

カーネル部分空間法

鷺沢 嘉一[†] 山下 幸彦^{††}

[†](独) 理化学研究所 脳科学総合研究センター 〒 351-0198 埼玉県和光市広沢 2-1

^{††} 東京工業大学 〒 152-8552 東京都目黒区大岡山 2-12-1

E-mail: [†]washizawa@brain.riken.jp, ^{††}yamasita@ide.titech.ac.jp

あらまし カーネル部分空間法とその正則化法, 抑制型への拡張について解説をする.

キーワード カーネル部分空間法, 主成分分析, KL 変換, KL 部分空間, CLAFIC 法, 相対 KL 変換, マーサーカーネル, サポートベクタマシン, カーネル主成分分析, カーネル標本空間射影, カーネル相対主成分分析, 正則化, 打ち切り特異値分解, Tikhonov 正則化, 縮小ランクウィーナーフィルタ

A family of kernel subspace classifiers

Yoshikazu WASHIZAWA[†] and Yukihiro YAMASHITA^{††}

[†] Brain Science Institute, RIKEN, 2-1, Hirosawa, Wako-shi, Saitama, 351-0198, Japan

^{††} Tokyo Institute of Technology, 2-12-1, Ookayama, Meguro-ku, Tokyo 152-8552 Japan

E-mail: [†]washizawa@brain.riken.jp, ^{††}yamasita@ide.titech.ac.jp

Abstract We present a family of kernel subspace classifiers and their extensions. There are two kinds of regularization methods and they have a degree of freedom about a null space of the operator in the family. Regularization is important to avoid the over-fitting problem in the machine learning theory. The applicable null space of an operator is able to suppress the effect of the other classes. We describe the details and differences of these classifiers and show experimental results.

Key words Kernel subspace classifier, class feature information compression (CLAFIC) method, relative Karhunen-Loève transform, Mercer kernel, support vector machine (SVM), Kernel principal component analysis (KPCA), kernel sample space projection classifier (KSP), kernel relative Karhunen-Loève transform (KRKLT), kernel relative principal component analysis (KRPCA), truncated singular value decomposition (TSVD), Tikhonov regularization, reduced rank Wiener filter

1. Introduction

Support vector machines have been developed since the early of 1990s and they showed very high generalization capability [1][2][3][4][5]. Their achievements are supported by the principle that maximizes the minimum margin and the (Mercer) kernel method. In the kernel method, an input vector $f \in \mathbb{R}^d$ is mapped from an input space to a higher or an infinite dimensional space \mathcal{F} called the feature space or the linearization space by a non-linear mapping $\Phi : \mathbb{R}^d \rightarrow \mathcal{F}$. Then the classifier is constructed and an unknown input vector is classified in the feature space. Instead of calculating Φ , a function $k(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ which satisfies the equation

$$k(f_1, f_2) = \langle \Phi(f_1), \Phi(f_2) \rangle, \quad (1)$$

is used. Mercer kernel function satisfies the eq. (1)[6][7][8]. Using Mercer kernel function, we can calculate an inner product in \mathcal{F} without care of the form of Φ . In other words, Φ is defined auto-

matically if we define Mercer kernel function $k(\cdot, \cdot)$.

Schölkopf et al. introduced another possibility of Mercer kernel function by solving an eigenvalue problem of a correlation matrix in the feature space [9]. This result was applied to the kernel principal component analysis (KPCA) and used for non-linear representation or image restoration [10]. Kernel methods are also applied to the Fisher discrimination [11][12], the independent component analysis (ICA) [13] or a Wiener filter [14][15][16].

Subspace methods are used widely in an industrial area [17]. A class feature information compression (CLAFIC) method is one of the subspace classifiers. It classifies unknown input pattern by comparing projection norms of an unknown input vector onto Karhunen-Loève (KL) subspaces which are constructed by the samples of each class [18][19].

Yamashita et al. extended a Karhunen-Loève transform (KLT) [20][21] which is almost equivalent to a principal component analysis (PCA) [22][23] to the relative KLT (RKLT) or the relative PCA

(RPCA) [24] [25]. It was also applied to pattern classifier [26] [27] [28]. We call this classifier RKLTL method. In RKLTL method, features of other classes are considered as noise against an objective class. Under this optimization problem, an optimal subspace for the classification is obtained. We describe details in Section 4.

On the other hand, Tsuda and Maeda et al. applied kernel method to CLAFIC method using results of Schölkopf et al. [29] [30] [31] [32]. We call this method KPCA. KPCA shows much higher accuracy rate than CLAFIC method. Furthermore its computational complexity in the multi-class classification problem is lower than SVM. We describe it in detail in Section 6. 1.

Hikida et al. and Washizawa et al. integrated KPCA and RKLTL, and proposed kernel relative KLT (KRKLT) or kernel relative PCA (KRPCA) [33] [34] [35]. Washizawa et al. also proposed a kernel based pattern classification method called kernel sample space projection (KSP) method [36] [37]. An operator of KSP is an orthogonal projection operator onto the kernel sample space which is spanned by samples mapped to high dimensional feature space \mathcal{F} . They also introduced some regularization to KSP. KPCA can be interpreted as the regularized version of KSP by truncated singular value decomposition (TSVD). Furthermore they extended KSP to suppressed KSP (SKSP) whose operator which extracts the feature of a class is an oblique projection along to the kernel sample space of the other classes onto kernel sample space of the objective class [38] [39]. KRPCA can be also interpreted as the regularized version of SKSP. We summarize these methods in Table 1.

Table 1 Summary of Kernel Subspace Methods

Regularization	Input space	Feature space	
	TSVD	TSVD	Tikhonov
Orthogonal Projection	CLAFIC	KPCA	KSP
Non-orthogonal transform	RKLT	KRKLT	SKSP

In Section 8., we show some experimental results of kernel subspace methods.

Table 2 denotes notations and symbols used in this paper. Except as otherwise noted, capital alphabets denote matrices and operators, lower-case alphabets denote vectors and functions, and Greek alphabet characters denote scalars.

2. Pattern recognition and discriminant function

The objective of pattern recognition is to classify unknown input pattern $f_x \in \mathbb{R}^d$ correctly, i.e, obtain applicable function $y(f_x)$ maps from input space \mathbb{R}^d to a set of class (or category) labels. Let \mathcal{K} be a set of class labels. We often use discriminant functions $d_k(f_x)$ ($k = 1, \dots, n \in \mathcal{K}$) to represent function $y(f_x)$.

Definition 1 (Discriminant function). Functions $d_k(f_x)$ ($k = 1, \dots, n \in \mathcal{K}$) which measure the similarity between a class k and unknown input pattern f_x are called discriminant functions. The class label $y(f_x)$ is obtained as

Table 2 Notation and symbols	
Φ	Non-linear mapping to the feature space $\mathbb{R}^d \rightarrow \mathcal{F}$
$k(\cdot, \cdot)$	Mercer kernel function
$d_k(\cdot)$	Discriminant function of class k
A^*	Adjoint operator of A
A^\dagger	Moore-Penrose generalized inverse of A
$\langle \cdot, \cdot \rangle$	inner product
$a \otimes \bar{b}$	Neumann-Shatten product, $(a \otimes \bar{b})c = c\langle b, a \rangle$
e_i	i -th natural basis, $(e_i)_j = \begin{cases} 0 & (i \neq j) \\ 1 & (i = j) \end{cases}$
$\mathcal{B}(S_1, S_2)$	Set of bounded linear operator from S_1 to S_2
Ω_i	Set of samples of class i
f_n^i	n -th sample of Ω_i
$ \Omega_i $	Number of samples in Ω_i
Ψ_i	Set of samples should be suppressed except class i
g_n^i	n -th sample of Ψ_i
$ \Psi_i $	Number of samples in Ψ_i
S_i	$S_i = \sum_{j=1}^{ \Omega_i } \Phi(f_j^i) \otimes \bar{e}_j \in \mathcal{B}(\mathbb{R}^{ \Omega_i }, \mathcal{F})$, $e_j \in \mathbb{R}^{ \Omega_i }$
T_i	$T_i = \sum_{j=1}^{ \Psi_i } \Phi(g_j^i) \otimes \bar{e}_j \in \mathcal{B}(\mathbb{R}^{ \Psi_i }, \mathcal{F})$, $e_j \in \mathbb{R}^{ \Psi_i }$
U_i	$U_i = \sum_{j=1}^{ \Omega_i } \Phi(f_j^i) \otimes \bar{e}_j + \sum_{k=1}^{ \Psi_i } \Phi(g_k^i) \otimes \bar{e}_{ \Omega_i +k}$
K_{S_i}	Kernel Gram matrix of class i , $K_{S_i} = S_i^* S_i$
K_{U_i}	Kernel Gram matrix of all classes $K_{U_i} = U_i^* U_i$
f_x	Unknown input vector
$\mathcal{R}(A), \mathcal{N}(A)$	Range and null space of A respectively
$P_{\mathcal{R}(A)}$	Orthogonal projection operator onto $\mathcal{R}(A)$
$\ f\ $	l_2 norm $\ f\ = \sqrt{\langle f, f \rangle}$
$\ A\ _{\text{lub}}$	Operator norm $\ A\ _{\text{lub}} = \sup_{f \neq 0} \frac{\ Af\ }{\ f\ }$
$\ A\ _F$	Frobenius norm $\ A\ _2 = \sqrt{\text{tr}(A^* A)}$
I	identity matrix, identity operator

