

Functional Feature Selection by Weighted Projections in Pathological Voice Detection*

Luis Sánchez Giraldo^{1, **}, Fernando Martínez Tabares²,
and Germán Castellanos Domínguez²

¹ University of Florida, Gainesville, FL, 32611, USA
luisitobarcito@ufl.edu

² Universidad Nacional de Colombia Km 7 vía al Magdalena, Manizales, Colombia
{fmartinez, cgcastellanosd}@unal.edu.co

Abstract. In this paper, we introduce an adaptation of a multivariate feature selection method to deal with functional features. In our case, observations are described by a set of functions defined over a common domain (e.g. a time interval). The feature selection method consists on combining variable weighting with a feature extraction projection. Although the employed method was primarily intended for observations described by vectors in \mathbb{R}^n , we propose a simple extension that allows us to select a set of functional features, which is well suited for classification. This study is complemented by the incorporation of Functional Principal Component Analysis (FPCA) that project functions into a finite dimensional space where we can perform classification easily. Another remarkable property of FPCA is that it can provide insight about the nature of the functional features. The proposed algorithms are tested on a pathological voice detection task. Two databases are considered: Massachusetts Eye and Ear Infirmary Voice Laboratory voice disorders database and Universidad Politécnica de Madrid voice database. As a result, we obtain a canonical function whose time average is enough to reach similar performances to the ones reported in the literature.

1 Introduction

Pattern recognition from the side of machine learning is more concerned with rather general methods for extracting information from the available data, and thence the task of handcrafting complex features for each individual problem turns to be less crucial. Consequently, large sets of simpler features are employed, however, since these variables are less refined for each problem they require of some processing. Feature selection has been discussed in the past by several authors [1,2] as an important preprocessing stage to improve results during and after training in learning processes that also attempts to overcome the curse of dimensionality. More recent studies on this subject are found in [3,4]. A common issue in machine learning approach takes place when the number

* This is part of the project 20201004208, funded by Universidad Nacional de Colombia.

** The author was a student at Universidad Nacional de Colombia.

relevant features is considerably smaller and the computation time, that grows exponentially with the number of features, becomes prohibitively large for real time applications [5]. Yet there is a more fundamental issue related to interpretation, when there is a large amount of incoming data. Reducing the size of data either by encoding or removing irrelevant information, becomes necessary if one wants to achieve good performance in the system as well as insightful results. In our work, we attempt to combine feature selection and extraction with a twofold purpose: interpretation and generalization.

Functional data analysis can be regarded as an extension of the existing multivariate methods to more involved representations. The typical framework in multivariate statistics deals with descriptions of the objects as vectors in \mathbb{R}^n . In the case of functional data, we have a set of functions with the same domain that are extracted from each observation. Even if these functions are in a discrete domain, the dimensionality of data poses a challenging problem. Ramsay and Silverman [6] give a thorough presentation of the topic. Methods such as PCA on functional data can be found in [7], and nonparametric extensions are discussed in [8]. Despite there is clear interest on developing methods or adaptations for this functional representations, none of these works address the problem of selecting the functions that might provide the relevant information for the analysis; what is more, most of the analysis focuses on objects described by a single function. Even though the authors may argue the extension of these methods to several functions is straightforward, there must be some concern on the choice of the set of functions that ought be analyzed. Having irrelevant functions describing the problem may hinder the effect of the relevant ones.

In this work, we use a weighting method presented in [9] for attaining subset selection. The method combines feature selection and feature extraction on the same process. Particularly, we employ Weighted Regularized Discriminant Analysis (WRDA), which allows us to obtain a rather simple generalization to functional data. The optimization process consists on maximizing a trace ratio over a fixed number of discriminant directions. The paper starts with the description of the employed method for the case of regular features (vectors in \mathbb{R}^n). Then, we describe the adaptation process for functional data. In order to observe the effectiveness of the proposed adaptation for the weighting algorithm, tests on Pathological voice databases were carried out. Two databases are considered: Massachusetts Eye and Ear Infirmary Voice Laboratory voice disorders database distributed by Kay Elemetrics Corp. (KLM) and the Universidad Politécnica de Madrid voice disorders database (UPM).

