Machine Learning Algorithms

Tommy (tommy.winarta@inergi.id)

Review

- Langkah-langkah:
    1. Merumuskan masalah
    2. Mengumpulkan dataset
    3. Exploratory Data Analysis
    4. Training
    5. Evaluasi

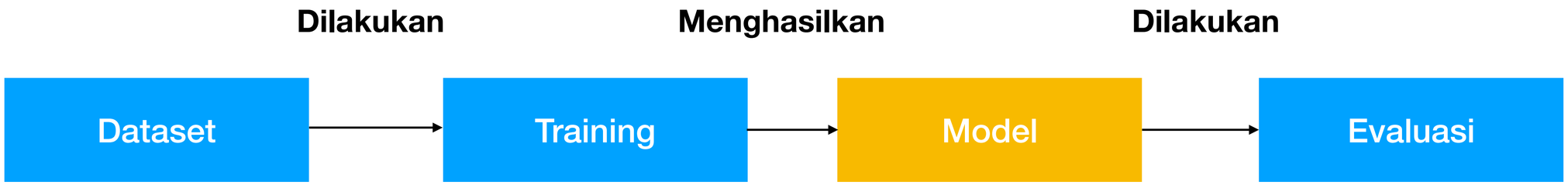
Tommy (tommy.winarta@inergi.id)

Review

- Machine learning tasks:
  - Classification
  - Regression
  - Clustering
  - Time-series Prediction

Tommy (tommy.winarta@inergi.id)

Training



- Struktur & size dari file model tergantung dari **algoritma** yang digunakan

Terhadap testing set & validation set

Tommy (tommy.winarta@inergi.id)

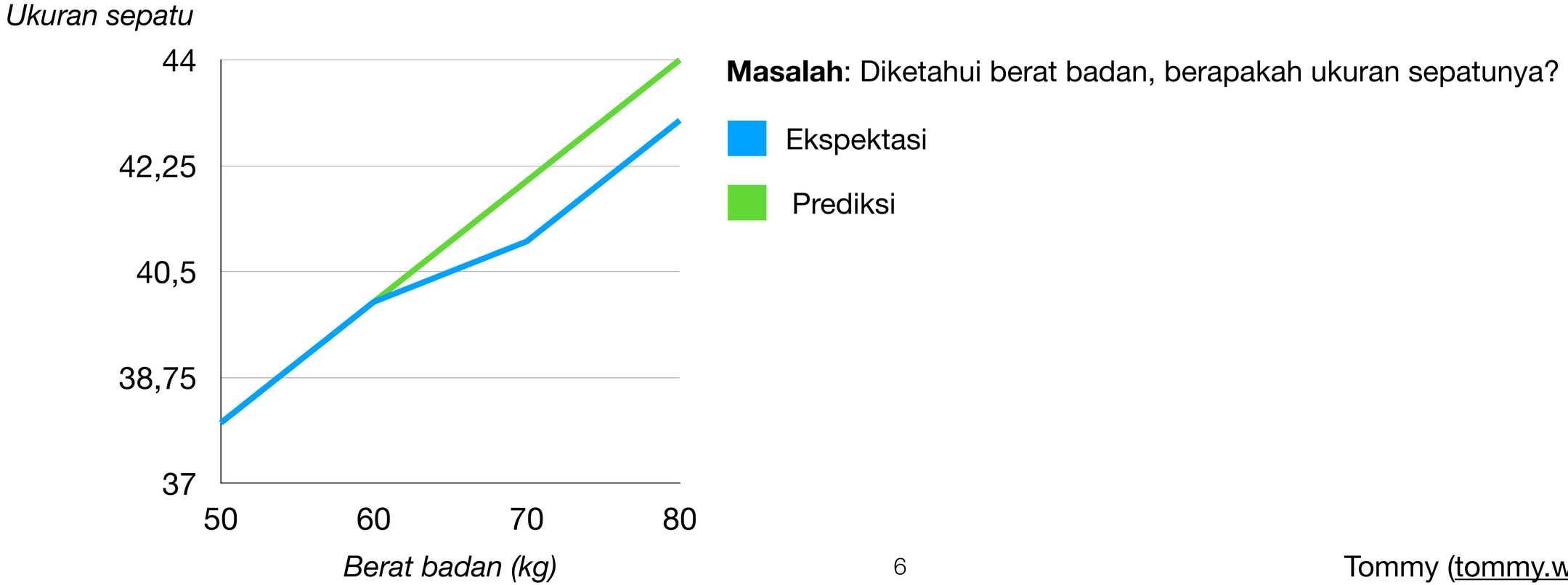
Algoritma?

- Tujuan:
  - Memastikan bahwa **output** yang dihasilkan ketika melakukan **prediksi** menggunakan **model hasil training** sesuai dengan ekspektasi (akurat)
  - Setiap algoritma memiliki cara masing-masing dalam mempelajari pola/ pattern pada data
  - Tidak ada algoritma yang pasti bekerja baik untuk memecahkan semua masalah

Tommy (tommy.winarta@inergi.id)

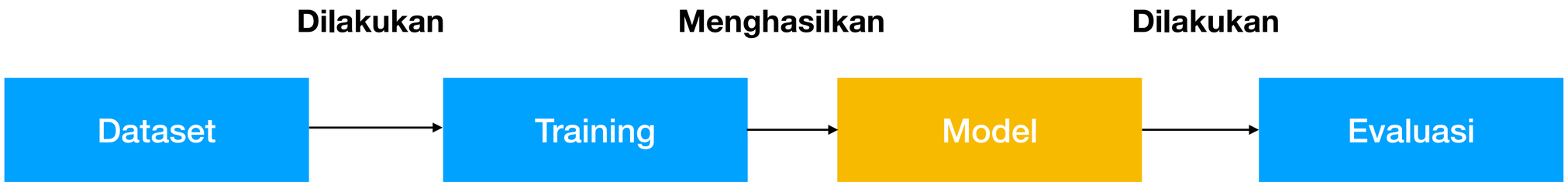
Algoritma?

- Tujuan:
  - Memastikan bahwa **output** yang dihasilkan ketika melakukan **prediksi** menggunakan **model hasil training** sesuai dengan ekspektasi (akurat)



Tommy (tommy.winarta@inergi.id)

Training



- Training dilakukan menggunakan banyak algoritma dan hasil evaluasinya dibandingkan.
- Pada akhirnya, model dengan performa terbaik yang akan digunakan.

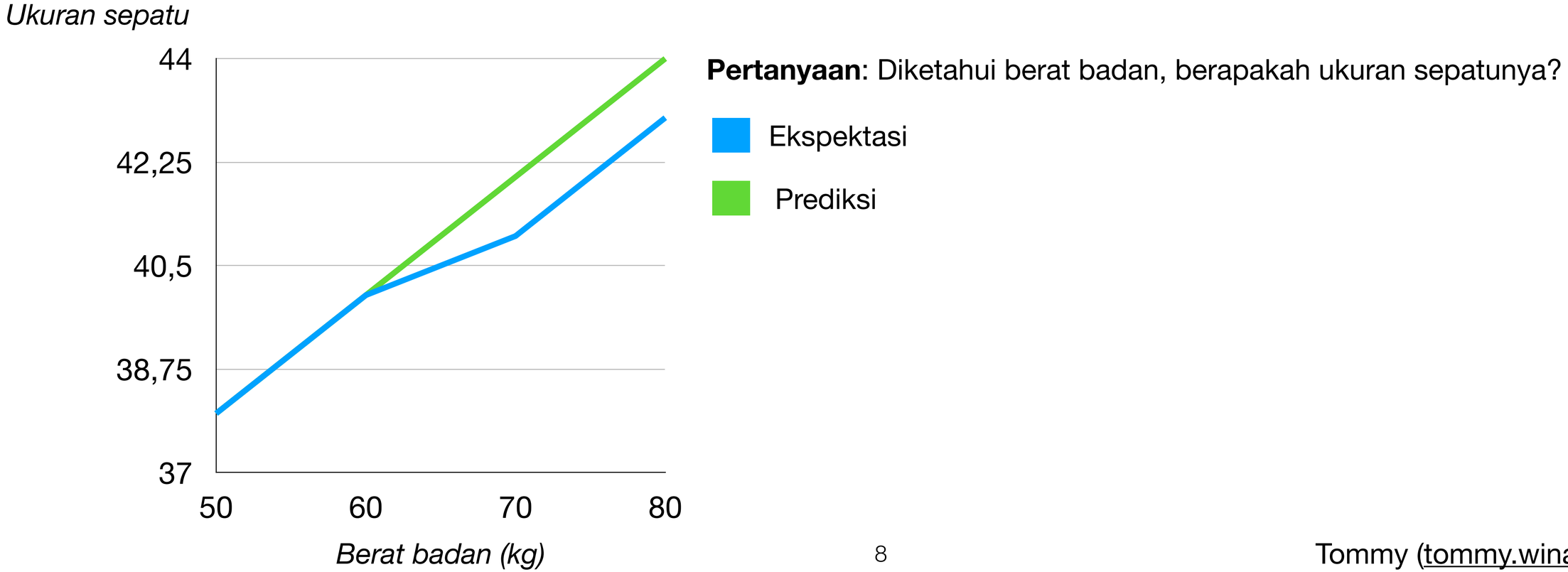
- Struktur & size dari file model tergantung dari **algoritma** yang digunakan

Terhadap testing set & validation set

Tommy (tommy.winarta@inergi.id)

Algoritma?

