README

James Watt

11/1/2020

This is an R script that cleans up the mobile computing dataset from the UCI machine learning repository. It produces two tidy datsets: one LONG dataset, where each row includes subject and activity, and a single measurement value; and a WIDE dataset, where each row contains subject, activity, and each measurement value.

The measurement values included in the final data sets are explained in the codebook, but, in brief, are all mean calculations of repeated measures of that variable within a subject and activity.

To start, the 'test' and 'train' data sets are merged together.

The code below installs the necessary packages.

```
install.packages("plyr")
install.packages("dplyr")
install.packages("reshape2")
install.packages("magrittr")
```

```
library(plyr)
library(dplyr)
library(reshape2)
library(magrittr)
```

This code imports activity labels from 'activity_labels.txt' and the list of features (variables) from 'features.txt'. 'feat_list' is a vector of feature names.

```
activity_labels <- read.table("./UCI HAR Dataset/activity_labels.txt")
features <- read.table("./UCI HAR Dataset/features.txt")
feat_list <- features[,2]
```

```
feat_list[1:15]
```

```
##  [1] "tBodyAcc-mean()-X" "tBodyAcc-mean()-Y" "tBodyAcc-mean()-Z"
##  [4] "tBodyAcc-std()-X"  "tBodyAcc-std()-Y"  "tBodyAcc-std()-Z"
##  [7] "tBodyAcc-mad()-X"  "tBodyAcc-mad()-Y"  "tBodyAcc-mad()-Z"
## [10] "tBodyAcc-max()-X"  "tBodyAcc-max()-Y"  "tBodyAcc-max()-Z"
## [13] "tBodyAcc-min()-X"  "tBodyAcc-min()-Y"  "tBodyAcc-min()-Z"
```

This code merges 'subject_test.txt' and 'y_test.txt', which assigns each activity to each subject. It also reads in the 'X_test.txt' file, which is a list of values of feature results.

```
X_test <- read.table("./UCI HAR Dataset/test/X_test.txt")
subject_test <- read.table("./UCI HAR Dataset/test/subject_test.txt")
y_test <- read.table("./UCI HAR Dataset/test/y_test.txt")

test_merge <- data.frame("subject" = subject_test[,1],
                         "activities" = y_test[,1])
```

Then, I add the 'feat_list' vector to 'X_test' to label the columns of 'X_test' with their variable names.

```
colnames(X_test) <- feat_list
```

Finally, I merge the subjects, activity codes, and X_test results together by columns using **cbind**.

```
test_full <- cbind(test_merge, X_test)
```

This process is repeated for the 'train' data sets.

```
X_train <- read.table("./UCI HAR Dataset/train/X_train.txt")
subject_train <- read.table("./UCI HAR Dataset/train/subject_train.txt")
y_train <- read.table("./UCI HAR Dataset/train/y_train.txt")

train_merge <- data.frame("subject" = subject_train[,1],
                          "activities" = y_train[,1])

colnames(X_train) <- feat_list
train_full <- cbind(train_merge, X_train)
```

To complete Objective #1, the 'test' and 'train' datasets are joined using **rbind** to produce a full data set with subject, activity code, and feature values.

```
full <- rbind(train_full, test_full)
```

To simplify future analyses, only the features which are means or standard deviations (std) are extracted from the 'full' dataset into a data frame called 'subset'. This is accomplished using the **grepl** function. The resulting extracted data are re-merged with datasets containing subject and activity into a new dataframe called 'subset2'. This completes Objective #2.

```
subset <- full[,grepl("mean()", names(full))|grepl("std()", names(full))]
subj_activity <- rbind(train_merge, test_merge)
subset2 <- cbind(subj_activity, subset)
```

