

Answer Key for EDA

Ted Laderas

May 19, 2016

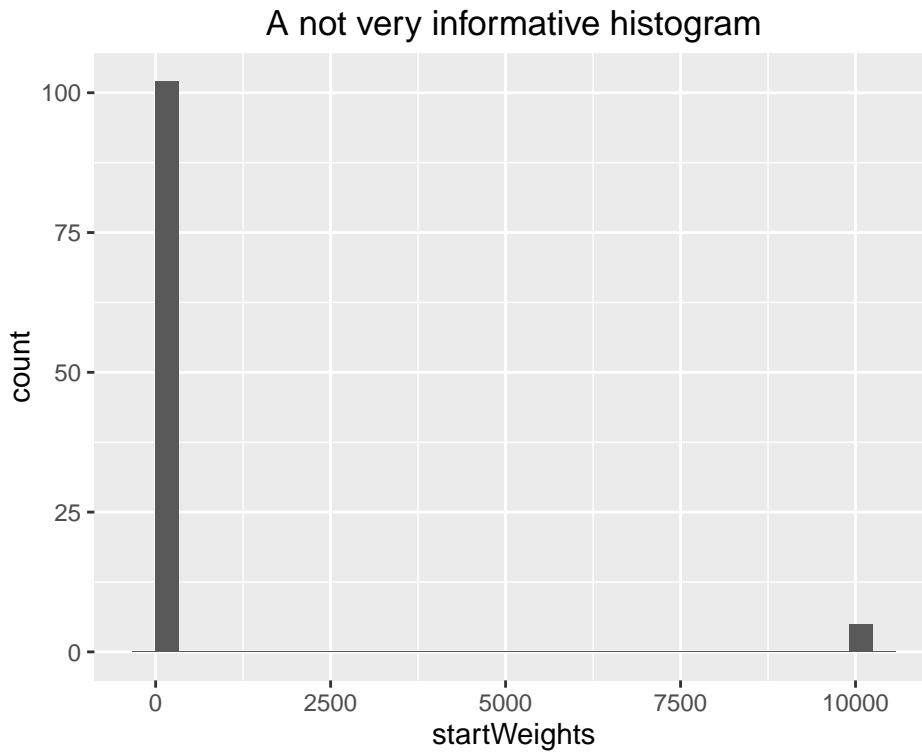
Dataset A

There are two things to notice for dataset A. The first can be noticed by looking at the data dictionary. The value of 9999 indicates an error on the scale. If they did not catch it via the data dictionary, then they will notice it when they look at the summaries, histogram and boxplots.

```
library(ggplot2)
library(dplyr)
datasetA <- read.delim("data/datasetA.txt",row.names = 1)
#show summary of data
summary(datasetA)
```

```
##   startWeights      endWeights      treatment      timeElapsed
##   Min.       : 91.83   Min.       : 85.65   Control    :54   Min.       :10.00
##   1st Qu.: 129.10   1st Qu.: 124.50   Treatment:53   1st Qu.:25.00
##   Median : 154.30   Median : 150.70                      Median :35.00
##   Mean   : 610.49   Mean   : 607.04                      Mean   :35.61
##   3rd Qu.: 177.65   3rd Qu.: 169.70                      3rd Qu.:48.00
##   Max.    :9999.00   Max.    :9999.00                      Max.    :60.00
##   staffID1 staffID2   gender      age
##   N1:33     N1:33     female:60   Min.    :20.00
##   N2:31     N2:31     male   :37   1st Qu.:23.00
##   N3:43     N3:43     NA's   :10   Median :26.00
##                                     Mean    :25.36
##                                     3rd Qu.:28.00
##                                     Max.    :30.00
```

