Bayesian Kernel Fisher Discriminant Analysis

Sung-Chiang Lin
Institute of Statistical Science, Academia Sinica

1. Characterization of Kernels

Assume $K \in R^{n \times n}$ be positive semi-definite

$$K = V \Lambda V^{T} = [v^1; v^2; \ldots; v^n] \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} \begin{bmatrix} v^{1\prime} \\ \vdots \\ v^{n\prime} \end{bmatrix}$$

$$[\sqrt{\lambda_1} v^1; \ldots; \sqrt{\lambda_n} v^n]_{ij} = (x^i); (x^j)$$

where $: x^i \to (\sqrt{\lambda_1} v_i^1; \sqrt{\lambda_2} v_i^2; :::; \sqrt{\lambda_n} v_i^n)$

The use of kernels is a way of expanding linear discriminant hyperplanes to versatile nonlinear discriminant ones. Let a mapping : $X \to Z = \left(\sqrt{\lambda_1}\psi_1(x),\ldots,\sqrt{\lambda_d}\psi_d(x)\right)^T \equiv \Lambda^{1/2}\psi(x)$ where $\{\psi_k\}_{k=1}^{d}$ are linear independent and with unit L_2-length, and $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d \geq 0$, $d \leqq \infty$

Then the Mercer kernel $k(x,u) = \sum_{k=1}^{d} \lambda_k \psi_k(x) \psi_k(u)$ (a positive definite kernel)

2. Kernel Fisher Discriminant Analysis

- Map the data of input space R^n into some Hilbert space (call the feature space, and is usually of high dimension to allow flexible nonlinear discriminant functions), and perform the FDA in the feature space
- Let ψ be a non-linear mapping to some feature space F. To find the linear discriminant in F we need to maximize

$$J(w) = \frac{w^T S_B^{\psi} w}{w^T S_w^{\psi} w}$$

where now $w \in F$ and S_B^{ψ} and S_w^{ψ} are the corresponding matrices in F

i.e. $S_B^{\psi} = (m_1^{\psi} - m_2^{\psi})(m_1^{\psi} - m_2^{\psi})^T$ and

$$S_w^{\psi} = \sum_{i=1,2} \sum_{x \in \chi_i} (\psi(x) - m_i^{\psi})(\psi(x) - m_i^{\psi})^T$$

are the between and within class scatter matrices and m_i^{ψ} is defined by $m_i^{\psi} = \frac{1}{l_i} \sum_{j=1}^{l_i} \psi(x_j^i)$

- The kernel FDA finds the discriminant function *f(x)* of the form

$$f(x) = \sum_{k=1}^{d} w_k \sqrt{\lambda_k} \psi_k(x) + b \equiv w^T \Lambda^{1/2} \psi(x) + b$$

where w is the maximizing argument for

$$\arg\max_{w} \frac{w^T S_B^{\psi} w}{w^T S_w^{\psi} w}, \text{ (some form of regularization is necessary)}$$

and where

$$S_B^{\psi} = (m_1^{\psi} - m_2^{\psi})(m_1^{\psi} - m_2^{\psi})^T \text{ (between-class)}, \; m_i^{\psi} = \frac{1}{l_i} \sum_{j \in I_i} \Lambda^{1/2} \psi(x_j)$$

$$S_w^{\psi} = \sum_{j \in I} \Lambda \psi(x_j) \psi(x_j)^T - (l_1 m_1^{\psi} (m_1^{\psi})^T + l_2 m_2^{\psi} (m_2^{\psi})^T) \text{ (within-class)}$$

The constant b is so determined that the separating surface *f(z)*=0 passes through the mid point of group centroids in the feature space, i.e.,

$$b = -\frac{(m_1^{\psi} + m_2^{\psi})^T w_{KFDA}}{2}$$

- Then consider a regularized within-class covariance of the form *Sw+rW* for two purposes
  - One is to overcome the numerical problem cause by singular within-class covariance in a high-dimensional feature space
  - The other is to control the smoothness and the shape of the fitted separating surface by adding in a penalty matrix *W*

$$\arg\max_{w} \frac{w^T S_B^{\psi} w}{w^T (S_w^{\psi} + rW) w}, b = -\frac{(m_1^{\psi} + m_2^{\psi})^T w_{RKFDA}}{2}$$

3. Bayesian Kernel FDA

◆ Consider the coding scheme (see Anderson, 1984)

$$y_j^{(c)} = \begin{cases} -l_2 / l, j \in I_1 \\ -l_1 / l, j \in I_2 \end{cases}$$

The joint posterior density function of a given $y^{(c)}$ is

$$p(a \mid y^{(c)}) \propto \exp\left\{ -\frac{1}{2\sigma^2} \sum_{j=1}^{l} (y_j^{(c)} - f(x_j) + \bar{f})^2 - \frac{a^T \sum^{-1} a}{2\tau^2} \right\}$$

$$= \exp\left\{ -\frac{1}{2\sigma^2} \| y^{(c)} - (K - \frac{1}{l} 1 1^T K) a \|^2 - \frac{a^T \sum^{-1} a}{2\tau^2} \right\}$$

