

**LAPORAN PRAKTIKUM
UTS DASI**



LAPORAN PRAKTIKUM UTS DASI
Analisis smartphone Indian Market

Disusun Oleh:

11423004 : Marselino Tambunan
11423018 : Eduward Simanjuntak
11423019 : Hans Manalu

Teknologi Rekayasa Perangkat Lunak

FAKULTAS VOKASI

INSTITUT TEKNOLOGI DEL

2025

DAFTAR ISI

BAB I	2
Pendahuluan.....	3
1.1 Latar Belakang.....	3
1.2 Tujuan.....	3
1.3 Rumusan Masalah	4
BAB II LANDASAN TEORI	5
2.1 Konsep Data Science	5
2.2 Data Preprocessing.....	5
2.3 Visualisasi Data	6
2.4 Konsep Analisis Statistik	7
BAB III Metode Penelitian.....	8
3.1. Tahapan Analysis Data	8
3.1.1 Dataset	8
3.1.3. Data Preprocessing (Advance Preprocessing)	13
3.1.4. Data Visualization	23
BAB IV Hasil Dan Pembahasan	27
4.1. Analisis dan Visualisasi Data	27
BAB V Kesimpulan.....	32

BAB I

Pendahuluan

1.1 Latar Belakang

Smartphone saat ini sudah menjadi kebutuhan utama bagi banyak orang. Setiap merek berlomba-lomba menghadirkan produk dengan spesifikasi dan fitur yang beragam, seperti RAM besar, kamera bagus, baterai tahan lama, hingga dukungan 5G.

Melihat banyaknya variasi smartphone di pasaran, kelompok kami tertarik melakukan analisis data smartphone menggunakan pendekatan data science. Tujuannya adalah untuk mengetahui hubungan antara spesifikasi dan harga serta melihat pola umum dari data yang tersedia.

Dataset yang digunakan berisi sekitar 3.260 data smartphone dari berbagai merek dan sistem operasi, dengan atribut seperti harga, RAM, penyimpanan, baterai, prosesor, dan fitur tambahan. Melalui tahapan data collection, data preprocessing, data visualization, dan data analysis, proyek ini diharapkan dapat memberikan gambaran tentang faktor yang memengaruhi harga smartphone di pasaran.

Selain itu, proyek ini juga menjadi sarana bagi kelompok kami untuk menerapkan ilmu data science secara nyata, mulai dari pengolahan data hingga menghasilkan insight sederhana yang bermanfaat. Dengan cara ini, kami belajar bagaimana data yang ada di sekitar kita bisa diubah menjadi informasi yang berguna dan mendukung pengambilan keputusan yang lebih baik.

1.2 Tujuan

Tujuan utama dari proyek ini adalah untuk menganalisis hubungan antara spesifikasi dan harga smartphone di pasaran menggunakan pendekatan data science. Melalui analisis ini, diharapkan dapat ditemukan pola umum yang membantu memahami faktor-faktor apa saja yang paling berpengaruh terhadap harga sebuah smartphone.

Secara lebih rinci, tujuan penelitian ini adalah:

1. Bagaimana pola distribusi harga smartphone berdasarkan merek dan spesifikasi yang dimiliki?
2. Fitur atau spesifikasi apa yang paling berpengaruh terhadap tinggi rendahnya harga smartphone?
3. Apakah kapasitas RAM, penyimpanan, dan baterai memiliki hubungan yang signifikan dengan harga?
4. Sejauh mana fitur modern seperti 5G, NFC, dan fast charging memengaruhi nilai jual smartphone?

5. Apakah terdapat perbedaan mencolok dalam harga antara smartphone dengan jenis prosesor yang berbeda?

1.3 Rumusan Masalah

Berdasarkan tujuan yang telah disebutkan, maka rumusan masalah dalam proyek analisis ini dapat dijabarkan sebagai berikut:

1. Faktor-faktor apa saja yang paling memengaruhi harga smartphone di pasaran?
2. Apakah terdapat hubungan yang signifikan antara kapasitas RAM, penyimpanan, atau daya baterai dengan harga smartphone?
3. Bagaimana perbandingan strategi harga antara merek smartphone yang berbeda?
4. Apakah smartphone dengan spesifikasi tinggi selalu memiliki harga yang lebih mahal, atau ada pengecualian tertentu?
5. Pola umum seperti apa yang dapat ditemukan dari data smartphone yang dianalisis?

BAB II LANDASAN TEORI

2.1 Konsep Data Science

Data Science (Ilmu Data) merupakan sebuah bidang interdisipliner yang menggabungkan metode ilmiah, proses, algoritma, dan sistem untuk mengekstrak pengetahuan atau wawasan dari data dalam berbagai bentuk, baik terstruktur maupun tidak terstruktur. Secara esensial, Data Science berada di persimpangan tiga pilar utama: Matematika dan Statistik, Keahlian Domain, dan Ilmu Komputer (Pemrograman).

Tujuan utama Data Science bukan sekadar mengumpulkan data, melainkan untuk mengubah data mentah menjadi informasi yang bernilai dan dapat digunakan untuk pengambilan keputusan yang lebih baik (data-driven decision making). Prosesnya sering kali melibatkan serangkaian tahapan yang dikenal sebagai Siklus Hidup Data Science, yang umumnya meliputi:

1. **Pengumpulan Data:** Mengidentifikasi sumber data yang relevan.
2. **Pemrosesan Awal (Preprocessing):** Membersihkan, mentransformasi, dan menyiapkan data (dibahas lebih lanjut di Sub-bab 2.2).
3. **Eksplorasi Data (EDA):** Menganalisis set data untuk meringkas karakteristik utamanya, sering kali menggunakan metode visualisasi.
4. **Pemodelan:** Menerapkan algoritma machine learning atau statistik untuk menemukan pola, membuat prediksi, atau mengklasifikasikan data.
5. **Evaluasi dan Implementasi:** Menguji akurasi model dan menerapkannya dalam sistem yang sesungguhnya.
6. **Komunikasi Hasil:** Menyajikan temuan dan wawasan kepada *stakeholder* dengan cara yang mudah dimengerti.

2.2 Data Preprocessing

Data Preprocessing (Pra-pemrosesan Data) adalah tahapan krusial dan sering kali memakan waktu paling banyak dalam proyek Data Science. Ini adalah serangkaian teknik yang digunakan untuk mengubah data mentah menjadi format yang bersih, konsisten, dan siap untuk diolah oleh algoritma atau model analisis. Data mentah di dunia nyata sering kali memiliki masalah seperti data hilang, *noise* (gangguan/nilai ekstrem), dan format yang tidak seragam, yang jika tidak ditangani dapat mengurangi akurasi model secara signifikan.

Langkah-langkah utama dalam Data Preprocessing meliputi:

1. **Pembersihan Data (Data Cleaning):**
 - **Menangani Nilai Hilang (Missing Values):** Mengisi nilai yang kosong (imputasi) menggunakan rata-rata, median, modus, atau bahkan model prediksi, atau menghapus baris/kolom yang memiliki terlalu banyak data hilang.
 - **Menghilangkan Noise:** Mengidentifikasi dan menghaluskan data yang mengandung *noise* atau *outlier* (pencilan) yang dapat mengganggu analisis.

2. **Integrasi Data (Data Integration):** Menggabungkan data dari berbagai sumber yang berbeda (misalnya, database, *file CSV*, *web services*) menjadi satu set data yang kohesif.
3. **Transformasi Data (Data Transformation):** Mengubah format data agar sesuai dengan kebutuhan analisis. Ini termasuk:
 - Normalisasi/Standardisasi: Mengubah skala data numerik ke dalam rentang tertentu agar semua fitur memiliki bobot yang setara.
 - Agregasi Data: Meringkas data (misalnya, menghitung total penjualan per bulan).
 - Encoding Variabel Kategorikal: Mengubah variabel non-numerik (misalnya, jenis kelamin, warna) menjadi bentuk numerik yang dapat dipahami oleh algoritma (misalnya, *One-Hot Encoding*).
4. **Reduksi Data (Data Reduction):** Mengurangi volume data namun tetap mempertahankan integritas informasi. Tujuannya adalah mempercepat proses komputasi. Teknik yang digunakan bisa berupa reduksi dimensi (seperti PCA) atau kompresi data

2.3 Visualisasi Data

Visualisasi Data adalah representasi grafis dari informasi dan data. Dengan menggunakan elemen visual seperti diagram, grafik, dan peta, alat visualisasi data menyediakan cara yang mudah diakses untuk melihat dan memahami *trend*, *outlier*, dan pola dalam data. Bagi manusia, lebih mudah mencerna informasi visual daripada membaca deretan angka yang panjang.

Tujuan Visualisasi Data:

1. Eksplorasi Data Awal (EDA): Sebelum analisis mendalam, visualisasi membantu *data scientist* untuk menemukan anomali, distribusi data, dan hubungan antar variabel.
2. Komunikasi Temuan: Menyampaikan hasil analisis yang kompleks kepada audiens non-teknis dengan cara yang jelas, ringkas, dan persuasif.
3. Pemantauan Real-time: Memungkinkan pemantauan kinerja atau proses bisnis melalui *dashboard* interaktif.
4. Identifikasi Pola dan Tren: Memudahkan identifikasi pola musiman, pergeseran tren, atau *outlier* yang tersembunyi.

Jenis-jenis Visualisasi Umum:

- Histogram: Menunjukkan distribusi frekuensi data numerik.
- Diagram Batang (*Bar Chart*): Membandingkan data kategorikal.
- Diagram Garis (*Line Chart*): Ideal untuk menunjukkan perubahan data sepanjang waktu (*time-series*).
- Diagram Sebar (*Scatter Plot*): Menggambarkan hubungan atau korelasi antara dua variabel numerik.
- Diagram Lingkaran (*Pie Chart*): Menunjukkan proporsi bagian terhadap keseluruhan.

2.4 Konsep Analisis Statistik

Analisis Statistik adalah ilmu yang berkaitan dengan pengumpulan, analisis, interpretasi, dan presentasi data. Dalam konteks Data Science, Statistik menyediakan kerangka matematis yang ketat untuk menguji hipotesis, mengukur ketidakpastian, dan menyimpulkan hasil dari sampel ke populasi yang lebih besar.

Dua cabang utama Analisis Statistik yang digunakan dalam penelitian ini adalah:

1. Statistik Deskriptif:

- Fokus pada peringkasan dan penggambaran fitur utama dari data yang ada.
- Menggunakan ukuran pemusatan data (seperti Rata-rata/Mean, Median, dan Modus) untuk mengidentifikasi nilai tipikal.
- Menggunakan ukuran penyebaran data (seperti Standar Deviasi, Varians, dan Jangkauan/Range) untuk mengukur seberapa jauh data tersebar dari nilai tengah.
- Memberikan pemahaman awal mengenai karakteristik set data tanpa membuat kesimpulan yang digeneralisasi.

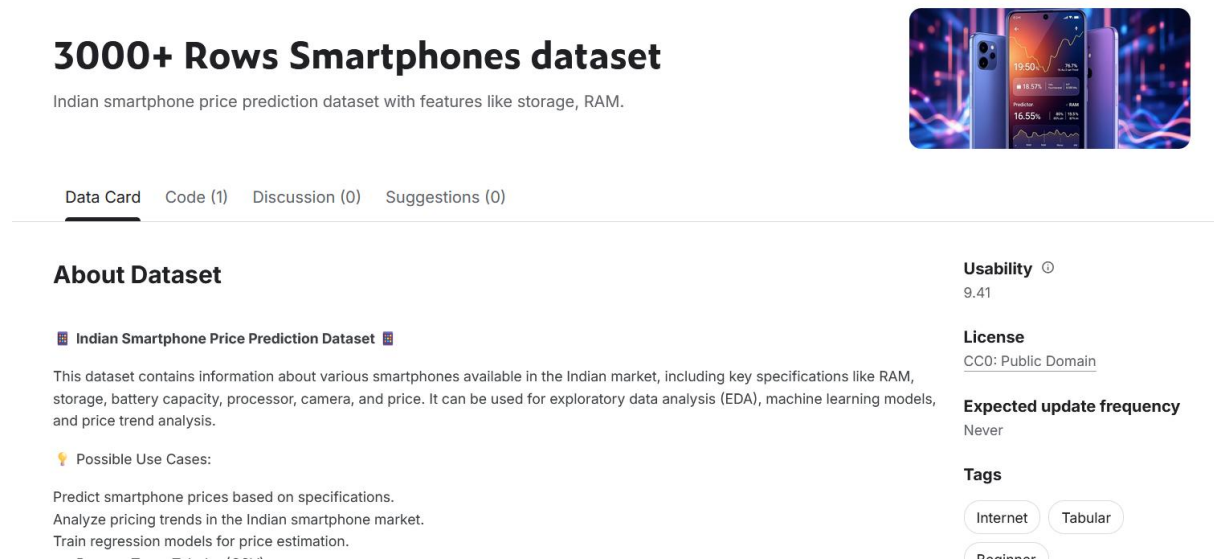
2. Statistik Inferensial:

- Fokus pada pengambilan kesimpulan atau membuat prediksi tentang suatu populasi berdasarkan sampel data.
- Menggunakan teknik seperti Uji Hipotesis (misalnya, Uji-t, ANOVA) untuk menentukan apakah perbedaan yang diamati antar kelompok signifikan secara statistik.
- Menggunakan Analisis Regresi untuk memodelkan hubungan antara satu atau lebih variabel independen (prediktor) dan variabel dependen (target). Regresi memungkinkan prediksi nilai di masa depan atau estimasi dampak satu variabel terhadap variabel lainnya.
- Hasil dari statistik inferensial selalu disertai dengan tingkat ketidakpastian (P-value atau *Confidence Interval*), yang menjadi ciri khas validitas ilmiah dari kesimpulan yang dibuat.

