run_analysis explained

Patrick Ball

Loading the Data

The following part of the script loads the txt data files into R data frames.

```
test_s <- read.table("test/subject_test.txt")
test_y <- read.table("test/y_test.txt")
test_x <- read.table("test/X_test.txt")
train_s <- read.table("train/subject_train.txt")
train_y <- read.table("train/y_train.txt")
train_x <- read.table("train/X_train.txt")
features_labels <- read.table("features.txt")
act_labels <- read.table("activity_labels.txt")
```

Append Descriptive Activity Names

The following part of the script appends the activities list with a column of descriptive text entries. I chose not to completely replace the original numeric values.

```
for (i in 1:dim(test_y)[1]) {
    j <- test_y[i,1]
    test_y[i,2] <- act_labels[j,2]
}
for (i in 1:dim(train_y)[1]) {
    j <- train_y[i,1]
    train_y[i,2] <- act_labels[j,2]
}
```

Name the Variables

This part of the script adds column names to the existing data frames.

```
names(test_s) <- names(train_s) <- "subject"
names(test_y) <- names(train_y) <- c("activity_code","activity_desc")
names(test_x) <- names(train_x) <- features_labels[,2]
```

Combine the Data - Part 1

This part of the script combines the data horizontally, creating one data frame for all test data and another for all training data.

```
testdata <- cbind(test_s, test_y, test_x)
traindata <- cbind(train_s, train_y, train_x)
```

Combine the Data - Part 2

This part of the script combines the data vertically, creating one data frame of all data available.

```
alldata <- rbind(testdata, traindata)
alldata$subject <- as.factor(alldata$subject)
alldata$activity_code <- as.factor(alldata$activity_code)
```

Subset the Data

This part of the script records which columns have "mean" or "std" in them. I chose to retain the mean frequency measures for completeness (a future user could remove them if they aren't necessary, but could not add them if they weren't present). The script then creates the subdata data frame with the subject, activity, and mean & std columns.

```
meancols <- grep(".*mean", names(alldata))
stdcols <- grep(".*std", names(alldata))
neededcols <- sort(c(1:3,meancols,stdcols))
subdata <- alldata[,neededcols]
```

Create Aggregated Data Frame

The following part of the script aggregates the subdata data frame by subject, activity_code, and activity_desc. It them removes excess columns and fixes column names.

```
attach(subdata)
aggdata <- aggregate(subdata, by=list(subject, activity_code, activity_desc),
                     FUN=mean)
detach(subdata)
aggdata <- aggdata[,c(1:3,7:85)]
names(aggdata)[1:3] <- c("subject", "activity_code", "activity_desc")
```

Output Results

This part of the script writes the tidy, subset, and aggregated data to a "tidy_data.txt" file.

```
write.table(aggdata, file="tidy_data.txt", row.names=FALSE)
```