

LINEAR DAN POLYNOMIAL REGRESSION

disusun untuk memenuhi
tugas mata kuliah Pembelajaran Mesin

oleh :

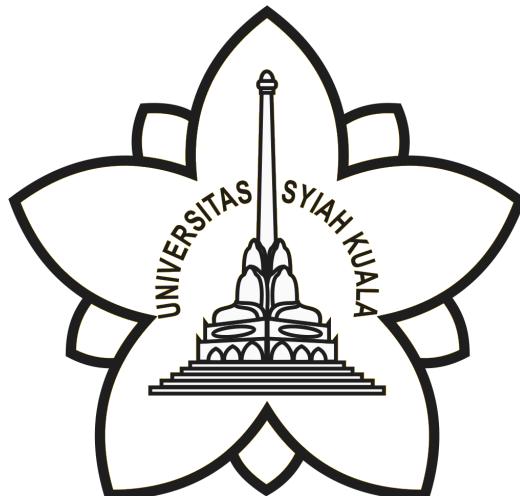
Alvia Zuhra (2208107010003)

Al-Mahfuzh Fadhlur Rohman (2208107010016)

Nurul Uzratun Nashriyyah (2208107010030)

Ganang Setyo Hadi (2208107010052)

Azimah Al-Huda (2208107010069)



**JURUSAN INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SYIAH KUALA
2025**

DAFTAR ISI

DAFTAR ISI.....	2
A. PEMAHAMAN DATASET.....	3
B. EKSPLORASI DATA DAN PRA-PEMROSESAN.....	8
C. IMPLEMENTASI MODEL.....	10
D. EVALUASI MODEL.....	12
E. ANALISIS HASIL.....	13

A. PEMAHAMAN DATASET

Sumber Dataset

Dataset yang digunakan adalah Cars4u Dataset yang diperoleh dari platform Kaggle, disediakan oleh Sukhmani Bedi, pada tahun 2021. Berikut adalah link Dataset: <https://www.kaggle.com/datasets/sukhmanibedi/cars4u/data>

Deskripsi Singkat tentang Dataset

Dataset ini merupakan kumpulan data mobil bekas yang mencakup berbagai merek, model, spesifikasi teknis, dan harga jual. Data ini sangat berharga untuk analisis pasar mobil bekas di India dan dapat digunakan untuk memprediksi harga mobil berdasarkan berbagai faktor.

Jumlah Data

- Jumlah sampel: 7253 observasi
- Jumlah fitur: 12 fitur prediktor (S.No, Name, Location, Year, Kilometers_Driven, Fuel_Type, Transmission, Owner_Type, Mileage, Engine, Power, Seats)
- Label: 2 kolom target (New_Price dan Price)

Format Data

Dataset tersedia dalam format CSV (Comma Separated Values), yang merupakan format standar untuk data tabular yang mudah diproses menggunakan berbagai library pemrograman seperti Pandas dalam Python.

Variabel yang Digunakan

No.	Nama Variabel	Tipe Data	Deskripsi
1	S.No.	Integer	Nomor urut data sebagai pengidentifikasi unik setiap entri
2	Name	String	Nama mobil yang mencakup merek dan model
3	Location	String	Lokasi di mana mobil dijual atau tersedia untuk dibeli
4	Year	Integer	Tahun pembuatan mobil
5	Kilometers_Driven	Integer	Total kilometer yang telah ditempuh mobil (dalam KM)
6	Fuel_Type	String	Jenis bahan bakar (Petrol, Diesel, Electric, CNG, LPG)
7	Transmission	String	Jenis transmisi (Otomatis/Manual)
8	Owner_Type	String	Tipe kepemilikan mobil
9	Mileage	String	Efisiensi bahan bakar dalam kmpl atau km/kg
10	Engine	String	Volume mesin dalam satuan CC
11	Power	String	Tenaga maksimum mesin dalam satuan bhp
12	Seats	Float	Jumlah kursi dalam mobil
13	New_Price	String	Harga mobil baru dengan model yang sama (dalam Lakh Rupee)
14	Price	Float	Harga mobil bekas (dalam Lakh Rupee, 1 Lakh = 100.000)

Statistik Deskriptif dan Visualisasi Awal

1. Melihat 5 baris pertama Dataset

df.head()																
S.No.	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	New_Price	Price			
0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	First	26.6 km/kg	998 CC	58.16 bhp	5.00000	NaN	1.75000			
1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.67 kmpl	1582 CC	126.2 bhp	5.00000	NaN	12.50000			
2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	18.2 kmpl	1199 CC	88.7 bhp	5.00000	8.61 Lakh	4.50000			
3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.77 kmpl	1248 CC	88.76 bhp	7.00000	NaN	6.00000			
4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15.2 kmpl	1968 CC	140.8 bhp	5.00000	NaN	17.74000			

2. Informasi Dasar Dataset

Informasi Dasar Dataset:	
Jumlah Baris: 7253	
Jumlah Kolom: 14	
Ukuran Memory: 0.77 MB	
Tipe Data Masing-masing Kolom	
S.No. int64	
Name object	
Location object	
Year int64	
Kilometers_Driven int64	
Fuel_Type object	
Transmission object	
Owner_Type object	
Mileage object	
Engine object	
Power object	
Seats float64	
New_Price object	
Price float64	

Dataset ini terdiri dari 7.253 baris dan 14 kolom, dengan total penggunaan memori sekitar 0,77 MB. Terdapat kombinasi tipe data numerik dan kategorikal di dalamnya. Kolom numerik seperti S.No., Year, Kilometers_Driven, dan Seats menggunakan tipe int64 atau float64, sementara sebagian besar kolom lainnya seperti Name, Location, Fuel_Type, Transmission, dan Owner_Type bertipe object, yang menunjukkan data kategorikal atau teks. Kolom-kolom seperti Mileage, Engine, Power, New_Price, dan Price meskipun berisi angka, saat ini masih bertipe object karena format penulisannya, sehingga perlu diproses lebih lanjut agar dapat digunakan dalam analisis kuantitatif.

3. Memperbaiki Struktur Data

- Mengonversi kolom Mileage, Engine, dan Power menjadi numerik karena saat ini masih dalam format teks yang mengandung satuan seperti “kmpl”, “CC”, dan “bhp”.
- Membersihkan dan mengonversi kolom New_Price yang masih bertipe object dan mengandung satuan “Lakh”, sehingga perlu diubah menjadi float.
- Memastikan kolom Seats tidak mengandung nilai kosong dan sudah sesuai dalam format float.

4. Statistik Deskriptif untuk Kolom Numerik

Statistik Deskriptif untuk Kolom Numerik:

	count	mean	std	min	25%	50%	75%	max	range
New_Price	1005.00000	2278884.57711	2777164.54159	391000.00000	788000.00000	1156000.00000	2614000.00000	3750000.00000	37109000.00000
Kilometers_Driven	7252.00000	58700.26269	84433.48037	171.00000	34000.00000	53429.00000	73000.00000	6500000.00000	6499829.00000
S.No.	7252.00000	3625.88693	2094.02732	0.00000	1812.75000	3625.50000	5439.25000	7252.00000	7252.00000
Engine	7206.00000	1616.78782	595.04824	624.00000	1198.00000	1493.00000	1968.00000	5998.00000	5374.00000
Power	7077.00000	112.77535	53.49053	34.20000	75.00000	94.00000	138.10000	616.00000	581.80000
Price	6018.00000	9.47888	11.18875	0.44000	3.50000	5.64000	9.95000	160.00000	159.56000
Mileage	7170.00000	18.34653	4.15791	6.40000	15.30000	18.20000	21.10000	33.54000	27.14000
Year	7252.00000	2013.36500	3.25450	1996.00000	2011.00000	2014.00000	2016.00000	2019.00000	23.00000
Seats	7199.00000	5.27976	0.81171	0.00000	5.00000	5.00000	5.00000	10.00000	10.00000

Jumlah Nilai Unik pada Kolom Numerik:
S.No.: 7252 nilai unik
Year: 23 nilai unik
Kilometers_Driven: 3660 nilai unik
Mileage: 437 nilai unik
Engine: 149 nilai unik
Power: 382 nilai unik
Seats: 9 nilai unik
New_Price: 625 nilai unik
Price: 1373 nilai unik

Dataset Cars4u menunjukkan variasi besar pada fitur numerik, dengan beberapa kolom seperti Price, New_Price, Kilometers_Driven, dan Power memiliki nilai ekstrem yang mengindikasikan outlier atau kemungkinan error input. Kolom New_Price memiliki banyak missing values, sementara skala antar fitur seperti Mileage, Engine, dan Power sangat berbeda, menandakan perlunya normalisasi sebelum modeling. Rata-rata mobil diproduksi sekitar tahun 2013, dan mayoritas memiliki 5 kursi.

5. Statistik Deskriptif untuk Kolom Kategorikal

Analisis data kategorikal menunjukkan bahwa pasar mobil bekas di India sangat beragam, dengan lebih dari 2000 model unik, didominasi oleh Mahindra XUV500. Penjualan tertinggi terjadi di kota besar seperti Mumbai dan Hyderabad, dengan transmisi manual masih mendominasi. Mayoritas mobil berasal dari pemilik pertama (82%) yang lebih dipercaya kualitasnya. Dari sisi bahan bakar, pasar masih sangat bergantung pada diesel dan petrol, sementara mobil listrik belum signifikan digunakan.

