Nonlinear Dimension Reduction with Kernel Sliced Inverse Regression

Yuh-Jye Lee[1]

Joint work with Su-Yun Huang[2] and Yi-Ren Yeh[1]

[1]Department of Computer Science and Information Engineering, NTUST
[2]Institute of Statistical Science, Academia Sinica

Mini Workshop on Optimization
Dept. of Math., National Taiwan Normal University

October 18, 2007

Two Dimension Reduction Methods

Principal Component Analysis (PCA)

- The most popular dimension reduction method
- Based on the covariance matrix of input attributes
- Unsupervised method

Two Dimension Reduction Methods

Principal Component Analysis (PCA)

- The most popular dimension reduction method
- Based on the covariance matrix of input attributes
- Unsupervised method

Sliced Inverse Regression (SIR)

- SIR has won its reputation to perform well in dimension reduction and related applications
- Based on the conditional covariance matrix of input attributes on the responses
- Supervised method

PCA and SIR Directions for Ionosphere Dataset ($p = 35$)

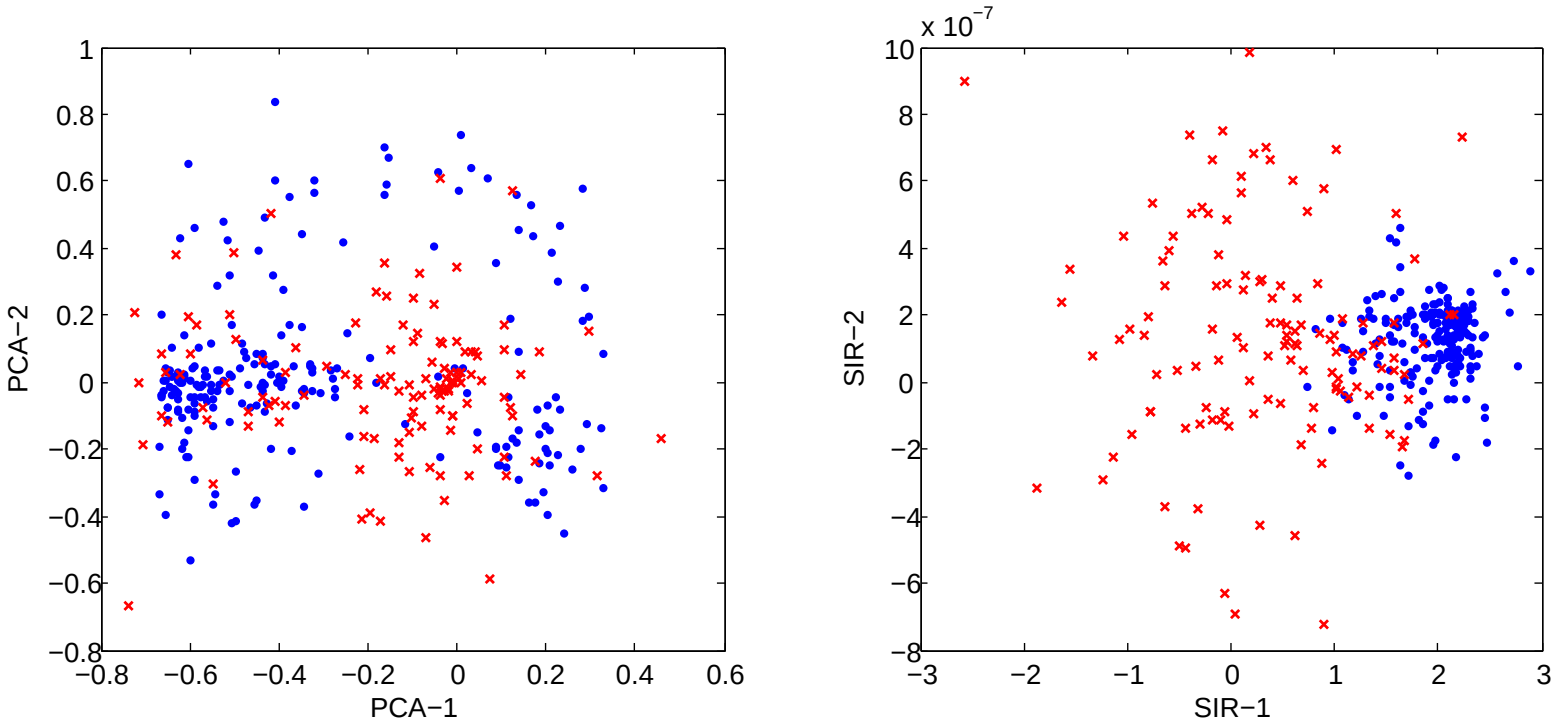


Figure: Positive: blue points, Negative: red cross

Let $A = [\mathbf{x_1}'; \cdots ; \mathbf{x_n}'] \in \mathbb{R}^{n \times p}$ be the data matrix of input attributes and $Y = [y_1; \cdots ; y_n] \in \mathbb{R}^n$ be the corresponding responses.

There Exists an *e.d.r.* Subspace

$$Y = f(\beta_1' \mathbf{x}, \ldots, \beta_d' \mathbf{x}; \epsilon), \tag{1}$$

where $\beta_j, \mathbf{x} \in \mathbb{R}^p$, d (often $\ll p$) is the effective dimensionality and $\{\beta_1, \ldots, \beta_d\}$ forms a basis of this effective dimension reduction (*e.d.r.*) subspace. Note f is unknown and can be nonlinear or linear form.

Linear Design Condition

For any b in $\mathbb{R}^p$, the conditional $E(b'\mathbf{x}|\beta_1'\mathbf{x}, \ldots, \beta_d'\mathbf{x})$ is linear in $\beta_1'\mathbf{x}, \ldots, \beta_d'\mathbf{x}$; that is, for some constants $c_0, c_1, \ldots, c_d$, $E(b'\mathbf{x}|\beta_1'\mathbf{x}, \ldots, \beta_d'\mathbf{x}) = c_0 + c_1\beta_1'\mathbf{x} + \ldots + c_d\beta_d'\mathbf{x}$.

