“World Happiest Report 2023”

Exploring Data Analysis

Analisis data eksploratori adalah sebuah pendekatan untuk menganalisis kumpulan data untuk meringkas karakteristik utamanya, sering kali dengan metode visual. Model statistik dapat digunakan atau tidak, tetapi pada dasarnya EDA adalah untuk melihat apa yang dapat disampaikan oleh data kepada kita di luar pemodelan formal atau pengujian hipotesis.

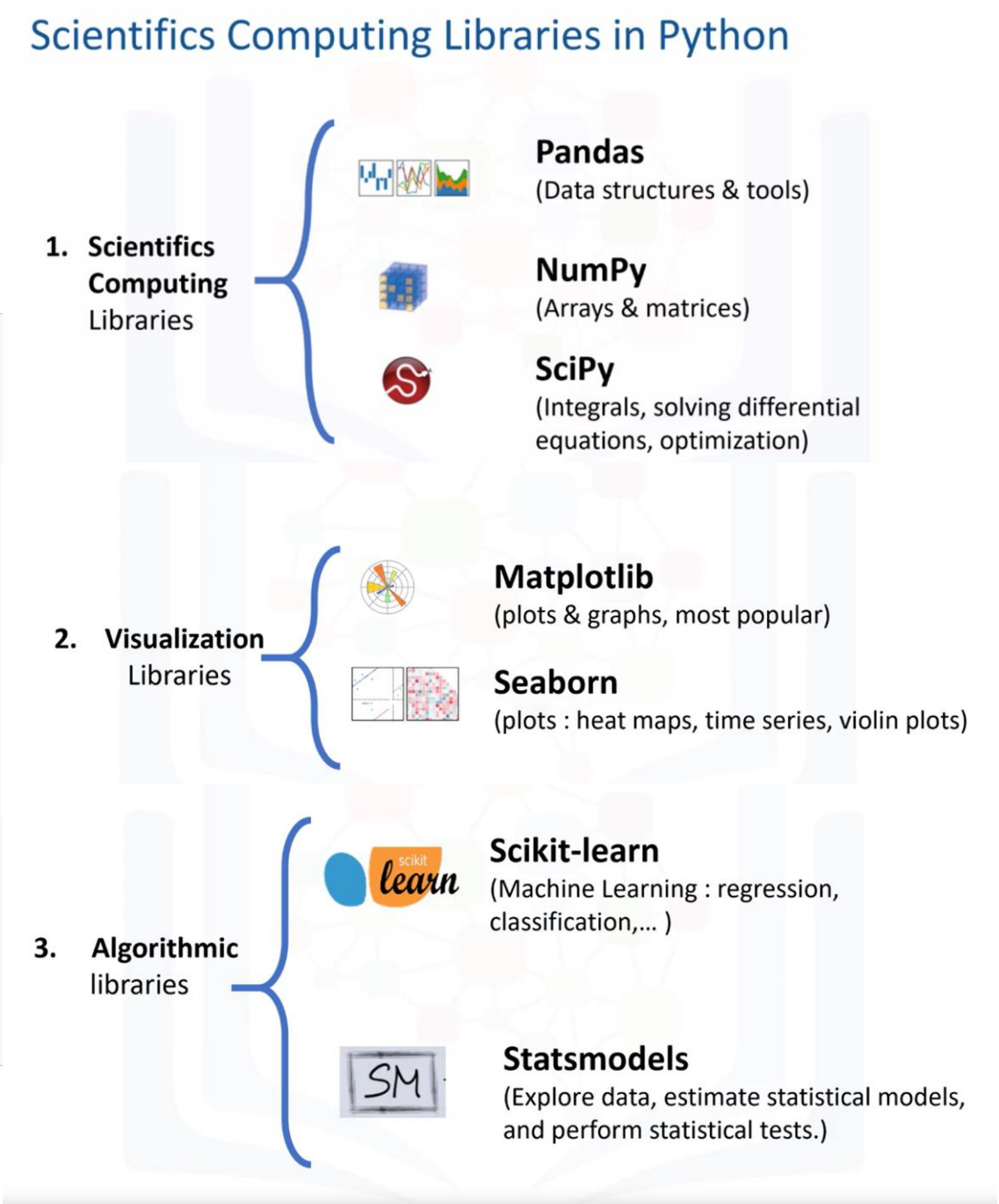
EDA di Python menggunakan visualisasi data untuk menggambar pola dan wawasan yang bermakna. Hal ini juga melibatkan persiapan kumpulan data untuk analisis dengan menghilangkan ketidakteraturan dalam data.

Data Understanding

1. Collect initial data

-Required libraries for EDA :

```
In [1]: #IMPORT LIBRARIES

        #data manipulation
        import pandas as pd
        import numpy as np

        #data viz
        import seaborn as sns
        import matplotlib.pyplot as plt
        import plotly.express as px

        #functions for optimization, stats and signal processing
        from scipy import stats

        #supplies classes for manipulating dates and times
        import datetime

        #apply some cool styling
        plt.style.use("ggplot")

        #for Warnings are messages about errors or anomalies
        import warnings
        warnings.filterwarnings("ignore")
```



- Describe data

```
In [5]: import gdown
```

download a file from Google Drive

```
In [11]: 'https://drive.google.com/file/d/1FrnCZDorxY_vwVY2jjCp--WJ0ukbNyv9/view?usp=share_link'

Out[11]: 'https://drive.google.com/file/d/1FrnCZDorxY_vwVY2jjCp--WJ0ukbNyv9/view?usp=share_link'
```

