Mobile Price Prediction Analysis

Koguru Varshitha

varshitha3110@gmail.com

Intoduction

- The mobile market has a wide range of products with varying features and prices.
- Consumers and manufacturers both benefit from accurate price prediction.
- Goal: Predict price range of smartphones based on specifications using ML.

Problem Statement

We want to predict the **price range** of a mobile phone based on its **technical specifications**, **without knowing the actual price**.
This is a **supervised classification problem** where the target (label) is a **categorical variable**.

🧠 **Why is this useful?**

- **For consumers**: Helps estimate phone price before buying or comparing.
- **For manufacturers**: Supports competitive pricing strategies based on features.
- **For sellers**: Can automate the price categorization for large inventories.

Dataset Overview

🗂️ **What is the dataset about?**

The dataset contains information about various mobile phones and their specifications. Each row represents a unique phone, and each column is a feature (e.g., RAM, battery, screen size).

📦 **Dataset Summary:**

- **Total Records**: ~2000 mobile phone entries
- **Total Features**: 20 columns (excluding target variable)
- **Target Variable**: `price_range` (values: 0 to 3)

✅ **Why this dataset is good:**

- Clean: No missing values
- Balanced: Even distribution of price categories
- Ideal for classification tasks
- Offers both **numerical** and **categorical** features

Input Features

- id:ID
- battery_power:Total energy a battery can store in one time measured in mAh
- blue:Has bluetooth or not
- clock_speed:speed at which microprocessor executes instructions
- dual_sim:Has dual sim support or not
- fc:Front Camera mega pixels
- four_g:Has 4G or not
- int_memory:Internal Memory in Gigabytes
- m_dep:Mobile Depth in cm
- mobile_wt:Weight of mobile phone
- n_cores:Number of cores of processor
- pc:Primary Camera mega pixels
- px_height:Pixel Resolution Height

- px_width:Pixel Resolution Width
- ram:Random Access Memory in Megabytes
- sc_h:Screen Height of mobile in cm
- sc_w:Screen Width of mobile in cm
- talk_time:longest time that a single battery charge will last when you are
- three_g:Has 3G or not
- touch_screen:Has touch screen or not
- wifi:Has wifi or not

Exploratory Data Analysis (EDA)

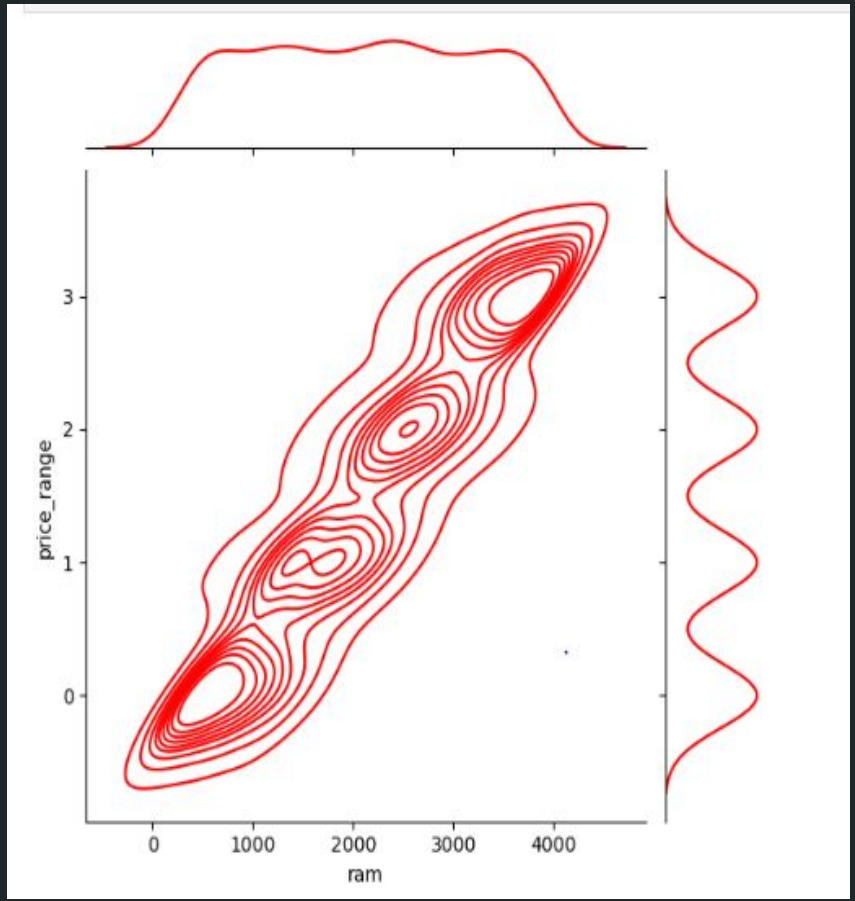
EDA is the process of visually and statistically analyzing the dataset to understand patterns, relationships, and anomalies before applying machine learning.

