ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 299-304

SVM Learning with the SH Inner Product

Peter Geibel, Brijnesh J. Jain, Fritz Wysotzki

Methods of Artificial Intelligence, Sekr. Fr 5–8, Faculty IV, TU Berlin,
Franklinstr. 28/29, D-10587 Berlin, Germany

Abstract. We apply support vector learning to attributed graphs where the kernel matrices are based on approximations of the Schur-Hadamard (SH) inner product. We present and discuss experimental results of different classifiers constructed by a SVM operating on positive semi-definite (psd) and non-psd kernel matrices.

1 Introduction

Support vector machines (SVM) [12] have proven to be widely applicable and successful in data classification. Given a set $\mathcal{X} = \{\boldsymbol{x_1}, \dots, \boldsymbol{x_M}\}$ of training objects with corresponding labels $\mathcal{Y} = \{y_1, \dots, y_M\} \subseteq \{+1, -1\}^M$, a SVM learns an optimal hyperplane to separate the training objects in a feature space by solving the constrained optimization problem

$$\begin{array}{lll} \min\limits_{\boldsymbol{w},\boldsymbol{\xi}} & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum\limits_{i=1}^{M}\xi_i & \\ \text{subject to} & y_i\left(\boldsymbol{w}^T\phi(\boldsymbol{x_i}) + b\right) \geq 1 - \xi_i & (1) \\ & \xi_i \geq 0,\ i = 1, \dots, M & \end{array}$$

Solving (1) is accomplished through minimizing the Lagrangian dual problem

$$\begin{array}{lll} \min\limits_{\alpha} & \frac{1}{2}\boldsymbol{\alpha}^T Q\boldsymbol{\alpha} - \sum\limits_{i=1}^{M}\alpha_i & \\ \text{subject to} & 0 \leq \alpha_i \leq C,\ i = 1, \dots, M & (2) \\ & \boldsymbol{y}^T\boldsymbol{\alpha} = 0 & \end{array}$$

where $Q = (q_{ij})$ with $q_{ij} = y_i y_j \phi(\boldsymbol{x_i})^T\phi(\boldsymbol{x_j})$. Here, $k(\boldsymbol{x_i}, \boldsymbol{x_j}) = \phi(\boldsymbol{x_i})^T\phi(\boldsymbol{x_j})$ is called the *kernel.* The kernel k gives rise to a psd *kernel matrix* $K = (k_{ij})$ with $k_{ij} = k(\boldsymbol{x_i}, \boldsymbol{x_j})$.

So far, most research on kernel methods has focused on learning from attribute value data. The investigation on kernel methods for attributed graphs, however, has recently started [2] and is still widely unexplored, though graphs are a more adequate representation of patterns in structured domains than feature vectors.

In this paper we propose the *Schur-Hadamard inner product* for support vector learning of attributed graphs. The SH inner product shares some properties of a kernel, but is in general not a kernel. In experiments we investigate the applicability of the SH inner product for support vector learning of graphs.

This article is structured as follows. Section 2 describes the SH inner product. Section 3 presents experimental results. Finally, Section 4 concludes.

2 The Schur-Hadamard Inner Product

Let $\mathcal{S}$ be a set. By $\mathcal{S}^{[2]}$ we denote the set of all ordered tuples $(i, j) \in \mathcal{S}^2$ with $i \neq j$. The set of all $n \times m$-matrices $A = (a_{ij})$ with entries a_{ij} from a set $\mathcal{S}$ is denoted by $\mathcal{M}_{n\times m}(\mathcal{S})$.

Let $\mathcal{A}$ be an inner product space over $\mathbb{R}$, for example $\mathcal{A} = \mathbb{R}^m$. An *attributed graph* is a tuple $X = (V, \mu)$ consisting of a finite set $V \neq \emptyset$ and a function $\mu : V^2 \to \mathcal{A}$. The elements of V are the *vertices* of the graph X and the pairs $(i, j) \in V^{[2]}$ with $\mu(i, j) \neq \mathbf{0}$ are its edges. The function μ is the *attribute function* of X. By $\mathcal{G} = \mathcal{G}_\mathcal{A}$ we denote the set of attributed graphs with attributes from $\mathcal{A}$. By $\mathcal{G}^n = \mathcal{G}^n_\mathcal{A}$ we denote the set of all attributed graphs with $m \leq n$ vertices. The vertex set of a graph X is referred to as $V(X)$, its edge set as $E(X)$, and its attribute function as μ_X. Let X be an attributed graph of order $|X| = |V(X)| = n$. The *(attributed) adjacency matrix* of X is a matrix $A(X) = (x_{ij}) \in \mathcal{M}_{n\times n}(\mathcal{A})$ with entries $x_{ij} = \mu_X(i, j)$.

