R Final Project
20226
Restaurant Rankings 2020 Dataset

Restaurant. This is a place which each of us must have visited at least once. The varieties of items in the menu would be going on and on. And half the time would be spent on discussing what to order among the wide variants of dishes. And the Delicacies would make us feel heavenly. Tasty food and wonderful service would make customers revisit the place at least once. Various factors affecting the sales and revenues can be seen in the dataset given below.

For a dataset to be analysed completely, we need to use all the tools available. As inference from each tool will be unique and extremely useful. We use some of them to get the inferences we need.

Let's have a peek into the datasets,
Firstly, the Future50

	R... <int>	Restaurant <chr>	Location <chr>	Sales <int>	YOY_Sales <chr>	Units <int>	YOY_Units <chr>	Unit_Volume <int>	Franchising <chr>
1	1	Evergreens	Seattle, Wash.	24	130.5%	26	116.7%	1150	No
2	2	Clean Juice	Charlotte, N.C.	44	121.9%	105	94.4%	560	Yes
3	3	Slapfish	Huntington Beach, Calif.	21	81.0%	21	90.9%	1370	Yes
4	4	Clean Eatz	Wilmington, N.C.	25	79.7%	46	58.6%	685	Yes
5	5	Pokeworks	Irvine, Calif.	49	77.1%	50	56.3%	1210	Yes
6	6	Playa Bowls	Belmar, N.J.	39	62.9%	76	28.8%	580	Yes

```
## Rows: 50
## Columns: 9
## $ Rank        <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17…
## $ Restaurant  <chr> "Evergreens", "Clean Juice", "Slapfish", "Clean Eatz", "P…
## $ Location    <chr> "Seattle, Wash.", "Charlotte, N.C.", "Huntington Beach, C…
## $ Sales       <int> 24, 44, 21, 25, 49, 39, 24, 20, 24, 29, 30, 39, 41, 48, 2…
## $ YOY_Sales   <chr> "130.5%", "121.9%", "81.0%", "79.7%", "77.1%", "62.9%", "…
## $ Units       <int> 26, 105, 21, 46, 50, 76, 36, 19, 60, 17, 41, 50, 63, 48, …
## $ YOY_Units   <chr> "116.7%", "94.4%", "90.9%", "58.6%", "56.3%", "28.8%", "3…
## $ Unit_Volume <int> 1150, 560, 1370, 685, 1210, 580, 775, 1260, 465, 1930, 86…
## $ Franchising <chr> "No", "Yes", "Yes", "Yes", "Yes", "Yes", "Yes", "Yes", "Y…
```

So the above are the parameters which are used for the evaluation of the rank of a restaurant. This Future50 consists of a total 50 entries and 9 columns. And the parameters being Sales, YOY_Sales, Units, Units_Volume, YOY_Units, Franchising. Even the location of a restaurant matters a lot in terms of earning good revenues.

Next is the Independence100, let's look at it…

	R…	Restaurant	Sales	Average.Check	City	State	Meals.Served
	<int>	<chr>	<dbl>	<int>	<chr>	<chr>	<dbl>
1	1	Carmine's (Times Square)	39080335	40	New York	N.Y.	469803
2	2	The Boathouse Orlando	35218364	43	Orlando	Fla.	820819
3	3	Old Ebbitt Grill	29104017	33	Washington	D.C.	892830
4	4	LAVO Italian Restaurant & Nightclub	26916180	90	New York	N.Y.	198500
5	5	Bryant Park Grill & Cafe	26900000	62	New York	N.Y.	403000
6	6	Gibsons Bar & Steakhouse	25409952	80	Chicago	Ill.	348567

```
## Rows: 100
## Columns: 7
## $ Rank          <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, …
## $ Restaurant    <chr> "Carmine's (Times Square)", "The Boathouse Orlando", "O…
## $ Sales         <dbl> 39080335, 35218364, 29104017, 26916180, 26900000, 25409…
## $ Average.Check <int> 40, 43, 33, 90, 62, 80, 103, 99, 87, 107, 75, 135, 86, …
## $ City          <chr> "New York", "Orlando ", "Washington", "New York", "New …
## $ State         <chr> "N.Y.", "Fla.", "D.C.", "N.Y.", "N.Y.", "Ill.", "Nev.",…
## $ Meals.Served  <dbl> 469803, 820819, 892830, 198500, 403000, 348567, 246054,…
```

Similar to the Future50, this consists of Sales. Moreover this consists of extra features like Average check,i.e. numbers of sales by number of customers, And also the Meals Served. Consists of 100 rows and 7 columns.

Next in the line is, the Top250…

	R... <int>	Restaurant <chr>	Content <chr>	Sales <int>	YOY_Sales <chr>	Units <int>	YOY_Units <chr>	Headquarters <chr>	Segment_Category <chr>
245	245	Gyu-Kaku	*NA*	129	18.6%	52	8.3%	*NA*	Asian
246	246	Rainforest Cafe	*NA*	129	-10.4%	18	-5.3%	*NA*	Varied Menu
247	247	PDQ	*NA*	127	-5.5%	56	-11.1%	*NA*	Chicken
248	248	Lupe Tortilla	*NA*	127	12.1%	25	8.7%	*NA*	Mexican
249	249	Cook-Out Restaurant	*NA*	126	10.1%	270	7.1%	*NA*	Burger
250	250	Jollibee	*NA*	126	15.2%	40	11.1%	*NA*	Chicken

```
## Rows: 250
## Columns: 9
## $ Rank             <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 1…
## $ Restaurant       <chr> "McDonald's", "Starbucks", "Chick-fil-A", "Taco Bell…
## $ Content          <chr> NA, NA, "While Popeyes got a lot of the chicken buzz…
## $ Sales            <int> 40412, 21380, 11320, 11293, 10204, 10200, 9762, 9228…
## $ YOY_Sales        <chr> "4.9%", "8.6%", "13.0%", "9.0%", "2.7%", "-2.0%", "4…
## $ Units            <int> 13846, 15049, 2470, 6766, 7346, 23801, 5852, 9630, 6…
## $ YOY_Units        <chr> "-0.5%", "3.0%", "5.0%", "2.7%", "0.2%", "-4.0%", "0…
## $ Headquarters     <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, …
## $ Segment_Category <chr> "Quick Service & Burger", "Quick Service & Coffee Ca…
```

Among the other features the Top250 consists of Content and Segment Category. We can have a look at tail entries of the data.

Summaries of each dataset:
Future50:

```
      Rank        Restaurant          Location
 Min.   : 1.00   Length:50          Length:50
 1st Qu.:13.25   Class :character   Class :character
 Median :25.50   Mode  :character   Mode  :character
 Mean   :25.50
 3rd Qu.:37.75
 Max.   :50.00
     Sales        YOY_Sales            Units
 Min.   :20.00   Length:50          Min.   :  7.0
 1st Qu.:24.25   Class :character   1st Qu.: 16.0
 Median :34.50   Mode  :character   Median : 27.0
 Mean   :33.78                      Mean   : 34.7
 3rd Qu.:42.00                      3rd Qu.: 45.5
 Max.   :49.00                      Max.   :105.0
  YOY_Units          Unit_Volume     Franchising
 Length:50          Min.   : 465.0   Length:50
 Class :character   1st Qu.: 867.5   Class :character
 Mode  :character   Median :1260.0   Mode  :character
                    Mean   :1592.6
                    3rd Qu.:2020.0
                    Max.   :4300.0
   yoy_sales        yoy_units            y
 Min.   : 14.40   Min.   :  4.00   Min.   :3.291
 1st Qu.: 20.90   1st Qu.: 14.30   1st Qu.:3.859
 Median : 25.50   Median : 19.90   Median :5.062
 Mean   : 33.70   Mean   : 27.45   Mean   :4.918
 3rd Qu.: 33.83   3rd Qu.: 32.67   3rd Qu.:5.848
 Max.   :130.50   Max.   :116.70   Max.   :6.521
```

Independence100:

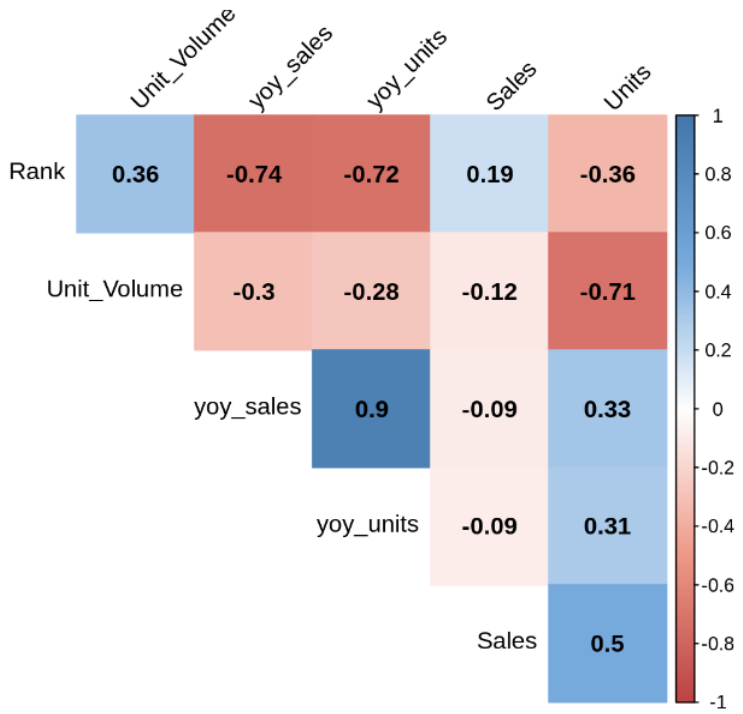
```
     Rank         Restaurant            Sales       
Min.   :  1.00   Length:100         Min.   :11391678  
1st Qu.: 25.75   Class :character   1st Qu.:14094836  
Median : 50.50   Mode  :character   Median :17300776  
Mean   : 50.50                      Mean   :17833434  
3rd Qu.: 75.25                      3rd Qu.:19903916  
Max.   :100.00                      Max.   :39080335  
Average.Check        City              State          
Min.   : 17.00   Length:100         Length:100        
1st Qu.: 39.00   Class :character   Class :character  
Median : 65.50   Mode  :character   Mode  :character  
Mean   : 69.05                                        
3rd Qu.: 95.00                                        
Max.   :194.00                                        
 Meals.Served          y          
Min.   : 87070   Min.   :0.8115  
1st Qu.:189492   1st Qu.:0.8115  
Median :257097   Median :0.8115  
Mean   :317167   Mean   :0.8115  
3rd Qu.:372079   3rd Qu.:0.8115  
Max.   :959026   Max.   :0.8115  
```

Top250:

```
     Rank         Restaurant          Content
Min.   :  1.00   Length:250         Length:250
1st Qu.: 63.25   Class :character   Class :character
Median :125.50   Mode  :character   Mode  :character
Mean   :125.50
3rd Qu.:187.75
Max.   :250.00
    Sales          YOY_Sales            Units
Min.   :  126.0   Length:250         Min.   :   13.0
1st Qu.:  181.0   Class :character   1st Qu.:   85.0
Median :  330.0   Mode  :character   Median :  207.0
Mean   : 1242.7                      Mean   :  850.1
"https://cloud.r-project.org"                  3rd Qu.:  555.2
Max.   :40412.0                      Max.   :23801.0
 YOY_Units         Headquarters       Segment_Category
Length:250         Length:250         Length:250
Class :character   Class :character   Class :character
Mode  :character   Mode  :character   Mode  :character



  yoy_sales         yoy_units             y
Min.   :-21.200   Min.   :-32.800   Min.   :1.338
1st Qu.: -2.375   1st Qu.: -2.025   1st Qu.:1.345
Median :  2.200   Median :  0.000   Median :1.351
Mean   :  2.938   Mean   :  1.219   Mean   :1.351
3rd Qu.:  6.575   3rd Qu.:  3.475   3rd Qu.:1.358
Max.   : 39.500   Max.   : 38.500   Max.   :1.365
```

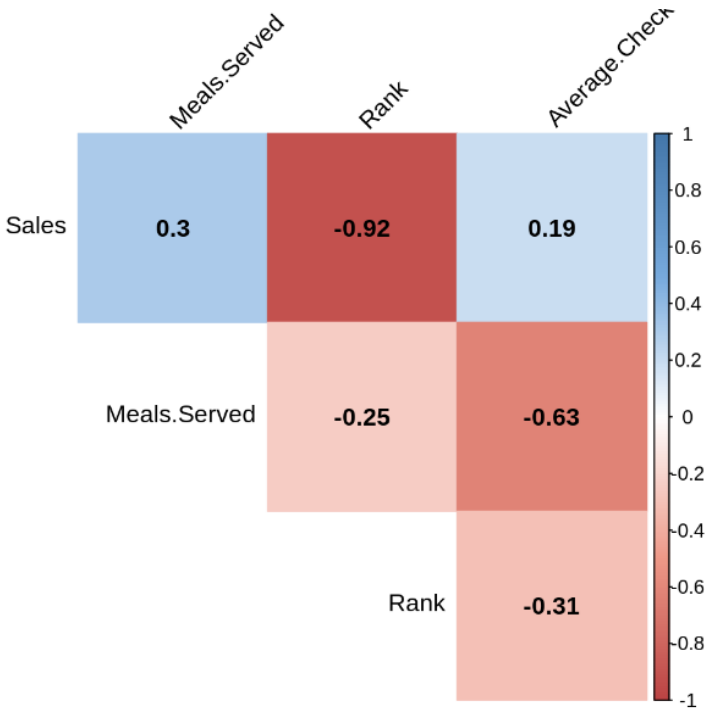
EDA(Exploratory Data Analysis)

