```
#################### Tópico de econometría: Microeconometría
 ######################
#################### Karoll Gómez Portilla ######################
#################### Universidad Nacional de Colombia ######################
#################### Lab 1.1: Regression Methods: OLS ######################

# Loading packages
library(carData)
library(car)
library(zoo)
library(lmtest)
library(sandwich)
library(survival)
library(AER)

# -----------------------------Example 1: Single regression
 --------------------------------

# We want to relate test scores to student-teacher ratios measured in
 Californian schools. The test score is the district-wide average of reading
 and math scores for fifth graders. The class size is measured as the number of
 students divided by the number of teachers (the student-teacher ratio). The
 California School data set (CASchools) comes with an R package called AER.

# load the the data set in the workspace
data(CASchools)
class(CASchools)
head(CASchools)

# The two variables we are interested in (i.e., average test score and the
 student-teacher ratio) are not included. However, it is possible to calculate
 both from the provided data. To obtain the student-teacher ratios, we simply
 divide the number of students by the number of teachers. The average test
 score is the arithmetic mean of the test score for reading and the score of
 the math test. The next code chunk shows how the two variables can be
 constructed as vectors and how they are appended to CASchools.

# Compute STR and append it to CASchools
CASchools$STR <- CASchools$students/CASchools$teachers

# Compute TestScore and append it to CASchools
CASchools$score <- (CASchools$read + CASchools$math)/2

# Compute sample averages of STR and score
avg_STR <- mean(CASchools$STR)
avg_score <- mean(CASchools$score)

# Compute sample standard deviations of STR and score
sd_STR <- sd(CASchools$STR)
sd_score <- sd(CASchools$score)
```

```
# Set up a vector of percentiles and compute the quantiles
quantiles <- c(0.10, 0.25, 0.4, 0.5, 0.6, 0.75, 0.9)
quant_STR <- quantile(CASchools$STR, quantiles)
quant_score <- quantile(CASchools$score, quantiles)

# Gather everything in a data.frame
DistributionSummary <- data.frame(Average = c(avg_STR, avg_score),
                                  StandardDeviation = c(sd_STR, sd_score),
                                  quantile = rbind(quant_STR, quant_score))

# Print the summary to the console
DistributionSummary

plot(score ~ STR,
     data = CASchools,
     main = "Scatterplot of TestScore and STR",
     xlab = "STR (X)",
     ylab = "Test Score (Y)")

# Compute the correlation between STR and score
cor(CASchools$STR, CASchools$score)

# The goal is to test if STR determines the score. Estimate a OLS regression by
 hand
attach(CASchools) # allows to use the variables contained in CASchools directly

# compute beta_1_hat
beta_1 <- sum((STR - mean(STR)) * (score - mean(score))) / sum((STR -
 mean(STR))^2)

# compute beta_0_hat
beta_0 <- mean(score) - beta_1 * mean(STR)

# print the results to the console
beta_1
beta_0

# Estimate the model and assign the result to linear_model
linear_model <- lm(score ~ STR, data = CASchools)

# Print the standard output of the estimated lm object to the console
linear_model
mod_summary <- summary(linear_model)
mod_summary

# plot the data
plot(score ~ STR,
     data = CASchools,
     main = "Scatterplot of TestScore and STR",
     xlab = "STR (X)",
     ylab = "Test Score (Y)",
     xlim = c(10, 30),
     ylim = c(600, 720))

# Add the regression line
abline(linear_model)

# Compute R^2 manually
SSR <- sum(mod_summary$residuals^2)
TSS <- sum((score - mean(score))^2)
R2 <- 1 - SSR/TSS

# Print the value to the console
R2

#### -----------------------------Example 2: Multiple regression: now include
 more explanatory variables ----------------
# set seed for reproducibility
set.seed(1)

# generate artificial data on location
CASchools$direction <- sample(c("West", "North", "South", "East"),
                              420,
                              replace = T)

# estimate the model
mult.mod <- lm(score ~ STR + english + direction, data = CASchools)

# obtain a model summary
summary(mult.mod)

#################### Assumptions verification
# The Error Term has Conditional Mean of Zero
# set a seed to make the results reproducible
set.seed(321)

# simulate the data
X <- runif(50, min = -5, max = 5)
u <- rnorm(50, sd = 5)

# the true relation
Y <- X^2 + 2 * X + u

# estimate a simple regression model
mod_simple <- lm(Y ~ X)

# predict using a quadratic model
prediction <- predict(lm(Y ~ X + I(X^2)), data.frame(X = sort(X)))

# plot the results
plot(Y ~ X)
abline(mod_simple, col = "red")
```

```
lines(sort(X), prediction)

# Using the quadratic model (represented by the black curve) we see that there
 are no systematic deviations of the observation from the predicted relation.
 It is credible that the assumption is not violated when such a model is
 employed. However, using a simple linear regression model we see that the
 assumption is probably violated

# Assump 2: Independently and Identically Distributed Data
# set seed
set.seed(123)

# generate a date vector
Date <- seq(as.Date("1951/1/1"), as.Date("2000/1/1"), "years")

# initialize the employment vector
X <- c(5000, rep(NA, length(Date)-1))

# generate time series observations with random influences
for (i in 2:length(Date)) {

    X[i] <- -50 + 0.98 * X[i-1] + rnorm(n = 1, sd = 200)
}

#plot the results
plot(x = Date, y = X, type = "l", col = "steelblue", ylab = "Workers", xlab =
 "Time")

# It is evident that the observations on the number of employees cannot be
 independent in this example: the level of today’s employment is correlated
 with tomorrows employment level. Thus, the i.i.d. assumption is violated.
```