AW_HealthDataPrep_14Feb

me

14/02/2019

```
# Created by Bo Simango
# M.Sc. candidate
# Supervisor: Dr. Daniel Fuller
# Memorial University of Newfoundland
# Date: February 5, 2019
# Function: This code appends the Health Data Collector app data by participant ID (107-111) and plots
# Data source: Health Data Collector App - AW data
# Output: Appended CSVs, ggplot facet wrap and mice imputation
# First install all libraries used throughout this exercise
```

========================Libraries==============

```
library(knitr)
```

```
## Warning: package 'knitr' was built under R version 3.5.2
```

```
library(haven)
```

```
## Warning: package 'haven' was built under R version 3.5.2
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.5.2
```

```
library(plyr)
```

```
## Warning: package 'plyr' was built under R version 3.5.2
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.5.2
```

```
##
## 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
```

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 3.5.2
```

```r
library(stargazer)
```

```
## Warning: package 'stargazer' was built under R version 3.5.2
```

```
##
## Please cite as:
```

```
##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```r
library(missForest)
```

```
## Warning: package 'missForest' was built under R version 3.5.2
```

```
## Loading required package: randomForest
```

```
## Warning: package 'randomForest' was built under R version 3.5.2
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
## Loading required package: foreach
```

```
## Warning: package 'foreach' was built under R version 3.5.2
```

```
## Loading required package: itertools
```

```
## Warning: package 'itertools' was built under R version 3.5.2
```

```
## Loading required package: iterators
```

```
## Warning: package 'iterators' was built under R version 3.5.2
```

```r
library(mice)
```

```
## Warning: package 'mice' was built under R version 3.5.3
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'mice'
```

```
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```r
library(VIM)
```

```
## Warning: package 'VIM' was built under R version 3.5.3
```

```
## Loading required package: colorspace
```

```
## Warning: package 'colorspace' was built under R version 3.5.2
## Loading required package: grid

## Loading required package: data.table

## Warning: package 'data.table' was built under R version 3.5.2

##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## VIM is ready to use.
##  Since version 4.0.0 the GUI is in its own package VIMGUI.
##
##           Please use the package to use the new (and old) GUI.

## Suggestions and bug-reports can be submitted at: https://github.com/alexkowa/VIM/issues

##
## Attaching package: 'VIM'

## The following object is masked from 'package:datasets':
##
##     sleep
```

```r
library(magrittr)
```

```
## Warning: package 'magrittr' was built under R version 3.5.2
```

=======================Working Directory========

```r
# Set working directory
setwd("Z:/Research/dfuller/Walkabilly/studies/smarphone_accel/data/HealthData")
```

===================Read Health Data Collector App data=================

```r
# Read Health Data Collector App data
a_107 <- read.csv("Z:/Research/dfuller/Walkabilly/studies/smarphone_accel/data/HealthData/107/applewatc
a_108 <- read.csv("Z:/Research/dfuller/Walkabilly/studies/smarphone_accel/data/HealthData/108/applewatc
a_109 <- read.csv("Z:/Research/dfuller/Walkabilly/studies/smarphone_accel/data/HealthData/109/applewatc
a_110 <- read.csv("Z:/Research/dfuller/Walkabilly/studies/smarphone_accel/data/HealthData/110/applewatc
a_111 <- read.csv("Z:/Research/dfuller/Walkabilly/studies/smarphone_accel/data/HealthData/111/applewatc
```

===================Add participant ID column========

```r
# Add participant ID column
a_107$Participant_ID <- 107
a_108$Participant_ID <- 108
a_109$Participant_ID <- 109
```

```r
a_110$Participant_ID <- 110
a_111$Participant_ID <- 111
```

# ====================# Add activity column=============

```r
## Can combine Magrittr with case_when for this:  https://cran.r-project.org/web/packages/magrittr/inde

# Add activity column
a_107$activity
```

```
## NULL
```

```r
a_108$activity
```

```
## NULL
```

```r
a_109$activity
```

```
## NULL
```

```r
a_110$activity
```

```
## NULL
```

```r
a_111$activity
```

```
## NULL
```

# ====================# Append all participant ID CSVs==================

```r
# Append all participant ID CSVs
applewatch_all <- rbind (a_107, a_108, a_109, a_110, a_111)
```

# ====================# Add activity stage colum to csv ==================

```r
# Add activity stage colum to csv
applewatch_all$activity_stage <- ""
```

# ====================# Summary of statistical descriptives ==================

```r
# Summary of statistical descriptives
summary(applewatch_all)
```

```
##                  DateTime                               Username  
##  2018-12-13 13:13:00:  1   acceltrial2019@walkabilly.ca:365  
##  2018-12-13 13:14:00:  1                                     
##  2018-12-13 13:15:00:  1                                     
##  2018-12-13 13:16:00:  1                                     
##  2018-12-13 13:17:00:  1                                     
##  2018-12-13 13:18:00:  1                                     
##  (Other)            :359                                     
##      DeviceName      Heart           Calories            Steps       
##  AppleWatch:365   Min.   : 51.00   Min.   : 0.05627   Min.   :   1.0  
##                   1st Qu.: 78.89   1st Qu.: 0.06100   1st Qu.:  30.0  
##                   Median : 89.22   Median : 0.06100   Median :  71.0  
##                   Mean   : 96.53   Mean   : 2.85216   Mean   : 346.1  
##                   3rd Qu.:116.71   3rd Qu.: 0.64350   3rd Qu.: 637.0  
##                   Max.   :168.86   Max.   :21.53600   Max.   :1232.0  
##                   NA's   :265      NA's   :256        NA's   :334     
##     Distance       Participant_ID activity_stage    
##  Min.   :0.00069   Min.   :107    Length:365        
##  1st Qu.:0.00336   1st Qu.:108    Class :character  
##  Median :0.00447   Median :109    Mode  :character  
##  Mean   :0.03558   Mean   :109                      
##  3rd Qu.:0.02513   3rd Qu.:110                      
##  Max.   :0.69775   Max.   :111                      
##  NA's   :288
```

# ====================# Summary of statistical descriptives using Stargazer ==================

```r
# Summary of statistical descriptives using Stargazer
stargazer(applewatch_all, type = "text")
```

