Codebook_UCI-HAR_Tidy_DB

Leonardo Moreno Naranjo

17/7/2020

Introduction

This document includes a brief data summary from UCI-HAR tidy dataset, that was transformed from the original source UCI HAR Dataset. The data was collected in the context of an experiment about human mobility, measured with accelerometers from the Samsung Galaxy S smartphone.

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.

A full description is available at the site where the data was obtained:

> Human Activity Recognition Using Smartphones Data Set

> 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

Files

- Getting-and-Cleaning-Data-Course-Project.Rproj (R Data Project)
- run_analysis.R (R Script:detailing Data cleaning process)
- README.md (General info)
- UCI HAR Dataset (Folder: Original data source)
- DB_UCI_HAR_Tidy.csv (Tidy Dataset)
- Summary_UCI_HAR_Tidy.csv (Tidy Summary with means grouped by subjects, experimental groups and activities)

Variables

[1] "ID": Subject ID from 1 to 30 (train group) and 1 to 24 (test group)
(*Note:* the id's from each group represent diferent people, even if the number is the same in the 2 groups)
[2] "Group": groups set by train or test group
* 1 = "train"
* 2 = "test"

[3] “activity”: types of activity identified
* 1 = “WALKING”
* 2 = “WALKING_UPSTAIRS”
* 3 = “WALKING_DOWNSTAIRS”
* 4 = “SITTING”
* 5 = “STANDING”
* 6 = “LAYING”

Mobiliy Measures

The mobility variables follow the next code name system:

> The features selected for this database come from the accelerometer and gyroscope 3-axial raw signals time_accelerometer-XYZ and time_Gyroscope-XYZ. These time domain signals (time) were captured at a constant rate of 50 Hz. Then they were filtered using a median filter and a 3rd order low pass Butterworth filter with a corner frequency of 20 Hz to remove noise. Similarly, the acceleration signal was then separated into body and gravity acceleration signals (time_Body_Accelerometer-XYZ and time_Gravity_Accelerometer-XYZ) using another low pass Butterworth filter with a corner frequency of 0.3 Hz. Subsequently, the body linear acceleration and angular velocity were derived in time to obtain Jerk signals (time_Body_AccelerometerJerk-XYZ and time_Body_GyroscopeJerk-XYZ). Also the magnitude of these three-dimensional signals were calculated using the Euclidean norm (time_Body_Accelerometer_Magnitude, time_GravityAccelerometer_Magnitude, time_Body_AccelerometerJerk_Magnitude, time_Body_Gyroscope_Magnitude, time_Body_GyroscopeJerk_Magnitude). Finally a Fast Fourier Transform (FFT) was applied to some of these signals producing frequency_Body_Accelerometer-XYZ, frequency_Body_AccelerometerJerk-XYZ,frequency_Body_Gyroscope-XYZ, frequency_Body_AccelerometerJerk_Magnitude, frequency_Body_Gyroscope_Magnitude, frequency_Body_GyroscopeJerk_Magnitude. These signals were used to estimate variables of the feature vector for each pattern:
> ‘-XYZ’ is used to denote 3-axial signals in the X, Y and Z directions.
>
> Examples:
> [4] “time_Body_Accelerometer-mean(X)”:
> [5] “time_Body_Accelerometer-mean(Y)”
> [6] “time_Body_Accelerometer-mean(Z)” [62] “frequency_Body_Gyroscope-mean(X)”
> [63] “frequency_Body_Gyroscope-mean(Y)”
> [64] “frequency_Body_Gyroscope-mean(Z)”
>
> The set of variables that were estimated from these signals are:
> mean(): Mean value
> std(): Standard deviation

[4] “time_Body_Accelerometer-mean(X)”:
[5] “time_Body_Accelerometer-mean(Y)”
[6] “time_Body_Accelerometer-mean(Z)”
[7] “time_Body_Accelerometer-std(X)”
[8] “time_Body_Accelerometer-std(Y)”
[9] “time_Body_Accelerometer-std(Z)”
[10] “time_Gravity_Accelerometer-mean(X)”
[11] “time_Gravity_Accelerometer-mean(Y)”
[12] “time_Gravity_Accelerometer-mean(Z)”