The activity codes are listed as numerals 1-6. To replace these with corresponding descriptive activities, the column names of 'activity_labels' (see above) are replaced with "activities" and "activity": the number and description, respectively. This facilitates merging using **left_join** by the "activities" value, which exists in both data sets. For clarity, the new column, "activity", is moved adjacent to the exisiting "activities" column in the new data frame, 'subset3'. This completes Objective #3.

```
colnames(activity_labels) <- c("activities", "activity")
subset3 <- left_join(subset2, activity_labels, by = "activities")
subset3 <- relocate(subset3, activity, .after = activities)
```

Next, each of the feature names is replaced with a more descriptive feature name, using information from the 'features_info.txt' file, provided by the data source. A vector is created of the column names from 'subset3', named 'features'. The first three column names ("subject", "activities", and "activity") are already descriptive, so they are separated out into a vector 'featurestop'. The remaining features are separated into a vector named 'featuresbot'. 'featuresbot' uses piping to replace specific text strings with descriptive strings using **gsub()**. Each element of the vector is made lowercase by **tolower()**. Because each of the final values will be calculated mean values (below), the phrase "mean of" is appended to the front of each variable name in 'featuresbot'. 'features' is then updated by pasting 'featurestop' and 'featuresbot' together using **paste()**. Last, the column names of 'subset3' are updated to those created in 'features' using **colnames()**. This completes Objective #4.

```
features <- names(subset3)
featurestop <- features[1:3]
featuresbot <- features[4:length(features)]
featuresbot %<>%
```

```
        gsub("Acc", "accelerometer ", .) %>%
        gsub("Gyro", "gyroscope ", .) %>%
        gsub("Jerk", "jerksignal ", .) %>%
        gsub("Mag", "magnitude ", .) %>%
        gsub("^tBody", "time domain body ", .) %>%
        gsub("^fBody", "frequency domain body ", .) %>%
        gsub("Body", "", .) %>%
        gsub("^tGravity", "time domain gravity ", .) %>%
        gsub("-mean", "mean", .) %>%
        gsub("-std", "std", .) %>%
        gsub("\\()", "", .) %>%
        gsub("\\-", " ", .) %>%
        gsub("meanFreq", "mean frequency", .) %>%
        tolower() %>%
        paste("mean of", ., sep = " ")

features <- c(featurestop, featuresbot)
colnames(subset3) <- features
```

```
features
```

```
##  [1] "subject"
##  [2] "activities"
##  [3] "activity"
##  [4] "mean of time domain body accelerometer mean x"
##  [5] "mean of time domain body accelerometer mean y"
##  [6] "mean of time domain body accelerometer mean z"
##  [7] "mean of time domain body accelerometer std x"
##  [8] "mean of time domain body accelerometer std y"
##  [9] "mean of time domain body accelerometer std z"
## [10] "mean of time domain gravity accelerometer mean x"
## [11] "mean of time domain gravity accelerometer mean y"
## [12] "mean of time domain gravity accelerometer mean z"
## [13] "mean of time domain gravity accelerometer std x"
## [14] "mean of time domain gravity accelerometer std y"
## [15] "mean of time domain gravity accelerometer std z"
## [16] "mean of time domain body accelerometer jerksignal mean x"
## [17] "mean of time domain body accelerometer jerksignal mean y"
## [18] "mean of time domain body accelerometer jerksignal mean z"
## [19] "mean of time domain body accelerometer jerksignal std x"
## [20] "mean of time domain body accelerometer jerksignal std y"
## [21] "mean of time domain body accelerometer jerksignal std z"
## [22] "mean of time domain body gyroscope mean x"
## [23] "mean of time domain body gyroscope mean y"
## [24] "mean of time domain body gyroscope mean z"
## [25] "mean of time domain body gyroscope std x"
## [26] "mean of time domain body gyroscope std y"
## [27] "mean of time domain body gyroscope std z"
## [28] "mean of time domain body gyroscope jerksignal mean x"
## [29] "mean of time domain body gyroscope jerksignal mean y"
## [30] "mean of time domain body gyroscope jerksignal mean z"
## [31] "mean of time domain body gyroscope jerksignal std x"
## [32] "mean of time domain body gyroscope jerksignal std y"
## [33] "mean of time domain body gyroscope jerksignal std z"
```

