JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA

Edge Computing Hardware Devices Analysis with AI/ML Techniques

Aviral Tanwar 20103094 B4

Bachelor of Technology In Computer Science Engineering

Internship Company/Organization

INDIAN INSTITUTE OF INFORMATIONAL TECHNOLOGY, JODHPUR

Duration: 2 weeks
Supervisor: Dr Binod Kumar

ACKNOWLEDGEMENT

I would like to extend my heartfelt gratitude to everyone who contributed to the successful completion of my summer internship and this report.

I am immensely thankful to Dr. Binod Kumar for their guidance, support, and mentorship throughout my internship at IIT, JODHPUR. Their expertise and encouragement played a significant role in shaping my project and enhancing my skills.

I would like to acknowledge the support of my professors at JIIT for their continuous encouragement and for imparting knowledge that contributed to the success of my internship project. Additionally, I extend my thanks to my friends and family for their unwavering support and understanding, which motivated me to excel in my internship.

While my gratitude is profound, any errors or omissions in this acknowledgment are entirely my own.

Sincerely,
Aviral Tanwar

INTERNSHIP CERTIFICATE

COMPANY OVERVIEW

3.1 what To write in it?

The purpose of this report is to present a comprehensive analysis of edge computing hardware devices, with a specific focus on

LIST OF FIGURES

TABLE OF CONTENTS

EXECUTIVE SUMMARY

1.1 Summary:

The report presents an analysis of edge computing hardware devices, focusing on smartphones, using Artificial Intelligence (AI) and Machine Learning (ML) techniques. The primary objective is to identify the features utilized in various smartphones within a particular price range. The research also expands to include a wide range of smart devices beyond phones, predicting their prices based on their features.

1.2 Problem Statement:

"To identify the kind of features being used in different smartphones with ML for a particular price-range."

1. Feature Analysis and Selection: Conduct a comprehensive examination of various smart devices, including smartphones, tablets, and smartwatches, to identify the most influential features impacting their prices.
2. Device Type Classification: Develop an AI-based model capable of accurately classifying smart devices into distinct categories (e.g., smartphones, tablets, smartwatches) based on their unique feature combinations.
3. Price Prediction: Implement regression models tailored to each device type to predict their respective prices with enhanced accuracy and precision.

1.3 Methodology:

To accomplish the objectives, the following approach was adopted:

1. Data Collection: The creation of a comprehensive dataset was the first step, involving the collection of relevant data on smartphones and other smart devices. Python was employed to perform web scraping on multiple websites to extract the required information, which was subsequently stored in a CSV file. Many different comparing sites were used so as to ease the complications and were stored in many different csv files which were then concatenated in one as to create

one dataset. The new testing data were required and hence again python scraping was used and stored in another csv file.

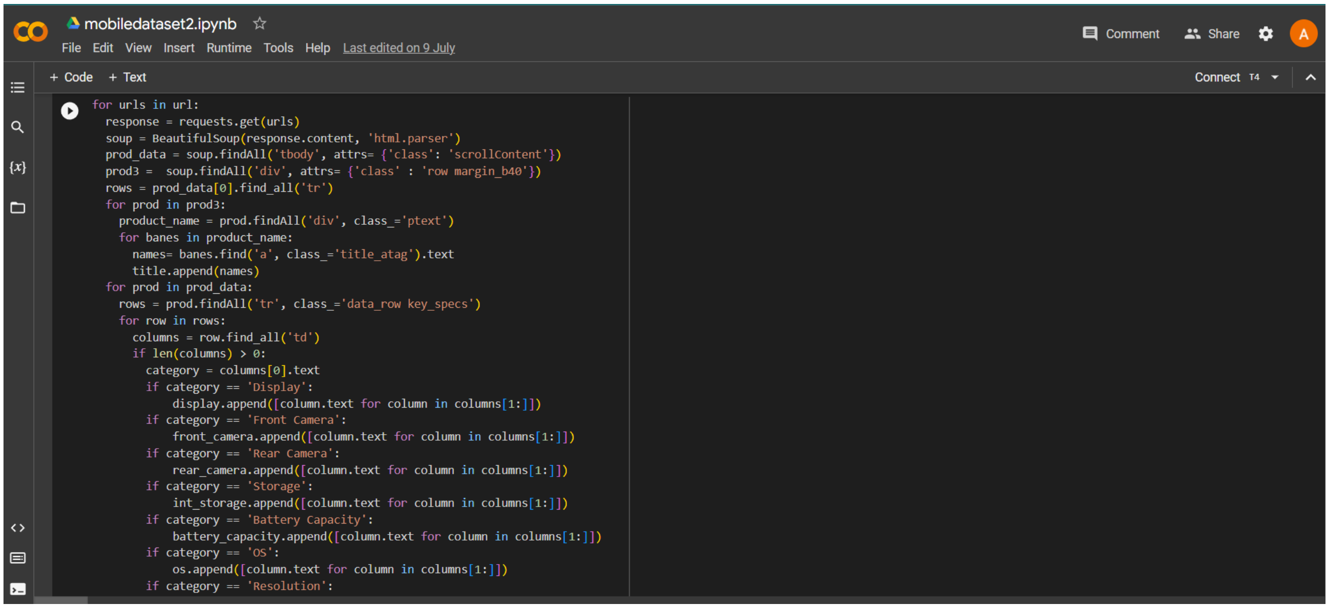
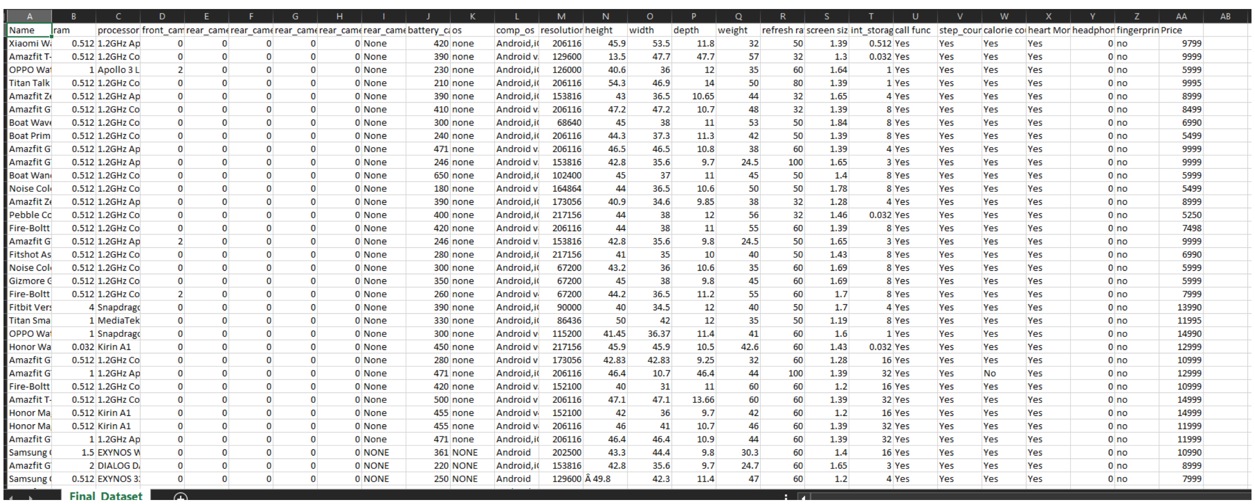


