# REGRESSION MODEL PERFORMANCE ANALYSIS

Arista Huerta Angeles, IRS A01369984, Tecnológico de Monterrey Campus Querétaro.

ABSTRACT. This document presents the implementation of a model regression, its performance, and an analysis of the results to predict the behavior to some

## I. INTRODUCTION

The purpose of the Machine Learning application is to implement a linear regression model to predict the cartage of diamonds with respect to their price, length, width, and depth.

## II. DATASET

The dataset to use was download on the platform of Kaggle.

Link to access: https://www.kaggle.com/datasets/nancyalaswad90/diamonds-prices.

The dataset consists of 53,940 values attributes of diamonds with 10 features to describe them (carat, cut, color, clarity, depth, table, price, x, y, and z).
Most variables are numeric in nature, like depth, table, price, x, y, z, and the carat; but the variables cut, color, and clarity are variables of classification.

The column cut represent a classify of possible cuts, such as:

- Good
- Ideal
- Premium
- Very Good

The column color describes a classification of possible colors to take the diamond, these possibilities are represented with the following letters:

| D | E | F | J | H | I | J |
|---|---|---|---|---|---|---|

The column clarity it referred a classify the qualitative metric that grades the visual appearance of the diamonds. That describe with values:

| | |
|---|---|
| IF | VS1 |
| I1 | VS2 |
| SI1 | VVS1 |
| SI2 | VVS2 |

The columns x, y, and z are referend diamond measurements as:

- x: length in mm
- y: width in mm
- z: depth in mm

In *figure 1*, is possible to look the composition of the dataset.

| | carat | cut | color | clarity | depth | table | price | x | y | z |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0.23 | Ideal | E | SI2 | 61.5 | 55.0 | 326 | 3.95 | 3.98 | 2.43 |
| 1 | 0.21 | Premium | E | SI1 | 59.8 | 61.0 | 326 | 3.89 | 3.84 | 2.31 |
| 2 | 0.23 | Good | E | VS1 | 56.9 | 65.0 | 327 | 4.05 | 4.07 | 2.31 |
| 3 | 0.29 | Premium | I | VS2 | 62.4 | 58.0 | 334 | 4.20 | 4.23 | 2.63 |
| 4 | 0.31 | Good | J | SI2 | 63.3 | 58.0 | 335 | 4.34 | 4.35 | 2.75 |

*Figure 1*. Visualization data.

## III. MODEL PROPOSAL

The next step after knowing the data was to decide the variable to wish would predict. I decide to the variables to applicate are numeric, so the desired prediction would be a linear regression.
For this approach was necessarily know the correlation between variables. In *figure 2*, is possible to appreciate the correlation.

| | carat | depth | table | price | x | y | z |
|---|---|---|---|---|---|---|---|
| carat | 1.000000 | 0.060455 | 0.075822 | 0.914849 | 0.985407 | 0.981809 | 0.985633 |
| depth | 0.060455 | 1.000000 | -0.378928 | -0.003255 | -0.050412 | -0.057843 | 0.166681 |
| table | 0.075822 | -0.378928 | 1.000000 | 0.046337 | 0.082141 | 0.065589 | -0.007677 |
| price | 0.914849 | -0.003255 | 0.046337 | 1.000000 | 0.943096 | 0.946188 | 0.933106 |
| x | 0.985407 | -0.050412 | 0.082141 | 0.943096 | 1.000000 | 0.997307 | 0.975230 |
| y | 0.981809 | -0.057843 | 0.065589 | 0.946188 | 0.997307 | 1.000000 | 0.973542 |
| z | 0.985633 | 0.166681 | -0.007677 | 0.933106 | 0.975230 | 0.973542 | 1.000000 |

*Figure 2*. Correlation table.

After visualizing the table, the variables with more correlated are carat, price, x, y, and z.
Finally, I chose the carat as the variable to predict, with respect to the price, and diamond measurements.

## IV. LINEAL REGRESSION IMPLEMENTATION

Start the code importing the libraries to auxiliar the implementation of model.

```
import numpy as np
import pandas as pd

from sklearn.linear_model
import LinearRegression
from sklearn.model_selection
import train_test_split
```

Next step is read the file .csv to load the data.

```
data =
pd.read_csv('c:\\Users\\angix\\D
ownloads\\MachineLearning\\Diamo
nds_Prices2022.csv')
```

As the dataset contains many samples, I decided to reduce the data and only use 500 samples for the application. And deleted the default index of the data.

```
data = data.iloc[:500]
data = data.drop(['Unnamed:
0'], axis=1)
```

After that, I defined the variables X and Y to use in the model. X contains the independent variables (price and diamond measurements) and Y is the dependent variable desired to predict (carat).

```
x = data[['price','x','y','z']]
y = data['carat']
```

To implement, the model has divided the dataset into training data and testing data, with the function *train_test_split ()*. This function consists of split sets into random train and test subsets.

```
Xtrain, Xtest, ytrain, ytest =
train_test_split(x, y)
```

Subsequently, the model was defined (Linear Regression) and trained with the corresponding data.

```
model = LinearRegression()
model.fit(Xtrain, ytrain)
```

Finally, the prediction was made with the trained model and the test data was used.

```
ypred = model.predict(Xtest)
```

To visualize the prediction, I plotted the prediction data and test data, which can be seen in *Figure 3*.

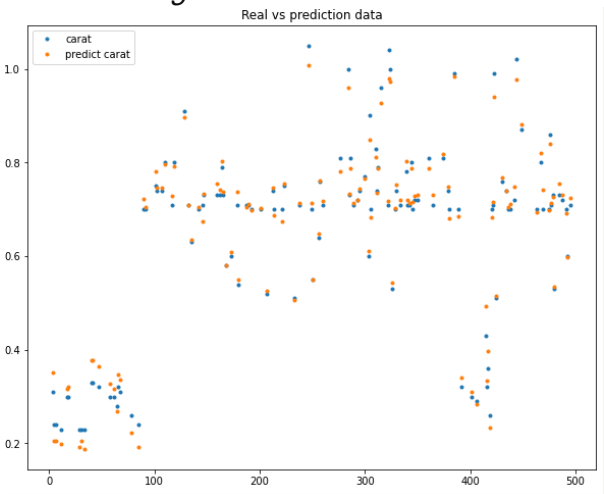


*Figure 3*. Predictions plot with test data.

The plot shows that the predictions are in different values, but to verify the error of the prediction its necessary made some calculations.

# V. VALIDATION