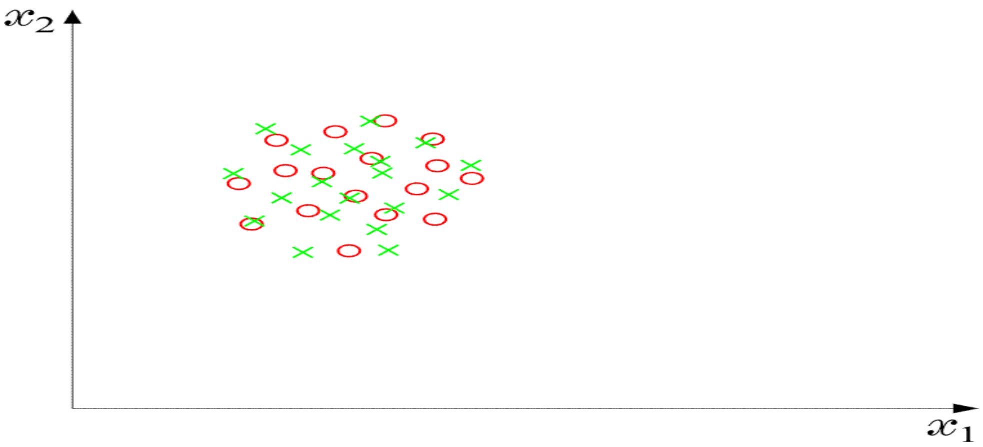
Ch5: FEATURE SELECTION

- The goals:
  - Select the "optimum" number l of features
  - Select the "best" l features

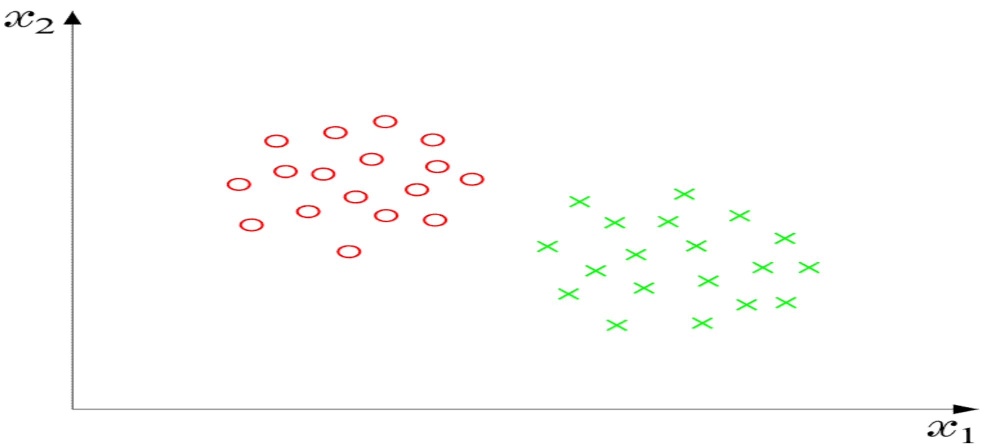
- Large l has a three-fold disadvantage:
  - High computational demands
  - Low generalization performance
  - Poor error estimates

- Given N
  - l must be large enough to learn
    - what makes classes different
    - what makes patterns in the same class similar
  - l must be small enough not to learn what makes patterns of the same class different
  - In practice, $l < N/3$ has been reported to be a sensible choice for a number of cases

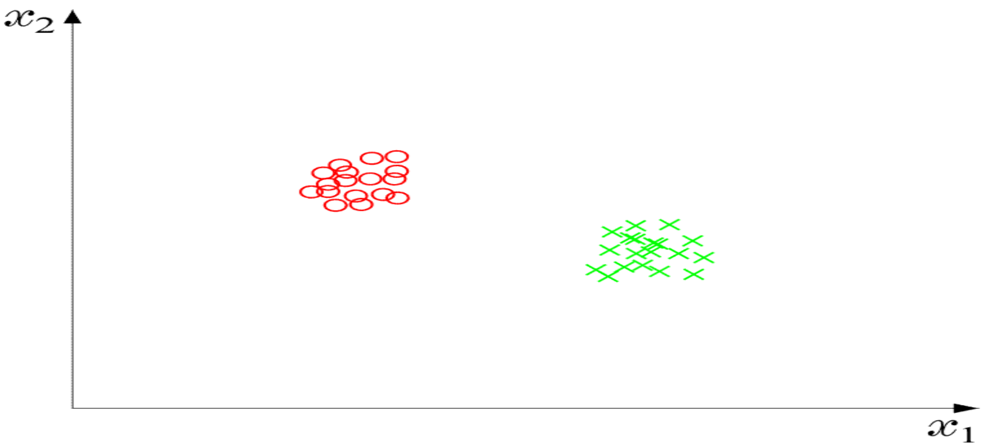
- Once l has been decided, choose the l most informative features
  - Best: **Large** between class distance,
Small within class variance



Bad choice



Not bad choice



Good choice

- ❖ The basic philosophy
  - ➢ Discard individual features with poor information content
  - ➢ The remaining rich information features are examined jointly as vectors

- ❖ *5.4 Feature Selection based on statistical Hypothesis Testing
  - ➢ The Goal: For each individual feature, find whether the values, which the feature takes for the different classes, differ significantly.

    That is, answer

    - $H_1: \ \theta_1 \neq \theta_0$: The values differ significantly
    - $H_0: \ \theta_1 = \theta_0$: The values do not differ significantly

    If they do not differ significantly reject feature from subsequent stages.

- ❖ * Hypothesis Testing Basics

- The steps:
  - N measurements $x_i,\ i = 1,\ 2,\ \ldots,\ N$ are known
  - Define a function of them

    $$q = f(x_1, x_2, \ldots, x_N):$$ test statistic

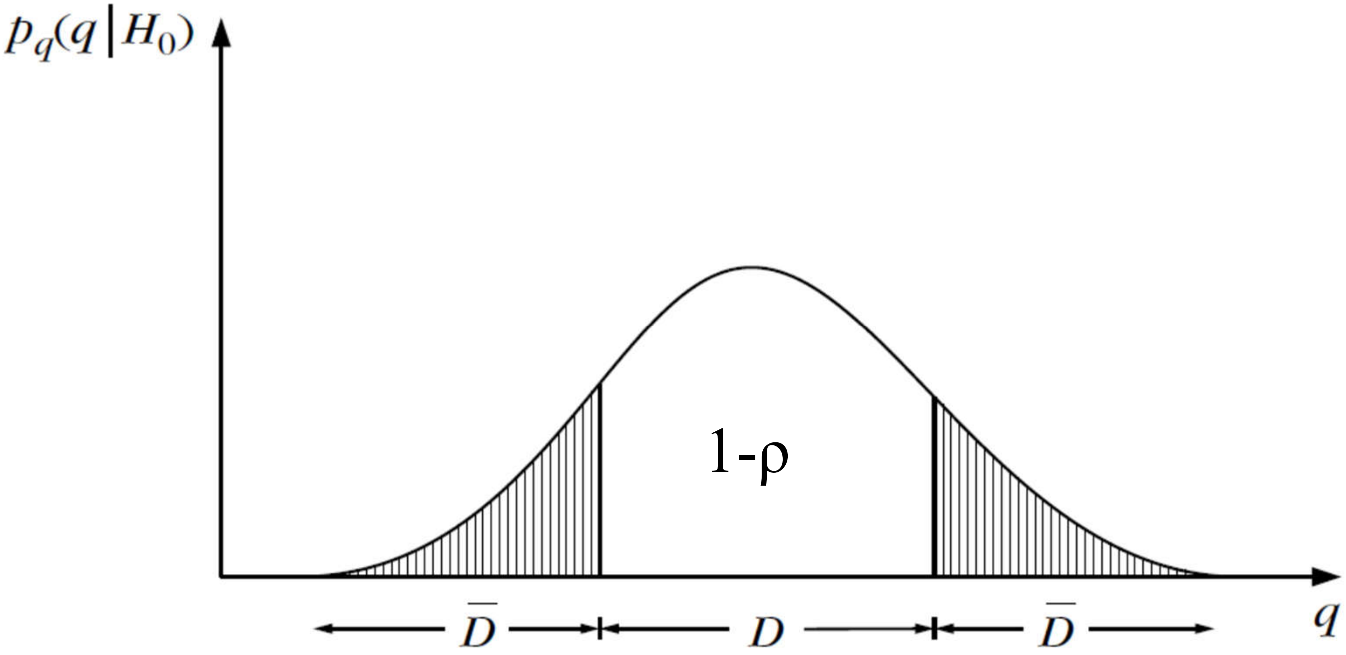
    so that $\boxed{p_q(q;\theta)}$ is easily parameterized in terms of θ.
  - Let D be an interval, where q has a high probability to lie under H_0, i.e., $p_q(q|\theta_0)$
  - Let $\bar{D}$ be the complement of D

    $D \longrightarrow$ Acceptance Interval

    $\bar{D} \longrightarrow$ Critical Interval
  - If q, resulting from $x_1, x_2, \ldots, x_N$, lies in D we accept H_0, otherwise we reject it.

- Probability of an error

$$p_q(q \in \overline{D} | H_0) = \rho$$



- ρ is preselected and it is known as the significance level.

❖ Application: The known variance case:

- Let x be a random variable and the experimental samples, $x_i = 1,2,...,N$, are assumed mutually independent. Also let

$$E[x] = \mu$$

$$E[(x-\mu)^2] = \sigma^2$$

- Compute the sample mean

$$\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

- This is also a random variable with mean value

$$E[\overline{x}] = \frac{1}{N}\sum_{i=1}^{N} E[x_i] = \mu$$

That is, it is an Unbiased Estimator

- The variance $\sigma_{\bar{x}}^{2}$

$$E[(\bar{x}-\mu)^2] = E\left[\left(\frac{1}{N}\sum_{i=1}^{N} x_i - \mu\right)^2\right]$$

$$= \frac{1}{N^2}\sum_{i=1}^{N} E[(x_i-\mu)^2] + \frac{1}{N^2}\sum_{i}\sum_{j\neq i} E[(x_i-\mu)(x_j-\mu)]$$

Due to independence

$$\sigma_{\bar{x}}^{2} = \frac{1}{N}\sigma_{x}^{2}$$

That is, it is Asymptotically Efficient

- Hypothesis test

$$H_1 : E[x] \neq \hat{\mu}$$

$$H_0 : E[x] = \hat{\mu}$$

- Test Statistic: Define the variable

$$\boxed{q = \frac{\bar{x} - \hat{\mu}}{\sigma / \sqrt{N}}}$$

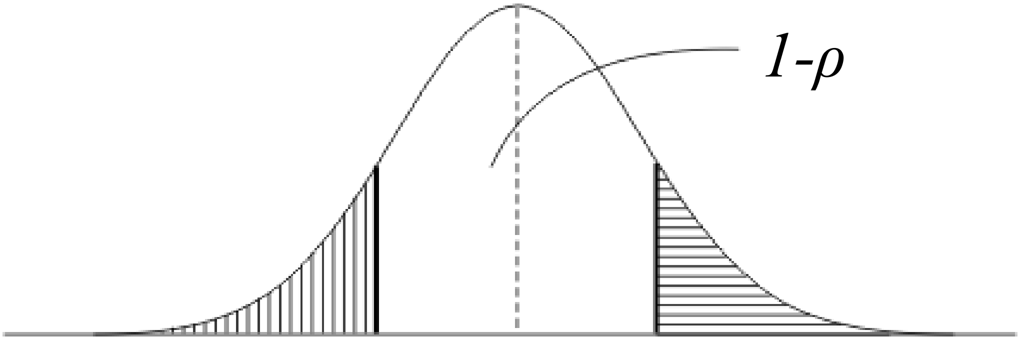
- Central limit theorem under H_0

$$p_{\bar{x}}(\bar{x}) \approx \frac{\sqrt{N}}{\sqrt{2\pi}\sigma} \exp\left(-\frac{N(\bar{x} - \hat{\mu})^2}{2\sigma^2} \right)$$

- Thus, under H_0

$$p_q(q) \approx \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{q^2}{2} \right) \quad \approx N(0,1)$$

- The decision steps
  - Compute q from x_i, $i=1,2,\ldots,N$
  - Choose significance level ρ
  - Compute from $N(0,1)$ tables $D=[-x_\rho, x_\rho]$



  - $\text{if} \quad q \in D \quad \text{accept } H_0$

    $\text{if} \quad q \in \overline{D} \quad \text{reject } H_0$

- An example: A random variable x has variance $\sigma^2=(0.23)^2$. $N=16$ measurements are obtained giving $\overline{x} = 1.35$. The significance level is $\rho=0.05$.

Test the hypothesis

$$H_0 : \mu = \hat{\mu} = 1.4$$

$$H_1 : \mu \neq \hat{\mu}$$

- Since σ² is known, $q = \dfrac{\overline{x} - \hat{\mu}}{\sigma / 4}$ is $N(0,1)$.

  From tables, we obtain the values with acceptance intervals $[-x_\rho, x_\rho]$ for normal $N(0,1)$

$1\text{-}\rho$	0.8	0.85	0.9	0.95	0.98	0.99	0.998	0.999
x_ρ	1.28	1.44	1.64	1.96	2.32	2.57	3.09	3.29

- Thus

$$\text{Prob}\left\{-1.967 < \frac{\overline{x} - \hat{\mu}}{0.23/4} < 1.967\right\} = 0.95$$

or

$$\text{Prob}\{-0.113 < \overline{x} - \hat{\mu} < 0.113\} = 0.95$$

or

$$\boxed{\text{Prob}\{1.237 < \hat{\mu} < 1.463\} = 0.95}$$

- Since $\hat{\mu} = 1.4$ lies within the above acceptance interval, we accept H_0, i.e.,

$$\mu = \hat{\mu} = 1.4$$

The interval [1.237, 1.463] is also known as confidence interval at the $1\text{-}\rho = 0.95$ level.