BAB III Metode Penelitian

3.1. Tahapan Analysis Data

3.1.1 Dataset




Dataset yang kelompok kami gunakan berisi data spesifikasi berbagai smartphone dari berbagai merek dan sistem operasi. Total terdapat 3.260 baris data dengan 20 atribut yang mencakup informasi seperti merek, nama produk, harga, RAM, penyimpanan, kapasitas baterai, jenis prosesor, dan fitur-fitur tambahan seperti fast charging, NFC, dan 5G.

Data ini dikumpulkan dalam format CSV (Comma Separated Values) sehingga mudah untuk diolah menggunakan bahasa pemrograman Python. Dataset ini sangat cocok digunakan untuk proyek *data science* karena memiliki kombinasi data numerik dan kategorikal, sehingga bisa dianalisis dari berbagai sisi seperti tren harga, hubungan antarspesifikasi, dan perbandingan antar merek.

Beberapa atribut penting dalam dataset ini antara lain:

- Price – menunjukkan harga smartphone.
- RAM & Storage – menggambarkan kapasitas memori dan penyimpanan.
- Battery_cap – menunjukkan daya tahan baterai.
- Processor_brand – menunjukkan jenis prosesor yang digunakan.
- has_5g, has_nfc, has_fast_charging – menggambarkan fitur-fitur modern yang tersedia.


```

m6m0uL n29B6: 29a'2+ KB
qfLb62: tJ9fey(λ)' iufey(v)' opJecf(a)
T0 qI2bJ9λ flb62 3300 wou-untJ opJecf
T8 L6tL62u L9f6(μx) J23a wou-untJ tJ9fey
T1 qI2bJ9λ 2I36(Jucμ) 3300 wou-untJ tJ9fey
T0 unu flouf c9w6L9 3300 wou-untJ iufey
T2 bLJw6Lλ flouf c9w6L9 3300 wou-untJ tJ9fey
T4 unu 969L c9w6L92 3300 wou-untJ iufey
T3 bLJw6Lλ L69L c9w6L9 3300 wou-untJ tJ9fey
T5 unu c0L6 3082 wou-untJ tJ9fey
T1 bL0C620L pr9uq 3300 wou-untJ opJecf
T0 μ92_2B 5234 wou-untJ opJecf
0 μ92 ufc 5234 wou-untJ opJecf
8 μ92 tJ9feybLJuf2 5234 wou-untJ opJecf
λ μ92 t92f cμ9L9Juf 3300 wou-untJ opJecf
e 9afL6L c9b 3300 wou-untJ iufey
2 2f0L9B6 3300 wou-untJ tJ9fey
4 02 3300 wou-untJ opJecf
3 9wW 3300 wou-untJ tJ9fey
5 bLJc6 3300 wou-untJ iufey
T n9w6 3300 wou-untJ opJecf
0 pr9uq n9w6 3300 wou-untJ opJecf
---
# c0Jnmu wou-untJ c0nu fclb6
p9f9 c0Jnmu2 (f0f9J 30 c0Jnmu2):
9u96iufex: 3300 6ufL62' 0 f0 3320
<CJ922 'b9u92.c0L6'fL9w6'p9f9fL9w6,>
 Iuf0 q9f92ef:

```

```
df = pd.read_csv("smartphones_data.csv")
df.head()
```

	brand_name	Name	Price	RAM	OS	storage	Battery_cap	has_fast_charging	has_fingerprints	has_nfc	has_5g	processor_brand	num_core	primery_rear_camera	Num_Re
0	vivo	vivo v50	34999	8.0	android	128.0	6000	Yes	Yes	No	Yes	snapdragon	8.0	50.0	
1	realme	realme p3 pro	21999	8.0	android	128.0	6000	Yes	Yes	No	Yes	snapdragon	8.0	50.0	
2	realme	realme 14 pro plus	27999	8.0	android	128.0	6000	Yes	Yes	No	Yes	snapdragon	8.0	50.0	
3	samsung	samsung galaxy s25 ultra	129999	12.0	android	256.0	5000	Yes	Yes	Yes	Yes	snapdragon	8.0	200.0	
4	vivo	vivo t3 pro	22999	8.0	android	128.0	5500	Yes	Yes	No	Yes	snapdragon	8.0	50.0	

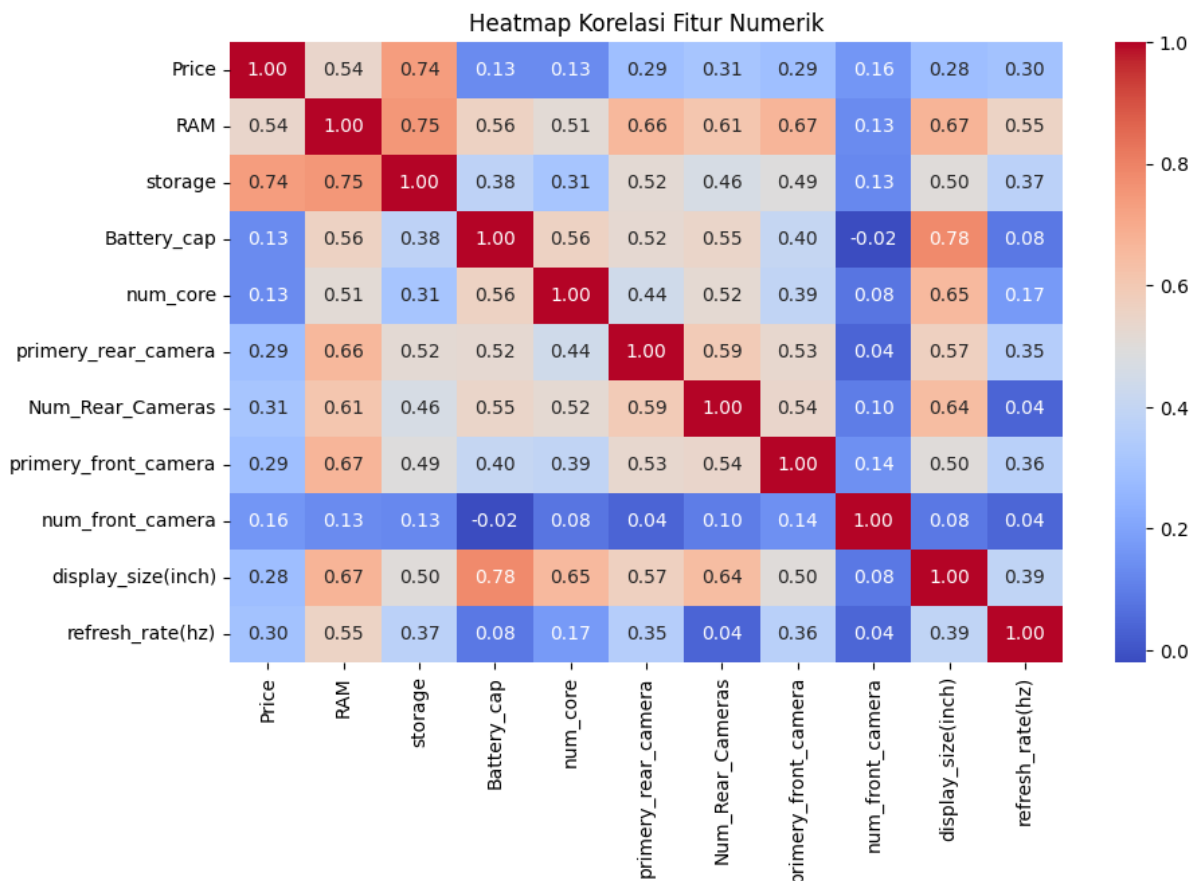
```
print(df.columns)
```

```
Index(['brand_name', 'Name', 'Price', 'RAM', 'OS', 'storage', 'Battery_cap',
      'has_fast_charging', 'has_fingerprints', 'has_nfc', 'has_5g',
      'processor_brand', 'num_core', 'primery_rear_camera',
      'Num_Rear_Cameras', 'primery_front_camera', 'num_front_camera',
      'display_size(inch)', 'refresh_rate(hz)', 'display_types'],
      dtype='object')
```

Hasil output menunjukkan bahwa dataset memiliki kolom sebagai berikut: brand_name, Name, Price, RAM, OS, storage, Battery_cap, has_fast_charging, has_fingerprints, has_nfc, has_5g, processor_brand, num_core, primery_rear_camera, Num_Rear_Cameras, primery_front_camera, num_front_camera, display_size(inch), refresh_rate(hz), dan display_types.

Kolom-kolom ini merepresentasikan kombinasi data numerik, kategorikal, dan boolean yang menggambarkan karakteristik setiap smartphone.

3.1.2.2 Visualization Data Mentah



Visualisasi ini, yang menampilkan frekuensi kemunculan berbagai jenis layar pada *smartphone*, berfungsi sebagai langkah awal (EDA) untuk memahami struktur data kategorikal sebelum dilakukan *preprocessing* dan analisis inferensial.

1. Struktur Data Mentah yang Terlihat

Visualisasi ini secara langsung menampilkan jumlah (frekuensi absolut) dari setiap kategori dalam kolom *display_types* pada *dataset* mentah:

- Mayoritas Data (Modus): Jenis layar AMOLED adalah kategori yang paling sering muncul (*modus*). Ini memberikan indikasi awal bahwa, meskipun data belum diolah, teknologi AMOLED adalah yang paling dominan di pasar produk yang dicakup oleh *dataset*.
- Kategori Signifikan Kedua: Jenis layar LCD IPS adalah kategori yang paling banyak kedua. Ini menunjukkan bahwa meskipun AMOLED memimpin, LCD masih memegang porsi yang signifikan.
- Kategori Minor (Data Langka): Kategori seperti *TFT*, *OLED*, *Retina*, dan *Fluid* memiliki frekuensi yang sangat rendah. Kategori-kategori ini perlu diwaspadai dalam tahap *preprocessing* karena jumlahnya yang sedikit dapat menyebabkan masalah ketidakseimbangan kelas (*class imbalance*) jika data ini digunakan untuk klasifikasi.

2. Implikasi Terhadap Tahap Preprocessing

Sebagai data mentah, visualisasi ini memberi petunjuk kritis tentang langkah *preprocessing* yang harus diambil selanjutnya:

Temuan Data Mentah Tindakan Preprocessing yang Diperlukan

Banyak Kategori dengan Frekuensi

Rendah (*OLED, Retina, TFT, Fluid*)

Perlu dilakukan Penggabungan Kategori (*Feature Binning*). Misalnya, semua jenis layar yang langka atau merupakan varian dari OLED/LCD dapat digabungkan menjadi satu kategori "Lainnya" untuk mengurangi kompleksitas dan menghindari masalah ketidakseimbangan kelas saat *Encoding*.

Data Kategorikal

Nominal (Nama jenis layar)

Perlu dilakukan Encoding Variabel, seperti One-Hot Encoding, sebelum dimasukkan ke dalam model regresi. Alasannya, model matematis tidak dapat memproses nilai "AMOLED" atau "LCD IPS," melainkan membutuhkan representasi biner (0 atau 1) untuk setiap jenis layar.

