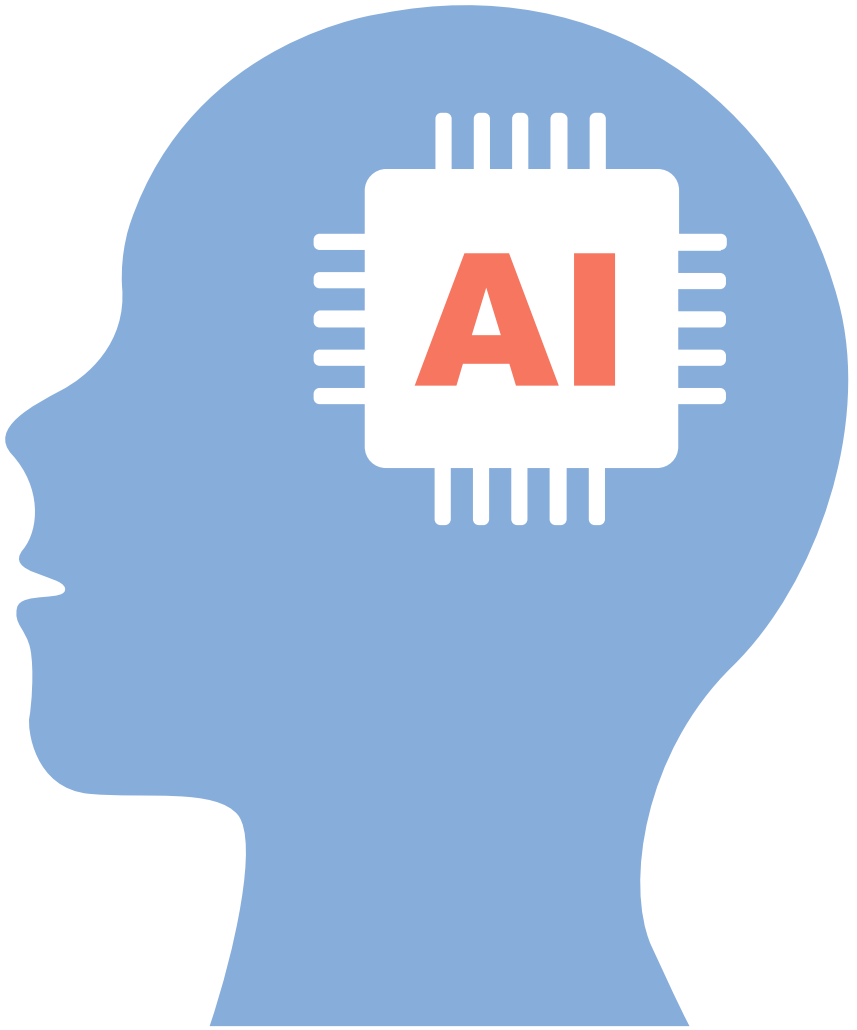
Kecerdasan Buatan

Teknik Pengujian Model (Confusion Matrix)

