Practicum 1

Peter Claussen Ben Derenge Stephanie Leibl

9/16/2021

Overview

We have been given an Excel data file with data collected for a study from a call center. These data include health metrics from **employees** gathered over an eight month period.

Variables that are of particular interest have been highlighted in the original data. These include variables relating to weight change and demographics. Specifically, we will consider in this preliminary proposal the variables

Highlighted in yellow

- `shift`
- `Total_Met_Min`

Highlighted in orange

- `gender`
- `Age`
- `height`
- `weightgain`
- `lbs_gained`
- `BMI`
- `Vig.ex.Time`
- `Mod.ex.time`
- `Walk.ex.Time`

We will use the convention that text rendered in `sans serif` font denote variable or column names found in the original data set, or derived variables calculated from columns in the original data, while *italics* will denote real-world processes or phenomena of interest. Thus, `weightgain` denotes the data column in the original data file, while *weight gain* denotes some measure of the change in weight by individuals over the study period.

. We have been tasked to provide an analysis to address two specific aims:

- **(SA1)** Does *total metabolic minutes* have an effect on *weight gain*?
- **(SA2)** Does *shift* have an effect on *weight gain*?

We will refer to as **SA1** and **SA2** in further discussion. We address *total metabolic minutes*, *weight gain* and *shift* in the following sections.

Total MET minutes

Total MET Minutes is a composite measure obtained from survey responses (International Physical Activity Questionnaire (IPAQ) short form) to estimate overall physical activity. Physical activity is divided into categories described as *vigorous*, *moderate* and *walking*.
Total Metabolic Minutes can be calculated from 3 data columns. We are given the formula

```
Total_met_min = 8*Vig_ex_time + 4*Mod_ex_time + 3.3*Walk_ex_time
```

to calculate *Total MET Minutes* from the data.

`Total_met_min` contains many missing values. We will create a new data column, `CalcTMM` that is calclulated from existing data columns as given in the formula above. To visualize the number of missing values, we plot missing values as 0 in the following figure.

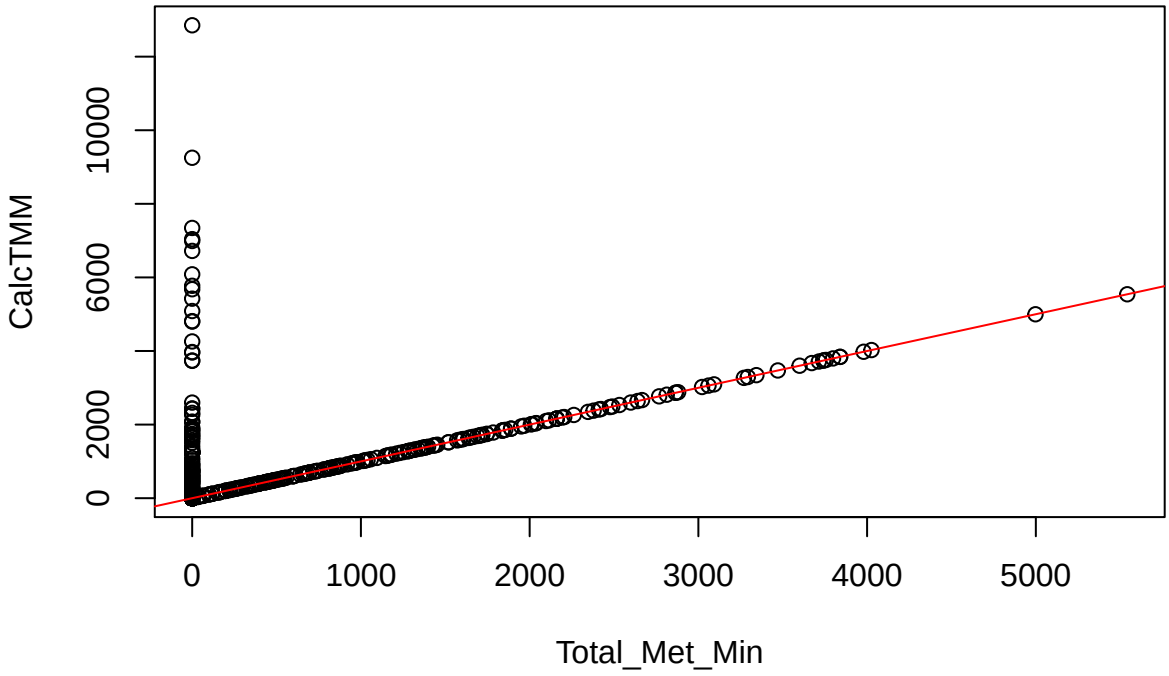


Figure 1: Total MET Minutes calculated from data, plotted against the data column title Total_Met_Min

We can see that the original data column `Total_met_min` has a large number of missing values. If we use this data column as given, we will reduce the number of observations in the analysis. This graph also confirms that the formula given above was used to compute `Total_met_min` from the data.