Sample data

```
In [12]: output = "data.csv"

In [13]: # Download file drive dengan ID
         id = "1FrnCZDorxY_vwVY2jjCp--WJ0ukbNyv9"
         gdown.download(id=id, output=output, quiet=False)

         Downloading...
         From: https://drive.google.com/uc?id=1FrnCZDorxY_vwVY2jjCp--WJ0ukbNyv9
         To: C:\Users\widya\Project File\data.csv
         100%|██████████| 16.8k/16.8k [00:00<00:00, 1.94MB/s]

Out[13]: 'data.csv'
```

Did not display the copyright and version at Python startup in interactive mode

Membaca file .csv

```
In [14]: data = pd.read_csv('C:/Users/widya/Project File/data.csv')
```

filepath

mengonversi dataset menjadi dataframe Pandas

Membaca file .csv

- Describe data

Menampilkan dataset

```
In [10]: #Menampilkan Data
         data
```

Out[10]:

	Country name	Ladder score	Standard error of ladder score	upperwhisker	lowerwhisker	Logged GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption	Ladder score in Dystopia	Explained by: Log GDP per capita	Explain by: Soc supp
0	Finland	7.804	0.036	7.875	7.733	10.792	0.969	71.150	0.961	-0.019	0.182	1.778	1.888	1.5
1	Denmark	7.586	0.041	7.667	7.506	10.962	0.954	71.250	0.934	0.134	0.196	1.778	1.949	1.5
2	Iceland	7.530	0.049	7.625	7.434	10.896	0.983	72.050	0.936	0.211	0.668	1.778	1.926	1.6
3	Israel	7.473	0.032	7.535	7.411	10.639	0.943	72.697	0.809	-0.023	0.708	1.778	1.833	1.5
4	Netherlands	7.403	0.029	7.460	7.346	10.942	0.930	71.550	0.887	0.213	0.379	1.778	1.942	1.4
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
132	Congo (Kinshasa)	3.207	0.095	3.394	3.020	7.007	0.652	55.375	0.664	0.086	0.834	1.778	0.531	0.7
133	Zimbabwe	3.204	0.061	3.323	3.084	7.641	0.690	54.050	0.654	-0.046	0.766	1.778	0.758	0.8
134	Sierra Leone	3.138	0.082	3.299	2.976	7.394	0.555	54.900	0.660	0.105	0.858	1.778	0.670	0.5
135	Lebanon	2.392	0.044	2.479	2.305	9.478	0.530	66.149	0.474	-0.141	0.891	1.778	1.417	0.4
136	Afghanistan	1.859	0.033	1.923	1.795	7.324	0.341	54.712	0.382	-0.081	0.847	1.778	0.645	0.0

137 rows × 19 columns

- Explore data:

Mencetak informasi tentang dataset

```
In [9]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 137 entries, 0 to 136
Data columns (total 19 columns):
 #   Column                                    Non-Null Count  Dtype
---  ------                                    --------------  -----
 0   Country name                              137 non-null    object
 1   Ladder score                              137 non-null    float64
 2   Standard error of ladder score            137 non-null    float64
 3   upperwhisker                              137 non-null    float64
 4   lowerwhisker                              137 non-null    float64
 5   Logged GDP per capita                     137 non-null    float64
 6   Social support                            137 non-null    float64
 7   Healthy life expectancy                   136 non-null    float64
 8   Freedom to make life choices              137 non-null    float64
 9   Generosity                                137 non-null    float64
 10  Perceptions of corruption                 137 non-null    float64
 11  Ladder score in Dystopia                  137 non-null    float64
 12  Explained by: Log GDP per capita          137 non-null    float64
 13  Explained by: Social support              137 non-null    float64
 14  Explained by: Healthy life expectancy     136 non-null    float64
 15  Explained by: Freedom to make life choices 137 non-null   float64
 16  Explained by: Generosity                  137 non-null    float64
 17  Explained by: Perceptions of corruption   137 non-null    float64
 18  Dystopia + residual                       136 non-null    float64
dtypes: float64(18), object(1)
memory usage: 20.5+ KB
```

Menampilkan daftar tipe data setiap kolom

```
In [10]: data.dtypes
```

```
Out[10]: Country name                                object
         Ladder score                               float64
         Standard error of ladder score             float64
         upperwhisker                               float64
         lowerwhisker                               float64
         Logged GDP per capita                      float64
         Social support                             float64
         Healthy life expectancy                    float64
         Freedom to make life choices               float64
         Generosity                                 float64
         Perceptions of corruption                  float64
         Ladder score in Dystopia                   float64
         Explained by: Log GDP per capita           float64
         Explained by: Social support               float64
         Explained by: Healthy life expectancy      float64
         Explained by: Freedom to make life choices float64
         Explained by: Generosity                   float64
         Explained by: Perceptions of corruption    float64
         Dystopia + residual                        float64
         dtype: object
```

- Explore data:

Menampilkan statistik deksriptif dari dataset

```
In [11]: data.describe()
```

Out[11]:

	Ladder score	Standard error of ladder score	upperwhisker	lowerwhisker	Logged GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption	Ladder score in Dystopia	Explained by: Log GDP per capita
count	137.000000	137.000000	137.000000	137.000000	137.000000	137.000000	136.000000	137.000000	137.000000	137.000000	1.370000e+02	137.000000
mean	5.539796	0.064715	5.666526	5.412971	9.449796	0.799073	64.967632	0.787394	0.022431	0.725401	1.778000e+00	1.406985
std	1.139929	0.023031	1.117421	1.163724	1.207302	0.129222	5.750390	0.112371	0.141707	0.176956	2.897173e-15	0.432963
min	1.859000	0.029000	1.923000	1.795000	5.527000	0.341000	51.530000	0.382000	-0.254000	0.146000	1.778000e+00	0.000000
25%	4.724000	0.047000	4.980000	4.496000	8.591000	0.722000	60.648500	0.724000	-0.074000	0.668000	1.778000e+00	1.099000
50%	5.684000	0.060000	5.797000	5.529000	9.567000	0.827000	65.837500	0.801000	0.001000	0.774000	1.778000e+00	1.449000
75%	6.334000	0.077000	6.441000	6.243000	10.540000	0.896000	69.412500	0.874000	0.117000	0.846000	1.778000e+00	1.798000
max	7.804000	0.147000	7.875000	7.733000	11.660000	0.983000	77.280000	0.961000	0.531000	0.929000	1.778000e+00	2.200000

- Explore data:

Menampilkan baris pertama dari baris ke-n pada dataset

```
In [12]: data.head(10)
```

Out[12]:

	Country name	Ladder score	Standard error of ladder score	upperwhisker	lowerwhisker	Logged GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption	Ladder score in Dystopia	Explained by: Log GDP per capita	Explained by: Socia suppor
0	Finland	7.804	0.036	7.875	7.733	10.792	0.969	71.150	0.961	-0.019	0.182	1.778	1.888	1.58
1	Denmark	7.586	0.041	7.667	7.506	10.962	0.954	71.250	0.934	0.134	0.196	1.778	1.949	1.54
2	Iceland	7.530	0.049	7.625	7.434	10.896	0.983	72.050	0.936	0.211	0.668	1.778	1.926	1.62
3	Israel	7.473	0.032	7.535	7.411	10.639	0.943	72.697	0.809	-0.023	0.708	1.778	1.833	1.52
4	Netherlands	7.403	0.029	7.460	7.346	10.942	0.930	71.550	0.887	0.213	0.379	1.778	1.942	1.48
5	Sweden	7.395	0.037	7.468	7.322	10.883	0.939	72.150	0.948	0.165	0.202	1.778	1.921	1.51
6	Norway	7.315	0.044	7.402	7.229	11.088	0.943	71.500	0.947	0.141	0.283	1.778	1.994	1.52
7	Switzerland	7.240	0.043	7.324	7.156	11.164	0.920	72.900	0.891	0.027	0.266	1.778	2.022	1.46
8	Luxembourg	7.228	0.069	7.363	7.093	11.660	0.879	71.675	0.915	0.024	0.345	1.778	2.200	1.35
9	New Zealand	7.123	0.038	7.198	7.048	10.662	0.952	70.350	0.887	0.175	0.271	1.778	1.842	1.54

Menampilkan data terbawah pada dataset

```
In [13]: data.tail(10)
```

Out[13]:

	Country name	Ladder score	Standard error of ladder score	upperwhisker	lowerwhisker	Logged GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption	Ladder score in Dystopia	Explained by: Log GDP per capita	Explain by: Soc supp
127	Zambia	3.982	0.094	4.167	3.797	8.074	0.694	55.032	0.791	0.098	0.818	1.778	0.914	0.8
128	Tanzania	3.694	0.075	3.840	3.547	7.857	0.653	59.401	0.838	0.182	0.554	1.778	0.836	0.7
129	Comoros	3.545	0.117	3.774	3.317	8.075	0.471	59.425	0.470	-0.014	0.727	1.778	0.914	0.3
130	Malawi	3.495	0.090	3.671	3.320	7.302	0.531	58.475	0.750	0.005	0.749	1.778	0.637	0.4
131	Botswana	3.435	0.136	3.702	3.168	9.629	0.753	54.725	0.742	-0.215	0.830	1.778	1.471	1.0
132	Congo (Kinshasa)	3.207	0.095	3.394	3.020	7.007	0.652	55.375	0.664	0.086	0.834	1.778	0.531	0.7
133	Zimbabwe	3.204	0.061	3.323	3.084	7.641	0.690	54.050	0.654	-0.046	0.766	1.778	0.758	0.8
134	Sierra Leone	3.138	0.082	3.299	2.976	7.394	0.555	54.900	0.660	0.105	0.858	1.778	0.670	0.5
135	Lebanon	2.392	0.044	2.479	2.305	9.478	0.530	66.149	0.474	-0.141	0.891	1.778	1.417	0.4
136	Afghanistan	1.859	0.033	1.923	1.795	7.324	0.341	54.712	0.382	-0.081	0.847	1.778	0.645	0.0

Data Preparation

- Handling Missing Data

Menampilkan summary null value

```
In [16]: #Missing Values
         data.isnull().sum()
```

```
Out[16]: Negara                                    0
         Peringkat                                 0
         Kesalahan Umum dalam peringkat            0
         upperwhisker                              0
         lowerwhisker                              0
         Logged GDP per capita                     0
         Social support                            0
         Healthy life expectancy                   1
         Freedom to make life choices              0
         Generosity                                0
         Perceptions of corruption                 0
         Ladder score in Dystopia                  0
         Explained by: Log GDP per capita          0
         Explained by: Social support              0
         Explained by: Healthy life expectancy     1
         Explained by: Freedom to make life choices 0
         Explained by: Generosity                  0
         Explained by: Perceptions of corruption   0
         Dystopia + residual                       1
         dtype: int64
```

Mengganti menggantikan atau menambahkan sebuah karakter di dalam sebuah string.

```
In [17]: #Replace null values
         data.replace(np.nan,'0',inplace = True)

         #Check the changes again
         data.isnull().sum()
```

