**Introduction**

The script “run\_analysis.R” executes the 5 steps outlined in the Getting Data and Cleaning project's directions. The five steps are:

1. Merge the training and the test sets to create one data set.
2. Extract 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 variable names.
5. From the data set in step 4, creates a second, independent tidy data set with the average of each variable for each activity and each subject.

**Files Used**

***Training files (X\_train.txt, y\_train.txt, and subject\_train.txt)***

X\_train.txt: Contains 561 Variables with measurements for feature readings. Note: This file does not contain a header row.

y\_train.txt: Contains 1 column activityId

subject\_train.txt: Contains 1 column subjectId

***Test files (X\_test.txt, y\_test.txt, and subject\_test.txt)***

X\_test.txt: Contains 561 Variables with measurements for readings. Note: This file does not contain a header row.

y\_test.txt: Contains 1 column activityId

subject\_test.txt: Contains 1 column subjectId

***Features files (features.txt)***

Contain two columns: V1 – Feature Id and V2 Feature Description

***Activity labels (activity\_labels.txt)***

Contain two columns: activityId – Activity Id and activityType – Description of the activity

(values: 1 – Walking, 2 – Walking\_Upstairs, 3 – Walking\_downstairs, 4 – Sitting, 5 – Standing, 6 – Laying)

Provides the lookup values to link y\_train:activityID & y\_test:ActivityId

Object Names

X\_train – Creates a vector of training measurements

y\_train – Creates a vector of Activity Ids for each observation of the X\_train measurements

sub\_train - Creates a vector of subject Ids for each observation of the X\_train measurements

X\_test – Creates a vector of test measurements

y\_test – Creates a vector of Activity Ids for each observation of the X\_test measurements

sub\_test - Creates a vector of subject Ids for each observation of the X\_test measurements

features – Creates a vector of Measurement Ids & measurement descriptive names .

act\_labels - Creates a vector of Activity Ids & Activity Descriptive names

bind\_train – Combines the columns of x\_train, y\_train, and sub\_train using the cbind function to create a training data set.

bind\_test– Combines the columns of X\_test, y\_test, and sub\_test using the cbind function to create a test data set.

alldata – Combines the training data set (bind\_train) and test data set (bind\_test) using the rbind function to create a master data set of all observations.

remove\_dup\_col\_names - Removes duplicate column names. This step was required as a result of duplicate columns detected while select using dplyr when creating the mean\_std data set.

mean\_std – Creates a data set with only mean and standard deviation measures for each measurement.

mean\_std\_add\_act\_names – Add activity names to mean\_std data set using merge. Activity names are appended to the end of the data set.

tidy – Creates the tidy data set.

**Feature Descriptions**

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

The features selected for this database come from the accelerometer and gyroscope 3-axial raw signals tAcc-XYZ and tGyro-XYZ. These time domain signals (prefix 't' to denote 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 (tBodyAcc-XYZ and tGravityAcc-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 (tBodyAccJerk-XYZ and tBodyGyroJerk-XYZ). Also the magnitude of these three-dimensional signals were calculated using the Euclidean norm (tBodyAccMag, tGravityAccMag, tBodyAccJerkMag, tBodyGyroMag, tBodyGyroJerkMag).

Finally a Fast Fourier Transform (FFT) was applied to some of these signals producing fBodyAcc-XYZ, fBodyAccJerk-XYZ, fBodyGyro-XYZ, fBodyAccJerkMag, fBodyGyroMag, fBodyGyroJerkMag. (Note the 'f' to indicate frequency domain signals).

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.

tBodyAcc-XYZ

tGravityAcc-XYZ

tBodyAccJerk-XYZ

tBodyGyro-XYZ

tBodyGyroJerk-XYZ

tBodyAccMag

tGravityAccMag

tBodyAccJerkMag

tBodyGyroMag

tBodyGyroJerkMag

fBodyAcc-XYZ

fBodyAccJerk-XYZ

fBodyGyro-XYZ

fBodyAccMag

fBodyAccJerkMag

fBodyGyroMag

fBodyGyroJerkMag

The set of variables that were estimated from these signals are:

mean(): Mean value

std(): Standard deviation

mad(): Median absolute deviation

max(): Largest value in array

min(): Smallest value in array

sma(): Signal magnitude area

energy(): Energy measure. Sum of the squares divided by the number of values.

iqr(): Interquartile range

entropy(): Signal entropy

arCoeff(): Autorregresion coefficients with Burg order equal to 4

correlation(): correlation coefficient between two signals

maxInds(): index of the frequency component with largest magnitude

meanFreq(): Weighted average of the frequency components to obtain a mean frequency

skewness(): skewness of the frequency domain signal

kurtosis(): kurtosis of the frequency domain signal

bandsEnergy(): Energy of a frequency interval within the 64 bins of the FFT of each window.

angle(): Angle between to vectors.