2 Methods

Variable selection consist basically on selecting a subset of features from a larger set. This type of search is driven by some evaluation function that have been defined as the relevance of a given set [2]. When this process implies exhaustive search (binary selection), the relevance measure must take into account the dimensionality. Feature weighting relax this constraint allowing the calculation of

derivatives of the target function or the use mathematical programming tools to optimize these weights[10]. One important point is that the weights of the irrelevant variables should be as close as possible to zero, and the weights of the relevant features should be bounded. On the other hand. Feature extraction aims at encoding data efficiently for the problem at hand. In the case of linear projections for feature extraction, we can capitalize on this property to keep a fixed dimension (projected space) and assess the relevancy of the projected set. Thence, we can combine feature extraction methods with weighted data to maintain a fixed set size and accommodate weights in such a manner a relevance criterion is somehow maximized. Surprisingly, this consideration plays crucial role in guaranteeing that low weights will vanish.

2.1 Regularized Discriminant Analysis

RDA was proposed by [11] for small sample, high dimensional data sets to overcome the degradation of the discriminant rule. The aim of the linear variant of this technique is to find a projection of the space where scatter between classes is maximized maintaining the within scatter as minimal as possible. This is achieved by maximizing the ratio between the projected between class and within class matrices $J = \frac{|\mathbf{U}^T \boldsymbol{\Sigma}_B \mathbf{U}|}{|\mathbf{U}^T \boldsymbol{\Sigma}_W \mathbf{U}|}$, where \mathbf{U} is the projection matrix whose dimension is given by the number of classes (k) to be linearly separated, $\boldsymbol{\Sigma}_B$ is the between class matrix and can be associated to the dispersion of the mean values of each class, and $\boldsymbol{\Sigma}_W$ is the within class matrix and can be linked to the average class covariance matrix. The problem is defined as the constrained maximization of $|\mathbf{U}^T \boldsymbol{\Sigma}_B \mathbf{U}|$, that is, $\max_{\mathbf{U}} |\mathbf{U}^T \boldsymbol{\Sigma}_B \mathbf{U}|$, subject to $|\mathbf{U}^T \boldsymbol{\Sigma}_W \mathbf{U}| = 1$. Conditional extremes can be obtained from Lagrange multipliers; the solutions are the $k - 1$ leading generalized eigenvectors of $\boldsymbol{\Sigma}_B$ and $\boldsymbol{\Sigma}_W$ that are the leading eigenvectors of $\boldsymbol{\Sigma}_W^{-1} \boldsymbol{\Sigma}_B$. The need of regularization arises from small samples were $\boldsymbol{\Sigma}_W$ can not be directly inverted. The solution is rewritten as:

$$(\boldsymbol{\Sigma}_W + \delta \mathbf{I})^{-1} \boldsymbol{\Sigma}_B \mathbf{U} = \mathbf{U} \Lambda. \quad (1)$$

After weighting data, that is $\mathbf{X}\mathbf{D}$ where \mathbf{D} is diagonal matrix with ideally zero entries corresponding to the irrelevant features, J becomes $J_{\mathbf{D}} = \frac{|\mathbf{U}^T \mathbf{D} \boldsymbol{\Sigma}_B \mathbf{D} \mathbf{U}|}{|\mathbf{U}^T \mathbf{D} \boldsymbol{\Sigma}_W \mathbf{D} \mathbf{U}|}$.

2.2 Variable Weighting and Relevance Criterion

Data will be projected onto a fixed dimension subspace. For WRDA the fixed dimension should range between 1 to $k - 1$; being k the number of classes. This consideration depends on the distribution of the classes within the subspace. The relevance of a given weighted projection of a fixed dimension is evaluated by a separability measure. The search function will fall in some local maximum of the target function. The parameter to be optimized is the weight matrix \mathbf{D} , and selected criterion is the ratio traces of the aforementioned within and between matrices; this criterion is called J_4 in [12]. For weighted-projected data this measure is given by: $J_4(\mathbf{D}, \mathbf{U}) = \frac{\text{trace}(\mathbf{U}^T \mathbf{D} \boldsymbol{\Sigma}_B \mathbf{D} \mathbf{U})}{\text{trace}(\mathbf{U}^T \mathbf{D} \boldsymbol{\Sigma}_W \mathbf{D} \mathbf{U})}$ The size of \mathbf{U} is $(c \times p)$ and p denotes the

