An Assessment of Alternative Strategies for Constructing EMD-Based Kernel Functions for Use in an SVM for Image Classification

Anton Zamolotskikh[1] and Pádraig Cunningham[2]

[1] Trinity College Dublin anton.zamolotskikh@cs.tcd.ie
[2] University College Dublin padraig.cunningham@ucd.ie

Technical Report UCD-CSI-2007-2
March 21, 2007

Abstract. Because of their sound theoretical underpinnings, Support Vector Machines (SVMs) have very impressive performance in classification. However, the use of SVMs is constrained by the fact that the affinity measure that is used to build the classifier must produce a kernel matrix that is positive semi-definite (PSD). This is normally not a problem, however many very effective affinity measures are known that will not produce a PSD kernel matrix. One such measure is the Earth-Mover's Distance (EMD) for quantifying the difference between images. In this paper we consider three methods for producing a PSD kernel from the EMD and compare SVM-based classifiers that use these measures against a Nearest Neighbour classifier built directly on the EMD. We find that two of these *kernelised* EMD measures are effective and the resulting SVMs are better than the Nearest Neighbour alternatives.

1 Introduction

In recent years SVMs have taken a leading role in machine learning research – this is largely due to their impressive performance that derives from sound theoretical foundations. Part of the elegance of the SVM idea derives from a neat separation of concerns. The affinity between objects is captured by the kernel function and stored in the kernel matrix, then this is the only representation of the problem domain that is considered in the quadratic optimisation process that produces the classifier [1]. This does mean however that there is a considerable burden on the kernel function to encode knowledge about the data and much of the success of SVM applications depends on identifying appropriate kernel functions [2].

This is not straightforward because a researcher cannot directly take his favourite affinity measure and build a kernel function from it. In order for the quadratic optimisation process that constructs the maximum margin classifier to converge to a global optimum it is necessary for the kernel matrix to be positive semi-definite.

So there is a variety of affinity measures that are very effective for specific types of data but will not readily produce a PSD kernel matrix, e.g.

- Kullback-Leibler divergence for comparing data distributions [3],
- Compression-based similarity for text [4] or gene sequences [5],
- Earth-mover distance (EMD) for comparing images[6] (also knows as the Mallows distance [7]).

In this paper we are concerned with ways to modify the EMD so that it will produce a kernel matrix that is PSD. We set out to answer two research questions:

1. *What is a good way to produce a kernel matrix based on the EMD without loosing the capability of the EMD to capture similarity between images?*
2. *When an SVM is built with this modified measure is it better at classification than a k-NN classifier based on the original measure?*

We have identified three ways in which the EMD can be *kernelized*.

- Generalised Radial Basis Function kernels [8, 9]
- Diagonal shift: shift the spectrum (eigenvalues) of the kernel matrix so that it is PSD.
- Empirical Kernel Map: $\mathbf{K}' = \mathbf{K}\mathbf{K}^T$

These techniques are explained in more detail in section 2.4. The paper is organized as follows: section 2 describes methods used in our experiments including classifier algorithms in subsection 2.1, colour signature extraction method in subsection 2.2, Earth Mover's Distance in Subsection 2.3 and methods of EMD kernelizing in subsection 2.4. Section 3 reports the evaluation results and finally section 4 contains our conclusions.

2 Methods

2.1 Classifiers used in the experiments

***k*-Nearest Neighbour** The intuition underlying Nearest Neighbour classification is quite straightforward, examples are classified based on the class of their nearest neighbours. It is often useful to take more than one neighbour into account so the technique is more commonly referred to as k-Nearest Neighbour (k-NN) Classification where k nearest neighbours are used in determining the class. $k-$NN classification has two stages; the first is the determination of the nearest neighbours and the second is the determination of the class using those neighbours.

