Retail Marketing Analysis

Final Report

Student:
Pooja Chitradurga Vasan - 100799175

Faculty:
Marcos Bittencourt

Executive summary:

Retail industry is the major driving cause of many powerful economies in the world. It is estimated 16.9% of the US economy is from retail industry. There has been surge in online shopping activity in the past 4 years due to many companies starting to sell their products online and the convenience that online shopping gives to the customers. Due to prevalence of both online and in-store availability of products, companies always have faced a challenge to provide customers with a seamless experience across both platforms.

There has been a downward trend in in-store purchases in the past few years due to very competitive markdowns online vs that of in-store. Hence finding the right markdown percentage or markdown amount would greatly improve the in-store experience for the customer and enhance margins for the company.

Rationale statement:

Most of the companies have the biggest sales seasonally in their business based on the product that is being sold by them. The biggest time frame when you have the highest bottom-line sales would be during holidays (valentine's day, Mother's Day, super bowl, Christmas, thanksgiving). This is when most of the companies tend to offer their products with a heavy markdown percentage which would affect that company's margins. The dataset that' is being used has sales data broken down by departments over a period of 3 years for physical stores.

The model developed would predict which departments will be affected and to what extent during the key holiday time frames so that executives and marketing department can make informed and data-driven decisions about markdown amounts to preserve margins.

Data requirements:

- Data is required to have at least 2 years' worth sales information by department
- Population should be represented across all seasons in a year
- population should have no more than 10% of outliers and missing values
- data should be provided in csv or json format for injection
- dependent variables should be normally distributed
- The data size must be feasible according to the available machine.

Assumptions

- Assuming good data quality from source system
- Assuming the dataset is the representation of whole population
- The features are relevant to the dataset
- Assuming that the definitions provided to the dataset are correct
- Assuming that holiday sales in the dataset are weighed more than that of the non holiday time frame
- Assuming there is no wild variation in the data
- Assuming that there is at least 2 years worth data to overcome underfitting/ overfitting problems
- Assuming stores that have no sales in the first year are new stores

Limitations and constraints:

- Definitions of features given in the data source not very clear
- Missing values in at least 3 columns that need to be imputed
- Size of the stores vary across the whole dataset
- Type of the store is not consistent with size of the store
- Some stores have no sales during the 2year period
- Markdowns 1-5 are not available

Data source:

The data set contains anonymized information about the 45 stores, their sales over 3 years and features like store size, department, etc. The dataset is split into 3 csv files:

1. **Stores** - Type and Size of 45 stores
2. **Features** – Store, Date, Temperature of the region, Fuel Price, Mark down, CPI(customer price index), Unemployment rate, Holiday flag
3. **Sales** - Store, Department, Date, Weekly Sales, Holiday flag

https://www.kaggle.com/manjeetsingh/retaildataset?select=stores+data-set.csv *by Manjeet Singh.*

Data set:

Stores

Anonymized information about the 45 stores, indicating the type and size of store

Features

- Store - the store number
- Date - the week
- Temperature - average temperature in the region
- Fuel_Price - cost of fuel in the region
- MarkDown1-5 - anonymized data related to promotional markdowns. Markdown data is only available after Nov 2011 and is not available for all stores all the time. Any missing value is marked with an NA
- CPI - the consumer price index
- Unemployment - the unemployment rate
- IsHoliday - whether the week is a special holiday week

Sales

- Store - the store number
- Dept - the department number
- Date - the week
- Weekly_Sales - sales for the given department in the given store
- IsHoliday - whether the week is a special holiday week

Test Process:

- Make sure the features are normally distributed across the population
- Check distribution of all features in the dataset by conducting EDA
- Identify outliers and missing data in the dataset
- Make sure that the data is balanced and impute data using the most appropriate imputation method if unbalanced
- Observe corelation between the features
- Remove highly corelated features and perform feature selection to solve for over/underfitting problems and to eliminate any bias
- Check for linearity requirements

- Split the data into training and validation sets
- Train and test relevant models
- Plot ROC curves to understand the sensitivity and accuracy of models
- Based on ROC curves and confusion matrix, perform hyper parameter tuning to improve efficiency of the model
- Choose the model with best accuracy.

Exploratory Data Analysis:

- Data source consists of 3 separate data sets. One for “sales”, one that contains “store characteristics” and another for “features”.

