INSTITUT TEKNOLOGI BANDUNG

1920

Analisis Data dengan Python

IF2220 – Probabilitas dan Statistika

Fariska Z. Ruskanda, S.T., M.T.
(fariska@informatika.org)

KK IF -Teknik Informatika - STEI ITB

Analisis Data dengan Python

- Statistik yang Penting
- Mean, Variance
- Quartile & Boxplot
- Skewness & Kurtosis
- Histogram & Scatter Plot

Statistik yang Penting

Bila $X_1, X_2, \ldots, X_n$, sampel acak ukuran n:

- **Rataan sampel** dinyatakan oleh statistik:

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n}$$

- **Modus** = nilai X_i yang paling banyak muncul
- **Median**

$$X_{med} = \begin{cases} X_{(n+1)/2} & bila\ n\ ganjil \\ \dfrac{X_{n/2} + X_{(n/2)+1}}{2} & bila\ n\ genap \end{cases}$$

Dimana $X_1, X_2, \ldots, X_n$ diurut membesar

Statistik yang Penting (2)

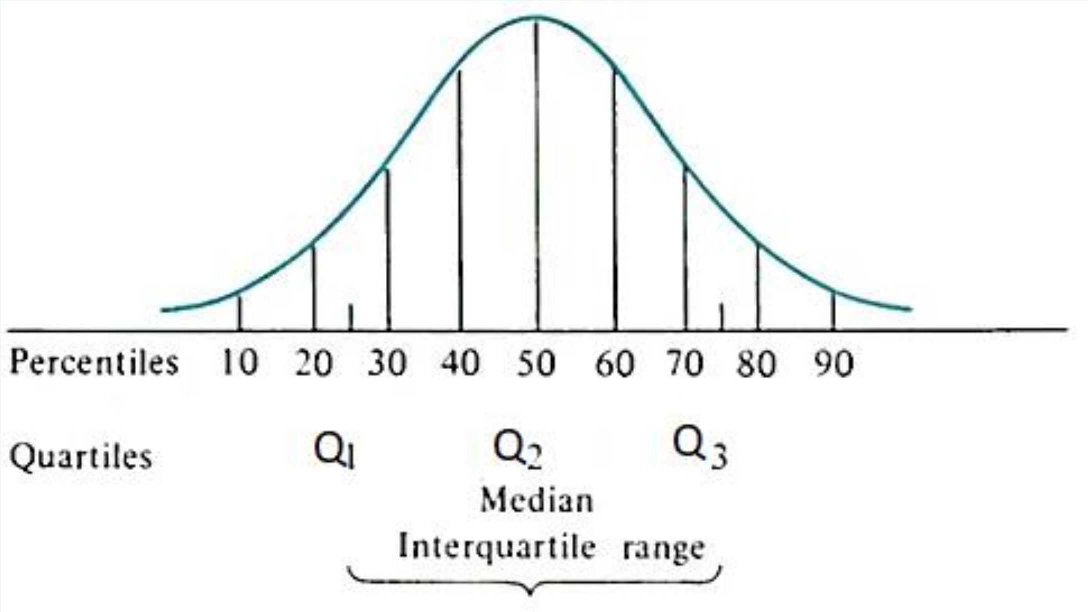
- **Jangkauan** dari sampel acak $X_1, X_2, \ldots, X_n$ didefinisikan sebagai statistik $J = X_{(n)} - X_{(1)}$, bila $X_{(n)}$ dan $X_{(1)}$ menyatakan masing-masing nilai terbesar dan terkecil dari sampel.
- **Variansi** sampel dinyatakan oleh

$$S^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n-1}$$

- **Simpangan baku** $S = \sqrt{S^2}$

Statistik yang Penting (3)

- **Persentil rank** , P_{10} = nilai X_i di posisi persentil 10 % dari semua data $X_1, X_2, \ldots, X_n$ diurut membesar.
- **Kuartil**, Q_1, Q_2,Q_3 = nilai X_i di posisi 25 % ,50 %, 75 % dari semua data $X_1, X_2, \ldots, X_n$ diurut membesar. Q_2 = median.
- **Interquartile range**, IQR = $Q_3 - Q_1$



Statistik Deskriptif: Sample Mean & Variance

```
In [4]: import pandas as pd
        s = pd.Series([3,2,3,2,3,4,4,2,3,4])
        s.describe()

Out[4]: count    10.000000
        mean      3.000000
        std       0.816497
        min       2.000000
        25%       2.250000
        50%       3.000000
        75%       3.750000
        max       4.000000
        dtype: float64
```

Rataan sample (sample mean): $\overline{X} = \frac{\sum_{i=1}^{n} X_i}{n}$

Σx_i=3+2+3+2+3+4+4+2+3+4=30 ⇒ X_bar=30/10=3

Variansi sampel (sample variance):

$$S^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n-1}$$

$$S^2 = \frac{1}{n(n-1)}\left[n\sum_{i=1}^{n} X_i^2 - \left(\sum_{i=1}^{n} X_i\right)^2\right]$$