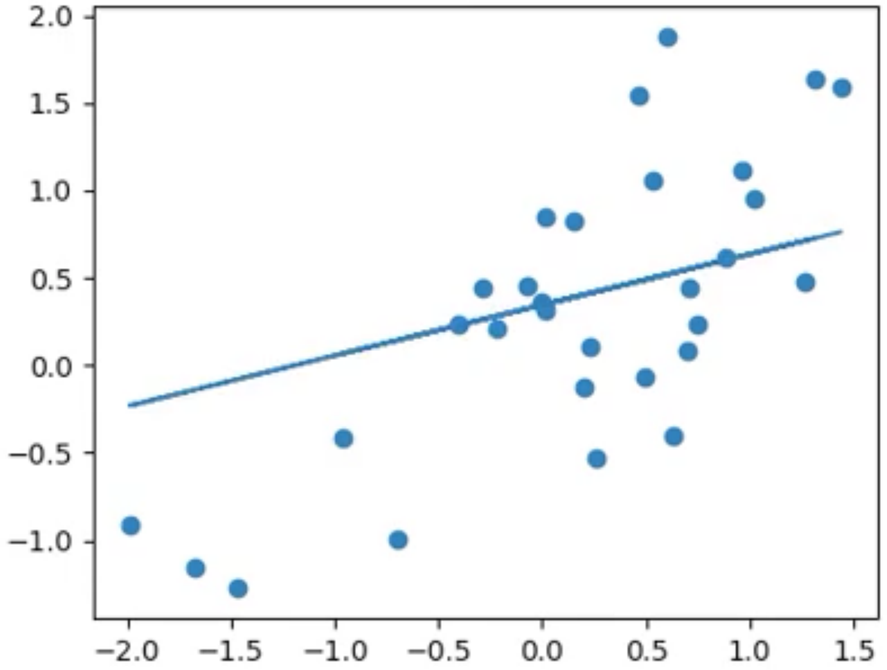
- Tujuan:
  - Memastikan bahwa **output** yang dihasilkan ketika melakukan **prediksi** menggunakan **model hasil training** sesuai dengan ekspektasi (akurat)



Tommy (tommy.winarta@inergi.id)

Algoritma?

- Tujuan:
  - Memastikan bahwa **output** yang dihasilkan ketika melakukan **prediksi** menggunakan **model hasil training** sesuai dengan ekspektasi (akurat)



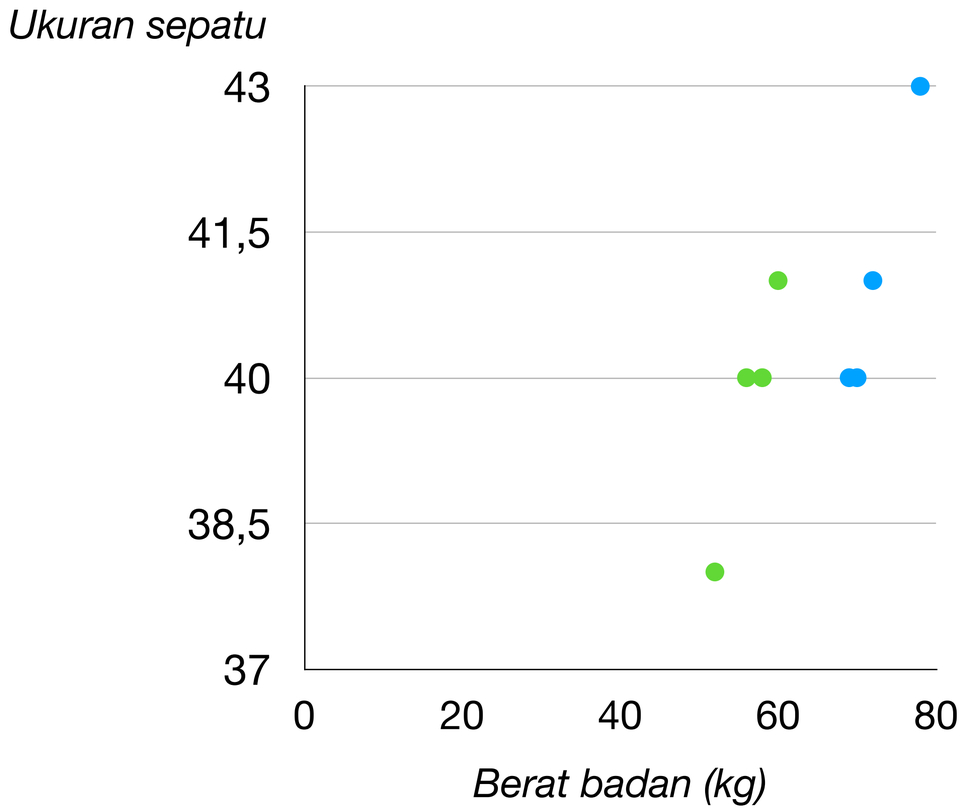
Contoh visualisasi proses training:
https://www.youtube.com/watch?v=J1Rl2qhxg0I

Tommy (tommy.winarta@inergi.id)

Beberapa Contoh Algoritma

Tommy (tommy.winarta@inergi.id)

k-Nearest Neighbor



Input:
Apabila ada titik baru, misal:
- Berat badan: 61 kg
- Ukuran sepatu: 38

Output:
Pria/wanita?

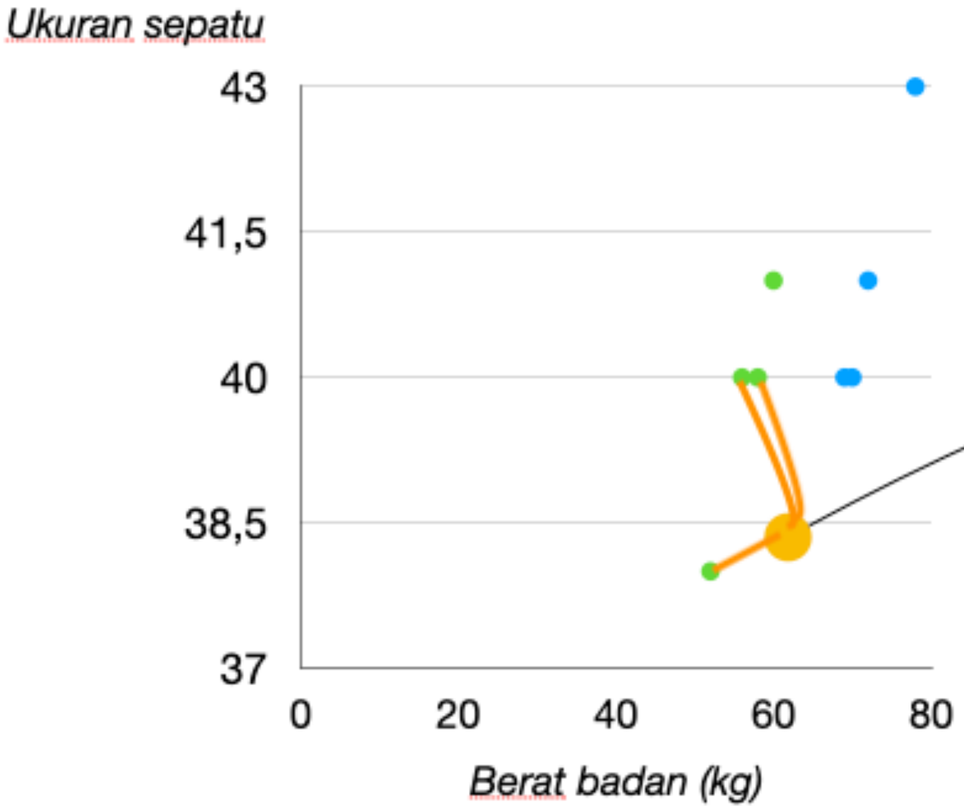
Klasifikasi

Menentukan label menurut *k* data tetangga terdekat.
Dimana *k* merupakan {0, 1, …. n}

Nilai *k* disebut sebagai **hyperparameter** atau variabel pada algoritma yang dapat di-set untuk menyesuaikan output.

Tommy (tommy.winarta@inergi.id)

k-Nearest Neighbor



Pria

Wanita

Input:
Apabila ada titik baru, misal:

- Berat badan: 61 kg
- Ukuran sepatu: 38

Output:
Pria/wanita?

Menentukan label menurut *k* data tetangga terdekat.
Dimana *k* merupakan {0, 1, …. n}

Misal: *k = 3*

Catatan: *k* harus bernilai ganjil untuk mendapatkan nilai mayoritas

Jarak diketahui dengan menghitung **jarak kedua titik**

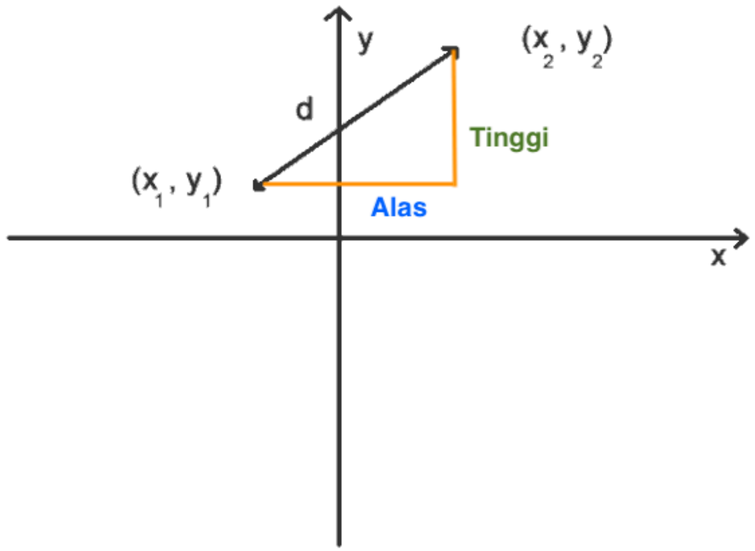
Tommy (tommy.winarta@inergi.id)