Shift

Shift takes values of the form `7am`, `8am`, ..., `2pm`, `other`. There are some columns that have missing values for `shift`. We propose that *shift* be modeled as an ordinal data type, with missing values grouped with `other` and `other` takes an ordinal value greater than `2pm`.

```
metrics.dat$shift[metrics.dat$shift==''] <- 'other'
shift.levels <- c(paste(c(7:11),'am',sep=''),paste(c(12,1:2),'pm',sep=''),'other')
metrics.dat$shift <- factor(metrics.dat$shift,shift.levels)
summary(metrics.dat$shift)
```

```
##   7am   8am   9am  10am  11am  12pm   1pm   2pm other
##    31   115    56    50    44    14     8    15    19
```

Weight Gain

While the data include a variable (data column) named `weightgain`, we are asked to consider other reponse variables, including change in weight (`lbs_gained`) and change in BMI. Thus, our first task is to determine the appropriate response variable. The choice of response variable will dictate both choice of statistical method (i.e. logistic regression vs linear regression) and methods for data cleaning.

weightgain (Binomial Response)

`weightgain` in the original data contains missing values.

```
par(mfrow=c(1,2))
plot(weightgain ~ CalcTMM,data=metrics.dat)
plot(weightgain ~ shift,data=metrics.dat)
```

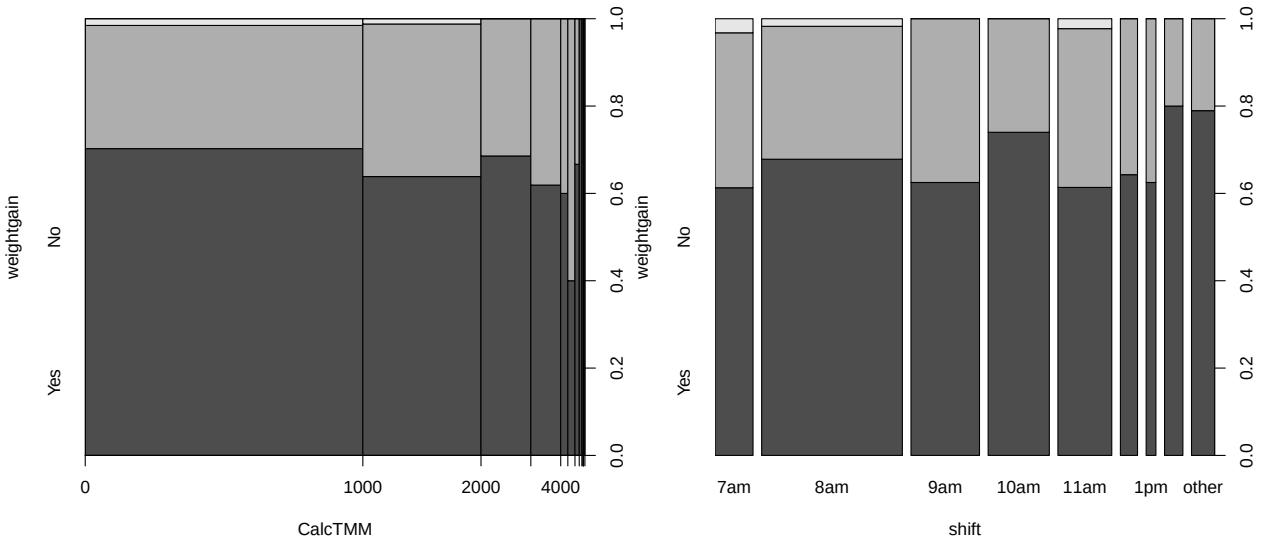


Figure 2: `weightgain` versus Total MET Minutes (calculated) and Shift

At first glance, this appears to be a logistic regression - the response variable (weightgain) should be binary (Yes/No). However, we are asked to consider alternate response variables (BMI, pounds gained) as markers for the general response "weight gain', as opposed to the specific data variable 'weightgain'

Pounds gained (Continuous Response)

Change in BMI

BMI at the end of the study period has been recorded, while initial BMI has not. We can, however, calculate an initial BMI from other data columns.

