Retail Sales Prediction

Individual Capstone Project - 2

Presented By:
Prasanna R Kadlayyanavaramath

Outline:

1. Problem Statement
2. Data Summary
3. Data Preparation
4. EDA
5. Feature Engineering
6. Model Training
7. ML Model Building
8. Hyper parameter Tuning

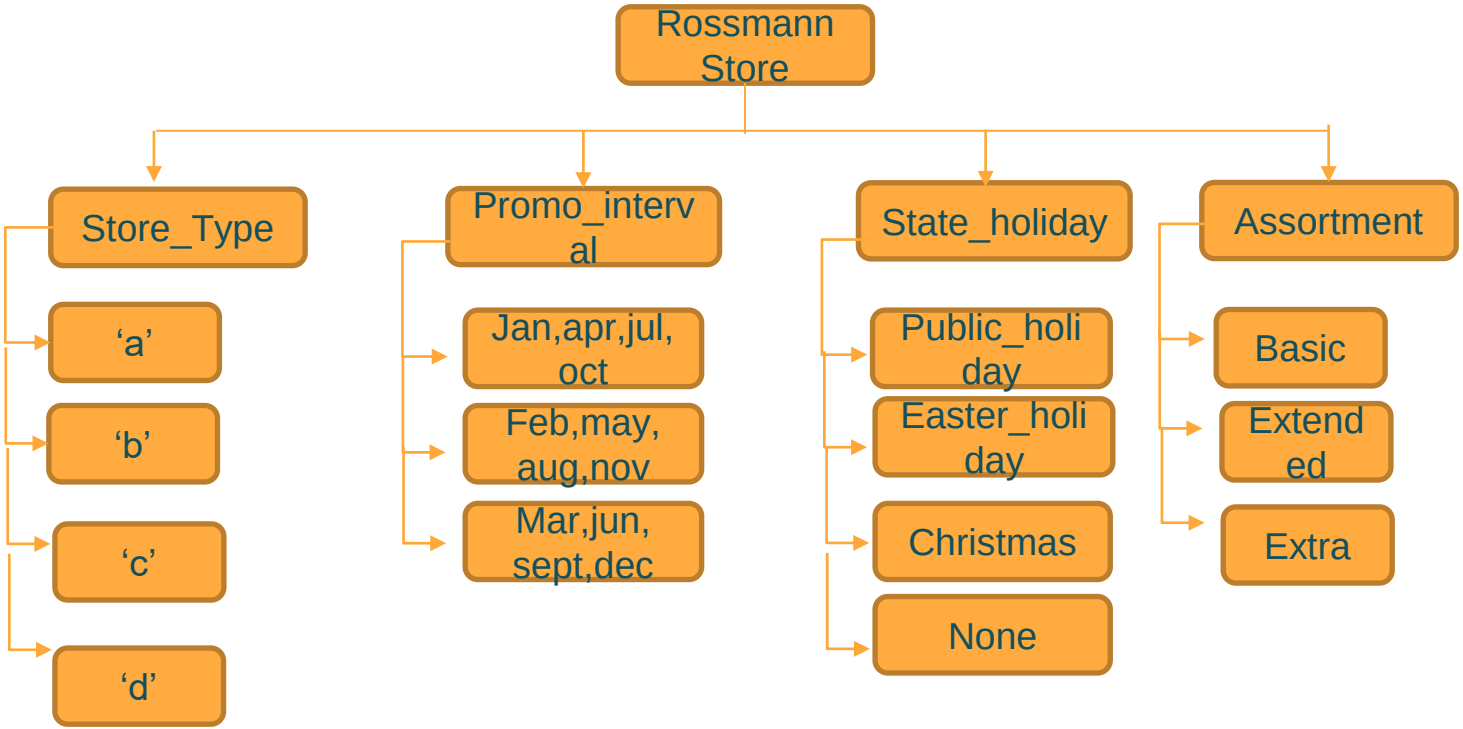
Problem Statement:

Sales forecasting has always been a very significant area to concentrate upon. An efficient and optimal way of forecasting has become essential for all the vendors in order to sustain the efficacy of the marketing organizations. Manual infestation of this task could lead to drastic errors leading to poor management of the organization, and most importantly would be time consuming, which is something not desirable in this expedited world. This is where machine learning can be exploited in a great way.

We are provided with historical sales data for 1,115 Rossmann stores. The task is to forecast the "Sales" column for the test set. Note that some stores in the dataset were temporarily closed for refurbishment. So,

we have proposed the machine learning algorithms towards the data collected from the previous sales of a Rossmann store.

Data Summary:



Dataset Details:

- **Id** – an Id that represents a Store
- **Store** – a unique Id for each store
- **Sales** – the turnover for any given day (this is what we are predicting)
- **Customers** – the number of customers on a given day
- **Open** – an indicator for whether the store was open: 0 = closed, 1 = open
- **StateHoliday** – indicates a state holiday. Normally all stores, with few exceptions, are closed on state holidays. Note that all schools are closed on public holidays and weekends. a = public holiday, b = Easter holiday, c = Christmas, 0 = None
- **SchoolHoliday** – indicates if the (Store) was affected by the closure of public schools
- **StoreType** – differentiates between 4 different store models: a, b, c, d
- **Assortment** – describes an assortment level: a = basic, b = extra, c = extended
- **CompetitionDistance** – distance in meters to the nearest competitor store
- **CompetitionOpenSince[Month/Year]** – gives the approximate year and month of the time the nearest competitor was opened
- **Promo** – indicates whether a store is running a promo on that day
- **Promo2** – Promo2 is a continuing and consecutive promotion for some stores: 0 = store is not participating, 1 = store is participating
- **Promo2Since[Year/Week]** – describes the year and calendar week when the store started participating in Promo2
- **PromoInterval** – describes the consecutive intervals Promo2 is started, naming the months the promotion is started anew. E.g. “Feb,May,Aug,Nov” means each round starts in February, May, August, November of any given year for that store.

Data Reading and Data cleaning:

Data cleaning is the very first important fundamental thing which every data scientist must know. It is the process of finding the inaccurate, incorrect and irrelevant or missing part of a data and then, modifying the data according to our necessity.

❖ Initially merge the training and testing dataset into store data.

- **Steps:**
- 1.Remove duplicate rows.
- 2.Removing NULL values by replacing mean, mode and zero.
- 3.Converting datatypes.
- 4.Adding new columns if necessary.

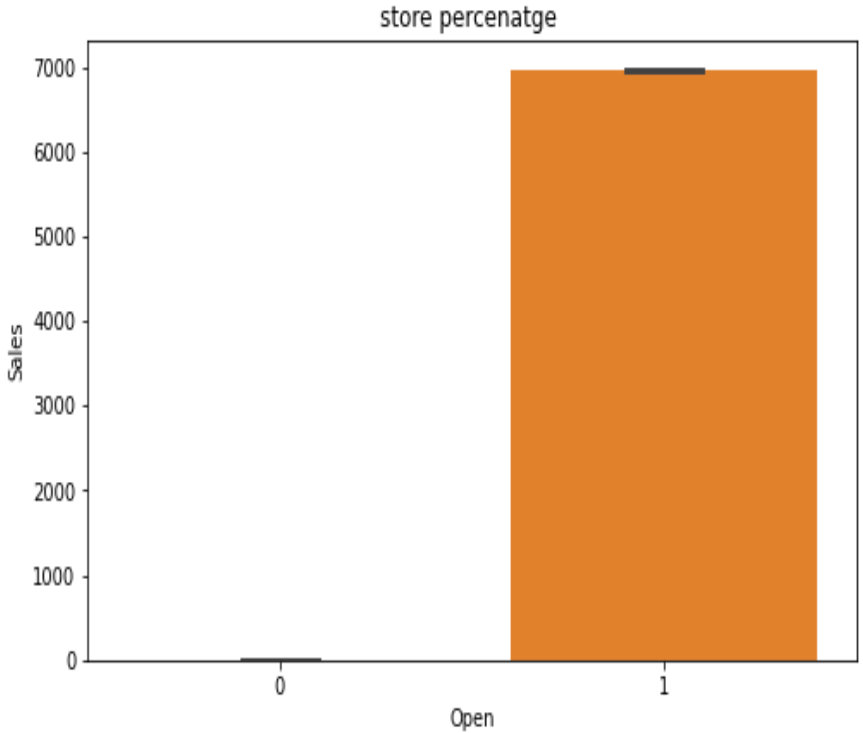
Data Visualization:

Mainly performed using Matplotlib and Seaborn library and the following graph and plots had been used:

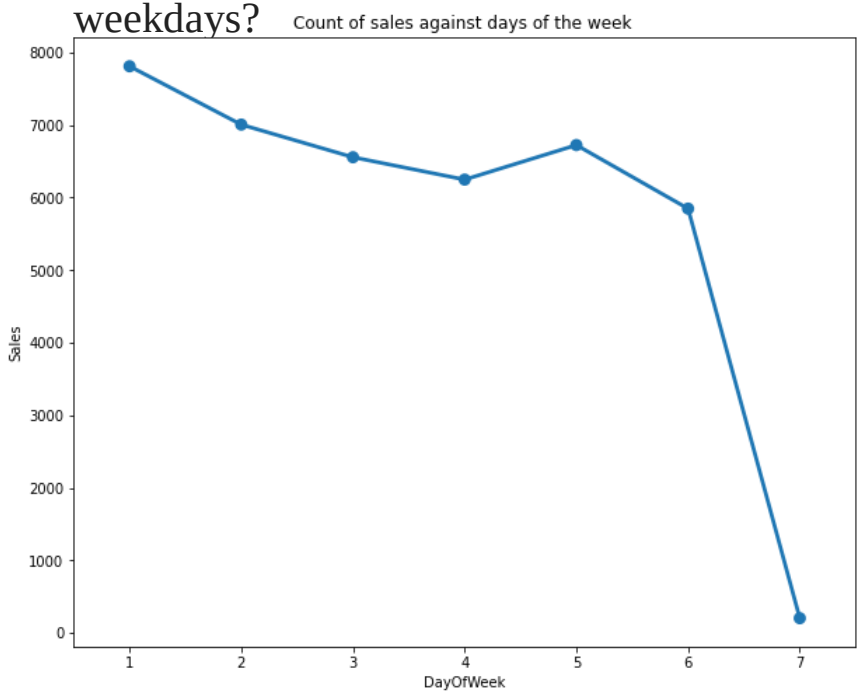
1) Bar Plot.
2) Histogram.
3) Scatter Plot.
4) Pie Chart.
5) Line Plot.
6) Heatmap.
7) Box Plot

