University of Szeged
Institute of Informatics

Various Kernel Methods with Applications

Summary of the PhD Thesis

by

Kornél Kovács

Advisor:

Dr. András Kocsor

Szeged
2008

Introduction

Model building is probably the most significant component of scientific thinking baring abstract similarities in all branches of science. We build models and examine how they function, we apply modifications to them as long as we reach our desired goal. Artificial intelligence is a special branch of science, which provides an endless horizon to the lovers of computational model building.

Machine learning forms a branch of artificial intelligence [25]. It comprises a set of algorithms that enable computers to learn. The concept of learning usually builds on two different approaches: inductive and deductive learning. My thesis follows the inductive learning approach, which retrieves rules or descriptive patterns from massive data sets. Presently, the main focus of machine learning is the complex task of automatic information extraction. Further important application possibilities lie in natural language processing, syntactic pattern recognition, the improvement of search engines, medical diagnosis, bio-informatics, speech recognition, object recognition, and enhancing computer games with humanlike intelligence - only to mention a few fields. Certain machine learning methods attempt to eliminate human resource from the process of data analysis, while others try to make human-machine interaction more human.

Considering the current level of technological advances, machine learning has become the most researched and most intensively developing field of artificial intelligence. The present thesis focuses on the examination and elaboration of the most novel approach in machine learning, namely that of kernel methods.

Kernel methods (KM) are a family of pattern recognition algorithms [26], whose most significant member is the Support Vector Machine (SVM) [31]. The general task of pattern recognition is to identify and examine representative correlations (e.g., clusters, classification decisions, etc.) on general data (e.g., vectors, documents, sequences, pictures, etc.). The KM approach was named after kernel functions, which work in a derived feature space where the real coordinates of patterns never have to be calculated. These methods only rely on the dot product of paired sample points, which are calculated implicitly by applying the kernel functions. Apart from SVM, there are a number of other algorithms belonging to the family of KM: e.g., various regression methods, Fisher's linear discriminant analysis (LDA) [9], principal components analysis (PCA) [10], canonical correlation analysis (CCA) [2], ridge regression [22], spectral clustering [21], and many others. Generally speaking, the majority of kernel methods lead to effectively soluble problems, convex optimisation or eigenproblems.

The Kernel Mapping Idea

Let X, y be the training data, where $X = (x_1, \ldots, x_n)$ are the input patterns ($x_j \in \mathbb{R}^d$) and $y \in \{-1, +1\}^n$ are the corresponding target labels for classification tasks or alternatively $y \in \mathbb{R}$ for regression problems. We will assume that (x_i, y_i), $i = 1, \ldots, n$ are independent, identically distributed random variables.

It is often the case that the problem of classification, regression or relevant feature extraction can be made substantially easier when the data is mapped into an appropriate high dimensional space by some non-linear mapping ϕ and linear methods are applied to the transformed data. If the algorithm

is expressible in terms of dot products and if the non-linear mapping

$$\phi : \mathbb{R}^d \to \mathcal{H} \tag{1}$$

is such that the dot products of the images of any two points x_1 and x_2 under ϕ can be computed as a function of x_1 and x_2 only and in $\mathrm{poly}(d)$-time without explicitly calculating $\phi(x_1)$ or $\phi(x_2)$ then the algorithm remains tractable, regardless of the dimensionality of $\mathcal{H}$. This allows us to consider very high or even infinite dimensional image spaces $\mathcal{H}$. We may as well start by choosing a symmetric positive definite function

$$k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}, \tag{2}$$

called the kernel function (see e.g. [4]). Then the closure of the linear span of the set

$$\{\, k(x, \cdot) \,|\, x \in \mathbb{R}^d \,\} \tag{3}$$

gives rise to a Hilbert space $\mathcal{H}$ where the inner product is defined such that it satisfies

$$\langle k(x_1, \cdot), k(x_2, \cdot) \rangle = k(x_1, x_2) \tag{4}$$

for all points $x_1, x_2 \in \mathbb{R}^d$ [20]. The choice of k automatically gives rise to the mapping

$$\phi : \mathbb{R}^d \to \mathcal{H} \tag{5}$$

defined by $\phi(x) = k(x, \cdot)$. This is called the kernel mapping idea [1; 23; 30], which is used for the derivation of novel kernel based algorithm described in the thesis.

Results

The theses of the present dissertation can be differentiated in two ways and can be separated into two different groups. In one reading, the results acquired by the author fall into the topic of kernel-based feature extraction and classification methods within the field of machine learning. In an other reading, we can learn about algorithmic constructions and practical applications. Hereunder, following the structure of the dissertation, the results are introduced according to the first approach. It is important to note that the list of results below enumerate only those parts of the novelties to which the author has contributed in major part.

The author's kernel-based feature extraction methods form the first group of results. These results are described in detail in Chapter 2 and 3 of the dissertation.

I/1. The author has defined the direct version of the MMDA algorithm [11; 16]. This method employs a feature extraction technique that increases the efficiency of classification methods. The author managed to prove the feasibility of the defined method by applying it on several examples of the UCI machine learning database [3].

