3. Bayesian Decision Theory

We begin with the fish classification example.

Let ω denote class, with $\omega = \omega_1$ for sea bass and $\omega = \omega_2$ for salmon. We assume:

- There is a prior probability $p(\omega_1)$ that the fish is sea bass and prior probability $p(\omega_2)$ for salmon.
- If there is no other types of fish, then:

$$p(\omega_1) + p(\omega_2) = 1$$

Prior probability

Prior probability is the probability of an event occurring before considering new evidence. It represents initial beliefs about an event based on existing knowledge or assumptions

Example

If the proportion of male and female students in the EE6407 class is 60% and 40%, respectively, then the prior probabilities are:

$$p(male) = 0.6$$

$$p(female) = 0.4$$

Suppose we are forced to decide the type of fish without seeing the fish, it is then logical to decide (guess) fish type using the prior probability:

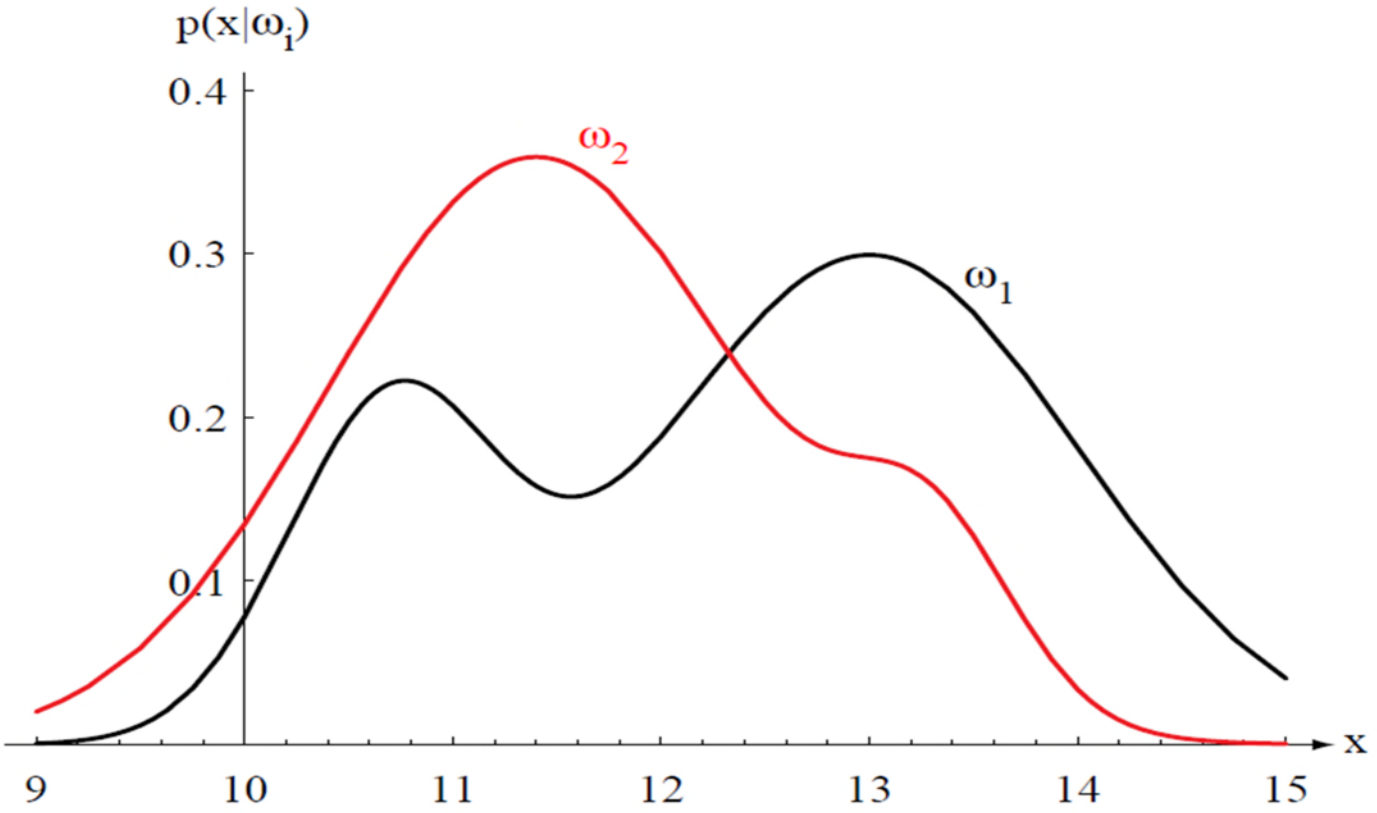
Decide ω_1 if $p(\omega_1) > p(\omega_2)$

Decide ω_2 if $p(\omega_2) > p(\omega_1)$

In practice, we may have additional information such as the lightness measurement x to decide the type of fish.

Different fish (of the same type) may have different lightness measurements, this variability can be expressed in probability terms: x is considered as a continuous random variable having a class-conditional probability density function $p(x|\omega)$.

$p(x|\omega)$ specifies the probability density function for x given that the class is ω. The difference between $p(x|\omega_1)$ and $p(x|\omega_2)$ describes the difference in lightness between the two classes:



Suppose we know

- ❑ the prior probability
- ❑ the conditional probability density
- ❑ the lightness measurement x of the fish

How could we decide the category of the fish?

Based on Bayes theorem:

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

where

$$p(x) = \sum_{j=1}^{2} p(x|\omega_j)p(\omega_j)$$

$p(\omega_j|x)$ is called posterior probability.

Bayes decision rule

Decide ω_1 if $p(\omega_1|x) > p(\omega_2|x)$;

Decide ω_2 if $p(\omega_2|x) > p(\omega_1|x)$.

Note: $p(x)$ is just a scale factor and is unimportant as far as making decision is concerned.

By eliminating this scale factor, we actually use the joint probability $p(x, \omega_i)$ to make decision:

Decide ω_1 if $p(x|\omega_1)p(\omega_1) > p(x|\omega_2)p(\omega_2)$;

Decide ω_2 if $p(x|\omega_2)p(\omega_2) > p(x|\omega_1)p(\omega_1)$

Joint Probability

Joint probability refers to the probability of two (or more) events occurring simultaneously.

$$p(X, Y) = p(X|Y)P(Y)$$

We next generalize the above in 2 ways:

1) More than one feature, i.e. multiple features
2) More than two classes (categories), i.e. multiple classes

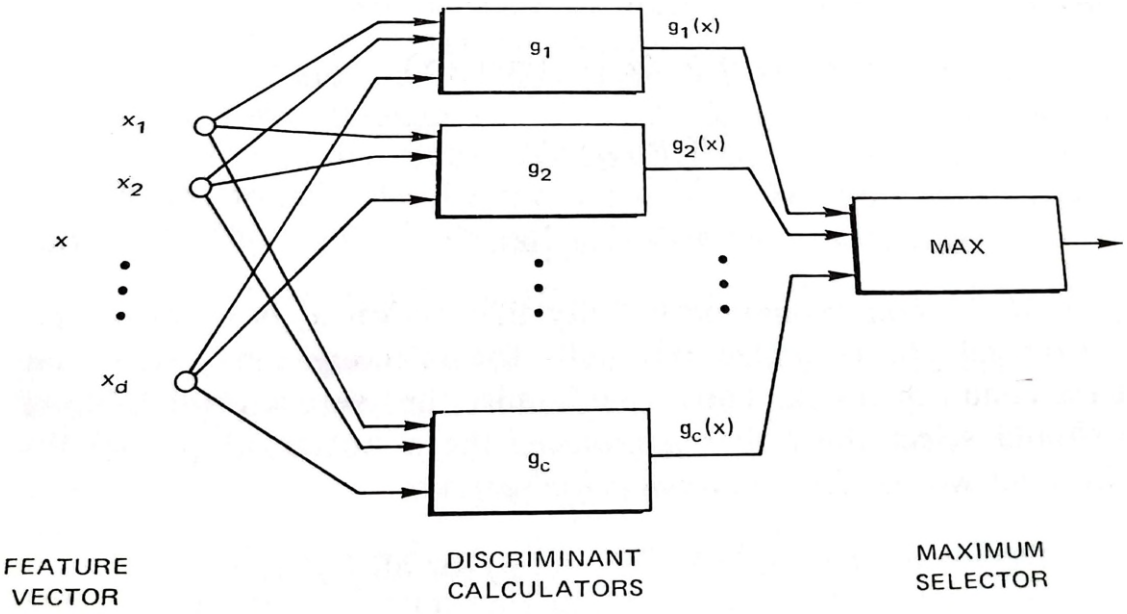
Let $\mathbf{x}$ denotes the feature vector, and $\{\omega_1, \omega_2, \cdots, \omega_c\}$ denotes the c classes. Then the posterior probability can be computed as follows:

$$p(\omega_j|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_j)p(\omega_j)}{p(\mathbf{x})}$$

where

$$p(\mathbf{x}) = \sum_{j=1}^{c} p(\mathbf{x}|\omega_j)p(\omega_j)$$

We introduce the concept of discriminant function in pattern classification. Assuming the discriminant functions are denoted by $g_i(\mathbf{x})$, $i = 1,2,\ldots,c$.



The classifier assigns $\mathbf{x}$ to class ω_i if

$$g_i(\mathbf{x}) > g_j(\mathbf{x}) \quad \text{for all } j \neq i$$

In Bayes decision rule, the posterior probability is used as the discriminant function:

$$g_i(\mathbf{x}) = p(\omega_i|\mathbf{x}), \qquad i = 1,2,\dots,c$$

In addition, the following variants can also be used:

$$g_i(\mathbf{x}) = p(\mathbf{x}|\omega_i)p(\omega_i)$$

$$g_i(\mathbf{x}) = \log p(\mathbf{x}|\omega_i) + \log p(\omega_i)$$

A Bayes classifier is determined by the conditional probability density function as well as the prior probability..

Univariate normal density function

$$p(x|\omega) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$$

Where μ is mean or the expected value of x belonging to class ω:

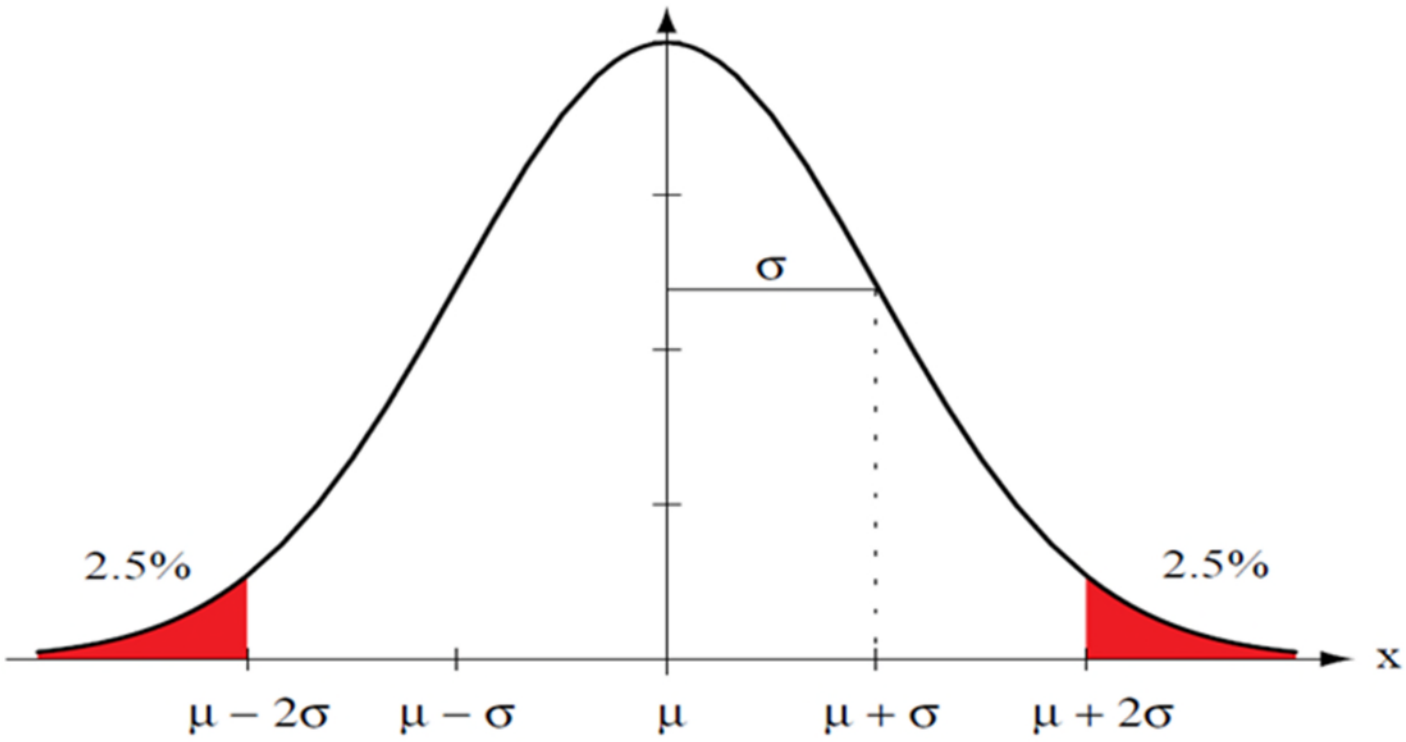
$$\mu = \int_{-\infty}^{+\infty} xp(x|\omega)dx$$

