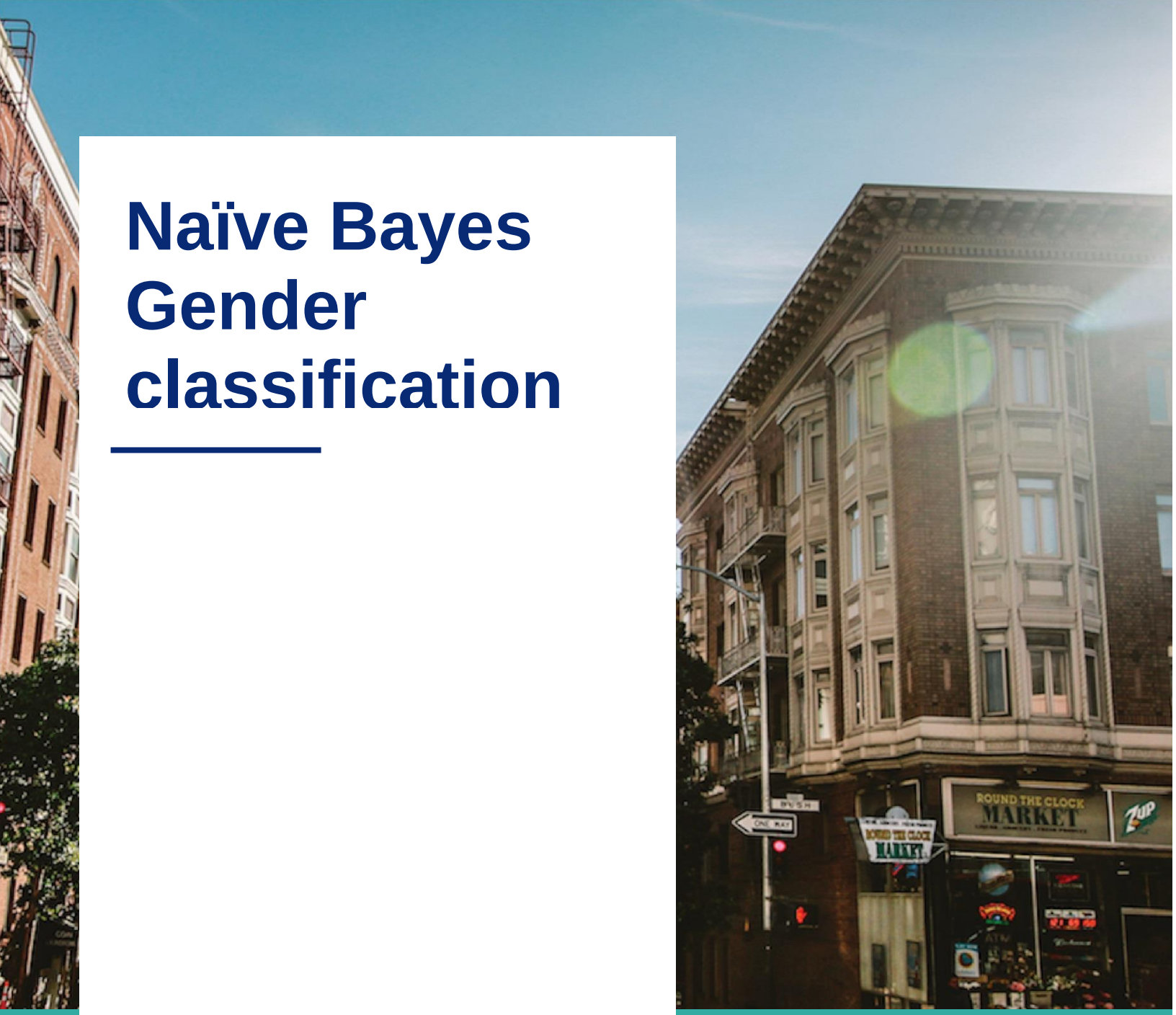


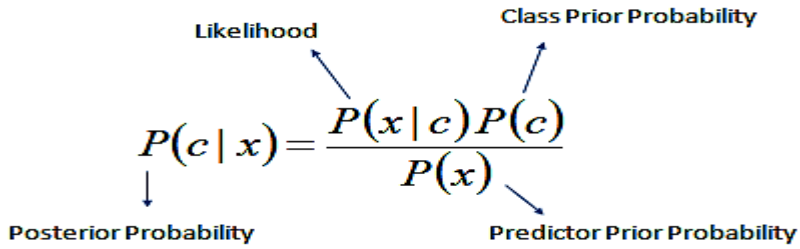
Naïve Bayes Gender classification

MARCH 19

Authored by: Mohamed Elrasheed
00249123622963

Naïve Bayes algorithm using R

The ***Naïve Bayes classifier*** is a simple probabilistic classifier which is based on Bayes theorem but with strong assumptions regarding independence.



$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Problem :

We want to build gender classification based on attributes[Height,weight,footsize] model naïve Bayes using R

The idea

In the context of our data, we are seeking the probability of a person belonging to male or female class Ck (where C1=male and C2=female) given that its predictor values are Height, Weight, Footsize. This can be written as P (Ck| Height, Weight, and Footsize).

The Bayesian formula for calculating this probability is

$$P(C_k|X) = \frac{P(C_k) \cdot P(X|C_k)}{P(X)} \qquad (1)$$

Where:

- P(Ck)

Is the *prior* probability of the outcome. Essentially, based on the historical data, what is the probability of a person are male or female. Our prior probability of a person are male was about 50% and the probability of person are female was about 50%.

