

INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT AKRUDI, PUNE

DOCUMENTATION ON

**“LAPTOP PRICE PREDICTION”**

e-DBDA May-2021

**Submited by:**

Group No. :12

Pooja Rathod(1344)

Deeksha Ingle(1317)

Mr. Prashant Karhale

Mr.Akshay Tilekar

**Centre Coordinator**

**Project Guide**

**INDEX**

**Table of Contents**

1. Introduction…………………………………………………………………………..6

1.1 Purpose………………………………………………………………….........7

1.2 Scope of the project………………………………………………………….7

1.2.1 Initial functional requirement ………………………………………....7

1.2.2 Initial non functional requirement……………………………………..7

1.3 Software Life Cycle Model………………………………………………….8

1.4 Overview of document………………………………………………………10

1. Overall Description………………………………………………………………….11

2.1 System Environment………………………………………………………...11

2.2 Functional requirement specification:……………………………………….11

2.3 Non-functional requirement…………………………………………………11

1. Requirement Specification…………………………………………………………..12

3.1 External requirement specification………………………………………….12

3.2 Detailed Non-Functional Requirements…………………………………….12

3.2.1 Functional Requirement……………………………………………….12

3.2.2 Hardware Requirement………………………………………………...12

3.2.3 Software Requirement…………………………………………………12

1. System Design………………...……………………………………………………..15 4.1 Basic Data Exploration……………………………………..………………..15

4.2 Feature Engineering…...…………………………………………………......16

4.3 Explanatory Data Analysis…………………………………………………17

4.4 Data Preprocessing………….…...………………………………………….17

4.5 Modelling…………………...………………………….…………………….17

4.5.1 Random Forest Regressor.……………………………………………17

4.5.2 Decision Tree Regressor………………………………………………17

4.5.3 Linear Regression Model.……………………………………………17

4.5.4 XGBoost Model………..………………………………………………17

1. Conclusion ………………………….……………………………………………….20
2. Future Scope………………………………………………………………………....21
3. References……………………………………………………………………………

**Chapter 1**

**Introduction**

Price is the most effective attribute of marketing and business. The very first question of costumer is about the price of items. All the costumers are first worried and thinks “If he would be able to purchase something with given specifications or not”. So to estimate price at home is the basic purpose of the work. This paper is only the first step toward the above mentioned destination. Artificial Intelligence-which makes machine capable to answer the questions intelligently- now a days is very vast engineering field. Machine learning provides us best techniques for artificial intelligence like classification, regression, supervised learning and unsupervised learning and many more. Different tools are available for machine learning tasks like MATLAB, Python, cygwin, WEKA etc. We can use any of classifiers like Decision tree , Naïve Bayes and many more. Different type of feature selection algorithms are available to select only best features and minimize dataset. This will reduce computational complexity of the problem. As this is optimization problem so many optimization techniques are also used to reduce dimensionality of the dataset. Mobile now a days is one of the most selling and purchasing device. Every day new mobiles with new version and more features are launched. Hundreds and thousands of mobile are sold and purchased on daily basis. So here the mobile price\_class prediction is a case study for the given type of problem i.e finding optimal product. The same work can be done to estimate real price of all products like cars, bikes , generators, motors, food items, medicine etc.

Many features are very important to be considered to estimate price of mobile. For example Processor of the mobile. Battery timing is also very important in todays busy schedule of human being. Size and thickness of the mobile are also important decision factors. Internal memory, Camera pixels, and video quality must be under consideration. Internet browsing is also one of the most important constraints in this technological era of 21st century. And so is the list of many features based upon those, mobile price is decided. So we will use many of above mentioned features to classify whether the mobile would be very\_economical, economical, expensive or very\_ expensive. The structure of the paper is as follows. Next section is review of previous work.3 rd Section contains Methodology and Experimental procedure. Section 4 is the summary of the results. Comparative study is done in section 5. After that paper is concluded in section 6. Outcomes of the work are discussed in section 7. At last in 8th section some suggestions about future work are given.

* 1. **Purpose**

To predict “If the Laptop with given features will be Economical or Expensive” is the main motive of this research work. Real Dataset is collected from Kaggel . Different feature selection algorithms are used to identify and remove less important and redundant features and have minimum computational complexity. Different classifiers are used to achieve as higher accuracy as possible. Results are compared in terms of highest accuracy achieved and minimum features selected. Conclusion is made on the base of best feature selection algorithm and best classifier for the given dataset. This work can be used in any type of marketing and business to find optimal product(with minimum cost and maximum features).

**1.2 Scope of the project**

**1.2.1 Initial functional requirement will be:-**

* Selecting the algorithm meeting requirement.
* Choosing the optimum algorithm form set of algorithm.
* Testing it on the datasets.
* After getting the result if the result is low change the hyperparameters.
* Out of all result get best of all.

**1.2.2 Initial nonfunctional requirement will be:-**

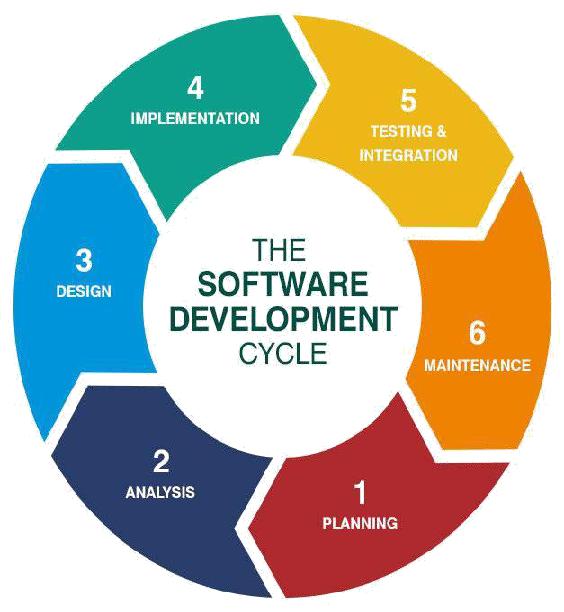
* Getting the large datasets which can provide developer enough image to train the model.
* Maintain the minimum variance& bias so the model is successfully work.
* Avoid the underfitting and overfitting.

