TASK 2- Experimentation and uplift testing

SIDDHESH POTE

20/12/2020

From task 1 solution model i learned how simple and efficient the way of writing the code could be.... I used their way for the further analysis in the task and most probably use it for any code i write in future

INTRODUCTION

For this part of the project we will be examining the performance in trial vs control stores to provide a recommendation for each location based on our insight. Below are the steps that were followed during analysis

- Select control stores – exploroing the data and define metrics for control store selection . Thinking about what would make them a control store. Looking at the drivers and make sure to visualise these in a graph to better determine if they are suited. For this piece it may even be worth creating a function for help.
- Assessment of the trial – this one should give us some interesting insights into each of the stores, check each trial store individually in comparison with the control store to get a clear view of its overall performance. We want to know if the trial stores were successful or not.
- Collate findings – summarise our findings for each store and provide an recommendation that we can share outlining the impact on sales during the trial period.

Load required libraries and datasets

```
pacman::p_load(ggplot2, dplyr, tidyr , data.table , readr)

filepath <- "C:/Users/Siddhesha/Desktop/R commom directory/quantinum virtual internship/"
data <- fread(paste0(filepath, "QVI_data.csv"))
```

Set themes for plots

```
theme_set(theme_bw())
theme_update(plot.title = element_text(hjust = 0.5))
```

#Select Control Stores

The client has selected store numbers 77, 86 and 88 as trial stores and want control stores to be established stores that are operational for the entire observation period. We would want to match trial stores to control stores that are similar to the trial store prior to the trial period of Feb 2019 in terms of :

- Monthly overall sales revenue
- Monthly number of customer
- Monthly number of transactions per customer

Let's first create the metrics of interest and filter to stores that are present throughout the pre-trial period.

```
# First checking for the nulls in the data
colSums(is.na(data))
```

```
##   LYLTY_CARD_NBR             DATE        STORE_NBR           TXN_ID
##                0                0                0                0
##         PROD_NBR        PROD_NAME         PROD_QTY        TOT_SALES
##                0                0                0                0
##        PACK_SIZE            BRAND        LIFESTAGE PREMIUM_CUSTOMER
##                0                0                0                0
```

```
# Calculate the above mentioned measures over time for each store
# Add a new month ID column in the data with the format yyyymm.

# First converting date into month format and adding a new column of MONTHYEAR

data[["MONTHYEAR"]] <- format(data$DATE, "%Y%m")
# Next, we define the measure calculations to use during the analysis.
#For each store and month calculate total sales, number of customers,transactions per customer, chips p

measureOverTime <- data[, .(totSales = sum(TOT_SALES),
                            nCustomers = uniqueN(LYLTY_CARD_NBR),
                            nTxnPerCust = uniqueN(TXN_ID)/uniqueN(LYLTY_CARD_NBR),
                            nChipsPerTxn = sum(PROD_QTY)/uniqueN(LYLTY_CARD_NBR),
                            avgPricePerUnit = sum(TOT_SALES)/sum(PROD_QTY)
), by = c("MONTHYEAR", "STORE_NBR")][order(STORE_NBR, MONTHYEAR)]

# Filter to the pre-trial period and stores with full observation periods
storesWithFullObs <- measureOverTime[, .N, STORE_NBR][N == 12, STORE_NBR]
preTrailMeasures <- measureOverTime[MONTHYEAR < 201902 & STORE_NBR %in% storesWithFullObs, ]
```

Now we need to work out a way of ranking how similar each potential control store is to the trial store. We can calculate how correlated the performance of each store is to the trial store

Lets write a function for this so that we dont have to calulate this for each trail store and control store pair

Inputs and output of the function

- metricCol (str): Name of column containing store's metric to perform correlation test on.
- storeComparison (int): Trial store's number.
- inputTable (dataframe): Metric table with potential comparison stores. Returns:
- DataFrame: Monthly correlation table between Trial and each Control stores.

```
# Create a function to calculate correlation for a measure, looping through each control store.
calculateCorr <- function(inputTable , metricCol, storeComparision){
    calcCorrTable = data.table(Store1 = numeric(), Store2 = numeric(), corr_measure = numeric())
    storeNumbers <- unique(inputTable[, STORE_NBR])
```

```
    for(i in storeNumbers){
           calculatedMeasure = data.table("Store1" = storeComparision,
                                          "Store2" = i,
                                          "corr_measure" =
                                              cor(inputTable[STORE_NBR %in% storeComparision, eval(metri
                                          )
           calcCorrTable <- rbind( calcCorrTable, calculatedMeasure)
           }
    return(calcCorrTable)
}

head(calculateCorr(preTrailMeasures, quote(totSales), 77))
```

```
##    Store1 Store2 corr_measure
## 1:     77      1   0.07521784
## 2:     77      2  -0.26307873
## 3:     77      3   0.80664364
## 4:     77      4  -0.26329960
## 5:     77      5  -0.11065231
## 6:     77      6   0.04248975
```

```
# similarly we can take metricCol as any another column with any other column and find the correlation
```

Apart from correlation, we can also calculate a standardised metric based on the absolute difference between the trial store's performance and each control store's performance.

Let's write a function for this.

```
calculateMagnitudeDistance <- function(inputTable, metricCol, storeComparision){
       calcDistTable <- data.table(Store1 = numeric(), Store2 = numeric(), MONTHYEAR =numeric(), measure
       storeNumbers = unique(inputTable[, STORE_NBR])
       for (i in storeNumbers) {
              calculatedMeasure = data.table("Store1" = storeComparision,
                                             "Store2" = i ,
                                             "MONTHYEAR"= inputTable[STORE_NBR == storeComparision, MONT
                                             "measure" =abs(inputTable[STORE_NBR == storeComparision, ev
                                             )
              calcDistTable <- rbind(calcDistTable , calculatedMeasure)
       }
       ## standardise the magnitude distance so that the measure ranges from 0 to 1
       minMaxDist <- calcDistTable[, .(minDist = min(measure), maxDist = max(measure)),
                                   by = c("Store1", "MONTHYEAR")]
       distTable <- merge(calcDistTable, minMaxDist, by = c("Store1", "MONTHYEAR"))
       distTable[, magnitudeMeasure := 1 - (measure - minDist)/(maxDist - minDist)]

       finalDistTable <- distTable[, .(mag_measure = mean(magnitudeMeasure)), by = .(Store1, Store2)]
       return(finalDistTable)
}
head(calculateMagnitudeDistance(preTrailMeasures, Quote(nCustomers), 77))    # example
```

```
##    Store1 Store2 mag_measure
## 1:     77      1   0.9403206
```

```
## 2:     77      2   0.9246380
## 3:     77      3   0.3450667
## 4:     77      4   0.1895787
## 5:     77      5   0.4811990
## 6:     77      6   0.9396196
```

Now let's use the functions to find the control stores! We'll select control stores based on how similar monthly total sales in dollar amounts and monthly number of customers are to the trial stores. So we will need to use our functions to get four scores, two for each of total sales and total customers

TRAIL STORE- 77

```
# using store 77 as our trail store
trail_store <- 77
corr_nSales <- calculateCorr(preTrailMeasures, quote(totSales), trail_store)
corr_nCustomers <- calculateCorr(preTrailMeasures, quote(nCustomers), trail_store)

# using function to calculate the mangnitude distance
magnitude_nsales <- calculateMagnitudeDistance(preTrailMeasures, quote(totSales), trail_store)
magnitude_nCustomers <- calculateMagnitudeDistance(preTrailMeasures, quote(nCustomers), trail_store)
```

We'll need to combine the all the scores calculated using our function to create a composite score to rank on.