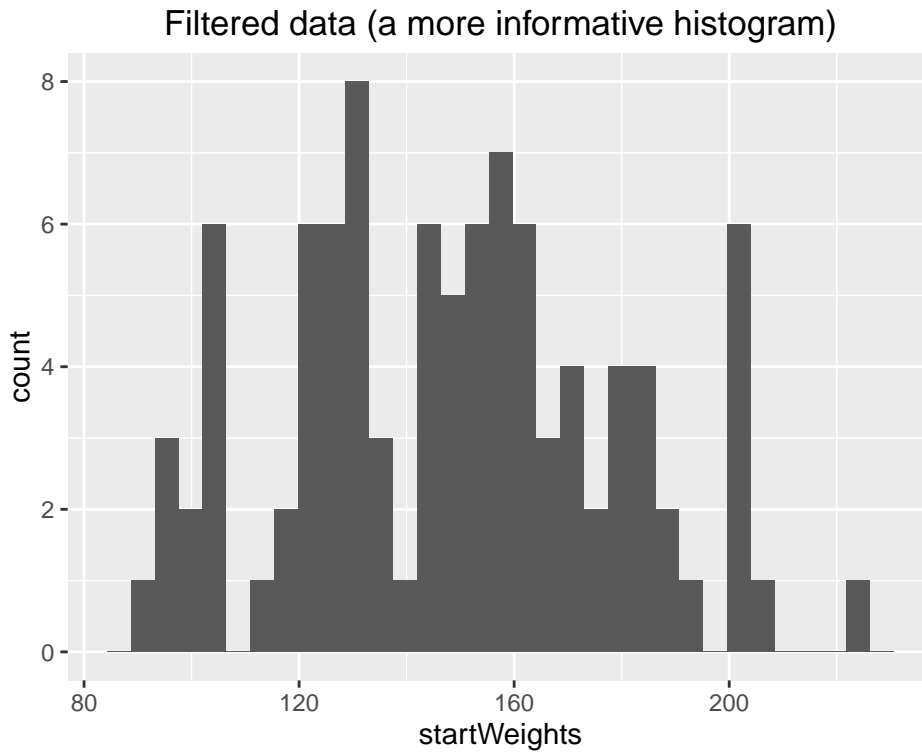
```
#show histogram of startWeights - is not very informative because of error value!
ggplot(datasetA, aes(x=startWeights)) + geom_histogram() +
  ggtitle("A not very informative histogram")
```



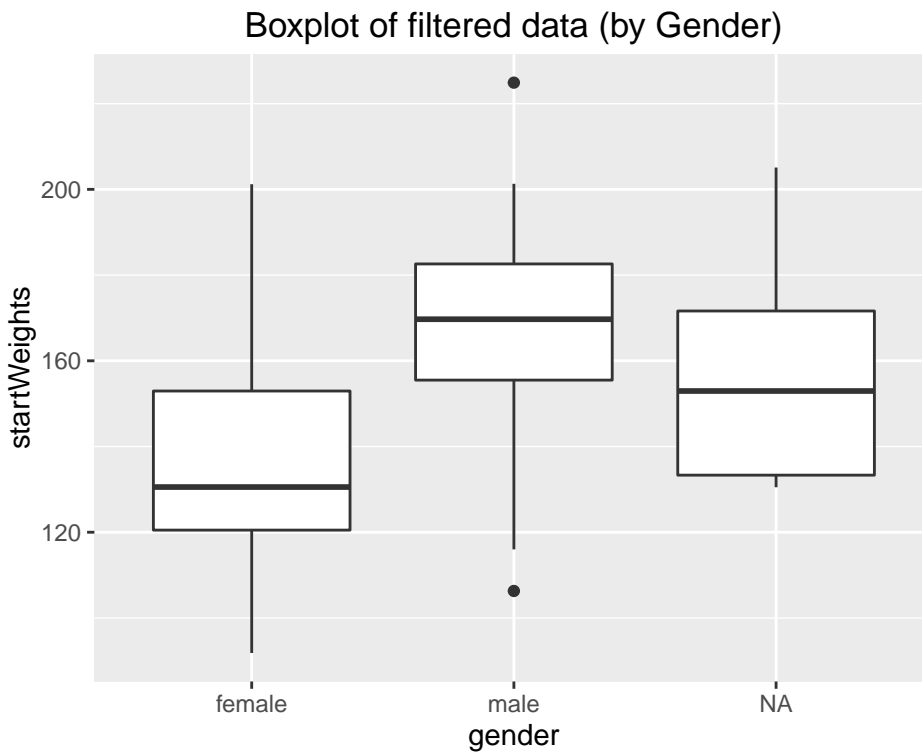
So the first step is remove rows with these values. After removing those rows with error values, the histogram looks much more informative.

```
#do filtering to remove those lines with error values
datasetA <- datasetA %>% filter(startWeights != 9999 & endWeights != 9999)

#do the histogram again
ggplot(datasetA, aes(x=startWeights)) + geom_histogram() +
  ggtitle("Filtered data (a more informative histogram)")
```