Main Theorem (Li, 1991)

Theorem

Under condition (1) and L.D.C., the centered inverse regression curve $E(\mathbf{x}|y) - E(\mathbf{x})$ *is contained in the linear subspace spanned by* $\Sigma_{\mathbf{x}}\beta_i, i = 1, \ldots, d$, *where* $\Sigma_{\mathbf{x}}$ *is the covariance matrix of* $\mathbf{x}$.

- SIR is based on L.D.C. and (1)
- The theorem shows the inverse regression $E(\mathbf{x}|y) - E(\mathbf{x})$ indeed lies in a d-dimensional subspace which can be related to the *e.d.r.* subspace under these two conditions
- SIR estimate $E(\mathbf{x}|y)$ by a step function consisting of $A_h \subseteq A$ (ie., slice the data into several slices and estimate $E(\mathbf{x}|y)$ by the slice means)

The Formulation of SIR

From the theorem, SIR finds the *e.d.r.* directions by solving the following generalized eigenvalue problem:

$$\Sigma_{E(A|Y_J)}\beta = \lambda\Sigma_A\beta, \tag{2}$$

where Σ_A is the sample covariance matrix of A, Y_J denotes the membership of slices and there are J many slices, and $\Sigma_{E(A|Y_J)}$ denotes the between-slice sample covariance matrix based on sliced means given by

$$\Sigma_{E(A|Y_J)} = \frac{1}{n}\sum_{j=1}^{J} n_j(\bar{x}^j - \bar{x})(\bar{x}^j - \bar{x})'.$$

Here $\bar{x}$ is the sample grand mean, $\bar{x}^j = \frac{1}{n_j}\sum_{i\in S_j} x^i$ is the sample mean for the jth slice and S_j is the index set for jth slice.

Intuition Behind SIR

There is an intuitive way to describe SIR:

$$\max_{\beta \in \mathbb{R}^p} \ \beta' \Sigma_{E(A|Y_J)} \beta \ \text{ subject to } \beta' \Sigma_A \beta = 1. \tag{3}$$

Repeatedly solving (3) with the orthogonality constraints $\beta_k \Sigma_A \beta_l = \delta_{k,l}$, where $\delta_{k,l}$ is the Kronecker delta, the sequence of solutions form the basis of *e. d .r.* subspace.
Note that the slices are extracted from A according to the sorted responses Y. For classification, $\bar{x}^j$ is simply the sample mean of input attributes for the jth class.

- The classical SIR is designed to find a *linear* transformation from the input space to a low dimensional *e.d.r.* subspace
- SIR does not work for nonlinear feature extraction and it fails to find linear directions being in the null space or having small angles to the null space of $\Sigma_{E(\mathbf{x}|\mathbf{y})}$
- A remedy to this problem is the kernel extension for SIR
- We go back to the Support Vector Machine and see the "kernel trick"

Binary Classification Problem

Given a training dataset

$$S = \{(\mathbf{x}_i, y_i) | \mathbf{x}_i \in \mathbb{R}^p, y_i \in \{-1, 1\}, i = 1, \ldots, n\}$$

$$\mathbf{x}_i \in A_+ \Leftrightarrow y_i = 1 \ \& \ \mathbf{x}_i \in A_- \Leftrightarrow y_i = -1$$

Main Goal:

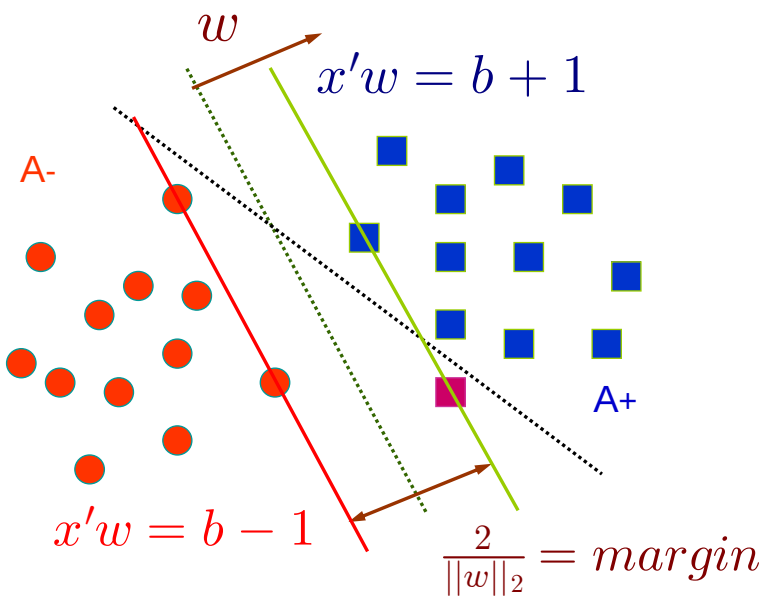
Predict the unseen class label for new data

Find a function $f : \mathbb{R}^n \to \mathbb{R}$ by learning from data

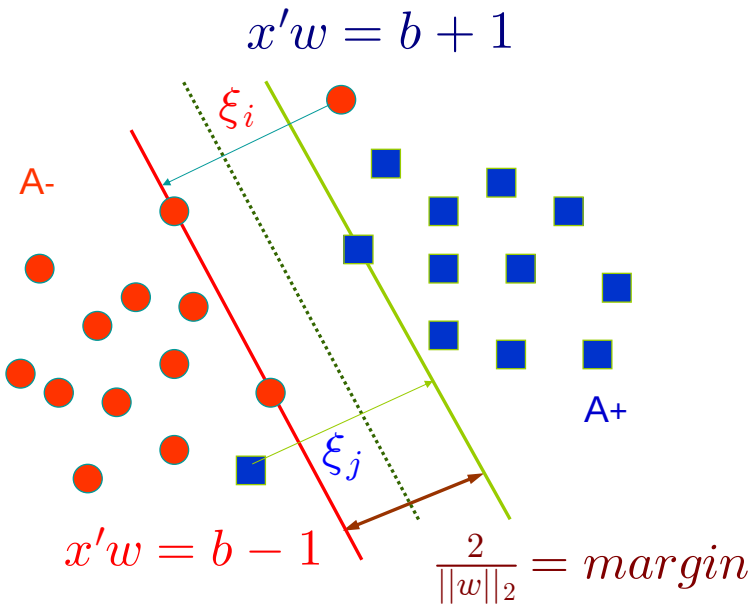
$$f(\mathbf{x}) \geq 0 \Rightarrow \mathbf{x} \in A_+ \text{ and } f(\mathbf{x}) < 0 \Rightarrow \mathbf{x} \in A_-$$