$$y(f_x) = \underset{k \in \mathcal{K}}{\text{argmax}} d_k(f_x). \quad (2)$$

Then the objective is to obtain applicable functions $d_k(f_x)$ ($k = 1, \dots, n$) that extract the intrinsic feature of classes. In some cases, argmax in the eq. (2) is replaced by argmin . Then we can express with argmax by adding a negative sign. The simple example of the discriminant function is $d_k(f_x) = -\|f_k - f_x\|$, where f_k is a prototype of class k .

Note that this kind of discriminant function differs from it of binary classifiers such as SVM or Fisher discriminant. In those cases, its sign denotes the class.

3. CLAFIC method

Let $\{f_1^i, f_2^i, \dots, f_{|\Omega_i|}^i\} \in \Omega_i$ be a set of samples of class i .

Definition 2 (CLAFIC [18] [17]). The discriminant function of CLAFIC method is defined as

$$d_k(f_x) = \|P_k f_x\|^2 \quad (3)$$

$$P_k = \underset{X: \text{rank}(X) \leq \eta}{\text{argmin}} \sum_{f \in \Omega_k} \|f - Xf\|^2. \quad (4)$$

Let a sample correlation matrix of class k be R_k , and its eigenvalue decomposition is given as

$$R_k = \sum_{f \in \Omega_k} f \otimes \bar{f} = \sum_{l=1}^d \lambda_l^k (u_l^k \otimes \bar{u}_l^k), \quad (5)$$

where the operator $(\cdot \otimes \bar{\cdot})$ is Neumann-Shatten product defined by $(a \otimes b)c = \langle c, b \rangle a$. It is equivalent as ab^T in a real finite dimensional space, however we use this notation consistently because we have to deal with infinite space when we introduce the kernel method.

Theorem 1. *Suppose that eigenvalues $\lambda_l^k, l = 1, 2, \dots, d$ are sorted in descending order in eq. (5). The solution of eq. (4) is a projection matrix [17] [19] [18]*

$$P_k = \sum_{j=1}^{\eta^k} (u_j^k \otimes \bar{u}_j^k). \quad (6)$$

The parameter η is obtained using cumulative proportion. However its accuracy is sensitive against η . If η is too small, feature of a class cannot extract enough. If η is too large, overlap of classes decreases its accuracy.

4. Relative Karhunen-Loève Transform (RKL) method

A feature extraction operator of CLAFIC method P extracts feature of a class. However if plural classes have similar feature, the operator extracts both feature and they cannot be discriminated. Let Ψ_k be a set of sample which should be suppressed with respect to class k . Relative Karhunen-Loève Transform (RKL) method overcome this problem using following optimization problem [24] [25]:

Definition 3. *The operator of RKL is defined as*

$$B_k = \operatorname{argmin}_{X: \operatorname{rand}(X) \leq \eta} \sum_{f \in \Omega_k} \|f - Xf\|^2 + \alpha \sum_{g \in \Psi_k} \|Xg\|^2, \quad (7)$$

where α is a parameter which control the effect of the second term.

The first term of eq. (7) is the same as CLAFIC method. The second term suppresses the feature of other classes. RKL is equivalent to CLAFIC when $\alpha = 0$, and equivalent to reduced rank Wiener filter when $\alpha = 1$ and f, g have no cross correlation [40].

Theorem 2. *Let $R_k = \sum_{f \in \Omega_k} (f \otimes \bar{f})$, $Q_k = \sum_{g \in \Psi_k} (g \otimes \bar{g})$. A^\dagger denotes a Moore-Penrose generalized inverse of the operator A [41]. Suppose that eigenvalue decomposition of $R_k(R_k + Q_k)^\dagger R_k$ is expressed as*

$$R_k(R_k + Q_k)^\dagger R_k = \sum_{j=1}^d \lambda_j^k (u_j^k \otimes \bar{u}_j^k), \quad (8)$$

and λ_j^k is sorted in descending order. Then the solution of RKL is given as

$$B_k = \sum_{j=1}^{\eta} (u_j^k \otimes \bar{u}_j^k) R(R + Q)^\dagger + W(I - (R + Q)(R + Q)^\dagger), \quad (9)$$

where W is an arbitrary matrix and I is an identity matrix.

Note that if $(R + Q)$ is full rank matrix, the second term is zero.

Since the feature extraction operator of RKL B_k is not orthogonal projector, there are two kinds of discriminant functions:

$$d_k^1(f_x) = \|B_k f_x\|^2 \quad (10)$$

$$d_k^2(f_x) = -\|f_x - B_k f_x\|^2. \quad (11)$$

In the case of CLAFIC method, these are equivalent. Eq. (11) shows higher accuracy rate at an experiment in [28].

5. Kernel sample space projection

Here, we introduce the kernel sample projection classifier (KSP) to describe kernel subspace method systematically. All of kernel subspace methods can be interpreted as the extension of KSP.

Let $\{f_1^i, f_2^i, \dots, f_{|\Omega_i|}^i\} \in \Omega_i$ be a set of samples of class i , $k(\cdot, \cdot)$ be a Mercer kernel function, and $\Phi: \mathbb{R}^d \rightarrow \mathcal{F}$ be a mapping led from the Mercer kernel function. At first, we introduce a kernel sample space which is spanned by samples mapped to feature space \mathcal{F} . Let an operator

$$S_i = \sum_{j=1}^{|\Omega_i|} \Phi(f_j^i) \otimes \bar{e}_j, \quad (12)$$

where $e_j \in \mathbb{R}^{|\Omega_i|}$ is a natural basis which has only one '1' element in j -th row and the rest elements are zero. If $\Phi(f_j^i)$ is a column vector in a finite dimensional space, S_i is a matrix expressed as $S_i = [\Phi(f_1^i) \ \Phi(f_2^i) \ \dots \ \Phi(f_{|\Omega_i|}^i)]$. A kernel sample space of class i is expressed as $\mathcal{R}(S_i)$, where $\mathcal{R}(A)$ denotes a range of A .

An operator $S_i^* \Phi(\cdot): \mathbb{R}^d \rightarrow \mathbb{R}^{|\Omega_i|}$ so-called the empirical kernel map [7] is reduced as

$$\begin{aligned} S_i^* \Phi(f) &= \sum_{j=1}^{|\Omega_i|} (e_j \otimes \overline{\Phi(f_j^i)}) \Phi(f) \\ &= \sum_{j=1}^{|\Omega_i|} \langle \Phi(f), \Phi(f_j^i) \rangle e_j \\ &= (k(f, f_1^i) \ k(f, f_2^i) \ \dots \ k(f, f_{|\Omega_i|}^i))^T, \end{aligned} \quad (13)$$

where A^* denotes the adjoint operator of A .

A (kernel) Gram matrix K_{S_i} is defined as

$$K_{S_i} = S_i^* S_i = \begin{bmatrix} k(f_1^i, f_1^i) & \dots & k(f_1^i, f_{|\Omega_i|}^i) \\ \vdots & \ddots & \vdots \\ k(f_{|\Omega_i|}^i, f_1^i) & \dots & k(f_{|\Omega_i|}^i, f_{|\Omega_i|}^i) \end{bmatrix} \quad (14)$$

Definition 4 (Kernel sample projection [37] [36]). *The discriminant function of KSP is expressed as*

$$d_i(f_x) = \|P_{\mathcal{R}(S_i)} \Phi(f_x)\|^2 = \langle S_i^* \Phi(f_x), K_{S_i}^\dagger S_i^* \Phi(f_x) \rangle, \quad (15)$$

where $P_{\mathcal{R}(S_i)} = S_i K_{S_i}^\dagger S_i^*$ is an orthogonal projector onto kernel sample space $\mathcal{R}(S_i)$.

