# MSPR 6: Classification

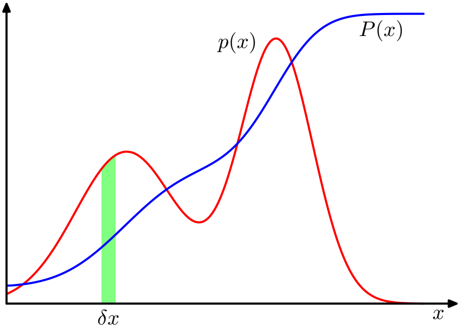
Dr. Hendrik Purwins

AAU CPH

October 8, 2015

- *The part on discriminant analysis follows closely Prof. Ulrich Kockelkorn: (emeritus, Berlin Institute of Technology): Lecture Notes Multivariate Statistics. (unpublished)*
- *As a background van der Heijden et al.: Classification, Parameter Estimation and State Estimation, Chapter 2 Detection and Classification p.13-31 can be read. Although the PRTools examples are from that book, the lecture uses another terminology than in the book and presents the topics in a different order.*

# Outline



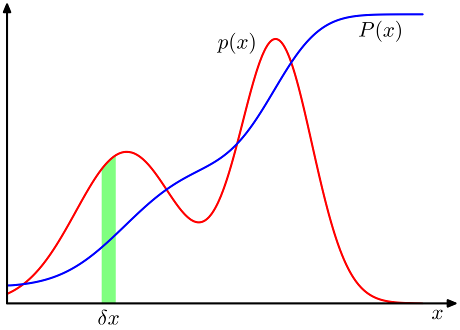

The probability of $x$ is given by probability distribution $p(x)$.

```
mu = 0; sigma = 2;
pd = makedist('Normal',mu,
    sigma);
```

The probability of $x \in (-\infty, z)$ is given by the *cumulative distribution function* (CDF) $P(x)$ Matlab `y = cdf(pd,x)` Consider a probability in the range $\delta x$ E.g. that a normally distributed random variable falls within $\pm\sigma$ :

```
y = cdf(pd,[a b])}
x = [-sigma sigma]; y =
    cdf(pd,x)y(2)-y(1)
```

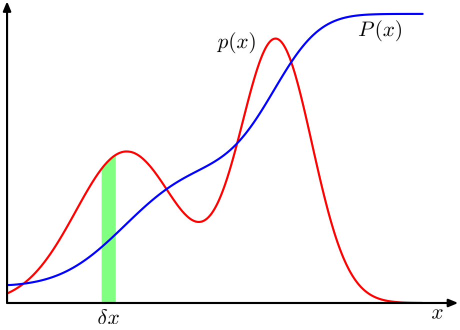
Result: Probability: 0.6827




- Let $p(x)\delta x$ be the probability of $x$ falling in the interval $(x, x + \delta x)$ for $\delta x \to 0$
- Then $p(x)$ is called *probability density function* (PDF) over $x$ with the properities

$$p(x) \geq 0$$

$$\int_{-\infty}^{\infty} p(x) dx = 1$$




- The probability of $x \in (a, b)$ is given by

$$p(x \in (a, b)) = \int_a^b p(x)dx$$

- The probability of $x \in (-\infty, z)$ is given by the *cumulative distribution function* (CDF)

$$P(x, z) = \int_{-\infty}^{z} p(x)dx$$
$$P'(z) = p(x)$$

# Multi-variate Gaussians

- The multivariate Gaussian distribution for a J-dimensional variable $\mathbf{x}$ is given by

$$\mathcal{N}(\mathbf{x}|\mu, \mathbf{\Sigma}) = \frac{1}{2\pi^{J/2} \det \mathbf{\Sigma}^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu)^T \mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)}$$

- Defining parameters: $\mu$ and $\mathbf{\Sigma}$
- Mean and Covariance
  - Mean (vector) $= \mu$ ($J$ dimensional)
  - Covariance (matrix) $= \mathbf{\Sigma}$ ($J \times J$)
  - $\det \mathbf{\Sigma}$ denotes the determinant of $\mathbf{\Sigma}$