Concatenating all datasets

- Merged all three datasets into one using left join.

(Diagram)

- Explored summary statistics of all the dataset.
- Removed any negative sales values for all stores.

Missing values and duplicates:

- Looked for any duplicates for stores across all features and found none.
- Looked for missing values in the whole dataset and found that all MarkDowns had missing values.

Imputation:

- Analysed central measures of all MarkDown features and found greater standard deviations across all of them.
- Imputed all the markdown values with median for that feature.

Outliers:

- Looked for outliers in the dataset for Customer Price Index (CPI).
- Computed Inter-Quartile Range (IQR), upper and lower limits.
- Found that there were no outliers

Sampling and EDA:

- Picked 10% randomized sample from the population dataset due to computational issues.

Figure 1 : Looked at the distribution of holidays in the dataset.

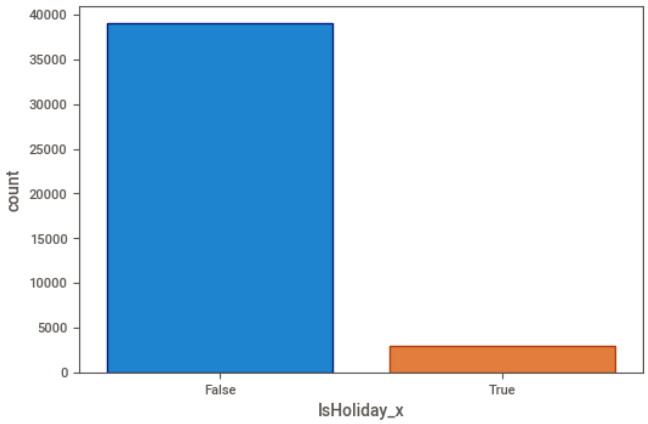


Figure 2

Figure 3 : The week over week sales were normally distributed across all years.

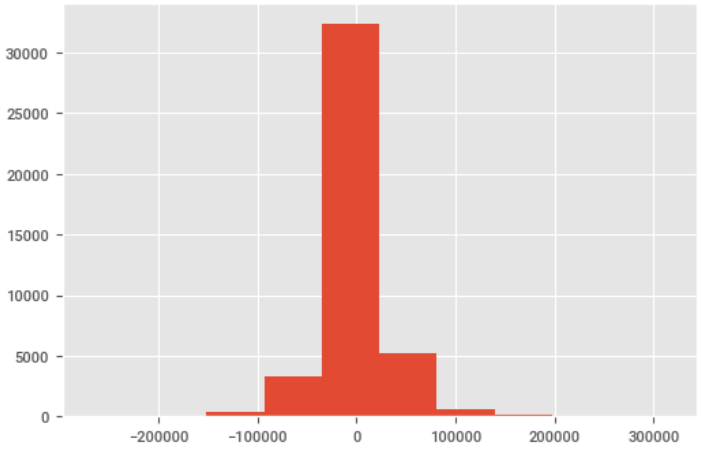


Figure 4

Figure 5: Computed the correlation between each feature within the dataset.

Observations:

- There is a high positive correlation between store size and Weekly sales
- There is a high correlation between markdowns and holiday time frame
- There is a negative correlation between weekly sales and unemployment, CPI, fuel price.

