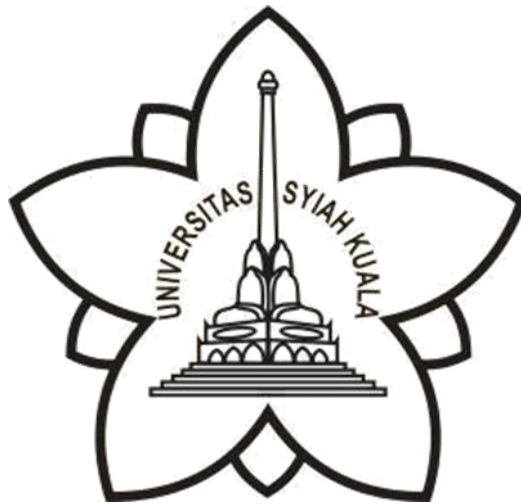


TUGAS 2
LINEAR DAN POLYNOMIAL REGRESSION

disusun untuk memenuhi
tugas mata kuliah Pembelajaran Mesin Kelas A

Oleh:

Aulia Vika Rahman	2208107010001
Shofia Nurul Huda	2208107010015
M. Ihsan Rizqullah Adfa	2208107010029
Nashed Bayazid Ali	2208107010049
Muhammad Raza Adzani	2208107010066
T. Farhan Syah	2208107010090



DEPARTEMEN INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SYIAH KUALA
TAHUN 2025

1. PENDAHULUAN

a. Latar Belakang

Prediksi harga rumah merupakan salah satu aplikasi penting dalam bidang data science yang memiliki nilai praktis tinggi dalam industri real estate. Kemampuan untuk memprediksi harga rumah dengan akurat dapat membantu pembeli, penjual, dan agen properti dalam mengambil keputusan investasi yang tepat. Dalam studi ini, kami menggunakan teknik regresi linear dan polinomial untuk membangun model prediktif harga rumah berdasarkan berbagai fitur properti.

b. Tujuan

Tujuan dari proyek ini adalah:

1. Memahami dan mengimplementasikan model regresi linear dan polinomial pada dataset harga rumah.
2. Menganalisis hubungan antara fitur-fitur properti (variabel independen) dengan harga rumah (variabel dependen).
3. Membandingkan performa model regresi linear dan polinomial dalam memprediksi harga rumah.
4. Menginterpretasikan hasil dan memberikan wawasan tentang faktor-faktor yang mempengaruhi harga rumah.

2. PEMAHAMAN DATASET

a. Sumber Data

Dataset yang digunakan adalah Boston Housing Dataset, yang berasal dari Boston Standard Metropolitan Statistical Area (SMSA) tahun 1970. Dataset ini sering digunakan untuk masalah regresi dalam machine learning dan analisis statistik.

Sumber: [Boston Housing Dataset di Kaggle](#).

b. Deskripsi Variabel

Setiap entri dalam dataset ini merepresentasikan sebuah kota atau daerah di Boston. Dataset ini memiliki berbagai variabel yang mencerminkan faktor ekonomi, lingkungan, dan sosial yang dapat mempengaruhi harga rumah (MEDV). Berikut adalah penjelasan setiap variabel dalam dataset:

1. Variabel Independen (Fitur)

- CRIM: Tingkat kejahatan per kapita di tiap kota.

- ZN: Proporsi lahan perumahan yang memiliki luas lebih dari 25.000 kaki persegi.
- INDUS: Persentase area bisnis non-ritel di suatu kota.
- CHAS: Variabel dummy (1 jika daerah berbatasan dengan Sungai Charles, 0 jika tidak).
- NOX: Konsentrasi nitrogen oksida dalam satuan bagian per 10 juta.
- RM: Rata-rata jumlah kamar dalam setiap rumah.
- AGE: Proporsi unit rumah yang dimiliki dan dibangun sebelum tahun 1940.
- DIS: Jarak tertimbang ke lima pusat pekerjaan utama di Boston.
- RAD: Indeks aksesibilitas ke jalan raya utama (radial highways).
- TAX: Tingkat pajak properti penuh per \$10.000.
- PTRATIO: Rasio jumlah siswa terhadap guru di tiap kota.
- B: Nilai hitungan dari $1000(B_k - 0.63)^2$, di mana B_k adalah persentase populasi kulit hitam di suatu kota.
- LSTAT: Persentase populasi dengan status ekonomi rendah.