Thus the similarity between class i and unknown input pattern f_x is measured by the projection norm onto the kernel sample space $\mathcal{R}(S_i)$.

In order to extend KSP, we redefine the feature extraction operator of KSP:

Definition 5 (Kernel sample space projection). *The operator of KSP is defined by following optimization problem.*

$$\begin{aligned} \min_{X_i} : \quad & J[X_i] = \frac{1}{|\Omega_i|} \sum_{f \in \Omega_i} \|\Phi(f) - X_i \Phi(f)\|^2 \\ \text{subject to:} \quad & \mathcal{N}(X_i) \supset \mathcal{R}(S_i)^\perp, \end{aligned} \quad (16)$$

where $\mathcal{N}(A)$ denotes the null space of A and $^\perp$ denotes an orthogonal complement space.

The constraint means that components which is not in samples are projected to the kernel sample space by X_i . Actually, in the CLAFIC or RKLTL methods are also needed this constraint. However in almost cases, since samples span whole space, this constraint has no effect.

Proposition 1. *The solution of the optimization problem (16) equals to $P_{\mathcal{R}(S_i)}$.*

Proposition 1 can be proved from results of the Appendix easily.

6. Regularization of KSP

KSP method can classify all train samples correctly if the kernel Gram matrix K_{S_i} is not singular. If we use mapping Φ that maps to an infinite functional space, e.g., Gaussian kernel function $k(f_1, f_2) = \exp(-\|f_1 - f_2\|^2/c)$, kernel Gram matrix K_{S_i} is always nonsingular unless the same samples exist. However, since there are noisy samples or outliers in a sample set, the generalization error is large because of the over-fitting problem. In order to avoid the over-fitting problem, we can introduce the regularization technique, e.g., soft margin techniques for the SVM or the Ada-Boost [42], weight decay parameter for the neural networks [43], or the ridge regression. In a field of the linear inverse problems, the regularization has been discussed for long time. The truncated singular decomposition (TSVD) and Tikhonov regularization are major techniques of the regularization [44] [45] [46] [47].

6.1 Kernel principal component analysis

KPCA can be interpreted as KSP introduced TSVD. We characterize its operator by an optimization problem.

Definition 6 (KPCA [9] [10]). *The operator of KPCA is defined by following optimization problem.*

$$\begin{aligned} \min_{X_i} : \quad & J[X_i] = \frac{1}{|\Omega_i|} \sum_{f \in \Omega_i} \|\Phi(f) - X_i \Phi(f)\|^2 \\ \text{subject to:} \quad & \mathcal{N}(X_i) \supset \mathcal{R}(S_i)^\perp, \quad \text{rank}(X_i) \leq \eta \end{aligned} \quad (17)$$

6.2 Regularized KSP

Another major regularization technique is Tikhonov regularization which is used in ridge regression in the area of the multivariate analysis [45] [44]. The over-fitting problem is caused by ill-condition of a feature extraction operator. One of the measure of the condition of operator is an operator (or a spectral) norm which is defined by

$$\|A\|_{\text{ub}} = \sup_{\|x\| \neq 0} \frac{\|Ax\|}{\|x\|} = \sup_{\|x\|=1} \|Ax\|. \quad (18)$$

If the operator norm of a feature extraction operator is large, small amount of components which have the specific direction dominate the value of the feature. Then the decision boundary become complex and the over-fitting problem appears. However, since it is difficult to suppress the operator norm in eq. (16), we use Frobenius norm defined by $\|A\|_F = \sqrt{\text{tr}(A^*A)} \leq \|A\|_{\text{ub}}$ instead of using operator norm.

Definition 7 (Regularized KSP [39] [38]). *Regularized KSP is defined as the solution of following optimization problem:*

$$\min_{X_i} : \quad J[X_i] = \frac{1}{|\Omega_i|} \sum_{f \in \Omega_i} \|\Phi(f) - X_i \Phi(f)\|^2 + \epsilon \|X_i\|_F^2 \quad (19)$$

$$\text{subject to:} \quad \mathcal{N}(X_i) \supset \mathcal{R}(S_i)^\perp,$$

where $\epsilon > 0$ is a regularization parameter which controls strength of regularization.

Theorem 3 (Solution of regularized KSP). *The solution of optimization problem (19) is*

$$P_{\mathcal{R}(S_i)}^\mu = S_i(K_{S_i} + \mu I)^{-1} S_i^*, \quad (20)$$

where $u = \epsilon|\Omega_i|$.

The proof of the theorem is in Appendix. We call regularized KSP just KSP from now.

6.3 Toy example

Here, we show a toy example in order to show the effect of regularization. Figure 1 shows the problem. Training vectors are 200 two dimensional vectors generated by uniform distribution in $[0, 500] \times [0, 500]$. Actual decision boundaries which are concentric circles shown in the figure discriminate samples to two classes shown by red circles and green crosses. The objective is the reconstructing the decision boundaries from samples.

Results are shown in Figure 2. We used Gaussian kernel function $k(x, y) = \exp(-\frac{\|x-y\|^2}{2.50^2})$. In the case that regularization is not used i.e., $\mu = 0$ or the rank of KPCA is maximum, decision boundary is complex, but all train samples can be classified correctly. On the other hand, in the case that $\mu = 1$ or rank equals to 20, decision boundaries are smooth, but some train samples are misclassified. GE denotes generalization error measured by percentage of pixels which are out of the actual decision boundary because samples are distributed uniformly. In both KPCA and KSP, applicable regularization decrease the generalization error even if some training samples are misclassified.

7. Suppression of the effect of other classes

KSP and KPCA can be extended like RKLTL. Let

$$U_i = \sum_{j=1}^{|\Omega_i|} \Phi(f_j^i) \otimes \bar{e}_j + \sum_{k=1}^{|\Psi_i|} \Phi(g_k^i) \otimes \bar{e}_{|\Omega_i|+k}. \quad (21)$$

Then definitions of Kernel RKLTL and Suppressed KSP is given as follows.

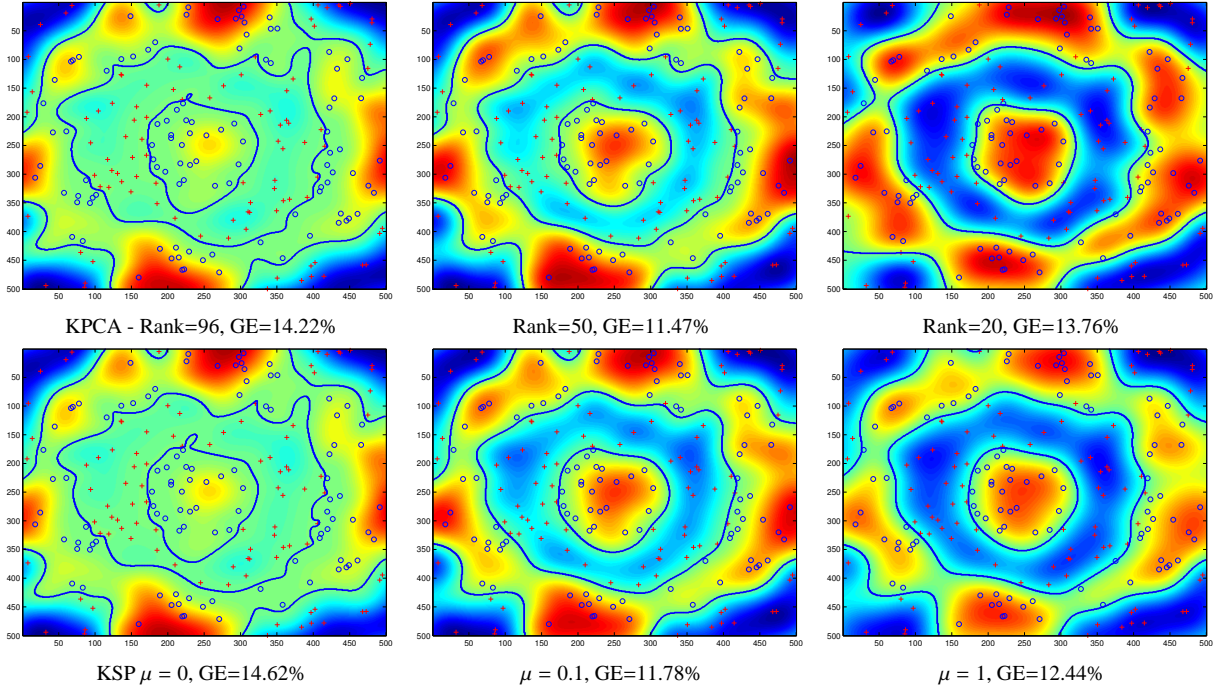


Figure 2 Results of toy example problem; Upper column: KPCA, Lower column: KSP. Blue line denotes estimated decision boundary and base color shows the value of feature. GE is generalization error i.e., percentage of pixels which are out of the actual decision boundary because samples are distributed uniformly.

