

PREDIKSI HARGA MOBIL BEKAS MENGGUNAKAN REGRESI LINEAR

RIZKY BARUNA
5A - INFORMATIKA

REGRESI LINEAR

REGRESI LINEAR ADALAH SALAH SATU ALGORITMA MACHINE LEARNING PALING DASAR YANG DIGUNAKAN UNTUK MEMODELKAN HUBUNGAN ANTARA SATU VARIABEL DEPENDEN (TARGET) DENGAN SATU ATAU LEBIH VARIABEL INDEPENDEN (FITUR). TUJUANNYA ADALAH UNTUK MENEMUKN GARIS LURUS TERBAIK (BEST-FIT LINE) YANG DAPAT MEREPRESENTASIKAN HUBUNGAN ANTAR VARIABEL TERSEBUT, SEHINGGA DAPAT DIGUNAKAN UNTUK MELAKUKAN PREDIKSI.

PRE-PROCESSING

PROFILING DATA

TAHAP DATA PROFILING BERTUJUAN
UNTUK MEMAHAMI KARAKTERISTIK
DASAR DATASET.

DISINI DATASET YANG DIGUNAKAN
ADALAH "USED CARS PRICE
PREDICTION"

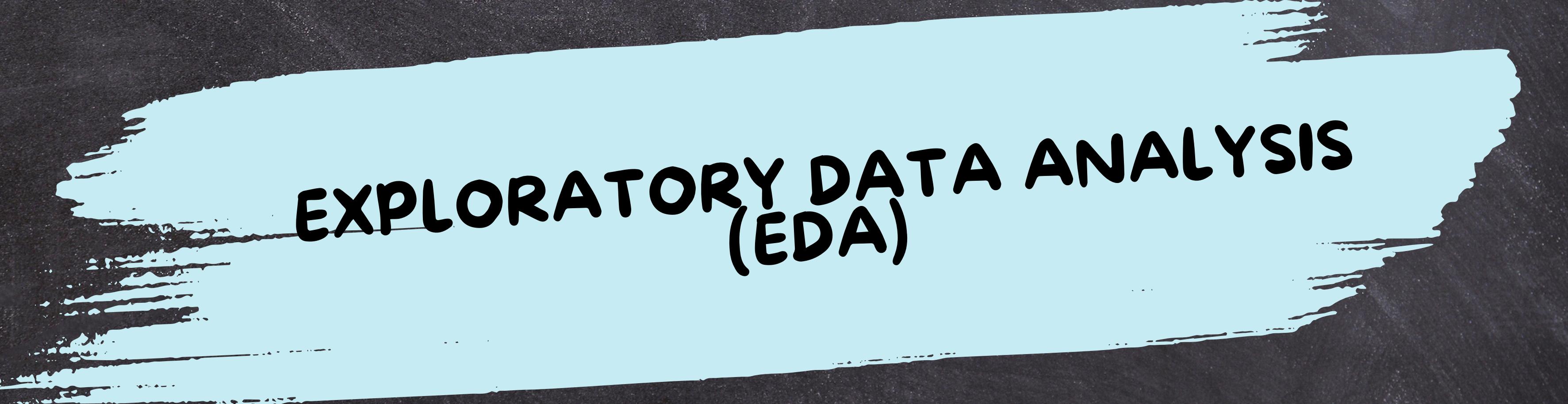
SUMBER DATASET:

[HTTPS://RAW.GITHUBUSERCONTENT.COM/FARRELLADITYAAA/DATASETS-UTS-DATAMINING/REFS/HEADS/MAIN/USED_CARS_PRICE_FIKS.CSV](https://raw.githubusercontent.com/farrelladityaaa/datasets-uts-datamining/reviews/main/used_cars_price_fiks.csv)

