A Project Report on

Online Advertisement Revenue Prediction

By
Pratham Kishor Patil
Roll No :- 212CD014

Guide
Prof. Sonali Chakraborty

Department of
Computational and Data Science
National Institute of Technology,
Karnataka 575025
[2021-22]

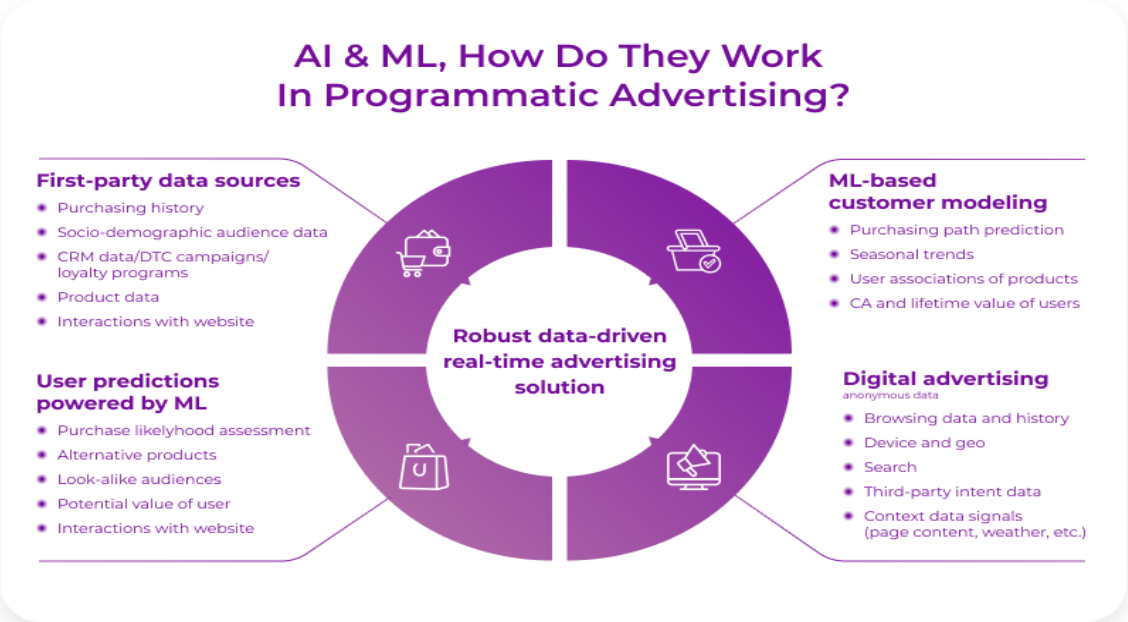
INTRODUCTION

Machine learning is a subfield of artificial intelligence, which is broadly defined as the capability of a machine to imitate intelligent human behaviour Machine learning algorithms have a wide variety of applications, like fraud detections, email filtering etc.

Today in the digital age lots of revenue is generated using the online advertisement. Advertisement are the core source of income of the various websites and mobile application. Online advertising, also known as online marketing, Internet advertising, digital advertising or web advertising, is a form of marketing and advertising which uses the Internet to deliver promotional marketing messages to consumers.

In the following project we will be using all the useful data generated from the online advertisement to calculate the revenue generated by the particular advertisement so that we can find out the most effective advertisement.

The dataset is provided by DeltaX a digital advertising platform. In this project Machine Learning algorithms were deployed to improve performance across the business funnel of advertisers.



PROBLEM STATEMENT AND OBJECTIVE

1. Problem Statement :

The problem here is the predict the revenue of online advertisement through the given data and find out how effective is the advertisement.

2. Objectives :

The objective of this article is to predict flight prices given the various parameters. This will be a regression problem since the target or dependent variable is the revenue which is a continuous data.

- Analysing the data : Checking for any irregularities in the data and making the data ready for pre-processing.
- Data Cleaning and Preprocessing : Use the preprocessing techniques to make the data such that it can be fed to the machine learning algorithm.
- Exploratory Data Analysis : Plot the available data to get visual clues about the trend of the data.
- Model Training : Train the appropriate model using the pre processed and clean data.
- Hyper Parameter Tuning : Perform hyper parameter tuning to optimise the model performance.
- Try Different Models : Compare the results of different models.

