Probabilistic (Bayes) Classifier

生醫光電所 吳育德

Credit: Chap5. A First Course in Machine Learning, 2ed, Simon Rogers and Mark Girolami, 2017
Pattern Recognition, 4^{th} Ed., S. Theodoridis and K. Koutroumbas, 2014

The General Classification Problem

- Given a set of N training samples, $\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_N}$, each is D-dimensional.

$$\boldsymbol{X} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,D} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,D} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N,1} & x_{N,2} & \cdots & x_{N,D} \end{bmatrix} \text{ or } \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_D^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \cdots & x_D^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{(N)} & x_2^{(N)} & \cdots & x_D^{(N)} \end{bmatrix}$$

- Each $\boldsymbol{x_i}$ is labelled by $t_i \in \{1, 2, \cdots \mathrm{C}\}$

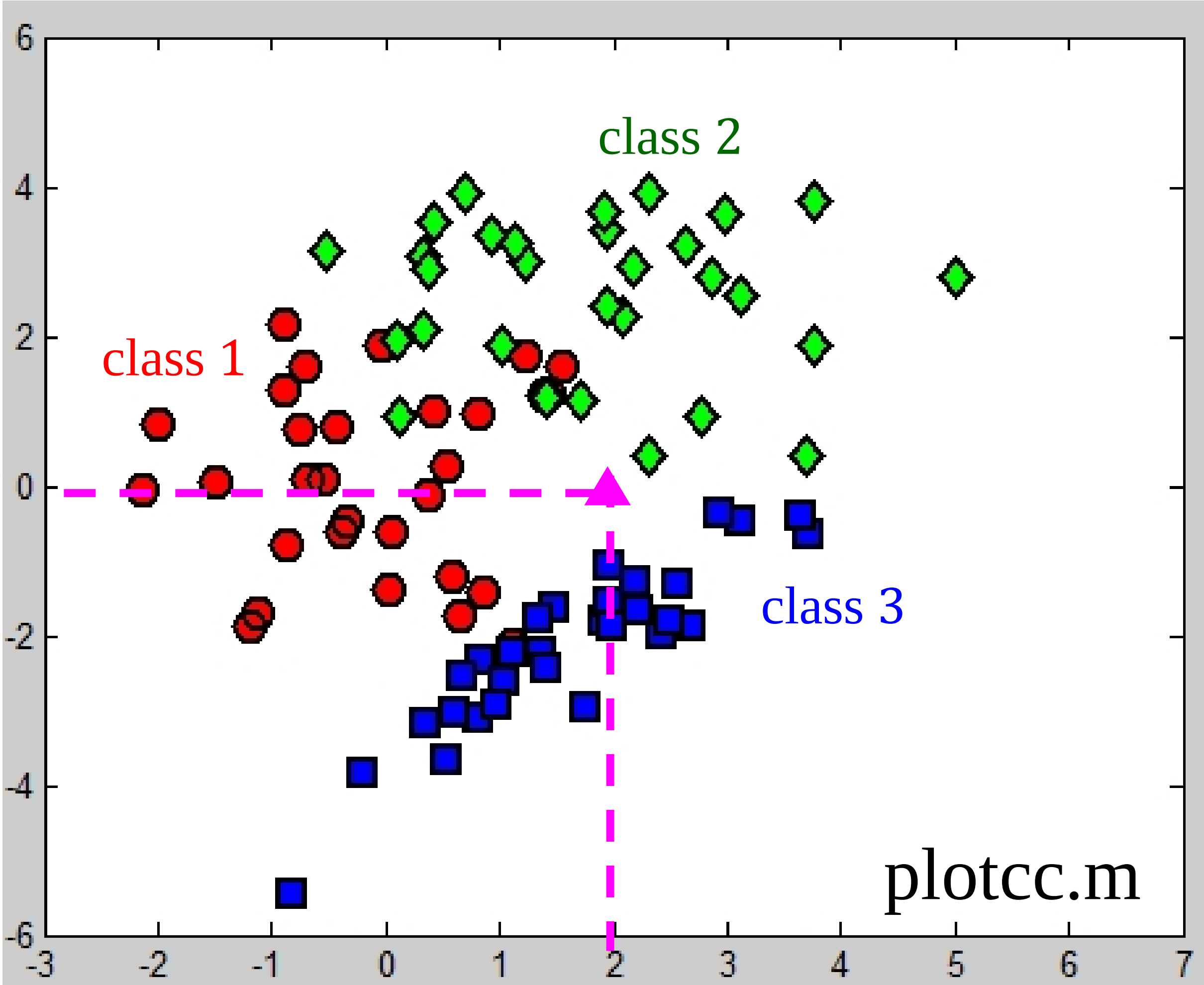
$$\boldsymbol{t} = \begin{bmatrix} t_1 \\ \vdots \\ t_N \end{bmatrix}$$

- Predict the class t_{new} for a new sample $\boldsymbol{x_{new}}$ based on $\boldsymbol{X}$ and $\boldsymbol{t}$.

An example with three classes

[Q]: $\boldsymbol{x}_{\boldsymbol{new}} = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$, $t_{new} = 1?\ 2?\ 3?$ or $p(t_{new} = 1 | \boldsymbol{x}_{\boldsymbol{new}}, \boldsymbol{X}_{\boldsymbol{1}}, \boldsymbol{t}_{\boldsymbol{1}}) =?$

$p(t_{new} = 2 | \boldsymbol{x}_{\boldsymbol{new}}, \boldsymbol{X}_{\boldsymbol{2}}, \boldsymbol{t}_{\boldsymbol{2}}) =?$ $p(t_{new} = 3 | \boldsymbol{x}_{\boldsymbol{new}}, \boldsymbol{X}_{\boldsymbol{3}}, \boldsymbol{t}_{\boldsymbol{3}}) =?$



Assume
$\boldsymbol{X}_{\boldsymbol{1}} = \{\boldsymbol{x}_{\boldsymbol{1}}, \ldots, \boldsymbol{x}_{\boldsymbol{N}_{\boldsymbol{1}}}\}$,
$\boldsymbol{X}_{\boldsymbol{2}} = \{\boldsymbol{x}_{\boldsymbol{1}}, \ldots, \boldsymbol{x}_{\boldsymbol{N}_{\boldsymbol{2}}}\}$ and
$\boldsymbol{X}_{\boldsymbol{3}} = \{\boldsymbol{x}_{\boldsymbol{1}}, \ldots, \boldsymbol{x}_{\boldsymbol{N}_{\boldsymbol{3}}}\}$ are
training data of class 1, 2 and 3 and can be generated by three separate Gaussian distribution

Plot the data (plotcc.m)

```
% Plot the data
cl = unique(t);
col = {'ko','kd','ks'}
fcol = {[1 0 0],[0 1 0],[0 0 1]};
figure(1);
for c = 1:length(cl)
    pos = find(t==cl(c));
    plot(X(pos,1),X(pos,2),col{c},...
        'markersize',10,'linewidth',2,...
        'markerfacecolor',fcol{c});
    hold on
end
xlim([-3 7])
ylim([-6 6])
```

```
>> X(1:5,:)

ans =

    1.1107   -2.1079
   -0.5498    0.0943
   -0.0382    1.8829
    0.0555   -0.6139
    0.5870   -1.2067
```

```
>> t(1:5)

ans =

     1
     1
     1
     1
     1
```

```
>> cl

cl =

     1
     2
     3
```

```
>> find(t==cl(2))

ans =

    31
    32
    33
    34
    35
    36
    37
```

We need to know

- What is the multivariate Gaussian distribution $N(\boldsymbol{x}; \mu, \Sigma)$?

