DigitalSkola

LEARNING PROGRESS REVIEW

Week 10

Entropy Team

DAFTAR ISI

01

ADVANCED DATA VISUALIZATION

Visualisasi data
menggunakan Seaborn

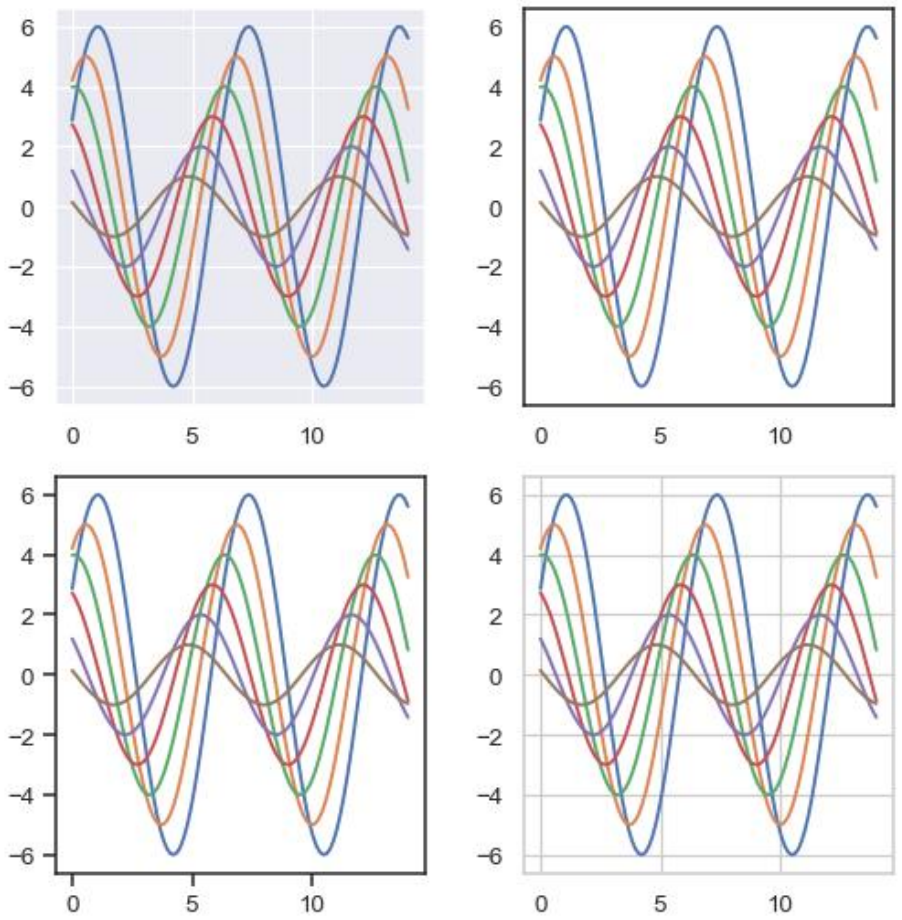
Seaborn

Seaborn adalah ***library*** **visualisasi data** yang dibangun **berdasarkan *library* Matplotlib** untuk membuat grafik statistik yang menarik dan informatif

Sumber: https://seaborn.pydata.org/

DigitalSkola

Figure Style



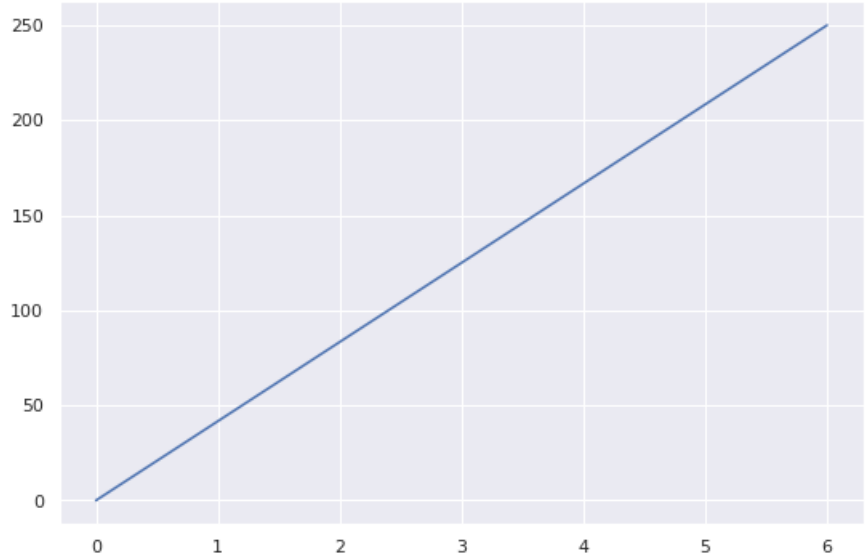
- Seaborn memiliki 5 buah tema, yaitu:
  - darkgrid
  - whitegrid
  - dark
  - white
  - ticks

```
syntax | sns.set_style("whitegrid")
       | # or
       | sns.set_theme(style="whitegrid")
```