🧪 **Purpose of EDA:**

- Understand how features are distributed
- Detect outliers or skewed data
- Identify which features impact the target variable (`price_range`)
- Check correlation between features

🔍 What this code does:

- `sns.jointplot`: Creates a **joint plot**, which combines a **bivariate plot** (between x and y) and **univariate plots** (distributions of x and y) in one figure.
- `x='ram'`: x-axis will show **RAM** values.
- `y='price_range'`: y-axis will show the **price category** (0 to 3).
- `data=dataset`: Uses your DataFrame.
- `color='red'`: Sets the plot color to red.
- `kind='kde'`: Uses a **kernel density estimate**, which shows the probability density of data rather than actual points.

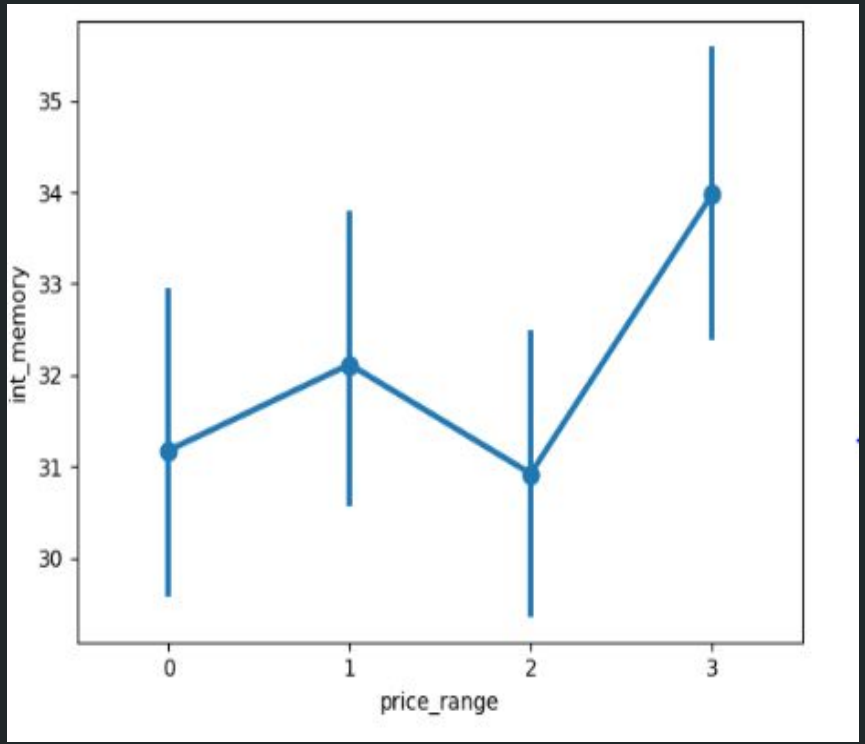


What it does:

- **X-axis:** price categories (0 to 3)
- **Y-axis:** average internal memory (in GB)
- Each point represents the mean `int_memory` for that price range.
- Error bars show confidence intervals, giving a sense of variation.

