CodeBook

Eduardo Zornoff

6/23/2020

Course Project Getting and Cleaning Data - Johns Hopkins University

1. Merging the files and creating one dataset

The first step was setup the variables to be able to collect the data and manipulate the data

“ver” is the two types of data train and test

“name” is the file names to access the data

“features” is the name vector of the data variables

“activities” is the data frame (df) that contains the description of the activities

- temporary variables are not described

```
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
```

```
ver = c("train", "test") # two sets of data

name = c("subject", "X", "Y") #, "body_acc_x", "body_acc_y", "body_acc_z",
         #"body_gyro_x", "body_gyro_y", "body_gyro_z",
         #"total_acc_x", "total_acc_y", "total_acc_z") Define which file to be read

features = read.csv("./data/features.txt", sep = " ", header = FALSE)[,2] # extract features variables

activities = read.csv("./data/activity_labels.txt", sep = " ", header = FALSE) # extract activities
names(activities) = c("activity", "activity description")
```

Next we read the files that were need and assemble the data df

“data” is the desired table

“n” is the selector of the file order

“v” is the selector of the train or test file

```
data = data.frame() # initializing data data frame

readFile = function(n) { # n is the file to be read / readi train and test files and combine
        n = n
        # read train
        v = 1
        dFile = paste("./data/", ver[v], "/", name[n], "_", ver[v], ".txt", sep = "")
        if (!file.exists(dFile)) {dFile = paste("./data/", ver[v], "/Inertial Signals/",
                                        name[n], "_", ver[v], ".txt", sep = "")}
        train = read.csv(dFile)
        names(train)[1] = name[n]
        # read test
        v = 2
        dFile = paste("./data/", ver[v], "/", name[n], "_", ver[v], ".txt", sep = "")
        if (!file.exists(dFile)) {dFile = paste("./data/", ver[v], "/Inertial Signals/",
                                        name[n], "_", ver[v], ".txt", sep = "")}

        test = read.csv(dFile)
        names(test)[1] = name[n]
        # bind rows
        temp = rbind(train, test)
        temp
}

# initialize data data.frame with subject column
data = readFile(1)

# populate data.frame with all the column data from the other variables

for (i in (2:3)) { # i is the file from the list / there's is still other 9 basic features totalling 12
        temp = readFile(i)
        data = cbind(data, temp)
}
rm("temp") # releasing memory
```

2. Extracting measurements (variables) / 4. Labelling the data set

Indentifing and selecting the variables that contains ‘mean’ or ‘std’

“mn” is the boolean vector that identifies which variables contains “mean”

“st” is the boolean vector that identifies which variables contains “std”

“cl” is the boolean vector that combines both mn and st

“feat” is the desired names for the columns that were selected

```
library(stringi)

# identify which variables contains mean or std

mn = stri_detect_fixed(features, "mean") & !stri_detect_fixed(features, "meanFreq")
st = stri_detect_fixed(features, "std")
cl = mn | st            # aggregate both criteria
feat = features[cl]     # create variable name vector
```

Split the variables into independent numeric columns because all the variables are into a consolidated character string

"Xdata" subset of "data" containing only the variables data

"i" is the line being processed

"Xsplit" is the resulting df of this stage

```
# splitinng X into variables

Xdata = data[,"X"] # selecting only the variable column

# cut X data from data, rename Y column to create a correspondence with activities df
data = data %>% select(subject, activity = Y)

# function to character split to numeric
spt = function(i) {
        t = stri_trim(Xdata[i]) # trim the sting to remove more than one spaces
        t = as.numeric(simplify2array(strsplit(t, " "))) # extract the numbers from the string
        t = t[!is.na(t)] # remove NAs
        t = t[cl] # get only the info for the desired variables (mean and std variables)
        t
}

Xsplit = spt(1) # initialize Xplit

# run spt for all lines of X data

for (i in (2:10297)) { # i is the line of the list

        temp = spt(i)
        Xsplit = rbind(Xsplit, temp)

}
rm("temp") # releasing memory
rm("Xdata")
Xsplit = as.data.frame(Xsplit)

library(plyr)
```

```
## ------------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------------------

##
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
Xsplit = unrowname(Xsplit) # strip row names
names(Xsplit) = feat # naming columns of Xsplit
```