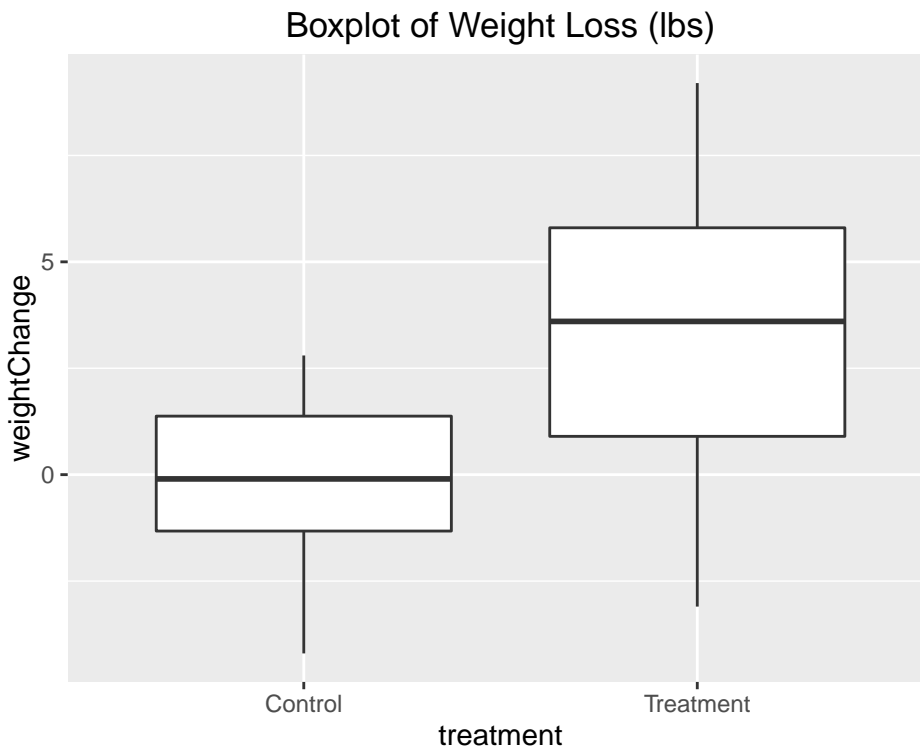
#Also there is a difference in weight among genders (most easily seen in boxPlots)
`ggplot(datasetA, aes(x=gender, y=startWeights)) + geom_boxplot() +
 ggtitle("Boxplot of filtered data (by Gender)")`



The second thing to notice is that the data needs to be transformed somehow in order to see the weight loss effect. Here I define a new variable called `weightChange`, which is just `startWeights - endWeights`. Because `timeElapsed` is variable, I also try to scale `weightChange` by the `timeElapsed` in order to look for a possible linear relationship between the two.

