals in a high dimensional space to a lower dimensional space, in such a way that the geometry of the set is approximately preserved. The concept of embedding has been successfully used in the context of information retrieval [1], where it is usually called "hashing". An important class of signal embeddings are those preserving the distances among pairs of signals. Johnson and Lindenstrauss [2] famously stated that an embedding can be realized with a Lipschitz mapping to approximately preserve Euclidean distances with a dimension of the embedding space that only depends on the desired distortion and logarithmically in the number of signals to be embedded. Random projections have been shown to implement such embedding with high probability. Several extensions have later been proposed, allowing one to approximately preserve the angle between signals [3], [4], control the maximum distance that is embedded [5], or preserve the Jaccard distance [6], [7]. Recently, some works have studied learning embeddings [8]–[10] from training data to derive compact codes by exploiting the particular geometry of the dataset (e.g., signals living close to a manifold).

In this paper we propose an approach to construct an embedding that is not based on learning and does not require a training set of data, but rather is adapted to a single reference signal. This choice maintains to some degree the universality of random projections and it is useful when the data present no particular structure. Jegou et al. [11] empirically explored a similar idea by proposing to choose hash functions with a robustness criterion, that essentially measures how far a random projection falls from the edges of a quantization interval. Our work presents a rigorous analytical treatment of a binary embedding obtained from the selection of the random projections of a reference signal with largest magnitude. The analysis of the embedding provides insights on its advantages, particularly in mitigating the difficulty of low-contrast nearest neighbor problems and superior performance on classification tasks, e.g. in a neural network.

The authors are with Politecnico di Torino, Torino, Italy.This work was supported by the European Research Council through the European Community Seventh Framework Programme (FP7/2007-2013) under Grant 279848.

II. Proposed method

A. Preliminaries

Definition 1. *A mapping $\phi : \mathcal{X} \rightarrow \mathcal{Y}$ of metric spaces, endowed with distances $d_{\mathcal{X}}$ and $d_{\mathcal{Y}}$ is called an embedding with distortion $C > 0$ if $Ld_{\mathcal{X}}(\mathbf{u},\mathbf{v}) \leq d_{\mathcal{Y}}(\phi(\mathbf{u}),\phi(\mathbf{v})) \leq CLd_{\mathcal{X}}(\mathbf{u},\mathbf{v})$ for some constant $L > 0$ and for all $\mathbf{u},\mathbf{v} \in \mathcal{X}$.*

A well-known binary embedding is the sign random projections [3] where a random matrix $\mathbf{\Phi}$ made of independent and identically distributed (i.i.d.) Gaussian entries is used to compute some random projections, which are then quantized to a binary representation by keeping their sign. The Hamming distance between the binary vectors approximately preserves the angle between the signals in the original space [3], i.e., $\mathbb{P}\left(\text{sign}(\mathbf{\Phi}_i\mathbf{u}) = \text{sign}(\mathbf{\Phi}_i\mathbf{v})\right) = 1 - \frac{\theta}{\pi}$, being $\theta = \cos^{-1}\left(\frac{\mathbf{u}^T\mathbf{v}}{\|\mathbf{u}\|\|\mathbf{v}\|}\right)$ and $\mathbf{\Phi}_i$ the i-th row of $\mathbf{\Phi}$.

B. Proposed adaptive embedding

A reference signal $\mathbf{u}$ is used to generate the adaptive embedding in the following way. A number m_{pool} of random projections is computed by means of an i.i.d. Gaussian matrix $\mathbf{\Phi} \in \mathbb{R}^{m_{\text{pool}} \times n}$ as $\mathbf{y} = \mathbf{\Phi}\mathbf{u}$. The m entries with largest magnitude are identified and their locations stored in vector $\mathbf{l}$. The m-bit resulting binary code is:

$$\mathbf{p} = \text{sign}(\mathbf{\Phi}_{\mathbf{l}}\mathbf{u}) \ , \tag{1}$$

where $\mathbf{\Phi}_{\mathbf{l}}$ is the matrix $\mathbf{\Phi}$ restricted to the rows indexed by $\mathbf{l}$. The locations vector $\mathbf{l}$ is saved as side information of the embedding and used as in (1) whenever a new signal is to be embedded, i.e., $\mathbf{q} = \text{sign}(\Phi_{\mathbf{l}}\mathbf{v})$, where $\mathbf{v}$ is a generic signal and $\mathbf{q}$ its embedding.