"Beginning BMI is unknown. The beginning weight can be calculated by body weight at 8 months –pounds gained. Then this value along with the height variable can be used to calculate beginning BMI; not that this calculation differs whether using kg/meters or pounds/inches"

```
# Calculate body weight at 8 months
metrics.dat['beginning_bweight'] <- metrics.dat$bweight - metrics.dat$pounds_gained

# Need to account for instances where body weight is not entered
metrics.dat$beginning_bweight[metrics.dat$beginning_bweight<='0'] <- '0'

# Take a look at all BMI values
unique(metrics.dat$BMI)
```

```
##   [1] 21.79 22.46 22.68    NA 25.12 35.95 24.40 21.48 26.58 25.11 25.77 26.50
##  [13] 31.15 38.97 28.35 26.83 24.21 34.86 20.00 37.20 26.57 31.17 31.87 31.09
##  [25] 41.09 21.13 40.41 33.45 30.95 27.32 27.26 33.73 29.27 22.85 32.77 24.45
##  [37] 24.03 23.40 28.12 32.61 25.51 37.59 21.95 29.95 25.10 28.40 25.54 17.97
##  [49] 20.60 25.79 19.97 27.25 22.66 30.41 22.31 27.09 38.16 27.19 21.03 46.17
##  [61] 23.38 30.04 25.84 22.15 24.37 21.82 20.90 24.27 19.76 27.46 27.61 22.59
##  [73] 36.91 33.90 30.54 20.42 25.09 29.53 43.82 21.30 28.50 33.20 25.80 22.71
##  [85] 36.39 19.20 20.36 25.01 20.66 19.48 41.74 27.37 37.29 26.63 37.69 26.26
##  [97] 30.81 26.62 30.43 41.62 32.48 22.13 26.38 18.46 39.45 22.39 35.44 31.47
## [109] 21.77 33.11 39.06 24.41 29.34 27.44 33.72 27.28 21.70 20.39 29.44 33.83
## [121] 25.74 22.96 30.89 31.42 27.88 23.67 38.73 16.31 28.69 26.08 35.14 37.30
## [133] 32.00 29.29 22.80 26.96 26.97 23.03 19.13 36.21 32.06 21.28 20.37 39.93
## [145] 27.80 25.24 18.79 32.07 39.15 35.70 27.43 29.18 23.12 24.39 28.48 24.89
## [157] 34.43 30.85 26.12 31.48 23.75 44.38 22.92 32.89 21.62 25.85 22.27 34.20
## [169] 28.79 23.57 26.44 48.25 30.82 17.77 20.52 39.17 20.40 22.04 31.01 22.83
## [181] 29.99 25.18 21.87 21.41 43.45 27.45 32.41 32.32 29.05 24.96 21.45 22.67
## [193] 41.72 23.24 33.44 42.53 35.07 37.97 29.98 23.43 32.28 28.29 30.38 26.09
```

```
# Calculate beginning BMI -- internet tells me BMI=(weight/height^2) * 703
# Since height values ~60/65, I am assuming they are inches not meters
metrics.dat['beginning_bweight'] <- as.numeric(metrics.dat$beginning_bweight)
metrics.dat['beginning_BMI'] <- (metrics.dat$beginning_bweight / (metrics.dat$height)^2)*703

# Take a look at all beginning_BMI values
unique(metrics.dat$beginning_BMI)
```

```
##   [1]       NA 19.38554 24.12663 25.10187 23.05363 25.79944 27.11295 31.52784
##   [9] 25.60354 25.41078 19.36639 18.99662 33.65331 24.79718 29.64641 29.15606
##  [17] 37.55001 34.98727 30.40657 24.36358 30.50207 27.83306 21.96321 22.28655
##  [25] 21.45386 23.12089 22.67132 36.01916 18.28824 25.42066 21.46438 17.13822
##  [33] 19.73755 18.63574 23.59183 26.60575 26.30965 32.49557 21.63077 19.20265
##  [41] 44.45239 28.31909 23.62529 20.96744 18.24364 23.49076 25.74463 20.98026
##  [49] 26.57845 29.83410 27.24948 21.14168 41.80437 24.58699 27.33889 19.57563
##  [57] 25.91626 20.59570 18.59788 38.61813 30.51215 24.02832 25.31631 28.79332
##  [65] 27.38487 36.31015 19.04362 29.53161 28.47801 21.29799 25.72434 27.36592
##  [73] 21.72669 17.27599 37.30204 21.61149 29.95048 24.20799 24.79219 21.41042
##  [81] 23.08594 25.01492 28.12608 25.82185 31.56327 26.03704 18.81916 27.87570
##  [89] 31.63866 23.56510 25.05811 27.46094 35.50505 26.54184 23.98651 24.32526
##  [97] 33.66604 28.30966 25.79053 24.10286 18.30296 19.76427 19.83774 37.43787
## [105] 25.22974 22.88700 22.49454 17.69637 27.89129 31.32101 33.63965 23.77471
## [113] 21.62580 21.12124 22.23776 29.93374 27.12191 23.68506 30.12857 16.45942
## [121] 24.45941 29.12854 23.17017 21.46442 30.53798 20.91847 19.85055 31.45578
## [129] 16.30493 44.92544 25.50567 37.73245 21.28460 19.39429 28.89542 19.99644
## [137] 25.10714 19.19555 27.09977 28.36273 27.91988 23.29467 20.25244 26.31757
## [145] 19.79878 24.62006 22.29347 30.11669 39.74509 32.54630 22.31746 26.49673
```

```
## [153] 22.03412 24.95858 24.89965
```

```
# ignore non-positive values
metrics.dat$beginning_BMI[is.nan(metrics.dat$beginning_BMI)] <- 0
metrics.dat$beginning_BMI[!is.finite(metrics.dat$beginning_BMI)] <- 0
```

