Question 1: Shopify Data Intern Challenge - Ata Meshkani

The first thing I did was to download the data into my desktop and open it with a R Markdown file in RStudio.

```
knitr::opts_chunk$set(echo = TRUE)
library(readr)
library(ggplot2)
library(dplyr)
shopify <- read_csv("/Users/atameshkani/Desktop/shopify/newdata.csv")
```

I first took a look at the summary of the order_amount in the data.

```
summary(shopify$order_amount)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      90     163     284    3145     390  704000
```

I noticed that the Mean was far beyond our inter-quartile range. We also see that the Max order value is at 704000. This seems to be an outlier in our case.

```
IQR(shopify$order_amount)
```

```
## [1] 227
```

```
quantile(shopify$order_amount)
```

```
##     0%    25%    50%    75%   100%
##     90    163    284    390 704000
```

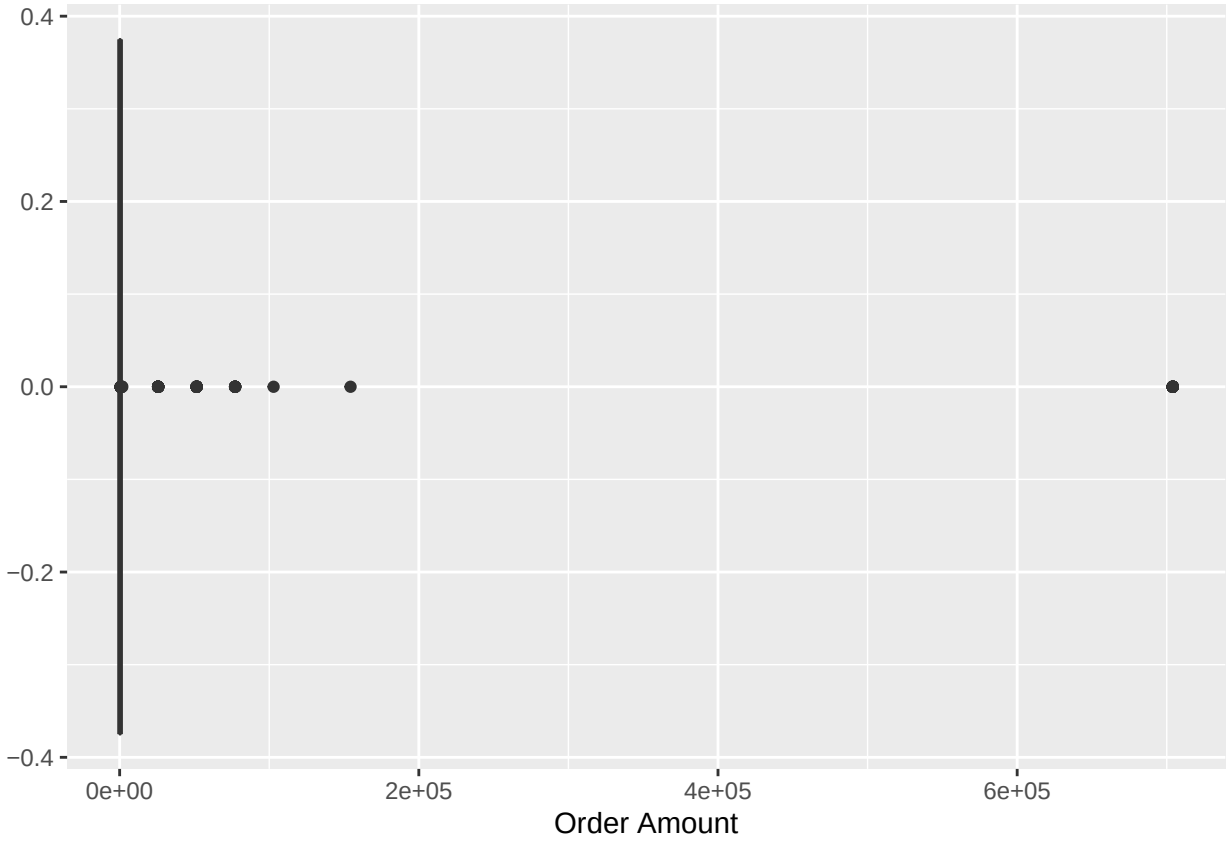
```
sd(shopify$order_amount)
```

```
## [1] 41282.54
```