- How to compute the mean μ, and the covariance Σ

→ Using maximizing the likelihood function

→ Empirical mean and covariance

- How to make prediction $p(t_{new} = c|\boldsymbol{x_{new}}, \boldsymbol{X_c}, \boldsymbol{t_c}), c = 1, 2, 3$

→ Using the Bayes' rule

$$p(A|B) = \frac{p(A \cap B)}{p(B)} = \frac{p(B|A)p(A)}{p(B)}$$

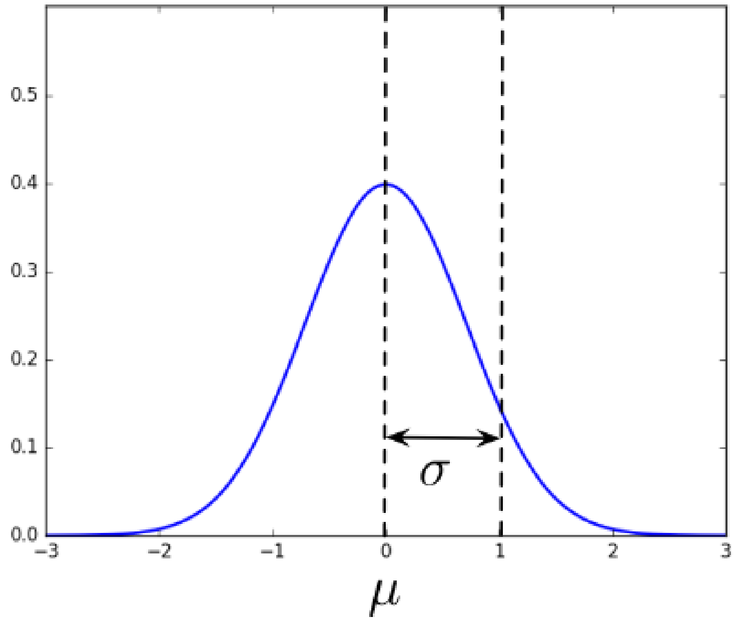
→ $$p(t_{new} = c|\boldsymbol{x_{new}}, \boldsymbol{X}, \boldsymbol{t}) = \frac{p(\boldsymbol{x_{new}}|\, t_{new} = c, \boldsymbol{X_c}, \boldsymbol{t_c})p(t_{new} = c|\boldsymbol{X_c}, \boldsymbol{t_c})}{p(\boldsymbol{x_{new}}|\boldsymbol{X}, \boldsymbol{t})}$$

Gaussian Distribution

Gaussian Distribution

- Recall the Gaussian, or normal, distribution:

$$N(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

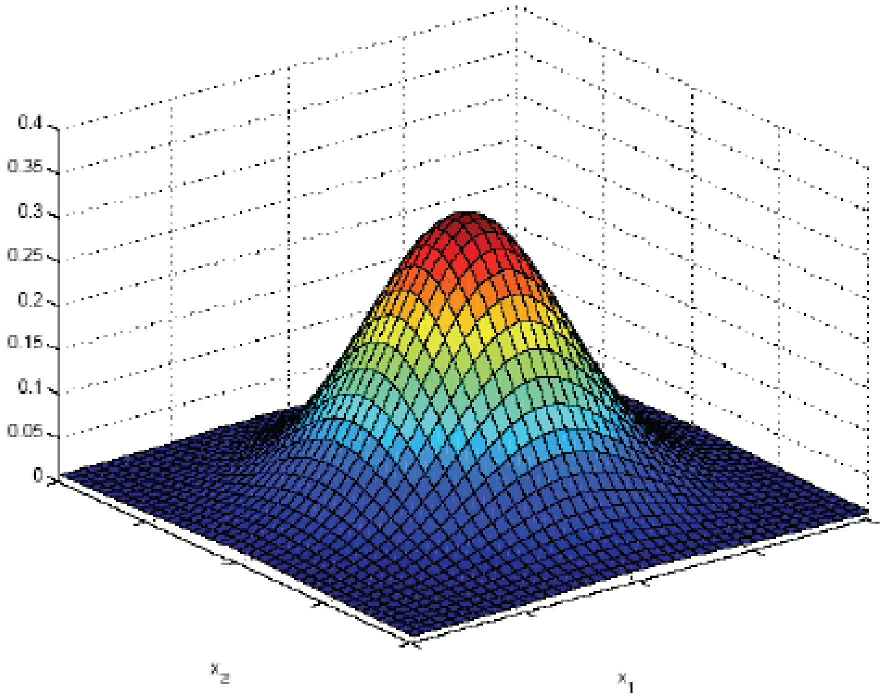


- In machine learning, we use Gaussians a lot because they make the calculations easy.

Multivariate Gaussian Distribution

- Multivariate Gaussian Distribution $\boldsymbol{x} \sim N(\mu, \Sigma)$, or $N(\boldsymbol{x}; \mu, \Sigma)$ is

$$p(\boldsymbol{x}) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp[-\frac{1}{2}\ (\boldsymbol{x} - \mu)^T \Sigma^{-1} (\boldsymbol{x} - \mu)]$$



Multivariate parameters

- Mean: $E[\boldsymbol{x}] = [\mu_1, \cdots, \mu_d]^T$

- Covariance

$$\sum = \mathrm{Cov}(\boldsymbol{x}) = E[(\boldsymbol{x} - \mu)(\boldsymbol{x} - \mu)^T] = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_d^2 \end{bmatrix}$$

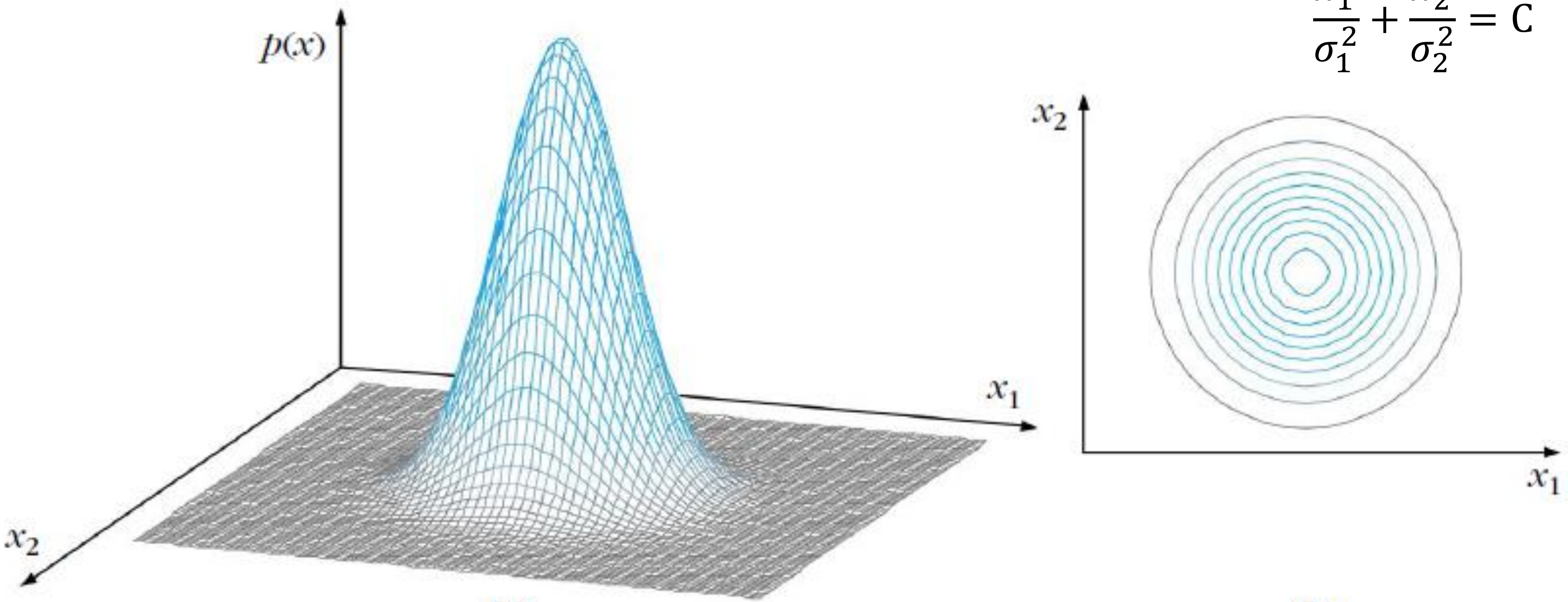
- For Gaussians - all you need to know is mean and covariance