Look at BMI in the beginning vs at the end

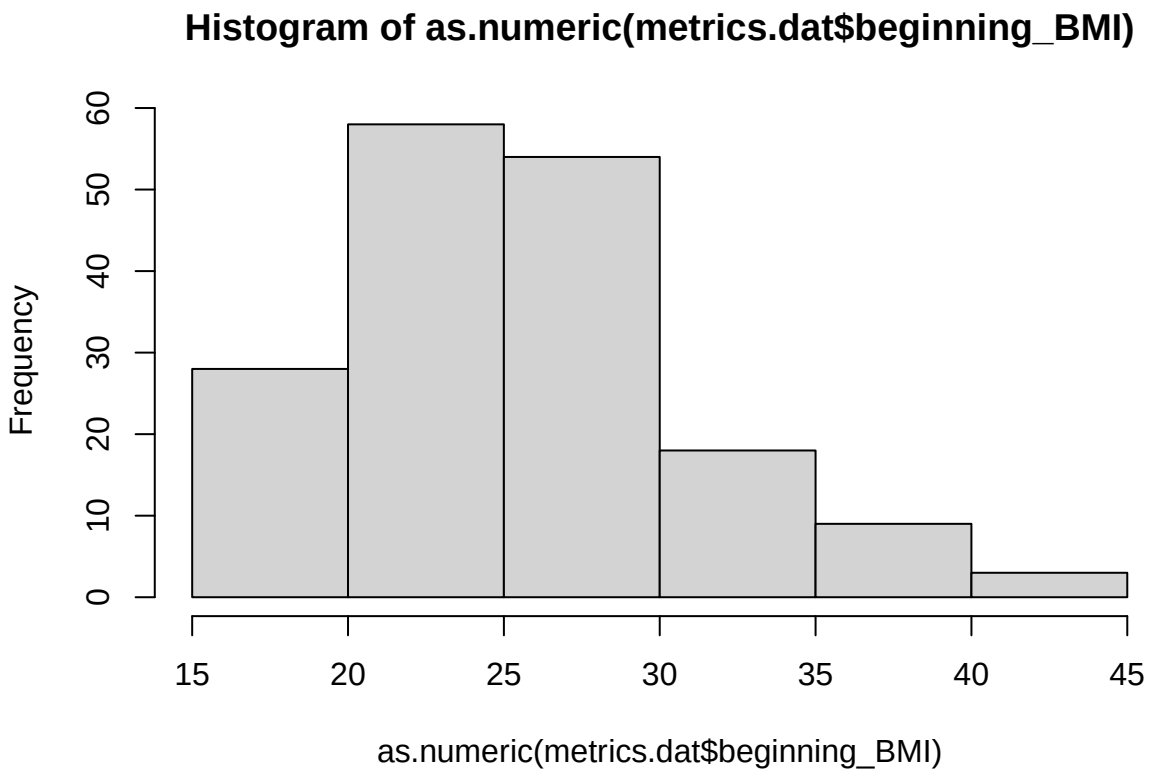
```
# convert to numeric for calculations
metrics.dat$beginning_BMI <- as.numeric(metrics.dat$beginning_BMI)

# Are the means different? --Yes, BMI is bigger than beginning_BMI
mean(metrics.dat$beginning_BMI)
```

```
## [1] 12.35762
```