2. Variabel Dependen (Target)

- MEDV: Nilai median rumah yang dimiliki dalam satuan ribuan dolar.

c. Statistik Deskriptif

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 506 entries, 0 to 505
Data columns (total 14 columns):
#   Column      Non-Null Count  Dtype
---  -
0    CRIM        506 non-null    float64
1    ZN          506 non-null    float64
2    INDUS       506 non-null    float64
3    CHAS        506 non-null    int64
4    NOX         506 non-null    float64
5    RM          506 non-null    float64
6    AGE         506 non-null    float64
7    DIS         506 non-null    float64
8    RAD         506 non-null    int64
9    TAX         506 non-null    float64
10   PTRATIO     506 non-null    float64
11   B           506 non-null    float64
12   LSTAT       506 non-null    float64
13   MEDV        506 non-null    float64
dtypes: float64(12), int64(2)
memory usage: 55.5 KB
```

gambar 2.c.1. memeriksa informasi dataset

Dari informasi dasar dataset menggunakan `data.info()`, diperoleh bahwa:

1. Dataset terdiri dari 506 sampel
2. Terdapat 14 kolom (13 fitur dan 1 target)
3. 12 kolom bertipe float64 dan 2 kolom bertipe int64
4. Tidak terdapat missing value pada dataset

	count	mean	std	min	25%	50%	75%	max
CRIM	506.0	3.613524	8.601545	0.00632	0.082045	0.25651	3.677083	88.9762
ZN	506.0	11.363636	23.322453	0.00000	0.000000	0.00000	12.500000	100.0000
INDUS	506.0	11.136779	6.860353	0.46000	5.190000	9.69000	18.100000	27.7400
CHAS	506.0	0.069170	0.253994	0.00000	0.000000	0.00000	0.000000	1.0000
NOX	506.0	0.554695	0.115878	0.38500	0.449000	0.53800	0.624000	0.8710
RM	506.0	6.284634	0.702617	3.56100	5.885500	6.20850	6.623500	8.7800
AGE	506.0	68.574901	28.148861	2.90000	45.025000	77.50000	94.075000	100.0000
DIS	506.0	3.795043	2.105710	1.12960	2.100175	3.20745	5.188425	12.1265
RAD	506.0	9.549407	8.707259	1.00000	4.000000	5.00000	24.000000	24.0000
TAX	506.0	408.237154	168.537116	187.00000	279.000000	330.00000	666.000000	711.0000
PTRATIO	506.0	18.455534	2.164946	12.60000	17.400000	19.05000	20.200000	22.0000
B	506.0	356.674032	91.294864	0.32000	375.377500	391.44000	396.225000	396.9000
LSTAT	506.0	12.653063	7.141062	1.73000	6.950000	11.36000	16.955000	37.9700
MEDV	506.0	22.532806	9.197104	5.00000	17.025000	21.20000	25.000000	50.0000

gambar 2.c.2 memeriksa statistik deskriptif dataset

Dari statistik deskriptif dataset yang diperoleh menggunakan `data.describe()`, dapat disimpulkan beberapa hal berikut:

1. Jumlah sampel: 506
2. CRIM (tingkat kejahatan): Rata-rata 3.61, maksimum 88.98 (variabilitas tinggi)
3. ZN (zona perumahan besar): Rata-rata 11.36, mayoritas 0
4. INDUS (area bisnis non-ritel): Rata-rata 11.14, maksimum 27.74
5. CHAS (dekat sungai Charles): Mayoritas 0, hanya sebagian kecil 1
6. NOX (polusi udara): Rata-rata 0.55, rentang 0.38 - 0.87
7. RM (jumlah kamar): Rata-rata 6.28, rentang 3.56 - 8.78
8. AGE (rumah tua): Rata-rata 68.57%, ada yang 100% rumah tua
9. DIS (jarak ke pusat kerja): Rata-rata 3.79, maksimum 12.13
10. RAD (akses jalan raya): Rata-rata 9.55, variasi tinggi
11. TAX (pajak properti): Rata-rata 408.24, rentang 187 - 711
12. PTRATIO (rasio murid-guru): Rata-rata 18.46, maksimum 22.0

- 13. B (proporsi penduduk kulit hitam): Rata-rata 356.67, maksimum 396.9
- 14. LSTAT (persentase ekonomi rendah): Rata-rata 12.65%, maksimum 37.97%
- 15. MEDV (harga rumah median): Rata-rata 22.53, rentang 5.00 - 50.00
- 16. Catatan: Variasi tinggi pada CRIM, ZN, AGE, RAD, berpotensi memengaruhi harga rumah.

3. EKSPLORASI DATA DAN PRA-PEMROSESAN

a. Exploratory Data Analysis (EDA)

1. Pemeriksaan Missing Value