I/2. The MMDA algorithm is apt by nature to reduce high dimensional feature spaces in order to increase classification efficiency. The author has developed a modified version of this algorithm for face recognition. With the help of the FERET gold standard face recognition database [6], he managed to prove the usability of the introduced method. He also managed to surpass the state-of-the-art results in the field [16].

I/3. Beyond classification, regression problems may also form the focus of feature extraction. The author has also developed a version of the MMDA algorithm for solving regression tasks, with a focus on the retrieval of correlation-free features. The name of the method is Kernel Decorrelated Learning Regression (KDLR) [28]. Based on tests performed over standard regression problems we can state that the approach leads to more efficient regression in practice.

I/4. The author proposed the combination of the statistics-based average derivative estimation method on the one hand, and kernel functions on the other hand. The aim of this novel method called Kernel Average Derivative Estimation (KADE) is the identification of sub-spaces that are relevant from a regression's point of view [28]. By testing on artificial data and comparing relating algorithms the author proved that the identified sub-spaces enable more effective regression in a good number of cases.

Novel kernel-based classification algorithms form the second group of results. These results are detailed in Chapter 4 and 5.

II/1. The author defined a family of hyperplane-based classification methods [13]. He proposed three modifications, each following traditional geometrical concepts: i) he used various loss functions in hyperplane-based classification; ii) he applied linear regression in a unique way to improve classification; and iii) he developed the Minor Component Classifier method, which defines a classification hyperplane with the help of an eigenvector pertaining to the smallest eigenvalue of a sample point matrix. The author formed the testing environment of the methods and performed demonstrational tests. Results prove that the methods developed by the author are comparable to results performed by SVM [26].

II/2. The author constructed a classification scheme called Convex Machine Technique, which applies a rare combination of basis functions. The method contains a number of machine learning techniques, such as: Support Vector Machine (SVM) [26], Smooth Support Vector Machine (SSVM) [18], Least Square Support Vector Machine (LSVM) [27], Kernel Logistic Regression (KLR) [8], just to mention a few. Inspired by basic numerical mathematical methods, the author also developed three basis function selection techniques (RANDOM, MGRAMM, CORR) [14], which he tested on certain elements of the UCI data repository [3].

II/3. He further developed three complex basis function selection techniques (SFS, SFFS, PTA) in order to improve the efficiency of classification on the one hand, and to decrease the size complexity of the classification model on the other hand. The defined methods build on the analogy of state-of-the-art feature space selection techniques. Base on test results, it can be declared that these methods support effective classification [28].

Thesis Topics	[11]	[13]	[14]	[15]	[16]	[28]	*Chapter*	*Type*
MMDA	•						2	Feature Extraction
MMDA FACE version	•				•		2	Feature Extraction
KDLR	•					•	3	Feature Extraction
KADE						•	3	Feature Extraction
Hyperplane Classifiers		•					4	Classification
Convex Networks			•	•			5	Classification
Basic Basis Selection Methods			•				5	Basis Selection
Complex Basis Selection Methods				•			5	Basis Selection

1. táblázat. The relation between the thesis topics and the corresponding publications.

Finally, Table 1 summarizes which publication covers which method of the thesis.

Conclusions

The thesis is built around the kernel idea, which is suitable for the transformation of linear models into non-linear ones, while only minimally increasing the complexity of the model. The kernel idea is able to transform algorithms that only use the dot product of a set of sample vectors as their input. In this case, we can acquire alternative models with the non-linear re-definition of the dot product. Thus, the main objective is to re-define the basic tasks of machine learning, such as feature extraction, classification, regression in the form of dot products. The results included in my dissertation contribute to the described idea with a number of novel algorithms.

In the first part of the thesis, I defined feature extraction techniques that effectively increase the precision of classification and regression methods. We succeeded in justifying the grounds for these techniques by making them work on standard data bases of machine learning and on the task of face recognition. In summary, we can state that the four elaborated methods pave the way to the effective reduction of high dimensional feature spaces.

In the second part, I focused on the topic of classification. I first introduced a family of hyperplane-based classification methods. These outline a group of methods that gives a deeper insight into the nature of how variably the hyperplane can be used for classification. The leading motif of the newly defined method group is the underlying geometrical approach. In the same part, we define a general classification scheme leading to the optimisation of convex functions, which comprises a number of techniques developed over the past few years. The unified approach that determined our way of thinking in this case supposes a traditional numerical mathematical thinking. The methods introduced in part two performed well on both real and artificial data. With this, we managed to emphasize usability beyond the constructional results.

