k-Nearest Neighbours

Ali Akbar Septiandri

Universitas Al-Azhar Indonesia

aliakbars@live.com

June 15, 2019

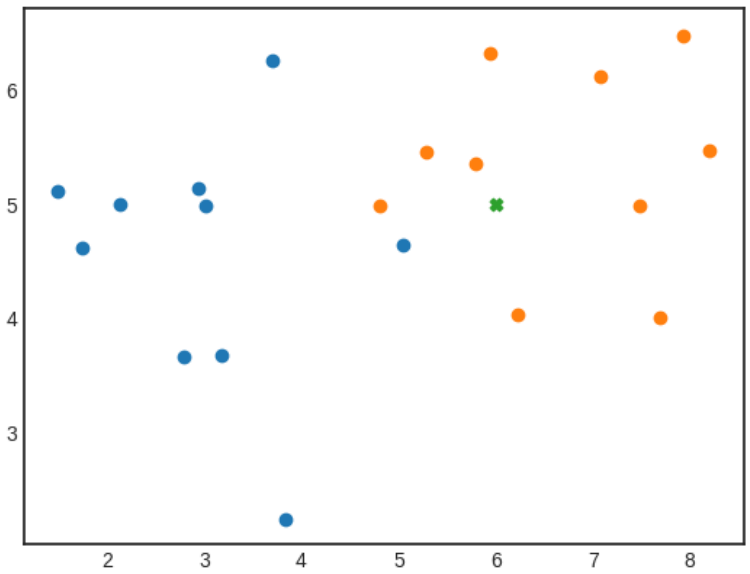
Selayang Pandang

Bahan Bacaan

1. Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann. (Section 4.7, 4.8, & 7.1)
2. VanderPlas, J. (2016). Python Data Science Handbook. (In Depth: k-Means Clustering) `http://nbviewer.jupyter.org/github/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/05.11-K-Means.ipynb`
3. "Klasifikasi: k-Nearest Neighbours." *Cerita Tentang Data*. 31 Agustus 2015. `https://tentangdata.wordpress.com/2015/08/31/klasifikasi-k-nearest-neighbours/`