Sumber gambar: http://seaborn.pydata.org/tutorial/aesthetics.html

DigitalSkola

Line Plot

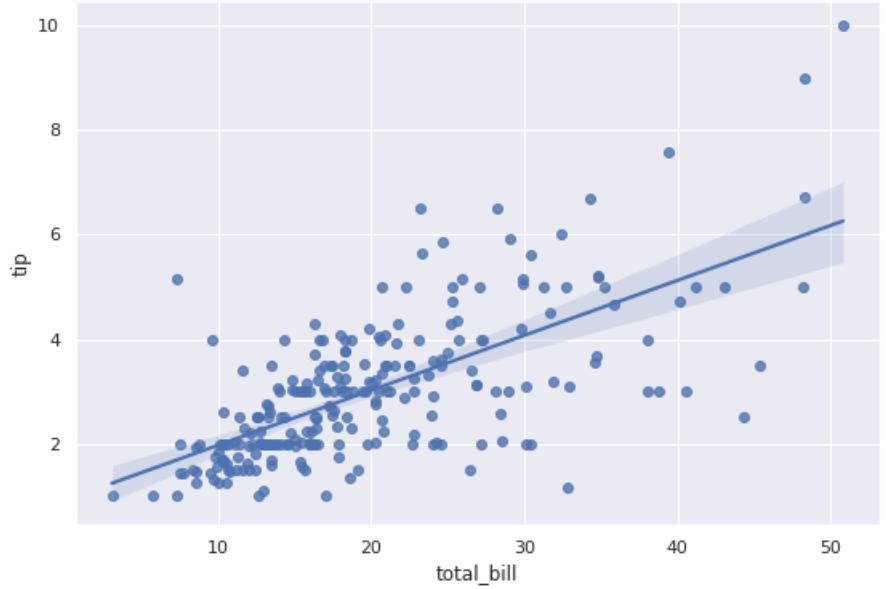


- Digunakan untuk melihat **perubahan** atau *trend*
- Digunakan untuk membandingkan perubahan seiring waktu

syntax	sns.lineplot(x, y)
	# x = the horizontal (X-axis) coordinates of the data points # y = the vertical (Y-axis) coordinates of the data points

DigitalSkola

Regression Plot

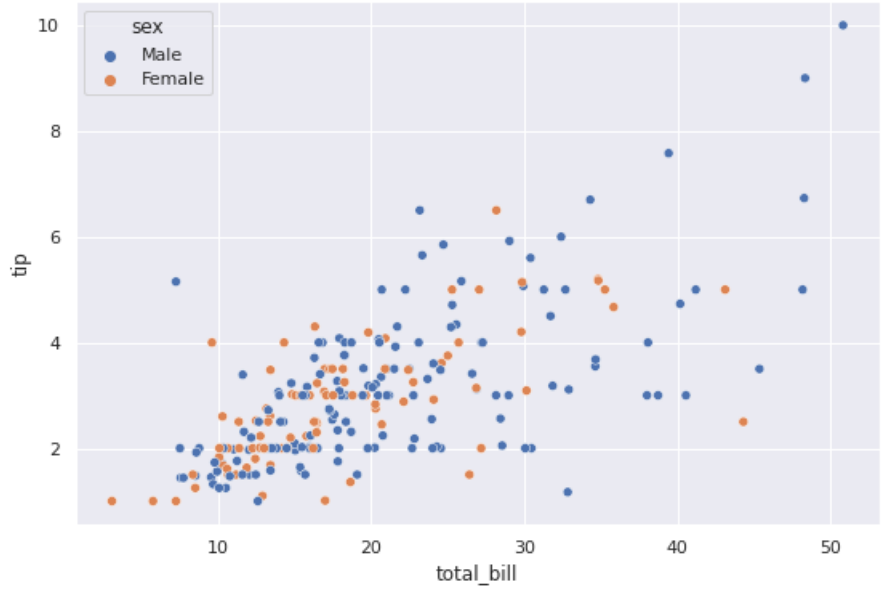


- Digunakan untuk melihat **sebaran data** dan model regresi linier

syntax	sns.regplot(x, y)
	# x = the horizontal (X-axis) coordinates of the data points # y = the vertical (Y-axis) coordinates of the data points

DigitalSkola

Scatter Plot

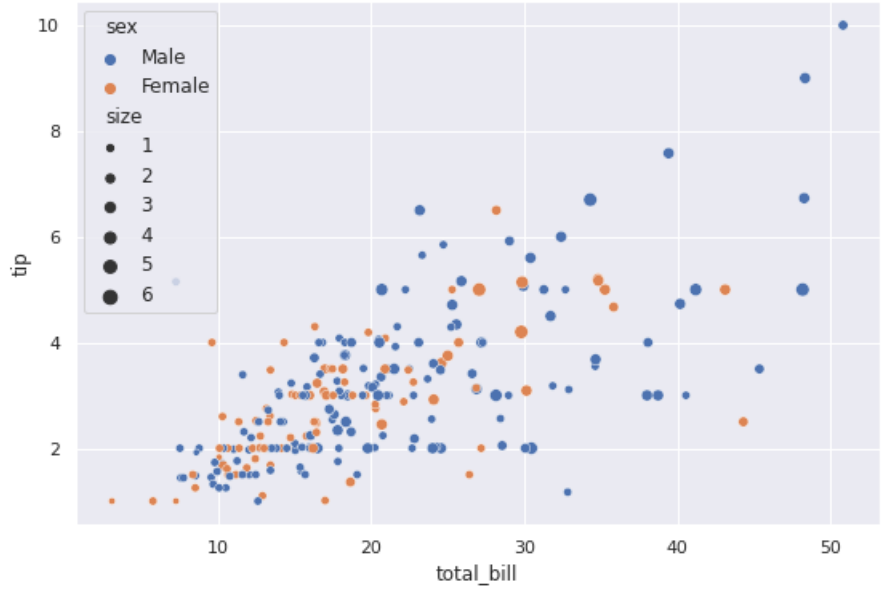


- Digunakan untuk melihat **sebaran data**
- Digunakan untuk melihat **korelasi antarvariabel**

syntax	sns.scatterplot(x, y, hue)
	# x = the horizontal (X-axis) coordinates of the data points # y = the vertical (Y-axis) coordinates of the data points # hue = Grouping variable that will produce points with different colors

