기계학습 기초 및 응용

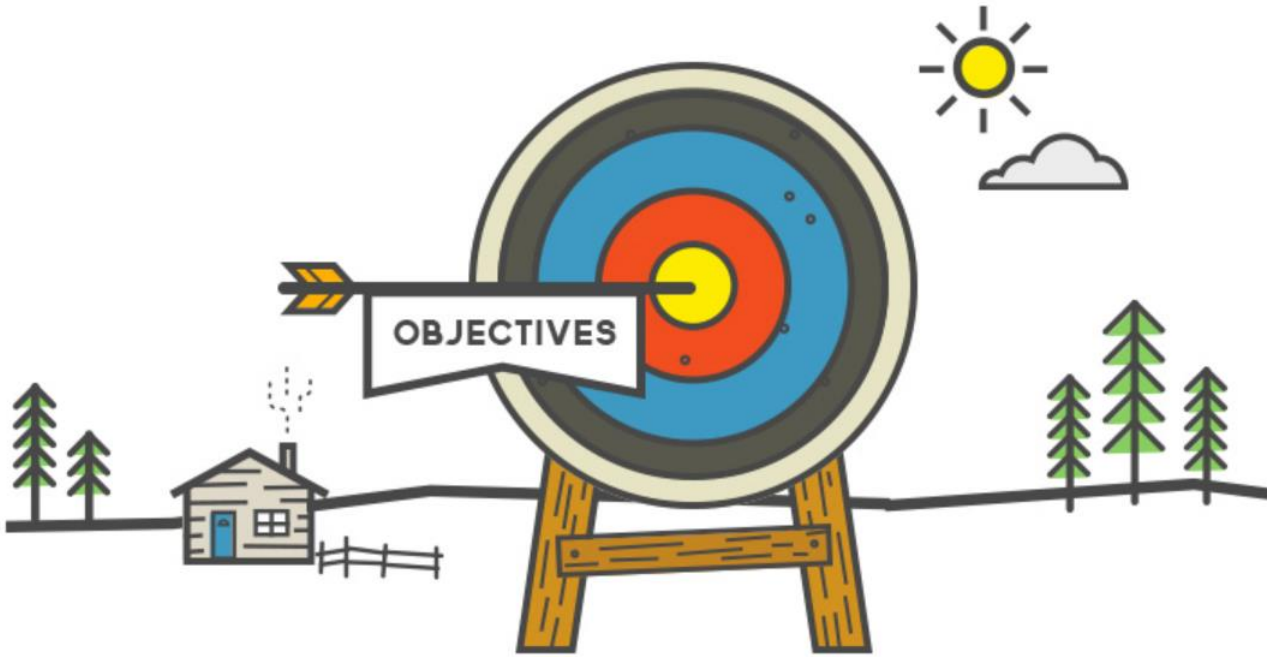
기계학습 기초

소프트웨어융합대학
소프트웨어학부

KMU 국민대학교
KOOKMIN UNIVERSITY

수업 목적

- main objectives:
  - understand fundamentals of machine learning
  - have hands-on experience
  - motivate to learn recent breakthroughs in machine learning



평가

- 3 hands-on experience (45%)
- midterm (30%)
- class contribution (15%)
- class attendance (10%)

수업 계획

Week	Contents	
1	수업 소개	
2	인공지능, 기계학습 개념 I	
3	인공지능, 기계학습 개념 II	
4	인공 신경망 I	
5	인공 신경망 II	
6	PYTHON 및 PYTORCH 실습	
7	심층 학습 I	
8	심층 학습 II	
9	중간 시험	
10	NVIDIA DLI 실습	
11	Cloud Platform 활용 I	
12	Cloud Platform 활용 II	
13	Cloud Platform 활용 III	
14	Kaggle 실습	
15	초청 강연	

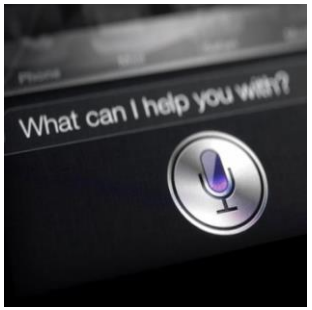
인공지능

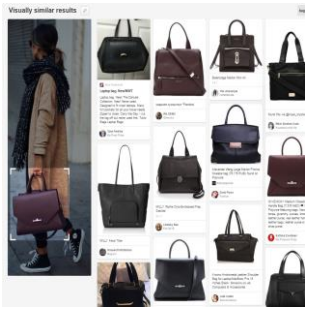
- 인공지능Artificial intelligence (AI) 정의
  - 인간의 지능을 기기에 인공적으로 구현한 것

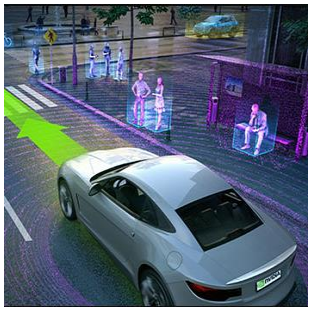
  - 인공지능의 요소
    - 학습learning
    - 추론inference
    - 지식knowledge
    - 이해understanding language