|  |  |  |  |
| --- | --- | --- | --- |
| **Terms** | **Definitions** |  |  |
| Dataset | Data for training and testing for the model |  |  |
| Variance | Difference between the training and testing accuracy |  |  |
| Bias | Both learning and training accuracy is low |  |  |
| Overfitting | Model is very complex |  |  |
|  |  |
| Underfitting | Model is bias |  |  |
| Developer | Who is developing the model |  |  |
| Review | A written recommendation about the appropriateness of an Product for |  |  |
|  | selling and buying may include suggestions for improvement. |  |  |
| Reviewer | A person that examines an Product and has the ability to recommend |  |  |
|  | approval Product for buying or to request that changes be made in the |  |  |
|  | Product. |  |  |
| Software | A document that completely describes all of the functions of a proposed |  |  |
| Requirement | system and the constraints under which it must operate. For example, |  |  |
| Specification | this document |  |  |
| User | Reviewer |  |  |

**Table 1.2.2.1: Terms and Definitions.**

**1.3 Software Life Cycle Model:**

In order to make this Project we are going to use Classic LIFE CYCLE MODEL .Classic life cycle model is also known as WATER FALL MODEL. The life cycle model demands a Systematic sequential approach to software development that begins at the system level and progress through analysis design coding, testing and maintenance.



**Figure 1.3.1. Software life cycle model**

**The Classic Life Cycle Model**

The waterfall model is sequential software development process, in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of conception initiation, Analysis, Design (validation), construction. Testing and maintained.

1. **System Engineering and Analysis:**

Because software is always a part of larger system work. Begins by establishing requirement for all system elements and Then allocating some subset of these requirement to the software system Engineering and analysis encompasses the requirement gathering at the system level with a small amount of top level design and analysis.

1. **Software requirement Analysis:**

The requirement gathering process is intensified and focused specifically on the software Requirement for the both system and software are discussed and reviewed with the customer. The customer specifies the entire requirement or the software and the function to be performed by the software

**3) Design:**

Software design is actually a multi-step process that focuses on distinct attributes of the program data structure, software architecture, procedural detail and interface of the software that can be assessed or quality before coding begins .Like requirement the design is documented and becomes part of the software.

**4) Coding:**

The design must be translated into a machine readable form. The coding step performs this task. If design is programmed in a detailed manner, coding can be accomplished mechanically.

**5) Testing:**

Once code has been generated programmed testing begins. The testing process focuses on the internals of the software ensuring that all statement have been tested and on the functional externals hat is conducting tests to uncover the errors and ensure that defined input will produce the results that agree with the required results.

* 1. **Unit testing:**

In computer programming, Unit testing is software Verification and validation method where the programmer gains confidence that individual units of source code are fit to use A unit is the smallest testable part of an application. In procedural programming a unit may be an individual programmed, function, procedure, etc. while in object-oriented programming, the smallest Unit is a class, which may belong to a base/super class abstract class or derived/child class.

**5.2 Benefits:**

The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. A unit test provides a strict written contract that the piece of code must satisfy.

**5.3 Documentation:**

Unit testing provides a sort of living documentation of the system. Developers looking to learn what functionality is provided by a unit and how to use it can look at the tests to gain a basic understanding of the unit API.

**5.3.1 Limitation of unit testing**:

Testing cannot be expected to catch error in the program –It is impossible to evaluate all execution paths for all but the most trivial programs. The same is true for unit testing. Additionally, by unit testing only types the functionality if the units themselves.

**6) Maintenance:**

Software will undoubtedly undergo change after it is Delivered to the customer .Change will occur because errors have been encountered because the software must be able adopted to accommodate changes in its external environment because the customer requires functional or performance enhancement enhancements. The classic life cycle is the oldest and most widely used paradigm or software engineering

**1.4 Overview of document**

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

**Chapter 2**

**Overall Description**

The Laptop Price Detection Dataset has the huge information about the configuration of different System. This dataset contains about 20K Row Values and also have column named as Company TypeName (Gaming, Notebook, Ultrabook, 2 in 1 convertible, WorkStation), ScreenResolution, Inches, CPU, RAM, Memory, Gpu, Opsys, Weight, Price.

|  |
| --- |
|  |

**2.1 System Environment**

The system environment is evolving around the main thing which is the Prediction of Price Based on the Configurations of Laptop

**2.2 Functional requirement specification:**

1. **Use case**- Prediction of Price for the Laptop of a particular System to help theConsumer to buy laptop according to configurations needed by a consumer

**Brief Description:**

In this case Prediction, the Model is carried out by using libraries such as Pandas, numpy, seaborn scikit-learn, matplotlib.py, ngrok. Keras.

**Initial Step by step description:**

**Step 1 –** Take Dataset as Input.

**Step 2 –** Perform EDA and Data visualization techniques.

**Step 3 –** Set Price as Target Column and split the Train and Test Data.

**Step 4 –** Perform ML models and Check best ML model with High R2 score and Low MSE value.

**Step 5 –** Deployed the Model on ngrok.

**2.3. Non-functional requirement:**

Python libraries such as Pandas, numpy, seaborn scikit-learn, matplotlib.py, ngrok, keras, os installation of the system.

**Chapter 3**

**Requirement Specification**

**3.1 External requirement specification:**

The only link to an external system is the link to the image with customer. The Administrator believes that a site member is much more likely to be an effective reviewer and has imposed a membership requirement for a Reviewer.

**3.2 Detailed Non-Functional Requirements:**

**3.2.1 Functional Requirement:**

First of all model should be successfully trained by developer.

Download and install Anaconda (windows version).

**3.2.2 Hardware Requirement:**

* **Processor:** Intel Dual Core
* **RAM:** Minimum 1GB
* **OS:** Windows,Linux.MacOS

**3.2.3 Software Requirement:**

**Anaconda Navigator**

Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda® distribution that allows you to launch applications and easily manage conda packages, environments and channels without using command-line commands. It is available for Windows, macOS and Linux.

**Following libraries of python should be install:**

1. **TensorFlow –** pip install tensorflow (keras uses TensorFlow as backend).
2. **Keras –** pip install keras (to build our classification model).
3. **ngrok–** pip install ngrok (for application deployment).

