Shopify Technical Challenge

Zhilin (Catherine) Zhou

07/01/2022

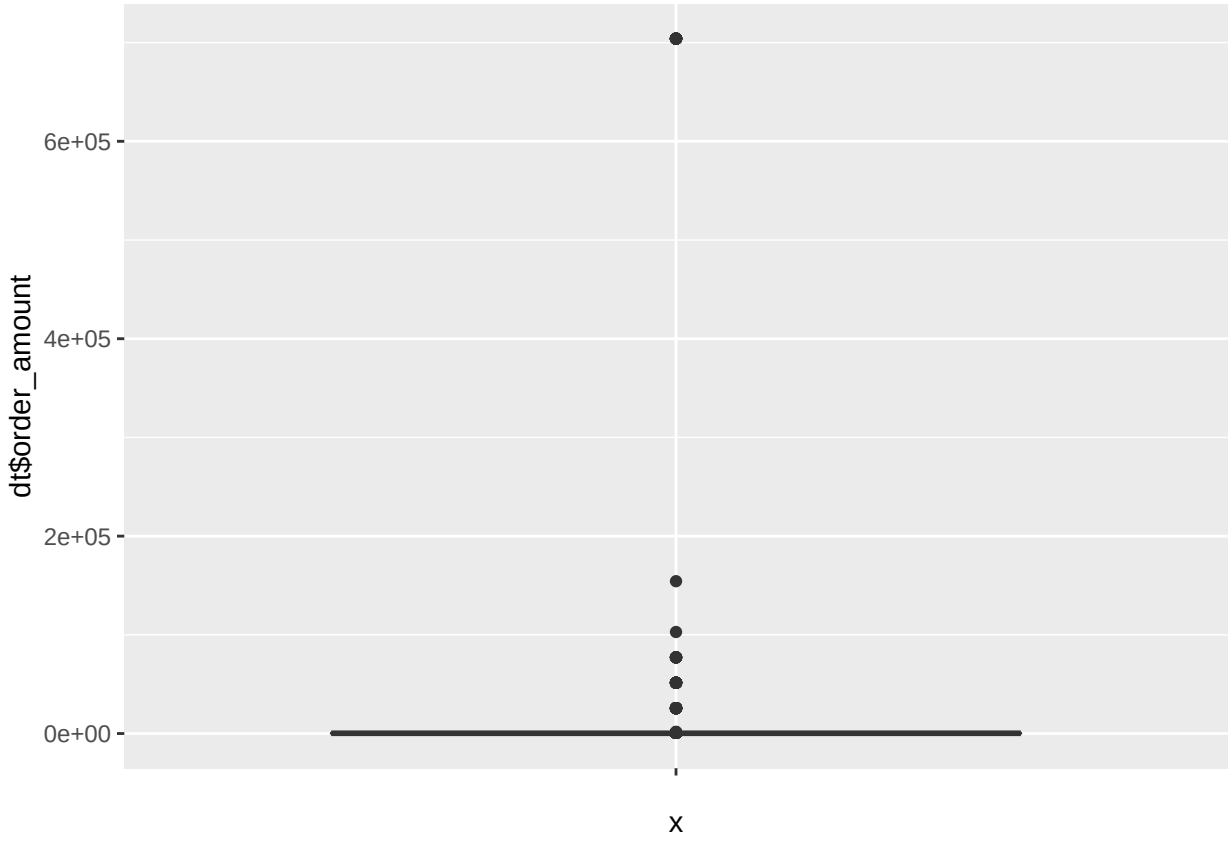
Question 1

```
dt <- read.csv(file = 'Dataset.csv')
summary(dt$order_amount)
```

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

```
ggplot(dt) +
  aes(x="", y=dt$order_amount) +
  geom_boxplot(fill="#4F8FF0")
```

```
## Warning: Use of `dt$order_amount` is discouraged. Use `order_amount` instead.
```