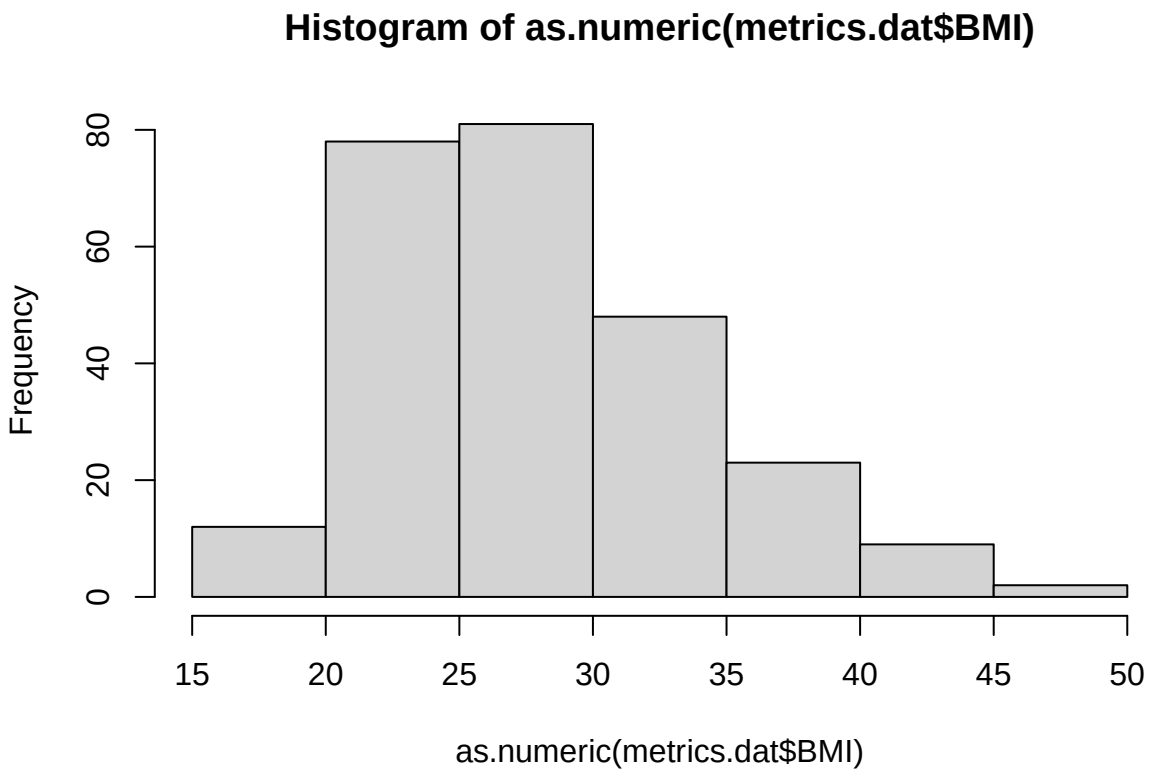
```
mean(metrics.dat$BMI)
```

```
## [1] NA
```