k-Nearest Neighbor

Jarak diketahui dengan menghitung **jarak kedua titik (*Euclidean distance*)**

$$d=\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

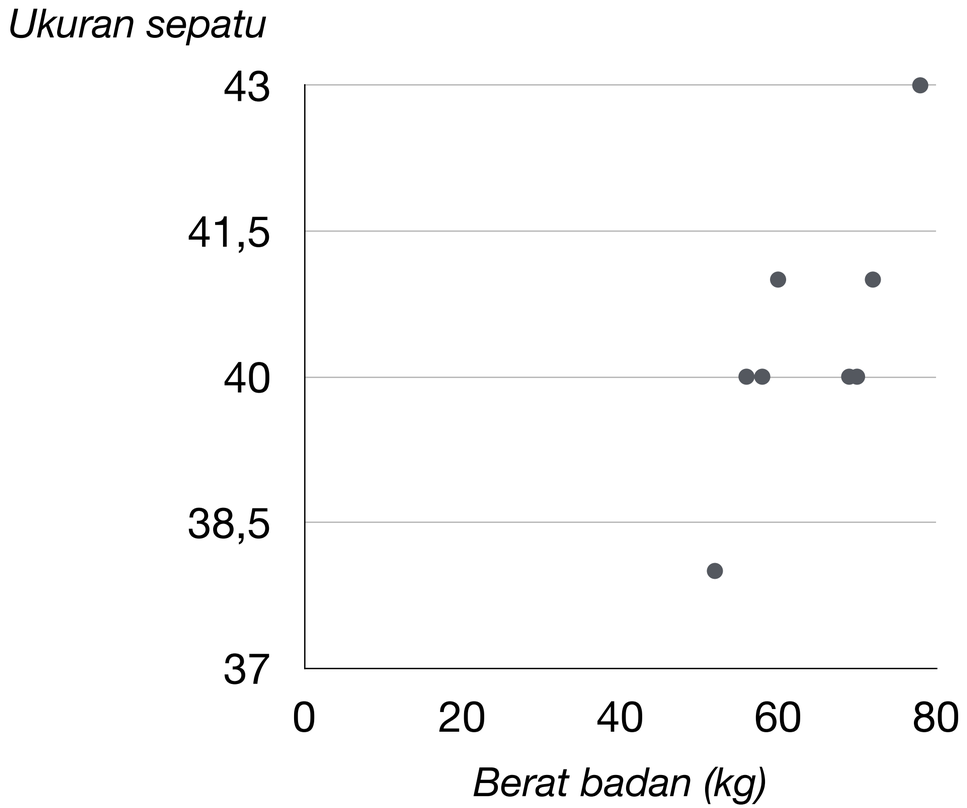
Alas Tinggi



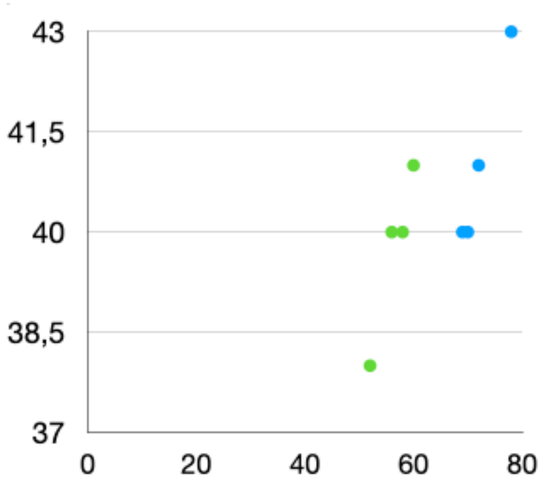
https://www.oreilly.com/library/view/hands-on-recommendation-systems/9781788993753/assets/1c808a35-3c9d-4bbe-a6ae-e858a3961159.png

Tommy (tommy.winarta@inergi.id)

k-Means Clustering



Bandingkan dengan:



Input:
Data yang sudah ada

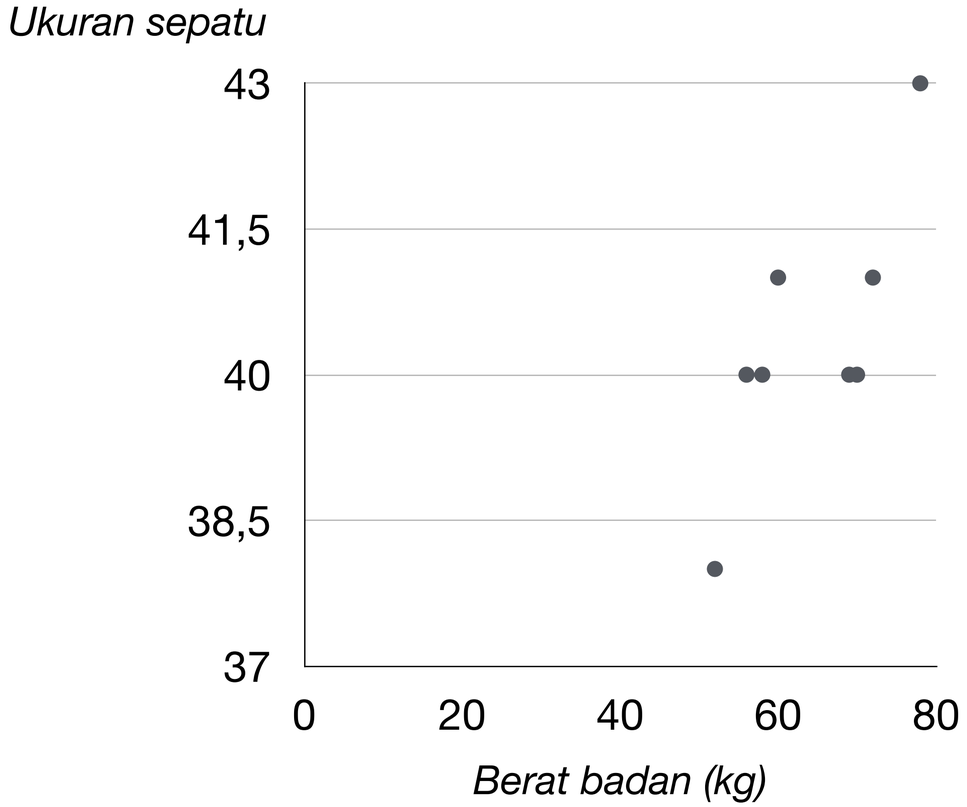
Output:
Data yang sudah dikategorikan

Mengkategorikan data menjadi *k* kategori.
Dimana *k* merupakan {0, 1, …. n}
Nilai *k* tidak harus bernilai ganjil

Nilai *k* merupakan **hyperparameter**

Tommy (tommy.winarta@inergi.id)

k-Means Clustering



Langkah-langkah:

1. Buat *k* titik sebagai *centroid* (nilai tengah dari suatu kelompok)
2. Untuk masing-masing titik, cari nilai jarak dari titik tersebut ke *k* centroid
3. Tentukan centroid yang paling terdekat. Maka, titik tersebut masuk ke kelompok tersebut.

Input:
Data yang sudah ada

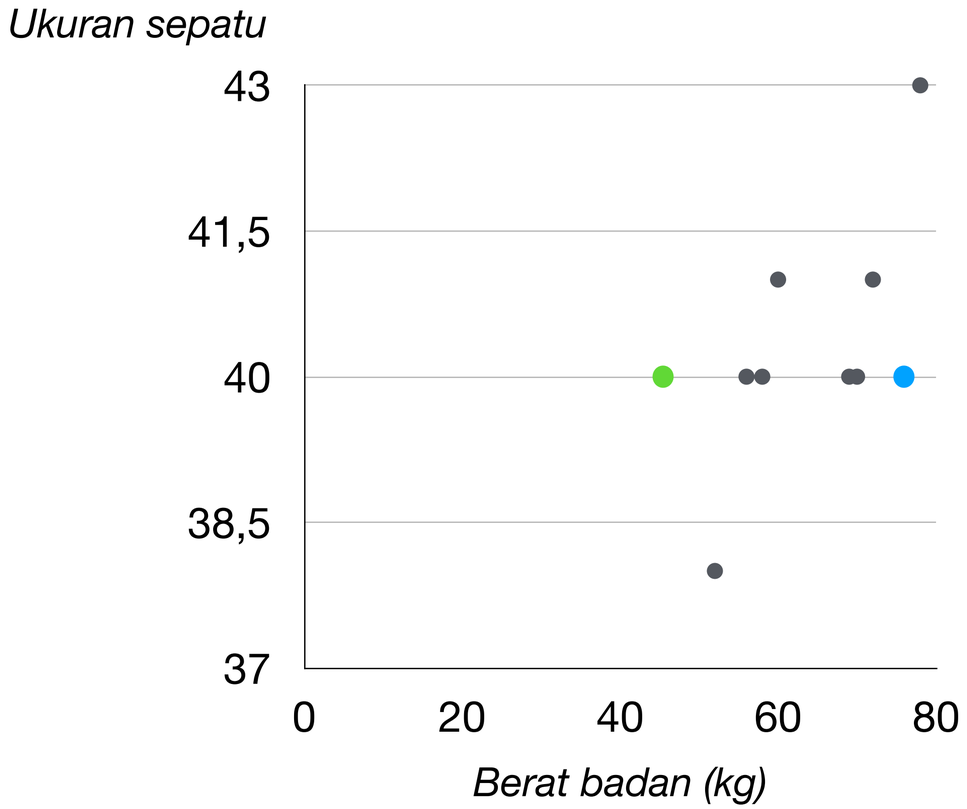
Output:
Data yang sudah dikategorikan

Mengkategorikan data menjadi *k* kategori.
Dimana *k* merupakan {0, 1, …. n}
Nilai *k* tidak harus bernilai ganjil

Nilai *k* merupakan **hyperparameter**

Tommy (tommy.winarta@inergi.id)

k-Means Clustering



Langkah-langkah:

1. Buat *k* titik sebagai *centroid* (nilai tengah dari suatu kelompok)
2. Untuk masing-masing titik, cari nilai jarak dari titik tersebut ke *k* centroid
3. Tentukan centroid yang paling terdekat. Maka, titik tersebut masuk ke kelompok tersebut.

