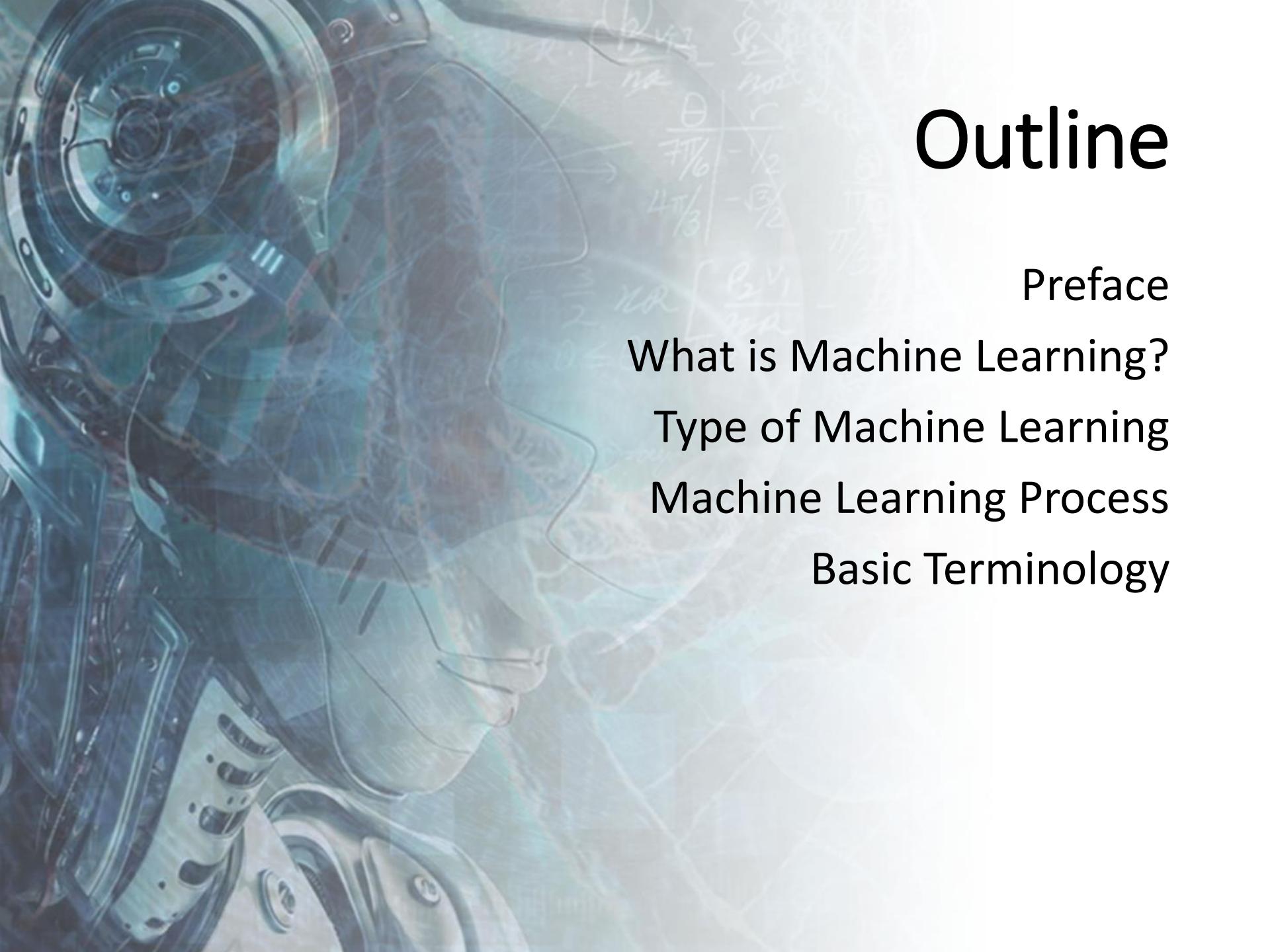


BASIC OF MACHINE LEARNING

Mohamad Irvan Septiar Musti

DATA
SCIENCE
INDONESIA





Outline

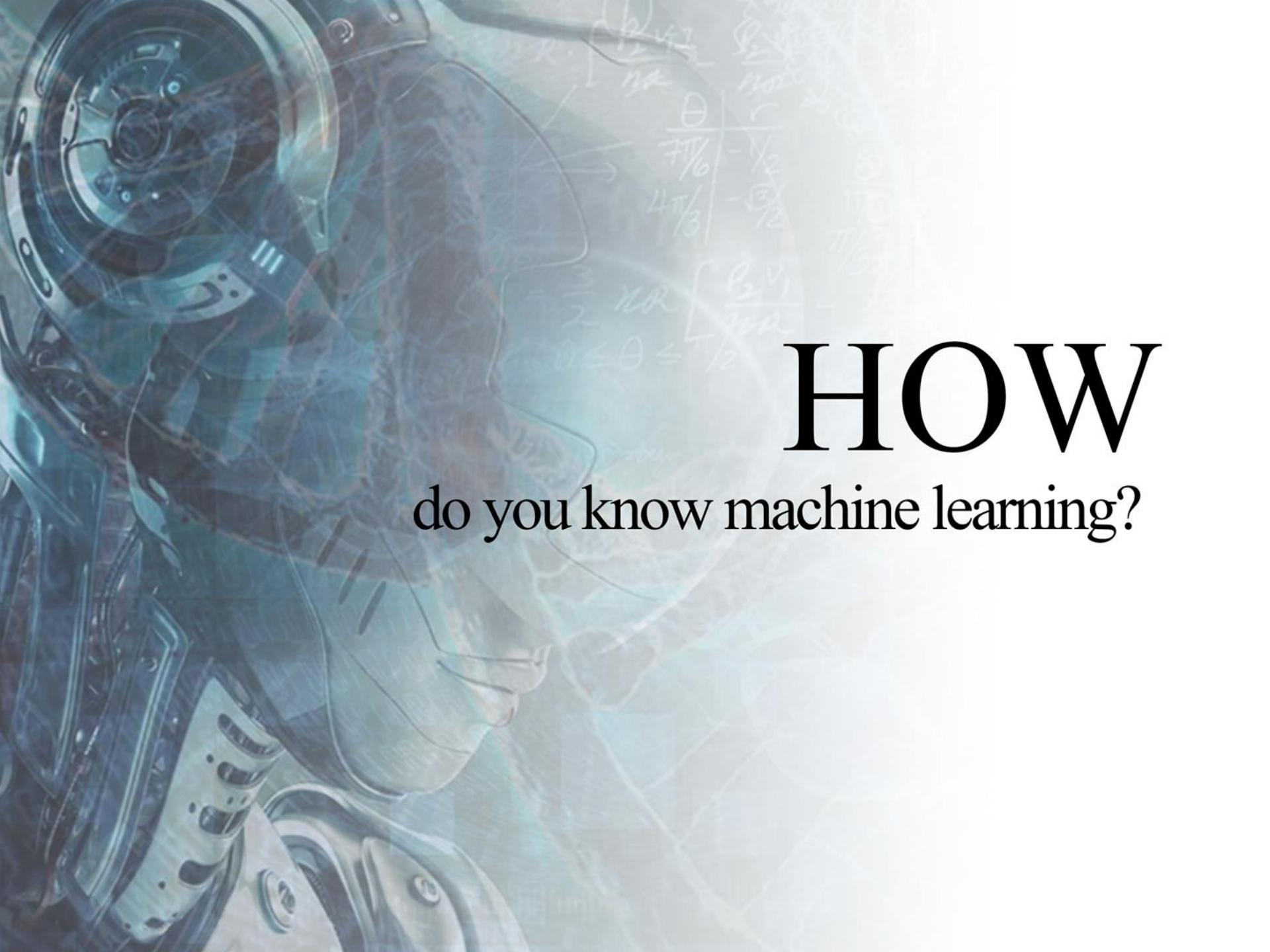
Preface

What is Machine Learning?

