Last Class

- Artificial Intelligence
  - An artificial machine that acts and thinks like humans
- 기호주의 인공지능 vs 연결주의 인공지능, 그리고 인지주의 인공지능 (장병탁 교수)

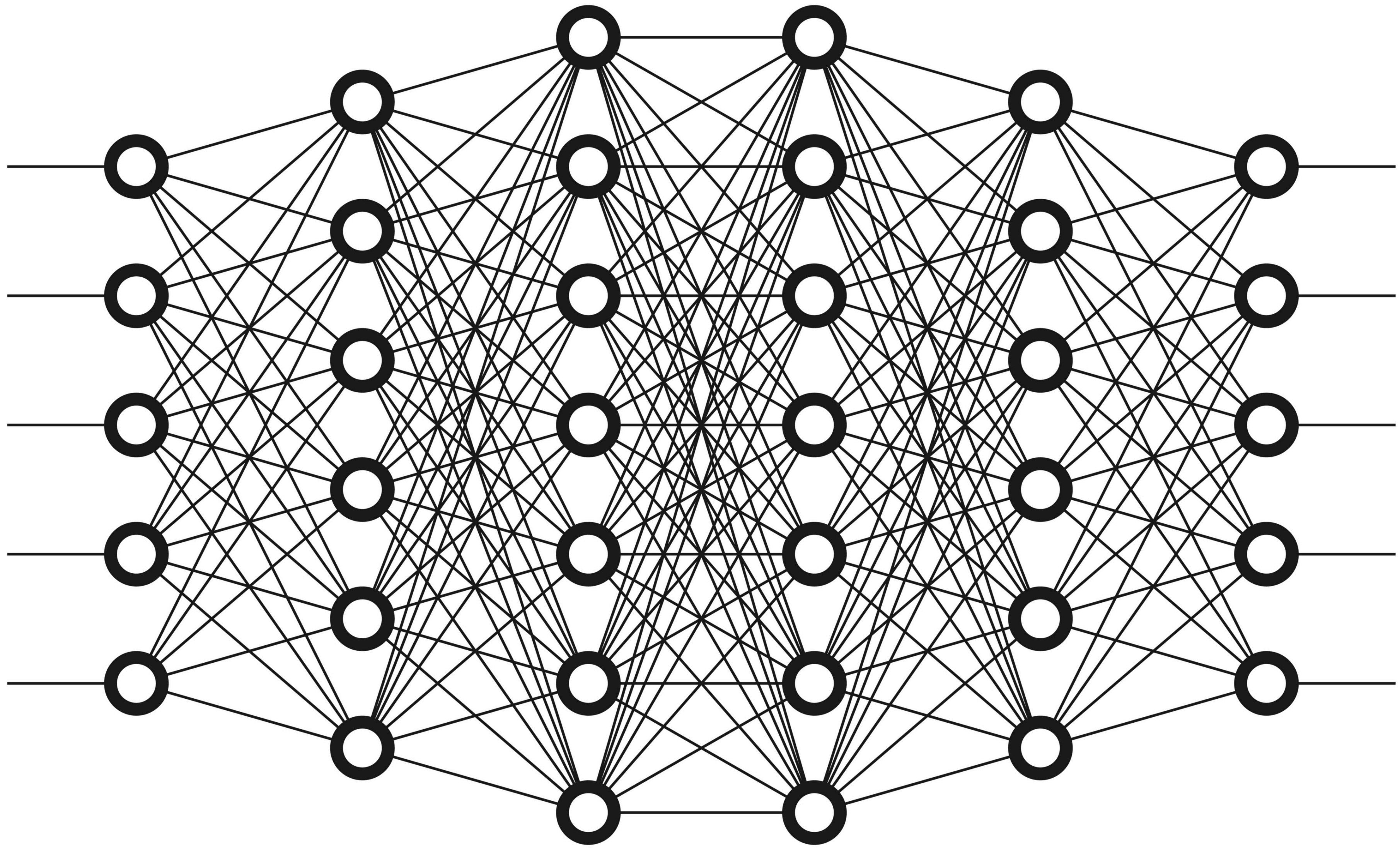
	신경망	규칙기반 인공지능
영역	연결주의	기호주의
도구	뉴런의 연결	기호와 규칙
대표적 개발자	로젠블럿, 힌턴 등	민스키와 매카시 등
핵심 기술	머신러닝, 딥러닝	규칙기반의 추론
응용 분야	음성인식 등 패턴인식	문제 해결, 전문가 시스템

Assignment 2

- John Searle의 아래의 강연듣고
  - https://www.ted.com/talks/john_searle_our_shared_condition_consciousness
  - 다음 질문에 대한 여러분의 생각을 1페이지(12pt) 이상 작성해서 제출하세요.
  - 우선 의식이 존재한다고 가정하고, 의식은 과연 AI로 구현이 가능할 것인가?
  - 기호주의 AI 혹은 연결주의 AI 중 “의식”을 구현 하기에 더 좋은 방법은

Assignment 3 (Bonus)

- 이 과제는 여러분이 선택적으로 할 수 있습니다. 반드시 제출해야 하는 것은 아닙니다. 하지만 보너스 점수를 받을 수는 있습니다
- 아래의 비디오를 시청하고 다음의 질문에 자신의 생각을 기술하여 제출하세요.
  - 과연 “의식”은 인공지능으로구현하는 것이 가능하다고 생각하는가?
  - 의식을 인공지능으로 구현하는 것이 가능하면, 인공지능과 복제인간간 차이는무엇인가? 과연 복제인간도 인공지능이라 할 수 있을것인가?
- 뇌, 현실, 그리고 인공지능 (김대식 교수)
  - https://www.youtube.com/watch?v=tKLRQs_nOxM



Introduction to Machine Learning

Jin Hyun Kim

In This Class

- More about AI, Thinking Machine
- Machine Learning vs Classical Programming
- What is Machine Learning?
- Simple Learning Algorithm using Perceptron

References

- 장교수의 딥러닝 - 장병탁, 홍릉출판사
- Python Machine Learning (2nd edition) - Sebastian Raschka and Jared Huffman
  - (한국어) 머신러닝 교과서 with 파이썬, 사이킷런, 텐서플로- 박해선 번역
- Deep Learning from Scratch - 사이토 고기, 한빛미디어

Thinking Machine

Deep Blue

1997

딥 블루는 가능한
모든 경우를 조사하
여 다음 수를 결정

Watson

2011

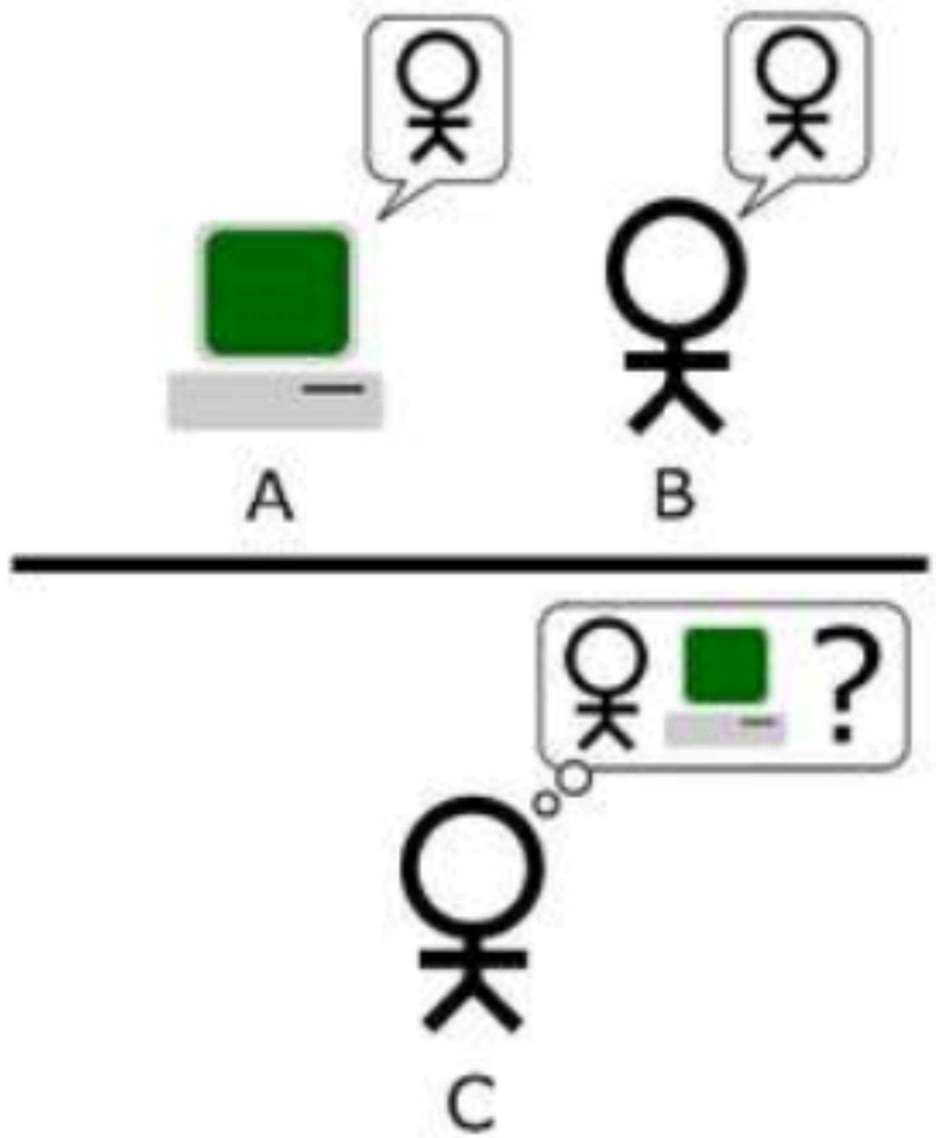
2011년 기능 시험으로서
왓슨은 퀴즈 쇼 제퍼디!에
참가하였으며, 이는 이제
까지도 유일한 인간 대 컴
퓨터 대결

