**RETAIL SALES PREDICTION**

**Jayanth V, Gowthaam Kumarasamy,**

**Data Science trainees,**

**Almabetter, Bangalore.**

**Abstract:**

The Retail store chain predicts their sales for the next six weeks to improve their profits and sales using the different factors influencing the sales and which is also used for future inventory management. We were provided with many independent labels which is the reason for the sales lying already in our dataset.

Our experiment can help understand what could be the reason for the distribution of sales in different circumstances by feature selection, data analysis and prediction with machine learning algorithms taking into account previous trends to determine the correct prediction (regression).

**Keywords: Machine learning, sales, time series forecasting, Exploratory data analysis, feature selection.**

**Problem Statement:**

### Rossmann operates over 3,000 drug stores in 7 European countries. Currently, Rossmann store managers are tasked with 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. With thousands of individual managers predicting sales based on their unique circumstances, the accuracy of results can be quite varied.

### You 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.

**Introduction:**

The interest for a product continues to change occasionally. No business can work on its monetary growth without precisely assessing client interest and future demand of items. Sales forecasting refers to the process of estimating demand for or sales of a particular product over a specific period of time. For a good sales forecast, it is extremely important to get a good dataset as well. Forecasts heavily depend on the past records, trends and patterns observed for sales of a particular store.

In this Retail Sales Prediction, machine learning models have been created that predict the sale of these 1115 drug stores across the European market and compare the results of these models. In addition, an effort has been made to analyze and find all the features that are contributing to higher sales and the features that lead to lower sales, so that improvement plans can be worked upon.

**Approach**:

The approach followed here is to first check the sanctity of the data and then understand the features involved. The events followed were in our approach:

**1.** **Understanding the business problem and the datasets**

**2. Data cleaning and preprocessing**: finding null values and imputing them with appropriate values. Converting categorical values into appropriate data types and merging the datasets provided to get a final dataset to work upon.

**3. Exploratory data analysis**: of categorical and continuous variables against our target variable.

**4. Data manipulation**: Feature selection and engineering, feature scaling, outlier detection and treatment, and encoding of categorical features.

**5.** **Modeling**: The Decision tree Regression model was chosen considering our features were mostly categorical with few having continuous importance.

**6.** **Model Performance and Evaluation**

**7.** **Conclusion and Recommendations**

**1. Understanding the business problem and the datasets:**

The first step involved is understanding the data and getting answers to some basic questions like; What is the data about? How many rows or observations are there in it? How many features are there in it? What are the data types? Are there any missing values? And anything that could be relevant and useful to our investigation.

Let’s just understand the dataset first and the terms involved before proceeding further. Our dataset consists of two CSV files, the first consists of historical data with 1017209 rows or observations and 9 columns with no null values.

The second dataset was supplementary information about the stores with 1115 rows and 10 columns and a lot of missing values in a few columns. The data types were integer, float, and object in nature. There are a total of 2 data sets given to predict future sales. They are:

### **Rossmann Stores Data.csv** - historical data including Sales

### **store.csv** - supplemental information about the stores

Let’s define the features involved:

* **Store** - a unique Id for each store
* **Sales** - the turnover for any given day (this is what you 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, Date) 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** – the 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

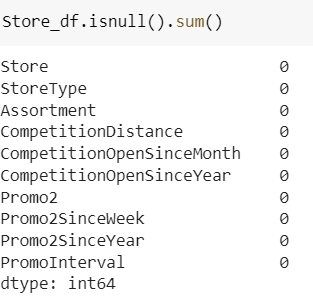
#### **Promo2Since[Year/Week]** - describes the year and calendar week when the store started participating in Promo2

#### **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, and November of any given year for that store.

**2. Data Cleaning and Preprocessing:**

Handling missing values is an important skill in the data analysis process. If there are very few missing values compared to the size of the dataset, we may choose to drop rows that have missing values. Otherwise, it is better to replace them with appropriate values.

It is necessary to check and handle these values before feeding them to the models, so as to obtain good insights into what the data is trying to say and make great characterization and predictions which will in turn help improve the business's growth. The historical records dataset had no null values and duplicated values.





The dataset had a lot of nulls in the following columns:

● CompetitionOpenSinceMonth

● CompetitionOpenSinceYear

● CompetitionDistance

● Promo2SinceYear

● PromoInterval

● CompetitionOpenSinceMonth: gives the approximate month of the time the nearest competitor was opened. The mean of the column is used to impute the missing values in the column as it gives the most occurring month

● CompetitionOpenSinceYear: gives the approximate year of the time the nearest competitor was opened. The mean of the column is used to impute the missing values in the column as it gives the most occurring month.

● CompetitionDistance: Competition Distance is the distance in meters to the nearest competitor store. The mean of the column is used to impute the missing values in the column as it gives the most occurring month.