Next we add the Xsplit to the data and get the tidy data needed

```
# adding Xsplit to data

data = cbind(data, Xsplit)
rm("Xsplit") # releasing memory
```

3. Naming the activities by the descriptive name

```
# adding activity description
data = left_join(data, activities, by = "activity")
# moving next to activity code and elimination the activity code
data = data %>% relocate(`activity description`, .after = activity) %>%
        select(-activity)

print(head(data))
```

```
##   subject activity description tBodyAcc-mean()-X tBodyAcc-mean()-Y
## 1       1             STANDING         0.2784188       -0.01641057
## 2       1             STANDING         0.2796531       -0.01946716
## 3       1             STANDING         0.2791739       -0.02620065
## 4       1             STANDING         0.2766288       -0.01656965
## 5       1             STANDING         0.2771988       -0.01009785
## 6       1             STANDING         0.2794539       -0.01964078
##   tBodyAcc-mean()-Z tBodyAcc-std()-X tBodyAcc-std()-Y tBodyAcc-std()-Z
## 1        -0.1235202       -0.9982453       -0.9753002       -0.9603220
## 2        -0.1134617       -0.9953796       -0.9671870       -0.9789440
## 3        -0.1232826       -0.9960915       -0.9834027       -0.9906751
## 4        -0.1153619       -0.9981386       -0.9808173       -0.9904816
## 5        -0.1051373       -0.9973350       -0.9904868       -0.9954200
## 6        -0.1100221       -0.9969210       -0.9671859       -0.9831178
##   tGravityAcc-mean()-X tGravityAcc-mean()-Y tGravityAcc-mean()-Z
## 1            0.9665611           -0.1415513           0.10937881
## 2            0.9668781           -0.1420098           0.10188392
## 3            0.9676152           -0.1439765           0.09985014
## 4            0.9682244           -0.1487502           0.09448590
```

```
## 5            0.9679482           -0.1482100           0.09190972
## 6            0.9679295           -0.1442821           0.09314463
##   tGravityAcc-std()-X tGravityAcc-std()-Y tGravityAcc-std()-Z
## 1          -0.9974113          -0.9894474          -0.9316387
## 2          -0.9995740          -0.9928658          -0.9929172
## 3          -0.9966456          -0.9813928          -0.9784764
## 4          -0.9984293          -0.9880982          -0.9787449
## 5          -0.9989793          -0.9867539          -0.9973064
## 6          -0.9993325          -0.9885747          -0.9920159
##   tBodyAccJerk-mean()-X tBodyAccJerk-mean()-Y tBodyAccJerk-mean()-Z
## 1            0.07400671           0.005771104           0.029376633
## 2            0.07363596           0.003104037          -0.009045631
## 3            0.07732061           0.020057642          -0.009864772
## 4            0.07344436           0.019121574           0.016779979
## 5            0.07793244           0.018684046           0.009344434
## 6            0.08217077          -0.017014670          -0.015798166
##   tBodyAccJerk-std()-X tBodyAccJerk-std()-Y tBodyAccJerk-std()-Z
## 1           -0.9955481           -0.9810636           -0.9918457
## 2           -0.9907428           -0.9809556           -0.9896866
## 3           -0.9926974           -0.9875527           -0.9934976
## 4           -0.9964202           -0.9883587           -0.9924549
## 5           -0.9948136           -0.9887145           -0.9922663
## 6           -0.9952056           -0.9848308           -0.9884251
##   tBodyGyro-mean()-X tBodyGyro-mean()-Y tBodyGyro-mean()-Z tBodyGyro-std()-X
## 1        -0.01611162        -0.08389378         0.10058429        -0.9831200
## 2        -0.03169829        -0.10233542         0.09612688        -0.9762921
## 3        -0.04340998        -0.09138618         0.08553770        -0.9913848
## 4        -0.03396042        -0.07470803         0.07739203        -0.9851836
## 5        -0.02877551        -0.07039311         0.07901214        -0.9851808
## 6        -0.02860025        -0.08304673         0.09546456        -0.9881772
##   tBodyGyro-std()-Y tBodyGyro-std()-Z tBodyGyroJerk-mean()-X
## 1        -0.9890458        -0.9891212            -0.11050283
## 2        -0.9935518        -0.9863787            -0.10848567
## 3        -0.9924073        -0.9875542            -0.09116989
## 4        -0.9923781        -0.9874019            -0.09077010
## 5        -0.9921175        -0.9830768            -0.09424758
## 6        -0.9892057        -0.9791538            -0.09708861
##   tBodyGyroJerk-mean()-Y tBodyGyroJerk-mean()-Z tBodyGyroJerk-std()-X
## 1            -0.04481873            -0.05924282            -0.9898726
## 2            -0.04241031            -0.05582883            -0.9884618
## 3            -0.03633262            -0.06046466            -0.9911194
## 4            -0.03763253            -0.05828932            -0.9913545
## 5            -0.04335526            -0.04193600            -0.9916216
## 6            -0.04158928            -0.04470456            -0.9904185
##   tBodyGyroJerk-std()-Y tBodyGyroJerk-std()-Z tBodyAccMag-mean()
## 1            -0.9972926            -0.9938510         -0.9792892
## 2            -0.9956321            -0.9915318         -0.9837031
## 3            -0.9966410            -0.9933289         -0.9865418
## 4            -0.9964730            -0.9945110         -0.9928271
## 5            -0.9960147            -0.9930906         -0.9942950
## 6            -0.9954146            -0.9904868         -0.9874657
##   tBodyAccMag-std() tGravityAccMag-mean() tGravityAccMag-std()
## 1        -0.9760571            -0.9792892           -0.9760571
## 2        -0.9880196            -0.9837031           -0.9880196
```

```
## 3        -0.9864213            -0.9865418           -0.9864213
## 4        -0.9912754            -0.9928271           -0.9912754
## 5        -0.9952490            -0.9942950           -0.9952490
## 6        -0.9827460            -0.9874657           -0.9827460
##   tBodyAccJerkMag-mean() tBodyAccJerkMag-std() tBodyGyroMag-mean()
## 1             -0.9912535           -0.9916944          -0.9806831
## 2             -0.9885313           -0.9903969          -0.9763171
## 3             -0.9930780           -0.9933808          -0.9820599
## 4             -0.9934800           -0.9958537          -0.9852037
## 5             -0.9930177           -0.9954243          -0.9858944
## 6             -0.9913143           -0.9894478          -0.9855007
##   tBodyGyroMag-std() tBodyGyroJerkMag-mean() tBodyGyroJerkMag-std()
## 1         -0.9837542              -0.9951232             -0.9961016
## 2         -0.9860515              -0.9934032             -0.9950910
## 3         -0.9873511              -0.9955022             -0.9952666
## 4         -0.9890626              -0.9958076             -0.9952580
## 5         -0.9864403              -0.9952748             -0.9952050
## 6         -0.9846253              -0.9937188             -0.9952695
##   fBodyAcc-mean()-X fBodyAcc-mean()-Y fBodyAcc-mean()-Z fBodyAcc-std()-X
## 1        -0.9974507        -0.9768517        -0.9735227       -0.9986803
## 2        -0.9935941        -0.9725115        -0.9833040       -0.9963128
## 3        -0.9954906        -0.9835697        -0.9910798       -0.9963121
## 4        -0.9972859        -0.9823010        -0.9883694       -0.9986065
## 5        -0.9966567        -0.9869395        -0.9927386       -0.9976438
## 6        -0.9958491        -0.9676116        -0.9841233       -0.9974612
##   fBodyAcc-std()-Y fBodyAcc-std()-Z fBodyAccJerk-mean()-X fBodyAccJerk-mean()-Y
## 1       -0.9749298       -0.9554381            -0.9950322            -0.9813115
## 2       -0.9655059       -0.9770493            -0.9909937            -0.9816423
## 3       -0.9832444       -0.9902291            -0.9944466            -0.9887272
## 4       -0.9801295       -0.9919150            -0.9962920            -0.9887900
## 5       -0.9922637       -0.9970459            -0.9948507            -0.9882443
## 6       -0.9679258       -0.9828873            -0.9947551            -0.9832403
##   fBodyAccJerk-mean()-Z fBodyAccJerk-std()-X fBodyAccJerk-std()-Y
## 1            -0.9897398           -0.9966523           -0.9820839
## 2            -0.9875663           -0.9912488           -0.9814148
## 3            -0.9913542           -0.9913783           -0.9869269
## 4            -0.9906244           -0.9969025           -0.9886067
## 5            -0.9901575           -0.9952180           -0.9901788
## 6            -0.9873372           -0.9962421           -0.9882634
##   fBodyAccJerk-std()-Z fBodyGyro-mean()-X fBodyGyro-mean()-Y fBodyGyro-mean()-Z
## 1           -0.9926268         -0.9773867         -0.9925300         -0.9896058
## 2           -0.9904159         -0.9754332         -0.9937147         -0.9867557
## 3           -0.9943908         -0.9871096         -0.9936015         -0.9871913
## 4           -0.9929065         -0.9824465         -0.9929838         -0.9886664
## 5           -0.9930667         -0.9848902         -0.9927862         -0.9807784
## 6           -0.9879085         -0.9860273         -0.9904991         -0.9784560
##   fBodyGyro-std()-X fBodyGyro-std()-Y fBodyGyro-std()-Z fBodyAccMag-mean()
## 1        -0.9849043        -0.9871681        -0.9897847         -0.9808566
## 2        -0.9766422        -0.9933990        -0.9873282         -0.9877948
## 3        -0.9928104        -0.9916460        -0.9886776         -0.9875187
## 4        -0.9859818        -0.9919558        -0.9879443         -0.9935909
## 5        -0.9852871        -0.9916595        -0.9853661         -0.9948360
## 6        -0.9887881        -0.9884058        -0.9811471         -0.9821347
##   fBodyAccMag-std() fBodyBodyAccJerkMag-mean() fBodyBodyAccJerkMag-std()
```

```
## 1        -0.9758658                -0.9903355                -0.9919603
## 2        -0.9890155                -0.9892801                -0.9908667
## 3        -0.9867420                -0.9927689                -0.9916998
## 4        -0.9900635                -0.9955228                -0.9943890
## 5        -0.9952833                -0.9947329                -0.9951562
## 6        -0.9847729                -0.9878858                -0.9905461
##   fBodyBodyGyroMag-mean() fBodyBodyGyroMag-std() fBodyBodyGyroJerkMag-mean()
## 1              -0.9882956             -0.9833219                  -0.9958539
## 2              -0.9892548             -0.9860277                  -0.9950305
## 3              -0.9894128             -0.9878358                  -0.9952207
## 4              -0.9914330             -0.9890594                  -0.9950928
## 5              -0.9905000             -0.9858609                  -0.9951433
## 6              -0.9882692             -0.9845685                  -0.9956415
##   fBodyBodyGyroJerkMag-std()
## 1                 -0.9963995
## 2                 -0.9951274
## 3                 -0.9952369
## 4                 -0.9954648
## 5                 -0.9952387
## 6                 -0.9946391
```

