



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-3: Regresi Linear**  
Mata Kuliah Pembelajaran Mesin

**Ketentuan;**

Simpanlah file lembar jawaban ini dengan format; **Kelas\_Nama Lengkap**

Contoh; **TI 3C\_Rifki Fakhruhin**

**Upload tugas jobsheet ini dengan batas maksimum sesuai jadwal perkuliahan masing masing kelas**

Upload file tugas jobsheet di website **lms.polinema**

<b>Nama</b>	:	Rifki Fakhruhin
<b>Nim</b>	:	2241720218
<b>Kelas</b>	:	3C

Tulislah Jawaban Pada Kolom Yang tersedia di bawah ini;

**LEMBAR JAWABAN JOBSHEET-3**

<b>Soal No</b>	<b>Jawaban</b>
1	<b>Langkah 1: Import Library</b> <pre># import package import numpy as np import pandas as pd</pre>
2	<b>Langkah 2: Baca Data</b> <pre># baca data dari file CSV from google.colab import files uploaded = files.upload() # upload dataset  df = 'dataset.csv' data = pd.read_csv(df)</pre>
3	<b>Langkah 3: Pemahaman Terhadap Data</b> <pre># melihat beberapa data awal data.head()  # mengecek ukuran data data.shape  # informasi tentang data data.info()  # deskripsi data data.describe()</pre>



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-3: Regresi Linear**  
Mata Kuliah Pembelajaran Mesin

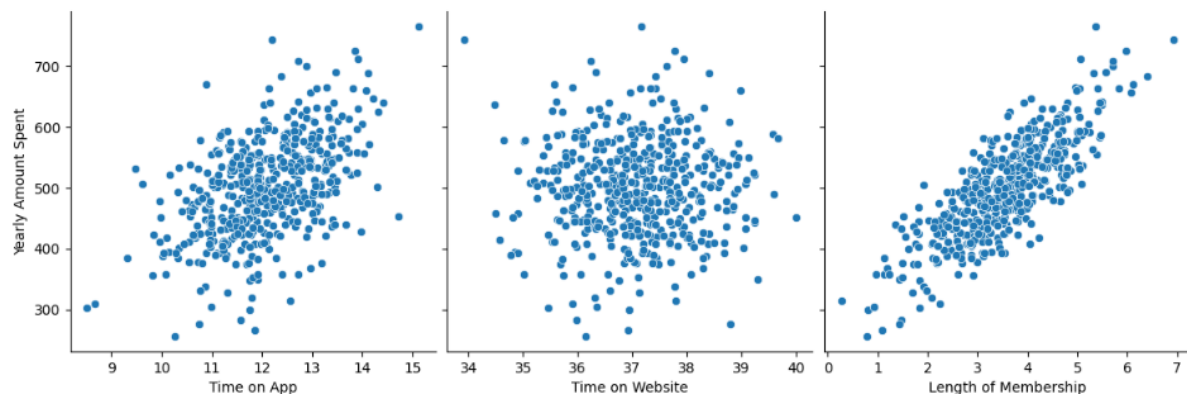
	Email	Address	Avatar	Avg. Session Length	Time on App	Time on Website	Length of Membership	Yearly Amount Spent
0	mstephenson@fernandez.com	835 Frank Tunnelv\nWrightmouth, MI 82180-9605	Violet	34.497268	12.655651	39.577668	4.082621	587.951054
1	hduke@hotmail.com	4547 Archer Commonv\nDiazchester, CA 06566-8576	DarkGreen	31.926272	11.109461	37.268959	2.664034	392.204933
2	pallen@yahoo.com	24645 Valerie Unions Suite 582v\nCobbborough,...	Bisque	33.000915	11.330278	37.110597	4.104543	487.547505
3	riverarebecca@gmail.com	1414 David Throughwayv\nPort Jason, OH 22070-...	SaddleBrown	34.305557	13.717514	36.721283	3.120179	581.852344
4	mstephens@davidson-herman.com	14023 Rodriguez Passagev\nPort Jacobville, PR...	MediumAquaMarine	33.330673	12.795189	37.536853	4.446308	599.406092

#### 4 Langkah 4: Visualisasi Data

```
# import library untuk visualisasi
import matplotlib.pyplot as plt
import seaborn as sns
```

Import library Matplotlib dan Seaborn untuk visualisasi data.

```
# visualisasi data dengan pairplot
sns.pairplot(data, x_vars=['Time on App', 'Time on Website', 'Length of Membership'],
             y_vars='Yearly Amount Spent', size=4, aspect=1, kind='scatter')
plt.show()
```



Gunakan pairplot untuk menampilkan hubungan antara variabel bebas dan variabel target dalam bentuk scatter plot.