We say that: There is no evidence at the 5% level that the mean value is not equal to $\hat{\mu}$.

❖ The Unknown Variance Case

- ➢ Estimate the variance. The estimate $\hat{\sigma}^2 = \dfrac{1}{N-1}\sum_{i=1}^{N}(x_i - \overline{x})^2$ is unbiased, i.e., $E[\hat{\sigma}^2] = \sigma^2$

$$E[\hat{\sigma}^2] = \frac{1}{N-1}\sum_{i=1}^{N} E[(x_i - \bar{x})^2] = \frac{1}{N-1}\sum_{i=1}^{N} E\left[((x_i - \mu) - (\bar{x} - \mu))^2\right]$$

$$= \frac{1}{N-1}\sum_{i=1}^{N}\left(\sigma^2 + \frac{\sigma^2}{N} - 2E[(x_i - \mu)(\bar{x} - \mu)]\right)$$

Due to the independence

$$E[(x_i - \mu)(\bar{x} - \mu)] = \frac{1}{N}E\big[(x_i - \mu)\big((x_1 - \mu) + \cdots + (x_N - \mu)\big)\big] = \frac{\sigma^2}{N}$$

$$E[\hat{\sigma}^2] = \frac{N}{N-1}\frac{N-1}{N}\sigma^2 = \sigma^2$$

- ➢ Define the test statistic $q = \dfrac{\overline{x} - \mu}{\hat{\sigma} / \sqrt{N}}$

- This is no longer Gaussian. If x is Gaussian, then q follows a t-distribution, with N-1 degrees of freedom

- An example:

x is Gaussian, $N = 16$, obtained from measurements, $\bar{x} = 1.35$ and $\hat{\sigma}^2 = (0.23)^2$. Test the hypothesis

$H_0: \quad \mu = \hat{\mu} = 1.4$

at the significance level $\rho = 0.025$.

- Table of acceptance intervals for t-distribution

Degrees of Freedom	1-ρ	0.9	0.95	0.975	0.99
12		1.78	2.18	2.56	3.05
13		1.77	2.16	2.53	3.01
14		1.76	2.15	2.51	2.98
15		1.75	2.13	2.49	2.95
16		1.75	2.12	2.47	2.92
17		1.74	2.11	2.46	2.90
18		1.73	2.10	2.44	2.88

- $\text{Prob}\left\{-2.49 < \dfrac{\bar{x} - \hat{\mu}}{\hat{\sigma}/4} < 2.49\right\} = 0.975$

$1.207 < \hat{\mu} < 1.493$

Thus, $\hat{\mu} = 1.4$ is accepted

❖ Application in Feature Selection

- The goal here is to test, against zero, the difference μ_1-μ_2 of the respective means in ω_1, ω_2 of a single feature.

- Let x_i , i=1,…,N , the values of a feature in ω_1

- Let y_i , i=1,…,N , the values of the same feature in ω_2

- Assume in both classes $\sigma_1^2 = \sigma_2^2 = \sigma^2$ (unknown or not)

- The test becomes

$$H_0 : \Delta\mu = \mu_1 - \mu_2 = 0$$

$$H_1 : \Delta\mu \neq 0$$

- Define

  z=x-y

- Obviously

  $E[z]=\mu_1-\mu_2$ independence $\rightarrow \sigma_z^2 = 2\sigma^2$

- Define the average

$$\bar{z} = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i) = \bar{x} - \bar{y} \qquad \bar{z} \approx N\left(\mu_1 - \mu_2, \frac{2\sigma^2}{N}\right)$$

- Known Variance Case: Define

$$q = \frac{(\bar{x} - \bar{y}) - (\hat{\mu}_1 - \hat{\mu}_2)}{\sigma\sqrt{2/N}}$$

- This is $N(0,1)$ and one follows the procedure as before.

- **Unknown Variance Case:** Define the test statistic …

$$q = \frac{(\overline{x} - \overline{y}) - (\mu_1 - \mu_2)}{S_z \sqrt{2/N}}$$

$$S_z^2 = \frac{1}{2N-2}\left(\sum_{i=1}^{N}(x_i - \overline{x})^2 + \sum_{i=1}^{N}(y_i - \overline{y})^2\right)$$

  - $S_z^2(2N-2)/\sigma^2$ is Chi-square distribution with $2N$-2 degrees of freedom,
  - q is t-distribution with $2N$-2 degrees of freedom,
  - Then apply appropriate tables as before.

- Example: The values of a feature in two classes are:

ω_1: 3.5, 3.7, 3.9, 4.1, 3.4, 3.5, 4.1, 3.8, 3.6, 3.7

ω_2: 3.2, 3.6, 3.1, 3.4, 3.0, 3.4, 2.8, 3.1, 3.3, 3.6

Test if the mean values in the two classes differ significantly, at the significance level ρ=0.05

- We have

$$\omega_1: \quad \bar{x} = 3.73, \quad \hat{\sigma}_1^2 = 0.0601$$

$$\omega_2: \quad \bar{y} = 3.25, \quad \hat{\sigma}_2^2 = 0.0672$$

For $N=10$

$$S_z^2 = \frac{1}{2}(\hat{\sigma}_1^2 + \hat{\sigma}_2^2)$$

$$q = \frac{(\bar{x} - \bar{y}) - 0}{S_z \sqrt{2/10}}$$

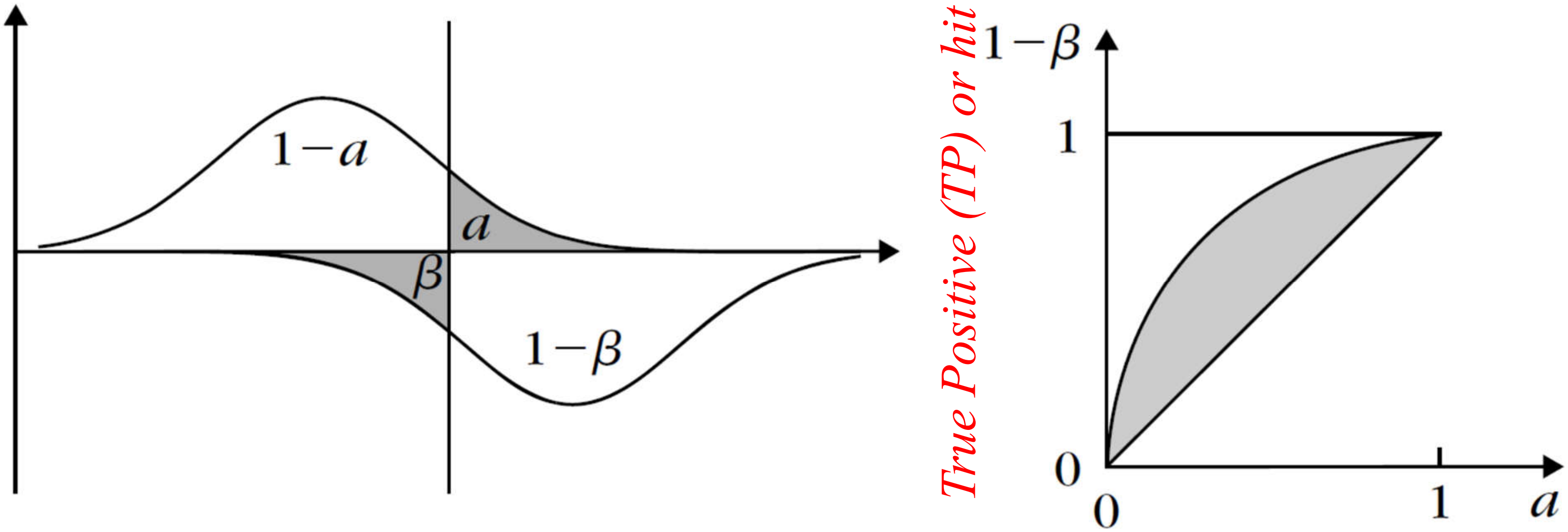
$$\boxed{q = 4.25}$$

- From the table of the t-distribution with $2N$-2=18 degrees of freedom and $\rho=0.05$, we obtain $D=[-2.10,\ 2.10]$ and since $q=4.25$ is outside D, H_1 is accepted and the feature is selected.

❖ The Receiver Operating Characteristics (ROC) Curve

- ➢ The hypothesis tests offer statistical evidence about the difference of the mean values of a single feature in the various classes.
- ➢ This information may not be sufficient to guarantee good discrimination properties of a feature passing the test.
- ➢ We will now focus on techniques providing information about the overlap between the classes.

two overlapping pdf



True Positive (TP) or hit

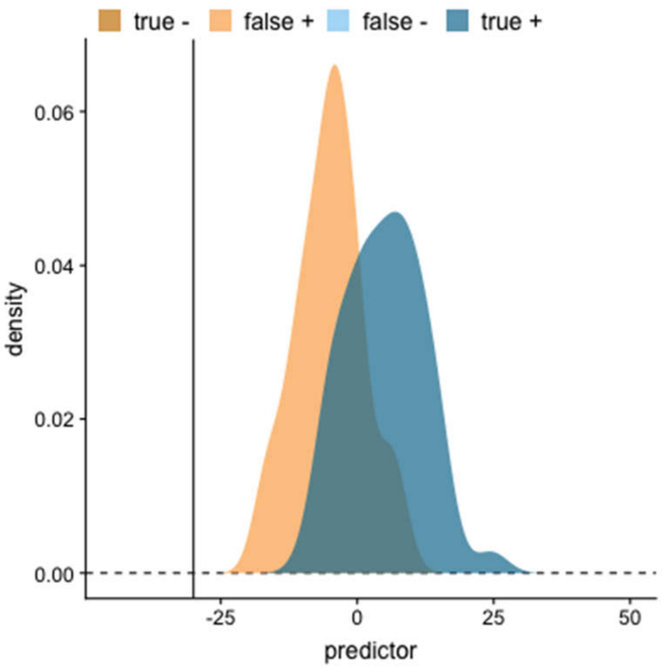
(a)

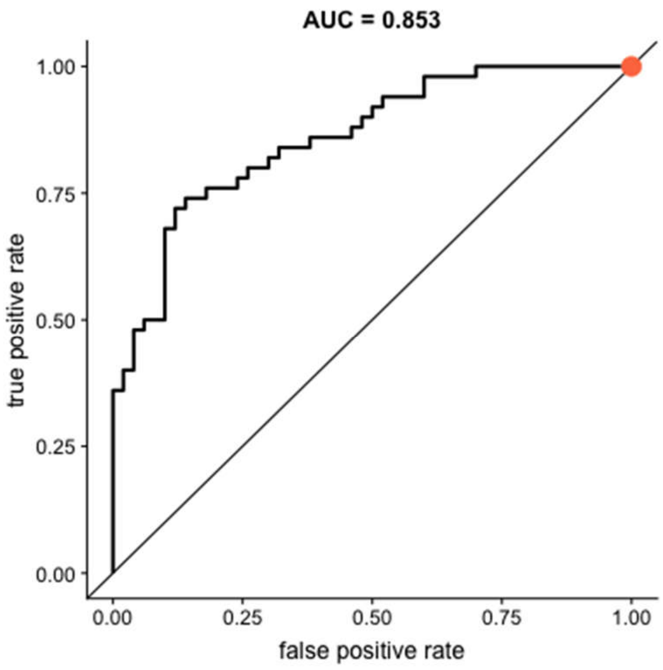
(b)

one pdf has been inverted for illustration purposes

False Positive (FP) Or false alarm

- We decide class ω_1 for values on the left of the threshold and class ω_2 for the values on the right.
- By moving the threshold over “all” possible positions, different values of α and β result.
- The less the overlap of the classes, the larger the area between the curve and the straight line (complete overlap).
- Thus, the aforementioned area varies between zero, for complete overlap, and 1/2 (the area of the upper triangle), for complete separation, and it is a measure of the class discrimination capability of the specific feature. In practice, the ROC curve can easily be constructed by sweeping the threshold and computing percentages of wrong and correct classifications over the available training feature vectors.





