# CodeBook

Mark Edney

23/09/2020

## Data cleaning assignment

This is the codebook for the final assignment in the data cleaning course. This codebook describes the r script run_analysis.R

## Variables

train_x- data from the training group
train_y- activity data for the training group
test_x- data from the test group
test_y- activity data for the test group
subject_train- subject data for the training group
subject_test- subject data for the test group
data2- the merged data frame
featurelist- labels for the feature data
activitylabel- labels for the activity data
activity- merged activity data
sortfeatures- logical vector for sorting df by std and mean
datasecond- second dataframe grouped by activity and subject and summarized

## Data

X_train.txt- The raw data for the training group
Y_train.txt- The raw data describing the activities of the training group
X_test.txt- The raw data for the test group
Y_test.txt- The raw data describing the activities of the test group
subject_train.txt- The raw data for the subject ID in the training group
subject_test.txt- The raw data for the subject ID in the test group
features.txt- The labels of the features for the recorded data
activity_labels.txt- The labels for the activities

## Transformations

The raw data transformation follows the following procedure:
1. Load the dplyr packages

```
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
```

2. Load the different datafiles into the system as dataframes

```
train_x<-read.table("UCI HAR Dataset\\train\\X_train.txt")
train_y<-read.table("UCI HAR Dataset\\train\\Y_train.txt")
test_x<-read.table("UCI HAR Dataset\\test\\x_test.txt")
test_y<-read.table("UCI HAR Dataset\\test\\y_test.txt")
subject_train<-read.table("UCI HAR Dataset\\train\\subject_train.txt")
subject_test<-read.table("UCI HAR Dataset\\test\\subject_test.txt")
```

3. Merge the train and test dataframes together and ensure there is no missing data

```
data2<-merge(train_x,test_x,all=TRUE)
data2<-data2[complete.cases(data2),]
```

4. Load the featurelist and activity labels into the system as dataframes

```
featurelist<-read.table("UCI HAR Dataset\\features.txt")
activitylabel<-read.table("UCI HAR Dataset\\activity_labels.txt")
```

5. Merge the activity list for the train and test data

```
activity<-rbind(train_y,test_y)
```

6. Sort the merged data from for varialbes that only include std and mean

```
sortfeatures<-grepl(".*[Mm][Ee][Aa][Nn].*|.*[Ss][Tt][Dd].*",featurelist[,2])
data2<-data2[,sortfeatures]
```

7. Change the variables of the merged data with the featurelist df

```
names(data2)<-featurelist[sortfeatures,2]
```

8. Amends the activity data to the merged data with mutate and changes the activity to the description with the activity label df

```
data2<-mutate(data2,"activity"=activity)
data2$activity<-activitylabel[unlist(data2$activity),2]
```

9. Amends the subject data to the merged df

```
data2<-mutate(data2,"subject"=rbind(subject_train,subject_test))
```

10. Creates a second df grouped by activity and then by subject

```
datasecond<-group_by(data2,activity, subject)
```

11. Summarize the second df with the summarize_all by mean

```
datasecond<-summarise_at(datasecond,c(1:86),mean,na.rm = TRUE)
```