First in the line, Correlation plot of Future50



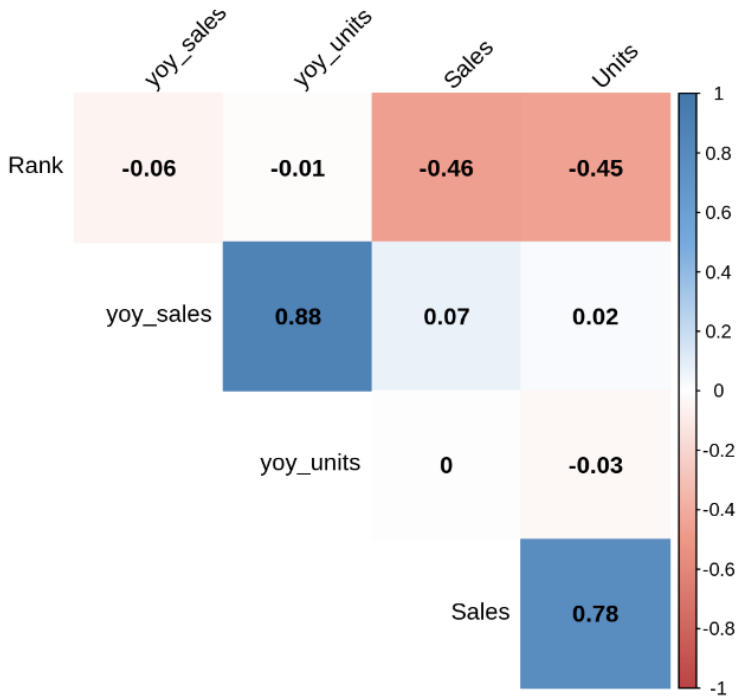
Here, it's apparent that yoy_sales and yoy_units are highly positively correlated, which implies that an increase in yoy_units leads to an increase in yoy_sales. And the yoy_sales and yoy_units are highly negatively correlated implying, Higher the yoy_sales and yoy_units, Higher the rank(as High rank implies small number).

Next, the correlation plot of Independence100



Conversely, Here Sales and Rank are highly negatively correlated implying Higher the Sales ,Higher the Rank. And the important thing to notice here is, Meals.Served and Average.Check are Negatively correlated to the extent of 0.63. This is because, Average.Check is the ratio between Number of Sales and Number of customers. As Meals.Served increases, the denominator of Average.Check increases which results in less Average.check. They are inversely proportional.

Next comes, the top250…



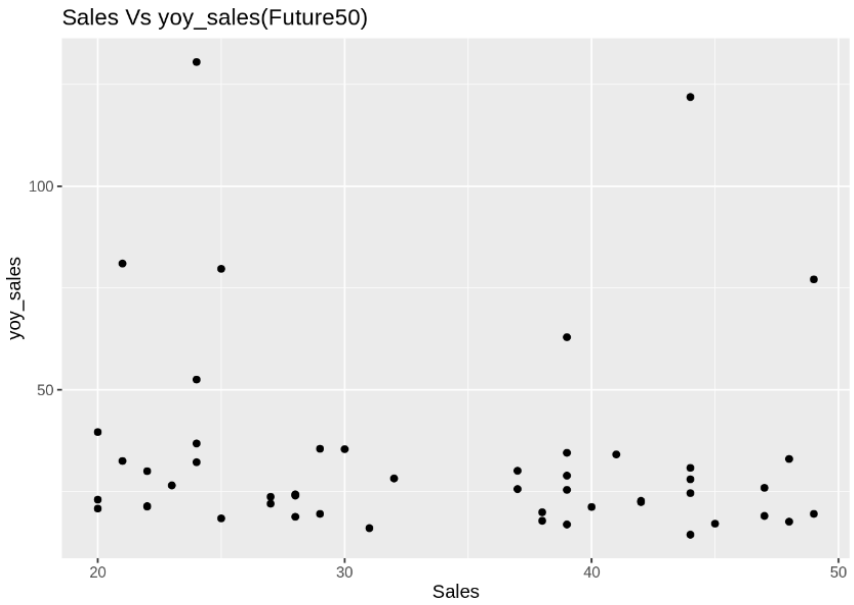
Similar to future50, Here yoy_sales and yoy_units are highly positively correlated. And to some extent Rank and Sales, Rank and Units are negatively correlated. But they are not very significant. Whereas Sales and Units are highly positively correlated.

Scatter Plot

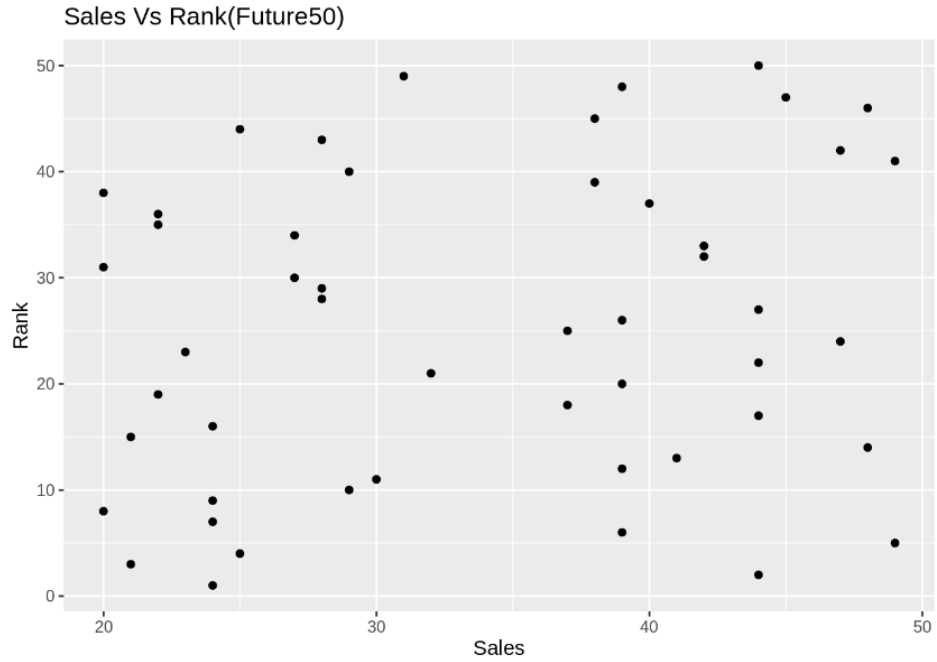
The scatter plot shows as to how the values are scattered across the graph.

Firstly, Future50…

Here,An Increase in sales does not exactly suggest that there is an increase in yoy_sales. Even when sales are low, yoy_sales is high. So this implies that yoy_sales can be more even when the sales are low. Unlike Rank and Sales, yoy_sales depend on the previous years' data.

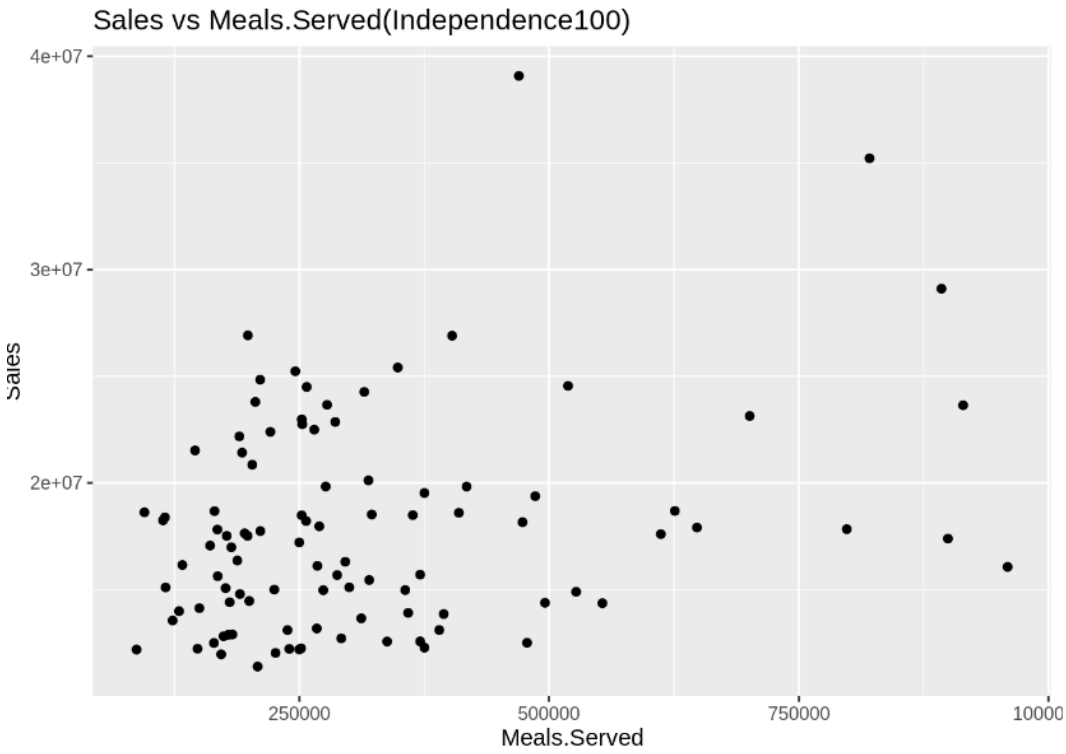


The above graph suggests the aforementioned information.



The above scatter plot, clearly shows that Sales is not the only feature that decides the rank. As Sales of 24 yields rank 1 whereas Sales of 47 yields a rank near to 50.

