

# PROJECT BIG DATA PROCESSING

Aaron Adriano - 2702276866

Cornelius Jason Aslim - 2702337905

Leon - 2702251604

Nicholas Nathanael Lo-2702313470

Edric Emerson - 2702229592

Khuang Ming Jeremy Alexander - 2702277553

# JUDUL PROJECT

**PEMODELAN FAKTOR-FAKTOR YANG  
MEMPENGARUHI REVENUE PENJUALAN  
SMARTPHONE DENGAN KORELASI,  
ANOVA, DAN RANDOM FOREST**

# DAFTAR ISI

- LATAR BELAKANG
- METODOLOGI DAN ALUR PEKERJAAN
- EVALUASI DAN DETAIL DARI ALUR PEKERJAAN
- KESIMPULAN

# LATAR BELAKANG

Industri smartphone telah berkembang pesat dalam satu dekade terakhir, memicu persaingan ketat antar produsen. Untuk menyusun strategi bisnis yang efektif, penting untuk memahami faktor-faktor yang memengaruhi keputusan konsumen dalam membeli smartphone, seperti harga, spesifikasi, merek, strategi pemasaran, dan tren konsumen.

Pendekatan kuantitatif berbasis data dibutuhkan untuk menganalisis pengaruh faktor-faktor tersebut terhadap penjualan. Metode seperti:

- Analisis Korelasi: mengukur kekuatan hubungan antar variabel.
- Analisis Varians (ANOVA): menguji perbedaan penjualan berdasarkan kategori seperti merek atau kelas harga.
- Random Forest: mengidentifikasi variabel paling berpengaruh dan memprediksi penjualan dari data historis.

Dengan menggabungkan ketiga metode ini, penelitian dapat memberikan pemahaman komprehensif mengenai faktor-faktor kunci dalam penjualan smartphone. Hasilnya dapat menjadi dasar dalam pengambilan keputusan strategis seperti pengembangan produk, penetapan harga, hingga perencanaan promosi dan distribusi.

# DATASET

Mobiles & laptop Sales Data

By: VINOTH KANNA S

<https://www.kaggle.com/datasets/vinothkannaece/mobiles-and-laptop-sales-data>

# METODOLOGI DAN ALUR PEKERJAAN

## 2.1 Data Preparation

```
df['Inward Date'] = pd.to_datetime(df['Inward Date'])
df['Dispatch Date'] = pd.to_datetime(df['Dispatch Date'])
```

```
df['DaysInStock'] = (df['Dispatch Date'] - df['Inward Date']).dt.days
```

```
def extract_numeric_storage(value):
    if pd.isna(value):
        return 0
    value_str = str(value).upper().replace(' ', '')
    if 'TB' in value_str:
        numeric_part = ''.join(filter(lambda x: x.isdigit() or x == '.', 
value_str.replace('TB', '')))
        if numeric_part:
            return int(float(numeric_part) * 1024)
        else:
            return 0
    elif 'GB' in value_str:
        numeric_part = ''.join(filter(lambda x: x.isdigit() or x == '.', 
value_str.replace('GB', '')))
        if numeric_part:
            return int(float(numeric_part))
        else:
            return 0
    try:
        return int(float(value_str))
    except ValueError:
        return 0
```

```
df['Revenue'] = df['Price'] * df['Quantity Sold']
print("Kolom 'Revenue' berhasil dibuat.")
```

- Seleksi Data
- Konversi Format Tanggal.
- Perhitungan Lama Produk di Gudang
- Penanganan Nilai Kosong (Missing Values) dan Mengekstrak nilai numerik
- Pembuatan Kolom Revenue

# METODOLOGI DAN ALUR PEKERJAAN

## 2.2 Eksplorasi Data

```
brand_sales = df_mobile_cleaned.groupby('Brand')['Quantity Sold'].sum().sort_values(ascending=False)
brand_sales.plot(kind='bar', title='Total Quantity Sold per Brand')
plt.ylabel('Quantity Sold')
plt.show()
```

```
brand_revenue =
df_mobile_cleaned.groupby('Brand')['Revenue'].mean().sort_values(ascending=False)
brand_revenue.plot(kind='bar', color='orange', title='Average Revenue per Brand')
plt.ylabel('Revenue')
plt.show()
```

```
sns.heatmap(df_mobile_cleaned[['Price', 'Quantity Sold', 'RAM_GB', 'ROM_GB', 'Revenue', 'DaysInStock']].corr(), annot=True, cmap='coolwarm')
plt.title("Correlation Matrix")
plt.show()
```

