Course-Net
Empowering IT People

Introduction to Classification

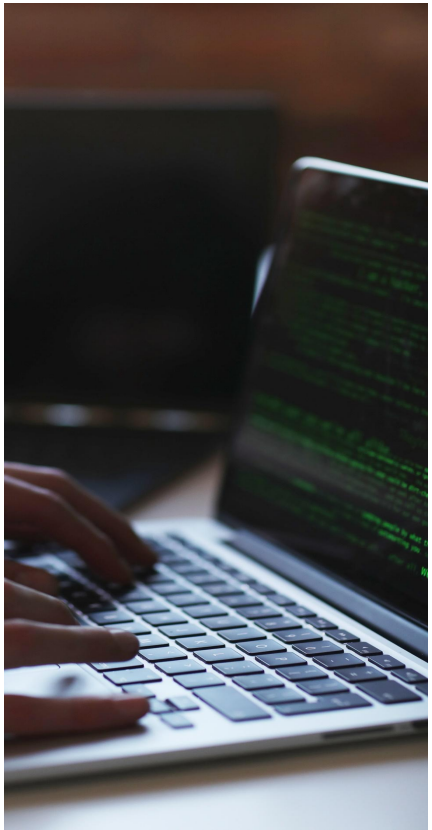
Course-Net
Empowering IT People

#EmpoweringITPeople

Apa itu classification mengapa kita pelajari di data science ?

Course-Net
Empowering IT People

EmpoweringITPeople

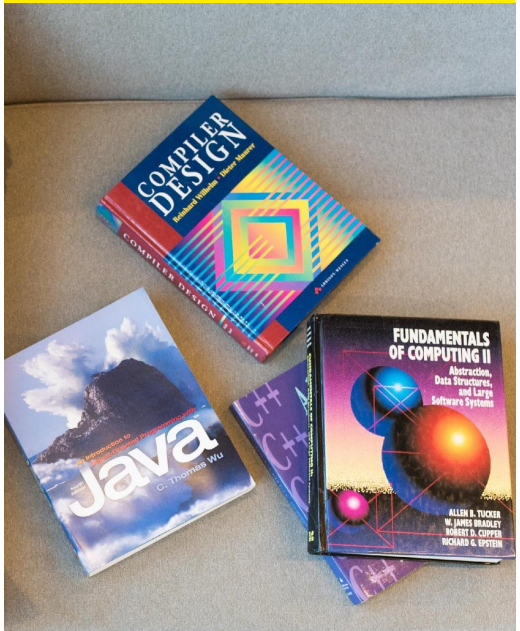


APA YANG AKAN KITA PELAJARI

Classification and applications

- What is Classification
- Classification Intuition
- Classification Algorithm
- Applications of Classification
- Classification Cases

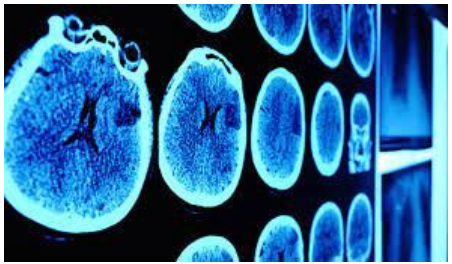
Categorical



Type of Data

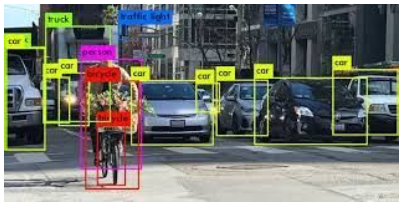
Numeric

Classification Use Case



Cancer Prediction (1,0)

Customer Churn Analysis



Object Detection

Classification Use Case

1. Credit Scoring - Predict Default, Propensity
2. Sentiment Analysis - (Positif, Negatif, Netral)
3.

Classification Type

1. Binary Classification (Cancer prediction, Customer Churn)
2. Multiclass Classification (Object classification)
3. Multilabel Classification (Weather classification)

Binary & Multiclass Classification : setiap data only belong to one class

Multilabel : setiap data bisa terkategori di 2 class

MultiLabel

nama	usia	casa	kredit	punya CC?	…	Beli produk A?	Beli produk B?	Beli produk C?
Adi	34	$3500	$500	Ya	…	1	1	0
Budi	25	$1200	$100	Tidak	…	0	1	0
Citra	17	$500	$300	Ya	…	1	1	1
Doni	45	$2000	$1000	Tidak	…	1	0	1
Eka	15	$700	$20	Ya	…	1	0	0
Feri	30	$1575	$1000	Tidak	…	0	0	0
Andi	50	$1000	$5000	$2000	…	?	?	?