Exploratory Data Analysis(EDA):

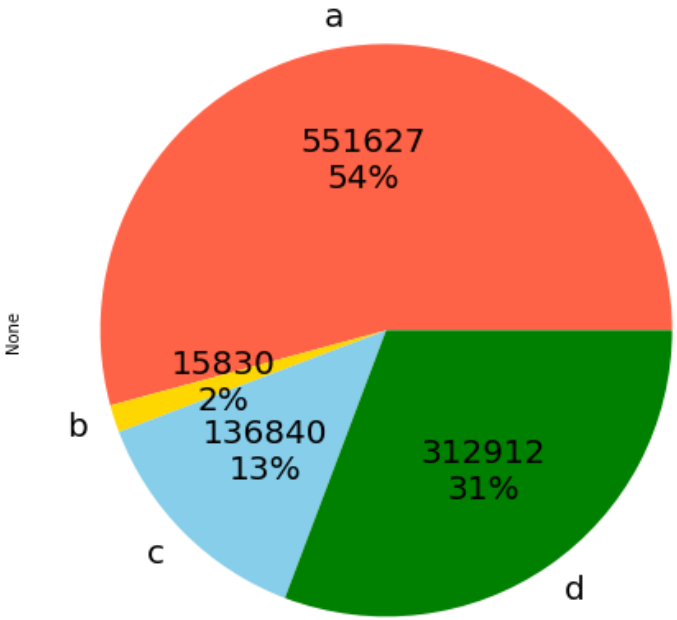
How 'open' Variable impacted the 'sales'?



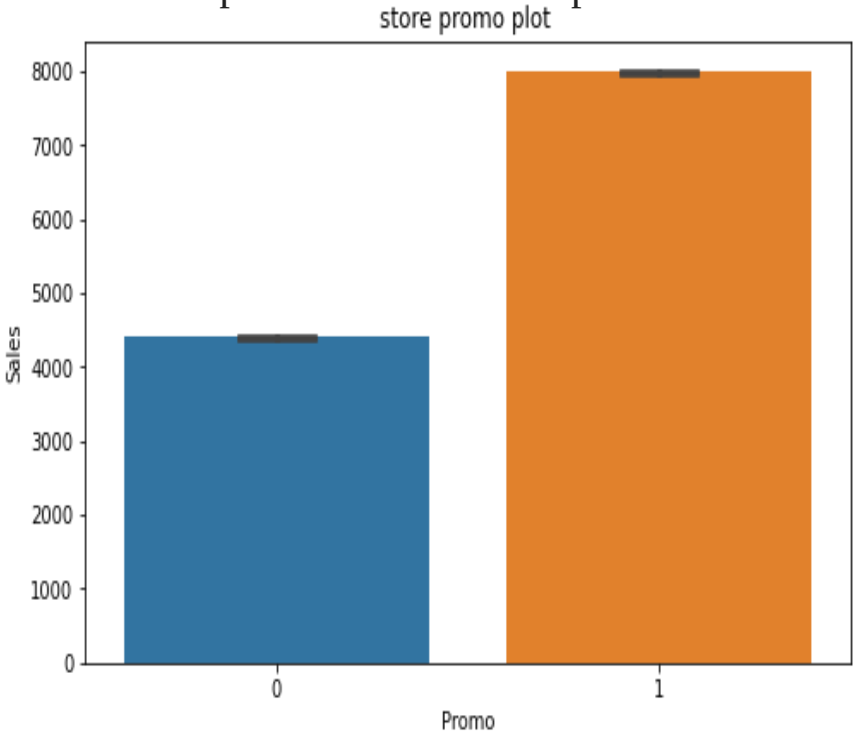
What is the total number of sales over the weekdays?



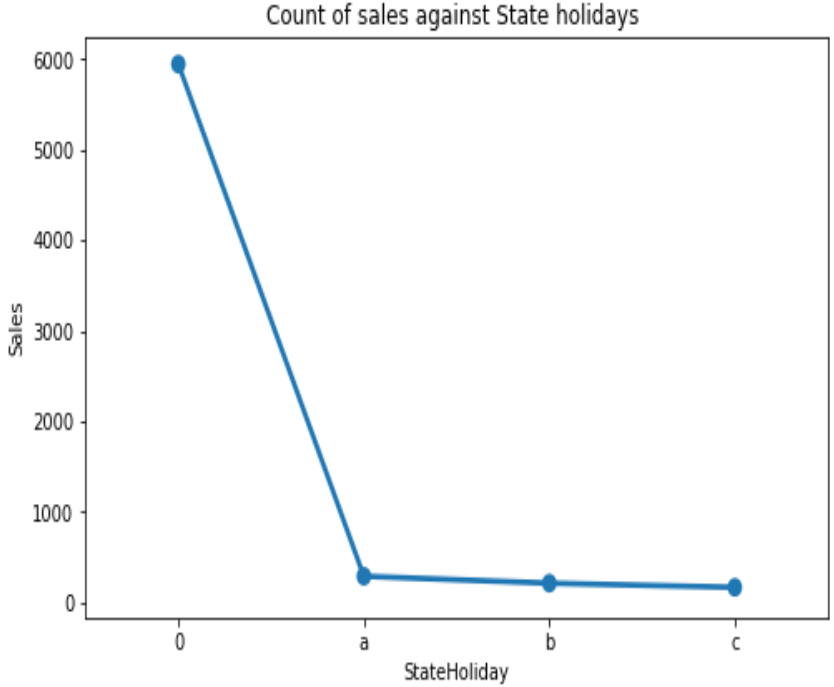
Which store type sales highest?



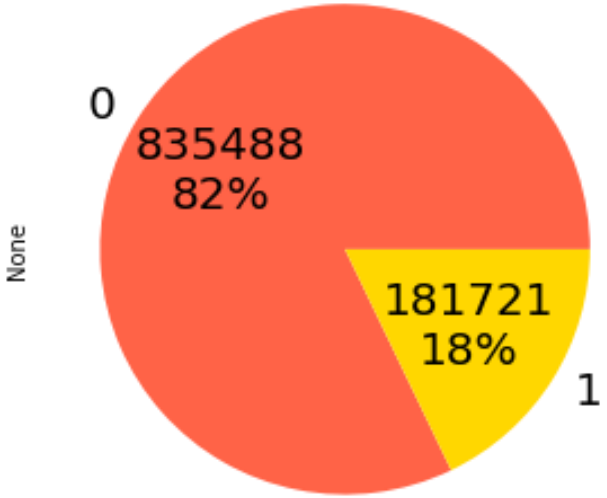
Relationship between sales and promotions



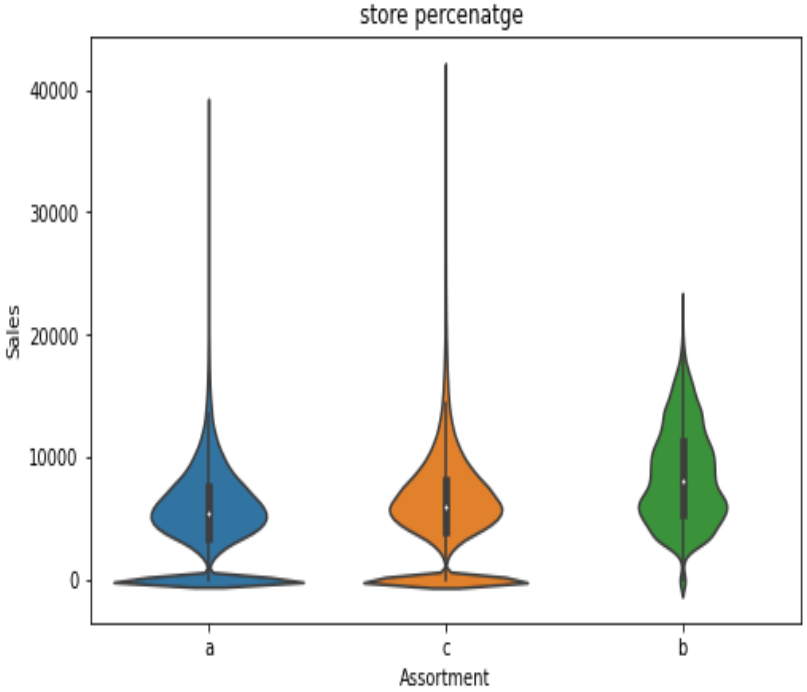
What is the count of sales against state holidays?



