Linear Discriminant Analysis (2)

Son Nguyen

Classification Problem

- Given a dataset that has x and y (class or label)

x_1, x_2

x	y
1.49	1
1.23	1
0.95	1
0.89	1
2.58	2
2.65	2
2.27	2
1.94	2
1.39	2
1.44	2

- Given a new value x, what class the it belongs to? (what is the predicted y value)
- If $x = 1.4$, what is its associated y value? (What class it belongs to?)

Approach

- We will estimate two probabilities $p_1 = P(y = 1|x = 1.4)$ and $p_2 = P(y = 2|x = 1.4)$.
- If $p_1 > p_2$, we will classify the new point to class 1 and vice versa.

Approach

We have, using the Bayes' Rule,

$$p_1 = P(y = 1|x = 1.4) = \frac{P(y = 1) * L(x = 1.4|y = 1)}{L(x = 1.4)}$$

where L(A) denotes the likelihood of the event A. Similarly,

$$p_2 = P(y = 2|x = 1.4) = \frac{P(y = 2) * L(x = 1.4|y = 1)}{L(x = 1.4)}$$

Since the denominator is the same we just need to compare the numerator.

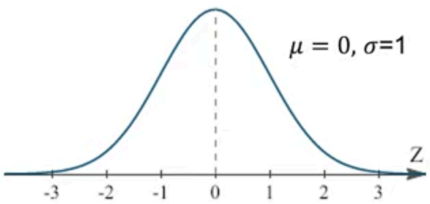
LDA Assumptions

- x is normally distributed in each class
- Assume that x has the same variance in both classes

Likelihood:

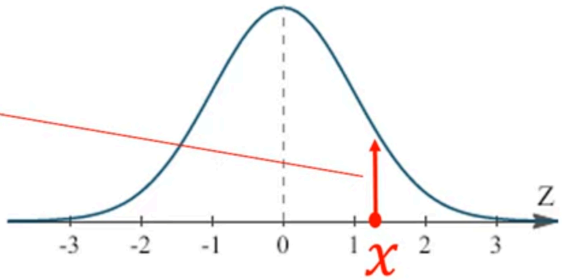
The *probability density function* for a normal distribution $N(\mu, \sigma^2)$ is:

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



For a given distribution the *likelihood* of the distribution parameters being μ, σ^2 given the observation x is:

$$L(\mu, \sigma^2|x) = f(x|\mu, \sigma^2)$$



Calculation

Calculation

https://planetcalc.com/4986/

https://www.standarddeviationcalculator.io/normal-distribution-calculator

Decision Boundary

- The decision boundary is where $p_1 = p_2$

- Thus,

$$\frac{P(y=1) * L(x=x_0|y=1)}{L(x=x_0)} = \frac{P(y=2) * L(x=x_0|y=1)}{L(x=x_0)}$$

$$\implies P(y=1) * L(x=x_0|y=1) = P(y=2) * L(x=x_0|y=1)$$

$$\implies \pi_1 * L(x=x_0|y=1) = \pi_2 * L(x=x_0|y=1)$$

$$\implies \pi_1 * f(x_0|\mu_1, \sigma^2) = \pi_2 * f(x_0|\mu_2, \sigma^2)$$

$$\implies \pi_1 * \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-(x_0-\mu_1)^2/(2\sigma^2)} = \pi_2 * \frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-(x_0-\mu_2)^2/(2\sigma^2)}$$

$$\implies \pi_1 * \frac{1}{\cancel{\sigma_1}} e^{-(x_0-\mu_1)^2/\cancel{(2\sigma_1^2)}} = \pi_2 * \frac{1}{\cancel{\sigma_2}} e^{-(x_0-\mu_2)^2/\cancel{(2\sigma_2^2)}}$$

In LDA $\boxed{\sigma_1 = \sigma_2}$

$$\Rightarrow \pi_1 \cdot e^{-(x_0-\mu_1)^2} = \pi_2 \, e^{-(x_0-\mu_2)^2}$$

LDA

- In LDA, we assume that the two groups have the same variance, or $\sigma_1 = \sigma_2$. Thus, the decision boundary becomes

$$2x_0 = \mu_1 + \mu_2 + \frac{2\sigma^2(\ln \pi_2 - \ln \pi_1)}{\mu_1 - \mu_2}$$

- We notice, this is a linear equation on x_0.

QDA

- In Quadratic Discriminant Analysis (QDA), we still assume the predictors are normally distributed.
- But in QDA, we do not assume the variance of the predictors in the two groups are the same. Thus $\sigma_1 \neq \sigma_2$.
- The decision boundary becomes a $\underbrace{\text{quadratic equation of } x_0}$.