Statistik Deskriptif untuk Kolom Kategorikal:

Name: 2040 nilai unik

	Name	Jumlah	Persentase
0	Mahindra XUV500 W8 2WD	55	0.75841
1	Maruti Swift VDI	49	0.67568
2	Maruti Swift Dzire VDI	42	0.57915
3	Honda City 1.5 S MT	39	0.53778
4	Maruti Swift VDI BSIV	37	0.51020
5	Maruti Ritz VDi	35	0.48263
6	Toyota Fortuner 3.0 Diesel	35	0.48263
7	Honda City 1.5 V MT	32	0.44126
8	Honda Amaze S i-Dtech	32	0.44126
9	Honda Brio S MT	32	0.44126

Location: 11 nilai unik

	Location	Jumlah	Persentase
0	Mumbai	949	13.08605
1	Hyderabad	876	12.07943
2	Coimbatore	772	10.64534
3	Kochi	772	10.64534
4	Pune	765	10.54881
5	Delhi	660	9.10094
6	Kolkata	654	9.01820
7	Chennai	590	8.13569
8	Jaipur	499	6.88086
9	Bangalore	440	6.06729
10	Ahmedabad	275	3.79206

Fuel_Type: 5 nilai unik

	Fuel_Type	Jumlah	Persentase
0	Diesel	3852	53.11638
1	Petrol	3325	45.84942
2	CNG	62	0.85494
3	LPG	12	0.16547
4	Electric	1	0.01379

Transmission: 2 nilai unik

	Transmission	Jumlah	Persentase
0	Manual	5204	71.75951
1	Automatic	2048	28.24049

Owner_Type: 4 nilai unik

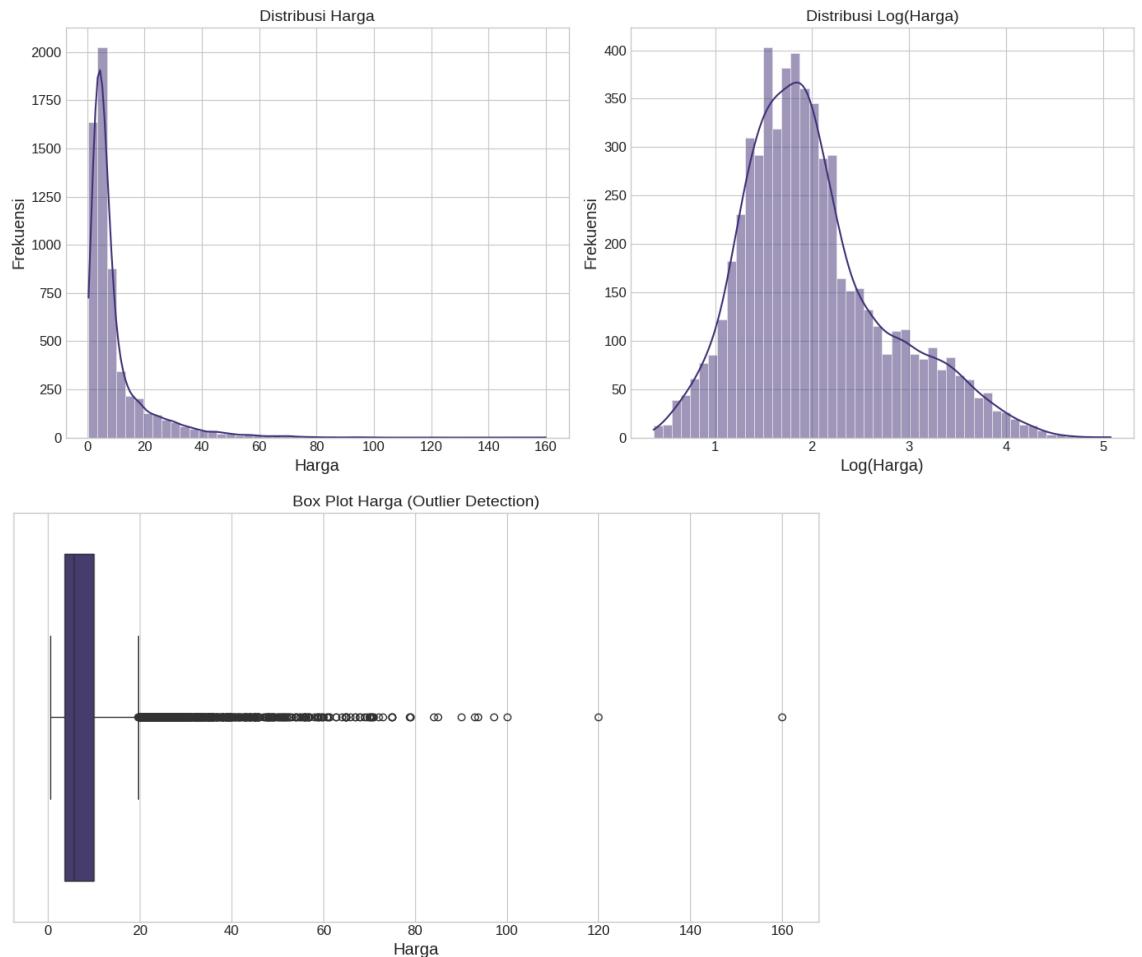
	Owner_Type	Jumlah	Persentase
0	First	5951	82.06012
1	Second	1152	15.88527
2	Third	137	1.88913
3	Fourth & Above	12	0.16547

6. Analisis Missing Values

Analisis Missing Values:			
	Jumlah	Missing	Persentase (%)
New_Price	6247	86.14175	
Price	1234	17.01600	
Power	175	2.41313	
Mileage	82	1.13072	
Seats	53	0.73083	
Engine	46	0.63431	

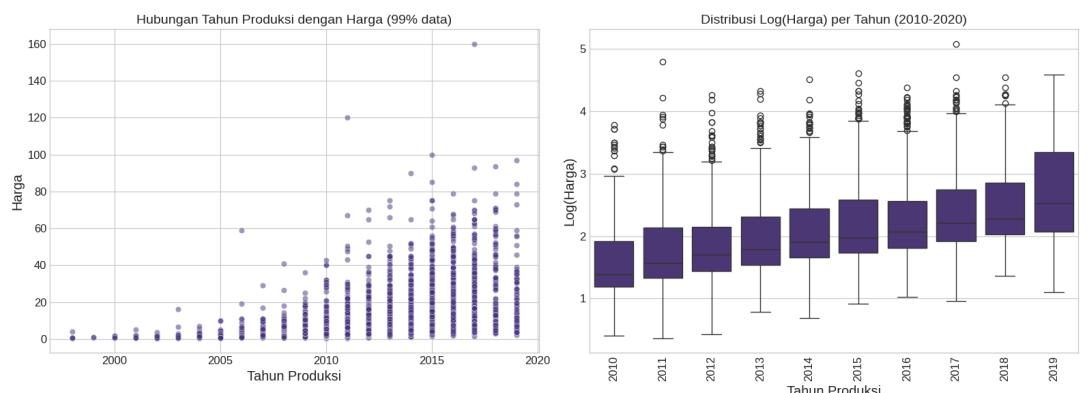
Hasil analisis menunjukkan kolom New_Price memiliki missing values paling banyak, yaitu 6.247 data atau 86,14%, sehingga perlu dipertimbangkan untuk dihapus atau diimputasi. Kolom Price juga cukup signifikan dengan 1.234 data hilang (17,02%). Sementara itu, kolom Power (2,41%), Mileage (1,13%), Seats (0,73%), dan Engine (0,63%) memiliki persentase missing values yang lebih rendah, namun tetap perlu ditangani agar tidak memengaruhi hasil analisis.

7. Analisis Distribusi Harga (Variabel Target)



Distribusi harga mobil bekas menunjukkan pola skewed ke kanan dengan banyak harga berada di level rendah dan sedikit outlier dengan harga sangat tinggi. Setelah dilakukan transformasi logaritmik, distribusi harga menjadi lebih simetris, yang lebih ideal untuk pemodelan. Visualisasi dengan box plot juga menegaskan keberadaan outlier signifikan di rentang harga atas. Analisis ini penting untuk memutuskan apakah outlier perlu dihapus atau ditangani agar model prediktif dapat bekerja lebih akurat dan stabil.

8. Harga Produksi Berdasarkan Tahun



Analisis ini bertujuan memahami distribusi harga mobil dengan fokus pada pengaruh tahun produksi. Scatter plot menunjukkan tren positif: mobil yang lebih baru cenderung memiliki harga lebih tinggi, dengan variasi harga yang semakin lebar di tahun-tahun terbaru. Box plot logaritmik memperkuat temuan ini, memperlihatkan kenaikan median harga dari 2010 hingga 2019, meski masih ada outlier. Sebagian besar mobil berasal dari tahun-tahun terbaru, mencerminkan tren pasar. Secara keseluruhan, tahun produksi terbukti menjadi faktor penting dalam menentukan harga mobil, di mana mobil baru umumnya lebih mahal sebelum mengalami depresiasi.

B. EKSPLORASI DATA DAN PRA-PEMROSESAN

1. Memperbaiki Nilai 'Seats' yang Tidak Valid (0.0) Menjadi NaN

```
df.query("Seats == 0.0")
df.loc[3999, 'Seats'] = np.nan
```

Perintah df.query("Seats == 0.0") digunakan untuk mencari baris dengan nilai Seats nol, lalu df.loc[3999, 'Seats'] = np.nan mengganti nilai tersebut menjadi NaN agar dianggap sebagai data hilang dan bisa diproses lebih lanjut dalam pembersihan data.

2. Mengonversi Tipe Data

```
df["Fuel_Type"] = df["Fuel_Type"].astype("category")
df["Transmission"] = df["Transmission"].astype("category")
df["Owner_Type"] = df["Owner_Type"].astype("category")
```

Perintah tersebut mengubah tipe data kolom Fuel_Type, Transmission, dan Owner_Type menjadi tipe kategori. Ini dilakukan untuk mengoptimalkan penggunaan memori dan mempermudah analisis atau pemrosesan data kategorikal dalam model machine learning.

3. Membuat Fitur Baru yaitu Usia Mobil

```
df['Age'] = 2025 - df['Year']
```

Perintah tersebut digunakan untuk membuat kolom baru bernama Age dengan cara menghitung usia mobil, yaitu dengan mengurangi tahun referensi 2025 dengan nilai pada kolom Year. Tujuannya agar model dapat memahami seberapa tua kendaraan tersebut, yang bisa memengaruhi harga mobil bekas.

4. Mengekstrak Merek dan Model dari Kolom 'Name'

```
df['Brand'] = df['Name'].str.split(' ').str[0] #Separating Brand name from the Name  
df['Model'] = df['Name'].str.split(' ').str[1] + df['Name'].str.split(' ').str[2]
```

Perintah ini memisahkan informasi merek dan model mobil dari kolom Name. Kolom Brand diisi dengan kata pertama dari Name sebagai merek mobil, sedangkan kolom Model menggabungkan kata kedua dan ketiga dari Name untuk membentuk nama model mobil. Ini dilakukan agar data lebih terstruktur dan mudah dianalisis.