σ^2 is the variance of x:

$$\sigma^2 = \int_{-\infty}^{+\infty} (x-\mu)^2 p(x|\omega)dx$$

Normal density function is completely specified by the mean and the variance, and is often expressed as:

$$p(x|\omega) \sim N(\mu, \sigma^2)$$



Multivariate normal density function

The general multivariate normal density function in d dimensions is as follow:

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

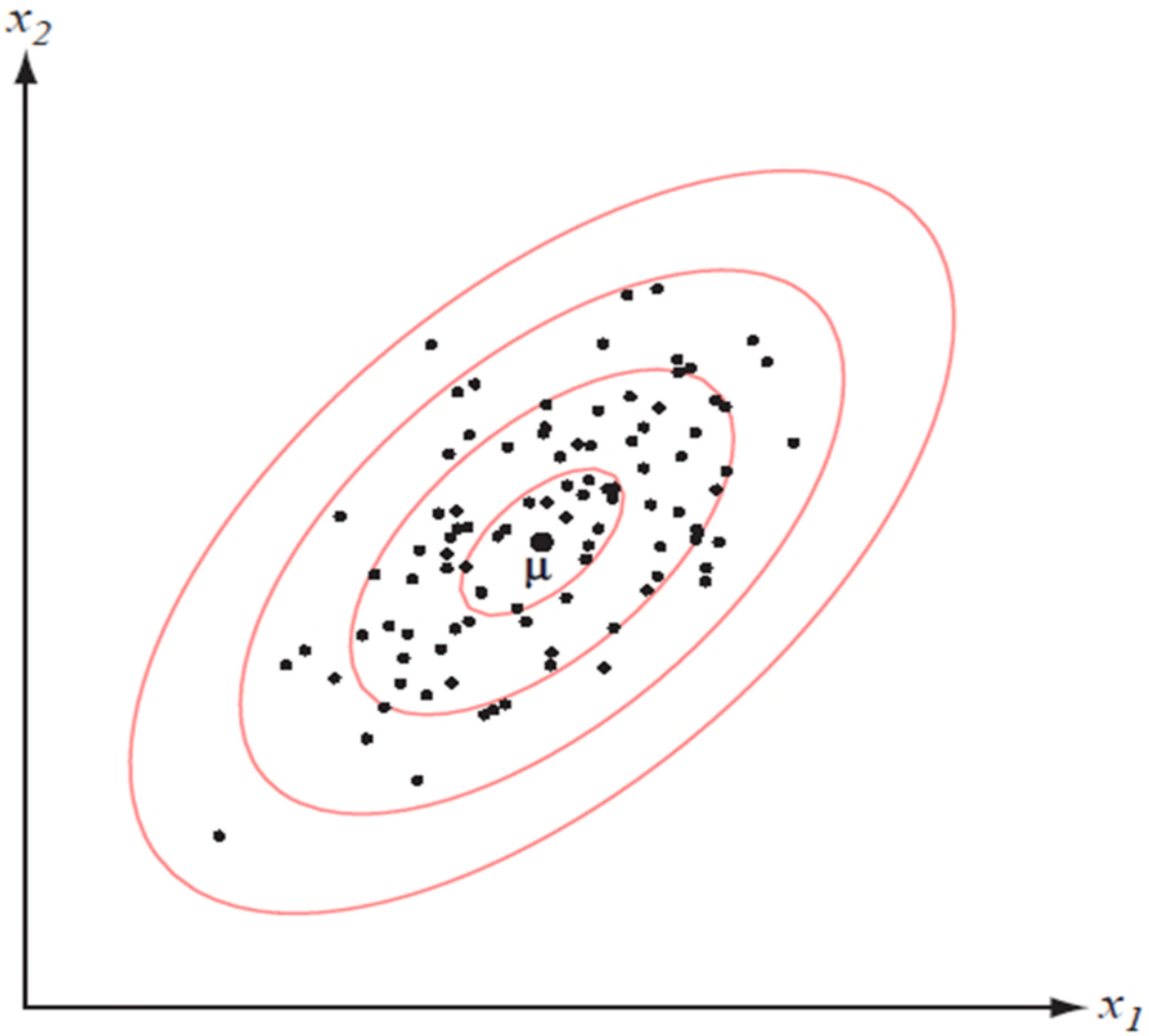
Where

$\boldsymbol{\mu}$ is the d-dimensional mean vector,

$\mathbf{C}$ is the d-by-d covariance matrix,

$|\mathbf{C}|$ and $\mathbf{C}^{-1}$ denote the determinant and inverse of $\mathbf{C}$.

Samples drawn from a two-dimensional Gaussian centred at $\boldsymbol{\mu}$:



Parameter estimation for Gaussian density function

In practice, the prior probabilities $p(\omega_i)$ and the class-conditional densities $p(\mathbf{x}|\omega_i)$ are not given. We can use the training samples to estimate them:

$$p(\mathbf{x}|\omega) = \frac{1}{(2\pi)^{d/2}|\widehat{\mathbf{C}}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\widehat{\boldsymbol{\mu}})^T \hat{\mathbf{C}}^{-1}(\mathbf{x}-\widehat{\boldsymbol{\mu}})\right]$$

Where $\widehat{\boldsymbol{\mu}}$ and $\hat{\mathbf{C}}$ denotes estimation of mean vector $\boldsymbol{\mu}$ and covariance matrix $\mathbf{C}$, respectively.

But how to estimate mean vector and covariance matrix using the training samples?

Maximum-likelihood parameter estimation

Suppose we have c datasets, D_1, D_2, …, D_c, with samples in D_j having been drawn independently according to the probability density function $p(\mathbf{x}|\omega_j)$.

Assume $p(\mathbf{x}|\omega_j) \sim N(\boldsymbol{\mu}_j, \mathbf{C}_j)$, and samples in D_i give no information about $\boldsymbol{\mu}_j$ and $\mathbf{C}_j$ if $i \neq j$. This permits us to work with each class separately.

With this assumption, we thus have c separate problems of the following form:

Use a set D of training samples drawn independently from the probability density $\mathrm{p}(\mathbf{x}|\boldsymbol{\theta})$ to estimate the unknown parameter vector $\boldsymbol{\theta}$, where $\boldsymbol{\theta}$ consists of $\boldsymbol{\mu}$ and $\mathbf{C}$.

Suppose that D contains n samples: $\mathbf{x}_1$, $\mathbf{x}_2$,…, $\mathbf{x}_n$. Because the samples are drawn independently, we have:

$$p(D|\boldsymbol{\theta}) = \prod_{k=1}^{n} p(\mathbf{x}_k|\boldsymbol{\theta})$$

$p(D|\boldsymbol{\theta})$ is called the likelihood of $\boldsymbol{\theta}$ with respect to the set of samples D.

The maximum-likelihood estimate of $\boldsymbol{\theta}$ is, by definition, the value of $\widehat{\boldsymbol{\theta}}$ that maximizes $p(D|\boldsymbol{\theta})$.

We define the log-likelihood function as:

$$l(\boldsymbol{\theta}) = \ln p(D|\boldsymbol{\theta}) = \sum_{k=1}^{n} \ln p(\mathbf{x}_k|\boldsymbol{\theta})$$

where $\ln$ denotes natural logarithm.

Then the solution can be written as:

$$\widehat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} l(\boldsymbol{\theta})$$

Because the logarithm is monotonically increasing, the $\widehat{\boldsymbol{\theta}}$ that maximizes the log-likelihood $l(\boldsymbol{\theta})$ also maximizes the likelihood $p(D|\boldsymbol{\theta})$.

The set of necessary conditions for the maximum-likelihood estimate for $\boldsymbol{\theta}$ is as follow:

$$\frac{\partial l(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{k=1}^{n} \frac{\partial \ln p(\mathbf{x}_k|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = 0$$

Case 1: Gaussian function with unknown $\boldsymbol{\mu}$

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

Then:

$$\ln p(\mathbf{x}_k|\boldsymbol{\mu}) = -\frac{1}{2}\ln[(2\pi)^d|\mathbf{C}|] - \frac{1}{2}(\mathbf{x}_k - \boldsymbol{\mu})^T \mathbf{C}^{-1}(\mathbf{x}_k - \boldsymbol{\mu})$$

Thus, we have:

$$\frac{\partial \ln p(\mathbf{x}_k|\boldsymbol{\mu})}{\partial \boldsymbol{\mu}} = \mathbf{C}^{-1}(\mathbf{x}_k - \boldsymbol{\mu})$$

In terms of the set of necessary conditions for the maximum-likelihood estimate, we have:

$$\sum_{k=1}^{n} \mathbf{C}^{-1}(\mathbf{x}_k - \boldsymbol{\mu}) = 0$$

Multiplying $\mathbf{C}$ and rearranging, we obtain the maximum-likelihood estimate of $\boldsymbol{\mu}$:

$$\hat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{k=1}^{n} \mathbf{x}_k$$

Case 2: Gaussian function with unknown $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$

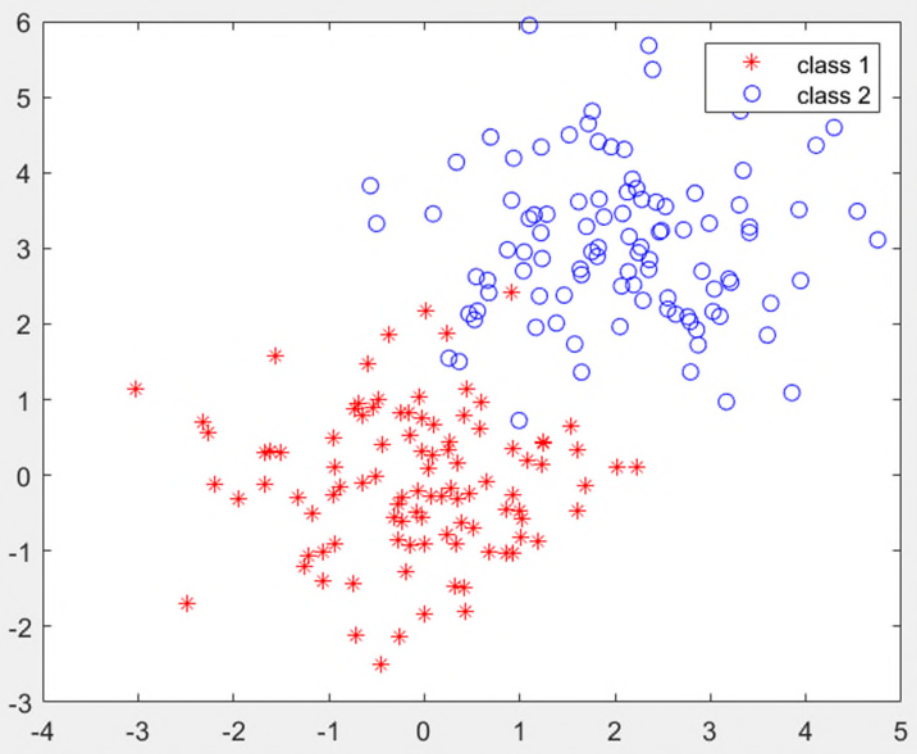
With a similar deviation process, the maximum-likelihood estimate of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are obtained as follows:

$$\hat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{k=1}^{n} \mathbf{x}_k$$

$$\hat{\mathbf{C}} = \frac{1}{n}\sum_{k=1}^{n} (\mathbf{x}_k - \hat{\boldsymbol{\mu}})(\mathbf{x}_k - \hat{\boldsymbol{\mu}})^T$$

Example:

200 training samples from two classes are shown below. Assuming the data follows normal distribution, design a Bayesian decision rule using the training data.