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	Price
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000.0	CNG	Manual	First	26.60	998.0	58.16	5.0	1.75
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000.0	Diesel	Manual	First	19.67	1582.0	126.20	5.0	12.50
2	2	Honda Jazz V	Chennai	2011	46000.0	Petrol	Manual	First	18.20	1199.0	88.70	5.0	4.50
3	3	Maruti Ertiga VDI	Chennai	2012	87000.0	Diesel	Manual	First	20.77	1248.0	88.76	7.0	6.00
4	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670.0	Diesel	Automatic	Second	15.20	1968.0	140.80	5.0	17.74
...
6014	6014	Maruti Swift VDI	Delhi	2014	27365.0	Diesel	Manual	First	28.40	1248.0	74.00	5.0	4.75
6015	6015	Hyundai Xcent 1.1 CRDi S	Jaipur	2015	100000.0	Diesel	Manual	First	24.40	1120.0	71.00	5.0	4.00
6016	6016	Mahindra Xylo D4 BSIV	Jaipur	2012	55000.0	Diesel	Manual	Second	14.00	2498.0	112.00	8.0	2.90
6017	6017	Maruti Wagon R VXI	Kolkata	2013	46000.0	Petrol	Manual	First	18.90	998.0	67.10	5.0	2.65
6018	6018	Chevrolet Beat Diesel	Hyderabad	2011	47000.0	Diesel	Manual	First	25.44	936.0	57.60	5.0	2.50

	count	mean	std	min	25%	50%	75%	max
Unnamed: 0	6019.0	3009.000000	1737.679967	0.00	1504.50	3009.00	4513.50	6018.00
Year	6019.0	2013.358199	3.269742	1998.00	2011.00	2014.00	2016.00	2019.00
Kilometers_Driven	5719.0	57545.592586	37988.496154	171.00	33923.00	53000.00	72998.00	775000.00
Mileage	6017.0	18.134961	4.582289	0.00	15.17	18.15	21.10	33.54
Engine	5983.0	1621.276450	601.355233	72.00	1198.00	1493.00	1984.00	5998.00
Power	5876.0	113.253050	53.874957	34.20	75.00	97.70	138.10	560.00
Seats	5977.0	5.278735	0.808840	0.00	5.00	5.00	5.00	10.00
Price	6019.0	9.479468	11.187917	0.44	3.50	5.64	9.95	160.00