Let us assume that we have a training dataset D made up of $(\mathbf{x}_i)_{i\in[1,|D|]}$ training samples. The examples are described by a set of features F and any numeric features have been normalised to the range [0,1]. Each training example is labelled with a class label $y_j \in Y$. Our objective is to classify an unknown

example $\mathbf{q}$. For each $\mathbf{x}_i \in D$ we can calculate the distance between $\mathbf{q}$ and $\mathbf{x}_i$ as follows:

$$d(\mathbf{q}, \mathbf{x}_i) = \sum_{f \in F} w_f \delta(\mathbf{q}_f, \mathbf{x}_{if}) \tag{1}$$

There are a large range of possibilities for this distance metric; a basic version for continuous and discrete attributes would be:

$$\delta(\mathbf{q}_f, \mathbf{x}_{if}) = \begin{cases} 0 & f \text{ discrete and } \mathbf{q}_f = \mathbf{x}_{if} \\ 1 & f \text{ discrete and } \mathbf{q}_f \neq \mathbf{x}_{if} \\ |\mathbf{q}_f - \mathbf{x}_{if}| & f \text{ continuous} \end{cases} \tag{2}$$

The k nearest neighbours are selected based on this distance metric. Then there are a variety of ways in which the k nearest neighbours can be used to determine the class of $\mathbf{q}$. The most straightforward approach is to assign the majority class among the nearest neighbours to the query.

It will often make sense to assign more weight to the nearer neighbours in deciding the class of the query. A fairly general technique to achieve this is distance weighted voting where the neighbours get to vote on the class of the query case with votes weighted by the inverse of their distance to the query.

$$Vote(y_j) = \sum_{c=1}^{k} \frac{1}{d(\mathbf{q}, \mathbf{x}_c)^n} 1(y_j, y_c) \tag{3}$$

Thus the vote assigned to class y_j by neighbour $\mathbf{x}_c$ is 1 divided by the distance to that neighbour, i.e. $1(y_j, y_c)$ returns 1 if the class labels match and 0 otherwise. In equation 3 n would normally be 1 but values greater than 1 can be used to further reduce the influence of more distant neighbours.

Another approach to voting is based on Shepard's work and uses an exponential function rather than inverse distance, i.e:

$$Vote(y_j) = \sum_{c=1}^{k} e^{-\frac{d(\mathbf{q}, \mathbf{x}_c)}{h}} 1(y_j, y_c) \tag{4}$$

We used the latter approach as it produces better results according to [10].

Support Vector Machines A Support Vector Machine (SVM) classifier is based on the linear maximum margin classifier that finds the solution to the optimisation problem of finding a hyperplane, that produces the largest margin between two classes in the feature space of samples. Support vectors are the samples closest to the hyperplane, that define it.

As the original feature space is not alway linearly separable it can be projected into an artificially constructed space of higher dimensionality. The solution to the optimisation problem does not require to transform the features of the samples to do that, but only uses a kernel function - a similarity measure between the samples, that is a dot product in the constructed space. In this sense the form of the kernel function fully defines the constructed feature space.

The 1-norm soft-margin SVM [1] used in this evaluation allows some cases from the training set to be within the margin or even on the "wrong" side of the hyperplane. Such an approach makes the SVM less sensitive to noise. An error cost parameter defines the degree to which such errors are allowed.

2.2 Colour signatures

One of the basic global features of the image is the distribution of colour within the image. Two popular quantisations for working with colour distributions are colour histograms and colour signatures. The former is constructed is based on the separation of the colour space of an image into bins and calculating for each bin a ratio of all pixels of the image that occur in that bin. The result is a three-dimensional array, each element h_{ijk} represents the corresponding bin of the colour space.

In this work we used the alternative popular representation – colour signatures. Unlike histograms, a colour signature does not fix the colour resolution of the model. The colour signature of an image is constructed by applying a clustering algorithm to all pixels of the image, and recording centers of each cluster and the ratio of the pixels that belong to that cluster. The resulting signature is a set of vectors as described in the next section (2.3).

