run_analysis

Stephanie R. Beck Roth

July 21, 2015

This program is written to fulfill the requirements of

the Getting and Cleaning Data Coursera Course. Smartphone

Data was extracted from the following websites belonging

to UCI Machine Learning Repository.

Òhttp://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones#Ó ### Program written by Stephanie R. Beck Roth ### Due 7-26-2015

There are 2 parts to this CodeBook.

1: The first is the code to produce the tidytable.txt file and the second is a description of the variables inclcuded in the text file.

The following is the R code to Load and Produce the alldatatrim master tidy dataset as required from objective 4.

```
library(doBy)
```

```
## Loading required package: survival
```

```
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
```

```
library(tidyr)
library(gtools)
#Get Data if not already downloaded to save time.  :)

if (!file.exists("./UCI\ HAR\ Dataset/test")){
    fileUrl <- "https://d396qusza40orc.cloudfront.net/getdata%2Fprojectfiles%2FUCI%20HAR%20Dat
aset.zip"
    download.file(fileUrl,destfile = "./SmartPhone.zip", method="curl")
    mycsv <- unzip("./SmartPhone.zip", exdir = ".")
}

# The first 5 text files are descriptors of the data.
# Files 5-13 contain Inertial Signals tests data.
# Files 14-16 contain subject, X and Y tests data, respectively.
# Files 17-25 contain the Inertial Signals training data.
# Files 26-28 contain subject, X and Y training data, respectively.
# Only including the following files for analysis per discussion forum post by course TA,
# David Hood at https://class.coursera.org/getdata-030/forum/thread?thread_id=37

# Read file containing activity labels for dytest and dytrain activity codes
aln <- read.delim("./UCI\ HAR\ Dataset/activity_labels.txt", sep=" ",
                  col.names=c("code","activitylabel"),header=FALSE)

dytest <- read.table("./UCI\ HAR\ Dataset/test/y_test.txt",
                     col.names = "code", header=FALSE)
dytrain <- read.table("./UCI\ HAR\ Dataset/train/y_train.txt",
                      col.names = "code", header=FALSE)

# Read file containing column headers for dxtest and dxtrain
coln <- read.delim("./UCI\ HAR\ Dataset/features.txt", sep=" ",header=FALSE)
cnames <- as.character(coln$V2)

# Read file containing participant identifiers for dstest
# Note: There are only 30 participants and each participant is either in the
# training dataset or the test dataset.
dstest <- read.delim("./UCI\ HAR\ Dataset/test/subject_test.txt",header=FALSE)
part_id_test <- as.factor(dstest$V1)

# Read test Content Files and wrap the data frames as data tables and setting variables to tho
se in features.txt
dxtest <- tbl_df(read.table("./UCI\ HAR\ Dataset/test/X_test.txt", col.names=cnames))
# Create testdata which will have the following new variables stored in the beginning of the t
able dxtest
# dxtest$set <- "test"
# dxtest$part_id <- part_id_test
# dxtest$code <- dytest$code
testdata <- cbind(part_id=part_id_test,set="test",code=dytest$code,dxtest)

# Read file containing participant identifiers for dstrain
dstrain <- read.delim("./UCI\ HAR\ Dataset/train/subject_train.txt",header=FALSE)
part_id_train <- as.factor(dstrain$V1)

# Read train Content Files and wrap the data frames as data tables and setting variables to th
```

```
ose in features.txt
dxtrain <- tbl_df(read.table("./UCI\ HAR\ Dataset/train/X_train.txt", col.names=cnames))

# Create traindata which will have the following new variables stored in the beginning of the
table dxtrain
# dxtrain$set <- "train"
# dxtrain$part_id <- part_id_train
# dxtrain$code <- dytrain$code
traindata <- cbind(part_id=part_id_train,set="train",code=dytrain$code,dxtrain)

# Create Data set with both test and train data
# Using rbind since the variable names are identical
alldata <- rbind(testdata,traindata)
# Introducde the activitylabel variable which exlains the information in the variable code.
alldata <- merge(alldata, aln, by="code")
# Reorder the table so that the activity label variable is near the beginning in the dataset
alldata <- alldata[, c(1,2,3,565,4:564)]
# Deleting the code variable to make it tidy since the activitylabel variable is now in the da
taset
# to convey the same information.
alldata <- select(alldata, -code)
# Select only part_id, set, activitylabel (columns 1-3)
# and those variables containing mean or std in their names.
cols <- 1:3
colsm <- grep("mean", colnames(alldata))
cols <- append(cols,colsm)
colstd <- grep("std", colnames(alldata))
cols <- sort(append(cols,colstd))
alldatatrim <- alldata[,cols]
```

