```
R version 4.3.3 (2024-02-29 ucrt) -- "Angel Food Cake"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

> 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 co
mp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question("| __trunca
ted__ "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 ...
> 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$label, p = 0.7, list = FALSE)
> trainData <- dtm_df[trainIndex, ]
> testData  <- dtm_df[-trainIndex, ]
>
>
> nbModel <- naiveBayes(label ~ ., data = trainData)
>
> nbPred <- predict(nbModel, testData)
> spam_counts <- table(spam_data$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")
> 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"
>
>
> 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()
> save.image("C:\\Users\\nkl\\Desktop\\GROUP SPAM MAIL\\spam mail r")
>
```