```
## 
## ======================================================================
## Statistic       N   Mean   St. Dev.  Min   Pctl(25) Pctl(75)    Max   
## ----------------------------------------------------------------------
## Heart          100 96.528   26.608  51.000  78.886  116.708   168.857 
## Calories       109  2.852   6.031   0.056   0.061    0.644     21.536 
## Steps          31  346.091 428.576  1.000   30.000  637.000  1,232.000
## Distance       77   0.036   0.092   0.001   0.003    0.025     0.698  
## Participant_ID 365 108.984  1.422    107     108      110       111   
## ----------------------------------------------------------------------
```

```r
# Add heart rate summary by participant

summary_heart <- applewatch_all %>%
              group_by(Participant_ID) %>%
                summarize(heart_mean = mean(Heart, na.rm = TRUE),
                          heart_sd = sd(Heart, na.rm =TRUE))
```

```r
# Add calories summary by participant
summary_calories <- applewatch_all %>%
              group_by(Participant_ID) %>%
                summarize(calories_mean = mean(Calories, na.rm = TRUE),
                          calories_sd = sd(Calories, na.rm =TRUE))
```

```
# Add steps summary by participant
summary_steps <- applewatch_all %>%
              group_by(Participant_ID) %>%
                summarize(steps_mean = mean(Steps, na.rm = TRUE),
                         steps_sd = sd(Steps, na.rm =TRUE))
```

```
## Add stages for each participant
if (applewatch_all$Participant_ID == "107"){

}
```

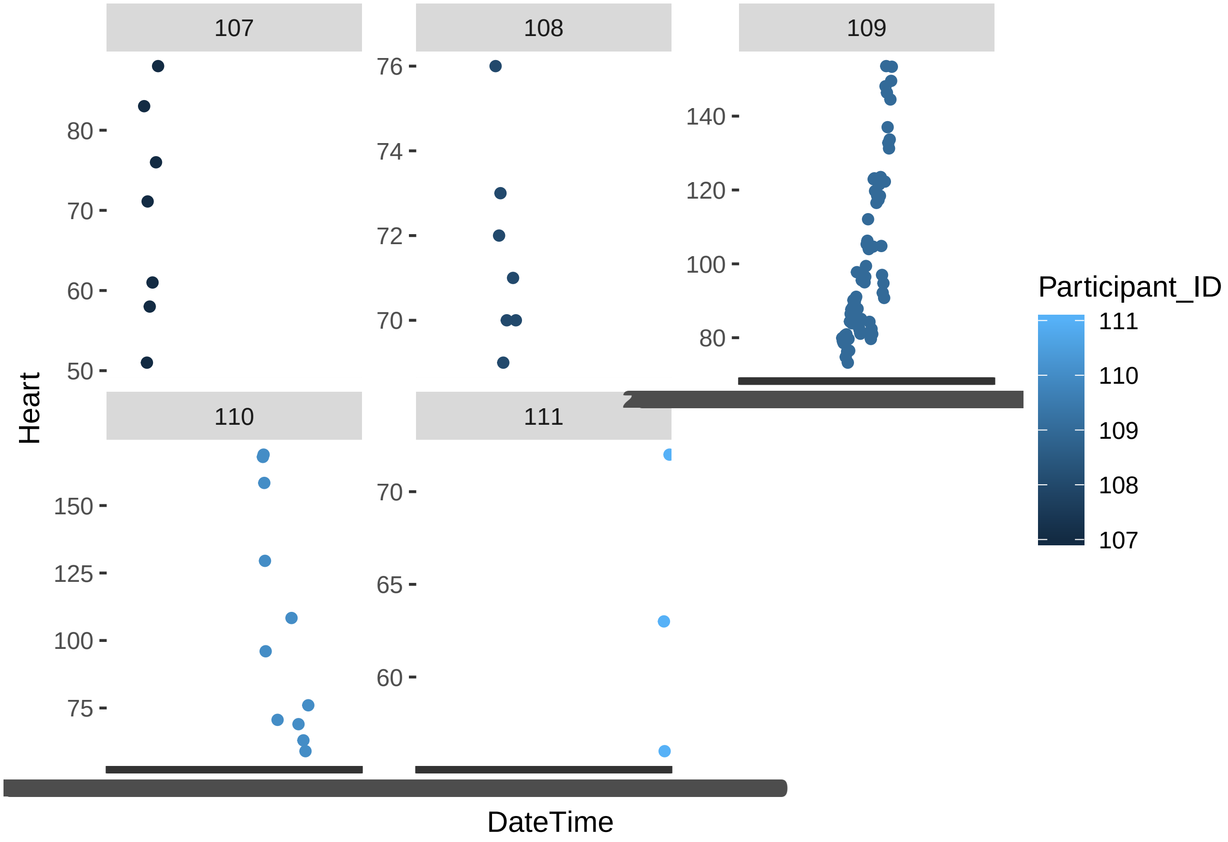
```
## Warning in if (applewatch_all$Participant_ID == "107") {: the condition has
## length > 1 and only the first element will be used

## NULL
```

Scatterplot of Heart rate vs. Time, Condensed Scatter Plot

```
heart_na <- ggplot(applewatch_all, aes(x = DateTime, y = Heart, colour=Participant_ID, group=Participant
  geom_point() +
facet_wrap(~Participant_ID, scales ="free_y" )
plot(heart_na)
```