Algorithm 1. WRDA

-
- 1: Set dimension $p = k - 1$, being k the number of classes
 - 2: Normalize each feature vector to have zero mean and $\|\cdot\|_2 = 1$
 - 3: Start with some initial set of orthonormal vectors $\mathbf{U}^{(0)}$
 - 4: Compute $\mathbf{d}^{(r)}$ from solution given in section 2.2, and reweigh data.
 - 5: Compute the $\mathbf{U}^{(r)}$ from solution given in section 2.1.
 - 6: Compare $\mathbf{U}^{(r)}$ and $\mathbf{U}^{(r-1)}$ for some ε and return to step 3 if necessary.
-

We make use of the sum of absolute values of the diagonal elements of $(\mathbf{U}^{(r)})^T \mathbf{U}^{(r-1)}$, which are compared to the value obtained for $(\mathbf{U}^{(r-1)})^T \mathbf{U}^{(r-2)}$

fixed dimension, which is the number of projection vectors $\mathbf{U} = (\mathbf{u}_1 \mathbf{u}_2 \cdots \mathbf{u}_f)$. In order to apply matrix derivatives conveniently, we may want to rewrite \mathbf{D} in terms of its diagonal entries and represent it as a column vector \mathbf{d} . For this purpose, we can use the identity $\text{trace}(\mathbf{U}^T \mathbf{D} \mathbf{K} \mathbf{D} \mathbf{U}) = \mathbf{d}^T (\sum_{i=1}^p \mathbf{K} \circ \mathbf{u}_i \mathbf{u}_i^T) \mathbf{d}$ to rewrite the trace ratio can be rewritten in terms of Hadamard products as

$$J_4(\mathbf{d}) = \left\{ \mathbf{d}^T \left(\sum_{i=1}^p \boldsymbol{\Sigma}_B \circ \mathbf{u}_i \mathbf{u}_i^T \right) \mathbf{d} \right\} / \left\{ \mathbf{d}^T \left(\sum_{i=1}^p \boldsymbol{\Sigma}_W \circ \mathbf{u}_i \mathbf{u}_i^T \right) \mathbf{d} \right\} \quad (2)$$

This target function is quite similar to the one obtained for the regularized LDA. Therefore, the solution of \mathbf{d} with constrained L^2 norm is given by the leading eigenvector of

$$\left(\sum_{i=1}^p \boldsymbol{\Sigma}_W \circ \mathbf{u}_i \mathbf{u}_i^T + \delta \mathbf{I} \right)^{-1} \left(\sum_{i=1}^p \boldsymbol{\Sigma}_B \circ \mathbf{u}_i \mathbf{u}_i^T \right) \quad (3)$$

3 Functional Data Adaptation

Let \mathcal{X} be a set of objects that we want to classify into k different classes. Each observation $x \in \mathcal{X}$ is represented by a c -tuple of functions defined in the same domain, for instance $x = (f_1, f_2, \dots, f_c)$ and $f_l \in L_2[a, b]$, for $l = 1, \dots, c$. If we want to plug in the feature selection algorithm 1, presented in the previous sections, we need to define a way of quantifying the variation in the space of real square integrable functions $L_2[a, b]$. To this end, we will define the following key elements: **the expected function** $E[f_l(t)] = \int_{\mathbb{R}} f_l dF_l(f; t)$, **the expected squared norm** $E[\|f_l\|^2] = \int_a^b (\int_{\mathbb{R}} |f|^2 dF_l(f; t)) dt$, **the expected inner product** $E[\langle f_l, g_m \rangle] = \int_a^b (\int_{\mathbb{R}} (f_l g_m) dF_{lm}(f, g; t)) dt$; where $F_l(f; t)$ is the first-order probability distribution of l -th stochastic process represented by $f_l(t, x)$, and $F_{lm}(f, g; t)$ is the joint probability distribution of the l -th and m -th stochastic processes $f_l(t, x)$ and $f_m(t, x)$. In general, we just have access to a discrete version $f_l[t]$ of the function f_l ; besides, $F_l(f; t)$ and $F_{lm}(f, g; t)$ are unknown. The only available information is provided by the sample $\{(x_i, y_i)\}_{i=1}^n$, where $x_i = (f_{1i}[t], f_{2i}[t], \dots, f_{ci}[t])$ for $1 \leq t \leq T$, and $y_i \in \mathcal{Y} = \{1, 2, \dots, k\}$ is the class label for the observed x_i . Under these conditions we define the discrete empirical estimations of the expected