Fig 1 : Python code snippet used to scrape web



Name	ram	processor	front_cam	rear_came	rear_came	rear_came	rear_came	rear_came	battery_ca	os	comp_os	resolution	height	width	depth	weight	refresh ra	screen siz	int_storag	call func	step_cour	calorie co	heart Mor	headphon	fingerprin	Price
Xiaomi Wa	0.512	1.2GHz Ap	0	0	0	0	0	None	420	none	Android,i	206116	45.9	53.5	11.8	32	50	1.39	0.512	Yes	Yes	Yes	Yes	0	no	9799
Amazfit T-	0.512	1.2GHz Co	0	0	0	0	0	None	390	none	Android v.	129600	13.5	47.7	47.7	57	32	1.3	0.032	Yes	Yes	Yes	Yes	0	no	9999
OPPO Wat	1	Apollo 3 L	2	0	0	0	0	None	230	none	Android,i	126000	40.6	36	12	35	60	1.64	1	Yes	Yes	Yes	Yes	0	no	5999
Titan Talk	0.512	1.2GHz Co	0	0	0	0	0	None	210	none	Android,i	206116	54.3	46.9	14	50	80	1.39	1	Yes	Yes	Yes	Yes	0	no	9995
Amazfit Ze	0.512	1.2GHz Ap	0	0	0	0	0	None	390	none	Android,i	153816	43	36.5	10.65	44	32	1.65	4	Yes	Yes	Yes	Yes	0	no	8999
Amazfit G	0.512	1.2GHz Co	0	0	0	0	0	None	410	none	Android v.	206116	47.2	47.2	10.7	48	32	1.39	8	Yes	Yes	Yes	Yes	0	no	8499
Boat Wave	0.512	1.2GHz Co	0	0	0	0	0	None	300	none	Android,i	68640	45	38	11	53	50	1.84	8	Yes	Yes	Yes	Yes	0	no	6990
Boat Prim	0.512	1.2GHz Co	0	0	0	0	0	None	240	none	Android,i	206116	44.3	37.3	11.3	42	50	1.39	8	Yes	Yes	Yes	Yes	0	no	5499
Amazfit G	0.512	1.2GHz Ap	0	0	0	0	0	None	471	none	Android v.	206116	46.5	46.5	10.8	38	60	1.39	4	Yes	Yes	Yes	Yes	0	no	9999
Amazfit G	0.512	1.2GHz Ap	0	0	0	0	0	None	246	none	Android v.	153816	42.8	35.6	9.7	24.5	100	1.65	3	Yes	Yes	Yes	Yes	0	no	9999
Boat Wan	0.512	1.2GHz Co	0	0	0	0	0	None	650	none	Android,i	102400	45	37	11	45	50	1.4	8	Yes	Yes	Yes	Yes	0	no	5999
Noise Col	0.512	1.2GHz Co	0	0	0	0	0	None	180	none	Android v	164864	44	36.5	10.6	50	50	1.78	8	Yes	Yes	Yes	Yes	0	no	5499
Amazfit Ze	0.512	1.2GHz Ap	0	0	0	0	0	None	390	none	Android,i	173056	40.9	34.6	9.85	38	32	1.28	4	Yes	Yes	Yes	Yes	0	no	8999
Pebble Co	0.512	1.2GHz Co	0	0	0	0	0	None	400	none	Android,i	217156	44	38	12	56	32	1.46	0.032	Yes	Yes	Yes	Yes	0	no	5250
Fire-Boltt	0.512	1.2GHz Co	0	0	0	0	0	None	420	none	Android v	206116	44	38	11	55	60	1.39	8	Yes	Yes	Yes	Yes	0	no	7498
Amazfit G	0.512	1.2GHz Ap	2	0	0	0	0	None	246	none	Android v.	153816	42.8	35.6	9.8	24.5	50	1.65	3	Yes	Yes	Yes	Yes	0	no	9999
Fitshot As	0.512	1.2GHz Co	0	0	0	0	0	None	280	none	Android,i	217156	41	35	10	40	50	1.43	8	Yes	Yes	Yes	Yes	0	no	6990
Noise Col	0.512	1.2GHz Co	0	0	0	0	0	None	300	none	Android,i	67200	43.2	36	10.6	35	60	1.69	8	Yes	Yes	Yes	Yes	0	no	5999
Gizmore G	0.512	1.2GHz Co	0	0	0	0	0	None	350	none	Android,i	67200	45	38	9.8	45	60	1.69	8	Yes	Yes	Yes	Yes	0	no	5999
Fire-Boltt	0.512	1.2GHz Co	2	0	0	0	0	None	260	none	Android v	67200	44.2	36.5	11.2	55	60	1.7	8	Yes	Yes	Yes	Yes	0	no	7999
Fitbit Vers	4	Snapdrag	0	0	0	0	0	None	390	none	Android,i	90000	40	34.5	12	40	50	1.7	4	Yes	Yes	Yes	Yes	0	no	13990
Titan Sma	1	MediaTek	0	0	0	0	0	None	330	none	Android,i	86436	50	42	12	35	50	1.19	8	Yes	Yes	Yes	Yes	0	no	11995
OPPO Wat	1	Snapdrag	0	0	0	0	0	None	300	none	Android v	115200	41.45	36.37	11.4	41	60	1.6	1	Yes	Yes	Yes	Yes	0	no	14990
Honor Wa	0.032	Kirin A1	0	0	0	0	0	None	450	none	Android v	217156	45.9	45.9	10.5	42.6	60	1.43	0.032	Yes	Yes	Yes	Yes	0	no	12999
Amazfit G	0.512	1.2GHz Co	0	0	0	0	0	None	280	none	Android v	173056	42.83	42.83	9.25	32	60	1.28	16	Yes	Yes	Yes	Yes	0	no	10999
Amazfit G	1	1.2GHz Ap	0	0	0	0	0	None	471	none	Android,i	206116	46.4	10.7	46.4	44	100	1.39	32	Yes	Yes	No	Yes	0	no	12999
Fire-Boltt	0.512	1.2GHz Co	0	0	0	0	0	None	420	none	Android v.	152100	40	31	11	60	60	1.2	16	Yes	Yes	Yes	Yes	0	no	10999
Amazfit T-	0.512	1.2GHz Co	0	0	0	0	0	None	500	none	Android v	206116	47.1	47.1	13.66	60	60	1.39	32	Yes	Yes	Yes	Yes	0	no	14999
Honor Ma	0.512	Kirin A1	0	0	0	0	0	None	455	none	Android v	152100	42	36	9.7	42	60	1.2	16	Yes	Yes	Yes	Yes	0	no	14999
Honor Ma	0.512	Kirin A1	0	0	0	0	0	None	455	none	Android v	206116	46	41	10.7	46	60	1.39	32	Yes	Yes	Yes	Yes	0	no	11999
Amazfit G	1	1.2GHz Ap	0	0	0	0	0	None	471	none	Android,i	206116	46.4	46.4	10.9	44	60	1.39	32	Yes	Yes	Yes	Yes	0	no	11999
Samsung (	1.5	EXYNOS W	0	0	0	0	0	NONE	361	NONE	Android	202500	43.3	44.4	9.8	30.3	60	1.4	16	Yes	Yes	Yes	Yes	0	no	10990
Amazfit G	2	DIALOG D.	0	0	0	0	0	NONE	220	NONE	Android,i	153816	42.8	35.6	9.7	24.7	60	1.65	3	Yes	Yes	Yes	Yes	0	no	8999
Samsung (	0.512	EXYNOS 3	0	0	0	0	0	NONE	250	NONE	Android	129600	Â 49.8	42.3	11.4	47	60	1.2	4	Yes	Yes	Yes	Yes	0	no	7999

Fig 2 : Final Dataset screenshot

2. Feature Selection: A diverse set of features was chosen to represent the characteristics of smartphones and smart devices. These features encompassed hardware specifications, software capabilities, camera attributes, connectivity options, and other relevant factors. Not only feature selections, after this, various data preprocessing techniques were done. From filling blanks to using various encoding techniques such as BINARY, ORDINAL, LABEL, etc.