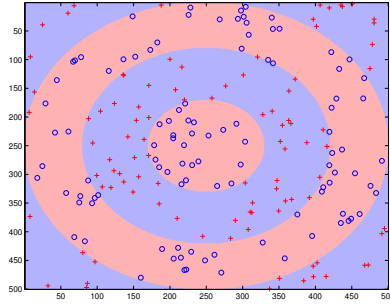


Figure 1 Train vectors of toy example problem: 200 train vectors are uniformly distributed. Actual decision boundaries are concentric circles.

Definition 8 (Kernel RKL (KRKL), Kernel RPCA (KRPCA) [35] [33] [34]). KRKL is defined by the solution of following optimization problem:

$$\min_{X_i} : J[X_i] = \frac{1}{|\Omega_i|} \sum_{f \in \Omega_i} \|\Phi(f) - X_i \Phi(f)\|^2 + \frac{\alpha}{|\Psi_i|} \sum_{g \in \Psi_i} \|X_i \Phi(g)\|^2 \quad (22)$$

subject to: $\mathcal{N}(X_i) \supset \mathcal{R}(U_i)^\perp$, $\text{rank}(X_i) \leq \eta$,

where α is a parameter which controls the strength of suppression.

Definition 9 (Suppressed KSP (SKSP) [39] [38]). SKSP is defined by the solution of following optimization problem:

$$\min_{X_i} : J[X_i] = \frac{1}{|\Omega_i|} \sum_{f \in \Omega_i} \|\Phi(f) - X_i \Phi(f)\|^2 + \frac{\alpha}{|\Psi_i|} \sum_{g \in \Psi_i} \|X_i \Phi(g)\|^2 + \mu \|X_i\|_F^2 \quad (23)$$

subject to: $\mathcal{N}(X_i) \supset \mathcal{R}(U_i)^\perp$,

Theorem 4 (Solution of KRKL). We omit an index i which denotes class label to simplify expression. Let $\Lambda_0 = \begin{bmatrix} I_{|\Omega|} & \mathbf{0}_{|\Omega||\Psi|} \\ \mathbf{0}_{|\Psi||\Omega|} & \mathbf{0}_{|\Psi|} \end{bmatrix}$,

$\Lambda = \begin{bmatrix} \frac{1}{\sqrt{|\Omega|}} I_{|\Omega|} & \mathbf{0}_{|\Omega||\Psi|} \\ \mathbf{0}_{|\Psi||\Omega|} & \sqrt{\frac{\alpha}{|\Psi|}} I_{|\Psi|} \end{bmatrix}$, $K_U = U^* U$, $A = K_U^{1/2} \Lambda_0 \Lambda$, and v_i be i -th eigen vector of $A^\top A$, where suppose that corresponding eigenvalues are sorted in descending order. Then one of the solutions of the optimization problem (22) is given as

$$X^{KRPCA} = U(K_U^{1/2})^\dagger A \sum_{i=1}^{\eta} (v_i \otimes \bar{v}_i) \Lambda^{-1} K_U^\dagger U^*. \quad (24)$$

Proposition 2. If K_U is nonsingular, X^{KRPCA} is a projector.

The proofs of these theorem and proposition appear in Appendix.

Theorem 5 (Solution of SKSP). The solution of the optimization problem (23) is given as

$$\tilde{P}_{\mathcal{R}(S)}^\mu = U \Lambda_0 (K_U + \mu \Lambda^{-2})^{-1} U^*. \quad (25)$$

Proposition 3. If K_U is nonsingular and $\mu = 0$, $\tilde{P}_{\mathcal{R}(S)}^\mu$ is a projector onto $\mathcal{R}(S_i)$.

Proposition 4. If K_U is nonsingular and $\mu = 0$, we have

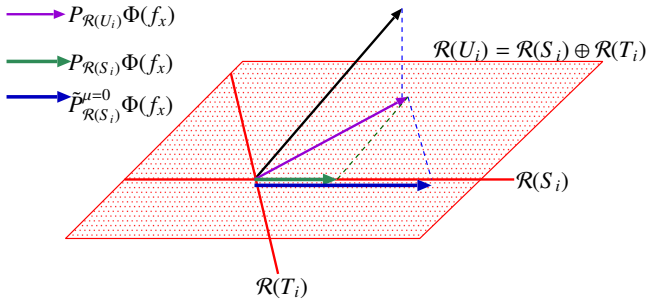


Figure 3 Suppressed kernel sample space projection

Table 3 Results of the handwritten digits classification problem

Method	Parameter	Error rate[%]
KSP with Polynomial kernel	$d = 8$	2.29
KSP with Gaussian kernel	$\sigma = 7$	2.25
KPCA with Polynomial kernel	$d = 7, \text{rank} = 460$	2.36
KPCA with Gaussian kernel	$\sigma = 10, \text{rank} = 500$	2.32
SKSP with Gaussian kernel	$\sigma = 8$	1.79
3-NN	–	2.4
Polynomial SVM	$d = 4$	1.1

$$\tilde{P}_{\mathcal{R}(S)}^{\mu} P_{\mathcal{R}(U)} = \tilde{P}_{\mathcal{R}(S)}^{\mu} \quad (26)$$

$$P_{\mathcal{R}(U)} \tilde{P}_{\mathcal{R}(S)}^{\mu} = \tilde{P}_{\mathcal{R}(S)}^{\mu}, \quad (27)$$

where $P_{\mathcal{R}(U)}$ is the orthogonal projector onto $\mathcal{R}(U)$.

Proposition 5. If K_U is nonsingular and $\mu = 0$, $\tilde{P}_{\mathcal{R}(S)}^{\mu} v = 0$ for all $v \in \mathcal{R}(T)$, where $T = \sum_{k=1}^{|\Psi|} \Phi(g_k) \otimes \bar{e}_k$.

The proofs of these theorem and propositions appear in Appendix.

We show the sketch of SKSP in Fig. 3. From Propositions 3, 4, and 5, $\tilde{P}_{\mathcal{R}(S)}^{\mu} \Phi(f_x)$ can be considered as follows. At first, $\Phi(f_x)$ is orthogonally projected onto $\mathcal{R}(U)$, then it is projected onto $\mathcal{R}(S)$ along $\mathcal{R}(T)$. The similarity between f_x and Ω against Ψ is given as $\|\tilde{P}_{\mathcal{R}(S)}^{\mu} \Phi(f_x)\|$. If $\Psi = \phi$, $\tilde{P}_{\mathcal{R}(S)}^{\mu} = P_{\mathcal{R}(S)}$. Thus SKSP is an extension of KSP.

8. Computational Simulation

8.1 Handwritten digits classification

In order to compare abilities of kernel subspace classifiers, we used a handwritten digit database ‘MNIST’ provided by the U.S. National Institute of Standard and Technology. It is consisted by 60,000 characters for training and 10,000 characters for testing. Each character is 28x28 pixels and 256 gray scale image.

We show the classification error rate with the parameters which show the lowest error rate in Table 3. The results of the 3-NN (3-Nearest Neighborhood) and the Polynomial SVM is referred from [48] and [5] respectively.

8.2 Binary classification problem

We employ several practical data sets used in [42], [11] and [49]^(*). All the data sets we use here are binary classification prob-

lems and consist of 100 or 20 realizations.

We use Gaussian kernel function as a kernel function. The parameter of kernel function and the regularization is fixed for all sets. If there are identical samples, we added only one of them to learning set. We use all samples in the other class for Ψ , since the learning set is not large.

The mean test error rates and their standard deviations are described in Table 4. The results except for KSP and SKSP are referred from the papers above.