AlphaGo

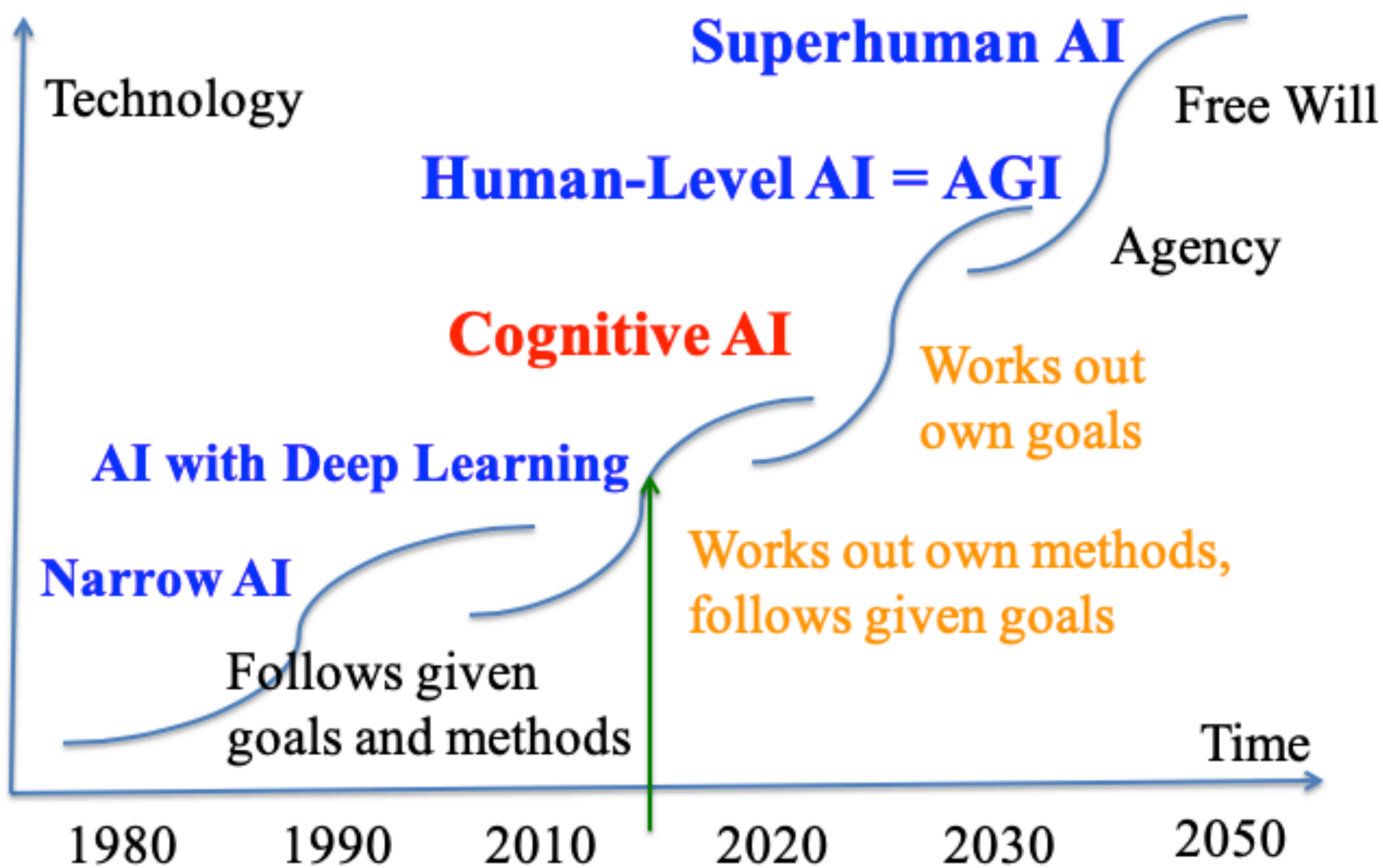
2016

구글(Google)의 딥마인드
(DeepMind Technologies
Limited)가 개발한 인공지능
(AI, Artificial Intelligence)
바둑 프로그램

