A little bit of analysis

Read in the data

Due to the inconsistent column naming covention, we manually convert the cx column to cX in an effort to reduce obfuscation.

```
# Read in the data
data <- read.csv('Greatest_Aussie_Groceries_sales_data.csv', header=TRUE, sep=",")
# Change column name of cx to match style of capital X and Y
colnames(data)[colnames(data)=="cx"] <- "cX"
```

Mutate Data

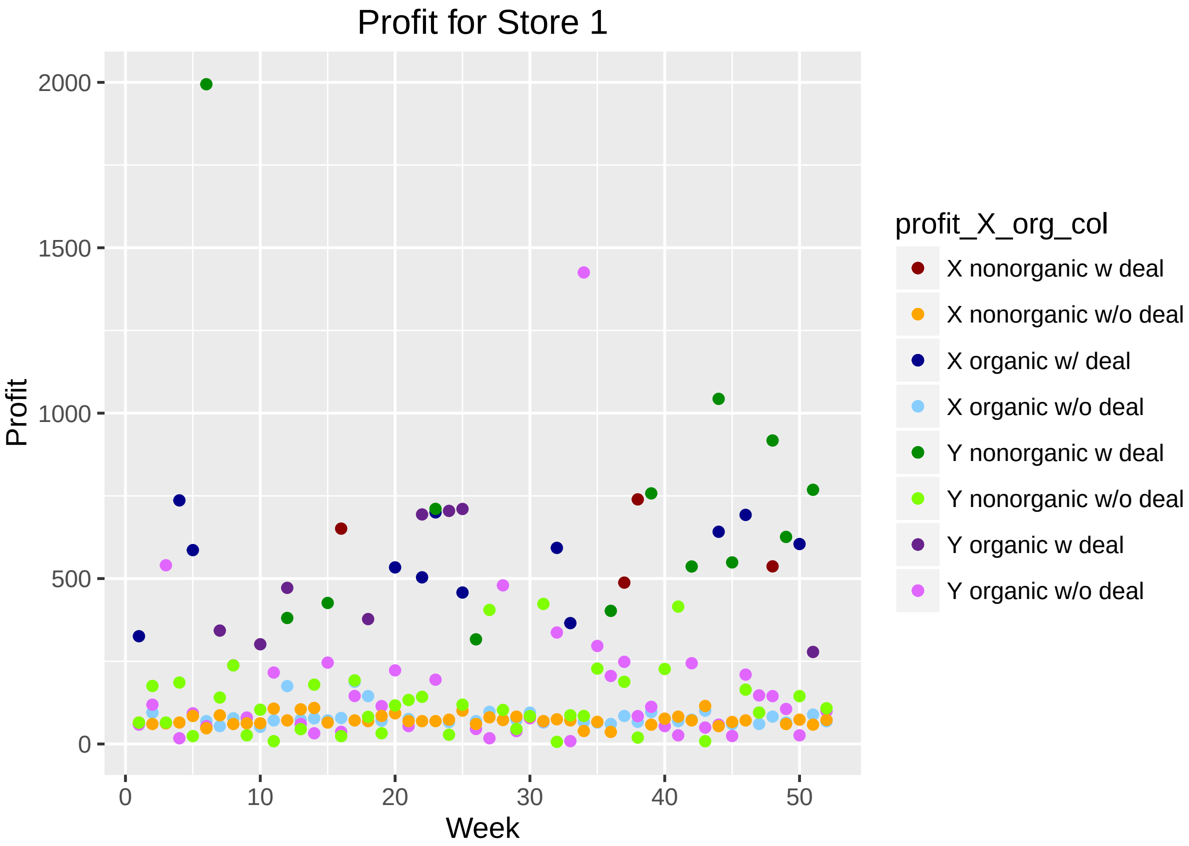
We mutate the data to include columns for deal_feat (an indictator for both deal and features), revenue, and profit.

```
# Append a deal_feat column for X and Y
data <- mutate(data, deal_feat_Y = deal_Y*10 + feat_Y, deal_feat_X = deal_X*10 + feat_X)
# Append a revenue column for X and Y
data <- mutate(data, rev_X = oz_X * pX, rev_Y = oz_Y * pY)
# Append a profit column for X and Y
data <- mutate(data, profit_X = rev_X - cX * pX, profit_Y = rev_Y - cY * pY)
```

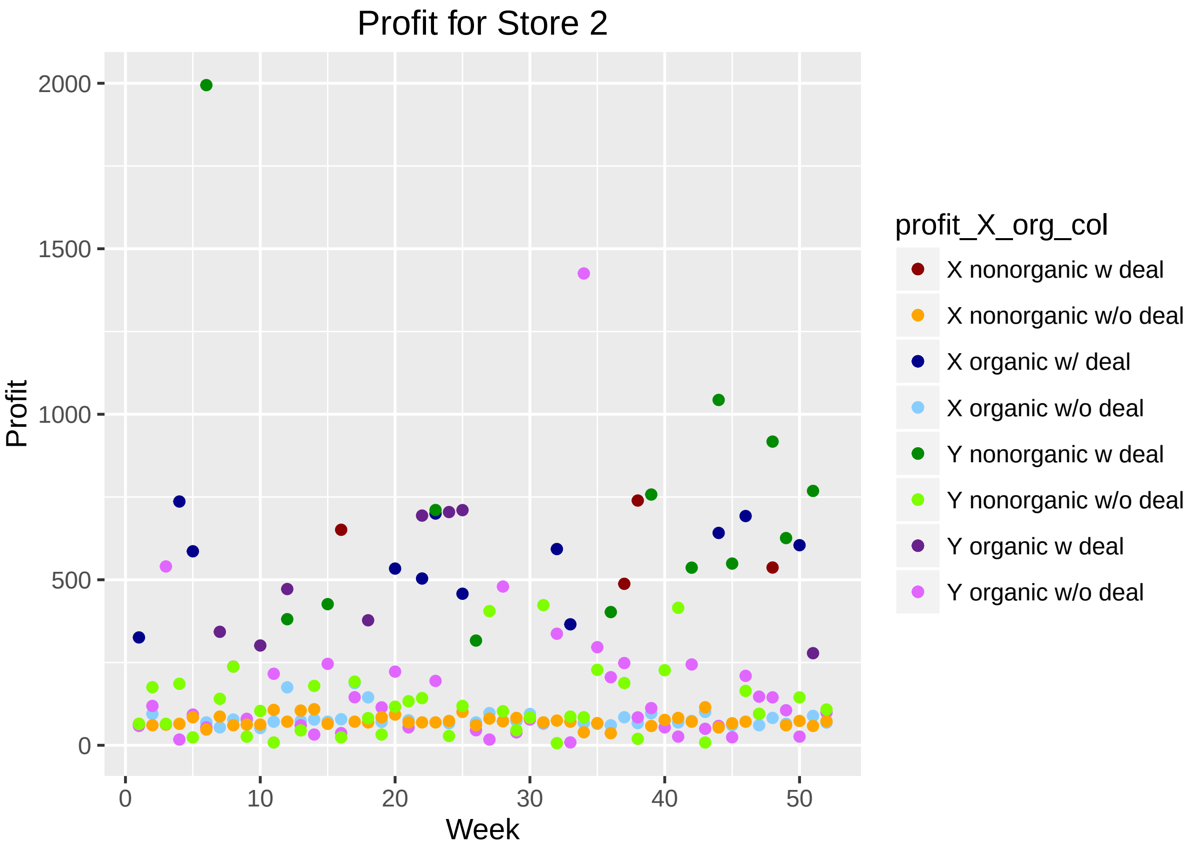
Summary Plots

Lets start by just plotting the profit over the 52 weeks for each store.

