

# Introduction to Machine Learning

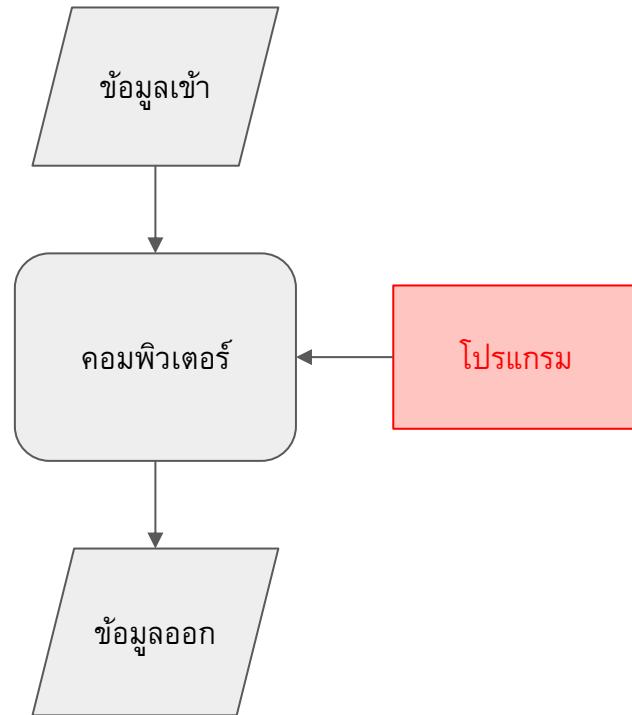
Kasetsart University & DEPA

# Outline

- Data Modeling
- Model and Program
- Machine Learning Problems
- Model Evaluation Metrics
- Train Validation Test Set
- Overfitting / Underfitting

# Data Modeling

- โปรแกรมคือตัวแบบในการเปลี่ยน input ให้เป็น output
  - แสดงความสัมพันธ์ระหว่าง input กับ output
- ไม่เดลก์คล้ายกับโปรแกรม
- ข้อแตกต่างคือ
  - โปรแกรมนั้นถูกสร้างขึ้นโดยโปรแกรมเมอร์
  - ไม่เดลนั้นถูกอุปนายขึ้นจากข้อมูล
- การสกัดตัวเรียนก input เป็นตัวแปรต้น และ output เป็นตัวแปรตาม



# ตัวแปรต้นและตัวแปรตาม

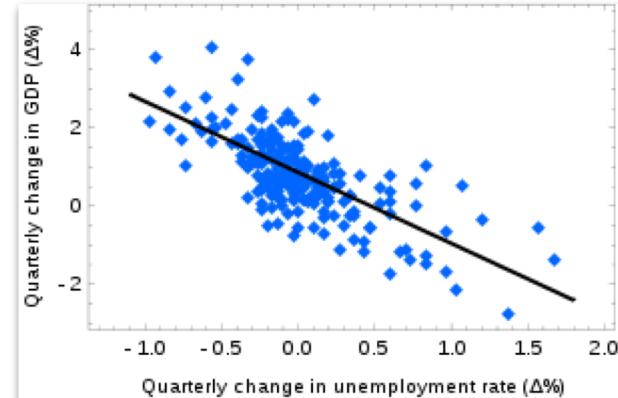
- ค่าของตัวแปรตามจะขึ้นกับตัวแปรต้น
- ศาสตร์ต่างกันอาจจะเรียกชื่อตัวแปรต้นตัวแปรตามต่างกัน

ตัวแปรต้น	ตัวแปรตาม	ศาสตร์
Input	Output	Computer Science
Independent variable	Dependent variable	คณิตศาสตร์, สัมบูรณ์
Regressor, control variable, explanatory variable	Response variable, outcome variable	สถิติ
Feature, attribute	Label, target attribute	Machine Learning

# โมเดล กับ โปรแกรม

- เราสร้างโปรแกรมเมื่อเข้าใจกฎเกณฑ์ในการเปลี่ยน input เป็น output ชัดเจน
  - ค่าจอดรถ: 15 บาทแรกฟรี ต่อไปซึ่งมองละ 20 บาท เศษของซึ่งมองคิดเป็นหนึ่งซึ่งมอง สูงสุด 8 ซึ่งมอง
- เราสร้างโมเดลในกรณีที่กฎเกณฑ์ในการแปลงข้อมูลเข้าเป็นข้อมูลอุปกรณ์ชั้บช้อนไม่ชัดเจน หรือมีความไม่แน่นอนเข้ามาเกี่ยวข้อง
  - การเปลี่ยนแปลงจำนวนผู้ว่างงานส่งผลต่อ GDP อย่างไร
  - สร้างโมเดลจากข้อมูลการเปลี่ยนแปลงของ GDP กับ การเปลี่ยนแปลงจำนวนคนว่างงานในแต่ละไตรมาส
  - %Change GDP =  

$$0.789 - 1.654 * (\text{Change Unemployment Rate})$$



[https://en.wikipedia.org/wiki/Okun%27s\\_law](https://en.wikipedia.org/wiki/Okun%27s_law)

# ประเภทของข้อมูล

- ข้อมูลตัวเลข (numeric, continuous)
  - สามารถนำมาคำนวณได้ เช่น จำนวนประชากร
- ข้อมูลแบบประเภท (category, nominal)
  - ไม่มีความสัมพันธ์ระหว่างประเภท เช่น ชาย/หญิง หรือ ชื่อจังหวัด
- ข้อมูลแบบลำดับ (ordinal)
  - สามารถเรียงลำดับประเภทได้ เช่น ดี/ปานกลาง/พอใช้ หรือ ร้อน/อบอุ่น/เย็น/หนาว
- ข้อมูลแบบช่วง (interval)
  - มีลำดับและวัดความแตกต่างระหว่างลำดับได้เป็นช่วง เช่น ปีค.ศ.
- ตัวแปรตัวบวกและตัวแปรตามเป็นประเภทไหนก็ได้

