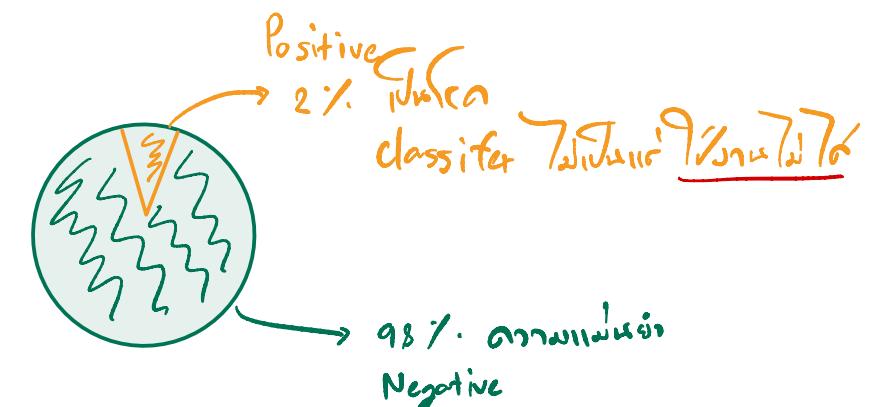


Model Evaluation and Selection

- Evaluation metrics
 - How can we measure accuracy?
 - Other metrics to consider?
- Use **validation test set** of class-labeled tuples instead of training set when assessing accuracy
- Methods for estimating a classifier's accuracy
 - Holdout method
 - Cross-validation
 - Bootstrap
- Comparing classifiers:
 - ROC Curves



Classifier Evaluation Metrics: Confusion Matrix

□ Confusion Matrix:

Actual class\Predicted class	C_1	$\neg C_1$
C_1	True Positives (TP)	False Negatives (FN)
$\neg C_1$	False Positives (FP)	True Negatives (TN)

Precision
↓

Recall
→

- In a confusion matrix w. m classes, $CM_{i,j}$ indicates # of tuples in class i that were labeled by the classifier as class j
- May have extra rows/columns to provide totals
- Example of Confusion Matrix:

test

Actual class\Predicted class	buy_computer = yes	buy_computer = no	Total
buy_computer = yes	positive 6954	negative 46	7000
buy_computer = no	positive 412	negative 2588	3000
Total	7366	2634	10000

Classifier Evaluation Metrics: Accuracy, Error Rate, Sensitivity and Specificity

A\P	C	$\neg C$	
C	TP	FN	P
$\neg C$	FP	TN	N
	P'	N'	All

- Classifier accuracy, or recognition rate
 - Percentage of test set tuples that are correctly classified
accuracy
$$\text{Accuracy} = (\text{TP} + \text{TN})/\text{All}$$
- Error rate: $1 - \text{accuracy}$, or
$$\text{Error rate} = (\text{FP} + \text{FN})/\text{All}$$

- Class imbalance problem
 - One class may be *rare*
 - E.g., fraud, or HIV-positive
 - Significant *majority of the negative class* and minority of the positive class
 - Measures handle the class imbalance problem
 - **Sensitivity** (recall): True positive recognition rate
 - $\text{Sensitivity} = \text{TP}/\text{P}$
 - **Specificity**: True negative recognition rate
 - $\text{Specificity} = \text{TN}/\text{N}$

Classifier Evaluation Metrics: Precision and Recall, and F-measures

- **Precision:** Exactness: what % of tuples that the classifier labeled as positive are actually positive?

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

model made positive predictions

- **Recall:** Completeness: what % of positive tuples did the classifier label as positive?