Turing Test

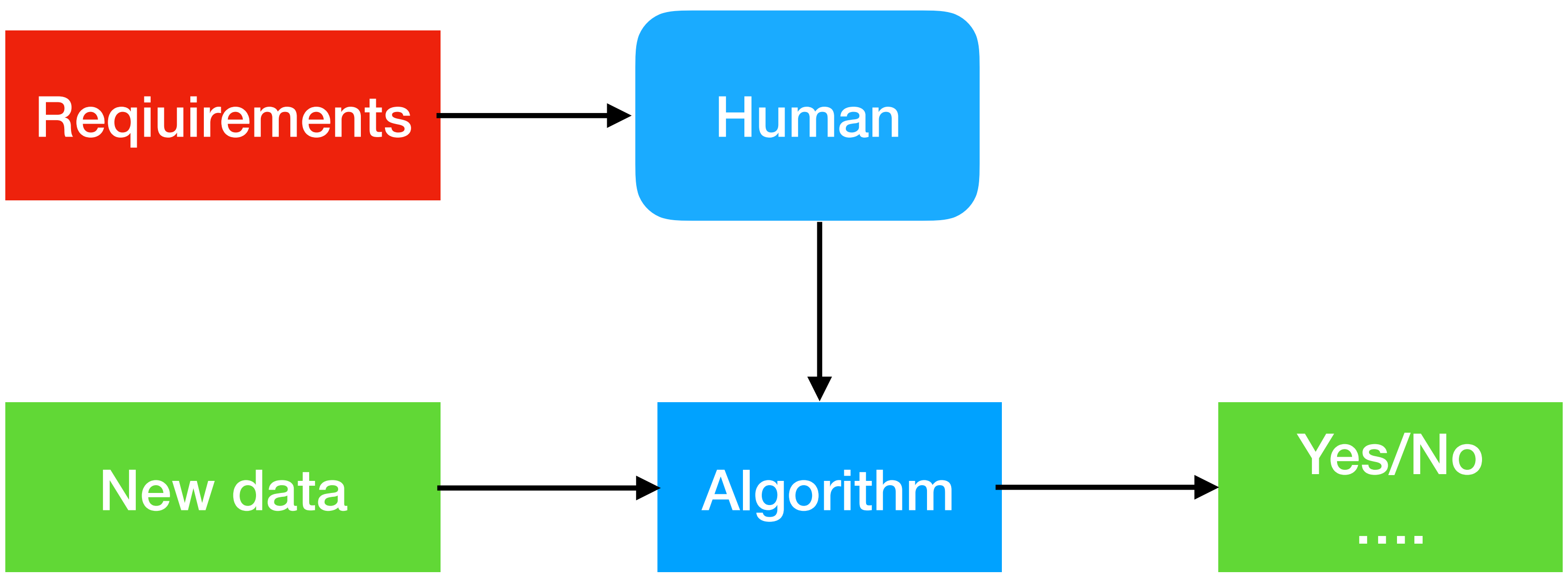


Advancing AI



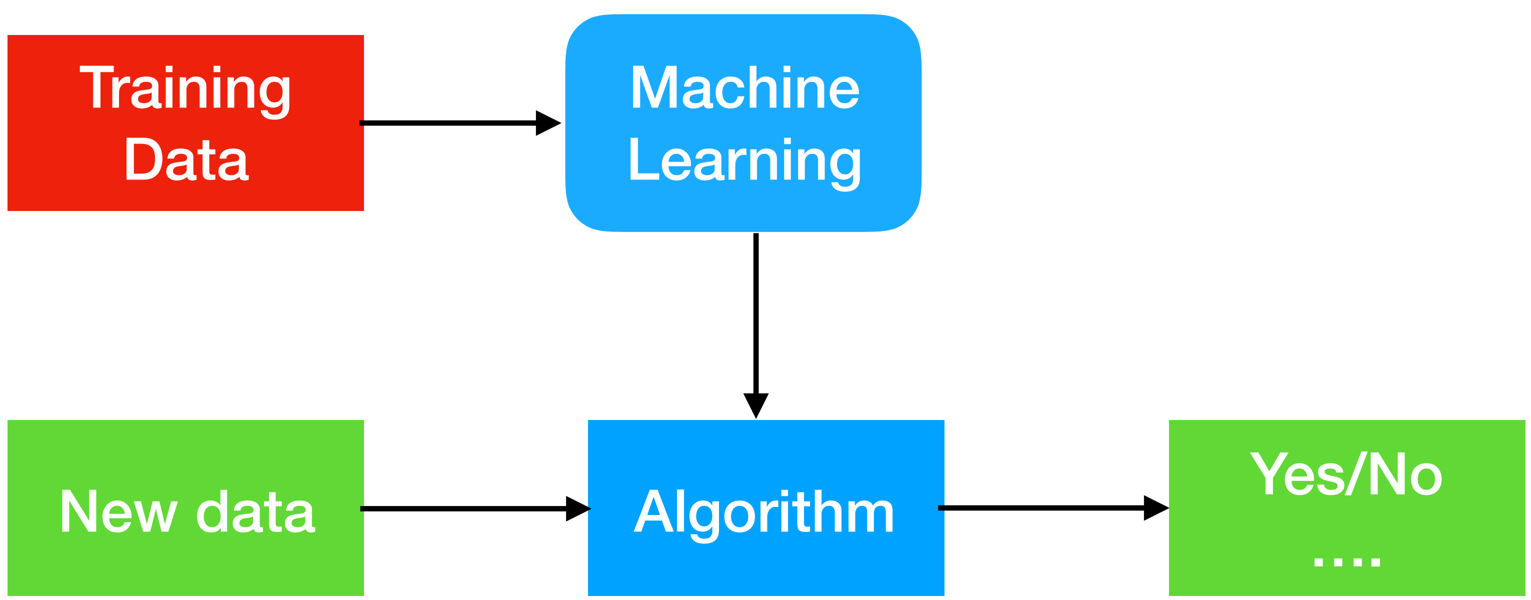
Human-Oriented Algorithm

- Human-oriented algorithm development based on requirements

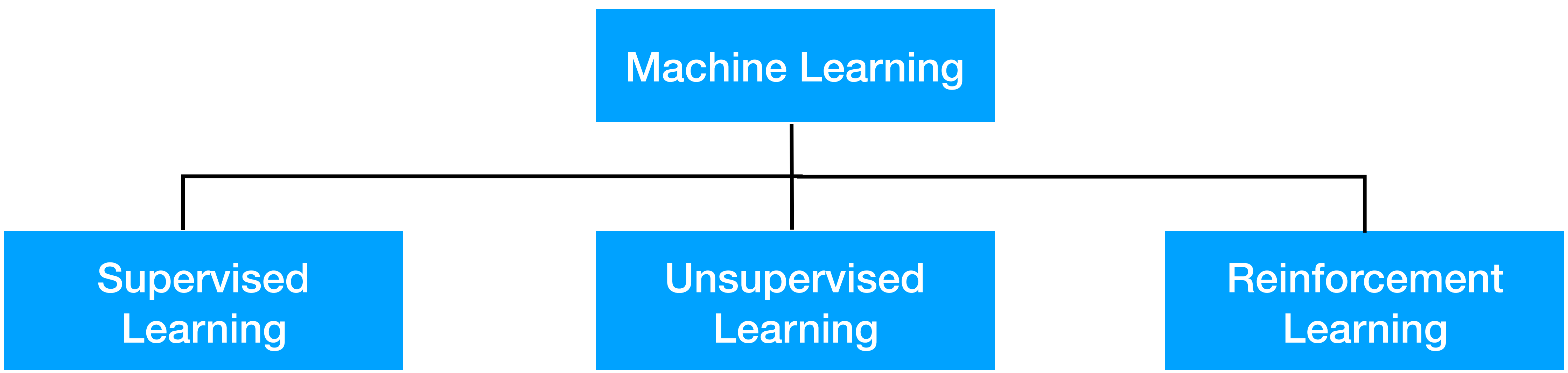


Machine-Oriented Learning

- AI-oriented algorithm synthesis based on data



Machine Learning Classification

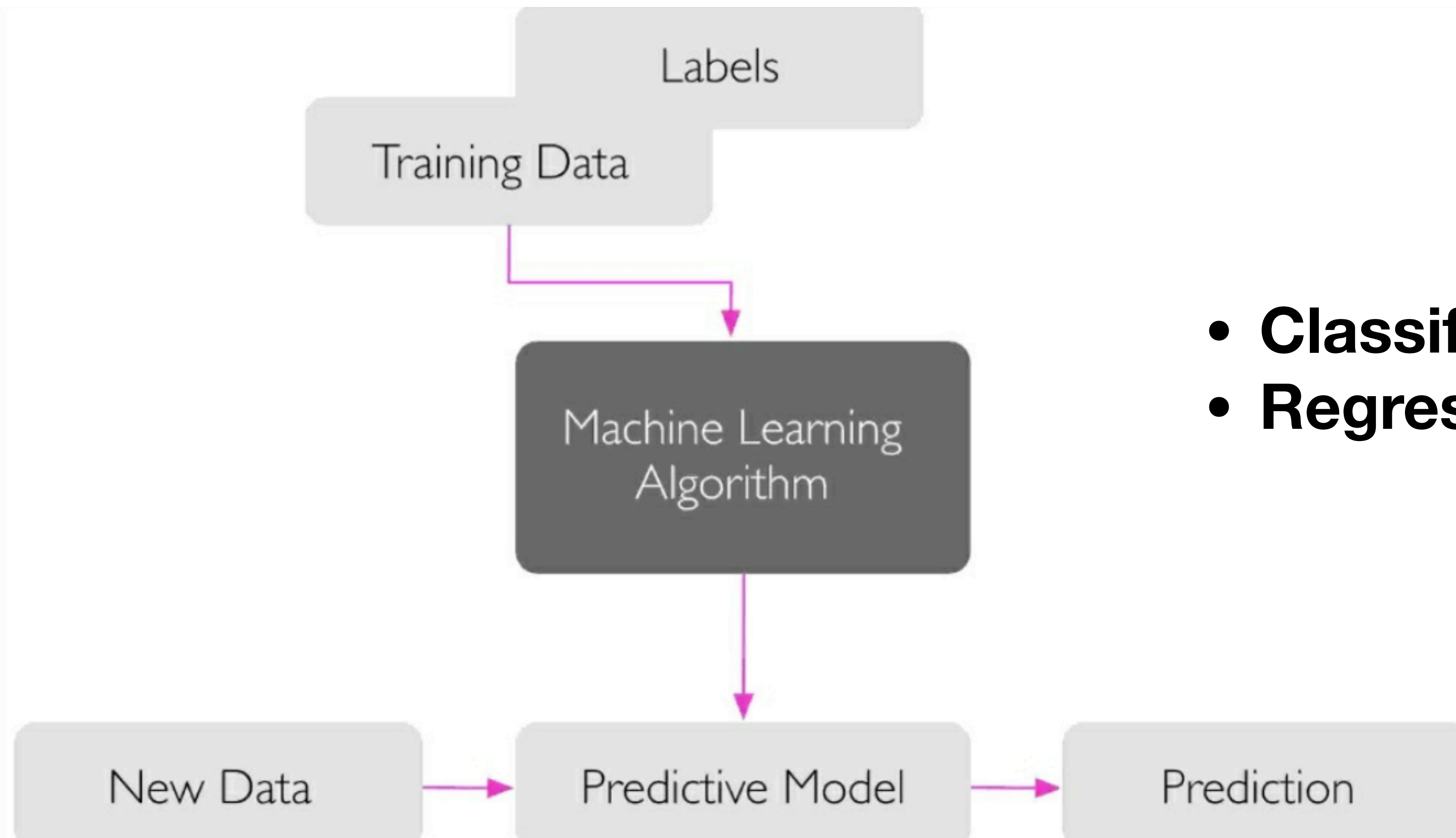


- Labeled data
- Direct feedback
- Predict outcome/future

- No labels
- No feedback
- Find hidden structure in data

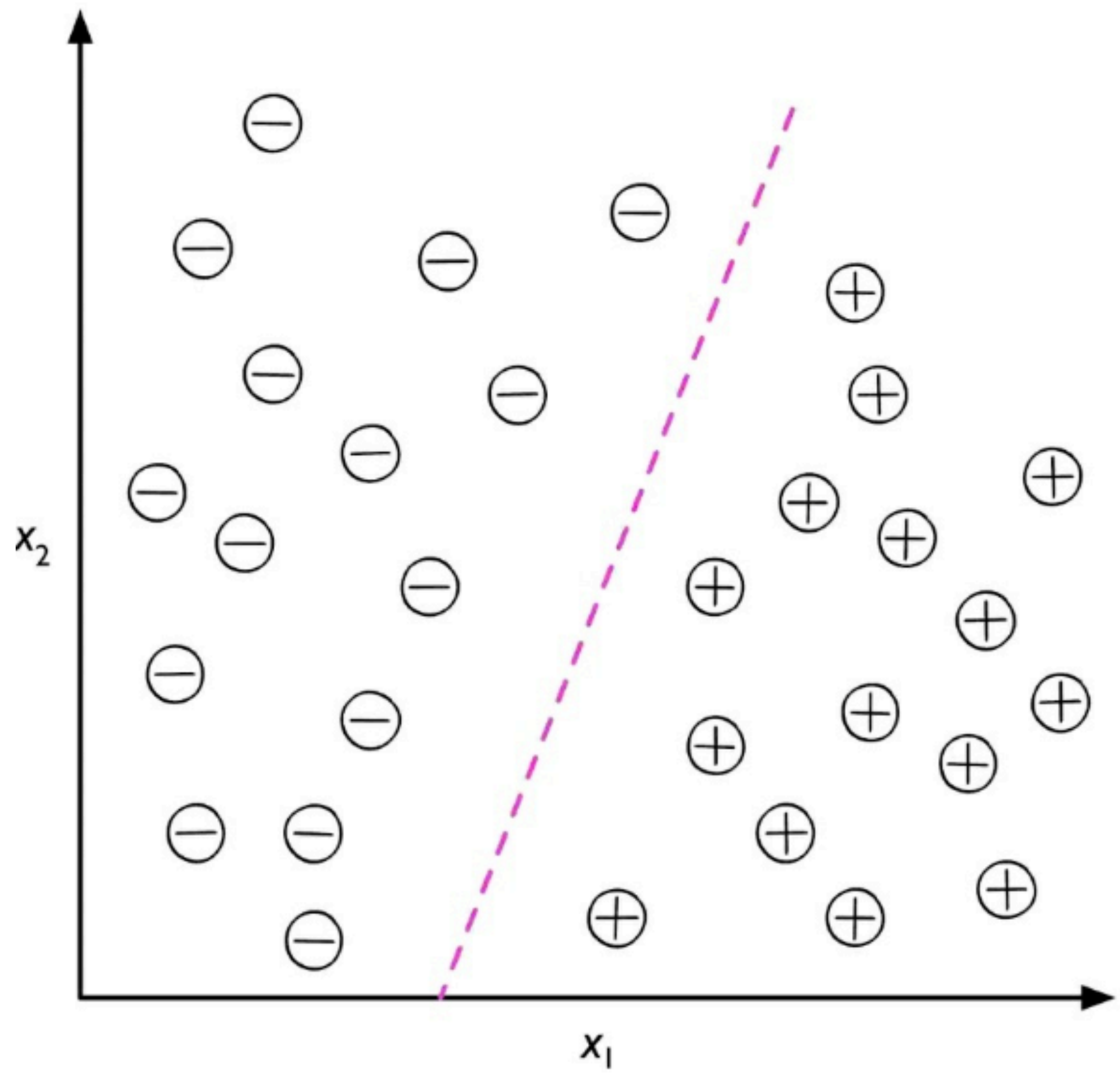
- Decision process
- Reward system
- Learn series of actions