```
## [34] "mean of time domain body accelerometer magnitude mean"
## [35] "mean of time domain body accelerometer magnitude std"
## [36] "mean of time domain gravity accelerometer magnitude mean"
## [37] "mean of time domain gravity accelerometer magnitude std"
## [38] "mean of time domain body accelerometer jerksignal magnitude mean"
## [39] "mean of time domain body accelerometer jerksignal magnitude std"
## [40] "mean of time domain body gyroscope magnitude mean"
## [41] "mean of time domain body gyroscope magnitude std"
## [42] "mean of time domain body gyroscope jerksignal magnitude mean"
## [43] "mean of time domain body gyroscope jerksignal magnitude std"
## [44] "mean of frequency domain body accelerometer mean x"
## [45] "mean of frequency domain body accelerometer mean y"
## [46] "mean of frequency domain body accelerometer mean z"
## [47] "mean of frequency domain body accelerometer std x"
## [48] "mean of frequency domain body accelerometer std y"
## [49] "mean of frequency domain body accelerometer std z"
## [50] "mean of frequency domain body accelerometer mean frequency x"
## [51] "mean of frequency domain body accelerometer mean frequency y"
## [52] "mean of frequency domain body accelerometer mean frequency z"
## [53] "mean of frequency domain body accelerometer jerksignal mean x"
## [54] "mean of frequency domain body accelerometer jerksignal mean y"
## [55] "mean of frequency domain body accelerometer jerksignal mean z"
## [56] "mean of frequency domain body accelerometer jerksignal std x"
## [57] "mean of frequency domain body accelerometer jerksignal std y"
## [58] "mean of frequency domain body accelerometer jerksignal std z"
## [59] "mean of frequency domain body accelerometer jerksignal mean frequency x"
## [60] "mean of frequency domain body accelerometer jerksignal mean frequency y"
## [61] "mean of frequency domain body accelerometer jerksignal mean frequency z"
## [62] "mean of frequency domain body gyroscope mean x"
## [63] "mean of frequency domain body gyroscope mean y"
## [64] "mean of frequency domain body gyroscope mean z"
## [65] "mean of frequency domain body gyroscope std x"
## [66] "mean of frequency domain body gyroscope std y"
## [67] "mean of frequency domain body gyroscope std z"
## [68] "mean of frequency domain body gyroscope mean frequency x"
## [69] "mean of frequency domain body gyroscope mean frequency y"
## [70] "mean of frequency domain body gyroscope mean frequency z"
## [71] "mean of frequency domain body accelerometer magnitude mean"
## [72] "mean of frequency domain body accelerometer magnitude std"
## [73] "mean of frequency domain body accelerometer magnitude mean frequency"
## [74] "mean of frequency domain body accelerometer jerksignal magnitude mean"
## [75] "mean of frequency domain body accelerometer jerksignal magnitude std"
## [76] "mean of frequency domain body accelerometer jerksignal magnitude mean frequency"
## [77] "mean of frequency domain body gyroscope magnitude mean"
## [78] "mean of frequency domain body gyroscope magnitude std"
## [79] "mean of frequency domain body gyroscope magnitude mean frequency"
## [80] "mean of frequency domain body gyroscope jerksignal magnitude mean"
## [81] "mean of frequency domain body gyroscope jerksignal magnitude std"
## [82] "mean of frequency domain body gyroscope jerksignal magnitude mean frequency"
```