● Promo2SinceWeek, Promo2SinceYear, and PromoInterval are NaN wherever Promo2 is 0 or False as can be seen in the first look of the dataset. They are replaced with 0.

Finally, before proceeding further, we merge both datasets into one for further analysis.

**3. Exploratory Data Analysis:**

Exploratory data analysis is a crucial part of data analysis. It involves exploring and analyzing the dataset given to find out patterns, trends and conclusions to make better decisions related to the data, often using statistical graphics and other data visualization tools to summarize the results.

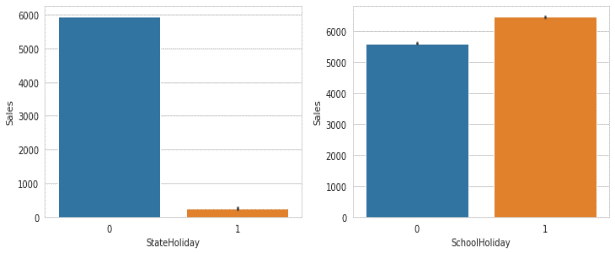
The visualization tools involved in the investigation are python libraries- matplotlib and seaborn. The goal here is to explore the relationships of different variables with ‘Sales’ to see what factors might be contributing to the high and low sales numbers.

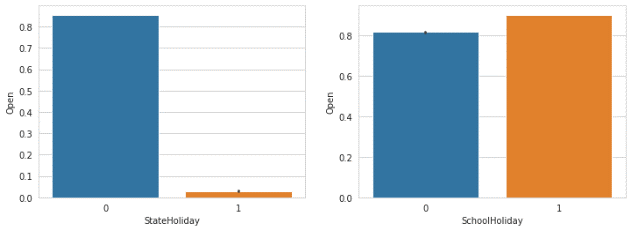
**Approach:**

There are two kinds of features in the dataset: Categorical and Non-Categorical Variables. Categorical- A categorical variable is a variable that can take on one of a limited, and usually fixed, number of possible values putting a particular category to the observation. Non-Categorical- A non-categorical or continuous variable is a variable whose value is obtained by measuring, i.e., one which can take on an uncountable set of values.

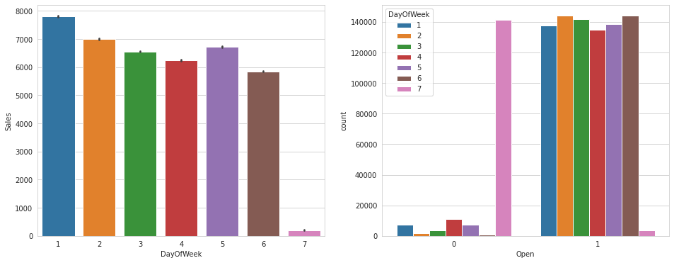
Both of them are analyzed separately. Categorical data is usually analyzed through count plots and bar plots in accordance with the target variable and that is what is done here too. On the other hand, Numeric or Continuous variables were analyzed through distribution plots, box plots and scatterplots to get useful insights.

**Categorical insights:**

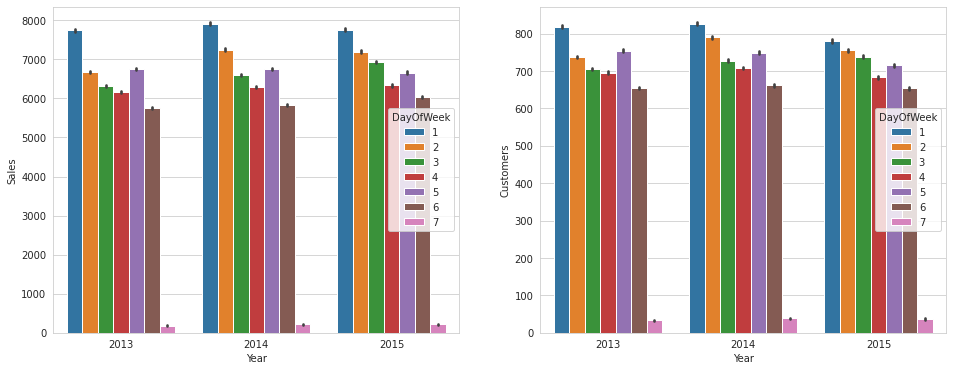


These bar plots explain that on a state holiday the sales were very less compared to not a state holiday. Further, whether it is a school holiday or not the sales are almost high with slight differences.

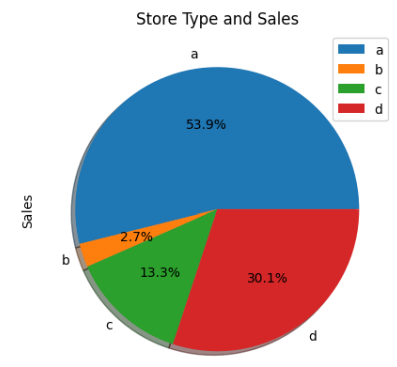
These bar plots infer that during the state holidays the stores were open very rarely. And during the school holidays or not, the stores were opened consistently.