Lalu Mutawalli, M.Kom., M.I.Kom

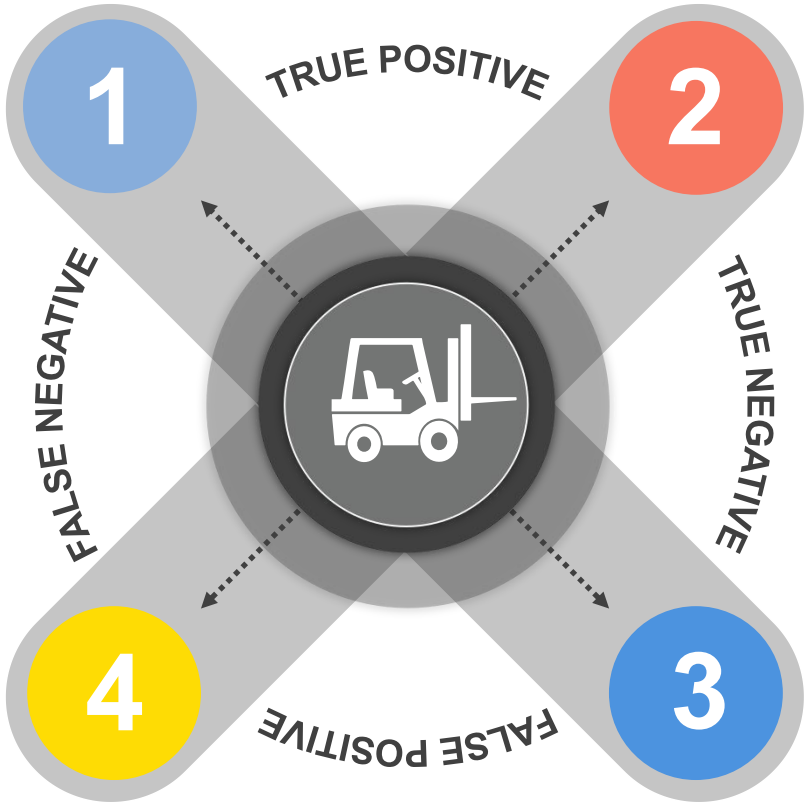
Cofusion Matrix

Confusion matrix adalah suatu metode yang biasanya digunakan untuk melakukan perhitungan akurasi pada konsep data mining atau Sistem Pendukung Keputusan.



Representasi Hasil Proses Klasifikasi

1. True Positive
Merupakan data positif yang terdeteksi benar.



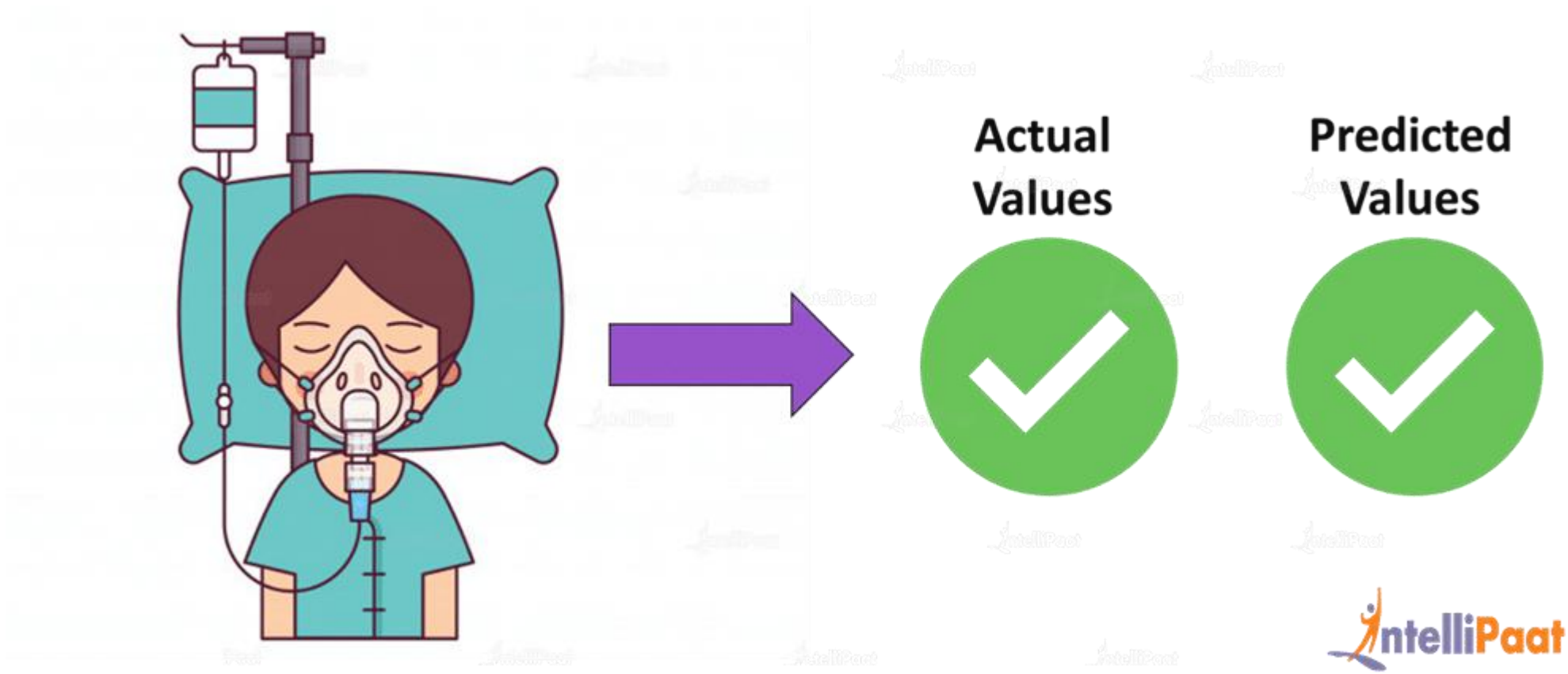
2. True Negative
Merupakan jumlah data negatif yang terdeteksi dengan benar

