8-2010

A Novel Hybrid Dimensionality Reduction Method using Support Vector Machines and Independent Component Analysis

Sangwoo Moon
University of Tennessee - Knoxville, smoon3@utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Sangwoo Moon entitled "A Novel Hybrid Dimensionality Reduction Method using Support Vector Machines and Independent Component Analysis." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Computer Engineering.

Hairong Qi, Major Professor

We have read this dissertation and recommend its acceptance:

Itamar Arel, Seddik M. Djouadi, Xiaobing H. Feng

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a dissertation written by Sangwoo Moon entitled "A Novel Hybrid Dimensionality Reduction Method using Support Vector Machines and Independent Component Analysis." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Computer Engineering.

Hairong Qi, Major Professor

We have read this dissertation
and recommend its acceptance:

Itamar Arel

Seddik M. Djouadi

Xiaobing H. Feng

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of Graduate School

(Original signatures are on file with official student records.)

A Novel Hybrid Dimensionality Reduction Method using Support Vector Machines and Independent Component Analysis

A Dissertation
Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Sangwoo Moon

August 2010

Dedication

I dedicate this dissertation to my family. Particulary to my parents who have supported me with full of love and believed in the pursuit of academic excellence.

Acknowledgments

First of all, I would like to thank my advisor, Dr. Hairong Qi from the bottom of my heart. I cannot imagine how to complete the dissertation without her dedicated support, guidance, and great insight into the various research areas. Her sincere and broad interests in various engineering fields have been a great inspiration to me. She was not just a mentor for the research. From her guidance, I could also learn and feel the way of open-minded thinking and the effective communication skill.

I also would like to thank all the other committee members, Dr. Itamar Arel, Dr. Seddik M. Djouadi, and Dr. Xiaobing H. Feng for the time and efforts to improve this dissertation. I also appreciate all the help from AICIP group members as well as alumni.

Abstract

Due to the increasing demand for high dimensional data analysis from various applications such as electrocardiogram signal analysis and gene expression analysis for cancer detection, dimensionality reduction becomes a viable process to extracts essential information from data such that the high-dimensional data can be represented in a more condensed form with much lower dimensionality to both improve classification accuracy and reduce computational complexity. Conventional dimensionality reduction methods can be categorized into *stand-alone* and *hybrid* approaches. The stand-alone method utilizes a single criterion from either supervised or unsupervised perspective. On the other hand, the hybrid method integrates both criteria. Compared with a variety of stand-alone dimensionality reduction methods, the hybrid approach is promising as it takes advantage of both the supervised criterion for better classification accuracy and the unsupervised criterion for better data representation, simultaneously. However, several issues always exist that challenge the efficiency of the hybrid approach, including (1) the difficulty in finding a subspace that seamlessly integrates both criteria in a single hybrid framework, (2) the robustness of the performance regarding noisy data, and (3) nonlinear data representation capability.

This dissertation presents a new hybrid dimensionality reduction method to seek projection through optimization of both structural risk (supervised criterion) from Support Vector Machine (SVM) and data independence (unsupervised criterion) from Independent Component Analysis (ICA). The projection from SVM directly contributes to classification performance improvement in a *supervised* perspective whereas maximum independence among features by ICA construct projection indirectly achieving classification accuracy improvement due to better intrinsic data representation in an *unsupervised* perspective. For linear dimensionality reduction model, I introduce orthogonality to interrelate both projections

from SVM and ICA while redundancy removal process eliminates a part of the projection vectors from SVM, leading to more effective dimensionality reduction. The orthogonality-based linear hybrid dimensionality reduction method is extended to uncorrelatedness-based algorithm with nonlinear data representation capability. In the proposed approach, SVM and ICA are integrated into a single framework by the uncorrelated subspace based on kernel implementation.

Experimental results show that the proposed approaches give higher classification performance with better robustness in relatively lower dimensions than conventional methods for high-dimensional datasets.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Dimensionality reduction transforms the observation onto reduced dimensional space so as to resolve the "curse of dimensionality" problem in which the increase of the observation dimension leads to the exponential increase in volume. The data mapped from the observation to lower dimensional space by dimensionality reduction procedure must present the entire observations effectively. The effectiveness of the observations in reduced dimensional space is measured by the corresponding criteria defined in various dimensionality reduction algorithms. For example, Principal Component Analysis (PCA) [Ekenel and Sankur 2005; Martinez and Kak 2001; Nishino et al. 2005; Vidal et al. 2005], Linear Discriminant Analysis (LDA) [Martinez and Kak 2001; Ye et al. 2004], and Independent Component Analysis (ICA) [Comon 1994; Hyvarinen and Oja 2000] are all popular dimensionality reduction algorithms that have been successfully applied to diverse range of real-world applications [Chang et al. 2005; Lu et al. 2006; Park et al. 2002]. PCA uses eigenvalue decomposition to find orthogonal projection vectors, also referred to as principal components, that minimize squared error between the original and projected observations. LDA forms a criterion function by between-class and within-class covariance matrices such that the between-class scatter matrix is maximized and the within-class scatter matrix is minimized so as to obtain better separability in reduced space. ICA pursues statistically independent projection vectors from observation by criterion representing independence

such as Kullback-Liebler (KL) divergence, mutual information, and correlation.

Dimensionality reduction methods can be categorized from three perspectives: 1) supervised or unsupervised, 2) stand alone or hybrid, and 3) capability of supporting nonlinearity in data. From the aspect if the formulation needs the class index to construct the optimization criterion, dimensionality reduction algorithms can be categorized as supervised and unsupervised. LDA is a representative supervised method due to the within- and between-class covariance built by the observations grouped by classes. On the contrary, the unsupervised approach does not utilize class index to build the criteria. PCA and ICA are representative unsupervised dimensionality reduction algorithms since their criteria are irrelevant to class index.

From the aspect if only supervised or unsupervised criterion is used or both criteria are used, dimensionality reduction methods can be categorized as standalone or hybrid. For example, PCA, LDA, and ICA are all standalone algorithms. PCA+LDA [Yang and Yang 2001, 2003] that combines the supervised LDA with the unsupervised PCA removing null space prior to LDA, is a representative hybrid algorithm.

Nonlinear capability acts as an essential component in dimensionality reduction methods for accurate data representation in lower dimensional space since the real world applications always include nonlinearity resulting in performance degradation based on linear model. PCA, LDA, and ICA are based on linear dimensionality reduction model consisting only of linear data projection. The nonlinearity can be introduced by nonlinear mapping from input to hyperdimensional feature space. The linear data analysis over the observation projected onto the feature space reveals corresponding nonlinear nature of observation in the input space. However, direct use of non-linear mapping for entire data is computationally expensive. Therefore, kernel trick [Herbrich 2001] is applied where kernel makes the inner product equivalent to single point mapping from input to feature space.

This dissertation focuses on the study of hybrid dimensionality reduction algorithms that take advantage of both the supervised criterion resulting in mapping vectors aimed for better classification accuracy and the unsupervised criterion yielding mapping vectors that better represent the original data, simultaneously.

1.2 Contribution

Although with great potential, hybrid dimensionality reduction algorithms also bring unique challenges that can be summarized from four aspects. First, it is essential to choose appropriate supervised and unsupervised dimensionality reduction methods. Conventional hybrid methods mostly depend on supervised LDA so that the problems inherited from LDA reside in the hybrid design regardless of the way of supervised and unsupervised criteria integration. Secondly, for arbitrary complicated objective functions with constraints, subspace-based methods are easier to couple the objectives into single framework compared with the method using unified criterion through constraint optimization, in which case the construction of an appropriate subspace becomes a challenging problem. Third, due to the nonlinear nature of real-world data, it is important to incorporate nonlinearity in the algorithm design. The difficulty resides in the fact that the nonlinear extension should be accomplished in both criteria without affecting the seamless integration of the two criteria. Fourth, we need to consider the robustness of performance (or the generalization capability of the algorithm) regarding noisy data or partial information.

The dissertation work presents a set of innovative hybrid dimensionality reduction algorithms that effectively answers to the challenging issues discussed above.

First, for hybrid dimensionality reduction, the proposed method adopts Support Vector Machine (SVM) and Independent Component Analysis (ICA) for the supervised and unsupervised algorithms, respectively. SVM provides better classification performance for arbitrary observation by its generalization capability from the structural risk minimization over tradeoff between empirical error and complexity of the decision surface. SVM can be used for dimensionality reduction purpose in the similar way as LDA where decision boundaries for classification can also be treated as projection vectors for dimensionality reduction. By using SVM instead of LDA, the hybrid dimensionality reduction algorithm is not influenced by the weaknesses inherited from LDA criterion. For unsupervised criteria integration, independence maximization is incorporated into the hybrid dimensionality reduction framework since the concept of independence is known as an effective measure to find intrinsic data representation, compared with other unsupervised method such as PCA. The hybrid framework utilizes ICA to perform maximization of independence ap-

proximated by mutual information [Hyvarinen 1999].

Second, robustness is achieved by incorporating SVM into the proposed method. The generalization capability in SVM results in the decision surface satisfying maximum separation margin from the decision surface to the closest training observations. Consequently, the arbitrary input in the outer closest training observations becomes better generalized in classification. The generalization in classification works identically in dimensionality reduction since better separation delivers more information of the data.

Third, in order to seamlessly integrate SVM and ICA, subspace-based approaches are designed which helps yield minimum relevance between SVM and ICA. To achieve the minimum relevance, orthogonal and uncorrelated subspace are introduced to couple the objectives of SVM and ICA into single framework. The orthogonal subspace provides the orthogonal property between the bases from SVM and ICA based on the definition of the projection by minimum distance objective function. However, due to the better applicability of the subspace based on correlation for nonlinear data representation, the maximally uncorrelated subspace is designed, referred to as the "uncorrelated subspace" to emphasize the relationship with the projection from SVM. The subspace construction is formulated by introducing Lagrangian multipliers and finally summarized in the form of eigenvalue decomposition. Over the uncorrelated subspace in nonlinear feature space, ICA is performed to reveal the nature of observations.

Fourth, the nonlinear extension of the linear hybrid dimensionality reduction based on SVM and ICA is developed based on uncorrelated subspace construction with kernel function. As a result, the data dimensionality is reduced by the proposed method based on the nonlinear projection consisting of SVM and ICA projections with uncorrelated subspace.

1.3 Dissertation Outline

The dissertation is organized as follows: Chapter 2 provides literature reviews for support vector machines, conventional dimensionality reduction methods, and constrained optimization techniques. Chapter 3.1 introduces SVM as robust dimensionality reduction criterion with redundancy removal process. Based on SVM, the orthogonal subspace-based

linear SVM plus ICA is described in Chapter 3.2. The uncorrelated subspace-based nonlinear SVM plus ICA is developed in Chapter 3.3. Experimental results are shown in Chapter 4. This dissertation is concluded in Chapter 5.

3

Chapter 2

Background

2.1 Support Vector Machines

This section introduces support vector machines [Vapnik 1999] with the related concepts such as structural risk [Vapnik 1999], Mercer's theorem [Herbrich 2001], and kernel machine [Muller et al. 2001].

2.1.1 Structural Risk vs. Empirical Risk

The empirical risk, R_{emp}, is well-known measure of learning machine, $f(\boldsymbol{x}, \boldsymbol{\alpha})$ where $\boldsymbol{x}$ dataset and $\boldsymbol{\alpha}$ denotes a set of parameters of the corresponding learning machine, f. The training dataset consists of N-many pairs of $\boldsymbol{x}_i$ and y_i, $\forall i = \{1, \cdots, N\}$ where $y_i = \{1, -1\}$. The one of the representative learning machine is neural networks. The structure with activation function of fixed neural network corresponds to f and the connection weights are to $\boldsymbol{\alpha}$. R_{emp} is defined by measured mean error on training dataset as follows,

$$R_{\text{emp}}(\boldsymbol{\alpha}) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{2} |y_i - f(\boldsymbol{x}_i, \boldsymbol{\alpha})| \tag{2.1}$$

where $|y_i - f(\boldsymbol{x}_i, \boldsymbol{\alpha})|/2$ is called the loss which only can take 0 and 1. In spite of the wide utilization of empirical error, R_{emp} has in general a certain distance away from the actual

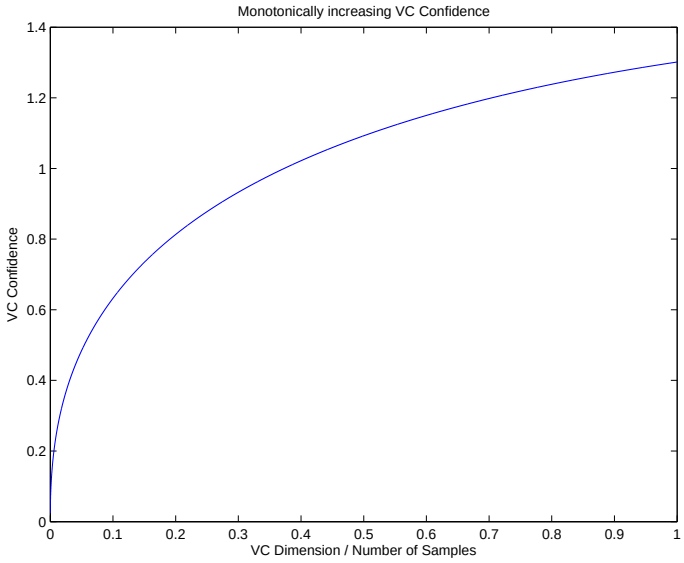


Figure 2.1: Monotonically increasing VC confidence

risk, R defined by the cumulative density function of $P(\boldsymbol{x}, y)$ as follows,

$$R(\boldsymbol{\alpha}) = \int \frac{1}{2} \left| y - f(\boldsymbol{x}, \boldsymbol{\alpha}) \right| dP(\boldsymbol{x}, y) \tag{2.2}$$

By choosing $\eta = [0, 1]$, for losses with probability of $1 - \eta$, Vapnik showed the bound holds as follows,

$$R(\boldsymbol{\alpha}) \leq R_{\text{emp}}(\boldsymbol{\alpha}) + \sqrt{\left(\frac{h(\log(2N/h) + 1) - \log(\eta/4)}{N} \right)} \tag{2.3}$$

where $h \geq 0$ is Vapnik Chervonenkis (VC) dimension indicating the complexity of the hypothesis space. $\sqrt{(h(\log(2N/h) + 1) - \log \eta/4)/N}$ in Eq. (2.3) is called VC confidence. Figure 2.1 shows the monotonically increasing VC confidence when VC dimension increases. The tradeoff relationship between training error and complexity is clearly shown in Eq. (2.3) with the monotonically increasing VC confidence over h in Fig. 2.1. The smaller h for simpler hypothesis space is highly probable not to include appropriate approximation capability, resulting in higher R_{emp}. On the contrary, larger h might decrease R_{emp} with higher VC confidence.

To obtain minimum actual risk, nested structure with certain VC dimension of the

hypothesis space is introduced as,

$$H_1 \subset H_2 \subset \cdots \subset H_i \cdots \tag{2.4}$$

where H_i is the nested structure of hypothesis space with the i-th VC dimension, $h_i \leq h_{i+1}$, $\forall i$. The goal is to find H_i with the tightest bound over the nested structure in Eq. (2.4). To find the nested structure, SVM directly obtain the upper bound of VC dimension by the definition of separation margin which is independent to the dimensionality of input, $\boldsymbol{x}$.

2.1.2 Nonlinear Implementation via Kernel

$f : \chi \times \chi \rightarrow \mathcal{R}$ is a kernel providing inner product of $\boldsymbol{x} \in \chi$ in feature space $\mathcal{F}$ without direct analysis of $\phi(\boldsymbol{x})$. The utilization of kernel function easily delivers nonlinear capability in the algorithm only if the objective function is built only by a set of inner product such as covariance matrix. In case of SVM, kernel implementation is especially useful since the nested structure with the lowest bound described in Sec.2.1.1 is found not based on the feature-by-feature analysis with predefined rank but constructing the feature dimension, each dimension of which corresponds to the individual input data. The kernel function is characterized in Reproducing Kernel Hilbert Space (RKHS) and Mercer's Theorem as a generalization of spectral decomposition.

Suppose K is symmetric positive-definite matrix where $K = [k_{ij}]$, $k_{ij} = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$. K is called "Gram matrix" for the kernel evaluations on the data. For the finite N-many input $\boldsymbol{x}_i$,$i \in \{1, \cdots, N\}$, assume K has full rank. The eigen-decomposition of K becomes as follows,

$$K = U \Lambda U^{\mathrm{T}} \tag{2.5}$$

where U is unitary matrix consisting of normalized eigenvectors in columns. Λ is a diagonal matrix with eigenvalues of $\lambda_1 \geq \cdots \geq \lambda_N > 0$ along the diagonal. From Eq. (2.5), The

kernel k_{ij} is obtained with the corresponding feature mapping ϕ as follows,

$$\begin{aligned} k_{ij} &= \left(\Lambda^{\frac{1}{2}} U_i\right)^{\mathrm{T}} \left(\Lambda^{\frac{1}{2}} U_j\right) \\ &= \langle \sqrt{\lambda_i}\phi(\boldsymbol{x}_i), \sqrt{\lambda_j}\phi(\boldsymbol{x}_j) \rangle \\ &= f(\boldsymbol{x}_i, \boldsymbol{x}_j) \end{aligned} \tag{2.6}$$

where U_i denotes the i-th row vector of U. It is clear that the eigenvalues should be non-negative and less than positive infinity to obtain inner product space due to $\|\phi(\boldsymbol{x}_i)\|^2 = U_i^{\mathrm{T}} \Lambda U_i = \lambda_i$. The generalization of this concept is called Mercer's Theorem as follows,

Theorem 1. *(Mercer's). Suppose that f is a continuous positive semi-definite kernel on a compact set, χ as $f : \chi \times \chi \to \mathcal{R}$ is symmetric and $\sup_{\boldsymbol{x},\boldsymbol{y}} f(\boldsymbol{x}, \boldsymbol{y}) < \infty$. Let define the integral operator $T_f : L^2(\chi) \to L^2(\chi)$ as,*

$$(T_f p)(\cdot) = \int_{\chi} f(\cdot, \boldsymbol{x}) p(\boldsymbol{x}) d\boldsymbol{x} \tag{2.7}$$

is positive semi-definite, $\forall p \in L^2(\chi)$,

$$\int_{\chi} f(\boldsymbol{u}, \boldsymbol{v}) p(\boldsymbol{u}) p(\boldsymbol{v}) d\boldsymbol{u} d\boldsymbol{v} \geq 0 \tag{2.8}$$

Then there is an orthonormal basis ϕ_i of $L_2(\chi)$ as eigenfunction corresponding to the nonzero eigenvalue of λ_i with $\int_{\chi} f(\cdot, \boldsymbol{x}) \phi_i(\boldsymbol{x}) d\boldsymbol{x} = \lambda_i \phi_i(\cdot)$, then $f(\boldsymbol{u}, \boldsymbol{v})$ has the representation of,

$$f(\boldsymbol{u}, \boldsymbol{v}) = \sum_{i=1}^{\infty} \lambda_i \phi_i(\boldsymbol{u}) \phi_i(\boldsymbol{v}) \tag{2.9}$$

where $\sum_i \lambda_i < \infty$ and $\sup_{\boldsymbol{x}} \phi_i(\boldsymbol{x}) < \infty$. The convergence is absolute and uniform in $\boldsymbol{u}$,$\boldsymbol{v}$.

RKHS is fundamentally defined in a Hilbert space $\mathcal{H}$ with reproducing kernel. Consider the vector space with $\phi : \chi \to \mathcal{R}^{\chi}$, $\phi(\boldsymbol{x}) = f(\cdot, \boldsymbol{x})$ as,

$$\mathrm{span}\left(\{\phi(\boldsymbol{x}) : \boldsymbol{x} \in \chi\}\right) = \left\{ g(\cdot) = \sum_i \alpha_i f(\cdot, \boldsymbol{x}_i) | \boldsymbol{x}_i \in \chi, \alpha_i \in \mathcal{R} \right\} \tag{2.10}$$

For $p = \sum_i \alpha_i f(\cdot, \boldsymbol{u}_i)$ and $q = \sum_i \beta_i f(\cdot, \boldsymbol{v}_i)$, the inner product of p and q is defined as,

$$\begin{aligned}\langle p, q\rangle &= \sum_{i,j} \alpha_i \beta_j f(\boldsymbol{u}_i, \boldsymbol{v}_j) \\ &= \sum_i \beta_i p(\boldsymbol{v}_i) \\ &= \sum_i \alpha_i q(\boldsymbol{u}_i)\end{aligned} \tag{2.11}$$

Eq. (2.11) is summarized to reproducing property as

$$\langle g, f(\cdot, \boldsymbol{x})\rangle = \sum_i \alpha_i f(\boldsymbol{x}, \boldsymbol{u}_i) = g(\boldsymbol{x}) \tag{2.12}$$

To show that $\langle p, q\rangle$ is an inner product, three properties must be check: 1) symmetry, 2) bilinearity, and 3) positive definiteness. The symmetry is confirmed as $\langle p, q\rangle = \sum_{i,j} \alpha_i \beta_j f(\boldsymbol{u}_i, \boldsymbol{v}_j) = \langle q, p\rangle$. The bilinearity is already shown in Eq. (2.11). Since $\langle p, p\rangle = \boldsymbol{\alpha}^{\mathrm{T}} K \boldsymbol{\alpha}$ is a quadratic form with $\boldsymbol{\alpha} = [\alpha_1 \cdots \alpha_N]^{\mathrm{T}}$ and K is positive definite, positive definite property holds with $\langle p, p\rangle = 0$ iff $p = 0$. From the inner product space defined, K with the reproducing property spans $\mathcal{H} = span\{f(\cdot, \boldsymbol{x}) | \boldsymbol{x} \in \chi\}$

In summary, f is the reproducing kernel of an RKHS of functions on χ and the kernel represents a legitimate inner product in feature space if f satisfies Mercer's theorem. By using the kernel, f, the algorithm simply incorporates the nonlinear data analysis capability instead of relying on multiple linear manifold analysis.

2.1.3 Fundamental of Support Vector Machines

SVM [Vapnik 1999] searches for a decision boundary which minimizes the upper bound of the actual risk in Eq. (2.3) over the tradeoff between empirical risk and complexity based on Vapnik-Cervonenkis (VC) theory in Sec. 2.1.1. Instead of feature-based analysis requiring pre-defined rank of the features in input observations, SVM introduces separation margin independent to the input dimensionality but relying on the number of training data.

The dataset for two class problem is defined by $(\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_N, y_N)$, $\boldsymbol{x} \in R^n$, $y_i \in \{-1, 1\}$ where $\boldsymbol{x}_i$ and y_i, $i = 1, \cdots, N$ are data vector and class index. The decision is made by the linear hyperplane represented by $\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b = 0$ where $\boldsymbol{w}$ is a vector orthogonal

to the plane and b shifts the plane to be placed in the middle of two classes. The linear hyperplane should satisfy the equally distant condition as follows,

$$y_i[\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b] \geq 1 \tag{2.13}$$

where $i = 1, \cdots, N$. The geometric distance for $\boldsymbol{x}$ from the hyperplane is as follows,

$$dist(\boldsymbol{w}, b; \boldsymbol{x}) = |\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b| / \|\boldsymbol{w}\| \tag{2.14}$$

Suppose $\|\boldsymbol{w}\| < 1/\triangle$. From Eq. (2.13) and Eq. (2.14), $dist(\boldsymbol{w}, b; \boldsymbol{x}) \geq \triangle$ where the decision hyperplane separates data with the margin of $[-\triangle, \triangle]$. The VC dimension is then bounded by $h \leq \min([R^2/\triangle^2], n) + 1$ in [Vapnik 1999] where R is the radius of a hypersphere enclosing all the training data. Therefore, minimization of Eq. (2.14) with the equal distant separation margin constraint in Eq. (2.13) is equivalent to the minimization of the upper bound of the actual risk in Eq. (2.3) on the VC dimension.

To solve SVM's constraint optimization problem, the normalized margin is given by

$$\begin{aligned} \rho(\boldsymbol{w}, b) =& \min_{\{\boldsymbol{x}_i; y_i = 1\}} dist(\boldsymbol{w}, b; \boldsymbol{x}_i) + \min_{\{\boldsymbol{x}_j; y_j = -1\}} dist(\boldsymbol{w}, b; \boldsymbol{x}_j) \\ =& \frac{1}{\|\boldsymbol{w}\|} \left(\min_{\{\boldsymbol{x}_i; y_i = 1\}} |\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b| + \min_{\{\boldsymbol{x}_j; y_j = -1\}} |\langle \boldsymbol{w}, \boldsymbol{x}_j \rangle + b| \right) \\ =& \frac{2}{\|\boldsymbol{w}\|} \end{aligned} \tag{2.15}$$

Therefore, the hyperplane which separates a two-class dataset with maximum separation margin is obtained by maximization of $\rho(\boldsymbol{w}, b)$, which is equal to minimization of $\|\boldsymbol{w}\|^2/2$. Since the minimization problem must satisfy $y_i[\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b] \geq 1$, Lagrangian formulation is utilized to incorporate the inequality constraint into the minimization problem as follows,

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + \sum_{i=1}^{N} \alpha_i [1 - y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b)] \tag{2.16}$$

where $\alpha_i \geq 0$ is the Lagrange multiplier. $L(\boldsymbol{w}, b, \boldsymbol{\alpha})$ in Eq. (2.16) must be minimized with respect to $\boldsymbol{w}$ and b while maximized with respect to α. Based on convexity of $L(\boldsymbol{w}, b, \boldsymbol{\alpha})$, I can solve for $\boldsymbol{w}$ and b by taking the partial derivatives $\partial L/\partial \boldsymbol{w} = 0$ and $\partial L/\partial b = 0$,

which yield $\boldsymbol{w} = \sum_{i=1}^{N} \alpha_i y_i \boldsymbol{x}_i$ and $\sum_{i=1}^{N} \alpha_i y_i = 0$. The duality allows us to convert the minimization problem in Eq. (2.16) of $\boldsymbol{w}$ and b to the maximization problem of $\boldsymbol{\alpha}$ as follows,

$$\begin{aligned} \max_{\boldsymbol{\alpha}} W(\boldsymbol{\alpha}) &= \max_{\boldsymbol{\alpha}} \left\{ \frac{1}{2} \| \sum_{i=1}^{N} \alpha_i y_i \boldsymbol{x}_i \|^2 + \sum_{i=1}^{N} \alpha_i [1 - y_i (\sum_{j=1}^{N} \alpha_j y_j \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle + b)] \right\} \\ &= \min_{\boldsymbol{\alpha}} \left\{ \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle - \sum_{i=1}^{N} \alpha_i \right\} \end{aligned} \tag{2.17}$$

The $\boldsymbol{w}$ from Eq. (2.17) with the constraints of $\sum_{i=1}^{N} \alpha_i y_i = 0$ and $\boldsymbol{\alpha} \geq 0$ provides hard decision boundary which does not include any misclassified training data. To allow the misclassification in training phase which is more general, Vapnik [Vapnik 1999] introduces penalty function, $F(\boldsymbol{\xi}) = \sum_i \xi_i^{\sigma}$ where $\boldsymbol{\xi} \geq 0$ denotes misclassification error measure and $\sigma > 0$. The penalty function affect to SVM's minimum separation margin constraint in Eq. (2.13) as follows,

$$y_i [\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b] \geq 1 - \xi_i \tag{2.18}$$

$\xi_i \geq 0, \forall i$. Hence, the minimization of the function inversely proportional to the minimum separation margin in Eq. (2.14) becomes $\|\boldsymbol{w}\|^2 + C \sum_i \xi_i$ constrained by Eq. (2.18) where C is a given regularization parameter. The Lagrangian formulation in Eq. (2.16) is updated with the penalty term as follows,

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\xi}) = \frac{1}{2} \|\boldsymbol{w}\|^2 + \sum_{i=1}^{N} \alpha_i [1 - \xi_i - d_i (\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b)] + C \sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} \beta_i \xi_i \tag{2.19}$$

where both $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are Lagrange multipliers. $L(\boldsymbol{w}, b, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\xi})$ in Eq. (2.19) must be minimized with respect to $\boldsymbol{w}$,$\boldsymbol{b}$, and $\boldsymbol{\xi}$ while maximized with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. Since $\alpha_i + \beta_i = C$ from $\partial L(\boldsymbol{w}, b, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\xi}) / \partial \boldsymbol{\xi} = 0$ cancels $\sum_i \alpha_i (-\xi_i)$, $C \sum_i \xi_i$, and $-\sum_i \beta_i \xi_i$, the dual problem shows identical formulation as Eq. (2.17) with additional constraint of $\boldsymbol{\alpha} \leq C$ from $\alpha_i + \beta_i = C, \forall i$. The separation margin from SVM's decision hyperplane increases when C decreases.