5. Melakukan Standarisasi Nama Merek

```
df.loc[df.Brand == 'ISUZU','Brand']='Isuzu'  
df.loc[df.Brand=='Mini','Brand']='Mini Cooper'  
df.loc[df.Brand=='Land','Brand']='Land Rover'
```

Perintah ini melakukan pembersihan data pada kolom Brand dengan memperbaiki penulisan merek mobil. Jika merek bertuliskan 'ISUZU', akan diubah menjadi 'Isuzu', 'Mini' menjadi 'Mini Cooper', dan 'Land' menjadi 'Land Rover'. Tujuannya agar penulisan nama merek konsisten dan rapi untuk analisis data.

6. Menghapus Mobil yang Tidak Memiliki Model

```
df.dropna(subset=['Model'],axis=0,inplace=True)
```

Perintah ini digunakan untuk menghapus baris-baris dalam DataFrame yang memiliki nilai kosong (NaN) pada kolom Model. Dengan axis=0, yang dihapus adalah baris, dan inplace=True berarti perubahan dilakukan langsung pada DataFrame tanpa membuat salinan baru. Ini membantu memastikan data pada kolom Model bersih dan lengkap untuk analisis.

7. Menangani Missing Value

```
df['Engine'] = df['Engine'].fillna(df.groupby(['Name', 'Year'])['Engine'].transform('median'))  
df['Power'] = df['Power'].fillna(df.groupby(['Name', 'Year'])['Power'].transform('median'))  
df['Mileage'] = df['Mileage'].fillna(df.groupby(['Name', 'Year'])['Mileage'].transform('median'))  
df['Engine'] = df['Engine'].fillna(df.groupby(['Brand', 'Model'])['Engine'].transform('median'))  
df['Power']= df['Power'].fillna(df.groupby(['Brand', 'Model'])['Power'].transform('median'))  
df['Mileage'] = df['Mileage'].fillna(df.groupby(['Brand', 'Model'])['Mileage'].transform('median'))  
df['Seats'] = df.groupby('Name')[['Seats']].transform(lambda x: x.fillna(x.median()))  
df['Seats'] = df.groupby('Model')[['Seats']].transform(lambda x: x.fillna(x.median()))  
df['Seats']=df['Seats'].fillna(5)  
df['New_Price'] = df['New_Price'].fillna(df.groupby(['Name','Year'])['New_Price'].transform('median'))  
df['New_Price'] = df['New_Price'].fillna(df.groupby(['Name'])['New_Price'].transform('median'))  
df['New_Price'] = df['New_Price'].fillna(df.groupby(['Brand', 'Model'])['New_Price'].transform('median'))  
df['New_Price'] = df['New_Price'].fillna(df.groupby(['Brand'])['New_Price'].transform('median'))  
df.dropna(inplace=True,axis=0)
```

Kode tersebut bertujuan untuk menangani missing values (nilai kosong) pada beberapa kolom penting di dataset seperti 'Engine', 'Power', 'Mileage', 'Seats', dan 'New_Price'. Caranya, kode ini mengisi nilai kosong dengan median berdasarkan pengelompokan tertentu agar lebih kontekstual. Awalnya, pengisian dilakukan dengan median berdasarkan kombinasi 'Name' dan 'Year', lalu jika masih ada yang kosong, dilanjutkan dengan median berdasarkan 'Brand' dan 'Model'. Untuk kolom 'Seats', jika masih kosong setelah pengelompokan, diisi dengan median umum atau default angka 5. Terakhir, setelah semua pengisian selesai, kode menggunakan df.dropna(inplace=True, axis=0) untuk menghapus baris-baris yang masih memiliki nilai kosong agar dataset menjadi bersih sepenuhnya sebelum diproses lebih lanjut.

8. Klasifikasi Merek Berdasarkan Kelas Harga

```
Low=['Maruti',
      'Hyundai',
      'Ambassdor',
      'Hindustan',
      'Force',
      'Chevrolet',
      'Fiat',
      'Tata',
      'Smart',
      'Renault',
      'Datsun',
      'Mahindra',
      'Skoda',
      'Ford',
      'Toyota',
      'Isuzu',
      'Mitsubishi', 'Honda']

High=['Audi',
      'Mini Cooper',
      'Bentley',
      'Mercedes-Benz',
      'Lamborghini',
      'Volkswagen',
      'Porsche',
      'Land Rover',
      'Nissan',
      'Volvo',
      'Jeep',
      'Jaguar',
      'BMW']|
```

Kode tersebut bertujuan untuk mengelompokkan merek mobil ke dalam dua kategori berdasarkan pengetahuan bisnis, yaitu "Low" untuk mobil dengan harga lebih terjangkau dan "High" untuk mobil premium atau mewah dengan harga di atas 30 lakh. Daftar Low berisi merek-merek seperti Maruti, Hyundai,

Honda, dan lain-lain yang dikenal sebagai mobil massal dengan harga ekonomis, sementara daftar High mencakup merek-merek seperti BMW, Mercedes-Benz, dan Audi yang dikenal sebagai mobil premium dengan harga lebih tinggi. Kategori ini nantinya bisa digunakan untuk analisis lebih lanjut, seperti segmentasi pasar atau prediksi harga.

```
def classrange(x):
    if x in Low:
        return "Low"
    elif x in High:
        return "High"
    else:
        return x

df['Brand_Class'] = df['Brand'].apply(lambda x: classrange(x))
```

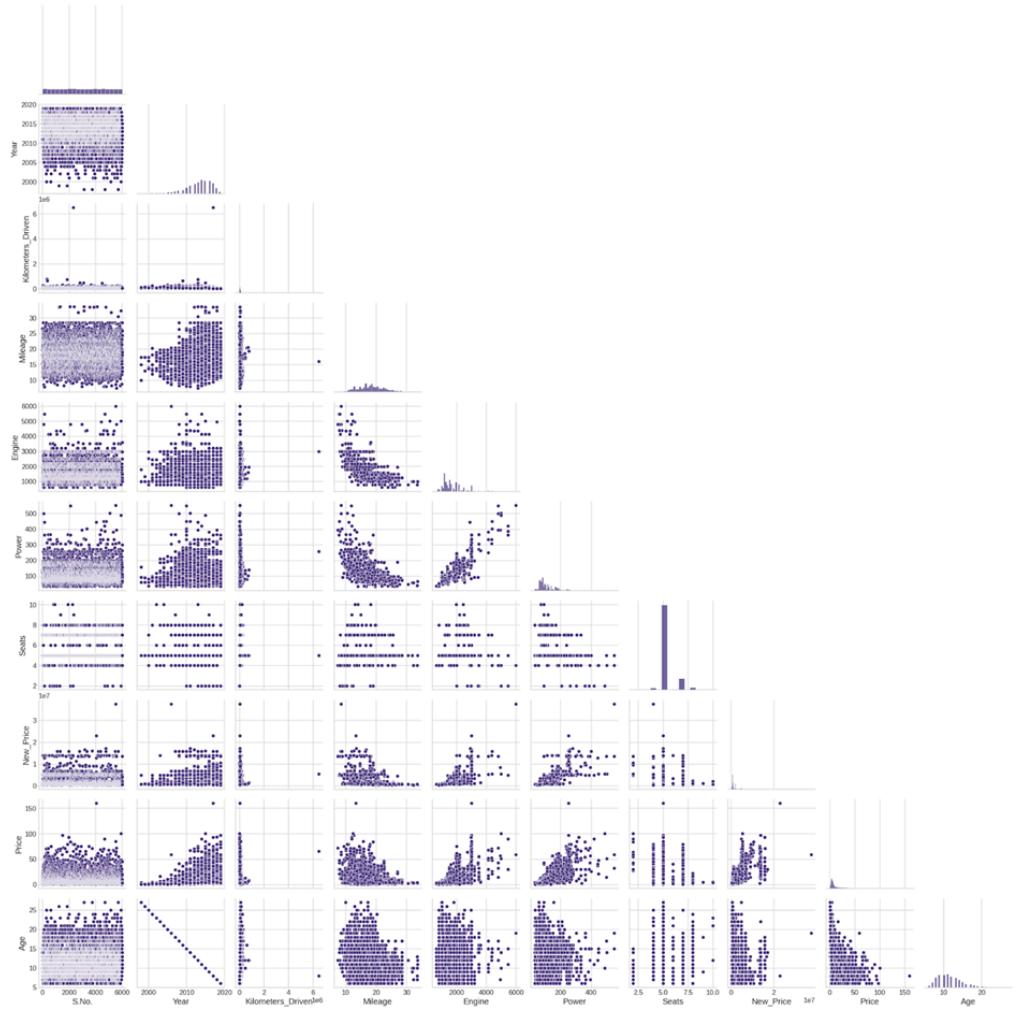
9. Perubahan Tipe Data pada Kolom ‘Engine’ dan ‘Brand_Class’

```
df['Engine']=df['Engine'].astype(int)
df['Brand_Class']=df["Brand_Class"].astype('category')
```

Kode tersebut melakukan konversi tipe data pada dua kolom dalam DataFrame. Kolom Engine diubah menjadi tipe data integer agar nilainya berupa angka bulat, sehingga lebih sesuai untuk analisis numerik. Sementara itu, kolom Brand_Class dikonversi menjadi tipe data kategori untuk menghemat memori dan mempermudah analisis data kategorikal seperti segmentasi atau visualisasi.