Appendices

A.1 The MMDA algorithm

Let us fix a positive real number C that we will use to weight the misclassification cost. Now define the maximum margin separation problem with orthogonality constraint (MMSO problem) as follows: Let u be a d-dimensional vector: $u \in \mathbb{R}^d$. The MMSO problem parameterized by (X, y, C, u) is to find $w \in \mathbb{R}^d$, $b \in \mathbb{R}$ and $\xi = (\xi_1, \ldots, \xi_n) \in \mathbb{R}^n$ such that

$$\begin{aligned} \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{n}\xi_i \to \min \text{ s.t.} \\ y_i(w^T x_i + b) \geq 1 - \xi_i, \\ \xi_i \geq 0, \quad i = 1, \ldots, n, \\ u^T w = 0. \end{aligned} \tag{6}$$

The formalism remains similar when the number of orthogonality constraints – r, say – is bigger than one. The corresponding MMSO problem will be denoted by (X, y, C, U), where $U = (u_1, \ldots, u_r)$ is the matrix of vectors that are used to define the orthogonality constraints (i.e. $U^T w = 0$).

It is not difficult to prove that the solution of an MMSO problem (X, y, C, U) may be obtained by solving the following dual quadratic programming problem:

$$\begin{aligned} -\frac{1}{2}\left(\alpha^\top Y K Y \alpha + \gamma^\top U^T U \gamma\right) + \alpha^\top 1 + \gamma^\top U^T X Y \alpha \to \max_{\alpha,\gamma} \\ \text{such that} \quad y^T\alpha = 0, \quad 0 \leq \alpha \leq C1, \end{aligned} \tag{7}$$

where $K = X^T X$ and $1 = (1, \ldots, 1)^T$. Since the number of columns of U is r, the dimensionality of γ will also be r, and hence the number of variables in the above quadratic programming problem will be $n + r$.

The direct method works as follows: Given the data (X, y, C), let (w_1, b_1) be the solution of the MMSO problem $(X, y, C, 0)$. Assuming that the solution vectors $(w_1, b_1), \ldots, (w_{r-1}, b_{r-1})$ have already been computed, (w_r, b_r) is obtained as the solution of the MMSO problem (X, y, C, W_{r-1}), where $W_{r-1} = (w_1, \ldots, w_{r-1})$.

A.2 Feature Extraction by MMDA for Face Recognition

Human face recognition is a special classification problem where the number of classes is high, there are only a few samples per class and the input space is high-dimensional. These properties make face recognition an especially challenging classification problem.

Since MMDA is defined for binary classification problems, with multi-class problems one needs to group certain classes together. In [11] it was suggested that MMDA should be used following a „one-vs.-all" approach: basically when the number of classes is m, MMDA is run m times with one class against all the others. This is a simple approach and in [11] it was suggested that although it

Algorithm 1 Feature Extraction by MMDA for Face Recognition

input: $(m, (x_1, y_1) \ldots, (x_N, y_N))$ // no. of subjects, list of face-image, person id pairs

$F := ()$; $X^i := \{x_j | y_j = i\}$, $i = 1, \ldots, m$; // images of person i

$(w_1, \ldots, w_n) := \mathrm{FE}((x_1, y_1) \ldots, (x_N, y_N))$; // extract n features using method FE

for $i \in \{1, \ldots, n\}$ **do**

$z_j := w_i^T x_j$, $j = 1, \ldots, N$; // project images

$Z^i := \{z_j | y_j = i\}$, $i = 1, \ldots, m$; // collect projected images of person i

Find $(v_1, \ldots, v_m) \in \{-1, 1\}^m$ such that

$$\sum_{\substack{v_i=-1, v_j=-1 \\ i \neq j}} \sum_{\substack{z \in Z^i \\ z' \in Z^j}} (z - z')^2 + \sum_{\substack{v_i=1, v_j=1 \\ i \neq j}} \sum_{\substack{z \in Z^i \\ z' \in Z^j}} (z - z')^2$$

is minimized.

$F_0 := MMDA(\cup_{v_i=-1} X^i, \cup_{v_j=+1} X^j)$; // extract features using MMDA

append$(F, F0)$; // append features to the list of features extracted so far

end for

return F

is likely to be suboptimal, it can yield a sufficiently good performance even when compared with the more involved approach based on output-coding [12].

It should be mentioned that the one-vs.-all approach cannot produce useful features in face recognition tasks as here all the m subproblems are seriously skewed with one class having only a few elements and the other has lots. Such skewed distributions will yield highly correlated features for the independent subproblems since the subproblems are „well aligned" (it is easy to see that forcing independent features to be decorrelated does not help either due to the large overlap of the problems). The same conclusion holds for the features obtained using the output-coding approach [12] since there classes are grouped together without taking into account their relations in the input space. In a typical subtask persons with very different faces could be grouped together while persons with similar faces might be assigned to different classes.

This gives us the idea of using the available data to create the binary classification subproblems for MMDA. The particular approach suggested here is to use information in the images to create the subproblems. One approach for doing just this is the following. Some method (like LDA [9] or PCA [10]) is used to generate a number of unrelated features. For each feature, the projections of training images on a selected feature are computed. This produces a number of points on the real-line. Next, the persons in the training set are grouped into two groups so as to minimize the total within-group distortion between the previously calculated points on the real-line (this is a special case of k-means clustering and can be implemented efficiently). MMDA is then run on this binary classification problem

(on the original, untransformed images) and the corresponding features are then saved. The process is continued with the next feature. The union of features extracted this way defines the extracted feature space. The proposed algorithm is listed above.