A *permutation* acting on X is a bijection $\pi : V(X) \to V(X)$ from $V(X)$ onto itself. The image graph of a permutation π acting on X is denoted by X^π. The set $\mathcal{S}_X$ of all permutations acting on X is called the *symmetric group* of X. A permutation π acting on X corresponds to a relabeling of X and thus yields a reordering of its adjacency matrix. In general we have $A(X) \neq A(X^\pi)$.

For purely technical reasons we align graphs of different order to graphs of equal order by inserting additional nodes and edges into the smaller graphs that are all labeled with $\mathbf{0}$. Let $A, B \in \mathcal{M}_{n\times n}(\mathcal{A})$. The inner product $\langle\ ,\ \rangle$ associated with $\mathcal{A}^{n\cdot n}$ induces an inner product on $\mathcal{M}_{n\times n}(\mathcal{A})$ by $\langle A, B\rangle = \langle \boldsymbol{v}(A), \boldsymbol{v}(B)\rangle$, where $\boldsymbol{v}(A), \boldsymbol{v}(B) \in \mathcal{A}^{n\cdot n}$ are the vectors obtained by concatenating the rows of A and B, respectively.

The concept of a Schur-Hadamard inner product of graphs can be regarded as a degenerated counterpart of the concept of an inner product defined on vector spaces. It is defined by

$$\sigma : \mathcal{G}^n \times \mathcal{G}^n \to \mathbb{R}, \quad (X, Y) \mapsto \max_{\pi \in \mathcal{S}_X} \langle A(X^\pi), A(Y)\rangle .$$

Despite its name the SH inner product is in fact not an inner product, since it is not bilinear. Nevertheless, the Euclidean norm induced by the SH inner product is a metric. In addition the SH inner product is symmetric, positive, and the Cauchy-Schwarz inequality holds [5]. The SH inner product, however, shares some properties of a kernel, but it is in general not a kernel as can be shown by counterexamples. Under certain restrictions, it is possible to show that the SH inner product is a kernel.

Regardless whether the SH inner product σ is a kernel or not, its computation is an NP-complete problem [7]. Thus for large datasets of graphs computing the pairwise similarities might be intractable in a practical setting. One way around is to resort to heuristics which return approximate solutions within an acceptable time limit. But this may yield a non-psd kernel matrix.

Occasionally, non-psd matrices K are applied to (2). Examples include kernel matrices induced by the well known sigmoid kernel [12] or the tangent distance kernel [3]. If K is indefinite the primal-dual relationship does not exist. Thus, it is not clear what kind of classification problem we are solving. In addition, non-psd kernels may cause difficulties in solving (2). Surprisingly, non-psd matrices arising from kernels similar to the sigmoid kernel or tangent distance kernel have been applied successfully in several practical cases. According to [3] the experimental results show that the class of 'kernels' which produce accurate results is not restricted to *conditionally positive definite* (cpsd) *kernels.*

Nevertheless, theoretically sound solutions of support vector learning based on non-psd matrices have been proposed, for example the approach by Graepel et al. [4] where each pattern is described by the vector of proximities to all other patterns.

3 Experiments

In all experiments we used the SVMLight [8] embedded into a 10-fold cross validation. We first computed the matrix $K = (k_{ij})$ of pairwise similarities with respect to the normalized SH inner product. All SH inner products were approximated by solving the maximum weighted clique problem in an *inner product graph* with a special Hopfield network [6], [7]. Next we learned a support vector classifier using the following methods: (PPC) the pairwise proximities classifier as proposed by [4], (PPC-RBF) the PPC classifier with RBF kernel, (SH) support vector learning which directly operates on K, (SH-RBF$_N$) support vector learning using a RBF function on naive distances $d_{ij}^N = 1 - k_{ij}$, (SH-RBF$_E$) support vector learning using a RBF functions on Euclidean distances $d_{ij}^E = \sqrt{k_{ii}^2 - 2k_{ij} + k_{jj}^2} = \sqrt{2d_{ij}^N}$.

Synthetic Characters.
In our first experiment we investigated how a SVM can deal with both types of errors occuring in graph based representations, structural variations and noisy attributes.