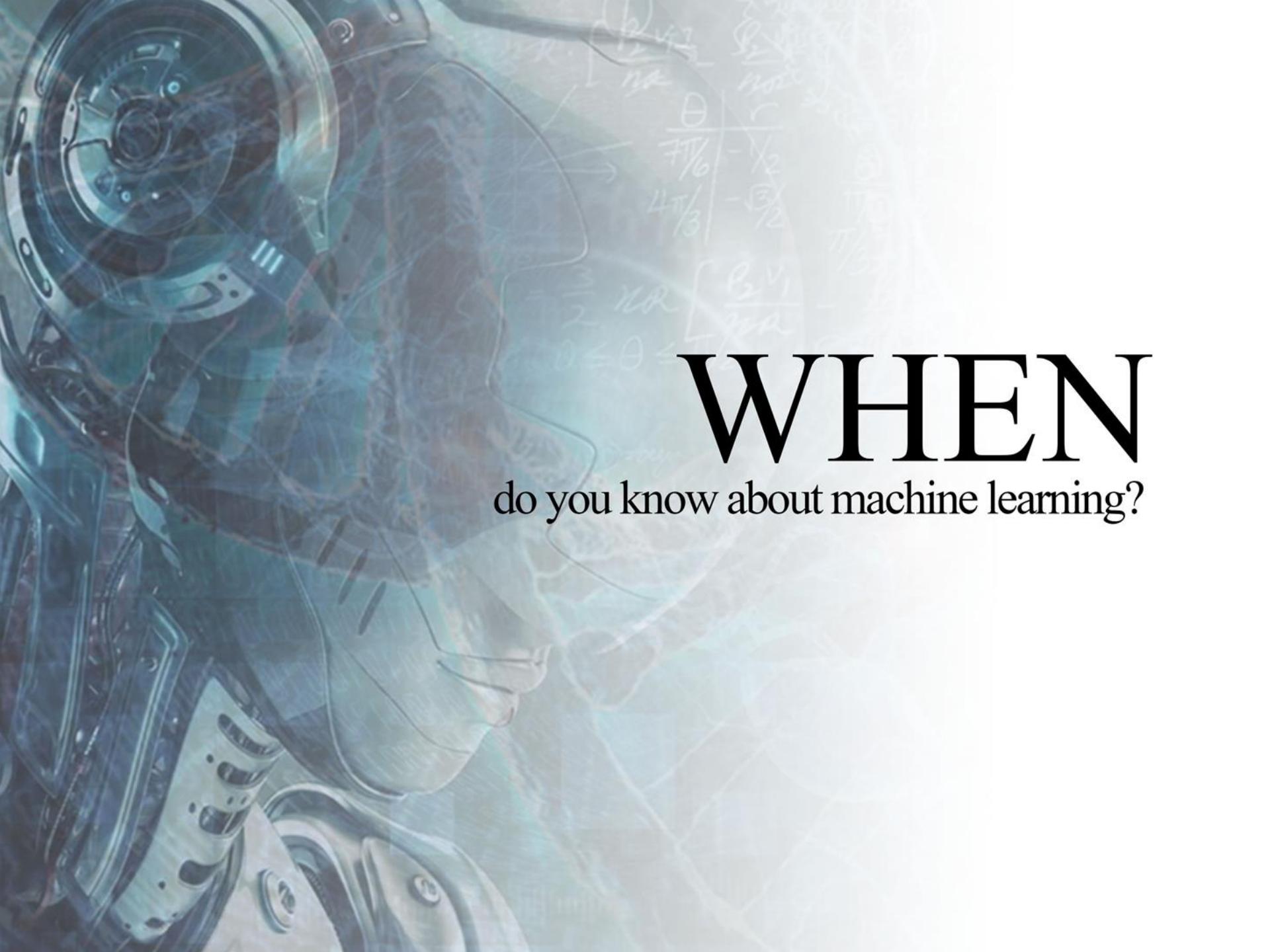
Type of Machine Learning

Machine Learning Process

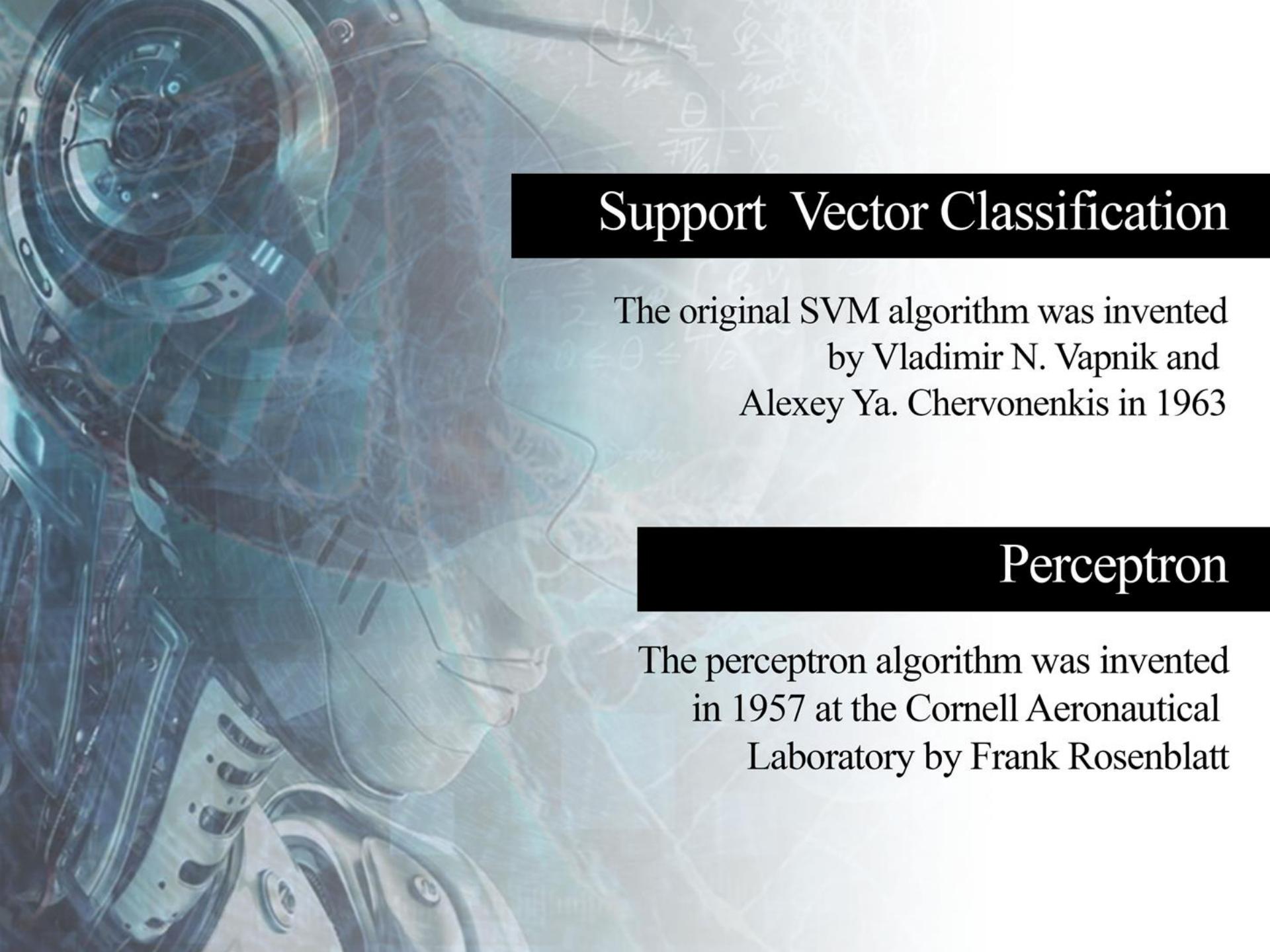
Basic Terminology



HOW
do you know machine learning?



WHEN
do you know about machine learning?



Support Vector Classification

The original SVM algorithm was invented
by Vladimir N. Vapnik and
Alexey Ya. Chervonenkis in 1963

Perceptron

The perceptron algorithm was invented
in 1957 at the Cornell Aeronautical
Laboratory by Frank Rosenblatt

TANTANGAN BIG DATA



Data



Wawasan



Keputusan

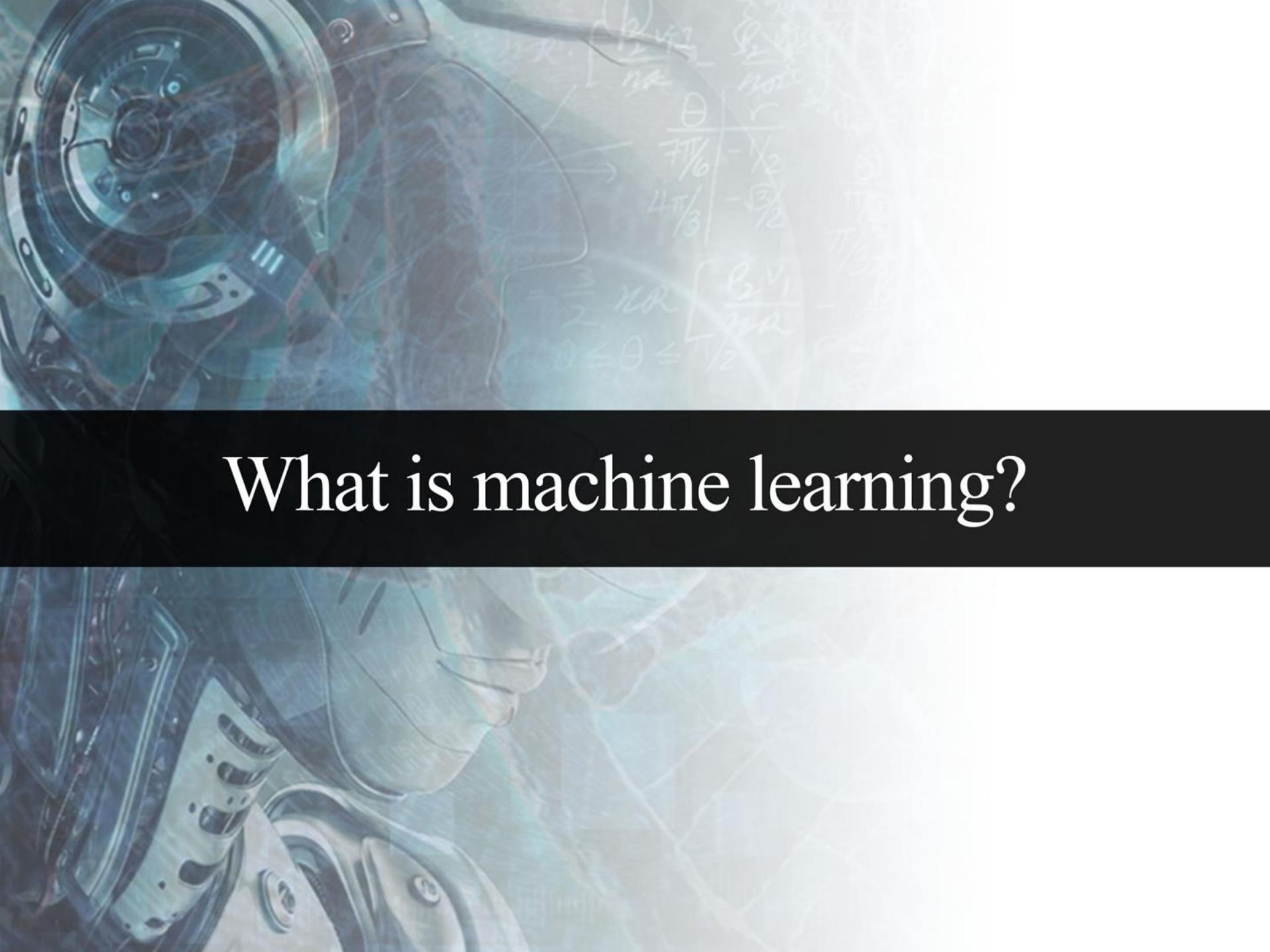
Pertumbuhan data yang sangat besar dan cepat membuat analisis konvensional atau statistika biasa yang sering digunakan sebagai alat untuk memperoleh wawasan dari data semakin sulit digunakan atau bahkan tidak mungkin dilakukan, sehingga dibutuhkan metode baru untuk menganalisis data tersebut