Supervised Learning



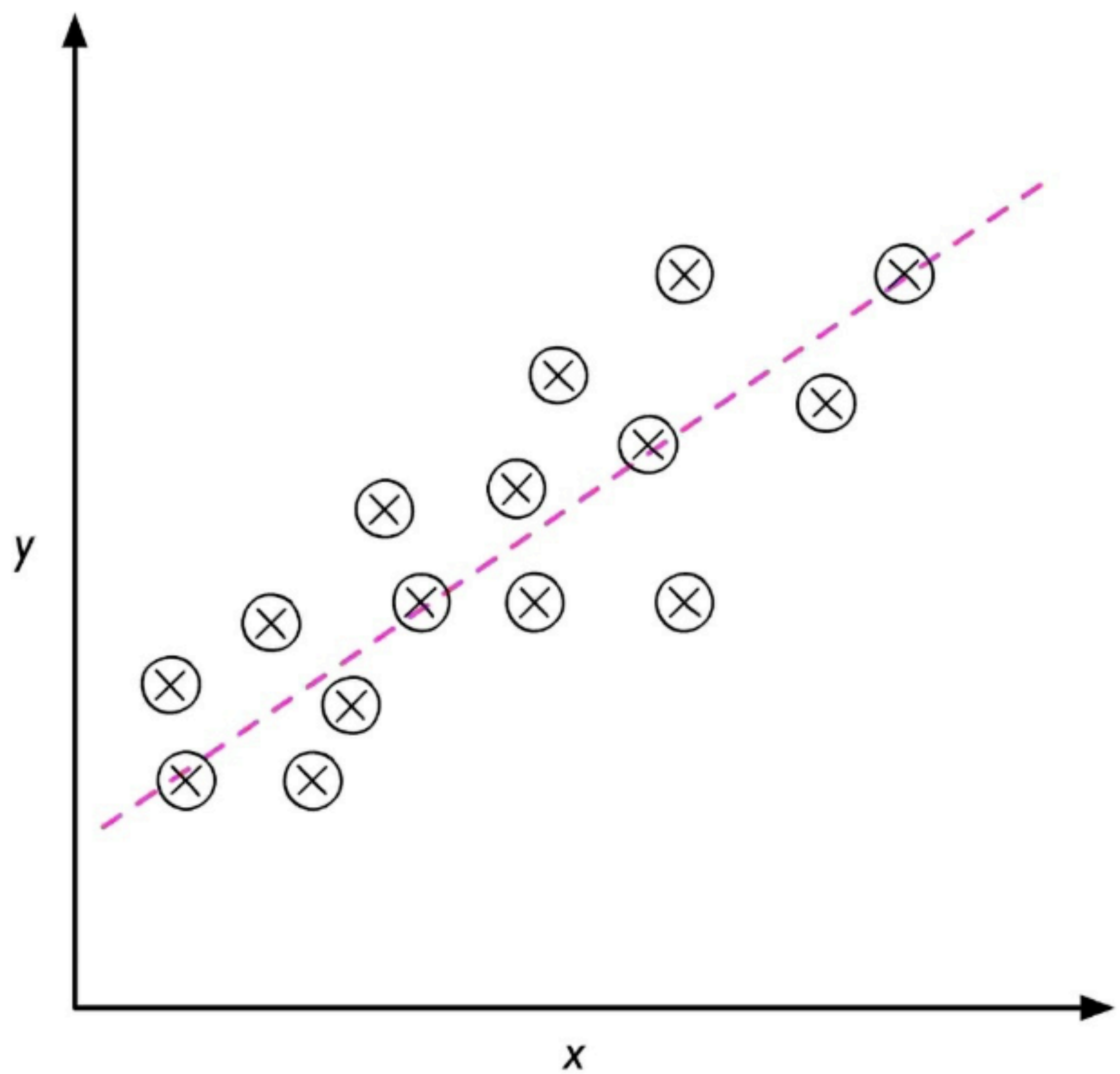
- **Classification**
- **Regression**

Classification



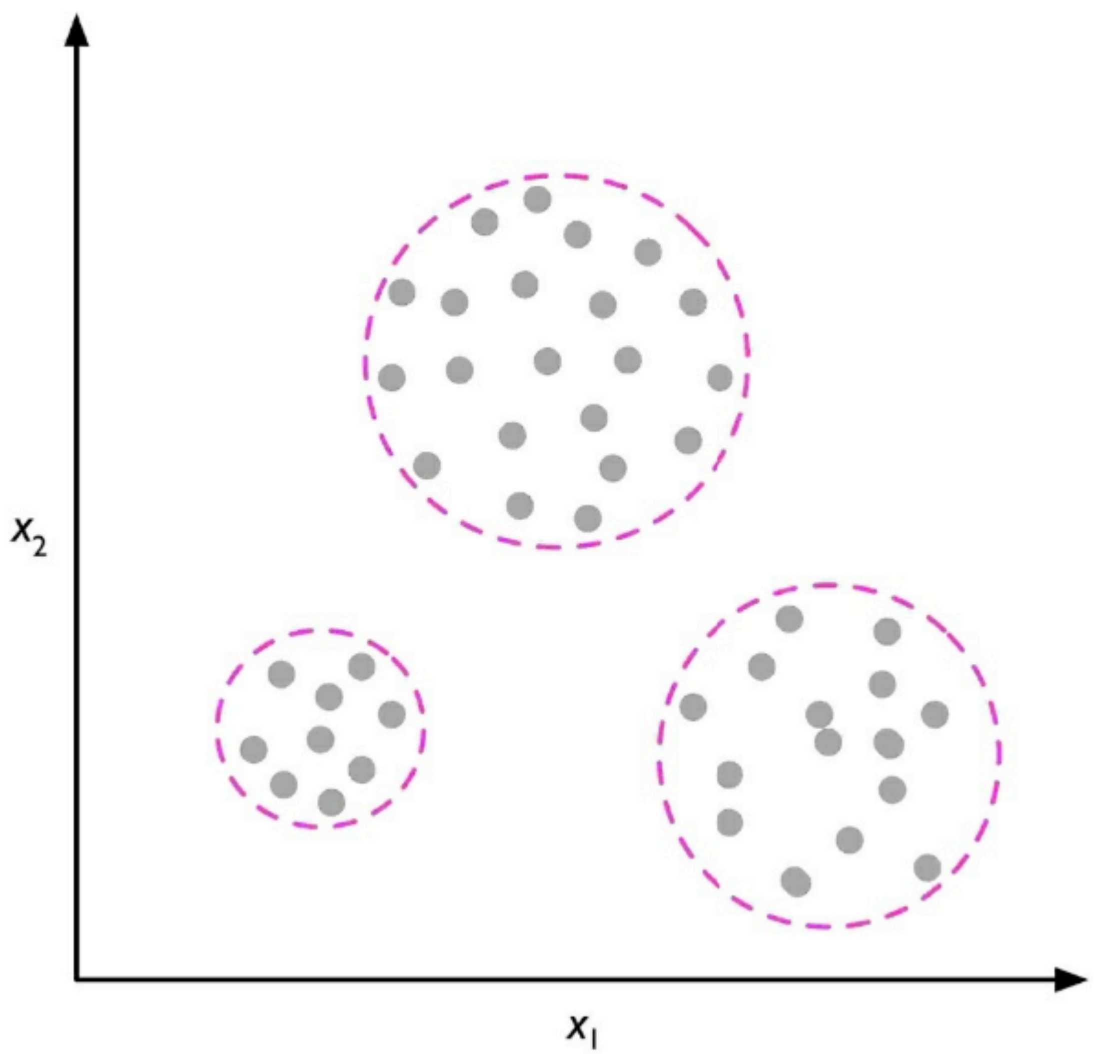
- To predict the categorical class labels of new instances, based on past observations.
- E.g. Spam filter

Regression



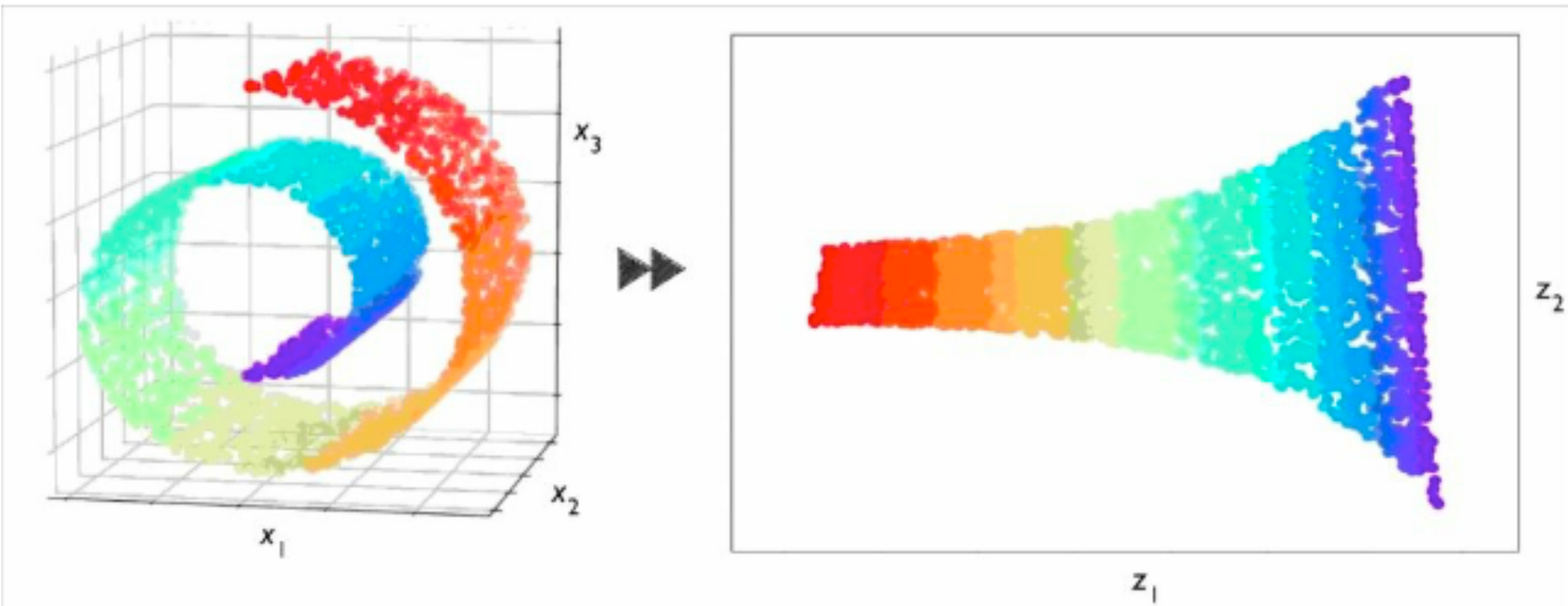
- The prediction of continuous outcomes
- Given a number of predictor (explanatory) variables and a continuous response variable (outcome or target), find a relationship between those variables that allows us to predict an outcome.
- E.g. Predict SAT grade according to study time