values: $E_{\text{emp}}[f_l[t]] = \frac{1}{n} \sum_{i=1}^n f_{li}[t]$, $E_{\text{emp}}[\|f_l\|^2] = \frac{1}{n} \sum_{i=1}^n \left(\sum_{t=1}^T |f_{li}[t]|^2 \right)$, and $E_{\text{emp}}[\langle f_l, f_m \rangle] = \frac{1}{n} \sum_{i=1}^n \left(\sum_{t=1}^T f_{li}[t] f_{mi}[t] \right)$. With this elements we can construct analogs for Σ_W and Σ_B . Notice that Algorithm 1, requires of a previous normalization of data. In the functional case this can be achieved by removing to each observation x_i the overall empirical mean of the sample, that is, $\hat{f}_{li}[t] = f_{li}[t] - E_{\text{emp}}[f_l[t]]$ for $1 \leq l \leq c$, and scaling the values that each function takes $\tilde{f}_{li}[t] = \frac{\hat{f}_{li}[t]}{\sqrt{nE_{\text{emp}}[\|\hat{f}_l[t]\|^2]}}$ for $1 \leq l \leq c$. From now and on, to ease the notation, we will assume that $f_{li}[i]$ is the normalized version of the function, that is $E_{\text{emp}}[f_l[t]] = 0$ and $E_{\text{emp}}[\|f_l\|^2] = 1/n$. For each j class, we define the empirical class-conditional expected values, which are computed as follows:

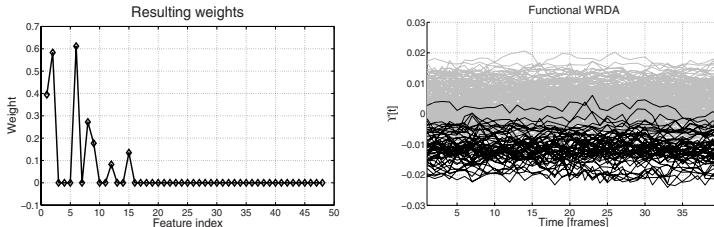
$$E_{\text{emp}}[T\{f_l\}|j] = \frac{1}{n_j} \sum_{x_i|y_i=j} T\{f_{li}\} \quad E_{\text{emp}}[T\{f_l, f_m\}|j] = \frac{1}{n_j} \sum_{x_i|y_i=j} T\{f_{li}, f_{mi}\} \quad (4)$$

where n_j is the number of observations that belong to j -th class, $T\{\cdot\}$ and $T\{\cdot, \cdot\}$ are functions over f . The j -th within class matrix Σ_{Wj} has the following elements $w_{jl,m} = E_{\text{emp}}[\langle f_l - E_{\text{emp}}[f_l|j], f_m - E_{\text{emp}}[f_m|j] \rangle|j]$, and the pooled within class matrix Σ_W can be computed as $\Sigma_W = \sum_{j=1}^k n_j \Sigma_{Wj}$. The between class matrix Σ_B elements are $b_{lm} = \sum_{j=1}^k n_j \langle E_{\text{emp}}[f_l|j], E_{\text{emp}}[f_l|j] \rangle$. Once Σ_W and Σ_B have been obtained, we can proceed with the rest of Algorithm 1, normally.