# วิธีการสร้างโมเดล

- วิธีการทางสกัตติ
  - ใช้คณิตศาสตร์ทางสกัตติในการสร้างโมเดล
- วิธีการทาง Machine Learning
  - ใช้วิธีด้าน optimization ในการสร้างโมเดล
- ตอนนี้ก็สองวิธีเริ่มใช้รวมกันและไม่มีเส้นแบ่งที่ชัดเจนนัก

# ประเภทของโมเดล

- Regression
  - หาความสัมพันธ์ระหว่างตัวแปรต้นกับตัวแปรตาม
  - ตัวแปรตามต้องเป็นประเภทตัวเลข (numeric)
  - ตัวแปรต้นอาจมีมากกว่าหนึ่งตัว แต่ไม่จำเป็นต้องเป็นประเภท numeric ทุกตัว
  - เช่น โมเดลทำนายอุณหภูมิของเมือง ด้วยอุณหภูมิของเมืองใกล้ ๆ
- Classification
  - ทำนายว่าตัวอย่าง(ใหม่)จัดอยู่ในประเภทใด
  - ตัวแปรตามต้องเป็นประเภท category
  - เช่น โมเดลทำนายพันธุ์ดอกไอริส ด้วยความยาวและความกว้างกลีบจริงและกลีบเลี้ยง
- Regression กับ Classification ถือเป็น **Supervised Learning** คือตัวอย่างต้องมีผลเฉลย (ตัวแปรตาม)

# ตัวอย่างข้อมูล

คอลัมน์ผลเดลย์ (label)

	city	country	latitude	longitude	temperature
41	Bradford	United Kingdom	53.80	-1.75	8.39
198	Trikala	Greece	39.56	21.77	16.00
70	Daugavpils	Latvia	55.88	26.51	5.38
175	Salzburg	Austria	47.81	13.04	4.62
124	Limoges	France	45.83	1.25	10.32
83	Eskisehir	Turkey	39.79	30.53	11.11
29	Bialystok	Poland	53.15	23.17	6.07
121	Kryvyy Rih	Ukraine	47.93	33.34	8.61
181	Sivas	Turkey	39.75	37.03	8.05

regression

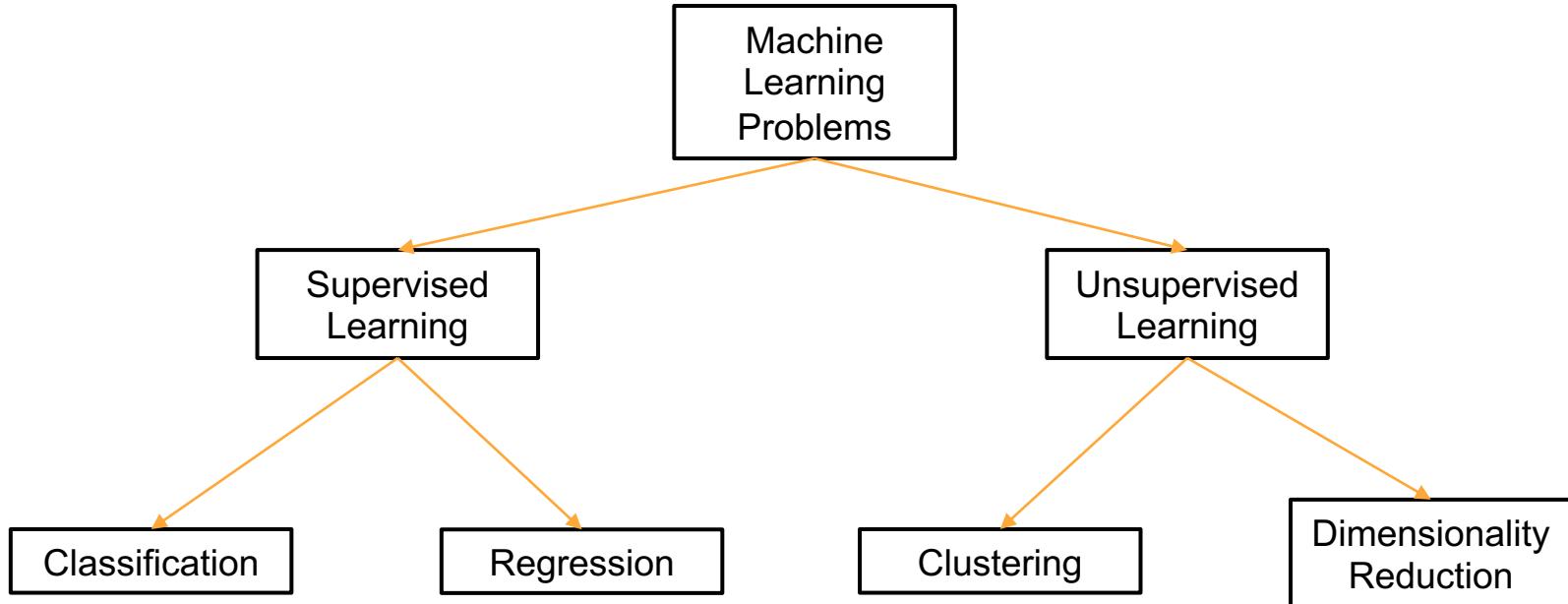
	sepal length	sepal width	petal length	petal width	iris
91	6.1	3.0	4.6	1.4	Iris-versicolor
89	5.5	2.5	4.0	1.3	Iris-versicolor
115	6.4	3.2	5.3	2.3	Iris-virginica
24	4.8	3.4	1.9	0.2	Iris-setosa
76	6.8	2.8	4.8	1.4	Iris-versicolor
136	6.3	3.4	5.6	2.4	Iris-virginica
138	6.0	3.0	4.8	1.8	Iris-virginica
62	6.0	2.2	4.0	1.0	Iris-versicolor
29	4.7	3.2	1.6	0.2	Iris-setosa

