Retail Sales Prediction

**Abstract:**

Retail sales is the sale of consumer goods, or final goods, by businesses to end consumers and includes in-store sales as well as online sales. Products may be durable (with a significant expected shelf life) or perishable (such as groceries). There are different retail stores in the market. They are specialty stores, department stores, supermarkets, convenience stores, and discount stores. This study is related to the sales of drug stores in European countries. These sales depend on state holiday, school holiday, day of the week and competition distance, etc.

We have two datasets. One dataset contains the details about sales in stores from 2013 to 2015, and the other dataset gives extra information about all the stores in the first dataset. We will perform the necessary manipulations on our datasets for better analysis. Then we will do Exploratory Data Analysis (EDA) to find a relationship between features. As part of data wrangling, we will remove unnecessary features like having multicollinearity and a high Variance Inflation Factor (VIF).

Different regression algorithms are used for machine learning models. They are Linear Regression, Ridge Regression, Lasso Regression, Elastic-Net Regression and Decision Tree Regression.

**Keywords:** VIF, EDA, Linear Regression, Ridge Regression, Lasso Regression, Decision Tree Regression.

**Problem Statement:**

Look at the given datasets and study the relationship between different features or trends in different features. Find the correlation between sales and other features and build an efficient machine learning model to predict future sales for given input variables. Rossmann operates over 3,000 drug stores in seven European countries. Currently, Rossmann store managers are tasked with predicting their daily sales for up to six weeks in advance. Many factors influence store sales, including promotions, competition, school and state holidays, seasonality, and location. With thousands of individual managers forecasting sales based on their specific circumstances, the accuracy of the results can vary greatly.

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.

**About Data:**

* **Id** **-** an Id that represents a (Store, Date) duple within the test set
* **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 -** 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.
* **Date:** the day when transaction took place.
* **DayOfWeek:** a particular day in a week.

**EDA:**

1. **Null Values Treatment:** Null

values are present in features named CompetitionOpenSinceMonth, CompetitionOpenSinceYear, Promo2SinceWeek,

Promo2SinceYear and PromoInterval. These null values are replaced with median and mode values depending on the

type of data.

1. **Univariate Analysis:**

A distribution plot is used to know

the distribution of numerical features,

and a bar plot is used to know the categorical feature distribution. And also, we used count plot.

1. **Bivariate Analysis:** We used a

correlation matrix to find a relationship between all the features. We also used a scatter plot.

1. **New Features Creation:** new

features created by extracting day,

month and year from the date feature. Sales per customer feature derived

from sales and customers. Encoding

the categorical features by renaming

values or by using one hot encoding

technique.

1. **Dropping Features:** we removed few features which are showing multi collinearity, this multi collinearity measured by using VIF.

**Observations From EDA:**

* Average sales first increased from

quarter one to two, then decreased in quarter three, and then increased to maximum average sales in the fourth quarter. And average sales are

increased from 2013 to 2015.

* Our sales record is from January 2013 to July 2015 in 1115 different

stores in Europe.

* Customer feature is highly correlated to dependent variable sales.
* Month and WeekOfYear features are highly correlated and they are independent, so we removed the feature which is less correlated to dependent variable.
* Sales feature is positively skewed, to make it normal we applied log

transformation on sales.

* Most of the time, the stores are open. But on Sunday, the probability of a

store being opened is less. Highest number of happening on Monday and lowest on

Sunday.

* Most of the store's sales (82%) are not affected by school holidays.
* Sales and number of customers features has linear relationship.
* Store type b has a greater number of sales.
* Large number of stores are type a.

**Steps involved in Model**

**Building:**

* **Scaling the data:** Because the units and quantities of different feature

different. So, the importance may go to unimportant feature. To overcome

this difficulty we choose scaling, it can be done by using minmax scaler

and standardised scaler.