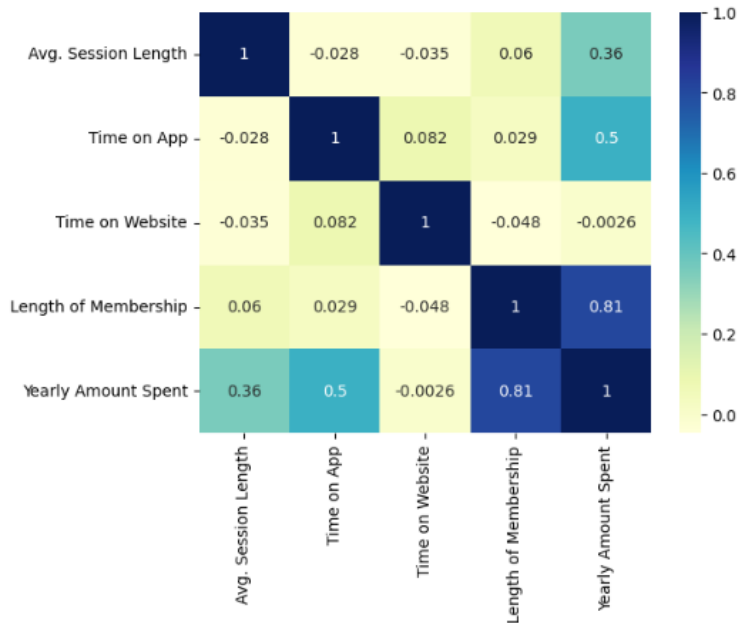
```
# Pilih hanya kolom numerik
numerical_data = data.select_dtypes(include=[np.number])

# Hitung korelasi hanya pada kolom numerik
correlation = numerical_data.corr()

# Visualisasikan korelasi dengan heatmap
sns.heatmap(correlation, cmap="YlGnBu", annot=True)
plt.show()
```



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-3: Regresi Linear**  
Mata Kuliah Pembelajaran Mesin



5 **Langkah 6: Regresi Linier**

```
#Regresi linear
# Membuat variabel bebas X dan Y, contoh pengambilan dari analisis korelasi sebelumnya
X = data['Length of Membership']
y = data['Yearly Amount Spent']
```

Pisahkan variabel bebas (X) dan variabel target (y).

```
# Pembagian data latih dan data uji dengan proporsi 7:3
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, train_size=0.7, test_size=0.3, random_state=100)
```

Bagi data menjadi data latih (70%) dan data uji (30%) menggunakan `train_test_split`.

Lakukan training model regresi linier menggunakan library `StatsModels`. Tambahkan konstanta (intercept) ke variabel bebas.

```
# Training model
import statsmodels.api as sm

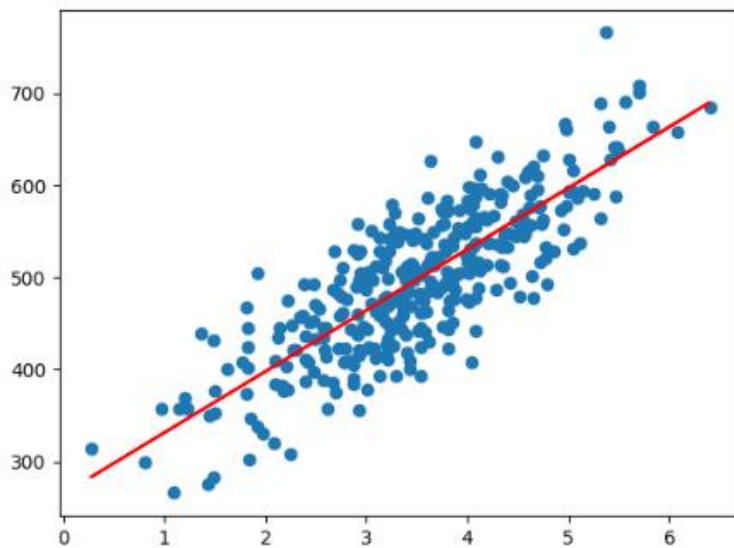
X_train_sm = sm.add_constant(X_train)
lr = sm.OLS(y_train, X_train_sm).fit()
```



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-3: Regresi Linear**  
Mata Kuliah Pembelajaran Mesin

Visualisasikan garis regresi pada data latih.

```
# Visualisasi garis regresi
plt.scatter(X_train, y_train)
plt.plot(X_train, 265.2483 + 66.3015*X_train, 'r')
plt.show()
```



