

การวิเคราะห์และประเมินผล โมเดล Classification



ไม่ถูกห่วย ไม่เป็นไร
แต่เจ็บใจ แหม..เจียดไป
เพียงนิดเดียว

นาวาโท ดร.บัญชา ช่วยสี

หัวหน้าฝึกและจำลองยุทธ์ไซเบอร์ ศูนย์ไซเบอร์
กรมการสื่อสารและเทคโนโลยีสารสนเทศทหารเรือ

ประสบการณ์ที่เกี่ยวข้อง

- กองปฏิบัติการไซเบอร์ ศูนย์ไซเบอร์
กรมการสื่อสารและเทคโนโลยีสารสนเทศทหารเรือ (สสท.ทร.)
- PhD การจัดการเทคโนโลยีสารสนเทศ คณะวิศวกรรมศาสตร์ มหิดล
- นักวิจัยสมาคมอุตสาหกรรมดิจิทัลไทย มหิดล
- วิทยากรหลักสูตรนักวิทยาศาสตร์ข้อมูล มหิดล



Data Analytics

วิทยากรร่วม : มหิดล



Data Engineering

วิทยากรร่วม : มหิดล



Machine Learning

ที่ปรึกษางานวิจัยด้าน ML



01 Parameters

Parameters เป็นค่าที่โมเดลเรียนรู้จากข้อมูลฝึกอบรม (Training Data) และถูกใช้ในการทำนายผลลัพธ์ (Predictions) โดยอัตโนมัติ

02 การวิเคราะห์ผลการพยากรณ์

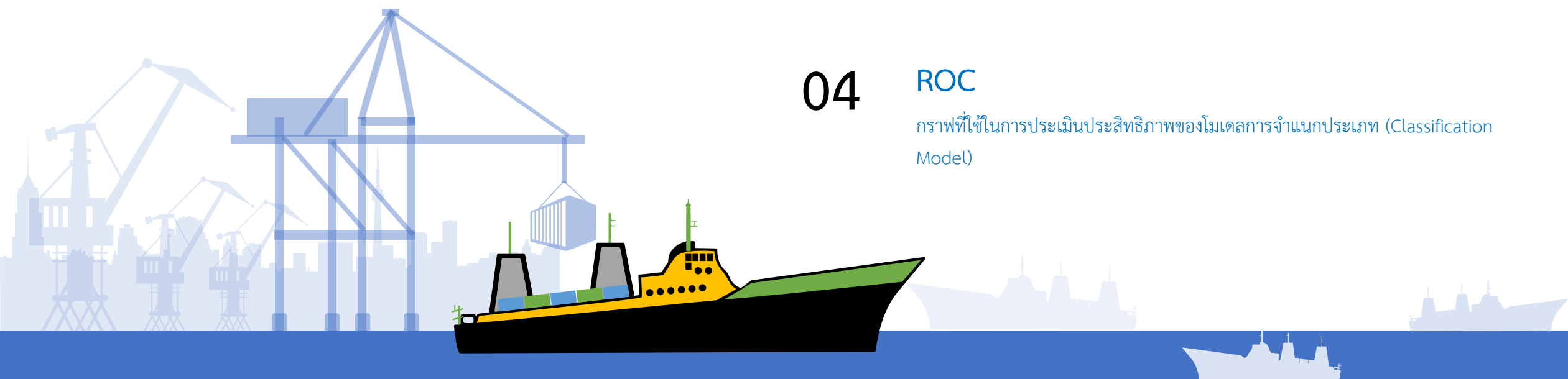
ขั้นตอนสำคัญหลังจากที่โมเดล Machine Learning ทำการทำนายผลลัพธ์ เพื่อประเมินความแม่นยำและประสิทธิภาพของโมเดล

03 Confusion Matrix

ตารางที่ใช้ในการประเมินประสิทธิภาพของโมเดลการจำแนก (Classification Model)

04 ROC

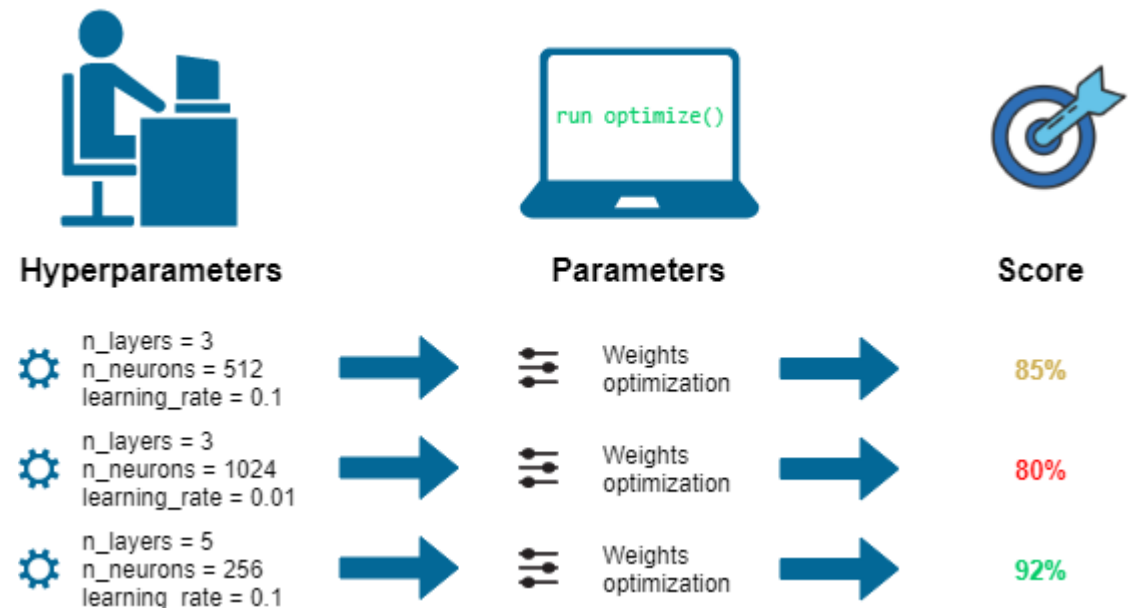
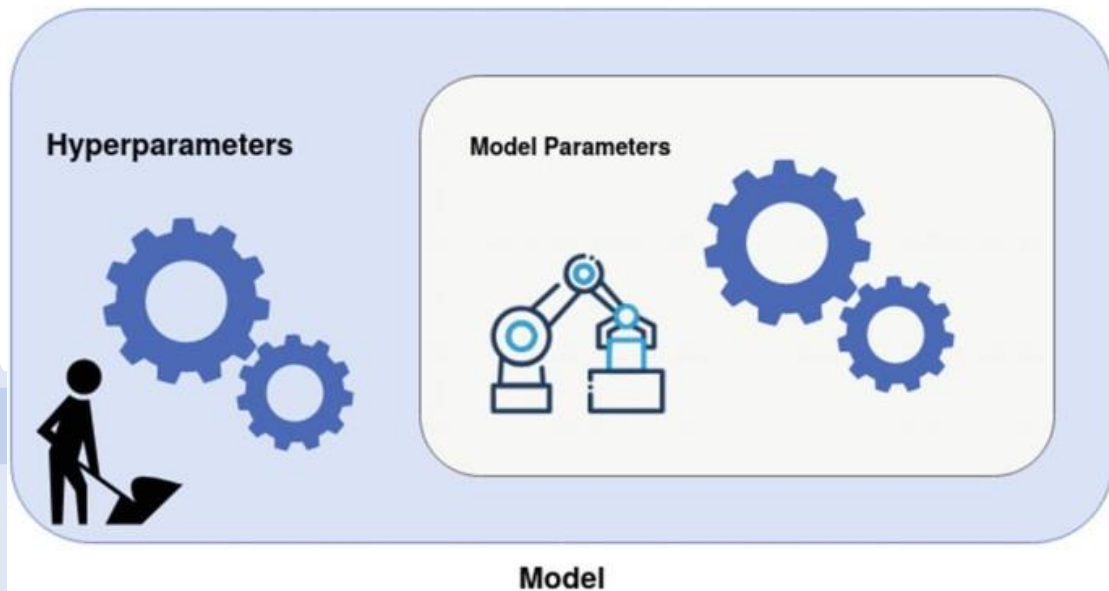
กราฟที่ใช้ในการประเมินประสิทธิภาพของโมเดลการจำแนกประเภท (Classification Model)