DigitalSkola

Bubble Chart

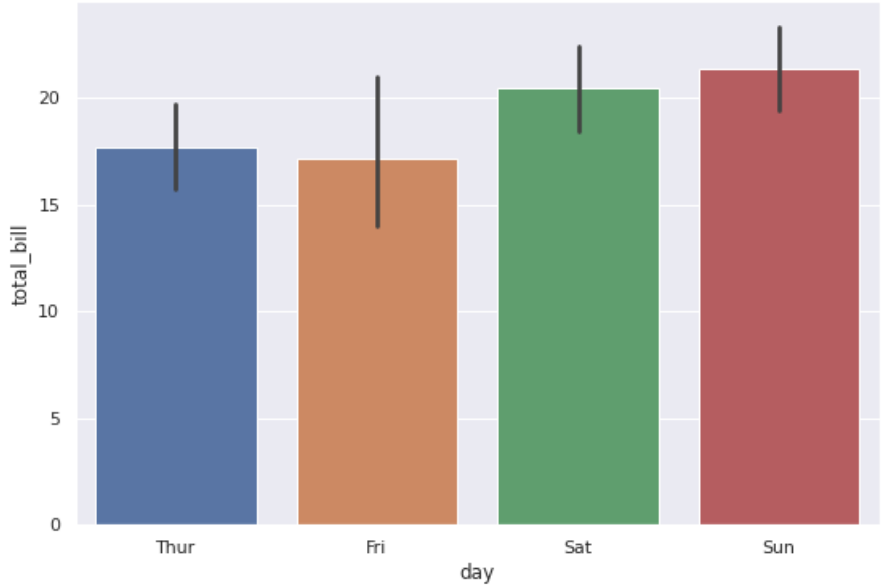


- Digunakan untuk melihat **sebaran data**
- Digunakan untuk melihat **korelasi antarvariabel**

syntax	sns.scatterplot(x, y, size)
	# x = the horizontal (X-axis) coordinates of the data points # y = the vertical (Y-axis) coordinates of the data points # size = grouping variable that will produce points with different size

DigitalSkola

Vertical Bar Plot

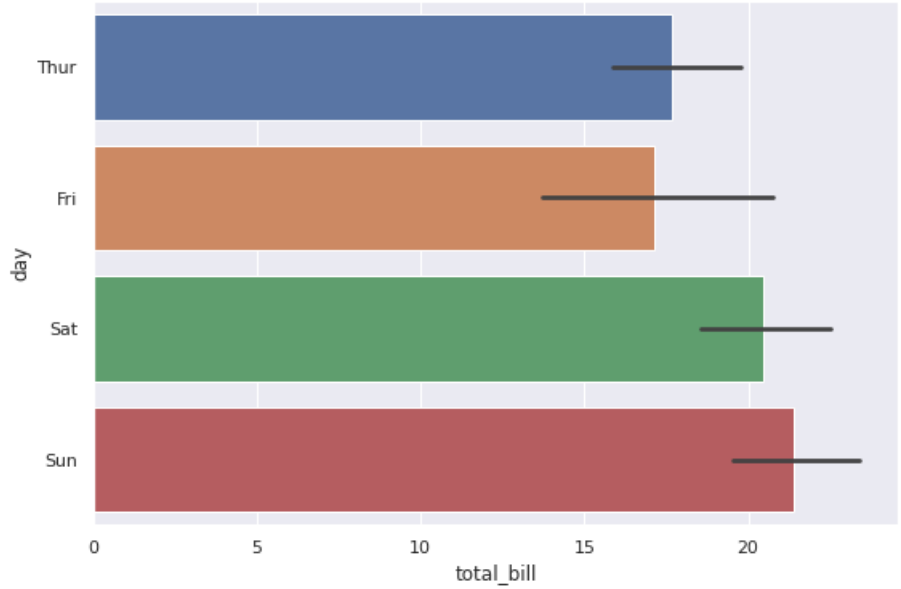


- Digunakan untuk melihat **perbandingan** dari beberapa kategori
- Digunakan untuk melihat ***ranking***

```
syntax | sns.barplot(x, y)
       | # x     = the horizontal (X-axis) coordinates of the data points
       | # y     = the height of the bars (Y-axis)
```

DigitalSkola

Horizontal Bar Plot

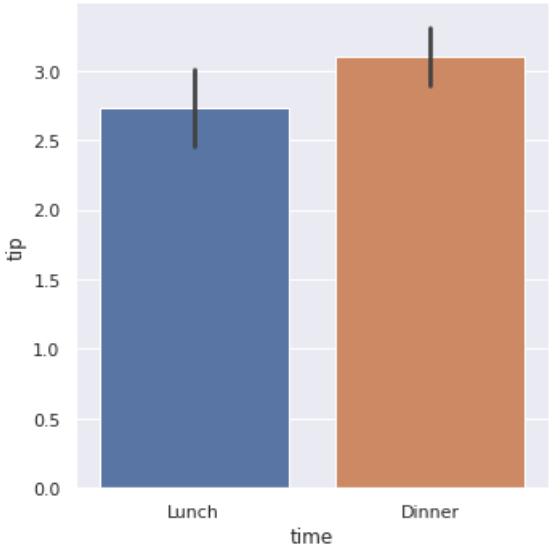


- Digunakan untuk melihat **perbandingan** dari beberapa kategori
- Digunakan untuk melihat ***ranking***

```
syntax | sns.barplot(x, y)
       | # x     = the width of the bars (X-axis)
       | # y     = the vertical (Y-axis) coordinates of the data points
```

DigitalSkola

Categorical Plot

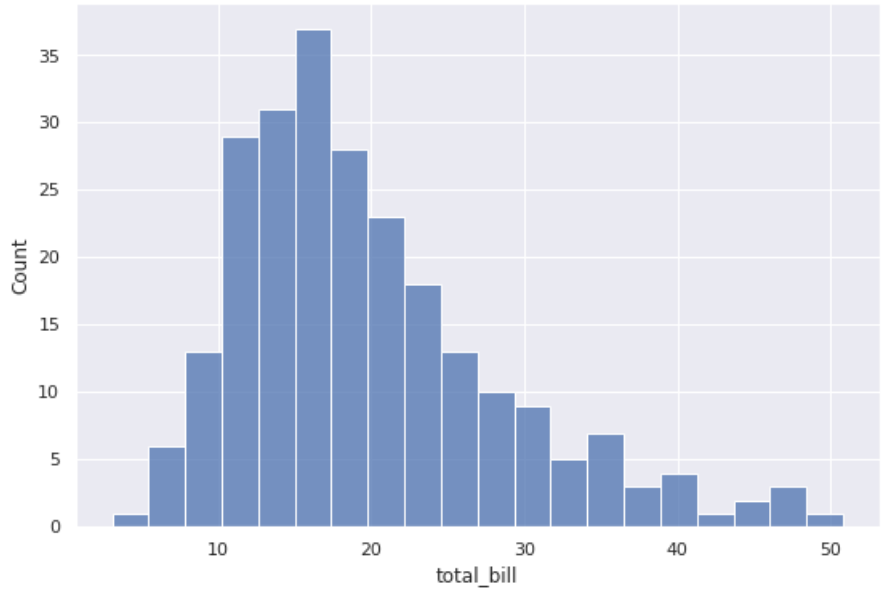


- Digunakan untuk melihat **sebaran data**
- Digunakan untuk melihat **hubungan** antara **variabel numerik** dan variabel **kategoris**

```
syntax | sns.catplot(x, y, kind)
       | # x     = the width of the bars (X-axis)
       | # y     = the vertical (Y-axis) coordinates of the data points
       | # kind  = the kind of plot to draw
```

