Comparative Analysis of Various Regression Models on Laptop Price Prediction

A S M Nasim Khan
Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
a.s.m.nasim.khan@g.bracu.ac.bd

Mohammad Nasif Sadique Khan
Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
mohammad.nasif.sadique.khan@g.bracu.ac.bd

MD Mohibur Zaman
Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
md.mohibur.zaman@g.bracu.ac.bd

Ateya Ahmed Subarna
Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
ateya.ahmed.subarna@g.bracu.ac.bd

***Abstract*—This project compares four regression models - Linear Regression, KNN, Decision Tree, and Random Forest - for predicting laptop prices based on various features. The evaluation is based on four metrics, and the goal is to identify the most suitable model for accurate predictions. The results can provide valuable insights for businesses and consumers, aiding informed decisions on pricing and purchasing strategies.**

I. Introduction

Predicting the price of a laptop is an important task that has significant practical applications in the tech industry. Accurately predicting the price of a laptop can help businesses make informed decisions on pricing and marketing strategies. Additionally, consumers can use these predictions to make informed purchasing decisions. In recent years, the availability of vast amounts of data and advances in machine learning techniques have made it possible to develop accurate prediction models for various domains. In this paper, we focus on predicting laptop prices using various regression models and comparing their performance.

II. Motivation

With the increasing demand for laptops, predicting their prices accurately is becoming more crucial than ever before. A regression model can help predict the prices of laptops based on various features such as brand, processor, RAM, storage, and screen size, etc. However, choosing the most suitable regression model for this problem is challenging, as several factors need to be considered, including the size and complexity of the dataset, the number of features, and the distribution of the target variable.

Therefore, in this paper, we compare and evaluate the performance of four popular regression models - Linear Regression, KNN, Decision Tree, and Random Forest - to determine which algorithm performs best for predicting laptop prices. We use four different metrics to measure the performance and error of each algorithm, namely R2 Score, MAE, MSE, and RMSE. By comparing these metrics, we aim to identify the most suitable model for predicting laptop prices accurately. The results of this analysis could have significant implications for businesses and consumers alike, by providing valuable insights into the best method for predicting laptop prices.

III. Dataset Description

The dataset used in this analysis consists of laptop information, including the manufacturer, model name, category, screen size, CPU, RAM, storage, GPU, operating system, weight, and price. The dataset contains 13 columns and 1302 rows.

A. Source of the dataset

B. Features

- Manufacturer: The name of the laptop manufacturer.
- Model Name: The name of the laptop model.
- Category: The category of the laptop (e.g., gaming, business, personal).
- Screen Size: The size of the laptop screen in inches.
- Screen: The type of laptop screen and resolution
- CPU: The type of CPU used in the laptop.
- RAM: The amount of RAM in the laptop.
- Storage: The type and size of storage (HDD/SSD) in the laptop.
- GPU: The type of graphics processing unit (GPU) used in the laptop.
- Operating System: The operating system used in the laptop.
- Operating System Version: The version of the operating system used in the laptop.
- Weight: The weight of the laptop in kilograms.
- Price: The price of the laptop is Indian Rupees.

IV. Data Analysis

In our dataset, there are a total of 1302 instances of laptops each having 13 features. Among these 977 instances are for

training the model and 325 are for testing our model.
Here we see that in our data the majority of the instances belong to lower to mid-valued prices.

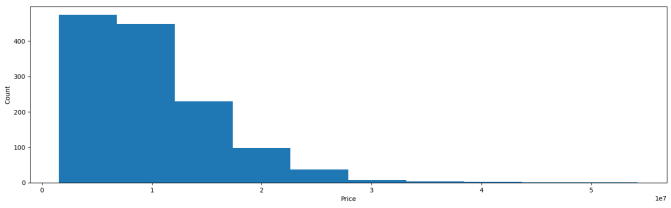

Fig. 1. Count v/s price

This graph shows that the dataset does not have any irregular RAM size present, also there is no laptop with less than 2GB ram.

