ISOM5610 HW2

Team 1

22 November 2018

```
setwd("~/MSBA/ISOM5610/HW2")
library(readxl)
bluestem <- as.data.frame(read_excel("bluestem.xls", sheet=1))
bluestem[,2:9] <- lapply(bluestem[,2:9], factor)
colnames(bluestem)[1] <- "Sales"
colnames(bluestem)[10] <- "Index"
summary(bluestem)
```

```
##      Sales        Promotion Monday  Tuesday Wednesday Thursday Friday
##  Min.   :  30.0   0:244     0:210   0:208   0:209     0:208    0:208
##  1st Qu.: 456.5   1:  5     1: 39   1: 41   1: 40     1: 41    1: 41
##  Median : 715.6
##  Mean   : 944.3
##  3rd Qu.:1360.5
##  Max.   :2994.3
##  Saturday Sunday      Index
##  0:207    0:244   Min.   :0.920
##  1: 42    1:  5   1st Qu.:1.000
##                   Median :1.180
##                   Mean   :1.664
##                   3rd Qu.:2.530
##                   Max.   :2.870
```

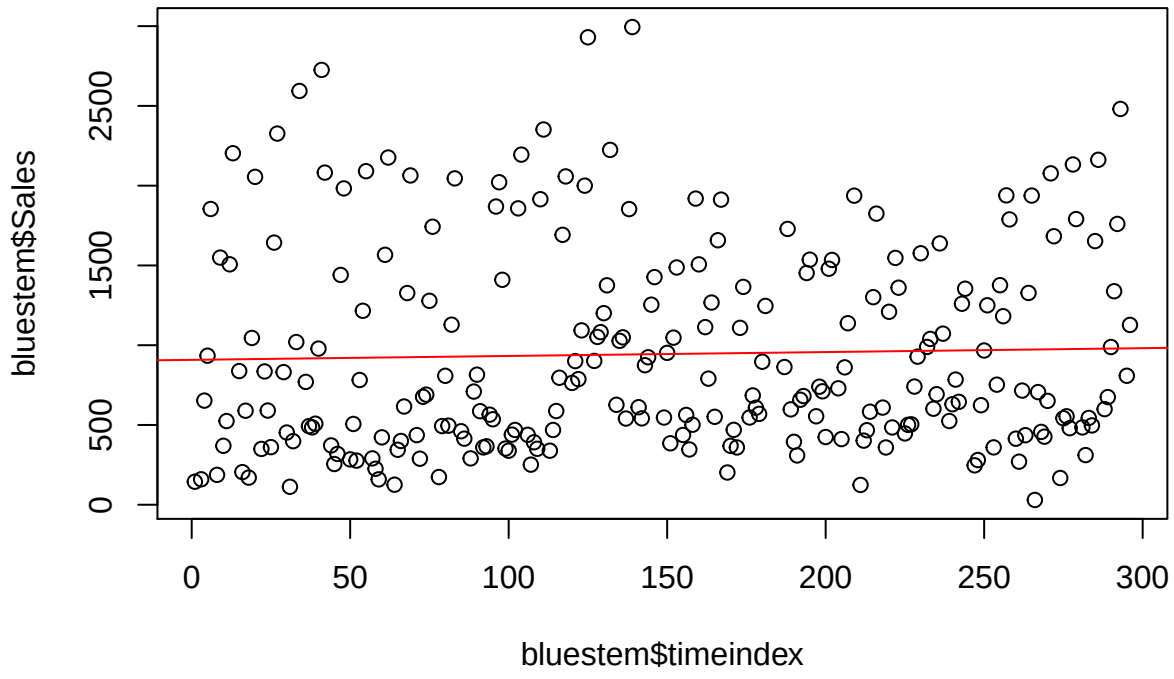
Data Exploration

```
#Adding the time index predictor

bluestem$Weekday <- factor(names(bluestem[3:9])[max.col(bluestem[3:9])], levels = c("Sunday", "Monday",
stan <- rep(c('Monday','Tuesday','Wednesday','Thursday','Friday','Saturday','Sunday'),50)
st <- 1
timeindex <- c()
for (lp in 1:249){
  timeindex[lp] <- st
  while (stan[st] != bluestem$Weekday[lp]){
    st <- st+1
    timeindex[lp] <- st
  }
}
bluestem$timeindex <- timeindex

#data processing

bluestem$desSales <- bluestem$Sales/bluestem$Index

#See if there is linear trend along the time line
```

```r
fit0 <- lm(Sales ~ timeindex,data=bluestem)
summary(fit0)
```

```
##
## Call:
## lm(formula = Sales ~ timeindex, data = bluestem)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -943.0 -493.1 -228.1  402.2 2052.2
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 908.1840    80.2032   11.32   <2e-16 ***
## timeindex     0.2438     0.4687    0.52    0.604
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 635.2 on 247 degrees of freedom
## Multiple R-squared:  0.001094,   Adjusted R-squared:  -0.002951
## F-statistic: 0.2704 on 1 and 247 DF,  p-value: 0.6035
```

