Dimensionality Reduction – Fisher's Linear Discriminant

**Review**

\[
P(o_j | x) = \frac{p(x | o_j) p(o_j)}{p(x)} = \frac{\sum_{i} \lambda(x_i | o_j) p(x_i | x)}{g(x)} = \ln p(x | o_j) + \ln P(o_j)
\]

- **Maximum Posterior Probability**
  - For a given \(x\), if \(P(o_j | x) > P(o_k | x)\), then \(x\) belongs to class \(o_j\), otherwise, 2.

- **Maximum Likelihood**
  - If \(R(x_i | x) > R(x_j | x)\) then decide \(o_j\), that is
    \[
    \begin{align*}
    p(x_i | o_j) & > \lambda_j p(x_i) - \lambda_j p(x) \\
    p(x_i | o_k) & > \lambda_k p(x_i) - \lambda_k p(x)
    \end{align*}
    \]

- **Discriminant Function**
  - The classifier will assign a feature vector \(x\) to class \(o_j\) if
    \[
    g_j(x) = g_k(x)
    \]

- **Pattern Classification**
  - Statistical Approach
    - Supervised
      - Bayes decision rule
      - Parameter estimate (ML, BL)
    - Non-Parametric learning (kNN)
  - Non-Statistical Approach
    - Basic concepts
    - Winner-takes-all
    - Kohonen maps
    - Mean-shift

- **Dimensionality Reduction**
  - FL, PCA

- **Performance Evaluation**
  - ROC curve (TP, TN, FN, FP)
  - Cross validation

- **Classifier Fusion**
  - Majority voting

- **Stochastic Methods**
  - Local opt (GD)
  - Global opt (SA, GA)

- **Decision Tree**
  - Syntactic approach
  - NN (BP)

- **Support Vector Machine**

- **Deep Learning (DL)**
The Curse of Dimensionality – 1st Aspect

- The number of training samples
- What would the probability density function look like if the dimensionality is very high?
  - For a 7-dimensional space, where each variable could have 20 possible values, then the 7-d histogram contains $20^7$ cells. To distribute a training set of some reasonable size (1000) among these many cells is to leave virtually all the cells empty.

Curse of Dimensionality – 2nd Aspect

- Accuracy and overfitting
- In theory, the higher the dimensionality, the less the error, the better the performance. However, in realistic pattern recognition problems, the opposite is often true. Why?
  - The assumption that pdf behaves like Gaussian is only approximately true.
  - When increasing the dimensionality, we may be overfitting the training set.
  - Problem: excellent performance on the training set, poor performance on new data points which are in fact very close to the data within the training set.

Curse of Dimensionality – 3rd Aspect

- Computational complexity
Dimensionality Reduction

- Fisher’s linear discriminant
  - Best discriminating the data
- Principal component analysis (PCA)
  - Best representing the data

Fisher’s Linear Discriminant

- For two-class cases, projection of data from d-dimension onto a line
- Principle: We’d like to find vector \( \mathbf{w} \) (direction of the line) such that the projected data set can be best separated

\[
J(\mathbf{w}) = \frac{1}{n_1} \sum x_i^T \mathbf{w} - \mathbf{m}_1^T \mathbf{w} + \frac{1}{n_2} \sum x_i^T \mathbf{w} - \mathbf{m}_2^T \mathbf{w}
\]

Projected mean
Sample mean

Other Approaches?

- Solution 1: make the projected mean as apart as possible
- Solution 2?

\[
\chi^2 = \sum_{i=1}^{n_1} (x_i - \bar{x})' \mathbf{w} \sum_{i=1}^{n_2} (x_i - \bar{x})' \mathbf{w}
\]

\[
\chi^2 = \sum_{i=1}^{n_1} (x_i - \bar{x})' \mathbf{w} \sum_{i=1}^{n_2} (x_i - \bar{x})' \mathbf{w}
\]

Between-class scatter matrix \( \mathbf{S}_b = (\mathbf{m}_1 - \mathbf{m}_2)' \mathbf{S} (\mathbf{m}_1 - \mathbf{m}_2) \)

Within-class scatter matrix \( \mathbf{S}_w = \sum_{i=1}^{n_1} (x_i - \bar{x})' (x_i - \bar{x}) \)
The Generalized Rayleigh Quotient

\[ J(w) = \frac{w^T S_w w}{w^T S_w w} \]
\[ \frac{dJ(w)}{dw} = 2S_w w (w^T S_w w) - 2S_w w (w^T S_w w) = 0 \]

\[ S_w w = \frac{w^T S_w w}{w^T S_w w} S_w w \Rightarrow S_w^{-1} S_w w = \lambda w \]

\[ S_w w \text{ is always in the direction of } m_1 - m_2. \]

\[ w = S_w^{-1} (m_1 - m_2) \text{ Canonical variate} \]

Some Math Preliminaries

- **Positive definite**
  - A matrix \( S \) is positive definite if \( y^T S x > 0 \) for all \( R^d \) except \( 0 \)
  - \( x^T S x \) is called the quadratic form
  - The derivative of a quadratic form is particularly useful

\[ \frac{d}{dx} (x^T S x) = (S + S^T) x \]

- **Eigenvalue and eigenvector**
  - \( x \) is called the eigenvector of \( A \) iff \( x \) is not zero, and \( Ax = \lambda x \)
  - \( \lambda \) is the eigenvalue of \( x \)

* Multiple Discriminant Analysis

- For \( c \)-class problem, the projection is from \( d \)-dimensional space to a \((c-1)\)-dimensional space (assume \( d \geq c \))
- Sec. 3.8.3