classification

# ประเภทของโมเดล

- Clustering
  - ใช้จัดข้อมูลให้เป็นกลุ่ม โดยตัวอย่างในกลุ่มเดียวกันจะมีความ “เหมือนกัน” มากกว่าตัวอย่างบนอกกลุ่ม
  - ความเหมือนกันนิยามได้หลากหลายวิธี
  - เช่น จัดกลุ่มคะแนนสอบให้เป็น 7 กลุ่มเพื่อตัดเกรด, จัดกลุ่มลูกค้าตามประเภทและราคาสินค้าที่ซื้อ
- Dimensionality reduction
  - ใช้ลดจำนวนตัวแปรต้น (คอลัมน์) ของชุดข้อมูล โดยยังคงปริมาณสารสนเทศไว้
  - ลดขนาดของโมเดล เพิ่มความเร็วในการเรียนรู้และทำนาย
- ก็ง Clustering และ Dimensionality reduction จัดเป็น **Unsupervised Learning** คือตัวอย่างไม่มีผลเฉลย

# ປະເທດຂອງໄມເດວ



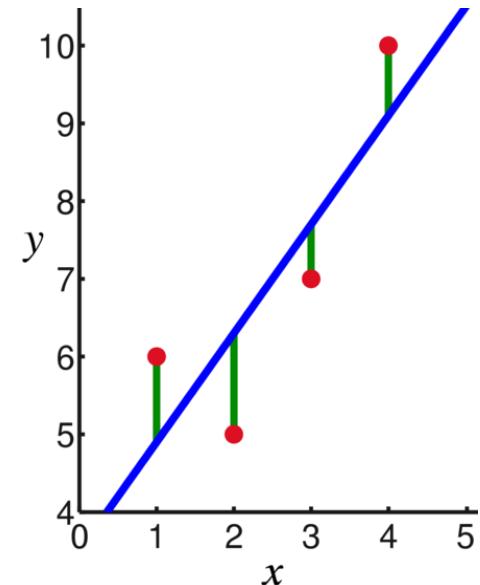
# Evaluation Metrics for Supervised Learning Models

ເພກ-ສົນລະເຊີນຈາກ ດັກທະນາໄມ

- การสร้างโมเดลจากข้อมูลนั้นเป็นใช้วิธีการทางอุปนัย (induction) เพื่อ"เหมารวม" (generalization) สิ่งประเทกนันกึ่งหนดจากตัวอย่างที่มี
  - ไม่สามารถ"พิสูจน์"ได้ว่าโมเดลที่ได้นั้นถูกต้อง
- จึงต้องใช้มาตรฐานวัดความใกล้เคียงของผลการทำนายของโมเดลเทียบกับเฉลย/คำตอบ
- ตอนนี้จะกล่าวถึงการวัดความถูกต้องของโมเดล **ຮະຫວ່າງການສ້າງໂມເດລ**
  - วัดจากผลเฉลยของชุดข้อมูลที่มีอยู่แล้ว
  - ดังนั้นจะใช้กับ Model แบบ Supervised Learning เท่านั้น ↗ **ວິຊາ ,ເຫຼຸ້ນໆ**

# Regression Model Metrics

- ผลเฉลยและผลทำนายที่ได้จากโมเดลเป็นค่าตัวเลข ดังนั้นใช้การคำนวณมา มาตรวัดได้
- Error: ผลต่างระหว่างผลเฉลยและค่าทำนาย
  - ถ้านำ Error ของหลายการทำนายมาบวกกันอาจมีค่าที่หักล้างกันไป
- Absolute Error: ค่าสัมบูรณ์ของผลต่างระหว่างค่าทำนายและผลเฉลย
  - ผลต่างจะเป็นค่าบวกเสมอ
- Squared Error: ค่ายกกำลังสองของผลต่างระหว่างค่าทำนายและผลเฉลย
  - ผลต่างจะเป็นค่าบวกเสมอ ผลต่างที่มากจะทำให้ค่า squared error ยิ่งมากตามไปด้วย
- Percentage Error: Error หารด้วยผลเฉลย คิดเป็นเปอร์เซ็นต์



# Regression Model Metrics

2 ตัวนี้ไม่ต้องกู้, แต่ใช้ของขึ้นมาเท่านั้น

- **Mean Absolute Error (MAE)**: คำนวณ Absolute error ของทุกการทำนายมาหาค่าเฉลี่ย
- **Mean Squared Error (MSE)**: คำนวณ Squared Error ของทุกการทำนายมาหาค่าเฉลี่ย
- **Root Mean Squared Error (RMSE)**: คำนวณ Squared Error ของทุกการทำนายมาหาค่าเฉลี่ย และถอดรากที่สอง
  - หน่วยของมาตรฐานจะเป็นหน่วยเดียวกันกับข้อมูล
- **Mean Absolute Percentage Error (MAPE)**: คำนวณ Percentage error มาหาค่า Absolute และเฉลี่ยกัน