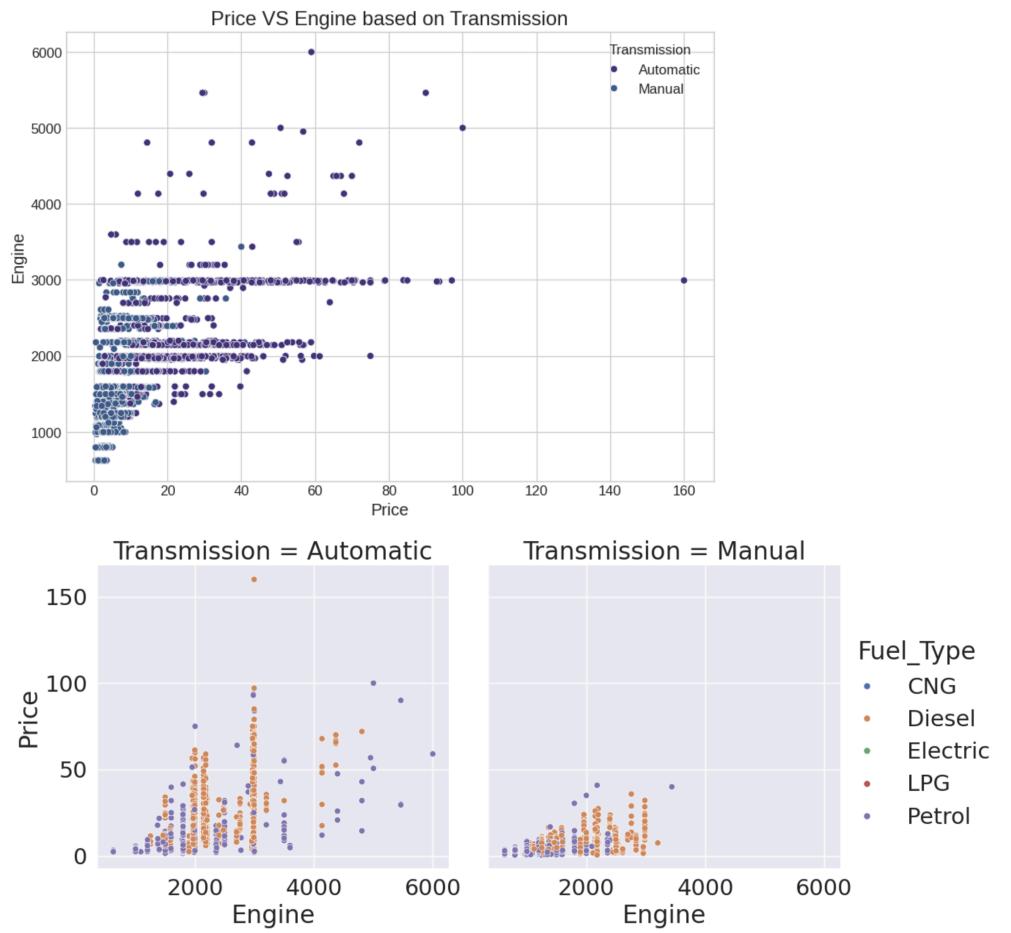
10. Analisis Korelasi

- Pairplot



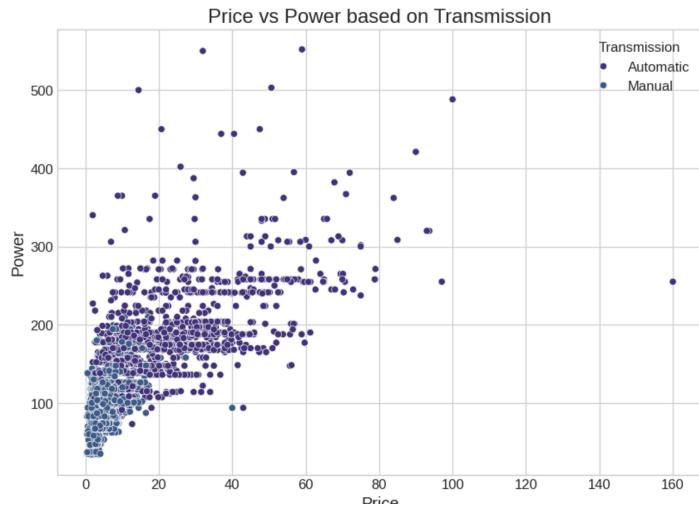
Gambar yang ditampilkan adalah pairplot yang memperlihatkan hubungan antar variabel numerik dalam dataset mobil. Dari visualisasi ini, terlihat beberapa pola menarik. Pertama, terdapat hubungan negatif yang jelas antara kilometers driven dan price, yang artinya semakin banyak jarak tempuh mobil, harga cenderung lebih rendah. Sebaliknya, engine capacity, power, dan new price menunjukkan korelasi positif dengan harga mobil semakin besar kapasitas mesin atau tenaga mobil, harga cenderung lebih tinggi. Selain itu, fitur age, yang merupakan hasil feature engineering dari tahun produksi, juga memperlihatkan tren negatif dengan harga: semakin tua usia mobil, harga semakin turun. Distribusi dari masing-masing variabel juga terlihat di diagonal plot, yang memberikan gambaran awal sebaran data, seperti distribusi mileage yang agak miring ke kanan, atau sebaran price yang tampak memiliki outlier ke arah harga tinggi. Secara keseluruhan, visualisasi ini membantu mengidentifikasi pola hubungan dan potensi korelasi antar fitur, yang nantinya bisa dimanfaatkan dalam proses pemodelan regresi.

- Harga vs Mesin



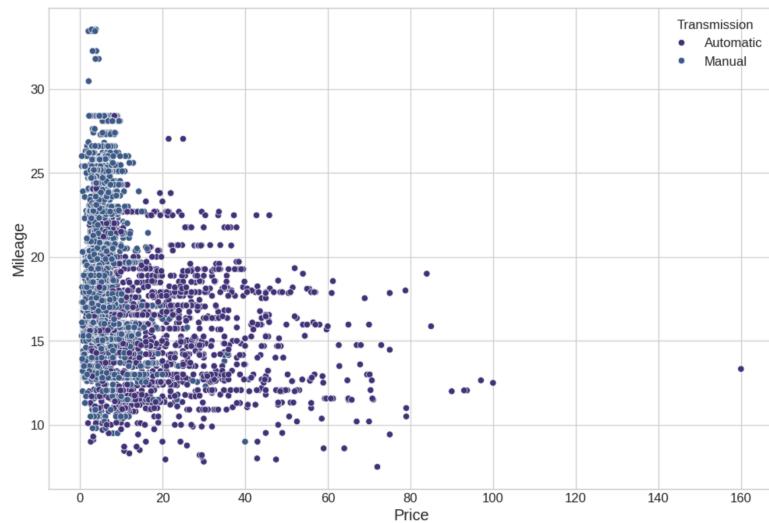
Gambar ini menunjukkan hubungan antara harga mobil dan kapasitas mesin berdasarkan jenis transmisi dan bahan bakar. Plot atas memperlihatkan bahwa mobil dengan transmisi otomatis maupun manual cenderung memiliki mesin kecil dengan harga yang bervariasi, namun mobil otomatis tampak memiliki rentang harga yang lebih luas. Pada dua plot bawah yang dipisahkan berdasarkan transmisi, terlihat bahwa mobil otomatis lebih banyak didominasi oleh bahan bakar diesel pada mesin besar, sementara mobil manual cenderung terkonsentrasi pada mesin dengan kapasitas lebih kecil dan harga lebih rendah, dengan variasi bahan bakar yang juga lebih beragam. Visualisasi ini membantu memahami bagaimana spesifikasi teknis memengaruhi harga mobil.

