Project

Project Title – Rossmann Sales Prediction

By : Shaurabh Pandey

Analysing the Rossmann Stores Data and predicting their daily sales for up to six weeks in advance.

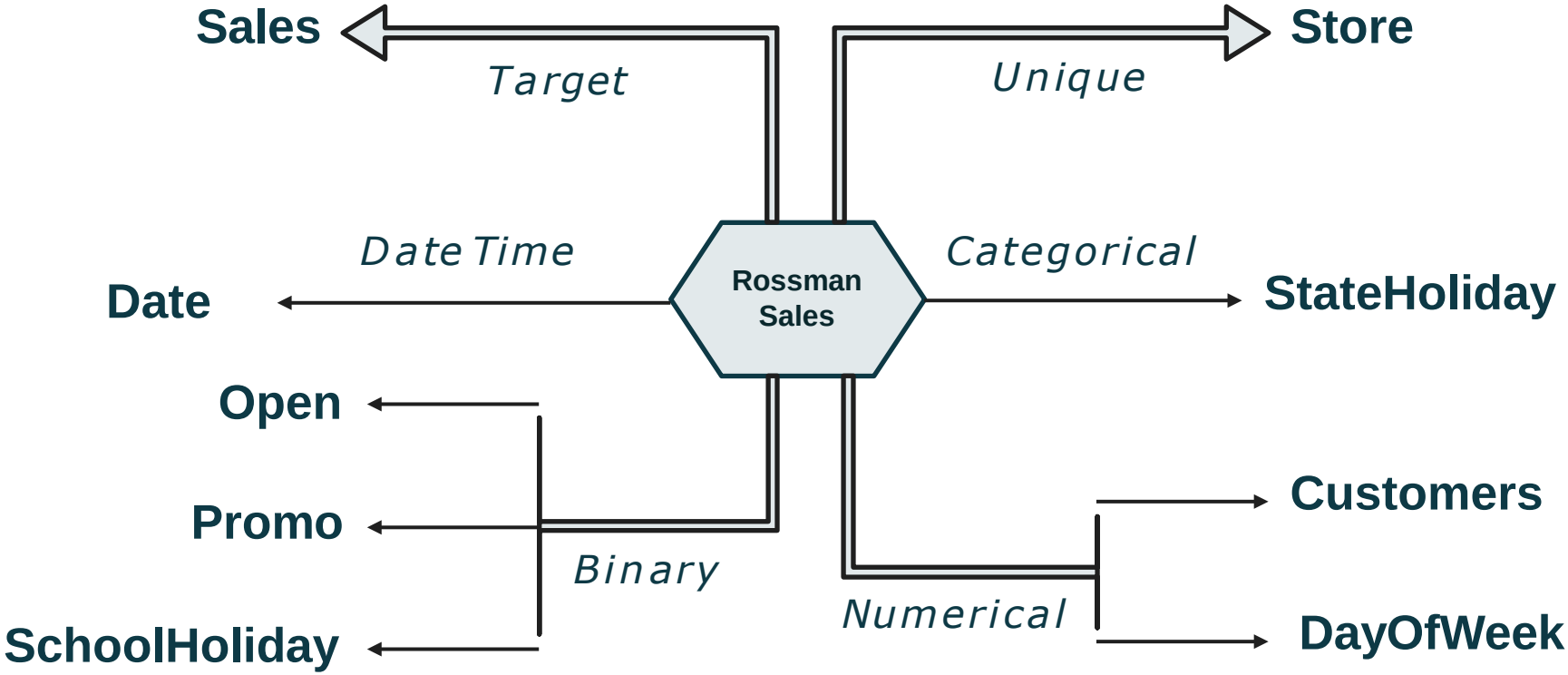
Content

- Defining the problem statement
- Data cleaning and arranging.
- Feature Engineering and EDA
- Model Implementation
- Model Validation and Selection

Rossmann Store

- **Dirk Rossmann GmbH** (Rossmann) is Germany's second-largest drugstore chain and operates over 3,000 stores in seven countries.
- Its product range includes up to 21,700 items and can vary depending on the size of the shop and the location. The products majorly consist of drugstore goods with a focus on skin, hair, body, baby and health.
- We know the store sales are influenced by many factors, including promotions, competition, school and state holidays, seasonality, and locality and so we are tasked with predicting the daily sales at the rossmann stores for up to six weeks in advance
- For this project we are provided with two datasets one containing historical data including sales for 1115 stores and the other one containing supplemental information about the stores, at the last our job is to predict the "Sales" column

Defining the Variables(1st DataFrame)



Defining the Variables (continued)

- **Store** – A unique id for each store
- **DayOfWeek** – Day of the week for which the observation is .
- **Date** – Date of the observation
- **Customers** – Number of customers for the day
- **Open** – Store is open or not
- **Promo** – Store is running a promo on the day or not
- **StateHoliday** – It states whether the day is a state holiday and if yes then which one
- **SchoolHoliday** – It states whether the day is a school holiday or not
- **Sales** – The Daily Sales volume for the day

Defining the Variables(2nd DataFrame)

Defining the Variables (continued)

- **Store** – A unique id for each store
- **StoreType** – Model of the store , we have 4 in our dataset namely a, b, c and d.
- **Assortment** – Tells the assortment level of the store whether it is basic, standard or extended
- **CompetitionDistance** – Distance in meters to the nearest competitor store
- **CompetitionOpenSinceMonth** – The month on which the nearest competitor store opened
- **CompetitionOpenSinceYear** – The year on which the nearest competitor store opened
- **Promo2** – If the store is participating in a continuing and consecutive promotion named Promo2
- **Promo2SinceWeek** – The week of the year when the store started participating in Promo2
- **Promo2SinceYear** – The year when the store started participating in Promo2
- **PromoInterval** – Naming the months when the promotion is started anew again

Data cleaning and arranging.

- We first started with merging our two dataset on the basis of store number so that we have a singular dataset with all the supplemental information for each of the store that was present in the second dataset
- Checked for duplicates in our dataset on the basis of same store number and date and the result was an empty dataset meaning we don’t have any duplicate observations
- We dropped some observations from our dataset on two conditions, first of those days on which the store was closed and then of those days on which the recorded Sales figure was zero
- Values of CompetitionOpenSinceYear, CompetitionOpenSinceMonth and CompetitionDistance were filled with zero for those rows where the values of CompetitionDistance were null making the assumption that if CompetitionDistance is null there is probably no competitor store nearby

- Null values of CompetitionOpenSinceMonth and CompetitionOpenSinceYear were filled with median values of their respective columns
- Values of CompetitionDistance , CompetitionOpenSinceYear and CompetitionOpenSinceMonth were set to zero for all those observation which are of before the date on which the nearby competitor store opened as for those dates there isn’t a competitor store in nearby
- We noticed that for any observation if the value of Promo2 is zero i.e the store isn’t participating in Promo2 then for those observations the values of Promo2SinceWeek , Promo2SinceYear and PromoInterval were all collectively null and so we replaced the null values of all these three rows with zero
- For those stores which participated in promo2 , the values of promo2 were set to zero for those observations which are of before the time store started participating in promo2

Feature Engineering

Our dataset was a time series one and because data points in time series are collected at adjacent time periods there is potential for correlation between observations hence it was important for us to use different feature engineering methods in order to best learn from the past values and better forecast the future Sales figure

- Using the **CompetitionOpenSinceMonth** and **CompetitionOpenSinceYear** columns we created a new column named **NumberOfMonthsFacedCompetition** which denoted the number of months gone by since the nearby competitor store opened
- Using the **PromoInterval** and **Date column** we created a new column named **isPromoMonth** which denoted whether the promotion started anew in the month for which the observation is
- Using **Promo2SinceWeek** and **Promo2SinceYear** we created a new column named **MonthsOfPromo2** which denoted the number of months passed since the time store started participating in Promo2