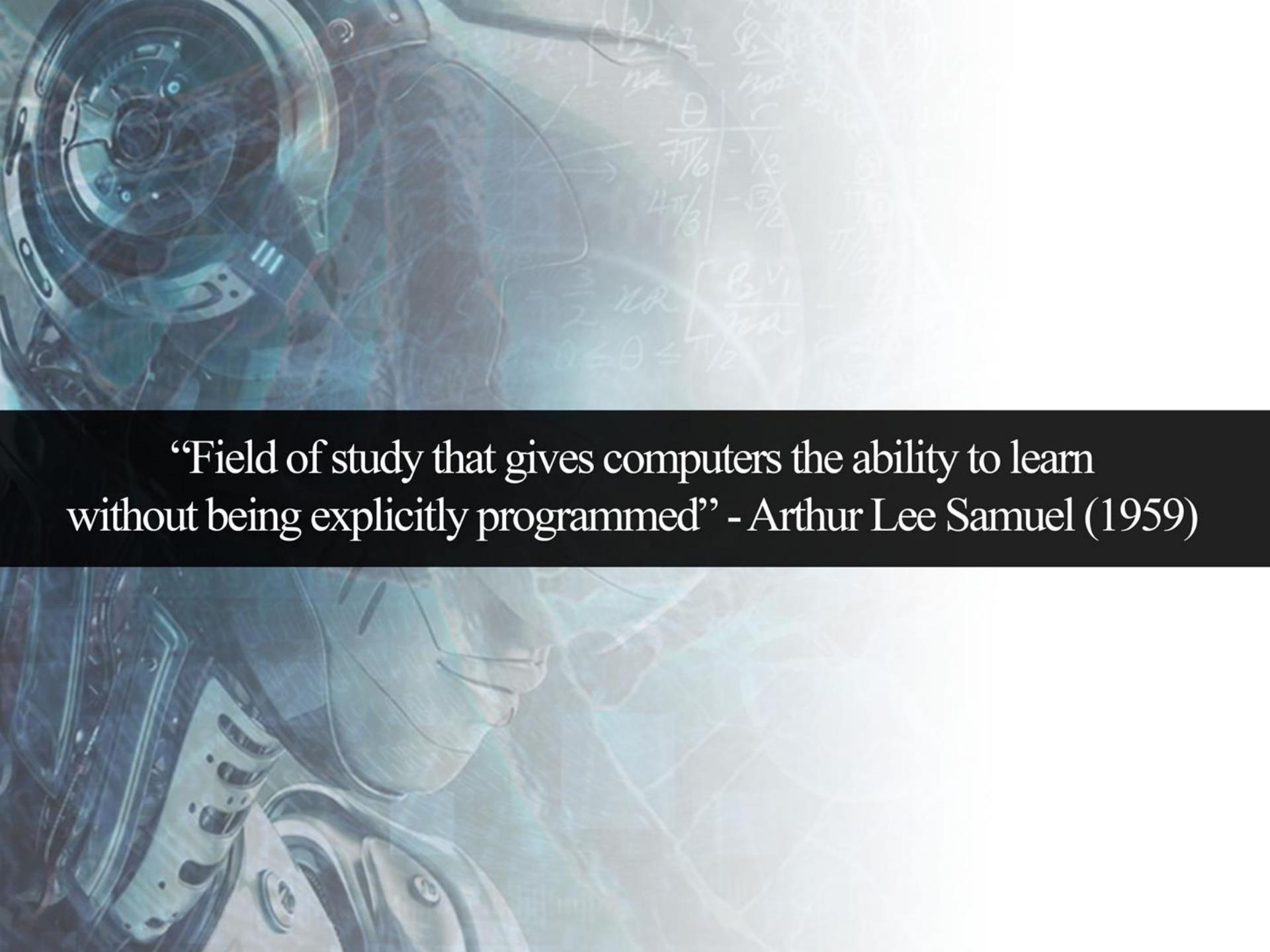
Business Needs

Big Data Phenomenon

Information Technology Development



What is machine learning?



“Field of study that gives computers the ability to learn without being explicitly programmed” - Arthur Lee Samuel (1959)



Essential Kit : Don't be Allergic!

Mathematics
Statistics
Algorithm
Programming
Domain Knowledge



Applicable Field

Bioinformatics
Financial Processing
Episode Mining
Music Retrieval
Cyber Security
Text Mining



Supervised Learning

Unsupervised Learning

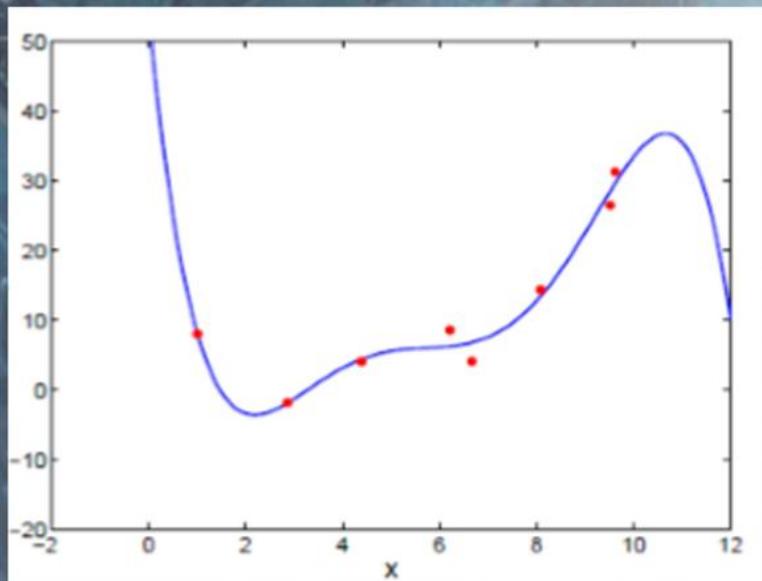
Supervised Learning

Pada supervised learning, data pelatihan disertai nilai target, sehingga masing-masing data pelatihan berupa pasangan data. Misalkan $\{x_i, t_i\}, i = 1 \text{ s/d } N$

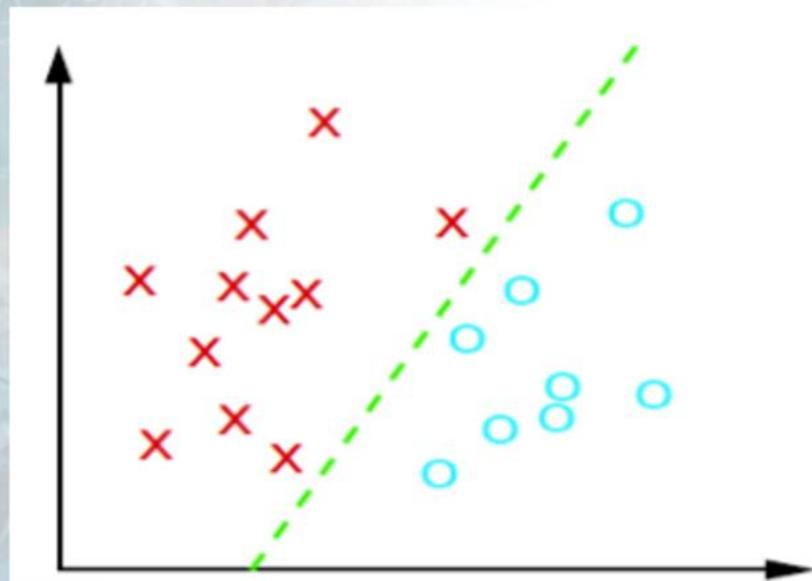
Tujuan supervised learning adalah membangun suatu algoritma atau model yang dapat menghasilkan output yang paling sesuai dengan nilai target (t_i) untuk semua data pelatihan (x_i)