DATASET DESCRIPTION

The dataset is provided by DeltaX is the pioneering cross-channel digital advertising platform. The cloud-based platform leverages big data, user behavior, and machine learning algorithms to improve performance across the business funnel of advertisers.

	date	campaign	adgroup	ad	impressions	clicks	cost	conversions	revenue
0	01-08-2020	campaign 1	adgroup 1	ad 1	24	6	0.08	0	0.00
1	01-08-2020	campaign 1	adgroup 2	ad 1	1	0	0.00	0	0.00
2	01-08-2020	campaign 1	adgroup 3	ad 1	13	4	0.04	0	0.00
3	01-08-2020	campaign 1	adgroup 4	ad 1	5	4	0.08	0	0.00
4	01-08-2020	campaign 1	adgroup 1	ad 2	247	126	1.29	4	925.71

The Dataset

Some basic information about dataset :

Number of attributes = 9

Number of entries = 4571

Number of categorical feature = 3

Number of numerical feature = 5

There are 9 Attributes in the given data. Below is the description of each variable.

1) **Date** : Date on which the data is recorded.
2) **Campaign** : Campaign code of the advertisement.
3) **Adgroup** : There are 4 group of ads which are divided as adgroup 1, adgroup 2, adgroup 3, adgroup 4. This is a categorical variable.

```
data.adgroup.unique()
```

```
array(['adgroup 1', 'adgroup 2', 'adgroup 3', 'adgroup 4'], dtype=object)
```

**adgroup attribute has 4 unique categories

4) **Ad** : It represents the type of advertisement the advertisement and its revenue vary as per the ad type. There are total 70 types of ad named ad 1 to ad 70. This is a categorical attribute.

```
data.ad.unique()
```

```
array(['ad 1', 'ad 2', 'ad 3', 'ad 4', 'ad 5', 'ad 6', 'ad 7', 'ad 8',
       'ad 9', 'ad 10', 'ad 11', 'ad 12', 'ad 13', 'ad 14', 'ad 15',
       'ad 16', 'ad 17', 'ad 18', 'ad 19', 'ad 20', 'ad 21', 'ad 22',
       'ad 23', 'ad 24', 'ad 25', 'ad 26', 'ad 27', 'ad 28', 'ad 29',
       'ad 30', 'ad 31', 'ad 32', 'ad 33', 'ad 34', 'ad 35', 'ad 36',
       'ad 37', 'ad 38', 'ad 39', 'ad 40', 'ad 41', 'ad 42', 'ad 43',
       'ad 44', 'ad 45', 'ad 46', 'ad 47', 'ad 48', 'ad 49', 'ad 50',
       'ad 51', 'ad 52', 'ad 53', 'ad 54', 'ad 55', 'ad 56', 'ad 57',
       'ad 58', 'ad 59', 'ad 60', 'ad 61', 'ad 62', 'ad 63', 'ad 64',
       'ad 65', 'ad 66', 'ad 67', 'ad 68', 'ad 69', 'ad 70'], dtype=object)
```

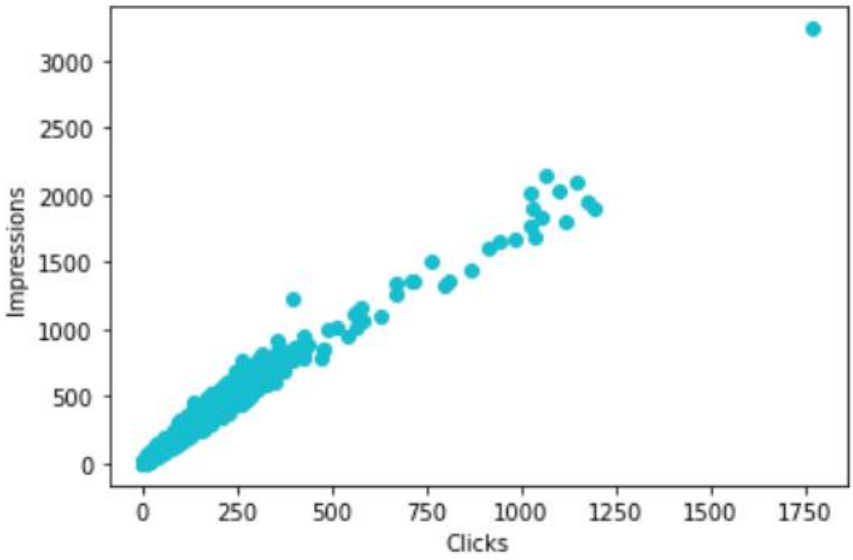
** ad attribute has 70 unique categories.