We used synthetic data to emulate handwriting recognition of alphanumeric characters as it typically occurs in pen technology of small hand-held devices, for example PDAs. We do not apply additional on-line information. To simulate handwriting recognition as a classification task for structured objects, we

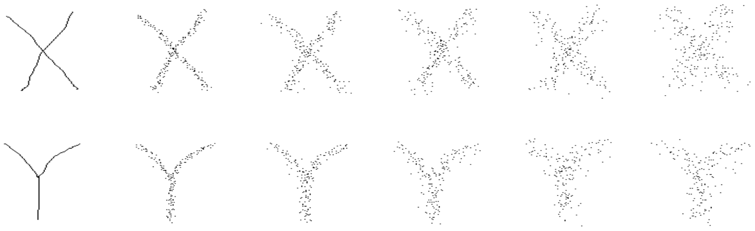


Figure 1: Images of handwritten characters $'X'$ and $'Y'$. Explanation see text.

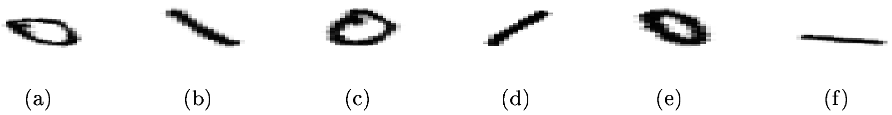
draw two handwritten characters models $\mathcal{C} = \{\text{'X', 'Y'}\}$ using an X windows interface. The contours of each image were discretized and expressed as a set of points in the 2D plane. Both model images are shown in the first column of Figure 1.

For each model character we generated 50 corrupted data characters as follows: First we randomly rotated the model image. Then to each point we added $N(0, \sigma)$ Gaussian noise with standard deviation $\sigma = 2, 4, 6, 8, 10$. Each point had 10% probability to be deleted. Columns 2-6 of Figure 1 show examples of corrupted data images for different standard deviations σ. For sake of presentation the graphics does not show the rotation of the images.

From each point set we randomly selected points such that the pairwise normalized distances between the chosen points is larger than a given threshold θ. We transformed this point set $\mathcal{P}$ to a fully connected attributed graph. The vertices $v(\boldsymbol{p})$ represent the points $\boldsymbol{p} \in \mathcal{P}$, and the edges represent an abstract line between two points. To each vertex $v(\boldsymbol{p})$ we assigned a three dimensional attribute vector $\boldsymbol{a} = (a_1, a_2, a_3)$ where a_1 is the normalized distance of the point $\boldsymbol{p}$ to the center of gravity of the corrupted image, a_2 is the mean distance of the normalized distances from point $\boldsymbol{p}$ to all other points $\boldsymbol{q} \in \mathcal{P}$, and a_3 is the variance of the normalized distances between $\boldsymbol{p}$ an all other points $\boldsymbol{q} \in \mathcal{P}$. To each edge connecting vertex $v(\boldsymbol{p})$ and $v(\boldsymbol{q})$ we assigned a two dimensional attribute vector $\boldsymbol{b} = (b_1, b_2)$. The first attribute is the normalized distance between $\boldsymbol{p}$ and $\boldsymbol{q}$. The second attribute measures the normalized distance between the center of gravity of the corrupted image and the abstract line passing through $\boldsymbol{p}$ and $\boldsymbol{q}$. Thus each graph is a representation of a character, which is invariant to rotation, translation, and scaling.

The average order of the graphs for each standard deviation σ is 40.0 with variance 23.7. The smallest graph was of order 32 and the largest of order 50. Thus besides noise in the attributes there is a strong structural variation.

Table 3(a) summarizes the mean predictive accuracy for different noise levels σ. The results show that SVM with kernels based on the SH inner product can cope with both, noisy attributes and structural variation of the data. As expected, the performance decreases with the noise level though the recognition rate is very good even for highly corrupted and randomly rotated characters.



(a) (b) (c) (d) (e) (f)

Figure 2: Examples of rotated images of handwritten digits.

Handwritten Digits
In the second of our experiments we applied the SVM to classify handwritten digits invariant to rotation. We used the training set $\mathcal{T}$ of the well-known MNIST database containing 60,000 gray-level images of handwritten digits [10]. We selected a subset of $\mathcal{T}$ consisting of 100 images of '0' and 100 images of '1'. We randomly rotated all 200 images as shown in Figure 2 for six examples. Each image was transformed into an attributed graph in a similar way as in the previous section to obtain a representation of the numbers '0' and '1', resp., which is invariant to rotation, translation, and scaling.