The first theorem we prove confirms that the Hamming distance $d_H(\mathbf{p},\mathbf{q}) = \frac{1}{m}\sum_i p_i \oplus q_i$, i.e. the number of differing entries, between binary codes obtained with the adaptive embedding concentrates around its expected value.

Theorem 2. *Let $\mathcal{X} \subset \mathbb{R}^n$ be a set of N signals and $\mathbf{u},\mathbf{v} \in \mathcal{X}$. Let $\mathbf{\Phi} \in \mathbb{R}^{m_{pool} \times n}$ with $\mathbf{\Phi}_{i,j} \sim \mathcal{N}(0,\sigma^2)$, $\mathbf{y} = \mathbf{\Phi}\mathbf{u}$ and the locations $\mathbf{l}$ of the $m \leq m_{pool}$ entries in $\mathbf{y}$ with largest*

magnitude be known. Let $\mathbf{p} = \text{sign}(\mathbf{\Phi_l u})$ *and* $\mathbf{q} = \text{sign}(\mathbf{\Phi_l v})$. *Then for* $0 < \varepsilon \leq 2e - 1$,

$$\mathbb{P}\left(|d_H(\mathbf{p},\mathbf{q}) - \mu| > \epsilon\mu\right) < e^{-\mu\frac{\varepsilon^2}{2}} + e^{-\mu\frac{\varepsilon^2}{4}} \tag{2}$$

with

$$\mu = \mathbb{E}\left[d_H(\mathbf{p},\mathbf{q})\right] = \frac{1}{m}\sum_{i=1}^{m} p_i$$

$$p_i = \frac{1}{2} + \frac{1}{2}\text{erf}\left(-y_{l_i}\frac{\mathbf{u}^T\mathbf{v}}{\sqrt{2}\sigma\|\mathbf{u}\|\sqrt{\|\mathbf{u}\|^2\|\mathbf{v}\|^2 - (\mathbf{u}^T\mathbf{v})^2}}\right)$$

Proof: Let us consider a single measurement in the l_i location $y_i = \mathbf{\Phi}_{l_i}\mathbf{u}$ and $z_i = \mathbf{\Phi}_{l_i}\mathbf{v}$. Then $\zeta = [y_i, z_i]$ is a bivariate Gaussian with zero mean and covariance $\Sigma = \sigma^2\begin{bmatrix}\|\mathbf{u}\|^2 & \mathbf{u}^T\mathbf{v}\\ \mathbf{u}^T\mathbf{v} & \|\mathbf{v}\|^2.\end{bmatrix}$. Suppose that y_i is observed to be $y_i = \tau_i$, then the conditional distribution of z_i given $y_i = \tau_i$ is $\eta_{\tau_i} = (z_i|y_i = \tau_i) \sim \mathcal{N}\left(\tau_i\frac{\mathbf{u}^T\mathbf{v}}{\|\mathbf{u}\|^2}, \sigma^2\left(\|\mathbf{v}\|^2 - \frac{(\mathbf{u}^T\mathbf{v})^2}{\|\mathbf{u}\|^2}\right)\right)$.

After quantization of the measurements, the probability of mismatching bits in position l_i is

$$\begin{aligned} p_i &= \mathbb{P}\left(\eta_\tau \leq 0|\tau_i > 0\right)\\ &= \frac{1}{2} + \frac{1}{2}\text{erf}\left(-\tau_i\frac{\mathbf{u}^T\mathbf{v}}{\sqrt{2}\sigma\|\mathbf{u}\|\sqrt{\|\mathbf{u}\|^2\|\mathbf{v}\|^2 - (\mathbf{u}^T\mathbf{v})^2}}\right)\end{aligned}$$

Define the following random variable

$$E_i = \begin{cases} 0 & \text{with probability } 1 - p_i\\ 1 & \text{with probability } p_i\end{cases}. \tag{3}$$

Then, $D_H = \frac{1}{m}\sum_{i=1}^{m} E_i$ is a Poisson Binomial random variable measuring the Hamming distance between $\mathbf{p}$ and $\mathbf{q}$. Eq. (2) can be readily obtained using Chernoff bounds [12] for the tails of D_H. ■

The previous theorem holds for a fixed pair of signals. It is customary to derive an asymptotic result on the number of measurements needed to provide a distortion δ around the expectation when signals are drawn from a finite set of cardinality N. Standard derivation using a union bound on Eq. (2) yields $m = \mathcal{O}(\delta^{-2}\log N)$, which is exactly the same as classic results on non-adaptive random projections [2]. The advantages of the proposed method are, in fact, due to the modified expected value rather than the variance.

