# MLT: Week 8

## GENERATIVE MODELS - NAIVE BAYES

Sherry Thomas

# Naive Bayes Algorithm

# Naive Bayes Algorithm

Given a dataset $\{\mathrm{x}_1, \ldots, \mathrm{x}_n\}$ where $\mathrm{x}_i \in \{0,1\}^d$,
let $\{y_1, \ldots, y_n\}$ be the labels, where $y_i \in \{0,1\}$.

$$\mathrm{X} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathrm{y} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

The Naive Bayes Algorithm is characterized by two main things:

1. Class Conditional Independence Assumption
2. Bayes Rule

The Naive Bayes Algorithm is characterized by two main things:

1. Class Conditional Independence Assumption
2. Bayes Rule

Based on the first characteristic, we get $2d+1$ parameters where $d$ is the number of dimensions of the features set.
Using MLE, the estimates of the parameters are given by,

$$\widehat{p} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

The Naive Bayes Algorithm is characterized by two main things:

1. Class Conditional Independence Assumption
2. Bayes Rule

Based on the first characteristic, we get $2d+1$ parameters where $d$ is the number of dimensions of the features set.
Using MLE, the estimates of the parameters are given by,

$$\widehat{p} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

$$\widehat{p}_j^{\,y} = \frac{\sum_{i=1}^{n} \mathbb{I}(f_j^i = 1, y_i = y)}{\sum_{i=1}^{n} \mathbb{I}(y_i = y)} \quad \forall j \in \{1, 2, \ldots, d\} \quad \forall y \in \{0, 1\}$$

Estimating the GMM parameters for our dataset would result in parameters whose values are zero. Therefore, we need to apply **Laplace Smoothing**.
After applying it on our dataset, we get,

$$X = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \quad y = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Estimating the GMM parameters for our dataset would result in parameters whose values are zero. Therefore, we need to apply **Laplace Smoothing**.
After applying it on our dataset, we get,

$$X = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \quad y = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

In this new dataset, parameters don't end up as zero probability estimates.

Estimating the GMM parameters for our dataset would result in parameters whose values are zero. Therefore, we need to apply **Laplace Smoothing**.

After applying it on our dataset, we get,

$$\mathrm{X} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \quad \mathrm{y} \ = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

In this new dataset, parameters don't end up as zero probability estimates.

Calculating $\widehat{p}$ using MLE for our dataset, we get,

$$\widehat{p} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

Estimating the GMM parameters for our dataset would result in parameters whose values are zero. Therefore, we need to apply **Laplace Smoothing**.
After applying it on our dataset, we get,

$$\mathrm{X} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \quad \mathrm{y} \;\; = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

In this new dataset, parameters don't end up as zero probability estimates.

Calculating $\widehat{p}$ using MLE for our dataset, we get,

$$\widehat{p} = \frac{1}{n}\sum_{i=1}^{n} y_i = \frac{1}{8}(1+1+0+0+1+0+1+0) = \frac{4}{8} = 0.5$$

Estimating the GMM parameters for our dataset would result in parameters whose values are zero. Therefore, we need to apply **Laplace Smoothing**.
After applying it on our dataset, we get,

$$\mathrm{X} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \quad \mathrm{y} \; = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

In this new dataset, parameters don't end up as zero probability estimates.
Calculating $\widehat{p}$ using MLE for our dataset, we get,

$$\widehat{p} = \frac{1}{n}\sum_{i=1}^{n} y_i = \frac{1}{8}(1+1+0+0+1+0+1+0) = \frac{4}{8} = 0.5$$

$$\therefore \widehat{p} = 0.5$$

Calculating $\widehat{p}_j^y$ using MLE for our dataset, we get,

$$\widehat{p}_j^y = \frac{\sum_{i=1}^{n} \mathbb{I}(f_j^i = 1, y_i = y)}{\sum_{i=1}^{n} \mathbb{I}(y_i = y)}$$

Calculating $\widehat{p}_j^y$ using MLE for our dataset, we get,

$$\widehat{p}_j^y = \frac{\sum_{i=1}^{n} \mathbb{I}(f_j^i = 1, y_i = y)}{\sum_{i=1}^{n} \mathbb{I}(y_i = y)}$$

$$\widehat{p}_1^1 = \frac{1+1+1}{4} = 0.75 \qquad \widehat{p}_1^0 = \frac{1}{4} = 0.25$$

Calculating $\widehat{p}_j^y$ using MLE for our dataset, we get,

$$\widehat{p}_j^y = \frac{\sum_{i=1}^{n} \mathbb{I}(f_j^i = 1, y_i = y)}{\sum_{i=1}^{n} \mathbb{I}(y_i = y)}$$

$$\widehat{p}_1^1 = \frac{1+1+1}{4} = 0.75 \qquad \widehat{p}_1^0 = \frac{1}{4} = 0.25$$

$$\widehat{p}_2^1 = \frac{1+1}{4} = 0.50 \qquad \widehat{p}_2^0 = \frac{1+1}{4} = 0.50$$