Mengganti NaN value dengan 0

```
Out[17]: Negara                                    0
         Peringkat                                 0
         Kesalahan Umum dalam peringkat            0
         upperwhisker                              0
         lowerwhisker                              0
         Logged GDP per capita                     0
         Social support                            0
         Healthy life expectancy                   0
         Freedom to make life choices              0
         Generosity                                0
         Perceptions of corruption                 0
         Ladder score in Dystopia                  0
         Explained by: Log GDP per capita          0
         Explained by: Social support              0
         Explained by: Healthy life expectancy     0
         Explained by: Freedom to make life choices 0
         Explained by: Generosity                  0
         Explained by: Perceptions of corruption   0
         Dystopia + residual                       0
         dtype: int64
```

- Find Duplicates

```
In [18]: #Find the duplicates

         data.duplicated().sum()

Out[18]: 0
```

Menampilkan summary data duplikat

- Creating new column

```
[41] # Create rank column
     data['rank'] = data['Ladder score'].rank(ascending=False)
     data['rank'] = data['rank'].astype(int)
     data
```

Mengubah tipe data menjadi int

Membuat kolom baru bernama 'rank' dengan data 'Ladder score' descending order

- Rename columns

Nama kolom lama

Nama kolom baru

Memanggil kembali dataset

Renaming columns

In [14]:
```
#rename fitur
data = data.rename(columns ={'Country name':'Negara','Ladder score':'Peringkat',
                             'Standard error of ladder score':'Kesalahan Umum dalam peringkat'})
```

In [15]:
```
#panggil data
data
```

Out[15]:

	Negara	Peringkat	Kesalahan Umum dalam peringkat	upperwhisker	lowerwhisker	Logged GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption	Ladder score in Dystopia	Explained by: Log GDP per capita	Exp by: s
0	Finland	7.804	0.036	7.875	7.733	10.792	0.969	71.150	0.961	-0.019	0.182	1.778	1.888	
1	Denmark	7.586	0.041	7.667	7.506	10.962	0.954	71.250	0.934	0.134	0.196	1.778	1.949	
2	Iceland	7.530	0.049	7.625	7.434	10.896	0.983	72.050	0.936	0.211	0.668	1.778	1.926	
3	Israel	7.473	0.032	7.535	7.411	10.639	0.943	72.697	0.809	-0.023	0.708	1.778	1.833	
4	Netherlands	7.403	0.029	7.460	7.346	10.942	0.930	71.550	0.887	0.213	0.379	1.778	1.942	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
132	Congo (Kinshasa)	3.207	0.095	3.394	3.020	7.007	0.652	55.375	0.664	0.086	0.834	1.778	0.531	
133	Zimbabwe	3.204	0.061	3.323	3.084	7.641	0.690	54.050	0.654	-0.046	0.766	1.778	0.758	
134	Sierra Leone	3.138	0.082	3.299	2.976	7.394	0.555	54.900	0.660	0.105	0.858	1.778	0.670	
135	Lebanon	2.392	0.044	2.479	2.305	9.478	0.530	66.149	0.474	-0.141	0.891	1.778	1.417	
136	Afghanistan	1.859	0.033	1.923	1.795	7.324	0.341	54.712	0.382	-0.081	0.847	1.778	0.645	

137 rows × 19 columns

Mencari rata-rata dari suatu data

```
In [20]: data.mean()
```

```
Out[20]: Peringkat                                  5.539796
         Kesalahan Umum dalam peringkat             0.064715
         upperwhisker                               5.666526
         lowerwhisker                               5.412971
         Logged GDP per capita                      9.449796
         Social support                             0.799073
         Freedom to make life choices               0.787394
         Generosity                                 0.022431
         Perceptions of corruption                  0.725401
         Ladder score in Dystopia                   1.778000
         Explained by: Log GDP per capita           1.406985
         Explained by: Social support               1.156212
         Explained by: Freedom to make life choices 0.540000
         Explained by: Generosity                   0.148474
         Explained by: Perceptions of corruption    0.145898
         dtype: float64
```

Mencari nilai tengah dari suatu data

```
In [21]: data.median()
```

```
Out[21]: Peringkat                                  5.684
         Kesalahan Umum dalam peringkat             0.060
         upperwhisker                               5.797
         lowerwhisker                               5.529
         Logged GDP per capita                      9.567
         Social support                             0.827
         Healthy life expectancy                   65.825
         Freedom to make life choices               0.801
         Generosity                                 0.001
         Perceptions of corruption                  0.774
         Ladder score in Dystopia                   1.778
         Explained by: Log GDP per capita           1.449
         Explained by: Social support               1.227
         Explained by: Healthy life expectancy      0.389
         Explained by: Freedom to make life choices 0.557
         Explained by: Generosity                   0.137
         Explained by: Perceptions of corruption    0.111
         Dystopia + residual                        1.845
         dtype: float64
```

Menghitung variance dari suatu data

```
In [22]: data.var()
```

```
Out[22]: Peringkat                                  1.299438e+00
         Kesalahan Umum dalam peringkat             5.304257e-04
         upperwhisker                               1.248629e+00
         lowerwhisker                               1.354254e+00
         Logged GDP per capita                      1.457579e+00
         Social support                             1.669838e-02
         Freedom to make life choices               1.262727e-02
         Generosity                                 2.008078e-02
         Perceptions of corruption                  3.131334e-02
         Ladder score in Dystopia                   8.393611e-30
         Explained by: Log GDP per capita           1.874567e-01
         Explained by: Social support               1.064864e-01
         Explained by: Freedom to make life choices 2.235065e-02
         Explained by: Generosity                   5.784134e-03
         Explained by: Perceptions of corruption    1.605871e-02
         dtype: float64
```