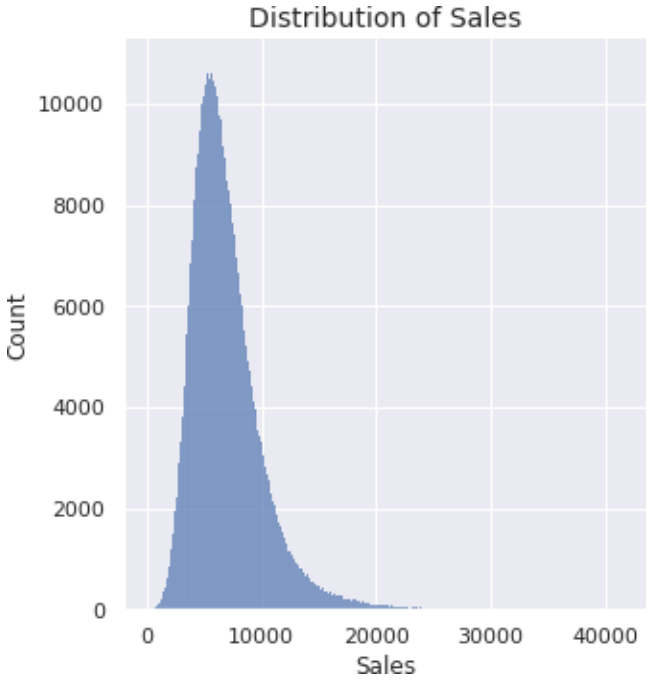
Feature Engineering (continued)

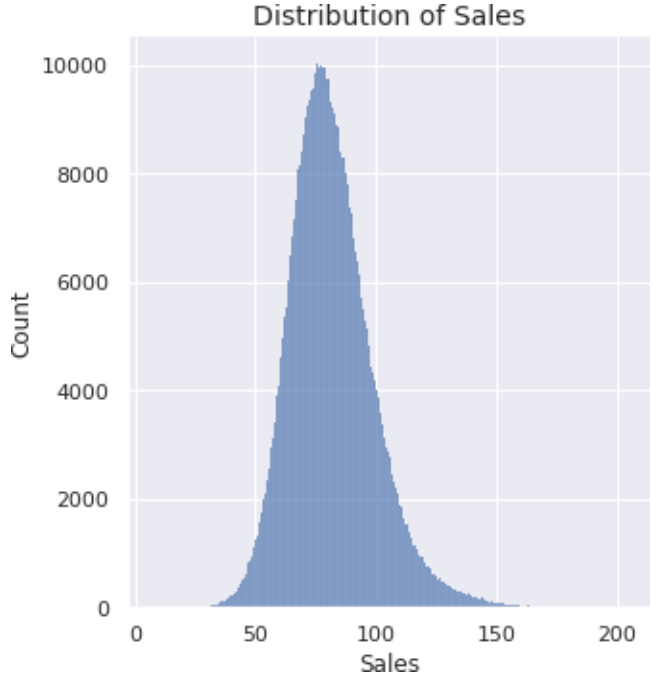
- To better learn from the past sales data we at first grouped our dataset by store, month and year and calculated mean on the number of customers and sales , this gave us average Sales and Customers data for each month of each year and then we shifted the Avg Sales and Customers column values by two places forward for each of the store such that each row now had Avg Sales and Customers value from 2 months back

	Store	Month	Year	2MonthAgoSale	2MonthAgoCustomers
0	1	1	2013	0.000000	0.000
1	1	2	2013	0.000000	0.000
2	1	3	2013	4939.653846	611.500
3	1	4	2013	5219.625000	632.875
4	1	5	2013	5806.760000	702.960

We merged this dataset with our main dataset on the basis of store , month and year such that now we had two new columns named **2MonthAgoSale** and **2MonthAgoCustomers** for each of the observation in our main dataset

Target variable (sales) analysis

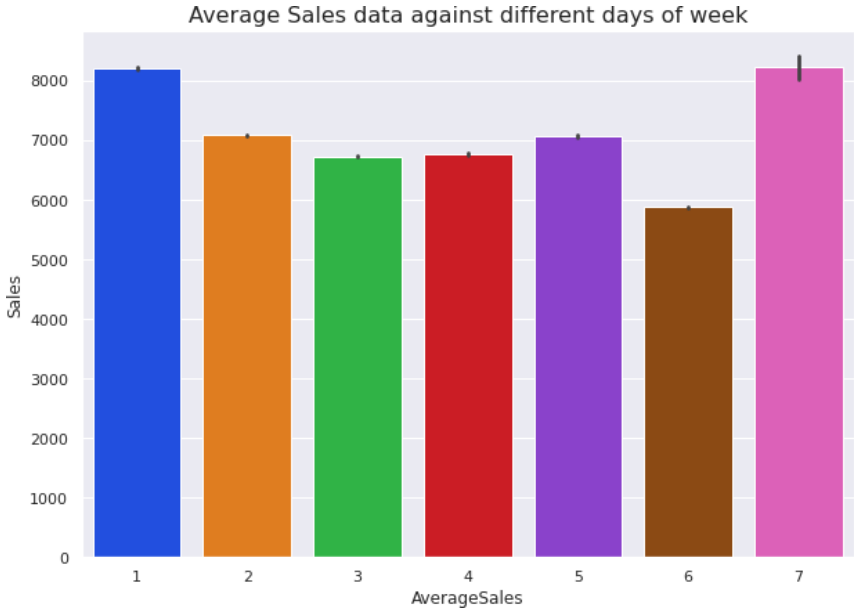




As the distribution of our target variable was positively skewed we took square root to make it a normal distribution as visible in the right hand side graph and performed all prediction on the transformed dependent variable

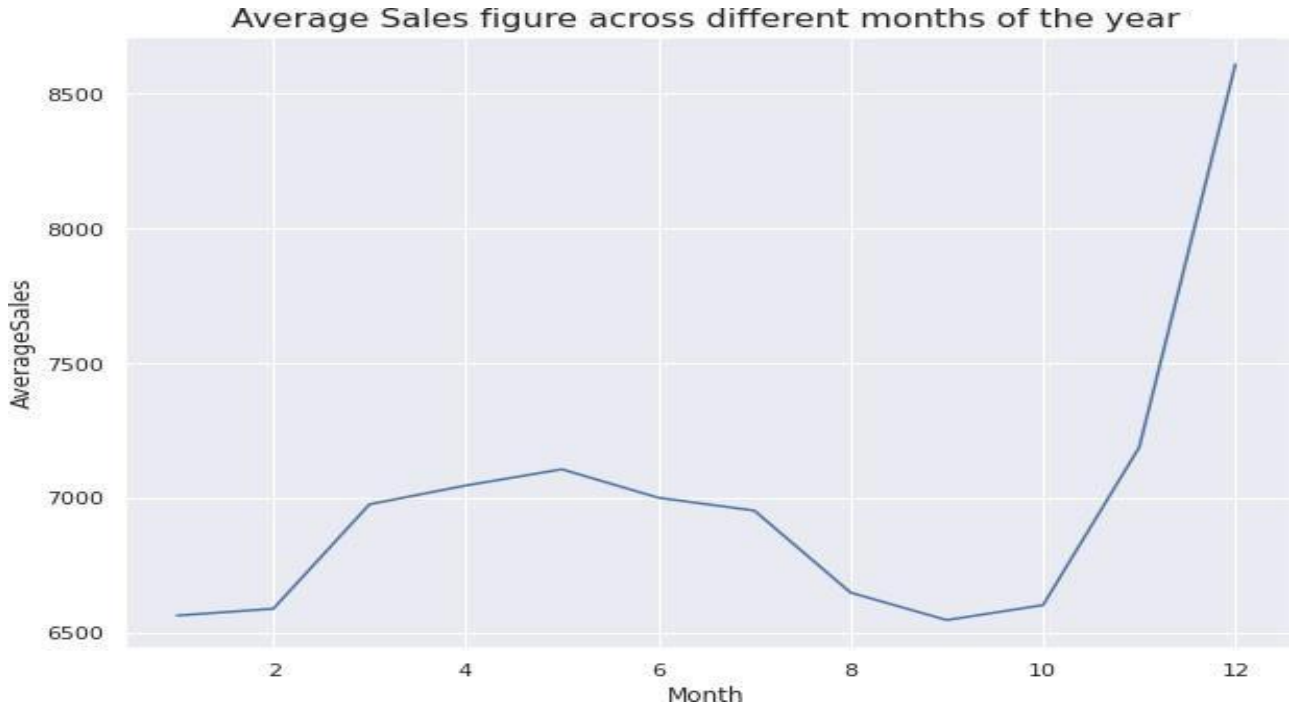
Relation of Day of Week with Sales and Customers