Objective #5 is to calculate the mean of each variable (feature) for each subject at each activity. The first step is to **melt()** the data into a *long* data set. The dataframe 'subset6m' is created ('subsets 4 and 5' were temporary and discarded).

```
subset6m <- melt(subset3, id.vars = c("subject", "activities", "activity"))
```

This allows **dplyr** to use the **summarize()** function. First, the data needs to be grouped, using **group_by()**, creating individual groups of "subject" x "activity" x "variable". "variable" was created by default by **melt()** and consists of the feature names.

```
subset6m <- group_by(subset6m, subject, activity, variable)
```

I then create a data frame of the mean values by the above groups, named 'mean_subset'.

```
mean_subset <- summarize(subset6m, mean(value))
```

This is a *long* data set. Each line shows the subject, activity, and variable, and the mean value of the repeated measurements of that variable calculated above by **summarize()**.

```
head(mean_subset)
```

```
## # A tibble: 6 x 4
## # Groups:   subject, activity [1]
##   subject activity variable                                    `mean(value)`
##     <int> <chr>    <fct>                                               <dbl>
## 1       1 LAYING   mean of time domain body accelerometer mean x       0.222
## 2       1 LAYING   mean of time domain body accelerometer mean y      -0.0405
## 3       1 LAYING   mean of time domain body accelerometer mean z      -0.113
## 4       1 LAYING   mean of time domain body accelerometer std x       -0.928
## 5       1 LAYING   mean of time domain body accelerometer std y       -0.837
## 6       1 LAYING   mean of time domain body accelerometer std z       -0.826
```

```
tail(mean_subset)
```

```
## # A tibble: 6 x 4
## # Groups:   subject, activity [1]
##   subject activity      variable                                     `mean(value)`
##     <int> <chr>         <fct>                                               <dbl>
## 1      30 WALKING_UPST~ mean of frequency domain body gyroscope m~      -0.449
## 2      30 WALKING_UPST~ mean of frequency domain body gyroscope m~      -0.151
## 3      30 WALKING_UPST~ mean of frequency domain body gyroscope m~      -0.457
## 4      30 WALKING_UPST~ mean of frequency domain body gyroscope j~      -0.774
## 5      30 WALKING_UPST~ mean of frequency domain body gyroscope j~      -0.791
## 6      30 WALKING_UPST~ mean of frequency domain body gyroscope j~      -0.0714
```

To make it a little more easy to interpret, a *wide* data set is created using **dcast**.

```
mean_subsetw <- dcast(mean_subset, subject + activity ~ variable,
                      value.var = "mean(value)")
```

Each row of 'mean_subsetw' consists of the subject, activity, and the mean value of the repeated measurements of each variable, with variables as columns.

```
mean_subsetw[1:10,1:6]
```

```
##    subject           activity mean of time domain body accelerometer mean x
## 1        1             LAYING                                      0.2215982
## 2        1            SITTING                                      0.2612376
## 3        1           STANDING                                      0.2789176
## 4        1            WALKING                                      0.2773308
## 5        1 WALKING_DOWNSTAIRS                                      0.2891883
## 6        1   WALKING_UPSTAIRS                                      0.2554617
## 7        2             LAYING                                      0.2813734
```

```
## 8        2            SITTING                                    0.2770874
## 9        2           STANDING                                    0.2779115
## 10       2            WALKING                                    0.2764266
##    mean of time domain body accelerometer mean y
## 1                                    -0.040513953
## 2                                    -0.001308288
## 3                                    -0.016137590
## 4                                    -0.017383819
## 5                                    -0.009918505
## 6                                    -0.023953149
## 7                                    -0.018158740
## 8                                    -0.015687994
## 9                                    -0.018420827
## 10                                   -0.018594920
##    mean of time domain body accelerometer mean z
## 1                                     -0.1132036
## 2                                     -0.1045442
## 3                                     -0.1106018
## 4                                     -0.1111481
## 5                                     -0.1075662
## 6                                     -0.0973020
## 7                                     -0.1072456
## 8                                     -0.1092183
## 9                                     -0.1059085
## 10                                    -0.1055004
##    mean of time domain body accelerometer std x
## 1                                   -0.92805647
## 2                                   -0.97722901
## 3                                   -0.99575990
## 4                                   -0.28374026
## 5                                    0.03003534
## 6                                   -0.35470803
## 7                                   -0.97405946
## 8                                   -0.98682228
## 9                                   -0.98727189
## 10                                  -0.42364284
```