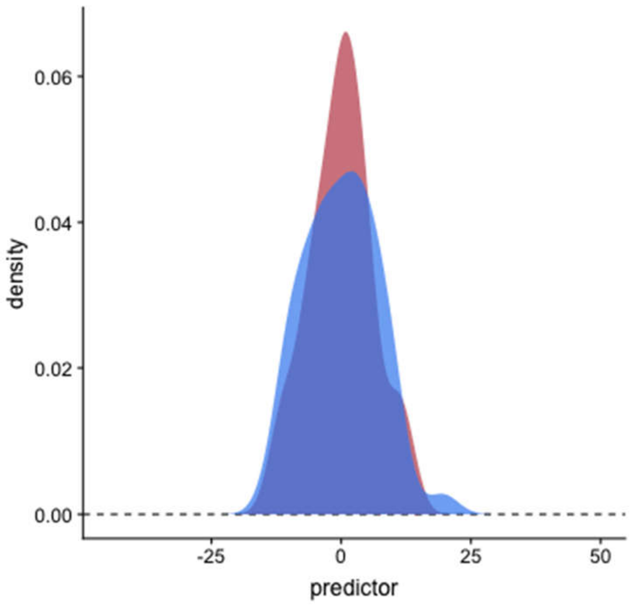
$$\text{Precision} = \frac{tp}{tp + fp}$$

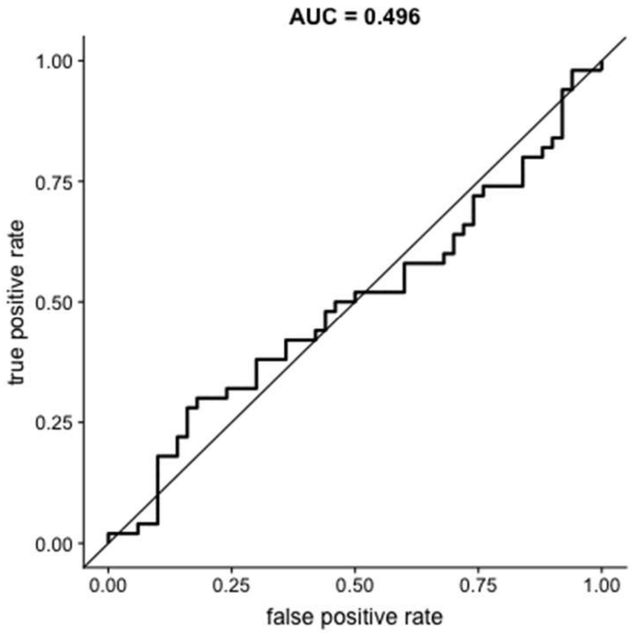
$$\text{Recall} = \frac{tp}{tp + fn}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

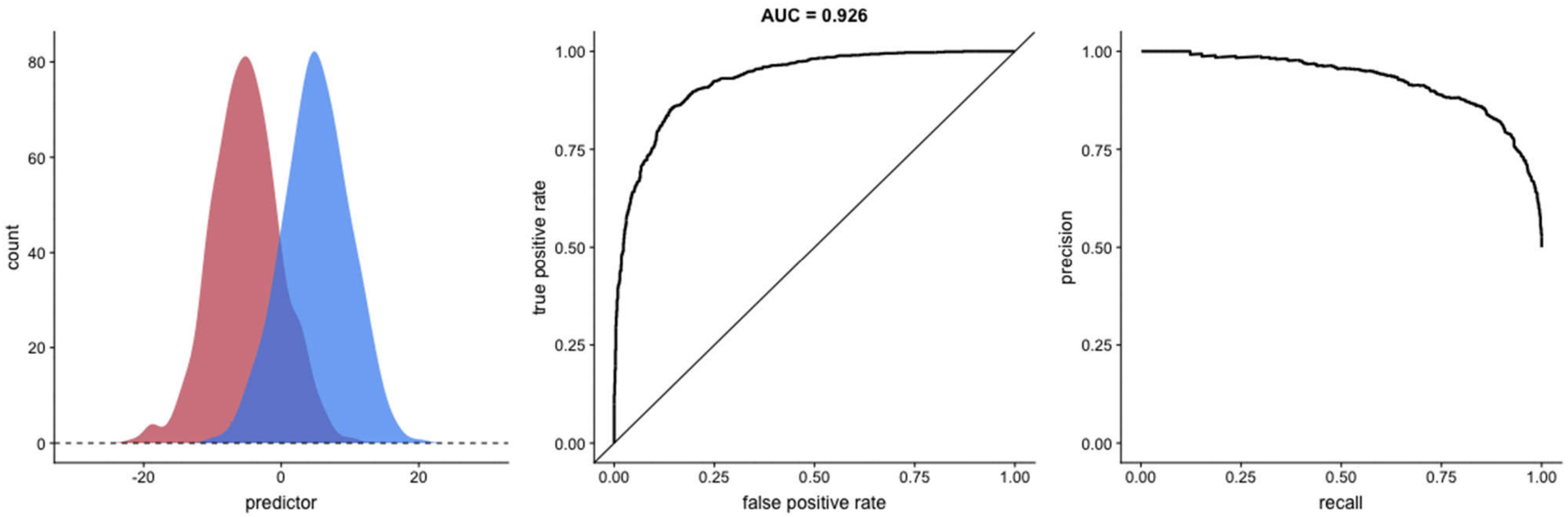
$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

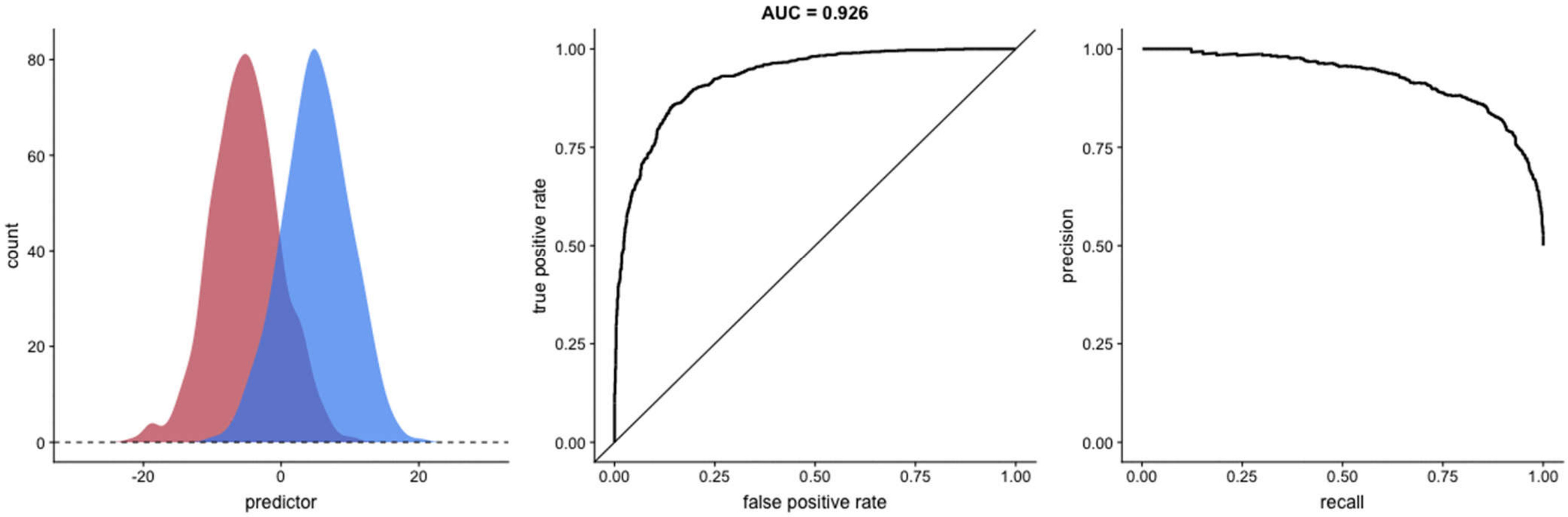
Recall (also called sensitivity)





https://paulvanderlaken.com/2019/08/16/roc-auc-precision-and-recall-visually-explained/


count
80
60
40
20
0
-20
0
20
predictor
AUC = 0.926
true positive rate
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
false positive rate
precision
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
recall


count
80
60
40
20
0
-20
0
20
predictor
AUC = 0.926
true positive rate
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
false positive rate
precision
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
recall

Class Separability Measures

The emphasis so far was on individually considered features. However, such an approach cannot take into account existing correlations among the features. That is, two features may be rich in information, but if they are highly correlated we need not consider both of them. To this end, in order to search for possible correlations, we consider features jointly as elements of vectors. To this end:

- Discard poor in information features, by means of a statistical test.

- Choose the maximum number, ℓ , of features to be used. This is dictated by the specific problem (e.g., the number, N, of available training patterns and the type of the classifier to be adopted).

- Combine remaining features to search for the "best" combination. To this end:

  - Use different feature combinations to form the feature vector. Train the classifier, and choose the combination resulting in the best classifier performance.

    A major disadvantage of this approach is the high complexity. Also, local minima, may give misleading results.

  - Adopt a class separability measure and choose the best feature combination against this cost.

- Class separability measures: Let $\underline{x}$ be the current feature combination vector.
  - Divergence. To see the rationale behind this cost, consider the two – class case. Obviously, if on the average the value of $D_{12}(\underline{x}) = \ln \dfrac{p(\underline{x}|\omega_1)}{p(\underline{x}|\omega_2)}$ is close to zero, then $\underline{x}$ should be a poor feature combination. Define:
    - $D_{12} = \int_{-\infty}^{+\infty} p(\underline{x}|\omega_1) \ln \dfrac{p(\underline{x}|\omega_1)}{p(\underline{x}|\omega_2)} d\underline{x}$ the mean value over class ω_1
    - $D_{21} = \int_{-\infty}^{+\infty} p(\underline{x}|\omega_2) \ln \dfrac{p(\underline{x}|\omega_2)}{p(\underline{x}|\omega_1)} d\underline{x}$ the mean value over class ω_2
    - $d_{12} = D_{12} + D_{21}$

    d_{12} is known as the divergence and can be used as a class separability measure.

- For the multi-class case, define d_{ij} for every pair of classes ω_i, ω_j and the average divergence is defined as

$$d = \sum_{i=1}^{M} \sum_{j=1}^{M} P(\omega_i) P(\omega_j) d_{ij}$$

$$d_{ij} = D_{ij} + D_{ji} = \int_{-\infty}^{+\infty} \left(p(\underline{x} \mid \omega_i) - p(\underline{x} \mid \omega_j) \right) \ln \frac{p(\underline{x} \mid \omega_i)}{p(\underline{x} \mid \omega_j)} d\underline{x}$$

- Some properties:

$$d_{ij} \geq 0$$

$$d_{ij} = 0, \text{if } i = j$$

$$d_{ij} = d_{ji}$$

- Large values of d are indicative of good feature combination.

- ❖ If the components of the feature vector are statistically independent

$$d_{ij}(x_1, x_2, \ldots, x_l) = \sum_{r=1}^{l} d_{ij}(x_r)$$

- ❖ If the density functions are Gaussians $N(\boldsymbol{\mu}_i , \boldsymbol{\Sigma}_i)$ and $N(\boldsymbol{\mu}_j , \boldsymbol{\Sigma}_j)$ the divergence is simplified as:

$$d_{ij} = \frac{1}{2} trace\left\{\boldsymbol{\Sigma}_i^{-1}\boldsymbol{\Sigma}_j + \boldsymbol{\Sigma}_j^{-1}\boldsymbol{\Sigma}_i - 2\mathbf{I}\right\} + \frac{1}{2}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T \left(\boldsymbol{\Sigma}_i^{-1} + \boldsymbol{\Sigma}_j^{-1}\right)(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)$$

- ❖ For the one-dimensional case this becomes

$$d_{ij} = \frac{1}{2}\left(\frac{\sigma_j^2}{\sigma_i^2} + \frac{\sigma_i^2}{\sigma_j^2} - 2\right) + \frac{1}{2}(\mu_i - \mu_j)^2\left(\frac{1}{\sigma_i^2} + \frac{1}{\sigma_j^2}\right)$$

- ❖ A class separability measure cannot depend only on the difference of the mean values; it must also be variance dependent.

❖ If the covariance matrices of the two Gaussian distributions are equal $\mathbf{\Sigma}_i = \mathbf{\Sigma}_j = \mathbf{\Sigma}$

$$d_{ij} = (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T \mathbf{\Sigma}^{-1} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)$$

which is nothing other than the Mahalanobis distance between the corresponding mean vectors.

❖ In this case we have a direct relation between the divergence d_{ij} and the Bayes error—that is, the minimum error we can achieve by adopting the specific feature vector.

✓ In the sequel, we will try to define class separability measures with a closer relationship to the Bayes error.

Error Bounds for Normal Densities

- The full calculation of the error for the Gaussian case would be quite difficult, especially in high dimensions, because of the discontinuous nature of the decision regions in the integral.

$$\begin{aligned} P(correct) &= \sum_{i=1}^{c} P(\mathbf{x} \in \mathcal{R}_i, \omega_i) = \sum_{i=1}^{c} P(\mathbf{x} \in \mathcal{R}_i | \omega_i) P(\omega_i) \\ &= \sum_{i=1}^{c} \int_{\mathcal{R}_i} p(\mathbf{x}|\omega_i) P(\omega_i) \, d\mathbf{x}. \end{aligned}$$

- In the two-category case the general error integral can be approximated analytically to give us an upper bound on the error.

Chernoff Bound

- To derive a bound for the error, we need the following inequality:

  $\min[a, b] \le a^{\beta} b^{1-\beta}$ for $a, b \ge 0$ and $0 \le \beta \le 1$.

- Assume $a \ge b$. Thus we need only show that $b \le a^{\beta} b^{1-\beta} = (a/b)^{\beta} b$. But this inequality is manifestly valid, since $(a/b)^{\beta} \ge 1$.

- We had $$P(error) = \int_{-\infty}^{\infty} p(error, x)dx = \int_{-\infty}^{\infty} P(error \mid x) p(x) dx$$

$$\boxed{P(\omega_j|x) = \frac{p(x|\omega_j)P(\omega_j)}{p(x)},}$$

$$P(error|x) = \min\,[P(\omega_1|x), P(\omega_2|x)].$$

Thus we apply this inequality to get the bound:

$$\boxed{P(error) \leq P^{\beta}(\omega_1)P^{1-\beta}(\omega_2)\int p^{\beta}(\mathbf{x}|\omega_1)p^{1-\beta}(\mathbf{x}|\omega_2)\ d\mathbf{x}}$$

$$\text{for } 0 \leq \beta \leq 1.$$

This integral is over *all* feature space.