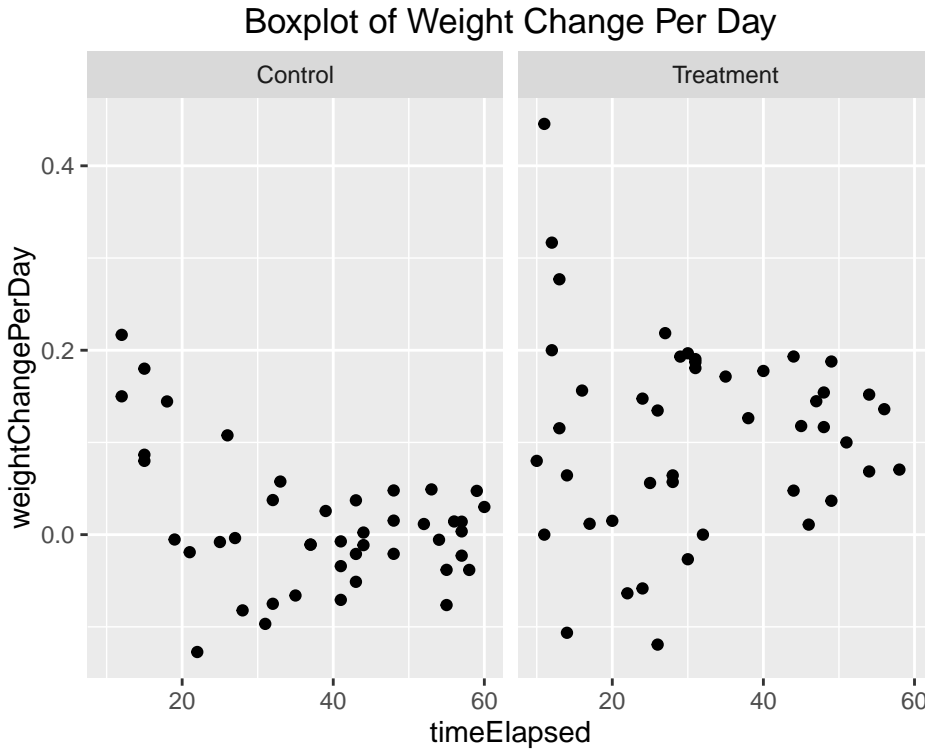
An optional step is to omit NAs, which occur in gender.

```
datasetA <- datasetA %>% mutate(weightChange = startWeights - endWeights,  
                                weightChangePerDay = weightChange / timeElapsed)  
datasetA <- na.omit(datasetA)  
  
#show that there is a weightChange effect in those treated  
#versus those who are not  
ggplot(datasetA, aes(x = treatment, y = weightChange)) + geom_boxplot() +  
  ggtitle("Boxplot of Weight Loss (lbs)")
```



There is a slight linear relationship between `timeElapsed` and `weightChangePerDay` if we condition by `treatment`, but it can't really be seen by eye due to noise in the data.

```
ggplot(datasetA, aes(x = timeElapsed, y = weightChangePerDay)) + geom_point() +  
  facet_wrap(facets = c("treatment")) + ggtitle("Boxplot of Weight Change Per Day")
```



Dataset B

Dataset B is a little trickier. Reading the data dictionary, we notice that the units for `startWeight` and `endWeight` are in kg, so that complicates comparing the two datasets. Also, there are patients who do not have a second measurement. Referring to the data dictionary, these patients are ones who dropped out of the study and are thus uninformative to our research question, so we'll remove them.

```
datasetB <- read.csv("data/datasetB.csv", row.names= 1)
#show summary before
summary(datasetB)
```

```
##      age      gender  treatment  startWeight
## Min.   :64.00  Female:30   Control :30   Min.    : 51.00
## 1st Qu.:69.00  Male  :42   Treatment:42  1st Qu.: 51.00
## Median :73.50
## Mean   :74.75
## 3rd Qu.:80.00
## Max.   :88.00
##
##      endWeight  timeElapsed  staffID1 staffID2
## Min.    : 57.50  Min.     :30.00  S1: 9    S1:22
## 1st Qu.: 77.58  1st Qu.:40.00  S2:20    S2:23
## Median : 86.90  Median :55.00  S3:21    S4:27
## Mean    : 84.79  Mean    :54.75  S4:22
## 3rd Qu.: 93.17  3rd Qu.:67.50
## Max.    :115.00  Max.     :80.00
## NA's     :14
```

```
datasetB <- na.omit(datasetB)
#show summary after removing NAs
summary(datasetB)
```

```
##      age      gender      treatment      startWeight
## Min.   :64.00  Female:24   Control  :24   Min.    : 51.00
## 1st Qu.:69.25  Male   :34   Treatment:34  1st Qu.: 51.00
## Median :73.50                                     Median : 83.25
## Mean   :75.05                                     Mean   : 76.81
## 3rd Qu.:80.00                                     3rd Qu.: 92.22
## Max.   :88.00                                     Max.   :118.00
##      endWeight      timeElapsed      staffID1 staffID2
## Min.    : 57.50   Min.      :30.00   S1: 6    S1:19
## 1st Qu.: 77.58   1st Qu.:38.00   S2:18    S2:18
## Median : 86.90   Median :54.00   S3:17    S4:21
## Mean    : 84.79   Mean     :53.81   S4:17
## 3rd Qu.: 93.17   3rd Qu.:67.00
## Max.    :115.00   Max.      :80.00
```