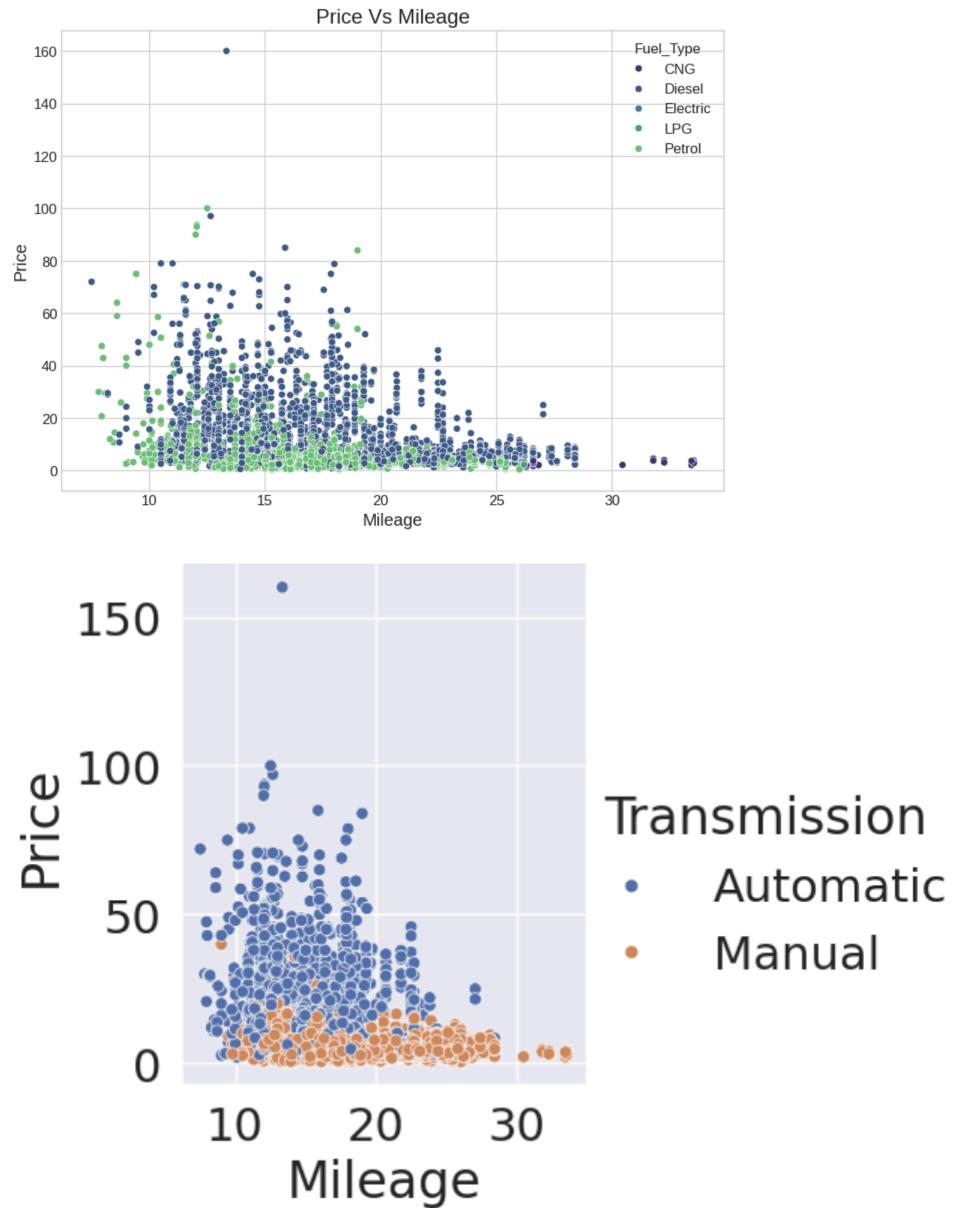
- Harga vs Power



Gambar ini menunjukkan hubungan antara harga mobil dengan tenaga mesin (power) berdasarkan jenis transmisi. Terlihat bahwa semakin tinggi tenaga mesin, cenderung harga mobil juga meningkat, walaupun sebarannya cukup luas. Mobil dengan transmisi otomatis dan manual memiliki pola yang mirip, tetapi mobil otomatis tampak sedikit lebih tersebar di rentang harga yang lebih tinggi. Mayoritas mobil dengan tenaga di bawah 200 cenderung berada pada kisaran harga yang lebih rendah, sedangkan mobil dengan tenaga di atas 300 biasanya masuk kategori harga yang jauh lebih mahal. Visualisasi ini membantu memahami bahwa tenaga mesin adalah salah satu faktor yang memengaruhi harga mobil.

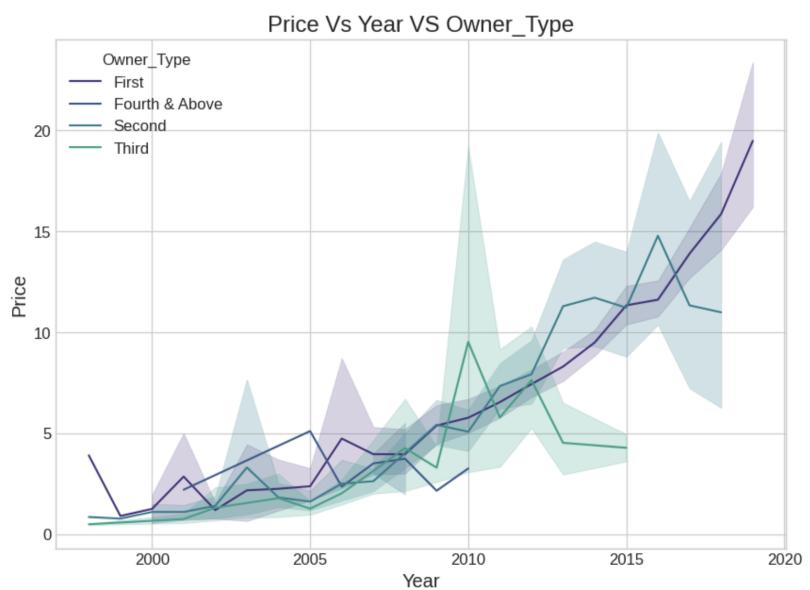
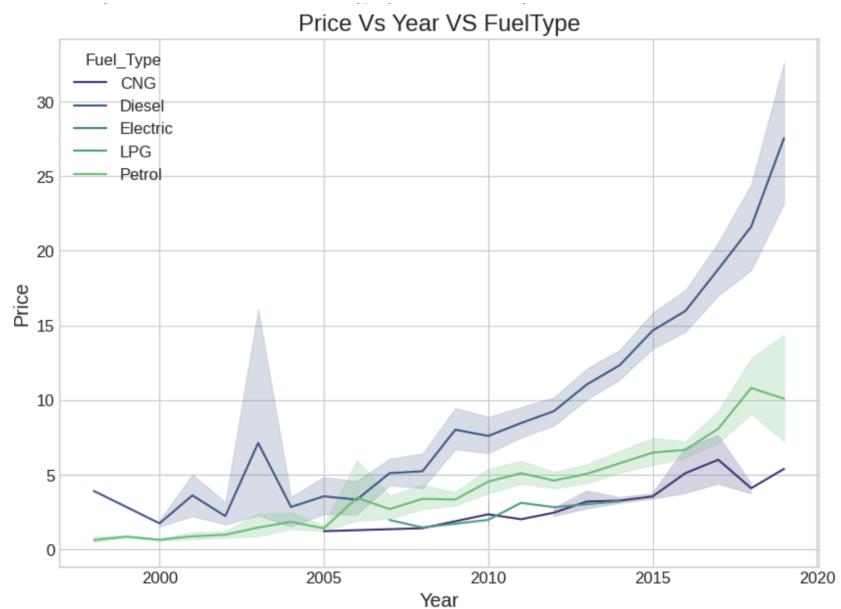
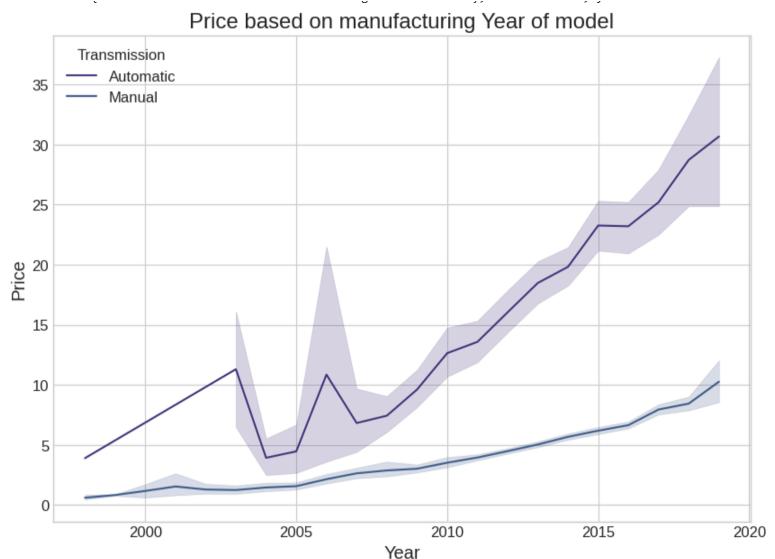
- Harga vs Jarak Tempuh Kendaraan

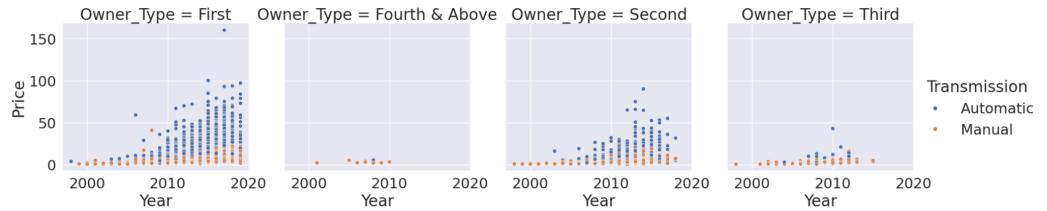




Visualisasi menunjukkan bahwa mobil dengan transmisi otomatis cenderung lebih mahal dibanding manual. Harga mobil umumnya menurun seiring bertambahnya mileage. Selain itu, mobil berbahan bakar diesel memiliki variasi harga yang lebih lebar dibanding bahan bakar lain.

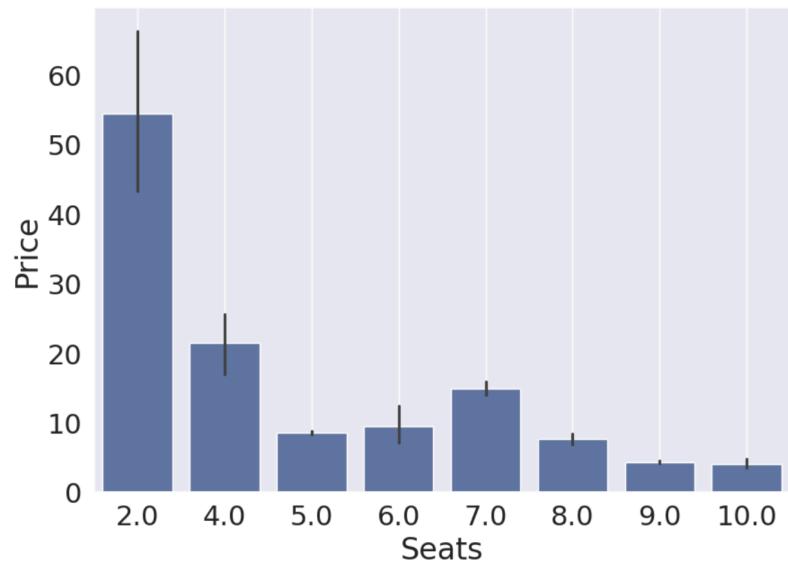
- Harga vs Tahun Produksi





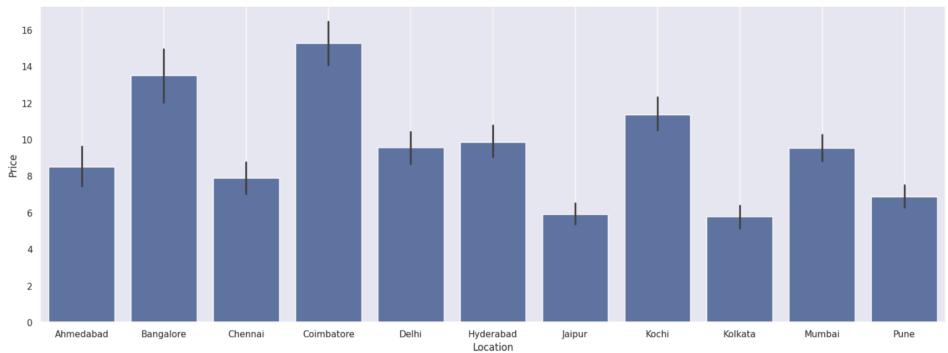
Grafik-grafik di atas menunjukkan hubungan antara harga mobil dengan tahun produksi berdasarkan berbagai faktor. Grafik pertama menunjukkan bahwa mobil transmisi otomatis cenderung memiliki harga lebih tinggi dibandingkan manual, terutama setelah tahun 2010. Grafik kedua menunjukkan bahwa mobil berbahan bakar diesel mengalami peningkatan harga paling signifikan dari waktu ke waktu dibandingkan tipe bahan bakar lain. Grafik ketiga memperlihatkan bahwa mobil milik pemilik pertama cenderung memiliki harga tertinggi, diikuti oleh pemilik kedua dan ketiga, sementara mobil dengan pemilik keempat atau lebih memiliki harga paling rendah. Grafik terakhir memperkuat hal ini, menunjukkan bahwa transmisi otomatis mendominasi harga tinggi terutama pada mobil milik pemilik pertama.