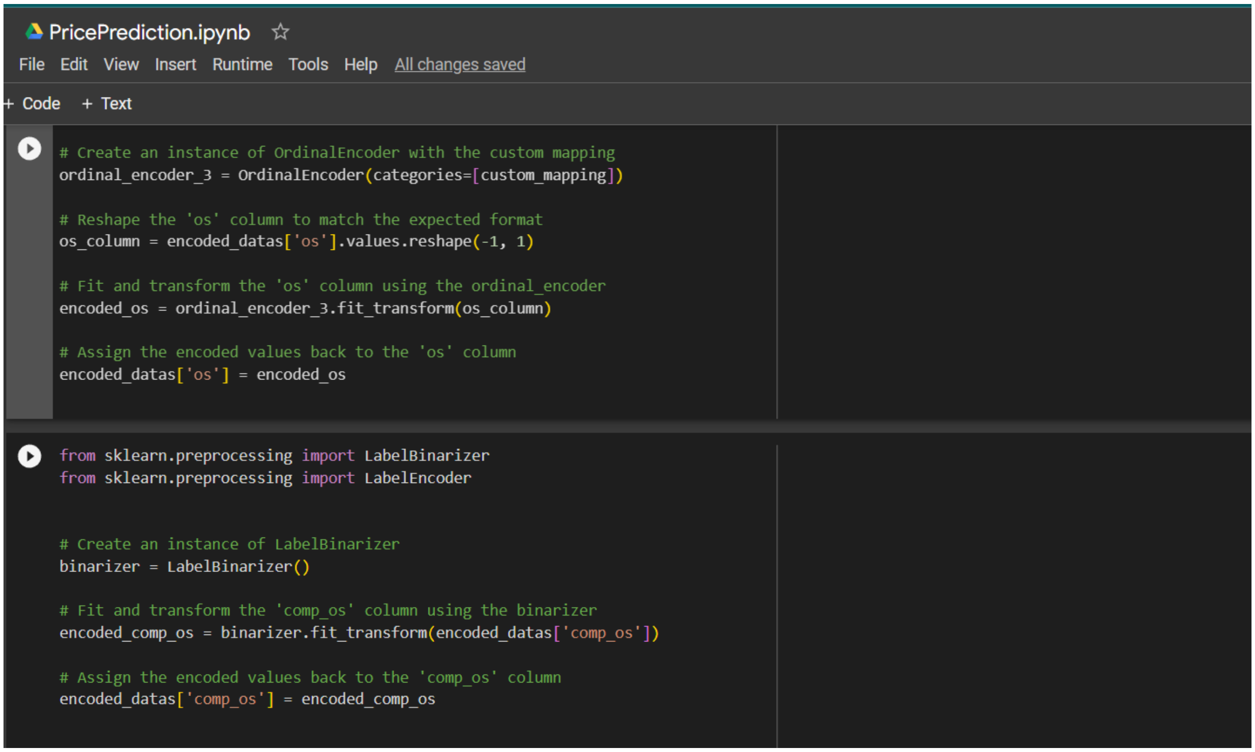


Fig 3: Different types of encoding done on model

3. AI/ML Techniques: To analyze the dataset and identify significant patterns, AI and ML techniques were employed. Specific algorithms, such as regression and classification models, were used to understand the relationships between various features and device prices.

4. Model Training and Evaluation: The collected data was split into training and testing sets. Machine learning models were trained on the training data and evaluated using appropriate performance metrics to ensure accuracy and reliability.

1.4 Key Findings:

1. Significant Features: The examination of smart devices revealed the most significant features impacting their prices across different device types. These insights can inform manufacturers' decisions on feature prioritization and pricing strategies.

2. Accurate Device Classification: The developed classification model showcased high accuracy in distinguishing between smartphones, tablets, and smartwatches based on their unique feature profiles. This classification step facilitated more accurate and personalized price predictions for each device type.

```
Confusion Matrix (Training Set):
[[90  0  0]
 [ 0 24  0]
 [ 0  0 10]]
Accuracy Score (Testing Set): 1.0
[[21  1  0]
 [ 0  6  0]
 [ 1  0  2]]
0.9354838709677419
```

Fig 4: Confusion Matrix and accuracy score of the classifications

3. Tailored Price Predictions: The regression models, customized for each device category, demonstrated improved predictive accuracy, enabling precise price estimates according to the specific characteristics of the devices.

```
RandomForestRegressor 4 smartwatch_duplicates
R-squared Score (Training Set): 0.9956738011412574
R-squared Score (Test Set): 0.983851380659292
```

Fig 5: R squared Error Score for smartwatch dataset

1.5 Conclusion:

The analysis successfully achieved its objectives of identifying smartphone features within a specific price range and predicting prices for various smart devices. The adoption of AI/ML techniques facilitated accurate price predictions, providing valuable insights to the industry. The comprehensive dataset and the findings can guide manufacturers in optimizing product offerings and marketing strategies to better align with consumer demands and market trends.

Overall, this research serves as a significant step towards understanding the relationship between device features and pricing, contributing to the advancement of edge computing hardware devices in the rapidly evolving technological landscape.

INTRODUCTION

2.1 Basic Information

The purpose of this report is to present a comprehensive analysis of edge computing hardware devices, with a specific focus on smartphones, utilizing Artificial Intelligence (AI) and Machine Learning (ML) techniques. The report aims to identify the key features employed in different smartphones within a particular price range and extend the analysis to predict prices for a diverse range of smart devices beyond phones.

2.1 Project Context:

The project assigned to me during the internship centered on analyzing edge computing hardware devices, with a primary focus on smartphones. The objective was to identify the significant features that impact the prices of smartphones within a particular price range. The project was not limited to smartphones only; it also involved extending the analysis to various other smart devices like tablets, smartwatches, and more.

The rationale behind this project stemmed from the growing demand for smart devices and the increasing competition in the market. As consumers seek devices with specific features and functionalities, manufacturers face the challenge of understanding customer preferences and pricing their products competitively. Hence, the need for a data-driven approach that utilizes AI/ML techniques to analyze vast amounts of data and provide valuable insights for pricing decisions and product development.