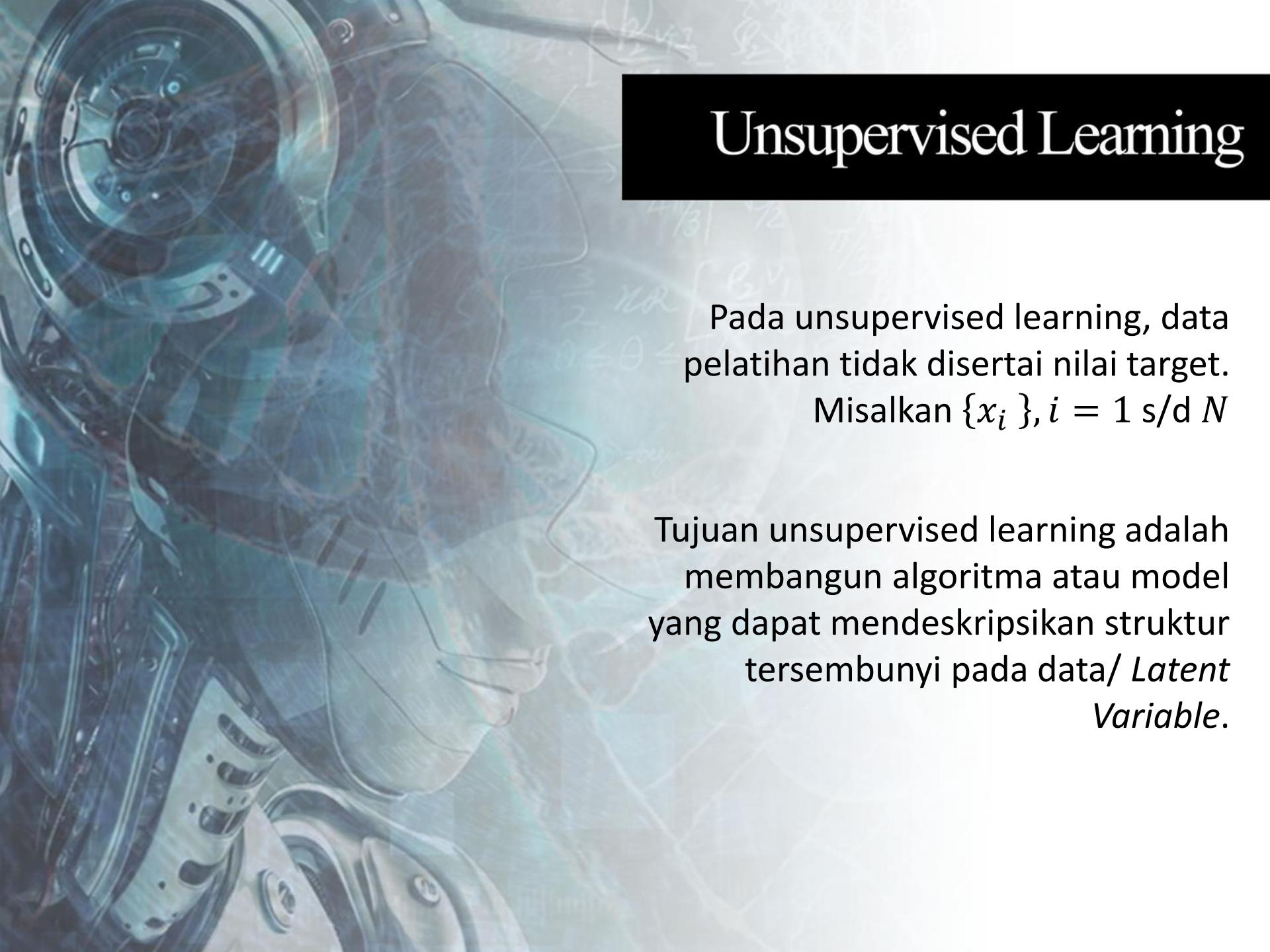
Supervised Learning

Regresi



Klasifikasi





Unsupervised Learning

Pada unsupervised learning, data pelatihan tidak disertai nilai target.
Misalkan $\{x_i\}, i = 1 \text{ s/d } N$

Tujuan unsupervised learning adalah membangun algoritma atau model yang dapat mendeskripsikan struktur tersembunyi pada data/ *Latent Variable*.



Unsupervised Learning

Clustering

Dimension Reduction

Dimension Reduction : SVD

$$A \\ n \times d$$

=

$$\hat{U} \\ n \times r$$

$$\hat{\Sigma} \\ r \times r$$

$$\hat{V}^T \\ r \times d$$

$$U$$

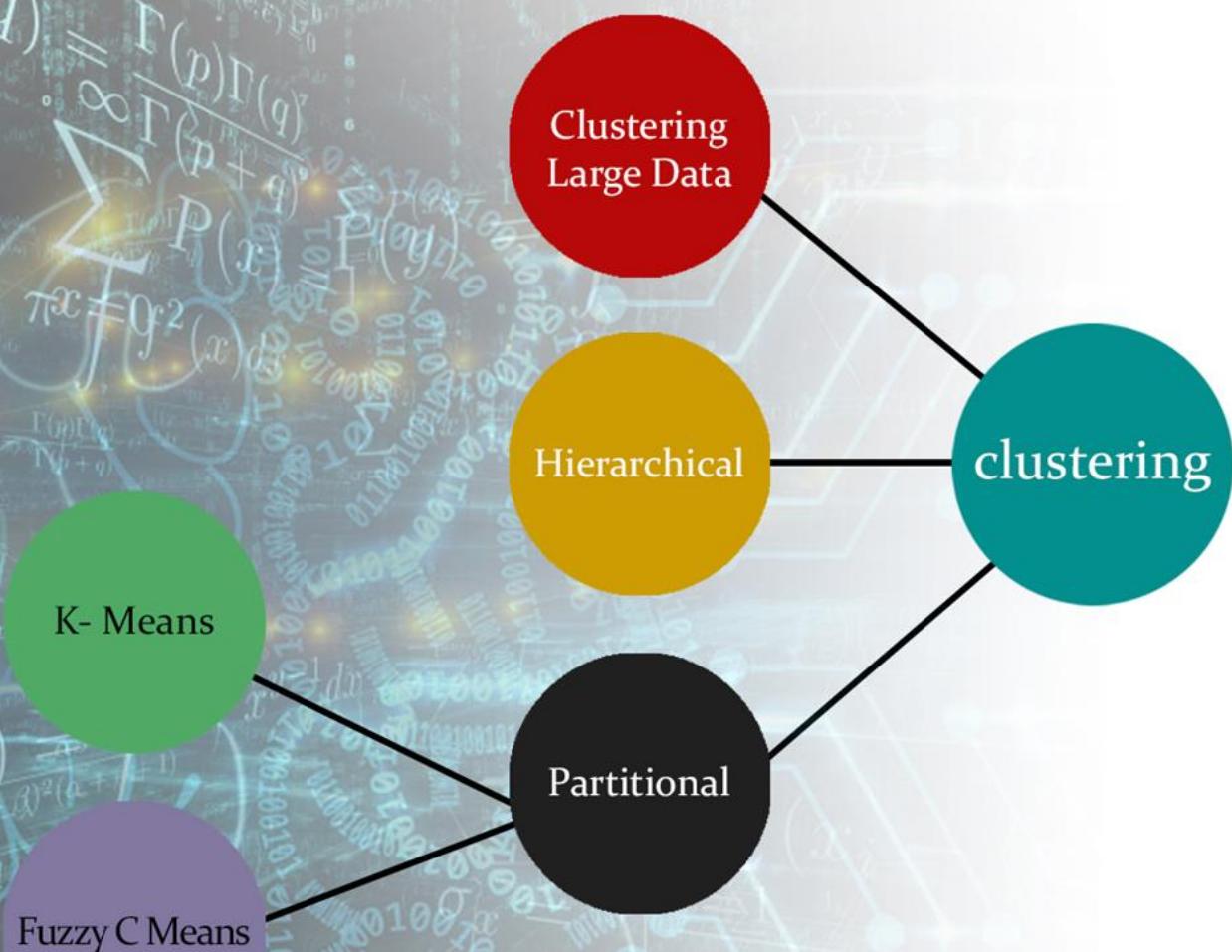
$$\Sigma$$

$$V^T$$

$$n \times d$$

$$n \times d$$

$$d \times d$$





K-Means Algorithm

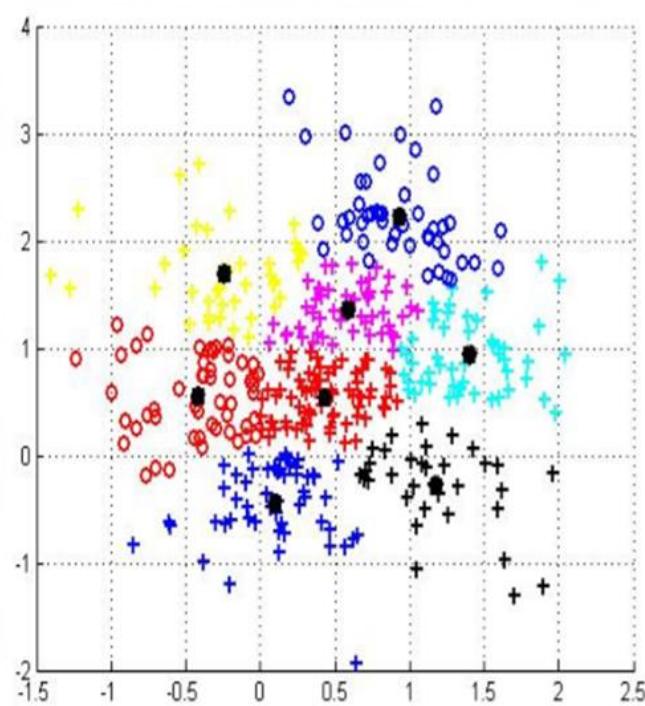
K-Means adalah salah satu metode clustering berbasis centroid

Metode k-means merepresentasikan setiap kelompok dengan suatu pusat kelompok atau centroid

Pertama-tama, metode ini akan menentukan jumlah kelompok dan centroid berdasarkan data yang ada. Selanjutnya, suatu data akan menjadi anggota suatu kelompok jika jarak antara data dan centroid dari kelompok tersebut adalah terkecil

K-Means Algorithm

Formulasi Masalah



Diberikan himpunan data observasi tak berlabel.

