

Machine Learning & Pattern Recognition

Ali Akbar Septiandri

Universitas Al-Azhar Indonesia

aliakbars@live.com

May 8, 2017

1 Machine Learning

- Pendahuluan
- scikit-learn

2 Neural Networks

- Deep Learning
- Keras

- ① VanderPlas, J. (2016). Python Data Science Handbook. O'Reilly Media Inc. (Chapter 5)
<https://github.com/jakevdp/PythonDataScienceHandbook>
- ② Dokumentasi scikit-learn:
<http://scikit-learn.org/stable/documentation.html>
- ③ Dokumentasi Keras: <https://keras.io/>
- ④ Getting Started with TensorFlow
https://www.tensorflow.org/get_started/

Machine Learning

Apa itu *machine learning*?

*“Fundamentally, machine learning involves
building mathematical models to help understand data.”*
[VanderPlas, 2016]

Tugas dalam Machine Learning

- Memprediksi harga saham
- Membedakan gambar kucing dan anjing
- Mengidentifikasi spam
- Menemukan kelompok-kelompok pertemanan
- Meniru gaya lukisan untuk diaplikasikan pada foto
- Membuat agen untuk memenangkan permainan go
- dsb.

- Supervised learning
 - ▶ Memprediksi harga saham
 - ▶ Membedakan gambar kucing dan anjing
 - ▶ Mengidentifikasi spam
- Unsupervised learning
 - ▶ Menemukan kelompok-kelompok pertemanan
 - ▶ Meniru gaya lukisan untuk diaplikasikan pada foto
- Reinforcement learning
 - ▶ Membuat agen untuk memenangkan permainan go

- Untuk semua kasus tersebut, representasi termudah untuk diolah komputer adalah dengan tabel

- Untuk semua kasus tersebut, representasi termudah untuk diolah komputer adalah dengan tabel
- Sangat erat kaitannya dengan pengolahan data dengan NumPy dan Pandas

- Untuk semua kasus tersebut, representasi termudah untuk diolah komputer adalah dengan tabel
- Sangat erat kaitannya dengan pengolahan data dengan NumPy dan Pandas
- **baris = objek, kolom = atribut**

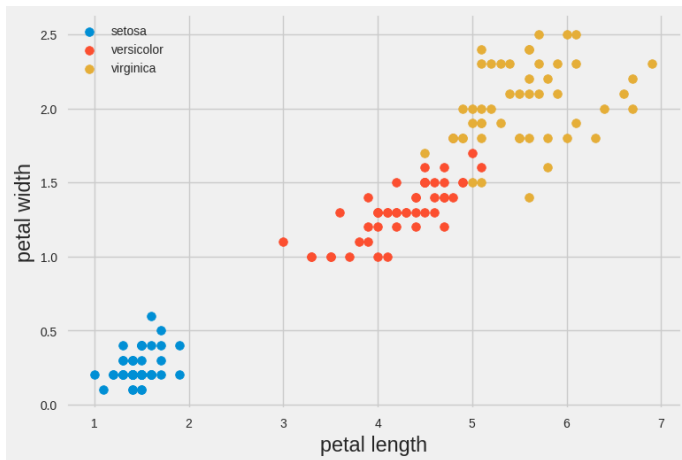
Deskripsi Dataset

- Iris dataset
- Pembuat: R.A. Fisher (1936)
- <http://archive.ics.uci.edu/ml/>
- 4 atribut: sepal length, sepal width, petal length, petal width
- 3 label: Iris Setosa, Iris Versicolour, Iris Virginica