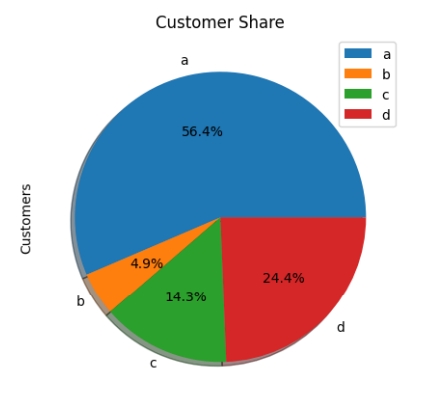


The first plot derives us the insight of during Sundays the sales were very less and the remaining days were the same with minimum differences. The second plot shows us how many shops opened during the days of the week.



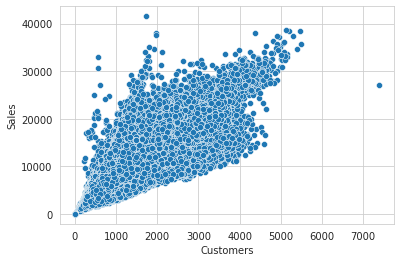
The first plot shows the Sales achieved on different days of the week in three years. They are almost the same. The second plot shows the Customers count gained on different days of the week in three years.



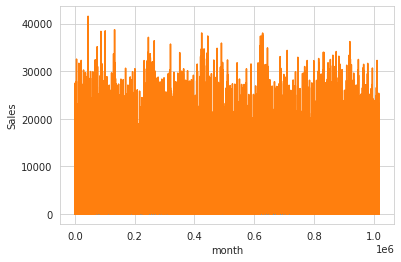


The first pie chart shows the Store type and sales distribution. It clearly shows that the store type ‘a’ cover more than 50% of sales and ‘b’ hold the lowest percentage of sales (2.7%).

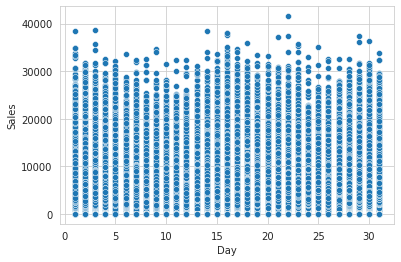
The second pie chart shows the Customer’s share according to the store type. It infers that the store type ‘a’ holds the largest amount of customer share (56.4%) and ‘b’ holds the lowest customer share (4.9%).

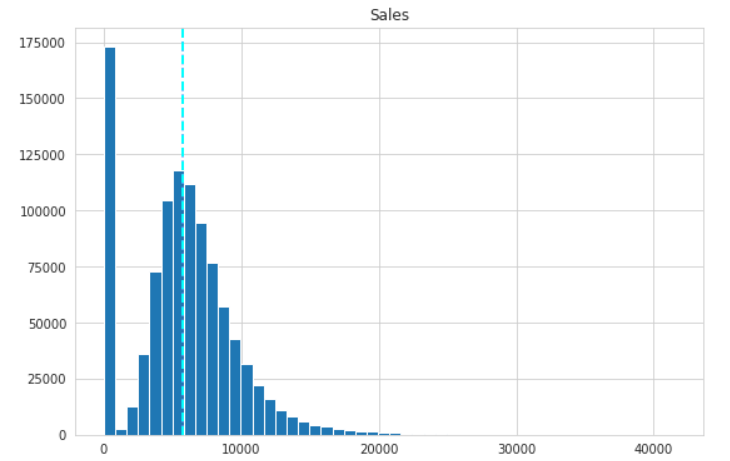
**Continuous Insights:**

This scatter plot shows the clusters of customers and their sales across the range. The most number of customers buy from INR 10000 to INR 20000.



This plot shows us the Sales that occurred in each month of the year.