```
## Warning: Removed 265 rows containing missing values (geom_point).
```



```
ggsave("heart_na.pdf")
```

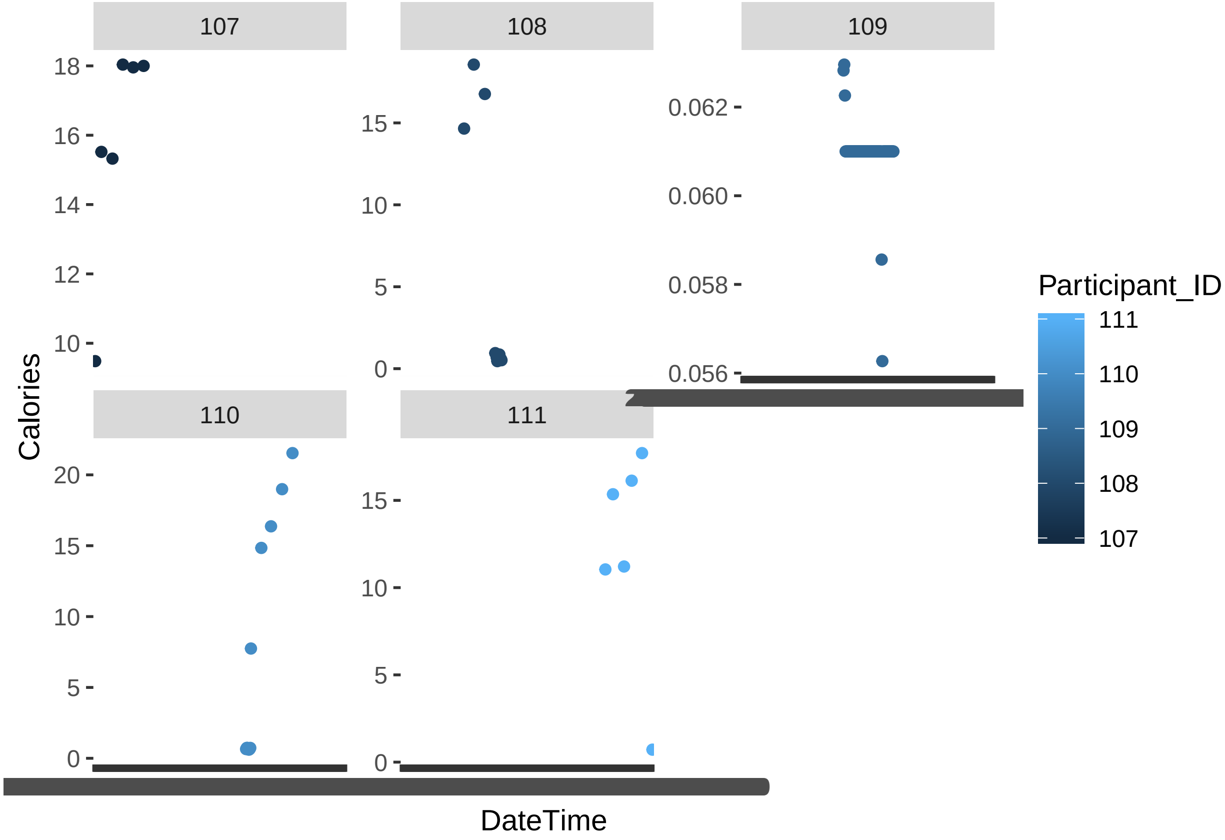
```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 265 rows containing missing values (geom_point).
```

Scatterplot of Calories vs. Time, Condensed Scatter Plot

```
ggplot(applewatch_all, aes(x = DateTime, y = Calories, colour=Participant_ID, group=Participant_ID)) +
  geom_point() +
facet_wrap(~Participant_ID, scales ="free_y")
```

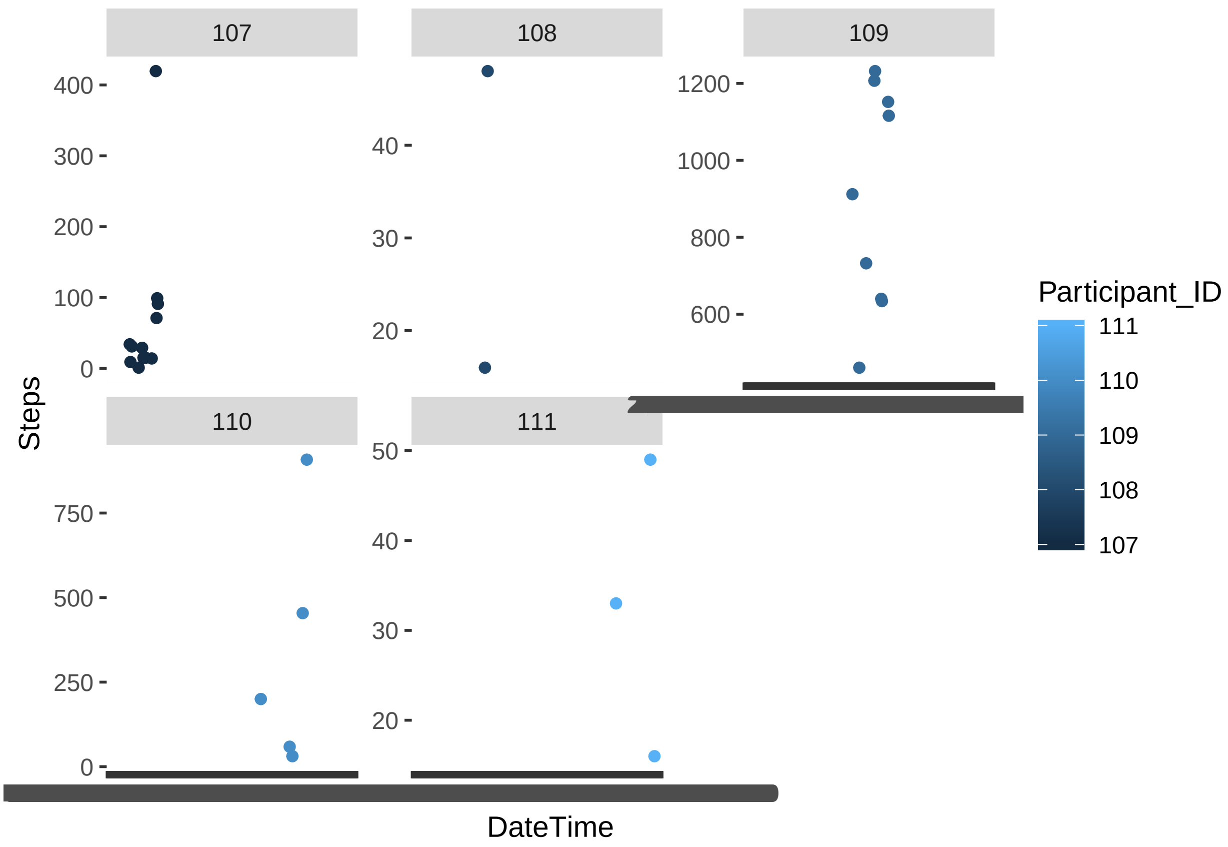
```
## Warning: Removed 256 rows containing missing values (geom_point).
```



Scatterplot of Steps vs. Time, Condensed Scatter Plot

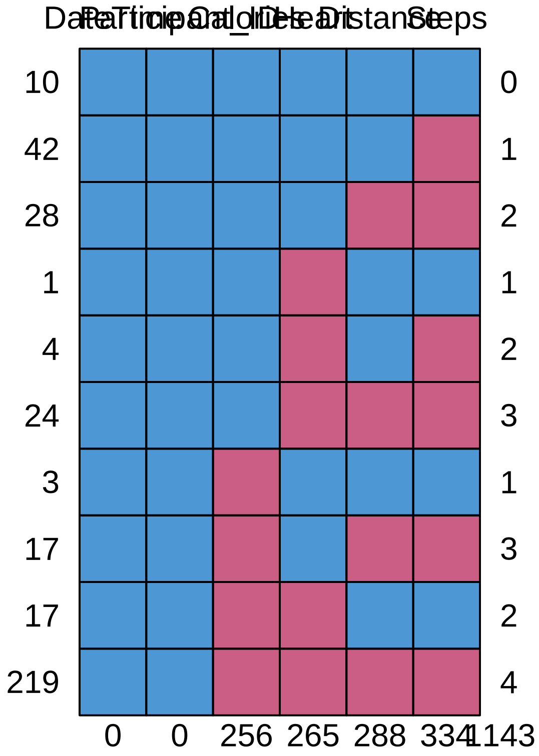
```
ggplot(applewatch_all, aes(x = DateTime, y = Steps, colour=Participant_ID, group=Participant_ID)) +
  geom_point() +
facet_wrap(~Participant_ID, scales ="free_y")
```

