Assignment 4

Siddharth Harsukh Pansuria | CWID: 20005837

3/6/2022

Clearing Environmental Variables

```
rm(list = ls())
```

Loading the breast cancer dataset and replacing '?' with NA

```
data <- read.csv("breast-cancer-wisconsin.csv", na.strings = '?')
```

Displaying top 10 rows of the data loaded

```
head(data, n = 10)
```

```
##     Sample F1 F2 F3 F4 F5 F6 F7 F8 F9 Class
## 1  1000025  5  1  1  1  2  1  3  1  1     2
## 2  1002945  5  4  4  5  7 10  3  2  1     2
## 3  1015425  3  1  1  1  2  2  3  1  1     2
## 4  1016277  6  8  8  1  3  4  3  7  1     2
## 5  1017023  4  1  1  3  2  1  3  1  1     2
## 6  1017122  8 10 10  8  7 10  9  7  1     4
## 7  1018099  1  1  1  1  2 10  3  1  1     2
## 8  1018561  2  1  2  1  2  1  3  1  1     2
## 9  1033078  2  1  1  1  2  1  1  1  5     2
## 10 1033078  4  2  1  1  2  1  2  1  1     2
```

Deleting rows with missing values

```
data <- na.omit(data)
```

Removing first column since it only contains ID

```
data <- data[-1]
```

Setting column 'Class' as a factor of benign and malignant

```
data$Class <- factor(data$Class, levels = c(2,4), labels = c("benign", "malignant"))
```

Displaying top 10 rows of the data loaded

```
head(data, n = 10)
```

```
##    F1 F2 F3 F4 F5 F6 F7 F8 F9     Class
## 1   5  1  1  1  2  1  3  1  1    benign
## 2   5  4  4  5  7 10  3  2  1    benign
## 3   3  1  1  1  2  2  3  1  1    benign
## 4   6  8  8  1  3  4  3  7  1    benign
## 5   4  1  1  3  2  1  3  1  1    benign
## 6   8 10 10  8  7 10  9  7  1 malignant
## 7   1  1  1  1  2 10  3  1  1    benign
## 8   2  1  2  1  2  1  3  1  1    benign
## 9   2  1  1  1  2  1  1  1  5    benign
## 10  4  2  1  1  2  1  2  1  1    benign
```

Creating a function for min max normalization

```
min_max_norm <- function(x, minX, maxX) {
  ans <- (x - minX)/(maxX - minX)
  return (ans)
}
```

Getting number of columns without the 'Class' column

```
target <- ncol(data) - 1
```

Normalizing the data

```
normalized_data <- data.frame(matrix(NA, nrow = nrow(data), ncol = ncol(data)))
colnames(normalized_data) <- c(colnames(data))

for (i in c(1 : target)) {
  normalized_data[,i] <- min_max_norm(data[,i], min(data[,i]), max(data[,i]))
}

normalized_data[,ncol(data)] <- data[,ncol(data)]
```

Displaying top 10 rows of the data loaded

```r
head(normalized_data, n = 10)
```

```
##           F1        F2        F3        F4        F5        F6        F7
## 1  0.4444444 0.0000000 0.0000000 0.0000000 0.1111111 0.0000000 0.2222222
## 2  0.4444444 0.3333333 0.3333333 0.4444444 0.6666667 1.0000000 0.2222222
## 3  0.2222222 0.0000000 0.0000000 0.0000000 0.1111111 0.1111111 0.2222222
## 4  0.5555556 0.7777778 0.7777778 0.0000000 0.2222222 0.3333333 0.2222222
## 5  0.3333333 0.0000000 0.0000000 0.2222222 0.1111111 0.0000000 0.2222222
## 6  0.7777778 1.0000000 1.0000000 0.7777778 0.6666667 1.0000000 0.8888889
## 7  0.0000000 0.0000000 0.0000000 0.0000000 0.1111111 1.0000000 0.2222222
## 8  0.1111111 0.0000000 0.1111111 0.0000000 0.1111111 0.0000000 0.2222222
## 9  0.1111111 0.0000000 0.0000000 0.0000000 0.1111111 0.0000000 0.0000000
## 10 0.3333333 0.1111111 0.0000000 0.0000000 0.1111111 0.0000000 0.1111111
##           F8        F9     Class
## 1  0.0000000 0.0000000    benign
## 2  0.1111111 0.0000000    benign
## 3  0.0000000 0.0000000    benign
## 4  0.6666667 0.0000000    benign
## 5  0.0000000 0.0000000    benign
## 6  0.6666667 0.0000000 malignant
## 7  0.0000000 0.0000000    benign
## 8  0.0000000 0.0000000    benign
## 9  0.0000000 0.4444444    benign
## 10 0.0000000 0.0000000    benign
```

Splitting the dataset into training and test data

```r
id <- sort(sample(nrow(normalized_data), as.integer(.70 * nrow(normalized_data))))
train_data <- normalized_data[id,]
test_data <- normalized_data[-id,]
```

Installing package to implement Naïve Bayes methodology

```r
install.packages("e1071", dependencies = TRUE, repos = "http://cran.us.r-project.org")
```

```
## Installing package into 'C:/Users/cookie_/Documents/R/win-library/4.1'
## (as 'lib' is unspecified)

## package 'e1071' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'e1071'

## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying C:
## \Users\cookie_\Documents\R\win-library\4.1\00LOCK\e1071\libs\x64\e1071.dll to C:
## \Users\cookie_\Documents\R\win-library\4.1\e1071\libs\x64\e1071.dll: Permission
## denied

## Warning: restored 'e1071'
```

```
##
## The downloaded binary packages are in
##  C:\Users\cookie_\AppData\Local\Temp\RtmpgjuJuV\downloaded_packages
```

```
library(e1071)
```

Implementing Naïve Bayes methodology

```
NaiveBayes <- naiveBayes(Class ~., data = train_data)
predict_NaiveBayes <- predict(NaiveBayes, test_data)
```

Frequency table for the Naïve Bayes predictions

```
table(NaiveBayes = predict_NaiveBayes, Class = test_data$Class)
```

```
##            Class
## NaiveBayes  benign malignant
##   benign       129         2
##   malignant      6        68
```

Calculating the error rate

```
wrong_preds <- sum(predict_NaiveBayes != test_data$Class)
```

Number of wrong predictions

```
error_rate <- wrong_preds/length(predict_NaiveBayes)
print(paste("Error Rate:" , error_rate))
```

Error Rate in Naïve Bayes predictions

```
## [1] "Error Rate: 0.0390243902439024"
```