

Assignment Getting Cleaning Data

Andre Burgaud

March 11, 2016

1 - Merge the training and the test sets to create one data set

Training Data

```
library(plyr) # mapvalues
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:plyr':
##
##   arrange, count, desc, failwith, id, mutate, rename, summarise,
##   summarize

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

setwd(".")
# Assuming the data is in the current directory
# Load training signals in data frame
trainSignals <- read.table("UCI HAR Dataset/train/X_train.txt", header=FALSE)
# Load training subjects in a vector
subject <- scan("UCI HAR Dataset/train/subject_train.txt")
# Load training activities in a vector
activity <- scan("UCI HAR Dataset/train/y_train.txt")
# Add set type column (train vs. test)
trainDF <- cbind(set="train", trainSignals)
# Add activity column
trainDF <- cbind(activity, trainDF)
# Add subject column
trainDF <- cbind(subject, trainDF)
```

Test Data

```

# Assuming the data is in the current directory
# Load testing signals in data frame
testSignals <- read.table("UCI HAR Dataset/test/X_test.txt", header=FALSE)
# Load testing subjects in a vector
subject <- scan("UCI HAR Dataset/test/subject_test.txt")
# Load testing activities in a vector
activity <- scan("UCI HAR Dataset/test/y_test.txt")
# Add set type column (test vs. train)
testDF <- cbind(set="test", testSignals)
# Add activity column
testDF <- cbind(activity, testDF)
# Add subject column
testDF <- cbind(subject, testDF)

```

Merge Data Sets

```

DF <- merge(trainDF, testDF, all=TRUE)
str(DF)

```

```

## 'data.frame': 10299 obs. of 564 variables:
## $ subject : num 1 1 1 1 1 1 1 1 1 1 ...
## $ activity: num 1 1 1 1 1 1 1 1 1 1 ...
## $ set : Factor w/ 2 levels "train","test": 1 1 1 1 1 1 1 1 1 1 ...
## $ V1 : num 0.156 0.18 0.19 0.202 0.204 ...
## $ V2 : num -0.04961 -0.0178 -0.0389 -0.00904 -0.03051 ...
## $ V3 : num -0.1129 -0.0393 -0.0987 -0.0791 -0.1371 ...
## $ V4 : num -0.297 -0.377 -0.331 -0.221 -0.247 ...
## $ V5 : num 0.1756 0.0235 0.0448 0.1231 0.2986 ...
## $ V6 : num -0.254 -0.247 -0.354 -0.222 -0.206 ...
## $ V7 : num -0.392 -0.434 -0.385 -0.272 -0.3 ...
## $ V8 : num 0.088 -0.0419 0.0128 0.0727 0.2651 ...
## $ V9 : num -0.237 -0.243 -0.329 -0.21 -0.246 ...
## $ V10 : num 0.1165 -0.2292 -0.1669 -0.1195 -0.0859 ...
## $ V11 : num -0.0194 -0.1187 -0.0623 -0.105 -0.0922 ...
## $ V12 : num -0.355 -0.227 -0.41 -0.238 -0.147 ...
## $ V13 : num -0.1079 0.1099 -0.0316 0.2378 0.1655 ...
## $ V14 : num -0.069 0.0808 0.0654 -0.0447 -0.1629 ...
## $ V15 : num 0.38 0.42 0.454 0.429 0.407 ...
## $ V16 : num -0.1473 -0.2068 -0.1922 -0.0682 -0.0426 ...
## $ V17 : num -0.749 -0.802 -0.773 -0.694 -0.714 ...
## $ V18 : num -0.732 -0.797 -0.788 -0.756 -0.675 ...
## $ V19 : num -0.748 -0.736 -0.809 -0.725 -0.714 ...
## $ V20 : num -0.539 -0.534 -0.467 -0.335 -0.375 ...
## $ V21 : num -0.392 -0.396 -0.245 -0.35 -0.133 ...
## $ V22 : num -0.244 -0.185 -0.371 -0.107 -0.372 ...
## $ V23 : num 0.269 0.304 0.36 0.222 0.297 ...
## $ V24 : num 0.322 0.467 0.364 0.457 0.416 ...
## $ V25 : num 0.2292 0.0989 0.122 0.0747 -0.0262 ...
## $ V26 : num -0.0866 -0.4113 -0.412 -0.4492 -0.1353 ...
## $ V27 : num 0.104 0.458 0.468 0.472 0.096 ...
## $ V28 : num 0.15 -0.304 -0.243 -0.359 0.102 ...

