

Pattern Recognition III

Statistical Classifiers

Michal Haindl

Faculty of Information Technology, KTI
Czech Technical University in Prague
Institute of Information Theory and Automation
Czech Academy of Sciences
Prague, Czechia



<http://ro.utia.cz/>



MI/MIE-ROZ 2023-2024

© Haindl

MI-ROZ - 03

1/48

Outline

- 1 Multivariate Normal Density
- 2 Lossy Bayesian Classifiers
- 3 Lossless Bayesian Classifiers
 - Linear Transformation of the Bayesian Classifier
- 4 Sequential Classifiers
- 5 Fisher Classifier
- 6 Deterministic Classifiers
- 7 Nonparametric Classifiers
 - Parzen Classifiers
 - Nearest Neighbour Classifier
 - Bark Recognition Example
- 8 Contextual Classification



© Haindl

MI-ROZ - 03

2/48

Multivariate Normal Density

$$p(X|\omega_i) = \frac{1}{(2\pi)^{\frac{l}{2}} |\Sigma_i|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(X - \mu_i)^T \Sigma_i^{-1}(X - \mu_i)\right\}$$

$$\mu_i = E\{X\} \quad X \in S_i$$

$$\Sigma_i = E\{(X - \mu_i)(X - \mu_i)^T\}$$

- Σ symmetric
- Σ positive semidefinite - restriction to PD for simplicity
- $d_M(X, \mu_i) = (X - \mu_i)^T \Sigma_i^{-1}(X - \mu_i)$ Mahalanobis distance
- loci of const. p. density points - hyperellipsoids
 $d_M(X, \mu_i) = \text{const.}$
- principal axes of the hyperellipsoids - eigenvectors of Σ_i



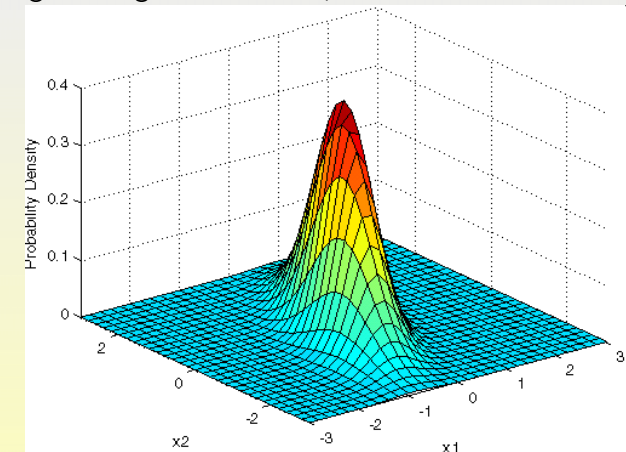
© Haindl

MI-ROZ - 03

4/48

Multivariate Normal Density 2

- axes lengths - eigenvalues of Σ_i



- the hyperellipsoid volume for a given $d_M(X, \mu_i) = d^2$,
 V_l l -dim unit hypersphere volume

$$V = V_l |\Sigma_i|^{\frac{1}{2}} d^l$$



© Haindl

MI-ROZ - 03

5/48

Multivariate Normal Density 2

- axes lengths - eigenvalues of Σ_i
- the hyperellipsoid volume for a given $d_M(X, \mu_i) = d^2$,
 V_l l - dim unit hypersphere volume

$$V = V_l |\Sigma_i|^{\frac{1}{2}} d^l$$

$$V_l = \begin{cases} \frac{\pi^{\frac{l}{2}}}{(\frac{l}{2})!} & l \text{ even} \\ \frac{2^l \pi^{(l-1)/2} (\frac{l-1}{2})!}{l!} & l \text{ odd} \end{cases}$$

- discriminant function

$$g_i(X) = -\frac{1}{2}(X - \mu_i)^T \Sigma_i^{-1} (X - \mu_i) - \frac{l}{2} \log(2\pi) - \frac{1}{2} \log |\Sigma_i| + \log P(\omega_i)$$

$$n(*) = l(l+1)$$

$$n(+) = (l-1)(l+3)$$



Multivariate Normal Density 2

- axes lengths - eigenvalues of Σ_i
- the hyperellipsoid volume for a given $d_M(X, \mu_i) = d^2$,
 V_l l - dim unit hypersphere volume

$$V = V_l |\Sigma_i|^{\frac{1}{2}} d^l$$

$$V_l = \begin{cases} \frac{\pi^{\frac{l}{2}}}{(\frac{l}{2})!} & l \text{ even} \\ \frac{2^l \pi^{(l-1)/2} (\frac{l-1}{2})!}{l!} & l \text{ odd} \end{cases}$$

- discriminant function

$$g_i(X) = -\frac{1}{2}(X - \mu_i)^T \Sigma_i^{-1} (X - \mu_i) - \frac{l}{2} \log(2\pi) - \frac{1}{2} \log |\Sigma_i| + \log P(\omega_i)$$

$$n(*) = l(l+1)$$

$$n(+) = (l-1)(l+3)$$

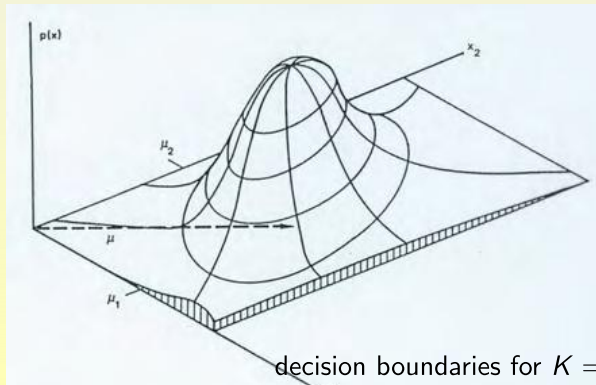


Lossy Bayesian Classifiers

- $P(\omega_i) = \frac{1}{K} \quad \forall i$ - maximum likelihood classifier

$$g_i(X) = -X^T \Sigma_i^{-1} X + 2(\Sigma_i^{-1} \mu_i)^T X - \mu_i^T \Sigma_i^{-1} \mu_i - \log |\Sigma_i|$$

decision surface - hyperquadrics (hyperplanes, hyperspheres, hyperellipsoids, hyperparaboloids, hyperhyperparaboloids)



decision boundaries for $K = 2, l = 2$



Euclidean (minimum-distance) Classifier

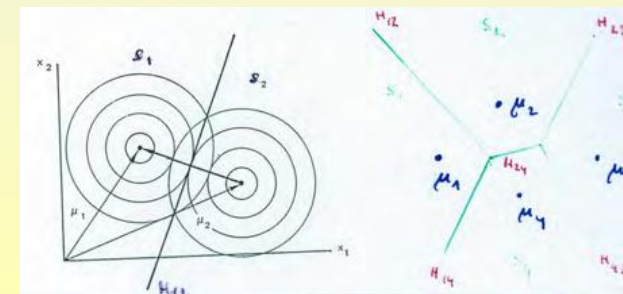
- $\Sigma_i = \sigma^2 I$
the samples create equal size hyperspherical clusters, linear classifier

$$g_i(X) = -\frac{1}{2\sigma^2} [X^T X - 2\mu_i^T X + \mu_i^T \mu_i] + \log P(\omega_i)$$

$$g_i(X) = \frac{1}{\sigma^2} \mu_i^T X - \frac{1}{2\sigma^2} \mu_i^T \mu_i + \log P(\omega_i)$$

$$n(*) = l$$

$$n(+) = l$$



Euclidean Classifier 2

$$g_i(X) = g_j(X)$$

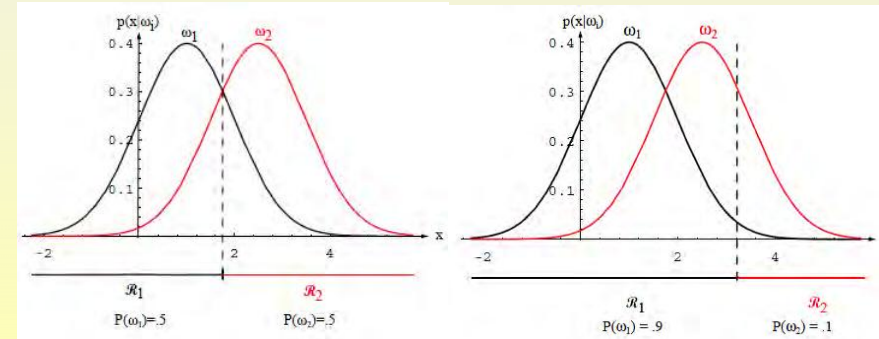


Euclidean Classifier 2

decision surface - hyperplane through point X_0 , orthogonal to $\mu_i - \mu_j$

$$(\mu_i - \mu_j)^T (X - X_0) = 0$$

$$X_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{\sigma^2}{\|\mu_i - \mu_j\|^2} \log \frac{P(\omega_i)}{P(\omega_j)} (\mu_i - \mu_j)$$



