D) Data mining

i)

> library(caret)

Loading required package: ggplot2

Learn more about the underlying theory at https://ggplot2-book.org/

Loading required package: lattice

> library(tm)

Loading required package: NLP

Attaching package: ‘NLP’

The following object is masked from ‘package:ggplot2’:

annotate

> library(e1071)

> library(SnowballC)

> library(readxl)

> library(ggplot2)

> xlsx\_file <- "C:/Users/nkl/Desktop/spam.xlsx"

> spam\_data <- read\_excel(xlsx\_file)

-/ / - New names:

• `` -> `...3`

• `` -> `...4`

• `` -> `...5`

> print("XLSX file loaded successfully.")

[1] "XLSX file loaded successfully."

>

> head(spam\_data)

# A tibble: 6 × 5

v1 v2 ...3 ...4 ...5

<chr> <chr> <chr> <chr> <chr>

1 ham Go until jurong point, crazy.. Available only in bugi… <NA> <NA> <NA>

2 ham Ok lar... Joking wif u oni... <NA> <NA> <NA>

3 spam Free entry in 2 a wkly comp to win FA Cup final tkts … <NA> <NA> <NA>

4 ham U dun say so early hor... U c already then say... <NA> <NA> <NA>

5 ham Nah I don't think he goes to usf, he lives around her… <NA> <NA> <NA>

6 spam FreeMsg Hey there darling it's been 3 week's now and … <NA> <NA> <NA>

> str(spam\_data)

tibble [5,572 × 5] (S3: tbl\_df/tbl/data.frame)

$ v1 : chr [1:5572] "ham" "ham" "spam" "ham" ...

$ v2 : chr [1:5572] "Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat..." "Ok lar... Joking wif u oni..." "Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question("| \_\_truncated\_\_ "U dun say so early hor... U c already then say..." ...

$ ...3: chr [1:5572] NA NA NA NA ...

$ ...4: chr [1:5572] NA NA NA NA ...

$ ...5: chr [1:5572] NA NA NA NA ...

ii)

> spam\_counts <- table(spam\_data$v1)

> spam\_counts\_df <- as.data.frame(spam\_counts)

> colnames(spam\_counts\_df) <- c("Label", "Count")

> ggplot(spam\_counts\_df, aes(x = Label, y = Count, fill = Label)) +

+ geom\_bar(stat = "identity") +

+ labs(title = "Distribution of Spam and Ham Emails",

+ x = "Email Type",

+ y = "Count") +

+ theme\_minimal() +

+ scale\_fill\_manual(values = c("ham" = "steelblue", "spam" = "red")) +

+ theme(plot.title = element\_text(hjust = 0.5),

+ legend.position = "none")

iii)

> corpus <- VCorpus(VectorSource(spam\_data$v2))

> corpus <- tm\_map(corpus, content\_transformer(tolower))

> corpus <- tm\_map(corpus, removePunctuation)

> corpus <- tm\_map(corpus, removeNumbers)

> corpus <- tm\_map(corpus, removeWords, stopwords("english"))

> corpus <- tm\_map(corpus, stripWhitespace)

> corpus <- tm\_map(corpus, stemDocument)

>

> dtm <- DocumentTermMatrix(corpus)

>

> dtm\_df <- as.data.frame(as.matrix(dtm))

>

> dtm\_df$label <- spam\_data$v1

> dtm\_df$label <- factor(dtm\_df$label)

> set.seed(123)

> trainIndex <- createDataPartition(dtm\_df$label, p = 0.7, list = FALSE)

> trainData <- dtm\_df[trainIndex, ]

> testData <- dtm\_df[-trainIndex, ]

>

>

> nbModel <- naiveBayes(label ~ ., data = trainData)

>

> nbPred <- predict(nbModel, testData)

> nbPred <- predict(nbModel, testData)

> confMatrix <- confusionMatrix(nbPred, testData$label)

> print(confMatrix)

Confusion Matrix and Statistics

Reference

Prediction ham spam

ham 0 0

spam 1447 224

Accuracy : 0.1341

95% CI : (0.1181, 0.1513)

No Information Rate : 0.8659

P-Value [Acc > NIR] : 1

Kappa : 0

Mcnemar's Test P-Value : <2e-16

Sensitivity : 0.0000

Specificity : 1.0000

Pos Pred Value : NaN

Neg Pred Value : 0.1341

Prevalence : 0.8659

Detection Rate : 0.0000

Detection Prevalence : 0.0000

Balanced Accuracy : 0.5000

'Positive' Class : ham

> accuracy <- sum(nbPred == testData$label) / nrow(testData)

> print(paste("Accuracy:", round(accuracy, 4)))

[1] "Accuracy: 0.1341"

>

Iv We are using Naïve Bayers methods to classify the email either it spam or ham. The result from the data processing is that more ham than spam email receive.

e) Evaluation

i)

> confMatrix <- confusionMatrix(nbPred, testData$label)

> print(confMatrix)

Confusion Matrix and Statistics

Reference

Prediction ham spam

ham 0 0

spam 1447 224

Accuracy : 0.1341

95% CI : (0.1181, 0.1513)

No Information Rate : 0.8659

P-Value [Acc > NIR] : 1

Kappa : 0

Mcnemar's Test P-Value : <2e-16

Sensitivity : 0.0000

Specificity : 1.0000

Pos Pred Value : NaN

Neg Pred Value : 0.1341

Prevalence : 0.8659

Detection Rate : 0.0000

Detection Prevalence : 0.0000

Balanced Accuracy : 0.5000

'Positive' Class : ham

> accuracy <- sum(nbPred == testData$label) / nrow(testData)

> print(paste("Accuracy:", round(accuracy, 4)))

[1] "Accuracy: 0.1341"

>

>

> nbProb <- predict(nbModel, testData, type = "raw")

> rocCurve <- roc(testData$label, nbProb[, 2], levels = rev(levels(testData$label)))

Error in roc(testData$label, nbProb[, 2], levels = rev(levels(testData$label))) :

could not find function "roc"

> plot(rocCurve, col = "blue", main = "ROC Curve for Naive Bayes Model")

Error: object 'rocCurve' not found

> library(pROC)

Type 'citation("pROC")' for a citation.

Attaching package: ‘pROC’

The following objects are masked from ‘package:stats’:

cov, smooth, var

> rocCurve <- roc(testData$label, nbProb[, "spam"], levels = rev(levels(testData$label)))

Setting direction: controls < cases

> rocCurve <- roc(testData$label, nbProb[, 2], levels = rev(levels(testData$label)))

Setting direction: controls < cases

> plot(rocCurve, col = "blue", main = "ROC Curve for Naive Bayes Model")

> aucValue <- auc(rocCurve)

> print(paste("AUC:", round(aucValue, 4)))

[1] "AUC: 0.5"

> confMatrixDF <- as.data.frame(confMatrix$table)

> ggplot(confMatrixDF, aes(Prediction, Reference, fill = Freq)) +

+ geom\_tile() +

+ scale\_fill\_gradient(low = "white", high = "red") +

+ geom\_text(aes(label = Freq), color = "black", size = 4) +

+ labs(title = "Confusion Matrix", x = "Predicted", y = "Actual") +

+ theme\_minimal()

ii) the Naïve Bayers methods, is showing best performance in classifying spam and ham email. The parameter tuning really helpful in making results more accurate.