Calculating $\widehat{p}_j^y$ using MLE for our dataset, we get,

$$\widehat{p}_j^y = \frac{\sum_{i=1}^{n} \mathbb{I}(f_j^i = 1, y_i = y)}{\sum_{i=1}^{n} \mathbb{I}(y_i = y)}$$

$$\widehat{p}_1^1 = \frac{1+1+1}{4} = 0.75 \qquad \widehat{p}_1^0 = \frac{1}{4} = 0.25$$

$$\widehat{p}_2^1 = \frac{1+1}{4} = 0.50 \qquad \widehat{p}_2^0 = \frac{1+1}{4} = 0.50$$

$$\widehat{p}_3^1 = \frac{1}{4} = 0.25 \qquad \widehat{p}_3^0 = \frac{1+1}{4} = 0.50$$

## Prediction using Naive Bayes

If

$$\left(\prod_{i=1}^{d}\left(\widehat{p}_i^1\right)^{f_i}\left(1-\widehat{p}_i^1\right)^{1-f_i}\right)\widehat{p} \geq \left(\prod_{i=1}^{d}\left(\widehat{p}_i^0\right)^{f_i}\left(1-\widehat{p}_i^0\right)^{1-f_i}\right)(1-\widehat{p})$$

we predict $y = 1$, otherwise $y = 0$.

## Prediction using Naive Bayes

If

$$\left(\prod_{i=1}^{d}\left(\widehat{p}_i^{1}\right)^{f_i}\left(1-\widehat{p}_i^{1}\right)^{1-f_i}\right)\widehat{p} \geq \left(\prod_{i=1}^{d}\left(\widehat{p}_i^{0}\right)^{f_i}\left(1-\widehat{p}_i^{0}\right)^{1-f_i}\right)(1-\widehat{p})$$

we predict $y=1$, otherwise $y=0$.

Predicting $y_1$ for our dataset,

$$y_1 = \mathbb{I}\left(\left(\prod_{i=1}^{3}\left(\widehat{p}_i^{1}\right)^{f_i}\left(1-\widehat{p}_i^{1}\right)^{1-f_i}\right)\widehat{p} \geq \left(\prod_{i=1}^{3}\left(\widehat{p}_i^{0}\right)^{f_i}\left(1-\widehat{p}_i^{0}\right)^{1-f_i}\right)(1-\widehat{p})\right)$$

## Prediction using Naive Bayes

If

$$\left(\prod_{i=1}^{d}\left(\widehat{p}_i^{\,1}\right)^{f_i}\left(1-\widehat{p}_i^{\,1}\right)^{1-f_i}\right)\widehat{p} \geq \left(\prod_{i=1}^{d}\left(\widehat{p}_i^{\,0}\right)^{f_i}\left(1-\widehat{p}_i^{\,0}\right)^{1-f_i}\right)(1-\widehat{p})$$

we predict $y=1$, otherwise $y=0$.

Predicting $y_1$ for our dataset,

$$\begin{aligned} y_1 &= \mathbb{I}\left(\left(\prod_{i=1}^{3}\left(\widehat{p}_i^{\,1}\right)^{f_i}\left(1-\widehat{p}_i^{\,1}\right)^{1-f_i}\right)\widehat{p} \geq \left(\prod_{i=1}^{3}\left(\widehat{p}_i^{\,0}\right)^{f_i}\left(1-\widehat{p}_i^{\,0}\right)^{1-f_i}\right)(1-\widehat{p})\right) \\ &= \mathbb{I}\Big(\big((0.75)^1(0.75)^0(0.50)^0(0.50)^1(0.25)^0(0.75)^1\big)0.5 \geq \big((0.25)^1(0.75)^0(0.50)^0(0.50)^1(0.50)^0(0.50)^1\big)0.5\Big) \end{aligned}$$

## Prediction using Naive Bayes

If

$$\left(\prod_{i=1}^{d}\left(\widehat{p}_i^{1}\right)^{f_i}\left(1-\widehat{p}_i^{1}\right)^{1-f_i}\right)\widehat{p} \geq \left(\prod_{i=1}^{d}\left(\widehat{p}_i^{0}\right)^{f_i}\left(1-\widehat{p}_i^{0}\right)^{1-f_i}\right)(1-\widehat{p})$$

we predict $y=1$, otherwise $y=0$.