- Bar Chart Total Penjualan per Merek
- Rata-rata Pendapatan (Revenue) per Merek
- Heatmap Korelasi Antar Variabel Numerik

# METODOLOGI DAN ALUR PEKERJAAN

## 2.3 Model Prediksi

```
X = df.drop(['Revenue', 'Customer Name', 'Product Code', 'Dispatch Date',  
'Inward Date'], axis=1)  
y = df['Revenue']  
  
from sklearn.compose import ColumnTransformer  
from sklearn.preprocessing import OneHotEncoder, StandardScaler  
from sklearn.pipeline import Pipeline  
  
num_cols = X.select_dtypes(include=['int64', 'float64']).columns  
cat_cols = X.select_dtypes(include=['object']).columns  
  
preprocessor = ColumnTransformer([  
    ('num', StandardScaler(), num_cols),  
    ('cat', OneHotEncoder(handle_unknown='ignore'), cat_cols)  
])  
  
from sklearn.ensemble import RandomForestRegressor  
from sklearn.model_selection import train_test_split  
from sklearn.metrics import r2_score, mean_absolute_error  
  
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,  
random_state=42)  
  
model = Pipeline([  
    ('preprocessor', preprocessor),  
    ('regressor', RandomForestRegressor(n_estimators=100, random_state=42))  
])  
  
model.fit(X_train, y_train)  
  
y_pred = model.predict(X_test)
```

Setelah EDA, dibangun model prediktif untuk memprediksi Revenue berdasarkan spesifikasi produk.

- Tujuan: Mengukur pengaruh fitur terhadap pendapatan & prediksi untuk produk baru.
- Model: Random Forest Regressor, karena:
  - Cocok untuk data numerik & kategorik
  - Tahan terhadap overfitting
  - Akurat untuk regresi

# METODOLOGI DAN ALUR PEKERJAAN

## 2.1 Evaluasi Model

```
print("R2 Score:", r2_score(y_test, y_pred))
print("MAE:", mean_absolute_error(y_test, y_pred))
```

Evaluasi dilakukan untuk mengukur akurasi model dalam memprediksi Revenue.

Metode evaluasi yang digunakan:

- R<sup>2</sup> Score → mengukur seberapa baik prediksi mengikuti nilai aktual
- MAE (Mean Absolute Error) → menghitung rata-rata selisih absolut prediksi dan nilai asli

# METODOLOGI DAN ALUR PEKERJAAN

## 2.5 Analysis Feature Importance

```
import matplotlib.pyplot as plt

feature_names = model.named_steps['preprocessor'].transformers_[0][2].tolist()
+ \

list(model.named_steps['preprocessor'].transformers_[1][1].get_feature_names_out(cat_cols))

importances = model.named_steps['regressor'].feature_importances_
feat_imp = pd.Series(importances,
index=feature_names).sort_values(ascending=False)

feat_imp.head(10).plot(kind='barh')
plt.title('Top 10 Feature Importance')
```