Moreover, the previous theorem supposed we knew the values of the projections of the reference signal at the locations kept as side information. This allows us to compute the exact expected value of the Hamming distance in the embedded space as function of the inner product (or correlation coefficient) between the original signals. However, it might be useful to have some a-priori knowledge about the embedding without the need to know the reference signal. The following theorem approximately bounds the expected value of the embedding by characterizing the order statistics of $|\mathbf{y}|$. The k-th order statistic of a statistical sample is equal to its kth-smallest value.

Let us call $f_k(\tau)$ the probability density function of the k-th order statistic of $|\mathbf{y}|$. We could then in principle compute the probability of bit mismatch as

$$p_i = \int \mathbb{P}\left(\eta_\tau \leq 0|\tau\right) f_{m_{\text{pool}}-i+1}(\tau)d\tau \quad \text{for } i = 1, 2, \ldots, m$$

and then repeat the same Poisson binomial argument as before. However, this is cumbersome to compute and we instead derive some bounds.

Theorem 3. *Under the same assumptions as Theorem 1, and being* $e_{2(m_{pool}-m+1);2m_{pool}}$ *the expected value of the* $2(m_{pool} - m + 1)$*-th order statistic of a sample of* $\mathcal{N}(0, \sigma^2\|u\|^2)$ *of size* $2m_{pool}$*:*

$$\mathbb{E}\left[d_H(\mathbf{p},\mathbf{q})\right] \leq \frac{1}{2} + \frac{1}{2}\text{erf}\left(\frac{-e_{2(m_{pool}-m+1);2m_{pool}}\mathbf{u}^T\mathbf{v}}{\sqrt{2}\sigma\|\mathbf{u}\|\sqrt{\|\mathbf{u}\|^2\|\mathbf{v}\|^2 - (\mathbf{u}^T\mathbf{v})^2}}\right)$$

Proof: We first notice that $p_i \leq p_m$, $\forall i = 1, \ldots, m$ so that $\mathbb{E}\left[d_H(\mathbf{p},\mathbf{q})\right] = \frac{1}{m}\sum_{i=1}^{m} p_i \leq p_m$. Then, $p_m = \mathbb{E}_\tau\left[g(\tau)\right] \leq g(\mathbb{E}[\tau])$ by applying Jensen's inequality to $g(\tau) = \mathbb{P}\left(\eta_\tau \leq 0|\tau\right)$, i.e., the same Gaussian tail probability as before. Also notice that the convexity of g allows us to use Jensen's inequality. $\mathbb{E}[\tau] = \tilde{e}_{(m_{\text{pool}}-m+1);m_{\text{pool}}}$ is the expected value of the $(m_{\text{pool}} - m + 1)$-th order statistic of a sample of size m_{pool} from a half Gaussian (since we consider $|\mathbf{y}|$). We then notice that that is equivalent [13] to $e_{2(m_{\text{pool}}-m+1);2m_{\text{pool}}}$, i.e., the $2(m_{\text{pool}}-m+1)$-th order statistic of a sample of size $2m_{\text{pool}}$ of a full Gaussian with zero mean and $\sigma^2\|u\|^2$ variance. ■

As a further remark, according to [14] an approximation of the expected value of the desired order statistic is:

$$e_{2(m_{\text{pool}}-m+1);2m_{\text{pool}}} \approx F^{-1}\left(\frac{2(m_{\text{pool}} - m + 1) - \alpha}{2m_{\text{pool}} - 2\alpha + 1}\right)$$

being $\alpha = 0.375$ and F^{-1} the inverse CDF of a normal distribution with zero mean and variance $\sigma^2\|u\|^2$.