Predicting $y_1$ for our dataset,

$$\begin{aligned}
y_1 &= \mathbb{I}\left(\left(\prod_{i=1}^{3}\left(\widehat{p}_i^{1}\right)^{f_i}\left(1-\widehat{p}_i^{1}\right)^{1-f_i}\right)\widehat{p} \geq \left(\prod_{i=1}^{3}\left(\widehat{p}_i^{0}\right)^{f_i}\left(1-\widehat{p}_i^{0}\right)^{1-f_i}\right)(1-\widehat{p})\right) \\
&= \mathbb{I}\Big(\big((0.75)^1(0.75)^0(0.50)^0(0.50)^1(0.25)^0(0.75)^1\big)0.5 \geq \big((0.25)^1(0.75)^0(0.50)^0(0.50)^1(0.50)^0(0.50)^1\big)0.5\Big) \\
&= \mathbb{I}(0.141 \geq 0.031)
\end{aligned}$$

**Prediction using Naive Bayes**

If

$$\left(\prod_{i=1}^{d}\left(\widehat{p}_i^{1}\right)^{f_i}\left(1-\widehat{p}_i^{1}\right)^{1-f_i}\right)\widehat{p} \geq \left(\prod_{i=1}^{d}\left(\widehat{p}_i^{0}\right)^{f_i}\left(1-\widehat{p}_i^{0}\right)^{1-f_i}\right)(1-\widehat{p})$$

we predict $y=1$, otherwise $y=0$.

Predicting $y_1$ for our dataset,

$$\begin{aligned}
y_1 &= \mathbb{I}\left(\left(\prod_{i=1}^{3}\left(\widehat{p}_i^{1}\right)^{f_i}\left(1-\widehat{p}_i^{1}\right)^{1-f_i}\right)\widehat{p} \geq \left(\prod_{i=1}^{3}\left(\widehat{p}_i^{0}\right)^{f_i}\left(1-\widehat{p}_i^{0}\right)^{1-f_i}\right)(1-\widehat{p})\right) \\
&= \mathbb{I}\Big(\big((0.75)^1(0.75)^0(0.50)^0(0.50)^1(0.25)^0(0.75)^1\big)0.5 \geq \big((0.25)^1(0.75)^0(0.50)^0(0.50)^1(0.50)^0(0.50)^1\big)0.5\Big) \\
&= \mathbb{I}(0.141 \geq 0.031) \\
y_1 &= 1
\end{aligned}$$

Prediction table for all datapoints:

Prediction table for all datapoints:

| Label | $P(\widehat{y}=1\|\widehat{x})$ | $P(\widehat{y}=0\|\widehat{x})$ | Prediction | Actual |
|---|---|---|---|---|
| $y_1$ | $0.141$ | $0.031$ | 1 | 1 |
| $y_2$ | $0.141$ | $0.031$ | 1 | 1 |
| $y_3$ | $0.047$ | $0.094$ | 0 | 0 |
| $y_4$ | $0.016$ | $0.094$ | 0 | 0 |
| $y_5$ | $0.047$ | $0.031$ | 1 | 1 |
| $y_6$ | $0.047$ | $0.031$ | 1 | 0 |
| $y_7$ | $0.047$ | $0.094$ | 0 | 1 |
| $y_8$ | $0.047$ | $0.094$ | 0 | 0 |

# Gaussian Naive Bayes Algorithm

# Gaussian Naive Bayes Algorithm

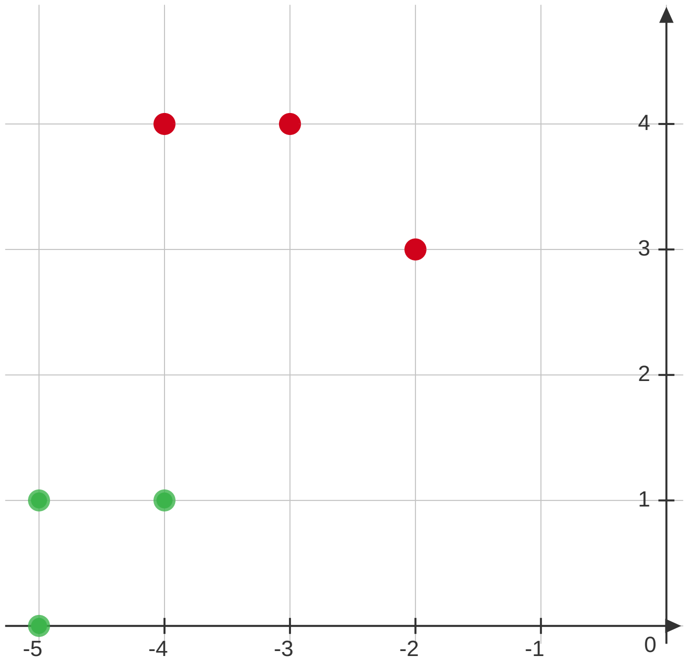
Given a dataset $\{\mathrm{x}_1, \ldots, \mathrm{x}_n\}$ where $\mathrm{x}_i \in \mathbb{R}^d$,
let $\{y_1, \ldots, y_n\}$ be the labels, where $y_i \in \{0, 1\}$.