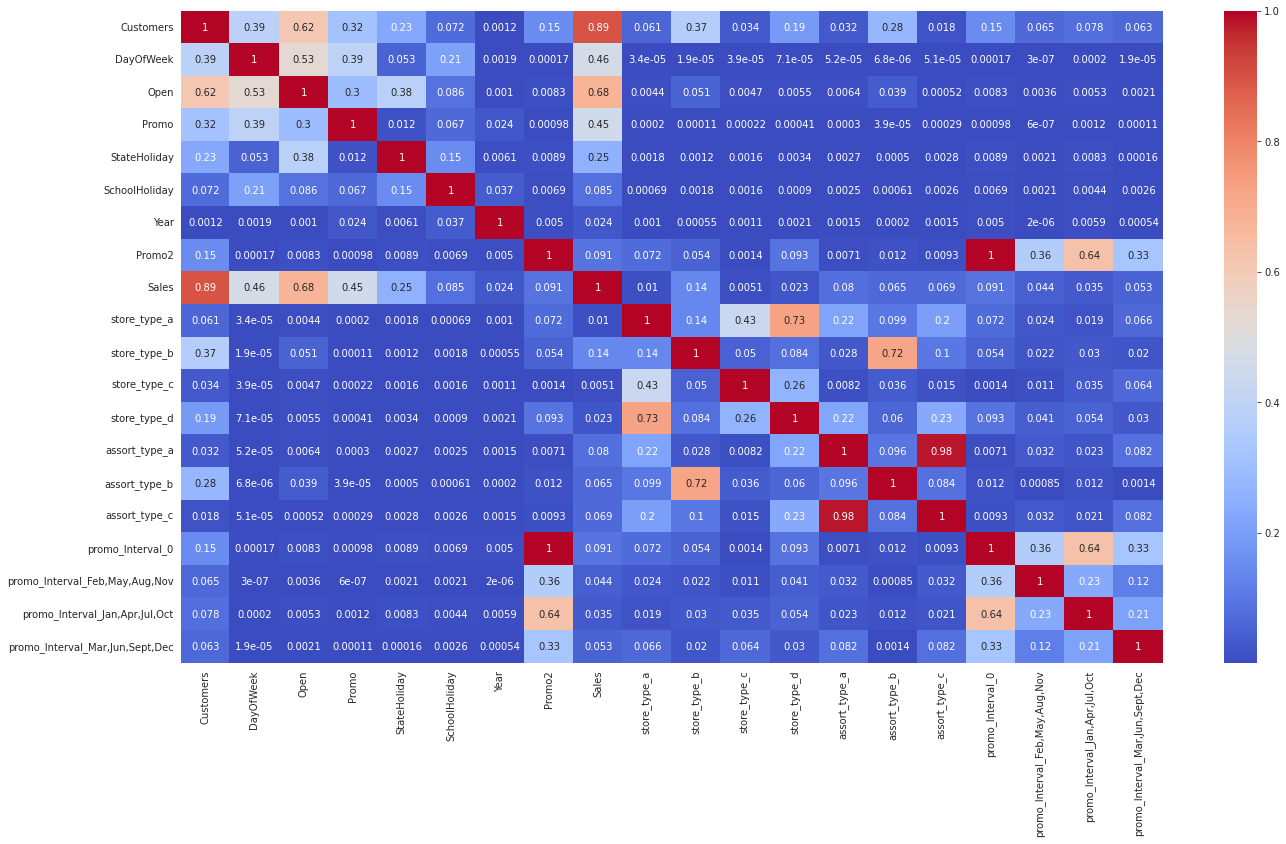


This scatters plot shows the number of sales that the stores gained during different days of the month. It infers the Sales are quite normally split in every day of the month.

This is the distribution of sales happening in the stores. This shows that the Sales distribution has been positively skewed.

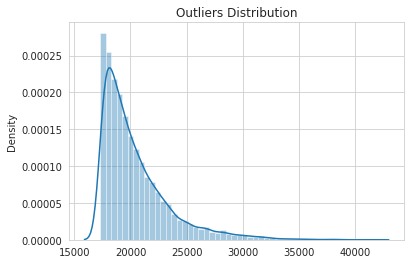
**Correlation:**

Correlation is a statistical term used to measure the degree to which two variables move in relation to each other. A perfect positive correlation means that the correlation coefficient is exactly 1. This implies that as one variable moves, either up or down, the other moves in the same direction. A perfect negative correlation means that two variables move in opposite directions, while a zero correlation implies no linear relationship at all. The correlation we obtained is



● Day of the week has a negative correlation indicating low sales as the weekends, and promo, customers and open has positive correlation.

● State Holiday has a negative correlation suggesting that stores are mostly closed on state holidays indicating low sales



**4. Data Manipulation:**

Data manipulation involves manipulating and changing our dataset before feeding it to various regression machine learning models. This involves keeping important features, outlier treatment, feature scaling and creating dummy variables if necessary.

**Feature Selection:**

After the EDA process, we move on to the feature selection process. Feature selection will be done on both **Numerical and Categorical columns.** We will consider date, month, week as it has many categories. Feature selection on numerical columns.

Here the feature engineering is done based on the Variance Inflation Factor values. Hence importing the necessary libraries and we check for the multi collinearity and safely removing the columns which has the highly correlated. Then it is also done for the Categorical features.

**Outlier Detection:**

In statistics, an outlier is a data point that differs significantly from other observations. Outliers can occur by chance in any distribution, but they often indicate either measurement error or that the population has a heavy-tailed distribution.