So far we considered distances between a test signal and the reference used to adapt the embedding. We will now consider what happens to the distance between any arbitrary pair of signals $\mathbf{v}$ and $\mathbf{w}$. Qualitatively, we can say that the curve of the expected value of the Hamming distance in the embedded space as function of the original distance between $\mathbf{v}$ and $\mathbf{w}$ will be somewhere between the one predicted by Theorem 2 and the one of non-adaptive sign random projections depending on how much $\mathbf{w}$ is close to the reference $\mathbf{u}$. The following theorem formalizes this concept.

Theorem 4. *Let* $\mathcal{X} \subset \mathbb{R}^n$ *be a set of* N *signals and* $\mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathcal{X}$. *Let* $\mathbf{\Phi} \in \mathbb{R}^{m_{pool}\times n}$ *with* $\mathbf{\Phi}_{i,j} \sim \mathcal{N}(0, \sigma^2)$, $\mathbf{y} = \mathbf{\Phi u}$ *and the locations* $\mathbf{l}$ *of the* $m \leq m_{pool}$ *entries in* $\mathbf{y}$ *with largest magnitude be known. Let* $\mathbf{p} = \text{sign}(\mathbf{\Phi_l u})$, $\mathbf{q} = \text{sign}(\mathbf{\Phi_l v})$, *and* $\mathbf{r} = \text{sign}(\mathbf{\Phi_l w})$. *Then for* $0 < \varepsilon \leq 2e - 1$,

$$\mathbb{P}\left(|d_H(\mathbf{q},\mathbf{r}) - \mu| > \epsilon\mu\right) < e^{-\mu\frac{\varepsilon^2}{2}} + e^{-\mu\frac{\varepsilon^2}{4}} \tag{4}$$

with

$$\mu = \mathbb{E}\left[d_H(\mathbf{q},\mathbf{r})\right] = \frac{1}{m}\sum_{i=1}^{m} p_i$$

$$\begin{aligned} p_i = &\left[F\left(\begin{bmatrix}0\\+\infty\end{bmatrix}\right) - F\left(\begin{bmatrix}0\\0\end{bmatrix}\right)\right]\left[1 - F\left(\begin{bmatrix}+\infty\\0\end{bmatrix}\right)\right]\\ &+ \left[F\left(\begin{bmatrix}+\infty\\0\end{bmatrix}\right) - F\left(\begin{bmatrix}0\\0\end{bmatrix}\right)\right]F\left(\begin{bmatrix}+\infty\\0\end{bmatrix}\right),\end{aligned}$$

being F *the CDF of a bivariate Gaussian with mean* $\mu' = \frac{y_{l_i}}{\|\mathbf{u}\|^2}\begin{bmatrix}\mathbf{u}^T\mathbf{v}\\ \mathbf{u}^T\mathbf{w}\end{bmatrix}$ *and covariance*

$$\mathbf{\Sigma}' = \sigma^2 \begin{bmatrix} \|\mathbf{v}\|^2 - \frac{(\mathbf{u}^T\mathbf{v})^2}{\|\mathbf{u}\|^2} & \mathbf{v}^T\mathbf{w} - \frac{(\mathbf{u}^T\mathbf{v})(\mathbf{u}^T\mathbf{w})}{\|\mathbf{u}\|^2} \\ \mathbf{v}^T\mathbf{w} - \frac{(\mathbf{u}^T\mathbf{v})(\mathbf{u}^T\mathbf{w})}{\|\mathbf{u}\|^2} & \|\mathbf{w}\|^2 - \frac{(\mathbf{u}^T\mathbf{w})^2}{\|\mathbf{u}\|^2} \end{bmatrix}.$$