The Customers traffic is much higher on Sundays as compared to rest of the days of the week and one can relate this trend to the fact that Sunday is a holiday and people like to go out for shopping on holidays



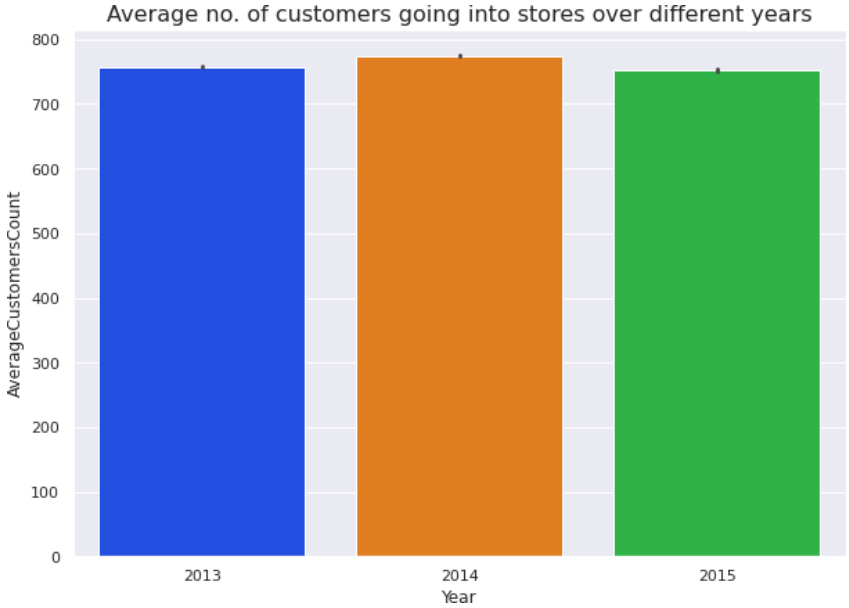
Daily Sales figure is lowest on Saturday and comparatively higher on Sundays and Mondays a which can be due to the fact that Sunday is holiday and on Mondays people tend to but stuffs for the rest of the week

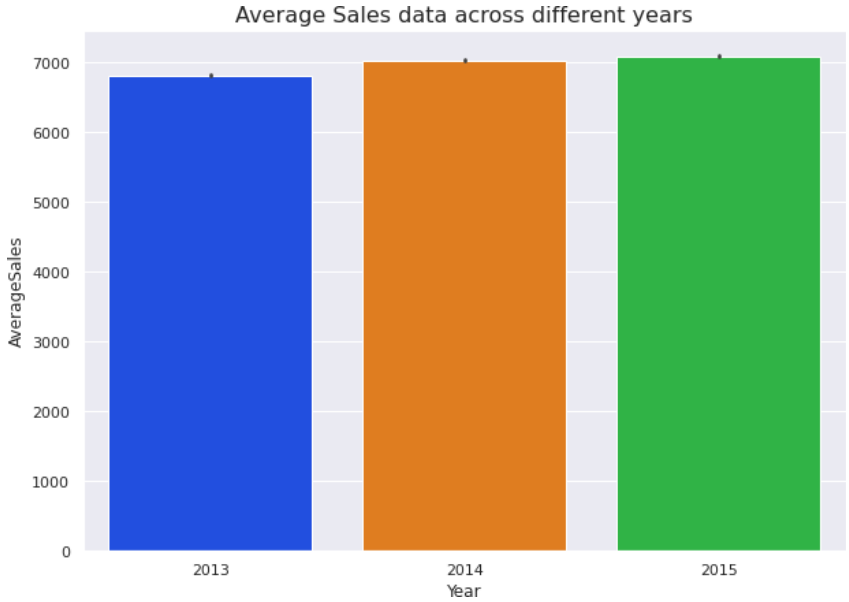
Relation between Month of the year and Sales



At two times of the year we can see Average Daily figure experiencing a boost first around the time of may-july and then a very sharp second rise in the month of November , December when the festive season starts

Relation of Year with Sales and Customers

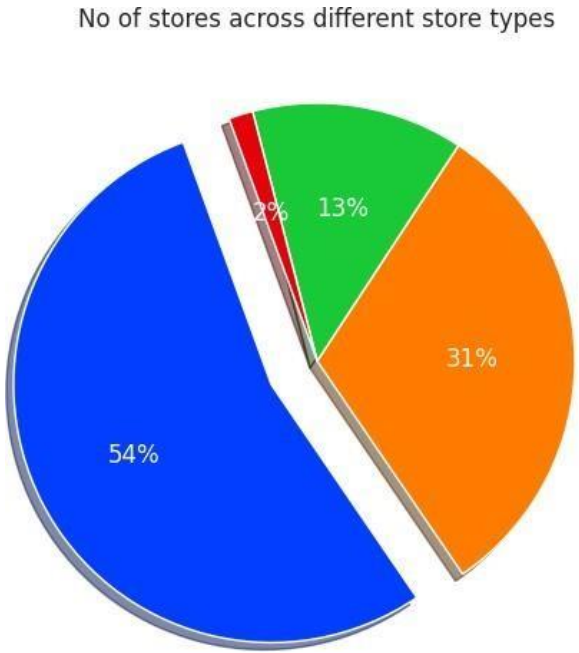




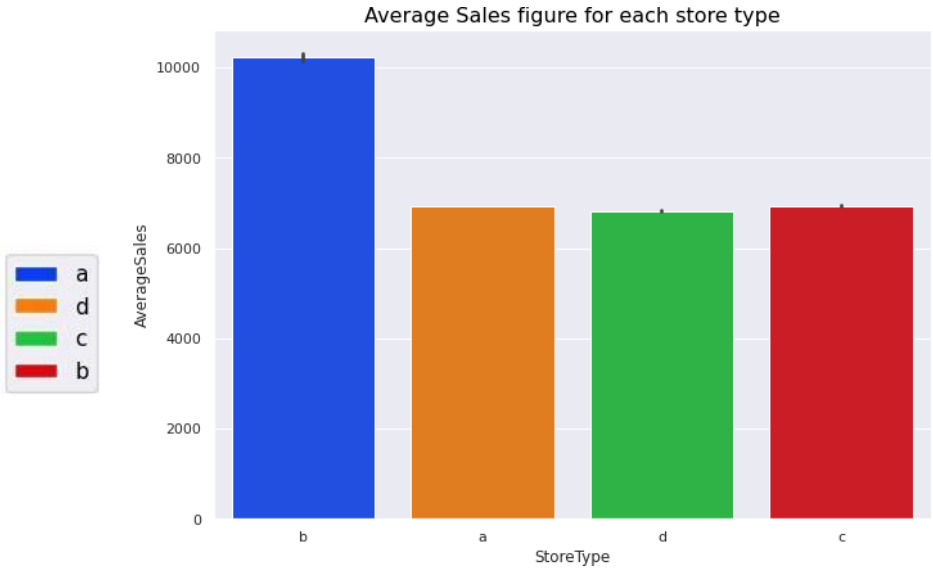
Average Daily Customers count was relatively highest in 2014 and almost on the same levels In 2013 and 2015

With each passing year we can notice the Average daily sales figure slightly increasing