The following R code creates the second independent tidy data set containing the average of each variable for each activity and each subject.

```
# Creating the tidy dataset
tidytable <-summaryBy(. ~set+part_id+activitylabel,data=alldatatrim,FUN=mean)
```

The following R code writes the second independent tidy data set containing the average of each variable for each activity and each subject.

```
# Writing the tidy dataset
write.table(tidytable, file="./tidytable.txt", row.name=FALSE)
```

2: This begins the second portion of the CodeBook containing the variable descriptions.

The following variables were created and extracted from the data contained in

the zip file:

https://d396qusza40orc.cloudfront.net/getdata%2Fprojectfiles%2FUCI%20HAR%20Dataset.zip ### Further documentation on the collection of the original variables can be found within the README.txt of the data which upon extraction will reside within a directory labeled ÒUCI HAR DatasetÓ. This original README.txt file provides the necessary details regarding the specifics of the data collection methodologies from which the following variables were created and should be referenced prior to interpretation of the following information for a more thorough and detailed understanding.

This dataset contains summary measurements on the 30 participants who were recruited into the two distinct groups, training and test, to perform the original analysis. Thus, every participant is a member of only one dataset.

ÒsetÓ contains a 2 level factor where ÒtestÓ and ÒtrainÓ indicate data belonging to the test dataset and the training dataset, respective.

Òpart_idÓ contains a 30 level factor comprised of integers in the 1-30 range. This variable indicates the original participant identification number.

ÒactivitylabelÓ contains one of the 6 activity labels: ÒLAYINGÓ, ÒSITTINGÓ, ÒSTANDINGÓ, ÒWALKINGÓ, ÒWALKING_UPSTAIRSÓ, ÒWALKING_DOWNSTAIRSÓ.

The remaining 79 variables contain the mean of their respective measurements from the original datasets calculated separately by the participant and activity labels. These summaries exist only for the original measurements containing mean and standard deviation.

Note:

The run_analysis.R code also indicates the means were performed by the set variable but as no participant is a member of both the training and test datasets, it merely provides a more thorough grouping summary and has no bearing on the calculations.)