Proof: The proof is similar to the proof of Theorem 2. Let us consider a single measurement in the l_i location $y_i = \mathbf{\Phi}_{l_i}\mathbf{u}$, $z_i = \mathbf{\Phi}_{l_i}\mathbf{v}$, $a_i = \mathbf{\Phi}_{l_i}\mathbf{w}$. Then $\zeta = [z_i, a_i, y_i]$ is Gaussian with zero mean and covariance $\Sigma = \sigma^2 \begin{bmatrix} \|\mathbf{u}\|^2 & \mathbf{u}^T\mathbf{v} & \mathbf{u}^T\mathbf{w} \\ \mathbf{u}^T\mathbf{v} & \|\mathbf{v}\|^2 & \mathbf{v}^T\mathbf{w} \\ \mathbf{u}^T\mathbf{w} & \mathbf{v}^T\mathbf{w} & \|\mathbf{w}\|^2. \end{bmatrix}$. Suppose that y_i is observed to be $y_i = \tau_i$, then the conditional distribution of $[z_i, a_i]$ given $y_i = \tau_i$ is $\eta = ([z_i, a_i]|y_i = \tau_i) \sim \mathcal{N}(\mu', \mathbf{\Sigma}')$ with $\mu' = \frac{y_{l_i}}{\|\mathbf{u}\|^2}\begin{bmatrix} \mathbf{u}^T\mathbf{v} \\ \mathbf{u}^T\mathbf{w} \end{bmatrix}$ and covariance $\mathbf{\Sigma}' = \sigma^2 \begin{bmatrix} \|\mathbf{v}\|^2 - \frac{(\mathbf{u}^T\mathbf{v})^2}{\|\mathbf{u}\|^2} & \mathbf{v}^T\mathbf{w} - \frac{(\mathbf{u}^T\mathbf{v})(\mathbf{u}^T\mathbf{w})}{\|\mathbf{u}\|^2} \\ \mathbf{v}^T\mathbf{w} - \frac{(\mathbf{u}^T\mathbf{v})(\mathbf{u}^T\mathbf{w})}{\|\mathbf{u}\|^2} & \|\mathbf{w}\|^2 - \frac{(\mathbf{u}^T\mathbf{w})^2}{\|\mathbf{u}\|^2} \end{bmatrix}$.

After quantization of the measurements, the probability of mismatching bits in position l_i is

$$p_i = \mathbb{P}(\eta_1 < 0|\eta_2 > 0)\,\mathbb{P}(\eta_2 > 0) + \mathbb{P}(\eta_1 > 0|\eta_2 < 0)\,\mathbb{P}(\eta_2 < 0)$$

Define the random variable E_i as in (3), then $D_H = \frac{1}{m}\sum_{i=1}^{m} E_i$ is a Poisson Binomial random variable measuring the Hamming distance between $\mathbf{q}$ and $\mathbf{r}$. Eq. (4) can be readily obtained using Chernoff bounds [12] for the tails of D_H. ■

The key distinction between sign random projections [3] and the method presented in this paper is that the former provides a linear relationship between the Hamming distance in the embedded space and the angle in the original space. On the other hand, the binary adaptive embedding provides a nonlinear relationship between the two distances, with the important property that the Hamming distances observed with the adaptive embedding are always smaller than those observed with sign random projections. This property is at the core of the improved performance of the embedding for tasks such as binary classification, as discussed in Sec. III. Fig. 1 shows the Hamming distance in the embedded space as function of the inner product between the test signal and the reference signal in the original space. It can be noticed how the curve of the adaptive embedding always lies below the one for the non-adaptive embedding. For a fixed value of m_{pool} increasing the umber of measurements m will reduce the variance and move the expected value towards that of the non-adaptive embedding. Viceversa, for a fixed m increasing m_{pool} will lower the curve. Fig. 2 shows the Hamming distance between the first test signal and the second test signal in the embedded space as function of the inner product between the first test signal and the second test signal ($\mathbf{v}^T\mathbf{w}$) and between the second test signal and the reference signal ($\mathbf{u}^T\mathbf{w}$) in the original space. Notice how for decreasing $\mathbf{u}^T\mathbf{w}$ the shape of the embedding tends to the one of a non-adaptive embedding.