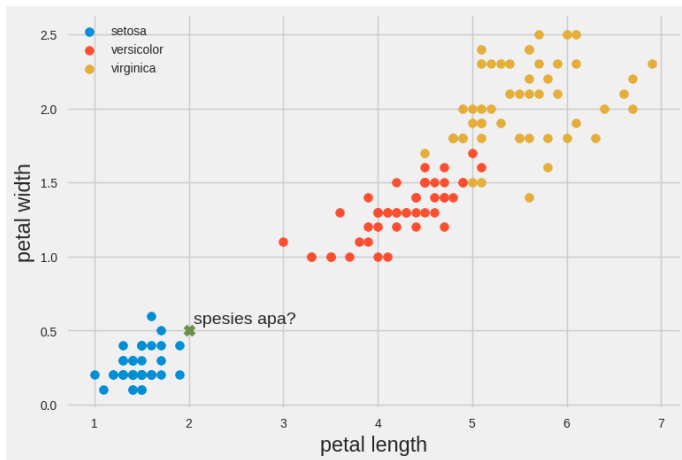


Gambar: Tanaman Iris

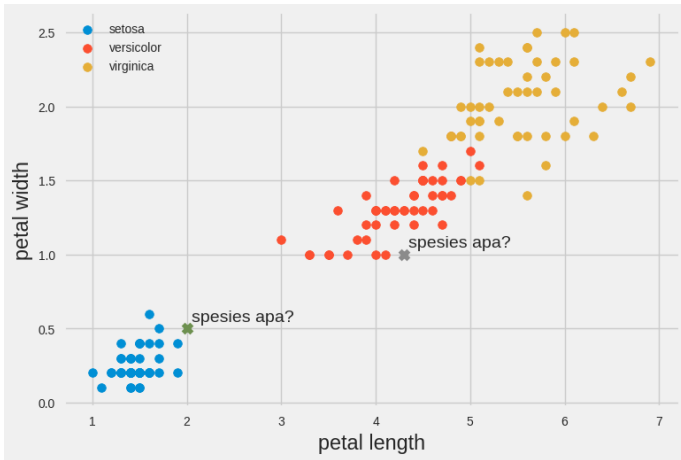
Iris Dataset



Data Baru



Data Baru



Nearest Neighbour

- Mencari referensi dari tetangga terdekat

Nearest Neighbour

- Mencari referensi dari tetangga terdekat
- Apa definisi “terdekat”?

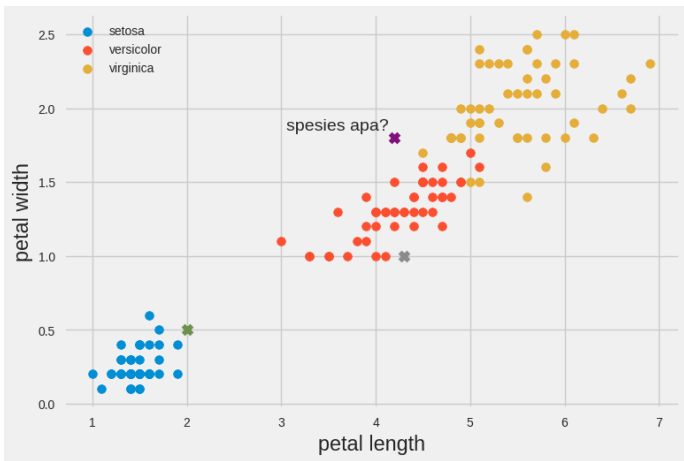
Nearest Neighbour

- Mencari referensi dari tetangga terdekat
- Apa definisi “terdekat”?
- Metode umum: **Euclidean distance**

Euclidean Distance

$$d([x_1, x_2, \dots, x_d], [y_1, y_2, \dots, y_d]) = \sqrt{\sum_{i=1}^d (x_i - y_i)^2}$$

Masalah



Gambar: Seberapa yakin kita dengan referensi terdekat?

k-Nearest Neighbours

- Mencari referensi dari **beberapa** (k) tetangga terdekat
- Melihat **label mayoritas** dari tetangga terdekat
- Perhatikan bahwa harus dihitung jaraknya dengan **semua** data yang ada
- Kompleksitas: $O(nd)$
- *Tidak mungkin kita hitung sendiri!*

Example (3-NN)

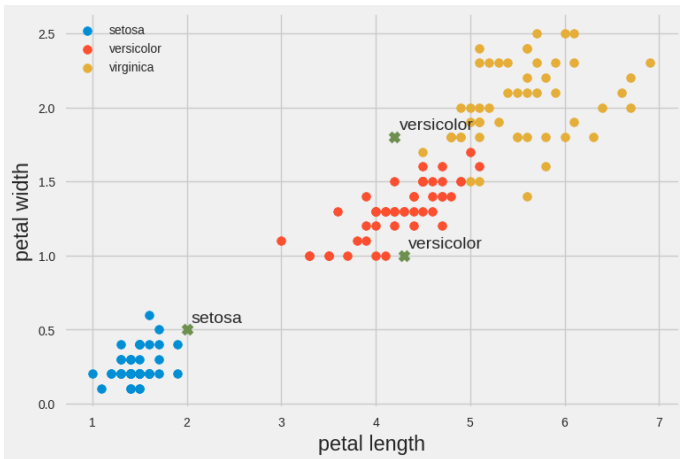
```
from sklearn.neighbors import KNeighborsClassifier
import numpy as np
import pandas as pd
import seaborn as sns

df = sns.load_dataset('iris')

X_train = df[['petal_length', 'petal_width']]
y_train = df['species']
X_test = np.array([[2, .5], [4.3, 1.], [4.2, 1.8]])

clf = KNeighborsClassifier(3)
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)
```

