Diffusion Ensemble Classifiers

Alon Schclar[1], Lior Rokach[2] and Amir Amit[3]

[1]*School of Computer Science, Academic College of Tel Aviv-Yaffo, P.O.B 8401, Tel Aviv 61083, Israel*

[2]*Department of Information Systems Engineering, Ben-Gurion University of the Negev,*
P.O.B 653, Beer-Sheva 84105, Israel

[3]*The Efi Arazi School of Computer Science, Interdisciplinary Center (IDC) Herzliya, P.O.B 167, Herzliya 46150, Israel*

Keywords: Ensemble Classifiers, Dimensionality Reduction, Out-of-Sample Extension, Diffusion Maps, Nyström Extension.

Abstract: We present a novel approach for the construction of ensemble classifiers based on the Diffusion Maps (DM) dimensionality reduction algorithm. The DM algorithm embeds data into a low-dimensional space according to the connectivity between every pair of points in the ambient space. The ensemble members are trained based on dimension-reduced versions of the training set. These versions are obtained by applying the DM algorithm to the original training set using different values of the input parameter. In order to classify a test sample, it is first embedded into the dimension reduced space of each individual classifier by using the Nyström out-of-sample extension algorithm. Each ensemble member is then applied to the embedded sample and the classification is obtained according to a voting scheme. A comparison is made with the base classifier which does not incorporate dimensionality reduction. The results obtained by the proposed algorithms improve on average the results obtained by the non-ensemble classifier.

1 INTRODUCTION

Classifiers are predictive models which label data based on a training dataset T whose labels are known *a-priory*. A classifier is constructed by applying an induction algorithm, or inducer, to T - a process that is commonly known as *training*. Classifiers differ by the induction algorithms and training sets that are used for their construction. Common induction algorithms include nearest neighbors (NN), decision trees (CART (Breiman et al., 1993), C4.5 (Quinlan, 1993)), Support Vector Machines (SVM) (Vapnik, 1999) and Artificial Neural Networks - to name a few. Since every inducer has its advantages and weaknesses, methodologies have been developed to enhance their performance. Ensemble classifiers are one of the most common ways to achieve that.

The need for dimensionality reduction techniques emerged in order to alleviate the so called *curse of dimensionality* - the fact that the complexity of many algorithms grows exponentially with the increase of the input data dimensionality (Jimenez and Landgrebe, 1998). In many cases a high-dimensional dataset lies approximately on a low-dimensional manifold in the ambient space. Dimensionality reduction methods *embed* datasets into a low-dimensional space while preserving as much possible the information that is conveyed by the dataset. The low-dimensional representation is referred to as the *embedding* of the dataset. Since the information is inherent in the geometrical structure of the dataset (e.g. clusters), a good embedding distorts the structure as little as possible while representing the dataset using a number of features that is substantially lower than the dimension of the original ambient space. Furthermore, an effective dimensionality reduction algorithm also removes noisy features and inter-feature correlations.

1.1 Ensembles of Classifiers

Ensembles of classifiers (Kuncheva, 2004) mimic the human nature to seek advice from several people before making a decision where the underlying assumption is that combining the opinions will produce a decision that is better than each individual opinion. Several classifiers (ensemble *members*) are constructed and their outputs are combined - usually by voting or an averaged weighting scheme - to yield the final classification (Polikar, 2006; Opitz and Maclin, 1999). In order for this approach to be effective, two crite-

Schclar A., Rokach L. and Amit A..
Diffusion Ensemble Classifiers.
DOI: 10.5220/0004102804430450
In *Proceedings of the 4th International Joint Conference on Computational Intelligence* (NCTA-2012), pages 443-450
ISBN: 978-989-8565-33-4

ria must be met: *accuracy* and *diversity* (Kuncheva, 2004). Accuracy requires each individual classifier to be as accurate as possible i.e. individually minimize the generalization error. Diversity requires minimizing the correlation among the generalization errors of the classifiers. These criteria are contradictory since optimal accuracy achieves a minimum and unique error which contradicts the requirement of diversity. Complete diversity, on the other hand, corresponds to random classification which usually achieves the worst accuracy. Consequently, individual classifiers that produce results which are moderately better than random classification are suitable as ensemble members.

