KMOD-A New Support Vector Machine Kernel With Moderate Decreasing for Pattern Recognition. Application to Digit Image Recognition.

N.E. Ayat [1,2] M. Cheriet [1] L. Remaki [1] C.Y. Suen [2]

[1] LIVIA, École de Technologie Supérieure
1100, rue Notre Dame Ouest
Montreal, H3C 1K3, Canada
ayat@livia.etsmtl.ca
cheriet@gpa.etsmtl.ca

[2] CENPARMI, Concordia University
1455 de Maisonneuve Blvd West
Montreal, H3G 1M8, Canada
suen@cenparmi.concordia.ca

Abstract

A new direction in machine learning area has emerged from Vapnik's theory in support vectors machine and its applications on pattern recognition. In this paper, we propose a new SVM kernel family (KMOD) with distinctive properties that allow better discrimination in the feature space. The experiments that we carry out show its effectiveness on synthetic and large-scale data. We found KMOD behaving better than RBF and Exponential RBF kernels on the two-spiral problem. In addition, a digit recognition task was processed using the proposed kernel. The results show, at least, comparable performances to state of the art kernels.

1. Introduction

A new direction in machine learning area emerged from Vapnik's theory on the Structural Risk Minimization principle [10], [11]. As a major effect, automatic selection of the optimal classifier capacity tailored on the given task problem became effective. In fact, it was shown that one could limit the generalization error if the ratio of the decision surface margin separating the classes by the diameter of the hyper-sphere including all the data points is maximized. The greater this value, the smaller is the upper bound on the generalization error whereby the machine prediction power is increased [3], [9]. Learning algorithms based on this paradigm brought the Support Vectors Machine theory and its efficient applicability to pattern recognition [11], [12]. Basically, the SV Machines operate a linear separation in an augmented space different from the original one by means of some defined kernels respecting Mercer's condition [5], [10], [9]. These kernels map the input vectors into a very high-dimensional space, possibly of infinite dimension, where linear separation is more likely. This process amounts to do a non-linear separation in the original input space. Hence, the complexity of the achieved boundaries depends on the nature and the properties of the used kernel. It is well established that the SVM classifier, may behave as an MLP if a tangent hyperbolic kernel is used, as an RBF network if a gaussian kernel is used or as a linear classifier if no kernel function is plugged to the model.

In this paper, we present a new SVM kernel for pattern recognition. Our motivation is twofold. First, we explain intuitively its behavior with respect to the duality between spatial similarity in the original space and the correlation in the augmented space. Second, we carry out some experimental tests that show its effectiveness on synthetic and real-life data. KMOD, Kernel with Moderate Decreasing, has two parameters that allow at once, penalizing largely the far apart input vectors, while maintaining the closeness information from vanishing. In particular, this prevents any information loss inside the SVM model through the kernel application. As a result, the accuracy of the classifier is increased. This additional precision would let the SVM deal better with sparse data. In section 2, we briefly review the related state of the art. In section 3, we yield a theoretical analysis about KMOD properties. In section 4, we report some experiments benchmarking our kernel. A comparison with other kernels' performances is made. In section 5, we present our SVM based digit recognition system and some of the results we got with. Finally a conclusion ends our paper.

2. Review of SV Machine

The support vector machine is a classifier based on the structural risk minimization which goal is to find the optimum decision region. Let us have a data set $\{x_i, y_i\}, i = 1, \ldots, l$, where $y_i \in \{-1, 1\}$ and $x_i \in \Re^d$ where x is a data sample and y its label. Also, let us define a linear decision surface by the equation: *f(x)=wx+b=0.* The original formulation of support vector machine algorithm seeks a linear decision surface maximizing the margin between positive and negative examples. This may be achieved through a minimization of $||w||^2$ [7]. This yields:

$$w = \sum_i \alpha_i y_i x_i \tag{1}$$

$$\sum_i \alpha_i y_i = 0 \tag{2}$$

where the parameters α_i are the solution of the following quadratic optimization problem to be maximized:

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j \tag{3}$$

The data subset examples which corresponding α_i values are different of zero are called support vectors.