Permasalahan : Bagaimana mengelompokkan data tersebut sedemikian sehingga data dalam suatu kelompok adalah lebih mirip dibandingkan dengan data dalam kelompok lainnya

K-Means Algorithm

Diberikan himpunan data $x_n, n = 1 \text{ s/d } N$

Misal terdapat K kelompok dengan centroid adalah
 $\mu_k, k = 1 \text{ s/d } K$

Setiap data x_n memiliki suatu variabel indikator
 $r_{nk} \in \{0,1\}, \sum_k r_{nk} = 1$

$r_{nk} = 1$ berarti bahwa data x_n dikelompokkan ke klaster k

$r_{nk} = 0$ berarti bahwa data x_n tidak dikelompokkan ke klaster k

K-Means Algorithm

Selanjutnya, jumlah jarak masing-masing data ke centroid dapat dinyatakan sebagai:

$$J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \|x_n - \mu_k\|^2$$

Sehingga, penyelesaian masalah clustering dengan menggunakan metode k-means menjadi:

$$\min_{\mu_k, r_{nk}} J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \|x_n - \mu_k\|^2$$

K-Means Algorithm

Permasalahan diatas dapat diselesaikan secara iteratif, yaitu:

Inisialisasi μ_k

Lakukan dua langkah berikut sampai r_{nk} dan μ_k konvergen

- a. Menentukan r_{nk} yang meminimumkan J dengan μ_k konstan
- b. Menentukan μ_k yang meminimumkan J dengan r_{nk} konstan

K-Means Algorithm

Algorithm

1. Inisialisasi μ_k , misal secara acak
2. Hitung r_{nk} , yaitu:

$$r_{nk} = \begin{cases} 1, & \text{jika } k = \min_j \|x_n - \mu_k\|^2 \\ 0, & \text{untuk lainnya} \end{cases}$$

3. Hitung μ_k , yaitu:

$$\frac{\partial J}{\partial \mu_k} \rightarrow \mu_k = \frac{\sum_{n=1}^N r_{nk} x_n}{\sum_{n=1}^N r_{nk}}$$

3. Evaluasi konvergensi parameter μ_k . Jika kriteria konvergensi belum terpenuhi, maka kembali ke langkah 2

K-Means Algorithm

Contoh

Diberikan data $x = \{1, 2, 3, 6, 7, 8\}$, jika data tersebut dibuat menjadi dua kluster, tentukan pusat dan anggota dari masing-masing kluster tersebut!

K-Means Algorithm

Contoh

- Misal, inisialisasi $\mu_1 = 3$ dan $\mu_2 = 6$

Iterasi 1:

$$r_{11} = 1, r_{21} = 1, r_{31} = 1, r_{41} = 0, r_{51} = 0, r_{61} = 0 \rightarrow \mu_1 = 2$$

$$r_{12} = 1, r_{22} = 1, r_{32} = 1, r_{42} = 0, r_{52} = 0, r_{62} = 0 \rightarrow \mu_2 = 7$$

Iterasi 2:

$$r_{11} = 1, r_{21} = 1, r_{31} = 1, r_{41} = 0, r_{51} = 0, r_{61} = 0 \rightarrow \mu_1 = 2$$

$$r_{12} = 1, r_{22} = 1, r_{32} = 1, r_{42} = 0, r_{52} = 0, r_{62} = 0 \rightarrow \mu_2 = 7$$

Karena μ_1 dan μ_2 konvergen, maka selesai

x	$\mu_1=3$	$\mu_2=6$
1	1	0
2	1	0
3	1	0
6	0	1
7	0	1
8	0	1
$\mu_1=2$		$\mu_2=7$



Text Mining

Sentiment Analysis

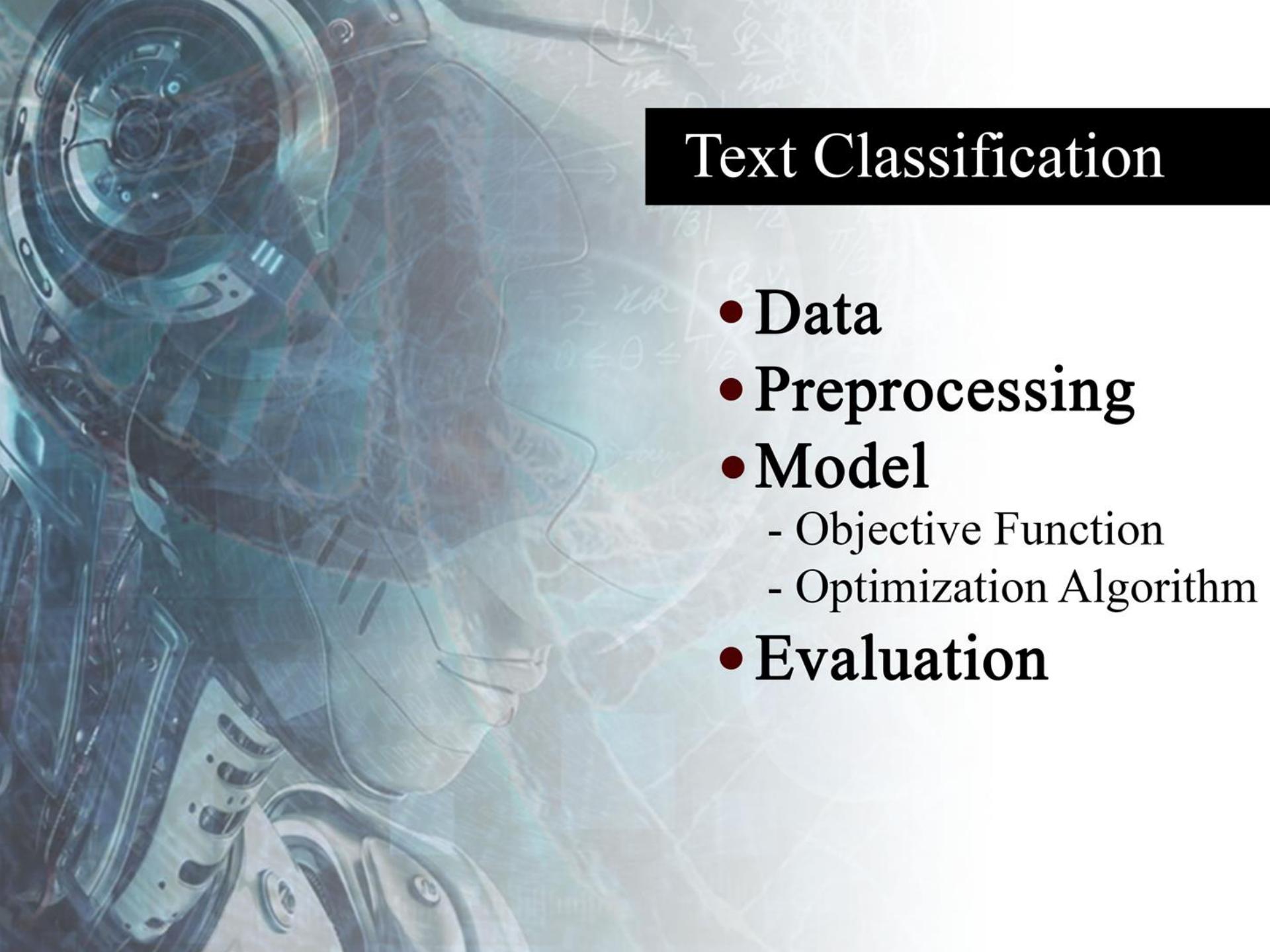
Document Classification

Supervised Learning

Topic Modelling

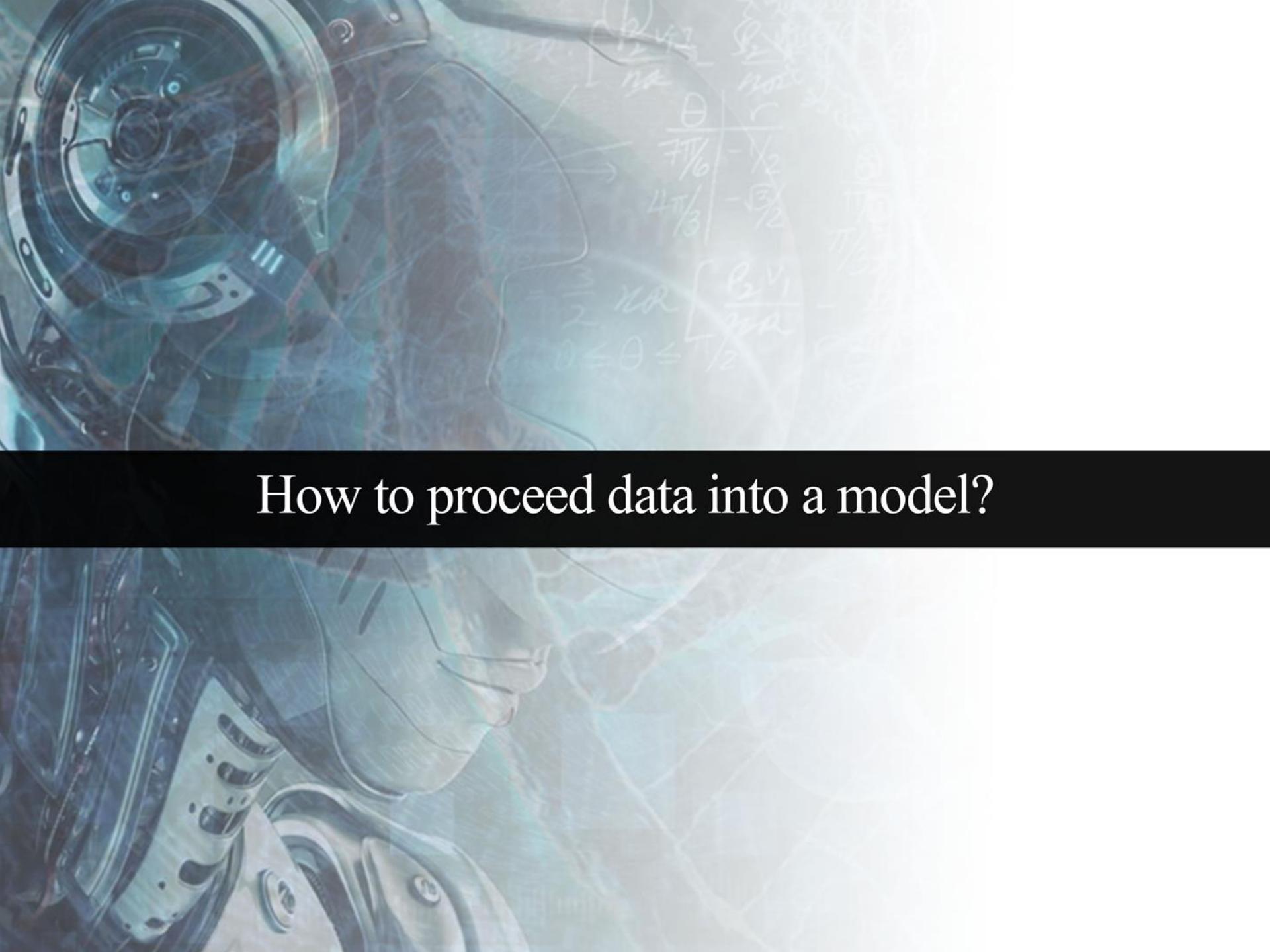
Unsupervised Learning

Text Classification



Text Classification

- Data
- Preprocessing
- Model
 - Objective Function
 - Optimization Algorithm
- Evaluation



How to proceed data into a model?