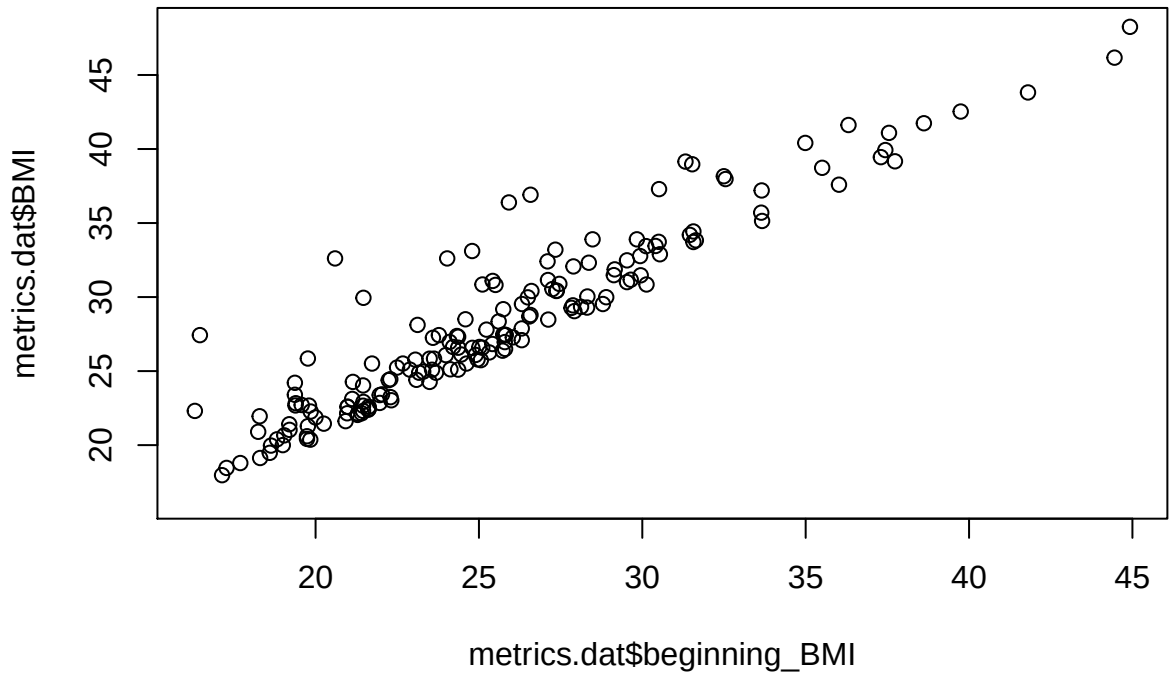
```
# Visualize the columns
metrics.dat$beginning_BMI[metrics.dat$beginning_BMI==0] <- NA
hist(as.numeric(metrics.dat$beginning_BMI))
```




```
metrics.dat$BMI[metrics.dat$BMI==0] <- NA
hist(as.numeric(metrics.dat$BMI))
```



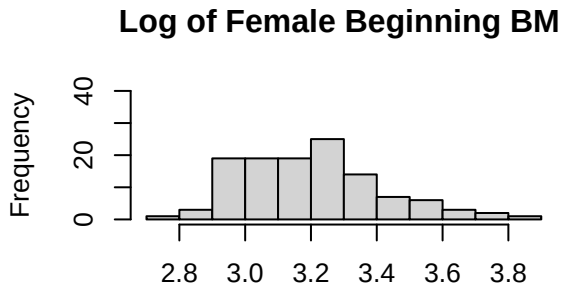

```
# Look at relationship between BMI and beginning BMI -- appears that BMI only increases or stays the sa
plot(metrics.dat$beginning_BMI, metrics.dat$BMI)
```

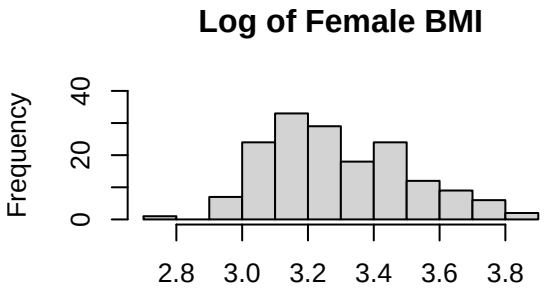


I want to see if gender affects BMI

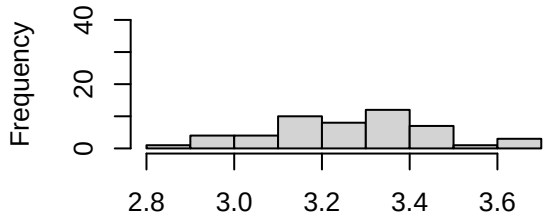
```
# To view all plots together
par(mfrow=c(2,2))

# Female Beginning BMI
```

