INTRODUCTION TO
Machine Learning

ETHEM ALPAYDIN
© The MIT Press, 2004

alpaydin@boun.edu.tr
http://www.cmpe.boun.edu.tr/~ethem/i2ml
CHAPTER 5: Multivariate Methods
Multivariate Data

- Multiple measurements (sensors)
- \(d\) inputs/features/attributes: \(d\)-variate
- \(N\) instances/observations/examples

\[
X = \begin{bmatrix}
X_1^1 & X_2^1 & \cdots & X_d^1 \\
X_1^2 & X_2^2 & \cdots & X_d^2 \\
\vdots & \vdots & \ddots & \vdots \\
X_1^N & X_2^N & \cdots & X_d^N
\end{bmatrix}
\]
Multivariate Parameters

Mean: \( E[x] = \mu = [\mu_1, ..., \mu_d]^T \)

Covariance: \( \sigma_{ij} \equiv \text{Cov}(X_i, X_j) \)

Correlation: \( \text{Corr}(X_i, X_j) \equiv \rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j} \)

\[
\Sigma \equiv \text{Cov}(X) = E[(X - \mu)(X - \mu)^T] = \\
\begin{bmatrix}
\sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1d} \\
\sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2d} \\
\vdots & \vdots & \ddots & \vdots \\
\sigma_{d1} & \sigma_{d2} & \cdots & \sigma_d^2
\end{bmatrix}
\]
Parameter Estimation

Sample mean $\mathbf{m} : m_i = \frac{\sum_{t=1}^{N} x_i^t}{N}, i = 1, \ldots, d$

Covariance matrix $\mathbf{S} : s_{ij} = \frac{\sum_{t=1}^{N} (x_i^t - m_i)(x_j^t - m_j)}{N}$

Correlation matrix $\mathbf{R} : r_{ij} = \frac{s_{ij}}{s_i s_j}$
Estimation of Missing Values

- What to do if certain instances have missing attributes?
- Ignore those instances: not a good idea if the sample is small
- Use ‘missing’ as an attribute: may give information
- **Imputation**: Fill in the missing value
  - Mean imputation: Use the most likely value (e.g., mean)
  - Imputation by regression: Predict based on other attributes
\( \mathbf{x} \sim \mathcal{N}_d(\mu, \Sigma) \)

\[
p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp\left[- \frac{1}{2} (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu)\right]
\]
Multivariate Normal Distribution

- Mahalanobis distance: \((x - \mu)^T \Sigma^{-1} (x - \mu)\)
  measures the distance from \(x\) to \(\mu\) in terms of \(\Sigma\)
  (normalizes for difference in variances and correlations)
- Bivariate: \(d = 2\)
  \[
  \Sigma = \begin{bmatrix}
  \sigma_1^2 & \rho \sigma_1 \sigma_2 \\
  \rho \sigma_1 \sigma_2 & \sigma_2^2
  \end{bmatrix}
  \]

\[
p(x_1, x_2) = \frac{1}{2\pi \sigma_1 \sigma_2 \sqrt{1 - \rho^2}} \exp\left[-\frac{1}{2(1 - \rho^2)} \left(z_1^2 - 2\rho z_1 z_2 + z_2^2\right)\right]
\]

\[
z_i = (x_i - \mu_i) / \sigma_i
\]
Bivariate Normal

\[
\text{Cov}(x_1, x_2) = 0, \ Var(x_1) = \Var(x_2)
\]

\[
\text{Cov}(x_1, x_2) = 0, \ Var(x_1) > \Var(x_2)
\]

\[
\text{Cov}(x_1, x_2) > 0
\]

\[
\text{Cov}(x_1, x_2) < 0
\]
$\text{Cov}(x_1, x_2) = 0, \ Var(x_1) = \Var(x_2)$

$\text{Cov}(x_1, x_2) = 0, \ Var(x_1) > \Var(x_2)$

$\text{Cov}(x_1, x_2) > 0$

$\text{Cov}(x_1, x_2) < 0$
**Independent Inputs: Naive Bayes**

- If $x_i$ are independent, offdiagonals of $\Sigma$ are 0, Mahalanobis distance reduces to weighted (by $1/\sigma_i$) Euclidean distance:

  \[
p(x) = \prod_{i=1}^{d} p_i(x_i) = \frac{1}{(2\pi)^{d/2} \prod_{i=1}^{d} \sigma_i} \exp\left[-\frac{1}{2} \sum_{i=1}^{d} \left(\frac{x_i - \mu_i}{\sigma_i}\right)^2\right]
  \]

- If variances are also equal, reduces to Euclidean distance
Parametric Classification

- If $p(x \mid C_i) \sim \mathcal{N}(\mu_i, \Sigma_i)$
  
  $$p(x \mid C_i) = \frac{1}{(2\pi)^d/2 |\Sigma_i|^{1/2}} \exp \left[ -\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) \right]$$

- Discriminant functions are
  
  $$g_i(x) = \log p(x \mid C_i) + \log P(C_i)$$
  
  $$= -\frac{d}{2} \log 2\pi - \frac{1}{2} \log |\Sigma_i| - \frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) + \log P(C_i)$$
Estimation of Parameters

\[
\hat{P}(C_i) = \frac{\sum_t r_i^t}{N}
\]

\[
m_i = \frac{\sum_t r_i^t x^t}{\sum_t r_i^t}
\]

\[
S_i = \frac{\sum_t r_i^t (x^t - m_i)(x^t - m_i)^T}{\sum_t r_i^t}
\]

\[
g_i(x) = -\frac{1}{2} \log |S_i| - \frac{1}{2} (x - m_i)^T S_i^{-1} (x - m_i) + \log \hat{P}(C_i)
\]
Different $S_i$