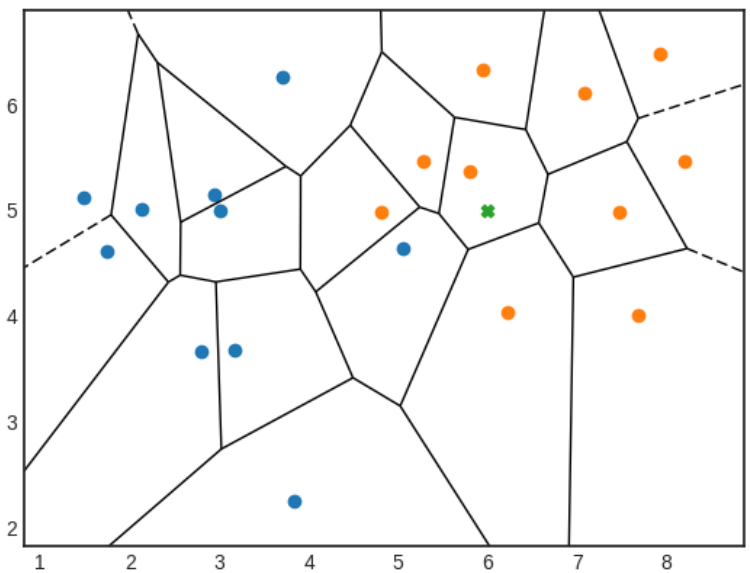
k-Nearest Neighbours

Intuisi



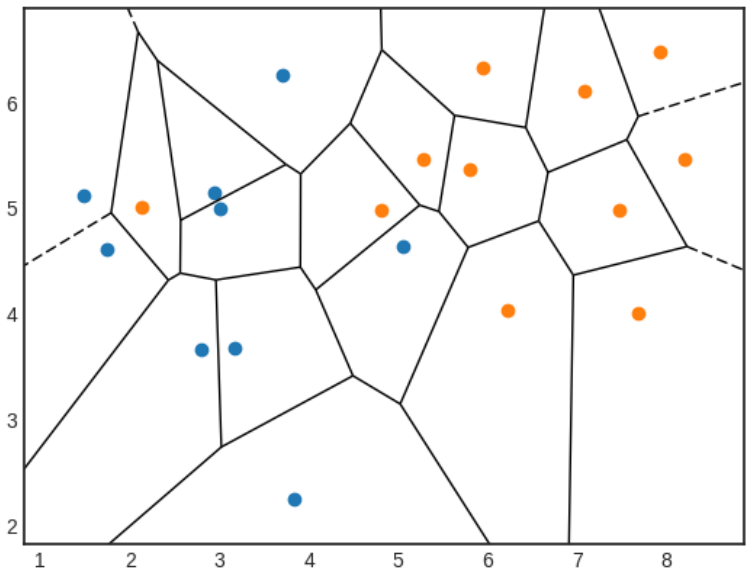
- Terdapat dua variabel: x_1, x_2
- Dua kelas: biru dan jingga
- Apa kelas dari *instance* tanda silang?

Klasifikasi Nearest-Neighbour



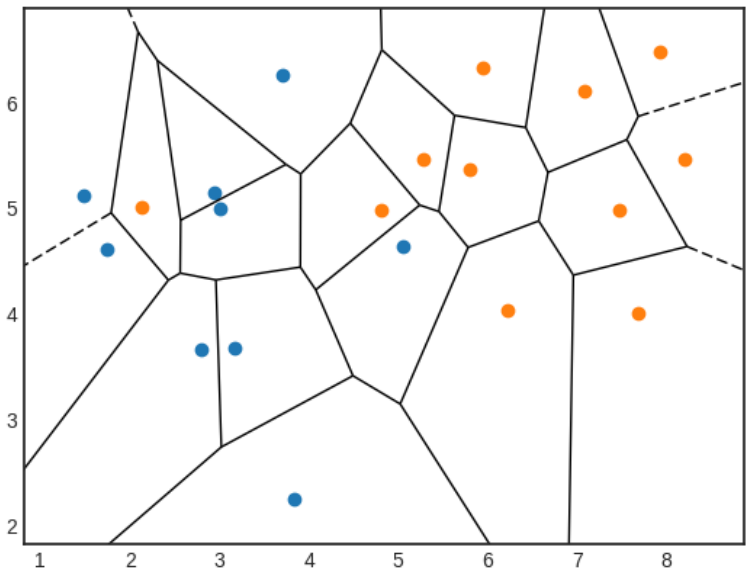
- Cari yang paling mirip, lalu gunakan kelas yang sama!
- *Voronoi tessellation*: membagi region dengan titik yang memiliki jarak yang sama dari dua contoh data latih
- Batas klasifikasi: non-linear