6 **Langkah 7: Analisis Residual**

```
# Prediksi nilai y_value dari data x yang telah dilatih
y_train_pred = lr.predict(X_train_sm)

# Menghitung residual
res = (y_train - y_train_pred)
```

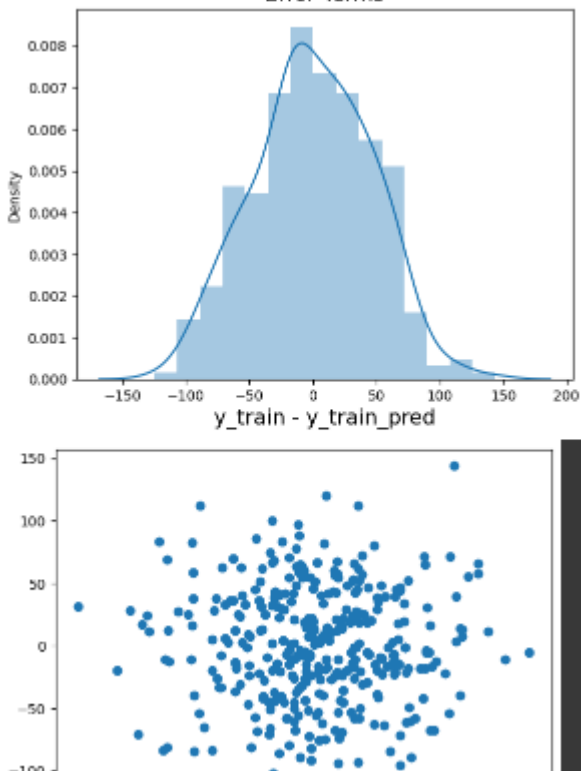
Lakukan prediksi nilai y dari data latih dan hitung residual (selisih antara nilai sebenarnya dan nilai prediksi).

```
# Histogram residual
fig = plt.figure()
sns.distplot(res, bins=15)
plt.title('Error Terms', fontsize=15)
plt.xlabel('y_train - y_train_pred', fontsize=15)
plt.show()

# Scatter plot residual
plt.scatter(X_train, res)
plt.show()
```

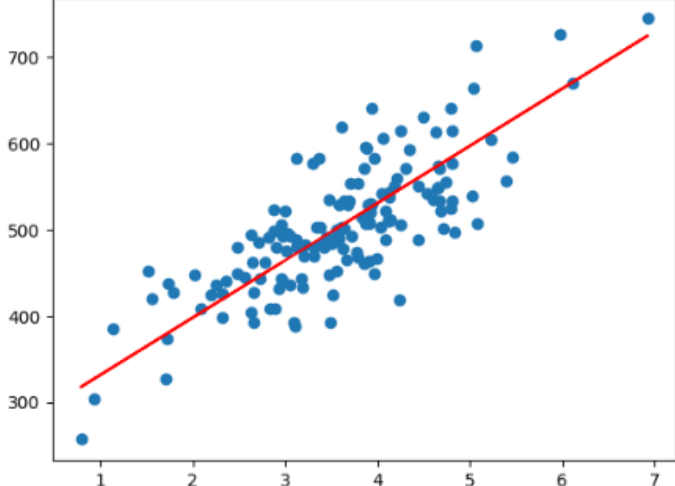


Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-3: Regresi Linear**  
 Mata Kuliah Pembelajaran Mesin

	<p style="text-align: center;">Error Terms</p> 
7	<p><b>Langkah 8: Prediksi pada Data Uji dan Evaluasi Model</b></p> <pre># Prediksi pada data uji X_test_sm = sm.add_constant(X_test) y_test_pred = lr.predict(X_test_sm)</pre> <p>Lakukan prediksi pada data uji.</p> <pre># Hitung nilai R-squared from sklearn.metrics import r2_score r_squared = r2_score(y_test, y_test_pred)</pre> <p>Hitung nilai R-squared untuk mengukur kinerja model pada data uji.</p> <pre># Hitung nilai R-squared from sklearn.metrics import r2_score r_squared = r2_score(y_test, y_test_pred)</pre>
8	<p><b>Langkah 9: Visualisasi Hasil</b></p>

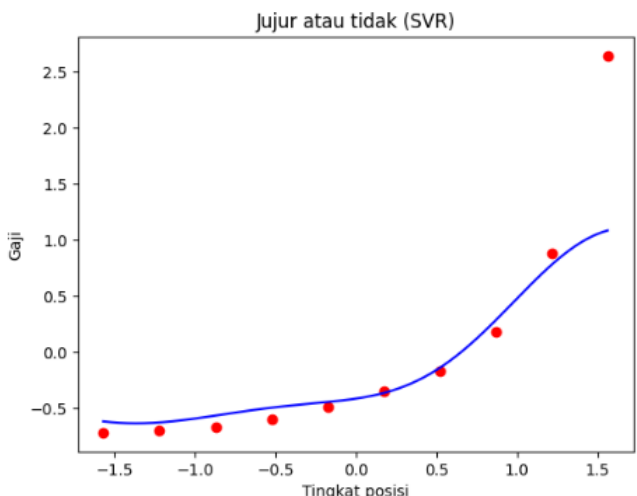


Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-3: Regresi Linear**  
Mata Kuliah Pembelajaran Mesin