การสร้างโมเดล ประกอบไปด้วยพารามิเตอร์ที่แตกต่างกัน 2 ชนิด ได้แก่

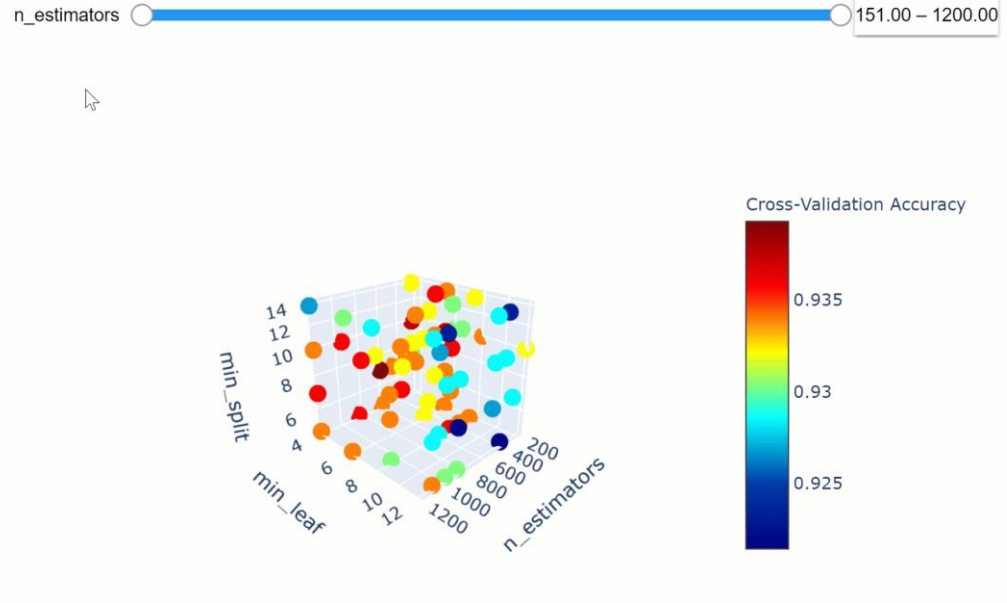
1. Model Parameters คือ พารามิเตอร์ที่ได้มาระหว่างขั้นตอนการเรียนรู้ข้อมูลของโมเดล (Model Training) เช่น ค่า Weights ที่ใช้ใน Neural Network หรือค่า Coefficients ที่ได้จากการทำ Linear Regression เป็นต้น

2. Hyperparameters คือ พารามิเตอร์ต่าง ๆ ที่ผู้ใช้สามารถกำหนดเองได้ก่อนที่โมเดลจะทำการเรียนรู้ เช่น ค่า Learning Rate ที่ใช้ในการควบคุมว่าใน 1 Step ของการเรียนรู้จะปรับค่า Weights ของ Neural Network อย่างไร หรือค่า n_estimators ซึ่งกำหนดจำนวนต้นไม้สำหรับการสร้างโมเดล Random Forest เป็นต้น



Hyperparameter สำคัญอย่างไร?

- ✓ ควบคุมวิธีที่โมเดลเรียนรู้
- ✓ ปรับให้โมเดลทำงานได้ดีที่สุด
- ✓ ป้องกันปัญหา Overfitting / Underfitting
- ✓ ช่วยให้โมเดลเรียนรู้ได้เร็วขึ้น



Hyperparameter Tuning (การปรับค่าที่เหมาะสม)

Hyperparameter ไม่สามารถเรียนรู้เองได้ ต้องปรับให้เหมาะสมโดยใช้วิธีการต่อไปนี้



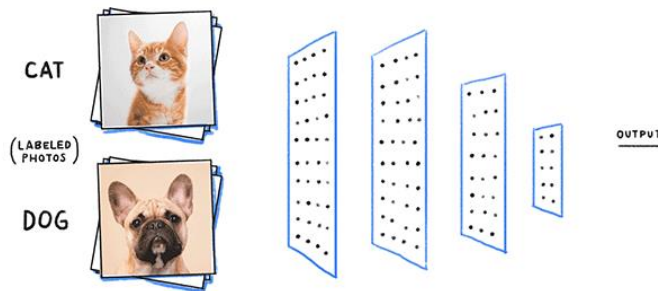
1. Grid Search

- ✓ ทดลองค่าต่าง ๆ ทุกค่าที่เป็นไปได้
- ✗ ใช้เวลานานมากถ้าค่าที่เป็นไปได้มีเยอะ



2. Random Search

- ✓ สุ่มค่ามาลอง แทนที่จะไล่ค่าทั้งหมด
- ✓ เร็วกว่า Grid Search
- ✗ อาจไม่ได้ค่าที่ดีที่สุด



ตัวอย่าง Hyperparameter ในโมเดลต่าง ๆ



1. Decision Tree

- Max Depth → กำหนดความลึกสูงสุดของต้นไม้
- Min Samples Split → จำนวนตัวอย่างขั้นต่ำที่ต้องมีเพื่อแยกโหนด
- Criterion → วิธีการคำนวณค่า Gini หรือ Entropy



2. Random Forest

- Number of Trees (n_estimators) → จำนวนต้นไม้ในป่า
- Max Features → จำนวนคุณสมบัติที่ใช้แยกแต่ละโหนด



3. Neural Networks (Deep Learning)

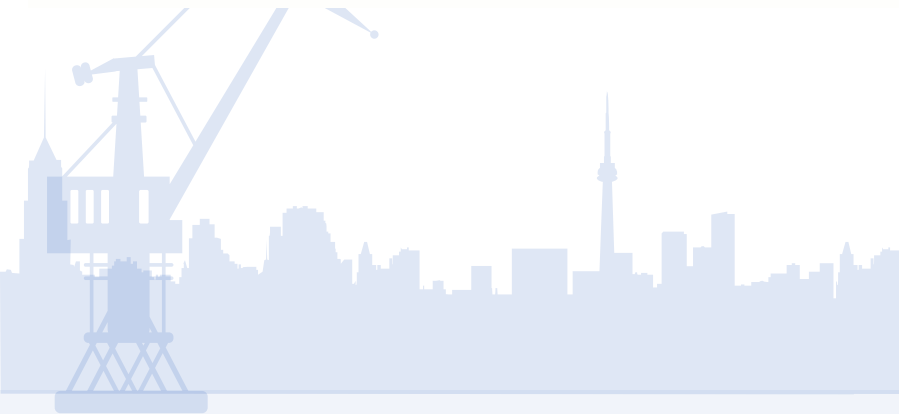
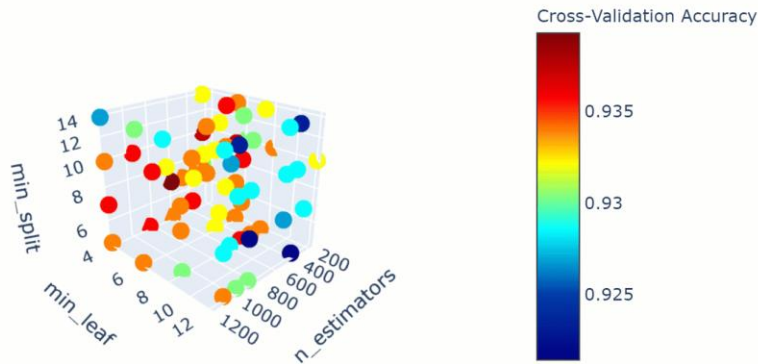
- Learning Rate → ควบคุมอัตราการปรับค่าของน้ำหนัก
- Batch Size → จำนวนตัวอย่างข้อมูลที่ใช้ในการอัปเดตพารามิเตอร์แต่ละครั้ง
- Number of Layers & Neurons → กำหนดโครงสร้างของโครงข่าย