A.3 Kernel Decorrelated Learning Regression

We consider regression problems, where the data (X_i, Y_i) are independent, identically distributed random variables, L is loss function such as e.g. quadratic loss function $L(y, z) = (y - z)^2$, and we seek to determine the regressor $f(x) = \operatorname{argmin}_y E[L(Y, y)|X = x]$.

Let us first consider the model

$$Y = \sum_i \beta_i g_i(X) + \epsilon, \tag{8}$$

where $g_i : X \to \mathbb{R}$ are unknown functions, and ϵ is noise variable, independent of Y, X. We shall consider estimating g_i by means of an iterative procedure. One view of the model is then to treat the $Y = \beta^T \gamma + \epsilon$ as a linear regression problem, where $\gamma = (g_1, \ldots, g_m)$. We shall assume that the vector β is such that $0 \leq \beta \leq 1$, $\beta^T e = 1$, where $e = (1, 1, \ldots, 1)^T$, i.e., the output can be obtained as a noisy convex combination of the 'features' $g_1(X), \ldots, g_m(X)$. We shall further assume that the loss function is the quadratic loss.

Let $g = \sum_i \beta_i g_i$, f arbitrary. Then, it is not hard to see that

$$\operatorname{Loss}(g) = \sum_i \beta_i \operatorname{Loss}(g_i) - \sum_i \beta_i E[(g_i(X) - g(X))^2] \tag{9}$$

and

$$\operatorname{Loss}(g) = E[(g(X) - f(X))^2]. \tag{10}$$

This formula, first given in [17] is called „ambiguity decomposition" (AD). The ensemble loss can be decreased if the ambiguity of the ensemble is maximized whilst keeping the loss of the individual members low. Now, we obtain easily

$$\sum_i \beta_i E[(g_i(X) - g(X))^2] = \sum_i (\beta_i^2 - \beta_i)\Big(E[g_i(X)]^2 + \operatorname{Var}[g_i(X)]\Big) - \sum_{i \neq j} \beta_i \beta_j \operatorname{Cov}(g_i(X), g_j(X)). \tag{11}$$

Therefore, given two ensembles (g_i), $(\hat{g}_i)$ satisfying $E[g_i(X)] = E[\hat{g}_i(X)]$, $\operatorname{Var}[g_i(X)] = \operatorname{Var}[\hat{g}_i(X)]$, if

$$\sum_{i \neq j} \beta_i \beta_j E[g_i(X)\, g_j(X)] < \sum_{i \neq j} \beta_i \beta_j E[\hat{g}_i(X)\, \hat{g}_j(X)] \tag{12}$$

then $\operatorname{Loss}(g) < \operatorname{Loss}(\hat{g})$. The assumption of equal expected values and variances is motivated by assuming that each g_i should match the regressor function f as closely as it is possible and hence the expected value and variance of $g_i(X)$ are controlled by this desire. As a conclusion, we have that one way of have a small ensemble loss is to enforce orthogonality: $E[g_i(X) g_j(X)] = 0, \quad i \neq j$.

Now, let $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a positive definite kernel, $\mathcal{H}$ be the RKHS corresponding to k. Let $\{(x_i, y_i)\}_{i=1}^n$ denote the observed data (again, x_i, y_i are i.i.d.) and let $L(y, z)$ be a loss function, e.g. $L(y, z) = (y - z)^2$, $f \in \mathcal{H}$. Define

$$R(f) = \frac{1}{n} \sum_{i=1}^{n} L(f(x_i), y_i) + \lambda \|f\|_2, \tag{13}$$

where $\|f\|_2$ is in the norm of $\mathcal{H}$ (i.e. this is ridge regression in the case of the quadratic loss). By the „Representer Theorem" of Wahba [33] $f \in \operatorname{span}(\Phi)$, where $\Phi = (\phi_1, \ldots, \phi_n)$ and $\phi_i : \mathbb{R}^d \to \mathbb{R}$ is defined by $\phi_i(x) = k(x_i, x)$. E.g. assume $f = \Phi\alpha$ for some $\alpha \in \mathbb{R}^n$. Equation (13) can be solved by

$$R(\alpha; X; k) = \frac{1}{n} \sum_{i=1}^{n} L((\Phi\alpha)(x_i), y_i) + \lambda \alpha^T K \alpha, \tag{14}$$

where $K_{ij} = k(x_i, x_j)$ and $X = (x_1, \ldots, x_n)$. When the dataset X and the kernel k are fixed we will often write $R(\alpha)$ instead of $R(\alpha; X; k)$. Similarly, when the kernel is fixed we will use $R(\alpha; X)$.