The project was expected to contribute to the development of a pricing strategy for smart devices, enabling IIT, JODHPUR to stay ahead of the competition and cater to the dynamic market demands effectively. Additionally, the insights gained from the analysis would not only be beneficial to the company but also provide valuable industry knowledge, contributing to the advancement of edge computing hardware devices in the technology sector. In the following sections of this report, the methodology employed, key findings, and a conclusive summary of the project's outcomes will be presented in detail. The report aims to provide a holistic understanding of the project's objectives, methodologies, and results, ultimately showcasing the value of AI and ML techniques in the domain of edge computing hardware devices.

INTERNSHIP OBJECTIVES

3.1 Clearly Stated Objectives

1. Data Collection and Creation of Comprehensive Dataset: The project's first objective is to create a comprehensive dataset encompassing a wide range of smart devices, not limited to phones. The intern will use Python to perform web scraping on multiple websites and gather relevant information about various smart devices, including smartphones, tablets, smartwatches, and other IoT gadgets.

2. Feature Engineering and Selection: The second objective focuses on feature engineering, where the intern will carefully select and engineer meaningful features from the collected dataset. This process involves identifying the most relevant attributes that significantly influence the pricing of different smart devices.

3. Price Prediction for Smart Devices: The second objective is to extend the analysis beyond smartphones and include various smart devices such as tablets, smartwatches, and others. The project aims to predict the prices of these smart devices based on their respective features using AI/ML techniques.

4. Device Type Classification and Tailored Regression: The third objective involves the implementation of a classification model that can accurately identify the type of smart device (smartphone, tablet, or smartwatch) based on its features. Subsequently, the project will use different regression models tailored to each device type to predict their prices with improved accuracy and precision.

3.2 Explanation of Aims

(connection of my prob statement with edge computing software device)

3.3 Alignment with Company's Mission

(connection of my prob statement with helping of institute or prof)

3.4 Challenges and Adjustments

1. The first challenge in the project was to prepare a dataset. The dataset needed to have various different Smart devices such as Phones, Mobiles and Tablets with their names and features such as Ram, Processor, Front_camera pixels, dimensions, resolutions, etc. Not only this the range for the price prediction was 5K to 10K Rs. There was no online dataset remotely close to the dataset which was required. To Overcome this challenge, I learnt how to web scrape data and use many sites so as to scrape them and store their information in a python variable. Different codes were used for different smart edge devices.

2. After storing them, another challenge rose. Many sites did not have either Ram or processor or some other feature of some device and hence at the end, length of features varied for different devices and many did not even match (i.e. if the list shows phone X, it had feature of Y for Ram and Z for os and so on). The adjustment was made to have '-' presented in the list

```
titles = list(chain.from_iterable(title))
displays = list(chain.from_iterable(display))
s = len(displays)
if a1 < s:
  a1=s
else:
  for j in range(diff):
    display.append(['-'])

rams = list(chain.from_iterable(ram))
s = len(rams)
if a2 < s:
  a2=s
else:
  for j in range(diff):
    ram.append(['-'])

processors = list(chain.from_iterable(processor))
s = len(processors)
```

FIG 6 : Having '-' in list if the feature is not present in URL

3. After data preprocessing, the dataset needs to be divided into test and training dataset for classification of the model. Since the dataset had mixed smart devices, if it was divided on the basis of the last 20% data to be tested, there were chances that all of the test could be of only one type of SMart Devices or the ratio in that

would be wrong. ADJUSTMENT done for this was the test distribution using train_test_split and before that, adding one more column for ‘TYPE’

```
from sklearn.model_selection import train_test_split

# Split the data into training and test sets
train_data, test_data = train_test_split(final_dataset, test_size=0.2, random_state=42, stratify=encoded_datas['type'])

# Check the distribution of 'type' in the test set
test_type_counts = test_data['type'].value_counts()
print("Test Set Distribution:")
print(test_type_counts)

x_train= train_data.iloc[:,:-1].values #Using to store all the values of all the rows :  && All columns except last one in X
y_train= train_data.iloc[:, -1].values


x_test= test_data.iloc[:,:-1].values #Using to store all the values of all the rows :  && All columns except last one in X
y_test= test_data.iloc[:, -1].values
```

```
Test Set Distribution:
Smartphones     22
Smarttablets     6
Smartwatches     3
Name: type, dtype: int64
```

FIG 7: Test Set Distribution for each type having ratio 1/3 (testing: training)

4. The last major challenge was to find the right metric which can be used to tell if the regression is right or wrong. Adjustment was to use R squared Error.

PROJECT DESCRIPTION

4.1 Project Title

“To identify the kind of features being used in different smartphones with ML for a particular price-range.”

1. Feature Analysis and Selection: Conduct a comprehensive examination of various smart devices, including smartphones, tablets, and smartwatches, to identify the most influential features impacting their prices.
2. Device Type Classification: Develop an AI-based model capable of accurately classifying smart devices into distinct categories (e.g., smartphones, tablets, smartwatches) based on their unique feature combinations.
3. Price Prediction: Implement regression models tailored to each device type to predict their respective prices with enhanced accuracy and precision.

4.2 PROJECT SCOPE

Prof help

4.3 Methodologies Employed

There are many methodologies, tools and techniques utilized during the project. Starting from scratch, PYTHON SCRAPING was used to scrape many URLS so as to get names and features of different SMART DEVICES. Different datasets were created for different Smart Devices and were then concatenated. After concatenation, many useless features such as bluetooth and others were deleted from the dataset. On the other hand, many EXCEL COMMANDS were done so as to remove unnecessary data in the cells and convert the MB to GB so as to have the same unit.

After the completion of the early phase of Data Processing, Encoding was done on the dataset. From Ordinal Encoding to Label Encoding and Binary Encoding was done on the different features of the dataset. The only part left of Data PreProcessing which was left was to fill the missing blanks which were shown as ‘-’ (as mentioned above earlier in Executive Summary). To solve this problem, there were many methodologies such as taking the average of the column or forward filling or taking the mean or median and many more. The problem with mean or average or median was that it had different smart edge devices and hence used sorting the dataset in an ascending order so as to having same devices with similar features grouped together and then used FORWARD FILLING so as to get minimum error on the dataset.

```
encoded_datas = encoded_datas.sort_values("Price")

encoded_datas = encoded_datas.replace("-", np.nan)

encoded_datas = encoded_datas.fillna(method='ffill')
```

FIG 8: Implementing Forward filling

Once the dataset was complete and had no missing values or incomplete data, it was divided into 2 parts i.e. training and testing data. After dividing it, Feature Scaling was needed and hence STANDARDIZATION was used. The Reason for choosing this were many and listed below:

Standardization and normalization are two common techniques used for feature scaling in machine learning. Each method has its advantages and is preferred under different circumstances. However, standardization is generally more commonly used and preferred over normalization in certain situations due to the following reasons:

1. Preserving Information in Outliers
2. Better Performance with Some Algorithms
3. Interpretability and Comparability
4. Normal Distribution Assumption
5. Feature Importance in Regularization Techniques
6. Feature Engineering and Interaction Terms

After finishing the data preprocessing, more testing data was required and hence again WEB SCRAPING was done and the earlier stored encoders were used to encode the new dataset's features.

