DIAMOND PRICE PREDITION:

A Data-Driven Approach

PROJECT REPORT

Submitted by

MEGHA K THOMAS

ABSTRACT

The diamond industry has been a high financial interest due to the significant value and investment potential of diamonds like gold. Predicting the prices of diamonds accurately is a critical task for jewelers, investors, and consumers. This project aims to develop a predictive model using regression analysis to estimate diamond prices based on various characteristics of diamond.

The main objective of this project is to create a regression model that can accurately predict the price of a diamond based on its characteristics like carat, cut, color, clarity, depth, table, and dimensions (length, width, and depth).

The dataset used in this project contains information about diamonds, including:

- **Carat**: The weight of the diamond.
- **Cut**: The quality of the cut (e.g., Fair, Good, Very Good, Premium, Ideal).
- **Color**: The color grade of the diamond (ranging from D to J, with D being the best).
- **Clarity**: The clarity grade of the diamond (e.g., I1, SI2, SI1, VS2, VS1, VVS2, VVS1, IF).
- **Depth**: The total depth percentage.
- **Table**: The width of the top of the diamond relative to the widest point.
- **Dimensions**: Length, width, and depth of the diamond.
- **Price**: The price of the diamond in US dollars.

Methodology

1. **Data Preprocessing**:
    - Handle missing values and outliers.
    - Encode categorical variables (e.g., cut, color, clarity).
    - Standardize numerical features.
2. **Exploratory Data Analysis (EDA)**:
    - Visualize relationships between diamond characteristics and prices.
    - Identify significant predictors of diamond prices.
3. **Model Selection and Training**:
    - Split the data into training and testing sets.
    - Train multiple regression models including:
        - Linear Regression
        - Polynomial Regression
        - Decision Tree Regression

        - Random Forest Regression
4. **Model Evaluation**:
    - Evaluate models using metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R^2).
    - Perform cross-validation to ensure model robustness.
5. **Model Tuning**:
    - Optimize hyper parameters using grid search or random search techniques.
    - Select the best-performing model based on evaluation metrics.
6. **Prediction and Validation**:
    - Validate the final model on the testing set.
    - Compare predicted prices with actual prices to assess model accuracy.

The project will present the performance of various regression models and identify the best model for predicting diamond prices. The selected model is expected to provide accurate price predictions, aiding stakeholders in making informed decisions.

Accurate prediction of diamond prices using regression analysis can significantly benefit the diamond industry. By leveraging machine learning techniques and thorough data analysis, this project aims to develop a reliable predictive model that can be used for valuation and investment purposes.

Future enhancements could include integrating additional features, exploring more advanced machine learning algorithms, and incorporating real-time market data to further improve prediction accuracy.

INTRODUCATION

Problem statement

The aim of this project is to predict the price of the diamond accurately according to different characteristic of diamond. But it is complex task to predict the price. Various characteristics of diamond are carat, cut, color, clarity and size.

Objective

The main objective of this project is to create a model that predict the price of the diamond accurately according to the different characteristics of diamond.

The other objectives are:-

- To find the factors of diamond which is mostly affecting the price.
- Use the different machine learning algorithm to find out best regression problem.

Scope

The aim of the project is to provide a support to the industries related to diamond like jewelers, diamond traders and customers by providing estimated price of diamond according to its different characteristics. And the model can be used as diamond price prediction tool.

DATA COLLECTION AND PREPROCESSING

Data Source

Dataset is collected from Kaggle. It includes the information about diamond like carrot, cut, color, clarity, depth, table, price and size.

Data Description

- **Carat**: - Weight of the diamond
- **Cut**: - Quality of cut of diamond (Fair, Good, Very Good, Premium, Ideal).
- **Color**: - Color of diamond ranging from J(worst) to D(best)
- **Clarity**: - Measure of clarity of diamond (IF, VVS1, VVS2, VS1, VS2, SI1, SI2, I1)
- **Depth**: - Total depth of diamond
- **Table**: - Width of the top of the diamond
- **Price**: - Price of the diamond in dollar
- **Size(x,y,z)**: - Size is measured by multiplying the dimensions of the diamond

Data Cleaning

- Removed/handled the missing values if any.
- Removed the outliers using statistical technique.
- Converted the categorical data (cut, color, clarity) into numerical data using label encoder.