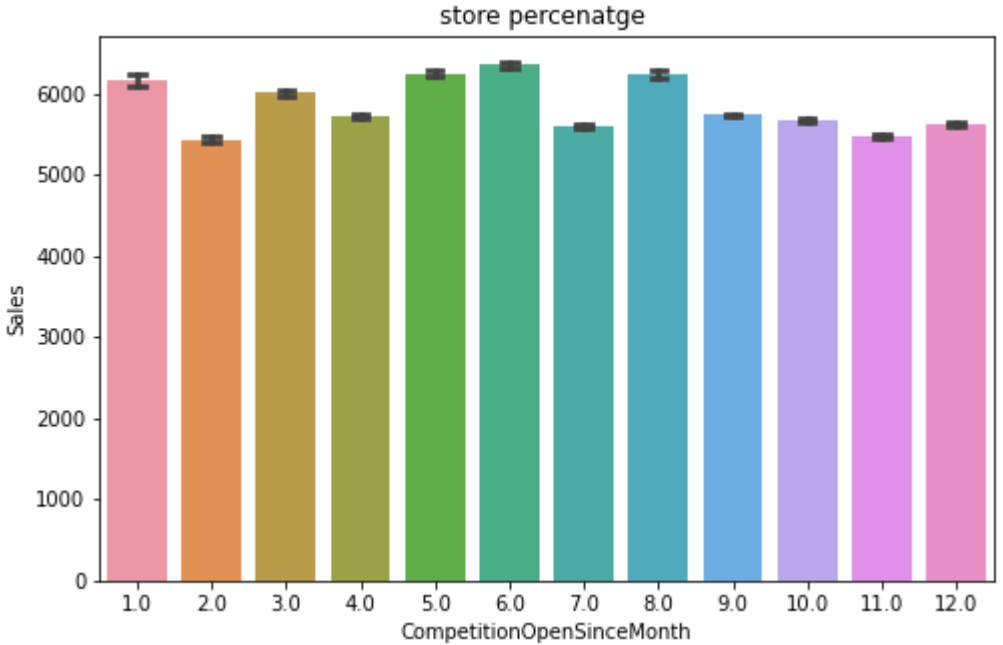
What is the count of sales against state holidays?



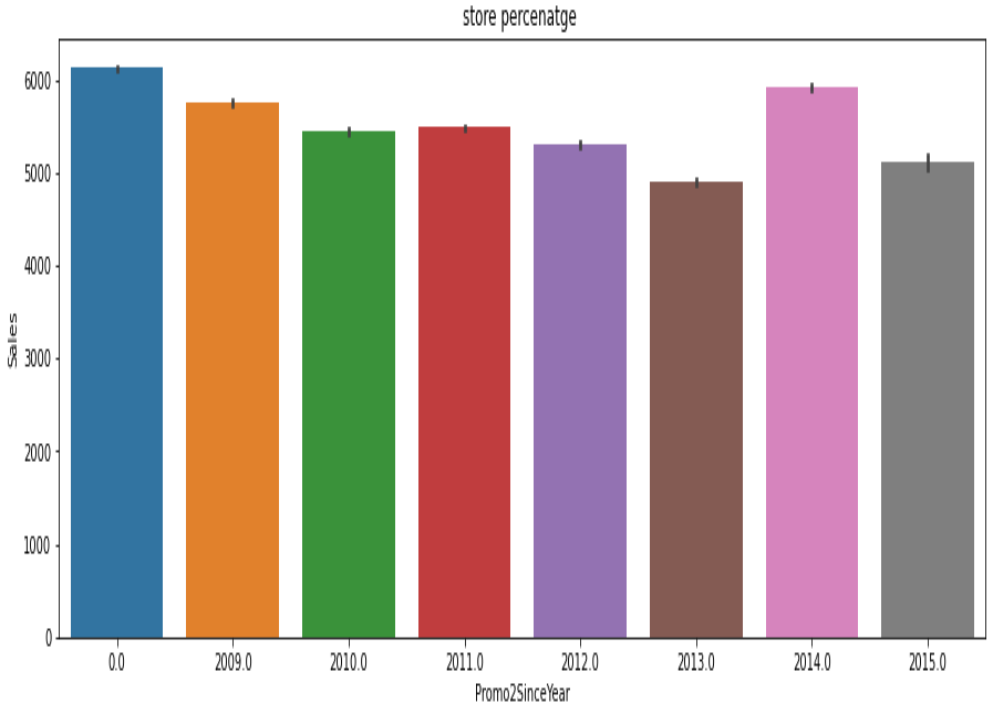
Assortment v/s sales



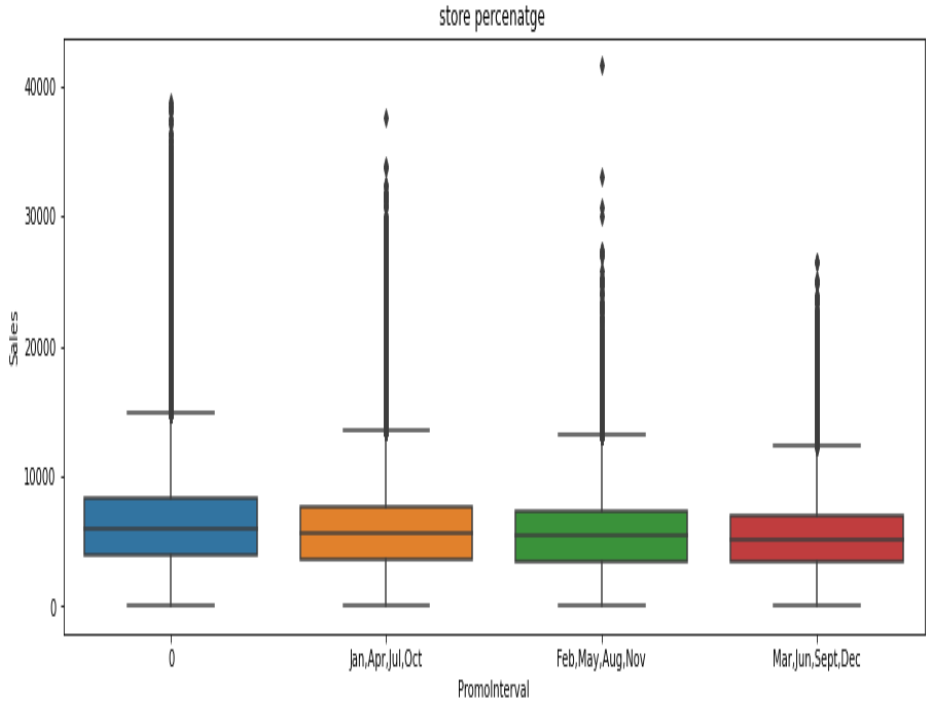
Relationship between sales and CompetitionOpenSinceMonth



In which promoyear sales sells more?

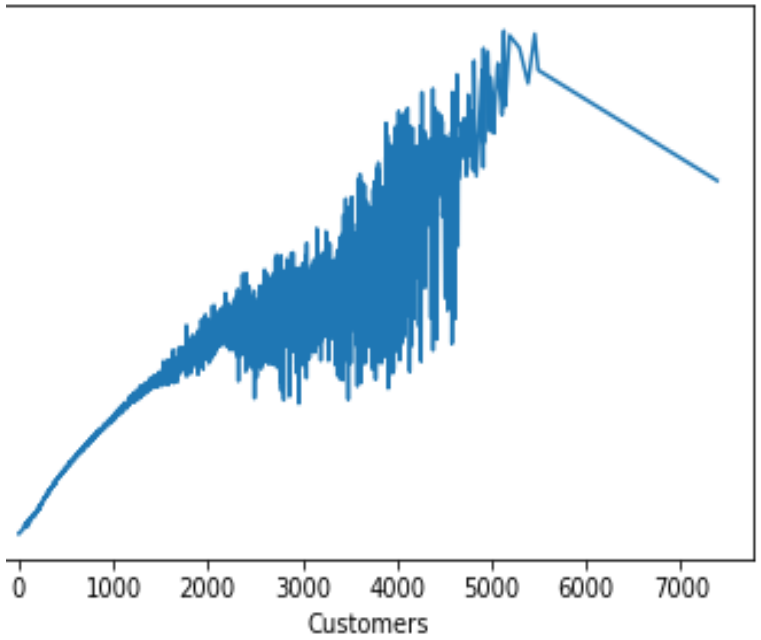


In which months sales reached peak level?

