```
################################################################################
## Course: Machine Learning for Economists and Business Analysts
## Topic: Penalized Regression
################################################################################


getwd()
#setwd("")

################################################################################
################################################################################

# Install packages
#install.packages("glmnet")
#install.packages("corrplot")

# Load packages
library(glmnet)
library(corrplot)

# Load data
load("student-mat-train.Rdata")
load("student-mat-test.Rdata")

#####################################################
### Exercise 1: Data Description and Visualization ###
#####################################################

################################################################################
# Task 1
################################################################################

# No. of observations
nrow(train)
nrow(test)

################################################################################
# Task 2
################################################################################

# Average grade
mean(train$G3)

# Minimum grade
min(train$G3)

# Maximum grade
max(train$G3)

################################################################################
# Task 3
################################################################################

# Histogram of the final grades (with base R)
hist(train$G3, breaks = 17, main = "Histogram - Final Grades", xlab = "Grades")

#install.packages("ggplot2")
#library(ggplot2)

# Histogram of the final grades (with ggplot2)
ggplot(aes(x = G3), data = train) +
  geom_histogram(bins = 16, col = "white") +
  xlab("Grades") +
  labs(title = "Histogram - Final Grades")

################################################################################
# Task 4
################################################################################

# Correlation analysis
cor <- round(cor(train[,c(1:25)]),2) # Variable 26 is the depedendent variable
corrplot(cor)


################################################################################
################################################################################

########################
### Exercise 2: OLS ###
########################

################################################################################
# Task 1
################################################################################
```

```
# OLS
ols <- lm(G3 ~ ., data = train)
summary(ols)

# Calculate the MSE
test$predols <- predict(ols, newdata = test)

predMSEols <- mean((test$G3 - test$predols)^2)
print(predMSEols)

#############################################################################
# Task 3
#############################################################################

ols_parsimonious <- lm(G3 ~ failures , data = train)
summary(ols_parsimonious)

# Calculate the MSE
test$predols_parsimonious <- predict(ols_parsimonious, newdata = test)

predMSEols_parsimonious <- mean((test$G3 - test$predols_parsimonious)^2)
print(predMSEols_parsimonious)

#############################################################################
#############################################################################

###################################
### Exercise 3: Lasso and Ridge ###
###################################

#############################################################################
# Task 1
#############################################################################

# Whole lasso path
lasso <- glmnet(as.matrix(train[,c(1:25)]), train$G3, family = "gaussian", alpha = 1)
plot(lasso, label = TRUE)

set.seed(27112019)
lasso.cv <- cv.glmnet(as.matrix(train[,c(1:25)]), train$G3, type.measure = "mse", family = "gaussian", nfolds = 5,
alpha = 1)
coef_lasso1 <- coef(lasso.cv, s = "lambda.min") # save for later comparison
print(coef_lasso1)
plot(lasso.cv)

# Calculate the MSE
test$predlasso <- predict(lasso.cv, newx = as.matrix(test[,c(1:25)]), s = lasso.cv$lambda.min)

predMSElasso <- mean((test$G3 - test$predlasso)^2)
print(predMSElasso)

#############################################################################
# Task 2
#############################################################################

# Another lasso solution based on the 5-fold cross validation
set.seed(27112025) # 27112024
lasso.cv <- cv.glmnet(as.matrix(train[,c(1:25)]), train$G3, type.measure = "mse", family = "gaussian", nfolds = 5,
alpha = 1)
coef_lasso2 <- coef(lasso.cv, s = "lambda.min")

print(cbind(coef_lasso1, coef_lasso2))

# Calculate the correlation between the predictions in task 1 and 2
test$predlasso2 <- predict(lasso.cv, newx = as.matrix(test[,c(1:25)]), s = lasso.cv$lambda.min)
cor(test$predlasso,test$predlasso2)

#############################################################################
# Task 3
#############################################################################

# Ridge
ridge <- glmnet(as.matrix(train[,c(1:25)]), train$G3, family = "gaussian", alpha = 0)
plot(ridge, label = TRUE)

set.seed(27112019)
ridge.cv <- cv.glmnet(as.matrix(train[,c(1:25)]), train$G3, type.measure = "mse", nfolds = 5, family = "gaussian",
alpha = 0)
coef_ridge <- coef(ridge.cv, s = "lambda.min") # save for later comparison
print(coef_ridge)
plot(ridge.cv)

# Calculate the MSE
test$predridge <- predict(ridge, newx = as.matrix(test[,c(1:25)]), s = ridge.cv$lambda.min)
```

```
predMSEridge <- mean((test$G3 - test$predridge)^2)
print(predMSEridge)

#############################################################################
# Task 4
#############################################################################

# Report the coefficients of Dalc and Walc from the OLS, Ridge and Lasso models
corcoeff <- cbind(coef(ols)[23:24], coef_ridge[23:24], coef_lasso1[23:24]) # Pick Dalc and Walc
colnames(corcoeff) <- c("OLS", "Ridge", "Lasso")
print(corcoeff)


#############################################################################
# Task 5
#############################################################################

# Print the MSE of the OLS, Lasso and Ridge models
print(c(predMSEols, predMSElasso, predMSEridge))

#############################################################################
# Task 6
#############################################################################

# Visualize the predictions (Predicted vs Actual)
plot(test$G3,test$predols,xlim=c(5,20),ylim=c(4,16), col= "darkgreen", xlab = "Observed Grades", ylab = "Predicted
Grades" )
par(new=TRUE)
plot(test$G3,test$predlasso,xlim=c(5,20),ylim=c(4,16), col= "blue", xlab = "", ylab = "" )
par(new=TRUE)
plot(test$G3,test$predridge,xlim=c(5,20),ylim=c(4,16), col= "red", xlab = "", ylab = "" )
abline(a=0,b=1)
legend(16, 9, c("OLS", "Lasso", "Ridge"), col = c("darkgreen", "blue", "red"), pch = c(21, 21, 21))
```