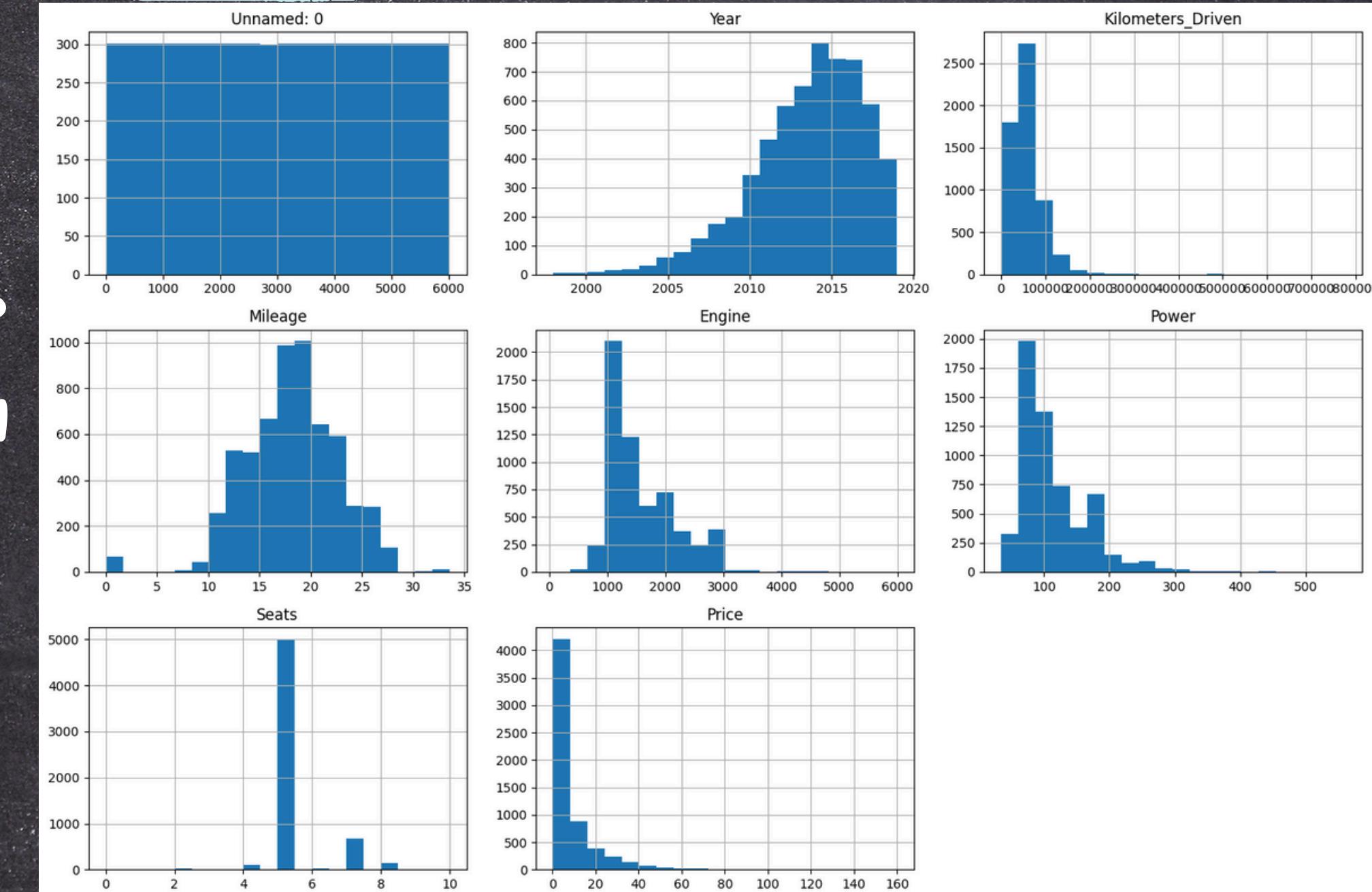
RIZKY



**EXPLORATORY DATA ANALYSIS
(EDA)**

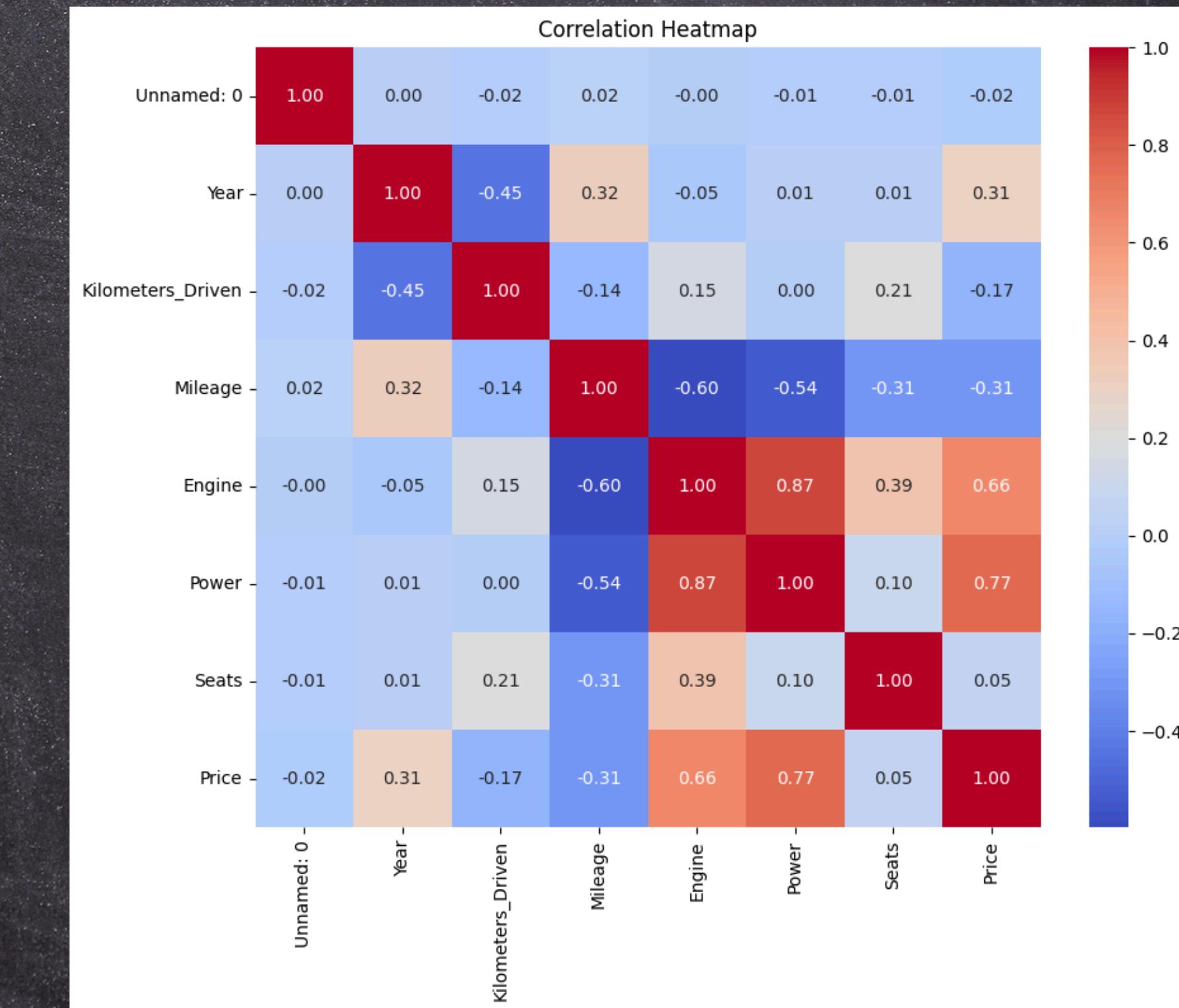
HISTOGRAM

GRAFIK HISTOGRAM DIGUNAKAN UNTUK MELIHAT DISTRIBUSI DATA DI SETIAP KOLOM. DARI HASIL VISUALISASI, TERLIHAT BAHWA KOLOM PRICE, POWER, DAN KILOMETERS_DRIVEN DISTRIBUSINYA CONDONG KE KANAN.INI MENANDAKAN ADANYA BEBERAPA DATA EKSTREM DENGAN NILAI YANG SANGAT TINGGI. SEMENTARA ITU, KOLOM SEATS SANGAT DIDOMINASI OLEH ANGKA 5, YANG BERARTI MOBIL 5 KURSI ADALAH YANG PALING UMUM DI DATASET INI.