```
## Warning: Removed 334 rows containing missing values (geom_point).
```



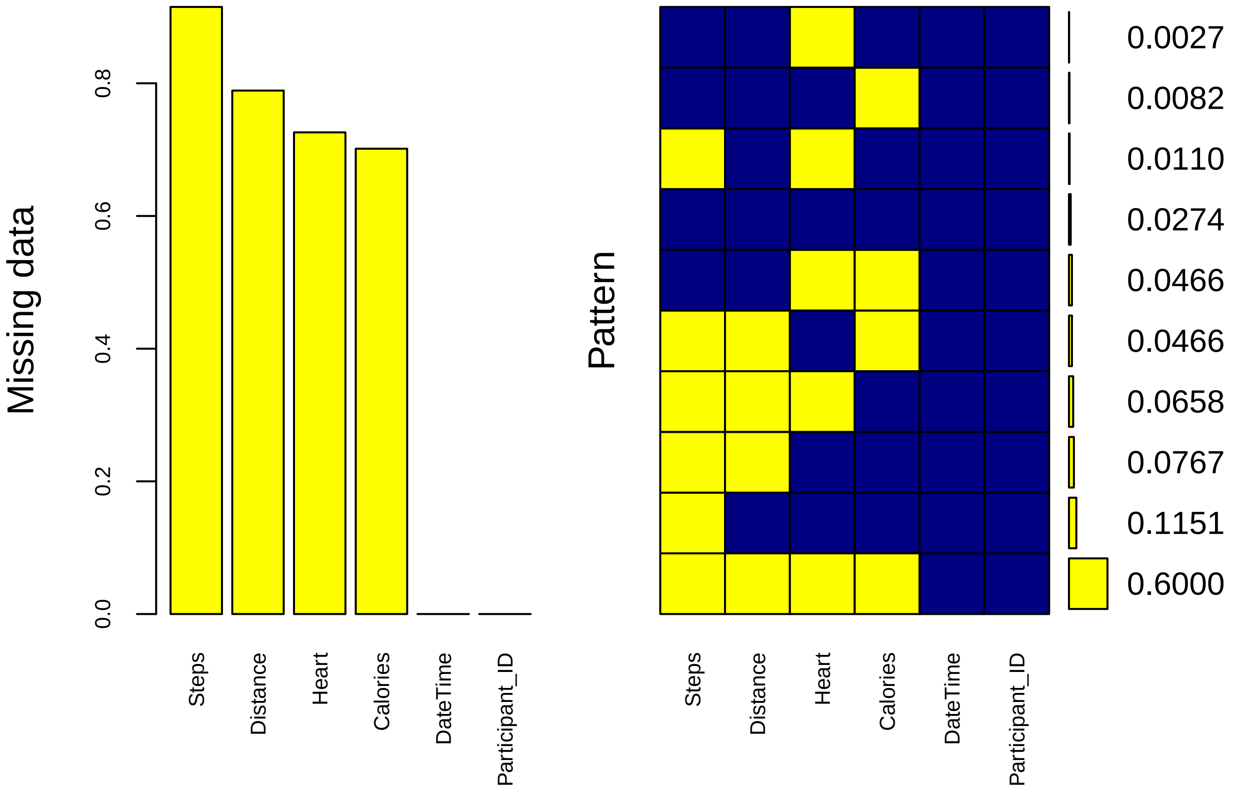
```
#Tabular form of missing value present in each variable in a data set.
health_vars <- select(applewatch_all, Heart, Calories, Steps, Distance, DateTime, Participant_ID)

md.pattern(health_vars)
```



```
##     DateTime Participant_ID Calories Heart Distance Steps
## 10         1              1        1     1        1     1    0
## 42         1              1        1     1        1     0    1
## 28         1              1        1     1        0     0    2
## 1          1              1        1     0        1     1    1
## 4          1              1        1     0        1     0    2
## 24         1              1        1     0        0     0    3
## 3          1              1        0     1        1     1    1
## 17         1              1        0     1        0     0    3
## 17         1              1        0     0        1     1    2
## 219        1              1        0     0        0     0    4
##            0              0      256   265      288   334 1143
```

```
#Visual representation of missing data
applewatch_all_plot <- aggr(health_vars, col=c('navyblue','yellow'),
                    numbers=TRUE, sortVars=TRUE,
                    labels=names(health_vars), cex.axis=.7,
                    gap=3, ylab=c("Missing data","Pattern"))
```



```
##
## Variables sorted by number of missings:
##        Variable     Count
##           Steps 0.9150685
##        Distance 0.7890411
##           Heart 0.7260274
##        Calories 0.7013699
##        DateTime 0.0000000
##  Participant_ID 0.0000000
```