Pencilan



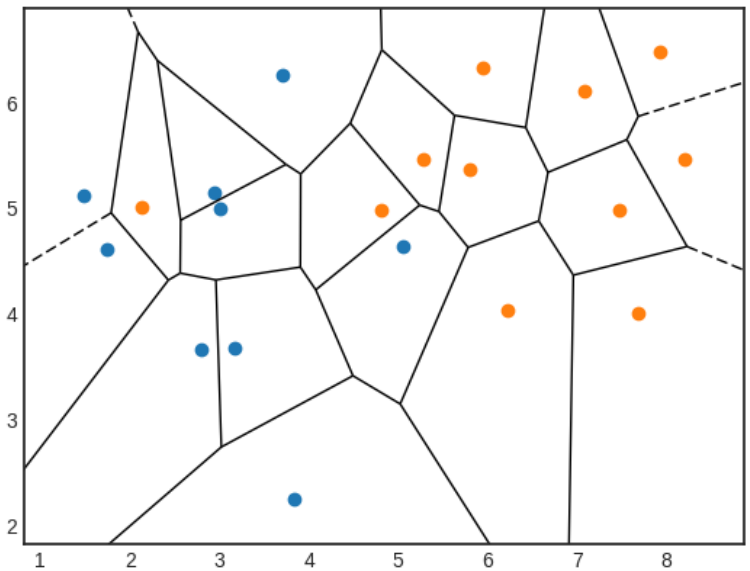
- Sensitif terhadap pencilan

Pencilan



- Sensitif terhadap pencilan
- Tidak ada $P(y|x)$

Pencilan



- Sensitif terhadap pencilan
- Tidak ada $P(y|x)$
- Tidak sensitif terhadap *class prior*

Perbaiki dengan menggunakan lebih dari satu tetangga (k-tetangga) terdekat!