The Bayesian predictor for a is the posterior MLE:

$$\arg\max_{a} \| y^{(c)} - PKa \|^2 + r a^T A a, \quad \text{(regularization)}$$

where $r = \sigma^2 / \tau^2$, $A = \sum^{-1}$ *and* $P = (I_l - \frac{1}{l} 1 1^T)$. The solution is

$$a = \frac{l_1 l_2}{l} (KPK + rA)^{-1} (\bar{k}_1 + \bar{k}_2)$$

◆ Quadratic programming algorithm

The parameter values for a in the Bayesian kernel FDA can be obtained as the solution of the following quadratic programming problem, and vice versa

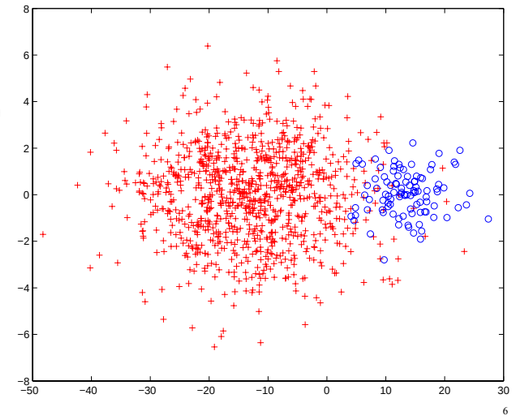
$$\min_{a \in R^l} \| PKa - y^{(c)} \|^2 + r a^T A a$$

◆ Thus, the discriminant function can be formulated as

$$f(x) = \sum_{j=1}^{l} a_j k(x_j, x) + b$$

4. Simulation setting

- Positive data :
  - X-axis: Mean= - 10 , Y-axis:Mean= 0
  - X-axis: STD= 10 , Y-axis:STD= 2
- Negative data :
  - X-axis: Mean= 10 , Y-axis:Mean= 0
  - X-axis: STD= 5 , Y-axis:STD= 1
- Number of Positive : Number of Negative = 10 : 1



◆ **Simulation Result（10 times）**

	KFDA with Linear Kernel			LibSVM with Linear Kernel			KFDA with RBF Kernel			LibSVM with RBF Kernel		
SimData	Recall	Precision	Accuracy	Recall	Precision	Accuracy	Recall	Precision	Accuracy	Recall	Precision	Accuracy
1	0.93	0.97895	0.955	0.98	0.80992	0.875	0.98	0.875	0.92	0.98	0.79675	0.865
2	0.95	1	0.975	0.98	0.90741	0.94	0.98	0.96078	0.97	1	0.84746	0.91
3	0.95	1	0.975	0.98	0.90741	0.94	0.98	0.96078	0.97	1	0.84746	0.91
4	0.92	1	0.96	0.98	0.875	0.92	0.97	0.92381	0.945	1	0.78125	0.86
5	0.94	1	0.97	0.99	0.90826	0.945	0.99	0.93396	0.96	0.99	0.81818	0.885
6	0.93	1	0.965	0.98	0.81667	0.88	0.98	0.92453	0.95	0.98	0.784	0.855
7	0.9	1	0.95	0.95	0.84821	0.89	0.95	0.86364	0.9	0.95	0.74219	0.81
8	0.91	0.97849	0.945	1	0.81301	0.885	0.99	0.88393	0.93	1	0.70423	0.79
9	0.95	1	0.975	0.98	0.82353	0.885	0.98	0.89091	0.93	0.99	0.76154	0.84
10	0.93	1	0.965	0.98	0.89908	0.935	0.96	0.92308	0.94	0.97	0.75194	0.825
Mean	**0.931**	**0.995744**	**0.9635**	**0.98**	**0.86085**	**0.9095**	**0.976**	**0.914042**	**0.9415**	**0.986**	**0.7835**	**0.855**
STD	**0.017288**	**0.008973**	**0.010814**	**0.012472**	**0.042988**	**0.028911**	**0.012649**	**0.034253**	**0.022367**	**0.016465**	**0.045988**	**0.040069**

◆ **Dataset**：*Pen-Based Recognition of Handwritten Digits*

- Source：E. Alpaydin, Fevzi. Alimoglu
- Number of Instances：pendigits.tra Training 7494
  pendigits.tes Testing 3498
- Number of Attributes：16 input+1 class attribute
- Class code：0,1,2,3,4,5,6,7,8,9
- Best classification accuracy reported in the past:97.8% (k-nn with k=3)
- Accuracy on the testing set with one vs. rest method in “Reducing multiclass to binary by coupling probability estimates”(Bianca Zadrozny, 2002) is 92.85%

◆ Use KFDA with one vs. rest voting method

- Tuning parameters：RBF kernel, penalty matrix A=I, lambda=50,r=50
- Multi-correct：3415
- Multi-error：83
- Accuracy：97.63 %