4, False Negative
Merupakan data positif namun terdeteksi sebagai data negatif

3. False Positive
Merupakan data negatif namun terdeteksi sebagai data positif

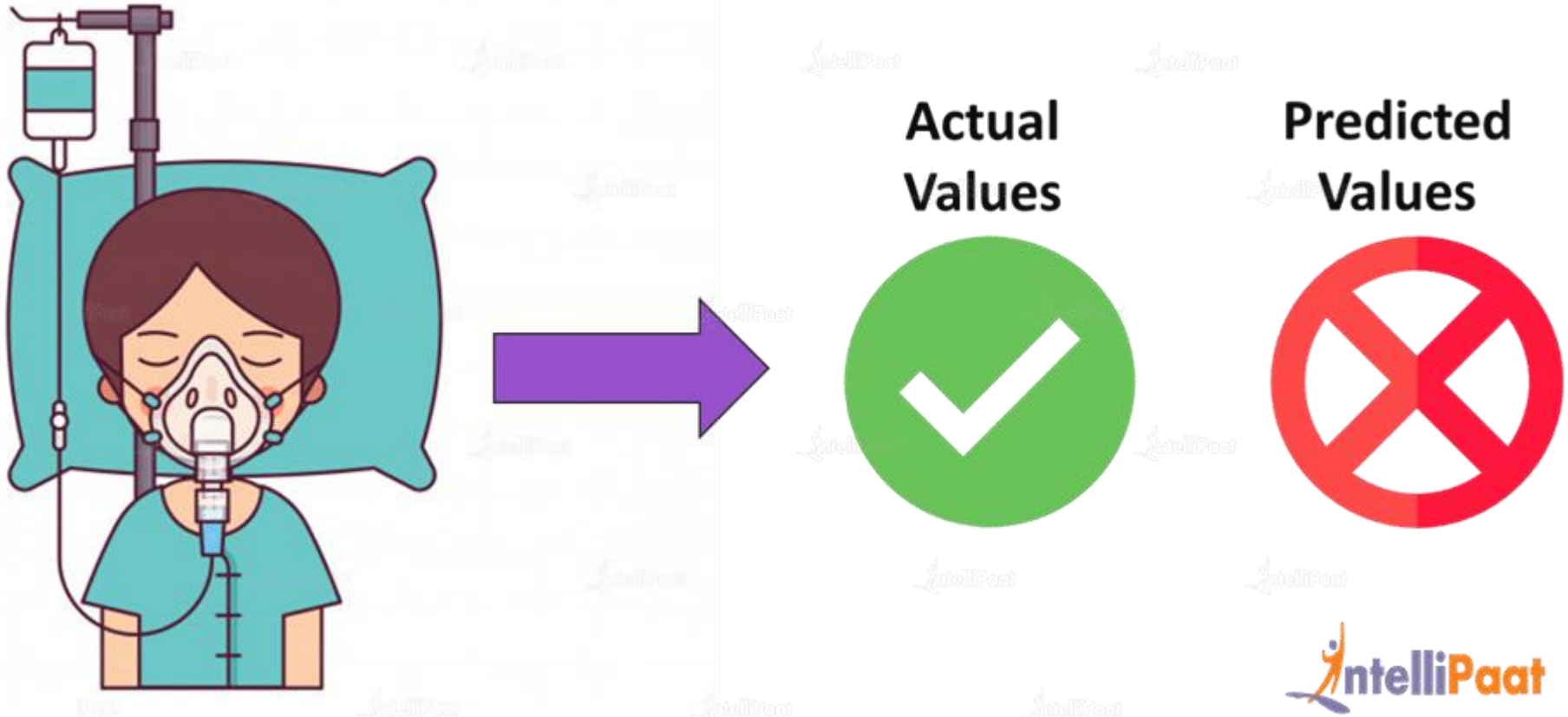
True Positive

True positive is nothing but the case where the actual value as well as the predicted value are true. The patient has