

# Modul Praktikum

## Pembelajaran Mesin (*Machine Learning*)



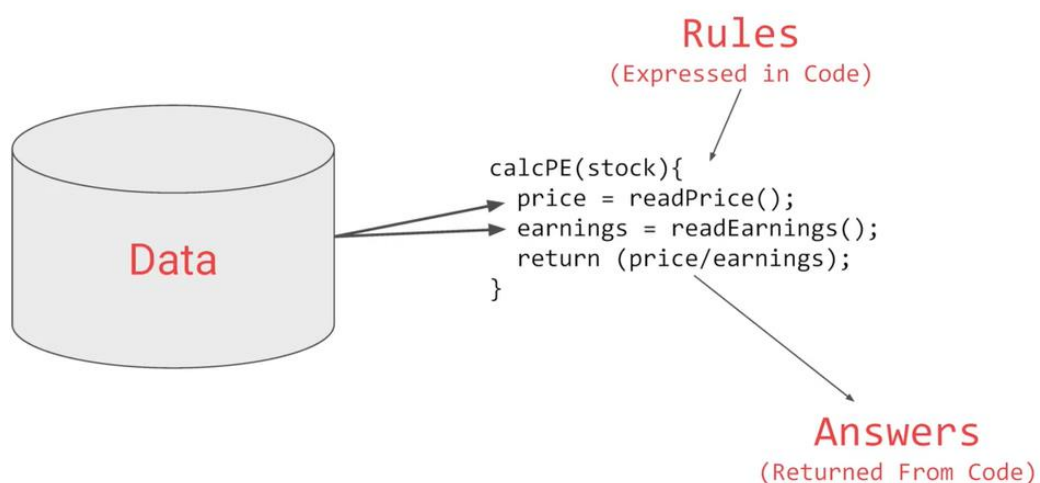
### Pertemuan 1

## Introduction to Machine Learning and Google Colab

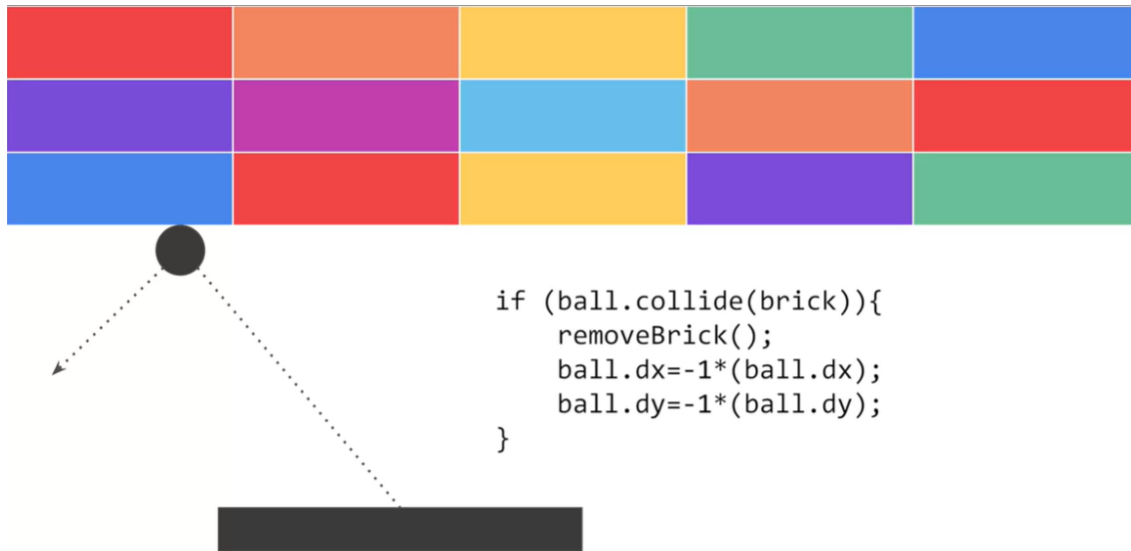
# Introduction to Machine Learning and Google Colab

## Introduction to Machine Learning

Apa itu pembelajaran mesin (*machine learning*)? *Programmer* biasanya membuat aplikasi dengan memecah *requirements* menjadi beberapa sub-masalah yang kemudian kita dapat menulis kode untuk memenuhi *requirements*. Sebagai contoh, misalnya kita diminta untuk membuat sebuah aplikasi yang dapat mengetahui analitik stok, mungkin rumus harga dibagi dengan penghasilan dapat menjadi solusi untuk masalah ini. Dari ide tersebut, biasanya kita dapat menulis kode untuk mendapatkan nilai harga dan penghasilan dari sumber data (misalnya database), melakukan perhitungan, hingga kemudian kita mendapat hasil yang kita inginkan.



