Data Wrangling: Human Activity Recognition

Pratik Gandhi

April 18, 2016

0: Loading all the data

```
library(tidyr)
library(dplyr)

## Warning: package 'dplyr' was built under R version 3.2.4

##
## 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

basedir <- "C:/Users/Pratik Gandhi/Documents/Data Science
Stuff/Foundation_DS/UCI_HAR_Dataset/"
setwd(basedir)
dirs <- list.dirs(path=basedir, full.names=TRUE, recursive=FALSE)

## Loading the test dataset
setwd(paste0(basedir,"test"))
test_files <- list.files(paste0(basedir,"test"))
X_test <- read.table(test_files[grepl("X_test.txt$",test_files)])
y_test <- read.table(test_files[grepl("y_test.txt$",test_files)])
subject_test <- read.table(test_files[grepl("subject_test.txt$",test_files)])

## Loading the train dataset
setwd(paste0(basedir,"train"))
train_files <- list.files(paste0(basedir,"train"))
X_train <- read.table(train_files[grepl("X_train.txt$",train_files)])
y_train <- read.table(train_files[grepl("y_train.txt$",train_files)])
subject_train <-
read.table(train_files[grepl("subject_train.txt$",train_files)])

## Loading the activity labels and features
setwd(basedir)
base_files <- list.files(basedir)
features_file <- read.table(base_files[grepl("features.txt",base_files)])
activity_labels <-
```

```
read.table(base_files[grepl("activity_labels.txt",base_files)])

## Showing few rows of the train dataset
head(X_train[1:10,1:10],n=2)

##          V1          V2         V3         V4         V5         V6
## 1 0.2885845 -0.02029417 -0.1329051 -0.9952786 -0.9831106 -0.9135264
## 2 0.2784188 -0.01641057 -0.1235202 -0.9982453 -0.9753002 -0.9603220
##           V7         V8         V9        V10
## 1 -0.9951121 -0.9831846 -0.9235270 -0.9347238
## 2 -0.9988072 -0.9749144 -0.9576862 -0.9430675

head(y_train,n=5)

##   V1
## 1  5
## 2  5
## 3  5
## 4  5
## 5  5

head(subject_train, n=5)

##   V1
## 1  1
## 2  1
## 3  1
## 4  1
## 5  1

head(features_file)

##   V1                V2
## 1  1 tBodyAcc-mean()-X
## 2  2 tBodyAcc-mean()-Y
## 3  3 tBodyAcc-mean()-Z
## 4  4  tBodyAcc-std()-X
## 5  5  tBodyAcc-std()-Y
## 6  6  tBodyAcc-std()-Z

head(activity_labels)

##   V1                 V2
## 1  1            WALKING
## 2  2   WALKING_UPSTAIRS
## 3  3 WALKING_DOWNSTAIRS
## 4  4            SITTING
## 5  5           STANDING
## 6  6             LAYING
```

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

```
X_combined <- rbind(X_train, X_test)
y_combined <- rbind(y_train, y_test)
subject_combined <- rbind(subject_train, subject_test)

## Showing few rows of combined dataset
head(X_combined[1:10,1:10],n=2)
```

```
##          V1          V2         V3         V4         V5         V6
## 1 0.2885845 -0.02029417 -0.1329051 -0.9952786 -0.9831106 -0.9135264
## 2 0.2784188 -0.01641057 -0.1235202 -0.9982453 -0.9753002 -0.9603220
##           V7         V8         V9        V10
## 1 -0.9951121 -0.9831846 -0.9235270 -0.9347238
## 2 -0.9988072 -0.9749144 -0.9576862 -0.9430675
```

```
head(y_combined, n=5)
```

```
##   V1
## 1  5
## 2  5
## 3  5
## 4  5
## 5  5
```

```
head(subject_combined, n=5)
```

```
##   V1
## 1  1
## 2  1
## 3  1
## 4  1
## 5  1
```