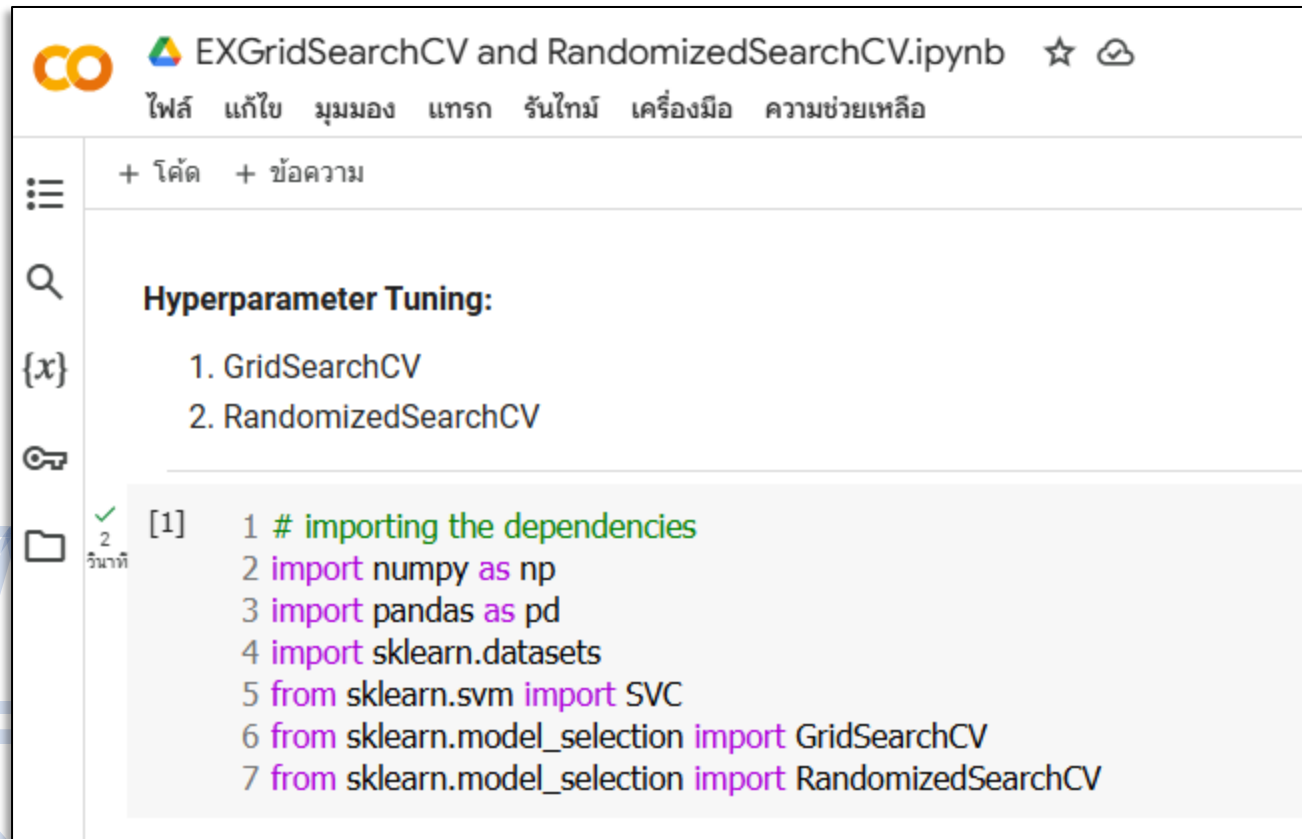


4. Support Vector Machine (SVM)

- Kernel Type → เลือกรูปแบบของ Kernel เช่น Linear, RBF
- C (Regularization Parameter) → ควบคุมความซับซ้อนของโมเดล
- Gamma → ควบคุมอิทธิพลของตัวอย่างข้อมูลแต่ละตัว

n_estimators  151.00 – 1200.00





The image shows a Jupyter Notebook interface for a file named "EXGridSearchCV and RandomizedSearchCV.ipynb". The top bar includes the Colab logo, the file name, and icons for star and share. Below this is a navigation bar with links: "ไฟล์" (Files), "แก้ไข" (Edit), "มุมมอง" (View), "แทรก" (Insert), "รันใหม่" (Run), "เครื่องมือ" (Tools), and "ความช่วยเหลือ" (Help). The left sidebar contains icons for a menu, search, variables, keys, and a file explorer showing a folder named "วันอาทิตย์". The main content area has a section titled "Hyperparameter Tuning:" with a list: "1. GridSearchCV" and "2. RandomizedSearchCV". Below this is a code cell labeled "[1]" with a green checkmark and the number "2" in the margin, containing the following Python code:

```
1 # importing the dependencies
2 import numpy as np
3 import pandas as pd
4 import sklearn.datasets
5 from sklearn.svm import SVC
6 from sklearn.model_selection import GridSearchCV
7 from sklearn.model_selection import RandomizedSearchCV
```

EXGridSearchCV and RandomizedSearchCV.ipynb

การวิเคราะห์ผลการพยากรณ์

- ☐ ไม่ถูกห่วย (ถูก / ไม่ถูก) : Classification
- ☐ เฉียด (เฉียดก็แต้ม) : Regression



การวิเคราะห์ผลการพยากรณ์

Classification



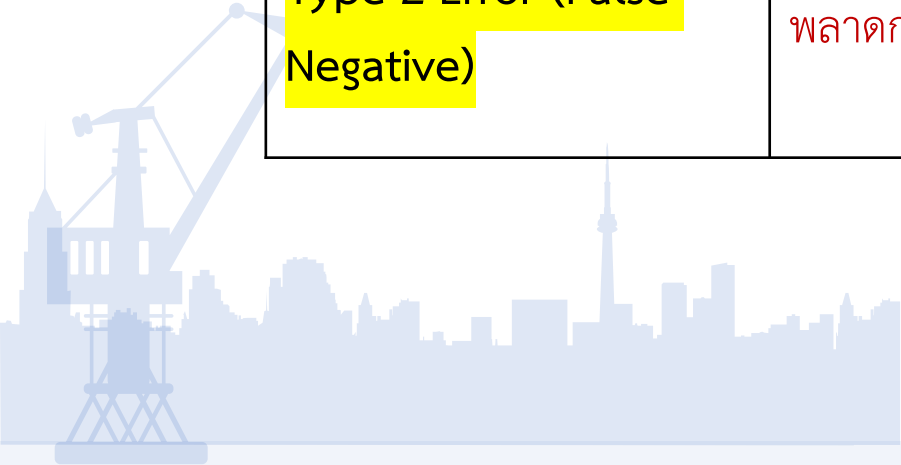
งานวิจัย / การวินิจฉัย มีข้อผิดพลาดอยู่ 2 ประเภท

- ❑ ข้อผิดพลาดประเภทที่ 1 (Type I Error)
เมื่อเราปฏิเสธสมมติฐาน ทั้งที่เป็นจริง
พบหลักฐานของผลกระทบ ที่ไม่มีอยู่จริง
(สงสัยว่าเป็นของโจร เลยไม่ซื้อ -- อดได้ของถูก)
- ❑ ข้อผิดพลาดประเภทที่ 2 (Type II Error)
เมื่อเราไม่ปฏิเสธสมมติฐานทั้งที่เป็นเท็จ
พลาดผลกระทบที่มีอยู่จริง
(ไม่สงสัยว่าเป็นของโจร เลยซื้อ -- จำคุก)



Type 1 และ Type 2 Error

ประเภท	คำอธิบาย	ตัวอย่าง	ผลกระทบ
Type 1 Error (False Positive)	แจ้งเตือนผิดพลาด	ทำนายว่าผู้ป่วย“ท้อง” แต่จริง ๆ ไม่ท้อง	ค่าใช้จ่ายเกินจำเป็น, แจ้งเตือนผิด
Type 2 Error (False Negative)	พลาดการตรวจจับ	ทำนายว่าผู้ป่วยไม่ท้อง แต่จริง ๆ ท้อง	อันตรายร้ายแรง, ความเสียหายที่ตามมา

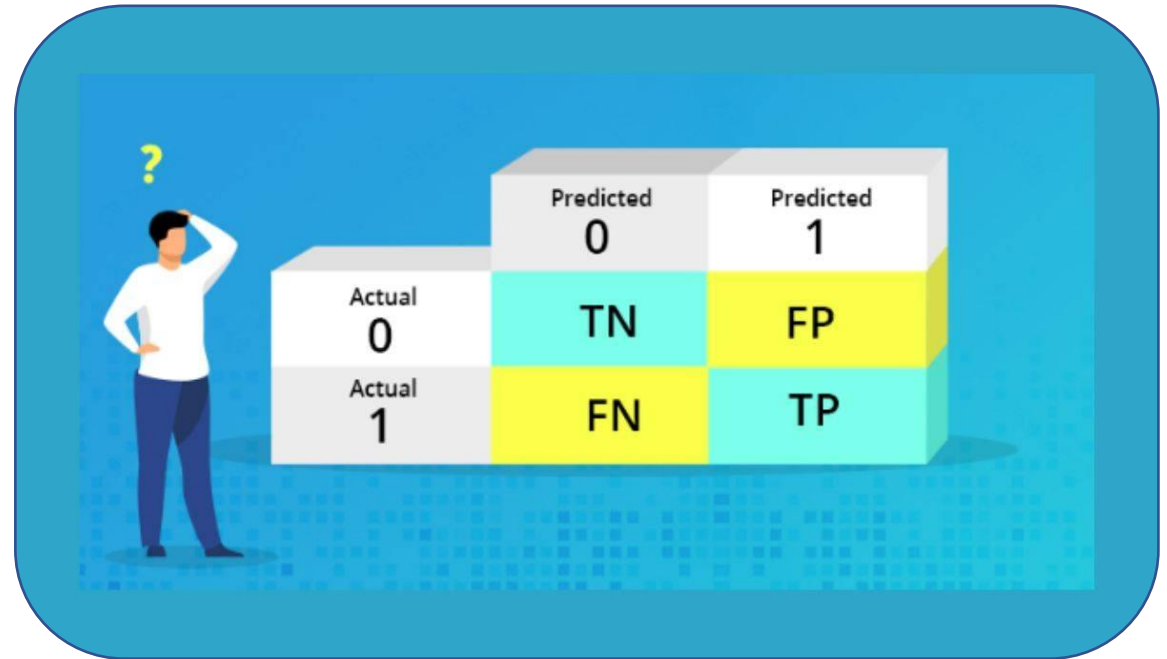


Confusion Matrix

Confusion Matrix

เป็นตารางที่ช่วยให้เราเห็นภาพรวมของผลลัพธ์
ที่โมเดลทำนาย (Predict) เทียบกับค่าจริง
(Actual) (ให้สังเกตตำแหน่งคำ)
ซึ่งใช้ประเมินประสิทธิภาพของโมเดล

