# 5.8 Minimum Squared Error Procedures

- Criterion function involves *all* of the samples, not just misclassified ones
- Previously we were interested in making all of the inner products $a^t y_i$ positive
- Now try to make $a^t y_i = b_i$ where $b_i$ are some arbitrarily specified positive constants
- 

# 5.8 Minimum Squared Error Procedures

- Thus replace the problem of solving a set of linear inequalities with more stringent but better understood problem of finding a solution to a set of linear equations

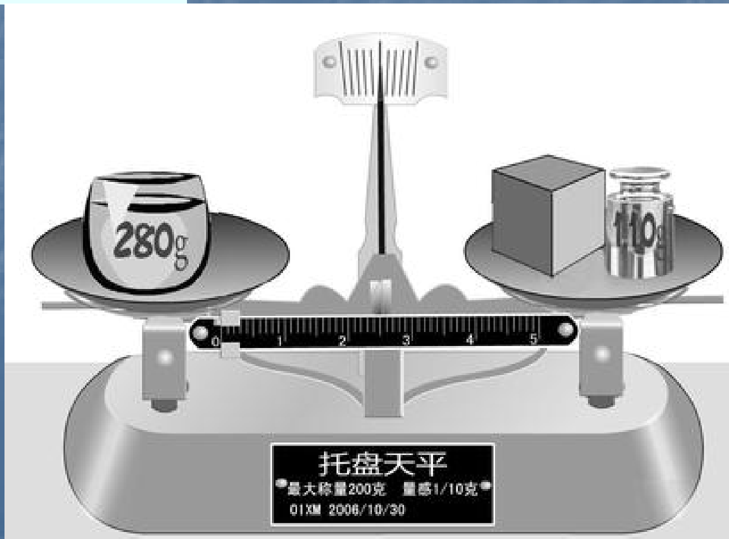
- Minimum Squared Error and Pseudoinverse

  For all the samples $y_1, y_2, \ldots, y_n$ we want a weight vector a so that $a^t y_i = b_i$ for some arbitrarily specified positive numbers. The matrix notation :

$$\begin{pmatrix} y_{10} & y_{11} & \cdots & y_{1d} \\ y_{20} & y_{21} & \cdots & y_{2d} \\ \cdots & \cdots & \cdots & \cdots \\ y_{n0} & y_{n1} & \cdots & y_{nd} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \cdots \\ a_d \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ b_n \end{pmatrix} \Leftrightarrow Ya = b$$

Error vector:

$$e = Ya - b$$



- Sum-of-squared-error criterion function:

$$J_s(a) = \|Ya - b\|^2 = \sum_{i=1}^{n} (a^t y_i - b_i)^2$$

- The gradient

$$\nabla J_s = \sum_{i=1}^{n} 2(a^t y_i - b_i) y_i = 2Y^t(Ya - b)$$

Set it to zero, we get $Y^tYa = Y^tb$

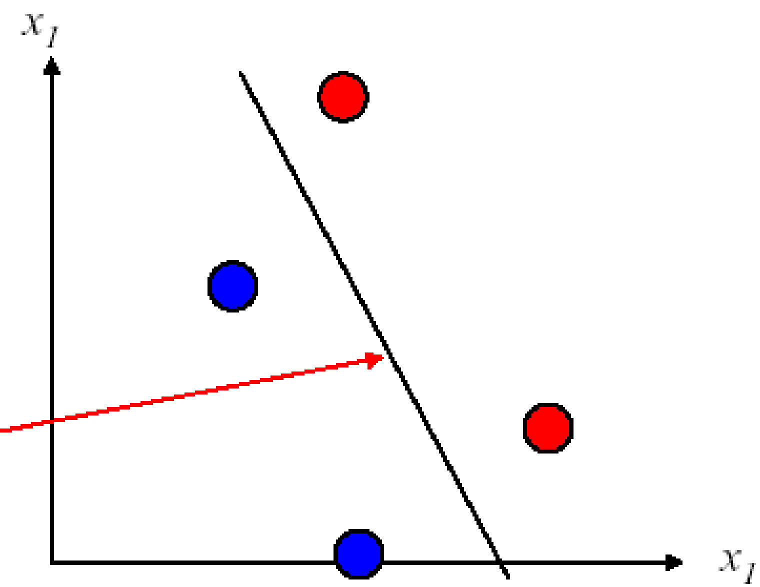
If $Y^tY$ is nonsingular, $a = (Y^tY)^{-1}Y^tb = Y^+b$

The d by n matrix $Y^+$ is call the pseudoinverse of Y.