$$\mathrm{X} = \begin{bmatrix} -3 & -4 & -2 & -4 & -5 & -5 \\ 4 & 4 & 3 & 1 & 1 & 0 \end{bmatrix} \quad \mathrm{y} \ = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

# Gaussian Naive Bayes Algorithm

Given a dataset $\{\mathrm{x}_1, \ldots, \mathrm{x}_n\}$ where $\mathrm{x}_i \in \mathbb{R}^d$, let $\{y_1, \ldots, y_n\}$ be the labels, where $y_i \in \{0, 1\}$.

$$\mathrm{X} = \begin{bmatrix} -3 & -4 & -2 & -4 & -5 & -5 \\ 4 & 4 & 3 & 1 & 1 & 0 \end{bmatrix} \quad \mathrm{y} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



Let $P(x|y=1) \sim \mathcal{N}(\mu_1, \Sigma_1)$ and $P(x|y=0) \sim \mathcal{N}(\mu_0, \Sigma_0)$.

The parameters to be estimated are $\widehat{p}$, $\widehat{\mu}_0$, $\widehat{\mu}_1$, and $\widehat{\Sigma}_0$ and $\widehat{\Sigma}_1$.

Let $P(x|y=1) \sim \mathcal{N}(\mu_1, \Sigma_1)$ and $P(x|y=0) \sim \mathcal{N}(\mu_0, \Sigma_0)$.

The parameters to be estimated are $\widehat{p}$, $\widehat{\mu}_0$, $\widehat{\mu}_1$, and $\widehat{\Sigma}_0$ and $\widehat{\Sigma}_1$.

Using Maximum Likelihood Estimation, we get the following results:

Let $P(x|y=1) \sim \mathcal{N}(\mu_1, \Sigma_1)$ and $P(x|y=0) \sim \mathcal{N}(\mu_0, \Sigma_0)$.

The parameters to be estimated are $\widehat{p}$, $\widehat{\mu}_0$, $\widehat{\mu}_1$, and $\widehat{\Sigma}_0$ and $\widehat{\Sigma}_1$.

Using Maximum Likelihood Estimation, we get the following results:

$$\widehat{p} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

Let $P(x|y=1) \sim \mathcal{N}(\mu_1, \Sigma_1)$ and $P(x|y=0) \sim \mathcal{N}(\mu_0, \Sigma_0)$.

The parameters to be estimated are $\widehat{p}$, $\widehat{\mu}_0$, $\widehat{\mu}_1$, and $\widehat{\Sigma}_0$ and $\widehat{\Sigma}_1$.

Using Maximum Likelihood Estimation, we get the following results:

$$\widehat{p} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

$$\widehat{\mu}_1 = \frac{\displaystyle\sum_{i=1}^{n} \mathbb{I}(y_i = 1)x_i}{\displaystyle\sum_{i=1}^{n} \mathbb{I}(y_i = 1)}$$

$$\widehat{\mu}_0 = \frac{\displaystyle\sum_{i=1}^{n} \mathbb{I}(y_i = 0)x_i}{\displaystyle\sum_{i=1}^{n} \mathbb{I}(y_i = 0)}$$

$$\widehat{\Sigma}_k = \frac{\sum_{i=1}^{n} \mathbb{I}(y_i = k)\Big((x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^T\Big)}{\sum_{i=1}^{n} \mathbb{I}(y_i = k)}$$

Where $\widehat{p}$ is the proportion of data points labeled $1$, $\widehat{\mu}_1$ is the sample mean of data points labeled $1$, $\widehat{\mu}_0$ is the sample mean of data points labeled $0$, and $\widehat{\Sigma}_1$ and $\widehat{\Sigma}_0$ are the covariance matrices for classes $1$ and $0$ respectively.

$$\widehat{\Sigma}_k = \frac{\sum_{i=1}^{n} \mathbb{I}(y_i = k)\Big((x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^T\Big)}{\sum_{i=1}^{n} \mathbb{I}(y_i = k)}$$

Where $\widehat{p}$ is the proportion of data points labeled $1$, $\widehat{\mu}_1$ is the sample mean of data points labeled $1$, $\widehat{\mu}_0$ is the sample mean of data points labeled $0$, and $\widehat{\Sigma}_1$ and $\widehat{\Sigma}_0$ are the covariance matrices for classes $1$ and $0$ respectively.

