

# Introduction to Machine Learning



# Outline Kelas

- Pengenalan machine Learning
- Tipe-tipe machine learning
- Workflow machine learning: CRISP-DM
- Praktik membangun model ML: Regresi

# Pengenalan Machine Learning

**Menurutmu,  
“Machine Learning” itu apa?**

# QUIZ

Prediksi Seleksi Maba  
Prodi Pendidikan Dokter  
Universitas Cetar  
Membahana

Cuplikan data tahun lalu

<i>Fitur</i>		<i>Target</i>	
<b>Skor IPA</b>	<b>Skor TPA</b>	<b>Donasi Orangtua</b>	<b>Lulus</b>
90	60	1 Milyar	YA
70	70	0	TIDAK
90	60	0.5 Milyar	TIDAK
50	100	1 Milyar	YA
100	60	0	TIDAK
20	10	5 Milyar	YA
80	80	1 Milyar	YA

Performa salah satu peserta tahun ini.

<b>90</b>	<b>0</b>	<b>10 Milyar</b>	<b>?</b>
-----------	----------	------------------	----------

# QUIZ

Prediksi Seleksi Maba  
Prodi Pendidikan Dokter  
Universitas Cetar  
Membahana

Cuplikan data tahun lalu

Fitur		Target	
Skor IPA	Skor TPA	Donasi Orangtua	Lulus
90	60	1 Milyar	YA
70	30	0	TIDAK
90	50	0.5 Milyar	TIDAK
50	100	1 Milyar	YA
100	60	0	TIDAK
20	60	5 Milyar	YA
80	80	1 Milyar	YA

Performa salah satu peserta tahun ini.

90	0	10 Milyar	YA
----	---	-----------	----

Kriteria lulus panitia SMB:

- Donasi ortu minimum 1 Milyar

# Selamat, Anda baru saja melakukan bagian “Learning” pada Machine Learning

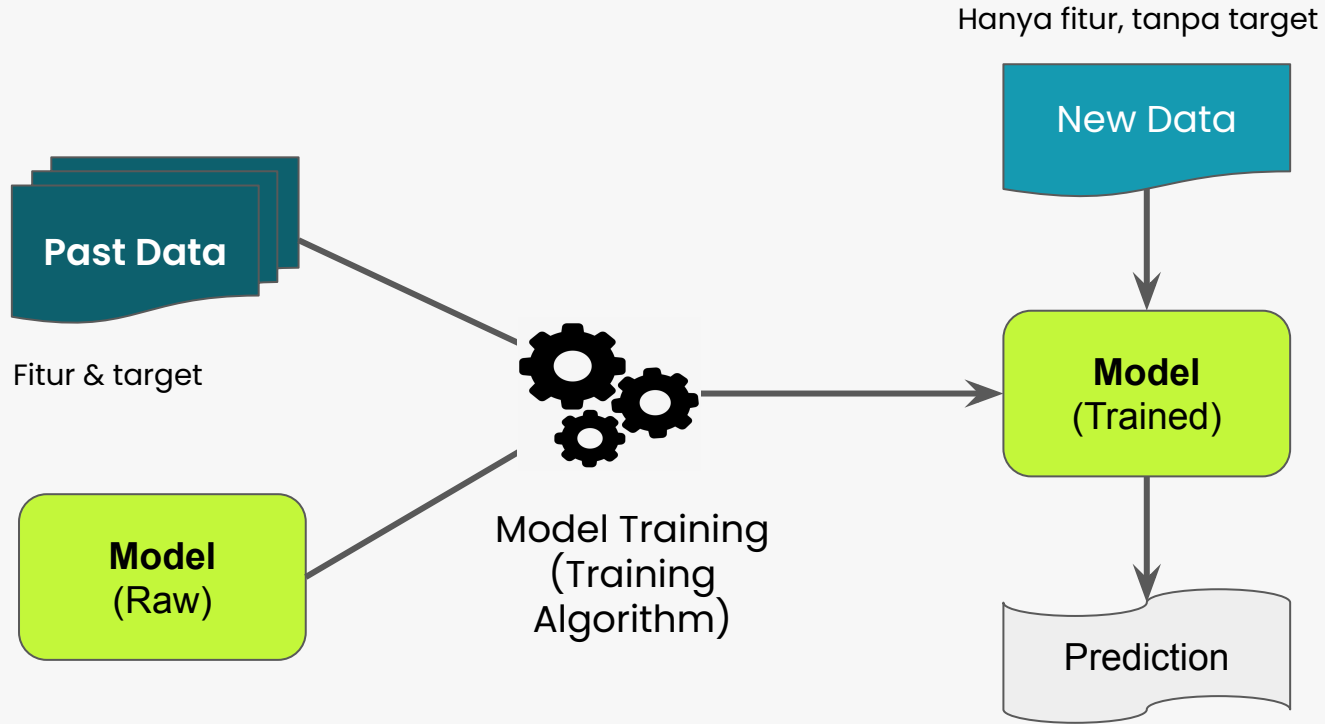
1. Kita memiliki sample data yang “lengkap”
  - a. Ada komponen “fitur”
  - b. Ada komponen “target prediksi”
2. **Pelajari/educated guess pola/logic yang menghubungkan komponen “fitur” terhadap nilai “target prediksi”**
  - a. Bagaimana aturan yang menghubungkan “fitur” dengan “target prediksi”?
3. **Gunakan logic yang diduga tersebut untuk memprediksi nilai “target prediksi” pada data “baru”**
  - a. Data baru: data yang hanya memiliki komponen “fitur” saja
  - b. Harapan: semoga prediksi kita akurat