DigitalSkola

Histogram

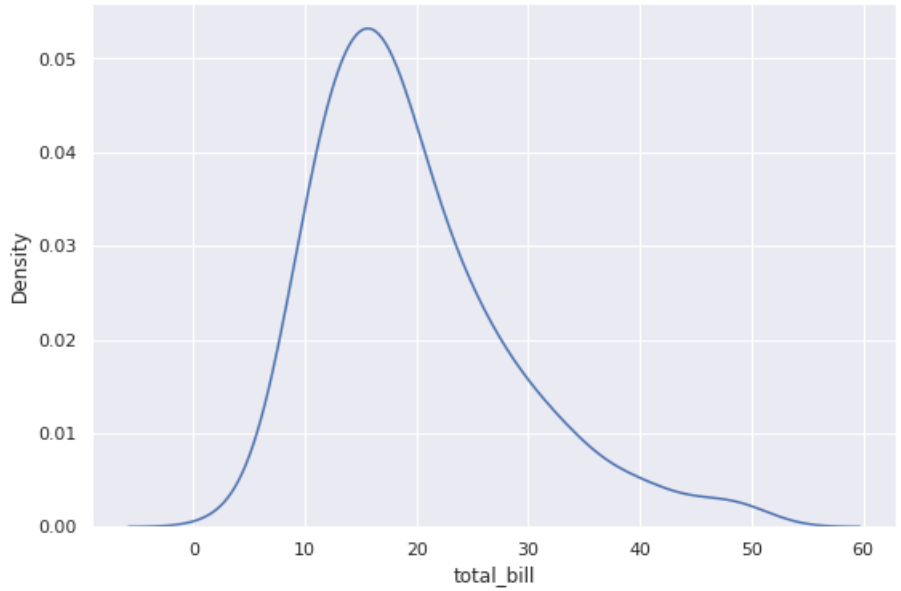


- Digunakan untuk melihat **sebaran data**
- Digunakan untuk melihat apakah data **terdistribusi** secara **normal** atau tidak

```
syntax | sns.histplot(x, bins)
       | # x     = the input data
       | # bins  = the number of bins or the bin edges
```

DigitalSkola

Density Plot

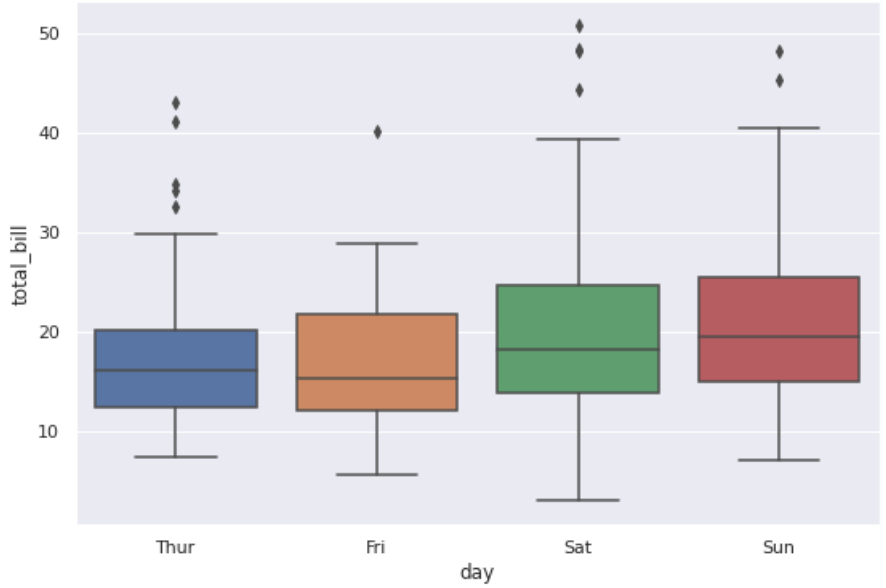


- Digunakan untuk melihat **sebaran data** menggunakan *kernel density estimation* (KDE)
- Digunakan untuk melihat apakah data **terdistribusi** secara **normal** atau tidak

```
syntax | sns.kdeplot(x)
       | # x = the input data
```

DigitalSkola

Boxplot

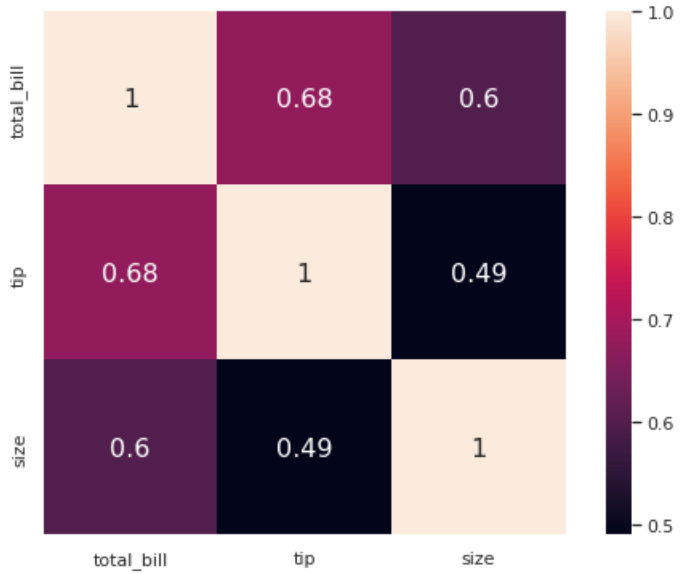


- Digunakan untuk melihat **sebaran data**
- Digunakan untuk melihat apakah ada ***outlier*** pada data

```
syntax | sns.boxplot(x, y)
       | # x = the input data
       | # y = the input data
```