↑ ที่ใช้กันสุด (มากกว่าค่าของ RMSE กว่า 10 เท่า)

# R-squared

ต้องใช้กับข้อมูลในทรงฟังก์ท่านนั้น

*r - รากที่สอง*

- มาตรวัดว่าโมเดลอธิบายความแปรปรวนของผลเฉลยได้มากแค่ไหน
- ค่า  $R^2$  โดยทั่วไปอยู่ระหว่าง 0-1
 

*หมายเหตุทุกวันนี้คงจะใช้ R^2 ก. ไม่ใช่ R^0.5*

  - ถ้า  $R^2 = 0.7$  : ความแปรปรวน 70% ของผลเฉลยอธิบายได้ด้วยโมเดล **70%**
- เช่น ความแปรปรวนของอุณหภูมิของเมืองในยุโรปจากชุดข้อมูลคือ 12.68
  - นั่นคือถ้าให้เดาค่าอุณหภูมิของเมืองโดยไม่มีข้อมูลอื่นเลย ความแปรปรวนของคำต่อbacจะอยู่ที่ 12.68
- พอดีร่างโมเดลเสร็จ ค่า  $R^2$  เป็น 0.7
  - ข้อมูลเพิ่มเติมและวิธีการที่ใช้สร้างโมเดลนั้นอธิบายความแปรปรวนไปได้ 70% คือ  $12.68 * 0.7 = 8.88$  ดังนั้น ยังมีความแปรปรวนอีก 3.5 ที่ไม่ครอบคลุมด้วยโมเดล

# Classification Model Metrics

ต้องพิจารณาอย่างไร

- การวัดประสิทธิภาพด้านการแยกแยะของโมเดล มาตรวัดง่ายสุดคือความแม่นยำ (accuracy)
- สมมติว่าตัวอย่างมี 2 classes คือ Yes และ No
- นิยามของความแม่นยำ

True positive

จำนวนตัวอย่างที่เป็น Yes และ  
โมเดลกำหนดว่าเป็น Yes

ความแม่นยำ =

$$\frac{\text{จำนวนตัวอย่างที่ถูกต้อง}}{\text{จำนวนตัวอย่างทั้งหมด}}$$

จำนวนตัวอย่างทั้งหมด

ทำนายถูก  $\sum_{i=1}^N (y_i - \hat{y}_i)^2$  ลูกบาศก์  $B$

พื้นที่ไม่ถูก  $\sum_{i=1}^N (y_i - \hat{y}_i)^2$  ลูกบาศก์  $A$

True Negative  
จำนวนตัวอย่างที่เป็น No และ  
โมเดลกำหนดว่าเป็น No

$$\frac{y (\text{ถูกต้อง})}{\text{จำนวนตัวอย่าง}}$$

ไม่ถูก  $\text{จำนวนตัวอย่าง}$

$B \leftarrow \text{จำนวนตัวอย่างที่ถูกต้อง}$

$A \leftarrow \text{จำนวนตัวอย่างที่ไม่ถูกต้อง}$

$\beta \leftarrow \text{จำนวนตัวอย่างที่ถูกต้อง}$

$\alpha \leftarrow \text{จำนวนตัวอย่างที่ไม่ถูกต้อง}$

$\gamma \leftarrow \text{จำนวนตัวอย่างที่ถูกต้อง}$

$\delta \leftarrow \text{จำนวนตัวอย่างที่ไม่ถูกต้อง}$

# ปัญหาของความแม่นยำ

- โมเดลที่แม่นยำ 90% นั้นเป็นโมเดลที่ดีหรือไม่
- เกี่ยวกับอะไร?
- สมมติว่ามีข้อมูล 450 ตัวอย่าง เป็น Class NO 407 ตัวอย่างและ Class YES อีก 43 ตัวอย่าง
 

→ ใน 450 ตัวอย่าง 407 คือ print ("NO")
- โมเดลโง่ๆ คำน่ายิ่งว่าเป็น NO เสมอ จะคำนยถูก 407 จาก 450 ตัวอย่าง คิดเป็น 90.44% ดีกว่าโมเดลแรก
- เรียกโมเดลโง่ๆ แบบนี้ว่า Dummy Model เอาไว้ใช้เป็นมาตรฐานการเปรียบเทียบ (baseline)
 

→ ใช้ตัวทั้งหมด 450 ตัวอย่าง

# ปัญหาที่ 2 ของความแม่นยำ

- สมมติว่ามีการตรวจหาโรคที่แม่นยำ 99% ถ้าสุ่มคนขึ้นมา 1 คนแล้วพาไปตรวจโรคด้วยวิธีนี้แล้วผลการตรวจบอกว่าเป็นโรค โอกาสที่คุณนั้นจะเป็นโรคจริงนั้นมีกี่ %
  - 99%
  - 1%
  - ข้อมูลไม่พอที่จะตอบ
- ขึ้นกับความพบรากของโรคด้วย
- สมมติต่อว่าโรคนั้นพบใน 1 คน จาก 10000 คน

