RETAIL SALES PREDICTION

Capstone Project II

By:
Neel Naik
Prasad Khadatkar
Piyush Nirwan
Shivank Shukla
Saransh Jain

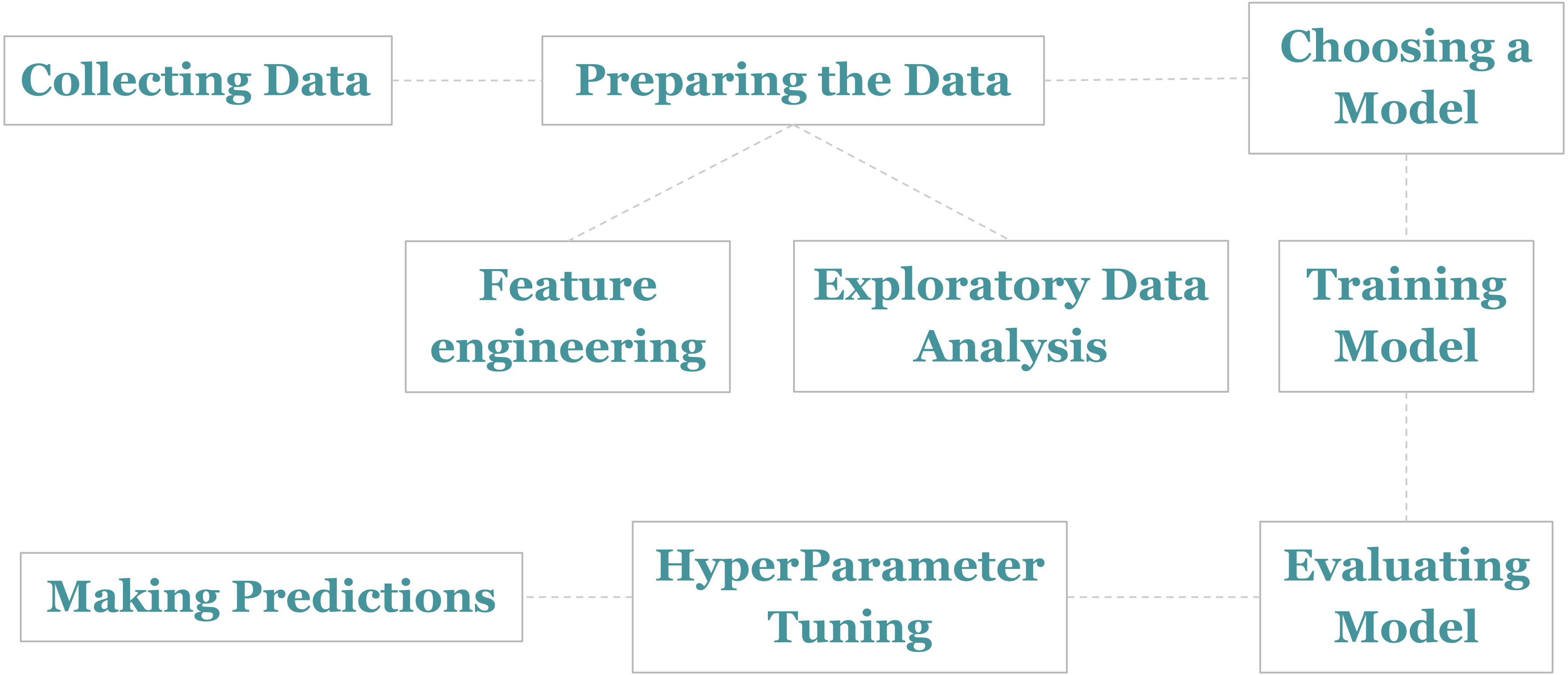
About the Company

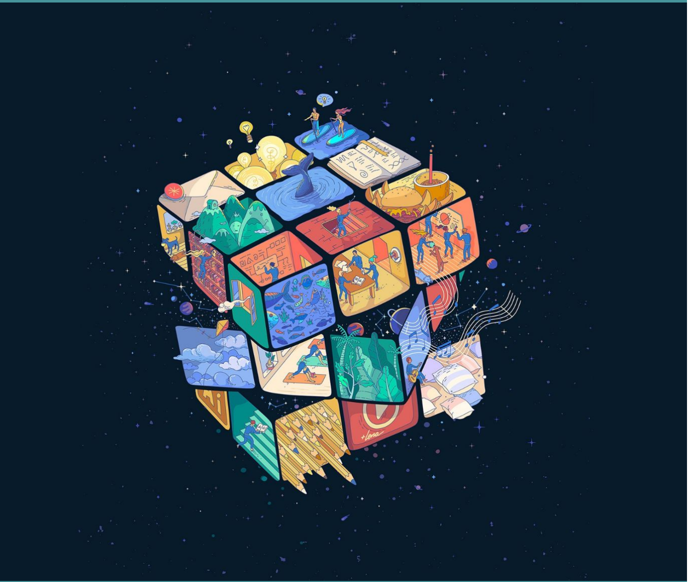
- **German drugstore chain Rossmann sells a range of goods:**
  - Cosmetics
  - Health and wellness items,
  - Home goods, and more.
- **The corporation runs approximately 3,600 stores in several nations across Europe:**
  - Including Germany
  - Poland
  - The Czech Republic
  - Slovakia
  - Hungary
  - Croatia, and others.
- **Additionally, they could provide services like:**
  - Photo printing
  - Pharmacy assistance
  - and other healthcare-related services.

Problem Statement

- Predicting their **daily sales for up to six weeks in advance.**
- Store sales are influenced by many factors, including **promotions, competition, school and state holidays, seasonality, and locality.**
- Historical sales data for **1,115 Rossmann stores.**
- **Forecast the "Sales" column for the test set.**

Approach


Collecting Data
Preparing the Data
Choosing a Model
Feature engineering
Exploratory Data Analysis
Training Model
Making Predictions
HyperParameter Tuning
Evaluating Model

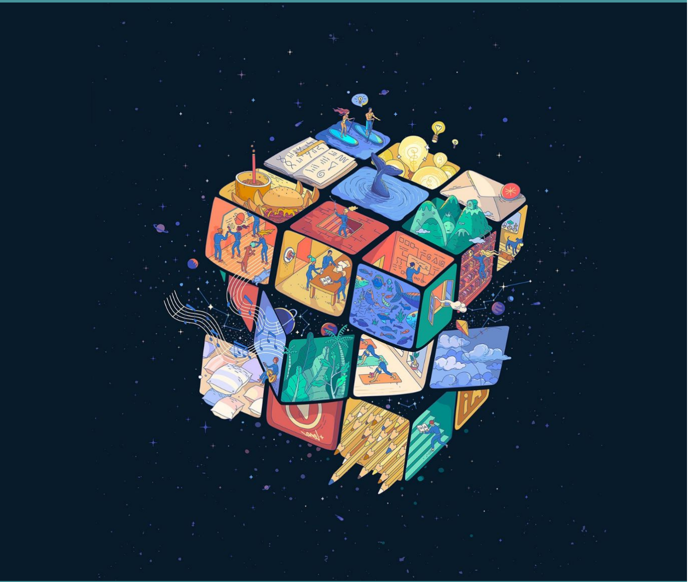


Understanding Dataset

We have been provided with 2 data sets.