It is clear that the nonlinear data becomes manageable by introducing ϕ which represents the nonlinear nature of data, although ϕ is usually unknown and the direct mapping

of data through ϕ requires heavy computational burden. "kernel technique" in Sec. 2.1.2 provides indirect way of obtaining $\langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$ without prior knowledge of ϕ and high computational complexity through kernel as follows,

$$f(\boldsymbol{x}, \boldsymbol{y}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{y}) \rangle \tag{2.20}$$

which is to map inner product of $\boldsymbol{x}$ and $\boldsymbol{y}$ into feature space, $\mathcal{F}$, though the kernel, f, without using mapping function, ϕ. The constrained minimization problem of SVM in Eq. (2.17) is now summarized for both linear and nonlinear data representation using kernel function with regularization parameter C as follows,

$$\begin{aligned} \boldsymbol{\alpha}^* = & \underset{\boldsymbol{\alpha}}{\text{argmin}} \left(\frac{1}{2} \boldsymbol{\alpha}^{\mathrm{T}} K \boldsymbol{\alpha} - \boldsymbol{\alpha}^{\mathrm{T}} \mathbf{1}_{N \times 1} \right) \\ & st.\ 0 \leq \boldsymbol{\alpha} \leq C \\ & \sum_{i=1}^{N} \alpha_i y_i = 0 \end{aligned} \tag{2.21}$$

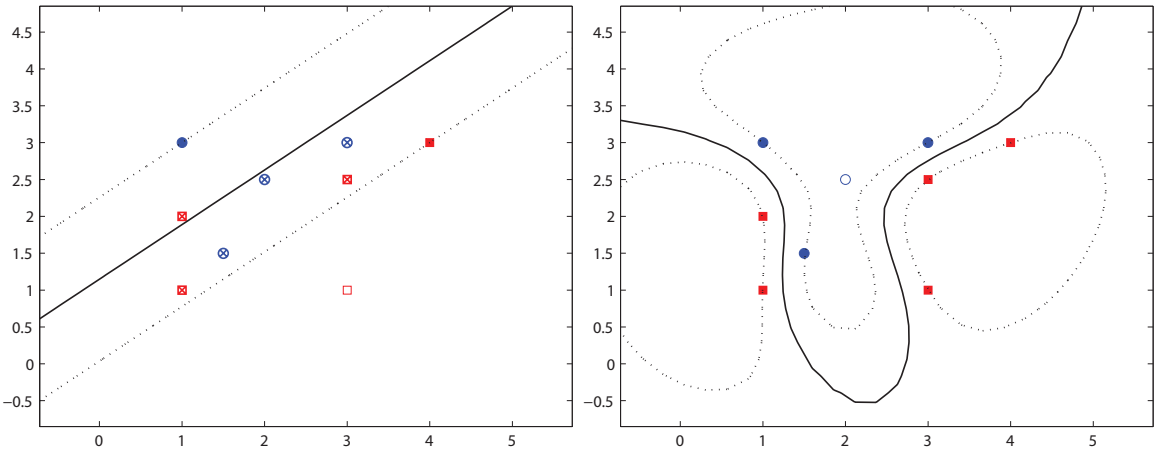
where $\boldsymbol{\alpha} = [\alpha_1 \cdots \alpha_N]^{\mathrm{T}}$, $K = [k_{ij}]$, $k_{ij} = f(\boldsymbol{x}_i, \boldsymbol{x}_j)$, and $1_{P \times Q}$ denotes $P \times Q$ matrix consisting only of 1. The quadratic formulation can be solved by quadratic programming and the non-zero α_i's among optimal α_i's $\forall i$ construct SVM's decision surface with corresponding training data $\boldsymbol{x}_i$'s called "Support Vector". The decision is made by

$$\begin{aligned} \langle \boldsymbol{w}, \phi(\boldsymbol{x}) \rangle + b = & \sum_{i=1}^{N} \alpha_i y_i f(\boldsymbol{x}_i, \boldsymbol{x}) + b \\ & \underset{\text{class2}}{\overset{\text{class1}}{\gtrless}} 0 \end{aligned} \tag{2.22}$$

where $\boldsymbol{x}$ is an arbitrary input. The bias, b is defined by support vectors as follows,

$$b = \frac{1}{n(S)} \sum_{i \in S} \left(y_i - \sum_{j \in S} \alpha_j y_j f(\boldsymbol{x}_i, \boldsymbol{x}_j) \right) \tag{2.23}$$

where S represents a set of support vectors. There exist several kernel functions such as gaussian (radial basis), exponential, fourier, splines, and additive kernels. However, it is widely accepted that gaussian kernel function works sufficient in most cases since each



(a) Linear decision boundary (b) Nonlinear decision boundary

Figure 2.2: Examples of Linear and nonlinear decision boundary by SVM based on kernel

of the support vectors contributes one local gaussian function centered at the support vector. The set of support vectors with local gaussian functions corresponds to Radial Basis Function Network (RBFN) which is proven that RBFN can fit any function with infinite many hidden neurons. The gaussian kernel function is defined as follows,

$$f(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma^2}\right) \tag{2.24}$$

where σ is a given gaussian kernel width.

Figure 2.2 shows linear and nonlinear decision boundaries by SVM with linear and gaussian kernel, respectively. The solid dots indicate data points as support vectors in which corresponding Lagrange multipliers become zero. As shown in Figure 2.2, nonlinear kernel generates nonlinear decision boundary which provides better discriminant capability for the given dataset.

2.1.4 Multiclass Extension

Due to the limitation of SVM designed only for two-class dataset, there are two distinctive multiclass SVM approaches, referred to as one-against-all and one-against-one [Hsu and Lin 2002]. The one-against-all approach compares data in a single class with all the others to generate the decision boundary. This method builds c-many decision boundaries from

c-many one-against-all data combinations, where c denotes the number of classes. The one-against-one approach creates decision boundaries from all possible combinations of two different classes. It basically generates ${}_cC_2$-many decision boundaries. For a 2-class pattern, one-against-one is equivalent to one-against-all.

The one-against-all provides relatively small number of projection vectors than one-against-one, resulting in lower dimensional data representation since $c \leq {}_cC_2$ for $c \geq 3$. However, the one-against-all also requires at least equal or more amount of data per SVM for training compared with the one-against-one, resulting in higher computational complexity to solve the quadratic problem in Eq. (2.21) since the number of unknown variables, α's increases proportionally to the number of training samples increasing.

For computational efficiency, one-against-one approach is extended with tree structure for fast decision making such as Directed Acyclic Graph SVM (DAGSVM) [Platt et al. 2000], Binary Tree of SVM (BTS) [Fei and Jinbai 2006] and SVM with Binary Tree Architecture (SVM-BTA) [Cheong et al. 2004]. These methods shorten the decision path in the tree resulting in less number of decision making compared with the original one-against-one requiring full ${}_cC_2$-many times of decision making. Additionally, the tree-based method does not require any decision fusion such as majority voting.

2.2 Dimensionality Reduction

Due to the increasing demand for high dimensional data analysis from various applications such as electrocardiogram (ECG) signal analysis, gene expression analysis for cancer detection/DNA forensic, and content-based image retrieval (CBIR), dimensionality reduction becomes a viable process to provide robust data representation in relatively low dimensional space. Dimensionality reduction is a process to extract essential information from data such that the high-dimensional data can be represented in a more condensed form with much lower dimensionality to both improve classification accuracy and reduce computational complexity. Conventional dimensionality reduction methods can be categorized into *stand-alone* and *hybrid* approaches. The stand-alone method utilizes a single criterion from either supervised or unsupervised perspective, where supervised approaches require the prior knowledge of class assignment for training data whereas the unsupervised meth-

ods are free from this requirement. On the other hand, the hybrid method integrates both criteria. Compared with a variety of stand-alone dimensionality reduction methods, the hybrid approach is promising as it takes advantage of both the supervised criterion that results in mapping vectors aimed for better classification accuracy and the unsupervised criterion yielding mapping vectors that better represent the original data, simultaneously. However, two issues always exist that challenge the efficiency of the hybrid approach, including (1) the difficulty in finding a subspace that seamlessly integrates both criteria in a single hybrid framework, and (2) the robustness of the performance (or the generalization capability of the algorithm) regarding noisy data. Existing hybrid approaches usually combine stand-alone methods of Linear Discriminant Analysis (LDA) [Martinez and Kak 2001], Principal Component Analysis (PCA) [Martinez and Kak 2001], Independent Component Analysis (ICA) [Hyvarinen 1999; Hyvarinen and Oja 2000], and their variations [Jiang 2009].

2.2.1 Supervised Methods

Linear Discriminant Analysis (LDA) [Martinez and Kak 2001] is a representative supervised dimensionality reduction method. The projection in traditional LDA [Fisher 1938; Rao 1948] is obtained by maximizing the variance between classes while minimizing the variance within class so as to achieve better separability in reduced dimensional space as follows,

$$S_B = \sum_{i=1}^{c} N_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \tag{2.25}$$

where S_B represents between-class scatter matrix. $X_i \subset X$ include N_i-many i-th class data. $\boldsymbol{\mu}_i$ is mean of data in X_i whereas $\boldsymbol{\mu}$ is mean of entire data X. c denotes the number of class in X.

$$S_W = \sum_{i=1}^{c} \sum_{\boldsymbol{x}_i \in X_i} (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^T \tag{2.26}$$

where S_W is within-class scatter matrix.

$$W^* = \underset{W}{\operatorname{argmax}} \frac{|W^T S_B W|}{|W^T S_W W|} \tag{2.27}$$

Based on S_B and S_W, the criterion for LDA is shown in (2.27). The projection W^* is chosen as the matrix which maximizes the ratio of determinant of the between-class scatter matrix to within-class scatter matrix of the projected samples. $W^* = [\boldsymbol{w}_1\ \boldsymbol{w}_2 \cdots\ \boldsymbol{w}_m]$ where $\boldsymbol{w}_i$ is n-dimensional generalized eigenvector of S_B and S_W corresponding to the i-th largest generalized eigenvalue. Fisher proved that if S_W is non-singular matrix, then the ratio $\frac{|W^T S_B W|}{|W^T S_W W|}$ is maximized when the column vectors of the projection matrix W are the eigenvectors of $S_W^{-1} S_B$ [Fisher 1938].

$$S_W^{-1} S_B \boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i \tag{2.28}$$

Therefore, W^* is obtained by solving Eq. (2.28).

LDA is extended to kernel Discriminant Analysis (kDA) [Mika et al. 1999] for nonlinear data representation using kernel trick introduced in Sec. 2.1.2. Inherited from the LDA criterion are the major issues of the small sample size (S3) problem, the common mean (CM) problem, and the robustness problem.

The small sample size often makes the within-class variance singular, so that the LDA criterion becomes infinite regardless of the between-class variance. Face recognition, for example, is a well-known application suffering from the S3 problem due to the limited number of face samples per person. Several approaches have been introduced to overcome the S3 problem such as Shrunken Centroids Regularized Discriminant Analysis (SCRDA) [Guo et al. 2007], LDA with Generalized Singular Value Decomposition (LDA/GSVD) [Howland et al. 2003; Ye et al. 2004], Null space LDA (NLDA) [Chen et al. 2000], Discriminative Common Vector (DCV) [Cevikalp et al. 2005], Orthogonal Centroid Method (OCM) [Park et al. 2003], and Weighted Piecewise LDA (WPLDA) [Kyperountas et al. 2007]. SCRDA [Guo et al. 2007] or or equivalently regularized LDA (RLDA) is proposed to resolve singularity problem in LDA by adding a constant to the diagonal elements of total scatter matrix. LDA/GSVD [Howland et al. 2003; Ye et al. 2004] applies generalized singular value decomposition to pseudo-inverse computation for between-scatter matrix for dimensionality reduction through generalized LDA criterion for minimization instead of maximization of Eq. (2.27). DCV [Cevikalp et al. 2005] is a variation of LDA using discriminative common vectors which are on the null space of within-scatter to be minimized resulting in maxi-

mum between-scatter of data in LDA criterion. For efficient computation, Gram-Schmidt orthogonalization process is applied instead of solving eigenproblem. Nonlinear extension of DCV is kernel DCV which resolves the problem of DCV inapplicable to the general problems except for small sampled one due to its assumption of discriminative common vectors on the null space of the singular within-scatter matrix. Since dataset in the kernel space is treated as inherited one with small sample problem due to hyperdimensional feature space, kernel DCV is no longer limited by the type of the problems. OCM [Park et al. 2003] only maximizes between-scatter matrix from LDA formulation so as to avoid singularity in within-scatter matrix. WPLDA [Kyperountas et al. 2007] builds piecewise linear discriminants by weighting the multiple linear discriminants from data subsets with smaller dimensionality obtained by breaking the samples down. Compared with these LDA-based criteria using the within-class variance, support vector machine (SVM) minimizes the empirical error by maximizing the separation margin which is measured by the distance from the separation hyperplane to the support vectors, nearest samples of any class. The separation margin is also regularized by additional parameter based on the nature of the data to prevent the overfitting problem from happening. The maximum separation margin and regularization lead SVM to search for the optimal trade-off between empirical error and complexity such that the decision hyperplane in SVM delivers better generalization capability for arbitrary input, resulting in robustness under noisy environment.Therefore, the lack of the sample data per class does not degrade the classification performance in SVM as significantly as in LDA due to the generalization of decision for arbitrary data.

The common mean problem is caused by non-distinguishable between-class variances from overlapped centers among different classes. As a solution, Hsieh proposed Common Mean Feature Extraction (CMFE) [Hsieh and Landgrebe 1998], Discriminant Analysis Common Mean (DACM) [Hsieh and Landgrebe 1998], and CMFE with Approximate Pairwise Accuracy Criterion (aPAC [Loog et al. 2001]) [Hsieh et al. 2006]. CMFE [Hsieh and Landgrebe 1998] is designed to reduce dimensionality with maximum ratio of the largest to the smallest class covariance so as to resolve the problem with data having common means resulting in between-scatter matrix to be zero. Since the projection onto null space of mapping vectors from LDA transforms original data into ones with common mean problem, the following CMFE over the projected data can provide additional mapping vectors regarding

to the information especially from data covariance. DACM [Hsieh and Landgrebe 1998] results from the integration of LDA and CMFE. [Loog et al. 2001] proposes Approximate Pairwise Accuracy Criterion (aPAC) by extending Fisher's LDA criterion in [Fisher 1938] to approximate weighted formulation of pairwise Fisher criteria based on one-against-one expansion of Fisher criterion. aPAC approximates the mean accuracy among pairs of classes to construct weights where the contribution of each class pair depends on the Bayes error rate. CMFE with aPAC is proposed in the same strategy of DACM but replacing LDA to aPAC with redundancy removal among features by classification accuracy estimation. SVM is not influenced by the common mean problem since structural risk in SVM does not rely on the training data center.

Robustness improvement is pursued as the other critical issue in LDA for better classification performance in noisy environment. Several methods have been proposed under the LDA framework, including Asymmetric Discriminant Analysis (ADA) [Jiang 2009] and LDA over significant nodes [Xu et al. 2004]. ADA [Jiang 2009] incorporates LDA and CMFE into single formulation with weighting parameters to adjust class asymmetry and to denote discriminatory information about class mean for robustness data representation especially against imbalanced data. [Xu et al. 2004] proposed a way for efficient classification in Discriminant Analysis by introducing recursively selected significant nodes which only include a part of original dataset without any violation against LDA's criterion. Due to SVM's complexity suppression in addition to maximum margin, the projection vectors from SVMs deliver data representation with improved robustness compared with LDA. The robustness is enhanced especially under biased and noisy environment. According to [Shashua 1999], LDA can only obtain a decision boundary identical to the one from SVM when there exist sufficiently large number of observations for effective representation of the internal structure of data.

Beyond the LDA criteria, SVM-related approaches like Recursive SVM (RSVM) [Tao et al. 2008] and Large-scale Maximum Margin Discriminant Analysis (Large-scale MMDA) [Tsang et al. 2008] have been applied for dimensionality reduction purpose. Both are based on a series of SVMs with orthogonality RSVM is motivated by Recursive LDA (RLDA) [Xiang et al. 2006] but utilizes SVM instead of LDA to iteratively extract the projection vector. Large-scale MMDA extracts projection with maximum separability by Core Vector

Machine (CVM) which provides an approximation of SVM pursuing fast computation in large-scale dataset. Although both RSVM and large-scale MMDA utilize SVM to obtain projections resulting in no struggle with the S3 or the common mean problems under improved robustness, there exists possible redundancy issue due to no analysis of the similarity among the extracted projections from the multiple series of SVMs/CVMs under orthogonal relationship.

Regression is another type of the supervised dimensionality reduction approach which finds reduced dimensional space for the input variables maximally correlated with the response variables. Regression based approaches can be categorized as supervised when the response is actually the class assignment for training data represented by the input variables. The regression model for supervised dimensionality reduction includes Partial Least Squares regression (PLS regression) [Dhanjal et al. 2009; Momma and Bennett 2006; Wold 1966], kernel Partial Least Squares regression (kPLS) [Rosipal and Trejo 2002], and Kernel Dimensionality Reduction (KDR) [Fukumizu et al. 2004]. PLS is to find linear relationship between the explanatory input and the corresponding response using the regression model by projecting the data onto reduced dimensional space consisting of latent variables based on the covariance structure analysis. However, the covariance-based analysis might lead to lower classification performance compared with stronger statistical measure of independent relationship among variables in ICA. kPLS extends the correlation measurement in covariance structure by using kernel function to provide nonlinear representation capability to reduced dimensional space. KDR extends PLS/kPLS's correlation analysis to canonical correlation analysis (CCA) in the Reproducing Kernel Hilbert Space (RKHS) to provide better statistical relationship of conditional independence between the input and the response variables. However, KDR does not provide robust data representation as shown in SVM due to the lack of generalization capability.

2.2.2 Unsupervised Methods

The data correlation and independence are representative unsupervised dimensionality reduction criteria to deliver the nature of data into the reduced dimensional space. Principal Component Analysis (PCA) [Martinez and Kak 2001; Pearson 1901] seeks a projection

which maximally uncorrelates data in a least-squares sense as follows,

PCA [Martinez and Kak 2001] is based on linearly projecting raw data to a low dimensional feature spaces which yields projection directions that maximize the total scatter across all class resulting in minimum squared-error.

$$W^* = \underset{W}{\operatorname{argmax}} \, |W^T S_T W| \tag{2.29}$$

$$S_T = \sum_{i=1}^{N} (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^T \tag{2.30}$$

where $\boldsymbol{x}_i$ is i-th n-dimensional data among N-many dataset. When m is the dimension of feature vector $\boldsymbol{s}$ satisfying $m \leq n$, $\boldsymbol{s}_i = W^{*\mathrm{T}} \boldsymbol{x}_i \in R^m$ where W^* represents the mapping with optimal scatter of the features described by $W^T S_T W$. Based on pre-determined m, $W = [\boldsymbol{w}_1 \ \boldsymbol{w}_2 \cdots \boldsymbol{w}_m]$ where $\boldsymbol{w}_i \in R^n$. The projection W^* is chosen to maximize the determinant of the total scatter matrix of the projected samples. Fisher proved that $|W^T S_T W|$ is maximized when the column vectors of the projection matrix W are the eigenvectors of S_T [Fisher 1938].

$$S\boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i \tag{2.31}$$

By the fisher's proof, the optimal mapping for squared error criterion is denoted by eigenvalue decomposition in (2.31). The number of eigenvectors $\boldsymbol{w}_i$ corresponding to eigenvalues λ_i in descending order determines the amount of the error by features based on PCA. A drawback of this approach is that the scatter S represented by data correlation to be maximized is not only due to between-class scatter which is useful for classification, but also due to within-class scatter which is unwanted information for better classification accuracy.

To improve the data representation capability of PCA, there exist various approaches such as PCA with L1-norm, kernel Component Analysis (KCA), 2-dimensional PCA (2DPCA), Multi-linear PCA (MPCA), and manifold based PCA. [Kwak 2008] incorporates L1-Norm into PCA for distance measurement to achieve robustness and rotational invariance in PCA framework. It also provides proof of global optimal solution to be obtained based on PCA with L1-norm. For the robustness in PCA, [Alzate and Suykens 2008] proposed KCA

based on kernel PCA with "LS-SVM"-like formulation with robust loss function which consists of the Huber and the epsilon-insensitive loss function in SVR [Vapnik 1999] for robust dimensionality reduction with sparsity. 2DPCA [Yang et al. 2004] is proposed to alleviate computation burden by directly using of two-dimensional image matrix instead of lexicographical representation resulting in the inapplicable covariance matrix for eigenvalue decomposition in PCA. [Xu et al. 2008] also proposes two schemes of 2DPCA where the first scheme enhances the transverse characters of images and the second one improves vertical characters of images with theoretical analysis of traditional 2DPCA. The features from the two schemes are utilized to classify arbitrary data based on distance measurement. MPCA [Lu et al. 2008] is for 3-dimensional tensor object dimensionality reduction by directly utilizing the tensor representation in the algorithm framework. MPCA also includes tensor classification strategy by weighting. Data manifold analysis is applied to the extension of PCA for subspace segmentation such as Probabilistic Principal Component Analysis (PPCA) [Archambeau et al. 2008; Tipping and Bishop 1999a,b; Wang and Wang 2006], clustered data based PPCA [Sanguinetti 2008], and Generalized Principal Component Analysis (GPCA) [Ma et al. 2008; Vidal et al. 2005]. PPCA [Tipping and Bishop 1999a,b] is proposed as a probabilistic interpretation of PCA through latent variable. It proved that the principal subspace of the data is spanned with placing a spherical unitary normal prior on the latent variable by the mapping vectors at maximum likelihood through Expectation-Maximization (EM), resulting in the generative model with mean vector and noise being able to provide a probabilistic equivalent of PCA. Robust PPCA [Archambeau et al. 2008] tried to overcome the problem of PPCA inherited from Gaussian noise model which is sensitive to atypical outliers. To achieve the robustness, it replace gaussian to Student-t density to formulate maximum likelihood estimation, where Student-t density includes additional parameter to regulate the thickness of the distribution tails so as to reduce the sensitive to outliers. The PPCA series has theoretical equivalence with subspace in an aspect of Gaussian density estimation shown in [Wang and Wang 2006] although these arise from different motivation: PPCA integrates the condition density in the latent space over maximum likelihood framework whereas subspace method minimizes the Kullback-Leibler (KL) divergence between principal and orthogonal subspaces. Clustered data based PPCA [Sanguinetti 2008] is proposed similar to PPCA but over clustered data

based on the latent variables with principal and orthogonal subspace matrix representation over maximum likelihood with EM. It shows that the likelihood of the proposed model is a monotonic function of Rayleigh's coefficient, so that LDA can be retrieved as a subset of Clustered data based PPCA. GPCA [Ma et al. 2008; Vidal et al. 2005] is unsupervised multiple manifolds searching algorithms in geometric point of view through polynomial data embedding. Although the manifold-based approaches showed improved data representation capability, the methods usually suffered from the sensitivity to free variables such as the segmented subspace dimensionality as well as computational complexity.

Independent Component Analysis (ICA) [Hyvarinen 1999; Hyvarinen and Oja 2000] maximizes the independence among components based on the independence measure such as mutual information. Generally, ICA provides more intrinsic information resulting generally in contributing more to performance improvement than PCA with maximum uncorrelatedness [Yang et al. 2005, 2007]. FastICA is one of the representative independence maximization approach based on non-Gaussianity in linear mixing model. The source s is acquired by linear transformation, $\boldsymbol{s} = W^T\boldsymbol{x}$ using unmixing matrix, W with observation $\boldsymbol{x}$. Based on Central Limit Theorem, a sum of two independent random variable with identical distribution has a distribution that has less non-Gaussianity than any of the two original random variables. Therefore, $\boldsymbol{w}$ as a part of W is then taken for maximizing the non-Gaussianity of $\boldsymbol{w}^T\boldsymbol{x}$ since $\boldsymbol{w}^T\boldsymbol{x}$ is least gaussian when there exists only one non-zero weight for s_i, $i \in \{1, \cdots, n\}$ based on Central Limit Theorem. Kurtosis is a classical quantitative measure for non-Gaussianity.

$$kurt(\boldsymbol{s}) = E\{\boldsymbol{s}^4\} - 3(E\{\boldsymbol{s}^2\})^2 \tag{2.32}$$

where $kurt(\boldsymbol{s})$ represents kurtosis measure for $\boldsymbol{s}$. Kurtosis can simply be estimated by the fourth moment of $\boldsymbol{s}$. Although kurtosis is well-defined measure due to its computational and theoretical simplicity, it is not a robust measure for non-Gaussianity due to the sensitivity oven the given data [Hyvarinen and Oja 2000]. Negentropy is another measure for non-Gaussianity defined by

$$J(\boldsymbol{s}) = H(\boldsymbol{s}_{\text{gaussian}}) - H(\boldsymbol{s}) \tag{2.33}$$

where the entropy is defined by $H(Y) = -\sum_i P(Y = a_i) log P(Y = a_i)$. $\boldsymbol{s}_{\text{gaussian}}$ denotes gaussian random variable. Since gaussian random variable has the largest entropy in all random variables of equal variance, negative entropy can be treated as a measure of non-Gaussianity to be maximized. Therefore, $J(\boldsymbol{s})$ is always positive or zero. Although negentropy has well-justified statistical theory, it requires pdf estimation resulting in high computational complexity. To reduce computational difficulty, there exists an approximation of negentropy by

$$J(\boldsymbol{s}) \approx \sum_i k_i [E\{G_i(\boldsymbol{s})\} - E\{G_i(\boldsymbol{s}_{\text{gaussian}})\}]^2 \tag{2.34}$$

where k_i is positive constant. G_i's for $i = 1, 2$ are non-quadratic functions defined by

$$G_1(\boldsymbol{s}) = \frac{1}{a_1} \log \cosh (a_1 \boldsymbol{s}) \tag{2.35}$$

$$G_2(\boldsymbol{s}) = -\exp\left(-\frac{\boldsymbol{s}^2}{2}\right) \tag{2.36}$$

where $1 \leq a_1 \leq 2$. G_1 and G_2 are heuristically chosen non-quadratic functions by [Hyvarinen and Oja 2000]. The maximization of negentropy has equivalent relationship with minimization of mutual information [Hyvarinen and Oja 2000]. Therefore, it can also be called by independence maximization based on mutual information due to the equivalence.