Euclidean Classifier 2

- $\Sigma_i = I \wedge \mu_i^T \mu_i = \text{const.} \forall i$ - correlation classifier

$$g_i(X) = 2\mu_i^T X + 2 \log P(\omega_i)$$

decision surface - hyperplane through point X_0 , orthogonal to $\mu_i - \mu_j$

$$(\mu_i - \mu_j)^T (X - X_0) = 0$$

$$X_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{1}{\|\mu_i - \mu_j\|^2} \log \frac{P(\omega_i)}{P(\omega_j)} (\mu_i - \mu_j)$$



Minimum-Mahalanobis Distance Classifier

- $\Sigma_i = \Sigma$ - the samples create equal size and shape hyperellipsoids, linear classifier

$$g_i(X) = -\frac{1}{2}(X - \mu_i)^T \Sigma^{-1} (X - \mu_i) + \log P(\omega_i)$$

$$g_i(X) = (\Sigma^{-1} \mu_i)^T X - \frac{1}{2}(\mu_i^T \Sigma^{-1} \mu_i + \log P(\omega_i))$$

$$\text{if } P(\omega_i) = P(\omega_j) \forall i, j \Rightarrow g_i(X) = -d_M(X, \mu_i)$$

$$n(*) = I$$

$$n(+) = I$$

decision surface - hyperplane through point X_0 but not orthogonal to $\mu_i - \mu_j$

$$(\Sigma^{-1}(\mu_i - \mu_j))^T (X - X_0) = 0$$

$$X_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{\log \frac{P(\omega_i)}{P(\omega_j)}}{(\mu_i - \mu_j)^T \Sigma^{-1} (\mu_i - \mu_j)} (\mu_i - \mu_j)$$



Minimum-Mahalanobis Distance Classifier

- $\Sigma_i = \Sigma$ - the samples create equal size and shape hyperellipsoids, linear classifier

$$g_i(X) = -\frac{1}{2}(X - \mu_i)^T \Sigma^{-1}(X - \mu_i) + \log P(\omega_i)$$

$$g_i(X) = (\Sigma^{-1} \mu_i)^T X - \frac{1}{2}(\mu_i^T \Sigma^{-1} \mu_i + \log P(\omega_i))$$

$$\text{if } P(\omega_i) = P(\omega_j) \quad \forall i, j \Rightarrow g_i(X) = -d_M(X, \mu_i)$$

$$n(*) = l \quad n(+) = l$$

decision surface - hyperplane through point X_0 but not orthogonal to $\mu_i - \mu_j$

$$(\Sigma^{-1}(\mu_i - \mu_j))^T (X - X_0) = 0$$

$$X_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{\log \frac{P(\omega_i)}{P(\omega_j)}}{(\mu_i - \mu_j)^T \Sigma^{-1}(\mu_i - \mu_j)}(\mu_i - \mu_j)$$



Linearized Bayesian Classifier

- linearized Bayesian classifier

$$g_i(X) = 2\mu_i^T \Sigma_i^{-1} X - \log |\Sigma_i| + 2 \log P(\omega_i)$$

$$n(*) = l \quad n(+) = l$$

Haindl, M., Linearised Bayesian classifier, Proceedings of International Symposium Progress in Data Analysis, ISPRS Wuhan 1990



Normalised Euclidean Classifier

- $\Sigma_i = \text{diag}[\sigma_1^2, \dots, \sigma_l^2] \quad \forall i$ - normalised Euclidean classifier
uncorrelated features with different dispersions

$$g_i(X) = -\sum_{j=1}^l \frac{(x_j - \mu_{j,i})^2}{\sigma_{j,i}^2} - \sum_{j=1}^l \log \sigma_{j,i}^2 + 2 \log P(\omega_i)$$

$$n(*) = 2l \quad n(+) = 2l - 1$$



Naive Bayesian Classifier

Assumption: conditional independence between features (seldom true)

$$P(\omega_i|X) = \frac{p(\omega_i)p(X|\omega_i)}{p(X)} = \frac{p(X, \omega_i)}{p(X)}$$



Naive Bayesian Classifier

Assumption: conditional independence between features (seldom true)

$$P(\omega_i|X) = \frac{p(\omega_i)p(X|\omega_i)}{p(X)} = \frac{p(X, \omega_i)}{p(X)}$$

chain rule:

$$p(X, \omega_i) = p(x_1 | x_2, \dots, x_l, \omega_i) p(x_2 | x_3, \dots, x_l, \omega_i) \dots p(x_l | \omega_i) p(\omega_i)$$



Naive Bayesian Classifier

Assumption: conditional independence between features (seldom true)

$$P(\omega_i|X) = \frac{p(\omega_i)p(X|\omega_i)}{p(X)} = \frac{p(X, \omega_i)}{p(X)}$$

$$p(x_i | x_j, \omega_i) = p(x_i | \omega_i) \quad \forall j \neq i$$

$$p(X, \omega_i) = p(x_1 | \omega_i) p(x_2 | \omega_i) \dots p(x_l | \omega_i) p(\omega_i)$$



Naive Bayesian Classifier

Assumption: conditional independence between features (seldom true)

$$P(\omega_i|x_1, \dots, x_l) = \frac{p(\omega_i) \prod_{j=1}^l p(x_j|\omega_i)}{p(X)}$$

$$g_i(X) = p(\omega_i) \prod_{j=1}^l p(x_j|\omega_i)$$

$$g_i(X) = p(\omega_i) \prod_{j=1}^l \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left\{-\frac{(x_j - \mu_{i,j})^2}{2\sigma_i^2}\right\} \quad X \in \mathcal{N}$$

($X \in \mathcal{N} \rightarrow$ normalised Euclidean classifier)



Naive Bayesian Classifier

Assumption: conditional independence between features (seldom true)

$$P(\omega_i|x_1, \dots, x_l) = \frac{p(\omega_i) \prod_{j=1}^l p(x_j|\omega_i)}{p(X)}$$

for large number of probabilities

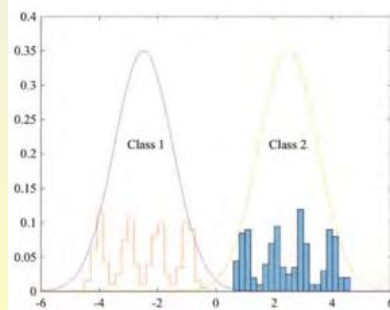
$$g_i(X) = \log p(\omega_i) + \sum_{j=1}^l \log p(x_j|\omega_i)$$

$$g_i(X) = \log p(\omega_i) - \sum_{j=1}^l \left\{ \frac{1}{2} \log\{2\pi\sigma_i^2\} + \left\{ \frac{(x_j - \mu_{i,j})^2}{2\sigma_i^2} \right\} \right\} \quad X \in \mathcal{N}$$



Naive Bayesian Classifier

- ⊕ simple learning (each feature 1D pdf independent) for high dimensional data
- ⊕ low number of required parameters (reduces the memory)
- ⊕ classifies well for poorly fitting feature models



- ⊖ oversimplified classifier for good feature models
- ⊖ poor performance if conditional independence assumption is not valid