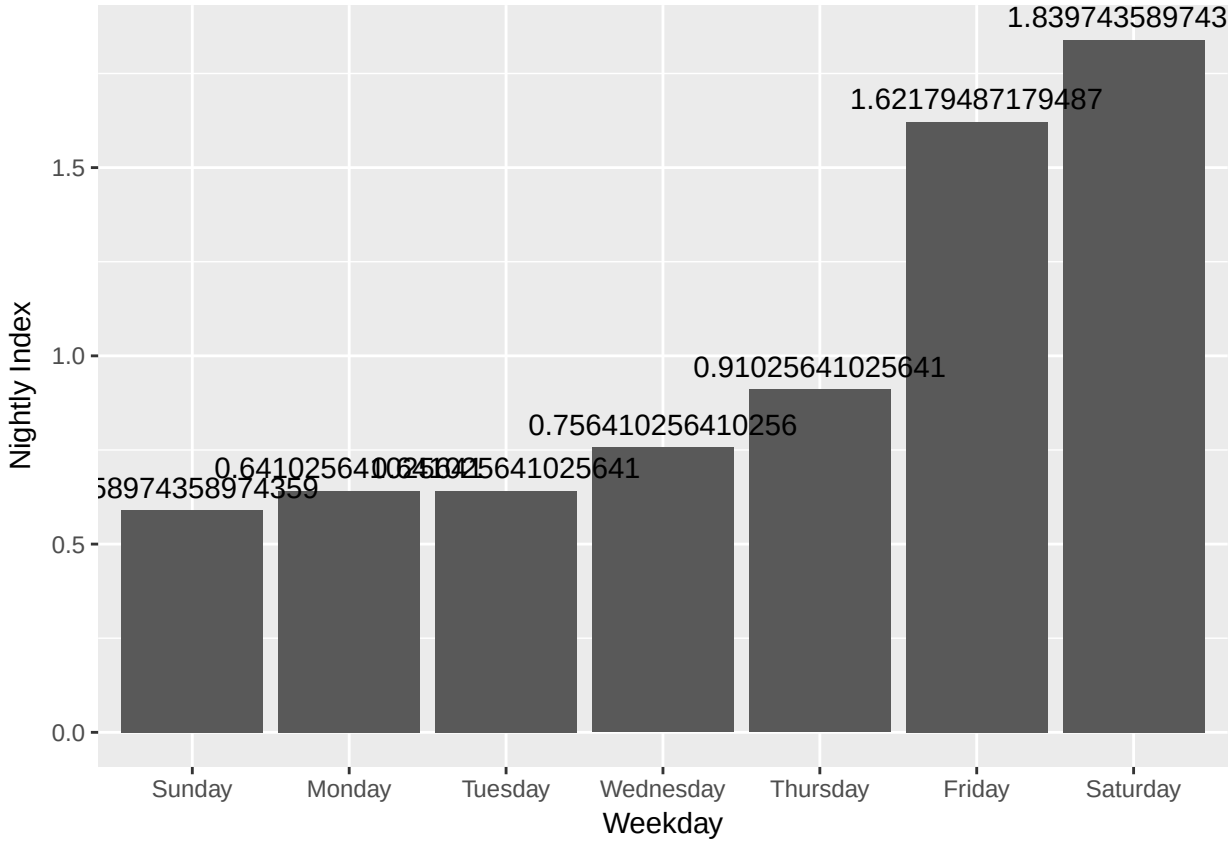
```r
plot(bluestem$timeindex,bluestem$Sales)
abline(fit0,col=2)
```