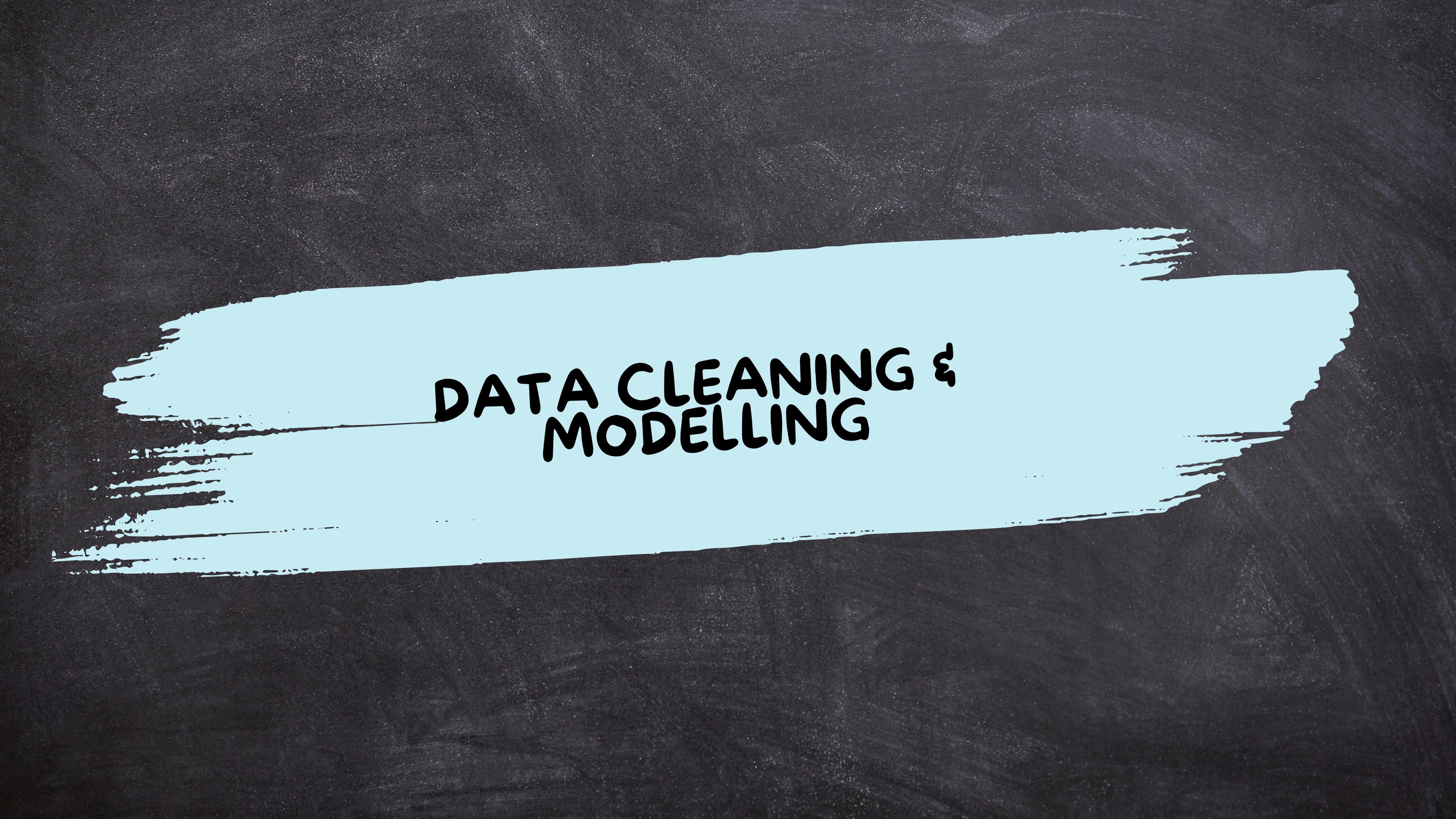


HEAT MAP

HEATMAP KORELASI MENUNJUKKAN
SEBERAPA KUAT HUBUNGAN ANTAR
VARIABEL. DITEMUKAN BAHWA PRICE
MEMILIKI KORELASI POSITIF TERKUAT
DENGAN POWER (0.77) DAN ENGINE
(0.66). ARTINYA, SEMAKIN BESAR
TENAGA DAN KAPASITAS MESIN, MAKA
HARGA MOBIL CENDERUNG SEMAKIN
TINGGI.