# Lalu bagaimana dengan “Machine” pada Machine Learning?

- Penambahan “machine” pada “machine learning” simply berarti proses learning tadi dilakukan oleh sebuah “model” via suatu “algoritma pembelajaran”
- **Model:** sederhananya persamaan matematika
  - E.g.  $\text{sales} = 2 * \text{marketing\_spend} + 3 * \text{selling\_hours}$
- **Algoritma pembelajaran:** aturan sistematis/resep untuk mengajari model menangkap pola/logic yang menghubungkan antara “fitur” dan “target prediksi”
  - Output: koefisien dari setiap fitur pada model (e.g. angka 2 pada marketing spend)



# Diagram Machine Learning



# Tipe-tipe Machine Learning

Berdasarkan keberadaan “target variabel”

## 1. Supervised learning

- Supervised = variabel target prediksi ( $y$ ) diberikan di past data
  - $(x_1, y_1), (x_2, y_2), (x_3, y_3), \text{etc.}$
- Objective: memprediksi  $y$  seakurat mungkin dari data  $x$  yang baru
- Berdasarkan jenis target variabel, dapat dibedakan jadi 2
  - Regresi
  - Klasifikasi

## 2. Unsupervised learning

- Unsupervised = tidak ada target variabel di past data
  - $(x_1), (x_2), (x_3), \dots$
- Objective: mencari pattern yang tersembunyi dari data
  - Clustering
  - Dimensionality reduction

# Supervised Learning

## Logika

1. Gunakan data empiris/lampau agar model dapat mempelajari pola hubungan antara feature vs target variabel
2. Gunakan model yang sudah belajar tadi untuk memprediksi nilai target variabel dari data point baru

Fitur			Target
Skor IPA	Skor TPA	Donasi	Lulus
90	60	1 Milyar	YA
70	30	0	TIDAK
90	50	0.5 Milyar	TIDAK
50	100	1 Milyar	YA
100	60	0	TIDAK
20	60	5 Milyar	YA

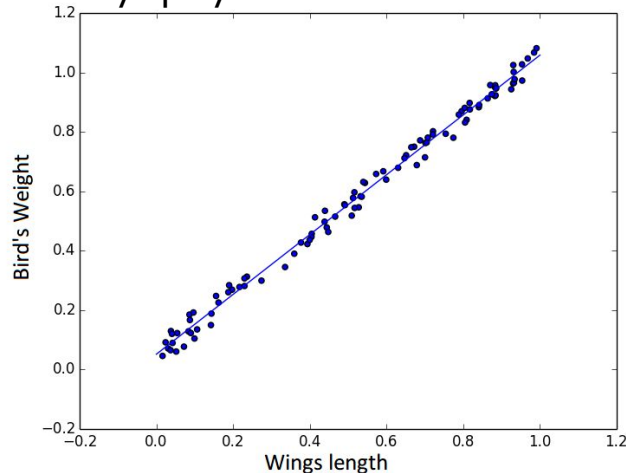
Skor IPA	Skor TPA	Donasi	Lulus
90	0	10 Milyar	???