Algoritma Klasifikasi

- 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

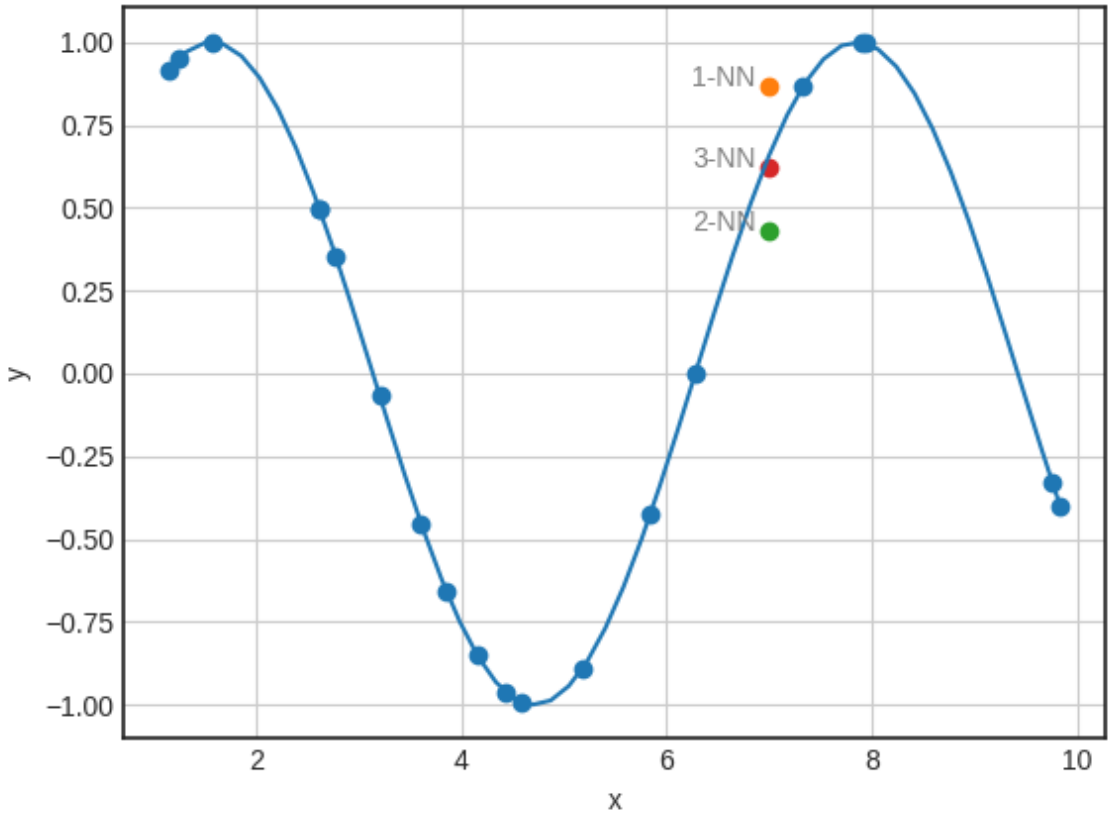
Klasifikasi k-NN

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

Algoritma Regresi

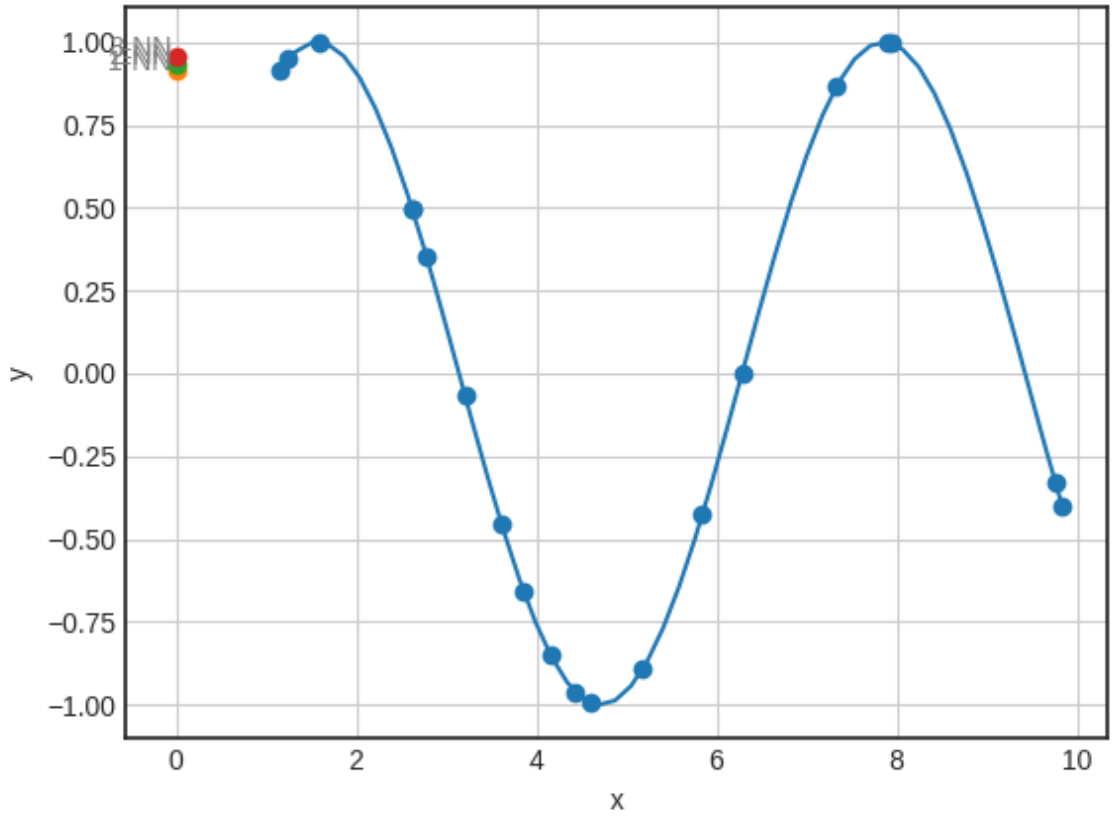
- Diketahui
  - data latih $\{x_i, y_i\}$
    - x_i: nilai atribut
    - y_i: nilai numerik sebenarnya
  - *instance* uji x
- Algoritma:
  1. Hitung jarak $D(x, x_i)$ untuk semua x_i
  2. Pilih k tetangga terdekat dengan labelnya
  3. $\hat{y} = f(x) = \frac{1}{k} \sum_{j=1}^{k} y_{ij}$ (nilai rata-rata)

Regresi k-NN



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

Regresi k-NN



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

Bagaimana cara memilih nilai k?