Table 4 Mean test error rates and their standards deviations (AB Reg: Regularized AdaBoost, KFD: kernel fisher discriminant). The best method is written in bold face and the second best is emphasized.

dataset	SKSP	KSP	AB Reg	SVM	KFD
Banana	10.4 ± 0.5	10.4 ± 0.5	10.9 ± 0.4	11.5 ± 0.7	10.8 ± 0.5
Breast-cancer	26.0 ± 4.6	29.7 ± 4.5	26.5 ± 4.5	26.0 ± 4.7	24.5 ± 4.6
Diabetes	23.0 ± 1.6	24.5 ± 1.9	23.8 ± 1.8	23.5 ± 1.73	23.2 ± 1.6
Flare-solar	37.2 ± 4.5	39.1 ± 2.4	34.2 ± 2.2	32.4 ± 1.8	33.2 ± 1.7
German	23.4 ± 2.1	31.3 ± 2.5	24.7 ± 2.4	23.6 ± 2.1	23.7 ± 2.2
Heart	15.8 ± 3.1	15.4 ± 3.3	16.5 ± 3.5	16.0 ± 3.3	16.1 ± 3.4
Image	2.8 ± 0.4	2.9 ± 0.5	2.7 ± 0.6	3.0 ± 0.6	4.8 ± 0.6
Ringnorm	18.0 ± 2.3	19.9 ± 1.8	1.6 ± 0.1	1.7 ± 0.1	1.5 ± 0.1
Splice	11.2 ± 0.7	12.6 ± 0.7	9.5 ± 0.7	10.9 ± 0.7	10.5 ± 0.6
Thyroid	4.0 ± 2.3	4.2 ± 2.3	4.6 ± 2.2	4.8 ± 2.2	4.2 ± 2.1
Titanic	29.4 ± 10.3	28.3 ± 9.4	22.6 ± 1.2	22.4 ± 1.0	23.3 ± 2.1
Twonorm	2.4 ± 0.1	2.3 ± 0.1	2.7 ± 0.2	3.0 ± 0.2	2.6 ± 0.2
Waveform	9.6 ± 0.4	11.2 ± 0.6	9.8 ± 0.8	9.9 ± 0.4	9.9 ± 0.4
# of bold	5	3	2	2	2
# of emph.	4	1	3	2	3

Consequently, SKSP classifier shows the lowest error rates among those methods in many problems, and SKSP outperformed KSP in most of problems. We can say SKSP can suppress the effect of features in the other class, and can extract important features.

9. Discussion

9.1 Comparison of Regularization

As mentioned above, KPCA and KRPCA use the TSVD, and KSP and SKSP use the Tikhonov regularization. We summarize the difference between the TSVD and the Tikhonov regularization in Table 5.

Table 5 Comparison of regularization

	TSVD	Tikhonov
Parameter	Discrete	Continuous
Computational complexity in classification stage	Low	High
Computational complexity in constructing stage	High	Low
Additive Learning	△	○

Parameters of KPCA and KRPCA is a rank of the operator. If the number of samples are finite, the rank of the operator is discrete.

(*) : All the data sets are downloaded from

‘<http://ida.first.fraunhofer.de/projects/bench/benchmarks.htm>’.

While regularization parameters of KSP and SKSP is continuous. If the problem is sensitive against these parameter, the Tikhonov regularization is better to control the degree of regularization. For example, if samples are quite few, since the ranks of KPCA or KRPCA equal to the number of samples, it is difficult to control the degree of the regularization.

Since KPCA and KRPCA can reduce the rank of operator, computational complexity in classification stage is lower than Tikhonov methods. While computational complexity of KPCA and KRPCA in constructing stage is high because KPCA and KRPCA require eigenvalue decomposition in constructing stage. KSP and SKSP require the inverse operation which is lower computational complexity than the eigenvalue decomposition.

We can apply additive learning to KSP and SKSP easily by using Sherman-Morrison-Woodbury formula and its extension [50].

Proposition 6 (Additive learning of KSP). *Let $\{f_1, \dots, f_N, f_{N+1}\}$ be samples, $S = \sum_{i=1}^N \Phi(f_i) \otimes \bar{e}_i$, $S' = \sum_{i=1}^{N+1} \Phi(f_i) \otimes \bar{e}'_i$, where e_i and e'_j are natural basis in \mathbb{R}^N and \mathbb{R}^{N+1} respectively. Suppose that an inverse matrix of kernel Gram matrix of S , $K_S^{-1} = (S^*S)^{-1}$ is known and new sample f_{N+1} is given. Then an inverse of new kernel Gram matrix $K_{S'}^{-1}$ is given as*

$$K_{S'}^{-1} = \begin{bmatrix} K_S^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \frac{1}{\tau} tt^T, \quad (28)$$

where

$$t = \begin{bmatrix} K_S^{-1} S^* \Phi(f_{N+1}) \\ -1 \end{bmatrix}$$

$$\tau = k(f_{N+1}, f_{N+1}) - \langle S^* \Phi(f_{N+1}), K_S^{-1} S^* \Phi(f_{N+1}) \rangle.$$

Furthermore we can introduce the Gaussian elimination to solve an inverse operation. The computational complexity of Gaussian elimination is extremely low. However we cannot store the inverse matrix. Thus it is useful in the case that the number of classification is low e.g., cross validation or leave one out.

9.2 Suppression of other classes effects

From the computational simulations and experiments in [35], SKSP and KRPCA show higher accuracy rate than KSP and KPCA. Thus suppression of effects of other classes improves the accuracy rate. However suppression increases computational complexity simultaneously. Using Sherman-Morrison-Woodbury formula, SKSP requires inverse operations of $|\Omega| \times |\Omega|$ matrix and $|\Psi| \times |\Psi|$ matrix, and several matrix products.

SKSP and KRPCA differ from ordinary binary classifiers e.g., SVM or Fisher discriminant, in that they not need to use all samples of other classes because KSP and KPCA can extract feature of other classes itself. In actual problems, KSP and KPCA can extract the almost enough features. Thus we do not have to use all samples of other classes. Only samples which are similar to Ω_i have to be included in Ψ_i . Since the similarity of an input vector is evaluated by the projection norm onto $\mathcal{R}(S_i)$ in KSP, it sufficient that samples of

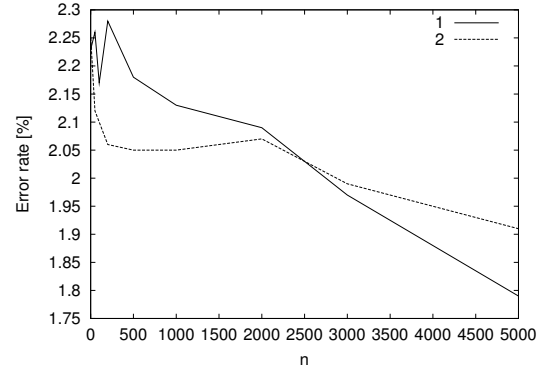


Figure 4 Error rate of SKSP: vertical axis: error rate [%], horizontal axis: number of suppression samples

which projection norms are large are included in Ψ_i . For example, we can use following criterion to choose suppression set:

$$t(g_i) = \|P_{\mathcal{R}(S_i)} \Phi(g_i)\|^2, \quad (29)$$

where $P_{\mathcal{R}(S_i)}$ is an operator of KSP. In this case, samples which have large t should be used.

Unlike KSP and KPCA which are orthogonal projectors, SKSP and KRPCA are not orthogonal projector. Thus there are two kinds of discriminant function:

$$d_1(f_x) = \|X\Phi(f_x)\|^2 \quad (30)$$

$$d_2(f_x) = \|\Phi(f_x) - X\Phi(f_x)\|^2. \quad (31)$$

Figure 4 shows the relation between error rates of handwritten digit classification problem and the number of suppression samples. ‘1’ and ‘2’ mean eq. (30) and eq. (31) respectively. Suppressing samples are chosen by the criterion (29).

From the figure, when the number of suppression samples is small, discriminant function (31) shows better accuracy. While when the number of suppression samples is large, discriminant function (30) shows better accuracy. However there is no theoretical evidence that which is better.

9.3 Comparison with other classification method

The remarkable difference between SKSP and SVM or KFD is that SKSP is a quadratic discriminant function, while SVM and KFD are linear discriminant functions in the feature space. In the case of classification in the input space (not using a kernel method), generally quadratic classifiers shows higher performance than linear ones, because quadratic classifiers have more degree of freedom than linear ones. Thus if they are extended by kernel methods, quadratic discriminants will show higher performance in feature space.

In SVM, a separating hyperplane is determined by a few samples called support vectors (SVs). The separating hyperplane depends only on samples around boundary and does not depend on other samples or its distribution. If there are noisy samples or outliers in learning samples, a separating hyperplane is deteriorated by

them because they become SVs in high probability. Thus SVM is not robust fundamentally, even if the regularizing methods e.g. soft margin ([5]) or ν -SVM ([51]) are used.