If the conditional probabilities are normal, this integral can be evaluated analytically, yielding:

$$\int p^{\beta}(\mathbf{x}|\omega_1)p^{1-\beta}(\mathbf{x}|\omega_2)\ d\mathbf{x} = e^{-k(\beta)}$$

where

$$\begin{aligned} k(\beta) \quad &= \quad \frac{\beta(1-\beta)}{2}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)^t\left[(1-\beta)\boldsymbol{\Sigma}_1+\beta\boldsymbol{\Sigma}_2\right]^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2) \\ & \quad +\frac{1}{2}\ln\frac{(1-\beta)\boldsymbol{\Sigma}_1+\beta\boldsymbol{\Sigma}_2}{|\boldsymbol{\Sigma}_1|^{1-\beta}|\boldsymbol{\Sigma}_2|^{\beta}}. \end{aligned}$$

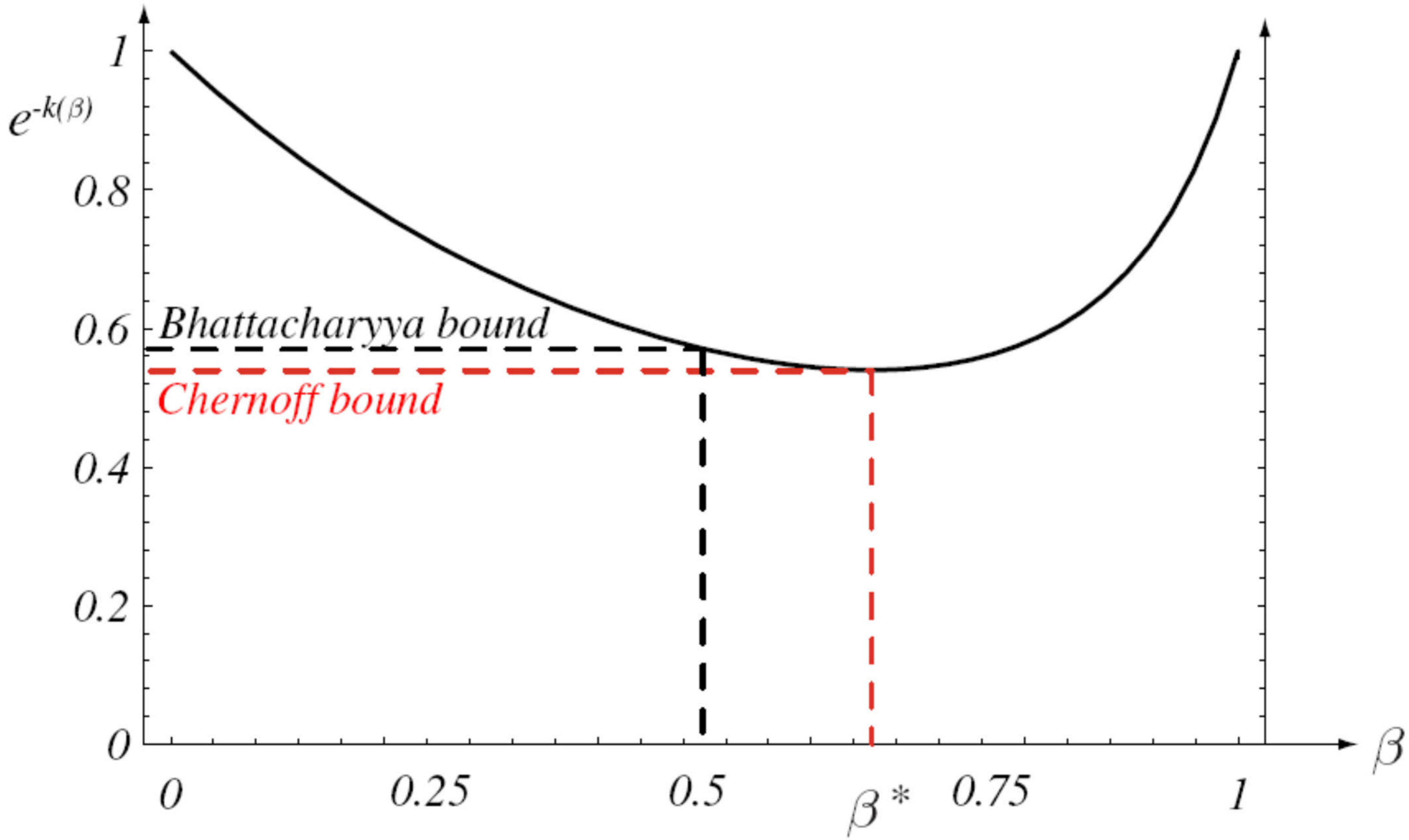


FIGURE 2.18. The Chernoff error bound is never looser than the Bhattacharyya bound. For this example, the Chernoff bound happens to be at $\beta^*= 0.66$, and is slightly tighter than the Bhattacharyya bound ($\beta = 0.5$).

- *k(β)* is called *Chernoff distance*. The *Chernoff bound*, on *P*(*error*) is found by analytically or numerically finding the value of *β* that minimizes $P^{\beta}(\omega_1)P^{1-\beta}(\omega_2)\, e^{-k(\beta)}$ and substituting the results in Eq. *P(error)*=...

Bhattacharyya Bound

- Slightly less tight bound can be derived simply by setting the results for *β* = 1/2. This result is the so-called *Bhattacharyya bound* on the error. Thus,

$$\begin{aligned} P(error) &\leq \sqrt{P(\omega_1)P(\omega_2)} \int \sqrt{p(\mathbf{x}|\omega_1)p(\mathbf{x}|\omega_2)}\, d\mathbf{x} \\ &= \sqrt{P(\omega_1)P(\omega_2)} e^{-k(1/2)}, \end{aligned}$$

where

$$k(1/2) \;=\; 1/8(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^t \left[\frac{\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2}{2}\right]^{-1} (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) + \frac{1}{2}\ln \frac{\left|\frac{\boldsymbol{\Sigma}_1+\boldsymbol{\Sigma}_2}{2}\right|}{\sqrt{|\boldsymbol{\Sigma}_1||\boldsymbol{\Sigma}_2|}}.$$

The term *k(1/2)* is called *Bhattacharyya distance*, and will be used as an important measure of the separability of two distributions. The Chernoff and Bhattacharyya bounds may still be used even if the underlying distributions are not Gaussian. However, for distributions that deviate markedly from a Gaussian, the bounds will not be informative.

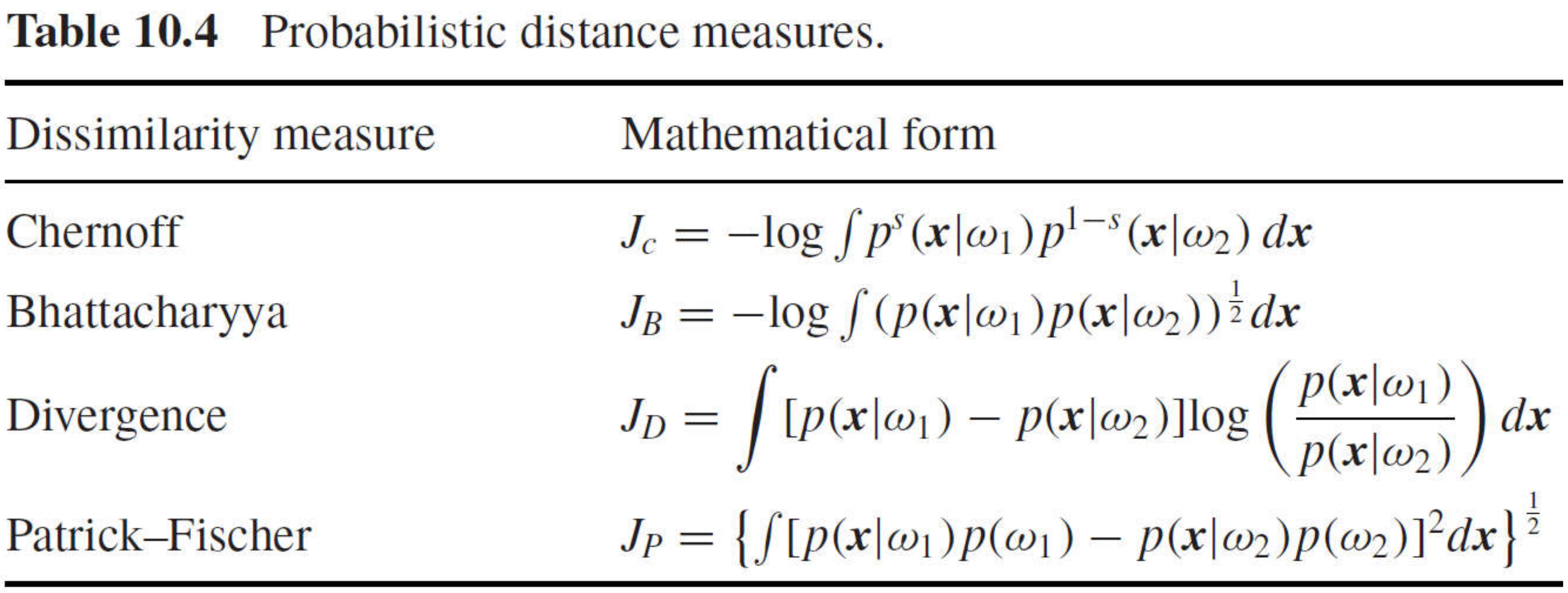


Table 10.4 Probabilistic distance measures.

Dissimilarity measure	Mathematical form
Chernoff	$J_c = -\log \int p^s(\boldsymbol{x}\|\omega_1)\, p^{1-s}(\boldsymbol{x}\|\omega_2)\, d\boldsymbol{x}$
Bhattacharyya	$J_B = -\log \int (p(\boldsymbol{x}\|\omega_1)\, p(\boldsymbol{x}\|\omega_2))^{\frac{1}{2}} d\boldsymbol{x}$
Divergence	$J_D = \int [p(\boldsymbol{x}\|\omega_1) - p(\boldsymbol{x}\|\omega_2)] \log \left(\frac{p(\boldsymbol{x}\|\omega_1)}{p(\boldsymbol{x}\|\omega_2)}\right) d\boldsymbol{x}$
Patrick–Fischer	$J_P = \left\{\int [p(\boldsymbol{x}\|\omega_1)\, p(\omega_1) - p(\boldsymbol{x}\|\omega_2)\, p(\omega_2)]^2 d\boldsymbol{x}\right\}^{\frac{1}{2}}$

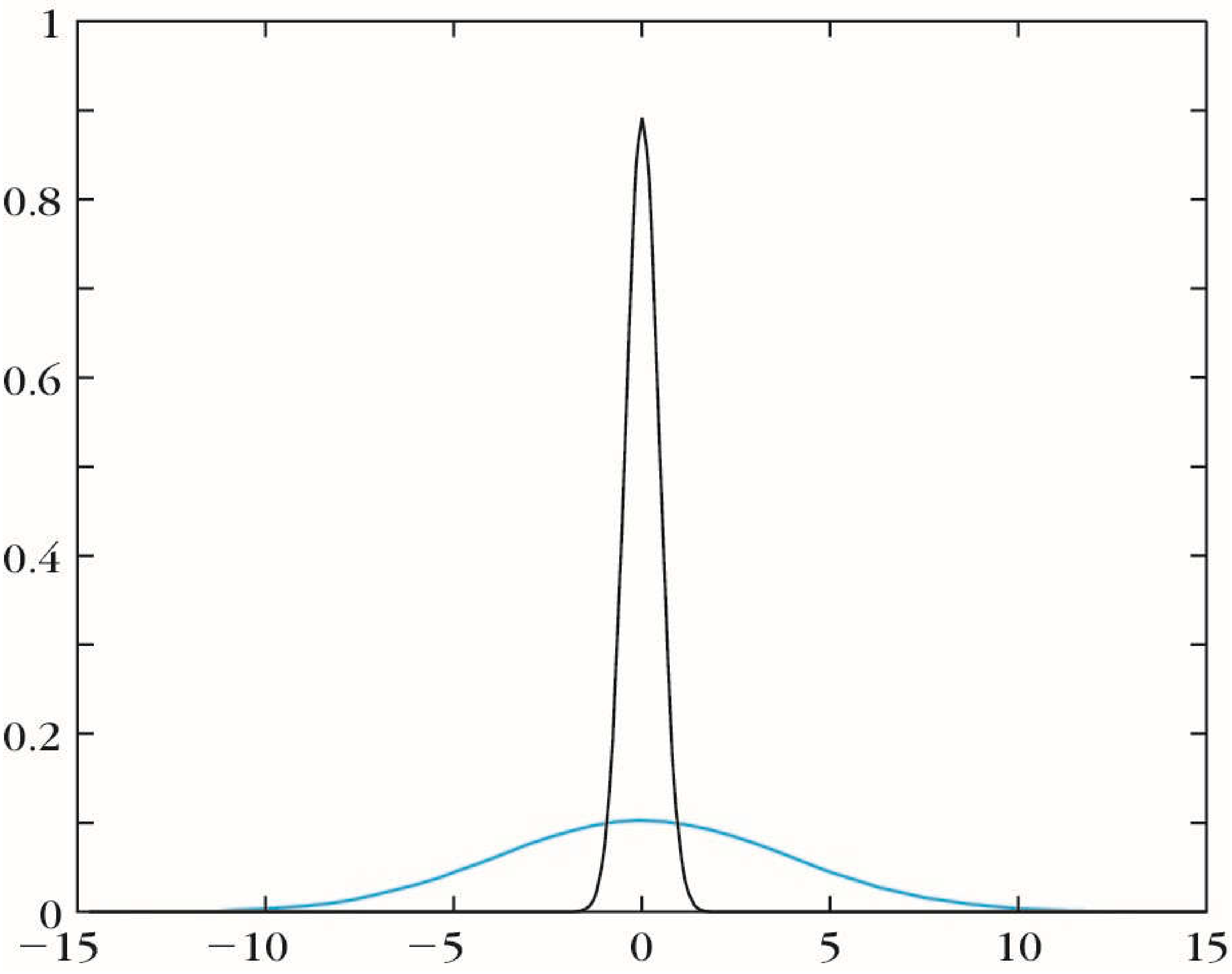
Example 5.4

Assume that $P(\omega_1)=P(\omega_2)$ and that the corresponding distributions are Gaussians, $N(\boldsymbol{\mu}\ ,\sigma_1{}^2\mathbf{I})$ and $N(\boldsymbol{\mu}\ ,\ \sigma_2{}^2\mathbf{I})$. The Bhattacharyya distance becomes

$$k\left(\frac{1}{2}\right)=\frac{1}{2}\ln\frac{\left(\dfrac{\sigma_1^2+\sigma_2^2}{2}\right)^l}{\sqrt{\sigma_1^{2l}\sigma_2^{2l}}}=\frac{1}{2}\ln\left(\frac{\sigma_1^2+\sigma_2^2}{2\sigma_1\sigma_2}\right)^l$$

For the one-dimensional case $l = 1$ and for $\sigma_1 = 10\sigma_2$, $k(1/2)= 0.8097$ and $P_e \geq 0.2225$.
If $\sigma_1 = 100\sigma_2$, $k(1/2)= 1.9561$ and $P_e \geq 0.0707$.