# Classification: Applications and Questions

- Application examples:
  - Description (classification of customer types that buy brands/ no-name products based on buys in those products)
  - Prediction (avalanche warning based on snow analysis)
  - Decision (credit risk of a client based on income, employment duration, number of credit cards)
- Questions:
  - Find reasonable decision rule to optimally separate classes
  - Feature vectors: feature construction (e.g. MFSSs for sound), dimension reduction (e.g. eigenvalue decomposition), feature selection (later in the lecture)
  - Do we have pervious knowledge?
    - We know class labels for the training set (*Supervised learning*) $\Rightarrow$ apply discriminant analysis
    - We have no class labels (*Unsupervised learning*) $\Rightarrow$ Cluster analysis
  - Error, costs, quality of a decision for a class
  - Division of data in training set $\leftrightarrow$ test set

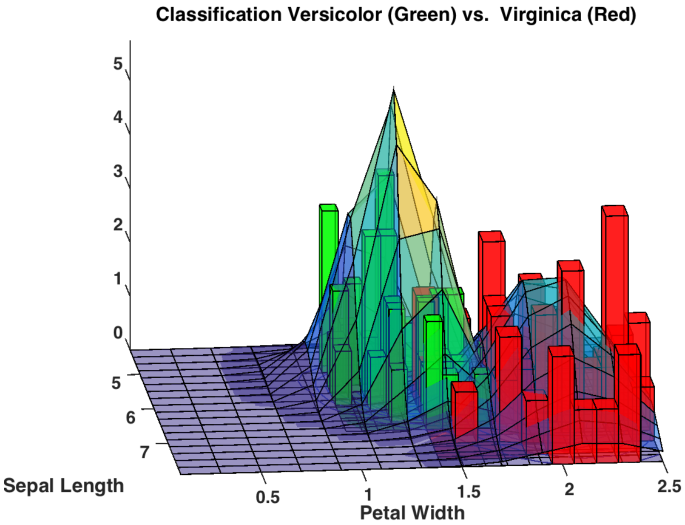
# Decision Rule for Classification

- Assume a data point $\mathbf{x}$ can belong to two classes: $1, 0$.
- The probability distribution of each class modelled by a Gaussian with mean $\mu_0$ ($\mu_1$) and covariance $\Sigma_0$ ($\Sigma_1$) .
- Predict the class $\mathbf{x}$ according to the decision rule:

$$d(x) = \begin{cases} 1 & \text{if } \mathcal{N}(\mathbf{x}|\mu_\mathbf{1}, \mathbf{\Sigma}_1) > \mathcal{N}(\mathbf{x}|\mu_\mathbf{0}, \mathbf{\Sigma}_0) \\ 0 & \text{else} \end{cases}$$

# Gaussian Fit for Predicting the Iris Type Based on Sepal Length and Petal Width only

- We had fitted two Gaussians $\mathcal{N}(\mathbf{x}|\bar{\mathbf{x}}_{virg}, \mathbf{S}_{virg}), \mathcal{N}(\mathbf{x}|\bar{\mathbf{x}}_{vers}, \mathbf{S}_{vers})$ to the points 6-50 of Virginica and Versicolor Iris data ('Sepal Length' and 'Petal Width').
- The first 5 points of Virginica and Versicolor iris data ('Sepal Length' and 'Petal Width') have not been used for the fitting.
- Let us use the fitted Gaussians, to determine whether these points belong to Virginica or Versicolor according to the rule: If for a point $\mathbf{x}$ $\mathcal{N}(\mathbf{x}|\bar{\mathbf{x}}_{virg}, \mathbf{S}_{virg}) > \mathcal{N}(\mathbf{x}|\bar{\mathbf{x}}_{vers}, \mathbf{S}_{vers})$ we predict its class to be Virginica, otherwise Versicolor.
- Let us count the wrong predictions.