In this paper we focus on ensemble classifiers that use a single induction algorithm, for example the Naïve Bayes inducer. This ensemble construction approach achieves its diversity by manipulating the training set. A well known way to achieve diversity is by bootstrap aggregation (*Bagging*) (Breiman, 1996). Several training sets are constructed by applying bootstrap sampling (each sample may be drawn more than once) to the original training set. Each training set is used to construct a different classifier where the repetitions fortify different training instances. This method is simple yet effective and has been successfully applied to a variety of problems such as spam detection (Yang et al., 2006), analysis of gene expressions (Valentini et al., 2003) and user identification (Feher et al., 2012).

The award winning Adaptive Boosting (*AdaBoost*) (Freund and Schapire, 1996) algorithm and its subsequent versions e.g. (Drucker, 1997) and (Solomatine and Shrestha, 2004) provide a different approach for the construction of ensemble classifiers based on a single induction algorithm. This approach iteratively assigns weights to each training sample where the weights of the samples that are misclassified are increased according to a global error coefficient. The final classification combines the logarithm of the weights to yield the ensemble's classification.

Successful applications of the ensemble methodology can be found in many fields such as recommender systems (Schclar et al., 2009), classification (Schclar and Rokach, 2009), finance (Leigh et al., 2002), manufacturing (Rokach, 2008) and medicine (Mangiameli et al., 2004), to name a few.

1.2 Dimensionality Reduction

The dimensionality reduction problem can be formally described as follows. Let

$$\Gamma = \{x_i\}_{i=1}^{N} \tag{1}$$

be the original high-dimensional dataset given as a set of column vectors where $x_i \in \mathbb{R}^n$, n is the dimension of the ambient space and N is the size of the dataset. All dimensionality reduction methods embed the vectors into a lower dimensional space $\mathbb{R}^q$ where $q \ll n$. Their output is a set of column vectors in the lower dimensional space

$$\widetilde{\Gamma} = \{\widetilde{x}_i\}_{i=1}^{N}, \widetilde{x}_i \in \mathbb{R}^q \tag{2}$$

where q is chosen such that it approximates the intrinsic dimensionality of Γ (Hein and Audibert, 2005; Hegde et al., 2007). We refer to the vectors in the set $\widetilde{\Gamma}$ as the *embedding vectors*.

Dimensionality techniques can be divided into *global* and *local* methods. The former derive embeddings in which *all* points satisfy a given criterion. Examples for global methods include: Principal Component Analysis (PCA) (Hotelling, 1933), Kernel PCA (KPCA) (Schölkopf et al., 1998; Schölkopf and Smola, 2002), Multidimensional scaling (MDS) (Kruskal, 1964; Cox and Cox, 1994), ISOMAP (Tenenbaum et al., 2000), etc. Contrary to global methods, local methods construct embeddings in which only *local* neighborhoods are required to meet a given criterion. The global description of the dataset is derived by the aggregation of the local neighborhoods. Common local methods include Local Linear Embedding (LLE) (Roweis and Saul, 2000), Laplacian Eigenmaps (Belkin and Niyogi, 2003), Hessian Eigenmaps (Donoho and Grimes, 2003) and Diffusion Maps (Coifman and Lafon, 2006a; Schclar, 2008) which is used in this paper and is described in Section 3.

A key aspect of dimensionality reduction is how to efficiently embed a *new point* into a *given* dimension-reduced space. This is commonly referred to as *out-of-sample extension* where the sample stands for the original dataset whose dimensionality was reduced and does not include the new point. An accurate embedding of a new point requires the recalculation of the entire embedding. This is impractical in many cases, for example, when the time and space complexity that are required for the dimensionality reduction is quadratic (or higher) in the size of the dataset. An efficient out-of-sample extension algorithm embeds the new point without recalculating the entire embedding - usually at the expense of the embedding accuracy.

The Nyström extension (Nyström, 1928) algorithm, which is used in this paper, embeds a new point in linear time using the quadrature rule when the dimensionality reduction involves eigen-decomposition of a kernel matrix. Algorithms such as Diffusion Maps, Laplacian Eigenmaps, ISOMAP, LLE, etc. are

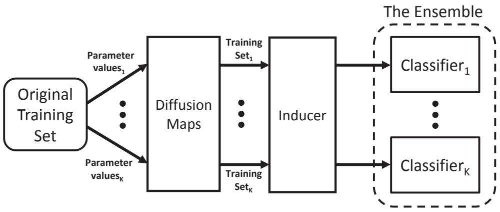

Figure 1: Ensemble training.

examples that fall into this category and, thus, the embeddings that they produce can be extended using the Nyström extension (Ham et al., 2004; Bengio et al., 2004). A formal description of the Nyström extension is given in the Sec. 4.