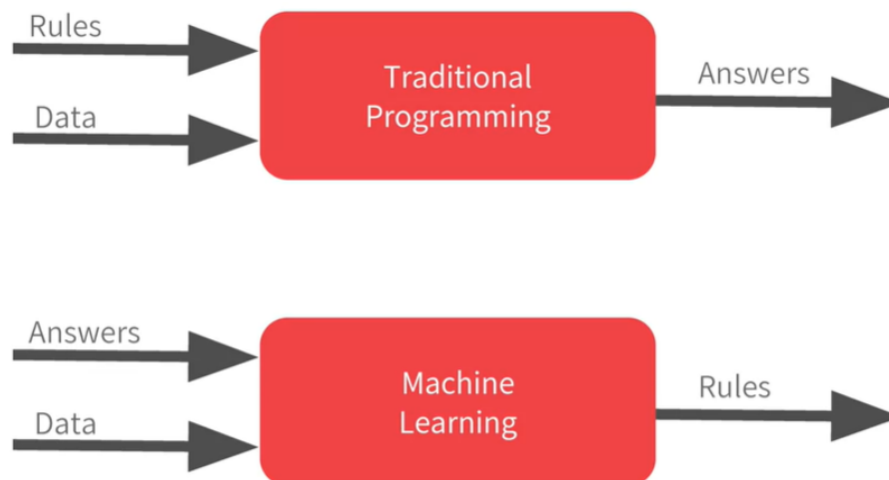
Contoh lain, jika kita menulis kode untuk membuat game, kita biasanya menentukan sendiri aturan bermain yang akan digunakan pada game tersebut. Misalnya, jika bolanya mengenai batu, maka batu tersebut akan hilang/ lenyap dan bolanya akan memantul. Tetapi, jika bolanya jatuh ke bagian bawah layer, maka pemain akan kalah.



Kita dapat merepresentasikan ide tersebut dalam diagram di bawah ini :



Diagram di atas menggambarkan cara kerja dari *traditional programming*, yang mana *programmer* menulis aturan dalam bahasa pemrograman dan mengambil data dari sebuah sumber data (misalnya database), kemudian komputer akan memberikan jawaban seperti yang diinginkan. Dengan bahasa sederhananya aturan dan data masuk, kemudian jawaban akan keluar. *Machine learning* mengubah alur tersebut, yang mana jawaban dan data akan masuk menjadi *input*, kemudian kita akan mendapatkan aturan.



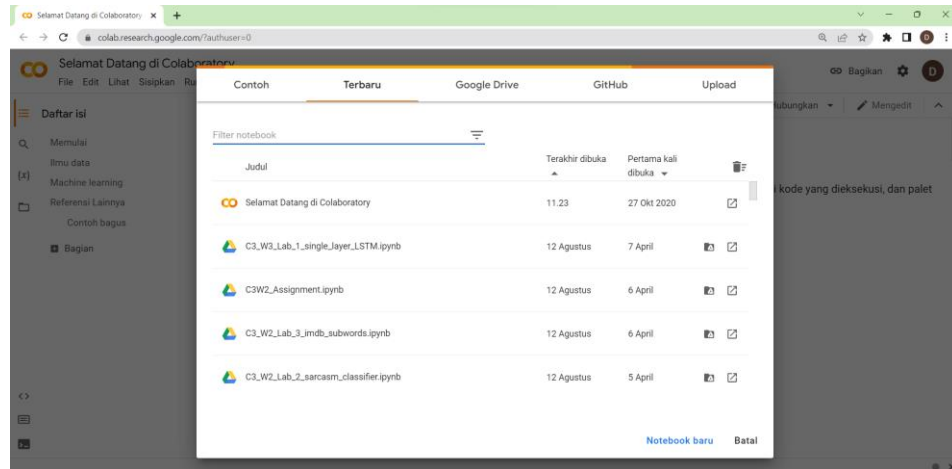
Jadi, alih-alih *programmer* mencari tahu mengenai aturannya, dengan data-data beserta jawaban yang masuk, komputer akan mempelajari data-data tersebut, kemudian mengenali pola datanya, kemudian komputerlah yang akan mencari tahu sendiri aturannya, sampai akhirnya komputer ini memiliki kecerdasan seperti manusia. Jadi, pembelajaran mesin adalah bagian dari kecerdasan buatan (*Artificial Intelligence*) yang berfokus pada tujuan tertentu, yaitu: membuat komputer untuk dapat melakukan tugas tanpa perlu pemrograman eksplisit.