Classification Algorithm

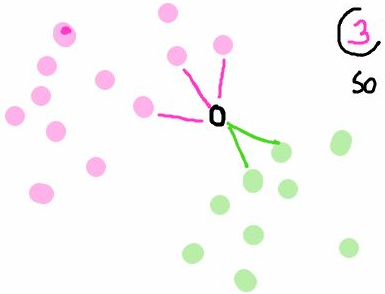
1. KNN
2. SVM
3. Logistic Regression
4. Decision Tree
5. Random Forest
6. XGBoost

Course-Net
Empowering IT People

#EmpoweringITPeople

KNN

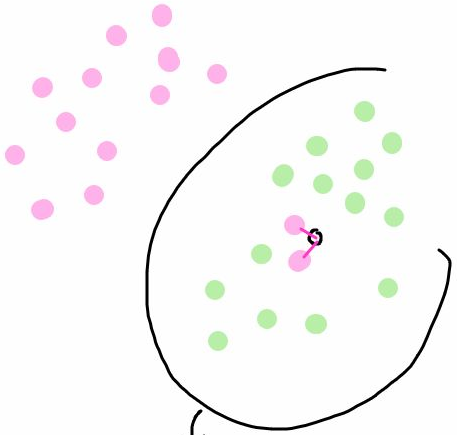
melihat (k) neighbor terdekat



3 vs 2
so terklasifikasi sebagai

k = 5

Kekurangannya : outlier mempengaruhi

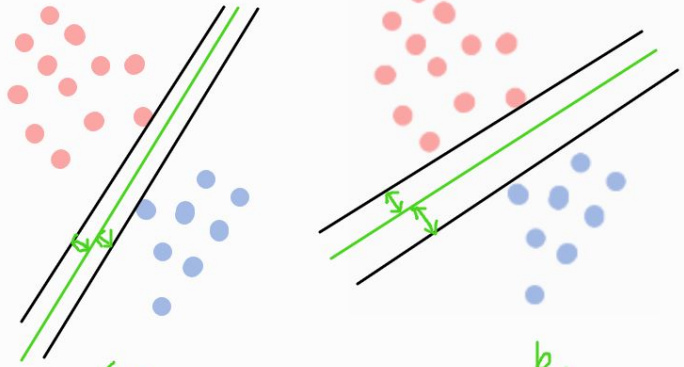


asumsi kita bahwa seharusnya titik yang ada disini terklasify sebagai

Supaya hasilnya bagus, data harus di-transform

SVM

membuat pemisah data yang berbeda class
dengan sebuah hyperplane (garis untuk 2D, bidang untuk 3D)
dengan margin yang besar