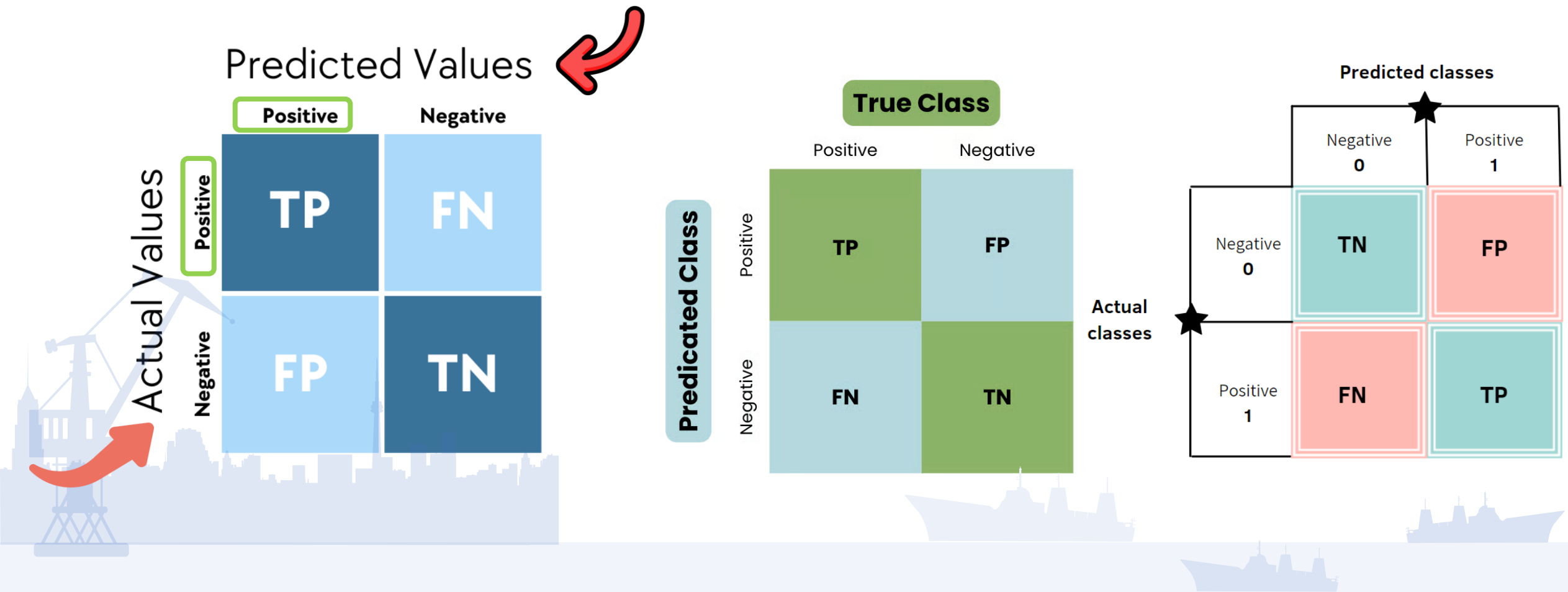
Classification



Confusion Matrix เป็นตารางที่ช่วยให้เราเห็นภาพรวมของผลลัพธ์

ที่โมเดลทำนาย (Predict) เทียบกับค่าจริง (Actual) (ให้สังเกตตำแหน่งคำ)

ซึ่งใช้ประเมินประสิทธิภาพของโมเดล Classification โดยแบ่งผลลัพธ์การพยากรณ์ออกเป็น 4 ประเภท :



Confusion Matrix เป็นตารางที่ช่วยให้เราเห็นภาพรวมของผลลัพธ์ที่โมเดลทำนาย เทียบกับค่าจริง

ความเป็นจริง (Actual)	ทำนาย (Predict)	
	ถูก Positive (ผลดี)	ผิด Negative (ผลเสีย)
เป็นจริง Positive	✓ ถูก -- เป็นผลดี	✗ ผิด -- เป็นผลเสีย
ไม่จริง Negative	✗ ผิด -- เป็นผลดี	✓ ถูก -- เป็นผลเสีย

คำทำนาย - ส่งผล ดี / ร้าย ต่อกรณีศึกษา



คำทำนาย - ส่งผล ดี / ร้าย ต่อกรณีศึกษา



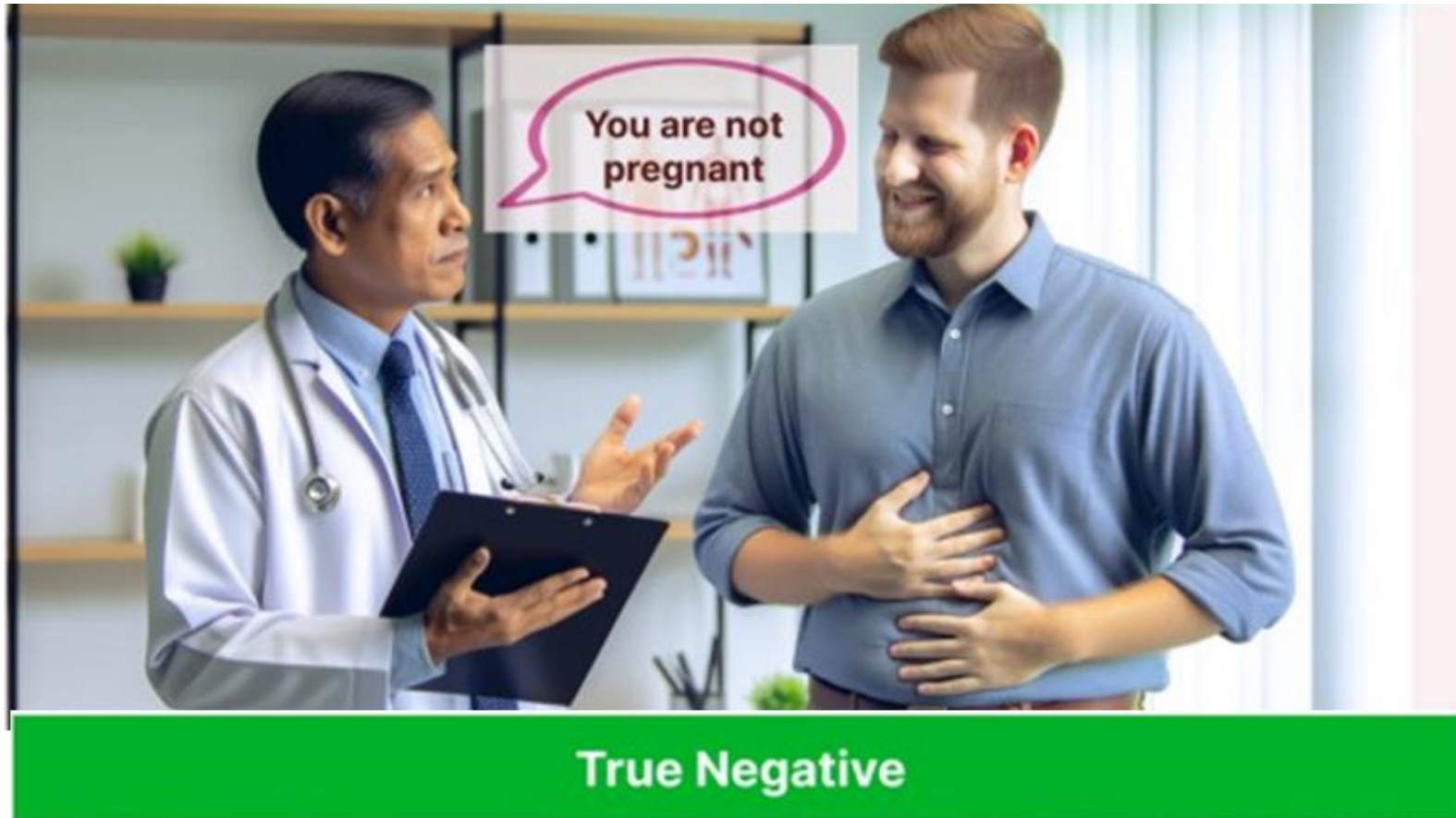
False Positive - Type 1 Error



คำทำนาย - ส่งผล ดี / ร้าย ต่อกรณีศึกษา



คำทำนาย - ส่งผล ดี / ร้าย ต่อกรณีศึกษา



Confusion Matrix เป็นตารางที่ช่วยให้เราเห็นภาพรวมของผลลัพธ์ที่โมเดลทำนาย เทียบกับค่าจริง