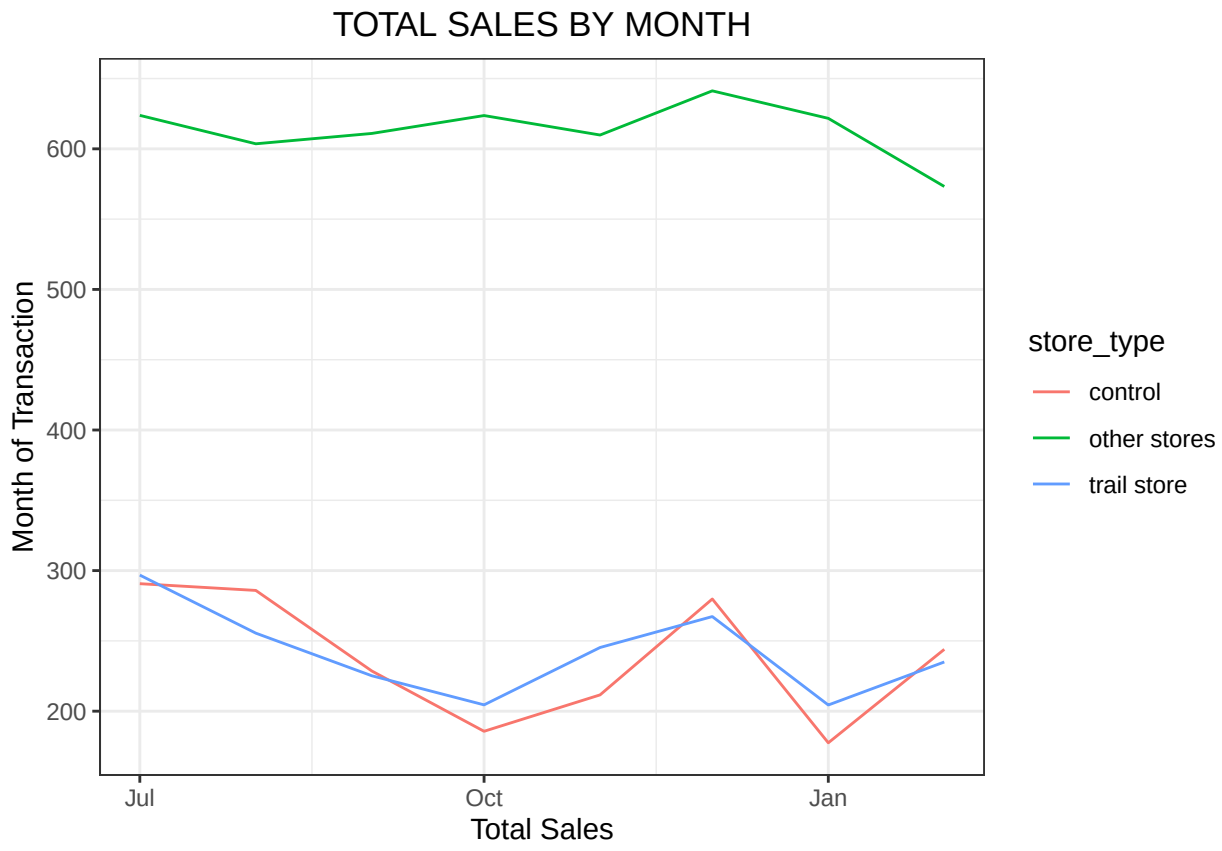
Let's take a simple average of the correlation and magnitude scores for each driver. Note that if we consider it more important for the trend of the drivers to be similar, we can increase the weight of the correlation score (a simple average gives a weight of 0.5 to the corr_weight) or if we consider the absolute size of the drivers to be more important, we can lower the weight of the correlation score

```
# creating a combined score composed of correlation and magnitude by first merging the correlations tab

corr_weight <- 0.5
score_nSales <- merge(corr_nSales, magnitude_nsales, by = c("Store1", "Store2"))[, score_nSales := 0.5*
score_nCustomers <- merge(corr_nCustomers, magnitude_nCustomers, by = c("Store1", "Store2"))[, score_nCu

# now we have a score for each of the total number of sales and number of customers. lets combine the t

score_control <- merge(score_nSales, score_nCustomers, by = c("Store1", "Store2"))

score_control[, finalControlStore := score_nSales*0.5+score_nCustomers*0.5]

# the store with the second highest score is then selected as the control since it is most similar to t

## selecting the most appropriate control stroe for the trail store 77

control_store <- score_control[order(-finalControlStore), ]
(control_store <- control_store[,Store2][[2]])
```

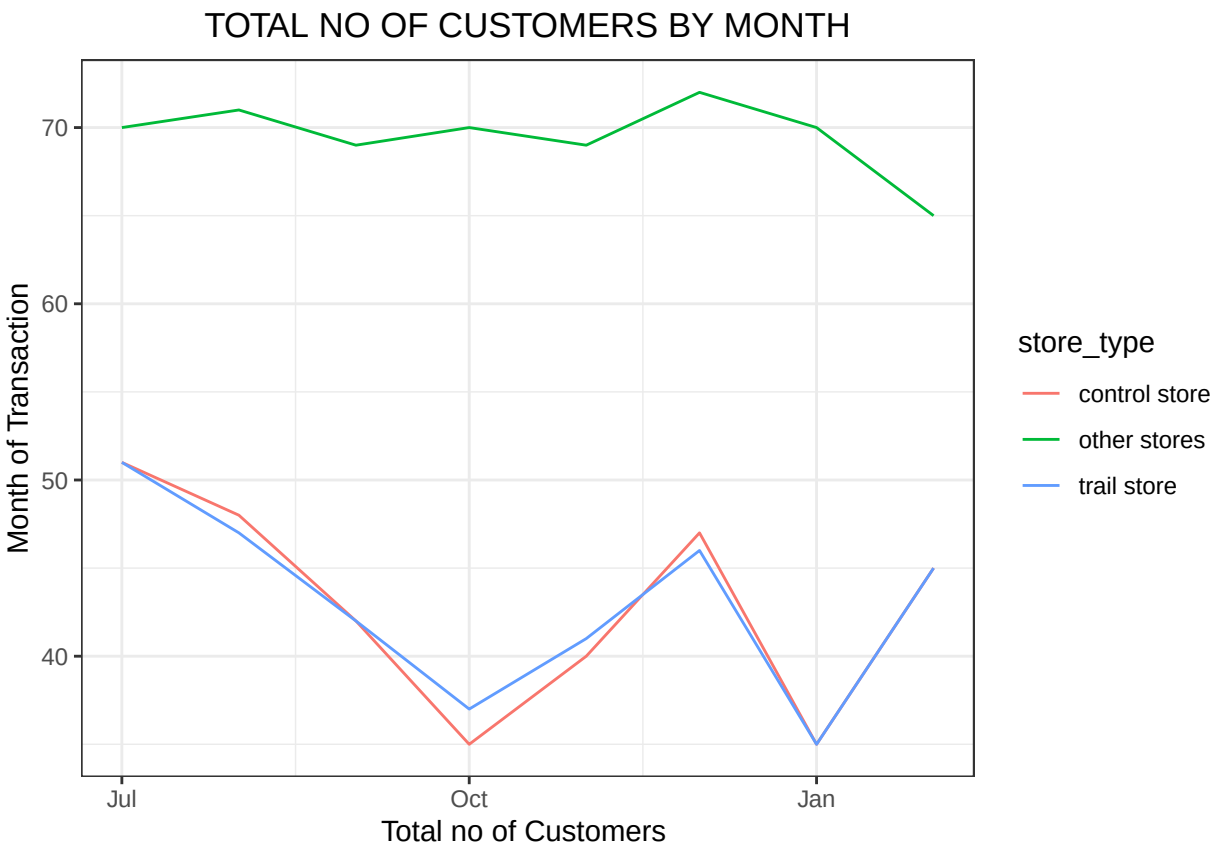
```
## [1] 233
```

Now that we have found a control store, let's check visually if the drivers are indeed similar in the period before the trial.

```
# Visual checks on trends based on the drivers
# first we'll check for sales
measureOverTimeSales <- measureOverTime
pastSales <- measureOverTimeSales[, store_type := ifelse(STORE_NBR == trail_store, "trail store", ifelse

# plotting the data
ggplot(pastSales, aes(transactionMonth, totSales , col = store_type))+ geom_line()+ labs(x = "Total Sale
```




```
# for Customers
pastCustomers <- measureOverTimeSales[, store_type:= ifelse(STORE_NBR == trail_store, "trail store", if

# plotting the data
ggplot(pastCustomers, aes(transactionMonth, nCustomers , col = store_type))+ geom_line()+ labs(x = "Tota
```



Assessment of Trial

The trial period goes from the start of February 2019 to April 2019. We now want to see if there has been an uplift in overall chip sales.