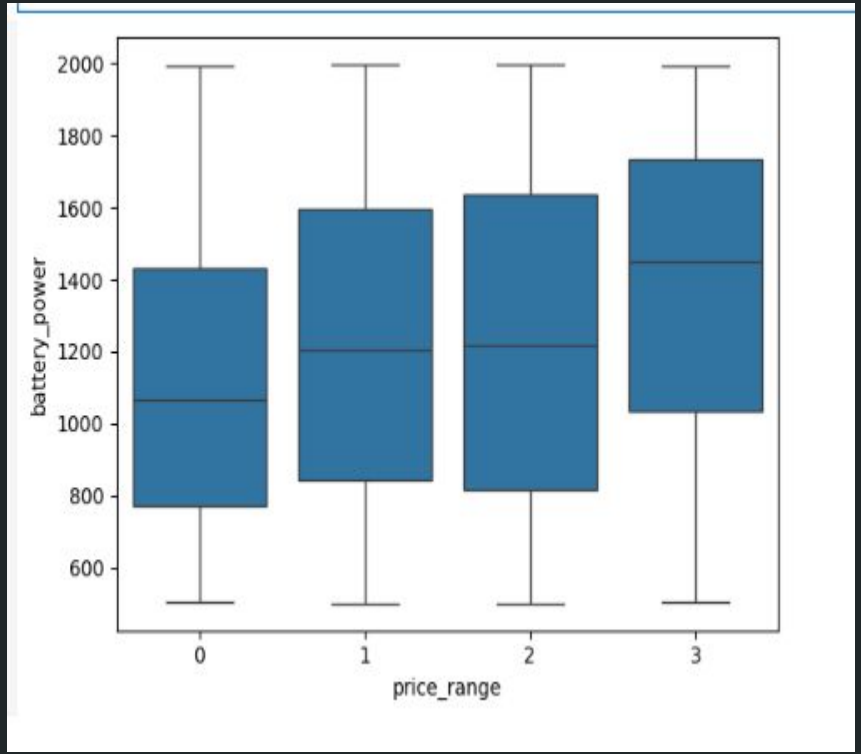
Interpretation:

You'll likely see that **internal memory increases as price range increases** — phones in higher price brackets generally have more storage.



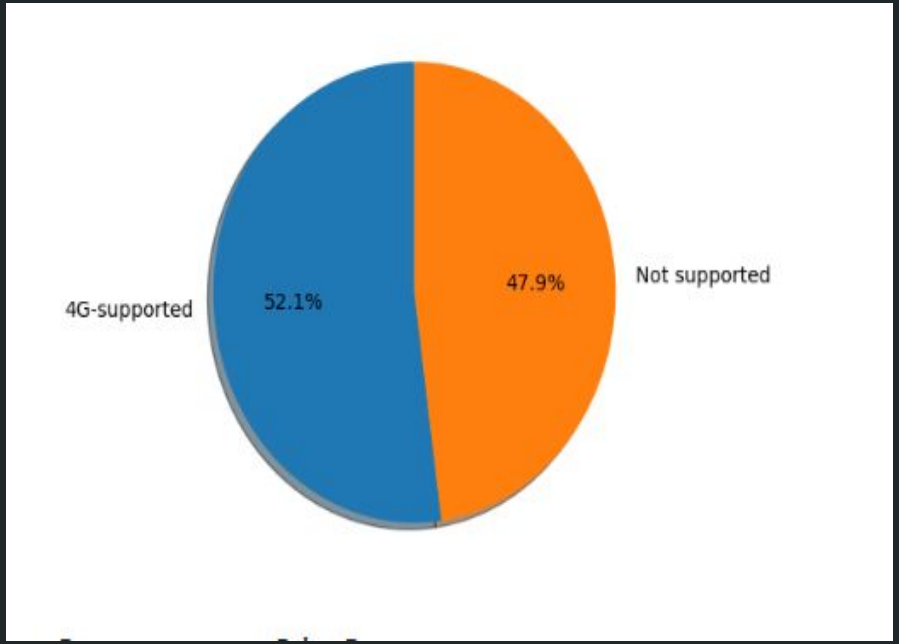
Insights found:

- If higher price ranges have boxes shifted upwards, it means more expensive phones generally have higher battery power.
- If lower price ranges have a tighter box and lower median, budget phones may have smaller batteries with less variation.
- Outliers can be detected



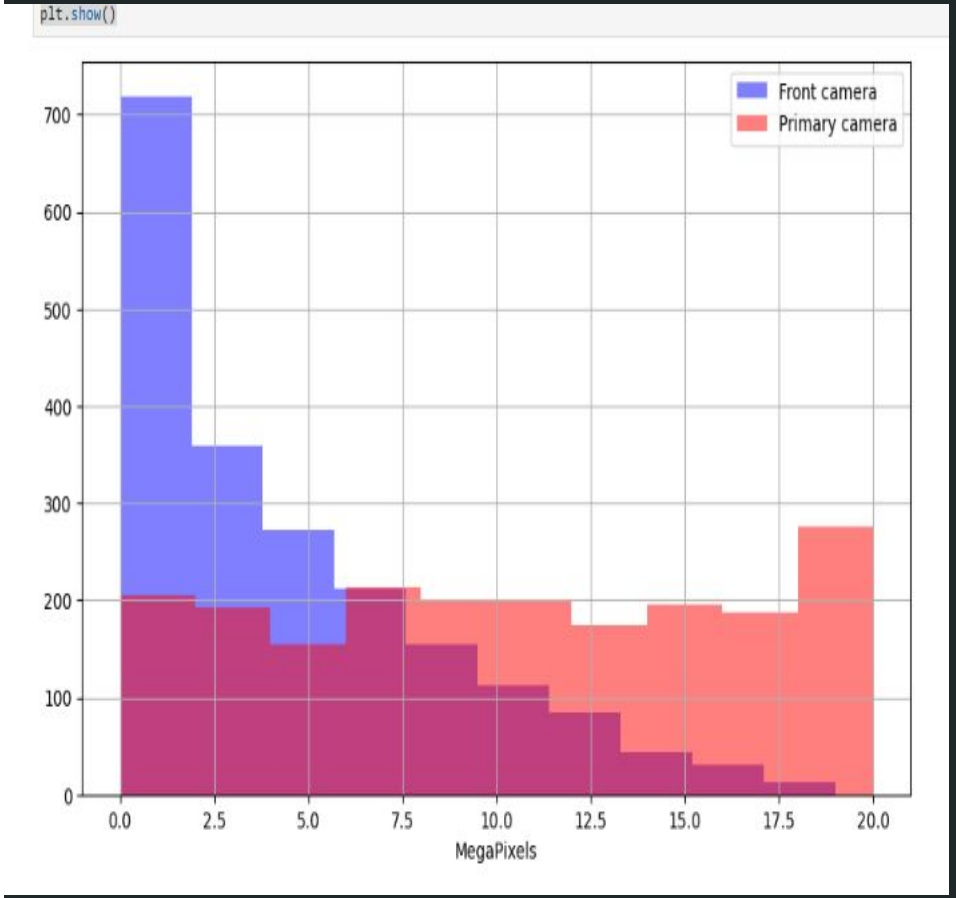
What the pie chart tells you:

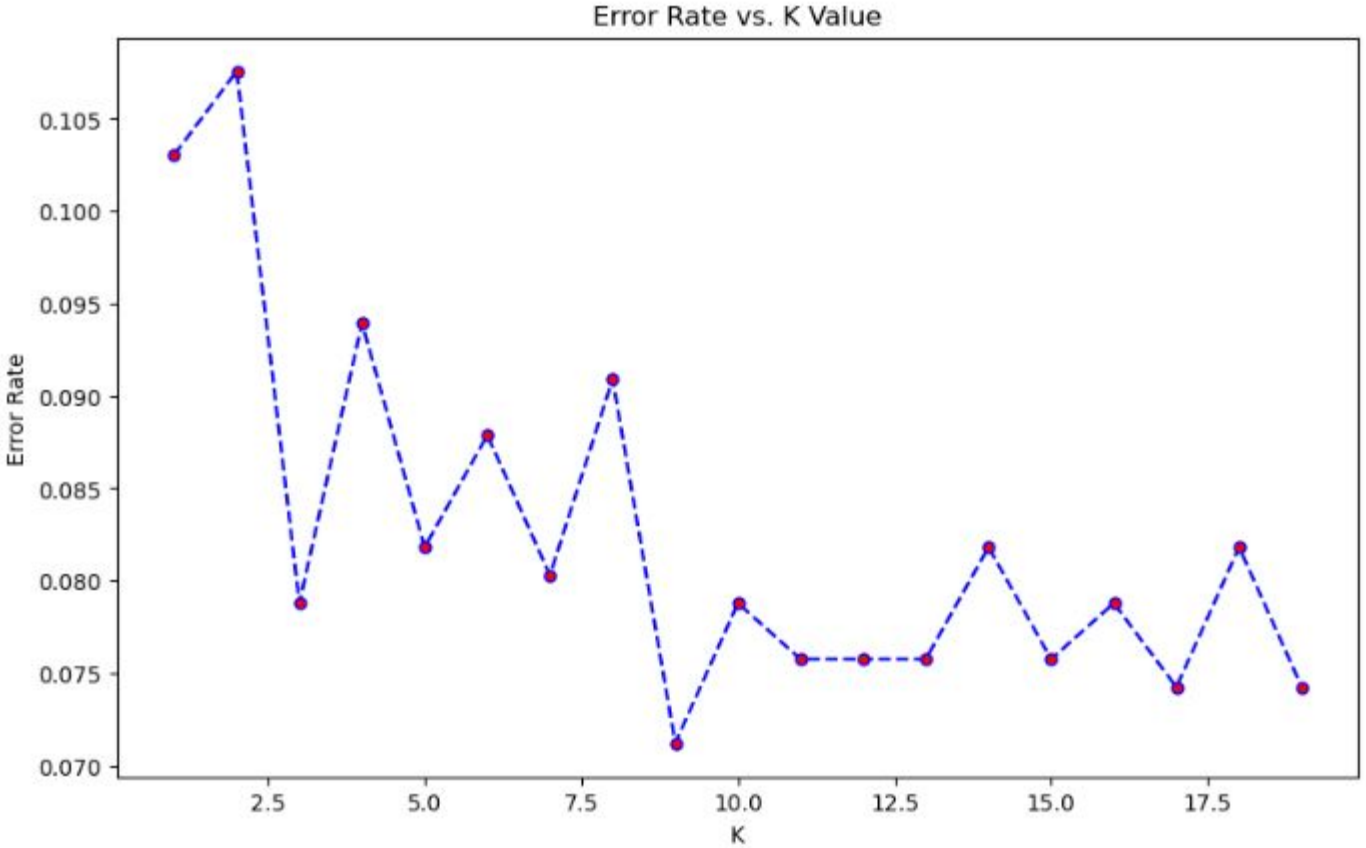
- Percentage of phones **with 4G capability** vs **without 4G** in your dataset.
- Useful for showing how common 4G is across mobile phones analyzed.



What you'll see:

- Overlapping histograms showing how megapixel counts are distributed for front vs primary cameras.
- Usually, the primary camera has a wider and higher megapixel range (more pixels) compared to the front camera.
- Transparency (`alpha=0.5`) helps you see overlapping areas.




Error Rate vs. K Value
Error Rate
0.105
0.100
0.095
0.090
0.085
0.080
0.075
0.070
2.5
5.0
7.5
10.0
12.5
15.0
17.5
K

What will I do next?

Enhance my code with present it.