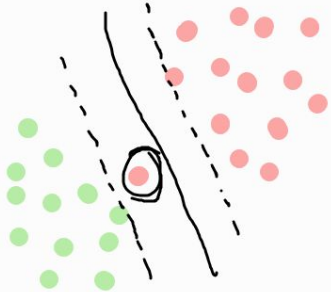


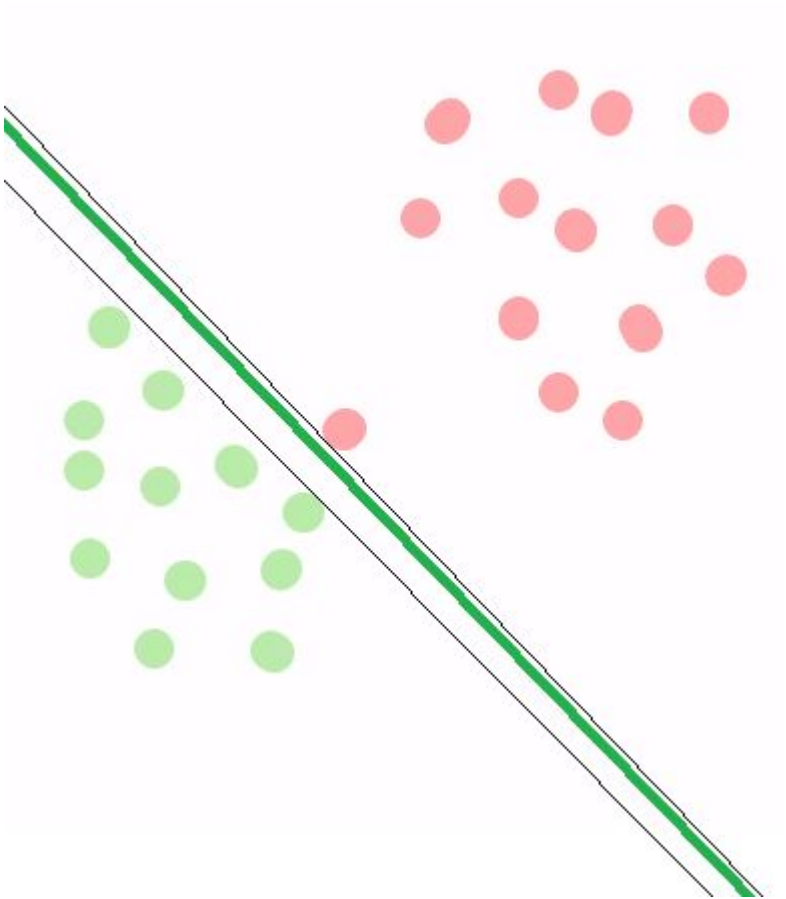
a.

b.

(b) memiliki margin terbesar

Challenge : Bagaimana kalau ada outlier?
Solution : Tuning parameter C (penalty)
Turunkan C untuk abaikan oullier



