Fisher's Linear Discriminent (DHS 3.8.2)

Reduce feature dimension N to 1 if possible. For a 3D problem, if 2D gives comparable results then make a 2D problem.

M^N cells, N=dimensions

N-dimensions -> 1 dimension

2-D case, 2-class case (See Fig. 4.27)

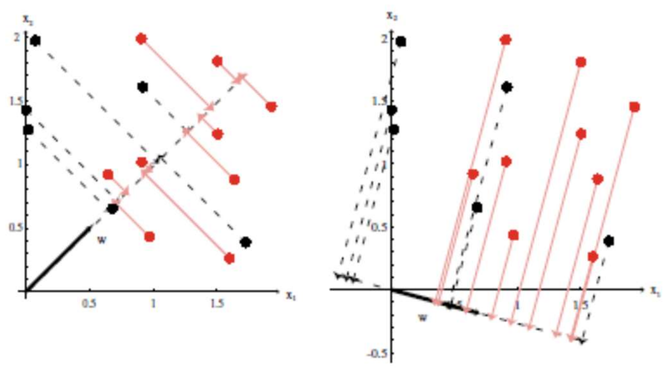

Figure 4.27: Projection of samples onto two different lines. The figure on the right shows greater separation between the red and black projected points.

Find 1-D line that will optimally separate prototypes in a 1-D problem. Prototypes are projections of original N-D prototypes onto the line.

2-class version

J N-D samples

x: $x_1, \ldots, x_j, \ldots. x_J$

J_1 samples in S_1

J_2 samples in S_2

$J=J_1+J_2$

Let $y_j=w^Tx_j$

where y_j=1-D (projected) prototype

x_j=original (N-D) prototype

if ||w||=1, there is no scale change

Objective: Find the direction of the line that gives best separation of prototypes.

Consider the ratio:

[Distance between class means]/[Some measure of standard deviation of each class]

Let m_i=class mean

$m_j=1/J_i \sum_{x_j \in S_i} x_j$

Sample mean of projected points:

$$\tilde{m}_i = \frac{1}{J_i} \sum_{y_j \in S_i} y_j = \frac{1}{J_i} \sum w^T x_j = w^T m_i$$

Distance between projected means:

$$\left|\tilde{m}_1 - \tilde{m}_2\right| = \left|w^T (m_1 - m_2)\right|$$

Define scatter measure

$$\tilde{S}_i^2 = \sum_{x_j \in S_i} (y_j - \tilde{m}_i)^2$$

Variance of projected prototypes can be estimated by:

$1/J(\tilde{S}_1^2 + \tilde{S}_2^2)$

$(\tilde{S}_1^2 + \tilde{S}_2^2)$ = within class scatter

Definition

Fisher's linear discriminant:

The linear function $w^T x$ for which the criterian function J(w)= $\frac{|\tilde{m}_1 - \tilde{m}_2|^2}{\tilde{S}_1^2 + \tilde{S}_2^2}$ is maximized.

In terms of S_B and S_W J(w) can be rewritten (without derivation)

$J(w)= [w^T S_B w]/[w^T S_w w]$

where $w=S_w^{-1}(m_1-m_2)$

See details in DHS 4.10

Iterative Optimization (DHS 10.8)

J samples, K clusters ~ $K^J/J!$ possible partitions

K=5, J=100 => 10^{67} possible partitions!

Given a partitioning, move a sample from one cluster to another. If this improves the value of $J_{extremum}$ accept it as a new partitioning. Otherwise, keep old partitioning. Iterate.

- Find local extremum of $J_{extremum}$
- Statistical techniques can find a global extremum.

For sum-of-squared errors criterion

$J_{ext}=J_e=\sum_{i=1}^{K} J_i$

where $J_i=\sum\|x_j-m_i\|^2$

$m_i=(1/j_i)\sum x_j$

want to minimize J_e

Suppose sample $\tilde{x}$, currently in z_i, is moved to cluster z_l.

For an index "l"

New mean m_l* becomes:

$m_l^* = [m_l j_l + \tilde{x}]/[j_l+1] = [m_l j_l+m_l+\tilde{x} -m_l]/ [j_l+1]$

$m_l^* = m_l + (\tilde{x} -m_l)/(j_l+1)$ adding one sample to z_l

New criterion function J_l*

$$J_l^* = J_l + \frac{j_l}{(j_l+1)}\|\tilde{x}-m_l\|^2$$

For an index "i"

Similarly, if $\tilde{x}$ is removed from class z_i, updating rules are:

$m_i^* = m_i - (\tilde{x} - m_i)/(j_i - 1)$

$\mathbf{J}_i^* = \mathbf{J}_i - [j_i / (j_i - 1)] \, ||\tilde{x} - m_i||^2$

$J_e = \sum_{n=1}^{K} J_n$

Transfer $\tilde{x}$ from z_i to z_l if decrease in J_i is larger than increase in J_l, that is if

$[j_i / (j_i - 1)] \, ||\tilde{x} - m_i||^2 > [j_l / (j_l + 1)] \, ||\tilde{x} - m_l||^2$

Then J_e decreases (➔ accept the new partitioning)

As J_i, J_l -> ∞, put $\tilde{x}$ in the same class as the closest (old) mean.

The greatest decrease in J_e is obtained by putting $\tilde{x}$ in the cluster for which $[j_l / (j_l + 1)] \, ||\tilde{x} - m_l||^2$ is minimum.

<u>Clustering Procedures – Basic Iterative Min. Squared Error Clustering</u>

1. Choose no. of clusters, K
2. Select an initial partition of J samples into clusters. Compute J_e and means m_1, …, m_k.
3. Select the next candidate sample $\tilde{x}$. Move $\tilde{x}$ to z_i.
4. Update J_e & m_i
5. Check if J_e reduces. Go to Step 3
6. If J_e has not changed in J attempts, stop.

Possible selection of initial conditions (initial clusters):

1. Use sample mean for K=1
2. Get K initial points (means) from K-1 points (clusters) by selecting sample farthest from K-1 points
3. Use Min. distance classifier to get initial clusters.

Next Topic: K-means clustering