Classification Versicolor (Green) vs. Virginica (Red)
5
4
3
2
1
0
5
6
7
Sepal Length
0.5
1
1.5
2
2.5
Petal Width

```
sig_virginica=cov(X(idx_virginica_tr,:));
sig_versicolor=cov(X(idx_versicolor_tr,:));
idx_virginica_tst=idx_virginica(1:5);
idx_versicolor_tst=idx_versicolor(1:5);
F_versicolor_tst = mvnpdf(X([idx_virginica_tst;
    idx_versicolor_tst],:),...
      mean_versicolor,sig_versicolor);
F_virginica_tst = mvnpdf(X([idx_virginica_tst;
    idx_versicolor_tst],:),...
      mean_virginica,sig_virginica);
[F_virginica_tst'; F_versicolor_tst']
%ans =
%   0.18  0.43  0.69   0.66   0.69  0.03  0.16  0.09  0.02  0.15
%   0.00  0.00  0.00   0.10   0.00  0.08  0.98  0.19  1.30  0.80
```

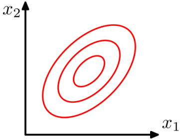
Perfect Prediction!

# Parameters to Estimate

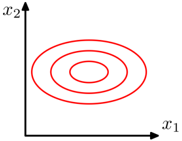
- In its most general form, the multi-variate Gaussian of $J$ dimensions has
  - $J(J+1)/2$ parameters for the covariance matrix
  - $J$ parameters for the mean vector.
  - All this for $K$ classes
- The number of parameters increases quadratically with $J$ and hence poses a problem both in parameter estimation and matrix inversion.

# Reduction of Parameters to be Estimated

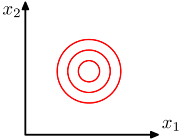
How to simplifying this problem in a classification context




- Estimate just one covariance matrix for all classes that is the weighted mean of each classes individual covariance matrix.




- Assume diagonal covariance matrix ($2 \times J$ parameters).
- Assume equal covariance $\sigma = \sigma_k$for all classes $k$: $\mathbf{\Sigma} = \sigma^2 \mathbf{I}$ ($(J+1)$ parameters). (minimum distance classifier )




The more parameters, the more complex/more flexible the classifier, but we need more data to have reliable estimates. The less parameters the less

# Linear Discriminant Analysis (=Minimal Mahalanobis Distance Classifier)

Pooled covariance matrix for $l$ observations belonging to $K$ classes, $l_k$ in each class (in case all classes have the same size):

$$\mathbf{S} = \frac{1}{K} \sum_{k=1}^{K} S_k$$

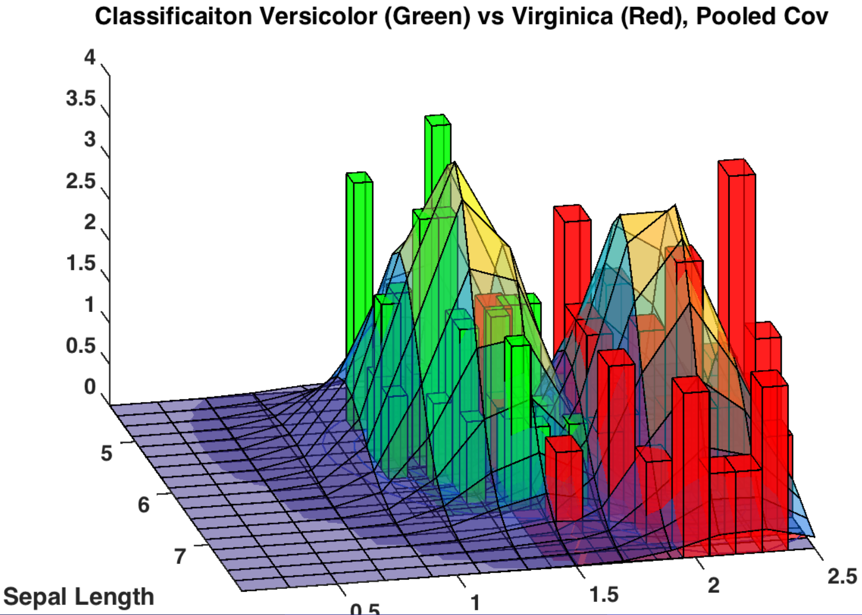
## Class Assignment

*Calculate the pooled covariance matrix for the species Virginicia and Versicolor based on the features 'Sepal Length' and 'Petal Width' using the perviously estimated class sample covariances* $\mathbf{S}_{virg}, \mathbf{S}_{vers}$ *Then perform the classification based on the decision rule:*