- Remarks: For an arbitrarily fixed b, MSE solution may not be a separating vector.

# Example of Linear Classifier by Pseudoinverse

- $\omega_1$: $(1,2)^t$ and $(2,0)^t$
- $\omega_2$: $(3,1)^t$ and $(2,3)^t$




Sample Matrix (d = 1+2, n = 4)

$$a^t \begin{pmatrix} 1 \\ x_1 \\ x_2 \end{pmatrix} = 0$$

$$Y = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 2 & 0 \\ -1 & -3 & -1 \\ -1 & -2 & -3 \end{bmatrix}$$

Pseudo-inverse

$$Y^* = (Y^tY)^{-1}Y^t = \begin{bmatrix} 5/4 & 13/12 & 3/4 & 7/12 \\ -1/2 & -1/6 & -1/2 & -1/6 \\ 0 & -1/3 & 0 & -1/3 \end{bmatrix}$$

Assuming $b = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$

our solution is $a = Y^t b = \begin{bmatrix} 11/3 \\ -4/3 \\ -2/3 \end{bmatrix}$

# How to classify new samples (test samples)?

$$a.y > 0$$ → First class

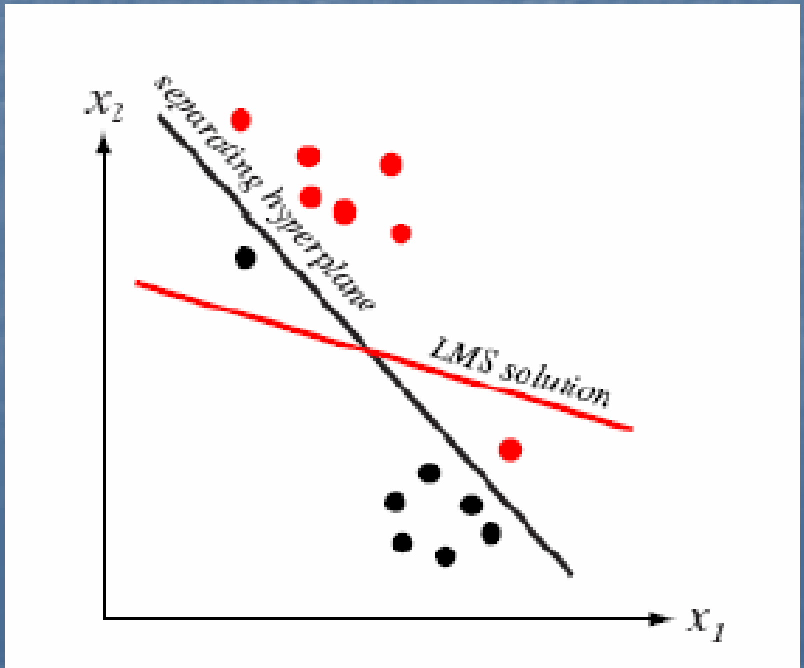
$$a.y < 0$$ → Second class

$y$ : new sample

- The Widrow-Hoff or LMS Procedure

(1) Iterative procedure: no matrix inverse

(2) Need not converge to a separating hyperplane even if there exist one



# 5.9 The Ho-Kashyap Procedure

- Take the criterion function as a function of two variables a and b:

$$J_s(\mathrm{a},\mathrm{b}) = \|Ya - b\|^2, \text{ where b} > 0$$

- If the training samples are linearly separable, then there should exist an $\hat{a}$ and $\hat{\mathrm{b}}$ such that: $Y\hat{a} = \hat{b} > 0$