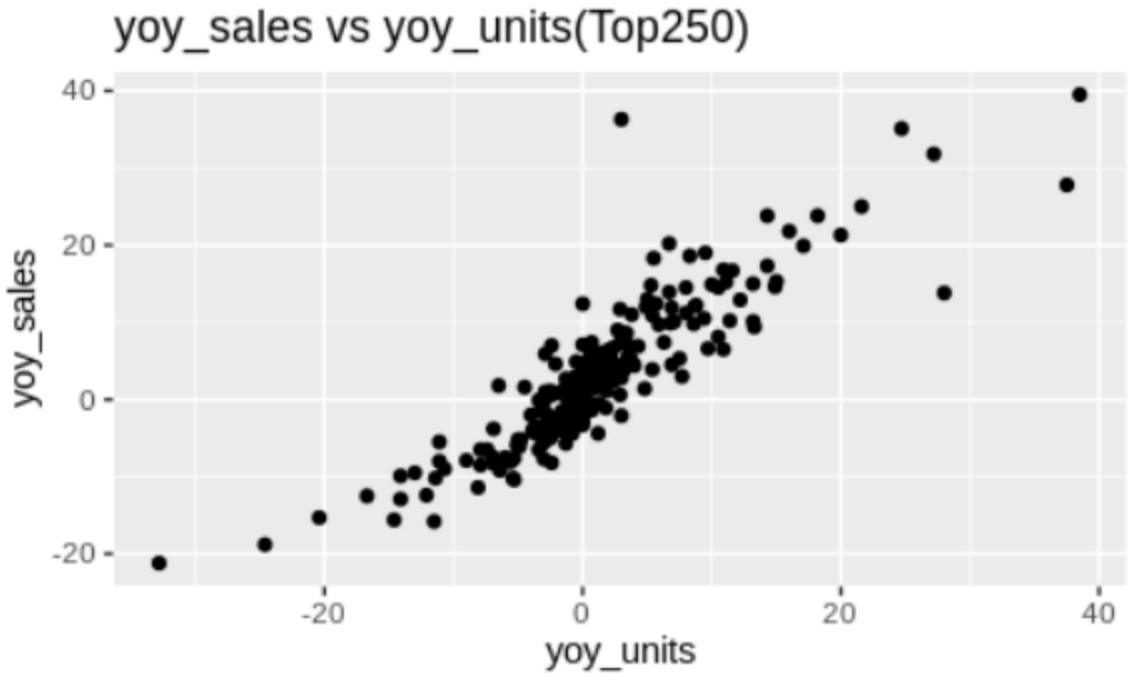
Next is , Independence100



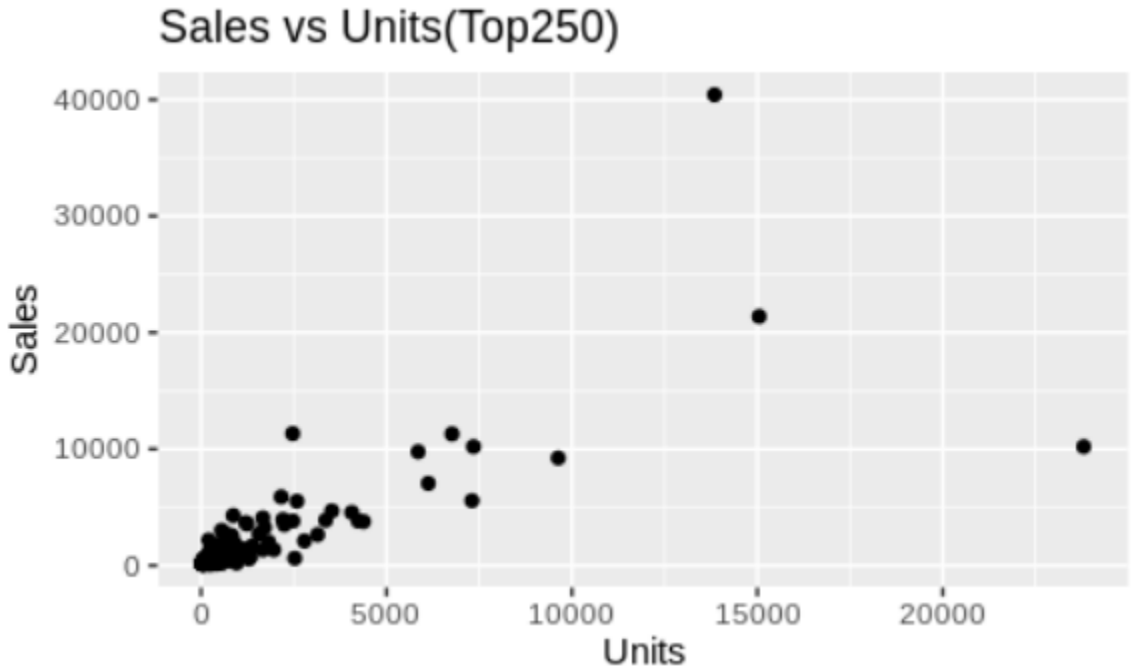
Sales and Meals.Served mostly centered at 250000 Meals.Served and 20000000 Sales. So many of the restaurants in Independence100 have medium range Meals.served the proportional Sales. Few outliers, where meals.served is more but sales are less and vice versa.

Top250

It's evident here that yoy_sales of most of the restaurants are around zero. The range of yoy_sales and yoy_units is approx. -20 and 40. In most cases, there is a significant increase in yoy_units, and at the same time with yoy_sales. With few outliers whose high yoy_units yielding higher yoy_sales.



And followed by Sales vs Units,

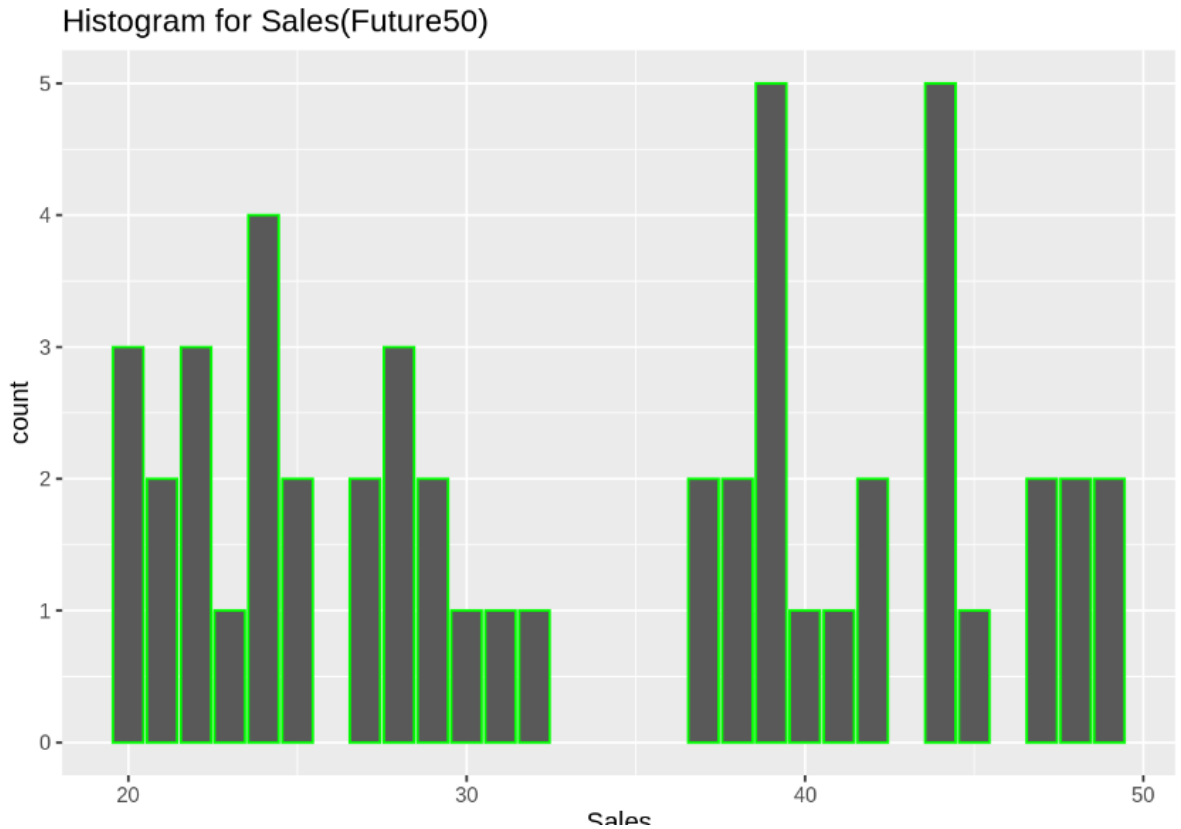


Nearly 95 % of the observations are less than 5000 units leading to nearly 10000 sales. Less than 5 % outliers are yielding less sales with more units and vice versa.

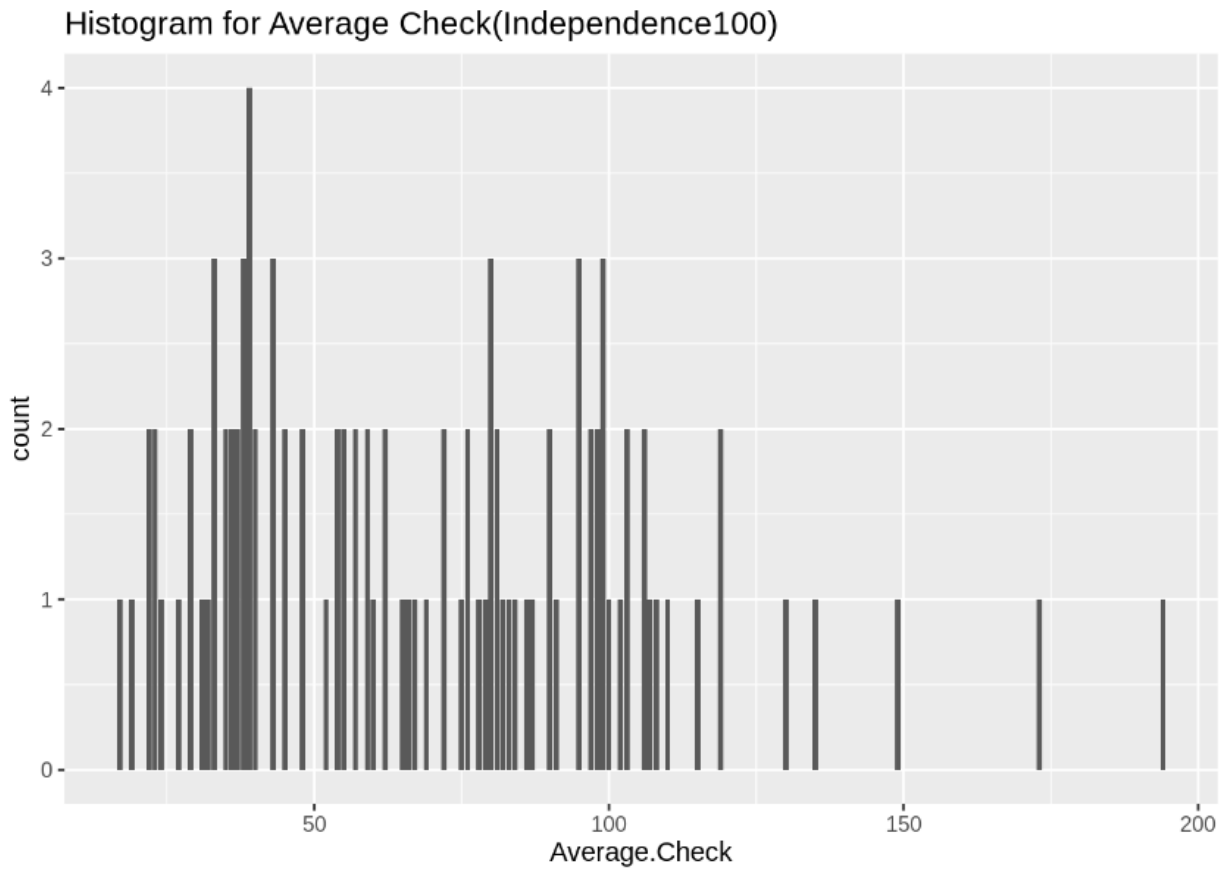
That ends the Scatter Plot story….Now let’s go to

Histogram

Histogram shows the frequency of instances to the variable given. For each of future50, Independence100,top250...we have histograms…



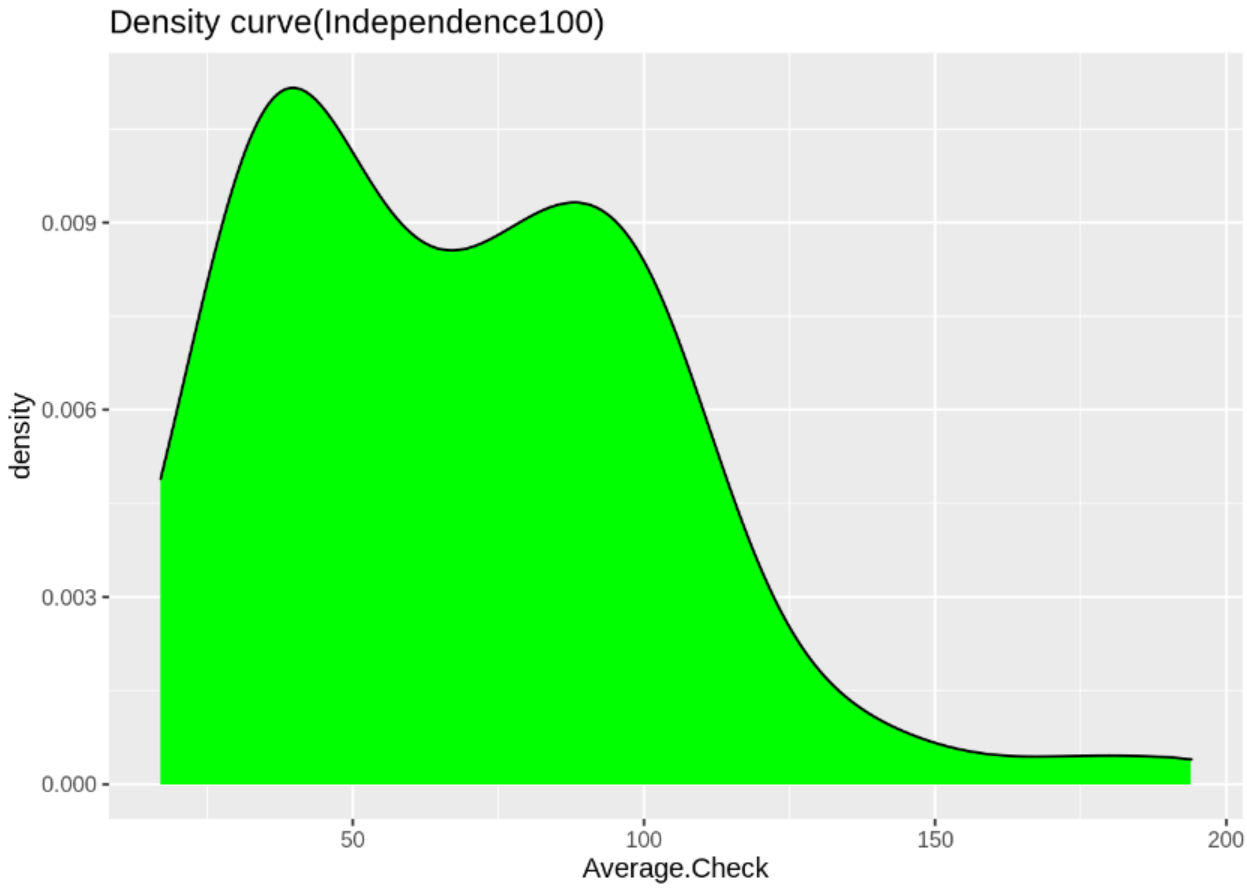
Sales ranged from 20 to approx 50. With 39 and 44 repeating for 5 times each. Nearly more than half of the restaurants' sales are between 35 and 50.



Average.check ranged between 10 and 190. Nearly 90% have less than 100 Average.Check. This average check is the number of sales by number of customers.