For the digits dataset we achieved an accuracy of 0.995 for PPC and PPC-RBF. The other variants performed slightly worse but were still better than 0.98.

Mutagenicity
The mutagenicity of a chemical compound is closely related to its cancerogenicity. A particular problem is to discover rules to predict mutagenicity in a database of nitro aromatic compounds (e.g. [11]). The mutagenesis dataset is usually considered in two subsets containing 188 and 42 examples respectively.

Each compound in the dataset is described by its atoms and their bonds. An atom is described by its element symbol, its type (e.g. aromatic) and a partial electrical charge. There also exist some attributes that describe the molecule as a whole. For our experiments we have only used the structural information.

The estimated predictive accuracies are shown in Table 3(b). Surprisingly, the naive SH-RBF$_N$ approach outperforms all other algorithms reported in the literature. Although the 'kernel matrix' was indefinite, the SVMLight terminated for the optimal parameter setting while non-termination occurred for other parameter settings. The results for the random walk kernel (RWK) and the description logic kernel (DLK) are taken from [9] and [1], respectively.

4 Conclusion

In this paper we have successfully applied support vector learning with kernels based on the SH inner product. We directly operated on non-psd matrices of approximated pairwise SH similarities and also considered different variants of the theoretical sound PPC approach. Though the latter approach works reasonably well, it is in general outperformed by the former approach. The results confirm that in practice the property of a kernel being cpsd is a sufficient

σ	2	4	6	8	10
PPC	1.0	1.0	0.94	0.92	0.71
PPC-RBF	1.0	1.0	0.96	0.94	0.77
SH	1.0	1.0	0.95	0.96	0.71
SH–RBF$_E$	1.0	1.0	0.95	0.96	0.80

(a) Synthetic handwritten letters

dataset	188	42
RWK	0.90	-
DLK	0.89	-
PPC	0.84	0.83
PPC-RBF	0.88	0.86
SH	0.85	0.86
SH-RBF$_N$	**0.92**	**0.90**
SH-RBF$_E$	0.90	0.86

(b) Mutagenesis dataset

Table 1: Estimated predictive accuracy.

but not necessary condition to learn an classifier with low error rate. Future work will focus on conditions when the SH inner product is a kernel.

References

[1] C. Cumby and D. Roth. On kernel methods for relational learning. In T. Fawcett and N. Mishra, editors, *Proceedings of the Twentieth International Conference on Machine Learning*, pages 107–115. AAAI Press, 2003.

[2] T. Gärtner. A survey of kernels for structured data. *SIGKDD Explorations*, 5(2):49–58, 2003.

[3] B. Haasdonk and D. Keysers. Tangent Distance Kernels for Support Vector Machines. In *ICPR 2002, International Conference on Pattern Recognition*, volume II, pages 864–868, 2002.

[4] T. Graepel, R. Herbrich, P. Bollmann-Sdorra, and K. Obermayer. Classification on pairwise proximity data. In *Advances in Neural Information Processing Systems 11*, pages 438–444, 1999.

[5] B.J. Jain and F. Wysotzki. Perceptron Learning in the Domain of Graphs. In *Proc. of the International Joint Conference on Neural Networks, IJCNN 2003.*

[6] B.J. Jain and F. Wysotzki. The Maximum ω-Clique Problem and the Hopfield ω-clique Model. Submitted for publication.

[7] B.J. Jain and F. Wysotzki. The Schur-Hadamard Inner Product and Maximum ω-Cliques in an Inner Product Graph. Submitted for publication.

[8] T. Joachims. *Learning to Classify Text using Support Vector Machines: Machines, Theory and Algorithms.* Kluwer Academic Publishers, Boston, 2002.

[9] H. Kashima, K. Tsuda, and A. Inokuchi. Marginalized kernels between labeled graphs. In T. Fawcett and N. Mishra, editors, *Proceedings of the Twentieth International Conference on Machine Learning*, pages 321–328. AAAI Press, 2003.

[10] Y. LeCun. The MNIST Database of Handwritten Digits. NEC Research Institute, Princeton, NJ. URL = http://yann.lecun.com/exdb/mnist/, 2003.

[11] A. Srinivasan, S. Muggleton, M. J. E. Sternberg, and R. D. King. Theories for mutagenicity: A study in first-order and feature-based induction. *Artificial Intelligence*, 85(1,2):227–299, 1996.

[12] V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer, New York, 1995.