Shopify Data Science Intern Challenge (Question 1)

Camille Bergeron

5/2/2021

Link to github for these questions: github

Question 1

First Look

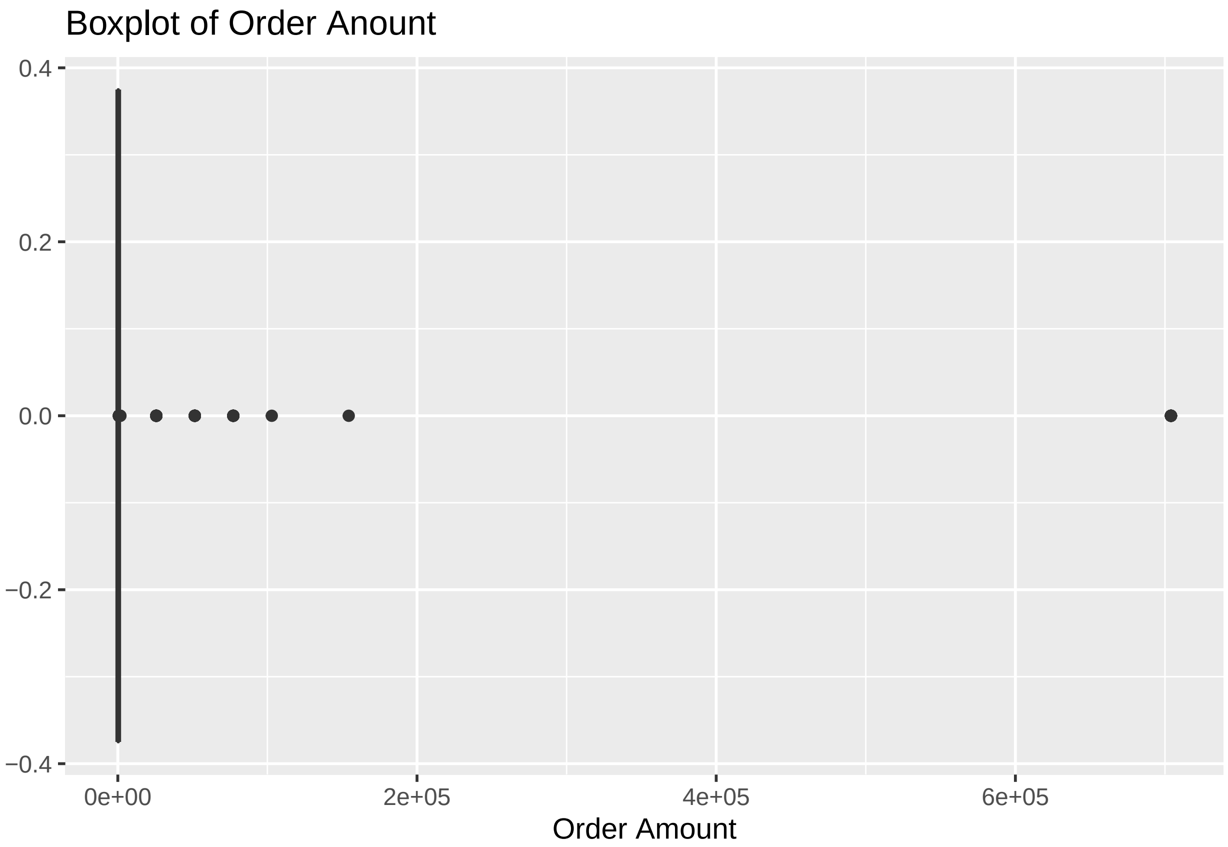
```
# looking at data

# loading the data
orders <- read.csv("2019 Winter Data Science Intern Challenge Data Set - Sheet1.csv")

# avg order value
orders %>%
  select(order_amount) %>%
  summarise(n = mean(order_amount)) %>%
  kable()
```

n
3145.128

```
# looking at potential outliers
ggplot(orders) +
  geom_boxplot(aes(x = order_amount)) +
  labs(title = "Boxplot of Order Anount", x = "Order Amount")
```



```r
# extracting the outliers
boxplot.stats(orders$order_amount)$out
```