Semi-Naive Bayesian Classifier I

Assumption: conditional independence between features except one dependent feature for every feature

$$p(\omega_i | x_1, \dots, x_I) = \frac{p(\omega_i) \prod_{j=1}^I p(x_j | \omega_i, x_k)}{p(X)}$$

where x_k is a parent feature for x_j

- ⊕ simple, efficient
- ⊕ improvement for not independent features



Semi-Naive Bayesian Classifier II

Super-parent (SPODE) semi-naive Bayesian classifier

Assumption: conditional independence between features except all are dependent on the same feature x_s

$$p(\omega_i | x_1, \dots, x_I) = \frac{p(\omega_i) \prod_{j=1}^I p(x_j | \omega_i, x_s)}{p(X)}$$

where x_s is a super-parent feature

- ⊕ simple, efficient
- ⊕ improvement for not independent features



Semi-Naive Bayesian Classifier III

SNBC classifier

Assumption: conditional independence between features, dependent features are removed using filter or wrapper FS

$$p(\omega_i | x_1, \dots, x_I) = \frac{p(\omega_i) \prod_{j=1}^I p(x_j | \omega_i)}{p(X)}$$

- ⊕ simple, efficient
- ⊕ improvement for not independent features



Lossless Bayesian Classifiers

- $\Sigma_i \neq \Sigma_j$ for $i \neq j$ quadratic classifier

$$g_i(X) = -X^T \Sigma_i^{-1} X + 2(\Sigma_i^{-1} \mu_i)^T X - \mu_i^T \Sigma_i^{-1} \mu_i - \log |\Sigma_i| + 2 \log P(\omega_i)$$

decision surface - hyperquadrics (hyperplanes, hyperspheres, hyperellipsoids, hyperparaboloids, hyperhyperparaboloids)

$$n(*) = l(l+1)$$

$$n(+) = (l-1)(l+3)$$

- Bayesian classifier with Cholesky factorization (or $U^T D U$)

$$\Sigma_i = L_i L_i^T$$

$$n(*) = \frac{l(l+3)}{2}$$

$$n(+) = \frac{(l-1)(l+2)}{2}$$



Lossless Bayesian Classifiers

- $\Sigma_i \neq \Sigma_j$ for $i \neq j$ quadratic classifier

$$g_i(X) = -X^T \Sigma_i^{-1} X + 2(\Sigma_i^{-1} \mu_i)^T X - \mu_i^T \Sigma_i^{-1} \mu_i - \log |\Sigma_i| + 2 \log P(\omega_i)$$

decision surface - hyperquadrics (hyperplanes, hyperspheres, hyperellipsoids, hyperparaboloids, hyperhyperparaboloids)

$$n(*) = l(l+1)$$

$$n(+) = (l-1)(l+3)$$

- Bayesian classifier with Cholesky factorization (or $U^T D U$)

$$\Sigma_i = L_i L_i^T$$



$$n(*) = \frac{l(l+3)}{2}$$

$$n(+) = \frac{(l-1)(l+2)}{2}$$



Bayesian Classifier with Cholesky Factorization

- recursive Bayesian classifier (UDU Banachiewicz factorization)
speed up for large l

$$\Sigma_i = U_i^T \text{diag}[\sigma_{1i}^2, \dots, \sigma_{li}^2] U_i$$

U_i orthogonal i.e. $U_i^T U_i = I$ denote $Y_i = U_i(X - \mu_i)$

$$d_M(X, \mu_i) = \sum_{j=1}^l \frac{y_j^2}{\sigma_{j,i}^2}$$

$$d_{M,k}(X, \mu_i) = d_{M,k-1}(X, \mu_i) + \frac{y_k^2}{\sigma_{k,i}^2}$$

X cannot belong to ω_i if for some $1 \leq k \leq l$

$$d_{M,k}(X, \mu_i) \geq d_{M,l}(X, \mu_j) - C_{i,j}$$



Interval Estimation of the Mahalanobis Distance

- $\lambda_{i,j}$ eigenvalues of Σ_i^{-1}

$$\lambda_{i,\min} |Y_i|^2 \leq d_M(X, \mu_i) \leq \lambda_{i,\max} |Y_i|^2$$

for non overlapping interval estimation some classes can be eliminated



Linear Transformation of the Bayesian Classifier

$$\tilde{X} = QX + R$$

$Q \quad I \times I$ regular, $R \quad I \times 1$

e.g.



Linear Transformation of the Bayesian Classifier

$$\tilde{X} = QX + R$$

$Q \quad I \times I$ regular, $R \quad I \times 1$

e.g.



Linear Transformation of the Bayesian Classifier

$$\tilde{X} = QX + R$$

$Q \quad I \times I$ regular, $R \quad I \times 1$

e.g.



Alot database



Linear Transformation of the Bayesian Classifier

$$\tilde{X} = QX + R$$

$Q \quad I \times I$ regular, $R \quad I \times 1$

e.g.

- Karhunen - Loeve tr. $\tilde{X} = QX$
- variable brightness & spectrum of illumination sources
- linear contrast modification (enhancement)
- data normalization
- feature extraction (Tasseled cap tr., etc.)

$$g_i(X) = X^T A_i X + B_i X + C_i$$

$$\tilde{g}_i(X) = \tilde{X}^T \tilde{A}_i \tilde{X} + \tilde{B}_i \tilde{X} + \tilde{C}_i$$



Linear Transformation of the Bayesian Classifier 2

T:
 $S_i = \tilde{S}_i \quad \forall i \text{ iff } \exists K, L, M \text{ matrices independent of } \omega_i \text{ such as:}$

$$\begin{aligned} A_i &= Q^T \tilde{A}_i Q + M \\ B_i &= (\tilde{B}_i + 2R^T \tilde{A}_i)Q + L \\ C_i &= \tilde{C}_i + (R^T \tilde{A}_i + \tilde{B}_i)R + K \end{aligned}$$

Bayesian classifier

$$\begin{aligned} A_i &= -\Sigma_i^{-1} \\ B_i &= 2\mu_i^T \Sigma_i^{-1} \\ C_i &= -\mu_i^T \Sigma_i^{-1} \mu_i - \ln |\Sigma_i| + 2 \ln P(\omega_i) \end{aligned}$$



Linear Transformation of the Bayesian Classifier 2

T:
 $S_i = \tilde{S}_i \quad \forall i \text{ iff } \exists K, L, M \text{ matrices independent of } \omega_i \text{ such as:}$

$$\begin{aligned} A_i &= Q^T \tilde{A}_i Q + M \\ B_i &= (\tilde{B}_i + 2R^T \tilde{A}_i)Q + L \\ C_i &= \tilde{C}_i + (R^T \tilde{A}_i + \tilde{B}_i)R + K \end{aligned}$$

Bayesian classifier

$$\begin{aligned} A_i &= -\Sigma_i^{-1} \\ B_i &= 2\mu_i^T \Sigma_i^{-1} \\ C_i &= -\mu_i^T \Sigma_i^{-1} \mu_i - \ln |\Sigma_i| + 2 \ln P(\omega_i) \end{aligned}$$



Linear Transformation of the Bayesian Classifier 3

$$\begin{aligned} \tilde{\mu}_i &= E\{\tilde{X}\} = Q\mu_i + R \\ \tilde{\Sigma}_i &= Q\Sigma_i Q^T \\ |\tilde{\Sigma}_i| &= |Q|^2 |\Sigma_i| \\ (\tilde{X} - \tilde{\mu}_i)^T \tilde{\Sigma}_i^{-1} (\tilde{X} - \tilde{\mu}_i) &= (X - \mu_i)^T \Sigma_i^{-1} (X - \mu_i) \\ \tilde{p}(\tilde{X} | \omega_i) &= \frac{1}{||Q||} p(Q^{-1}(\tilde{X} - R) | \omega_i) \\ X \in \mathcal{N} &\Rightarrow \tilde{X} \in \mathcal{N} \end{aligned}$$



Linear Transformation of the Bayesian Classifier 4

maximum likelihood classifier $P(\omega_i) = \frac{1}{K}$

$$\begin{aligned} A_i &= -\Sigma_i^{-1} \\ B_i &= 2\mu_i^T \Sigma_i^{-1} \\ C_i &= -\mu_i^T \Sigma_i^{-1} \mu_i - \ln |\Sigma_i| \end{aligned}$$

linear Bayesian classifier $\Sigma_i = \Sigma$

$$\begin{aligned} A_i &= 0 \\ B_i &= 2\mu_i^T \Sigma^{-1} \\ C_i &= -\mu_i^T \Sigma^{-1} \mu_i - \ln |\Sigma_i| + 2 \ln P(\omega_i) \end{aligned}$$