The simplest function is linear: $f(\mathbf{x}) = w'\mathbf{x} + b$

Maximum Margin Discriminant Hyperplane



(a) separable



(b) non-separable

Summary of Notation

Let $A = [\mathbf{x_1}'; \cdots ; \mathbf{x_n}'] \in \Re^{n \times p}$ be the data matrix of input attributes and $Y = [y_1; \cdots ; y_n] \in \{-1 \text{ or } 1\}^n$ be the corresponding responses as in SIR. For convenient, we also define

$$D = \begin{bmatrix} y_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & y_n \end{bmatrix} \in \Re^{n \times n}$$

$$\begin{array}{l} A_i w + b \geq +1, \quad for \quad D_{ii} = +1, \\ A_i w + b \leq -1, \quad for \quad D_{ii} = -1, \end{array} \text{ equivalent to}$$

$$D\,(Aw + 1b) \geq \mathbf{1}, \text{ where } \quad \mathbf{1} = [1, 1, \ldots, 1]' \in \Re^n.$$

Two Different SVM Formulations

2-Norm Soft Margin (Primal form):

$$\min_{(w,b,\xi)\in\mathbb{R}^{p+1+n}} \quad \frac{1}{2}\|w\|_2^2 + \frac{C}{2}\|\xi\|_2^2$$
$$D\left(Aw + \mathbf{1}b\right) + \xi \geq \mathbf{1}$$

1-Norm Soft Margin (Primal form):

$$\min_{(w,b,\xi)\in\mathbb{R}^{p+1+n}} \quad \frac{1}{2}\|w\|_2^2 + C\mathbf{1}'\xi$$
$$D\left(Aw + \mathbf{1}b\right) + \xi \geq \mathbf{1}, \quad \xi \geq 0$$

- Margin is maximized by minimizing reciprocal of margin.

SVM as an Unconstrained Minimization Problem

$$\begin{array}{ll} \min\limits_{w,b} & \frac{C}{2}\|\xi\|_2^2 + \frac{1}{2}\left(\|w\|_2^2 + b^2\right) \\ \textbf{s.t.} & D\left(Aw + \mathbf{1}b\right) + \xi \geq \mathbf{1} \end{array} \qquad (QP)$$

At the solution of (QP) : $\xi = (\mathbf{1} - D(Aw + \mathbf{1}b))_+$ where $(\cdot)_+ = max\{\cdot, 0\}$.

Hence (QP) is equivalent to the nonsmooth SVM:

$$\min_{w,b} \frac{C}{2}\|(\mathbf{1} - D(Aw + \mathbf{1}b))_+\|_2^2 + \frac{1}{2}(\|w\|_2^2 + b^2)$$

- Change (QP) into an unconstrained MP
- Reduce $(p + 1 + n)$ variables to $(p + 1)$ variables

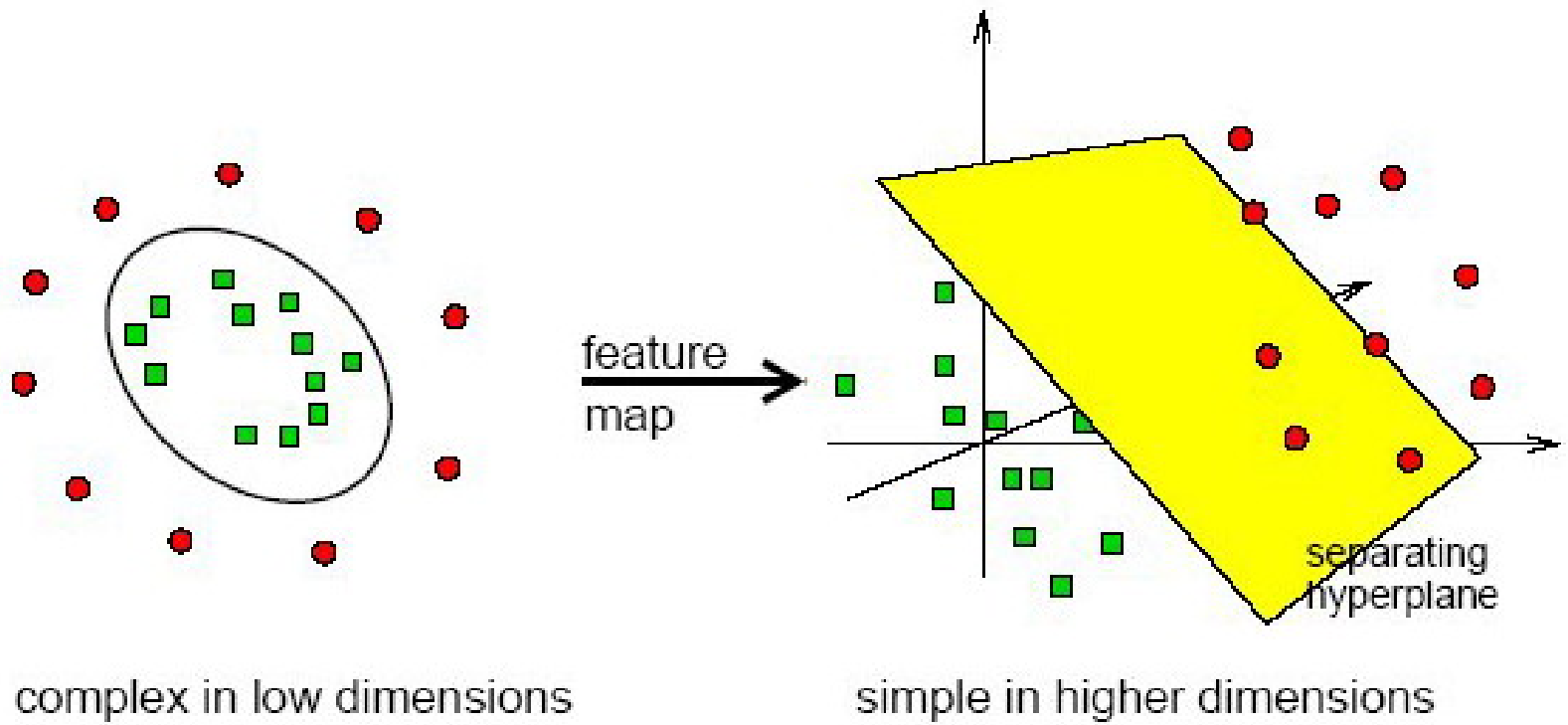
SSVM: Smooth Support Vector Machine

- Replacing the plus function $(\cdot)_+$ in the nonsmooth SVM by the smooth $p(\cdot, \beta)$, gives our SSVM:

$$\min_{(w,b)\in\mathbb{R}^{p+1}} \quad \frac{C}{2}\left\|p((\mathbf{1}-D(Aw+\mathbf{1}b)),\beta)\right\|_2^2+\frac{1}{2}(\|w\|_2^2+b^2),$$

  where $p(x,\beta) := x + \frac{1}{\beta}\log(1+e^{-\beta x})$.
- The solution of SSVM converges to the solution of nonsmooth SVM as β goes to infinity.
- It can be solved by Newton-Armijo Method and the complexity depends on dimension of input space (columns)

The Illustration of Nonlinear SVM



Nonlinear SVM Motivation

- Linear SVM: (Linear separator: $\mathbf{x}'w + b = 0$)

$$\begin{array}{ll} \min\limits_{\xi \geq 0, w, b} & \frac{C}{2}\|\xi\|_2^2 + \frac{1}{2}(\|w\|_2^2 + b^2) \\ \textbf{s.t.} & D(Aw + \mathbf{1}b) + \xi \geqslant \mathbf{1} \end{array} \quad (QP)$$

  By QP "duality", $w = A'D\alpha$ Maximizing the margin in the "dual space" gives:

$$\begin{array}{ll} \min\limits_{\xi \geq 0, \alpha, b} & \frac{C}{2}\|\xi\|_2^2 + \frac{1}{2}(\|\alpha\|_2^2 + b^2) \\ \textbf{s.t.} & D(AA'D\alpha + \mathbf{1}b) + \xi \geqslant \mathbf{1} \end{array}$$

- Dual SSVM with separator: $\mathbf{x}'A'D\alpha + b = 0$

$$\min_{\alpha, b} \frac{C}{2}\|p(\mathbf{1} - D(AA'D\alpha + \mathbf{1}b), \beta)\|_2^2 \quad + \frac{1}{2}(\|\alpha\|_2^2 + b^2)$$

Kernel Trick

- We can use the value of kernel function to represent the inner product of two training points in feature space as follows:

$$K(\mathbf{x}, \mathbf{z}) =< \phi(\mathbf{x}), \phi(\mathbf{z}) > .$$

- The most popular kernel function is the Gaussian kernel

$$K(\mathbf{x}, \mathbf{z}) = e^{-\gamma||\mathbf{x}-\mathbf{z}||_2^2}.$$

- The kernel matrix $K(A, A')_{n\times n}$ represents the inner product of all points in the feature space where $K(A, A')_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$.

Nonlinear SSVM Formulation

- Replace AA' by a nonlinear kernel $K(A, A')$ without defining a explicit feature map ϕ:

$$\min_{\alpha,b} \frac{C}{2}\|p(\mathbf{1} - D(K(A, A')\alpha + \mathbf{1}b), \beta)\|_2^2 \quad + \frac{1}{2}(\|\alpha\|_2^2 + b^2)$$

- Use Newton-Armijo algorithm to solve the problem
  - Each iteration solves $n + 1$ linear equations in $n + 1$ variables
- Nonlinear classifier depends on the data points with nonzero coefficients :

$$K(\mathbf{x}', A')\alpha + b = \sum_{\alpha_j \neq 0} \alpha_j K(A_j, x) + b$$

Reduced Kernel SSVM Formulation

- In the process of replacing the full kernel matrix by a reduced kernel, we use the Nyström approximation for the full kernel matrix:

$$K(A, A') \approx K(A, \tilde{A}')K(\tilde{A}, \tilde{A}')^{-1}K(\tilde{A}, A'), \tag{4}$$

  where $K(A, A') = K_{n\times n}$, $\tilde{A}_{\tilde{n}\times p}$ is a subset of A and $K(A, \tilde{A}) = \tilde{K}_{n\times\tilde{n}}$ is a reduced kernel.
- For a vector $\alpha \in \Re^n$ and $\tilde{\alpha} \in \Re^{\tilde{n}}$, we have

$$K(A, A')\alpha \approx K(A, \tilde{A}')K(\tilde{A}, \tilde{A}')^{-1}K(\tilde{A}', A)\alpha = K(A, \tilde{A}')\tilde{\alpha}.$$

- $\tilde{\alpha}$ is an approximated solution of α via the reduced kernel technique.

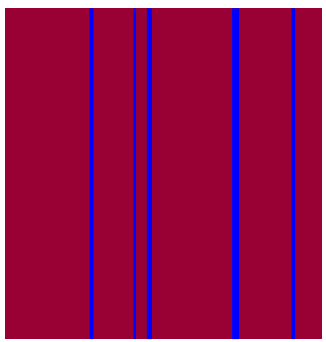
Nonlinear SVM vs. RSVM

Nonlinear SVM

$$\min_{\alpha,b,\xi\geqslant 0} \quad C\sum_{j=1}^{m}\xi_j + \frac{1}{2}\|\alpha\|_2^2$$
$$D(K(A,A')\alpha + \mathbf{1}b) + \xi \geqslant \mathbf{1}$$

RSVM

$$\min_{\tilde{\alpha},b,\xi\geqslant 0} \quad C\sum_{j=1}^{m}\xi_j + \frac{1}{2}\|\tilde{\alpha}\|_2^2$$
$$D(K(A,\tilde{A}')\tilde{\alpha} + \mathbf{1}b) + \xi \geqslant \mathbf{1}$$

$K(A,A'):$  $K(A,\tilde{A}'):$ 

Applying Kernel/Reduced Kernel Tricks on SIR

- Similar to applying the kernel trick on SSVM, we can extend SIR to Kernel SIR (KSIR)
- In the feature space, KSIR works with kernel data $K(A, A')$ as nonlinear SSVM
- The LDC in the finite basis approximation can be stated as:

$$E(a'K|\alpha_1'K, \ldots, \alpha_d'K) = c_0 + c_1\alpha_1'K + \cdots + c_d\alpha_d'K, \quad \forall a \in \mathbb{R}^n \tag{5}$$