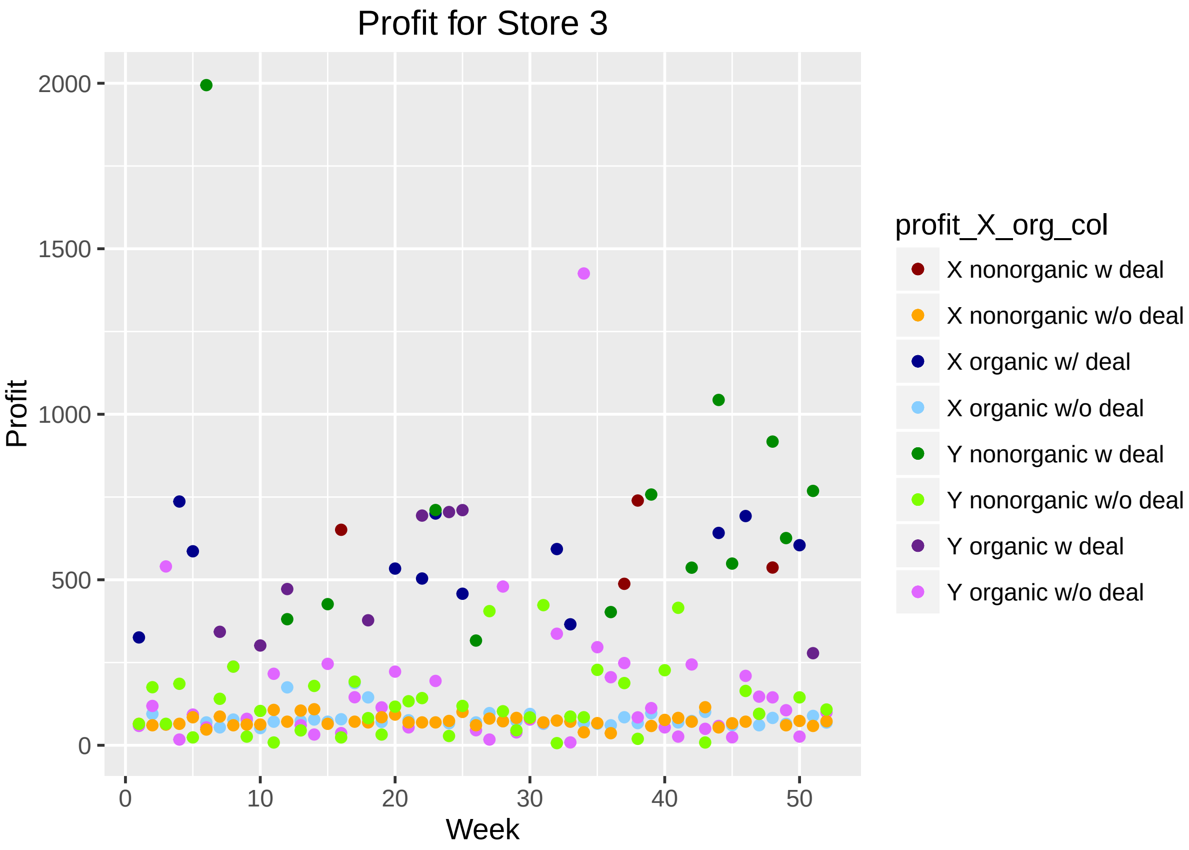
```
plotStore <- function(STORE) {
# The palette with black:
cbbPalette <- c("red4", "orange1", "blue4", "skyblue1", "green4", "chartreuse1", "darkorchid4", "medium
# Pull data into temp results dataframe
results <- data.frame(WEEK=c(1:52))
results[,c("profit_X_org","profit_Y_org")] <- data %>% filter(.,STORE==STORE, class=="organic") %>% sel
results[,c("profit_X_non","profit_Y_non")] <- data %>% filter(.,STORE==STORE, class=="nonorganic") %>%
results[,c("profit_X_org_col","profit_Y_org_col")] <- data %>% filter(.,STORE==STORE, class=="organic")
results[,c("profit_X_non_col","profit_Y_non_col")] <- data %>% filter(.,STORE==STORE, class=="nonorgani
# Assign legend name to categorical data
results$profit_X_org_col[results$profit_X_org_col == 0] <- "X organic w/o deal"
results$profit_X_org_col[results$profit_X_org_col == 1] <- "X organic w/ deal"
results$profit_Y_org_col[results$profit_Y_org_col == 0] <- "Y organic w/o deal"
results$profit_Y_org_col[results$profit_Y_org_col == 1] <- "Y organic w deal"
results$profit_X_non_col[results$profit_X_non_col == 0] <- "X nonorganic w/o deal"
results$profit_X_non_col[results$profit_X_non_col == 1] <- "X nonorganic w deal"
results$profit_Y_non_col[results$profit_Y_non_col == 0] <- "Y nonorganic w/o deal"
results$profit_Y_non_col[results$profit_Y_non_col == 1] <- "Y nonorganic w deal"
# Plot results
ggplot(results, aes(x=WEEK)) +
  geom_point(aes(y=profit_X_org, colour=profit_X_org_col)) +
  geom_point(aes(y=profit_Y_org, colour=profit_Y_org_col)) +
  geom_point(aes(y=profit_X_non, colour=profit_X_non_col)) +
  geom_point(aes(y=profit_Y_non, colour=profit_Y_non_col)) +
  scale_colour_manual(values=cbbPalette) +
  labs(x = "Week", y = "Profit", title = paste("Profit for Store",STORE))
}

# Plot the data
plotStore(1)
```



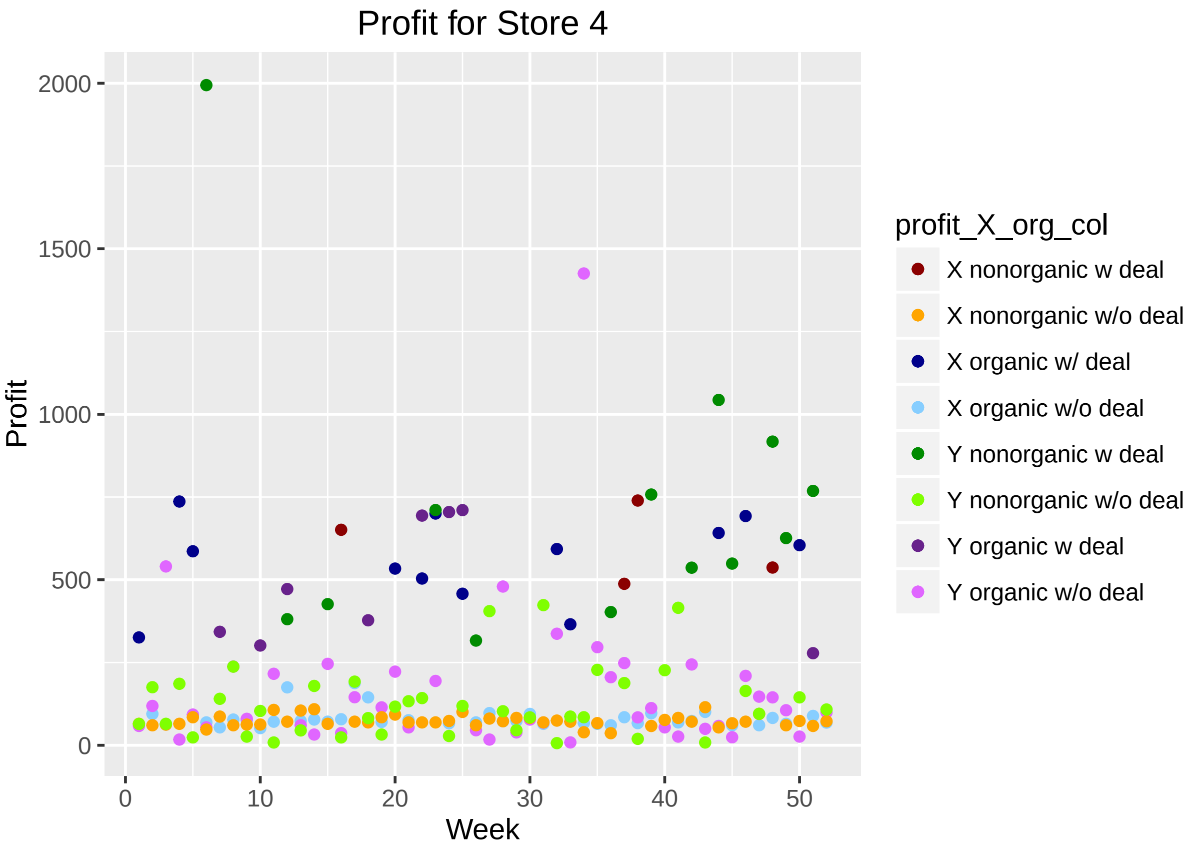
```
plotStore(2)
```


Profit for Store 2
Profit
2000
1500
1000
500
0
Week
0
10
20
30
40
50
profit_X_org_col
● X nonorganic w deal
● X nonorganic w/o deal
● X organic w/ deal
● X organic w/o deal
● Y nonorganic w deal
● Y nonorganic w/o deal
● Y organic w deal
● Y organic w/o deal