The major improvement in ICA occurs at the independent measure represented by the fixed nonlinear function in FastICA [Hyvarinen 1999] to the function built by nonlinear search through kernel in Reproducing Kernel Hilbert Space (RKHS) by Kernel Canonical Component Analysis (kernel CCA) and kernel Generalized Variance (kGV) [Bach and Jordan 2002; Fukumizu et al. 2004] which formulate canonical correlation in RKHS so as to provide characterizations of general notions of independence among data for linear unmixing matrix/projection. Kernel ICA (kICA) utilizes different measure compared with FastICA, F-correlation based on canonical correlation analysis (CCA) to adopt mapping of source into feature space using kernel technique although kICA starts at the same concept of independence maximization of source. The kernel method helps to search the function space instead of utilizing the heuristic non-quadratic functions of G_1 and G_2. Since pair-wise zero F-correlation is equivalent that variables are pair-wise independent,

minimization problem of CCA using F-correlation can be considered as a method for achieving maximum independence [Bach and Jordan 2002]. The definition of F-correlation, ρ to obtain correlation between two mappings of data having two variables in N dimensional space is represented as follows,

$$\begin{aligned}\rho&=\max\ corr(\langle\phi(x^{(1)}),f_1\rangle,\langle\phi(x^{(2)}),f_2\rangle)\\&=\max\ \frac{\mathrm{cov}(\langle\phi(x^{(1)}),f_1\rangle,\langle\phi(x^{(2)}),f_2\rangle)}{\{\mathrm{var}(\langle\phi(x^{(1)}),f_1\rangle)\}^{1/2}\{\mathrm{var}(\langle\phi(x^{(2)}),f_2\rangle)\}^{1/2}}\end{aligned} \tag{2.37}$$

where f_i represents a spanned subspace in feature space, F. $\boldsymbol{x}$ is a given data and superscript i in $x^{(i)}$ denotes the variable index in the observation space. Based on reproducing property of the kernel in Hilbert spaces, $f(\boldsymbol{x}) = \langle K(\cdot,\boldsymbol{x}),f\rangle$ where $K(\cdot,\boldsymbol{x}) = \phi(\boldsymbol{x})$ for ϕ satisfying Mercer's theorem. The correlation in Eq. (2.37) is equivalent to $corr(f_1(x^{(1)}),f_2(x^{(2)}))$ denoting correlation between mapping of two variables $x^{(1)}$ and $x^{(2)}$ onto f_1 and f_2, respectively. Since $f_1,f_2 \in F$, $f_1 = \sum_{i=1}^{N}\beta_i^{(1)}\phi(x_i^{(1)}) + f_1^{\perp}$ and $f_2 = \sum_{i=1}^{N}\beta_i^{(2)}\phi(x_i^{(2)}) + f_2^{\perp}$ where $f_1^{\perp}$ and $f_2^{\perp}$ are orthogonal to linear spaces spanned by the $\phi(x_i^{(j)})$ representing $\sum_{i=1}^{N}\beta_i^{(1)}\phi(x_i^{(1)})$ and $\sum_{i=1}^{N}\beta_i^{(2)}\phi(x_i^{(2)})$. A subscript of $\boldsymbol{x}$ denotes observation index. The numerator as covariance in Eq. (2.37) is therefore expanded as,

$$\mathrm{cov}(\langle\phi(x^{(1)}),f_1\rangle,\langle\phi(x^{(2)}),f_2\rangle) = \frac{1}{N}(\boldsymbol{\beta}^{(1)})^T K_1 K_2 \boldsymbol{\beta}^{(2)} \tag{2.38}$$

where $K_r = [k_{ij}] = [K(\phi(x_i^{(r)}),\phi(x_j^{(r)}))]$ is the so called Gram matrix. The denominator as variance in Eq. (2.37) is represented in the same way of Eq. (2.38) as,

$$\mathrm{var}(\langle\phi(x^{(1)}),f_1\rangle) = \frac{1}{N}(\boldsymbol{\beta}^{(1)})^T K_1^2 \boldsymbol{\beta}^{(1)} \tag{2.39}$$

$$\mathrm{var}(\langle\phi(x^{(2)}),f_2\rangle) = \frac{1}{N}(\boldsymbol{\beta}^{(2)})^T K_2^2 \boldsymbol{\beta}^{(2)} \tag{2.40}$$

The substitution of Eq. (2.38), (2.40), and (2.40) into Eq. (2.37) results in the following F-correlation.

$$\rho = \max\ \frac{(\boldsymbol{\beta}^{(1)})^T K_1 K_2 \boldsymbol{\beta}^{(2)}}{\{(\boldsymbol{\beta}^{(1)})^T K_1 K_1 \boldsymbol{\beta}^{(1)}\}^{1/2}\{(\boldsymbol{\beta}^{(2)})^T K_2 K_2 \boldsymbol{\beta}^{(2)}\}^{1/2}} \tag{2.41}$$

The generalized eigenvalue problem in CCA is adopted to form a same type of the problem

for Eq. (2.41) as follows,

$$\begin{bmatrix} K_1^2 & K_1K_2 \\ K_2K_1 & K_2^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{\beta}^{(1)} \\ \boldsymbol{\beta}^{(2)} \end{bmatrix} = \lambda \begin{bmatrix} K_1^2 & 0 \\ 0 & K_2^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{\beta}^{(1)} \\ \boldsymbol{\beta}^{(2)} \end{bmatrix} \tag{2.42}$$

where λ denotes eigenvalue for the generalized eigenvalue problem. Since the first canonical correlation, max (λ) is equivalently found by the problem of finding min (λ), the problem to maximize ρ in Eq. (2.41) is now represented by the problem for finding min (λ) in Eq. (2.42). However, invertible matrixes of K_1 and K_2 always result in $\rho = 1$. Therefore, regularization is required for Eq. (2.37) as follows,

$$\rho_\kappa = \max \frac{\mathrm{cov}(\langle \phi(x^{(1)}), f_1 \rangle, \langle \phi(x^{(2)}), f_2 \rangle)}{\{\mathrm{var}(\langle \phi(x^{(1)}), f_1 \rangle) + \kappa \|f_1\|^2\}^{1/2} \{\mathrm{var}(\langle \phi(x^{(2)}), f_2 \rangle) + \kappa \|f_2\|^2\}^{1/2}} \tag{2.43}$$

where κ is a small positive constant. By second order estimation of the norm, $\|f_i\|^2$, with a finite sample, the variance in the denominator of Eq. (2.43) is denoted by ignoring constant term as,

$$\begin{aligned} \mathrm{var}(\langle \phi(x^{(j)}), f_j \rangle) + \kappa \|f_j\|^2 =& \frac{1}{N} (\boldsymbol{\beta}^{(j)})^T K_j^2 \boldsymbol{\beta}^{(j)} + \kappa (\boldsymbol{\beta}^{(j)})^T K_j \boldsymbol{\beta}^{(j)} \\ \approx& \frac{1}{N} (\boldsymbol{\beta}^{(j)})^T (K_j + \kappa \frac{N}{2} I)^2 \boldsymbol{\beta}^{(j)} \end{aligned} \tag{2.44}$$

Finally, I obtain a generalized eigenvalue problem with regularization using the variance in Eq. (2.44) as follows,

$$C\beta = \lambda D\beta \tag{2.45}$$

where $C = [c_{ij}]$, $c_{ij} = (K_i + \kappa(N/2)I)^2$ for $i = j$, $c_{ij} = K_iK_j$ otherwise. $D = [d_{ij}]$, $d_{ij} = (K_i + \kappa(N/2)I)^2$ for $i = j$, c_{ij} otherwise with $i, j = \{1, 2\}$. To generalize the two-variable problem to more than two variables, Eq. (2.45) is simply extended with $i, j = \{1, 2, \cdots, m\}$ using pair-wise independence over entire variables. If I assume the variable follows Gaussian distribution, the problem to find the minimum eigenvalue in Eq. (2.45) is interpreted as an equivalent problem of minimizing mutual information. The link between

canonical correlation and mutual information [Bach and Jordan 2002] is denoted as,

$$\begin{aligned} I(x^{(1)}, x^{(2)}, \cdots, x^{(m)}) &= -\frac{1}{2}\sum_{i=1}^{m}\lambda_i \\ &= -\frac{1}{2}\log\frac{\det(C)}{\det(D)} \end{aligned} \tag{2.46}$$

Therefore, the problem of pursuing minimum eigenvalue is translated by minimization of Eq. (2.46) and is especially known as kernel Generalized Variance (kGV). Since the problem is based on the source obtained through the unmixing process by $\boldsymbol{s} = W^{\mathrm{T}}\boldsymbol{x}$, kICA is finally represented by a function of W under given observation.

$$W^* = \underset{W}{\operatorname{argmin}}\ g(W) \tag{2.47}$$

$$g(W) = -\frac{1}{2}\log\frac{\det(C)}{\det(D)} \tag{2.48}$$

Although kICA show better data independence, kICA are extremely slow compared with FastICA due to the computationally burdensome gradient calculation of Eq. (2.48).

Since the unsupervised approaches focus on searching for the better data representation without separability concern, the lack of consideration in separability might limit the core information for for classification performance improvement to be delivered into reduced dimensional space.

2.2.3 Hybrid Methods

The hybrid dimensionality reduction consists of both supervised and unsupervised criteria so as to find better data representation for classification performance improvement compared with either the supervised or unsupervised method. The conventional hybrid approaches improve/resolve various problem with limitation. Asymmetric Principal and Discriminant Analysis (APCDA) [Jiang 2009] alleviates common mean problem and improves robustness since APCDA combines Asymmetric Discriminant Analysis (ADA) in the Asymmetric PCA (APCA) subspace where ADA incorporates LDA and CMFE into single formulation with weighting and APCA utilizes asymmetric pooled covariance matrix regulated by class covariance reliability for unbalanced amount of data per class. LDA over

PCA [Belhumeur et al. 1997; Yang and Yang 2001, 2003] aims at alleviation of S3 problem in LDA criteria by null space elimination through PCA, so that it only eliminate disadvantage of LDA criteria from S3 problem. ICA augmented by LDA [Kwak and Pedrycz 2007] consists of sequential combination of PCA, ICA, and LDA to reduce dimensionality by LDA over ICA subspace constrained by PCA. The common mean problems still resides in ICA augmented by LDA due to between-class variance in ICA augmented by LDA. The supervised MI-based ICA [Leiva-Murillo and Artes-Rodriguez 2007] proposes supervised one-unit projection vector extraction which maximizes mutual information between the extracted components and the data classes. Due to the method only with regularization of classes on ICA, the supervised Mutual Information(MI)-based ICA does not sufficiently incorporate separability into the hybrid framework. Discriminant Nonnegative Matrix Factorization (DNMF) [Zafeiriou et al. 2006] is a hybrid of NMF plus LDA represented by NMF formulation [Lee and Seung 1999] constrained by LDA criteria where within- and between-class variance are from the decompositions of NMF. Nonnegative Tensor Factorization (NTF) with LDA [Zafeiriou 2009] extends DNMF to 3-dimensional tensor with arbitrary valence based on within- and between-class variance by tensor decompositions from NTF. However, in DNMF and NTF with LDA, S3 and common mean problems inherited from LDA criteria nullify the LDA's discriminant characteristic from the hybrid frameworks since DNMF and NTF with LDA include LDA criteria as a part of their cost functions.

The traditional hybrid approaches introduced above can be categorized either into subspace-based or objective-level hybridization. The subspace-based method utilizes subspace in between the supervised and unsupervised criteria to construct single algorithm. For example, Asymmetric Principal and Discriminant Analysis (APCDA) performs ADA in APCA subspace [Jiang 2009]. LDA over PCA removes null space by PCA for LDA. ICA augmented by LDA build subspace by ICA for LDA. The objective-level hybridization usually adopts supervised information into unsupervised criterion. The supervised MI-based ICA, DNMF, and NTF rely on the objective-level hybridization. The supervised MI-based ICA utilizes the class label in the dataset during the mutual information maximization. The objective functions in DNMF and NTF are both constrained by LDA criterion to incorporate the supervised discriminant information into unsupervised matrix/tensor factorization.

Although the conventional hybrid approaches try to provide improved data representation capability, most of them in both subspace-based and objective-level hybridization schemes are based on LDA-like criteria consisting only of between- and within-class covariance analysis which result in S3 and common mean problem with robustness concern. Therefore, it is required to improve the classification performance in the reduced dimensional space that the better supervised and unsupervised criteria be incorporated into the concept of hybrid dimensionality reduction.

2.3 Constrained Optimization

The constraints in optimization problem is generally managed either by deterministic or stochastic approach. The deterministic approach is usually focused on primal and dual formulation based on Lagrange multipliers [Ciarlet 1989] whereas the stochastic approach relies on the stochastic search such as genetic algorithm [Goldberg 1989] with constraints treated as independent objectives [Tan et al. 2005, 2003].

2.3.1 Deterministic Approach

Introducing Lagrange multiplier is a traditional deterministic approach to handle constraints in optimization problem where the differentiable objective function, $J : \Omega \rightarrow \mathcal{R}$ has a relative minimum as $J(\boldsymbol{u}) \leq J(\boldsymbol{v})$ at a point $\boldsymbol{u}$ for every $\boldsymbol{v}$. Let $\boldsymbol{u} = (u_1, u_2)$ is a point of the set, $U = \{(v_1, v_2) \in \Omega : \ \varphi(v_1, v_2) = 0\} \subset \Omega$ where Ω is an open subset of a product $V_1 \times V_2$ of normed vector spaces, the space V_1 being complete, and $\varphi : \Omega \rightarrow V_2$ is a function over Ω. $\partial\varphi(u_1, u_2) \in \text{Isom}(V_2)$. If J has a relative minimum at $\boldsymbol{u}$ with respect to the set U, then there exist an element $\Lambda(\boldsymbol{u})$ such that

$$J'(\boldsymbol{u}) + \Lambda(\boldsymbol{u})\varphi'(\boldsymbol{u}) = 0 \tag{2.49}$$

Eq. (2.49) is further expanded for each of the φ_i's, $\forall i$ as follows,

$$J'(\boldsymbol{u}) + \sum_i \lambda_i(\boldsymbol{u})\varphi_i'(\boldsymbol{u}) = 0 \tag{2.50}$$

where φ_i's are linearly independent constraints. λ_i's in Eq. (2.50) are called Lagrange multipliers associated with the constrained minimum $\boldsymbol{u}$. By introducing Lagrange multipliers, the differentiable function, J at $\boldsymbol{u}$ can incorporate the constraints into single formulation as Eq. (2.50), and the $\boldsymbol{u}$ for the minimum of J is found by solving Eq. (2.50) with $\varphi_i'(\boldsymbol{u}) = 0$, $\forall i$. For the problem as follows,

$$\begin{aligned} \boldsymbol{u} \in U = \{\boldsymbol{v} \in V : \varphi_i(\boldsymbol{v}) \leq 0, 1 \leq i \leq m\} \\ J(\boldsymbol{u}) = \inf_{\boldsymbol{v} \in U} J(\boldsymbol{v}) \end{aligned} \tag{2.51}$$

The point $\boldsymbol{u}$ belonging to the set of U is a solution of the problem in Eq. (2.51) if $(\boldsymbol{u}, \boldsymbol{\lambda}) \in V \times R_+^m$ is a saddle point of L where R_+ denotes semi-positive subspace of R. Additionally, there exists at least one vector $\boldsymbol{\lambda}$ such that the pair $(\boldsymbol{u}, \boldsymbol{\lambda})$ is a saddle point of L when J is convex and differentiable at $\boldsymbol{u}$, the solution of Eq. (2.51). The constrained minimization problem in Eq. (2.51) is represented by Lagrange multiplier in single formulation as follows,

$$L(\boldsymbol{v}, \boldsymbol{\mu}) = J(\boldsymbol{v}) + \sum_i \mu_i \varphi_i(\boldsymbol{v}) \tag{2.52}$$

where L is called 'Lagrangian'. Therefore, if $\boldsymbol{\lambda}$ is known, then Eq. (2.51) becomes unconstrained problem at the saddle point in the representation of Eq. (2.52) as follows,

$$\begin{aligned} L(\boldsymbol{u}_\lambda, \boldsymbol{\lambda}) &= \inf_{\boldsymbol{v} \in V} L(\boldsymbol{v}, \boldsymbol{\lambda}) \\ &= \sup_{\boldsymbol{\mu} \in R_+^m} \inf_{\boldsymbol{v} \in V} L(\boldsymbol{v}, \boldsymbol{\mu}) \end{aligned} \tag{2.53}$$

where $\boldsymbol{u}_\lambda$ denotes $\boldsymbol{u}$ with the given $\boldsymbol{\lambda}$. Eq. (2.53) is represented as maximization instead of minimization as follows,

$$G(\boldsymbol{\lambda}) = \sup_{\boldsymbol{\mu} \in R_+^m} G(\boldsymbol{\mu}) \tag{2.54}$$

where $G(\boldsymbol{\mu}) = \inf_{\boldsymbol{v} \in V} L(\boldsymbol{v}, \boldsymbol{\mu})$ is called the dual problem of the primal problem in Eq. (2.51). The primal and dual solution are identical when there primal/dual has unique solution. The duality is helpful to convert the optimization problem in easier formulation as shown in Eq. (2.17) for the formulation of SVM.

2.3.2 Stochastic Approach

Stochastic optimization methods are optimization algorithms which incorporate probabilistic (random) elements, either in the problem data (the objective function, the constraints, etc.), or in the algorithm itself (through random parameter values, random choices, etc.), or in both. The concept contrasts with the deterministic optimization methods, where the values of the objective function are assumed to be exact, and the computation is completely determined by the values sampled so far.

This section introduces evolutionary algorithm and its multiobjective extension for constraint handling. Evolution in optimization is an approach to overcome local minimum problem which usually results from gradient-based search such as steepest descent and Newton's method. The search algorithms based on evolution utilize multiple search point whereas conventional gradient-based approaches use single search point toward gradient decreasing. Genetic algorithm (GA) as one of the evolutionary optimization techniques performs single objective optimization successfully for many engineering problems. Extension for multiobjective problem based on GA also provides promising performance due to the isolation between data and search space.

Genetic Algorithm

Genetic algorithm is an optimization algorithm which mimic evolution in nature. For example, there is a question, "Why is giraffe's neck long?". The answer in an aspect of evolution is that giraffes with longer neck can have more change to survive since they can reach to leaves in taller tree to feed themselves. Therefore, longer neck is treated as dominant characteristic for their descendant by nature. The evolution can be applied as an optimization algorithm in similar way by presenting the gene from real world input, generating various possibility by search operators, and providing search direction and speed. Holland [Holland 1992] introduced schema theory which is widely accepted as a basis of genetic algorithm although it does not provide a strict proof as a global optimum finder. Schema theory shows convergence from one generation to the next based on building block hypothesis (BBH). BBH attempts to explain how GA solves a problem by positing that near optimal solutions were forged from small, low-order, above-average schemata which is

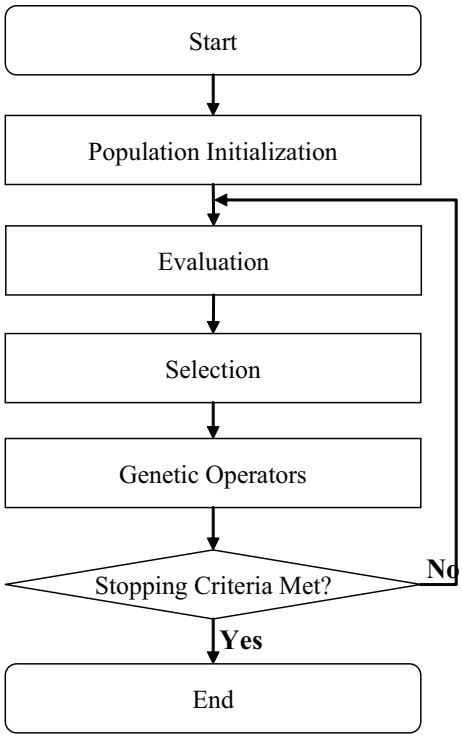


Figure 2.3: Genetic algorithm framework

a template allowing exploration of similarities among chromosomes.

GA consists of 4 major elements: encoding, evaluation, selection, and operators. Encoding provides a scheme to map between phenotype and genotype. By isolation of search space represented by genotype from the data space by phenotype, GA is easily facilitated into wide range of applications by appropriate encoding scheme. Figure 2.4 denotes genetic algorithm framework based on pre-determined encoding scheme to build internal structure of individuals. Evaluation denotes a measure for objective function. Based on the result from evaluation, selection gives overall search direction for individuals to be gradually improved. Operators perturb the location of individuals so as to search better candidate for next generation.

Encoding is one of the essential components for GA to perform successfully since optimization is performed in the search space converted from solution space by encoding scheme. [Ronald 1997] provides 9 ideal encoding features. Most problems are not able to fit all these requirements but adopt compromising encoding. Figure 2.4 shows conversion between coding space and solution space by encoding and decoding. There are two

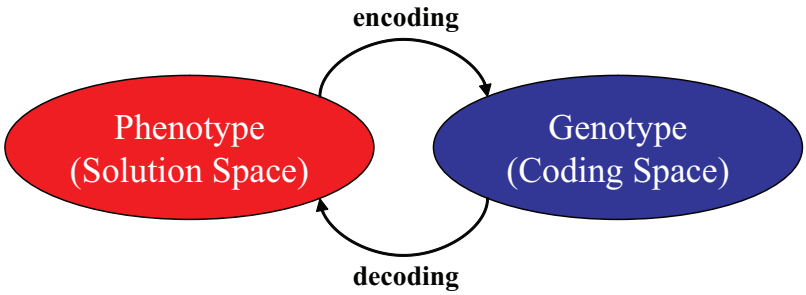


Figure 2.4: Encoding and decoding in Genetic Algorithm

distinct encoding scheme, binary and real encoding. Binary encoding utilizes only 0 or 1 to consist of individuals whereas real encoding directly use real value of which individuals are composed. Real-valued encoding provides much higher precision than binary whereas binary offers lower computational complexity than real encoding due to restricted search space by 0 and 1.

A stochastic selection by the roulette wheel method is a basic selection/reproduction mechanism used frequently in the genetic algorithm. The roulette wheel selection method is based on the fitness ratio, which has some weaknesses. In early stage of evolution, a chromosome with a larger fitness value than other chromosomes has a high survival probability in the reproduction process, which might cause premature convergence. Also, when individuals converge to near solution, an average fitness might be close to the populations best fitness. If this is the situation, the solution candidates with average and best fitness will have nearly the same number of copies in future generations. Then competition between individuals by genetic operators becomes low, and so individuals wander around the solution.

To overcome this problem, one can reduce the relatively high fitness values of the individual chromosomes, and the fitness difference between the chromosomes can be scaled by the distribution of the individual state of all fitness. The fitness scaling and ranking methods [Michalewicz 1996] are the representative solutions for the problem [Goldberg 1989]. The fitness ranking method ranks the chromosomes by fitness values and then redistributes fitness exponentially according to rank. The fitness ranking method does not consider the relation between object function and fitness. The fitness scaling method scales all fitness using maximum, minimum, and average fitness by a linear function. The fitness

scaling considers the state of all fitness, but if the average fitness is close to the maximum fitness, a fitness less than the average fitness can be evaluated as a negative value.

Tournament selection [Goldberg et al. 1991] is a selection scheme to reproduce a child by competing among the candidate set which includes more than one individuals randomly picked from parents. The size of the candidate set is called as tournament size. Larger tournament size results in faster convergence. Tournament selection is heuristically known as a selection method including the advantages of both ranking and scaling based selection especially against genetic drift.

There are two operators widely utilized, crossover and mutation. Crossover is to exchange information between individuals. Since crossover does not provide any new external information, it is usually treated as local search operator. Mutation, on the other hand, is to insert new information into individuals as a random perturbation resulting in "big jump" for global search.

Extension to Multiobjective Optimization for Constraint Handling

Since genetic algorithm isolates data and search space resulting in providing flexibility to perform optimization regardless of problems, genetic algorithm can easily extended for problem with multiple objectives, so called multiobjective optimization problem. The major difference between single and multiple objective case is that additional measure is required to determine dominance among individuals during multiobjective optimization. Pareto optimality [Steuer 1986] is a major trend of dominance determination for evolutionary multiobjective optimization although there are several different dominance determination measures such as weight [Hajela and Lin 1992], minmax [Coello and Christiansen 1999], and sub-population [Richardson et al. 1989; Schaffer 1985]based algorithms. The review for evolutionary multiobjective optimization here is focused on methods based on Pareto optimality. The detail of the Pareto optimality is as follows,

1. Global Pareto optimality: A decision vector $\boldsymbol{x}^* \in S$ is global Pareto optimal if there does not exist another decision vector $\boldsymbol{x} \in S$ such that $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^*)$ for $\forall i$ and $f_j(\boldsymbol{x}) \leq f_j(\boldsymbol{x}^*)$ for at least one index j. $\boldsymbol{x}^* \in S$ also is global Pareto optimal by Pareto dominance if there

does not exist another $\boldsymbol{x} \in S$ which dominate $\boldsymbol{x}^*$.

2. Local Pareto optimality: $\boldsymbol{x}^* \in S$ is locally Pareto optimal if there exist $\delta > 0$ such that $\boldsymbol{x}^*$ is Pareto optimal in $S \cap B(\boldsymbol{x}^*, \delta)$ where $B(\boldsymbol{x}^*, \delta) = \{\boldsymbol{x} \in R^n | \|\boldsymbol{x}^* - \boldsymbol{x}\| < \delta\}$.

3. Local Pareto optimum in convex problem: Let $\boldsymbol{x}^* \in S \cap B(\boldsymbol{x}^*, \delta)$ be local Pareto optimum. If $\boldsymbol{x}^*$ is not globally Pareto optimal, then there exist some other point, $\boldsymbol{x}^o \in S$ which is more optimized than $\boldsymbol{x}^*$. Let $\hat{\boldsymbol{x}} = \beta \boldsymbol{x}^o + (1 - \beta)\boldsymbol{x}^*$, where $0 < \beta < 1$ is selected such that $\hat{\boldsymbol{x}} \in B(\boldsymbol{x}^*, \delta)$, then there does not exist any dominant points in B by the definition of Pareto optimality. By the convexity of the objective functions and global Pareto optimality, $f_i(\hat{\boldsymbol{x}}) \leq \beta f_i(\boldsymbol{x}^o) + (1 - \beta) f_i(\boldsymbol{x}^*) \leq \beta f_i(\boldsymbol{x}^*) + (1 - \beta) f_i(\boldsymbol{x}^*) = f_i(\boldsymbol{x}^*)$, $\forall i$. Because $\boldsymbol{x}^*$ is locally Pareto optimal and $\hat{\boldsymbol{x}} \in B(\boldsymbol{x}^*, \delta)$, $f_i(\hat{\boldsymbol{x}}) = f_i(\boldsymbol{x}^*)$, $\forall i$. Further, $f_i(\boldsymbol{x}^*) \leq \beta f_i(\boldsymbol{x}^o) + (1 - \beta) f_i(\boldsymbol{x}^*)$ for $\forall i$. Because of $\beta > 0$, $f_i(\boldsymbol{x}^*) \leq f_i(\boldsymbol{x}^o)$ for $\forall i$. According to global Pareto optimality, $(f_i(\boldsymbol{x}^*) > f_i(\boldsymbol{x}^o)$ for some i. Contradiction. Thur, $\boldsymbol{x}^*$ is globally Pareto optimal.

4. Local Pareto optimum in quasiconvex problem: Let $\boldsymbol{x}^* \in S \cap B(\boldsymbol{x}^*, \delta)$ be local Pareto optimum. If $\boldsymbol{x}^*$ is not globally Pareto optimal, then there exist some other point, $\boldsymbol{x}^o \in S$ which is more optimized than $\boldsymbol{x}^*$. Let $\hat{\boldsymbol{x}} = \beta \boldsymbol{x}^o + (1 - \beta)\boldsymbol{x}^*$, where $o < \beta < 1$ is selected such that $\hat{\boldsymbol{x}} \in B(\boldsymbol{x}^*, \delta)$. There does not exist any dominant points in B by the definition of Pareto optimality. By $f_i(\boldsymbol{x}^o) \leq f_i(\boldsymbol{x}^*)$ for $\forall i$ and $f_j(\boldsymbol{x}^o) < f_j(\boldsymbol{x}^*)$ for some j, and the quasiconvexity of the objective functions, respectively, for each index i such that $f_i(\boldsymbol{x}^o) = f_i(\boldsymbol{x}^*)$, it is obtained that $f_i(\hat{\boldsymbol{x}}) \leq \max[f_i(\boldsymbol{x}^o), f_i(\boldsymbol{x}^*)] = f_i(\boldsymbol{x}^*)$ (i.e. $f_i(\hat{\boldsymbol{x}}) \leq f_i(\boldsymbol{x}^*)$ if f_i is quasiconvex and for each index j such that $f_j(\boldsymbol{x}^o) < f_j(\boldsymbol{x}^*)$, $f_j(\hat{\boldsymbol{x}}) \leq \max[f_j(\boldsymbol{x}^o), f_j(\boldsymbol{x}^*)] = f_j(\boldsymbol{x}^*)$ (i.e. $f_j(\hat{\boldsymbol{x}}) \leq f_j(\hat{\boldsymbol{x}}) \leq f_j(\boldsymbol{x}^*)$ if f_j is quasiconvex. Because at least one of the objective functions is strictly quasiconvex, at least one of the inequalities above is strict. Contradition with local Pareto optimality of $\boldsymbol{x}^*$. Thus, $\boldsymbol{x}^*$ is globally Pareto optimal.

There exist various evolutionary multiobjective optimization algorithms based on Pareto optimality. Multiobjective Genetic Algorithm (MOGA) [Fonseca and Fleming 1993] is one of the representative evolutionary algorithm for the problem with multiple objectives.

MOEA is based on Pareto ranking and fitness sharing [Goldberg 1989] in objective domain.

$$f_i = (1 + q_i)^{-1} \tag{2.55}$$

where q_i denotes the number of individuals dominating i-th individual in objective domain. Smaller q_i, higher f_i. The fitness sharing is originally a method to equivalently evaluated optima by measuring distance among near individuals in coding space. MOGA utilizes sharing not in coding space, but in objective space so as to find more accurate Pareto front by applying orthogonal pressure with the pressure directed toward Pareto front in selection process as follows,

$$f_i' = \frac{f_i}{\sum_i S(i,j)} \tag{2.56}$$

$$S(i,j) = \begin{cases} 1 - [d(f_i, f_j)/\sigma]^{\alpha} & \text{if } d(f_i, f_j) < \sigma_{share} \\ 0 & \text{otherwise} \end{cases} \tag{2.57}$$

where f_i' is fitness with sharing for MOGA. $d(f_i, f_j) = \|f_i - f_j\|_2$. α controls the shape of S and σ_{share} is to decide the radius for sharing process to be applied. The key of Niched Pareto Genetic Algorithm (NPGA) [Horn et al. 1994] is the utilization of Pareto dominance tournament for selection. The selection scheme is based on tournament approach with two individuals and subpopulation. The size of subpopulation acts as a convergence control parameter like tournament size but they are not allowed to be reproduced. By comparison of two individuals based on Pareto optimality, dominating one is selected. If there is no dominance relationship between the two individuals, then NPGA utilizes sub-population to count the number of dominated individuals for each one of the two individuals. The winner is chosen as a child. Nondominated Sorting Genetic Algorithm II (NSGA-II) [Deb et al. 2002] includes two key components: fast nondominated sorting and crowding distance measure. Fast nondominated sorting is to provide rapid method to acquire Pareto ranking q_i in Eq. (2.55) based on the idea of speedup in sorting algorithms. Crowding distance offers orthogonal selection pressure to the pressure toward Pareto front. The difference between sharing and crowding is that sharing utilizes sphere area defined by σ_{share} to measure

density whereas crowding distance only measures distance to the nearest individual in objective space to relieve computational complexity. To increase search capability, crowding distance for individuals at the border is set to infinite to be always reproduced.