1) Rosemann store Data: Information about sales and related factors

- **Store**: Unique Store Id.
- **DayOfWeek**: No. of day of the week.
- **Date**: Current Date of the day.
- **Sales**: No. of sales of the day.
- **Customers**: footfall of the day.
- **Open**: Store is open or closed.
- **Promo**: Store running promotion or not.
- **StateHoliday**: State holiday or not.
- **SchoolHoliday**: School holiday or not.



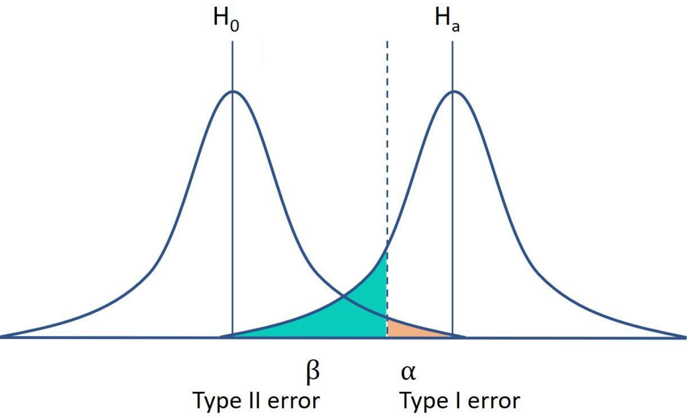
Understanding Dataset

2) Store: Information about the store

- **Store:** Unique Store Id.
- **StoreType:** 4 different type of stores a,b,c,d.
- **Assortment:** A collection of goods or services that a business provides to a consumer.
- **CompetitionDistance:** Distance in meters to the nearest competitor store.
- **CompetitionOpenSinceMonth:** Month in which the competition store was open.
- **CompetitionOpenSinceYear:** Year in which the competition store was open.
- **Promo2:** Store running consecutive promotion or not.
- **Promo2SinceWeek:** Calendar week when the store started participating in Promo2.
- **Promo2SinceYear:** Year when the store started participating in Promo2.
- **PromoInterval:** The month in which the promotion starts eg: Jan,Apr,Jul,Oct.

Exploratory Data Analysis

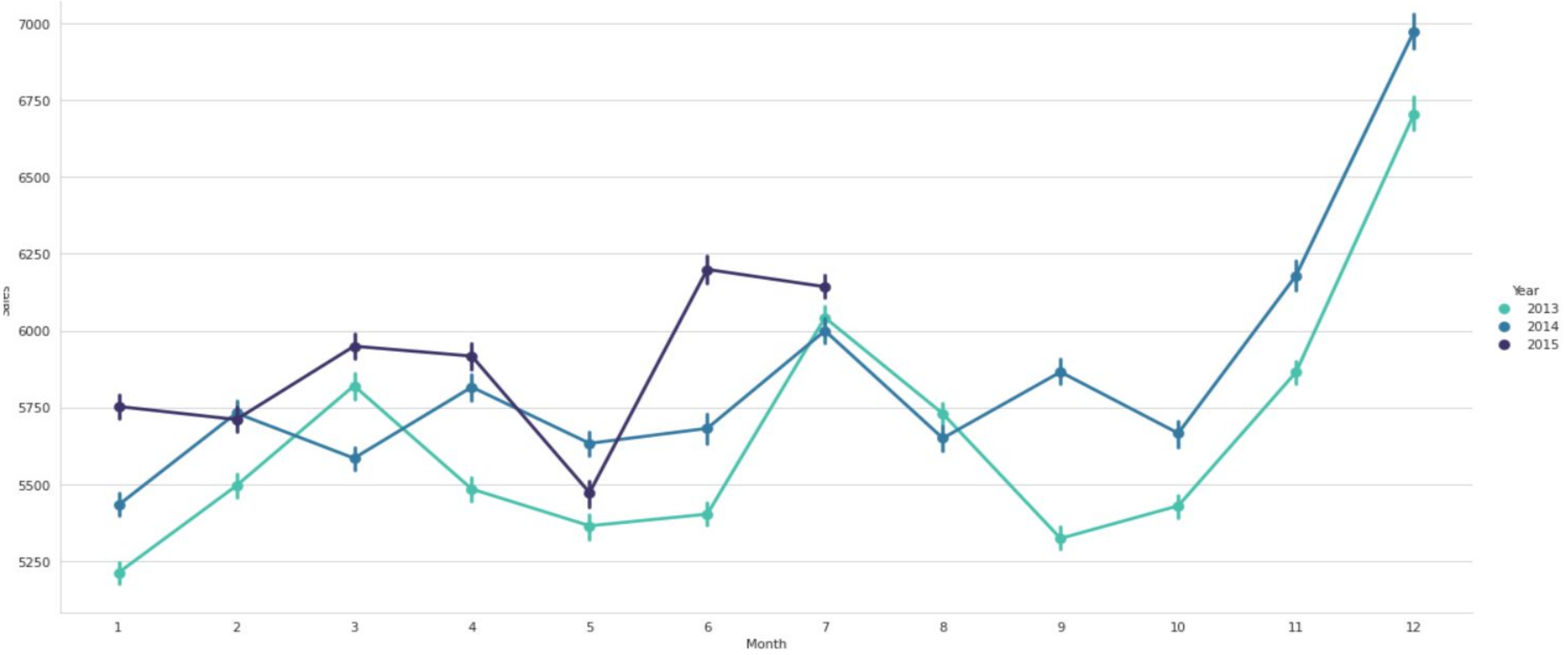
- Exploratory data analysis (EDA) is a process of analyzing and summarizing a dataset in order to **better understand its properties and characteristics**.
- It is an iterative process that involves **visualizing** and summarizing the data, **identifying patterns** and relationships, and **testing hypotheses** about the data.
- It helps researchers to **gain insights** into the data, identify potential **issues** or problems that can be tested through further analysis.



Hypothesis testing

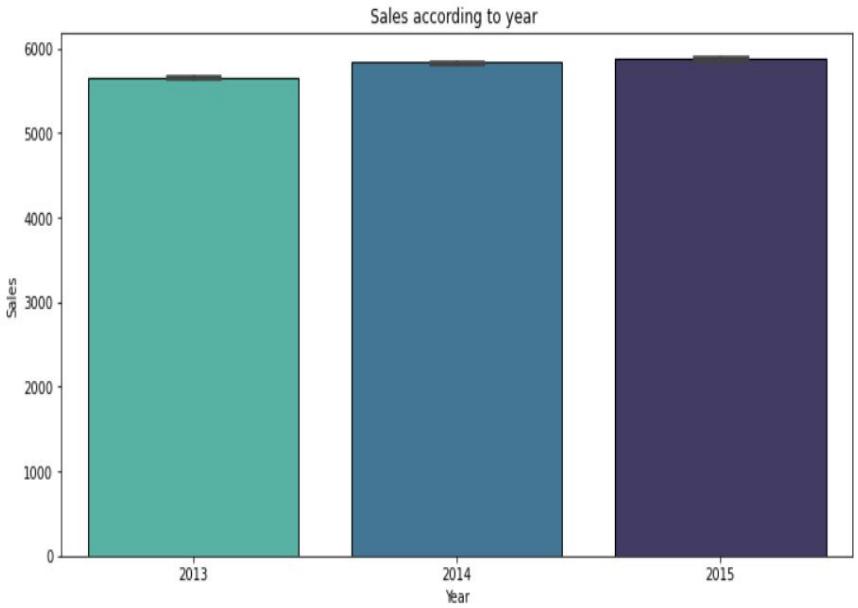
- Due to the high number of **public holidays in December,** sales will be at their highest.
- Due to **weekends, sales** ought to be at their peak on Saturday or Sunday.
- **Sales and promotion** ought to be closely related in a favourable way.
- Due to its **small number** of stores, **Store B** will have the lowest sales.
- The aggregate sales are increased when **competitors are close** to one another.