```

```

## $ V29 : num -0.1149 0.1609 0.0855 0.2435 -0.0194 ...
## $ V30 : num -0.224 -0.4 -0.286 -0.426 -0.154 ...
## $ V31 : num 0.219 0.393 0.287 0.434 0.138 ...
## $ V32 : num 0.14518 -0.00193 0.11167 -0.05746 0.25176 ...
## $ V33 : num -0.09331 -0.05973 -0.13985 0.00446 -0.14631 ...
## $ V34 : num 0.0665 -0.2198 -0.0923 -0.1015 -0.1917 ...
## $ V35 : num 0.0206 0.167 0.1309 0.1121 0.2389 ...
## $ V36 : num 0.36086 -0.00316 0.21078 0.00516 -0.19821 ...
## $ V37 : num -0.63902 -0.12758 -0.52724 -0.17857 0.00423 ...
## $ V38 : num -0.3356 -0.1265 -0.0995 -0.2879 -0.0878 ...
## $ V39 : num -0.4477 -0.1133 -0.3152 -0.222 -0.0816 ...
## $ V40 : num 0.437 0.387 0.279 0.39 0.39 ...
## $ V41 : num 0.911 0.951 0.929 0.946 0.954 ...
## $ V42 : num -0.334 -0.248 -0.302 -0.255 -0.243 ...
## $ V43 : num -0.1447 -0.0385 -0.1194 -0.0226 -0.023 ...
## $ V44 : num -0.933 -0.983 -0.981 -0.99 -0.989 ...
## $ V45 : num -0.935 -0.958 -0.98 -0.963 -0.979 ...
## $ V46 : num -0.725 -0.956 -0.979 -0.954 -0.977 ...
## $ V47 : num -0.932 -0.984 -0.985 -0.99 -0.99 ...
## $ V48 : num -0.94 -0.959 -0.979 -0.961 -0.98 ...
## $ V49 : num -0.732 -0.958 -0.979 -0.956 -0.977 ...
## $ V50 : num 0.86 0.882 0.858 0.874 0.883 ...
## $ V51 : num -0.337 -0.254 -0.316 -0.268 -0.258 ...
## $ V52 : num -0.057 -0.0339 -0.1214 -0.0205 -0.0264 ...
## $ V53 : num 0.91 0.965 0.934 0.961 0.969 ...
## $ V54 : num -0.332 -0.227 -0.277 -0.238 -0.222 ...
## $ V55 : num -0.1851 -0.0468 -0.1227 -0.0328 -0.026 ...
## $ V56 : num 0.1567 -0.2819 0.0366 -0.3163 -0.3346 ...
## $ V57 : num 0.763 0.868 0.809 0.853 0.874 ...
## $ V58 : num -0.807 -0.898 -0.844 -0.891 -0.903 ...
## $ V59 : num -0.952 -0.996 -0.969 -0.998 -0.998 ...
## $ V60 : num -0.931 -0.983 -0.992 -0.992 -0.991 ...
## $ V61 : num -0.949 -0.96 -0.977 -0.964 -0.98 ...
## $ V62 : num -0.768 -0.964 -0.98 -0.965 -0.976 ...
## $ V63 : num -0.0667 -1 -0.5078 -1 -1 ...
## $ V64 : num -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
## $ V65 : num -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
## $ V66 : num -0.321 -0.224 -0.285 -0.193 -0.195 ...
## $ V67 : num 0.405 0.306 0.377 0.277 0.286 ...
## $ V68 : num -0.488 -0.384 -0.468 -0.353 -0.37 ...
## $ V69 : num 0.571 0.456 0.558 0.421 0.446 ...
## $ V70 : num -0.0907 -0.0559 0.023 -0.0262 0.0266 ...
## $ V71 : num 0.1182 0.1099 0.0229 0.0929 0.0355 ...
## $ V72 : num -0.21 -0.231 -0.14 -0.229 -0.171 ...
## $ V73 : num 0.326 0.376 0.283 0.39 0.332 ...
## $ V74 : num -0.27579 -0.55437 0.00322 -0.38426 -0.30142 ...
## $ V75 : num 0.2981 0.581 0.0216 0.4088 0.336 ...
## $ V76 : num -0.3206 -0.6067 -0.0432 -0.4314 -0.368 ...
## $ V77 : num 0.3408 0.6287 0.0597 0.4495 0.3948 ...
## $ V78 : num 0.966 0.885 0.423 -0.428 -0.72 ...
## $ V79 : num 0.891 -0.353 0.246 -0.705 -0.2 ...
## $ V80 : num 0.7743 -0.0059 0.3784 0.8055 0.1683 ...
## $ V81 : num 0.2151 0.3631 0.2202 -0.1378 -0.0752 ...
## $ V82 : num 0.2801 0.1097 0.0691 0.4212 0.2401 ...