Chapter 3

SVM plus ICA

This chapter presents linear and nonlinear hybrid dimensionality framework based on optimization of both structural risk and independence. Two different criteria of structural risk and independence are satisfied with intermediate stage so called projection and uncorrelatedness in between structural risk and independence optimization. Linear/Nonlinear mapping obtained through the hybrid framework will provide improves classification performance compared with other traditional stand alone or hybrid methods such as PCA, LDA, ICA, and PCA plus LDA.

3.1 Dimensionality Reduction based on Support Vector Machine

3.1.1 Support Vector Machine for Dimensionality Reduction

Support Vector Machine (SVM) provides robust nonlinear decision boundary of $\langle \boldsymbol{w}, \phi(\boldsymbol{x}) \rangle + b = 0$ which minimizes structural risk consisting not only of empirical risk but also of complexity of the boundary [Vapnik 1999], where $\boldsymbol{w}$ and b are the projection vector and bias respectively for arbitrary input, $\boldsymbol{x}$. $\boldsymbol{w}$ is utilized as a projection vector for dimensionality reduction to explicitly incorporate decision information into data representation while robustness in SVM is preserved in reduced dimensional space [Tao et al. 2008].

The mappings from structural risk minimization by SVM might be more robust than LDA due to the generalization capability especially when observations in the same class are

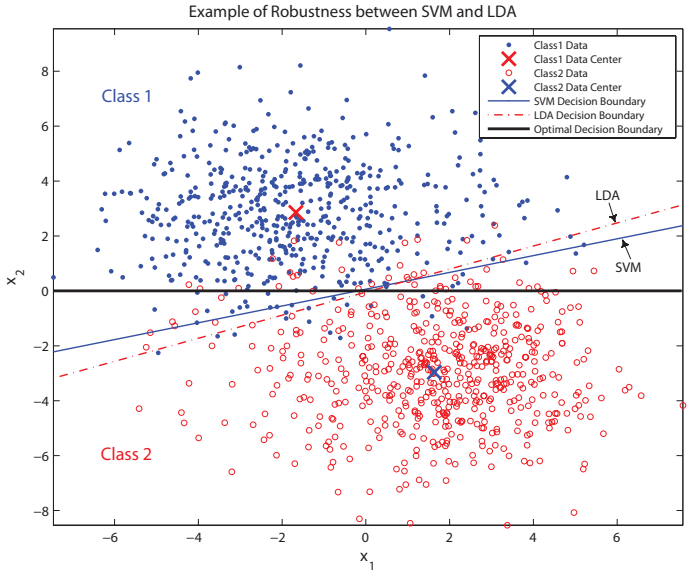


Figure 3.1: Example of robustness between decision boundaries from SVM and LDA

biased or corrupted with noise. Additionally, structural risk based dimensionality reduction shows equal or better classification accuracy than LDA or kDA since LDA can only obtain a decision boundary identical to the one from SVM when there exist sufficiently large number of observations for effective representation of the internal structure of data [Shashua 1999]. In order to demonstrate that SVM presents better robustness than LDA in noisy environments, I put together an example using a two-class synthetic dataset in a two-dimensional space. The data in each class consists of mixture of two Gaussians with biased number of samples corrupted by noise of SNR=5 where SNR is the signal-to-noise ratio. The two Gaussians in class 1 have 500 and 50 samples centered at $[-2\ 3]^{\mathrm{T}}$ and $[2\ 3]^{\mathrm{T}}$ respectively whereas class 2 includes two Gaussians with 50 and 500 samples centered at $[-2\ -3]^{\mathrm{T}}$ and $[2\ -3]^{\mathrm{T}}$, respectively. The covariance for the Gaussians are all identity matrices. The distribution of data for the example is shown in Fig. 3.1, where the filled circle indicates data in class 1 and the empty circle is for data in class 2. Due to the symmetric location between class 1 and class 2 data, the optimal decision should be made at the linear decision boundary, $\boldsymbol{x}_2 = 0$, indicated by the dark solid line in the figure. I also observe that the decision boundary by linear SVM is closer to the optimal than the one by LDA, which shows that SVM is more robust than LDA in noisy and biased environment. This is because SVM is to find support vectors usually located close to the

decision surface whereas LDA utilizes sample mean to form decision criteria. Consequently, the robustness of LDA is determined by the accuracy of the sample mean over the true mean. The decision boundaries by both LDA and SVM should converge to the optimal when there exist sufficient amount of clean data with unbiased data distribution.

The multiclass extension [Hsu and Lin 2002] is applied to SVM to make it applicable to multiclass dataset by providing l-many SVM's, each of which corresponds to $\{\boldsymbol{w}_i, b_i\}$ for the i-th decision boundary, where the number of SVM's represented by l depends on the type of the multiclass extension applied. In this dissertation, I utilize one-against-all (1-a) multiclass extension constructing c-many SVM's based on the dataset consisting of all X_i's, $i = \{1, \cdots, c\}$, where c denotes the number of classes in the dataset and X_i is a set of data belonging to the i-th class. The i-th SVM in 1-a approach builds a decision boundary by $\boldsymbol{w}_i$ and b_i to separate the data in X_i and the others. The multiple projection vectors from SVM's based on multiclass extension are consolidated into the projection matrix, $W_{1,l} = [\boldsymbol{w}_1 \cdots \boldsymbol{w}_l]$ for dimensionality reduction, where $\boldsymbol{w}_i$ is as follows,

$$\boldsymbol{w}_i = \sum_{k=1}^{N} \alpha_k^{(i)} y_k^{(i)} \phi(\boldsymbol{x}_k) \tag{3.1}$$

where $\alpha_k^{(i)}$ denotes the k-th Lagrange multiplier corresponding to $\boldsymbol{x}_k$ for $\boldsymbol{w}_i$. The desired output is set to $y_k^{(i)} = 1$ for $\boldsymbol{x}_k \in X_i$ and $y_k^{(i)} = -1$ otherwise. N is the total number of data satisfying $N = \sum_{i=1}^{c} n(X_i)$. ϕ is nonlinear embedding function to transform data directly from source to hyperdimensional feature space, $\mathcal{F}$. Eq. (3.1) is also represented in the compact form as $\boldsymbol{w}_i = \Phi a^{(i)}$ where $\Phi = [\phi(\boldsymbol{x}_1) \cdots \phi(\boldsymbol{x}_N)]$ and $\boldsymbol{a}^{(i)} = [a_1^{(i)} \cdots a_N^{(i)}]^{\mathrm{T}}$ with $a_k^{(i)} = \alpha_k^{(i)} y_k^{(i)}$. The compact form of the projection matrix is obtained based on the compact representation of Eq. (3.1) as follows,

$$W_{1,l} = \Phi A \tag{3.2}$$

where $A = [a^{(1)} \cdots a^{(l)}]$. $W_{1,l} = [\boldsymbol{w}_1 \cdots \boldsymbol{w}_l]$ where l is set to c due to c-many SVM's in 1-a multiclass extension. The bias, b_i centers the projected data on the decision boundary

corresponding to $\boldsymbol{w}_i$ and is denoted as follows,

$$b_i = \frac{1}{n(S_i)} \sum_{p \in S_i} \left(y_p^{(i)} - \sum_{q \in S_i} \alpha_q^{(i)} y_q^{(i)} \langle \phi(\boldsymbol{x}_q), \phi(\boldsymbol{x}_p) \rangle \right) \tag{3.3}$$

where S_i represents a set of support vectors for $\boldsymbol{w}_i$.

3.1.2 Redundancy Removal by Asymmetric Decorrelation Metric

The redundancy among all $\boldsymbol{w}_i$'s, $i \in \{1, \cdots, l\}$ in $W_{1,l}$ should be removed since c-many $\boldsymbol{w}_i$'s are obtained based on 1-a SVM to minimize structural risk which is irrelevant to the similarity in the projected data onto $W_{1,l}$. Although ICA includes symmetric decorrelation [Hyvarinen 1999] as a redundancy removal process for $\boldsymbol{w}_i$'s in $W_{l+1,m}$ allowing orientation change, it is inappropriate for $\boldsymbol{w}_i$'s in $W_{1,l}$ from SVM since the orientation of SVM's projection vector delivers essential information of decision. Instead of symmetric decorrelation in ICA, this dissertation introduces the concept of *asymmetric decorrelation* to alleviate redundancy among $\boldsymbol{w}_i$'s in $W_{1,l}$ with no orientation alteration.

Asymmetric decorrelation between two projection vectors is conducted based on two metrices, the angular distance between the vectors and the classification performance of training data on the projections. The angular distance is represented as follows,

$$\theta_{ij} = \arccos \frac{\langle \boldsymbol{w}_i, \boldsymbol{w}_j \rangle}{\|\boldsymbol{w}_i\|_2 \|\boldsymbol{w}_j\|_2} \tag{3.4}$$

where θ_{ij} represents the angular distance between $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ which is symmetric, satisfying $\theta_{ij} = \theta_{ji} \in [0, \pi]$. The inner product of $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ in Eq.(3.4) is obtained based on the compact representation of Eq. (3.1) as follows,

$$\begin{aligned} \langle \boldsymbol{w}_i, \boldsymbol{w}_j \rangle &= \langle \Phi \boldsymbol{a}^{(i)}, \Phi \boldsymbol{a}^{(j)} \rangle \\ &= {\boldsymbol{a}^{(i)}}^{\mathrm{T}} \Phi^{\mathrm{T}} \Phi \boldsymbol{a}^{(j)} \\ &= {\boldsymbol{a}^{(i)}}^{\mathrm{T}} K \boldsymbol{a}^{(j)} \end{aligned} \tag{3.5}$$

where K is $N \times N$ Gram matrix composed of k_{ij}'s, each of which is represented by $k_{ij} = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle = f(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $i, j \in \{1, N\}$, an element in the i-th row and the j-th column

of K. $f(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is a kernel function which provides a point mapping for the inner product between $\phi(\boldsymbol{x}_i)$ and $\phi(\boldsymbol{x}_j)$ without direct use of $\phi(\cdot)$ based on Mercer's theorem [Herbrich 2001]. The Euclidean norm in Eq. (3.4) is also obtained by using Eq. (3.5) as $\|\boldsymbol{w}_i\|_2 = \langle \boldsymbol{w}_i, \boldsymbol{w}_i \rangle^{1/2} = (\boldsymbol{a}^{(i)^{\mathrm{T}}} K \boldsymbol{a}^{(i)})^{1/2}$. The classification performance of r_i from $\boldsymbol{w}_i$ over training dataset is as follows,

$$\begin{aligned} r_i &= \frac{1}{2N} \sum_{k=1}^{N} \left| y_k^{(i)} - \mathrm{sign}(\langle \boldsymbol{w}_i, \phi(\boldsymbol{x}_k) \rangle + b_i) \right| \\ &= \frac{1}{2N} \sum_{k=1}^{N} \left| y_k^{(i)} - \mathrm{sign}\Big(\boldsymbol{a}^{(i)^{\mathrm{T}}} u(\boldsymbol{x}_k) + b_i \Big) \right| \end{aligned} \tag{3.6}$$

where sign$(\cdot)$ is signum function and $u(\boldsymbol{x})$ is a projection of $\boldsymbol{x}$ onto $\phi(\boldsymbol{x}_i)$,$\forall i$ in $\mathcal{F}$ as $u(\boldsymbol{x}) = \Phi^{\mathrm{T}} \phi(\boldsymbol{x}) = [f(\boldsymbol{x}_1, \boldsymbol{x}) \cdots f(\boldsymbol{x}_N, \boldsymbol{x})]^{\mathrm{T}}$. $r_i \in [0, 1], \forall i$. I formulate the joint effect of these two metrics of angular distance and classification performance as follows,

$$d_{ij} = \theta_{ij} \frac{r_i}{r_j} \frac{\gamma_{\min}}{\pi} \tag{3.7}$$

where θ_{ij} denotes the angular distance between $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$. r_i and r_j are the classification accuracies using $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$, respectively. Set to 0.5 is $\gamma_{\min}$ which provides lower bound of classification performance for all $\boldsymbol{w}_i$'s. Any $\boldsymbol{w}_i$ with $r_i \leq \gamma_{\min}$ is discarded prior to the redundancy removal process. π is normalization factor such that $d_{ij} \in [0, 1]$. d_{ij} represents how close $\boldsymbol{w}_i$ is to $\boldsymbol{w}_j$. It is asymmetric due to $d_{ij} \neq d_{ji}$. Smaller d_{ij} denotes more redundancy between $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$. I choose to remove $\boldsymbol{w}_i$ instead of $\boldsymbol{w}_j$ because $d_{ij} < d_{ji}$ when $r_i < r_j$, i.e., $\boldsymbol{w}_i$ becomes less meaningful due to lower classification accuracy of r_i than r_j.

Figure 3.2 shows the pseudocode for the redundancy removal process for $\boldsymbol{w}_i$'s from SVM. $\delta \in [0, 1]$ is a threshold for asymmetric decorrelation to guide decision whether to discard $\boldsymbol{w}_i$ so as to control the amount of redundancy among $\boldsymbol{w}_i$'s. The removal process iteratively eliminates $\boldsymbol{w}_{i^*}$'s with minimum asymmetric decorrelation of $\boldsymbol{w}_i$'s in I until minimum decorrelation found is greater than δ or there is nothing to eliminate. When $\delta = 1$, the removal process eliminates all $\boldsymbol{w}_i$'s whereas $\delta = 0$ does not remove any $\boldsymbol{w}_i$'s. I add small positive value to $d_{i^*j^*}$ only when $d_{i^*j^*} = 0$ at the first iteration. The small

Begin

Require: $\delta \in [0, 1]$

Initialize $I = \{1, \cdots, l\}$, $i^* = 0$

Evaluate d_{ij} for $\forall i, j \in I$, $i \neq j$

repeat

$I \leftarrow (I - \{i^*\})$

$(i^*, j^*) = \underset{(i,j) \in I, i \neq j}{\text{argmin}} (d_{ij})$

until $d_{i^*j^*} > \delta$ or $n(I) == 0$

return $\boldsymbol{w}_i$'s where $i \in I$

End

Figure 3.2: Pseudocode for redundancy removal

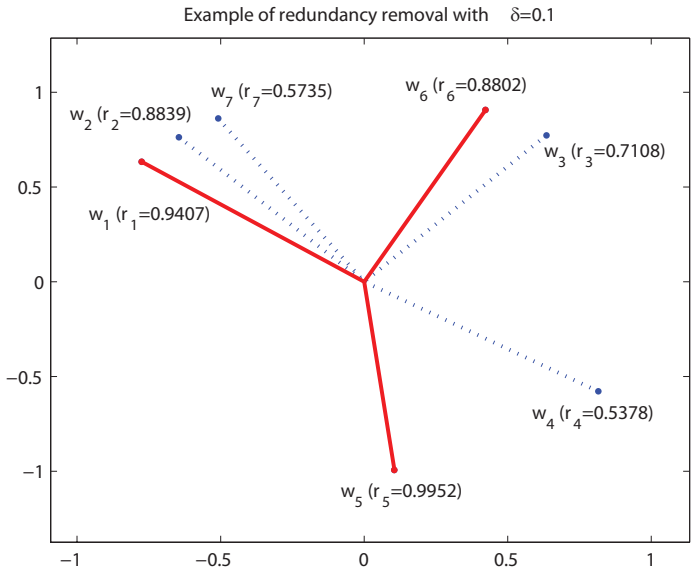


Figure 3.3: Example of redundancy removal

positive value makes $d_{i^*j^*} \in (0, 1]$, compared with $d_{ij} \in [0, 1]$ so that the removal process avoids the case that $\boldsymbol{w}_{i^*}$ at the first iteration is eliminated with $\delta = 0$. Since the removal process eliminates $\boldsymbol{w}_i$'s based on the redundancy evaluation without orientation change, the decision information in $\boldsymbol{w}_i$'s from SVM holds. After the redundancy removal, I considers $W_{1,l} = [\cdots \boldsymbol{w}_i \cdots]$, $\forall i \in I$ where l becomes $n(I) \leq l$.

Figure 3.3 shows an example for the redundancy removal process with the threshold, $\delta = 0.1$. The example includes 7 $\boldsymbol{w}_i$'s, $i \in I = \{1, \cdots, 7\}$ generated from SVM's with linear kernel function, $f(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$. The solid lines denote $\boldsymbol{w}_i$'s survived at the end whereas dotted lines indicate $\boldsymbol{w}_i$'s eliminated during the redundancy removal process. In

the first iteration, the minimum asymmetric decorrelation is found between $\boldsymbol{w}_2$ and $\boldsymbol{w}_7$ with $\min(d_{ij}) = d_{72} = 0.0174$ resulting in the elimination of $\boldsymbol{w}_7$ due to $r_7 < r_2$. In the second iteration, $\boldsymbol{w}_2$ is selected to be removed based on $\min(d_{ij}) = d_{21} = 0.0275$ with $r_2 < r_1$. $\boldsymbol{w}_3$ is chosen to be eliminated with $\min(d_{ij}) = d_{36} = 0.0322$ and $r_3 < r_6$ in the third iteration. $\boldsymbol{w}_4$ is the last one to be discarded with $\min(d_{ij}) = d_{45} = 0.0729$ and $r_4 < r_5$ in the fourth iteration. The removal process terminates at the fifth iteration since $\min(d_{ij}) = d_{61} = 0.1968 > \delta$ leaving only $\boldsymbol{w}_1$, $\boldsymbol{w}_5$, and $\boldsymbol{w}_6$ which are sufficiently far away from each other with relatively higher classification accuracies.

3.2 Linear SVM plus ICA

This section presents an effective linear hybrid dimensionality reduction method based on Support Vector Machine (SVM) and Independent Component Analysis (ICA), referred to as SVM plus ICA (SVM+ICA), to maintain high classification accuracy in lower dimensional space that is less sensitive to noise. Since SVM+ICA is not based on LDA, it does not suffer from the S3 or common mean problems inherited from the LDA criteria.

SVM minimizes structural risk so as to offer projection with better generalization capability to improve classification/estimation performance for unknown samples. Since maximum margin among features provides better data representation to improve classification performance [Gilad-Bachrach et al. 2004] and SVM projection itself is capable of building an effective subspace for dimensionality reduction [Tao et al. 2008; Tsang et al. 2008], I adopt SVM as a supervised component in the proposed hybrid algorithm.

On the other hand, ICA offers projection which maximizes independence among features with better data representation [Hyvarinen 1999] and has been shown [Yang et al. 2005, 2007] to play an important role in classification performance improvement, I incorporate ICA as the unsupervised component in the proposed hybrid algorithm.

In order to combine projections derived from SVM and ICA into a unified framework for effective dimensionality reduction, the orthogonal relationship is sought between mapping vectors from SVM and ICA, such that contribution made by the supervised and unsupervised processes have minimum correlation, leading to much reduced dimensionality. This idea is similar to the Orthogonal Centroid Method (OCM) [Foley and Sammon Jr. 1975;

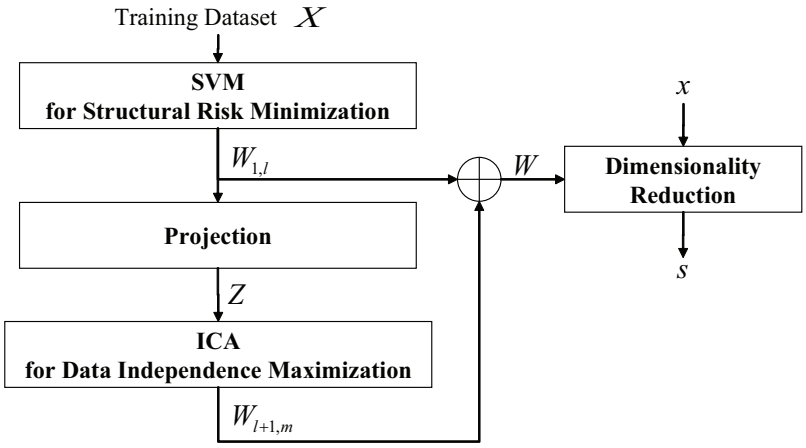


Figure 3.4: The linear SVM plus ICA

Ye 2005] but I replace OCM's maximum margin criterion with SVM's structural risk minimization which does not suffer from the S3 problem. Under the orthogonal relationship between SVM and ICA, ICA over the subspace orthogonal to SVM projection vectors allows us to merge two projections from both SVM and ICA into one concatenated projection matrix. Therefore, SVM+ICA improves classification performance with robustness resulting from minimum structural risk with independence.

3.2.1 The Concept of Linear SVM plus ICA

This section describes the new hybrid dimensionality deduction method that consists of the simultaneous minimization of structural risk (the supervised criterion) and maximization of data independence (the unsupervised criterion), as each criterion has shown better performance individually compared to the corresponding traditional criterion, such as LDA or PCA. I refer to this method as SVM+ICA. Figure 3.4 provides a block diagram of the proposed linear SVM+ICA method. It consists of three components, structural risk minimization, projection, and independence maximization. In Fig. 3.4, $X = \{\boldsymbol{x}_i \in R^n, \forall i\}$ represents a training data set of dimension n, which is to be reduced to another set, S, of dimension, m, where $m \ll n$, using the projection matrix, W, of m mapping column vectors constructed from the SVM+ICA process. The structural risk minimization component generates the first l mapping vectors of W, denoted as $W_{1,l}$ and the data independence maximization component yields the other $m-l$ vectors of W, denoted as $W_{l+1,m}$. This con-

catenation process is denoted using the symbol $\bigoplus$ in Fig. 3.4. Z is the projected data set from X based on $W_{1,l}$, which is to be fed to the data independent maximization component to derive $W_{l+1,m}$. I will elaborate on the rationale behind the proposed linear SVM+ICA in the following three subsections.

3.2.2 Orthogonality

Intuitively, the most effective set of mapping vectors derived from the structural risk minimization process ($W_{1,l}$) and the independence maximization process ($W_{l+1,m}$) should be the ones without any redundant information for the reduced space construction spanned by $W_{1,l}$ and $W_{l+1,m}$. The least amount of redundancy results from the pair-wise orthogonality between $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ where $i \in \{1, \cdots, l\}$ and $j \in \{l+1, \cdots, m\}$. The pair-wise orthogonality is also represented by $W_{1,l} \perp W_{l+1,m}$ or equivalently $W_{l+1,m}^{\mathrm{T}} W_{1,l} = \mathbf{0}$.

The projection component, as an intermediate step in the linear SVM+ICA, allows for mapping vectors derived from structural risk minimization and independence maximization to achieve minimum correlation. It does so by projecting the given data X onto the subspace satisfying $W_{1,l}^{\mathrm{T}} \boldsymbol{x} = \mathbf{0}$, yielding the projected data, Z, such that the subsequent independence maximization process based on Z is least affected or correlated with the previous structural risk minimization process. After the projection procedure, the projected data, Z, would lose information along the direction of $W_{1,l}$, which indicates that decision information through $W_{1,l}$ is no longer valid in the projection subspace. Therefore, the projection guarantees that any mapping vectors from structural risk minimization, $W_{1,l}$, and independence maximization, $W_{l+1,m}$, are uncorrelated since $W_{l+1,m} \perp W_{1,l}$.

The projection onto the subspace, orthogonal to the decision hyperplane from structural risk minimization, $W_{1,l}$, is formulated as a constrained optimization problem as follows,

$$\begin{aligned} \boldsymbol{z}^* &= \underset{\boldsymbol{z}}{\operatorname{argmin}} \|\boldsymbol{x} - \boldsymbol{z}\|^2 \\ &\text{subject to } W_{1,l}^{\mathrm{T}} \boldsymbol{z} = \mathbf{0} \end{aligned} \tag{3.8}$$

where $\boldsymbol{z}$ represents the projected data onto the subspace orthogonal to $W_{1,l}$ and parallel to the decision hyperplane(s). Due to the orthogonality between $W_{1,l}$ and any components in the decision hyperplane, the structural risk minimization and independence maximization

are isolated and performed one by one holding independence between any pair of $\boldsymbol{w}_i$'s and $\boldsymbol{w}_j$'s where $i \in \{1, \cdots, l\}$ and $j \in \{l+1, \cdots, m\}$. In order to solve the constrained optimization problem, I apply Lagrange optimization by introducing the Lagrangian multipliers, $\boldsymbol{\lambda} \in R^l$ as follows,

$$L(\boldsymbol{z}, \boldsymbol{\lambda}) = \|\boldsymbol{x} - \boldsymbol{z}\|^2 + \boldsymbol{\lambda}^{\mathrm{T}}(W_{1,l}^{\mathrm{T}}\boldsymbol{z}) \tag{3.9}$$

Taking the partial derivative of L with respect to $\boldsymbol{z}$ and $\boldsymbol{\lambda}$, I have

$$\frac{\partial L(\boldsymbol{z}, \boldsymbol{\lambda})}{\partial \boldsymbol{z}} = -2(\boldsymbol{x} - \boldsymbol{z}^*) + W_{1,l}\boldsymbol{\lambda} = \mathbf{0} \tag{3.10}$$

$$\frac{\partial L}{\partial \boldsymbol{\lambda}} = W_{1,l}^{\mathrm{T}}\boldsymbol{z}^* = \mathbf{0} \tag{3.11}$$

By summarizing Eqs. (3.10) and (3.11), I have

$$\begin{bmatrix} 2I_n & W_{1,l} \\ W_{1,l}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{z}^* \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} 2\boldsymbol{x} \\ \mathbf{0} \end{bmatrix} \tag{3.12}$$

where I_n is the identity matrix of n dimension. The $\boldsymbol{z}^*$'s form the projected dataset Z which will be used by the subsequent independent maximization process in the orthogonal subspace to $\boldsymbol{w}_i$'s from SVM's.

3.2.3 Linear Projection from ICA over Orthogonal Subspace

As the unsupervised dimensionality reduction component in the proposed SVM+ICA framework, independence maximization is applied over the projected data, Z. Independence maximization searches for a linear non-orthogonal coordinate system whose axes are determined by both the second and higher order statistics of the original data. Since independence maximization is known as a method providing better data representation than other conventional techniques such as PCA, higher classification accuracy is expected, leading to the adoption of independence maximization in the proposed hybrid dimensionality reduction framework. To find mappings which maximize independence, I adopt the approximated negative entropy criterion introduced in [Hyvarinen and Oja 2000], also referred to as FastICA, due to well-justified statistical theory and computational efficiency. The Fas-

tICA algorithm involves two sequential processes, the *one unit (weight vector) estimation* and the *decorrelation* among weight vectors. The one unit process estimates the weight vectors as follows,

$$\boldsymbol{w}_i^+ = E\left\{\boldsymbol{z}g(\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{z})\right\} - E\left\{g'(\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{z})\right\}\boldsymbol{w}_i \tag{3.13}$$

where $\boldsymbol{w}_i^+$ is the temporal approximation of the independent component with $i \in \{l+1, \cdots, m\}$. g is the derivative of the non-quadratic function introduced in [Hyvarinen and Oja 2000], and $g(u) = \tanh(au)$. g' is the derivative of g, and $g'(u) = sech^2(u)$.

The purpose of the decorrelation process is to keep different weight vectors from converging to the same maximum. The deflation scheme based on symmetric decorrelation [Karhunen et al. 1997] helps remove dependency among $\boldsymbol{w}_i^+$'s as follows,

$$W_{l+1,m} = W_{l+1,m}^+ \left[{W_{l+1,m}^+}^{\mathrm{T}} W_{l+1,m}^+ \right]^{-\frac{1}{2}} \tag{3.14}$$

where $W_{l+1,m}$ represents decorrelated mappings based on $W_{l+1,m}^+ = [\boldsymbol{w}_{l+1}^+ \cdots \boldsymbol{w}_m^+]$ from independence maximization.