Σx_i^2=3*4+4*9+3*16=12+36+48=96

S^2=(10*96-(30)^2)/(10*9)=2/3=0.67 ⇒S=0.8165

Quartile

Data terurut: 2,2,2,3,3,3,3,4,4,4
Q2=$(X_5+X_6)/2=3$
Q1=$X_3=2$ (median dari {2,2,2,3,3})
Q3=$X_8=4$ (median dari {3,3,4,4,4})

```
In [19]: df.X.quantile([0,0.25,0.5,0.75,1],interpolation='midpoint')
Out[19]: 0.00    2.0
         0.25    2.5
         0.50    3.0
         0.75    3.5
         1.00    4.0
         Name: X, dtype: float64

In [20]: df.X.quantile([0,0.25,0.5,0.75,1],interpolation='linear')
Out[20]: 0.00    2.00
         0.25    2.25
         0.50    3.00
         0.75    3.75
         1.00    4.00
         Name: X, dtype: float64
```

interpolation : *{'linear', 'lower', 'higher', 'midpoint', 'nearest'}*

This optional parameter specifies the interpolation method to use when the desired quantile lies between two data points `i < j`:

- linear: `i + (j - i) * fraction`, where `fraction` is the fractional part of the index surrounded by `i` and `j`.
- lower: `i`.
- higher: `j`.
- nearest: `i` or `j`, whichever is nearest.
- midpoint: `(i + j) / 2`.

Quartile

Data terurut: 2,2,2,3,3,3,3,4,4,4
Q2=(X_5+X_6)/2=3
Q1=X_3=2 (median dari {2,2,2,3,3})
Q3=X_8=4 (median dari {3,3,4,4,4})

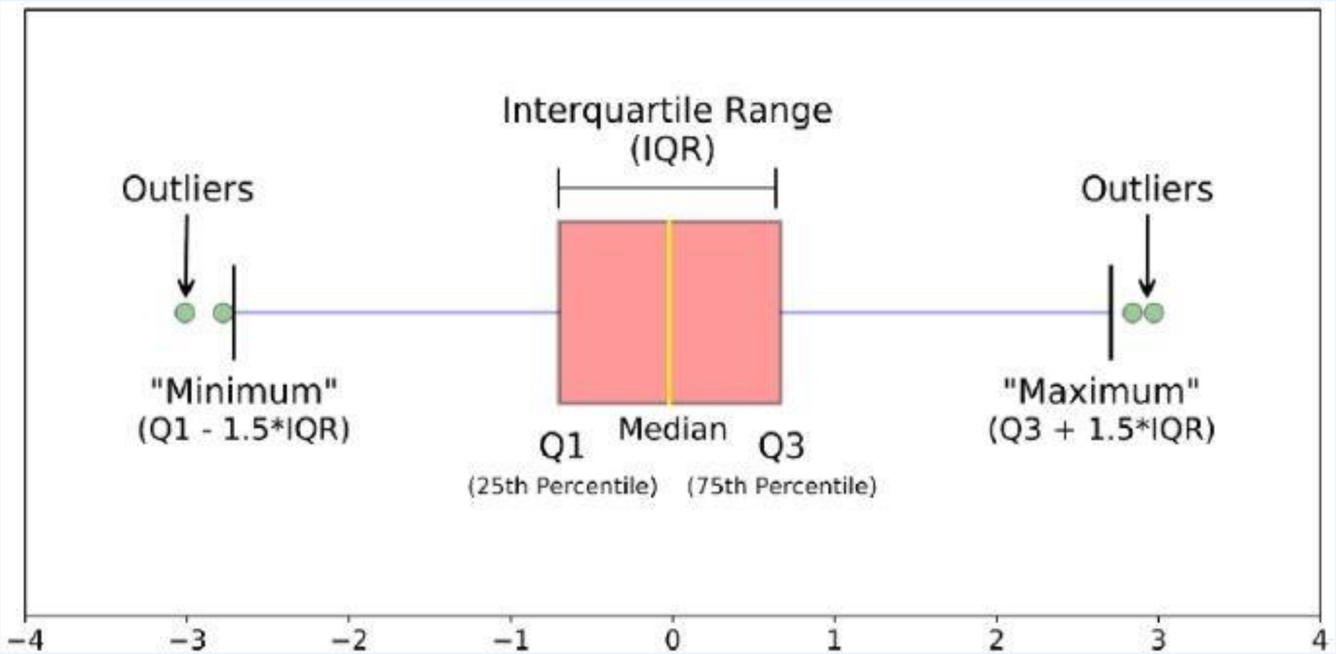
```
In [21]: df.X.quantile([0,0.25,0.5,0.75,1],interpolation='lower')
Out[21]: 0.00    2
         0.25    2
         0.50    3
         0.75    3
         1.00    4
         Name: X, dtype: int64

In [22]: df.X.quantile([0,0.25,0.5,0.75,1],interpolation='higher')
Out[22]: 0.00    2
         0.25    3
         0.50    3
         0.75    4
         1.00    4
         Name: X, dtype: int64
```

interpolation : *{'linear', 'lower', 'higher', 'midpoint', 'nearest'}*

This optional parameter specifies the interpolation method to use when the desired quantile lies between two data points `i < j`:

- linear: `i + (j - i) * fraction`, where `fraction` is the fractional part of the index surrounded by `i` and `j`.
- lower: `i`.
- higher: `j`.
- nearest: `i` or `j`, whichever is nearest.
- midpoint: `(i + j) / 2`.

BoxPlot



- A boxplot is a graph that gives you a good indication of how the values in the data are spread out.
- Boxplots are a standardized way of displaying the distribution of data based on a five number summary (“minimum”, first quartile (Q1), median, third quartile (Q3), and “maximum”).