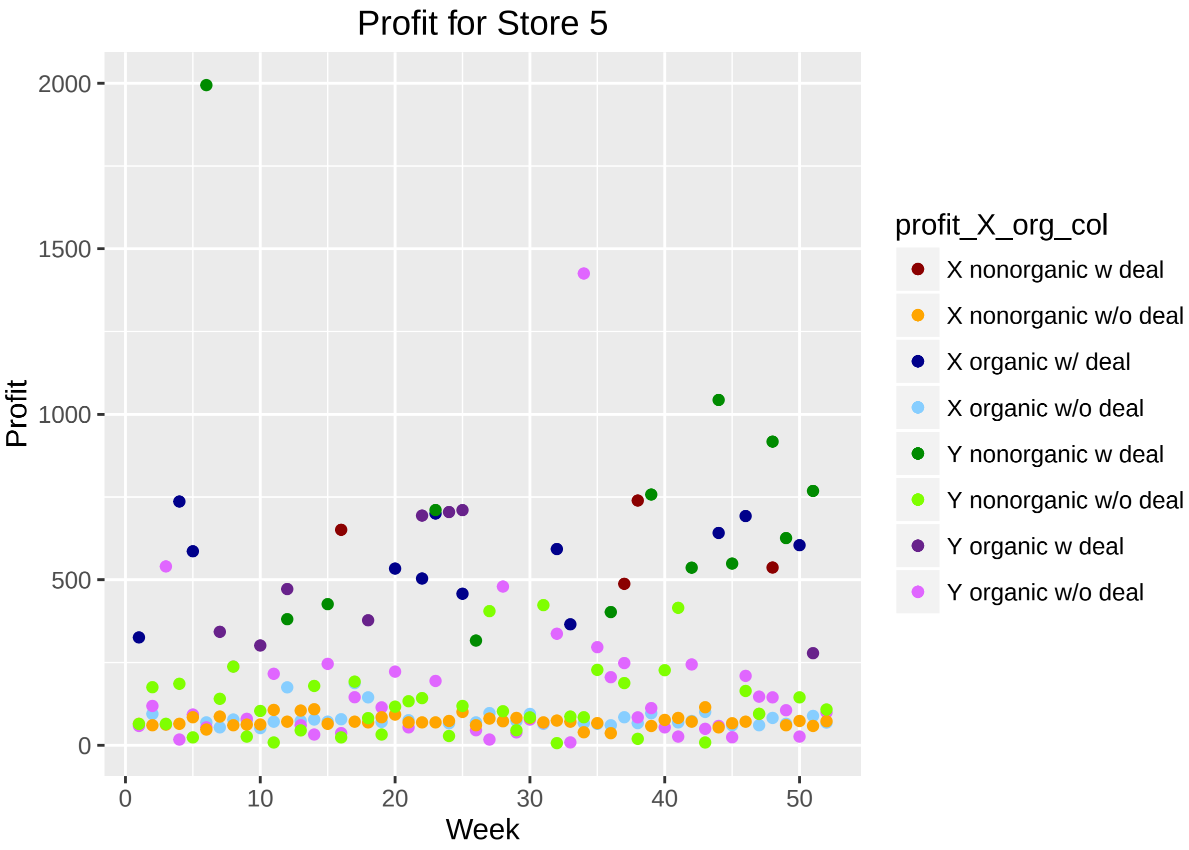
```
plotStore(3)
```


Profit for Store 3
Profit
2000
1500
1000
500
0
Week
0
10
20
30
40
50
profit_X_org_col
● X nonorganic w deal
● X nonorganic w/o deal
● X organic w/ deal
● X organic w/o deal
● Y nonorganic w deal
● Y nonorganic w/o deal
● Y organic w deal
● Y organic w/o deal

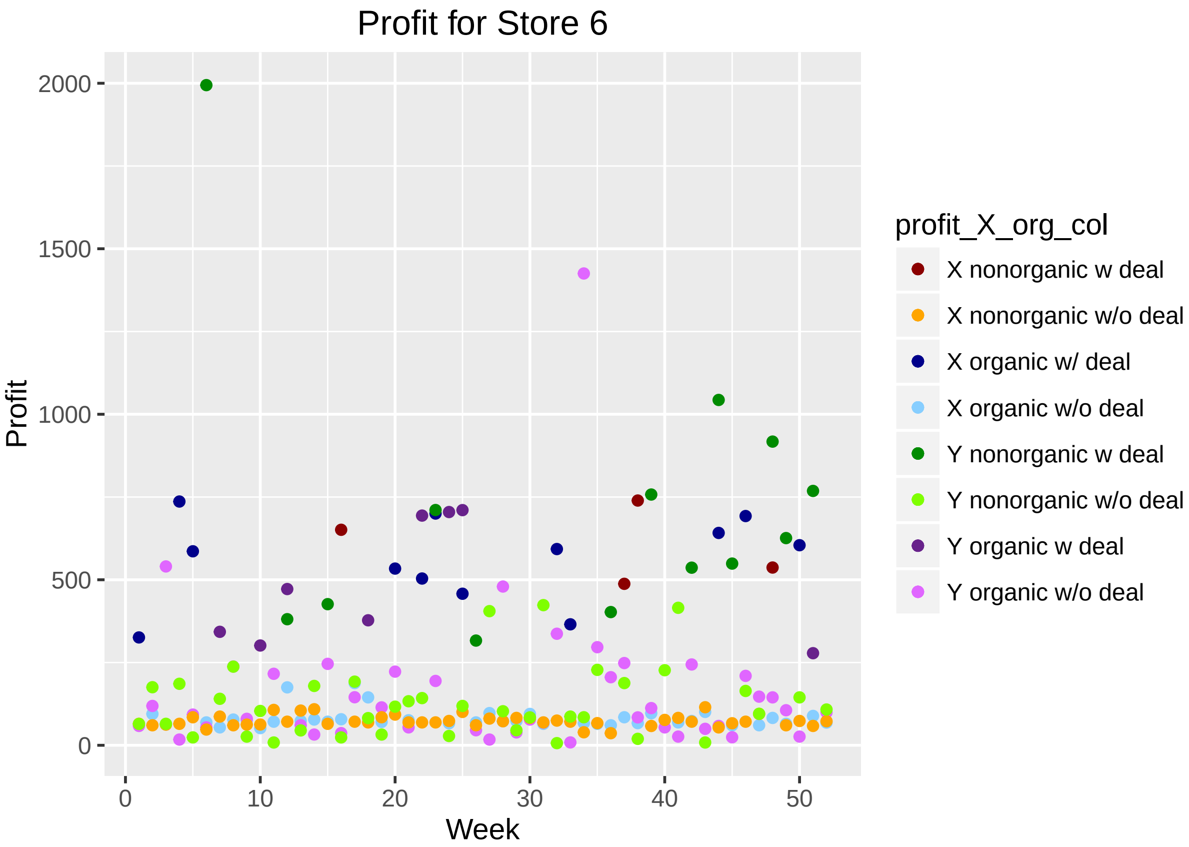
```
plotStore(4)
```


Profit for Store 4
Profit
2000
1500
1000
500
0
0
10
20
30
40
50
Week
profit_X_org_col
X nonorganic w deal
X nonorganic w/o deal
X organic w/ deal
X organic w/o deal
Y nonorganic w deal
Y nonorganic w/o deal
Y organic w deal
Y organic w/o deal

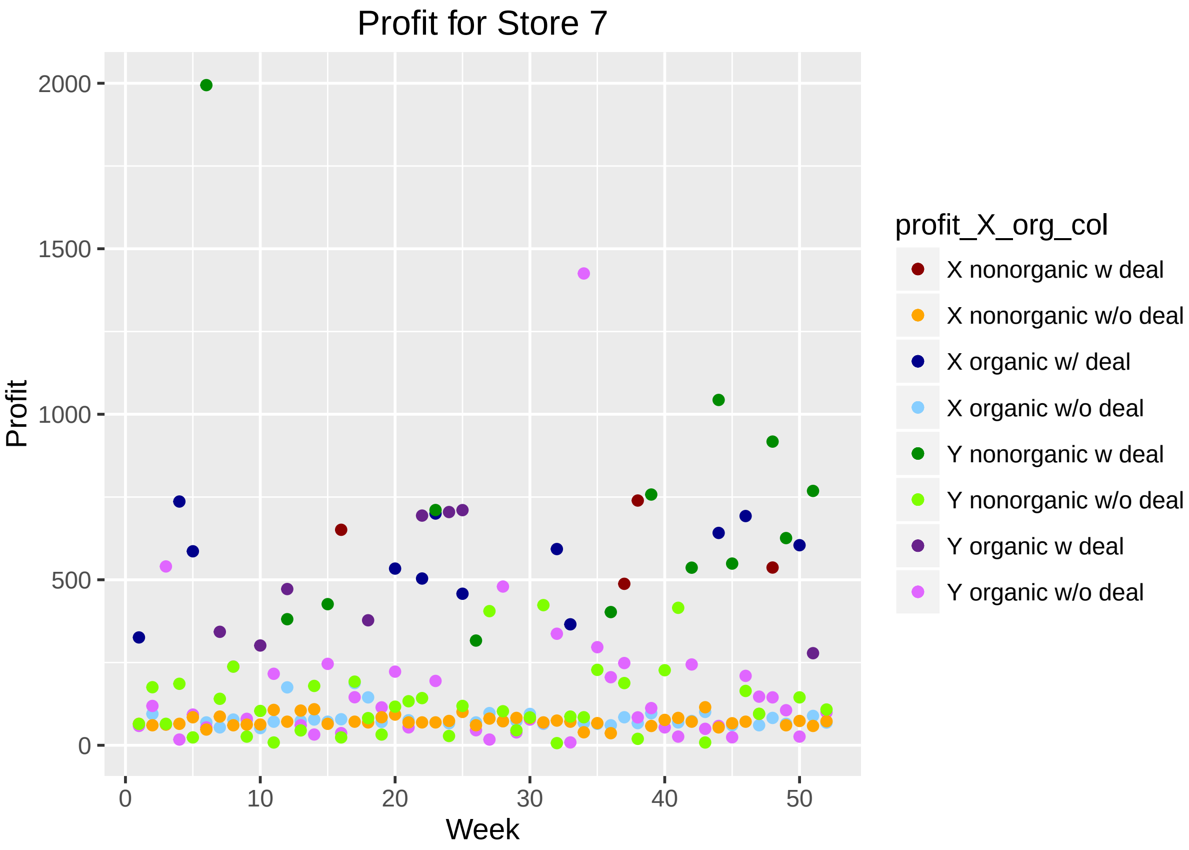
```
plotStore(5)
```


Profit for Store 5
2000
1500
1000
500
0
Profit
0
10
20
30
40
50
Week
profit_X_org_col
X nonorganic w deal
X nonorganic w/o deal
X organic w/ deal
X organic w/o deal
Y nonorganic w deal
Y nonorganic w/o deal
Y organic w deal
Y organic w/o deal

```
plotStore(6)
```