Potensi Variabel

Penting (AMOLED vs. LCD)

Setelah *Encoding*, kedua kategori dominan (AMOLED dan LCD IPS) akan menjadi variabel biner penting yang harus diuji korelasi atau dimasukkan ke dalam model regresi untuk melihat apakah jenis layar secara signifikan memengaruhi harga.

3.1.3. Data Preprocessing (Advance Preprocessing)

3.1.3.1. Data Cleaning

```
# Cek ulang jumlah missing value per kolom
missing_values = df.isnull().sum()
print("Missing values per kolom:\n", missing_values[missing_values > 0])
```

Missing values per kolom:

has_fingerprints	726
has_nfc	726
has_5g	726
num_core	175
refresh_rate(hz)	1731

dtype: int64

Sebelum di proses data harus dilakukan Cleaning guna mengisi nilai yang hilang, smooth noisy data, mengidentifikasi atau menghapus outlier, dan mengatasi inkonsistensi

```
[16] # Are there any missing values
      df.isnull().sum()
```

brand_name	0	primery_rear_camera	0
Name	0	Num_Rear_Cameras	0
Price	0	primery_front_camera	0
RAM	0	num_front_camera	0
os	0	display_size(inch)	0
storage	0	refresh_rate(hz)	1731
Battery_cap	0	display_types	0
has_fast_charging	0		
has_fingerprints	726		
has_nfc	726		
has_5g	726		
processor_brand	0		
num_core	175		

dtype: int64

Hasil pemeriksaan menunjukkan bahwa sebagian besar kolom tidak memiliki nilai kosong, namun terdapat beberapa kolom dengan *missing values*, yaitu:

- has_fingerprints sebanyak 726 nilai hilang
- has_nfc sebanyak 726 nilai hilang
- has_5g sebanyak 726 nilai hilang
- num_core sebanyak 175 nilai hilang

- refresh_rate(hz) sebanyak 1731 nilai hilang

Ini menunjukkan bahwa sebagian atribut fitur memiliki data yang tidak lengkap, terutama pada kolom refresh_rate(hz) yang memiliki jumlah nilai kosong terbesar.

[17]
✓ 0 d

```
# Persentase missing value
(df.isnull().mean() * 100).round(2)
```

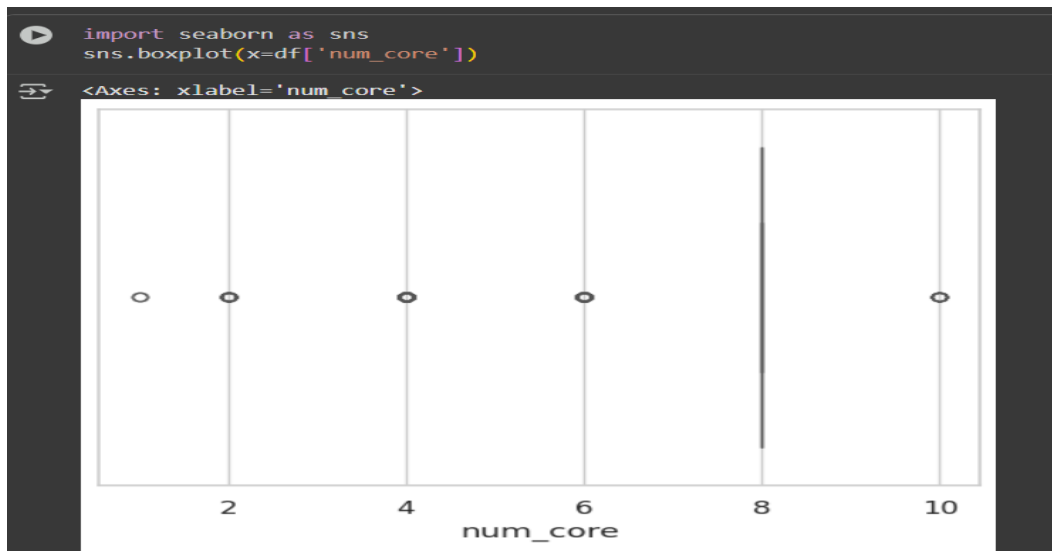
	0
brand_name	0.00
Name	0.00
Price	0.00
RAM	0.00
OS	0.00
storage	0.00
Battery_cap	0.00
has_fast_charging	0.00
has_fingerprints	22.27
has_nfc	22.27
has_5g	22.27
processor_brand	0.00
num_core	5.37
primary_rear_camera	0.00
Num_Rear_Cameras	0.00
primary_front_camera	0.00
num_front_camera	0.00
display_size(inch)	0.00
refresh_rate(hz)	53.10
display_types	0.00

dtype: float64

Hasilnya menunjukkan bahwa:

- Kolom has_fingerprints, has_nfc, dan has_5g masing-masing memiliki 22.27% nilai hilang.
- Kolom num_core memiliki 5.37% nilai hilang.
- Kolom refresh_rate(hz) memiliki 53.10% nilai hilang, yang merupakan persentase tertinggi di antara semua atribut.

Persentase ini mengindikasikan bahwa sebagian besar data masih dalam kondisi baik, namun beberapa fitur seperti refresh_rate(hz) memiliki jumlah data kosong yang cukup besar sehingga perlu dipertimbangkan untuk dihapus atau diisi dengan metode tertentu (misalnya *mean* atau *mode imputation*).



kami melakukan penanganan pada num_core yang total missing valuenya itu dibawah 5%

Hilang < 5% = Isi dengan mean/modus/median

kenapa kami milih modus karena kami sudah cek datany tidak ada outlier (bila ada outlier kami akan pakai median)

- Mean bisa menghasilkan pecahan (tidak logis untuk jumlah core)
- Median akan sama dengan mode (karena 8 paling banyak)
- Modus menjaga data tetap realistis sesuai domain (octa-core paling umum)

```
df['num_core'] = df['num_core'].fillna(df['num_core'].mode()[0])
```

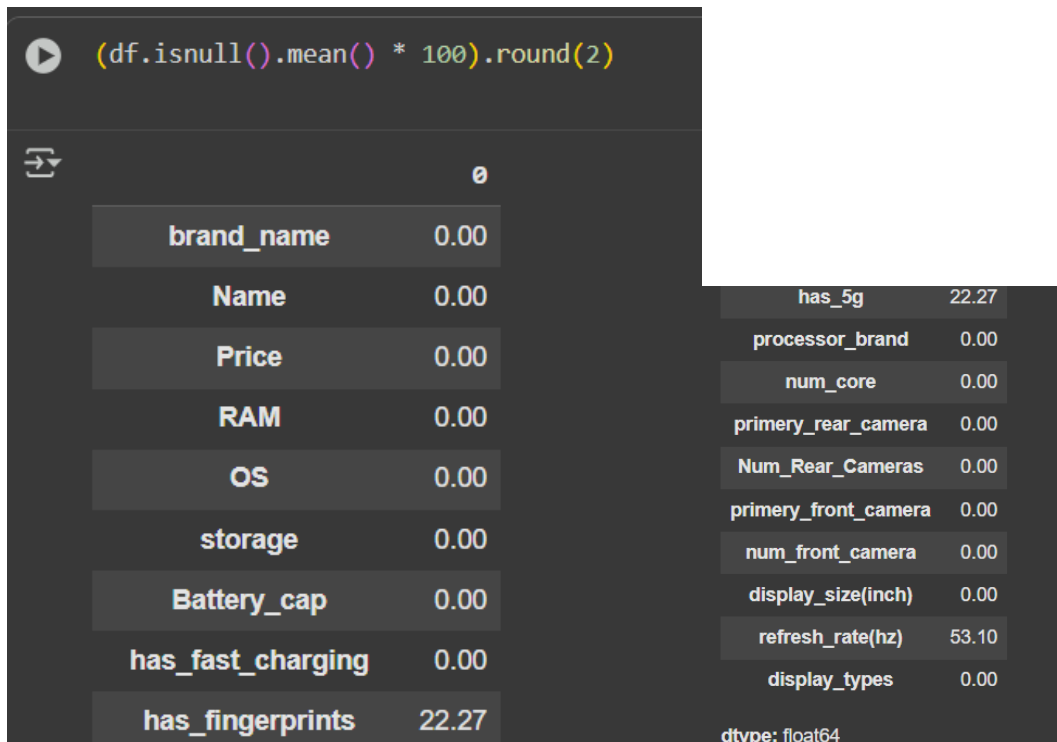
	brand_name	Name	Price	RAM	OS	storage	Battery_cap	has_fast_charging	has_fingerprints	has_nfc	has_5g	processor_brand	num_core	primary_rear_camera	Num_Rear_...
0	vivo	vivo v50	34999	8.0	android	128.0	6000	Yes	Yes	No	Yes	snapdragon	8.0	50.0	
1	realme	realme p3 pro	21999	8.0	android	128.0	6000	Yes	Yes	No	Yes	snapdragon	8.0	50.0	
2	realme	realme 14 pro plus	27999	8.0	android	128.0	6000	Yes	Yes	No	Yes	snapdragon	8.0	50.0	
3	samsung	samsung galaxy s25 ultra	129999	12.0	android	256.0	5000	Yes	Yes	Yes	Yes	snapdragon	8.0	200.0	
4	vivo	vivo t3 pro	22999	8.0	android	128.0	5500	Yes	Yes	No	Yes	snapdragon	8.0	50.0	
...
3255	ikall	ikall k570	4799	4.0	android	64.0	4000	No	Yes	No	No	tru-mediatek	8.0	13.0	
3256	Other	mafe z2	5299	2.0	android	16.0	3300	No	Yes	No	No	quad	4.0	5.0	
3257	Other	mafe v9	5599	2.0	android	16.0	3500	No	NaN	NaN	NaN	quad	4.0	5.0	
3258	Other	marq m3 smart	6499	2.0	android	32.0	5000	No	NaN	NaN	NaN	tru-mediatek	8.0	13.0	

Pada tahap pembersihan data, dilakukan proses untuk mengatasi nilai kosong (*missing values*) pada kolom num_core. Kolom ini berisi informasi mengenai jumlah inti prosesor dari setiap smartphone. Karena adanya data yang kosong bisa memengaruhi hasil analisis, maka dilakukan pengisian nilai agar dataset menjadi lebih lengkap dan tidak ada data yang hilang.

Nilai kosong pada kolom tersebut diisi menggunakan modus, yaitu nilai yang paling sering muncul di seluruh data. Cara ini dipilih karena kolom num_core berisi data numerik yang

bersifat diskrit (seperti 2, 4, 6, atau 8 core), sehingga penggunaan modus dianggap paling cocok untuk menggambarkan nilai yang umum pada dataset.

Setelah proses ini dilakukan, kolom *num_core* tidak lagi memiliki nilai kosong. Hasil akhirnya menunjukkan bahwa sebagian besar smartphone memiliki 8 core, sehingga nilai tersebut juga digunakan sebagai pengganti untuk data yang sebelumnya kosong.



```
(df.isnull().mean() * 100).round(2)
```

	0
brand_name	0.00
Name	0.00
Price	0.00
RAM	0.00
OS	0.00
storage	0.00
Battery_cap	0.00
has_fast_charging	0.00
has_fingerprints	22.27
has_5g	22.27
processor_brand	0.00
num_core	0.00
primery_rear_camera	0.00
Num_Rear_Cameras	0.00
primery_front_camera	0.00
num_front_camera	0.00
display_size(inch)	0.00
refresh_rate(hz)	53.10
display_types	0.00

dtype: float64

Pada tahap awal pembersihan data, dilakukan pengecekan terhadap keberadaan nilai kosong pada setiap kolom dalam dataset. Proses ini bertujuan untuk mengetahui seberapa besar persentase data yang hilang sehingga dapat ditentukan langkah penanganan yang tepat.

Berdasarkan hasil pengecekan, sebagian besar kolom memiliki persentase nilai kosong sebesar 0%, yang berarti datanya lengkap. Namun, terdapat beberapa kolom yang masih memiliki nilai kosong, seperti:

- **has_fingerprints** sebesar 22.27%,
- **has_5g** sebesar 22.27%, dan
- **refresh_rate(hz)** sebesar 53.10%.