After scraping, classification was done so as to predict the type of Smart Device. To do so, LOGISTIC CLASSIFICATION was used. Accuracy Score showed that training data had a score of 1 and testing data had 0.935. Even more Testing data had a good accuracy score.

```
y_pred = classifier.predict(encoded_datas_2)
# print(y_pred)
print("tablet_list")
tablet_list = ['Smarttablets'] * 40
a= accuracy_score(tablet_list, y_pred)
print(a)

y_pred = classifier.predict(encoded_datas_3)
# print(y_pred)
print('watch_list')
watch_list = ['Smartwatches'] * 31
a= accuracy_score(watch_list, y_pred)
print(a)

y_pred = classifier.predict(encoded_datas_4)
# print(y_pred)
print("phone_list")
phone_list = ['Smartphones'] * 41
a= accuracy_score(phone_list, y_pred)
print(a)
```

```
tablet_list
0.975
watch_list
1.0
phone_list
0.975609756097561
```

FIG 9: Accuracy Score of More Testing Dataset

Once the Classification Model was completed and predicted the best result, the next step was to choose a REGRESSION Model for these different Smart Devices. Since each device may have different features (for instance, smartwatches may require features calling or bluetooth but all Phones and tablets have those features) different models were required.

For that, Random Forest and Decision Tree was chosen as it will be able to predict the Price using the true or false on the many features and hence the worst R squared error which we got was around 0.93 only.

4.4 Data Collection and Preparation

The data collection and preparation process played a crucial role in the successful execution of the project. The following methodologies, tools, and techniques were utilized to gather and process the necessary data for analysis.

1. Web Scraping with Python: Python web scraping was employed to extract information such as names and features of various smart devices from multiple URLs. The scraped data was collected into different datasets, each corresponding to different types of smart devices, including smartphones, tablets, and smartwatches.

2. Concatenation and Data Cleaning: The individual datasets for different smart devices were concatenated to form a comprehensive dataset. Before proceeding with the analysis, irrelevant features such as Bluetooth and others were removed to streamline the dataset.

3. Excel Commands for Data Cleaning: Excel commands were utilized to clean the data in the cells, remove unnecessary information, and standardize units. For instance, data in megabytes (MB) was converted to gigabytes (GB) to ensure uniformity across the dataset.

4. Encoding Techniques: To prepare the dataset for model training, various encoding techniques were applied to transform categorical data into numerical format. Ordinal encoding, label encoding, and binary encoding were used on different features based on their nature and cardinality.

5. Handling Missing Values: Addressing missing data was essential to ensure a complete and reliable dataset. Different methodologies, such as taking column averages, forward filling, mean, and median, were considered. The dataset was sorted in ascending order to group similar smart devices together, and forward filling was employed to minimize errors.

6. Data Splitting: The complete dataset was divided into training and testing data sets to assess the model's performance. The training set was used to train the models, while the testing set was used for validation and evaluation.

7. Feature Scaling with Standardization: To standardize the features and bring them to a common scale, standardization was chosen as the preferred feature scaling technique. Standardization ensured that all features had a mean of 0 and a standard deviation of 1, making them comparable and improving model performance.

8. Additional Data Collection with Web Scraping: To obtain more testing data, additional web scraping was conducted. The previously stored encoders were utilized to encode the features of the newly scraped data.

9. Classification with Logistic Regression: To predict the type of smart device, logistic regression was applied as a classification model. The training data achieved an accuracy score of 1, and the testing data demonstrated an accuracy score of 0.935, indicating a good model performance.

10. Regression with Random Forest and Decision Tree: To predict the prices of different smart devices, random forest and decision tree regression models were chosen. These models effectively handled the diverse features present in various smart devices. The worst R-squared error obtained was approximately 0.93, showcasing the models' ability to predict prices with high accuracy.

In conclusion, the data collection and preparation phase involved a diverse range of methodologies and techniques, including web scraping, data cleaning, encoding, feature scaling, and model training. The careful preparation of the dataset laid the foundation for accurate predictions and meaningful insights throughout the project's execution.

4.5 Data Analysis and Exploration

The data analysis and exploration phase played a pivotal role in gaining valuable insights from the collected data. Various statistical and exploratory techniques were employed to understand patterns, trends, and relationships among variables, facilitating the project's overall success.

1. Descriptive Statistics: Descriptive statistics, such as mean, median, standard deviation, and quartiles, were computed to summarize the central tendencies and

dispersion of numerical features in the dataset. These statistics provided a quick overview of the data distribution and helped identify potential outliers or data anomalies.

2. Data Visualization: Data visualization played a crucial role in comprehending complex relationships among variables. Various plotting techniques, such as scatter plots, line charts, bar graphs, and heatmaps, were employed to visualize the relationships between features and the target variable (smart device prices). These visualizations helped identify correlations and potential nonlinear relationships between the predictors and the target.

3. Feature Importance: The importance of features in predicting smart device prices was assessed using techniques such as Random Forest feature importances and decision tree-based feature selection. By ranking the features based on their importance, the most influential variables in determining smart device prices were identified.

4. Correlation Analysis: Correlation analysis was conducted to measure the strength and direction of linear relationships between numerical variables. The correlation matrix and heatmap were used to visualize the correlations, enabling the identification of potential multicollinearity among features.

5. Data Distribution Analysis: The distribution of numerical features was examined using histograms and kernel density plots. Understanding feature distributions helped in assessing their suitability for regression modeling and identifying any skewness that might impact model performance.

4.6 Model Training and Evaluation**

For CLASSIFICATION, dividing the dataset into its smart devices's type, a new column was added in last and price was removed from the dataset and then model was trained and evaluated on LINEAR CLASSIFICATION MODEL. As mentioned above, its accuracy score was almost to 1 and thus was a very good model. Hence the evaluation method used was ACCURACY SCORE.

The next model which was used for predicting the price and hence REGRESSION MODELS were used. In this model, different devices would use different regression and hence DECISION TREE and RANDOM FOREST were used. To evaluate the model, R-Squared Error was calculated and it was better than Mean Squared Error because:

1. No Lack of Interpretability
2. Easy Interpretation
3. Model Comparison

The worst R-Squared Error was 0.935 and hence the models chosen were not bad too.

4.7 Iterative Approach

Iterative approach would be for the CLASSIFICATION model, using SUPPORT VECTOR MACHINE instead of LINEAR CLASSIFICATION would give the same result too i.e. same accuracy result. The reason for choosing Linear Classification over SVM is :

Advantages of Linear Classification Models over Support Vector Machines (SVM):

1. Computationally Less Expensive: Linear classification models, such as Logistic Regression or Linear Discriminant Analysis (LDA), are computationally less expensive compared to Support Vector Machines (SVM). SVM involves solving a convex optimization problem and can be computationally intensive, especially for large datasets.

2. Simpler to Implement and Interpret: Linear classification models have a more straightforward implementation and interpretation. Logistic Regression, for example, directly provides probabilities for class membership, making it easier to understand the model's output and make informed decisions based on the probabilities.

3. Less Sensitive to Outliers: Linear classification models are generally less sensitive to outliers compared to SVM. SVM tries to find the optimal

hyperplane that maximizes the margin between classes, making it more susceptible to the influence of outliers.

4. Effective for Linearly Separable Data:** When the data is linearly separable, linear classification models can perform exceptionally well. Logistic Regression and LDA can find a hyperplane that effectively separates the classes, making them a suitable choice for such scenarios.