Unsupervised Learning

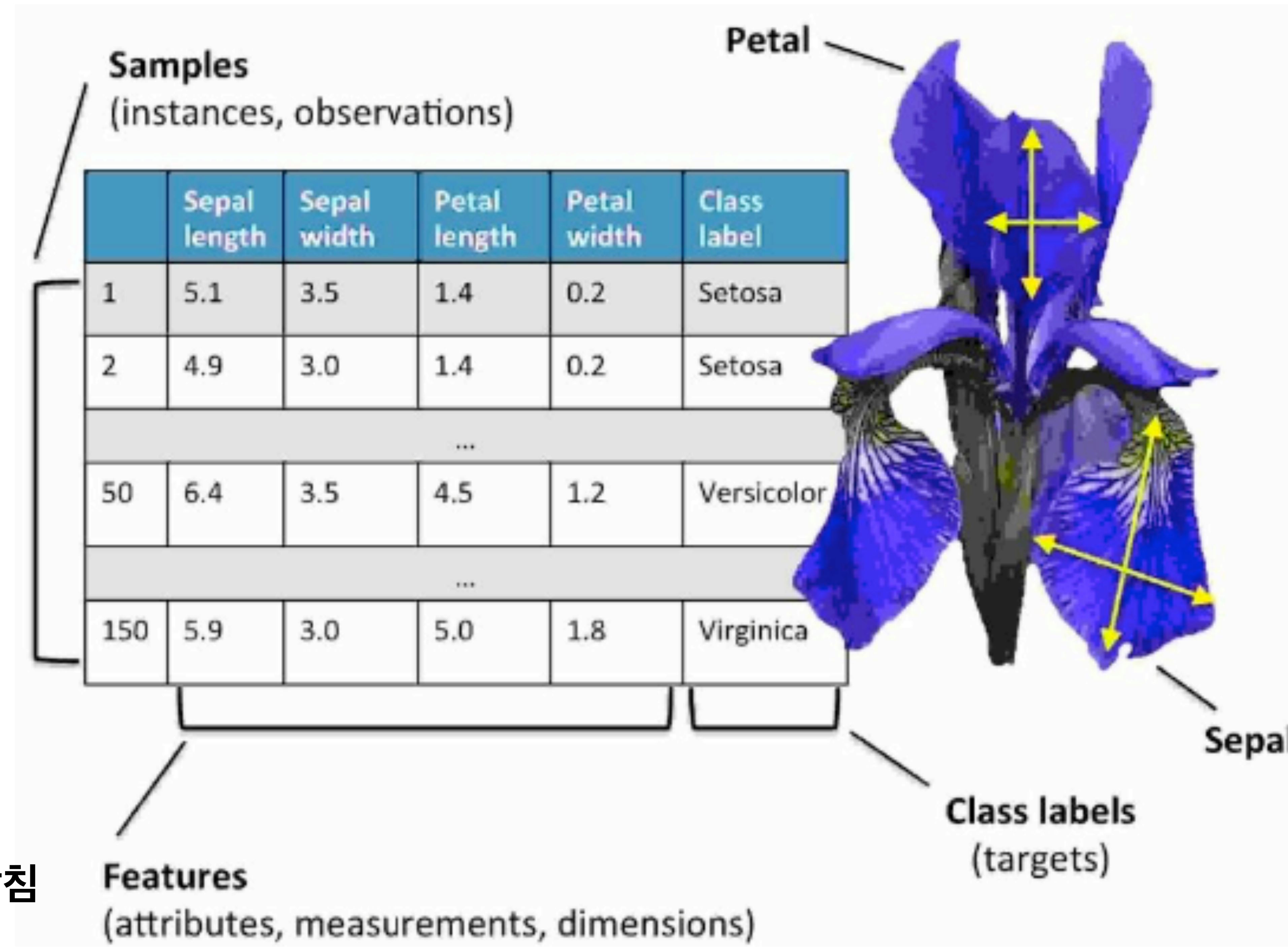


- Clustering
  - An exploratory data analysis technique that organizes a pile of information into **meaningful subgroups** (clusters) without having any prior knowledge of their group memberships.

Unsupervised Learning



Mathematics Notation



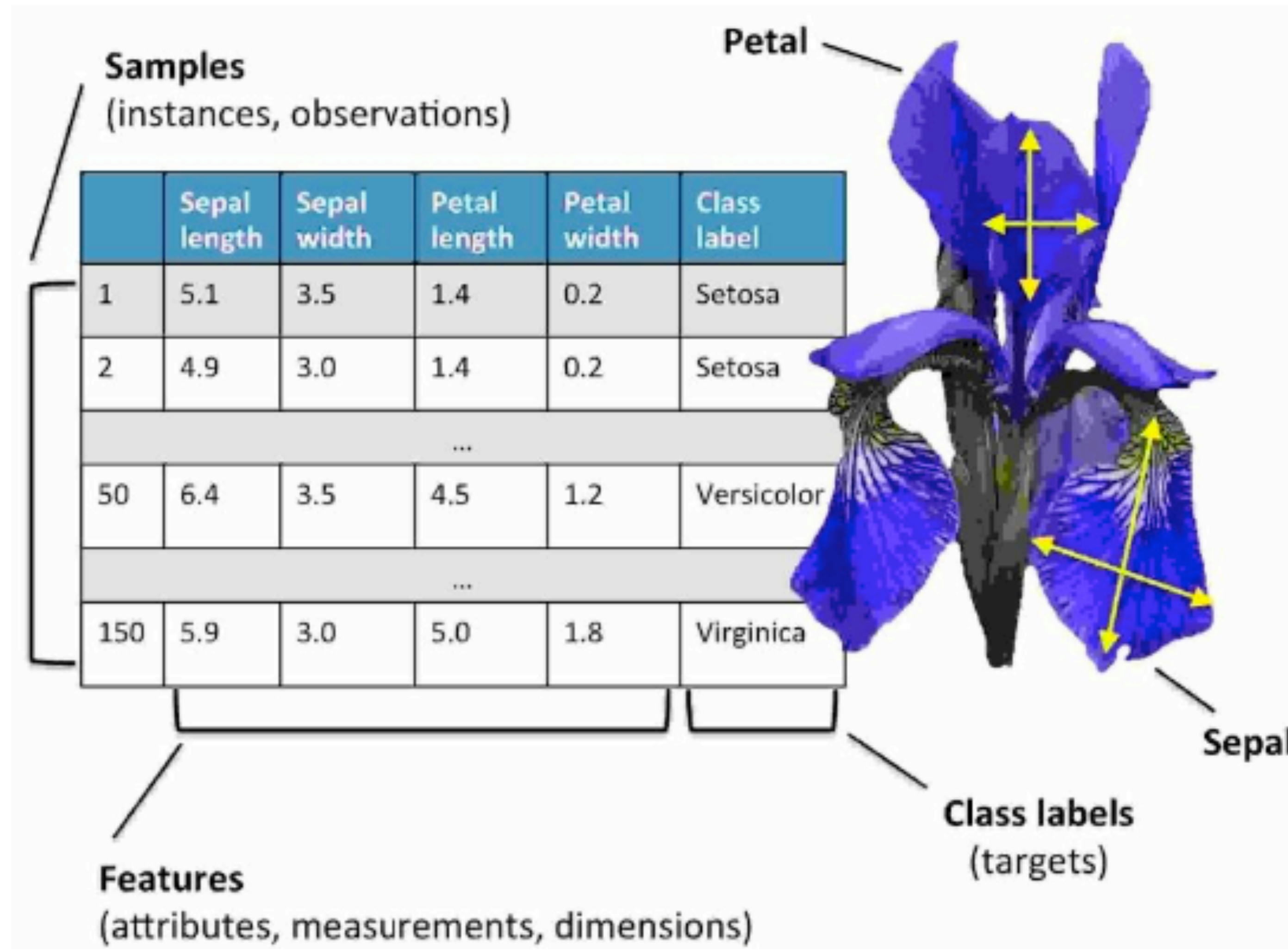
	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica

Sepal: 꽃받침
Petal: 꽃잎

Mathematical Notation

$$\begin{bmatrix} x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & x_4^{(1)} \\ x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & x_4^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{(150)} & x_2^{(150)} & x_3^{(150)} & x_4^{(150)} \end{bmatrix}$$

$$x_j^{(i)}$$



	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica

Mathematical Notation

$$
\begin{array}{l}
 \\
\text{Sample \#1} \\
\text{Sample \#2} \\
 \\
\text{Sample \#150}
\end{array}
\begin{array}{c}
\begin{array}{cccc}
\text{Sepal length} & \text{Sepal width} & \text{Petal length} & \text{Petal width}
\end{array} \\
\begin{bmatrix}
x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & x_4^{(1)} \\
x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & x_4^{(2)} \\
\vdots & \vdots & \vdots & \vdots \\
x_1^{(150)} & x_2^{(150)} & x_3^{(150)} & x_4^{(150)}
\end{bmatrix}
\end{array}
$$

$$x_j^{(i)}$$

The number i of sample with j feature
j 피처를 가진 i번째 샘플

Mathematical Notation

- Target variable

$$y = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ y^{(3)} \\ \vdots \\ y^{(150)} \end{bmatrix} \quad (y \in Setosa, Versicolor, Virginica)$$