Linear Transformation of the Bayesian Classifier 5

Euclidean classifier $\Sigma_i = I$

$$A_i = 0$$

$$B_i = 2\mu_i^T$$

$$C_i = -\mu_i^T \mu_i + 2 \ln P(\omega_i)$$

correlation classifier $\Sigma_i = I \wedge \mu_i^T \mu_i = k$

$$A_i = 0$$

$$B_i = 2\mu_i^T$$

$$C_i = 2 \ln P(\omega_i)$$



Linear Transformation of the Bayesian Classifier 6

linearised Bayesian classifier

$$A_i = 0$$

$$B_i = 2\mu_i^T \Sigma_i^{-1}$$

$$C_i = -\mu_i^T \Sigma_i^{-1} \mu_i - \ln |\Sigma_i| + 2 \ln P(\omega_i)$$

normalized Euclidean classifier $\Sigma_i = \text{diag}[\dots]$

$$A_i = -\Sigma_i^{-1}$$

$$B_i = 2\mu_i^T \Sigma_i^{-1}$$

$$C_i = -\mu_i^T \Sigma_i^{-1} \mu_i - \ln |\Sigma_i| + 2 \ln P(\omega_i)$$



Bayesian Classifier Equivalence

T:

- Bayesian classifier in $S \equiv$ Bayesian classifier in $\tilde{S}$
- ML classifier in $S \equiv$ ML classifier in $\tilde{S}$
- linear Bayesian classifier in $S \equiv$ linear Bayesian classifier in $\tilde{S}$ ($\Sigma_i = \Sigma$)
- Euclidean classifier ($\Sigma_i = I$)
 - $Q Q^T = I$ (orthogonal) $\Rightarrow$ Euclidean classifier in $S \equiv$ Euclidean classifier in $\tilde{S}$
 - $Q Q^T \neq I \Rightarrow$ Euclidean classifier in $S \equiv$ linear Bayesian classifier with $\tilde{\Sigma}_i = Q Q^T$ in $\tilde{S}$



Bayesian Classifier Equivalence

T:

- Bayesian classifier in $S \equiv$ Bayesian classifier in $\tilde{S}$
- ML classifier in $S \equiv$ ML classifier in $\tilde{S}$
- linear Bayesian classifier in $S \equiv$ linear Bayesian classifier in $\tilde{S}$ ($\Sigma_i = \Sigma$)
- Euclidean classifier ($\Sigma_i = I$)
 - $Q Q^T = I$ (orthogonal) $\Rightarrow$ Euclidean classifier in $S \equiv$ Euclidean classifier in $\tilde{S}$
 - $Q Q^T \neq I \Rightarrow$ Euclidean classifier in $S \equiv$ linear Bayesian classifier with $\tilde{\Sigma}_i = Q Q^T$ in $\tilde{S}$



Bayesian Classifier Equivalence

T:

- Bayesian classifier in $S \equiv$ Bayesian classifier in $\tilde{S}$
- ML classifier in $S \equiv$ ML classifier in $\tilde{S}$
- linear Bayesian classifier in $S \equiv$ linear Bayesian classifier in $\tilde{S}$ ($\Sigma_i = \Sigma$)
- Euclidean classifier ($\Sigma_i = I$)
 - $Q Q^T = I$ (orthogonal) $\Rightarrow$
Euclidean classifier in $S \equiv$ Euclidean classifier in $\tilde{S}$
 - $Q Q^T \neq I \Rightarrow$
Euclidean classifier in $S \equiv$ linear Bayesian classifier with $\tilde{\Sigma}_i = Q Q^T$ in $\tilde{S}$



Bayesian Classifier Equivalence

T:

- Bayesian classifier in $S \equiv$ Bayesian classifier in $\tilde{S}$
- ML classifier in $S \equiv$ ML classifier in $\tilde{S}$
- linear Bayesian classifier in $S \equiv$ linear Bayesian classifier in $\tilde{S}$ ($\Sigma_i = \Sigma$)
- Euclidean classifier ($\Sigma_i = I$)
 - $Q Q^T = I$ (orthogonal) $\Rightarrow$
Euclidean classifier in $S \equiv$ Euclidean classifier in $\tilde{S}$
 - $Q Q^T \neq I \Rightarrow$
Euclidean classifier in $S \equiv$ linear Bayesian classifier with $\tilde{\Sigma}_i = Q Q^T$ in $\tilde{S}$



Bayesian Classifier Equivalence 2

- correlation classifier ($\Sigma_i = I \wedge \mu_i^T \mu_i = k$)
 - $Q Q^T = I \wedge R = 0 \Rightarrow$
correlation classifier in $S \equiv$ correlation classifier in $\tilde{S}$
 - $Q Q^T = I \wedge R \neq 0 \Rightarrow$
correlation classifier in $S \equiv$

$$\begin{aligned}\tilde{A}_i &= 0 \\ \tilde{B}_i &= 2\tilde{\mu}_i^T \\ \tilde{C}_i &= -2\tilde{\mu}_i^T R + 2 \ln P(\omega_i)\end{aligned}$$



Bayesian Classifier Equivalence 3

- correlation class. constant average class energy not preserved
 - $Q Q^T \neq I \Rightarrow$
correlation classifier in $S \equiv$

$$\begin{aligned}\tilde{A}_i &= 0 \\ \tilde{B}_i &= 2\tilde{\mu}_i^T Q^{-T} Q^{-1} \\ \tilde{C}_i &= -2\tilde{\mu}_i^T Q^{-T} Q^{-1} R + 2 \ln P(\omega_i)\end{aligned}$$

- normalized Euclidean classifier ($\Sigma_i = \text{diag}[\dots]$)
 - $Q = \text{diag}[\dots] \Rightarrow$
normalized Euclidean classifier in $S \equiv$ normalized Euclidean classifier in $\tilde{S}$
 - $Q \neq \text{diag}[\dots] \Rightarrow$
normalized Euclidean classifier in $S \equiv$ Bayesian classifier in $\tilde{S}$



Bayesian Classifier Equivalence 3

- correlation class. constant average class energy not preserved
 - $Q Q^T \neq I \Rightarrow$
correlation classifier in $S \equiv$

$$\begin{aligned}\tilde{A}_i &= 0 \\ \tilde{B}_i &= 2\tilde{\mu}_i^T Q^{-T} Q^{-1} \\ \tilde{C}_i &= -2\tilde{\mu}_i^T Q^{-T} Q^{-1} R + 2 \ln P(\omega_i)\end{aligned}$$

- normalized Euclidean classifier ($\Sigma_i = \text{diag}[\dots]$)
 - $Q = \text{diag}[\dots] \Rightarrow$
normalized Euclidean classifier in $S \equiv$ normalized Euclidean classifier in $\tilde{S}$
 - $Q \neq \text{diag}[\dots] \Rightarrow$
normalized Euclidean classifier in $S \equiv$ Bayesian classifier in $\tilde{S}$



Sequential Classifiers

- let m consecutive $X_1, \dots, X_m \in S_j$ $K = 2$
- X_i statistically independent

likelihood ratio classifier: assign X to ω_1 if

$$\ln \frac{p(X_1, \dots, X_m | \omega_1)}{p(X_1, \dots, X_m | \omega_2)} = \sum_{j=1}^m \ln \frac{p(X_i | \omega_1)}{p(X_i | \omega_2)} > t$$

$$f(X) = \ln \frac{p(X | \omega_1)}{p(X | \omega_2)}$$

$$\tilde{f}(X) = \frac{1}{m} \sum_{j=1}^m f(X_j)$$

$$E\{\tilde{f}(X) | \omega_i\} = E\{f(X) | \omega_i\}$$



Sequential Classifiers 2

$$\text{var}\{\tilde{f}(X) | \omega_i\} = \frac{1}{m} \text{var}\{f(X) | \omega_i\}$$

$$\oplus m \rightarrow \infty \Rightarrow \text{var}\{\tilde{f}(X) | \omega_i\} \rightarrow 0 \quad \tilde{f} \rightarrow \mathcal{N}$$



