

RETAIL ANALYSIS on WALMART by Sahil Nasa

//This file contains all the necessary insights and outputs for the dataset “Walmart_Store_sales.csv”

Statistical Analysis

➤ Maximum sales: Store 20

The screenshot shows the RStudio interface with the following components:

- Script Editor:** Contains R code for summarizing and arranging data by store to find the maximum sales.
- Console:** Displays the output of the R script, showing a tibble with store IDs and total sales.
- Environment:** Lists the objects in the global environment, including 'res', 'wlm', 'wlm_copy', 'wlm_max_sales', 'wlm_max_sales...', and 'wlm_missing'.
- Files:** Shows the project files, including 'Retail-Project.R', 'wlm_max_sales', and 'wlm'.

```
18 wlm_copy <- wlm
19
20 #-----
21 #which store has max sales and stdev
22 library('dplyr')
23
24 wlm_max_sales <- summarize(group_by(wlm,Store),total_sales=sum(weekly_
25 view(wlm_max_sales)
26 res <- arrange(wlm_max_sales,desc(total_sales))
27
```

```
> res <- arrange(wlm_max_sales,desc(total_sales))
> res
# A tibble: 45 x 2
  store total_sales
  <int>     <dbl>
1    20 301397792.
2     4 299543953.
3    14 288999911.
4    13 286517704.
5     2 275382441.
6    10 271617714.
7    27 253855917.
8     6 223756131.
9     1 222402809.
10   39 207445542.
# with 35 more rows
```

➤ Max Standard Deviation: Store 14

The screenshot shows the RStudio interface with the following components:

- Script Editor:** Contains R code for summarizing and arranging data by store to find the maximum standard deviation.
- Console:** Displays the output of the R script, showing a tibble with store IDs and standard deviation.
- Environment:** Lists the objects in the global environment, including 'res', 'wlm', 'wlm_copy', 'wlm_max_sales', 'wlm_maxstdev', 'wlm_maxstdev...', and 'wlm_missing'.
- Files:** Shows the project files, including 'Retail-Project.R', 'wlm_maxstdev', 'wlm_max_sales', and 'wlm'.

```
29 #which store has max stdev
30
31
32 wlm_maxstdev <- summarize(group_by(wlm,Store),stdev=sd(weekly_Sales))
33 view(wlm_maxstdev)
34 res <- arrange(wlm_maxstdev,desc(stdev))
35 # By seeing the output, we can see Store 14 has the most variation
36
37 #Find out coeff of mean to stdev or better to say COEFF OF VARIATION
38 wlm_cv <- summarize(group_by(wlm,Store),cv=sd(weekly_Sales)/mean(weekl
39 view(wlm_cv)
40
41 #-----
42
```

```
> res <- arrange(wlm_maxstdev,desc(stdev))
> res
# A tibble: 45 x 2
  store stdev
  <int>   <dbl>
1    14 317570.
2    10 302262.
3    20 275901.
4     4 266201.
5    13 265507.
6    23 249788.
7    27 239930.
```

➤ Coefficient of Variance:

The screenshot shows the RStudio interface with the following components:

- Environment Pane:** Lists variables in the Global Environment:
 - res: 45 obs. of 2 variables
 - wlm: 6435 obs. of 8 variables
 - wlm_copy: 6435 obs. of 8 variables
 - wlm_cv: 45 obs. of 2 variables
 - wlm_maxsal...: 45 obs. of 2 variables
 - wlm_maxstdev: 45 obs. of 2 variables
 - wlm_missing: 0 obs. of 8 variables
 - x: logi [1:6435, 1:8] FALSE FALS...
- Console:** Shows the R code used to calculate the Coefficient of Variance (CV):


```
# ... with 35 more rows
> #Find out coeff of mean to stdev or better to say COEFF OF VARIATION
> wlm_cv <- summarize(group_by(wlm,Store),CV=sd(Weekly_Sales)/mean(Weekly_Sales))
`summarise()` ungrouping output (override with `.groups` argument)
> view(wlm_cv)
>
```

➤ Good quarterly growth rate in Quarter 3 of 2012:

Store 7 has max quarterly growth rate (refer to source code for better explanation)

The screenshot shows the RStudio interface with the following components:

- Source Editor:** Contains the R code for calculating the Quarterly Growth Rate (QGR):


```
50 wlm_qgr <- arrange(wlm_qgr,desc(QGR))
51 wlm_qgr
52 #Seeing the weekly sales of a quarter for each store, Store 7 has the max
53 #quarterly growth rate which is 13.3% followed by store 16 and 35.
54 #-----Holidays Impact-----
55
```
- Console:** Shows the R code and the output of the QGR calculation:


```
> wlm_qgr$lag1 <- lag(wlm_qgr$qtr_sales,1)
> wlm_qgr$QGR <- ((wlm_qgr$qtr_sales - wlm_qgr$lag1)*100/wlm_qgr$lag1)
> wlm_qgr <- filter(wlm_qgr,qtr=="2012.3")
> wlm_qgr <- arrange(wlm_qgr,desc(QGR))
> wlm_qgr
# A tibble: 45 x 5
# Groups:   store [45]
  store qtr qtr_sales lag1 QGR
  <int> <dbl> <dbl> <dbl> <dbl>
1 7 2012. 8262787. 7290859. 13.3
2 16 2012. 7121542. 6564336. 8.49
3 35 2012. 11322421. 10838313 4.47
4 26 2012. 13675692. 13155336. 3.96
5 39 2012. 20715116. 20214128. 2.48
```

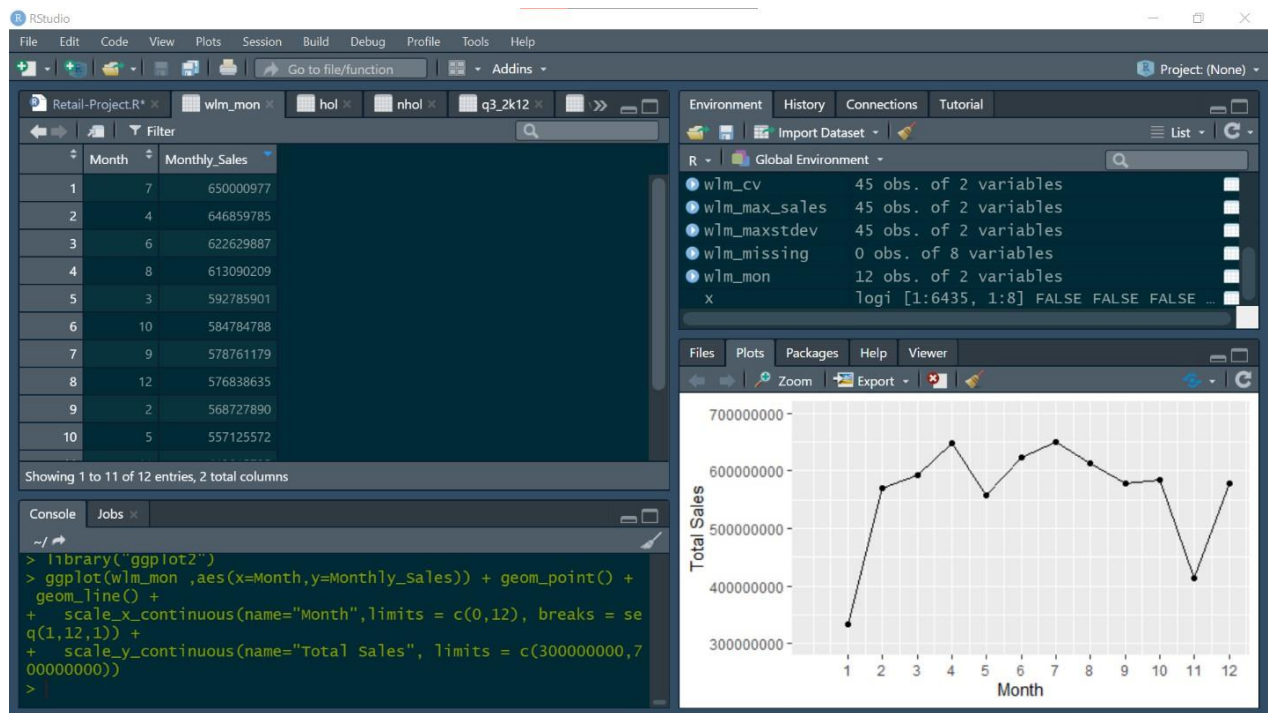
- Holidays with higher sales than the average sales in non-holiday season:
“hol” dataset contains all the desired entries

The screenshot shows the RStudio interface. The main editor displays a data frame with columns: Store, Date, Weekly_Sales, Holiday_Flag, Temperature, Fuel_Price, CPI, and Unemployment. The first 10 rows are visible, showing data for various dates in 2010 and 2011. The Environment pane on the right lists the datasets: hol (220 obs. of 8 variables), nhol (1 obs. of 1 variable), q3_2k12 (45 obs. of 4 variables), res (45 obs. of 2 variables), wlm (6435 obs. of 8 variables), wlm_copy (540 obs. of 4 variables), wlm_cv (45 obs. of 2 variables), wlm_max_sal... (45 obs. of 2 variables), wlm_maxstdev (45 obs. of 2 variables), wlm_missing (0 obs. of 8 variables), and x (logi [1:6435, 1:8] FALSE FALSE...). The Console pane at the bottom shows the following R commands:

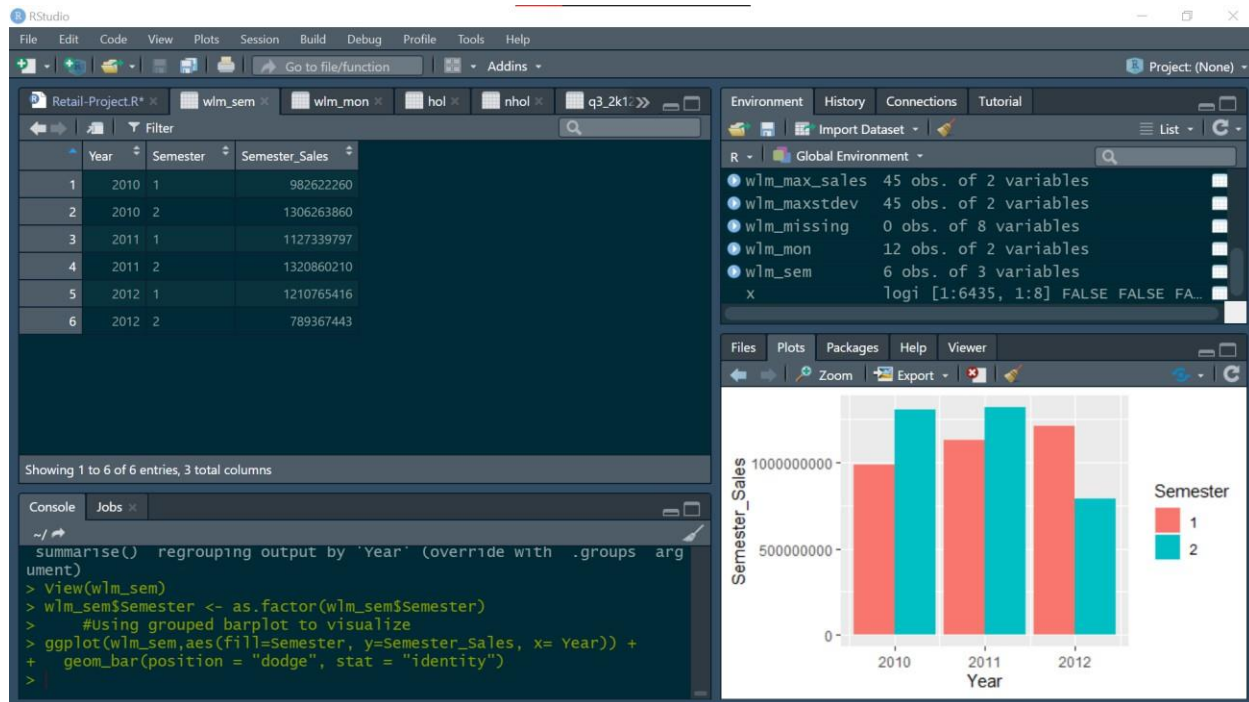
```
> nhol <- filter(wlm, Holiday_Flag==0)
> View(nhol)
> nhol <- summarize(nhol, Avg_sales=mean(weekly_Sales))
> #Average sales on Holidays (getting those whose weekly sales are
> #higher than the sales in non-holiday days)
> hol <- filter(wlm, Holiday_Flag==1, weekly_Sales>1041256)
> View(hol)
>
```

1. Monthly and semester view of sales in units and insights

- >> Monthly Sales Insights: Sales is least in JAN but shoots up in FEB by a lot.
Main insight is Sales decreases a lot in NOV but increases a lot in DEC.



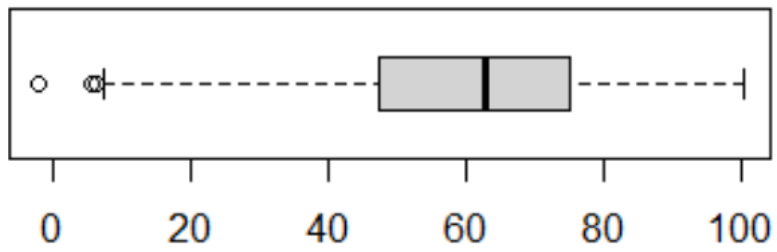
>> Semester sales insights: Sales increased in every second semester except for year 2012