Now assume $g_i = \Phi\alpha_i$, $g_j = \Phi\alpha_j$. By replacing the expectation operation with the the empirical mean in orthogonality criterion we obtain

$$0 = \sum_{k=1}^{n} g_i(x_k) g_j(x_k) = \alpha_k^T K^2 \alpha_j. \tag{15}$$

Therefore an iterative procedure that optimizes $R(\alpha)$ and respects the orthogonality criterion is as follows: Given $\alpha_1, \ldots, \alpha_i$, let

$$\alpha_{i+1} = \operatorname*{argmin}_{\alpha} \{\, R(\alpha) \,|\, \alpha_j^T K^2 \alpha = 0,\ 1 \le j \le i \,\}. \tag{16}$$

Once $\alpha_1, \ldots, \alpha_k$ are computed for some $k > 0$, one may estimate the optimal mixing coefficients β_i by e.g. ordinary or regularized (linear) least squares. We call the method obtained by solving (16) together with the method used to obtain the mixing coefficients β_i, *decorrelated learning regression* (DLR).

The solution of (16) can be obtained by solving the Langrangian dual of the quadratic programming problem (16). For this, assume that the solutions up to step i are obtained in the form ΦA_i where we have collected the vectors $\alpha_1, \ldots, \alpha_i$ into the matrix A_i. Also, consider now the ϵ-loss of function of Vapnik [30]: $L(y, z) = \max(0, |y - z| - \epsilon)$. It is relatively easy to derive that the problem reduces to the following quadratic programing problem:

$$\begin{aligned} L(\alpha, \alpha^*, \beta) = &-\frac{1}{2}(\alpha - \alpha^*)^T K (\alpha - \alpha^*) - (\alpha - \alpha^*)^T K^2 A_i \beta \\ &- \frac{1}{2}\beta^T A_i K^3 A_i \beta + (\alpha - \alpha^*)^T y - \epsilon(\alpha + \alpha^*)^T e \to \max \\ \text{s.t. } &0 \le \alpha_i, \alpha_i^* \le C \ \forall i. \end{aligned} \tag{17}$$

A.4 Kernel Average Derivative Estimation

Here we assume that the unknown regressor function f can be written in the form

$$f(x) = f_0(Bx) \tag{18}$$

for some matrix $B \in \mathbb{R}^{m \times d}$ with $m \ll d$ (i.e. $BB^T = I_m$). Here f_0 is an unknown *link* function. Our goal here is to find the effective dimension m and to describe the effective dimension reducing space $\mathcal{S} = \Im B^T$ [19]. The basic idea of average derivative estimation is as follows: Considering the derivative of f we get that for all $x \in \mathbb{R}^d$ and for

$$F(x) \stackrel{\text{def}}{=} B^T f_0'(Bx) \tag{19}$$

we have $F(x) \in \mathcal{S}$.

The basic idea now is to estimate f using a non-parametric estimator. Let $\hat{f}$ denote such an estimate obtained and let $x_1, \ldots, x_n$ be the data points used. Then define

$$\hat{F}(x) = d/dx \hat{f} \tag{20}$$

and compute the eigenvalue decomposition of

$$M = \sum_i \hat{F}(x_i)\hat{F}(x_i)^T. \tag{21}$$

If $\hat{F} = F$ then it is easy to see that only the first m eigenvalues of M differ from zero. Since $\hat{F}$ is only an approximation of F we may expect that M will have more than m non-zero eigenvalues. However, the hope is that the dimensionality of the effective dimension reducing subspace can be recovered by detecting a gap in the spectrum.

Here we propose to use kernel machines to obtain $\hat{f}$, an estimate of f. We shall call the resulting method KADE (Kernel based Average Derivative Estimation). The choice of using kernel machines is motivated by the widely accepted view that kernel machines are less sensitive to the dimensionality of the input space which is important in the first step of the algorithm.

A.5 Hyperplane Classifiers

Hyperplane-based classification methods may seem a weak approximators for separating positive and negative samples. This is because in low dimension it is very easy to define a sample set, where linear separation is not viable. The kernel idea, however, "step over" these limitations and, therefore examining the hyperplane-based techniques has become the main focus nowadays.

In the following we briefly describe three methods. The first one extend the hyperplane-based approach with loss functions, the second one uses the regression formalism in a unique way for classification, while the third one - which may be the most important result - is a novel method called Minor Component Classifier. Here the classification is carried out via merging the input and output space of the classification task. By employing the kernel-idea we got the following methods:

a) Linear Classifier with Loss functions:

$$\min_{\boldsymbol{\alpha}} \sum_{i=1}^{n} g\left(y_i \sum_{j=1}^{n} \alpha_i k\left(\begin{pmatrix} \mathbf{x}_i \\ 1 \end{pmatrix}, \begin{pmatrix} \mathbf{x}_j \\ 1 \end{pmatrix} \right) \right), \tag{22}$$

where g is a loss function, while k is a kernel function. For solving the above unconstrained minimization problem we can use the Quasi-Newton or even the Newton iteration method.
b) Linear Regression for Classification: the decision rule for an arbitrary sample z is