The main contribution of this paper is a novel framework for the construction of ensemble classifiers based on the Diffusion Maps dimensionality reduction algorithm coupled with the Nyström out-of-sample extension. The rest of this paper is organized as follows. In Section 2 we describe the proposed approach. In Section 3 we describe the Diffusion Maps dimensionality reduction algorithm. The Nyström out-of-sample extension algorithm is described in Section 4. Experimental results are given in Section 5. We conclude and describe future work in Section 6.

2 DIFFUSION ENSEMBLE CLASSIFIERS

The proposed approach achieves the diversity requirement of ensemble classifiers by applying the DM dimensionality reduction algorithm to a given training set using different values for its input parameter. After the training sets are produced by the DM dimensionality reduction algorithms, each set is used to train a classifier to produce one of the ensemble members. The training process is illustrated in Fig. 1.

Employing the DM dimensionality reduction to a training set has the following advantages:

- It reduces noise and decorrelates the data.
- It reduces the computational complexity of the classifier construction and consequently the complexity of the classification.
- It can alleviate over-fitting by constructing combinations of the variables (Plastria et al., 2008).

These points meet the accuracy and diversity criteria which are required to construct an effective ensemble classifier and thus render dimensionality reduction a technique which is tailored for the construction of ensemble classifiers. Specifically, removing noise from

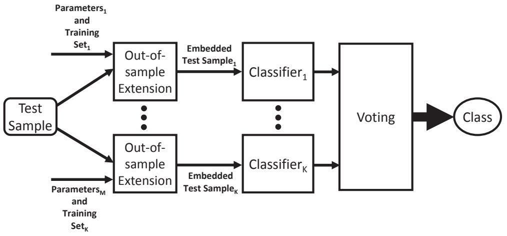

Figure 2: Classification process of a test sample.

the data contributes to the accuracy of the classifier while diversity is obtained by the various dimension-reduced versions of the data.

In order to classify test samples they are first embedded into the low-dimensional space of each of the training sets using the Nyström out-of-sample extension. Next, each ensemble member is applied to its corresponding embedded test sample and the produced results are processed by a voting scheme to derive the result of the ensemble classifier. Specifically, each classification is given as a vector containing the probabilities of each possible label. These vectors are aggregated and the ensemble classification is chosen as the label with the largest probability. Figure 2 depicts the classification process of a test sample.

3 DIFFUSION MAPS

The Diffusion Maps (DM) (Coifman and Lafon, 2006a) algorithm embeds data into a low-dimensional space where the geometry of the dataset is defined in terms of the connectivity between every pair of points in the ambient space. Namely, the similarity between two points x and y is determined according to the number of paths connecting x and y via points in the dataset. This measure is robust to noise since it takes into account all the paths connecting x and y. The Euclidean distance between x and y in the dimension-reduced space approximates their connectivity in the ambient space.

Formally, let Γ be a set of points in $\mathbb{R}^n$ as defined in Eq. 1. A weighted undirected graph $G(V,E)$, $|V| = N$, $|E| \ll N^2$ is constructed, where each vertex $v \in V$ corresponds to a point in Γ. The weights of the edges are chosen according to a weight function $w_\varepsilon(x,y)$ which measures the similarities between every pair of points where the parameter ε defines a local neighborhood for each point. The weight function is defined by a kernel function obeying the following properties:

Symmetry: $\forall x_i, x_j \in \Gamma,\ w_\varepsilon(x_i,x_j) = w_\varepsilon(x_j,x_i)$

Non-negativity: $\forall x_i, x_j \in \Gamma,\ w_\varepsilon(x_i,x_j) \geq 0$

Positive Semi-definite: for every real-valued bounded function f defined on Γ,

$\sum_{x_i,x_j\in\Gamma} w_\varepsilon(x_i,x_j)\, f(x_i)\, f(x_j) \geq 0$.