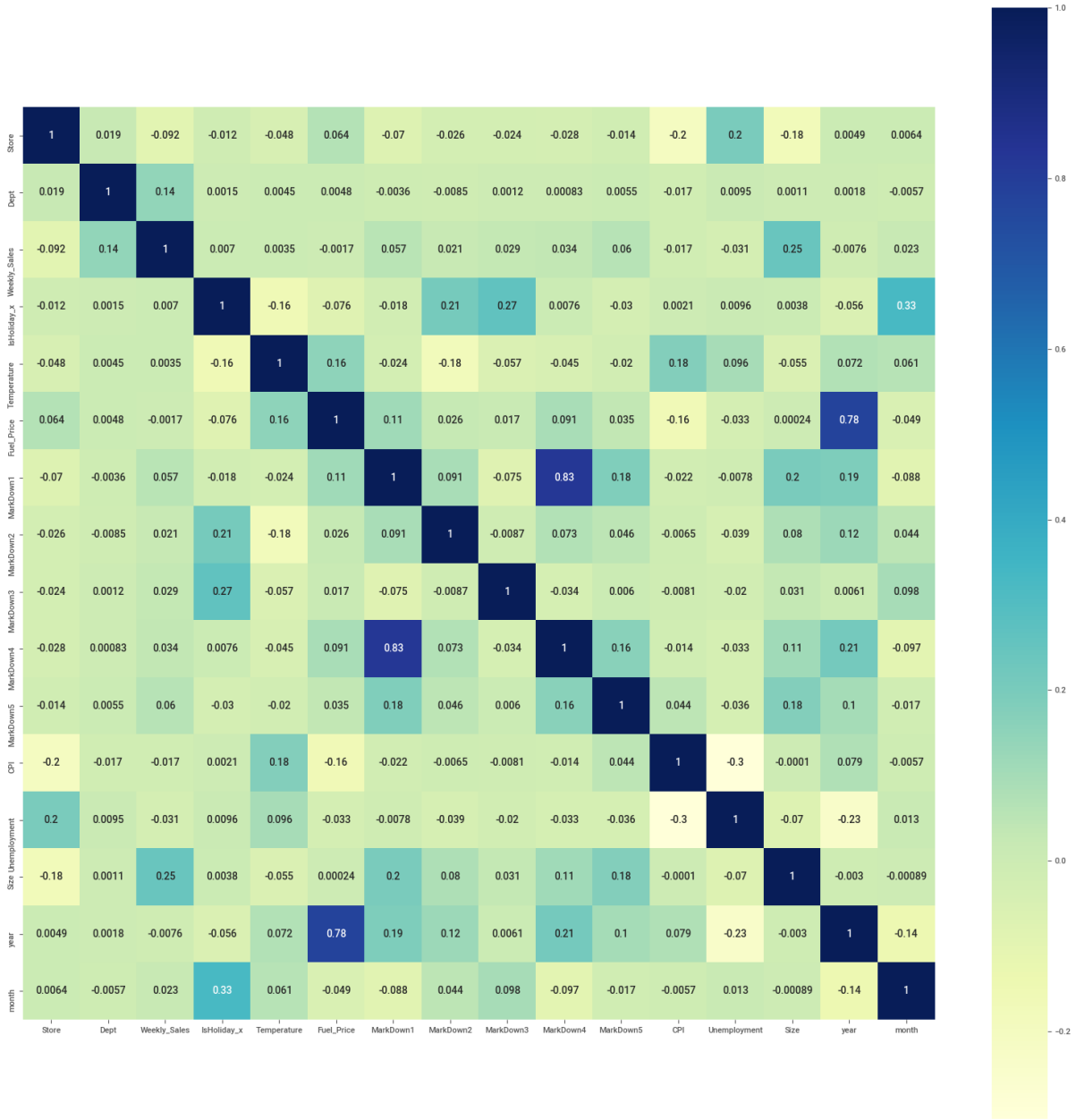


Figure 6

Figure 7 : It is noticeable that the business is seasonal with sales peaking during the holiday months.

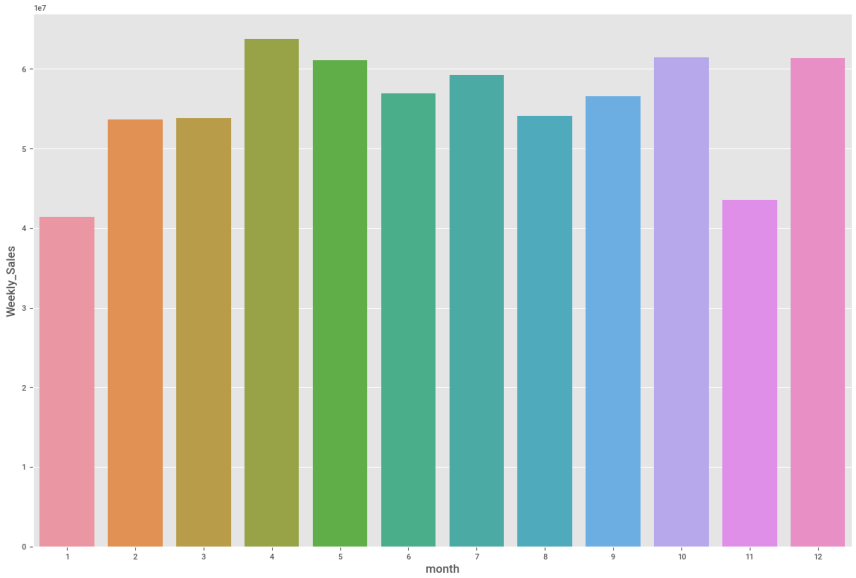


Figure 8

Figure 9: It is noticeable that there were more holiday weeks in 2011 than any other year. This shows bias of holidays within the dataset.