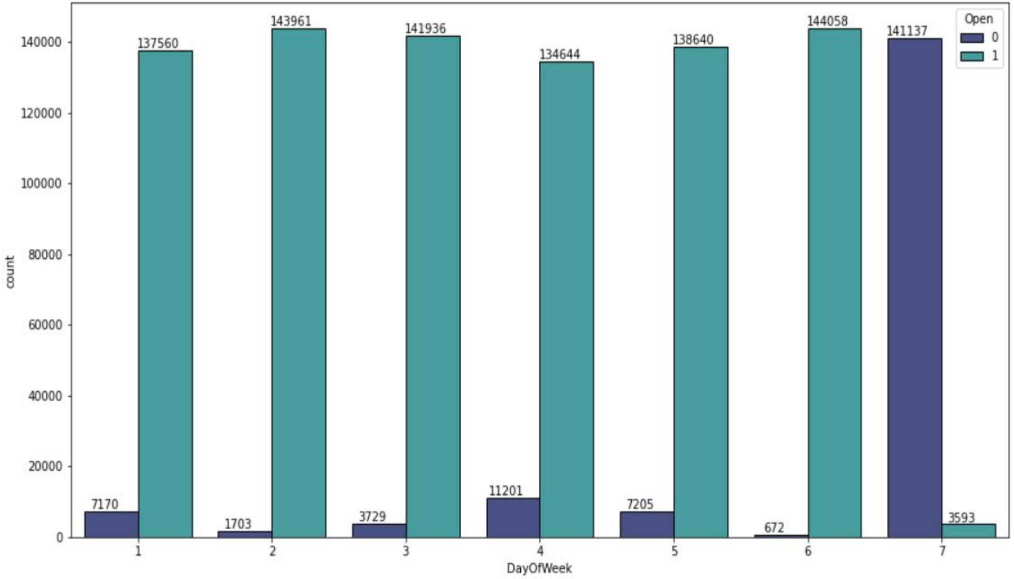
EDA



- Here the trend shows that the **sales increase** significantly in the **month of October to December** due to the holiday season.
- From the chart we can see that there Are roughly **3 cycle of sales**.

EDA

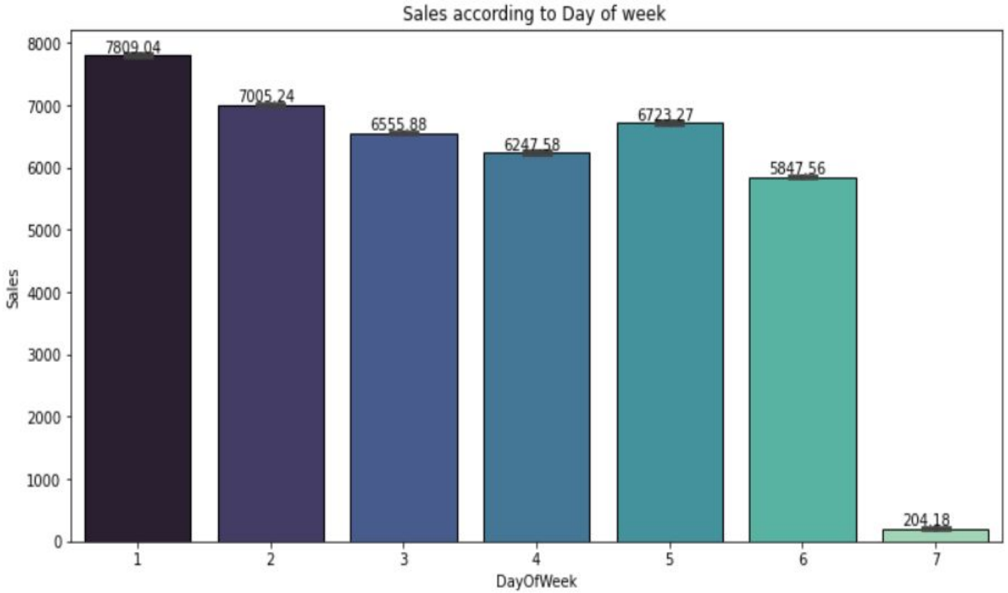


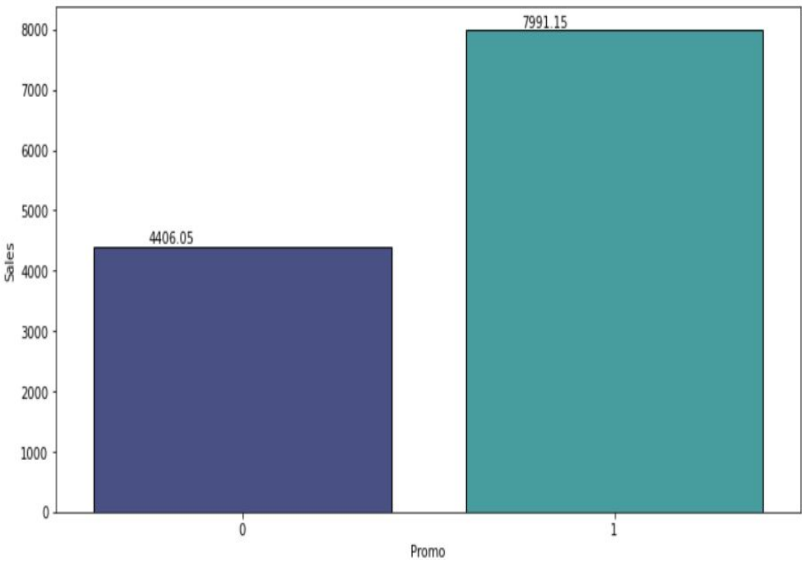


- From above chart we can see that there is **YoY increase** in sales from 2013-2015
- Despite having data available for 7 months in **year 2015**.It has already **crossed the sales of 2014.**

- This plot shows open and close of the shop on days of the week.
- Here, the store in open for **maximum** no. of days **on Saturday** and **minimum** no. of days **Sunday**.