Input:
Data yang sudah ada

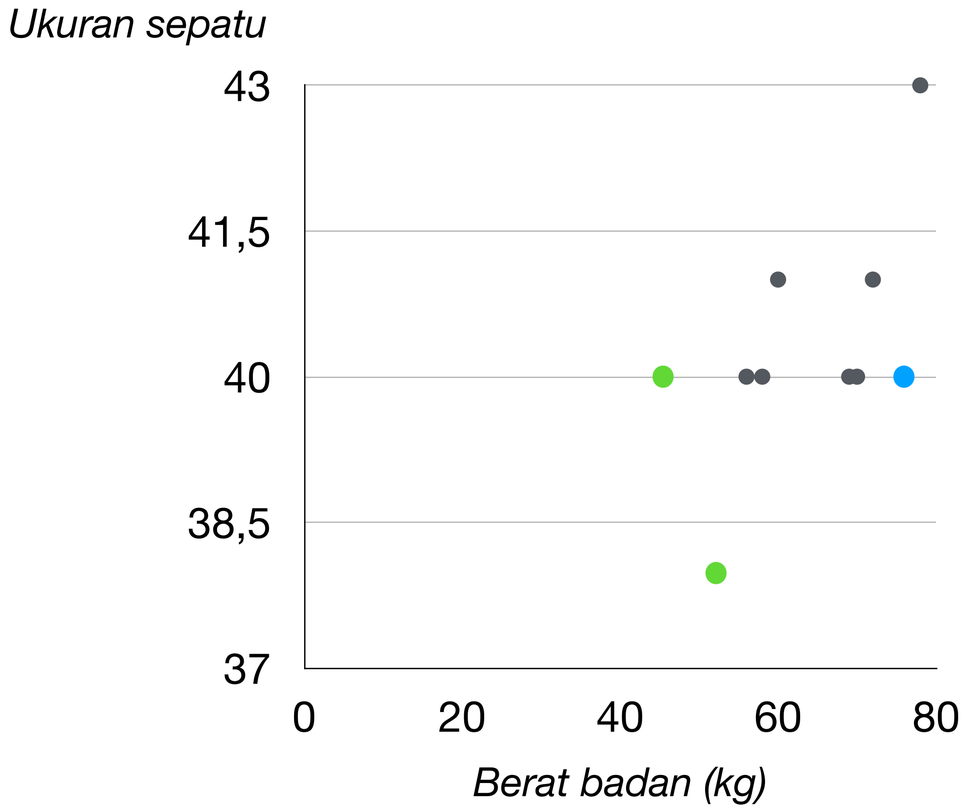
Output:
Data yang sudah dikategorikan

Mengkategorikan data menjadi *k* kategori.
Dimana *k* merupakan {0, 1, …. n}
Nilai *k* tidak harus bernilai ganjil

Nilai *k* merupakan **hyperparameter**

Tommy (tommy.winarta@inergi.id)

k-Means Clustering



Langkah-langkah:

1. Buat *k* titik sebagai *centroid* (nilai tengah dari suatu kelompok)
2. Untuk masing-masing titik, cari nilai jarak dari titik tersebut ke *k* centroid
3. Tentukan centroid yang paling terdekat. Maka, titik tersebut masuk ke kelompok tersebut.

Input:
Data yang sudah ada

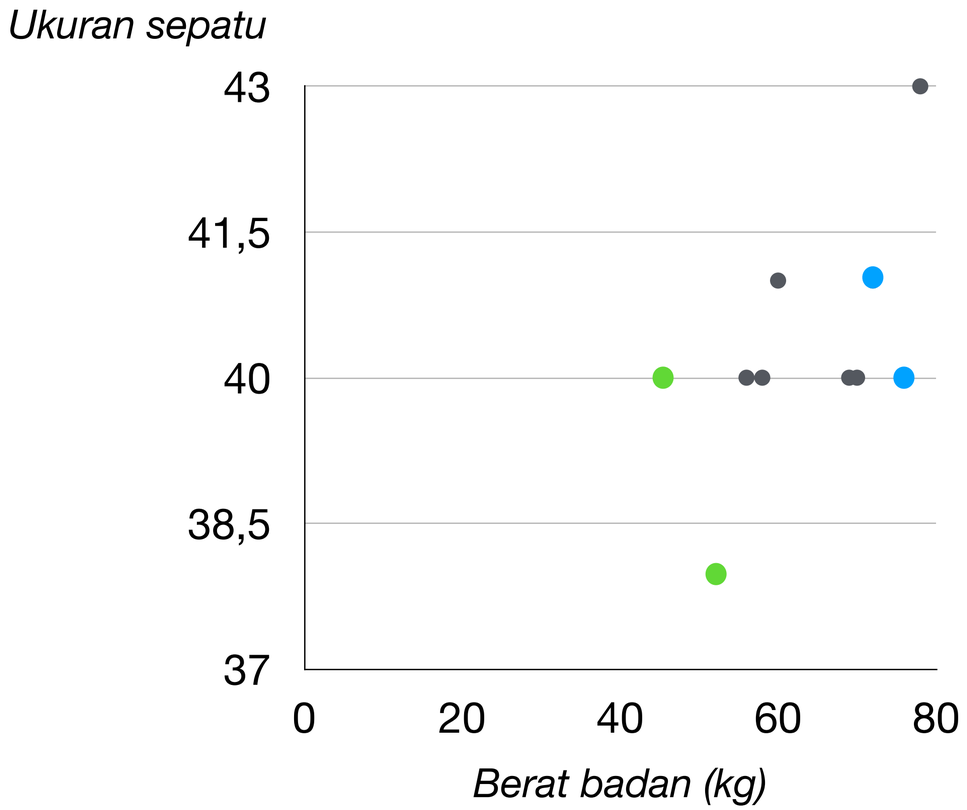
Output:
Data yang sudah dikategorikan

Mengkategorikan data menjadi *k* kategori.
Dimana *k* merupakan {0, 1, …. n}
Nilai *k* tidak harus bernilai ganjil

Nilai *k* merupakan **hyperparameter**

Tommy (tommy.winarta@inergi.id)

k-Means Clustering



Langkah-langkah:

1. Buat *k* titik sebagai *centroid* (nilai tengah dari suatu kelompok)
2. Untuk masing-masing titik, cari nilai jarak dari titik tersebut ke *k* centroid
3. Tentukan centroid yang paling terdekat. Maka, titik tersebut masuk ke kelompok tersebut.

Input:
Data yang sudah ada

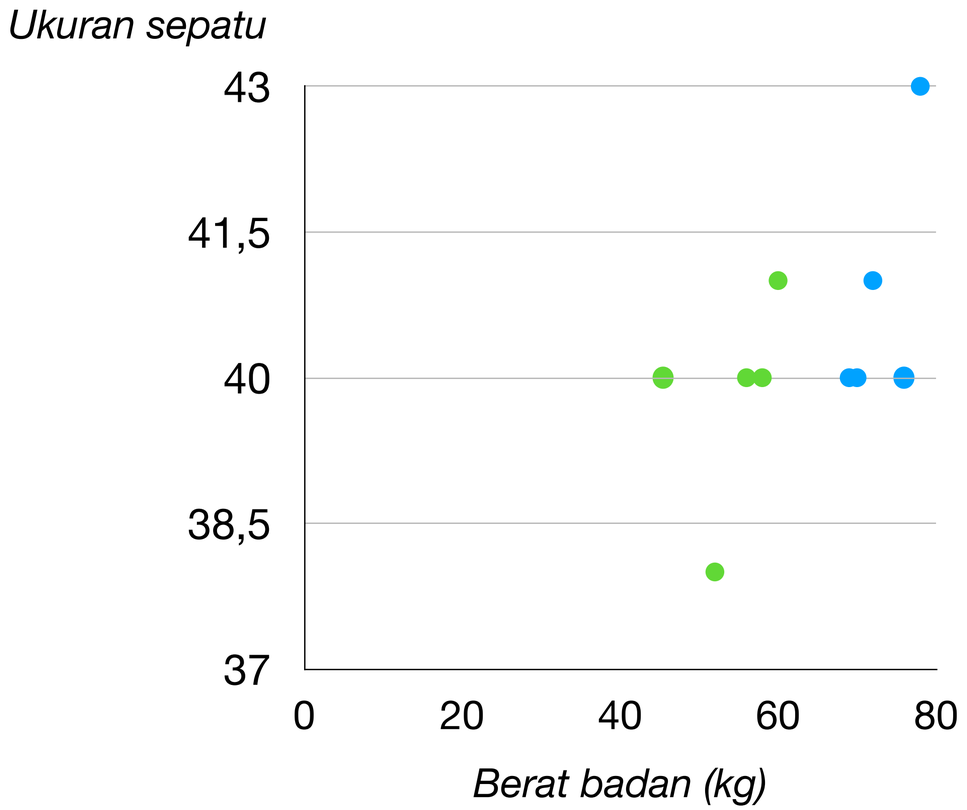
Output:
Data yang sudah dikategorikan

Mengkategorikan data menjadi *k* kategori.
Dimana *k* merupakan {0, 1, …. n}
Nilai *k* tidak harus bernilai ganjil

Nilai *k* merupakan **hyperparameter**

Tommy (tommy.winarta@inergi.id)

k-Means Clustering



Langkah-langkah:

1. Buat *k* titik sebagai *centroid* (nilai tengah dari suatu kelompok)
2. Untuk masing-masing titik, cari nilai jarak dari titik tersebut ke *k* centroid
3. Tentukan centroid yang paling terdekat. Maka, titik tersebut masuk ke kelompok tersebut.