Nilai-nilai tersebut menunjukkan bahwa ketiga kolom tersebut memiliki data yang belum lengkap dan perlu dilakukan penanganan lebih lanjut, seperti pengisian nilai dengan metode tertentu (misalnya modus atau median) atau penghapusan data jika dianggap tidak relevan.


```
df[['has_fingerprints', 'has_nfc', 'has_5g']].isna().mean() * 100
```

	0
has_fingerprints	22.269939
has_nfc	22.269939
has_5g	22.269939

dtype: float64

Ditampilkan hasil analisis *missing values* atau data yang kosong pada tiga kolom, yaitu `has_fingerprints`, `has_nfc`, dan `has_5g`. Kode `df[['has_fingerprints', 'has_nfc', 'has_5g']].isna().mean() * 100` digunakan untuk menghitung persentase nilai kosong (NaN) pada setiap kolom dalam DataFrame. Fungsi `.isna()` berfungsi untuk mendeteksi data yang hilang, `.mean()` menghitung rata-rata jumlah data yang hilang pada tiap kolom, dan hasilnya dikalikan 100 agar diperoleh dalam bentuk persentase. Berdasarkan hasil yang muncul, ketiga kolom tersebut memiliki persentase *missing value* yang sama yaitu sebesar 22,27%. Artinya, sekitar seperlima dari data pada kolom tersebut tidak memiliki nilai atau informasi yang lengkap. Hal ini penting diperhatikan karena dapat memengaruhi hasil analisis fitur smartphone yang berkaitan dengan keberadaan sensor sidik jari, dukungan NFC, dan konektivitas 5G.

```
for col in ['has_fingerprints', 'has_nfc', 'has_5g']:
    print(f"\n{col} value counts:")
    print(df[col].value_counts(normalize=True) * 100)
```

```
has_fingerprints value counts:
has_fingerprints
Yes    94.356748
No      5.643252
Name: proportion, dtype: float64

has_nfc value counts:
has_nfc
No    67.048145
Yes   32.951855
Name: proportion, dtype: float64

has_5g value counts:
has_5g
No    60.418311
Yes   39.581689
Name: proportion, dtype: float64
```

has_nfc kami isi dengan YES karena sangat logis kalau nilai kosong diisi dengan "Yes", karena probabilitas besar memang begitu.

Analisis dilakukan untuk memahami proporsi nilai pada tiga fitur biner, yaitu `has_fingerprints`, `has_nfc`, dan `has_5g`. Berdasarkan hasil perhitungan proporsi menggunakan fungsi `value_counts(normalize=True)`, diketahui bahwa fitur `has_fingerprints` memiliki nilai “Yes” sebesar 94,36% dan “No” sebesar 5,64%. Hasil ini menunjukkan bahwa hampir seluruh perangkat dalam dataset telah dilengkapi dengan sensor sidik jari.

Selanjutnya, pada fitur *has_nfc*, diperoleh distribusi nilai “No” sebesar 67,05% dan “Yes” sebesar 32,95%. Hal ini menandakan bahwa hanya sekitar sepertiga perangkat yang mendukung teknologi NFC, sedangkan sebagian besar belum memilikinya. Adapun fitur *has_5g* menunjukkan nilai “No” sebesar 60,42% dan “Yes” sebesar 39,58%, yang berarti sekitar empat dari sepuluh perangkat dalam dataset sudah mendukung jaringan 5G.

Berdasarkan distribusi tersebut, dilakukan penyesuaian terhadap nilai kosong (*missing values*) pada kolom *has_nfc* dengan cara mengisinya menggunakan kategori “Yes”. Keputusan ini didasarkan pada pertimbangan logis bahwa perangkat modern umumnya sudah banyak yang dilengkapi dengan fitur NFC, sehingga secara kontekstual lebih masuk akal apabila data yang hilang diasumsikan bernilai “Yes”

```
# Encode kolom Yes/No menjadi 1/0
cols_to_encode = ['has_fingerprints', 'has_nfc', 'has_5g']
df[cols_to_encode] = df[cols_to_encode].replace({'Yes': 1, 'No': 0})
df = df.infer_objects(copy=False)
df[cols_to_encode].head()
```

/tmp/ipython-input-2747241906.py:3: FutureWarning: Downcasting behavior in `replace`
df[cols_to_encode] = df[cols_to_encode].replace({'Yes': 1, 'No': 0})

	has_fingerprints	has_nfc	has_5g
0	1	0.0	1.0
1	1	0.0	1.0
2	1	0.0	1.0
3	1	1.0	1.0
4	1	0.0	1.0

hasil analisis terhadap data yang hilang (*missing values*) pada tiga kolom dalam dataset, yaitu *has_fingerprints*, *has_nfc*, dan *has_5g*. Potongan kode `df[['has_fingerprints', 'has_nfc', 'has_5g']].isna().mean() * 100` digunakan untuk menghitung persentase nilai kosong pada masing-masing kolom tersebut. Langkah pertama, fungsi `.isna()` akan memeriksa setiap baris data pada kolom yang dipilih dan memberikan nilai *True* jika terdapat data yang kosong (NaN), serta *False* jika nilainya ada. Kemudian, fungsi `.mean()` menghitung rata-rata dari nilai *True* tersebut. Karena dalam Python nilai *True* dihitung sebagai 1 dan *False* sebagai 0, maka hasil rata-rata ini menggambarkan proporsi data kosong di setiap kolom. Selanjutnya, hasil tersebut dikalikan dengan 100 agar ditampilkan dalam bentuk persentase sehingga lebih mudah dipahami.

```

corr_matrix = df.corr(numeric_only=True)

# Korelasi terhadap has_5g
print("Korelasi terhadap has_5g:")
print(corr_matrix['has_5g'].sort_values(ascending=False))

# Korelasi terhadap has_nfc
print("\nKorelasi terhadap has_nfc:")
print(corr_matrix['has_nfc'].sort_values(ascending=False))

```

Korelasi terhadap has_5g:

has_5g	1.000000
RAM	0.641724
refresh_rate(hz)	0.620941
storage	0.531113
primery_rear_camera	0.520036
display_size(inch)	0.438399
primery_front_camera	0.392582
Price	0.390545
has_nfc	0.332195
Battery_cap	0.279225
Num_Rear_Cameras	0.259996
num_core	0.221703
num_front_camera	-0.010136
has_fingerprints	-0.039861

Name: has_5g, dtype: float64

Tahap ini dilakukan untuk menganalisis hubungan antara variabel numerik dengan fitur biner *has_5g* dan *has_nfc* menggunakan matriks korelasi (*correlation matrix*). Hasil perhitungan menunjukkan bahwa fitur *has_5g* memiliki korelasi tertinggi dengan RAM (0,64), *refresh_rate(hz)* (0,62), dan *storage* (0,53). Hal ini mengindikasikan bahwa perangkat dengan dukungan 5G umumnya memiliki spesifikasi yang lebih tinggi, terutama dari sisi kapasitas RAM, kecepatan layar, dan penyimpanan internal. Korelasi positif yang kuat juga terlihat pada fitur kamera belakang dan ukuran layar, menunjukkan bahwa perangkat berteknologi 5G cenderung berada di kelas menengah ke atas.

```

cond_5g = (df['RAM'] >= df['RAM'].median()) & (df['refresh_rate(hz)'] >= df['refresh_rate(hz)'].median())
df.loc[cond_5g & (df['has_5g'].isna()), 'has_5g'] = 1
df['has_5g'] = df['has_5g'].fillna(0)

cond_nfc = (df['Price'] >= df['Price'].median()) & (df['storage'] >= df['storage'].median())
df.loc[cond_nfc & (df['has_nfc'].isna()), 'has_nfc'] = 1
df['has_nfc'] = df['has_nfc'].fillna(0)

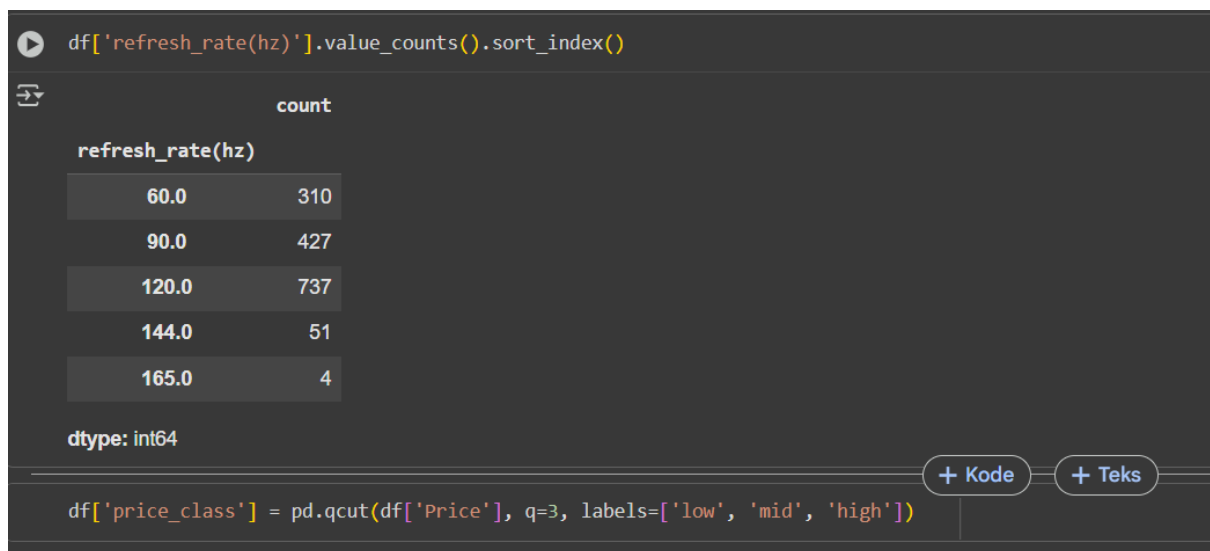
```

Proses Imputasi Nilai Kosong Berdasarkan Kondisi Logis

Tahap ini bertujuan untuk mengisi nilai kosong (*missing values*) pada kolom *has_5g* dan *has_nfc* dengan pendekatan berbasis kondisi logis (*conditional imputation*). Untuk fitur *has_5g*, nilai kosong diisi dengan 1 (Yes) apabila perangkat memiliki nilai RAM dan *refresh_rate(hz)* yang lebih besar atau sama dengan nilai median masing-masing kolom. Hal ini didasarkan pada hasil analisis korelasi sebelumnya yang menunjukkan bahwa perangkat dengan dukungan 5G umumnya memiliki kapasitas RAM tinggi dan kecepatan layar yang lebih besar. Nilai selain kondisi tersebut diisi dengan 0 (No).

Selanjutnya, untuk fitur *has_nfc*, imputasi dilakukan dengan prinsip serupa, yaitu memberikan nilai 1 (Yes) pada perangkat yang memiliki harga (*Price*) dan kapasitas

penyimpanan (storage) di atas atau sama dengan nilai median. Pendekatan ini merefleksikan bahwa fitur NFC cenderung terdapat pada perangkat dengan spesifikasi dan harga yang lebih tinggi.



distribusi refresh rate layar serta **pengelompokan harga ponsel**. Dari hasil perintah `value_counts()` terhadap kolom `refresh_rate(hz)`, dapat dilihat bahwa sebagian besar perangkat memiliki refresh rate sebesar 120 Hz (737 unit), diikuti oleh 90 Hz (427 unit) dan 60 Hz (310 unit). Sementara itu, refresh rate tinggi seperti 144 Hz dan 165 Hz hanya dimiliki oleh sebagian kecil perangkat (masing-masing 51 dan 4 unit). Pola ini menunjukkan bahwa mayoritas ponsel di dataset berada pada kategori menengah ke atas, dengan refresh rate tinggi yang umum ditemukan pada perangkat gaming atau flagship masih tergolong jarang.