Data exploration

- Visualized and analyzed the relationship between different characters of diamond using scatter plot.

METHEDOLOGY

Model Selection

- Linear Regression
- Decision Tree Regression
- Random Forest Regression

Used correlation analysis technique to find out the appropriate model.

Model Training

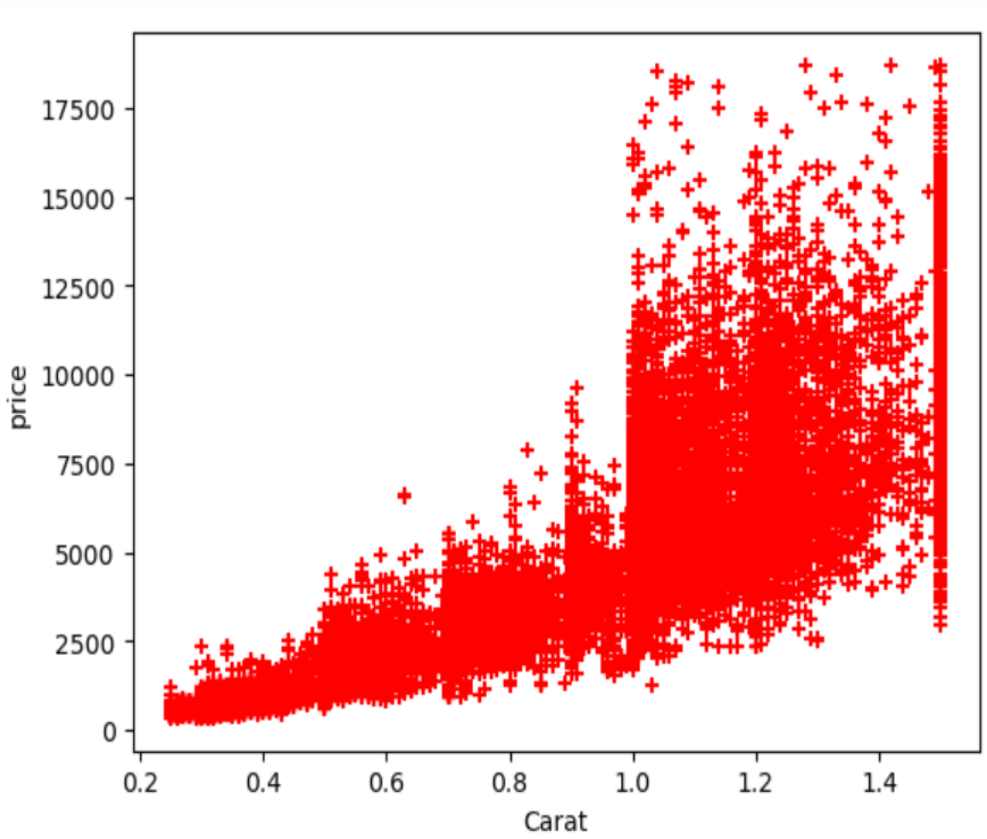
Split the data into train and test set. 80% of the data is used for the training and 20% of the data is used for testing.