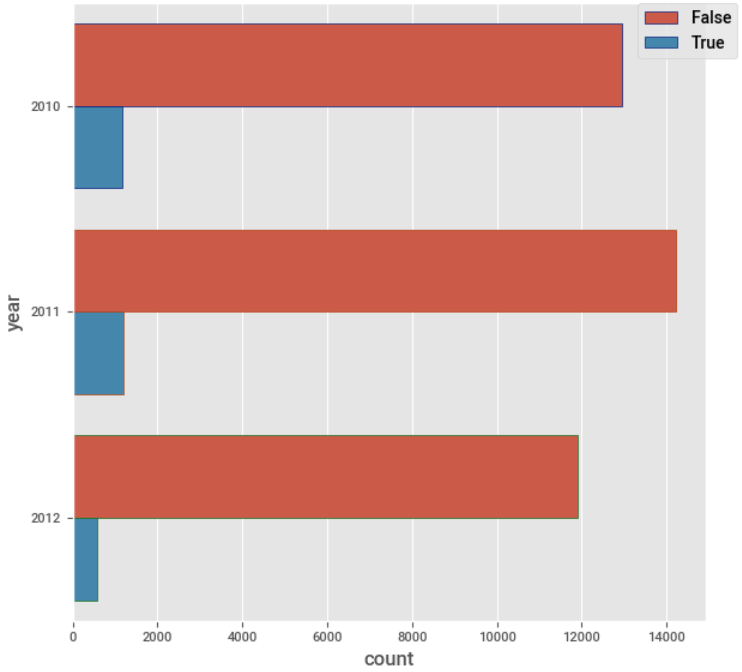


Figure 10

Figure 11 : Across all years there are a greater number of type B store

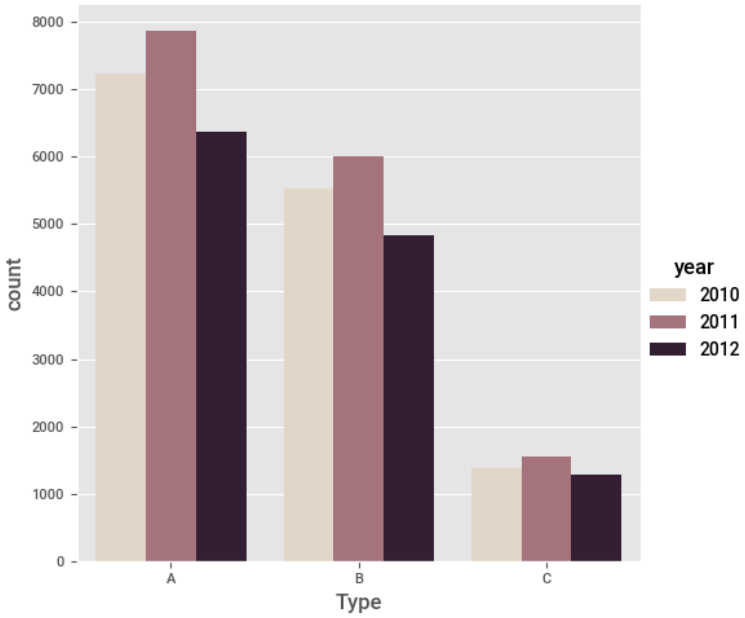


Figure 12

Figure 13: Departments 38,40,92,95 have yielded the highest sales across all years.