$$d(x) = \begin{cases} \textit{Virginica} & \textit{if } \mathcal{N}(\mathbf{x}|\bar{\mathbf{x}}_{virg}, \mathbf{S}) > \mathcal{N}(\mathbf{x}|\bar{\mathbf{x}}_{\mathbf{vers}}, \mathbf{S}) \\ \textit{Versicolor} & \textit{else} \end{cases}$$

```
sig_pool=(sig_versicolor +sig_virginica)/2;
F_versicolor_tst = mvnpdf(X([idx_virginica_tst; idx_versicolor_tst],:),mean_versicolor,sig_pool);
F_virginica_tst = mvnpdf(X([idx_virginica_tst; idx_versicolor_tst],:),mean_virginica,sig_pool);
true_labs=[ones(1,no_tst) zeros(1,no_tst)];
pred_labs= (F_virginica_tst>F_versicolor_tst)';
errors=sum(true_labs~=pred_labs)
```

0 Errors



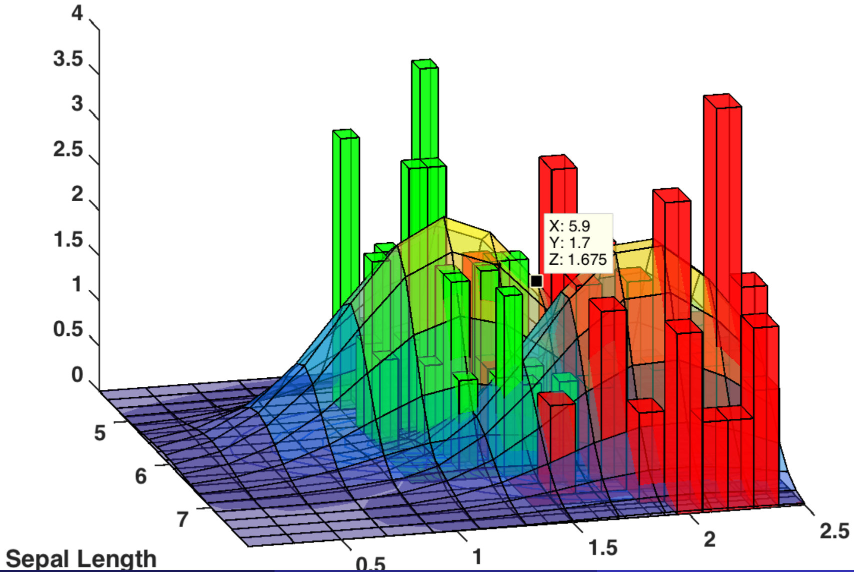
Classificaiton Versicolor (Green) vs Virginica (Red), Pooled Cov
4
3.5
3
2.5
2
1.5
1
0.5
0
5
6
7
Sepal Length
0.5
1
1.5
2
2.5

