

# Codebook

Tobi

12/13/2021

##The run\_analysis.R script performs the data preparation and then followed by the 5 steps required as described in the course project's definition.

##Download the dataset Dataset downloaded and extracted under the folder called UCI HAR Dataset

```
##Assign each data to variables features <- features.txt activities <- activity_labels.txt subject_test <- test/subject_test.txt x_test <- test/X_test.txt y_test <- test/y_test.txt subject_train <- test/subject_train.txt x_train <- test/X_train.txt y_train <- test/y_train.txt
```

##Merges the training and the test sets to create one data set X is created by merging x\_train and x\_test using rbind() function Y is created by merging y\_train and y\_test using rbind() function Subject is created by merging subject\_train and subject\_test using rbind() function Merged\_Data is created by merging Subject, Y and X using cbind() function

##Extract only the measurements on the mean and standard deviation for each measurement TidyData is created by subsetting Merged\_Data, selecting only columns: subject, code and the measurements on the mean and standard deviation (std) for each measurement

##Uses descriptive activity names to name the activities in the data set Entire numbers in code column of the TidyData replaced with corresponding activity taken from second column of the activity's variable

##Appropriately labels the data set with descriptive variable names code column in TidyData renamed into activities All Acc in column's name replaced by Accelerometer All Gyro in column's name replaced by Gyroscope All BodyBody in column's name replaced by Body All Mag in column's name replaced by Magnitude All start with character f in column's name replaced by Frequency All start with character t in column's name replaced by Time

##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 FinalData is created by summarizing TidyData taking the means of each variable for each activity and each subject, after grouped by subject and activity.

```
library(dplyr)
```

```
##
```

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

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
## filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
## intersect, setdiff, setequal, union
```

```

# Checking if archive already exists
fileused<- "Coursera_GCD_Project.zip"
if(!file.exists("./data")){dir.create("./data")}
fileUrl<-"https://d396qusza40orc.cloudfront.net/getdata%2Fprojectfiles%2FUCI%20HAR%20Dataset.zip"
download.file(fileUrl, fileused)

# Checking if folder exists
if (!file.exists("UCI HAR Dataset")) {
  unzip(fileused)
}

features <- read.table("UCI HAR Dataset/features.txt", col.names = c("n","functions"))
activities <- read.table("UCI HAR Dataset/activity_labels.txt", col.names = c("code", "activity"))
subject_test <- read.table("UCI HAR Dataset/test/subject_test.txt", col.names = "subject")
x_test <- read.table("UCI HAR Dataset/test/X_test.txt", col.names = features$functions)
y_test <- read.table("UCI HAR Dataset/test/y_test.txt", col.names = "code")
subject_train <- read.table("UCI HAR Dataset/train/subject_train.txt", col.names = "subject")
x_train <- read.table("UCI HAR Dataset/train/X_train.txt", col.names = features$functions)
y_train <- read.table("UCI HAR Dataset/train/y_train.txt", col.names = "code")
xtable <- rbind(x_train, x_test)
ytable <- rbind(y_train, y_test)
Subject <- rbind(subject_train, subject_test)
Merged_Data <- cbind(Subject, ytable, xtable)

NewData <- Merged_Data %>% select(subject, code, contains("mean"), contains("std"))

NewData$code <- activities[NewData$code, 2]

names(NewData)[2] = "activity"
names(NewData)<-gsub("Acc", "Accelerometer", names(NewData))
names(NewData)<-gsub("Gyro", "Gyroscope", names(NewData))
names(NewData)<-gsub("BodyBody", "Body", names(NewData))
names(NewData)<-gsub("Mag", "Magnitude", names(NewData))
names(NewData)<-gsub("^t", "Time", names(NewData))
names(NewData)<-gsub("^f", "Frequency", names(NewData))
names(NewData)<-gsub("tBody", "TimeBody", names(NewData))
names(NewData)<-gsub("-mean()", "Mean", names(NewData), ignore.case = TRUE)
names(NewData)<-gsub("-std()", "STD", names(NewData), ignore.case = TRUE)
names(NewData)<-gsub("-freq()", "Frequency", names(NewData), ignore.case = TRUE)
names(NewData)<-gsub("angle", "Angle", names(NewData))
names(NewData)<-gsub("gravity", "Gravity", names(NewData))

