Jason Christopher Wijaya

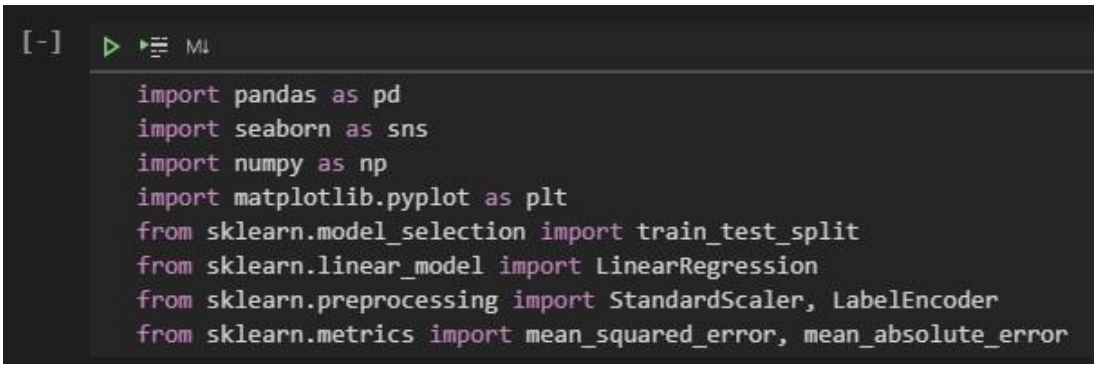
LAPORAN PROJECT MACHINE LEARNING

-Unemployment Rate Prediction-

Methods :
Metode yang akan digunakan adalah dengan memilih algoritma Linear Regression, SVR(Support Vector Regression), dan Ridge untuk melatih model machine learning. Dengan adanya tiga model ini, maka akan dibandingkan error dari masing-masing model untuk menentukan model yang terbaik diantara tiga model tersebut. Error yang akan ditampilkan adalah dalam 3 bentuk yaitu MSE(Mean Square Error), MAE(Mean Absolute Error), dan RMSE(Root Mean Square Error).

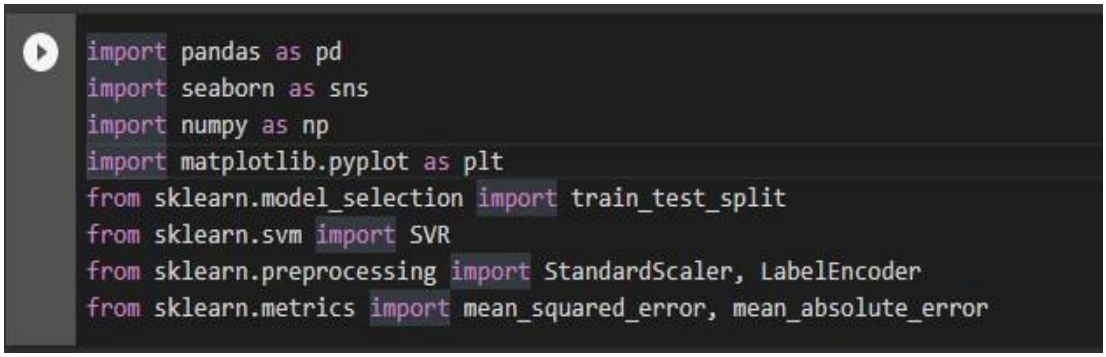
Experiment :
Pertama akan dilakukan import pada library yang dibutuhkan dalam eksperimen, dari ketiga code tersebut hanya ada satu bagian yang berbeda yaitu dimana ketika melakukan import untuk Linear Regression, SVR(Support Vector Regression), dan Ridge.