Density Curve

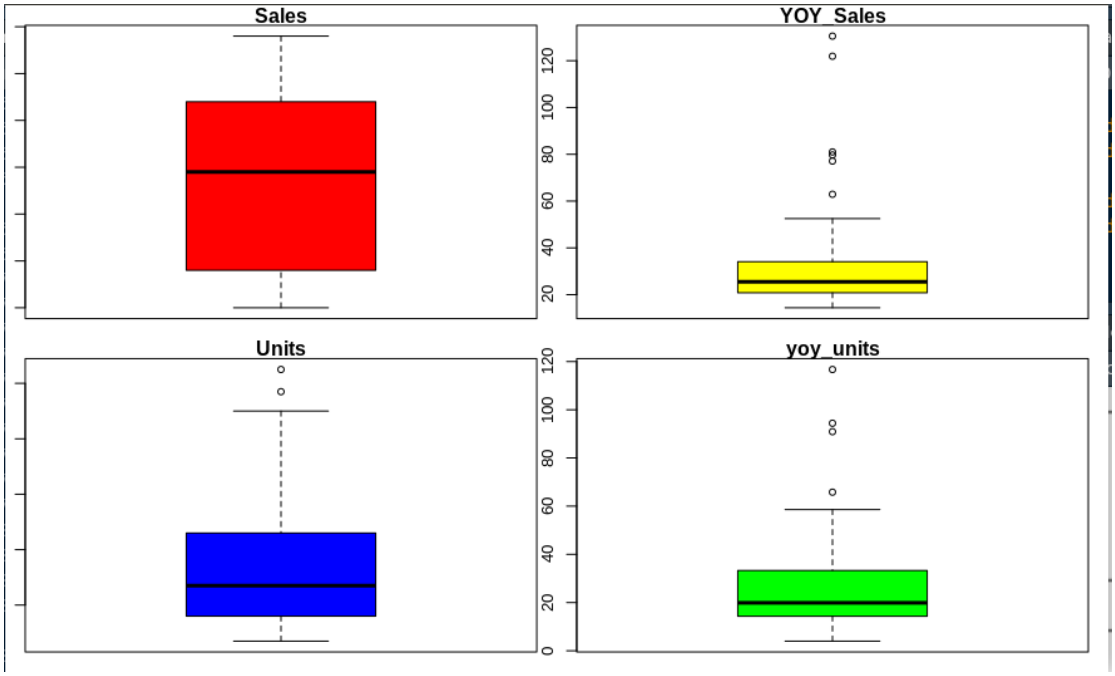
This density curve shows how data is spanned across numbers. Here this is left skewed and the curve bent down after reaching 150. This curve is showing Average.Check Density.



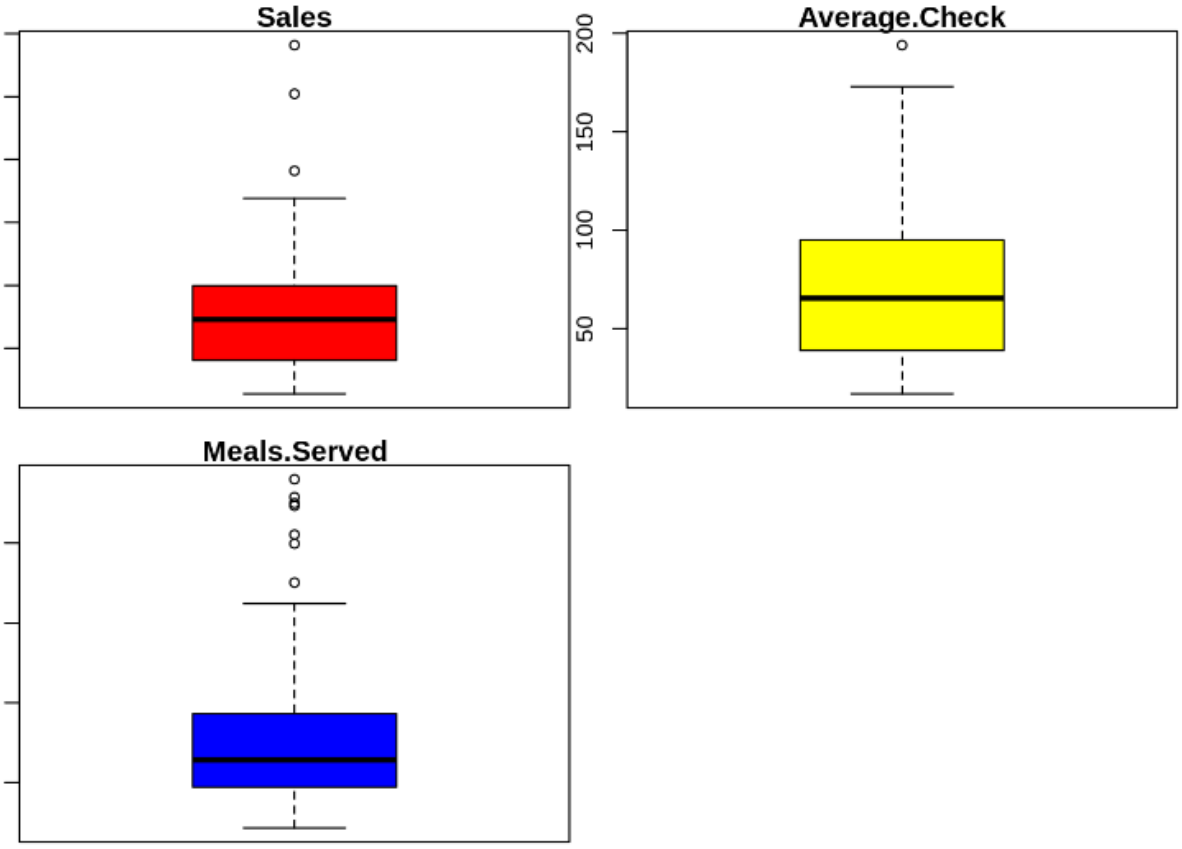
Next in the order is...Boxplot.

Boxplot

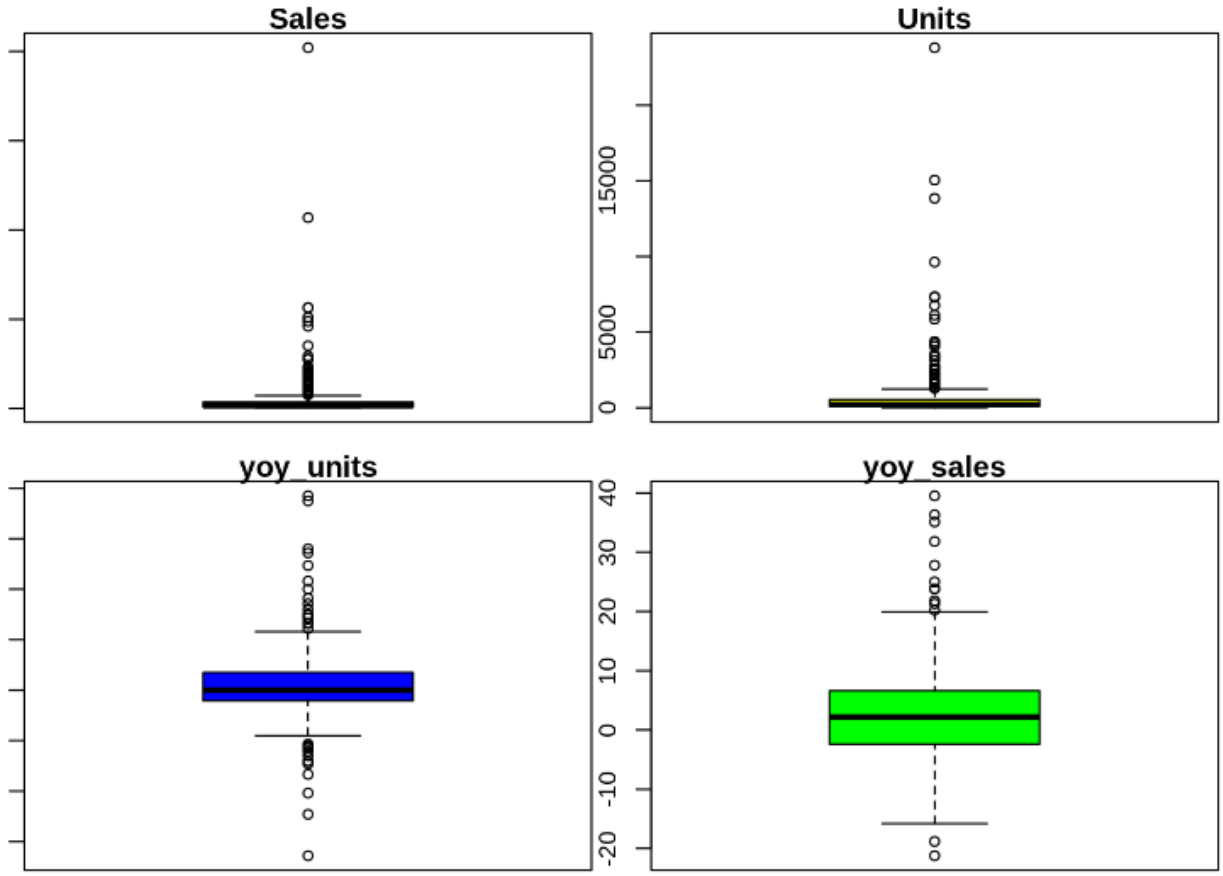
Boxplot is used to identify outliers. Outliers can be seen far away from the median point and out of the colored box.
For future50,
Sales seem to be distributed equally, whereas there seems to be many outliers in yoy_sales.
Units have just 2 to 3 outliers and yoy_units consists of nearly 5 in number.



For, Independence100…



Sales and Meals.Served have more outliers compared to Average.Check. Problem with these outliers is that they tend to pull the central tendency towards them giving out clearly the wrong picture about the data. In many cases we need to remove them if they are wrongly occurring instances.



For Top250, This consists of four boxplots…
Sales values are distributed very abnormally. And it contains many outliers. And the central point also seems to be near to zero.
Same is the case with yoy_sales, where many outliers are present and the central point for this also is near to zero.
Yoy_units has a strange scenario altogether. There are outliers before and after the central point and these outliers are many in number.
Yoy_sales follow the same suit as others and consist of outliers beyond the green box.

Next in the line is…

Piechart

Pie chart showing the percentage of Franchising restaurants present in the data.

Nearly, 58 % of them are Franchisees and remaining are non-franchise restaurants.

This finishes our EDA…
Upcoming is the Mathematical modelling…

Mathematical Modelling

Every dataset consists of regressor variables and regressand variables. We use many xi's to predict y. Various combinations of xi's are done until a best suited model is found.
First we deal with Linear Models,

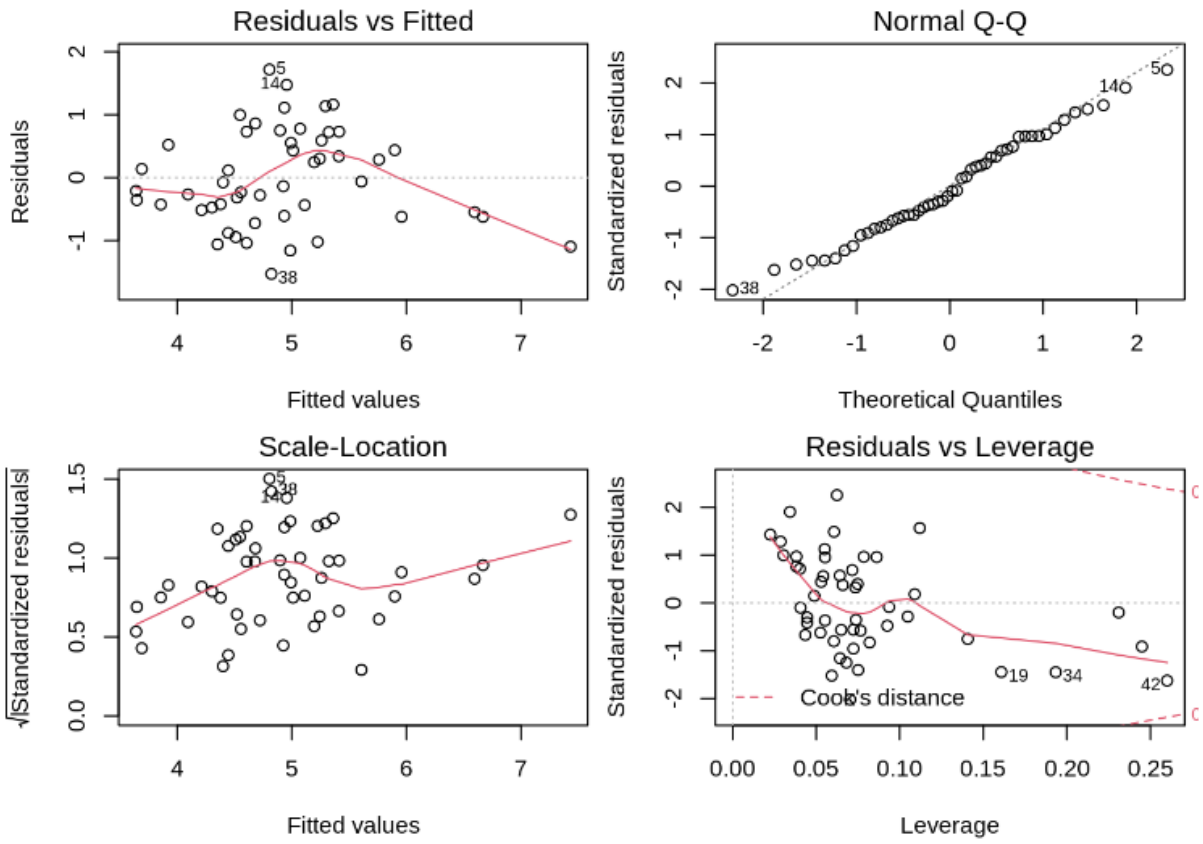
Linear Models

On to the best model of future50 dataset,

```
##
## Call:
## lm(formula = y ~ Rank + Units + Unit_Volume, data = future50_cleaned)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5309 -0.5409 -0.1067  0.5791  1.7166
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.0875502  0.5057036   4.128 0.000152 ***
## Rank        0.0275155  0.0083651   3.289 0.001931 **
## Units       0.0407001  0.0066157   6.152 1.71e-07 ***
## Unit_Volume 0.0004500  0.0001632   2.757 0.008346 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7868 on 46 degrees of freedom
## Multiple R-squared:  0.4921, Adjusted R-squared:  0.459
## F-statistic: 14.86 on 3 and 46 DF,  p-value: 6.74e-07
```