There are a number of methods for solving quadratic optimization problems as of L_D. In this work we used a special strategy based on successive shrinking [6]. Basically, it aims to iteratively decompose the optimization problem on a working set of size q for which the corresponding α_i variables are to be tuned. The remaining variables are fixed at their current values. An optimization of the working set sub-problem is further done until the termination criterion is met.However, in real-life classification problems, the algorithm as stated above is unable to achieve perfect separation between two classes, especially in case of noisy data. Cortes et al. in [4] slightly modified the model by adding a heuristic that accounts for accepting misclassified examples and penalizing them inside the model cost function in such a way their distances from the boundaries are minimized. Mathematically, this does not imply any major modification except that α_i should satisfy an upper bound limit: $0 \leq \alpha_i \leq C$ where C is a penalization parameter. An infinite value of C yields a model that seeks a well separated data. Moreover, Boser et al. in [2] added an important feature that enables these machines to produce complex nonlinear boundaries inside the original space. Their technique consists of projecting the data into higher order spaces, possibly of infinite dimension, through a mapping function ϕ and separating the patterns there. This function, must however, keep the inner product formulation inside the SVM model (equation 3) useful. The kernel trick keeps the SVM model still solvable, i.e. we do not need to know the explicit analytical form of functions ϕ themselves. Only the expression of their pairwise inner product $K(u,v) = \phi(u).\phi(v)$ in the augmented space must be defined [2], [9]. We report in table 1 some of the classical SVM kernels.

Kernels	Formula
Linear	$K(u,v) = u.v$
Sigmoid	$K(u,v) = \tanh(a\mathbf{u.v} + b)$
Polynomial	$K(u,v) = (1 + u.v)^d$
RBF	$K(u,v) = \exp(-a\|u - v\|^2)$
Exponential RBF	$K(u,v) = \exp(-a\|u - v\|)$

Table 1. Common kernels.

3. Our SVM kernel: theoretical analysis and intuitive idea

In general, the function that imbeds the original space into the augmented feature space is unknown. The existence of such function is however assured by Mercer's theorem [5]. The effect of such function is confined within the constructed kernel, which must express a dot product in the feature space. Moreover, all used kernels in the literature are either dot product functions $(k(x,y) = k(x.y))$ or distance functions $(k(x,y) = k(||x-y||))$. By adopting the latter formulation, knowing an estimation of the Euclidean distance between two points in the original space, we find how much they are correlated in the augmented space. The following questions however arise: What could be the best criterion for constructing such a kernel? Is the kernel spatial behavior of any importance?

In most commonly distance based kernels (eg. RBF), points very close to each other are strongly correlated whereas points far apart have uncorrelated images in the augmented space. Our concern is to force the images of the original points to be linearly separable in the augmented space. In order to get such a behavior, a kernel must turn very close points from the original space into weakly correlated elements (as weak as possible) while still maintaining the closeness information from vanishing. To achieve this tradeoff, we need the following couple of features: a quick decrease in the neighborhood of zero and a moderate decrease toward infinity. The RBF kernel may satisfy correctly the first requirement but not the second, whereas the exponential RBF does not respond correctly for both of the requirements (Figure 1). Alternatively, we propose KMOD whose analytic expression is as:

$$KMOD(x,y) = K[\exp(\frac{\gamma}{||x-y||^2 + \sigma^2}) - 1] \tag{4}$$

Where K is a normalization constant; γ and σ are two parameters controlling respectively the decreasing speed around zero and the width of the kernel. This kernel was derived from the author's work [8]; where the formula was modified such a way it ensures necessary conditions to be a Mercer kernel.

Remark that the second property of our kernel amounts to

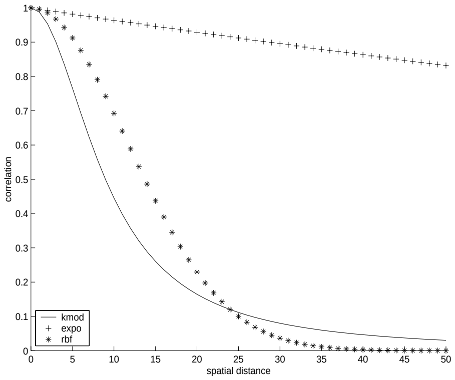

Figure 1. Correlation in feature space versus spatial distance in input space.