Thus, the greater the difference of the variances, the smaller the error bound. The decrease is bigger for higher dimensions due to the dependence on l.



Gaussian pdfs with the same mean and different variances (σ_1 =1, σ_2=0.01).

- **Scatter Matrices.** These are used as a measure of the way data are scattered in the respective feature space.
  - Within-class scatter matrix

$$S_w = \sum_{i=1}^{M} P_i S_i$$

where

$$S_i = \sum_{\underline{x} \in D_i} \left(\underline{x} - \underline{\mu}_i\right)\left(\underline{x} - \underline{\mu}_i\right)^T$$

and

$$P_i \equiv P(\omega_i) \approx \frac{n_i}{N}$$

n_i the number of training samples in ω_i.

$\text{trace}\{S_w\}$ is a measure of the average variance of the features.

- Between-class scatter matrix

$$S_b = \sum_{i=1}^{M} P_i \left(\underline{\mu}_i - \underline{\mu}_0\right)\left(\underline{\mu}_i - \underline{\mu}_0\right)^T$$

$$\underline{\mu}_0 = \sum_{i=1}^{M} P_i \underline{\mu}_i$$

$\mathrm{trace}\{S_b\}$ is a measure of the average distance of the mean of each class from the respective global one.

- Mixture scatter matrix

$$S_m = \sum_{\underline{x}} \left(\underline{x} - \underline{\mu}_0\right)\left(\underline{x} - \underline{\mu}_0\right)^{\mathrm{T}}$$

It turns out that:

$$S_m = S_w + S_b$$

Measures based on Scatter Matrices.

- $$J_1 = \frac{\text{trace}\{S_m\}}{\text{trace}\{S_w\}}$$
- $$J_2 = \frac{|S_m|}{|S_w|} = |S_w^{-1} S_m|$$
- $$J_3 = \text{trace}\{S_w^{-1} S_m\}$$

- Other criteria are also possible, by using various combinations of S_m, S_b, S_w.
- The above J_1, J_2, J_3 criteria take high values for the cases where:
  - Data are clustered together within each class.
  - The means of the various classes are far.

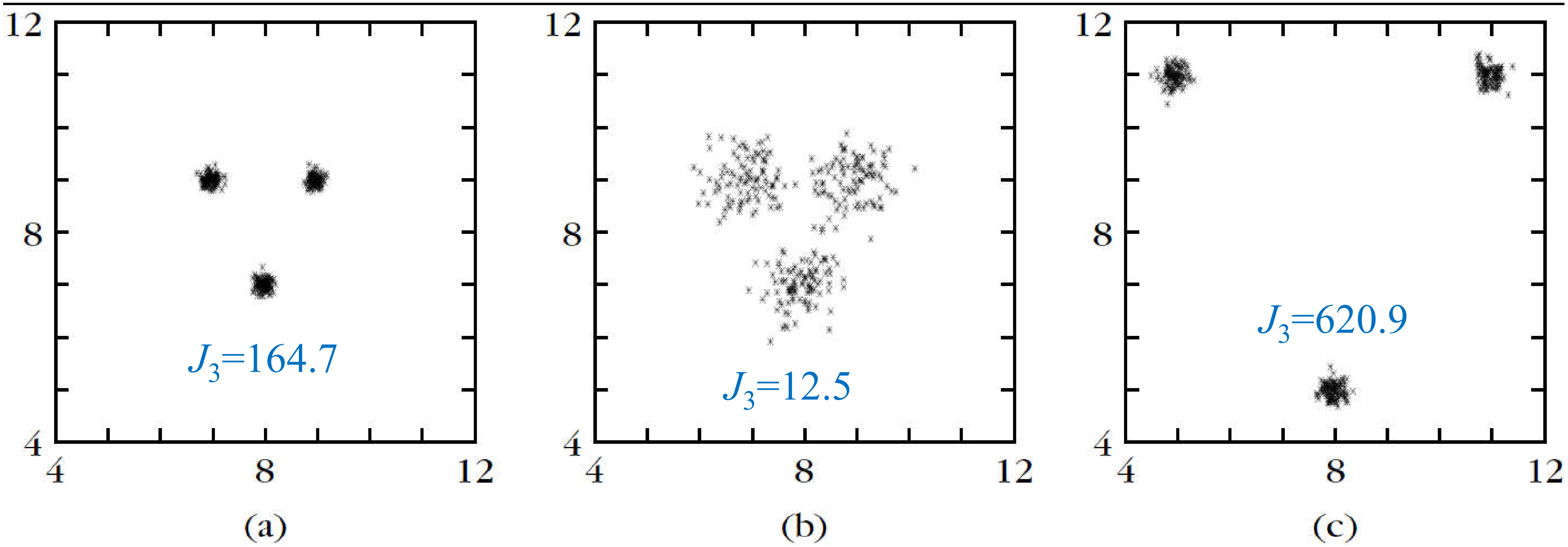


Figure 5.5 Shows three cases of classes at different locations and within-class variances. The resulting values for the J_3 criterion involving the S_w and S_m matrices are 164.7, 12.5, and 620.9 for the cases in Figures 5.5a, b, and c, respectively. That is, the best is for distant well-clustered classes and the worst for the case of closely located classes with large within- class variance.

Other Scatter Matrices Criteria

- The sum of squared error is defined as

$$J_e = \sum_{i=1}^{M} \sum_{\underline{x} \in D_i} \left\| \underline{x} - \underline{\mu}_i \right\|^2$$

- The trace (sum of diagonal elements) is the simplest scalar measure of the scatter matrix, as it is proportional to the sum of the variances in the coordinate directions

$$tr[S_W] = \sum_{i=1}^{M} tr[S_i] = \sum_{i=1}^{M} \sum_{\underline{x} \in D_i} \left\| \underline{x} - \underline{\mu}_i \right\|^2 = J_e$$

- $J_d = |S_W| = \left|\sum_{i=1}^{M} S_i\right|$
- $\frac{|S_W|}{|S_m|} = \prod_{i=1}^{l} \frac{1}{1+\lambda_i}$

- $tr[S_B] = \sum_{i=1}^{M} n_i \left\|\underline{\mu}_i - \underline{\mu}\right\|^2$

- $tr[S_W^{-1} S_B] = \sum_{i=1}^{l} \lambda_i$
- $J_f = tr[S_m^{-1} S_W] = \sum_{i=1}^{l} \frac{1}{1+\lambda_i}$

Why?

- Fisher's discriminant ratio. In one dimension and for two equiprobable classes the determinants become:

$$\left|S_w\right| \propto \sigma_1^2 + \sigma_2^2$$

$$\left|S_b\right| \propto \left(\mu_1 - \mu_2\right)^2$$

and

$$\frac{\left|S_b\right|}{\left|S_w\right|} = \frac{\left(\mu_1 - \mu_2\right)^2}{\sigma_1^2 + \sigma_2^2}$$

known as Fisher's Discriminant Ratio (FDR).

- For the multiclass case, averaging forms of FDR can be used

$$FDR_1 = \sum_{i}^{M} \sum_{j \neq i}^{M} \frac{\left(\mu_i - \mu_j\right)^2}{\sigma_i^2 + \sigma_j^2}$$

Feature Subset Selection

- Ways to combine features:

  Trying to form all possible combinations of ℓ features from an original set of m selected features is a computationally hard task. Thus, a number of suboptimal searching techniques have been derived.

- Sequential backward selection. Let x_1, x_2, x_3, x_4 the available features (m=4). The procedure consists of the following steps:
  - Adopt a class separability criterion C (could also be the error rate of the respective classifier). Compute its value for ALL features considered jointly $[x_1, x_2, x_3, x_4]^T$.
  - Eliminate one feature and for each of the possible resulting combinations, that is $[x_1, x_2, x_3]^T$, $[x_1, x_2, x_4]^T$, $[x_1, x_3, x_4]^T$, $[x_2, x_3, x_4]^T$, compute the class separability criterion value C. Select the best combination, say $[x_1, x_2, x_3]^T$.

- From the above selected feature vector eliminate one feature and for each of the resulting combinations, $[x_2, x_3]^T$, $[x_1, x_3]^T$, $[x_1, x_2]^T$, compute C and select the best combination.

  The above selection procedure shows how one can start from m features and end up with the "best" ℓ ones. Obviously, the choice is suboptimal. The number of required calculations is:

$$1+\frac{1}{2}\big((m+1)m-\ell(\ell+1)\big)$$

(for m=20, l =5, the number equals 196.)

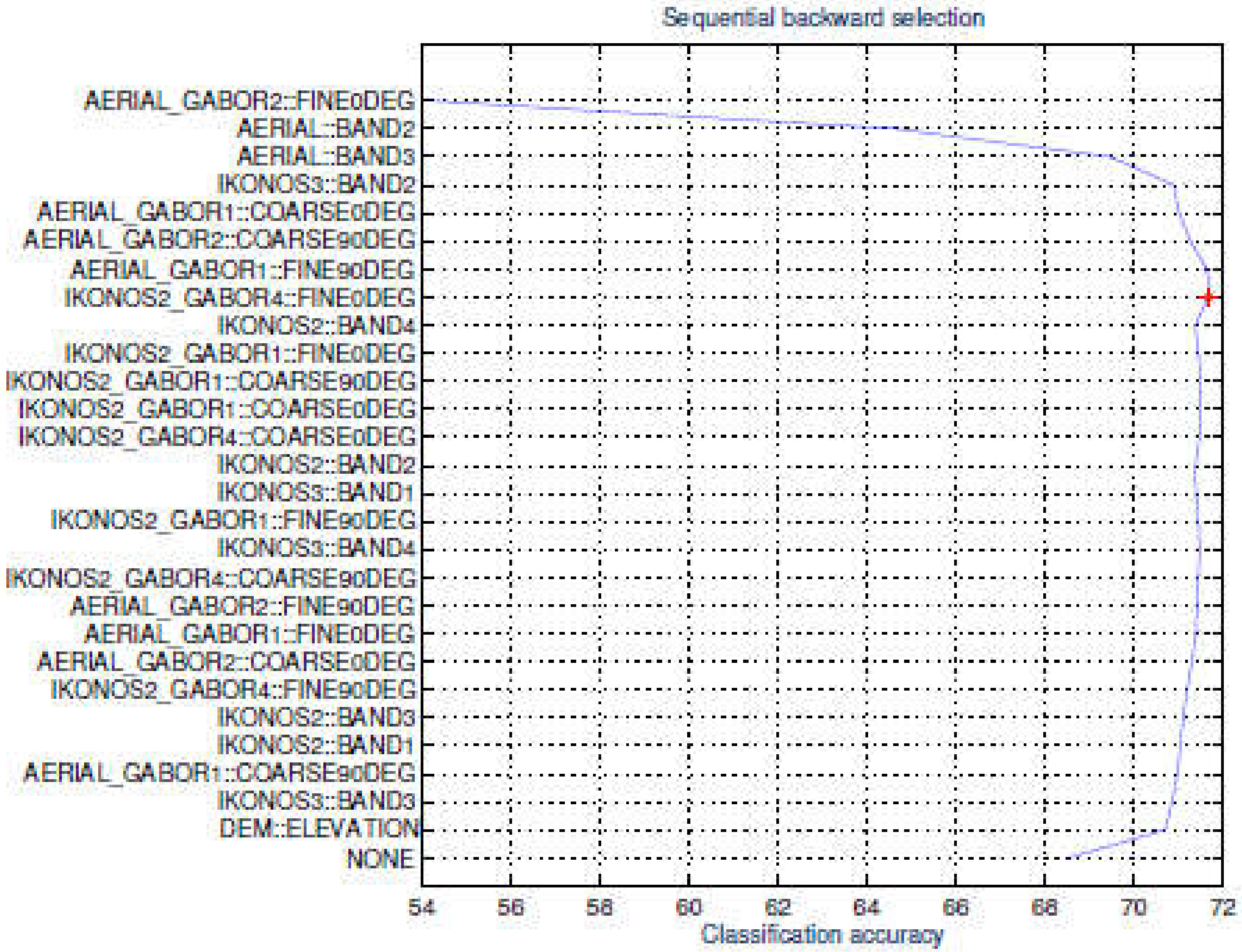
In contrast, a full search requires:

$$\binom{m}{\ell}=\frac{m!}{\ell!(m-\ell)!}$$

operations (for m=20, l =5, the number equals 15,504.).

Example

features removed at each iteration



Results of sequential backward feature selection for classification of a satellite image using 28 features. x-axis shows the classification accuracy (%) and y-axis shows the features removed at each iteration (the first iteration is at the top). The highest accuracy value is shown with a star.

- Sequential forward selection. Here the reverse procedure is followed.
  - Compute C for each feature. Select the "best" one, say x_1
  - For all possible 2D combinations of x_1, i.e., $[x_1, x_2]$, $[x_1, x_3]$, $[x_1, x_4]$ compute C and choose the best, say $[x_1, x_3]$.
  - For all possible 3D combinations of $[x_1, x_3]$, e.g., $[x_1, x_3, x_2]$, etc., compute C and choose the best one.