2D Gaussian pdf, diagonal $\sum$ with $\sigma_1^2 = \sigma_2^2$

$$\Sigma = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$$

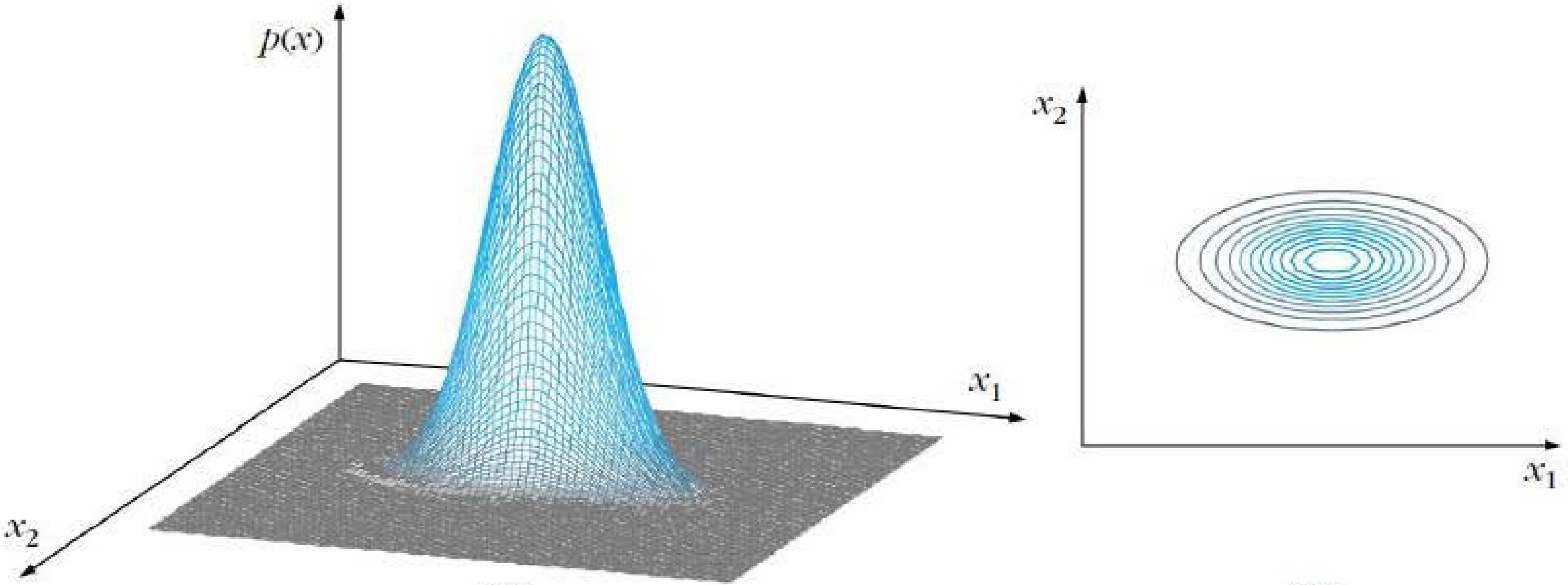
$$(\boldsymbol{x} - \mu)^T \sum\nolimits^{-1} (\boldsymbol{x} - \mu) = [x_1\ x_2] \begin{bmatrix} \frac{1}{\sigma_1^2} & 0 \\ 0 & \frac{1}{\sigma_2^2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \mathrm{C}$$

$$\frac{x_1^2}{\sigma_1^2} + \frac{x_2^2}{\sigma_2^2} = \mathrm{C}$$



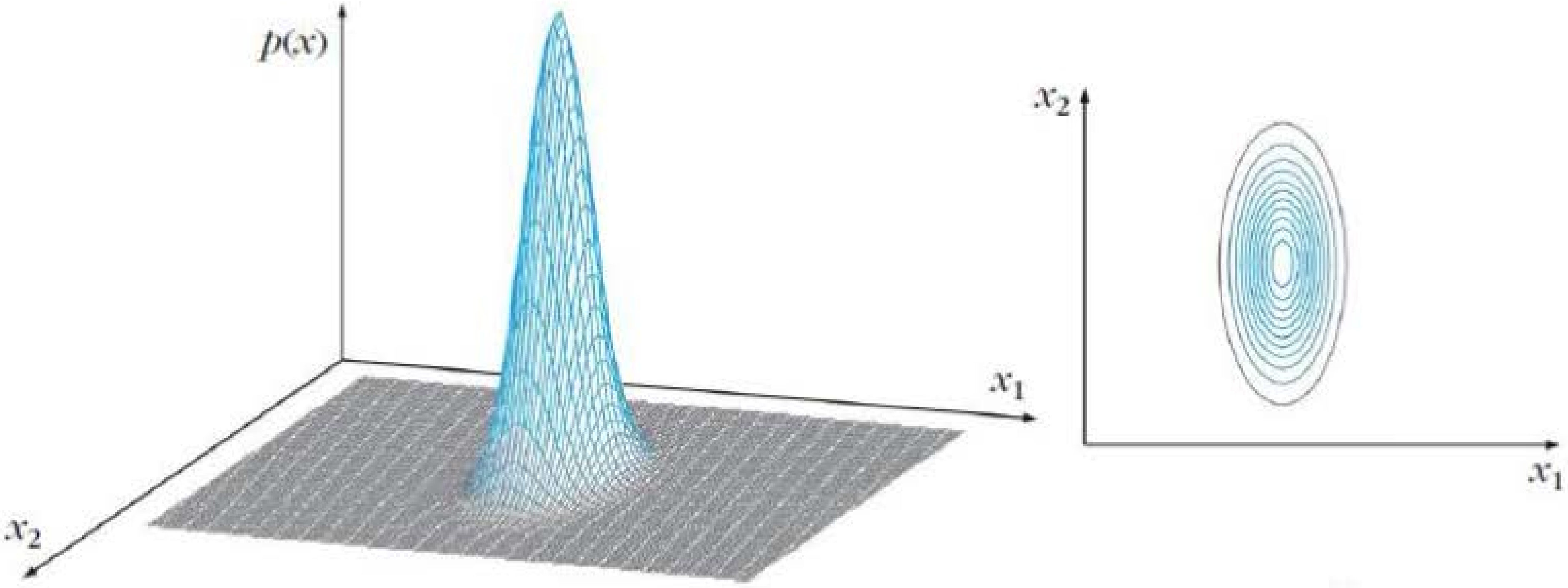
(a) (b)

2D Gaussian pdf, diagonal $\sum$ with $\sigma_1^2 = 15 \gg \sigma_2^2 = 3$



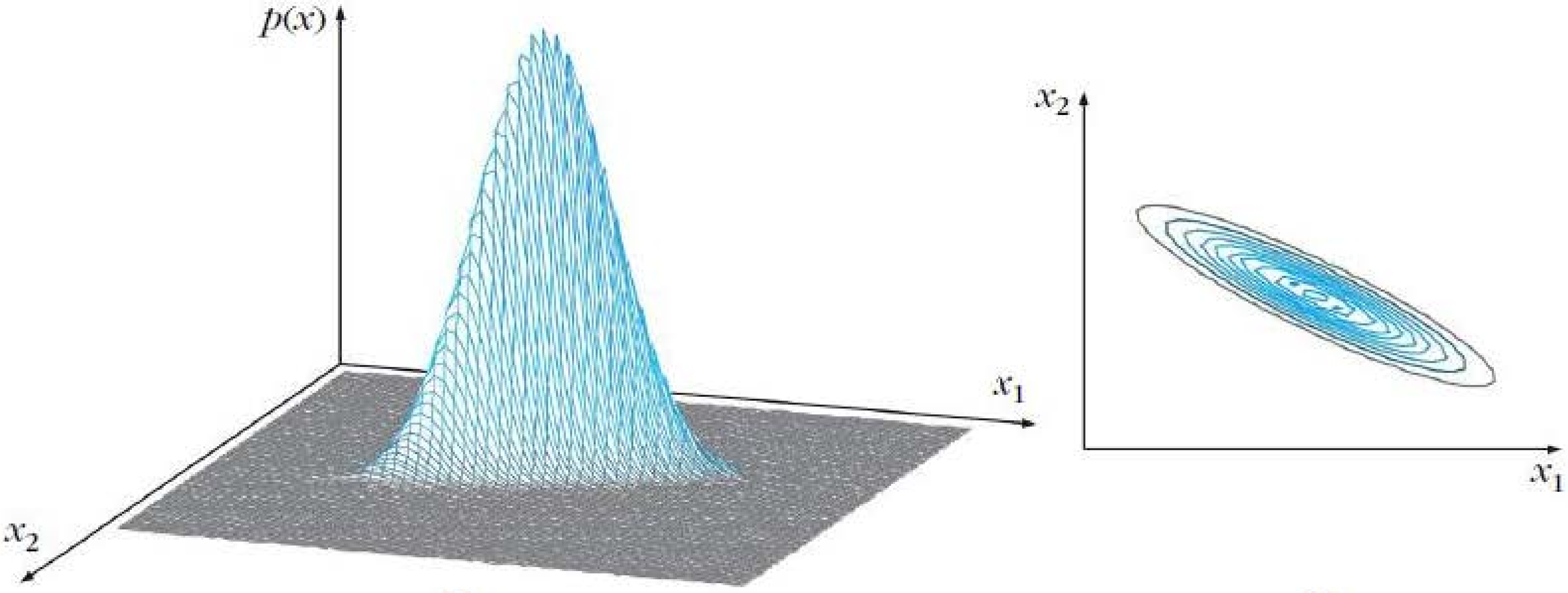
(a) (b)