RIZKY



**DATA CLEANING &
MODELLING**

DATA CLEANING

PROSES PEMBERSIHAN DATA DILAKUKAN UNTUK MEMASTIKAN KUALITAS DATA. NILAI NULL YANG SEBELUMNYA DITEMUKAN, DIISI DENGAN NILAI RATA-RATA DARI MASING-MASING KOLOM. KOLOM ID (UNNAMED: 0) DIHAPUS, DAN TIDAK DITEMUKAN ADANYA DATA DUPLIKAT. LANGKAH YANG PALING SIGNIFIKAN ADALAH PENGHAPUSAN OUTLIERS, DI MANA SEBANYAK 1754 BARIS DATA (29.14%) DIHAPUS KARENA DIANGGAP SEBAGAI DATA PENCILAN.

```
[50] ✓ 0 d
# Buat salinan data untuk proses cleaning
df_clean = df.copy()

# Cek nilai Null pada dataset sebelum dibersihkan
print("Jumlah nilai null SEBELUM dibersihkan:\n", df_clean.isnull().sum())

# Kolom-kolom yang akan diisi nilai null-nya
cols_with_missing = ['Kilometers_Driven', 'Mileage', 'Engine', 'Power', 'Seats']

# Proses pengisian nilai null dengan rata-rata
for col in cols_with_missing:
    df_clean[col] = df_clean[col].fillna(df_clean[col].mean())

print("\nJumlah nilai null SETELAH dibersihkan:\n", df_clean.isnull().sum())

↳ Jumlah nilai null SEBELUM dibersihkan:
Unnamed: 0      0
Name            0
Location        0
Year            0
Kilometers_Driven  300
Fuel_Type       0
Transmission    0
Owner_Type      0
Mileage          2
Engine           36
Power            143
Seats            42
Price             0
dtype: int64

Jumlah nilai null SETELAH dibersihkan:
Unnamed: 0      0
Name            0
Location        0
Year            0
Kilometers_Driven  0
Fuel_Type       0
Transmission    0
Owner_Type      0
Mileage          0
Engine           0
Power            0
Seats            0
Price             0
dtype: int64
```

```
[51] ✓ 0 d
▼ Remove Duplicate Value
[52] ✓ 0 d
# Cek duplikasi dataset
duplicate_count = df_clean.duplicated().sum()
print(f"Jumlah data duplikat yang ditemukan: {duplicate_count}")

↳ Jumlah data duplikat yang ditemukan: 0

▼ Remove Column ID
[53] ✓ 0 d
# Menghapus kolom 'Unnamed: 0'
df_clean.drop(['Unnamed: 0'], axis=1, inplace=True)
print("Kolom 'Unnamed: 0' berhasil dihapus.")
print("Kolom saat ini:\n", df_clean.columns)

↳ Kolom 'Unnamed: 0' berhasil dihapus.
Kolom saat ini:
Index(['Name', 'Location', 'Year', 'Kilometers_Driven', 'Fuel_Type',
       'Transmission', 'Owner_Type', 'Mileage', 'Engine', 'Power', 'Seats',
       'Price'],
       dtype='object')

Kolom Unnamed: 0 dihapus karena merupakan duplikat dari indeks baris dan tidak mengandung informasi yang relevan untuk proses pemodelan.

▼ Remove Outliers
[54] ✓ 0 d
# Buat salinan data khusus untuk proses penghapusan outliers
df_outlier = df_clean.copy()
num_cols = df_outlier.select_dtypes(include=np.number).columns

# Fungsi untuk menghapus outlier
def remove_outliers_iqr(df, columns):
    df_no_outlier = df.copy()
    Q1 = df_no_outlier[columns].quantile(0.25)
    Q3 = df_no_outlier[columns].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR

    # Filter untuk menyimpan data yang BUKAN outlier
    outlier_filters = ~((df_no_outlier[columns] < lower_bound) | (df_no_outlier[columns] > upper_bound)).any(axis=1)
    df_no_outlier = df_no_outlier[outlier_filters]
    return df_no_outlier

# Terapkan fungsi untuk menghapus outlier
df_no_outlier = remove_outliers_iqr(df_outlier, num_cols)

print(f"Jumlah data sebelum hapus outlier: {len(df_outlier)}")
print(f"Jumlah data setelah hapus outlier: {len(df_no_outlier)}")
print(f"Persentase data yang dihapus: {round((len(df_outlier) - len(df_no_outlier)) / len(df_outlier) * 100, 2)}%")

↳ Jumlah data sebelum hapus outlier: 6019
Jumlah data setelah hapus outlier: 4265
Persentase data yang dihapus: 29.14%
```