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

model identified positive documents

- Range: $[0, 1]$
- The “inverse” relationship between precision & recall
- **F measure (or F-score):** harmonic mean of precision and recall
- In general, it is the weighted measure of precision & recall

$$F_{\beta} = \frac{1}{\alpha \cdot \frac{1}{P} + (1 - \alpha) \cdot \frac{1}{R}} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

Assigning β times as much weight to recall as to precision)

- **F1-measure (balanced F-measure)**

- That is, when $\beta = 1$,

$$F_1 = \frac{2PR}{P + R}$$

Precision
Recall

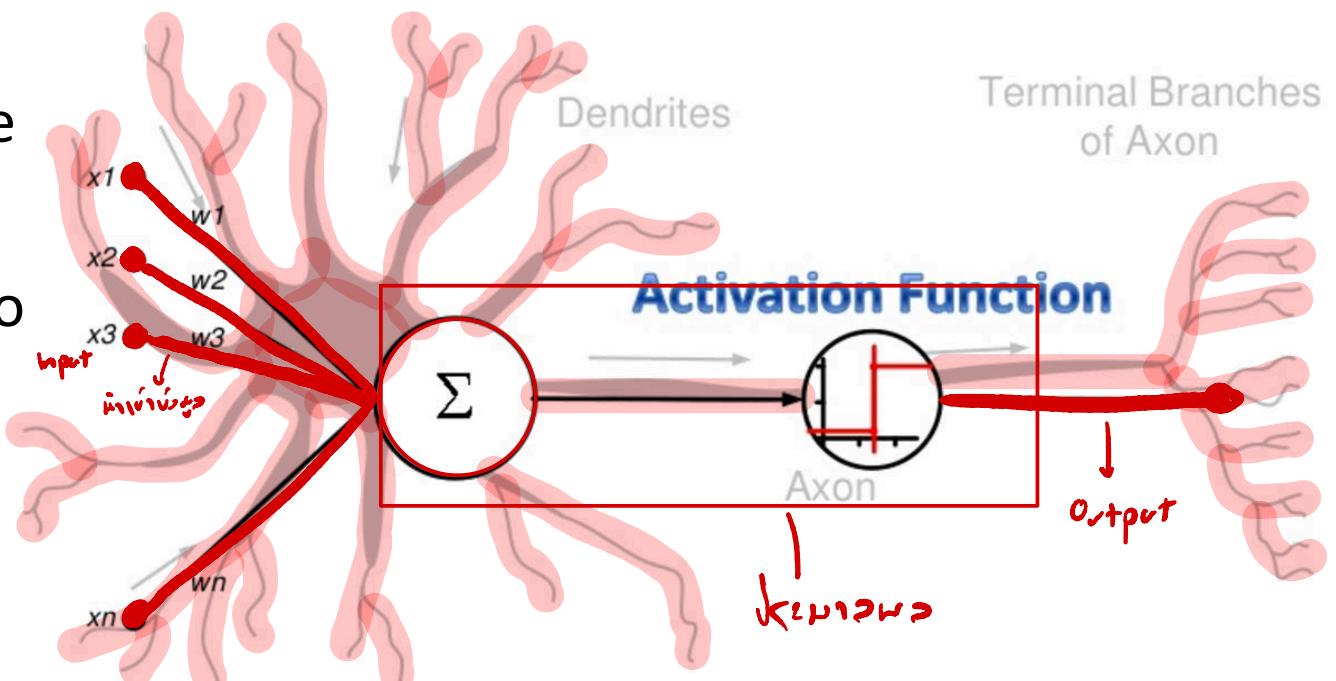
Chapter 9. Classification: Advanced Methods

- Bayesian Belief Networks
- Support Vector Machines
Support Vector Machines
- Neural Networks and Deep Learning
- Pattern-Based Classification
- Lazy Learners and K-Nearest Neighbors
- Other Classification Methods
- Summary



Neural Network for Classification

- Started by psychologists and neurobiologists to develop and test computational analogues of neurons
- A neural network: A set of connected input/output units where each connection has a **weight** associated with it
- During the learning phase, the **network learns by adjusting the weights** so as to be able to predict the correct class label of the input tuples