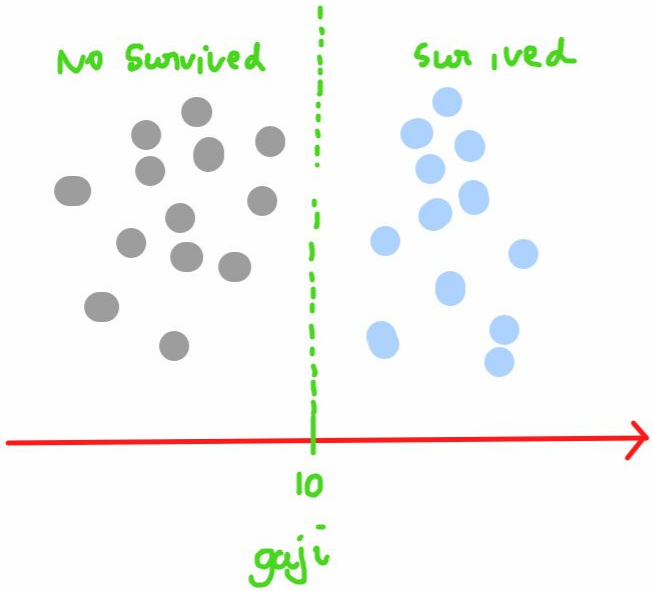


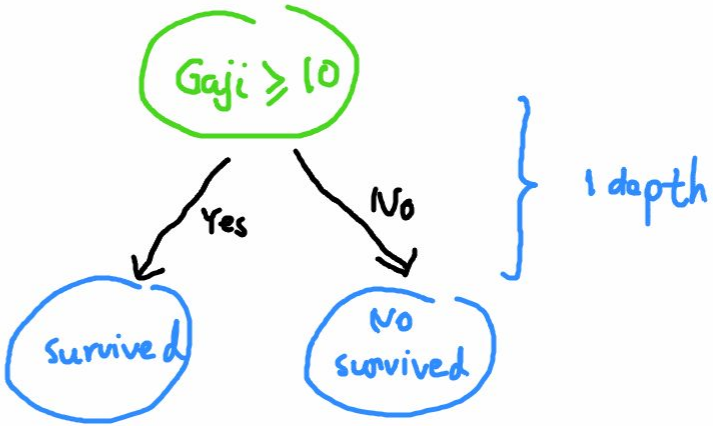
Training & Testing

Course-Net
Empowering IT People

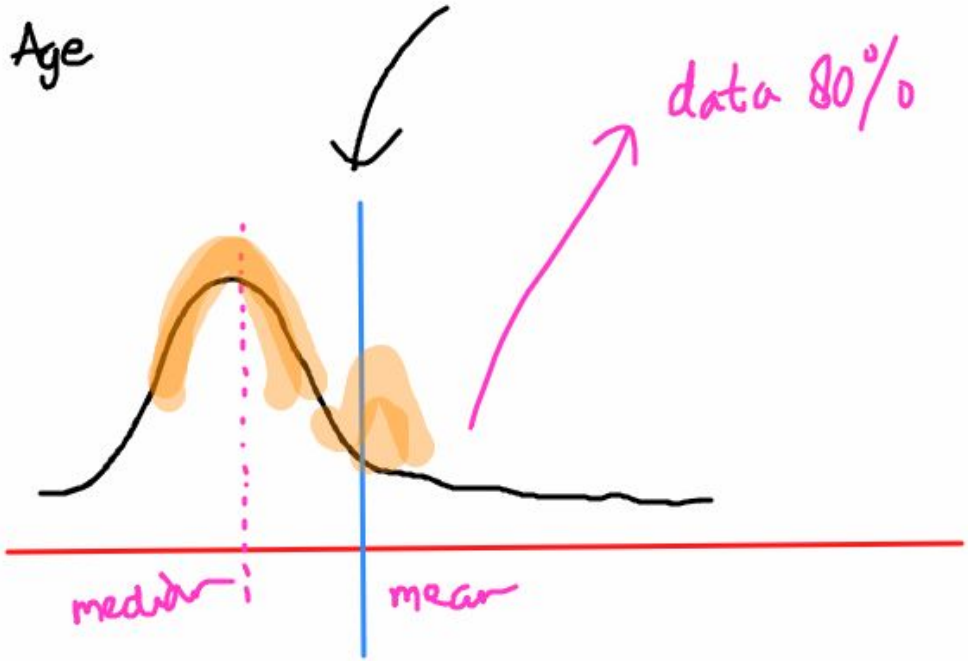
#EmpoweringITPeople

Tree-Based





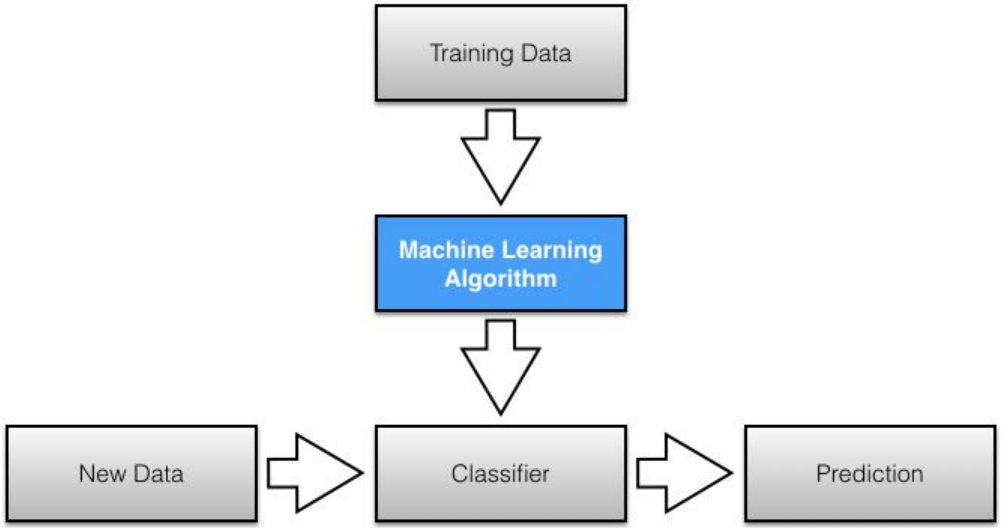
Course-Net
Empowering IT People
EmpoweringITPeople



untuk data yg berdistribusi tidak normal, imputing dengan nilai mean mengubah distribusi data