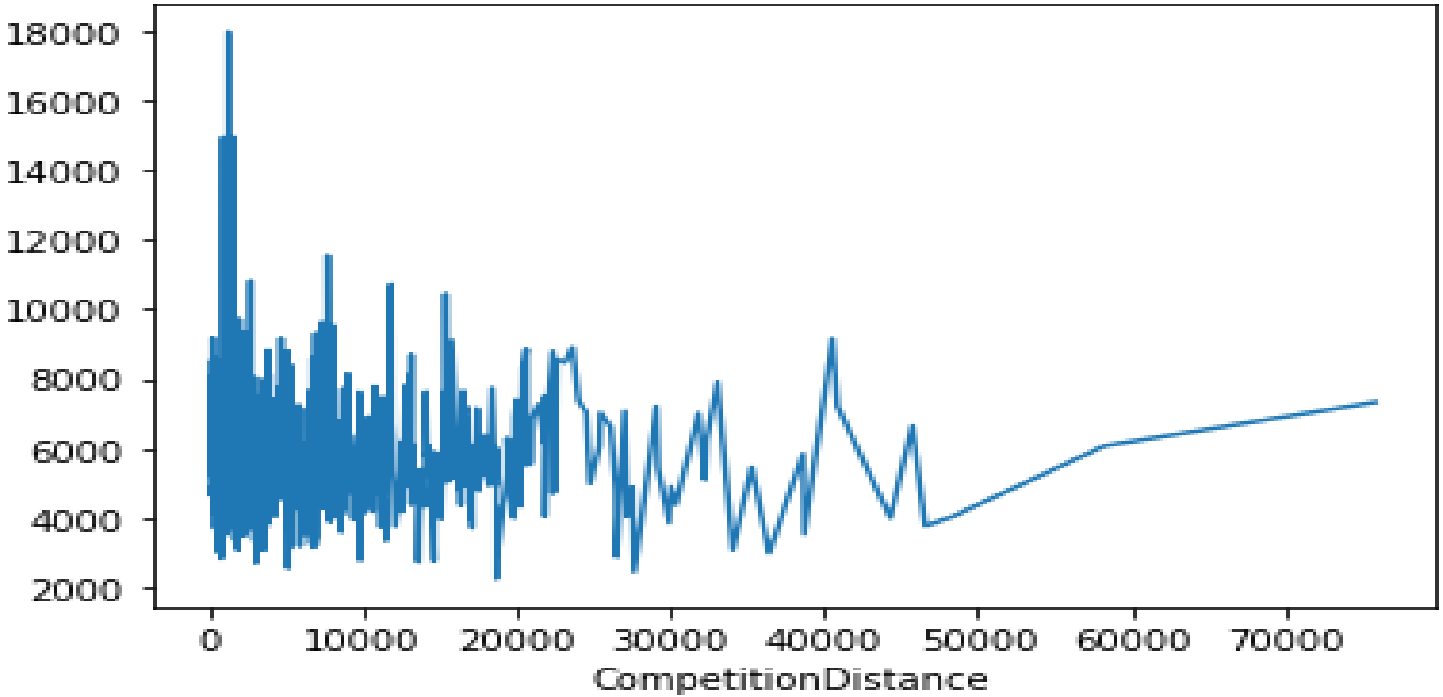


- Relationship between Monthly sales over the years

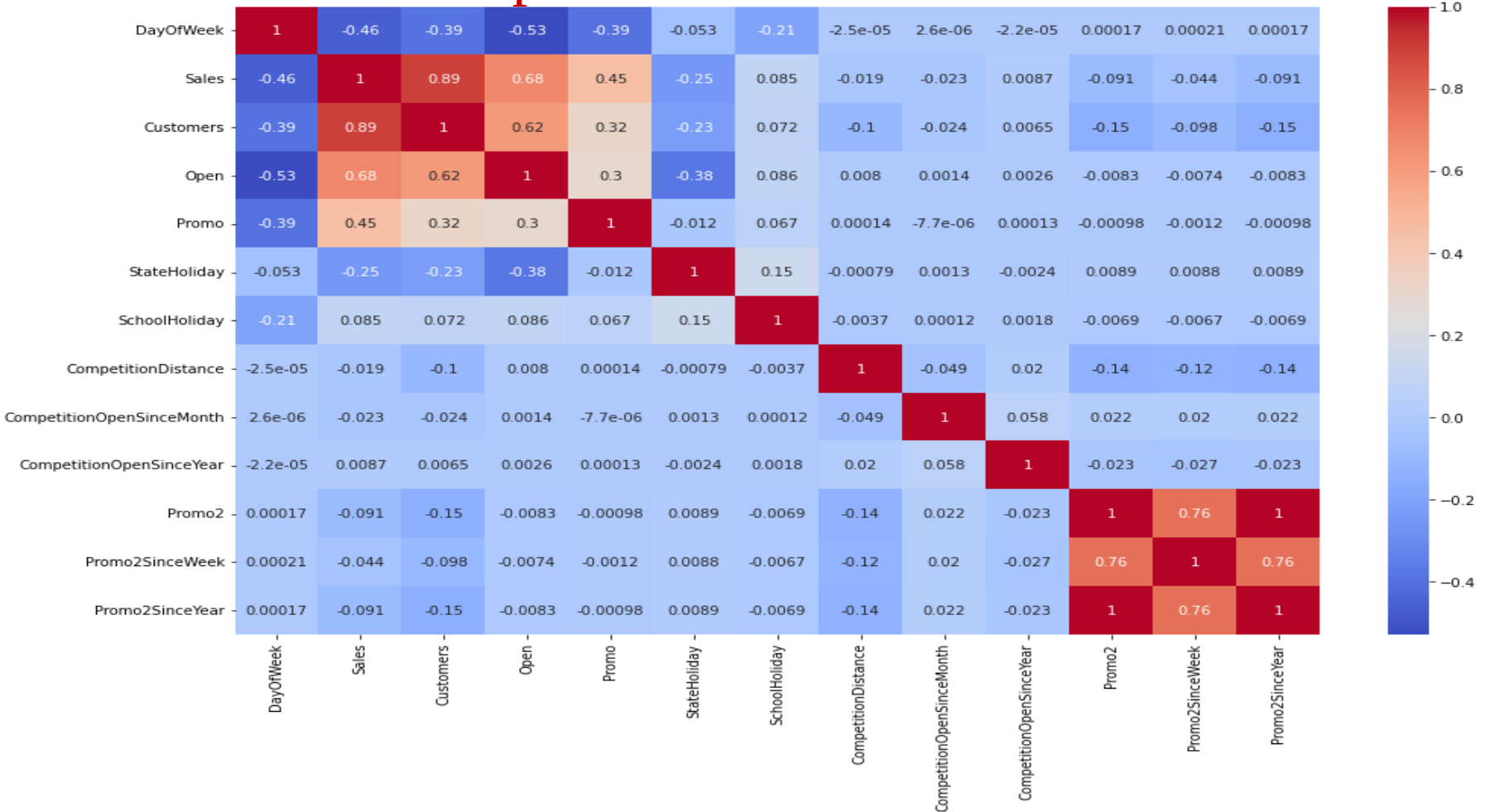
Relationship between "Sales" and "Customers"



Relationship between "Sales" and "CompetitionDistance"



Correlation Heatmap:



Data Exploration:

Checking in OLS Model:

OLS Regression Results

Dep. Variable:	Sales	**R-squared:**	0.737
Model:	OLS	**Adj. R-squared:**	0.737
Method:	Least Squares	**F-statistic:**	2.954e+05
Date:	Sun, 11 Dec 2022	**Prob (F-statistic):**	0.00
Time:	14:13:27	**Log-Likelihood:**	-7.4240e+06
No. Observations:	844392	**AIC:**	1.485e+07
Df Residuals:	844383	**BIC:**	1.485e+07
Df Model:	8		
Covariance Type:	nonrobust		

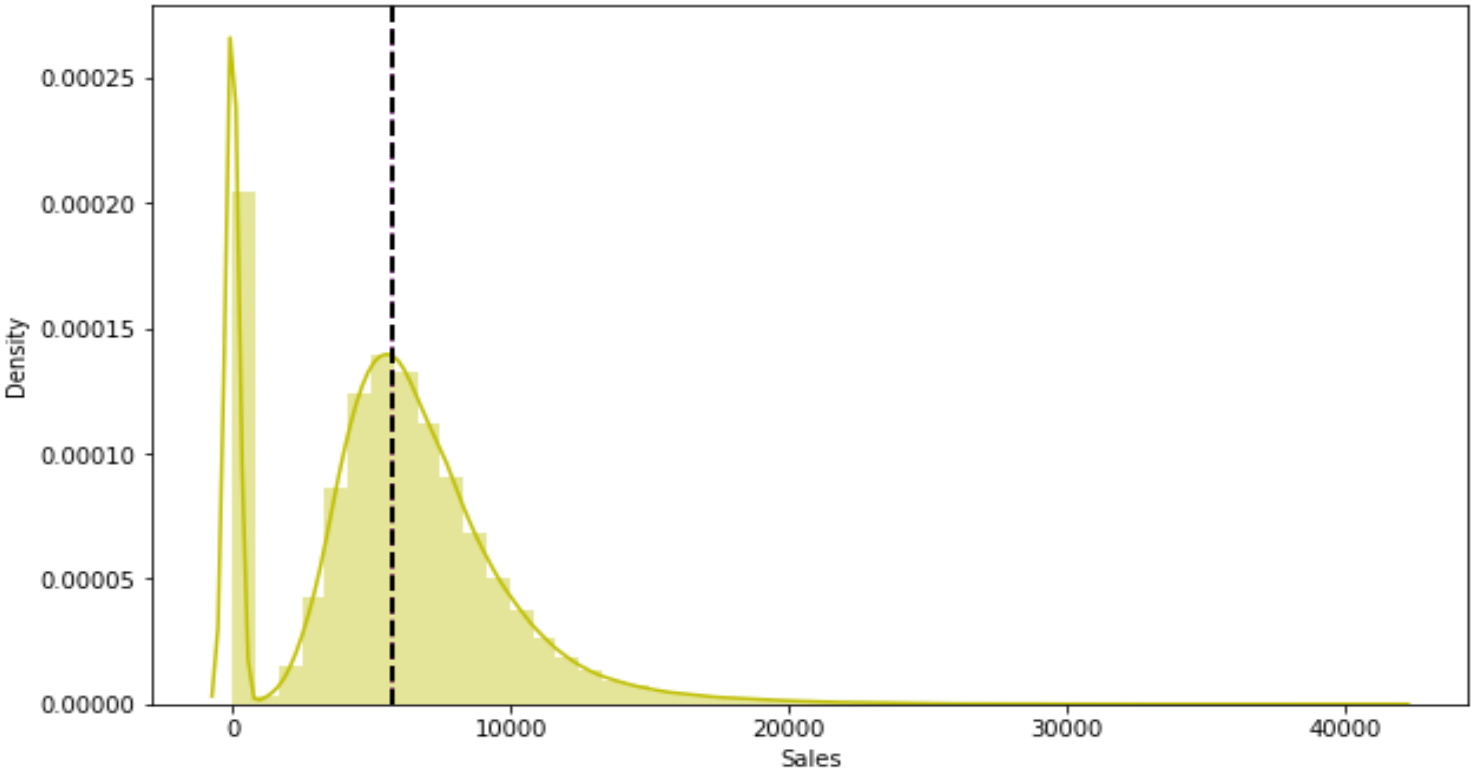
	coef	std err	t	P>|t|	[0.025	0.975]
const	-3.464e+05	4665.459	-74.253	0.000	-3.56e+05	-3.37e+05
SchoolHoliday	69.2782	4.467	15.509	0.000	60.523	78.033
Month	50.0924	1.954	25.638	0.000	46.263	53.922
Year	172.6205	2.316	74.523	0.000	168.081	177.160
WeekOfYear	-2.4707	0.448	-5.512	0.000	-3.349	-1.592
Day	1.6456	0.205	8.011	0.000	1.243	2.048
Promo	1393.3589	3.588	388.352	0.000	1386.327	1400.391
Customers	6.1320	0.004	1377.170	0.000	6.123	6.141
CompetitionDistance	0.0325	0.000	144.596	0.000	0.032	0.033

Omnibus:	134549.931	**Durbin-Watson:**	1.768
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	2137181.614
Skew:	0.251	**Prob(JB):**	0.00
Kurtosis:	10.778	**Cond. No.**	2.58e+07