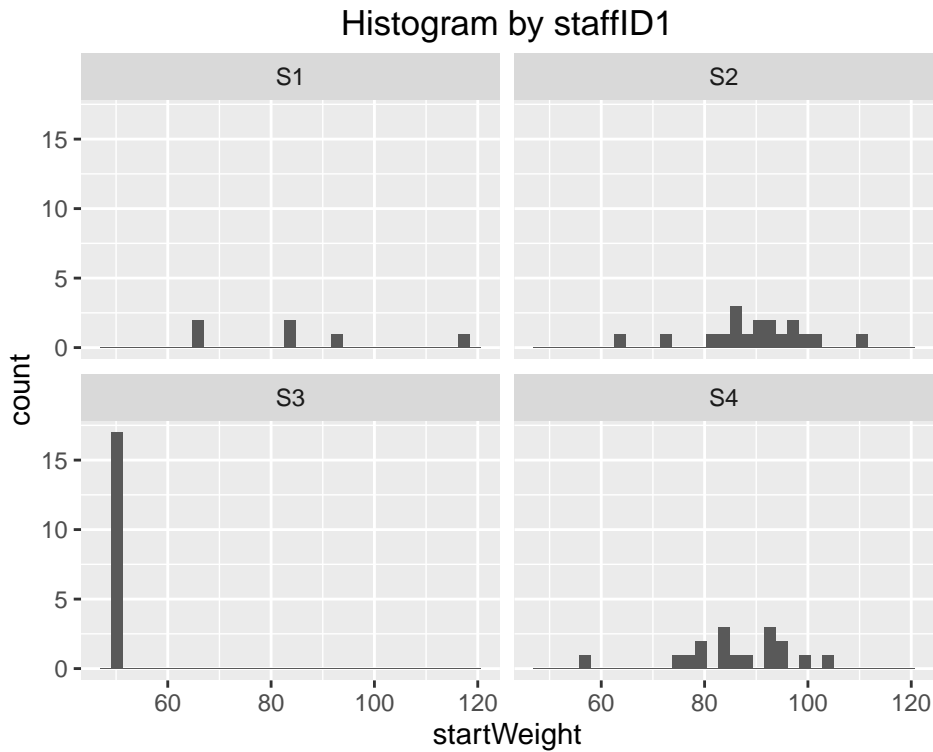
However, all is not well. We should start getting suspicious if we look at the crosstabs between `staffID1` and `staffID2`. Why is there S3 in `staffID1`, but not `staffID2`?

```
table(datasetB$staffID1, datasetB$staffID2)
```

```
##
##      S1 S2 S4
## S1  2  1  3
## S2  4  5  9
## S3  5  7  5
## S4  8  5  4
```