5) **Impressions** : It represents how many times the advertisement has been displayed on the screen.
6) **Clicks** : It represents how many times the advertisement has been clicked by the user.
7) **Cost** : Cost of displaying each advertisement.
8) **Conversion** : It shows the conversion of the clicks into revenue.
9) **Revenue** : It represents the revenue earned from that advertisement. This is the target variable.

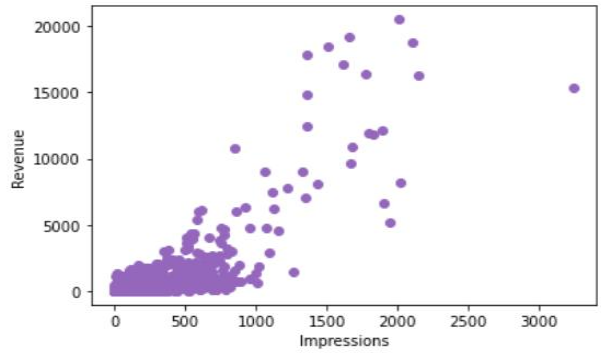
Exploratory Data Analysis

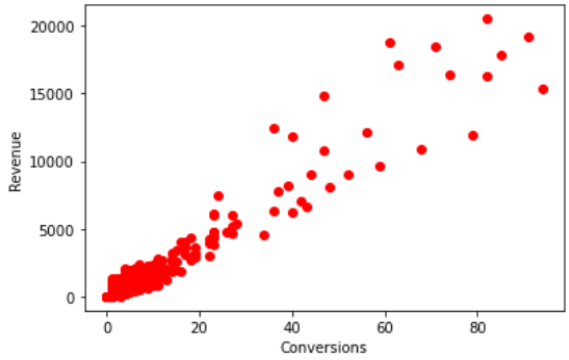
The data is plotted to see the trend and variation between various features.

The independent features clicks and impressions are linearly related.



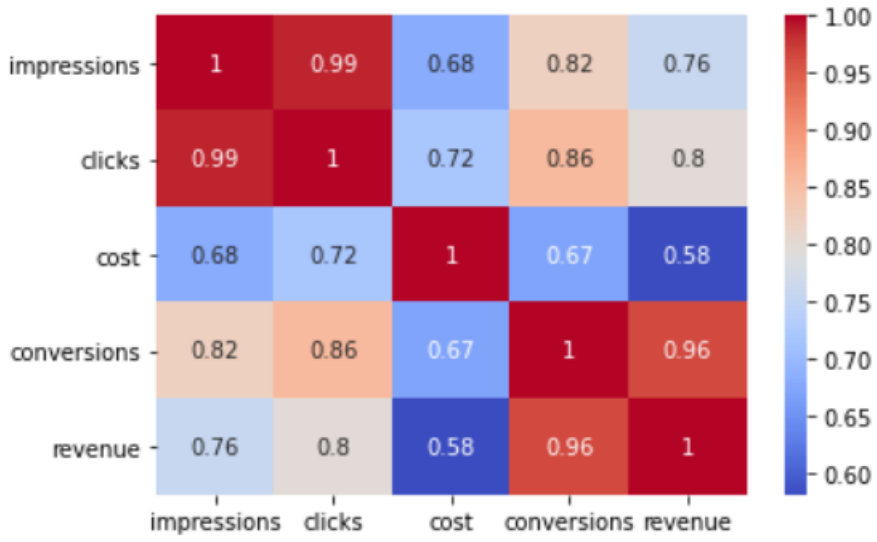
Whereas as other independent features are not linearly related with the dependent feature revenue only the independent feature conversion seems to have been related linearly with the dependent feature revenue.





The correlation matrix also shows the correlation between various attributes in the dataset. Correlation is a statistical measure that expresses the extent to which two variables are linearly related. The correlation coefficient is measured on a scale that varies from + 1

through 0 to – 1. Complete correlation between two variables is expressed by either + 1 or -1.



Correlation Matrix

It shows that there is strong correlation between some of the attributes.