3.2.4 Conducting Dimensionality Reduction

The dimensionality reduction by the linear SVM+ICA is performed by linear projection as follows,

$$\boldsymbol{s} = W^{\mathrm{T}}\boldsymbol{x} \tag{3.15}$$

where $\boldsymbol{x}$ is an arbitrary input and $\boldsymbol{s} \in R^m$ denotes the input represented in reduced dimension space. $W = [W_{1,l}\ W_{l+1,m}]$ and m is the number of dimensions to be reduced to. When $m \leq l$, the dimensionality reduction is driven only by $\boldsymbol{w}_i$'s from SVM without ICA. When $m > l$, $(m-l)$-many mapping vectors from ICA will be added to the mapping matrix, W.

3.3 Nonlinear SVM plus ICA

The linear SVM plus ICA in Sec. 3.2 is extended to nonlinear SVM plus ICA, nonlinear hybrid dimensionality reduction approach to provide improved classification performance and robustness based on the integration of the supervised criterion from SVM and the unsupervised criterion from ICA through the uncorrelated subspace construction. The proposed approach consists of three components, nonlinear projection through SVM where the directions of the decision surfaces are used as a part of the projection vectors in dimensionality reduction, uncorrelated subspace construction such that projection vectors from SVM are pair-wise uncorrelated with those from ICA, and nonlinear projection through ICA over the uncorrelated subspace. I am not the first to use projection vectors for dimensionality reduction purpose. Previous works, e.g., Decision Boundary Feature Extraction (DBFE) [Lee and Landgrebe 1993] and RSVM, have showed that decision information can be explicitly utilized as projection vectors for dimensionality reduction. The projection vectors built by the set of support vectors make SVM less computationally expensive than DBFE. The weakness of RSVM regarding ineffective projection vectors due to the multilevel decomposition does not reside in the proposed approach since the redundancy in multiple SVM's from one-against-all multiclass extension [Hsu and Lin 2002] is removed by the so-called redundancy removal process using the asymmetric decorrelation metric introduced in Sec. 3.1.2. All the processes in the proposed nonlinear SVM plus ICA are completed in hyperdimensional space for nonlinear data representation through kernel function based on Mercer's theorem [Herbrich 2001]. Therefore, the nonlinear SVM plus ICA improves classification performance with robustness resulting from minimum structural risk and maximum data independence with nonlinear data representation capability.

3.3.1 The Fundamentals of Nonlinear SVM plus ICA

Nonlinear SVM plus ICA is a dimensionality reduction algorithm consisting of both supervised SVM and unsupervised ICA based on the noiseless nonlinear dimensionality reduction model, $s = \langle W, \phi(\boldsymbol{x}) \rangle$ where x and s are the observation input and the corresponding output, respectively, with nonlinear function, ϕ which projects data into the hyperdimensional space, $\mathcal{F}$.

SVM plays an important role in dimensionality reduction to deliver projection with maximum separability for arbitrary input [Tao et al. 2008] since structural risk minimization in SVM provides the best trade-off between minimum empirical error and complexity of the projection over the given dataset, $(\boldsymbol{x}_i, y_i)$ for $i \in \{1, \cdots, N\}$ where N is the number of data samples. Since SVM requires supervised directive, y_i to measure both empirical risk and complexity [Vapnik 1999], SVM is categorized as supervised approach. ICA searches for the projection which maximizes component-wise independence by imposing the criteria where the probability density function of output factorizes in reduced dimensional space. The better data representation capability inherited from the independent relationship makes ICA another important approach in dimensionality reduction. Since independent components is constructed by the understanding of the given data, $\boldsymbol{x}_i$, without corresponding y_i , ICA belongs to the unsupervised approach.

Hybrid dimensionality reduction consists of both supervised and unsupervised criteria to provide better data representation for classification performance improvement compared with either the supervised or unsupervised method. Based on how the supervised and unsupervised criteria are integrated, hybrid methods can be categorized as subspace-based and unified criterion-based.

The subspace-based method utilizes subspace in between the supervised and unsupervised criteria. Due to the intermediate subspace, subspace based methods simply couple two distinctive criteria into one although it requires two-stage optimization, one for the supervised component and the other for the unsupervised component. Since the reduced dimensional space from the subspace-based method is partially regulated by the subspace, the construction of the subspace becomes critical. Several hybrid dimensionality reduction methods fall into this category, including LDA over PCA [Belhumeur et al. 1997; Yang and Yang 2001, 2003], APCDA [Jiang 2009], and ICA augmented by LDA [Kwak and Pedrycz 2007]. LDA over PCA combines LDA with PCA to resolve the S3 problem by removing singularity through PCA so that LDA is performed in the PCA subspace. APCDA consists of Asymmetric Discriminant Analysis (ADA) and and Asymmetric PCA (APCA) where ADA extends LDA with Common Mean Feature Extraction (CMFE) and APCA regulates PCA with supervised directive for unbalanced number of data per class. APCA provides subspace for ADA in a similar way as LDA over PCA. ICA augmented by LDA builds

subspace by ICA instead of PCA for LDA to provide better discriminant capability.

In contrast to the subspace-based method, in unified criterion-based hybrid methods, the two distinctive supervised and unsupervised criteria are integrated into a single objective function through constrained optimization where the two distinctive criteria are optimized simultaneously with no additional computational cost for subspace construction. However, complicate formulations of supervised and unsupervised criteria make it very difficult for a seamless integration of the two criteria. To simplify the process, some portions of the original criteria are ignored resulting in somewhat performance degradation. The unified criterion-based methods include the supervised MI-based ICA [Leiva-Murillo and Artes-Rodriguez 2007], DNMF [Zafeiriou et al. 2006], and Non-negative Tensor Factorization (NTF) with LDA [Zafeiriou 2009]. The supervised MI-based ICA only incorporates supervised class directive into mutual information maximization in ICA. Since the supervised MI-based ICA does not strictly incorporate between-class separability, the integration does not contribute directly to the classification performance improvement in reduced dimensional space. In DNMF and NTF with LDA, LDA's within- and between-class variance are linearly added with control parameters to the factorization objective functions. Although DNMF and NTF with LDA successfully integrate the two objectives into single formulation, there still exists common mean problem inherited from the direct incorporation of LDA's between-class variance and the unified criterion does not provide maximum separability shown in LDA due to the linear integration of LDA's within-class variance.

To summarize, when designing hybrid dimensionality reduction methods, there are two key factors need to be taken into consideration. First of all, it is essential to choose appropriate supervised and unsupervised dimensionality reduction methods. Conventional hybrid methods mostly depend on supervised LDA so that the problems inherited from LDA reside in the hybrid design regardless of the way of supervised and unsupervised criteria integration. Secondly, for arbitrary complicated objective functions with constraints, subspace-based methods are easier to couple the objectives into single framework compared with the method using unified criterion, in which case the construction of an appropriate subspace becomes essential.

I propose a new dimensionality reduction algorithm, nonlinear SVM plus ICA as a

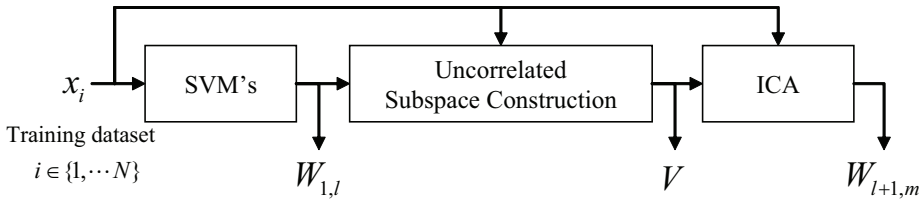


Figure 3.5: The nonlinear SVM plus ICA

subspace-based method to integrate SVM as a supervised and ICA as an unsupervised criterion over the subspace with uncorrelatedness constraint. I refer to this subspace as the "*uncorrelated subspace*. In this method, SVM delivers generalization capability for better classification performance for arbitrary input and the intrinsic information extracted by ICA provides better data representation capability. The uncorrelated subspace provides minimum relation between SVM and ICA where the empirical correlation formulation is adopted to measure this relationship. The uncorrelated subspace is especially effective for the integration of SVM and ICA in nonlinear dimensionality reduction model via kernel. The kernel method transforms data into hyperdimensional feature space, $\mathcal{F}$, so that the number of data becomes much less than the data dimensionality. Since ICA requires whitened input [Hyvarinen and Oja 2000] for better performance and fast computation, the subspace on which ICA is performed should be reduced to the most extent by eliminating the null space found in the centered covariance of the training data in $\mathcal{F}$ while SVM is minimally correlated with ICA over the subspace. The eigen-decomposition is an effective tool to remove the null space and the minimal correlation between SVM and ICA can be incorporated into the eigen-problem as a constraint. I will introduce the in-depth design of nonlinear SVM plus ICA in the later sub-chapters. Since the projection from the proposed algorithm is from nonlinear dimensionality model, $s = \langle W, \phi(\boldsymbol{x}) \rangle$, nonlinearity in data is better represented by the proposed compared with the methods based on the linear model, $s = \langle W, \boldsymbol{x} \rangle$. I expect that nonlinear SVM plus ICA finds nonlinear projection which provides better data representation capability resulting in the classification performance improvement with robustness under noisy environment.

Figure 3.5 shows the nonlinear SVM plus ICA to obtain nonlinear projection matrix for dimensionality reduction from input to m-dimensional reduced space based on N-many training dataset, $\boldsymbol{x}_i$, $i \in \{1, \cdots, N\}$. $W_{1,l}$ denotes projection matrix composed of l-many

projection vectors from SVM. The uncorrelated subspace is spanned by $\boldsymbol{v}_i$'s where $\boldsymbol{v}_i$ is the i-th column vector of V. ICA finds $W_{l+1,m}$ containing $(m-l)$-many projection vectors in the uncorrelated subspace. The overall projection matrix, W is built by $W_{1,l}$, V, and $W_{l+1,m}$ for which the proposed algorithm requires three parts of SVM, uncorrelated subspace construction, and ICA.

SVM in the proposed algorithm explicitly contributes to classification performance improvement based on structural risk minimization. The column vectors in $W_{1,l}$ correspond to l-many projection vectors orthogonal to the decision surfaces obtained by l-many SVM's where l is determined by multiclass extension strategy for multiclass dataset. The redundancy removal is a post-processing to eliminate redundancy among the l-many projection vectors based on asymmetric decorrelation metric. I introduced the detail of SVM with the redundancy removal process for dimensionality reduction in Sec. 3.1.

The uncorrelated subspace construction provides maximally uncorrelated subspace with $W_{1,l}$ spanned by $\boldsymbol{v}_i$ in V,$\forall i$ for ICA to find $W_{l+1,m}$ in the subspace so as to contribute the projections from both SVM and ICA in the tradeoff between class separability and independent data representation. The detail of the uncorrelated subspace construction is introduced in Sec. 3.3.2.

ICA is utilized as an unsupervised method in the proposed algorithm to extract intrinsic information from data in the subspace uncorrelated with $W_{1,l}$. The intrinsic information in data is obtained by nonlinear projection using $W_{l+1,m}$ which consists of $(m-l)$-many column vectors each of which denotes a projection corresponding to one of the independent components for dimensionality reduction. Sec. 3.3.3 will provide detail description for nonlinear ICA as a dimensionality reduction process in the proposed algorithm.

3.3.2 Uncorrelated Subspace Construction

The construction of the uncorrelated subspace with the projection from SVM's, $W_{1,l}$, nullifies the component-wise separability offered by $\boldsymbol{w}_i$'s from SVM's so that ICA performed on the subspace delivers maximally independent component without interference from separability.

Formulation of the Optimization Problem

The t-th component in V is built by the maximum correlation search in $\mathcal{F}$ with three constraints of uncorrelated relationship with $W_{1,l}$, pair-wise orthogonality, and unit length as follows,

$$\begin{aligned} \boldsymbol{v}_t^* = & \underset{\boldsymbol{v}_t}{\operatorname{argmax}} \ \mathrm{E}\left[\|X_{\tilde{\phi}}^{\mathrm{T}}\boldsymbol{v}_t\|^2\right] \\ \text{s.t.} \quad & \mathrm{E}\big[(X_{\tilde{\phi}}^{\mathrm{T}}\tilde{W}_{1,l})^{\mathrm{T}}(X_{\tilde{\phi}}^{\mathrm{T}}\boldsymbol{v}_t)\big] = 0 \\ & V_{1,t-1}^{\mathrm{T}}\boldsymbol{v}_t = 0 \\ & \|\boldsymbol{v}_t\|^2 = 1 \end{aligned} \tag{3.16}$$

where $\boldsymbol{v}_t$ is the t-th component spanning the uncorrelated subspace. For the correlation based objective function and the constraint in Eq. (3.16), the data must be centered in search space. $\tilde{\phi}$ is based on ϕ but centered in the embedding space to remove the degree of freedom that ϕ be translated by a constant amount,

$$\tilde{\phi}(\boldsymbol{x}) = \phi(\boldsymbol{x}) - \boldsymbol{\mu} \tag{3.17}$$

where $\boldsymbol{\mu} = \frac{1}{N}\sum_{k=1}^{N}\phi(\boldsymbol{x}_k)$. The compact representation of Eq. (3.17) for all $\boldsymbol{x}_i$'s are as follows,

$$\begin{aligned} \tilde{\Phi} = & [\phi(\boldsymbol{x}_1)\cdots\phi(\boldsymbol{x}_N)]\left(I - \frac{1}{N}1_{N\times N}\right) \\ = & \Phi H \end{aligned} \tag{3.18}$$

where $H = \left(I - \frac{1}{N}1_{N\times N}\right)$ satisfying $H^{\mathrm{T}} = H$ and $\Phi = [\phi(\boldsymbol{x}_1)\cdots\phi(\boldsymbol{x}_N)]$.

$1_{N\times N}$ denotes an $N \times N$ matrix with all elements being 1s. $X_{\tilde{\phi}}$ denotes a random vector satisfying $\mathrm{E}[X_{\tilde{\phi}}] = 0$ in $\mathcal{F}$ through centered nonlinear embedding.

$V_{1,t-1}$ and $\boldsymbol{v}_t$ are defined as follows,

$$\begin{aligned}\boldsymbol{v}_t=&\sum_{k=1}^{N}\beta_k^{(t)}\tilde{\phi}(\boldsymbol{x}_k)\\=&\tilde{\Phi}\boldsymbol{\beta}^{(t)}\end{aligned} \tag{3.19}$$

$$\begin{aligned}V_{1,t-1}=&[\boldsymbol{v}_1\cdots\boldsymbol{v}_{t-1}]\\=&\tilde{\Phi}\Upsilon_{1,t-1}\end{aligned} \tag{3.20}$$

where $\boldsymbol{v}_t$ is represented as the linear combination of $\tilde{\phi}(\boldsymbol{x}_k)$ with the corresponding weight, $\beta_k^{(t)}$ since the maximally correlated $\boldsymbol{v}_t$ for the data projected onto $\mathcal{F}$ is found by the data covariance analysis of $\left(\sum_{k=1}^{N}\tilde{\phi}(\boldsymbol{x}_k)\tilde{\phi}(\boldsymbol{x}_k)^{\mathrm{T}}\right)\boldsymbol{v}_t=\sum_{k=1}^{N}\left(\tilde{\phi}(\boldsymbol{x}_k)^{\mathrm{T}}\boldsymbol{v}_t\right)\tilde{\phi}(\boldsymbol{x}_k)$ showing that $\boldsymbol{v}_t$ lies in the span of $\tilde{\phi}(\boldsymbol{x}_k)$'s. $\boldsymbol{\beta}^{(t)}=[\beta_1^{(t)}\cdots\beta_N^{(t)}]^{\mathrm{T}}\in R^{N\times 1}$. $\Upsilon_{1,t-1}=[\boldsymbol{\beta}^{(1)}\cdots\boldsymbol{\beta}^{(t-1)}]$.

The projection matrix from SVM, according to Eq. (3.2), is $W_{1,l}=\Phi A$. Therefore, $\tilde{W}_{1,l}$, through centered nonlinear embedding, is $\tilde{W}_{1,l}=\tilde{\Phi}\tilde{A}$. Since $W_{1,l}$ are made up of normalized projection vectors whose orientations do not change with the data center, $\tilde{W}_{1,l}=W_{1,l}$. I can now solve $\tilde{A}$ with respect to A based on $\tilde{\Phi}^{\mathrm{T}}\tilde{W}_{1,l}=\tilde{\Phi}^{\mathrm{T}}W_{1,l}$ with $\tilde{\Phi}^{\mathrm{T}}\tilde{W}_{1,l}=\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\tilde{A}=\tilde{K}\tilde{A}$ and $\tilde{\Phi}^{\mathrm{T}}W_{1,l}=\tilde{\Phi}^{\mathrm{T}}\Phi A=H\Phi^{\mathrm{T}}\Phi A=HKA$ as follows,

$$\tilde{A}=\tilde{K}^{-1}HKA \tag{3.21}$$

where $\tilde{K}$ is centered Gram matrix introduced in [Bach and Jordan 2002; Scholkopf et al. 1998]. Based on Eq. (3.18), $\tilde{K}$ can be further written as

$$\begin{aligned}\tilde{K}=&\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\\=&H^{\mathrm{T}}\Phi^{\mathrm{T}}\Phi H\\=&HKH\end{aligned} \tag{3.22}$$

Note that $\tilde{K}^{\mathrm{T}}=\tilde{K}$ since both K and H are symmetric.

Solving the Optimization Problem

I use Lagrangian formulation to obtain $\boldsymbol{v}_t$ from the constrained maximization problem in Eq. (3.16). The objective function is rewritten as

$$\begin{aligned} L=&\boldsymbol{v}_t^{\mathrm{T}}\mathrm{E}\left[X_{\tilde{\phi}}X_{\tilde{\phi}}^{\mathrm{T}}\right]\boldsymbol{v}_t+\sum_{i=1}^{l}\lambda_i^{(1)}\mathrm{E}\left[(X_{\tilde{\phi}}^{\mathrm{T}}\tilde{\boldsymbol{w}}_i)^{\mathrm{T}}(X_{\tilde{\phi}}^{\mathrm{T}}\boldsymbol{v}_t)\right]+\sum_{i=1}^{t-1}\lambda_i^{(2)}\boldsymbol{v}_i^{\mathrm{T}}\boldsymbol{v}_t+\lambda^{(3)}\left(\boldsymbol{v}_t^{\mathrm{T}}\boldsymbol{v}_t-1\right) \\ =&\frac{1}{N}\boldsymbol{v}_t^{\mathrm{T}}\tilde{\Phi}\tilde{\Phi}^{\mathrm{T}}\boldsymbol{v}_t+\frac{1}{N}\left\{\tilde{\Phi}\tilde{\Phi}^{\mathrm{T}}W_{1,l}\boldsymbol{\lambda}^{(1)}\right\}^{\mathrm{T}}\boldsymbol{v}_t+\left(V_{1,t-1}\boldsymbol{\lambda}^{(2)}\right)^{\mathrm{T}}\boldsymbol{v}_t+\lambda^{(3)}\left(\boldsymbol{v}_t^{\mathrm{T}}\boldsymbol{v}_t-1\right) \end{aligned} \tag{3.23}$$

where, in total, $(l+t)$-many Lagrange multipliers are used including $\boldsymbol{\lambda}^{(1)}=[\lambda_1^{(1)}\cdots\lambda_l^{(1)}]^{\mathrm{T}}\in R^{l\times 1}$, $\boldsymbol{\lambda}^{(2)}=[\lambda_1^{(2)}\cdots\lambda_{t-1}^{(2)}]^{\mathrm{T}}\in R^{(t-1)\times 1}$ ($\boldsymbol{\lambda}^{(2)}$ is activated only when there exist pre-obtained $\boldsymbol{v}_i$'s), and $\lambda^{(3)}\in R^1$.

To solve the problem, we take partial derivatives of the Lagrangian formulation in Eq. (3.23) with respect to four different sets of parameters, $\boldsymbol{\beta}^{(t)}$, $\boldsymbol{\lambda}^{(1)}$, $\boldsymbol{\lambda}^{(2)}$, and $\lambda^{(3)}$ as follows,

$$\begin{aligned} \frac{\partial L}{\partial\boldsymbol{\beta}^{(t)}}=&\frac{\partial\boldsymbol{v}_t}{\partial\boldsymbol{\beta}^{(t)}}\frac{\partial L}{\partial\boldsymbol{v}_t} \\ =&\tilde{\Phi}^{\mathrm{T}}\left(\frac{2}{N}\tilde{\Phi}\tilde{\Phi}^{\mathrm{T}}\boldsymbol{v}_t+\frac{1}{N}\tilde{\Phi}\tilde{\Phi}^{\mathrm{T}}W_{1,l}\boldsymbol{\lambda}^{(1)}+V_{1,t-1}\boldsymbol{\lambda}^{(2)}+2\lambda^{(3)}\boldsymbol{v}_t\right) \\ =&\frac{2}{N}\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\boldsymbol{\beta}^{(t)}+\frac{1}{N}\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}(H\Phi^{\mathrm{T}}\Phi A)\boldsymbol{\lambda}^{(1)}+\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\Upsilon_{1,t-1}\boldsymbol{\lambda}^{(2)}+2\lambda^{(3)}\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\boldsymbol{\beta}^{(t)} \\ =&\frac{2}{N}\tilde{K}^2\boldsymbol{\beta}^{(t)}+\frac{1}{N}\tilde{K}G^{\mathrm{T}}\boldsymbol{\lambda}^{(1)}+\tilde{K}\Upsilon_{1,t-1}\boldsymbol{\lambda}^{(2)}+2\lambda^{(3)}\tilde{K}\boldsymbol{\beta}^{(t)} \end{aligned} \tag{3.24}$$

where $G=A^{\mathrm{T}}KH$. $\partial L/\partial\boldsymbol{\beta}^{(t)}\in R^{N\times 1}$.

$$\begin{aligned} \frac{\partial L}{\partial\boldsymbol{\lambda}^{(1)}}=&\mathrm{E}\left[(X_{\tilde{\phi}}^{\mathrm{T}}W_{1,l})^{\mathrm{T}}(X_{\tilde{\phi}}^{\mathrm{T}}\boldsymbol{v}_t)\right] \\ =&\frac{1}{N}\left(\tilde{\Phi}^{\mathrm{T}}W_{1,l}\right)^{\mathrm{T}}\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\boldsymbol{\beta}^{(t)} \\ =&\frac{1}{N}\left(A^{\mathrm{T}}\Phi^{\mathrm{T}}\Phi H\right)\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\boldsymbol{\beta}^{(t)} \\ =&\frac{1}{N}(A^{\mathrm{T}}KH)\tilde{K}\boldsymbol{\beta}^{(t)} \\ =&\frac{1}{N}G\tilde{K}\boldsymbol{\beta}^{(t)} \end{aligned} \tag{3.25}$$

$$\begin{aligned}\frac{\partial L}{\partial \boldsymbol{\lambda}^{(2)}}=&V_{1,t-1}^{\mathrm{T}}\boldsymbol{v}_t\\ =&\left(\tilde{\Phi}\Upsilon_{1,t-1}\right)^{\mathrm{T}}\tilde{\Phi}\boldsymbol{\beta}^{(t)}\\ =&\Upsilon_{1,t-1}^{\mathrm{T}}\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\boldsymbol{\beta}^{(t)}\\ =&\Upsilon_{1,t-1}^{\mathrm{T}}\tilde{K}\boldsymbol{\beta}^{(t)}\end{aligned} \tag{3.26}$$

$$\begin{aligned}\frac{\partial L}{\partial \lambda^{(3)}}=&\boldsymbol{v}_t^{\mathrm{T}}\boldsymbol{v}_t-1\\ =&\left(\tilde{\Phi}\boldsymbol{\beta}^{(t)}\right)^{\mathrm{T}}\tilde{\Phi}\boldsymbol{\beta}^{(t)}-1\\ =&{\boldsymbol{\beta}^{(t)}}^{\mathrm{T}}\tilde{K}\boldsymbol{\beta}^{(t)}-1\end{aligned} \tag{3.27}$$

where $\partial L/\partial \boldsymbol{\lambda}^{(1)} \in R^{l\times 1}$, $\partial L/\partial \boldsymbol{\lambda}^{(2)} \in R^{(t-1)\times 1}$, and $\partial L/\partial \lambda^{(3)} \in R^{1}$.

By setting Eqs. (3.24)- (3.27) to zeros, we first simplify $\boldsymbol{\lambda}^{(1)}$ by multiplying G to $\partial L/\partial \boldsymbol{\beta}^{(t)}$ as follows,

$$\begin{aligned}G\frac{\partial L}{\partial \boldsymbol{\beta}^{(t)}}=&G\left\{\frac{2}{N}\tilde{K}^2\boldsymbol{\beta}^{(t)}+\frac{1}{N}\tilde{K}G^{\mathrm{T}}\boldsymbol{\lambda}^{(1)}+\tilde{K}\Upsilon_{1,t-1}\boldsymbol{\lambda}^{(2)}+2\lambda^{(3)}\tilde{K}\boldsymbol{\beta}^{(t)}\right\}\\ =&\frac{2}{N}G\tilde{K}^2\boldsymbol{\beta}^{(t)}+\frac{1}{N}G\tilde{K}G^{\mathrm{T}}\boldsymbol{\lambda}^{(1)}+(A^{\mathrm{T}}\Phi^{\mathrm{T}}\Phi H)\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\Upsilon_{1,t-1}\boldsymbol{\lambda}^{(2)}+2\lambda^{(3)}G\tilde{K}\boldsymbol{\beta}^{(t)}\\ =&\frac{2}{N}G\tilde{K}^2\boldsymbol{\beta}^{(t)}+\frac{1}{N}G\tilde{K}G^{\mathrm{T}}\boldsymbol{\lambda}^{(1)}+(W_{1,l}^{\mathrm{T}}\tilde{\Phi})(\tilde{\Phi}^{\mathrm{T}}V_{1,t-1})\boldsymbol{\lambda}^{(2)}+2\lambda^{(3)}N\frac{\partial L}{\partial \boldsymbol{\lambda}^{(1)}}\\ =&\frac{2}{N}G\tilde{K}^2\boldsymbol{\beta}^{(t)}+\frac{1}{N}G\tilde{K}G^{\mathrm{T}}\boldsymbol{\lambda}^{(1)}\end{aligned} \tag{3.28}$$

where $(W_{1,l}^{\mathrm{T}}\tilde{\Phi})(\tilde{\Phi}^{\mathrm{T}}V_{1,t-1})=0$ due to Eq. (3.25)such that $\frac{\partial L}{\partial \boldsymbol{\lambda}^{(1)}}=\frac{1}{N}(\tilde{\Phi}^{\mathrm{T}}W_{1,l})^{\mathrm{T}}\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\boldsymbol{\beta}^{(t)}=\frac{1}{N}W_{1,l}^{\mathrm{T}}\tilde{\Phi}\tilde{\Phi}^{\mathrm{T}}\boldsymbol{v}_t=0$, $\forall t$. $\boldsymbol{\lambda}^{(1)}$ can then be derived as

$$\boldsymbol{\lambda}^{(1)}=-2\left(G\tilde{K}G^{\mathrm{T}}\right)^{-1}G\tilde{K}^2\boldsymbol{\beta}^{(t)} \tag{3.29}$$