Mathematical Notation

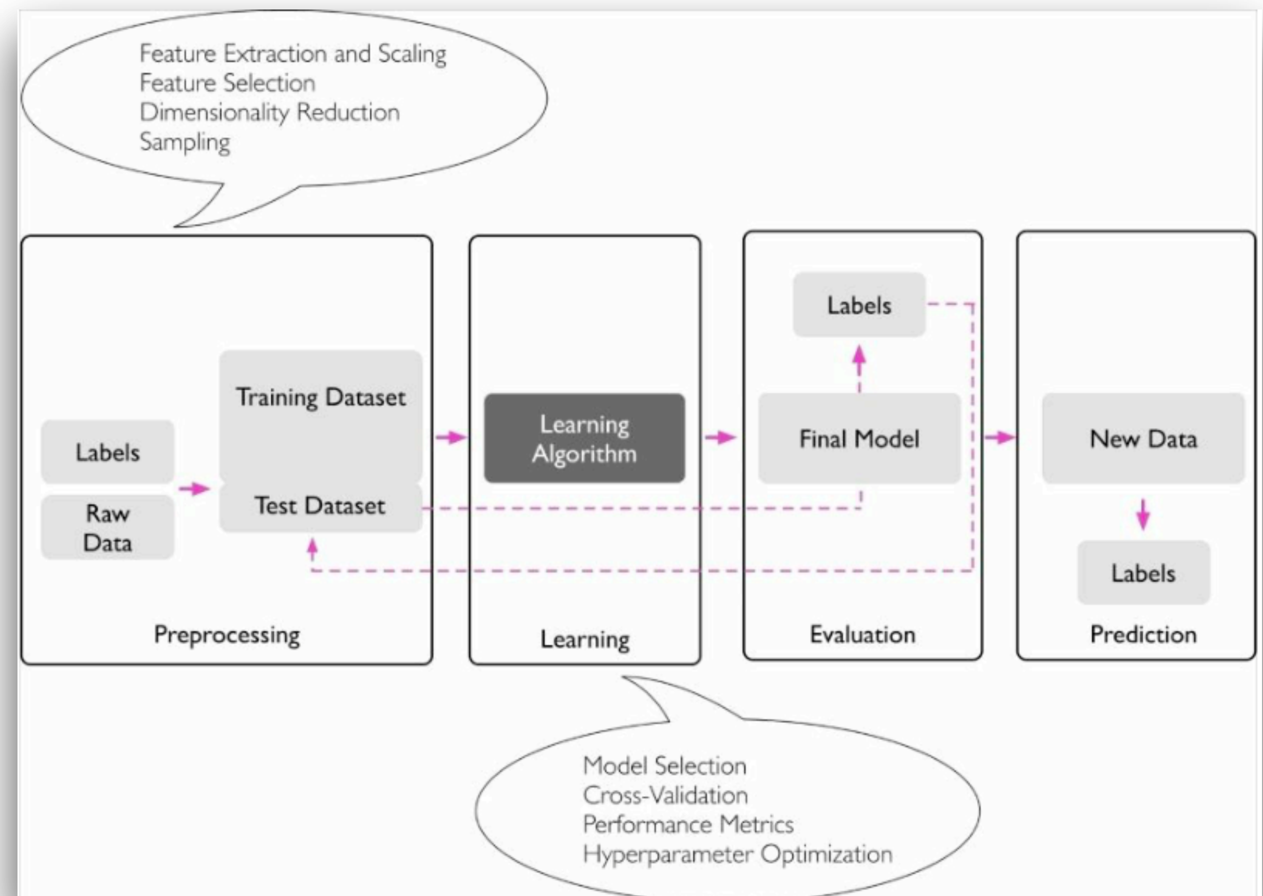
- Row vector

$$x^{(i)} = [x_1^{(i)} \ x_2^{(i)} \ x_3^{(i)} \ x_4^{(i)}]$$

- Column vector

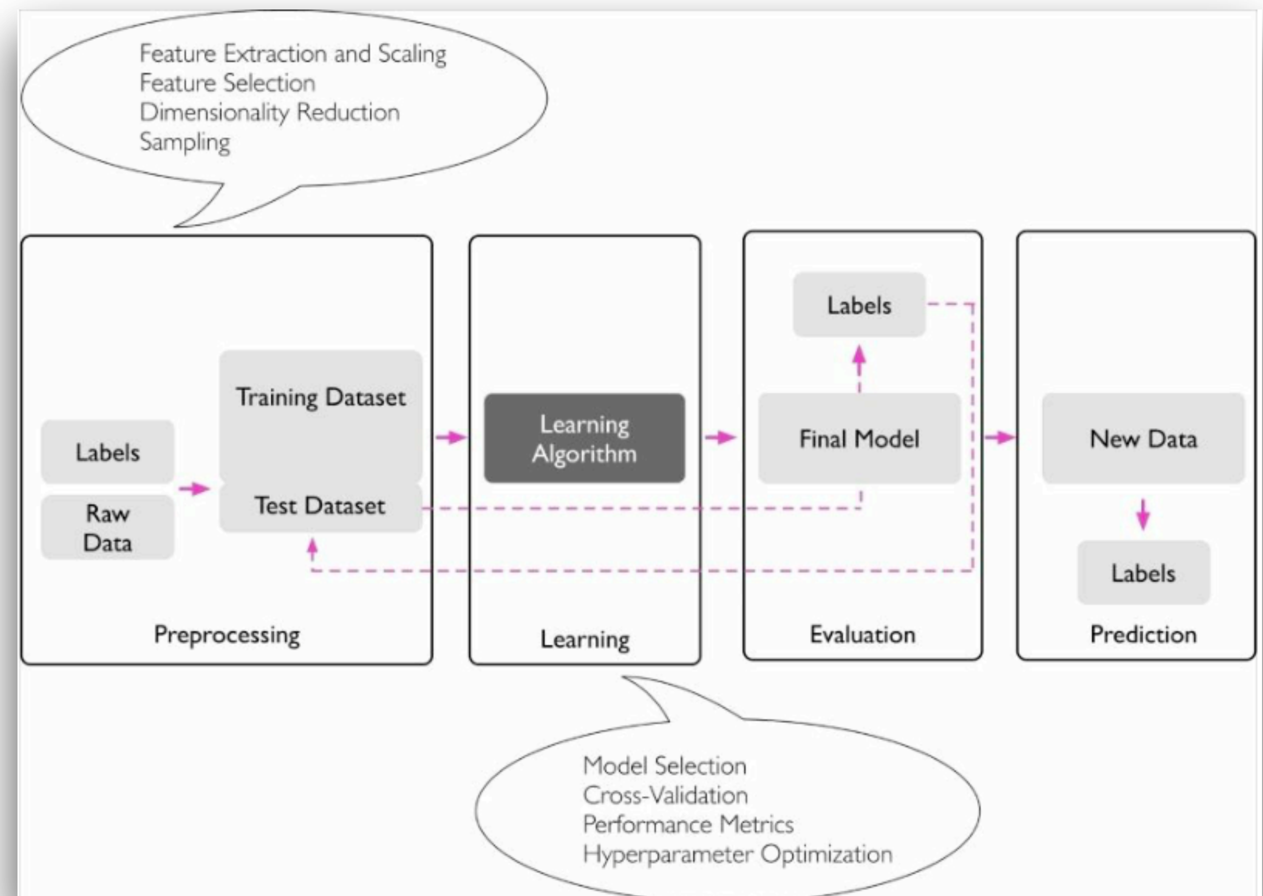
$$x^{(i)} = \begin{bmatrix} x_j^{(1)} \\ x_j^{(2)} \\ x_j^{(3)} \\ \vdots \\ x_j^{(150)} \end{bmatrix}$$

Machine Learning Model Construction



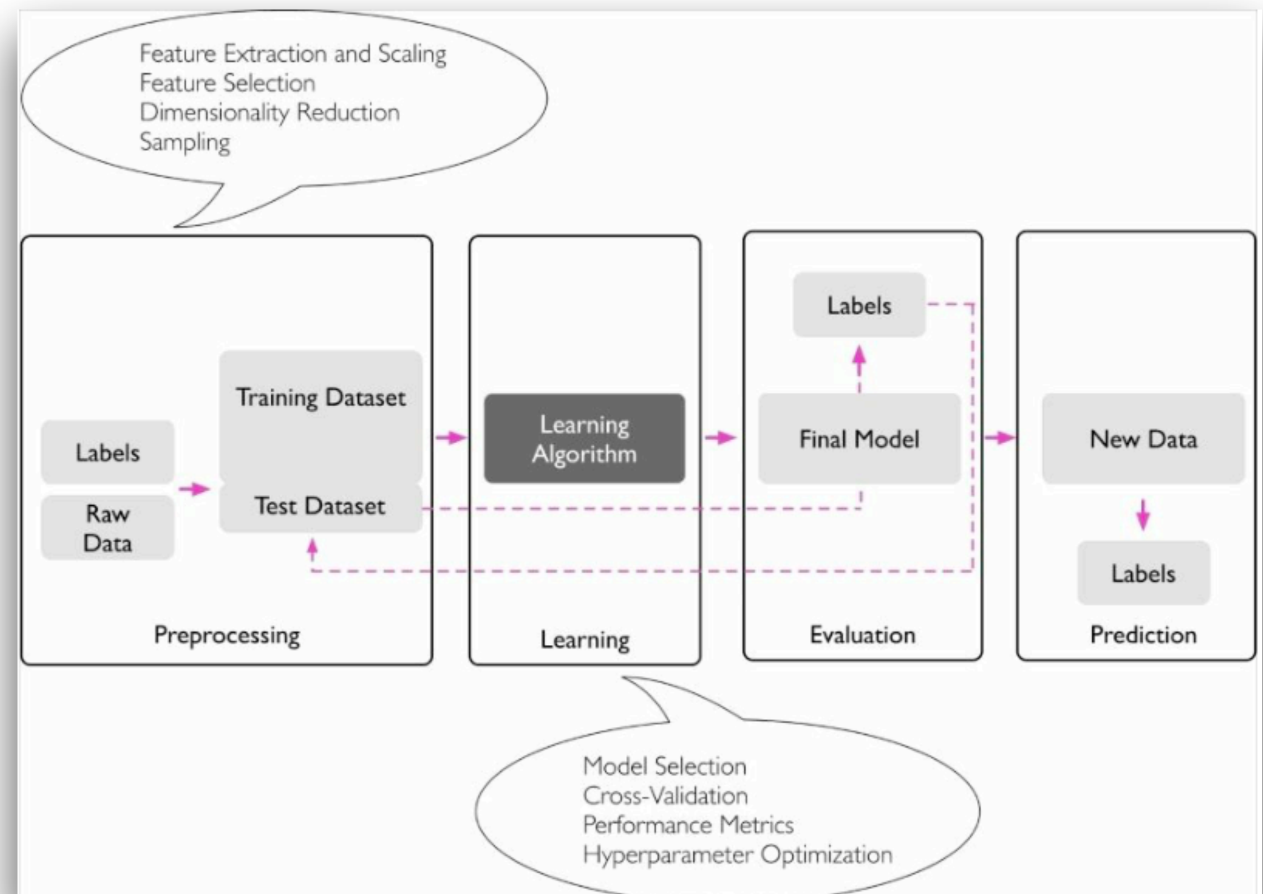
- Feature extraction
  - e.g. For be series of flower images, features could be the color, the hue, the intensity of the flowers, the height, and the flower lengths and widths.

Machine Learning Model Construction



- Feature scaling
  - Transforming the features to be in a common scale, such as [0, 1] or a standard normal distribution with zero mean and unit variance
  - E.g. X has 2 features (house size(0-2000 ft^2), and number of bedrooms(0~5)
    - $x_{houseSize}$ = size(ft^2)/2000
    - x_{rooms} = # of bedrooms/5