been diagnosed with cancer, and the model also predicted that the patient had cancer.



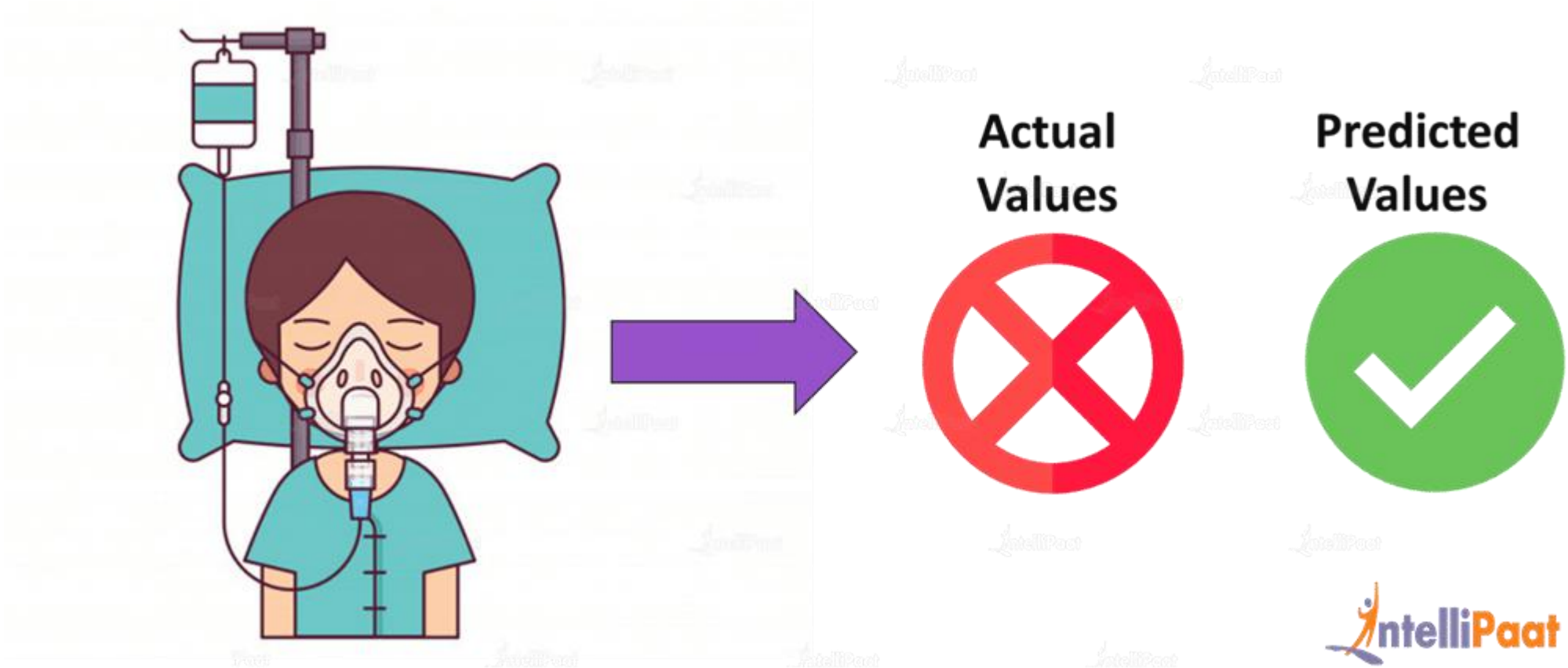
False Negative

In false negative, the actual value is true, but the predicted value is false, which means that the patient has cancer, but the model predicted that the patient did not have cancer.