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Normalise Sales column data:

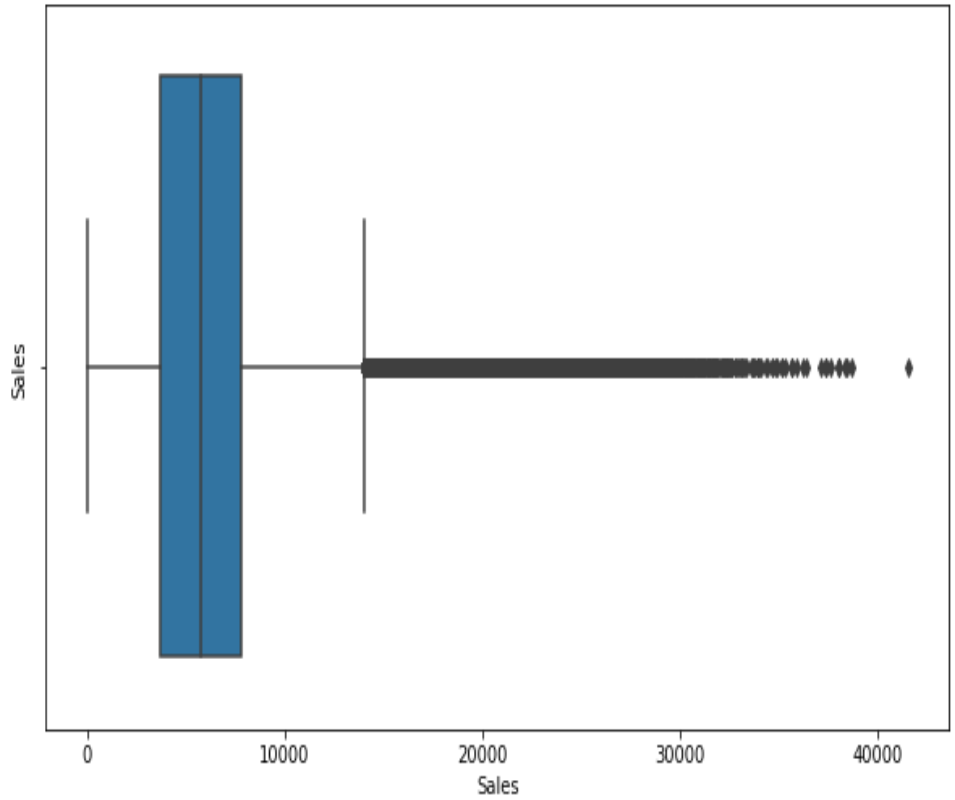
- 0 is raised because most of the times store was closed.



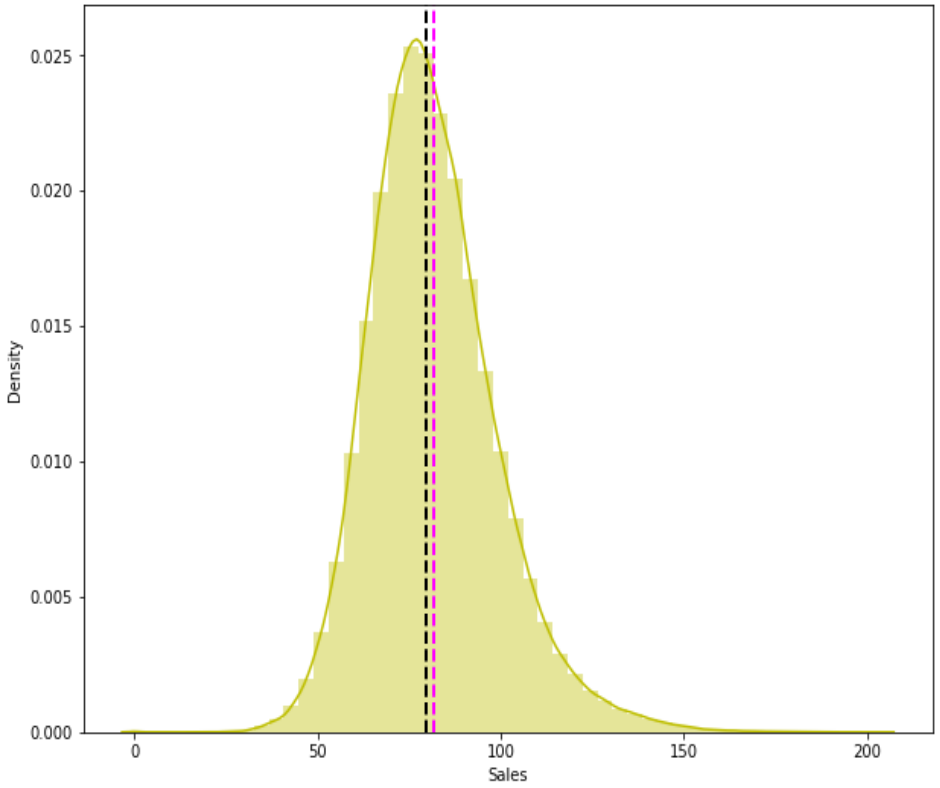
Feature Engineering :

- Feature Selection
- Handling missing values
- Handling imbalanced data
- Handling outliers
- Encoding

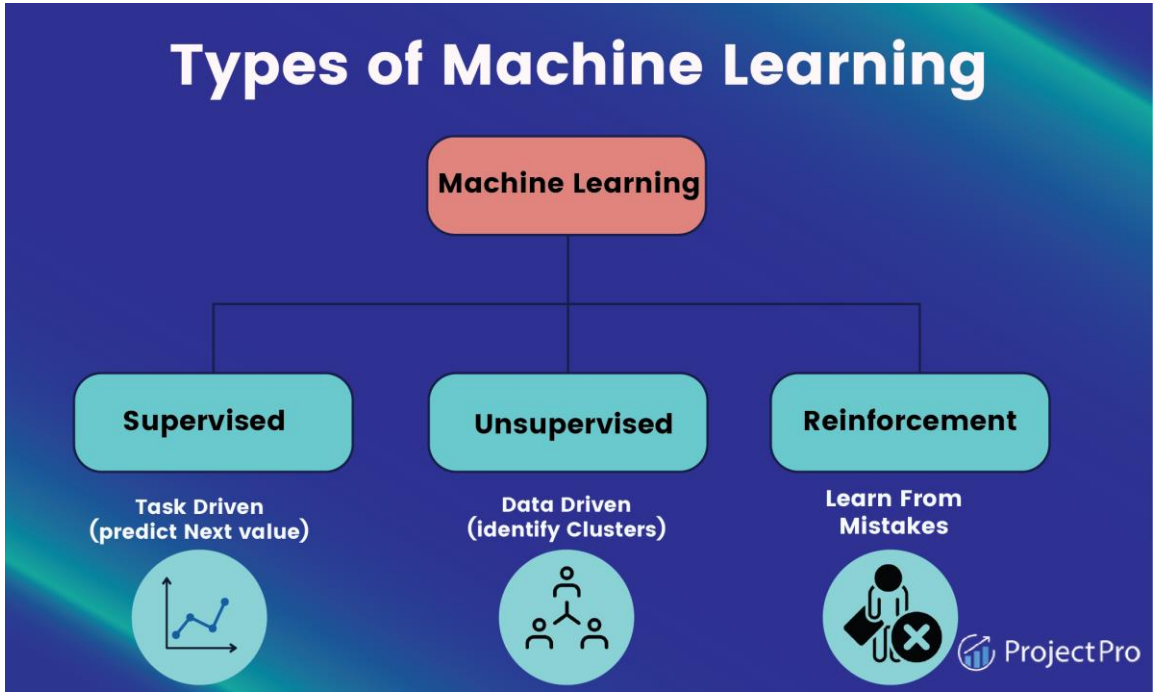
Handling Outliers:

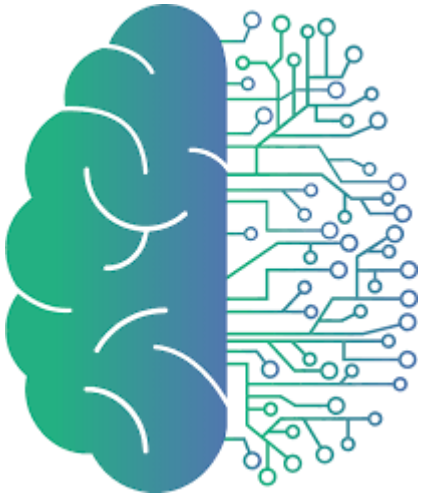


Data Transformation:



Machine Learning:





We used Supervised machine learning algorithm to predict the results of the given rossmann store dataset.

Model Training :

- Train Test split for regression
- Evaluation metric are:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2$$

$$RMSE = \sqrt{\sum_{i=1}^{n}\frac{(\hat{y}_i - y_i)^2}{n}}$$

$$\text{R}^2 = 1 - \frac{\text{Unexplained Variation}}{\text{Total Variation}}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

$$Adjusted\ R^2 = 1 - (1 - R^2) * \frac{n-1}{n-k-1}$$

ML Model Implementation:

1. LINEAR REGRESSION :

$y\_pred = \beta0+\beta1x$

where β0 and β1 are intercept and slope respectively.

In case of multiple features the formula translates into:

$y\_pred = \beta0+\beta1x1+\beta2x2+\beta3x3+.....$

where x_1,x_2,x_3 are the features values and β0,β1,β2..... are weights assigned to each of the features. These become the parameters which the algorithm tries to learn using Gradient descent.

Assumptions:

- No multicollinearity in the dataset.
- Independent variables should show linear relationship with dv.
- Residual mean should be 0 or close to 0.
- There should be no heteroscedasticity i.e., variance should be constant along the line of best fit.