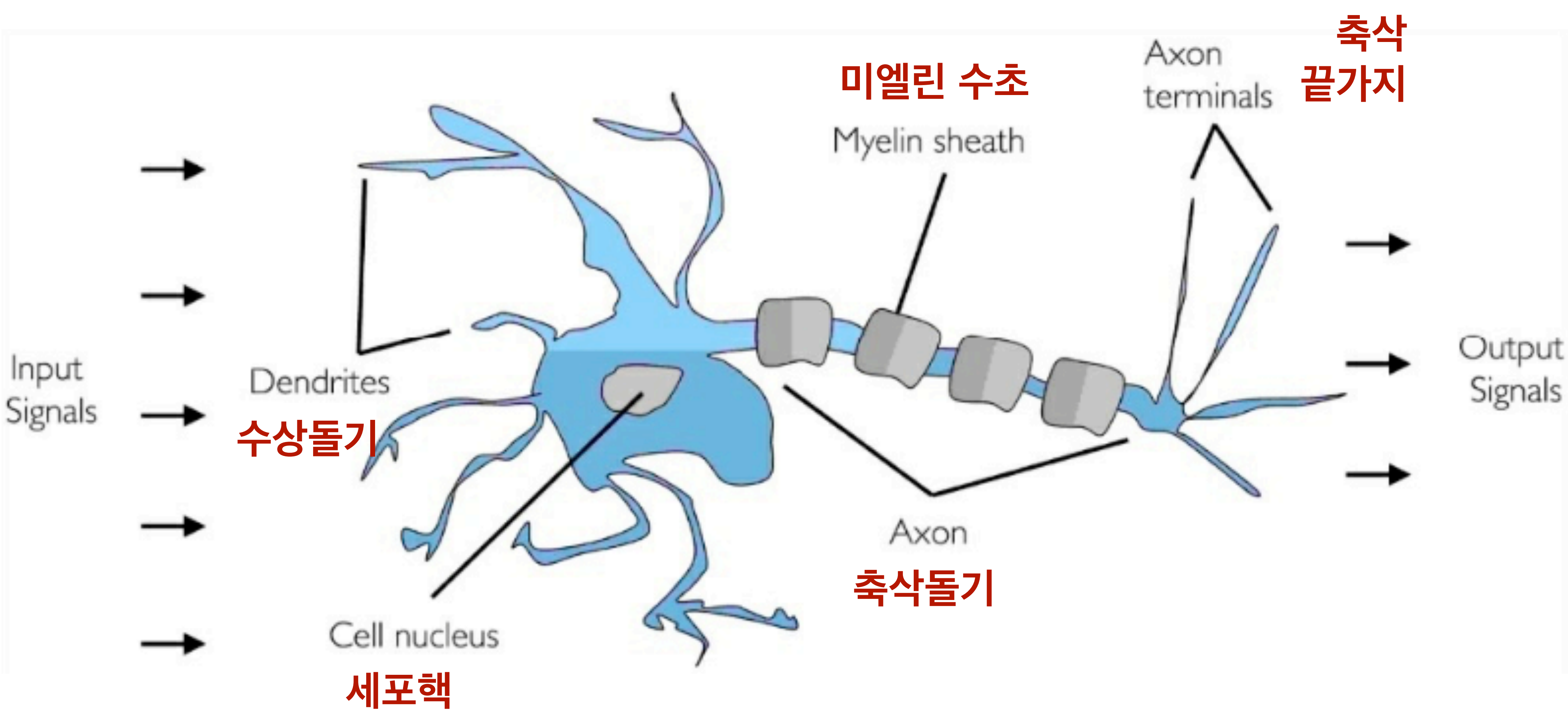
Machine Learning Model Construction



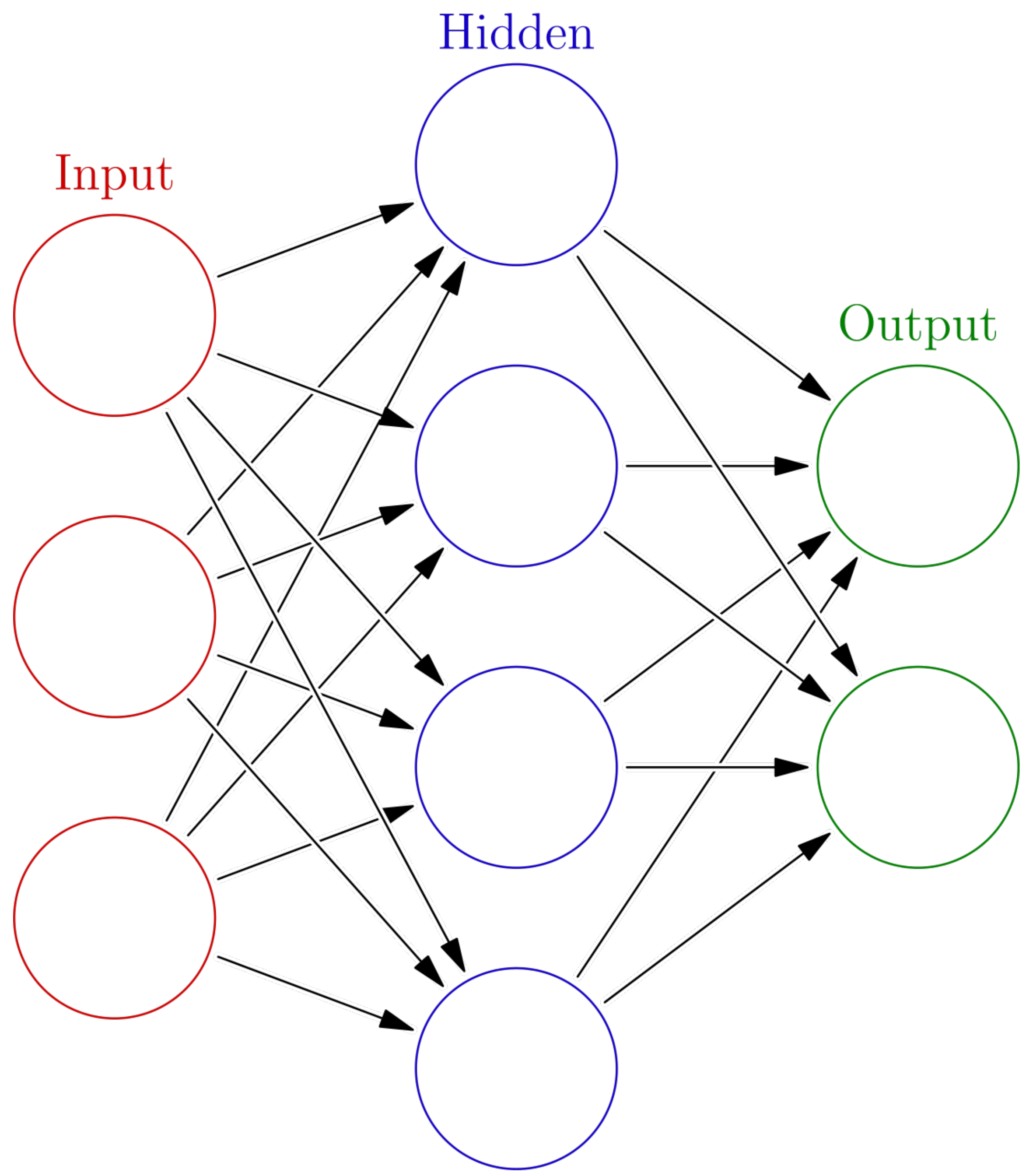
- Model Selection
- Cross-Validation 교차검증
    - Training data set
    - Test data set
- Performance Metrics 성능지표
    - Accuracy
- Hyperparameters
    - Improving model performance

Simple Learning Algorithm: Perceptron

Neurons 신경세포



Neural Networks



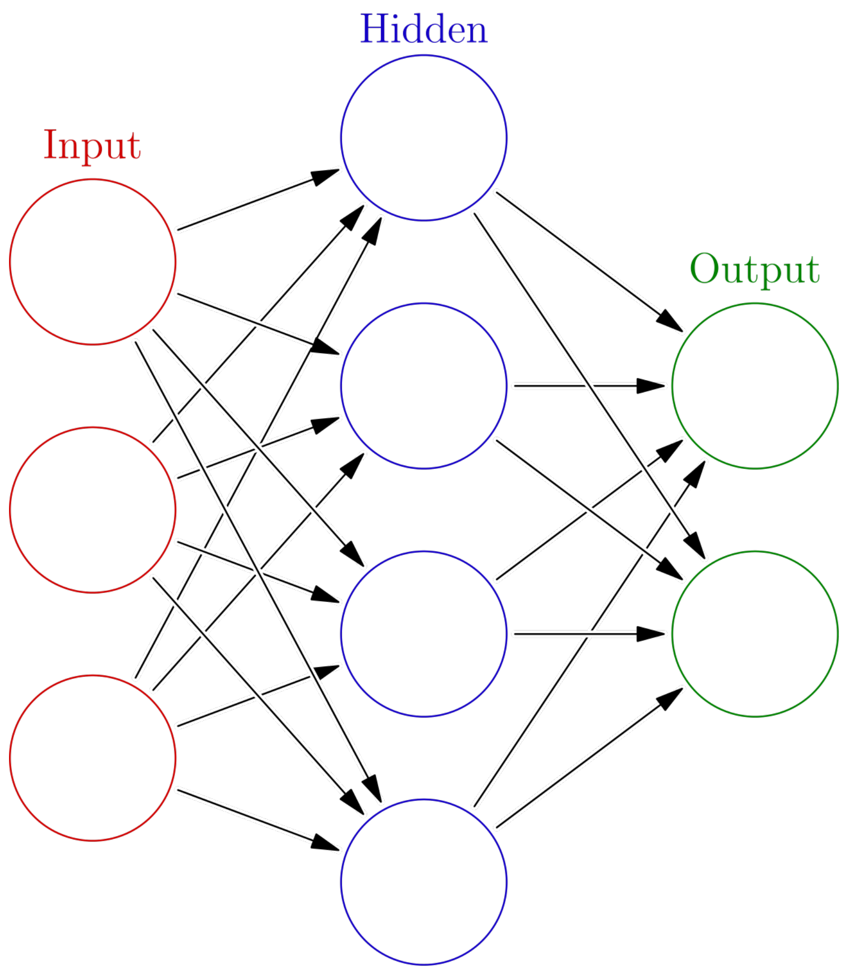
Artificial Neuron: Perceptron

- A node

$$y = wx$$

- A single layer neuron

$$y = w_1 \cdot x_1 + \ldots + w_m \cdot x_m = \sum_{i=1}^{m} w_i \cdot x_i,$$



Perceptron

- A single layer neuron

$$y = w_1 \cdot x_1 + \ldots + w_m \cdot x_m = \sum_{i=1}^{m} w_i \cdot x_i,$$

- In matrix notation

$$w \cdot x = [w_1 w_2 \ldots w_m] \cdot \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_m \end{bmatrix} = w_1 \cdot x_1 + \ldots + w_m \cdot x_m$$

Challenge Problem

- Can we teach neural networks something?
  - Something - Functions, Functionalities, Behaviors, Actions …
- Can we teach neural networks to learn a function?

A Problem

- AND gate

x1	x2	y
0	0	0
0	1	0
1	0	0
1	1	1

$$y = w_1 x_2 + w_2 x_2$$

Problem

- Build a reasoning system, i.e., a perceptron model for the AND gate. In particular, decide w_1, w_2, Θ.

  - AND gate

x1	x2	y
0	0	0
0	1	0
1	0	0
1	1	1

$$y = \begin{cases} 0 & \text{if } w_1 \cdot x_1 + w_2 \cdot x_2 <= \theta \\ 1 & \text{if } w_1 \cdot x_1 + w_2 \cdot x_2 > \theta \end{cases}$$

Problem

- For OR and NAND gates, determine w_i and θ.

OR gate

x1	x2	y
0	0	0
0	1	1
1	0	1
1	1	1

NAND gate

x1	x2	y
0	0	1
0	1	1
1	0	1
1	1	0

$$y = \begin{cases} 0 & \text{if } w_1x_1 + w_2x_2 \leq \theta \\ 1 & \text{if } w_1x_1 + w_2x_2 > \theta \end{cases}$$

Problem

- What about XOR gate, can we determine w_i and θ?