    → 기기가 사람처럼 행동하고 생각하도록 하는 모든 요소들 포함 A machine that thinks or acts like a human

  - 인공지능 사례
    - 음성 인식
    - 추천 시스템
    - 기계 번역
    - 자율 주행







인공지능 접근 방법

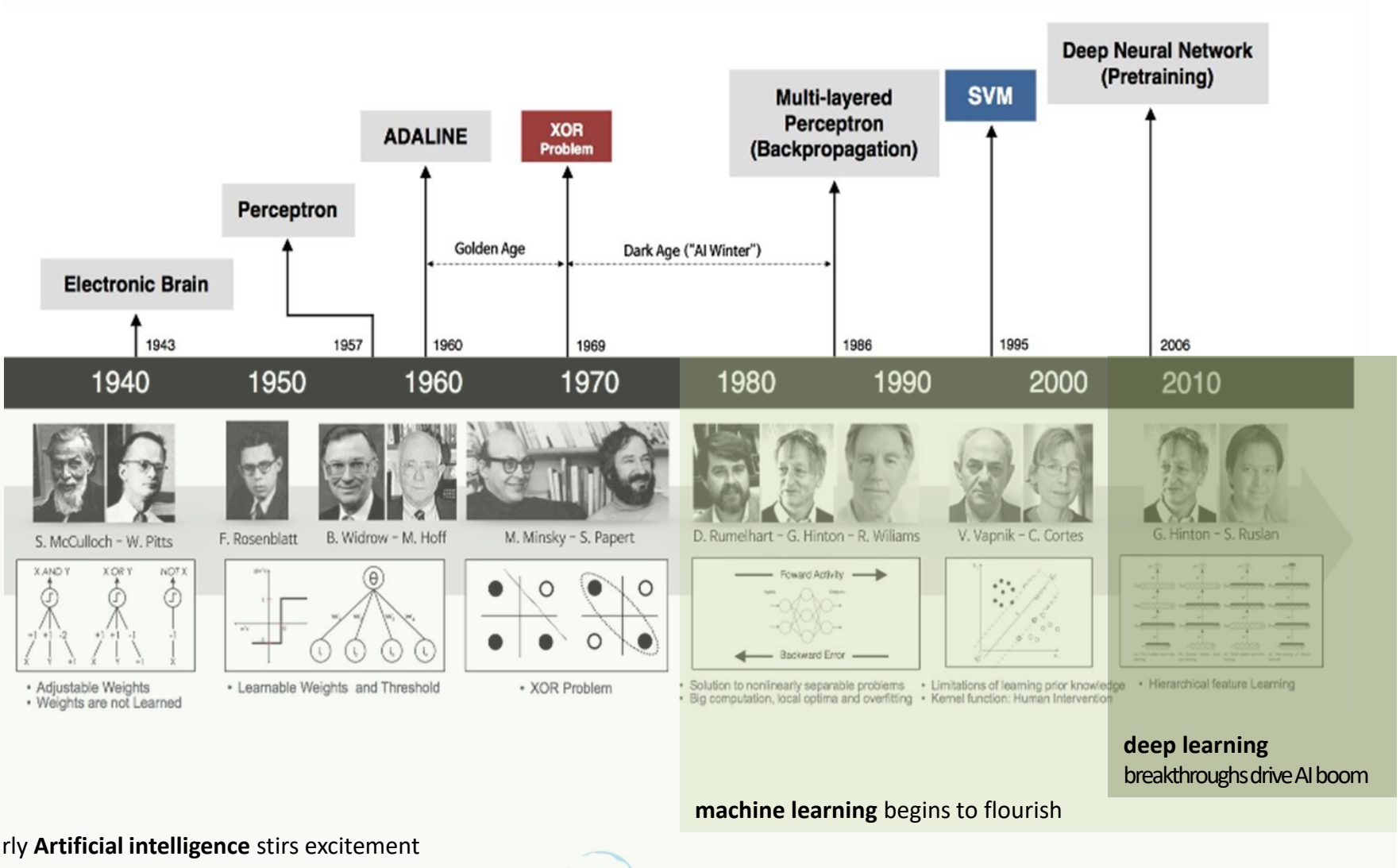
인공지능 주요 접근 방법

- 인공 두뇌Cybernetics
- 기호Symbolic
- 통계Statistical
- 통합Integrative

인공지능 도구

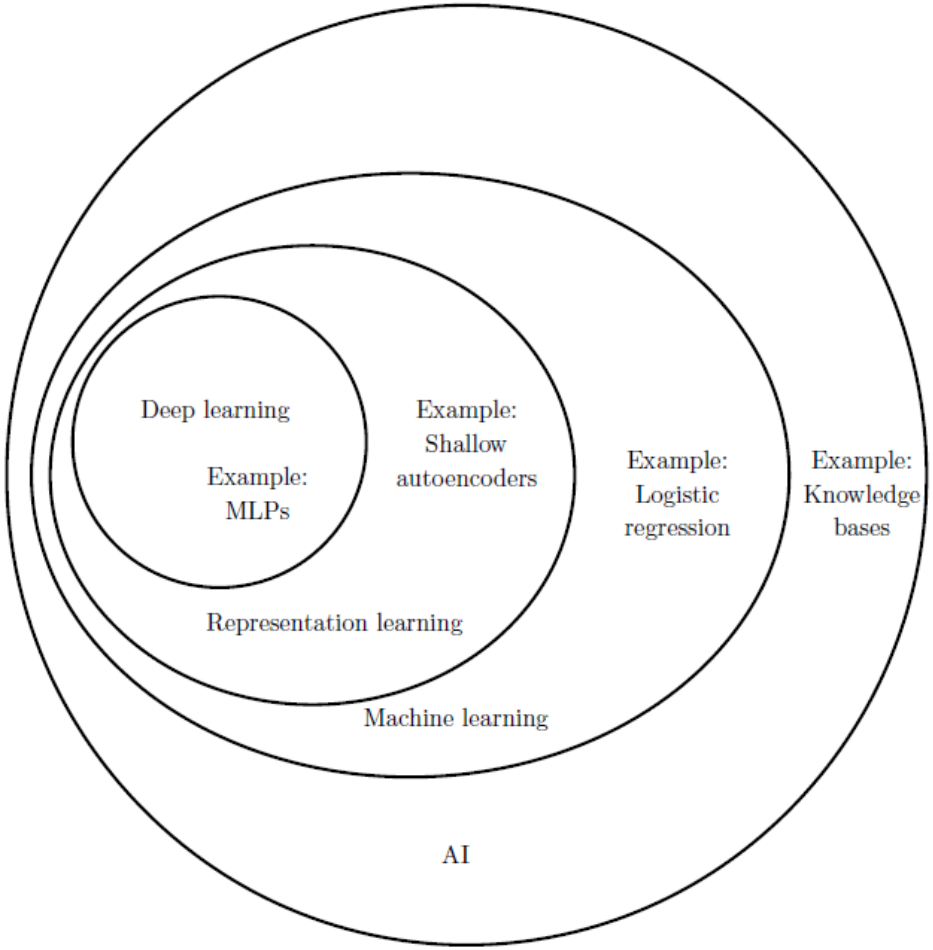
- 탐색 및 최적화 Search and optimization
- 논리 및 선형대수 Logic and linear algebra
- 확률 및 정보 이론 Probabilistic method and information theory
- 통계 Statistical method
- 인공 신경망과 심층 학습 Artificial neural networks and deep learning

인공지능의 변천사

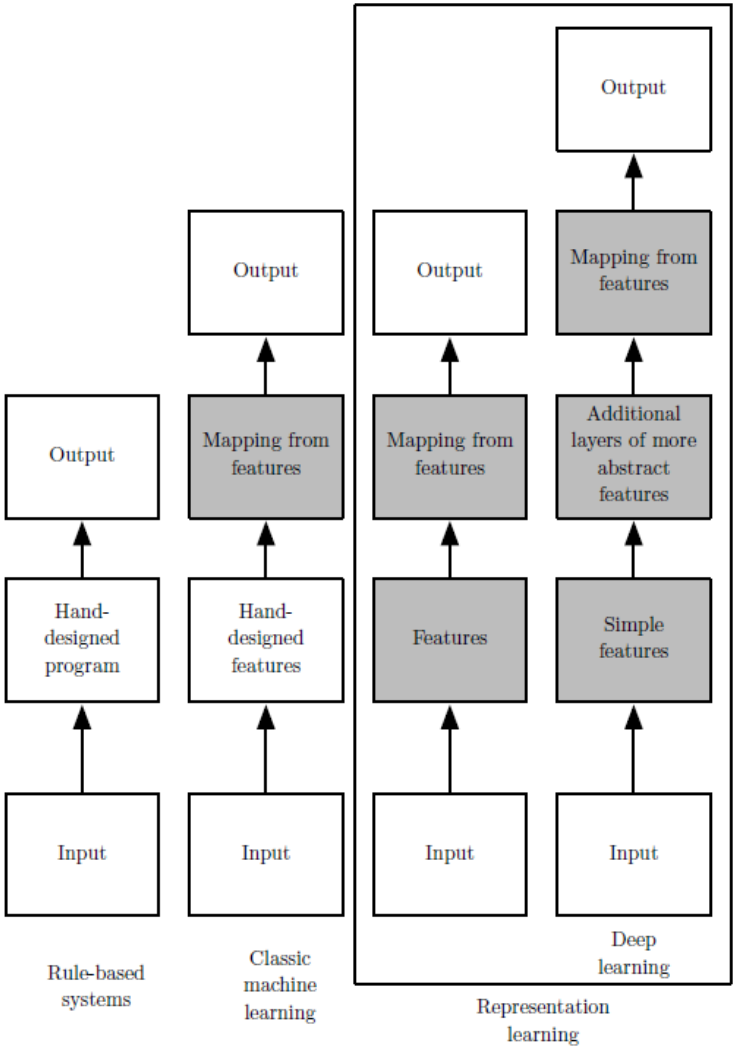

Deep Neural Network (Pretraining)
SVM
Multi-layered Perceptron (Backpropagation)
ADALINE
XOR Problem
Perceptron
Electronic Brain
Golden Age
Dark Age ("AI Winter")
1943
1957
1960
1969
1986
1995
2006
1940
1950
1960
1970
1980
1990
2000
2010
S. McCulloch - W. Pitts
F. Rosenblatt
B. Widrow - M. Hoff
M. Minsky - S. Papert
D. Rumelhart - G. Hinton - R. Wiliams
V. Vapnik - C. Cortes
G. Hinton - S. Ruslan
X AND Y
X OR Y
NOT X
Foward Activity
Backward Error
• Adjustable Weights
• Weights are not Learned
• Learnable Weights and Threshold
• XOR Problem
• Solution to nonlinearly separable problems
• Big computation, local optima and overfitting
• Limitations of learning prior knowledge
• Kernel function: Human Intervention
• Hierarchical feature Learning
deep learning
breakthroughs drive AI boom
machine learning begins to flourish
early Artificial intelligence stirs excitement

인공지능과 기계학습

- 인공지능 분류



회색 표시: 데이터에 의해 학습됨



기계학습

기계학습Machine learning 정의

- 인공지능의 파생 방법
- 기기를 인간처럼 학습시켜 스스로 규칙을 형성
  - 명시적인 프로그래밍 없이 데이터를 학습
  - 학습: 특정 작업task의 성능performance을 점진적으로 개선

- A computer program
  - improve their **performance P** (accuracy, error rate,…)
  - at some **task T** (classification, regression, detection,…)
  - with **experience E** (data)

→ well-defined learning task: <**P**, **T**, **E**>

"using a set of observations to uncover an underlaying process"

기계학습 예

Example: predicting how a viewer will rate a movie

- good recommender systems are important
  - 20% sales are from recommendation (Amazon.com)
  - 10% improvement = 1 million dollar prize (Netflix)