We first estimate the parameters of the multivariate normal probability density function:

$$\hat{\boldsymbol{\mu}}_1 = \frac{1}{n_1}\sum_{\mathbf{x}_k \in D_1} \mathbf{x}_k = \begin{bmatrix} -0.1055 \\ -0.0974 \end{bmatrix}$$

$$\hat{\boldsymbol{\mu}}_2 = \frac{1}{n_2}\sum_{\mathbf{x}_k \in D_2} \mathbf{x}_k = \begin{bmatrix} 2.0638 \\ 3.0451 \end{bmatrix}$$

$$\hat{\mathbf{C}}_1 = \frac{1}{n_1}\sum_{\mathbf{x}_k \in D_1} (\mathbf{x}_k - \hat{\boldsymbol{\mu}}_1)(\mathbf{x}_k - \hat{\boldsymbol{\mu}}_1)^T = \begin{bmatrix} 1.0253 & -0.0036 \\ -0.0036 & 0.8880 \end{bmatrix}$$

$$\hat{\mathbf{C}}_2 = \frac{1}{n_2}\sum_{\mathbf{x}_k \in D_2} (\mathbf{x}_k - \hat{\boldsymbol{\mu}}_2)(\mathbf{x}_k - \hat{\boldsymbol{\mu}}_2)^T = \begin{bmatrix} 1.1884 & -0.013 \\ -0.013 & 1.0198 \end{bmatrix}$$

The class-conditional probability densities are obtained as:

$$p(\mathbf{x}|\omega_1) = \frac{1}{(2\pi)^{2/2}\sqrt{|\hat{\mathbf{C}}_1|}} \exp\left[-\frac{1}{2}(\mathbf{x}-\hat{\boldsymbol{\mu}}_1)^T \hat{\mathbf{C}}_1^{-1}\,(\mathbf{x}-\hat{\mathbf{u}}_1)\right]$$

$$= \frac{1}{2\pi \times 0.9542} \exp\left[-\frac{1}{2}\left(\mathbf{x} - \begin{bmatrix}-0.1055\\-0.0974\end{bmatrix}\right)^T \times \begin{bmatrix}0.9753 & 0.004\\0.004 & 1.1261\end{bmatrix} \times \left(\mathbf{x} - \begin{bmatrix}-0.1055\\-0.0974\end{bmatrix}\right)\right]$$

$$p(\mathbf{x}|\omega_2) = \frac{1}{(2\pi)^{2/2}\sqrt{|\hat{\mathbf{C}}_2|}} \exp\left[-\frac{1}{2}(\mathbf{x}-\hat{\boldsymbol{\mu}}_2)^T \hat{\mathbf{C}}_2^{-1}\,(\mathbf{x}-\hat{\mathbf{u}}_2)\right]$$

$$= \frac{1}{2\pi \times 1.1008} \exp\left[-\frac{1}{2}\left(\mathbf{x} - \begin{bmatrix}2.0638\\3.0451\end{bmatrix}\right)^T \times \begin{bmatrix}0.8416 & 0.0107\\0.0107 & 0.9807\end{bmatrix} \times \left(\mathbf{x} - \begin{bmatrix}2.0638\\3.0451\end{bmatrix}\right)\right]$$

The discriminant functions are:

$$g_1(\mathbf{x}) = p(\mathbf{x}|\omega_1)p(\omega_1) = \frac{100}{200}p(\mathbf{x}|\omega_1) = \frac{1}{2}p(\mathbf{x}|\omega_1)$$

$$g_2(\mathbf{x}) = p(\mathbf{x}|\omega_2)p(\omega_2) = \frac{100}{200}p(\mathbf{x}|\omega_2) = \frac{1}{2}p(\mathbf{x}|\omega_2)$$

1) $\mathbf{x}$ is classified into class 1 if

$$g_1(\mathbf{x}) > g_2(\mathbf{x})$$

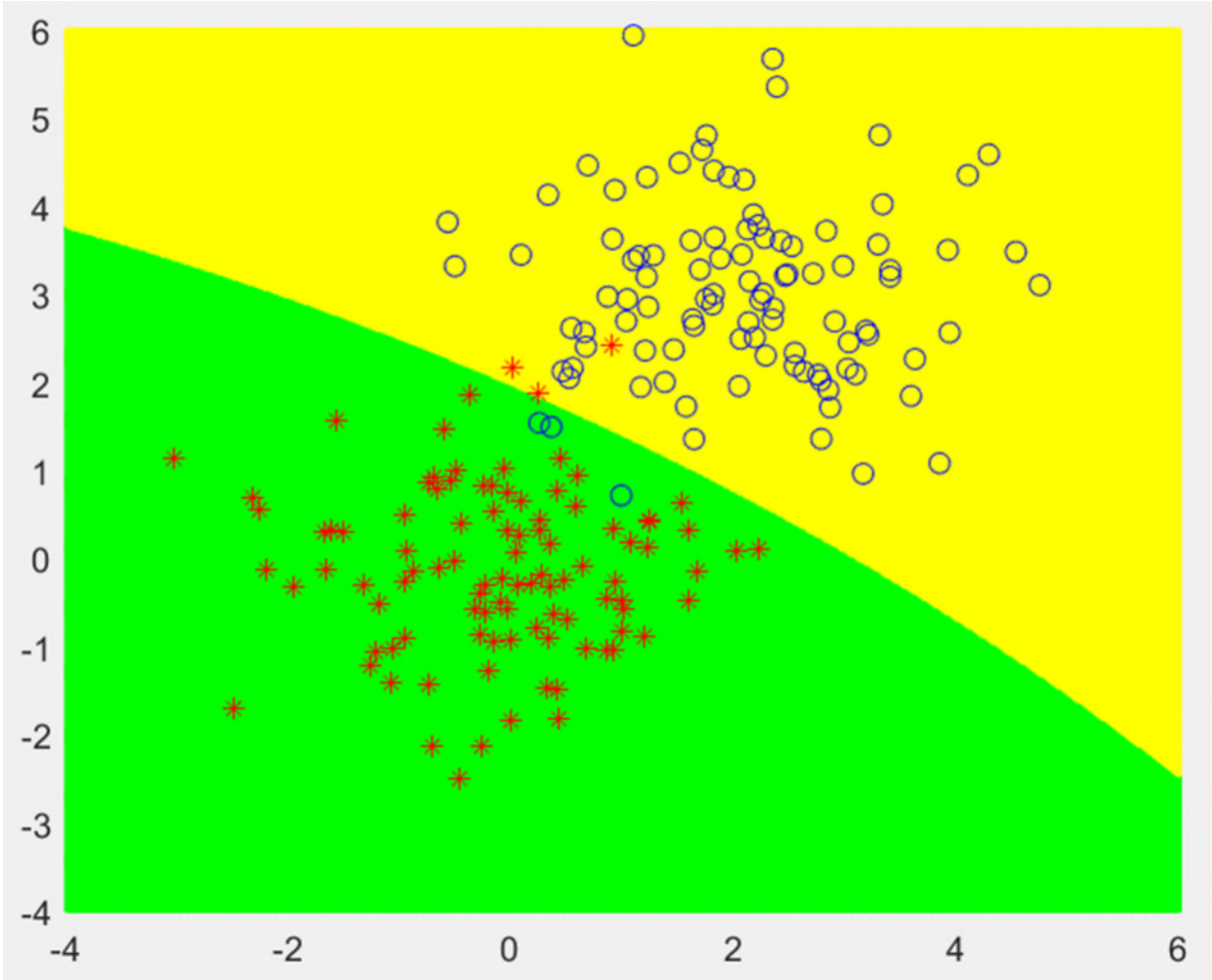
2) $\mathbf{x}$ is classified into class 2 if