Logistic Regression

Supervised Learning Process



Course-Net
Empowering IT People

#EmpoweringITPeople

Logistic Regression Intuition

A Glance on Classification Data Example

# Age	# Smokes	# AreaQ	# Alkhol	# Result
35	3	5	4	1
27	20	2	5	1
30	0	5	2	0
28	0	8	1	0
68	4	5	6	1

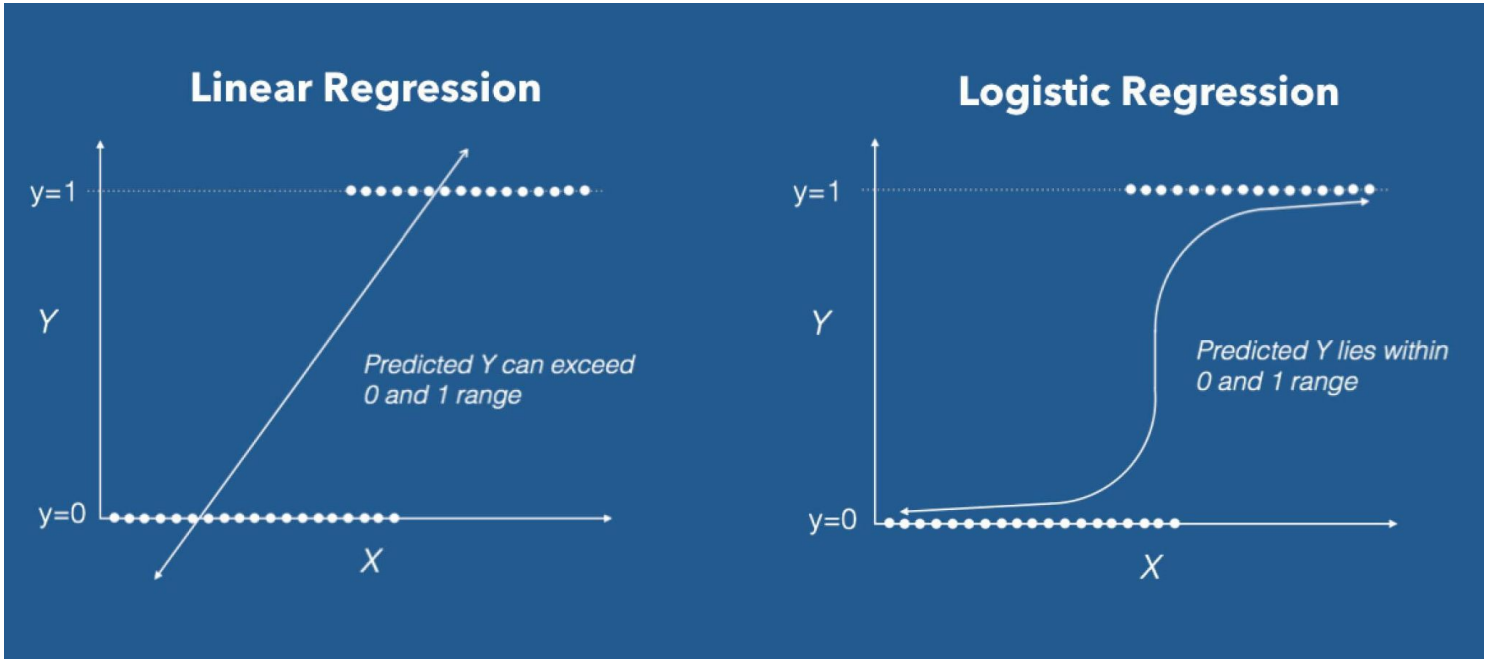
X(Independent) :

- Age
- Smokes
- AreaQ
- Alcohol
- ...

y(Dependent):