*Lakukan prediksi nilai "target" pada data baru*

*Dan masih banyak lagi data lainnya*

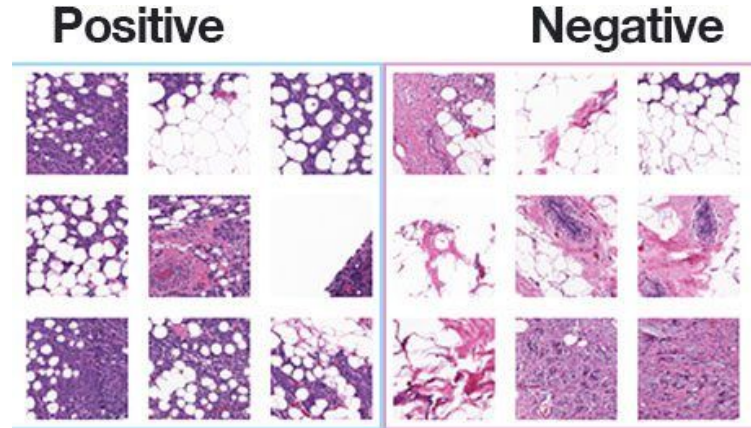
## Regresi

- Supervised learning dengan target berupa variabel **numerik/continuous**
- E.g. prediksi berat burung berdasarkan panjang rentang sayapnya



## Klasifikasi

- Supervised learning dengan target variabel berupa kelas-kelas **kategorikal**
  - E.g. male/female, yes/no, etc
- E.g. prediksi diagnosis kanker



# Unsupervised Learning

## Logika

1. Hanya fitur/atribut yang tersedia di dataset
2. Tujuan: segmentasi data points secara otomatis sesuai kemiripannya

Fitur		
Skor IPA	Skor TPA	Donasi
90	60	1 Milyar
70	30	0
90	50	0.5 Milyar
50	100	1 Milyar
100	60	0
20	60	5 Milyar

Bagaimana mengelompokkan calon maba berdasarkan keseluruhan performanya?

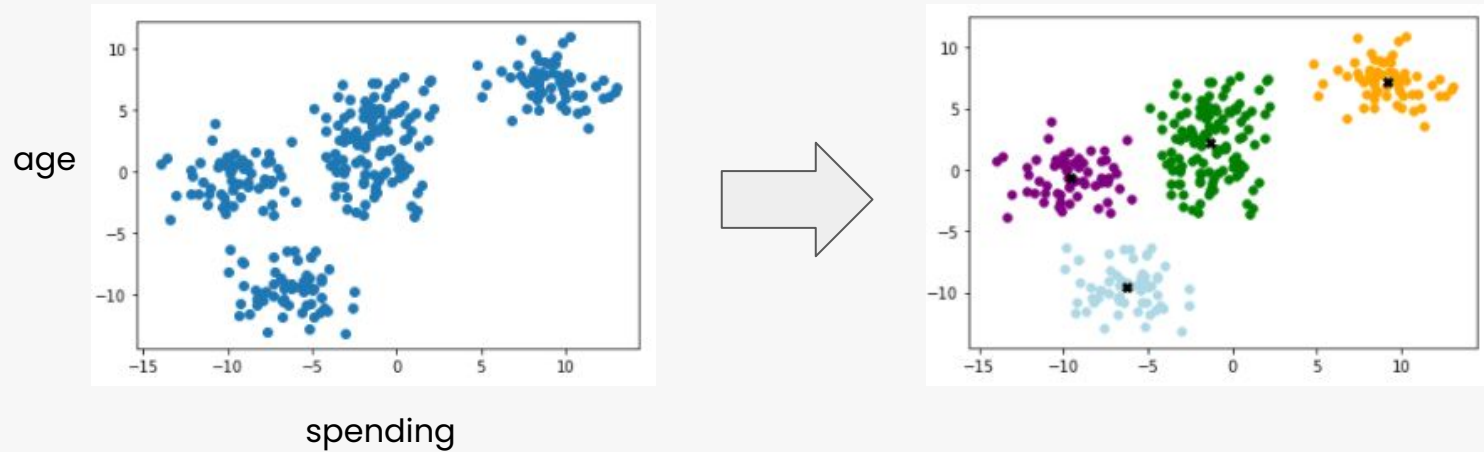
- Siapa saja Good Performer?
- Siapa saja B-aja Performer?
- Siapa saja Bad Performer?