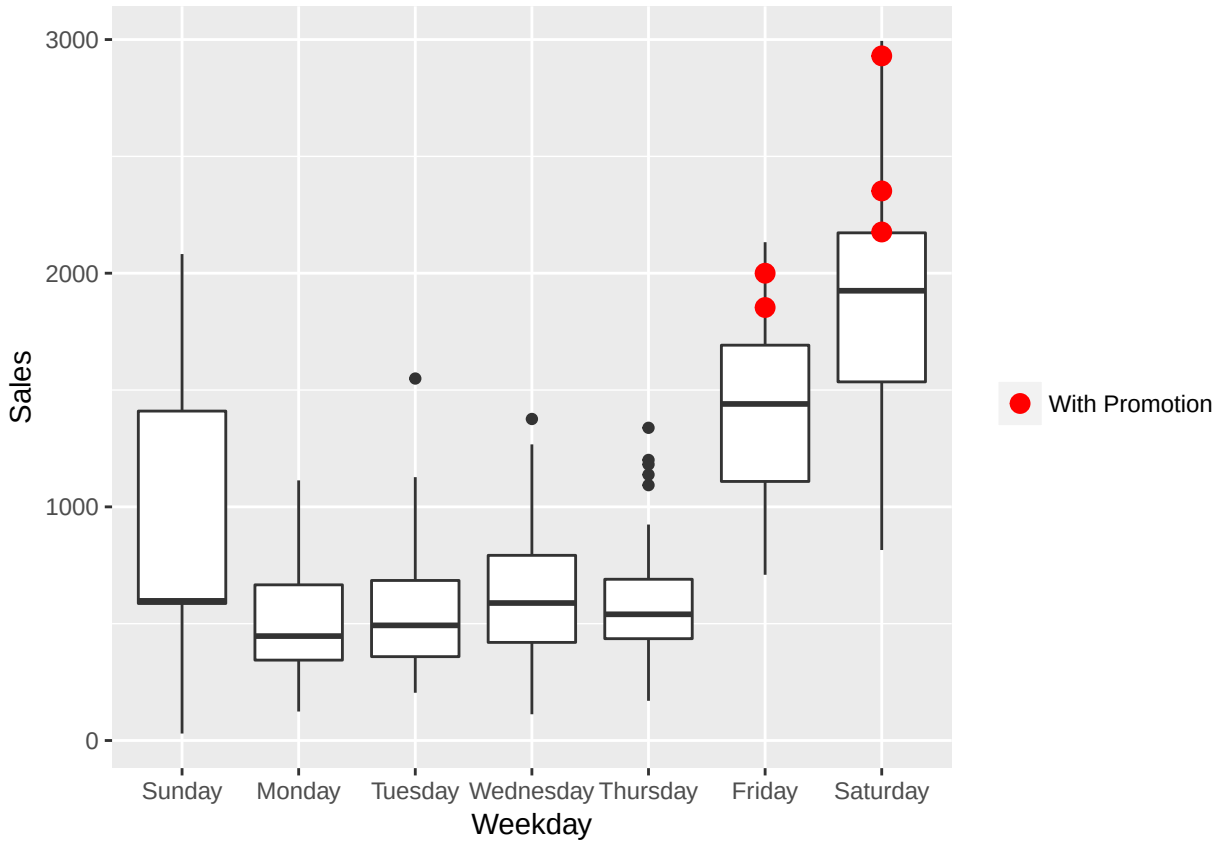

```r
## standardize index

index_table <- unique(bluestem[names(bluestem) %in% c("Weekday", "Index")])
bluestem$Index <- bluestem$Index*7/sum(index_table[,1])
```

```r
#Different Nightly Index for different Weekdays
index_table <- unique(bluestem[names(bluestem) %in% c("Weekday", "Index")])
```

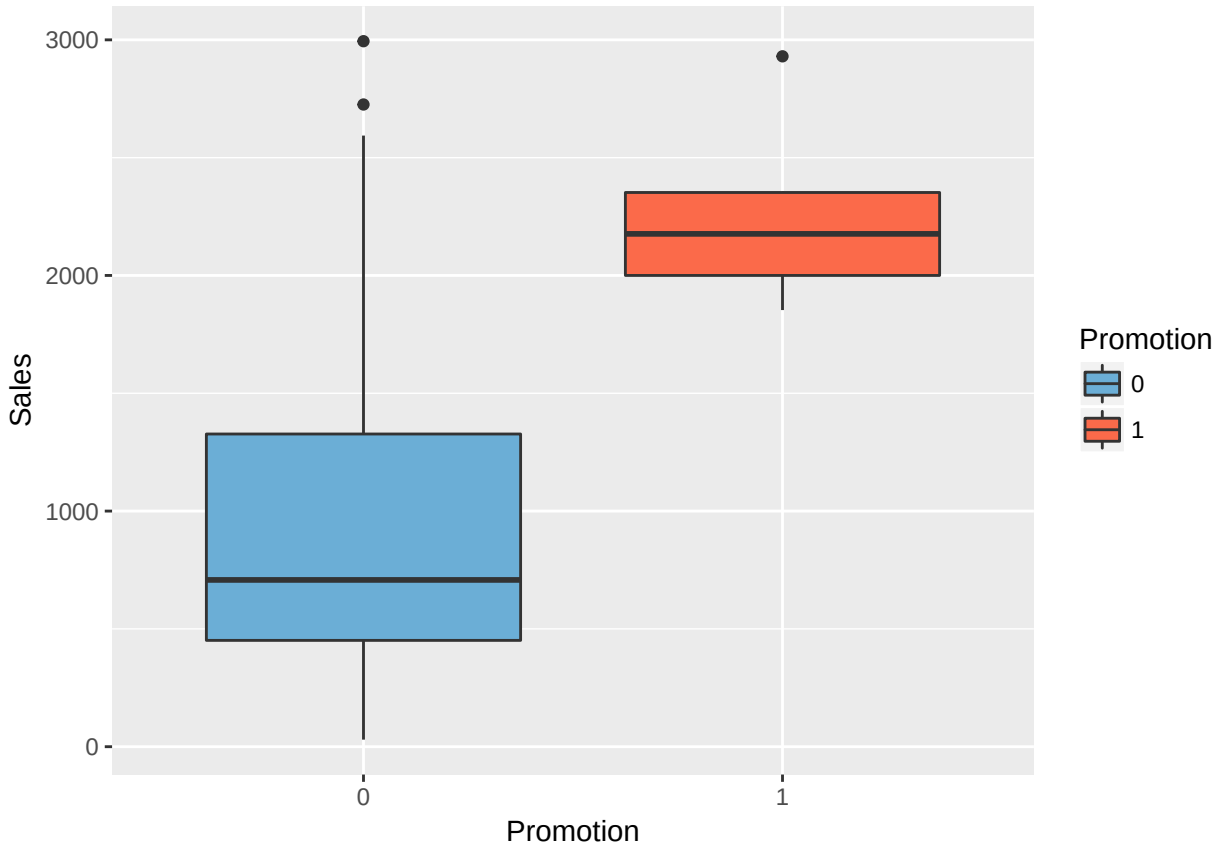
```
library(ggplot2)
library(RColorBrewer)

ggplot(index_table, aes(x = Weekday, y = Index)) +
    geom_bar(stat = "identity") +
    geom_text(aes(label = Index), vjust=-0.6) +
    labs(y="Nightly Index")
```