กรณี ทำนาย “ถูก”



ความเป็นจริง (Actual)	ทำนาย (Predict)	
	ถูก Positive (ผลดี)	ผิด Negative (ผลเสีย)
เป็นจริง Positive	✓ ถูก - เป็นผลดี True Positive (TP)	✗
ไม่จริง Negative	✗	✓ ถูก - เป็นผลเสีย True Negative (TN)

Confusion Matrix เป็นตารางที่ช่วยให้เราเห็นภาพรวมของผลลัพธ์ที่โมเดลทำนาย เทียบกับค่าจริง

กรณี ทำนาย “ผิด” ✕

ความเป็นจริง (Actual)	ทำนาย (Predict)	
	ถูก Positive (ผลดี)	ผิด Negative (ผลเสีย)
เป็นจริง Positive	✓ ถูก - เป็นผลดี True Positive (TP)	✕ ผิด - เป็นผลเสีย False Negative (FN)
ไม่จริง Negative	✕ ผิด - เป็นผลดี False Positive (FP)	✓ ถูก - เป็นผลเสีย True Negative (TN)

ความสำคัญของแต่ละค่า

เมตริก	ถ้าค่าสูง	ถ้าค่าต่ำ	ควรให้ความสำคัญเมื่อ
TP	โมเดลทำนายถูกเยอะขึ้น	โมเดลจับ Positive ได้ไม่ดี	เมื่อเราต้องการให้โมเดลทำนาย Positive ได้ถูกต้อง
FP	แจ้งเตือนผิดเยอะขึ้น	โมเดลแม่นยำขึ้น	เมื่อ False Positive มีผลกระทบมาก เช่น ทดสอบโรคร้ายแรง
FN	พลาดการตรวจจับเยอะขึ้น	โมเดลจับ Positive ได้ดีขึ้น	เมื่อ False Negative อันตราย เช่น การตรวจจับโรคร้าย
TN	โมเดลกรอง Negative ได้แม่นยำ	โมเดลพลาดจำแนก Negative	เมื่อเราต้องการลดการแจ้งเตือนผิดพลาด

คำทำนาย - ส่งผล ดี / ร้าย ต่อกรณีศึกษา

กรณีตัวอย่าง



ระบบตรวจโรคมะเร็ง

- True Positive (TP) → ตรวจพบว่าผู้ป่วยเป็นโรค และเขาเป็นจริง ✓
- False Positive (FP) → ตรวจพบว่าผู้ป่วยเป็นโรค แต่เขาไม่เป็น ✗ → อาจทำให้ผู้ป่วยเครียดโดยไม่จำเป็น
- False Negative (FN) → บอกว่าผู้ป่วยไม่เป็นโรค แต่เขาเป็นจริง ✗ → อันตราย เพราะอาจไม่ได้รับการรักษา
- True Negative (TN) → บอกว่าผู้ป่วยไม่เป็นโรค และเขาไม่เป็นจริง ✓

ผลตรวจจริง	ทำนายว่า เป็นมะเร็ง	ทำนายว่า ไม่เป็นมะเร็ง
เป็นมะเร็งจริง	✓ TP (ตรวจพบโรค ถูกต้อง)	✗ FN (พลาด ไม่เตือน)
ไม่เป็นมะเร็ง	✗ FP (แจ้งเตือน ผิดพลาด)	✓ TN (ปกติ ถูกต้อง)



ค่าชี้วัดที่ได้จาก Confusion Matrix

การประเมินผลโมเดล **Classification** มีหลายตัวชี้วัดที่ช่วยบอกว่าโมเดลมีประสิทธิภาพดีแค่ไหน โดยที่นิยมใช้ คือ Accuracy, Precision, Recall, และ F1-score

```
[[4 1 1]
 [6 2 2]
 [3 0 6]]
```

	precision	recall	f1-score	support
Cat	0.308	0.667	0.421	6
Fish	0.667	0.200	0.308	10
Hen	0.667	0.667	0.667	9
accuracy			0.480	25
macro avg	0.547	0.511	0.465	25
weighted avg	0.581	0.480	0.464	25

ค่าชี้วัดที่ได้จาก Confusion Matrix

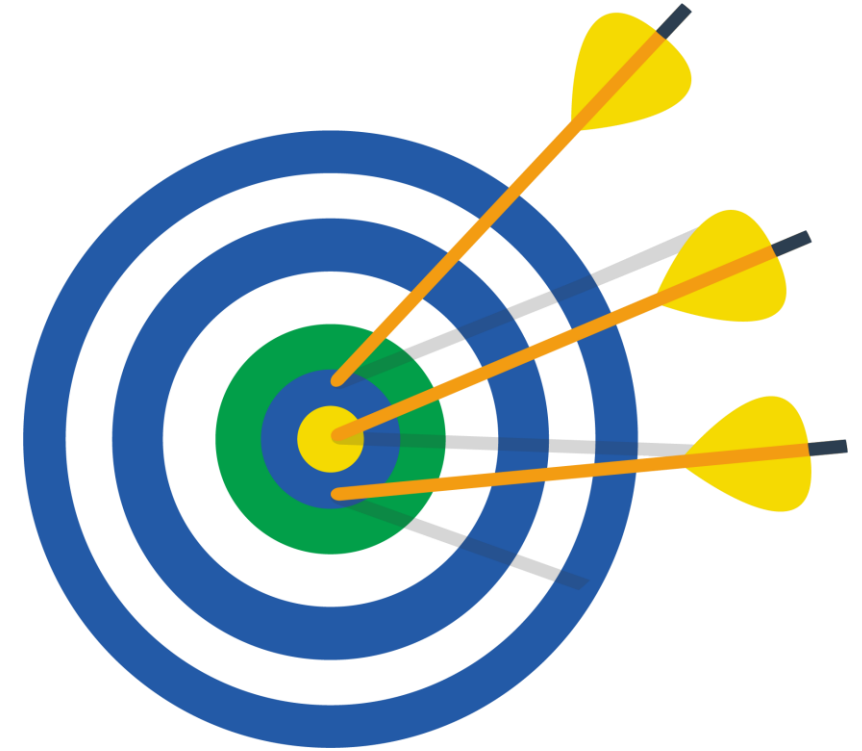
การประเมินผลโมเดล **Classification** มีหลายตัวชี้วัดที่ช่วยบอกว่าโมเดลมีประสิทธิภาพดีแค่ไหน โดยที่นิยมใช้ คือ Accuracy, Precision, Recall, และ F1-score

Accuracy (ความแม่นยำรวม)

เป็นตัวชี้วัดพื้นฐานที่ใช้ประเมินประสิทธิภาพของโมเดล

Classification โดยบอกว่าโมเดลทำนายถูกต้องเป็นกี่เปอร์เซ็นต์ของข้อมูลทั้งหมด

✓ ใช้ได้ดีถ้า dataset มีการกระจายของคลาสสมดุล



ค่าชี้วัดที่ได้จาก Confusion Matrix

การประเมินผลโมเดล Classification มีหลายตัวชี้วัดที่ช่วยบอกว่าโมเดลมีประสิทธิภาพดีแค่ไหน โดยที่นิยมใช้ คือ Accuracy, Precision, Recall, และ F1-score

Precision (ความแม่นยำของการทำนาย Positive)

เป็นตัวชี้วัดที่ใช้วัดว่า ในบรรดาตัวอย่างที่โมเดลทำนายว่าเป็น Positive มีสัดส่วนที่ถูกต้องจริงกี่เปอร์เซ็นต์

✓ ใช้ในกรณีที่ต้องการลด False Positive เช่น การตรวจจับโรค



Precision

Of all **positive predictions**,
how many are **really positive**?

$$\frac{TP}{TP + FP}$$

		Real Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

ค่าชี้วัดที่ได้จาก Confusion Matrix

การประเมินผลโมเดล Classification มีหลายตัวชี้วัดที่ช่วยบอกว่าโมเดลมีประสิทธิภาพดีแค่ไหน โดยที่นิยมใช้ คือ Accuracy, Precision, Recall, และ F1-score

Accuracy (ความแม่นยำรวม)

เป็นตัวชี้วัดพื้นฐานที่ใช้ประเมินประสิทธิภาพของโมเดล Classification โดยบอกว่าโมเดลทำนายถูกต้องเป็นกี่เปอร์เซ็นต์ของข้อมูลทั้งหมด