Profit for Store 6
Profit
2000
1500
1000
500
0
Week
0
10
20
30
40
50
profit_X_org_col
X nonorganic w deal
X nonorganic w/o deal
X organic w/ deal
X organic w/o deal
Y nonorganic w deal
Y nonorganic w/o deal
Y organic w deal
Y organic w/o deal

```
plotStore(7)
```

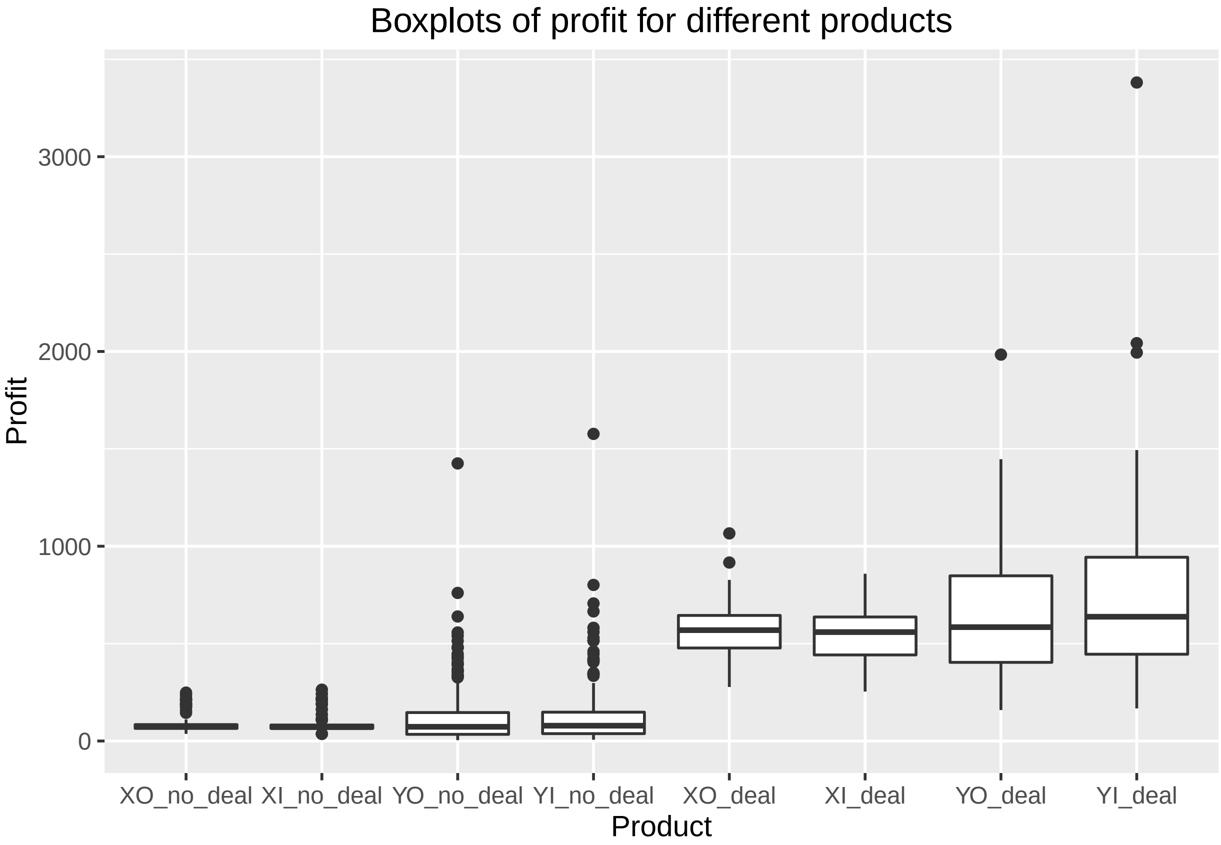


Box plots of profit for the different products

This is just a simple box plot analysis

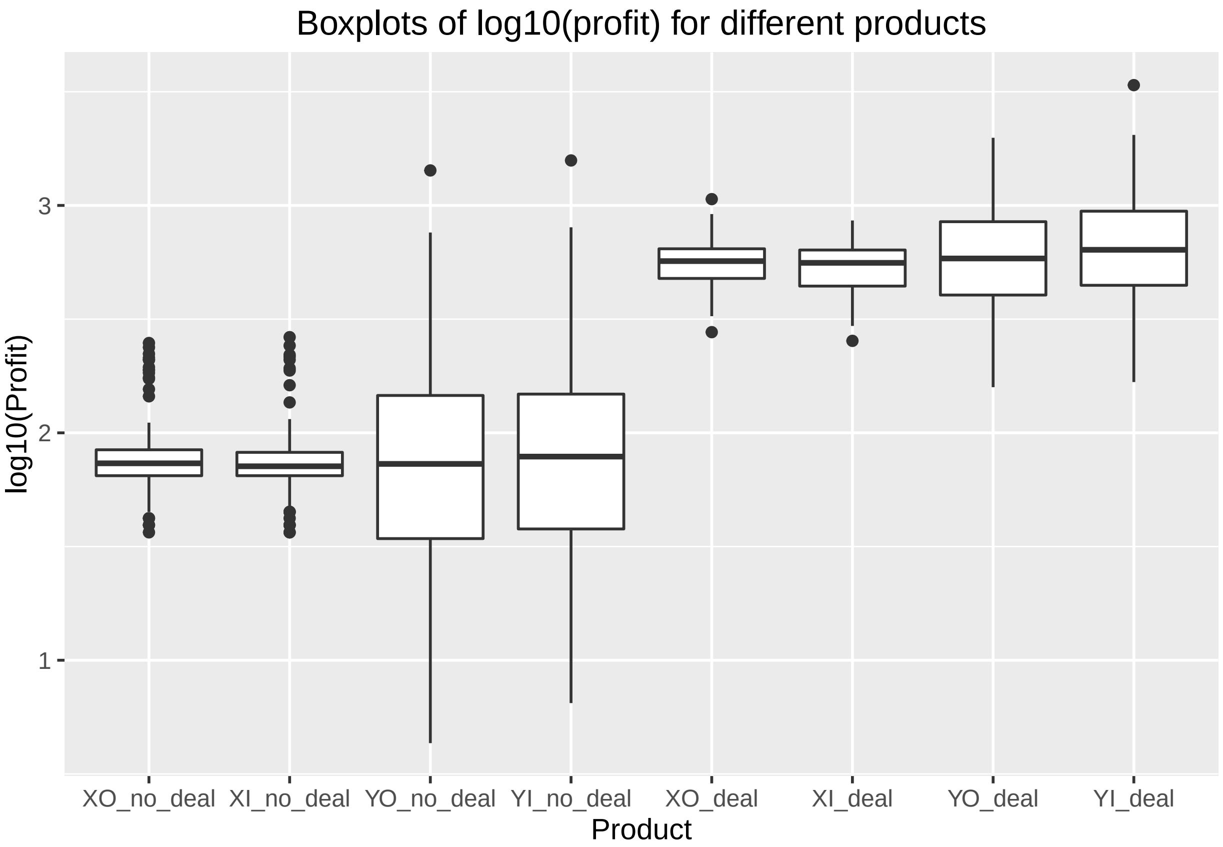
```
# Function to retreve data from dataframe and composite into factor
retrieveData <- function(data, deal, CLASS, xy) {
  names <- c("STORE", "PROFIT")
  if (xy == "x")
    temp <- data %>% filter(deal_X==deal, class==CLASS) %>% select(STORE, profit_X)
   else
    temp <- data %>% filter(deal_Y==deal, class==CLASS) %>% select(STORE, profit_Y)
  colnames(temp) <- names
  name <- if (xy == "x") "X" else "Y"
  name <- if (CLASS == "organic") paste(name,"O",sep="") else paste(name,"I",sep="")
  name <- if (deal == 1) paste(name,"deal",sep="_") else paste(name,"no_deal",sep="_")
  return(data.frame(type=rep(name,nrow(temp)),temp))
}

boxplot_data <- retrieveData(data, deal=0, CLASS="organic", xy="x")
boxplot_data <- rbind(boxplot_data,retrieveData(data, deal=0, CLASS="nonorganic", xy="x"))
boxplot_data <- rbind(boxplot_data,retrieveData(data, deal=0, CLASS="organic", xy="y"))
boxplot_data <- rbind(boxplot_data,retrieveData(data, deal=0, CLASS="nonorganic", xy="y"))
boxplot_data <- rbind(boxplot_data,retrieveData(data, deal=1, CLASS="organic", xy="x"))
boxplot_data <- rbind(boxplot_data,retrieveData(data, deal=1, CLASS="nonorganic", xy="x"))
boxplot_data <- rbind(boxplot_data,retrieveData(data, deal=1, CLASS="organic", xy="y"))
boxplot_data <- rbind(boxplot_data,retrieveData(data, deal=1, CLASS="nonorganic", xy="y"))
```

```
ggplot(boxplot_data, aes(x=type, y=PROFIT)) + geom_boxplot() + labs(title="Boxplots of profit for diffe
```



Given the massive spread between no deal and deal data, let's take a log10() scale transform of the y-axis