**Tensorflow:**

A couple of years ago, deep learning started to outperform all other machine learning algorithms when giving a massive amount of data. Google saw it could use these deep neural networks to improve its services:

* Gmail
* Photo
* Google search engine

They build a framework called **Tensorflow** to let researchers and developers work together on an AI model. Once developed and scaled, it allows lots of people to use it. It was first made public in late 2015, while the first stable version appeared in 2017. It is open source under Apache Open Source license. You can use it, modify it and redistribute the modified version for a fee without paying anything to Google.

**TensorFlow Architecture(** Tensorflow architecture works in three parts**):**

* Preprocessing the data
* Build the model
* Train and estimate the model

**Keras:**

Keras is an Open Source Neural Network library written in Python that runs on top of Theano or Tensorflow. It is designed to be modular, fast and easy to use. It was developed by François Chollet, a Google engineer. Keras High-Level API handles the way we make models, defining layers, or set up multiple input-output models.

**ngrok:**

ngrok is  a cross-platform application that enables developers to expose a local development server to the Internet with minimal effort.

**Os module:**

The OS module in python provides functions for interacting with the operating system. OS, comes under Python’s standard utility modules. This module provides a portable way of using operating system dependent functionality.

**Chapter 4**

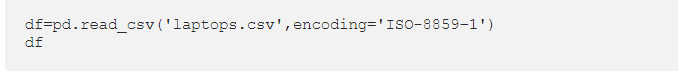
**System Design**

**4.1 Flowchart of the System:**

The flowchart of the algorithm is represented in Figure 4.1.1.

# **4.1 Basic Data Exploration**

After loading the dataset via Pandas, we can see a list of laptops and specs that are associated with each laptop.





Looking at the dataset, we can see that some columns such as ScreenResolution and Cpu have alphanumeric data while other features consist of purely numerical or alphabetical values. These data would need to be filtered and engineered later.

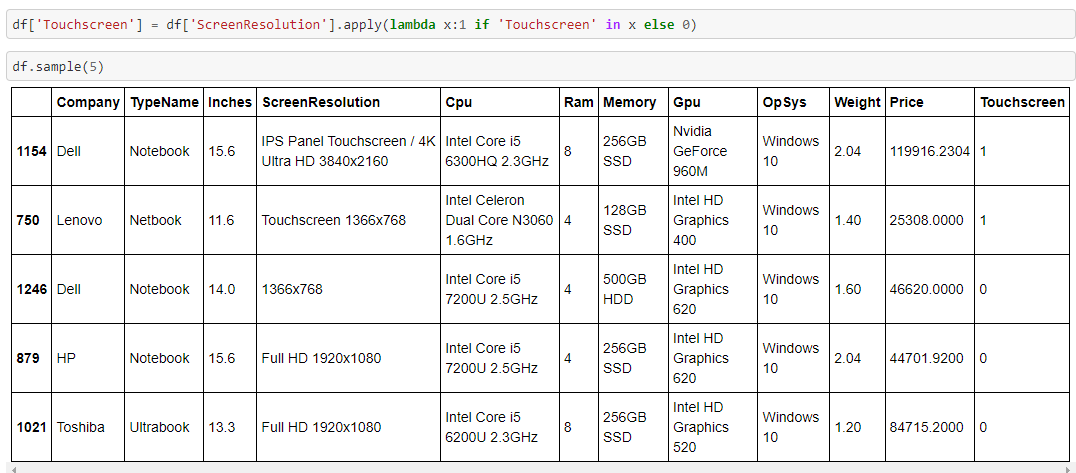
To avoid any complications and error-prone predictions, useless features such as “Unamed:0”, “Company” and “Product” will be removed from the dataset. (Having an apple laptop with windows OS doesn’t seem to make any sense for price prediction.)

We would then check for any missing values that are present in the dataset. We wouldn’t want any errors to result from analysis or modeling later.

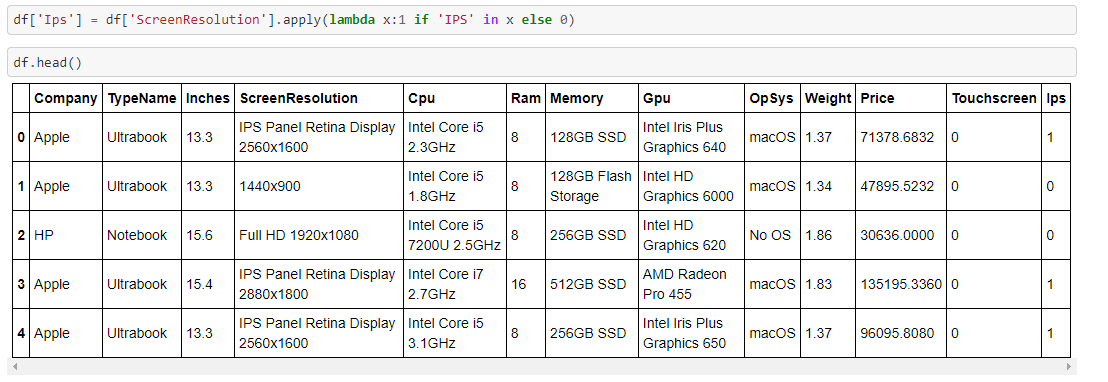
# **4.2 Feature Engineering**

We would now extract and reorganize our data to better understand the underlying factors that contribute to the price of laptops.

If we take a look at the ScreenResolution column, there seems to be laptops with touchscreen capabilities. Since touchscreen laptops are known to be more expensive than those without them, a TouchScreen feature would be added to mark laptops with such capabilities.