$$g_2(\mathbf{x}) > g_1(\mathbf{x})$$

3) $\mathbf{x}$ is on the decision boundary (undecided) if

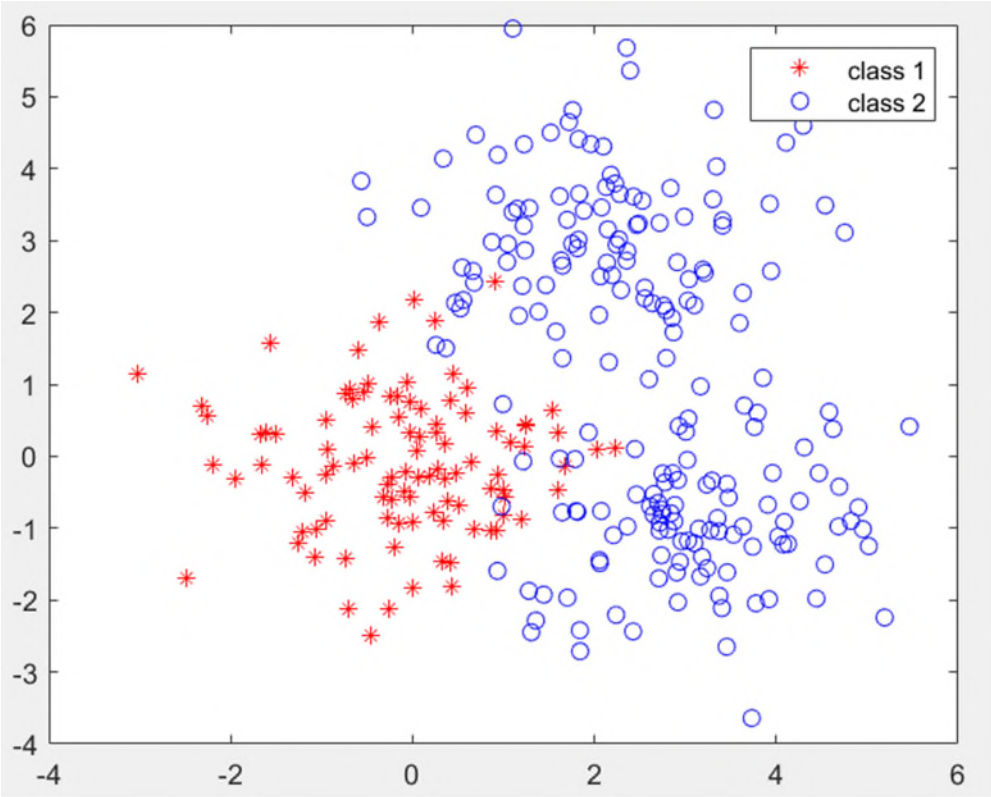
$$g_1(\mathbf{x}) = g_2(\mathbf{x})$$

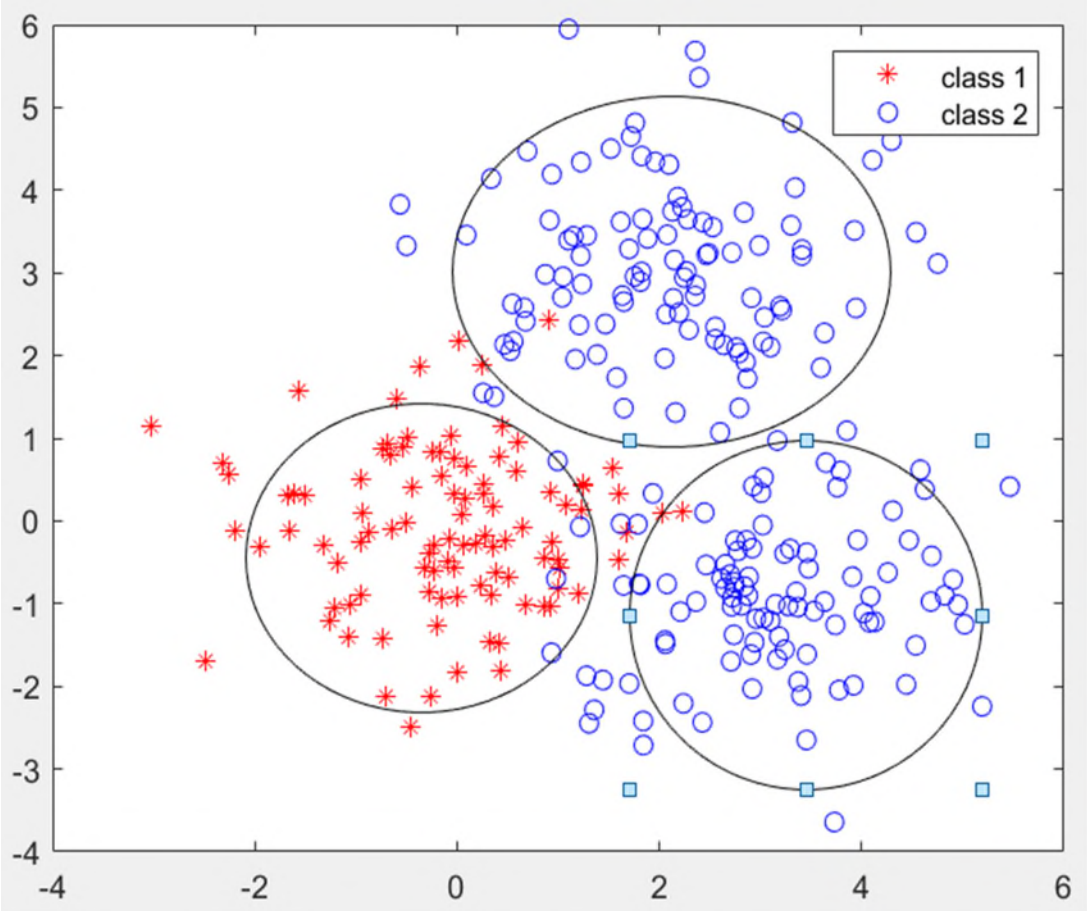
Decision boundary



Gaussian Mixture Models (GMM)

In some applications, the data distribution might be multimodal as shown below for class 2:




class 1
class 2
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-2
0
2
4
6

One way to exploit the nice properties of normal distributions for such multimodal data is to use the Gaussian Mixture Model (GMM).

A GMM model combines several normal distributions with different parameters:

$$p(\mathbf{x}|\omega) = \sum_{i=1}^{m} \alpha_i N(\mathbf{x}|\boldsymbol{\mu}_i, \mathbf{C}_i)$$

Where m is the number of Gaussian components, α_i is the weight of the i-th component, and

$$\sum_{i=1}^{m} \alpha_i = 1$$

Parameter estimation for GMM

Assume the data is generated from a Gaussian mixture model, however, the parameters of the distribution remain unknown. How do we learn the parameters?

As in the multivariate normal distribution, we may use the maximum-likelihood, or equivalently, the maximum-log-likelihood method:

$$l(\boldsymbol{\theta}) = \ln p(D|\boldsymbol{\theta}) = \sum_{k=1}^{n} \ln p(\mathbf{x}_k|\boldsymbol{\theta}) = \sum_{k=1}^{n} \ln \sum_{i=1}^{m} p(\mathbf{x}_k|\alpha_i, \boldsymbol{\mu}_i, \mathbf{C}_i)$$

First, we define a random variable $\gamma_{ik} = p(o_i|\mathbf{x}_k)$, i.e. the probability of $\mathbf{x}_k$ belonging to Gaussian component o_i.

Based on Bayes theorem, we have:

$$\gamma_{ik} = p(o_i|\mathbf{x}_k) = \frac{p(\mathbf{x}_k|o_i)P(o_i)}{\sum_{i=1}^{m} p(\mathbf{x}|o_i)P(o_i)} = \frac{p(\mathbf{x}_k|o_i)\alpha_i}{\sum_{i=1}^{m} p(\mathbf{x}_k|o_i)\alpha_i}$$

For the log likelihood function $l(\boldsymbol{\theta})$ to be maximum, its derivative with respect to $\alpha_i, \boldsymbol{\mu}_i, \mathbf{C}_i$ must be zero.

Let the derivative of $l(\boldsymbol{\theta})$ with respect to $\boldsymbol{\mu}_i$ be zero, we obtain:

$$\boldsymbol{\mu}_i = \frac{\sum_{k=1}^{n} \gamma_{ik}\, \mathbf{x}_k}{\sum_{k=1}^{n} \gamma_{ik}}$$

Similarly, let the derivative of $l(\boldsymbol{\theta})$ with respect to $\mathbf{C}_i$ and α_i be zero, we obtain:

$$\boldsymbol{\Sigma}_i = \frac{\sum_{k=1}^{n} \gamma_{ik} \, (\mathbf{x}_k - \boldsymbol{\mu}_i)(\mathbf{x}_k - \boldsymbol{\mu}_i)^T}{\sum_{k=1}^{n} \gamma_{ik}}$$

$$\alpha_i = \frac{1}{n} \sum_{k=1}^{n} \gamma_{ik}$$

Clearly, α_i is a function of γ_{ik}, and γ_{ik} is a function of α_i. Thus, the parameters cannot be estimated in closed form.

Expectation-Maximization (EM) algorithm can be used to solve the above problem.

The Expectation-Maximization (EM) algorithm is an iterative way to find maximum-likelihood estimates for model parameters

There are two basic steps in the EM algorithm, namely E Step or Expectation Step or Estimation Step and M Step or Maximization Step.

- Estimation step:

  Estimate γ_{ik} for the given values of $\alpha_i, \boldsymbol{\mu}_i, \mathbf{C}_i$

- Maximization step:

  Update $\alpha_i, \boldsymbol{\mu}_i, \mathbf{C}_i$ using the maximum-likelihood method

EM Algorithm for GMM parameter estimation

(1) Initialization

Set $j = 0$ and initialize $\hat{\alpha}_i(0), \hat{\boldsymbol{\mu}}_i(0), \hat{\mathbf{C}}_i(0)$ with random values

(2) Estimation

Set iteration $j = j + 1$. Estimate $\gamma_{ik}(j)$ based on values of $\hat{\alpha}_i(j-1), \hat{\boldsymbol{\mu}}_i(j-1), \hat{\mathbf{C}}_i(j-1)$:

$$p(\mathbf{x_k}|o_i) = \frac{1}{(2\pi)^{d/2}\sqrt{|\hat{\mathbf{C}}_i(j-1)|}} \exp\left[-\frac{1}{2}(\mathbf{x}_k - \hat{\boldsymbol{\mu}}_i(j-1))^T \hat{\mathbf{C}}_i^{-1}(j-1)\,(\mathbf{x}_k - \hat{\mathbf{u}}_i(j-1))\right]$$

$$\gamma_{ik}(j) = p(o_i|\mathbf{x}_k) = \frac{p(\mathbf{x}_k|o_i)\hat{\alpha}_i(j-1)}{\sum_{i=1}^{m} p(\mathbf{x}_k|o_i)\hat{\alpha}_i(j-1)}$$

(3) Maximization

Update values of $\hat{\boldsymbol{\mu}}_i(j), \hat{\mathbf{C}}_i(j), \hat{\alpha}_i(j)$ with $\gamma_{ik}(j)$:

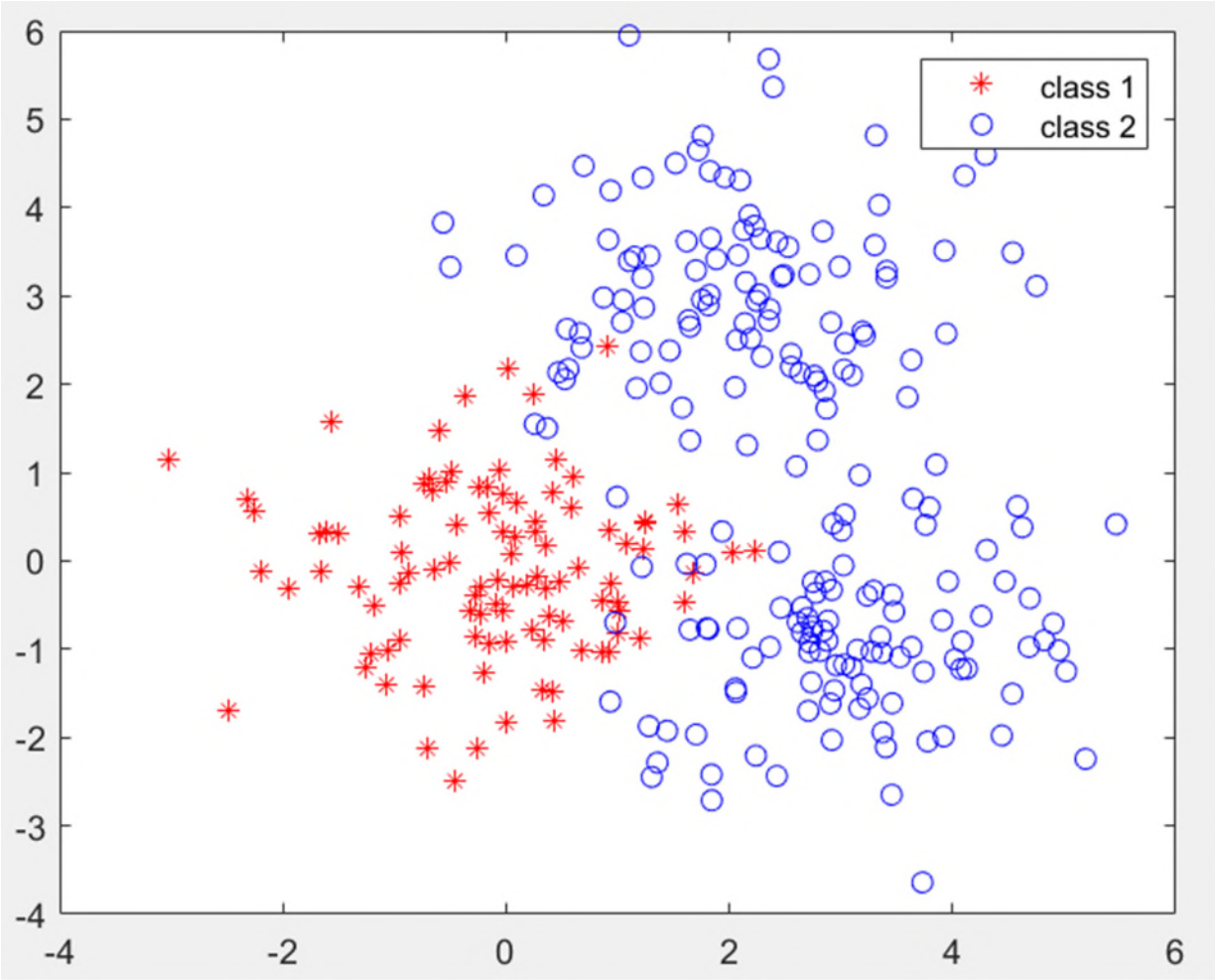
$$\hat{\mathbf{u}}_i(j) = \frac{\sum_{k=1}^{n} \gamma_{ik}(j)\, \mathbf{x}_k}{\sum_{k=1}^{n} \gamma_{ik}(j)}$$

$$\hat{\mathbf{C}}_i(j) = \frac{\sum_{k=1}^{n} \gamma_{ik}(j)[\mathbf{x}_k - \hat{\mathbf{u}}_i(j)][\mathbf{x}_k - \hat{\mathbf{u}}_i(j)]^T}{\sum_{k=1}^{n} \gamma_{ik}(j)}$$

$$\hat{\alpha}_i(j) = \frac{1}{n} \sum_{k=1}^{n} \gamma_{ik}(j)$$

Return to Step (2) Estimation until the stopping criterion (such as number of iterative step is reached) is satisfied.