Z-score is a statistical measure that tells you how far a data point is from the rest of the dataset. In a more technical term, Z-score tells how many standard deviations away a given observation is from the mean.

z = (x- mean)/standard deviation

More than 3 standard deviations were considered as an outlier. Exploring the outlier’s data frame, some important insights were generated:

● The data points with sales value higher than 28000 are very low and hence they can be considered outliers.

● The outliers had the day of the week as 7 i.e., Sunday and the store type for those observations was ‘b’.

● Other outliers had promotions running on that day.

● It can be well established that the outliers are showing this behavior for the stores with promotion = 1 and store type B. It would not be wise to treat them because the reasons behind this behavior seem fair.

● Being open 24\*7 along with all kinds of assortments available is probably the reason why it had higher average sales than any other store type.

● If the outliers are a valid occurrence, it would be wise not to treat them by deleting or manipulating them especially when we have established the ups and downs of the target variable in relation to the other features. It is well established that there is seasonality involved and no linear relationship is possible to fit. For these kinds of datasets tree-based, machine learning algorithms are used which are robust to outlier effect.

**Feature Scaling:**

Feature Scaling is a technique to standardize the independent features present in the data in a fixed range. It is done to prevent the biased nature of machine learning algorithms towards features with greater values and scale.

Normalization: is a scaling technique in which values are shifted and rescaled so that they end up ranging between 0 and 1. It is also known as Min-Max scaling. [0,1]



**One hot encoding:**

For categorical variables where no such ordinal relationship exists, the integer encoding is not enough. We have categorical data integers encoded with us, but assuming a natural order and allowing this data to the model may result in poor performance.

Many of the features such as DayofWeek, StoreType, and Assortments were categorical in nature and had to be one hot encoded to continue

**5. Modelling:**

Factors affecting choosing the model:

Determining which algorithm to use depends on many factors like the problem statement and the kind of output you want, the type and size of the data, the number of features, and observations in the data, to name a few.

The dataset used in this analysis has:

● A multivariate time series relation with sales and hence a linear relationship cannot be assumed in this analysis. This kind of dataset has patterns such as peak days, festive seasons etc. which would most likely be considered as outliers in simple linear regression.

● Having X columns with 30% continuous and 70% categorical features. Businesses prefer the model to be interpretable in nature and decision-based algorithms work better with categorical data.

**Train-Test Split:**

In machine learning, train/test split splits the data randomly, as there’s no dependence from one observation to the other. That’s not the case with time series data. Here, it’s important to use values at the rear of the dataset for testing and everything else for training.

The latest six weeks were kept as a testing set and the rest of the historical data was used in the training set.

**Fitting Different models:**

For modelling we tried various classification algorithms like:

**1. Linear Regression**

**2. Lasso Regularization**

**3. Decision Tree**

**4. Random Forest.**

**Algorithms:**

**1. Linear Regression:**

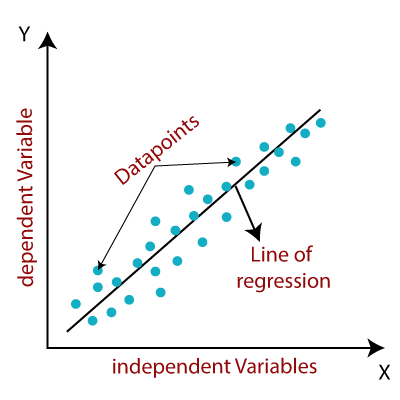
It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables such as **sales, salary, age, product price,** etc.

Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (y) variables, hence called linear regression. Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.

y= a0+a1x+ ε

**Here,**

Y= Dependent Variable (Target Variable)  
X= Independent Variable (predictor Variable)  
a0= intercept of the line (Gives an additional degree of freedom)  
a1 = Linear regression coefficient (scale factor to each input value).  
ε = random error



For Linear Regression, we use the **Mean Squared Error (MSE)** cost function, which is the average of squared error occurred between the predicted values and actual values. It can be written as:

For the above linear equation, MSE can be calculated as:



**Where,**

N=Total number of observations  
Yi=Actualvalue  
(a1xi+a0) = Predicted value.

**2. Lasso Regularization:**

**Lasso regression** is a type of **linear regression** that uses shrinkage. Shrinkage is where data values are shrunk towards a central point, like the mean. The lasso procedure encourages simple, sparse models (i.e., models with fewer parameters). This particular type of regression is well-suited for models showing high levels of multicollinearity or when you want to automate certain parts of model selection, like variable selection/parameter elimination.