If we knew such $\hat{\mathrm{b}}$ beforehand. We would get the separating vector $\hat{a}$ using the MSE procedure

$$\nabla_a J_s = 2Y^t(Ya - b)$$

$$\nabla_b J_s = -2(Ya - b)$$

$$a = Y^+ b$$

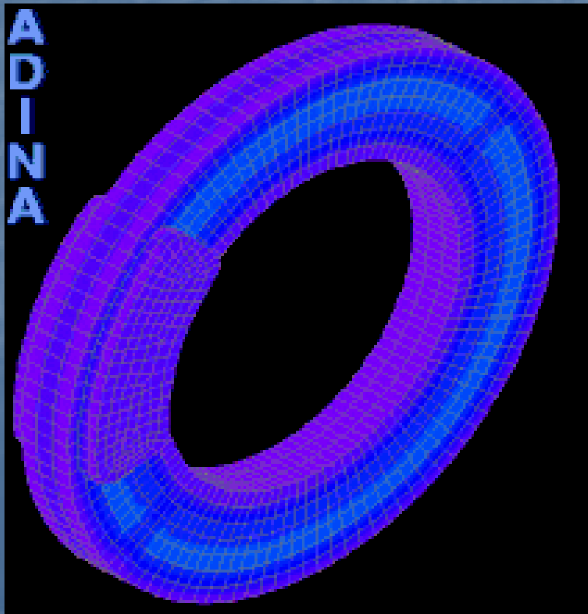
$$b(k+1) = b(k) - \eta \frac{1}{2}[\nabla_b J_s - |\nabla_b J_s|]$$

# Ho-Kashyap Procedure

$$b(1) > 0$$

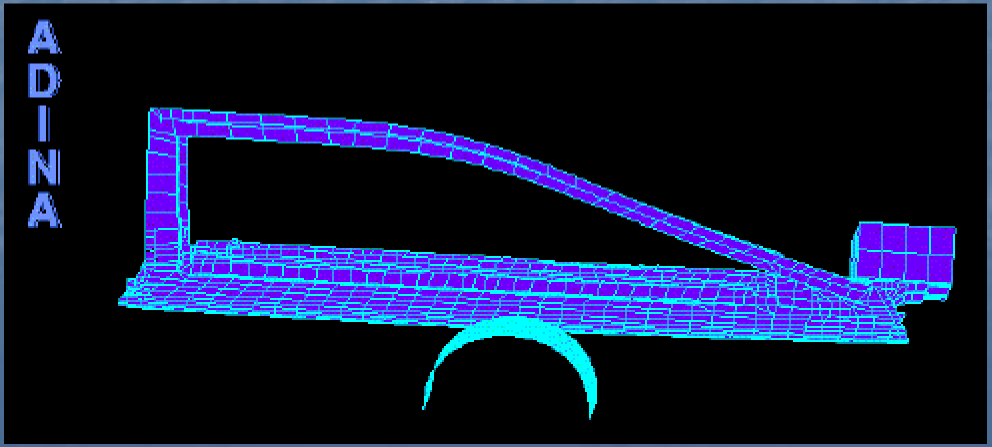
$$b(k+1) = a(k) + 2\eta(k)e+(k)$$

$$e+(k) = (e(k) + |e(k)|)/2$$



$$e(k) = Ya(k) - b(k)$$

$$a(k) = inv(Y'Y)Y'b(k)$$



- 5.12 Multicategory Generalizations


- Generalization for MSE Procedure

  consider multicategory case as a set of c two-class problem

$$a_i^t y = 1 \quad \text{for all } \mathrm{y} \in \mathrm{Y_i}$$

$$a_i^t y = 0 \quad \text{for all } \mathrm{y} \notin \mathrm{Y_i}$$

$$A = \begin{bmatrix} a_1 & a_2 & \dots & a_c \end{bmatrix} = \begin{bmatrix} a_{11} & a_{21} & \dots & a_{c1} \\ a_{12} & a_{22} & \dots & a_{c2} \\ \dots & \dots & \dots & \dots \\ a_{1d} & a_{2d} & \dots & a_{cd} \end{bmatrix}$$