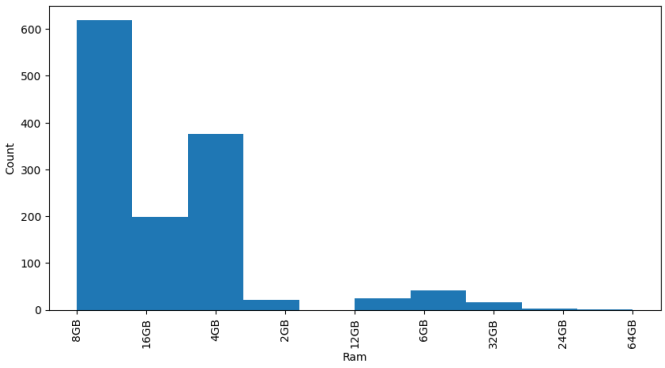

Fig. 2. Ram v/s count

Here we see, in our dataset, the maximum number of laptops belongs to Dell, Lenovo, Hp, and Asus brands. However, these brands' prices are not too high, and the highest prices of laptops belong to the Razer brand's laptops. Since there are more instances for those brands, our model can be slightly biased toward those brands.

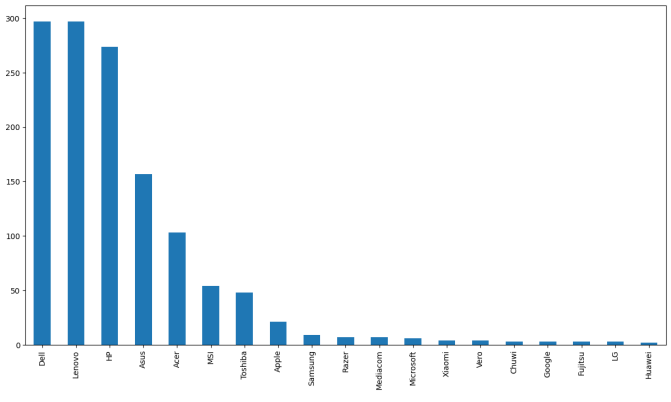

Fig. 3. Manufacturer distribution

The dataset has almost 50% of notebooks and 50% of other variants of laptops, which indicates the popularity of notebooks among buyers. Also, we see the prices of notebooks

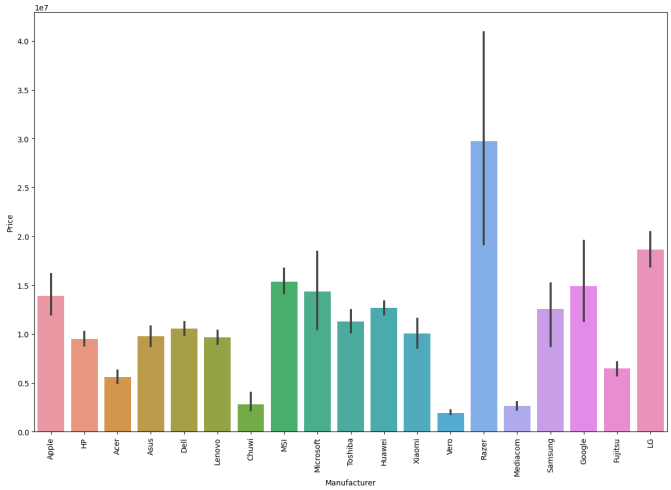

Fig. 4. Manufacturer distribution according to price

are the second lowest. Whereas the prices of workstations are highest, the instances of workstations present in our dataset are very few, which may lead our model to falsely analyze the prices of the workstation.

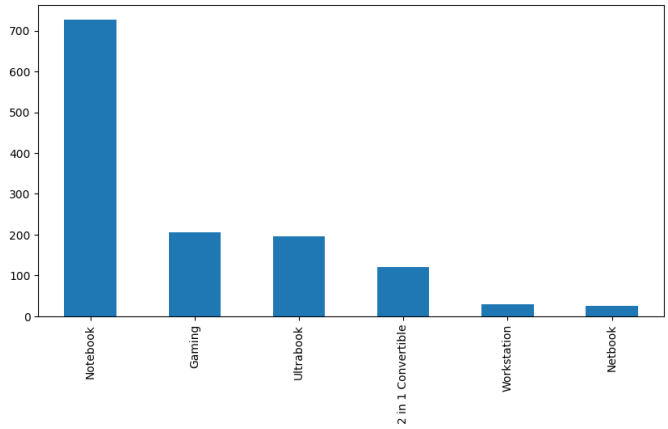

Fig. 5. Category distribution

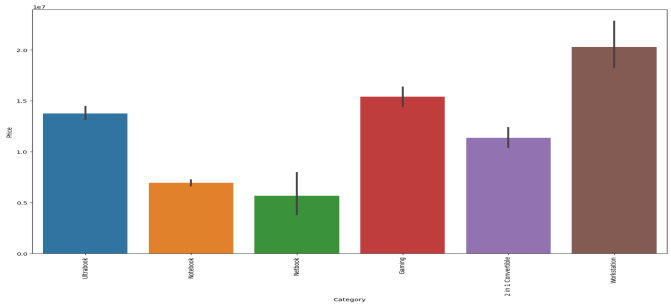

Fig. 6. Category distribution according to price

In the dataset we see that there is no clock speed present in some of the instances, so we have to impute those instances.