Selanjutnya, dilakukan pembuatan kolom baru bernama `price_class` dengan memanfaatkan fungsi `pd.qcut()`. Kolom ini mengelompokkan harga (`Price`) ke dalam tiga kelas, yaitu `low`, `mid`, dan `high`, berdasarkan pembagian kuantil ($q=3$). Dengan cara ini, setiap kategori harga mencakup sepertiga dari seluruh data, sehingga memudahkan analisis perbandingan spesifikasi atau fitur berdasarkan kelas harga. Langkah ini juga menjadi dasar penting untuk melakukan analisis lanjutan seperti melihat keterkaitan antara kelas harga dengan keberadaan fitur seperti 5G, NFC, atau kapasitas RAM dan storage.

```
1 df['refresh_rate(hz)'] = df.groupby('price_class', observed=True)['refresh_rate(hz)'].transform(
    lambda x: x.fillna(x.median())
)
```

Langkah selanjutnya dalam proses pengolahan data adalah menangani nilai yang hilang (missing values) pada kolom `refresh_rate(hz)`. Untuk mengatasi hal ini, digunakan metode pengisian berdasarkan median dari tiap kelompok harga yang telah dibentuk sebelumnya melalui kolom `price_class`. Proses ini dilakukan dengan perintah `groupby()` yang mengelompokkan data sesuai kategori harga (`low`, `mid`, `high`), kemudian fungsi `transform()` digunakan untuk mengganti nilai yang kosong dengan median refresh rate dari masing-masing kelompok tersebut.

Pendekatan ini dipilih karena lebih akurat dibandingkan menggunakan median secara keseluruhan. Hal ini disebabkan oleh adanya perbedaan karakteristik antara ponsel di tiap kelas harga. Misalnya, ponsel dengan kategori harga tinggi umumnya memiliki refresh rate yang lebih besar dibandingkan dengan ponsel di kelas menengah atau bawah. Dengan demikian, metode ini membantu menjaga konsistensi data serta memastikan bahwa pengisian nilai hilang tetap sesuai dengan tren alami dari setiap segmen harga.

```
pd.set_option('display.max_rows', None)
(df.isnull().mean() * 100).round(2)
```

	0
brand_name	0.0
Name	0.0
Price	0.0
RAM	0.0
OS	0.0
storage	0.0
Battery_cap	0.0
has_fast_charging	0.0
has_fingerprints	0.0
has_nfc	0.0

has_5g	0.0
processor_brand	0.0
num_core	0.0
primery_rear_camera	0.0
Num_Rear_Cameras	0.0
primery_front_camera	0.0
num_front_camera	0.0
display_size(inch)	0.0
refresh_rate(hz)	0.0
display_types	0.0
price_class	0.0

Langkah selanjutnya dalam proses pengolahan data adalah menangani nilai yang hilang (missing values) pada kolom `refresh_rate(hz)`. Untuk mengatasi hal ini, digunakan metode pengisian berdasarkan median dari tiap kelompok harga yang telah dibentuk sebelumnya melalui kolom `price_class`. Proses ini dilakukan dengan perintah `groupby()` yang mengelompokkan data sesuai kategori harga (low, mid, high), kemudian fungsi `transform()` digunakan untuk mengganti nilai yang kosong dengan median refresh rate dari masing-masing kelompok tersebut.

Pendekatan ini dipilih karena lebih akurat dibandingkan menggunakan median secara keseluruhan. Hal ini disebabkan oleh adanya perbedaan karakteristik antara ponsel di tiap kelas harga. Misalnya, ponsel dengan kategori harga tinggi umumnya memiliki refresh rate yang lebih besar dibandingkan dengan ponsel di kelas menengah atau bawah. Dengan demikian, metode ini membantu menjaga konsistensi data serta memastikan bahwa pengisian nilai hilang tetap sesuai dengan tren alami dari setiap segmen harga.

3.1.4.2 One-Hot Encoding pada Data Kategorikal

```
df = pd.get_dummies(df, columns=['brand_name', 'os', 'processor_brand', 'display_types'])

print(df.columns)

Index(['Name', 'Price', 'RAM', 'storage', 'Battery_cap', 'has_fast_charging',
       'has_fingerprints', 'has_nfc', 'has_5g', 'num_core',
       'primary_rear_camera', 'Num_Rear_Cameras', 'primary_front_camera',
       'num_front_camera', 'display_size(inch)', 'refresh_rate(hz)',
       'price_class', 'brand_name_other', 'brand_name_apple',
       'brand_name_asus', 'brand_name_coolpad', 'brand_name_pioneer',
       'brand_name_google', 'brand_name_honor', 'brand_name_htc',
       'brand_name_kall', 'brand_name_infinix', 'brand_name_intex',
       'brand_name_ipo', 'brand_name_itel', 'brand_name_karbonn',
       'brand_name_lava', 'brand_name_lenovo', 'brand_name_lg',
       'brand_name_lyf', 'brand_name_micromax', 'brand_name_moto',
       'brand_name_motorola', 'brand_name_nokia', 'brand_name_oneplus',
       'brand_name_oppo', 'brand_name_pantonic', 'brand_name_poco',
       'brand_name_realme', 'brand_name_samsung', 'brand_name_sony',
       'brand_name_tecno', 'brand_name_vivo', 'brand_name_xiaomi',
       'brand_name_xolo', 'os_android', 'os_ios', 'os_other',
       'processor_brand_apple', 'processor_brand_broadcom',
       'processor_brand_google', 'processor_brand_hisilicon',
       'processor_brand_huawei', 'processor_brand_intel',
       'processor_brand_mEDIATEK', 'processor_brand_nvidia',
       'processor_brand_qual', 'processor_brand_samsung',
       'processor_brand_snapdragon', 'processor_brand_spreadtrum',
       'processor_brand_st-ericsson', 'processor_brand_tru-mEDIATEK',
       'processor_brand_unisoc', 'display_types AMOLED display',
       'display_types_lcd display', 'display_types_oled display',
       'display_types_other display', 'display_types_tft display'],
      dtype='object')
```

```
# Cek distribusi data untuk menentukan metode penanganan outlier (normal atau tidak normal)

import pandas as pd

# Pilih hanya kolom numerik
numeric_cols = df.select_dtypes(include=['int64', 'float64']).columns

# Buat dictionary untuk menyimpan hasil jumlah outlier
outlier_counts = {}

for col in numeric_cols:
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR

    # Hitung jumlah outlier
    outliers = df[(df[col] < lower_bound) | (df[col] > upper_bound)]
    outlier_counts[col] = len(outliers)

# Tampilkan hasil dalam bentuk Dataframe biar rapi
outlier_summary = pd.DataFrame.from_dict(outlier_counts, orient='index', columns=['Jumlah Outlier'])
outlier_summary = outlier_summary.sort_values(by='Jumlah Outlier', ascending=False)

print(outlier_summary)
```

One-Hot Encoding digunakan untuk mengubah kolom kategorikal non-ordinal (yang tidak memiliki urutan logis, seperti brand atau sistem operasi) menjadi bentuk numerik biner (0/1) agar dapat diproses oleh model machine learning.

Contoh: “Samsung”, “Vivo”, “Oppo” → menjadi tiga kolom: brand_Samsung, brand_Vivo, brand_Oppo. Alasan pemilihan metode ini:

- Kolom seperti brand dan os tidak memiliki hierarki nilai (Samsung ≠ lebih besar dari Vivo).
- Jika menggunakan Label Encoding, model akan salah menafsirkan nilai label sebagai urutan, sehingga menyebabkan bias dan kesalahan interpretasi.
- Oleh karena itu, One-Hot Encoding lebih tepat untuk kategori tanpa urutan logis (nominal data).

```
Jumlah Outlier
num_core      753
Price         318
has_fingerprints 143
primary_rear_camera 135
storage       119
primary_front_camera 107
num_front_camera 88
RAM           36
Battery_cap   19
display_size(inch) 12
has_nfc       0
has_5g        0
Num_Rear_Cameras 0
refresh_rate(hz) 0
```

```
import matplotlib.pyplot as plt
import seaborn as sns

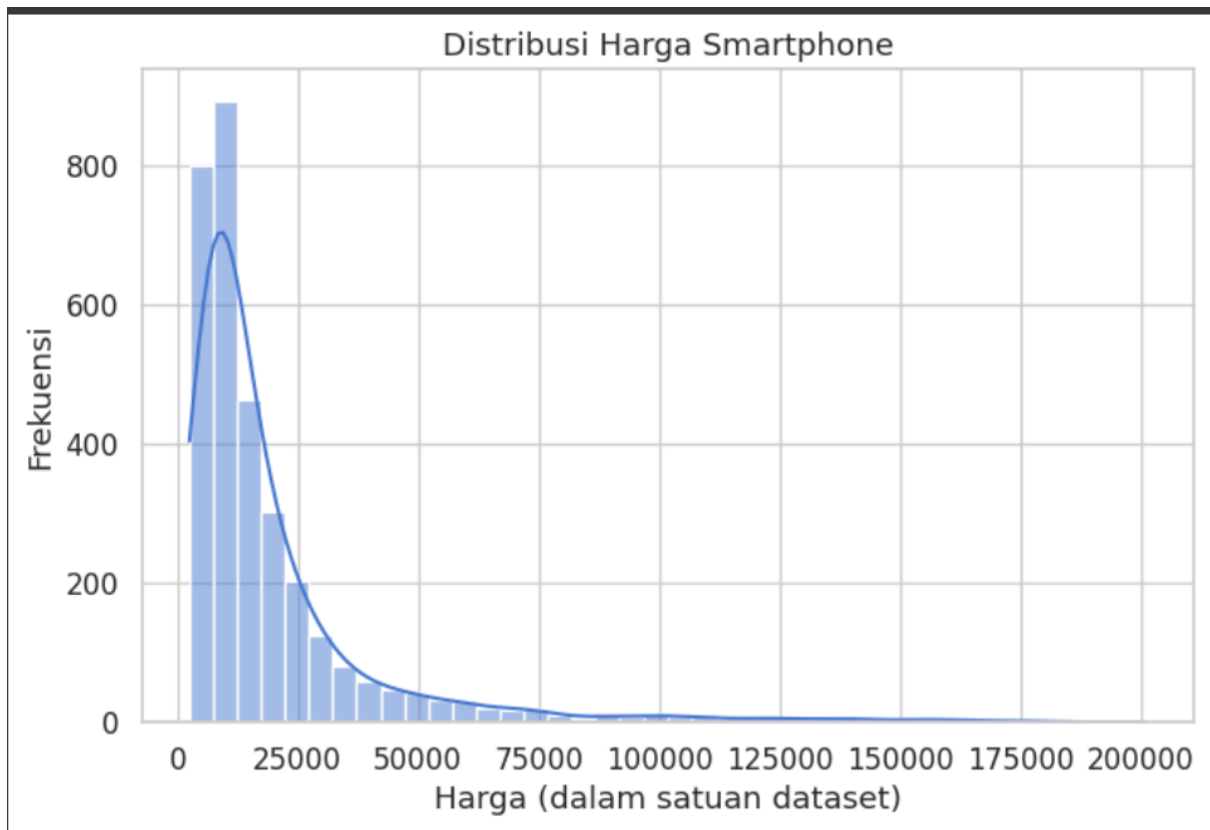
# Pilih kolom numerik utama
num_cols = [
    'Price', 'RAM', 'storage', 'Battery_cap', 'num_core',
    'primary_rear_camera', 'Num_Rear_Cameras',
    'primary_front_camera', 'num_front_camera',
    'display_size(inch)', 'refresh_rate(hz)'
]

# Plot boxplot untuk semua kolom numerik
plt.figure(figsize=(15, 10))
for i, col in enumerate(num_cols, 1):
    plt.subplot(4, 3, i)
```

Hasilnya dataset kini hanya berisi nilai numerik, di mana setiap kategori direpresentasikan dengan 0 dan 1. Data siap digunakan untuk visualisasi, analisis statistik, atau pelatihan model machine learning tanpa kehilangan makna kategorinya

3.1.4. Data Visualization

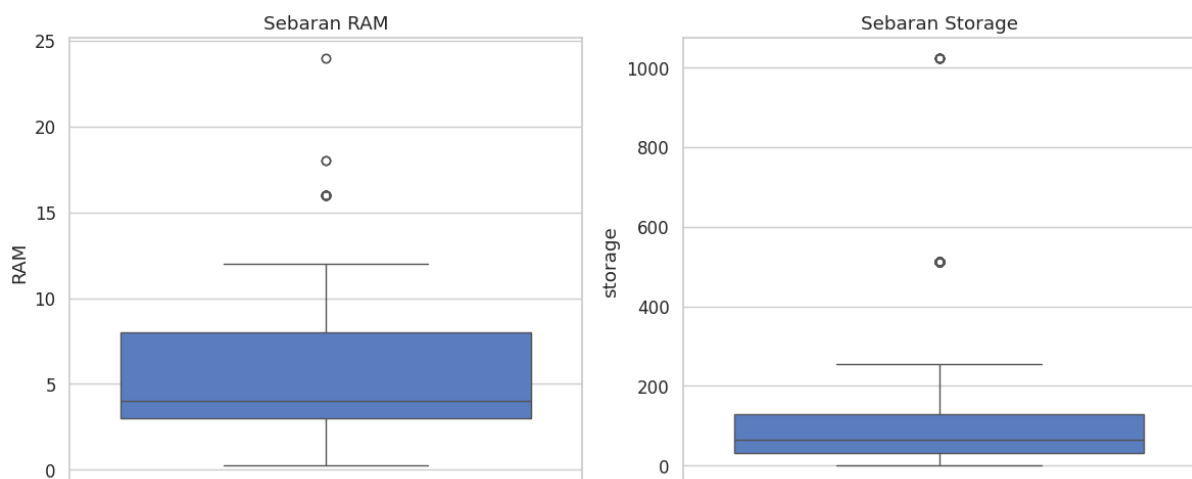
- Distribusi Harga Smartphone



Grafik di atas menunjukkan distribusi harga smartphone dalam dataset.