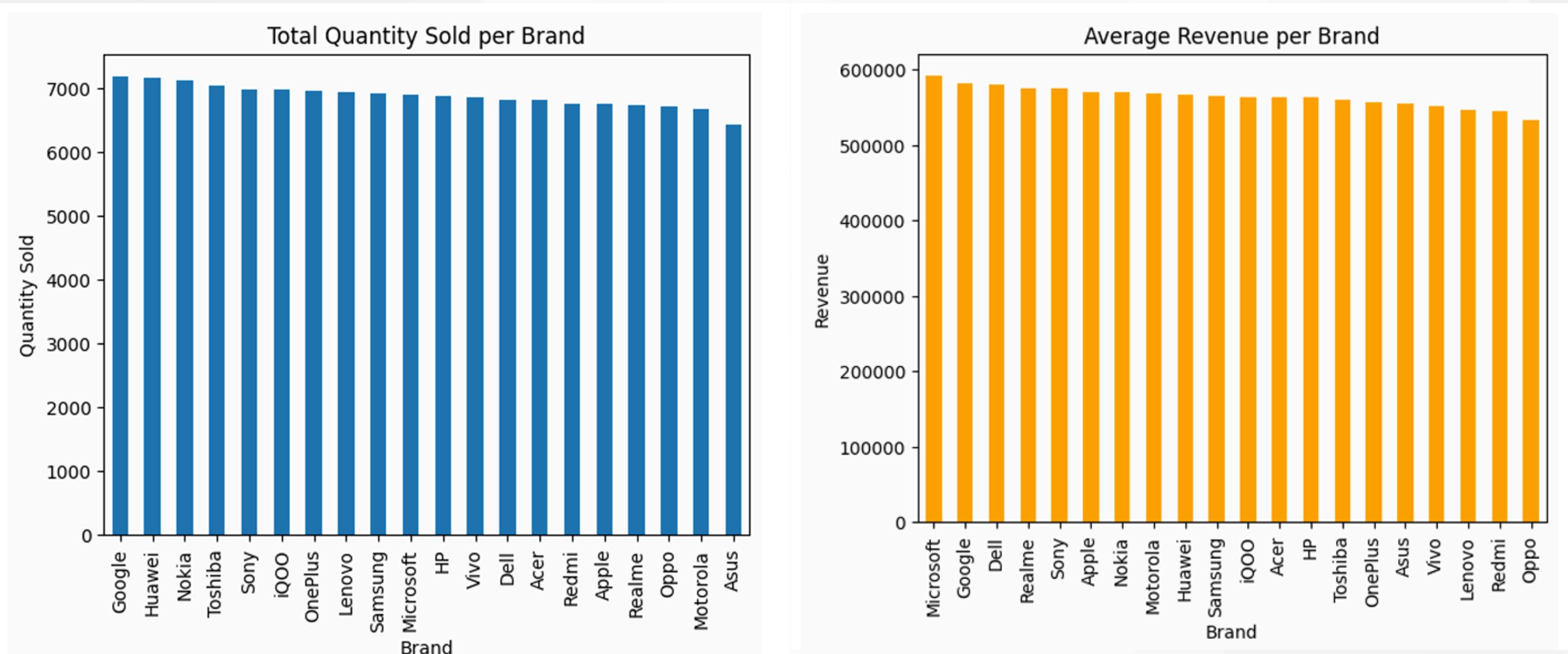
```
plt.gca().invert_yaxis()
plt.show()
```

Random Forest memberikan nilai feature importance yang menunjukkan kontribusi masing-masing fitur terhadap performa model.

Visualisasi dibuat untuk menunjukkan fitur paling signifikan, seperti:

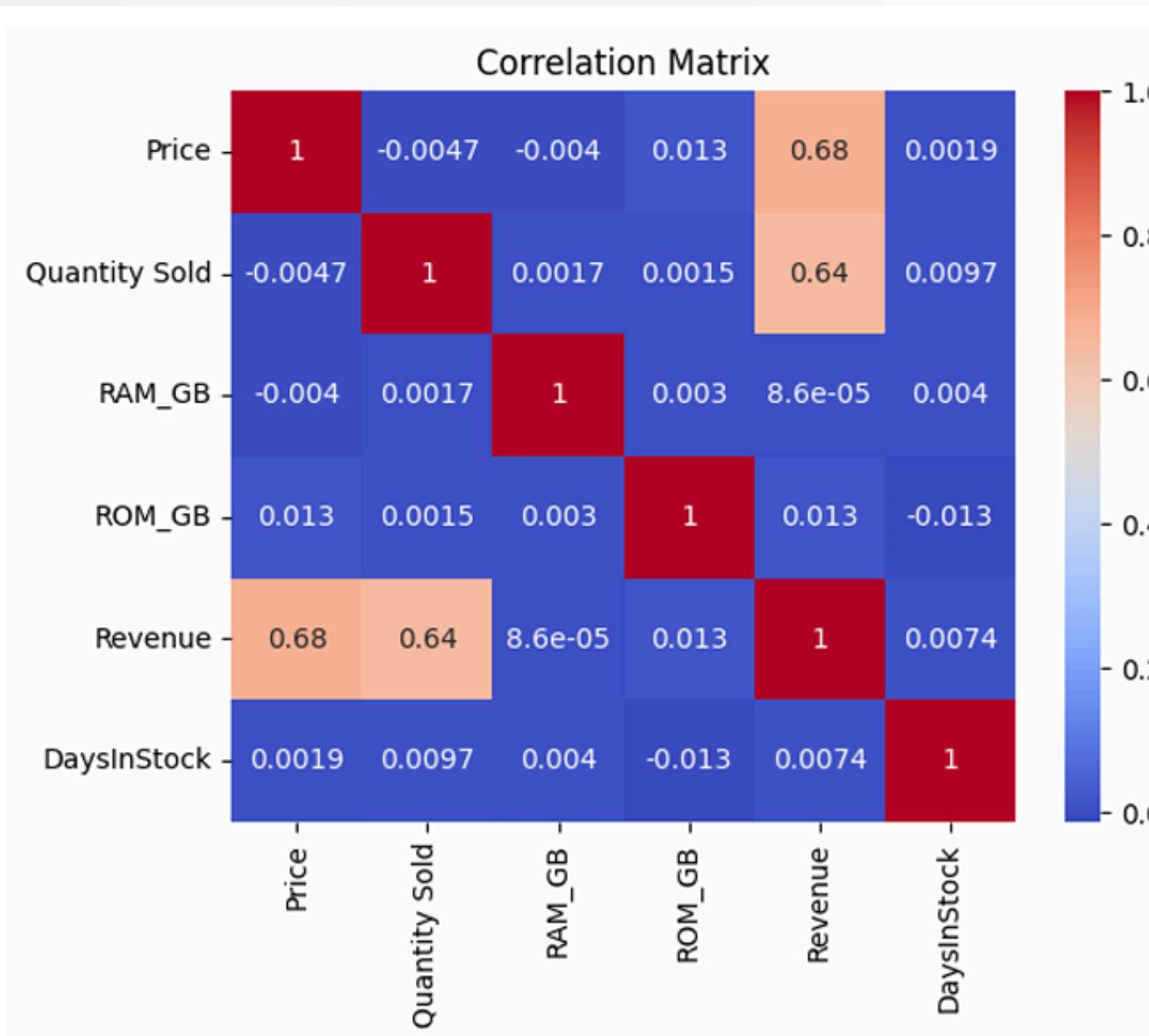
- Price
- Quantity Sold
- Brand
- RAM / ROM

# EVALUASI DAN DETAIL DARI ALUR PEKERJAAN



- Visualisasi bar chart menunjukkan bahwa merek seperti Google, Huawei, dan Nokia mendominasi penjualan berdasarkan jumlah unit terjual. menandakan bahwa brand tersebut memiliki daya tarik yang tinggi di pasar.
- Di sisi lain, Microsoft dan Dell memiliki rata-rata pendapatan per produk yang lebih tinggi, yang mengindikasikan bahwa mereka menargetkan segmen premium atau menjual produk dengan margin lebih besar.

# EVALUASI DAN DETAIL DARI ALUR PEKERJAAN



- Analisis korelasi menunjukkan bahwa dua variabel yang memiliki pengaruh terbesar dan terhadap pendapatan:
  - Price ( $r = +0.68$ ) dan Quantity Sold ( $r = +0.64$ ),
- Sebaliknya, fitur teknis seperti RAM dan ROM memiliki korelasi yang sangat rendah terhadap revenue.
  - RAM ( $r = 8.6e-05$ ) dan ROM ( $r = 0.013$ )

# KESIMPULAN

Model prediksi revenue berhasil dibangun dengan Random Forest Regressor dan menunjukkan performa sangat tinggi ( $R^2 = 0.99999$ , MAE = 177.75).

Fitur paling berpengaruh terhadap pendapatan:

- Quantity Sold
- Price

Fitur teknis seperti RAM, ROM, dan prosesor memiliki pengaruh kecil.

Model ini efektif untuk:

- Prediksi revenue produk baru
- Simulasi strategi penjualan
- Evaluasi dampak perubahan harga
- Penyusunan target pendapatan berbasis data

**TERIMAKASIH  
ATAS PERHATIANNYA**