```
# Cek jumlah missing value per kolom
missing_values = data.isnull().sum()

# Cek total missing value dalam dataset
total_missing = missing_values.sum()

print("Jumlah missing value per kolom:")
print(missing_values)
print(f"\nTotal missing value dalam dataset:
{total_missing}")
```

gambar 3.a.1.1 Code untuk memeriksa missing value

```
Jumlah missing value per kolom:
CRIM      0
ZN        0
INDUS     0
CHAS      0
NOX       0
RM        0
AGE       0
DIS       0
RAD       0
TAX       0
PTRATIO   0
B         0
LSTAT     0
MEDV     0
dtype: int64

Total missing value dalam dataset: 0
```

gambar 3.a.1.2 Output pemeriksaan missing value

Hasil pemeriksaan menunjukkan bahwa tidak terdapat missing value pada dataset. Ini merupakan keuntungan karena kita tidak perlu melakukan penanganan data yang hilang.

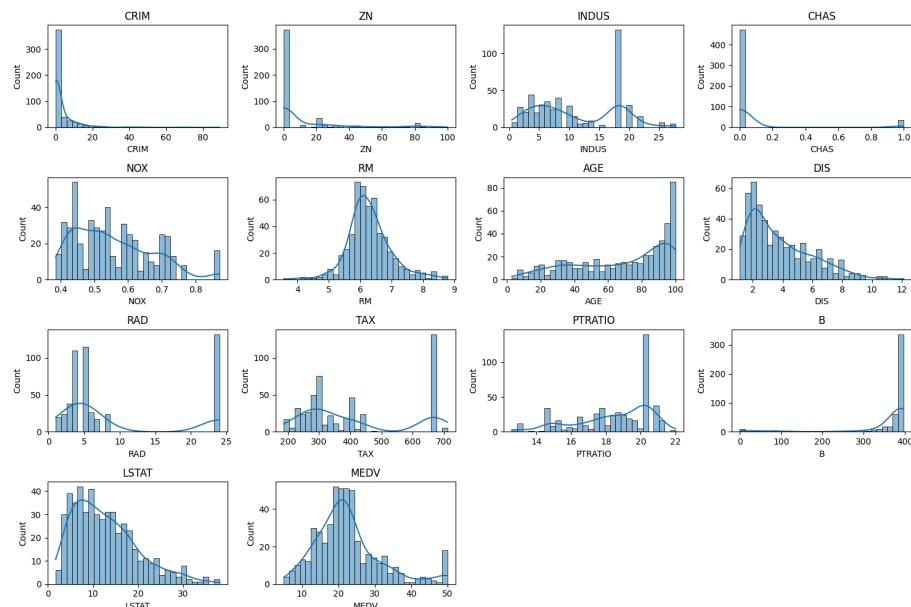
2. Distribusi Nilai Fitur

```
# Set ukuran figure
plt.figure(figsize=(15, 10))