Visualisasi ini menggunakan histogram untuk menampilkan frekuensi kemunculan berbagai rentang harga, serta kurva KDE (Kernel Density Estimate) untuk memperlihatkan pola sebaran data secara halus.

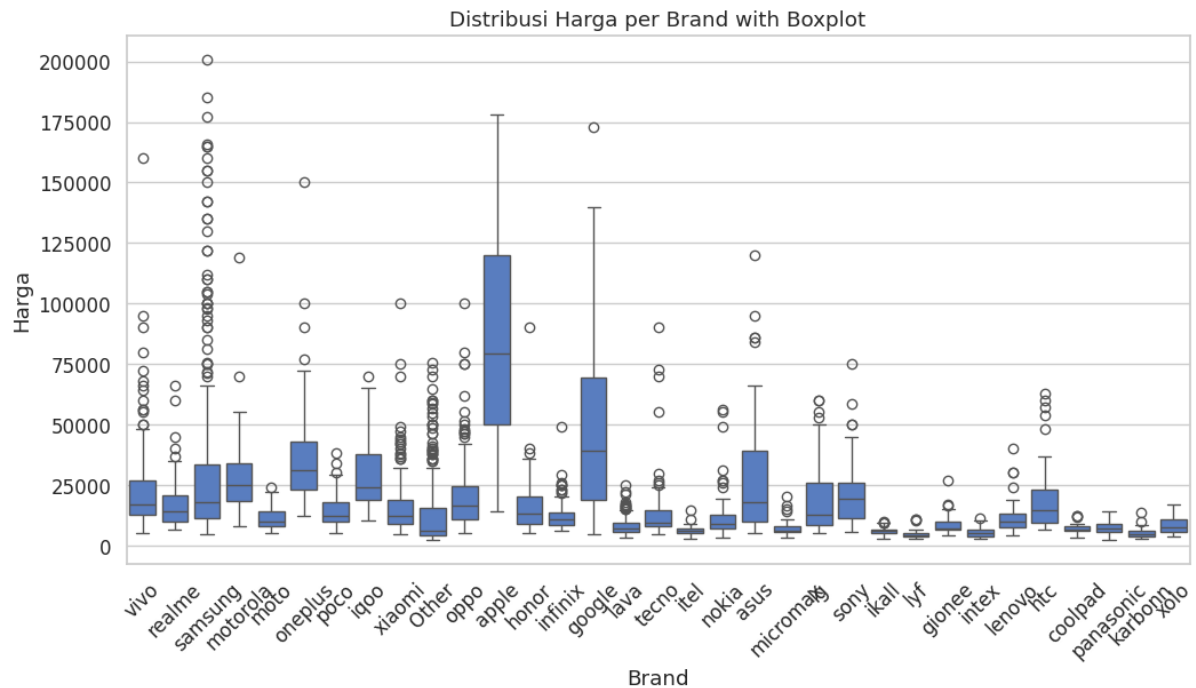
- Analisis Sebaran RAM dan Storage Smartphone



Sebagian besar smartphone pada dataset berada di segmen menengah, dengan RAM 4 GB dan storage 64 GB sebagai nilai umum. Adanya outlier pada kedua variabel menunjukkan diversifikasi produk —

produsen menawarkan model dengan spesifikasi tinggi namun jumlahnya terbatas. Secara umum, distribusi RAM dan storage menunjukkan variabilitas yang cukup besar, yang bisa berpengaruh terhadap harga dan performa smartphone.

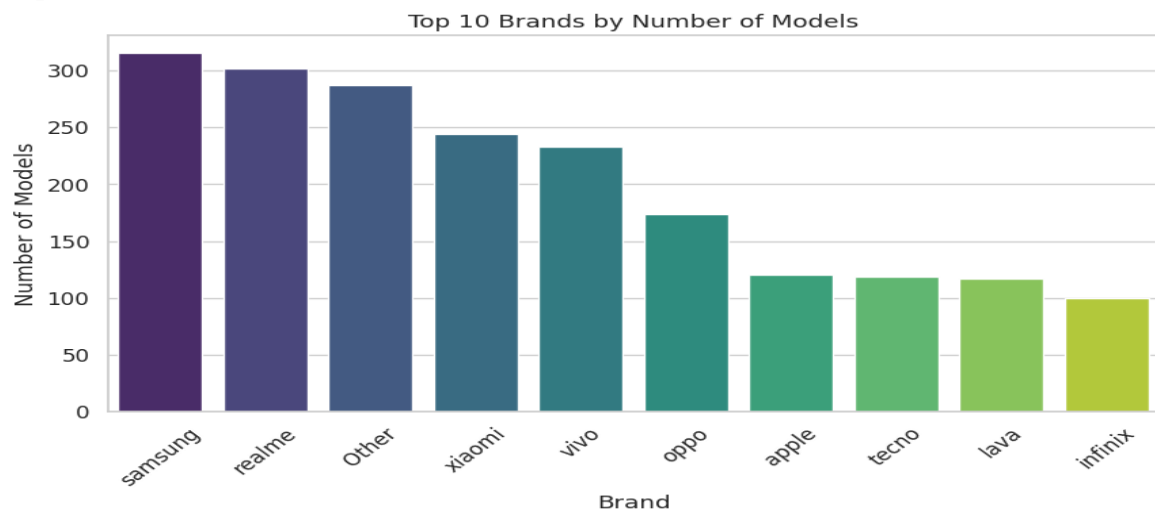
- Distribusi Harga per Brand



Apple secara konsisten menjadi brand dengan harga tertinggi di pasar, memperkuat citranya sebagai produk premium. Samsung, Oppo, dan Google memiliki portofolio yang luas, melayani berbagai segmen konsumen. Brand seperti Vivo, Realme, dan Xiaomi berfokus pada harga terjangkau, bersaing di pasar menengah dan entry-level.

Secara keseluruhan, distribusi harga menunjukkan bahwa pasar smartphone sangat bervariasi antar brand, dengan sebagian besar merek berfokus pada harga rendah hingga menengah, sedangkan hanya segelintir yang mendominasi segmen premium.

- Top 10 Brand berdasarkan model



- Samsung menjadi brand dengan jumlah model terbanyak dalam dataset, menunjukkan strategi diversifikasi produk yang luas untuk menjangkau berbagai segmen pasar.

- Realme dan Xiaomi juga menempati posisi tinggi, menandakan fokus kuat pada variasi model dengan spesifikasi dan harga beragam.
- Vivo dan Oppo berada di tengah, memperlihatkan keseimbangan antara kualitas dan kuantitas model.
- Apple memiliki jumlah model yang relatif sedikit dibandingkan brand lain, yang menunjukkan strategi fokus pada lini produk premium dan eksklusif.
- Brand seperti Tecno, Lava, dan Infinix muncul dalam 10 besar, menandakan popularitas mereka di pasar menengah ke bawah dengan penawaran produk yang cukup banyak.

Secara keseluruhan, distribusi ini menunjukkan bahwa pasar smartphone didominasi oleh brand-brand dengan portofolio produk luas, sementara brand premium seperti Apple tetap mempertahankan strategi eksklusifitas dengan jumlah model terbatas.

- Korelasi antara Harga, RAM, dan Kamera utama



- Harga memiliki korelasi sedang dengan RAM (0.54), menunjukkan bahwa semakin besar kapasitas RAM, cenderung semakin tinggi pula harga smartphone.
- Korelasi harga dengan kamera utama (0.29) tergolong lemah, artinya peningkatan resolusi kamera utama tidak selalu berbanding lurus dengan kenaikan harga.
- RAM dan kamera utama memiliki korelasi cukup kuat (0.66), yang menandakan bahwa smartphone dengan RAM besar umumnya juga dilengkapi dengan kamera utama yang lebih baik.

Dari hasil diatas dapat di simpulkan faktor RAM memiliki pengaruh yang lebih signifikan terhadap harga smartphone dibandingkan dengan kamera utama. Hal ini menunjukkan bahwa performa (RAM) lebih menentukan harga dibandingkan fitur fotografi.

3.1.4. Statistical Analysis

- **Data Uji Parametrik menggunakan Correlation**

Uji parametrik menggunakan Pearson correlation untuk mengetahui sejauh mana hubungan antara dua variabel numerik, yaitu RAM dan Price.

Metode ini bertujuan untuk mengukur kekuatan dan arah hubungan linier antara kedua variabel.

Berdasarkan hasil perhitungan menggunakan fungsi `pearsonr` dari pustaka `scipy.stats`, diperoleh nilai koefisien korelasi (r) sebesar 0.539 dan nilai signifikansi (p -value) sebesar 3.714×10^{-245} .

Nilai koefisien korelasi tersebut menunjukkan adanya hubungan positif dengan kekuatan sedang hingga kuat antara kapasitas RAM dan harga produk.

Hal ini berarti bahwa semakin besar kapasitas RAM, maka cenderung semakin tinggi pula harga produk tersebut.

- Uji Parametrik menggunakan T-test

```
uji parametrik menggunakan T-Test

from scipy.stats import ttest_ind
import numpy as np
import scipy.stats as stats

# Pisahkan dua grup
group_5g = df[df['has_5g'] == 1]['Price']
group_non_5g = df[df['has_5g'] == 0]['Price']

# Uji Welch's t-test (unequal variances)
t_stat, p_val = ttest_ind(group_5g, group_non_5g, equal_var=False)
print("t-statistic:", t_stat)
print("p-value:", p_val)

# ---- Hitung Effect Size (Cohen's d) ----
n1, n2 = len(group_5g), len(group_non_5g)
mean1, mean2 = np.mean(group_5g), np.mean(group_non_5g)
std1, std2 = np.std(group_5g, ddof=1), np.std(group_non_5g, ddof=1)
pooled_std = np.sqrt(((n1 - 1)*std1**2 + (n2 - 1)*std2**2) / (n1 + n2 - 2))
cohen_d = (mean1 - mean2) / pooled_std
print(f"Cohen's d: {cohen_d:.3f}")

# ---- Hitung Confidence Interval ----
mean_diff = mean1 - mean2
se_diff = np.sqrt((std1**2/n1) + (std2**2/n2))
ci = stats.t.interval(0.95, df=min(n1, n2)-1, loc=mean_diff, scale=se_diff)
print(f"95% Confidence Interval: {ci}")

t-statistic: 21.820864119861756
p-value: 1.1388391708892913e-88
Cohen's d: 1.096
95% Confidence Interval: (np.float64(21495.51010976442), np.float64(25743.697461142703))
```

Tujuan dari uji ini adalah untuk mengetahui apakah terdapat perbedaan harga yang signifikan secara statistik antara kedua kelompok tersebut.

Metode t-test dipilih karena mengasumsikan bahwa kedua kelompok memiliki varians yang tidak sama (unequal variances) — kondisi ini lebih realistis untuk data dunia nyata.