✓ ใช้ได้ดีถ้า dataset มีการกระจายของคลาสสมดุล

Precision (ความแม่นยำของการทำนาย Positive)

เป็นตัวชี้วัดที่ชี้วัดว่า ในบรรดาตัวอย่างที่โมเดลทำนายว่าเป็น Positive มีสัดส่วนที่ถูกต้องจริงกี่เปอร์เซ็นต์

✓ ใช้ในกรณีที่ต้องการลด False Positive เช่น การตรวจจับโรค

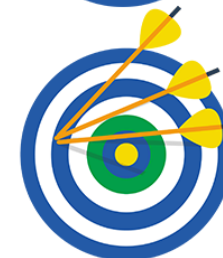
ACCURACY VS. PRECISION



Low Accuracy & Low Precision



High Accuracy & Low Precision



Low Accuracy & High Precision



High Accuracy & High Precision

ค่าชี้วัดที่ได้จาก Confusion Matrix

การประเมินผลโมเดล Classification มีหลายตัวชี้วัดที่ช่วยบอกว่าโมเดลมีประสิทธิภาพดีแค่ไหน โดยที่นิยมใช้ คือ Accuracy, Precision, Recall, และ F1-score

Recall (Sensitivity) (ความสามารถในการจับ Positive ได้ถูกต้อง)

หรือ True Positive Rate เป็นตัวชี้วัดที่ใช้วัดว่า ในบรรดาตัวอย่างที่เป็น Positive จริงๆ โมเดลสามารถจับได้ถูกต้องกี่เปอร์เซ็นต์

✓ สำคัญเมื่อ False Negative เป็นปัญหา เช่น การตรวจมะเร็ง

Recall

Of all **real positive** cases, how many are **predicted positive**?

$$\frac{TP}{TP + FN}$$

		Real Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

ค่าชี้วัดที่ได้จาก Confusion Matrix

การประเมินผลโมเดล Classification มีหลายตัวชี้วัดที่ช่วยบอกว่าโมเดลมีประสิทธิภาพดีแค่ไหน โดยที่นิยมใช้ คือ Accuracy, Precision, Recall, และ F1-score

Recall (Sensitivity) (ความสามารถในการจับ Positive ได้ถูกต้อง)

หรือ True Positive Rate เป็นตัวชี้วัดที่ใช้วัดว่า ในบรรดาตัวอย่างที่เป็น Positive จริงๆ โมเดลสามารถจับได้ถูกต้องกี่เปอร์เซ็นต์

✓ สำคัญเมื่อ False Negative เป็นปัญหา เช่น การตรวจมะเร็ง

Precision (ความแม่นยำของการทำนาย Positive)

เป็นตัวชี้วัดที่ใช้วัดว่า ในบรรดาตัวอย่างที่โมเดลทำนายว่าเป็น Positive มีสัดส่วนที่ถูกต้องจริงกี่เปอร์เซ็นต์

✓ ใช้ในกรณีที่ต้องการลด False Positive เช่น การตรวจจับโรค

Precision

Of all **positive predictions**, how many are **really positive**?

$$\frac{TP}{TP + FP}$$

		Real Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

ทำนายว่าเป็น Positive

Recall

Of all **real positive cases**, how many are **predicted positive**?

$$\frac{TP}{TP + FN}$$

		Real Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

เป็น Positive จริงๆ

ค่าชี้วัดที่ได้จาก Confusion Matrix

การประเมินผลโมเดล Classification มีหลายตัวชี้วัดที่ช่วยบอกว่าโมเดลมีประสิทธิภาพดีแค่ไหน โดยที่นิยมใช้ คือ Accuracy, Precision, Recall, และ F1-score

F1-score (ค่าเฉลี่ยของ Precision และ Recall)

ค่าชี้วัดที่เป็นค่าเฉลี่ยเชิงฮาร์โมนิก (Harmonic Mean) ของ Precision และ Recall ซึ่งช่วยให้เราตัดสินใจได้ดีขึ้นในกรณีที่ต้องการสมดุลระหว่าง

Precision และ Recall

✓ เหมาะกับปัญหาที่มีคลาสไม่สมดุล

Precision

Of all **positive predictions**, how many are **really positive**?

$$\frac{TP}{TP + FP}$$

		Real Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Recall

Of all **real positive cases**, how many are **predicted positive**?

$$\frac{TP}{TP + FN}$$

		Real Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

ค่าชี้วัดที่ได้จาก Confusion Matrix

Support คือจำนวนตัวอย่าง (Samples) ของแต่ละคลาสที่ใช้ในการคำนวณค่าต่างๆ เช่น Precision, Recall, และ F1-score ในโมเดล Classification

	precision	recall	f1-score	support	
0	0.80	0.82	0.81	1000	Individual Sample Size
1	0.95	0.98	0.96	1000	
2	0.77	0.76	0.77	1000	
3	0.88	0.87	0.87	1000	
4	0.75	0.82	0.79	1000	
5	0.95	0.88	0.91	1000	
6	0.68	0.58	0.63	1000	
7	0.90	0.93	0.92	1000	
8	0.93	0.95	0.94	1000	
9	0.92	0.95	0.93	1000	
accuracy			0.85	10000	Total Sample Size
macro avg	0.85	0.85	0.85	10000	
weighted avg	0.85	0.85	0.85	10000	

ค่าชี้วัดที่ได้จาก Confusion Matrix

การประเมินผลโมเดล Classification มีหลายตัวชี้วัดที่ช่วยบอกว่าโมเดลมีประสิทธิภาพดีแค่ไหน โดยที่นิยมใช้ คือ Accuracy, Precision, Recall, และ F1-score

เมตริก	คำนิยาม	สูตรคำนวณ	ใช้ในกรณีไหน?
Accuracy (ความถูกต้อง)	เปอร์เซ็นต์ของจำนวนตัวอย่างทั้งหมดที่โมเดลทำนายถูกต้อง	$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$	ใช้เมื่อข้อมูลมีการกระจายคลาสที่สมดุลกัน
Precision (ความแม่นยำ)	เปอร์เซ็นต์ของผลลัพธ์ที่โมเดลทำนายเป็น Positive แล้วถูกต้องจริง	$Precision = \frac{TP}{TP+FP}$	ใช้เมื่อ False Positive มีผลเสียมาก
Recall (การครอบคลุม)	เปอร์เซ็นต์ของข้อมูล Positive จริง ที่โมเดลสามารถจำแนกออกมาได้	$Recall = \frac{TP}{TP+FN}$	ใช้เมื่อ False Negative มีผลเสียมาก
F1-score	ค่าเฉลี่ยถ่วงน้ำหนักระหว่าง Precision และ Recall	$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$	ใช้เมื่อต้องการสมดุลระหว่าง Precision และ Recall