$$sign\left((z^T 1) X_1 (K^T K)^+ K^T Y \right), \tag{23}$$

where

$$X_1 = \begin{pmatrix} \mathbf{x}_1^T & 1 \\ \vdots & \vdots \\ \mathbf{x}_n^T & 1 \end{pmatrix} \text{ and } K_{ij} = k\left(\begin{pmatrix} \mathbf{x}_i \\ 1 \end{pmatrix}, \begin{pmatrix} \mathbf{x}_j \\ 1 \end{pmatrix} \right). \tag{24}$$

c) Minor Component Classifier:

$$\min_{\boldsymbol{\beta}} \frac{\boldsymbol{\beta}^T \bar{K} \bar{K} \boldsymbol{\beta}}{\boldsymbol{\beta}^T \bar{K} \boldsymbol{\beta}}, \tag{25}$$

where the matrix $\bar{K}$ contain the pairwise dot products of transformed points:

$$\bar{K}_{ij} = k\left(\begin{pmatrix} \mathbf{x}_i \\ y_i \\ 1 \end{pmatrix}, \begin{pmatrix} \mathbf{x}_j \\ y_j \\ 1 \end{pmatrix} \right). \tag{26}$$

The solution of (25) can be obtained by finding the eigenvector corresponding to the smallest nontrivial eigenvalue of the generalized eigenproblem

$$\bar{K} \bar{K} \boldsymbol{\beta} = \lambda \bar{K} \boldsymbol{\beta}. \tag{27}$$

A.5 Convex Machines

Now - as in earlier - consider the problem of classifying *n* points in a compact set $\mathcal{X}$ over $\mathbb{R}^m$, represented by vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$, according to the membership of each point $\mathbf{x}_i$ in the classes $\{1, \ldots, c\}$, as specified by $y_1, \ldots, y_n$. A multiclass problem can be transformed into a set of binary classification tasks – where we usually have y_i in $\{-1, +1\}$ – using various algorithms like the one-against-all method [34] or the output coding scheme [12], say. Hence our investigation here can be restricted to the problem of binary classification without loss of generality.

Now let V be a vector space, which will be viewed as a function space here. Let $S \subset V$ denote a finite set of basis functions

$$S = \{f_1(\mathbf{x}), \ldots, f_k(\mathbf{x})\}, \qquad f_i : \mathcal{X} \to \mathbb{R} \tag{28}$$

and let $Span(S)$ stand for the linear subspace spanned by S, that is

$$Span(S) = \left\{ f_{\boldsymbol{\alpha}} : \mathcal{X} \to \mathbb{R} \mid f_{\boldsymbol{\alpha}}(\mathbf{x}) = \sum_{i=1}^{k} \alpha_i f_i(\mathbf{x}),\ \mathbf{x} \in \mathcal{X},\ \boldsymbol{\alpha} \in \mathbb{R}^k \right\}. \tag{29}$$

The classification problem may be defined by the following optimization problem

$$\min_{f_{\boldsymbol{\alpha}}(x) \in Span(S)} E_{\mathbf{x},y} L(f_{\boldsymbol{\alpha}}(x), y), \tag{30}$$

where L is a loss function measuring the quality of the predictor $f_{\boldsymbol{\alpha}}(x)$ and E denotes the expectation over (x, y). A possible convex restriction of Eq. (30) is

$$\min_{f_{\boldsymbol{\alpha}}(x) \in Box(S,\mathcal{B})} E_{\mathbf{x},y} H(y_i f_{\boldsymbol{\alpha}}(x_i)). \tag{31}$$

Here the loss function $L(f_{\boldsymbol{\alpha}}(x), y)$ is assumed to be of the form $H(y_i f_{\boldsymbol{\alpha}}(x_i))$, where $H : \mathbb{R} \to \mathbb{R}$ is a twice continuously differentiable, non-negative, decreasing, convex function. $Box(S, \mathcal{B})$, furthermore, is a box constrained subset of $Span(S)$, i.e. $\mathcal{B} = \mathcal{B}_1 \times \cdots \times \mathcal{B}_n \subseteq \mathbb{R}^n$ is a cartesian product space of non-empty intervals and

$$Box(S, \mathcal{B}) = \left\{ f_{\boldsymbol{\alpha}} : \mathcal{X} \to \mathbb{R} \mid f_{\boldsymbol{\alpha}}(\mathbf{x}) = \sum_{i=1}^{k} \alpha_i f_i(\mathbf{x}),\ \mathbf{x} \in \mathcal{X},\ \boldsymbol{\alpha} \in \mathcal{B} \right\}. \tag{32}$$

Due to the fact that the problem of approximating a function from sparse data is ill-posed [32] we have already assumed two different types of restrictions on the shape of the predictor function: i) $f_{\boldsymbol{\alpha}}(x)$ is a linear combination of a *finite* set of basis functions and ii), this linear combination is constrained components by components using *intervals*. In addition, inspired by regularization theory [7; 29] we add a regularization term to the cost function to be optimized:

$$\min_{f_{\boldsymbol{\alpha}}(x) \in Box(S,\mathcal{B})} E_{\mathbf{x},y} H(y_i f_{\boldsymbol{\alpha}}(x_i)) + \lambda \|\boldsymbol{\alpha}\|_A^2, \tag{33}$$

where $\lambda > 0$ and $A \in \mathbb{R}^{k \times k}$ is an arbitrary symmetric positive-definite matrix. We call this general formula as 'Convex Machines'.

