Getting and Cleaning Data Course Project

CodeBook with code

This dataset is a subset of data collected from the link below.

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

The original dateset has data collect during a experiments have been carried out with a group of 30 volunteers within an age bracket of 19-48 years. Each person performed six activities (WALKING, WALKING_UPSTAIRS, WALKING_DOWNSTAIRS, SITTING, STANDING, LAYING) wearing a smartphone (Samsung Galaxy S II) on the waist.

For more deitails about the original date set please take a look in this link

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

Steps to reach the gol of Two tidy data sets

Loads all libraries need for this analysis

```
library(tidyr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:lubridate':
##
##     intersect, setdiff, union
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union
```

```
library(stringr)
```

Define some constans to be used in the Script

```
  DataFilePath <- "./data/UCI HAR Dataset"
     TrainPath <- "./data/UCI HAR Dataset/train/"
      TestPath <- "./data/UCI HAR Dataset/test/"

   ZipFileName <- "./data/dataforprojetc.zip"

           Url <- "https://d396qusza40orc.cloudfront.net/getdata%2Fprojectfiles%2FUC
I%20HAR%20Dataset.zip"
```

First I download the original date set, Unziped on *./data* directory

If data directory dont exist create it

```
      if ( !dir.exists("./data") )
           dir.create("./data")
```

Avoid to download file if it is already was download saving time during debugin time

```
      if( !file.exists(ZipFileName) )
          download.file(Url, ZipFileName)
```

Avoid to unzip files if it is already unziped

```
      if( !dir.exists(DataFilePath) )
           unzip( ZipFileName, exdir = "./data/" )
```

Loaded activity_labels and features(variables names) into R objects

```
      activity_labels  <- read.delim2("./data/UCI HAR Dataset/activity_labels.txt",
                          header = FALSE, sep="", stringsAsFactors = FALSE,
                          col.names = c("ActivityId","ActivityName"))

             features  <- read.delim2("./data/UCI HAR Dataset/features.txt",
                          header = FALSE, sep="", stringsAsFactors = FALSE,
                          col.names = c("FeatureId","FeatureDescription"))

             features  <- tbl_df(features) %>% select(FeatureDescription)
```

Make Variables names more clean I took out characters “*-()*”

```
    features$FeatureDescription <- gsub("[-()]", "",features$FeatureDescription )
```

Loaded Train data files (*Xtrain.txt, y_train.txt and subject_train.txt*) into R DataFrames - Those files are loacated in “*./data/UCI HAR Dataset/train/*”

```
  xtraindf  <- read.delim2("./data/UCI HAR Dataset/train/X_train.txt", header = FALSE
, sep="",
                          stringsAsFactors = FALSE, dec = ".", numerals = "no.loss",
                          col.names = features$FeatureDescription )

  ytraindf  <- read.delim2("./data/UCI HAR Dataset/train/y_train.txt", header = FALSE
, sep="",
                           stringsAsFactors = FALSE, dec = ".", numerals = "no.loss",
                           col.names = "ActivityId" )

  subjecttraindf <- read.delim2("./data/UCI HAR Dataset/train/subject_train.txt", hea
der = FALSE, sep="",
                           stringsAsFactors = FALSE, dec = ".", numerals = "no.loss",
                           col.names = "SubjectNum" )

    traindf <- bind_cols(subjecttraindf, ytraindf, xtraindf)
```

Loaded Test data files (*Xtest.txt, y_teste.txt and subject_test.txt*) into R DataFrames - Those files are loacated in “*./data/UCI HAR Dataset/test/*” Creating **traindf** and **testDf**

```
   xtestdf  <- read.delim2("./data/UCI HAR Dataset/test/X_test.txt", header = FALSE,
 sep="",
                             stringsAsFactors = FALSE, dec = ".", numerals = "no.los
s",
                             col.names = features$FeatureDescription )

    ytestdf  <- read.delim2("./data/UCI HAR Dataset/test/y_test.txt", header = FALSE,
sep="",
                             stringsAsFactors = FALSE, dec = ".", numerals = "no.los
s",
                             col.names = "ActivityId")

    subjecttestdf <- read.delim2("./data/UCI HAR Dataset/test/subject_test.txt", head
er = FALSE, sep="",
                                  stringsAsFactors = FALSE, dec = ".", numerals = "n
o.loss",
                                  col.names = "SubjectNum")

    testdf <- bind_cols(subjecttestdf, ytestdf, xtestdf)
```

Merge **traindf** and *testdf* into *TrainTestdf* tha contains all records from **traindf** an **testdf**

```
  TrainTest <- bind_rows(traindf, testdf)
```

Joing activity names to **TrainTest** Data set in order to use descriptive activity names to name the activities in the data set

```
TrainTest <-  left_join(activity_labels, TrainTest)
```

```
## Joining, by = "ActivityId"
```

Selects only the measures for **SDT** and **Means** and Arrange by Subject & ActivityName genarating a **TidyDataSet**

```
    TidyDataset <-  select(TrainTest, SubjectNum, ActivityName, contains("STD", ignor
e.case = TRUE),
                           contains("mean", ignore.case = TRUE ) ) %>%
                           arrange(SubjectNum, ActivityName)
```

Write **TidyDataSet** to a CSV file to prepare to submit

```
 write.csv(TidyDataset, "./TidyDataSet.csv")
```

For each record it is provided:

- Subject who performed the activity that generat the data
- Activity Label
- A 88 -feature vector with time and frequency domain variables, for detais about feature names see TidydataVar.txt

Create a second independentely dataset **SummarizedTidyDataSet** that summarizes **TidyDataSet** with the means of every Variable grouped by Subject & Activity

```
  SummarizedTidyDataSet <- TidyDataset %>% group_by(SubjectNum, ActivityName) %>%
                           summarize_all(mean )
```

Write **SummarizedTidyDataSet**to a CSV file to prepare to submit

```
   write.csv(SummarizedTidyDataSet, "./SummarizedTidyDateSet.csv")
```

Zip datasets to prepare for submit

```
   zip(zipfile = "./TidyDataset.zip", files =  "./TidyDataSet.csv",
               zip = Sys.getenv("R_ZIPCMD", "zip"), flags = "-r9X")

   zip(zipfile = "./SummarizedTidyDateSet.zip", files = "./SummarizedTidyDateSet.csv"
,
               zip = Sys.getenv("R_ZIPCMD", "zip"), flags = "-r9X")

   file.remove("./TidyDataSet.csv", "./SummarizedTidyDateSet.csv")
```

```
## [1] TRUE TRUE
```

The dataset includes the following files:

- TidyDataSet.csv - Data set with Train & Test merged for Mean and STD measures
- SummarizedTidyDataSet.cvs - the means of every Variable grouped by Subject & Activity
- TidayDatavar.txt - All var names
- CodeBook.md - CodeBook describing the data set, varnames and the transformations required to create this data set

License:

Use of this dataset in publications must be acknowledged by referencing the following publication [1]

[1] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra and Jorge L. Reyes-Ortiz. Human Activity Recognition on Smartphones using a Multiclass Hardware-Friendly Support Vector Machine. International Workshop of Ambient Assisted Living (IWAAL 2012). Vitoria-Gasteiz, Spain. Dec 2012

This dataset is distributed AS-IS and no responsibility implied or explicit can be addressed to the authors or their institutions for its use or misuse. Any commercial use is prohibited.

Jorge L. Reyes-Ortiz, Alessandro Ghio, Luca Oneto, Davide Anguita. November 2012.