ค่าชี้วัดที่ได้จาก Confusion Matrix

1) Accuracy ความถูกต้อง แม่นยำรวม

2) Precision หรือ Positive Predictive Value ความแม่นยำของการทำนาย Positive

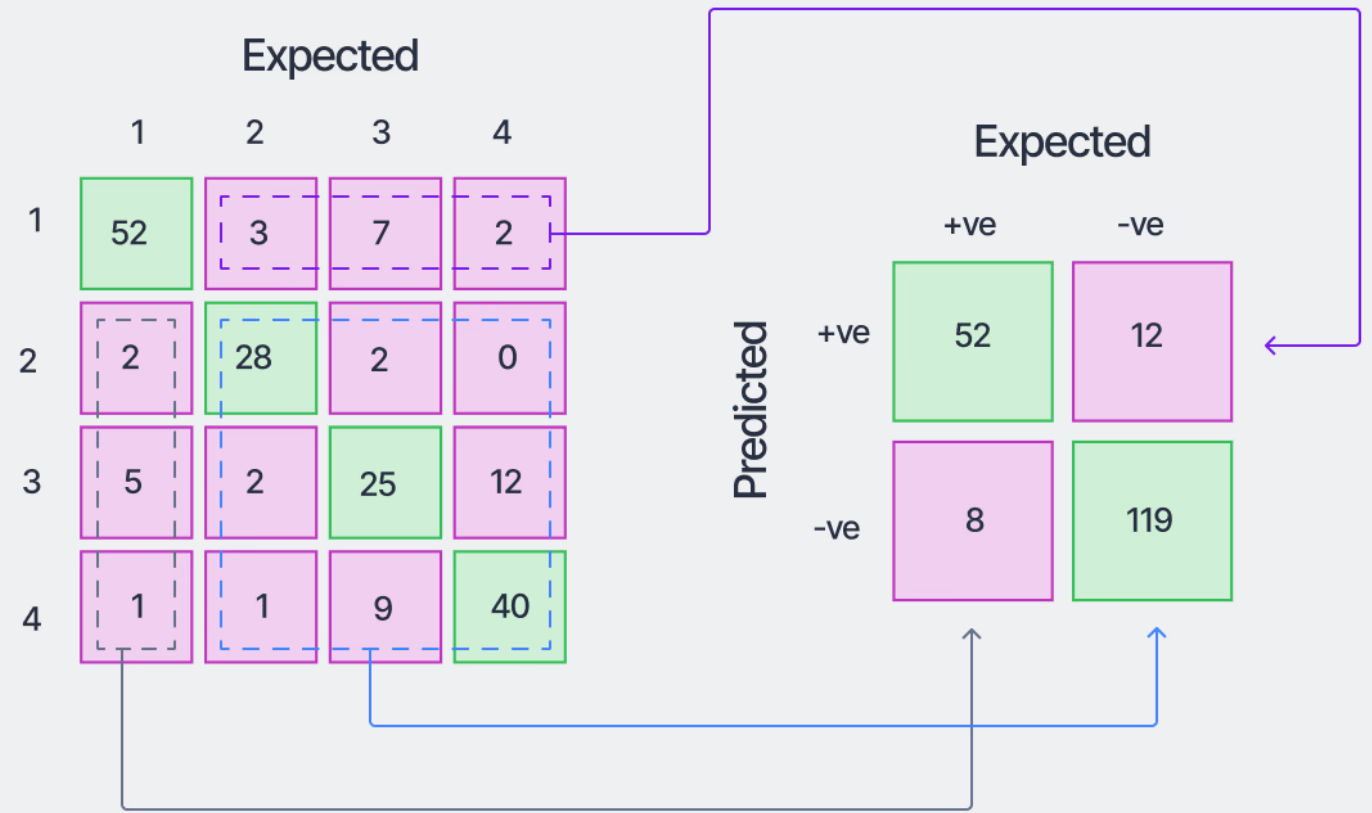
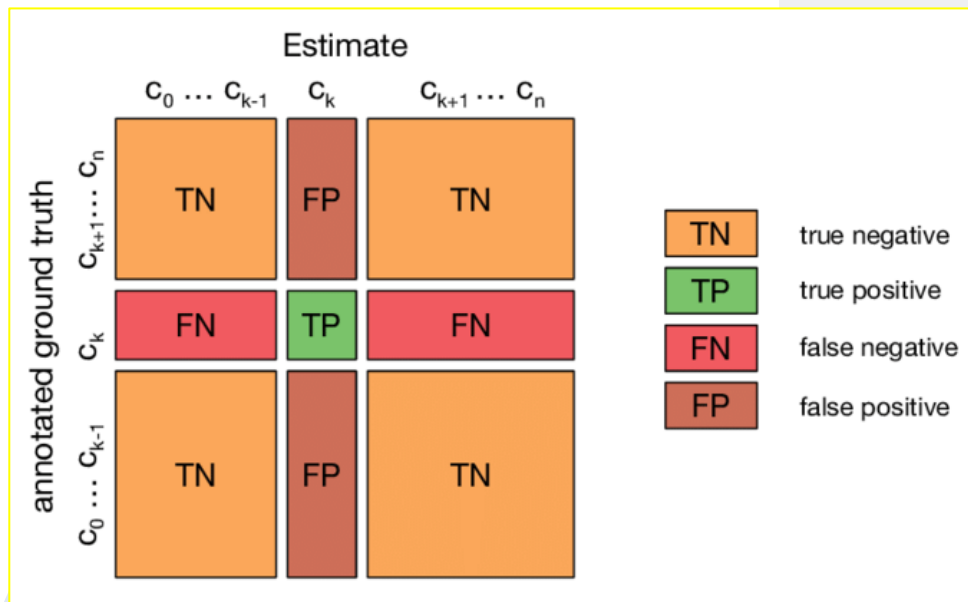
3) Sensitivity หรือ Recall ความไว ในการจับ Positive

4) Specificity ความจำเพาะ

5) F1-Score เฉลี่ย precision และ recall

		Predicted Class		
		Positive	Negative	
Actual Class	Positive	True Positive (TP)	False Negative (FN) Type II Error	Sensitivity $\frac{TP}{(TP + FN)}$
	Negative	False Positive (FP) Type I Error	True Negative (TN)	Specificity $\frac{TN}{(TN + FP)}$
		Precision $\frac{TP}{(TP + FP)}$	Negative Predictive Value $\frac{TN}{(TN + FN)}$	Accuracy $\frac{TP + TN}{(TP + TN + FP + FN)}$

กรณี Confusion matrix ที่เป็น multi-class classification.



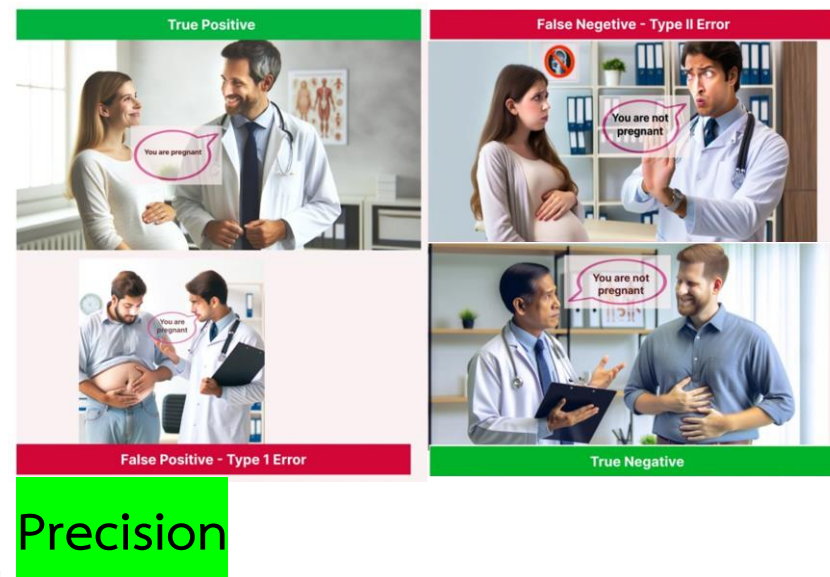
การเลือกใช้เมตริก

- ✓ True Positive (TP) → ทำนายว่าเป็น **Positive** และถูกต้อง
- ✓ False Positive (FP) → ทำนายว่าเป็น **Positive** แต่ผิดพลาด (แจ้งเตือนเกิน)
- ✓ False Negative (FN) → ทำนายว่าเป็น **Negative** แต่ผิดพลาด (พลาดการตรวจจับ)
- ✓ True Negative (TN) → ทำนายว่าเป็น **Negative** และถูกต้อง

- ❖ ถ้าต้องการความแม่นยำสูง (ลด FP / Type I Error) → ใช้ **Precision**
- ❖ ถ้าต้องการตรวจจับให้ครบ (ลด FN / Type II Error) → ใช้ **Recall**
- ❖ ถ้าต้องการสมดุล Precision และ Recall → ใช้ **F1-score**

เลือกเมตริกให้เหมาะสมกับปัญหา จะช่วยให้โมเดลทำงานได้ดีที่สุด!

Recall



การลดข้อผิดพลาด (Trade-off ระหว่าง Type 1 และ Type 2 Error)

- หากต้องการลด Type 1 Error (False Positive):
 - เพิ่ม ความแม่นยำ (Precision)
 - ใช้ ค่า Threshold สูงขึ้น เพื่อลดการแจ้งเตือนผิด
- หากต้องการลด Type 2 Error (False Negative):
 - เพิ่ม ความไว (Recall) ตรวจจับให้ครบ
 - ใช้ ค่า Threshold ต่ำลง เพื่อตรวจจับให้มากขึ้น



✚ ขึ้นอยู่กับสถานการณ์ว่าควรลดข้อผิดพลาดประเภทใด

- การแพทย์ → ลด Type 2 Error (False Negative) → ควรให้ความสำคัญกับ Recall
- การตรวจจับสเปม → ลด Type 1 Error (False Positive) → ควรให้ความสำคัญกับ Precision