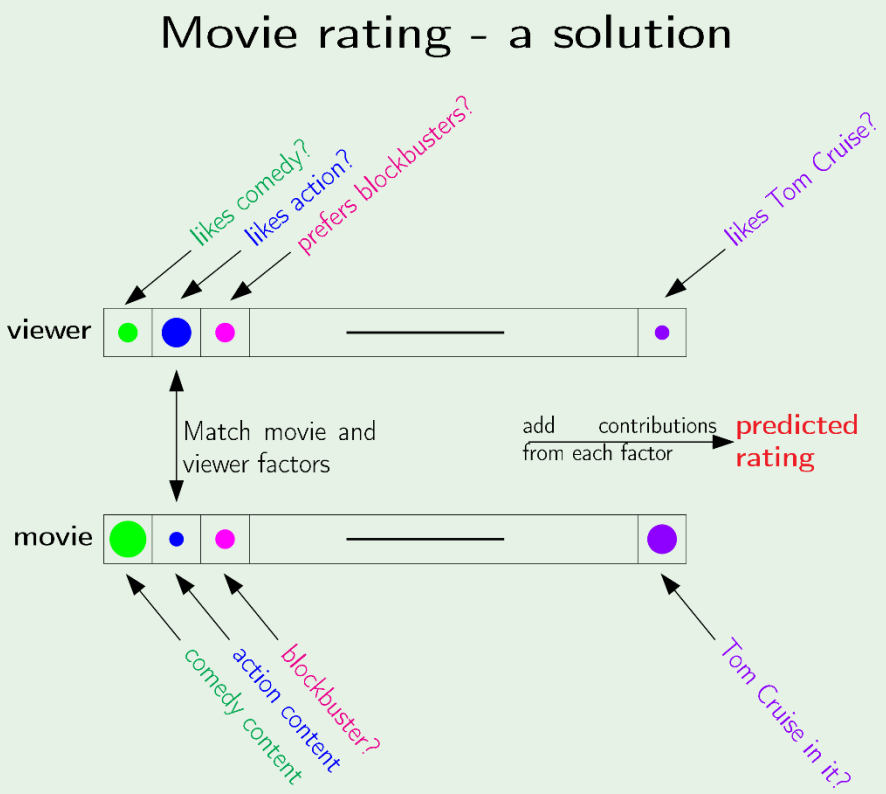
추천 시스템

Simple viewpoint

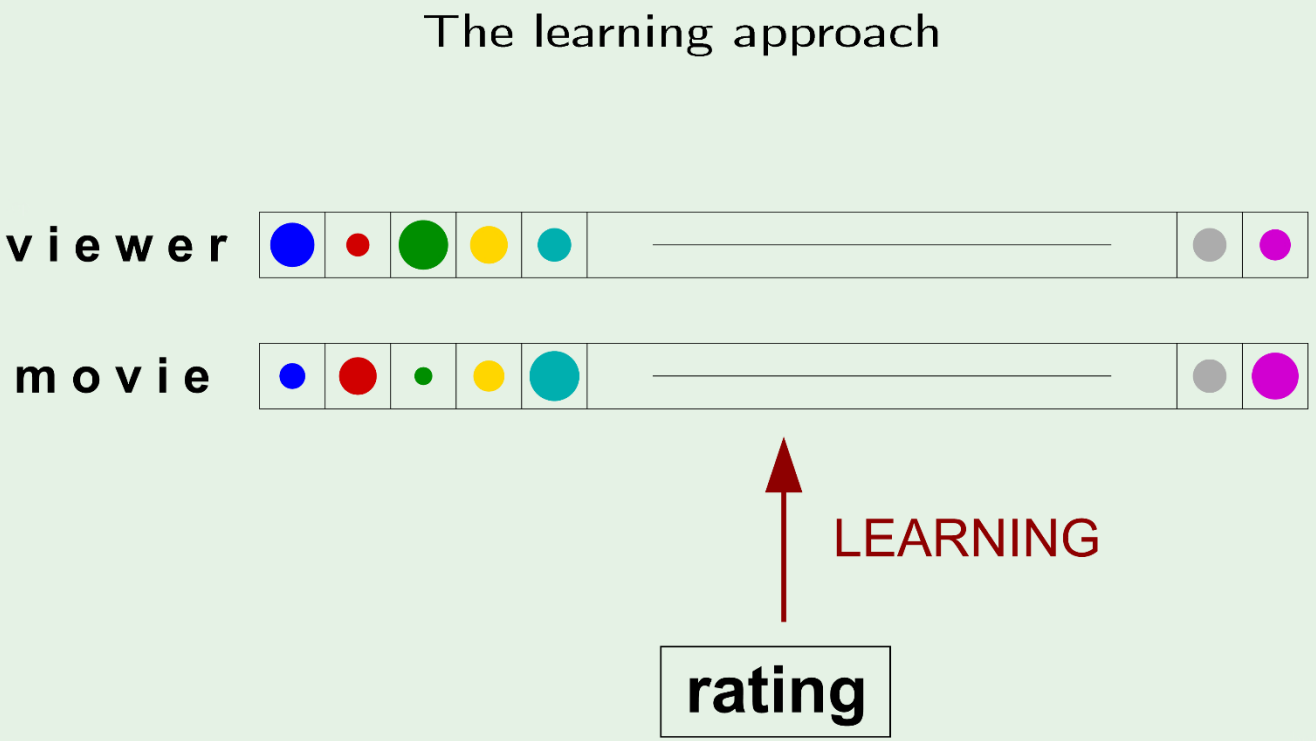
Movie					
The Lion King	👍			👎	
Lethal Weapon	👎	👍	👍	👍	
The Sound of Music		👍			👎
Amadeus	👍	👎		👍	
When Harry Met Sally	👍	👍		👎	👍

추천 시스템

Another viewpoint



추천 시스템



기계학습 예

Which of the following problems are best suited for the learning approach?

(i) Classifying numbers into primes and non-primes.

(ii) Detecting potential fraud in credit card charges.

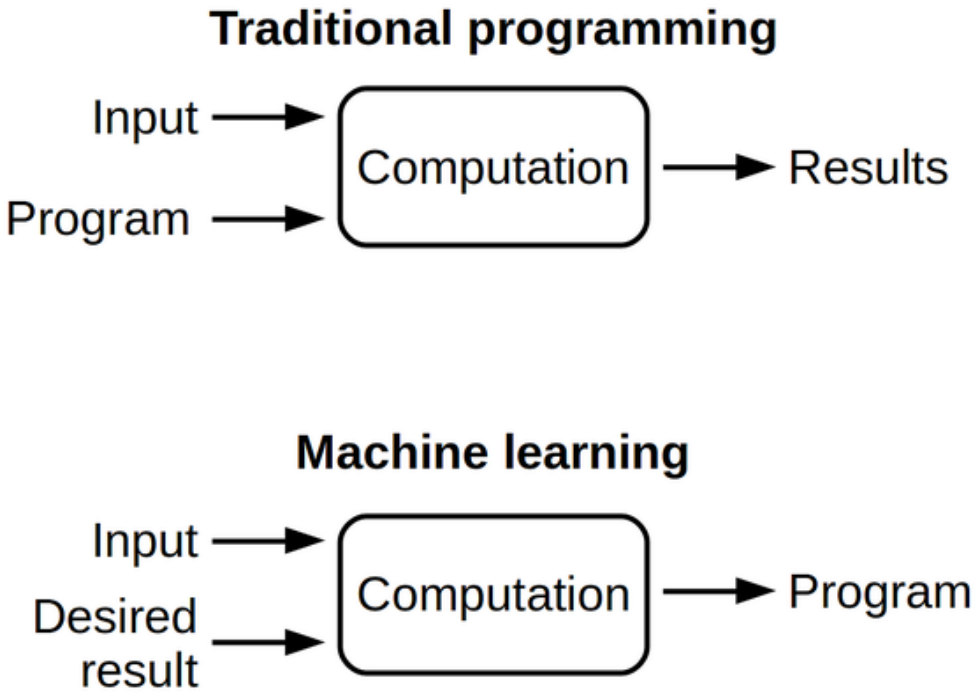
(iii) Determining the time it would take a falling object to hit the ground.

(iv) Determining the optimal cycle for traffic lights in a busy intersection.

Answer: (ii)

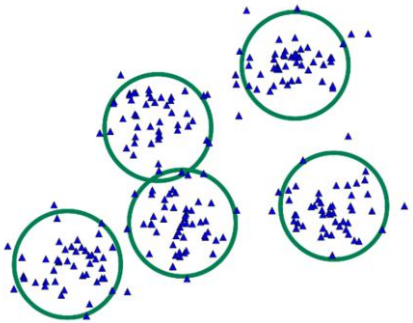
명시적인 프로그래밍의 한계

- 명시적인 프로그래밍과 기계학습 비교



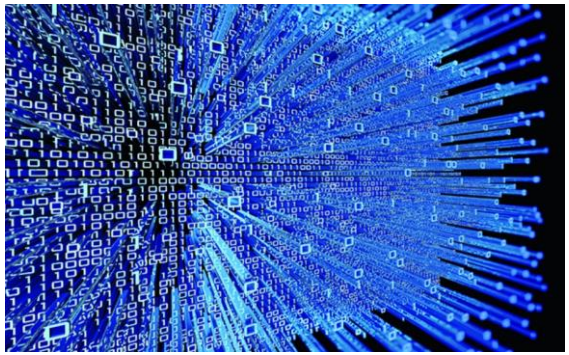
기계학습 필수 요소

- 규칙 존재



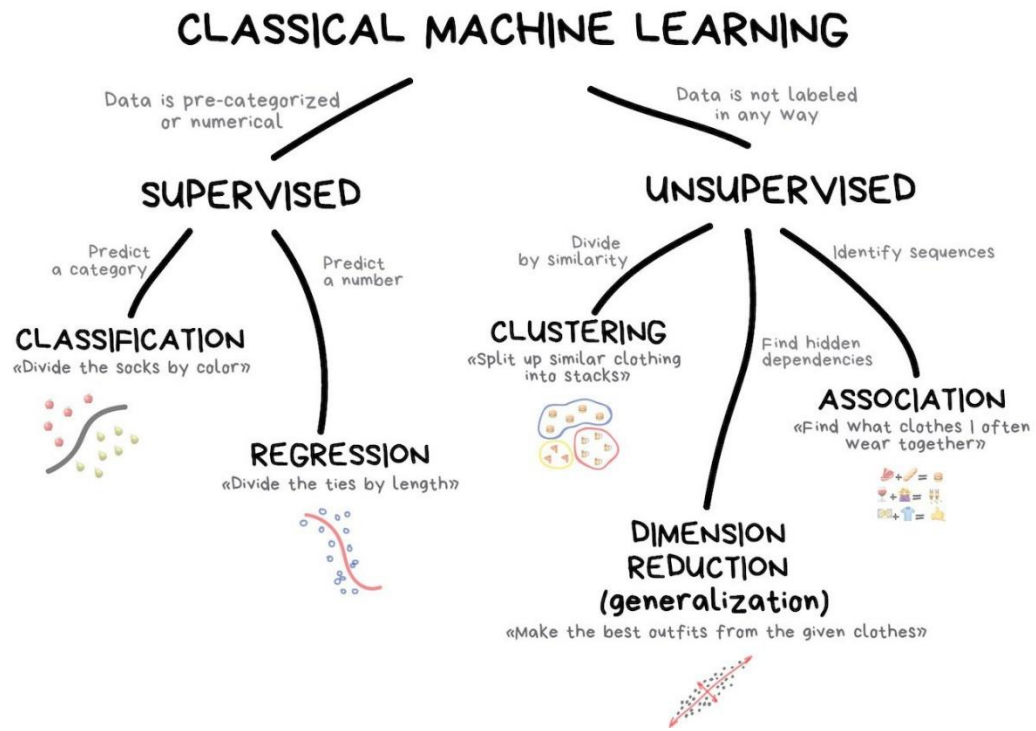
- 수학적으로 명시적인 규칙 확인 불가

- 데이터 존재



기계학습 문제

- 기계학습 문제 분류
  - 교사 학습 Supervised learning: 정답 answer (label) 있는 학습
  - 비교사 학습 Unsupervised learning: 정답 answer 없는 학습