# Minimum Distance Classifier

## Class Assignment

- *Fit two Gaussians* $\mathcal{N}(\mathbf{x}|\bar{\mathbf{x}}_{virg}, \sigma^2\mathbf{I})\mathcal{N}(\mathbf{x}|\bar{\mathbf{x}}_{vers}, \sigma^2\mathbf{I})$ *using a covariance matrix of type* $\sigma^2\mathbf{I}$ *with* $\sigma^2$ *being the mean variance of for 'Sepal Length' and 'Petal Width' on the diagonal of the pooled covariance matrix* $\mathbf{S}$*. Use to the features 'Sepal Length' and 'Petal Width' of instances 6-50 of the Virginica and Versicolor Iris data.*
- *Use the fitted Gaussians, to determine whether these points belong to Virginica or Versicolor according to the rule: If for a point* $\mathbf{x}$ $\mathcal{N}(\mathbf{x}|\bar{\mathbf{x}}_{virg}, \sigma^2\mathbf{I}) > \mathcal{N}(\mathbf{x}|\bar{\mathbf{x}}_{vers}, \sigma^2\mathbf{I})$ *we predict its class to be Virginica, otherwise Versicolor.*
- *Count the wrong predictions for the first 5 points of Virginica and Versicolor iris data.*

Classificaiton Versicolor (Green) vs Virginica (Red), cov=$\sigma^2$ I

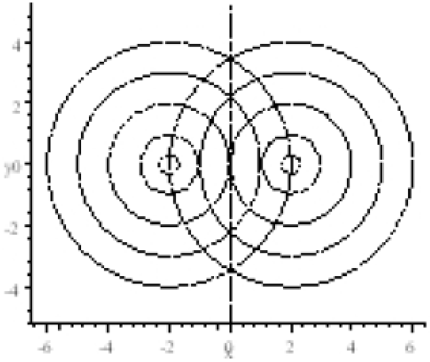


# Solution

```
sig_pool=(sig_versicolor +sig_virginica)/2;
sig_I=mean(diag(sig_pool))*diag(ones(2,1));
F_versicolor_tst = mvnpdf(X([idx_virginica_tst;
    idx_versicolor_tst],:),mean_versicolor,sig_I);
F_virginica_tst = mvnpdf(X([idx_virginica_tst;
    idx_versicolor_tst],:),mean_virginica,sig_I);
true_labs=[ones(1,no_tst) zeros(1,no_tst)];
pred_labs= (F_virginica_tst>F_versicolor_tst)';
errors=sum(true_labs~=pred_labs)
```

4 Errors!

# Geometrical Visualization of Euclidean Distance




Points of equal distance to centers $\mu_1 = \begin{bmatrix} -2 \\ 0 \end{bmatrix}$ and $\mu_2 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$

- With Euclidean distances, all points $\mathbf{x}$ with constant distance $r$ to center $\mu$ lie on the circle

$$\|\mathbf{x} - \mu\|^2 = r^2$$

with radius $r$ around $\mu$.

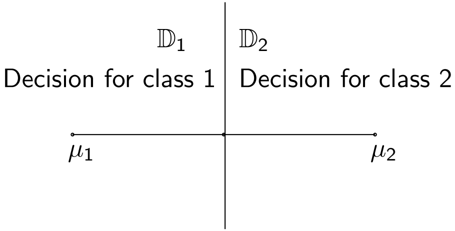
- All points with equal distance to two centers $\mu_1$ and $\mu_2$ lie on a line.

# Discriminant Analysis (DA)

- To element with feature vector $\mathbf{x}$, assign class $j$ with closest class mean $\mu_j$:

$$\mathbb{D}_j = \{\mathbf{x} \mid \|\mathbf{x} - \mu_k\| > \|\mathbf{x} - \mu_j\| \quad \text{for} \quad 1 \leq k \leq K, k \neq j\}.$$

- For a two-dimensional feature vector $\mathbf{x} \in \mathbb{R}^2$, the class border between $\mathbb{D}_1$ and $\mathbb{D}_2$ is the vertical line rectangular to the connection line of both class means $\mu_1$ and $\mu_2$, crossing this line in the middle:




- On the borders between $\mathbb{D}_1$ and $\mathbb{D}_2$ decision for one class is arbitrary $\Rightarrow$ Decision can be randomized

# Decision Rule for 3 Classes

Assuming equal priors, for 3 classes there are three decision regions $\mathbb{D}_1, \mathbb{D}_2, \mathbb{D}_3$ in right angles to the connection lines between pairs of means $\mu_1, \mu_2, \mu_3$:

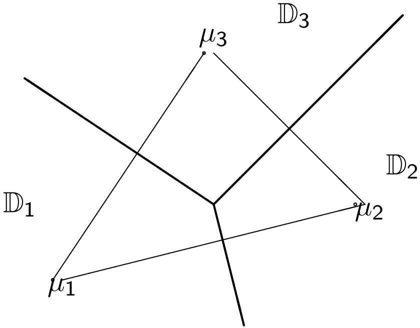


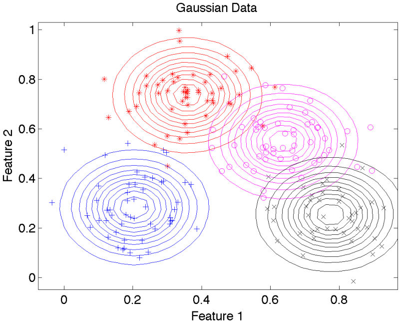

Image for equal costs and equal priors for each class.

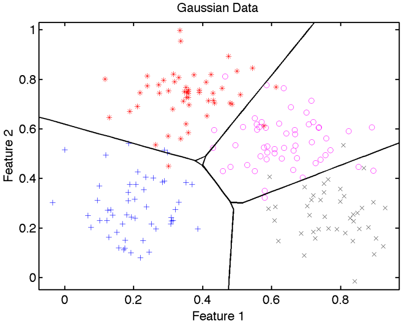
# PRTools Example: Euclidean Distance in Minimum Distance Classifier

CPESE book p. 31

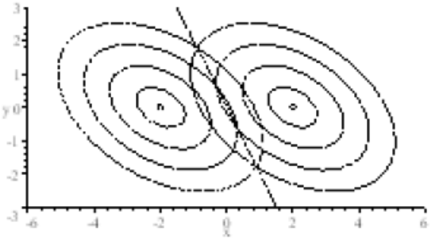
```
mus = [0.2 0.3; 0.35 0.75; 0.65 0.55; 0.8 0.25];
C = 0.01*eye(2); z = gauss(200,mus,C);
% Normal densities, uncorrelated noise with equal variances
w = nmsc(z);
figure (1); scatterd (z); hold on; plotm (w);
figure (2); scatterd (z); hold on; plotc (w);
```

# PRTools Example: Euclidean Metric II



# Geometrical Visualization of Malhalanobis Distance



Points of equal distance to centers $\mu_1 = \begin{bmatrix} -2 \\ 0 \end{bmatrix}$ and $\mu_2 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$ with the Mahalanobis distance with $\mathbf{A} = \begin{bmatrix} 2 & 1 \\ 1 & 3 \end{bmatrix}$

- With respect to the Mahalanobis distance all points $\mathbf{x}$ with constant distance $r$ to center $\mu$ lie on the ellipse

$$(\mathbf{x} - \mu)^T \mathbf{A} (\mathbf{x} - \mu) = r^2$$

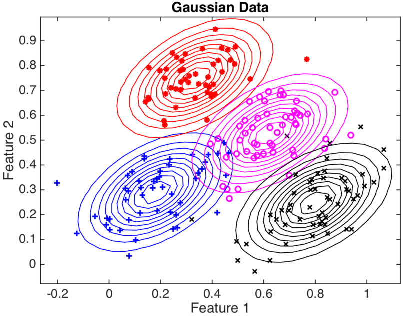
  around $\mu$.
- All points with equal distance to two centers $\mu_1$ and $\mu_2$ still lie on a line.