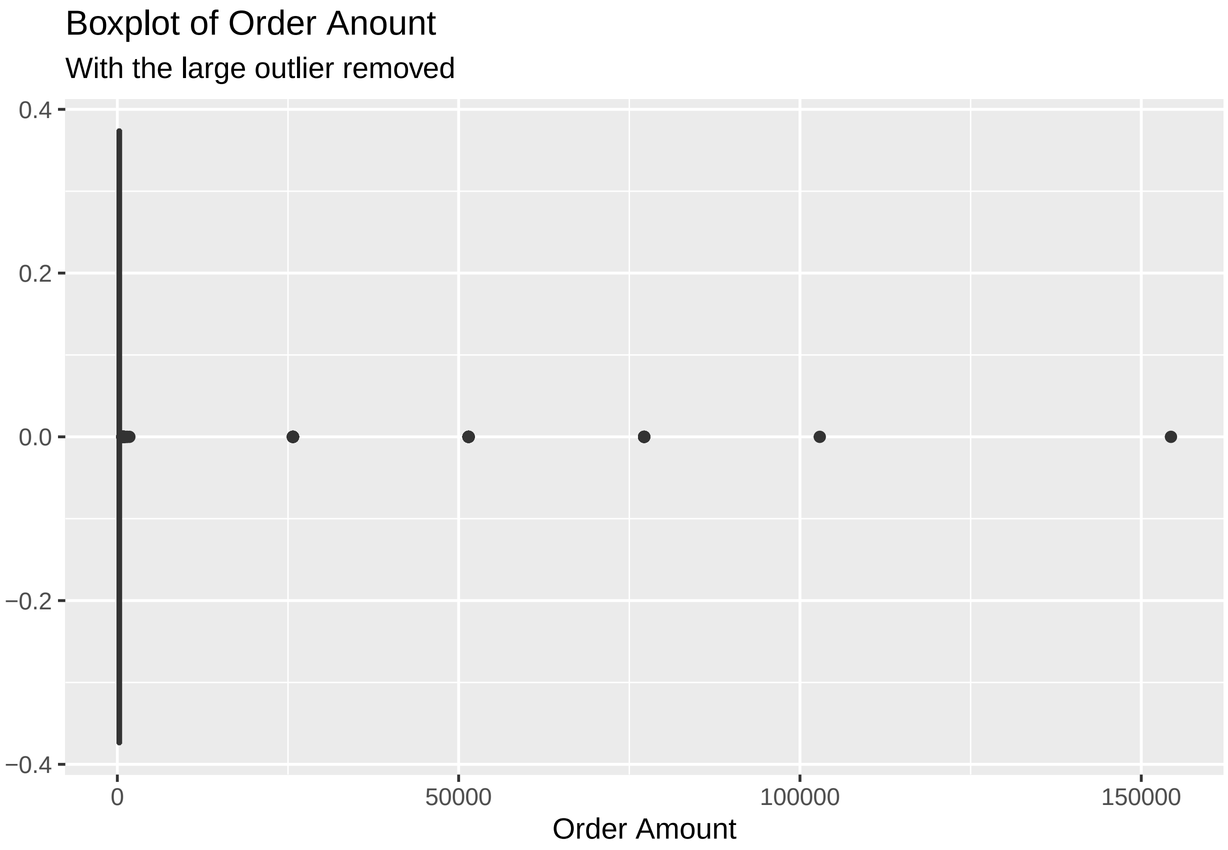
```
##   [1] 704000 704000    780    765  25725    780    765    780    780  51450
##  [11]  51450  51450 704000    830  51450    748 154350    772    804    815
##  [21]    885   1056    784  25725 704000    815    885  25725  25725    935
##  [31]  77175 704000   1760   1408  25725  25725 704000  25725   1408    765
##  [41]    736  51450 704000    960 704000    800    804    800    865    745
##  [51]    830    880    920    765    774    790    784 704000  25725 704000
##  [61]    948    845    760    745  51450 102900    965  51450  51450  25725
##  [71]    935  77175    780  77175    805  25725  51450  51450 704000  77175
##  [81]  25725    830 704000   1056    890    980  25725  51450    760  25725
##  [91]  51450    748    786 704000  77175    736    805  25725   1056    736
## [101]    935   1086    736  51450  77175  25725    816    810    740  25725
## [111] 704000  51450   1064  77175    780  51450  51450  77175    735  25725
## [121]    760    880    780    748    748  25725    748    800 704000    780
## [131]  77175    960 704000    790 704000    760  25725    765    880    865
## [141]    772
```

```r
orders[which(orders$order_amount == max(orders$order_amount)), ]
```

```
##      order_id shop_id user_id order_amount total_items payment_method
## 16         16      42     607       704000        2000    credit_card
## 61         61      42     607       704000        2000    credit_card
## 521       521      42     607       704000        2000    credit_card
## 1105     1105      42     607       704000        2000    credit_card
## 1363     1363      42     607       704000        2000    credit_card
## 1437     1437      42     607       704000        2000    credit_card
## 1563      1563       42      607        704000         2000     credit_card
## 1603      1603       42      607        704000         2000     credit_card
## 2154      2154       42      607        704000         2000     credit_card
## 2298      2298       42      607        704000         2000     credit_card
## 2836      2836       42      607        704000         2000     credit_card
## 2970      2970       42      607        704000         2000     credit_card
## 3333      3333       42      607        704000         2000     credit_card
## 4057      4057       42      607        704000         2000     credit_card
## 4647      4647       42      607        704000         2000     credit_card
## 4869      4869       42      607        704000         2000     credit_card
## 4883      4883       42      607        704000         2000     credit_card
##               created_at
## 16   2017-03-07 4:00:00
## 61   2017-03-04 4:00:00
## 521  2017-03-02 4:00:00
## 1105 2017-03-24 4:00:00
## 1363 2017-03-15 4:00:00
## 1437 2017-03-11 4:00:00
## 1563 2017-03-19 4:00:00
## 1603 2017-03-17 4:00:00
## 2154 2017-03-12 4:00:00
## 2298 2017-03-07 4:00:00
## 2836 2017-03-28 4:00:00
## 2970 2017-03-28 4:00:00
## 3333 2017-03-24 4:00:00
## 4057 2017-03-28 4:00:00
## 4647 2017-03-02 4:00:00
## 4869 2017-03-22 4:00:00
## 4883 2017-03-25 4:00:00
```

