CAPSTONE PROJECT

MOBILE PRICE RANGE PREDICTION

By Yana Bhagchandani

CONTENT

Executive Summary

The Mobile Price Classification Project aims to develop a predictive model that categorizes mobile phones into price ranges based on various technical specifications.

By leveraging machine learning techniques, the project seeks to provide insights that can assist manufacturers, retailers, and consumers in understanding how different features impact the price category of a mobile phone.

Problem Statement

- **Challenge for Manufacturers and Retailers:** Mobile phone manufacturer and retailers face difficulty in accurately categorizing phones into price ranges based on their features.

- **Complexity of Specifications:** With varied technical features like battery power, RAM, and camera quality, predicting price categories has become increasingly complex.

- **Consumer Challenge:** The vast selection of mobile phones makes it difficult for consumer to compare models based solely on technical specifications.

Load and Read Data

id	battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory	m_dep	mobile_wt	n_cores	pc	px_height	px_width	ram	sc_h	sc_w	talk_time	three_g	touch_screen	wifi	price_range
1	842	0	2.2	0	1	0	7	0.6	188	2	2	20	756	2549	9	7	19	0	0	1	1
2	1021	1	0.5	1	0	1	53	0.7	136	3	6	905	1988	2631	17	3	7	1	1	0	2
3	563	1	0.5	1	2	1	41	0.9	145	5	6	1263	1716	2603	11	2	9	1	1	0	2
4	615	1	2.5	0	0	0	10	0.8	131	6	9	1216	1786	2769	16	8	11	1	0	0	2
5	1821	1	1.2	0	13	1	44	0.6	141	2	14	1208	1212	1411	8	2	15	1	1	0	1
6	1859	0	0.5	1	3	0	22	0.7	164	1	7	1004	1654	1067	17	1	10	1	0	0	1
7	1821	0	1.7	0	4	1	10	0.8	139	8	10	381	1018	3220	13	8	18	1	0	1	3
8	1954	0	0.5	1	0	0	24	0.8	187	4	0	512	1149	700	16	3	5	1	1	1	0
9	1445	1	0.5	0	0	0	53	0.7	174	7	14	386	836	1099	17	1	20	1	0	0	0
10	509	1	0.6	1	2	1	9	0.1	93	5	15	1137	1224	513	19	10	12	1	0	0	0
11	769	1	2.9	1	0	0	9	0.1	182	5	1	248	874	3946	5	2	7	0	0	0	3
12	1520	1	2.2	0	5	1	33	0.5	177	8	18	151	1005	3826	14	9	13	1	1	1	3
13	1815	0	2.8	0	2	0	33	0.6	159	4	17	607	748	1482	18	0	2	1	0	0	1
14	803	1	2.1	0	7	0	17	1	198	4	11	344	1440	2680	7	1	4	1	0	1	2
15	1866	0	0.5	0	13	1	52	0.7	185	1	17	356	563	373	14	9	3	1	0	1	0
16	775	0	1	0	3	0	46	0.7	159	2	16	862	1864	568	17	15	11	1	1	1	0
17	838	0	0.5	0	1	1	13	0.1	196	8	4	984	1850	3554	10	9	19	1	0	1	3
18	595	0	0.9	1	7	1	23	0.1	121	3	17	441	810	3752	10	2	18	1	1	0	3
19	1131	1	0.5	1	11	0	49	0.6	101	5	18	658	878	1835	19	13	16	1	1	0	1
20	682	1	0.5	0	4	0	19	1	121	4	11	902	1064	2337	11	1	18	0	1	1	1
21	772	0	1.1	1	12	0	39	0.8	81	7	14	1314	1854	2819	17	15	3	1	1	0	3
22	1709	1	2.1	0	1	0	13	1	156	2	2	974	1385	3283	17	1	15	1	0	0	3

About the Dataset:

- The mobile price classification dataset consists of 2000 records with 21 features that describe the characteristics of mobile phones.
- The dataset aims to classify mobile phones into one of four price ranges (0 to 3). By leveraging classification techniques, it is possible to predict the price range of a mobile device based on its specifications.

Methodology

Data Collection

The dataset contains various attributes of mobile phones, including technical specifications.

Exploratory Data Analysis (EDA)

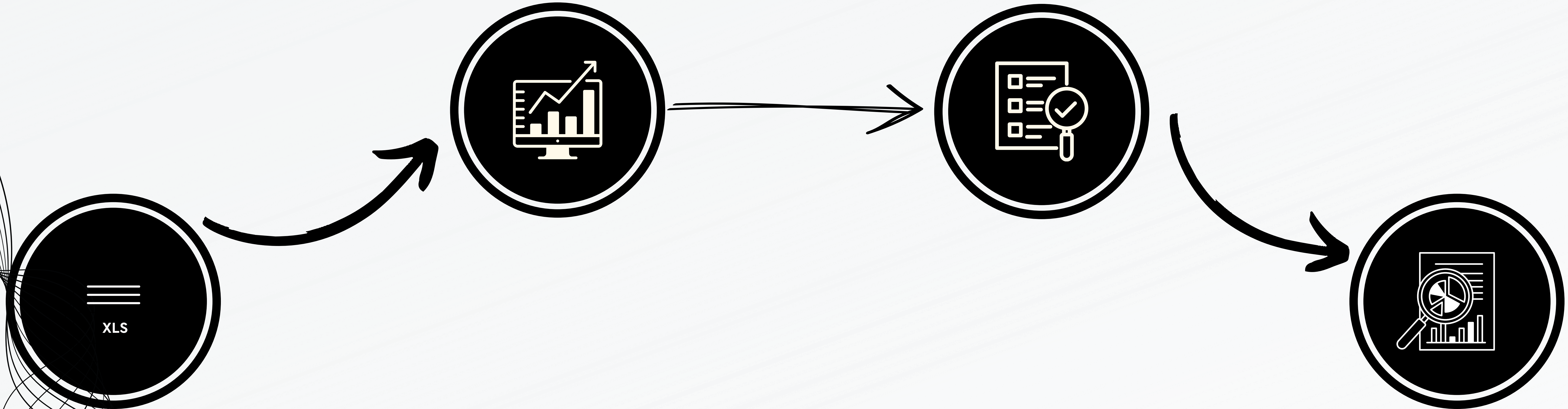
Conduct an in-depth analysis to identify patterns and relationships between mobile features and their price categories.

Model Selection

Different classification algorithms (e.g., Logistic Regression, Decision Trees, Random Forest, etc.) will be applied.

Model Evaluation

The chosen model will be evaluated using cross-validation, confusion matrix analysis, and performance metrics.

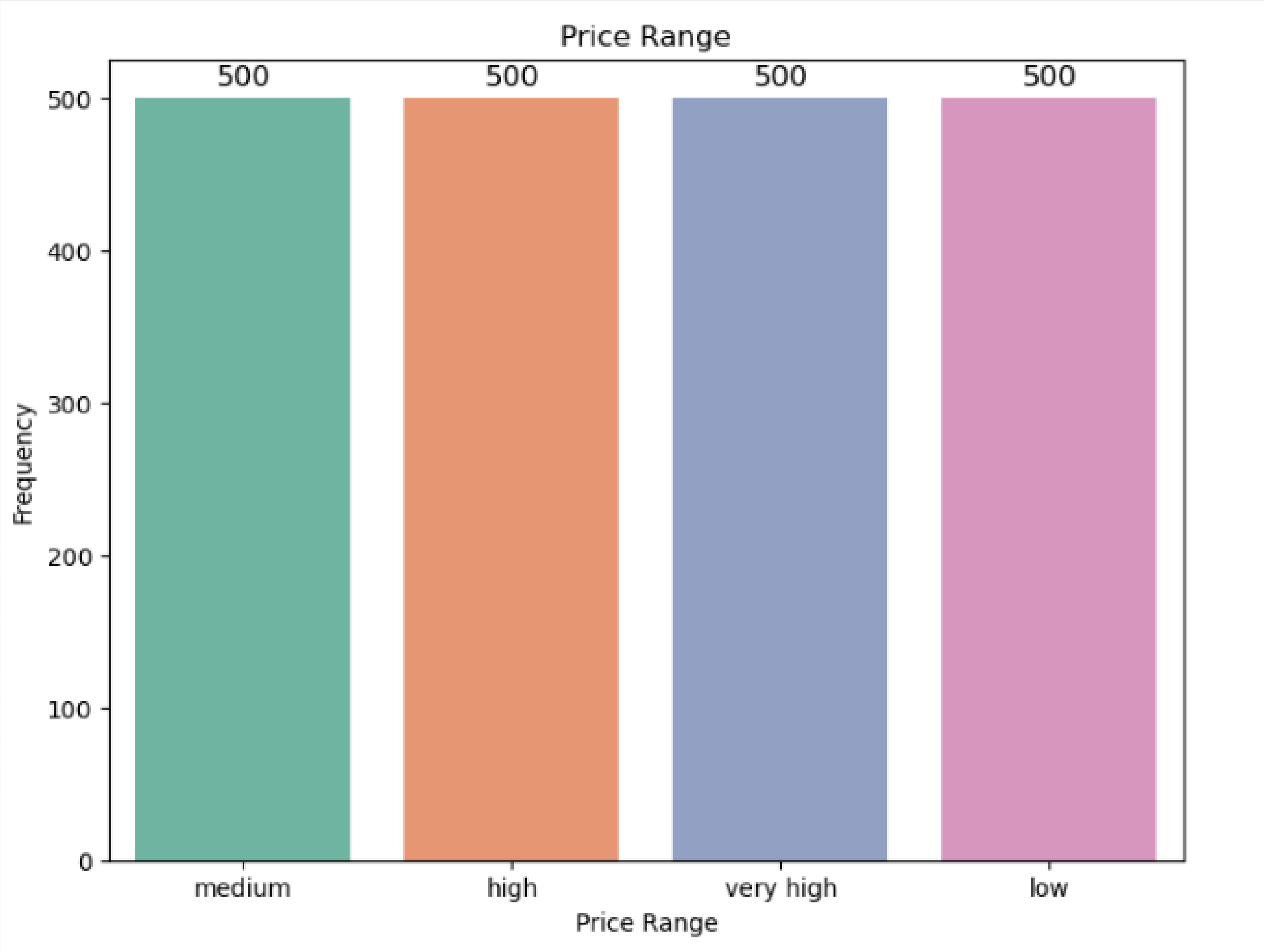


Exploratory Data Analysis (EDA) Question?