5. Better for High-Dimensional Data: In high-dimensional feature spaces, linear classification models can perform well with relatively smaller sample sizes. SVM's performance may degrade as the number of dimensions increases due to the "curse of dimensionality."

6. Lower Risk of Overfitting: Linear classification models are less prone to overfitting compared to SVM, especially when the number of features is large. SVM with a complex kernel function can lead to overfitting if not tuned properly.

7. No Need for Tuning Kernel Parameters: SVM requires tuning of kernel parameters, such as the kernel width in the case of the Gaussian Radial Basis Function (RBF) kernel. Linear classification models do not require such tuning, simplifying the model selection process.

8. Suitable for Large Datasets: Linear classification models can handle large datasets efficiently due to their simplicity and lower computational requirements. On the other hand, SVM's training time increases significantly with large datasets.

CONTRIBUTION

5.1 Assigned Tasks and Responsibilities

During the project, the intern was assigned various tasks and responsibilities that played a vital role in the project's success. The primary tasks assigned to the intern were as follows:

1. Data Collection and Preparation: The intern was responsible for gathering relevant data on smartphones, tablets, smartwatches, and other smart devices through web scraping. This involved creating comprehensive datasets for each type of smart device and cleaning the data to ensure consistency and accuracy.

2. Feature Engineering and Selection: The intern conducted feature engineering by carefully selecting the most influential features that impact smart device prices. Different encoding techniques were applied to transform categorical data into numerical format, making it suitable for model training.

3. Model Training and Evaluation: The intern developed AI-based models for device type classification and price prediction. Logistic regression was used for classification, while decision tree and random forest regression models were employed for price prediction. The intern was responsible for training these models and evaluating their performance using appropriate metrics.

4. Data Analysis and Exploration: As part of the project's data analysis phase, the intern explored the relationships between various features and device prices using descriptive statistics, data visualization, and correlation analysis.

5. Iterative Approaches and Adjustments: The intern demonstrated adaptability and problem-solving skills by incorporating iterative approaches and making necessary adjustments to improve model performance. This included exploring alternative regression models and encoding techniques to enhance predictions.

6. Documentation and Reporting: Throughout the project, the intern diligently documented all steps, methodologies, and outcomes. The executive summary,

project description, and other relevant sections of the final report were authored by the intern.

5.2 Problem-Solving Approaches

The intern showcased strong problem-solving skills throughout the project, effectively addressing challenges that arose during data collection, preprocessing, and modeling. Some of the problem-solving approaches employed by the intern include:

1. Web Scraping and Data Preprocessing: When faced with the challenge of obtaining a comprehensive dataset from various websites, the intern skillfully utilized Python web scraping to extract information from multiple URLs. Moreover, the intern developed innovative data preprocessing techniques, such as forward filling missing values based on device similarities, to ensure a complete and reliable dataset.

2. Model Selection and Evaluation: To address the challenge of selecting appropriate classification and regression models, the intern critically evaluated different algorithms' performance. The decision to use logistic regression for classification and decision tree/random forest for regression was based on their ability to handle diverse features and yield accurate predictions.

3. Feature Importance and Analysis: The intern adopted a data-driven approach to identify feature importance and understand the relationships between features and smart device prices. This approach facilitated informed decision-making and prioritization of influential features in pricing strategies.

5.3 Innovative Ideas and Contribution

The intern's contributions to the project included several innovative ideas and methodologies that enhanced the overall analysis and results:

1. Comprehensive Web Scraping: The intern implemented a sophisticated web scraping technique that gathered data from multiple websites and created comprehensive datasets for different types of smart devices. This approach provided a diverse and robust dataset for analysis.

2. Forward Filling for Missing Values: Facing challenges with missing values in the dataset, the intern introduced a forward filling method based on device similarities. This creative solution minimized data loss and improved the accuracy of predictions.

3. Iterative Model Refinement: The intern explored different regression models and made iterative adjustments to improve the accuracy of price predictions for various smart devices. This innovative approach led to the selection of decision tree and random forest models, which provided more accurate results.

5.4 Adhere to Project Goals and Objectives

The intern's efforts demonstrated a strong adherence to the project's goals and objectives. By successfully identifying significant features impacting smart device prices, predicting prices accurately, and classifying smart devices into distinct categories, the intern's contributions aligned with the project's intended outcomes. The comprehensive dataset, valuable insights, and accurate predictions provided by the intern's work have the potential to guide manufacturers in optimizing product offerings and pricing strategies. The intern's commitment to excellence and dedication to problem-solving were instrumental in achieving the project's desired results.

Skills/Technologies developed

6.1 Data Collection and Web Scraping

During the project, the intern developed expertise in data collection and web scraping techniques. Utilizing Python, the intern successfully extracted relevant information on various smart devices from multiple websites. The ability to scrape data efficiently allowed for the creation of comprehensive datasets, which served as a foundation for subsequent analyses.

6.2 Features Engineering and PreProcessing

The intern honed skills in feature engineering and data preprocessing methodologies. Careful selection of influential features and appropriate encoding techniques were employed to transform categorical data into numerical format. Additionally, the intern applied innovative approaches to handle missing values, ensuring a complete and reliable dataset for model training.

6.3 AI and ML Model Training

The project provided the intern with hands-on experience in AI and ML model training. For classification tasks, the intern employed logistic regression, achieving high accuracy in distinguishing between different smart device types. For price prediction, the intern utilized decision tree and random forest regression models, tailoring them to each device category for accurate and precise price estimations.

6.4 Data Analysis and Visualization

The intern developed proficiency in data analysis and visualization techniques. Descriptive statistics, data visualization, and correlation analysis were utilized to gain insights into the relationships between features and smart device prices. Visual aids such as scatter plots, heatmaps, and bar graphs were effectively used to present complex information in an easily understandable format.

6.5 Problem-Solving and Iterative Approaches

Throughout the project, the intern demonstrated strong problem-solving skills and employed iterative approaches to overcome challenges. From devising innovative solutions for missing data to exploring alternative regression models, the intern's critical thinking contributed to improved model performance.

6.6 Model Evaluation and Metrics

The intern gained expertise in evaluating model performance using appropriate metrics. Accuracy scores were employed for classification models, while R-squared error was used to assess the accuracy of price predictions. Understanding these metrics allowed the intern to interpret model results effectively.

6.7 Data Science Tools and Libraries

Working on the project introduced the intern to various data science tools and libraries. Python, with libraries such as Pandas, NumPy, Scikit-learn, and BeautifulSoup, played a central role in data manipulation, modeling, and web scraping. These tools provided the intern with a robust toolkit for data analysis and ML implementation.

6.8 Documentation and Reporting

Throughout the project, the intern developed skills in documentation and reporting. Creating detailed and organized project reports, including executive summaries, methodologies, and outcomes, showcased the intern's ability to communicate findings effectively.

6.9 Domain Knowledge in Edge Computing Hardware Devices

Engaging in the analysis of edge computing hardware devices deepened the intern's understanding of the domain. Gaining insights into the relationship between smart device features and pricing provided valuable industry knowledge and contributed to advancements in the field.