```
#Missing data: 8% of steps data, 11% of heart and calories data
```

Impute the missing values with summary

- m - refers to 5 imputed data sets
- maxit - refers to number of iterations taken to impute missing values
- method-method used
- predictive mean matching

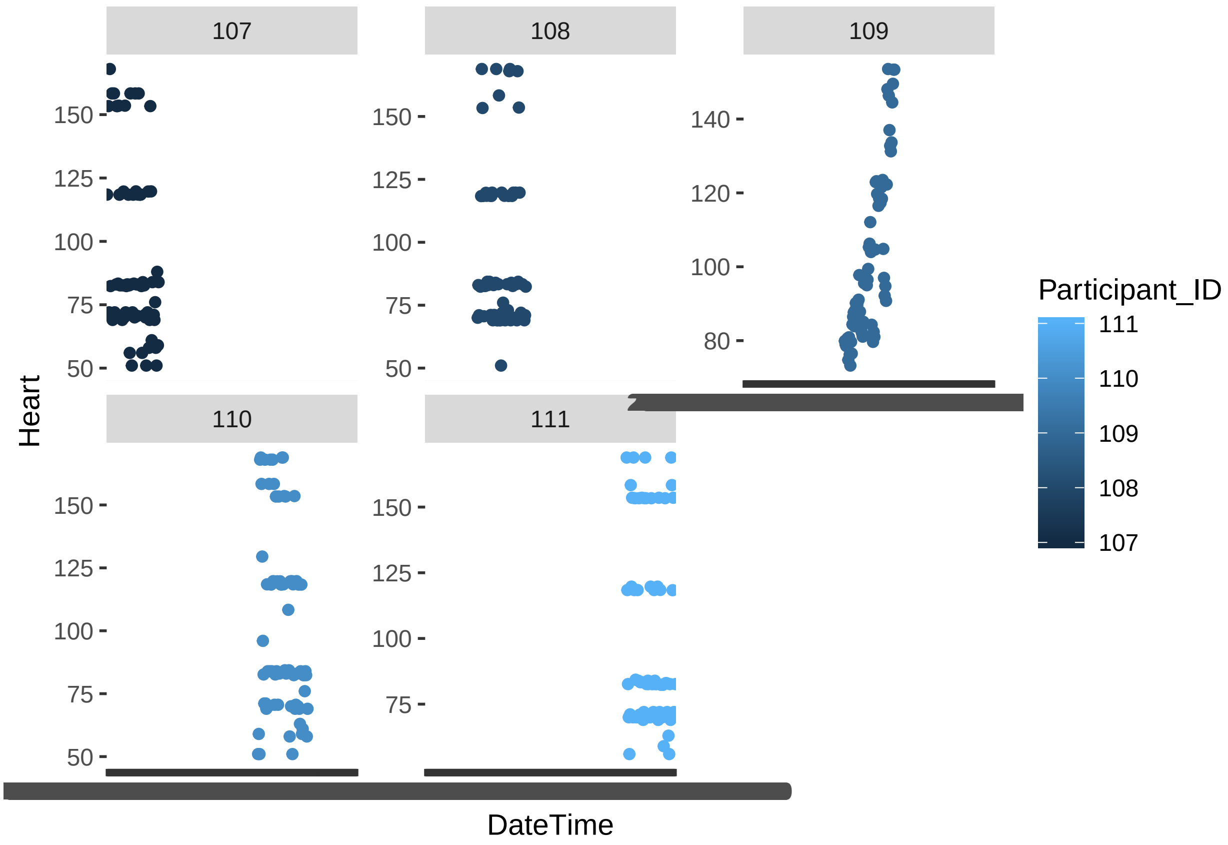
```
imputed_Data <- mice(health_vars, m = 5, maxit = 10, method = 'pmm', seed = 50)
```

```
##
##  iter imp variable
##   1   1  Heart*  Calories*  Steps*  Distance*
##   1   2  Heart*  Calories*  Steps*  Distance*
##   1   3  Heart*  Calories*  Steps*  Distance*
##   1   4  Heart*  Calories*  Steps*  Distance*
##   1   5  Heart*  Calories*  Steps*  Distance*
##   2   1  Heart*  Calories*  Steps*  Distance*
##   2   2  Heart*  Calories*  Steps*  Distance*
##   2   3  Heart*  Calories*  Steps*  Distance*
##   2   4  Heart*  Calories*  Steps*  Distance*
##   2   5  Heart*  Calories*  Steps*  Distance*
##   3   1  Heart*  Calories*  Steps*  Distance*
##   3   2  Heart*  Calories*  Steps*  Distance*
##   3   3  Heart*  Calories*  Steps  Distance*
##   3   4  Heart*  Calories*  Steps*  Distance*
##   3   5  Heart*  Calories*  Steps*  Distance*
##   4   1  Heart*  Calories*  Steps*  Distance*
##   4   2  Heart*  Calories*  Steps*  Distance*
##   4   3  Heart*  Calories*  Steps  Distance
##   4   4  Heart  Calories*  Steps*  Distance*
##   4   5  Heart*  Calories*  Steps*  Distance*
##   5   1  Heart*  Calories*  Steps*  Distance*
##   5   2  Heart*  Calories*  Steps*  Distance*
##   5   3  Heart*  Calories*  Steps*  Distance*
##   5   4  Heart  Calories*  Steps*  Distance*
##   5   5  Heart*  Calories*  Steps*  Distance*
##   6   1  Heart*  Calories*  Steps*  Distance*
##   6   2  Heart*  Calories*  Steps*  Distance*
##   6   3  Heart*  Calories*  Steps*  Distance*
##   6   4  Heart*  Calories*  Steps  Distance*
##   6   5  Heart*  Calories*  Steps*  Distance*
##   7   1  Heart*  Calories*  Steps*  Distance*
##   7   2  Heart*  Calories*  Steps*  Distance*
##   7   3  Heart*  Calories*  Steps*  Distance*
##   7   4  Heart  Calories*  Steps*  Distance*
##   7   5  Heart*  Calories*  Steps*  Distance*
##   8   1  Heart*  Calories*  Steps*  Distance*
##   8   2  Heart*  Calories*  Steps*  Distance*
##   8   3  Heart*  Calories*  Steps*  Distance*
##   8   4  Heart*  Calories*  Steps*  Distance*
##   8   5  Heart*  Calories*  Steps*  Distance
##   9   1  Heart*  Calories*  Steps*  Distance*
##   9   2  Heart*  Calories  Steps*  Distance*
##   9   3  Heart*  Calories*  Steps  Distance*
##   9   4  Heart*  Calories*  Steps*  Distance*
##   9   5  Heart*  Calories*  Steps*  Distance*
##   10   1  Heart*  Calories*  Steps  Distance*
##   10   2  Heart*  Calories*  Steps*  Distance*
##   10   3  Heart*  Calories*  Steps*  Distance*
##   10   4  Heart*  Calories*  Steps*  Distance*
##   10   5  Heart*  Calories*  Steps*  Distance*
##  * Please inspect the loggedEvents
```