Attributes	Correlation Coefficient
Impressions and Clicks	0.99
Conversion and Revenue	0.96
Conversion and Clicks	0.86

Data Cleaning and Preprocessing

The Data Cleaning ad preprocessing invloes various steps

- **Converting Categorical data to Numerical data :-**

Firstly the adgroup column was one hot encoded. One hot encoding is a process by which categorical variables are converted into a form that could be provided to ML algorithms to do a better job in prediction. In one hot encoding sepearte columns for all the uinque values are converted into spearate rows. Here the adgroup column is converted into 4 separate columns adgroup1, adgroup2, adgroup 3, adgroup 4. So

4 new columns are added to the dataset and the categorical column adgroup is removed from the dataset.

adgroup
adgroup 1
adgroup 2
adgroup 3
adgroup 4
adgroup 1

Before One Hot Encoding

adgroup 1	adgroup 2	adgroup 3	adgroup 4
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1
1	0	0	0

After One Hot Encoding

- **Converting Object data to Integer or Float :-**

The ad column is also transformed the ad part is removed and the number is column converted from object type to integer.

```
data.dtypes
```

```
date            object
campaign        object
adgroup         object
ad              object
impressions      int64
clicks           int64
cost           float64
conversions      int64
revenue        float64
dtype: object
```

Ad with data type as Object Before Operation

ad
ad 1
ad 1
ad 1
ad 1
ad 2

Ad column before Trasformation

Now, the ‘ad’ from the all the rows of ads column is removed and the datatype of the ads column is converted to integer.

```
data.dtypes
```

```
date            object
campaign        object
adgroup         object
ad               int32
impressions      int64
clicks           int64
cost           float64
conversions      int64
revenue        float64
dtype: object
```

Ad with data type as Integer After Operation

ad
1
1
1
1
2

After Transformation

- **Checking for Null Values :-**

There are zero null values in the dataset.

```
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   date         4571 non-null   object
 1   campaign     4571 non-null   object
 2   ad           4571 non-null   int32
 3   impressions  4571 non-null   float64
 4   clicks       4571 non-null   float64
 5   cost         4571 non-null   float64
 6   conversions  4571 non-null   float64
 7   revenue      4571 non-null   float64
 8   adgroup 1    4571 non-null   uint8
 9   adgroup 2    4571 non-null   uint8
 10  adgroup 3    4571 non-null   uint8
 11  adgroup 4    4571 non-null   uint8
dtypes: float64(5), int32(1), object(2), uint8(4)
```

- **Removing Unwanted Data :-**

The column named Campaign is a categorical column and it has only one value stored in it i.e campaign 1. So the columns does not make

nay impact on our final trained model therefore the column campaign in removed. The column which contains the date also does not have any significance with the rest of the data so there is no point in continuing with that column so the date column is also dropped from the dataset. Now we have total 10 attributes.

- **Transformation of Numerical Data :-**

The data is transformed and scaled such the it fits the normal distribution well and we get uniform and well organised data. For data transformation various methods like square root, reciprocal, cube root, logarithmic etc. are used.

The data of all the columns is Transformed as follows

Column Name	Transformation
Impressions	Logarithmic
Clicks	Square Root
Cost	Square Root
Conversions	Sqaure Root
Revenue	Sqaure Root

Later after the above transformation. Standard Scaler Transformation is applied to the data. It is a process to make the mean of data zero and standard deviation as 1. Standardization of a dataset is a common requirement for many machine learning estimators: they might behave badly if the individual features do not more or less look like standard normally distributed data (e.g.

Gaussian with 0 mean and unit variance). Standard Scaler is done by calculating mean and standard deviation of each column.

$$z = \frac{x - \mu}{\sigma}$$

Standardization Formula

- **Outlier Ananlysis and Removal :-**

  The plot gives us the idea of outlier in our data.There are various technique by which we can remove the outlier the technique used here is z score analysis. Z score is also called standard score. This score helps to understand if a data value is greater or smaller than mean and how far away it is from the mean. More specifically, Z score tells how many standard deviations away a data point is from the mean.

  Z score = (x -mean) / std. deviation

  If the absolute value of z score of a data point is more than 3, it indicates that the data point is quite different from the other data points.

  Before Outlier analysis the number of data points where 4571. After the Z score outlier removal the number of data points reduced to 4433. Therefore a total of 138 records were remove after the outlier analysis.

```
z_scores = stats.zscore(train)
abs_z_scores = np.abs(z_scores)
filtered_entries = (abs_z_scores < 3).all(axis=1)
train_no = train[filtered_entries]
```

```
Before Outlier Ananlysis No. of records->  (4571, 10)
After Outlier Ananlysis No. of records ->  (4433, 10)
```

Model Traning and Evaluation

- **Model 1 – Random Forest Regression**

  The exploratory data analysis suggests that the data will be best fit for a tree based model instead of linear regression. Now we choose Random Forest over Decision Tress as there are higher chances of overfitting on Decision Tree whereas Random Forest being part of ensemble learning process the overtiffing is low. Random Forest Regression is a supervised learning algorithm that uses ensemble learning method 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.