In the same manner of obtaining $\boldsymbol{\lambda}^{(1)}$ in Eq. (3.28), we acquire $\boldsymbol{\lambda}^{(2)}$ by multiplying $\Upsilon_{1,t-1}^{\mathrm{T}}$

to $\partial L/\partial \boldsymbol{\beta}^{(t)}$ as follows,

$$\begin{aligned}
&\Upsilon_{1,t-1}^{\mathrm{T}} \frac{\partial L}{\partial \boldsymbol{\beta}^{(t)}} \\
=&\Upsilon_{1,t-1}^{\mathrm{T}} \{ \frac{2}{N} \tilde{K}^2 \boldsymbol{\beta}^{(t)} + \frac{1}{N} \tilde{K} G^{\mathrm{T}} \boldsymbol{\lambda}^{(1)} + \tilde{K} \Upsilon_{1,t-1} \boldsymbol{\lambda}^{(2)} + 2\lambda^{(3)} \tilde{K} \boldsymbol{\beta}^{(t)} \} \\
=&\frac{2}{N} \Upsilon_{1,t-1}^{\mathrm{T}} \tilde{K}^2 \boldsymbol{\beta}^{(t)} + \frac{1}{N} \Upsilon_{1,t-1}^{\mathrm{T}} \tilde{\Phi}^{\mathrm{T}} \tilde{\Phi} G^{\mathrm{T}} \boldsymbol{\lambda}^{(1)} + \Upsilon_{1,t-1}^{\mathrm{T}} \tilde{\Phi}^{\mathrm{T}} \tilde{\Phi} \Upsilon_{1,t-1} \boldsymbol{\lambda}^{(2)} + 2\lambda^{(3)} \Upsilon_{1,t-1}^{\mathrm{T}} \tilde{K} \boldsymbol{\beta}^{(t)} \quad (3.30) \\
=&\frac{2}{N} \Upsilon_{1,t-1}^{\mathrm{T}} \tilde{K}^2 \boldsymbol{\beta}^{(t)} + \frac{1}{N} (V_{1,t-1}^{\mathrm{T}} \tilde{\Phi} G^{\mathrm{T}}) \boldsymbol{\lambda}^{(1)} + V_{1,t-1}^{\mathrm{T}} V_{1,t-1} \boldsymbol{\lambda}^{(2)} + 2\lambda^{(3)} \frac{\partial L}{\partial \boldsymbol{\lambda}^{(2)}} \\
=&\frac{2}{N} \Upsilon_{1,t-1}^{\mathrm{T}} \tilde{K}^2 \boldsymbol{\beta}^{(t)} + \boldsymbol{\lambda}^{(2)}
\end{aligned}$$

where $G\tilde{\Phi}^{\mathrm{T}} V_{1,t-1} = G\tilde{K}\Upsilon_{1,t-1} = 0$ according to Eqs. (3.25) and (3.20) for $\boldsymbol{v}_i$'s, $\forall i \in \{1, \cdots, t-1\}$ and $V_{1,t-1}^{\mathrm{T}} V_{1,t-1} = I$ due to the unity and orthogonal constraints of $\|\boldsymbol{v}_t\|^2 = 1$ and $V_{1,t-1}^{\mathrm{T}} \boldsymbol{v}_t = 0$ in Eq. (3.16). Based on Eq. (3.30), we can then solve for $\boldsymbol{\lambda}^{(2)}$ as follows,

$$\boldsymbol{\lambda}^{(2)} = -\frac{2}{N} \Upsilon_{1,t-1}^{\mathrm{T}} \tilde{K}^2 \beta^{(\mathrm{t})} \tag{3.31}$$

$\lambda^{(3)}$ is obtained by multiplying ${\boldsymbol{\beta}^{(t)}}^{\mathrm{T}}$ to $\partial L/\partial \boldsymbol{\beta}^{(t)}$ as follows,

$$\begin{aligned}
{\boldsymbol{\beta}^{(t)}}^{\mathrm{T}} \frac{\partial L}{\partial \boldsymbol{\beta}^{(t)}} =& {\boldsymbol{\beta}^{(t)}}^{\mathrm{T}} \left\{ \frac{2}{N} \tilde{K}^2 \boldsymbol{\beta}^{(t)} + \frac{1}{N} \tilde{K} G^{\mathrm{T}} \boldsymbol{\lambda}^{(1)} + \tilde{K} \Upsilon_{1,t-1} \boldsymbol{\lambda}^{(2)} + 2\lambda^{(3)} \tilde{K} \boldsymbol{\beta}^{(t)} \right\} \\
=& \frac{2}{N} {\boldsymbol{\beta}^{(t)}}^{\mathrm{T}} \tilde{K}^2 \boldsymbol{\beta}^{(t)} + \left(\frac{\partial L}{\partial \boldsymbol{\lambda}^{(1)}} \right)^{\mathrm{T}} \boldsymbol{\lambda}^{(1)} + \left(\frac{\partial L}{\partial \boldsymbol{\lambda}^{(2)}} \right)^{\mathrm{T}} \boldsymbol{\lambda}^{(2)} + 2\lambda^{(3)} \left(\frac{\partial L}{\partial \lambda^{(3)}} + 1 \right) \quad (3.32) \\
=& \frac{2}{N} {\boldsymbol{\beta}^{(t)}}^{\mathrm{T}} \tilde{K}^2 \boldsymbol{\beta}^{(t)} + 2\lambda^{(3)}
\end{aligned}$$

From Eq. (3.32), $\lambda^{(3)}$ can be derived as follows,

$$\lambda^{(3)} = -\frac{1}{N} {\boldsymbol{\beta}^{(t)}}^{\mathrm{T}} \tilde{K}^2 \boldsymbol{\beta}^{(t)} \tag{3.33}$$

Substitute $\boldsymbol{\lambda}^{(1)}$, $\boldsymbol{\lambda}^{(2)}$, and $\lambda^{(3)}$ in Eqs. (3.29), (3.31), (3.33) to Eq. (3.24), we obtain an expression that is only dependent on $\boldsymbol{\beta}^{(t)}$,

$$\begin{aligned}
\frac{N}{2} \tilde{K}^{-1} \frac{\partial L}{\partial \boldsymbol{\beta}^{(t)}} =& \tilde{K} \boldsymbol{\beta}^{(t)} - G^{\mathrm{T}} \left(G \tilde{K} G^{\mathrm{T}} \right)^{-1} G \tilde{K}^2 \boldsymbol{\beta}^{(t)} \\
& - \Upsilon_{1,t-1} \Upsilon_{1,t-1}^{\mathrm{T}} \tilde{K}^2 \boldsymbol{\beta}^{(t)} - \left({\boldsymbol{\beta}^{(t)}}^{\mathrm{T}} \tilde{K}^2 \boldsymbol{\beta}^{(t)} \right) \boldsymbol{\beta}^{(t)}
\end{aligned} \tag{3.34}$$

Setting $\partial L/\partial \boldsymbol{\beta}^{(t)}$ to zero, we obtain the following eigen-formulation

$$E_t D_t = D_t \Lambda_t \tag{3.35}$$

where

$$E_t = \left[\mathrm{I} - \left\{ G^{\mathrm{T}} \left(G \tilde{K} G^{\mathrm{T}} \right)^{-1} G + \Upsilon_{1,t-1} \Upsilon_{1,t-1}^{\mathrm{T}} \right\} \tilde{K} \right] \tilde{K} \tag{3.36}$$

$D_t = [\boldsymbol{\beta}^{(t),1} \cdots \boldsymbol{\beta}^{(t),N}]$ with $\boldsymbol{\beta}^{(t),i}$ representing the i-th eigenvector and $\Lambda_t = diag(\lambda_{t,1}, \cdots, \lambda_{t,N})$ with $\lambda_{t,i}$ denoting the i-th eigenvalue corresponding to $\boldsymbol{\beta}^{(t),i}$ as $\lambda_{t,i} = \boldsymbol{\beta}^{(t),i^{\mathrm{T}}} \tilde{K}^2 \boldsymbol{\beta}^{(t),i}$.

By solving the eigen-formulation in Eq. (3.35) for E_t, the eigenvector, $\boldsymbol{\beta}^{(t)}$, corresponding to the maximum eigenvalue, $\lambda_t = \max_i(\lambda_{t,i})$ is chosen for $\boldsymbol{v}_t$ which maximizes correlation while satisfying the constraints in Eq. (3.16).

In $E_t \boldsymbol{\beta}^{(t),i} = \lambda_{t,i} \boldsymbol{\beta}^{(t),i}$, eigenvalue decomposition does not guarantee that eigenvalue $\lambda_{t,i}$ of E_t is identical to $\boldsymbol{\beta}^{(t),i^{\mathrm{T}}} \tilde{K}^2 \boldsymbol{\beta}^{(t),i}$ from Eq. (3.34) since there exist infinite many eigenvectors with different length but same direction, i.e., scaling of $\boldsymbol{\beta}^{(t),i}$ is required as follows,

$$E_t \left(\rho_i^{(t)} \boldsymbol{\beta}^{(t),i} \right) = \left\{ \left(\rho_i^{(t)} \boldsymbol{\beta}^{(t),i} \right)^{\mathrm{T}} \tilde{K}^2 \rho_i^{(t)} \boldsymbol{\beta}^{(t),i} \right\} \rho_i^{(t)} \boldsymbol{\beta}^{(t),i} \tag{3.37}$$

where $\rho_i^{(t)}$ is a scaling factor for $\boldsymbol{\beta}^{(t),i}$ and can be derived as follows based on Eq. 3.37,

$$\begin{aligned} E_t \boldsymbol{\beta}^{(t),i} &= \left\{ \left(\rho_i^{(t)} \boldsymbol{\beta}^{(t),i} \right)^{\mathrm{T}} \tilde{K}^2 \rho_i^{(t)} \boldsymbol{\beta}^{(t),i} \right\} \boldsymbol{\beta}^{(t),i} \\ &= \lambda_{t,i} \boldsymbol{\beta}^{(t),i} \\ \rho_i^{(t)} &= \sqrt{\lambda_{t,i}} \left(\boldsymbol{\beta}^{(t),i^{\mathrm{T}}} \tilde{K}^2 \boldsymbol{\beta}^{(t),i} \right)^{-1/2} \end{aligned} \tag{3.38}$$

Therefore, $\rho_i^{(t)} \boldsymbol{\beta}^{(t),i}$ is selected for the t-th eigenvector with maximum eigenvalue $\left((\rho_i^{(t)} \boldsymbol{\beta}^{(t),i})^{\mathrm{T}} \tilde{K}^2 \rho_i^{(t)} \boldsymbol{\beta}^{(t),i} \right)$ equivalent to the return of the objective function in Eq. (3.16).

To reduce computational complexity when performing the eigenvalue decomposition of Eq. (3.35), we further investigate into the problem and develop a non-iterative approach. First of all, the relationship between E_{t-1} and E_t can be derived from Eq. (3.36) as follows,

$$E_{t-1} = E_t + \boldsymbol{\beta}^{(t-1)} \boldsymbol{\beta}^{(t-1)^{\mathrm{T}}} \tilde{K}^2 \tag{3.39}$$

Multiplying both sides by $\boldsymbol{\beta}^{(t)}$, we have

$$\begin{aligned} E_{t-1}\boldsymbol{\beta}^{(t)} &= E_t\boldsymbol{\beta}^{(t)} + \boldsymbol{\beta}^{(t-1)}{\boldsymbol{\beta}^{(t-1)}}^{\mathrm{T}}\tilde{K}^2\boldsymbol{\beta}^{(t)} \\ &= \lambda_t\boldsymbol{\beta}^{(t)} \end{aligned} \tag{3.40}$$

where ${\boldsymbol{\beta}^{(t-1)}}^{\mathrm{T}}\tilde{K}^2\boldsymbol{\beta}^{(t)} = {\boldsymbol{\beta}^{(t-1)}}^{\mathrm{T}}\tilde{\Phi}^{\mathrm{T}}\tilde{\Phi}\boldsymbol{\beta}^{(t)} = \boldsymbol{v}_{t-1}^{\mathrm{T}}\boldsymbol{v}_t = 0$ by the orthogonal constraint in Eq. (3.16). Eq. (3.40) implies that we can find $\lambda_t = \max_i(\lambda_{t,i})$ from E_{t-1} and $\lambda_{t-1} \geq \lambda_t$ since λ_{t-1} is the maximum value among $\{\lambda_{t-1,i}, \forall i | E_{t-1}\}$ which includes λ_t. These relationships can be extended to

$$\lambda_1 \geq \cdots \geq \lambda_{t-1} \geq \lambda_t \geq \cdots \geq \lambda_N$$

and

$$E_1\boldsymbol{\beta}^{(t)}) = \lambda_t\boldsymbol{\beta}^{(t)}$$

which means the eigenvalues from E_1 include all λ_i's from E_i's, $\forall i \in \{1, \cdots, N\}$. Therefore, the one-time eigenvalue decomposition of E_1 without the orthogonal constraint provides complete set of eigenvalues corresponding to $\boldsymbol{\beta}^{(i)}$'s from E_i's, $\forall i$ in descending order without iterative eigenvalue decompositions.

I choose $\boldsymbol{\beta}^{(i)}$'s, $i \in \{1, \cdots, N\}$ only when the normalized eigenvalue is greater than or equal to the threshold, set to 0.1% in this dissertation. $\boldsymbol{v}_i$'s based on the selected $\boldsymbol{\beta}^{(i)}$'s with scaling span uncorrelated subspace, as $V = [\boldsymbol{v}_1 \cdots \boldsymbol{v}_{m'}]$, $m' \leq N$ where m' denotes the number of selected $\boldsymbol{\beta}^{(i)}$'s.

3.3.3 Nonlinear Projection from ICA over Uncorrelated Subspace

After obtaining the uncorrelated subspace spanned by V, the observation data, $\boldsymbol{x}$ is projected onto the subspace, resulting in $\boldsymbol{z}$, where the independent component analysis (ICA) is applied. ICA provides linearly unmixed signal $\boldsymbol{s}$, from mixed data $\boldsymbol{z}$, through unmixing matrix W as $\boldsymbol{s} = W^{\mathrm{T}}\boldsymbol{z}$. The projection of data, $\tilde{\Phi}$ onto the uncorrelated subspace is

obtained as follows,

$$\begin{aligned} Z &= V_{1,m'}^{\mathrm{T}} \tilde{\Phi} \\ &= \Upsilon_{1,m'}^{\mathrm{T}} \tilde{\Phi}^{\mathrm{T}} \tilde{\Phi} \\ &= \Upsilon_{1,m'}^{\mathrm{T}} \tilde{K} \end{aligned} \tag{3.41}$$

where $Z = [\boldsymbol{z}_1 \cdots \boldsymbol{z}_N]$. $\boldsymbol{z}_k \in R^{m'}$ represents the k-th projected data corresponding to $\boldsymbol{x}_k$. The linear ICA is then applied to Z in the uncorrelated subspace so as to derive the linear unmixing matrix $W_{l+1,m}$, consisting of $(m-l)$-many column vectors of $\boldsymbol{w}_i \in R^{m'}$, $i \in \{l+1, \cdots, m\}$. The linear ICA offers $W_{l+1,m}$ by maximizing independence among the components in S over Z.

$$S = W_{l+1,m}^{\mathrm{T}} Z \tag{3.42}$$

3.3.4 Conducting Dimensionality Reduction

The nonlinear SVM plus ICA aims at providing nonlinear embedding with minimum structural risk by SVM (Sec. 3.1) and maximum independence among data by ICA (Sec. 3.3.3). Figure 3.6 shows the proposed dimensionality reduction process to represent arbitrary input, $\boldsymbol{x}$ of higher dimension to $\boldsymbol{s}$ in reduced dimensional space. The $\bigoplus$ in the figure indicates the union of $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ into single vector representation as $\boldsymbol{s} = [\boldsymbol{s}_1^{\mathrm{T}}\ \boldsymbol{s}_2^{\mathrm{T}}]^{\mathrm{T}} \in R^{m \times 1}$. $\boldsymbol{s}_1 \in R^{l \times 1}$ satisfies minimum structural risk through nonlinear SVM-based supervised dimensionality reduction ($W_{1,l}$) whereas $\boldsymbol{s}_2 \in R^{(m-l) \times 1}$ is from maximum independence through ICA-based unsupervised dimensionality reduction ($W_{l+1,m}$) over nonlinear uncorrelated sub-

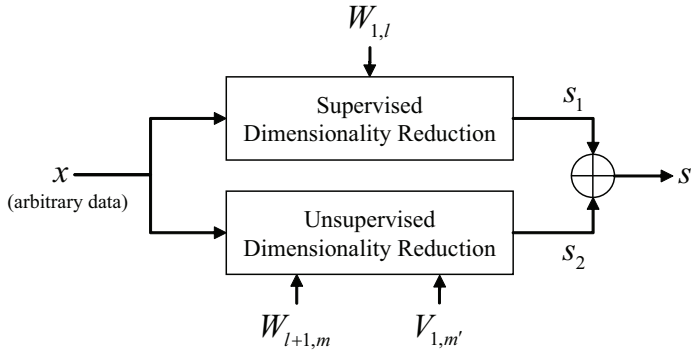


Figure 3.6: Dimensionality reduction in nonlinear SVM plus ICA

space spanned by $(V_{1,m})$.

The arbitrary input, $\boldsymbol{x}$ is interpreted in the space spanned by $\tilde{\Phi}$ with centered data representation as

$$\tilde{u}(\boldsymbol{x}) = \tilde{\Phi}^{\mathrm{T}}\tilde{\phi}(\boldsymbol{x}) = \left[\tilde{f}(\boldsymbol{x}_1, \boldsymbol{x}) \cdots \tilde{f}(\boldsymbol{x}_N, \boldsymbol{x})\right]^{\mathrm{T}}$$

where the centered kernel function $\tilde{f}$ can be calculated as follows,

$$\begin{aligned}
\tilde{f}(\boldsymbol{x}_k, \boldsymbol{x}) =& \langle (\phi(\boldsymbol{x}_k) - \boldsymbol{\mu}), (\phi(\boldsymbol{x}) - \boldsymbol{\mu}) \rangle \\
=& \langle \phi(\boldsymbol{x}_k), \phi(\boldsymbol{x}) \rangle + \frac{1}{N^2} \sum_{p,q=1}^{N} \langle \phi(\boldsymbol{x}_p), \phi(\boldsymbol{x}_q) \rangle \\
& - \frac{1}{N} \sum_{q=1}^{N} \langle \phi(\boldsymbol{x}_k), \phi(\boldsymbol{x}_q) \rangle - \frac{1}{N} \sum_{p=1}^{N} \langle \phi(\boldsymbol{x}_p), \phi(\boldsymbol{x}) \rangle \\
=& f(\boldsymbol{x}_k, \boldsymbol{x}) + \frac{1}{N^2} \left(1_{1\times N} K 1_{N\times 1}\right) - \frac{1}{N} \left(K_{k,k}^{\mathrm{T}} 1_{N\times 1} + \sum_{p=1}^{N} f(\boldsymbol{x}_p, \boldsymbol{x}) \right)
\end{aligned} \tag{3.43}$$

where $\boldsymbol{\mu}$ is the sample mean of $\phi(\boldsymbol{x}_k)$'s $\forall k$ as Eq. (3.17). $K_{p,q}$, $p \leq q$, denotes the submatrix consisting of the column vectors from the p-th to the q-th column in the Gram matrix K. The centered kernel implementation in Eq. (3.43) allows us to represent $\tilde{u}(\boldsymbol{x})$ by $u(\boldsymbol{x}) = \Phi^{\mathrm{T}}\phi(\boldsymbol{x}) = [f(\boldsymbol{x}_1, \boldsymbol{x}) \cdots f(\boldsymbol{x}_N, \boldsymbol{x})]^{\mathrm{T}}$ as follows,

$$\begin{aligned}
\tilde{u}(\boldsymbol{x}) =& u(\boldsymbol{x}) + \frac{1}{N^2} \left(1_{N\times N} K 1_{N\times 1}\right) - \frac{1}{N} \left(K 1_{N\times 1} + 1_{N\times N} u(\boldsymbol{x})\right) \\
=& \left(I - \frac{1}{N} 1_{N\times N}\right) u(\boldsymbol{x}) + \frac{1}{N} \left(\frac{1}{N} 1_{N\times N} - I\right) K 1_{N\times 1} \\
=& H u(\boldsymbol{x}) - \frac{1}{N} H K 1_{N\times 1}
\end{aligned} \tag{3.44}$$

Based on Eq. (3.44), the supervised SVM-based dimensionality reduction projects arbitrary data $\boldsymbol{x}$ onto nonlinear embedding, $W_{1,l}$ with kernel function as follows,

$$\begin{aligned}
\boldsymbol{s}_1 =& \tilde{W}_{1,l}^{\mathrm{T}} \tilde{\phi}(\boldsymbol{x}) \\
=& \tilde{A}^{\mathrm{T}} \tilde{\Phi}^{\mathrm{T}} \tilde{\phi}(\boldsymbol{x}) \\
=& \tilde{A}^{\mathrm{T}} \tilde{u}(\boldsymbol{x}) \\
=& (H\tilde{A})^{\mathrm{T}} u(\boldsymbol{x}) - \frac{1}{N} (H\tilde{A})^{\mathrm{T}} K 1_{N\times 1} \\
=& A^{\mathrm{T}} u(\boldsymbol{x}) + \boldsymbol{b}
\end{aligned} \tag{3.45}$$

where $\boldsymbol{s}_1 \in R^l$. Eq. (3.45) is simplified by A based on $H\tilde{A} = A$ from Eqs. (3.21) and (3.22) that $\tilde{A} = (HKH)^{-1}HKA = H^{-1}A$. $\boldsymbol{b} = -\frac{1}{N}A^{\mathrm{T}}K1_{N\times 1} \in R^{m\times 1}$.

The dimensionality reduction onto subspace constructed from unsupervised ICA is as follows,

$$\begin{aligned}
\boldsymbol{s}_2 =& W_{l+1,m}^{\mathrm{T}}\langle V_{1,m'}, \tilde{\phi}(\boldsymbol{x})\rangle \\
=& W_{l+1,m}^{\mathrm{T}}\Upsilon_{1,m'}^{\mathrm{T}}\tilde{\Phi}^{\mathrm{T}}\tilde{\phi}(\boldsymbol{x}) \\
=& \left(\Upsilon_{1,m'}W_{l+1,m}\right)^{\mathrm{T}}\tilde{u}(\boldsymbol{x}) \\
=& \left(H\Upsilon_{1,m'}W_{l+1,m}\right)^{\mathrm{T}}u(\boldsymbol{x}) - \frac{1}{N}W_{l+1,m}^{\mathrm{T}}\Upsilon_{1,m'}^{\mathrm{T}}HK1_{N\times 1} \\
=& B^{\mathrm{T}}u(\boldsymbol{x}) + \boldsymbol{h}
\end{aligned} \tag{3.46}$$

where $\boldsymbol{s}_2 \in R^{(m-l)}$, $B = H\Upsilon_{1,m'}W_{l+1,m} \in R^{N\times(m-l)}$, and $\boldsymbol{h} = -\frac{1}{N}W_{l+1,m}^{\mathrm{T}}\Upsilon_{1,m'}^{\mathrm{T}}HK1_{N\times 1} \in R^{(m-l)\times 1}$.

Consequently, the dimensionality reduction process in Fig. 3.6 is summarized by Eqs. (3.45) and (3.46) as follows,

$$\begin{aligned}
\boldsymbol{s} =& \begin{bmatrix} \boldsymbol{s}_1 \\ \boldsymbol{s}_2 \end{bmatrix} \\
=& \begin{bmatrix} \frac{A}{\zeta_1} & \frac{B}{\zeta_2} \end{bmatrix}^{\mathrm{T}} u(\boldsymbol{x}) + \begin{bmatrix} \boldsymbol{b} \\ \boldsymbol{h} \end{bmatrix}
\end{aligned} \tag{3.47}$$

where $\boldsymbol{s} \in R^m$. ζ_1 and ζ_2 are normalization factors for A and B to be equally contributed in magnitude to $\boldsymbol{s}$. The normalization factors should be determined based on the projection matrix $\tilde{W}_{1,l}$ for A and $V_{1,m'}W_{l+1,m}$ for B. Frobenius norm is adopted for normalization since the inner product for Frobenius norm in $\mathcal{F}$ can be obtained through kernel function while keeping the magnitude-wise relationship for individual projection vectors in either A or B. Hence, ζ_1 and ζ_2 can be derived as follows,

$$\begin{aligned}
\zeta_1 =& \mathrm{tr}\left(\tilde{W}_{1,l}^{\mathrm{T}}\tilde{W}_{1,l}\right)^{1/2} \\
=& \mathrm{tr}\left(W_{1,l}^{\mathrm{T}}W_{1,l}\right)^{1/2} \\
=& \mathrm{tr}\left(A^{\mathrm{T}}KA\right)^{1/2}
\end{aligned} \tag{3.48}$$

where $\tilde{W}_{1,l} = W_{1,l}$ is utilized. $\mathrm{tr}(\cdot)$ denotes the trace function.

$$\begin{aligned}\zeta_2 &= \mathrm{tr}\left(W_{l+1,m}^{\mathrm{T}} V_{1,m'}^{\mathrm{T}} V_{1,m'} W_{l+1,m}\right)^{1/2} \\ &= \mathrm{tr}\left(W_{l+1,m}^{\mathrm{T}} W_{l+1,m}\right)^{1/2}\end{aligned} \tag{3.49}$$

$V_{1,m'}^{\mathrm{T}} V_{1,m'}$ is canceled from ζ_2 due to the orthonormality of $V_{1,m'}$ in Eq. (3.16).

Chapter 4

Experimental Results

The performance of SVM+ICA is evaluated based on classification accuracy over two dataset, 'Arrhythmia' from UCI Machine Learning databases [Asuncion and Newman 2007] and 'Cancer' from the Center for Genome Research at MIT Whitehead Institute [Center for Genome Research MIT Whitehead Institute 2009] under noisy environment so as to demonstrate the effectiveness of the proposed dimensionality reduction method toward noisy data input.

The Arrhythmia dataset consists of 452 samples in R^{279} to classify 16 types of cardiac arrhythmia. I remove 3 cardiac types from the Arrhythmia dataset due to no corresponding samples found to the excluded 3 types in the dataset. Since the Arrhythmia dataset includes 408 missing elements in the samples, I replace the missing elements with uniformly distributed random numbers between the minimum and maximum values of those elements. The Cancer dataset is to distinguish 14 types of cancers using 16063 tumor gene expression signatures with no missing elements and is composed of the given 144 training and 46 testing data samples. The Arrhythmia and Cancer data shows distinctive characteristics in the number of samples per class and dimensionality. The number of samples per class varies from 2 to 245 in the Arrhythmia dataset whereas the Cancer dataset has relatively consistent amount of data from 8 to 24 per class. However, the samples in the Cancer dataset are represented in R^{16063} compared with R^{279} for the data in the Arrhythmia dataset. For noisy environment construction, I add gaussian noise to individual dimension independently for the entire data with Signal-To-Noise (SNR) ratio from 5[dB] to 50[dB].

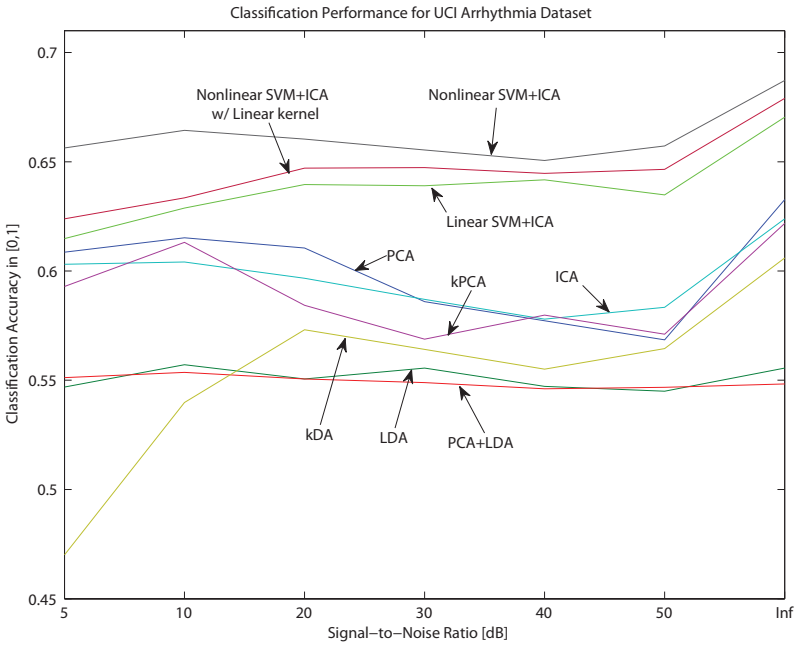
The noiseless data has SNR of ∞[dB].

I utilize the k-Nearest Neighbor classifier (kNN) as a performance measure of dimensionality reduction method due to its non-parametric nature. The classification accuracy by kNN is measured by average of multiple runs with n-fold cross validation. I apply 2-fold cross validation to the Arrhythmia dataset due to the minimum number of samples per class being 2 whereas no cross validation applied to the Cancer dataset due to the given training and testing data samples. The parameters are set to achieve the highest performance for each dataset where the SVM relaxation parameter C ranges from 0.01 to ∞, gaussian or equivalently Radial Basis Function (BRF) kernel with the kernel width of [0.01,∞], and the number of nearest neighbors, k in kNN from 1 to 50. I report the result that reaches 99.5% of the highest classification accuracy to avoid the case of extremely high dimensionality but with very little performance improvement.