Menghitung standar deviasi dari suatu data

```
In [23]: data.std()
```

```
Out[23]: Peringkat                                  1.139929e+00
         Kesalahan Umum dalam peringkat             2.303097e-02
         upperwhisker                               1.117421e+00
         lowerwhisker                               1.163724e+00
         Logged GDP per capita                      1.207302e+00
         Social support                             1.292222e-01
         Freedom to make life choices               1.123711e-01
         Generosity                                 1.417067e-01
         Perceptions of corruption                  1.769558e-01
         Ladder score in Dystopia                   2.897173e-15
         Explained by: Log GDP per capita           4.329627e-01
         Explained by: Social support               3.263225e-01
         Explained by: Freedom to make life choices 1.495013e-01
         Explained by: Generosity                   7.605349e-02
         Explained by: Perceptions of corruption    1.267230e-01
         dtype: float64
```

Feature Understanding

Visualization for Univariate Analysis :

1. Box Plot
2. Histogram
3. Pie Charts

Visualization for Bivariate analysis :

1. Correlation matrix
2. Regression Plot

- Correlation

Korelasi mengacu pada sejauh mana sepasang variabel berhubungan secara linear

Menampilkan nilai korelasi (Pearson Correlation)

```
In [19]: data.corr()
```

Out[19]:

	Peringkat	Kesalahan Umum dalam peringkat	upperwhisker	lowerwhisker	Logged GDP per capita	Social support	Freedom to make life choices	Generosity	Perceptions of corruption	Ladder score in Dystopia	Explained by: Log GDP per capita	Explained by: Social support	Expl Fre to ch
Peringkat	1.000000	-0.512628	0.999401	0.999448	0.784367	0.834532	0.662924	0.044082	-0.471911	NaN	0.784342	0.834604	0.6
Kesalahan Umum dalam peringkat	-0.512628	1.000000	-0.482622	-0.540876	-0.584666	-0.472056	-0.297075	0.093627	0.305107	NaN	-0.584614	-0.472235	-0.2
upperwhisker	0.999401	-0.482622	1.000000	0.997700	0.776570	0.832243	0.664284	0.048691	-0.469169	NaN	0.776546	0.832310	0.6
lowerwhisker	0.999448	-0.540876	0.997700	1.000000	0.790993	0.835762	0.660893	0.039581	-0.474083	NaN	0.790967	0.835840	0.6
Logged GDP per capita	0.784367	-0.584666	0.776570	0.790993	1.000000	0.738069	0.451439	-0.156456	-0.436961	NaN	1.000000	0.738095	0.4
Social support	0.834532	-0.472056	0.832243	0.835762	0.738069	1.000000	0.541630	0.036574	-0.272490	NaN	0.737967	0.999997	0.5
Freedom to make life choices	0.662924	-0.297075	0.664284	0.660893	0.451439	0.541630	1.000000	0.170229	-0.383786	NaN	0.451389	0.541592	0.9
Generosity	0.044082	0.093627	0.048691	0.039581	-0.156456	0.036574	0.170229	1.000000	-0.122653	NaN	-0.156405	0.036350	0.1
Perceptions of corruption	-0.471911	0.305107	-0.469169	-0.474083	-0.436961	-0.272490	-0.383786	-0.122653	1.000000	NaN	-0.436934	-0.272657	-0.3
Ladder score in Dystopia	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
Explained by: Log GDP per capita	0.784342	-0.584614	0.776546	0.790967	1.000000	0.737967	0.451389	-0.156405	-0.436934	NaN	1.000000	0.737993	0.4
Explained by: Social support	0.834604	-0.472235	0.832310	0.835840	0.738095	0.999997	0.541592	0.036350	-0.272657	NaN	0.737993	1.000000	0.5
Explained by: Freedom to make life choices	0.662909	-0.296961	0.664272	0.660873	0.451456	0.541869	0.999994	0.170230	-0.383800	NaN	0.451405	0.541832	1.0
Explained by: Generosity	0.043680	0.093585	0.048279	0.039188	-0.156831	0.035961	0.169685	0.999990	-0.122016	NaN	-0.156780	0.035736	0.1
Explained by: Perceptions of corruption	0.471913	-0.305107	0.469173	0.474086	0.437006	0.272372	0.383658	0.122345	-0.999996	NaN	0.436979	0.272538	0.3

Histogram

adalah alat visualisasi klasik yang merepresentasikan distribusi satu atau beberapa variabel dengan menghitung jumlah pengamatan yang termasuk dalam tempat sampah diskrit. Histogram menampilkan Berapa kali (frekueni) setiap nilai muncul dalam kumpulan data.

Data = Nama Data frame

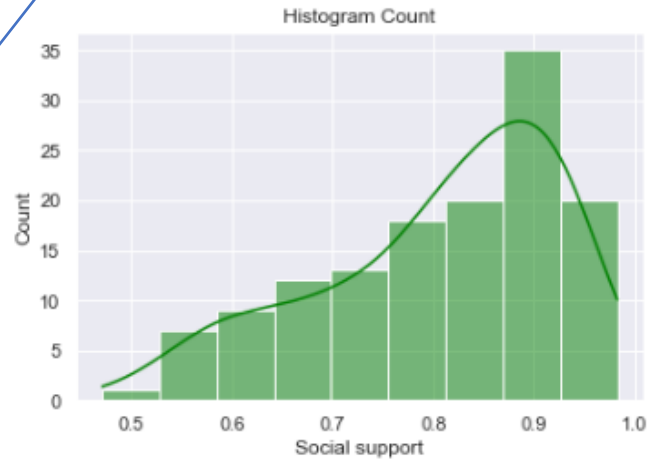
Jika True, dapat menghitung estimasi kepadatan kernel untuk memperhalus distribusi dan menampilkannya pada plot sebagai (satu atau lebih) garis.