DigitalSkola

Heatmap

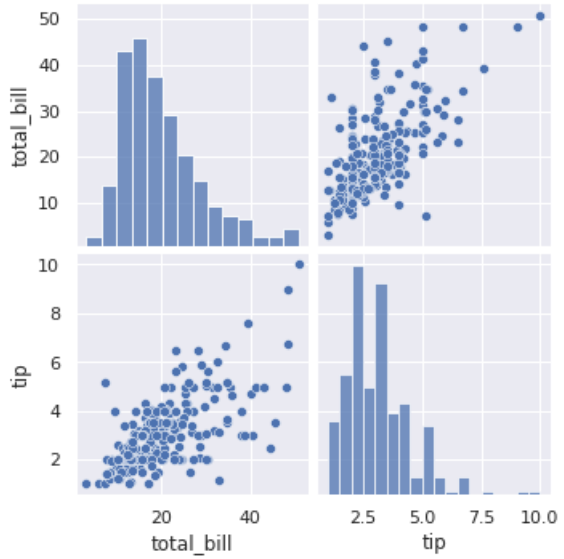


- Digunakan untuk membuat **data *tabular*** memiliki **warna** yang **bervariasi** sesuai nilainya
- Sering digunakan untuk melihat **korelasi antarvariabel**

```
syntax | sns.heatmap(data, annot)
       | # data  = 2D dataset that can be coerced into a ndarray
       | # annot = if True, write the data value in each cell
```

DigitalSkola

Pair Plot

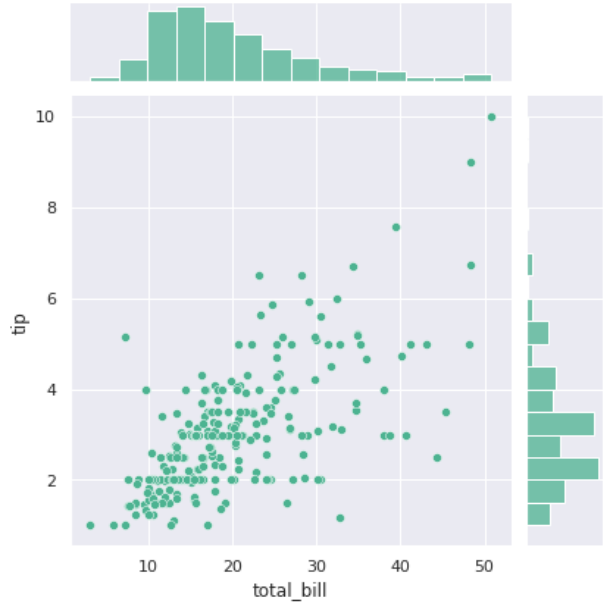


- Digunakan untuk melihat **sebaran data**
- Digunakan untuk melihat **hubungan antarvariabel**

```
syntax | sns.pairplot(data, kind, diag_kind)
       | # data          = the input data
       | # kind          = kind of plot to make
       | # diag_kind     = kind of plot for the diagonal subplots
```

DigitalSkola

Joint Plot



- Digunakan untuk membuat **grafik bivariat** dan **univariat** dari 2 variabel

syntax	sns.jointplot(x, y)
	# x = the horizontal (X-axis) coordinates of the data points # y = the vertical (Y-axis) coordinates of the data points

DigitalSkola

Beberapa Library Lain

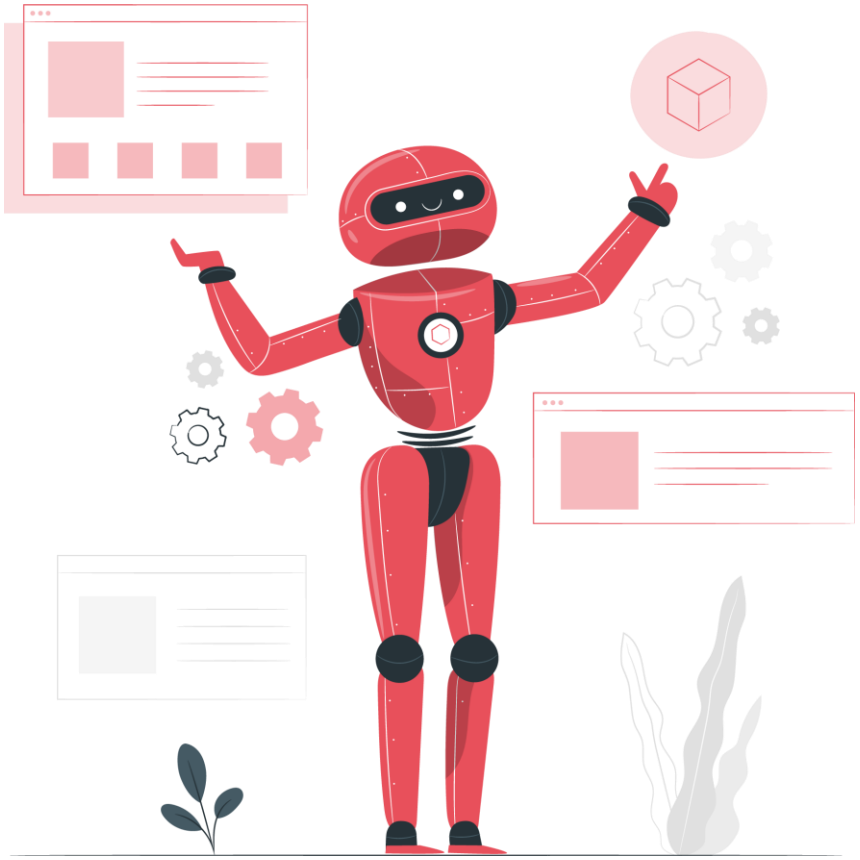
- **Folium**
  Visualisasi peta
- **Wordcloud**
  Visualisasi kemunculan kata-kata
- **Plotly**
  Visualisasi yang lebih interaktif
- **Bokeh**
  Visualisasi yang lebih interaktif