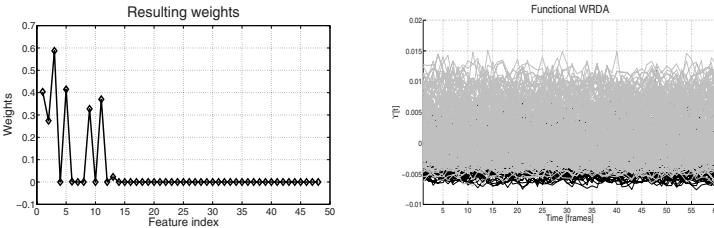
4 Experiments and Discussion

We refer to [13] for a complete description of KLM and UPM databases. Functional features correspond to windowed estimations of Harmonic Noise Ratio *HNR*, Normalized Noise Energy *NNE*, Glottal Noise Energy *GNE*, Energy, and 12 Mel Freq Cepstral Coefficients along with their first and second order derivatives obtained as in [14] for a total 48 functional features. These time vectors were clipped to a fixed number of windows moving from the central window to the sides, symmetrically. The fixed length of the sampled functions was 40 instances per functional feature in KLM, and 60 in UPM. The preliminary analysis consists on finding a set p of canonical functions resulting from a linear combination of the original set of c functional features using the Functional WRDA algorithm, $\Upsilon_{ji}[t] = \sum_{l=1}^c \alpha_{jl} f_{li}[t]$. where α_{jl} is obtained from the entries of the weighting and rotation matrices $\mathbf{D} = \text{diag}(d_1, d_2, \dots, d_l)$ and $\mathbf{U} = (\mathbf{u}_1 \mathbf{u}_2 \dots \mathbf{u}_p)$, and i is the index for the i -th observation.

In the two class case, which is our case Pathological vs Normal, the set of canonical functions reduces to a single function, that is, $\mathbf{U} = \mathbf{u}_1$. The regularization parameter δ introduced in equations (1) and (3) was set to 0.137 for KLM and 0.12 for UPM. Some graphical results for both databases are depicted in Figure 1. Right plots form Figures 1(a) and 1(b) present the weighted linear combination of the original functional features (canonical function) for KLM and UPM databases. In both databases most of the zero weights correspond to the first and second derivatives of the original features. The right plots show



(a) Massachusetts Eye and Ear Infirmary Voice Laboratory voice disorders database (KLM)



(b) Universidad Politécnica de Madrid voice disorders database (UPM)

Fig. 1. Weights and resulting canonical functions for KLM and UPM databases. Gray lines are canonical functions from pathological examples and black lines are from normal observations. The left plots are the resulting weights for each one of the functional features. The first 16 indexes are the short term energy and MFCC features. Notice that indexes from 17 to 48 obtained zero weights; these indexes correspond to first and second order derivatives of the functional features.

the resulting canonical functions from the whole set of examples for the two databases.

It is possible to use these functional features to perform classification with a kernel classifier or a distance based classifier by simply computing the inner product or the Euclidean distance between pairs of observations that now are represented by a canonical function. In here, we use Functional PCA (FPCA)[6] to embed this canonical function into a smaller dimension Euclidean space and then perform discrimination with a pooled covariance matrix classifier whose decision function is linear. This approach is equivalent to Kernel PCA [15] using the inner product $k(x_i, x_j) = \langle Y_{1i}[t], Y_{1j}[t] \rangle = \sum_{l=1}^c \sum_{r=1}^c \alpha_{1l} \alpha_{1r} \langle f_{li}[t], f_{rj}[t] \rangle$. Figures 2(a) and 2(b) display the clustered classes and how the first principal component may suffice for accurate classification of the sample. Moreover, the shape of the first principal functions and how points are distributed in the embedding suggest a particular phenomenon. The first principal function for both databases is approximate constant, so the inner product between the canonical function and the first PC is equivalent to a time average of the canonical function, which in turn is a sum of time averages of the selected original functional features (features with non-zero weights). PCA result seem to coincide with a LDA projection; a particular situation when both methods coincide is for a two class problem where the within class covariance functions are isotropic

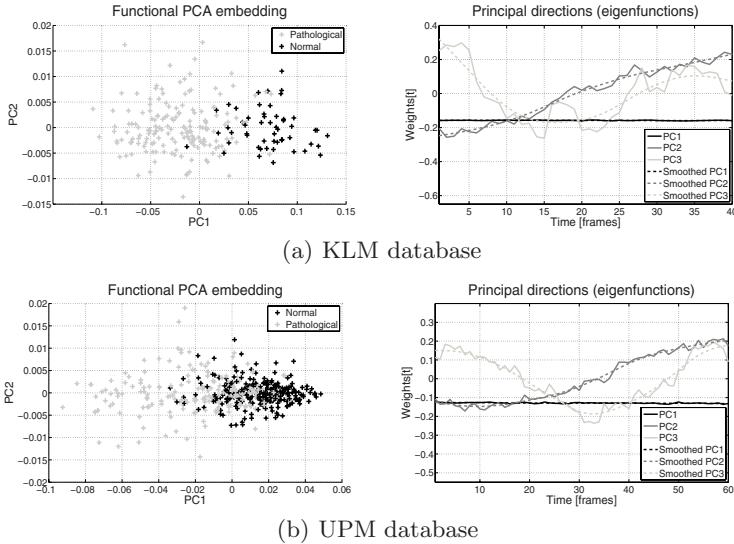


Fig. 2. Functional PCA embedding using the first two principal components and principal functions for KLM and UPM databases. Right plots are the principal functions for both databases. Notice how the functions are very similar, even though, the origin of the databases differ. The overlap of classes is higher for UPM database; this might be due to the larger diversity in the pathologies for this database.