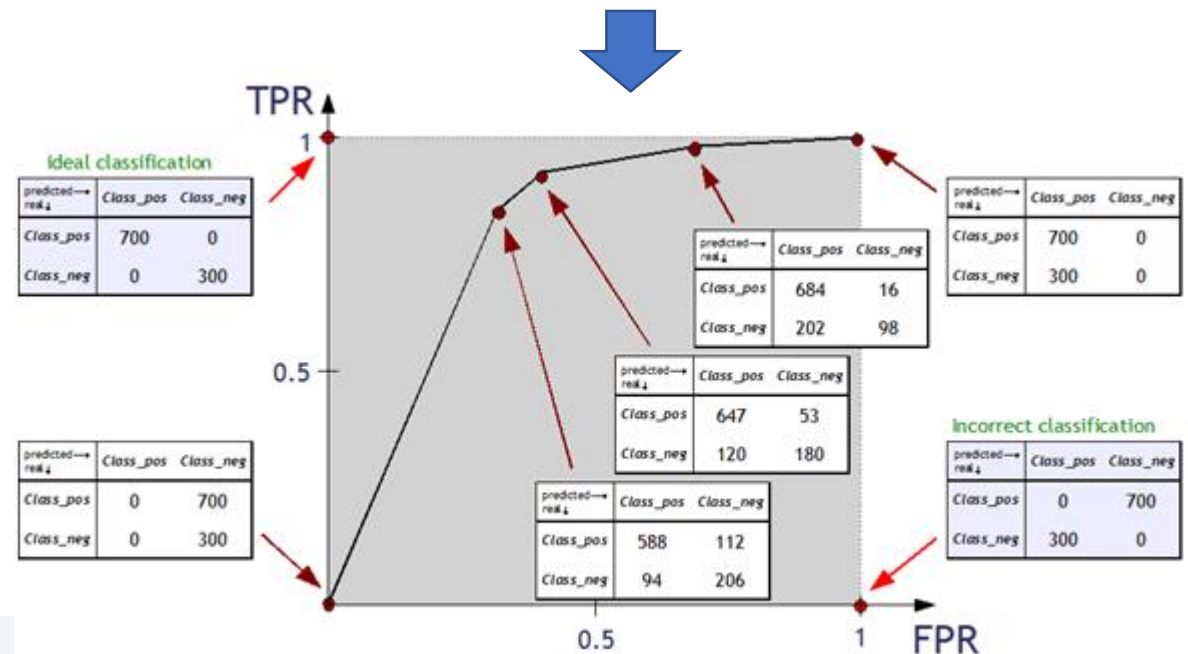
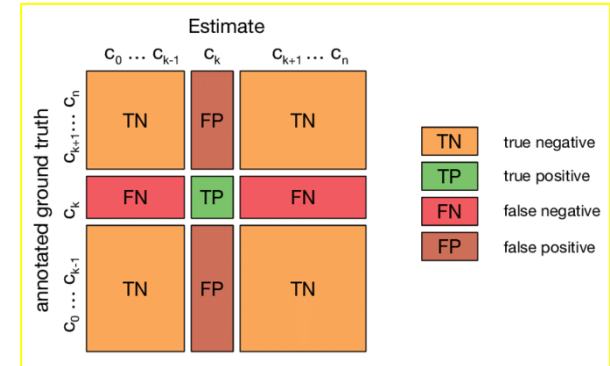
ROC (Receiver Operating Characteristic) Curve เป็นกราฟที่ใช้วิเคราะห์ประสิทธิภาพของโมเดล Classification โดยเฉพาะ Binary Classification (เช่น ทำนายว่า "เป็นโรค" หรือ "ไม่เป็นโรค")

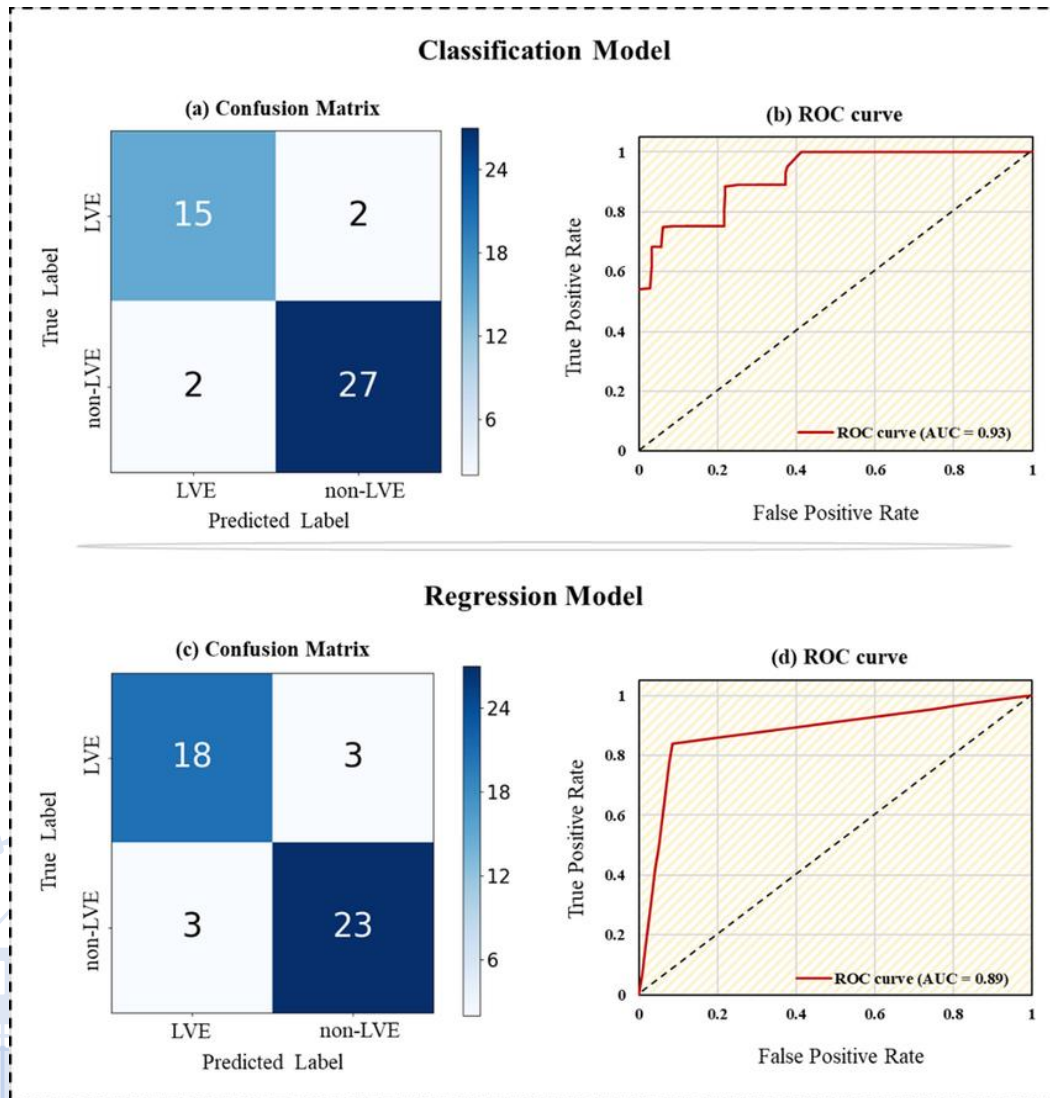
ช่วยให้เราเข้าใจว่าโมเดลสามารถแยกแยะ Positive และ Negative ได้ดีแค่ไหนเมื่อเปลี่ยน Threshold (ค่ากำหนดในการตัดสินใจเป็น Positive หรือ Negative)

- 1 True Positive Rate (TPR) หรือ Recall
- 2 False Positive Rate (FPR)

🔗 การอ่านค่า ROC Curve

- เส้นโค้งอยู่ด้านบน (ใกล้ 100% TPR และ 0% FPR)
- 👉 หมายถึง โมเดลดีมาก แยกแยะ Positive และ Negative ได้แม่นยำ
- เส้นทแยงมุม (Baseline Model, 50%)
- 👉 หมายถึง โมเดลทำนายแบบสุ่ม (ไม่มีความสามารถในการจำแนก)
- เส้นโค้งอยู่ใกล้เส้นทแยงมุม
- 👉 หมายถึง โมเดลไม่ดี คาดการณ์ได้ใกล้เคียงการสุ่ม





✂ ค่า AUC (Area Under Curve) → พื้นที่ใต้กราฟ ROC

- AUC ใกล้ 1 → โมเดลแม่นยำสูง
- $AUC \approx 0.5$ → โมเดลแย่มาก (เดาแบบสุ่ม)
- $AUC < 0.5$ → โมเดลอาจมีการกลับหัว (ต้องแก้ไขโมเดล)

ความสำคัญของ ROC Curve

1 ช่วยเปรียบเทียบโมเดล

- ถ้ามีหลายโมเดล สามารถใช้ AUC-ROC เปรียบเทียบว่าโมเดลไหนแม่นยำกว่า

2 ช่วยเลือก Threshold ที่เหมาะสม

- หาก Threshold สูง → ลด False Positive แต่เพิ่ม False Negative
- หาก Threshold ต่ำ → ลด False Negative แต่เพิ่ม False Positive

3 เหมาะกับปัญหาที่มี Class ไม่สมดุล (Imbalanced Data)

- กรณีที่ Positive น้อย (เช่น ตรวจจับ Fraud, ตรวจโรคหายาก) Accuracy อาจใช้ไม่ได้ แต่ ROC-AUC ยังคงใช้ได้



THANK

YOU