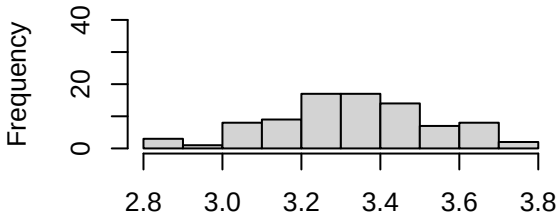
```
hist(log(metrics.dat[which(metrics.dat$gender=='Female'), 'beginning_BMI']), main="Log of Female Beginn

# Female BMI
hist(log(metrics.dat[which(metrics.dat$gender=='Female'), 'BMI']), main="Log of Female BMI", ylim=c(0,4

# Male Beginning BMI
hist(log(metrics.dat[which(metrics.dat$gender=='Male'), 'beginning_BMI']), main="Log of Male Beginning

# Male BMI
hist(log(metrics.dat[which(metrics.dat$gender=='Male'), 'BMI']), main="Log of Male BMI", ylim=c(0,40))
```





ics.dat[which(metrics.dat$gender == "Female"), "begi(metrics.dat[which(metrics.dat$gender == "Female"),





trics.dat[which(metrics.dat$gender == "Male"), "begin(metrics.dat[which(metrics.dat$gender == "Male"), '

Data Review

There are many missing rows in the data. While complete data screening will not be determined until we've chosen both response and predictor variables, we can perform some prelimiries.

We have inspected the data in Excel format manually, and have exported to CSV to read into R for analysis. For convenience, we have identified the columns of interest.

Two orange highlighted columns ('lbs_gained', 'pounds_gained') appear to be identical. We will include only `lbs_gained`

Also from inspection, the column `Snumber` appears to uniquely identify observations, and when this column is empty, the remaining columns are also empty. Thus, we will use this to in a first pass to screen for missing data.

```
summary(metrics.dat[,c(yellow,orange)])
```

```
##      shift     Total_Met_Min       gender              Age
```

```
##  8am    :115   Min.   :   0.0   Length:352         Min.   :19.00
##  9am    : 56   1st Qu.:   0.0   Class :character   1st Qu.:26.00
##  10am   : 50   Median : 357.8   Mode  :character   Median :31.00
##  11am   : 44   Mean   : 797.4                      Mean   :33.76
##  7am    : 31   3rd Qu.:1315.5                      3rd Qu.:40.00
##  other  : 19   Max.   :5542.0                      Max.   :64.00
##  (Other): 37                                       NA's   :30
##      height      weightgain   lbs_gained         BMI         Vig.ex.Time
##  Min.   :57.00      :  4    Min.   : 0.00   Min.   :16.31   Min.   :  0.00
##  1st Qu.:64.00   No :111    1st Qu.: 8.00   1st Qu.:22.85   1st Qu.:  0.00
##  Median :66.00   Yes:237    Median :15.00   Median :26.62   Median : 27.00
##  Mean   :66.64              Mean   :16.76   Mean   :27.85   Mean   : 76.16
##  3rd Qu.:69.00              3rd Qu.:20.00   3rd Qu.:31.42   3rd Qu.:120.00
##  Max.   :82.00              Max.   :70.00   Max.   :48.25   Max.   :840.00
##  NA's   :21                 NA's   :120     NA's   :99
##   Mod.ex.time       Walk.ex.Time
##  Min.   :   0.00   Min.   :   0.00
##  1st Qu.:   0.00   1st Qu.:   9.75
##  Median :  30.00   Median :  60.00
##  Mean   :  73.85   Mean   : 122.92
##  3rd Qu.:  90.00   3rd Qu.: 136.25
##  Max.   :1680.00   Max.   :1680.00
##  NA's   :1
```

There are two responses, gender and weightgain, that should be binomial. However, there are three levels.

```
metrics.dat$gender[metrics.dat$gender==''] <- 'Missing'
metrics.dat$weightgain[metrics.dat$weightgain==''] <- 'Missing'
```

```
## Warning in `[<-.factor`(`*tmp*`, metrics.dat$weightgain == "", value =
## structure(c(2L, : invalid factor level, NA generated
```

```
unique(metrics.dat$gender)
```

```
## [1] "Female"  "Male"    "Missing"
```

```
unique(metrics.dat$weightgain)
```

```
## [1] No   Yes  <NA>
## Levels:  No Yes
```

```
metrics.dat$shift <- factor(metrics.dat$shift)
metrics.dat$gender <- factor(metrics.dat$gender)
metrics.dat$weightgain <- factor(metrics.dat$weightgain)
```

