Data Cleaning Project

Ahmed Ezzat

20-11-2014

As always start any task by carefully reading the task requirements and for this task the data README file is very important. especially the part of what files contains what.

I thought using packages like plyr or reshapre is going to make things more confusing for you, so i didn't use them extensively, but one function of plyr package is really worth using because the standard R way of doing it without it is going to be way more confusing. so you might want first to run

```
install.packages('plyr')
```

The process is quite long and the full script will take some seconds to run, *not very much and depends on your machine* but its more than couple of seconds, so i will break every step in the script to explain it, then I will paste it as a whole at the end. I feel it would be best if you write down the script yourself and start writing it line by line to check how the data looks like at every step

So our steps are

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 activity names. ######5.- Creates a second, independent tidy data set with the average of each variable for each activity and each subject.**

So let's start with step 1

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

First we need to locate the data on the disk and read them

First let's locate the test data

```
#Change to your path
test_data <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\test\\X_test.txt")
test_subject <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\test\\subject_test.txt")
test_activity <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\test\\y_test.txt")
```

Then the train data

```
train_data <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\train\\X_train.txt")
train_subject <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\train\\subject_train.txt")
train_activity <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\train\\y_train.txt")
```

Now step 1 is simply combine all tables into two data sets for test and train data.

```
BigDataTablelTest <- cbind(test_subject, test_activity, test_data)
BigDataTablelTrain <- cbind(train_subject, train_activity, train_data)
```

Now build the big data set

```
BigDataTable <- rbind(BigDataTablelTest, BigDataTablelTrain)
```

Leave step 2 for now, we will process it later on the last step

3.- Uses descriptive activity names to name the activities in the data set

This is an easy one, all we want to do is to map data inside activity_labels.txt “check the file to see how it looks like” numbers to the numbers we have in the Second column of our Total data set. To make this R has a very good built in function called match,

```
#Remember to change the path
Activity <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\activity_labels.txt")
#This means add to the second column of our Tota data.froame
#Whatever value you match on the Activity data we read
BigDataTable[,2] <- Activity[match(BigDataTable[,2], Activity[,1]),2]
```

4.- Appropriately labels the data set with descriptive activity names.

This means we have to insert descriptive value names “column names” according to what we have in features.txt file and remember that the subject column should remain as subject because it descries subjects and y test column descries activities

```
#After the first line take a look at Features
#View(Features) to see how it was loaded
Features <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\/features.txt")
#Create a variable to keep names
features_names <- as.vector(Features[,2])
#Another Variable just to add what we want Subject and Activity unchanged
features_names2 <- c("Subject","Activity", features_names)
#Apply column names to our Tota data
names(BigDataTable) <- features_names2
```

2.- Extracts only the measurements on the mean and standard deviation for each measurement.

Sorry for the confusion but i feel its best to start this step at this moment,

We want to extract only the measurements of mean and sd on all columns

In the original data set (the “X-files”) the original measurements are located in 18 columns. Namely, there are three sets for “tBody”, “tGravity” and “tBodyGyro”. For each of those 3 sets, there are 3 dimensions - the X, Y and Z dimensions. This makes 3x3 = 9 columns. Finally, for each combination, we keep both the results for the Mean and Standard Deviation (SD). So the total number of columns to keep = 2 x 9 = 18 columns.

The actual names of the columns are already defined in the previous step, in the help vector “features_names22”. The column naming in this vector is identical to the column naming in the table containing all the information

```
NameColumnKeep <- c(features_names2[1],features_names2[2],
                    features_names2[3],features_names2[4],features_names2[5],
                    features_names2[6],features_names2[7],features_names2[8],
                    features_names2[43],features_names2[44],features_names2[45],
                    features_names2[46],features_names2[47],features_names2[48],
                    features_names2[123],features_names2[124],features_names2[125],
                    features_names2[126],features_names2[127],features_names2[128])
BigDataTable <- BigDataTable[NameColumnKeep]
```

5.- Creates a second, independent tidy data set with the average of each variable for each activity and each subject.

Since the course discussed the plyr its good to put a function of it in use

?ddply => Split data frame, apply function, and return results in a data frame. exactly what we wants in a simple and elegant function

we want to split data by each subject doing each activity. and apply the mean function to it. so it goes like this

```
BigDataTableTidy <- ddply(BigDataTable, .(Subject, Activity), numcolwise(mean))
```

and we simply write the Total data to a file

```
write.table(BigDataTableTidy, file = "BigDataTableTidy.txt")
```

The whole script

```
library(plyr)

x <- c("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\")

# Reading the data
test_data <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\test\\X_test.txt")
test_subject <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\test\\subject_test.txt")
test_activity <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\test\\y_test.txt")

BigDataTablelTest <- cbind(test_subject, test_activity, test_data)
#
train_data <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\train\\X_train.txt")
train_subject <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\train\\subject_train.txt")
train_activity <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\train\\y_train.txt")

#Binding all columns
BigDataTablelTest <- cbind(test_subject, test_activity, test_data)
BigDataTablelTrain <- cbind(train_subject, train_activity, train_data)
#
```

```
# Merging all data into one big table ie: stacking rows of training and test data
BigDataTable <- rbind(BigDataTablelTest, BigDataTablelTrain)

#Reading the activity and maching it to replace numbers with activity names

Activity <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\activity_labels.txt")
BigDataTable[,2] <- Activity[match(BigDataTable[,2], Activity[,1]),2]



# Adding the descriptive names to the dataset
Features <- read.table("C:\\Users\\is7yX\\Documents\\UCI HAR Dataset\\/features.txt")
features_names <- as.vector(Features[,2])
features_names2 <- c("Subject","Activity", features_names)
names(BigDataTable) <- features_names2

#subtracts only the original measurements on the mean and SD.
NameColumnKeep <- c(features_names2[1],features_names2[2],
                    features_names2[3],features_names2[4],features_names2[5],
                    features_names2[6],features_names2[7],features_names2[8],
                    features_names2[43],features_names2[44],features_names2[45],
                    features_names2[46],features_names2[47],features_names2[48],
                    features_names2[123],features_names2[124],features_names2[125],
                    features_names2[126],features_names2[127],features_names2[128])
BigDataTable <- BigDataTable[NameColumnKeep]



#Creates a second, independent tidy data set with the average of each
#variable for each activity and each subject
BigDataTableTidy <- ddply(BigDataTable, .(Subject, Activity), numcolwise(mean))
write.table(BigDataTableTidy, file = "BigDataTableTidy.txt")
```

The Code Book

Well, you can get the column names from the data set

```
colnames(BigDataTableTidy)
```

then try to organize it your own way, check this thread (https://class.coursera.org/getdata-009/forum/thread?thread_id=89)

The README.md

Its importnat too to add a README.md file, just descripe in your own words every step the script does. since its human graded its important to to remove my comments on the code and add your's what what you understood. maybe change variable names too :D

That's pretty much it. :)