Evaluation metric for Linear Regression:

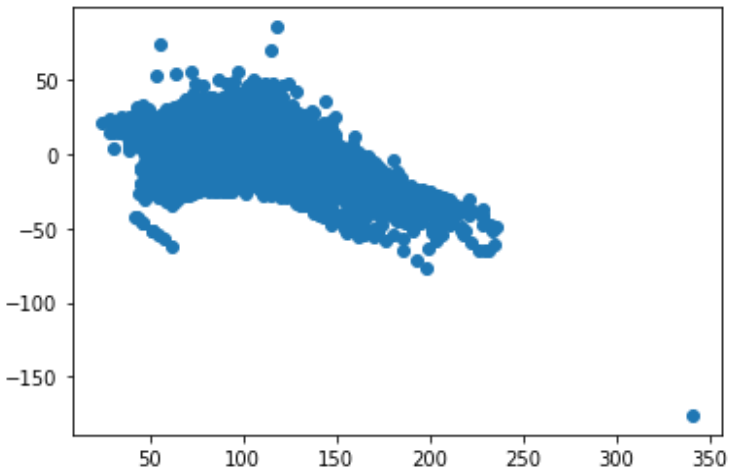
Training Set

MSE : 59.276727972734506
RMSE : 7.69913813181284
MAE : 5.8838789055412875
R2 : 0.8059731013774146
Adjusted R2 : 0.8059492008919658

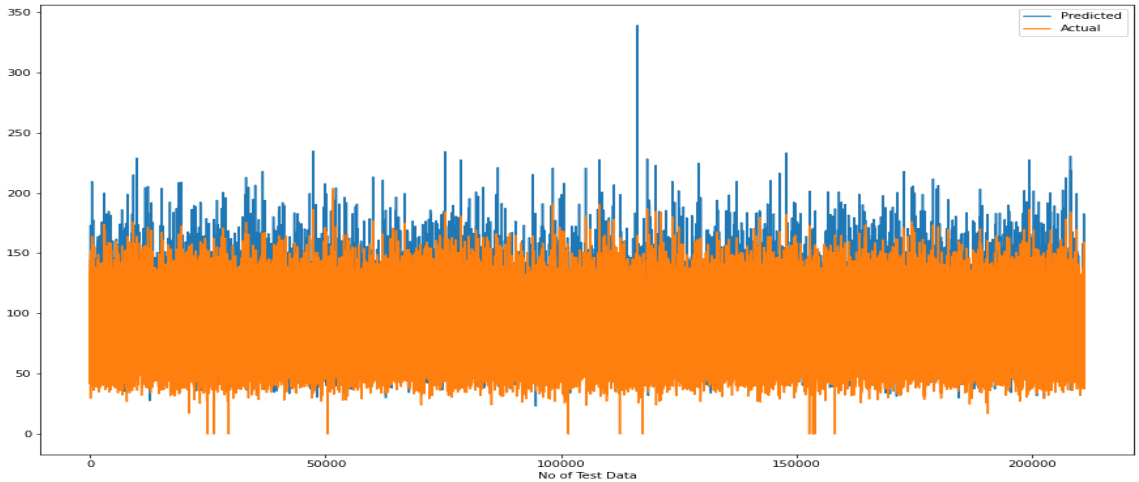
Testing Set:

MSE : 58.96999326073211
RMSE : 7.679192227098636
MAE : 5.868394244951874
R2 : 0.8064733653936786
Adjusted R2 : 0.8064458584928453

Heteroscadacity:



Predicted v/s Actual:



2.LASSO REGRESSION:

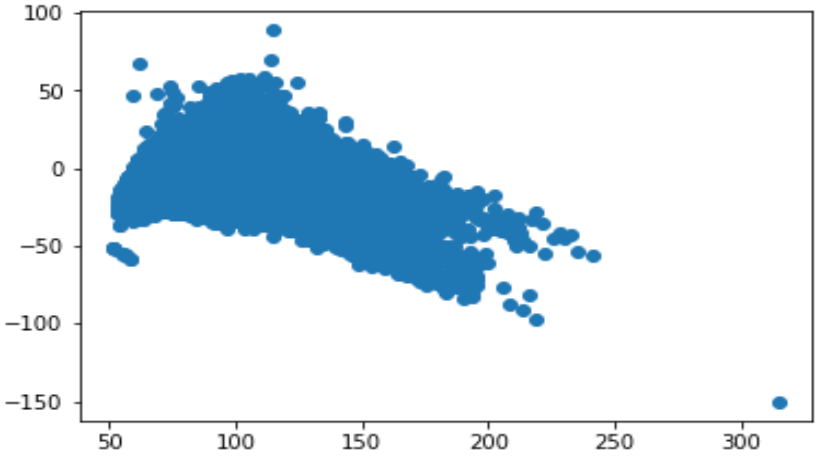
Training Set

MSE : 86.29989806371088
RMSE : 9.289773843517983
MAE : 6.896837496975858
R2 : 0.7175198067536885
Adjusted R2 : 0.7173524384576313

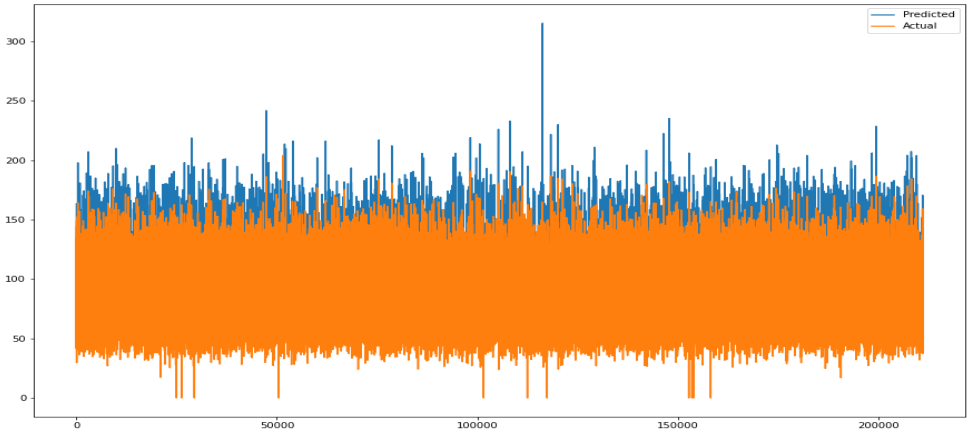
Testing Set:

MSE : 85.5437560685491
RMSE : 9.248986759021179
MAE : 6.865872607516707
R2 : 0.7192640814738858
Adjusted R2 : 0.7190977466530766

Heteroscadacity:



Predicted v/s Actual:



3.RIDGE REGRESSION

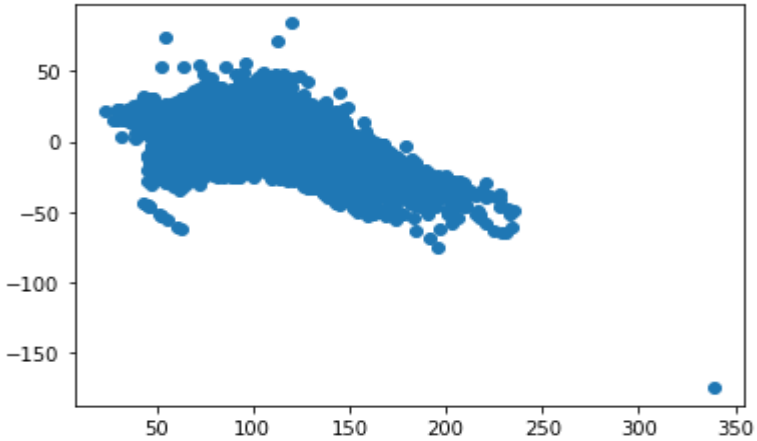
Training Set

MSE : 59.27942404088632
RMSE : 7.699313218780381
MAE : 5.88446325107593
R2 : 0.8059642765019557
Adjusted R2 : 0.8059366972417921

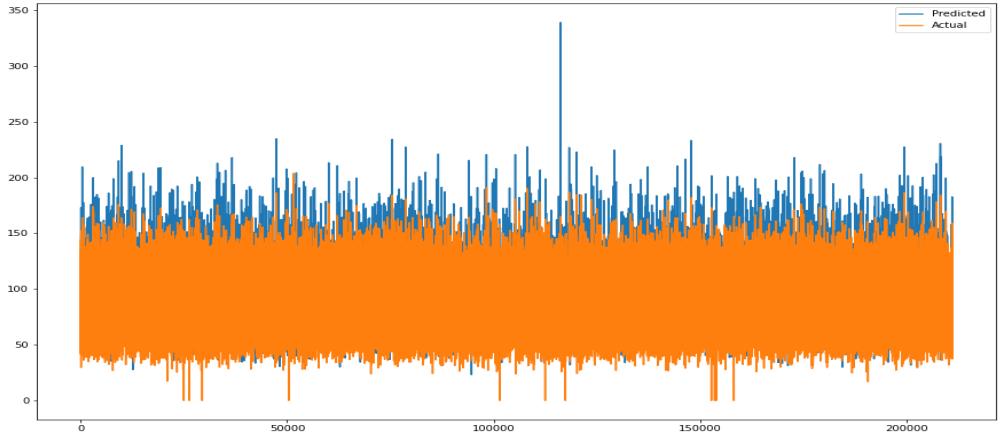
Testing Set:

MSE : 58.972947200770896
RMSE : 7.679384558724149
MAE : 5.869074379545238
R2 : 0.8064636712078237
Adjusted R2 : 0.8064361629291076

Heteroscadacity:



Predicted v/s Actual:



4.ELASTIC NET REGRESSION

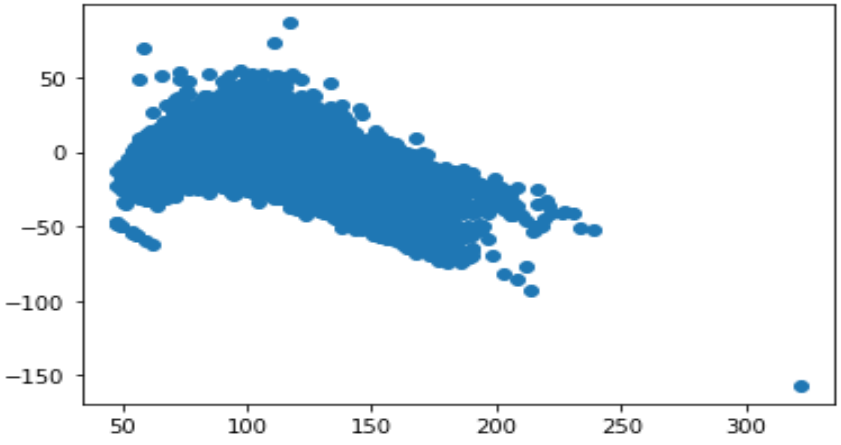
Training Set

MSE : 68.77317252102458
RMSE : 8.292959213756244
MAE : 6.218439953887624
R2 : 0.7748889685876694
Adjusted R2 : 0.7748569724397999

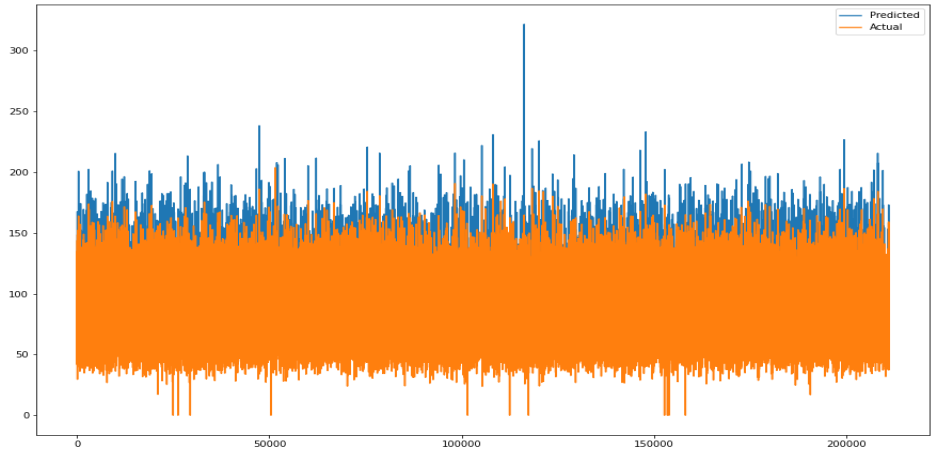
Testing Set:

MSE : 68.22971072133679
RMSE : 8.260127766647244
MAE : 6.195000323639775
R2 : 0.776084995674303
Adjusted R2 : 0.7760531695236978

Heteroscadacity:



Predicted v/s Actual:



5.DECISION TREE

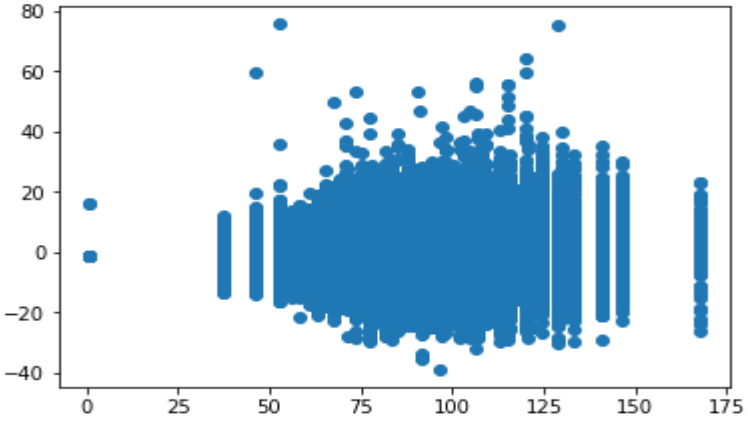
- Training Set

```
Model Score: 0.8534250504924248
MSE : 44.7797881183391
RMSE : 6.691770178236779
MAE : 5.208217452004753
R2 : 0.8534250504924248
Adjusted R2 : 0.8533382054670733
```

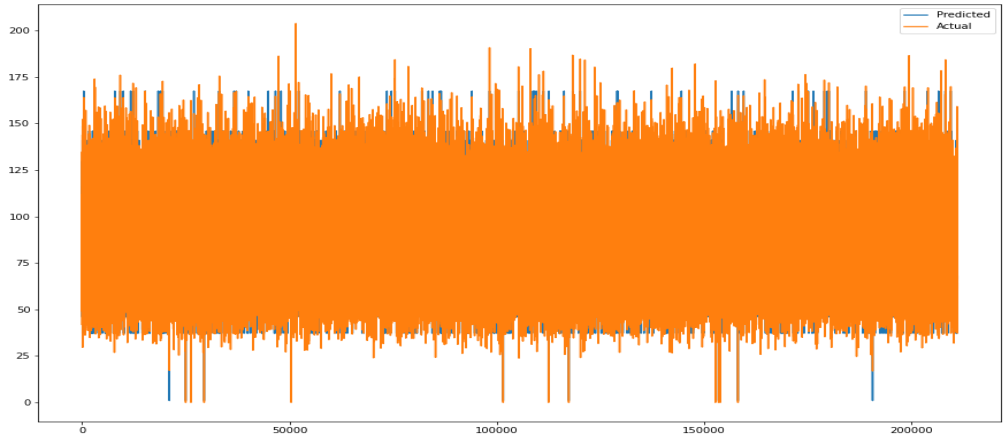
Testing Set:

```
MSE : 44.899403933635455
RMSE : 6.700701749342039
MAE : 5.222535119407028
R2 : 0.8526499655394666
Adjusted R2 : 0.8525626612796238
```

Heteroscadacity:



Predicted v/s Actual:



6.RANDOM FOREST

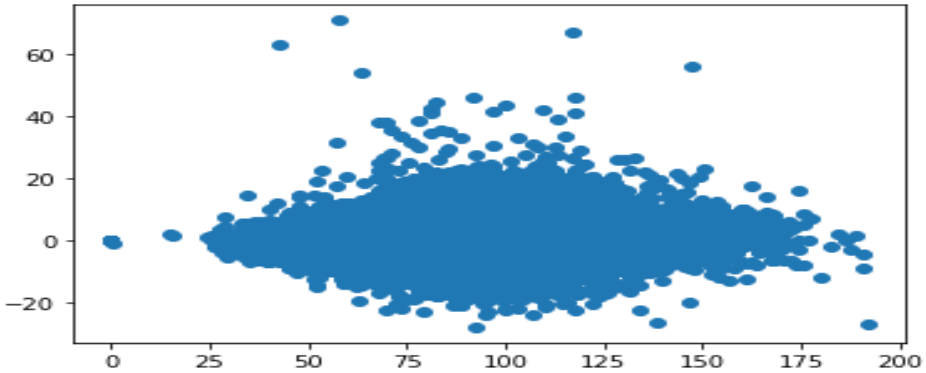
- Training Set

```
Model Score: 0.9610248152637217
MSE : 11.907222347523065
RMSE : 3.4506843303210255
MAE : 2.559175739692221
R2 : 0.9610248152637217
Adjusted R2 : 0.9610017226348798
```

Testing Set:

```
MSE : 13.825271636857405
RMSE : 3.7182350163562017
MAE : 2.755382587730081
R2 : 0.9546284789186014
Adjusted R2 : 0.9546015964880648
```

Heteroscadacity:



7.GRADIENT BOOSTING

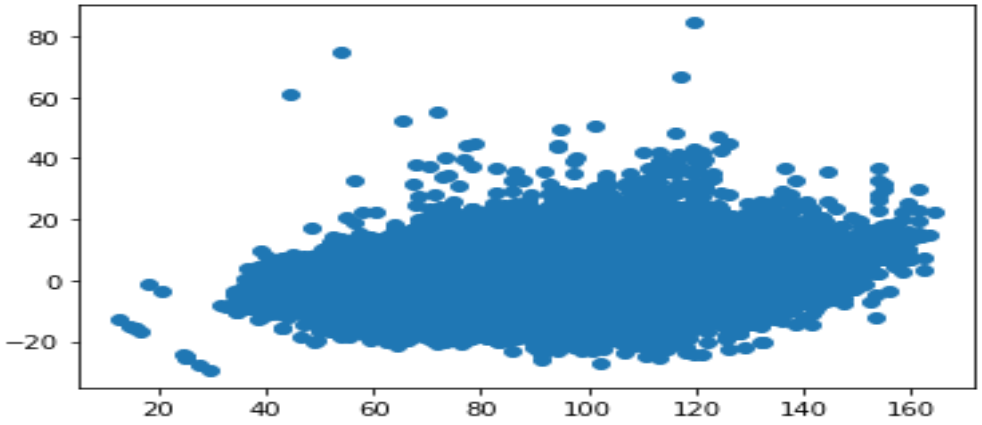
- Training Set

```
Model Score: 0.892241248557284
MSE : 32.9211374366
RMSE : 5.737694435624818
MAE : 4.473589414186537
R2 : 0.892241248557284
Adjusted R2 : 0.8921779130348291
```

Testing Set:

```
MSE : 32.74567494212779
RMSE : 5.722383676592107
MAE : 4.467254023034908
R2 : 0.8925358488436113
Adjusted R2 : 0.8924726864733392
```

Heteroscadacity:



Hyperparameter tuning:

Using GridSearchCV

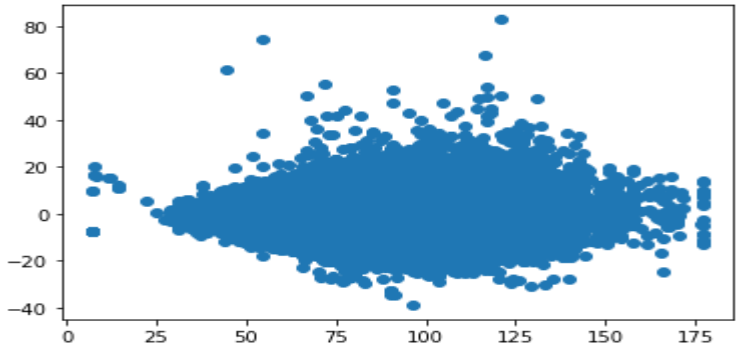
Training Set

```
Model Score: 0.8741386433080742
MSE : 38.4516242638027
RMSE : 6.200937369769404
MAE : 4.81257362747888
R2 : 0.8741386433080742
Adjusted R2 : 0.8741207540089382
```

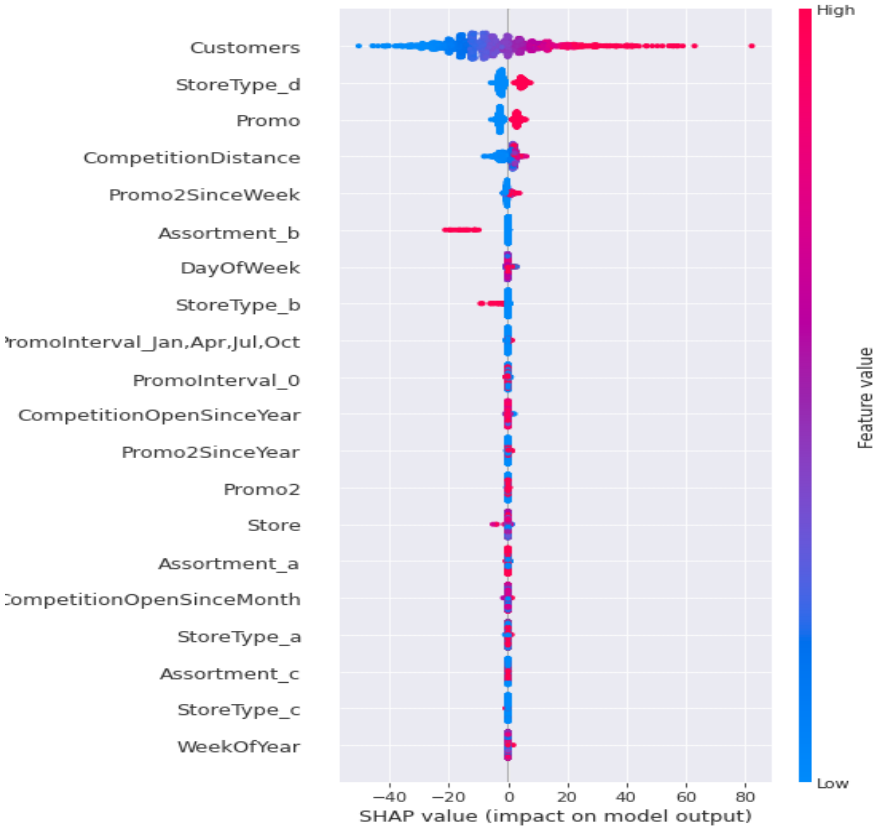
Testing Set:

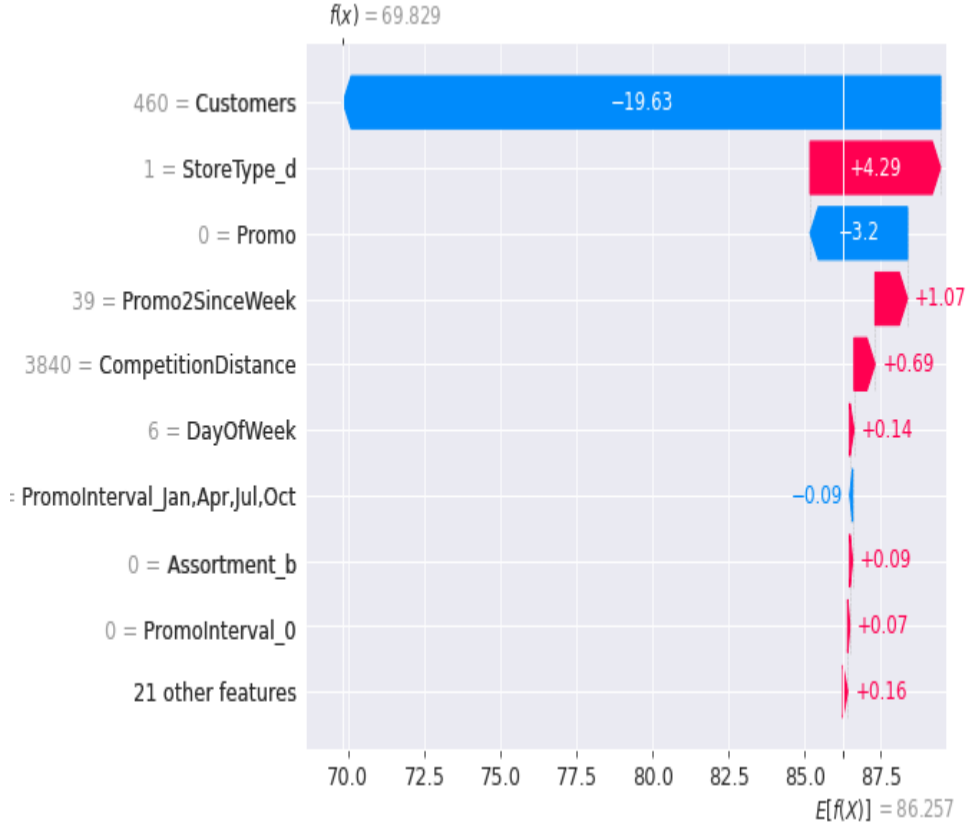
```
MSE : 38.50578960989461
RMSE : 6.205303345517817
MAE : 4.819207983937441
R2 : 0.873632410926115
Adjusted R2 : 0.8736144496736585
```

Heteroscadacity:



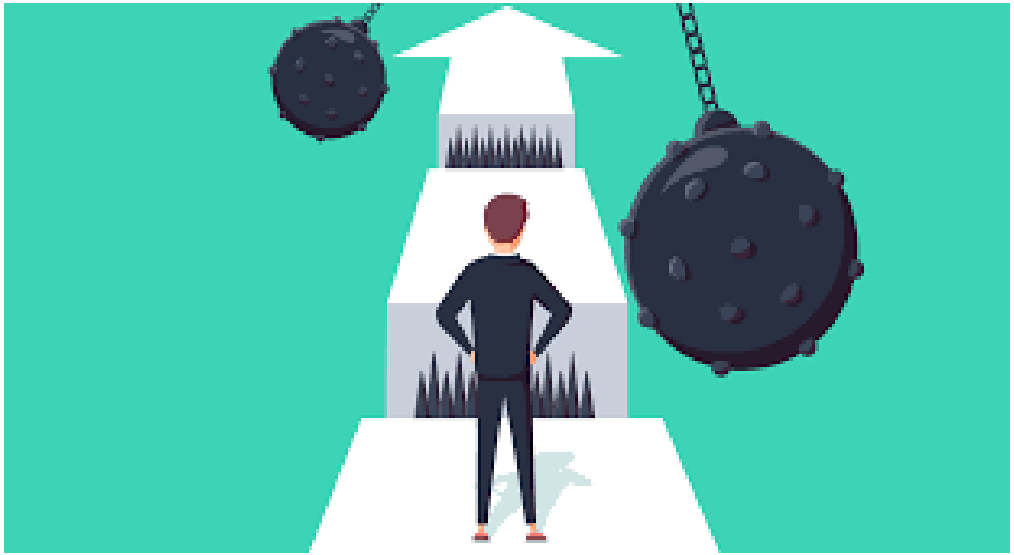
SHAP Implementation:





Challenges And Learnings:

- Poor Quality of Data. Data plays a significant role in the machine learning process.
- It was hard to find which graph technique to use.
- Removing null values.
- ML Models takes lot of time to implement and provide efficient results.



Conclusion:

- Conclusions of EDA and ML Models:
- Sales is highly correlated to number of Customers.
- For all stores, Promotion leads to increase in Sales and Customers both.
- During starting days of the week like Monday,Tuesday, wednesday, friday the sales got a positive growth.But due to holiday on sunday completely it leads to zero.It gives the negative growth. So, Keeping store open on starting days of the week is must to grow the business. in peak level.
- The most selling and crowded store type is A and D.Store type b is the least one.
- More stores are opened during School holidays than State holidays. Sales are increased during normal days than the public holidays.
- Sales rise up by the end of the year before the holidays. Sales for 2014 went down there for a couple months - July to September, indicating stores closed due to refurbishment.
- Highest average sales were seen with Assortment levels-a which is 'basic'. We can see the drop of sales on 'b'(extra) assortment.
- During the initial consecutive years we could see positive impact on sales due to promotions.
- Day of the week has a negative correlation indicating low sales as the weekends, and promo, customers and open has positive correlation.
- CompetitionDistance showing negative correlation suggests that as the distance increases sales reduce, which was also observed through the plot earlier.
- There's multicollinearity involved in the dataset as well. The features telling the same story like Promo2, Promo2 since week and year are showing multicollinearity.

ML Model:

Next we implemented 7 machine learning algorithms Linear Regression,lasso,ridge,elasticnet,decission tree, Random Forest and XGBoost. We did hyperparameter tuning to improve our model performance. The results of our evaluation are:

- No overfitting is seen.
- Random forest Regressor and Gradient Boosting gridsearchcv gives the highest R2 score of 96% and 88% recpectively for Train Set and 89% for Test set.
- Feature Importance value for Random Forest and Gradient Boost are almost same.
- We are able to see the importance of the features and how it is effecting to dependent variable. We can ignore the 'customers' feature because, The task is not to predict the number of customers, so fitting a model based on the given variables in the test set to predict the number of customers surely makes no sense. But features like 'CompetitionDistance','Promo','StoreType','DayOfWeek' and 'Assortment' are very important features where dependent variable depends on these features.
- We can deploy this model.

Q & A

THANK YOU