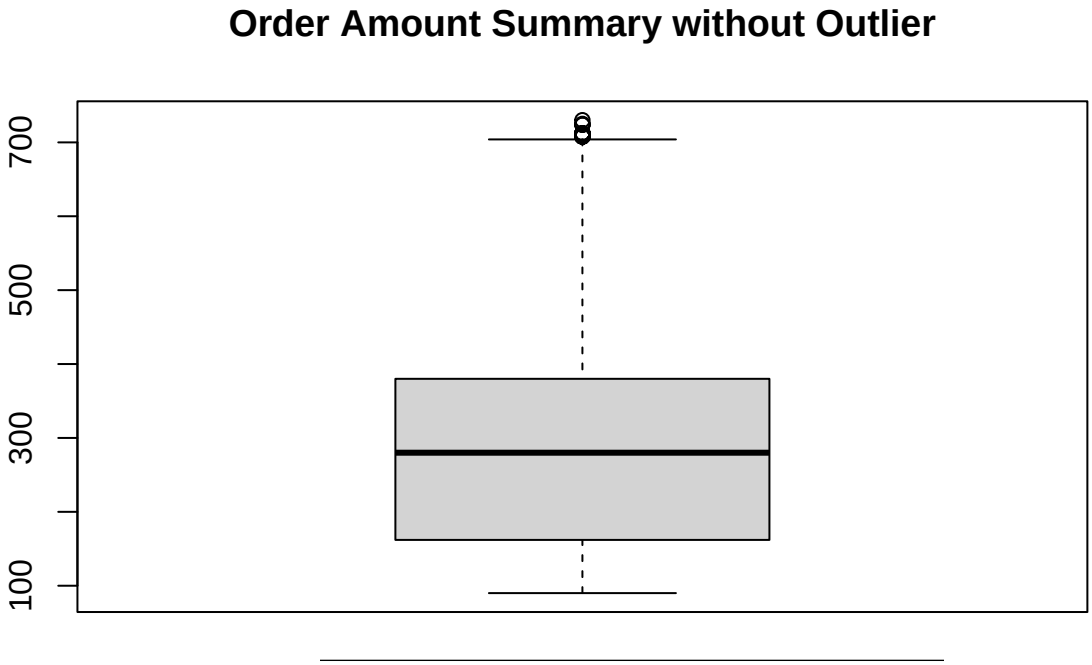
Based on the database, the `order_amount` value is the total amount for each order, while some orders contain more than one sneakers. Therefore, if some customers buy a huge amount of sneakers in one order, such as order 16, it increases AOV when calculating the average. This is why the AOV is so high ($3145.13) for

sneakers. One alternation is to remove the outliers and analyze the rest to get a more general idea of the AOV.

```
outliers <- boxplot.stats(dt$order_amount)$out
dt_no <- dt[!dt$order_amount %in% outliers, ]
dt_out <- dt[dt$order_amount %in% outliers, ]
summary(dt_no$order_amount)
```

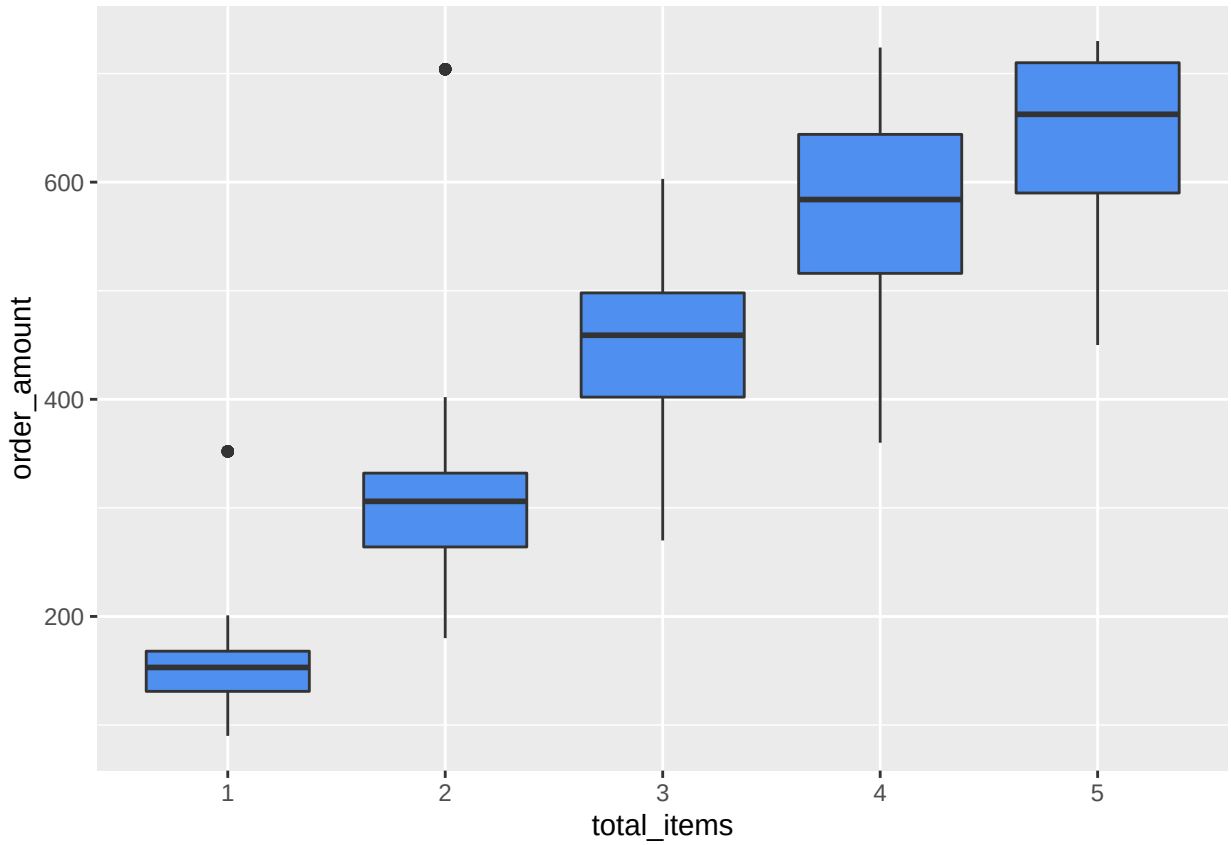
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    90.0   162.0   280.0   293.7   380.0   730.0
```