โอกาสเป็น 1/10000

10,000,000

เป็นโรคจริง

1,000

ไม่เป็นโรคจริง

9,999,000

WRะเจ้ารู้

**รู้เท็จ = เผื่องโรค**    **รู้เท็จ = ไม่เป็นโรค**

กดสอบ

ความแม่นยำ 99%

990

10  
*false Negative*

9,899,010

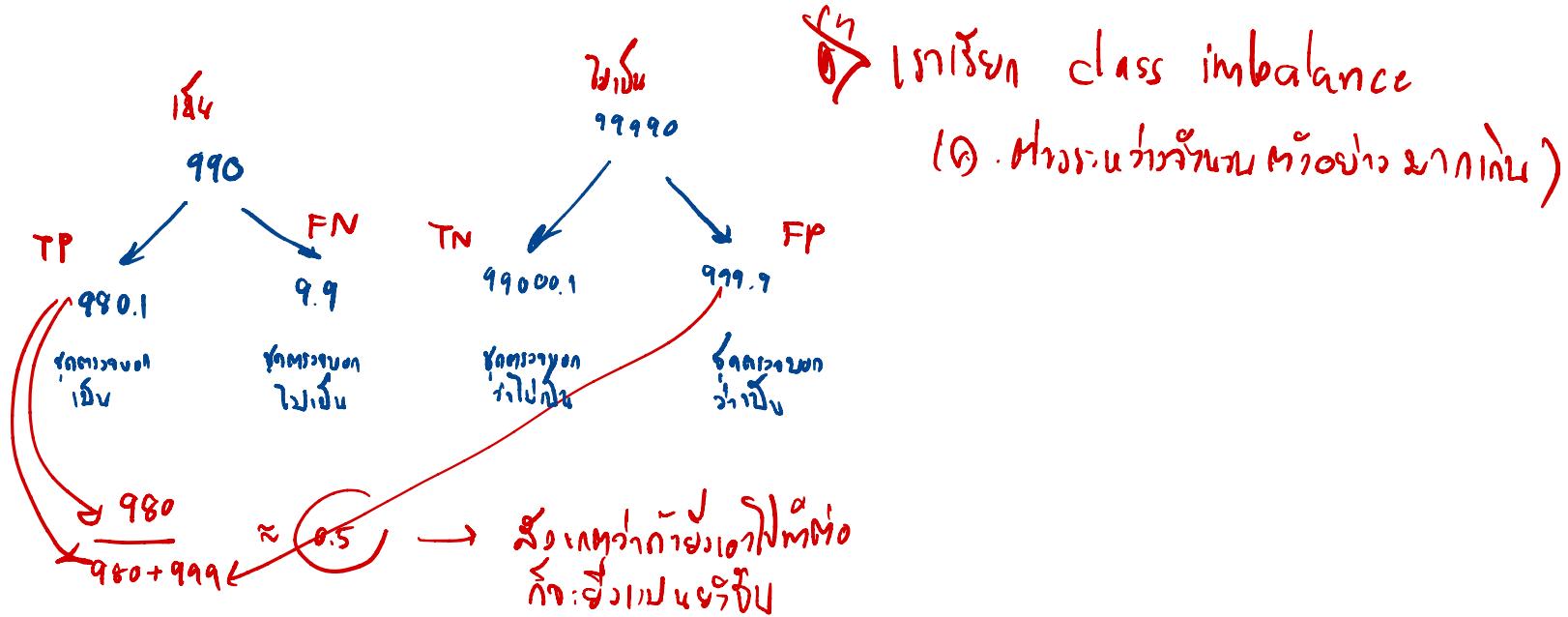
99,990  
*false positive*

โอกาสเป็น  
โรคจริงเมื่อ<sup>+</sup>  
ตรวจว่าเป็น

990 TP

0.0098

๙๙๗, ๑٪



# Confusion Matrix

ความก้าวหน้าที่ต้องการ  
ให้เกิดในเวลาเร็วๆ นี้

- N: จำนวนตัวอย่างทั้งหมด
- TN (True Negative): กำหนดว่าไม่ใช่และตัวอย่างนั้นไม่ใช่จริงๆ
- TP (True Positive): กำหนดว่าใช่และตัวอย่างนั้นใช่จริงๆ
- FN (False Negative): กำหนดว่าไม่ใช่แต่ตัวอย่างนั้นใช่
  - Type II Error ความลับซ่อนอยู่
- FP (False Positive): กำหนดว่าใช่แต่ตัวอย่างนั้นไม่ใช่
  - Type I Error ความลับซ่อนอยู่
- Confusion Matrix แสดงจำนวนของ TN, TP, FN, FP ทั้งหมด

<u>N = 450</u>	<u>Predicted NO</u>	<u>Predicted YES</u>
<u>Actual NO</u>	<u>TN = 400</u>	<u>FP = 7</u>
<u>Actual YES</u>	<u>FN = 17</u>	<u>TP = 26</u>

Accuracy: for what fraction of all instances is the classifier's prediction correct (for either positive or negative class)?

True negative	TN = 400	FP = 7	Accuracy = $\frac{TN+TP}{TN+TP+FN+FP}$
True positive	FN = 17	TP = 26	$= \frac{400+26}{400+26+17+7}$
Predicted negative	Predicted positive	$N = 450$	<i>มีน้ำใจมาก ✓</i>

**Classification error (1 - Accuracy): for what fraction of all instances is the classifier's prediction incorrect?**

True negative	TN = 400	FP = 7	
True positive	FN = 17	TP = 26	
Predicted negative			N = 450

$$\begin{aligned} \text{ClassificationError} &= \frac{FP + FN}{TN + TP + FN + FP} \\ &= \frac{7+17}{400+26+17+7} \\ &= 0.060 \end{aligned}$$

မြန်မာစာတော်

**Recall, or True Positive Rate (TPR): what fraction of all positive instances does the classifier correctly identify as positive?**

True negative	TN = 400	FP = 7	
True positive	FN = 17	TP = 26	
Predicted negative	Predicted positive		N = 450

$$\text{Recall} = \frac{TP}{TP+FN}$$

ມີກົດຈຳນວດຕ່າງໆ  
 $= \frac{26}{26+17}$   
 $= 0.60$   
 ດັ່ງນີ້ແມ່ນ positive

Recall is also known as:

- True Positive Rate (TPR)
- Sensitivity
- Probability of detection

acc ≥ 0.95 recall ≥ 0.60 ດັ່ງນີ້ແມ່ນ Model  
 o ດັ່ງນີ້ແມ່ນ class not anomalous

Solution predicted in positive

Precision: what fraction of positive predictions are correct?