02

INTRODUCTION TO MACHINE LEARNING

Pengenalan tentang
Machine Learning

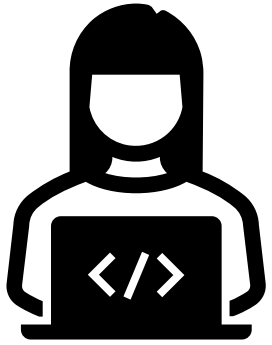
Machine Learning



- ***Machine learning*** **(**ML) merupakan cabang dari ***artificial intelligence*** (AI)
- ML adalah studi tentang **algoritma** di mana suatu sistem dapat **belajar dari data**, mengidentifikasi pola, dan membuat keputusan dengan **intervensi manusia yang minimal**

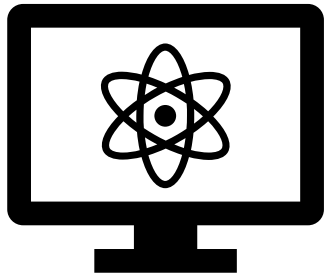
Sumber: https://www.sas.com/en_id/insights/analytics/machine-learning.html

ML vs Traditional Programming



Traditional Programming

Aturan-aturan dalam sistem **dibuat secara manual** oleh manusia



Machine Learning

Aturan-aturan dalam sistem **dibuat secara otomatis** oleh algoritma ML

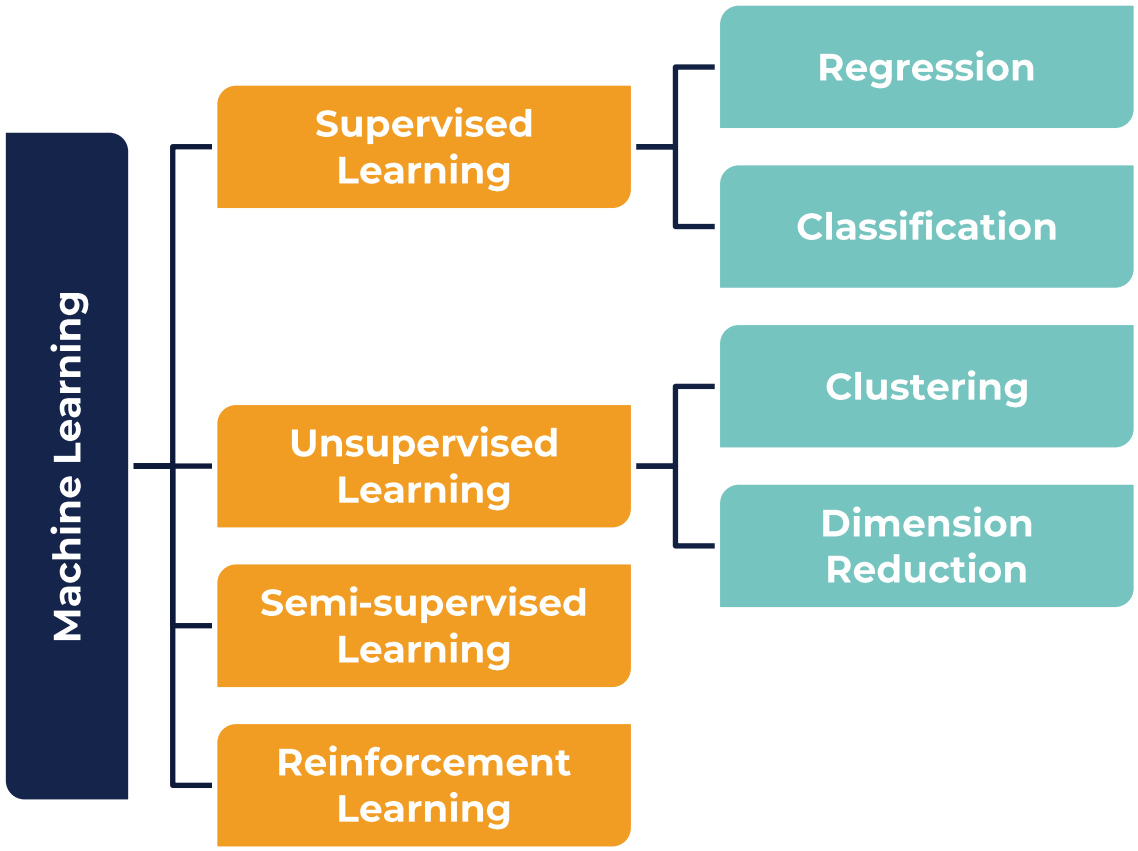
Jenis Machine Learning

Supervised Learning
Sistem belajar berdasarkan dataset yang **memiliki label**

Unsupervised Learning
Sistem belajar berdasarkan dataset yang **tidak memiliki label**

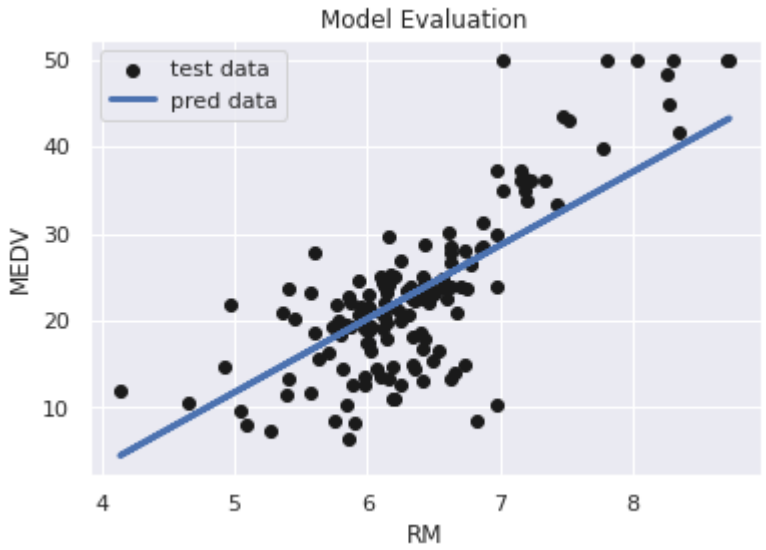
Semi-supervised Learning
Sistem belajar berdasarkan dataset yang **sebagian tidak memiliki label**

