```
# ------------------------------------------------------------------------------
#
# Generative Models - comprehension check
#
# ------------------------------------------------------------------------------

# Setup
library(tidyverse)
library(dslabs)
library(dplyr)
library(ggplot2)
library(Lahman)
library(HistData)
library(caret)
library(e1071)
library(matrixStats)

#
# Q1
#

# Create a dataset of samples from just cerebellum and hippocampus, two parts of
# the brain, and a predictor matrix with 10 randomly selected columns using the
# following code:

set.seed(1993)
data("tissue_gene_expression")
ind <- which(tissue_gene_expression$y %in% c("cerebellum", "hippocampus"))
y <- droplevels(tissue_gene_expression$y[ind])
x <- tissue_gene_expression$x[ind, ]
x <- x[, sample(ncol(x), 10)]

# Use the train function to estimate the accuracy of LDA. What is the accuracy?


#
# Q2
#

# In this case, LDA fits two 10-dimensional normal distributions. Look at the
# fitted model by looking at the finalModel component of the result of train.
# Notice there is a component called means that includes the estimated means of
# both distributions. Plot the mean vectors against each other and determine
# which predictors (genes) appear to be driving the algorithm.

# Which TWO genes appear to be driving the algorithm?
# PLCB1
# RAB1B
# MSH4
# OAZ2
# SPI1
# SAPCD1
# HEMK1


#
# Q3
#

# Repeat the exercise in Q1 with QDA. Create a dataset of samples from just
# cerebellum and hippocampus, two parts of the brain, and a predictor matrix
# with 10 randomly selected columns using the following code:

set.seed(1993)
data("tissue_gene_expression")
ind <- which(tissue_gene_expression$y %in% c("cerebellum", "hippocampus"))
y <- droplevels(tissue_gene_expression$y[ind])
x <- tissue_gene_expression$x[ind, ]
x <- x[, sample(ncol(x), 10)]

# Use the train function to estimate the accuracy of QDA.
# What is the accuracy?


#
# Q4
#

# Which TWO genes drive the algorithm when using QDA instead of LDA?
# PLCB1
# RAB1B
# MSH4
# OAZ2
# SPI1
# SAPCD1
```

```
# HEMK1


#
# Q5

# One thing we saw in the previous plots is that the values of the predictors
# correlate in both groups: some predictors are low in both groups and others
# high in both groups. The mean value of each predictor found in colMeans(x) is
# not informative or useful for prediction and often for purposes of
# interpretation, it is useful to center or scale each column. This can be
# achieved with the preProcessing argument in train. Re-run LDA with
# preProcessing = "scale". Note that accuracy does not change, but it is now
# easier to identify the predictors that differ more between groups than based
# on the plot made in Q2.

# Which TWO genes drive the algorithm after performing the scaling?
# C21orf62
# PLCB1
# RAB1B
# MSH4
# OAZ2
# SPI1
# SAPCD1
# IL18R1


#
# Q6
#

# Now we are going to increase the complexity of the challenge slightly: we will
# consider all the tissue types. Use the following code to create your dataset:

set.seed(1993)
data("tissue_gene_expression")
y <- tissue_gene_expression$y
x <- tissue_gene_expression$x
x <- x[, sample(ncol(x), 10)]

# What is the accuracy using LDA?
```