Main Theorem in KSIR

Theorem

Assume the existence of an e.d.r. subspace $\mathcal{H} = \text{span}\{K(\mathbf{x}, A)\alpha_1, \ldots, K(\mathbf{x}, A)\alpha_d\}$ *and the LDC (5). Then the central inverse regression vector falls into the subspace spanned by* $\{\Sigma_K\alpha_1, \ldots, \Sigma_T\alpha_d\}$*, i.e.,*

$$E(K|\mathbf{y}) - E(K) \in \text{span}\{\Sigma_K\alpha_1, \ldots, \Sigma_K\alpha_d\}, \tag{6}$$

where Σ_K *is the covariance matrix of* $T = K(\mathbf{x}, A)'$.

- Note that we estimate $E(K|y)$ by the slice means of kernel data

Kernel Sliced Inverse Regression

- From the theorem above, the kernel sliced inverse regression finds the dimension reduction directions in feature space by solving the following generalized eigenvalue problem:

$$\Sigma_{E(K|\mathbf{y})}\alpha = \lambda\Sigma_K\alpha \tag{7}$$

  where $K = K(A, A) \in \mathbb{R}^{n\times n}$
- We implement KSIR in a different way for numerical stability and fast computation

Finding the Orthonormal Basis for *e.d.r* Subspace

- Define centered slice means of kernel data $W = [w_1 \cdots w_J]$ and the jth column is given by

$$w_j = \sqrt{n_j/n}\left(\frac{\mathbf{1}'_{n_j} K(A_{S_j}, A)}{n_j} - \frac{\mathbf{1}'_n K(A, A)}{n}\right)',$$

  where $\mathbf{1}'_{n_j} K(A_{S_j}, A)/n_j$ and $\mathbf{1}'_n K(A, A)/n$ are respectively the jth slice sample mean and the grand mean of $K(A, A)$
- $W'W = \Sigma_{E(K|Y_J)}$ is the between-slice sample covariance

Proposition

The orthonormalized *e.d.r.* directions are given by columns of $\Sigma_K^{-1} WUD^{-1/2}$, where U is computed form a small eigenvalue decomposition $W'\Sigma_K^{-1} W = UDU'$.

Finding the Orthonormal Basis for *e.d.r* Subspace

- We have $\mathcal{C}(\Sigma_K^{-1}W)$ is in the *e.d.r.* subspace from the theorem
- SVD is applied to the matrix $\Sigma_K^{-1/2}W$ as the normalization is in terms of $V'\Sigma_K V = I$,
- Only right singular vectors are needed and it can be solved from the following small eigenvalue decomposition:

$$(\Sigma_K^{-1/2}W)'(\Sigma_K^{-1/2}W) = W'\Sigma_K^{-1}W = UDU'$$

- Let $V = \Sigma_K^{-1}WUD^{-1/2}$. Its columns are still in the column space $\mathcal{C}(\Sigma_K^{-1}W)$ and hence are still in the *e.d.r.* subspace
- V is satisfying the orthonormality:

$$\begin{aligned} V'\Sigma_K V &= (D^{-1/2}U'W'\Sigma_K^{-1})\Sigma_K(\Sigma_K^{-1}WUD^{-1/2}) \\ &= D^{-1/2}U'UDU'UD^{-1/2} = I \end{aligned}$$

Motivation for Approximated KSIR

- In many real world applications, the effective rank of the covariance matrix of kernel data is very low
- The sample covariance matrix Σ_K is usually singular and causes numerical instability and poor *e.d.r.* directions estimation
- Adding a ridge-type regularization term is a common way to solve the numerical instability but acts like appending unnecessary and nuisance coordinate axes to the effective and useful axes
- An appropriate way to deal with the problem is to find a reduced-column approximation to K, denoted by $\tilde{K}$ which provides a good approximation to $\mathcal{C}(K)$

Approximation of KSIR

- Let $\tilde{P}$ be a projection matrix of size $n \times \tilde{n}$, which satisfies $\tilde{P}'\tilde{P} = I_{\tilde{n}}$
- Given a reduced-column kernel data $\tilde{K} := K\tilde{P}$, the approximation of KSIR is to solve the following reduced generalized eigenvalue problem:

$$\Sigma_{E(\tilde{K}|Y_J)}\tilde{\alpha} = \lambda\Sigma_{\tilde{K}}\tilde{\alpha}, \tag{8}$$

  which is of much smaller size, as $\tilde{n} \ll n$
- We can also apply Proposition to the reduced problem (8) and the resulting *e.d.r.* directions are given by $\tilde{V} = \Sigma_{\tilde{K}}^{-1}\tilde{W}\tilde{U}\tilde{D}^{-1/2}$, where $\tilde{U}$ and $\tilde{D}$ are the eigenvectors and eigenvalues for $\tilde{W}'\Sigma_{\tilde{K}}^{-1}\tilde{W}$

KSIR Algorithm

KSIR Algorithm

Input: reduced kernel matrix $\tilde{K}$ an $n \times \tilde{n}$ matrix and Y_J an n-vector.

Output: KSIR directions $V_{\tilde{n}\times(J-1)}$ and associated eigenvalues $d_{(J-1)\times 1}$.

1. Compute the centered and weighted slice means $\tilde{W}_{\tilde{n}\times J}$;

   // J is the number of slices //

2. Compute the covariance matrix $\Sigma_{\tilde{K}}$ of the reduced kernel;

3. Compute the eigenvalue decomposition of $\tilde{W}'\Sigma_{\tilde{K}}^{-1}\tilde{W}$ as $\tilde{U}\tilde{D}\tilde{U}'$;

   // $O(J^3)$ for solving the eigenvalue problem //

   // $\tilde{D}$ and $\tilde{U}$ consist of non-zero eigenvalues and associated eigenvectors //

   // $O(\tilde{n}^3)$ for solving the linear system $\Sigma_{\tilde{K}}X = \tilde{W}$ to get $\Sigma_{\tilde{K}}^{-1}\tilde{W}$ //

4. $V \leftarrow \Sigma_{\tilde{K}}^{-1}\tilde{W}\tilde{U}\tilde{D}^{-\frac{1}{2}}$; $d \leftarrow \text{diagonal}\{\tilde{D}\}$.