Adj.R square looks very small. But this was obtained after so many ordeals. First, I tried 3 models, whose Adj.R square is quite low. Then for the best of them, I applied boxcox , even after applying it hasn't improved much.
But to my amazement, Shapiro test showed it as normal followed by the plots…

```
##
##  Shapiro-Wilk normality test
##
## data:  bcm1$residuals
## W = 0.98043, p-value = 0.5702
```



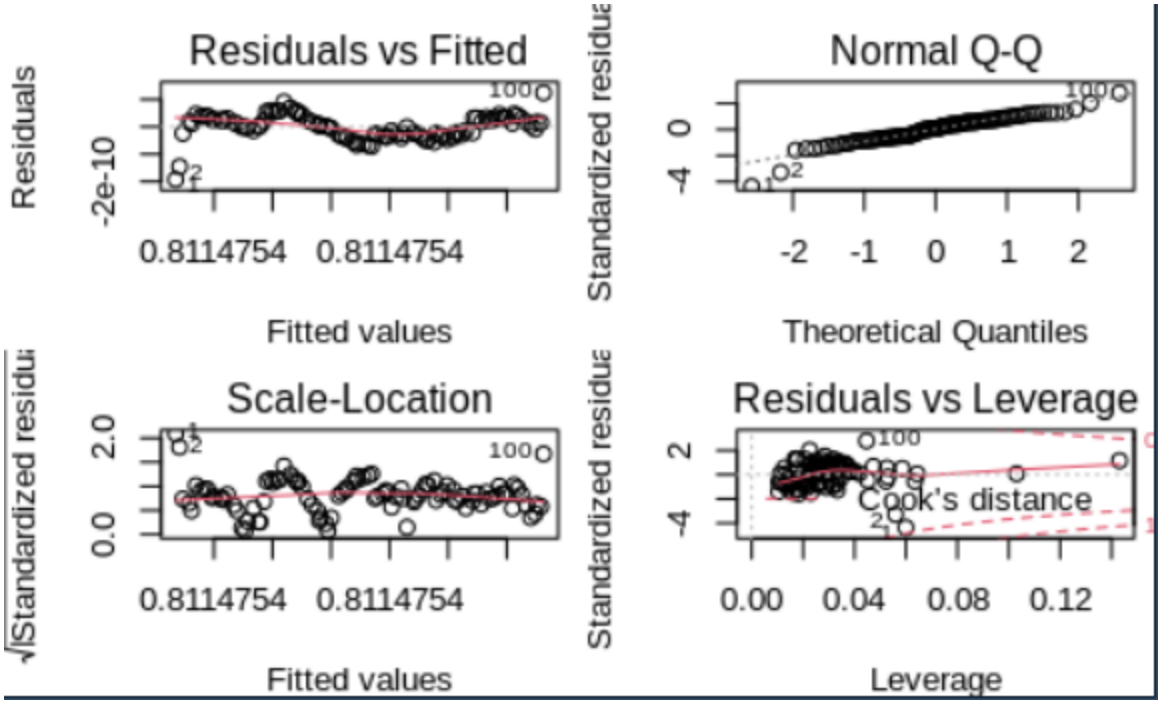
Next best model is from Independence100,
Even this model has been obtained after various boxcox transformations.
And Adj.R square is quite significant, but this doesn't assure us of the normality of the model.
Here Adj.R square is 0.98 but is it normal?
P value is 0.0003, so it's quite obvious, isn't it?
It's not normal.

```
Call:
lm(formula = y ~ Rank + Average.Check, data = independence100)

Residuals:
       Min         1Q     Median         3Q        Max
-1.927e-10 -2.854e-11  3.309e-12  3.058e-11  1.245e-10

Coefficients:
              Estimate Std. Error   t value Pr(>|t|)
(Intercept)   8.115e-01  1.521e-11 5.336e+10   <2e-16 ***
Rank          1.270e-11  1.658e-13 7.662e+01   <2e-16 ***
Average.Check 1.478e-13  1.385e-13 1.067e+00    0.288
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.556e-11 on 97 degrees of freedom
Multiple R-squared:  0.9851,    Adjusted R-squared:  0.9848
F-statistic:  3212 on 2 and 97 DF,  p-value: < 2.2e-16
```

Followed by Shapiro test, delivering the verdict of model being not normal.

```
        Shapiro-Wilk normality test

data:  bcm2$residuals
W = 0.94457, p-value = 0.0003706
```

And even plots show the same evidence of model being not normal…



On to the next best model of Top250,
It looks like nowadays each model is demanding a boxcox transformation without which it's not going to give a satisfactory Adj.R square and p value.
Model after applying boxcox,

```
Call:
lm(formula = y ~ Rank + Units, data = top250_cleaned)

Residuals:
       Min         1Q     Median         3Q        Max
-6.377e-04 -2.336e-04 -3.259e-05  2.320e-04  7.738e-04

Coefficients:
             Estimate Std. Error  t value Pr(>|t|)
(Intercept) 1.338e+00  4.610e-05 29020.72  < 2e-16 ***
Rank        1.081e-04  2.998e-07   360.58  < 2e-16 ***
Units       2.625e-08  9.441e-09     2.78  0.00585 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0003053 on 247 degrees of freedom
Multiple R-squared:  0.9985,    Adjusted R-squared:  0.9985
F-statistic: 8.104e+04 on 2 and 247 DF,  p-value: < 2.2e-16
```

See that magic, Adj.R square is 0.9985...Looks like all variables included are highly significant.

What about the Normality you ask….

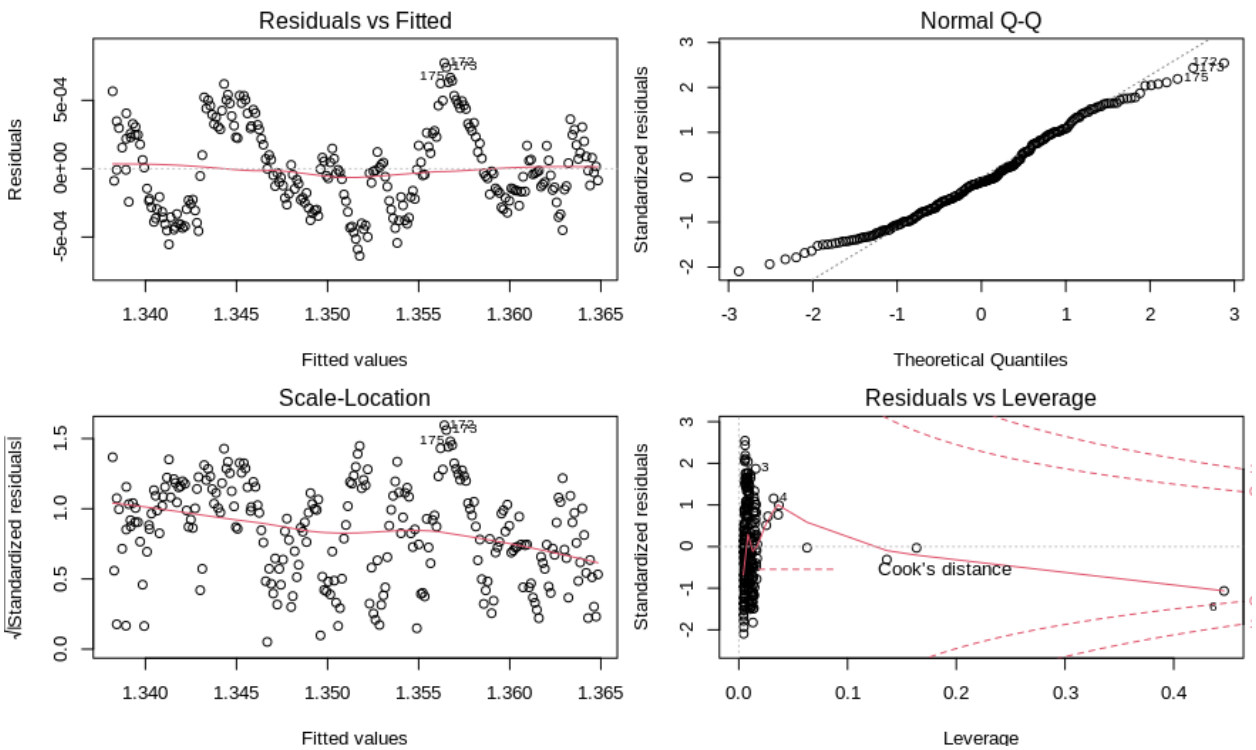
Coming to that…

```
        Shapiro-Wilk normality test

data:  bcm3$residuals
W = 0.97763, p-value = 0.000561
```

Do you see it?

P value is 0.0005.

The subsequent plots show the above result in graphs.



And to summarise all the models, the following table comes handy…

Models and their respective summary stats:

adj.r.squared <dbl>	sigma <dbl>	AIC <dbl>	BIC <dbl>
0.4491968	7.122175e+00	346.6740	361.9702
0.3844723	7.529016e+00	348.6765	356.3246
0.4585348	7.061544e+00	343.1912	352.7513
0.8452413	1.971064e+06	3188.5223	3201.5482
0.8398966	2.004810e+06	3189.9794	3197.7950
0.8466731	1.961924e+06	3186.6290	3197.0497
0.6219313	2.069188e+03	4533.8744	4555.0032
0.6208604	2.072116e+03	4532.6140	4546.6999
0.4589569	7.867665e-01	123.7424	133.3025
0.9848170	4.556447e-11	-4473.6368	-4463.2161
0.9984661	3.053232e-04	-3332.6187	-3318.5329

This brings us close to the linear modelling except for the predicting values, let's get that one done…

Predict and actual values for future50 model:

```
         1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18
predict 35 29 26 42 29 35 25 32 38 36 32 33 38 30 37 31 39 28
actual  48 27 21 47 23 28 24 28 39 44 39 20 42 25 44 49 39 24
        19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
predict 41 32 37 29 28 34 37 31 32 49 34 35 49 40 38 57 28 30
actual  45 32 49 30 25 39 48 22 40 44 24 38 44 42 41 47 20 28
        37 38 39 40
predict 25 21 41 36
actual  24 20 44 31
```

They seem pretty close…

Predict and actual values for indendence100:

```
                1        2        3        4        5        6
predict1 20852040 12695650 23244518 21964216 13149758 13955201
actual1  18601433 13100000 23800000 22181607 13177468 13987843
                7        8        9       10
predict1 11052834 20722898 21554678 11961576
actual1  12228168 18490719 18687601 12566618
```

Not very close, but satisfactory though…

Predict and actual values for top250:

```
           1    2   3   4    5   6   7   8   9   10
predict2  48 -312 916 465 8771 759 554 512 538 -109
actual2  142  129 410 340 5558 366 402 313 265  166
```

Alas, We have negative values too.

And Next modelling is,

Logistic Regression

Logistic regression is done when prediction of a categorical variable is to be done using all other xi's. Keeping the categorical variable as y. For example, the Default dataset of Library ISLR, here to predict whether a student is a defaulter or not, we use balance and all other xi's to predict yes or no for default.

Here also we can apply that, as we have Franchising as a categorical variable in future50.

```
##
## Call:
## glm(formula = Franchising ~ Rank + Sales + Units, family = "binomial",
##     data = train_lm)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.00972  -0.62056   0.07516   0.60701   1.97214
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.43522    1.72260   0.253   0.8005
## Rank         0.04968    0.03805   1.306   0.1917
## Sales       -0.18224    0.07654  -2.381   0.0173 *
## Units        0.16946    0.05920   2.863   0.0042 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 53.841  on 39  degrees of freedom
## Residual deviance: 31.523  on 36  degrees of freedom
## AIC: 39.523
##
## Number of Fisher Scoring iterations: 6
```

This follows the Pseudo R Square value using Mcfadden,

I.e.

```
##  McFadden
## 0.4145124
```

Even though A model is said to be the best fit model if mcfadden value is in the range of 0.2 and 04, here we got 0.41. Since it's very near to 0.4, this is so far the best model we have got.