* **Fitting different models:** For modelling we tried various classification algorithms

Like:

1. Linear regression
2. Lasso Regression
3. Ridge Regression
4. Elastic-Net Regression
5. Decision Tree Regressor

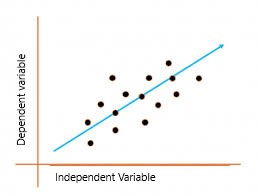
* **Tuning the hyperparameters for better accuracy:**

Tuning the hyperparameters of respective algorithms is necessary for getting better accuracy and to avoid overfitting in case of tree-based models like Decision Tree Regressor.

**Algorithms:**

1. **Linear Regression:** It is a supervised machine learning. For simple Linear

Regression the relationship between independent and dependent variable is linear.



To calculate best-fit line linear regression uses a traditional slope-intercept form which

is given as Y= B0 + B1 X. The goal of the linear regression algorithm is to get the best

values for B0 and B1 to find the best fit line. The best fit line is a line that has the least error which means the error between predicted values and actual values should be

minimum.

In Linear Regression, generally **Mean Squared Error (MSE)** cost function is used,

which is the average of squared error that occurred between the **ypredicted**and **yi.** Using

the MSE function, we’ll update the values of B0 and B1 such that the MSE value settles at the minima. These parameters can be determined using the gradient descent method such that the value for the cost function is minimum.

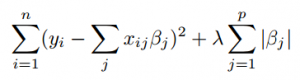
1. **Lasso Regression:** This is extension of Linear Regression algorithm. Lasso (least

absolute shrinkage and selection operator; also Lasso or LASSO) is a regression

analysis method that performs both variable selection and regularization in order to

enhance the prediction accuracy and interpretability of the resulting statistical model.

The cost function for this algorithm is



The goal of the algorithm is to minimize above cost function. A tuning parameter, λ controls the strength of the L1 penalty. λ is basically the amount of shrinkage.

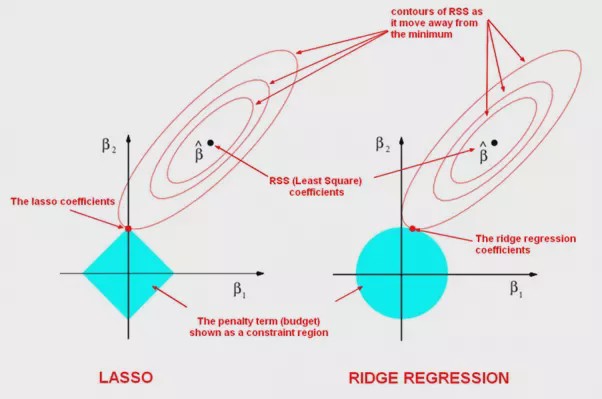
When λ = 0, no parameters are eliminated. The estimate is equal to the one found with

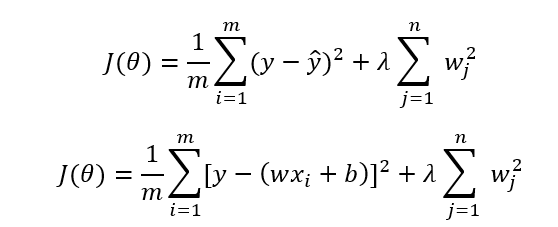
linear regression. As λ increases, more and more coefficients are set to zero and eliminated (theoretically, when λ = ∞, all coefficients are eliminated). As λ increases, bias

increases. As λ decreases, variance increases.

1. **Ridge Regression:** Like Lasso Regression it reduces the overfitting but it won’t

absolutely shrinkage the features.

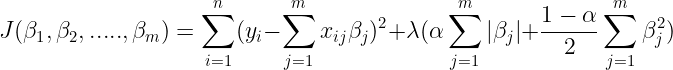




1. **Elastic-Net Regression:** Elastic Net Regularization is a regularization technique that uses both L1 (Lasso) and L2 (Ridge) regularizations to produce most optimized

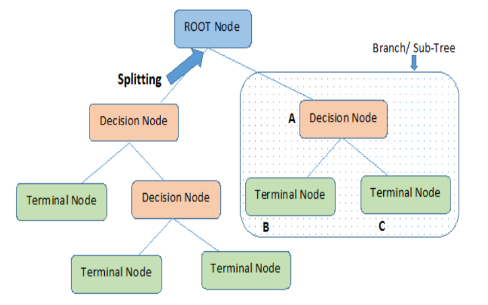
output. This is one of the best regularization techniques as it takes the best parts of

other techniques. The below is cost function:



In addition to setting and choosing a lambda value elastic net also allows us to tune the alpha parameter where 𝞪 = 0 corresponds to ridge and 𝞪 = 1 to lasso.