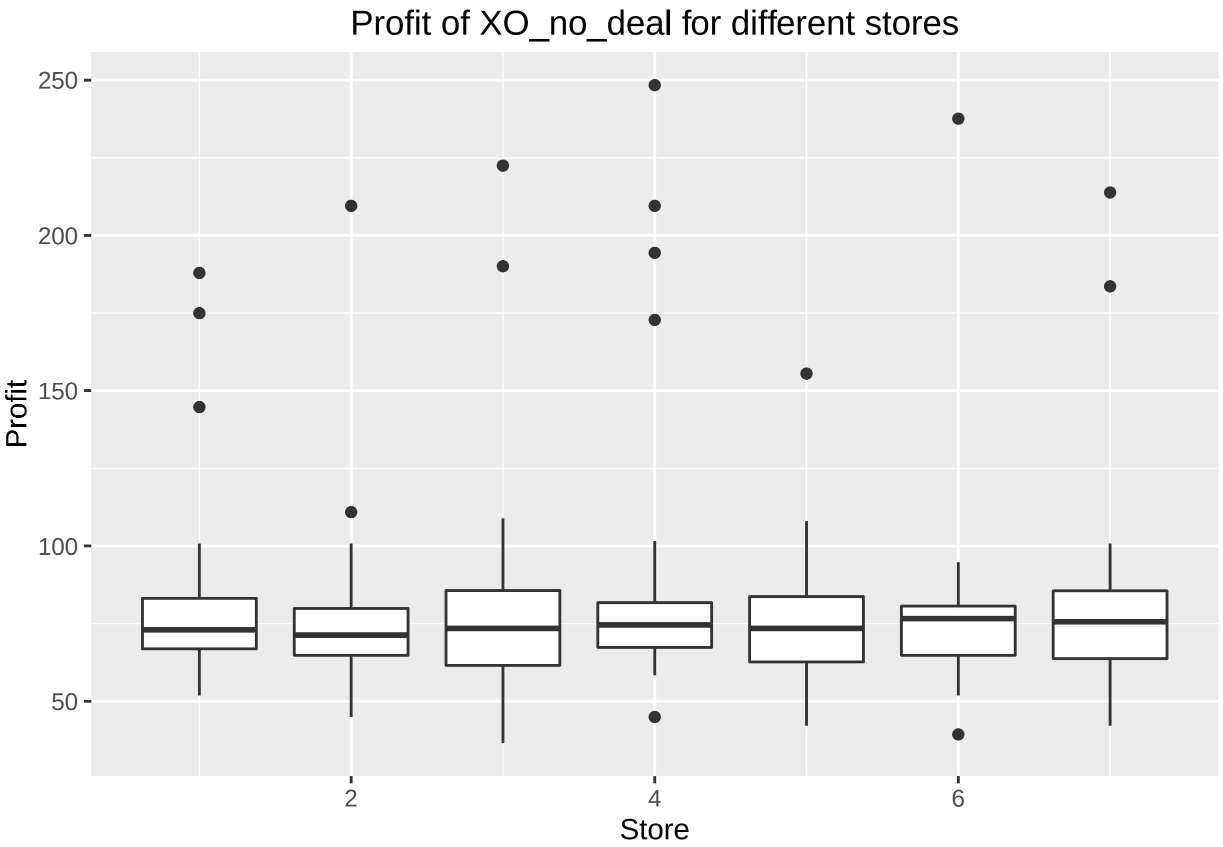
```
boxplot_data <- mutate(boxplot_data, log10PROFIT=log10(PROFIT))
ggplot(boxplot_data, aes(x=type, y=log10PROFIT)) + geom_boxplot() + labs(title="Boxplots of log10(profi
```



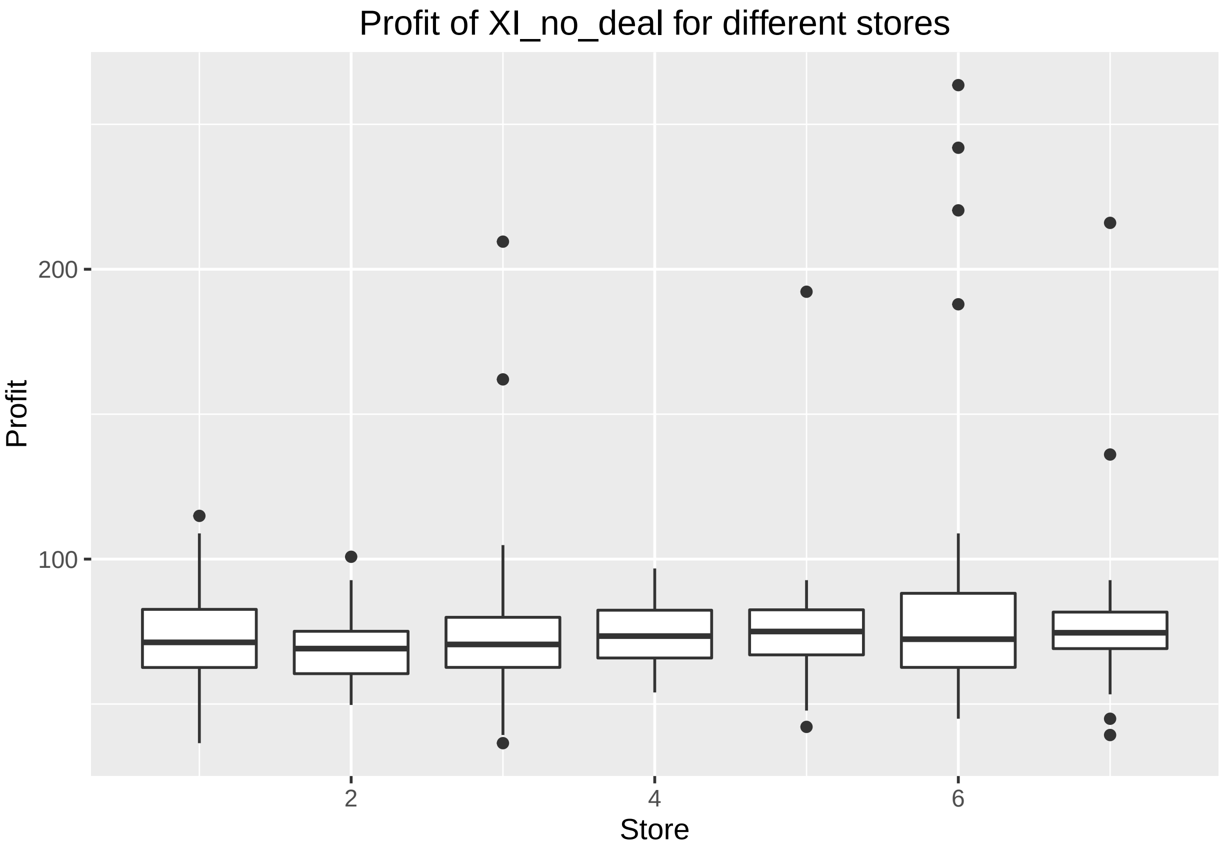
It is clear from these boxplots that the median values for X and Y products are approximately equal. The spread of profit for Y is much larger than X. Profit increases dramatically when a deal is going on.

Just as an additional spam of figures, lets look at the box plots for each product seperated by store (again with a log10 scaling)

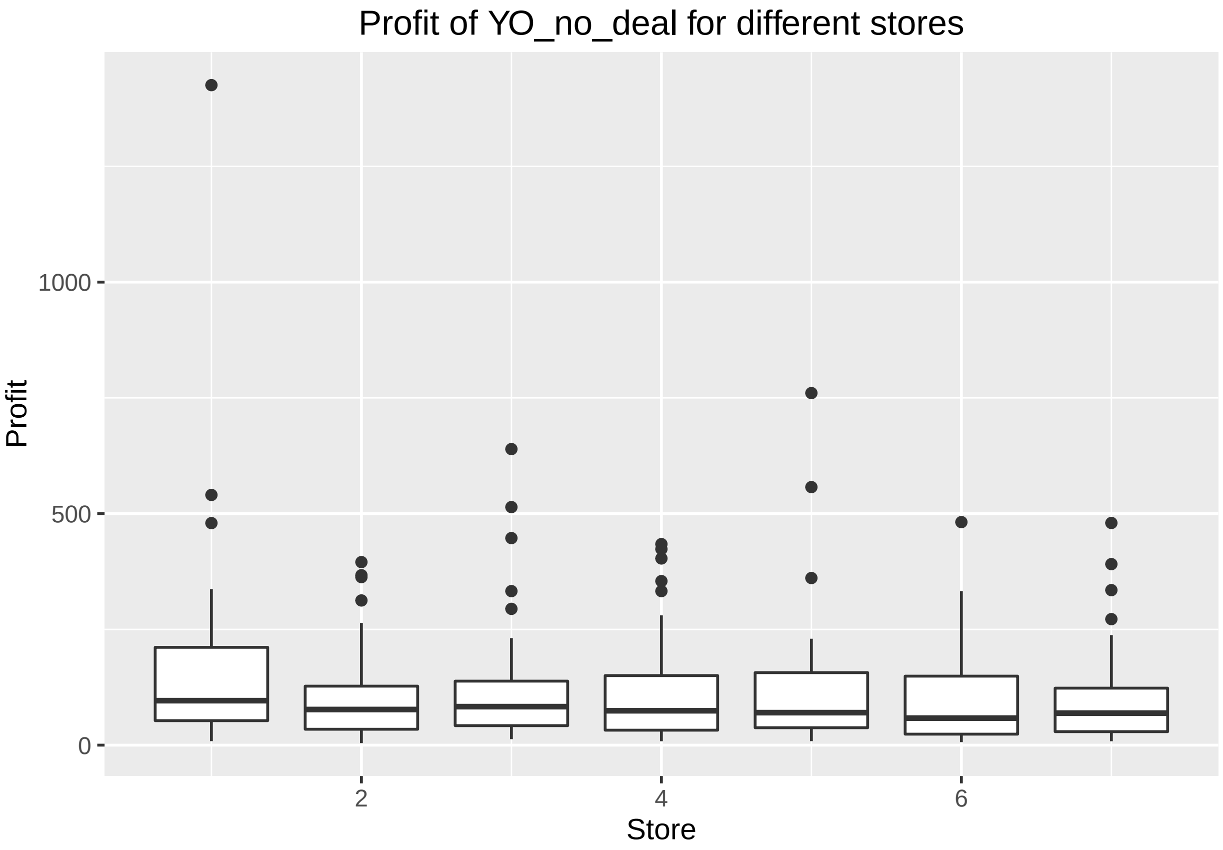
```
boxplot_data %>% filter(type=="XO_no_deal") %>% ggplot(., aes(x=STORE, y=PROFIT)) + geom_boxplot(aes(gr
```



