codebook

dawky

Friday, December 08, 2017

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```r
setwd("D:/r/coursera/gettingcleandata")
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
```

```r
#1
#combine the data,read the x_train,y_train data,subject_train data and
combine them into train data, do the same work for the test data, and t
hen put the data together
trainx <- read.table("./UCI HAR Dataset/train/X_train.txt")
trainy <- read.table("./UCI HAR Dataset/train/y_train.txt")
trainsubject <- read.table("./UCI HAR Dataset/train/subject_train.txt")
testx <- read.table("./UCI HAR Dataset/test/X_test.txt")
testy <- read.table("./UCI HAR Dataset/test/y_test.txt")
testsubject <- read.table("./UCI HAR Dataset/test/subject_test.txt")

train <- cbind(trainsubject, trainy, trainx)
test <- cbind(testsubject, testy, testx)
dat <- rbind(train, test)

#2
#find the mean and std, from the feature_info file, we know the "mean"
and ¡°std" word appear in the variable name
#I firstly select the needed variable from the features.txt file, and f
ind the position of these variables from the featurename(feature2), and
 then apply it to the original data
```

```r
feature<- read.table("./UCI HAR Dataset/features.txt")
feature2 <- grep(("mean\\(\\)|std\\(\\)"), feature[, 2])
featurename<-feature[feature2,2]
dat2 <- dat[, c(1, 2, feature2+2)]
colnames(dat2)<-c("subject","activity",paste(featurename))

#3
#change the activity number with name,view the activity name, find the
number of each activity, and replace them
activity<- read.table("./UCI HAR Dataset/activity_labels.txt")
View(activity)
dat2[,2][dat2[,2]== 1] <- "Walking"
dat2[,2][dat2[,2] == 2] <- "Walking Upstairs"
dat2[,2][dat2[,2] == 3] <- "Walking Downstairs"
dat2[,2][dat2[,2] == 4] <- "Sitting"
dat2[,2][dat2[,2] == 5] <- "Standing"
dat2[,2][dat2[,2] == 6] <- "Laying"

#4
#redo the step2, change the short word into their full name, rewrite th
e featurename, and apply it to the colname
#for example, from the feature_info.txt file, we know that Acc represen
t for Accelerator, Mag represent for Magnitude,and it's them same for "
Gyro"and "Gyroscope","t"and "time","f"and "frequency"
#we change our featurename list, screen for the original word and repla
ce it with full name, and apply it to the colname
featurename2<-featurename
featurename2<-gsub("Acc", "Accelerator", featurename2)
featurename2<-gsub("Mag", "Magnitude", featurename2)
featurename2<-gsub("Gyro", "Gyroscope", featurename2)
featurename2<-gsub("^t", "time", featurename2)
featurename2<-gsub("^f", "frequency", featurename2)

colnames(dat2)<-c("subject","activity",paste(featurename2))



#5
#aggregate each col by the two listed variable using group_by, and then
 export it to clean txt file

cleandata<-dat2%>%group_by(subject,activity)%>%summarise_each(funs(mea
n))

write.table(cleandata, file = "cleandata.txt", row.names = FALSE)

##summary and decription for the variables in the cleandata
##variable for the feature
# "timeBodyAccelerator-mean()-X"                     "timeBodyAccelerat
or-mean()-Y"
```

```
# "timeBodyAccelerator-mean()-Z"                     "timeBodyAccelerat
or-std()-X"
# "timeBodyAccelerator-std()-Y"                      "timeBodyAccelerat
or-std()-Z"
# "timeGravityAccelerator-mean()-X"                  "timeGravityAccele
rator-mean()-Y"
# "timeGravityAccelerator-mean()-Z"                  "timeGravityAccele
rator-std()-X"
# "timeGravityAccelerator-std()-Y"                   "timeGravityAccele
rator-std()-Z"
# "timeBodyAcceleratorJerk-mean()-X"                 "timeBodyAccelerat
orJerk-mean()-Y"
# "timeBodyAcceleratorJerk-mean()-Z"                 "timeBodyAccelerat
orJerk-std()-X"
# "timeBodyAcceleratorJerk-std()-Y"                  "timeBodyAccelerat
orJerk-std()-Z"
# "timeBodyGyroscope-mean()-X"                       "timeBodyGyroscope
-mean()-Y"
# "timeBodyGyroscope-mean()-Z"                       "timeBodyGyroscope
-std()-X"
# "timeBodyGyroscope-std()-Y"                        "timeBodyGyroscope
-std()-Z"
# "timeBodyGyroscopeJerk-mean()-X"                   "timeBodyGyroscope
Jerk-mean()-Y"
# "timeBodyGyroscopeJerk-mean()-Z"                   "timeBodyGyroscope
Jerk-std()-X"
# "timeBodyGyroscopeJerk-std()-Y"                    "timeBodyGyroscope
Jerk-std()-Z"
# "timeBodyAcceleratorMagnitude-mean()"              "timeBodyAccelerat
orMagnitude-std()"
# "timeGravityAcceleratorMagnitude-mean()"           "timeGravityAccele
ratorMagnitude-std()"
# "timeBodyAcceleratorJerkMagnitude-mean()"          "timeBodyAccelerat
orJerkMagnitude-std()"
# "timeBodyGyroscopeMagnitude-mean()"                "timeBodyGyroscope
Magnitude-std()"
# "timeBodyGyroscopeJerkMagnitude-mean()"            "timeBodyGyroscope
JerkMagnitude-std()"
# "frequencyBodyAccelerator-mean()-X"                "frequencyBodyAcce
lerator-mean()-Y"
# "frequencyBodyAccelerator-mean()-Z"                "frequencyBodyAcce
lerator-std()-X"
# "frequencyBodyAccelerator-std()-Y"                 "frequencyBodyAcce
lerator-std()-Z"
# "frequencyBodyAcceleratorJerk-mean()-X"            "frequencyBodyAcce
leratorJerk-mean()-Y"
# "frequencyBodyAcceleratorJerk-mean()-Z"            "frequencyBodyAcce
leratorJerk-std()-X"
# "frequencyBodyAcceleratorJerk-std()-Y"             "frequencyBodyAcce
leratorJerk-std()-Z"
```

```
# "frequencyBodyGyroscope-mean()-X"                  "frequencyBodyGyro
scope-mean()-Y"
# "frequencyBodyGyroscope-mean()-Z"                  "frequencyBodyGyro
scope-std()-X"
# "frequencyBodyGyroscope-std()-Y"                   "frequencyBodyGyro
scope-std()-Z"
# "frequencyBodyAcceleratorMagnitude-mean()"         "frequencyBodyAcce
leratorMagnitude-std()"
# "frequencyBodyBodyAcceleratorJerkMagnitude-mean()" "frequencyBodyBody
AcceleratorJerkMagnitude-std()"
# "frequencyBodyBodyGyroscopeMagnitude-mean()"       "frequencyBodyBody
GyroscopeMagnitude-std()"
# "frequencyBodyBodyGyroscopeJerkMagnitude-mean()"   "frequencyBodyBody
GyroscopeJerkMagnitude-std()"

##subject:
#The participant ("subject") ID

##acticity:
#The label of the activity performed when the corresponding measurement
s were taken
#WALKING (1)
#WALKING_UPSTAIRS (2)
#WALKING_DOWNSTAIRS (3)
#SITTING (4)
#STANDING (5)
#LAYING (6)
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.