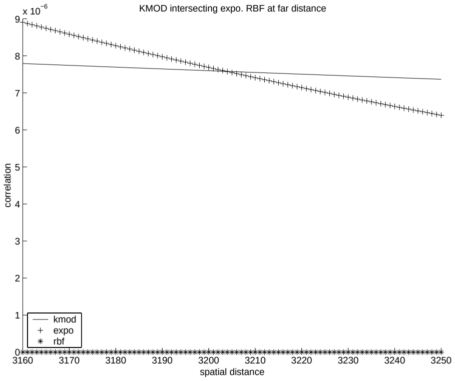

Figure 2. KMOD preserving the far points closeness information.

consider more closeness information during the SVM optimization process, whereas RBF kernel is penalizing quickly intermediate neighborhood and getting its correlation almost zero. Figure 2 shows that in far neighborhood, the RBF kernel yields almost zero value. The KMOD kernel however is decreasing moderately and intersecting the Exponential RBF function at an arbitrary point. Although the profile of KMOD changes with its parameters, it still preserves the aforementioned tradeoff.

4. Benchmarking its behavior

One typical test for pattern recognition is the two-spiral problem which goal is to separate two highly interrelated spirals. It is a difficult classification task that good classifiers must deal with successfully. Through this benchmark, some insight about the discrimination power and the compactness of KMOD is being established. The number of support vectors among the training data characterizes the compactness of a SVM. We used a soft margin model for the SVM [4]. Our objective was to test KMOD versus RBF and exponential RBF, two popular distance based kernels. First, we experiment on a one-point thickness spirals made of 96 points. The parameters of the kernels have been tuned in such a way a good separation is ensured while the number of support vectors is minimized. We tried out the following sets of values for σ and γ respectively: 0.1, 0.5, 1.0, 3.0, 5.0, 10.0, 30.0 and 0.1, 0.5, 1.0, 2.0, 3.0. We report in figure 3 the best KMOD resulting boundary[1]. The latter is perfectly separating the spirals. However, surprisingly, only 76% of the points are support vectors, even if all of them might be potential support vectors. This is a really interesting behavior that improves the compactness feature aforementioned. RBF fails in separating the two patterns. In figure 4, we plot its best resulting boundary. Table 2 shows the corresponding margin values for KMOD, RBF and Exponential RBF. These values would not be comparable unless we assume that the given data set has unbounded support (very sparse data). This case implies that there would be at least one couple of points from the original space, which images are orthogonal in the augmented space. Then, one may afford enclosing hyper-spheres of a diameter equal to $\sqrt{2}$ for all the kernels (with the assumption that all kernels are normalized to unity at zero). Thus, a comparison of the kernels generalization abilities through the VC dimension upper bound amounts to compare the corresponding kernels' margin values [10]. Since KMOD has the largest margin value, it would expect the best generalization performance (table 3).

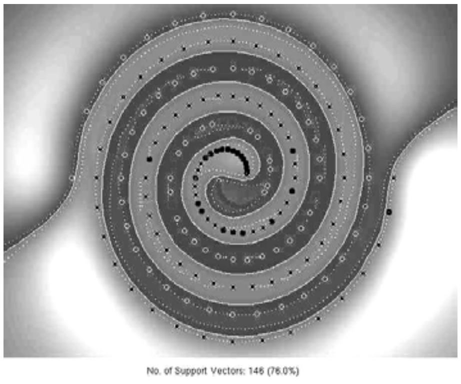

Figure 3. Boundaries with KMOD kernel.