Fast Decay: $w_\varepsilon(x_i,x_j) \to 0$ when $\|x_i - x_j\| \gg \varepsilon$ and $w_\varepsilon(x_i,x_j) \to 1$ when $\|x_i - x_j\| \ll \varepsilon$. This property facilitates the representation of w_ε by a sparse matrix.

A common choice that meets these criteria is the Gaussian kernel:

$$w_\varepsilon(x_i,x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\varepsilon}}.$$

A weight matrix w_ε is used to represent the weights of the edges. Given a graph G, the Graph Laplacian normalization (Chung, 1997) is applied to the weight matrix w_ε and the result is given by M:

$$M_{i,j} \triangleq m(x,y) = \frac{w_\varepsilon(x,y)}{d(x)}$$

where $d(x) = \sum_{y\in\Gamma} w_\varepsilon(x,y)$ is the degree of x. This transforms w_ε into a Markov transition matrix corresponding to a random walk through the points in Γ. The probability to move from x to y in *one* time step is denoted by $m(x,y)$. These probabilities measure the connectivity of the points within the graph.

The transition matrix M is conjugate to a symmetric matrix A whose elements are given by $A_{i,j} \triangleq a(x,y) = \sqrt{d(x)}m(x,y)\frac{1}{\sqrt{d(y)}}$. Using matrix notation, A is given by $A = D^{\frac{1}{2}}MD^{-\frac{1}{2}}$, where D is a diagonal matrix whose values are given by $d(x)$. The matrix A has n real eigenvalues $\{\lambda_l\}_{l=0}^{n-1}$ where $0 \leq \lambda_l \leq 1$, and a set of orthonormal eigenvectors $\{v_l\}_{l=1}^{N-1}$ in $\mathbb{R}^n$. Thus, A has the following spectral decomposition:

$$a(x,y) = \sum_{k\geq 0} \lambda_k v_l(x) v_l(y). \tag{3}$$

Since M is conjugate to A, the eigenvalues of both matrices are identical. In addition, if $\{\phi_l\}$ and $\{\psi_l\}$ are the left and right eigenvectors of M, respectively, then the following equalities hold:

$$\phi_l = D^{\frac{1}{2}} v_l, \quad \psi_l = D^{-\frac{1}{2}} v_l. \tag{4}$$

From the orthonormality of $\{v_i\}$ and Eq. 4 it follows that $\{\phi_l\}$ and $\{\psi_l\}$ are bi-orthonormal i.e. $\langle \phi_m, \psi_l \rangle = \delta_{ml}$ where $\delta_{ml} = 1$ when $m = l$ and $\delta_{ml} = 0$, otherwise. Combing Eqs. 3 and 4 together with the bi-orthogonality of $\{\phi_l\}$ and $\{\psi_l\}$ leads to the following eigen-decomposition of the transition matrix M

$$m(x,y) = \sum_{l\geq 0} \lambda_l \psi_l(x) \phi_l(y). \tag{5}$$

When the spectrum decays rapidly (provided ε is appropriately chosen - see Sec. 3.1), only a few terms are required to achieve a given accuracy in the sum. Namely,

$$m(x,y) \simeq \sum_{l=0}^{n(p)} \lambda_l \psi_l(x) \phi_l(y)$$

where $n(p)$ is the number of terms which are required to achieve a given precision p.

We recall the *diffusion distance* between two data points x and y as it was defined in (Coifman and Lafon, 2006a):

$$D^2(x,y) = \sum_{z\in\Gamma} \frac{(m(x,z) - m(z,y))^2}{\phi_0(z)}. \tag{6}$$

This distance reflects the geometry of the dataset and it depends on the number of paths connecting x and y. Substituting Eq. 5 in Eq. 6 together with the bi-orthogonality property allows to express the diffusion distance using the right eigenvectors of the transition matrix M:

$$D^2(x,y) = \sum_{l\geq 1} \lambda_l^2 (\psi_l(x) - \psi_l(y))^2. \tag{7}$$

Thus, the family of Diffusion Maps $\{\Psi(x)\}$ which is defined by

$$\Psi(x) = (\lambda_1\psi_1(x), \lambda_2\psi_2(x), \lambda_3\psi_3(x), \cdots) \tag{8}$$

embeds the dataset into a Euclidean space. In the new coordinates of Eq. 8, the *Euclidean* distance between two points in the embedding space is equal to the *diffusion* distance between their corresponding two high dimensional points as defined by the random walk. Moreover, this facilitates the embedding of the original points into a low-dimensional Euclidean space $\mathbb{R}^q$ by:

$$\Xi_t : x_i \to \left(\lambda_2^t \psi_2(x_i), \lambda_3^t \psi_3(x_i), \ldots, \lambda_{q+1}^t \psi_{q+1}(x_i)\right). \tag{9}$$

which also endows coordinates on the set Γ. Since $\lambda_1 = 1$ and $\psi_1(x)$ is constant, the embedding uses $\lambda_2, \ldots, \lambda_{q+1}$. Essentially, $q \ll n$ due to the fast decay of the eigenvalues of M. Furthermore, q depends only on the dimensionality of the data as captured by the random walk and not on the original dimensionality of the data. Diffusion maps have been successfully applied for acoustic detection of moving vehicles (Schclar et al., 2010) and fusion of data and multicue data matching (Lafon et al., 2006).

3.1 Choosing ε

The choice of ε is critical to achieve the optimal performance by the DM algorithm since it defines the size of the local neighborhood of each point. On one hand, a large ε produces a coarse analysis of the data as the neighborhood of each point will contain a large number of points. In this case, the diffusion distance will be close to 1 for most pairs of points. On the other hand, a small ε might produce many neighborhoods that contain only a single point. In this case, the diffusion distance is zero for most pairs of points. The best choice lies between these two extremes. Accordingly, the ensemble classifier which is based on the the Diffusion Maps algorithm will construct different versions of the training set using different values of ε which will be chosen between the shortest and longest pairwise distances.

4 THE NYSTRÖM OUT-OF-SAMPLE EXTENSION

The Nyström extension (Nyström, 1928) is an extrapolation method that facilitates the extension of any function $f : \Gamma \to \mathbb{R}$ to a set of new points which are added to Γ. Such extensions are required in on-line processes in which new samples arrive and a function f that is defined on Γ needs to be extrapolated to include the new points. These settings exactly fit the settings of the proposed approach since the test samples are given *after* the dimensionality of the training set was reduced. Specifically, the Nyström extension is used to embed a new point into the reduced-dimension space where every coordinate of the low-dimensional embedding constitutes a function that needs to be extended.

We describe the Nyström extension scheme for the Gaussian kernel that is used by the Diffusion Maps algorithm. Let Γ be a set of points in $\mathbb{R}^n$ and Ψ be its embedding (Eq. 8). Let $\bar{\Gamma}$ be a set in $\mathbb{R}^n$ such that $\Gamma \subset \bar{\Gamma}$. The Nyström extension scheme extends Ψ onto the dataset $\bar{\Gamma}$. Recall that the eigenvectors and eigenvalues form the dimension-reduced coordinates of Γ (Eq. 9). The eigenvectors and eigenvalues of a Gaussian kernel with width ε which is used to measure the pairwise similarities in the training set Γ are computed according to

$$\lambda_l \varphi_l(x) = \sum_{y \in \Gamma} e^{-\frac{\|x-y\|^2}{2\varepsilon}} \varphi_l(y), \; x \in \Gamma. \qquad (10)$$

If $\lambda_l \neq 0$ for every l, the eigenvectors in Eq. 10 can be extended to any $x \in \mathbb{R}^n$ by

$$\bar{\varphi}_l(x) = \frac{1}{\lambda_l} \sum_{y \in \Gamma} e^{-\frac{\|x-y\|^2}{2\varepsilon}} \varphi_l(y), \; x \in \mathbb{R}^n. \qquad (11)$$

Let f be a function on the training set Γ and let $x \notin \Gamma$ be a new point. In the Diffusion Maps setting, we are interested in approximating

$$\Psi(x) = \left(\lambda_2 \psi_2(x), \lambda_3 \psi_3(x), \cdots, \lambda_{q+1} \psi_{q+1}(x)\right).$$

The eigenfunctions $\{\varphi_l\}$ are the outcome of the spectral decomposition of a symmetric positive matrix. Thus, they form an orthonormal basis in $\mathbb{R}^N$ where N is the number of points in Γ. Consequently, any function f can be written as a linear combination of this basis:

$$f(x) = \sum_l \langle \varphi_l, f \rangle \varphi_l(x), \; x \in \Gamma.$$

Using the Nyström extension, as given in Eq. 11, f can be defined for any point in $\mathbb{R}^n$ by

$$\bar{f}(x) = \sum_l \langle \varphi_l, f \rangle \bar{\varphi_l}(x), \; x \in \mathbb{R}^n. \qquad (12)$$