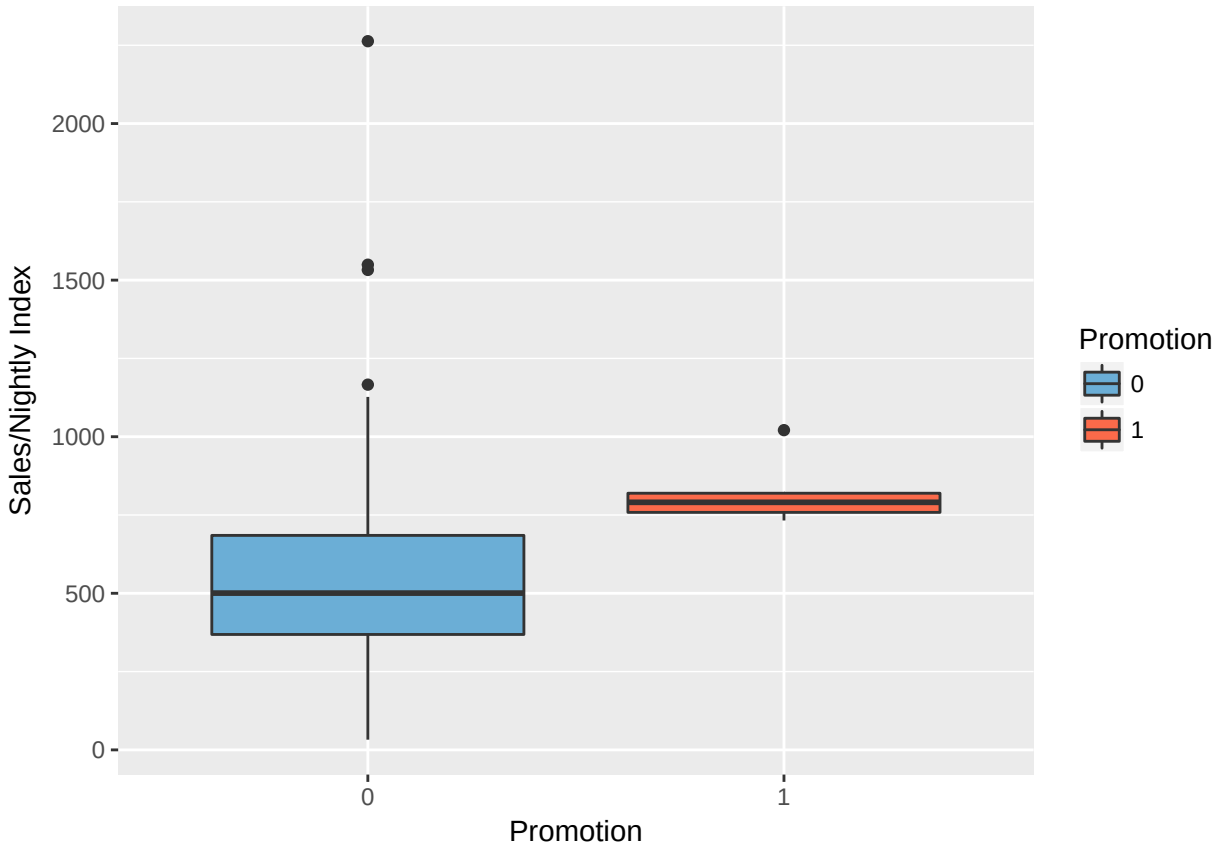

```
#Sales on different Weekdays
ggplot(bluestem, aes(x = Weekday, y = Sales)) +
    geom_boxplot() +
    geom_point(data=subset(bluestem, Promotion==1), mapping=aes(color=Promotion), size=3) +
    labs(color="") +
    scale_color_manual(labels = "With Promotion", values="red")
```




```
#Sales with or without promotions
ggplot(bluestem, aes(x = Promotion, y = Sales, fill= Promotion)) +
    geom_boxplot()+
    scale_fill_manual(values=c(brewer.pal(7, "Blues")[4],brewer.pal(7, "Reds")[4]))
```




```
#De-seasonalized Sales with or without promotions
ggplot(bluestem, aes(x = Promotion, y = desSales, fill= Promotion)) +
    geom_boxplot() +
    scale_fill_manual(values=c(brewer.pal(7, "Blues")[4],brewer.pal(7, "Reds")[4])) +
    labs(y="Sales/Nightly Index")
```