Relation Between StoreType and Sales

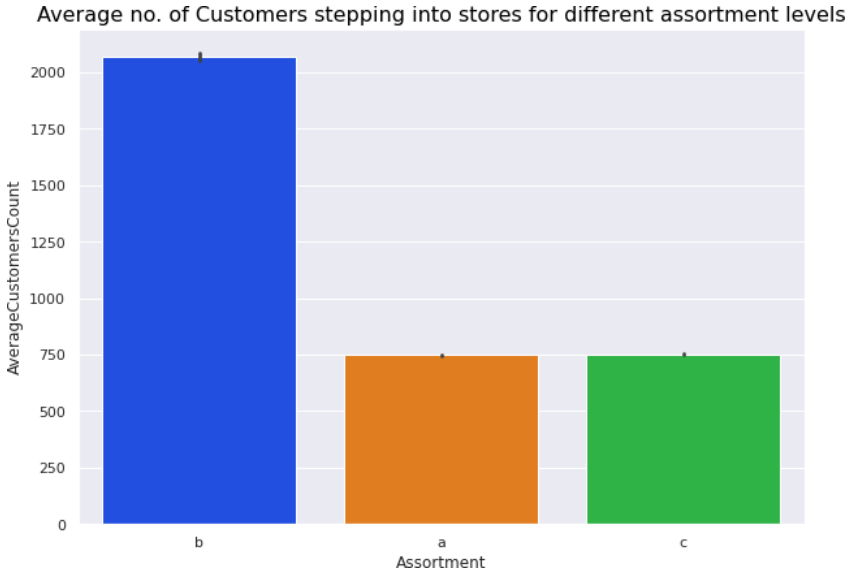


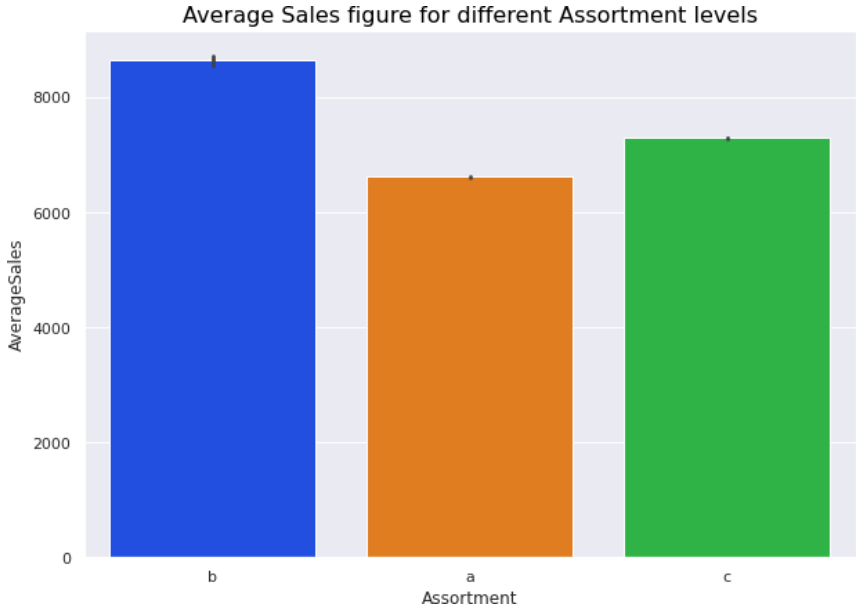
More than half of the store in our dataset are of type **a** followed by stores of type **d** which is 31% in total , store of type **b** are only 2% in total



Although stores of type b are very less in number as we can see in the pie chart on the left hand side they do record high numbers for daily sales as compared to all the other store types which more or less record the same average daily sales

Relation of Assortment with Sales and Customers

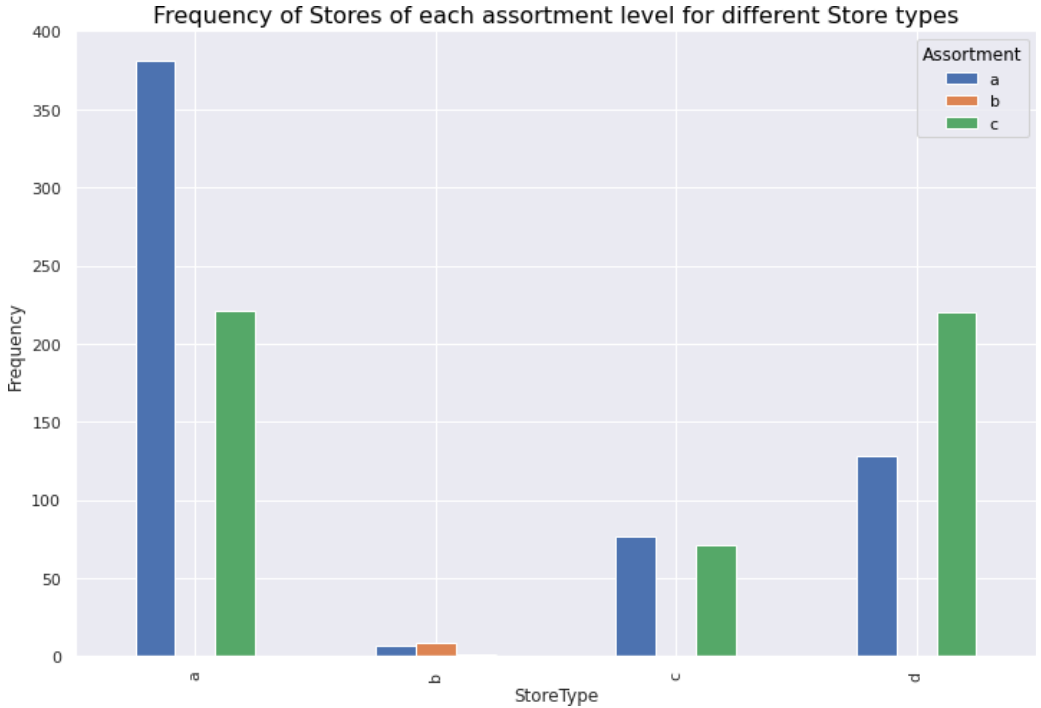




Stores with standard assortment level denoted by letter b can be seen welcoming more than double the number of customers on a daily basis as compared to store of other different assortment levels

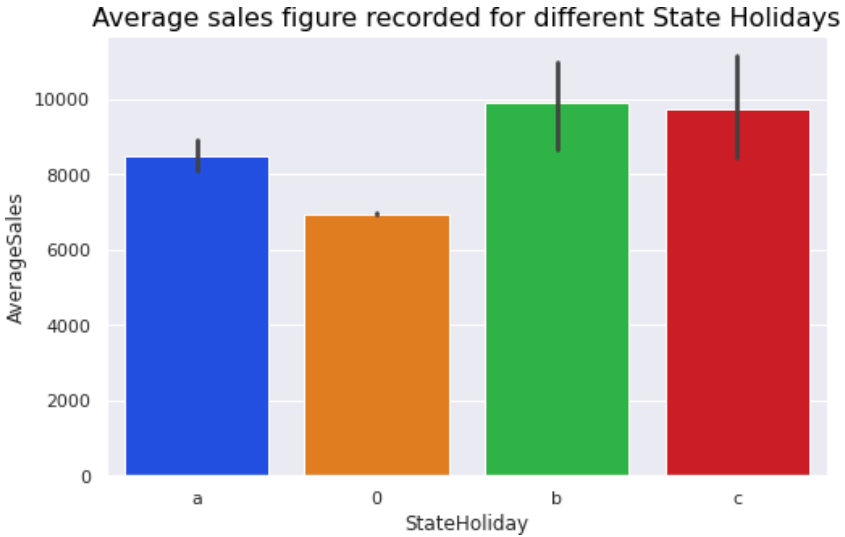
Similar to the trend that we saw in the customers graph on the left hand side stores with standard assortment level generally records higher daily sales figure comparatively , lowest daily sales is seen in stores with basic assortment level