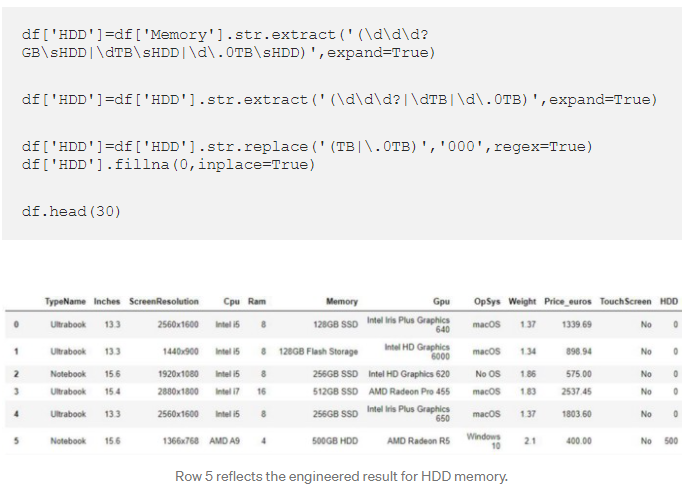


We would then extract and replace the screen resolution column with their respective pixel count using regular expressions. I find regular expressions incredibly useful when it comes to extracting/filtering alphanumeric values.



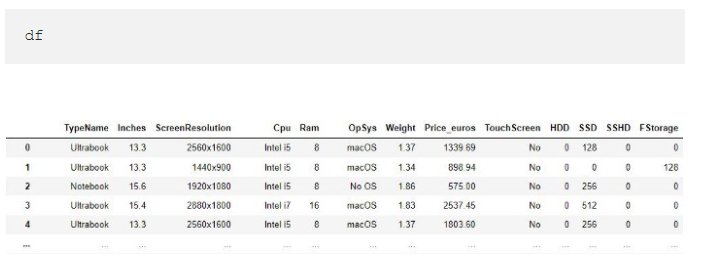
We would then apply the same process for engineering the Cpu, Ram, and Weight features. Our goal is to minimize or remove any units and words that are not essential for analysis later.

Now comes the most tiring part of feature engineering, dealing with memory feature. Upon closer inspection, the memory column contains various types of memory (SSD, HDD, SSHD, and Flash Storage). We would need to create 4 additional columns representing different memory types and extract their memory capacities individually. (Additional processing needs to be done for laptops having double memory configuration that uses the same memory types.( EX: 256GB SSD + 512GB SSD). This could be done using a similar process shown above.



Having those memory configurations handled, I’ve decided to drop the GPU column entirely as it contains a high variability of GPUs. Intel GPUs are integrated GPUs, Nvidia GPUs are discrete while AMD GPUs are either integrated or discrete. Labeling and classifying each would take significant effort and time which may or may not contribute to the modeling process later on.

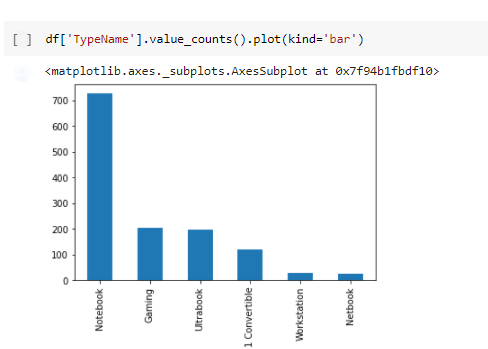
Our fully feature engineered dataset would look like this:



**4.3 Exploratory Data Analysis(EDA):**

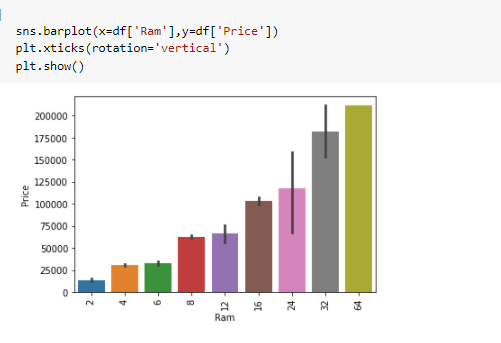
Using our feature-engineered dataset, we can now plot graphs and compute tables to visualize how each feature relates to the variability of laptop prices.

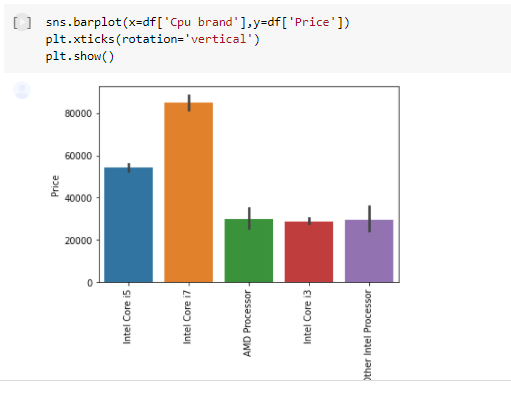
By using the .barplot method imported from Matplotlib, we can test and verify our hypothesis or initial opinions on how some features will affect the pricing of laptops. Here’s an illustration of plotting a barplot for the feature TypeName (type of laptop):



From the barplot above, we can rectify and conclude that, on average, workstation and gaming laptops have a higher price than other types of laptops. This is to be expected as these types of laptops often have better spec configurations (better CPU, more memory, etc) to meet the demands of clients in the professional workspace. Notebooks and netbooks have lower prices due to their low-powered configurations.

Higher Ram capacities also reflect higher prices in laptops:



Plotting bar graphs on the Cpu features shows some interesting results. In general, higher-powered processors should be priced higher than lower-powered ones. The prices for intel processors generally follow this pattern (Xeon > i7>i5>i3) and the same principles apply to AMD CPUs as well (Ryzen > AMD A series> E series). The barplot obtained from the dataset shows otherwise.

# **4.3 Data Preprocessing:**

In this section, we will relabel and convert categorical features into numerical features. This is essential for training our ML models as ML models only accept numerical values as inputs.

Starting off, we identify features that are non-numerical (Object type) and compute their cardinalities (categories present in each feature).

Knowing that the TouchScreen feature only has 2 categories, we can use One-hot-encoding can be used Using the Scikit-learn’s label encoding function, the variables present in TouchScreen (‘No’, ‘Yes’) will be encoded into 0s and 1s



Label encoding also handles features with high cardinalities. Applying label encoding to the Cpu feature, the label encoded values (associated with their pre-encoded variables) are recorded for predicting purposes later.

Other features with slightly lower cardinality were encoded via the one-hot-encoding method. Through the use of pandas’ .getdummies() method, a new column will be created to indicate the presence of each categorical variable.