Sequential Classifiers 2

$$\text{var}\{\tilde{f}(X) | \omega_i\} = \frac{1}{m} \text{var}\{f(X) | \omega_i\}$$

$$\oplus m \rightarrow \infty \Rightarrow \text{var}\{\tilde{f}(X) | \omega_i\} \rightarrow 0 \quad \tilde{f} \rightarrow \mathcal{N}$$

required property for $\hat{f}()$ (estimate):

$$E\{\hat{f}(X) | \omega_1\} \geq 0$$

$$E\{\hat{f}(X) | \omega_2\} \leq 0$$

is satisfied for any distribution of X :

$$\bullet \hat{f}(X) = (\mu_1 - \mu_2)^T \Sigma^{-1} X + \frac{1}{2} (\mu_2^T \Sigma^{-1} \mu_2 - \mu_1^T \Sigma^{-1} \mu_1)$$

Σ PD

$$\bullet \hat{f}(X) = \frac{1}{2} (X - \mu_2)^T \Sigma_2^{-1} (X - \mu_2) - \frac{1}{2} (X - \mu_1)^T \Sigma_1^{-1} (X - \mu_1) - \frac{1}{2} \ln \frac{|\Sigma_1|}{|\Sigma_2|}$$



Fisher Classifier

- ⊙ $K = 2$, $\hat{\mu}_1, \hat{\mu}_2, \hat{\Sigma}_1, \hat{\Sigma}_2$ linear classifier
- finding a threshold to separate both classes as much as possible

$$m_0 = W^T(\hat{\mu}_2 - \hat{\mu}_1)$$

$|m_0|$ class separation measure constant but selectable

$$m_1 = W^T(\hat{\Sigma}_2 + \hat{\Sigma}_1)W$$

average sample S_1, S_2 variance in the W direction



Fisher Classifier

- ⊙ $K = 2$, $\hat{\mu}_1, \hat{\mu}_2, \hat{\Sigma}_1, \hat{\Sigma}_2$ linear classifier

$$m_1 = W^T(\hat{\Sigma}_2 + \hat{\Sigma}_1)W$$

find W such as $m_1(W^*) = \min\{m_1(W)\} \wedge m_0 = \text{const.}$

$$\begin{aligned} \frac{\partial}{\partial W}(m_1 - \lambda m_0) &= 2(\hat{\Sigma}_2 + \hat{\Sigma}_1)W - \lambda(\hat{\mu}_2 - \hat{\mu}_1) = 0 \\ W &= \frac{\lambda}{2}(\hat{\Sigma}_2 + \hat{\Sigma}_1)^{-1}(\hat{\mu}_2 - \hat{\mu}_1) \end{aligned}$$



Fisher Classifier

- ⊙ $K = 2$, $\hat{\mu}_1, \hat{\mu}_2, \hat{\Sigma}_1, \hat{\Sigma}_2$ linear classifier

if

$$m_0 = (\hat{\mu}_2 - \hat{\mu}_1)^T(\hat{\Sigma}_2 + \hat{\Sigma}_1)^{-1}(\hat{\mu}_2 - \hat{\mu}_1)$$

$$W = (\hat{\Sigma}_2 + \hat{\Sigma}_1)^{-1}(\hat{\mu}_2 - \hat{\mu}_1)$$



Fisher Classifier

- ⊙ $K = 2$, $\hat{\mu}_1, \hat{\mu}_2, \hat{\Sigma}_1, \hat{\Sigma}_2$ linear classifier

assign X to ω_2 if $g(X) > t$

$$g(X) = (\hat{\mu}_2 - \hat{\mu}_1)(\hat{\Sigma}_2 + \hat{\Sigma}_1)^{-1}X$$

⊖ no general rule for the threshold t



Fisher Classifier

⊙ $K = 2$, $\hat{\mu}_1, \hat{\mu}_2, \hat{\Sigma}_1, \hat{\Sigma}_2$ linear classifier

assign X to ω_2 if $g(X) > t$

$$g(X) = (\hat{\mu}_2 - \hat{\mu}_1)(\hat{\Sigma}_2 + \hat{\Sigma}_1)^{-1}X$$

if $\hat{\Sigma}_i = \hat{\Sigma}$, $m_0 = 2(\hat{\mu}_2 - \hat{\mu}_1)^T(\hat{\Sigma}_2 + \hat{\Sigma}_1)^{-1}(\hat{\mu}_2 - \hat{\mu}_1)$

$$g(X) = (\hat{\mu}_2 - \hat{\mu}_1)\hat{\Sigma}^{-1}X$$

⊕ optimal minimum Mahalanobis distance classifier



Least Mean-Square Error Learning

$$G(X) = [g_1(X), \dots, g_K(X)]^T = W^T Y$$

$$W = [W_1, \dots, W_K] \quad (\eta + 1) \times K$$

$$\begin{aligned} \tilde{J} &= \min_W \left\{ \frac{1}{n} \sum_{i=1}^K \sum_{X \in \omega_i} \|W^T Y - col_i I\|^2 \right\} \\ &= \min_W \left\{ \sum_{i=1}^K \frac{n_i}{n} \frac{1}{n_i} \sum_{X \in \omega_i} \|W^T Y - col_i I\|^2 \right\} \end{aligned}$$

$$\lim_{n \rightarrow \infty} \tilde{J} = J$$



Least Mean-Square Error Learning 2

$$\begin{aligned} J &= \sum_{i=1}^K P(\omega_i) E_i \left\{ \|W^T Y - col_i I\|^2 \right\} \\ &= \sum_{i=1}^K P(\omega_i) \int \|W^T Y - col_i I\|^2 p(X | \omega_i) dX \end{aligned}$$

$$\nabla_W J = 2 \sum_{i=1}^K P(\omega_i) E_i \left\{ Y Y^T \right\} W^T - 2 \sum_{i=1}^K P(\omega_i) E_i \left\{ Y \right\} (col_i I)^T = 0$$



Least Mean-Square Error Learning 3

$$\tilde{\Sigma} = \sum_{i=1}^K P(\omega_i) E_i \left\{ Y Y^T \right\}$$

$$\tilde{\mu} = [\tilde{\mu}_1, \dots, \tilde{\mu}_K] = [P(\omega_1)\mu_1^Y, \dots, P(\omega_K)\mu_K^Y]$$

$$\tilde{\Sigma} W = \tilde{\mu}$$

$$W^* = \tilde{\Sigma}^{-1} \tilde{\mu}$$

$$J(W^*) = 1 + \text{tr}\{\tilde{\mu}^T \tilde{\Sigma}^{-1} \tilde{\mu}\}$$

assign X to ω_i if $g_i(X) = \max_j \{g_j(X)\}$

$$g_j(X) = Y^T \tilde{\Sigma}^{-1} \tilde{\mu}_j$$

⊕ DF is the best MMSE approximation of the Bayes DF



Linear Regression Classifier

find a linear transformation $\mathcal{R}^I \rightarrow \mathcal{R}^K$ such as transformed vectors from S_i are closed to each other

$$Y = BX + C$$

$$\vec{u}_1 = [1, 0, \dots, 0]^T, \dots, \vec{u}_K = [0, 0, \dots, 1]^T$$

$$\begin{aligned} J &= \sum_{i=1}^n (\vec{u}_i - \hat{Y}_i)^T (\vec{u}_i - \hat{Y}_i) \\ &= \sum_{i=1}^n (\vec{u}_i - C - BX_i)^T (\vec{u}_i - C - BX_i) \end{aligned}$$

$$\frac{\partial J}{\partial C} = 2 \sum_{i=1}^n (Y_i - C - BX_i) = 0$$



Linear Regression Classifier 2

$$C = \frac{1}{n} \sum_{i=1}^n Y_i - B \frac{1}{n} \sum_{i=1}^n X_i = \hat{\mu}_Y - B \hat{\mu}_X$$

$$\hat{\Sigma}_{YX} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{\mu}_Y)(X_i - \hat{\mu}_X)^T$$

$$\frac{\partial J}{\partial B} = 0 \quad \rightarrow \quad B = \hat{\Sigma}_{YX} \hat{\Sigma}_{XX}^{-1}$$

$$\hat{Y} = \hat{\Sigma}_{YX} \hat{\Sigma}_{XX}^{-1} (X - \hat{\mu}_X) + \hat{\mu}_Y \quad (1)$$