Judul dari Histogram

Nama Judul

Kolom dari DF

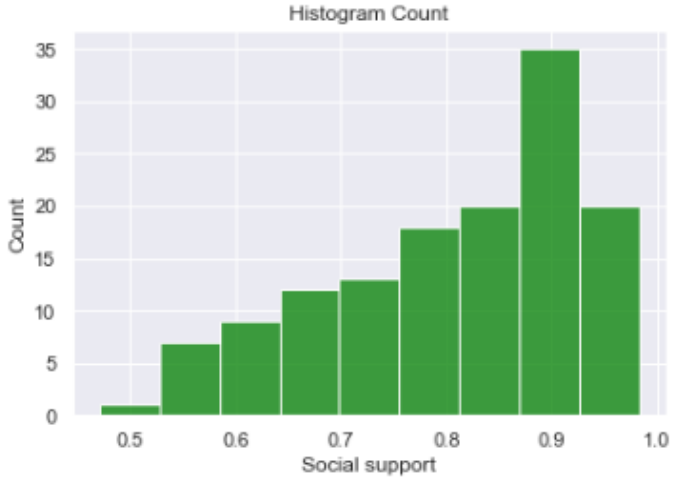
```
In [81]: plt.title("Histogram Count")
         sns.histplot(data=data, x="Social support", kde=True , color ='green')

Out[81]: <AxesSubplot:title={'center':'Histogram Count'}, xlabel='Social support', ylabel='Count'>
```



Warna

Fungsi untuk mewakili distribusi data univariat

Jika kde = False



Boxplot

1. Boxplot adalah ukuran seberapa baik distribusi data dalam kumpulan data.
2. Grafik ini membagi kumpulan data menjadi tiga kuartil.
3. Grafik ini mewakili nilai minimum, maksimum, median, kuartil pertama, dan kuartil ketiga dalam kumpulan data

data = data frame

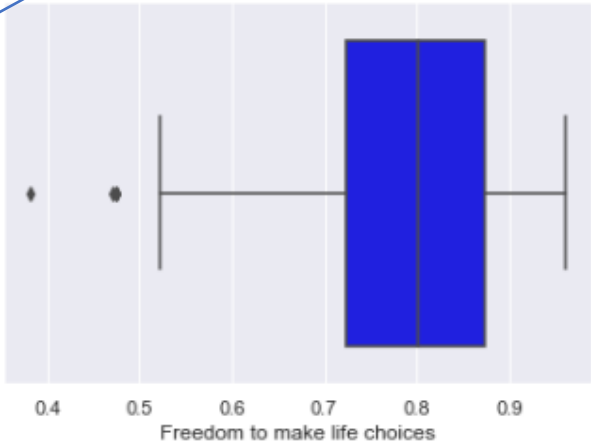
warna untuk elemen-elemen dalam plot

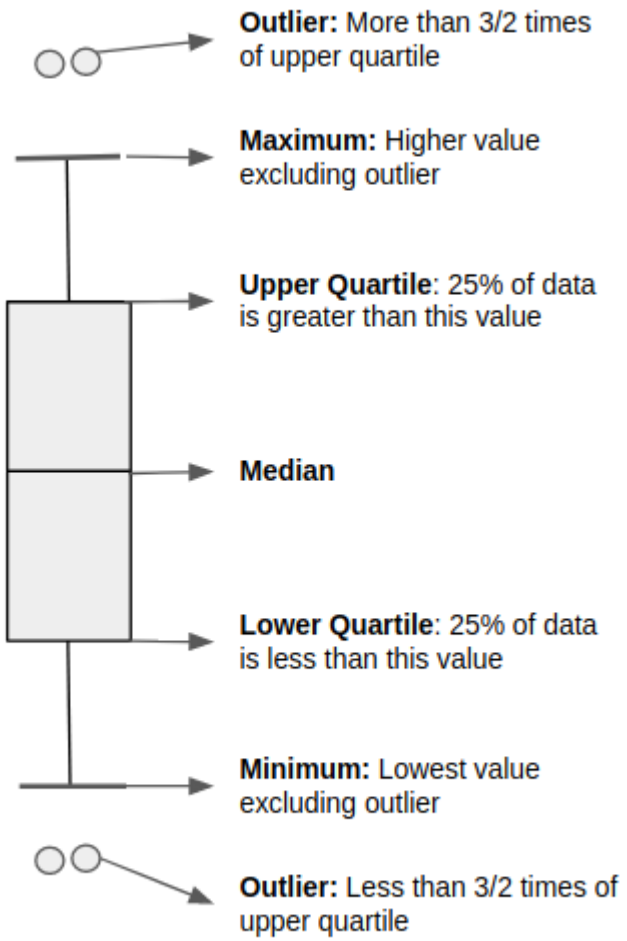
kolom referensi dari data frame

Fungsi Boxplot

```
In [83]: sns.boxplot(data['Freedom to make life choices'],color ='blue')
         plt.show()
```

digunakan untuk menampilkan semua gambar





Pie Chart

Diagram Lingkaran memberikan informasi tambahan mengenai persentase keberadaan setiap kategori dalam data yang berarti kategori mana yang mendapatkan bobot berapa dalam data. Dengan kita menganalisis data pada satu variabel/kolom dari kumpulan data, ini dikenal sebagai Analisis Univariat.

