**Code Book for the Getting and Cleaning Data project**

This code Book describes a project undertaken for a course in the data science specialization. The data sets used were provided as ".txt" files, therefore the study design described below was made by others[[1]](#footnote-2).

The student just had to figure out how to read, into Rstudio, the provided data sets.

The student certifies that the code Book section is her own work.

***1. Study design***

The 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. Using its embedded accelerometer and gyroscope, we captured 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz. The experiments have been video-recorded to label the data manually. The obtained dataset has been randomly partitioned into two sets, where 70% of the volunteers was selected for generating the training data and 30% the test data.   
  
The sensor signals (accelerometer and gyroscope) were pre-processed by applying noise filters and then sampled in fixed-width sliding windows of 2.56 sec and 50% overlap (128 readings/window). The sensor acceleration signal, which has gravitational and body motion components, was separated using a Butterworth low-pass filter into body acceleration and gravity. The gravitational force is assumed to have only low frequency components, therefore a filter with 0.3 Hz cutoff frequency was used. From each window, a vector of features was obtained by calculating variables from the time and frequency domain.

The 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. Using its embedded accelerometer and gyroscope, we captured 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz. The experiments have been video-recorded to label the data manually. The obtained dataset has been randomly partitioned into two sets, where 70% of the volunteers was selected for generating the training data and 30% the test data.

The sensor signals (accelerometer and gyroscope) were pre-processed by applying noise filters and then sampled in fixed-width sliding windows of 2.56 sec and 50% overlap (128 readings/window). The sensor acceleration signal, which has gravitational and body motion components, was separated using a Butterworth low-pass filter into body acceleration and gravity. The gravitational force is assumed to have only low frequency components, therefore a filter with 0.3 Hz cutoff frequency was used. From each window, a vector of features was obtained by calculating variables from the time and frequency domain. See 'features\_info.txt' for more details.

For each record it is provided:

======================================

- Triaxial acceleration from the accelerometer (total acceleration) and the estimated body acceleration.

- Triaxial Angular velocity from the gyroscope.

- A 561-feature vector with time and frequency domain variables.

- Its activity label.

- An identifier of the subject who carried out the experiment.

The dataset includes the following files:

=========================================

- 'README.txt'

- 'features\_info.txt': Shows information about the variables used on the feature vector.

- 'features.txt': List of all features.

- 'activity\_labels.txt': Links the class labels with their activity name.

- 'train/X\_train.txt': Training set.

- 'train/y\_train.txt': Training labels.

- 'test/X\_test.txt': Test set.

- 'test/y\_test.txt': Test labels.

The following files are available for the train and test data. Their descriptions are equivalent.

- 'train/subject\_train.txt': Each row identifies the subject who performed the activity for each window sample. Its range is from 1 to 30.

- 'train/Inertial Signals/total\_acc\_x\_train.txt': The acceleration signal from the smartphone accelerometer X axis in standard gravity units 'g'. Every row shows a 128 element vector. The same description applies for the 'total\_acc\_x\_train.txt' and 'total\_acc\_z\_train.txt' files for the Y and Z axis.

- 'train/Inertial Signals/body\_acc\_x\_train.txt': The body acceleration signal obtained by subtracting the gravity from the total acceleration.

- 'train/Inertial Signals/body\_gyro\_x\_train.txt': The angular velocity vector measured by the gyroscope for each window sample. The units are radians/second.

Notes:

- Features are normalized and bounded within [-1,1].

- Each feature vector is a row on the text file.

For more information about this dataset contact: activityrecognition@smartlab.ws

***2. Code Book***

As we were provided with 2 data sets (train and test) as ".txt" file, and required to produce a run\_analysis.R script that would load the raw data into RStudio, process it, reshape it and produce a tidy the data set with each numeric variable averaged by subject and activity, the following steps and choices were made:

**2.1. Loading features data**

1. reads into R the data in "features.txt" and stores it into a data.frame named "rawFeatures" with 561 observations of 2 variables

2. renames "rawFeatures" columns into "featureNumber" and "feature"

3. subsets the "feature" column from "rawFeature" data.frame and stores it into factor object named "features"

**2.2. Loading test data**

4. reads into R the data in "X\_test.txt" and stores it into a data.frame named "rawTestset" with 2947 observations of 561 variables

5. renames "rawTestset" columns into corresponding features names using the "features" object

6. reads into R the data in "y\_test.txt" and stores it into a data.frame named "rawTestActivity" with 2947 observations of 1 variable

7. renames "rawTestActivity" column into "Activity"

8. reads into R the data in "subject\_test.txt" and stores it into a data.frame named "rawSubjectsTest" with 2947 observations of 1 variable

9. renames "rawSubjectsTest" column into "SubjectId"

10. creates a data.frame with "SubjectId" and "Activity" columns named "subjectsNtestActivity"

11. creates a "testData" data.frame with columns "SubjectId", "Activity", and the features names as remaining columns i.e 2947 observations of 563 variables

**2.3 Loading train data**

12. reads into R the data in "X\_train.txt" and stores it into a data.frame named "rawTrainingset" with 7352 observations of 561 variables

13. renames "rawTrainingset" columns into corresponding features names using the "features" object

14. reads into R the data in "y\_train.txt" and stores it into a data.frame named "rawTrainingActivity" with 7352 observations of 1 variable

15. renames "rawTrainingActivity" column into "Activity"

16. reads into R the data in "subject\_train.txt" and stores it into a data.frame named "rawSubjectsTrain" with 7352 observations of 1 variable

17. renames "rawSubjectsTrain" column into "SubjectId"

18. creates a data.frame with "SubjectId" and "Activity" columns named "subjectsNtrainActivity"

19. creates a "trainData" data.frame with columns "SubjectId", "Activity", and the features names as remaining columns i.e 7352 observations of 563 variables

**2.4. Merging trainData set with testData set**

20. merges "trainData" and "testData" into "processedDataSet" data.frame of 10299 observations of 563 variables

**2.5. Extracting columns with mean and standard deviation in it**

21. extracts columns with "mean()", "mean()-X/Y/Z and meanFreq()-X/Y/Z from "processedDataSet" and storing them into "proDatasetwithmean" data.frame of 10299 observations of 46 variables

22. installs and loads "dplyr"

23. extracts columns with mean() and mean()-X/Y/Z only from "proDatasetwithmean" and stores them into "proDatasetwithmeanOnly" data.frame of 10299 observations of 33 variables

24. extracts columns with std() and std()-X/Y/Z from "processedDataSet" and storing them into "proDatasetwithstd" data.frame of 10299 observations of 33 variables

25. merges everything into one data.frame named "messyDataSet" data.frame of 10299 observations of 68 variables

**2.6.Renaming Activity values**

26. changes "messyDataSet$Activity" values from "Integer" to "character" type

27. replaces all "messyDataSet$Activity" values into corresponding activity's names

**2.7. Restructuring and aggregating data set**

28. installs and loads "reshape" package

29. restructures dataset in such way that each observation is in its own row and stores it into "tidyDataSet"

30. aggregates the Dataset by subjectId and Activity, computes the mean for each numeric value and stores them into "averagedTidyDataSet"

**2.8. Writing the dataset to a ".txt" file**

31. writes our "averagedTidyDataSet" to a "averagedTidyDataSet.txt" file inside the working directory

**2.9. Printing the first 6 rows of the "averagedTidyDataSet" on the console**

32. prints the first 6 rows of the "averagedTidyDataSet" on the console

1. http://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones [↑](#footnote-ref-2)