The above extension facilitates the decomposition of every diffusion coordinate ψ_i as $\psi_i(x) = \sum_l \langle \varphi_l, \psi_i \rangle \varphi_l(x), \; x \in \Gamma$. In addition, the embedding of a new point $\bar{x} \in \bar{\Gamma} \backslash \Gamma$ can be evaluated in the embedding coordinate system by $\bar{\psi_i}(\bar{x}) = \sum_l \langle \varphi_l, \psi_i \rangle \bar{\varphi_l}(\bar{x})$.

Note that the scheme is ill conditioned since $\lambda_l \longrightarrow 0$ as $l \longrightarrow \infty$. This can be solved by cutting-off the sum in Eq. 12 and keeping only the eigenvalues (and their corresponding eigenfunctions) that satisfy $\lambda_l \geq \delta \lambda_0$ (where $0 < \delta \leq 1$ and the eigenvalues are given in descending order of magnitude):

$$\bar{f}(x) = \sum_{\lambda_l \geq \delta \lambda_0} \langle \varphi_l, f \rangle \bar{\varphi_l}(x), \; x \in \mathbb{R}^n. \qquad (13)$$

The result is an extension scheme with a condition number δ. In this new scheme, f and $\bar{f}$ do not coincide on Γ but they are relatively close. The value of ε controls this error. Thus, choosing ε carefully may improve the accuracy of the extension.

5 EXPERIMENTAL RESULTS

In order to evaluate the proposed approach, we used the WEKA framework (Hall et al., 2009). We tested our approach on 13 datasets from the UCI repository (Asuncion and Newman, 2007) which contains benchmark datasets that are commonly used to evaluate machine learning algorithms. The number of features in the datasets range from 7 to 617 giving a broad spectrum of ambient space dimensionalities. The list of datasets and their properties are summarized in Table 1.

5.1 Experiment Configuration

All ensemble algorithms were tested using he following inducers: (a) decision tree (WEKA's J48 inducer); and (b) Naïve Bayes. The ensembles were composed of ten classifiers and the dimension-reduced space

Table 1: Properties of the benchmark datasets used for the evaluation.

Dataset Name	Instances	Features	Labels
Musk	6598	166	2
Ecoli	335	7	8
Glass	214	9	7
Hill Valley with noise	1212	100	2
Hill Valley without noise	1212	100	2
Ionosphere	351	34	2
Isolet	7797	617	26
Letter	20000	16	26
Madelon	2000	500	2
Sat	6435	36	7
Waveform with noise	5000	40	3
Waveform without noise	5000	21	3

Table 2: Results of the Diffusion Maps ensemble classifier based on the decision-tree (WEKA's J48) and Naïve Bayes inducers.

Dataset	Plain J48	DME (J48)	Plain NB	DME (NB)
Musk	96.88 ± 0.63	96.76 ± 0.72	83.86 ± 2.03	94.13 ± 0.50
Ecoli	84.23 ± 7.51	83.02 ± 4.10	85.40 ± 5.39	84.52 ± 5.43
Glass	65.87 ± 8.91	65.39 ± 10.54	49.48 ± 9.02	59.29 ± 11.09
Hill Valley with noise	49.67 ± 0.17	52.39 ± 3.56	49.50 ± 2.94	50.82 ± 2.93
Hill Valley w/o noise	50.49 ± 0.17	51.23 ± 4.40	51.57 ± 2.64	51.74 ± 3.25
Ionosphere	91.46 ± 3.27	88.04 ± 4.80	82.62 ± 5.47	92.59 ± 4.71
Isolet	83.97 ± 1.65	90.10 ± 0.62	85.15 ± 0.96	91.83 ± 0.96
Letter	87.98 ± 0.51	89.18 ± 0.79	64.11 ± 0.76	58.31 ± 0.70
Madelon	70.35 ± 3.78	76.15 ± 3.43	58.40 ± 0.77	55.10 ± 4.40
Multiple features	94.75 ± 1.92	93.25 ± 1.64	95.35 ± 1.40	89.05 ± 2.09
Sat	85.83 ± 1.04	91.34 ± 0.48	79.58 ± 1.46	85.63 ± 1.25
Waveform with noise	75.08 ± 1.33	86.52 ± 1.78	80.00 ± 1.96	84.36 ± 1.81
Waveform w/o noise	75.94 ± 1.36	86.96 ± 1.49	81.02 ± 1.33	82.94 ± 1.62
Average improvement	**4.3%**		**4.8%**	

