

## ANALISA STATISTIK DAN GRAFIK PADA DATA TABULAR MENGUNAKAN PYTHON

### A. Data

Untuk penugasan kali ini, saya menggunakan basis data berupa data tabular dengan format (.csv), data tabular adalah data deskriptif yang menyatakan nilai dari data grafis yang diterangkan. Data ini biasanya berbentuk tabel terdiri dari kolom dan baris. Kolom menyatakan jenis data (field), sedangkan baris adalah detail datanya (record). Data yang digunakan merupakan data hasil nilai UN jenjang SMP,SMA, dan SMK di DKI Jakarta pada rentang waktu 2012-2014.

Sumber data: <http://data.jakarta.go.id/dataset/936505d8-7d75-4fd6-9c76-5c2f720e91a0/resource/14d3e94d-7c68-45c7-b181-5abdc5442912/download/DATA-UN-2012-2014-Edited.csv>

Saya menggunakan data tersebut karena selain sesuai dengan peugasan yang diberikan baik dari segi jenis data dan formatnya, data tersebut tidak memuat terlalu banyak variabel sehingga terkesan mudah dan simpel bagi saya.

### B. Penjelasan Script

Script:

```
import pandas as pd
data = pd.read_csv('http://data.jakarta.go.id/dataset/936505d8-7d75-4fd6-9c76-5c2f720e91a0/resource/14d3e94d-7c68-45c7-b181-5abdc5442912/download/DATA-UN-2012-2014-Edited.csv')

data.head()

data.tail()

data.describe()

data.groupby('rata_rata').mean()

data.info()

import matplotlib.pyplot as plt

plt.figure(figsize=(10,8))
plt.plot(data['rata_rata'], label='Rata-Rata')
plt.plot(data['terendah'], label='Terendah')
plt.plot(data['tertinggi'], label='Tertinggi')

plt.xlabel('Jumlah Data')
plt.ylabel('Nilai UN')
plt.title('Grafik Hasil UN SMP,SMA, dan SMK Tahun 2012-2012 di DKI Jakarta')
plt.legend()
```

```
plt.show()
```

1. Dalam penugasan kali ini, hal yang pertama kali dilakukan setelah menemukan data adalah membuat python membaca file yang diinginkan dengan menggunakan python library pandas yang berfungsi untuk data frame. Sehingga diperlukan untuk mengimport library terlebih dahulu dengan menulis:

```
import pandas as pd
```

2. Kedua adalah untuk membaca file csv yang diinginkan digunakan fungsi `pd.read_csv`. File tersebut bisa dari link/url yang memuat file csv tersebut maupun dari folder lokal yang terdapat di dalam device kita sendiri

```
data = pd.read_csv('http://data.jakarta.go.id/dataset/936505d8-7d75-4fd6-9c76-5c2f720e91a0/resource/14d3e94d-7c68-45c7-b181-5abdc5442912/download/DATA-UN-2012-2014-Edited.csv')
```

3. Ketiga adalah menganalisa data tersebut secara statistik yang bisa dilakukan dengan :

- a. Melihat 5 baris pertama/terakhir dari data tersebut menggunakan `data.head()` untuk 5 baris pertama yang outputnya adalah

```
: 1 data.head()
```

```
:
```

	jenjang	tahun	rata_rata	terendah	tertinggi
0	SMP	2012	30.28	12.00	39.80
1	SMA IPA	2012	46.76	14.30	58.45
2	SMA IPS	2012	43.18	12.05	56.60
3	SMA BAHASA	2012	39.97	21.45	54.95
4	SMK	2012	29.87	13.75	38.94

`data.tail()` untuk 5 baris terakhir

```
1 data.tail()
```

	jenjang	tahun	rata_rata	terendah	tertinggi
10	SMP	2014	29.11	5.40	39.75
11	SMA IPA	2014	40.66	14.80	58.05
12	SMA IPS	2014	37.58	2.40	54.95
13	SMA BAHASA	2014	39.63	23.75	53.25
14	SMK	2014	27.58	13.81	38.66

- b. Selanjutnya adalah menggunakan `DataFrame.describe()` yang berguna untuk memperlihatkan parameter statistik dari setiap kolom seperti rata-rata, nilai minimum dan maksimum, persentil 25, 50, dan 75 dari data tersebut

1	<code>data.describe()</code>			
	tahun	rata_rata	terendah	tertinggi
count	15.000000	15.000000	15.000000	15.000000
mean	2013.000000	36.678000	14.387333	49.494667
std	0.845154	6.461949	6.314415	8.687196
min	2012.000000	27.580000	2.400000	38.660000
25%	2012.000000	29.965000	12.025000	39.675000
50%	2013.000000	39.630000	13.810000	54.950000
75%	2014.000000	41.100000	18.225000	56.525000
max	2014.000000	46.760000	24.450000	58.500000

- c. Kemudian jika ingin mengelompokkan suatu data berdasarkan suatu variabel(kolom) yang diinginkan, dapat menggunakan `dataframe.groupby(' '),` sebagai contoh saya ingin mengelompokkan data berdasarkan rata-rata nilai UN

1	<code>data.groupby('rata_rata').mean()</code>		
	tahun	terendah	tertinggi
rata_rata			
27.58	2014	13.81	38.66
29.11	2014	5.40	39.75
29.87	2012	13.75	38.94
29.94	2013	12.55	39.37
29.99	2013	8.65	39.60
30.28	2012	12.00	39.80
37.58	2014	2.40	54.95
39.63	2014	23.75	53.25
39.97	2012	21.45	54.95
40.33	2013	15.30	56.45
40.66	2014	14.80	58.05
41.54	2013	24.45	55.10
43.18	2012	12.05	56.60
43.75	2013	21.15	58.50
46.76	2012	14.30	58.45