*Dan masih banyak lagi data lainnya*

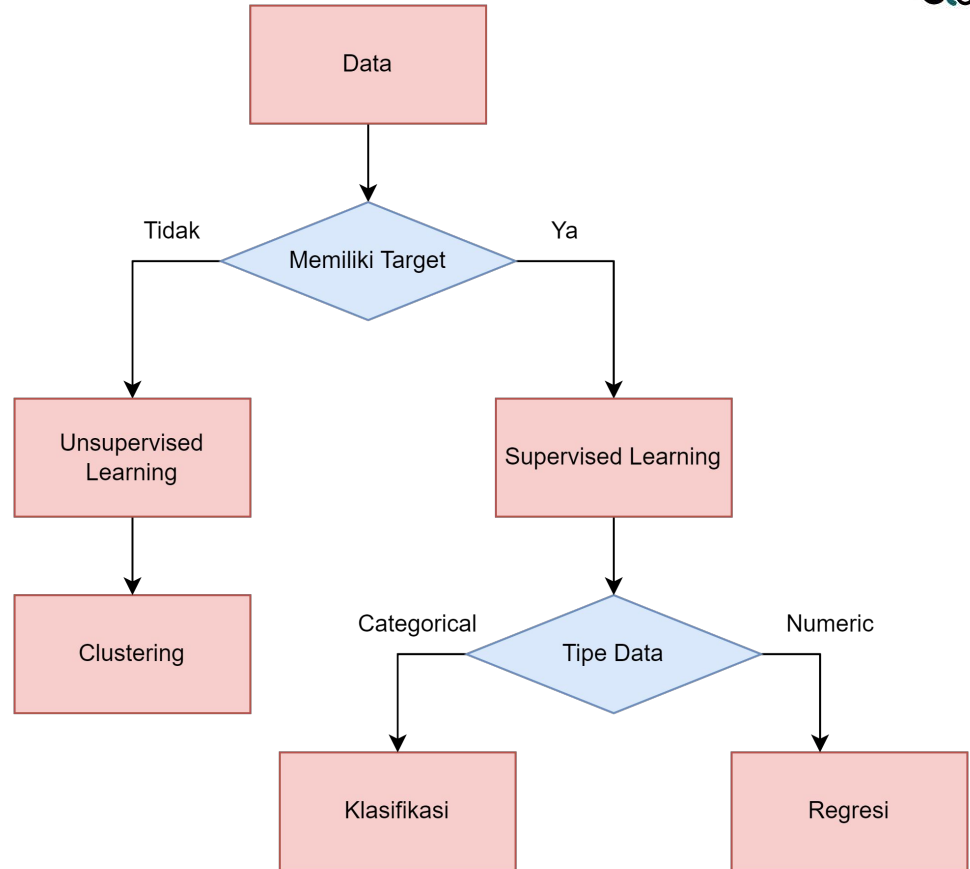
# Unsupervised Learning

Contoh: K-means clustering

Misalkan kita diberikan data customers dengan dua atribut/fitur (age & spending), bagaimana melakukan segmentasi dari sana?