- ① transform $X \rightarrow \hat{Y}$ using linear regression function (1)
- ② $g_i(\hat{Y}) = -(\hat{Y} - \vec{u}_i)^T (\hat{Y} - \vec{u}_i)$
- ③ assign X to ω_i if $g_i(\hat{Y}) = \max_j \{g_j(\hat{Y})\}$



Linear Regression Classifier 2

$$C = \frac{1}{n} \sum_{i=1}^n Y_i - B \frac{1}{n} \sum_{i=1}^n X_i = \hat{\mu}_Y - B \hat{\mu}_X$$

$$\hat{\Sigma}_{YX} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{\mu}_Y)(X_i - \hat{\mu}_X)^T$$

$$\frac{\partial J}{\partial B} = 0 \quad \rightarrow \quad B = \hat{\Sigma}_{YX} \hat{\Sigma}_{XX}^{-1}$$

$$\hat{Y} = \hat{\Sigma}_{YX} \hat{\Sigma}_{XX}^{-1} (X - \hat{\mu}_X) + \hat{\mu}_Y \quad (1)$$

- ① transform $X \rightarrow \hat{Y}$ using linear regression function (1)
- ② $g_i(\hat{Y}) = -(\hat{Y} - \vec{u}_i)^T (\hat{Y} - \vec{u}_i)$
- ③ assign X to ω_i if $g_i(\hat{Y}) = \max_j \{g_j(\hat{Y})\}$



Quadratic Classifiers

$$\odot \quad X \notin \mathcal{N}(\mu_i, \Sigma_i)$$

$$g_i(X) = X^T A_i X + B_i X + C_i$$

$$\ominus \quad \text{too complex} \quad \min_{A_i, B_i} J, \quad \text{unknown optimal design}$$



Deterministic Classifiers

Geometric classifier (linear, $K = 2$)
 Ho - Kashyap iterative learning algorithm



Linear Discriminant Functions

⊙ no pdf assumption

$$g_i(X) = B_i X + C_i \quad i = 1, \dots, K$$



Linear Discriminant Functions

⊙ no pdf assumption

$$g_i(X) = B_i X + C_i \quad i = 1, \dots, K$$

S_i convex polyhedron $K - 1$ polygonal faces (max.)

- K : $\omega_i \times \neg \omega_i \quad i = 1, \dots, K$
- $\frac{K(K-1)}{2}$: $\omega_i \times \omega_j \quad \forall i, j \in \{1, \dots, K\}$
- 1 : $g_*(X) = \arg \max_j \{g_j(X)\}$



Linear Discriminant Functions

⊙ no pdf assumption

$$g_i(X) = B_i X + C_i \quad i = 1, \dots, K$$

$h_{ij} = \{X : (B_i - B_j) X = C_j - C_i\}$ decision surface - hyperplane

$X_1, X_2 \in h_{ij}, \vec{n}$ unit normal vector to h_{ij} in the point X_1

$$\Rightarrow \vec{n}(X_1 - X_2) = 0$$

$(B_i - B_j)(X_1 - X_2) = 0 \Rightarrow (B_i - B_j)$ normal vector to h_{ij}

$$\vec{n} = \frac{B_i - B_j}{\|B_i - B_j\|}$$

$$\vec{n}X_1 = \frac{B_i - B_j}{\|B_i - B_j\|} X_1 = -\frac{C_i - C_j}{\|B_i - B_j\|}$$

- ⊙ $(B_i - B_j)$ defines hyperplane orientation
- ⊙ $(C_i - C_j)$ defines hyperplane location
- ⊖ only convex S_i



Minimum Distance Classifier

$\forall i = 1, \dots, K \quad X_i^j \quad j = 1, \dots, n_i \quad \omega_i \text{ prototypes}$

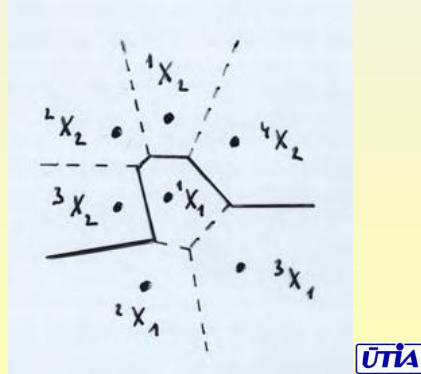
$X \rightarrow \omega_i \text{ iff } \delta(X, X_i) = \min_j \{\delta(X, X_j)\}$

$\delta(X, X_j) = X^T X - 2X^T X_j + X_j^T X_j \quad \text{squared Euclidean distance}$

$S = \bigcup_{j=1}^{n_1 + \dots + n_K} S_j \quad S_j \text{ convex polyhedral region - Dirichlet tessellation}$

(Euclidean classifier $X_i = \mu_i$)

⊖ piecewise linear DF cannot use
lin. DF learning methods



UTIA

Minimum Distance Classifier

$\forall i = 1, \dots, K \quad X_i^j \quad j = 1, \dots, n_i \quad \omega_i \text{ prototypes}$

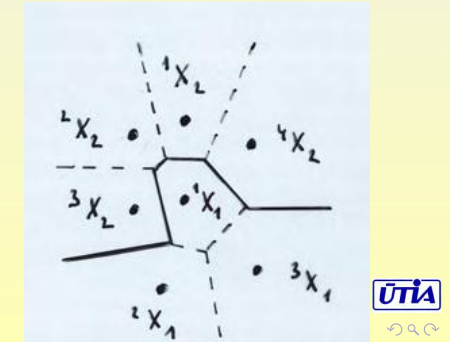
$$g_i(X) = \max_{j=1, \dots, n_i} \{g_i^j(X)\}$$

$$g_i^j(X) = X_i^{j,T} X - \frac{1}{2} X_i^{j,T} X_i^j$$

$S = \bigcup_{j=1}^{n_1 + \dots + n_K} S_j \quad S_j \text{ convex polyhedral region - Dirichlet tessellation}$

(Euclidean classifier $X_i = \mu_i$)

⊖ piecewise linear DF cannot use
lin. DF learning methods



UTIA

Generalized Linear Discriminant Functions

(Nonlinear Discriminant Functions)

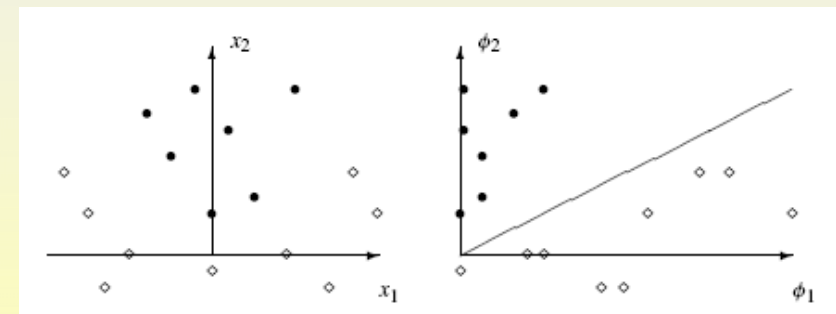
$$g_i(X) = \sum_j B_{ij} \quad {}^j x_{k_1}^{m_1} \quad {}^j x_{k_2}^{m_2} \dots {}^j x_{k_\nu}^{m_\nu} + C_i$$

UTIA

Generalized Linear Discriminant Functions

(Nonlinear Discriminant Functions)

$$g_i(X) = \sum_j B_{ij} \quad {}^j x_{k_1}^{m_1} \quad {}^j x_{k_2}^{m_2} \dots {}^j x_{k_\nu}^{m_\nu} + C_i$$



$$\phi_1(X) = x_1^2$$

$$\phi_2(X) = x_2$$

UTIA

Generalized Linear Discriminant Functions

(Nonlinear Discriminant Functions)

$$g_i(X) = \sum_j B_{ij} x_{k_1}^{m_1} x_{k_2}^{m_2} \dots x_{k_\nu}^{m_\nu} + C_i$$

$$m_i \in \{0, 1\} \quad j = 1, \dots, \eta$$

$$k_j = 1, \dots, l \quad \eta = \binom{l}{\nu}$$

linear functions $\nu = 1, \eta = l$ quadratic functions $\nu = 2, \eta = \frac{(l-1)l}{2}$ ν th order polynomials

$$Y = [x_{k_1}^{m_1} \dots x_{k_\nu}^{m_\nu}, \dots, x_{k_1}^{m_1} \dots x_{k_\nu}^{m_\nu}, 1]^T$$