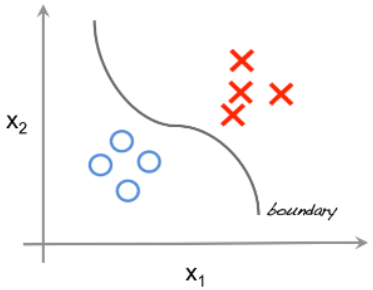


  - 강화 학습 Reinforcement learning: 환경과 상호작용 interaction 있는 학습
  - 기타: 준교사 학습 Semi-supervised learning, 지속 학습 Continual learning

기계학습 문제

- 기계학습 문제 분류 비교

교사 학습

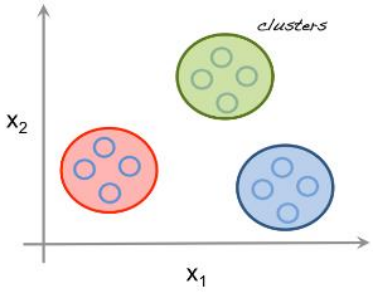


given (x, y)
x is data, y is its label

Goal:
learn a function
to map x → y

Examples:
Classification
Regression
Object detection
Segmentation
Image captioning

비교사 학습

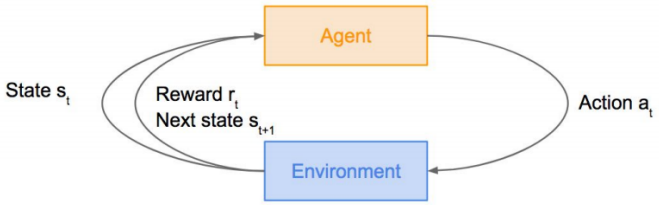


given (x)
just data, no label

Goal:
learn some underlying hidden
structure of the data

Examples:
Clustering
Dimensionality reduction
Feature learning
Density estimation

강화 학습

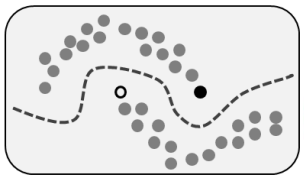


given
Problems involving an agent
interacting with an environment which
provides numeric reward signals

Goal:
Learn how to take actions in order to
maximize reward

Examples:
Robotics
Self-driving

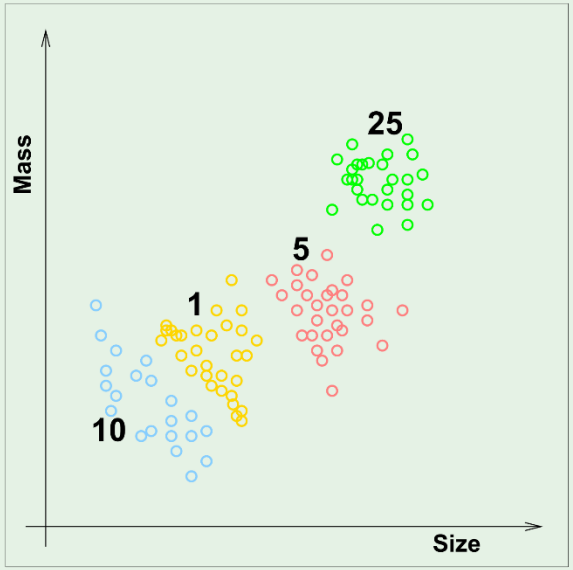
준교사 학습

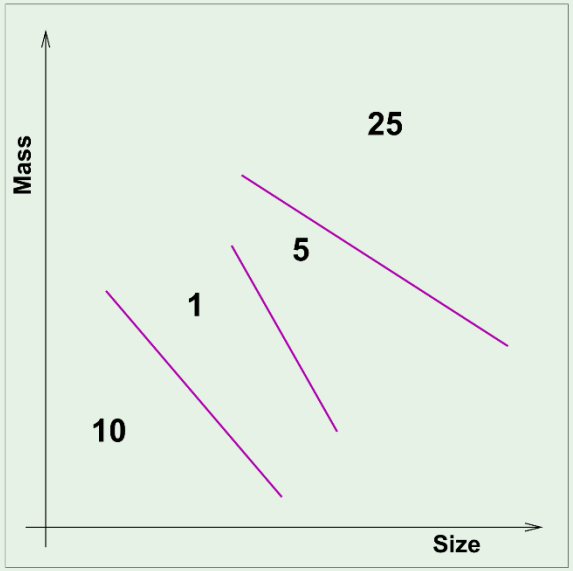


교사 학습

Supervised learning

Example from vending machines – **coin recognition**





교사 학습

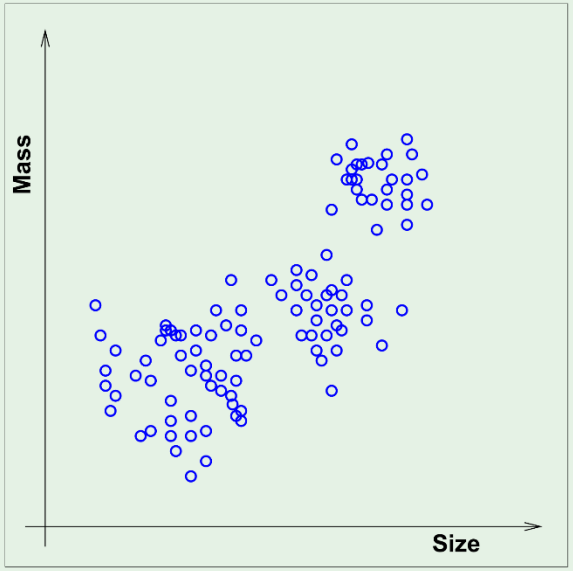
Supervised learning

- the most studied and most utilized type of learning
  - our main focus for a while
- supervised learning setting
  - training data contains explicit examples of what the correct outputs should be for given inputs
- learning is 'supervised' in that
  - some 'supervisor' has taken the trouble to look at each input and determine the correct output
  - the correct 'label' is available for each training sample
- most well-known approaches: classification & regression

비교사 학습

Unsupervised learning

Instead of (input,correct output), we get (input, ?)



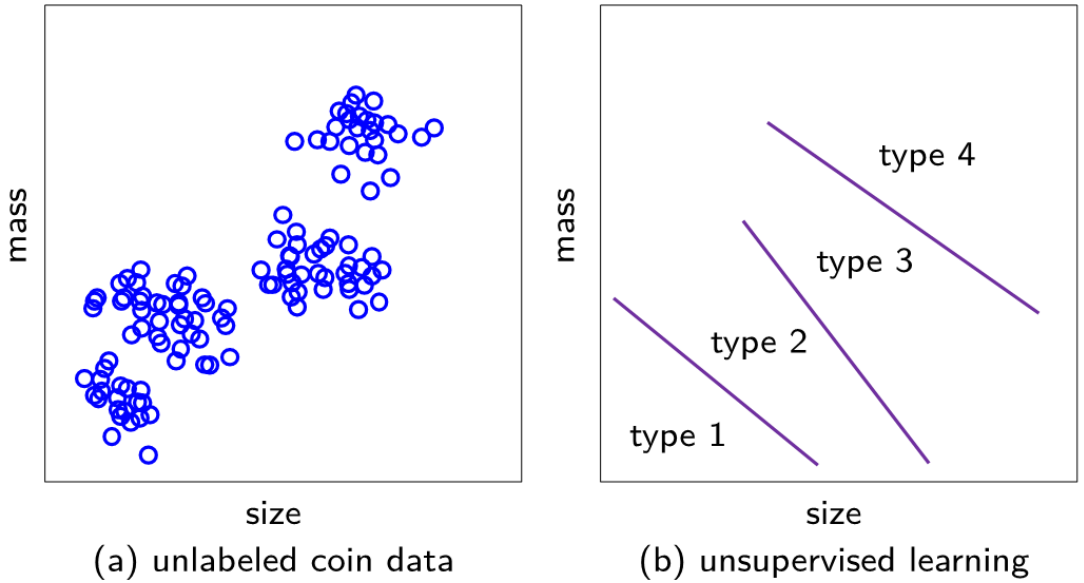
비교사 학습

Unsupervised learning

- the training data do not contain any output information at all
  - instead of (input, correct output), we get (input, ?)
  - that is, we are just given input examples $\mathbf{x}_1, \ldots, \mathbf{x}_N$
- how could we possibly learn anything from mere inputs?
- approaches to unsupervised learning
  - clustering (*e.g.*, k-means, mixture models, hierarchical)
  - hidden Markov models (HMMs)
  - feature extraction (*e.g.*, PCA, ICA, SVD)
- variant: semi-supervised learning