- Harga vs Tempat Duduk



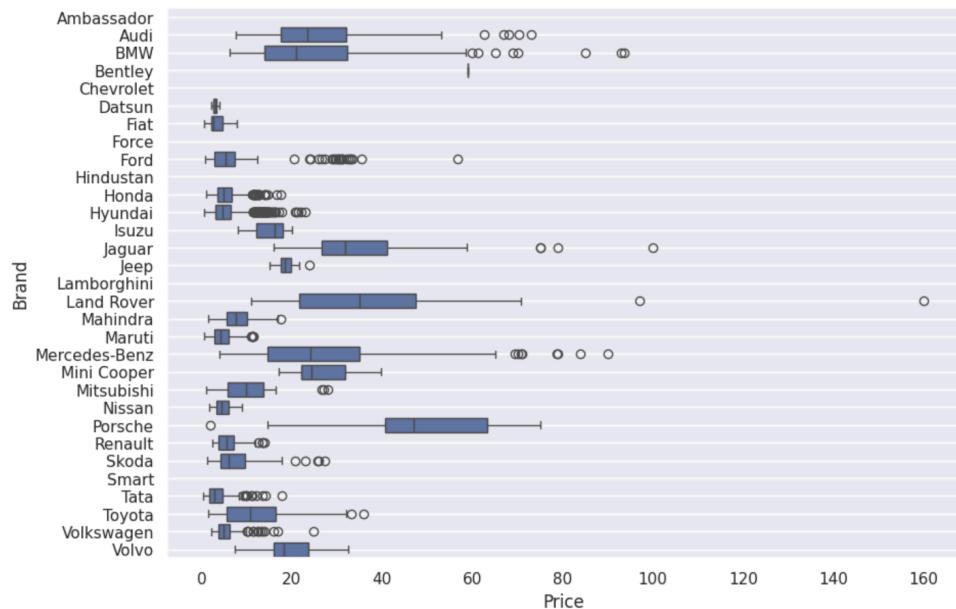
Grafik menunjukkan bahwa mobil dengan 2 kursi memiliki harga rata-rata paling tinggi, sedangkan mobil dengan jumlah kursi lebih banyak cenderung memiliki harga yang lebih rendah. Harga menurun seiring bertambahnya jumlah kursi.

- Harga vs Lokasi



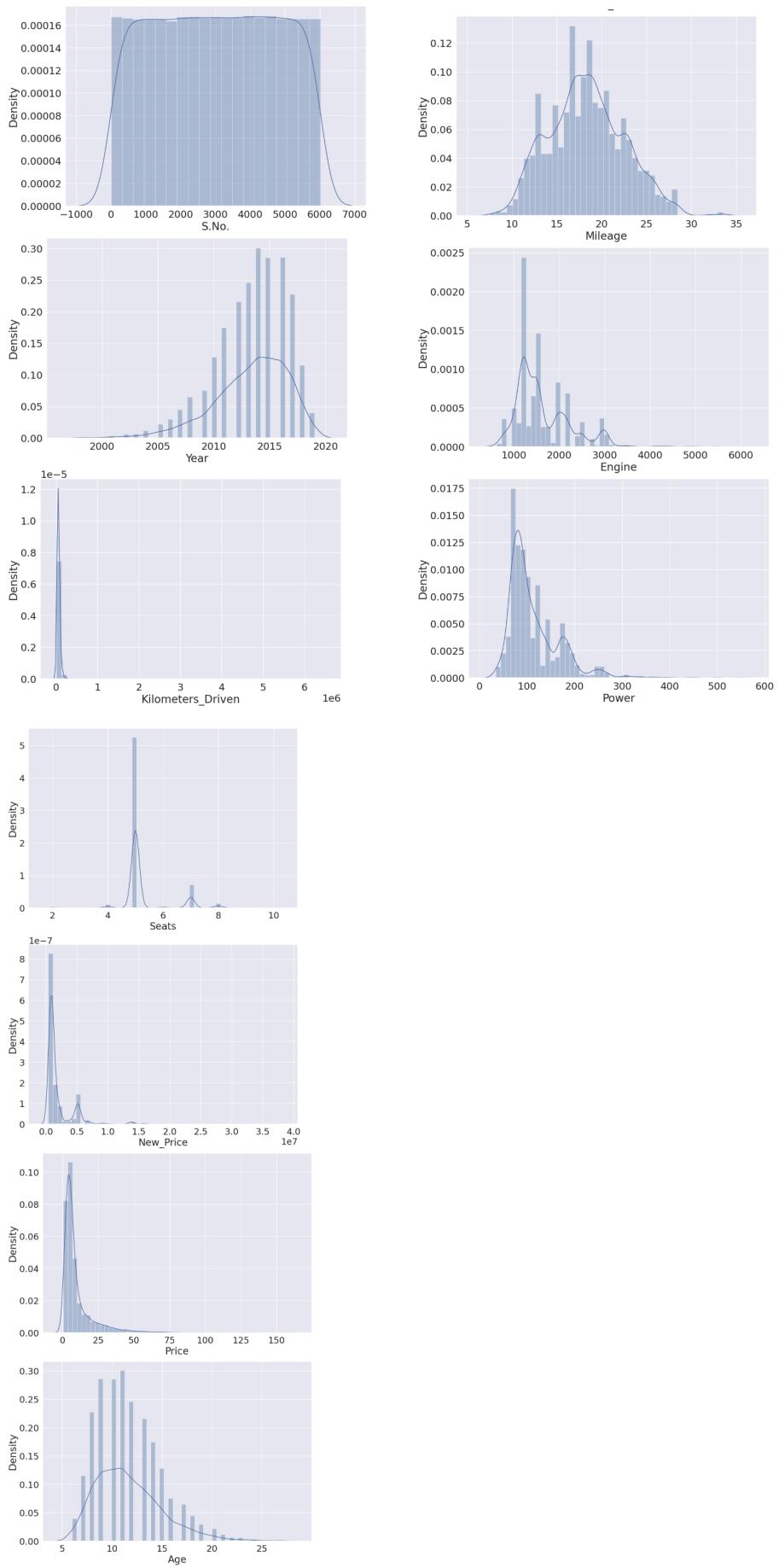
Grafik menunjukkan bahwa rata-rata harga mobil tertinggi ada di Coimbatore dan Bangalore, sementara Jaipur, Kolkata, dan Pune memiliki harga rata-rata mobil yang lebih rendah dibandingkan kota lainnya. Harga mobil bervariasi di tiap lokasi.

- Harga vs Merek



Gambar menunjukkan boxplot harga mobil berdasarkan merek. Merek seperti BMW dan Mercedes-Benz punya rentang harga luas dengan banyak outlier mahal, sedangkan Datsun dan Tata cenderung lebih murah dan stabil. Boxplot ini membantu melihat perbandingan harga dan sebarannya antar merek.

11. Menjelajahi Pola Distribusi



Gambar ini menunjukkan distribusi dari berbagai fitur dalam data mobil bekas. Sebagian besar mobil berasal dari tahun 2010 ke atas, dengan kepadatan tinggi di sekitar tahun 2015–2018. Distribusi mileage (konsumsi bahan bakar) berkisar antara 15 hingga 20 km/l, sedangkan engine (kapasitas mesin) dan power (tenaga mesin) menunjukkan banyak mobil dengan mesin dan tenaga di level rendah hingga menengah. Kilometers driven sebagian besar berkumpul di angka rendah, menunjukkan mobil dengan jarak tempuh rendah lebih umum. Harga mobil bekas (price) dan harga baru (new price) cenderung condong ke nilai rendah, dengan sebagian kecil yang sangat mahal. Jumlah kursi paling umum adalah 5, dan usia mobil (age) paling banyak berada di sekitar 5–10 tahun, mencerminkan pasar mobil bekas yang didominasi oleh mobil relatif baru.

C. IMPLEMENTASI MODEL

Membagi Data untuk Pengujian

```
x_train: (4123, 26)
x_test: (1768, 26)
y_train: (4123, 2)
y_test: (1768, 2)
```

Pembagian dataset dilakukan menggunakan fungsi `train_test_split`, dengan proporsi 70% untuk data latih dan 30% untuk data uji. Hasilnya, data fitur untuk pelatihan (`X_train`) terdiri dari 4.123 baris dan 26 kolom, sementara data uji (`X_test`) memiliki 1.768 baris dan 26 kolom. Untuk variabel target, baik data pelatihan (`y_train`) maupun data uji (`y_test`) masing-masing memiliki 2 kolom, yang menunjukkan bahwa target sudah melalui proses encoding sebelumnya. Pembagian ini bertujuan agar model dapat belajar dari sebagian besar data yang tersedia, sekaligus diuji pada data yang terpisah untuk mengevaluasi kinerjanya secara objektif.