EDA

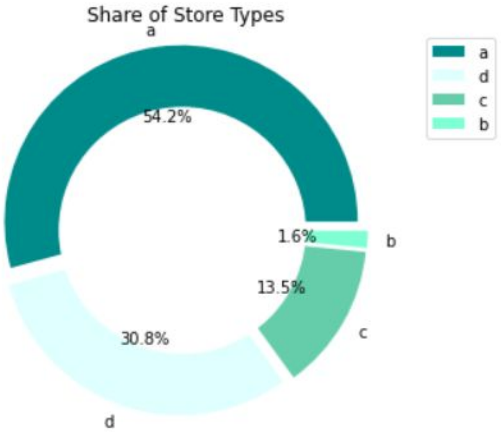


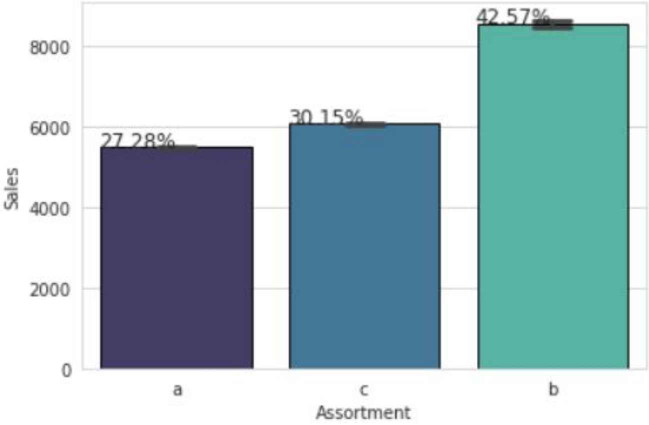


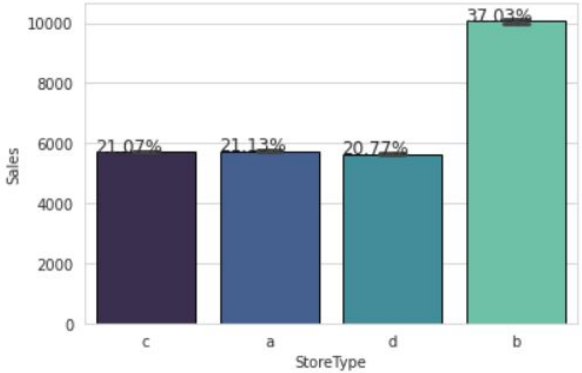
- As **sundays** has the most **store closed** so it has the **least** number of **sales**
- On the other hand **mondays** have the **maximum** number of **sales**
- **Saturday** Despite having the maximum number of stores open still have **third least sales** numbers.

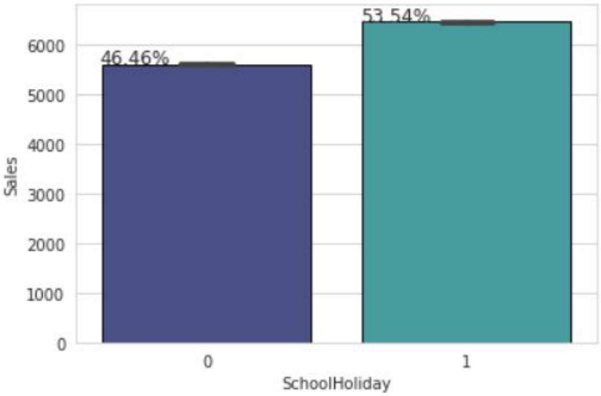
- Store who participating in promotion having more sales as compare to other.
- The **Sales** get almost **increases by 100 %** when **promo** takes place.

EDA



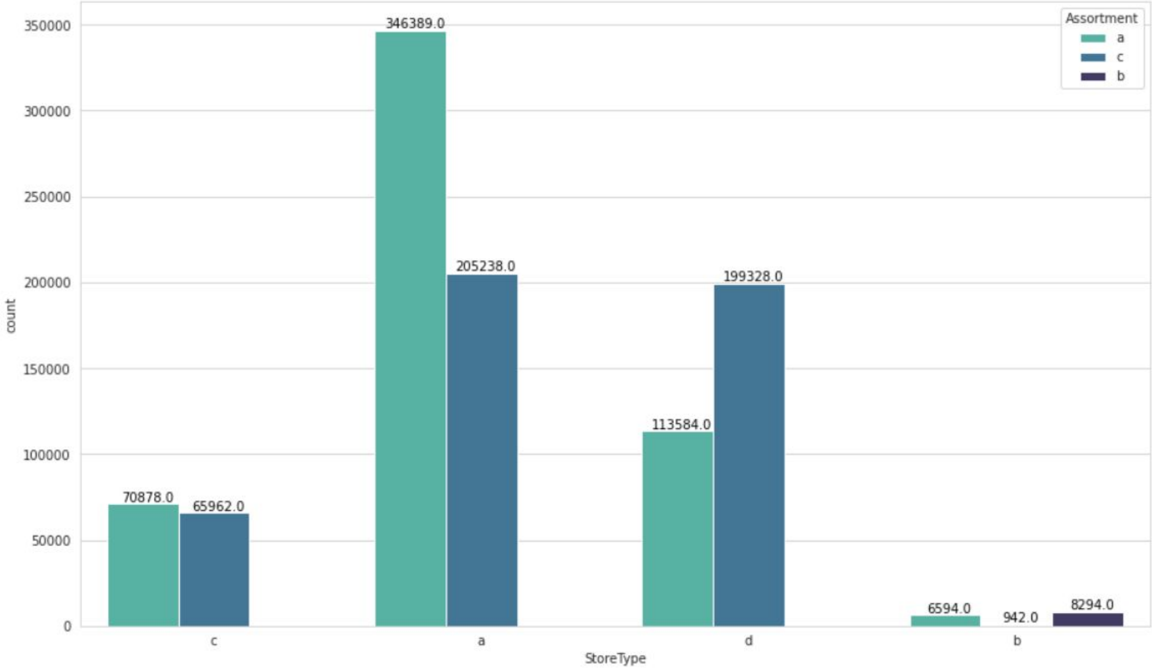


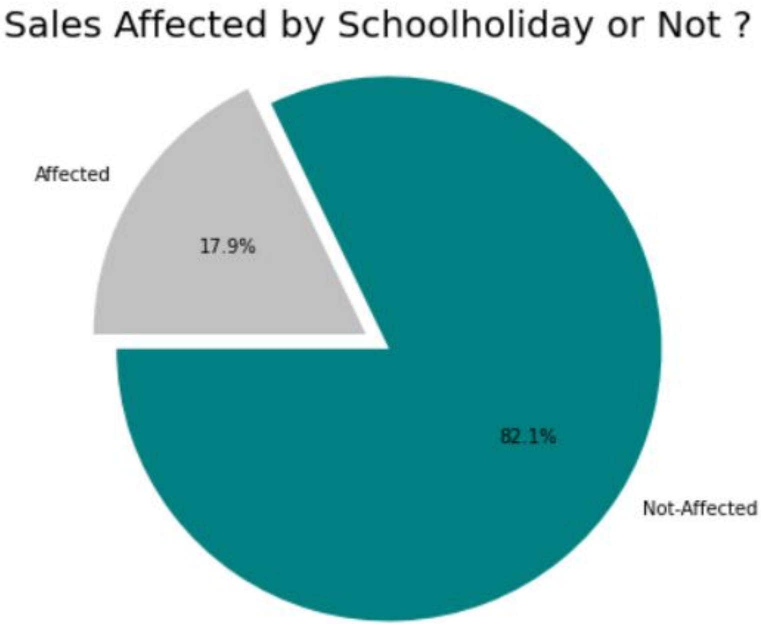




- **Highest sales** belonged to the store **type a** due to the high number of type a stores in our dataset.
- Almost **50% of school holidays** there were **stores open** resulting in sales.
- Store **type b** with **highest average sales** and per store revenue generation looks healthy as all three kinds of assortment strategies involved which was seen earlier.
- **Maximum** sales are from **store a i.e. 54.2%**
- **Minimum** sales are from **store b i.e. 1.6%**

