Data Mining

K-nearest Neighbors Algorithm

Definisi

- metode non-parametrik yang digunakan untuk klasifikasi dan regresi.
- Dalam kedua kasus, input terdiri dari k contoh pelatihan terdekat di ruang fitur.
- Outputnya tergantung pada apakah k-NN digunakan untuk klasifikasi atau regresi.

Klasifikasi

- Output berupa keanggotaan kelas.
- Suatu objek diklasifikasikan oleh suara pluralitas tetangganya, dengan objek yang ditugaskan ke kelas paling umum di antara tetangganya yang terdekat (k adalah bilangan bulat positif, biasanya kecil).
- Jika k = 1, maka objek hanya ditugaskan untuk kelas tetangga terdekat itu.

Regresi

- Dalam regresi k-NN, output adalah nilai properti untuk objek.
- Nilai ini adalah rata-rata dari nilai k tetangga terdekat.

Pembelajaran berbasis contoh

- k-NN adalah jenis pembelajaran berbasis contoh, atau *lazy learning*, di mana fungsinya hanya didekati secara lokal dan semua perhitungan ditangguhkan hingga klasifikasi.
- Algoritma k-NN adalah salah satu yang paling sederhana dari semua algoritma pembelajaran mesin.

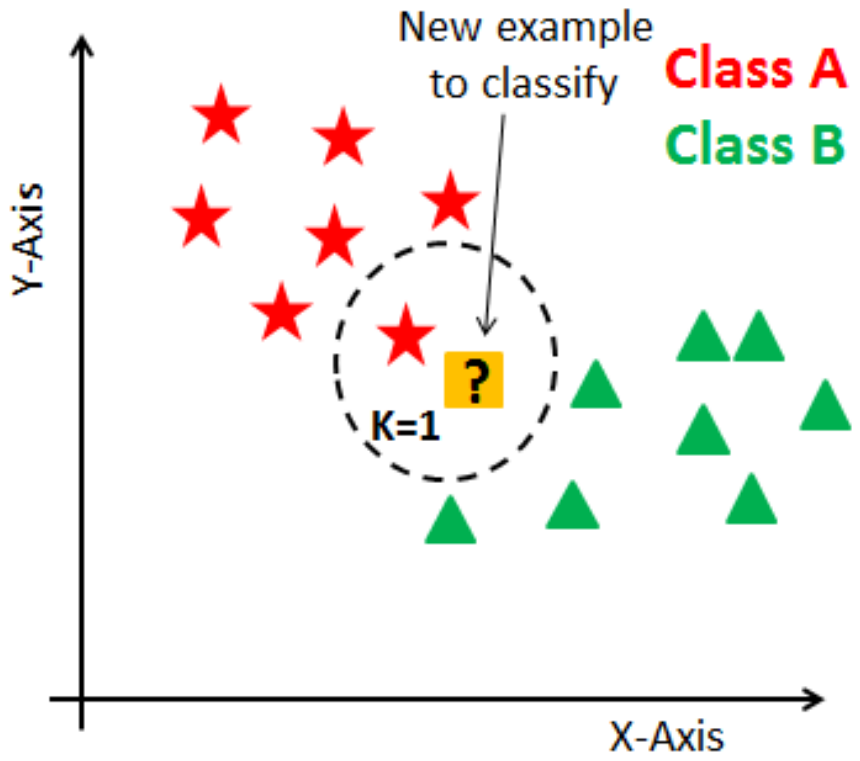
Lazy Learning

- Metode pembelajaran di mana generalisasi data pelatihan, ditunda hingga permintaan dibuat ke sistem.
- Berbeda dengan *eiger learning*, di mana sistem mencoba menggeneralisasi data pelatihan sebelum menerima permintaan.

Bobot tetangga

- K-NN menggunakan bobot pada kontribusi tetangga, sehingga tetangga yang lebih dekat dianggap berkontribusi lebih banyak daripada yang lebih jauh.
- Sebagai contoh, skema pembobotan umum terdiri dari memberikan masing-masing tetangga berat 1/d, di mana d adalah jarak ke tetangga.

Lebih dekat ke tetangga mana?