# Pohon Keputusan Penggunaan Model ML

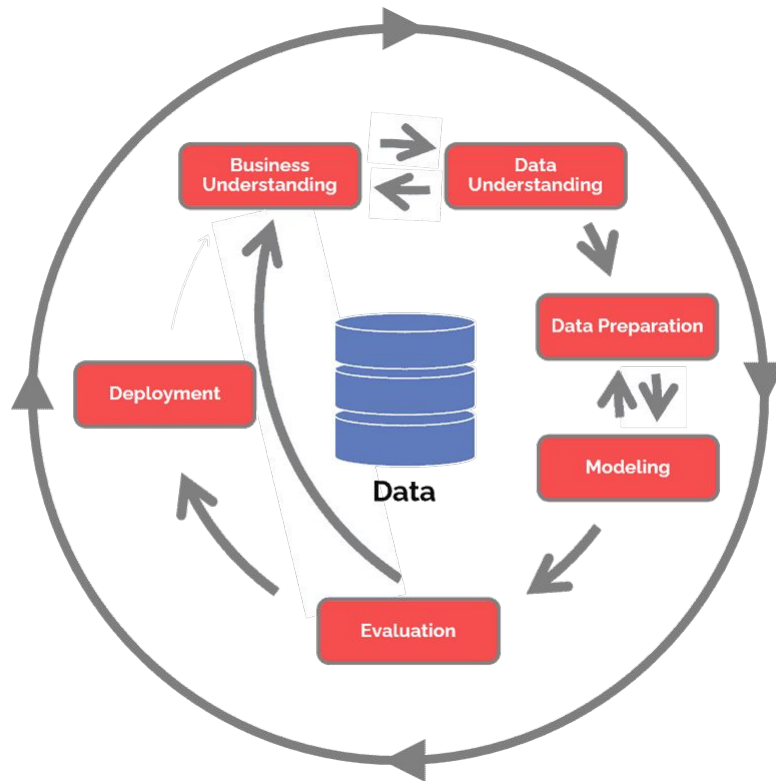


# Workflow Machine Learning CRISP-DM



# CRISP-DM

Framework project  
Machine Learning



# Business Understanding



- Ide utama data science: **menciptakan business value**
- Jadi, semuanya bermula dan berakhir pada konteks bisnis
  - **Start:** business requirements
  - **End:** business evaluation, apakah requirements terpenuhi?
- Tugas pertama data scientist adalah **memahami bisnis**
  - Bagaimana bisnis berjalan?
  - Apa masalah bisnis yang sedang terjadi?
- Setelahnya, kita dapat menyusun rancangan solusi analytics/data science untuk menyelesaikan masalah bisnis tsb

# Business Understanding



- Misal masalah nya adalah: **“Churn user kita meningkat 3 bulan terakhir”**
  - Churn: fenomena customer meninggalkan suatu layanan yg biasa ia beli
- Objektif nya dapat berupa: **“Bagaimana secara otomatis mendeteksi user yang berpeluang tinggi untuk churn?”**
- Dari sini kita dapat mengusulkan solusi berikut
  - Sebuah **ML model untuk memprediksi churn**

# Data Understanding



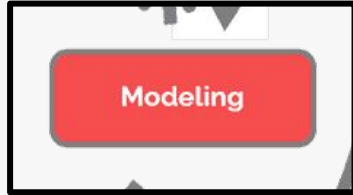
- **Data requirements:** daftarkan semua data/metrik yang dibutuhkan untuk masuk kedalam model
- **Data collection:** temukan dan collect data yang dibutuhkan
  - Biasanya dengan konsultasi pada Data Engineer
- **Data understanding:** seringkali, data yang ada belum langsung dapat digunakan (masih mentah, kotor, etc)
  - Tugas kita memahami data mentah ini

# Data Preparation



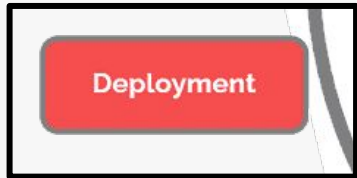
- Setelah memahami datanya, kita perlu **transform/siapkan datanya** agar ready untuk menjadi training data model kita
  - Ini adalah part yang paling melelahkan dari tugas seorang DS!
  - Hal-hal yang dikerjakan:
    - handling missing data,
    - feature encoding,
    - feature engineering, etc

# Modeling



- Akhirnya, kita masuk ke **modeling step**
- Kita bangun beberapa model, dan pilih satu yang terbaik (model selection)

# Evaluation & Deployment



- Setelah memilih model terbaik, kita **evaluasi model** tsb
  - Performa model pada data yang baru?
  - Apakah behavior model make sense secara bisnis POV?
- IF OK, **deploy the model** in production.
  - Kita “taruh” model sebagai sebuah automatic decision engine yang menentukan apakah seorang user akan churn atau tidak
  - Jika diprediksi churn, kita dapat memberikan voucher ke user tsb untuk mencegah dia churn