EDA

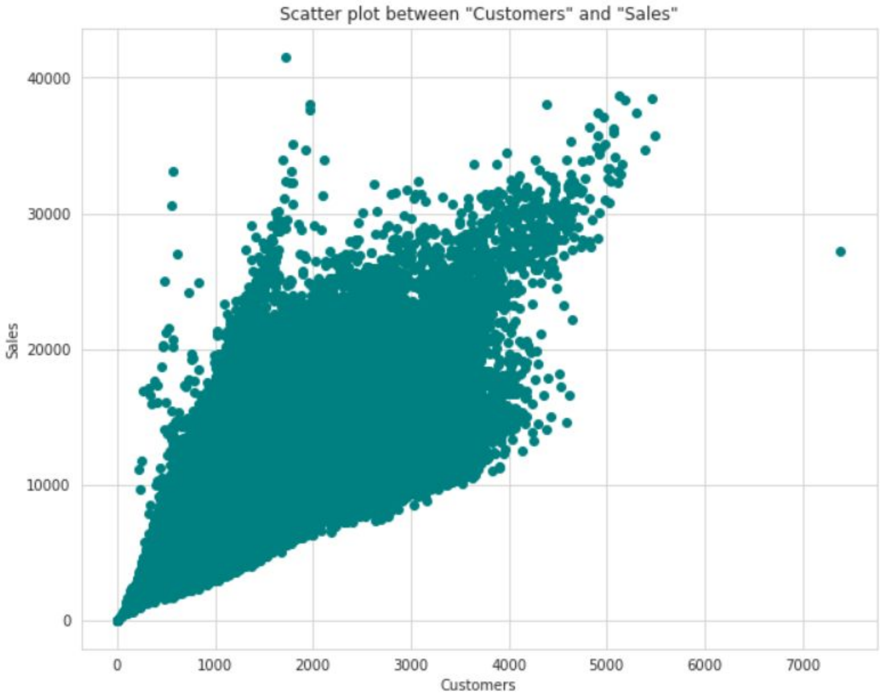


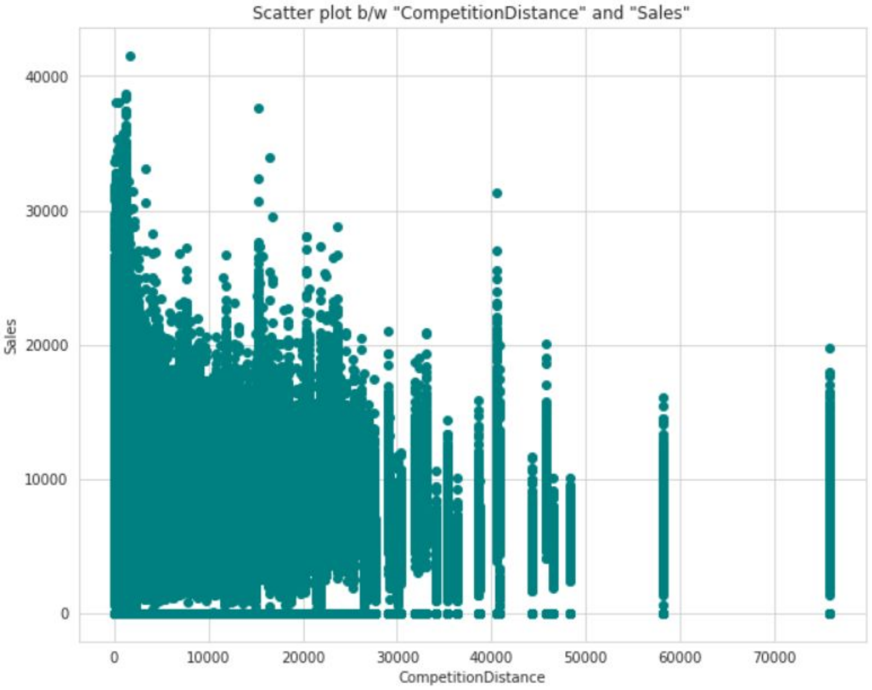


- Despite being scarce, **store type B** had the **greatest average sales.**
- The three types of assortments, especially **level B**, which is exclusively **sold at type B** stores, and the fact that the stores are open on Sundays are among the reasons.

- Only **18% Sales** are **affected** during school holiday. Rest 82% of sales are not affected by the School holidays.