  A Random Forest Model is fit on the data and to improve the accuracy and decrease the error following hyperparameters are tuned

  1. n_estimators = 100
  2. min_samples_split = 100

  Evaluation of the model 1

  - Mean Sqaured Error for Train Data : 7.31
  - Mean Sqaured Error for Test Data : 7.37
  - Model 1 Score = 92.08 %

- **Model 2 – Random Forest Regression with new Features**

  Now, three more features are added into the model which will help to improve the performance of the model. Feature engineering refers to the process of using domain knowledge to select and transform the most relevant variables from raw data when creating a predictive model using machine learning or statistical modeling. the three new features are -

  1. CTR = Clicks / Impressions
  2. CPC = Cost / Clicks
  3. CPA = Cost/ Conversions

These were the features that were highly correlated and will help to improve the model. Now the total number of features have incresed from 10 to 13.

Training Random Forest Regression model with new features and hyper parameters.

1. n_estimators = 100
2. min_samples_split = 50

Evaluation of the model 2

- Mean Sqaured Error for Train Data : 6.06
- Mean Sqaured Error for Test Data : 7.16
- Model 2 Score = 93.42 %

Model 2 performed better than model 1 despite they are same model as we have added new features which lead to decrease in the error and increase in the accuracy.

- **Model 3 – Linear Regression**

Linear regression is used for finding linear relationship between target and one or more predictors. After the addition of the new features the data has become more suitable for linear regression. Therefore applying linear regression on the new dataset with 13 features.

Evaluation of the model 3

- Mean Sqaured Error for Train Data : 7.68
- Mean Sqaured Error for Test Data : 7.02
- Model 2 Score = 91.67 %

As evident from the exploratory data analysis this model did not perform better than the random forest regression despite having new features.

- **Model 4 – Support Vector Regression**

Support Vector Regression (SVR) uses the same principle as SVM, but for regression problems. The problem of regression is to find a function that approximates mapping from an input domain to real numbers on the basis of a training sample.

Evaluation of the model 4

- Mean Sqaured Error for Train Data : 9.32
- Mean Sqaured Error for Test Data : 9.72
- Model 2 Score = 89.89 %

Support Vector Regression performs most poorly with the datset. SVR gave least score and highest Mean Sqaured Error.