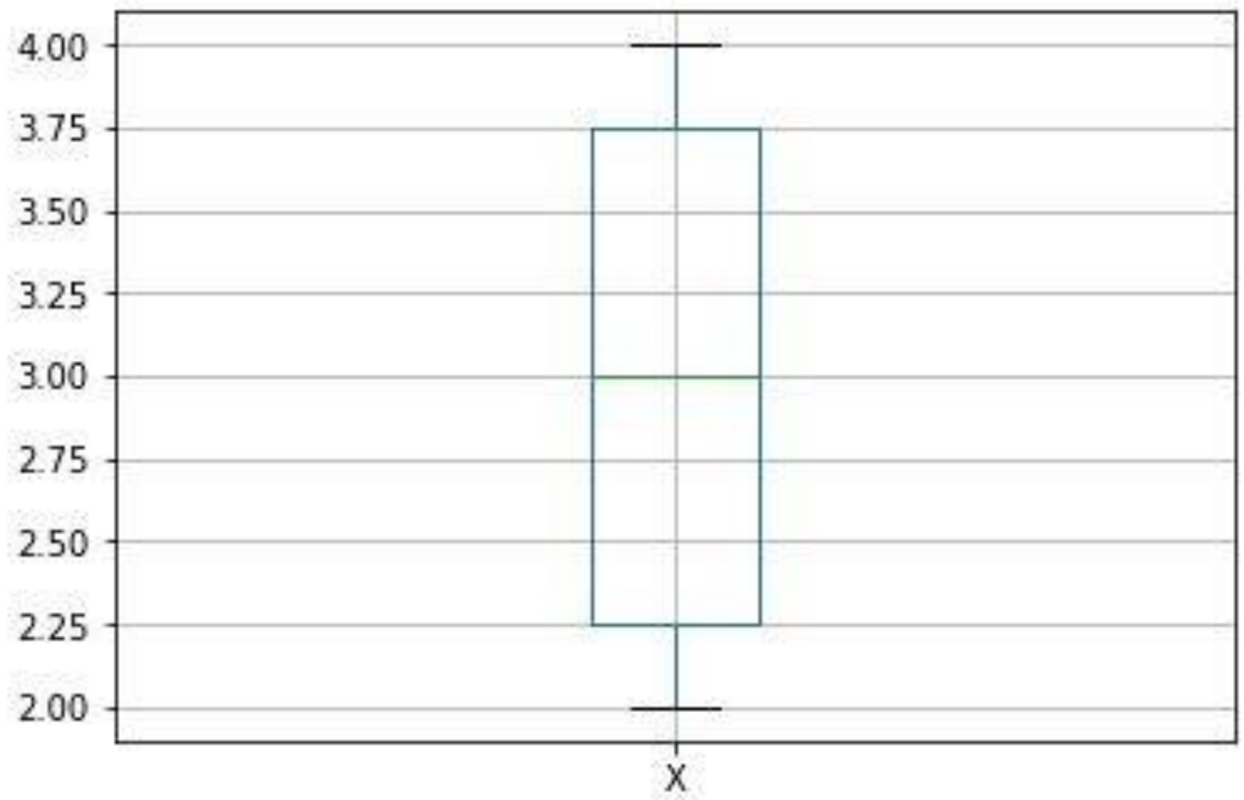
https://towardsdatascience.com/understanding-boxplots-5e2df7bcbd51