Input:
Data yang sudah ada

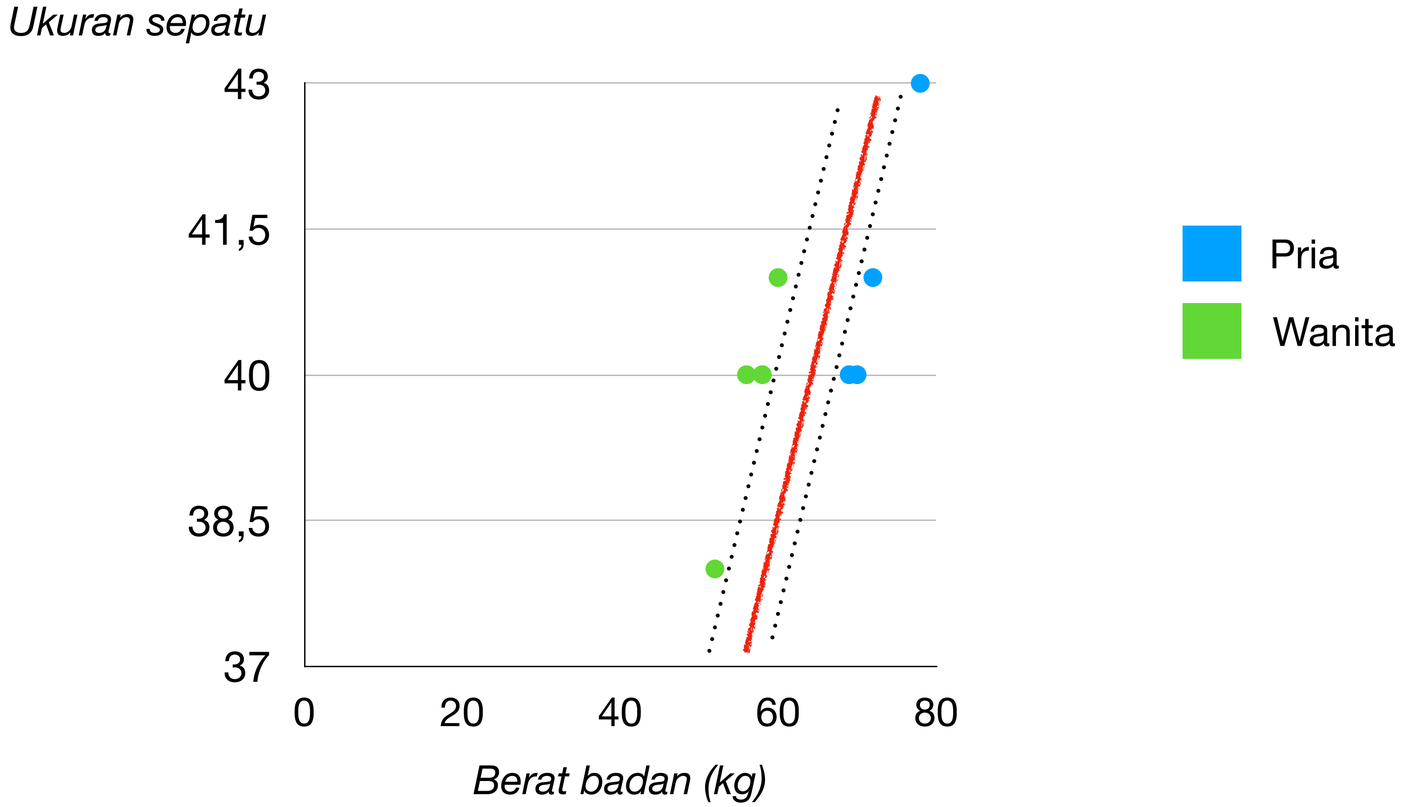
Output:
Data yang sudah dikategorikan

Mengkategorikan data menjadi *k* kategori.
Dimana *k* merupakan {0, 1, …. n}
Nilai *k* tidak harus bernilai ganjil

Nilai *k* merupakan **hyperparameter**

Tommy (tommy.winarta@inergi.id)

Support Vector Machine (SVM)



Input:
Apabila ada titik baru, misal:

- Berat badan: 61 kg
- Ukuran sepatu: 38

Output:
Pria/wanita?

Klasifikasi

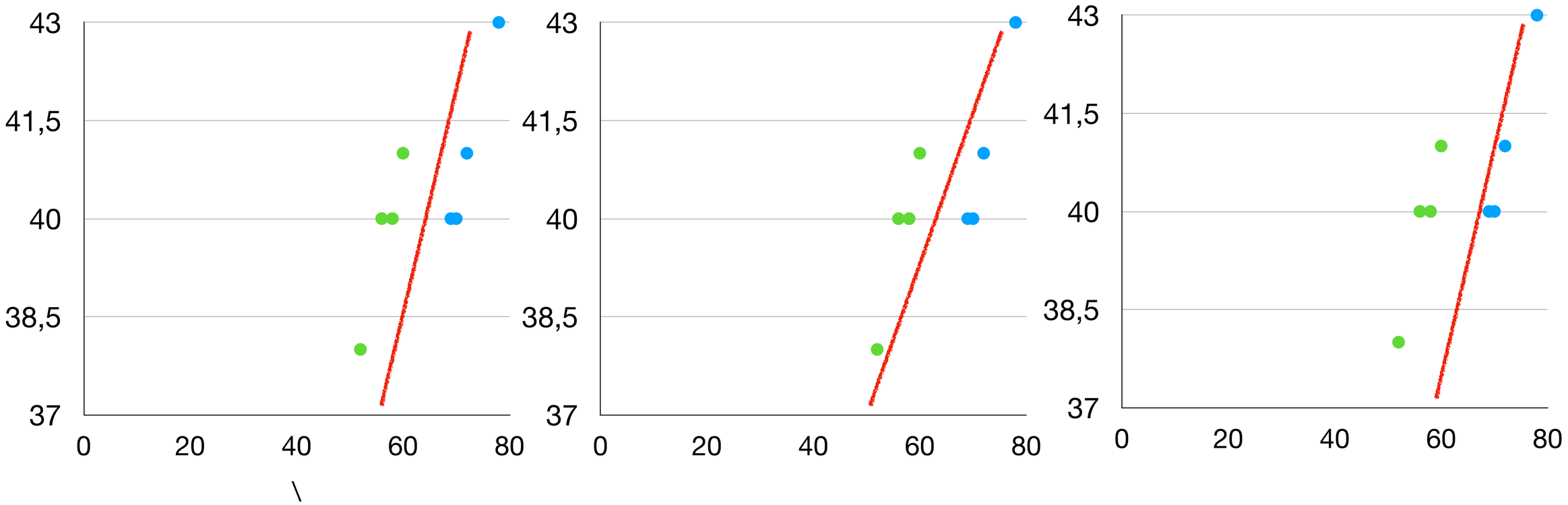
SVM adalah algoritma yang dapat menentukan suatu **hyperplane** yang dapat membatasi semua *class* yang ada secara optimal

Tommy (tommy.winarta@inergi.id)

Support Vector Machine (SVM)

SVM adalah algoritma yang dapat menentukan suatu garis yang dapat membatasi *class* secara optimal

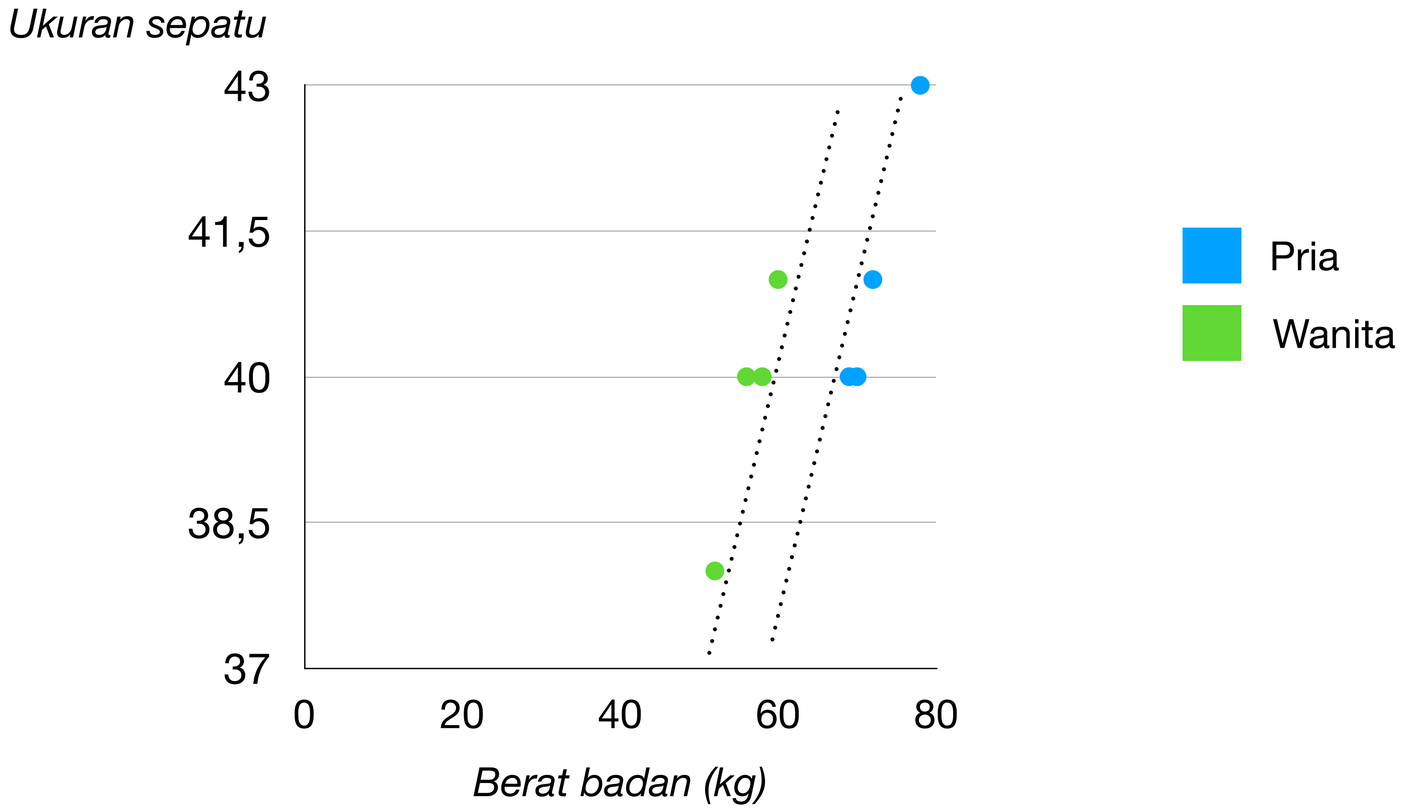
Terdapat banyak kemungkinan **hyperplane**:



\

Tommy (tommy.winarta@inergi.id)

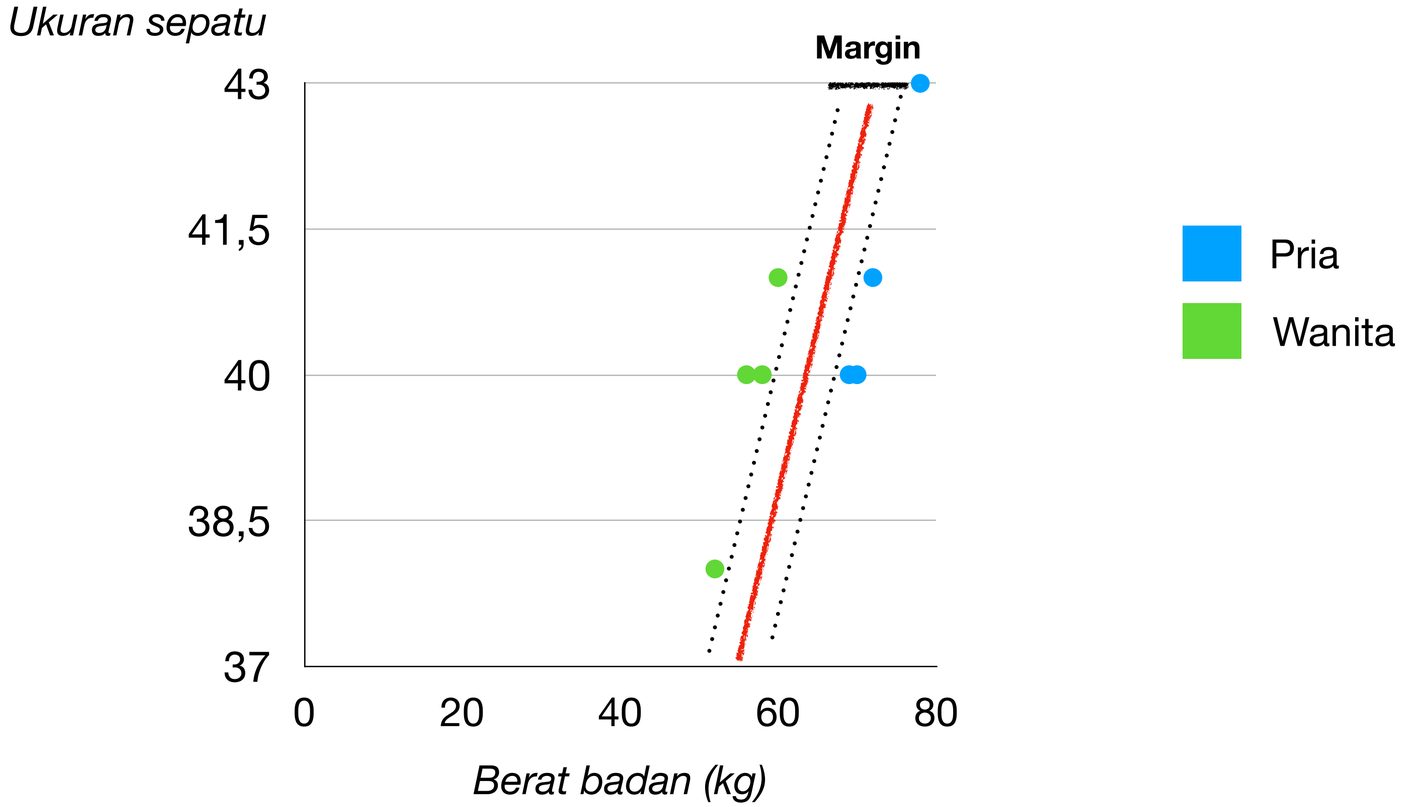
Support Vector Machine (SVM)



Mencari **bagian/*plane* ekstrem** dari masing-masing class **(support vectors)**

Tommy (tommy.winarta@inergi.id)

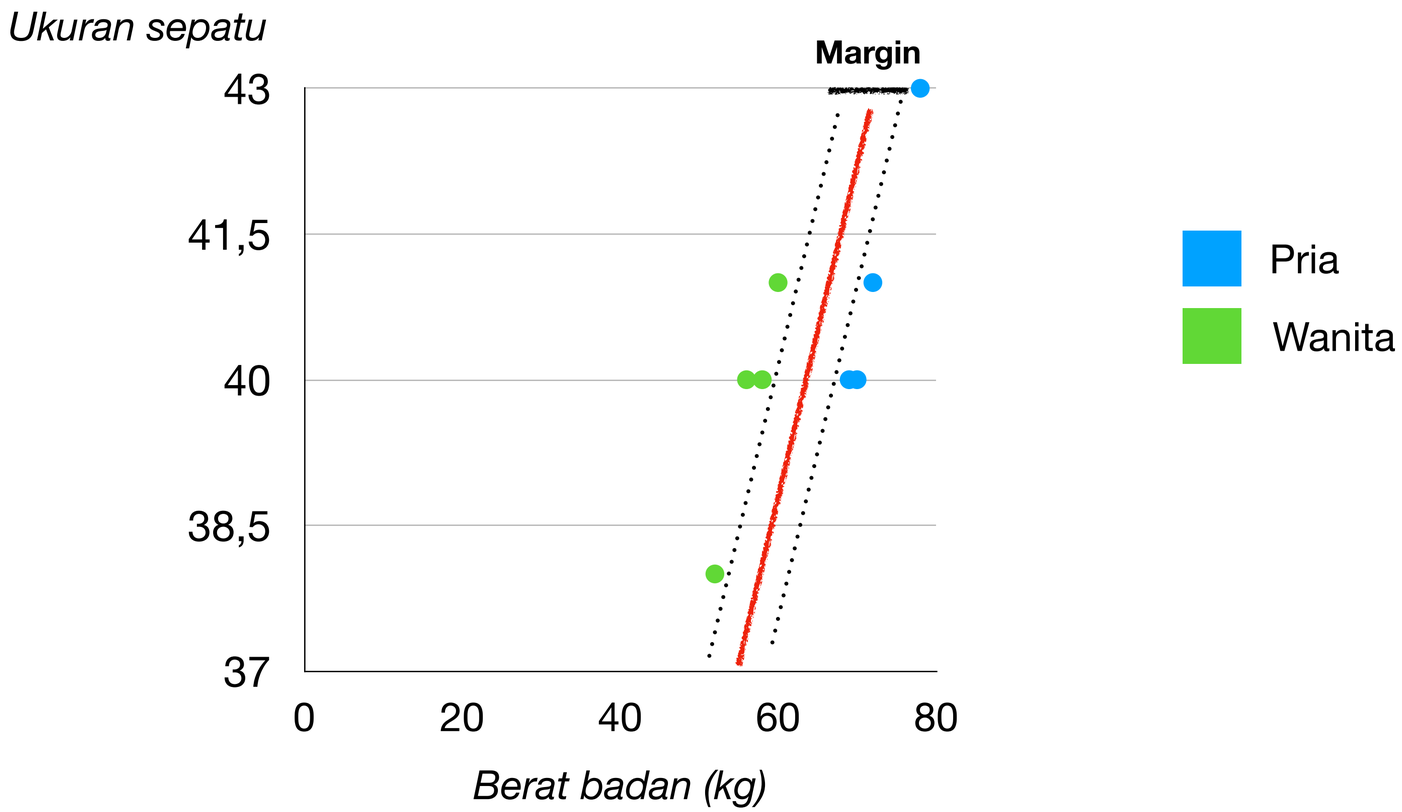
Support Vector Machine (SVM)



Menentukan *plane* yang dapat memisahkan seluruh **support vectors** dengan baik (**hyperplane)**, memastikan semua class memiliki perbedaan yang signifikan satu dengan yang lain (**nilai margin maksimum**).

Tommy (tommy.winarta@inergi.id)

Support Vector Machine (SVM)



Apabila data jauh lebih kompleks (tidak linear seperti grafik di kiri), diperlukan **kernel** untuk mengubah data yang kompleks (berdimensi tinggi) menjadi linear.