EXPLORATORY DATA ANALYSIS

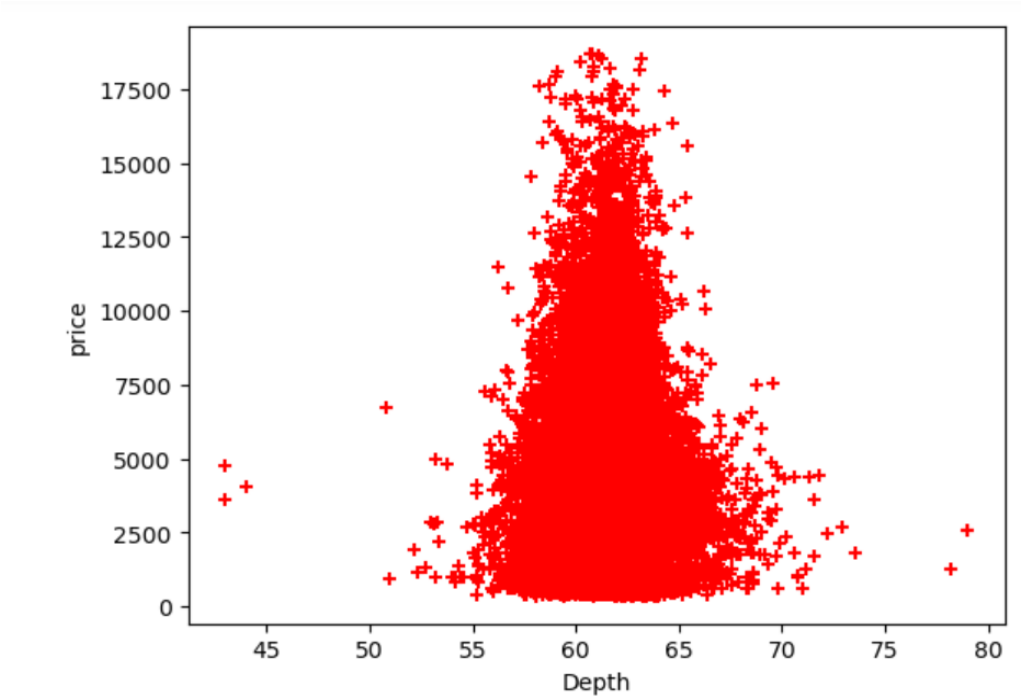
The chart below shows the information regarding different columns in the dataset.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 53940 entries, 0 to 53939
Data columns (total 10 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Carat(Weight of Daimond)   53940 non-null  float64
 1   Cut(Quality)               53940 non-null  object
 2   Color                      53940 non-null  object
 3   Clarity                    53940 non-null  object
 4   Depth                      53940 non-null  float64
 5   Table                      53940 non-null  float64
 6   Price(in US dollars)       53940 non-null  int64
 7   X(length)                  53940 non-null  float64
 8   Y(width)                   53940 non-null  float64
 9   Z(Depth)                   53940 non-null  float64
dtypes: float64(6), int64(1), object(3)
memory usage: 4.1+ MB
```

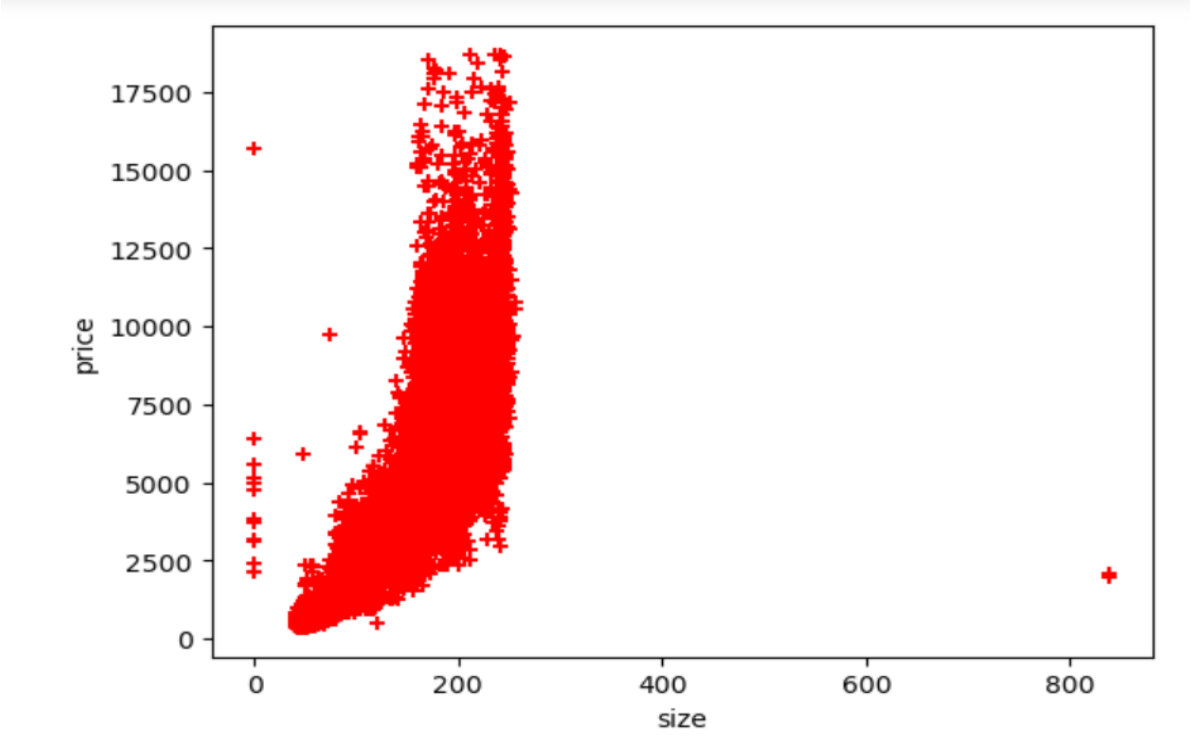
Scatter plot between price and carat



Scatter plot between price and depth



Scatter plot between price and size



CONCLUSION

In conclusion, Diamond Price Prediction Project has demonstrated significant value to the diamond industry by predicting the price accurately according to the characteristics given. Comparing, training and testing with the various algorithms, only linear regression gives an accuracy more than 80%. So, to train and test a model, linear regression is the best algorithm as it gives more accuracy than other algorithms.

Import necessary models:

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

Import the dataset:

```
df=pd.read_csv("Diamond Price Prediction.csv")
```

Analyzing the dataset:

```
df.info()
df.shape
```

Preprocessing the data:

```
max_th=df["Carat(Weight of Daimond)"].quantile(0.90)
min_th=df["Carat(Weight of Daimond)"].quantile(0.01)
df=df[(df["Carat(Weight of Daimond)"]<max_th) & (df["Carat(Weight of Daimond)"]>min_th)]
```

Converting caterological data to numerical:

```
label_cut=LabelEncoder()
label_color=LabelEncoder()
label_clarity=LabelEncoder()
df["cut_num"]=label_cut.fit_transform(df["Cut(Quality)"])
df["color_num"]=label_color.fit_transform(df["Color"])
df["clarity_num"]=label_clarity.fit_transform(df["Clarity"])
```

Calculating size of diamond:

```
def size(a,b,c):
    return a*b*c
df["size"]=df.apply(lambda k: size(k["X(length)"], k["Y(width)"], k["Z(Depth)"]), axis=1)
```

Dropping unwanted data:

```
df=df.drop(["Cut(Quality)","Color","Clarity"],axis="columns")
df=df.drop(["X(length)","Y(width)","Z(Depth)"],axis="columns")
```

Scatter plotting the data verses price:

```
 plt.scatter(df["Carat(Weight of Daimond)"],df["Price(in US dollars)"], color="red", marker="+")
plt.xlabel("Carat")
plt.ylabel("price")
plt.scatter(df["Depth"],df["Price(in US dollars)"],color="red",marker="+")
plt.xlabel("Depth")
plt.ylabel("price")
```

```
plt.scatter(df["size"], df["Price(in US dollars)"],color="r", marker="+")
plt.xlabel("size")
plt.ylabel("price")
```

Creating a new dataframe for training and testing purpose:

```
df1=df.drop(["Price(in US dollars)"],axis="columns")
```

Train and testing the model by fitting the algorithm:

```
X_train,X_test,y_train,y_test=train_test_split(df1,df["Price(in US dollars)"],
train_size=0.8)
model=LinearRegression()
model.fit(X_train,y_train)
model.predict(X_test)
```

Finding the accuracy of model and predicting the price:

```
model.score(X_test,y_test)
model.predict([[0.7,62,58,3,4,3,146.72]])
```