As a second part of our benchmark, we tested the kernels on a noisy case of the two-spiral problem with less severe non-linearity. A Gaussian noise was used to produce the

[1] The plots in this section were obtained using Svmtool at: http://www.isis.ecs.soton.ac.uk/resources/svminfo/

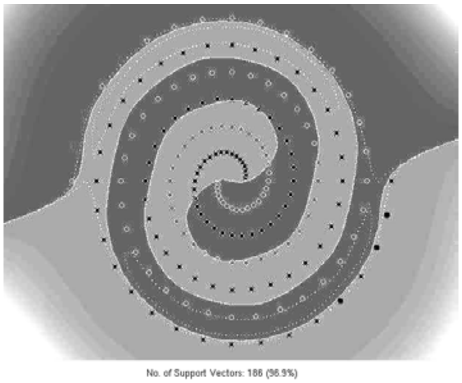

Figure 4. Boundaries with RBF kernel.

Kernel	KMOD	RBF	ERBF
Margin	0.239520	0.000002	0.039897
SV	146(76.0%)	196(96.9%)	192(100.0%)

Table 2. Results on the two-spiral problem.

spiral points. We plotted in figure 5 the solution boundaries for KMOD.

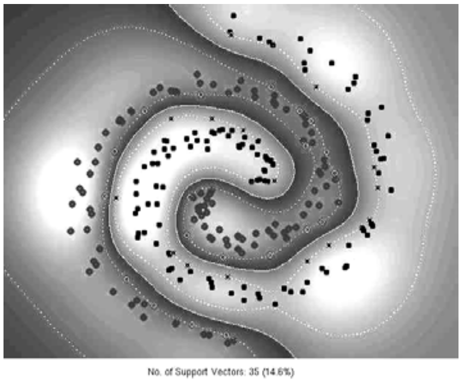

Figure 5. Boundaries with KMOD kernel on the noisy two-spiral problem.

In table 3, we report the best margin values for a minimal number of support vectors obtained with the kernels. KMOD uses 14.6% among the training data points as support vectors. As in the first benchmark, a better generalization, for KMOD, is expected through its margin value.

Kernel	KMOD	RBF	ERBF
Margin	2.563173	0.001399	0.094913
SV	35(14.6%)	21(8.8%)	97(40.4%)

Table 3. Results on the noisy two-spiral problem.

5. Recognition of handwritten digits

Support Vector Machines are binary classifiers, i.e. a seldom model is useful for two-class data only. However, multi-class classification problems (where one has $k \geq 2$) such as the digit recognition task could be solved using voting scheme methods based on a combination of many binary classifiers. One possible approach to solve a $k-class$ pattern recognition problem is to consider the problem as a collection of k binary classification problems. k classifiers can then be constructed, one for each class. The k^{th} classifier constructs a hyper-plane between class n and the $k-1$ other classes. A majority vote across the classifiers is then applied to classify a new example. Alternatively, $\frac{k(k-1)}{2}$ hyper-planes can be constructed, separating each class from each other and similarly some voting scheme applied. Our kernel model has been tested on NIST database images. For that purpose, we adopt the former multi-class recognition scheme. Namely, its implementation consists of building 10 different models, one for each class (Figure 6). Each of these models is a binary classifier that matches the specific model class data against the other nine classes data. This scheme was already used for solving multi-class data using linear classifiers and is commonly named as ”one against others strategy” [2]. We used a subset of 20,000 images from the hsf_123 part of NIST database for the training. Ten training processes were done. For classification, we consider the following simple scheme: $C_j = Arg\max_i(O_i)$; where C_j is the resulted class label and O_i is the i^{th} SVM output. The current test example will belong to the class for which the corresponding model output is maximal. No reject option was considered in this experiment. We used 10,000 images from the hsf_7 part of NIST database for testing. Prior to classification, from each digit image is extracted a set of values that well characterizes both of local and morphological features. Those are to be injected into the classifiers. For that purpose, we overlaid a sixteen zones mesh on each image. Inside each zone, 13 statistical features and 4 structural features are extracted. Eight of the former ones, are the freeman direction counts. The five remaining features capture statistics on the digit image edge curvatures. On the other hand, the morphological features we used are based on four kinds of regions [1]:

- A hole region to model concavities,
- an open region to model convexities,
- a mountain region,
- a valley region.