FinalData <- NewData %>%
  group_by(subject, activity) %>%
  summarise_all(funs(mean))

```

```

## Warning: 'funs()' was deprecated in dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with 'tibble::lst()':

```

```
## tibble::lst(mean, median)
##
## # Using lambdas
## list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was generated.
```

```
write.table(FinalData, "FinalData.txt", row.name=FALSE)
```

```
str(FinalData)
```

```
## grouped_df [180 x 88] (S3: grouped_df/tbl_df/tbl/data.frame)
```

```
## $ subject : int [1:180] 1 1 1 1 1 1 2 2 2 2 ...
## $ activity : chr [1:180] "LAYING" "SITTING" "STANDING" "WALKING" ...
## $ TimeBodyAccelerometer.mean...X : num [1:180] 0.222 0.261 0.279 0.277 0.289 ...
## $ TimeBodyAccelerometer.mean...Y : num [1:180] -0.04051 -0.00131 -0.01614 -0.0177 ...
## $ TimeBodyAccelerometer.mean...Z : num [1:180] -0.113 -0.105 -0.111 -0.111 -0.108 ...
## $ TimeGravityAccelerometer.mean...X : num [1:180] -0.249 0.832 0.943 0.935 0.932 ...
## $ TimeGravityAccelerometer.mean...Y : num [1:180] 0.706 0.204 -0.273 -0.282 -0.267 ...
## $ TimeGravityAccelerometer.mean...Z : num [1:180] 0.4458 0.332 0.0135 -0.0681 -0.0681 ...
## $ TimeBodyAccelerometerJerk.mean...X : num [1:180] 0.0811 0.0775 0.0754 0.074 0.0542 ...
## $ TimeBodyAccelerometerJerk.mean...Y : num [1:180] 0.003838 -0.000619 0.007976 0.0281 ...
## $ TimeBodyAccelerometerJerk.mean...Z : num [1:180] 0.01083 -0.00337 -0.00369 -0.0041 ...
## $ TimeBodyGyroscope.mean...X : num [1:180] -0.0166 -0.0454 -0.024 -0.0418 -0.0418 ...
## $ TimeBodyGyroscope.mean...Y : num [1:180] -0.0645 -0.0919 -0.0594 -0.0695 -0.0695 ...
## $ TimeBodyGyroscope.mean...Z : num [1:180] 0.1487 0.0629 0.0748 0.0849 0.0901 ...
## $ TimeBodyGyroscopeJerk.mean...X : num [1:180] -0.1073 -0.0937 -0.0996 -0.09 -0.09 ...
## $ TimeBodyGyroscopeJerk.mean...Y : num [1:180] -0.0415 -0.0402 -0.0441 -0.0398 -0.0398 ...
## $ TimeBodyGyroscopeJerk.mean...Z : num [1:180] -0.0741 -0.0467 -0.049 -0.0461 -0.0461 ...
## $ TimeBodyAccelerometerMagnitude.mean... : num [1:180] -0.8419 -0.9485 -0.9843 -0.137 0.137 ...
## $ TimeGravityAccelerometerMagnitude.mean... : num [1:180] -0.8419 -0.9485 -0.9843 -0.137 0.137 ...
## $ TimeBodyAccelerometerJerkMagnitude.mean... : num [1:180] -0.9544 -0.9874 -0.9924 -0.1414 -0.1414 ...
## $ TimeBodyGyroscopeJerkMagnitude.mean... : num [1:180] -0.8748 -0.9309 -0.9765 -0.161 -0.161 ...
## $ TimeBodyGyroscopeJerkMagnitude.mean... : num [1:180] -0.963 -0.992 -0.995 -0.299 -0.299 ...
## $ FrequencyBodyAccelerometer.mean...X : num [1:180] -0.9391 -0.9796 -0.9952 -0.2028 0.2028 ...
## $ FrequencyBodyAccelerometer.mean...Y : num [1:180] -0.86707 -0.94408 -0.97707 0.0897 ...
## $ FrequencyBodyAccelerometer.mean...Z : num [1:180] -0.883 -0.959 -0.985 -0.332 -0.222 ...
## $ FrequencyBodyAccelerometer.meanFreq...X : num [1:180] -0.1588 -0.0495 0.0865 -0.2075 -0.2075 ...
## $ FrequencyBodyAccelerometer.meanFreq...Y : num [1:180] 0.0975 0.0759 0.1175 0.1131 0.0631 ...
## $ FrequencyBodyAccelerometer.meanFreq...Z : num [1:180] 0.0894 0.2388 0.2449 0.0497 0.294 ...
## $ FrequencyBodyAccelerometerJerk.mean...X : num [1:180] -0.9571 -0.9866 -0.9946 -0.1705 -0.1705 ...
## $ FrequencyBodyAccelerometerJerk.mean...Y : num [1:180] -0.9225 -0.9816 -0.9854 -0.0352 -0.0352 ...
## $ FrequencyBodyAccelerometerJerk.mean...Z : num [1:180] -0.948 -0.986 -0.991 -0.469 -0.288 ...
## $ FrequencyBodyAccelerometerJerk.meanFreq...X : num [1:180] 0.132 0.257 0.314 -0.209 -0.253 ...
## $ FrequencyBodyAccelerometerJerk.meanFreq...Y : num [1:180] 0.0245 0.0475 0.0392 -0.3862 -0.3862 ...
## $ FrequencyBodyAccelerometerJerk.meanFreq...Z : num [1:180] 0.02439 0.09239 0.13858 -0.18553 0.18553 ...
## $ FrequencyBodyGyroscope.mean...X : num [1:180] -0.85 -0.976 -0.986 -0.339 -0.352 ...
## $ FrequencyBodyGyroscope.mean...Y : num [1:180] -0.9522 -0.9758 -0.989 -0.1031 -0.1031 ...
## $ FrequencyBodyGyroscope.mean...Z : num [1:180] -0.9093 -0.9513 -0.9808 -0.2559 -0.2559 ...
## $ FrequencyBodyGyroscope.meanFreq...X : num [1:180] -0.00355 0.18915 -0.12029 0.01478 ...
## $ FrequencyBodyGyroscope.meanFreq...Y : num [1:180] -0.0915 0.0631 -0.0447 -0.0658 0.0658 ...
## $ FrequencyBodyGyroscope.meanFreq...Z : num [1:180] 0.010458 -0.029784 0.100608 0.0007 ...
## $ FrequencyBodyAccelerometerMagnitude.mean... : num [1:180] -0.8618 -0.9478 -0.9854 -0.1286 0.1286 ...
## $ FrequencyBodyAccelerometerMagnitude.meanFreq... : num [1:180] 0.0864 0.2367 0.2846 0.1906 0.1196 ...
```