```
boxplot(dt_no$order_amount, main="Order Amount Summary without Outlier")
```



A better way to analyze the data is to analyze the relationship between the order amount, total items, and the payment method. Generally, we know that the more items you purchase, the higher price you pay.

```
dt_no$total_items <- as.factor(dt_no$total_items)
ggplot(data=dt_no, mapping=aes(x=total_items, y=order_amount)) +
  geom_boxplot(fill="#4F8FF0")
```




```
order_item <- lm(dt$order_amount ~ dt$total_items)
summary(order_item)
```

```
##
## Call:
## lm(formula = dt$order_amount ~ dt$total_items)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -1804   -565   -434   -262 152186
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     52.2544    75.0736   0.696    0.486
## dt$total_items 351.9749     0.6436 546.855   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5293 on 4998 degrees of freedom
## Multiple R-squared:  0.9836, Adjusted R-squared:  0.9836
## F-statistic: 2.991e+05 on 1 and 4998 DF,  p-value: < 2.2e-16
```

```
print("Average order amount per item")
```

```
## [1] "Average order amount per item"
```

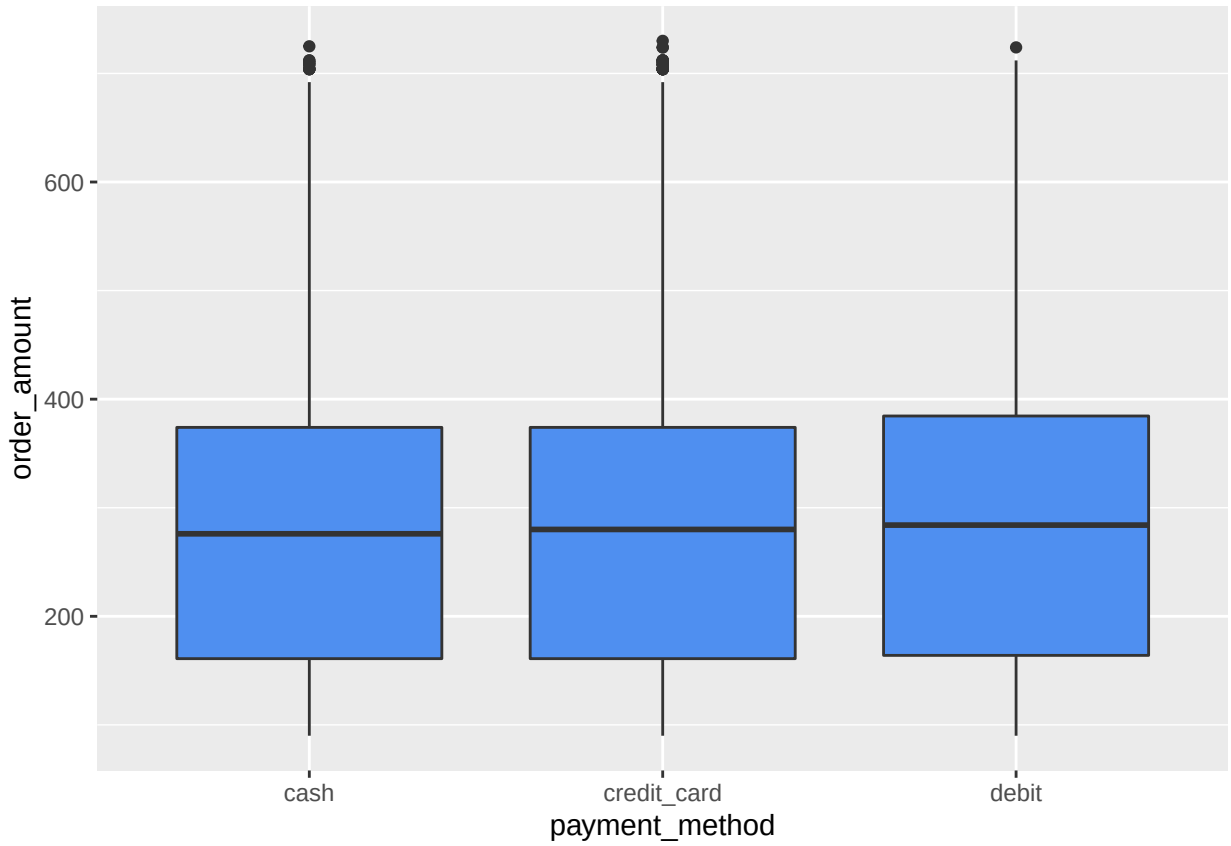
```
summary(dt$order_amount/dt$total_items)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

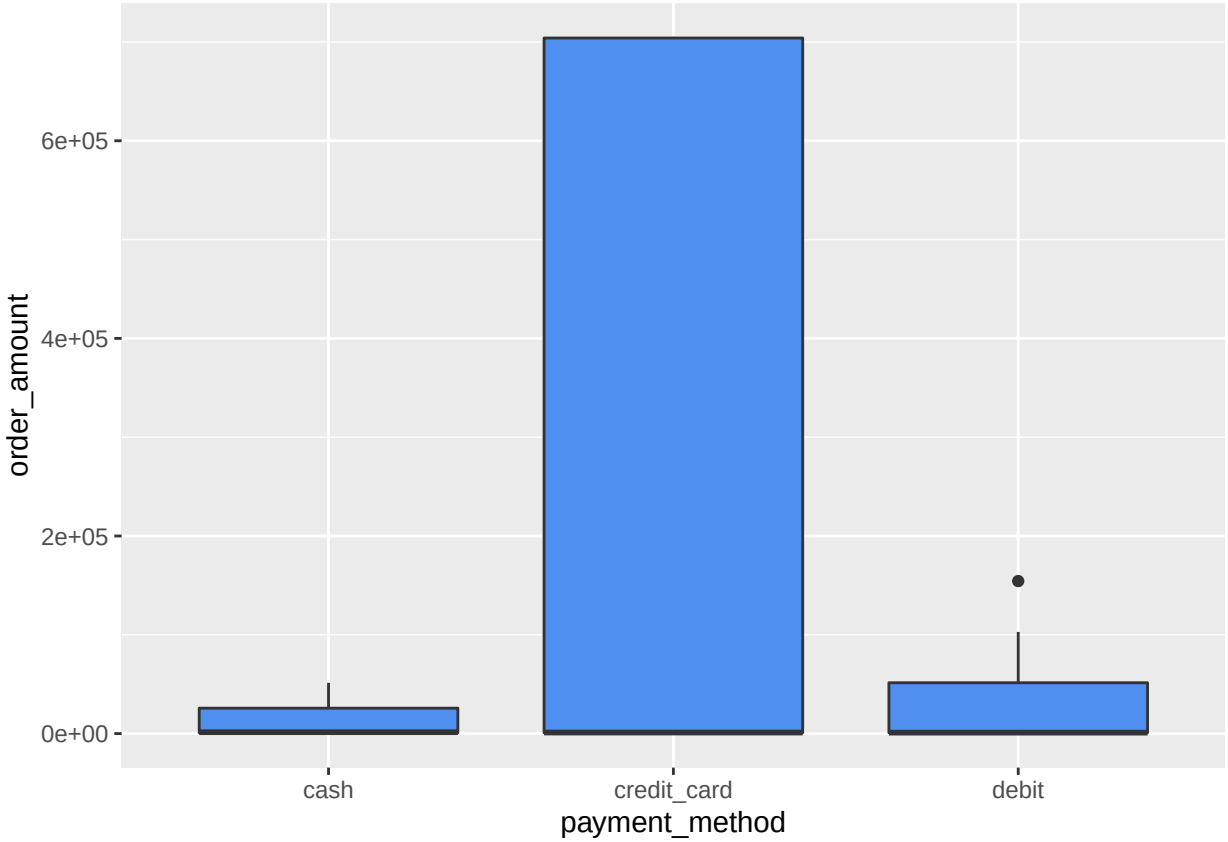
```
##    90.0   133.0   153.0   387.7   169.0 25725.0
```