True negative	TN = 400	FP = 7	
True positive	FN = 17	TP = 26	
	Predicted negative	Predicted positive	N = 450

$$\text{Precision} = \frac{TP}{TP+FP}$$

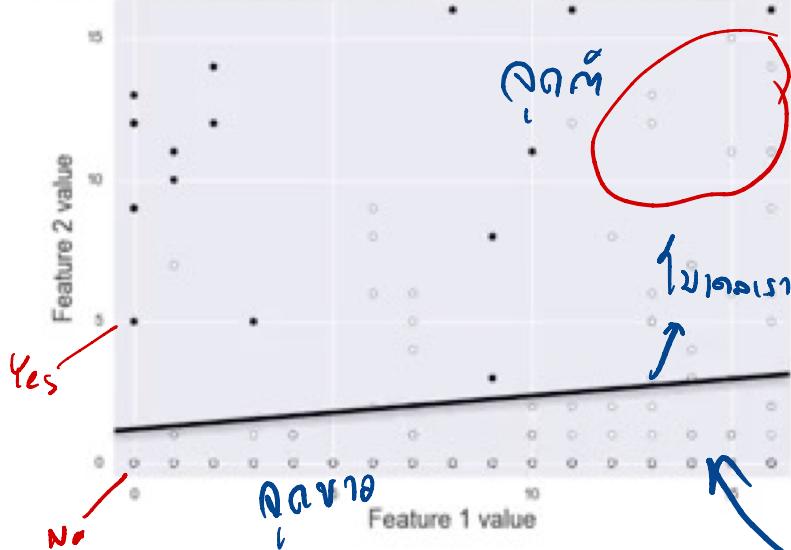
$$= \frac{26}{26+7}$$

$$= 0.79$$

all in  
positive

# Low Precision, High Recall

digits dataset: positive class (black) is digit 1, negative class (white) all others



	$FP \rightarrow 0\text{--}1\text{--}0\text{--}2$
$TN = 408$	$FP = 27$
$FN = 0$	$TP = 15$

$$\text{Precision} = \frac{TP}{TP+FP} = \frac{15}{42} = 0.36$$

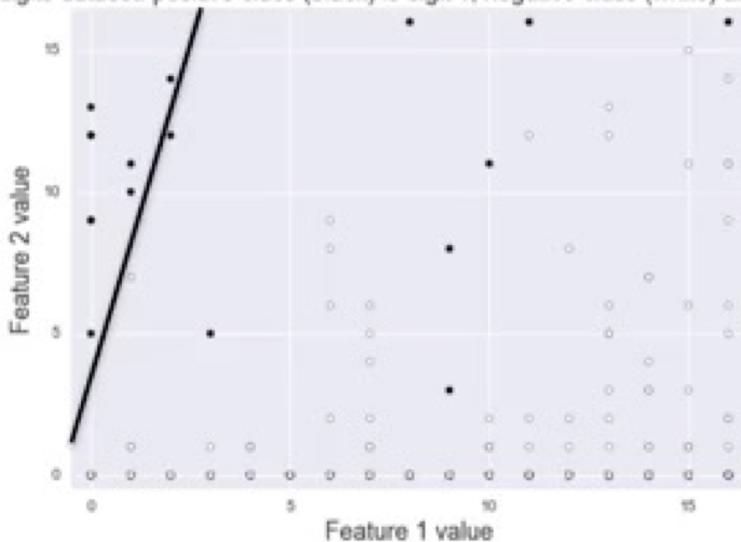
$$\text{Recall} = \frac{TP}{TP+FN} = \frac{15}{15} = 1.00$$

ไม่เกิดข้อผิดพลาดในชุดทดสอบ  
ผลลัพธ์ที่ได้ 100% ถูกต้อง

ผลลัพธ์ที่ได้ 100% ถูกต้อง

# High Precision, Lower Recall

digits dataset: positive class (black) is digit 1, negative class (white) all others



TN = 435	FP = 0
FN = 8	TP = 7

$$\text{Precision} = \frac{TP}{TP+FP} = \frac{7}{7} = 1.00$$

$$\text{Recall} = \frac{TP}{TP+FN} = \frac{7}{15} = 0.47$$

แม้จะมี FP 0 แต่ FN มาก

# Recall v.s. Precision

- งานที่เน้น Recall

- คืองานที่ต้องการตรวจจับตัวอย่างที่เป็น YES ให้ได้เยอะที่สุด
- ความเสียหายจากการปล่อยกรณีที่เป็น YES หลุดไปนั้นสูงมาก
- เช่น ตรวจจับเซลล์มะเร็ง เครื่องตรวจจับเพลิงไหม้
- ยอมให้ตรวจจับกรณี NO เป็น YES ได้ คัดกรองออกกีโดยผู้เชี่ยวชาญ