Four region counts are then computed inside each of the 16 zones grid. Both of the statistical and morphological features are normalized by dividing their counts through the zone's pixels count. This yields a whole feature set of 272 values that feed every SVM model. We consider only the best results of them. KMOD does perform better than polynomial and RBF kernels. It reaches 97.77% of recognition rate, whereas the best RBF model has 96.91% of precision. The polynomial SV machine however, does slightly better with 97.42% in the case of a polynomial kernel with degree 4. A polynomial kernel with degree 3 has a precision of 96.77%. It is worth to mention that the kernel parameters have been chosen in an empirical way in order to ensure good classification with respect to each kernel.

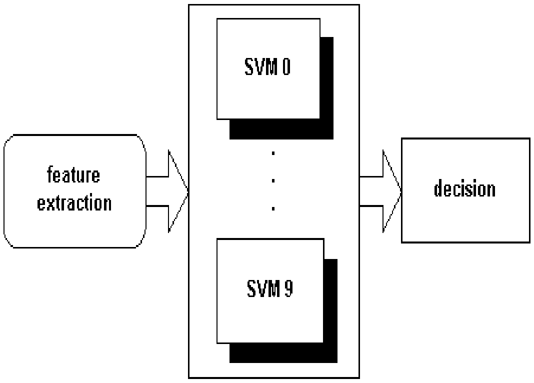

Figure 6. SVM digit recognition system.

6. Conclusion

Whilst it is possible to assume that the data fed into a SV Machine have bounded support, the sparseness of the data inside the original space can vary widely, depending on its distribution, the feature extraction method and the difficulty of the problem on hand. We believe that kernels preserving the whole data closeness information while still penalizing the far neighborhood are more reliable, especially in case of sparse data. KMOD is a new family of SVM kernels that allows such a behavior by ensuring at once a quick decreasing around zero and a moderate decreasing toward infinity. This tends to uncorrelate as much as possible very close points into the augmented space.

Experiments done on the two-spiral problem show the ability of KMOD in separating the patterns whereas RBF kenel fails. Moreover, we prove the effectiveness of KMOD in dealing with a large-scale problem such as the digit recognition task. For that, we built a digit recognition system gathering 10 SVM classifiers. KMOD gives the best results among polynomial and RBF kernels, which are proven to be good SVM kernels. An automatic optimization of KMOD parameters would let the SVM deal better with the spatial distribution of the data. This is an ongoing work that will be the object of future publication.

References

[1] N.E. Ayat, M. Cheriet, and C.Y. Suen. Un système neuro-flou pour la reconnaissance de montants numériques de chèques arabes. In *Actes du* 2^{eme} *Colloque International Francophone sur l'Écrit et le Document*, Lyon,France, July 2000. Presses Polytechniques et Universitaires Romandes.

[2] B. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In *Fifth Annual Workshop on Computational Learning Theory*, Pittsburg, 1992. ACM.

[3] C. Burges. A tutorial on support vector machine for pattern recognition. Kluwer Academic Publishers, 1998.

[4] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, Vol.20:273, 1995.

[5] Courant and R. Hilbert. *Methods of Mathematical Physics*. Interscience, 1953.

[6] T. Joachims. *Advances in Kernel Methods: Support Vector Learning*, chapter Making Large-Scale SVM Learning Practical. Chap.11. MIT Press, 1999.

[7] E. Osuna, R. Freund, and F. Girosi. Support vector machines: Training and applications. Technical report, MIT, 1997.

[8] L. Remaki and M. Cheriet. Kcs-new kernel family with compact support scale space. *IEEE Transactions On Image Processing*, 9(6):970, June 2000.

[9] B. Schlkopf, C. Burges, and A. Smola. *Advances in Kernel Methods - Support Vector Learning*, chapter Introduction to Support Vector Learning. Chap.1. MIT Press, 1999.

[10] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, NY, USA, 1995.

[11] V. Vapnik. *Statistical Learning Theory*. Wiley, NY, USA, 1998.

[12] V. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5), September 1999.