2D Gaussian pdf, diagonal $\sum$ with $\sigma_1^2 = 3 \ll \sigma_2^2 = 15$



(a) (b)

2D Gaussian pdf, non-diagonal ∑



(a) (b)

Maximum Likelihood

Maximum Likelihood

- Let $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N$ be independent random samples drawn from pdf $p(\boldsymbol{x};\boldsymbol{\theta})$. We form the joint pdf $p(X;\boldsymbol{\theta})$, where $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$

$$p(X;\boldsymbol{\theta}) \equiv p(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N;\boldsymbol{\theta}) = \prod_{k=1}^{N} p(\boldsymbol{x}_k;\boldsymbol{\theta})$$

  It is known as the likelihood function of $\boldsymbol{\theta}$ with respect to X

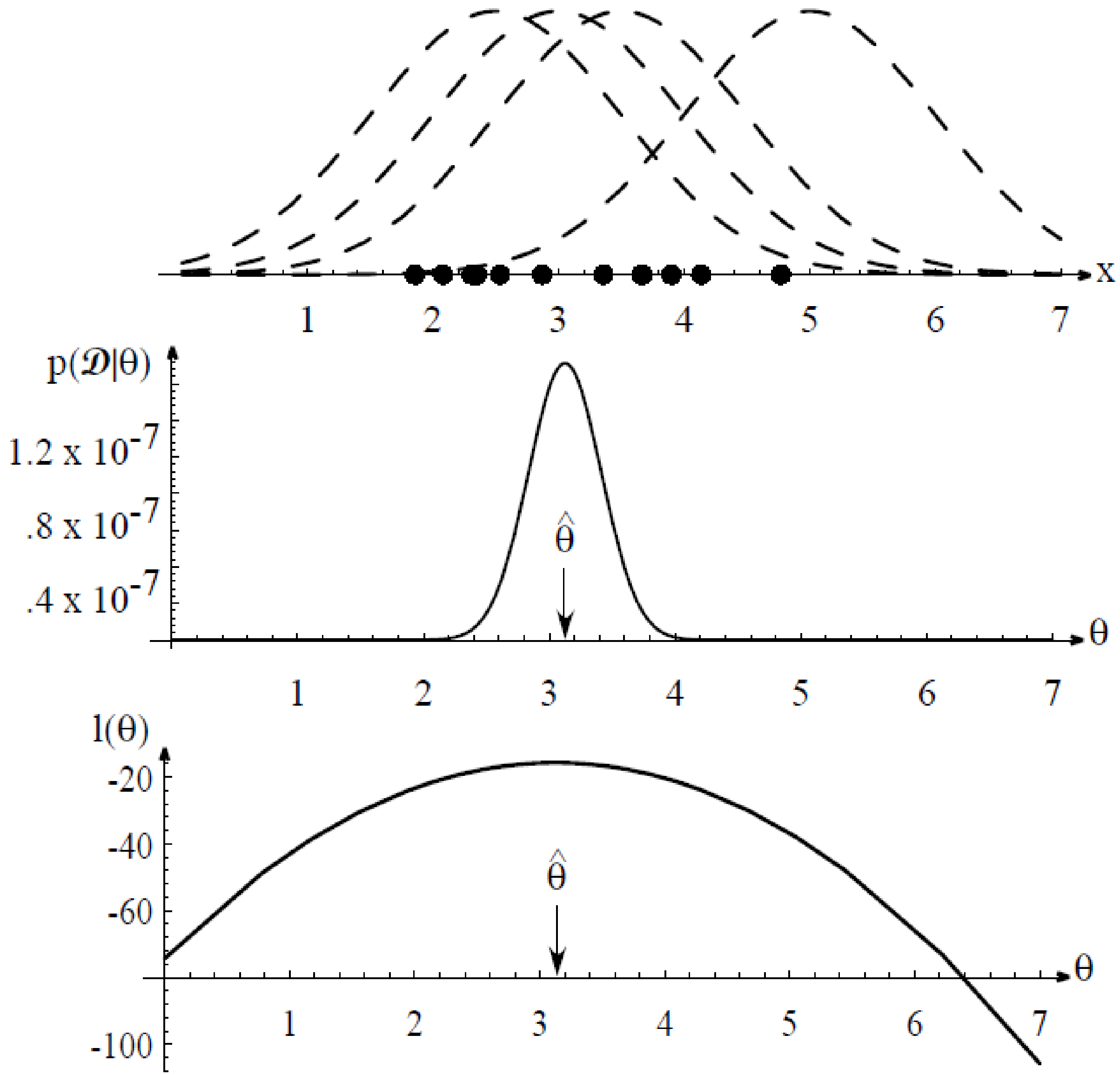
- Using the monotonicity of log, we define the *log-likelihood function*

$$l(\boldsymbol{\theta}) \equiv \log \prod_{k=1}^{N} p(\boldsymbol{x}_k;\boldsymbol{\theta}) = \sum_{k=1}^{N} \log p(\boldsymbol{x}_k;\boldsymbol{\theta})$$

$$\hat{\theta}_{ML} = \arg\max_{\boldsymbol{\theta}} l(\boldsymbol{\theta}) \Rightarrow \frac{\partial l(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{k=1}^{N} \frac{1}{p(\boldsymbol{x}_k;\boldsymbol{\theta})} \frac{\partial p(\boldsymbol{x}_k;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = 0$$

- The ML estimate corresponds to the peak of the log-likelihood function.

Maximum Likelihood



Assume 1D training points are drawn from a Gaussian of a particular variance, but unknown mean. Four of the infinite number of candidate source distributions are shown in dashed lines.

The likelihood $p(D;\theta)$ as a function of the mean. If we had a very large number of training points, this likelihood would be very narrow.

The value that maximizes the likelihood is marked $\hat{\theta}$;

$\hat{\theta}$ also maximizes the logarithm of the likelihood — i.e., the log-likelihood $l(\theta)$, shown at the bottom.

Maximum Likelihood

Assume that N data points, $x_1, x_2, \ldots, x_N$, have been generated by a 1D Gaussian pdf with unknown mean and variance. Derive the $\hat{\mu}_{ML}$ and $\hat{\sigma}^2_{ML}$.

$$L(\mu, \sigma^2) \equiv \log \prod_{k=1}^{N} p(x_k; \mu, \sigma^2) = \sum_{k=1}^{N} \log p(x_k; \mu, \sigma^2) = \sum_{k=1}^{N} \log \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_k-\mu)^2}{2\sigma^2}}$$

$$= -\frac{N}{2}(\log(2\pi) + \log \sigma^2) - \frac{1}{2\sigma^2} \sum_{k=1}^{N} (x_k - \mu)^2$$

$$\frac{\partial L(\mu,\sigma^2)}{\partial \mu} = -\frac{1}{\sigma^2} \sum_{k=1}^{N} (x_k - \mu) = 0 \Rightarrow \quad \hat{\mu}_{ML} = \frac{1}{N} \sum_{k=1}^{N} x_k$$

$$\frac{\partial L(\mu,\sigma^2)}{\partial \sigma^2} = -\frac{N}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{k=1}^{N} (x_k - \mu)^2 = 0 \;\Rightarrow\quad \hat{\sigma}^2_{ML} = \frac{1}{N} \sum_{k=1}^{N} (x_k - \mu)^2$$

Maximum Likelihood

Assume that N data points, $\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_N}$, are vectors generated from a Gaussian pdf with unknown mean and covariance matrix. Derive the ML estimate of the variance.