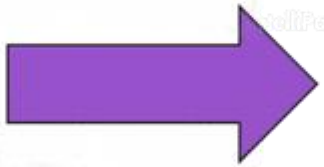
False Positive

This is the case where the predicted value is true, but the actual value is false. Here, the model predicted that the patient had cancer, but in reality, the patient doesn't have cancer. This is also known as Type 1 Error.



True Negative

This is the case where the actual value is false and the predicted value is also false. In other words, the patient is not diagnosed with cancer and our model predicted that the patient did not have cancer.



IntelliPaat

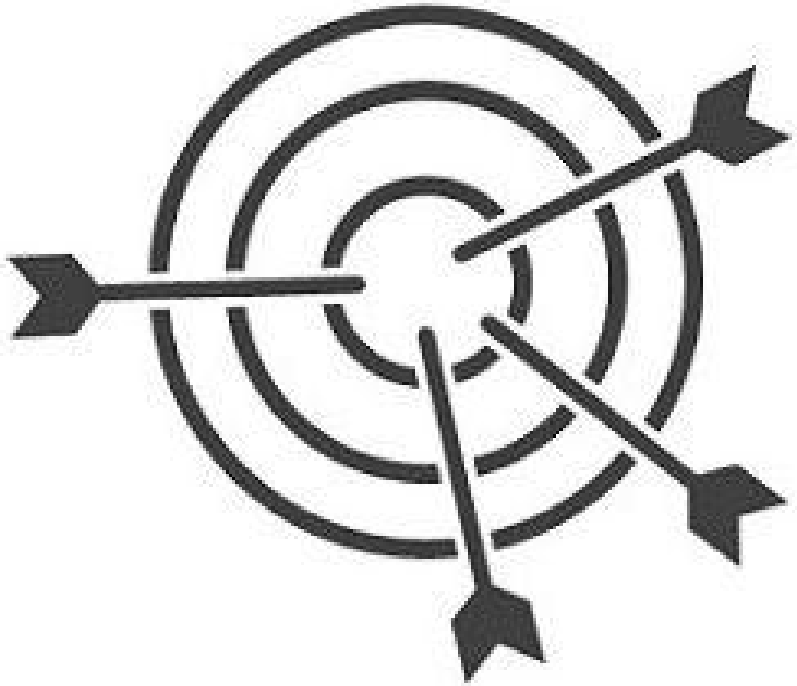
Confusion Matrix and ROC Curve

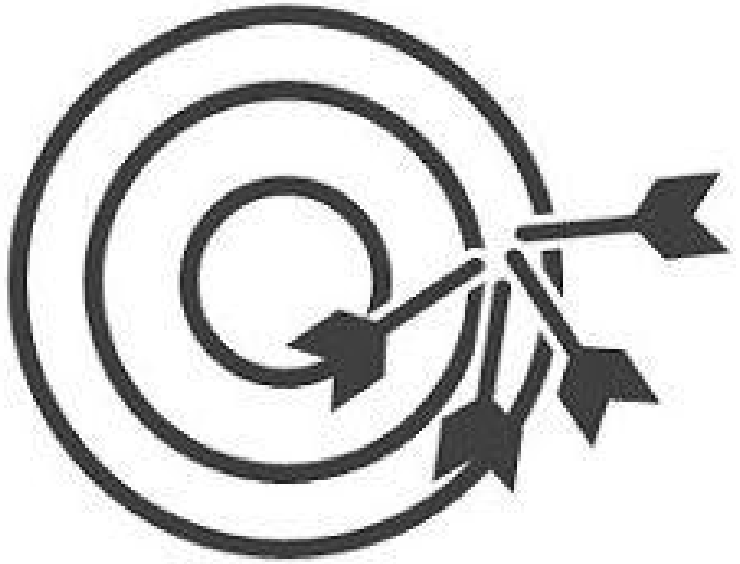
		Predicted Class	
		No	Yes
Observed Class	No	TN	FP
	Yes	FN	TP

TN	True Negative
FP	False Positive
FN	False Negative
TP	True Positive

Model Performance

Accuracy	= (TN+TP)/(TN+FP+FN+TP)
Precision	= TP/(FP+TP)
Sensitivity	= TP/(TP+FN)
Specificity	= TN/(TN+FP)





Akurasi v Presisi

Akurasi adalah “keadaan benar”, presisi adalah “keadaan pasti”. Pada saat mengambil pengukuran, dua hal ini selalu diperhitungkan, karena penting pada berbagai bidang, yang digunakan pada sains, statistika, penelitian, dan engineering.