비교사 학습

Example: coin clustering problem



(a) unlabeled coin data (b) unsupervised learning

Figure : the decision regions in unsupervised learning may be identical to those in supervised learning, but without the labels

비교사 학습

Unsupervised learning can be viewed as

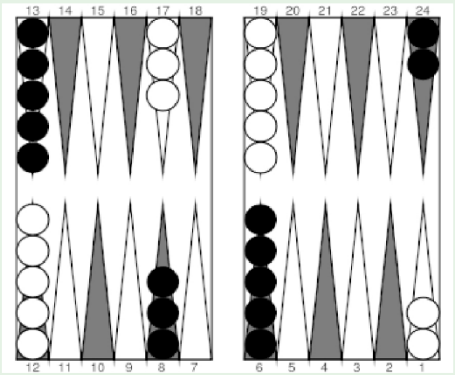
- spontaneously finding the inherent structure in input data
  - ex) categorize a set of books into topics
- a precursor to supervised learning
  - ex) learning Spanish without knowing the meaning first and then taking Spanish lessons will be easier to follow
- a way to create a dimensionality reduction of the data
  - ex) automated feature extraction

강화 학습

Reinforcement learning

Instead of (input,correct output),

we get (input,*some* output,grade for this output)



The world champion was a neural network!

강화 학습

Reinforcement learning

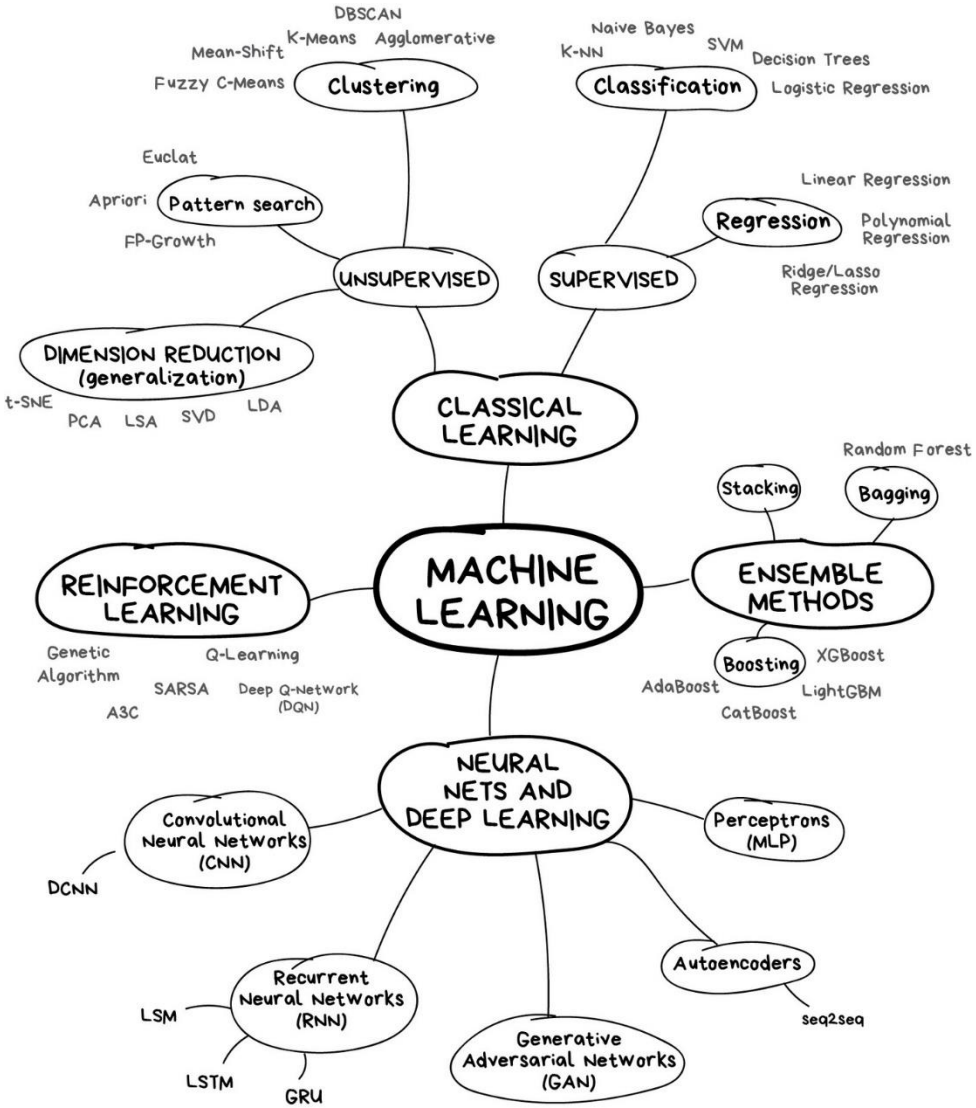
- when training data contain no correct output for each input
  - no longer in a supervised learning setting
- ex) a toddler learning not to touch a hot cup of tea
  - training examples do not say what to do
  - nevertheless, she uses the examples to reinforce better actions
  - eventually she learns what she should do in similar situations

강화 학습

- this characterizes *reinforcement* learning
  - the training example does not contain the target output
  - but instead contains some possible output
    together with a measure of how good that output is

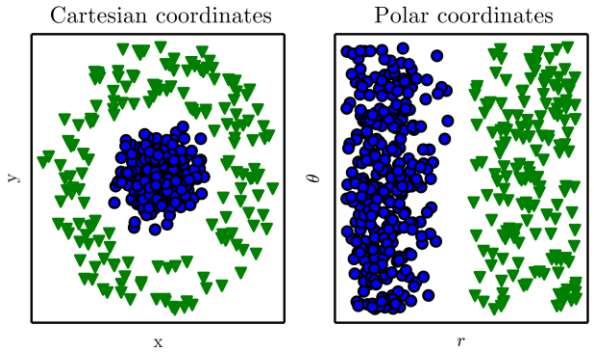
- compare how a training example looks:
  - supervised learning:
    (input, correct output)
  - reinforcement learning:
    (input, some output, grade for this output)

기계학습 분류 세분화


MACHINE LEARNING
CLASSICAL LEARNING
UNSUPERVISED
SUPERVISED
Clustering
DBSCAN
K-Means
Agglomerative
Mean-Shift
Fuzzy C-Means
Classification
Naive Bayes
SVM
K-NN
Decision Trees
Logistic Regression
Pattern search
Euclat
Apriori
FP-Growth
Regression
Linear Regression
Polynomial Regression
Ridge/Lasso Regression
DIMENSION REDUCTION (generalization)
t-SNE
PCA
LSA
SVD
LDA
REINFORCEMENT LEARNING
Genetic Algorithm
Q-Learning
SARSA
Deep Q-Network (DQN)
A3C
ENSEMBLE METHODS
Stacking
Bagging
Random Forest
Boosting
XGBoost
AdaBoost
LightGBM
CatBoost
NEURAL NETS AND DEEP LEARNING
Convolutional Neural Networks (CNN)
DCNN
Perceptrons (MLP)
Recurrent Neural Networks (RNN)
LSM
LSTM
GRU
Autoencoders
seq2seq
Generative Adversarial Networks (GAN)

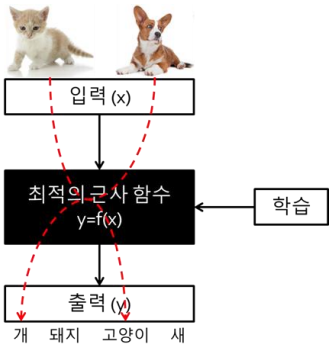
표현학습과 심층학습

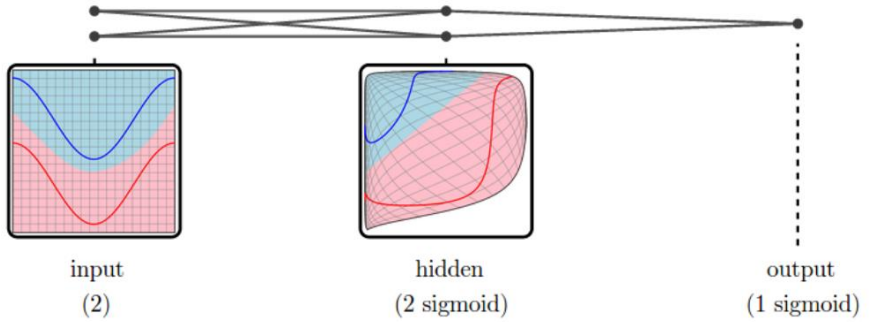
- 표현학습 Representation learning
  - 기계학습의 파생 방법
  - 표현 문제 Representation matter
    - 표현의 차이 비교