Decision Boundary

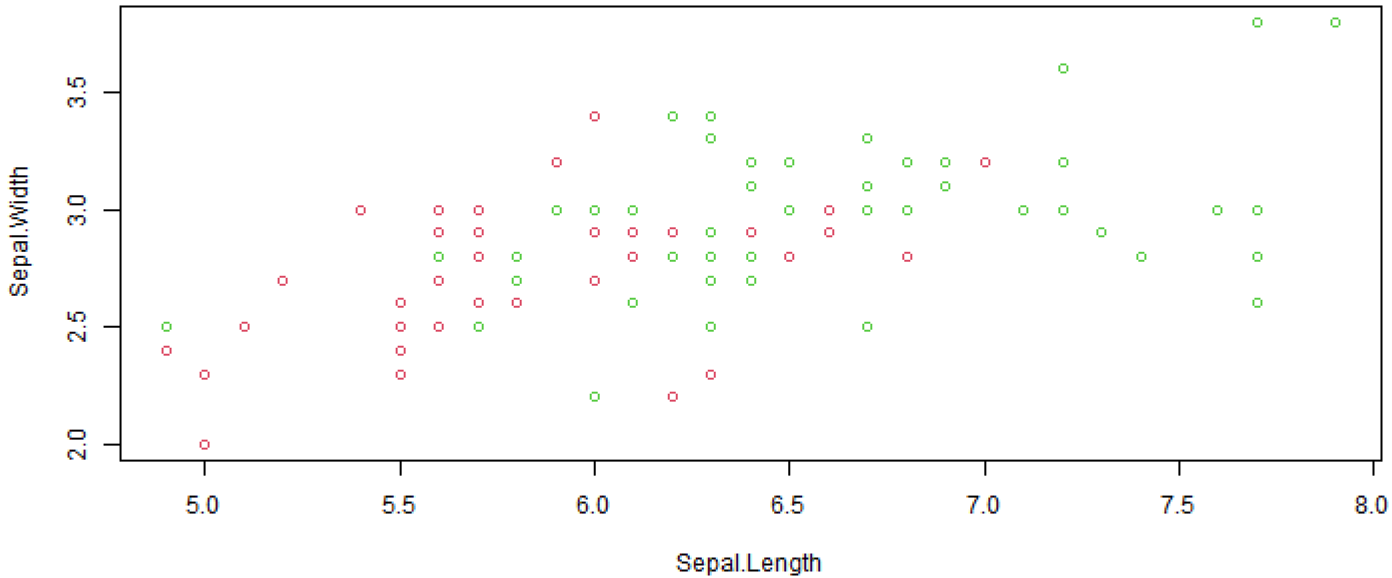
- Iris Dataset: This famous (Fisher's or Anderson's) iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 2 species of iris. The species are Iris versicolor and virginica.

Y

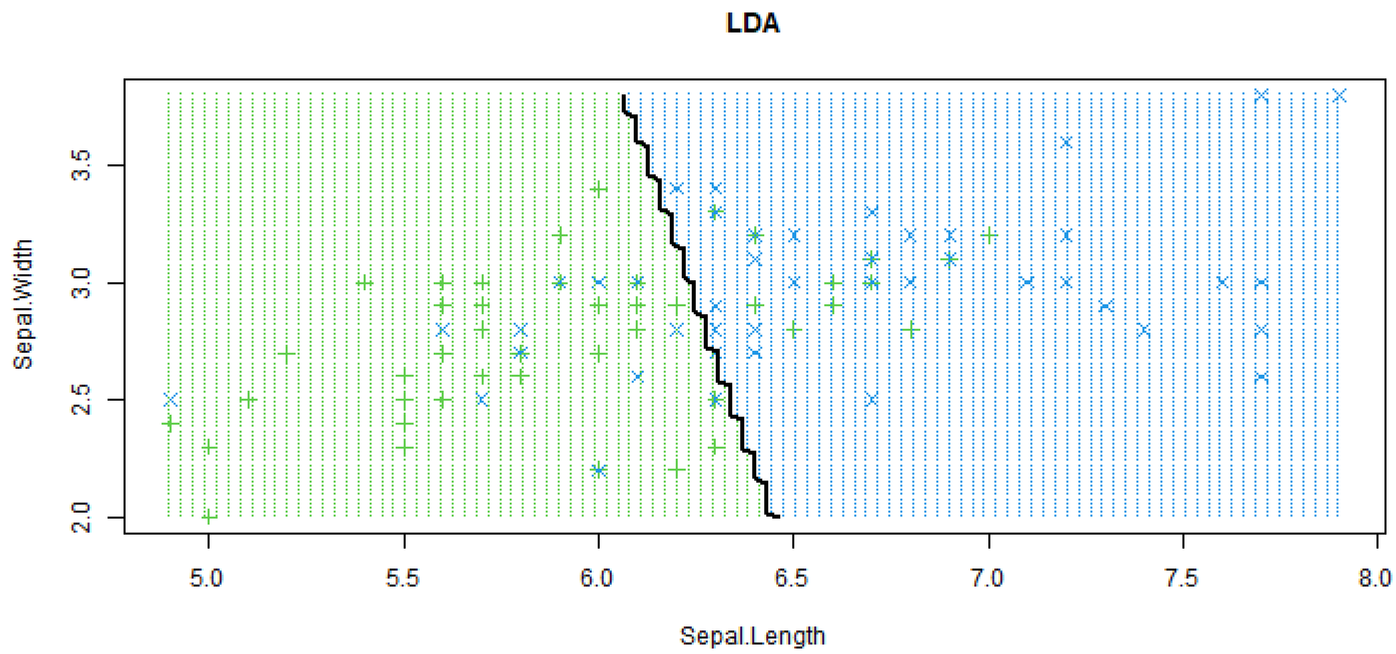
	Sepal.Length	Sepal.Width	Species
66	6.4	3.2	virginica
37	6.7	3.1	versicolor
45	5.6	2.7	versicolor
60	7.2	3.6	virginica
17	5.6	3.0	versicolor
32	5.5	2.4	versicolor
29	6.0	2.9	versicolor
11	5.0	2.0	versicolor
16	6.7	3.1	versicolor
94	6.8	3.2	virginica