Reinforcement Learning
Sistem belajar berdasarkan ***reward*** dan ***punishment*** dari pengalaman



DigitalSkola

Regression



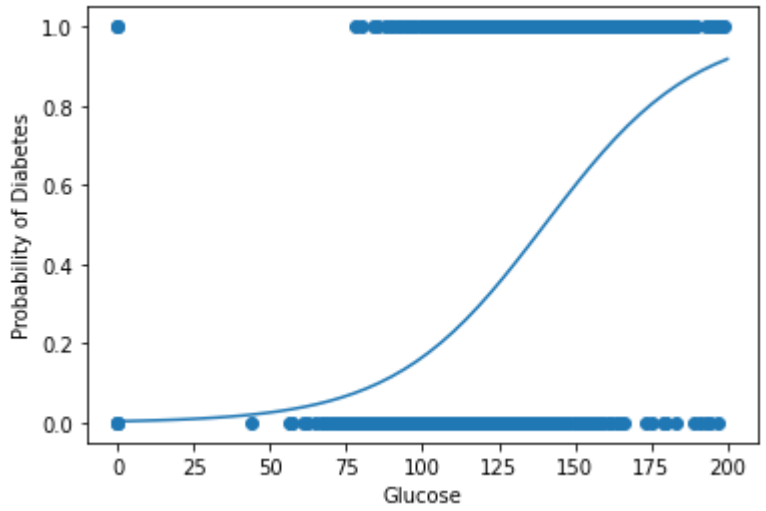
- ***Regression*** merupakan salah satu jenis *supervised* ML di mana target berupa **nilai kontinu**

- ***Linear regression*** merupakan salah satu jenis algoritma *regression* yang digunakan untuk **membuat garis lurus** berdasarkan hubungan antara variabel dependen dengan variabel independen

Sumber: https://www.kaggle.com/adhang/boston-house-prices-linear-regression

DigitalSkola

Classification



- ***Classification*** merupakan salah satu jenis *supervised* ML di mana target berupa **nilai diskrit**

- ***Logistic regression*** merupakan salah satu jenis algoritma *classification* yang digunakan untuk memisahkan 2 kelas berdasarkan **nilai probabilitas** yang dihitung dengan ***logistic/ sigmoid function***

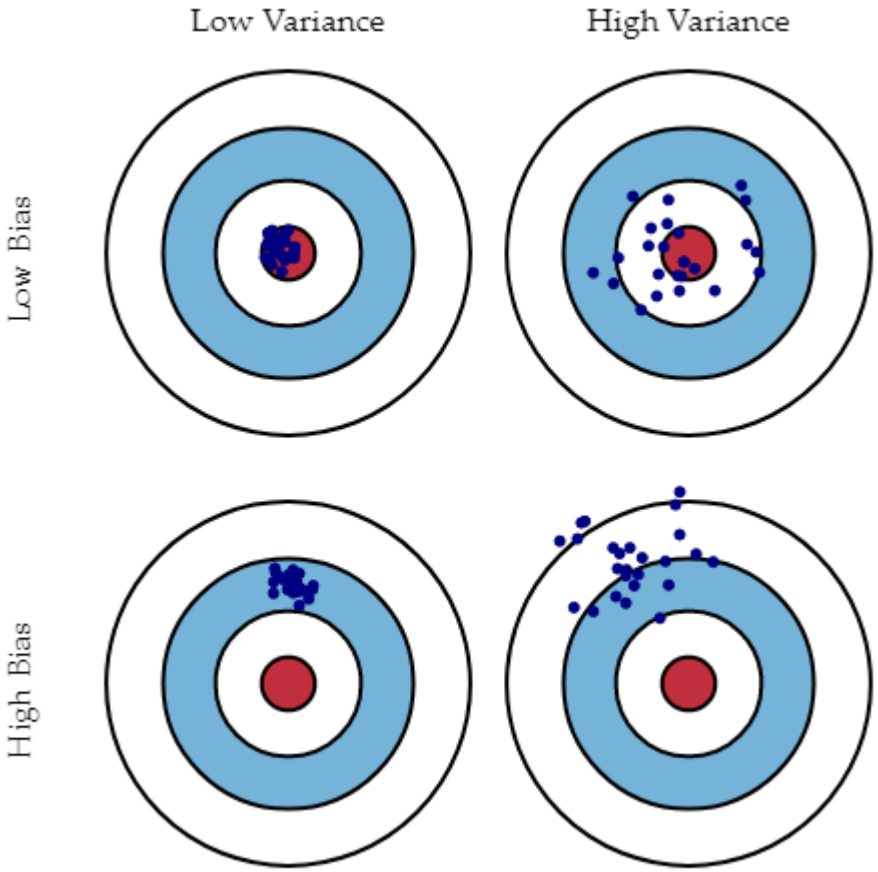
DigitalSkola

Alur Pembuatan ML

- **Dataset Understanding**
  Memahami dataset
- **Exploratory Data Analysis**
  Investigasi dataset
- **Data Preprocessing**
  Membersihkan dataset
- **Model Development**
  Pembuatan model
- **Model Evaluation**
  Penilaian performa model
- **Model Deployment**
  Implementasi model