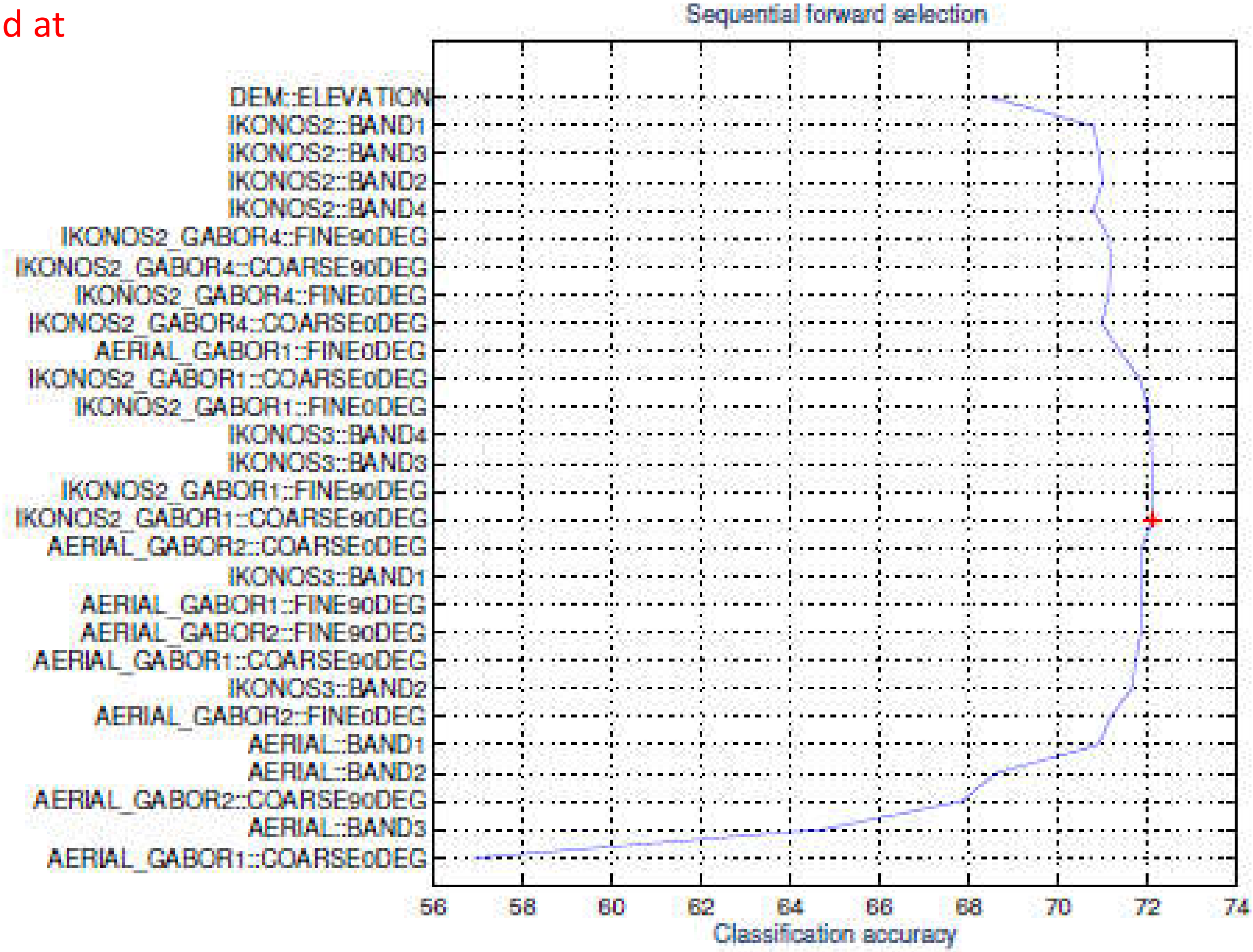
  The above procedure is repeated till the "best" vector with ℓ features has been formed. This is also a suboptimal technique, requiring:

  $$\ell m - \frac{\ell(\ell-1)}{2}$$

  operations.

Example

features added at each iteration



Results of sequential forward feature selection for classification of a satellite image using 28 features. x-axis shows the classification accuracy (%) and y-axis shows the features added at each iteration (the first iteration is at the bottom). The highest accuracy value is shown with a star.

- Floating Search Methods

The above two procedures suffer from the nesting effect. Once a bad choice has been done, there is no way to reconsider it in the following steps.

In the floating search methods one is given the opportunity in reconsidering a previously discarded feature or to discard a feature that was previously chosen.

The method is still suboptimal, however it leads to improved performance, at the expense of complexity.

- Remarks:
  - Besides suboptimal techniques, some optimal searching techniques can also be used, provided that the optimizing cost has certain properties, e.g., monotonicity. i.e.

$$C\,(x_1\,,\,\ldots,\,x_i\,) \leq C\,(x_1\,,\,\ldots,\,x_i\,,\,x_{i+1}\,)$$

  - Instead of using a class separability measure (filter techniques) or using directly the classifier (wrapper techniques), one can modify the cost function of the classifier appropriately, so that to perform feature selection and classifier design in a single step (embedded) method.
  - For the choice of the separability measure a multiplicity of costs have been proposed, including information theoretic costs.

- Filter Methods
  - Evaluation is **independent** of the classification algorithm.
  - The objective function evaluates feature subsets by their information content, typically interclass distance, statistical dependence or information-theoretic measures.
- Wrapper Methods
  - Evaluation uses criteria **related** to the classification algorithm.
  - The objective function is a pattern classifier, which evaluates feature subsets by their predictive accuracy (recognition rate on test data) by statistical resampling or cross-validation.

Optimal Feature Generation

- In general, feature generation is a problem-dependent task. However, there are a few general directions common in a number of applications. We focus on three such alternatives.
- **Two class Case**
- The goal is achieved by seeking the direction $\mathbf{w}$ in the m dimensional space, along which the two classes are best separated in some way.
- Given an $\mathbf{x} \in \Re^m$ the scalar $y = \frac{\mathbf{w}^T \mathbf{x}}{\|\mathbf{w}\|}$ is the projection of $\mathbf{x}$ along $\mathbf{w}$. So

$$FDR = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}$$

where μ_1, μ_2 are the mean values and σ_1^2, σ_2^2 the variances of y in the two classes ω_1, ω_2, respectively, after the projection along $\mathbf{w}$.

$$\mu_i = \mathbf{w}^T \boldsymbol{\mu}_i, \quad i = 1, 2.$$

$$\left(\mu_1 - \mu_2\right)^2 = \mathbf{w}^T \left(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\right)\left(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\right)^T \mathbf{w} \propto \mathbf{w}^T \mathbf{S}_b \mathbf{w}$$

$$\sigma_i^2 = E\left[(y - \mu_i)^2\right] = E\left[\mathbf{w}^T (\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^T \mathbf{w}\right] = \mathbf{w}^T \mathbf{S}_i \mathbf{w}, \qquad i = 1, 2$$

$$\sigma_1^2 + \sigma_2^2 \propto \mathbf{w}^T \mathbf{S}_w \mathbf{w}$$

$$\Rightarrow \qquad FDR(\mathbf{w}) = \frac{\left(\mathbf{w}^T \mathbf{S}_b \mathbf{w}\right)}{\left(\mathbf{w}^T \mathbf{S}_w \mathbf{w}\right)}$$

- This is the celebrated generalized Rayleigh quotient, which, as it is known from linear algebra (Problem 5.16), is maximized if $\mathbf{w}$ is chosen such that ...

$$\mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w}$$

where λ is the largest eigenvalue of $\mathbf{S}_w^{-1}\mathbf{S}_b$.

- **Proof:** From linear algebra we know that $\max\limits_{\mathbf{w}} \dfrac{\mathbf{w}^T \mathbf{S}\mathbf{w}}{\mathbf{w}^T \mathbf{w}}$

occurs if $\mathbf{w}$ is chosen to be an eigenvector of the largest eigenvalue of $\mathbf{S}$ (Rayleigh quotient), i.e.

$$\max_{\mathbf{w}} \mathbf{w}^T \mathbf{S}\mathbf{w} - \lambda\left(\mathbf{w}^T \mathbf{w} - 1\right) \Rightarrow 2\mathbf{S}\mathbf{w} - 2\lambda\mathbf{w} = 0 \Rightarrow \mathbf{S}\mathbf{w} = \lambda\mathbf{w}$$

- For the case of our problem (generalized Rayleigh quotient), let $\mathbf{y} \equiv \mathbf{S}_w^{1/2}\mathbf{w} \Rightarrow \mathbf{w} = \mathbf{S}_w^{-1/2}\mathbf{y}$

- Then the problem becomes equivalent with maximizing

$$\max_{\mathbf{y}} \frac{\mathbf{y}^T \mathbf{S}_w^{-1/2}\mathbf{S}_b\mathbf{S}_w^{-1/2}\mathbf{y}}{\mathbf{y}^T \mathbf{y}}$$

where the symmetry of $\mathbf{S}_w$ has been taken into account. The above is maximized if $\mathbf{y}$ is the eigenvector corresponding to the largest eigenvalue λ , i.e.,

$$\mathbf{S}_w^{-1/2}\mathbf{S}_b\mathbf{S}_w^{-1/2}\mathbf{y} = \lambda\mathbf{y}$$

and finally, by replacing $\mathbf{y}$ by $\mathbf{w}$, if $\mathbf{w}$ is chosen to satisfy

$$\mathbf{S}_b\mathbf{w} = \lambda\mathbf{S}_w\mathbf{w}$$

or equivalently solving the eigenvalue task

$$\mathbf{S}_w^{-1}\mathbf{S}_b\mathbf{w} = \lambda\mathbf{w}$$

- The corresponding maximum value is

$$\frac{\left(\mathbf{w}^T\mathbf{S}_b\mathbf{w}\right)}{\left(\mathbf{w}^T\mathbf{S}_w\mathbf{w}\right)} = \lambda\frac{\left(\mathbf{w}^T\mathbf{S}_w\mathbf{w}\right)}{\left(\mathbf{w}^T\mathbf{S}_w\mathbf{w}\right)} = \lambda$$

which justifies the choice of the maximum eigenvalue.

❖ By the definition of $\mathbf{S}_b$ we have that

$$\lambda \mathbf{S}_w \mathbf{w} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \mathbf{w} = \alpha (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$$

where α is a scalar.

$$\mathbf{w} = \mathbf{S}_w^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$$

assuming, of course, that $\mathbf{S}_w$ is invertible.

❖ Thus, we have reduced the number of features from m to 1 in an optimal way.

❖ All that remains is to find the threshold, i.e., the point along the one-dimensional subspace separating the projected points.

❖ When the conditional densities $p(\mathbf{x}|\omega_i)$ are multivariate normal with equal covariance matrices $\boldsymbol{\Sigma}$, we can calculate the threshold directly.

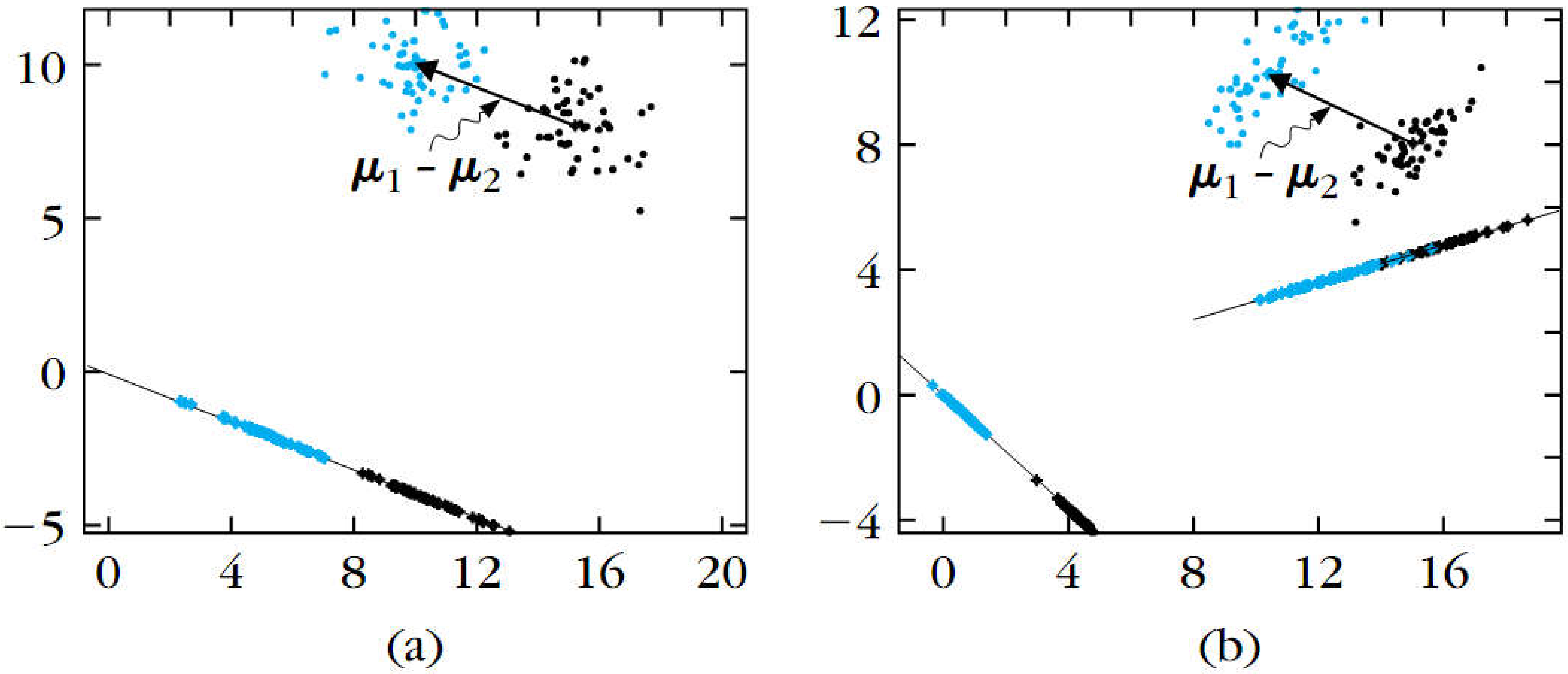


FIGURE 5.6 (a) The optimal line resulting from Fisher's criterion, for two Gaussian classes. Both classes share the same diagonal covariance matrix, with equal elements on the diagonal. The line is parallel to $\boldsymbol{\mu}_1$-$\boldsymbol{\mu}_2$. (b) The covariance matrix for both classes is nondiagonal. The optimal line is on the left. Observe that it is no more parallel to $\boldsymbol{\mu}_1$-$\boldsymbol{\mu}_2$. The line on the right is not optimal and the classes, after the projection, overlap.