Calculating $\widehat{p}$ using MLE for our dataset, we get,

$$\widehat{p} = \frac{1}{n}\sum_{i=1}^{n} y_i = \frac{1}{6}(1+1+1+0+0+0) = \frac{3}{6} = 0.5$$

$$\therefore \widehat{p} = 0.5$$

Calculating $\widehat{\mu}_k$ using MLE for our dataset, we get,

$$\widehat{\mu}_k = \frac{\displaystyle\sum_{i=1}^{n} \mathbb{I}(y_i = k) x_i}{\displaystyle\sum_{i=1}^{n} \mathbb{I}(y_i = k)}$$

Calculating $\widehat{\mu}_k$ using MLE for our dataset, we get,

$$\widehat{\mu}_k = \frac{\displaystyle\sum_{i=1}^{n} \mathbb{I}(y_i = k)x_i}{\displaystyle\sum_{i=1}^{n} \mathbb{I}(y_i = k)}$$

$$\widehat{\mu}_1 = \frac{\begin{bmatrix} -3 \\ 4 \end{bmatrix} + \begin{bmatrix} -4 \\ 4 \end{bmatrix} + \begin{bmatrix} -2 \\ 3 \end{bmatrix}}{3} = \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}$$

$$\widehat{\mu}_0 = \frac{\begin{bmatrix} -4 \\ 1 \end{bmatrix} + \begin{bmatrix} -5 \\ 1 \end{bmatrix} + \begin{bmatrix} -5 \\ 0 \end{bmatrix}}{3} = \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}$$

Calculating $\widehat{\Sigma}_k$ using MLE for our dataset, we get,

$$\widehat{\Sigma}_k = \frac{\sum_{i=1}^{n} \mathbb{I}(y_i = k)\Big((x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^T\Big)}{\sum_{i=1}^{n} \mathbb{I}(y_i = k)}$$

Calculating $\widehat{\Sigma}_k$ using MLE for our dataset, we get,

$$\widehat{\Sigma}_k = \frac{\sum_{i=1}^{n} \mathbb{I}(y_i = k)\Big((x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^T\Big)}{\sum_{i=1}^{n} \mathbb{I}(y_i = k)}$$

$$\widehat{\Sigma}_1 = \frac{\left(\begin{bmatrix}-3\\4\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)\left(\begin{bmatrix}-3\\4\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)^T + \left(\begin{bmatrix}-4\\4\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)\left(\begin{bmatrix}-4\\4\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)^T + \left(\begin{bmatrix}-2\\3\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)\left(\begin{bmatrix}-2\\3\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)^T}{3}$$

Calculating $\widehat{\Sigma}_k$ using MLE for our dataset, we get,

$$\widehat{\Sigma}_k = \frac{\sum_{i=1}^{n} \mathbb{I}(y_i = k)\Big((x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^T\Big)}{\sum_{i=1}^{n} \mathbb{I}(y_i = k)}$$

$$\widehat{\Sigma}_1 = \frac{\left(\begin{bmatrix} -3 \\ 4 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)\left(\begin{bmatrix} -3 \\ 4 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)^T + \left(\begin{bmatrix} -4 \\ 4 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)\left(\begin{bmatrix} -4 \\ 4 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)^T + \left(\begin{bmatrix} -2 \\ 3 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)\left(\begin{bmatrix} -2 \\ 3 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)^T}{3}$$

$$\widehat{\Sigma}_1 = \begin{bmatrix} 0.666 & 0.333 \\ -0.333 & 0.222 \end{bmatrix}$$

Calculating $\widehat{\Sigma}_k$ using MLE for our dataset, we get,

$$\widehat{\Sigma}_k = \frac{\sum_{i=1}^{n} \mathbb{I}(y_i = k)\Big((x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^T\Big)}{\sum_{i=1}^{n} \mathbb{I}(y_i = k)}$$

$$\widehat{\Sigma}_1 = \frac{\left(\begin{bmatrix} -3 \\ 4 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)\left(\begin{bmatrix} -3 \\ 4 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)^T + \left(\begin{bmatrix} -4 \\ 4 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)\left(\begin{bmatrix} -4 \\ 4 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)^T + \left(\begin{bmatrix} -2 \\ 3 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)\left(\begin{bmatrix} -2 \\ 3 \end{bmatrix} - \begin{bmatrix} -3 \\ 3.666 \end{bmatrix}\right)^T}{3}$$

$$\widehat{\Sigma}_1 = \begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}$$

$$\widehat{\Sigma}_0 = \frac{\left(\begin{bmatrix} -4 \\ 1 \end{bmatrix} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}\right)\left(\begin{bmatrix} -4 \\ 1 \end{bmatrix} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}\right)^T + \left(\begin{bmatrix} -5 \\ 1 \end{bmatrix} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}\right)\left(\begin{bmatrix} -5 \\ 1 \end{bmatrix} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}\right)^T + \left(\begin{bmatrix} -5 \\ 0 \end{bmatrix} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}\right)\left(\begin{bmatrix} -5 \\ 0 \end{bmatrix} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}\right)^T}{3}$$