Alur tidak harus selalu maju, ada kalanya kita kembali ke tahap awal

Bias dan Variance

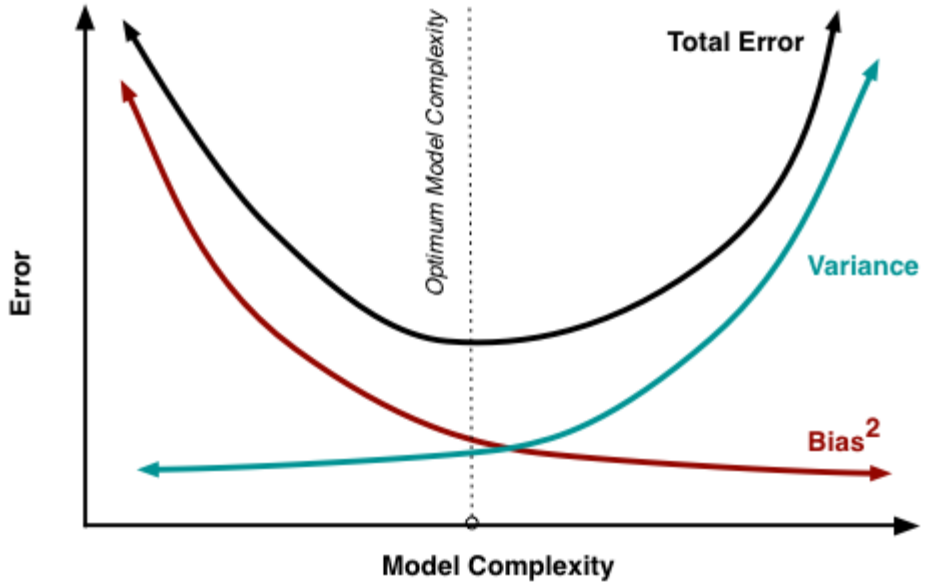


- *Bias* merupakan ukuran **seberapa jauh** hasil prediksi model dengan nilai yang sebenarnya
- *Variance* merupakan ukuran **seberapa menyebar** hasil prediksi model

Sumber: https://scott.fortmann-roe.com/docs/BiasVariance.html

DigitalSkola

Model Complexity

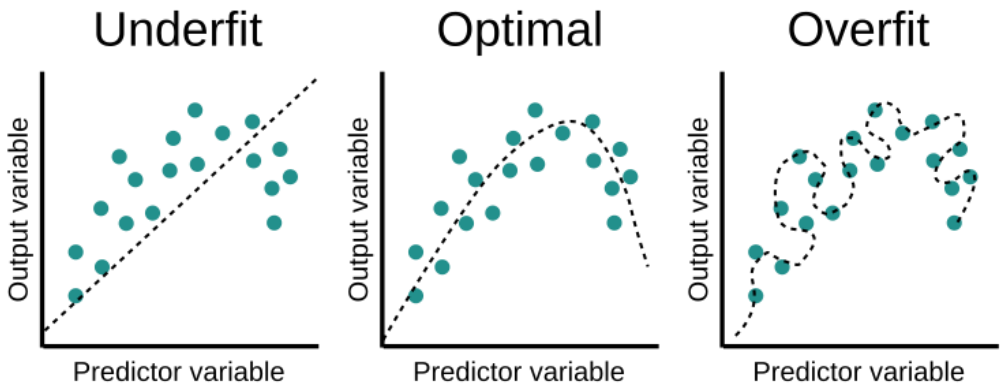


- Semakin kompleks suatu model, maka ***bias*** **akan semakin berkurang**
- Semakin kompleks suatu model, maka ***variance*** **akan semakin bertambah**

Sumber: https://scott.fortmann-roe.com/docs/BiasVariance.html

DigitalSkola

Undefitting & Overfitting



- ***Underfitting***
  - Model yang dibuat **terlalu sederhana**
  - *Bias*-nya tinggi
- ***Overfitting***
  - Model yang dibuat **terlalu kompleks**
  - *Variance*-nya tinggi

Sumber: https://www.fastaireference.com/overfitting

03

DATA PREPROCESSING

Pengenalan tentang
data preprocessing

Data Preprocessing

- Data **tidak** selalu **“sempurna”** dan “bersih”
- Data tersebut perlu “dibersihkan” agar menjadi **lebih berkualitas**
- Data yang berkualitas secara umum dapat menghasilkan ***output* yang lebih baik**

DigitalSkola

Data Preprocessing

Beberapa hal yang dilakukan dalam *data preprocessing*:

- *Data* ***cleansing***
- *Data* ***integration***
- *Data* ***dimension reduction***
- *Data* ***transformation***

DigitalSkola

Contoh Data Kotor

ID	Name	Gender	Age	Occupation	Salary
001	Entropy	Male	300	Data Analyst	750000
002	Team	Female	24	Data Engineer	800000
003	Digital	Woman	29	Data Scientist	850000
004		Male	40	Web Developer	850000
002	Team	Female	24	Data Engineer	800000

Warna	Keterangan
	Outlier
	Data tidak konsisten
	Data kosong
	Data duplikat

DigitalSkola

Data Cleansing

Beberapa hal yang dilakukan dalam *data cleansing*:

- Penanganan **data kosong**
  - Numerik: diisi dengan rata-rata atau median
  - Kategoris: diisi dengan modus
- Penanganan **data kotor** (salah atau adanya *outliers*)
  - *Binning*, *regression*, *clustering*, inspeksi manual
- Penanganan **data tidak konsisten**
- Penanganan **data yang sama** (duplikat)

Data Transformation

Beberapa hal yang dilakukan dalam *data transformation*:

- ***Feature encoding***, yaitu mengubah data kategoris menjadi numerik
  - *Label encoding* : hanya untuk **label** (target)
  - *Ordinal encoding* : untuk *feature* yang bersifat **ordinal**
  - *One hot encoding* : untuk *feature* yang bersifat **nominal**

- **Normalization**
  - Mengubah nilai *feature* menjadi **skala tertentu** (umumnya 0-1)

- **Standarization**
  - Mengubah nilai *feature* agar memiliki **rata-rata 0** dan **standar deviasi 1**
  - Gunakan standarisasi jika distribusi datanya normal (*gaussian*)

THANKS

Entropy Team