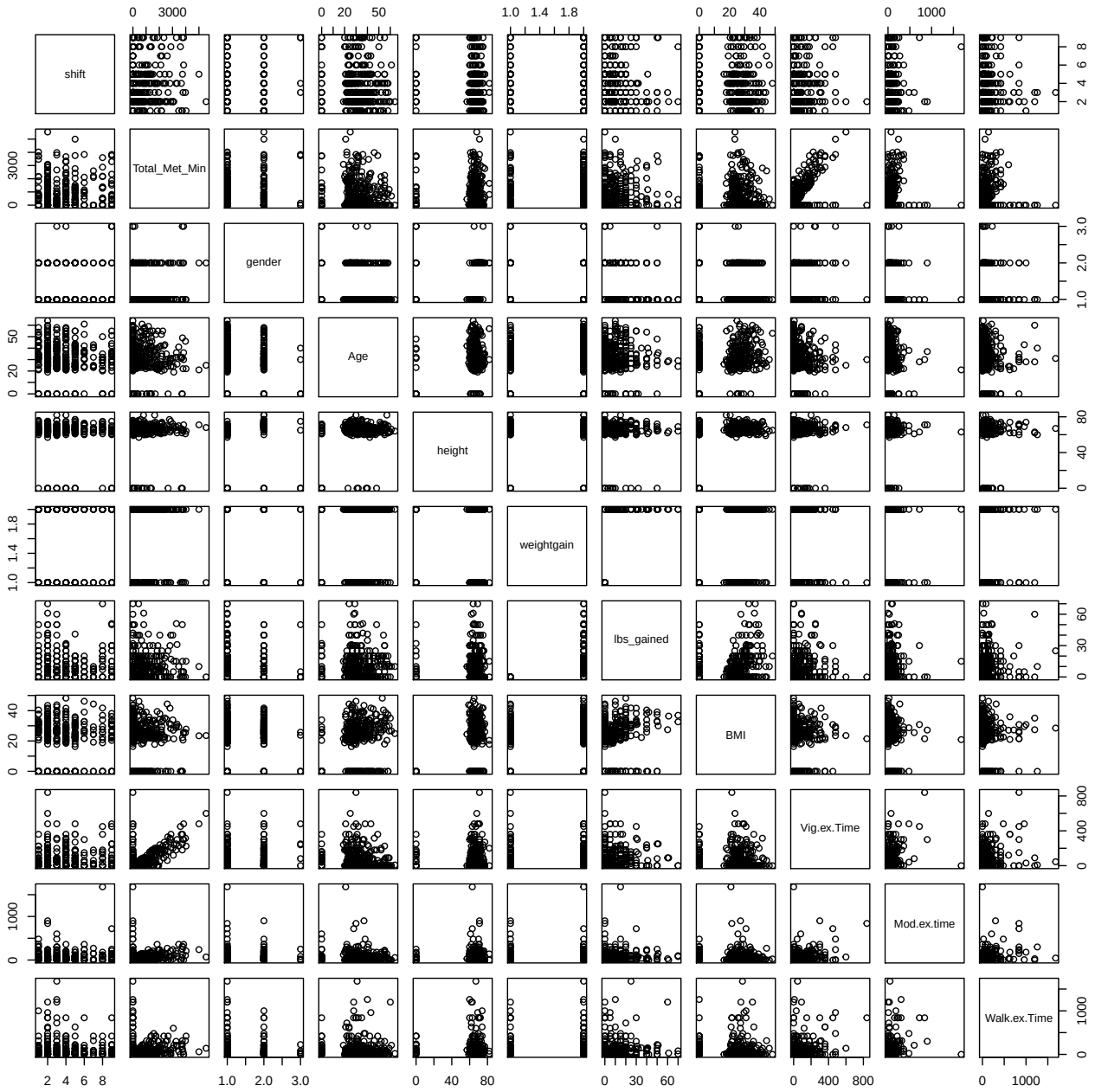
Other Predictor Variables

TODO

```
metrics.dat[is.na(metrics.dat)] <- 0
```

```
## Warning in `[<-.factor`(`*tmp*`, thisvar, value = 0): invalid factor level, NA
## generated
```

```
pairs(metrics.dat[,c(yellow,orange)])
```



Points for consideration

Selection of response variable.

We have proposed three response variables of interest as surrogates for weight gain, specifically

Scope of analysis

The questions of interest only reference two variables, while there are multiple highlighted data columns. The first suggests a single predictor variable, but it would be best to perform multiple regression