Additive Model

A regression model using dummy variables to account for the fixed Weekday??s effect.

```
## in this chunk, ".t" means timeindex is added

bluestem_1 <- bluestem[1:8]
bluestem_1.t <- bluestem_1
bluestem_1.t $ timeindex <- bluestem$timeindex
str(bluestem_1) #Sunday as the base dummy
```

```
## 'data.frame':    249 obs. of  8 variables:
##  $ Sales    : num  144 159 653 934 1854 ...
##  $ Promotion: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Monday   : Factor w/ 2 levels "0","1": 2 1 1 1 1 2 1 1 1 1 ...
##  $ Tuesday  : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 2 1 1 1 ...
##  $ Wednesday: Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 2 1 1 ...
##  $ Thursday : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 1 1 2 1 ...
##  $ Friday   : Factor w/ 2 levels "0","1": 1 1 1 2 1 1 1 1 1 2 ...
##  $ Saturday : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 1 1 1 ...
```

```
add.fit <- lm(formula = 'Sales ~ .', data = bluestem_1)
add.fit.t <- lm(formula = 'Sales ~ .', data = bluestem_1.t)
summary(add.fit)
```

```
##
## Call:
```

```
## lm(formula = "Sales ~ .", data = bluestem_1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1033.06  -215.68   -57.86   197.35  1145.85
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    941.2      153.6   6.126 3.64e-09 ***
## Promotion1     593.5      158.7   3.741 0.000229 ***
## Monday1       -441.3      163.2  -2.704 0.007331 **
## Tuesday1      -383.9      162.7  -2.359 0.019138 *
## Wednesday1    -316.6      163.0  -1.943 0.053203 .
## Thursday1     -329.0      162.7  -2.021 0.044347 *
## Friday1        458.3      162.9   2.813 0.005314 **
## Saturday1      907.2      162.9   5.568 6.85e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 343.5 on 241 degrees of freedom
## Multiple R-squared:  0.7149, Adjusted R-squared:  0.7067
## F-statistic: 86.35 on 7 and 241 DF,  p-value: < 2.2e-16
```

```
summary(add.fit.t)
```

```
##
## Call:
## lm(formula = "Sales ~ .", data = bluestem_1.t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1015.01  -210.15   -49.03   202.97  1168.33
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  901.4443   157.4466   5.725 3.07e-08 ***
## Promotion1   604.8240   158.8762   3.807 0.000179 ***
## Monday1     -444.2539   163.1095  -2.724 0.006931 **
## Tuesday1    -388.0640   162.6786  -2.385 0.017834 *
## Wednesday1  -319.4482   162.8767  -1.961 0.051002 .
## Thursday1   -331.6688   162.6541  -2.039 0.042535 *
## Friday1      454.8801   162.8479   2.793 0.005639 **
## Saturday1    902.8264   162.8656   5.543 7.80e-08 ***
## timeindex      0.2898     0.2539   1.141 0.254845
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 343.3 on 240 degrees of freedom
## Multiple R-squared:  0.7165, Adjusted R-squared:  0.707
## F-statistic: 75.81 on 8 and 240 DF,  p-value: < 2.2e-16
```

```
add.fit$coefficients[2]
```

```
## Promotion1
##   593.5352
```

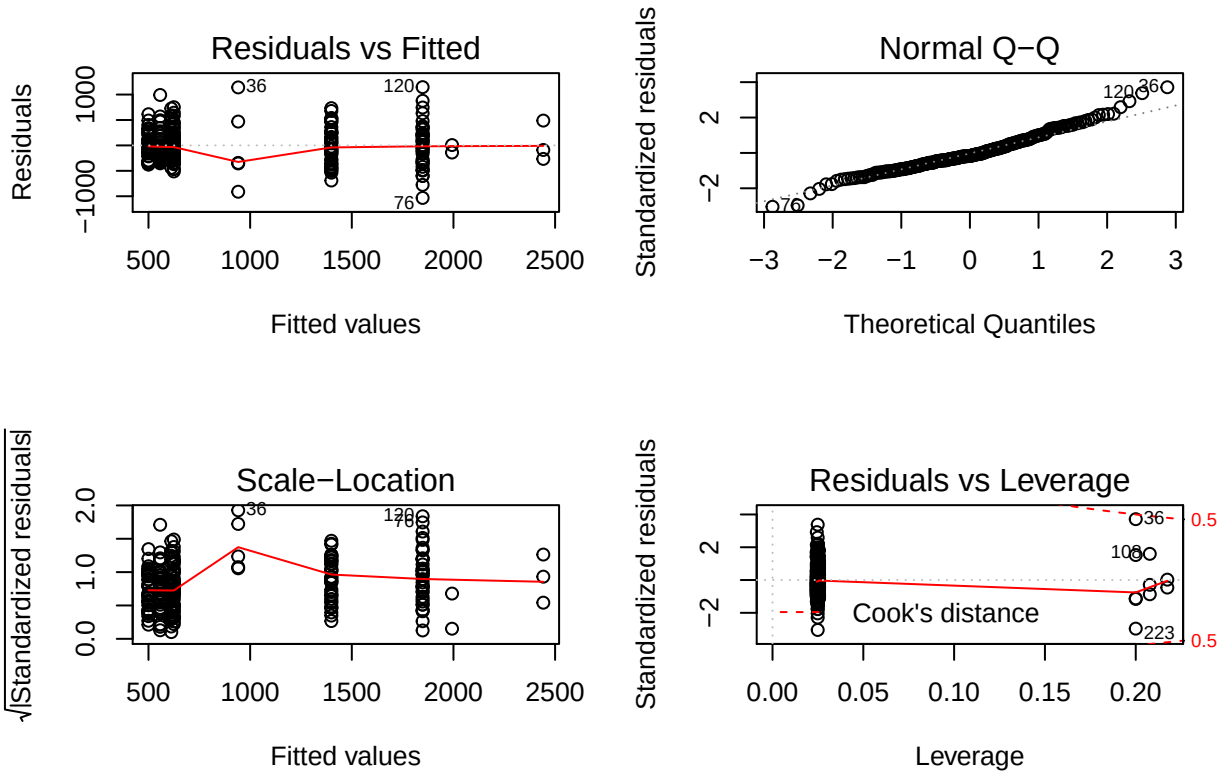
```
add.fit.t$coefficients[2]
```

```
## Promotion1
##    604.824
```