was set to half of the original dimension of the data. Ten-fold cross validation was used to evaluate each ensemble's performance on each of the datasets. The constructed ensemble classifiers were compared with: a non-ensemble classifier which applied the induction algorithm to the dataset without dimensionality reduction (we refer to this classifier as the *plain* classifier). We used the default values of the parameters of the WEKA built-in ensemble classifiers in all the experiments. For the sake of simplicity, in the following we refer to the Diffusion Maps ensemble classifier as the DME classifier.

5.2 Results

Table 2 describes the results obtained by the decision tree and Naïve Bayes inducers, respectively. The results provide the classification accuracy along with the variance of the results. The Plain J48 and Plain NB columns refer to the non-ensemble classifiers where the DME(J48) and DME(NB) columns refer to the Diffusion Maps ensemble classifiers which are constructed using the decision tree and Naïve Bayes inducers, respectively. It can be seen that in both cases the average classification accuracy is improved. Specifically, the decision-tree inducer improves the classification accuracy by 4.3% (8 out of the 13 datasets - 5 of which with statistical significance), while the Naïve Bayes inducer improves it by 4.8% (9 out of the 13 datasets - 6 with statistical significance).

6 CONCLUSIONS AND FUTURE WORK

In this paper we introduced the Diffusion Maps dimensionality reduction algorithm as a framework for the construction of ensemble classifiers which use a single induction algorithm. The DM algorithm was applied to the training set using different values for its input parameter. This produced different versions of the training set and the ensemble members were constructed based on these training set versions. In order to classify a new sample, it was first embedded into the dimension-reduced space of each training set using the Nyström out-of-sample extension algorithm. The results in this paper show that the proposed approach is effective. The results were superior in most of the datasets compared to the plain algorithm. The authors are currently extending this approach to other dimensionality reduction techniques. Additionally, other out-of-sample extension schemes should also be explored e.g. the Geometric Harmonics (Coifman and Lafon, 2006b). Lastly, a heterogeneous model which combines several dimensionality reduction techniques should be investigated.

REFERENCES

Asuncion, A. and Newman, D. J. (2007). UCI machine learning repository. http://archive.ics.uci.edu/ml/.

Belkin, M. and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396.

Bengio, Y., Delalleau, O., Roux, N. L., Paiement, J. F., Vincent, P., and Ouimet, M. (2004). Learning eigenfunctions links spectral embedding and kernel pca. *Neural Computation*, 16(10):2197–2219.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1993). *Classification and Regression Trees*. Chapman & Hall, Inc., New York.

Chung, F. R. K. (1997). *Spectral Graph Theory*. AMS Regional Conference Series in Mathematics, 92.

Coifman, R. R. and Lafon, S. (2006a). Diffusion maps. *Applied and Computational Harmonic Analysis: special issue on Diffusion Maps and Wavelets*, 21:5–30.

Coifman, R. R. and Lafon, S. (2006b). Geometric harmonics: a novel tool for multiscale out-of-sample extension of empirical functions. *Applied and Computational Harmonic Analysis: special issue on Diffusion Maps and Wavelets*, 21:31–52.

Cox, T. and Cox, M. (1994). *Multidimensional scaling*. Chapman & Hall, London, UK.

Donoho, D. L. and Grimes, C. (2003). Hessian eigenmaps: new locally linear embedding techniques for high-dimensional data. In *Proceedings of the National Academy of Sciences*, volume 100(10), pages 5591–5596.

Drucker, H. (1997). Improving regressor using boosting. In Jr., D. H. F., editor, *Proceedings of the 14th International Conference on Machine Learning*, pages 107–115. Morgan Kaufmann.

Feher, C., Elovici, Y., Moskovitch, R., Rokach, L., and Schclar, A. (2012). User identity verification via mouse dynamics. *Information Sciences*, 201:19–36.