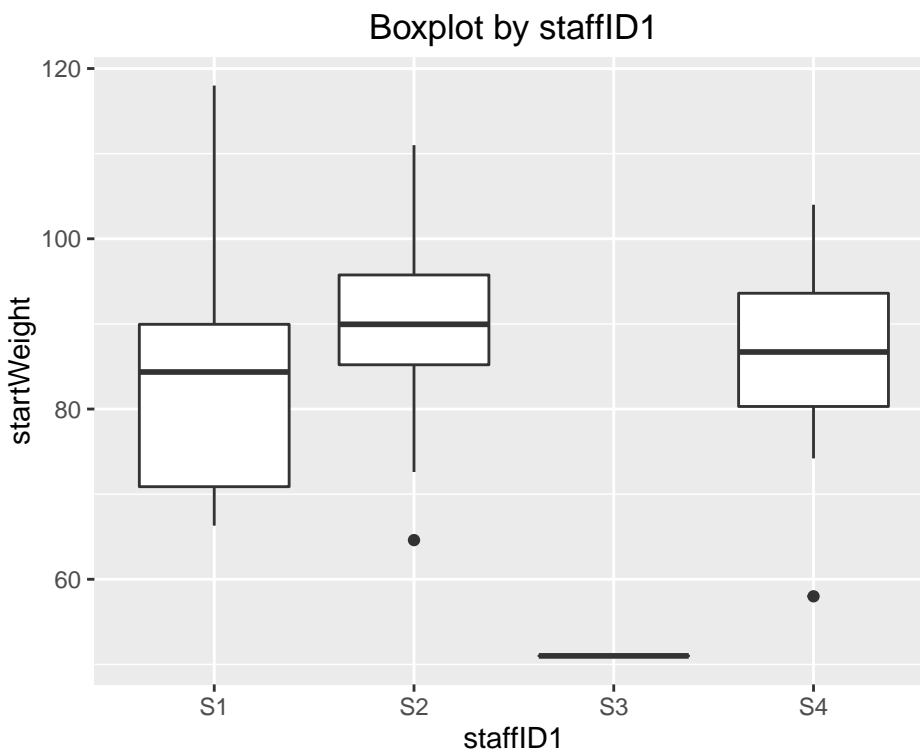
Doing our detective work, we start looking at histograms and boxplots of `startWeight`. Hmm, the weights for S3 look strange. They're all the same value!

```
ggplot(datasetB, aes(x = startWeight)) + geom_histogram() +
  facet_wrap(facet=c("staffID1")) +
  ggtitle("Histogram by staffID1")
```



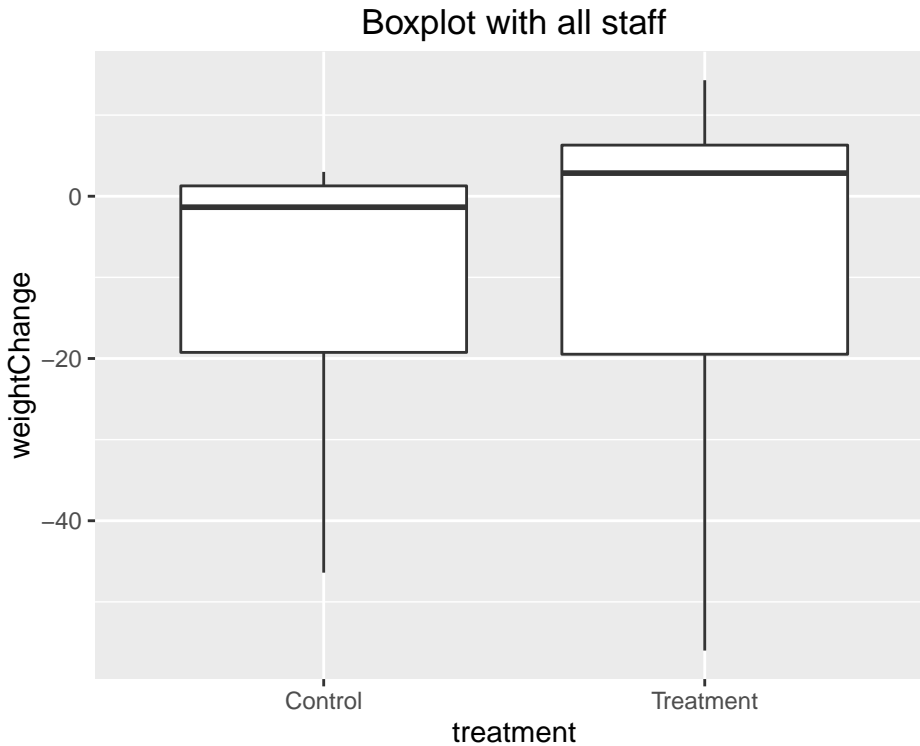
Looking at the boxplots will also show this.

```
ggplot(datasetB, aes(x= staffID1, y = startWeight)) + geom_boxplot() +
  ggtitle("Boxplot by staffID1")
```