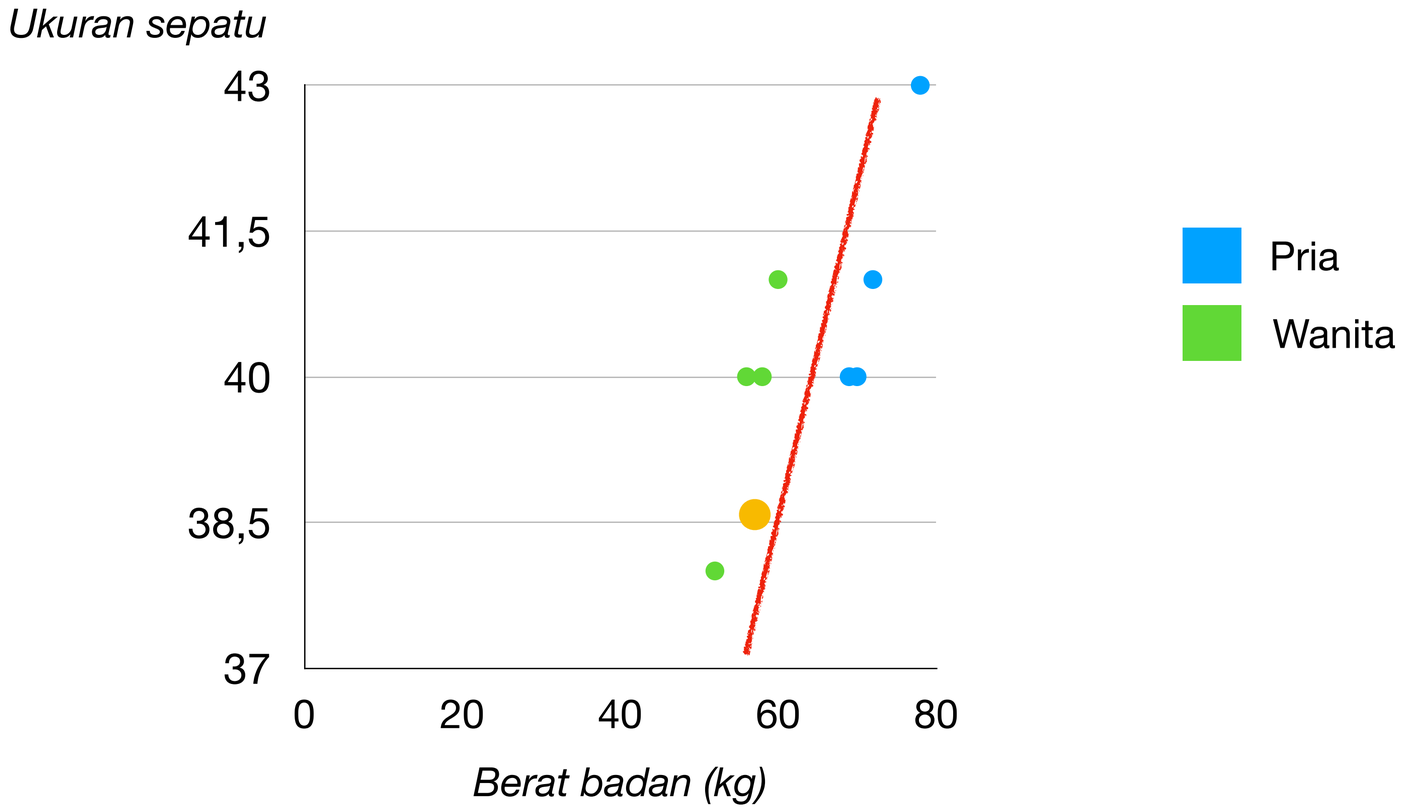
Terdapat beberapa contoh kernel:
1. Polynomial
2. Radial Basis Function (RBF)
3. Sigmoid

Menentukan kernel yang digunakan mempengaruhi hasil training.

Kernel yang dipakai adalah salah satu contoh **hyperparameter**.

Tommy (tommy.winarta@inergi.id)

Support Vector Machine (SVM)



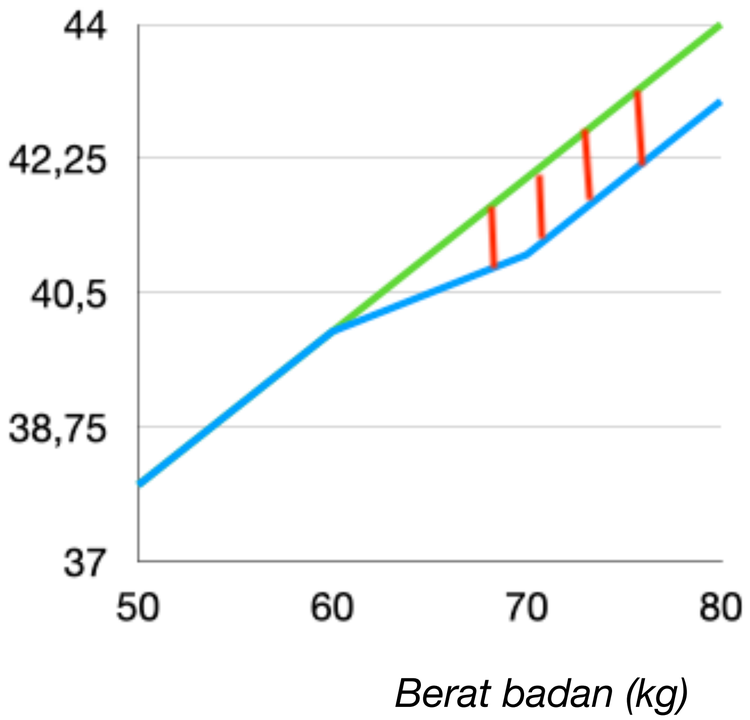
Input:
Apabila ada titik baru, misal:
- Berat badan: 61 kg
- Ukuran sepatu: 38

Output:
Pria/wanita?

Tommy (tommy.winarta@inergi.id)

Linear Regression

Dependent variable (yang dicaritahu)
Ukuran sepatu



Independent variable (yang diketahui)

Menentukan suatu garis linear/lurus untuk memprediksi nilai numerik.

Pertanyaan: Diketahui berat badan, berapakah ukuran sepatunya?

Ekspektasi

Error

Prediksi (**regression line**)

Mencari korelasi antara semua variabel yang ada

- Apakah berat badan berbanding lurus dengan ukuran sepatu?
  - Semakin kecil berat badan, ukuran sepatu makin kecil, dan sebaliknya?

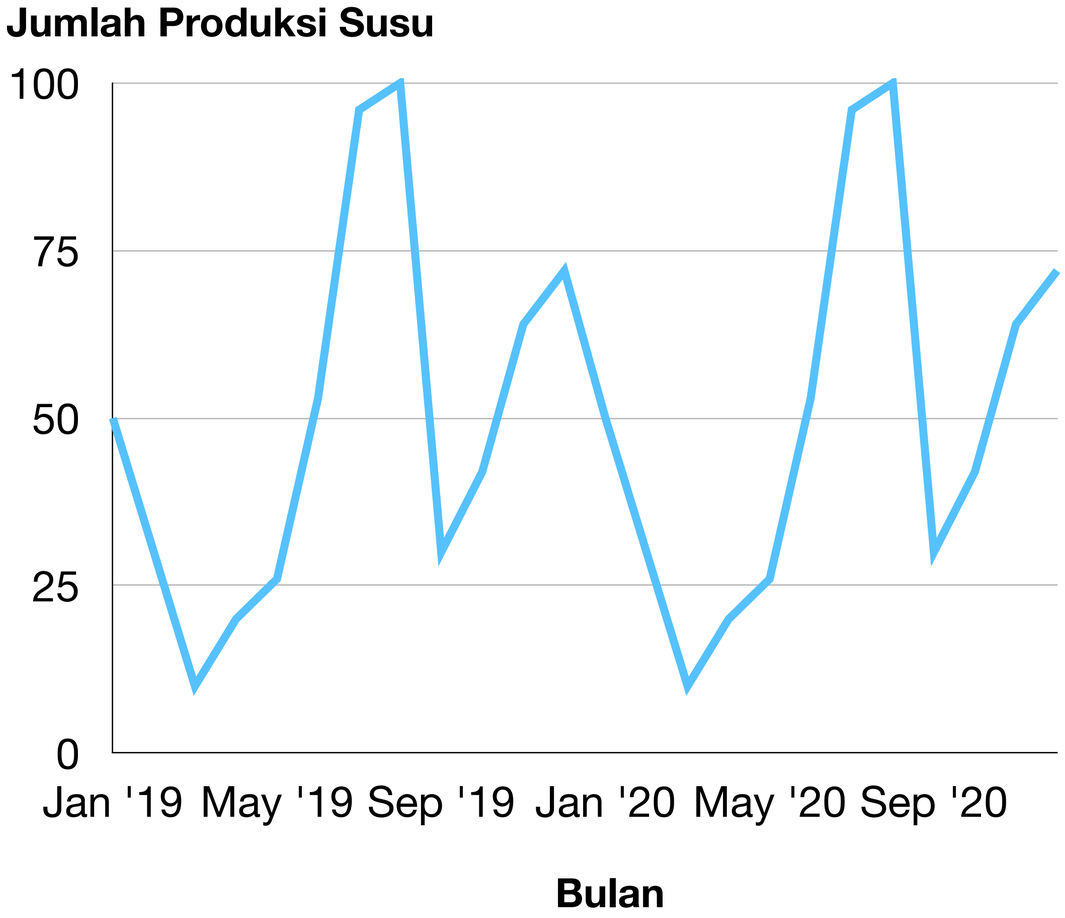
Fungsi regression line

$y = mx + c$

Yang dapat mendeskripsikan dataset kita dengan baik (meminimalisir error)

Tommy (tommy.winarta@inergi.id)

Auto Regressive (AR)

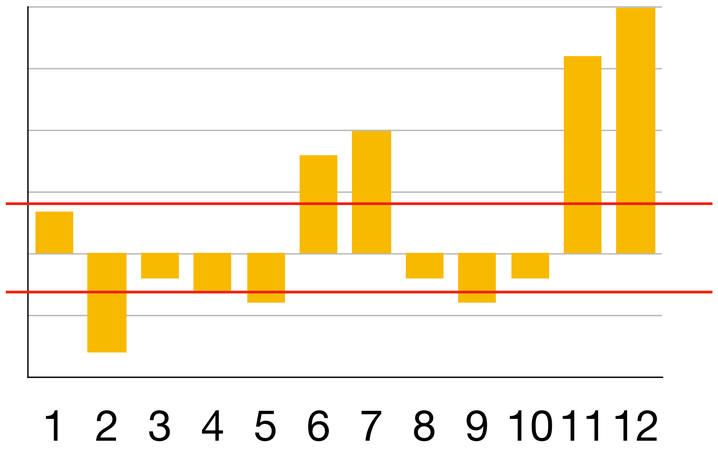


Memprediksi suatu nilai numerik pada data yang bergantung pada suatu rentang waktu.

Pertanyaan: Berapa banyak susu yang harus diproduksi di bulan Januari 2021?