# Loop untuk menampilkan histogram setiap kolom
for i, column in enumerate(data.columns, 1):
    plt.subplot(4, 4, i) # Sesuaikan jumlah baris dan
    kolom subplot
    sns.histplot(data[column], bins=30, kde=True)
    plt.title(column)

plt.tight_layout()
plt.show()
```

gambar 3.a.2.1 Code untuk melihat distribusi nilai fitur



gambar 3.a.2.2 Distribusi nilai fitur

Dari visualisasi histogram untuk setiap fitur, ditemukan bahwa:

- Sebagian besar fitur memiliki distribusi yang skewed (miring)
- Beberapa fitur seperti RM (rata-rata jumlah kamar) dan MEDV (target) sudah memiliki distribusi yang mendekati normal
- Fitur-fitur seperti CRIM, ZN, dan B memiliki distribusi yang sangat tidak merata

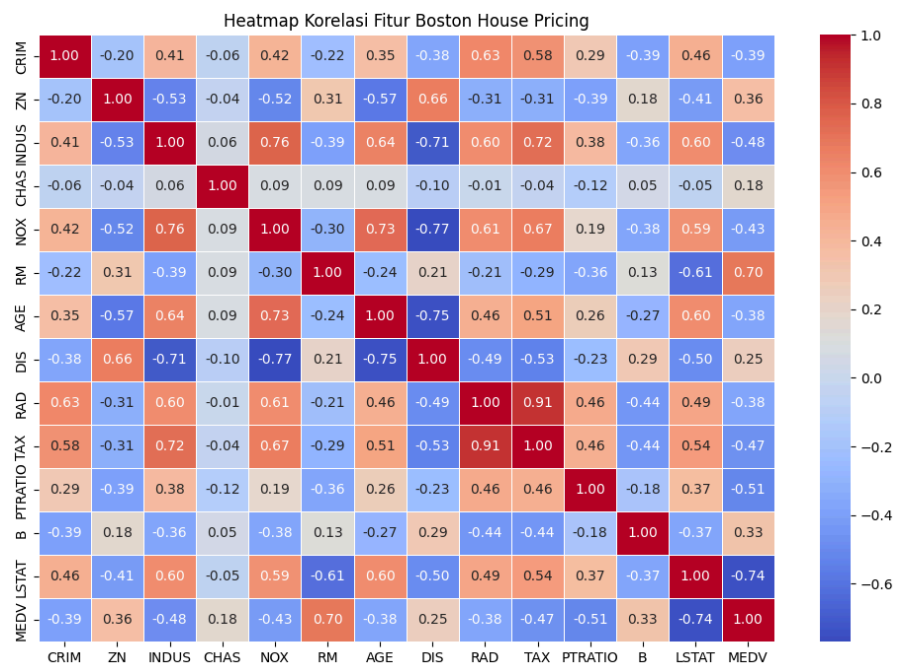
3. Analisis Korelasi

```
# Hitung matriks korelasi
corr_matrix = data.corr()

# Plot heatmap
plt.figure(figsize=(12, 8))
sns.heatmap(corr_matrix, annot=True, cmap="coolwarm",
            fmt=".2f", linewidths=0.5)

plt.title("Heatmap Korelasi Fitur Boston House Pricing")
plt.show()
```

gambar 3.a.3.1 Code untuk menghitung matriks korelasi



gambar 3.a.3.2 Heatmap korelasi antar fitur

Heatmap korelasi menunjukkan beberapa temuan penting:

- Terdapat korelasi tinggi antara RAD dan TAX (fitur aksesibilitas jalan raya dan tingkat pajak properti)
- Fitur RM (jumlah kamar) memiliki korelasi positif yang kuat dengan target MEDV
- Fitur LSTAT (persentase populasi dengan status ekonomi rendah) memiliki korelasi negatif yang kuat dengan target MEDV



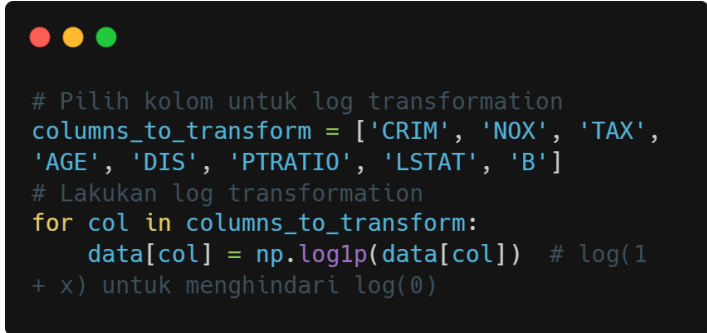
```
# Hapus fitur RAD
data.drop(['RAD'], axis=1, inplace=True)
```

gambar 3.a.3.3 menghapus fitur RAD

Dikarenakan RAD dan TAX memiliki korelasi yang tinggi, kami memutuskan untuk menghapus fitur RAD untuk mengatasi masalah multikolinearitas yang dapat mempengaruhi model regresi.

b. Log Transformation dan standarisasi

1. Log Transformation

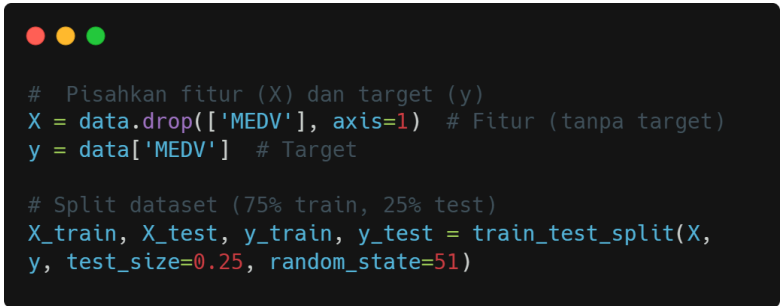