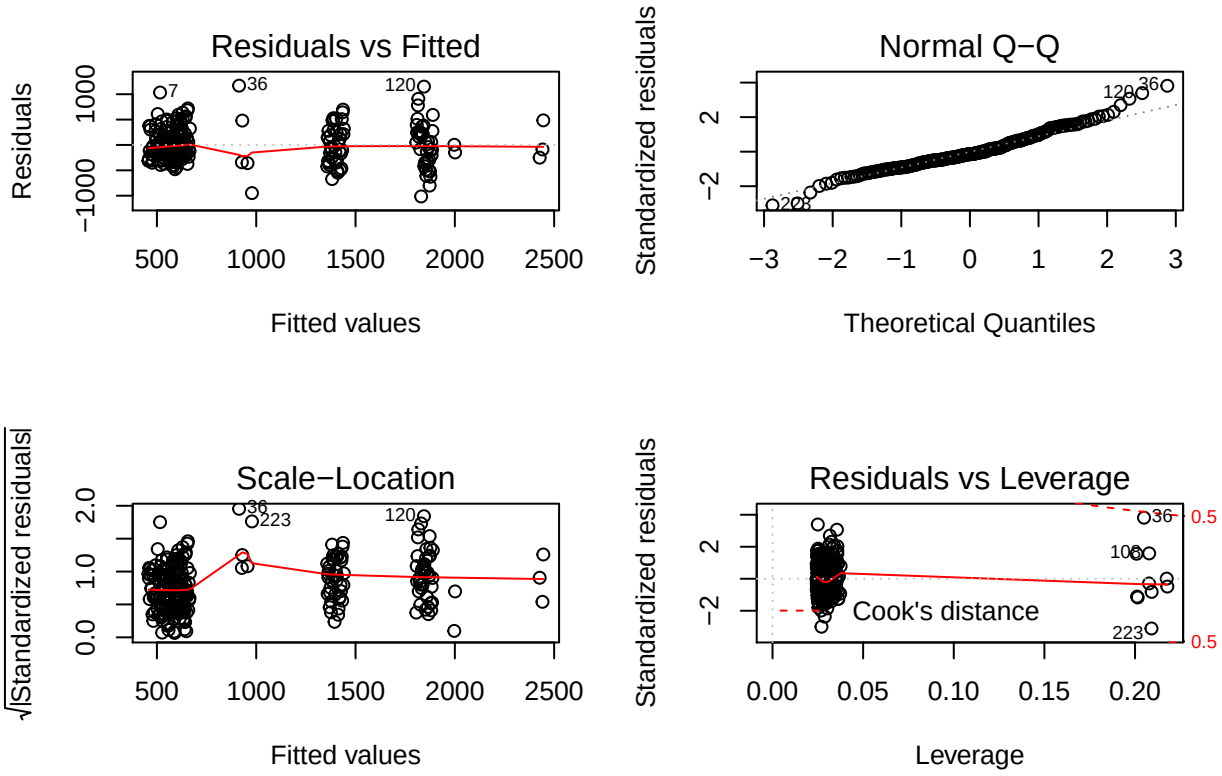
Answer = 593.5

** After the time index is added, Answer = 604.73 **

```
# Checking model assumptions
par(mfrow=c(2,2))
plot(add.fit)
```




```
plot(add.fit.t)
```




```
# Checking multicollinearity for independent variables
library(HH)
```

```
## Warning: package 'HH' was built under R version 3.4.4
```

```
## Loading required package: lattice
```

```
## Loading required package: grid
```

```
## Loading required package: latticeExtra
```

```
##
## Attaching package: 'latticeExtra'
```

```
## The following object is masked from 'package:ggplot2':
##
##     layer
```

```
## Loading required package: multcomp
```

```
## Loading required package: mvtnorm
```

```
## Warning: package 'mvtnorm' was built under R version 3.4.4
```

```
## Loading required package: survival
```

```
## Loading required package: TH.data
```

```
## Warning: package 'TH.data' was built under R version 3.4.4
```

```
## Loading required package: MASS
```

```
## Warning: package 'MASS' was built under R version 3.4.4
```

```
##
## Attaching package: 'TH.data'
```

```
## The following object is masked from 'package:MASS':
##
##     geyser
```

```
## Loading required package: gridExtra
```

```
vif(add.fit)
```

```
## Promotion1    Monday1   Tuesday1 Wednesday1  Thursday1    Friday1
##   1.045102   7.421687   7.685141   7.554217   7.685141   7.702524
##  Saturday1
##   7.852456
```

```
vif(add.fit.t)
```

```
## Promotion1    Monday1   Tuesday1 Wednesday1  Thursday1    Friday1
##   1.049168   7.423516   7.689097   7.555990   7.686782   7.705110
##  Saturday1  timeindex
##   7.856801   1.004672
```

Multiplicative Model

The nightly index expresses each Weekday??s effect on the popularity.