$$l(\boldsymbol{\mu}, \Sigma) \equiv \log \prod_{k=1}^{N} p(\boldsymbol{x_k}; \boldsymbol{\mu}) = \sum_{k=1}^{N} \log p(\boldsymbol{x_k}; \boldsymbol{\mu})$$

$$= \sum_{k=1}^{N} \log \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp[-\frac{1}{2} (\boldsymbol{x_k} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x_k} - \boldsymbol{\mu})]$$

$$= -\frac{N}{2} (\log(2\pi^d |\Sigma|)) - \frac{1}{2} \sum_{k=1}^{N} (\boldsymbol{x_k} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x_k} - \boldsymbol{\mu})$$

$$\frac{\partial L(\boldsymbol{\mu}, \Sigma)}{\partial \boldsymbol{\mu}} = 0 \Rightarrow \quad \hat{\boldsymbol{\mu}}_{ML} = \frac{1}{N} \sum_{k=1}^{N} \boldsymbol{x_k}$$

$$\frac{\partial L(\boldsymbol{\mu}, \Sigma)}{\partial \Sigma} = 0 \Rightarrow \quad \hat{\Sigma}_{ML} = \frac{1}{N} \sum_{k=1}^{N} (\boldsymbol{x_k} - \hat{\boldsymbol{\mu}}_{ML})(\boldsymbol{x_k} - \hat{\boldsymbol{\mu}}_{ML})^{\mathrm{T}}$$

Maximum Likelihood: An example with three classes (plotcc.m)

Now we have

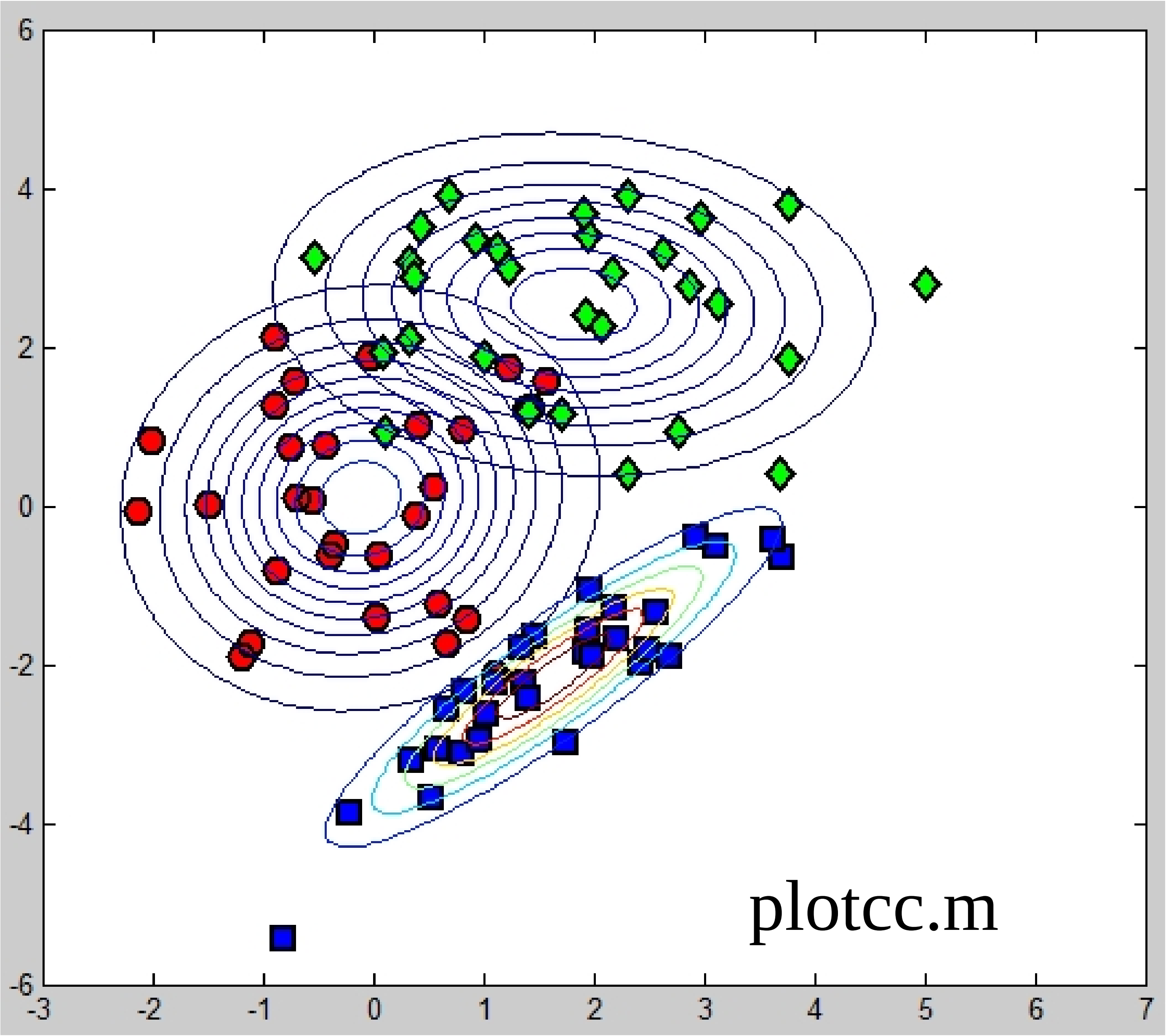
$N(\boldsymbol{\mu}_1, \Sigma_1)$, $N(\boldsymbol{\mu}_2, \Sigma_2)$, $N(\boldsymbol{\mu}_3, \Sigma_3)$

We can compute

$$p(\boldsymbol{x}_{\boldsymbol{new}} | \, t_{new} = 1, \boldsymbol{X}_{\mathbf{1}}, \boldsymbol{t}_{\mathbf{1}})$$
$$p(\boldsymbol{x}_{\boldsymbol{new}} | \, t_{new} = 2, \boldsymbol{X}_{\mathbf{2}}, \boldsymbol{t}_{\mathbf{2}})$$
$$p(\boldsymbol{x}_{\boldsymbol{new}} | \, t_{new} = 3, \boldsymbol{X}_{\mathbf{3}}, \boldsymbol{t}_{\mathbf{3}})$$

And

$$\sum_{\boldsymbol{c}=\mathbf{1}}^{\mathbf{3}} p(\boldsymbol{x}_{\boldsymbol{new}} | \, t_{new} = c, \boldsymbol{X}_{\boldsymbol{c}}, \boldsymbol{t}_{\boldsymbol{c}}) = \mathbf{1}$$



Fit class-conditional Gaussians for each class (plotcc.m)

```
class_var = [];
for c = 1:length(cl)
    pos = find(t==cl(c));
    % Find the means
    class_mean(c,:) = mean(X(pos,:));
    class_var(:,:,c) = cov(X(pos,:),1);
end
%% Plot the contours
for c = 1:length(cl)
    pos = find(t==cl(c));
    plot(X(pos,1),X(pos,2),col{c},...
        'markersize',10,'linewidth',2,'markerfacecolor',fcol{c});
end
xlim([-3 7]),ylim([-6 6])
[Xv,Yv] = meshgrid(-3:0.1:7,-6:0.1:6);
for c = 1:length(cl)
    temp = [Xv(:)-class_mean(c,1) Yv(:)-class_mean(c,2)];
    tempc = class_var(:,:,c);
    const = -log(2*pi) - log(det(tempc));
    Probs = exp(const - 0.5*diag(temp*inv(tempc)*temp'));
    contour(Xv,Yv,reshape(Probs,size(Xv)));
end
```

```
>> class_mean

class_mean =

   -0.1141    0.1117
    1.8161    2.5445
    1.6356   -2.1388
```

```
>> class_var

class_var(:,:,1) =

    0.9896    0.0886
    0.0886    1.5076


class_var(:,:,2) =

    1.7073   -0.1028
   -0.1028    1.0659


class_var(:,:,3) =

    1.1158    1.0470
    1.0470    1.1917
```

Bayes’ Rule

Joint, and Conditional Probability

- $p(A \cap B)$: the joint probability of the events A and B.