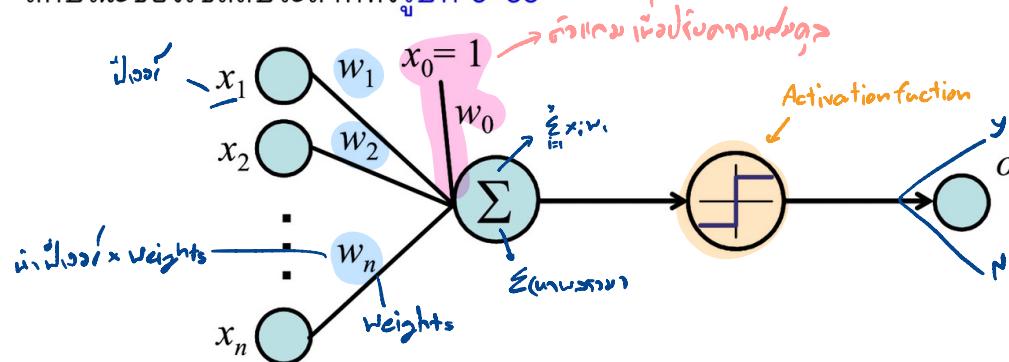


Artificial Neural Networks as an analogy of Biological Neural Networks

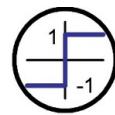
6.7.1 เพอร์เซปตรอน

เพอร์เซปตรอน (perceptron) เป็นข่ายงานประสาทเทียมแบบง่ายมีหน่วยเดียวที่จำลอง

ลักษณะของเซลล์ประสาทดังรูปที่ 6-35



พังก์ชันกราฟตุ้น



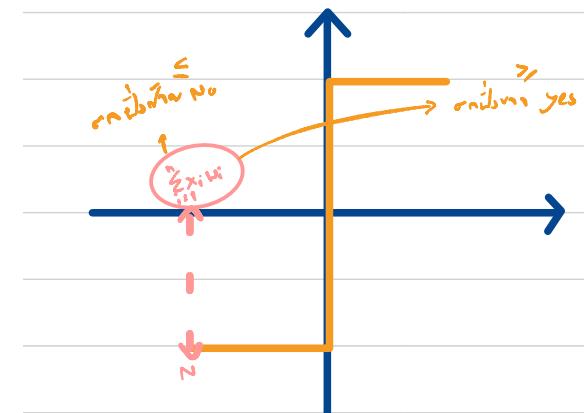
ในรูปแสดง พังก์ชันกราฟตุ้น (activation function) ชนิดที่เรียกว่า พังก์ชันสองขั้ว (bipolar function) ซึ่งแสดงผลของเอาต์พุตเป็น 1 กับ -1 พังก์ชันกราฟตุ้นอื่นๆ ที่นิยมใช้ก็อย่างเช่น พังก์ชันไบนารี (binary function) ซึ่งแสดงผลของเอาต์พุตเป็น 1 กับ 0 และเขียน



แทนด้วยรูป

เราสามารถแสดงเอาต์พุต (o) ในรูปของพังก์ชันของอินพุต (x_1, x_2, \dots, x_n) ได้ดังนี้

$$o(x_1, x_2, \dots, x_n) = \begin{cases} 1 & \text{if } w_1x_1 + w_2x_2 + \dots + w_nx_n > \theta \\ -1 & \text{if } w_1x_1 + w_2x_2 + \dots + w_nx_n < \theta \end{cases} \quad (6.7)$$



กระบวนการ
การหานุสูต

Algorithm: Perceptron-Learning-Rule

1. Initialize weights w_i of the perceptron.
2. UNTIL the termination condition is met DO
 - 2.1 FOR EACH training example DO
 - Input the example and compute the output.
 - Change the weights if the output from the perceptron is not equal to the target output using the following rule.

$$w_i \leftarrow w_i + \Delta w_i$$

penalty

$$\Delta w_i \leftarrow \alpha(t-o)x_i$$

learning rate

output of model

input x_0, x_1, x_2

o.s target-label mismatch

where t , o and α are the target output, the output from the perceptron and the learning rate, respectively.