We used the k-Means algorithm to perform clustering, that requires k to be defined prior to partitioning. This means that k must either be a fixed parameter for the experiment, i.e. the colour signatures of all images are fixed to the same length, or several signatures of different lengths must be extracted for each image and then some validation method must be applied to them to select the most representative one. In this work we evaluated both approaches using the Silhouette validation index [11] to select k.

2.3 Earth Mover's Distance

The Earth Mover's Distance (EMD) is a a transformation-based distance for image data. It overcomes many of the problems that arise from the arbitrariness of binning when using histograms. As the name implies, the distance is based on an assessment of the amount of effort required to convert one image to another based on the analogy of transporting *mass* from one distribution to another (see Figure 1).

In their analysis of the EMD Rubner et al. [6] argue that a measure based on the notion of a *signature* is better than one based on a histogram. A signature $\{\mathbf{s}_j = \mathbf{m}_j, w_{\mathbf{m}_j}\}$ is a set of j clusters where $\mathbf{m}_j$ is a vector describing the mode of cluster j and $w_{\mathbf{m}_j}$ is the fraction of pixels falling into that cluster. Thus a signature is a generalisation of the notion of a histogram where boundaries and the number of partitions are not set in advance; instead j should be 'appropriate' to the complexity of the image [6].

For two images described by signatures $S = \{\mathbf{m}_j, w_{\mathbf{m}_j}\}_{j=1}^{n}$ and $Q = \{\mathbf{p}_k, w_{\mathbf{p}_k}\}_{k=1}^{r}$ we are interested in the work required to transfer from one to the other for a given flow pattern $\mathbf{F}$:

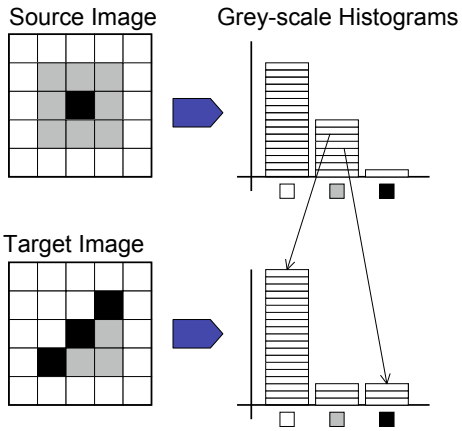

Fig. 1. An example of the EMD effort required to transform one image to another with images represented as histograms.

$$WORK(S, Q, \mathbf{F}) = \sum_{j=1}^{n} \sum_{k=1}^{r} d_{jk} f_{jk} \tag{5}$$

where d_{jk} is the distance between clusters $\mathbf{m}_j$ and $\mathbf{p}_k$ and f_{jk} is the flow between $\mathbf{m}_j$ and $\mathbf{p}_k$ that minimises overall cost. An example of this in a 2D colour space is shown in Figure 2. Once the transportation problem of identifying the flow that minimises effort is solved (using dynamic programming) the EMD is defined to be:

$$EMD(S, Q) = \frac{\sum_{j=1}^{n} \sum_{k=1}^{r} d_{jk} f_{jk}}{\sum_{j=1}^{n} \sum_{k=1}^{r} f_{jk}} \tag{6}$$

Efficient algorithms for the EMD are described in [6] however this measure is expensive to compute with cost increasing more than linearly with the number of clusters. Nevertheless it is an effective measure for capturing similarity between images.

2.4 Kernelizing the EMD

In this section we describe three techniques for *kernelizing* the EMD. The first technique is the generalised RBF kernel which uses the fact that EMD is a true metric. This technique makes the following transformation:

$$K_{exp} = e^{-\frac{EMD(S,Q)}{h}} \tag{7}$$

This produces a kernel matrix that is related to the Gaussian Kernel. In fact this matrix will be PSD if the exponent is any distance in the input space [8]. In calculating the EMD the signatures are normalised so that $0 \leq EMD(S, Q) \leq 1$.