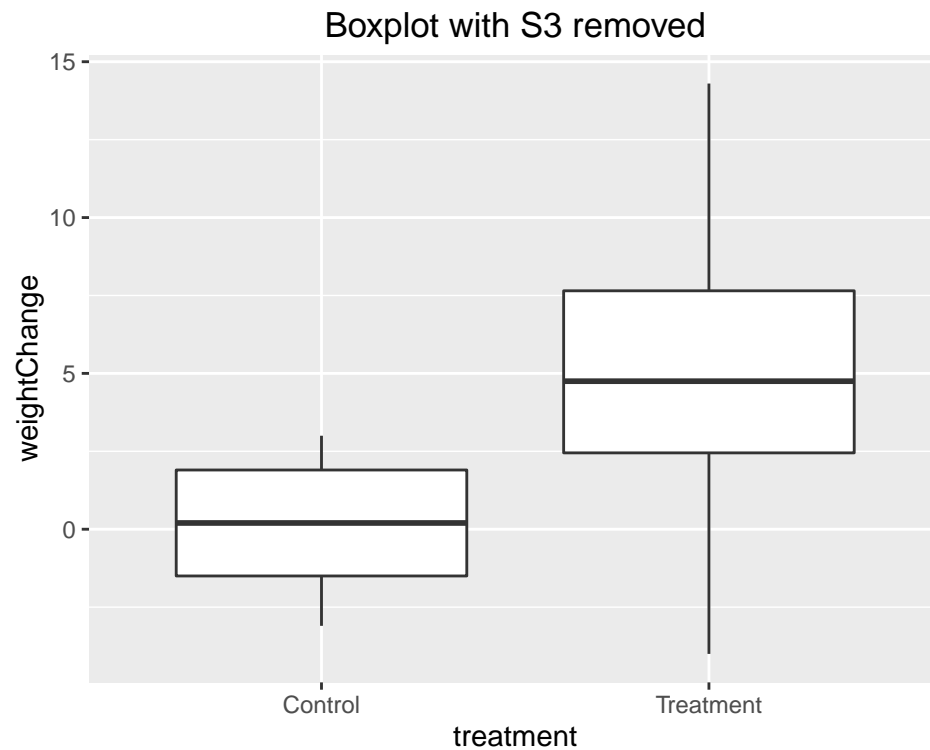


Is this data valid? Better not risk it. We'll take it out. We'll calculate the weight change as weightChange

```
datasetB <- datasetB %>% mutate(weightChange = startWeight - endWeight)
ggplot(datasetB, aes(x=treatment, y=weightChange)) + geom_boxplot() +
  ggtitle("Boxplot with all staff")
```

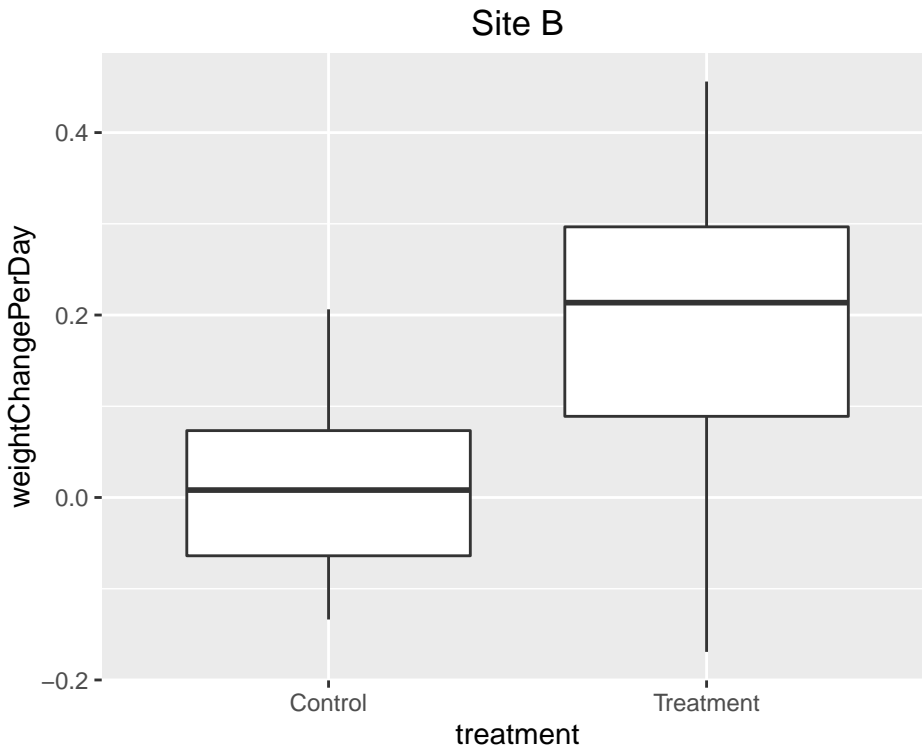


```
datasetB <- datasetB %>% filter(staffID1 != "S3")
ggplot(datasetB, aes(x=treatment, y=weightChange)) + geom_boxplot() +
  ggtitle("Boxplot with S3 removed")
```