6.7.2 ตัวอย่างการเรียนฟังก์ชัน AND และ XOR ด้วยกฎการเรียนรู้เพอร์เซปตรอน

พิจารณาตัวอย่างการเรียนรู้ของเพอร์เซปตรอนโดยจะให้เรียนรู้ฟังก์ชัน 2 ฟังก์ชัน ฟังก์ชัน แรกคือฟังก์ชัน AND แสดงในตารางที่ 6-18 ในกรณีนี้เราริเริ่มฟังก์ชันเป็นฟังก์ชันกระตุ้น

ตารางที่ 6-18 ฟังก์ชัน AND(x_1, x_2)

x_1	x_2	ผล
0	0	0
0	1	0
1	0	0
1	1	1

$$\begin{aligned}
 T \wedge T &= T \\
 T \wedge F &= F \\
 F \wedge T &= F \\
 F \wedge F &= F
 \end{aligned}$$

ฟังก์ชัน AND ตามตารางด้านบนจะให้ค่าที่เป็นจริงก็ต่อเมื่อ x_1 และ x_2 เป็นจริงทั้งคู่ (ถ้าที่สมมติเอาร์พุตเป้าหมาย) ผลการใช้กฎการเรียนรู้เพอร์เซปตรอนกับฟังก์ชัน AND แสดงในตารางที่ 6-19

ตารางที่ 6-19 ผลการเรียนรู้ฟังก์ชัน AND โดยกฎการเรียนรู้เพอร์เซปตรอน

ลักษณะของการเรียนรู้

1 epoch
1 row

Perceptron Learning Example - Function AND											
		Bias Input $x_0=+1$		$\sum_{i=0}^n (x_i \cdot y_i)$		Alpha = 0.5		Learning rate		target -	
Input	Input			Net Sum	Target	Actual	Alpha*	Weight Values			
x_1	x_2	$1.0 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2$	= Input		Output	Output	Error	w_0	w_1	w_2	
0	0	0.10	0.00	0.00	0.10	0	1	-0.50	-0.40	0.10	0.10
0	1	-0.40	0.00	0.10	-0.30	0	0	0.00	-0.40	0.10	0.10
1	0	-0.40	0.10	0.00	-0.30	0	0	0.00	-0.40	0.10	0.10
1	1	-0.40	0.10	0.10	-0.20	1	0	0.50	0.10	0.60	0.60
0	0	0.10	0.00	0.00	0.10	0	1	-0.50	-0.40	0.60	0.60
0	1	-0.40	0.00	0.60	0.20	0	1	-0.50	-0.90	0.60	0.10
1	0	-0.90	0.60	0.00	-0.30	0	0	0.00	-0.90	0.60	0.10
1	1	-0.90	0.60	0.10	-0.20	1	0	0.50	-0.40	1.10	0.60
0	0	-0.40	0.00	0.00	-0.40	0	0	0.00	-0.40	1.10	0.60
0	1	-0.40	0.00	0.60	0.20	0	1	-0.50	-0.90	1.10	0.10
1	0	-0.90	1.10	0.00	0.20	0	1	-0.50	-1.40	0.60	0.10
1	1	-1.40	0.60	0.10	-0.70	1	0	0.50	-0.90	1.10	0.60
0	0	-0.90	0.00	0.00	-0.90	0	0	0.00	-0.90	1.10	0.60
0	1	-0.90	0.00	0.60	-0.30	0	0	0.00	-0.90	1.10	0.60
1	0	-0.90	1.10	0.00	0.20	0	1	-0.50	-1.40	0.60	0.60
1	1	-1.40	0.60	0.60	-0.20	1	0	0.50	-0.90	1.10	1.10
0	0	-0.90	0.00	0.00	-0.90	0	0	0.00	-0.90	1.10	1.10
0	1	-0.90	0.00	1.10	0.20	0	1	-0.50	-1.40	1.10	0.60
1	0	-1.40	1.10	0.00	-0.30	0	0	0.00	-1.40	1.10	0.60
1	1	-1.40	0.00	0.60	-0.80	0	0	0.00	-1.40	1.10	0.60
0	1	-1.40	0.00	0.60	-0.80	0	0	0.00	-1.40	1.10	0.60
1	0	-1.40	1.10	0.00	-0.30	0	0	0.00	-1.40	1.10	0.60
1	1	-1.40	1.10	0.60	0.30	1	1	0.00	-1.40	1.10	0.60