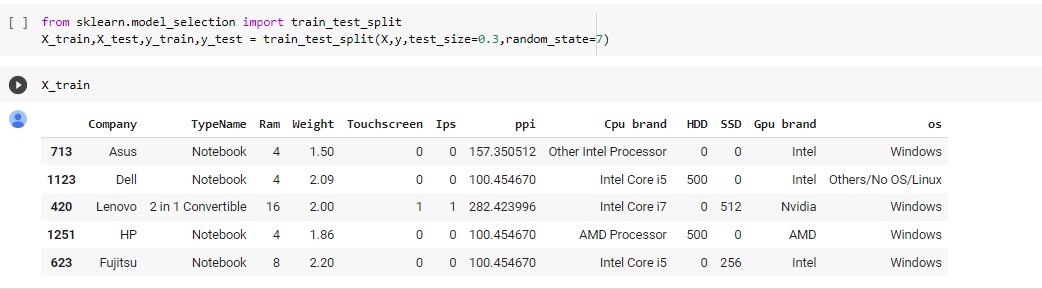
After applying One-hot-encoding to TypeName and OpSys features, we will use manual encoding to deal with features with high cardinality if we know the order of variables.

We can use python’s dictionary and mapping methods to specify and encode each category based on their magnitude/order.

# **4.4 Modeling:**

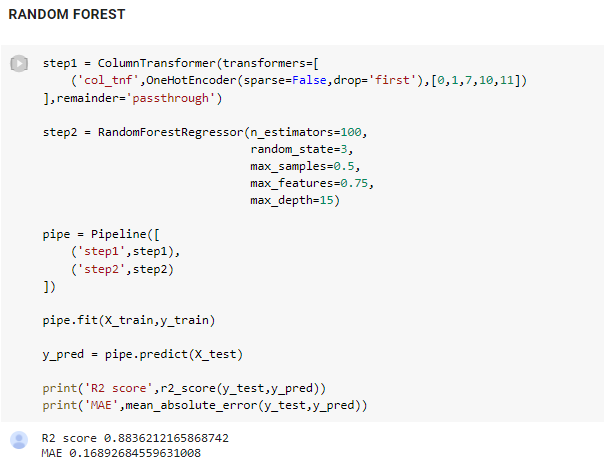
After loading the preprocessed .csv dataset, we identify our dependant variable (Price) and allocate a separate data frame for the target variable.

We can then split the dataset for training and validating the performance of the models we are going to apply later. Roughly 30% of the data would be used to test our ML models later.

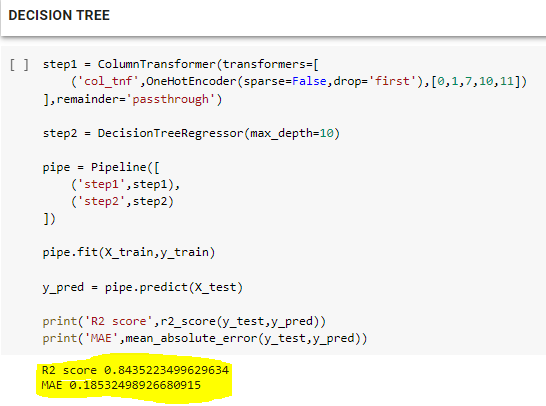


Next, we begin to train and validate the performance for different models. The two main metrics used for estimating the performance of our models would be the R-squared score and the Mean absolute error (MAE). In general, we want to achieve a higher R-squared score and lower MAE score with our models.

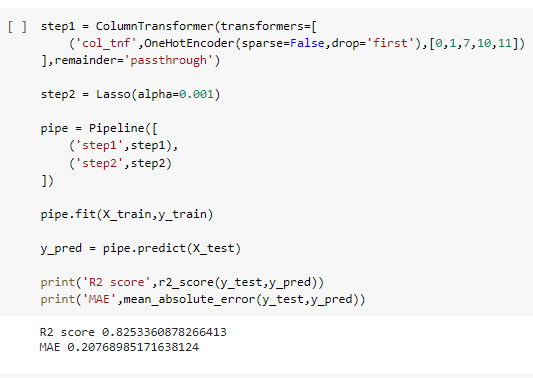
RandomForestRegressor with different number of nodes:



**Training with DecisionTreeRegressor**:



**Linear Regression Model:**





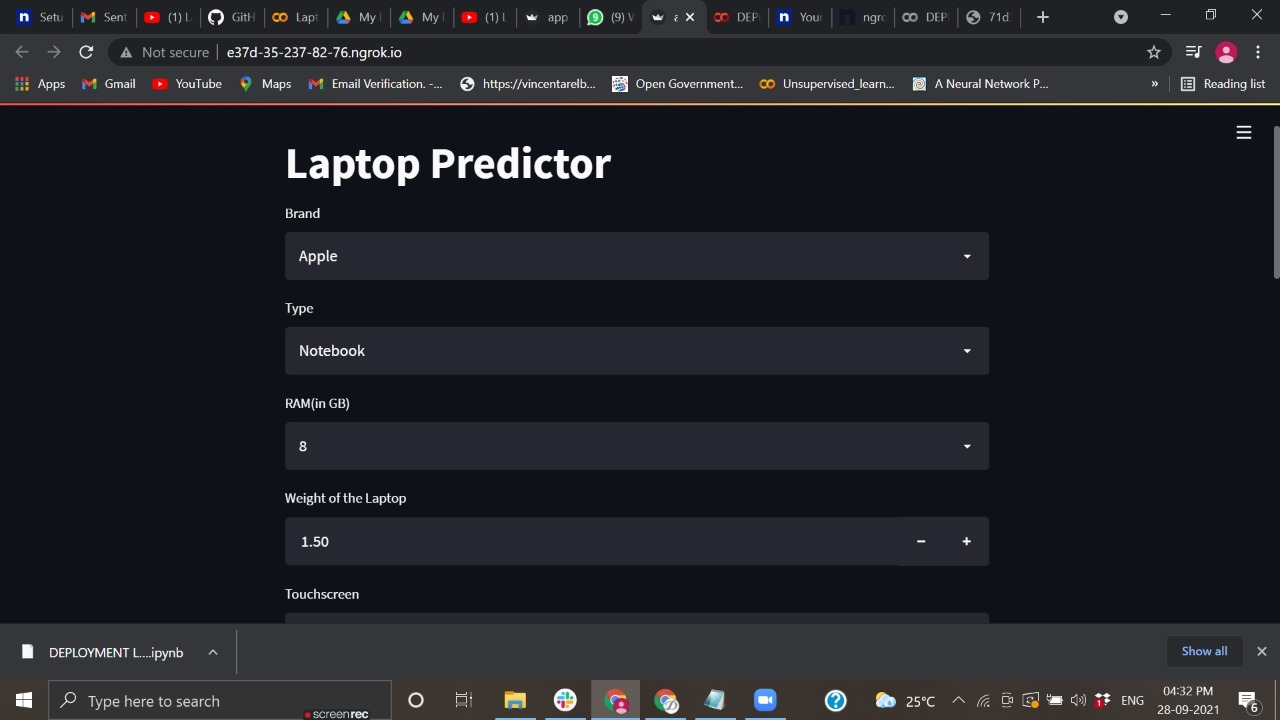
**XGBoost Model:**



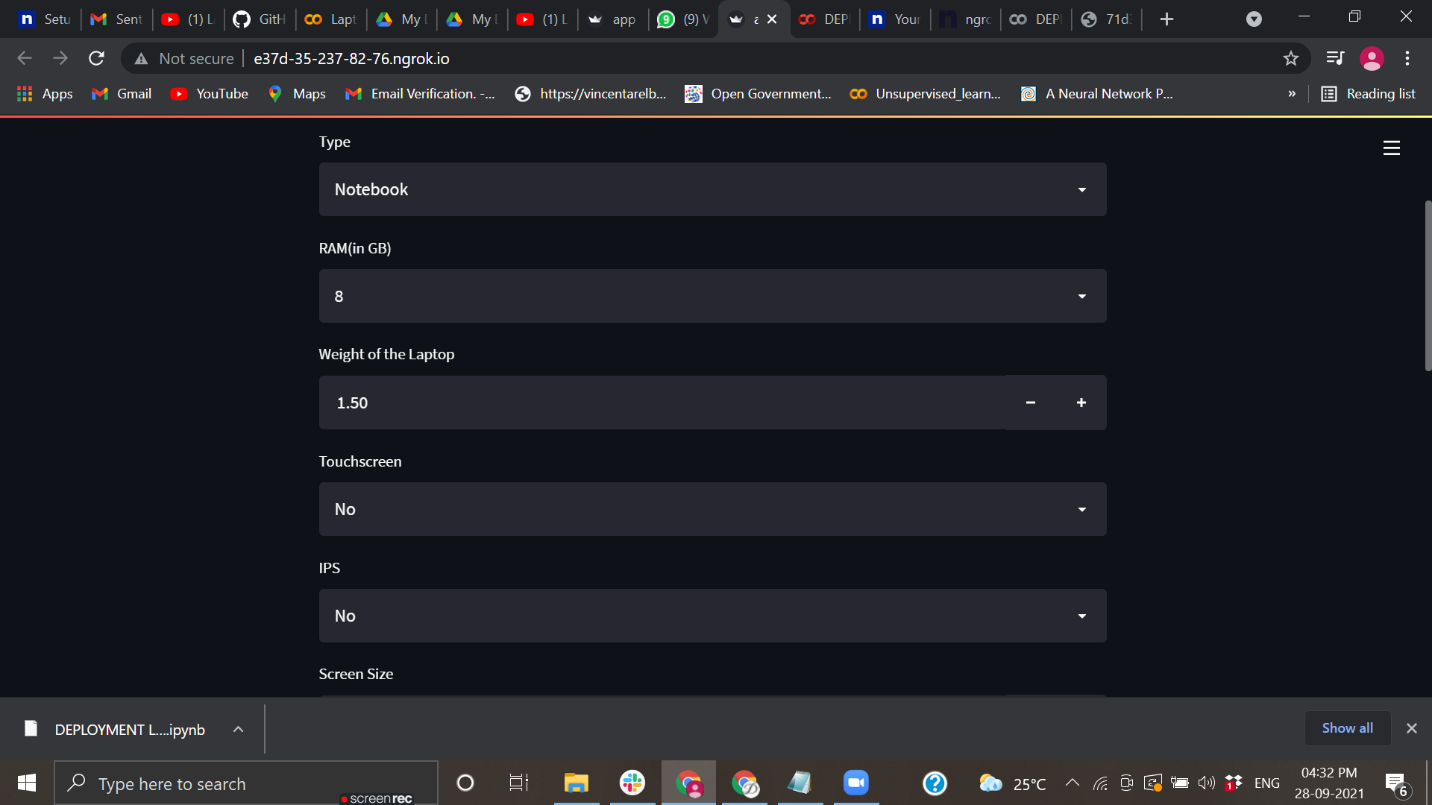
**Chapter 5**

**Results**

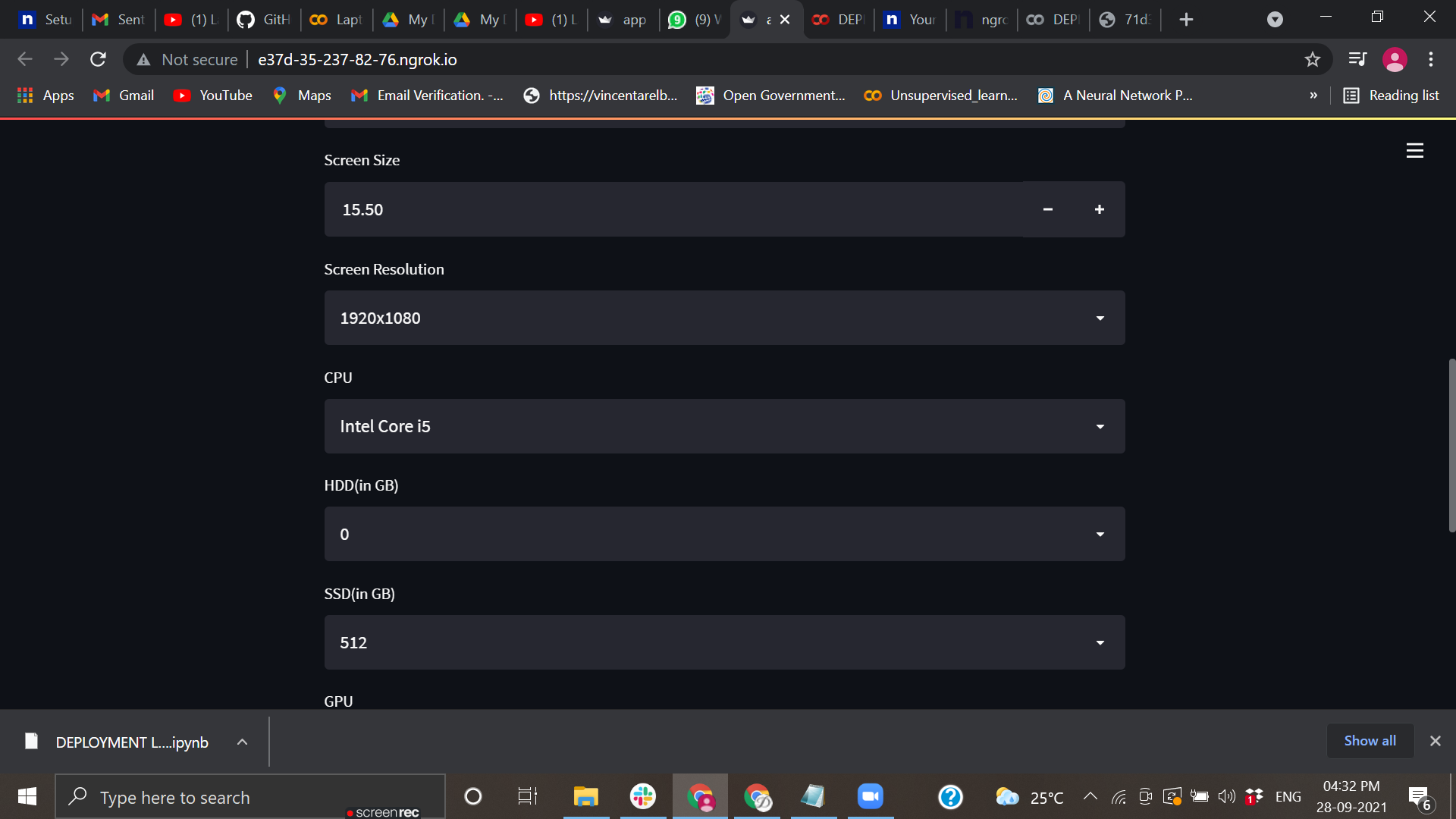
**Output:**



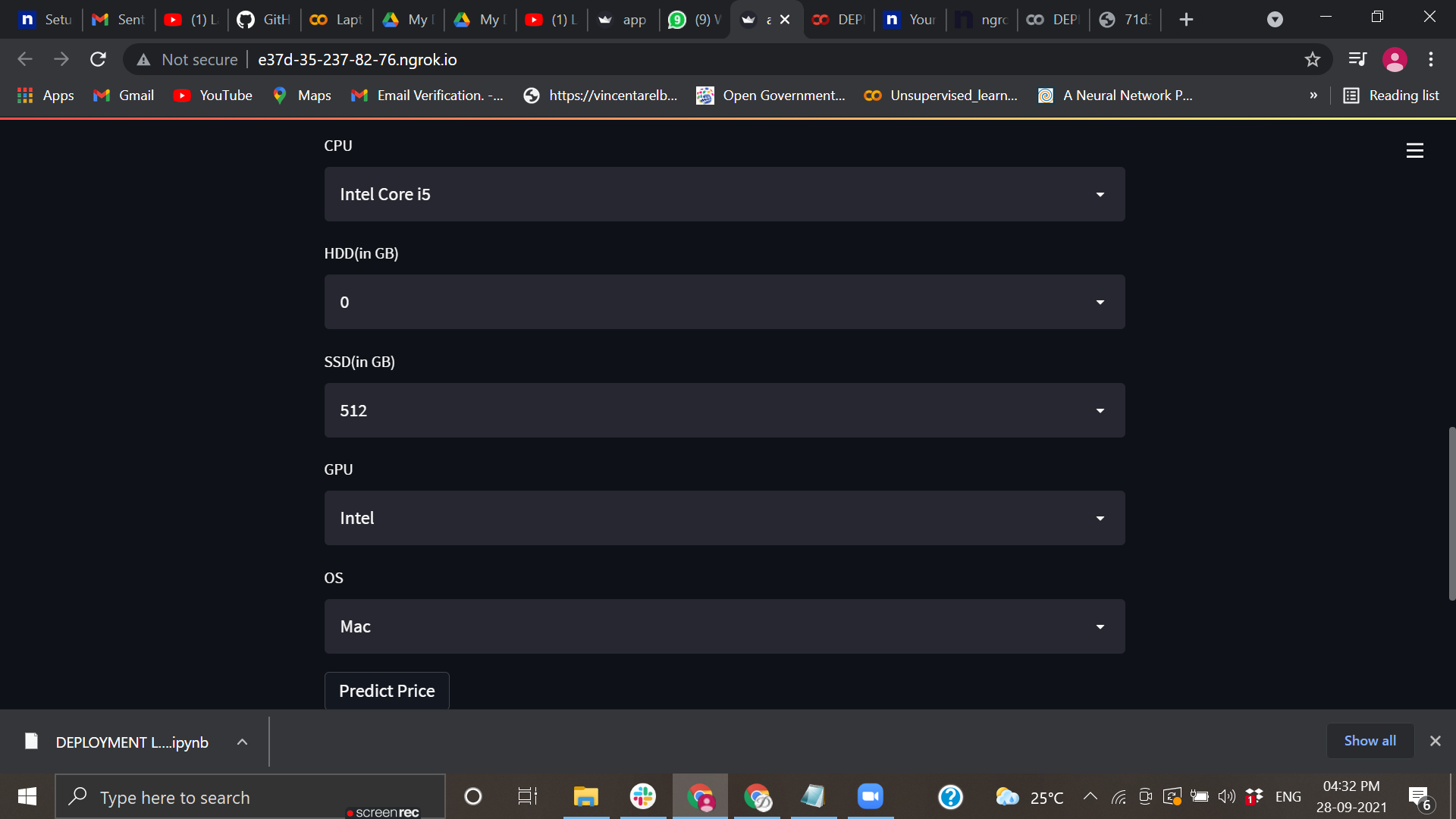
**Screenshot 5.1:**



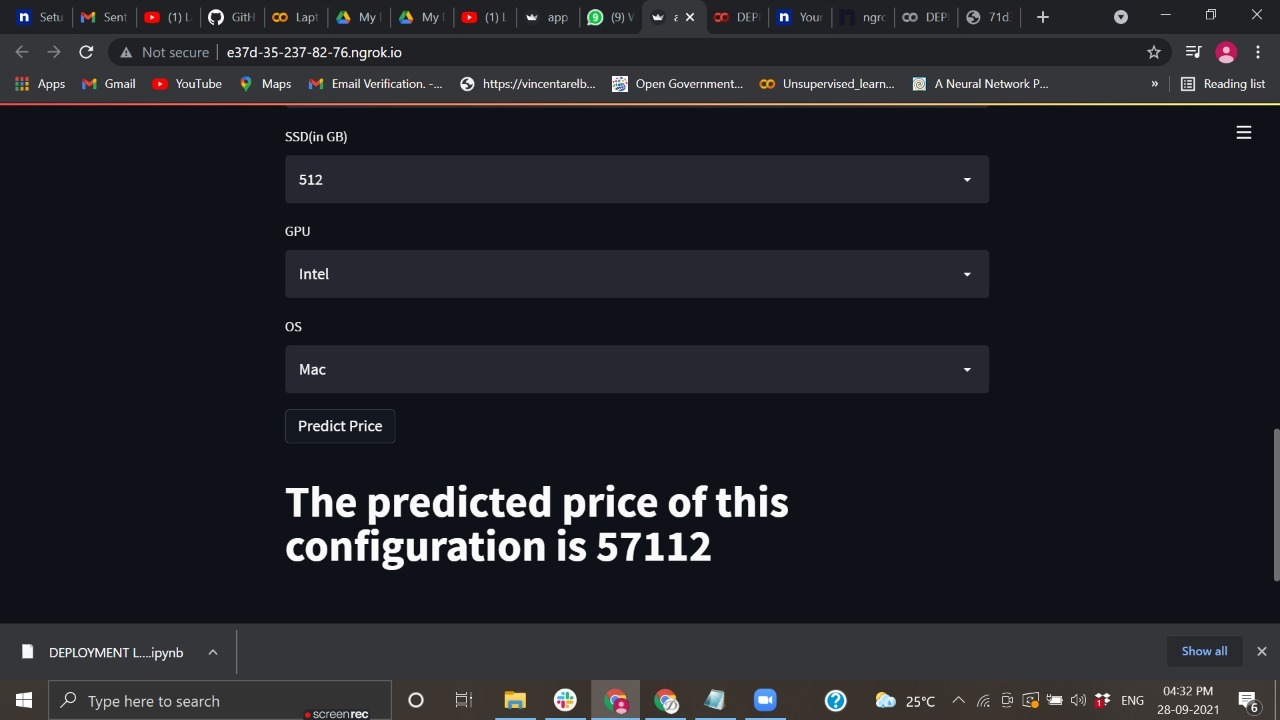
**Screenshot 5.2:**



**Screenshot 5.3:**



**Screenshot 5.4:**



**Screenshot 5.5:**

After Selecting all the values from Dropdown of the Brand ,Type ,Ram, Weight, Touch Screen ,IPS, Screen size, Screen Resolution , CPU,SDD, HDD,GPU and OS ,All these values are stored in an array with help of numpy libraries and the model is imported with the help of pickle which serialize and deserialize the model and to make new predictions hence predict function is used to predict the data for configurations given by consumers.

**Chapter 6**

**Conclusion**

This work can be concluded with the comparable results of both Feature selection algorithms and classifier except the combination of WrapperattributEval and Descision Tree J48 classifier. This combination has achieved maximum accuracy and selected minimum but most