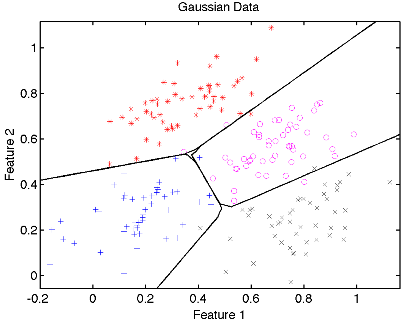
# PRTools Example: Mahalanobis Distance and Minimum Mahalanobis Classifier

CPESE book p. 23

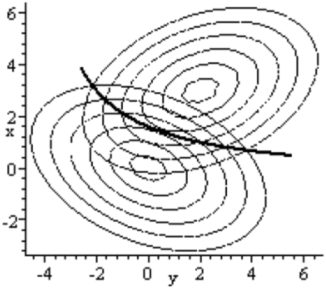
```
mus = [0.2 0.3; 0.35 0.75; 0.65 0.55; 0.8 0.25];
C = [0.018 0.007; 0.007 0.011];
z = gauss(200,mus,C);
w = ldc(z);
% Normal densities, identical covariances
h=figure(1); scatterd(z); hold on; plotm(w);
print_def(h,'ldc_cont_gauss')
g=figure(2); scatterd(z); hold on; plotc(w);
```

# PRTools Example: Mahalanobis Distance and Minimum Mahalanobis Classifier



# Borders of Discriminant Regions for Quadratic Classifier




- For $r = 1, 2, 3, 4, 5, 6$, ellipses

$$(\mathbf{x} - \mu_k)^T \mathbf{S}_k^{-1} (\mathbf{x} - \mu_k) = r^2$$

are shown for $\mu_1 = 0$ und $\mu_2 = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$, $\mathbf{S}_1 = \begin{pmatrix} 4 & 1 \\ 1 & 2 \end{pmatrix}$ and $\mathbf{S}_2 = \begin{pmatrix} 4 & -1 \\ -1 & 2 \end{pmatrix}$

- Borders of discrimination regions defined by:

$$\begin{pmatrix} x-3, & y-2 \end{pmatrix} \begin{pmatrix} 4 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x-3 \\ y-2 \end{pmatrix} = \begin{pmatrix} x, & y \end{pmatrix} \begin{pmatrix} 4 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}.$$

- Hyperbola: $y = -2\frac{5x-8}{2x+1}$.

```
X=[adata{1} adata{4}];
species=adata{5};
species=strcmp(species,'Iris-setosa')+2*strcmp(species,'Iris-virginica')...
    +3*strcmp(species,'Iris-versicolor');
idx_virginica=find(species==2); idx_versicolor=find(species==3);
idx_virginica_tr=idx_virginica(no_tst+1:50); idx_versicolor_tr=idx_versicolor(no_tst+1:50);
idx_virginica_tst=idx_virginica(1:no_tst); idx_versicolor_tst=idx_versicolor(1:no_tst);
idx_tr=[idx_virginica_tr; idx_versicolor_tr]; tst_idx=[idx_virginica_tst; idx_versicolor_tst];
priris_tr=prdataset(X([idx_tr],:),species([idx_tr]));
priris_tst=prdataset(X([tst_idx],:),species([tst_idx]));
priris_w=ldc(priris_tr); pred_lab=priris_tst*priris_w*labeld;
sum(pred_lab~=species(tst_idx))
g=figure; scatterd (priris_tr); hold on; plotm (priris_w);
h=figure; scatterd (priris_tr); hold on; plotc (priris_w);
```

## Class Assignment

*Perform Quadratic classification on the same trainings set and predict the labels of the same left out test instances.*

# Bayesian Decision Rule in Binary Classification

- Assume a data point **x** can belong to two classes: $1, 0$.
- The probability distribution of each class modelled by a Gaussian with mean $\mu_0$ ($\mu_1$) and covariance $\Sigma_0$ ($\Sigma_1$) .
- With class prior probabilities $P(\mu_1, \sigma_1), P(\mu_2, \sigma_2)$, predict the class of **x** according to the decision rule:

$$d(x) = \begin{cases} 1 & \text{if } \mathcal{N}(\mathbf{x}|\mu_{\mathbf{1}}, \mathbf{\Sigma}_1)P(\mu_1, \sigma_1) > \mathcal{N}(\mathbf{x}|\mu_{\mathbf{0}}, \mathbf{\Sigma}_0)P(\mu_0, \sigma_0) \\ 0 & \text{else} \end{cases}$$

- If no prior knowledge is known, the relative size of class $k$ can be used as priors $P(\mu_k, \sigma_k)$. If the classes are equal size, the priors need not be considered.