1. What is the distribution of mobile phones across different price ranges?
2. How is phone specification features distributed across different price ranges?
3. Are there significant correlations between phone specifications and price?
4. Which features show the strongest relationship with higher price categories?
5. What is the distribution of key features like Bluetooth, Dual SIM, WiFi, etc. across mobile phones?

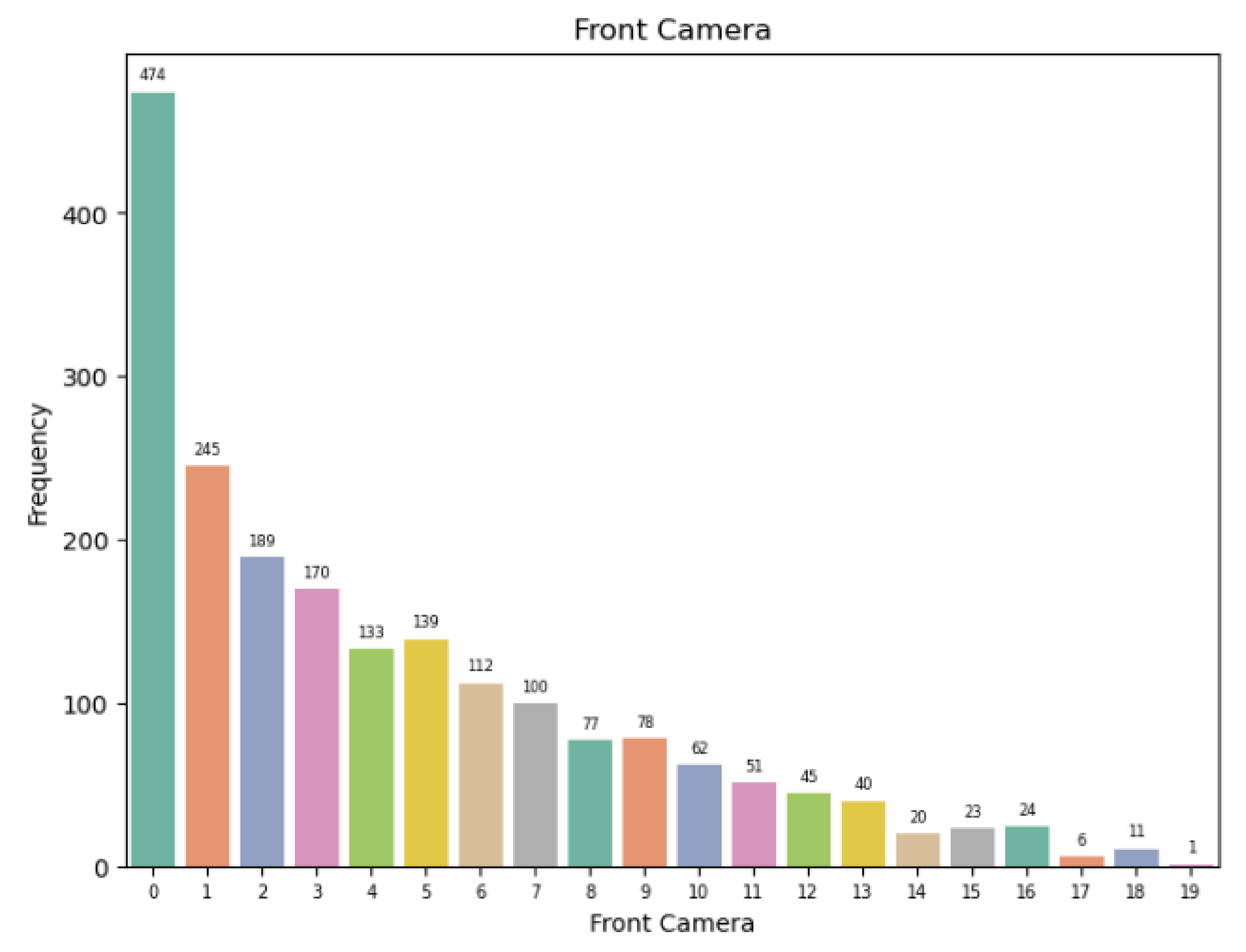
Analysis Result

Distribution Of Price Range Categories



- The dataset contains a balanced distribution of products across different price ranges.

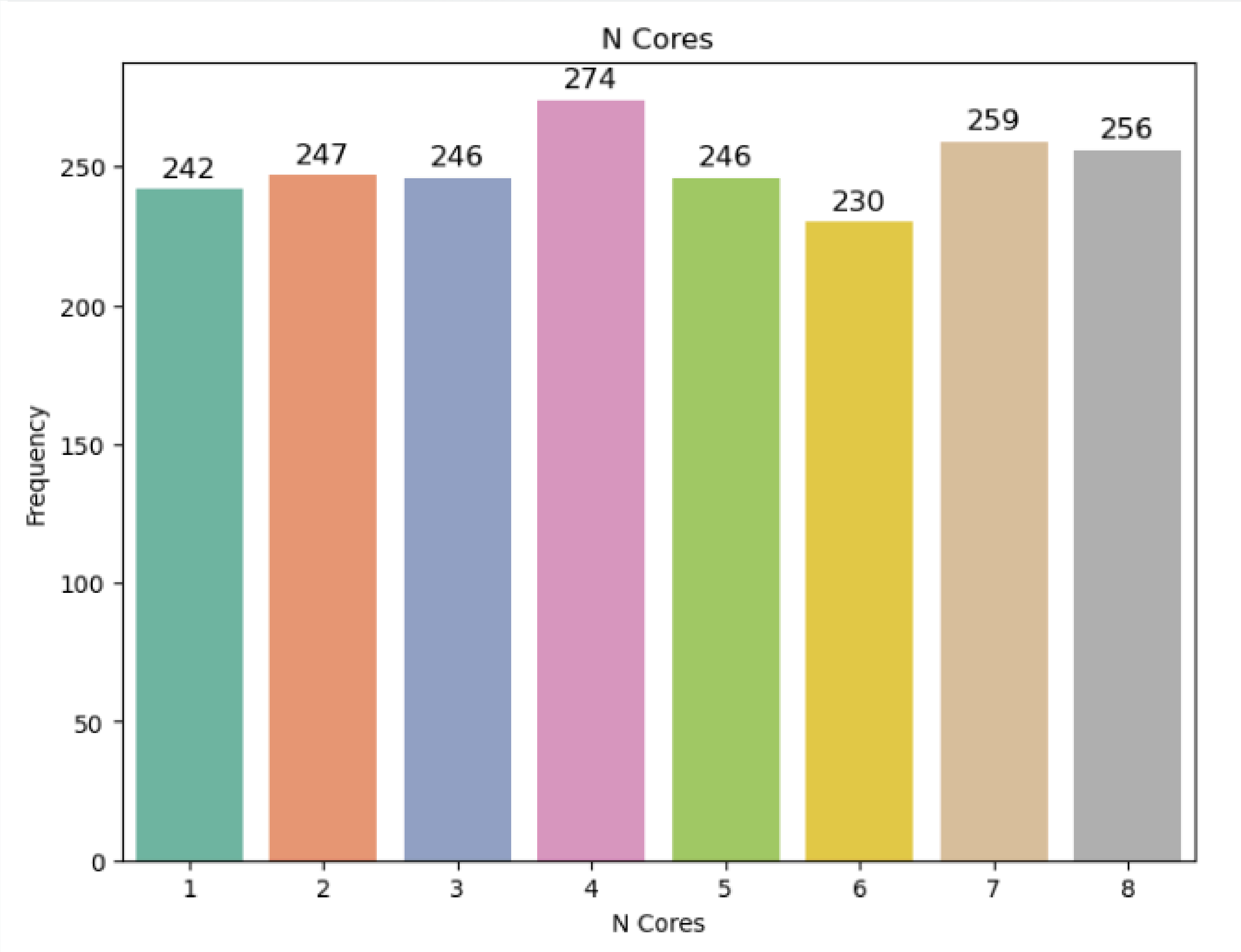
Distribution Of ‘Front Camera’



- The majority of devices have front cameras in the range of 0 to 5 megapixels, with a significant peak at 2 megapixels.

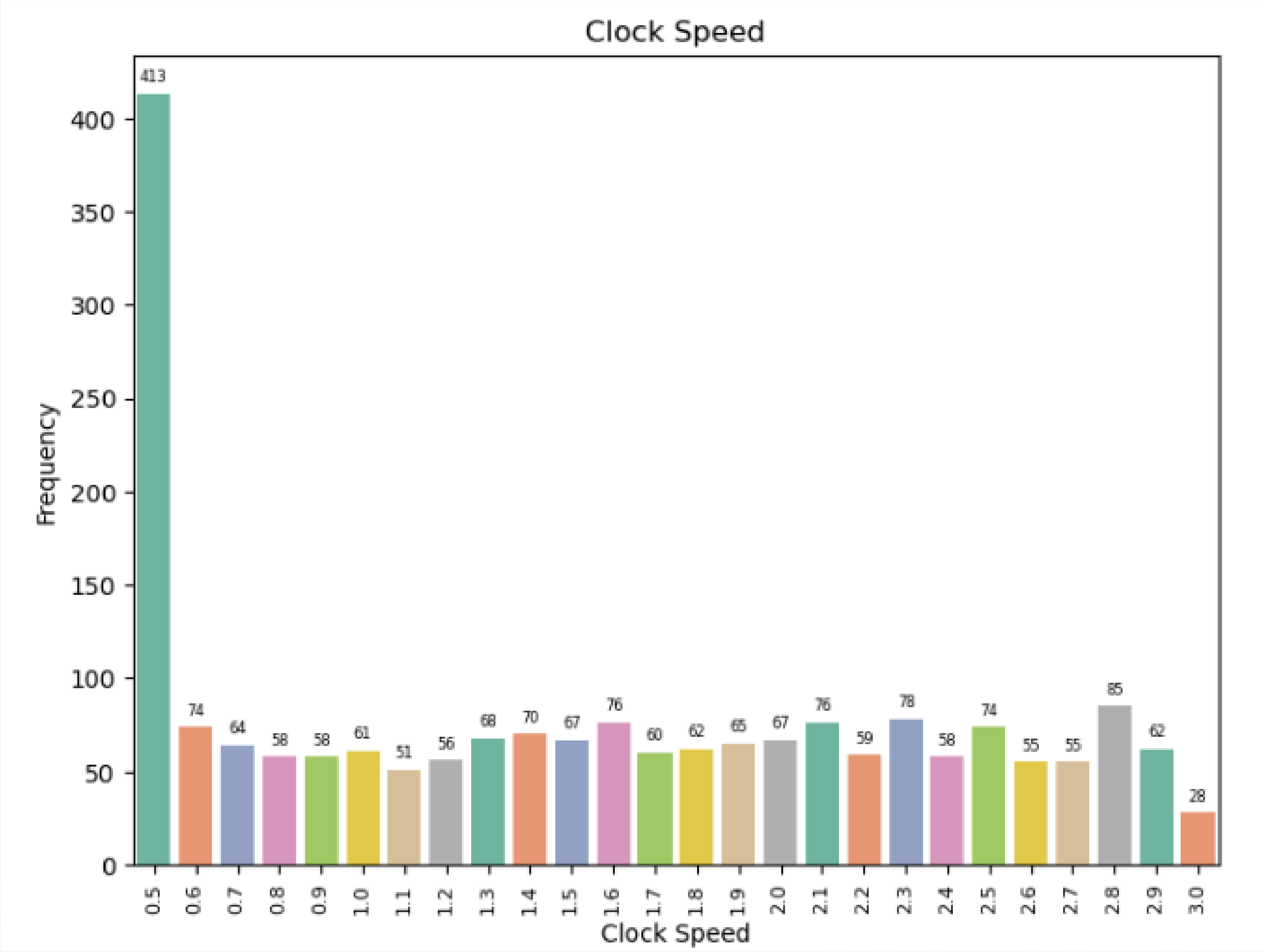
Analysis Result

Distribution Of 'N Cores'