appropriate features. It is important to note that in Forward selection by adding irrelevant or redundant features to the data set decreases the efficiency of both classifiers. While in backward selection if we remove any important feature from the data set, its efficiency decreases. The main reason of low accuracy rate is low number of instances in the data set. One more thing should also be considered while working that converting a regression problem into classification problem introduces more error

**Chapter 7**

**Future Scope**

* More sophisticated artificial intelligence techniques can be used to maximized the accuracy and predict the accurate price of the products.
* Software or Mobile app can be developed that will predict the market price of any new launched product.
* To achieve maximum accuracy and predict more accurate, more and more instances should be added to the data set. And selecting more appropriate features can also increase the accuracy. So data set should be large and more appropriate features should be selected to achieve higher accuracy.

**References**

1. Sameerchand Pudaruth . “Predicting the Price of Used Cars using Machine Learning Techniques”, International Journal of Information & Computation Technology. ISSN 0974-2239 Volume 4, Number 7 (2014), pp. 753- 764
2. Shonda Kuiper, “Introduction to Multiple Regression: How Much Is Your Car Worth? ” , Journal of Statistics Education · November 2008
3. Mariana Listiani , 2009. “Support Vector Regression Analysis for Price Prediction in a Car Leasing Application”. Master Thesis. Hamburg University of Technology

Limsombunchai, V. 2004. “House Price Prediction: Hedonic Price Model vs. Artificial Neural Network”, New Zealand Agricultural and Resource Economics Society Conference, New Zealand, pp. 25-26. 2004

1. Introduction to dimensionality reduction, A computer science portal for Geeks. <https://www.geeksforgeeks.org/dimensionality-reduction/>
2. InfoGainAttributeEval-Weka Online available from [http://weka.WrapperattributEval/doc.dev/weka/attributeS election/InfoGainAttributeEval.html](http://weka.WrapperattributEval/doc.dev/weka/attributeS%20election/InfoGainAttributeEval.html)
3. <https://www.pyimagesearch.com/2017/05/08/drowsiness-detection-opencv/>
4. Face Detection & Recognition  (English, Paperback, Danti Ajit)
5. Eye Tracking  (English, Paperback, Holmqvist Kenneth)