EDA

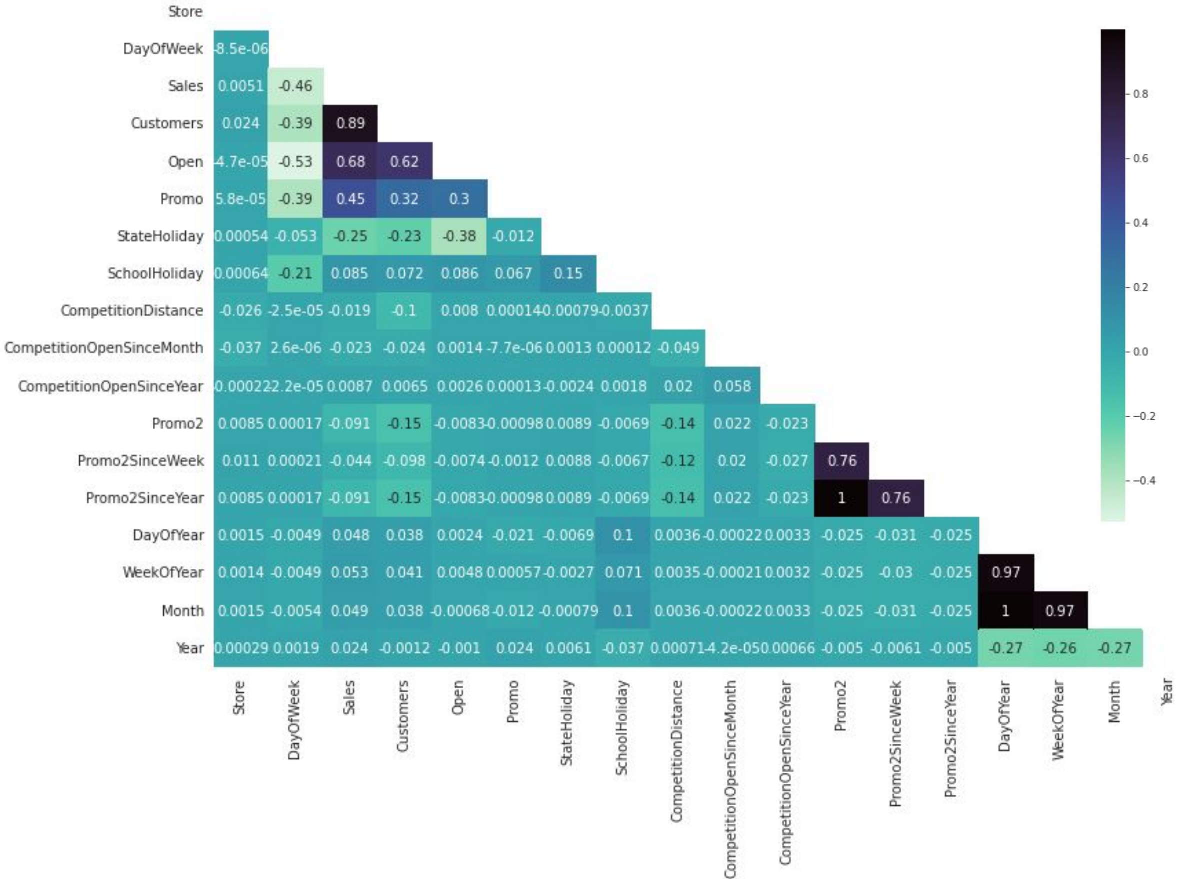




- **Positive** relation between no. of **Customers and Sales**.
- Linear regression with **high variance** & **few outliers.**

- As the **distance** between the competition **increases the sales decreases.**
- After certain distance (30,000) correlation between Competition Distance and Sales is very vague.

EDA



Positive Correlation

- **Day of the week** has a **negative correlation** indicating low sales as the weekends, and **promo, customers and open** has **positive correlation.**
- Customers and sales has the most positive correlation of **0.84**
- Followed by open and Sales with correlation of **0.68**

Negative Correlation

- Open and Days of week has most negative correlation of **-0.53**
- **Competition Distance** showing **negative correlation** suggests that as the distance increases sales reduce, which was also observed through the scatterplot earlier.

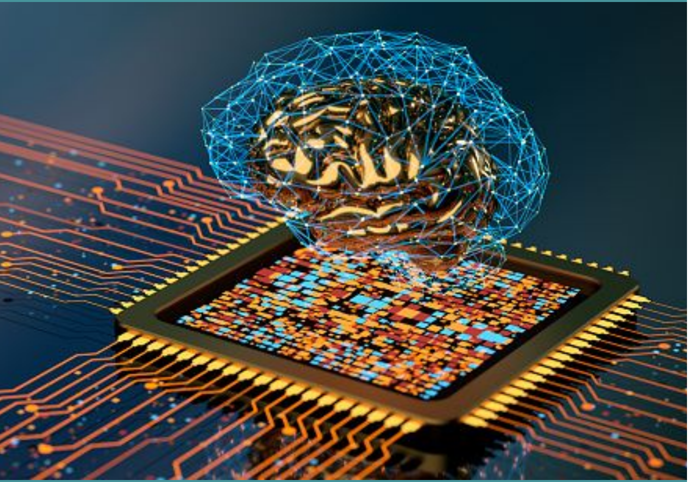
Multicollinearity

	variables	VIF
8	Month	564.101464
6	DayOfYear	514.428005
7	WeekOfYear	61.181227
9	Year	24.985713
0	Customers	6.418616
15	DayOfWeek_7	3.115522
16	StoreType_b	2.385302
14	DayOfWeek_6	2.330946
1	Promo	2.283923
5	Promo2	2.152976
19	Assortment_b	2.106967
20	Assortment_c	2.046538
11	DayOfWeek_3	2.027134
12	DayOfWeek_4	2.026790
10	DayOfWeek_2	2.023992
13	DayOfWeek_5	2.015608
18	StoreType_d	1.760616
4	CompetitionDistance	1.622251
3	SchoolHoliday	1.375732
17	StoreType_c	1.263415
2	StateHoliday	1.245737

	variables	VIF
0	Customers	4.309367
6	DayOfYear	3.254634
13	StoreType_b	2.348660
1	Promo	2.162611
16	Assortment_b	2.105486
17	Assortment_c	2.030084
5	Promo2	1.909813
15	StoreType_d	1.679569
11	DayOfWeek_6	1.654362
10	DayOfWeek_5	1.631738
7	DayOfWeek_2	1.629335
9	DayOfWeek_4	1.622604
8	DayOfWeek_3	1.604201
12	DayOfWeek_7	1.539358
4	CompetitionDistance	1.537660
3	SchoolHoliday	1.336624
14	StoreType_c	1.244757
2	StateHoliday	1.147969

- The VIF was calculated for the features in the DataFrame.
- At **every step** the **variable** with **highest VIF** value was **dropped**.
- And the VIF value was calculated again.
- Until the **value** was **under 5** for all variable.

** Before doing the multicollinearity all the categorical variables were converted into dummy variable*



Machine Learning Model

We have chosen to implement these models on our dataset:

1. Linear Regression.
   - Lasso
   - Ridge
2. Decision Tree
3. Random Forest
   - Random Forest with Optimization
4. XGBoost
   - XGBoost with Optimization.

Machine Learning Model

We will be using following Matrices to check our model performance:

1. Mean Absolute Error (MAE)
2. Mean Squared Error (MSE)
3. Root Mean Square Error (RMSE)
4. R Squared (R2)
5. Adjusted R Squared

Linear Regression


```
Train MSE: 1444076.9595
Test MSE: 1460445.7451

Train RMSE: 1201.6975
Test RMSE: 1208.489

Train MAPE: 3.965356421594898e+17
Test MAPE: 3.947954831061504e+17

Train R2: 0.9027
Test R2: 0.9027

Train Adjusted R2: 0.9028
Test Adjusted R2: 0.901
```

- Linear regression analysis is used to predict the **value of a variable** based on the **value of another variable.**
- The variable you **want to predict** is called the **dependent variable**.
- The **variable** you are **using** to predict the other variable's value is called the **independent variable**.
- The results show that a Linear Regression is performing pretty well on the validation set but it has completely overfitted the train set with a test **R^2 of 0.90.**

LARS Lasso Regression


```
Train MSE: 1445477.1383
Test MSE: 1461490.2814

Train RMSE: 1202.28
Test RMSE: 1208.9211

Train MAPE: 3.913935940440144e+17
Test MAPE: 3.896165527781138e+17

Train R2: 0.9026
Test R2: 0.9026

Train Adjusted R2: 0.9027
Test Adjusted R2: 0.9009
```

- **Coefficient is magnitude instead of squared.**
- Possibilities of many coefficients becoming zero, so that corresponding features become zero and dropped from the list.
- **Reduces the dimensions** and supports for dimensionality reduction
- This is **L1 regularization**, because of adding the **Absolute-Value** as **penalty-equivalent** to the magnitude of coefficients.

Ridge Regression


```
Train MSE: 1444076.9597
Test MSE: 1460445.8588

Train RMSE: 1201.6975
Test RMSE: 1208.4891

Train MAPE: 3.965331905716957e+17
Test MAPE: 3.94793026559066e+17

Train R2: 0.9027
Test R2: 0.9027

Train Adjusted R2: 0.9028
Test Adjusted R2: 0.901
```

- **Minimize** the sum of **squared errors** and sum of the **squared coefficients** (β).
- The **coefficients (β)** with a large magnitude will **generate** the **graph peak and deep slope**, to suppress this we're using the **lambda (λ)** use to be called a **Penalty Factor** and help us to **get a smooth surface** instead of an irregular-graph.
- Ridge Regression is used to push the **coefficients(β)** value nearing **zero** in terms of magnitude.
- This is **L2 regularization**, since it's adding a penalty-equivalent to the **Square-of-the Magnitude** of coefficients.