Generalized Linear Discriminant Functions 2

$$y_i = x_{k_1}^{m_1} \dots x_{k_\nu}^{m_\nu}$$

linearly independent

$$g_i(Y) = W_i Y$$

- ⊕ any nonlinear DF can be approximated
- ⊖ $\eta > l$ rapid dimensionality increase
- ⊖ mapping $X \rightarrow Y$ ($l \rightarrow \eta$) Y intrinsic dimensionality
 $l \rightarrow p(Y|\omega_i)$ degenerate pdf, $\exists$ moments
- ⊖ optimal mapping unknown



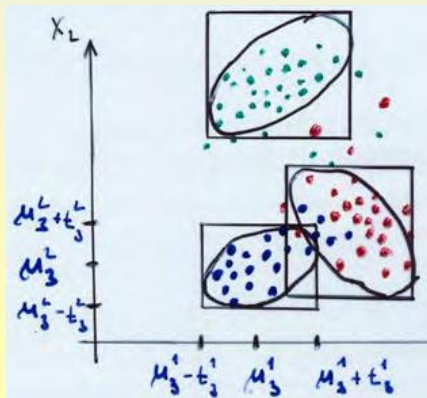
Parallelepiped Classifier

assign X to ω_i if $\exists! \omega_i$:

$$\mu_i - t_i \leq X \leq \mu_i + t_i$$

- ⊙ t_i - hyperellipsoid decision boundaries approximated by hyperrectangles (parallelepipeds)

- ⊖ overlapping parallelepipeds



Nonparametric Classifiers - Parzen Classifiers

$$\begin{aligned} g_i(X) &= \ln p(X|\omega_i) + \ln P(\omega_i) \\ &= \ln \frac{1}{n_i} \sum_{j=1}^{n_i} K(X, X_j) + \ln P(\omega_i) \end{aligned}$$

- ⊖ Parzen classifier needs all X_j as the prototype set, too time consuming for large n_i
- ⊖ small n_i leads to spiky approximations, big oversmooths the pdfs



Nearest Neighbour Decision Rule

NNR ideas:

- ① samples close to each other in a feature space are likely to belong to the same class $\rightarrow 1 - NNR$
- ② samples close to each other in a feature space are likely to have the about the same a posteriori distributions of their respective classes $\rightarrow k - NNR$

suppose we have

$T = \{(X_1, \theta_1), \dots, (X_n, \theta_n)\}$ where $\theta_i \in \{\omega_1, \dots, \omega_K\}$ is the true class of X_i



Nearest Neighbour Decision Rule

NNR ideas:

- ① samples close to each other in a feature space are likely to belong to the same class $\rightarrow 1 - NNR$
- ② samples close to each other in a feature space are likely to have the about the same a posteriori distributions of their respective classes $\rightarrow k - NNR$

suppose we have

$T = \{(X_1, \theta_1), \dots, (X_n, \theta_n)\}$ where $\theta_i \in \{\omega_1, \dots, \omega_K\}$ is the true class of X_i

$1 - NNR$

$\hat{\omega}_1 = \hat{\theta}$ if $\delta(X, \hat{X}) = \min_{i=1, \dots, n} \delta(X, X_i)$
for some metric $\delta()$



Nearest Neighbour Decision Rule

NNR ideas:

- ① samples close to each other in a feature space are likely to belong to the same class $\rightarrow 1 - NNR$
- ② samples close to each other in a feature space are likely to have the about the same a posteriori distributions of their respective classes $\rightarrow k - NNR$

suppose we have

$T = \{(X_1, \theta_1), \dots, (X_n, \theta_n)\}$ where $\theta_i \in \{\omega_1, \dots, \omega_K\}$ is the true class of X_i

$k - NNR$

$\hat{\omega}_k = \hat{\theta}$ if $\hat{\theta} = \text{majority}(\theta_1, \dots, \theta_k) \wedge$
 $\delta(X, X_1) \leq \delta(X, X_2) \leq \dots \leq \delta(X, X_k)$



Nearest Neighbour Decision Rule

NNR ideas:

- ① samples close to each other in a feature space are likely to belong to the same class $\rightarrow 1 - NNR$
- ② samples close to each other in a feature space are likely to have the about the same a posteriori distributions of their respective classes $\rightarrow k - NNR$

suppose we have

$T = \{(X_1, \theta_1), \dots, (X_n, \theta_n)\}$ where $\theta_i \in \{\omega_1, \dots, \omega_K\}$ is the true class of X_i

$(k, l) - NNR$

$\hat{\omega}_k = \hat{\theta}$ if $\hat{\theta} = \text{majority}(\theta_1, \dots, \theta_k) \geq l \wedge$
 $\delta(X, X_1) \leq \delta(X, X_2) \leq \dots \leq \delta(X, X_k)$



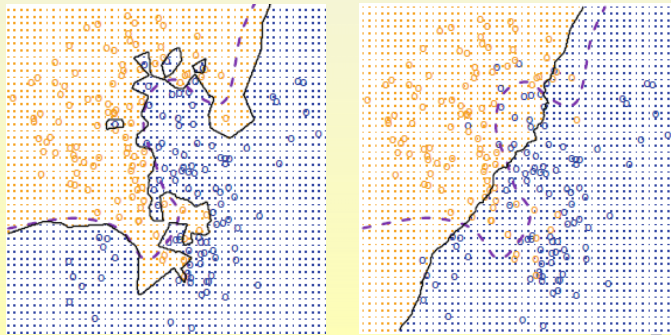
Nearest Neighbour Decision Rule

NNR ideas:

- 1 samples close to each other in a feature space are likely to belong to the same class $\rightarrow 1 - \text{NNR}$
- 2 samples close to each other in a feature space are likely to have the about the same a posteriori distributions of their respective classes $\rightarrow k - \text{NNR}$

1-NNR

100-NNR



NNR Properties

Convergence

assume continuous $p(X) \Rightarrow \hat{X}_n \xrightarrow{n} X$ in probabilityassume continuous $p(\omega | X) \Rightarrow P(\omega | \hat{X}_n) \xrightarrow{n} P(\omega | X)$ in probability

Error Bounds

finite-sample conditional probability of error

$$P(\omega \neq \hat{\theta} | X, \hat{X}_n) = \sum_i P(\omega_i | X) [1 - P(\omega_i | \hat{X}_n)]$$

large-sample conditional probability of error

$$P(\omega \neq \hat{\theta} | X, \hat{X}_n) = 1 - \sum_i P^2(\omega_i | X)$$

$$k - \text{NNR} \quad \forall k \quad P_k(\text{error}) \geq P_{\text{Bayes}}(\text{error})$$

$$1 - \text{NNR} \quad P_1(\text{error}) \leq P_{\text{Bayes}}(\text{error}) [2 - \frac{K}{K-1} P_{\text{Bayes}}(\text{error})]$$

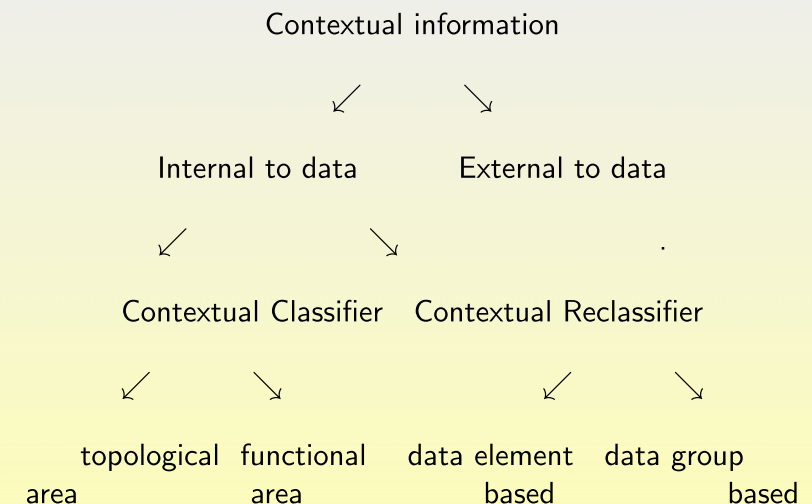


NNR Properties

- ⊕ no underlying distributions
- ⊖ large T computationally expensive
- ⊖ error $\geq$ Bayes rate
- ⊖ intolerant to noise in the features
- ⊖ intolerant to redundant features
- ⊖ sensitive to the choice of the similarity function
- ⊖ cannot handle missing values
- ⊖ provide little information regarding the structure of the data