The 79 variables following ÒsetÓ, Òpart_idÓ and ÒactivitylabelÓ contain the same name as the original data variables where each Ò-Ó, Ò(Ó and Ò)Ó are replaced by a . from the original variable and conclude with Ò.meanÓ to indicate the mean of the original variable was taken.

```
##  [1] "set"
##  [2] "part_id"
##  [3] "activitylabel"
##  [4] "tBodyAcc.mean...X.mean"
##  [5] "tBodyAcc.mean...Y.mean"
##  [6] "tBodyAcc.mean...Z.mean"
##  [7] "tBodyAcc.std...X.mean"
##  [8] "tBodyAcc.std...Y.mean"
##  [9] "tBodyAcc.std...Z.mean"
## [10] "tGravityAcc.mean...X.mean"
## [11] "tGravityAcc.mean...Y.mean"
## [12] "tGravityAcc.mean...Z.mean"
## [13] "tGravityAcc.std...X.mean"
## [14] "tGravityAcc.std...Y.mean"
## [15] "tGravityAcc.std...Z.mean"
## [16] "tBodyAccJerk.mean...X.mean"
## [17] "tBodyAccJerk.mean...Y.mean"
## [18] "tBodyAccJerk.mean...Z.mean"
## [19] "tBodyAccJerk.std...X.mean"
## [20] "tBodyAccJerk.std...Y.mean"
## [21] "tBodyAccJerk.std...Z.mean"
## [22] "tBodyGyro.mean...X.mean"
## [23] "tBodyGyro.mean...Y.mean"
## [24] "tBodyGyro.mean...Z.mean"
## [25] "tBodyGyro.std...X.mean"
## [26] "tBodyGyro.std...Y.mean"
## [27] "tBodyGyro.std...Z.mean"
## [28] "tBodyGyroJerk.mean...X.mean"
## [29] "tBodyGyroJerk.mean...Y.mean"
## [30] "tBodyGyroJerk.mean...Z.mean"
## [31] "tBodyGyroJerk.std...X.mean"
## [32] "tBodyGyroJerk.std...Y.mean"
## [33] "tBodyGyroJerk.std...Z.mean"
## [34] "tBodyAccMag.mean...mean"
## [35] "tBodyAccMag.std...mean"
## [36] "tGravityAccMag.mean...mean"
## [37] "tGravityAccMag.std...mean"
## [38] "tBodyAccJerkMag.mean...mean"
## [39] "tBodyAccJerkMag.std...mean"
## [40] "tBodyGyroMag.mean...mean"
## [41] "tBodyGyroMag.std...mean"
## [42] "tBodyGyroJerkMag.mean...mean"
## [43] "tBodyGyroJerkMag.std...mean"
## [44] "fBodyAcc.mean...X.mean"
## [45] "fBodyAcc.mean...Y.mean"
## [46] "fBodyAcc.mean...Z.mean"
## [47] "fBodyAcc.std...X.mean"
## [48] "fBodyAcc.std...Y.mean"
## [49] "fBodyAcc.std...Z.mean"
## [50] "fBodyAcc.meanFreq...X.mean"
## [51] "fBodyAcc.meanFreq...Y.mean"
## [52] "fBodyAcc.meanFreq...Z.mean"
## [53] "fBodyAccJerk.mean...X.mean"
## [54] "fBodyAccJerk.mean...Y.mean"
## [55] "fBodyAccJerk.mean...Z.mean"
## [56] "fBodyAccJerk.std...X.mean"
## [57] "fBodyAccJerk.std...Y.mean"
## [58] "fBodyAccJerk.std...Z.mean"
## [59] "fBodyAccJerk.meanFreq...X.mean"
## [60] "fBodyAccJerk.meanFreq...Y.mean"
## [61] "fBodyAccJerk.meanFreq...Z.mean"
## [62] "fBodyGyro.mean...X.mean"
## [63] "fBodyGyro.mean...Y.mean"
## [64] "fBodyGyro.mean...Z.mean"
## [65] "fBodyGyro.std...X.mean"
## [66] "fBodyGyro.std...Y.mean"
## [67] "fBodyGyro.std...Z.mean"
## [68] "fBodyGyro.meanFreq...X.mean"
## [69] "fBodyGyro.meanFreq...Y.mean"
## [70] "fBodyGyro.meanFreq...Z.mean"
## [71] "fBodyAccMag.mean...mean"
## [72] "fBodyAccMag.std...mean"
## [73] "fBodyAccMag.meanFreq...mean"
## [74] "fBodyBodyAccJerkMag.mean...mean"
## [75] "fBodyBodyAccJerkMag.std...mean"
## [76] "fBodyBodyAccJerkMag.meanFreq...mean"
## [77] "fBodyBodyGyroMag.mean...mean"
## [78] "fBodyBodyGyroMag.std...mean"
## [79] "fBodyBodyGyroMag.meanFreq...mean"
## [80] "fBodyBodyGyroJerkMag.mean...mean"
## [81] "fBodyBodyGyroJerkMag.std...mean"
## [82] "fBodyBodyGyroJerkMag.meanFreq...mean"
```