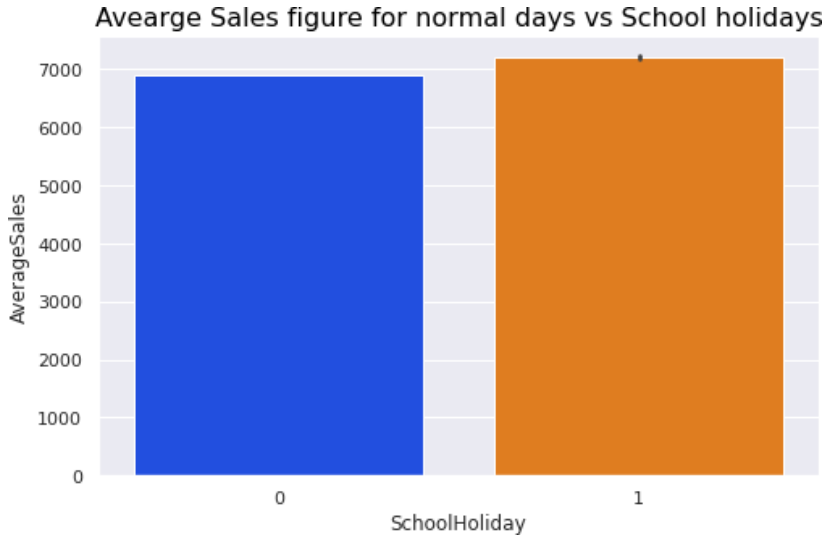
Relation between StoreType and Assortment



The multi bar plot above shows that only stores of type **b** has some stores with standard level assortment and that too very few in number and stores of all other store types has stores with only basic and extended level assortment

Relation between Holidays and Sales



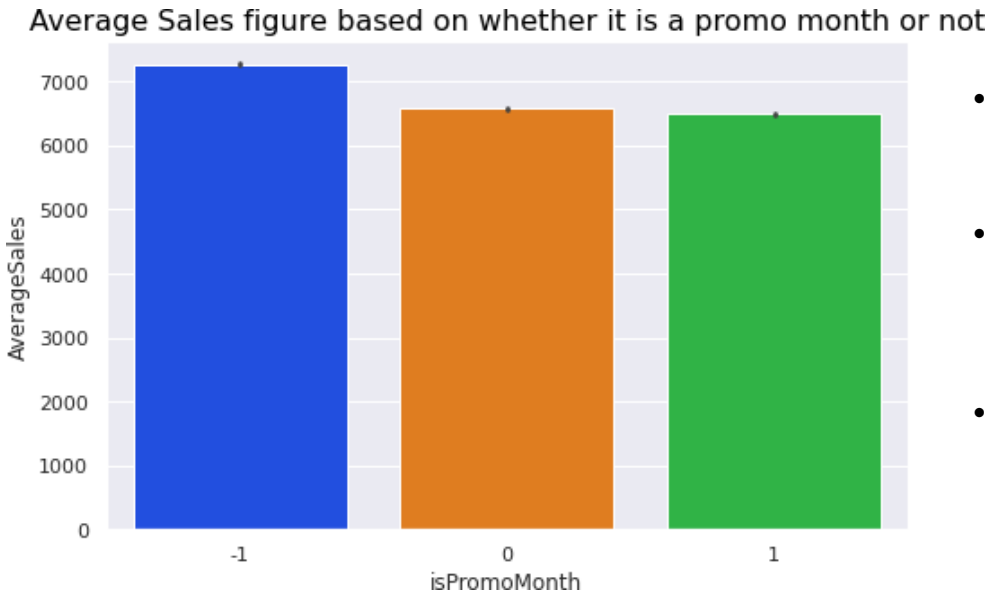


0 :- Not a State Holiday a :- Public Holiday
b :- Easter Holiday c :- Christmas

Highest daily sales figure is generally seen on Easters followed closely by Christmas . We can also notice a clear trend that bigger the holiday higher the Daily sales

Although the difference is not quite huge, the bar plot suggests that school holidays does boost daily sales figure

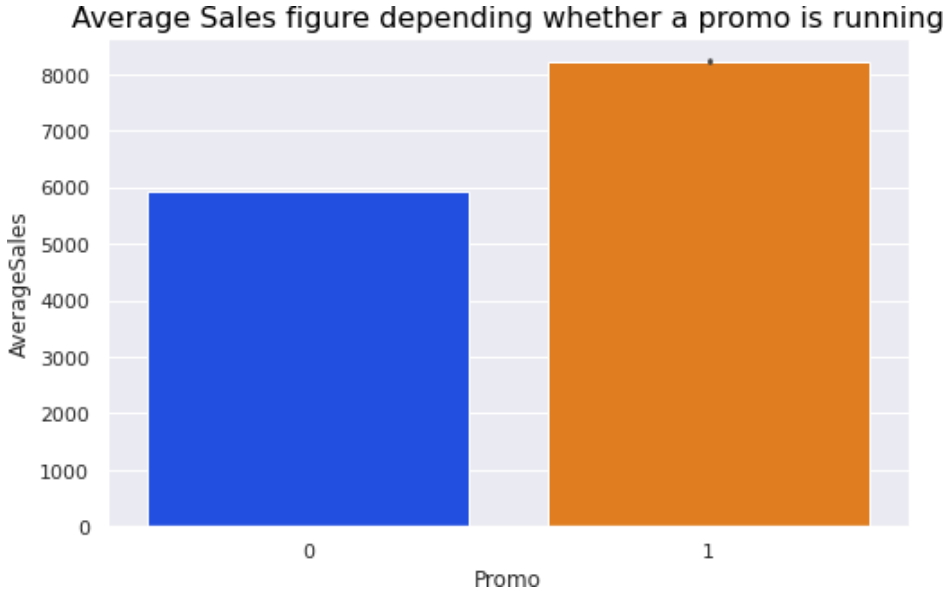
Relation Between Promo2 and Sales



-1 :- Store doesn't participates in Promo2
1 :- Store participates in promo2 but a new round doesn't starts in the month
2 :- Promotion i.e Promo2 is started anew in the month

- We can notice some kind of negative correlation between Promo2 and Daily Sales figure
- Stores which doesn't participates in Promo2 generally registers higher Daily sales as compared to the participating stores
- Also for the months in which a new round of Promo2 starts at the stores we can notice slightly lower Daily Sales figure

Relation Between Promo and Sales



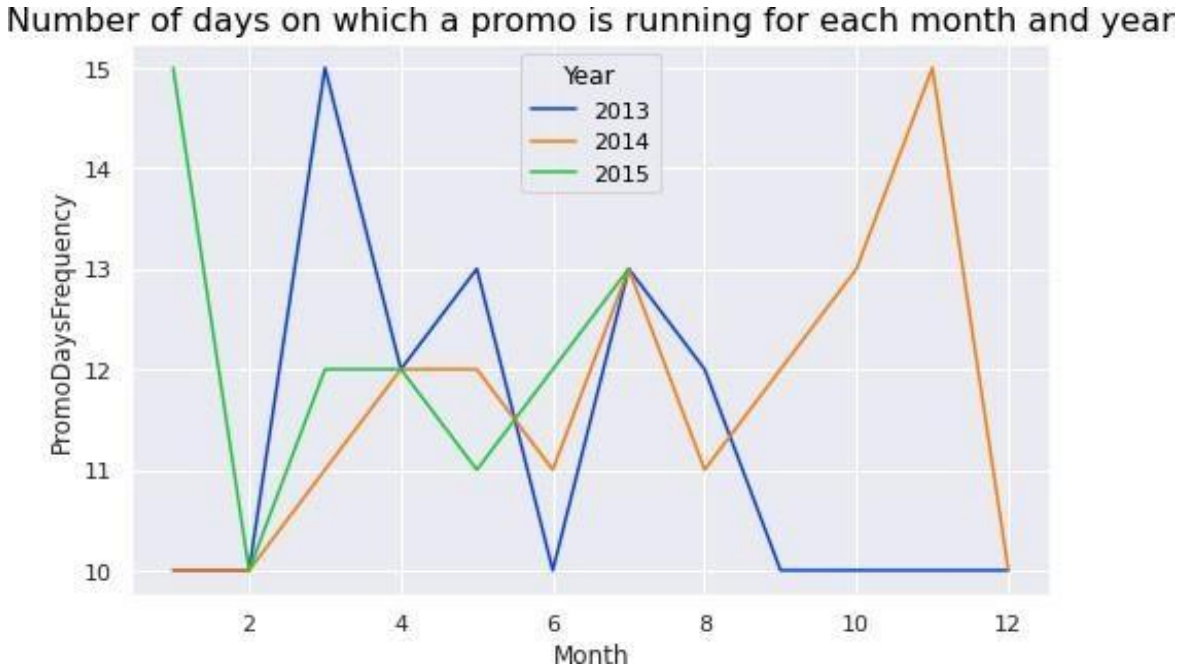
Running some kind or promo on a day can be seen giving significant boost to Sales at the stores