Vektorisasi

1 Prapengolahan

2 Tokenisasi

3 Filtering

4 Akar Kata

5 Pembobotan

Pembobotan

Word Count

Term Frequency

TF.IDF

Word Embedding

Modelling

Single Classifier

Support Vector Machine

K Nearest Neighbor

Linear Regression

Decision Tree

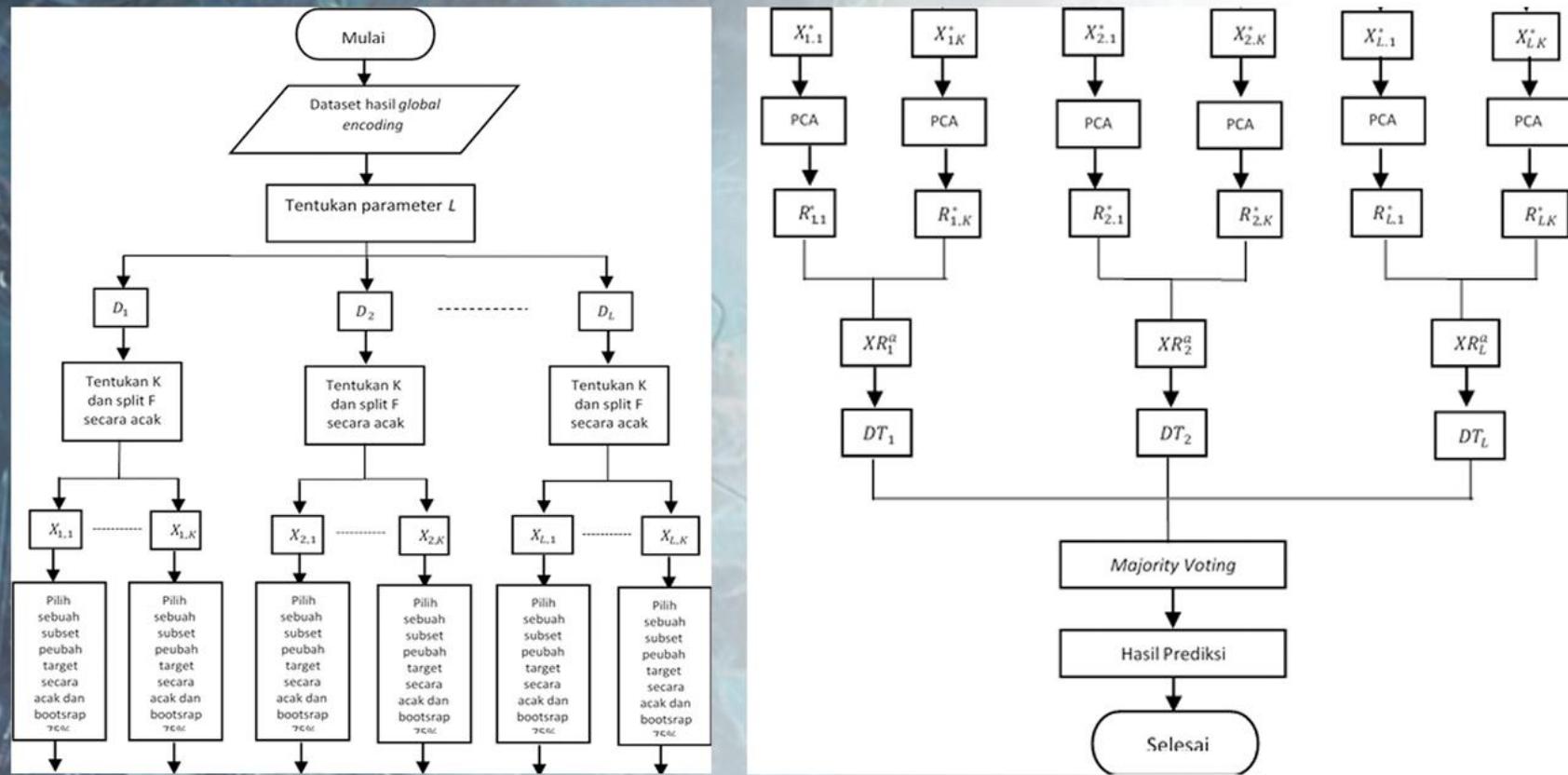
Ensemble Classifier

Bootstrap Aggregating
Rotation Forest

Binary Class

Multi Class

Ensemble Classifier: Rotation Forest



Evaluation

Seberapa prediktif model yang sudah terbentuk ?

Dengan kata lain berapa kapabilitas generalisasi dari model yang terbentuk?

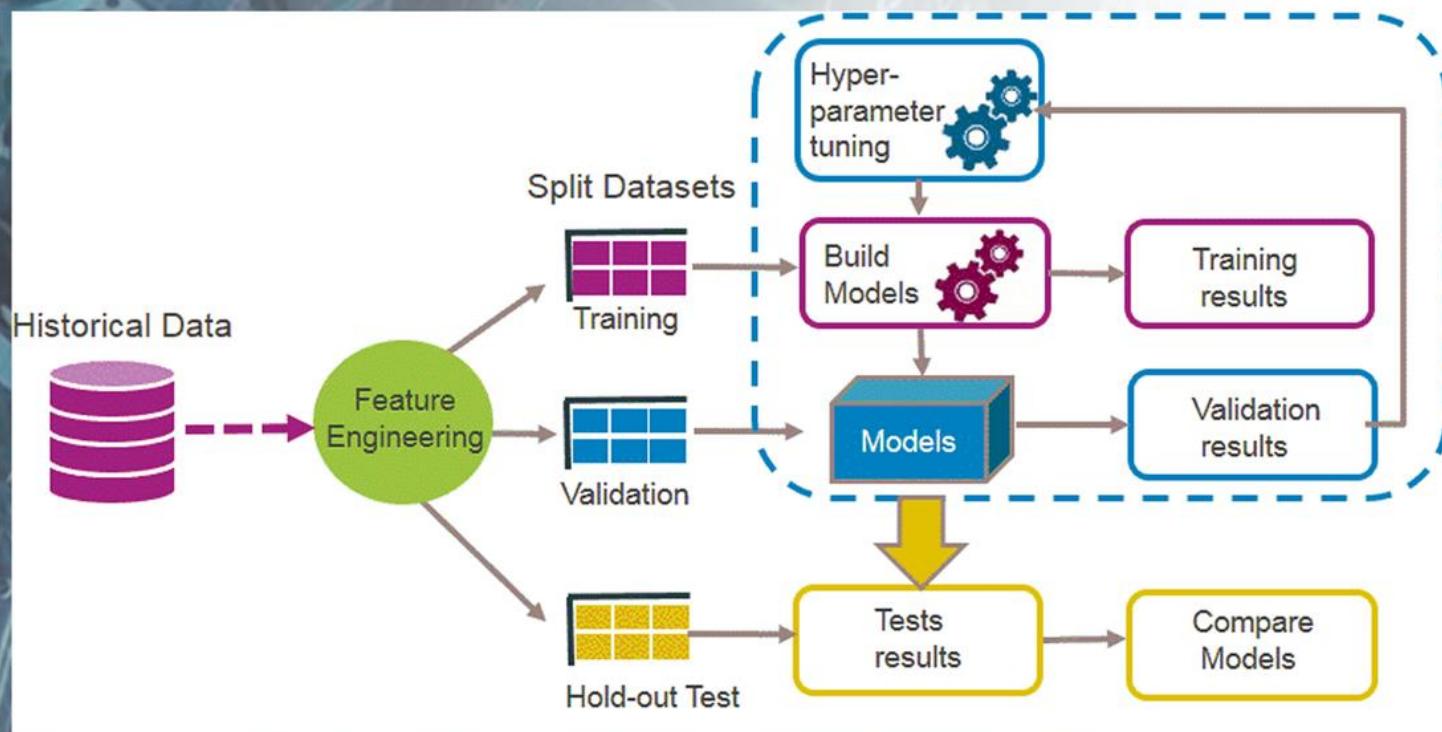
Evaluation

Secara umum, semakin besar data training maka akan semakin baik kinerja model

Sementara itu, semakin besar data testing maka akan semakin akurat estimasi kinerja

Machine Learning diharapkan handal juga untuk data training yang kecil

Evaluation: Holdout



Sumber gambar: <https://www.capgemini.com>

Evaluation: Holdout

Masalah:

Sample data boleh jadi tidak representatif

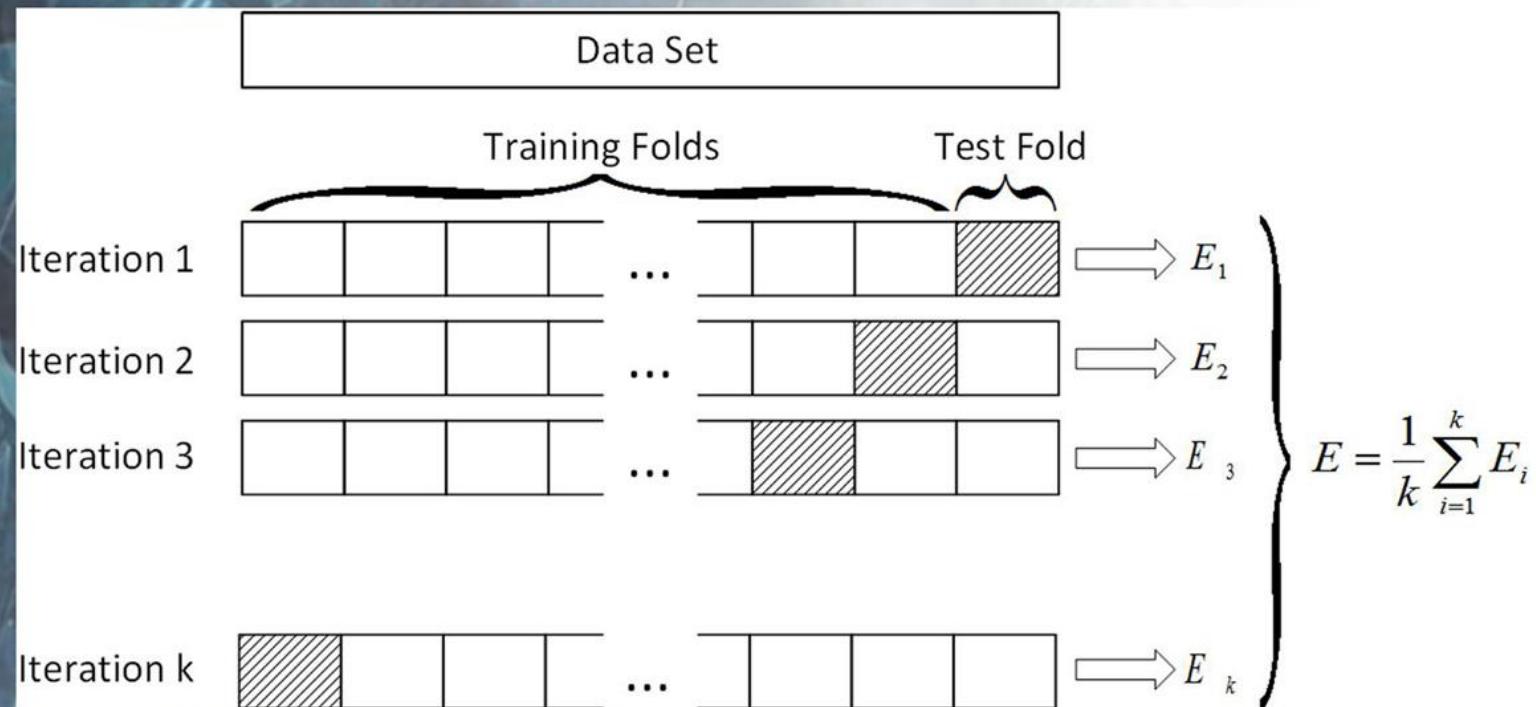
Contoh:

Suatu kelas bisa jadi tidak ada pada data testing

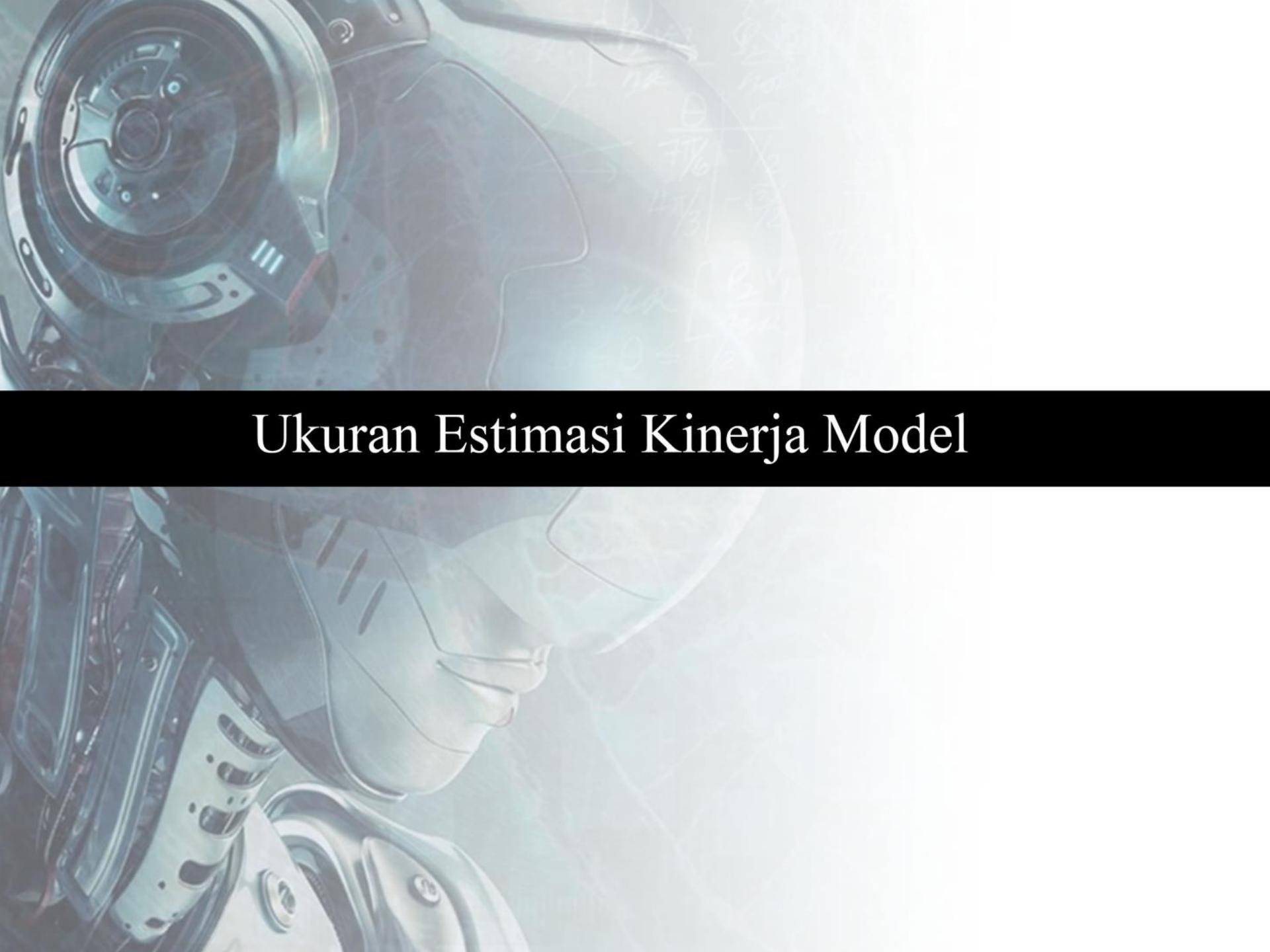
Solusi:

Menggunakan metode stratification

Evaluation: Cross Validation



Sumber Gambar: <http://www.scielo.br>



Ukuran Estimasi Kinerja Model

Confusion Matrix

Kelas Sebenarnya

Kelas Prediksi

Negatif

Negatif

Positif

Positif

TN

FP

FN

TP

Akurasi

Menurut Sammut & Webb (2011), akurasi mengacu pada ukuran sejauh mana hasil prediksi model sesuai dengan observasi penelitian yang sebenarnya, atau dengan kata lain, seberapa sering model yang dibentuk dapat memprediksi suatu permasalahan dengan benar.

$$\text{Akurasi} = \frac{(TN + TP)}{(TN + FP + FN + TP)}$$

Sensitivitas/ Recall

Menurut Rokach & Maimon (2008), sensitivitas atau sering disebut dengan recall adalah seberapa baik kemampuan sebuah model prediksi untuk mengenali semua observasi positif