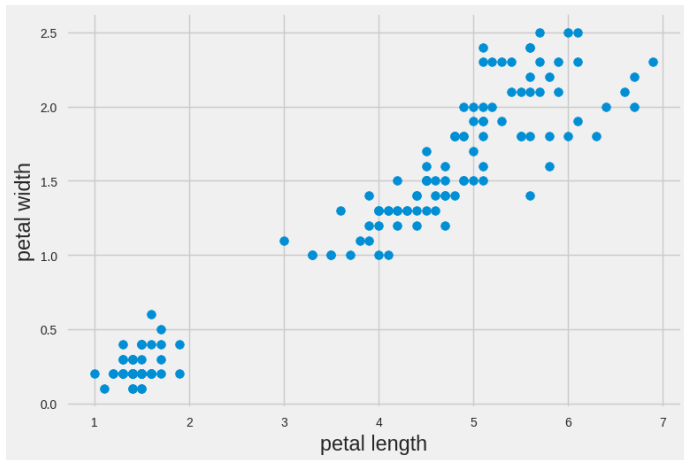
Prediksi



Bagaimana kalau kita hanya ingin melihat kelompok datanya saja?

- Mencari kelompok dalam data
- “Tetangga dalam satu kompleks, tidak peduli kelasnya”
- *Unsupervised learning*

Data Iris



k-Means Clustering

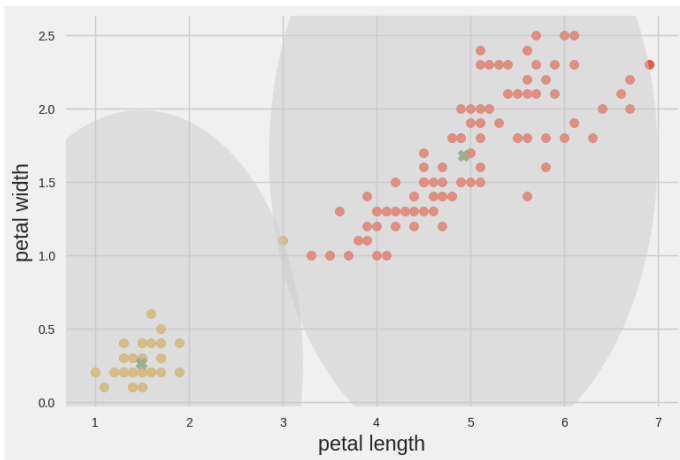
- k adalah **jumlah klaster**
- Siapa saja tetangga terdekat?
- Seperti k-NN, umum dihitung dengan Euclidean distance

Example (2-Means)

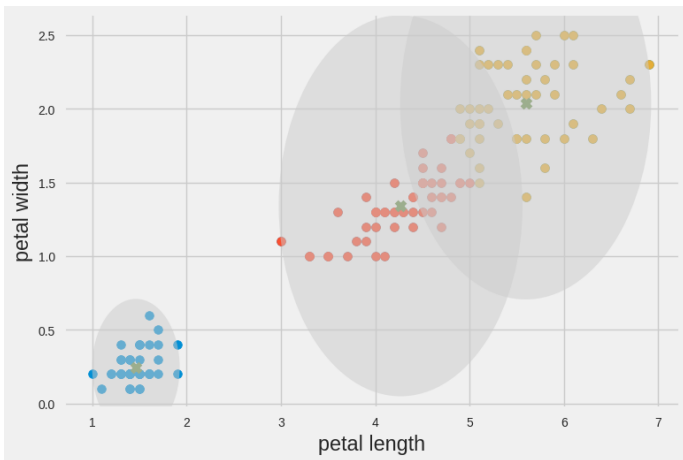
```
from sklearn.cluster import KMeans
import pandas as pd
import seaborn as sns

df = sns.load_dataset('iris')
X = df[['petal_length', 'petal_width']]

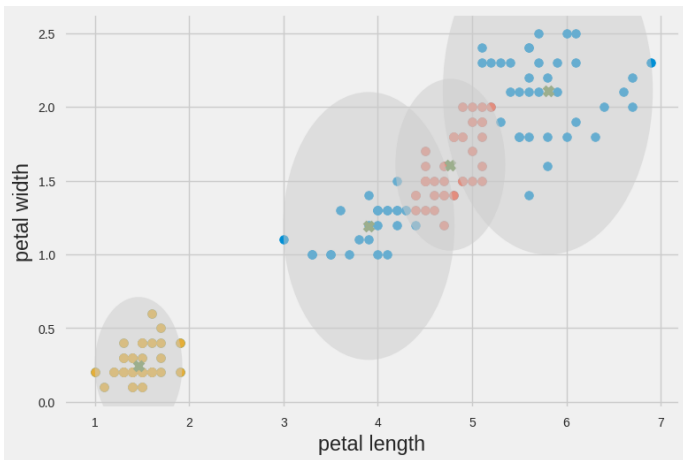
clf = KMeans(2)
clf.fit(X)
print(clf.labels_)
print(clf.cluster_centers_)
```