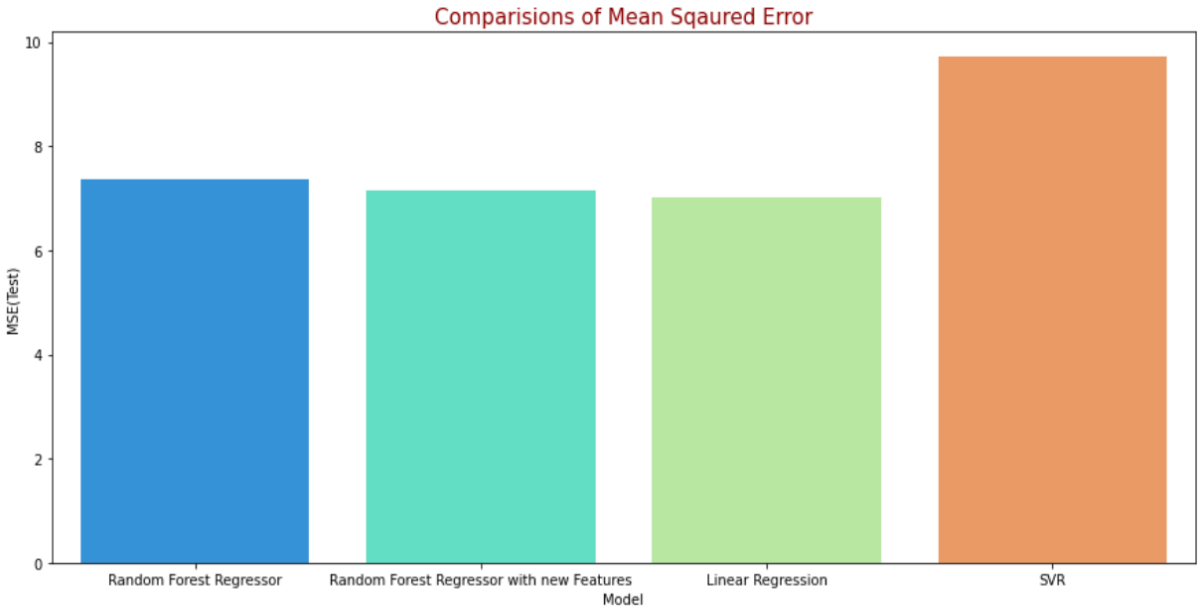
Comparison of the Models

The models can be compared with help of a table as follows

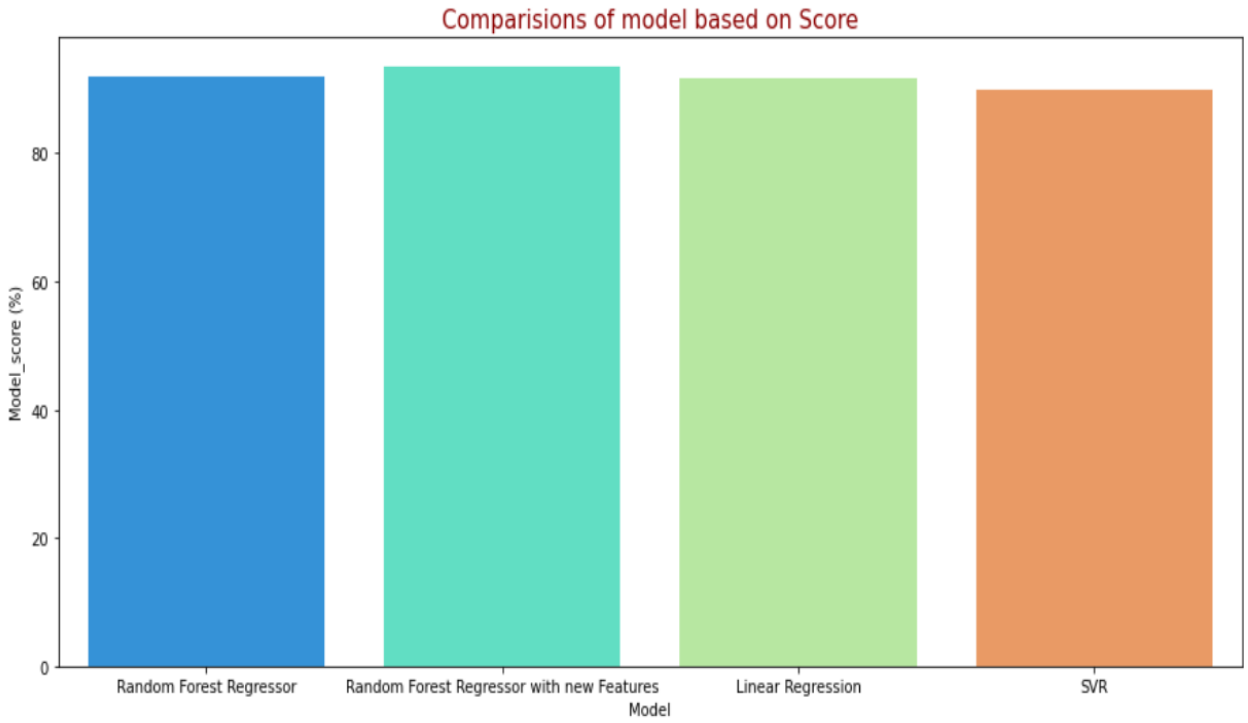
Model Name	MSE(Train)	MSE(Test)	Model Score(%)
Random Forest Regression	7.31	7.37	92.80
Random Forest Regressions with new Features	6.06	7.12	93.42
Linear Regression	7.68	7.02	91.67
Support Vector Regression	9.32	9.72	89.89

Comparing the models visually.

Comparison Based on Mean Sqaured Error :-



Comparison Based on Model Score :-



Conculsions

- The attributes were not much linearly related therefore Random Forest Regression performed better than other models.

- Support Vector Machine Performed worst among all models with accuracy of 89.89% and mean squared error of 9.71

- Adding three new features usind feature engineering improved the score of the Random Forest Regression Model.

- Highest Model Accuracy is given by Random Forest Regression with added features which is 93.42 % and lowest mean squared error of 7.12.

- All the models were hypertuned well to overcome overfitting and the error for the train and test data are nearly equal for all models.