```

```
## $ V83      : num  0.202 0.253 -0.209 0.507 -0.259 ...
## $ V84      : num  -0.0117 -0.2244 -0.1152 -0.1362 -0.0812 ...
## $ V85      : num  0.1059 -0.0527 0.014 0.0427 0.243 ...
## $ V86      : num  -0.351 -0.582 -0.505 -0.546 -0.492 ...
## $ V87      : num  -0.0763 -0.2129 -0.0872 -0.1281 -0.0488 ...
## $ V88      : num  0.0401 -0.0523 0.0177 0.05 0.2589 ...
## $ V89      : num  -0.286 -0.534 -0.452 -0.52 -0.438 ...
## $ V90      : num  -0.063 -0.426 -0.419 -0.424 -0.272 ...
## $ V91      : num  -0.133 -0.345 -0.263 -0.379 -0.31 ...
## $ V92      : num  -0.542 -0.752 -0.63 -0.693 -0.757 ...
## $ V93      : num  -0.3808 -0.1043 -0.2187 0.0663 0.1072 ...
## $ V94      : num  0.1074 0.3805 0.1511 0.1822 -0.0404 ...
## $ V95      : num  0.384 0.546 0.509 0.358 0.416 ...
## $ V96      : num  -0.0611 -0.2348 -0.1329 -0.1661 -0.0443 ...
## [list output truncated]
```