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \dots \\ Y_c \end{bmatrix} = \begin{bmatrix} \mathrm{y}_{111} & \mathrm{y}_{112} & \cdots & \mathrm{y}_{11\mathrm{d}} \\ \mathrm{y}_{121} & \mathrm{y}_{122} & \cdots & \mathrm{y}_{12\mathrm{d}} \\ \dots & \dots & \dots & \dots \\ y_{211} & y_{112} & \cdots & y_{21d} \\ \mathrm{y}_{221} & \mathrm{y}_{122} & \cdots & \mathrm{y}_{22\mathrm{d}} \\ \dots & \dots & \dots & \dots \\ \mathrm{y}_{\mathrm{c}11} & \mathrm{y}_{\mathrm{c}12} & \cdots & \mathrm{y}_{\mathrm{c}1\mathrm{d}} \\ y_{c21} & y_{c22} & \cdots & y_{c2d} \end{bmatrix}$$


- Generalization for MSE Procedure

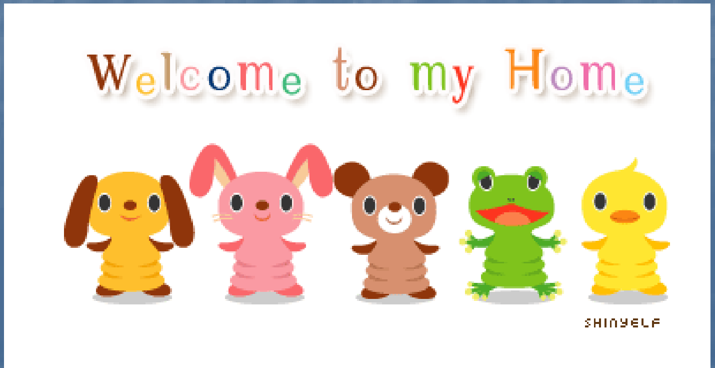
consider multicategory case as a set of c two-class problem

$$B = \begin{bmatrix} B_1 \\ B_2 \\ \dots \\ B_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 1 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 1 \\ \dots & \dots & \dots & \dots \end{bmatrix}$$

