README

Shuguang Ji

December 26, 2015

This is a script description document. This document explains how all of the scripts work and how they are connected.

Step 0: download data set

After the working directory is set up. The following codes can be used to download the Samsung online data. I renamed the unzipped data set from "UCI HAR Dataset" to "data".

There is one thing need to be mentioned. Even though the data links starts with "https", if we add method="curl" in download.file command, it will report error. If you encounter the same issue, just remove method="curl".

```
fileUrl <- "https://d396qusza40orc.cloudfront.net/getdata%2Fprojectfiles%2FUCI%20HAR%20Dataset.zip";
download.file(fileUrl, destfile = "data.zip");##No method="curl" used.
unzip("data.zip");
file.rename("UCI HAR Dataset","data");
```

Step 1: Merge to one data set

After data set is ready in your working directory, you need to walk through each file in you data set package to undstand the structure of each data set. In order to merge the data sets into one data set. There are four substeps needing to be followed.

1.1. Read data from the files into the variables. And combine the variables by rows.

```
TrainSet<-read.table("data/train/X_train.txt");
TestSet<-read.table("data/test/X_test.txt");
SetData<-rbind(TrainSet,TestSet);

TrainLabel<-read.table("data/train/y_train.txt");
TestLabel<-read.table("data/test/y_test.txt");
LabelData<-rbind(TrainLabel,TestLabel);

TrainSubj<-read.table("data/train/subject_train.txt");
TestSubj<-read.table("data/test/subject_test.txt");
SubjData<-rbind(TrainSubj,TestSubj);
```

1.2.Set names to combined variables.

```
names(SubjData)<-c("subject");
names(LabelData)<-c("activity");
DataFeatureNames<-read.table("data/features.txt",head=FALSE);
names(SetData)<-DataFeatureNames$V2;
```

1.3. Merge the combined variables into final data set.

```
DataCombine<-cbind(SubjData,LabelData);
Data<-cbind(SetData,DataCombine);
```

Step 2: Extract mean and standard deviation

Two substeps are used to extract mean and standard deviation values.First, Name of features by the mean and standard deviation values are extracted. Then, subset the data frame of the final merged data set ("Data") by selected names of features.Rename the new subset data set as "Data_Mean_Sd".

```
SubFeature<-DataFeatureNames$V2[grep("mean\\(\\)|std\\(\\)", DataFeatureNames$V2)];
SelectedNames<-c(as.character(SubFeature), "subject", "activity" );
Data_Mean_Sd<-subset(Data,select=SelectedNames);
```

Step 3: Describe the activities in the data set

Assigne the data set generated in step 2 to "DataDesc". Read descriptive activity names from activity_lables.txt. Then factorize variable "activity" in DataDesc by descriptive activity names.

```
DataDesc<-Data_Mean_Sd;
ActivityLabel<-read.table("data/activity_labels.txt",header = FALSE);
DataDesc$activity<-factor(DataDesc$activity);
DataDesc$activity<- factor(DataDesc$activity,labels=as.character(ActivityLabel$V2));
```

Step 4: Label data set

"t" is labeled by "Time"; "Acc" is labeled by "Accelerometer"; "Gyro" is labeled by "Gyroscope"; "Mag" is labled by "Magnitude"; "f" is labeled by "Frequency"; "BodyBody" is labeled by "Body".

```
names(DataDesc)<-gsub("^t", "Time", names(DataDesc));
names(DataDesc)<-gsub("Acc", "Accelerometer", names(DataDesc));
names(DataDesc)<-gsub("Gyro", "Gyroscope", names(DataDesc));
names(DataDesc)<-gsub("Mag", "Magnitude", names(DataDesc));
names(DataDesc)<-gsub("^f", "Frequency", names(DataDesc));
names(DataDesc)<-gsub("BodyBody", "Body", names(DataDesc));
```

Step 5: Create new data set

New independent tidy data set is created with the average of each variable for each activity and each subject of DataDesc. The new data set is saved in the working directory with name of "submission.txt"

```
D2<-aggregate(x=DataDesc, by=list(activities=DataDesc$activity, subj=DataDesc$subject), FUN=mean);
D2<-D2[, !(colnames(D2) %in% c("subj", "activity"))];
write.table(D2,'submission.txt', row.names = F);
```

Note: str() can be used to check the data frame of the data set.