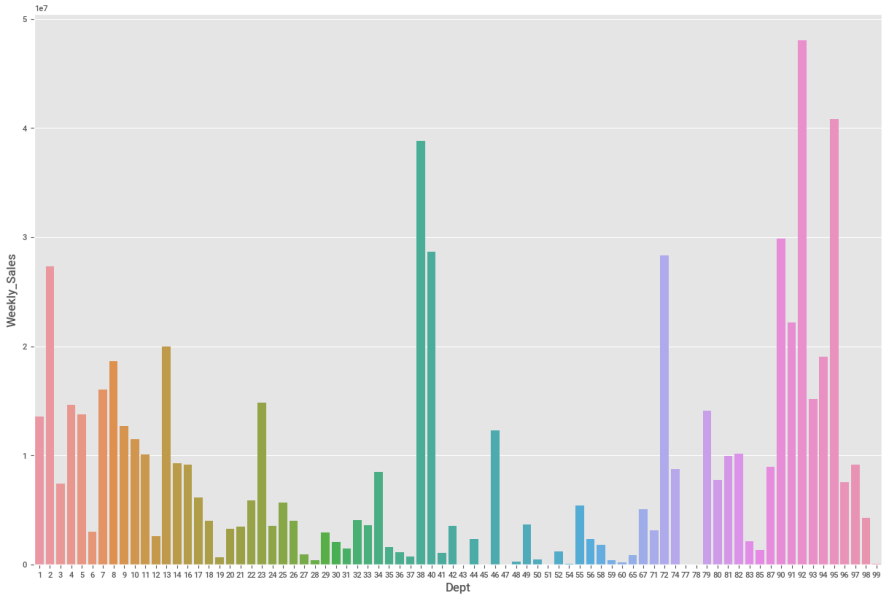


Figure 14

Data Pre-processing:

- One hot encoding - Used LabelEncoder to change categorical values to numeric values for the model to ingest.
- To make sure that the data us internally consistent, that is all the variables has same format, I have used Standardization feature scaling method.
- Split dependent and independent variables.
- Used Feature scaling, ensemble methods.

Model Evaluation:

- Target variable is 'Weekly_Sales' which is a continuous variable that needs to be predicted.
- Plan is to test the following models to evaluate which would yield the best accuracy and pick the best model:
    1. Multivariate linear regression
    2. Lasso regressor
    3. Ridge Regressor
    4. Gradient Booster
    5. Random forest regressor
    6. Decision tree regressor

As the target variable is continuous, the first algorithm that is being considered is Linear Regression – which yielded unfavourable score. I started tuning the model by changing hyperparameters. But then I felt like the manually tuning is an unnecessarily tedious approach.

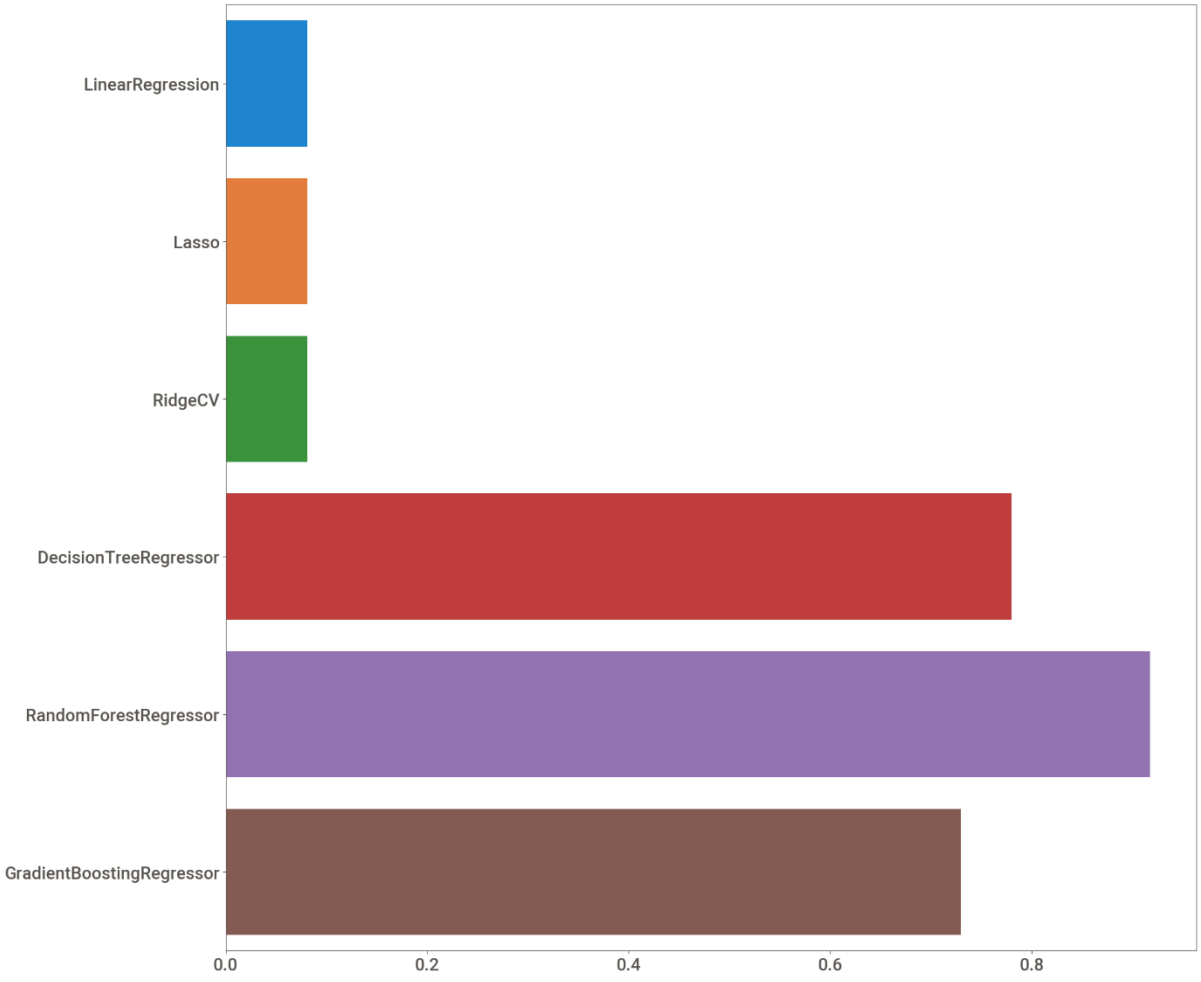
The next thing I did was to compare linear regression with Lasso and Ridge Regression. Lasso and Ridge are nothing but an extension of linear regression that adds regularization penalty to the loss function during training. As it is noticeable, all the three yielded almost same result.