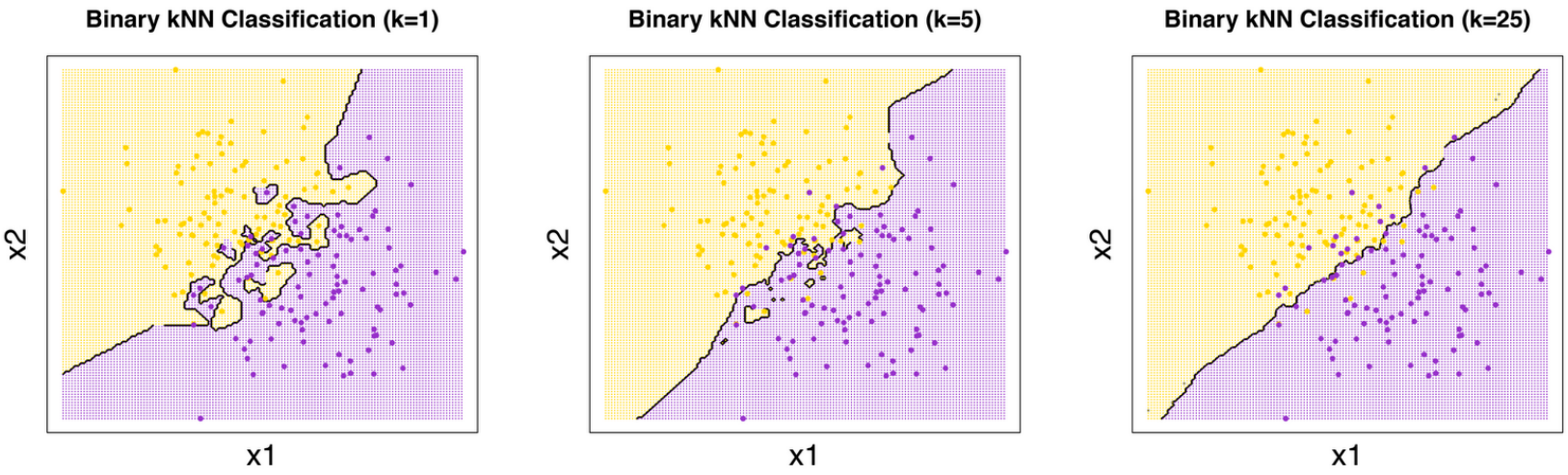
Memilih Nilai k

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

Memilih Nilai k

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

Batas Keputusan



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

Pengukuran Jarak

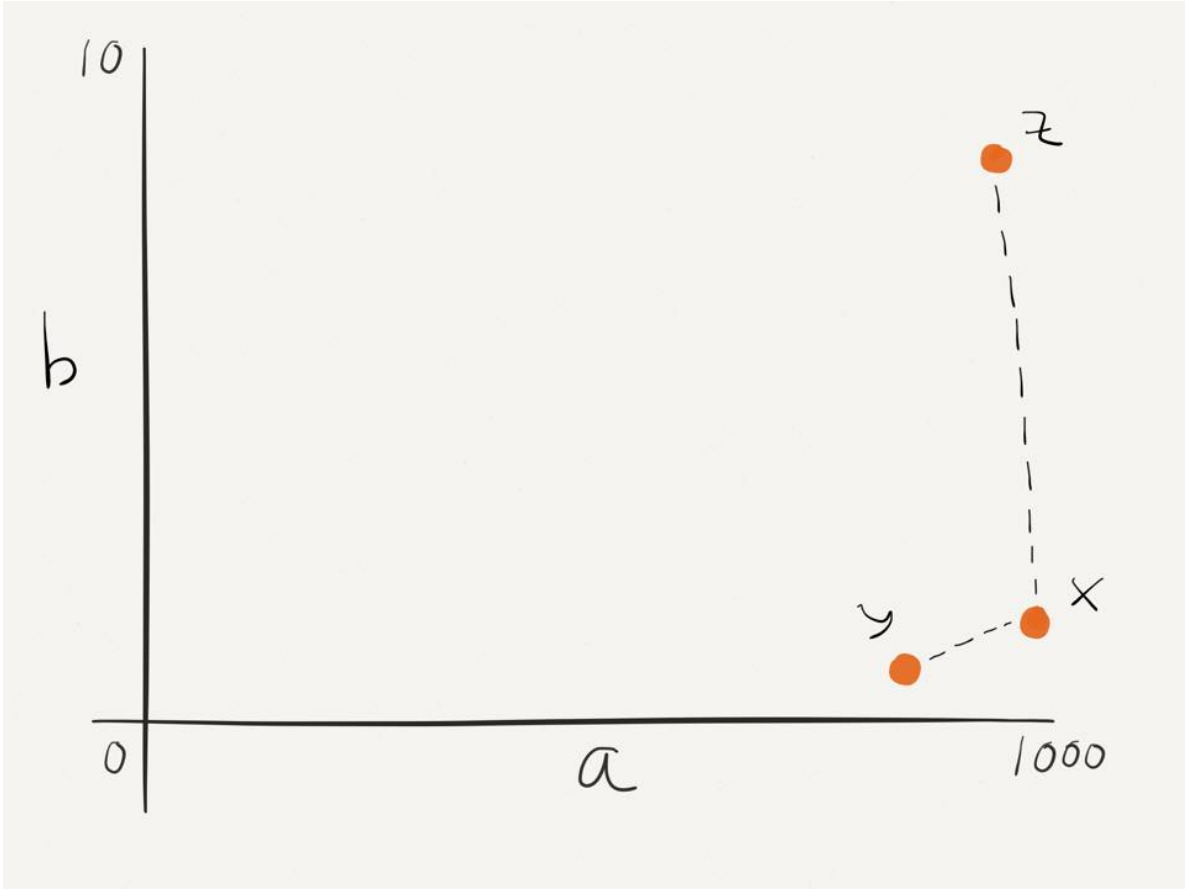
Minkowski distance (p-norm):