Quartile & BoxPlot

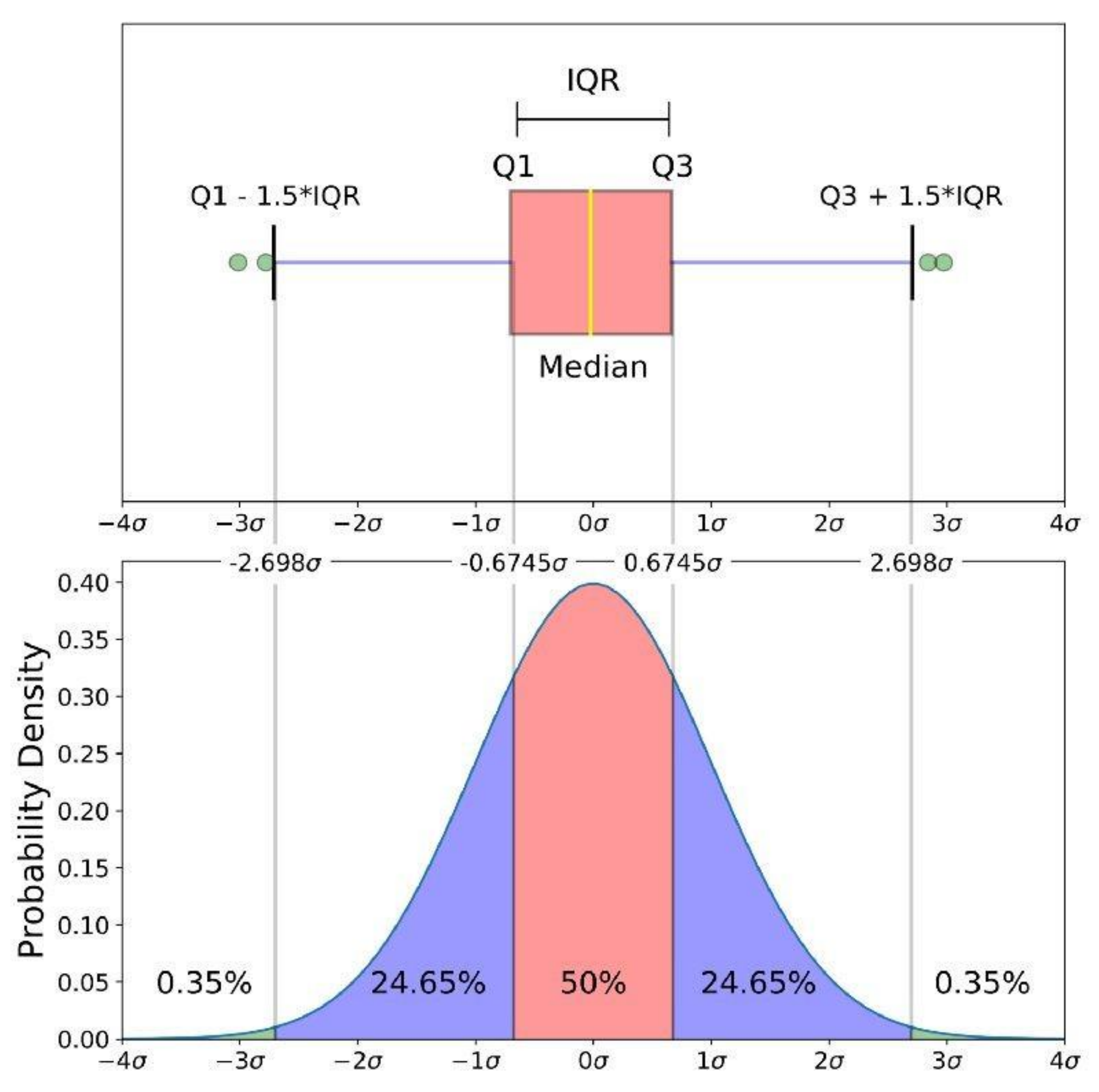
Data terurut: 2,2,2,3,3,3,3,4,4,4

```
In [12]: df.X.mean()
Out[12]: 3.0

In [18]: df.X.median()
Out[18]: 3.0

In [13]: df.X.std()
Out[13]: 0.816496580927726

In [17]: df.X.quantile([0,0.25,0.5,0.75,1])
Out[17]: 0.00    2.00
         0.25    2.25
         0.50    3.00
         0.75    3.75
         1.00    4.00
         Name: X, dtype: float64
```

```
%matplotlib inline