So people generally buy more when they are offered with attractive discounts or offers

1 :- Store is not running any promo on the day

2 :- Store is running a promo on the day

Frequency of Promo days each month

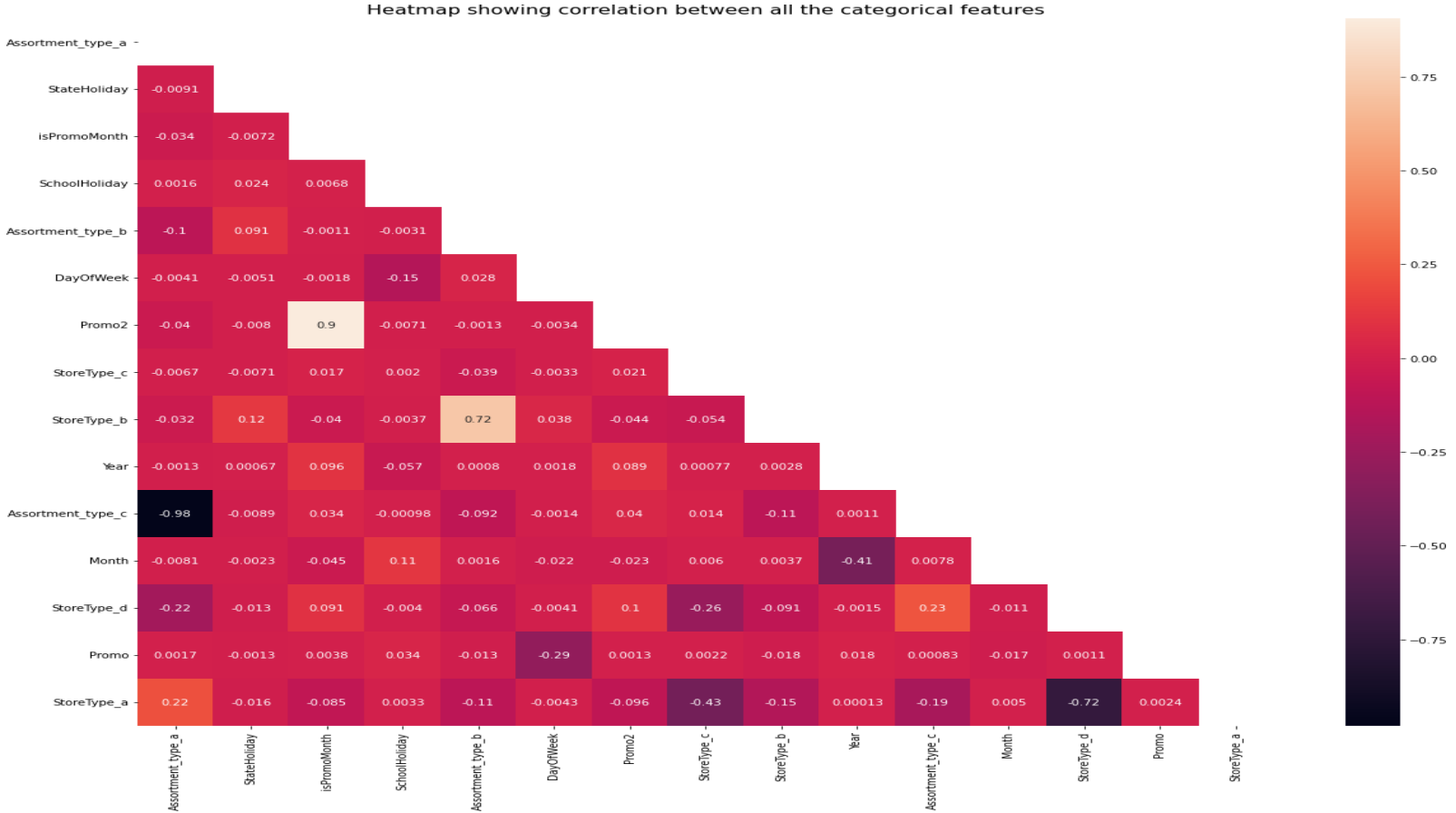


Our database has only data till 7th month of 2015 and that is why our line for 2015 terminates at July . Apart from July having 13 promo days each year there is no specific pattern in number of promo days

Customers vs Sales Correlation

Customers and sales can be seen having very strong correlation of 0.82 , which basically means the proportion by which customers count increases th sales at the store rises too by almost same proportions

Correlation Heatmap



Feature Selection

- From categorical variables due to multicollinearity issues we removed **Store_Type_a** , **Assortment_level_a** and **Promo2** from our list of features to train on
- For numerical features we calculated VIF values for each of them and at the end we got 4 numerical features with VIF value below 5 for each of them

	variables	VIF
0	CompetitionDistance	1.355844
1	NumberOfMonthsFacedCompetition	1.786912
2	MonthsOfPromo2	1.387000
3	2MonthAgoSale	2.198422

- As for **Open** column we only had values of “1” in our dataset , so we removed this too as it didn’t provided any new information and was constant for all of the rows

Preparing Dataset for modelling

Task : Regression

Train Set:- (709779,25)

	Store	DayOfWeek	Date	Sales	Customers	Open	Promo	StateHoliday	SchoolHoliday	CompetitionDistance	Promo2	NumberOfMonthsFacedCompetition
55547	373	5	2013-03-01	4074	308	1	0	0	0	11120.0	1	31.0
55548	369	5	2013-03-01	6643	542	1	0	0	0	0.0	0	0.0
55549	370	5	2013-03-01	7152	645	1	0	0	0	8250.0	1	149.0
55550	371	5	2013-03-01	5774	473	1	0	0	0	1970.0	0	44.0
55551	372	5	2013-03-01	8214	759	1	0	0	0	4880.0	0	31.0

Test Set:- (78864,25)

	Store	DayOfWeek	Date	Sales	Customers	Open	Promo	StateHoliday	SchoolHoliday	CompetitionDistance	Promo2	NumberOfMonthsFacedCompetition
765474	371	5	2015-05-08	7070	553	1	1	0	0	1970.0	1	70.0
765475	373	5	2015-05-08	4868	330	1	1	0	0	11120.0	1	57.0
765476	380	5	2015-05-08	15774	1400	1	1	0	0	2240.0	1	24.0
765477	375	5	2015-05-08	12012	979	1	1	0	0	15710.0	1	27.0
765478	376	5	2015-05-08	9117	1023	1	1	0	0	160.0	0	33.0

Applying Model (Baseline Model)

	scoring	Value
0	adjusted_R2	0.788448
1	root_mean_squared_error	1402.49
2	mean_squared_error	1966985.792867
3	mean_absolute_error	1029.125274
4	r2_score	0.788493
5	mean_squared_log_error	0.040001
6	median_absolute_error	811.107799
7	max_error	31013.694768
8	mean_absolute_percentage_error	0.159545

Our Baseline Model is Ridge Regression as can be seen in the pic on the left hand side it is giving us goodish **R2** score of **0.78** and a **MSE** score of **1402**

Model Validation and Selection

Model_Name	adjusted_R2	root_mean_squared_error	mean_squared_error	mean_absolute_error	r2_score	mean_squared_log_error	median_absolute_error	max_error
Ridge Regression	0.785782	1411.3	1991770.672538	1013.538104	0.785828	0.039959	768.861825	31335.572593
Decision Tree	0.843646	1205.72	1453758.734337	844.050406	0.843680	0.027673	617.538419	32764.399970
Gradient Boost	0.843909	1204.71	1451314.436266	851.995871	0.843942	0.030074	629.815354	31846.790044
Random Forest	0.874134	1081.8	1170285.984320	771.309379	0.874161	0.021699	574.771499	32869.461370

Model Validation and Selection (continued)

Observation 1: As seen in the table above, Ridge regression is not giving great results, Decision tree, on the other hand, gave better results than Ridge regression.