Membangun Model

1. Membuat Model Regresi

Training data shape after cleaning: (4123, 27)						
OLS Regression Results						
Dep. Variable:	y	R-squared:	0.894			
Model:	OLS	Adj. R-squared:	0.893			
Method:	Least Squares	F-statistic:	1322.			
Date:	Mon, 07 Apr 2025	Prob (F-statistic):	0.00			
Time:	04:27:18	Log-Likelihood:	-655.29			
No. Observations:	4123	AIC:	1365.			
Df Residuals:	4096	BIC:	1535.			
Df Model:	26					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	3.1331	0.145	21.571	0.000	2.848	3.418
x1	-4.135e-07	1.99e-07	-2.082	0.037	-8.03e-07	-2.4e-08
x2	-0.0132	0.002	-6.691	0.000	-0.017	-0.009
x3	0.0002	2.22e-05	9.603	0.000	0.000	0.000
x4	0.0065	0.000	29.206	0.000	0.006	0.007
x5	0.0252	0.008	3.137	0.002	0.009	0.041
x6	-0.1116	0.002	-57.067	0.000	-0.115	-0.108
x7	-0.0571	0.011	-5.025	0.000	-0.079	-0.035
x8	0.1456	0.030	4.887	0.000	0.087	0.204
x9	0.0038	0.028	0.134	0.893	-0.051	0.058
...						

Setelah proses pembersihan data, dataset pelatihan terdiri dari 4.123 observasi dan 27 fitur. Hasil regresi linier menggunakan metode Ordinary Least Squares (OLS) menunjukkan bahwa model memiliki R-squared sebesar 0.894, yang berarti sekitar 89,4% variasi dalam variabel target dapat dijelaskan oleh variabel-variabel independen yang digunakan. Adjusted R-squared yang hampir sama (0.893) mengindikasikan model cukup baik meskipun ada banyak fitur. Nilai F-statistic sebesar 1322 dengan p-value 0.00 menegaskan bahwa model secara keseluruhan signifikan. Koefisien untuk setiap variabel independen, bersama dengan p-value-nya, memberikan gambaran kontribusi masing-masing fitur terhadap prediksi. Sebagian besar fitur memiliki p-value rendah (kurang dari 0.05), yang menunjukkan bahwa mereka berpengaruh signifikan terhadap target. Secara umum, hasil ini menunjukkan bahwa model linier yang dibangun sudah cukup kuat untuk menjelaskan hubungan antara fitur dengan target.

2. Membuat Model Regresi Polinomial

Final training data shape: (4123, 405)						
OLS Regression Results						
Dep. Variable:	y	R-squared:	0.929			
Model:	OLS	Adj. R-squared:	0.924			
Method:	Least Squares	F-statistic:	212.4			
Date:	Mon, 07 Apr 2025	Prob (F-statistic):	0.00			
Time:	21:12:50	Log-Likelihood:	169.13			
No. Observations:	4123	AIC:	139.7			
Df Residuals:	3884	BIC:	1651.			
Df Model:	238					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	0.3166	1.392	0.227	0.820	-2.412	3.046
x1	6.334e-05	9.43e-05	0.672	0.502	-0.000	0.000
x2	0.0076	0.031	0.248	0.804	-0.052	0.068
x3	0.0004	0.000	0.831	0.406	-0.001	0.001
x4	-0.0024	0.006	-0.424	0.671	-0.014	0.009
x5	0.1238	0.146	0.848	0.396	-0.162	0.410
x6	-0.0331	0.032	-1.041	0.298	-0.095	0.029
x7	0.1983	0.298	0.666	0.505	-0.385	0.782
x8	0.0890	0.279	0.319	0.749	-0.457	0.636
x9	0.3377	0.318	1.061	0.289	-0.287	0.962

Setelah menerapkan teknik polynomial regression untuk meningkatkan performa model, data pelatihan akhir memiliki dimensi sebesar (4123, 405), yang menunjukkan bahwa jumlah fitur meningkat signifikan akibat penambahan fitur polinomial. Hasil regresi dengan metode OLS menunjukkan peningkatan performa yang cukup baik dengan nilai R-squared sebesar 0.929, artinya sekitar 92,9% variasi harga mobil dapat dijelaskan oleh fitur-fitur dalam model ini. Adjusted R-squared juga tinggi, yaitu 0.924, menunjukkan bahwa model tetap kuat meski jumlah fitur meningkat. Nilai F-statistic sebesar 212.4 dengan p-value 0.00 menandakan bahwa model secara keseluruhan signifikan secara statistik.

Namun, meskipun performa model meningkat secara umum, beberapa koefisien variabel dalam hasil regresi menunjukkan nilai p-value yang tinggi (di atas 0.05), yang berarti tidak semua fitur yang ditambahkan melalui polinomial memberikan kontribusi signifikan. Ini bisa menjadi indikasi adanya overfitting atau fitur yang kurang relevan. Oleh karena itu, meskipun model polynomial ini lebih kompleks dan akurasinya meningkat, perlu dipertimbangkan untuk melakukan regularisasi atau feature selection agar model lebih optimal dan tidak terlalu kompleks.

D. EVALUASI MODEL

Evaluasi Kinerja Model

Pada model Regresi Polinomial, model menunjukkan akurasi yang sangat tinggi dengan R-squared sebesar 0.986, artinya sekitar 98.6% variasi target dapat dijelaskan oleh fitur input. Ini menandakan bahwa model sangat baik dalam menangkap hubungan non-linear dalam data. Namun, meskipun performa di data train sangat baik dengan RMSE sebesar 3.67, performa di data test sangat buruk dengan RMSE tak terhingga, MAE sekitar 1.24e+107, dan MAPE mencapai ~1.90e+107%. Ini jelas menunjukkan bahwa model mengalami overfitting sangat parah: sangat akurat di data latih, tetapi gagal total di data baru.

Sementara itu, pada model Regresi Linear, hasilnya lebih stabil. Model ini memiliki R-squared sebesar 0.894, yang berarti 89.4% variasi harga mobil bekas dapat dijelaskan oleh variabel input. Dari sisi performa, model ini cukup baik di data train dengan RMSE sebesar 6.50, MAE sebesar 2.28, dan MAPE sebesar 23.30%. Saat diuji di data test, performanya tetap konsisten dengan RMSE sebesar 7.70, MAE sebesar 2.56, dan MAPE sebesar 23.66%. Ini menunjukkan bahwa Regresi Linear lebih andal untuk generalisasi ke data baru dibandingkan Regresi Polinomial.

Perbandingan Model: Regresi Linear vs Regresi Polinomial

Aspek	Regresi Linear	Regresi Polinomial
▢ Akurasi (R^2)	0.894 (cukup tinggi)	0.986 (sangat tinggi)
⌚ Generalitas ke Data Baru	Baik – performa train dan test serupa	Buruk – overfitting berat pada data test
☒ RMSE	Train: 6.50 Test: 7.70	Train: 3.67 Test: ∞
☒ MAE	Train: 2.28 Test: 2.56	Train: 1.68 Test: ~1.24e+107
☒ MAPE	Train: 23.30% Test: 23.66%	Train: 18.12% Test: ~1.90e+107%
⚠ Overfitting	Tidak – perbedaan train-test kecil	Sangat parah – test error ekstrem
🧠 Kemampuan Tangkap Pola Non-Linear	Terbatas – hanya linear	Kuat – tangkap hubungan kompleks
✖ Catatan Khusus	Multikolinearitas terdeteksi (perlu cek VIF)	Perlu regularisasi & normalisasi
✓ Kesimpulan	Stabil dan andal untuk prediksi	Hanya cocok untuk data latih, tidak untuk generalisasi

Perbandingan antara Regresi Linear dan Regresi Polinomial menunjukkan bahwa Regresi Linear lebih stabil dan andal untuk prediksi. Dengan R^2 sebesar 0.894, performanya seimbang antara data train dan test, serta error yang masih wajar, meskipun ada indikasi multikolinearitas yang perlu diperiksa. Sebaliknya, Regresi Polinomial memang mencapai R^2 sangat tinggi (0.986) di data latih, tapi model ini mengalami overfitting parah dengan error ekstrem di data test. Jadi, Regresi Linear lebih cocok untuk generalisasi, sementara Regresi Polinomial hanya aman digunakan di data pelatihan dan memerlukan perbaikan sebelum dipakai secara luas.