Then, as to how much the model's Sensitivity, Specificity and Accuracy is, to answer that , we will look at them all. In this c is the cut.

```
##         Accuracy Sensitivity Specificity
## c = 0.3    0.775   0.8333333      0.6875
## c = 0.5    0.775   0.9583333      0.5000
## c = 0.6    0.750   0.7500000      0.7500
```

Sensitivity is the total coverage of True positive rate
Specificity is the total coverage of False Positive rate
Accuracy is the correctness of the model.

Best values computed using predict function are,

```
 Accuracy Sensitivity Specificity
0.7750000   0.8333333   0.6875000
```

Coming up next is, PCA

PCA(Principal Component Analysis)

Whenever we have innumerable rows in our dataset and we need to predict something in very less time, then PCA comes to our rescue, as this divides the dataset into components and then , out of all the components, we go for just one or two, which cover the most proportion of the data.

For PCA computation we need, a correlation matrix.

PCA for Future50

```
##                   Rank       Sales      Units Unit_Volume   yoy_sales
## Rank         1.0000000  0.18753491 -0.3581368   0.3597591 -0.73914085
## Sales        0.1875349  1.00000000  0.5041528  -0.1170478 -0.09259725
## Units       -0.3581368  0.50415278  1.0000000  -0.7129938  0.33189758
## Unit_Volume  0.3597591 -0.11704776 -0.7129938   1.0000000 -0.30166438
## yoy_sales   -0.7391409 -0.09259725  0.3318976  -0.3016644  1.00000000
## yoy_units   -0.7237336 -0.08528003  0.3063470  -0.2760399  0.90217091
##                yoy_units
## Rank        -0.72373357
## Sales       -0.08528003
## Units        0.30634699
## Unit_Volume -0.27603991
## yoy_sales    0.90217091
## yoy_units    1.00000000
```

Next, we need to compute eigenvalues and eigenvectors using eigen function.The following are the eigenvalues.

```
## [1] 3.04658890 1.64217982 0.76089282 0.29990812 0.15370232 0.09672802
```

Which follows the eigenvectors,

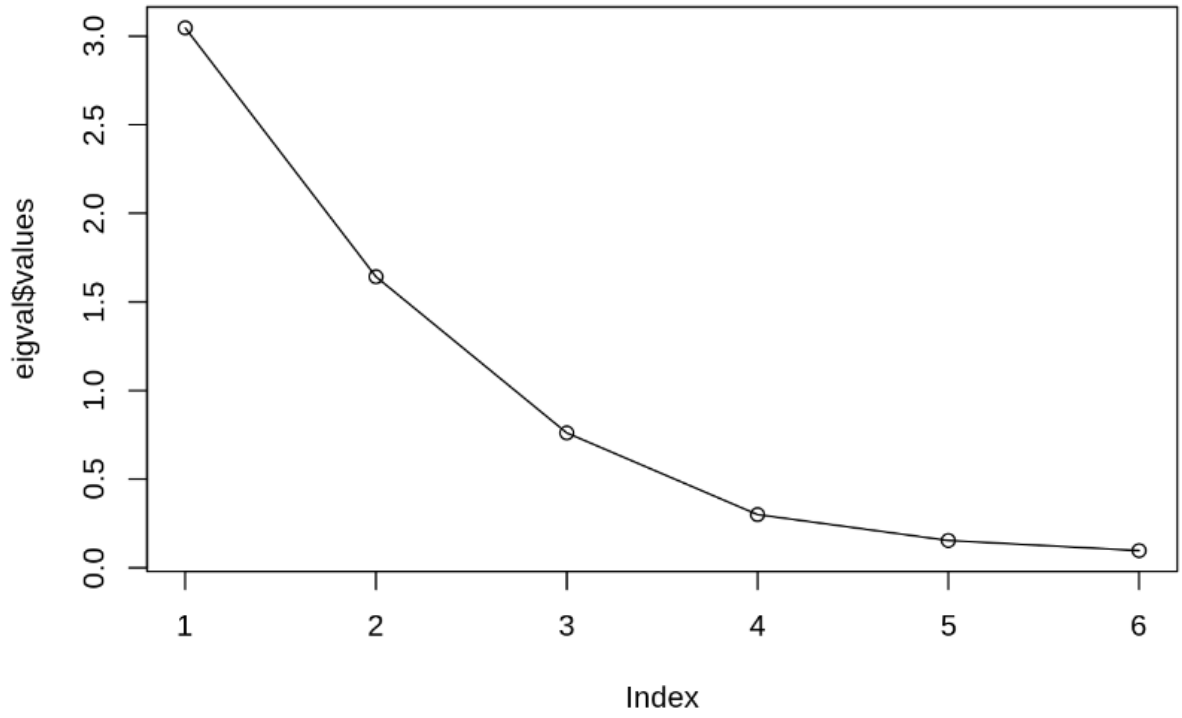
```
##             [,1]       [,2]        [,3]        [,4]        [,5]         [,6]
## [1,] -0.48320992 -0.2230331 -0.05312747  0.80641744 -0.25215264 -0.007199895
## [2,]  0.02399057 -0.6282278 -0.64997883 -0.07157454  0.41891353 -0.040870512
## [3,]  0.37092544 -0.5443919  0.08204651 -0.15044707 -0.72961465  0.065941414
## [4,] -0.35686473  0.3755301 -0.64788116 -0.29966913 -0.47076956  0.021192309
## [5,]  0.50486794  0.2380358 -0.25331529  0.32014175 -0.08031881 -0.717869641
## [6,]  0.49608756  0.2480718 -0.28992193  0.36005744  0.02275675  0.691479253
```

Now, we find PCAs using the above information.

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5     PC6
## Standard deviation     1.7454 1.2815 0.8723 0.54764 0.39205 0.31101
## Proportion of Variance 0.5078 0.2737 0.1268 0.04998 0.02562 0.01612
## Cumulative Proportion  0.5078 0.7815 0.9083 0.95826 0.98388 1.00000
```

Here, It can be observed that the first 3 components have a cumulative proportion of 90 %.

It can be viewed even in a plot.



Wherever the second bend comes, till there we can consider the components.

PCA for Independence100

Correlation plot of Independence100

```
##                    Rank      Sales Average.Check Meals.Served
## Rank          1.0000000 -0.9173406    -0.3059965   -0.2474337
## Sales        -0.9173406  1.0000000     0.1941948    0.3020141
## Average.Check -0.3059965  0.1941948     1.0000000   -0.6272059
## Meals.Served  -0.2474337  0.3020141    -0.6272059    1.0000000
```

Eigenvalues

```
## [1] 2.08002409 1.63054153 0.21652521 0.07290916
```

Eigenvectors

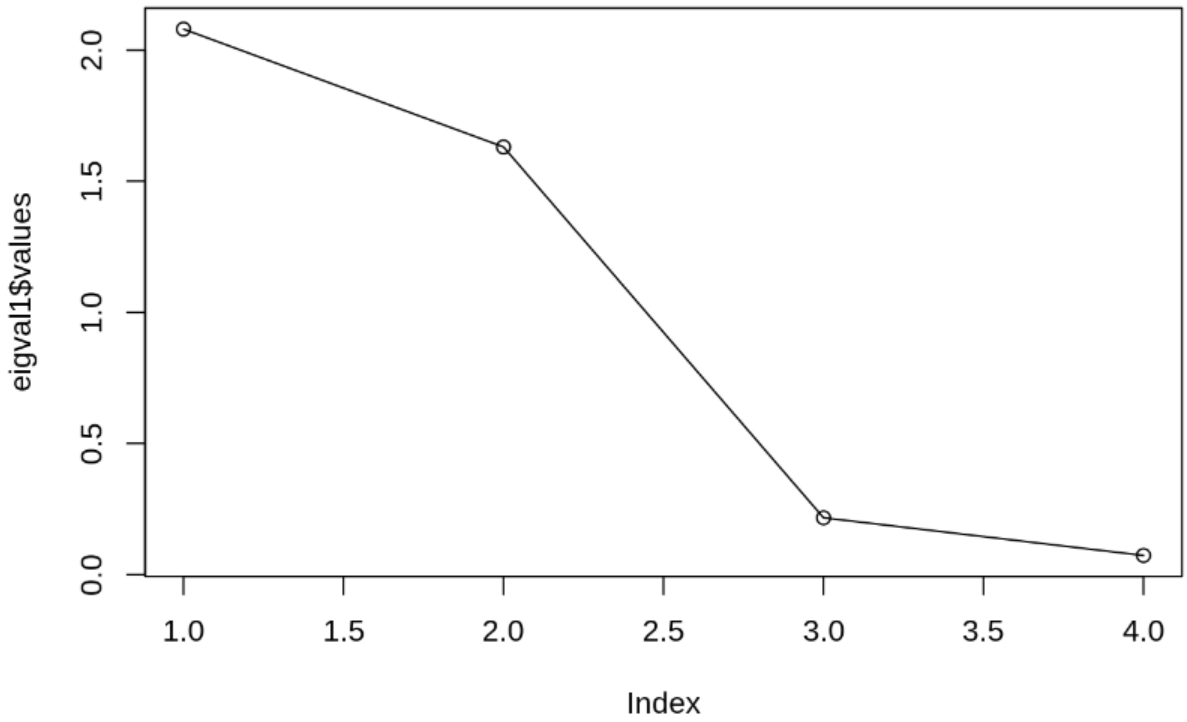
```
##            [,1]         [,2]        [,3]        [,4]
## [1,]  0.6764395  0.071604672  0.07021927 -0.72963802
## [2,] -0.6735302  0.004737486 -0.33938906 -0.65661991
## [3,] -0.1710770 -0.719947156  0.65166910 -0.16654164
## [4,] -0.2439659  0.690308930  0.67469378 -0.09350156
```

PCAs

```
## Importance of components:
##                           PC1    PC2     PC3     PC4
## Standard deviation     1.442 1.2769 0.46532 0.27002
## Proportion of Variance 0.520 0.4076 0.05413 0.01823
## Cumulative Proportion  0.520 0.9276 0.98177 1.00000
```

Here the cumulative proportion of PC1 and PC2 are constituting 92%.
Meaning which, we can just consider PC1 and PC2.
So, the new variables are PC1 and PC2.

Scree plot



Inferences:

- PCAs are new variables created out of original variables. And the maximum PCAs are equal to the original variables. Ideally the number of PCAs should be less than the original variables.
- Here we have taken 3 PCAs instead of 6 original variables in future50
- And 2 PCAs instead of 4 in Independence100.

KNN

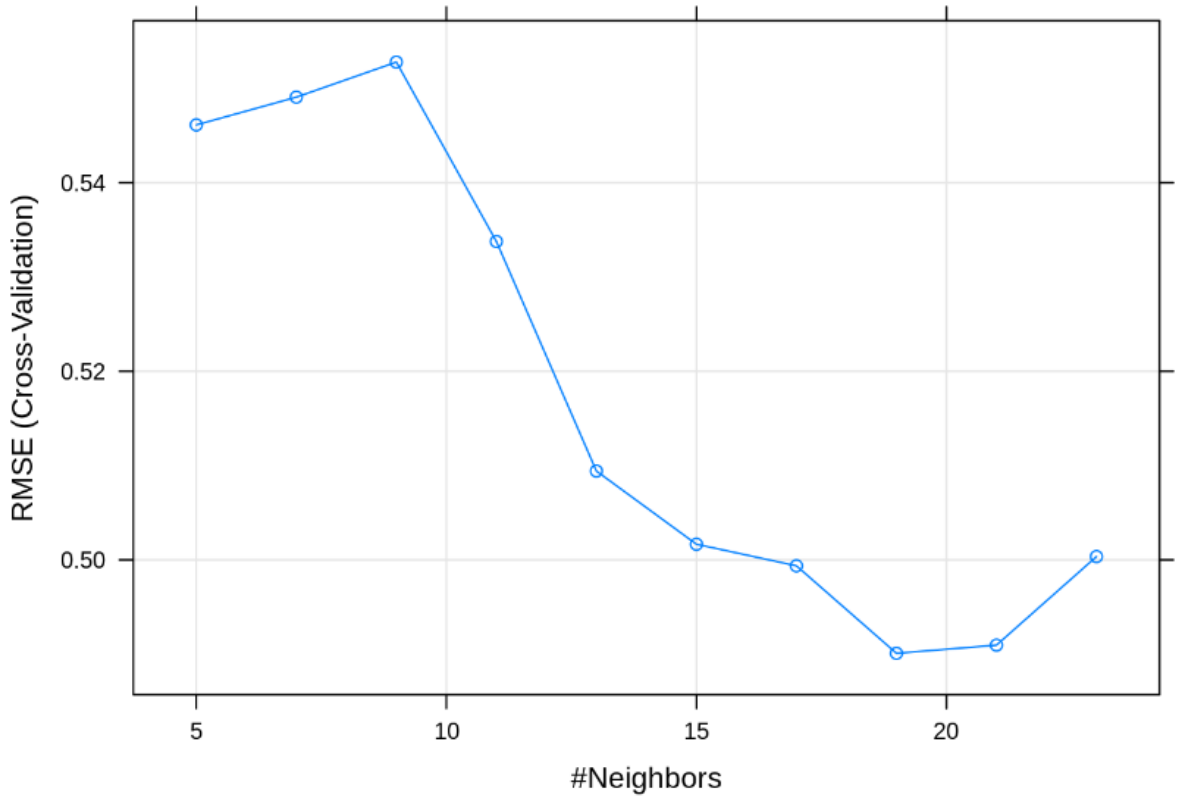
KNN is a supervised machine learning algorithm. KNN abbreviates to K-nearest neighbours. This is mostly used in classification problems.

Since we have a categorical variable in future50. We apply KNN to that dataset.

Output after applying KNN,