In general, a classifier has trade off between robustness and sparseness about the number of samples. The computational cost of SVM in recognition stage is low because its solution is sparse. Let k and s be computational costs of calculating a kernel function and a multiplication respectively. Let L and L_{SV} be the number of learning samples and SVs respectively. Then main computational cost in recognition stage are given as

$$\text{SVM} : (k + s)L_{SV}$$

$$\text{KFD} : (k + s)L$$

$$\text{SKSP} : sL^2 + (k + s)L.$$

Note that generally, $s < k$ and $L_{SV} < L$. Only SKSP has a 2nd-order term with respect to L , because it is a quadratic discriminant in feature space. But as mentioned above, all samples belonging to other classes do not have to be included. Thus we can decrease L and computational cost.

In learning stage, SVM costs a lot of time because it requires to solve a quadratic optimization problem. KFD also requires to solve it ([52]). On the other hand the solution of SKSP is given as a closed form with an inverse operation and multiplications. Moreover as stated above, all samples belonging to other classes do not have to be used. The calculation steps for inverse of matrices and multiplication are $O(L^3)$. Generally, inverse problems are easier than quadratic optimization problems. Thus the computational cost of KSP or SKSP is lower than SVM in learning stage.

Moreover in the case that there are a large number of leaning samples, we can introduce “multi-templates method” to SKSP or KSP easily, because they are not a two-class classifier. In multi-templates method, sub-classes in a class is prepared and an input vector is classified to a sub-class. It is no use for two-class classifiers (e.g. SVM or KFD) to realize this method, because they require to use all samples of all classes.

10. Conclusion

We exposit a family of kernel subspace classifiers and their extensions. They show adequate performance in computational simulations. Especially in the case of multi-class classification problems, they will be useful because it is high computational complexity to construct binary classifiers.

However there are still open problems that are how to obtain the parameter, how to choose suppression samples, and discriminant function of SKSP or KRPCA.

This works have been supported by JSPS Grand-in-Aid for Scientific Research 18300057.

References

[1] V. N. Vapnik: “Statistical Learning Theory”, Wiley, New-York (1998).

[2] V. N. Vapnik: “The Nature of Statistical Learning Theory”, Springer (1995).

[3] B. Schölkopf, C. J. C. Burges and A. J. Smola: “Advances in kernel methods: Support vector learning”, MIT Press (1999).

[4] C. J. C. Burges: “A tutorial on support vector machines for pattern recognition”, Data Mining and Knowledge Discovery, **2**, pp. 121–167 (1998).

[5] C. Cortes and V. Vapnik: “Support-Vector Networks”, Machine Learning, **20**, 3, pp. 273–297 (1995).

[6] J. Mercer: “Functions of positive and negative type, and their connection with the theory of integral equations.”, Trans. Lond. Phil. Soc. (A), **209**, pp. 415–446 (1909).

[7] B. Schölkopf, S. Mika, C. Burges, P. Knirsch, K.-R. Müller, G. Rätsch and A. Smola: “Input space vs. feature space in kernel-based methods”, IEEE Transactions on Neural Networks, **10**, 5, pp. 1000–1017 (1999).

[8] B. Schölkopf and A. J. Smola: “Learning with kernels”, MIT Press (2002).

[9] B. Schölkopf, A. Smola and K.-R. Müller: “Nonlinear component analysis as a kernel eigenvalue problem”, Neural Computation, **10**, 5, pp. 1299–1319 (1998).

[10] S. Mika, B. Schölkopf, A. J. Smola, K.-R. Müller, M. Scholz and G. Rätsch: “Kernel PCA and de-noising in feature spaces”, Advances in Neural Information Processing Systems 11 (Eds. by M. S. Kearns, S. A. Solla and D. A. Cohn), MIT Press (1999).

[11] S. Mika, G. Rätsch, J. Weston, B. Schölkopf and K.-R. Müller: “Fisher discriminant analysis with kernels”, Neural Networks for Signal Processing IX (Eds. by Y.-H. Hu, J. Larsen, E. Wilson and S. Douglas), IEEE, pp. 41–48 (1999).

[12] S. Mika, A. J. Smola and B. Schölkopf: “An improved training algorithm for kernel fisher discriminants”, Proceedings AISTATS 2001 (Eds. by T. Jaakkola and T. Richardson), Morgan Kaufmann, pp. 98–104 (2001).

[13] F. R. Bach and M. I. Jordan: “Kernel independent component analysis”, Journal of Machine Learning Research, **3**, pp. 1–48 (2002).

[14] 鷲沢, 山下: “カーネルウィナーフィルタ”, 第6回情報論の学習理論ワークショップ IBIS2003 予稿集, pp. 143–148 (2003).

[15] 鷲沢, 山下: “カーネルウィナーフィルタ”, 電子情報通信学会技術研究報告, NC2003, pp. 73–78 (2004).

[16] Y. Washizawa and Y. Yamashita: “Non-linear Wiener filter in reproducing kernel Hilbert space”, Proc. of 18th International Conference on Pattern Recognition (ICPR 2006) (2006 to appear).

[17] E. Oja: “Subspace methods of pattern recognition”, Wiley, New-York (1983).

[18] S. Watanabe and N. Pakvasa: “Subspace method in pattern recognition”, Proc. 1st Int. J. Conf on Pattern Recognition, Washington DC, pp. 25–32 (1973).

[19] H. Ogawa: “Karhunen-Loève subspace”, Proc. 11th IAPR Int. Conf. Patt. Recogn., Vol. 2, The Hague, The Netherlands, pp. 75–78 (1992).

[20] K. Karhunen: “Ueber lineare methoden in der wahrscheinlichkeit-rechnung”, Annalys Academiae Scientiarum Fennicae, Series A1: Mathematica-Physica, **37**, pp. 3–79 (1947).

[21] M. Loève: “Probability Theory”, Van Nostrand, New-York (1963).

[22] K. Pearson: “On lines and planes of closest fit to systems of points in space”, Philosophical Magazine, **2**, 6, pp. 559–572 (1901).

[23] H. Hotelling: “Analysis of a complex of statistical variables into principal components”, Journal of the American Statistical Association, **24**, p. 441 (1933).

[24] Y. Yamashita and H. Ogawa: “Relative Karhunen-Loève transform”, IEEE Transactions on signal processing, **44**, 2, pp. 1031–1033 (1996).

[25] Y. Yamashita: “Optimum sampling vectors for Wiener filter noise reduction”, IEEE Trans. of signal processing, **50**, 1, pp. 58–68 (2002).

[26] Y. Yamashita, Y. Ikeno and H. Ogawa: “Relative karhunen-loeve transform method for pattern recognition”, Proc. of International Conference on Pattern Recognition (ICPR 1998), pp. 1007–1010 (1998).