Here I convert datasetB's weight loss to pounds, and also scale the weightLoss per day, which seems to be a reasonable way to compare the weight loss across individuals.

```
datasetB <- datasetB %>% mutate(weightChangeLb = weightChange * 2.2, weightChangePerDay =  
  weightChangeLb / timeElapsed)  
  
ggplot(datasetB, aes(x=treatment, y=weightChangePerDay)) + geom_boxplot() +  
  ggtitle("Site B")
```



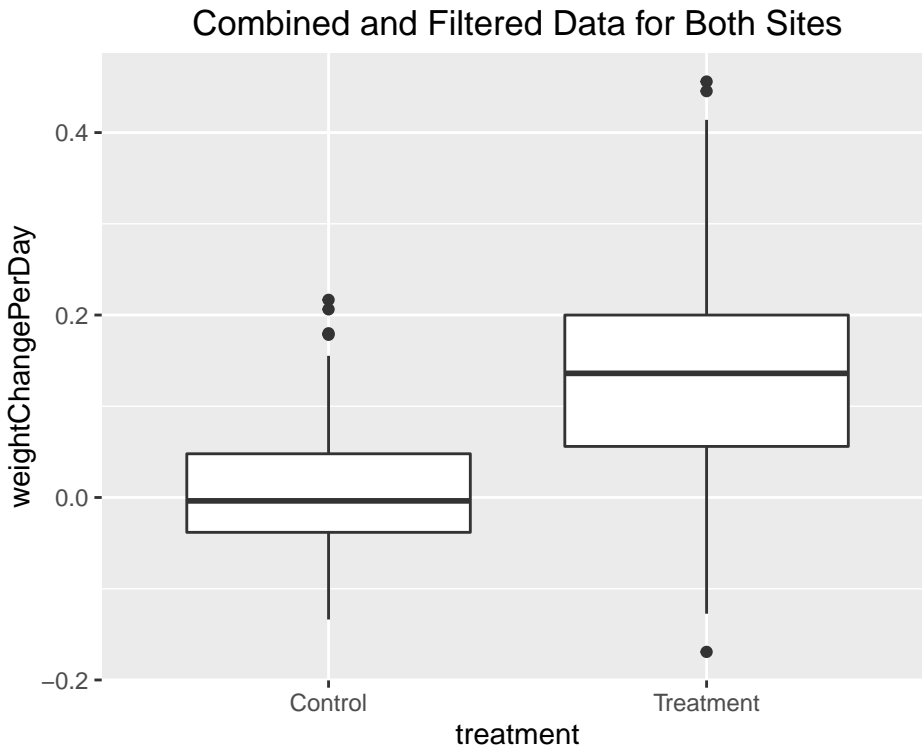
Combining the Data

What is the best way to compare the two datasets? Since the timeElapsed seems to differ among the two datasets, we should compare using `weightChangePerDay`. Students should also note the demographics (especially age) are different between the two datasets.

```
datasetAselect <- datasetA %>% mutate(site="A") %>% select(treatment, weightChangePerDay, site)
datasetBselect <- datasetB %>% mutate(site="B") %>% select(treatment, weightChangePerDay, site)

datasetCombined <- rbind(datasetAselect, datasetBselect)

#plot all combined data together
ggplot(datasetCombined, aes(x=treatment, y=weightChangePerDay)) +
  geom_boxplot() + ggtitle("Combined and Filtered Data for Both Sites")
```



```
#plot all combined data conditioned by site
ggplot(datasetCombined, aes(x=treatment, y=weightChangePerDay)) +
  geom_boxplot() + facet_wrap(facet=c("site")) +
  ggtitle("Combined Data By Site")
```