There are of course many regression algorithms that can be used. And the next algorithm that I went with was Decision Tree Regressor. When I tried tuning the tree manually, I got not-so-good results. With hyperparameter-max_depth being 3, I got an accuracy of 36% and with 9 I got 77%.

Since Decision Tree did not yield a good result, I chose one of the boosting algorithms which is Gradient boosting. It basically is good for regression models and what it does is to gradually minimize the loss function of the whole system using Gradient Descent

method. The boosting algorithms are applied to the model to increase their accuracy but as you can see it does almost nothing in this case.

The following model was a simple Random Forest Regressor by changing some hyperparameters, training the model, evaluating its performance, and repeating these steps. The reason as to why I picked that is because it has the highest number of hyperparameters.



After bringing up the accuracy from 37% to 91%, Random forest seems the best choice in this case.

Prediction:

The model that is being used to predict values is Random Forest as it has the highest accuracy comparatively.

The prediction of weekly sales done by considering 5 markdowns and IsHoliday flag.

Deployment:

Serialization is a process to send complex object hierarchies over a network or save the internal state of the objects to a disk or database for later use.

Python has three modules for serialization. One being ***Pickle***, another on being *Marshal* and one more *json*. Here, I have used pickle to load the model and to read it.

Heroku delivers tool that enable software development. It allows to deploy, build, manage, and scale enterprise-level applications. Heroku applications connect securely to on-premises systems on your network and to Cloud-based services like Salesforce.

Appendix:

Programming language	Python
Mark-up Language	HTML, CSS
IDE	Jupyter Notebook, Sublime text
Cloud application platform	Heroku

Further Improvement:

- Add more data.
- Try some more techniques like normalization, feature engineering, feature selection which would yield better accuracies.
- Add edge cases for the HTML form where alert messages pop up when the user input something other than required output.
- Improve web page using better CSS and JavaScript features.

References:

- Zixuan Zhang: Boosting Algorithms Explained
  https://towardsdatascience.com/boosting-algorithms-explained-d38f56ef3f30
  Published Jun 26,2019. Accessed Dec 16,2020.
- https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/
- Davide Mastromatteo : The Ptyhon pickle Module: How to persist objects in python
  https://realpython.com/python-pickle-module/
  Published Apr 27,2020. Accessed Dec 16,2020.
- Michael Rockford : An introduction to Heroku
  https://medium.com/@GoRadialspark/an-introduction-to-heroku-c11c6fcbffa
  Published Sep 12, 2017. Accessed Dec 16, 2020.