Reduced Kernel Approximation by Optimal Basis

- The SVD gives the optimal low-rank projection to get a reduced kernel
- P can be obtained from the SVD of $Cov(K)$:

$$Cov(K) := \Sigma_K = PSP' \approx \tilde{P}\tilde{S}\tilde{P}'.$$

Also note that

$$Cov(\tilde{K}) := \Sigma_{\tilde{K}} = \frac{1}{n}\tilde{P}'K\left(I_n - \frac{\mathbf{1}_n\mathbf{1}_n'}{n}\right)K\tilde{P} = \tilde{P}'\Sigma_K\tilde{P} = \tilde{S},$$

which makes the inverse of $\Sigma_{\tilde{K}}$ readily there

- This strategy only works for small to median sized kernel matrix and the complexity is $O(n^3)$

Reduced Kernel Approximation by Random Basis

- In the random subset approach we choose $\tilde{P}$ as a column subset from I_n
- It is the same idea with the reduced kernel SSVM and we have

$$K(A, A)\alpha \approx \tilde{K}K(\tilde{A}, \tilde{A})^{-1}\tilde{K}'\alpha = \tilde{K}\tilde{\alpha},$$

  where $\tilde{\alpha} = K(\tilde{A}, \tilde{A})^{-1}\tilde{K}'\alpha$ is an approximation to the full problem
- The resulting reduced kernel matrix $\tilde{K}$ has full column rank so that $\Sigma_{\tilde{K}}$ is well-conditioned and the complexity is $O(\tilde{n}^3)$
- The singularity problem can be resolved and the computational cost can be cut down at the same time

Experimental Setting

- We evaluate the effectiveness of KSIR on 5 binary classification datasets, 8 multi-class classification datasets and 6 regression datasets
- The R.S. column represents the ratio of the reduced set used in our experiments when applying reduced kernel technique
- Apply the hybrid of KSIR and linear learning algorithms on classification (FDA and SSVM) and regression problems (RLS)
- Compared our results with LIBSVM
- The Gaussian kernel $K(x, u) = \exp(-\gamma \|x - u\|^2)$ is used except for the `medline` data set

Description of Classification Datasets Used in Our Experiments

Data set	Classes	Training Size	Testing Size	Attributes	R.S. (%)
banana	2	400	4900	2	10%
tree	2	700	11692	18	10%
splice	2	1000	2175	60	10%
adult	2	32561	16281	123	1%
web	2	49749	14951	300	1%
Iris	3	150	-	4	10%
wine	3	178	-	13	10%
vehicle	4	846	-	18	20%
segment	7	2310	-	19	10%
dna	3	2000	1186	180	10%
satimage	6	4435	2000	36	20%
pendigits	10	7494	3498	16	4%
medline	5	1250	1250	22095	100%

Description of Regression Datasets Used in Our Experiments

Data set	Size	Attributes	R.S. (%)
housing	506	13	15%
Comp_Activ_1000	1000	21	5%
Kin_fh_1000	1000	32	5%
Comp_Activ	8129	21	5%
Kin_fh	8129	32	5%
Friedman	40768	10	1%

Peaks Dataset

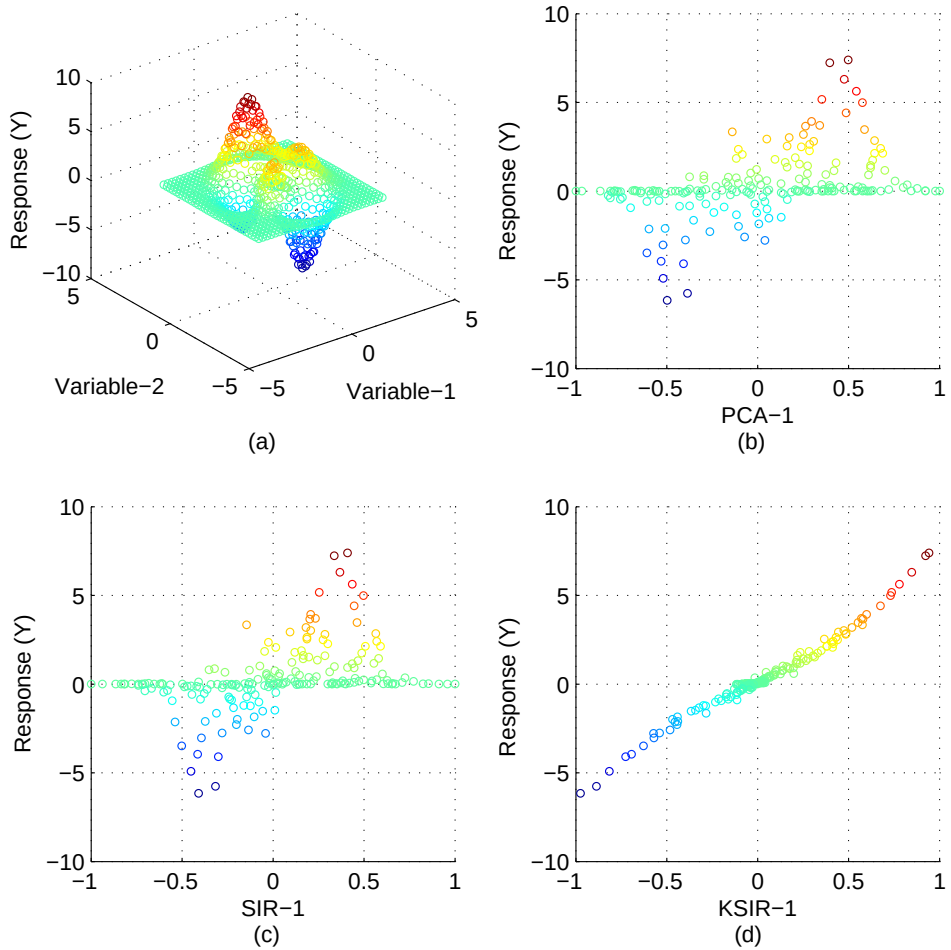


Figure: 2D views of response vs. the 1st variate by PCA, SIR and KSIR with `peaks` data.