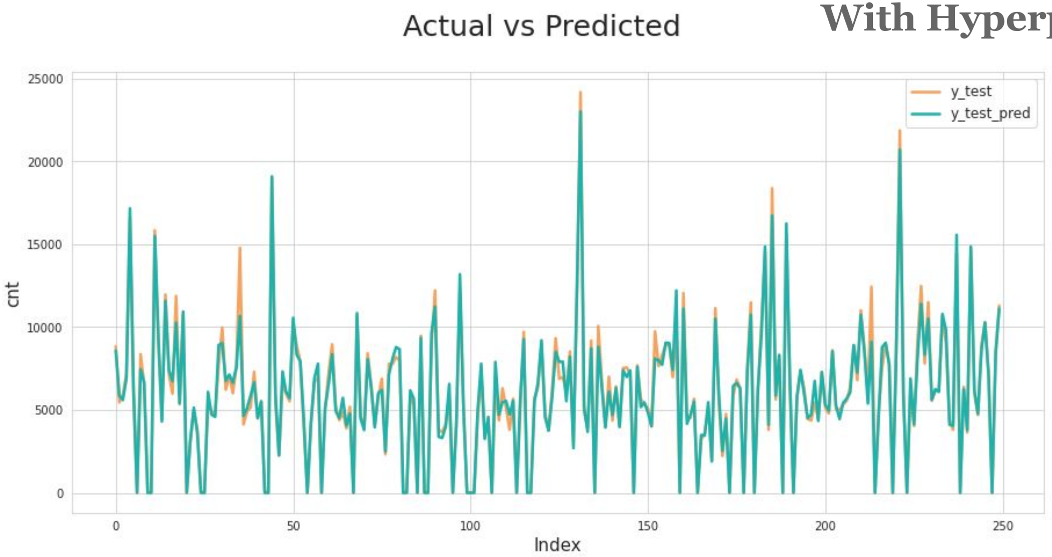
Hyperparameter tuning

Hyperparameter tuning is the process of **adjusting the hyperparameters** of a machine learning model to **optimize its performance**. They are model parameters that are set before training begins, and they can have a significant impact on the model's performance.

- **Grid search:** This involves specifying a **grid of hyperparameter values** and **training** a model for **each combination of values.** The best performing model is then selected based on a performance metric.
- **Random search:** This involves **sampling random combinations** of hyperparameter values and training a model for each combination. The best performing model is then selected based on a performance metric.

Decision Tree Regression

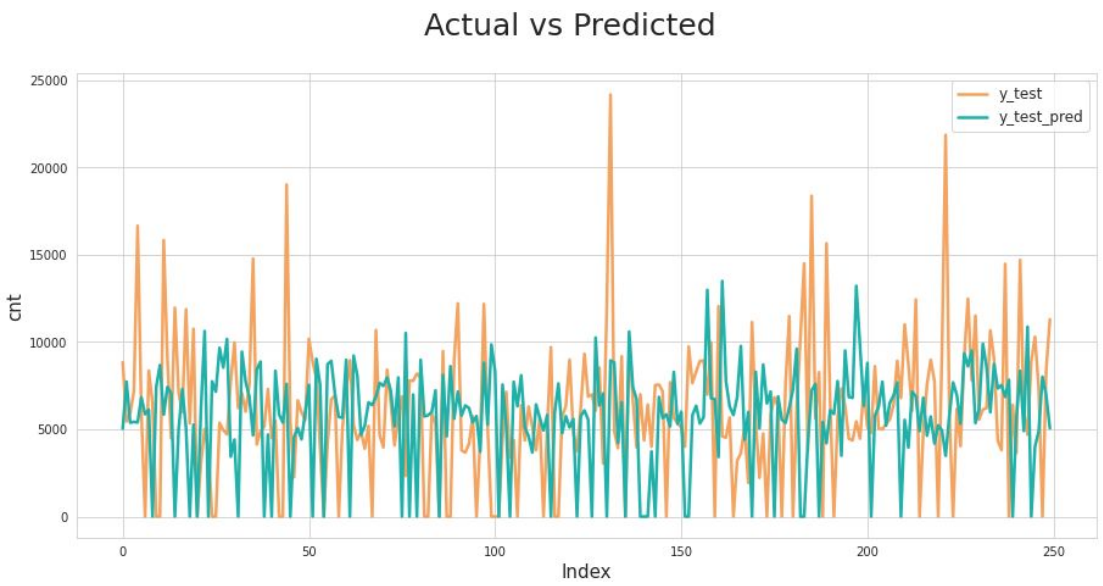
With Hyperparameter tuning




```
Train MSE: 159982.4172
Test MSE: 328917.4704

Train RMSE: 399.978
Test RMSE: 573.5133

Train MAPE: 817225993404.8306
Test MAPE: 0.0502

Train R2: 0.9892
Test R2: 0.9892

Train Adjusted R2: 0.9893
Test Adjusted R2: 0.9778
```

- It can be **used** to solve **both Regression and Classification** tasks with the latter being put more into practical application.
- It is a tree-structured classifier with **three types of nodes.**
- The **Root Node** is the initial node which represents the **entire sample** and may get split further into further nodes. The **Interior Nodes** represent the **features of a data set** and the **branches** represent the **decision rules**. Finally, the **Leaf Nodes** represent the **outcome**. This algorithm is very useful for solving decision-related problems.

K-Nearest Neighbors Regression

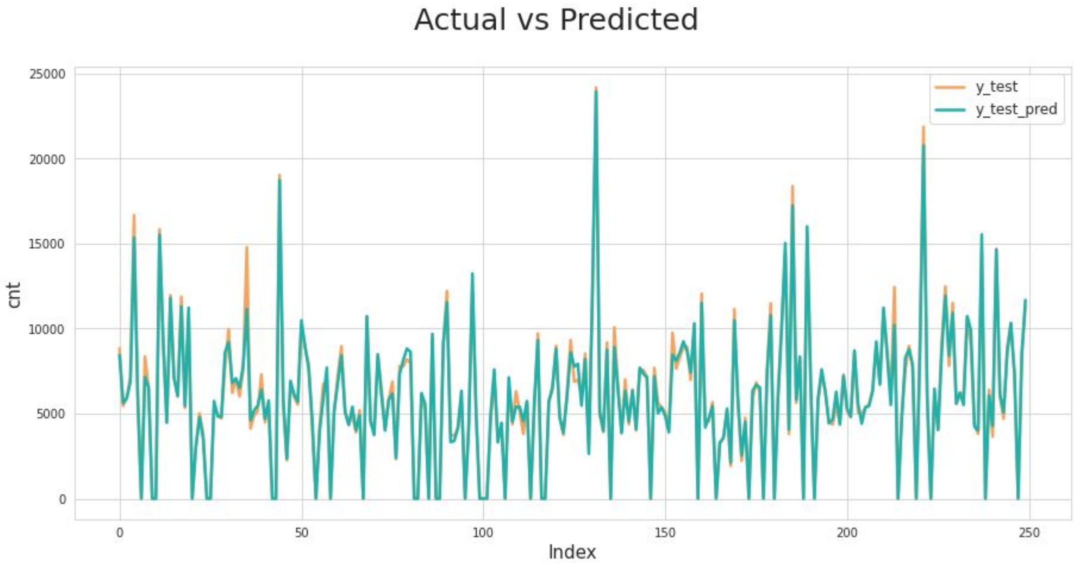



```
Train MSE: 4043078.6063
Test MSE: 4577365.8255

Train RMSE: 2010.7408
Test RMSE: 2139.4779

Train MAPE: 5.090112714282186e+16
Test MAPE: 4.561678798765004e+16

Train R2: 0.7271
Test R2: 0.6968

Train Adjusted R2: 0.7278
Test Adjusted R2: 0.6975
```