We'll start with scaling the control store's sales to a level similar to control for any differences between the two stores outside of the trial period.

```
## scale pre-trail control sales to match pre-trail store sales
scalingFactorForControlSales <- preTrailMeasures[STORE_NBR==trail_store & MONTHYEAR < 201902, sum(totSal

## apply the scaling factor
measureOverTimeSales <- measureOverTime
scaledControlSales <- measureOverTimeSales[STORE_NBR == control_store, ][, control_sales := totSales*sca
```

Now that we have comparable sales figures for the control store, we can calculate the percentage difference between the scaled control sales and the trial store's sales during the trial period.

```
percentage_diff <- merge(scaledControlSales[, c("MONTHYEAR" , "control_sales")], measureOverTime[STORE_N
```

Let's see if the difference is significant!

```
# null hypothesis - trail period is same as pretrail period.... lets take standard deviation based on t
stdDev <- sd(percentage_diff[MONTHYEAR < 201902, percentage_diff])
# since there are 8 months in pretrail period hence the degrees of freedom will be
dof <- 8-1
# we will test with a null hypothesis of there being 0 difference between trail and control stores

percentage_diff[, tvalue:= (percentage_diff - 0)/stdDev][, transactionMonth := as.Date(paste(as.numeric
```

```
##    transactionMonth    tvalue
## 1:       2019-02-01  1.183534
## 2:       2019-03-01  7.339116
## 3:       2019-04-01 12.476373
## 4:       2019-05-01  3.023650
## 5:       2019-06-01  3.406093
```

```
# finding the 95th percentile of t-distribution with the appropriate degrees of freedom to check whethe

qt(0.95, df = dof)
```

```
## [1] 1.894579
```

We can observe that the t-value is much larger than the 95th percentile value of the t-distribution for March and April i.e. the increase in sales in the trial store in March and April is statistically greater than in the control store.

Let's create a more visual version of this by plotting the sales of the control store, the sales of the trial stores and the 95th percentile value of sales of the control store.

```
#trail and control store sales
measureOverTimeSales <- measureOverTime
pastSales <-  measureOverTimeSales[, store_type:= ifelse(STORE_NBR == trail_store, "trail store", ifels

# control_store 95th percentile
pastSales_control95 <- pastSales[store_type =="control store",
                                 ][, totSales := totSales*(1 +stdDev*2)
                                   ][, store_type:= "control 95% confidence interval"]
# control_store 5th percentile
pastSales_control5 <- pastSales[store_type =="control store",
][, totSales := totSales*(1 - stdDev*2)
][, store_type:= "control 5% confidence interval"]

trailAssessment <- rbind(pastSales, pastSales_control95, pastSales_control5)



# visualize
ggplot(trailAssessment, aes(transactionMonth, totSales, col = store_type))+
    geom_rect(data = trailAssessment[as.numeric(MONTHYEAR) < 201905 & as.numeric(MONTHYEAR) > 201901, ]
    geom_line(aes(linetype = store_type))+
    labs(x = "MONTH OF OPERATION", y = "TOTAL SALES" , title = "TOTAL SALES BY MONTH")
```

The results show that the trail in store in 77 is signifinactly different to its control store in the trail period as the trail store performance lies outside the 5% to 95% confidence interval of the control store in two of the three trail months

Let's have a look at assessing this for number of customers as well.

```
# it will mostly be repeat of the steps we performed above

#Scale pre-trial control customers to match pre-trial trial store customers
scalingFactorForControlSales <- preTrailMeasures[STORE_NBR==trail_store & MONTHYEAR < 201902, sum(nCust

# finding scaled control customers
measureOverTimeCustomers <- measureOverTime
scaledControlCustomers <- measureOverTimeCustomers[STORE_NBR == control_store, ][, control_customers :=

#finally calculating the precentange difference
percentage_diff <- merge(scaledControlCustomers[, c("MONTHYEAR" , "control_customers")], measureOverTim

# checking whether the difference is significant visually

stdDev <- sd(percentage_diff[MONTHYEAR < 201902, percentage_diff])
# since there are 8 months in pretrail period hence the degrees of freedom will be
dof <- 8-1

# test with a null hypothesis of there being 0 difference between trail and control stores
percentage_diff[, tvalue:= (percentage_diff - 0)/stdDev][, transactionMonth := as.Date(paste(as.numeric
```

```
##    transactionMonth     tvalue
## 1:       2019-02-01  0.1833522
## 2:       2019-03-01 13.4763876
## 3:       2019-04-01 30.7787247
## 4:       2019-05-01  2.1005087
## 5:       2019-06-01  0.1833522
```

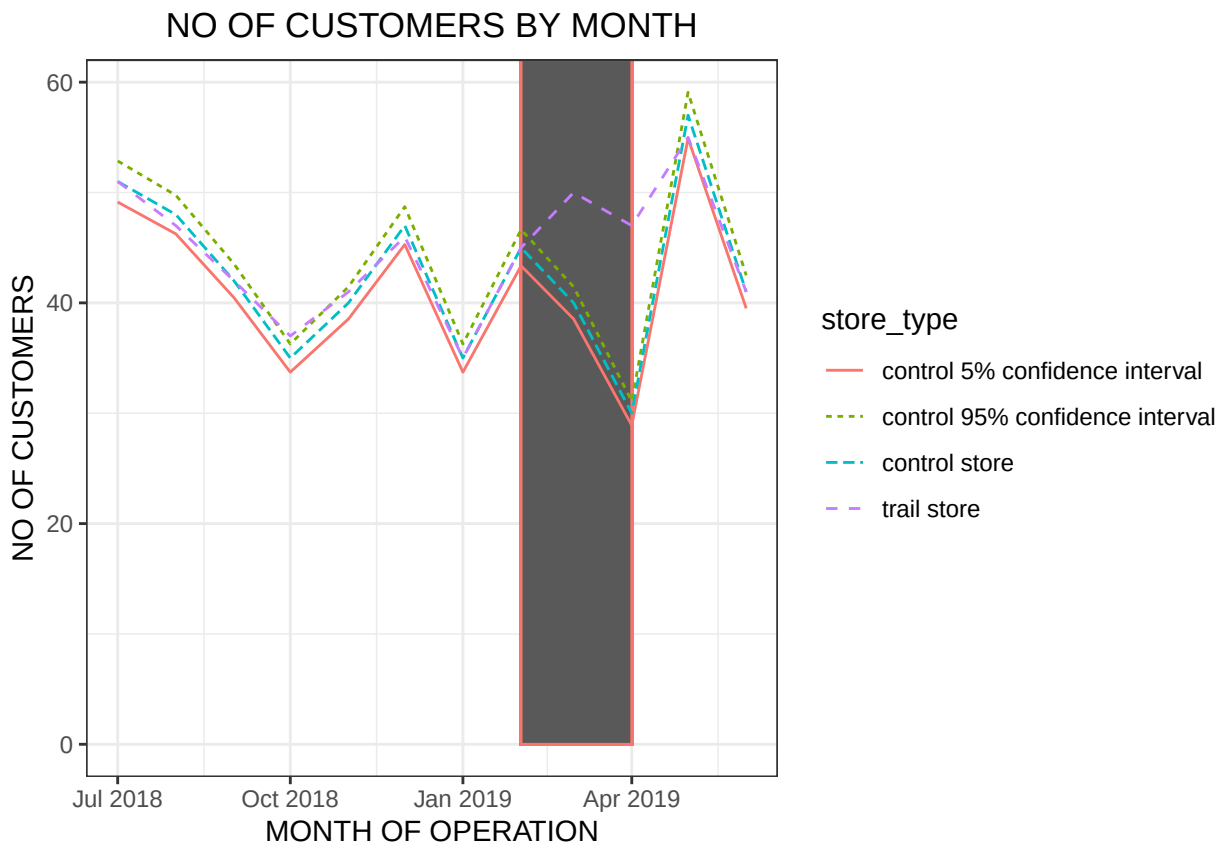
```
# here we can see that for the month of mar-april the tvalues are significantly higher
qt(0.95, dof)
```

```
## [1] 1.894579
```

```
# we can obeserve that the tvalue is much larger than the 95th percentile value the t- distribution for
```