approximately equal with a significant difference between their means. At the same time we carry out FPCA over the whole set of original functions. We employ a linear classifier using a pooled covariance matrix. Table 1, exhibit the Leave-One-Out (LOO) training and test errors for 1 to 3 principal components after applying the proposed functional feature selection process. This values are contrasted with the LOO training and test errors for 1, 10, and 20 principal components obtained from FPCA of the normalized original data. Our method conveys almost the same error estimate when varying the number of components. In the second case, we obtain incremental performance on training, but it should be noted that as dimensionality grows, so does the confidence interval.

Table 1. LOO training and test errors for FPCA after functional WRDA and FPCA for normalized original data. Errors are remarkably stable for the proposed method.

Database		FWRDA and FPCA			FPCA		
		1PC	2PCs	3PCs	1PC	10PCs	20PCs
KLM	train	9.09	9.23	8.15	13.03	7.78	7.0
	test	9.95	10.41	10.41	12.22	8.14	7.24
UPM	train	24.65	25.08	24.65	40.89	25.21	22.79
	test	25.91	26.14	25.91	40.91	27.27	25.00

5 Conclusions

We have presented a functional feature selection criterion based on weighting variables followed by a projection onto a fixed dimension subspace. Results showed how reducing dimensionality benefits the overall performance of the inference system. The canonical function devised from the application of our method was decomposed using FPCA, an interesting result of this analysis is that time averages can provide the necessary information to carry out successful classification. It is also important to highlight that the set of functional features selected for each of the databases is very similar. Although, both databases are voice disorder databases, their origins are quite different. The similarity of the results is also confirmed by observing the principal functions for both databases.

References

1. John, G.H., Kohavi, R., Pfleger, K.: Irrelevant features and the subset selection problem. In: ICML (1994)
2. Blum, A.L., Langley, P.: Selection of relevant features and examples in machine learning. In: AI, vol. 97(1-2) (1997)
3. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. In: JMLR (2003)
4. Yu, L., Liu, H.: Efficient feature selection via analysis of relevance and redundancy. In: JMLR (2004)
5. Wolf, L., Shashua, A.: Feature selection for unsupervised and supervised inference: the emergence of sparsity in a weighted-based approach. In: JMLR (2005)
6. Ramsay, J., Silverman, B.: Functional Data Analysis, 2nd edn. Springer, Heidelberg (2005)
7. Jolliffe, I.T.: Principal Component Analysis, 2nd edn. Springer, Heidelberg (2002)
8. Ferraty, F., Vieu, P.: Nonparametric Functional Data Analysis. Springer, Heidelberg (2006)
9. Sánchez, L., Martínez, F., Castellanos, G., Salazar, A.: Feature extraction of weighted data for implicit variable selection. In: CAIP. Springer, Heidelberg (2007)
10. Bradley, P.S., Mangasarian, O.L., Street, W.N.: Feature selection via mathematical programming. INFORMS Journal on Computing 10 (1998)
11. Friedman, J.H.: Regularized discriminant analysis. Journal of the American Statistical Association (1989)
12. Webb, A.R.: Statistical Pattern Recognition, 2nd edn. John Wiley & Sons, Chichester (2002)
13. Daza, G., Arias, J., Godino, J., Sáenz, N., Osma, V., Castellanos, G.: Dynamic feature extraction: An application to voice pathology detection. Intelligent Automation and Soft Computing 15(4) (2009)
14. Godino-Llorente, J.I., Gómez-Vilda, P., Blanco-Velasco, M.: Dimensionality reduction of pathological voice quality assesment system based on gaussian mixtures models and short-term cepstral parameters. IEEE Transactions on Biomedical Engineering 53(10), 1943–1953 (2006)
15. Schölkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, December 2002. MIT Press, Cambridge (2002)