

k-Nearest Neighbours & k-Means Clustering

Ali Akbar Septiandri

August 9, 2017

Universitas Al Azhar Indonesia

Table of contents

1. k-Nearest Neighbours
2. k-Means Clustering

k-Nearest Neighbours

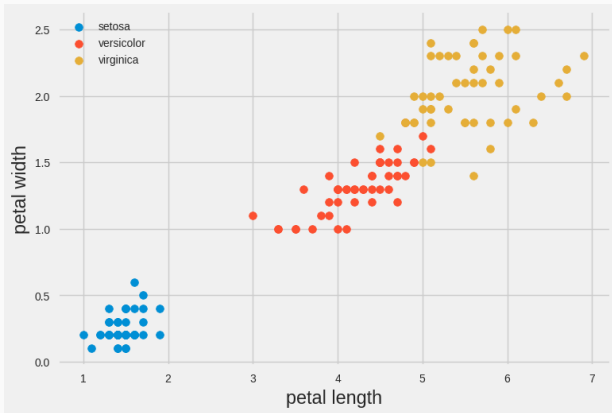
Deskripsi Dataset

- 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

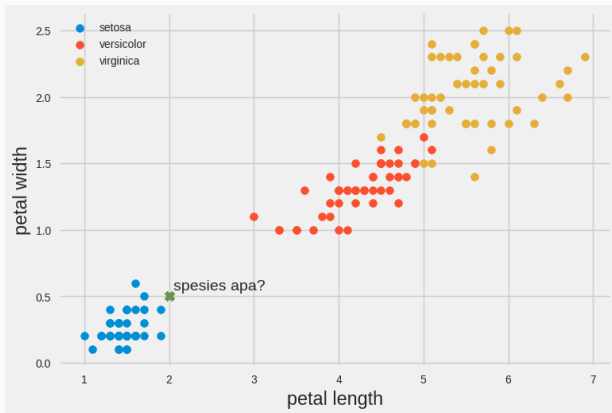


Figure 1: Tanaman Iris

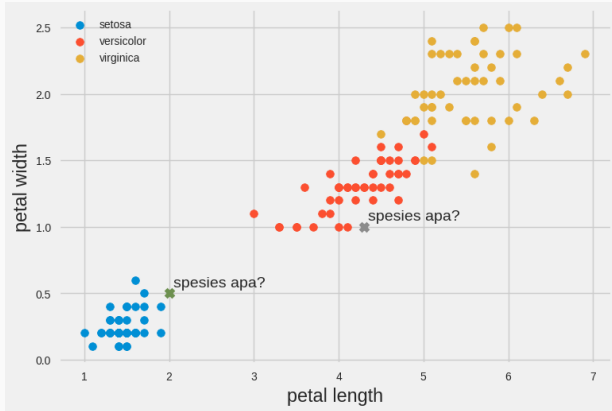
Iris Dataset



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

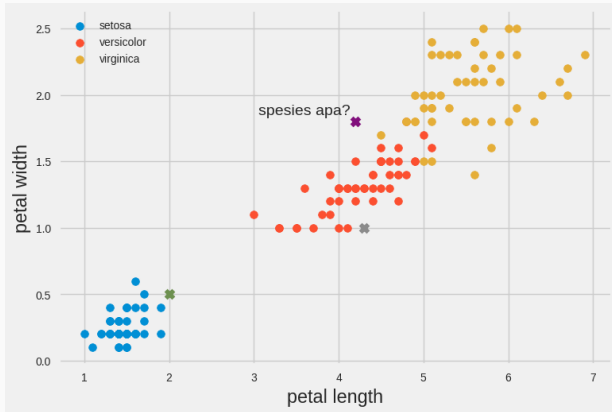


Figure 2: Seberapa yakin kita dengan referensi terdekat?