III. Applications and Experiments

A. Low-contrast approximate nearest neighbors

A measure of difficulty [15], [16] of a nearest neighbor search problem is the contrast r, which is defined as the ratio between the distance of the closest false neighbor and that of the farthest true neighbor. Locality sensitive hashing [17] is

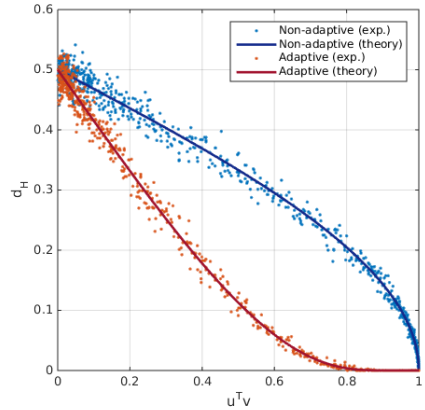

Fig. 1. Hamming distance in the embedding a function of inner product between reference and test signals. Unit norm signals, $m = 800$, $m_{\text{pool}} = 5000$. Non-adaptive curve uses sign random projections [3].

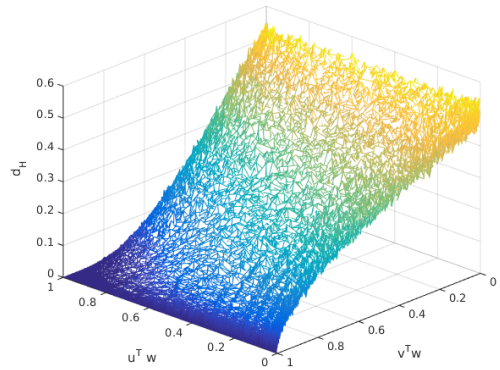

Fig. 2. Hamming distance in the embedding a function of inner product between reference and test signals and between both test signals. Unit norm signals, $m = 800$, $m_{\text{pool}} = 5000$.

able to find approximate nearest neighbors in a time $\mathcal{O}(N^r)$ that is sublinear in the database size N but that degenerates to linear search as the contrast approaches 1. The curse of dimensionality makes low contrast more probable when high-dimensional spaces are considered. The adaptive embedding presented in this paper can be used as a dimensionality reduction technique to solve approximate nearest neighbor search more efficiently in low-contrast scenarios.

In order to show this we develop an experiment in a high-dimensional space where the goal is to find the nearest neighbors of a given signal within a certain radius. True neighbors are generated as standard Gaussian vectors with $n = 8192$ i.i.d. entries with an expected correlation coefficient equal to 0.07 to a reference that is also used as query. Disturbing signals are i.i.d. Gaussian with zero expected correlation. Notice that they are almost orthogonal to each other but the contrast is low because the true neighbours are weakly correlated with our query. This problem is not unrealistic and, in fact, as an example, it occurs in the detection of photo-response non-uniformity artifacts from camera sensors [18] [19], used to attribute a given picture to a given camera sensor. In this experiment all the signals in the database are adaptively embedded, i.e., the locations of the m entries of largest magnitude are identified and stored with the binary code. The random projections of the query are computed and

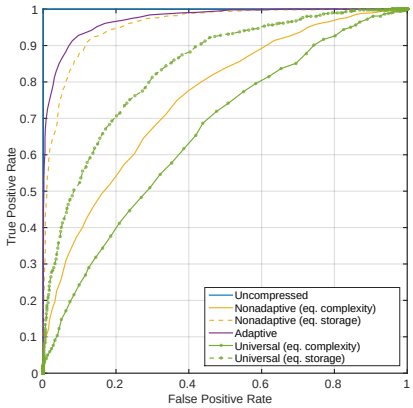

Fig. 3. ROC for nearest neighbor search. A point is declared a neighbor of the query if the Hamming distance is below a given threshold. $n = 8192$, $m_{\text{pool}} = 8192$, $m = 512$. The query is used as reference signal.

appropriately subsampled according to the locations stored for each database signal under test. The storage requirement for each database entry is the sum of the bits needed by the binary measurements and the overhead due to the adaptively chosen locations and it amounts to $m + \log_2 \left[\binom{m_{\text{pool}}}{m}\right]$ bits. Fig. 3 shows the Receiver Operating Characteristic (ROC) showing the probability of detection of a true neighbor against the probability of false alarm. We notice that the adaptive embedding provides a performance closer to the uncompressed case. Two non-adaptive strategies are presented for a fair comparison. A non-adaptive method using the same storage as the adaptive method would use $m' = m + \log_2 \left[\binom{m_{\text{pool}}}{m}\right]$ binary random projections, with the drawback of increased computational complexity in the Hamming distance evaluation. The second strategy equalizes computational complexity, thus using $m' = m$ non-adaptive measurements. This is advantageous in terms of storage but it performs significantly worse. Finally, we compared the proposed method with the Universal Embedding of Boufounos et al. [5] which is a kind of adaptive embedding where the quantizer can be parametrized in order to distort the expected value of signal distances, similarly to the embedding proposed in this paper. The Universal Embedding bounds the maximum distance that is embedded, beyond which points become indistinguishable. It is therefore expected to have poor performance in low-contrast, low-correlation scenarios, as it appears from Fig. 3., where the quantization step is $\Delta = 2$.