XOR gate

x1	x2	y
0	0	0
0	1	1
1	0	1
1	1	0

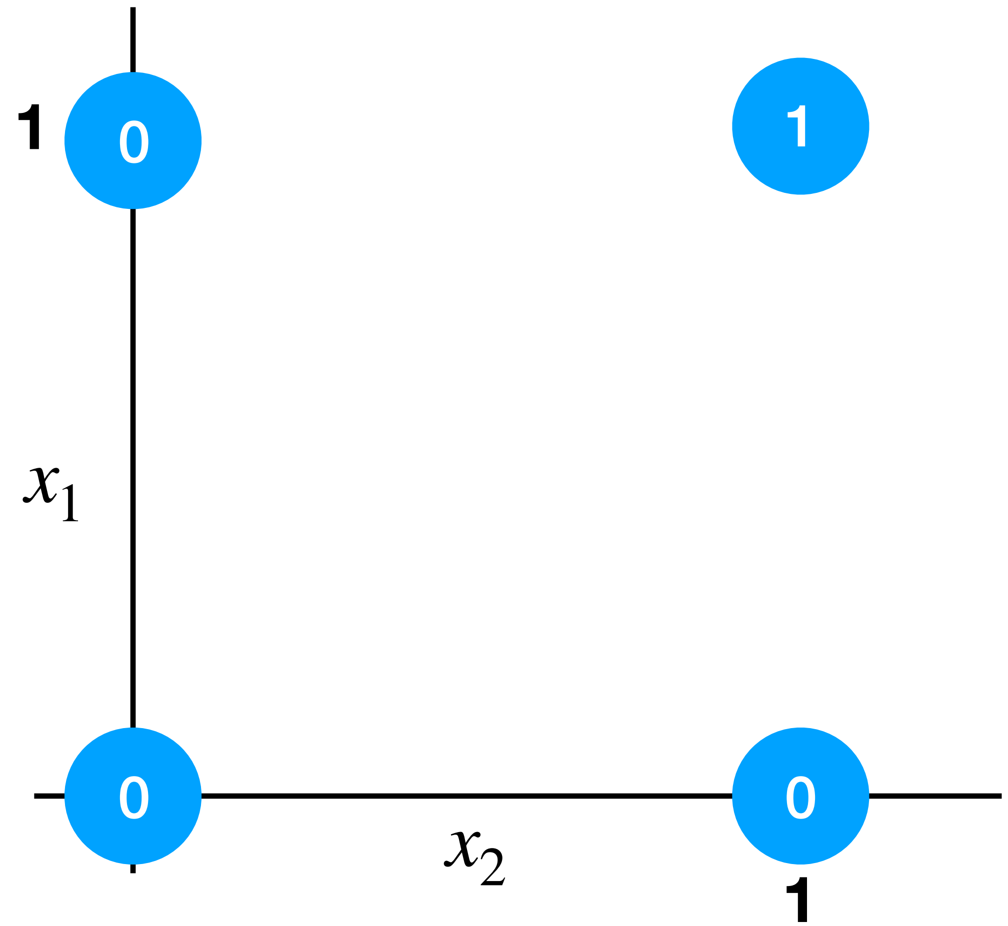
$$y = \begin{cases} 0 & \text{if } w_1x_1 + w_2x_2 \leq \theta \\ 1 & \text{if } w_1x_1 + w_2x_2 > \theta \end{cases}$$

Perceptron

- AND gate

$$y = \begin{cases} 0 & \text{if } w_1x_1 + w_2x_2 \leq \theta \\ 1 & \text{if } w_1x_1 + w_2x_2 > \theta \end{cases}$$

x1	x2	y
0	0	0
0	1	0
1	0	0
1	1	1

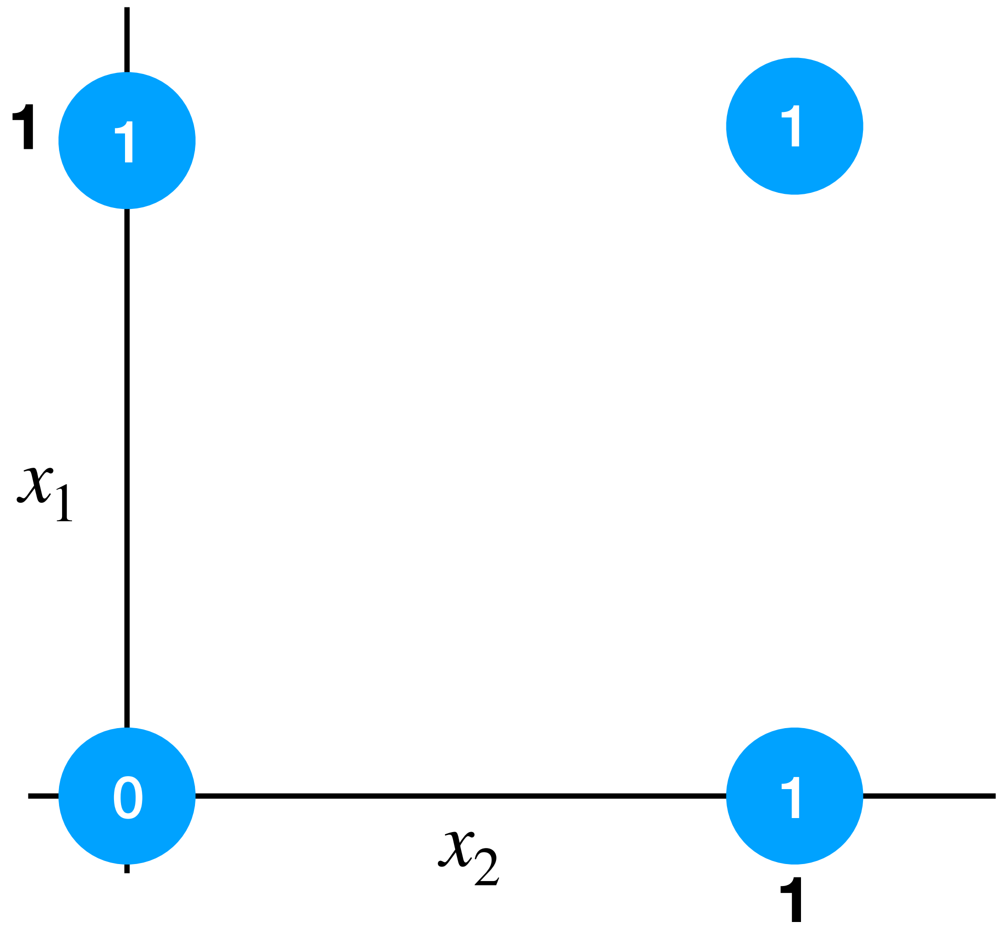


Perceptron

- OR gate

$$y = \begin{cases} 0 & \text{if } w_1x_1 + w_2x_2 \leq \theta \\ 1 & \text{if } w_1x_1 + w_2x_2 > \theta \end{cases}$$

x1	x2	y
0	0	0
0	1	1
1	0	1
1	1	1

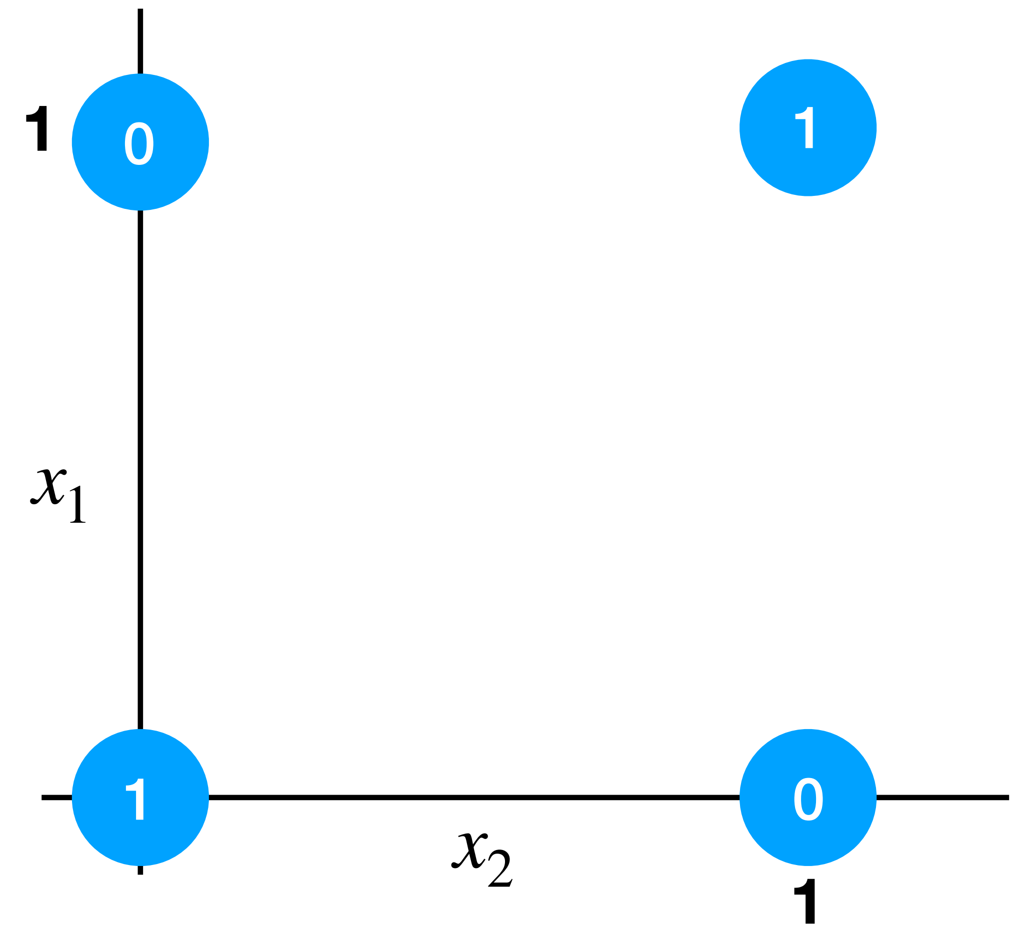


Perceptron

- XOR gate

$$y = \begin{cases} 0 & \text{if } w_1x_1 + w_2x_2 \leq \theta \\ 1 & \text{if } w_1x_1 + w_2x_2 > \theta \end{cases}$$

x1	x2	y
0	0	0
0	1	1
1	0	1
1	1	0



Transform Perceptron

$$y = \begin{cases} 0 & \text{if } w_1 \cdot x_1 + w_2 \cdot x_2 <= \theta \\ 1 & \text{if } w_1 \cdot x_1 + w_2 \cdot x_2 > \theta \end{cases}$$

$$y = \begin{cases} 0 & \text{if } b + w_1 \cdot x_1 + w_2 \cdot x_2 <= 0 \\ 1 & \text{if } b + w_1 \cdot x_1 + w_2 \cdot x_2 > 0 \end{cases}$$

- w_i is called weight 가중치
- b is called bias 편향

The Problem

- How to determine weights and bias for perceptron models such that a model returns right answers?

In next Class

- Dig into more maths for machine learning and Python, Numpy