# Pemodelan Regresi Linear



# Regresi linear

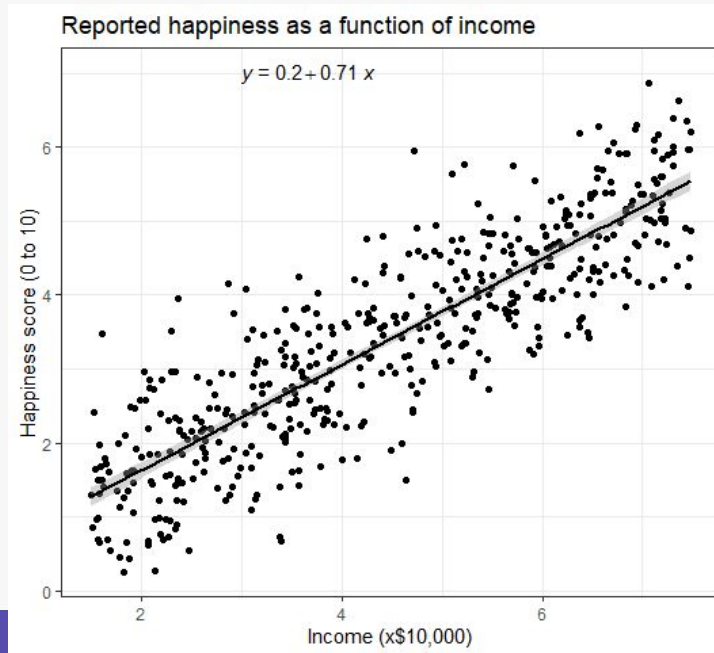
- Linear regression adalah model machine learning yang ditemukan pada tahun 1805
- Secara high-level, pemodelan linear regression adalah **upaya mencari suatu garis lurus (linear)** yang paling pas memodelkan (merepresentasikan) data
- Bentuk matematika general

$$y = b_0 + b_1x_1 + b_2x_2 + \dots + b_nx_n$$

- $y$  adalah target,  $x_1, x_2, \dots, x_n$  adalah predictors/features,  $b_0, b_1, b_2, \dots, b_n$  adalah parameters/coefficients to learn (nilainya didapat dari proses model training)

# Regresi linear

- Contoh grafik linear regression model dengan satu prediktor (income) untuk memprediksi target variabel (happiness score)



# Data yang dipakai

- Kita akan memakai `regression_data.csv`
- Data tsb tentang memprediksi peluang diterimanya seseorang (`admit_probability`), berdasarkan beberapa atribut berikut
  - GRE score
  - TOEFL score
  - University ranking
  - Motivation letter quality
  - Recommendation letter strength
  - GPA
  - Research experience

# Langkah pemodelan

1. Split data menjadi dua bagian

a. Training data (80%)

b. Test

data

(20%)

2. Convert

data

menjadi

numpy

array

3. Fit

model

linear

regression

pada

training

data

4. Evaluasi performa model pada test data

1. Menggunakan fungsi `train_test_split` dari library `sklearn`
2. Ingat: kita ingin memprediksi `admit_prob`
  - a. Jadi, target variabel kita adalah `admit_prob`

[illegible]

# Convert data menjadi numpy arrays

1. Ingat, data kita masih dalam format pandas dataframe
2. Library pemodelan sklearn bekerja dengan objek data numpy arrays

```
# convert data into numpy arrays
X_admit_train = feature_admit_train.to_numpy()
y_admit_train = target_admit_train.to_numpy().ravel()
```

# Training linear regression

1. Inisiasi model linear regression “kosongan” (belum dilatih)
2. Train model pada training data dengan sintaks `model.fit()`

```
from sklearn.linear_model import LinearRegression
```

```
# define the model
```

```
linreg = LinearRegression()
```

```
# train the model
```

```
linreg.fit(X_admit_train, y_admit_train)
```

# Model hasil training

1. Setelah training selesai, kita dapat melihat koefisien final model
2. Contoh interpretasi koefisien GPA = 0.1125:

*"Kenaikan 1 poin pada GPA, dengan menganggap Fitur lain nilainya tetap, berasosiasi dengan kenaikan Target variabel (admit probability) sebanyak 0.1125"*

	feature	coefficient
0	intercept	-1.421447
1	gre_score	0.002434
2	toefl_score	0.002996
3	univ_ranking	0.002569
4	motiv_letter_strength	0.001814
5	recommendation_strength	0.017238
6	gpa	0.112527
7	research_exp	0.024027



# Evaluasi performa model

1. Setelah model selesai di-train, kita perlu mengevaluasi performa model pada test data
2. Metrik yang dapat digunakan
  - a. MAE (mean absolute error): rata-rata error/selisih dari prediksi model vs nilai target variabel sebenarnya
  - b. MAPE (mean absolute percentage error): MAE, namun dalam bentuk persen.

# Hands-On

*Thanks!*