Let's again see if the difference is significant visually!

```
# trail and control store customers
measureOverTimeCustomers <- measureOverTime
pastCustomers <-  measureOverTimeCustomers[, store_type:= ifelse(STORE_NBR == trail_store, "trail store

# control_store 95th percentile
pastCustomers_control95 <- pastCustomers[store_type =="control store",
                                         ][, nCustomers := nCustomers*(1 +stdDev*2)
                                           ][, store_type:= "control 95% confidence interval"]

# control_store 5th percentile
pastCustomers_control5 <- pastCustomers[store_type =="control store", ][, nCustomers := nCustomers*(1 -

trailAssessment <- rbind(pastCustomers, pastCustomers_control95, pastCustomers_control5)

# visualize
ggplot(trailAssessment, aes(transactionMonth, nCustomers, col = store_type))+
    geom_rect(data = trailAssessment[as.numeric(MONTHYEAR) < 201905 & as.numeric(MONTHYEAR) > 201901, ]
    geom_line(aes(linetype = store_type))+
    labs(x = "MONTH OF OPERATION", y = "NO OF CUSTOMERS" , title = "NO OF CUSTOMERS BY MONTH")
```



It looks like the number of customers is significantly higher in all of the three months. This seems to suggest that the trial had a significant impact on increasing the number of customers in trial store 77. Hence during the trail period both sales and the customers increased significantly for the trail store 77.

Let's repeat finding the control store and assessing the impact of the trial for each of the other two trial stores.

TRAIL STORE NO- 86

Basically we have to repeat the above steps but change the trail store value to 86 as we have to perform the above steps for store 86

```
measureOverTime <- data[, .(totSales = sum(TOT_SALES),
                            nCustomers = uniqueN(LYLTY_CARD_NBR),
                            nTxnPerCust = uniqueN(TXN_ID)/uniqueN(LYLTY_CARD_NBR),
                            nChipsPerTxn = sum(PROD_QTY)/uniqueN(LYLTY_CARD_NBR),
                            avgPricePerUnit = sum(TOT_SALES)/sum(PROD_QTY)
), by = c("MONTHYEAR", "STORE_NBR")][order(STORE_NBR, MONTHYEAR)]

storesWithFullObs <- measureOverTime[, .N, STORE_NBR][N == 12, STORE_NBR]
preTrailMeasures <- measureOverTime[MONTHYEAR < 201902 & STORE_NBR %in% storesWithFullObs, ]

#using functions created for finding correlations and magnitude for each potential control store
trail_store<- 86
corr_nSales <- calculateCorr(preTrailMeasures, quote(totSales), trail_store)
corr_nCustomers<- calculateCorr(preTrailMeasures, quote(nCustomers), trail_store)
```

```
magnitude_nsales<- calculateMagnitudeDistance(preTrailMeasures, quote(totSales), trail_store)
magnitude_nCustomers<- calculateMagnitudeDistance(preTrailMeasures, quote(nCustomers), trail_store)

#create a combined score composed of correlation and magnitude

corr_weight<- 0.5
score_nSales <-  merge(corr_nSales, magnitude_nsales, by = c("Store1", "Store2"))[, score_nSales := 0.5
score_nCustomers<- merge(corr_nCustomers, magnitude_nCustomers, by = c("Store1", "Store2"))[, score_nCus

# Finally, combine scores across the drivers using a simple average.
score_control <- merge(score_nSales, score_nCustomers, by = c("Store1", "Store2"))
score_control[, finalControlStore := score_nSales*0.5+score_nCustomers*0.5]

# the store with the second highest score is then selected as the control since it is most similar to t

control_store <- score_control[order(-finalControlStore), ]
(control_store <- control_store[,Store2][[2]])
```

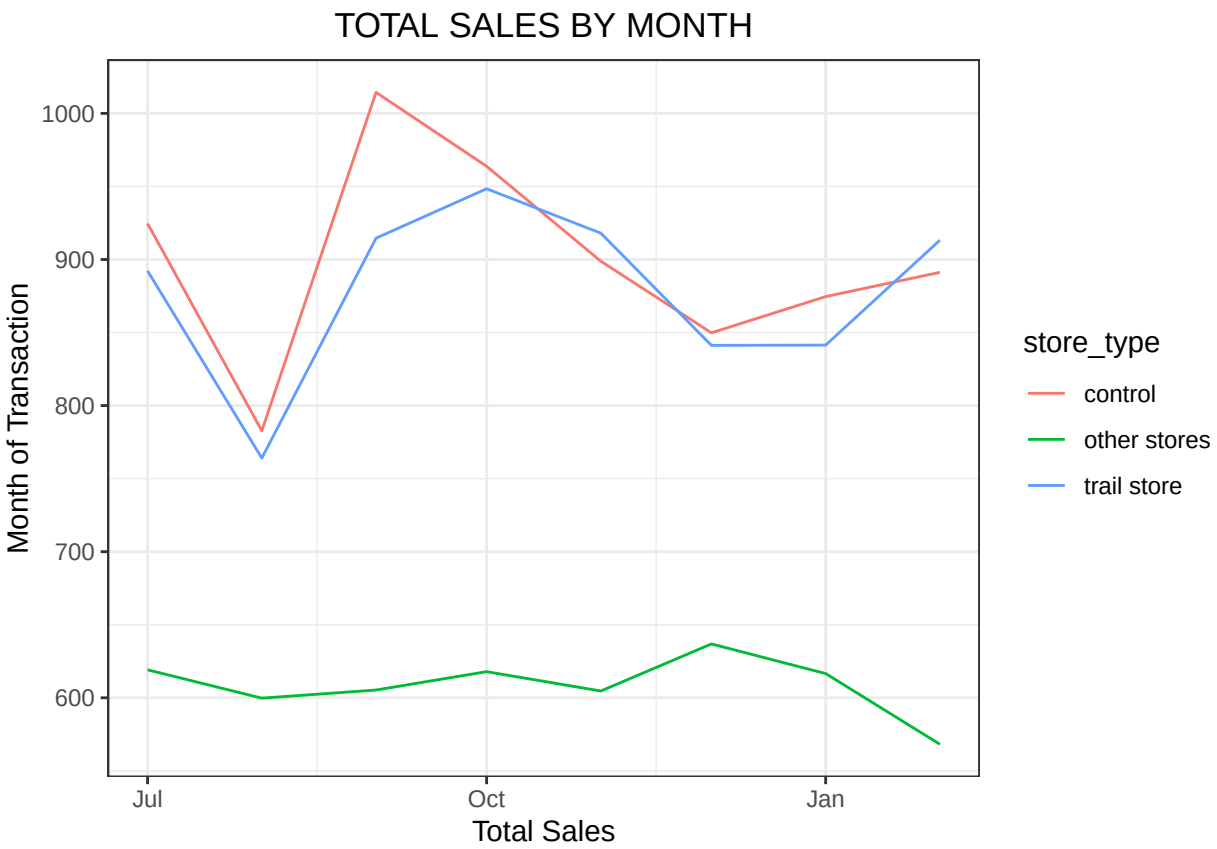
```
## [1] 155
```

Looks like store 155 will be a control store for trial store 86.

Again, let's check visually if the drivers are indeed similar in the period before the trial.

We'll look at total sales first.