$$YA = B$$

$$A = Y^{+}B$$

$$= inv(Y'Y)Y'B$$

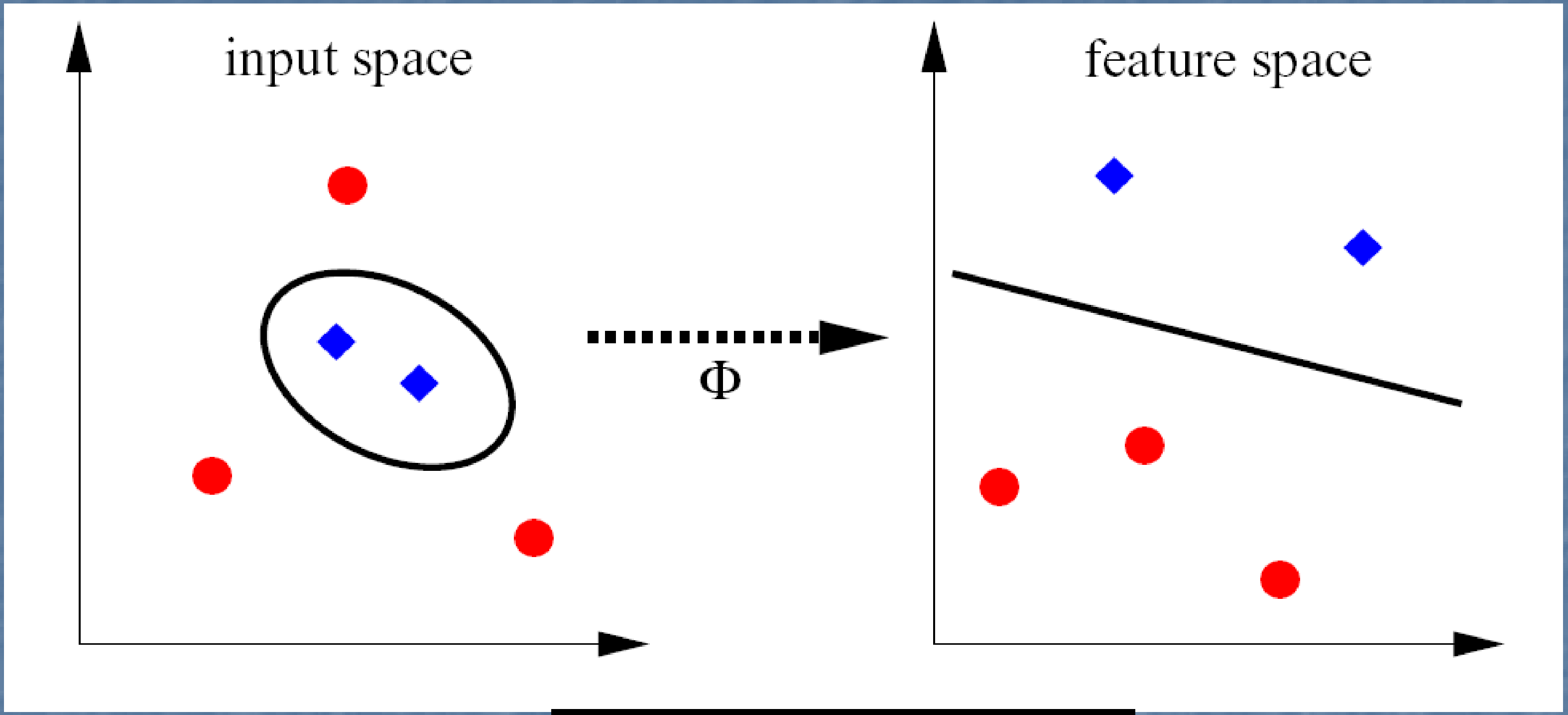


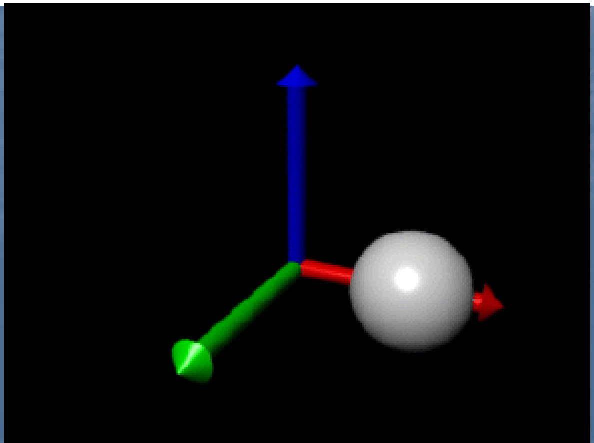
# Nonlinear Minimum Squared Error Procedures:

Just for your reference

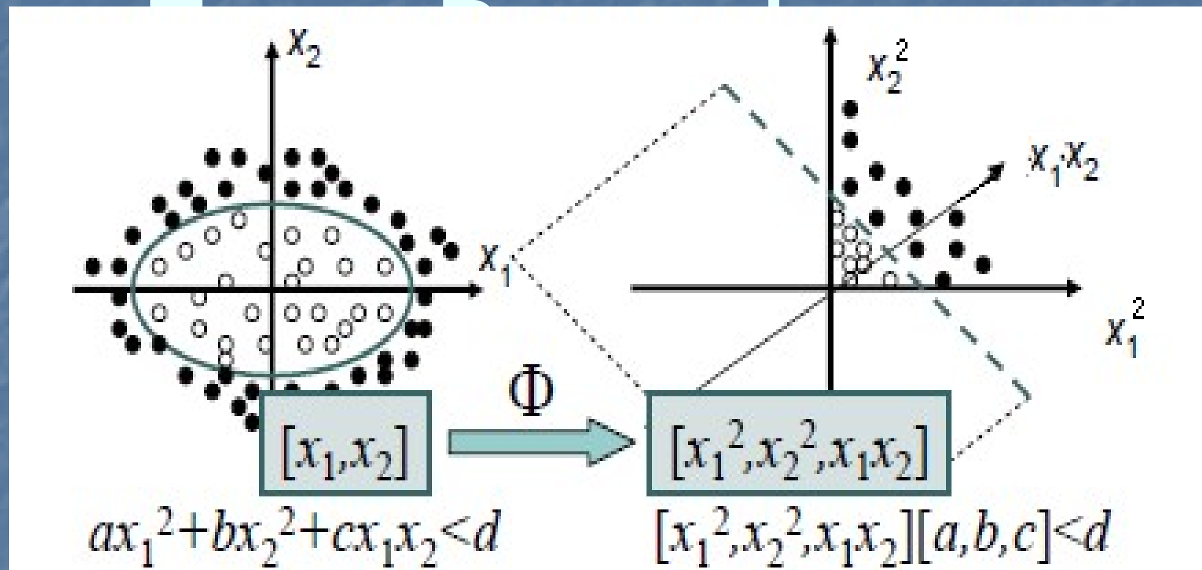

- Recently proposed new metho
- Extension of Minimum Squared Error Procedures
- Equivalent to the Minimum Squared Error Procedures in feature space

# Nonlinear transform

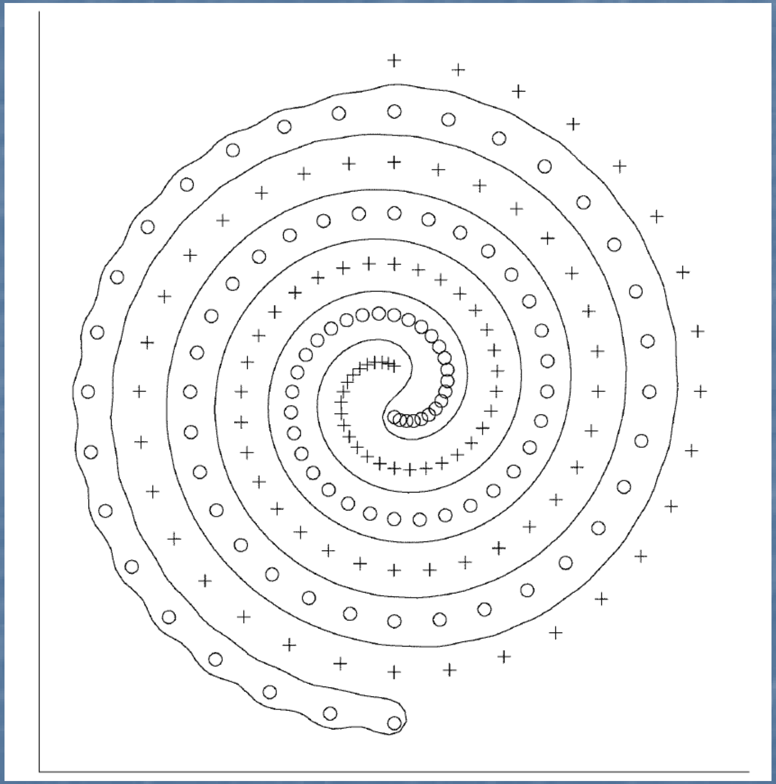


# Nonlinear Minimum Squared




# Nonlinear Minimum Squared Error Procedures



# Nonlinear Minimum Squared Error Procedures

Original Minimum Squared Error procedure in the original space:

$$\begin{pmatrix} y_{10} & y_{11} & \cdots & y_{1d} \\ y_{20} & y_{21} & \cdots & y_{2d} \\ \cdots & \cdots & \cdots & \cdots \\ y_{n0} & y_{n1} & \cdots & y_{nd} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \cdots \\ a_d \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ \cdots \\ -1 \end{pmatrix} \Leftrightarrow Ya = B$$

Minimum Squared Error procedure in the new space:

$$Z\beta = B \qquad Z = [Z_1 \ldots Z_n] \qquad Z_i = \varphi(Y_i)$$

# Nonlinear Minimum Squared Error Procedures: KMSE

Because of

$$\beta = \sum_{j=1,\ldots,n} \gamma_j \varphi(Y_j)$$

$$\varphi(Y_i)^T \varphi(Y_j) = k(Y_i, Y_j)$$

we have

$$K = \begin{pmatrix} k(Y_1,Y_1) & k(Y_1,Y_2) & \ldots & k(Y_1,Y_n) \\ k(Y_2,Y_1) & k(Y_2,Y_2) & \ldots & k(Y_2,Y_n) \\ \ldots & \ldots & \ldots & \ldots \\ k(Y_n,Y_1) & k(Y_n,Y_2) & \ldots & k(Y_n,Y_n) \end{pmatrix}$$

$$K\gamma = B$$

# Nonlinear Minimum Squared Error Procedures: KMSE

- Kernel functions:
- (1) $$k(Y_i,Y_j)=\exp(-\frac{\|Y_i-Y_j\|^2}{\sigma})$$

(2) $$k(Y_i,Y_j)=(Y_i^T Y_j+c)^d$$

# Nonlinear Minimum Squared Error Procedures: KMSE

- Training phase:
- Obtain $\gamma = K^{-1}B$

Testing phase ( $b$ is the output of testing sample $Y$ ):