Accuracy = (TP+TN)/(TP+FP+TN+FN)

Precision = TP/(TP+FP) = TP/ (Predicted Yes)

	Akurasi	Presisi
Pengertian	Akurasi mengacu pada level kesepakatan antara pengukuran actual dan pengukuran absolut	Presisi mengartikan level keberagaman yang terletak pada nilai beberapa pengukuran dari factor yang sama
Menggambarkan	Seberapa dekat hasil dengan nilai standart	Seberapa dekat hasil dengan yang lain
Derajat	Derajat kecocokan	Derajat reprodusibilitas
Faktor	Faktor tunggal	Banyak faktor
Pengukuran dari	Perkiraan statikal	Keberagaman statistikal
Terkait dengan	Kesalahan sistematik	Kesalahan acak

Sensitivity or Recall

Merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. Recall menjawab pertanyaan “Berapa persen yang diprediksi dibandingkan keseluruhan ”.

Recall = (TP) / (TP + FN)

pada contoh kasus di atas Recall = 4/(4+1) = 4/5 =80%.

F1 Score

F1 Score merupakan perbandingan rata-rata presisi dan recall yang dibobotkan

F1 Score = 2 * (Recall*Precission) / (Recall + Precission)

dalam kasus di atas, F1 Score = 2* (80%*67%) / (80% + 67%) = 72,93%

Confusion Matrix dengan Python

```
import numpy as np

cm = np.array(
[[5825,    1,   49,   23,    7,   46,   30,   12,   21,   26],
 [   1, 6654,   48,   25,   10,   32,   19,   62,  111,   10],
 [   2,   20, 5561,   69,   13,   10,    2,   45,   18,    2],
 [   6,   26,   99, 5786,    5,  111,    1,   41,  110,   79],
 [   4,   10,   43,    6, 5533,   32,   11,   53,   34,   79],
 [   3,    1,    2,   56,    0, 4954,   23,    0,   12,    5],
 [  31,    4,   42,   22,   45,  103, 5806,    3,   34,    3],
 [   0,    4,   30,   29,    5,    6,    0, 5817,    2,   28],
 [  35,    6,   63,   58,    8,   59,   26,   13, 5394,   24],
 [  16,   16,   21,   57,  216,   68,    0,  219,  115, 5693]])
```

Confusion Matrix dengan Python

```
def precision(label, confusion_matrix):
    col = confusion_matrix[:, label]
    return confusion_matrix[label, label] / col.sum()

def recall(label, confusion_matrix):
    row = confusion_matrix[label, :]
    return confusion_matrix[label, label] / row.sum()

def precision_macro_average(confusion_matrix):
    rows, columns = confusion_matrix.shape
    sum_of_precisions = 0
    for label in range(rows):
        sum_of_precisions += precision(label, confusion_matrix)
    return sum_of_precisions / rows

def recall_macro_average(confusion_matrix):
    rows, columns = confusion_matrix.shape
    sum_of_recalls = 0
    for label in range(columns):
        sum_of_recalls += recall(label, confusion_matrix)
    return sum_of_recalls / columns
```

Confusion Matrix dengan Python

```
print("label precision recall")
for label in range(10):
    print(f"{label:5d} {precision(label, cm):9.3f} {recall(label, cm):6.3f}")
```

Hasil

```
label precision recall
    0     0.983  0.964
    1     0.987  0.954
    2     0.933  0.968
    3     0.944  0.924
    4     0.947  0.953
    5     0.914  0.980
    6     0.981  0.953
    7     0.928  0.982
    8     0.922  0.949
    9     0.957  0.887
```