- The distribution of N cores is relatively balanced, with most devices having 2, 3, or 4 cores.

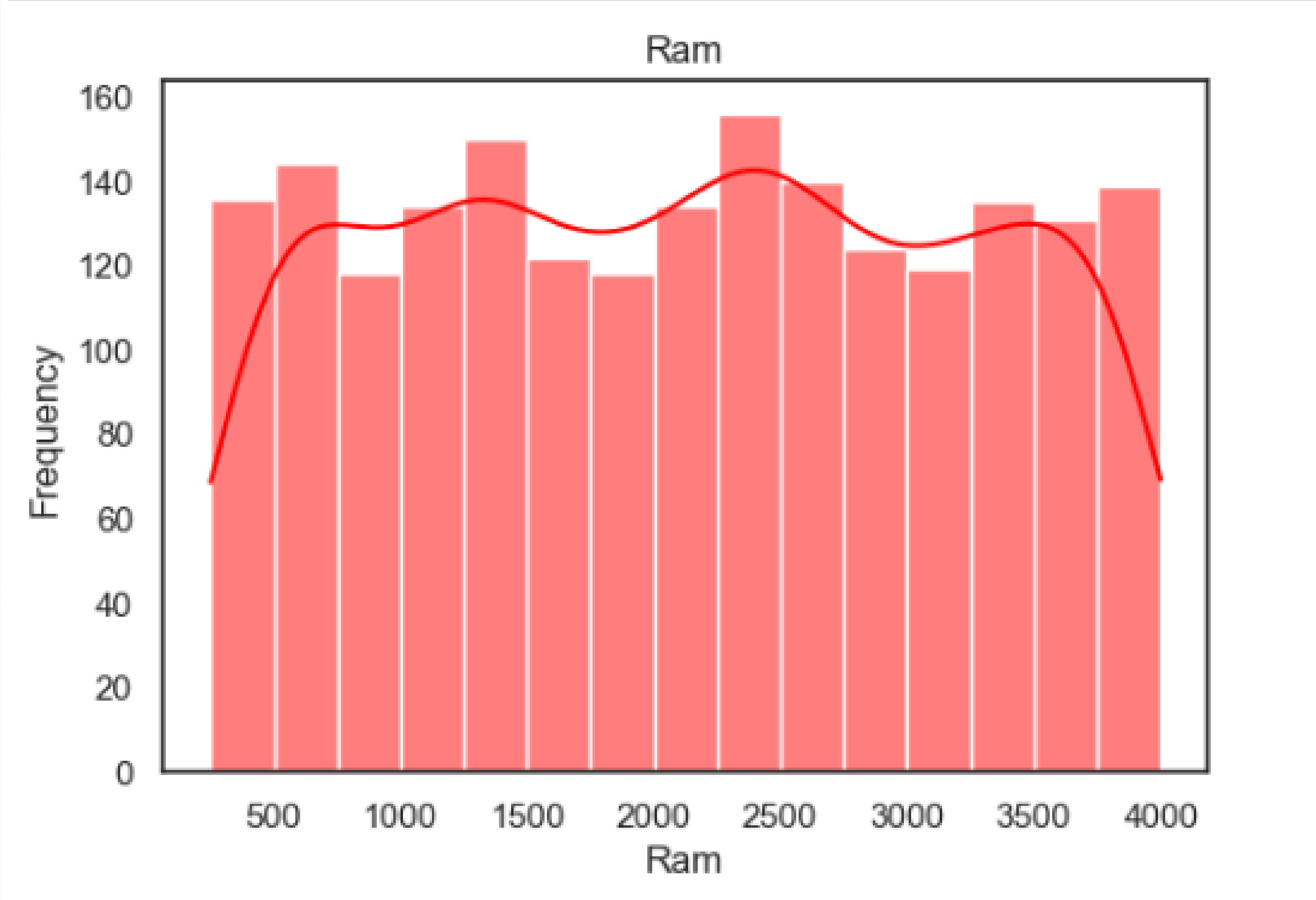
Distribution Of 'Clock Speed'



- Most devices have clock speeds between 0.5 GHz and 1.5 GHz, with a peak around 0.6 GHz.

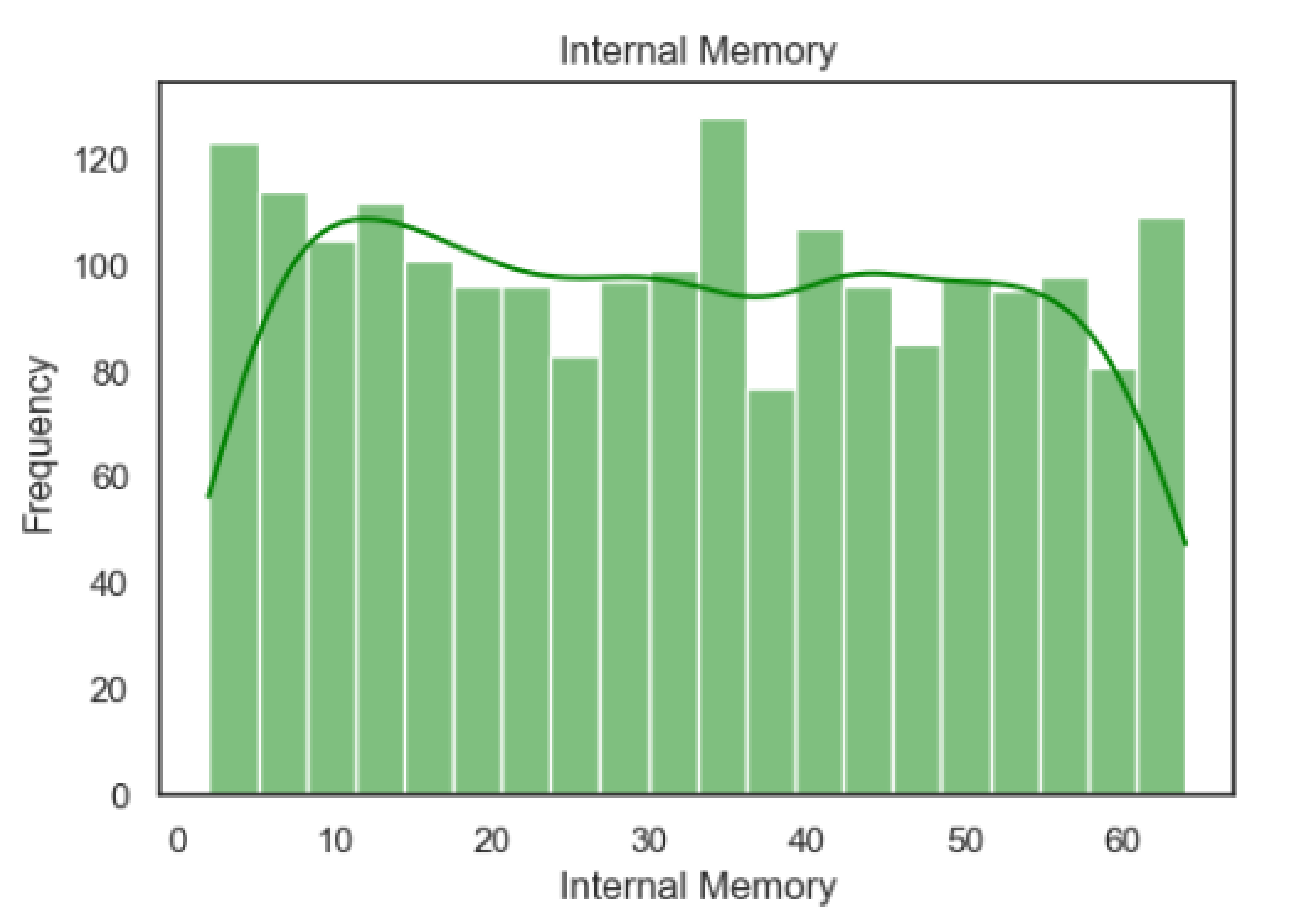
Analysis Result

Distribution Of 'RAM'