- งานที่เน้น Precision

- คืองานที่ต้องการผลการกำนาย YES ให้ถูกต้องที่สุด
- ความเสียหายจากการกำนาย NO เป็น YES นั้นสูงมาก
- เช่น Search Engine, Recommendation System, งานที่ต้องตอบกับผู้ใช้โดยตรง
- ยอมให้กรณีที่เป็น YES จริงหลุดรอดจากการตรวจจับไปได้ → ไม่ recommend หนังที่เราอาจชอบบันทึกไว้ในสิ่งของหนังที่ recommend ที่ชอบมากที่สุด

*minimizing precision or recall*

F-score: generalizes F1-score for combining precision & recall into a single number

Geometric Mean  $\rightarrow \sqrt{a \cdot b}$   
 $a, b \in [0, 1]$

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 \cdot TP}{2 \cdot TP + FN + FP}$$

*adjusting the formula for F<sub>β</sub>*

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{(\beta^2 \cdot \text{Precision}) + \text{Recall}} = \frac{(1 + \beta^2) \cdot TP}{(1 + \beta^2) \cdot TP + \beta \cdot FN + FP}$$

$\beta$  allows adjustment of the metric to control the emphasis on recall vs precision:

- Precision-oriented users:  $\beta = 0.5$  (false positives hurt performance more than false negatives)
- Recall-oriented users:  $\beta = 2$  (false negatives hurt performance more than false positives)

# Model Evaluation for Supervised Learning

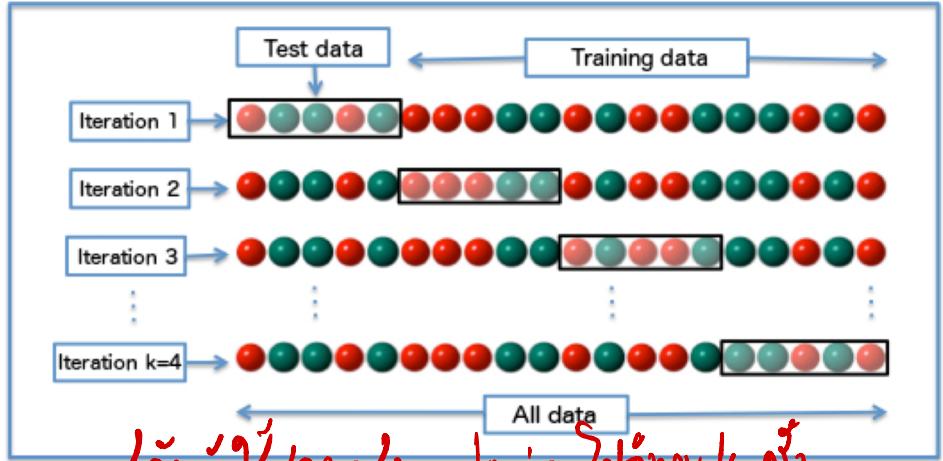
- Model Evaluation គឺ ការវัดປະສົກຮັກພາມໂນເດລດ້ວຍເຕັນິຄຕ່າງໆ ເພື່ອກຳໄຫ້ມື່ນໃຈວ່າ ໂມເດລສາມາດກຳຈານໄດ້ດີເນື້ອນຳໄປໃຫ້ກັບຂ້ອມູລໃນອນາຄຕ **ໄຟສິນໃຫ້ຂໍ້ມູນໃຫ້ກຸ່ນ**
  - ປ້ອງທາກີ່ເຮົາໄມ່ມີຂ້ອມູລໃນອນາຄຕມາກດສອບ (600)
- Hold Out
  - ເຕັນິຄກາຣແບ່ງຂ້ອມູລເປັນສອງຊຸດ ເຮັດວຽກວ່າຊຸດຝຶກ (Training Set) ເພື່ອໃຫ້ສ້າງໂນເດລ ແລະ ຊຸດ ກດສອບ (Test Set) ເພື່ອວັດປະສົກຮັກພາມຂອງໂນເດລ
  - ໂນເດລທີ່ສ້າງຈາກຊຸດຝຶກຈະໄມ່ເຄຍເຫັນຂ້ອມູລຈາກຊຸດກດສອບ ດັ່ງນັ້ນຈຶ່ງພວປະນາຄນໄດ້ວ່າຂ້ອມູລຊຸດ ກດສອບເປັນຂ້ອມູລໃນອນາຄຕ