Linear Regression.



```
[-]
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.metrics import mean_squared_error, mean_absolute_error
```

SVR(Support Vector Regression)



```
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.svm import SVR
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.metrics import mean_squared_error, mean_absolute_error
```

Ridge

```
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import Ridge
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.metrics import mean_squared_error, mean_absolute_error
```

Kedua akan dilakukan read pada dataset “gdp_improvement_rate.csv” sebagai df_1 dan “unemployment.csv” sebagai df_2. Kemudian akan dilakukan merge pada kedua dataset tersebut berdasarkan fitur country yang dinamakan menjadi df. Lalu terakhir akan dicoba untuk menampilkan/display df untuk mengecek apakah dataset sudah berhasil dimerge.

```
# read datasets
df_1 = pd.read_csv("gdp_improvement_rate.csv")
df_2 = pd.read_csv("unemployment.csv")

# merge datasets menjadi 1 dataset baru, merge berdasarkan fitur country
df = pd.merge(df_1, df_2, on='country')

# display dataset baru
df
```

	country	growth_rate	unemployment_percentage
0	Afghanistan	2.500	35.0
1	Albania	3.702	14.0
2	Algeria	1.457	11.7
3	Antigua and Barbuda	2.685	11.0
4	Argentina	2.464	8.1
...	...	...	...
154	Venezuela	-12.000	26.4
155	Vietnam	6.300	2.3
156	Yemen	-2.014	27.0
157	Zambia	3.978	15.0

Ketiga akan dilakukan pengecekan jumlah NULL data, karena sudah tidak ada data yang NULL maka bisa dilanjutkan pada step berikutnya.

```
[-]
# Check jumlah data null
df.isnull().sum()

country                  0
growth_rate              0
unemployment_percentage  0
dtype: int64
```

Keempat akan dilakukan selecting features untuk x_data dan y_data, tetapi sebelum itu akan diubah fitur country dengan label encoder karena fitur tersebut merupakan string. Setelah selesai maka akan langsung dilakukan select features yaitu country dan growth_rate untuk x_data, unemployment_percentage untuk y_data.

```
[-]
# Selecting features
df['country'] = LabelEncoder().fit_transform(df['country'])
x_data = df[['country', 'growth_rate']].values
y_data = df['unemployment_percentage'].values
```

Kelima akan dilakukan split dataset dengan ratio 0.8 untuk training dan 0.2 untuk test.

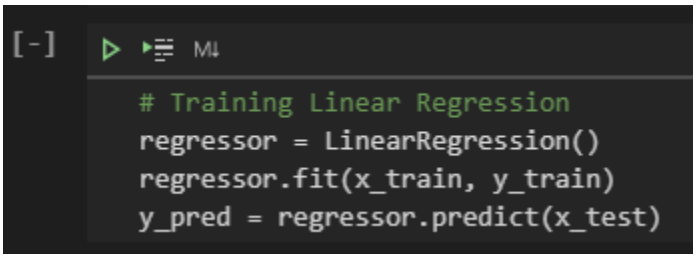
```
[-]
# Split dataset
x_train, x_test, y_train, y_test = train_test_split(x_data, y_data, test_size=0.2, random_state=0)
```

Keenam akan dilakukan normalisasi pada data dengan menggunakan standard scaler.

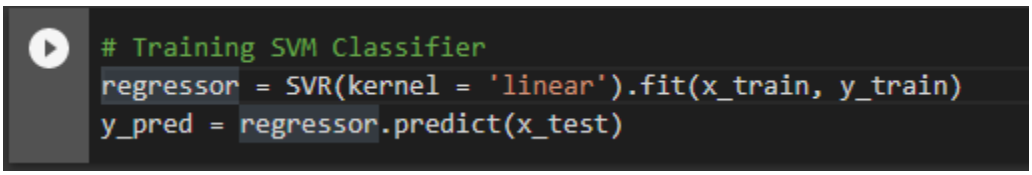
```
[-]
# Feature scaling
sc = StandardScaler().fit(x_train)
x_train = sc.fit_transform(x_train)
x_test = sc.fit_transform(x_test)
```

Ketujuh untuk bagian training akan dipisahkan menjadi tiga karena akan dilakukan training pada tiga model berupa Linear Regression, SVR(Support Vector Regression), dan Ridge.