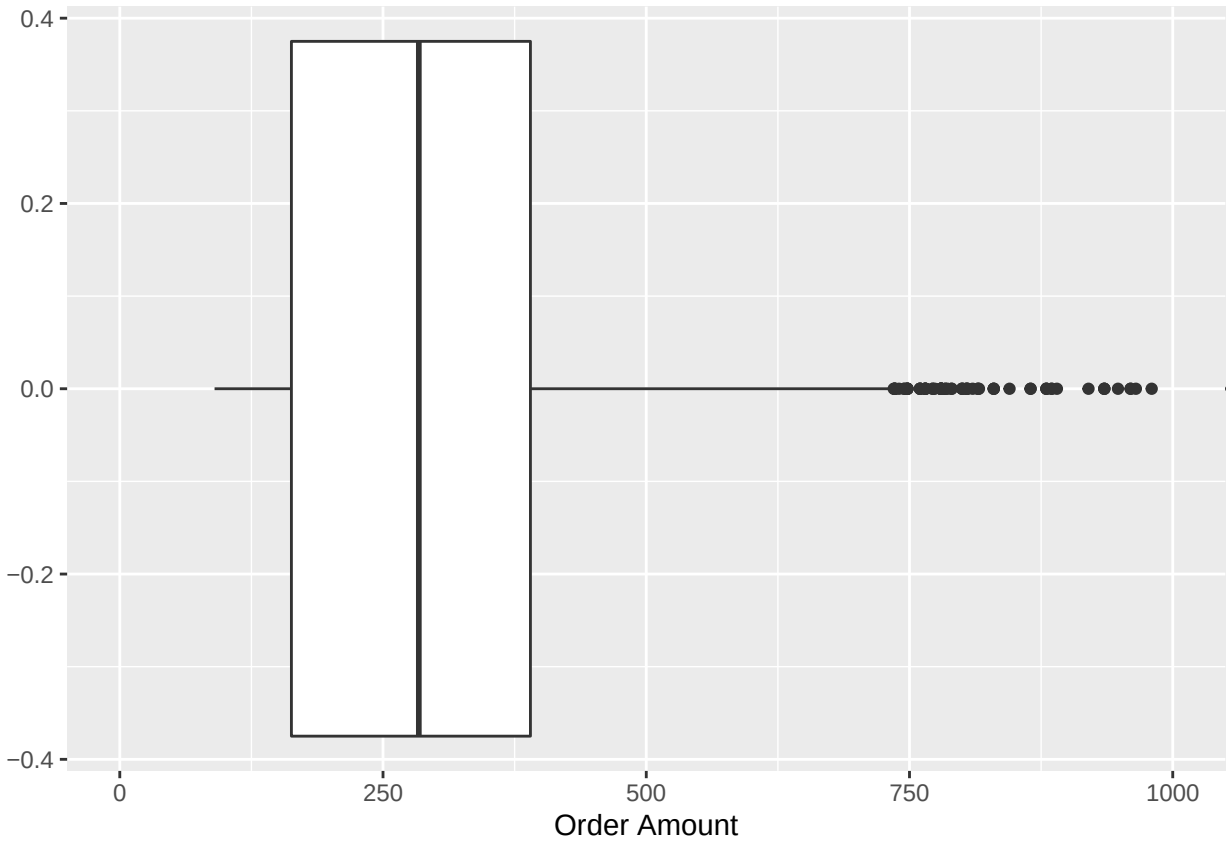
We can see an IQR of 227 and a standard deviation of 41282.54. This means that our data points are very spread out but most purchases are within our IQR.

To visualize this data I created a box-plot.

```
ggplot(shopify,aes(order_amount)) +
  geom_boxplot() +
  xlab("Order Amount")
```




```
ggplot(shopify,aes(order_amount)) +
  geom_boxplot() +
  coord_cartesian(xlim=c(0,1000)) +
  xlab("Order Amount")
```



This helps us vizualize the range for the order amounts and see how far out our outliers are. I then created a box-plot with a limited range so we can get a better look at the IQR.