1. **Decision Tree Regression:** Decision trees can be used for classification as well as regression problems. The name itself suggests that it uses a flowchart like a tree structure to show the predictions that result from a series of feature-based splits. It starts with a root node and ends with a decision made by leaves.



Here we may get doubt about which feature to be split first, for this we

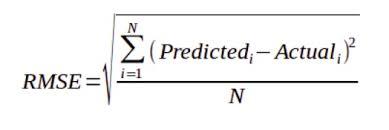
use idea of entropy and information gain. Entropy is nothing but the uncertainty in our dataset or

measure of disorder. Information gain measures the reduction of uncertainty given some feature and it is also a deciding factor for which attribute should be selected as a decision node or root node. There are many ways to tackle this problem through hyperparameter tuning. We can set the maximum depth of our decision tree using the max\_depth parameter. The more the value of max\_depth, the more complex your tree will be. The training error will off-course decrease if we increase the max\_depth value but when our test data comes into the picture, we will get a very bad accuracy. Hence, we need a value that will not overfit as well as underfit our data and for this, we can use GridSearchCV.

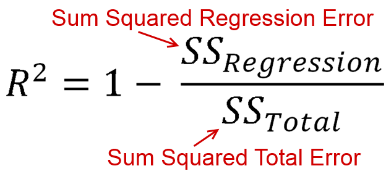
**Metrics used to find performance of Model:**

1. **RMSE:** The root-mean-square

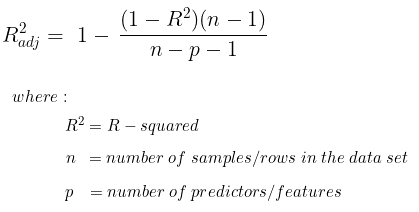
deviation (RMSD) or root-mean-square error (RMSE) is a frequently used measure of the differences between values (sample or population values) predicted by a model or an estimator and the values observed.



1. **R2 Score:** R2 (coefficient of determination) regression score function. Best possible score is 1.0 and it can be negative (because the model can be arbitrarily worse).



1. **Adjusted R2 Score:** It is modification of R2 score. As we increasing the input feature the R2 score will improve mathematically but in reality, the model performance could be worst. To overcome this, we defined adjusted R2 score.



**Results:**

1. From correlation matrix, we can say that 'Customers' feature is highly correlated to Sales (dependent Varible).
2. The 'Month' feature was removed instead of 'WeekOftheYear' because these both features were correlated and 'Month' is less correlated with 'Sales' compared to later one.
3. In linear regression, Customers is the most influencing feature and StateHoliday is at the second place.
4. In Decision Tree Regressor, Customers is the most influencing feature and CompetitionDistance is at the second place.
5. From metrics, we can see that our models are not overfitting.
6. RMSE Comparisons (For Test dataset):

A. Linear Regression: 0.2456

B. Decision Tree Regressor: 0.157

C. Lasso Regressor: 0.286

D. Ridge Regressor: 0.245

E. Elastic Net Regressor: 0.306

1. R2 Score of test dataset:

A. Linear Regression: 0.666

B. Decision Tree Regressor: 0.863

C. Lasso Regressor: 0.544

D. Ridge Regressor: 0.666

E. Elastic Net Regressor: 0.480

1. Adjusted R2 of test dataset:

A. Linear Regression: 0.666

B. Decision Tree Regressor: 0.863

C. Lasso Regressor: 0.545

D. Ridge Regressor: 0.666

E. Elastic Net Regressor: 0.480

Decision Tree Regressor giving good results compared to Linear regressor, lasso and ridge regressor.

**References:**

1. Wikipedia
2. Geeks For Geeks
3. Analytics Vidhya
4. Toward Data Science