Gambar: k-Means dengan $k = 2$



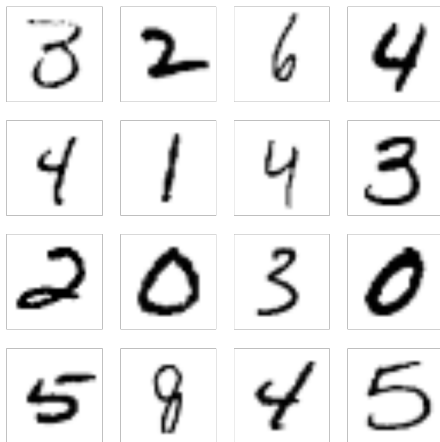
Gambar: k-Means dengan $k = 3$



Gambar: k-Means dengan $k = 4$

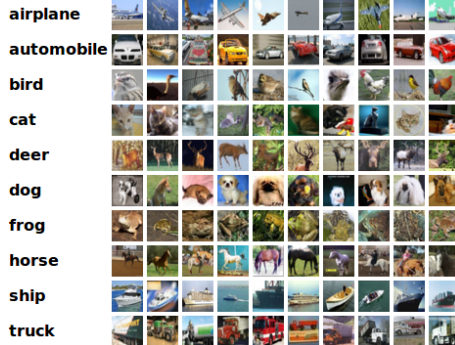
Neural Networks

Klasifikasi Digit



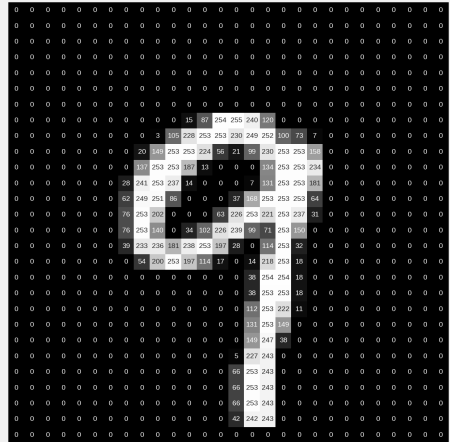
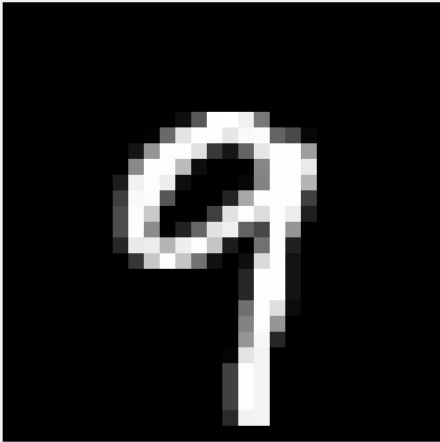
Gambar: Contoh data MNIST [O'Shea, 2016]

Klasifikasi Objek pada Gambar

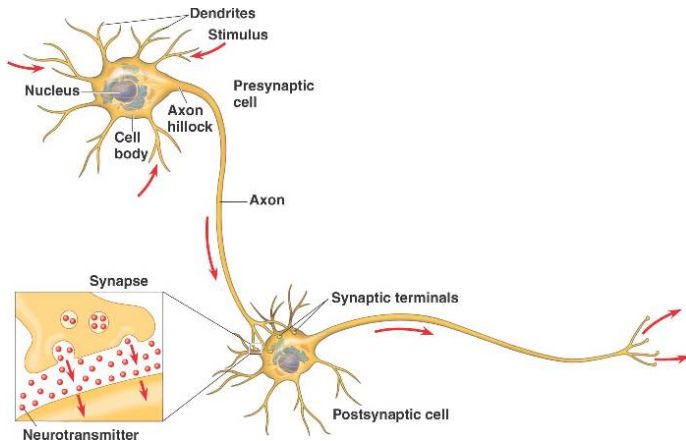


Gambar: Dataset CIFAR-10 [Krizhevsky, 2009]