	<pre># Visualisasi data uji dan hasil prediksi plt.scatter(X_test, y_test) plt.plot(X_test, y_test_pred, 'r') plt.show()</pre> 
9	<b>Praktikum 2</b> <b>Langkah 1 : Mengimpor Library:</b> <pre>#import library import numpy as np import matplotlib.pyplot as plt import pandas as pd</pre>
10	<b>Langkah 2 : Mengimpor Dataset:</b> <pre># baca data dari file CSV from google.colab import files uploaded = files.upload() # upload dataset  df = 'Posisi_gaji.csv' data = pd.read_csv(df)  dataset = pd.read_csv('Posisi_gaji.csv') X = dataset.iloc[:, 1:2].values y = dataset.iloc[:, 2].values # Ubah menjadi satu kolom saja</pre>
11	<b>Langkah 3 : Feature Scaling:</b>



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-3: Regresi Linear**  
Mata Kuliah Pembelajaran Mesin

	<pre># Feature Scaling from sklearn.preprocessing import StandardScaler sc_X = StandardScaler() sc_y = StandardScaler() X = sc_X.fit_transform(X.reshape(-1, 1)) y = sc_y.fit_transform(y.reshape(-1, 1))</pre>
12	<p><b>Langkah 4 : Fitting SVR ke Dataset</b></p> <pre># Fitting SVR ke dataset from sklearn.svm import SVR regressor = SVR(kernel='rbf') regressor.fit(X, y)</pre> <pre>/usr/local/lib/python3.10/dist-packages/sklearn/svm/_classes.py:145: UserWarning: y = column_or_1d(y, warn=True)   y = column_or_1d(y, warn=True)</pre> <p>SVR</p> <p>SVR()</p>
13	<p><b>Langkah 5 : Visualisasi Hasil SVR:</b></p> <pre># Visualisasi hasil SVR (resolusi tinggi dan kurva yang lebih halus) X_grid = np.arange(min(X), max(X), 0.01).reshape(-1, 1) plt.scatter(X, y, color='red') plt.plot(X_grid, regressor.predict(X_grid), color='blue') plt.title('Jujur atau tidak (SVR)') plt.xlabel('Tingkat posisi') plt.ylabel('Gaji') plt.show()</pre> 
14	<p><b>Langkah 6 : Prediksi Hasil:</b></p>



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-3: Regresi Linear**  
Mata Kuliah Pembelajaran Mesin

	<pre># Prediksi hasil # Buat array 2D yang berisi tingkat posisi yang akan diprediksi tingkat_posisi_prediksi = np.array([[6.5]]) # Penskalaan fitur untuk data yang akan diprediksi tingkat_posisi_prediksi = sc_X.transform(tingkat_posisi_prediksi) # Melakukan prediksi menggunakan model SVR gaji_prediksi = regressor.predict(tingkat_posisi_prediksi) # Kembalikan hasil prediksi ke skala aslinya gaji_prediksi = sc_y.inverse_transform(gaji_prediksi.reshape(-1, 1))</pre>
15	<p><b>Langkah 7 : Menampilkan Hasil:</b></p> <pre># Menampilkan hasil prediksi print("Prediksi Gaji untuk Tingkat Posisi 6.5:", gaji_prediksi[0])</pre> <p>Prediksi Gaji untuk Tingkat Posisi 6.5: [170370.0204065]</p>
16	<p><b>Langkah 8 : Evaluasi Model SVR</b></p> <pre># Evaluasi model from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score  y_actual = y y_pred = regressor.predict(X)  # Menghitung MAE mae = mean_absolute_error(y_actual, y_pred)  # Menghitung MSE mse = mean_squared_error(y_actual, y_pred)  # Menghitung RMSE rmse = np.sqrt(mse)  # Menghitung R-squared r2 = r2_score(y_actual, y_pred)  print("MAE:", mae) print("MSE:", mse) print("RMSE:", rmse) print("R-squared:", r2)</pre> <p>MAE: 0.22299274095734414 MSE: 0.24839989293792014 RMSE: 0.4983973243687411 R-squared: 0.7516001070620798</p>