Confusion Matrix dengan Python

```
rint("precision total:", precision_macro_average(cm))

print("recall total:", recall_macro_average(cm))
```

Hasil

```
precision total: 0.949688556405
recall total: 0.951453154788
```

Confusion Matrix dengan Python

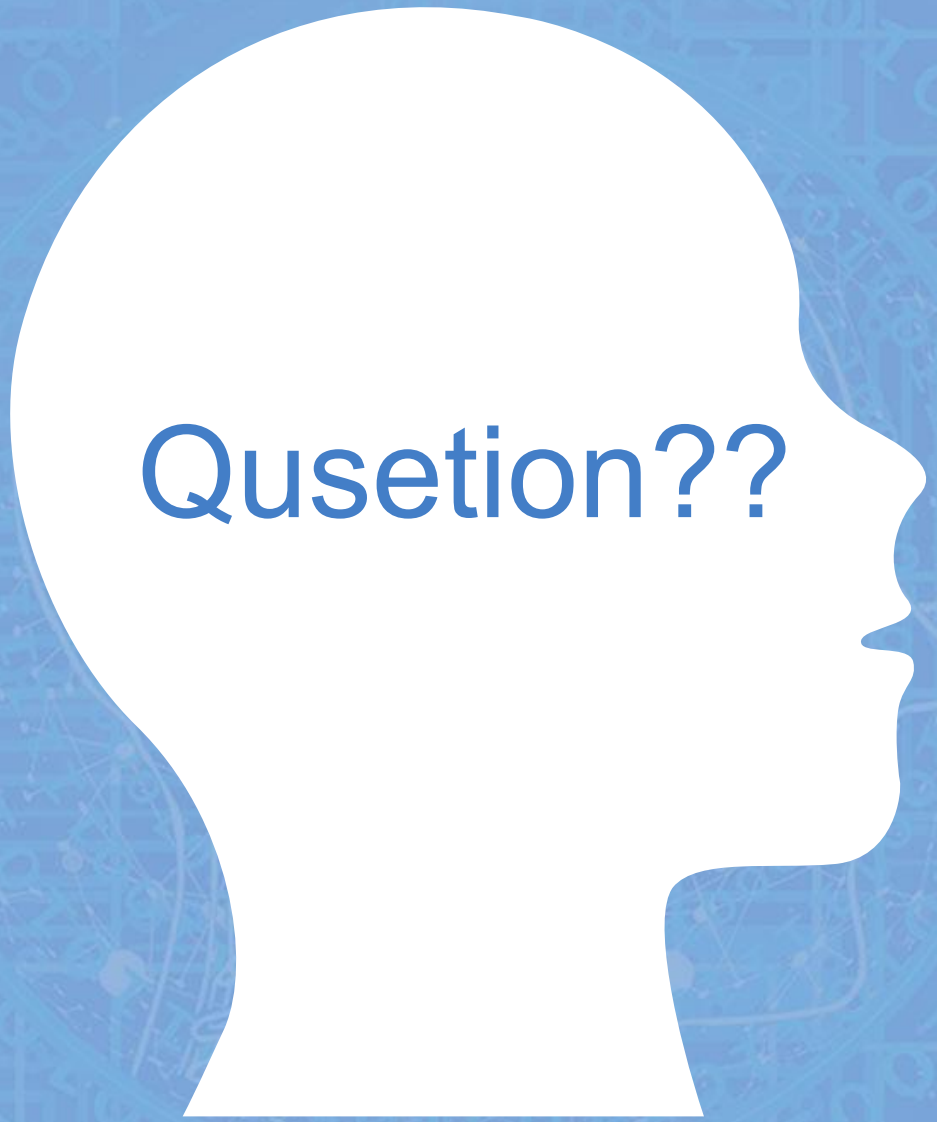
```
def accuracy(confusion_matrix):
    diagonal_sum = confusion_matrix.trace()
    sum_of_all_elements = confusion_matrix.sum()
    return diagonal_sum / sum_of_all_elements
```

accuracy(cm)

Output::

0.9503833333333336


Thank You


Qusetion??