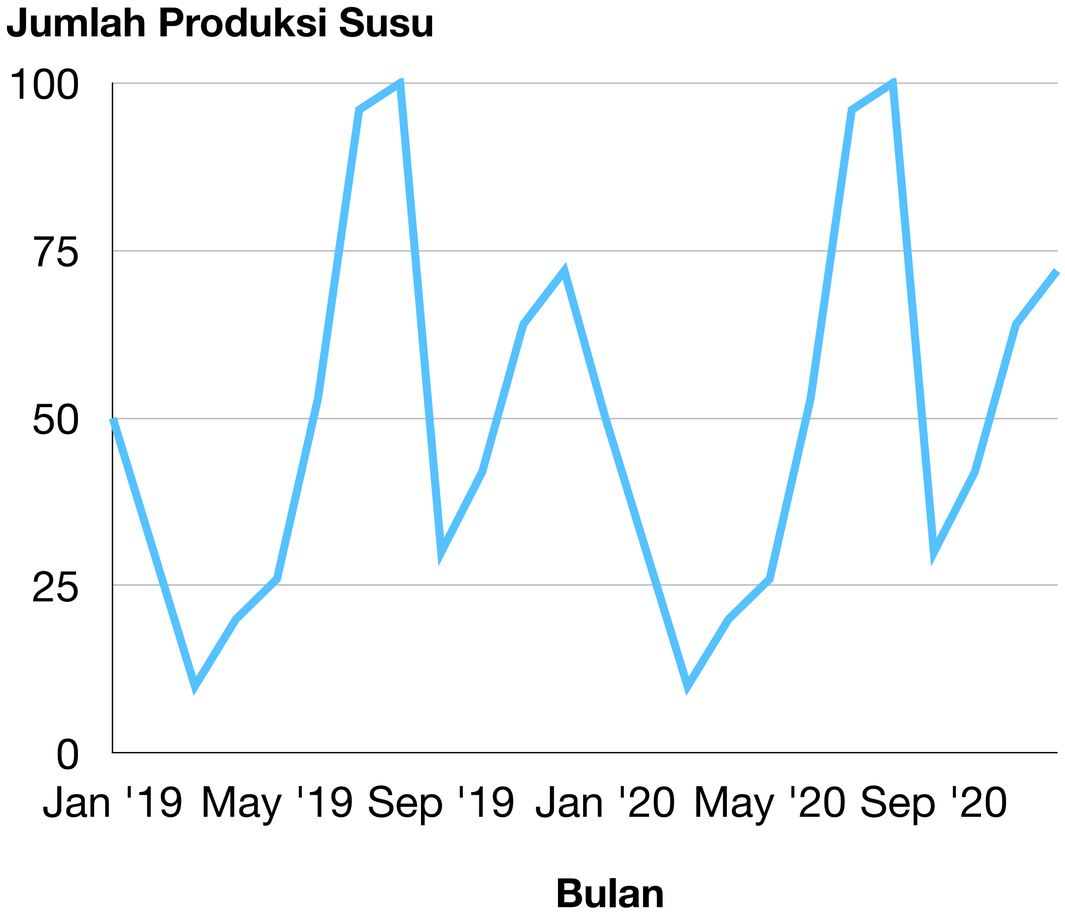
Mencari waktu yang memiliki nilai korelasi tinggi dengan waktu yang dicari (Januari).

Partial Autocorrelation Function (PACF)



Tommy (tommy.winarta@inergi.id)

Auto Regressive (AR)

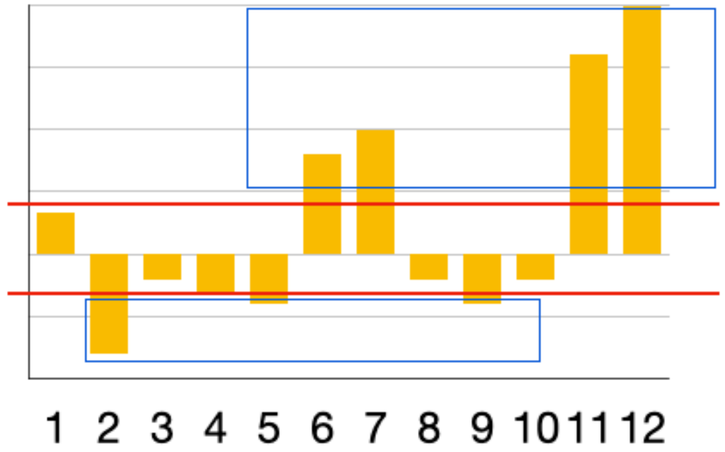


Memprediksi suatu nilai numerik pada data yang bergantung pada suatu rentang waktu.

Pertanyaan: Berapa banyak susu yang harus diproduksi di bulan Januari 2021?

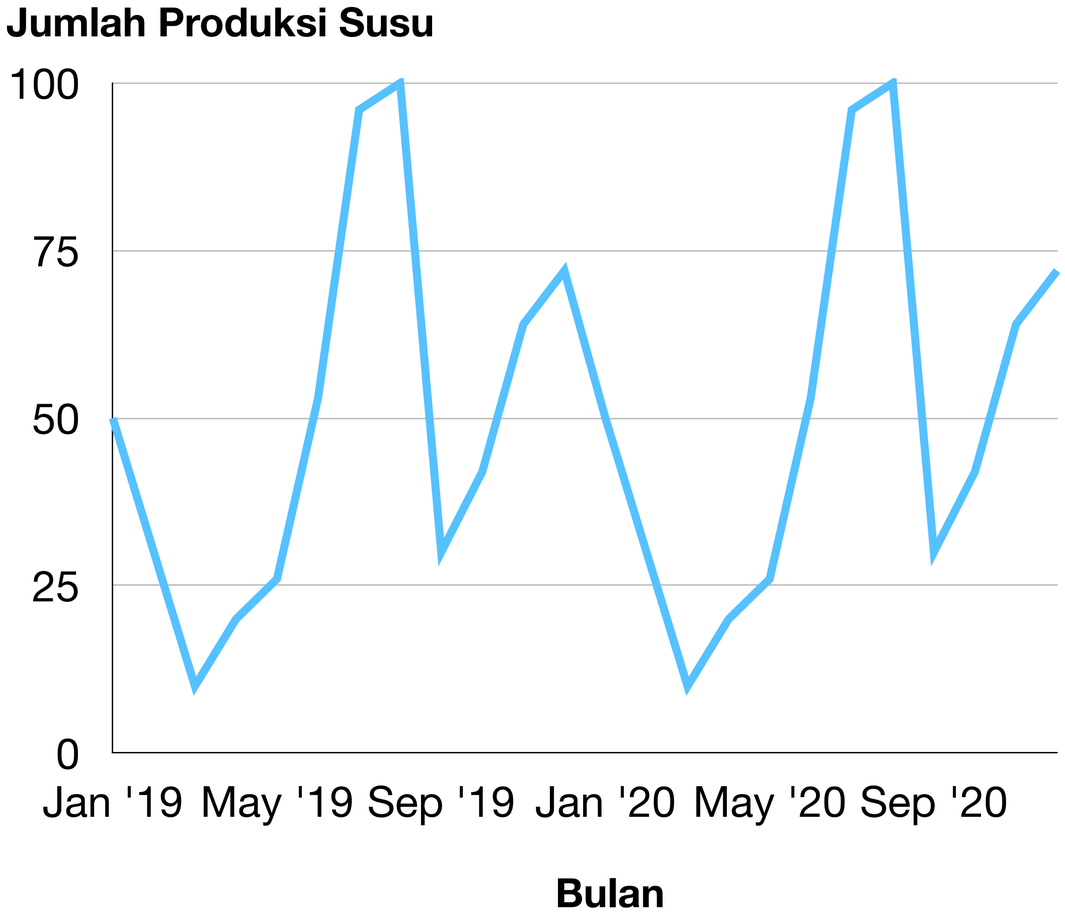
Mencari waktu yang memiliki nilai korelasi tinggi dengan waktu yang dicari (Januari).

Partial Autocorrelation Function (PACF)



Tommy (tommy.winarta@inergi.id)

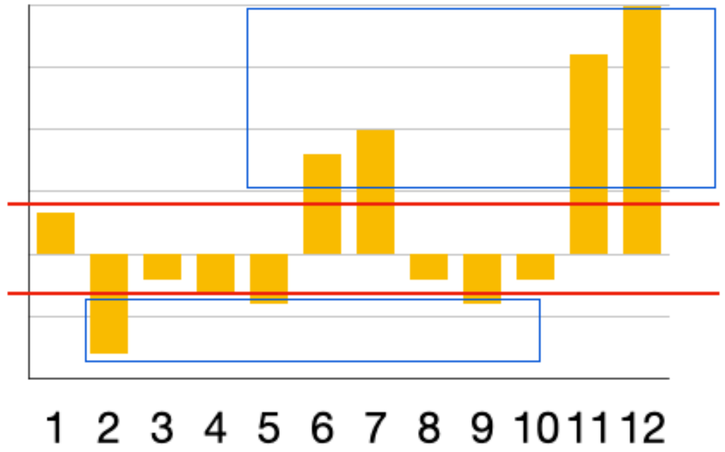
Auto Regressive (AR)



Pertanyaan: Berapa banyak susu yang harus diproduksi di bulan Januari 2021?

Jumlah produksi = Koefisien + (Koefisien bulan 12 * jumlah 12 bulan lalu) + (Koefisien bulan 11 * jumlah 11 bulan lalu) + (Koefisien bulan 9 * jumlah 9 bulan lalu) + …..

Partial Autocorrelation Function (PACF)



Tommy (tommy.winarta@inergi.id)

Catatan

- Algoritma yang dipakai bergantung pada masalah yang ingin diselesaikan & data yang dimiliki (misal, regresi untuk time-series berbeda dengan regresi untuk data numerik pada umumnya)

- Maka, tidak ada algoritma yang pasti efektif untuk semua data/kasus

Tommy (tommy.winarta@inergi.id)

Catatan

- Setiap algoritma memiliki kompleksitas masing-masing yang mempengaruhi:
  - Kecepatan training
  - Nilai accuracy dari model
- Nilai hyperparameter/variabel yang dapat di tune pada algoritma dapat mempengaruhi accuracy dari model yang dihasilkan

Tommy (tommy.winarta@inergi.id)