$$d([x_1, x_2, ..., x_n], [y_1, y_2, ..., y_n]) = \sqrt[r]{\sum_{i=1}^{n} |x_i - y_i|^r}$$

Isu

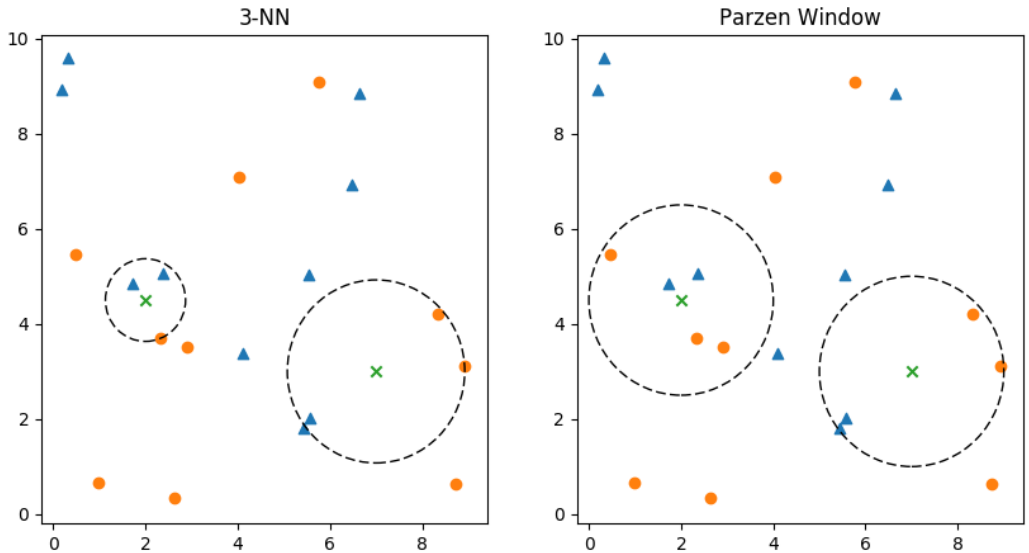
- Hasil seri:
  1. Gunakan jumlah k ganjil
  2. Acak, lemparan koin
  3. *Prior probability*
  4. 1-NN
- *Missing values:* **harus** diganti (*impute*)
- Rentan terhadap perbedaan rentang variabel