## Introduction to Google Colab

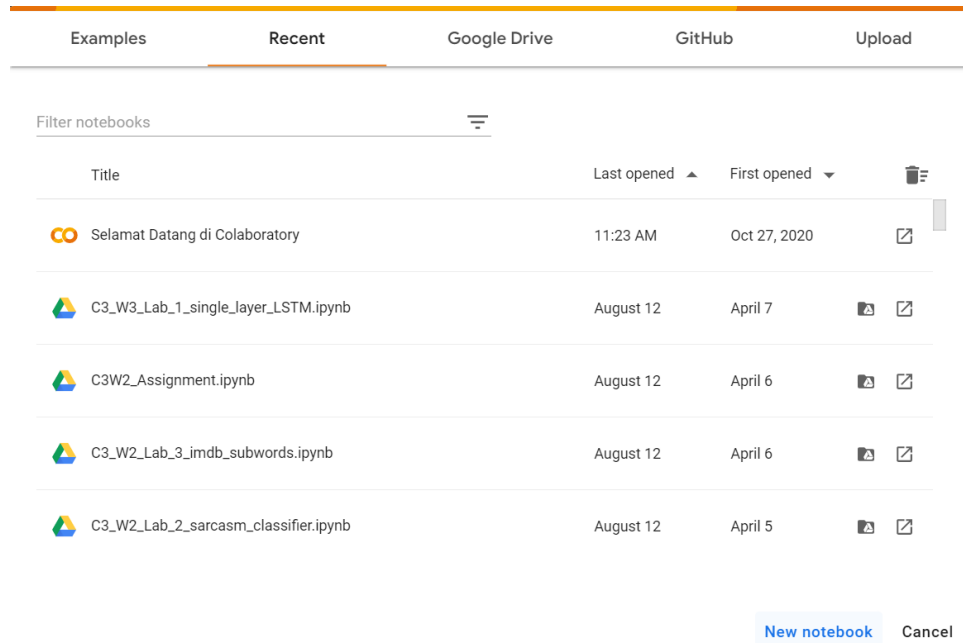
Google colab adalah sebuah dokumen yang executable, yang mana dapat digunakan untuk menulis, menjalankan (run), dan berbagi kode dalam Google Drive. Google colab dapat disebut sebagai Jupyter Notebook yang disimpan di dalam Google Drive.

Untuk memulai membuat dokumen baru menggunakan Google Colab, buka Google Colab pada <https://colab.research.google.com/>. Ada 2 cara untuk membuat dokumen atau notebook baru menggunakan Google Colab, yaitu :

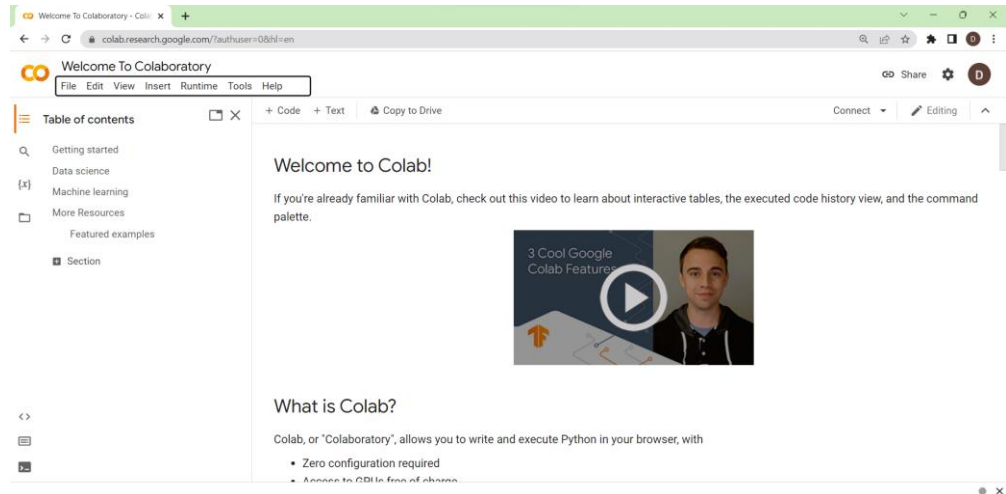
- 1) Setelah mengakses <https://colab.research.google.com/>, maka akan muncul tampilan box seperti di bawah ini