- It is a non-parametric, supervised learning regression, the output is a **continuous value rather than** a **discrete label.**
- **Looks at** the **K nearest data points** in the training set, where **K is a user-specified hyperparameter**. It then **averages** the **output values** of those K nearest data points to make the prediction for the input value.

Random Forest Regression

With Hyperparameter tuning



- Random Forest Regression is a supervised learning algorithm that uses **ensemble learning methods for regression.**
- Ensemble learning method is a technique that **combines predictions** from multiple machine learning algorithms to make a **more accurate prediction** than a single model.

Random Forest Regression

Train MSE: 153422.5643
Test MSE: 247870.7354

Train RMSE: 391.6919
Test RMSE: 497.8662

Train MAPE: 2574122356578.04
Test MAPE: 1778310030299.2266

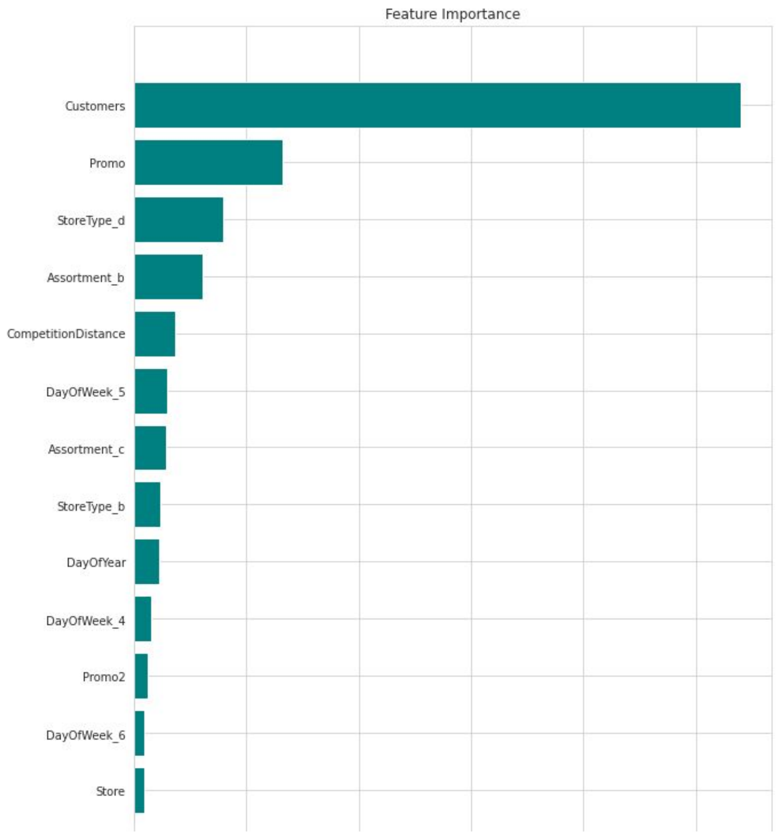
Train R2: 0.9897
Test R2: 0.9897

Train Adjusted R2: 0.9898
Test Adjusted R2: 0.9833

XGBoost Regression

- **XGBoost** is a **powerful approach** for building **supervised regression models.** The validity of this statement can be inferred by knowing about its (XGBoost) objective function and base learners.
- The objective function contains **loss function** and a **regularization term.**
- It tells about the difference between **actual values** and **predicted values**, i.e how far the model results are from the real values.
- The most common loss function in XGBoost for regression problems is reg:linear.

XGBoost Regression




```
Train MSE: 1147716.4266
Test MSE: 1164175.301

Train RMSE: 1071.3153
Test RMSE: 1078.9696

Train MAPE: 1.1149464387599122e+17
Test MAPE: 1.1025178733283864e+17

Train R2: 0.9227
Test R2: 0.9227

Train Adjusted R2: 0.9228
Test Adjusted R2: 0.9211
```

Result

	Modal Name	Train MSE	Test MSE	Train RMSE	Test RMSE	Train MAPE	Test MAPE	Train R2	Test R2	Train Adjusted R2	Test Adjusted R2
0	Linear Regression	1.444077e+06	1.460446e+06	1201.6975	1208.4890	3.965356e+17	3.947955e+17	0.9027	0.9009	0.9028	0.9010
1	LARS Lasso	1.445477e+06	1.461490e+06	1202.2800	1208.9211	3.913936e+17	3.896166e+17	0.9026	0.9008	0.9027	0.9009
2	Ridge	1.444077e+06	1.460446e+06	1201.6975	1208.4891	3.965332e+17	3.947930e+17	0.9027	0.9009	0.9028	0.9010
3	Desision Tree	1.599824e+05	3.292867e+05	399.9780	573.8351	8.172260e+11	5.020000e-02	0.9892	0.9776	0.9893	0.9777
4	K-Nearest	4.043079e+06	4.577366e+06	2010.7408	2139.4779	5.090113e+16	4.561679e+16	0.7271	0.6968	0.7278	0.6975
5	Random Forest	1.534466e+05	2.479205e+05	391.7226	497.9161	2.702823e+12	1.752626e+12	0.9897	0.9832	0.9898	0.9833
6	XGBoost	1.147716e+06	1.164175e+06	1071.3153	1078.9696	1.114946e+17	1.102518e+17	0.9227	**0.9210**	0.9228	0.9211

- The **XGBoost Model performs well** and provides **0.92 R-Squared** on the test set.
- All trends and patterns that could be caught by these models **without overfitting** were done, and the model reached its **maximum level of performance.**

Conclusion

- **Customers, promos, competition distance, store type b**, are the **major factors** the company should look out for to get the best results for next six weeks.

- **Game theory** and the **Nash Equilibrium** are validated by the fact that most stores have competition within a **distance of 0 to 10 km** and had more sales than stores farther away.This validates the hypothesis about this feature.

- The **dataset outliers displayed justified behaviour.** The anomalies either belonged to store type B or were running promotions that boosted sales.

Don't be the same,
be better!

Recommendations

- It is important to **encourage** more stores to **run promotions**.
- It might be possible to have **more stores of type B**. They have the highest average sales despite having the fewest stores.
- Because there is a seasonal component, retailers should be urged to **advertise and capitalise on the holidays**.