Freund, Y. and Schapire, R. (1996). Experiments with a new boosting algorithm. machine learning. In *Proceedings for the Thirteenth International Conference*, pages 148–156, San Francisco. Morgan Kaufmann.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: An update. *SIGKDD Explorations*, 11:1.

Ham, J., Lee, D., Mika, S., and Scholköpf, B. (2004). A kernel view of the dimensionality reduction of manifolds. In *Proceedings of the 21st International Conference on Machine Learning (ICML'04)*, pages 369–376, New York, NY, USA. ACM Press.

Hegde, C., Wakin, M., and Baraniuk, R. G. (2007). Random projections for manifold learning. In *Neural Information Processing Systems (NIPS)*.

Hein, M. and Audibert, Y. (2005). Intrinsic dimensionality estimation of submanifolds in Euclidean space. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 289–296.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417–441.

Jimenez, L. O. and Landgrebe, D. A. (1998). Supervised classification in high-dimensional space: geometrical, statistical and asymptotical properties of multivariate data. *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews,*, 28(1):39–54.

Kruskal, J. B. (1964). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29:1–27.

Kuncheva, L. I. (2004). Diversity in multiple classifier systems (editorial). *Information Fusion*, 6(1):3–4.

Lafon, S., Keller, Y., and Coifman, R. R. (2006). Data fusion and multicue data matching by diffusion maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:1784–1797.

Leigh, W., Purvis, R., and Ragusa, J. M. (2002). Forecasting the nyse composite index with technical analysis, pattern recognizer, neural networks, and genetic algorithm: a case study in romantic decision support. *Decision Support Systems*, 32(4):361–377.

Mangiameli, P., West, D., and Rampal, R. (2004). Model selection for medical diagnosis decision support systems. *Decision Support Systems*, 36(3):247–259.

Nyström, E. J. (1928). Über die praktische auflösung von linearen integralgleichungen mit anwendungen auf randwertaufgaben der potentialtheorie. *Commentationes Physico-Mathematicae*, 4(15):1–52.

Opitz, D. and Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169 to 198.

Plastria, F., Bruyne, S., and Carrizosa, E. (2008). Dimensionality reduction for classification. *Advanced Data Mining and Applications*, 1:411–418.

Polikar, R. (2006). ”ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6:21 t o 45.

Quinlan, R. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc.

Rokach, L. (2008). Mining manufacturing data using genetic algorithm-based feature set decomposition. *International Journal of Intelligent Systems Technologies and Applications*, 4(1/2):57–78.

Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326.

Schclar, A. (2008). A diffusion framework for dimensionality reduction. In *Soft Computing for Knowledge Discovery and Data Mining (Editors: O. Maimon and L. Rokach)*, pages 315–325. Springer.

Schclar, A., Averbuch, A., Rabin, N., Zheludev, V., and Hochman, K. (2010). A diffusion framework for detection of moving vehicles. *Digital Signal Processing*, 20:111–122.

Schclar, A. and Rokach, L. (2009). Random projection ensemble classifiers. In *Lecture Notes in Business Information Processing, Enterprise Information Systems 11th International Conference Proceedings (ICEIS'09)*, pages 309–316, Milan, Italy.

Schclar, A., Tsikinovsky, A., Rokach, L., Meisels, A., and Antwarg, L. (2009). Ensemble methods for improving the performance of neighborhood-based collaborative filtering. In *RecSys*, pages 261–264.

Schölkopf, B., Smola, A., and Muller, K. R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319.

Schölkopf, B. and Smola, A. J. (2002). *Learning with Kernels*. MIT Press, Cambridge, MA.

Solomatine, D. P. and Shrestha, D. L. (2004). Adaboost.rt: A boosting algorithm for regression problems. In *Proceedings of the IEEE International Joint Conference on Neural Networks*, pages 1163–1168.

Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323.

Valentini, G., Muselli, M., and Ruffino, F. (2003). Bagged ensembles of svms for gene expression data analysis. In *Proceeding of the International Joint Conference on Neural Networks - IJCNN*, pages 1844–1849, Portland, OR, USA. Los Alamitos, CA: IEEE Computer Society.

Vapnik, V. N. (1999). *The Nature of Statistical Learning Theory (Information Science and Statistics)*. Springer.

Yang, Z., Nie, X., Xu, W., and Guo, J. (2006). An approach to spam detection by naïve bayes ensemble based on decision induction. In *Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA'06)*.