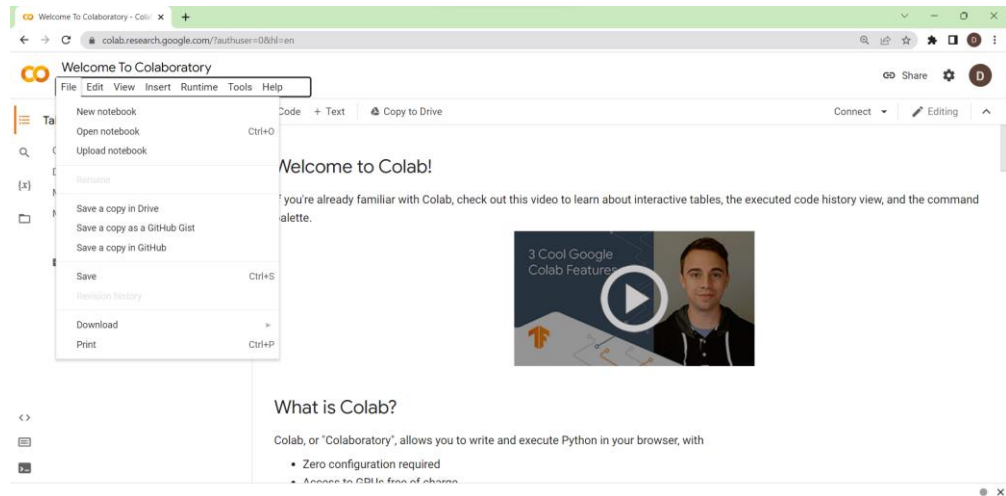
Untuk membuat dokumen baru, klik tombol “New notebook” pada box di atas.



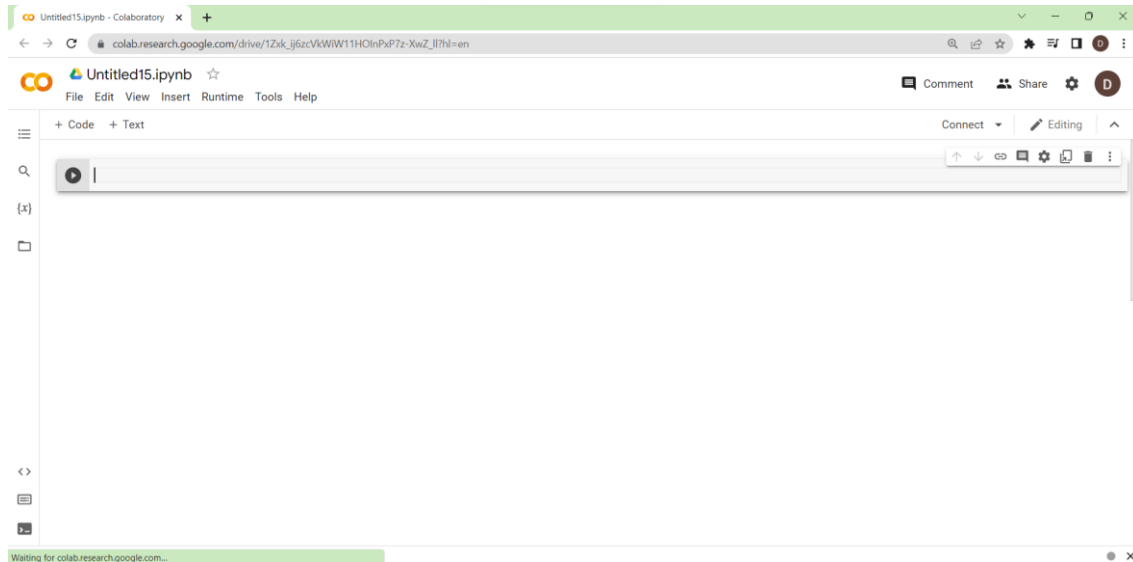
2) Atau, saat membuka Google Colab dan diarahkan ke halaman seperti di bawah ini,



Pilih “File” pada pojok kiri atas, kemudian saat muncul dropdown options, pilih “New notebook”.