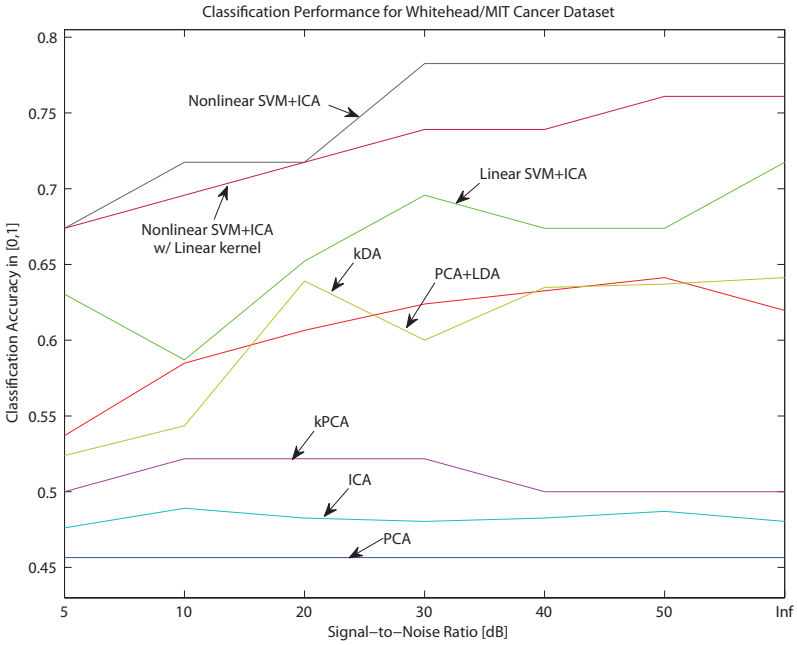
4.1 Comparison of Different Approaches

Figure 4.1 denotes the classification performance over noisy environment with various noise levels. I compare the the classification performance of linear and nonlinear SVM+ICA with that of PCA, kPCA, ICA for unsupervised, LDA, kDA for supervised, and PCA+LDA for hybrid approaches. For kernel-based method such as kPCA and kDA, I utilize the gaussian kernel with the kernel width selected in the range of [0.01,∞] to achieve the best classification performance. Since the cancer dataset has 144 training samples in 16063-dimensional space, its covariance matrix becomes 16063×16036 which is not applicable. Instead of direct covariance calculation, I apply Eigenface [Turk and Pentland 1991] to reduce the computational complexity when implementing the PCA and PCA+LDA approaches. I exclude LDA for the cancer dataset due to the difficulty in eigenvalue decomposition from the 16063×16063 matrix. Additionally, kPCA and kDA do not utilize covariance matrix, but the Gram matrix of 144×144.

I make four observations from Fig. 4.1. First of all, no matter what the SNR level is, the proposed SVM+ICA always presents the highest classification accuracy, demonstrating its supremacy over existing supervised, unsupervised or hybrid approaches. This remains true for both the balanced and imbalanced datasets. Second, the nonlinear SVM+ICA



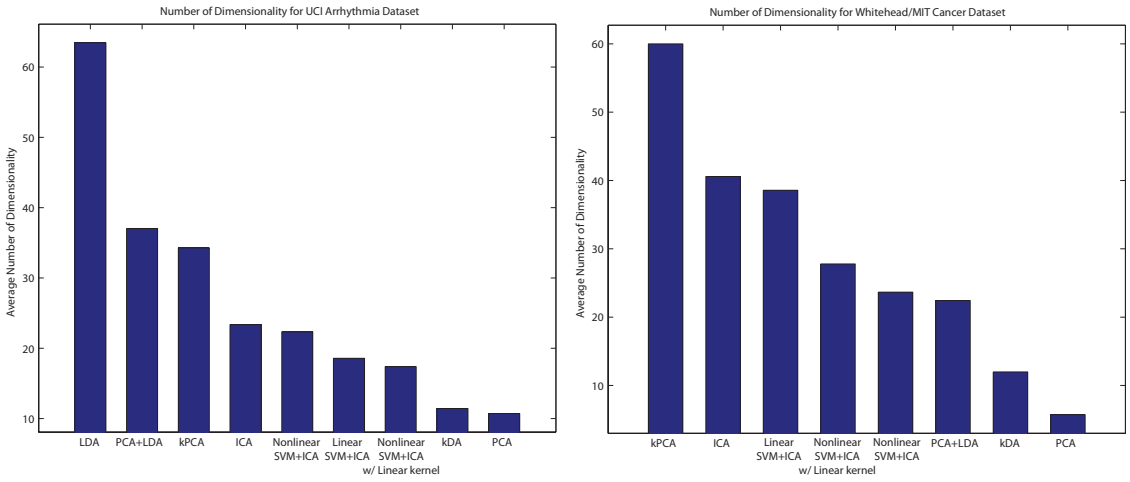
(a) Arrhythmia



(b) Cancer

Figure 4.1: Comparison of classification performance in noisy environment

achieves higher classification accuracy compared with linear one. It represents that nonlinear SVM+ICA is effective tool to reveal nonlinear nature of data than kPCA and kDA



(a) Arrhythmia (b) Cancer

Figure 4.2: Comparison of reduced dimensionality

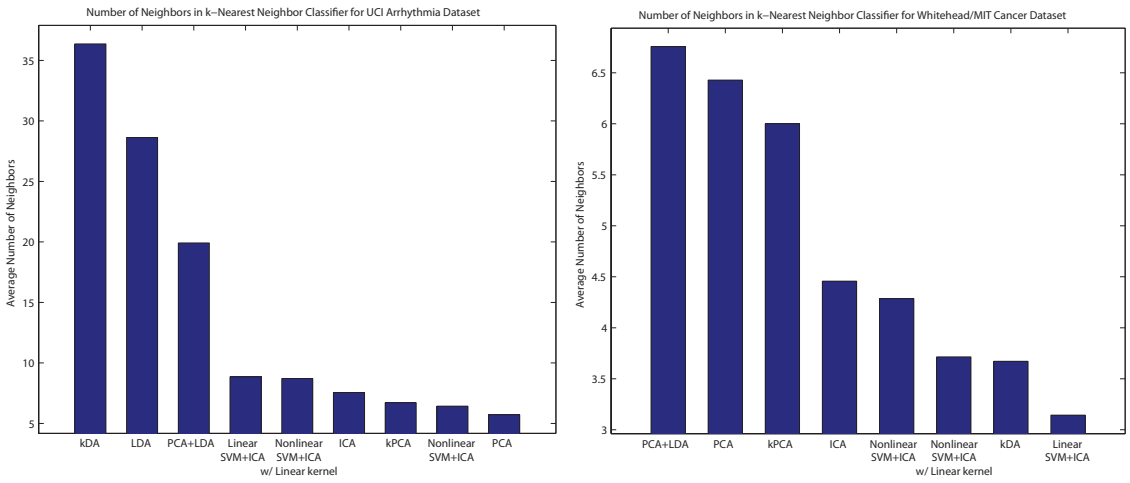
where they failed to outperform PCA and LDA. Third, for imbalanced datasets like the Arrhythmia, the unsupervised methods such as PCA, kPCA, and ICA work better than the supervised methods such as LDA and kDA or the hybrid approach of PCA+LDA. And for the cancer dataset with relatively well-balanced data distribution, the supervised methods generally outperform the unsupervised methods. In both cases, the proposed SVM+ICA, with its seamless integration of the supervised SVM and unsupervised ICA, presents the best overall performance. Fourth, the nonlinear SVM+ICA with linear kernel works better than the linear SM+ICA due to the uncorrelatedness subspace, especially when the dataset is balanced or equivalently there exists enough data to estimate sample covariance for correlation analysis. However, nonlinear data representation capability of RBF kernel results in better performance than linear kernel in the nonlinear SVM+ICA.

Another presentation of the overall summary of Fig. 4.1 is provided in Table 4.1 with the classification accuracy and the corresponding reduced dimensionality.

Figure. 4.2 denotes the average dimensionality over various SNR's.. Since I fix the maximum dimensionality of kDA to the number of classes minus 1 which is equivalent to the rank of the between-class scatter matrix for multiclass datasets, the maximum dimensionality of kDA for the Arrhythmia and the cancer dataset is 12 and 13 shown in Fig. 4.2a and 4.2b, respectively. However, I do not apply the upper bound restriction to LDA or

Table 4.1: The overall classification performance summary with reduced dimensionality (the two numbers within the parentheses indicate the number of projection vectors from SVM and ICA, respectively. Lin:Linear kernel. RBF:Radial Basis Function kernel)

Dataset	Method	Classification Accuracy [%] with Reduced Dimensionality for Various Signal-to-Noise Ratios						
		SNR=5 [dB]	SNR=10 [dB]	SNR=20 [dB]	SNR=30 [dB]	SNR=40 [dB]	SNR=50 [dB]	SNR=∞
Arrhythmia	PCA	60.9 (**10**)	61.5 (**10**)	61.1 (**10**)	58.6 (**10**)	57.7 (**10**)	56.9 (**10**)	63.3 (15)
	LDA	54.7 (39)	55.7 (64)	55.1 (78)	55.6 (59)	54.7 (68)	54.5 (93)	55.6 (45)
	PCA+LDA	55.1 (43)	55.4 (36)	55.1 (29)	54.9 (37)	54.6 (36)	54.7 (40)	54.8 (38)
	ICA	60.3 (19)	60.4 (18)	59.7 (34)	58.7 (25)	57.8 (27)	58.3 (20)	62.4 (21)
	kPCA	59.3 (**10**)	61.3 (**10**)	58.4 (60)	56.9 (30)	58.0 (65)	57.1 (55)	62.2 (**10**)
	kDA	47.0 (12)	54.0 (**10**)	57.3 (12)	56.4 (12)	55.5 (**10**)	56.5 (12)	60.6 (12)
	Linear SVM+ICA	**61.5** (20) (3.5+16.5)	**62.8** (**10**) (7+3)	**63.9** (30) (13+17)	**63.9** (15) (13+2)	**64.2** (15) (13+2)	**63.5** (20) (13+7)	**67.0** (20) (11+9)
	Nonlinear SVM+ICA(Lin)	**62.4** (20) (5+15)	**63.4** (11) (7+4)	**64.7** (22) (13+9)	**64.7** (20) (13+7)	**64.5** (21) (13+8)	**64.6** (13) (12+1)	**67.9** (15) (13+2)
	Nonlinear SVM+ICA(BRF)	**65.6** (29) (11+18)	**66.4** (22) (11+11)	**66.0** (24) (12+12)	**65.5** (27) (13+14)	**65.1** (18) (13+5)	**65.7** (20) (13+7)	**68.7** (17) (13+4)
Cancer	PCA	45.7 (**5**)	45.7 (**10**)	45.7 (**5**)	45.7 (**5**)	45.7 (**5**)	45.7 (**5**)	45.7 (**5**)
	PCA+LDA	53.7 (34)	58.5 (31)	60.7 (22)	62.4 (17)	63.3 (20)	64.1 (18)	62.0 (17)
	ICA	47.6 (38)	48.9 (51)	48.3 (39)	48.0 (34)	48.3 (36)	48.7 (45)	48.0 (42)
	kPCA	50.0 (50)	52.2 (65)	52.2 (60)	52.2 (65)	50.0 (60)	50.0 (60)	50.0 (60)
	kDA	52.4 (12)	54.4 (11)	63.9 (12)	60.0 (13)	63.5 (12)	63.7 (12)	64.1 (13)
	Linear SVM+ICA	**63.0** (45) (14+31)	**58.7** (35) (14+21)	**65.2** (40) (9+31)	**69.6** (45) (9+36)	**67.4** (35) (9+26)	**67.4** (30) (14+16)	**71.7** (40) (9+31)
	Nonlinear SVM+ICA(Lin)	**67.4** (35) (14+21)	**69.6** (23) (14+9)	**71.7** (40) (13+27)	**73.9** (17) (14+3)	**73.9** (28) (14+14)	**76.1** (30) (14+16)	**76.1** (23) (14+9)
	Nonlinear SVM+ICA(RBF)	**67.4** (28) (14+14)	**71.7** (19) (14+5)	**71.7** (25) (14+11)	**78.3** (17) (9+8)	**78.3** (34) (14+20)	**78.3** (19) (14+5)	**78.3** (24) (14+10)



(a) Arrhythmia (b) Cancer

Figure 4.3: Number of neighbors in kNN

PCA+LDA such that I can observe the behavior of classification performance improvement with the introduction of information from the null space to compensate the linear model used in LDA against the nonlinear model used in kDA. For the imbalanced Arrhythmia dataset, in general, supervised methods return higher dimensionality than unsupervised methods. SVM+ICA stands close to unsupervised methods since the unbalanced dataset degrades the performance of supervised SVM so that SVM+ICA acts closer to unsupervised ICA. kDA has relatively small dimensionality than LDA and PCA+LDA due to the application of the upper limit. On the contrary, the balanced cancer dataset shows higher dimensionality from the unsupervised approaches but lower dimensionality from supervised approaches, with SVM+ICA stands in between due to the balanced contribution from both SVM and ICA. PCA is an exception here due to early saturation with poor classification performance. For both datasets, the reduced dimensionality by the nonlinear SVM+ICA with linear kernel is placed in between the linear and nonlinear SVM+ICA.

I also use the number of neighbors (k) in kNN to observe the performance sensitivity to different patterns of data distribution in the dataset. In Fig. 4.3a, it is clear that supervised kDA and LDA and hybrid PCA+LDA achieve their highest classification accuracies with large k's whereas unsupervised methods such as ICA, kPCA, and PCA utilize k's of approximately no more than 10. The large k results from the degraded performance of supervised

Table 4.2: Classification performance summary of the linear SVM+ICA for the Arrhythmia dataset

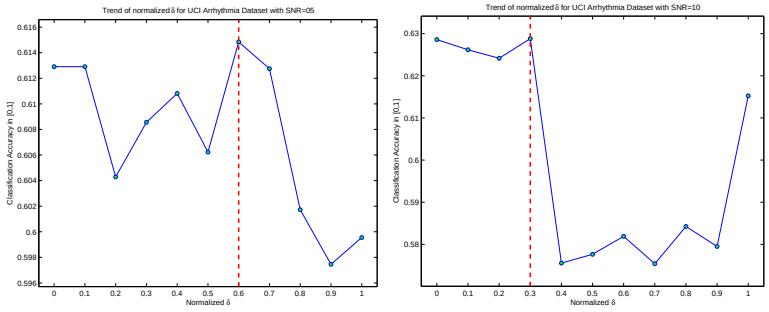
		Classification Accuracy [%] over various SNR's						
Class	samples	5[dB]	10[dB]	20[dB]	30[dB]	40[dB]	50[dB]	∞[dB]
1	245(54.2%)	94.7	95.1	93.9	90.6	97.1	92.7	95.9
2	44 (9.7%)	40.9	22.7	40.9	43.2	31.8	38.6	38.6
3	15 (3.3%)	60.0	66.7	26.7	46.7	20.0	26.7	60.0
4	15 (3.3%)	26.7	20.0	60.0	46.7	33.3	46.7	66.7
5	13 (2.9%)	0	7.7	0	0	0	0	0
6	25 (5.5%)	0	0	0	16.0	4.0	4.0	4.0
7	3 (0.7%)	0	0	0	0	0	0	0
8	2 (0.4%)	0	0	0	0	0	0	0
9	9 (2.0%)	0	11.1	11.1	33.3	11.1	0	11.1
10	50(11.1%)	30.0	52.0	54.0	54.0	56.0	62.0	60.0
11	4 (0.9%)	0	0	0	0	0	0	0
12	5 (1.1%)	0	0	0	0	0	0	0
13	22 (4.9%)	0	0	0	0	0	0	0
Total	accuracy	61.5	62.8	63.9	63.9	64.2	63.5	67.0
	dimension	20(4+16)	10(7+3)	30(13+17)	15(13+2)	15(13+2)	20(13+7)	20(11+9)

methods for the imbalanced Arrhythmia dataset shown in Fig. 4.1a. SVM+ICA requires intermediate k between the supervised and unsupervised, although k for SVM+ICA is close to the unsupervised. For the cancer dataset, all methods achieve their highest classification accuracies with relatively small k's where $k < 7$, as shown in Fig. 4.3b whereas SVM+ICA presents the relatively small k's among all. The nonlinear SVM+ICA with linear kernel requires the number of neighbors in between the linear and nonlinear SVM+ICA.

4.2 Class-wise Performance Comparison

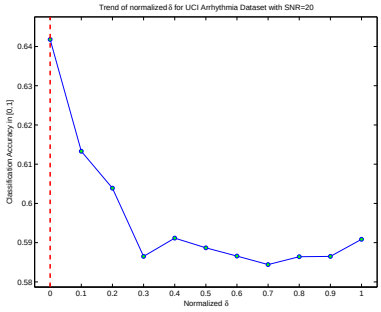
4.2.1 Linear SVM plus ICA

In order to study the effect of imbalanced vs. balanced data distribution, I study the class-wise classification performance using SVM+ICA. Table 4.2 shows the performance summary of the linear SVM+ICA for the Arrhythmia dataset. The 1st, 2nd, and 10-th classes include more than 9% of total number of data samples. Due to the sufficient number of data for training, the overall performance on this dataset is mostly dependent

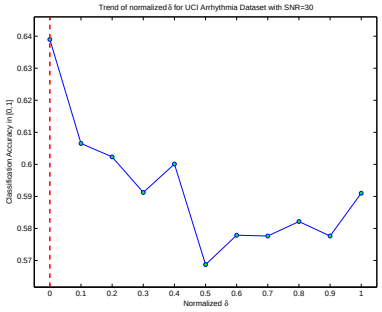


(a) SNR=5dB

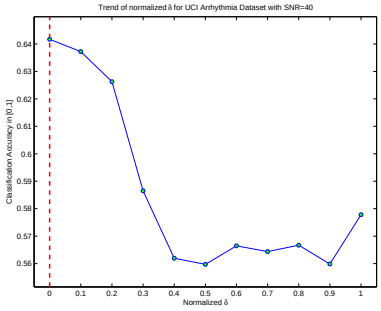
(b) SNR=10dB



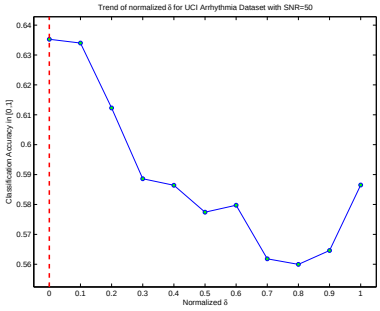
(c) SNR=20dB



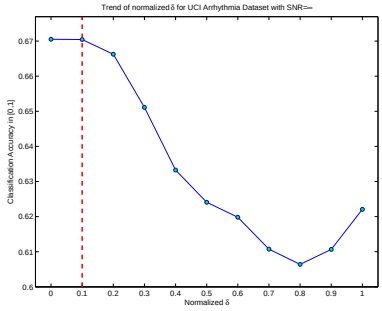
(d) SNR=30dB



(e) SNR=40dB



(f) SNR=50dB



(g) SNR=∞

Figure 4.4: Trend of classification accuracy corresponding to normalized δ in the linear SVM+ICA for the Arrhythmia dataset

on the performance of the three classes. However, classes 5, 7, 8, 9, 11, and 12 only have tiny portion of data samples (less than 3%) in 2-fold cross validation so that the linear SVM+ICA fails to construct appropriate dimensionality reduction model, resulting in poor classification accuracies.

Figure 4.4 shows the trend of classification accuracy corresponding to normalized δ (between 0 and 1) over various SNR's. By introducing normalized δ, I provide explicit correspondence of δ with the number of projection vectors from SVM. SVM+ICA selects

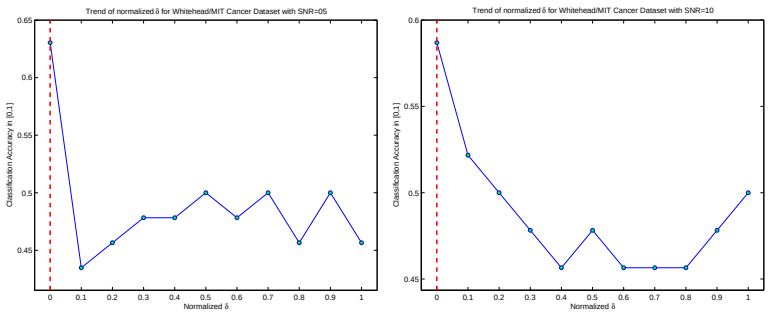
Table 4.3: Classification performance summary of the linear SVM+ICA for the Cancer dataset

		Classification Accuracy [%] over various SNR's						
Class	samples	5[dB]	10[dB]	20[dB]	30[dB]	40[dB]	50[dB]	∞[dB]
1	8 (5.6%)	66.7	0	33.3	33.3	33.3	0	33.3
2	8 (5.6%)	0	0	0	0	0	0	0
3	8 (5.6%)	33.3	33.3	33.3	0	33.3	33.3	33.3
4	8 (5.6%)	100.0	100.0	100.0	100.0	100.0	100.0	100.0
5	16(11.1%)	100.0	83.3	100.0	100.0	100.0	100.0	100.0
6	8 (5.6%)	66.7	66.7	33.3	66.7	33.3	66.7	100.0
7	8 (5.6%)	100.0	50.0	50.0	100.0	100.0	100.0	100.0
8	8 (5.6%)	50.0	50.0	100.0	100.0	100.0	100.0	100.0
9	24(16.7%)	53.3	66.7	66.7	83.3	66.7	66.7	66.7
10	8 (5.6%)	0	0	0	0	0	0	0
11	8 (5.6%)	33.3	66.7	100.0	100.0	100.0	100.0	100.0
12	8 (5.6%)	0	33.3	33.3	33.3	33.3	33.3	33.3
13	8 (5.6%)	66.7	100.0	100.0	100.0	100.0	100.0	100.0
14	16(11.1%)	100.0	100.0	100.0	100.0	100.0	100.0	100.0
Total	accuracy	63.0	58.7	65.2	69.6	67.4	67.4	71.7
	dimension	45(14+31)	35(14+21)	40(9+31)	45(9+36)	35(9+26)	30(14+16)	40(9+31)

normalized δ that generates the highest classification accuracy, represented by the dotted vertical line, so that any projection vectors from SVM with the normalized δ lower than the vertical line are eliminated by the redundancy removal process. I observe from Fig. 4.4 that the noise level largely affect the performance of SVM as the higher the noise level, the higher the normalized δ, and more projection vectors would be eliminated, resulting in degraded SVM.

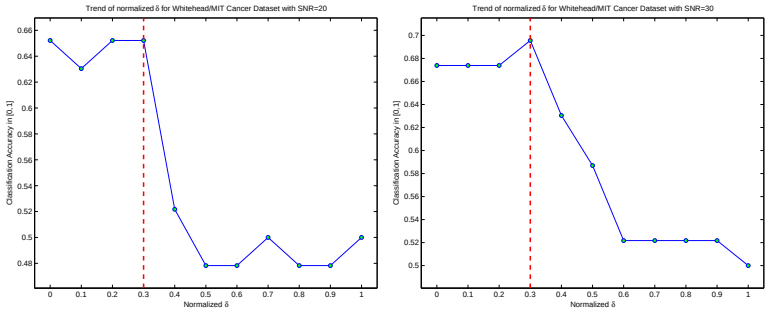
Table 4.3 shows the performance summary of the linear SVM+ICA for the cancer dataset. The cancer dataset has relatively balanced amount of data per class compared with the Arrhythmia dataset in Table 4.2. Since there exists no significant data imbalance in the cancer dataset, the poor performance from the 2nd and 10th classes is expected due to less informative training samples in the classes to reveal the nature of dataset by SVM.

As shown in Fig. 4.5, the noise level would not affect the performance much where the normalized δ across different noise levels is in general very small, when the dataset is with balanced data distribution such as the cancer dataset compared with Fig. 4.4 for the



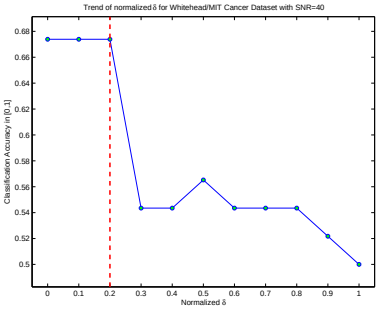
(a) SNR=5dB

(b) SNR=10dB

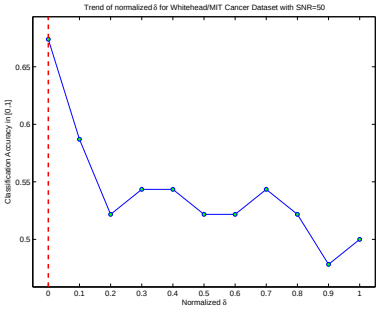


(c) SNR=20dB

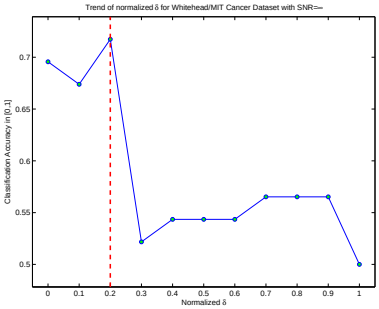
(d) SNR=30dB



(e) SNR=40dB



(f) SNR=50dB



(g) SNR=∞

Figure 4.5: Trend of classification accuracy corresponding to normalized δ in the linear SVM+ICA for Cancer dataset

imbalanced Arrhythmia dataset.

4.2.2 Nonlinear SVM plus ICA

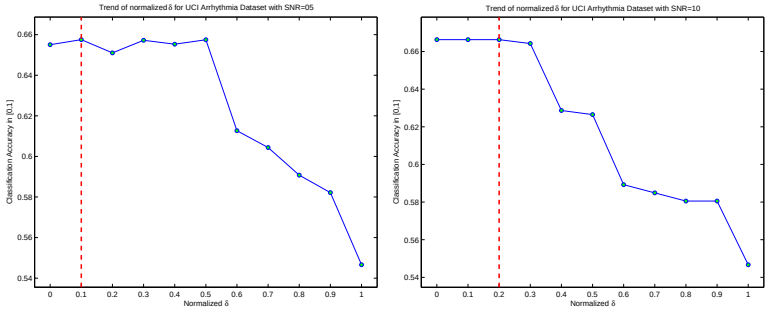
In order to study the effect of imbalanced vs. balanced data distribution, I study the class-wise classification performance using the nonlinear SVM+ICA. Table 4.4 shows the class-wise performance summary of the nonlinear SVM+ICA for the Arrhythmia dataset. From the data distribution per class in Table 4.4, I categorize the classes into three groups

Table 4.4: Classification performance summary of the nonlinear SVM+ICA for the Arrhythmia dataset

		Classification Accuracy [%] over various SNR's						
Class	samples	5[dB]	10[dB]	20[dB]	30[dB]	40[dB]	50[dB]	∞[dB]
1	245(54.2%)	93.1	94.7	94.1	91.8	94.3	91.8	95.3
2	44 (9.7%)	44.8	47.2	48.9	52.3	40.9	47.7	47.7
3	15 (3.3%)	72.4	80.0	66.7	73.3	46.7	40.0	70.0
4	15 (3.3%)	39.1	37.8	33.3	33.3	46.7	53.3	63.3
5	13 (2.9%)	0	0	3.9	0	0	0	0
6	25 (5.5%)	2.9	0	0	0	8.0	8.0	10.0
7	3 (0.7%)	0	0	0	0	0	0	0
8	2 (0.4%)	0	0	0	0	0	0	0
9	9 (2.0%)	58.7	55.3	55.6	44.4	11.1	11.1	22.2
10	50(11.1%)	51.4	49.5	52.0	56.0	56.0	66.0	61.0
11	4 (0.9%)	0	0	0	0	0	25.0	0
12	5 (1.1%)	0	0	0	0	0	0	0
13	22 (4.9%)	1.3	0	0	0	0	0	4.6
Total	accuracy	65.6	66.4	66.0	65.5	65.0	65.7	68.7
	dimension	29(11+18)	22(11+11)	24(12+12)	27(13+14)	18(13+5)	20(13+7)	17(13+4)

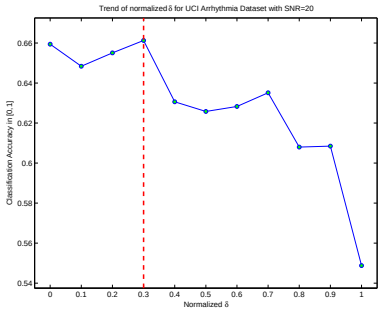
by the percentage of the number of data samples. The 1st group includes the 1st, 2nd, and 10-th classes with more than 9% of data. Due to the sufficient number of data for training especially in the 1st class, the overall performance on this dataset is mostly dependent on the performance of the three classes. The 3rd, 4th, 6th, and 13th classes are categorized into the 2nd group with $3\% \sim 9\%$ of the data resulting in minor contribution to the overall classification accuracy. From the 1st and 2nd groups, the 4th, 6th, and 10th classes show clear classification performance improvement toward the noise decreasing. However, classes 5, 7, 8, 9, 11, and 12 in the 3rd group only have tiny portion of data samples (less than 3%) in 2-fold cross validation so that the nonlinear SVM+ICA fails to construct appropriate dimensionality reduction model, resulting in poor classification accuracies.