MODELLING

DI TAHAP INI, DATA YANG MASIH BERBENTUK TEKS DIUBAH MENJADI ANGKA DENGAN METODE LABEL ENCODING. SELANJUTNYA, SEMUA DATA DIATUR KE DALAM SKALA YANG SAMA MENGGUNAKAN STANDARD SCALER. SETELAH ITU, DATA DIBAGI MENJADI DUA BAGIAN, YAITU DATA LATIH DAN DATA UJI. SELANJUTNYA, MODEL REGRESI LINEAR DILATIH UNTUK MEMAHAMI HUBUNGAN ANTARA BERBAGAI FITUR MOBIL DENGAN HARGA MOBIL TERSEBUT.

```
df_final = df_clean.copy()

# Pisahkan fitur (X) dan target (y)
X = df_final.drop('Price', axis=1)
y = df_final['Price']

# Mengubah fitur kategorikal menjadi angka
categorical_cols = ['Name', 'Location', 'Fuel_Type', 'Transmission', 'Owner_Type']
le = LabelEncoder()
for col in categorical_cols:
    X[col] = le.fit_transform(X[col])

# Scaling menggunakan StandardScaler()
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Split dataset ke data latih dan data uji
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=42)

print("Data siap untuk modelling.")
print(f"Ukuran data latih (X_train): {X_train.shape}")
print(f"Ukuran data uji (X_test): {X_test.shape}")

→ Data siap untuk modelling.
Ukuran data latih (X_train): (4815, 11)
Ukuran data uji (X_test): (1204, 11)
```

Pertumbuhan populasi memerlukan beberapa langkah penting dalam persiapan data sebelum dilatih oleh model:

- Pemisahan Fitur dan Target: Data dibagi menjadi variabel independen (X) dan variabel dependen (y).
- Label Encoding: Mengubah nilai kategorikal (teks) menjadi representasi numerik.
- Feature Scaling: Menyamakan skala nilai pada semua fitur menggunakan StandardScaler.
- Train-Test Split: Membagi dataset menjadi data latih dan data uji.

```
# Membuat dan melatih model Linear Regression
model_lr = LinearRegression()
model_lr.fit(X_train, y_train)

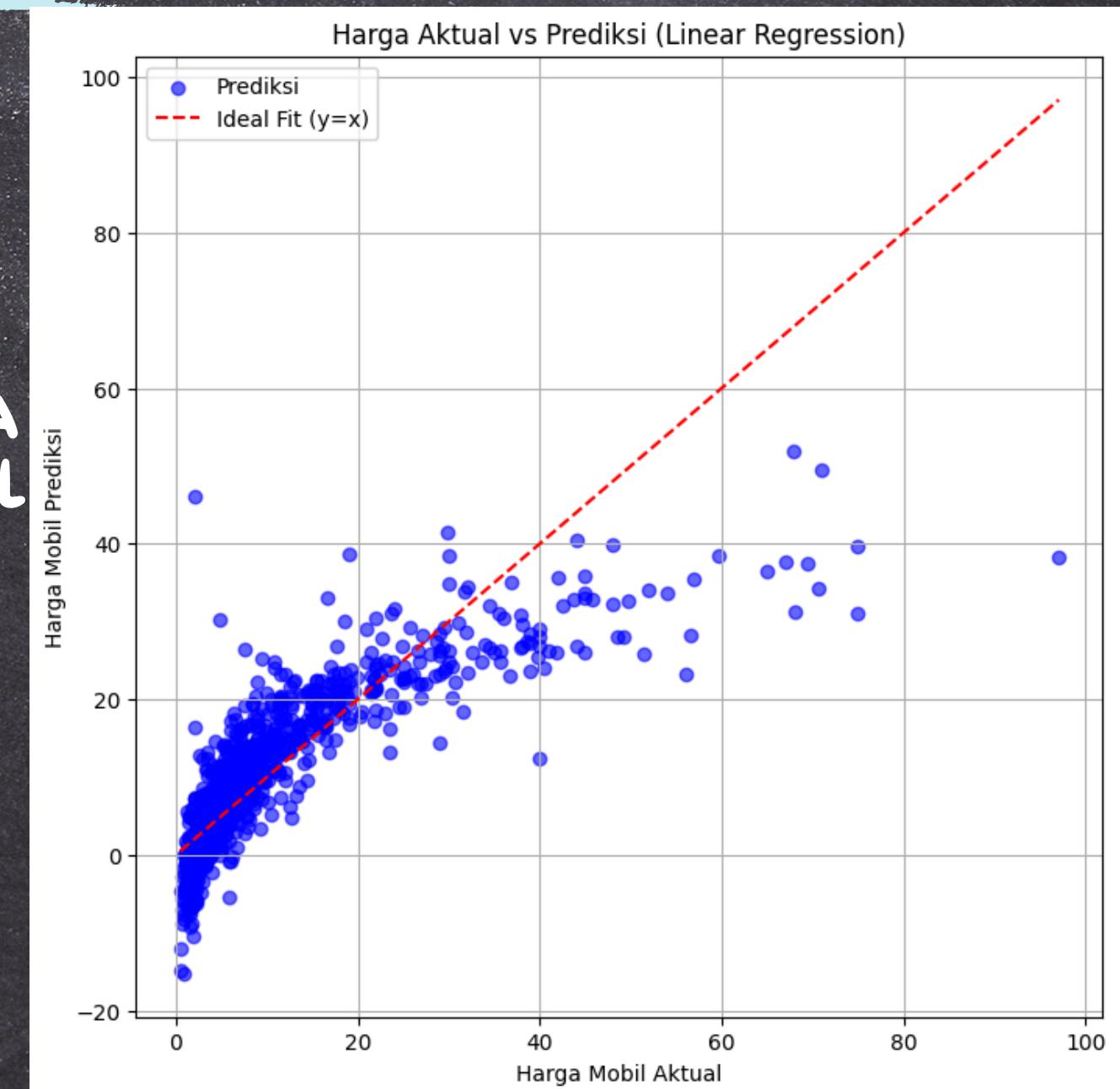
print("Model berhasil dilatih!")
print(f"Intercept (titik potong): {model_lr.intercept_}")
# print(f"Coefficient (kemiringan): {model_lr.coef_}")

→ Model berhasil dilatih!
Intercept (titik potong): 9.500834029011799
```