```
# Pilih kolom untuk log transformation
columns_to_transform = ['CRIM', 'NOX', 'TAX',
                        'AGE', 'DIS', 'PTRATIO', 'LSTAT', 'B']
# Lakukan log transformation
for col in columns_to_transform:
    data[col] = np.log1p(data[col]) # log(1
    + x) untuk menghindari log(0)
```

gambar 3.b.1 Code untuk melakukan log transformation

Untuk mengatasi distribusi yang skewed pada beberapa fitur, kami melakukan log transformation pada fitur-fitur berikut: CRIM, NOX, TAX, AGE, DIS, PTRATIO, LSTAT, dan B. Log transformation dilakukan dengan formula $\log(1+x)$ untuk menghindari error ketika nilai $x=0$.

2. Pembagian Data



```
# Pisahkan fitur (X) dan target (y)
X = data.drop(['MEDV'], axis=1) # Fitur (tanpa target)
y = data['MEDV'] # Target

# Split dataset (75% train, 25% test)
X_train, X_test, y_train, y_test = train_test_split(X,
                                                    y, test_size=0.25, random_state=51)
```

gambar 3.b.2 Code untuk membagi dataset

Dataset dibagi menjadi:

- 75% data training (untuk melatih model)
- 25% data testing (untuk evaluasi model)

3. Standarisasi

```
# Standarisasi hanya dengan data training (untuk  
menghindari data leakage)**  
scaler = StandardScaler()  
X_train_scaled = scaler.fit_transform(X_train)  
X_test_scaled = scaler.transform(X_test)
```

gambar 3.b.3 Code untuk melakukan standarisasi fitur-fitur

Standarisasi dilakukan pada fitur-fitur untuk menyamakan skala data dengan mean=0 dan standar deviasi=1. Proses standarisasi hanya dilakukan berdasarkan data training untuk menghindari data leakage.

4. IMPLEMENTASI MODEL

a. Regresi Linear

```
# **Train model Linear Regression**  
model = LinearRegression()  
model.fit(X_train_scaled, y_train)  
  
# **Interpretasi Koefisien Regresi**  
coef_df = pd.DataFrame(model.coef_, X.columns,  
columns=['Koefisien'])  
print("\n Koefisien Regresi:")  
print(coef_df.sort_values(by='Koefisien',  
ascending=False))
```

gambar 4.a.1 Kode untuk train model dan interpretasi koefisien regresi

Implementasi model regresi linear dilakukan dengan menggunakan library scikit-learn. Model dilatih menggunakan data training yang telah distandardisasi. Setelah pelatihan, koefisien model regresi linear dianalisis untuk memahami pengaruh masing-masing fitur terhadap harga rumah (MEDV).

Koefisien regresi dari model linear (diurutkan dari yang terbesar ke terkecil):

Koefisien Regresi:	
	Koefisien
RM	2.201587
B	0.887338
CHAS	0.789847
AGE	0.697692
CRIM	0.426548
ZN	0.009895
TAX	-0.137377
INDUS	-0.164907
PTRATIO	-1.751206
NOX	-2.259726
DIS	-2.756087
LSTAT	-5.371723

gambar 4.a.2 koefisien regresi

b. Regresi Polinomial