df = pd.DataFrame(data=s, columns=['X'])
boxplot = df.boxplot(column=['X'])
```



BoxPlot untuk Distribusi Normal



https://towardsdatascience.com/understanding-boxplots-5e2df7bcbd51

Contoh Analisis Data dengan BoxPlot

Diberikan hasil pengukuran isi (dalam liter) dua sampel jus jeruk perusahaan A dan B:

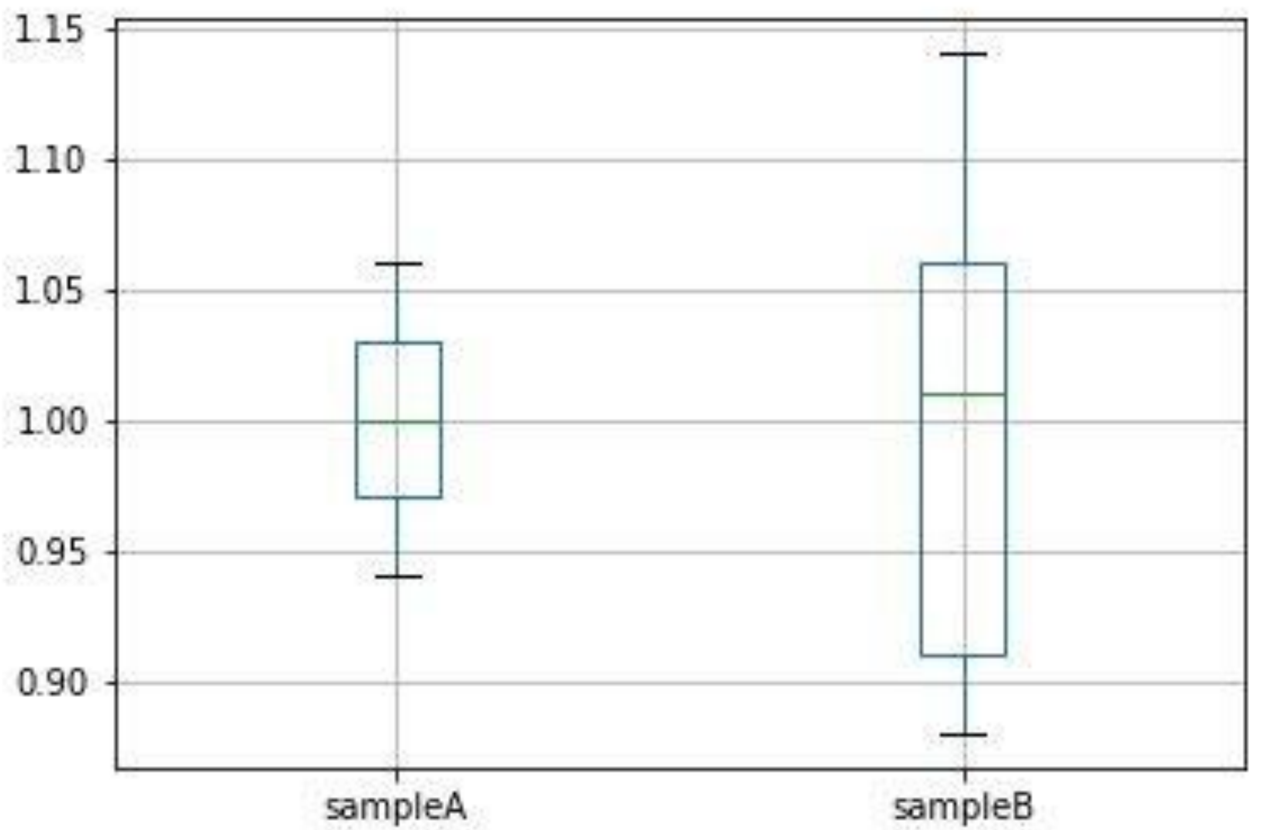
Sample A	0.97	1.00	0.94	1.03	1.06
Sample B	1.06	1.01	0.88	0.91	1.14