2 - Extracts only measurements on mean and standard deviation for each measurement

```
# Read the features in a table to extract features name
features <- read.table("UCI HAR Dataset/features.txt", header=FALSE, stringsAsFactors=FALSE)
# Convert id value to column names matching the consolidated dataframe DF column names
features <- mutate(features, V1=paste("V", V1, sep=""))
# Extract only the features matching mean of std functions
stdMeanFeatures <- features[grepl("mean\\(\\)|std\\(\\)", features$V2),]
# Extract measurements for std and mean functions in the consolidated DF (preserve subject, activity and test)
firstThreeCols <- names(DF)[1:3]
stdMeanDF <- select_(DF, .dots=c(firstThreeCols, stdMeanFeatures$V1))
str(stdMeanDF)
```

```
## 'data.frame': 10299 obs. of 69 variables:
## $ subject : num  1 1 1 1 1 1 1 1 1 1 ...
## $ activity: num  1 1 1 1 1 1 1 1 1 1 ...
## $ set      : Factor w/ 2 levels "train","test": 1 1 1 1 1 1 1 1 1 1 ...
## $ V1       : num  0.156 0.18 0.19 0.202 0.204 ...
## $ V2       : num  -0.04961 -0.0178 -0.0389 -0.00904 -0.03051 ...
## $ V3       : num  -0.1129 -0.0393 -0.0987 -0.0791 -0.1371 ...
## $ V4       : num  -0.297 -0.377 -0.331 -0.221 -0.247 ...
## $ V5       : num  0.1756 0.0235 0.0448 0.1231 0.2986 ...
## $ V6       : num  -0.254 -0.247 -0.354 -0.222 -0.206 ...
## $ V41      : num  0.911 0.951 0.929 0.946 0.954 ...
## $ V42      : num  -0.334 -0.248 -0.302 -0.255 -0.243 ...
## $ V43      : num  -0.1447 -0.0385 -0.1194 -0.0226 -0.023 ...
## $ V44      : num  -0.933 -0.983 -0.981 -0.99 -0.989 ...
## $ V45      : num  -0.935 -0.958 -0.98 -0.963 -0.979 ...
## $ V46      : num  -0.725 -0.956 -0.979 -0.954 -0.977 ...
## $ V81      : num  0.2151 0.3631 0.2202 -0.1378 -0.0752 ...
## $ V82      : num  0.2801 0.1097 0.0691 0.4212 0.2401 ...
## $ V83      : num  0.202 0.253 -0.209 0.507 -0.259 ...
## $ V84      : num  -0.0117 -0.2244 -0.1152 -0.1362 -0.0812 ...
## $ V85      : num  0.1059 -0.0527 0.014 0.0427 0.243 ...
## $ V86      : num  -0.351 -0.582 -0.505 -0.546 -0.492 ...
```

```

## $ V121 : num 0.2082 -0.044 -0.0428 0.0175 0.0368 ...
## $ V122 : num -0.22176 -0.00142 -0.1664 0.05153 -0.23542 ...
## $ V123 : num -0.1172 0.0757 0.1145 0.1078 0.167 ...
## $ V124 : num -0.225 -0.514 -0.461 -0.445 -0.482 ...
## $ V125 : num 0.0313 -0.0654 -0.0218 -0.0224 0.0137 ...
## $ V126 : num -0.301 -0.333 -0.365 -0.348 -0.316 ...
## $ V161 : num 0.0569 0.0644 -0.0951 -0.1927 -0.1607 ...
## $ V162 : num -0.4138 -0.3598 -0.0566 -0.5182 -0.0958 ...
## $ V163 : num -0.0336 -0.2291 -0.1246 0.0395 -0.0809 ...
## $ V164 : num -0.182 -0.293 -0.159 -0.174 -0.206 ...
## $ V165 : num -0.161 -0.362 -0.236 -0.319 -0.341 ...
## $ V166 : num -0.348 -0.462 -0.395 -0.426 -0.284 ...
## $ V201 : num -0.1518 -0.2065 -0.2043 -0.0836 -0.0465 ...
## $ V202 : num -0.119 -0.305 -0.27 -0.202 -0.186 ...
## $ V214 : num -0.1518 -0.2065 -0.2043 -0.0836 -0.0465 ...
## $ V215 : num -0.119 -0.305 -0.27 -0.202 -0.186 ...
## $ V227 : num -0.104 -0.245 -0.157 -0.175 -0.046 ...
## $ V228 : num 0.1621 -0.202 -0.0971 -0.0923 -0.0603 ...
## $ V240 : num 0.0536 -0.2215 -0.1511 -0.1438 -0.1131 ...
## $ V241 : num -0.124 -0.112 -0.167 -0.149 -0.195 ...
## $ V253 : num -0.203 -0.386 -0.233 -0.3 -0.293 ...
## $ V254 : num -0.202 -0.341 -0.293 -0.336 -0.365 ...
## $ V266 : num -0.151 -0.28 -0.216 -0.182 -0.161 ...
## $ V267 : num 0.1555 -0.1011 0.0173 0.1284 0.2504 ...
## $ V268 : num -0.259 -0.344 -0.397 -0.254 -0.279 ...
## $ V269 : num -0.364 -0.419 -0.382 -0.236 -0.284 ...
## $ V270 : num 0.11182 0.02044 -0.00666 0.04916 0.24154 ...
## $ V271 : num -0.312 -0.253 -0.38 -0.265 -0.228 ...
## $ V345 : num -0.038 -0.26 -0.18 -0.197 -0.115 ...
## $ V346 : num -0.0194 -0.2173 -0.0941 -0.0876 0.1191 ...
## $ V347 : num -0.351 -0.543 -0.491 -0.502 -0.452 ...
## $ V348 : num -0.0723 -0.2553 -0.1254 -0.1485 -0.1274 ...
## $ V349 : num 0.1686 0.0556 0.0647 0.114 0.296 ...
## $ V350 : num -0.35 -0.62 -0.516 -0.588 -0.529 ...
## $ V424 : num -0.0927 -0.4047 -0.2853 -0.2919 -0.4162 ...
## $ V425 : num 0.0203 -0.0819 -0.1039 -0.0771 -0.0965 ...
## $ V426 : num -0.21 -0.253 -0.247 -0.283 -0.227 ...
## $ V427 : num -0.268 -0.549 -0.517 -0.494 -0.504 ...
## $ V428 : num 0.03075 -0.06186 0.01886 0.00317 0.06913 ...
## $ V429 : num -0.398 -0.422 -0.467 -0.431 -0.41 ...
## $ V503 : num 0.0267 -0.2494 -0.1705 -0.085 -0.0688 ...
## $ V504 : num -0.35 -0.445 -0.446 -0.4 -0.388 ...
## $ V516 : num 0.206 -0.2047 -0.0456 -0.0865 -0.0187 ...
## $ V517 : num 0.106 -0.204 -0.172 -0.106 -0.116 ...
## $ V529 : num -0.0735 -0.1175 -0.1511 -0.1547 -0.2683 ...
## $ V530 : num -0.32 -0.263 -0.326 -0.292 -0.283 ...
## $ V542 : num -0.232 -0.333 -0.306 -0.36 -0.37 ...
## $ V543 : num -0.218 -0.397 -0.326 -0.352 -0.403 ...