- Quadratic discriminant

$$g_i(x) = -\frac{1}{2} \log |S_i| - \frac{1}{2} \left( x^T S_i^{-1} x - 2 x^T S_i^{-1} m_i + m_i^T S_i^{-1} m_i \right) + \log \hat{P}(C_i)$$

$$= x^T W_i x + w_i^T x + w_{i0}$$

where

$$W_i = -\frac{1}{2} S_i^{-1}$$

$$w_i = S_i^{-1} m_i$$

$$w_{i0} = -\frac{1}{2} m_i^T S_i^{-1} m_i - \frac{1}{2} \log |S_i| + \log \hat{P}(C_i)$$
likelihods

\[
P(x_{1}|C_1) = 0.5
\]

discriminant:

\[
P(C_1|x) = 0.5
\]

posterior for \( C_1 \)
Common Covariance Matrix $S$

- Shared common sample covariance $S$
  \[
  S = \sum_i \hat{P}(C_i) S_i
  \]

- Discriminant reduces to
  \[
  g_i(x) = -\frac{1}{2} (x - m_i)^T S^{-1} (x - m_i) + \log \hat{P}(C_i)
  \]

  which is a linear discriminant
  \[
  g_i(x) = w_i^T x + w_{i0}
  \]

  where
  \[
  w_i = S^{-1} m_i \quad w_{i0} = -\frac{1}{2} m_i^T S^{-1} m_i + \log \hat{P}(C_i)
  \]
Common Covariance Matrix $S$
Diagonal $S$

- When $x_j, j = 1, \ldots, d$, are independent, $\Sigma$ is diagonal
  
  $p(x|C_i) = \prod_j p(x_j|C_i)$  \hspace{1em} (Naive Bayes’ assumption)

  $$g_i(x) = -\frac{1}{2} \sum_{j=1}^{d} \left( \frac{x_j^t - m_{ij}}{s_j} \right)^2 + \log \hat{P}(C_i)$$

  Classify based on weighted Euclidean distance (in $s_j$ units) to the nearest mean
Diagonal $S$

variances may be different
Diagonal $S$, equal variances

- Nearest mean classifier: Classify based on Euclidean distance to the nearest mean

$$g_i(x) = -\frac{\|x - m_i\|^2}{2s^2} + \log \hat{P}(C_i)$$

$$= -\frac{1}{2s^2} \sum_{j=1}^{d} (x_j - m_{ij})^2 + \log \hat{P}(C_i)$$

- Each mean can be considered a prototype or template and this is template matching
Diagonal $S$, equal variances
## Model Selection

<table>
<thead>
<tr>
<th>Assumption</th>
<th>Covariance matrix</th>
<th>No of parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shared, Hyperspheric</td>
<td>$S_i = s^2 I$</td>
<td>1</td>
</tr>
<tr>
<td>Shared, Axis-aligned</td>
<td>$S_i = S$, with $s_{ij} = 0$</td>
<td>$d$</td>
</tr>
<tr>
<td>Shared, Hyperellipsoidal</td>
<td>$S_i = S$</td>
<td>$d(d+1)/2$</td>
</tr>
<tr>
<td>Different, Hyperellipsoidal</td>
<td>$S_i$</td>
<td>$K d(d+1)/2$</td>
</tr>
</tbody>
</table>

- As we increase complexity (less restricted $S$), bias decreases and variance increases
- Assume simple models (allow some bias) to control variance (regularization)
Discrete Features

- **Binary features**: $p_{ij} \equiv p(x_j = 1 \mid C_i)$
  
  if $x_j$ are independent (Naive Bayes’)

  $$p(x \mid C_i) = \prod_{j=1}^{d} p_{ij}^{x_j} (1 - p_{ij})^{(1-x_j)}$$

  the discriminant is **linear**

  $$g_i(x) = \log p(x \mid C_i) + \log P(C_i)$$

  $$= \sum_j [x_j \log p_{ij} + (1 - x_j) \log (1 - p_{ij})] + \log P(C_i)$$

  Estimated parameters  

  $$\hat{p}_{ij} = \frac{\sum_t x_j^t r_i^t}{\sum_t r_i^t}$$
Discrete Features

Multinomial (1-of-\(n_j\)) features: \(x_j \in \{v_1, v_2, ..., v_{n_j}\}\)

\[
p_{ijk} = p(z_{jk} = 1 \mid C_i) = p(x_j = v_k \mid C_i)
\]

if \(x_j\) are independent

\[
p(x \mid C_i) = \prod_{j=1}^{d} \prod_{k=1}^{n_j} p_{ijk}^{z_{jk}}
\]

\[
g_i(x) = \sum_j \sum_k z_{jk} \log p_{ijk} + \log P(C_i)
\]

\[
\hat{p}_{ijk} = \frac{\sum_t z_{jk}^t r_i^t}{\sum_t r_i^t}
\]
Multivariate Regression

\[
\mathbf{r}^t = g(\mathbf{x}^t | w_0, w_1, ..., w_d) + \varepsilon
\]

- Multivariate linear model
  \[
  w_0 + w_1 x_1^t + w_2 x_2^t + \cdots + w_d x_d^t
  \]
  \[
  E(w_0, w_1, ..., w_d | \mathbf{x}) = \frac{1}{2} \sum_t [\mathbf{r}^t - w_0 - w_1 x_1^t - \cdots - w_d x_d^t]^2
  \]

- Multivariate polynomial model:
  Define new higher-order variables
  \[
  Z_1 = x_1, Z_2 = x_2, Z_3 = x_1^2, Z_4 = x_2^2, Z_5 = x_1 x_2
  \]
  and use the linear model in this new \( \mathbf{z} \) space
  (basis functions, kernel trick, SVM: Chapter 10)