Now we have data df with identifiers "subject" and "activity description" on 1st and 2nd columns and variables on the next 66 columns in numeric form

5. Create an independent dataset for the mean of the variables per subject and activity

```
# creating the new data frame with the mean for each combination of subject and activity
# and each variable
dataSum = data %>% group_by(subject, `activity description`) %>% summarise_all(mean)

print(head(dataSum))
```

```
## # A tibble: 6 x 68
## # Groups:   subject [1]
##   subject `activity descr~ `tBodyAcc-mean(~ `tBodyAcc-mean(~ `tBodyAcc-mean(~
##     <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           -0.110
## 4       1 WALKING                    0.277         -0.0174           -0.111
## 5       1 WALKING_DOWNSTA~           0.289         -0.00992          -0.108
## 6       1 WALKING_UPSTAIRS           0.255         -0.0240           -0.0973
## # ... with 63 more variables: `tBodyAcc-std()-X` <dbl>,
## #   `tBodyAcc-std()-Y` <dbl>, `tBodyAcc-std()-Z` <dbl>,
## #   `tGravityAcc-mean()-X` <dbl>, `tGravityAcc-mean()-Y` <dbl>,
## #   `tGravityAcc-mean()-Z` <dbl>, `tGravityAcc-std()-X` <dbl>,
## #   `tGravityAcc-std()-Y` <dbl>, `tGravityAcc-std()-Z` <dbl>,
## #   `tBodyAccJerk-mean()-X` <dbl>, `tBodyAccJerk-mean()-Y` <dbl>,
## #   `tBodyAccJerk-mean()-Z` <dbl>, `tBodyAccJerk-std()-X` <dbl>,
```

```
## #   `tBodyAccJerk-std()-Y` <dbl>, `tBodyAccJerk-std()-Z` <dbl>,
## #   `tBodyGyro-mean()-X` <dbl>, `tBodyGyro-mean()-Y` <dbl>,
## #   `tBodyGyro-mean()-Z` <dbl>, `tBodyGyro-std()-X` <dbl>,
## #   `tBodyGyro-std()-Y` <dbl>, `tBodyGyro-std()-Z` <dbl>,
## #   `tBodyGyroJerk-mean()-X` <dbl>, `tBodyGyroJerk-mean()-Y` <dbl>,
## #   `tBodyGyroJerk-mean()-Z` <dbl>, `tBodyGyroJerk-std()-X` <dbl>,
## #   `tBodyGyroJerk-std()-Y` <dbl>, `tBodyGyroJerk-std()-Z` <dbl>,
## #   `tBodyAccMag-mean()` <dbl>, `tBodyAccMag-std()` <dbl>,
## #   `tGravityAccMag-mean()` <dbl>, `tGravityAccMag-std()` <dbl>,
## #   `tBodyAccJerkMag-mean()` <dbl>, `tBodyAccJerkMag-std()` <dbl>,
## #   `tBodyGyroMag-mean()` <dbl>, `tBodyGyroMag-std()` <dbl>,
## #   `tBodyGyroJerkMag-mean()` <dbl>, `tBodyGyroJerkMag-std()` <dbl>,
## #   `fBodyAcc-mean()-X` <dbl>, `fBodyAcc-mean()-Y` <dbl>,
## #   `fBodyAcc-mean()-Z` <dbl>, `fBodyAcc-std()-X` <dbl>,
## #   `fBodyAcc-std()-Y` <dbl>, `fBodyAcc-std()-Z` <dbl>,
## #   `fBodyAccJerk-mean()-X` <dbl>, `fBodyAccJerk-mean()-Y` <dbl>,
## #   `fBodyAccJerk-mean()-Z` <dbl>, `fBodyAccJerk-std()-X` <dbl>,
## #   `fBodyAccJerk-std()-Y` <dbl>, `fBodyAccJerk-std()-Z` <dbl>,
## #   `fBodyGyro-mean()-X` <dbl>, `fBodyGyro-mean()-Y` <dbl>,
## #   `fBodyGyro-mean()-Z` <dbl>, `fBodyGyro-std()-X` <dbl>,
## #   `fBodyGyro-std()-Y` <dbl>, `fBodyGyro-std()-Z` <dbl>,
## #   `fBodyAccMag-mean()` <dbl>, `fBodyAccMag-std()` <dbl>,
## #   `fBodyBodyAccJerkMag-mean()` <dbl>, `fBodyBodyAccJerkMag-std()` <dbl>,
## #   `fBodyBodyGyroMag-mean()` <dbl>, `fBodyBodyGyroMag-std()` <dbl>,
## #   `fBodyBodyGyroJerkMag-mean()` <dbl>, `fBodyBodyGyroJerkMag-std()` <dbl>
```

et voilà