B. Multiclass linear classifiers

In this section we apply the adaptive embedding to a multiclass linear classifier in order to improve its storage and computational efficiency. Linear classifiers are widely used in the context of deep neural networks, where the layers of the network are trained to disentangle the features of each class and a simple linear classification layer provides the class labels. A k-class linear classifier can be written as $l = \arg\max_{i=\{1,\dots,k\}} \mathbf{w}_i \mathbf{x}$, being l the class label, $\mathbf{w}_i$ the weights vectors and $\mathbf{x}$ a feature vector. The weights are learned during the training phase using a suitable loss function such as the hinge loss [20] for support vector machines or the softmax cross-entropy [21] for multinomial logistic regression which is more popular in deep neural networks. Since the feature vectors may be high dimensional and the number of classes large, this operation may require significant storage space for the real-valued weights as well as computational resources to compute all the inner products. This can be overcome using an embedding such as sign random projections. After the training phase is completed, the weights are embedded in a compact binary code for each class. During predictions the feature vectors are also embedded, the Hamming distance with the weights is computed and the class label corresponding to the minimum distance is selected, i.e. $l = \arg\min_{i=\{1,\dots,k\}} d_H(\omega_i, \mathbf{y})$, being $\omega_i = \text{sign}(\Phi\mathbf{w})$ and $\mathbf{y} = \text{sign}(\Phi\mathbf{x})$. Replacing sign random projections with the proposed adaptive embedding can improve the classification performance of the compressed system. Overall, there are as many adaptive embeddings as the number of classes. The random projections of each $\mathbf{w}_i$ are used to compute a different set of locations $\mathbf{l}_i$ for each class. At test time, the feature vector is embedded k times to generate $\mathbf{y}_i$ (this amounts to generating m_{pool} non-adaptive projections and then subsampling according to the corresponding $\mathbf{l}_i$). Hence, the class label is given by $l = \arg\min_{i=\{1,\dots,k\}} d_H(\omega_i, \mathbf{y}_i)$, being $\omega_i = \text{sign}(\Phi_{\mathbf{l}_i}\mathbf{w})$ and $\mathbf{y}_i = \text{sign}(\Phi_{\mathbf{l}_i}\mathbf{x})$.

TABLE I. CLASSIFICATION ACCURACY

Method	**Accuracy**			
Uncompressed	**92.64 %**			
	$m = 32$	$m = 64$	$m = 128$	$m = 256$
Adaptive	**92.30%**	**92.33%**	**92.40%**	**92.49%**
Sign random projections (eq. complexity)	75.49%	87.03%	91.09%	92.27%
Universal Embedding (eq. complexity)	77.13%	87.81%	92.10%	92.30%
	$m = 233$	$m = 405$	$m = 680$	$m = 1065$
Sign random projections (eq. storage)	91.90%	92.32%	92.38%	92.51%
Universal Embedding (eq. storage)	92.20%	92.32%	92.39%	92.49%

The following experiment is a classification problem on the CIFAR-10 dataset [22] comprising 10 classes. We implemented the same convolutional neural network architecture presented in [23]. This network is composed of 8 convolutional layers followed by 2 fully connected layers all with ReLU activation units [24] and a final linear layer. The last linear layer outputs one of the $k = 10$ class labels from a $n = 1024$-dimensional input feature vector. After conventional training of the network, we replaced the layer weights with its embedded codes as explained above. For the adaptive method we used $m_{\text{pool}} = 1024$. Table I shows the classification accuracy as function of the number of measurements used by the embedding. It can be noticed that the adaptive embedding allows to achieve a significant dimensionality reduction at a negligible loss in terms of classification accuracy, with respect to both sign random projections [3] and the universal embedding [5]. The quantization step size of the universal embedding has been optimized via cross validation to value $\Delta = 180$.