Fungsi Membuat pie chart

Nilai dari kolom atau array_like ini digunakan sebagai label untuk sektor

```
In [39]: fig = px.pie(data.head(35), values='Freedom to make life choices', names='Negara', title='World Happiness Report 2023 ')
         fig.show()
```

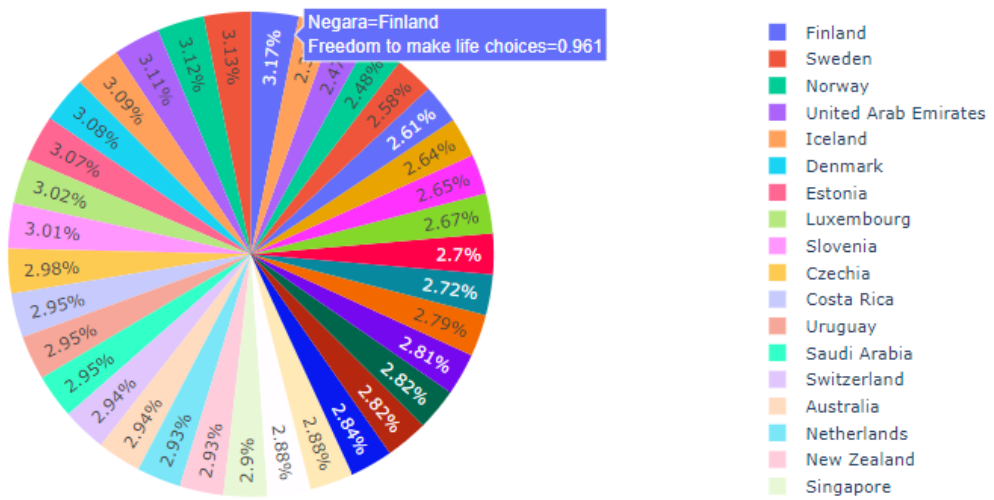
Judul figure

the function to create a new figure

Menampilkan figure

Data frame dengan 35 data teratas

Nilai dari kolom ini atau array_like digunakan untuk mengatur nilai yang terkait dengan sektor



Visualization for Bivariate analysis :

- Correlation matrix:

Dengan menemukan korelasi antara semua variabel numerik yang ada dalam kumpulan data untuk membangun matriks korelasi dan untuk mengukur sejauh mana satu variabel berubah sehubungan dengan variabel lain

Ukuran heatmap

pemformatan string untuk anotasi, yaitu satu angka dibelakang koma untuk tipe data float

```
In [47]: f,ax=plt.subplots(figsize = (18,18))
         sns.heatmap(data.corr(),annot= True,linewidths=0.5,fmt = ".1f",ax=ax)
         plt.xticks(rotation=90)
         plt.yticks(rotation=0)
         plt.title('Correlation Map')
         plt.savefig('graph.png')
         plt.show()
```

Heatmap disimpan sebagai 'graph.png'

Nama heat map

Jika false, maka heatmap tidak akan diberi angka desimal yang menyatakan hubungan variabel satu dengan lainnya

Visualization for Bivariate analysis :

- Regression Plot

Ini digunakan untuk memahami hubungan antara dua variabel. Hal ini sama dengan menganalisis korelasi antara dua variabel, tetapi digunakan untuk menganalisis hanya dua variabel, tidak seperti matriks korelasi. Hal ini dapat dilihat sebagai grafik normal di mana kita memplot semua titik data dan kemudian menemukan garis yang paling sesuai.

correspond with column names in

dataframe

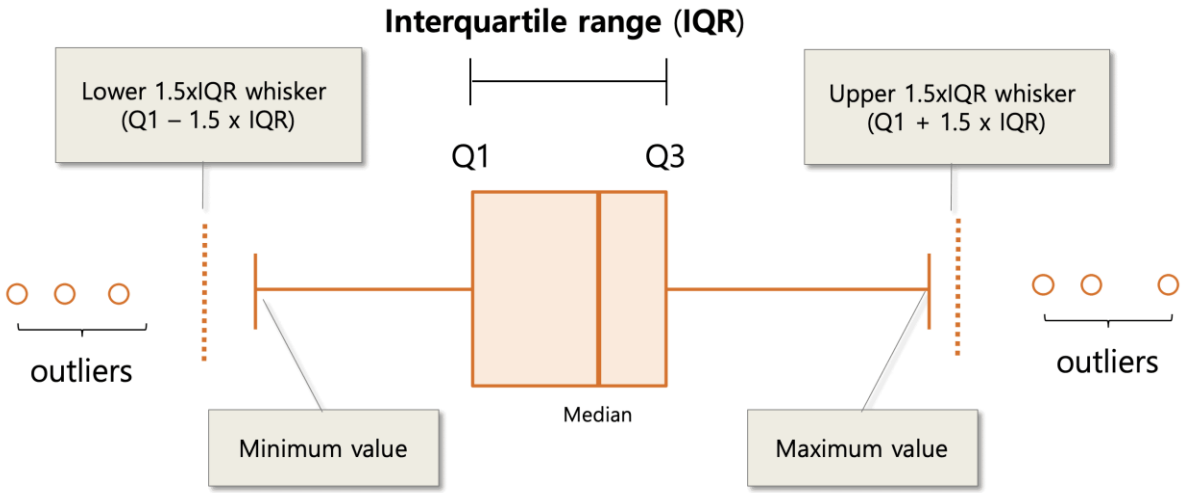
```
In [68]: sns.regplot(x='Freedom to make life choices',y='Perceptions of corruption',data=data)

Out[68]: <AxesSubplot:xlabel='Freedom to make life choices', ylabel='Perceptions of corruption'>
```