Dapat dilihat diatas bahwa variabel rata-rata berubah menjadi objek dari data tersebut sehingga variabel lainnya mengikuti sesuai dengan urutan rata-rata dari yang terkecil sampai terbesar

- d. Selain itu salah fungsi dari pandas yang lain yaitu `dataframe.info()` dapat memperlihatkan rentang indeks, berapa banyak kolom, dan baris pada setiap parameter, tipe data dari setiap parameter, sampai dengan berapa besar konsumsi memori yang digunakan.

```
1 data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15 entries, 0 to 14
Data columns (total 5 columns):
jenjang      15 non-null object
tahun        15 non-null int64
rata_rata    15 non-null float64
terendah     15 non-null float64
tertinggi    15 non-null float64
dtypes: float64(3), int64(1), object(1)
memory usage: 728.0+ bytes
```

4. Kemudian yang keempat adalah menganalisa data tersebut secara grafik yang bisa dilakukan dengan plotting, pada python untuk plotting grafik dapat menggunakan library matplotlib. Sehingga yang terlebih dahulu dilakukan adalah mengimport matplotlib dengan menulis:

```
import matplotlib.pyplot as plt
```

- a. Setelah itu tentukan ukuran dari grafik yang akan diplot dengan menulis

```
plt.figure(figsize=(n,n))
```

n disini adalah ukuran panjang dan lebar dari grafik yang akan dibuat

- b. Kemudian plot masing masing variabel(kolom) dengan memperhatikann info data, data yang bertipe object tidak usah diplot, dan beri label pada masing – masing plot. Plot dilakukan dengan menulis

```
plt.plot(data[' '], label=' ')
```

- c. Beri keterangan pada masing-masing sumbu x dan y dengan menulis

```
plt.xlabel('label x')
plt.ylabel('label y')
```

- d. Beri judul pada grafik dengan menuliskan

```
plt.title('judul grafik')
```

- e. Untuk memberi legenda pada grafik sebagai petunjuk pembeda antar masing-masing garis dapat dilakukan dengan menulis

```
plt.legend()
```

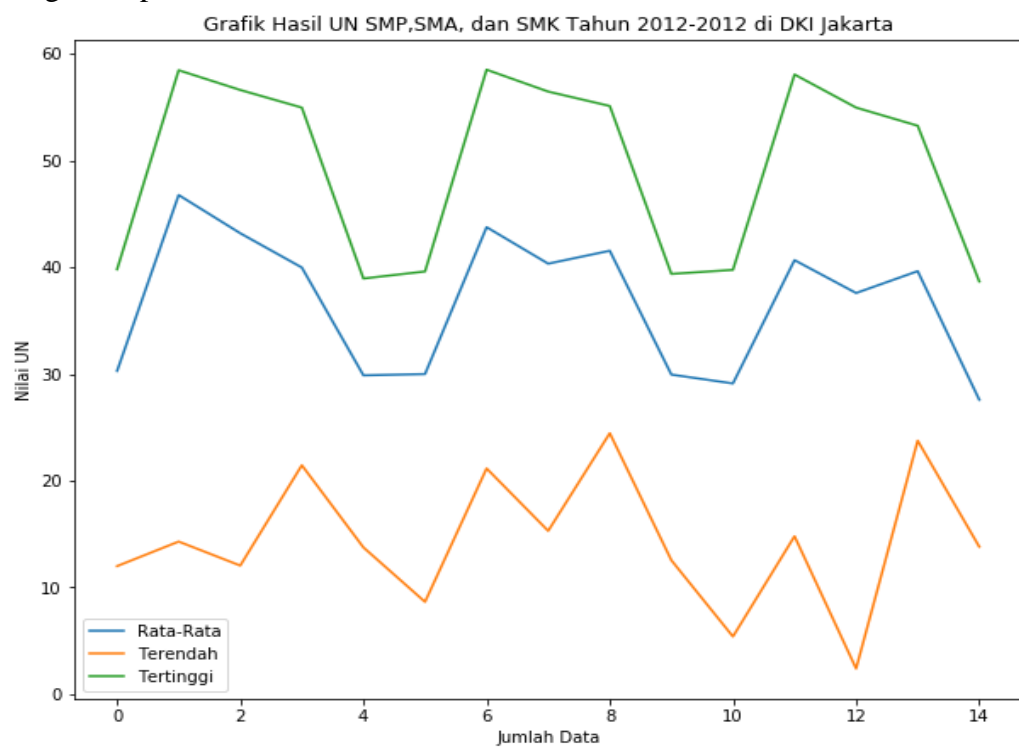
f. Yang terakhir, untuk menampilkan hasil grafik bisa menggunakan

```
plt.show()
```

```
plt.figure(figsize=(10,8))
plt.plot(data['rata_rata'], label='Rata-Rata')
plt.plot(data['terendah'], label='Terendah')
plt.plot(data['tertinggi'], label='Tertinggi')

plt.xlabel('Jumlah Data')
plt.ylabel('Nilai UN')
plt.title('Grafik Hasil UN SMP,SMA, dan SMK Tahun 2012-2012 di DKI Jakarta')
plt.legend()
plt.show()
```

Dengan output:



**SEKIAN DAN TERIMAKASIH**