Perbedaan Rentang



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

k-NN vs Parzen Windows



Gambar: Perbedaan radius klasifikasi pada k-NN dan Parzen Windows

Pros & Cons

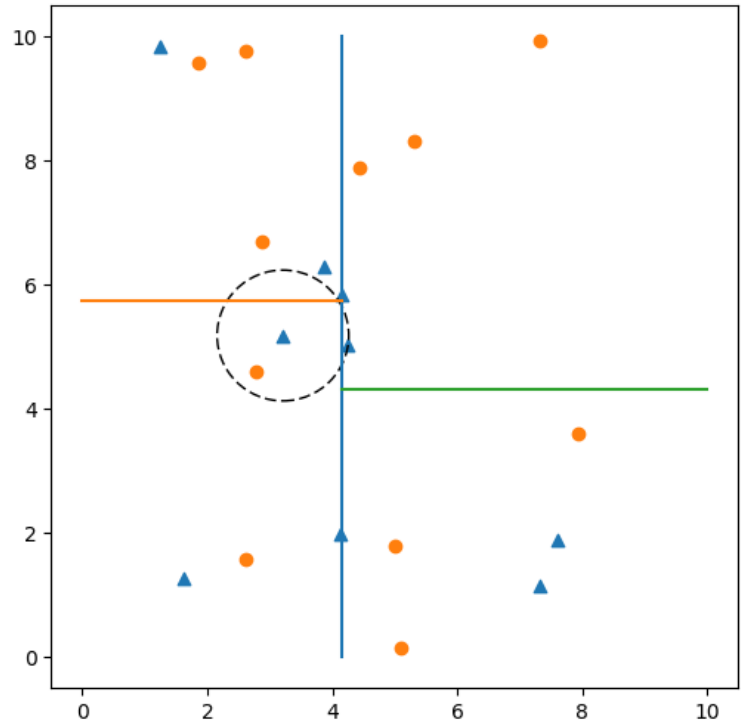
- Pros:
  - Tidak ada asumsi terhadap data, non-parametrik
  - *Asymptotically correct*
- Cons:
  - Harus mengganti nilai yang hilang
  - Sensitif terhadap kelas pencilan (data latih yang salah dilabeli)
  - Sensitif terhadap atribut yang irelevan
  - **Mahal secara komputasi** $O(nd)$

Mempercepat k-NN

- Pelatihan: $O(d)$, tetapi pengujian: $O(nd)$
- Mengurangi d: *dimensionality reduction*
- Mengurangi n: jangan bandingkan dengan **semua** data latih, i.e. cari $m \ll n$
  1. K-D trees
  2. Locality-sensitive hashing (LSH)
  3. Inverted lists