k-Nearest Neighbours

- Mencari referensi dari beberapa (k) tetangga terdekat

k-Nearest Neighbours

- Mencari referensi dari beberapa (k) tetangga terdekat
- Melihat label mayoritas dari tetangga 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

k-Nearest Neighbours

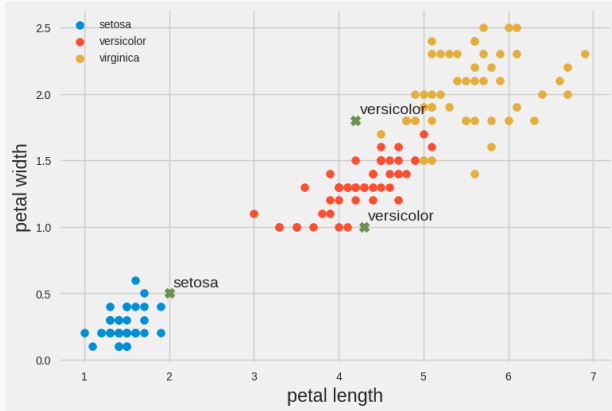
- Mencari referensi dari beberapa (k) tetangga terdekat
- Melihat label mayoritas dari tetangga terdekat
- Perhatikan bahwa harus dihitung jaraknya dengan semua data yang ada
- Kompleksitas: $O(nd)$

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!*

- Diketahui
 - data latih $\{x_i, y_i\}$
 - x_i : nilai atribut
 - y_i : label kelas
 - *instance* uji x
- Algoritma:
 1. Hitung jarak $D(x, x_i)$ untuk semua x_i
 2. Pilih k tetangga terdekat dengan labelnya
 3. \hat{y} = mayoritas dari label tetangga terdekat

Prediksi



Klasifikasi k-NN



Figure 3: 7-NN pada data MNIST dengan data uji di paling kanan

- Diketahui
 - data latih $\{x_i, y_i\}$
 - x_i : nilai atribut
 - y_i : nilai numerik sebenarnya
 - *instance* uji x
- Algoritma:
 1. Hitung jarak $D(x, x_i)$ untuk semua x_i
 2. Pilih k tetangga terdekat dengan labelnya
 3. $\hat{y} = f(x) = \frac{1}{k} \sum_{j=1}^k y_{ij}$ (nilai rata-rata)

Regresi k-NN

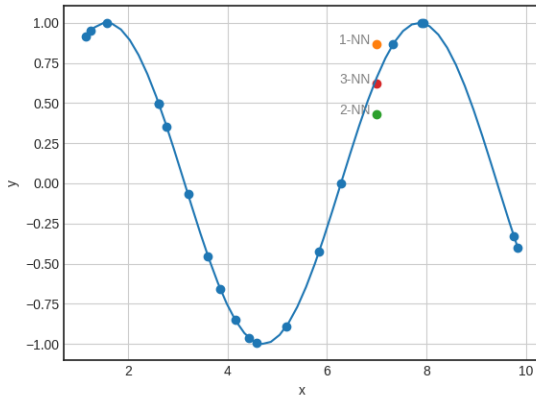


Figure 4: Interpolasi dengan $\{1,2,3\}$ -NN

Regresi k-NN

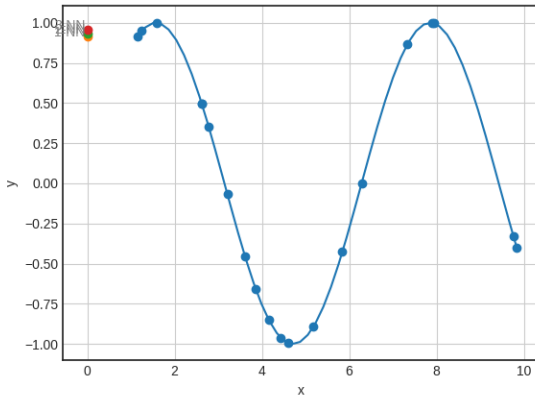


Figure 5: Ekstrapolasi dengan $\{1,2,3\}$ -NN

Bagaimana cara memilih nilai k ?

- Nilai yang besar $\rightarrow P(y)$ atau \bar{y}
- Nilai yang kecil \rightarrow terlalu variatif, batas keputusan yang tidak stabil

- Nilai yang besar $\rightarrow P(y)$ atau \bar{y}
- Nilai yang kecil \rightarrow terlalu variatif, batas keputusan yang tidak stabil
- **Solusi:** Gunakan data validasi!