```
## Warning: Number of logged events: 409
```

```r
completeData <- complete(imputed_Data, 2)

heart_impute <- ggplot(completeData, aes(x = DateTime, y = Heart, colour=Participant_ID, group=Participa
  geom_point() +
facet_wrap(~Participant_ID, scales ="free_y" )
plot(heart_impute)
```



```
#Check imputed Calories values
imputed_Data$imp$Calories
```

```
##            1      2      3        4      5
## 1    15.3250 14.845 14.654 11.04200 14.845
## 2    15.3510 15.351 14.654  7.74500 14.654
## 4    11.0420 15.325  7.745  0.94950 15.519
## 5     0.9495 14.654 11.211  0.94950 15.351
## 6    18.0370 15.351 11.042  7.74500 15.325
## 7    21.5360 15.351  9.484  0.94950 15.325
## 8    18.5590 14.845  9.484  0.94950 15.325
## 9     0.9495 15.351 11.211  0.94950 15.351
## 10   17.9990 14.845  9.484  0.86300 15.351
## 11   14.6540 15.351  9.484  0.73200 15.519
## 13   15.3250 15.325 11.211  0.94950 14.654
## 14   14.8450 15.519  9.484  0.73350 14.845
## 15    0.8630 15.351  7.745  0.94950 15.325
## 16    0.9495 15.519  9.484  0.94950 15.519
## 17   14.8450 15.519 11.042  0.73200 15.519
## 18   15.3250 14.654 11.042  0.86300 15.519
## 19   15.3250 15.325 14.654  0.94950 14.654
## 20   21.5360 14.654 14.654 14.65400 15.351
## 21   14.6540 15.325  7.745 14.65400 15.351
## 22  18.0370 15.325 11.211  7.74500 15.325
## 23  15.3250 15.519  9.484 11.04200 15.351
## 24  15.5190 15.519 11.211  9.48400 15.325
## 25  15.3250 14.654  7.745  0.94950 14.654
## 26   7.7450 14.845 11.211  0.73350 15.351
## 27   9.4840 15.351 11.211  9.48400 14.654
## 29   0.9495 15.519  9.484  0.94950 15.519
## 30  15.5190 15.351  9.484  0.94950 15.325
## 31  15.3250 15.351 14.654 14.65400 15.351
## 32  18.0370 14.845  9.484 11.04200 14.654
## 33   9.4840 15.325 14.654  9.48400 15.325
## 34  18.0370 15.351 11.042 18.98900 14.845
## 35  11.2110 14.654  9.484  7.74500 15.351
## 36  16.1280 15.325 11.042  9.48400 15.519
## 37  17.9990 15.519 14.654 18.98900 15.519
## 38   0.8630 15.325 14.654  0.94950 15.351
## 39  18.9890 14.845 11.211  0.94950 14.654
## 40  15.3510 15.519  7.745  0.94950 14.845
## 41  15.3510 15.519  7.745  0.94950 14.654
## 42  15.3250 15.519 11.042 11.21100 15.519
## 44  11.0420 14.845  7.745  0.86300 15.325
## 45  15.3250 15.351 14.654  7.74500 14.845
## 46  11.0420 14.845  9.484  0.73200 14.845
## 47   0.8630 14.654  9.484  0.05856 15.325
## 48   7.7450 14.845  9.484  7.74500 15.325
## 49  18.0370 15.519  9.484 11.21100 14.845
## 50  15.3510 15.325  9.484  7.74500 14.654
## 51   7.7450 15.325 11.211  0.73200 15.325
## 52  18.9890 14.654  9.484 18.03700 15.325
## 53  15.5190 15.325 14.654 11.04200 15.325
## 54  14.6540 14.654 11.211 18.55900 14.845
## 55  15.3250 15.351 11.042  0.94950 15.351
## 56  18.0370 14.654  7.745  0.94950 15.325
## 57  16.1280 15.519  7.745 18.98900 15.351
## 59  21.5360 15.351 11.211  0.94950 14.654
## 60   0.9495 15.325 14.654  0.73350 14.845
## 61  21.5360 15.325 11.211 11.04200 14.845
## 62  14.6540 15.325  7.745  0.94950 14.845
## 63  15.5190 15.351  7.745  7.74500 15.351
## 64   9.4840 14.845  7.745  0.73200 15.519
## 65  18.9890 15.325 11.211  0.73200 15.519
## 66  15.3250 15.519 11.211 18.98900 14.845
## 67   0.8630 15.325  7.745  9.48400 15.325
## 68  15.3510 15.519  9.484  0.94950 15.351
## 69  11.0420 15.519 11.211  0.73200 15.519
## 70  15.3510 15.351 11.211  0.94950 14.654
## 71   7.7450 14.845 14.654  9.48400 14.845
## 72  18.5590 15.351 11.042 17.99900 14.845
## 74  17.9990 15.519 14.654 17.99900 15.325
## 75  17.9990 14.845  9.484 18.03700 15.325
## 76  11.0420 15.325 14.654 11.21100 15.325
## 77  18.5590 15.519  9.484  0.94950 14.654
## 78  15.5190 14.654  7.745  0.73200 15.519
## 79  14.6540 15.351 11.211  0.86300 15.519
## 80   0.7335 15.351 14.654  7.74500 15.519
## 81  15.3250 15.519  9.484  9.48400 15.325
## 82  15.3510 15.519  9.484  0.73200 14.654
## 83  14.8450 15.325 11.211 11.04200 15.519
## 84  16.1280 14.654 14.654  7.74500 15.351
## 85  11.2110 15.519 11.042 11.04200 15.351
## 86  14.8450 15.351 11.042 11.21100 14.654
## 87   0.9495 14.845  9.484  0.73200 15.351
## 88  11.2110 15.351  7.745 14.65400 14.654
## 89  14.8450 15.351 11.211  0.73200 15.351
## 90   7.7450 15.519 14.654 11.04200 15.519
## 91  15.3250 15.325  9.484  0.86300 15.519
## 93  15.3250 14.654 11.211  7.74500 