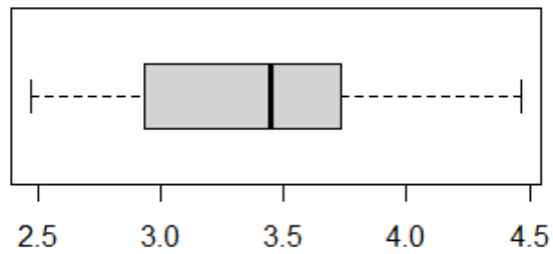
LINEAR REGRESSION MODEL

Boxplots for OUTLIER DETECTION

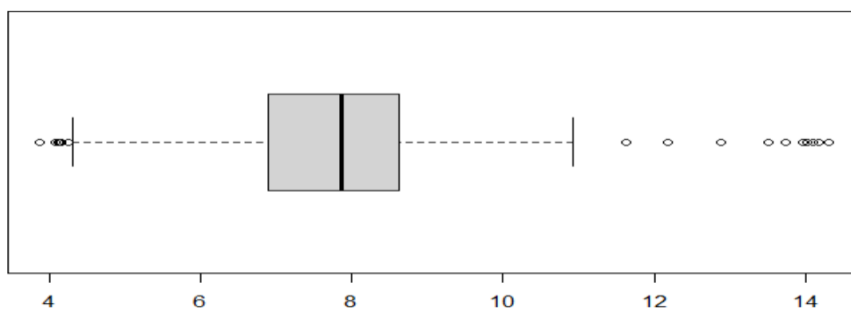
- Temperature (OUTLIERS <10)



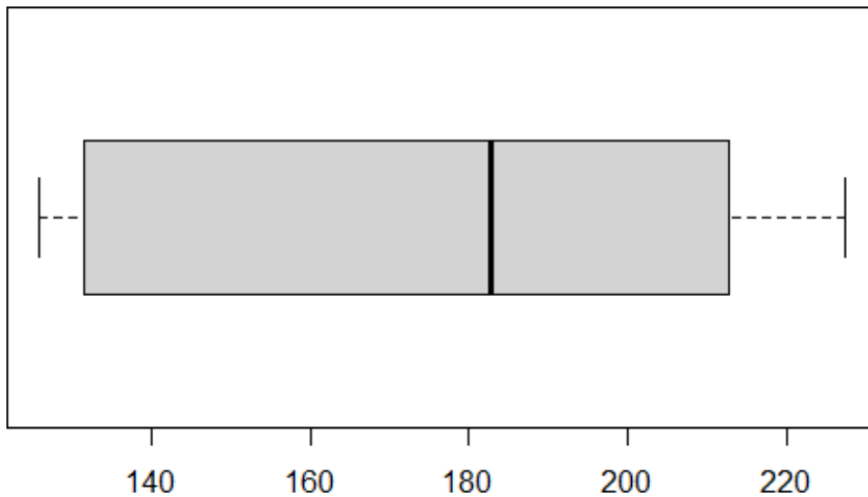
- Fuel Price (No outliers)



- Unemployment (Outliers <4.5 and >10)



- CPI (No outliers)



REGRESSION MODEL OUTPUTS:

Model before hypothesizing / dropping variables on the basis of p-values

```

Console  Jobs x
~/
> wlm_test <- subset(wlm_new, sel==FALSE)
> wlm_model <- lm(Weekly_Sales ~ Fuel_Price+Temperature+Holiday_Flag+CPI+Unemployment, data = wlm_new)
> summary(wlm_model)

Call:
lm(formula = weekly_Sales ~ Fuel_Price + Temperature + Holiday_Flag +
    CPI + Unemployment, data = wlm_new)

Residuals:
    Min       1Q   Median       3Q      Max
-305166  -78247  -18260   53643   854412

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -2427856    1752958  -1.385   0.1683
Fuel_Price     -24337      47335  -0.514   0.6080
Temperature    -2160         922  -2.343   0.0206 *
Holiday_Flag1   89376     49338   1.811   0.0723 .
CPI             16632      6786   2.451   0.0155 *
Unemployment   80209     58727   1.366   0.1742
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 146500 on 137 degrees of freedom
Multiple R-squared:  0.1495,    Adjusted R-squared:  0.1184
F-statistic: 4.815 on 5 and 137 DF,  p-value: 0.0004359

```

Output after dropping variables on the basis of p-values :

```
Console Jobs x
~/ ➔
Residual standard error: 146500 on 137 degrees of freedom
Multiple R-squared:  0.1495,    Adjusted R-squared:  0.1184
F-statistic: 4.815 on 5 and 137 DF,  p-value: 0.0004359

> #Dropping non-significant variables seeing the p-values : Holiday,Day,Year
> wlm_model <- lm(Weekly_Sales ~ Temperature+CPI, data = wlm_new)
> summary(wlm_model)

Call:
lm(formula = Weekly_Sales ~ Temperature + CPI, data = wlm_new)

Residuals:
    Min       1Q   Median       3Q      Max
-312205  -85704   -9198    57222   830489

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -233190     616327  -0.378  0.70574
Temperature    -2769         877  -3.157  0.00195 **
CPI             9156        2872   3.187  0.00177 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 147900 on 140 degrees of freedom
Multiple R-squared:  0.1139,    Adjusted R-squared:  0.1012
F-statistic: 8.998 on 2 and 140 DF,  p-value: 0.0002107

> |
```