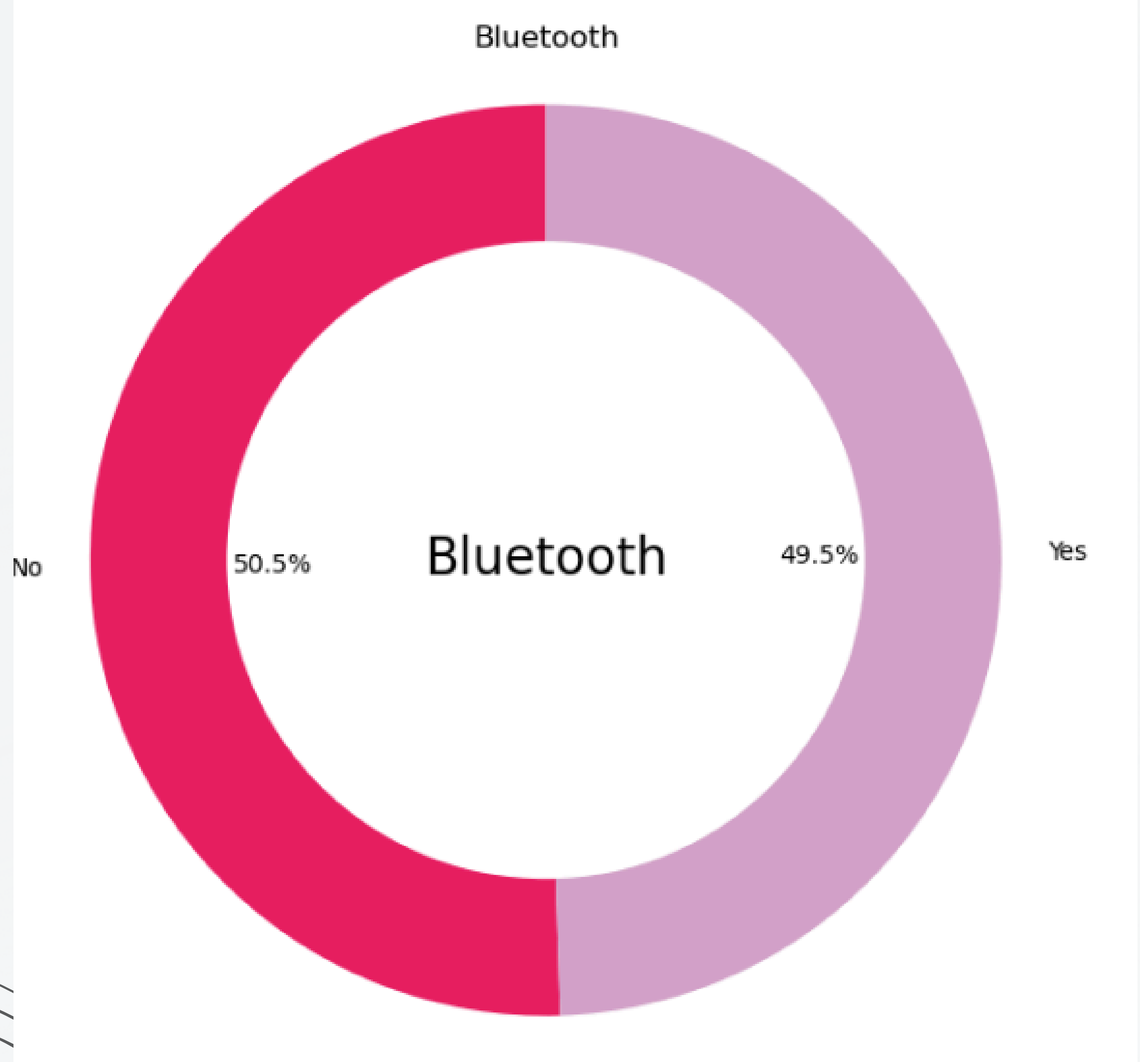
- RAM: The distribution of RAM is relatively uniform, with a slight increase in frequency for devices with 2000 MB RAM and a gradual decrease for larger capacities.

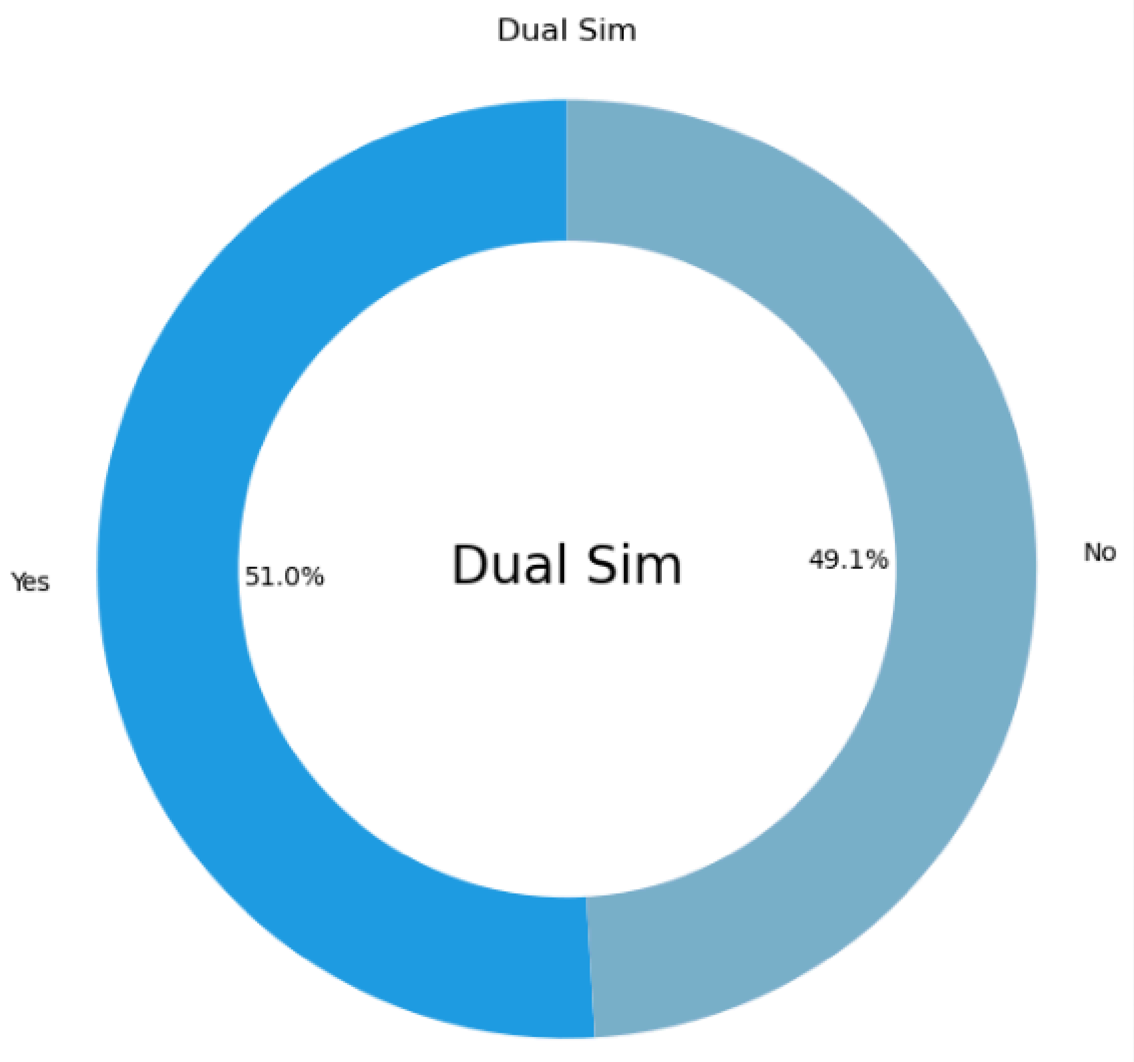
Distribution Of 'Internal Memory'

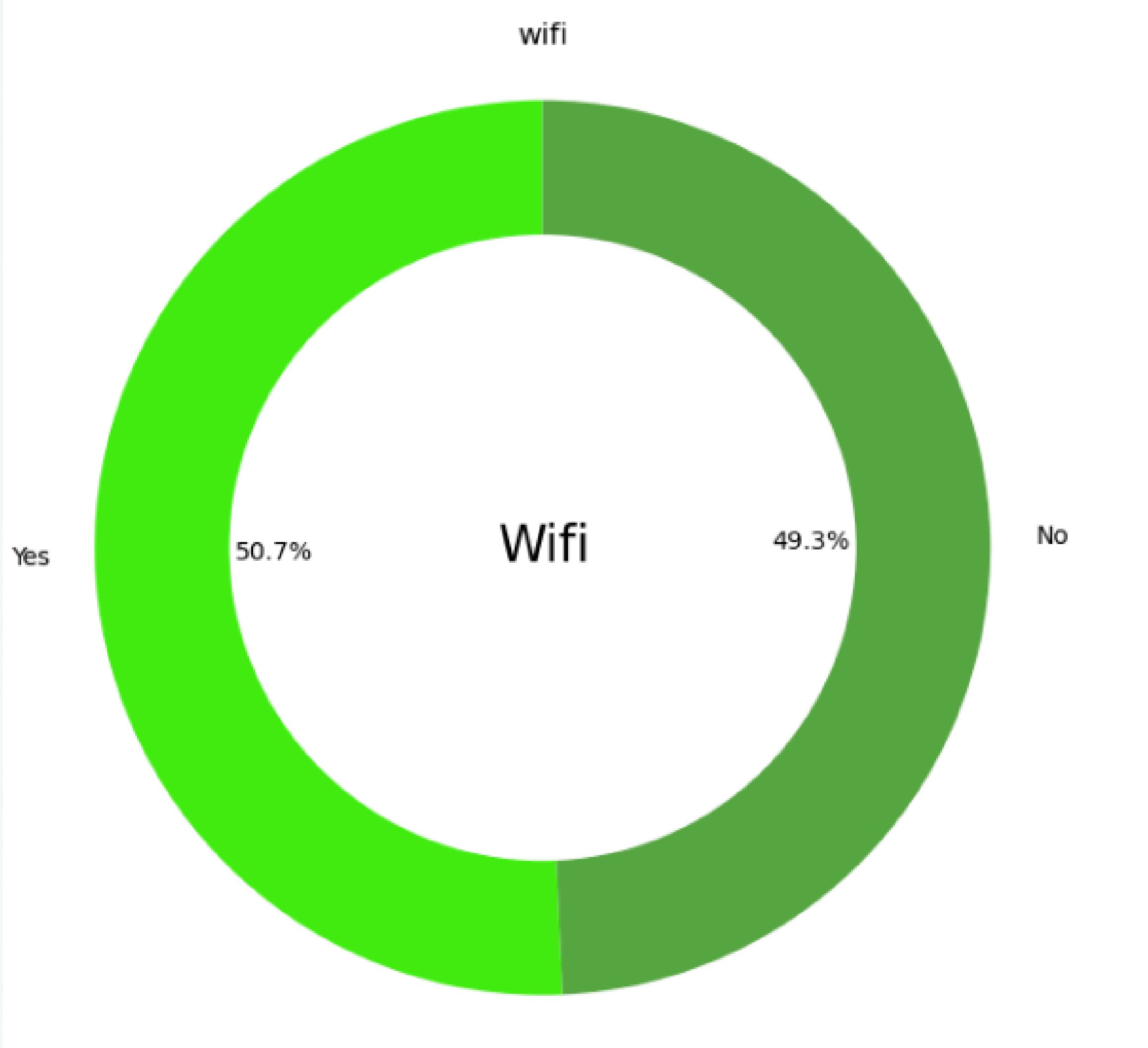


- Internal Memory: The distribution of internal memory is skewed to the right, with a higher frequency of devices having lower storage capacities (16 GB) and a decreasing frequency for larger capacities.