  - 심층학습 deep learning: 표현학습 representation learning의 주요 방법

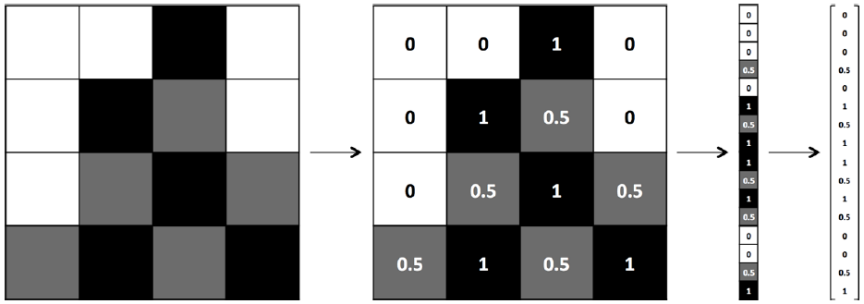
    : 표현에서 출력으로의 사상mapping뿐만 아니라 표현 자체를 학습하여 보다 좋은 성능을 가짐
    - 데이터에서 주어진 작업에 필요한 표현representation을 자동 추출
    - 데이터 중심 특징 data-driven feature의 계층적 학습

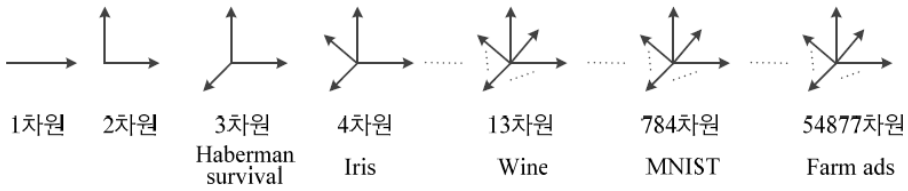


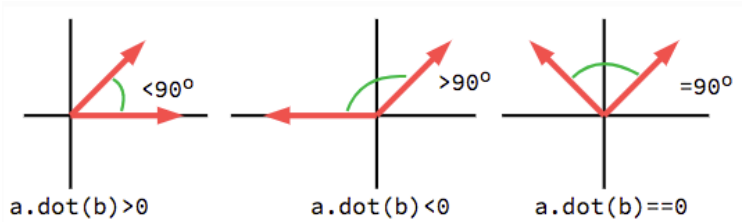


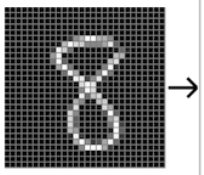
선형대수

- 데이터와 특징공간

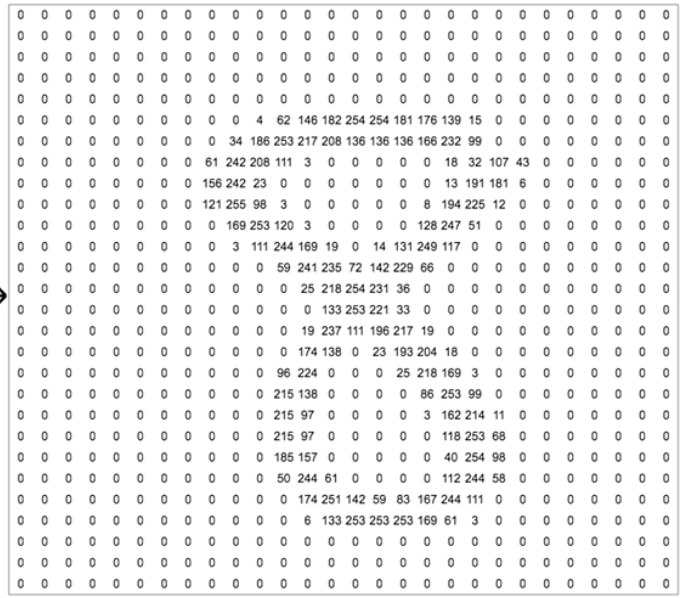


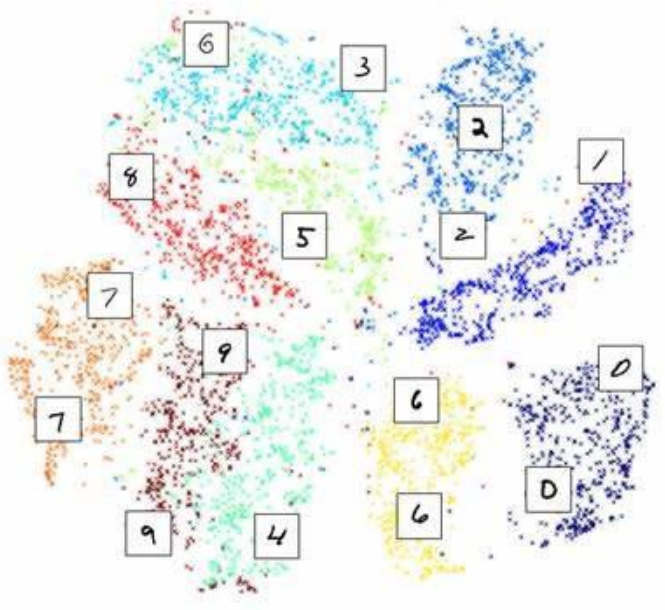






28 x 28
784 pixels





KMU 국민대학교
KOOKMIN UNIVERSITY

표현학습과 심층학습

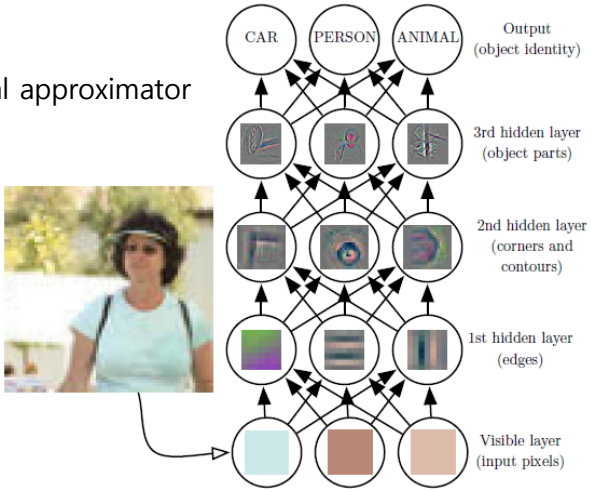
심층학습

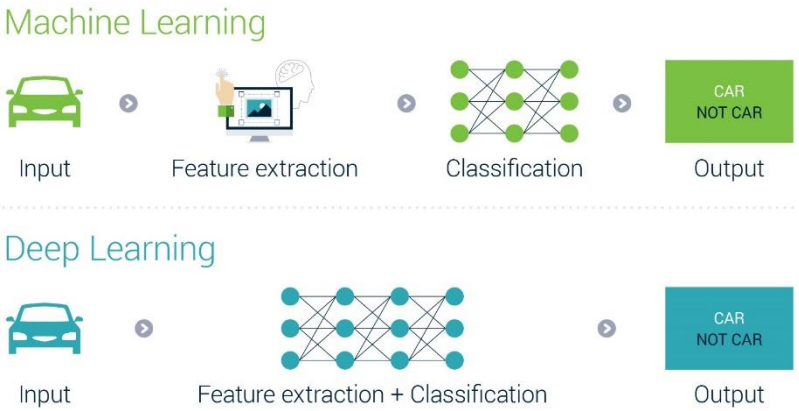
- 선형과 비선형 연산을 갖춘 깊은 인공신경망 deep artificial neural network
- 추상화 수준이 높아진 표현의 계층을 가짐
- 깊은 인공신경망 == 범용 근사 함수 universal approximator

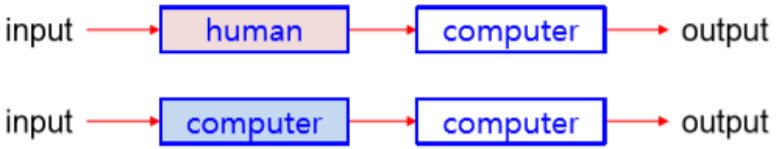
Image
pixel → edge → texton → motif → part → object

Text
character → word → word group → clause → sentence → story

Speech
sample → spectral band → sound → ... → phone → phoneme → word →

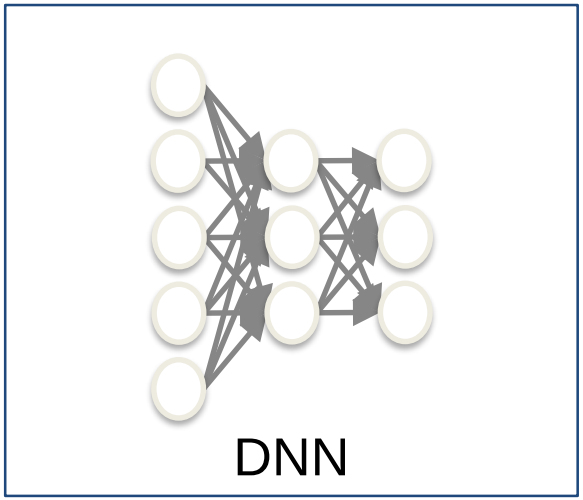




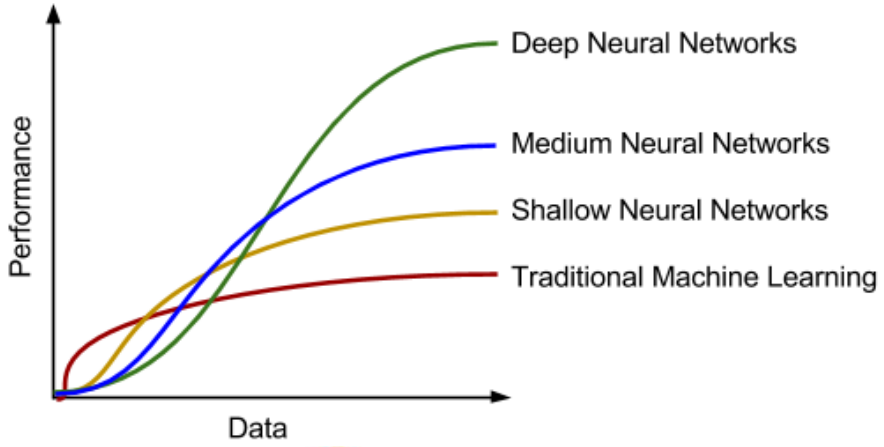


심층학습

- 심층학습의 성공 이유



- |data|와 성능 비교



기계학습 기초

- 기계학습 문제화
  - 사례: 신용 승인 credit approval
    - given: 신청자 정보

feature	value
age	23 years
gender	female
annual salary	$30,000
years in residence	1 year
years in job	1 year
current debt	$15,000

    - task: 승인? 혹은 거절?