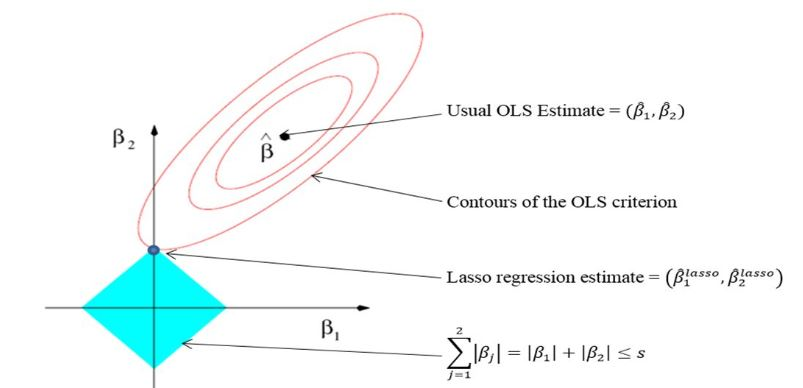
The acronym “LASSO” stands for **L**east **A**bsolute **S**hrinkage and **S**election **O**perator.

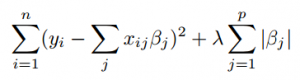
## **L1 Regularization**

Lasso regression performs L1 regularization, which adds a penalty equal to the absolute value of the magnitude of coefficients. This type of regularization can result in sparse models with few coefficients; Some coefficients can become zero and eliminated from the model. Larger penalties result in coefficient values closer to zero, which is the ideal for producing simpler models. On the other hand, L2 regularization (e.g. Ridge regression) *doesn’t* result in elimination of coefficients or sparse models. This makes the Lasso far easier to interpret than the Ridge.

## Performing the Regression

Lasso solutions are quadratic programming problems, which are best solved with software (like Matlab). The goal of the algorithm is to minimize:



  
  
  
Which is the same as minimizing the sum of squares with constraint Σ |Bj≤ s (Σ = summation notation). Some of the βs are shrunk to exactly zero, resulting in a regression model that’s easier to interpret.

A **tuning parameter**, λ controls the strength of the L1 penalty. λ is basically the amount of shrinkage:

* When λ = 0, no parameters are eliminated. The estimate is equal to the one found with linear regression.
* As λ increases, more and more coefficients are set to zero and eliminated (theoretically, when λ = ∞, *all* coefficients are eliminated).
* As λ increases, bias increases.
* As λ decreases, variance increases.

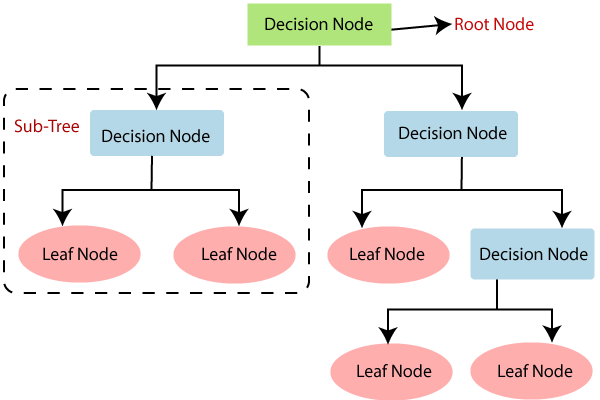
If an intercept is included in the model, it is usually left unchanged.

**3. Decision Tree Regressor:**

Decision tree regression observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output. Continuous output means that the output/result is not discrete, i.e., it is not represented just by a discrete, known set of numbers or values.

Discrete output example: A weather prediction model that predicts whether or not there’ll be rain in a particular day.  
Continuous output example: A profit prediction model that states the probable profit that can be generated from the sale of a product.

Here, continuous values are predicted with the help of a decision tree regression model.

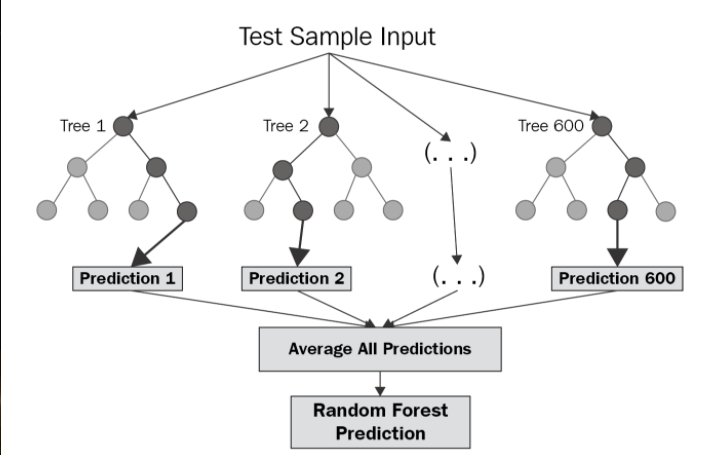


**4. Random Forest:**

It can be used for both Classification and Regression problems in ML. It is based on the concept of **ensemble learning,** which is a process of *combining multiple classifiers to solve a complex problem and to improve the performance of the model.*

As the name suggests, ***"Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset."*** Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

**The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.**



**6.1. Model performance:**

Model can be evaluated by various metrics such as:

● Mean Absolute Error (MAE)- MAE is a very simple metric which calculates the mean of the absolute difference between actual and predicted values.