The optimal decision boundary is $\mathbf{w}^t\mathbf{x} + w_0 = 0$, $\mathbf{w} = \Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$

and where w_0 is a constant involving **w** and the prior probabilities.

- Thus, for the normal, equal-covariance case, the optimal decision rule is merely to decide ω_1 if Fisher's linear discriminant exceed some threshold, and to decide ω_2 otherwise. (Choose w_0 where the posteriors in the one-dimensional distributions are equal).

- Fisher's method performed feature generation and at the same time the design of a (linear) classifier; it combined the stages of feature generation and classifier design into a single one. The resulting classifier is

$$g(\mathbf{x}) = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \ \mathbf{S}_w^{-1}\mathbf{x} + w_0$$

However, Fisher's criterion does not provide a value for w_0, which has to be determined. For example; if $N(\boldsymbol{\mu}_1 , \boldsymbol{\Sigma})$ & $N(\boldsymbol{\mu}_2 , \boldsymbol{\Sigma})$

$$g(\mathbf{x}) = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \ \mathbf{S}_w^{-1}\left(\mathbf{x} - \frac{1}{2}(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)\right) - \ln\frac{P(\omega_2)}{P(\omega_1)}$$

Optimal Feature Generation

❖ Multiclass Case

- Optimized features based on Scatter matrices (Fisher's linear discrimination).
  - The goal: Given an original set of m measurements $\underline{x} \in \Re^m$, compute $\underline{y} \in \Re^\ell$, by the linear transformation

$$\underline{y} = A^T \underline{x}$$

    so that the J_3 scattering matrix criterion involving $\boldsymbol{S}_w$, $\boldsymbol{S}_b$ is maximized. $\boldsymbol{A}^T$ is an $\ell \times m$ matrix.

  - The basic steps in the proof:
    - $J_3 = \mathrm{trace}\{\boldsymbol{S}_w{}^{-1} \boldsymbol{S}_b\}$
    - $\boldsymbol{S}_{yw} = \boldsymbol{A}^T\boldsymbol{S}_{xw}\boldsymbol{A}$, $\boldsymbol{S}_{yb} = \boldsymbol{A}^T\boldsymbol{S}_{xb}\boldsymbol{A}$,
    - $J_3(\boldsymbol{A})=\mathrm{trace}\{(\boldsymbol{A}^T\boldsymbol{S}_{xw}\boldsymbol{A})^{-1} (\boldsymbol{A}^T S_{xb}\boldsymbol{A})\}$
    - Compute A so that $J_3(A)$ is maximum.

$$\begin{aligned} S_{yi} &= \sum_{\underline{y}\in D_i} \left(\underline{y} - \underline{\mu}_{yi}\right)\left(\underline{y} - \underline{\mu}_{yi}\right)^T \\ &= \sum_{\underline{x}\in D_i} \left(A^T \underline{x} - A^T \underline{\mu}_{xi}\right)\left(A^T \underline{x} - A^T \underline{\mu}_{xi}\right)^T \\ &= \sum_{\underline{x}\in D_i} A^T \left(\underline{x} - \underline{\mu}_{xi}\right)\left(\underline{x} - \underline{\mu}_{xi}\right)^T A = A^T S_{xi} A \end{aligned}$$

$$\frac{\partial J_3(\mathbf{A})}{\partial \mathbf{A}} = \mathbf{0} \Rightarrow$$

$$-2\mathbf{S}_{xw}\mathbf{A}\left(\mathbf{A}^T\mathbf{S}_{xw}\mathbf{A}\right)^{-1}\left(\mathbf{A}^T\mathbf{S}_{xb}\mathbf{A}\right)\left(\mathbf{A}^T\mathbf{S}_{xw}\mathbf{A}\right)^{-1} + 2\mathbf{S}_{xb}\mathbf{A}\left(\mathbf{A}^T\mathbf{S}_{xw}\mathbf{A}\right)^{-1} = \mathbf{0}$$

The solution: $(S_{xw}{}^{-1}S_{xb})A=A(S_{yw}{}^{-1}S_{yb})$

- ✓ Let B be the matrix that diagonalizes simultaneously matrices S_{yw}, S_{yb} , i.e:

  $$B^TS_{yw}B = I\,, \qquad B^TS_{yb}B = D$$

  which are the within- and between-class scatter matrices of the transformed vector, $\hat{\mathbf{y}} = \mathbf{B}^T\mathbf{y} = \mathbf{B}^T\mathbf{A}^T\mathbf{x} = \mathbf{C}^T\mathbf{x}$

  where B is a $\ell\times\ell$ matrix and D a $\ell\times\ell$ diagonal matrix.

- ✓ Note that in going from $\mathbf{y}$ to $\hat{\mathbf{y}}$ there is no loss in the value of the cost J_3. Why?

$$J_3(\hat{\mathbf{y}}) = tr\left\{\mathbf{S}_{\hat{y},w}^{-1}\mathbf{S}_{\hat{y},b}\right\} = tr\left\{\left(\mathbf{B}^T\mathbf{S}_{yw}\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{S}_{yb}\mathbf{B}\right\} = tr\left\{\mathbf{B}^{-1}\mathbf{S}_{yw}^{-1}\mathbf{S}_{yb}\mathbf{B}\right\}$$

$$= tr\left\{\mathbf{S}_{yw}^{-1}\mathbf{S}_{yb}\mathbf{B}\mathbf{B}^{-1}\right\} = tr\left\{\mathbf{S}_{yw}^{-1}\mathbf{S}_{yb}\right\} = J_3(\mathbf{y})$$

$$\left.\begin{array}{l} \left(\mathbf{S}_{xw}^{-1}\mathbf{S}_{xb}\right)\mathbf{A} = \mathbf{A}\mathbf{S}_{yw}^{-1}\mathbf{S}_{yb} \\ \mathbf{B}^T\mathbf{S}_{yw}\mathbf{B} = \mathbf{I} \\ \mathbf{B}^T\mathbf{S}_{yb}\mathbf{B} = \mathbf{D} \end{array}\right\} \begin{array}{l} \Rightarrow \left(\mathbf{S}_{xw}^{-1}\mathbf{S}_{xb}\right)\underbrace{\mathbf{A}\mathbf{B}} = \mathbf{A}\mathbf{S}_{yw}^{-1}\mathbf{S}_{yb}\mathbf{B} = \underbrace{\mathbf{A}\mathbf{B}}\,\underbrace{\mathbf{B}^T\mathbf{S}_{yb}\mathbf{B}} \\ \Rightarrow \underbrace{\left(\mathbf{S}_{xw}^{-1}\mathbf{S}_{xb}\right)}_{m\times m}\underbrace{\mathbf{C}}_{m\times l} = \underbrace{\mathbf{C}\mathbf{D}}_{(m\times l)\times(l\times l)}, \quad \mathbf{C}_{m\times l} \equiv \mathbf{A}\mathbf{B} \end{array}$$

- ➢ Summarize: If $\mathbf{A}$ maximizes $J_3(\mathbf{A})$ then $\left(\mathbf{S}_{xw}^{-1}\mathbf{S}_{xb}\right)\mathbf{C} = \mathbf{C}\mathbf{D}$.

❖This is an eigenvalue-eigenvector problem with the diagonal matrix $\mathbf{D}$ having the eigenvalues of $S_{xw}^{-1}S_{xb}$ on its diagonal and $\mathbf{C}$ having the corresponding eigenvectors as its Columns.

- For an M-class problem, $S_{xw}^{-1}S_{xb}$ is of rank M-1.
- If ℓ=M-1, choose C to consist of the M-1 eigenvectors, corresponding to the non-zero eigenvalues.

$$\underline{\hat{y}} = C^T \underline{x}$$

- The above guarantees max. J_3 value. So, $J_{3,x}= J_{3,y} =J_{3,\hat{\mathbf{y}}}$.

$$J_{3,x} = tr\left\{\mathbf{S}_{xw}^{-1}\mathbf{S}_{xb}\right\} = \lambda_1 + \cdots + \lambda_{M-1} + 0$$

$$J_{3,\hat{y}} = tr\left\{(\mathbf{C}^T\mathbf{S}_{xw}\mathbf{C})^{-1}(\mathbf{C}^T\mathbf{S}_{xb}\mathbf{C})\right\} = tr\left\{(\mathbf{C}^{-1}\mathbf{S}_{xw}^{-1}\mathbf{C}^{-T})(\mathbf{C}^T\mathbf{S}_{xb}\mathbf{C})\right\}$$

$$= tr\left\{(\mathbf{C}^{-1}\mathbf{S}_{xw}^{-1}\mathbf{S}_{xb}\mathbf{C})\right\} = tr\left\{(\mathbf{C}^{-1}\mathbf{CD})\right\} = tr\left\{\mathbf{D}\right\}$$

$$\text{or} \quad \left(\mathbf{S}_{xw}^{-1}\mathbf{S}_{xb}\right)\mathbf{C} = \mathbf{CD} \Rightarrow \mathbf{C}^T\mathbf{S}_{xb}\mathbf{C} = \mathbf{C}^T\mathbf{S}_{xw}\mathbf{CD}$$

$$\Rightarrow J_{3,\hat{y}} = tr\left\{(\mathbf{C}^{-1}\mathbf{S}_{xw}^{-1}\mathbf{C}^{-T})\mathbf{C}^T\mathbf{S}_{xw}\mathbf{CD}\right\} = tr\left\{\mathbf{D}\right\}$$

$$= \lambda_1 + \cdots + \lambda_{M-1} = J_{3,x}$$

- It can be shown that for a two-class problem, this results to the well known Fisher's linear discriminant

$$\hat{y} = C^T \mathbf{x} \Rightarrow \hat{y} = \left(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\right)^T \mathbf{S}_w^{-1} \mathbf{x}$$

- For Gaussian classes, this is the optimal Bayesian classifier, with a difference of a threshold value.
- If $\ell < M\text{-}1$, choose the ℓ eigenvectors corresponding to the ℓ largest eigenvectors.
- In this case, $J_{3,\hat{y}} < J_{3,x}$, that is there is loss of information.

- Geometric interpretation. The vector $\underline{y}$ is the projection of $\underline{x}$ onto the subspace spanned by the eigenvectors of $S_{xw}^{-1} S_{xb}$. The eigenvectors are not necessarily mutually orthogonal!

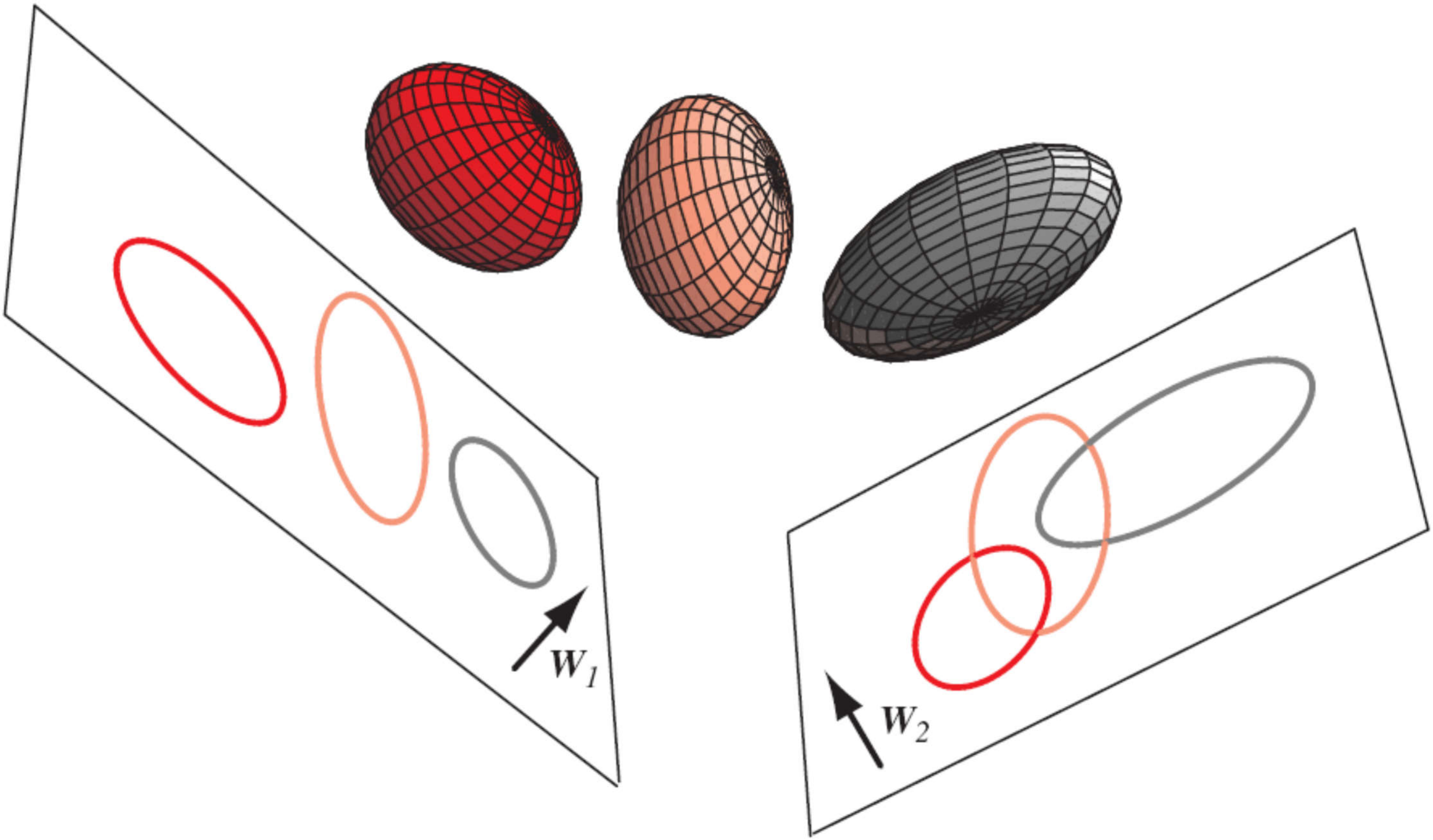


FIGURE 3.6. Three 3-dimensional distributions are projected onto two-dimensional subspaces, described by a normal vectors $\mathbf{W}_1$ and $\mathbf{W}_2$. Informally, multiple discriminant methods seek the optimum such subspace, that is, the one with the greatest separation of the projected distributions for a given total within-scatter matrix, here as associated with $\mathbf{W}_1$.