The largest outliers are all exactly $704000 by the same user at the same shop with the same payment method. The only difference is that the transactions take place on different days in March, but all at 4:00 am.

```r
# new boxplot without the large order
orders %>%
  filter(user_id != 607) %>%
  ggplot() +
  geom_boxplot(aes(x = order_amount)) +
  labs(title = "Boxplot of Order Anount", x = "Order Amount", subtitle = "With the large outlier remove
```



```
  # there are still lot of outliers so these clearly not it

# looking at this average
orders %>%
  filter(user_id != 607) %>%
  summarise(n = mean(order_amount)) %>%
  kable()
```

n
754.0919

```
# this is much cheaper, so maybe this is better?

# separating the date and time
orders <- orders %>%
  separate(created_at, c("date", "time"), " ") %>%
  mutate(date = as.Date(date))

# taking out the dates to see if they were closing out
orders %>%
  filter(shop_id == 42) %>%
  arrange(date) %>%
  head() %>%
  kable()
```

order_id	shop_id	user_id	order_amount	total_items	payment_method	date	time
2019	42	739	352	1	debit	2017-03-01	12:42:26
2492	42	868	704	2	debit	2017-03-01	18:33:33
4422	42	736	704	2	credit_card	2017-03-01	12:19:49
521	42	607	704000	2000	credit_card	2017-03-02	4:00:00
4647	42	607	704000	2000	credit_card	2017-03-02	4:00:00
2988	42	819	1056	3	cash	2017-03-03	9:09:25

```r
# this is the average amount spent on items
orders %>%
  mutate(avg = order_amount / total_items) %>%
  summarise(n = mean(avg)) %>%
  kable()
```

n
387.7428

Another way to do analysis similar to the average order value (AOV) is by looking at the average value per item. This number is $387 per sneaker.