```

3 - Use descriptive activity names to name activities in the data set

```

#library(plyr)
activityDF <- read.table("UCI HAR Dataset/activity_labels.txt", header=FALSE)
activityLabels = tolower(activityDF$V2)
# Set descriptive activities (i.e. 1="walking", 2="walking_upstairs"... )
stdMeanDF <- mutate(stdMeanDF, activity=mapvalues(activity, from=c(1,2,3,4,5,6), to=activityLabels))
# convert activity variable to factor
stdMeanDF <- mutate(stdMeanDF, activity=factor(activity, levels=tolower(activityDF$V2)))
stdMeanDF$activity[1:30]

```

```

## [1] walking walking walking walking walking walking walking walking
## [9] walking walking walking walking walking walking walking walking
## [17] walking walking walking walking walking walking walking walking
## [25] walking walking walking walking walking walking walking walking
## 6 Levels: walking walking_upstairs walking_downstairs ... laying

```

4 - Appropriately labels the data set with descriptive variable names

```

# Remove special characters from column names (features)
cleanFeatures <- gsub("[\\(\\)-]", "", stdMeanFeatures$V2)
colnames(stdMeanDF) <- c(firstThreeCols, cleanFeatures)
names(stdMeanDF)

```

```

## [1] "subject"          "activity"
## [3] "set"              "tBodyAccmeanX"
## [5] "tBodyAccmeanY"    "tBodyAccmeanZ"
## [7] "tBodyAccstdX"     "tBodyAccstdY"
## [9] "tBodyAccstdZ"     "tGravityAccmeanX"
## [11] "tGravityAccmeanY" "tGravityAccmeanZ"
## [13] "tGravityAccstdX"  "tGravityAccstdY"
## [15] "tGravityAccstdZ"  "tBodyAccJerkmeanX"
## [17] "tBodyAccJerkmeanY" "tBodyAccJerkmeanZ"
## [19] "tBodyAccJerkstdX" "tBodyAccJerkstdY"
## [21] "tBodyAccJerkstdZ" "tBodyGyromeanX"
## [23] "tBodyGyromeanY"   "tBodyGyromeanZ"
## [25] "tBodyGyrostdX"    "tBodyGyrostdY"
## [27] "tBodyGyrostdZ"    "tBodyGyroJerkmeanX"
## [29] "tBodyGyroJerkmeanY" "tBodyGyroJerkmeanZ"
## [31] "tBodyGyroJerkstdX" "tBodyGyroJerkstdY"
## [33] "tBodyGyroJerkstdZ" "tBodyAccMagmean"
## [35] "tBodyAccMagstd"    "tGravityAccMagmean"
## [37] "tGravityAccMagstd" "tBodyAccJerkMagmean"
## [39] "tBodyAccJerkMagstd" "tBodyGyroMagmean"
## [41] "tBodyGyroMagstd"   "tBodyGyroJerkMagmean"
## [43] "tBodyGyroJerkMagstd" "fBodyAccmeanX"
## [45] "fBodyAccmeanY"     "fBodyAccmeanZ"
## [47] "fBodyAccstdX"      "fBodyAccstdY"
## [49] "fBodyAccstdZ"      "fBodyAccJerkmeanX"
## [51] "fBodyAccJerkmeanY" "fBodyAccJerkmeanZ"
## [53] "fBodyAccJerkstdX"  "fBodyAccJerkstdY"
## [55] "fBodyAccJerkstdZ"  "fBodyGyromeanX"

```

```
## [57] "fBodyGyromeanY"          "fBodyGyromeanZ"
## [59] "fBodyGyrostdX"           "fBodyGyrostdY"
## [61] "fBodyGyrostdZ"           "fBodyAccMagmean"
## [63] "fBodyAccMagstd"          "fBodyBodyAccJerkMagmean"
## [65] "fBodyBodyAccJerkMagstd"  "fBodyBodyGyroMagmean"
## [67] "fBodyBodyGyroMagstd"     "fBodyBodyGyroJerkMagmean"
## [69] "fBodyBodyGyroJerkMagstd"
```