Analysis Result

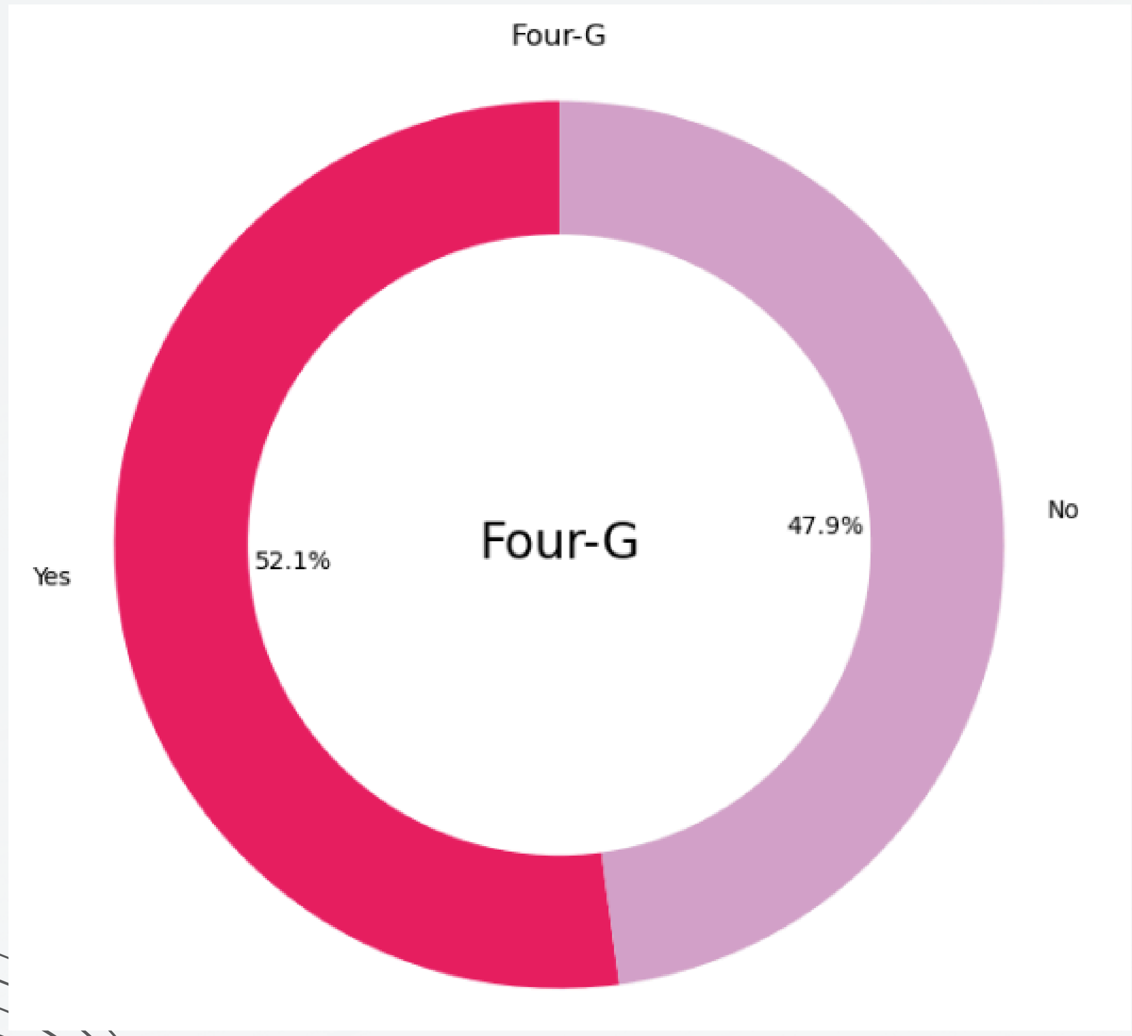


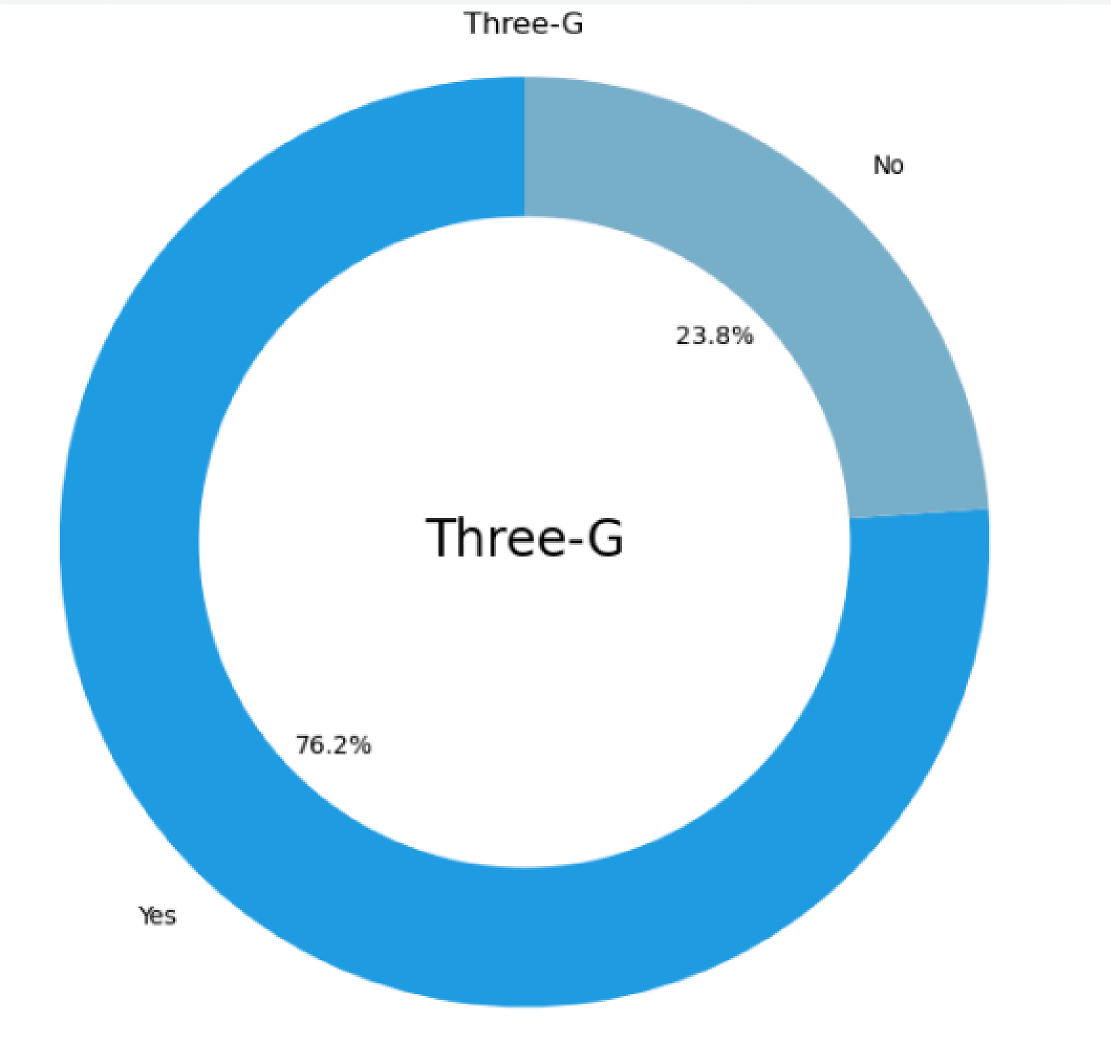


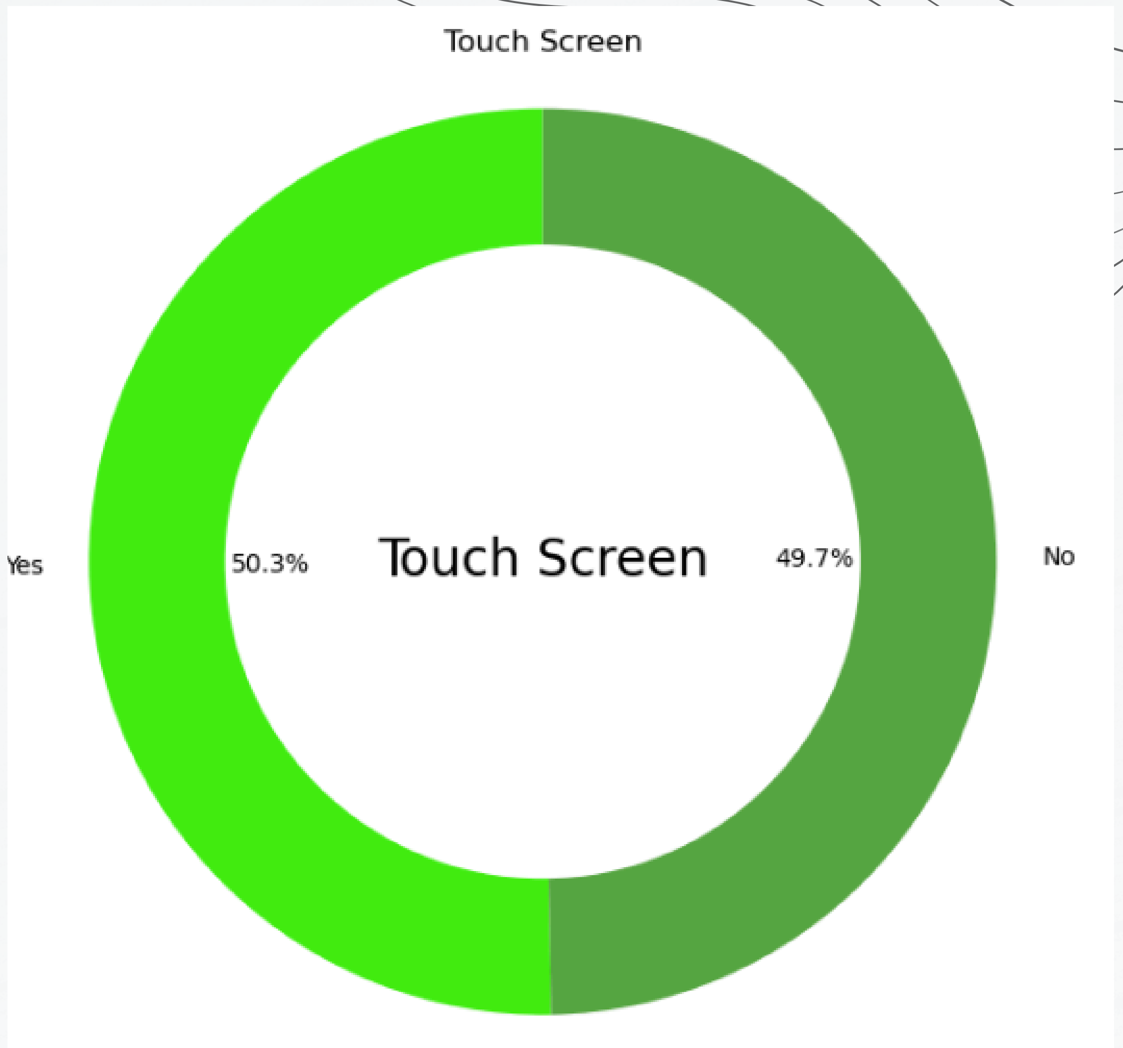


- Bluetooth: Roughly 50.5% of devices have Bluetooth, while 49.5% do not.
- Dual Sim: Approximately 51.0% of devices support dual SIM functionality, and 49.0% do not.
- WiFi: A majority of devices (50.7%) have WiFi, while 49.3% do not.

Analysis Result



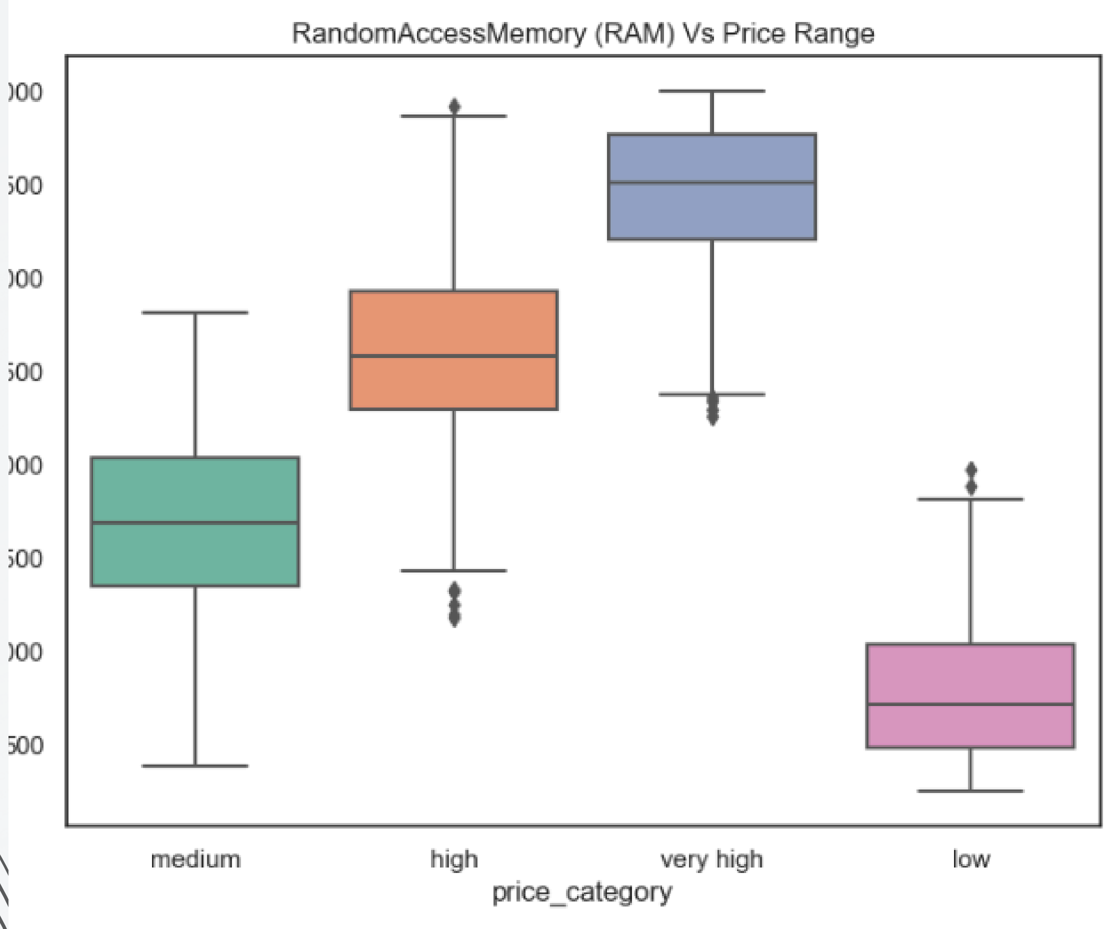




- Four-G: Approximately 52.2% of devices support 4G connectivity, while 47.8% do not.