A.5 Basic heuristics

Sparse solutions (i.e. sparse input-output models) for classification problems are beneficial for two reasons. First, we may avoid the problem of overfitting and second, both the optimization procedure and the evaluation of the classifier is faster. Therefore forcing as many $\boldsymbol{\alpha}_i$ parameters to be zero as possible in the predictor function

$$f_{\boldsymbol{\alpha}} = \sum_{i=1}^{k} \alpha_i f_i(\mathbf{x}) \tag{34}$$

is a reasonable strategy. For the sake of controlling the sparsity the number of basis functions with zero-coefficients will be restricted by making the following assumption

$$\sum_{i=1}^{k} |sign(\alpha_i)| \leq q, \tag{35}$$

where q is a preset parameter. Unluckily, such a condition makes the optimization problem of Eq. (33) combinatorial, so the suggested nonlinear Gauss-Seidel technique in its original form cannot be applied. Our aim is to select from the available basis functions a subset of order q at most with minimal function value. This task is NP hard [5] so the only effective way here is to employ heuristics.

RANDOM The simplest strategy is the random selection approach when we randomly select q basis functions from among the k basis functions. This approach does not have an objective function that can be minimized so we will choose instead the subset with the best performance after several executions.

MGRAMM Convex Machines approximates the optimal separator surface using a linear combination of the basis functions. Hence the approximation can be performed on an orthogonal basis of the function space, as in the case of the result of the Gramm-Schmidt orthogonalization algorithm. Despite this, the dimension of the basis is the rank of the function set which can exceed the desired number q. Moreover, the algorithm generates an orthogonal function system with linear combinations of basis functions instead of selecting the individual functions.

To solve the above we will define a greedy iterative selection strategy based on a modified version of the Gramm-Schmidt orthogonalization algorithm. Among the available basis functions we choose the one with a maximal residual norm after the Gramm-Schmidt process at each step. The result of this greedy method is not the orthogonal function system itself but the basis functions used in the linear combinations.

Assume that the basis functions are elements of L_2 so the dot product is the integral of the product function. When analytical computations of the integrals are not possible we utilize the following approximation in the algorithm using the sample points

$$\langle f, g \rangle = \sum_{i=1}^{n} f(\mathbf{x}_i) g(\mathbf{x}_i) \qquad f, g : \mathcal{X} \to \mathbb{R}. \tag{36}$$

CORR The MGRAMM method tries to choose an orthogonal basis of the functions with the help of the Gramm-Schmidt process. The choice might be good when the dot product of functions is available. Employing the approximation in Eq. (36) the result of the algorithm will be also just an approximation of the desired basis.

Such an estimation can be carried out in different ways. The orthogonality of the elements in the basis can be also employed, since the mutual correlation coefficients must be zero. Our aim is to select functions such that the squared sum of the element in the correlation matrix should be minimal. Similar to MGRAMM this method will be a greedy iterative process and also exploit the fact that the mutual correlation coefficient for normalized functions takes the form of Eq. (36).

A.6 Complex Heuristics

Measure-based subset selection is an active area of other fields in artificial intelligence like Feature Selection [24], say. In this context one should select r features from the available m to maximize the classification performance of a machine learning algorithm. The elaborated techniques can be employed for our subset selection problem if the required measure is replaced by the objective function value of the CM task.

SFS The Sequential Forward Selection method is a greedy approach for the measure-based subset selection problem. Starting with the empty index set it extends the indices with the locally optimal element without backtracking.

PTA The SFS method is a sequential algorithm, hence previous steps cannot be modified when detecting their latter impact on the result. A solution to the problem is the Plus l Take Away r approach. It periodically extends the actual index set by l elements and afterwards removes r so that the measure is locally optimal after each step. By doing this the effects of previous selections can be eliminated during the execution of the subroutine.

SFFS When running a PTA routine r removing steps always follow l extending ones. Hence it is possible to execute a removing step when the evolving set has a worse measure value than the previous one of the same order. Conversely, an extending step can be performed when we get a better solution at that particular level by removing a function. These problems are absent in the Sequential Forward Floating Selection algorithm. It sequentially removes elements after one extending step while the measure we obtain is better than the previous ones of the same order.

Hivatkozások

[1] M. Aizerman, E. Braverman, and L. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964.

[2] S. Akaho. A kernel method for canonical correlation analysis. In *International Meeting of Psychometric Society (IMPS), Osaka*, 2001.