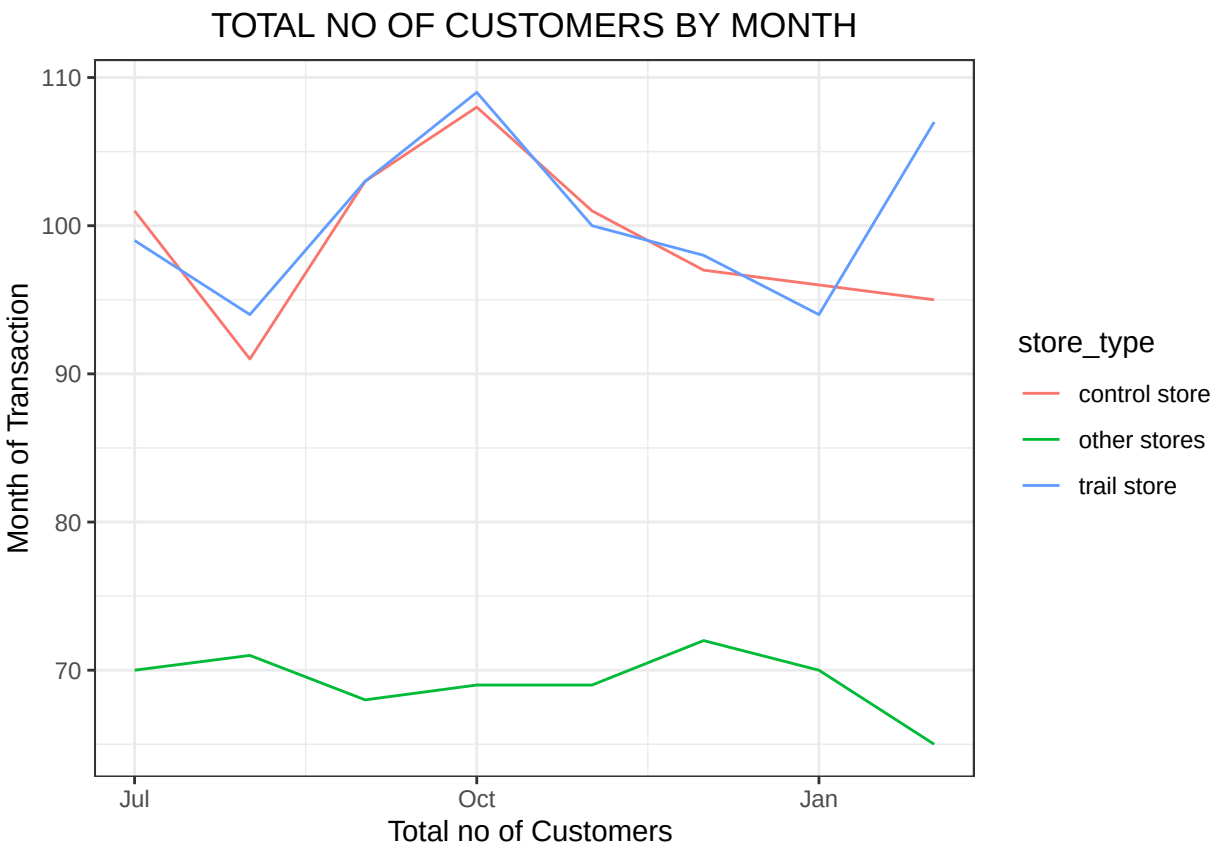
```
measureOverTimeSales <- measureOverTime
pastSales <- measureOverTimeSales[, store_type := ifelse(STORE_NBR == trail_store, "trail store", ifels

# plotting the data
ggplot(pastSales, aes(transactionMonth, totSales , col = store_type))+ geom_line()+ labs(x = "Total Sal
```



Sales are trending in a similar way

Next, number of customers.

```
pastCustomers <- measureOverTimeSales[, store_type:= ifelse(STORE_NBR == trail_store, "trail store", if

# plotting the data
ggplot(pastCustomers, aes(transactionMonth, nCustomers , col = store_type))+ geom_line()+ labs(x = "Tota
```

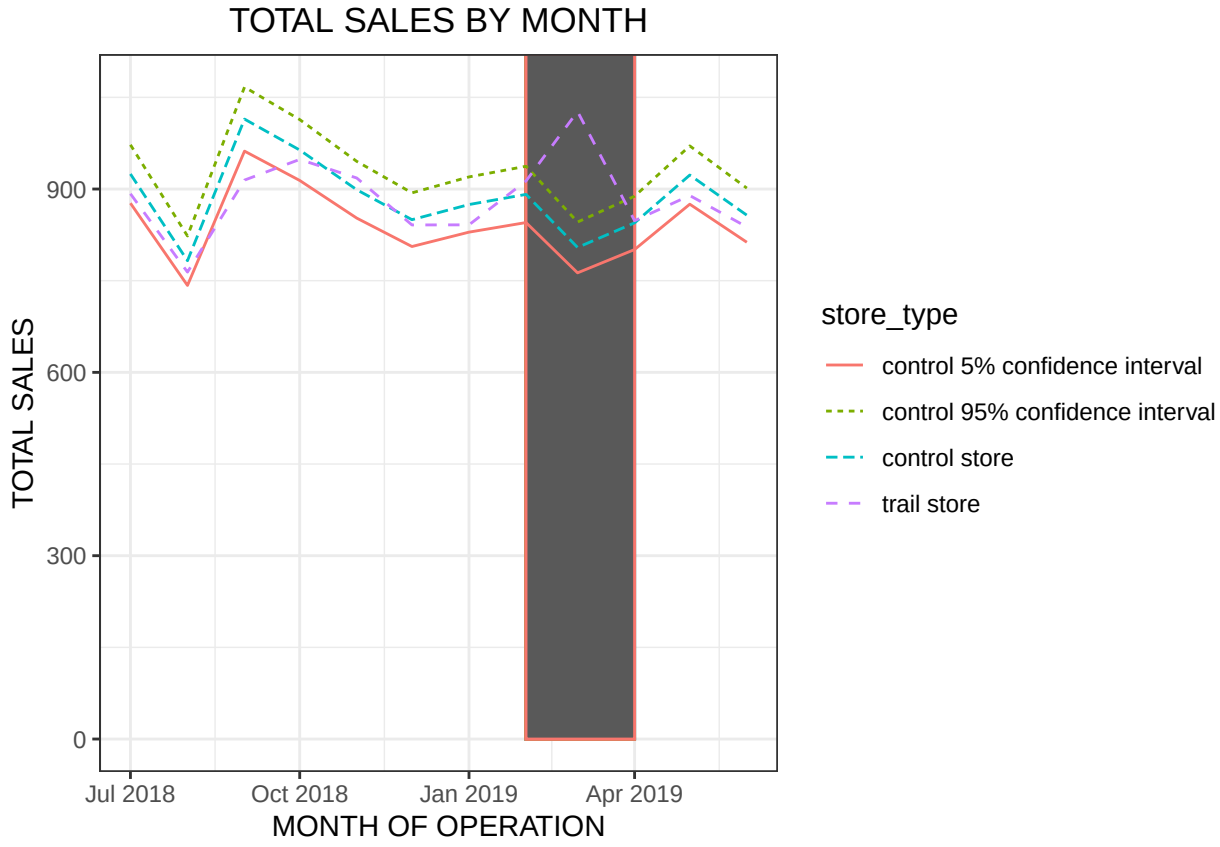


The trend in number of customers is also similar.

Let's now assess the impact of the trial on sales.

```
## scale pre-trail control sales to match pre-trail store sales
scalingFactorForControlSales <- preTrailMeasures[STORE_NBR==trail_store & MONTHYEAR < 201902, sum(totSal

## apply the scaling factor
measureOverTimeSales <- measureOverTime
scaledControlSales <- measureOverTimeSales[STORE_NBR == control_store, ][, control_sales := totSales*sca

# now that we have comparable sales figure for the control store , we can calculate the percentage diff

# calculating percentage difference between scaled control sales and trial sales

measureOverTime <- measureOverTime
percentage_diff <- merge(scaledControlSales[, c("MONTHYEAR" , "control_sales")], measureOverTime[STORE_

# As our null hypothesis is that the trial period is the same as the pre-trial period, let's take the s

stdDev <- sd(percentage_diff[MONTHYEAR < 201902, percentage_diff])
# since there are 8 months in pretrail period hence the degrees of freedom will be
dof <- 8-1
#lets create a more visual version of this by plotting the sales of control store and the sales of trai
#trail and control store sales

measureOverTimeSales <- measureOverTime
```