```

## $ FrequencyBodyAccelerometerJerkMagnitude.mean.. : num [1:180] -0.9333 -0.9853 -0.9925 -0.0571 0
## $ FrequencyBodyAccelerometerJerkMagnitude.meanFreq.. : num [1:180] 0.2664 0.3519 0.4222 0.0938 0.0761
## $ FrequencyBodyGyroscopeMagnitude.mean.. : num [1:180] -0.862 -0.958 -0.985 -0.199 -0.181
## $ FrequencyBodyGyroscopeMagnitude.meanFreq.. : num [1:180] -0.139775 -0.000262 -0.028606 0.2
## $ FrequencyBodyGyroscopeJerkMagnitude.mean.. : num [1:180] -0.942 -0.99 -0.995 -0.319 -0.282
## $ FrequencyBodyGyroscopeJerkMagnitude.meanFreq.. : num [1:180] 0.176 0.185 0.334 0.191 0.19 ...
## $ Angle.TimeBodyAccelerometerMean.Gravity. : num [1:180] 0.021366 0.027442 -0.000222 0.060
## $ Angle.TimeBodyAccelerometerJerkMean..GravityMean. : num [1:180] 0.00306 0.02971 0.02196 -0.00793
## $ Angle.TimeBodyGyroscopeMean.GravityMean. : num [1:180] -0.00167 0.0677 -0.03379 0.01306
## $ Angle.TimeBodyGyroscopeJerkMean.GravityMean. : num [1:180] 0.0844 -0.0649 -0.0279 -0.0187 -0
## $ Angle.X.GravityMean. : num [1:180] 0.427 -0.591 -0.743 -0.729 -0.744
## $ Angle.Y.GravityMean. : num [1:180] -0.5203 -0.0605 0.2702 0.277 0.26
## $ Angle.Z.GravityMean. : num [1:180] -0.3524 -0.218 0.0123 0.0689 0.06
## $ TimeBodyAccelerometer.std...X : num [1:180] -0.928 -0.977 -0.996 -0.284 0.03
## $ TimeBodyAccelerometer.std...Y : num [1:180] -0.8368 -0.9226 -0.9732 0.1145 -0
## $ TimeBodyAccelerometer.std...Z : num [1:180] -0.826 -0.94 -0.98 -0.26 -0.23 ..
## $ TimeGravityAccelerometer.std...X : num [1:180] -0.897 -0.968 -0.994 -0.977 -0.95
## $ TimeGravityAccelerometer.std...Y : num [1:180] -0.908 -0.936 -0.981 -0.971 -0.93
## $ TimeGravityAccelerometer.std...Z : num [1:180] -0.852 -0.949 -0.976 -0.948 -0.89
## $ TimeBodyAccelerometerJerk.std...X : num [1:180] -0.9585 -0.9864 -0.9946 -0.1136 -0
## $ TimeBodyAccelerometerJerk.std...Y : num [1:180] -0.924 -0.981 -0.986 0.067 -0.102
## $ TimeBodyAccelerometerJerk.std...Z : num [1:180] -0.955 -0.988 -0.992 -0.503 -0.34
## $ TimeBodyGyroscope.std...X : num [1:180] -0.874 -0.977 -0.987 -0.474 -0.45
## $ TimeBodyGyroscope.std...Y : num [1:180] -0.9511 -0.9665 -0.9877 -0.0546 -0
## $ TimeBodyGyroscope.std...Z : num [1:180] -0.908 -0.941 -0.981 -0.344 -0.12
## $ TimeBodyGyroscopeJerk.std...X : num [1:180] -0.919 -0.992 -0.993 -0.207 -0.48
## $ TimeBodyGyroscopeJerk.std...Y : num [1:180] -0.968 -0.99 -0.995 -0.304 -0.239
## $ TimeBodyGyroscopeJerk.std...Z : num [1:180] -0.958 -0.988 -0.992 -0.404 -0.26
## $ TimeBodyAccelerometerMagnitude.std.. : num [1:180] -0.7951 -0.9271 -0.9819 -0.2197 0
## $ TimeGravityAccelerometerMagnitude.std.. : num [1:180] -0.7951 -0.9271 -0.9819 -0.2197 0
## $ TimeBodyAccelerometerJerkMagnitude.std.. : num [1:180] -0.9282 -0.9841 -0.9931 -0.0745 -0
## $ TimeBodyGyroscopeMagnitude.std.. : num [1:180] -0.819 -0.935 -0.979 -0.187 -0.22
## $ TimeBodyGyroscopeJerkMagnitude.std.. : num [1:180] -0.936 -0.988 -0.995 -0.325 -0.30
## $ FrequencyBodyAccelerometer.std...X : num [1:180] -0.9244 -0.9764 -0.996 -0.3191 0
## $ FrequencyBodyAccelerometer.std...Y : num [1:180] -0.834 -0.917 -0.972 0.056 -0.113
## $ FrequencyBodyAccelerometer.std...Z : num [1:180] -0.813 -0.934 -0.978 -0.28 -0.298
## $ FrequencyBodyAccelerometerJerk.std...X : num [1:180] -0.9642 -0.9875 -0.9951 -0.1336 -0
## $ FrequencyBodyAccelerometerJerk.std...Y : num [1:180] -0.932 -0.983 -0.987 0.107 -0.135
## $ FrequencyBodyAccelerometerJerk.std...Z : num [1:180] -0.961 -0.988 -0.992 -0.535 -0.40
## $ FrequencyBodyGyroscope.std...X : num [1:180] -0.882 -0.978 -0.987 -0.517 -0.49
## $ FrequencyBodyGyroscope.std...Y : num [1:180] -0.9512 -0.9623 -0.9871 -0.0335 -0
## $ FrequencyBodyGyroscope.std...Z : num [1:180] -0.917 -0.944 -0.982 -0.437 -0.23
## $ FrequencyBodyAccelerometerMagnitude.std.. : num [1:180] -0.798 -0.928 -0.982 -0.398 -0.18
## $ FrequencyBodyAccelerometerJerkMagnitude.std.. : num [1:180] -0.922 -0.982 -0.993 -0.103 -0.10
## $ FrequencyBodyGyroscopeMagnitude.std.. : num [1:180] -0.824 -0.932 -0.978 -0.321 -0.39
## $ FrequencyBodyGyroscopeJerkMagnitude.std.. : num [1:180] -0.933 -0.987 -0.995 -0.382 -0.39
## - attr(*, "groups")= tibble [30 x 2] (S3: tbl_df/tbl/data.frame)
## ..$ subject: int [1:30] 1 2 3 4 5 6 7 8 9 10 ...
## ..$ .rows : list<int> [1:30]
## .. ..$ : int [1:6] 1 2 3 4 5 6
## .. ..$ : int [1:6] 7 8 9 10 11 12
## .. ..$ : int [1:6] 13 14 15 16 17 18
## .. ..$ : int [1:6] 19 20 21 22 23 24
## .. ..$ : int [1:6] 25 26 27 28 29 30

```