5 - From the data set in step 4, creates a second, independent tidy data set with the average of each variable for each activity and each subject

```
meanDF <- stdMeanDF %>%
  group_by(subject, activity) %>%
  summarise_each_(funs(mean), names(stdMeanDF[4:ncol(stdMeanDF)]))
str(meanDF)
```

```
## Classes 'grouped_df', 'tbl_df', 'tbl' and 'data.frame': 180 obs. of 68 variables:
```

```
## $ subject      : num  1 1 1 1 1 1 2 2 2 2 ...
## $ activity      : Factor w/ 6 levels "walking","walking_upstairs",...: 1 2 3 4 5 6 1 2 3 4
## $ tBodyAccmeanX : num  0.277 0.255 0.289 0.261 0.279 ...
## $ tBodyAccmeanY : num  -0.01738 -0.02395 -0.00992 -0.00131 -0.01614 ...
## $ tBodyAccmeanZ : num  -0.1111 -0.0973 -0.1076 -0.1045 -0.1106 ...
## $ tBodyAccstdX  : num  -0.284 -0.355 0.03 -0.977 -0.996 ...
## $ tBodyAccstdY  : num  0.11446 -0.00232 -0.03194 -0.92262 -0.97319 ...
## $ tBodyAccstdZ  : num  -0.26 -0.0195 -0.2304 -0.9396 -0.9798 ...
## $ tGravityAccmeanX : num  0.935 0.893 0.932 0.832 0.943 ...
## $ tGravityAccmeanY : num  -0.282 -0.362 -0.267 0.204 -0.273 ...
## $ tGravityAccmeanZ : num  -0.0681 -0.0754 -0.0621 0.332 0.0135 ...
## $ tGravityAccstdX : num  -0.977 -0.956 -0.951 -0.968 -0.994 ...
## $ tGravityAccstdY : num  -0.971 -0.953 -0.937 -0.936 -0.981 ...
## $ tGravityAccstdZ : num  -0.948 -0.912 -0.896 -0.949 -0.976 ...
## $ tBodyAccJerkmeanX : num  0.074 0.1014 0.0542 0.0775 0.0754 ...
## $ tBodyAccJerkmeanY : num  0.028272 0.019486 0.02965 -0.000619 0.007976 ...
## $ tBodyAccJerkmeanZ : num  -0.00417 -0.04556 -0.01097 -0.00337 -0.00369 ...
## $ tBodyAccJerkstdX : num  -0.1136 -0.4468 -0.0123 -0.9864 -0.9946 ...
## $ tBodyAccJerkstdY : num  0.067 -0.378 -0.102 -0.981 -0.986 ...
## $ tBodyAccJerkstdZ : num  -0.503 -0.707 -0.346 -0.988 -0.992 ...
## $ tBodyGyromeanX   : num  -0.0418 0.0505 -0.0351 -0.0454 -0.024 ...
## $ tBodyGyromeanY   : num  -0.0695 -0.1662 -0.0909 -0.0919 -0.0594 ...
## $ tBodyGyromeanZ   : num  0.0849 0.0584 0.0901 0.0629 0.0748 ...
## $ tBodyGyrostdX    : num  -0.474 -0.545 -0.458 -0.977 -0.987 ...
## $ tBodyGyrostdY    : num  -0.05461 0.00411 -0.12635 -0.96647 -0.98773 ...
## $ tBodyGyrostdZ    : num  -0.344 -0.507 -0.125 -0.941 -0.981 ...
## $ tBodyGyroJerkmeanX : num  -0.09 -0.1222 -0.074 -0.0937 -0.0996 ...
## $ tBodyGyroJerkmeanY : num  -0.0398 -0.0421 -0.044 -0.0402 -0.0441 ...
## $ tBodyGyroJerkmeanZ : num  -0.0461 -0.0407 -0.027 -0.0467 -0.049 ...
## $ tBodyGyroJerkstdX : num  -0.207 -0.615 -0.487 -0.992 -0.993 ...
## $ tBodyGyroJerkstdY : num  -0.304 -0.602 -0.239 -0.99 -0.995 ...
## $ tBodyGyroJerkstdZ : num  -0.404 -0.606 -0.269 -0.988 -0.992 ...
## $ tBodyAccMagmean   : num  -0.137 -0.1299 0.0272 -0.9485 -0.9843 ...
## $ tBodyAccMagstd    : num  -0.2197 -0.325 0.0199 -0.9271 -0.9819 ...
## $ tGravityAccMagmean : num  -0.137 -0.1299 0.0272 -0.9485 -0.9843 ...
```