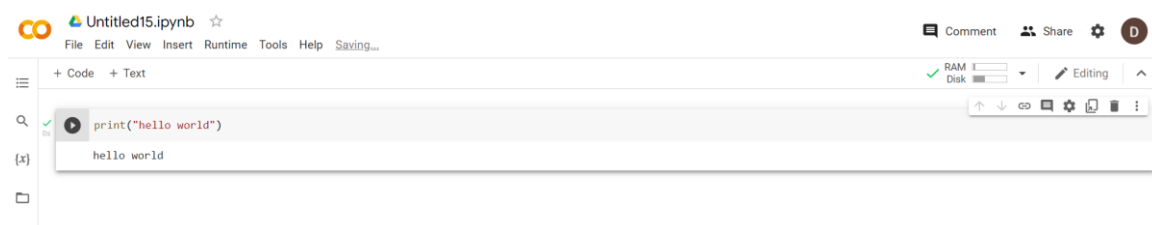
Jika notebook berhasil di buat, maka akan muncul tampilan seperti di bawah ini :



Satu dokumen notebook dapat terdiri atas satu atau lebih sel, yang mana masing-masing sel dapat berisi kode, teks, gambar, dan lain-lain. Google Colab mengkoneksikan notebook yang ditulis pengguna ke cloud-based runtime, artinya pengguna dapat menjalankan kode Python tanpa pengaturan khusus pada mesin/ perangkat yang digunakan pengguna. Google colab menawarkan pengalaman menulis kode secara interaktif, yang mana pengguna dapat menggunakan semua fungsionalitas yang ditawarkan Python. Sebagai contoh :

```
print("hello world")
```

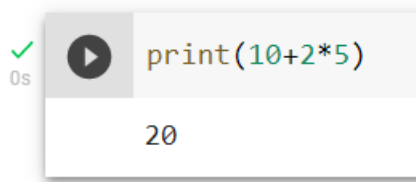
Untuk menjalankan kode di atas, klik tombol ▶



Contoh lain :

```
print(10+2*5)
```

Output :



## Data Yang Digunakan

Dalam kegiatan praktikum pembelajaran mesin ini, dataset yang akan digunakan adalah dataset klasifikasi Bunga Iris. Dataset ini berisi 150 pengamatan bunga Iris. Terdapat empat kolom pengukuran bunga dalam centimeter. Kolom kelima adalah spesies bunga yang diamati. Dataset ini merupakan dataset yang paling umum digunakan, sehingga sering disebut sebagai dataset “hello word” dalam *machine learning*. Informasi detail mengenai dataset tersebut dapat diakses melalui [https://en.wikipedia.org/wiki/Iris\\_flower\\_data\\_set](https://en.wikipedia.org/wiki/Iris_flower_data_set).

## Memuat Dataset

Untuk memuat dataset dapat langsung diambil dari alamat repository UCI Machine Learning. Berikut ini adalah script untuk memuat dataset:

```
>>>url = "http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"

>>>names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width', 'class']

>>>dataset = pandas.read_csv(url, names=names)
```

## Melakukan Summary Dataset

### I. Menentukan Dimensi dari Dataset

Dimensi dari dataset merupakan gambaran singkat mengenai banyaknya jumlah baris yang menunjukkan banyaknya sampel data dan jumlah kolom yang menunjukkan atribut data dari dataset terkait. Pada contoh dataset untuk task klasifikasi, kolom terakhir pada umumnya



menunjukkan kelas datanya. Berikut ini adalah script untuk menentukan dimensi dari dataset:

```
>>>print(dataset.shape)  
(150,5)
```

Hal tersebut berarti dataset tersebut terdiri dari 150 sampel data dan 5 atribut data.