Algoritma k-NN

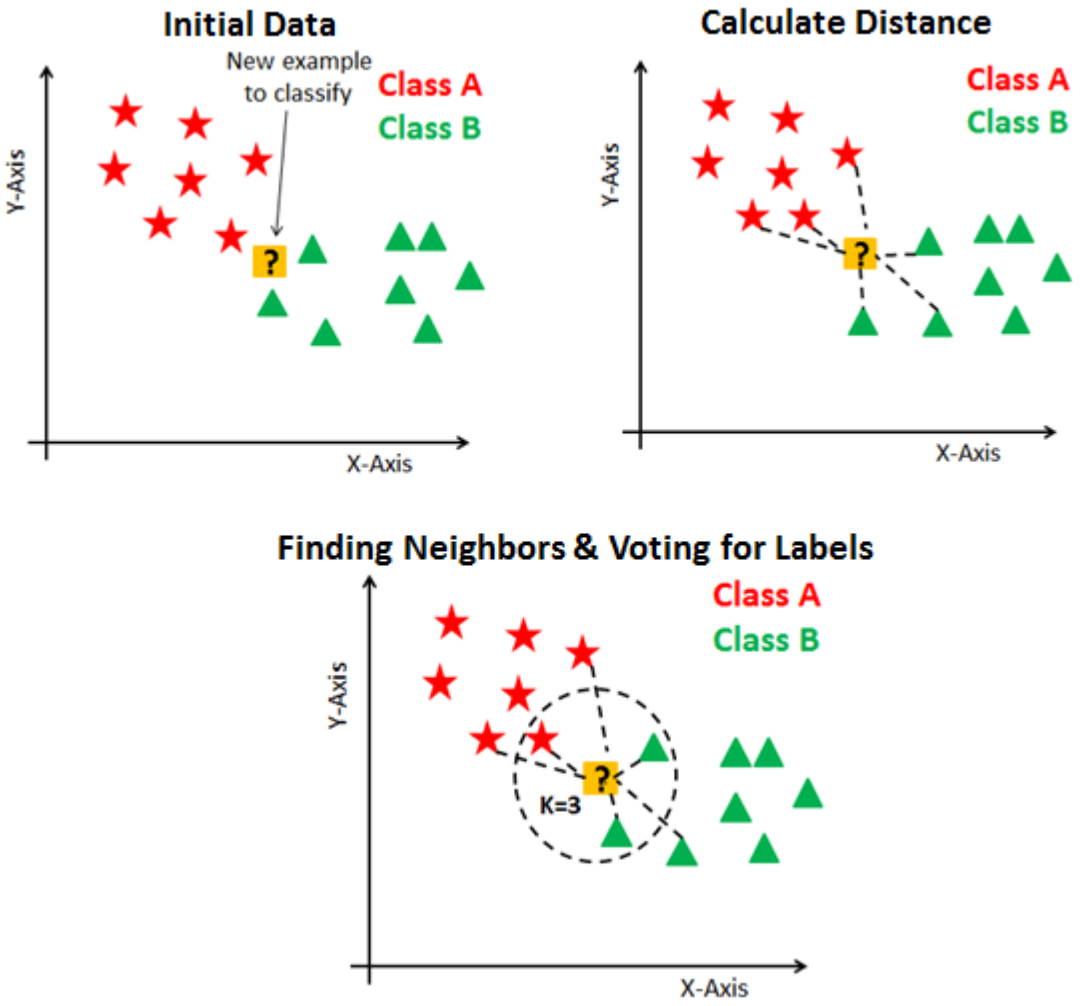
- Di KNN, K adalah jumlah tetangga terdekat. Jumlah tetangga adalah faktor penentu inti.
- K umumnya merupakan angka ganjil jika jumlah kelasnya adalah 2.
- Ketika K = 1, maka algoritma tersebut dikenal sebagai algoritma tetangga terdekat.

Pengukuran jarak tetangga terdekat

- Misalkan P1 perlu diprediksi labelnya.
- Pertama, temukan titik terdekat k dengan P1 dan kemudian mengklasifikasikan poin dengan suara terbanyak dari tetangganya.
- Setiap objek memberikan suara untuk kelas mereka dan kelas dengan suara terbanyak diambil sebagai prediksi.

Jarak vektor tetangga

- Untuk menemukan titik terdekat yang terdekat, Anda menemukan jarak antara titik menggunakan ukuran jarak :
  - Euclidean
  - Hamming
  - Manhattan
  - Minkowski


Initial Data
New example
to classify
Class A
Class B
Y-Axis
X-Axis
?
Calculate Distance
Class A
Class B
Y-Axis
X-Axis
?
Finding Neighbors & Voting for Labels
Class A
Class B
Y-Axis
X-Axis
?
K=3

Posisi menentukan prestasi?

Kenapa k-NN disebut lazy(malas)?

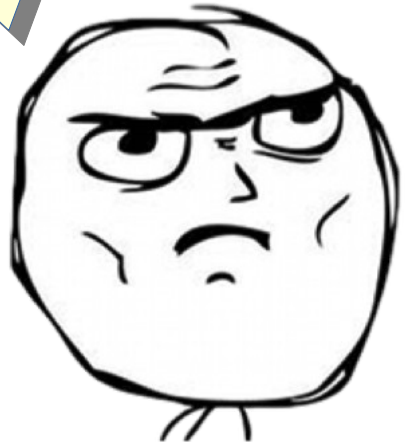
- Karena berbeda dengan SVM yang memetakan (generalisasi) *hyper-lane* dan *support vector* sebelum melakukan klasifikasi,
- K-NN baru bekerja mengukur jarak ketika dilakukan klasifikasi.

Logika pemalas (k-NN)...