[13] “time_Gravity_Accelerometer-std(X)”
[14] “time_Gravity_Accelerometer-std(Y)”
[15] “time_Gravity_Accelerometer-std(Z)”
[16] “time_Body_AccelerometerJerk-mean(X)”
[17] “time_Body_AccelerometerJerk-mean(Y)”
[18] “time_Body_AccelerometerJerk-mean(Z)”
[19] “time_Body_AccelerometerJerk-std(X)”
[20] “time_Body_AccelerometerJerk-std(Y)”
[21] “time_Body_AccelerometerJerk-std(Z)”
[22] “time_Body_Gyroscope-mean(X)”
[23] “time_Body_Gyroscope-mean(Y)”
[24] “time_Body_Gyroscope-mean(Z)”
[25] “time_Body_Gyroscope-std(X)”
[26] “time_Body_Gyroscope-std(Y)”
[27] “time_Body_Gyroscope-std(Z)”
[28] “time_Body_GyroscopeJerk-mean(X)”
[29] “time_Body_GyroscopeJerk-mean(Y)”
[30] “time_Body_GyroscopeJerk-mean(Z)”
[31] “time_Body_GyroscopeJerk-std(X)”
[32] “time_Body_GyroscopeJerk-std(Y)”
[33] “time_Body_GyroscopeJerk-std(Z)”
[34] “time_Body_Accelerometer_Magnitude-mean()”
[35] “time_Body_Accelerometer_Magnitude-std()”
[36] “time_Gravity_Accelerometer_Magnitude-mean()”
[37] “time_Gravity_Accelerometer_Magnitude-std()”
[38] “time_Body_AccelerometerJerk_Magnitude-mean()”
[39] “time_Body_AccelerometerJerk_Magnitude-std()”
[40] “time_Body_Gyroscope_Magnitude-mean()”
[41] “time_Body_Gyroscope_Magnitude-std()”
[42] “time_Body_GyroscopeJerk_Magnitude-mean()”
[43] “time_Body_GyroscopeJerk_Magnitude-std()”
[44] “frequency_Body_Accelerometer-mean(X)”
[45] “frequency_Body_Accelerometer-mean(Y)”
[46] “frequency_Body_Accelerometer-mean(Z)”
[47] “frequency_Body_Accelerometer-std(X)”
[48] “frequency_Body_Accelerometer-std(Y)”
[49] “frequency_Body_Accelerometer-std(Z)”
[50] “frequency_Body_Accelerometer-meanFreq(X)”
[51] “frequency_Body_Accelerometer-meanFreq(Y)”
[52] “frequency_Body_Accelerometer-meanFreq(Z)”
[53] “frequency_Body_AccelerometerJerk-mean(X)”
[54] “frequency_Body_AccelerometerJerk-mean(Y)”
[55] “frequency_Body_AccelerometerJerk-mean(Z)”
[56] “frequency_Body_AccelerometerJerk-std(X)”
[57] “frequency_Body_AccelerometerJerk-std(Y)”
[58] “frequency_Body_AccelerometerJerk-std(Z)”
[59] “frequency_Body_AccelerometerJerk-meanFreq(X)”
[60] “frequency_Body_AccelerometerJerk-meanFreq(Y)”
[61] “frequency_Body_AccelerometerJerk-meanFreq(Z)”
[62] “frequency_Body_Gyroscope-mean(X)”
[63] “frequency_Body_Gyroscope-mean(Y)”
[64] “frequency_Body_Gyroscope-mean(Z)”
[65] “frequency_Body_Gyroscope-std(X)”
[66] “frequency_Body_Gyroscope-std(Y)”

[67] "frequency_Body_Gyroscope-std(Z)"
[68] "frequency_Body_Gyroscope-meanFreq(X)"
[69] "frequency_Body_Gyroscope-meanFreq(Y)"
[70] "frequency_Body_Gyroscope-meanFreq(Z)"
[71] "frequency_Body_Accelerometer_Magnitude-mean()"
[72] "frequency_Body_Accelerometer_Magnitude-std()"
[73] "frequency_Body_Accelerometer_Magnitude-meanFreq()"
[74] "frequency_Body___AccelerometerJerk_Magnitude-mean()"
[75] "frequency_Body___AccelerometerJerk_Magnitude-std()"
[76] "frequency_Body___AccelerometerJerk_Magnitude-meanFreq()"
[77] "frequency_Body___Gyroscope_Magnitude-mean()"
[78] "frequency_Body___Gyroscope_Magnitude-std()"
[79] "frequency_Body___Gyroscope_Magnitude-meanFreq()"
[80] "frequency_Body___GyroscopeJerk_Magnitude-mean()"
[81] "frequency_Body___GyroscopeJerk_Magnitude-std()"
[82] "frequency_Body___GyroscopeJerk_Magnitude-meanFreq()"

Thanks for review me

This project was developed in the context of the course "Getting and Data Cleaning" from Coursera