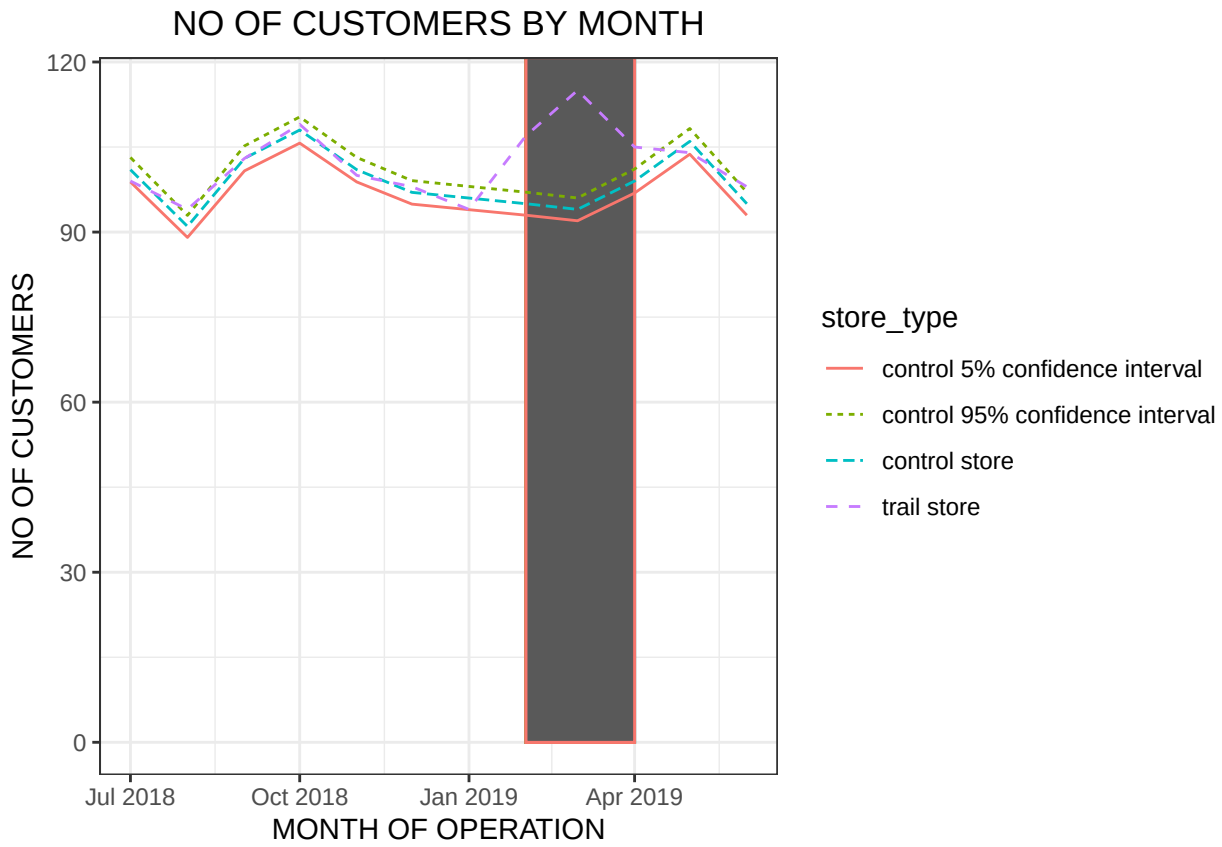
```
pastSales <-  measureOverTimeSales[, store_type:= ifelse(STORE_NBR == trail_store, "trail store", ifels

# control_store 95th percentile
pastSales_control95 <- pastSales[store_type =="control store",
][, totSales := totSales*(1 +stdDev*2)
][, store_type:= "control 95% confidence interval"]
# control_store 5th percentile
pastSales_control5 <- pastSales[store_type =="control store",
][, totSales := totSales*(1 - stdDev*2)
][, store_type:= "control 5% confidence interval"]

trailAssessment <- rbind(pastSales, pastSales_control95, pastSales_control5)

# visualize
ggplot(trailAssessment, aes(transactionMonth, totSales, col = store_type))+
    geom_rect(data = trailAssessment[as.numeric(MONTHYEAR) < 201905 & as.numeric(MONTHYEAR) > 201901, ]
    geom_line(aes(linetype = store_type))+
    labs(x = "MONTH OF OPERATION", y = "TOTAL SALES" , title = "TOTAL SALES BY MONTH")
```



The results show that the trial in store 86 is significantly different to its control store in the trial period as the trial store performance lies outside of the 5% to 95% confidence interval of the control store in two of the three trial months.

```
#Scale pre-trial control customers to match pre-trial trial store customers
scalingFactorForControlSales <- preTrailMeasures[STORE_NBR==trail_store & MONTHYEAR < 201902, sum(nCust
```

```r
# finding scaled control customers
measureOverTimeCustomers <- measureOverTime
scaledControlCustomers <- measureOverTimeCustomers[STORE_NBR == control_store, ][, control_customers :=

#finally calculating the precentange difference
percentage_diff <- merge(scaledControlCustomers[, c("MONTHYEAR" , "control_customers")], measureOverTim

# checking whether the difference is significant visually

stdDev <- sd(percentage_diff[MONTHYEAR < 201902, percentage_diff])
# since there are 8 months in pretrail period hence the degrees of freedom will be
dof <- 8-1

measureOverTimeCustomers <- measureOverTime
pastCustomers <-  measureOverTimeCustomers[, store_type:= ifelse(STORE_NBR == trail_store, "trail store

# control_store 95th percentile
pastCustomers_control95 <- pastCustomers[store_type =="control store",
][, nCustomers := nCustomers*(1 +stdDev*2)
][, store_type:= "control 95% confidence interval"]

# control_store 5th percentile
pastCustomers_control5 <- pastCustomers[store_type =="control store", ][, nCustomers := nCustomers*(1 -

trailAssessment <- rbind(pastCustomers, pastCustomers_control95, pastCustomers_control5)

# visualize
ggplot(trailAssessment, aes(transactionMonth, nCustomers, col = store_type))+
    geom_rect(data = trailAssessment[as.numeric(MONTHYEAR) < 201905 & as.numeric(MONTHYEAR) > 201901, ]
    geom_line(aes(linetype = store_type))+
    labs(x = "MONTH OF OPERATION", y = "NO OF CUSTOMERS" , title = "NO OF CUSTOMERS BY MONTH")
```



It looks like the number of customers is significantly higher in all of the three months. This seems to suggest that the trial had a significant impact on increasing the number of customers in trial store 86

TRAIL STORE- 88

```
measureOverTime <- data[, .(totSales = sum(TOT_SALES),
                            nCustomers = uniqueN(LYLTY_CARD_NBR),
                            nTxnPerCust = uniqueN(TXN_ID)/uniqueN(LYLTY_CARD_NBR),
                            nChipsPerTxn = sum(PROD_QTY)/uniqueN(LYLTY_CARD_NBR),
                            avgPricePerUnit = sum(TOT_SALES)/sum(PROD_QTY)
), by = c("MONTHYEAR", "STORE_NBR")][order(STORE_NBR, MONTHYEAR)]

storesWithFullObs <- measureOverTime[, .N, STORE_NBR][N == 12, STORE_NBR]
preTrailMeasures <- measureOverTime[MONTHYEAR < 201902 & STORE_NBR %in% storesWithFullObs, ]

#using functions created for finding correlations and magnitude for each potential control store
trail_store<- 88
corr_nSales <- calculateCorr(preTrailMeasures, quote(totSales), trail_store)
corr_nCustomers<- calculateCorr(preTrailMeasures, quote(nCustomers), trail_store)

magnitude_nsales<- calculateMagnitudeDistance(preTrailMeasures, quote(totSales), trail_store)
magnitude_nCustomers<- calculateMagnitudeDistance(preTrailMeasures, quote(nCustomers), trail_store)

#create a combined score composed of correlation and magnitude
```