14.845
## 94  15.5190 14.845 11.042  7.74500 15.519
## 95  15.3510 15.325 11.211 14.65400 15.519
## 96  15.3250 14.654 11.042  9.48400 14.654
## 97  11.2110 15.519  7.745 11.04200 14.845
## 98  15.3510 14.845 14.654  0.73350 14.845
## 99  14.6540 15.351 11.211 11.04200 15.351
## 100 17.9990 14.654  7.745 11.21100 14.845
## 101 16.1280 15.325 14.654  0.94950 15.351
## 102 16.3690 15.519  7.745  7.74500 14.654
## 103 15.3250 15.519  9.484 11.04200 15.351
## 104 15.3510 15.351 14.654  7.74500 15.519
## 105  9.4840 15.325 14.654 11.04200 14.845
## 107 14.6540 15.351  9.484 18.55900 15.325
## 108 15.3250 14.845 11.042 11.04200 14.654
## 109 18.9890 15.519  7.745  0.94950 15.351
## 110 18.5590 15.325 11.042  7.74500 15.351
## 111 18.9890 15.325  7.745 18.55900 15.325
## 112 14.8450 14.654 11.211  0.94950 14.845
## 113 15.5190 15.519  7.745  9.48400 15.519
## 114 14.8450 15.351 14.654  0.73200 15.351
## 115  9.4840 14.845 11.211  0.86300 14.654
## 116 18.5590 14.654 11.042  0.73200 14.845
## 117 15.3250 15.351 11.042 11.21100 15.351
## 118 14.6540 14.845 11.042  0.86300 15.351
## 119 18.5590 15.519  7.745  0.94950 15.351
## 120 15.3250 14.845 14.654  7.74500 15.351
## 121 16.1280 14.654 11.211  9.48400 14.845
## 123 14.6540 15.519 14.654  0.94950 14.654
## 124 11.2110 15.351 11.211  7.74500 15.351
## 125 15.3250 14.654 11.211  9.48400 15.351
## 126  0.9495 14.654 11.211 11.04200 15.519
## 127 17.9990 15.325 11.211 11.04200 15.325
## 128 16.1280 14.654 11.211  0.86300 15.519
## 129 15.3250 15.351 14.654  0.73200 15.519
## 130 18.9890 14.654 14.654  0.73350 14.654
## 131  0.7320 15.519  7.745  0.73200 15.519
## 132 15.3510 15.325  9.484  9.48400 14.654
## 133 11.0420 15.351  9.484 11.21100 15.325
## 134 11.2110 15.351 11.211  0.94950 14.845
## 135 15.3510 15.519 11.211  9.48400 14.845
## 136  7.7450 14.654 14.654  0.86300 15.325
## 138 15.3250 15.351 14.654  0.73350 15.351
## 147 18.9890 15.325 11.042  9.48400 15.325
## 229 18.9890 15.325  9.484  0.94950 15.325
## 230 18.9890 14.654 11.211  0.73200 14.654
## 231 14.6540 15.519  9.484  9.48400 14.845
## 232 14.8450 15.519 14.654  0.73350 15.325
## 233 15.3510 15.519 14.654  0.94950 14.654
## 234 15.3510 15.325  7.745  0.94950 14.654
## 235 11.0420 15.325 11.042  0.73350 15.519
## 236 18.9890 15.519  9.484  0.73200 14.845
## 237 15.3510 15.519  9.484  0.73200 14.845
## 238 17.9990 14.654  7.745  9.48400 14.845
## 239  9.4840 15.351 14.654  7.74500 15.519
## 240 18.5590 14.845 11.042  0.86300 15.325
## 241 11.2110 15.325 11.211  0.86300 15.351
## 242 15.3510 14.845  9.484  7.74500 15.351
## 244 11.0420 14.654  9.484  0.94950 14.845
## 245 15.5190 15.325  9.484  9.48400 15.351
## 246 18.0370 15.351  7.745  0.73200 14.654
## 247 15.5190 15.351 11.211  0.94950 15.519
## 248 18.5590 15.325 11.042 11.21100 14.845
## 249  0.9495 14.654  9.484  0.86300 15.325
## 250 14.6540 14.845  9.484 11.04200 15.351
## 251 11.2110 15.325 11.042  0.73350 15.325
## 252 15.3250 15.351  7.745  7.74500 14.845
## 253 18.5590 15.519 14.654 11.04200 15.519
## 254  9.4840 15.325 14.654  0.94950 15.325
## 255  0.7320 15.519  7.745 11.04200 14.654
## 256 16.1280 15.519 11.211  0.94950 14.845
## 258 14.8450 15.351  7.745  0.86300 15.325
## 259 14.8450 15.519  7.745  7.74500 14.654
## 260 11.0420 14.845  9.484 11.21100 14.654
## 261 15.5190 14.654 11.042 11.21100 15.351
## 262 16.1280 15.351 11.042  0.73200 14.654
## 263 15.3250 15.351 14.654 11.04200 15.325
## 264 18.5590 14.654 11.042  0.73200 14.845
## 265  0.7335 15.325  9.484  7.74500 14.845
## 266 14.8450 14.654  9.484 11.21100 14.845
## 267 21.5360 15.519  9.484 21.53600 15.325
## 268 21.5360 15.519  9.484 11.04200 15.351
## 269  0.8630 14.845 14.654  7.74500 14.654
## 270 15.5190 15.519  7.745  7.74500 15.519
## 271 17.9990 15.351  9.484 21.53600 15.519
## 272 17.9990 15.325  7.745 11.21100 15.351
## 274 14.6540 15.519 11.042  0.73200 15.351
## 275 21.5360 14.845 11.042  0.73350 15.519
## 276  9.4840 14.845  9.484 11.21100 15.325
## 277 11.0420 15.519 14.654  0.73200 15.519
## 278 11.0420 15.325  7.745  0.73200 14.654
## 279 11.2110 14.654  9.484 11.21100 14.845
## 280 18.0370 14.654 11.211  0.86300 15.519
## 281 17.9990 15.519 11.211 11.04200 15.351
## 282  9.4840 15.351 11.211 11.04200 14.654
## 283 21.5360 15.519 11.211  0.73350 15.325
## 284 14.6540 