2:Extracts columns containing mean and standard deviation for each measurement

```
feature_names <- features_file$V2
feature_names <- make.names(feature_names,unique = TRUE)

## Assigning the values to our dataset
colnames(X_combined) <- feature_names

init_mean_std_data <- X_combined %>% select( matches("(mean|std)"))
#col_mean <- grep("(mean|std)", names(X_combined), value = TRUE)
#mean_cols <- X_combined[,col_mean]

head(feature_names) ## This can be compared to the features_file and see how
make.names function works to create syntactically valid names
```

```
## [1] "tBodyAcc.mean...X" "tBodyAcc.mean...Y" "tBodyAcc.mean...Z"
## [4] "tBodyAcc.std...X"  "tBodyAcc.std...Y"  "tBodyAcc.std...Z"
```

```
head(init_mean_std_data[1:10,1:20], n=2)
```

```
##   tBodyAcc.mean...X tBodyAcc.mean...Y tBodyAcc.mean...Z tBodyAcc.std...X
## 1         0.2885845       -0.02029417        -0.1329051       -0.9952786
## 2         0.2784188       -0.01641057        -0.1235202       -0.9982453
##   tBodyAcc.std...Y tBodyAcc.std...Z tGravityAcc.mean...X
## 1       -0.9831106       -0.9135264             0.9633961
## 2       -0.9753002       -0.9603220             0.9665611
##   tGravityAcc.mean...Y tGravityAcc.mean...Z tGravityAcc.std...X
## 1           -0.1408397            0.1153749          -0.9852497
## 2           -0.1415513            0.1093788          -0.9974113
##   tGravityAcc.std...Y tGravityAcc.std...Z tBodyAccJerk.mean...X
## 1          -0.9817084          -0.8776250             0.07799634
## 2          -0.9894474          -0.9316387             0.07400671
##   tBodyAccJerk.mean...Y tBodyAccJerk.mean...Z tBodyAccJerk.std...X
## 1           0.005000803           -0.06783081           -0.9935191
## 2           0.005771104            0.02937663           -0.9955481
##   tBodyAccJerk.std...Y tBodyAccJerk.std...Z tBodyGyro.mean...X
## 1           -0.9883600           -0.9935750       -0.006100849
## 2           -0.9810636           -0.9918457       -0.016111620
##   tBodyGyro.mean...Y
## 1        -0.03136479
## 2        -0.08389378
```

3:Creates variables called ActivityLabel and ActivityName that label all observations with the corresponding activity labels and names respectively

```
names(activity_labels) <- c("ActivityLabel", "ActivityName")
names(subject_combined) <- c("subject")
names(y_combined) <- c("ActivityLabel")
total_data <- cbind(subject_combined, y_combined) %>%
left_join(activity_labels) %>% cbind(X_combined)
```

```
## Joining by: "ActivityLabel"
```

```
head(subject_combined,n=5)
```

```
##   subject
## 1       1
## 2       1
## 3       1
## 4       1
## 5       1
```

```
head(y_combined, n=5)
```

```
##   ActivityLabel
## 1             5
## 2             5
```

```
## 3             5
## 4             5
## 5             5
```

```
head(total_data[1:10,1:10], n=2)
```

```
##   subject ActivityLabel ActivityName tBodyAcc.mean...X tBodyAcc.mean...Y
## 1       1             5     STANDING         0.2885845       -0.02029417
## 2       1             5     STANDING         0.2784188       -0.01641057
##   tBodyAcc.mean...Z tBodyAcc.std...X tBodyAcc.std...Y tBodyAcc.std...Z
## 1        -0.1329051       -0.9952786       -0.9831106       -0.9135264
## 2        -0.1235202       -0.9982453       -0.9753002       -0.9603220
##   tBodyAcc.mad...X
## 1       -0.9951121
## 2       -0.9988072
```

4: From the data set in step 3, creates a second, independent tidy data set with the average of each variable for each activity and each subject.

```
mean_std_data <- total_data %>% select( matches("(mean|std)"))
tidy_data <- total_data %>% group_by(subject, ActivityName) %>%
summarise_each(funs(mean), -one_of(c('subjet', 'ActivityLabel',
'ActivityName')))
head(mean_std_data[1:10,1:10],n=2)
```

```
##   tBodyAcc.mean...X tBodyAcc.mean...Y tBodyAcc.mean...Z tBodyAcc.std...X
## 1         0.2885845       -0.02029417        -0.1329051       -0.9952786
## 2         0.2784188       -0.01641057        -0.1235202       -0.9982453
##   tBodyAcc.std...Y tBodyAcc.std...Z tGravityAcc.mean...X
## 1       -0.9831106       -0.9135264            0.9633961
## 2       -0.9753002       -0.9603220            0.9665611
##   tGravityAcc.mean...Y tGravityAcc.mean...Z tGravityAcc.std...X
## 1           -0.1408397            0.1153749          -0.9852497
## 2           -0.1415513            0.1093788          -0.9974113
```

```
head(tidy_data[1:10,1:10],n=2)
```

```
## Source: local data frame [2 x 10]
## Groups: subject [1]
##
##   subject ActivityName tBodyAcc.mean...X tBodyAcc.mean...Y
##     (int)       (fctr)             (dbl)             (dbl)
## 1       1       LAYING         0.2215982      -0.040513953
## 2       1      SITTING         0.2612376      -0.001308288
## Variables not shown: tBodyAcc.mean...Z (dbl), tBodyAcc.std...X (dbl),
##   tBodyAcc.std...Y (dbl), tBodyAcc.std...Z (dbl), tBodyAcc.mad...X (dbl),
##   tBodyAcc.mad...Y (dbl)
```