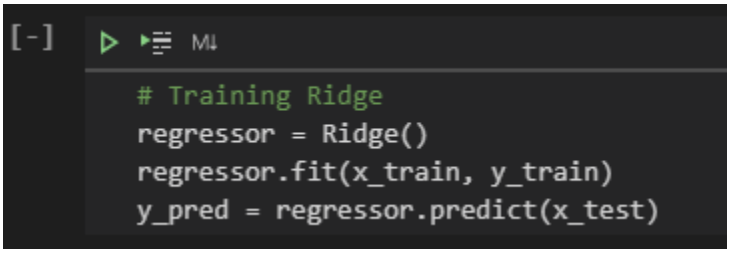
Linear Regression



SVR(Support Vector Regression)



Ridge



Kedelapan untuk bagian metrics evaluation akan ditampilkan MSE(Mean Square Error), MAE(Mean Absolute Error), dan RMSE(Root Mean Square Error) dari ketiga model tersebut.

Linear Regression

```
# Metrics Evaluation
MSE = mean_squared_error(y_test, y_pred)
MAE = mean_absolute_error(y_test, y_pred)
RMSE = mean_squared_error(y_test, y_pred, squared=False)

print(f'MSE : {MSE:.2f}')
print(f'MAE : {MAE:.2f}')
print(f'RMSE : {RMSE:.2f}')
```

```
MSE : 69.88
MAE : 6.45
RMSE : 8.36
```

SVR(Support Vector Regression)

```
# Metrics Evaluation
MSE = mean_squared_error(y_test, y_pred)
MAE = mean_absolute_error(y_test, y_pred)
RMSE = mean_squared_error(y_test, y_pred, squared=False)

print(f'MSE : {MSE:.2f}')
print(f'MAE : {MAE:.2f}')
print(f'RMSE : {RMSE:.2f}')
```

```
MSE : 77.59
MAE : 5.80
RMSE : 8.81
```

Jason Christopher Wijaya

Ridge

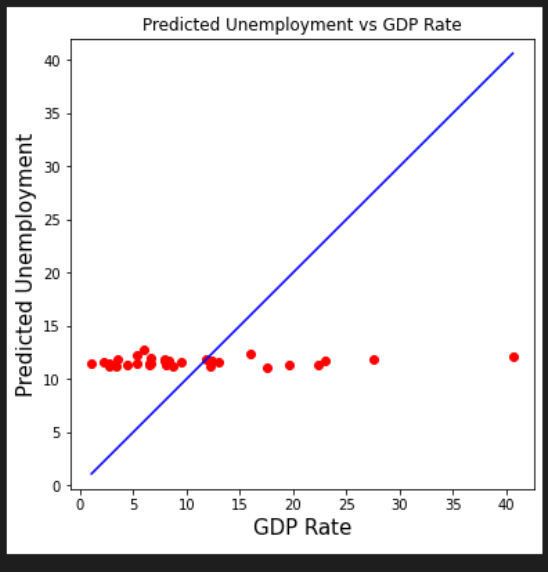
```
[-]
# Metrics Evaluation
MSE = mean_squared_error(y_test, y_pred)
MAE = mean_absolute_error(y_test, y_pred)
RMSE = mean_squared_error(y_test, y_pred, squared=False)

print(f'MSE : {MSE:.2f}')
print(f'MAE : {MAE:.2f}')
print(f'RMSE : {RMSE:.2f}')
```

```
MSE : 69.89
MAE : 6.45
RMSE : 8.36
```

Kesembilan yang terakhir akan ditampilkan visualisasi berupa Predicted Unemployment vs GDP Rate

Linear Regression

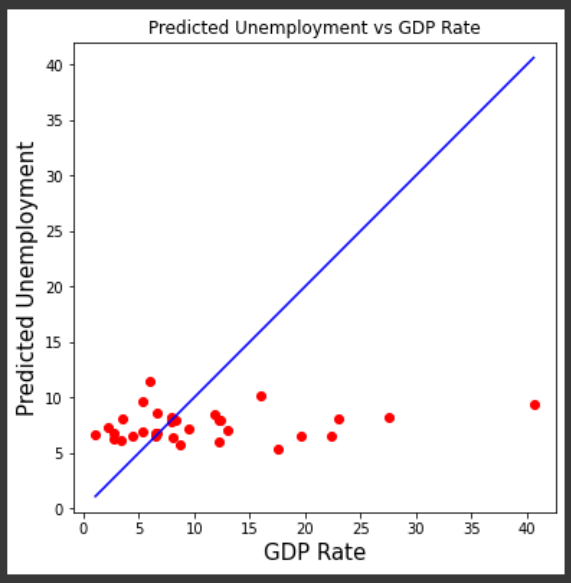
```
[-]
# Visualizing Plot Predicted Value vs Measure
plt.figure(figsize=(6,6))
plt.scatter(y_test, y_pred, color='red')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.title('Predicted Unemployment vs GDP Rate')
plt.xlabel('GDP Rate', fontsize=15)
plt.ylabel('Predicted Unemployment', fontsize=15)
plt.axis('equal')
plt.show()
```