- A conditional probability measure $p(A|B)$ is defined by

$$p(A|B) = \frac{p(A \cap B)}{p(B)}$$

- Similarly,

$$p(B|A) = \frac{p(B \cap A)}{p(A)}$$

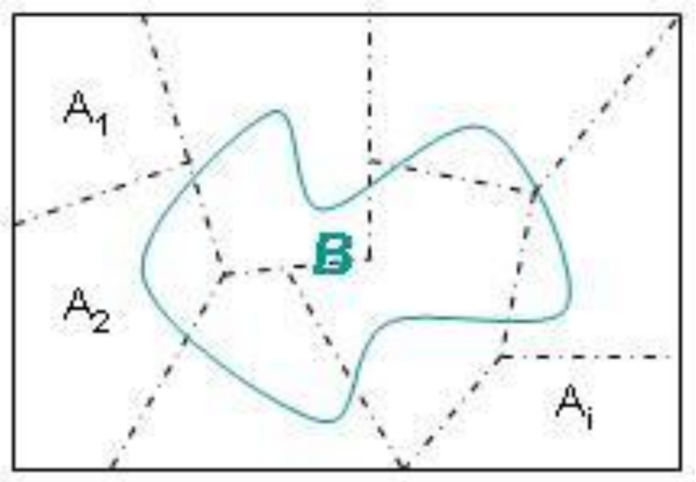
- Therefore

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)}$$

Total Probability

- Let $A_1, A_2, \cdots, A_n$ be n mutually exclusive events,
  i.e., $A_i \cap A_j = \emptyset, i \neq j$, and $\bigcup_{i=1}^{n} A_i = \Omega$ (sample space)

- Let B be any event so that



$$B = B \cap \bigcup_{i=1}^{n} A_i = \bigcup_{i=1}^{n} (B \cap A_i)$$

$$\Rightarrow p(B) = \sum_{i=1}^{n} p(B \cap A_i) = \sum_{i=1}^{n} p(B|A_i) p(A_i)$$

Bayes' Theorem

- Now

$$p(A_i|B) = \frac{p(B|A_i)p(A_i)}{p(B)} = \frac{p(B|A_i)p(A_i)}{\sum_{i=1}^{n} p(B|A_i)p(A_i)}$$

- Therefore

$$p(t_{new} = c|\boldsymbol{x}_{new}, \boldsymbol{X}, \boldsymbol{t}) = \frac{p(\boldsymbol{x}_{new}|\, t_{new} = c, \boldsymbol{X}_c, \boldsymbol{t}_c)p(t_{new} = c|\boldsymbol{X}_c, \boldsymbol{t}_c)}{\sum_{c'=1}^{3} p(\boldsymbol{x}_{new}|\, t_{new} = c', \boldsymbol{X}_{c'}, \boldsymbol{t}_{c'})p(t_{new} = c'|\boldsymbol{X}_{c'}, \boldsymbol{t}_{c'})}$$

$$= \frac{\textbf{likelihood} \times \textbf{prior}}{\textbf{Evidence}}$$

Example

Assume a certain class is given a midterm exam.

S : the event that a student studied, P(S)=0.7

P(a student pass the exam | S)= P(A|S)=0.9

P(a student pass the exam | S^c)= $P(A|S^c)=0.05$

Given that a student did not pass the exam, what is the probability that she or he studied?

$$P(S|A^c) = \frac{P(A^c \cap S) = P(A^c|S)P(S)}{P(A^c) = P(A^c \cap S) + P(A^c \cap S^c)}$$

$$= \frac{P(A^c|S)P(S)}{P(A^c|S)P(S) + P(A^c|S^c)P(S^c)}$$

$$= \frac{(1-0.9)*0.7}{(1-0.9)*0.7+(1-0.05)*(1-0.7)} = \frac{0.07}{0.335} = 19.7\%$$

Making prediction

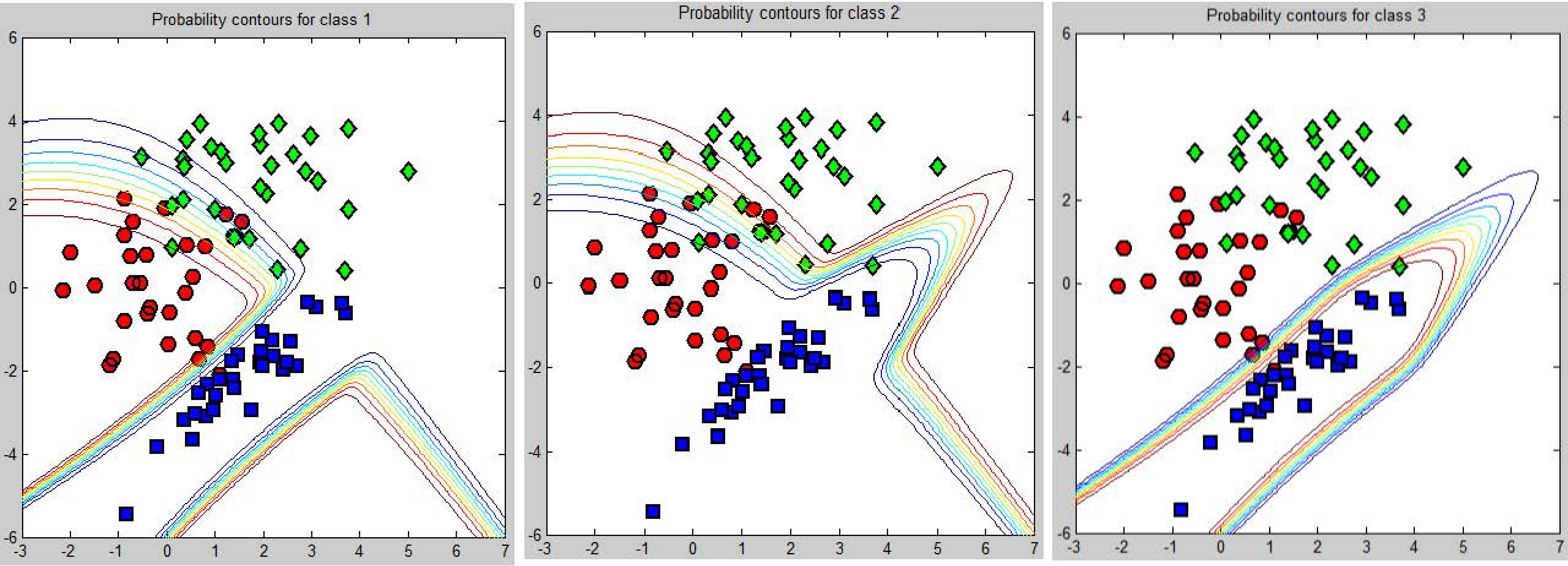
- Since we can use the known $N(\boldsymbol{\mu}_1, \Sigma_1)$, $N(\boldsymbol{\mu}_2, \Sigma_2)$, and $N(\boldsymbol{\mu}_3, \Sigma_3)$ to compute $p(\boldsymbol{x}_{\boldsymbol{new}} | t_{new} = 1, \boldsymbol{X}_1, \boldsymbol{t}_1)$, $p(\boldsymbol{x}_{\boldsymbol{new}} | t_{new} = 2, \boldsymbol{X}_2, \boldsymbol{t}_2)$ and $p(\boldsymbol{x}_{\boldsymbol{new}} | t_{new} = 3, \boldsymbol{X}_3, \boldsymbol{t}_3)$

- If we assume the prior $p(t_{new} = c | \boldsymbol{X}_c, \boldsymbol{t}_c) = \frac{N_c}{N_1 + N_2 + N_3}$
- Therefore

$$p(t_{new} = c | \boldsymbol{x}_{\boldsymbol{new}}, \boldsymbol{X}, \boldsymbol{t}) = \frac{p(\boldsymbol{x}_{\boldsymbol{new}} | t_{new} = c, \boldsymbol{X}_c, \boldsymbol{t}_c) p(t_{new} = c | \boldsymbol{X}_c, \boldsymbol{t}_c)}{\sum_{c'=1}^{3} p(\boldsymbol{x}_{\boldsymbol{new}} | t_{new} = c', \boldsymbol{X}_{c'}, \boldsymbol{t}_{c'}) p(t_{new} = c' | \boldsymbol{X}_{c'}, \boldsymbol{t}_{c'})}$$

$$= \frac{N(\boldsymbol{x}_{\boldsymbol{new}}; \boldsymbol{\mu}_c, \Sigma_c) \frac{N_c}{N_1 + N_2 + N_3}}{\sum_{c'=1}^{3} N(\boldsymbol{x}_{\boldsymbol{new}}; \boldsymbol{\mu}_{c'}, \Sigma_{c'}) \frac{N_{c'}}{N_1 + N_2 + N_3}}$$