Example 2



To build a GMM model for samples in class 2, we use the EM algorithm.

Initialization. Set $j = 0$, assign random values to the GMM model parameters:

$$\hat{\boldsymbol{\mu}}_{21}(0) = \begin{bmatrix} -1.7729 \\ -0.0725 \end{bmatrix} \qquad \hat{\boldsymbol{\mu}}_{22}(0) = \begin{bmatrix} 1.7175 \\ 0.8198 \end{bmatrix}$$

$$\hat{\mathbf{C}}_{21}(\mathbf{0}) = \begin{bmatrix} -1.0055 & 1.7906 \\ -1.1064 & 2.1624 \end{bmatrix}$$

$$\hat{\mathbf{C}}_{22}(\mathbf{0}) = \begin{bmatrix} -0.8193 & 1.9630 \\ -0.0370 & -0.5403 \end{bmatrix}$$

$$\hat{\alpha}_{21}(0) = 0.4 \qquad \hat{\alpha}_{22}(0) = 0.6$$

Estimation and maximization steps

After 1000 repeats of the estimation and maximization steps, we obtain the following estimates:

$$\hat{\boldsymbol{\mu}}_{21}(1000) = \begin{bmatrix} 3.0992 \\ -0.9364 \end{bmatrix} \qquad \hat{\boldsymbol{\mu}}_{22}(1000) = \begin{bmatrix} 2.0412 \\ 3.0483 \end{bmatrix}$$

$$\hat{\mathbf{C}}_{21}(1000) = \begin{bmatrix} 1.0245 & 0.1386 \\ 0.1386 & 0.7862 \end{bmatrix}$$

$$\hat{\mathbf{C}}_{22}(1000) = \begin{bmatrix} 1.1613 & -0.0057 \\ -0.0057 & 1.0778 \end{bmatrix}$$

$$\hat{\alpha}_{21}(1000) = 0.4952 \qquad \hat{\alpha}_{22}(1000) = 0.5048$$

With the estimated parameters of GMM, the class-conditional probability density function of class 2 is obtained as follows:

$$p(\mathbf{x}|\omega_2) = \sum_{i=1}^{2} \alpha_{2i} N\left(\mathbf{x}|\hat{\boldsymbol{\mu}}_{2i}, \hat{\mathbf{C}}_{2i}\right)$$

$$= \frac{0.4952}{2\pi\sqrt{|\hat{\mathbf{C}}_{21}|}} \exp\left[-\frac{1}{2}(\mathbf{x}-\hat{\boldsymbol{\mu}}_{21})^T \hat{\mathbf{C}}_{21}^{-1}(\mathbf{x}-\hat{\mathbf{u}}_{21})\right]$$

$$+ \frac{0.5048}{2\pi\sqrt{|\hat{\mathbf{C}}_{22}|}} \exp\left[-\frac{1}{2}(\mathbf{x}-\hat{\boldsymbol{\mu}}_{22})^T \hat{\mathbf{C}}_{22}^{-1}(\mathbf{x}-\hat{\mathbf{u}}_{22})\right]$$

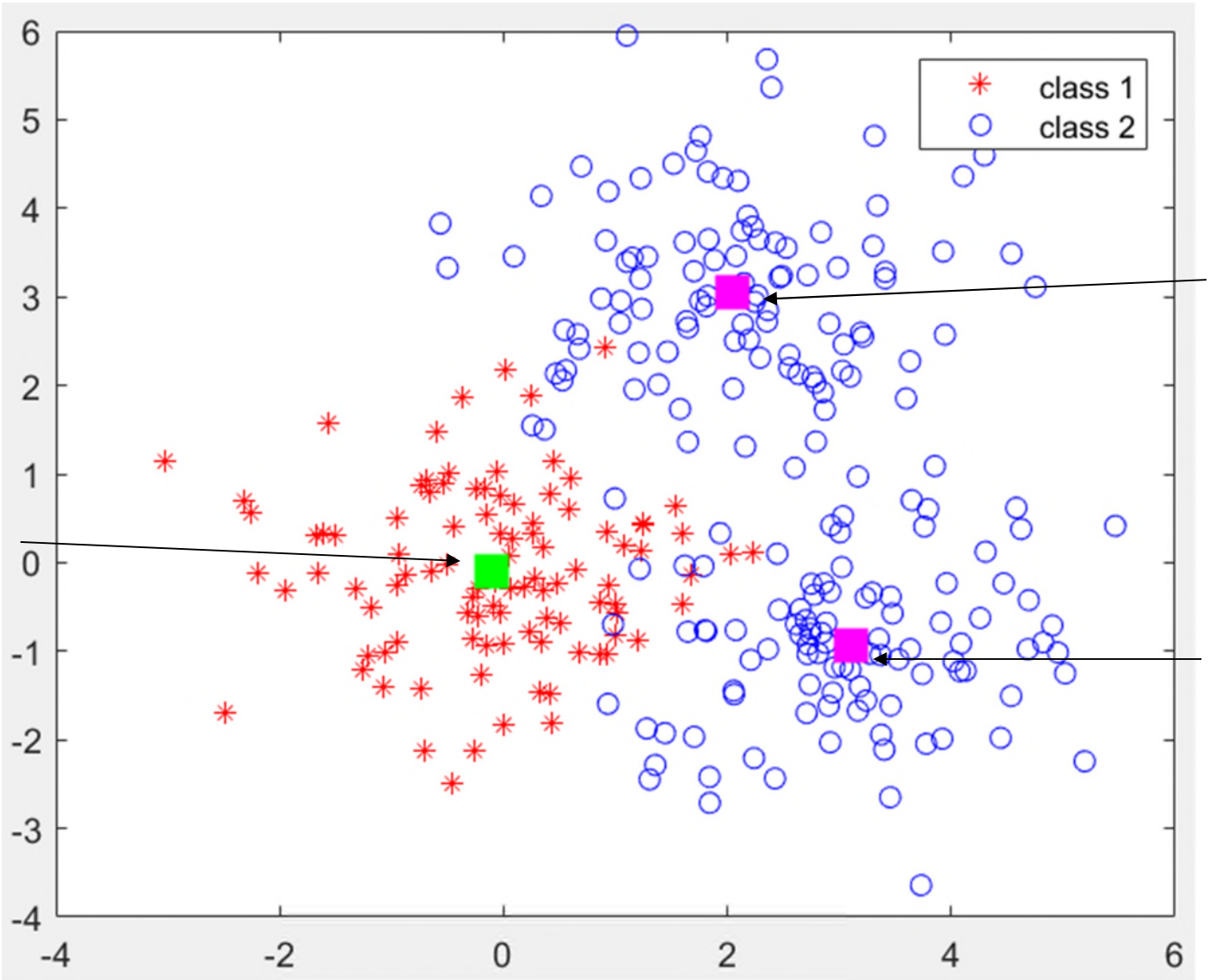
For class 1, we can use the maximum-likelihood method to estimate the mean vector $\hat{\boldsymbol{\mu}}_1$ and covariance matrix $\hat{\mathbf{C}}_1$ of the single Gaussian function:

$$\hat{\boldsymbol{\mu}}_1 = \begin{bmatrix} -0.1055 \\ -0.0974 \end{bmatrix}$$

$$\hat{\mathbf{C}}_1 = \begin{bmatrix} 1.0253 & -0.0036 \\ -0.0036 & 0.8880 \end{bmatrix}$$

Then the conditional probability density function of class 1 is obtained as follows:

$$p(\mathbf{x}|\omega_1) = \frac{1}{2\pi\sqrt{|\hat{\mathbf{C}}_1|}} \exp\left[-\frac{1}{2}(\mathbf{x} - \hat{\boldsymbol{\mu}}_1)^T \hat{\mathbf{C}}_1^{-1} (\mathbf{x} - \hat{\mathbf{u}}_1)\right]$$


class 1
class 2
$\hat{\mu}_{22}$
$\hat{\mu}_1$
$\hat{\mu}_{21}$

With the prior probabilities of the two classes :

$$p(\omega_1) = \frac{100}{300} = \frac{1}{3}$$

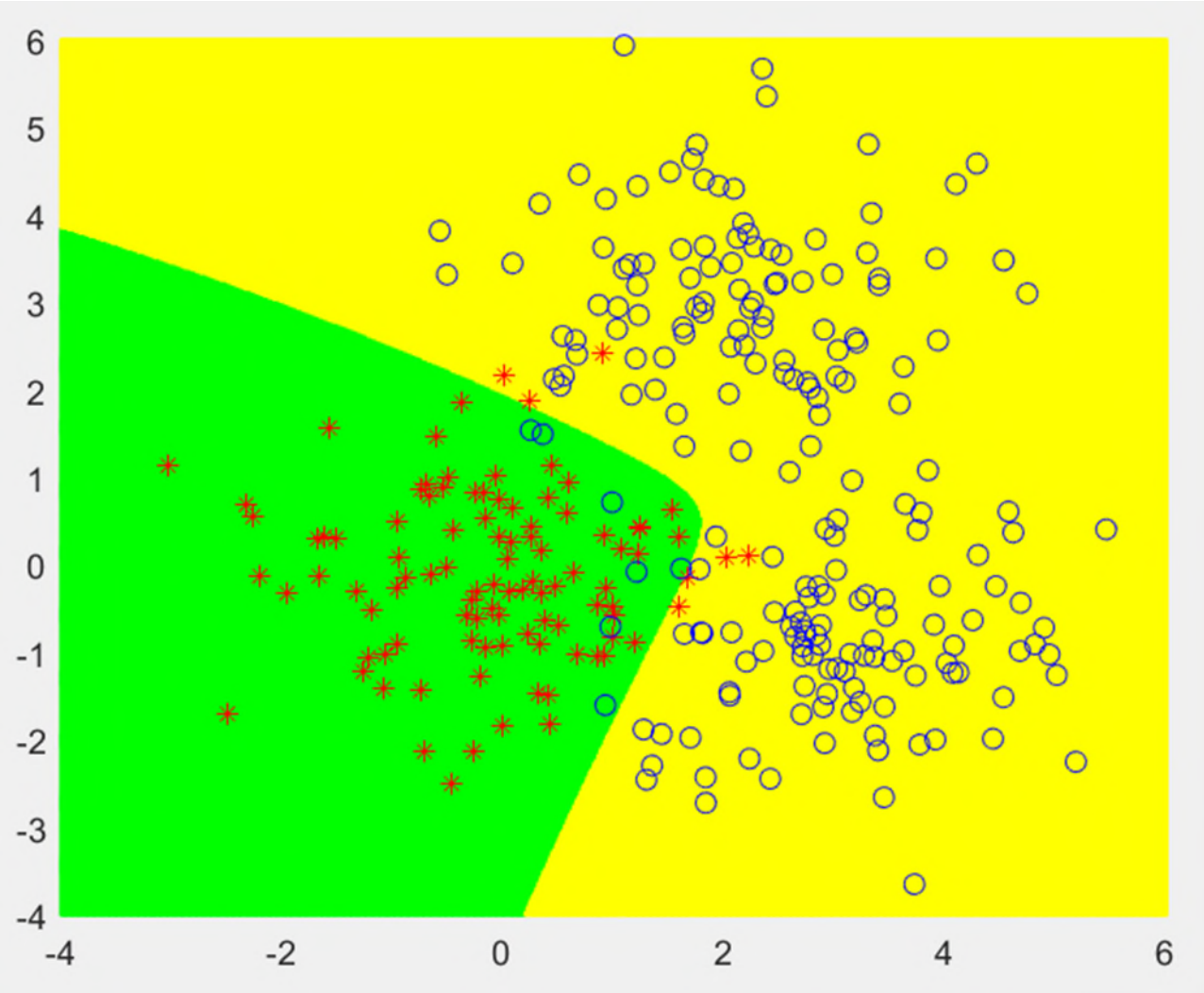
$$p(\omega_2) = \frac{200}{300} = \frac{2}{3}$$