Calculating $\widehat{\Sigma}_k$ using MLE for our dataset, we get,

$$\widehat{\Sigma}_k = \frac{\sum_{i=1}^{n} \mathbb{I}(y_i = k)\Big((x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^T\Big)}{\sum_{i=1}^{n} \mathbb{I}(y_i = k)}$$

$$\widehat{\Sigma}_1 = \frac{\left(\begin{bmatrix}-3\\4\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)\left(\begin{bmatrix}-3\\4\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)^T + \left(\begin{bmatrix}-4\\4\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)\left(\begin{bmatrix}-4\\4\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)^T + \left(\begin{bmatrix}-2\\3\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)\left(\begin{bmatrix}-2\\3\end{bmatrix} - \begin{bmatrix}-3\\3.666\end{bmatrix}\right)^T}{3}$$

$$\widehat{\Sigma}_1 = \begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}$$

$$\widehat{\Sigma}_0 = \frac{\left(\begin{bmatrix}-4\\1\end{bmatrix} - \begin{bmatrix}-4.66\\0.666\end{bmatrix}\right)\left(\begin{bmatrix}-4\\1\end{bmatrix} - \begin{bmatrix}-4.66\\0.666\end{bmatrix}\right)^T + \left(\begin{bmatrix}-5\\1\end{bmatrix} - \begin{bmatrix}-4.66\\0.666\end{bmatrix}\right)\left(\begin{bmatrix}-5\\1\end{bmatrix} - \begin{bmatrix}-4.66\\0.666\end{bmatrix}\right)^T + \left(\begin{bmatrix}-5\\0\end{bmatrix} - \begin{bmatrix}-4.66\\0.666\end{bmatrix}\right)\left(\begin{bmatrix}-5\\0\end{bmatrix} - \begin{bmatrix}-4.66\\0.666\end{bmatrix}\right)^T}{3}$$

$$\widehat{\Sigma}_0 = \begin{bmatrix} 0.222 & 0.111 \\ 0.111 & 0.222 \end{bmatrix}$$

# Prediction using Gaussian Naive Bayes

# Prediction using Gaussian Naive Bayes

Predict $y_i = 1$ if:

$$f(x_i; \widehat{\mu}_1, \widehat{\Sigma}_1)\widehat{p} \geq f(x_i; \widehat{\mu}_0, \widehat{\Sigma}_0)(1 - \widehat{p})$$

**Prediction using Gaussian Naive Bayes**

Predict $y_i = 1$ if:

$$f(x_i; \widehat{\mu}_1, \widehat{\Sigma}_1)\widehat{p} \geq f(x_i; \widehat{\mu}_0, \widehat{\Sigma}_0)(1 - \widehat{p})$$

$$e^{-(x_i - \widehat{\mu}_1)^T \widehat{\Sigma}_1^{-1} (x_i - \widehat{\mu}_1)} \widehat{p} \geq e^{-(x_i - \widehat{\mu}_0)^T \widehat{\Sigma}_0^{-1} (x_i - \widehat{\mu}_0)} (1 - \widehat{p})$$

## Prediction using Gaussian Naive Bayes

Predict $y_i = 1$ if:

$$f(x_i; \widehat{\mu}_1, \widehat{\Sigma}_1)\widehat{p} \geq f(x_i; \widehat{\mu}_0, \widehat{\Sigma}_0)(1 - \widehat{p})$$

$$e^{-(x_i - \widehat{\mu}_1)^T \widehat{\Sigma}_1^{-1} (x_i - \widehat{\mu}_1)} \widehat{p} \geq e^{-(x_i - \widehat{\mu}_0)^T \widehat{\Sigma}_0^{-1} (x_i - \widehat{\mu}_0)} (1 - \widehat{p})$$

$$-(x_i - \widehat{\mu}_1)^T \widehat{\Sigma}_1^{-1} (x_i - \widehat{\mu}_1) + \log(\widehat{p}) \geq -(x_i - \widehat{\mu}_0)^T \widehat{\Sigma}_0^{-1} (x_i - \widehat{\mu}_0) + \log(1 - \widehat{p})$$

## Prediction using Gaussian Naive Bayes

Predict $y_i = 1$ if:

$$f(x_i; \widehat{\mu}_1, \widehat{\Sigma}_1)\widehat{p} \geq f(x_i; \widehat{\mu}_0, \widehat{\Sigma}_0)(1-\widehat{p})$$

$$e^{-(x_i-\widehat{\mu}_1)^T\widehat{\Sigma}_1^{-1}(x_i-\widehat{\mu}_1)}\widehat{p} \geq e^{-(x_i-\widehat{\mu}_0)^T\widehat{\Sigma}_0^{-1}(x_i-\widehat{\mu}_0)}(1-\widehat{p})$$

$$-(x_i-\widehat{\mu}_1)^T\widehat{\Sigma}_1^{-1}(x_i-\widehat{\mu}_1) + \log(\widehat{p}) \geq -(x_i-\widehat{\mu}_0)^T\widehat{\Sigma}_0^{-1}(x_i-\widehat{\mu}_0) + \log(1-\widehat{p})$$

$$x_i^T\left(\widehat{\Sigma}_1^{-1} - \widehat{\Sigma}_0^{-1}\right)x_i - 2\left(\widehat{\mu}_1^T\widehat{\Sigma}_1^{-1} - \widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\right)x_i + \left(\widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\widehat{\mu}_0 - \widehat{\mu}_1^T\widehat{\Sigma}_1^{-1}\widehat{\mu}_1\right) + log\left(\frac{1-\widehat{p}}{\widehat{p}}\right) \geq 0$$