Making prediction (bayesclass.m)



```
%% bayesclass.m: Compute the predictive probabilities
[Xv,Yv] = meshgrid(-3:0.1:7,-6:0.1:6);
Probs = [];
for c = 1:length(cl)
    temp = [Xv(:)-class_mean(c,1) Yv(:)-class_mean(c,2)];
    tempc = class_var(:,:,c);
    const = -log(2*pi) - log(det(tempc));
    Probs(:,:,c) = reshape(exp(const - 0.5*diag(temp*inv(tempc)*temp')),size(Xv));
end
Probs = Probs./repmat(sum(Probs,3),[1,1,3]);
 %% Plot the predictive contours
figure(2);
for i = 1:3
    subplot(1,3,i);
    hold off
    for c = 1:length(cl)
        pos = find(t==cl(c));
        plot(X(pos,1),X(pos,2),col{c},...
            'markersize',10,'linewidth',2,'markerfacecolor',fcol{c});
        hold on
    end
    xlim([-3 7]),ylim([-6 6])
    contour(Xv,Yv,Probs(:,:,i));
    ti = sprintf('Probability contours for class %g',i);title(ti);
end
```

$$\boxed{p(t_{new} = c \mid \boldsymbol{X}_c, \boldsymbol{t}_c) = \mathbf{1/3}}$$

$$p(t_{new} = c \mid \boldsymbol{x}_{new}, \boldsymbol{X}, \boldsymbol{t}) = \frac{N(\boldsymbol{x}_{new}; \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)\cancel{\dfrac{N_c}{N_1+N_2+N_3}}}{\sum_{c'=1}^{3} N(\boldsymbol{x}_{new}; \boldsymbol{\mu}_{c'}, \boldsymbol{\Sigma}_{c'})\cancel{\dfrac{N_{c'}}{N_1+N_2+N_3}}}$$

$$\boldsymbol{x}_{new} = \begin{bmatrix} 2 \\ 0 \end{bmatrix},\ t_{new} = p(t_{new} = \boldsymbol{c} | \boldsymbol{x}_{new}, \boldsymbol{X}_c, \boldsymbol{t}_c) = ?$$

c	$p(\boldsymbol{x}_{new} \mid t_{new} = c, \boldsymbol{X}_c, \boldsymbol{t}_c)$	$p(t_{new} = c \mid \boldsymbol{X}_c, \boldsymbol{t}_c)$	$\frac{p(\boldsymbol{x}_{new} \mid t_{new} = c, \boldsymbol{X}_c, \boldsymbol{t}_c)}{\sum_{c'=1}^{3} p(\boldsymbol{x}_{new} \mid t_{new} = c', \boldsymbol{X}_{c'}, \boldsymbol{t}_{c'})}$
1	0.0109	0.333	0.7072
2	0.0042	0.333	0.2741
3	0.0003	0.333	0.0187

bayesclass_x_new.m

```
%% bayesclass_x_new.m: Repeat without Naive assumption
class_var = [];
for c = 1:length(cl)
    pos = find(t==cl(c));
    % Find the means
    class_mean(c,:) = mean(X(pos,:));
    class_var(:,:,c) = cov(X(pos,:),1);
end
%% Compute the predictive probabilities
Xv=2;
Yv=0 ;
Probs = [];
for c = 1:length(cl)
    temp = [Xv(:)-class_mean(c,1) Yv(:)-class_mean(c,2)];
    tempc = class_var(:,:,c);
    const = -log(2*pi) - log(det(tempc));
    Probs(:,:,c) = reshape(exp(const -
0.5*diag(temp*inv(tempc)*temp')),size(Xv));
end
[Probs(:,:,1) Probs(:,:,2) Probs(:,:,3) sum(Probs,3)]% 分子 與 分母

Probs = Probs./repmat(sum(Probs,3),[1,1,3])
```

The naive-Bayes assumption

The naive-Bayes assumption

- Fitting a 2-D Gaussian requires 5 parameter: 2 for $\boldsymbol{\mu}_c$, and 3 for $\sum_c$.

→ feasible for 30 training points in each class.

- But fitting a D-dimensional Gaussian requires $\mathrm{D} + \mathrm{D} + \mathrm{D}(\mathrm{D} - 1)/2$ parameters.

→ For 10 dimensions, 30 data points are not sufficient to fit 65 parameters.

- Naive Bayes assumption: the class-conditional distributions can be factorized into a product of univariate distributions.

→ $p(\boldsymbol{x_i} | t_i = c, \boldsymbol{X_c}, \boldsymbol{t_c}) = \prod_{d=1}^{D} p(x_{id} | t_i = c, \boldsymbol{X_c}, \boldsymbol{t_c})$

→ cannot model any within-class dependencies

→ 10-dimensional Gaussian requires 20 parameters instead of 65

Maximum Likelihood: An example with three classes

Now we have

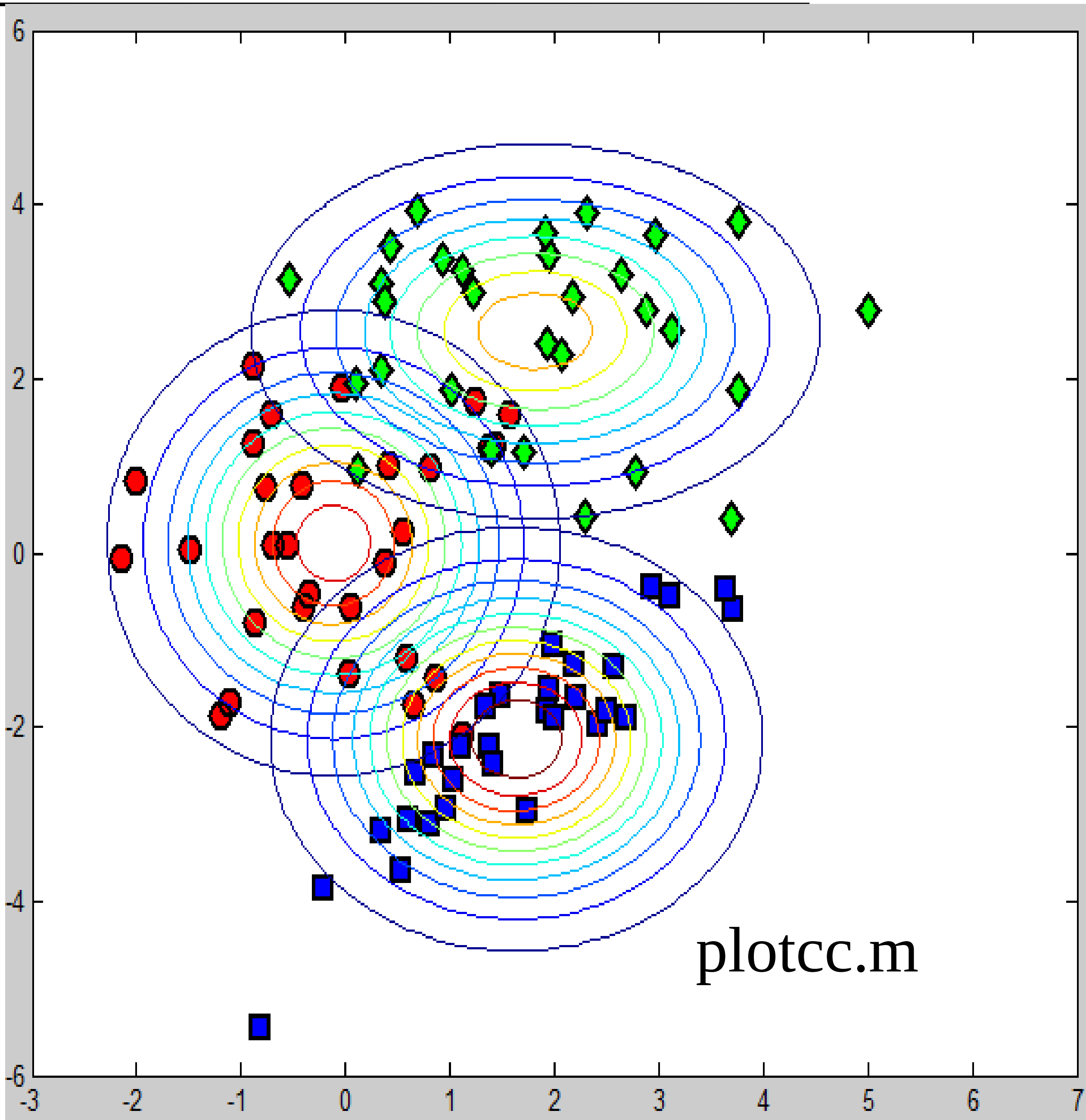
$N(\boldsymbol{\mu}_1, \sigma_1^2 \mathrm{I})$, $N(\boldsymbol{\mu}_2, \sigma_2^2 \mathrm{I})$, $N(\boldsymbol{\mu}_3, \sigma_3^2 \mathrm{I})$