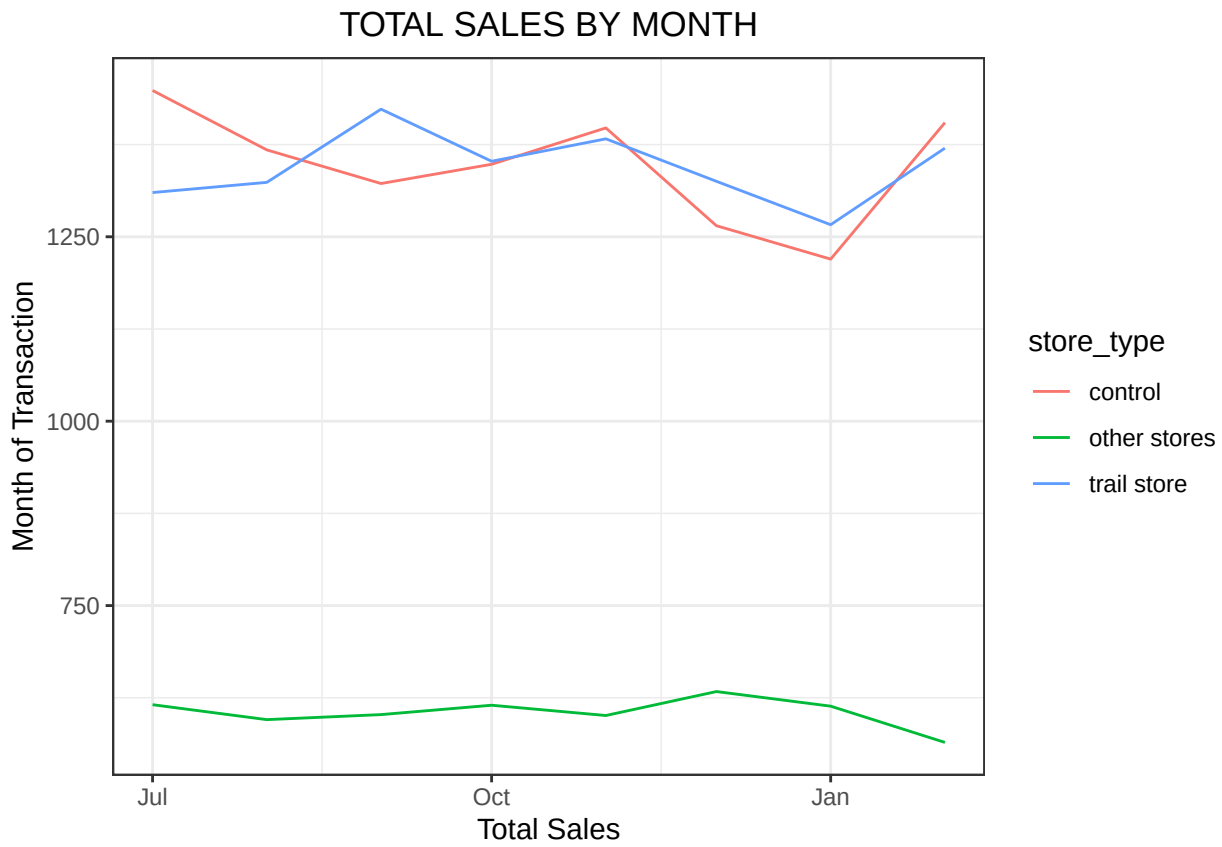
```
corr_weight<- 0.5
score_nSales <-  merge(corr_nSales, magnitude_nsales, by = c("Store1", "Store2"))[, score_nSales := 0.5
score_nCustomers<- merge(corr_nCustomers, magnitude_nCustomers, by = c("Store1", "Store2"))[, score_nCus

# Finally, combine scores across the drivers using a simple average.
score_control <- merge(score_nSales, score_nCustomers, by = c("Store1", "Store2"))
score_control[, finalControlStore := score_nSales*0.5+score_nCustomers*0.5]

# the store with the second highest score is then selected as the control since it is most similar to t

control_store <- score_control[order(-finalControlStore), ]
(control_store <- control_store[,Store2][[2]])
```

```
## [1] 237
```

We've now found store 237 to be a suitable control store for trial store 88.

Again, let's check visually if the drivers are indeed similar in the period before the trial

```
# first we'll check for sales
measureOverTimeSales <- measureOverTime
pastSales <- measureOverTimeSales[, store_type := ifelse(STORE_NBR == trail_store, "trail store", ifels

# plotting the data
ggplot(pastSales, aes(transactionMonth, totSales , col = store_type))+ geom_line()+ labs(x = "Total Sal
```

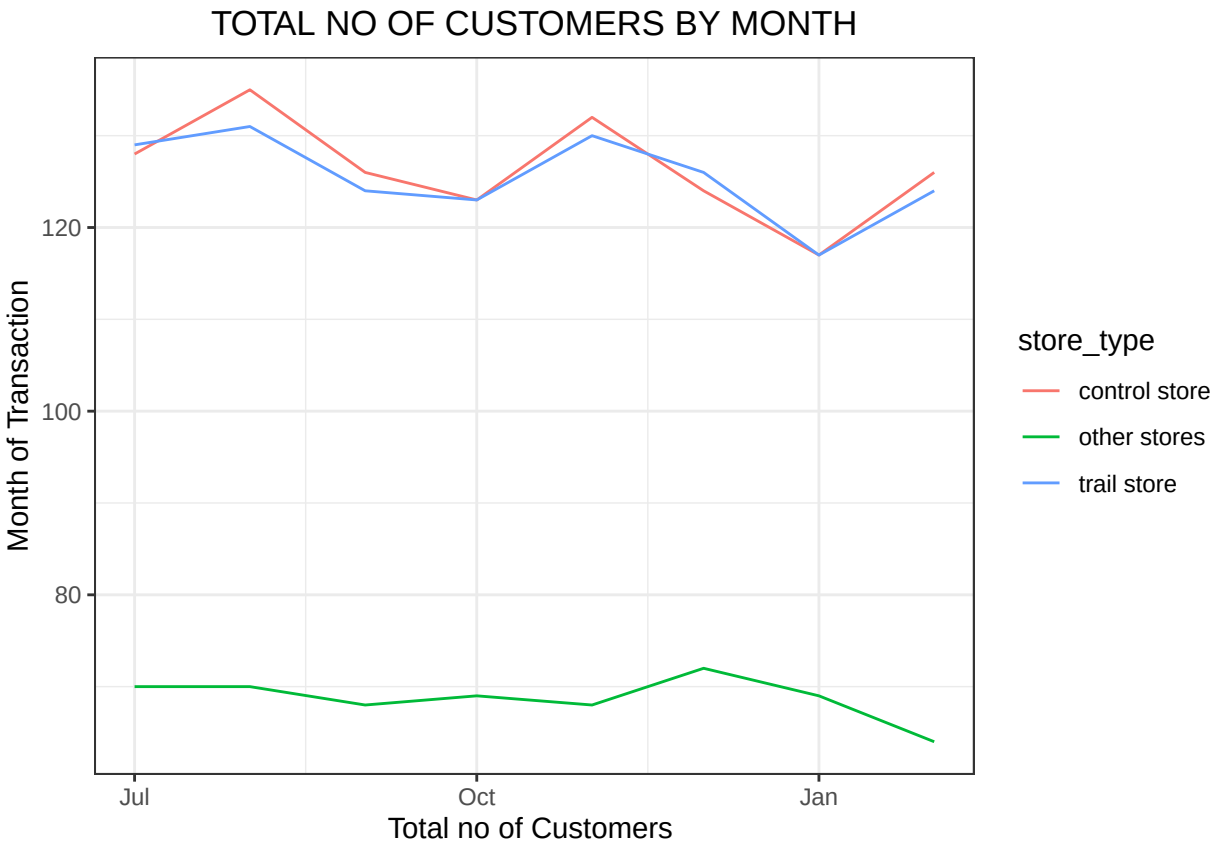


sales are trending in a similar way.

```r
# for Customers
pastCustomers <- measureOverTimeSales[, store_type:= ifelse(STORE_NBR == trail_store, "trail store", if
```

```
## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 1 column 'nCustomers': 70.162879 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 2 column 'nCustomers': 70.697318 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 3 column 'nCustomers': 68.477099 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 4 column 'nCustomers': 69.673004 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 5 column 'nCustomers': 68.843511 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 6 column 'nCustomers': 72.130268 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 7 column 'nCustomers': 69.842912 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 8 column 'nCustomers': 64.881679 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 9 column 'nCustomers': 70.889734 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 10 column 'nCustomers': 68.121673 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 11 column 'nCustomers': 70.310345 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'

## Warning in `[.data.table`(measureOverTimeSales[, `:=`(store_type,
## ifelse(STORE_NBR == : Group 12 column 'nCustomers': 68.793893 (type 'double') at
## RHS position 1 truncated (precision lost) when assigning to type 'integer'
```

```
# plotting the data
ggplot(pastCustomers, aes(transactionMonth, nCustomers , col = store_type))+ geom_line()+ labs(x = "Tota
```