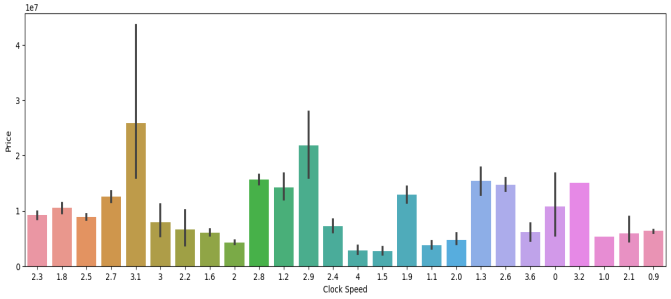

Fig. 7. Clock Speed distribution according to price

We see the Intel core i7 processor has the maximum number of instances present in this dataset, and the AMD processor has the lowest number of instances, even though the price of the Intel Core i7 is a lot higher than the AMD processor.

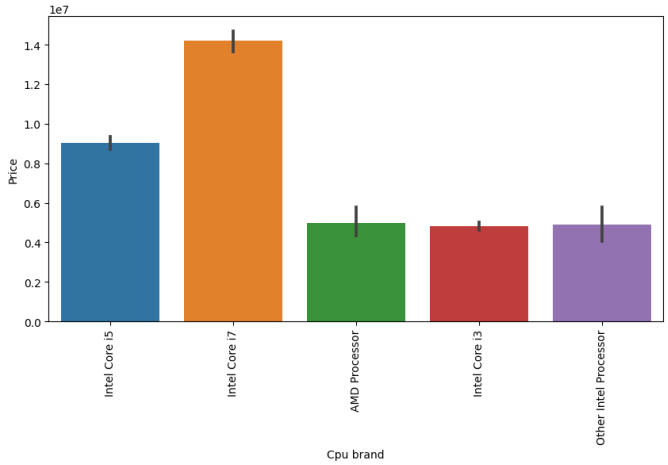

Fig. 8. Price distribution according to CPU

We see in our dataset, the distribution of RAM ranges from 2 GB to 64 GB, however, the most-selling laptops are 8 GB and then 4 GB. These values belong in between the 1st quantile to the 2nd quantile range. The prices of the laptops are proportional to the RAM.

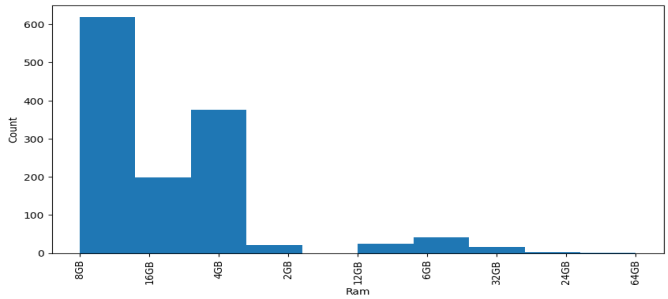

Fig. 9. Ram distribution

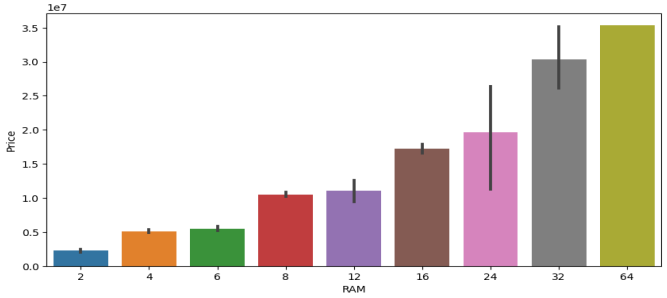

Fig. 10. Price distribution according to Ram

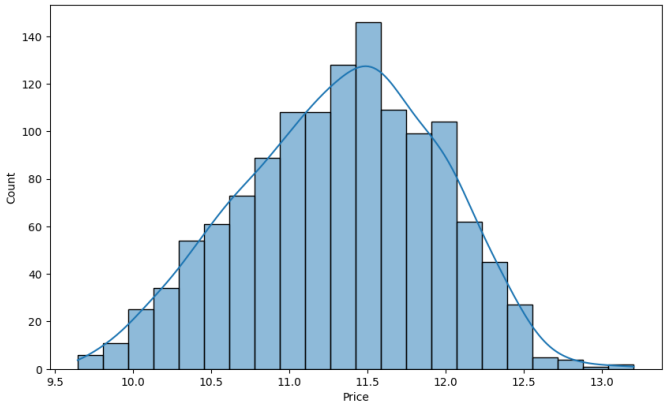

Fig. 11. Price distribution

V. Pre-processed Data

After processing the data, the dataset contains 15 columns and 1301 rows. The following columns were added or modified:

1) TouchScreen: A binary variable indicating whether the laptop has a touchscreen.
2) IPS: A binary variable indicating whether the laptop has an in-plane switching (IPS) display.
3) ppi: The pixel density of the laptop screen in pixels per inch.
4) Resolution: The resolution of the laptop screen in pixels.
5) Cpu brand: The brand of the CPU used in the laptop.
6) Clock Speed: The clock speed of the CPU in GHz.
7) HDD: The size of the hard disk drive (HDD) in the laptop.
8) SSD: The size of the solid-state drive (SSD) in the laptop.
9) OS: The name of the operating system used in the laptop.
10) GPU: The type of graphics processing unit (GPU) used in the laptop.
11) GPU: The type of graphics processing unit (GPU) used in the laptop. Weight: The weight of the laptop in kilograms.
12) Price: The price of the laptop in USD.
13) RAM: The amount of RAM in the laptop.

14) Manufacturer: The name of the laptop manufacturer.
15) Category: The category of the laptop.

VI. Types of Features

The dataset contains both categorical and numerical features. The categorical features include the manufacturer, model name, category, IPS(representing screen type, which is later on converted to numerical for ease of processing), CPU brand, GPU, and operating system. The numerical features include the screen size, RAM, storage, weight, and price. After processing, additional numerical features were added, such as clock speed, HDD, SSD, PPI, and Resolution.

A. Correlation Matrix

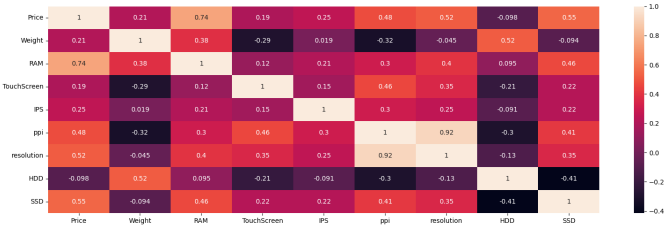

Fig. 12. Heatmap

VII. Handling Biasness

To ensure that the dataset is balanced, we checked if all unique classes have an equal number of instances or not. It was found that the operating system version was missing in 170 instances. To avoid any biases in the analysis, we decided to drop this column. Moreover, as the operating system is not a significant factor affecting laptop prices, it is acceptable to exclude it from the analysis. By taking this step, we have ensured that our dataset is balanced and free from any potential biases that could affect the accuracy of the results.

VIII. Data Pre-Processing

A. Faults:

During the data pre-processing stage, we encountered two faults. Firstly, we found that the operating system version was missing in 170 instances, which could potentially introduce biases in the analysis. Therefore, we decided to drop this column as it doesn't significantly affect the laptop price. Secondly, we noticed that several features such as the manufacturer, model name, category, IPS (representing screen type), CPU brand, GPU, and operating system were categorical variables, which needed to be transformed into numerical values for ease of processing.

B. Solution:

To address this, we converted some categorical features to numerical values manually. For instance, the IPS feature, which represented the type of screen, was converted to a numerical value by assigning a value of 1 to IPS screens and 0 to non-IPS screens. For the remaining categorical features, we applied one-hot encoding during column transformation to convert them into numerical features. This method helped us to transform the categorical features into numerical features without introducing biases in the analysis.

C. Feature Scaling:

During the feature scaling stage, we transformed two features: Storage and Price. Firstly, we converted the storage feature from TB (terabytes) to GB (gigabytes) to ensure that all values were on the same scale. This was important because the storage values ranged from a few hundred GB to several TBs, which could have affected the performance of some algorithms. Therefore, we multiplied the storage values by 1000 to convert TB to GB.

Secondly, we converted the price feature from INR (Indian Rupees) to USD (US Dollars) to ensure that the price values were in the same currency. This step was necessary because some of the algorithms might be sensitive to differences in currency values. To convert the price from INR to USD, which later on was converted to a logarithm for improving as the main feature scaling.