기계학습 기초

- 표기 정리

component	symbol	credit approval metaphor
input	$\mathbf{x}$	customer application
output	y	approve or deny
target function	$f : \mathcal{X} \to \mathcal{Y}$	ideal credit approval formula
data	$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$	historical records
hypothesis	$g : \mathcal{X} \to \mathcal{Y}$	formula to be used

- f: unknown target function
- $\mathcal{X}$: input space (set of all possible inputs $\mathbf{x}$)
- $\mathcal{Y}$: output space (set of all possible outputs)
- N: the number of input-output examples (*i.e.* training examples)
- $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$: data set where $y_n = f(\mathbf{x}_n)$

기계학습 기초

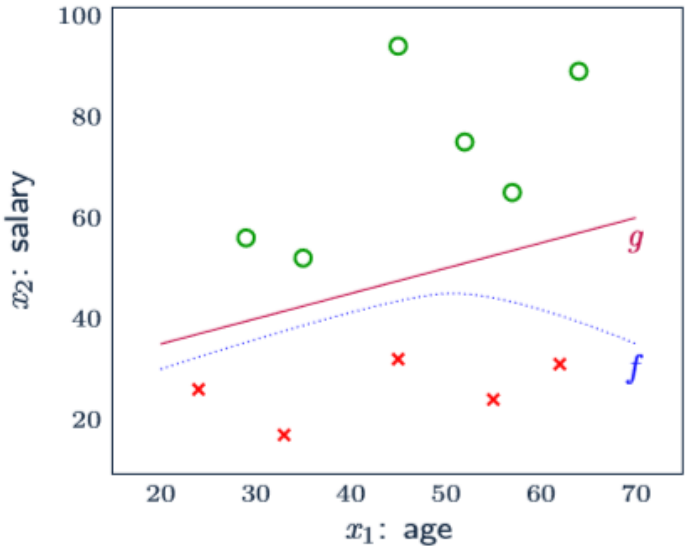
- 문제 해결

$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ where x_1: age and x_2: annual salary in USD

$N = 11$, $d = 2$, $\mathcal{X} = \mathbb{R}^2$, and $\mathcal{Y} = \{\text{approve}, \text{deny}\}$

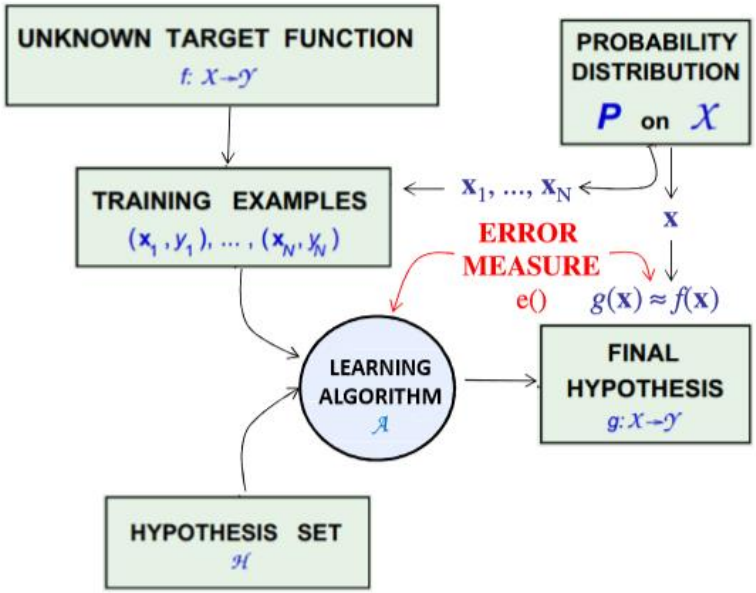
data set $\mathcal{D}$:

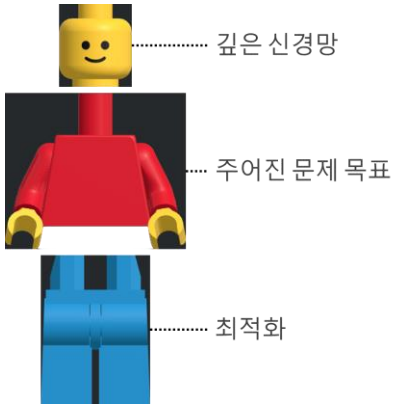
n	x_1	x_2	y
1	29	56k	approve
2	64	89k	approve
3	33	17k	deny
4	45	94k	approve
5	24	26k	deny
6	55	24k	deny
7	35	52k	approve
8	57	65k	approve
9	45	32k	deny
10	52	75k	approve
11	62	31k	deny