- Three-G: A significantly smaller proportion (23.0%) of devices have 3G support, with 77.0% lacking it.
- Touch Screen: The vast majority (85.7%) of devices have touch screens, while only 14.3% do not.

Analysis Result

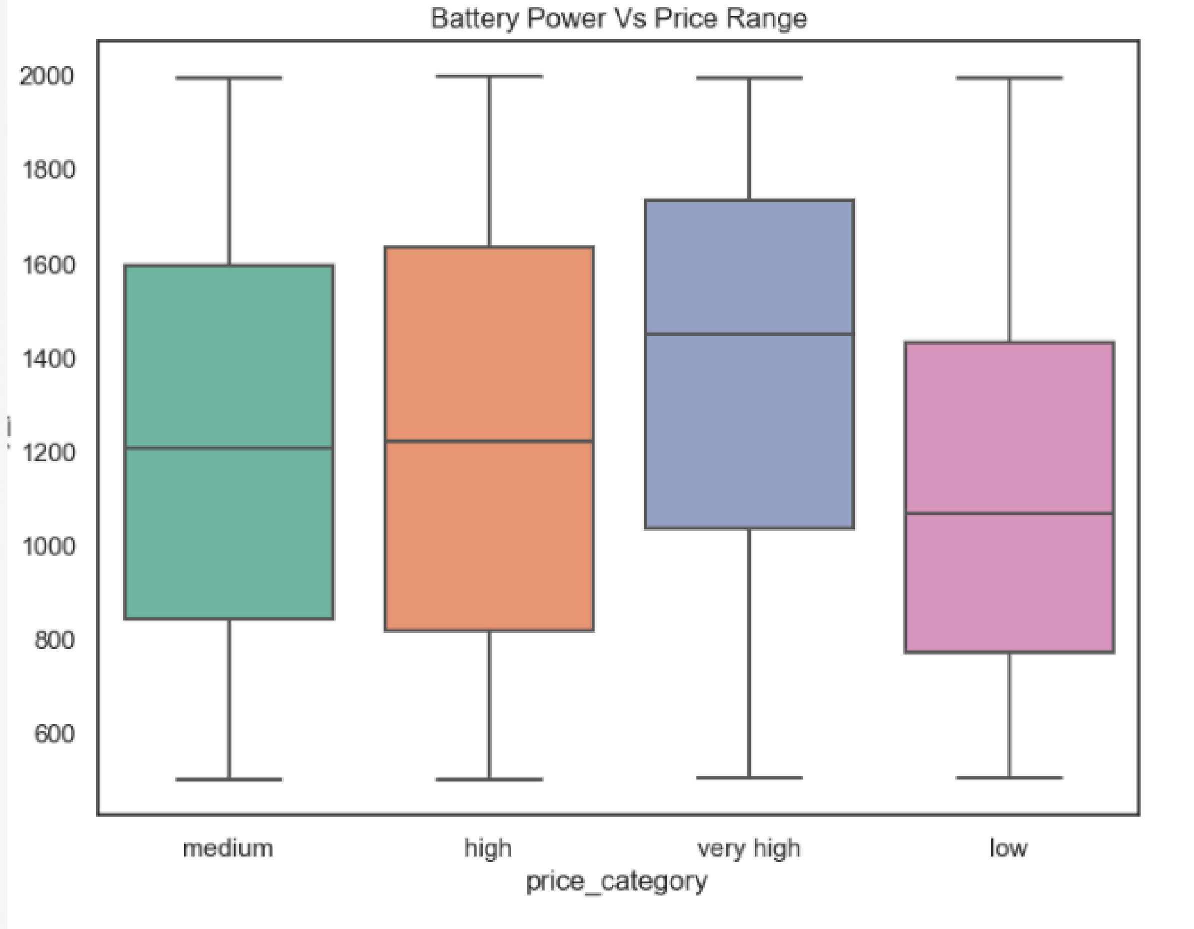
Relation with RAM to Prices



- **RAM vs Price Category::**

1. RAM shows a clear upward trend with price. Higher-priced phones (very high and high) have significantly more RAM, indicating a strong correlation between RAM and price.