- Uji Non-Parametrik Mann-Whitney U Test Non Parametrik

```
Mann-Whitney U Test Non Parametrik

from scipy.stats import mannwhitneyu, norm
import numpy as np

# Pisahkan data berdasarkan dukungan 5G
price_5g = df[df['has_5g'] == 1]['Price']
price_non_5g = df[df['has_5g'] == 0]['Price']

# Uji Mann-Whitney U
stat, p = mannwhitneyu(price_5g, price_non_5g, alternative='two-sided')
print(f"Mann-Whitney U Statistic: {stat}")
print(f"p-value: {p}")

# ---- Hitung Effect Size (r) ----
n1, n2 = len(price_5g), len(price_non_5g)
mean_U = n1*n2/2
std_U = np.sqrt(n1*n2*(n1+n2+1)/12)
z = (stat - mean_U) / std_U
r = abs(z) / np.sqrt(n1 + n2)
print(f"Effect size (r): {r:.3f}")

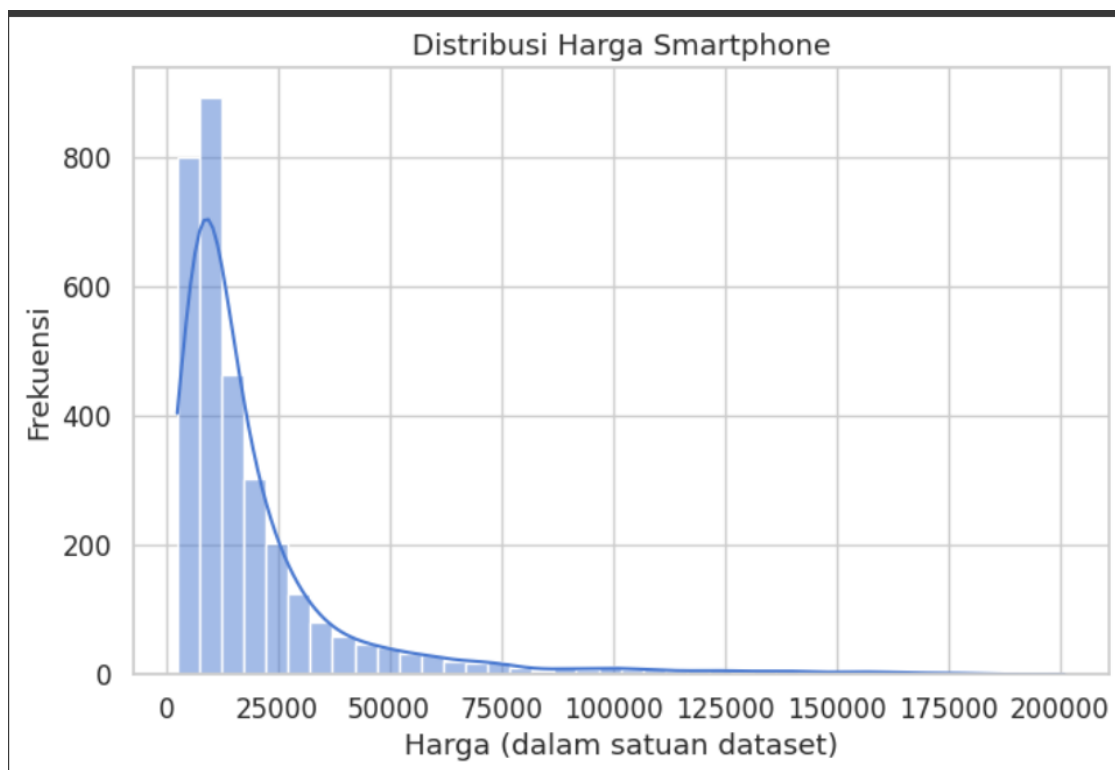
Mann-Whitney U Statistic: 1977531.0
p-value: 8.526131814948588e-255
Effect size (r): 0.597
```

Uji ini adalah untuk mengetahui apakah terdapat perbedaan yang signifikan secara statistik antara kedua kelompok tersebut, tanpa mengasumsikan data berdistribusi normal. Berbeda dengan uji t-test yang bersifat parametrik, uji Mann–Whitney U digunakan ketika data tidak memenuhi asumsi normalitas atau varians antar kelompok tidak homogen. Hasil uji Mann–Whitney U digunakan ketika asumsi normalitas tidak terpenuhi. Jika pada analisis ini diperoleh p-value yang kecil (<0.05) dan nilai effect size r yang cukup besar, maka dapat disimpulkan bahwa: Harga smartphone dengan fitur 5G secara signifikan lebih tinggi dibandingkan harga smartphone tanpa fitur 5G, dan perbedaan tersebut memiliki kekuatan efek yang berarti.

BAB IV Hasil Dan Pembahasan

4.1. Analisis dan Visualisasi Data

Histogram - Distribusi Harga Smartphone



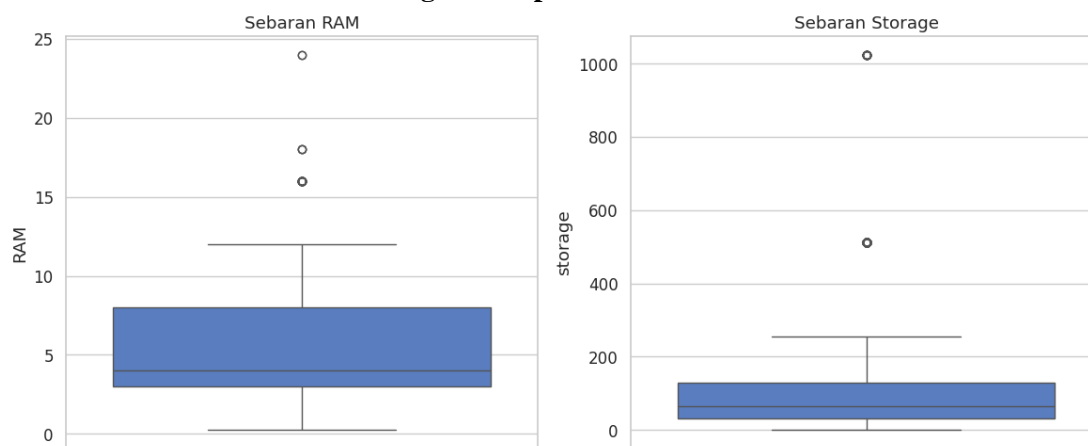
Visualisasi distribusi harga *smartphone* menggunakan Histogram dan Kurva *Kernel Density Estimate* (KDE) menghasilkan dua kesimpulan utama mengenai struktur harga dalam *dataset*:

1. **Dominasi Harga Menengah ke Bawah (*Right-Skewed*):** Distribusi harga menunjukkan pola menceng ke kanan (*Right-Skewed*), yang berarti mayoritas *smartphone* dalam *dataset* memiliki harga rendah hingga menengah. Frekuensi tertinggi (puncak kurva) terkonsentrasi di rentang harga yang lebih rendah. Hal ini

mengindikasikan bahwa pasar didominasi oleh segmen *entry-level* dan *mid-range* yang memiliki volume produk terbesar.

2. **Keberadaan Segmen *Flagship* sebagai *Outlier*:** Kurva KDE menunjukkan adanya ekor panjang (*long tail*) yang membentang ke sisi harga tinggi. Meskipun frekuensinya kecil, keberadaan *smartphone* ini mewakili segmen premium atau *flagship*. Segmen ini berfungsi sebagai *outlier* harga yang signifikan, yang nilai jualnya kemungkinan besar didorong oleh fitur eksklusif, merek, dan spesifikasi tertinggi, dan harus dipertimbangkan secara khusus dalam analisis inferensial berikutnya.

Analisis Sebaran RAM dan Storage Smartphone

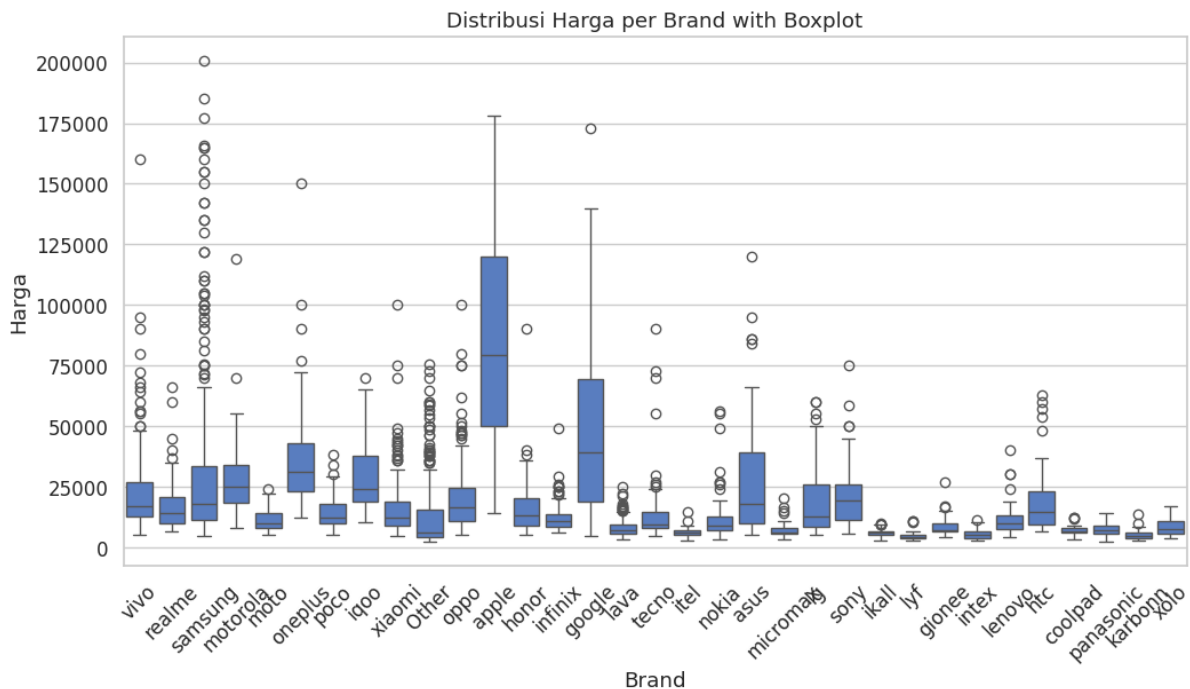


Visualisasi data mengenai kapasitas RAM dan Penyimpanan Internal (*Storage*) memberikan kesimpulan yang kuat mengenai fokus produksi dan diversifikasi penawaran produk di pasar *smartphone*:

1. **Fokus Segmen Menengah sebagai Modus:** Distribusi data menunjukkan bahwa sebagian besar *smartphone* dalam *dataset* terkonsentrasi pada spesifikasi RAM 4 GB dan Storage 64 GB. Titik konsentrasi ini mengidentifikasi segmen menengah (*mid-range*) sebagai nilai umum (modus) yang paling banyak diproduksi dan dipasarkan. Hal ini mengindikasikan bahwa produsen menargetkan spesifikasi ini sebagai titik keseimbangan terbaik antara biaya produksi, performa yang memadai, dan harga jual yang kompetitif.
2. **Bukti Diversifikasi Produk (*Outlier*):** Adanya pencilan (*outlier*) yang terpisah dari konsentrasi utama—terutama pada spesifikasi RAM tinggi (misalnya, 12 GB atau 16 GB) dan *storage* besar (misalnya, 512 GB)—membuktikan adanya diversifikasi produk yang disengaja. Pencilan ini merepresentasikan model *flagship* atau *gaming* yang ditawarkan produsen dalam jumlah terbatas untuk menargetkan ceruk pasar premium.
3. **Variabilitas sebagai Faktor Kunci Harga:** Secara umum, distribusi kedua variabel menunjukkan variabilitas yang cukup besar di luar nilai umum 4GB/64GB. Variabilitas ini sangat penting dalam analisis inferensial, karena peningkatan kapasitas RAM dan *storage* tersebut terbukti dalam analisis korelasi (Bab III)

memiliki pengaruh yang signifikan dan positif terhadap kenaikan harga jual *smartphone* serta penentuan performa.