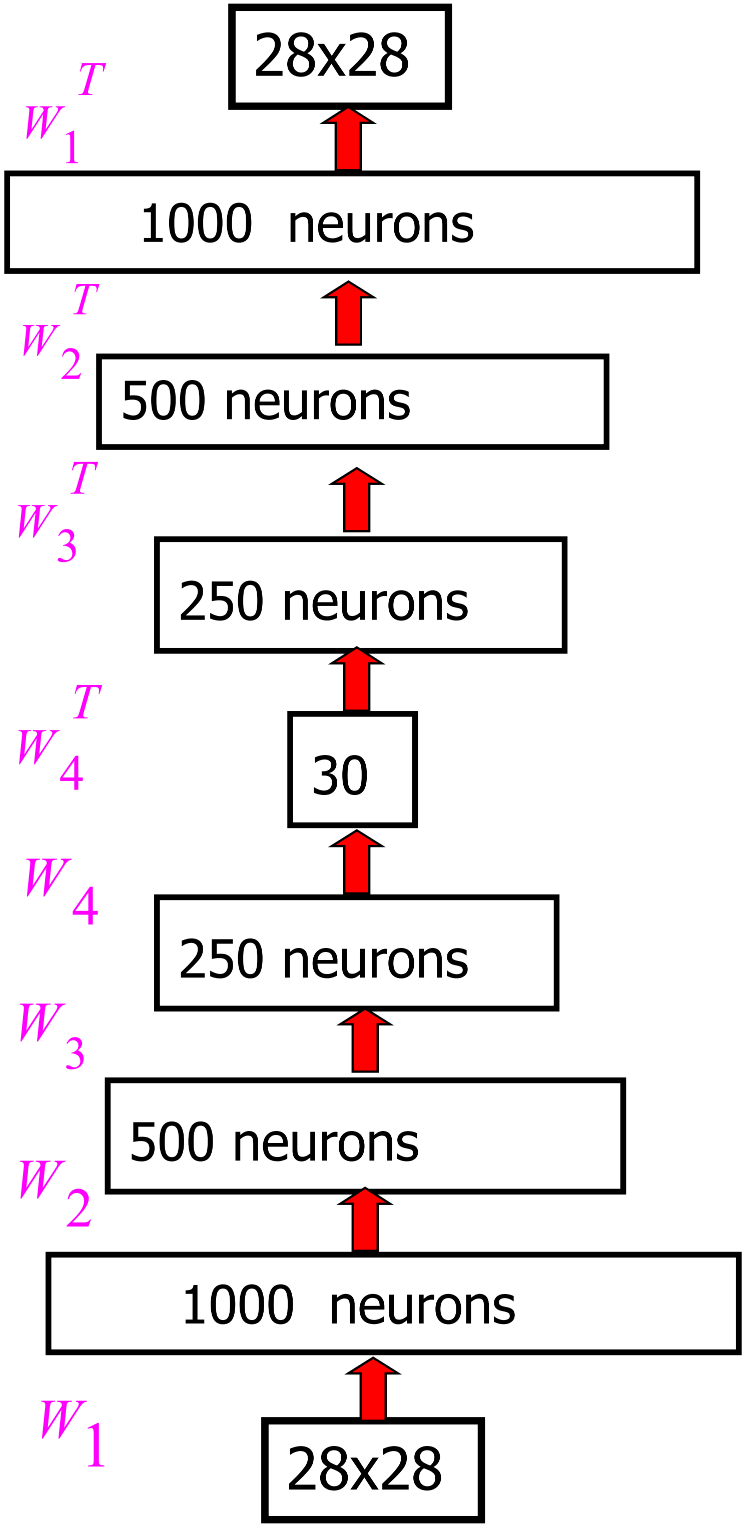
NEURAL NETWORKS AND FEATURE GENERATION/SELECTION

- Using neural networks for feature generation and selection.
- Possible solution is via the so-called auto-associative networks.
- A network is employed having m input and m output nodes and a single hidden layer with l nodes with linear activations.
- An extension of this idea is to use three hidden layers.
- Pruning a neural network is a form of feature selection integrated into the classifier design stage.

Deep Autoencoders

(Ruslan Salakhutdinov)

- They always looked like a really nice way to do non-linear dimensionality reduction:
  - But it is very difficult to optimize deep autoencoders using backpropagation.
- We now have a much better way to optimize them:
  - First train a stack of 4 RBM's
  - Then "unroll" them.
  - Then fine-tune with backprop.



- Hints from Generalization Theory. حذف تا آخر اسلایدها

Generalization theory aims at providing general bounds that relate the error performance of a classifier with the number of training points, N, on one hand, and some classifier dependent parameters, on the other. Up to now, the classifier dependent parameters that we considered were the number of its free parameters and the dimensionality, ℓ, of the subspace, in which the classifier operates. (ℓ also affects the number of free parameters).

- Definitions
  - Let the classifier be a binary one, i.e.,

$$f : \Re^{\ell} \rightarrow \{0,1\}$$

  - Let F be the set of all functions f that can be realized by the adopted classifier (e.g., changing the synapses of a given neural network different functions are implemented).

- The shatter coefficient $S(F,N)$ of the class F is defined as:

the **maximum** number of dichotomies of N points that can be formed by the functions in $\boldsymbol{F}$.

- The maximum possible number of dichotomies is 2^N. However, NOT ALL dichotomies can be realized by the set of functions in $\boldsymbol{F}$.

- The Vapnik – Chervonenkis (VC) dimension of a class $\boldsymbol{F}$ is the largest integer k for which $S(F,k) = 2^k$. If $S(F,N)=2^N, \ \forall N,$

  we say that the VC dimension is <u>infinite</u>.

  - That is, VC is the integer for which the class of functions F can achieve all possible dichotomies, 2^k.
  - It is easily seen that the VC dimension of the single perceptron class, operating in the ℓ-dimensional space, is $\ell+1$.

- It can be shown that

$$S(F,N) \leq N^{V_c} + 1$$

V_c: the VC dimension of the class.

That is, the shatter coefficient is either 2^N (the maximum possible number of dichotomies) or it is upper bounded, as suggested by the above inequality.

In words, for finite V_c and large enough N, the **shatter coefficient is bounded by a polynomial growth**.

  - Note that in order to have a polynomial growth of the shatter coefficient, N must be larger than the V_c dimension.

- The V_c dimension can be considered as an **intrinsic capacity** of the classifier, and, as we will soon see, only if the number of training vectors **exceeds** this number sufficiently, we can expect good generalization performance.

- The V_c dimension may or may **not** be related to the dimension ℓ and the number of free parameters.
  - Perceptron: $V_c = \ell + 1$
  - Multilayer perceptron with hard limiting activation function

$$2\left[\frac{k_n^h}{2}\right]\ell \le V_c \le 2k_w \log_2(ek_n)$$

where k_n^h is the total number of hidden layer nodes, k_n the total number of nodes, and k_w the total number of weights ℓ the input space dimension, e the base of the natural logarithm, and [·] the floor operator that gives the largest integer less than its argument.

Support Vector Machines: A Last Touch

- Let $\underline{x}_i$ be a training data sample and assume that

$$\left\|\underline{x}_i\right\| \le r, i = 1,\ 2,\ \ldots,\ N$$

Let also a hyperplane such that $\|w\|^2 \leq c$

and $y_i\left(\underline{w}^T \underline{x}_i + b\right) \geq 1$

(i.e., the constraints we met in the SVM formulation). Then

$$V_c \leq \min\left(r^2 c, \ell\right) + 1$$

That is, by controlling the constant c, the V_c of the linear classifier can be less than ℓ. In other words, V_c can be controlled independently of the dimension.

Thus, by minimizing $\|w\|^2$ in the SVM, one attempts to keep V_c as small as possible. Moreover, one can achieve **finite** V_c dimension, **even for infinite dimensional spaces**. This is an explanation of the potential for good generalization performance of the SVM's.

❖ Generalization Performance

- Let $P_e^N(f)$ be the error rate of classifier f, based on the N training points, also known as empirical error.

- Let $P_e(f)$ be the true error probability of f (also known as generalization error), when f is confronted with data outside the finite training set.

- Let P_e be the minimum error probability that can be attained over ALL functions in the set F.

- Let f^* be the function resulting by minimizing the empirical (over the finite training set) error function.

- It can be shown that:
  - $$\operatorname{prob}\left\{\max_{f\in F}\left(P_e^N(f)-P_e(f)\right)>\varepsilon\right\}\leq 8S(F,N)\exp\left(-\frac{N\varepsilon^2}{32}\right)$$
  - $$\operatorname{prob}\left\{P_e(f^*)-P_e>\varepsilon\right\}\leq 8S(F,N)\exp\left(-\frac{N\varepsilon^2}{128}\right)$$
  - Taking into account that for finite V_c dimension, the growth of $S(F,N)$ is only polynomial, the above bounds tell us that for a large N :
    - ✓ $P_e^N(f)$ is close to $P_e(f)$, with high probability.
    - ✓ $P_e(f^*)$ is close to P_e, with high probability.

Some more useful bounds

- The minimum number of points, $N(\varepsilon,\rho)$, that guarantees, with high probability, a good generalization error performance is given by

$$N(\varepsilon,\rho) \le \max\left\{ \frac{k_1 V_c}{\varepsilon^2} \ln \frac{k_2 V_c}{\varepsilon^2}, \frac{k_3}{\varepsilon^2} \ln \frac{8}{\rho} \right\}$$

- That is, for any $N \ge N(\varepsilon,\rho)$

$$\text{prob}\left\{ P_e(f) - P_e > \varepsilon \right\} \le \rho$$

Where, k_1, k_2, k_3 are constants. In words, for $N \ge N(\varepsilon,\rho)$ the performance of the classifier is guaranteed, with high probability, to be close to the optimal classifier in the class F. $N(\varepsilon,\rho)$ is known as the sample complexity.

- With a probability of at least $1-\rho$ the following bound holds:

$$P_e(f) \le P_e^N(f) + \Phi\left(\frac{V_c}{N}\right)$$

- where

$$\Phi\left(\frac{V_c}{N}\right) = \sqrt{\frac{V_c\left(\ln\left(\frac{2N}{V_c}+1\right) - \ln\left(\frac{\rho}{4}\right)\right)}{N}}$$

- Remark: Observe that all the bounds given so far are:
  - Dimension free
  - Distribution free

❖ * Model Complexity vs Performance

This issue has already been touched in the form of overfitting in neural networks modeling and in the form of bias-variance dilemma. A different perspective of the issue is dealt below.

➢ Structural Risk Minimization (SRM)

- Let P_B be he Bayesian error probability for a given task.
- Let $P_e(f^*)$ be the true (generalization) error of an optimally design classifier f^*, from class F, given a finite training set.

$$P_e(f^*) - P_B = \left(P_e(f^*) - P_e\right) + \left(P_e - P_B\right)$$

P_e is the minimum error attainable in F

– If the class F is small, then the first term is expected to be small and the second term is expected to be large. The opposite is true when the class F is large.

- Let $F^{(1)}, F^{(2)}, \dots$ be a sequence of nested classes:

$$F^{(1)} \subset F^{(2)} \subset \dots$$

with increasing, yet finite V_c dimensions.

$$V_{c,F^{(1)}} \le V_{c,F^{(2)}} \le \dots$$

Also, let

$$\lim_{i\to\infty} \inf_{f\in F^{(i)}} P_e(f) = P_B$$

For each N and class of functions $F^{(i)}$, i=1, 2, …, compute the optimum $f^*_{N,i}$, with respect to the empirical error. Then from all these classifiers choose the one than minimizes, over all i, the upper bound in:

$$\tilde{P}_e(f^*_{\mathrm{N},i}) \le P_e^N(f^*_{\mathrm{N},i}) + \Phi\left(\frac{V_{c,F^{(i)}}}{N}\right)$$

That is,

$$f^*_N = \arg\min_i \left[P_e^N(f^*_{\mathrm{N},i}) + \Phi\left(\frac{V_{c,F^{(i)}}}{N}\right)\right]$$

- Then, as $\mathrm{N} \to \infty$

$$\boxed{P_e(f_N^*) \to P_B}$$

  - The term $$\Phi\left(\frac{V_{c,F^{(i)}}}{N}\right)$$

  in the minimized bound is a complexity penalty term. If the classifier model is **simple** the penalty term is **small** but the empirical error term

$$P_e^N(f_{\mathrm{N},i}^*)$$

  - will be **large**. The opposite is true for complex models.

- The SRM criterion aims at achieving the best trade-off between performance and complexity.

➢ * Bayesian Information Criterion (BIC)

Let N the size of the training set, $\underline{\theta}_m$ the vector of the unknown parameters of the classifier, K_m the dimensionality of $\underline{\theta}_m$, and m runs over all possible models.

- The BIC criterion chooses the model by minimizing:

$$BIC = -2L\left(\hat{\underline{\theta}}_m\right) + K_m \ln N$$

  - $L\left(\hat{\underline{\theta}}_m\right)$ is the log-likelihood computed at the ML estimate $\hat{\underline{\theta}}_m$, and it is the performance index.
  - $K_m \ln N$ is the model complexity term.

- Akaike Information Criterion:

$$AIC = -2L\left(\hat{\underline{\theta}}_m\right) + 2K_m$$