## II. Melihat Isi Dataset

Berikut perintah untuk melihat 20 baris data pertama:

```
>>>print(dataset.head(20))
```

Output:

	sepal-length	sepal-width	petal-length	petal-width	class
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
5	5.4	3.9	1.7	0.4	Iris-setosa
6	4.6	3.4	1.4	0.3	Iris-setosa
7	5.0	3.4	1.5	0.2	Iris-setosa
8	4.4	2.9	1.4	0.2	Iris-setosa
9	4.9	3.1	1.5	0.1	Iris-setosa
10	5.4	3.7	1.5	0.2	Iris-setosa
11	4.8	3.4	1.6	0.2	Iris-setosa
12	4.8	3.0	1.4	0.1	Iris-setosa
13	4.3	3.0	1.1	0.1	Iris-setosa
14	5.8	4.0	1.2	0.2	Iris-setosa
15	5.7	4.4	1.5	0.4	Iris-setosa
16	5.4	3.9	1.3	0.4	Iris-setosa
17	5.1	3.5	1.4	0.3	Iris-setosa
18	5.7	3.8	1.7	0.3	Iris-setosa
19	5.1	3.8	1.5	0.3	Iris-setosa

## III. Distribusi Kelas Data

```
>>>print(dataset.groupby('class').size())
```

Output:

```
class
Iris-setosa      50
Iris-versicolor  50
Iris-virginica   50
dtype: int64
```

#### IV. Ringkasan Statistik

Berikut adalah perintah untuk mendeskripsikan statistik data meliputi count, mean, standard deviasi, nilai min, nilai max, dan quartile:

```
>>>print (dataset.describe())
```

Output:

	sepal-length	sepal-width	petal-length	petal-width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.054000	3.758667	1.198667
std	0.828066	0.433594	1.764420	0.763161
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

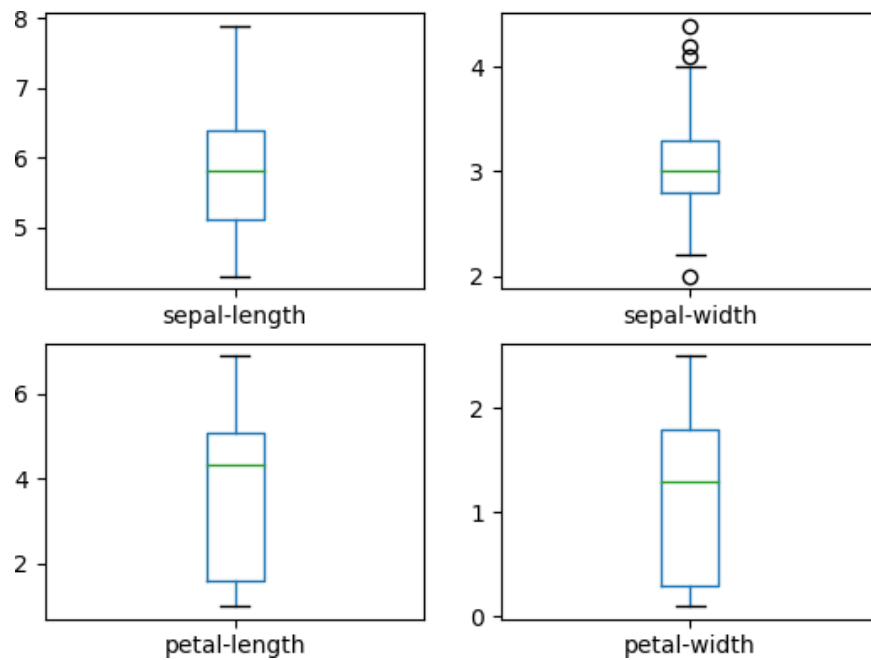
#### V. Visualisasi Data Menggunakan Plot Univariat

Plot univariat adalah plot dari masing-masing variabel individu.

Mengingat bahwa variabel inputnya numerik, kita bisa membuat jenis plot box.

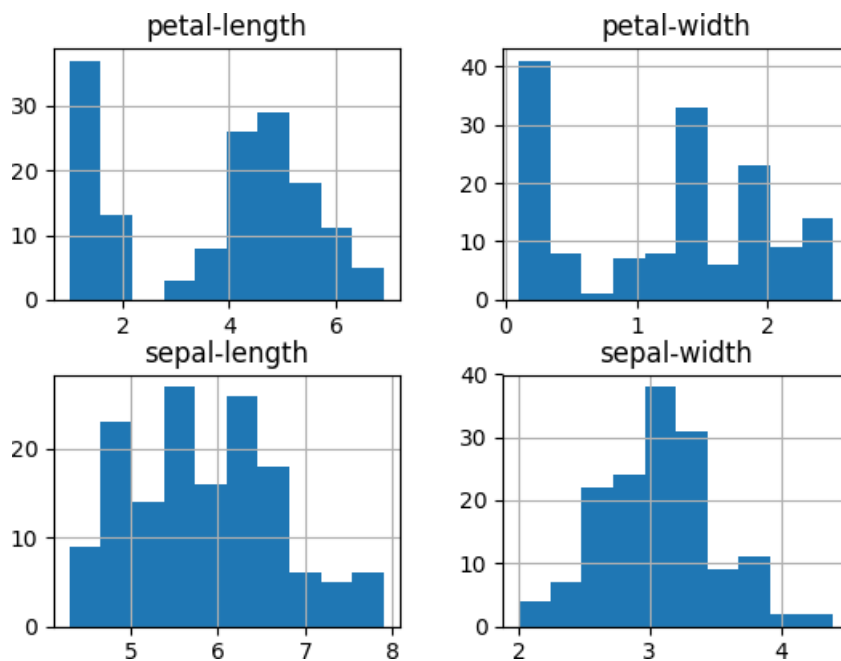
```
>>>import matplotlib.pyplot as plt
>>>dataset.plot(kind='box', subplots=True,
layout=(2,2), sharex=False, sharey=False)
>>>plt.show()
```