```
# Buat fitur polinomial (derajat 2)
poly = PolynomialFeatures(degree=2,
include_bias=False)
X_train_poly = poly.fit_transform(X_train)
X_test_poly = poly.transform(X_test)

# Latih model Linear Regression dengan fitur
polinomial
poly_model = LinearRegression()
poly_model.fit(X_train_poly, y_train)
```

gambar 4.b.1 penerapan model

Untuk menangkap pola non-linear dalam data, kami mengimplementasikan model regresi polinomial dengan derajat 2. Implementasi dilakukan dengan menggunakan `PolynomialFeatures` dari `scikit-learn` yang mentransformasi fitur-fitur menjadi bentuk polinomial. Selanjutnya, model regresi linear diterapkan pada fitur-fitur polinomial tersebut.

5. EVALUASI MODEL

a. Metrik Evaluasi

Evaluasi kinerja model dilakukan dengan menggunakan tiga metrik utama:

- Mean Squared Error (MSE): Rata-rata kuadrat selisih antara nilai prediksi dengan nilai sebenarnya
- Mean Absolute Error (MAE): Rata-rata nilai absolut dari selisih antara nilai prediksi dengan nilai sebenarnya
- R^2 Score: Proporsi variasi dalam variabel dependen yang dapat dijelaskan oleh model

b. Hasil Evaluasi Model Regresi Linear

```
# **Evaluasi model**
mse = mean_squared_error(y_test, y_pred)
mae = mean_absolute_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# **Tampilkan hasil evaluasi**
print(f"\n Mean Squared Error (MSE): {mse:.4f}")
print(f" Mean Absolute Error (MAE): {mae:.4f}")
print(f"  $R^2$  Score: {r2:.4f}")
```

gambar 5.b Evaluasi model regresi linear

Hasil evaluasi model regresi linear pada data testing:

- MSE: 20.7593
- MAE: 3.1947
- R^2 Score: 0.7888

c. Hasil Evaluasi Model Regresi Polinomial

```
# Prediksi dan hitung R2 Score
y_pred_poly = poly_model.predict(X_test_poly)
r2_poly = r2_score(y_test, y_pred_poly)
mse_poly = mean_squared_error(y_test, y_pred_poly)
mae_poly = mean_absolute_error(y_test, y_pred_poly)