● Mean Squared Error (MSE)- Mean squared error states the mean of the squared difference between actual and predicted value.

● Root Mean Squared Error (RMSE)- It is a simple square root of mean squared error.

● R Squared (R^2)- R2 score is a metric that tells the performance of your model, not the loss in an absolute sense that how well did your model perform. Hence, R2 squared is also known as Coefficient of Determination or sometimes also known as Goodness of fit. Its value ranges from 0 to 1. It can be negative if the model is performing worse than the base.

● Adjusted R Squared- The disadvantage of the R2 score is while adding new features in data the R2 score starts increasing or remains constant but it never decreases because it assumes that while adding more data variance of data increases. Adjusted R^2 is adjusted for this disadvantage and shows the real value.

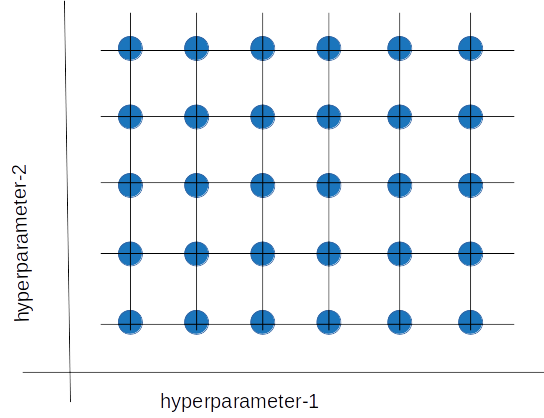
**Hyper parameter tuning:**

Hyperparameter tuning consists of finding a set of optimal hyperparameter values for a learning algorithm while applying this optimized algorithm to any data set. That combination of hyperparameters maximizes the model’s performance, minimizing a predefined loss function to produce better results with fewer errors.

Note that the learning algorithm optimizes the loss based on the input data and tries to find an optimal solution within the given setting. However, hyperparameters describe this setting exactly.

### 1. **Grid search CV:**

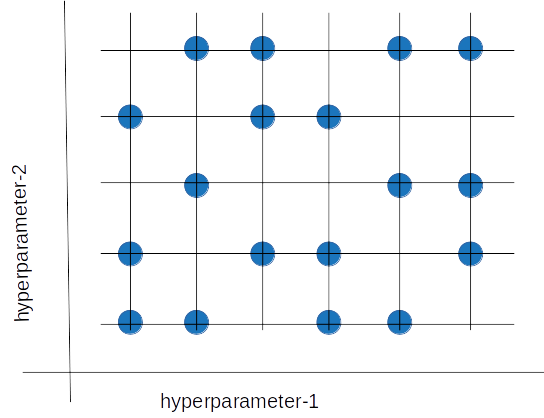
Grid search is a sort of “brute force” hyperparameter tuning method. We create a grid of possible discrete hyperparameter values and then fit the model with every possible combination. We record the model performance for each set and then select the combination that has produced the best performance.



Grid search is an exhaustive algorithm that can find the best combination of hyperparameters. However, the drawback is that it’s slow. Fitting the model with every possible combination usually requires a high computation capacity and significant time, which may not be available.

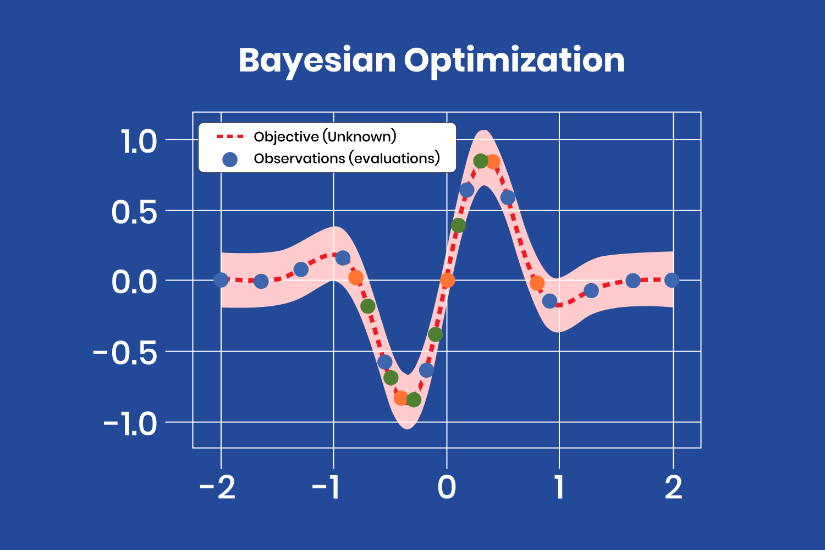
**2. Random Search CV:**

In Random Search, the hyperparameters are chosen at random within a range of values that it can assume. The advantage of this method is that there is a greater chance of finding regions of the cost minimization space with more suitable hyperparameters since the choice for each iteration is random. The disadvantage of this method is that the combination of hyperparameters is beyond the scientist’s control