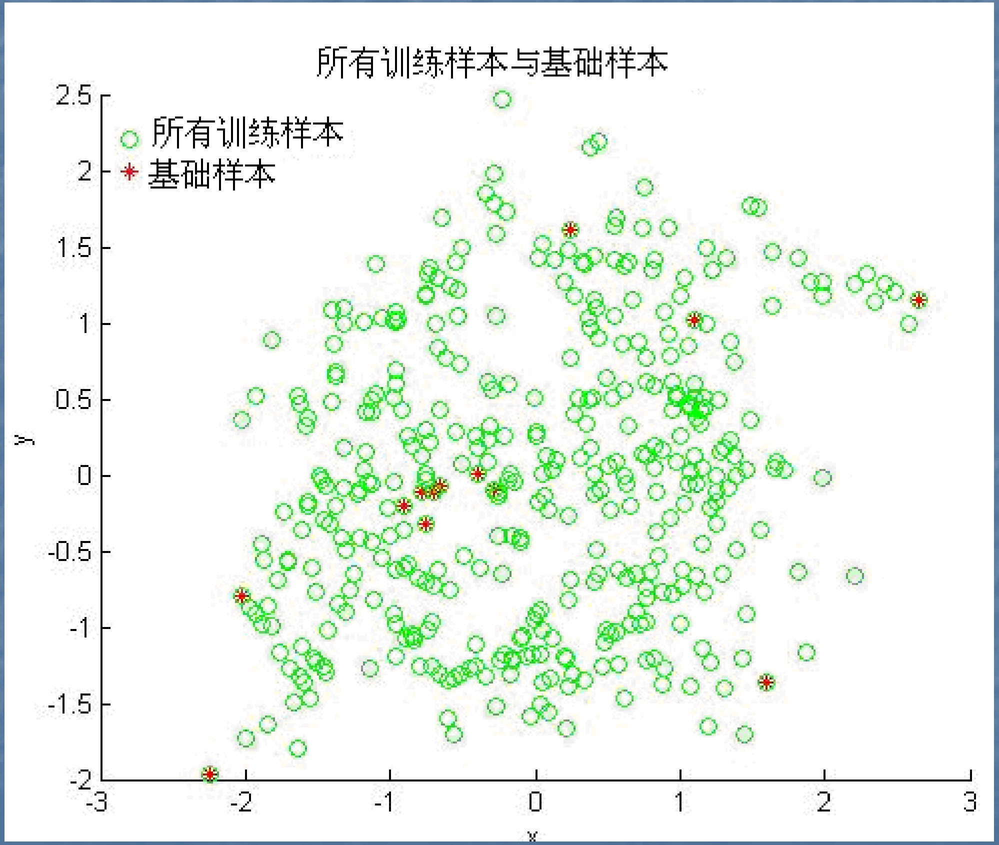
$$b = \sum_{i=1}^{n} \gamma_i k(Y_i, Y)$$

If $b$ is closer to 1 than -1, then the testing sample is classified into the firs otherwise, it is classified into the second class