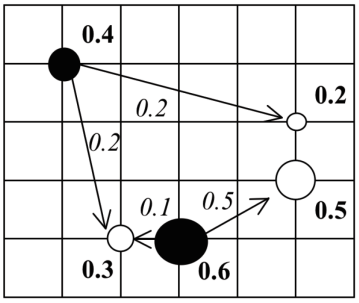

Fig. 2. An example of the EMD effort required to transform one image to another with images represented as signatures: the source image is represented by two clusters (black circles) and the target image by three clusters.

The second technique is based on the fact that it is straightforward to manipulate the eigenvalues of a matrix – adding $\sigma\mathbf{I}$ increases the eigenvalues by σ. Thus, in the following, $\mathbf{K}_{ds}$ will be PSD if λ_N is the most negative eigenvalue of $\mathbf{K}$ – this is illustrated graphically in Figure 3.

$$\mathbf{K}_{ds} = \mathbf{K} - \lambda_N \mathbf{I} \tag{8}$$

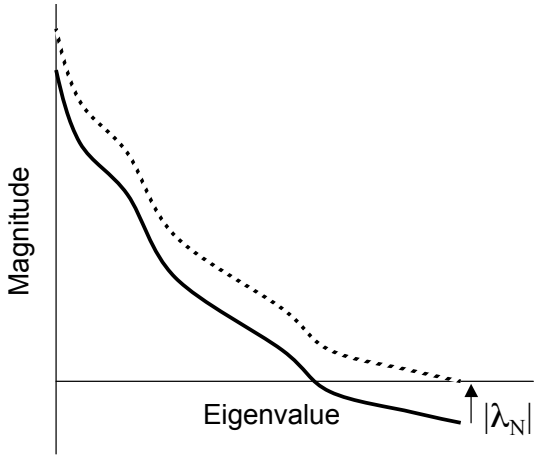

Fig. 3. A kernel matrix that is not PSD can be converted to a PSD matrix by shifting the eigenvalues so that the smallest eigenvalue is 0.

The third technique is based on the fact that every matrix multiplied by itself transposed gives a positive semi-definite matrix as a result (The Empirical Kernel Map). Because EMD metric is symmetric the transposition is not necessary:

$$\mathbf{K}_{sq} = \mathbf{K}\mathbf{K} \tag{9}$$

While the result of each of these transformations is a PSD matrix, the big question is whether the useful proximity information is distorted or lost in the process.

3 Evaluation

3.1 Dataset

The dateset used in our experiments consists of 800 low resolution images selected from a stock of Flickr (www.flickr.com) images published under Creative Commons Lincense by different contributors. 50% of the images are manually labelled "Mountains", while the other 50% comprises images labelled "France" with any mountain-related images excluded.

3.2 Classifier-specific parameter selection

A series of experiments were carried out using a fixed signature length of 10 for each of the classifiers to establish appropriate values for the classifier specific parameters, i.e. the error cost for the SVM and the number of nearest neighbours k for the k-NN.

In the k-NN case values of k between 2 and 100 were tried, and it was found that $k = 8$ is the most appropriate choice.

The optimal error cost estimation experiments were carried out for all three choices of kernel matrices in the SVM case and it was found that the optimal error costs are 5, 2 and 10 for SVMs based on exponential kernel, diagonal shift and squared proximity matrix respectively.

3.3 Comparison of algorithms for fixed length signatures

The k-Means algorithm used in our experiments to obtain colour signatures could produce any requested number of clusters (i.e. signature length), this affects the performance of both SVM and k-NN classifier.

We carried out 5-fold cross validation experiments for fixed signature lengths between 2 and 20. Figure 4 depicts the accuracy of four classifiers as a function of signature length.