Klasifikasi Digit

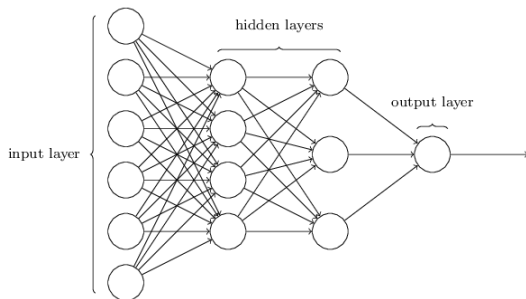


Jaringan Saraf



Gambar: Jaringan saraf

Jaringan Saraf Tiruan



Gambar: Ekstraksi fitur sebelum klasifikasi dengan jaringan saraf tiruan [Nielsen, 2016]

Prediksi Kemacetan



Gambar: Prediksi kemacetan dari gambar CCTV [Wibisono, 2016]

Keras: Deep Learning library for Theano and TensorFlow

Example

```
from keras.applications.resnet50 import ResNet50, preprocess_input
from keras.preprocessing import image
import numpy as np

resnet = ResNet50(include_top=False)

def extract_features(img_paths, batch_size=64):
    global resnet
    n = len(img_paths)
    img_array = np.zeros((n, 224, 224, 3))

    for i, path in enumerate(img_paths):
        img = image.load_img(path, target_size=(224, 224))
        img = image.img_to_array(img)
        img = np.expand_dims(img, axis=0)
        x = preprocess_input(img)
        img_array[i] = x

    X = resnet.predict(img_array, batch_size=batch_size, verbose=1)
    X = X.reshape((n, 2048))
    return X
```


Example

```
from keras.layers import Dense, Dropout
from keras.models import Sequential
from sklearn.model_selection import train_test_split

# X, y = code for obtaining image features and labels
X_train, X_test, y_train, y_test = train_test_split(X, y, train_size=0.8)

model = Sequential()
model.add(Dense(256, activation='relu', input_dim=2048))
model.add(Dense(1, activation='sigmoid'))
model.compile('adam', 'binary_crossentropy', metrics=['accuracy', 'fmeasure'])

model.fit(X_train, y_train, validation_data=(X_test, y_test))
```

- Neural networks sudah sedemikian canggihnya hingga bisa menghasilkan model generatif, i.e. menghasilkan data baru!
- Generative Adversarial Networks (GAN), Variational Autoencoder (VAE), Recurrent Neural Networks (RNN) adalah contoh arsitektur yang digunakan
- Contoh implementasi:
 - ▶ WaveNet dari Google DeepMind untuk menghasilkan audio
<https://deepmind.com/blog/wavenet-generative-model-raw-audio/>
 - ▶ Menghasilkan animasi untuk kendali karakter game
http://theorangeduck.com/media/uploads/other_stuff/phasefunction.mov

References



Jake VanderPlas (2016)

What is Machine Learning?

[http:](http://nbviewer.jupyter.org/github/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/05.01-What-Is-Machine-Learning.ipynb)

[//nbviewer.jupyter.org/github/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/05.01-What-Is-Machine-Learning.ipynb](http://nbviewer.jupyter.org/github/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/05.01-What-Is-Machine-Learning.ipynb)



Tim O'Shea (Juli 2016)

MNIST Generative Adversarial Model in Keras

[http://www.kdnuggets.com/2016/07/](http://www.kdnuggets.com/2016/07/mnist-generative-adversarial-model-keras.html)

[mnist-generative-adversarial-model-keras.html](http://www.kdnuggets.com/2016/07/mnist-generative-adversarial-model-keras.html)



Alex Krizhevsky (2009)

Learning Multiple Layers of Features from Tiny Images

<https://www.cs.toronto.edu/~kriz/cifar.html>

References



Michael Nielsen (2016)

Neural Networks and Deep Learning

<http://neuralnetworksanddeeplearning.com/>



Okiriza Wibisono (2016)

Keras Applications: Membuat Model Deteksi Gambar dengan Mudah

<https://tentangdata.wordpress.com/2016/12/26/>

[keras-applications-membuat-model-deteksi-gambar-dengan-mudah/](https://tentangdata.wordpress.com/2016/12/26/keras-applications-membuat-model-deteksi-gambar-dengan-mudah/)

Terima kasih