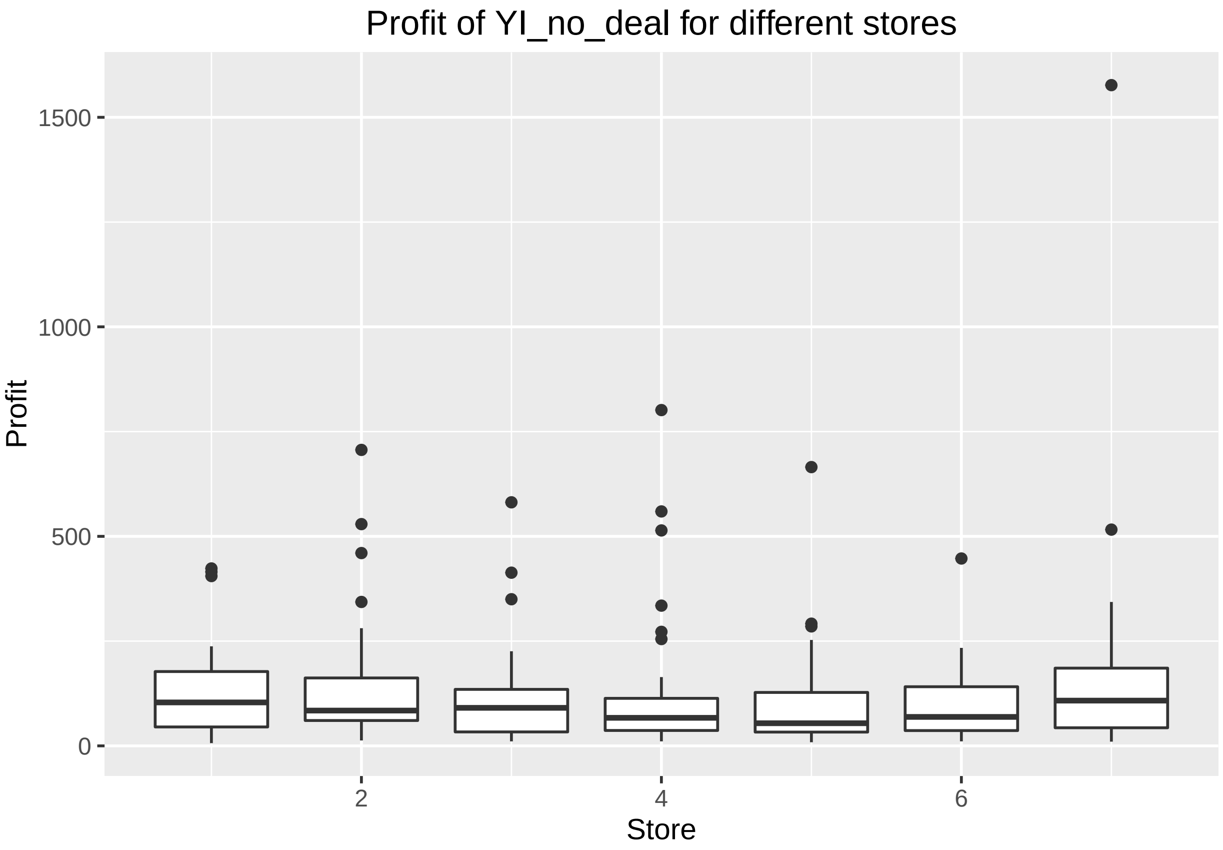
```
boxplot_data %>% filter(type=="XI_no_deal") %>% ggplot(., aes(x=STORE, y=PROFIT)) + geom_boxplot(aes(gr
```



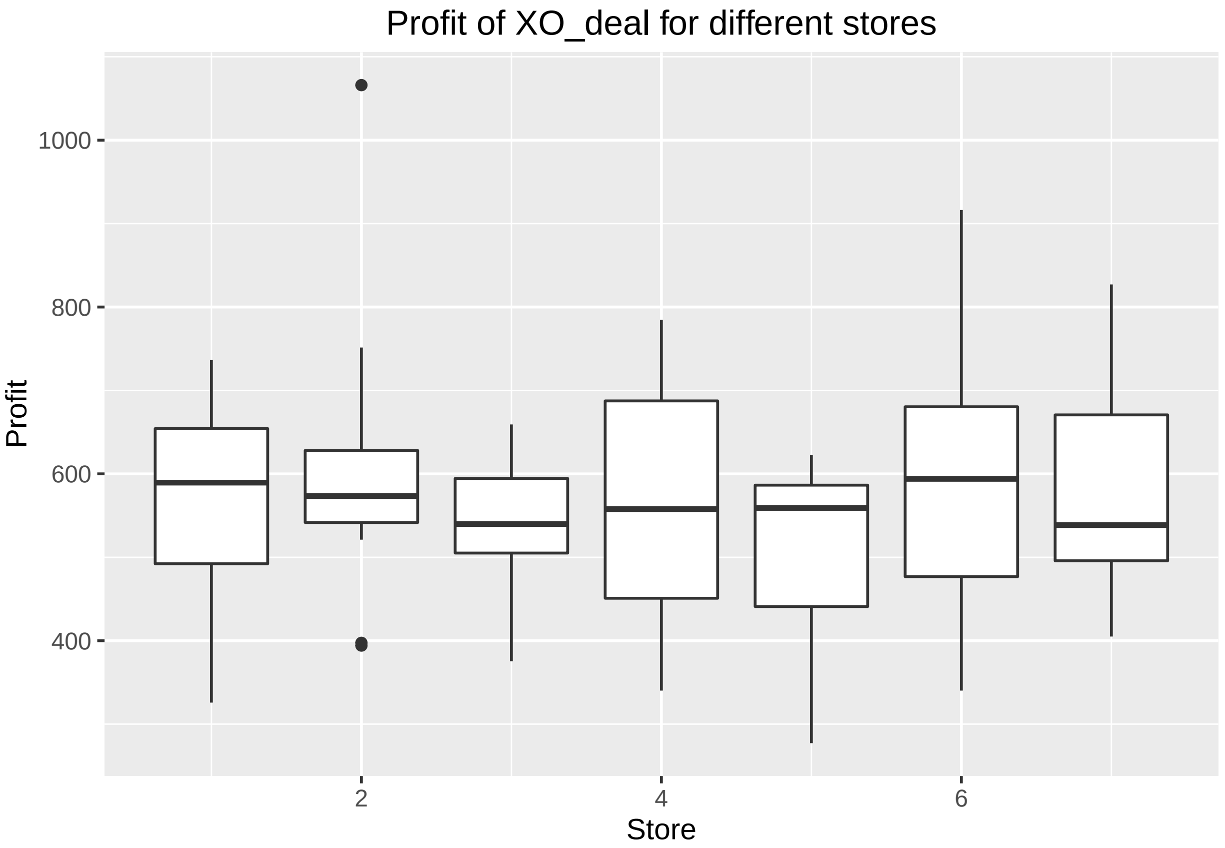
```
boxplot_data %>% filter(type=="YO_no_deal") %>% ggplot(., aes(x=STORE, y=PROFIT)) + geom_boxplot(aes(gr
```