Evaluation

Outlier Treatment

Dengan asumsi bahwa dataset Anda terlalu besar untuk menghapus pencilan secara manual baris demi baris, metode statistik akan diperlukan. Ada beberapa pendekatan yang umum digunakan sebagai berikut:

1. Standar deviasi (Standard deviation), yaitu Menghapus nilai yang memiliki sejumlah deviasi standar tertentu dari rata-rata, jika data memiliki distribusi Gaussian
2. Deteksi pencilan otomatis (Automatic outlier detection) yaitu Melatih model pembelajaran mesin pada sekumpulan pengamatan normal yang lebih kecil yang kemudian dapat memprediksi titik data di luar sekumpulan normal ini
3. Rentang interkuartil (Interquartile range) yaitu Menghapus nilai yang berada di atas persentil ke-75 atau di bawah persentil ke-25, tidak mengharuskan data menjadi Gaussian



Outlier Treatment

Rentang interkuartil (Interquartile range)

- Menangani outlier yang telah kita deteksi menggunakan Boxplot di bagian sebelumnya.
- Dengan menggunakan IQR, kita dapat mengikuti pendekatan di bawah ini untuk mengganti pencilan dengan nilai NULL:
- Hitung kuartil pertama dan ketiga (Q1 dan Q3).
- Selanjutnya, Evaluasi rentang antar kuartil, IQR = Q3-Q1.
- Perkirakan batas bawah, batas bawah = Q11.5 Perkirakan batas atas,
- batas atas = Q31.5 Ganti titik data yang berada di luar batas bawah dan batas atas dengan nilai NULL.

Variabel untuk Q3, Q1 dan interkuartil

Nama Kolum

Properti dari objek DataFrame

```
In [32]: for x in ['Freedom to make life choices']:
             q75,q25 = np.percentile(data.loc[:,x],[75,25])
             intr_qr = q75-q25
```

untuk kalkulasi persentil

scale, kolom x

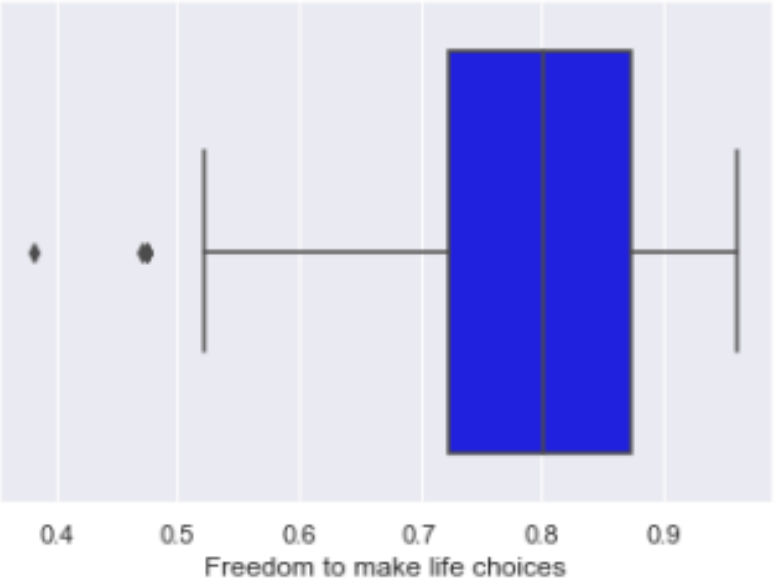
Menghitung batas atas = Q31.5

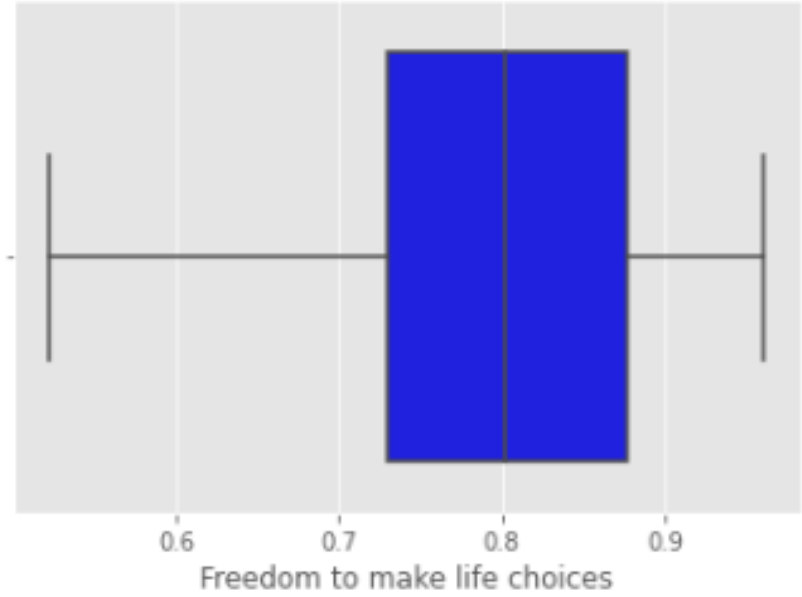
```
max = q75+(1.5*intr_qr)
min = q25-(1.5*intr_qr)

data.loc[data[x] < min,x] = np.nan
data.loc[data[x] > max,x] = np.nan
```

menghitung batas bawah = Q11.5

Batas atas , kolum x

Sebagai pengganti untuk mendeklarasikan nilai numerik yang nilainya hilang dalam sebuah array





Terima Kasih