# Prediction using Gaussian Naive Bayes

Predict $y_i = 1$ if:

$$f(x_i; \widehat{\mu}_1, \widehat{\Sigma}_1)\widehat{p} \geq f(x_i; \widehat{\mu}_0, \widehat{\Sigma}_0)(1-\widehat{p})$$

$$e^{-(x_i-\widehat{\mu}_1)^T \widehat{\Sigma}_1^{-1} (x_i-\widehat{\mu}_1)}\widehat{p} \geq e^{-(x_i-\widehat{\mu}_0)^T \widehat{\Sigma}_0^{-1} (x_i-\widehat{\mu}_0)}(1-\widehat{p})$$

$$-(x_i-\widehat{\mu}_1)^T \widehat{\Sigma}_1^{-1} (x_i-\widehat{\mu}_1) + \log(\widehat{p}) \geq -(x_i-\widehat{\mu}_0)^T \widehat{\Sigma}_0^{-1} (x_i-\widehat{\mu}_0) + \log(1-\widehat{p})$$

$$x_i^T\left(\widehat{\Sigma}_1^{-1} - \widehat{\Sigma}_0^{-1}\right)x_i - 2\left(\widehat{\mu}_1^T\widehat{\Sigma}_1^{-1} - \widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\right)x_i + \left(\widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\widehat{\mu}_0 - \widehat{\mu}_1^T\widehat{\Sigma}_1^{-1}\widehat{\mu}_1\right) + log\left(\frac{1-\widehat{p}}{\widehat{p}}\right) \geq 0$$

Hence, we can say that the decision function is of the form $x^TQx - 2b^Tx + c \geq 0$ where $Q = \widehat{\Sigma}_1^{-1} - \widehat{\Sigma}_0^{-1}$, $b = \widehat{\mu}_1^T\widehat{\Sigma}_1^{-1} - \widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}$, and $c = (\widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\widehat{\mu}_0 - \widehat{\mu}_1^T\widehat{\Sigma}_1^{-1}\widehat{\mu}_1) + log(\frac{1-\widehat{p}}{\widehat{p}})$.

Predicting $y_1$ for our dataset,

$$y_1 = \mathbb{I}\left(x_1^T\left(\widehat{\Sigma}_1^{-1} - \widehat{\Sigma}_0^{-1}\right)x_1 - 2\left(\widehat{\mu}_1^T\widehat{\Sigma}_1^{-1} - \widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\right)x_1 + \left(\widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\widehat{\mu}_0 - \widehat{\mu}_1^T\widehat{\Sigma}_1^{-1}\widehat{\mu}_1\right) + log\left(\frac{1-\widehat{p}}{\widehat{p}}\right) \geq 0\right)$$

Predicting $y_1$ for our dataset,

$$y_1 = \mathbb{I}\left(x_1^T\left(\widehat{\Sigma}_1^{-1} - \widehat{\Sigma}_0^{-1}\right)x_1 - 2\left(\widehat{\mu}_1^T\widehat{\Sigma}_1^{-1} - \widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\right)x_1 + \left(\widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\widehat{\mu}_0 - \widehat{\mu}_1^T\widehat{\Sigma}_1^{-1}\widehat{\mu}_1\right) + log\left(\frac{1-\widehat{p}}{\widehat{p}}\right) \geq 0\right)$$

$$= \mathbb{I}(\begin{bmatrix} -3 \\ 4 \end{bmatrix}^T \left(\begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}^{-1} - \begin{bmatrix} 0.222 & 0.111 \\ 0.111 & 0.222 \end{bmatrix}^{-1}\right)\begin{bmatrix} -3 \\ 4 \end{bmatrix} - 2\left(\begin{bmatrix} -3 \\ 3.666 \end{bmatrix}^T \begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}^{-1} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}^T \begin{bmatrix} 0.222 & 0.111 \\ 0.111 & 0.222 \end{bmatrix}^{-1}\right)\begin{bmatrix} -3 \\ 4 \end{bmatrix}$$

$$+\left(\begin{bmatrix} -3 \\ 3.666 \end{bmatrix}^T \begin{bmatrix} 0.222 & 0.111 \\ 0.111 & 0.222 \end{bmatrix}^{-1} \begin{bmatrix} -3 \\ 3.666 \end{bmatrix} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}^T \begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}^{-1} \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}\right) + log\left(\frac{1-0.5}{0.5}\right) \geq 0)$$