Batas Keputusan

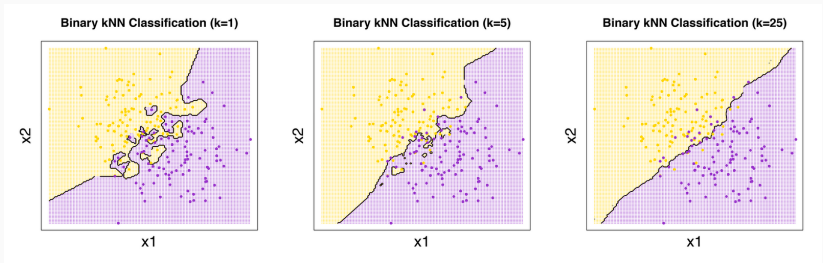


Figure 6: Pengaruh nilai k pada batas keputusan [DeWilde, 2012]

- Hasil seri:
 1. Gunakan jumlah k ganjil
 2. Acak, lemparan koin
 3. *Prior probability*
 4. 1-NN
- *Missing values*: harus diganti (*impute*)
- Rentan terhadap perbedaan rentang variabel

Perbedaan Rentang

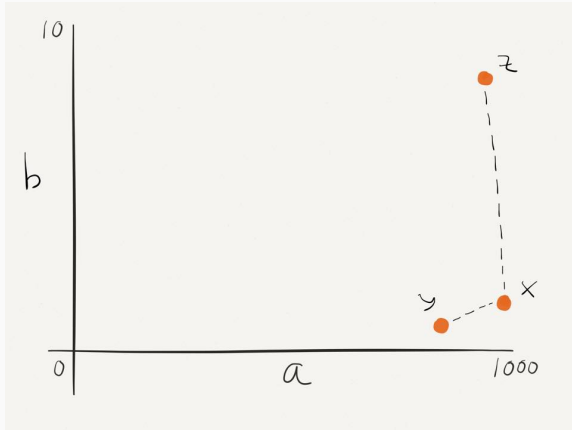


Figure 7: Perbedaan rentang variabel bisa mengacaukan klasifikasi k-NN
[Wibisono, 2015]

- Pros:
 - Tidak ada asumsi terhadap data, non-parametrik
 - *Asymptotically correct*
- Cons:
 - Harus mengganti nilai yang hilang
 - Sensitif terhadap kelas pencilan (data latih yang salah dilabeli)
 - Sensitif terhadap atribut yang irelevan
 - **Mahal secara komputasi** $O(nd)$

k-Means Clustering

- *Unsupervised learning*

- *Unsupervised learning*
- **Subpopulasi** apa yang ada dalam data?

- *Unsupervised learning*
- Subpopulasi apa yang ada dalam data?
- Apa kesamaan dari elemen di tiap subpopulasi?

- *Unsupervised learning*
- Subpopulasi apa yang ada dalam data?
- Apa kesamaan dari elemen di tiap subpopulasi?
- Bisa digunakan untuk menemukan pencila

Contoh Data

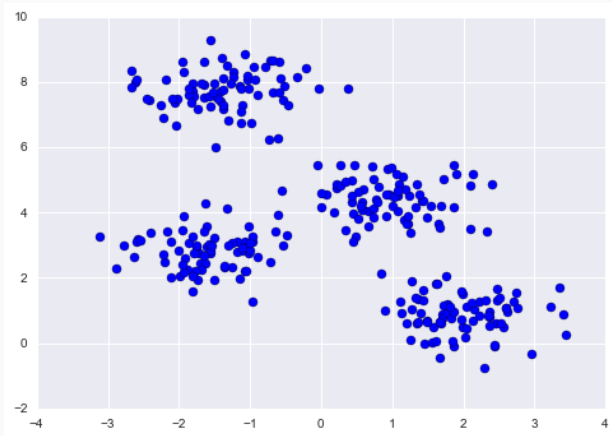


Figure 8: Contoh data dalam 2D [VanderPlas, 2016]

Subpopulasi

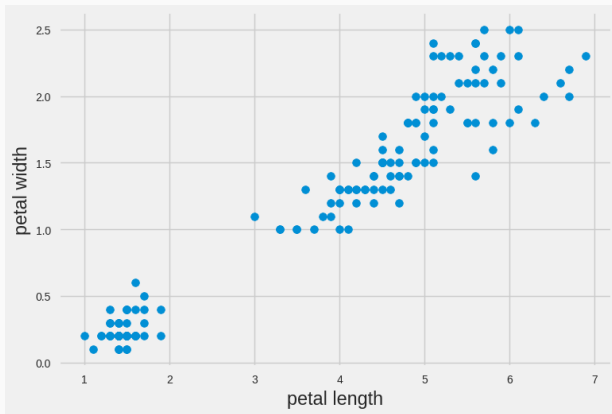


Figure 9: Subpopulasi dari algoritma k-Means [VanderPlas, 2016]

**Tetangga dalam satu kompleks,
tanpa memedulikan kelasnya**

Berapa jumlah subpopulasi (klaster)
yang ingin kita cari?