Sample mean A = sample mean B = 1.00 liter

Statistik Deskriptif dan BoxPlot

	sampleA	sampleB
count	5.000000	5.000000
mean	1.000000	1.000000
std	0.047434	0.107005
min	0.940000	0.880000
25%	0.970000	0.910000
50%	1.000000	1.010000
75%	1.030000	1.060000
max	1.060000	1.140000

```
%matplotlib inline
boxplot = df.boxplot(column=['sampleA','sampleB'])
```



Skewness

- A distribution is said to be symmetric if it can be folded along a vertical axis so that the two sides coincide. A distribution that lacks symmetry with respect to a vertical axis is said to be skewed. (Walpole)

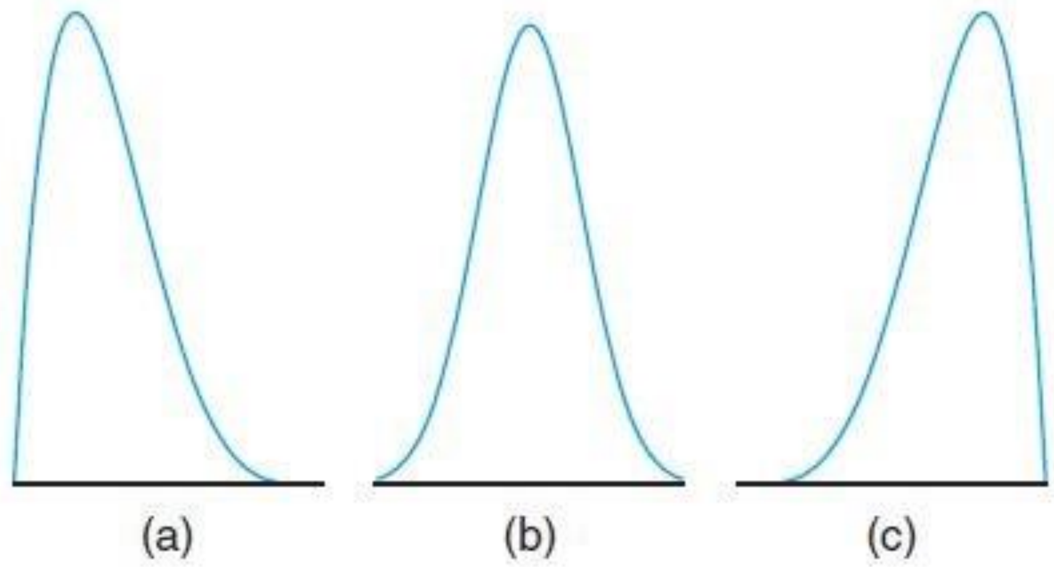
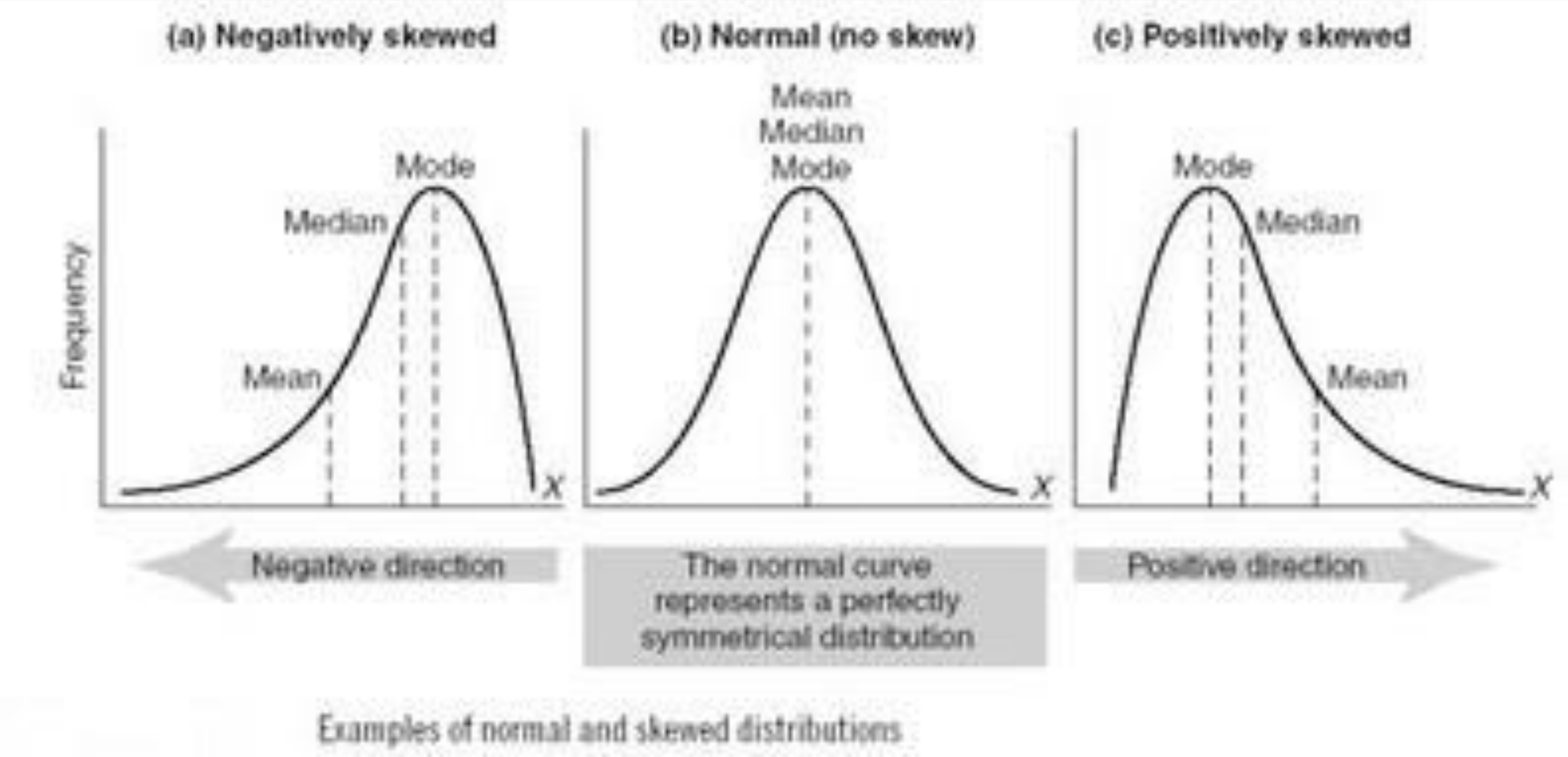