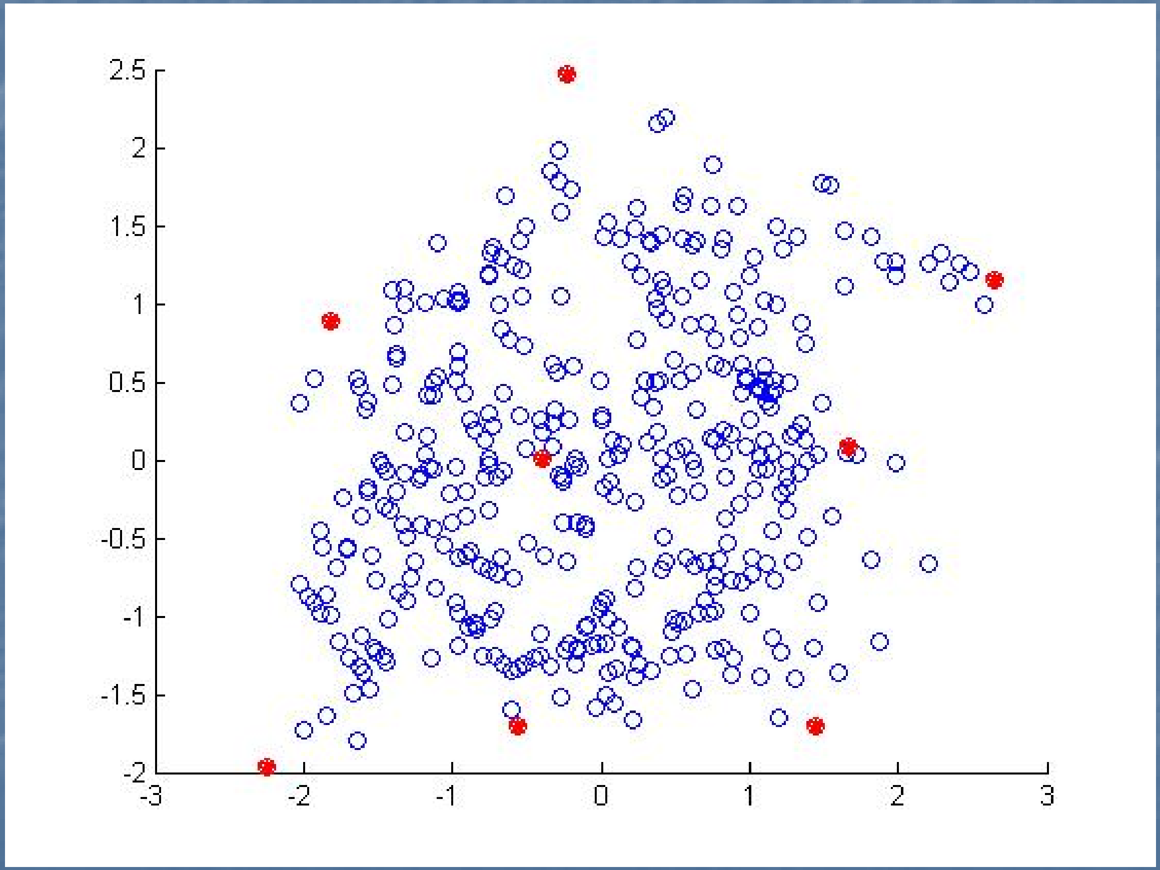
# Disadvantage of and improvement to KMSE

- The more the training samples, the higher the computational complexity !
- If $\beta = \sum_{j=1,\ldots,s} \gamma'_j \varphi(Y_j), s << n$
- Then $\boxed{b = \sum_{i=1}^{s} \gamma'_i k(Y_i, Y)}$ and the computational complexity will be greatly reduced.

# Disadvantage of and improvement to KMSE: one improvement



# Disadvantage of and improvement to KMSE: another improvement



# Nonlinear Minimum Squared Error Procedures : KMSE

- References


- Yong Xu, David Zhang, Zhong Jin, Miao Li, Jing-Yu Yang, A fast kernel-based nonlinear discriminant analysis for multi-class problems, Pattern Recognition, 2006, 39(6) : 1026-1033.
- Yong Xu, J.-Y. Yang, J.-F. Lu, An efficient kernel-based nonlinear regression method for two-class classification, Proceedings of 2005 International Conference on Machine Learning and Cybernetics, Guangzhou, China, August, 2005, pp.4442-4445.
- Yong Xu, David Zhang , Fengxi Song, Jing-Yu Yang, Zhong Jing , Miao Li, A method for speeding up feature extraction based on KPCA,Neurocomputing, 70, 1056-1061,2007
- 徐勇,张大鹏,杨健,模式识别中的核方法及其应用,北京:国防工业出版社(优秀图书二等奖),2010