X1 X2

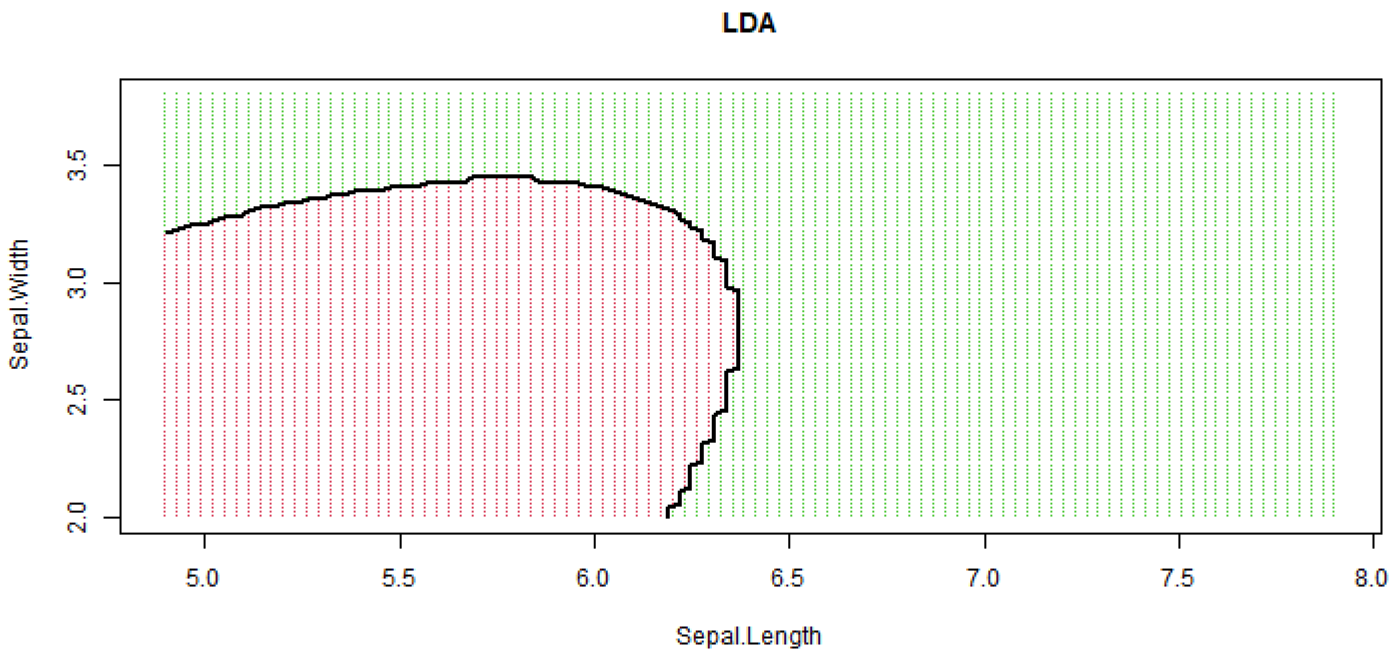
Decision Boundary

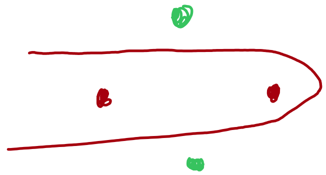


Decision Boundary



Decision Boundary





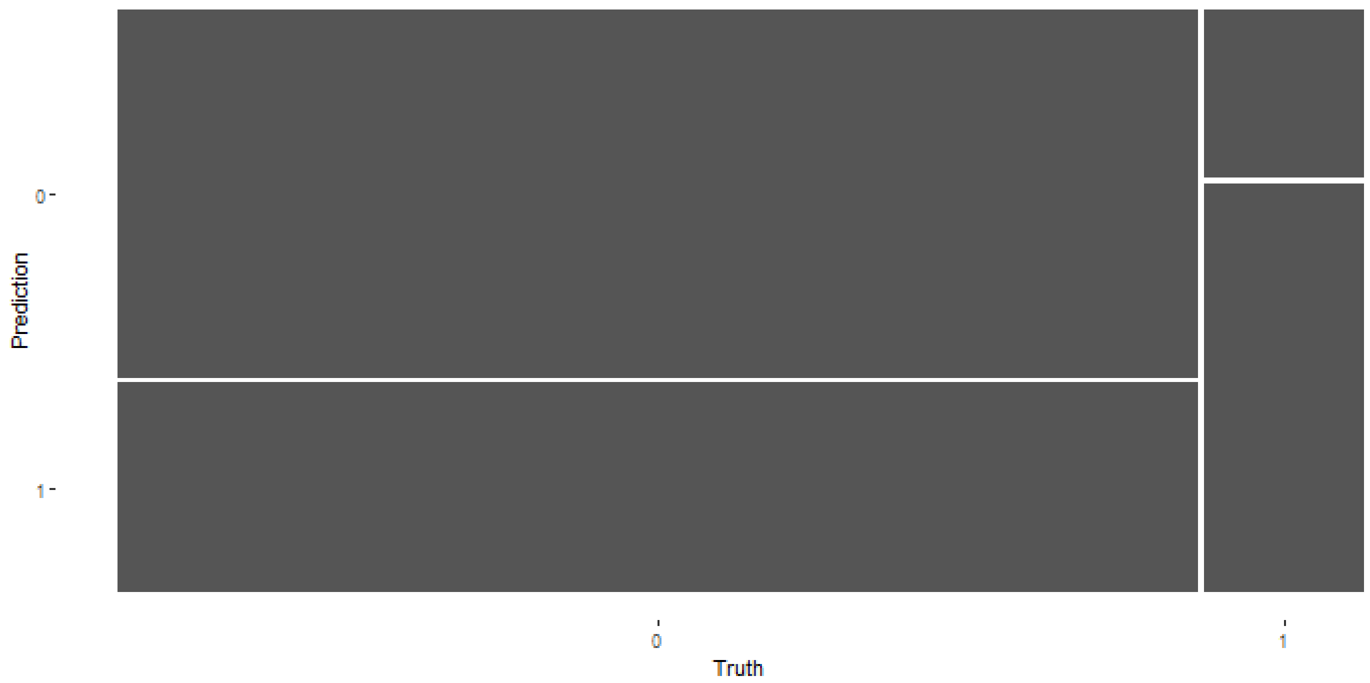
LDA on Titanic Dataset

Titanic Dataset

```
library(caret)
library(tidyverse)

# read the data
df = read_csv("titanic2.csv")

# create the target variable
df= df %>%  rename(target = Survived)

# train LDA model
model = lda(target ~ Age + Fare,
            data = df)

# make predictions
pred = predict(model, df,
        type = 'response')$class

# calculate accuracy
cm <- confusionMatrix(data = pred, reference = factor(df$target))
cm$overall[1]
```

```
##  Accuracy
## 0.6470588
```

```r
d = data.frame(pred = pred, obs = factor(df$target))
library(yardstick)
d %>% conf_mat(pred, obs) %>% autoplot
```



QDA on Titanic Dataset

```
model = qda(target ~ Age + Fare,
            data = df)

pred = predict(model, df,
        type = 'response')$class

cm <- confusionMatrix(data = pred, reference = factor(df$target))
cm$overall[1]
```

```
##  Accuracy
## 0.6470588
```

```r
d = data.frame(pred = pred, obs = factor(df$target))
library(yardstick)
d %>% conf_mat(pred, obs) %>% autoplot
```