print(f"R2 Score - Polynomial Regression (degree=2):
{r2_poly:.4f}")
print(f"MSE Score - Polynomial Regression
(degree=2): {mse_poly:.4f}")
print(f"MAE Score - Polynomial Regression
(degree=2): {mae_poly:.4f}")
```

gambar 5.c Evaluasi model regresi polinomial

Hasil evaluasi model regresi polinomial (derajat 2) pada data testing:

- MSE: 10.7539
- MAE: 2.4063
- R^2 Score: 0.8906

d. Perbandingan Model

Model	MSE	MAE	R^2 Score
Linear Regression	20.7593	3.1947	0.7888
Polynomial Regression (D=2)	10.7539	2.4063	0.8906

Berdasarkan hasil perbandingan, model regresi polinomial (derajat 2) memberikan performa yang lebih baik dibandingkan model regresi linear pada ketiga metrik evaluasi.

6. ANALISIS HASIL

a. Interpretasi Koefisien Regresi Linear

Dari koefisien regresi linear, dapat diinterpretasikan bahwa:

1. Fitur dengan pengaruh positif terbesar

RM (Jumlah rata-rata kamar) → Koefisien: 2.2016, Semakin banyak jumlah kamar dalam rumah, harga rumah cenderung meningkat. Ini masuk akal karena rumah dengan lebih banyak kamar biasanya lebih besar dan lebih mewah, sehingga memiliki nilai jual yang lebih tinggi.

2. Fitur dengan pengaruh negatif terbesar

LSTAT (Persentase penduduk berpenghasilan rendah) → Koefisien: -5.3717, Semakin tinggi persentase penduduk berpenghasilan rendah di suatu area, harga rumah cenderung turun. Hal ini menunjukkan bahwa faktor sosial-ekonomi berperan besar dalam menentukan nilai properti.

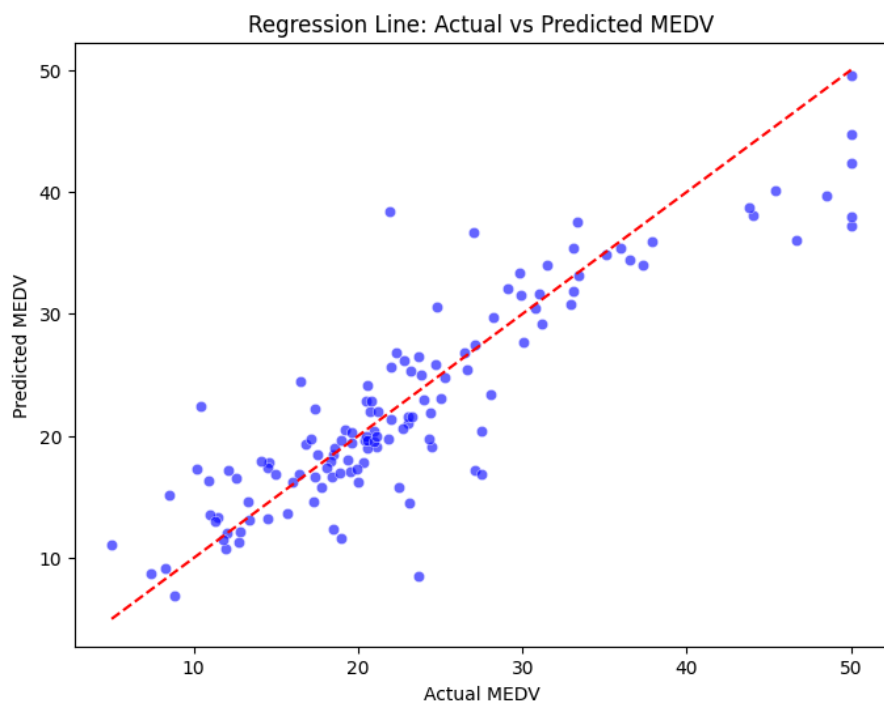
b. Analisis Grafik

1. Regression Line (Linear Regression)

```
plt.figure(figsize=(8, 6))
sns.scatterplot(x=y_test, y=y_pred, color='blue',
alpha=0.6)
sns.lineplot(x=y_test, y=y_test, color='red',
linestyle="--") # Ideal fit line

plt.xlabel("Actual MEDV")
plt.ylabel("Predicted MEDV")
plt.title("Regression Line: Actual vs Predicted MEDV")
plt.show()
```

gambar 6.b.1.1 Menampilkan regression Line



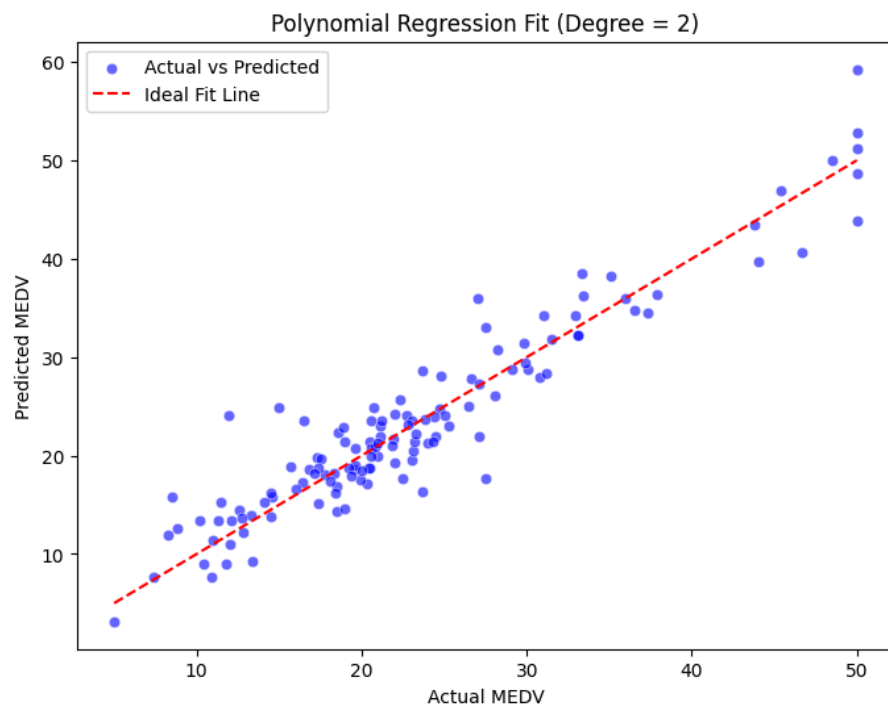
gambar 6.b.1.2 Regression Line

Grafik scatter plot antara nilai MEDV aktual dan nilai prediksi menunjukkan bahwa model regresi linear sudah cukup baik dalam memprediksi harga rumah, meskipun terdapat beberapa titik yang prediksinya cukup jauh dari nilai aktual, terutama untuk rumah dengan harga tinggi.

2. Polynomial Fit (Polynomial Regression)