15.519 11.211  0.94950 14.845
## 285 18.9890 15.519  7.745  0.73200 15.351
## 286 17.9990 14.654  7.745 18.55900 15.519
## 287 11.2110 15.519  7.745  7.74500 15.519
## 289 15.5190 15.325  7.745  9.48400 15.325
## 290 11.0420 15.519 11.042  7.74500 14.845
## 291 15.5190 15.325  9.484 11.04200 15.351
## 292 21.5360 14.845  7.745 18.03700 15.325
## 293 18.9890 14.654  9.484  0.94950 15.519
## 294 14.6540 15.519 11.211 11.21100 15.351
## 295 15.3250 15.351 11.042 11.21100 15.325
## 297 18.0370 14.845  7.745  0.94950 14.845
## 298 17.9990 15.519 14.654 18.98900 15.519
## 299 17.9990 15.351  9.484  0.73200 15.519
## 300 15.3510 15.325  9.484  7.74500 15.351
## 301 18.0370 15.519  9.484 11.21100 15.351
## 302  7.7450 15.351 11.042  9.48400 14.845
## 303 17.9990 14.845 14.654  9.48400 15.519
## 304 15.3510 14.654  9.484  0.73350 14.654
## 305 14.8450 14.654 14.654  7.74500 14.845
## 306  7.7450 15.351 14.654  7.74500 14.845
## 308 16.1280 15.325 11.211  0.73350 14.654
## 309 18.5590 15.351 11.211 11.21100 15.351
## 310 15.3510 15.519 11.042  0.86300 15.519
## 311 11.2110 14.845 11.211  0.86300 14.845
## 312 14.6540 14.654  7.745 11.04200 15.519
## 313 21.5360 15.519 11.042  7.74500 15.351
## 314 11.0420 14.845 14.654  0.94950 14.845
## 315  9.4840 15.325 14.654  0.94950 14.845
## 316 11.2110 15.351  9.484  0.94950 14.654
## 317 18.9890 15.351 11.211  0.73350 15.351
## 318  0.9495 15.351 14.654  9.48400 14.845
## 319 11.2110 15.325  9.484  0.73200 14.654
## 320 14.6540 14.654 11.211 11.04200 15.351
## 321 17.9990 14.654 11.042  9.48400 14.654
## 322 18.9890 15.351 11.211  0.86300 15.519
## 324 11.0420 15.325 11.042  7.74500 14.654
## 325 16.1280 14.654  9.484 11.21100 15.351
## 326  9.4840 14.845 11.042  0.73200 15.325
## 327 18.5590 15.325 11.211 11.04200 15.351
## 328 16.1280 14.845 14.654  0.73350 15.325
## 329 11.0420 14.654  7.745 11.21100 15.519
## 330 16.3690 15.325 11.042 11.21100 14.845
## 331 14.6540 15.519 11.042  7.74500 15.351
## 332  0.7320 15.325 14.654  7.74500 14.845
## 333 14.6540 15.325 11.042 11.21100 15.351
## 335 14.6540 15.351  7.745 11.04200 15.519
## 336  9.4840 15.519 11.042 11.21100 15.519
## 337 17.9990 15.519  9.484  0.94950 15.519
## 338  7.7450 14.845  9.484  0.86300 14.845
## 339 18.5590 15.325 11.042  0.73200 14.845
## 340 18.0370 15.325 11.042 11.21100 14.845
## 341 15.3510 15.519 14.654  0.73200 14.654
## 342 21.5360 15.351 11.042  7.74500 14.654
## 343 15.3250 15.519 14.654  7.74500 14.845
## 344 14.8450 15.351 11.211  0.94950 14.845
## 345 15.3510 14.654  9.484 11.04200 14.654
## 346 18.9890 14.845 14.654 11.04200 14.654
## 347 14.6540 15.325  9.484  0.73200 15.519
## 348 18.5590 15.519 14.654 18.55900 15.519
## 350 11.0420 15.325  9.484  0.94950 15.325
## 351 15.3250 14.845 14.654  9.48400 15.351
## 352 11.2110 14.845 11.042  7.74500 15.519
## 353 18.0370 14.845 11.042  0.86300 14.654
## 354 14.8450 15.351  7.745 18.03700 15.519
## 355  0.8630 15.325 14.654  7.74500 15.325
## 356 14.6540 15.351 11.042  7.74500 15.519
## 357 18.9890 14.845 14.654  0.73350 15.325
## 358 15.3510 15.351  9.484  0.73200 14.845
## 359 14.6540 15.325  9.484  0.86300 14.845
## 360 15.3250 15.519 11.042  0.94950 15.519
## 361  7.7450 14.654  9.484 11.21100 15.325
## 362 11.2110 14.845 11.042  0.73200 15.351
## 363 15.5190 15.351 11.211  0.73200 14.654
## 365 15.3250 14.654 11.211 11.04200 14.845
```

```
#Get complete data ( 2nd out of 5)
```

```
#build predictive model
fit <- with(data = completeData, exp = lm(Heart ~ Calories + Steps))
summary(fit)
```

```
##
## Call:
## lm(formula = Heart ~ Calories + Steps)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -49.76 -25.71 -13.61  21.64  72.20
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 100.923508   3.587856  28.129   <2e-16 ***
## Calories     -0.269389   0.263114  -1.024    0.307
## Steps         0.001521   0.010793   0.141    0.888
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.39 on 362 degrees of freedom
## Multiple R-squared:  0.003165,   Adjusted R-squared:  -0.002343
## F-statistic: 0.5746 on 2 and 362 DF,  p-value: 0.5634
```

```
completeData$fitted <- fit$fitted.values
```

setwd(“Z:/Research/dfuller/Walkabilly/studies/smarphone_accel/data/HealthData/Bo’s working folder”) write.csv (applewatch_all, file=‘applewatch_all.csv’) “‘