We obtain the following discriminant functions to classify the samples in the two classes:

$$g_1(\mathbf{x}) = p(\omega_1)p(\mathbf{x}|\omega_1) = \frac{1}{3}p(\mathbf{x}|\omega_1)$$

$$g_2(\mathbf{x}) = p(\omega_2)p(\mathbf{x}|\omega_2) = \frac{2}{3}p(\mathbf{x}|\omega_2)$$

Decision boundary and 14 misclassifications



If a single Gaussian function is used to model the conditional class probability density function for class 2, by using the maximum-likelihood estimation, we have:

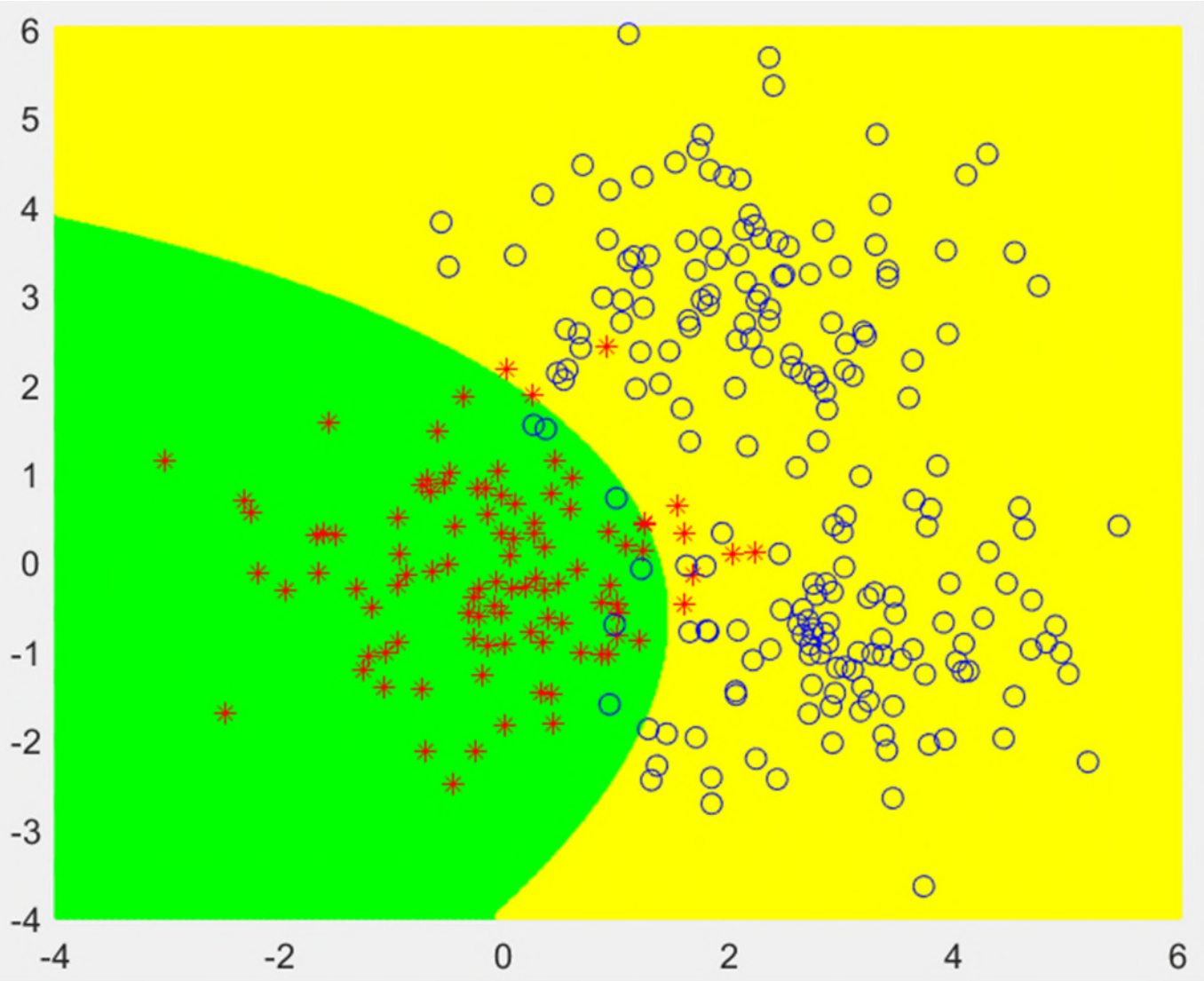
$$\hat{\boldsymbol{\mu}}_2 = \begin{bmatrix} 2.5702 \\ 1.0559 \end{bmatrix}$$

$$\hat{\mathbf{C}}_2 = \begin{bmatrix} 1.3615 & -0.9607 \\ -0.9607 & 4.8657 \end{bmatrix}$$

Then the conditional probability density function of class 2 is obtained as follows:

$$p(\mathbf{x}|\omega_2) = \frac{1}{2\pi\sqrt{|\hat{\mathbf{C}}_2|}} \exp\left[-\frac{1}{2}(\mathbf{x} - \hat{\boldsymbol{\mu}}_2)^T \hat{\mathbf{C}}_2^{-1} (\mathbf{x} - \hat{\mathbf{u}}_2)\right]$$

Decision boundary and 15 misclassifications

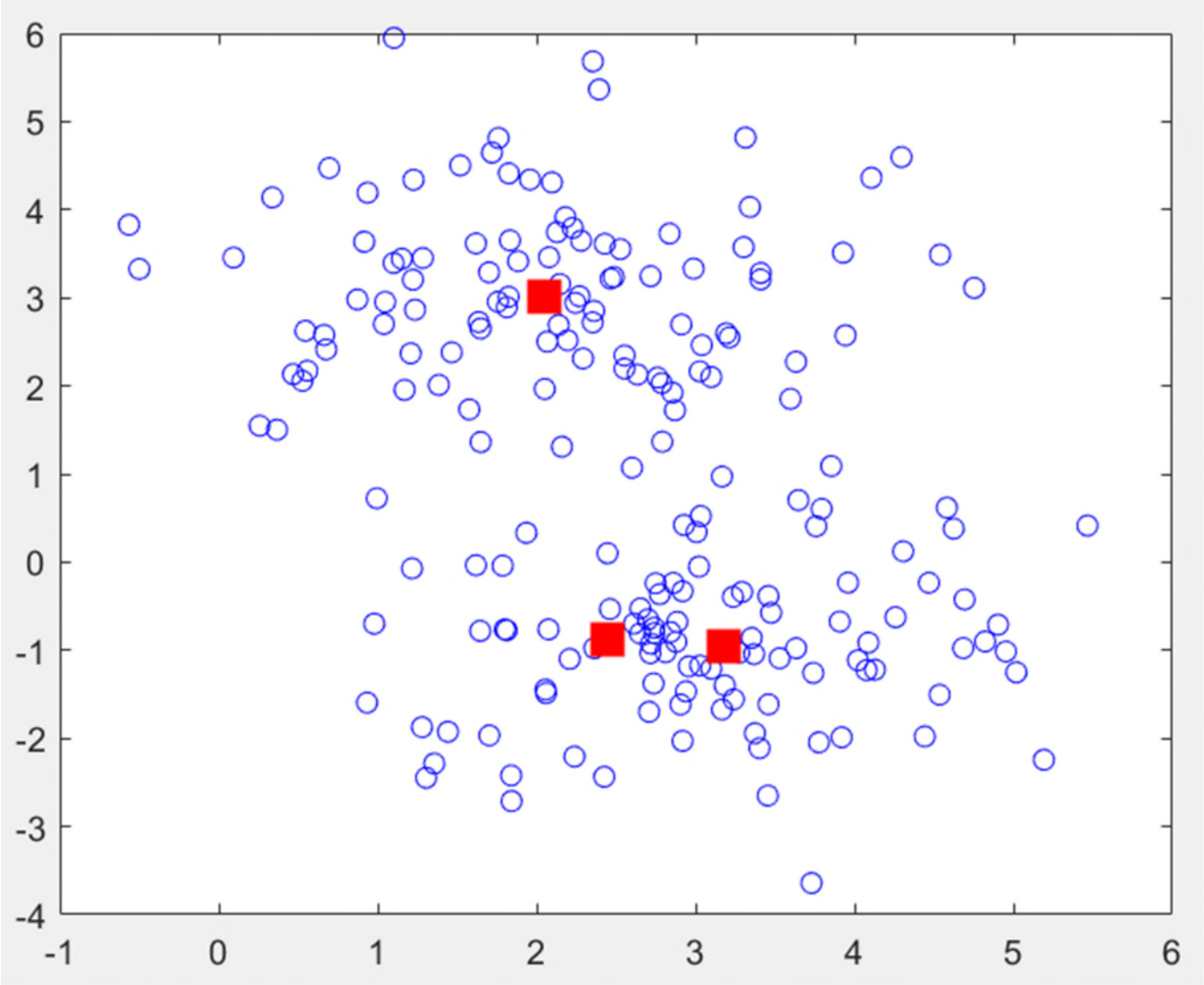


In practice, we may not know the number of Gaussian components underlying the data, in particular in high dimensional space where data is hardly to be visualized.

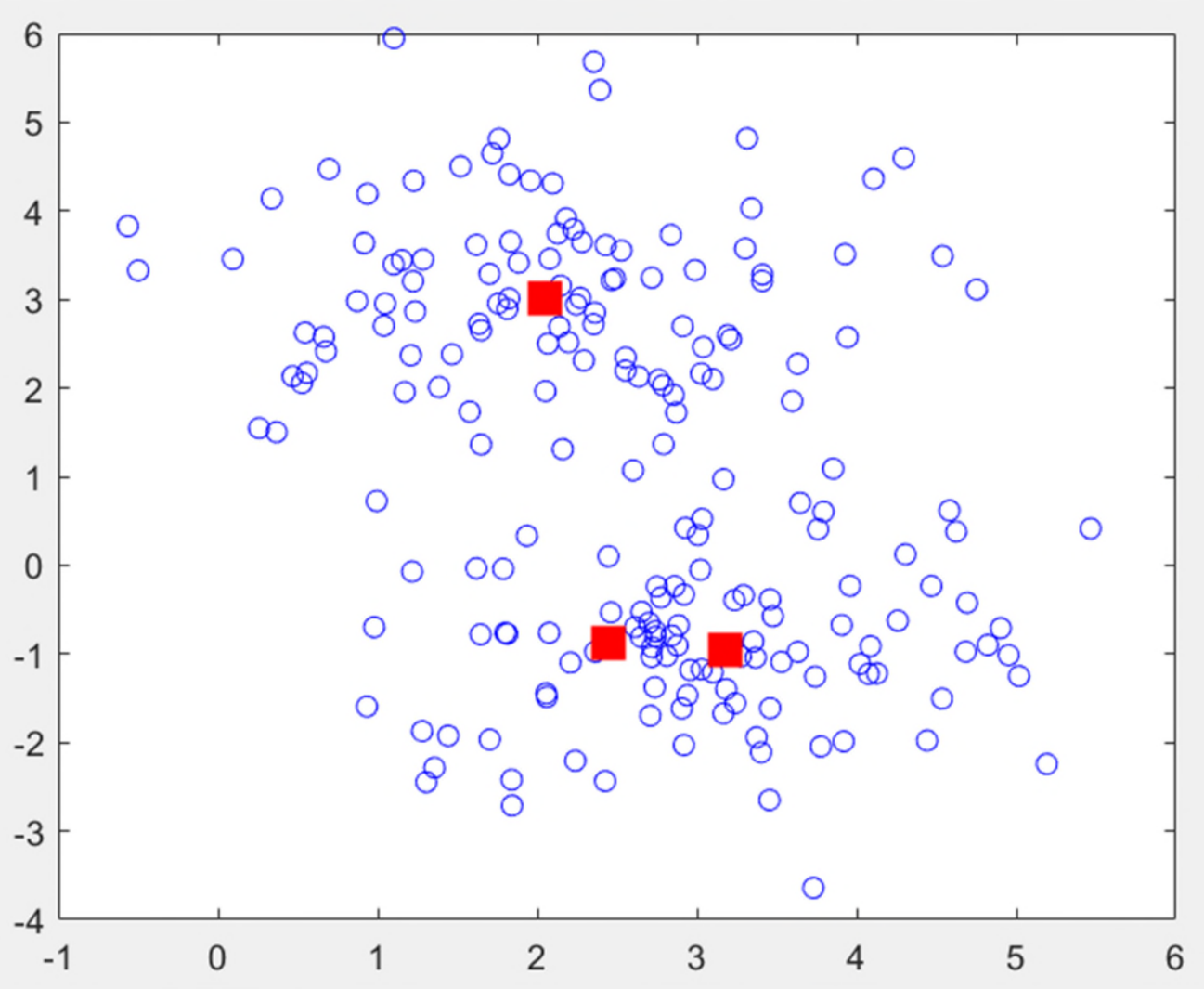
For class 2, assuming we do not know there are 2 Gaussian components. We may first guess the number of Gaussian components and then use the EM algorithm to find the parameters.

