**CODE BOOK - Getting and Cleaning Data Course Project**

**11/21/2015**

This code book that describes the variables, the data, and any transformations used to clean up the data supplied for this exercise. The point of the exercise is to demonstrate our ability to collect, work with, and clean a data set. The goal is to prepare tidy data that can be used for later analysis. The requirements are: 1) a tidy data set as described below, 2) a link to a Github repository with a script for performing the analysis, and 3) a code book that describes the variables, the data, and any transformations or work that were performed to clean up the data called CodeBook.md. You should also include a README.md in the repo with your scripts.

**The data**

*Human Activity Recognition Using Smartphones Dataset Version 1.0*

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

*Smartlab - Non Linear Complex Systems Laboratory*

*DITEN - Università degli Studi di Genova.*

*Via Opera Pia 11A, I-16145, Genoa, Italy.*

*activityrecognition@smartlab.ws*

*www.smartlab.ws*

*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.*

**Get and load the data.**

library(reshape2)

rawdata <- tempfile()

download.file("http://d396qusza40orc.cloudfront.net/getdata%2Fprojectfiles%2FUCI%20HAR%20Dataset.zip",rawdata)

unzip(rawdata, list = FALSE)

Y\_test <- read.table(unzip(rawdata, "UCI HAR Dataset/test/y\_test.txt"))

X\_test <- read.table(unzip(rawdata, "UCI HAR Dataset/test/X\_test.txt"))

subject\_test <- read.table(unzip(rawdata, "UCI HAR Dataset/test/subject\_test.txt"))

Y\_train <- read.table(unzip(rawdata, "UCI HAR Dataset/train/y\_train.txt"))

X\_train <- read.table(unzip(rawdata, "UCI HAR Dataset/train/X\_train.txt"))

subject\_train <- read.table(unzip(rawdata, "UCI HAR Dataset/train/subject\_train.txt"))

features <- read.table(unzip(rawdata, "UCI HAR Dataset/features.txt"))

activity\_labels <- read.table(unzip(rawdata,"UCI HAR Dataset/activity\_labels.txt"))

unlink(rawdata) # done with the temp file

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

datapoints <- grep(".\*mean.\*|.\*std.\*", features[,2])

datapoints.names <- features[datapoints,2]

datapoints.names = gsub('-mean', 'Mean', datapoints.names)

datapoints.names = gsub('-std', 'Std', datapoints.names)

datapoints.names <- gsub('[-()]', '', datapoints.names)

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

train <- cbind(X\_train, Y\_train, subject\_train)

test <- cbind(X\_test, Y\_test, subject\_test)

merged <- rbind(train, test)

**Appropriately label the data set with descriptive activity names.**

activity\_labels[,2] <- as.character(activity\_labels[,2])

features[,2] <- as.character(features[,2])

**Create a second, independent tidy data set with the average of each variable for each activity and each subject.**

* Create factors

merged$activity <- factor(merged$activity, levels = activity\_labels[,1], labels = activity\_labels[,2])

merged$subject <- as.factor(merged$subject)

* Get the data properly shaped

merged.melted <- melt(merged, id = c("subject", "activity"))

* Finish the tidying with dcast

merged.mean <- dcast(merged.melted, subject + activity ~ variable, mean)

* Write the text file “tidy.txt”

write.table(merged.mean, "tidy.txt", row.names = FALSE, quote = FALSE)