The results suggest that the accuracy of two SVM-based classifiers in our experiments are generally higher and less dependant on a signature length than the accuracy of k-NN and squared matrix based SVM. The latter classifier's behaviour is erratic, producing reasonable accuracy only for a few short signature

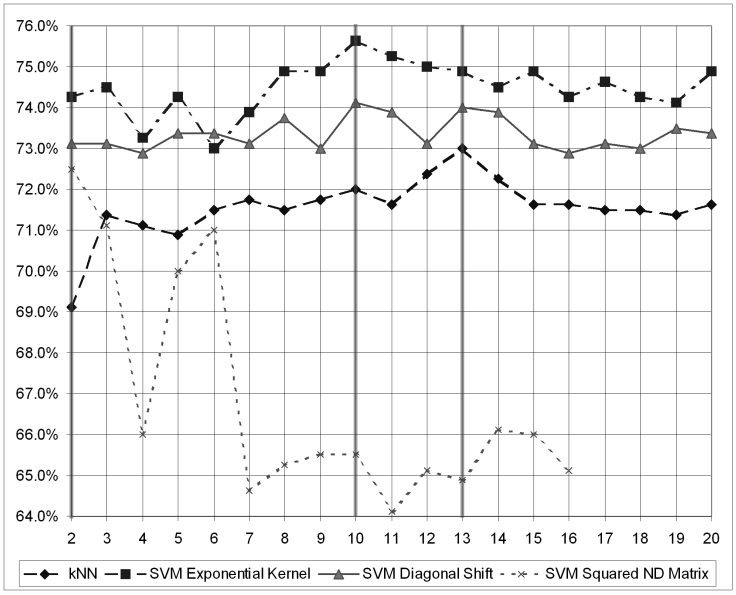

Fig. 4. The accuracy of kNN and three SVM-based classifiers as a function of signature length

lengths. The exponential EMD kernel classifier performs better than other three for most of the signature lengths.

A statistical pairwise comparison of the winning classifier against the three others has been carried out for the signature lengths of 2, 10 and 13. These signature lengths are highlighted on the Figure 4. McNemar's statistical test [12] has been employed for that purpose, i.e. the χ^2 statistic is calculated as follows:

$$\chi^2 = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}} \tag{10}$$

where n_{01} is a number of cases misclassified by first classifier and classified correctly by the second, and n_{10} is a number of cases misclassified by the second and classified correctly by the first. If the null hypothesis, which states that one classifier performs no better than the other, is correct then the probability of χ^2 being greater than $\chi^2_{1,0.95} = 3.841459$ is lower than 5%. So if the calculated statistic (χ^2) is greater than that, we may reject the null hypothesis, assuming that one classifier performs better than the other.

The results of this statistical comparison is provided in Table 1, where kNN designates the k-NN classifier, and SVM_{exp}, SVM_{ds} and SVM_{sq} designate an SVM classifier using an exponential kernel, diagonal shift and squared similarity matrix respectively.

Table 1. Pairwise comparison of a winning classifier vs its rivals for several signature lengths

Signature SZ	$Classifier_1$	$Accuracy_1$	$Classifier_2$	$Accuracy_2$	Significance
2	SVM_{exp}	74.3%	kNN	69.1%	Significant
2	SVM_{exp}	74.3%	SVM_{ds}	73.1%	Insignificant
2	SVM_{exp}	74.3%	SVM_{sq}	72.5%	Insignificant
10	SVM_{exp}	75.0%	kNN	72.0%	Significant
10	SVM_{exp}	75.0%	SVM_{ds}	74.1%	Significant
10	SVM_{exp}	75.0%	SVM_{sq}	65.5%	Significant
13	SVM_{exp}	74.9%	kNN	73.0%	Insignificant
13	SVM_{exp}	74.9%	SVM_{ds}	74.0%	Insignificant
13	SVM_{exp}	74.9%	SVM_{sq}	64.9%	Significant

3.4 The effect of the signature selection method

The alternative to a fixed colour signature size is to employ some method of signature selection on a sample by sample basis. In this work we used the Silhouette method [11] to select the the best signature length, i.e. to select the best value of k in the k-Means clustering that produces the signature.

Tables 2 and 3 present the results of the experiments with variable signature sizes for SVM and kNN respectively. The evaluation shows results with k selected from four different ranges: {2–5}, {5–10}, {10–20}, {2–20}. The first row of results shows the accuracy resulting from variable signature length selection using the Silhouette validation, the second row shows results with random selection and the last two rows show the best and worst results from fixed signature lengths in the interval in question.