Friedman Dataset

$\star X_1 \cdots X_{10} \in [0\ 1]$ and $Y = 10\sin(\pi X_1 X_2) + 20(X_3 - 0.5)^2 + 10\mathbf{x}_4 + 5\mathbf{x}_5 + \sigma(0,1)$

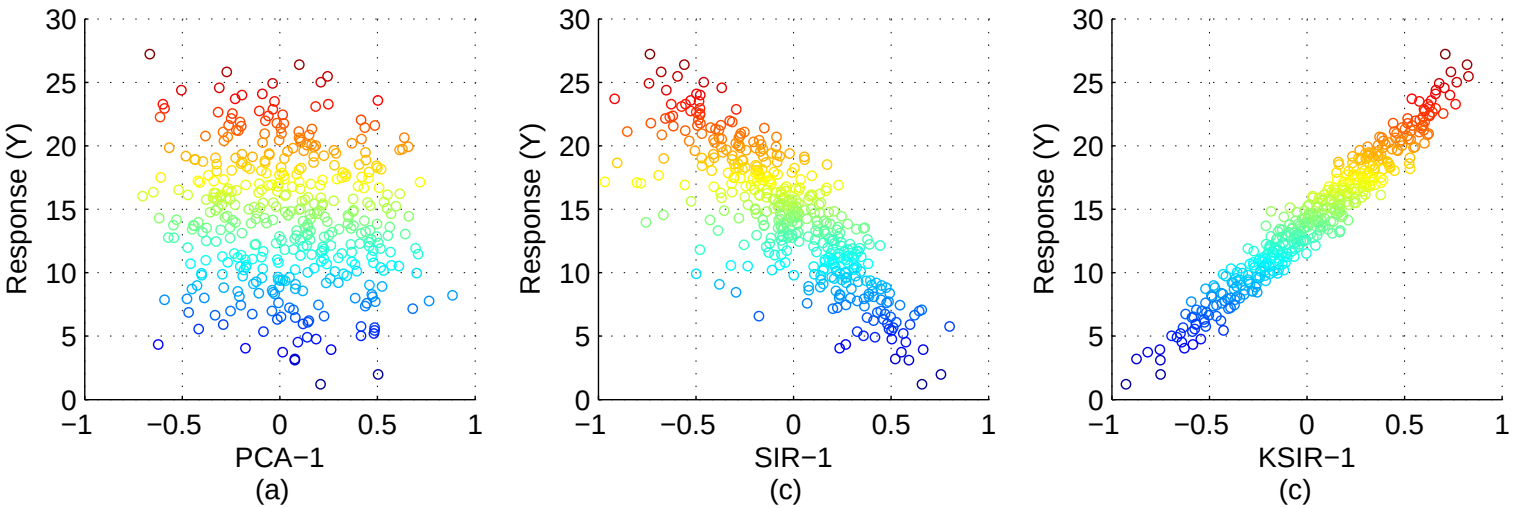


Figure: 2D views of Friedman data by PCA, SIR and KSIR.

Pendigits Dataset

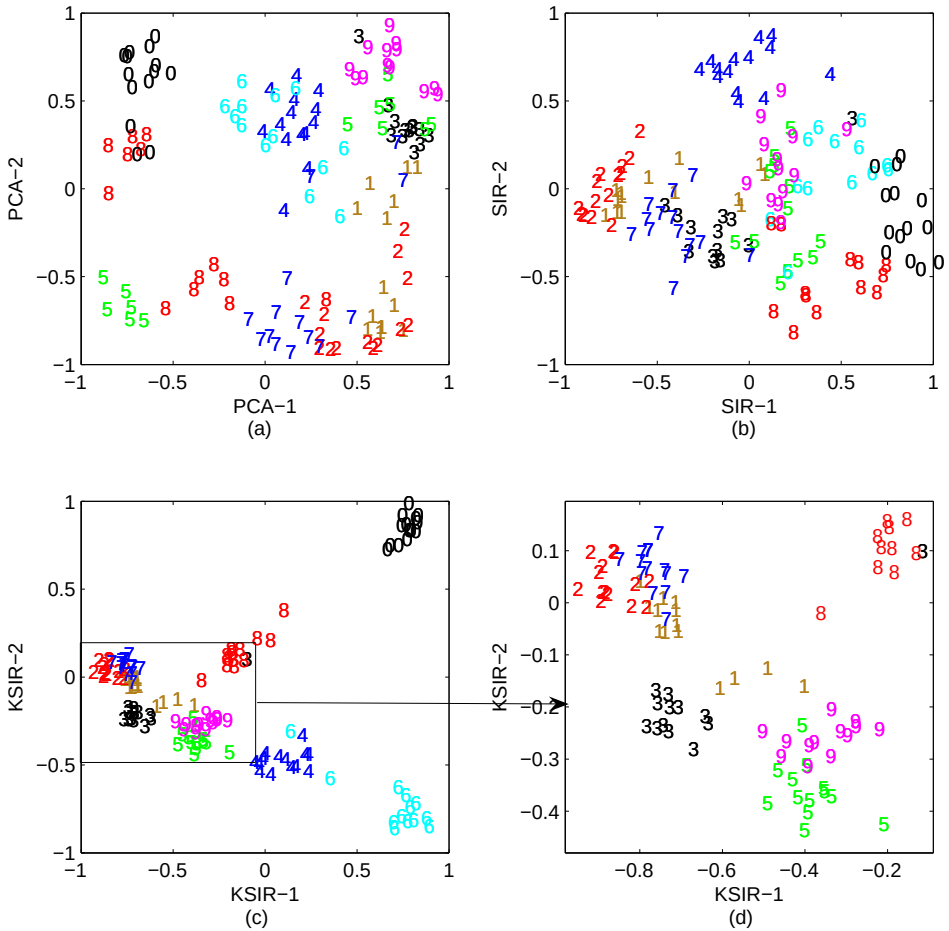


Figure: 2D views of `pendigits` data by PCA, SIR and KSIR.

Error Rates of Classification Datasets

Table: The average error rate for FDA and linear SSMV on KSIR variates compared with nonlinear LIBSVM on classification data sets.