E. ANALISIS HASIL

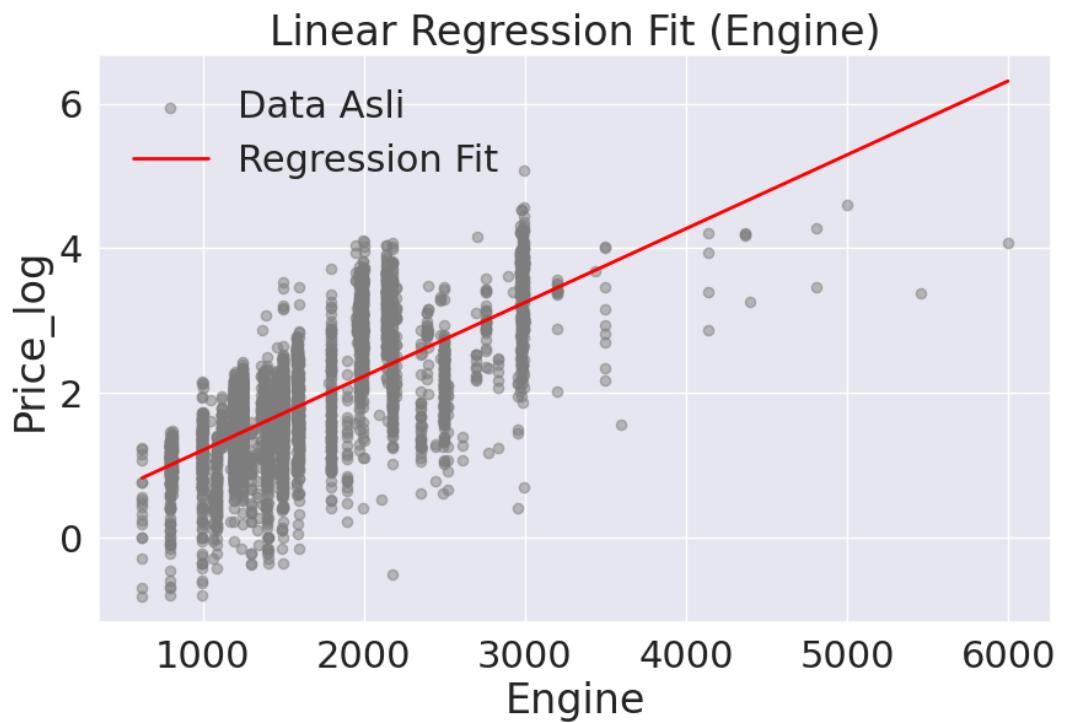
Interpretasi Koefisien Regresi

No.	Fitur	Koefisien	Signifikan ($p < 0.05$)?	Interpretasi
1	const	3.3549	<input checked="" type="checkbox"/> Ya	Nilai prediksi saat semua variabel independen bernilai nol.
2	Kilometers_Driven	-2.74e-07	<input type="checkbox"/> Tidak	Tidak signifikan, sedikit pengaruh terhadap harga.
3	Mileage	-0.0214	<input checked="" type="checkbox"/> Ya	Setiap kenaikan 1 km/l, harga turun ~2.14% (mungkin multikolinearitas).
4	Engine	0.0081	<input checked="" type="checkbox"/> Ya	Setiap kenaikan 1 cc mesin, harga naik ~0.81%.
5	Power	0.0503	<input checked="" type="checkbox"/> Ya	Setiap kenaikan 1 bhp, harga naik ~5.03%.
6	Seats	-0.1108	<input checked="" type="checkbox"/> Ya	Setiap tambahan 1 kursi, harga turun ~11.08%.
7	Age	-0.0621	<input checked="" type="checkbox"/> Ya	Mobil lebih tua 1 tahun → harga turun ~6.21%.
8	Kilometers_Driven_log	0.1477	<input checked="" type="checkbox"/> Ya	Log jarak tempuh lebih tinggi → harga naik ~14.77%.
9	Location_Bangalore	0.0105	<input type="checkbox"/> Tidak	Tidak signifikan terhadap harga.
10	Location_Chennai	0.0794	<input checked="" type="checkbox"/> Ya	Mobil di Chennai cenderung lebih mahal ~7.94%.
11	Location_Coimbatore	-0.0579	<input checked="" type="checkbox"/> Ya	Mobil di Coimbatore cenderung lebih murah ~5.79%.
12	Location_Delhi	0.0956	<input checked="" type="checkbox"/> Ya	Mobil di Delhi cenderung lebih mahal ~9.56%.
13	Location_Hyderabad	-0.0779	<input checked="" type="checkbox"/> Ya	Mobil di Hyderabad cenderung lebih murah ~7.79%.
14	Location_Jaipur	-0.0338	<input type="checkbox"/> Tidak	Tidak signifikan.
15	Location_Kochi	-0.2493	<input checked="" type="checkbox"/> Ya	Mobil di Kochi jauh lebih murah ~24.93%.
16	Location_Kolkata	-0.0775	<input checked="" type="checkbox"/> Ya	Mobil di Kolkata lebih murah ~7.75%.
17	Location_Mumbai	-0.0566	<input checked="" type="checkbox"/> Ya	Mobil di Mumbai lebih murah ~5.66%.
18	Location_Pune	0.1908	<input checked="" type="checkbox"/> Ya	Mobil di Pune lebih mahal ~19.08%.
19	Fuel_Type_Diesel	1.2610	<input checked="" type="checkbox"/> Ya	Mobil diesel cenderung lebih mahal ~126.1%.
20	Fuel_Type_Electric	-0.0514	<input type="checkbox"/> Tidak	Tidak signifikan.
21	Fuel_Type_LPG	-0.1349	<input checked="" type="checkbox"/> Ya	Mobil LPG lebih murah ~13.49%.
22	Fuel_Type_Petrol	-0.2286	<input checked="" type="checkbox"/> Ya	Mobil bensin lebih murah ~22.86%.
23	Transmission_Manual	0.2185	<input type="checkbox"/> Tidak	Belum signifikan meskipun koefisien positif.
24	Owner_Type_Fourth & Above	-0.0943	<input checked="" type="checkbox"/> Ya	Pemilik ke-4+ → harga turun ~9.43%.
25	Owner_Type_Second	-0.1444	<input checked="" type="checkbox"/> Ya	Pemilik kedua → harga turun ~14.44%.
26	Owner_Type_Third	-0.2178	<input checked="" type="checkbox"/> Ya	Pemilik ketiga → harga turun ~21.78%.
27	Brand_Class_Low	-0.2178	<input checked="" type="checkbox"/> Ya	Brand kelas bawah → harga lebih rendah ~21.78%.

Hasil regresi menunjukkan bahwa Power, Engine, dan Kilometers_Driven_log berpengaruh positif signifikan terhadap harga mobil. Setiap kenaikan tenaga mesin atau kapasitas mesin meningkatkan harga, begitu juga dengan log jarak tempuh. Sebaliknya, Seats, Mileage, dan Age berpengaruh negatif signifikan; lebih banyak kursi, efisiensi bahan bakar yang tinggi, dan usia mobil yang bertambah justru menurunkan harga. Lokasi juga berperan: mobil di Chennai cenderung lebih mahal, sedangkan di Coimbatore lebih murah. Variabel Kilometers_Driven dan Location_Bangalore tidak signifikan terhadap harga.

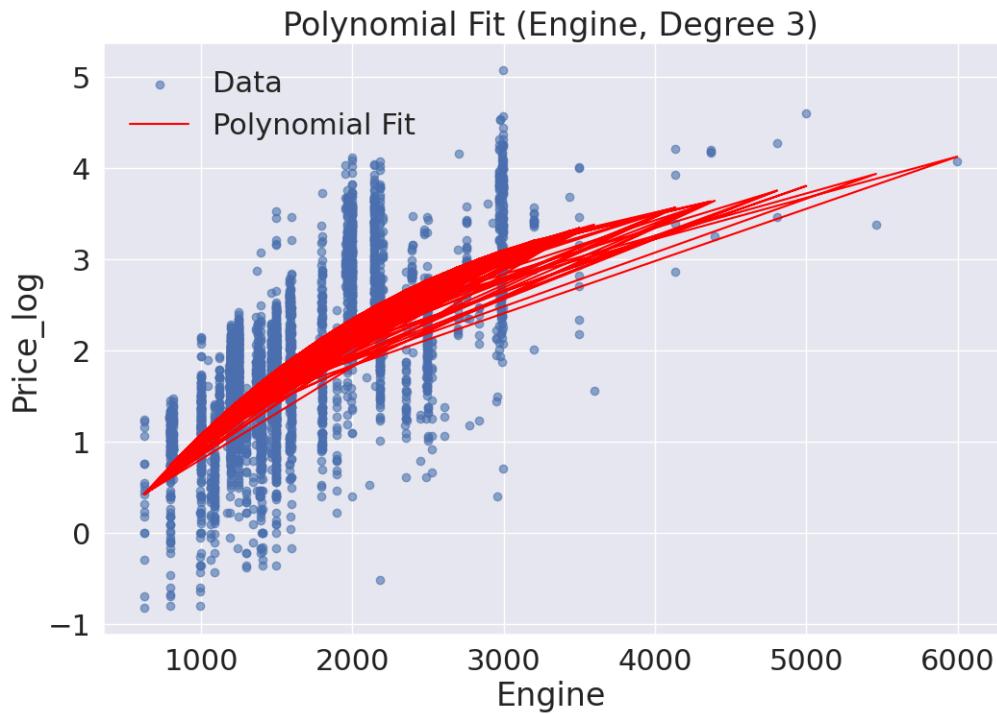
Grafik Regression Line dan Polynomial Fit

1. Regression Line



Gambar ini menunjukkan hubungan antara kapasitas mesin (Engine) dengan harga mobil bekas dalam skala logaritmik (Price_log) menggunakan model regresi linear. Titik-titik abu-abu merepresentasikan data asli, sedangkan garis merah menunjukkan garis regresi yang dihasilkan model. Terlihat pola yang cukup jelas: semakin besar kapasitas mesin, cenderung harga mobil semakin tinggi, yang juga diperkuat oleh arah kemiringan garis regresi ke atas. Meskipun ada penyebaran data yang cukup lebar di beberapa titik, secara umum model mampu menangkap tren positif antara kapasitas mesin dengan harga mobil bekas.

2. Polynomial Fit



Gambar ini menunjukkan hasil pemodelan hubungan antara kapasitas mesin (Engine) dengan harga mobil bekas dalam skala logaritmik (Price_log) menggunakan regresi polinomial orde 3. Titik-titik biru merepresentasikan data asli, sedangkan garis merah menggambarkan kurva hasil prediksi model polinomial. Dibandingkan dengan regresi linear, model polinomial ini lebih fleksibel dalam mengikuti pola data, terutama pada area dengan variasi harga yang lebih kompleks. Garis fit yang lebih melengkung menunjukkan bahwa model ini mencoba menangkap pola non-linear dalam data, sehingga memberikan hasil yang lebih baik pada rentang nilai mesin tertentu. Namun, tampak juga bahwa terdapat sedikit overfitting karena garis terlalu mengikuti data, terutama pada bagian tengah.

Kesimpulan

Setelah dilakukan serangkaian eksperimen dan evaluasi terhadap dua pendekatan regresi, yaitu linear dan polinomial, dapat disimpulkan bahwa regresi linear merupakan solusi terbaik karena terbukti lebih seimbang dan andal, dengan performa stabil baik pada data pelatihan maupun data pengujian, ditandai dengan R-squared sebesar 0.894 serta nilai RMSE, MAE, dan MAPE yang relatif rendah dan seimbang, sehingga mampu menggeneralisasi pola harga mobil bekas secara konsisten; selain itu, hasil pengecekan multikolinearitas (VIF) juga menunjukkan bahwa seluruh fitur dalam model linear aman, sehingga interpretasi terhadap pengaruh tiap variabel dapat dipercaya, sedangkan regresi polinomial, meskipun menghasilkan akurasi sangat tinggi pada data training dengan R-squared sebesar 0.986, justru mengalami overfitting parah dengan nilai error yang ekstrem pada data pengujian, menunjukkan bahwa model terlalu menyesuaikan diri dengan data pelatihan dan kehilangan

kemampuan generalisasi, sehingga berdasarkan hasil evaluasi menyeluruh, model regresi linear direkomendasikan untuk digunakan dalam memprediksi harga mobil bekas karena lebih sederhana, mudah diinterpretasikan, dan mampu memberikan prediksi yang andal serta stabil di luar sampel data pelatihan.