1. Use De-seasonalized Sales.

```
mtp.fit=lm(formula = 'desSales ~ Promotion', data = bluestem)
summary(mtp.fit)
```

```
##
## Call:
## lm(formula = "desSales ~ Promotion", data = bluestem)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -512.52 -175.82  -44.22  139.75 1717.86
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   545.12      16.76  32.527   <2e-16 ***
## Promotion1    279.25     118.27   2.361    0.019 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 261.8 on 247 degrees of freedom
## Multiple R-squared:  0.02207,    Adjusted R-squared:  0.01811
## F-statistic: 5.575 on 1 and 247 DF,  p-value: 0.019
```

```
mtp.fit.t=lm(formula = 'desSales ~ Promotion + timeindex', data = bluestem)
summary(mtp.fit)
```

```
##
## Call:
## lm(formula = "desSales ~ Promotion", data = bluestem)
##
## Residuals:
```

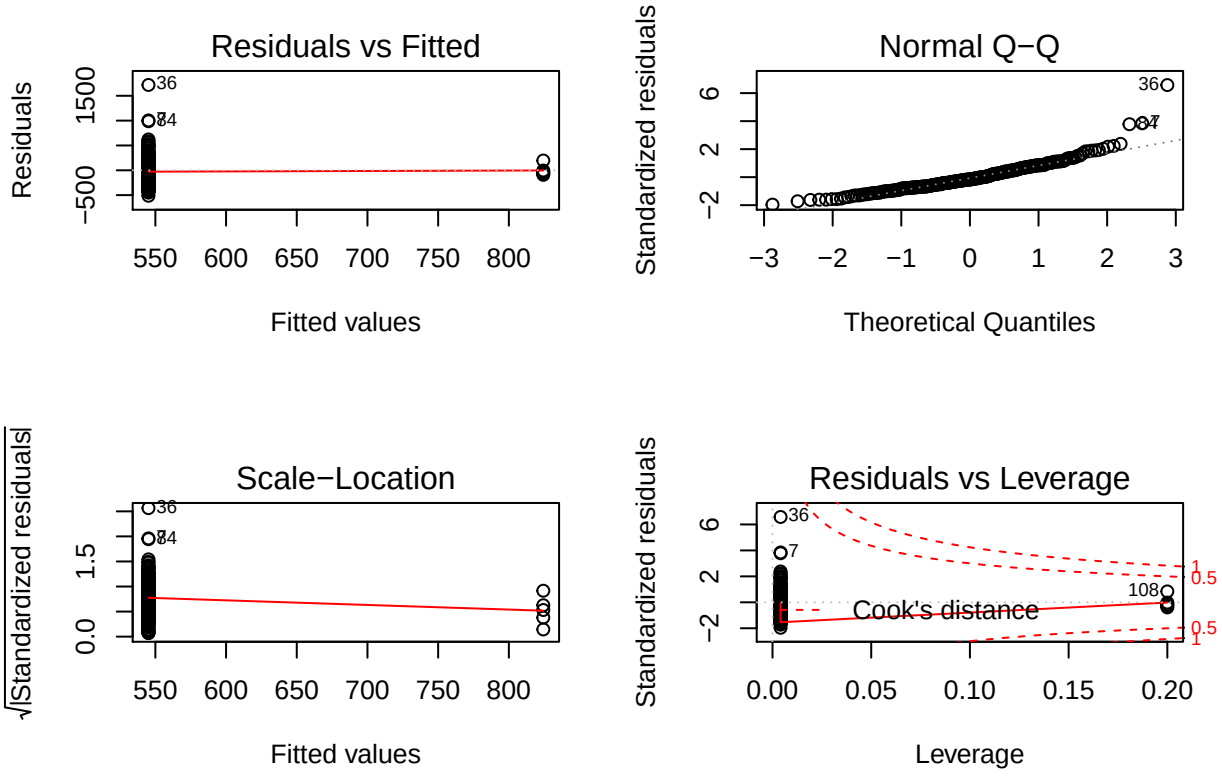
```
##     Min      1Q  Median      3Q     Max
## -512.52 -175.82  -44.22  139.75 1717.86
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   545.12      16.76  32.527   <2e-16 ***
## Promotion1    279.25     118.27   2.361    0.019 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 261.8 on 247 degrees of freedom
## Multiple R-squared:  0.02207,    Adjusted R-squared:  0.01811
## F-statistic: 5.575 on 1 and 247 DF,  p-value: 0.019
```

```
# The coefficient of Promotion stands for the boost in revenues without Weekday effect. Such boost woul
mtp.fit$coefficients[2] * index_table$Index[which(index_table$Weekday=="Saturday")]
```

```
## Promotion1
##   513.7401
```