Output:



Selanjutnya juga bisa membuat histogram masing-masing variabel input untuk mendapatkan ide tentang distribusi.

```
>>>dataset.hist()
>>>plt.show()
```

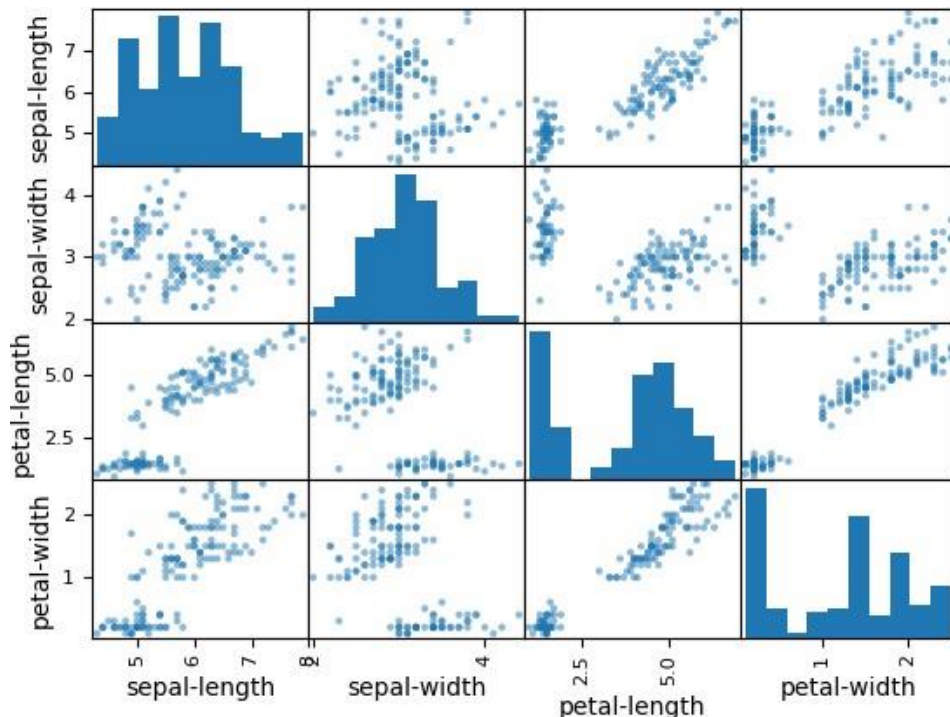


## VI. Visualisasi Data Menggunakan Plot Multivariat

Selanjutnya kita bisa melihat interaksi antar variabel. Pertama, kita lihat scatterplots dari semua pasang atribut. Hal ini dapat membantu melihat hubungan terstruktur antara variabel input.

```
>>>import pandas
>>>from pandas.plotting import scatter_matrix
>>>scatter_matrix(dataset)
>>>plt.show()
```

Output:



### Latihan

Cari dataset dataset lain dari repository UCI Machine Learning dan lakukan keenam langkah untuk menampilkan *summary dataset*!

## References

1. <http://www.belajarpython.com/>

2. <https://www.quora.com/topic/Python-programming-language-1>
3. [https://medium.com/@haydar\\_ai/learning-data-science-day-1-environment-and-python-9076c82f0298#.h9sjjomkw](https://medium.com/@haydar_ai/learning-data-science-day-1-environment-and-python-9076c82f0298#.h9sjjomkw)