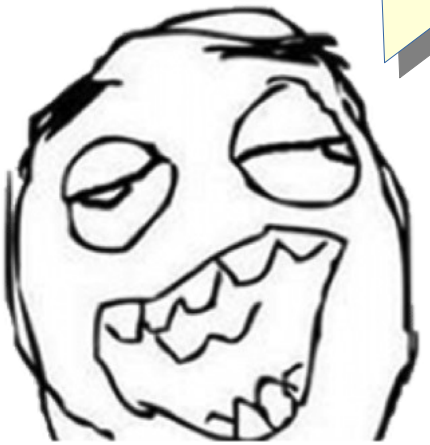
- Malas Belajar berarti tidak perlu untuk belajar atau melatih model dan semua titik data yang digunakan pada saat prediksi.
- Pelajar yang malas (k-NN) menunggu sampai menit terakhir sebelum mengklasifikasikan titik data apa pun.
- Pelajar malas menyimpan hanya set data pelatihan dan menunggu sampai klasifikasi perlu dilakukan.

Eigen vs Lazy

Ga atur strategi dulu?



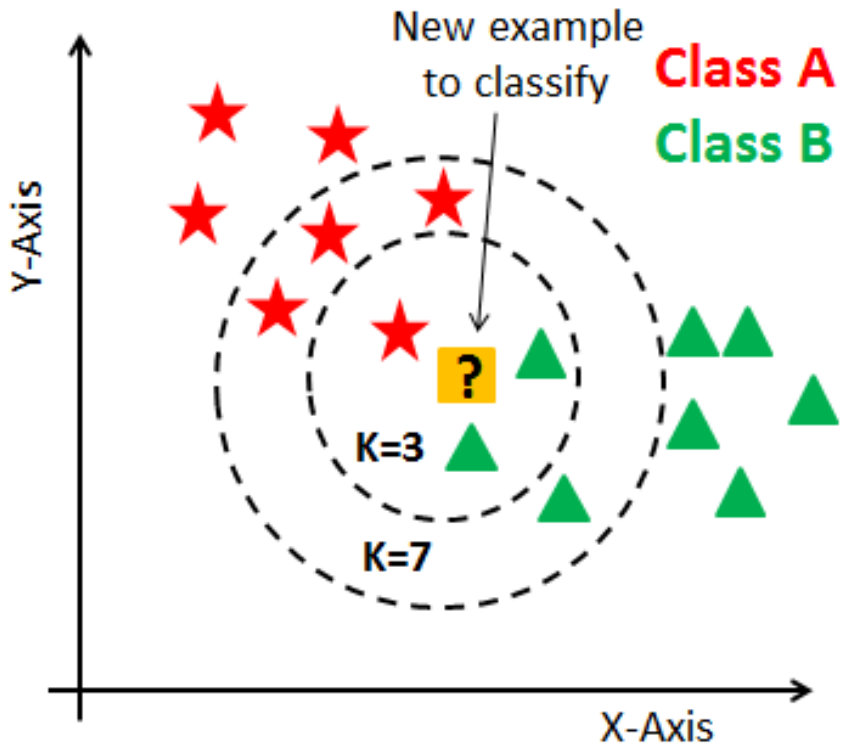
Aku ngandalin inting
Di lapangan aja braay...



Menentukan jumlah k

- Penelitian telah menunjukkan bahwa tidak ada jumlah optimal tetangga yang cocok dengan semua jenis set data.
- Setiap dataset memiliki persyaratannya sendiri.
- Dalam kasus sejumlah kecil tetangga, kebisingan akan memiliki pengaruh yang lebih tinggi pada hasilnya, dan sejumlah besar tetangga membuatnya mahal secara komputasi.

K = diukur dengan berapa tetangga?



Kelebihan k-NN

- Tahap pelatihan klasifikasi tetangga K-terdekat jauh lebih cepat dibandingkan dengan algoritma klasifikasi lainnya.
- Tidak perlu melatih model untuk generalisasi, Itulah sebabnya KNN dikenal sebagai algoritma pembelajaran sederhana dan berbasis instance.
- KNN dapat berguna jika ada data nonlinear.
- Nilai output untuk objek dihitung oleh rata-rata nilai tetangga terdekat k.
- **KNN tepat untuk kasus training data yang bisa bertambah secara cepat!**

Kekurangan k-NN

- Tahap pengujian klasifikasi tetangga K-terdekat lebih lambat dan lebih mahal dalam hal waktu dan memori.
- Ini membutuhkan memori yang besar untuk menyimpan seluruh dataset pelatihan untuk prediksi.
- KNN membutuhkan penskalaan data karena KNN menggunakan jarak Euclidean antara dua titik data untuk menemukan tetangga terdekat.
- Jarak Euclidean sensitif terhadap besaran.
- Fitur dengan magnitudo tinggi akan lebih berat daripada fitur dengan magnitudo rendah.
- **KNN juga tidak cocok untuk data dimensi besar (kebalikan SVM).**