K-D Trees

Pilih dimensi secara acak, cari mediannya, pisahkan data, ulangi

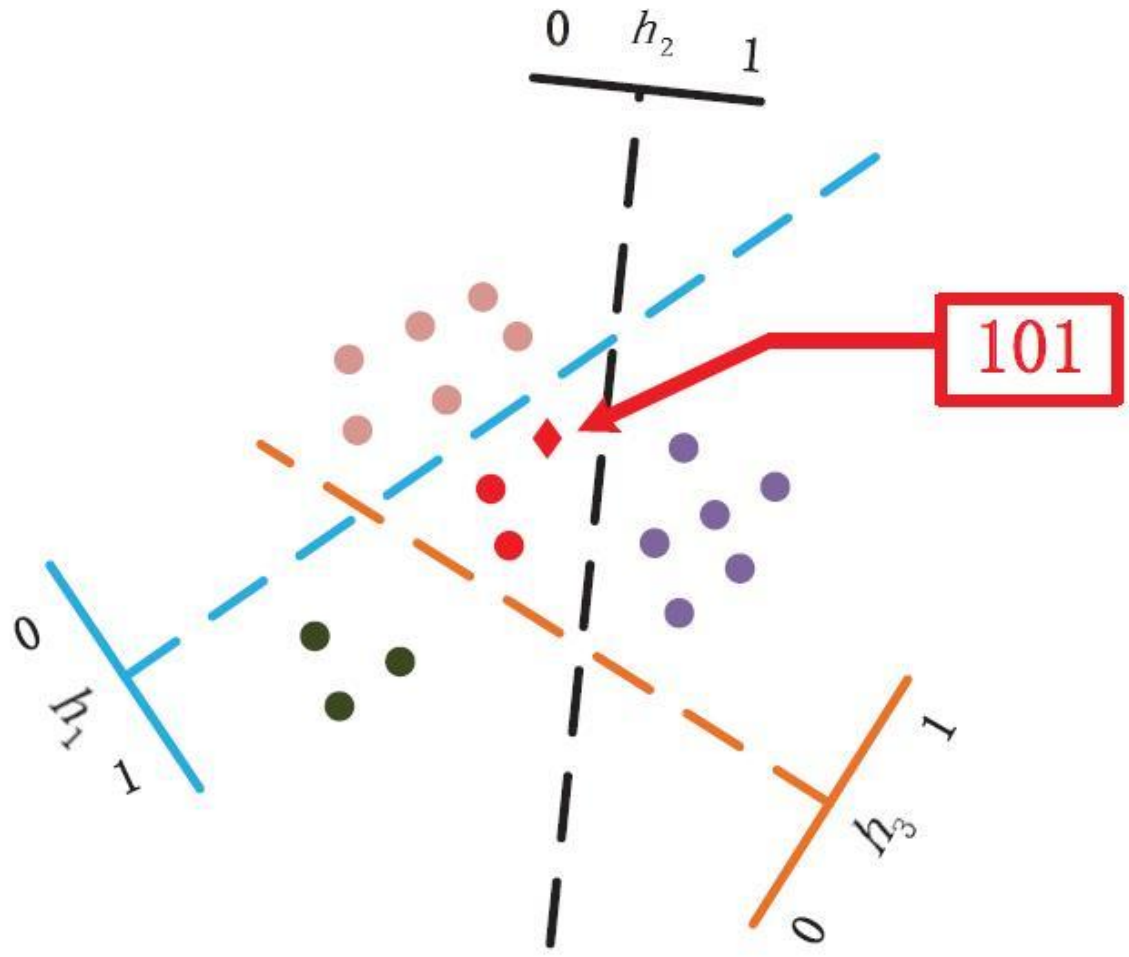


Gambar: 3-NN dari semua data berbeda dengan 3-NN yang berada pada region yang sama

Locality-Sensitive Hashing (LSH)

- Hyperplanes acak $h_1 ... h_k$ yang membagi ruang menjadi 2^k region
- Bandingkan x hanya dengan data latih dalam region yang sama: lakukan *dot-product* $\rightarrow$ *hash-code*
- Ada kemungkinan tetangga dekat yang terlewat: ulangi lagi dengan $h_1 ... h_k$ yang berbeda

Locality-Sensitive Hashing



Gambar: Menghasilkan *hash-code* dari *hyperplanes* [Li et al., 2017]

Inverted Lists

- Jika datanya berupa *bag-of-words*, matriksnya akan *sparse*
- Ide: buat daftar dokumen per atribut

Inverted Lists

D1: “send us your password” (s)
D2: “send us your review” (h)
D3: “review your password” (h)
D4: “review us” (s)
D5: “send your password” (s)
D6: “send us your account” (s)
Dokumen baru: “account review”

send $\rightarrow \{1, 2, 5, 6\}$
your $\rightarrow \{1, 2, 3, 5, 6\}$
review $\rightarrow \{3, 4\}$
account $\rightarrow \{6\}$
password $\rightarrow \{1, 3, 5\}$

Salindia ini dibuat dengan
sangat dipengaruhi oleh Lavrenko (2014)

Referensi

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/

Haisheng Li et al. (2017)

Feature Matching of Multi-view 3D Models Based on Hash Binary Encoding

Neural Network World. 27. 95-105.

Terima kasih