기계학습 기초

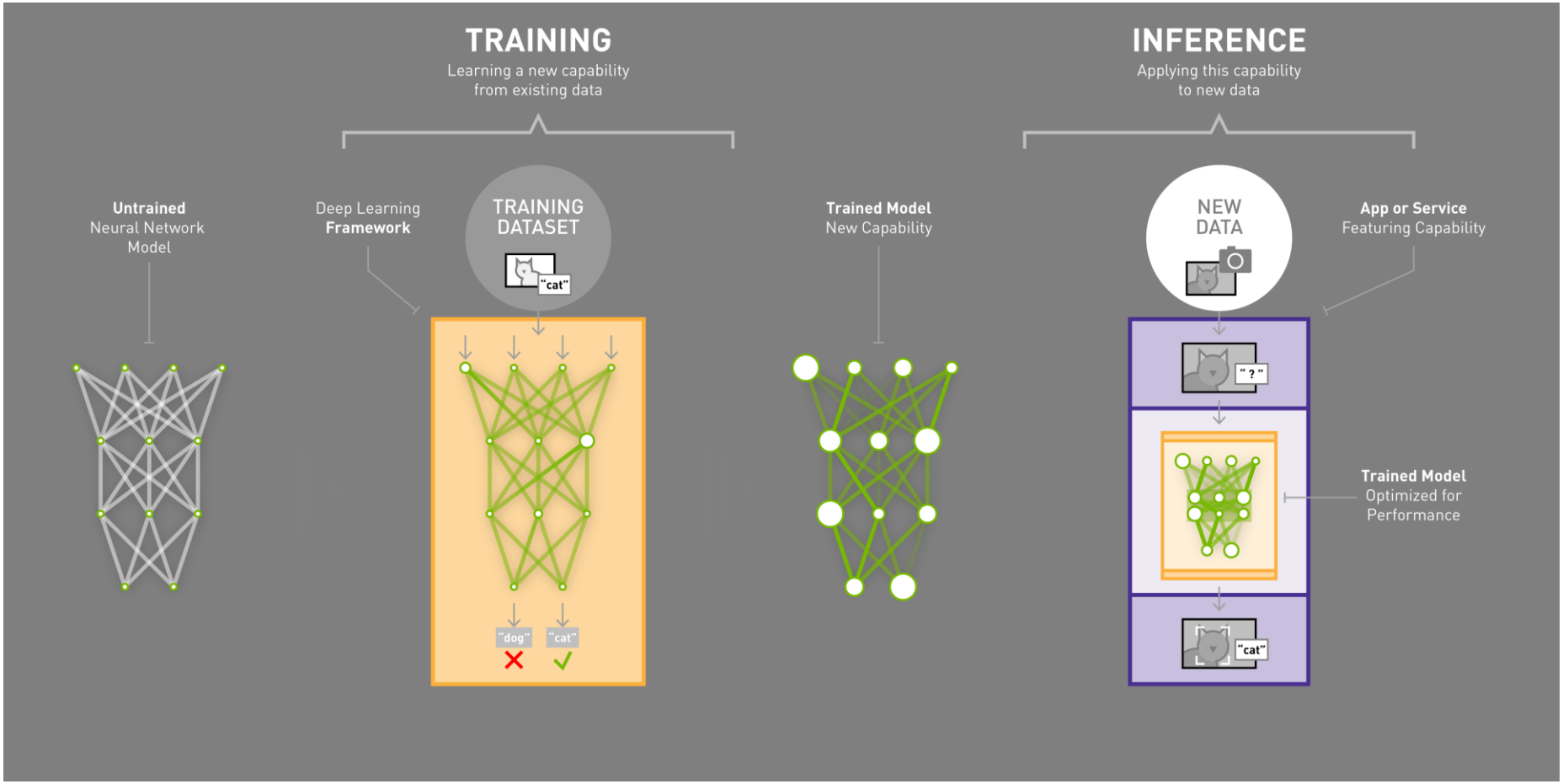
- 기계학습 개요





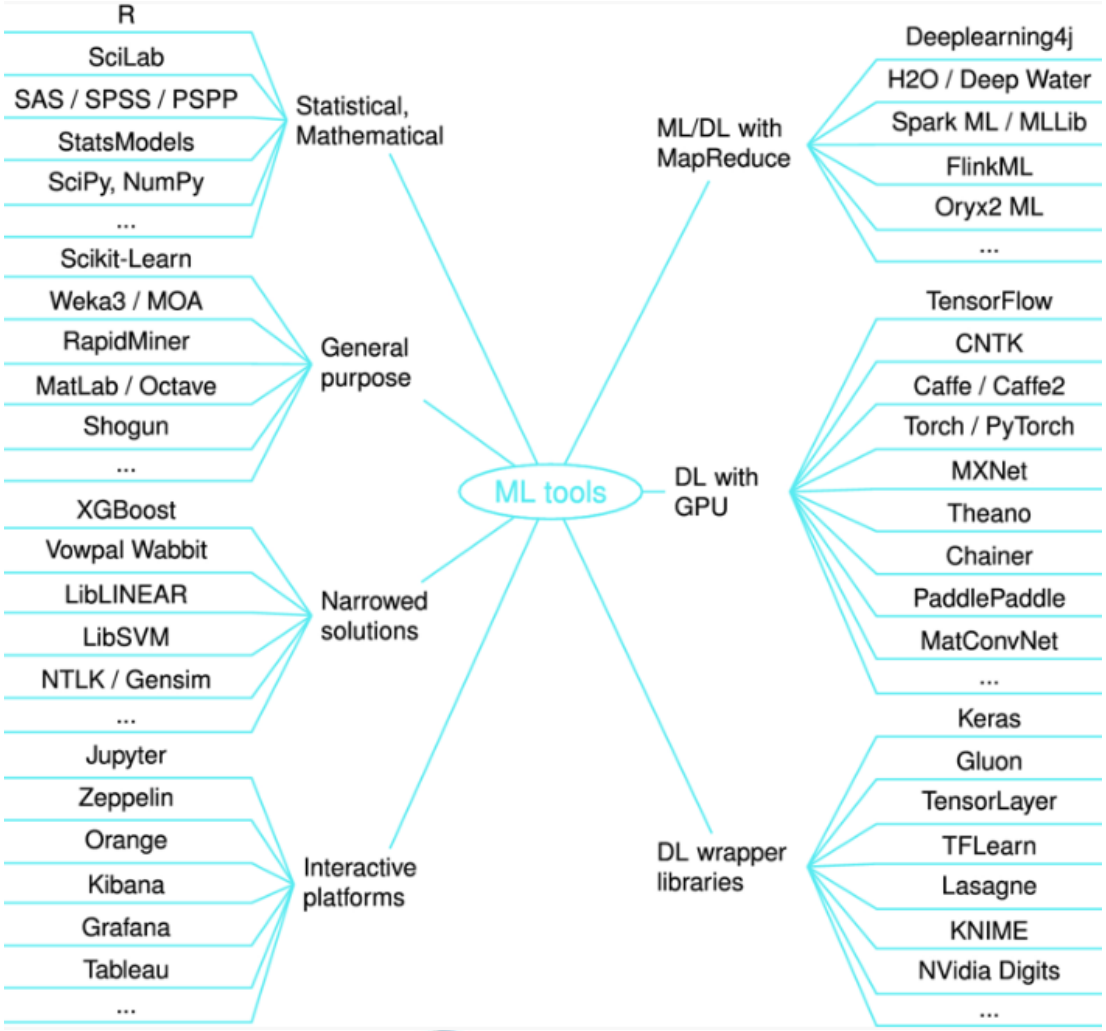
기계학습 기초

- 기계학습 절차: 훈련training 과 추론inference



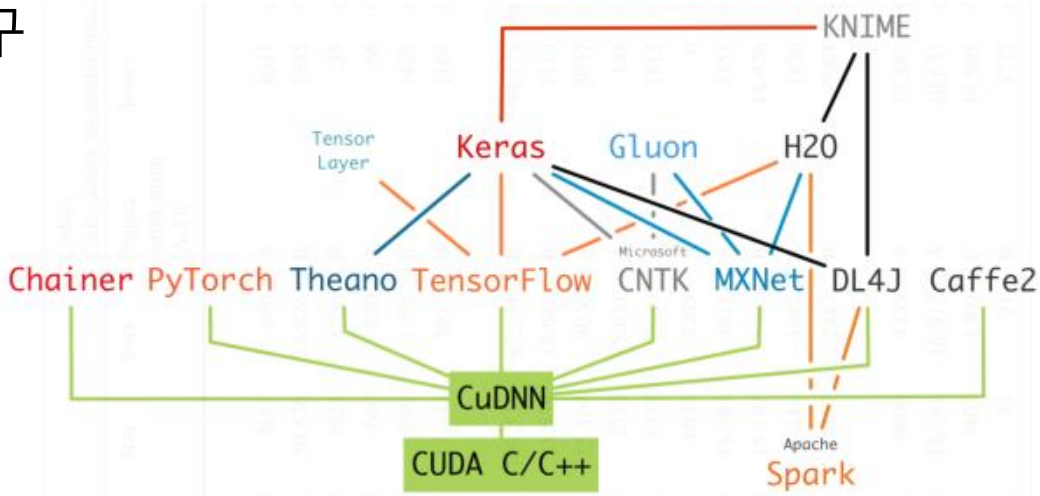
기계학습 기초

- 기계학습 도구



기계학습 기초

- 심층학습 도구



Tool	Licence	Written in	Computation graph	Interface	Popularity	Usage	Creator (notes)
TensorFlow (Numerical framework)	Open source, Apache 2.0	C++, Python	Static with small support for dynamic graph	Python, C++[a], Java[a], Go[a]	Very High Growing very fast	Academic Industrial	– **Google**
Keras (Library)	Open source, MIT	Python	Static	Python Wrapper for TensorFlow, CNTK, DL4J, MXNet, Theano	High Growing very fast	Academic Industrial	F. Chollet
CNTK (Framework)	Open source, Microsoft permissive license	C++	Static	Python, C++, BrainScript, ONNX	Medium Growing fast	Academic Industrial Limited mobile solution	– **Microsoft**
Caffe (Framework)	Open source, BSD 2-clause	C++	Static	C++, Python, MatLab	High Growing fast	Academic Industrial	Y. Jia **BAIR**
Caffe2 (Framework)	Open source, Apache 2.0	C++	Static	C++, Python, ONNX	Medium-low Growing fast	Academic Industrial Mobile solution	Y. Jia **Facebook**
Torch (Framework)	Open source, BSD	C++, Lua	Static	C, C++, LuaJIT, Lua, OpenCL	Medium–low Growing low	Academic Industrial	R. Collobert, K. Kavukcuoglu, C. Farabet
PyTorch (Library)	Open source, BSD	Python, C	Dynamic	Python, ONNX	Medium Growing very fast	Academic Industrial	A. Paszke, S. Gross, S. Chintala, G. Chanan
MXNet (Framework)	Open source, Apache 2.0	C++	Dynamic dependency scheduler	C++, Python, Julia, MatLab, Go, R, Scala, Perl, ONNX	Medium Growing fast	Academic Industrial	– **Apache**
Chainer (Framework)	Open source, Owners permissive license	Python	Dynamic	Python	Low Growing low	Academic Industrial	– **Preferred Networks**
Theano (Numerical framework)	Open source, BSD	Python	Static	Python	Medium-low Growing low	Academic Industrial	Y. Bengio **University of Montreal**

참고

- Google Colab
  - 심층학습 또는 기계학습 모델을 실행할 수 있는 Google의 무료 클라우드 서비스
  - https://colab.research.google.com/


```
import tensorflow as tf
device_name = tf.test.gpu_device_name()
if device_name != '/device:GPU:0':
  raise SystemError('GPU device not found')
print('Found GPU at: {}'.format(device_name))
```

```
Found GPU at: /device:GPU:0
```

Statistics

- shares the basic premise of learning from data
  - use of observations to uncover an underlying process
  - the process: a probability distribution
  - the observations: sampled from that distribution
- emphasis is given to situations where
  - most questions can be answered within rigorous proofs

기계학습과 통계

comparison:

- statistics
  - focuses on *idealized models* and analyzes them in great detail
- machine learning
  - makes less restrictive assumptions
  - deals with more *general models* than in statistics
  - ends up with weaker results that are broadly applicable

Data mining

- a practical field that focuses on
  - finding patterns, correlations, or anomalies
  - often in large relational databases
- examples
  - look at medical records to detect a long-term drug effect
  - look at credit card spending patterns to detect potential fraud
  - recommender systems

comparison:

- data mining vs machine learning
  - technically, the same
  - DM: more emphasis on *data analysis* than on prediction
  - DBs are usually huge $\Rightarrow$ computational issues critical in DM

Machine learning versus data mining (Wikipedia)

- two terms are commonly confused
  - often employ the same methods and overlap significantly
- they can be roughly defined as follows:
  - ML focuses on prediction, based on known properties learned from the *training data*
  - DM focuses on the discovery of (previously) unknown *properties* in the data; the analysis step of Knowledge Discovery in Databases (KDD)