- Result

Why Logistic Regression



Logistic Function

$$Y = b_0 + b_1 * X$$

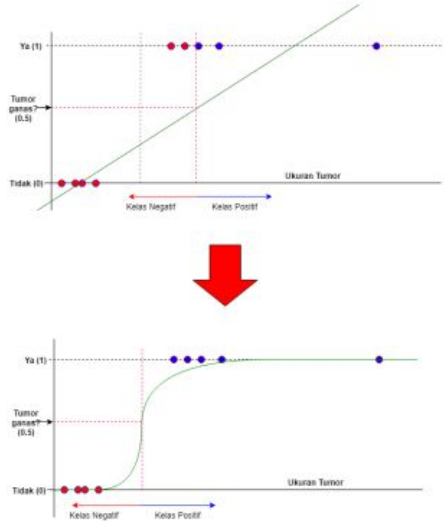
LINEAR FUNCTION

$$P = \frac{1}{1 + e^{-Y}}$$

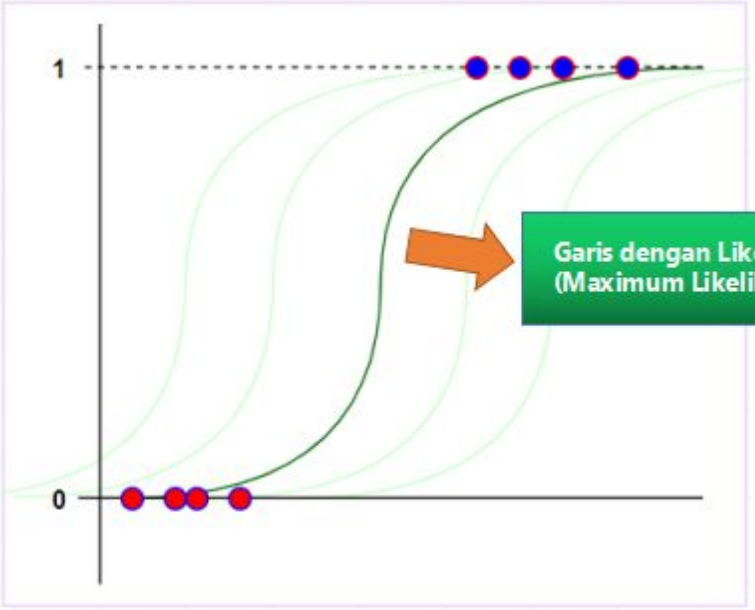
SIGMOID FUNCTION

$$P = \frac{1}{1 + e^{-(b0 + b1 * X)}}$$

LOGISTIC FUNCTION



Maximum Likelihood



Classification Evaluation

Metrics for Classification :

1. Confusion Matrix
2. True Positive Rate (Sensitivity)
3. True Negative Rate (Specificity)
4. False Positive Rate (Fall-Out)
5. False Negative Rate (Miss-Rate)

Confusion Matrix

Confusion Matrix

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

Accuracy

Setelah melakukan training model, kita dapat mengevaluasi performansi model dengan mempertimbangkan beberapa metrik, yang mana salah satunya adalah **akurasi**.

- Metrik **akurasi** merupakan metrik yang paling mudah diterapkan untuk mengukur hasil prediksi suatu model klasifikasi, dimana **akurasi** merupakan perbandingan antara jumlah hasil prediksi tepat dengan jumlah total data yang ada.
- Contoh di samping merupakan **label** dan **hasil prediksi** suatu model klasifikasi. Dari 6 sampel yang ada, model berhasil memprediksi 5 sampel data dengan benar sehingga akurasinya adalah 5/6 atau sekitar 83%.

spam?	prediksi
ya	ya
tidak	tidak
ya	ya
tidak	ya
tidak	tidak
tidak	tidak

Confusion Matrix

Selain itu, terdapat ***confusion matrix*** yang merupakan matriks yang berisikan 4 semua kemungkinan hasil prediksi model, diantaranya

- **True Positive (TP)** : sampel memiliki label True dan berhasil diprediksi model sebagai True
- **True Negative (TN)** : sampel memiliki label False dan berhasil diprediksi model sebagai False
- **False Positive (FP)** : sampel sebenarnya memiliki label False namun diprediksi model sebagai True
- **False Negative (FN)** : sampel sebenarnya memiliki label True namun diprediksi model sebagai False

Untuk mempermudah dalam memahaminya, *confusion matrix* biasanya divisualisasikan sebagai berikut.

		Prediksi	
		False	True
Target	False	TN	FP
	True	FN	TP

Accuracy

Dari confusion matrix tersebut, kita dapat menghitung ulang akurasi dengan rumus berikut.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Selain akurasi, kita juga dapat menghitung beberapa metrik lainnya diantaranya sebagai berikut.

1. True Positive Rate (TPR)
2. True Negative Rate (TNR)
3. False Negative Rate (FNR)
4. False Positive Rate (FPR)

Precision, Recall dan F1-Score

Dengan menggunakan beberapa nilai dari confusion matrix, kita juga dapat menghitung metrik ***precision***, ***recall***, dan ***F1-score*** dengan rumus berikut.

$$precision = \frac{TP}{TP + FP} \qquad recall = \frac{TP}{TP + FN} \qquad F1\ Score = 2 \times \frac{precison \times recall}{precison + recall}$$

Ketiga metrik ini sering dipakai jika fitur label pada data tidak seimbang (***imbalance label***) dimana :

1. ***Precision*** merupakan rasio antara prediksi benar positif dibandingkan dengan keseluruhan yang diprediksi positif.
   Sebagai contoh, precision dapat menjawab pertanyaan seperti
   “Berapa persen email yang benar-benar spam dari keseluruhan email yang diprediksi sebagai spam? ”.