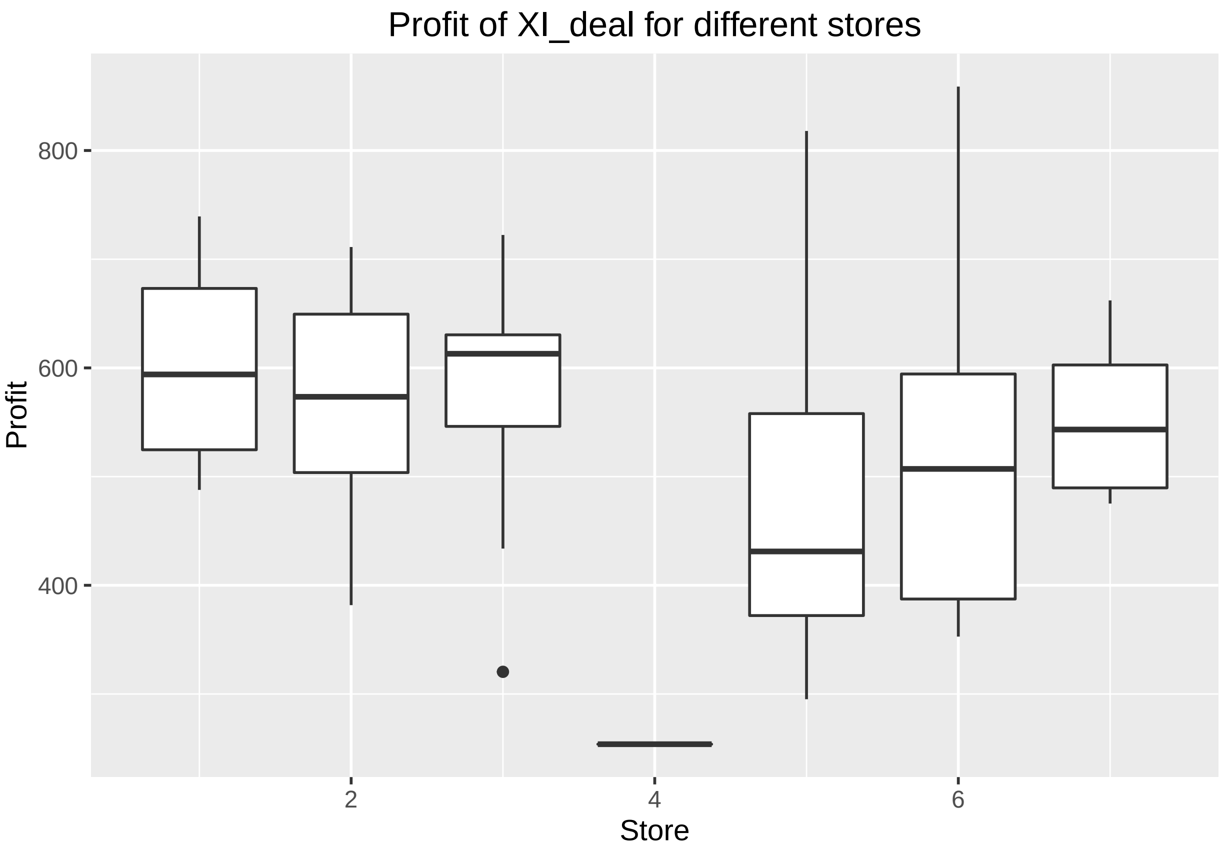
```
boxplot_data %>% filter(type=="YI_no_deal") %>% ggplot(., aes(x=STORE, y=PROFIT)) + geom_boxplot(aes(gr
```



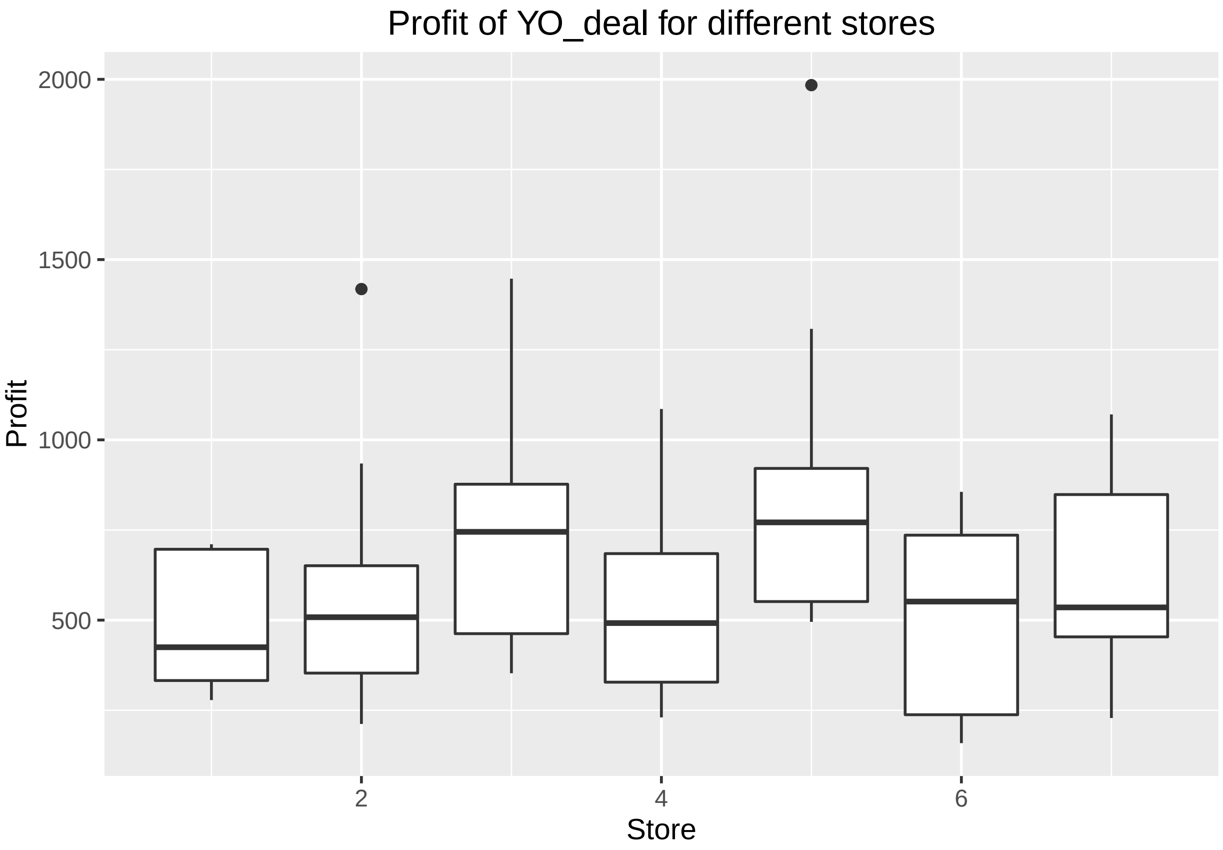
```
boxplot_data %>% filter(type=="XO_deal") %>% ggplot(., aes(x=STORE, y=PROFIT)) + geom_boxplot(aes(group
```



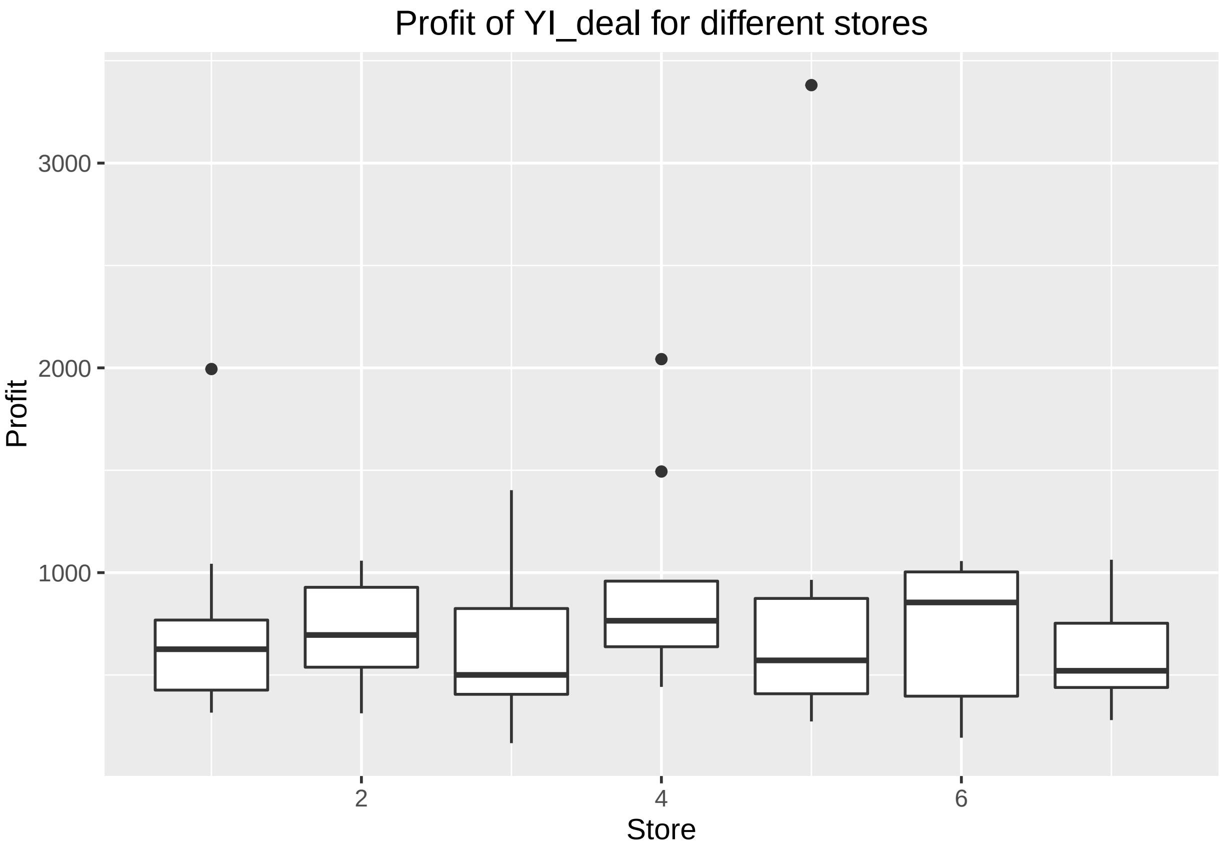
```
boxplot_data %>% filter(type=="XI_deal") %>% ggplot(., aes(x=STORE, y=PROFIT)) + geom_boxplot(aes(group
```



```
boxplot_data %>% filter(type=="YO_deal") %>% ggplot(., aes(x=STORE, y=PROFIT)) + geom_boxplot(aes(group
```



```
boxplot_data %>% filter(type=="YI_deal") %>% ggplot(., aes(x=STORE, y=PROFIT)) + geom_boxplot(aes(group
```