```
## ..$ : int [1:6] 31 32 33 34 35 36
## ..$ : int [1:6] 37 38 39 40 41 42
## ..$ : int [1:6] 43 44 45 46 47 48
## ..$ : int [1:6] 49 50 51 52 53 54
## ..$ : int [1:6] 55 56 57 58 59 60
## ..$ : int [1:6] 61 62 63 64 65 66
## ..$ : int [1:6] 67 68 69 70 71 72
## ..$ : int [1:6] 73 74 75 76 77 78
## ..$ : int [1:6] 79 80 81 82 83 84
## ..$ : int [1:6] 85 86 87 88 89 90
## ..$ : int [1:6] 91 92 93 94 95 96
## ..$ : int [1:6] 97 98 99 100 101 102
## ..$ : int [1:6] 103 104 105 106 107 108
## ..$ : int [1:6] 109 110 111 112 113 114
## ..$ : int [1:6] 115 116 117 118 119 120
## ..$ : int [1:6] 121 122 123 124 125 126
## ..$ : int [1:6] 127 128 129 130 131 132
## ..$ : int [1:6] 133 134 135 136 137 138
## ..$ : int [1:6] 139 140 141 142 143 144
## ..$ : int [1:6] 145 146 147 148 149 150
## ..$ : int [1:6] 151 152 153 154 155 156
## ..$ : int [1:6] 157 158 159 160 161 162
## ..$ : int [1:6] 163 164 165 166 167 168
## ..$ : int [1:6] 169 170 171 172 173 174
## ..$ : int [1:6] 175 176 177 178 179 180
## ..@ ptype: int(0)
## ..- attr(*, ".drop")= logi TRUE
```

FinalData

```
## # A tibble: 180 x 88
## # Groups:   subject [30]
##   subject activity      TimeBodyAcceler~ TimeBodyAcceler~ TimeBodyAcceler~
##   <int> <chr>          <dbl>          <dbl>          <dbl>
## 1     1 LAYING          0.222          -0.0405         -0.113
## 2     1 SITTING         0.261          -0.00131        -0.105
## 3     1 STANDING         0.279          -0.0161         -0.111
## 4     1 WALKING          0.277          -0.0174         -0.111
## 5     1 WALKING_DOWNSTAIRS 0.289          -0.00992        -0.108
## 6     1 WALKING_UPSTAIRS    0.255          -0.0240        -0.0973
## 7     2 LAYING          0.281          -0.0182         -0.107
## 8     2 SITTING         0.277          -0.0157         -0.109
## 9     2 STANDING         0.278          -0.0184         -0.106
## 10    2 WALKING          0.276          -0.0186         -0.106
## # ... with 170 more rows, and 83 more variables:
## #   TimeGravityAccelerometer.mean...X <dbl>,
## #   TimeGravityAccelerometer.mean...Y <dbl>,
## #   TimeGravityAccelerometer.mean...Z <dbl>,
## #   TimeBodyAccelerometerJerk.mean...X <dbl>,
## #   TimeBodyAccelerometerJerk.mean...Y <dbl>,
## #   TimeBodyAccelerometerJerk.mean...Z <dbl>, ...
```