Jason Christopher Wijaya

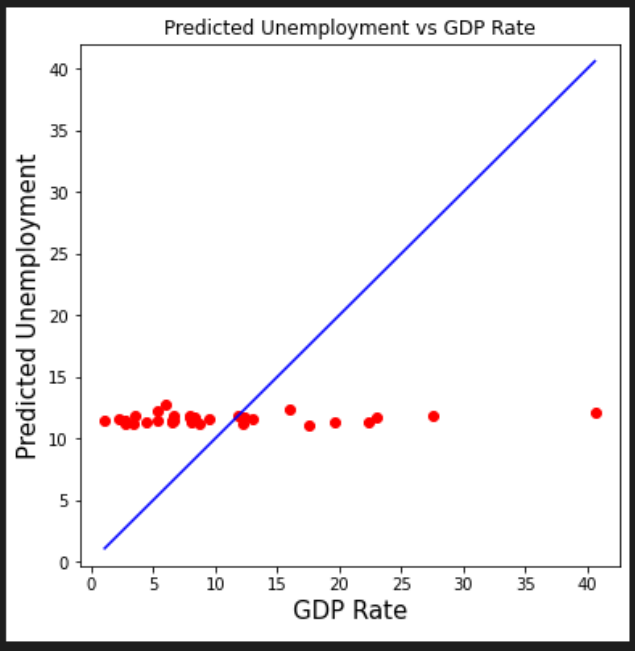
SVR(Support Vector Regression)

```
# Visualizing Plot Predicted Value vs Measure
plt.figure(figsize=(6,6))
plt.scatter(y_test, y_pred, color='red')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.title('Predicted Unemployment vs GDP Rate')
plt.xlabel('GDP Rate', fontsize=15)
plt.ylabel('Predicted Unemployment', fontsize=15)
plt.axis('equal')
plt.show()
```



Ridge

```
[-]
# Visualizing Plot Predicted Value vs Measure
plt.figure(figsize=(6,6))
plt.scatter(y_test, y_pred, color='red')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.title('Predicted Unemployment vs GDP Rate')
plt.xlabel('GDP Rate', fontsize=15)
plt.ylabel('Predicted Unemployment', fontsize=15)
plt.axis('equal')
plt.show()
```



Dari hasil tersebut bisa disimpulkan bahwa dari MSE dan RMSE Linear Regression merupakan model terbaik untuk melakukan Unemployment Rate Prediction, dengan MSE 69.88 dan RMSE 8.36. Untuk Ridge didapatkan RMSE yang sama dengan Linear Regression dan MSE yang memiliki perbedaan sangat minim dengan Linear Regression yaitu 69.89. Terakhir untuk SVR didapatkan MSE 77.59 dan RMSE 8.81. Jika dilihat dari MAE maka SVR merupakan model terbaik untuk melakukan Unemployment Rate Prediction, dengan MAE sebesar 5.80. Untuk Linear Regression dan Ridge memiliki MAE yang sama yaitu 6.45.

Thoughts and Details of future works :

Menurut saya hasil yang telah didapatkan masih belum sempurna dikarenakan error yang masih cukup tinggi, saya akan mencoba untuk mempelajari lebih dalam lagi tentang model-model alternatif lain untuk menyelesaikan masalah regresi. Untuk kedepannya saya dapat mencoba untuk menggunakan metode lain seperti Polinomial regression dan Lasso, dengan begitu saya dapat membandingkan error dan menentukan metode yang terbaik untuk digunakan dalam prediksi Unemployment Rate saat ini.