Figure 1.8: Skewness of data.

$$Sk = \frac{\{\sum_{i=1}^{n}(X_i - \bar{X})^3\}/n}{(S^2)^{3/2}}$$

The distribution illustrated in Figure 1.8(a) is said to be skewed to the right since it has a long right tail and a much shorter left tail. In Figure 1.8(b) we see that the distribution is symmetric, while in Figure 1.8(c) it is skewed to the left.

Skewness: Positive / Negative Direction



https://blog.usejournal.com/descriptive-statistics-with-python-6c7acb1d3671

Positively skewed: Most frequent values are low and tail is towards high values.
Negatively skewed: Most frequent values are high and tail is towards low values.

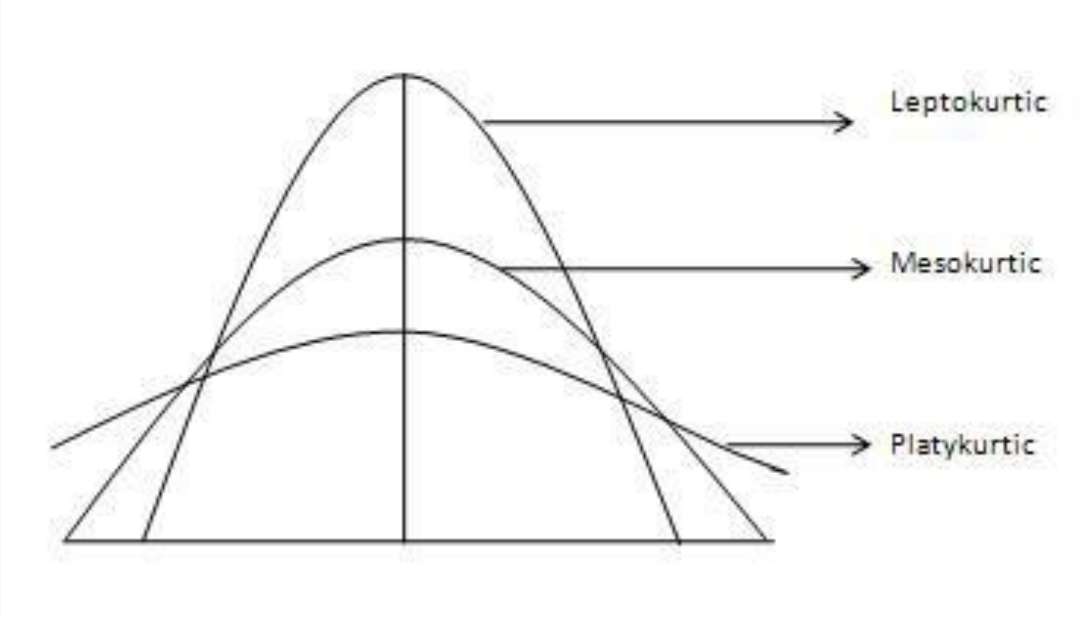
Kurtosis

- **Kurtosis** (keruncingan) = ukuran lancip dari fdp

$$Kur = \frac{(\frac{1}{n}\sum(X_i - \bar{X})^4)}{S^4}$$

Data berdistribusi normal,

Kur = 3



Skewness & Kurtosis

Data terurut: 2,2,2,3,3,3,3,4,4,4
X_bar=3 ; $S^2=0.67$

Skewness=$(3*(2-3)^3+4*(3-3)^3+3*(4-3)^3)/(0.67)^{3/2}=0$

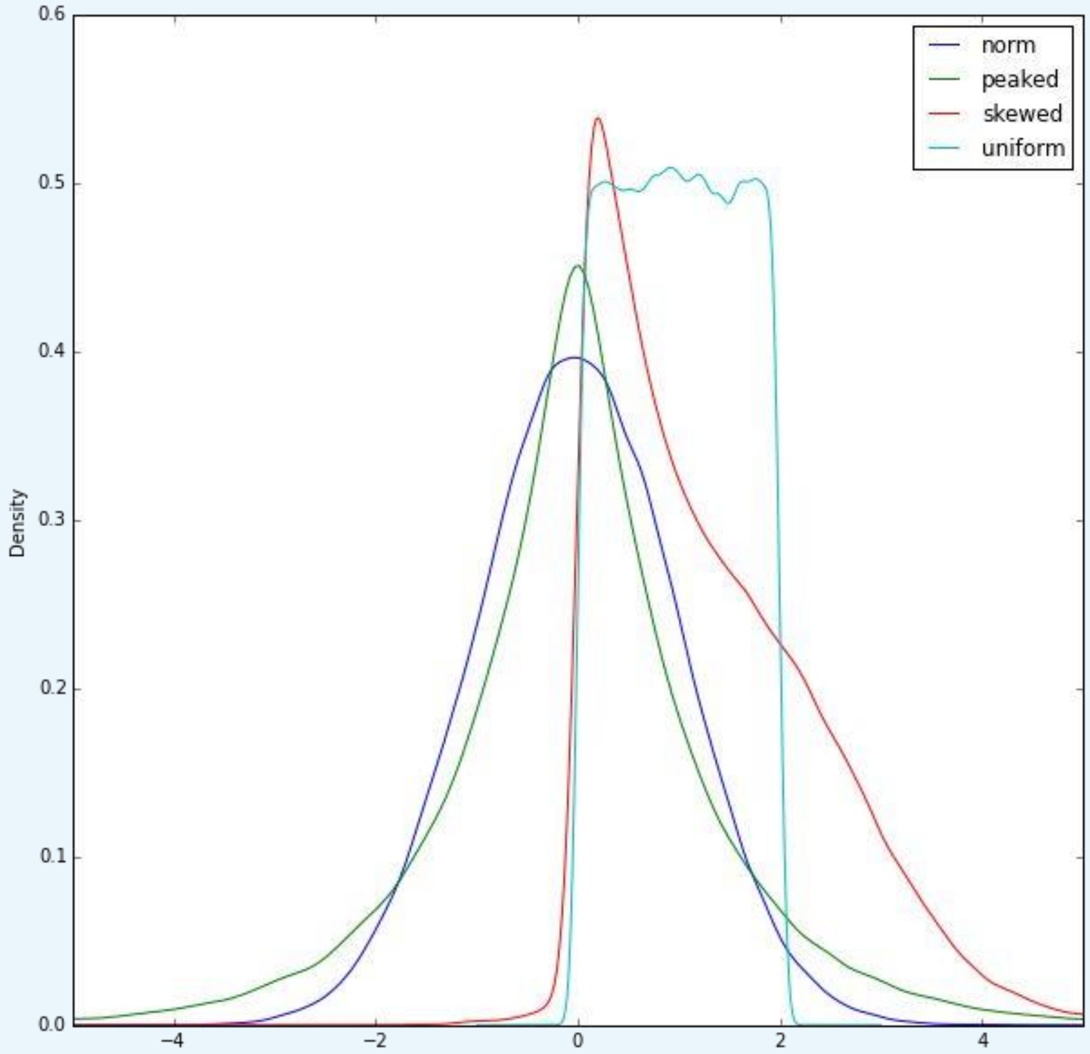
$$Sk = \frac{\{\sum_{i=1}^{n}(X_i - \bar{X})^3\}/n}{(S^2)^{3/2}}$$

```
In [26]: df.X.skew()
Out[26]: 0.0

In [27]: df.X.kurtosis()
Out[27]: -1.3928571428571428
```