# Discriminant Analysis as Bayesian Classification

- Bayes' Theorem for class $k$ and data point $\mathbf{x}$:

$$P(k|\mathbf{x}) = \frac{P(\mathbf{x}|k)P(k)}{P(\mathbf{x})}$$

- $P(k)$: prior
- $P(\mathbf{x}|k)$: likelihood (typically $\mathcal{N}(\mathbf{x}|\mu_{\mathbf{k}}, \mathbf{\Sigma}_k)$
- $P(k|\mathbf{x})$: posterior
- Multiclass classification rule:

$$k(\mathbf{x}) = \mathrm{argmax}_{1\leq 1\leq K} P(\mathbf{x}|k)P(k) = \mathrm{argmax}_{1\leq 1\leq K} \mathcal{N}(\mathbf{x}|\mu_{\mathbf{k}}, \mathbf{\Sigma}_k)P(k)$$

## Class Assignment

*How do classification borders change if one class is larger than the other?*

# Summary: Minimum Distance Classifier (=Nearest Mean Classifier)

- Theoretical assumption: Data of each class are drawn from a radialsymmetric Gaussian distribution with the identical covariance matrix of the form $\mathbf{\Sigma} = \sigma^2 \mathbf{I}$.
- Number of parameters to estimate: just $K \cdot J$ for the $K$ mean vectors ($K$: number of classes, $J$ number of features)
- Classification borders: straight line(s) vertical to the connection between class means.
- Classification criteria: $\mathbf{x}$ is assigned to the class $k$, whose mean $\mu_k$ is closest to it according to the Euclidean distance: $\|\mathbf{x} - \mu_k\|$.
- Matlab PRtools:`w=nmsc(z)`

# Summary: Minimum Mahalanobis Distance Classifier (=Linear Discriminant Analysis)

- Theoretical assumption: Data of each class are drawn from a Gaussian distribution with identical covariance matrix $\Sigma$ of any form.
- Number of parameters to estimate: $K \cdot J$ for the $K$ mean vectors + $J(J+1)/2$ for the covariance matrix ($K$: number of classes, $J$ number of features)
- Classification borders: straight line(s).
- Classification criteria: $\mathbf{x}$ is assigned to the class, whose mean is closest to it according to the Mahalanobis distance $(\mathbf{x} - \mu_k)^T \Sigma^{-1} (\mathbf{x} - \mu_k)$.
- Matlab PRtools:`w=ldc(z)`

# Summary: Quadratic Classifier (=Quadratic Discriminant Analysis)

- Theoretical assumption: Data of each class $k$ are drawn from a Gaussian distribution with its class-specific covariance matrices $\Sigma_k$ of any form.
- Number of parameters to estimate: $K \cdot J$ for the $K$ mean vectors + $J(J+1)/2 \cdot K$ for the covariance matrices ($K$: number of classes, $J$ number of features)
- Classification borders: hyperbolas.
- Classification criteria: $\mathbf{x}$ is assigned to class $k$, whose mean $\mu_k$ is closest to it according to the Mahalanobis distance $(\mathbf{x} - \mu_k)^T \Sigma_k^{-1} (\mathbf{x} - \mu_k)$.
- Matlab PRtools:`w=qdc(z)`

- In practice, often these classifiers are used, even if the theoretical assumptions are not strictly fulfilled.
- In practice, often regularization (with regularization parameter $\lambda$ yields a compromise between quadratic classifier and minimum Mahalanobis classifier using the sample covariance matrices $S_k$ for $K$ classes:

$$\mathbf{S}_k^{reg} = (1-\lambda)\mathbf{S}_k + \frac{\lambda}{K}\sum_{k=1}^{K}\mathbf{S}_k$$

## Class Assignment

*How is the classifier called if $\lambda = 0$ and if $\lambda = 1$?*