Regression

```r
# Function that pulls out the data
retrieveDataLM <- function(data, CLASS, xy) {
  names <- c("STORE", "oz", "p", "deal")
  if (xy == "x")
    temp <- data %>% filter(.,class==CLASS) %>% select(.,STORE, oz_X, pX, deal_X)
  else
    temp <- data %>% filter(class==CLASS) %>% select(STORE, oz_Y, pY, deal_Y)
  colnames(temp) <- names
  name <- if (xy == "x") "X" else "Y"
  name <- if (CLASS == "organic") paste(name,"O",sep="") else paste(name,"I",sep="")
  return(data.frame(type=rep(name,nrow(temp)),temp))
}

# Function that plots the loglog and actual curves
filteredLM <- function(data, TYPE, log=FALSE, include_Deal=FALSE) {
  response <- data %>% filter(type==TYPE) %>% select(oz) %>% {if (log) log(.) else (.)} %>% as.matrix()
  ind <- data %>% filter(type==TYPE) %>% select(p) %>% {if (log) log(.) else (.)} %>% as.matrix()
  deal <- data %>% filter(type==TYPE) %>% select(deal) %>% {if (include_Deal) (.) else (.)*0} %>% as.ma
  return(lm(response~ind+deal))
}

# Function that plots the loglog and actual curves
plotWithLM <- function(data, lm, TYPE, log=FALSE, include_Deal=FALSE) {
  response <- data %>% filter(type==TYPE) %>% select(oz) %>% {if (log) log(.) else (.)} %>% as.matrix()
  ind <- data %>% filter(type==TYPE) %>% select(p) %>% {if (log) log(.) else (.)} %>% as.matrix()
  deal <- data %>% filter(type==TYPE) %>% select(deal) %>% {if (include_Deal) (.) else (.)*0} %>% as.ma
  return(lm(response~ind+deal))
}

# Function that generates the inear model
lm_data <- retrieveDataLM(data, CLASS="organic", xy="x")
lm_data <- rbind(lm_data,retrieveDataLM(data, CLASS="nonorganic", xy="x"))
lm_data <- rbind(lm_data,retrieveDataLM(data, CLASS="organic", xy="y"))
lm_data <- rbind(lm_data,retrieveDataLM(data, CLASS="nonorganic", xy="y"))

# Create linear models
XO_lm_with_deal <- filteredLM(lm_data, TYPE="XO", log=TRUE, include_Deal=TRUE)
XO_lm_without_deal <- filteredLM(lm_data, TYPE="XO", log=TRUE, include_Deal=FALSE)
YO_lm_with_deal <- filteredLM(lm_data, TYPE="YO", log=TRUE, include_Deal=TRUE)
YO_lm_without_deal <- filteredLM(lm_data, TYPE="YO", log=TRUE, include_Deal=FALSE)
XI_lm_with_deal <- filteredLM(lm_data, TYPE="XI", log=TRUE, include_Deal=TRUE)
XI_lm_without_deal <- filteredLM(lm_data, TYPE="XI", log=TRUE, include_Deal=FALSE)
YI_lm_with_deal <- filteredLM(lm_data, TYPE="YI", log=TRUE, include_Deal=TRUE)
YI_lm_without_deal <- filteredLM(lm_data, TYPE="YI", log=TRUE, include_Deal=FALSE)
```

```
# Linear Model Summary for Organic X product without including the deal dummy variable
summary(XO_lm_without_deal)
```

```
##
## Call:
## lm(formula = response ~ ind + deal)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.65001 -0.23331 -0.06901  0.14514  1.17942
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -16.8432     0.5183  -32.49   <2e-16 ***
## ind          -7.0398     0.1445  -48.73   <2e-16 ***
## deal              NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3749 on 362 degrees of freedom
## Multiple R-squared:  0.8677, Adjusted R-squared:  0.8674
## F-statistic:  2375 on 1 and 362 DF,  p-value: < 2.2e-16
```

```
# Linear Model Summary for Organic X product including deal dummy variable
summary(XO_lm_with_deal)
```

```
##
## Call:
## lm(formula = response ~ ind + deal)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.48496 -0.12887 -0.03184  0.08066  1.21545
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.16811    0.71250  -4.446 1.16e-05 ***
## ind         -3.12964    0.20256 -15.450  < 2e-16 ***
## deal         1.39566    0.06387  21.851  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2463 on 361 degrees of freedom
## Multiple R-squared:  0.9431, Adjusted R-squared:  0.9427
## F-statistic:  2989 on 2 and 361 DF,  p-value: < 2.2e-16
```

```
# Linear Model Summary for Organic Y product without including the deal dummy variable
summary(YO_lm_without_deal)
```

```
##
## Call:
## lm(formula = response ~ ind + deal)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.85175 -0.65452  0.03862  0.62843  2.76432
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -15.7803     1.2829  -12.30   <2e-16 ***
## ind          -6.7308     0.3595  -18.72   <2e-16 ***
## deal              NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9754 on 362 degrees of freedom
## Multiple R-squared:  0.4919, Adjusted R-squared:  0.4905
## F-statistic: 350.5 on 1 and 362 DF,  p-value: < 2.2e-16
```

```
# Linear Model Summary for Organic Y product including deal dummy variable
summary(YO_lm_with_deal)
```

```
##
## Call:
## lm(formula = response ~ ind + deal)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.79111 -0.65105  0.03551  0.60161  2.80730
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -5.9350     2.3642  -2.510   0.0125 *
## ind          -3.9058     0.6740  -5.795 1.49e-08 ***
## deal          1.1975     0.2445   4.897 1.47e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9458 on 361 degrees of freedom
## Multiple R-squared:  0.5236, Adjusted R-squared:  0.5209 
## F-statistic: 198.4 on 2 and 361 DF,  p-value: < 2.2e-16
```

```
# Linear Model Summary for Nonorganic X product without including the deal dummy variable
summary(XI_lm_without_deal)
```

```
## 
## Call:
## lm(formula = response ~ ind + deal)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.62446 -0.23014 -0.09199  0.08494  1.27804 
## 
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -15.8704     0.5801  -27.36   <2e-16 ***
## ind          -6.7511     0.1630  -41.43   <2e-16 ***
## deal              NA         NA      NA       NA    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.3569 on 362 degrees of freedom
## Multiple R-squared:  0.8258, Adjusted R-squared:  0.8253 
## F-statistic:  1716 on 1 and 362 DF,  p-value: < 2.2e-16
```

```
# Linear Model Summary for Nonorganic X product including deal dummy variable
summary(XI_lm_with_deal)
```

```
## 
## Call:
## lm(formula = response ~ ind + deal)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.52573 -0.10589 -0.00324  0.06946  1.30459 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -3.27234    0.60919  -5.372  1.4e-07 ***
## ind         -3.15079    0.17312 -18.200  < 2e-16 ***
## deal         1.40060    0.05548  25.243  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.2149 on 361 degrees of freedom
## Multiple R-squared:  0.937,  Adjusted R-squared:  0.9367 
## F-statistic:  2685 on 2 and 361 DF,  p-value: < 2.2e-16
```

```
# Linear Model Summary for Nonorganic Y product without including the deal dummy variable
summary(YI_lm_without_deal)
```

```
## 
## Call:
## lm(formula = response ~ ind + deal)
## 
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5546 -0.6087  0.0357  0.6109  3.4261
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -15.4133     1.2595  -12.24   <2e-16 ***
## ind          -6.6570     0.3529  -18.86   <2e-16 ***
## deal              NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9586 on 362 degrees of freedom
## Multiple R-squared:  0.4957, Adjusted R-squared:  0.4943
## F-statistic: 355.8 on 1 and 362 DF,  p-value: < 2.2e-16
```

```
# Linear Model Summary for Nonorganic Y product including deal dummy variable
summary(YI_lm_with_deal)
```

```
##
## Call:
## lm(formula = response ~ ind + deal)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.46784 -0.56287  0.02177  0.56931  3.02660
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.3106     2.3526  -0.982    0.327
## ind          -2.8956     0.6712  -4.314 2.07e-05 ***
## deal          1.5425     0.2386   6.464 3.31e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9087 on 361 degrees of freedom
## Multiple R-squared:  0.548,  Adjusted R-squared:  0.5455
## F-statistic: 218.8 on 2 and 361 DF,  p-value: < 2.2e-16
```