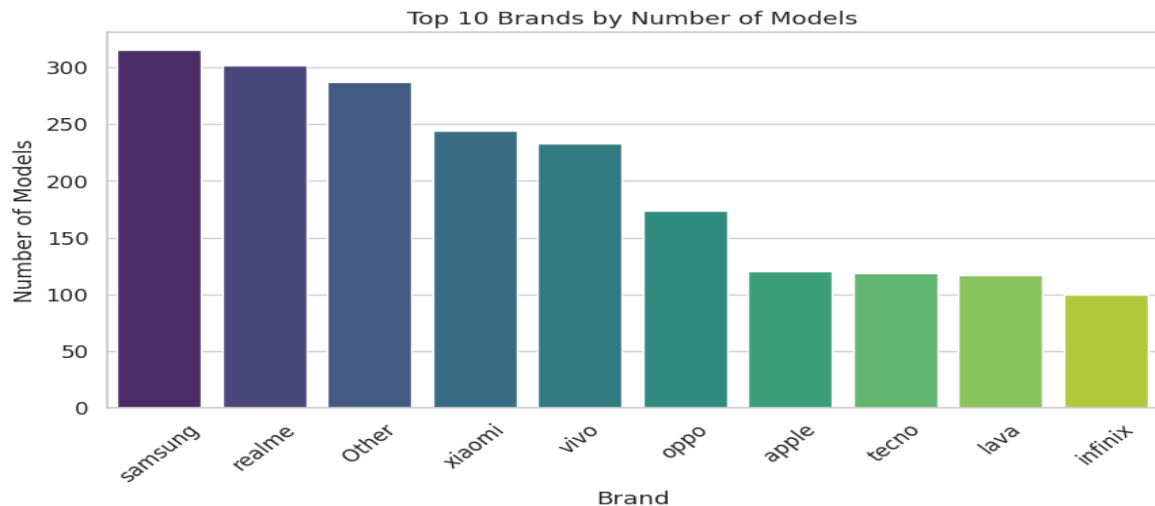
Distribusi Harga per Brand



Top 10 Brand berdasarkan model Visualisasi yang membandingkan rentang harga antar merek (*Brand*) memberikan kesimpulan penting mengenai strategi penetapan harga dan segmentasi pasar oleh produsen:

1. **Strategi Harga Premium (Apple):** Apple menunjukkan konsistensi sebagai merek dengan harga tertinggi dalam *dataset*. Hal ini memperkuat citra merek sebagai produk premium dan eksklusif. Strategi ini memosisikan Apple sebagai *benchmark* harga di segmen *flagship*, di mana harga cenderung tidak sensitif terhadap persaingan langsung dari sisi spesifikasi murni.
2. **Strategi Diversifikasi Luas (*Broad Diversification*):** Merek seperti Samsung, Oppo, dan Google menunjukkan portofolio harga yang luas. Hal ini mengindikasikan bahwa mereka menerapkan strategi diversifikasi, melayani berbagai segmen konsumen, mulai dari *entry-level* hingga *high-end*. Strategi ini memungkinkan mereka untuk memaksimalkan pangsa pasar di berbagai kategori daya beli.
3. **Fokus Segmen Menengah ke Bawah (*Value-Oriented*):** Brand seperti Vivo, Realme, dan Xiaomi mayoritas berfokus pada harga yang terjangkau, bersaing secara agresif di pasar menengah dan *entry-level*. Kesimpulan ini menunjukkan bahwa merek-merek ini mengedepankan nilai (*value*) dan volume penjualan dengan menawarkan spesifikasi yang baik pada harga yang lebih kompetitif.

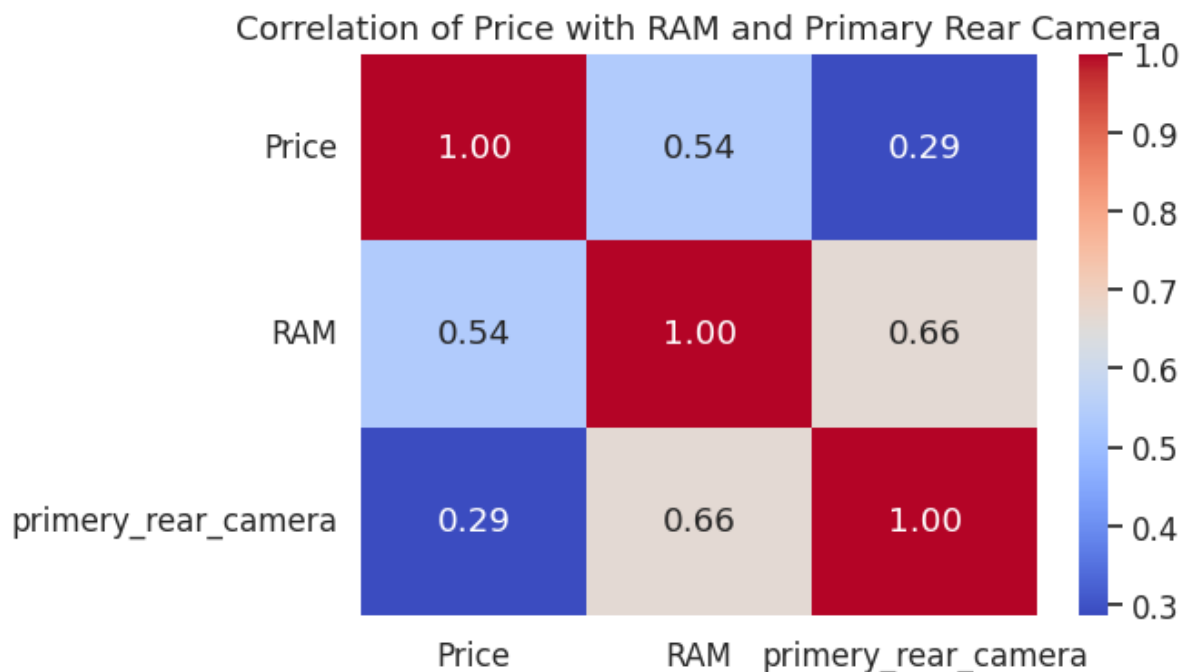
Top 10 Brands by Number of Models



Visualisasi yang menyajikan peringkat 10 merek teratas berdasarkan jumlah model *smartphone* memberikan kesimpulan penting mengenai strategi pemasaran dan jangkauan pasar yang diterapkan oleh produsen:

1. **Strategi Diversifikasi Volume (*Volume Diversification*):** Merek seperti Samsung, Realme, dan Xiaomi mendominasi posisi teratas dengan jumlah model terbanyak. Hal ini mengonfirmasi bahwa strategi utama mereka adalah diversifikasi produk yang luas, yakni dengan cepat mengisi berbagai segmen pasar—mulai dari harga terendah hingga menengah—untuk memaksimalkan *market share* dan memenuhi setiap kebutuhan konsumen secara spesifik.
2. **Keseimbangan Kuantitas dan Kualitas:** Merek di posisi tengah, seperti Vivo dan Oppo, menunjukkan keseimbangan strategis. Mereka memiliki jumlah model yang signifikan untuk bersaing secara volume, namun mungkin dengan fokus yang lebih terarah pada fitur tertentu (seperti kamera atau desain) di segmen menengah.
3. **Strategi Eksklusivitas (*Exclusivity Strategy*):** Apple secara jelas menonjol dengan jumlah model yang relatif sedikit dibandingkan dengan merek-merek Asia. Strategi ini menegaskan fokus mereka pada lini produk premium dan eksklusif. Apple memprioritaskan kualitas dan mempertahankan citra *flagship* tanpa perlu membanjiri pasar dengan banyak varian harga rendah.
4. **Munculnya Pesaing *Entry-Level*:** Kehadiran merek seperti Tecno, Lava, dan Infinix dalam 10 besar menunjukkan adanya persaingan yang kuat di pasar *entry-level* dan menengah ke bawah. Merek-merek ini juga menerapkan strategi variasi model yang cukup banyak untuk mendapatkan popularitas dan jangkauan di segmen yang sensitif terhadap harga.

Korelasi antara Harga, RAM, dan Kamera utama



Analisis visualisasi korelasi antara Harga, RAM, dan Kamera Utama (resolusi kamera utama) menghasilkan kesimpulan kunci mengenai faktor pendorong nilai jual *smartphone*:

1. Pengaruh Dominan RAM terhadap Harga:

Hubungan antara Harga dan RAM menunjukkan korelasi positif yang sedang hingga kuat (Koefisien Korelasi ≈ 0.54). Kesimpulan utamanya adalah kapasitas RAM memiliki pengaruh yang paling signifikan terhadap penentuan harga jual *smartphone*. Pasar menghargai peningkatan performa dan kemampuan multitasking (yang diwakili oleh RAM) lebih tinggi dibandingkan fitur lainnya.

2. Korelasi Lemah Kamera Utama dengan Harga:

Sebaliknya, korelasi antara Harga dan Kamera Utama hanya tergolong lemah (Koefisien Korelasi ≈ 0.29). Ini menunjukkan bahwa peningkatan resolusi kamera (dalam megapiksel) tidak selalu berbanding lurus dengan kenaikan harga jual. Faktor kamera, meskipun penting, tidak menjadi penentu harga utama, yang mengindikasikan bahwa fitur lain di luar spesifikasi megapiksel (seperti kualitas sensor, software processing, atau brand) mungkin lebih dominan.

3. Hubungan Kualitas Spesifikasi (Bundling):

Terdapat korelasi yang cukup kuat antara RAM dan Kamera Utama (Koefisien Korelasi ≈ 0.66). Kesimpulan ini menegaskan bahwa *smartphone* dengan kapasitas RAM yang besar (yang menandakan perangkat berorientasi performa tinggi atau flagship) umumnya juga dilengkapi dengan kualitas kamera utama yang lebih baik.

Korelasi ini mencerminkan praktik industri di mana peningkatan performa (RAM) cenderung dipaketkan (bundled) bersama fitur-fitur premium lainnya (Kamera Utama).

BAB V Kesimpulan

Berdasarkan hasil analisis data *smartphone* menggunakan pendekatan *Data Science* yang meliputi tahap *Data Preprocessing* tingkat lanjut, Visualisasi Data, dan Analisis Statistik (Korelasi serta Uji Hipotesis), maka dapat ditarik beberapa kesimpulan utama yang menjawab rumusan masalah penelitian

☐ Faktor Dominan Penentu Harga:

Faktor spesifikasi yang paling memengaruhi dan berkorelasi secara signifikan terhadap tinggi rendahnya harga *smartphone* adalah kapasitas RAM (Koefisien Korelasi $r \approx 0.54$) dan keberadaan fitur konektivitas 5G. Hal ini menunjukkan bahwa performa inti dan kesiapan teknologi jaringan merupakan nilai jual utama yang paling dihargai oleh pasar.

☐ Hubungan RAM dan Harga:

Terdapat hubungan positif dan signifikan antara kapasitas RAM dengan harga *smartphone*. Semakin besar kapasitas RAM dan penyimpanan internal (storage), semakin tinggi pula harga jualnya. Kesimpulan ini diperkuat oleh pengujian korelasi statistik yang menunjukkan hubungan yang kuat dan konsisten.

☐ Pengaruh Fitur Modern (5G):

Hasil Uji Mann-Whitney U menunjukkan bahwa harga *smartphone* dengan dukungan 5G secara statistik signifikan lebih tinggi dibandingkan dengan *smartphone* yang hanya mendukung 4G ($p\text{-value} < 0.05$). Keberadaan 5G menempatkan perangkat di segmen harga premium dan menengah ke atas.

☐ Perbandingan Merek dan Strategi Harga:

Strategi harga antar merek menunjukkan perbedaan mencolok. Merek seperti Apple secara konsisten mendominasi segmen harga premium dengan portofolio produk terbatas, sementara merek seperti Samsung, Realme, dan Xiaomi bersaing melalui diversifikasi produk massal di segmen entry-level hingga mid-range.

☐ Peran Fitur Fotografi:

Fitur atau spesifikasi seperti resolusi Kamera Utama memiliki korelasi yang tergolong lemah terhadap harga ($r \approx 0.29$). Ini menyiratkan bahwa setelah melewati ambang batas kualitas tertentu, peningkatan megapixels kamera tidak lagi menjadi faktor pendorong harga yang signifikan seperti halnya peningkatan RAM atau adopsi 5G.

Lampiran

(Hartini, 2017)Hartini, E. (2017). Classification of Missing Values Handling Method During Data Mining: Review. *Sigma Epsilon*, 21(2), 49–60.

Oktavian, R. S., & Budi, S. (2020). Menggunakan Metode Exploratory. *Jurnal Strategi*, 2(2), 636–649.

Sanjaya, G., Rinaldi, A. R., & Basysyar, F. M. (2025). Penerapan Algoritma K-Means Untuk Mengelompokkan. *Journal Of Computer Science And Artificial Intellegence (JCSAI)*, 1(4), 1–8. <https://ruangjurnal.or.id/index.php/jcsai/article/view/16>

(Sanjaya et al., 2025)

(Oktavian & Budi, 2020)