[3] C.L. Blake and C.J. Merz. UCI repository of machine learning databases, http://www.ics.uci.edu/∼mlearn/mlrepository.html, 1998.

[4] B.E. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In *Computational Learing Theory*, pages 144–152, 1992.

[5] Thomas M. Cover and Jan Van Campenhout. On the possible orderings in the measurement selection problem. *IEEE Trans. Systems, Man, and Cybernetics*, 7:657–661, 1977.

[6] M. Teixeira D. Bolme, R. Beveridge and B. Draper. The csu face identification evaluation system: Its purpose, features and structure. In *International Conference on Vision Systems*, pages 304–311. Springer-Verlag, 2003.

[7] T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Advances in Computational Mathematics*, 13(1):1–50, 2000.

[8] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, 28(2):337–407, 2000.

[9] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, San Diego, California, 1990.

[10] I. T Jolliffe. *Principal component analysis*. New York : Springer, 2002.

[11] A. Kocsor, K. Kovács, and Cs. Szepesvári. Margin maximizing discriminant analysis. In Fosca Giannotti et al. Jean-François Boulicaut, Floriana Esposito, editor, *Machine Learning: ECML 2004: 15th European Conference on Machine Learning, vol. 3201*, pages 227–238. Springer-Verlag GmbH, September 2004.

[12] E. B. Kong and T. G. Dietterich. Error-correcting output coding corrects bias and variance. In *International Conference on Machine Learning (ICML)*, pages 313–321, 1995.

[13] K. Kovács and A. Kocsor. Various hyperplane classifiers using kernel feature spaces. *Acta Cybernetica*, 16(2):271–278, 2003.

[14] K. Kovács and A. Kocsor. Classification using sparse combination of basis functions. *Acta Cybernetica*, 17(2):311–323, 2004.

[15] K. Kovács and A. Kocsor. Improving a basis function based classification method using feature selection algorithms, accepted for ieee international workshop on soft computing applications. In *IEEE International Workshop on Soft Computing Applications, (IEEE-SOFA)*, pages 208–211, 2005.

[16] K. Kovács, A. Kocsor, and Cs. Szepesvári. Maximum margin discriminant analysis based face recognition. In M. Vincze D. Chetverikov, L. Czuni, editor, *Proceedings of the Joint Hungarian-Austrian Conference on Image Processing and Pattern Recognition, HACIPPR*, pages 71–78. Oesterreichische Computer Gesellschaft, 2005.

[17] A. Krogh and J. Vedelsby. Neural network ensembles, cross validation, and active learning. In *Advances in Neural Information Processing Systems NIPS 11*, pages 231–238, 1995.

[18] Y. Lee and O. L. Mangasarian. SSVM: A smooth support vector machine. *Computational Optimization and Applications*, 20:5–22, 2001.

[19] K.-C. Li. Sliced inverse regression for dimension reduction. (With discussion). *J. Amer. Statist. Ass.*, 86(414):316–342, 1991.

[20] J. Mercer. Functions of positive and negative type and their connection with the theory of integral equations. *Philos. Trans. Roy. Soc. London, A*, 209:415–446, 1909.

[21] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14 (NIPS)*, pages 849–856. MIT Press, Cambridge, 2002.

[22] I. Nouretdinov, T. Melluish, and V. Vovk. Ridge regressioon confidence machine. In *Proc. 18th International Conf. on Machine Learning*, pages 385–392. Morgan Kaufmann, San Francisco, CA, 2001.

[23] T. Poggio. On optimal nonlinear associative recall. *Biological Cybernetics*, 19:201–209, 1975.

[24] P. Pudil, J. Novovicova, and J. Kittler. Feature selection based on the approximation of class densities by finite mixtures of the special type. *Pattern Recognition*, 28(9):1389–1397, 1995.

[25] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.

[26] B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, 2002.

[27] J. A. K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[28] Cs. Szepesvári, A. Kocsor, and K. Kovács. Kernel machine based feature extraction algorithm for regression problems. In Lorenza Saitta Ramon López de Mántaras, editor, *Proceedings of the 16th Eureopean Conference on Artificial Intelligence, ECAI 2004*, pages 1091–1091, Valencia, Spain, August 2004. IOS Press.

[29] A. N. Tikhonov and V. Y. Arsenin. *Solutions of Ill-posed Problems*. W. H. Winston, Washington, D.C., 1977.

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

[31] V. Vapnik. *Statistical Learning Theory*. John Wiley & Sons Inc., 1998.

[32] G. Wahba. *Splines models for Observational Data*. Vol. 59, SIAM, Philadelphia, 1990.

[33] Grace Wahba. Support vector machines, reproducing kernel Hilbert spaces, and randomized GACV. In *Advances in kernel methods: support vector learning*, pages 69–88. MIT Press, 1999.

[34] J. Weston and C. Watkins. Support vector machines for multiclass pattern recognition. In *Proceedings of the Seventh European Symposium On Artificial Neural Networks (ESANN)*, pages 219–224, 1999.