Data set	KSIR+FDA	KSIR+SSVM	LIBSVM
banana	0.1170	0.1214	0.1228
tree	0.1234	0.1179	0.1283
splice	0.1292	0.1200	0.1012
adult	0.1671	0.1488	0.1491
web	0.0169	0.0149	0.0090
Iris	0.0213	0.0227	0.0380
wine	0.0131	0.0094	0.0181
vehicle	0.1468	0.1483	0.1429
segment	0.0309	0.0288	0.0283
dna	0.0659	0.0453	0.0460
satimage	0.0914	0.0904	0.0872
pendigits	0.0224	0.0188	0.0177
medline	0.1208	0.1136	0.1106

Training Time of Classification Datasets

Table: The training time (seconds) of FDA and linear SSVM on KSIR variates compared with nonlinear LIBSVM on classification data sets.

Data set	KSIR+FDA	KSIR+SSVM	LIBSVM
banana	0.063	0.078	0.016
tree	0.141	0.078	0.078
splice	0.109	0.109	0.422
adult	6.032	6.110	255.631
web	37.374	37.406	174.190
dna	0.329	0.344	2.900
satimage	3.828	3.953	4.593
pendigits	1.390	2.058	2.953
medline	1.993	2.016	3.033

R^2 of Regression Datasets

Table: R^2 of RLS on 3 and 29 KSIR variates compared with R^2 of nonlinear LIBSVM on regression data sets.

Data set	KSIR(3)+RLS	KSIR(29)+RLS	LIBSVM
housing	0.8543	0.8462	0.8687
Comp_Activ_1000	0.9685	0.9732	0.9776
Kin_fh_1000	0.6452	0.6482	0.6491
Comp_Activ	0.9760	0.9789	0.9820
Kin_fh	0.6964	0.6975	0.7014
Friedman	0.9556	0.9556	0.9559

- $R^2 = 1 - \frac{\|y-\hat{y}\|^2}{\|y-\bar{y}\|^2}$
- We fix at 30 slices in all our regression examples

Training Time of Regression Datasets

Table: The training time (seconds) of RLS on 3 and 29 KSIR variates compared with the training time of nonlinear LIBSVM on regression data sets.

Data set	KSIR(3)+RLS	KSIR(29)+RLS	LIBSVM
housing	0.022	0.040	0.211
Comp_Activ_1000	0.023	0.056	0.505
Kin_fh_1000	0.022	0.041	0.395
Comp_Activ	1.423	1.508	27.606
Kin_fh	1.416	1.502	20.950
Friedman	8.452	8.745	2400.1

Effect on the Number of Slices

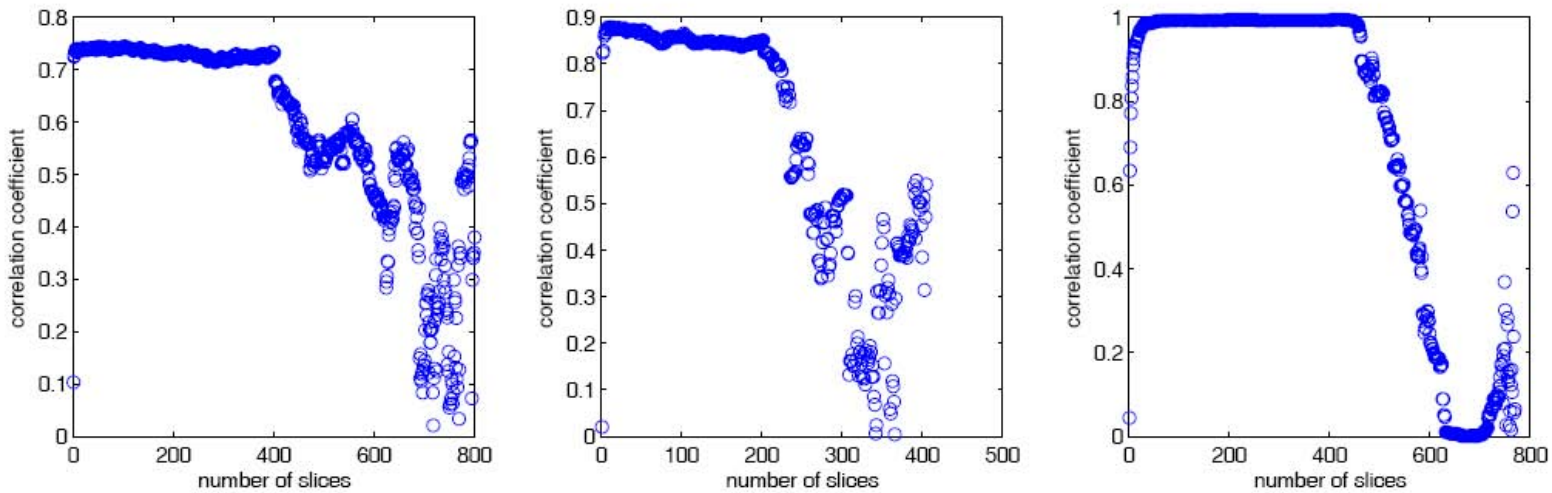


Figure: The variation of correlation coefficient on each slice number

Conclusion

- The KSIR algorithm first maps the pattern data to an appropriate feature space, and next extracts the main linear features in this embedded feature space
- After the extraction of the *e.d.r.* subspace, many supervised linear learning algorithms, such as FDA, SVM, and SVR, can be applied to the images of input data in this *e.d.r.* feature subspace
- In KSIR-based approach, it only involves solving the KSIR problem once and a series of C_2^J many linear binary SVMs in a $(J-1)$-dimensional space
- We have also incorporated reduced kernel approximation to cut down the computational load and to resolve the numerical instability

Parameter Tuning

- The naive tuning procedure, a two-dimensional grid search, in conventional SVMs is time consuming
- Tuning procedure for KSIR-based methods is *nearly* a one-dimensional search for γ
- KSIR-based methods are carried out in two stages
  - At the first stage, a parameter value for γ is needed for training KSIR *e.d.r.* subspace
  - At the second, a parameter value for C is needed for linear SSVM or RLS on KSIR variates
- This tuning procedure at the second stage for C is computationally light, as it is carried out in a very low-dimensional *e.d.r.* subspace
- For each fixed γ we can try a few C values to pair with this γ without much computing cost