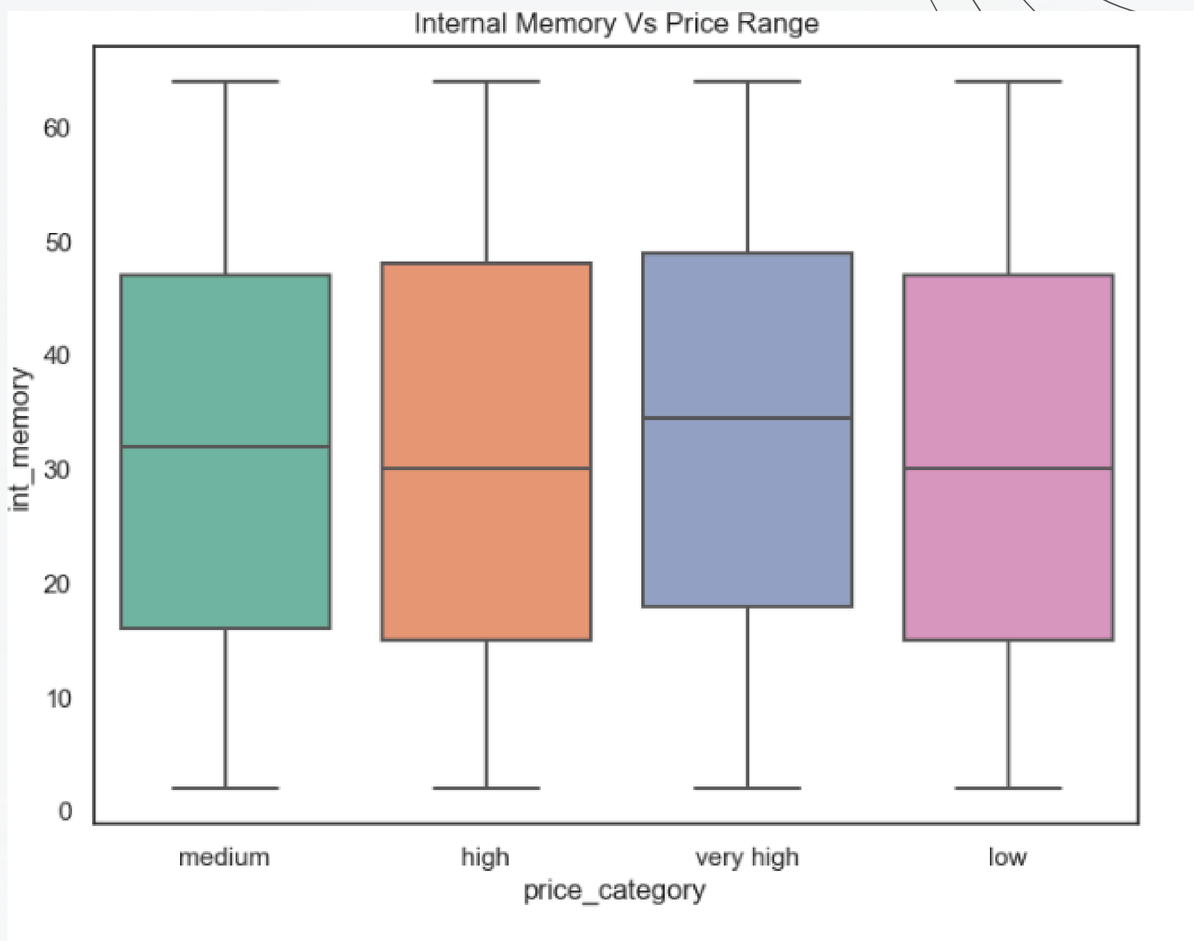
Relation with Battery Power to Prices



- **Battery Power vs Price Category:**

1. Higher price categories, such as medium and high, tend to have mobile phones with higher median battery power.
2. The lowest price category (low) generally has phones with lower battery capacities.

Relation with Internal Memory to Prices

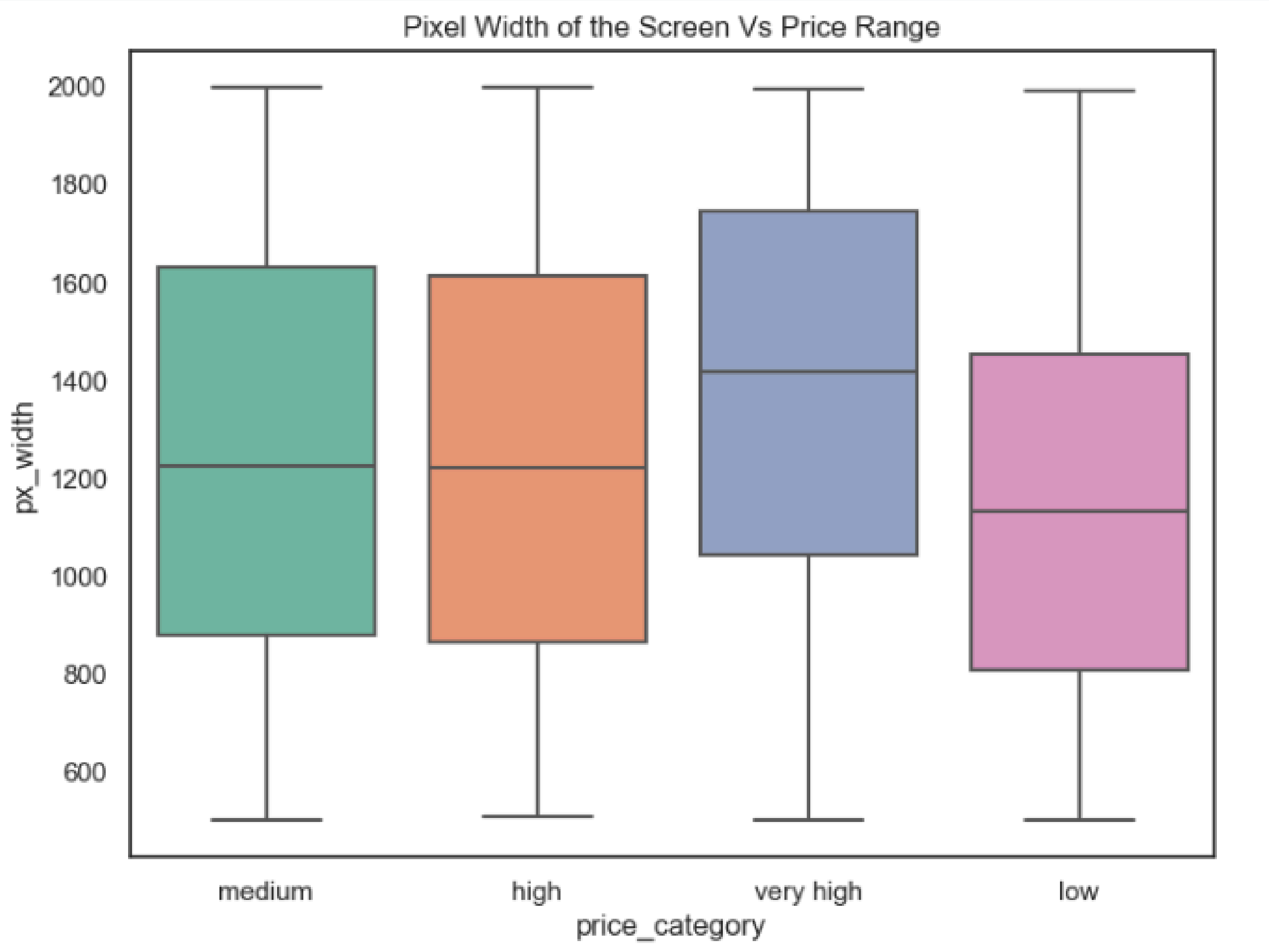


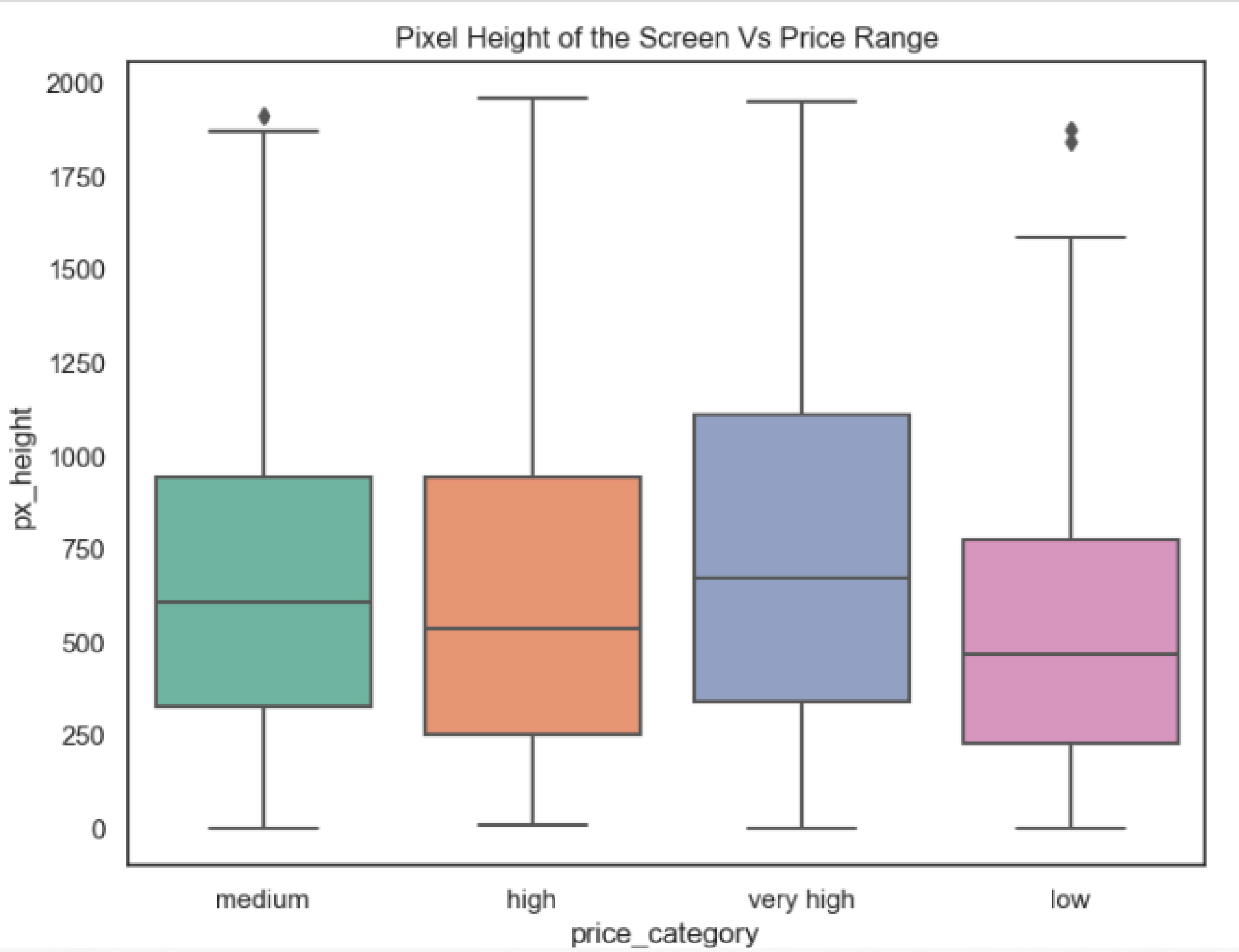
- **Internal Memory Vs Price Category:**

1. Internal memory appears to be relatively consistent across all price ranges, suggesting that other features may play a more significant role in determining the price of mobile phones.