For example, we guess there are 3 Gaussian components underlying class 2:

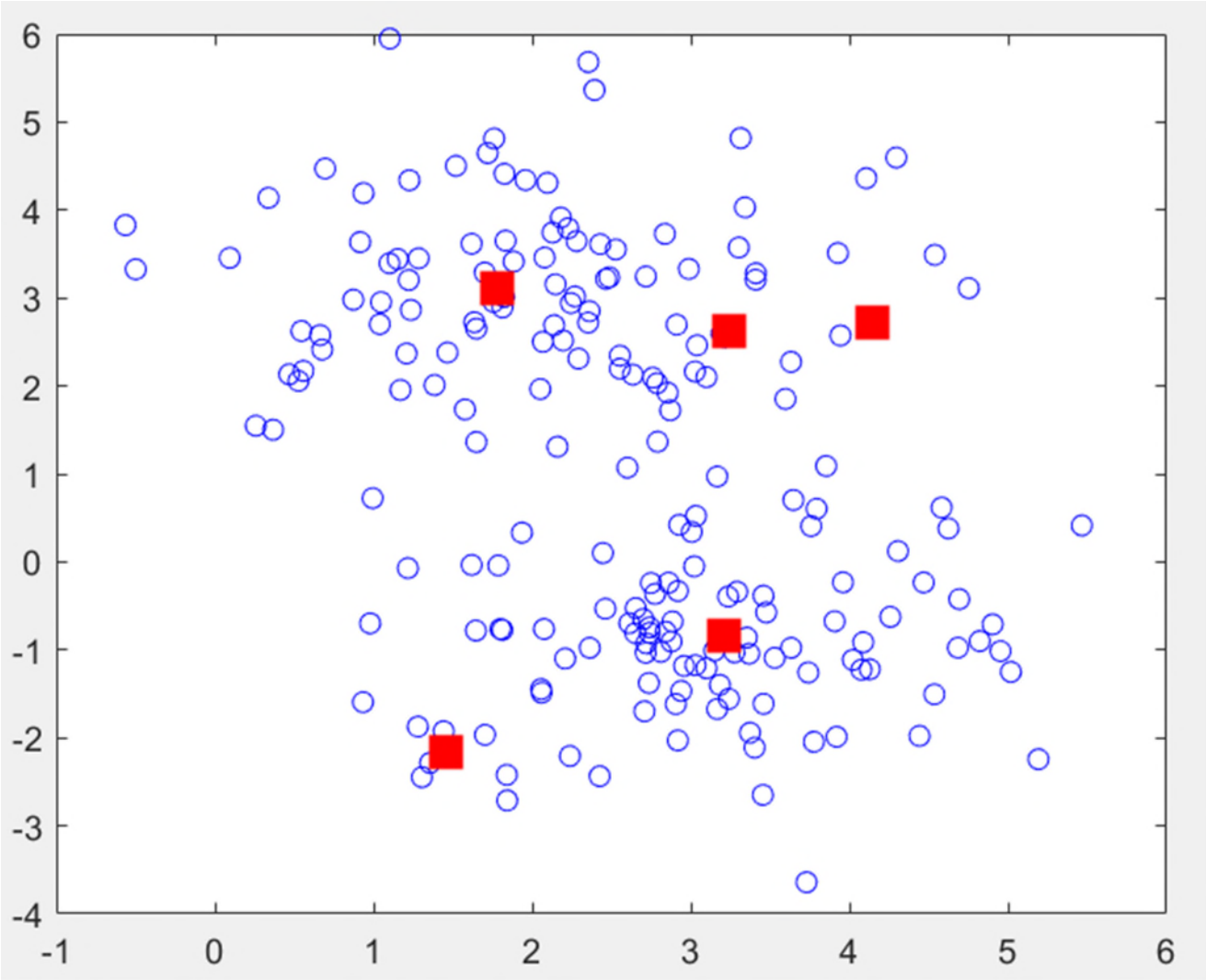
3 Gaussian components are used for class 2



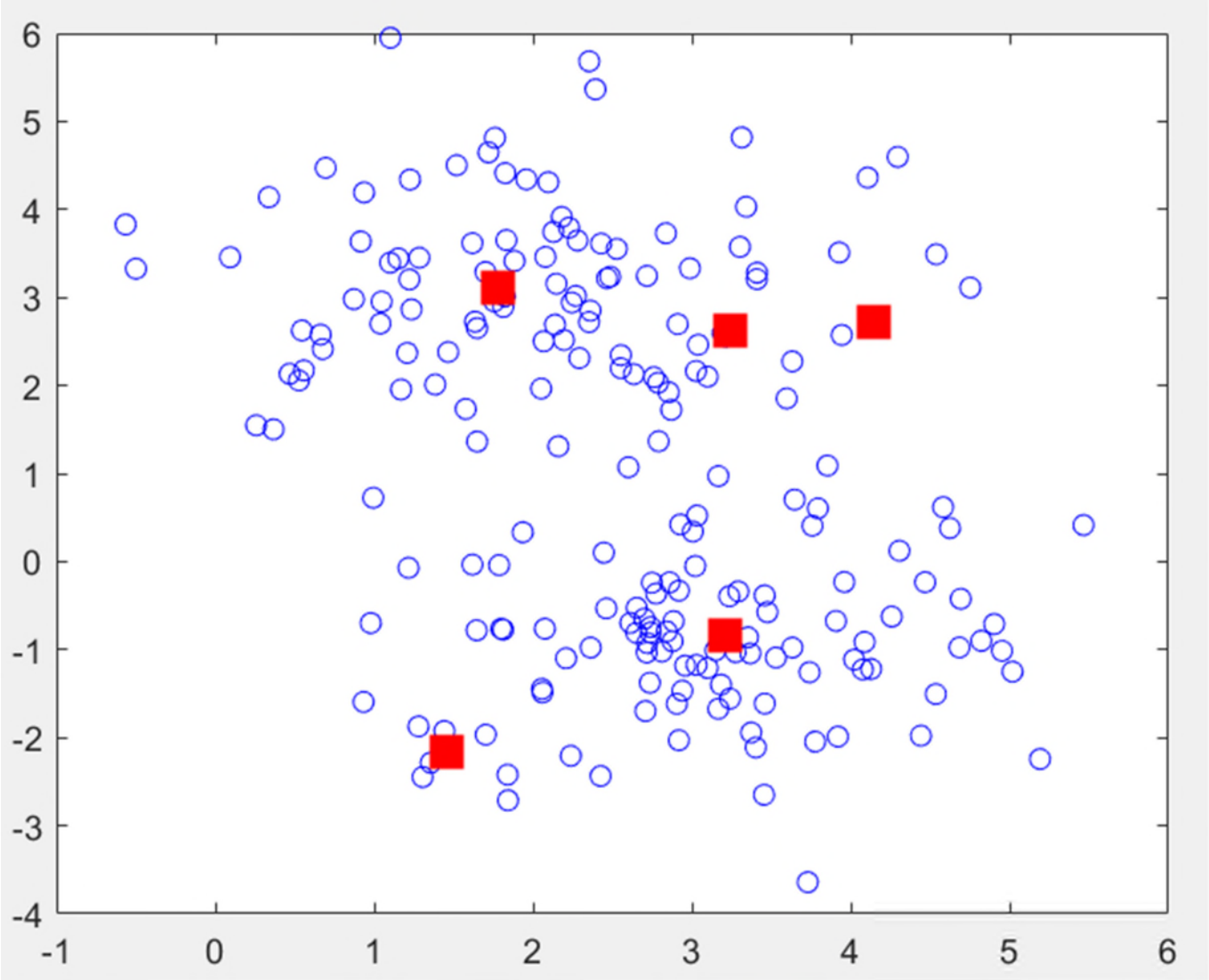
$$\alpha = [0.0523 \quad 0.5040 \quad 0.4437]$$



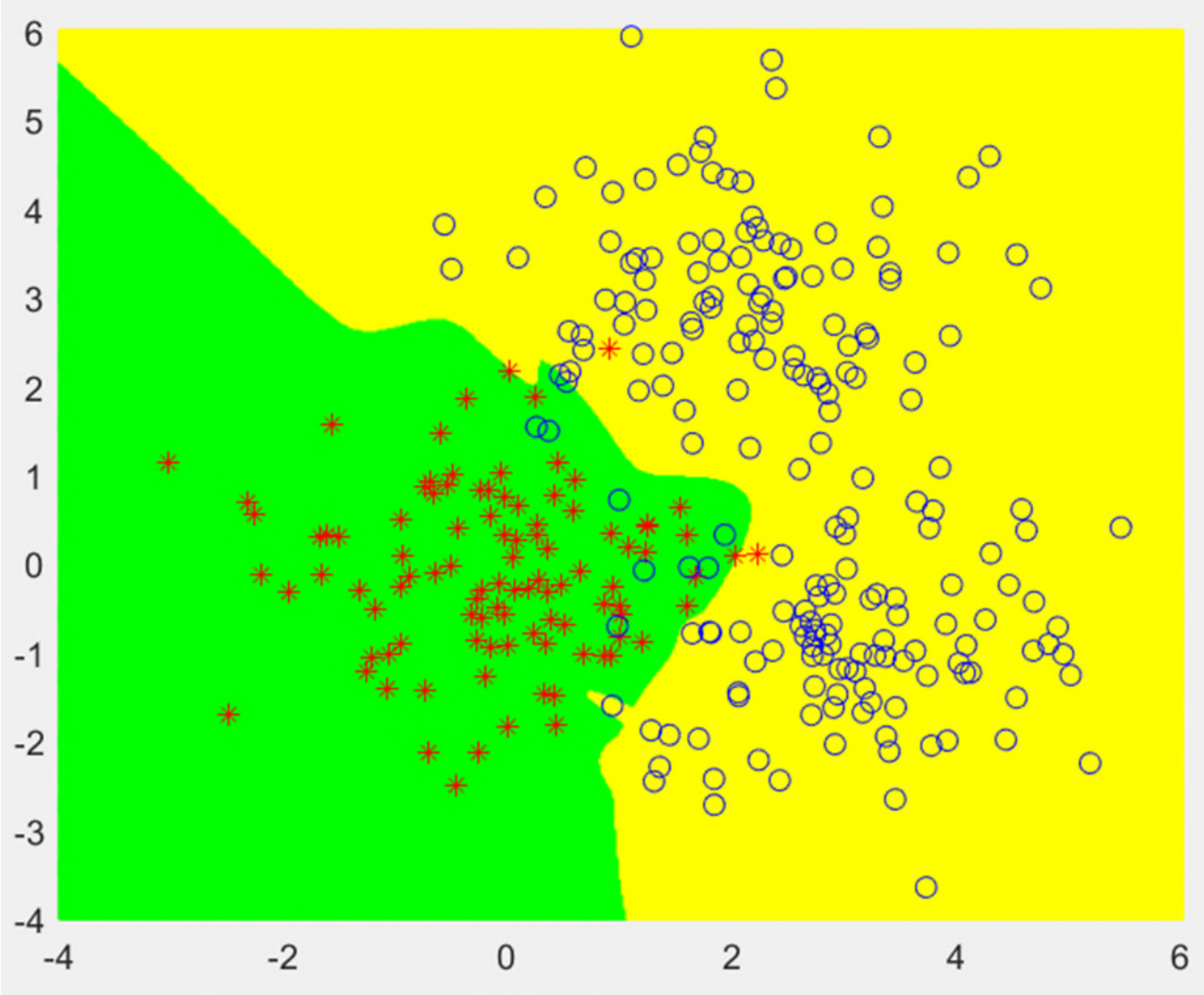
5 Gaussian components are used for class 2



$$\alpha = [0.4670 \quad 0.0333 \quad 0.0564 \quad 0.0206 \quad 0.4227]$$



Decision boundary when 2 and 5 Gaussian components are assumed for class 1 and 2, respectively



Observations:

- When excessive number of Gaussian components are used, the decision boundary (surface) becomes more complex, which may produce better performance on the training data, but worse performance on unseen testing data (i.e. overfitting problem).