```
## k-Nearest Neighbors
##
## 40 samples
## 11 predictors
##
## Pre-processing: centered (148), scaled (148)
## Resampling: Cross-Validated (3 fold)
## Summary of sample sizes: 27, 27, 26
## Resampling results across tuning parameters:
##
##   k   RMSE       Rsquared   MAE
##    5  0.5461381  0.1309704  0.5089744
##    7  0.5490820  0.3156496  0.5193616
##    9  0.5528007  0.2451899  0.5183150
##   11  0.5337734  0.3554300  0.5051060
##   13  0.5094205  0.3330608  0.4912651
##   15  0.5016573  0.3253490  0.4916667
##   17  0.4993658  0.2340080  0.4914889
##   19  0.4900884  0.3558300  0.4829381
##   21  0.4909547  0.2511319  0.4832548
##   23  0.5003489  0.1358923  0.4911610
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was k = 19.
```

Here the optimal model is obtained when Root Mean Square Error is the least.
So the final value used for the model is k = 19, here accuracy is 0.48.
Plotting this model results in…



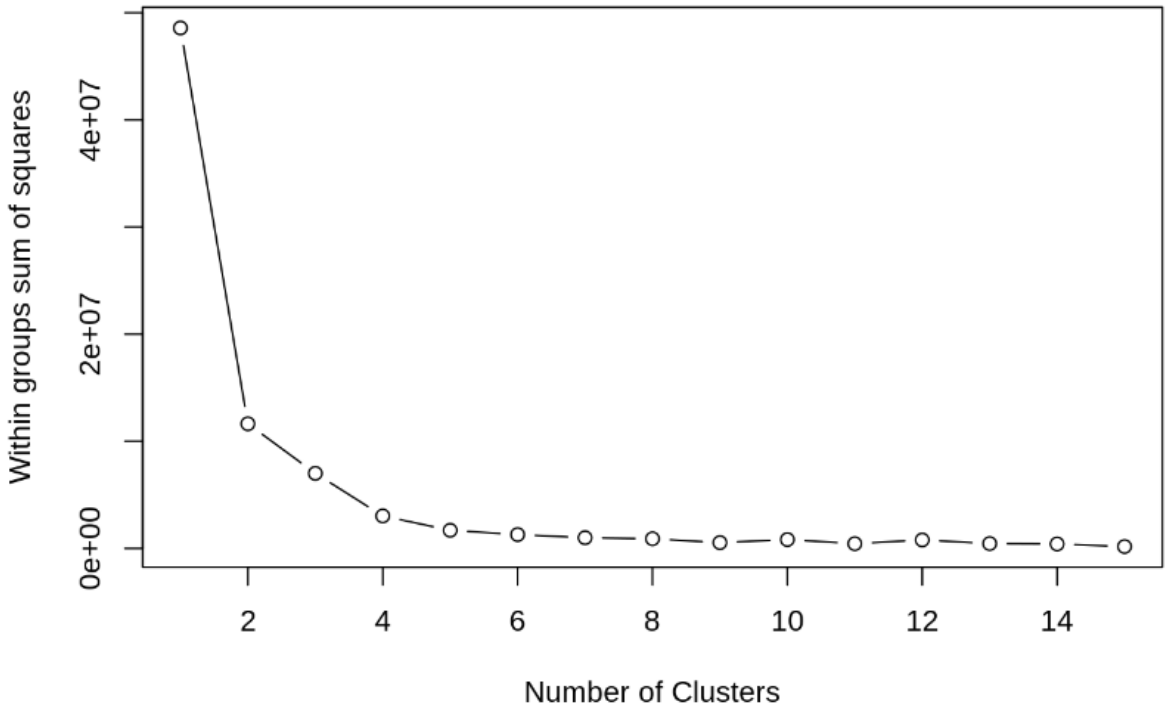
The above plot shows the variations of RMSE with respect to the K value.It has floor value at Neighbour 19.

Inferences:

- Cross validation is used to estimate the test error associated with the model's performance.
- Here we apply the cv method, classification of the test set where for each row k nearest neighbours of the training set are found.

Clustering

Clustering is the task of dividing the data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups.



Applying clustering to the future50,
We get the above plot.
For Sales and Rank.

```
## K-means clustering with 3 clusters of sizes 17, 22, 11
##
## Cluster means:
##       Rank    Sales    Units Unit_Volume yoy_sales yoy_units
## 1 18.70588 34.23529 58.35294    732.0588  40.02353  32.37647
## 2 26.09091 33.86364 28.00000   1452.2727  34.60909  29.38182
## 3 34.81818 32.90909 11.54545   3203.1818  22.10000  15.95455
##
## Clustering vector:
##  [1] 2 1 2 1 2 1 1 2 1 2 1 2 1 2 2 1 1 1 3 2 3 1 2 1 3 3 2 1 2 2 1 2 1 3 1 2 2 3
## [39] 3 2 3 1 3 2 2 2 2 3 3 2
##
## Within cluster sum of squares by cluster:
## [1]  549461.3 2159734.0 4292654.2
##  (between_SS / total_SS =  85.6 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

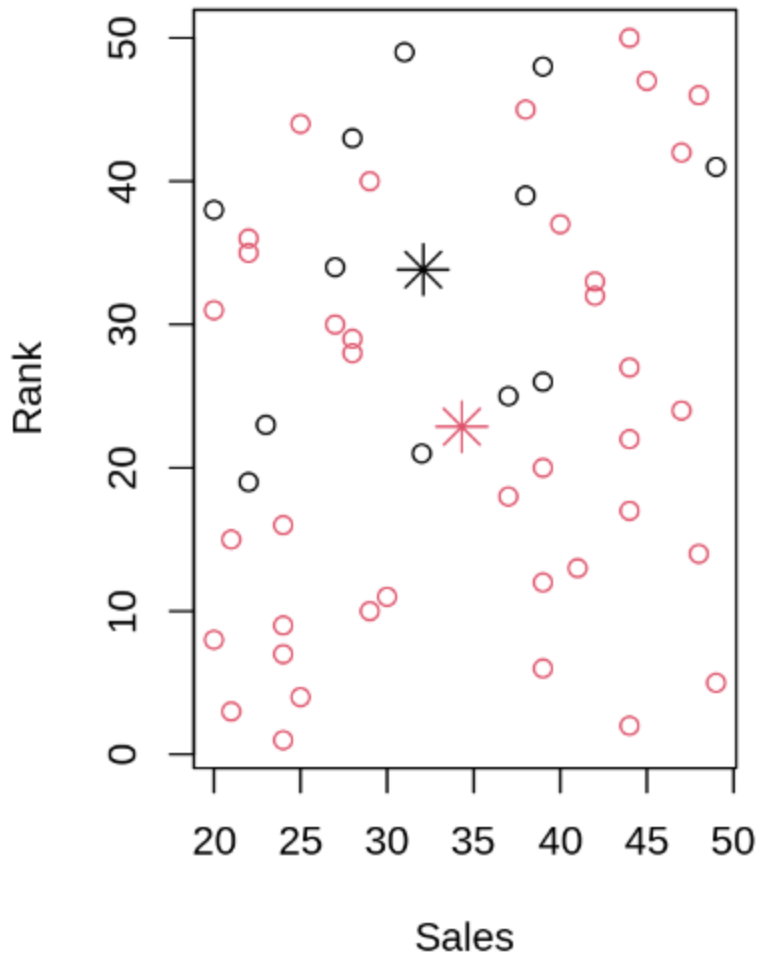
The above is the clustering a:vector and and various cluster means.

Initial division is 17,22,11.