MODEL EVALUATION

HASIL EVALUASI

GRAFIK INI MENUNJUKKAN PERBANDINGAN ANTARA HARGA MOBIL ASLI (AKTUAL) DENGAN HASIL PREDIKSI DARI MODEL. DAPAT DILIHAT BAHWA SEBAGIAN BESAR TITIK DATA BERADA DI SEKITAR GARIS IDEAL (MERAH), YANG MENANDAKAN MODEL SUDAH CUKUP AKURAT DALAM MEMPREDIKSI HARGA. MESKIPUN BEGITU, MASIH ADA BEBERAPA PREDIKSI YANG MELENCENG, YANG MUNGKIN DISEBABKAN OLEH FAKTOR-FAKTOR LAIN YANG BELUM TERWAKILI DI DALAM DATA.



KESIMPULAN

MODEL REGRESI LINEAR BERHASIL DIBUAT UNTUK MEMPERKIRAKAN HARGA MOBIL BEKAS BERDASARKAN BERBAGAI FITUR YANG ADA. HASIL MENUNJUKKAN BAHWA DAYA MESIN DAN VOLUME MESIN MERUPAKAN FITUR YANG PALING BERPENGARUH TERHADAP HARGA MOBIL. PROSES MEMBERSIKAN DATA, TERUTAMA MENGHAPUS 29,14% DATA YANG MERUPAKAN OUTLIER, MENJADI LANGKAH PENTING DALAM MEMERSIAPKAN DATASET. SECARA KESELURUHAN, MODEL MENUNJUKKAN HASIL YANG CUKUP BAIK DAN MAMPU MEMBERIKAN PERKIRAAN HARGA YANG AKURAT.



THANKS!