- P(X)
  is the probability of the predictor variables (same as P(Ck| Height, Weight, and Footsize). Essentially, based on the historical data, what is the probability of each observed combination of predictor variables. When new data comes in, this becomes our *evidence*.

- $P(X|C_k)$

is the conditional probability or likelihood. Essentially, for each class of the response variable (i.e. male or female), what is the probability of observing the predictor values.

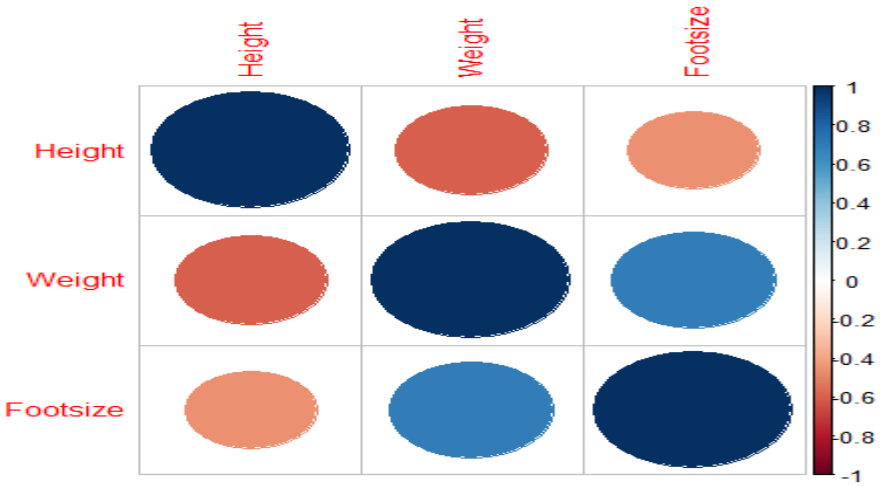
- $P(C_k|X)$

is called our posterior probability. By combining our observed information, we are updating our a priori information on probabilities to compute a posterior probability that an observation has class Ck.

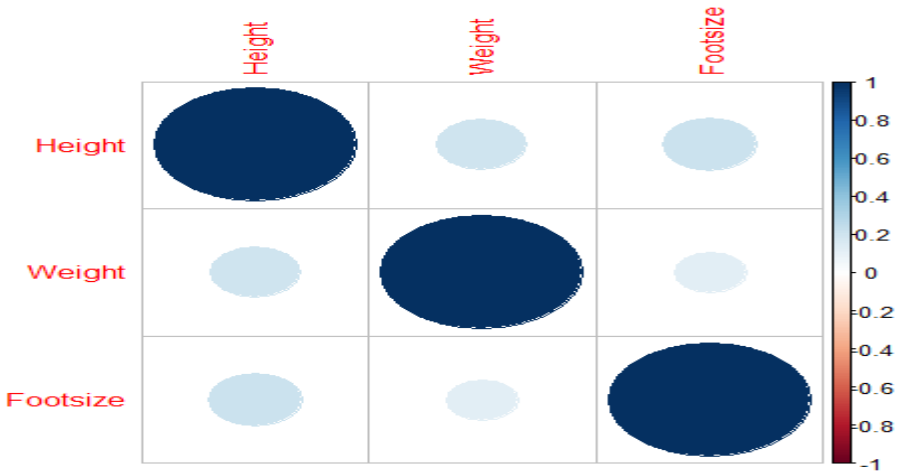
The simplified classifier

Consequently, the naïve Bayes classifier makes a simplifying assumption (hence the name) to allow the computation to scale. With naïve Bayes, we assume that the predictor variables are conditionally independent of one another given the response value. This is an extremely strong assumption.

Check assumption for male



Check asssmption for female



And it’s somehow good!

Let us go to implantation

Implementation

There are several packages to apply naïve Bayes (

caTools: splitting data for training data and test data

naivebayes:having naive_bayes to build and fitting model

CODE

```
# naive

# Importing the dataset

dataset = read.csv('gender.csv')

# Encoding the target feature as factor

dataset$person= factor(dataset$person,levels = c("male", "female"),labels = c(1,2))

# Splitting the dataset into the Training set and Test set

# install.packages('caTools')

library(caTools)

training_set = subset(dataset, split == TRUE)

set.seed(123)

split = sample.split(dataset$person, SplitRatio = 0.80)

training_set = subset(dataset, split == TRUE)

test_set = subset(dataset, split == FALSE)
```

Fitting naive to the Training set

library(naivebayes)

classifier=naive_bayes(person~Height+Weight+Footsize,usekernel = T,data=training_set)

Create your classifier here

Predicting the Test set results

y_pred = predict(classifier, newdata = test_set[-4])

Making the Confusion Matrix

cm = table(test_set[,4], y_pred)

#build data frame with unseen data to predict

mydata <- data.frame(Height=numeric(0), Weight=numeric(0), Footsize=numeric(0))

mydata <- fix(mydata)

mydata$newprediction=predict(classifier, newdata =mydata)

```
Confusion Matrix and Statistics

          Reference
Prediction 1 2
         1 2 0
         2 0 2

               Accuracy : 1
                 95% CI : (0.3976, 1)
    No Information Rate : 0.5
    P-Value [Acc > NIR] : 0.0625
```

And it is awesome and pretty result.