$$\text{Sensitivitas} = \frac{TP}{(TP + FN)}$$

Spesifisitas

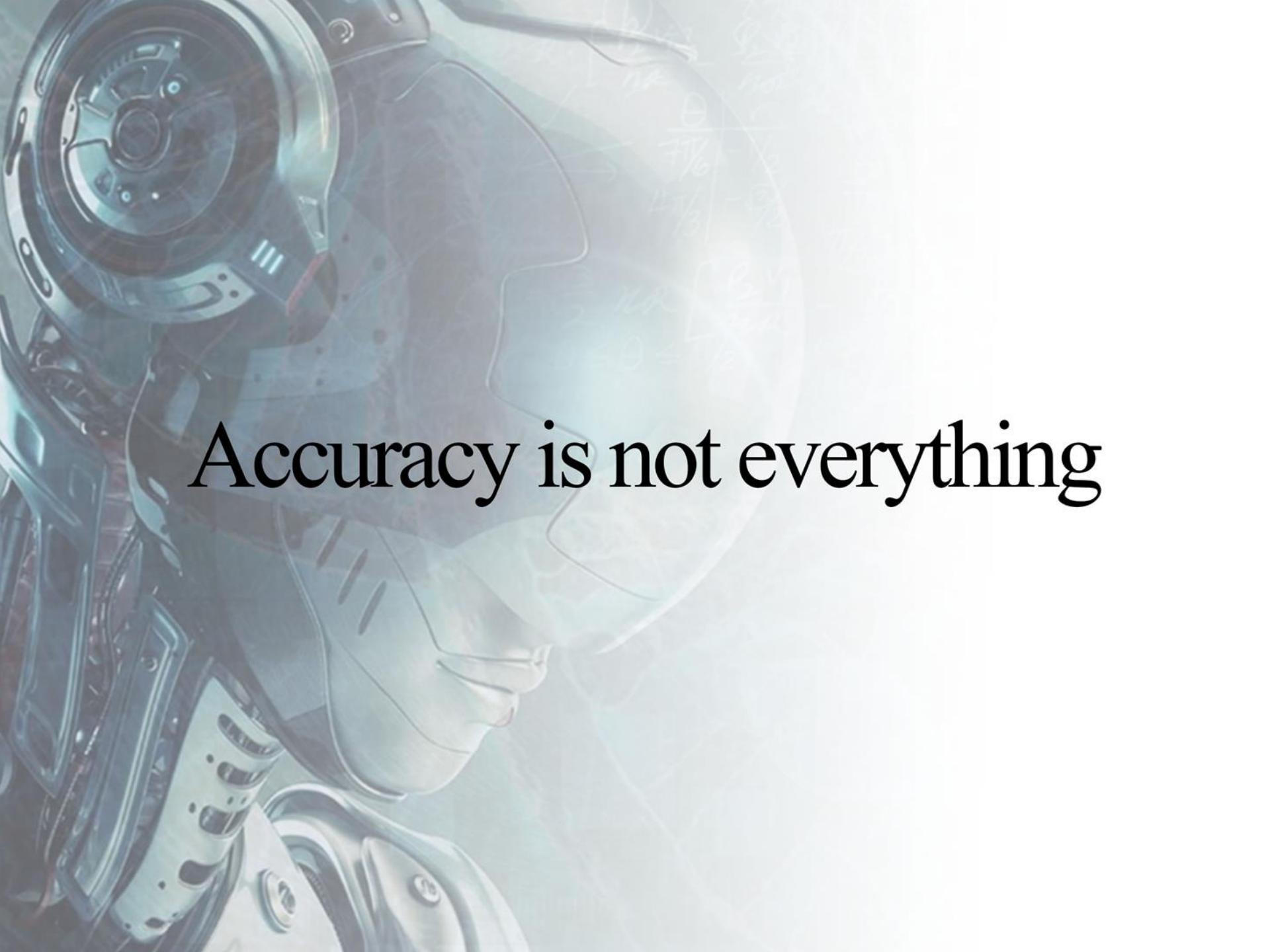
Berbeda dengan sensitivitas, spesifisitas atau true negative rate adalah ukuran seberapa baik kemampuan sebuah model prediksi untuk mengenali semua sample negatif

$$\text{Spesifisitas} = \frac{TN}{(TN + FP)}$$

Presisi

Presisi sering digambarkan dengan tingkat ketepatan antara informasi yang di inginkan dengan jawaban yang diberikan oleh model

$$\text{Presisi} = \frac{TP}{(TP + FP)}$$



Accuracy is not everything

“Misalkan kita ingin mengukur kinerja sebuah mesin prediksi interaksi protein HIV dengan manusia. Mesin tersebut bertugas mencari dan kemudian memprediksi interaksi protein tersebut dengan dua kategori, berinteraksi atau tidak berinteraksi. Kita menguji mesin tersebut dengan memberikan 100 protein dengan kategori berinteraksi dan 900 protein dengan kategori tidak berinteraksi. Hasilnya mesin tersebut mampu memisahkan 110 protein yang terdeteksi sebagai protein berinteraksi. Kemudian 110 protein hasil prediksi yang dilakukan oleh mesin tersebut diperiksa kembali. Dari 110 protein yang terdeteksi sebagai protein yang berinteraksi oleh mesin tersebut, ternyata hanya terdapat 90 protein yang terprediksi dengan benar, sedangkan 20 lainnya adalah protein yang tidak berinteraksi.”

Kelas Prediksi

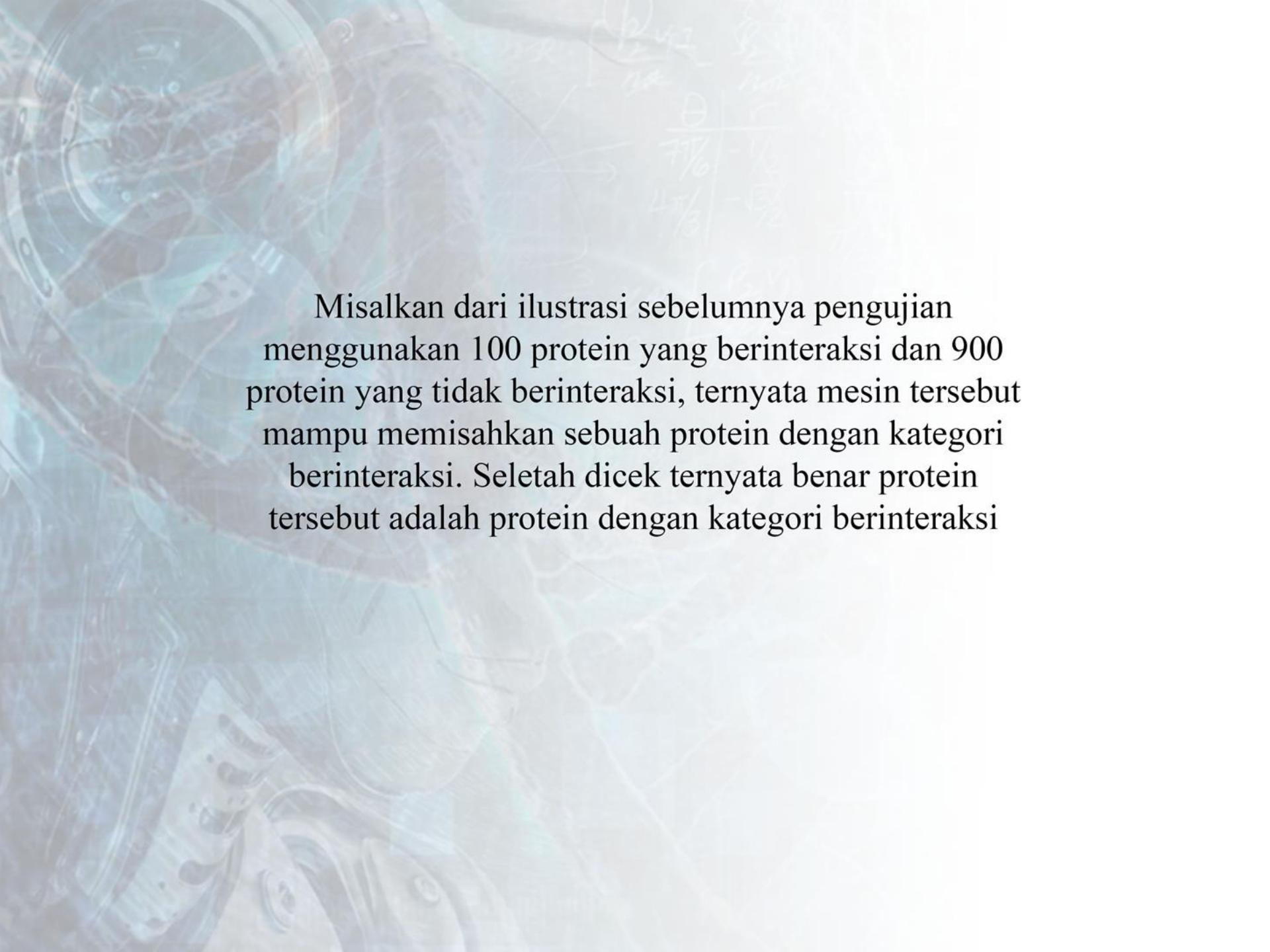
Kelas Sebenarnya	Tidak Berinteraksi	Interaksi
Tidak Berinteraksi	880	20
Interaksi	10	90

$$\text{Akurasi} = \frac{(880 + 90)}{(880 + 20 + 10 + 90)} = 0.97 = 97\%$$

$$\text{Sensitivitas} = \frac{90}{(90 + 10)} = 0.9 = 90\%$$

$$\text{Spesifisitas} = \frac{880}{(880 + 20)} = 0.97 = 97\%$$

$$\text{Presisi} = \frac{90}{(90 + 20)} = 0.82 = 82\%$$



Misalkan dari ilustrasi sebelumnya pengujian menggunakan 100 protein yang berinteraksi dan 900 protein yang tidak berinteraksi, ternyata mesin tersebut mampu memisahkan sebuah protein dengan kategori berinteraksi. Seletah dicek ternyata benar protein tersebut adalah protein dengan kategori berinteraksi

Kelas Prediksi

Kelas Sebenarnya	Tidak Berinteraksi	Interaksi
Tidak Berinteraksi	900	0
Interaksi	99	1

$$\text{Akurasi} = \frac{(900 + 1)}{(900 + 0 + 99 + 1)} = 0.901 = 90.1\%$$

$$\text{Sensitivitas} = \frac{1}{(1 + 99)} = 0.01 = 1\%$$

$$\text{Spesifisitas} = \frac{900}{(900 + 0)} = 1 = 100\%$$

$$\text{Presisi} = \frac{1}{(1 + 0)} = 1 = 100\%$$

Others Terminology

- Feature Extraction
- Feature Selection
- Overfitting and Underfitting
- Linearly Separable
- Biclustering
- Curse of Dimensionality
- Central of Gravity
- Hyper parameter
- Kernel Tricks