We can compute

$$p(\boldsymbol{x}_{\boldsymbol{new}} | \, t_{new} = 1, \boldsymbol{X}_1, \boldsymbol{t}_1)$$
$$p(\boldsymbol{x}_{\boldsymbol{new}} | \, t_{new} = 2, \boldsymbol{X}_2, \boldsymbol{t}_2)$$
$$p(\boldsymbol{x}_{\boldsymbol{new}} | \, t_{new} = 3, \boldsymbol{X}_3, \boldsymbol{t}_3)$$

And

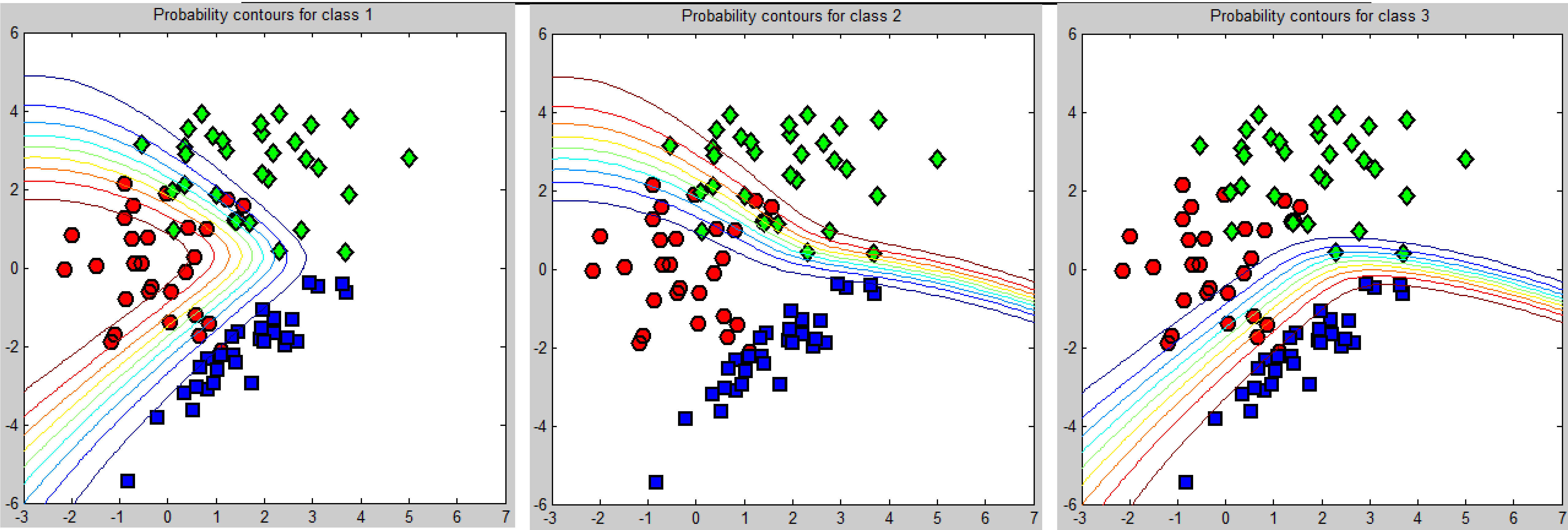
$$\sum_{\boldsymbol{c}=\mathbf{1}}^{\mathbf{3}} p(\boldsymbol{x}_{\boldsymbol{new}} | \, t_{new} = c, \boldsymbol{X}_{\boldsymbol{c}}, \boldsymbol{t}_{\boldsymbol{c}}) = \mathbf{1}$$



```
%% Fit class-conditional Gaussians for each class
% Using the Naive (independence) assumption
for c = 1:length(cl)
    pos = find(t==cl(c));
    % Find the means
    class_mean(c,:) = mean(X(pos,:));
    class_var(c,:) = var(X(pos,:),1);
end

%% Plot the contours
[Xv,Yv] = meshgrid(-3:0.1:7,-6:0.1:6);
for c = 1:length(cl)
    temp = [Xv(:)-class_mean(c,1) Yv(:)-class_mean(c,2)];
    tempc = diag(class_var(c,:));
    const = -log(2*pi) - log(det(tempc));
    Probs = exp(const - 0.5*diag(temp*inv(tempc)*temp'));
    contour(Xv,Yv,reshape(Probs,size(Xv)));
end
```

Making prediction (bayesclass_naive.m)



Although the class-conditional distribution $p(\boldsymbol{x_i}|\ t_i = 3, \boldsymbol{X_3}, \boldsymbol{t_3})$ for class 3 is not particularly appropriate, the classification contours are still reasonable

```
%% Compute the predictive probabilities
[Xv,Yv] = meshgrid(-3:0.1:7,-6:0.1:6);
Probs = [];
for c = 1:length(cl)
    temp = [Xv(:)-class_mean(c,1) Yv(:)-class_mean(c,2)];
    tempc = diag(class_var(c,:));
    const = -log(2*pi) - log(det(tempc));
    Probs(:,:,c) = reshape(exp(const - 0.5*diag(temp*inv(tempc)*temp')),size(Xv));
end
Probs = Probs./repmat(sum(Probs,3),[1,1,3]);
%% Plot the predictive contours
for i = 1:3
    subplot(1,3,i);
    for c = 1:length(cl)
        pos = find(t==cl(c));
        plot(X(pos,1),X(pos,2),col{c},...
            'markersize',10,'linewidth',2,'markerfacecolor',fcol{c});
        hold on
    end
    xlim([-3 7]), ylim([-6 6])
    contour(Xv,Yv,Probs(:,:,i));
    ti = sprintf('Probability contours for class %g',i);
    title(ti);
end
```

Summary

- We assume the class-conditional distribution $p(\boldsymbol{x_i}|\, t_i = c, \boldsymbol{X}_c, \boldsymbol{t}_c)$ is a multivariate Gaussian $N(\boldsymbol{\mu}_c, \Sigma_c)$

- $\hat{\boldsymbol{\mu}}_c = \frac{1}{N}\sum_{k=1}^{N} \boldsymbol{x_k}$ and $\hat{\Sigma}_c = \frac{1}{N}\sum_{k=1}^{N}(\boldsymbol{x_k} - \hat{\boldsymbol{\mu}}_c)(\boldsymbol{x_k} - \hat{\boldsymbol{\mu}}_c)^{\mathrm{T}}$ are the empirical mean and covariance computed from maximizing likelihood function.

- We make prediction using Bayes' rule

$$p(t_{new} = c|\boldsymbol{x_{new}}, \boldsymbol{X}, \boldsymbol{t}) = \frac{N(\boldsymbol{x_{new}}; \boldsymbol{\mu}_c, \Sigma_c)\dfrac{\boldsymbol{N_c}}{\boldsymbol{N_1} + \boldsymbol{N_2} + \boldsymbol{N_3}}}{\sum_{\boldsymbol{c'=1}}^{\boldsymbol{3}} N(\boldsymbol{x_{new}}; \boldsymbol{\mu}_{c'}, \Sigma_{c'})\dfrac{\boldsymbol{N_{c'}}}{\boldsymbol{N_1} + \boldsymbol{N_2} + \boldsymbol{N_3}}}$$

- Use Naïve assumption to reduce the number of estimated parameters

Remark: Discriminant Analysis

Assume $p(\boldsymbol{x_i}|\, t_i = c, \boldsymbol{X}_c, \boldsymbol{t}_c)$ are Gaussian densities,

1. the same $\sum_c = \sum$ in each class, this leads to linear discriminant analysis.

2. different $\sum_c$ in each class, we get quadratic discriminant analysis.

3. $\sum_c$ are diagonal, i.e., conditional independence in each class ,we get naïve Bayes.