Analysis Result

Relation with Pixel Height and Pixel Width to Prices



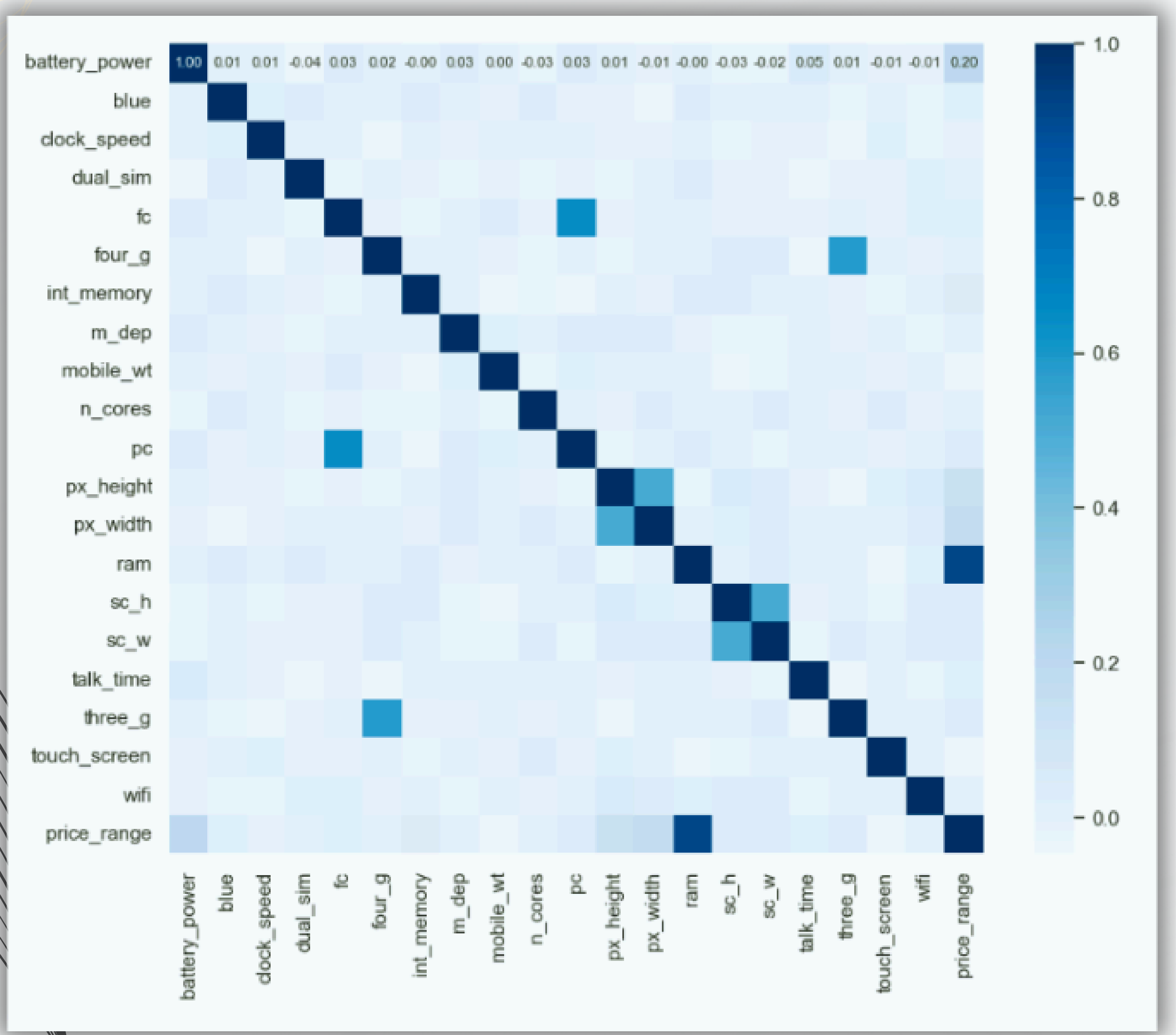


- **Pixel Height and Pixel Width vs Price Category:**

There is no distinct trend between pixel height/width and price categories. All price categories seem to have a wide range of pixel resolutions, suggesting other factors may be more significant for price determination.

Conclusion (Strongest Relationship):
Overall, features such as battery power and RAM show clear relationships with price categories, while internal memory and pixel dimensions appear to have a weaker or less consistent correlation with price.

Correlation



The correlation heatmap reveals several important relationships between the features and the target variable price_range. Key insights include:

- RAM has a strong positive correlation with price_range, meaning higher RAM is likely associated with higher price categories.
- Battery power and pixel resolution (width and height) also show positive correlations with the price range, indicating their influence on pricing.
- Other features like internal memory, primary camera (pc), and number of cores (n_cores) exhibit moderate correlations with price_range

Conclusion

	ACCURACY	PRECISION	RECALL	F1_SCORE	COHEN-KAPPA
Logistic Regression	0.9575	0.958041	0.9575	0.957563	0.943218
Decision Tree	0.8200	0.826024	0.8200	0.820109	0.759507
Random Forest	0.8875	0.889500	0.8875	0.888095	0.849787
Random Forest with Tuned Hyperparameter	0.8800	0.885293	0.8800	0.881247	0.839884
AdaBoost with RandomForest Estimator	0.8875	0.889970	0.8875	0.888151	0.849737
AdaBoost with DecisionTree Estimator	0.8300	0.826024	0.8200	0.820109	0.759507

Best Model Evaluation: Logistic Regression

```
#Classification Report
print(classification_report(y_test, y_pred_log))
```

```
              precision    recall  f1-score   support

           0       0.98      0.96      0.97       105
           1       0.93      0.98      0.95        91
           2       0.95      0.93      0.94        92
           3       0.97      0.96      0.96       112

    accuracy                           0.96       400
   macro avg       0.96      0.96      0.96       400
weighted avg       0.96      0.96      0.96       400
```

```
#Confusion Matrix
print(confusion_matrix(y_test, y_pred_log))
```

```
[[101   4   0   0]
 [  2  89   0   0]
 [  0   3  86   3]
 [  0   0   5 107]]
```

Interpretation of Model Performance Metrics:

Based on the provided evaluation metrics, the Random Forest model with tuned hyperparameters achieved the best overall performance:

- **Best Performer:** Logistic Regression is the top performer in all metrics, particularly in accuracy, precision, and Cohen-Kappa, indicating it’s the most robust model for this dataset.
- **Next Best:** Random Forest and AdaBoost with Random Forest Estimator provide solid performances, though they don't outperform Logistic Regression.
- **Least Reliable:** Decision Tree and its AdaBoost variant perform the worst, with lower accuracy and Cohen-Kappa, indicating these models are less suitable.

Prediction Using Test Data

Before Prediction

id	battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory	m_dep	mobile_wt	n_cores	pc	px_height	px_width	ram	sc_h	sc_w	talk_time	three_g	touch_screen	wifi
1	1043	1	1.8	1	14	0	5	0.1	193	3	16	226	1412	3476	12	7	2	0	1	0
2	841	1	0.5	1	4	1	61	0.8	191	5	12	746	857	3895	6	0	7	1	0	0
3	1807	1	2.8	0	1	0	27	0.9	186	3	4	1270	1366	2396	17	10	10	0	1	1
4	1546	0	0.5	1	18	1	25	0.5	96	8	20	295	1752	3893	10	0	7	1	1	0
5	1434	0	1.4	0	11	1	49	0.5	108	6	18	749	810	1773	15	8	7	1	0	1
6	1464	1	2.9	1	5	1	50	0.8	198	8	9	569	939	3506	10	7	3	1	1	1
7	1718	0	2.4	0	1	0	47	1	156	2	3	1283	1374	3873	14	2	10	0	0	0
8	833	0	2.4	1	0	0	62	0.8	111	1	2	1312	1880	1495	7	2	18	0	1	1
9	1111	1	2.9	1	9	1	25	0.6	101	5	19	556	876	3485	11	9	10	1	1	0
10	1520	0	0.5	0	1	0	25	0.5	171	3	20	52	1009	651	6	0	5	1	0	1
11	1500	0	2.2	0	2	0	55	0.6	80	7	6	503	1336	3866	13	7	20	0	1	0
12	1343	0	2.9	0	2	1	34	0.8	171	3	6	235	1671	3911	15	8	8	1	1	1
13	900	1	1.4	1	0	0	30	1	87	2	3	829	1893	439	6	2	20	1	0	0
14	1190	1	2.2	1	5	0	19	0.9	158	5	15	227	1856	992	13	0	16	1	1	0
15	630	0	1.8	0	8	1	51	0.9	193	8	9	1315	1323	2751	17	6	3	1	1	0
16	1846	1	1	0	5	1	53	0.7	106	8	7	185	1832	563	9	5	10	1	0	1
17	1985	0	0.5	1	14	1	26	1	163	2	17	613	1511	2083	13	3	14	1	1	0
18	1042	0	2.9	0	5	1	48	0.2	186	4	15	335	532	2187	9	2	5	1	0	0
19	1231	1	1.7	1	2	1	37	0.2	194	2	3	82	1771	3902	19	12	15	1	0	1
20	1488	0	2.6	0	9	0	37	0.7	189	4	20	47	559	2524	5	0	6	0	0	0
21	968	0	0.6	0	8	1	7	0.7	151	1	17	504	1930	1357	15	1	16	1	1	0

After Prediction

```
#Predictions using Test Data
y_pred_test = log_reg.predict(df_test1)
```

```
#Assigning predicted values to the new column
df_test['price_range'] = y_pred_test
```

```
#displaying the updated (where the unseen data is predicted) dataframe
df_test.head()
```

k_speed	dual_sim	fc	four_g	int_memory	m_dep	mobile_wt	n_cores	...	px_height	px_width	ram	sc_h	sc_w	talk_time	three_g	touch_screen	wifi	price_range
1.8	1	14	0	5	0.1	193	3	...	226	1412	3476	12	7	2	0	1	0	3
0.5	1	4	1	61	0.8	191	5	...	746	857	3895	6	0	7	1	0	0	3
2.8	0	1	0	27	0.9	186	3	...	1270	1366	2396	17	10	10	0	1	1	2
0.5	1	18	1	25	0.5	96	8	...	295	1752	3893	10	0	7	1	1	0	3
1.4	0	11	1	49	0.5	108	6	...	749	810	1773	15	8	7	1	0	1	1

Prediction Results Using Test Data:

The slide shows the application of a machine learning model to predict outcomes from test data. The "Before Prediction" section presents the input features, while the "After Prediction" section includes the predicted price range, demonstrating the model's capability to forecast based on the given parameters.

THANK'S FOR WATCHING