Getting and Cleaning Data Course Project

Shahadat Hossain

2/5/2020

Background

The purpose of this project is to demonstrate your ability to collect, work with, and clean a data set. The goal is to prepare tidy data that can be used for later analysis. You will be graded by your peers on a series of yes/no questions related to the project. You will be required to submit:

1. a tidy data set as described below,
2. a link to a Github repository with your script for performing the analysis, and
3. a code book that describes the variables, the data, and any transformations or work that you performed to clean up the data called `CodeBook.md`.

You should also include a `README.md` in the repo with your scripts. This repo explains how all of the scripts work and how they are connected.

One of the most exciting areas in all of data science right now is wearable computing. Companies like Fitbit, Nike, and Jawbone Up are racing to develop the most advanced algorithms to attract new users. The data linked to from the course website represent data collected from the accelerometers from the Samsung Galaxy S smartphone. A full description is available at the site where the data was obtained:

http://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones

Here are the data for the project:

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

```
# Downloading data for the project

# url = "https://d396qusza40orc.cloudfront.net/getdata%2Fprojectfiles%2FUCI%20HAR%20Dataset.zip"
# download.file(url = url, destfile = "projectfile.zip")
# unzip(zipfile = "projectfile.zip", overwrite = TRUE)
```

Objective

You should create one R script called `run_analysis.R` that does the following.

1. Merges the training and the test sets to create one data set.
2. Extracts only the measurements on the mean and standard deviation for each measurement.
3. Uses descriptive activity names to name the activities in the data set
4. Appropriately labels the data set with descriptive variable names.
5. 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.

```
# Loading packages and data
library(tidyverse)
library(data.table)

# Loading activity labels and features
```

```
activitylabels <- fread(file.path("UCI HAR Dataset/activity_labels.txt"),
                        col.names = c("classLabels", "activityName"))
features <- fread(file.path("UCI HAR Dataset/features.txt"),
                  col.names = c("index", "featureNames"))

featnames <- features %>%
  mutate(rownumber = row_number()) %>%
  filter(grepl("(mean|std)\\(\\)", featureNames) == TRUE) %>%
  mutate(featureNames = gsub("[()]", "", featureNames),
         featureNames = gsub("-", "_", featureNames))

# Loading training data

train <- fread(input = "UCI HAR Dataset/train/X_train.txt") %>%
  select(c(featnames$rownumber))

names(train) <- featnames$featureNames

trainActivities <- fread(input = "UCI HAR Dataset/train/Y_train.txt",
                         col.names = c("Activity"))

trainSubjects <- fread(input = "UCI HAR Dataset/train/subject_train.txt",
                       col.names = c("SubjectNum"))

train <- cbind(trainSubjects, trainActivities, train)


# Loading test data

test <- fread(input = "UCI HAR Dataset/test/X_test.txt") %>%
  select(c(featnames$rownumber))

names(test) <- featnames$featureNames

testActivities <- fread(input = "UCI HAR Dataset/test/Y_test.txt",
                        col.names = c("Activity"))

testSubjects <- fread(input = "UCI HAR Dataset/test/subject_test.txt",
                      col.names = c("SubjectNum"))

test <- cbind(testSubjects, testActivities, test)

# merge datasets
combined <- rbind(train, test)

# Relebeling variable
# Making wide by

combined <- combined %>%
  mutate(Activity = factor(Activity,
                           levels = activitylabels$classLabels,
                           labels = activitylabels$activityName),
         SubjectNum = as.factor(SubjectNum)) %>%
```

```
  gather(variable, value, -c(Activity, SubjectNum)) %>%
  group_by(Activity, SubjectNum, variable) %>%
  summarise(value = mean(value, na.rm = TRUE)) %>%
  spread(variable, value)

fwrite(x = combined, file = "tidyData.txt", quote = FALSE)
```