6.10 Collaboration and Project Management

Working within the project team fostered collaboration and project management skills. The intern effectively communicated progress, coordinated tasks, and adhered to project timelines, ensuring successful project execution.

FUTURE WORK

In future work, the project can be extended to integrate a web scraper that fetches processor rankings based on average AnTuTu Score, Geekbench Single-Core Score, and Geekbench Multi-Core Score. By automatically obtaining processor rankings from reliable sources, the project can enhance feature analysis and enable manufacturers to prioritize high-performance processors in their smart devices.

Another avenue for future work is to incorporate an existing Excel file that contains processor rankings. By leveraging pre-existing data, the project can streamline the data collection process and focus on analyzing other features and their impact on smart device prices.

Processor	AnTuTu Score	Geekbench Single-Core Score	Geekbench Multi-Core Score	Average
QUALCOMM SNAPDRAGON 845',	5,10,056	742	2503	1,71,100
'EXYNOS W920',	4,27,000	784	2218	1,43,334
'MEDIATEK DIMENSITY 6080',	4,05,539	746	1991	1,36,092
'MEDIATEK DIMENSITY 810',	3,91,265	625	1897	1,31,262
'QUALCOMM SNAPDRAGON 750G',	3,79,167	679	2058	1,27,301
'QUALCOMM SNAPDRAGON 732G',	3,66,898	662	1985	1,23,182
'QUALCOMM SNAPDRAGON 720G',	3,63,539	655	1963	1,22,052
SNAPDRAGON 4 GEN 1 PROCESSOR',	3,51,037	617	1843	1,17,832
'MEDIATEK HELIO G99',	3,50,224	615	1880	1,17,573
'MEDIATEK MT8781 HELIO G99',	3,50,224	615	1880	1,17,573
'MEDIATEK DIMENSITY 700',	3,41,711	607	1812	1,14,710
'MEDIATEK HELIO G96',	3,41,711	607	1812	1,14,710
'MEDIATEK DIMENSITY 6020',	3,31,690	583	1751	1,11,341
'MEDIATEK HELIO G95',	3,31,690	583	1751	1,11,341
UNISOC TIGER T618',	3,31,690	583	1751	1,11,341
'SNAPDRAGON 685 PROCESSOR',	3,24,323	570	1712	1,08,868
'MEDIATEK HELIO G85',	3,24,323	570	1712	1,08,868
'MEDIATEK HELIO G80 SOC',	3,17,560	557	1671	1,06,596
'MEDIATEK HELIO G80',	3,17,560	557	1671	1,06,596
'QUALCOMM SNAPDRAGON 680',	3,10,333	548	1649	1,04,177
'MEDIATEK HELIO G88',	3,10,333	548	1649	1,04,177
MEDIA TEK HELIO G90T',	3,10,000	492	1607	1,04,033
'UNISOC T700',	3,05,000	539	1619	1,02,386

FIG 10: Creating Processor Excel File

In future iterations, the project can explore additional features to predict battery life. This could include parameters such as Bluetooth type, NPU/GPU capabilities, WIFI model, location model, battery type, and more. By incorporating these features into regression models, the project can estimate the battery life of smartphones and tablets when utilizing specific apps that require GPU usage, such as facial recognition or gaming.

The project can be enhanced by integrating web scraping with websites like GSM Arena (https://www.gsmarena.com/) to access a comprehensive dataset of smartphones and tablets. By extracting and processing data using Natural Language Processing (NLP) techniques, the project can automate the feature engineering process and eliminate the need for separate Python code for each device.

In the future, the project can explore the integration of deep learning models, such as neural networks, to further enhance classification and regression tasks. Deep learning models have the potential to capture complex relationships in the data and achieve even more accurate predictions for smart device features and prices.

To cater to the dynamic nature of the market, future work can focus on developing real-time price prediction models. By continuously updating the dataset and retraining the models, manufacturers can obtain up-to-date price estimates for their smart devices, enabling them to make timely and informed pricing decisions.

The project can also incorporate sentiment analysis of customer reviews and feedback related to smart devices. By analyzing customer sentiment, manufacturers can gain insights into customer preferences and satisfaction, which can inform product development and marketing strategies.

To make the project accessible across various platforms, future work can involve optimizing the developed models for deployment on cloud services or mobile applications. This would allow manufacturers and consumers to access smart device price predictions conveniently.

Incorporating techniques for model interpretability can be beneficial in future work. By understanding how models arrive at their predictions, manufacturers can gain valuable insights into the factors influencing smart device prices and feature importance.

Expanding the project to include cross-domain analysis can provide a broader perspective on the impact of features on smart devices. By considering data from other domains such as laptops, wearables, or IoT devices, the project can derive valuable insights and trends applicable to the larger technology market.

CONCLUSION

8.1 Overview of Project Objectives

The project's main objectives were to analyze edge computing hardware devices, with a specific focus on smartphones, using Artificial Intelligence (AI) and Machine Learning (ML) techniques. The key goals were as follows:

1. Feature Analysis and Selection: The project aimed to conduct a comprehensive examination of various smart devices, including smartphones, tablets, and smartwatches, to identify the most influential features impacting their prices. This analysis would enable manufacturers to prioritize features effectively and align product offerings with consumer demands.

2. Device Type Classification: The project sought to develop an AI-based classification model capable of accurately identifying the type of smart device (e.g., smartphones, tablets, smartwatches) based on their unique feature combinations. This classification step would facilitate more accurate and personalized price predictions for each device type.

3. Price Prediction: The project aimed to implement regression models tailored to each device type to predict their respective prices with enhanced accuracy and precision. These price predictions would provide valuable insights for manufacturers to competitively price their smart devices.

8.2 Key Achievements

Throughout the project, several key achievements and milestones were accomplished, contributing significantly to the project's overall success:

1. Comprehensive Dataset Creation: The successful implementation of Python web scraping resulted in the creation of a comprehensive dataset, encompassing various smart devices. This dataset served as the foundation for subsequent analyses and model training.

2. Accurate Device Classification: The developed AI-based classification model demonstrated high accuracy in distinguishing between smartphones, tablets, and smartwatches based on their unique feature profiles. This achievement facilitated precise price predictions for each device category.

3. Improved Price Predictions: The regression models, tailored to each smart device category, yielded impressive results in predicting their respective prices. The models' high accuracy and precision provided valuable insights for manufacturers to make informed pricing decisions.