Data Iris



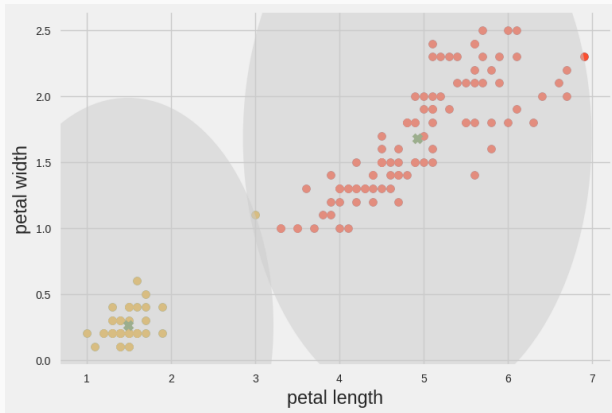


Figure 10: k-Means dengan $k = 2$

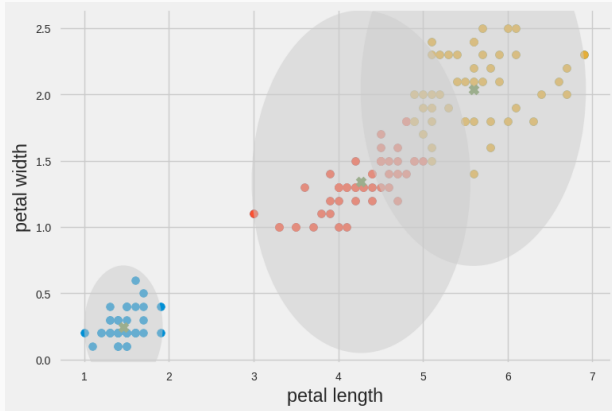


Figure 11: k-Means dengan $k = 3$

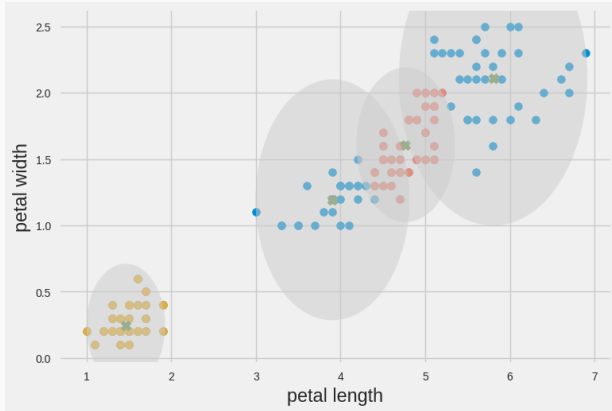


Figure 12: k-Means dengan $k = 4$

- Jumlah k ditentukan dari awal
- Tidak memerlukan label
- Menggunakan *centroid*, i.e. rata-rata nilai dari objek yang masuk dalam *cluster* tersebut
- Mencari *centroid* terdekat dari tiap objek

Algoritma: Expectation-Maximization

1. Inisialisasi k *centroid* secara acak
2. Ulangi hingga konvergen
 - A. E-step: Masukkan tiap titik/objek ke *centroid* terdekat

$$\arg \min_j D(x_i, c_j)$$

- B. M-step: Ubah nilai *centroid* menjadi rata-rata dari tiap titik/objek

$$c_j(a) = \frac{1}{n_j} \sum_{x_i \rightarrow c_j} x_i(a), \text{ for } a = 1..d$$

Visualisasi EM

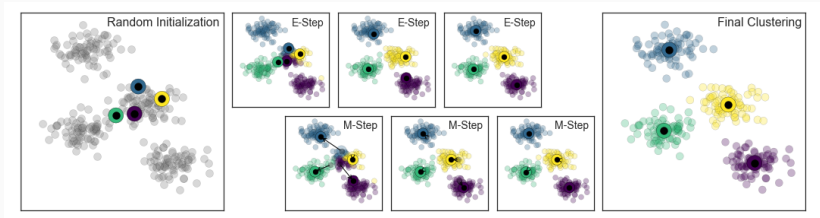


Figure 13: Konvergensi kluster tercapai hanya dalam tiga iterasi
[VanderPlas, 2016]

**Algoritma ini sangat bergantung
pada inisialisasi *centroid*!**

Berapa nilai k yang optimal?