Table 2. Assessment of SVM performance for variable signature lengths using the Silhouette validation index.

Signature lengths:	2 .. 5	5 .. 10	10 .. 20	2 .. 20
Silhouette	73.9%	72.8%	75.4%	73.9%
Random	72.5%	74.8%	74.6%	73.8%
Best of fixed	74.5%	75.6%	75.6%	75.6%
Worst of fixed	73.3%	73.0%	74.1%	73.0%

In both cases a classifier based on Silhouette selection of signature was generally outperformed by the corresponding best fixed signature length classifier with the exception of the interval {10–20}. Actually for intervals {5–10} and {2–20} in the SVM and kNN cases even the random selection of the signature outperforms Silhouette selection.

Such generally poor performance of the Silhouette selection could be explained by the slight bias of this validation method towards the smaller number of clusters – this can be detected by running the Silhouette algorithm accross

Table 3. Assessment of kNN performance for variable signature lengths using the Silhouette validation index.

Signature lengths:	2 .. 5	5 .. 10	10 .. 20	2 .. 20
Silhouette	69.9%	72.1%	72.5%	70.4%
Random	69.1%	71.9%	71.8%	70.8%
Best of fixed	71.4%	72.0%	73.0%	73.0%
Worst of fixed	69.1%	71.5%	71.4%	69.1%

random partitionings with different number of clusters. This bias is especially strong in the case of small number of clusters, for which the accuracy of the classifier could be worse.

The other possible explanation is that the EMD distance between two quite similar distributions represented by different number of clusters, could be substantially higher, than if they are represented by the same number of clusters. That is because the centers of the most similar clusters between the distributions in the latter case could be much closer to each other, than in the former case. So, the EMD metrics will carry less information about the actual dissimilarity of the distributions.

3.5 The effect of the diagonal shift

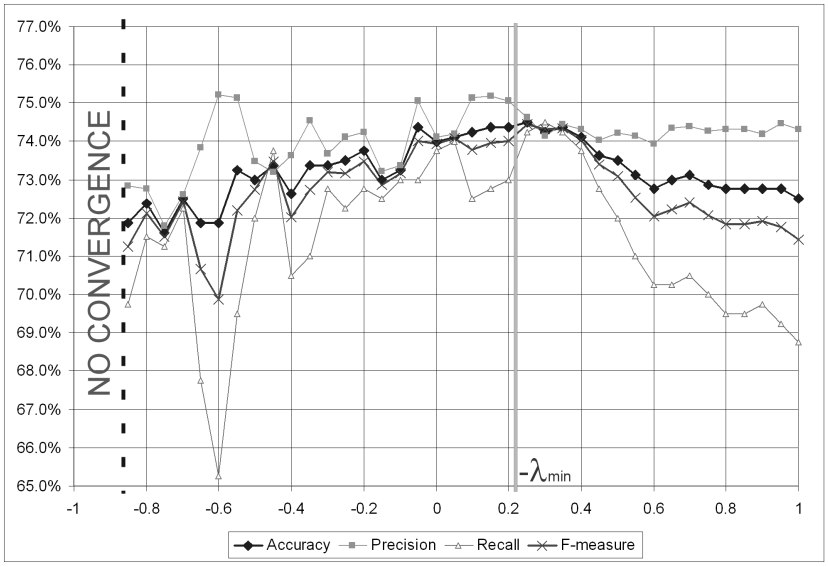

Fig. 5. The accuracy, precision, recall and f-score of SVM classifier for different values of the diagonal shift.

In section 2.4 we outlined how a matrix can be made PSD by shifting the spectrum of the matrix so that the smallest eigenvalue is zero. In this section we evaluate the effect of shifting the spectrum of the matrix in this way across a range of values. The results are shown on Figure 5.