```
# ** Plot Polynomial Fit**  
plt.figure(figsize=(8, 6))  
sns.scatterplot(x=y_test, y=y_pred_poly,  
color='blue', alpha=0.6, label="Actual vs  
Predicted")  
sns.lineplot(x=y_test, y=y_test, color='red',  
linestyle="--", label="Ideal Fit Line") # Ideal Fit  
  
plt.xlabel("Actual MEDV")  
plt.ylabel("Predicted MEDV")  
plt.title("Polynomial Regression Fit (Degree = 2)")  
plt.legend()  
plt.show()
```

gambar 6.b.2.1 Plot polynomial fit



gambar 6.b.2.2 Polynomial Regression Fit

Grafik scatter plot untuk model polinomial menunjukkan pola yang lebih mengikuti data aktual dibandingkan model linear. Hal ini

membuktikan bahwa model polinomial lebih mampu menangkap pola non-linear dalam data harga rumah.

7. KESIMPULAN DAN SARAN

a. Kesimpulan

Berdasarkan analisis yang telah dilakukan, dapat disimpulkan bahwa:

1. Performa Model:

- Model regresi polinomial (derajat 2) memiliki performa yang lebih baik dibandingkan model regresi linear dalam memprediksi harga rumah pada dataset Boston Housing.
- Peningkatan R^2 Score dari 0.7888 (linear) menjadi 0.8906 (polinomial) menunjukkan bahwa terdapat hubungan non-linear yang signifikan antara fitur-fitur dan harga rumah.

2. Faktor Penentu Harga Rumah:

- Jumlah kamar (RM) merupakan faktor positif terkuat yang mempengaruhi harga rumah.
- Persentase penduduk berpenghasilan rendah (LSTAT) merupakan faktor negatif terkuat yang mempengaruhi harga rumah.
- Faktor-faktor lain seperti tingkat polusi (NOX) dan jarak ke pusat pekerjaan (DIS) juga memiliki pengaruh negatif yang signifikan terhadap harga rumah.

b. Saran

Untuk pengembangan model prediksi harga rumah yang lebih baik, beberapa saran yang dapat dipertimbangkan:

1. Eksplorasi Model Lanjutan:

- Mencoba model regresi polinomial dengan derajat yang berbeda untuk mencari titik optimal antara bias dan varians.
- Menerapkan teknik regularisasi seperti Ridge atau Lasso untuk mengatasi masalah overfitting yang mungkin terjadi pada model polinomial.

2. Fitur Engineering:

- Menciptakan fitur-fitur baru yang mungkin memiliki relevansi dengan harga rumah, seperti rasio kamar per luas bangunan atau indeks kualitas lingkungan.

- Melakukan seleksi fitur yang lebih mendalam untuk mengurangi dimensi dan meningkatkan interpretabilitas model.

3. Validasi Model:

- Menerapkan teknik cross-validation untuk mendapatkan estimasi performa model yang lebih robust.
- Menggunakan teknik sampling seperti bootstrap untuk mengukur interval kepercayaan dari estimasi parameter model.