Figure 4.6 represents the trend of the best classification accuracy to corresponding normalized $\delta \in [0, 1]$ over various SNR's. The normalized δ helps to provide explicit correspondence of δ with the removed projection vectors from SVM. The dotted vertical lines denotes the normalized δ selected at the highest classification accuracy so that any projection vectors from SVM with the normalized δ lower than the vertical line are eliminated by

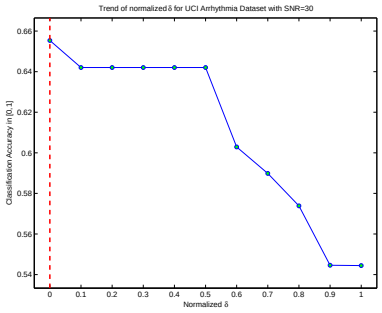


(a) SNR=5dB

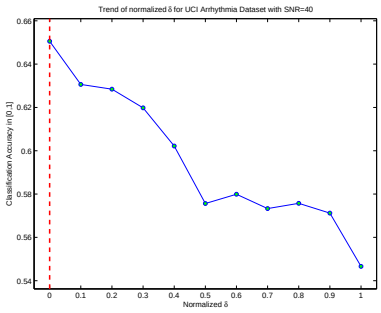
(b) SNR=10dB



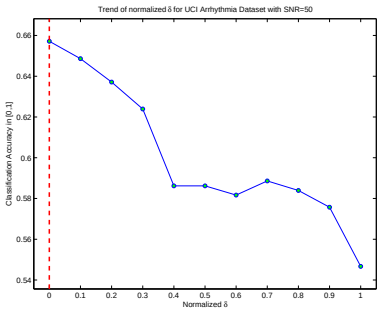
(c) SNR=20dB



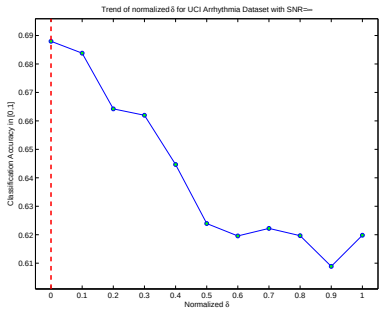
(d) SNR=30dB



(e) SNR=40dB



(f) SNR=50dB



(g) SNR=∞

Figure 4.6: Trend of classification accuracy corresponding to normalized δ in Nonlinear SVM+ICA for the Arrhythmia dataset

the redundancy removal process. I observe from Fig. 4.6 that the noise level largely affect the performance of SVM as the higher the noise level, the more projection vectors would be eliminated, resulting from more similarity among the projection vectors from SVM. For example, the number of projection vectors from SVM increases from 11 to 13 while SNR increases. Between 5[dB] and 10[dB] SNR's, normalized δ are chosen at 0.1 in Fig. 4.6a and 0.2 in Fig. 4.6b respectively, although the number of removed projection vectors are the same as 2. Therefore, projection vectors are more crowded around the minimum distance

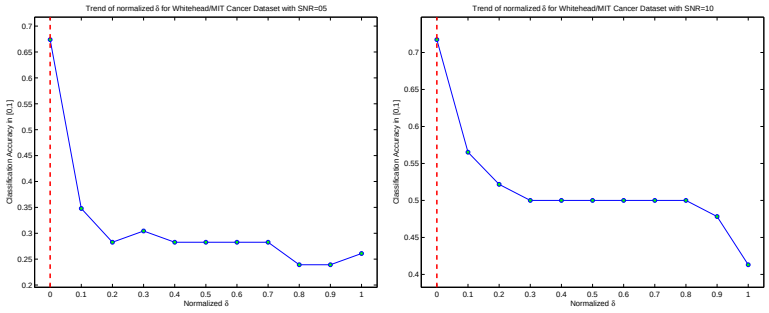
Table 4.5: The classification performance summary of the nonlinear SVM+ICA for the Cancer dataset

		Classification Accuracy [%] over various SNR's						
Class	samples	5[dB]	10[dB]	20[dB]	30[dB]	40[dB]	50[dB]	∞[dB]
1	8 (5.6%)	33.3	0	0	0	0	0	33.3
2	8 (5.6%)	0	0	0	0	0	0	0
3	8 (5.6%)	33.3	33.3	33.3	66.7	33.3	66.7	66.7
4	8 (5.6%)	100.0	100.0	100.0	100.0	100.0	100.0	100.0
5	16(11.1%)	83.3	66.7	83.3	100.0	100.0	100.0	83.3
6	8 (5.6%)	66.7	66.7	100.0	66.7	100.0	100.0	100.0
7	8 (5.6%)	100.0	100.0	50.0	100.0	100.0	100.0	100.0
8	8 (5.6%)	100.0	100.0	100.0	100.0	100.0	100.0	100.0
9	24(16.7%)	100.0	83.3	100.0	100.0	100.0	100.0	100.0
10	8 (5.6%)	0	100.0	33.3	33.3	33.3	33.3	33.3
11	8 (5.6%)	33.3	66.7	100.0	100.0	100.0	100.0	100.0
12	8 (5.6%)	33.3	66.7	33.3	66.7	66.7	33.3	33.3
13	8 (5.6%)	100.0	100.0	100.0	100.0	100.0	100.0	100.0
14	16(11.1%)	100.0	100.0	100.0	100.0	100.0	100.0	100.0
Total	accuracy	67.4	71.7	71.7	78.3	78.3	78.3	78.3
	dimension	28(14+14)	19(14+5)	25(14+11)	17(9+8)	34(14+20)	19(14+5)	24(14+10)

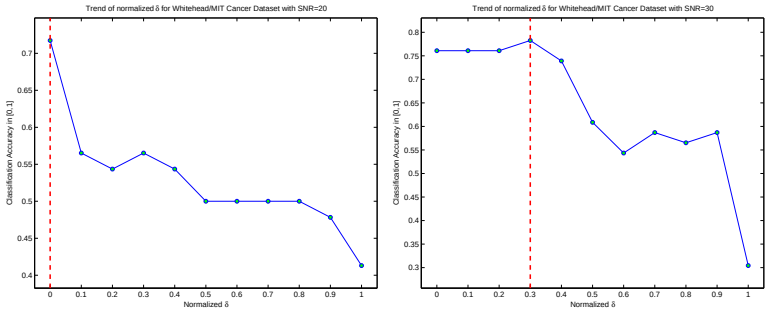
measured by asymmetric correlation metric under 5[dB] than 10[dB] noisy environment, resulting in more redundancy under higher noise. The identical interpretation can be made between the environment with SNR's of 10 and 20[dB] based on the increased normalized δ from 0.2 in Fig. 4.6b to 0.3 in Fig. 4.6c but the decreased number of SVM projection vectors from 2 to 1. More projection vectors from ICA work with the projection vector left from SVM for overall classification performance improvement under higher noise level.

Table 4.5 shows the class-wise performance summary of the nonlinear SVM+ICA for the Cancer dataset. The Cancer dataset has relatively balanced number of data per class compared with the class-wise distribution of number of data for Arrhythmia dataset in Table 4.4. Since there exists no significant data imbalance in the cancer dataset, the poor performance from the 2nd class is expected due to less informative training samples in the class to reveal the nature of the Cancer dataset by SVM. The 3rd, 6th, and 11th clearly contribute to the overall classification accuracy improvement when SNR's decrease.

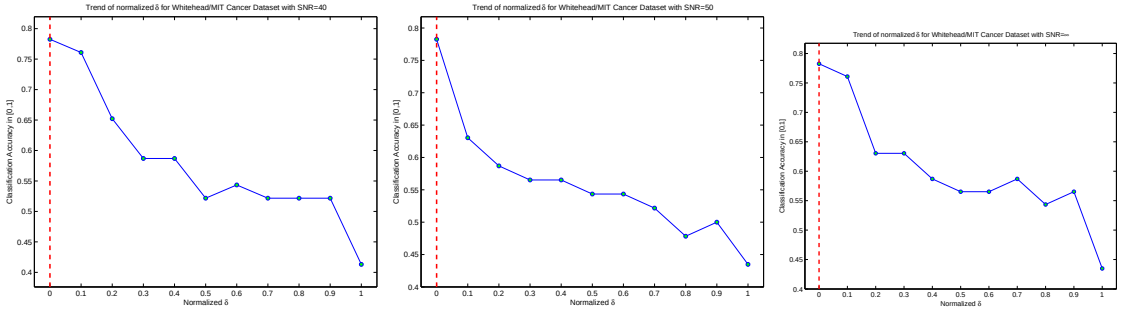
Figure 4.7 shows the trend of the best classification accuracy to corresponding normal-



(a) SNR=5dB (b) SNR=10dB



(c) SNR=20dB (d) SNR=30dB



(e) SNR=40dB (f) SNR=50dB (g) SNR=∞

Figure 4.7: Trend of classification accuracy corresponding to normalized δ in the nonlinear SVM+ICA for Cancer dataset

ized $\delta \in [0, 1]$ over various SNR's in nonlinear SVM+ICA for the Cancer dataset. Since the Cancer dataset is with balanced data distribution, the noise level would not affect the performance much, as shown in Fig. 4.7 where the normalized δ across different noise levels is very small in general.

Chapter 5

Conclusion

5.1 Summary

This dissertation proposed linear and nonlinear SVM plus ICA, hybrid dimensionality reduction algorithm. The linear SVM plus ICA provides projection that minimizes SVM-based structural risk in supervised manner and maximizes ICA-based data independence in unsupervised manner based on orthogonality whereas the nonlinear SVM+ICA provides nonlinear projection that optimizes SVM and ICA under uncorrelated relationship. Due to the power of structural risk minimization to pursue minimized empirical error and complexity in conjunction with independence maximization to find maximally independent features, the SVM plus ICA offers projection vectors as a mapping from observation to reduced dimensional space including advantages from both approaches simultaneously. The projection from nonlinear SVM plus ICA also offers nonlinear data representation capability by kernel. I showed experimental results that linear and nonlinear SVM plus ICA outperform other methods including conventional supervised, unsupervised, and hybrid approaches by providing better classification performance in relatively low reduced dimensional space under noisy and noise-free environment.

5.2 Future Research

There exist two major concerns for future research. First of all, the three optimization process of SVM, subspace construction, and ICA might be better to integrated into single

formulation for fast processing speed. In this case, the objective function will suffer from the computational complexity of the optimization. However, the simple/unified formulation might not only provide clearer understanding of the hybrid framework but also suppress the possible error accumulated from each of the steps in the subspace-based hybrid framework. The approach of nonlinear data representation is another concern. The clear advantage of the kernel-based method is easy-of-implementation. However, it is hard to determine which kernel to use as well as the free variables relying on the kernel. Manifold-based analysis is a promising alternative since it does not depend on nonlinear embedding through kernel function. However, the hardness to determine free variables in manifold-based approach results in the nonlinear data representation not to be extended easily based on manifold method.

Publications

Publications

1. **S. Moon** and H. Qi, "Nonlinear SVM plus ICA," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, under review.

2. **S. Moon** and H. Qi, "A Hybrid Dimensionality Reduction Method based on Support Vector Machine and Independent Component Analysis," in *IEEE Transactions on Neural Networks*, under review.

3. **S. Moon** and H. Qi, "Effective Dimensionality Reduction based on Support Vector Machine", in *International Conference on Pattern Recognition (ICPR)*, 2010.

4. **S. Moon** and H. Qi, "Hybrid Feature Extraction Framework based on Risk Minimization and Independence Maximization," *International Joint Conference on Neural Networks (IJCNN)*, pp.2141-2144, 2009.

Bibliography

Bibliography

Alzate, C. and Suykens, J. A. K. (2008). Kernel component analysis using an epsilon-insensitive robust loss function. *IEEE Transactions on Neural Networks*, 19(9):1583–1598.

Archambeau, C., Delannay, N., and Verleysen, M. (2008). Mixtures of robust probabilistic principal component analyzers. *Neurocomputing*, 71(7-9):1274–1282.

Asuncion, A. and Newman, D. (2007). UCI machine learning repository.

Bach, F. R. and Jordan, Michael, I. (2002). Kernel independent component analysis. *Journal of Machine Learning Research*, 3:1–48.

Belhumeur, P. N., Hespanha, J. P., and Kriegman, D. J. (1997). Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720.

Center for Genome Research MIT Whitehead Institute (2009). Cancer diagnosis dataset.

Cevikalp, H., Neamtu, M., Wilkes, M., and Barkana, A. (2005). Discriminative common vectors for face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(1):4–13.

Chang, K. I., Bowyer, K. W., and Flynn, P. J. (2005). An evaluation of multimodal 2D+3D face biometrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):619–624.

Chen, L.-F., Liao, H.-Y. M., Ko, M.-T., Lin, J.-C., and Yu, G.-J. (2000). A new LDA-

based face recognition system which can solve the small sample size problem. *Pattern Recognition*, 33(10):1713–1726.

Cheong, S., Oh, S. H., and Lee, S.-Y. (2004). Support vector machines with binary tree architecture for multi-class classification. *Neural Information Processing - Letters and Reviews*, 2(3):47–51.

Ciarlet, P. G. (1989). *Introduction to numerical linear algebra and optimisation*. Cambridge University Press.

Coello, C. A. C. and Christiansen, A. D. (1999). MOSES: A multiobjective optimization tool for engineering design. *Engineering Optimization*, 31(3):337 – 368.

Comon, P. (1994). Independent component analysis, a new concept? *Signal Processing*, 36(3):287–314.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.

Dhanjal, C., Gunn, S. R., and Shawe-Taylor, J. (2009). Efficient sparse kernel feature extraction based on partial least squares. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(8):1347–1361.

Ekenel, H. K. and Sankur, B. (2005). Multiresolution face recognition. *Image and Vision Computing*, 23(5):469–477.

Fei, B. and Jinbai, L. (2006). Binary tree of SVM: a new fast multiclass training and classification algorithm. *IEEE Transactions on Neural Networks*, 17(3):696–704.

Fisher, R. A. (1938). The statistical utilization of multiple measurements. *Annals of Eugenics (Cambridge)*, 8:376–386.

Foley, D. H. and Sammon Jr., J. W. (1975). An optimal set of discriminant vectors. *IEEE Transactions on Computers*, c-24(3):281–289.

Fonseca, C. M. and Fleming, P. J. (1993). Genetic algorithms for multiobjective optimization: formulation, discussion and generalization. In *Proceedings of the 5th International Conference on Genetic Algorithms*, pages 416–423.

Fukumizu, K., Bach, F. R., and Jordan, M. I. (2004). Dimensionality reduction for supervised learning with reproducing kernel hilbert spaces. *Journal of Machine Learning Research*, 5:73–99.

Gilad-Bachrach, R., Navot, A., and Tishby, N. (2004). Margin based feature selection - theory and algorithms. In *Proceedings of the International Conference on Machine Learning*, pages 43–50.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley Longman Publishing Co., Inc.

Goldberg, D. E., Deb, K., and Korb, B. (1991). Don't worry, be messy. In *Proceedings of the International Conference on Genetic Algorithms (ICGA)*, pages 24–30.

Guo, Y., Hastie, T., and Tibshirani, R. (2007). Regularized linear discriminant analysis and its application in microarrays. *Biostatistics*, 8(1):86–100.

Hajela, P. and Lin, C. Y. (1992). Genetic search strategies in multicriterion optimal design. *Structural and Multidisciplinary Optimization*, 4(2):99–107.

Herbrich, R. (2001). *Learning kernel classifiers: theory and algorithms*. MIT Press.

Holland, J. H. (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. The MIT Press, reprinted edition.

Horn, J., Nafpliotis, N., and Goldberg, D. E. (1994). A niched pareto genetic algorithm for multiobjective optimization. In *Proceedings of the IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, volume 1, pages 82–87.

Howland, P., Jeon, M., and Park, H. (2003). Structure preserving dimension reduction for clustered text data based on the generalized singular value decomposition. *SIAM Journal on Matrix Analysis & Applications*, 25(1):165.

Hsieh, P.-F. and Landgrebe, D. (1998). Linear feature extraction for multiclass problems. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, volume 4, pages 2050–2052 vol.4.

Hsieh, P.-F., Wang, D.-S., and Hsu, C.-W. (2006). A linear feature extraction for multiclass classification problems based on class mean and covariance discriminant information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(2):223–235.

Hsu, C.-W. and Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425.

Hyvarinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634.

Hyvarinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430.

Jiang, X. (2009). Asymmetric principal component and discriminant analyses for pattern classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):931–937.

Karhunen, J., Oja, E., Wang, L., Vigario, R., and Joutsensalo, J. (1997). A class of neural networks for independent component analysis. *IEEE Transactions on Neural Networks*, 8(3):486–504.

Kwak, K.-C. and Pedrycz, W. (2007). Face recognition using an enhanced independent component analysis approach. *IEEE Transactions on Neural Networks*, 18(2):530–541.

Kwak, N. (2008). Principal component analysis based on L1-norm maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9):1672–1680.

Kyperountas, M., Tefas, A., and Pitas, I. (2007). Weighted piecewise LDA for solving the small sample size problem in face verification. *IEEE Transactions on Neural Networks*, 18(2):506–519.

Lee, C. and Landgrebe, D. A. (1993). Feature extraction based on decision boundaries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):388–400.

Lee, D. D. and Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791.

Leiva-Murillo, J. M. and Artes-Rodriguez, A. (2007). Maximization of mutual information for supervised linear feature extraction. *IEEE Transactions on Neural Networks*, 18(5):1433–1441.

Loog, M., Duin, R. P. W., and Haeb-Umbach, R. (2001). Multiclass linear dimension reduction by weighted pairwise Fisher criteria. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(7):762–766.

Lu, H., Plataniotis, K. N., and Venetsanopoulos, A. N. (2008). MPCA: Multilinear principal component analysis of tensor objects. *IEEE Transactions on Neural Networks*, 19(1):18–39.

Lu, J., Plataniotis, K. N., Venetsanopoulos, A. N., and Li, S. Z. (2006). Ensemble-based discriminant learning with boosting for face recognition. *IEEE Transactions on Neural Networks*, 17(1):166–178.

Ma, Y., Yang, A. Y., Derksen, H., and Fossum, R. (2008). Estimation of subspace arrangements with applications in modeling and segmenting mixed data. *SIAM Review*, 50(3):413–458.

Martinez, A. M. and Kak, A. C. (2001). PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):228–233.

Michalewicz, Z. (1996). *Genetic algorithms + data structures = evolution programs*. Springer, 3rd and extended edition.

Mika, S., Ratsch, G., Weston, J., Scholkopf, B., and Mullers, K. R. (1999). Fisher discriminant analysis with kernels. In *Proceedings of the IEEE Signal Processing Society Workshop, Neural Networks for Signal Processing IX*, pages 41–48.

Momma, M. and Bennett, K. P. (2006). Constructing orthogonal latent features for arbitrary loss. In Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L. A., editors, *Feature Extraction: Foundations and Applications (Studies in Fuzziness and Soft Computing)*. Springer.

Muller, K. R., Mika, S., Ratsch, G., Tsuda, K., and Scholkopf, B. (2001). An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–201.

Nishino, K., Nayar, S. K., and Jebara, T. (2005). Clustered blockwise PCA for representing visual data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1675–1679.

Park, H., Jeon, M., and Rosen, J. B. (2003). Lower dimensional representation of text data based on centroids and least squares. *BIT Numerical Mathematics*, 43(2):427–448.

Park, H.-M., Oh, S.-H., and Lee, S.-Y. (2002). Adaptive noise cancelling based on independent component analysis. *Electronics Letters*, 38(15):832–833.

Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 6*, 2(11):559 – 572.

Platt, J. C., Cristianini, N., and Shawe-Taylor, J. (2000). Large margin DAGs for multiclass classification. In *Advances in Neural Information Processing Systems 12*, pages 547–553.

Rao, C. R. (1948). The utilization of multiple measurements in problems of biological classification. *Journal of the Royal Statistical Society. Series B (Methodological)*, 10(2):159–203.

Richardson, J. T., Palmer, M. R., Liepins, G. E., and Hilliard, M. R. (1989). Some guidelines for genetic algorithms with penalty functions. In *Proceedings of the 3rd Inter-*

national Conference on Genetic Algorithms, pages 191–197, George Mason University. Morgan Kaufmann Publishers Inc.

Ronald, S. (1997). Robust encodings in genetic algorithms: a survey of encoding issues. In *Proceedings of the IEEE International Conference on Evolutionary Computation*, pages 43–48.

Rosipal, R. and Trejo, L. J. (2002). Kernel partial least squares regression in reproducing kernel hilbert space. *Journal of Machine Learning Research*, 2:97–123.

Sanguinetti, G. (2008). Dimensionality reduction of clustered data sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(3):535–540.

Schaffer, J. D. (1985). Multiple objective optimization with vector evaluated genetic algorithms. In *Proceedings of the 1st International Conference on Genetic Algorithms*, pages 93–100. Lawrence Erlbaum Associates, Inc.

Scholkopf, B., Smola, A., and Muller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319.

Shashua, A. (1999). On the relationship between the support vector machine for classification and sparsified Fisher's linear discriminant. *Neural Processing Letters*, 9(2):129–139.

Steuer, J. (1986). *Multicriteria optimization: theory, computation, and application.* John Wiley, NY.

Tan, K. C., Khor, E. F., and Lee, T. H. (2005). *Multiobjective evolutionary algorithms and applications.* Advanced Information and Knowledge Processing. Springer-Verlag New York, Inc.

Tan, K. C., Khor, E. F., Lee, T. H., and Sathikannan, R. (2003). An evolutionary algorithm with advanced goal and priority specification for multiobjective optimization. *Journal of Artificial Intelligence Research*, 18:183–215.

Tao, Q., Chu, D., and Wang, J. (2008). Recursive support vector machines for dimensionality reduction. *IEEE Transactions on Neural Networks*, 19(1):189–193.

Tipping, M. E. and Bishop, C. M. (1999a). Mixtures of probabilistic principal component analyzers. *Neural Computation*, 11(2):443–482.

Tipping, M. E. and Bishop, C. M. (1999b). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, 61(3):611–622.

Tsang, I. W.-H., Kocsor, A., and Kwok, J. T.-Y. (2008). Large-scale maximum margin discriminant analysis using core vector machines. *IEEE Transactions on Neural Networks*, 19(4):610–624.

Turk, M. and Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86.

Vapnik, V. N. (1999). An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999.

Vidal, R., Ma, Y., and Sastry, S. (2005). Generalized principal component analysis (GPCA). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1945–1959.

Wang, C. and Wang, W. (2006). Links between PPCA and subspace methods for complete gaussian density estimation. *IEEE Transactions on Neural Networks*, 17(3):789–792.

Wold, H. (1966). Estimation of principal components and related models by iterative least squares. In *Multivariate Analysis*, pages 391–420. Academic Press, New Your.

Xiang, C., Fan, X. A., and Lee, T. H. (2006). Face recognition using recursive Fisher linear discriminant. *IEEE Transactions on Image Processing*, 15(8):2097–2105.

Xu, Y., Yang, J.-y., and Yang, J. (2004). A reformative kernel Fisher discriminant analysis. *Pattern Recognition*, 37(6):1299–1302.

Xu, Y., Zhang, D., Yang, J., and Yang, J.-Y. (2008). An approach for directly extracting features from matrix data and its application in face recognition. *Neurocomputing*, 71(10-12):1857–1865.

Yang, J. and Yang, J. (2001). Optimal FLD algorithm for facial feature extraction. In *Intelligent Robots and Computer Vision XX: Algorithms, Techniques, and Active Vision*, volume 4572, pages 438–444, Boston, MA, USA. SPIE.

Yang, J. and Yang, J.-Y. (2003). Why can LDA be performed in PCA transformed space? *Pattern Recognition*, 36(2):563–566.

Yang, J., Zhang, D., Frangi, A. F., and Yang, J.-y. (2004). Two-dimensional PCA: a new approach to appearance-based face representation and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(1):131–137.

Yang, J., Zhang, D., and Yang, J.-y. (2005). Is ICA significantly better than PCA for face recognition? In *Proceedings of the International Conference on Computer Vision*, volume 1, pages 198–203.

Yang, J., Zhang, D., and Yang, J.-Y. (2007). Constructing PCA baseline algorithms to reevaluate ICA-based face-recognition performance. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 37(4):1015–1021.

Ye, J. (2005). Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems. *Journal of Machine Learning Research*, 6(4):483–502.

Ye, J., Janardan, R., Park, C. H., and Park, H. (2004). An optimization criterion for generalized discriminant analysis on undersampled problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(8):982–994.

Zafeiriou, S. (2009). Discriminant nonnegative tensor factorization algorithms. *IEEE Transactions on Neural Networks*, 20(2):217–235.

Zafeiriou, S., Tefas, A., Buciu, I., and Pitas, I. (2006). Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification. *IEEE Transactions on Neural Networks*, 17(3):683–695.

Appendix

Appendix: Nomenclature

$1_{p,q}$	$p \times q$ matrix where all elements are 1's
A	$[\boldsymbol{a}^{(1)} \cdots \boldsymbol{a}^{(l)}]$
$E[\cdot]$	Expectation
I_n	$n \times n$ identity matrix
K	Gram matrix
$K_{i,j}$	Matrix consisting of the vectors from the i-th to j-th column of K, $i \leq j$
N	The number of training data
S_i	Set of support vectors for $\boldsymbol{w}_i$
$V_{1,m'}$	Uncorrelated subspace basis matrix
W	$[W_{1,l}\, W_{l+1,m}]$
$W_{1,l}$	Projection matrix from SVM, $W_{1,l} = [\boldsymbol{w}_1 \cdots \boldsymbol{w}_l]$
$W_{l+1,m}$	Projection matrix from ICA, $W_{l+1,m} = [\boldsymbol{w}_{l+1} \cdots \boldsymbol{w}_m]$
$X_{\tilde{\phi}}$	Random variable to represent the centered observation in $\mathcal{F}$
Φ	$[\phi(\boldsymbol{x}_1) \cdots \phi(\boldsymbol{x}_N)]$
$\Upsilon_{1,t-1}$	$[\boldsymbol{\beta}^{(1)} \cdots \boldsymbol{\beta}^{(t-1)}]$
$\alpha_k^{(i)}$	The k-th Lagrange multiplier corresponding to $\boldsymbol{x}_k$ for $\boldsymbol{w}_i$ in SVM's quadratic formulation
$\beta_k^{(t)}$	Weight corresponding to $\boldsymbol{x}_k$ for the t-th basis, $\boldsymbol{v}_t$ for uncorrelated subspace
$\boldsymbol{\beta}^{(t)}$	$[\beta_1^{(t)} \cdots \beta_N^{(t)}]^{\mathrm{T}}$
$\boldsymbol{a}^{(i)}$	$[a_1^{(i)} \cdots a_N^{(i)}]^{\mathrm{T}}$
$\boldsymbol{s}$	Data in the reduced dimensional space
$\boldsymbol{s}_1$	Data in reduced dimensional space by SVM
$\boldsymbol{s}_2$	Data in reduced dimensional space by ICA

$\boldsymbol{v}_t$	The t-th basis for uncorrelated subspace
$\boldsymbol{w}_i$	The i-th projection vector
$\boldsymbol{x}_k$	The k-th training data
$\boldsymbol{z}$	Projected data onto the subspace orthogonal to $W_{1,l}$
$\gamma_{\min}$	Lower bound of r_i
$\mathcal{F}$	Hyperdimensional space embedded by $\phi(\cdot)$
$\phi(\cdot)$	Nonlinear mapping function
θ_{ij}	Angular distance between $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$
$\tilde{K}$	Centered Gram matrix
$\tilde{f}(\cdot,\cdot)$	Centered kernel function
$\tilde{u}(\cdot)$	Projection onto the centered training data in $\mathcal{F}$
ζ_1	Scaling factor for SVM projection matrix
ζ_2	Scaling factor for ICA projection matrix
b_i	Bias for $\boldsymbol{w}_i$
c	The number of classes in the training dataset
d_{ij}	Asymmetric decorrelation metric
$f(\cdot,\cdot)$	Kernel function
$g(\cdot)$	Nonlinear function for non-Gaussianity
k_{ij}	Element in K at the i-th row and j-th column
m	The number of dimension to be reduced
r_i	Classification accuracy by $\boldsymbol{w}_i$
$sign(\cdot)$	Signum function
$y_k^{(i)}$	Class index $\in \{1,-1\}$ corresponding to $\boldsymbol{x}_k$ for $\boldsymbol{w}_i$

Vita

Sangwoo Moon was born in Seoul, Korea on September 30, 1975. He started his engineering career in Soongsil University in 1994 where he earned Bachelor of Science degree in 1998 and a Master of Science degree in 2000 from the Department of Electrical Engineering. To pursuit Ph.D., he enrolled the graduate program in the department of Electrical Engineering and Computer Science at the University of Tennessee, Knoxville in 2005. During his Ph.D. study, he focused on machine learning for face recognition, multimodal/multiscale image fusion, silicon stress analysis in wafer image, and hybrid dimensionality reduction for high-dimensional data. He will complete the Doctor of Philosophy degree in Computer Engineering in Summer 2010.