ແປງເຕັດ

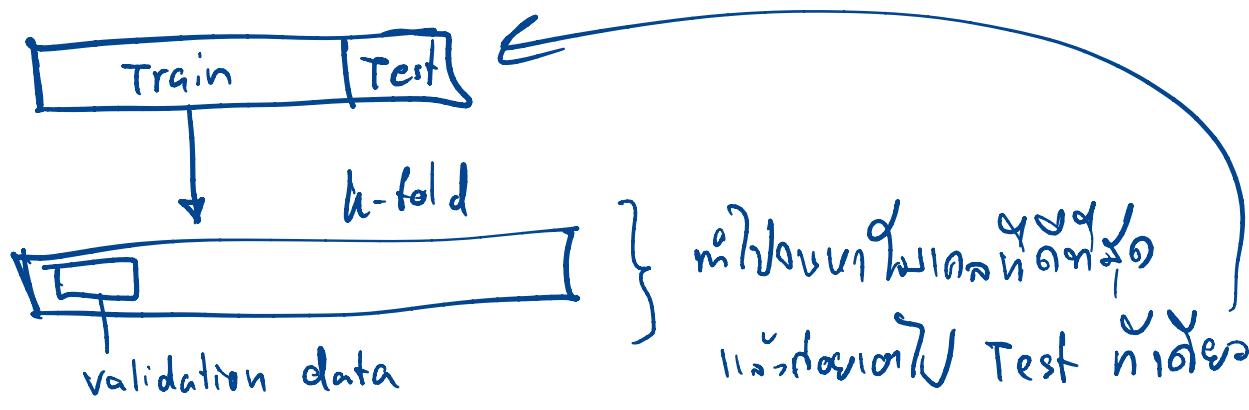
# k-fold Cross-validation

- ปัญหา: ผู้สร้างโมเดลอาจ “จูน” โมเดลไปเรื่อยๆ จนใช้งานกับ Test Set ได้ดี แต่อาจใช้กับข้อมูลจริงในอนาคตได้ไม่ดี
- การทำ k-fold Cross-validation จะแก้ปัญหานี้ โดย
  - แบ่งข้อมูลเป็น  $k$  ชุด จะสร้างโมเดลจาก  $k-1$  ชุดและใช้ Validation อีก 1 ชุด (นิยมใช้  $k = 10$ )
  - สร้างโมเดลและทดสอบ  $k$  รอบ แต่ละรอบจะเปลี่ยน Training กับ Validation Set ไปเรื่อยๆ ไม่ซ้ำกัน
  - เมื่อทดสอบครบ  $k$  ครั้ง จะคำนวณค่าเฉลี่ยผลลัพธ์ของประสิทธิภาพที่ได้
  - สร้างโมเดลด้วยวิธีที่ดีที่สุดโดยใช้ข้อมูลทั้งหมด และค่อยนำไปทดสอบกับ Test Set แค่ครั้งเดียว

[https://en.wikipedia.org/wiki/Cross-validation\\_\(statistics\)#/media/File:K-fold\\_cross\\_validation\\_EN.jpg](https://en.wikipedia.org/wiki/Cross-validation_(statistics)#/media/File:K-fold_cross_validation_EN.jpg)



หลักการใช้ชุดทดสอบ , train ไปซ้ำๆ กัน



အသုံး  
sliding window

# Overfitting / Underfitting

- **Overfitting:**

- โมเดลที่สร้างได้ มีความอ่อนไหวต่อข้อมูลมากเกินไป
- เปลี่ยนค่าบางค่าของข้อมูลเข้าเล็กน้อยจะทำให้คำนวณเปลี่ยนไปอย่างมาก
- เกิดจาก การที่โมเดลเรียนรู้ noise ของข้อมูลเข้าไปด้วย
- เช็คได้จากค่าของมาตรฐานความถูกต้องสูงมากเมื่อทดสอบกับ training set แต่ต่ำมากเมื่อทดสอบกับ test set

- **Underfitting**

- โมเดลที่สร้างได้เก็บ pattern ของข้อมูลได้ไม่ครบถ้วน ไม่ลงทะเบียนพอ
- ค่าของมาตรฐานความถูกต้องของโมเดลมีค่าต่ำมากก็การทดสอบกับ training และ test set

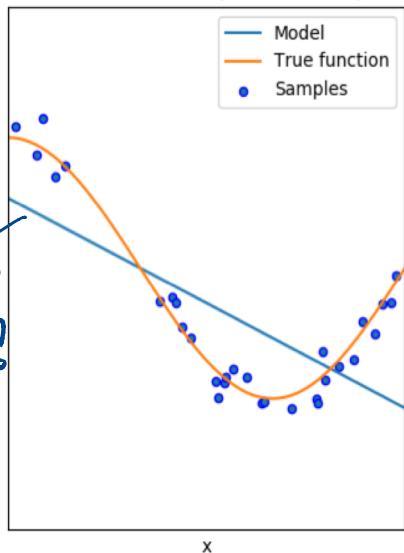
# Overfitting / Underfitting

underfit (ดี, pattern ไม่ถูก)

Degree 1  
MSE = 4.08e-01 (+/- 4.25e-01)

Model  
True function  
Samples

ไม่สามารถอธิบายข้อมูลได้



Degree 4  
MSE = 4.32e-02 (+/- 7.08e-02)

Model  
True function  
Samples

y

x

y

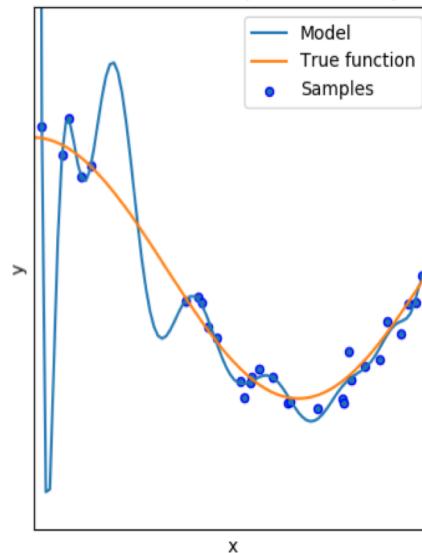
x



overfit! โมเดล สามารถอธิบายข้อมูลได้มากเกินไป แต่ทำให้เกิดความผันผวนสูง ค่า MSE มาก

Degree 15  
MSE = 1.82e+08 (+/- 5.45e+08)

Model  
True function  
Samples



Source: [http://scikit-learn.org/stable/\\_images/sphx\\_glr\\_plot\\_underfitting\\_overfitting\\_001.png](http://scikit-learn.org/stable/_images/sphx_glr_plot_underfitting_overfitting_001.png)