Menentukan Nilai k

- Gunakan label kelas, e.g. 10 untuk MNIST
- Gunakan V untuk menggambarkan *scree plot*

$$V = \sum_j \sum_{x_i \rightarrow c_j} D(c_j, x_i)^2$$

lalu gunakan *elbow method*, i.e. nilainya dapat dicari dengan menggunakan nilai optimal turunan kedua

Scree Plot

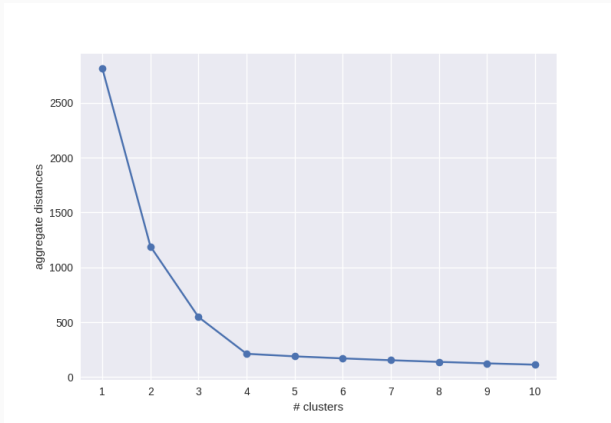


Figure 14: Secara visual, scree plot menunjukkan nilai optimal $k = 4$

Evaluasi Intrinsik: Klaster \sim Kelas

- Klaster c_1, c_2, \dots, c_K
- Kelas R_1, R_2, \dots, R_N
- Cocokkan R_i dengan c_j , hitung akurasinya

Contoh Evaluasi Intrinsik

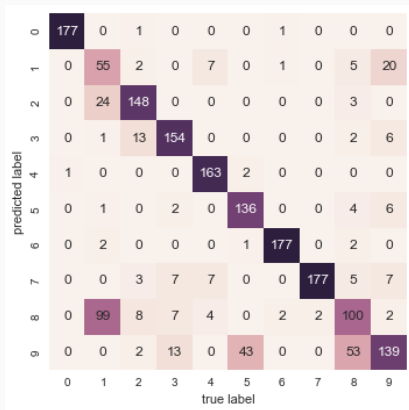


Figure 15: Confusion matrix dari MNIST clustering [VanderPlas, 2016]

Contoh Evaluasi Intrinsik

	G1	G2	G3	G4	G5	G6
C1	1	7	0	1	4	0
C2	0	0	0	0	2	7
C3	0	0	2	0	0	0
C4	3	1	0	0	1	0

Figure 16: Kluster karakter dalam Julius Caesar

- Representasi gambar: *bag of cluster id* atau fitur lain (lihat [Coates, 2012])
- Kompresi gambar (lihat [VanderPlas, 2016])
- Sistem rekomendasi

References



Burton DeWilde (26 Oktober 2012)

Classification of Hand-written Digits (3)

<http://bdewilde.github.io/blog/blogger/2012/10/26/classification-of-hand-written-digits-3/>



Okiriza Wibisono (16 September 2015)

kNN: Perhitungan Jarak, serta Batasan dan Keunggulan

<https://tentangdata.wordpress.com/2015/09/16/knn-perhitungan-jarak-serta-keunggulan-dan-batasan/>



Jake VanderPlas (2016)

In Depth: k-Means Clustering

<http://nbviewer.jupyter.org/github/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/05.11-K-Means.ipynb>



Adam Coates & Andrew Y. Ng (2012)

Learning feature representations with k-means.

Neural networks: Tricks of the trade (pp. 561-580). Springer
Berlin Heidelberg.

Thank you