**3. Bayesian Optimization:**

The Bayesian optimization method takes a different approach. This method treats the search for the optimal hyperparameters as an optimization problem. When choosing the next hyperparameter combination, this method considers the previous evaluation results. It then applies a probabilistic function to select the combination that will probably yield the best results. This method discovers a fairly good hyperparameter combination in relatively few iterations.



The Bayesian optimization method takes a different approach. This method treats the search for optimal hyperparameters as an optimization problem. When choosing the next hyperparameter combination, this method considers the previous evaluation results. It then applies a probabilistic function to select the combination that will probably yield the best results.

This method discovers a fairly good hyperparameter combination in relatively few iterations.

**6.2. Model Evaluation:**

The evaluation for various models implemented:

**1. Linear Regression:**

The metrics used to evaluate the model are:

* **Mean Squared Error:** 1502986.0053577924
* **Root Mean Squared Error:** 1225.963296904843
* **R2 score:** 0.898287012542967

**2. Lasso Regularization :**

The metrics used to evaluate the model are:

* **Mean Squared Error:** 1502533.032668067
* **Root Mean Squared Error:** 1225.7785414454222
* **R2 score:** 0.8983176669904096

**3. Decision Tree :**

The metrics used to evaluate the model are:

* **Mean Squared Error:** 1150508.9997343218
* **Root Mean Squared Error:** 1072.6178255717746
* **R2 score:** 0.9221405209083612

**4. Random Forest :**

The metrics used to evaluate the model are:

* **Mean Squared Error:** 1326056.5291162326
* **Root Mean Squared Error:** 1151.5452787955116
* **R2 score:** 0.9102605276213414

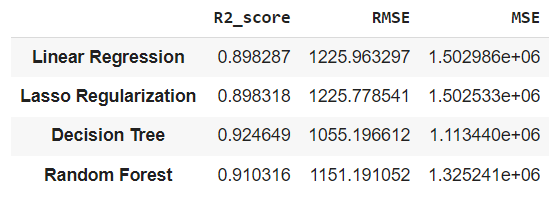
By implementing these above models and by evaluating the metrics , we can find that the R2 score of the decision tree model 0.9221405209083612 is the highest among the others and the RMSE value 1072.6178255717746 is the lowest score among other models.

Thus we can infer that the decision tree regression model is the optimal model to predict the sales for this problem and thus we are implementing the Hyperparameter tuning with the grid search CV and finding the best hyperparameter value for the model. We can find that the min\_samples\_leaf = 30 is the best value for this hyperparameter

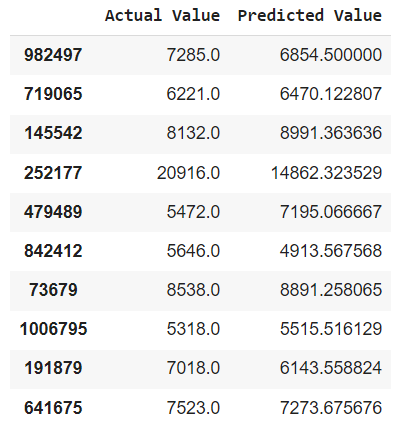
So by setting min\_samples\_leaf = 30 and implementing the model again we get the best optimal model with R2 score increasing from 92.22 to 92.46 and RMSE value decreased from 1072 to 1055.

Thus After implementing Hyperparameter tuning for this model we get the evaluation metrics as:

* **Mean Squared Error:** 1113483.5341185925
* **Root Mean Squared Error:** 1055.217292370909
* **R2 score:** 0.9246461801136623.



**Sales Prediction by the best model:**



**7. Conclusion and Recommendations :**

The main objective of sales forecasting is to paint an accurate picture of expected sales. Sales teams aim to either hit their expected target or exceed it.

When the sales forecast is accurate, operations go smoothly and future planning for the company's growth is done efficiently.

Upon having this analysis, it can be established that given the dataset, the model developed is able to explain **92.4649%** of the variations and is able to predict the sales values in a good range.

**Recommendations:**

● More stores should be encouraged for promotion.

● Store type B should be increased in number.

● There's a seasonality involved; hence the stores should be encouraged to promote and take advantage of the holidays

**Challenges:**

The major challenge would be the computational time and RAM needed to work upon such a dataset in a cloud environment.

**References:**

* Almabetter
* GeeksforGeeks
* Towards Data Science
* Tutorials point
* Stackoverflow
* Other blogs