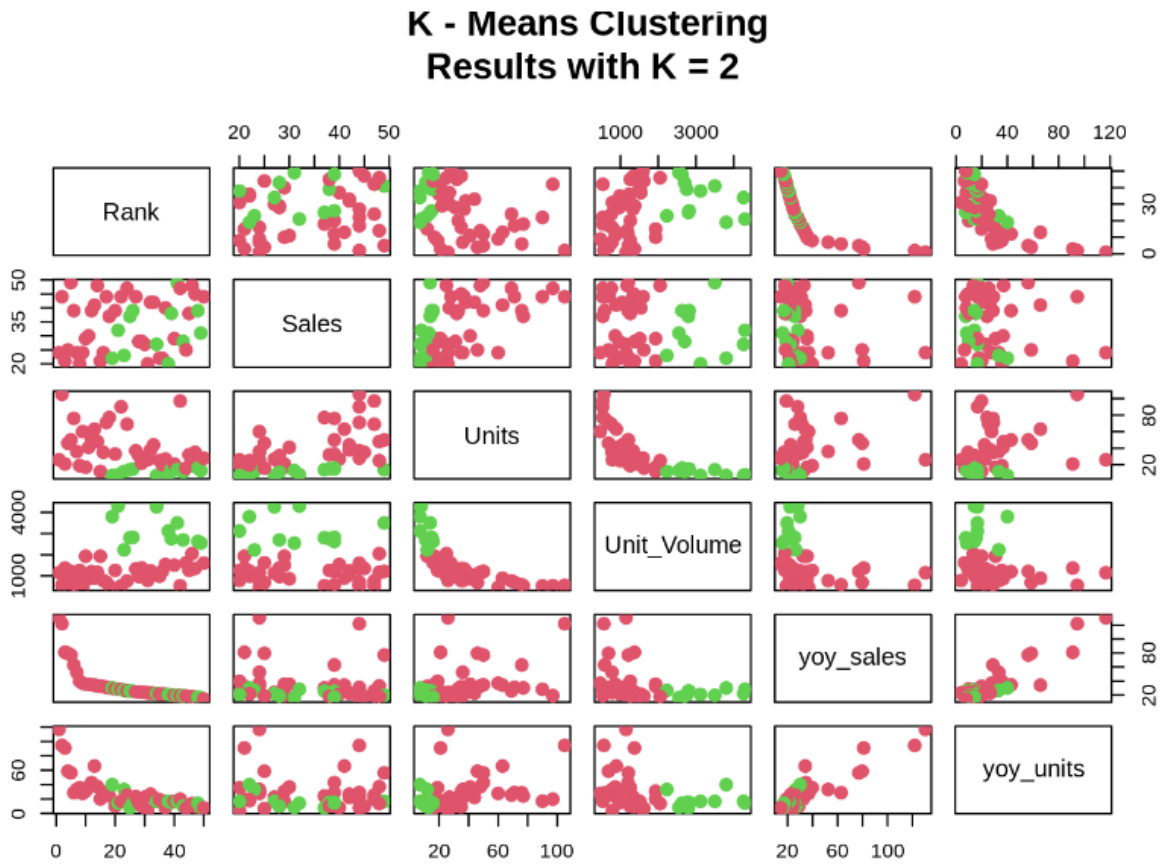
```
## K-means clustering with 2 clusters of sizes 12, 38
##
## Cluster means:
##       Rank    Sales    Units Unit_Volume yoy_sales yoy_units
## 1 33.83333 32.08333 11.58333    3122.083  22.46667  17.40000
## 2 22.86842 34.31579 42.00000    1109.605  37.24474  30.61842
##
## Clustering vector:
##  [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 1 2 1 1 2 2 2 2 2 2 2 1 2 2 2 1
## [39] 1 2 1 2 1 2 2 2 2 1 1 2
##
## Within cluster sum of squares by cluster:
## [1] 5161325 6468980
##  (between_SS / total_SS =  76.1 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

Next clustering is with two clusters of sizes 12 and 18.

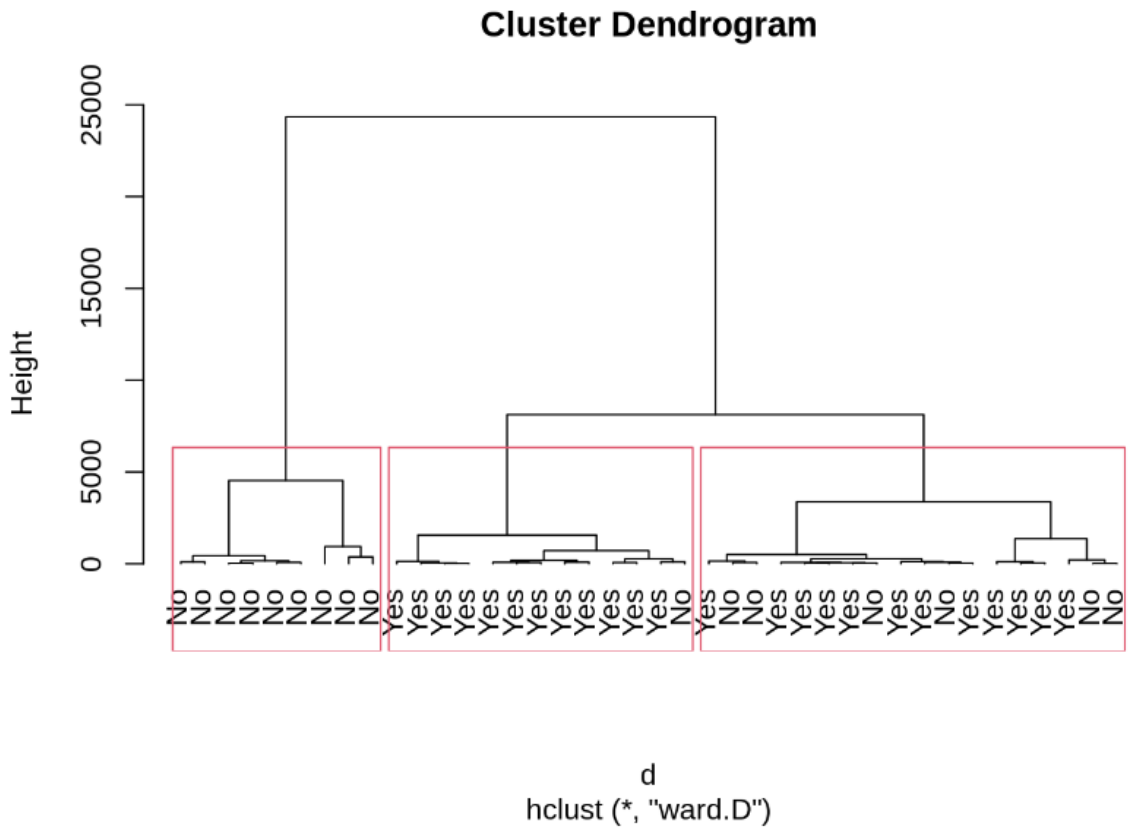
K means cluster for sales and Rank



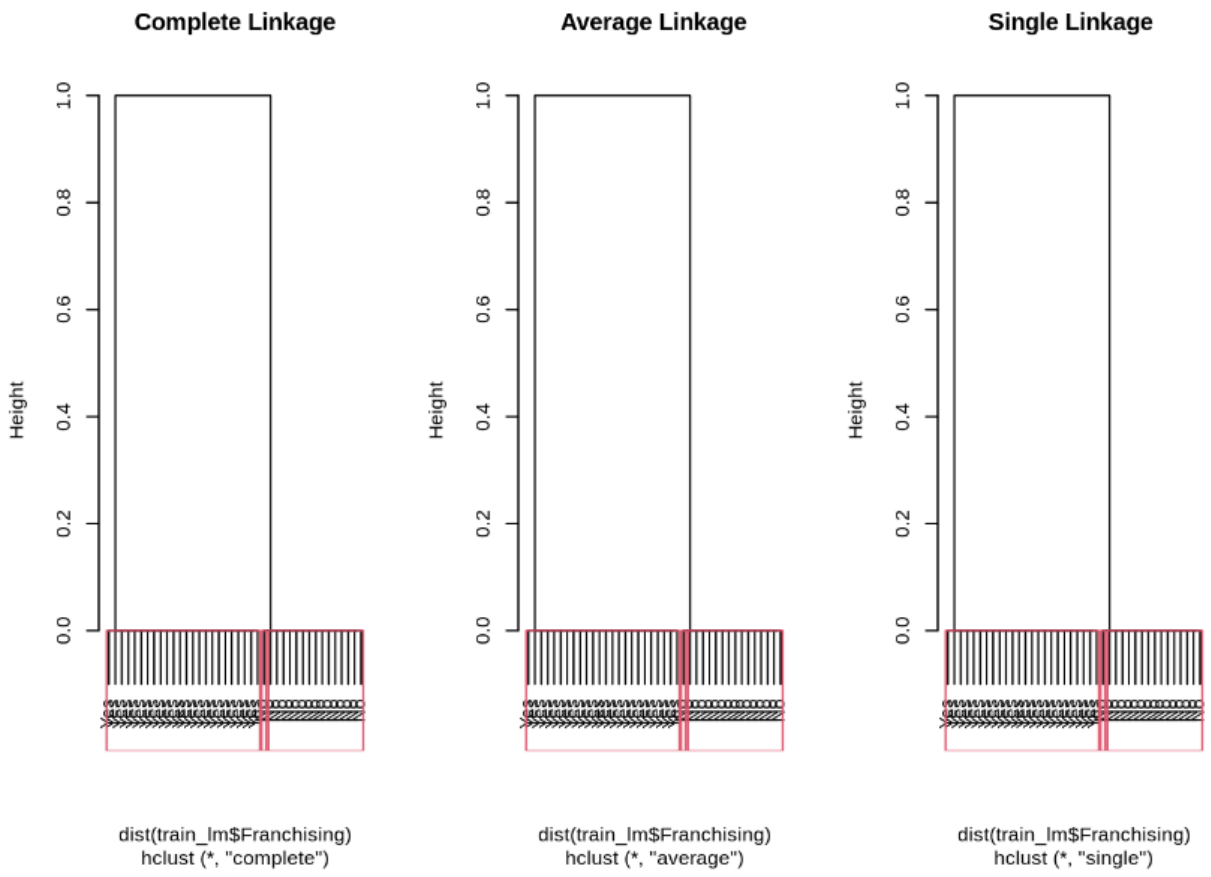
Plot after clustering:



Partitioning the whole data into clusters with k=2.
Here green colour data points belong to one cluster and pink color data points belong to the other cluster.



The above is the dendrogram showing the clusters at each k value and reaching all the to the last k value.



The above dendrograms show the clustering using the three methods, Complete linkage, Average linkage, And Single linkage.

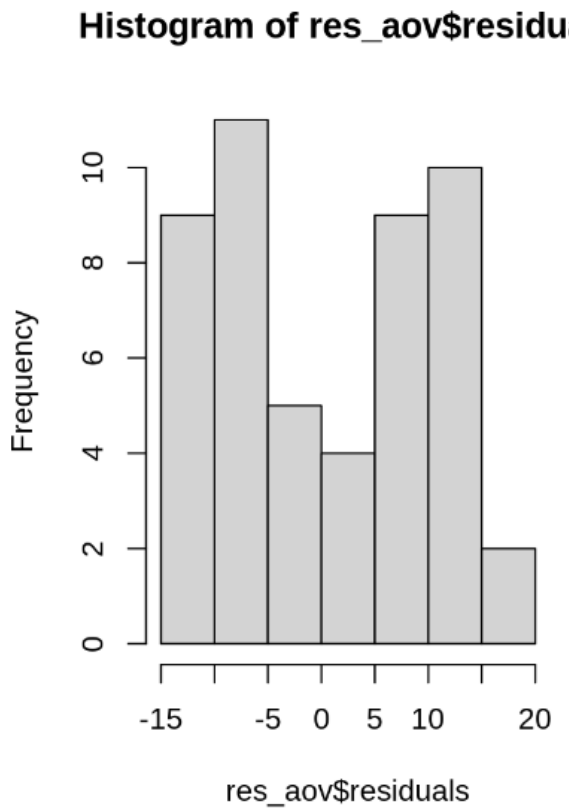
ANOVA

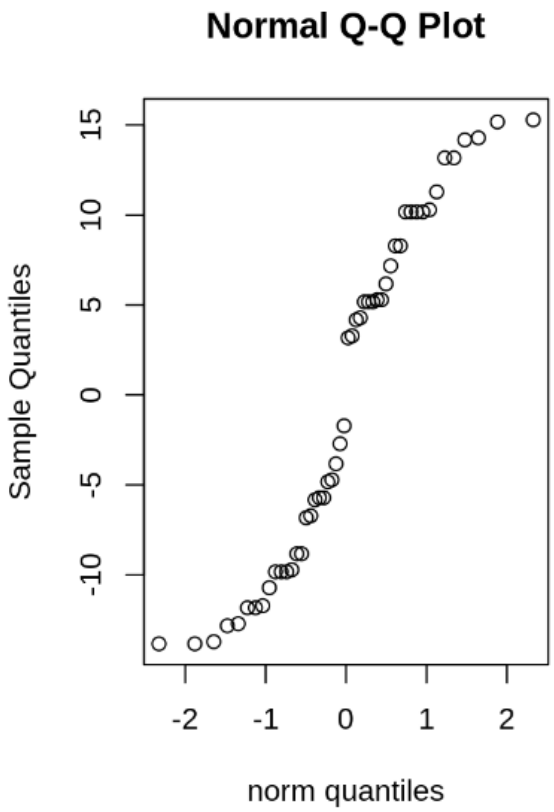
Analysis of variance is done to compare between two groups or to compare within the group.Here we use that to the variance of Sales with respect to Franchising.

Testing Hypothesis for Levene's Test:

$H_0:$ Variances are equal
$H_1:$ At least one variance is different

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## Franchising   1      0    0.16   0.002  0.968
## Residuals    48   4512   94.01
```





Residuals are normally distributed and p value is significantly high when applied to the levene test.

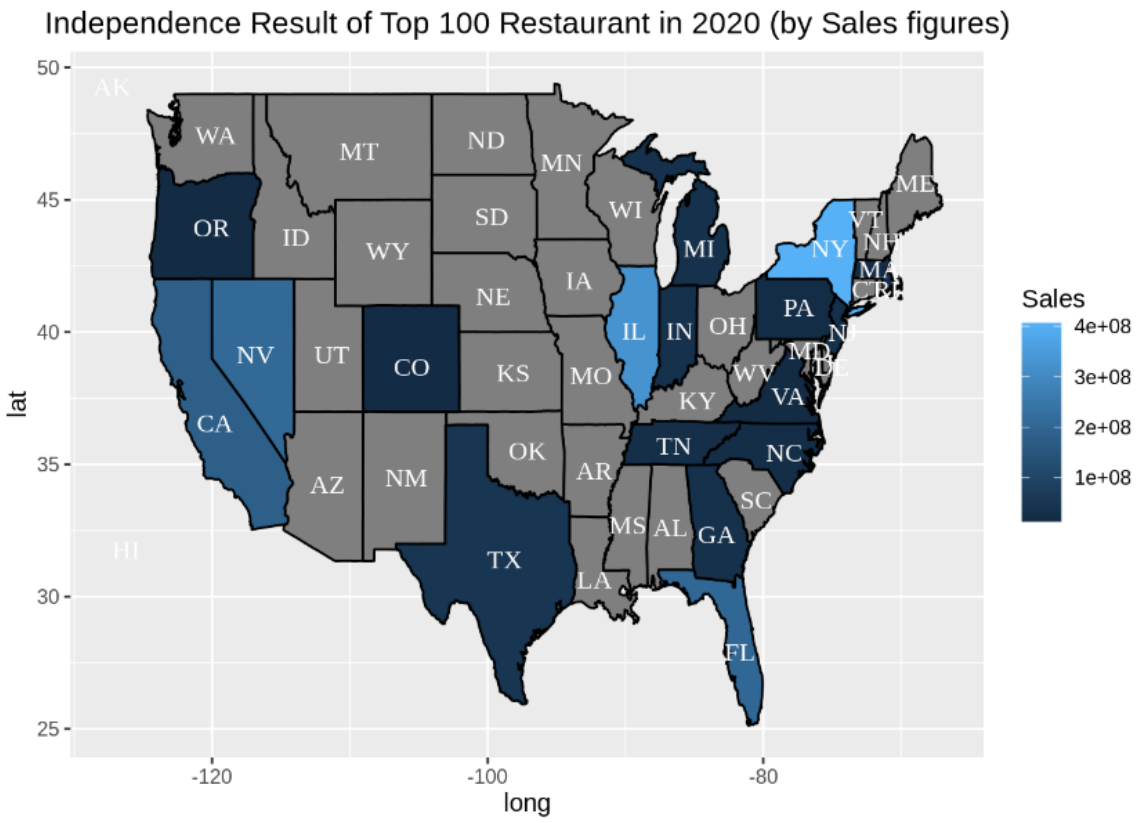
```
##
##  Shapiro-Wilk normality test
##
## data:  res_aov$residuals
## W = 0.91268, p-value = 0.001295
```

After applying the shapiro test for the residuals , p value turns out to be 0.001.

We accept our null hypothesis that the variances are equal for the yes or no groups of Franchising with respect to the sales.

Last but not least:

One beautiful part of EDA that has been left out intentionally…



This can be obtained using the maps library.
Proper sales figures in each state where restaurants are located.

Conclusion:

- For a restaurant to be placed in Future50 or Independence100 or Top250, It needs to have the highest figures in all categories mentioned.
- Franchising has more effect than a non chain restaurants, as sales are more and even the Average.Check.
- Reviews play an important role in sales, since they cannot be quantified, proper view cannot be given about this.