I also wanted to take a look at why we had these massive outliers within our data set

```
table(unlist(shopify$order_amount))
```

```
##
##   90   94  101  111  112  114  116  117  118  122  127
##   18   25   15   16   48   27   23   29   43   21   22
##  128  129  130  131  132  133  134  136  138  140  142
##   30   50   52   35   23   56   29   38   25    8   72
##  144  145  146  147  148  149  153  154  155  156  158
##   19   37   28   18   31   23   87   16   12   75   29
##  160  161  162  163  164  165  166  168  169  171  172
##   75   43   18   46   52   16   17   14   18   18   11
##  173  176  177  178  180  181  184  187  188  190  193
##   35   62   46   37   19   51   20   39   17   12   13
##  195  196  201  202  222  224  228  232  234  236  244
##   25   24   24   18   14   48   17   16   29   41   17
##  254  256  258  260  262  264  266  268  270  272  276
##   15   41   54   66   36   12   56   28    1   47   23
##  280  282  284  288  290  292  294  296  298  303  306
##   26   10   64    9   32   41   20   35   15    8   85
##  308  310  312  316  320  322  324  326  328  330  332
##   21   17   75   41   72   36   16   39   68   26   13
```

```
##    333    336    338    342    344    346    348    351    352    354    356
##      7     36     24     28     16     35     11     11     66     82     35
##    360    362    366    368    374    376    380    381    384    386    387
##      3     35      7     18     38      2     13      9     18     18     21
##    390    392    393    396    399    402    404    408    414    420    426
##     40     21     16      8     31     42      1     13      6     18     37
##    432    435    438    441    444    447    448    450    456    459    462
##     10     17     12     12     27      9      7      1      5     67     10
##    464    465    468    470    472    474    480    483    486    488    489
##      7     11     41      1      5     16     44     14     11      5     30
##    492    495    498    504    507    508    512    513    516    519    520
##     34     14      8      6      8      6      4     10     21     19     10
##    524    528    531    532    534    536    543    544    552    560    561
##      8     29     34      5     24      4     22      6     11      5     20
##    568    570    576    579    580    584    585    588    590    592    596
##     14      6      1     10      6      7      7     15      6     10      3
##    603    612    616    620    624    632    640    644    645    648    650
##      7     12      7      2      9      5     10      6      3      2      1
##    652    655    656    660    664    665    670    672    676    684    692
##      5      1      7      3      2      2      3      2      3      2      8
##    704    708    710    712    724    725    730    735    736    740    745
##     22      7     10     12      3      1      1      1      4      1      2
##    748    760    765    772    774    780    784    786    790    800    804
##      5      4      5      2      1      8      2      1      2      3      2
##    805    810    815    816    830    845    865    880    885    890    920
##      2      1      2      1      3      1      2      3      2      1      1
##    935    948    960    965    980   1056   1064   1086   1408   1760  25725
##      3      1      2      1      1      3      1      1      2      1     19
##  51450  77175 102900 154350 704000
##     16      9      1      1     17
```

We can see that there are a few purchases that are above 25000 that are heavily skewing our AOV

```
expensive <- subset(shopify, shopify$order_amount > 25000)
big_orders <- expensive %>% group_by(order_amount, order_id)
head(big_orders)
```

```
## # A tibble: 6 x 7
## # Groups:   order_amount, order_id [6]
##   order_id shop_id user_id order_amount total_items payment_method created_at
##      <dbl>   <dbl>   <dbl>        <dbl>       <dbl> <chr>          <chr>
## 1       16      42     607       704000        2000 credit_card    2017-03-07 4~
## 2       61      42     607       704000        2000 credit_card    2017-03-04 4~
## 3      161      78     990        25725           1 credit_card    2017-03-12 5~
## 4      491      78     936        51450           2 debit          2017-03-26 1~
## 5      494      78     983        51450           2 cash           2017-03-16 2~
## 6      512      78     967        51450           2 cash           2017-03-09 7~
```

Going through this subset we see that these big purchases share common user_id and shop_id. Every purchase of 704000 has a shop_id of 607 and a shop_id of 42. We can also see that purchases made at 25725 are for 1 item. Meaning the item is extremely expensive compared to other stores.

After this quick analysis we can see the reason for the high mean is the outliers.

Answers:

1. The main flaw with this calculation is that because the AOV is taking the mean value, it fails to consider outliers. We can look at the data and see that the highest order value is 704000, which severely skews our mean.

2. The metric I would use to figure out the AOV is to look at the median and the interquartile range (IQR). With these metrics we see that the outliers are not able to skew the data.

3. By looking at the quantiles we can see that our median = 284 and IQR = 277. The Median allows us to ignore the outliers and the IQR gives us a better picture of the dispersion of the order amounts. These 2 metrics together give us a better picture of what order value to expect and we can visualize this as well with a boxplot.

0% 25% 50% 75% 100%

90 163 284 390 704000