The run\_analysis.R script performs the data preparation and then followed by the 5 steps required as described in the course project’s definition.

1. **Download the dataset**
   * Dataset downloaded and extracted under the folder called UCI HAR Dataset

filename <- "Coursera\_DS3\_Final.zip"

# Checking if archieve already exists.

if (!file.exists(filename)){

fileURL <- "https://d396qusza40orc.cloudfront.net/getdata%2Fprojectfiles%2FUCI%20HAR%20Dataset.zip"

download.file(fileURL, filename, method="curl")

}

# Checking if folder exists

if (!file.exists("UCI HAR Dataset")) {

unzip(filename)

}

1. **Assign each data to variables**

features <- read.table("UCI HAR Dataset/features.txt", col.names = c("n","functions"))

activities <- read.table("UCI HAR Dataset/activity\_labels.txt", col.names = c("code", "activity"))

subject\_test <- read.table("UCI HAR Dataset/test/subject\_test.txt", col.names = "subject")

x\_test <- read.table("UCI HAR Dataset/test/X\_test.txt", col.names = features$functions)

y\_test <- read.table("UCI HAR Dataset/test/y\_test.txt", col.names = "code")

subject\_train <- read.table("UCI HAR Dataset/train/subject\_train.txt", col.names = "subject")

x\_train <- read.table("UCI HAR Dataset/train/X\_train.txt", col.names = features$functions)

y\_train <- read.table("UCI HAR Dataset/train/y\_train.txt", col.names = "code")

1. **Merges the training and the test sets to create one data set**

X <- rbind(x\_train, x\_test)

Y <- rbind(y\_train, y\_test)

Subject <- rbind(subject\_train, subject\_test)

Merged\_Data <- cbind(Subject, Y, X)

1. **Extracts only the measurements on the mean and standard deviation for each measurement**

TidyData <- Merged\_Data %>% select(subject, code, contains("mean"), contains("std"))

1. **Uses descriptive activity names to name the activities in the data set**

names(TidyData)[2] = "activity"

1. **Appropriately labels the data set with descriptive variable names**

names(TidyData)<-gsub("Acc", "Accelerometer", names(TidyData))

names(TidyData)<-gsub("Gyro", "Gyroscope", names(TidyData))

names(TidyData)<-gsub("BodyBody", "Body", names(TidyData))

names(TidyData)<-gsub("Mag", "Magnitude", names(TidyData))

names(TidyData)<-gsub("^t", "Time", names(TidyData))

names(TidyData)<-gsub("^f", "Frequency", names(TidyData))

names(TidyData)<-gsub("tBody", "TimeBody", names(TidyData))

names(TidyData)<-gsub("-mean()", "Mean", names(TidyData), ignore.case = TRUE)

names(TidyData)<-gsub("-std()", "STD", names(TidyData), ignore.case = TRUE)

names(TidyData)<-gsub("-freq()", "Frequency", names(TidyData), ignore.case = TRUE)

names(TidyData)<-gsub("angle", "Angle", names(TidyData))

names(TidyData)<-gsub("gravity", "Gravity", names(TidyData))

1. **From the data set in step 4, creates a second, independent tidy data set with the average of each variable for each activity and each subject**

FinalData <- TidyData %>%

group\_by(subject, activity) %>%

summarise\_all(funs(mean))

write.table(FinalData, "FinalData.txt", row.name=FALSE)

str(FinalData)

FinalData