The evaluation suggests that the maximum accuracy is achieved for the diagonal shift equal to the negated smallest eigenvalue of the proximity matrix – i.e. by setting the smallest eigenvalue to zero. The accuracy in the region of this value is higher than accuracies for both lower and higher values of the shift. It may be that below this value the SVM is not converging to a global optimum because the optimisation problem is no longer convex. Then above this value it may be that the kernel matrix is diagonal dominated [13] resulting in poorer performance. That suggests that the diagonal shift can be applied even for a PSD matrix to maximize the accuracy.

It is also interesting to note that the classifier continues to converge when the shift value is reduced, that makes the matrix non-PSD, although the precision and recall charts start to oscilate significantly. Finally for the values of the shift below -0.85, the classifier stops converging.

4 Conclusions

The main conclusion of this paper is that *kernelizing* the EMD so that it can be used in an SVM is worth while. A valid kernel matrix can be produced using an exponential RBF transformation or by using a diagonal shift. The SVM using the EMD is more effective that a k-NN classifier. We found that the alternative Empirical Kernel Map strategy was less effective showing performance comparable to the other two strategies only for short signatures (2 or 3). The experiments show a significant difference between SVM and k-NN in this respect at least for signature lengths that provide maximal accuracy in the SVM case.

The use of variable length signatures selected using the Silhouettte method does not give any advantage over the fixed signature length approach combined with cross-validation to select an appropriate signature length for the whole dataset. This question must be researched further with possible application of other validation techniques. Stability based validation [14] seems to be a good candidate.

Finally, it is evident from our experiments that the diagonal shift affects the accuracy of the classifier, with suspected maximum benefit at the point where such shift makes the smallest eigenvalue of the similarity matrix equal to zero. If further experiments confirm this, it may mean that the diagonal shift is not only a good technique in cases of non-PSD matrix, but also suits the purpose of increasing the accuracy of SVM that uses PSD matrices.

References

1. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines and Other Kernel-based learning Methods. Cambridge University Press (2000)

2. Noble, W.S.: Support vector machine applications in computational biology. In B. Schoelkopf, K.T., Vert, J.P., eds.: Kernel Methods in Computational Biology, MIT Press (2004) 71–92
3. Kullback, S., Leibler, R.A.: On information and sufficiency. Annals of Mathematical Statistics **22** (1951) 79–86
4. Li, M., Chen, X., Li, X., Ma, B., Vitányi, P.M.B.: The similarity metric. IEEE Transactions on Information Theory **50** (2004) 3250–3264
5. Chen, X., Kwong, S., Li, M.: A compression algorithm for DNA sequences and its applications in genome comparison. Proceedings of RECOMB **107** (2000)
6. Rubner, Y., Tomasi, C., Guibas, L.J.: The earth mover's distance as a metric for image retrieval. International Journal of Computer Vision **40** (2000) 99–121
7. Mallows, C.L.: A note on asymptotic joint normality. Annals of Mathematical Statistics **43** (1972) 508–515
8. Chapelle, O., Haffner, P., Vapnik, V.: SVMs for histogram-based image classification. IEEE Transactions on Neural Networks **10** (1999) 1055–1065
9. Zhang, J., Marszalek, M., Lazebnik, S., Schmid, C.: Local features and kernels for classification of texture and object categories: A comprehensive study. International Journal of Computer Vision **73** (2007) 213–238
10. Zavrel, J.: An empirical re-examination of weighted voting for k-nn (1997)
11. Rousseeuw, P.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics **20** (1987) 53–65
12. Dietterich, T.G.: Approximate statistical tests for comparing supervised classification learning algorithms. Neural Computation **10** (1998) 1895–1924
13. Greene, D., Cunningham, P.: Practical solutions to the problem of diagonal dominance in kernel document clustering. In Cohen, W.W., Moore, A., eds.: ICML, ACM (2006) 377–384
14. Lange, T., Roth, V., Braun, M.L., Buhmann, J.M.: Stability-based validation of clustering solutions. Neural Comput. **16** (2004) 1299–1323