2. ***Recall*** (***sensitivity***) merupakan rasio antara benar positif dibandingkan dengan keseluruhan data yang benar positif.
   Jika diambil contoh, metrik ini dapat menjawab pertanyaan seperti “Berapa persen email yang diprediksi spam dibandingkan dengan keseluruhan email yang sebenarnya adalah spam?”

3. **F1 Score** merupakan nilai harmonik antara recall dan precision.

Precision, Recall dan F1-Score

Dengan menggunakan beberapa nilai dari confusion matrix, kita juga dapat menghitung metrik ***precision***, ***recall***, dan ***F1-score*** dengan rumus berikut.

$$precision = \frac{TP}{TP + FP} \qquad recall = \frac{TP}{TP + FN} \qquad F1\ Score = 2 \times \frac{precison \times recall}{precison + recall}$$

Accuracy = 90%

Prediksi (**customer potensial**,customer tidak potensial)

100 900

Precision = 40%

Precision, Recall dan F1-Score

Dengan menggunakan beberapa nilai dari confusion matrix, kita juga dapat menghitung metrik ***precision***, ***recall***, dan ***F1-score*** dengan rumus berikut.

$$precision = \frac{TP}{TP + FP} \qquad recall = \frac{TP}{TP + FN} \qquad F1\ Score = 2 \times \frac{precison \times recall}{precison + recall}$$

Accuracy = 90%

Prediksi (pasien yang covid,**pasien yang tidak covid**)

100 900

Recall = 50%

Contoh (1)

covid	prediksi
ya	ya
tidak	tidak
ya	ya
tidak	ya
tidak	tidak
tidak	tidak

- **True Positive (TP)** : sampel memiliki label True dan berhasil diprediksi model sebagai True
- **True Negative (TN)** : sampel memiliki label False dan berhasil diprediksi model sebagai False
- **False Positive (FP)** : sampel sebenarnya memiliki label False namun diprediksi model sebagai True
- **False Negative (FN)** : sampel sebenarnya memiliki label True namun diprediksi model sebagai False

- **TP = 2**
- **TN = 3**
- **FP = 1**
- **FN = 0**

- **Accuracy = (TP+TN)/(TP+TN+FP+FN) = 5/6**

- **Precision = TP/(TP+1)=2/(2)=0.63**

- **Recall = TP/(TP+FN)=2/(2+0)=1**
  (penyakit)

- **F1-Score =**

Contoh (1)

Default debitur	prediksi
ya	ya
tidak	tidak
ya	ya
tidak	ya
tidak	tidak
tidak	tidak

- **True Positive (TP)** : sampel memiliki label True dan berhasil diprediksi model sebagai True
- **True Negative (TN)** : sampel memiliki label False dan berhasil diprediksi model sebagai False
- **False Positive (FP)** : sampel sebenarnya memiliki label False namun diprediksi model sebagai True
- **False Negative (FN)** : sampel sebenarnya memiliki label True namun diprediksi model sebagai False

- **TP = 2**
- **TN = 3**
- **FP = 1**
- **FN = 0**

- **Accuracy = (TP+TN)/(TP+TN+FP+FN) = 5/6**

- **Precision = TP/(TP+1)=2/(2)=0.63**

- **Recall = TP/(TP+FN)=2/(2+0)=1**
  (penyakit)

- **F1-Score =**

Contoh (2)

spam	prediksi
tidak	ya
ya	ya
ya	ya
ya	tidak
ya	ya
tidak	tidak

- **True Positive (TP)** : sampel memiliki label True dan berhasil diprediksi model sebagai True
- **True Negative (TN)** : sampel memiliki label False dan berhasil diprediksi model sebagai False
- **False Positive (FP)** : sampel sebenarnya memiliki label False namun diprediksi model sebagai True
- **False Negative (FN)** : sampel sebenarnya memiliki label True namun diprediksi model sebagai False

- **TP = 2**
- **TN = 1**
- **FP =**
- **FN =**

- **Accuracy =**

- **Precision =**

- **Recall =**

- **F1-Score =**