[27] 池野, 山下, 小川: “相対 KL 変換法によるパターン認識”, 信学論

- (D-II), **J80-D-II**, 2, pp. 541–547 (1997).
- [28] 鷲沢, 足田, 田中, 山下: “パターン認識のための相対 KL 変換法の高精度化”, 情報科学技術フォーラム 2002 一般講演論文集, I-48, pp. 95–96 (2002).
- [29] 前田, 村瀬: “カーネル非線形部分空間法によるパターン認識”, 信学論 (D-II), **J82-D-II**, 4, pp. 600–612 (1999).
- [30] 津田: “ヒルベルト空間における部分空間法”, 信学論 (D-II), **J82-D-II**, 4, pp. 592–599 (1999).
- [31] K. Tsuda: “Subspace classifier in the Hilbert space”, Pattern Recognition Letters, **20**, 5, pp. 513–519 (1999).
- [32] E. Maeda and H. Murase: “Multi-category classification by kernel based nonlinear subspace method”, IEEE International Conference On Acoustics, speech, and signal processing (ICASSP), Vol. 2, IEEE press., pp. 1025–1028 (1999).
- [33] 足田, 山下: “カーネル相対 KL 変換法によるパターン認識”, 信学技報, PRMU2001 (2002).
- [34] 鷲沢, 足田, 田中, 山下: “カーネル相対主成分分析による多クラスパターン認識”, 第二回 FIT (情報科学技術フォーラム) 情報レターズ, pp. 207–208 (2003).
- [35] Y. Washizawa, K. Hikida, T. Tanaka and Y. Yamashita: “Kernel relative principal component analysis for pattern recognition”, Proc. of Joint IAPR International Workshops on Syntactical and Structural Pattern Recognition and Statistical Pattern Recognition (SSPR/SPR 2004), pp. 1105–1113 (2004).
- [36] 鷲沢, 山下: “カーネル標準空間射影法によるパターン認識”, 第 6 回情報論的学習理論ワークショップ IBIS2003 予稿集, pp. 149–154 (2003).
- [37] Y. Washizawa and Y. Yamashita: “Kernel sample space projection classifier for pattern recognition”, 17th International Conference on Pattern Recognition (ICPR 2004), Vol. 2, pp. 435–438 (2004).
- [38] 鷲沢, 山下: “抑制型カーネル標準空間射影法によるパターン認識”, 電子情報通信学会技術研究報告, iPRMU2003, pp. 85–90 (2004).
- [39] Y. Washizawa and Y. Yamashita: “Kernel projection classifiers with suppressing features of other classes”, Neural Computation, **18**, 8, pp. 1932–1950 (2006).
- [40] L. L. Scharf: “The SVD and reduced rank signal processing”, Signal Processing, **25**, 2, pp. 113–133 (1991).
- [41] A. B. Israel and T. N. E. Greville: “Generalized Inverse: Theory and Applications”, John Wiley and Sons (1974).
- [42] G. Rätsch, T. Onoda and K.-R. Müller: “Soft margins for AdaBoost”, Machine Learning, **42**, 3, pp. 287–320 (2001). also NeuroCOLT Technical Report NC-TR-1998-021.
- [43] C. Bishop: “Neural Networks for Pattern Recognition”, Oxford Univ. Press (1995).
- [44] C. W. Groetsch: “Inverse problems in the mathematical sciences”, Vieweg (1993).
- [45] A. N. Tikhonov and V. Y. Arsenin: “Solution of Ill-posed problems”, V. H. Winston and Sons (1977).
- [46] 武者, 岡本: “逆問題とその解き方”, オーム社 (1992).
- [47] チャールズ W., 金子, 山本, 滝口: “数理科学における逆問題”, 別冊数理科学, サイエンス社 (1996).
- [48] Y. LeCun, L. D. Jackel, L. Bottou, A. Brunot, C. Cortes, J. S. Denker, H. Drucker, I. Guyon, U. A. Muller, E. Sackinger, P. Simard and V. Vapnik: “Comparison of learning algorithms for handwritten digit recognition”, International Conference on Artificial Neural Networks (Eds. by F. Fogelman and P. Gallinari), Paris, EC2 & Cie, pp. 53–60 (1995).
- [49] G. Rätsch: “Robust Boosting via Convex Optimization”, PhD thesis, University of Potsdam, Neues Palais 10, 14469 Potsdam, Germany (2001).
- [50] C. A. Rohde: “Generalized inverses of partitioned matrices”, Journal of Soc. Indust. Appl. Math., **13**, pp. 1033–1035 (1965).
- [51] B. Schölkopf, P. Bartlett, B. Smola and R. Williamson: “Shrinking the tube: A new support vector regression algorithm”, Advances in Neural Information Processing Systems, Vol. 11, Cambridge, MA. MIT Press. (1999).
- [52] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, A. Smola and K. Müller: “Invariant feature extraction and classification in kernel

spaces”, Advances in Neural Information Processing Systems 12, MIT Press, pp. 526–532 (2000).

Appendix

Lemma 1 (Operator equation [41]). Let $\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3$ and \mathcal{H}_4 be Hilbert spaces and $A \in \mathcal{B}(\mathcal{H}_3, \mathcal{H}_4)$, $B \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$, $C \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_4)$, where $\mathcal{B}(\mathcal{H}, \mathcal{H}')$ is bounded linear operator from \mathcal{H} to \mathcal{H}' . Assume that $\mathcal{R}(A)$, $\mathcal{R}(B)$ and $\mathcal{R}(C)$ are closed. Then operator equation

$$AXB = C \quad (32)$$

has a solution $X \in \mathcal{B}(\mathcal{H}_2, \mathcal{H}_3)$ when $\mathcal{R}(A) \supset \mathcal{R}(C)$ and $\mathcal{N}(B) \subset \mathcal{N}(C)$. A general form of a solution is given by

$$X = A^\dagger C B^\dagger + Y - A^\dagger A Y B B^\dagger, \quad (33)$$

where Y is an arbitrary operator in $\mathcal{B}(\mathcal{H}_2, \mathcal{H}_3)$.

Corollary 1. Let $A \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$, $B \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_3)$. If $\mathcal{R}(A)$ and $\mathcal{R}(B)$ are closed, an operator equation

$$A = XB \quad (34)$$

has a solution when

$$\mathcal{N}(B) \subset \mathcal{N}(A). \quad (35)$$

Proofs of Theorems 3, 4 and 5

Here, we omit the symbol i for a class i for brevity. If we let $\alpha = 0$ in eq.(23), it is reduced to eq. (19). Thus we consider eqs. (22), (23).

Since $\|u - P_{\mathcal{R}(U)}v\| \leq \|u - v\|$ for $\forall u \in \mathcal{R}(U)$, $\forall v \in \mathcal{H}$, X can be expressed as $X = UB$. From Corollary 1, a solution X in eqs. (19) and eqs. (23) can be expressed as $X = CU^*$. Then we can let

$$X = UAU^*, \quad (36)$$

where A is a real matrix of which size is $(|\Omega| + |\Psi|)$. Eq.(23) yields that

$$\begin{aligned} J &= \frac{1}{|\Omega|} \sum_{s=1}^{|\Omega|} \|\Phi(f_s) - UAU^*\Phi(f_s)\|^2 \\ &\quad + \frac{1}{|\Psi|} \sum_{t=1}^{|\Psi|} \|UAU^*\Phi(g_t)\|^2 + \mu \|UAU^*\|_2^2 \\ &= \frac{1}{|\Omega|} \sum_{s=1}^{|\Omega|} \{k(f_s, f_s) - 2\langle \Phi(f_s), UAU^*\Phi(f_s) \rangle \\ &\quad + \langle UAU^*\Phi(f_s), UAU^*\Phi(f_s) \rangle\} \\ &\quad + \frac{1}{|\Psi|} \sum_{t=1}^{|\Psi|} \langle UAU^*\Phi(g_t), UAU^*\Phi(g_t) \rangle + \mu \text{tr}[UA^*U^*UAU^*] \\ &= \frac{1}{|\Omega|} \sum_{s=1}^{|\Omega|} \{k(f_s, f_s) - 2\langle U^*\Phi(f_s), AU^*\Phi(f_s) \rangle \\ &\quad + \langle AU^*\Phi(f_s), K_U AU^*\Phi(f_s) \rangle\} \\ &\quad + \frac{1}{|\Psi|} \sum_{t=1}^{|\Psi|} \langle AU^*\Phi(g_t), K_U AU^*\Phi(g_t) \rangle + \mu \text{tr}[A^*K_U A K_U]. \end{aligned}$$

Note that $U^*\Phi(f_s)$ and $U^*\Phi(g_t)$ are the s -th and $(|\Omega| + t)$ -th column of K_U , respectively. Let $\tilde{D} = \Lambda^{-2}$. Then J is expressed as

$$\begin{aligned} J &= \frac{1}{|\Omega|} \text{tr}[K_U \Lambda_0 - 2(K_U \Lambda_0)^* A K_U \Lambda_0 + (K_U \Lambda_0)^* A^* K_U A K_U \Lambda_0] \\ &\quad + \frac{1}{|\Psi|} \text{tr}[(K_U \Lambda_0)^* A^* K_U A K_U \Lambda_0] + \mu \text{tr}[A^* K_U A K_U] \\ &= \text{tr}\left[\frac{1}{|\Omega|} K_U \Lambda_0 - \frac{2}{|\Omega|} \Lambda_0 K_U A K_U \Lambda_0\right. \\ &\quad \left.+ K_U A^* K_U A K_U \tilde{D}^{-1} + \mu A^* K_U A K_U\right], \end{aligned} \quad (37)$$

since $\Lambda_0 = \Lambda_0^*$, $\tilde{D}^* = \tilde{D}$ and $K_U^* = K_U$.