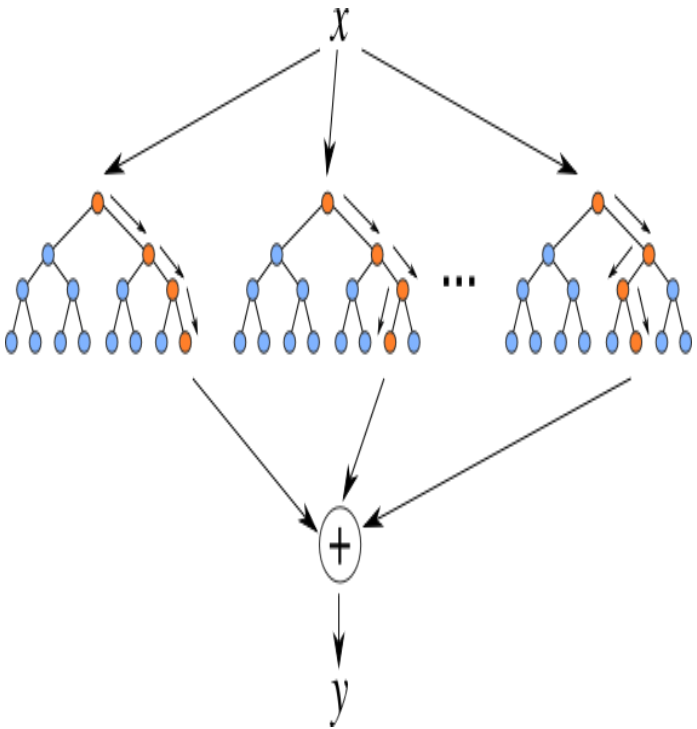
Observation 2: Gradient boost has worked quite well in comparison to Ridge regression and Decision tree in terms of R2 score and root mean error. Random forest regressor on the other hand has performed the best in all the models that we tried.

Conclusion: Based on the above observations we have chosen Random forest as our model for performing predictions.

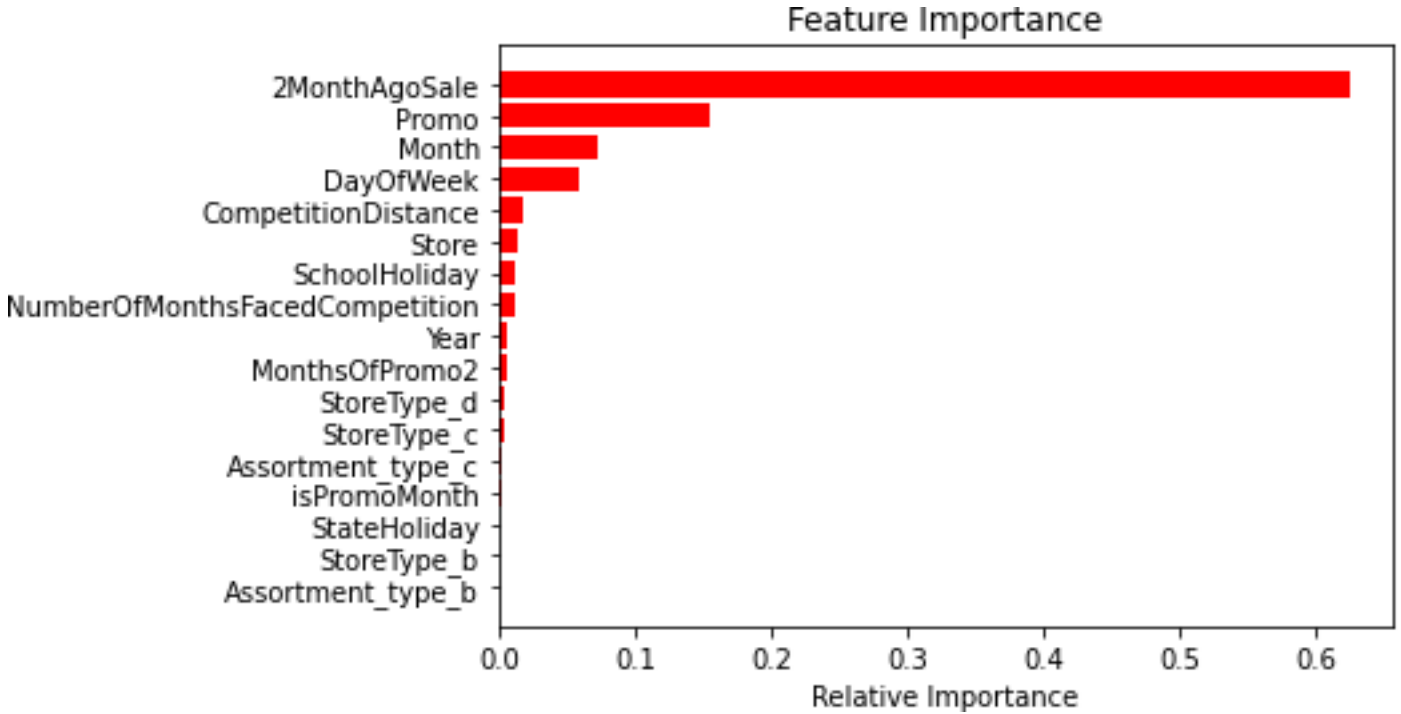
Model validation and Selection (continued)

We have chosen Random forest Regressor for our predictions and the best hyperparameters obtained are as below.

n_estimators = 125
criterion = ‘squared_error’
max_depth=18
min_samples_split = 2
min_samples_leaf = 1
max_features = ‘auto’
max_leaf_nodes = None
max_samples = None
min_impurity_decrease = 0.0
min_impurity_split = None,
min_weight_fraction_leaf = 0.0

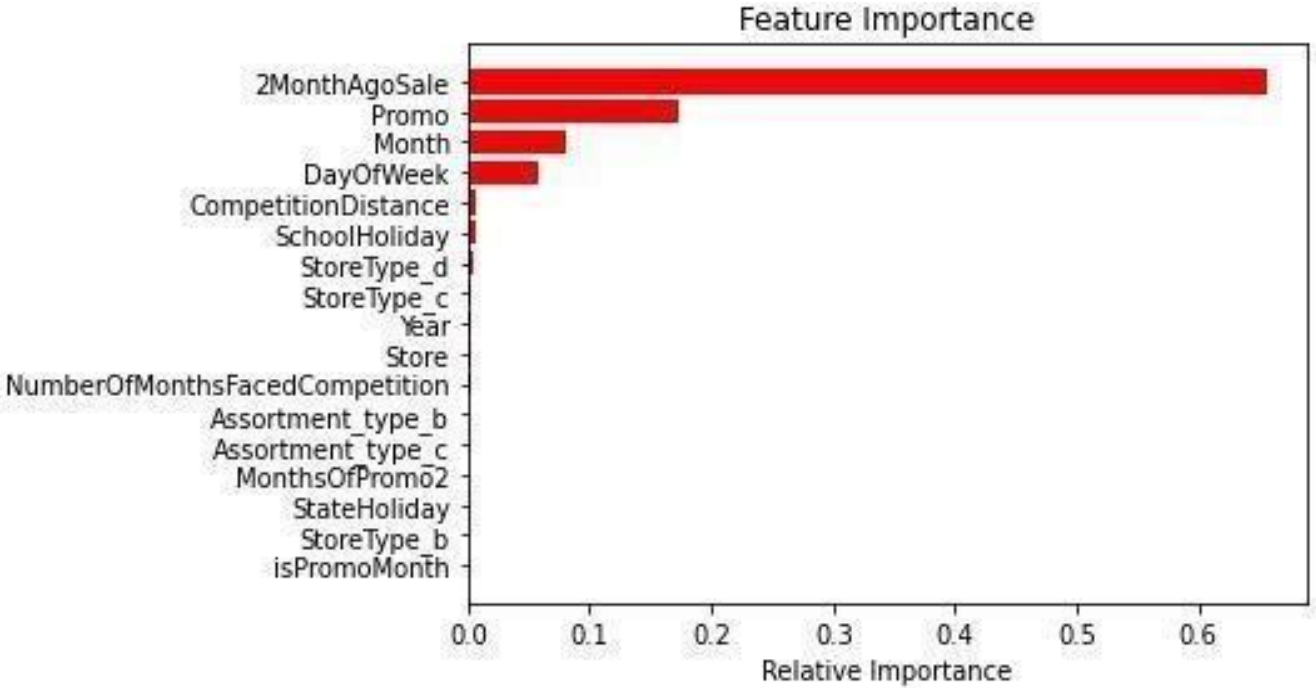


Model validation & Selection (Feature Importance)