Answer = 801.4345

```
# Checking model assumptions
par(mfrow=c(2,2))
plot(mtp.fit)
```

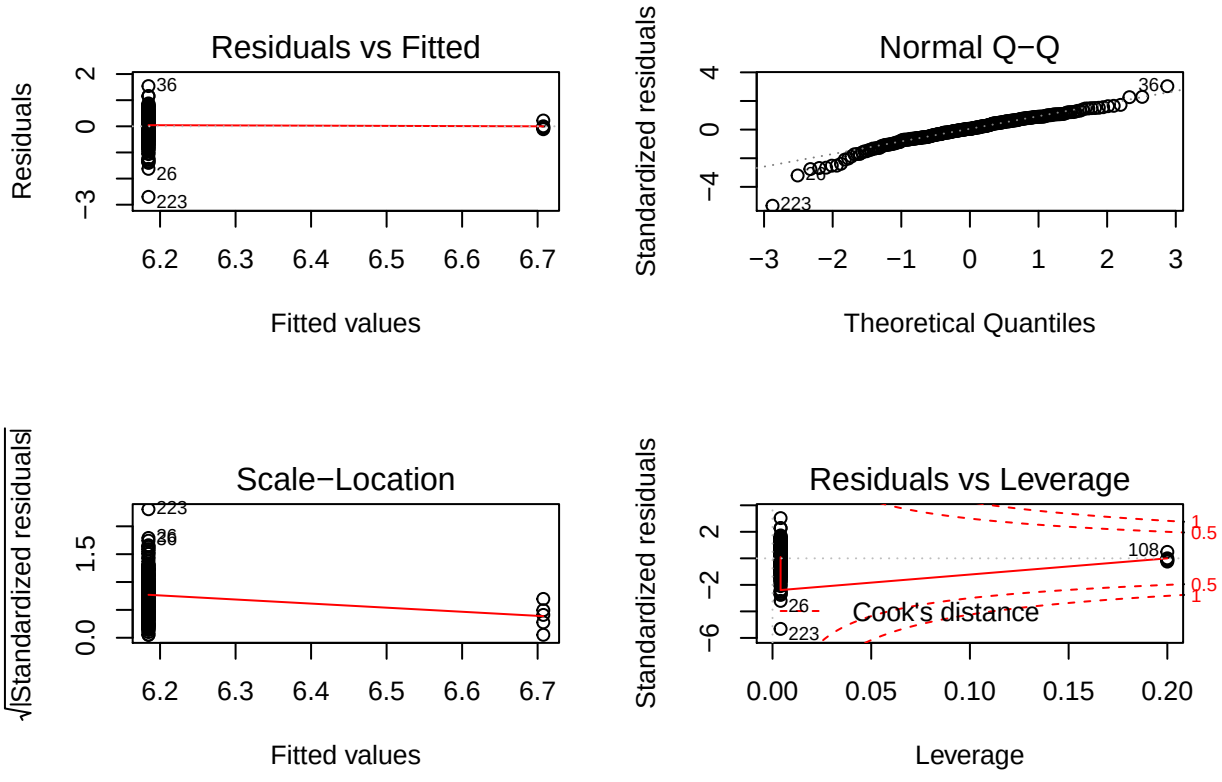


2. Try log(De-seasonalized Sales) Answer, need to multiply weekindex?

```
bluestem$log_desSales <- log(bluestem$desSales)
mtp.fit.2=lm(formula = 'log_desSales ~ Promotion', data = bluestem)
summary(mtp.fit.2)
```

```
##
## Call:
## lm(formula = "log_desSales ~ Promotion", data = bluestem)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7001 -0.2730  0.0291  0.3303  1.5398
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.18466    0.03257 189.917   <2e-16 ***
## Promotion1   0.52284    0.22981   2.275   0.0238 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5087 on 247 degrees of freedom
## Multiple R-squared:  0.02053,    Adjusted R-squared:  0.01656
## F-statistic: 5.176 on 1 and 247 DF,  p-value: 0.02376
```

```
# Checking model assumptions
par(mfrow=c(2,2))
plot(mtp.fit.2)
```




```
mtp.fit.2$coefficients[2]
```

```
## Promotion1
##  0.5228393
```

```r
# Promotion increases sales without Weekday effect by 52.284%. Now factor in the Saturday effect in ter
mtp.fit.2$coefficients[2] * ((index_table$Index[which(index_table$Weekday=="Saturday")] - 1)/1)
```

```
## Promotion1
##  0.4390509
```

Answer = 97.77094%

Comparison

Model assumptions:
The additive model is much better in terms of R-square and residual plots. The multiplicative model using log(De-seasonalized Sales) is better than the one directly using De-seasonalized Sales.
Business perspective:
The multiplicative model is better than the additive model. The additive model assumes that promotions have same boost effect on all weekdays, which is not the case. Promotional events should have better results with larger customer traffic. This is better addressed by the multiplicative model.