IV. CONCLUSIONS

This paper presented a technique to generate compact binary codes from high-dimensional signals adapting them to a reference signal. The resulting embedding displays interesting properties that allow to improve performance in classification tasks when those are performed in the reduced-dimensionality domain. Future work will focus on generalizing the approach to sub-Gaussian and structured sensing matrices.

REFERENCES

[1] D. Achlioptas, "Database-friendly random projections: Johnson-Lindenstrauss with binary coins," *Journal of computer and System Sciences*, vol. 66, no. 4, pp. 671–687, 2003.

[2] W. B. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," *Contemporary Mathematics*, vol. 26, 1984.

[3] M. S. Charikar, "Similarity estimation techniques from rounding algorithms," in *Proceedings of the Thiry-fourth Annual ACM Symposium on Theory of Computing*, ser. STOC '02. New York, NY, USA: ACM, 2002, pp. 380–388. [Online]. Available: http://doi.acm.org/10.1145/509907.509965

[4] L. Jacques, J. N. Laska, P. T. Boufounos, and R. G. Baraniuk, "Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors," *IEEE Trans. Inf. Theory*, vol. 59, no. 4, pp. 2082–2102, April 2013.

[5] P. T. Boufounos and S. Rane, "Efficient coding of signal distances using universal quantized embeddings," in *Data Compression Conference (DCC), 2013*, March 2013, pp. 251–260.

[6] A. Broder, "On the resemblance and containment of documents," in *Proceedings of the Compression and Complexity of Sequences 1997*. Washington, DC, USA: IEEE Computer Society, 1997, pp. 21–29. [Online]. Available: http://dl.acm.org/citation.cfm?id=829502.830043

[7] D. Valsesia, S. Fosson, C. Ravazzi, T. Bianchi, and E. Magli, "Sparse-Hash: Embedding Jaccard Coefficient between Supports of Signals," in *Proc. of MM-SPARSE Workshop at ICME 2016*, 2016.

[8] M. Norouzi, D. J. Fleet, and R. R. Salakhutdinov, "Hamming distance metric learning," in *Advances in neural information processing systems*, 2012, pp. 1061–1069.

[9] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Advances in Neural Information Processing Systems 21*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds. Curran Associates, Inc., 2009, pp. 1753–1760. [Online]. Available: http://papers.nips.cc/paper/3383-spectral-hashing.pdf

[10] K. He, F. Wen, and J. Sun, "K-means hashing: An affinity-preserving quantization method for learning binary compact codes," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013.

[11] H. Jegou, L. Amsaleg, C. Schmid, and P. Gros, "Query adaptative locality sensitive hashing," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, March 2008, pp. 825–828.

[12] M. Mitzenmacher and E. Upfal, *Probability and computing: Randomized algorithms and probabilistic analysis*. Cambridge University Press, 2005.

[13] N. Balakrishnan and A. C. Cohen, *Order statistics & inference: estimation methods*. Elsevier, 2014.

[14] H. L. Harter, "Expected values of normal order statistics," *Biometrika*, vol. 48, no. 1/2, pp. 151–165, 1961. [Online]. Available: http://www.jstor.org/stable/2333139

[15] J. He, S. Kumar, and S.-F. Chang, "On the difficulty of nearest neighbor search," in *ICML 2012*.

[16] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, *When Is "Nearest Neighbor" Meaningful?* Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 217–235. [Online]. Available: http://dx.doi.org/10.1007/3-540-49257-7_15

[17] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," *Commun. ACM*, vol. 51, no. 1, pp. 117–122, Jan. 2008.

[18] D. Valsesia, G. Coluccia, T. Bianchi, and E. Magli, "Compressed fingerprint matching and camera identification via random projections," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 7, pp. 1472–1485, July 2015.

[19] ——, "Large-scale image retrieval based on compressed camera identification," *IEEE Transactions on Multimedia*, vol. 17, no. 9, pp. 1439–1449, Sept 2015.

[20] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[21] C. M. Bishop, "Pattern recognition," *Machine Learning*, vol. 128, 2006.

[22] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," 2009.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[24] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.