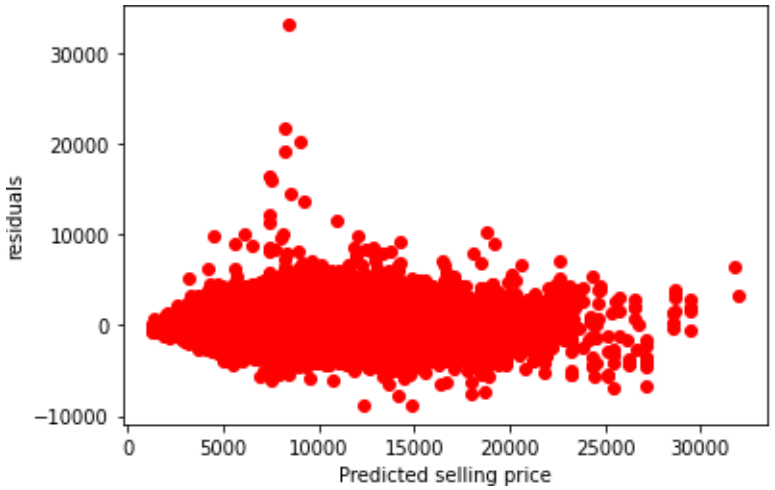
Random Forest Regressor

Model validation & Selection (Feature Importance)

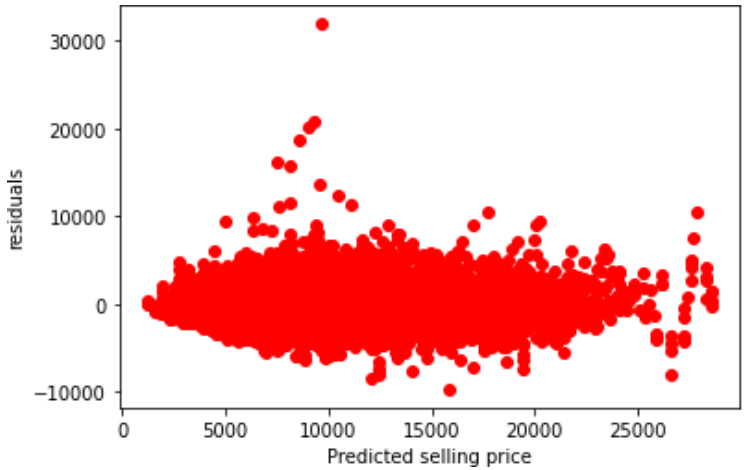


Gradient Boosting

Model validation & Selection (heteroscadacity Plot)

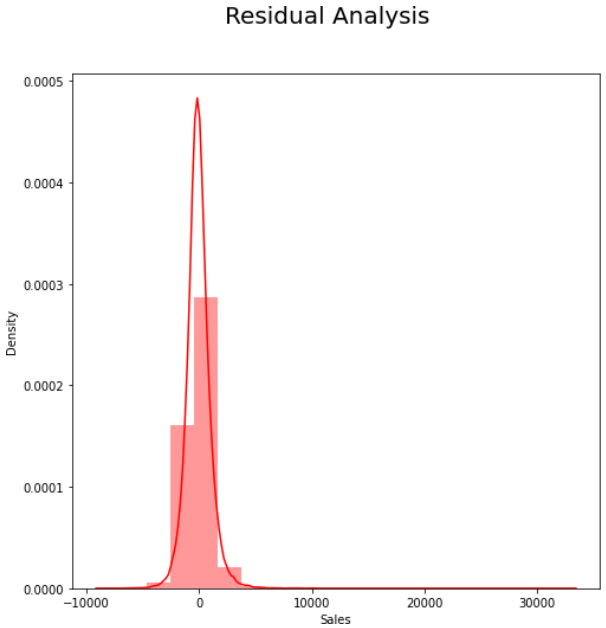


Random Forest Regressor

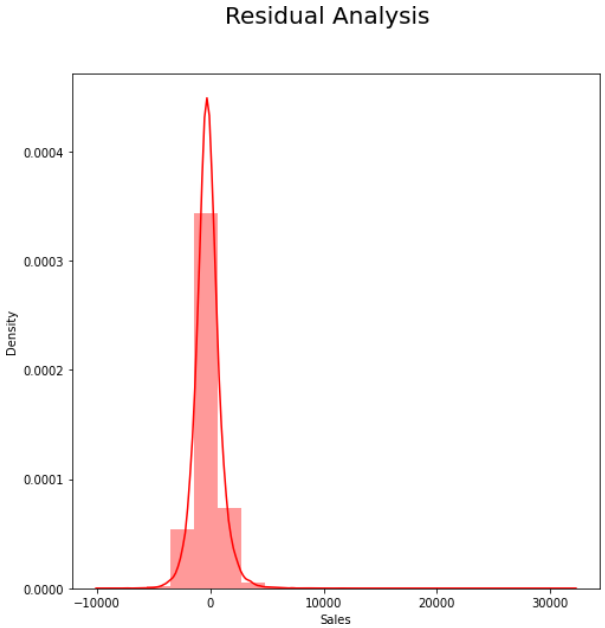


Gradient Boosting

Model validation & Selection (Residual Analysis)



Random Forest Regressor



Gradient Boosting

Conclusion

- Past sales data denoted by 2MonthAgoSale, Promo, Month, DayOfWeek and CompetitionDistance are the 5 major features for predicting the daily sales of rossmann stores
- Sunday and Monday are the two days of the week with highest Daily Sales figures
- In the final few months of each year the Daily Sales at the stores increases by significant amounts as compared to rest of the months
- With each passing year our daily Sales at the stores is slightly increasing
- Store of Type b coupled with assortment level of standard generally registers much higher daily sales number as compared to other store types with different assortment levels but such stores are very few in number
- Out of all the State Holidays daily sales are highest in Easter followed by Christmas
- Running some kind of promo for a day at the stores significantly boosts its sales count
- Promo2 is not helping at all to push the daily sales at the stores

Conclusion

- Random Forest model predicted the daily sales for the rossmann stores with the best accuracy . The RMSE scores were below 1100 and Adjusted R2 was above 0.87 . The features that we used in our model can predict the daily sales at the stores for up to 2 months in advance
- Although we used traditional sklearn models for our prediction like decision tree, gradient boost and random forest but there are models specialized in handling time series data such as Moving Average (MA) , Autoregressive Moving Average (ARMA) and others . Also we use can fine tune the hyperparameters even more for better overall predictions
- Using the rossmann stores prediction , the store managers at rossmann can get an estimate of their revenue generation for up to 2 months in advance the data which then can be used to make other informed decision to further amp up the sales figure

Challenges

- Our dataset was quite huge with a number of features so it was important to handle and analyse each and every feature carefully not to miss anything which is of any relevance
- As we had a time series data so we had to further refine some of the feature values based on the date of which they are
- Feature engineering was quite challenging and important as we needed to find different ways to better use values from the past that can be used to predict the future sales and not to use any futuristic features while predicting
- The computational time was huge