- The use of excessive number of Gaussian components results in some small alpha values. This property may be used to determine suitable number of Gaussian components so as to alleviate the overfitting problem.

Naïve Bayes

Based on Bayes Theorem, we have

$$p(\omega_j|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_j)p(\omega_j)}{p(\mathbf{x})}$$

Assuming $\mathbf{x}$ has d features $x_1, x_2,\ldots,x_d$, then:

$$\begin{aligned}
&p(\mathbf{x}|\omega_j)p(\omega_j) \\
&= p(\mathbf{x}, \omega_j) \\
&= p(x_1, x_2, \cdots, x_d, \omega_j) \\
&= p(x_1|x_2, x_3, \cdots, x_d, \omega_j)p(x_2, x_3, \cdots, x_d, \omega_j) \\
&= p(x_1|x_2, x_3, \cdots, x_d, \omega_j)p(x_2|x_3, \cdots, x_d, \omega_j)p(x_3, \cdots, x_d, \omega_j) \\
&= p(x_1|x_2, x_3, \cdots, x_d, \omega_j)p(x_2|x_3, \cdots, x_d, \omega_j) \cdots p(x_d|\omega_j)p(\omega_j)
\end{aligned}$$

Assuming that the individual features are independent from each other. Then:

$$p(x_1|x_2, x_3, \cdots, x_d, \omega_j) = p(x_1|\omega_j)$$

$$\vdots$$

$$p(x_i|x_{i+1}, \cdots, x_d, \omega_j) = p(x_i|\omega_j)$$

Thus, the joint probability of $\mathbf{x}$ and ω_j is:

$$p(\mathbf{x}|\omega_j)p(\omega_j) = p(x_1|\omega_j)p(x_2|\omega_j)\cdots p(x_n|\omega_j)p(\omega_j)$$

$$= \prod_{i=1}^{d} p(x_i|\omega_j)p(\omega_j)$$

Then the posterior probability is:

$$p(\omega_j|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_j)p(\omega_j)}{p(\mathbf{x})} = \frac{\prod_{i=1}^{d} p(x_i|\omega_j)p(\omega_j)}{p(\mathbf{x})}$$

Types of Naïve Bayes Classifier

(1) Gaussian Naive Bayes - used for continuous data.

(2) Bernoulli Naive Bayes: used for discrete data whose features have binary values.

(3) Multinomial Naive Bayes: often used for text classification where the count of words in the text is used to represent the text.

Gaussian Naive Bayes

Gaussian Naive Bayes is used for continuous data. For each continuous feature, assuming it follows normal distribution, then the class conditional probability density function is as follow:

$$p(x|\omega_j) = \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left[-\frac{1}{2}\left(\frac{x-\mu_j}{\sigma_j}\right)^2\right]$$

Where μ_j and σ_j denote the mean and standard deviation of feature x for class ω_j, which can be easily estimated from samples of class ω_j using the maximum likelihood method.

Bernoulli Naive Bayes

Bernoulli Naive Bayes is used for discrete data and it works on Bernoulli distribution. The main characteristic of Bernoulli Naive Bayes is that it assumes binary values like true or false, yes or no, success or failure, 0 or 1 and so on.

Bernoulli distribution

As we deal with binary values, let's consider r as the probability of success and q as probability of failure, then $q = 1 - r$. For a random variable in Bernoulli distribution,

$$p(x) = P(X = x) = \begin{cases} r & x = 1 \\ 1 - r & x = 0 \end{cases}$$

Consider the following dataset showing the result whether a person Pass or Fail in the exam:

Confident	Studied	Sick	Result
Yes	No	No	Fail
Yes	No	Yes	Pass
No	Yes	Yes	Fail
No	Yes	No	Pass
Yes	Yes	Yes	Pass

Our task is to classify instance **x** with Confident=Yes, Studied=Yes and Sick=No.

Assuming that the result of Pass and Fail are denoted by ω_1 and ω_2, respectively. Features Confident, Studied and Sick are denoted by X_1, X_2 and X_3 respectively. Then:

$$p(\omega_j|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_j)p(\omega_j)}{p(\mathbf{x})} = \frac{\prod_{i=1}^{3} p(x_i|\omega_j)p(\omega_j)}{p(\mathbf{x})}$$

For a sample with $x_1 = Yes,\ x_2 = Yes,\ x_3 = No$, to obtain the posterior probabilities, we need to first find:

- prior probabilities $p(\omega_j), j = 1,2$
- class conditional probabilities

$$p(x_1 = Yes|\omega_j)$$

$$p(x_2 = Yes|\omega_j)$$

$$p(x_3 = No|\omega_j)$$

First, we calculate the prior probabilities:

$$p(\omega_1) = \frac{3}{5} = 0.6$$

$$p(\omega_2) = \frac{2}{5} = 0.4$$

Second, we calculate the class conditional probability for each individual feature.

$$p(x_1 = Yes|\omega_1) = \frac{2}{3}$$

$$p(x_2 = Yes|\omega_1) = \frac{2}{3}$$

$$p(x_3 = No|\omega_1) = \frac{1}{3}$$

Confident	Studied	Sick	Result
Yes	No	No	Fail
Yes	No	Yes	Pass
No	Yes	Yes	Fail
No	Yes	No	Pass
Yes	Yes	Yes	Pass

Similarly,

$$p(x_1 = Yes|\omega_2) = \frac{1}{2}$$

$$p(x_2 = Yes|\omega_2) = \frac{1}{2}$$

$$p(x_3 = No|\omega_2) = \frac{1}{2}$$

Hence:

$$p(\mathbf{x}|\omega_1)p(\omega_1) = \frac{2}{3} \times \frac{2}{3} \times \frac{1}{3} \times \frac{3}{5} = \frac{4}{45} = 0.088$$

$$p(\mathbf{x}|\omega_2)p(\omega_2) = \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} \times \frac{2}{5} = \frac{1}{20} = 0.05$$

$p(\mathbf{x})$ is a common denominator, we can ignore it.

Since

$$p(\mathbf{x}|\omega_1)p(\omega_1) > p(\mathbf{x}|\omega_2)p(\omega_2)$$

The instance $\mathbf{x}$ with Confident=Yes, Studied=Yes and Sick=No can be predicted as “Pass”

Multinomial Naïve Bayes

Multinominal Naïve Bayes is widely used for document (text) classification. Consider the following problem with 5 labelled training samples:

Text	Category
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

Our task is to classify the testing sample "A very close game" into one of the two categories: Sports and Not sports.

Based on Naïve Bayes,

$$p(Sports|A\ very\ close\ game)$$

$$= \frac{p(A\ very\ close\ game|Sports)p(Sports)}{p(A\ very\ close\ game)}$$

$$= \frac{p(A|Sports)p(very|Sports)\cdots p(game|Sports)p(Sports)}{p(A\ very\ close\ game)}$$

Similarly,

$$p(Not\ sports|A\ very\ close\ game)$$

$$= \frac{p(A\ very\ close\ game|Not\ sports)p(Not\ sports)}{p(A\ very\ close\ game)}$$

$$= \frac{p(A|Not\ sports)\cdots p(game|Not\ sports)p(Not\ sports)}{p(A\ very\ close\ game)}$$

Assuming the i^{th} word is denoted by x_i, and the categories Sports and Not sports are denoted by ω_1 and ω_2, respectively. The class conditional probability $p(x_i|\omega_j)$ can be estimated from training data using the following formula:

$$p(x_i|\omega_j) = \frac{counts(x_i, \omega_j)}{\sum_{k=1}^{d} counts(x_k, \omega_j)}$$

Where $counts(x_k, \omega_j)$ denotes the total number of occurrence of word x_k in the training data belonging to category ω_j. d is the number of unique words in all training data.

But there is one issue in the above estimation. If a word is not seen in the training data, then the conditional probability will be zero, leading to zero posterior probability.

Text	Category
“A great game”	Sports
“The election was over”	Not sports
“Very clean match”	Sports
“A clean but forgettable game”	Sports
“It was a close election”	Not sports

For example, for word “close”,

$$p(close|Sports) = \frac{0}{11} = 0$$

To address the issue, we can use the so-called Laplace smoothing (add 1):

$$p(x_i|\omega_j) = \frac{counts(x_i, \omega_j) + 1}{\sum_{k=1}^{d}(counts(x_k, \omega_j) + 1)}$$

$$= \frac{counts(x_i,\omega_j)+1}{\left(\sum_{k=1}^{d} counts(x_k,\omega_j)\right)+d}$$

After Laplace smoothing,

$$p(close|Sports) = \frac{0+1}{11+14} = \frac{1}{25}$$

Accordingly, we can obtain the other class conditional probabilities:

Word	P(word | Sports)	P(word | Not Sports)
a	$\frac{2+1}{11+14}$	$\frac{1+1}{9+14}$
very	$\frac{1+1}{11+14}$	$\frac{0+1}{9+14}$
close	$\frac{0+1}{11+14}$	$\frac{1+1}{9+14}$
game	$\frac{2+1}{11+14}$	$\frac{0+1}{9+14}$

$$p(Sports|A\ very\ close\ game)$$

$$= \frac{p(A|Sports)p(very|Sports)\cdots p(game|Sports)p(Sports)}{p(A\ very\ close\ game)}$$

$$= \frac{\frac{3}{25}\times\frac{2}{25}\times\frac{1}{25}\times\frac{3}{25}\times\frac{3}{5}}{p(A\ very\ close\ game)} = \frac{2.765\times10^{-5}}{p(A\ very\ close\ game)}$$

$$p(Not\ sports|A\ very\ close\ game)$$

$$= \frac{p(A|Not\ sports) \cdots p(game|Not\ sports)p(Not\ sports)}{p(A\ very\ close\ game)}$$

$$= \frac{\frac{2}{23} \times \frac{1}{23} \times \frac{2}{23} \times \frac{1}{23} \times \frac{2}{5}}{p(A\ very\ close\ game)} = \frac{5.718 \times 10^{-6}}{p(A\ very\ close\ game)}$$

Because

$$p(Sports|A\ very\ close\ game) > p(Not\ sports|A\ very\ close\ game)$$

The text “A very close game” is classified as Sports.

Notes:

(1) In spite of the apparently over-simplified assumptions, naive Bayes classifiers have worked quite well in many real-world situations such as document classification and spam filtering. They require a small amount of training data to estimate the necessary parameters.

(2) Naive Bayes classifiers can be extremely fast compared to more sophisticated methods. The decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one-dimensional distribution. This in turn helps to alleviate problems stemming from the curse of dimensionality.