```

## $ tGravityAccMagstd      : num -0.2197 -0.325 0.0199 -0.9271 -0.9819 ...
## $ tBodyAccJerkMagmean    : num -0.1414 -0.4665 -0.0894 -0.9874 -0.9924 ...
## $ tBodyAccJerkMagstd     : num -0.0745 -0.479 -0.0258 -0.9841 -0.9931 ...
## $ tBodyGyroMagmean       : num -0.161 -0.1267 -0.0757 -0.9309 -0.9765 ...
## $ tBodyGyroMagstd        : num -0.187 -0.149 -0.226 -0.935 -0.979 ...
## $ tBodyGyroJerkMagmean   : num -0.299 -0.595 -0.295 -0.992 -0.995 ...
## $ tBodyGyroJerkMagstd    : num -0.325 -0.649 -0.307 -0.988 -0.995 ...
## $ fBodyAccmeanX          : num -0.2028 -0.4043 0.0382 -0.9796 -0.9952 ...
## $ fBodyAccmeanY          : num 0.08971 -0.19098 0.00155 -0.94408 -0.97707 ...
## $ fBodyAccmeanZ          : num -0.332 -0.433 -0.226 -0.959 -0.985 ...
## $ fBodyAccstdX           : num -0.3191 -0.3374 0.0243 -0.9764 -0.996 ...
## $ fBodyAccstdY           : num 0.056 0.0218 -0.113 -0.9173 -0.9723 ...
## $ fBodyAccstdZ           : num -0.28 0.086 -0.298 -0.934 -0.978 ...
## $ fBodyAccJerkmeanX      : num -0.1705 -0.4799 -0.0277 -0.9866 -0.9946 ...
## $ fBodyAccJerkmeanY      : num -0.0352 -0.4134 -0.1287 -0.9816 -0.9854 ...
## $ fBodyAccJerkmeanZ      : num -0.469 -0.685 -0.288 -0.986 -0.991 ...
## $ fBodyAccJerkstdX       : num -0.1336 -0.4619 -0.0863 -0.9875 -0.9951 ...
## $ fBodyAccJerkstdY       : num 0.107 -0.382 -0.135 -0.983 -0.987 ...
## $ fBodyAccJerkstdZ       : num -0.535 -0.726 -0.402 -0.988 -0.992 ...
## $ fBodyGyromeanX         : num -0.339 -0.493 -0.352 -0.976 -0.986 ...
## $ fBodyGyromeanY         : num -0.1031 -0.3195 -0.0557 -0.9758 -0.989 ...
## $ fBodyGyromeanZ         : num -0.2559 -0.4536 -0.0319 -0.9513 -0.9808 ...
## $ fBodyGyrostdX          : num -0.517 -0.566 -0.495 -0.978 -0.987 ...
## $ fBodyGyrostdY          : num -0.0335 0.1515 -0.1814 -0.9623 -0.9871 ...
## $ fBodyGyrostdZ          : num -0.437 -0.572 -0.238 -0.944 -0.982 ...
## $ fBodyAccMagmean        : num -0.1286 -0.3524 0.0966 -0.9478 -0.9854 ...
## $ fBodyAccMagstd         : num -0.398 -0.416 -0.187 -0.928 -0.982 ...
## $ fBodyBodyAccJerkMagmean : num -0.0571 -0.4427 0.0262 -0.9853 -0.9925 ...
## $ fBodyBodyAccJerkMagstd : num -0.103 -0.533 -0.104 -0.982 -0.993 ...
## $ fBodyBodyGyroMagmean   : num -0.199 -0.326 -0.186 -0.958 -0.985 ...
## $ fBodyBodyGyroMagstd    : num -0.321 -0.183 -0.398 -0.932 -0.978 ...
## $ fBodyBodyGyroJerkMagmean : num -0.319 -0.635 -0.282 -0.99 -0.995 ...
## $ fBodyBodyGyroJerkMagstd : num -0.382 -0.694 -0.392 -0.987 -0.995 ...
## - attr(*, "vars")=List of 1
## ..$ : symbol subject
## - attr(*, "drop")= logi TRUE

```

```

write.table(meanDF, file="tidy_data_set.txt", row.name=FALSE)

```