The trend in number of customers is also similar

Let's now assess the impact of the trial on sales

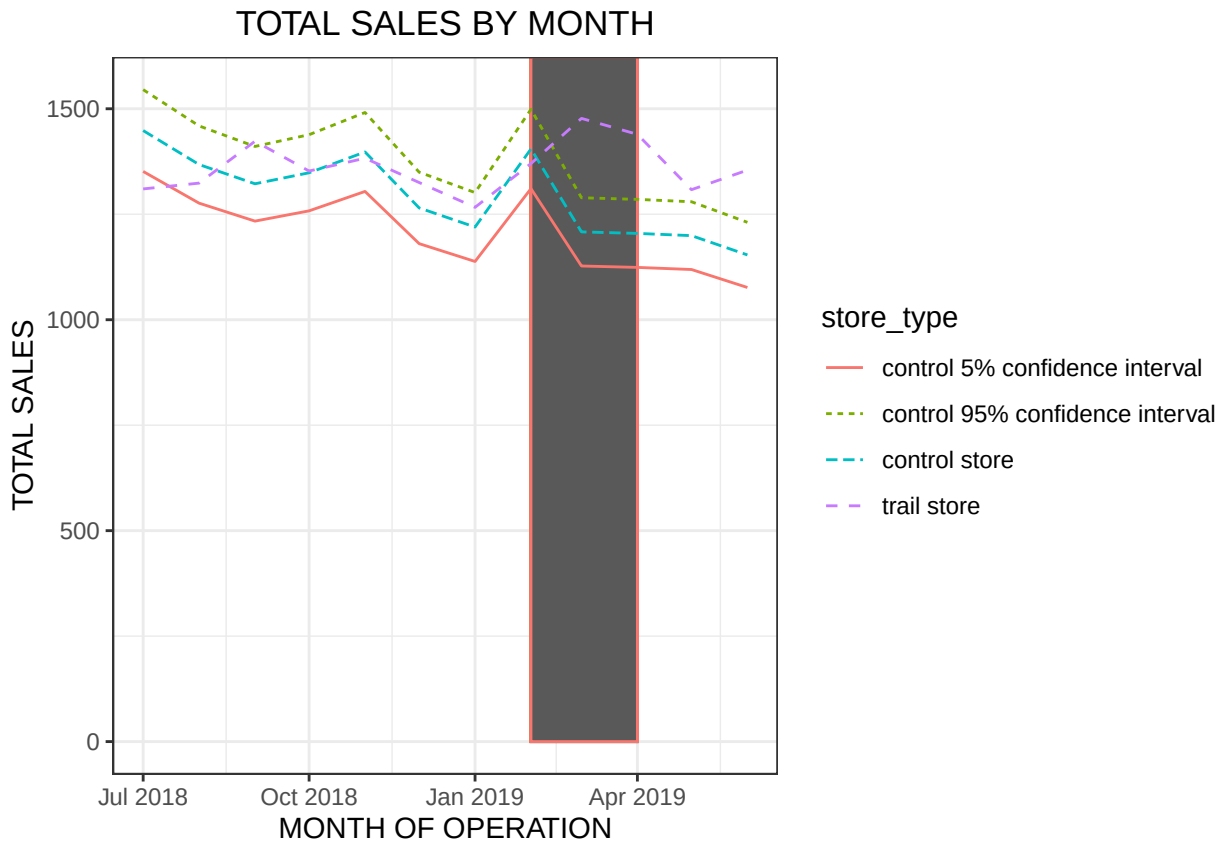
```
## scale pre-trail control sales to match pre-trail store sales
scalingFactorForControlSales <- preTrailMeasures[STORE_NBR==trail_store & MONTHYEAR < 201902, sum(totSal

## apply the scaling factor
measureOverTimeSales <- measureOverTime
scaledControlSales <- measureOverTimeSales[STORE_NBR == control_store, ][, control_sales := totSales*sca
```

```
# now that we have comparable sales figure for the control store , we can calculate the percentage diff
# calculating percentage difference between scaled control sales and trial sales
measureOverTime <- as.data.table(measureOverTime)
percentage_diff <- merge(scaledControlSales[, c("MONTHYEAR" , "control_sales")], measureOverTime[STORE_
```

Lets see if the difference is significant

```
# null hypothesis - trail period is same as pretrail period.... lets take standard deviation based on t
stdDev <- sd(percentage_diff[MONTHYEAR < 201902, percentage_diff])
# since there are 8 months in pretrail period hence the degrees of freedom will be
dof <- 8-1
#lets create a more visual version of this by plotting the sales of control store and the sales of trai
```

```
#trail and control store sales
measureOverTimeSales <- measureOverTime
pastSales <-  measureOverTimeSales[, store_type:= ifelse(STORE_NBR == trail_store, "trail store", ifels

# control_store 95th percentile
pastSales_control95 <- pastSales[store_type =="control store",
][, totSales := totSales*(1 +stdDev*2)
][, store_type:= "control 95% confidence interval"]
# control_store 5th percentile
pastSales_control5 <- pastSales[store_type =="control store",
][, totSales := totSales*(1 - stdDev*2)
][, store_type:= "control 5% confidence interval"]

trailAssessment <- rbind(pastSales, pastSales_control95, pastSales_control5)



# visualize
ggplot(trailAssessment, aes(transactionMonth, totSales, col = store_type))+
    geom_rect(data = trailAssessment[as.numeric(MONTHYEAR) < 201905 & as.numeric(MONTHYEAR) > 201901, ]
    geom_line(aes(linetype = store_type))+
    labs(x = "MONTH OF OPERATION", y = "TOTAL SALES" , title = "TOTAL SALES BY MONTH")
```



The results show that the trail in store 88 is signifinactly different to its control store in the trail period as the trail store performance lies outside the 5% to 95% confidence interval of the control store in two of the three trail months

Lets have a look at assessing this for number of customers as well

```
# it will mostly be repeat of the steps we performed above

#Scale pre-trial control customers to match pre-trial trial store customers
scalingFactorForControlSales <- preTrailMeasures[STORE_NBR==trail_store & MONTHYEAR < 201902, sum(nCust

# finding scaled control customers
measureOverTimeCustomers <- measureOverTime
scaledControlCustomers <- measureOverTimeCustomers[STORE_NBR == control_store, ][, control_customers :=

#finally calculating the precentange difference
percentage_diff <- merge(scaledControlCustomers[, c("MONTHYEAR" , "control_customers")], measureOverTim

# checking whether the difference is significant visually

stdDev <- sd(percentage_diff[MONTHYEAR < 201902, percentage_diff])
# since there are 8 months in pretrail period hence the degrees of freedom will be
dof <- 8-1

measureOverTimeCustomers <- measureOverTime
pastCustomers <-  measureOverTimeCustomers[, store_type:= ifelse(STORE_NBR == trail_store, "trail store

# control_store 95th percentile
pastCustomers_control95 <- pastCustomers[store_type =="control store",
][, nCustomers := nCustomers*(1 +stdDev*2)
][, store_type:= "control 95% confidence interval"]

# control_store 5th percentile
pastCustomers_control5 <- pastCustomers[store_type =="control store", ][, nCustomers := nCustomers*(1 -

trailAssessment <- rbind(pastCustomers, pastCustomers_control95, pastCustomers_control5)

# visualize
ggplot(trailAssessment, aes(transactionMonth, nCustomers, col = store_type))+
    geom_rect(data = trailAssessment[as.numeric(MONTHYEAR) < 201905 & as.numeric(MONTHYEAR) > 201901, ]
    geom_line(aes(linetype = store_type))+
    labs(x = "MONTH OF OPERATION", y = "NO OF CUSTOMERS" , title = "NO OF CUSTOMERS BY MONTH")
```

It looks like the number of customers is significantly higher in all of the three months. This seems to suggest that the trial had a significant impact on increasing the number of customers in trial store 88

Conclusion

We've found control stores 233, 155, 237 for trial stores 77, 86 and 88 respectively. The results for trial stores 77, 86 and 88 during the trial period show a significant difference in at least two of the three trial months.

For Task 1 click on the following link