(i) Solution of KRPCA

When $\mu = 0$, J is reduced to

$$\begin{aligned} J &= \text{tr}[(\Lambda K_U A^* - \Lambda \Lambda_0) K_U (A K_U \Lambda - \Lambda_0 \Lambda)] + \psi \\ &= \|K_U^{1/2} (A K_U \Lambda - \Lambda_0 \Lambda)\|_F^2 + \psi, \end{aligned}$$

where ψ is a term which is independent from A , $K_U^{1/2}$ is a square root matrix of K_U . Let singular value decomposition of $K_U^{1/2} \Lambda_0 \Lambda$ is given as

$$K_U^{1/2} \Lambda_0 \Lambda = \sum_{i=1}^{\gamma} \sqrt{\lambda_i} (u_i \otimes \bar{v}_i). \quad (38)$$

Then if a rank of $K_U^{1/2} \Lambda^{-1} K_U A^*$ is less than η , J is minimum when

$$K_U^{1/2} A K_U \Lambda = \sum_{i=1}^{\eta} \sqrt{\lambda_i} (u_i \otimes \bar{v}_i). \quad (39)$$

Since Lemma 1, and a property of singular value decomposition, $u_i = \frac{1}{\sqrt{\lambda_i}} (K_U^{1/2} \Lambda_0 \Lambda) v_i$, one of the solutions is given as

$$\begin{aligned} A &= (K_U^{1/2})^\dagger K_U^{1/2} \Lambda_0 \Lambda \sum_{i=1}^{\eta} (v_i \otimes \bar{v}_i) \Lambda^{-1} K_U^\dagger \\ X &= U A U^* \\ &= U (K_U^{1/2})^\dagger K_U^{1/2} \Lambda_0 \Lambda \sum_{i=1}^{\eta} (v_i \otimes \bar{v}_i) \Lambda^{-1} K_U^\dagger U^* \end{aligned} \quad (40)$$

(ii) Solution of SKSP

The variation of J with respect to A in eq.(37) is given as

$$\begin{aligned} \delta J &= \text{tr}[K_U (\delta A)^* K_U A K_U \tilde{D}^{-1} + K_U A^* K_U (\delta A) K_U \tilde{D}^{-1} \\ &\quad - \frac{2}{|\Omega|} \Lambda_0 K_U (\delta A) K_U \Lambda_0 + \mu (\delta A)^* K_U A K_U + \mu (\delta A) K_U A^* K_U] \\ &= \text{tr}[(\delta A)^* (K_U A K_U \tilde{D}^{-1} K_U - \frac{1}{|\Omega|} K_U \Lambda_0 K_U + \mu K_U A K_U) \\ &\quad + (\delta A) (K_U \tilde{D}^{-1} K_U A K_U - \frac{1}{|\Omega|} K_U \Lambda_0 K_U + \mu K_U A^* K_U)] \\ &= 2 \text{tr}[(\delta A)^* (K_U A K_U \tilde{D}^{-1} K_U - \frac{1}{|\Omega|} K_U \Lambda_0 K_U + \mu K_U A K_U) \\ &\quad + (\delta A) (K_U \tilde{D}^{-1} K_U A K_U - \frac{1}{|\Omega|} K_U \Lambda_0 K_U + \mu K_U A^* K_U)] \end{aligned}$$

J is minimum when

$$K_U A (K_U + \mu \tilde{D}) \tilde{D}^{-1} K_U = \frac{1}{|\Omega|} K_U \Lambda_0 K_U. \quad (42)$$

In the case of $\mu > 0$, from Lemma 1,

$$\begin{aligned} A (K_U + \mu \tilde{D}) \tilde{D}^{-1} &= \frac{1}{|\Omega|} K_U^\dagger K_U \Lambda_0 K_U K_U^\dagger + W - K_U^\dagger K_U W K_U K_U^\dagger \\ &= W + K_U^\dagger K_U \left(\frac{1}{|\Omega|} \Lambda_0 - W\right) K_U K_U^\dagger, \end{aligned} \quad (43)$$

where W is arbitrary operator. Let $W' = W - \frac{1}{|\Omega|} \Lambda_0$, it follows that

$$A (K_U + \mu \tilde{D}) \tilde{D}^{-1} = \frac{1}{|\Omega|} \Lambda_0 + W' - K_U^\dagger K_U W' K_U K_U^\dagger.$$

Then we have

$$\begin{aligned} A &= \frac{1}{|\Omega|} \Lambda_0 \tilde{D} (K_U + \mu \tilde{D})^{-1} + W' \tilde{D} (K_U + \mu \tilde{D})^{-1} \\ &\quad - K_U^\dagger K_U W' K_U K_U^\dagger \tilde{D} (K_U + \mu \tilde{D})^{-1}. \end{aligned} \quad (44)$$

Since $\frac{1}{|\Omega|} \Lambda_0 \tilde{D} = \Lambda_0$, X which minimizes J is given as

$$\begin{aligned} X &= U A U^* \\ &= U \Lambda_0 (K_U + \mu \tilde{D})^{-1} U^* + U W' \tilde{D} (K_U + \mu \tilde{D})^{-1} U^* \\ &\quad - U K_U^\dagger K_U W' K_U K_U^\dagger \tilde{D} (K_U + \mu \tilde{D})^{-1} U^*. \end{aligned}$$

Since $K_U = U^* U$, $\mathcal{R}(K_U) = \mathcal{R}(U^*)$. Then we have

$$\begin{aligned} K_U K_U^\dagger U^* &= U^*, \\ U K_U^\dagger K_U &= U. \end{aligned}$$

It is clear that

$$U^* (U \tilde{D}^{-1} U^* + \mu I) = (U^* U + \mu \tilde{D}) \tilde{D}^{-1} U^*,$$

so that

$$(K_U + \mu \tilde{D})^{-1} U^* = \tilde{D}^{-1} U^* (U \tilde{D}^{-1} U^* + \mu I)^{-1}. \quad (45)$$

Then we have

$$\begin{aligned} K_U K_U^\dagger \tilde{D} (K_U + \mu \tilde{D})^{-1} U^* &= K_U K_U^\dagger U^* (U \tilde{D}^{-1} U^* + \mu I)^{-1} \\ &= U^* (U \tilde{D}^{-1} U^* + \mu I)^{-1} \\ &= \tilde{D} (K_U + \mu \tilde{D})^{-1} U^*. \end{aligned}$$

Hence, eq.(45) yields that

$$\begin{aligned} X &= U \Lambda_0 (K_U + \mu \tilde{D})^{-1} U^* \\ &\quad + U W' \tilde{D} (K_U + \mu \tilde{D})^{-1} U^* - U W' \tilde{D} (K_U + \mu \tilde{D})^{-1} U^* \\ &= U \Lambda_0 (K_U + \mu \tilde{D})^{-1} U^*. \end{aligned} \quad (46)$$

In the case of $\mu = 0$, if K_U is nonsingular, eq.(42) yields

$$A = \Lambda_0 K_U^{-1}, \quad (47)$$

$$X = U \Lambda_0 K_U^{-1} U^*. \quad (48)$$

Proof of Proposition 2

An operator A is a projector if and only if $AA = A$. Since eq. (40),

$$\begin{aligned} XX &= U A U^* U A U^* \\ &= U A K_U A U^*. \end{aligned}$$

If K_U is nonsingular,

$$\begin{aligned} XX &= U \Lambda_0 \Lambda \sum_{i=1}^{\eta} (v_i \otimes \bar{v}_i) \Lambda^{-1} K_U^{-1} K_U U^* \\ &= U A U^* = X. \end{aligned}$$

Proof of Proposition 3

If K_U is nonsingular and $\mu = 0$,

$$\begin{aligned}\tilde{P}_{\mathcal{R}(S)}^\mu \tilde{P}_{\mathcal{R}(S)}^\mu &= U \Lambda_0 K_U^{-1} U^* U \Lambda_0 K_U^{-1} U^* \\ &= U \Lambda_0 K_U^{-1} U^* = \tilde{P}_{\mathcal{R}(S)}^\mu.\end{aligned}$$

Proof of Proposition 4

If K_U is nonsingular and $\mu = 0$,

$$\begin{aligned}\tilde{P}_{\mathcal{R}(S)}^\mu P_{\mathcal{R}(U)} &= U \Lambda_0 K_U^{-1} U^* U K_U^{-1} U^* = U \Lambda_0 K_U^{-1} U^* = \tilde{P}_{\mathcal{R}(S)}^\mu, \\ P_{\mathcal{R}(U)} \tilde{P}_{\mathcal{R}(S)}^\mu &= U K_U^{-1} U^* U \Lambda_0 K_U^{-1} U^* = U \Lambda_0 K_U^{-1} U^* = \tilde{P}_{\mathcal{R}(S)}^\mu.\end{aligned}$$

Proof of Proposition 5

If K_U is nonsingular and $\mu = 0$, since $\mathcal{R}(T) = \mathcal{R}(UT)$

$$\tilde{P}_{\mathcal{R}(S)}^\mu UT = U \Lambda_0 K_U^{-1} U^* UT = U \Lambda_0 T = 0.$$