Predicting $y_1$ for our dataset,

$$y_1 = \mathbb{I}\left(x_1^T\left(\widehat{\Sigma}_1^{-1} - \widehat{\Sigma}_0^{-1}\right)x_1 - 2\left(\widehat{\mu}_1^T\widehat{\Sigma}_1^{-1} - \widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\right)x_1 + \left(\widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\widehat{\mu}_0 - \widehat{\mu}_1^T\widehat{\Sigma}_1^{-1}\widehat{\mu}_1\right) + log\left(\frac{1-\widehat{p}}{\widehat{p}}\right) \geq 0\right)$$

$$= \mathbb{I}(\begin{bmatrix} -3 \\ 4 \end{bmatrix}^T\left(\begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}^{-1} - \begin{bmatrix} 0.222 & 0.111 \\ 0.111 & 0.222 \end{bmatrix}^{-1}\right)\begin{bmatrix} -3 \\ 4 \end{bmatrix} - 2\left(\begin{bmatrix} -3 \\ 3.666 \end{bmatrix}^T\begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}^{-1} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}^T\begin{bmatrix} 0.222 & 0.111 \\ 0.111 & 0.222 \end{bmatrix}^{-1}\right)\begin{bmatrix} -3 \\ 4 \end{bmatrix}$$

$$+\left(\begin{bmatrix} -3 \\ 3.666 \end{bmatrix}^T\begin{bmatrix} 0.222 & 0.111 \\ 0.111 & 0.222 \end{bmatrix}^{-1}\begin{bmatrix} -3 \\ 3.666 \end{bmatrix} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}^T\begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}^{-1}\begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}\right) + log\left(\frac{1-0.5}{0.5}\right) \geq 0)$$

$$= \mathbb{I}(1.0 \geq 0)$$

Predicting $y_1$ for our dataset,

$$y_1 = \mathbb{I}\left(x_1^T\left(\widehat{\Sigma}_1^{-1} - \widehat{\Sigma}_0^{-1}\right)x_1 - 2\left(\widehat{\mu}_1^T\widehat{\Sigma}_1^{-1} - \widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\right)x_1 + \left(\widehat{\mu}_0^T\widehat{\Sigma}_0^{-1}\widehat{\mu}_0 - \widehat{\mu}_1^T\widehat{\Sigma}_1^{-1}\widehat{\mu}_1\right) + log\left(\frac{1-\widehat{p}}{\widehat{p}}\right) \geq 0\right)$$

$$= \mathbb{I}(\begin{bmatrix} -3 \\ 4 \end{bmatrix}^T \left(\begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}^{-1} - \begin{bmatrix} 0.222 & 0.111 \\ 0.111 & 0.222 \end{bmatrix}^{-1}\right)\begin{bmatrix} -3 \\ 4 \end{bmatrix} - 2\left(\begin{bmatrix} -3 \\ 3.666 \end{bmatrix}^T \begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}^{-1} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}^T \begin{bmatrix} 0.222 & 0.111 \\ 0.111 & 0.222 \end{bmatrix}^{-1}\right)\begin{bmatrix} -3 \\ 4 \end{bmatrix}$$

$$+ \left(\begin{bmatrix} -3 \\ 3.666 \end{bmatrix}^T \begin{bmatrix} 0.222 & 0.111 \\ 0.111 & 0.222 \end{bmatrix}^{-1} \begin{bmatrix} -3 \\ 3.666 \end{bmatrix} - \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}^T \begin{bmatrix} 0.666 & -0.333 \\ -0.333 & 0.222 \end{bmatrix}^{-1} \begin{bmatrix} -4.66 \\ 0.666 \end{bmatrix}\right) + log\left(\frac{1-0.5}{0.5}\right) \geq 0)$$

$$= \mathbb{I}(1.0 \geq 0)$$

$$y_1 = 1$$

Prediction table for all datapoints:

Prediction table for all datapoints:

| Label | $P(\widehat{y}=1\|\widehat{x})$ | $P(\widehat{y}=0\|\widehat{x})$ | Prediction | Actual |
|---|---|---|---|---|
| $y_1$ | $1.0$ | $0$ | 1 | 1 |
| $y_2$ | $1.0$ | $0$ | 1 | 1 |
| $y_3$ | $1.0$ | $0$ | 1 | 1 |
| $y_4$ | $0$ | $1.0$ | 0 | 0 |
| $y_5$ | $0$ | $1.0$ | 0 | 0 |
| $y_6$ | $0.00019$ | $0.99981$ | 0 | 0 |

# Gaussian Naive Bayes Algorithm