8.3 Data Visualization

Data visualization played a crucial role in presenting the project's results in an intuitive and easily understandable manner. The following data visualizations aided in conveying complex information:

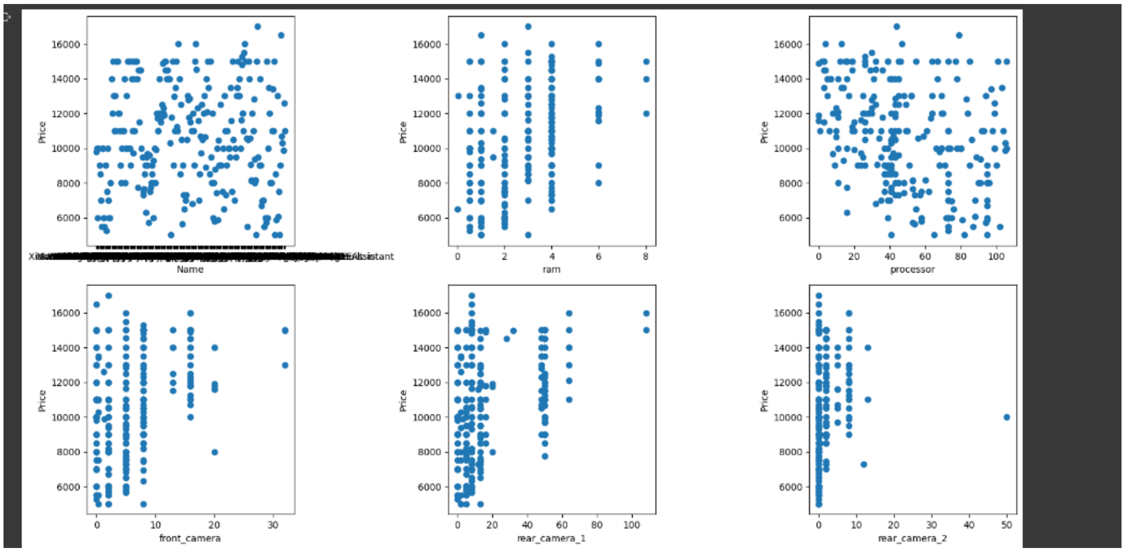


FIG 11: Features Vs Price

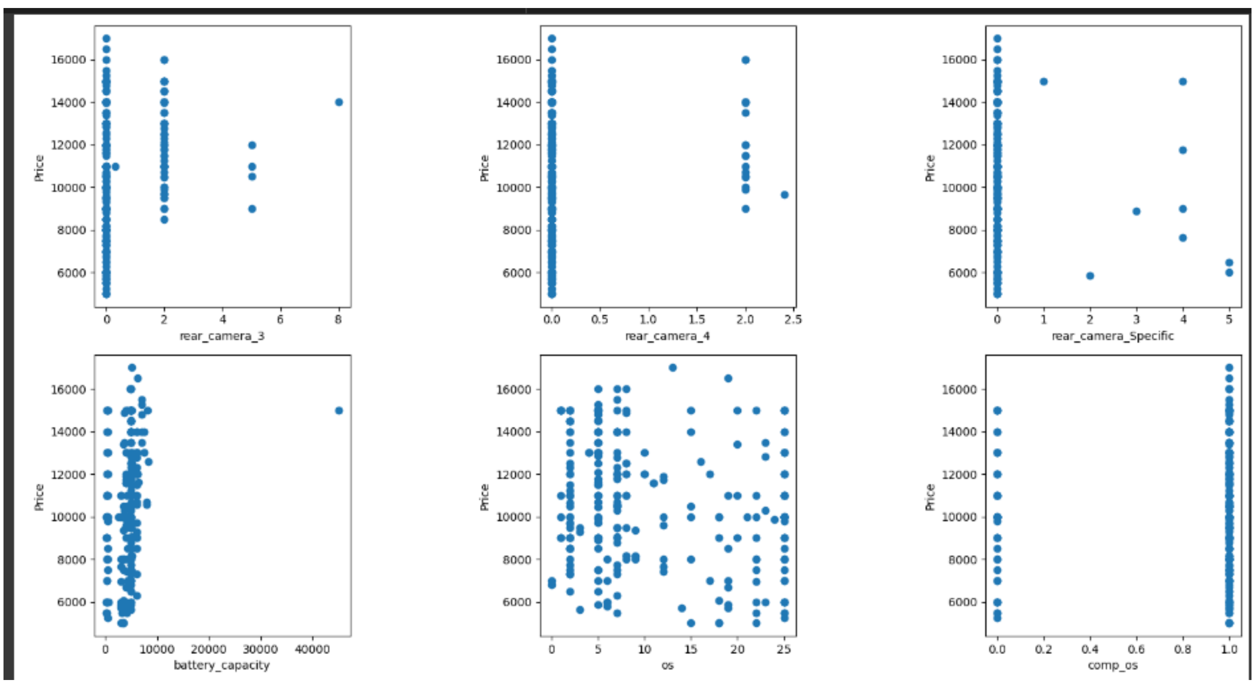


FIG 12: Features Vs Price

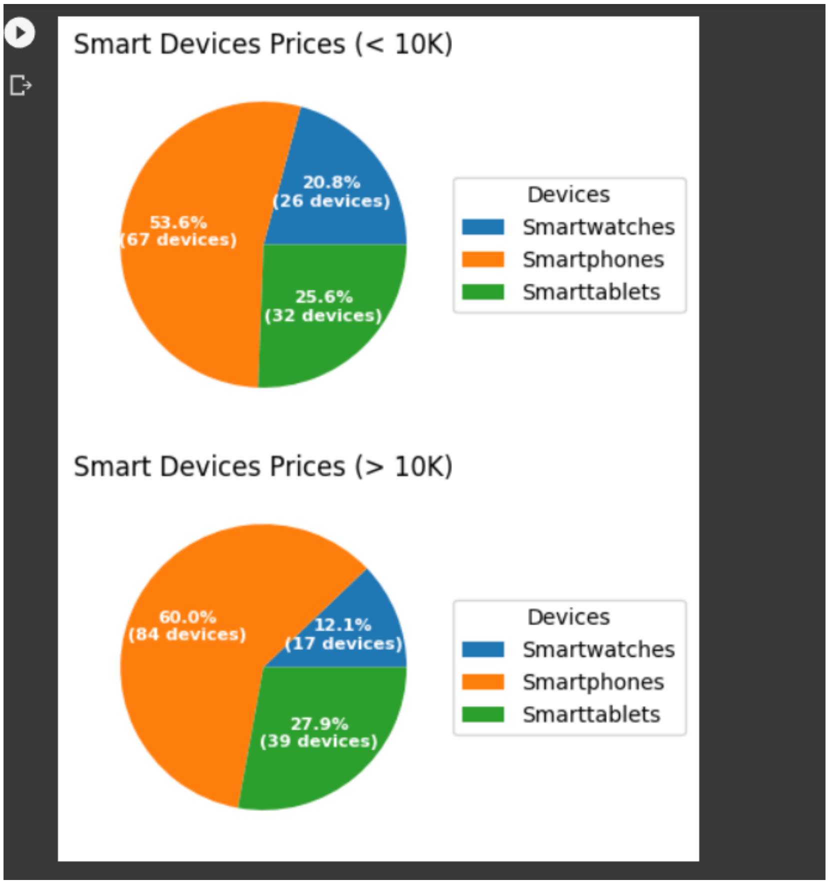


FIG 13: Smart Devices and Price Range

REFERENCES

- https://unite4buy.com/cpu/mobile-processors-ranking/
- https://nanoreview.net/en/soc-compare/unisoc-tiger-t610-vs-qualcomm-snapdragon-439
- https://www.gsmarena.com/
- https://www.giznext.com/find-your-next-phone?price=15000-20000
- https://www.gadgets360.com/mobiles/phone-finder?facet[instock_price]=5000-15000&sort=popularity_score&order=desc
- https://scikit-learn.org/stable/modules/generated/sklearn.metrics.r2_score.html
- https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OrdinalEncoder.html
- https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.pie.html
- https://chartio.com/learn/charts/pie-chart-complete-guide/