Kurtosis=$(3*(2-3)^4+4*(3-3)^4+3*(4-3)^4)/(10*(0.67)^2)=1.3366$

$$Kur = \frac{(\frac{1}{n}\Sigma(X_i - \bar{X})^4)}{S^4}$$

Skewness & Kurtosis: Contoh

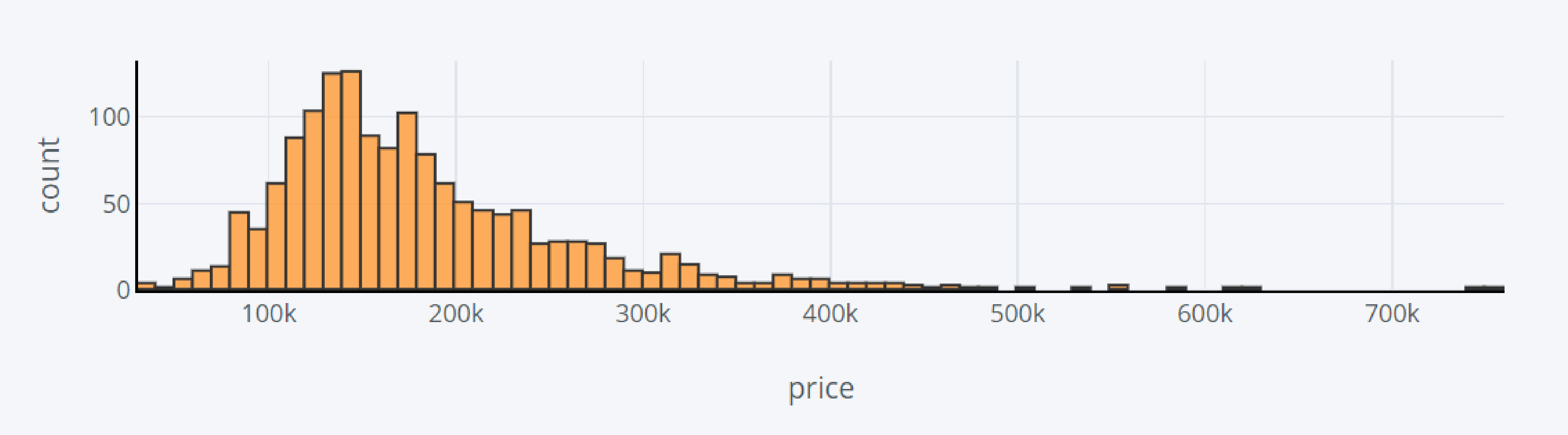



```
In [22]: data_df.skew()

Out[22]:
norm       0.005802
peaked    -0.007226
skewed     0.982716
uniform    0.001460
dtype: float64
```

```
In [23]: data_df.kurt()

Out[23]:
norm      -0.014785
peaked     2.958413
skewed     1.086500
uniform   -1.196268
dtype: float64
```

http://hamelg.blogspot.com/2015/11/python-for-data-analysis-part-21.html

Histogram Plot



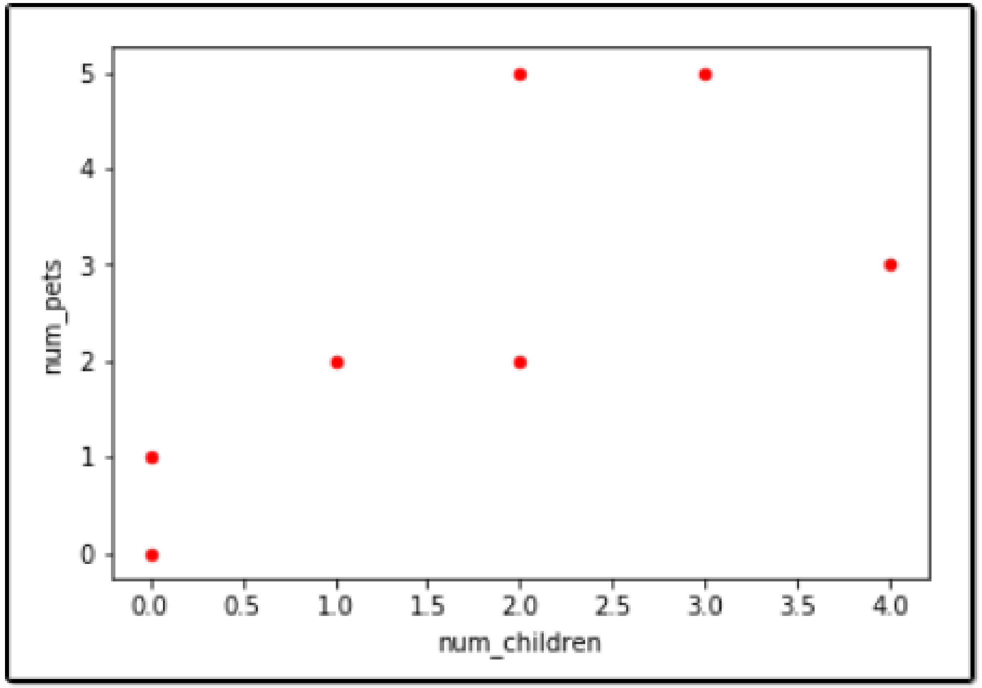

```
df['SalePrice'].iplot(
    kind='hist',
    bins=100,
    xTitle='price',
    linecolor='black',
    yTitle='count',
    title='Histogram of Sale Price')
```

https://towardsdatascience.com/practical-statistics-visualization-with-python-plotly-770e96e35067

Scatter Plot

	name	age	gender	state	num_children	num_pets
0	john	23	M	california	2	5
1	mary	78	F	dc	0	1
2	peter	22	M	california	0	0
3	jeff	19	M	dc	3	5
4	bill	45	M	california	2	2
5	lisa	33	F	texas	1	2
6	jose	20	M	texas	4	3

Source dataframe



Looks like we have a trend

```
import matplotlib.pyplot as plt
import pandas as pd

# a scatter plot comparing num_children and num_pets
df.plot(kind='scatter',x='num_children',y='num_pets',color='red')
plt.show()
```

https://queirozf.com/entries/pandas-dataframe-plot-examples-with-matplotlib-pyplot

Terima Kasih