By performing feature scaling, we ensured that all features were on the same scale and in the same currency, which helped to improve the accuracy and performance of the regression models that we applied.

IX. Data Splitting

During the dataset splitting stage, we divided the dataset into two sets: a training set and a test set. To ensure that our model could generalize well to new data, we randomly split the dataset into a training set and a test set. We did not use stratified sampling because our dataset did not have any significant class imbalances.

We allocated 80% of the dataset to the training set and 20% to the test set. By doing this, we ensured that our model was trained on a sufficiently large amount of data, while still having a substantial amount of data to test its performance on. This helped to ensure that our model could generalize well to new data and provide accurate predictions on unseen data.

X. Regression Models

In this comparative analysis of various regression models on laptop price prediction, we tested four different regression algorithms:

A. Linear Regression

This is a simple regression algorithm that models the relationship between the dependent variable (price in our case) and one or more independent variables (features) by fitting a linear equation to the data.

B. K-Nearest Neighbors (KNN) Regression

This algorithm predicts the price of a laptop by finding the K nearest neighbors in the training set and taking the average of their prices.

C. Decision Tree Regression

This algorithm uses a decision tree to model the relationship between the dependent variable and the independent variables. It works by recursively partitioning the data into subsets based on the values of the independent variables, and then fitting a simple model (such as a constant value or a linear equation) to each subset.

D. Random Forest Regression

This algorithm combines multiple decision trees to improve the accuracy of the predictions. It works by randomly selecting subsets of the features and subsets of the data, and then fitting a decision tree to each subset. The final prediction is obtained by averaging the predictions of all the decision trees.

By testing these four different regression algorithms, we aimed to find out which algorithm was most suitable for the laptop price prediction problem and which one provided the best accuracy and performance on our dataset.

XI. Results

These are the results of a comparative analysis of four different regression models on the task of laptop price prediction. The four models tested were Linear Regression, KNN, Decision Tree, and Random Forest Regression.

The evaluation metrics used to measure the performance and error of each model were R2 Score, Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).

From the results, we can see that Random Forest Regression outperformed the other three models, with an R2 score of 0.8734, the lowest MAE of 0.1585, the lowest MSE of 0.0460, and the lowest RMSE of 0.2145.

Linear Regression had an R2 Score of 0.790, which indicates that the model can explain 79% of the variance in the data. The MAE, MSE, and RMSE were 0.210, 0.076, and 0.276, respectively.

KNN with a k value of 6 had an R2 Score of 0.825, which is better than Linear Regression. The MAE, MSE, and RMSE were 0.201, 0.064, and 0.252, respectively. The Decision Tree model had an R2 Score of 0.795, which is similar to Linear Regression. The MAE, MSE, and RMSE were 0.207, 0.074, and 0.273, respectively.

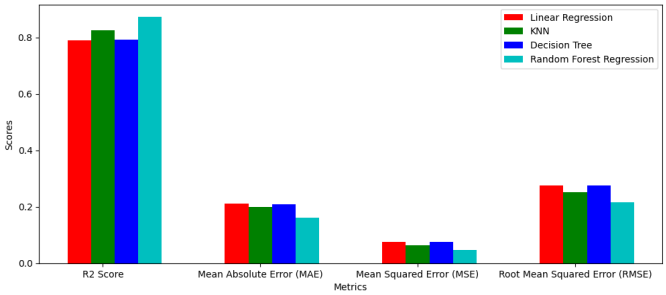

Fig. 13. Regression Model Comparison

XII. Conclusion

In this comparative analysis, we tested four regression models - Linear Regression, KNN, Decision Tree, and Random Forest Regression - on laptop price prediction. We used four different metrics - R2 Score, MAE, MSE, and RMSE - to measure the performance and error of each algorithm. The results show that the Random Forest Regression model outperforms the other three, with KNN performing second best.

XIII. Future Work/Extension

While Random Forest Regression and KNN proved to be better suited for this particular problem, there are still many other regression models that could be tested for price prediction. Additionally, further feature engineering could be done to improve the models' performance, such as incorporating external data sources like customer reviews or ratings. Another possible extension could be exploring different approaches to feature scaling or trying different splitting techniques for the dataset. Lastly, applying hyperparameter tuning to the models could further optimize their performance. Overall, there is still ample room for exploration and experimentation to improve the accuracy and reliability of laptop price-prediction models.