We can see from the boxplot that as the number of items increase, the order amount gradually increase. The result of the linear regression shows that the relationship is pretty linear since the p-value is extremely small.

```
dt_no$payment_method <- as.factor(dt_no$payment_method)
ggplot(data=dt_no, mapping=aes(x=payment_method, y=order_amount)) +
  geom_boxplot(fill="#4F8FF0")
```




```
dt_out$payment_method <- as.factor(dt_out$payment_method)
ggplot(data=dt_out, mapping=aes(x=payment_method, y=order_amount)) +
  geom_boxplot(fill="#4F8FF0")
```



For most customers, purchasing sneakers using cash, credit card, or debit doesn't affect the total amount they purchase, since the first boxplot shows that their range and average are almost the same. Then I analyze the outliers and find that credit card users are more likely to purchase large orders. This is legitimate since the amount of cash one can carry is limited, and credit cards mostly have purchase benefits, so people tend to use credit cards to purchase more items.

```
order_cash <- dt$order_amount[dt$payment_method == 'cash']
order_cash_item <- order_cash/dt$total_items[dt$payment_method == 'cash']
order_credit <- dt$order_amount[dt$payment_method == 'credit_card']
order_credit_item <- order_credit/dt$total_items[dt$payment_method == 'credit_card']
order_debit <- dt$order_amount[dt$payment_method == 'debit']
order_debit_item <- order_debit/dt$total_items[dt$payment_method == 'debit']

row_names <- c("AOV", "AOV per item")
col_names <- c("Cash", "Credit Card", "Debit")
data <- c(mean(order_cash), mean(order_credit), mean(order_debit),
          mean(order_cash_item), mean(order_credit_item), mean(order_debit_item))
result <- matrix(data, nrow=2, byrow=TRUE, dimnames=list(row_names, col_names))
print(result)
```

```
##                 Cash Credit Card    Debit
## AOV          730.3532   7461.5948 966.8402
## AOV per item 439.8645    301.0911 427.9934
```

I would use the AOV per item for different payment method as the metric for this dataset.

Question 2

1. How many orders are shipped by Speedy Express?

```
SELECT COUNT(OrderID) as SpeedyExpressOrders
FROM Orders INNER JOIN
(SELECT ShipperID FROM Shippers WHERE ShipperName = "Speedy Express") as Ship
ON Orders.ShipperID = Ship.ShipperID
```

The answer is 54 orders.

2. What is the last name of the employee with the most orders?

```
SELECT LastName FROM Employees INNER JOIN
(SELECT TOP 1 EmployeeID FROM Orders
GROUP BY EmployeeID
ORDER BY COUNT(OrderID) DESC
) AS MostOrders
ON Employees.EmployeeID = MostOrders.EmployeeID
```

The answer is Peacock.

3. What product was ordered the most by customers in Germany?

```
SELECT ProductName FROM Products INNER JOIN
(SELECT TOP 1 ProductID FROM OrderDetails INNER JOIN
  (SELECT OrderID FROM Orders INNER JOIN
    (SELECT CustomerID FROM Customers WHERE Country = "Germany"
    ) AS c
   ON Orders.CustomerID = c.CustomerID
  ) AS o
ON OrderDetails.OrderID = o.OrderID
GROUP BY ProductID
ORDER BY COUNT(ProductID) DESC
) AS MostOrders
ON Products.ProductID = MostOrders.ProductID
```

The answer is Gorgonzola Telino.