Contextual Classification



Contextual Classification 2

I index space corresponding to S

context-free decision rule
assign X to ω_i if

$$P(\omega_i|X) = \max_{j=1,\dots,K} \{P(\omega_j|X)\}$$

optimal context decision rule - unsolvable
assign X to ω_i if

$$P(\omega_i|X_k \forall k \in I) = \max_{j=1,\dots,K} \{P(\omega_j|X_k \forall k \in I)\}$$



Contextual Classification 3

simplifying assumptions

- 1 Markovianity

$$g_i(X_r) = P(\omega_i|X_k \forall k \in I_r \cup r)$$

- 2 conditional statistical independence
- 3 Markovianity & total probability theorem
- 4 3) & 2)
- 5 3) and known Ω_r
- 6 5) and $p(\theta_j|\theta_h \forall h \neq j) = p(\theta_j|\theta_r) \quad j \neq r$
- 7 3) and neglecting of class label arrangement
- 8 4) and $p(X_r|\theta_r, X_s, \theta_s, s \neq r) = p(X_r|\theta_r)$

$$g_i(X_r) = p(X_r|\omega_i) \sum_{\{\Omega_r\}} p(\Omega_r, \omega_i) \prod_{j=1}^{\text{card}(I_r)} p(X_j|\theta_j)$$



Contextual Classification 3

simplifying assumptions

- 1 Markovianity
- 2 conditional statistical independence

$$g_i(X_r) = \prod_{j=1}^{\text{card}(I_r)+1} P(\omega_j)p(X_j|\omega_j)$$

- 3 Markovianity & total probability theorem
- 4 3) & 2)
- 5 3) and known Ω_r
- 6 5) and $p(\theta_j|\theta_h \forall h \neq j) = p(\theta_j|\theta_r) \quad j \neq r$
- 7 3) and neglecting of class label arrangement
- 8 4) and $p(X_r|\theta_r, X_s, \theta_s, s \neq r) = p(X_r|\theta_r)$

$$g_i(X_r) = p(X_r|\omega_i) \sum_{\{\Omega_r\}} p(\Omega_r, \omega_i) \prod_{j=1}^{\text{card}(I_r)} p(X_j|\theta_j)$$



Contextual Classification 3

simplifying assumptions

- 1 Markovianity
- 2 conditional statistical independence
- 3 Markovianity & total probability theorem

$$g_i(X_r) = \sum_{\{\Omega_r\}} p(X_k \forall k \in I_r|\omega_i, \Omega_r) p(\Omega_r, \omega_i)$$

$\Omega_r = \{\theta_1, \dots, \theta_{\text{card}(I_r)}\}$ true classes from I_r

$p(X_k \forall k \in I_r|\omega_i, \Omega_r)$ multivariate GMRF

- 4 3) & 2)
- 5 3) and known Ω_r
- 6 5) and $p(\theta_j|\theta_h \forall h \neq j) = p(\theta_j|\theta_r) \quad j \neq r$
- 7 3) and neglecting of class label arrangement
- 8 4) and $p(X_r|\theta_r, X_s, \theta_s, s \neq r) = p(X_r|\theta_r)$

$$g_i(X_r) = p(X_r|\omega_i) \sum_{\{\Omega_r\}} p(\Omega_r, \omega_i) \prod_{j=1}^{\text{card}(I_r)} p(X_j|\theta_j)$$



Contextual Classification 3

simplifying assumptions

- 1 Markovianity
- 2 conditional statistical independence
- 3 Markovianity & total probability theorem
- 4 3) & 2)

$$g_i(X_r) = \sum_{\{\Omega_r\}} \prod_{j=1}^{\text{card}(I_r)} p(X_j|\omega_i, \Omega_r) p(\Omega_r, \omega_i)$$

⊖ too complex - small K and I_r

- 5 3) and known Ω_r
- 6 5) and $p(\theta_j|\theta_h \forall h \neq j) = p(\theta_j|\theta_r) \quad j \neq r$
- 7 3) and neglecting of class label arrangement
- 8 4) and $p(X_r|\theta_r, X_s, \theta_s, s \neq r) = p(X_r|\theta_r)$

$$g_i(X_r) = p(X_r|\omega_i) \sum_{\{\Omega_r\}} p(\Omega_r, \omega_i) \prod_{j=1}^{\text{card}(I_r)} p(X_j|\theta_j)$$



Contextual Classification 3

simplifying assumptions

- 1 Markovianity
- 2 conditional statistical independence
- 3 Markovianity & total probability theorem
- 4 3) & 2)
- 5 3) and known Ω_r

$$g_i(X_r) = p(\omega_i|X_k \forall k \in I_r, \Omega_r)$$

or

$$g_i(X_r) = p(X_k \forall k \in I_r|\Omega_r, \omega_i) p(\omega_i|\Omega_r)$$

- 6 5) and $p(\theta_j|\theta_h \forall h \neq j) = p(\theta_j|\theta_r) \quad j \neq r$
- 7 3) and neglecting of class label arrangement
- 8 4) and $p(X_r|\theta_r, X_s, \theta_s, s \neq r) = p(X_r|\theta_r)$

$$g_i(X_r) = p(X_r|\omega_i) \sum_{\{\Omega_r\}} p(\Omega_r, \omega_i) \prod_{j=1}^{\text{card}(I_r)} p(X_j|\theta_j)$$



Contextual Classification 3

simplifying assumptions

- 1 Markovianity
- 2 conditional statistical independence
- 3 Markovianity & total probability theorem
- 4 3) & 2)
- 5 3) and known Ω_r
- 6 5) and $p(\theta_j|\theta_h \forall h \neq j) = p(\theta_j|\theta_r) \quad j \neq r$
i.e. the knowledge of true class of X_r is sufficient

$$p(\Omega_r|\omega_i) = \prod_{j=1}^{\text{card}(I_r)} p(\theta_j|\omega_i)$$

$$g_i(X_r) = p(X_k \forall k \in I_r|\Omega_r, \omega_i) p^{1-\text{card}(I_r)}(\omega_i) \prod_{j=1}^{\text{card}(I_r)} p(\omega_i|\theta_j)$$

- 7 3) and neglecting of class label arrangement
- 8 4) and $p(X_r|\theta_r, X_s, \theta_s, s \neq r) = p(X_r|\theta_r)$



Contextual Classification 3

simplifying assumptions

- 1 Markovianity
- 2 conditional statistical independence
- 3 Markovianity & total probability theorem
- 4 3) & 2)
- 5 3) and known Ω_r
- 6 5) and $p(\theta_j|\theta_h \forall h \neq j) = p(\theta_j|\theta_r) \quad j \neq r$
- 7 3) and neglecting of class label arrangement

$$g_i(X_r) = \sum_{\{\Omega_r\}} p(X_k \forall k \in I_r|\omega_i, \Omega_r) p(\omega_i) \prod_{j=1}^{\text{card}(I_r)} p(\theta_j)$$

- 8 4) and $p(X_r|\theta_r, X_s, \theta_s, s \neq r) = p(X_r|\theta_r)$

$$g_i(X_r) = p(X_r|\omega_i) \sum_{\{\Omega_r\}} p(\Omega_r, \omega_i) \prod_{j=1}^{\text{card}(I_r)} p(X_j|\theta_j)$$



simplifying assumptions

- 1 Markovianity
- 2 conditional statistical independence
- 3 Markovianity & total probability theorem
- 4 3) & 2)
- 5 3) and known Ω_r
- 6 5) and $p(\theta_j|\theta_h \forall h \neq j) = p(\theta_j|\theta_r) \quad j \neq r$
- 7 3) and neglecting of class label arrangement
- 8 4) and $p(X_r|\theta_r, X_s, \theta_s, s \neq r) = p(X_r|\theta_r)$

$$g_i(X_r) = p(X_r|\omega_i) \sum_{\{\Omega_r\}} p(\Omega_r, \omega_i) \prod_{j=1}^{card(I_r)} p(X_j|\theta_j)$$

