

# 1. Machine Learning

## 2019년 가을 학기

2019년 08월 23일

[https://www.github.com/KU-BIG/KUBIG\\_2019\\_Autumn](https://www.github.com/KU-BIG/KUBIG_2019_Autumn)

---

### 1. 머신러닝 (Machine Learning)

말 그대로 기계를 학습시키는 과정이다. 주어진 학습데이터를 통해 기계를 학습시키고 이에 새로운 데이터를 입력하여 원하는 결과(예측, 분류 및 기타 결과값 등)를 얻는 것이다. 예를 들어 나무 그림을 생각해보자. 먼저 수많은 나무 그림 데이터를 컴퓨터에 학습시킨다. 이를 통해 '나무'의 특징을 학습한 컴퓨터는 새로운 그림이 주어졌을 때 이를 '나무이다' 혹은 '나무가 아니다'와 같이 구별할 수 있게 된다. 이러한 일련의 과정을 '머신러닝'이라 부른다.

일반적인 머신러닝의 정의는 다음과 같다.

“머신러닝은 명시적인 프로그래밍 없이 컴퓨터가 학습하는 능력을 갖추게 하는 연구 분야이다.” -아서 사무엘(Authur Samuel, 1959)

좀 더 공학적인 정의는 다음과 같다.

“어떤 작업  $T$ 에 대한 컴퓨터 프로그램의 성능을  $P$ 로 측정했을 때 경험  $E$ 로 인해 성능이 향상됐다면 이 컴퓨터 프로그램은 작업  $T$ 와 성능 측정  $P$ 에 대해 경험  $E$ 로 학습한 것이다.” -톰 미첼(Tom Mitchell, 1997)

아서 사무엘이 말하는 '명시적인 프로그래밍'이란 사전에 사람에 의해 정해진 파라미터(Parameter)를 의미한다. 특정한 형태의 입력값(Input Data)을 입력하였을 때, 원하는 형태의 출력값(Output Data)을 얻기 위해서는 알고리즘이 필요하다. 이러한 알고리즘을 포함한 모델을 만들 때 그 모델의 특징을 결정하는 역할을 하는 변수가 바로 파라미터이다. 학습데이터를 사용한다는 점과 파라미터가 정해져 있지 않다는 점이 전통적인 인공지능과 머신러닝의 주요 차이점이다. 즉, 학습데이터를 가지고 어떠한 모델을 정해준다면 컴퓨터는 스스로 그 모델에 가장 적합한 파라미터를 찾는다.

## 2. 머신러닝의 종류

머신러닝은 '학습하는 동안의 감독형태나 정보량'에 따라 크게 지도학습, 비지도학습, 준지도학습, 강화학습으로 분류할 수 있다.

### 2.1 지도학습 (Supervised learning)

y라는 Output(레이블(Label) : 원하는 답)이 정해져 있다. 학습데이터는 x라는 Input과 y라는 Output의 쌍으로 구성되어 있다. 학습 후, 기계가 새로운 데이터 x가 주어졌을 때 y라는 결과(어떠한 Label값인가)를 예측하는 것이 지도학습의 주요 목표이다.

**Classification (분류):** 결과값이 범주형일 경우의 지도 학습이며, 스팸 메일 필터가 좋은 예시이다. 이 경우 Label은 '스팸 메일' 그리고 '일반 메일'로 나뉜다. 스팸 필터는 많은 메일 샘플과 레이블 정보(스팸인지 아닌지)로 훈련되어야 하며 어떻게 새 메일을 분류할지 학습해야 한다.

**Regression (회귀분석):** 결과값(Output)이 연속형일 경우의 지도 학습 작업이다. 여러가지 예측변수(Predictor Variable 주로 X로 표현)를 통해 타깃이 되는 수치(주로 y로 표현)를 예측(Predict)하는 것이 목적이다. 많은 중고차의 주행거리, 연식 브랜드 등의 정보와 중고 가격을 학습한 컴퓨터가 새로운 중고차의 중고가격을 예측하는 것이 회귀분석의 대표적인 예이다.

대표적인 지도학습의 알고리즘은 다음과 같다.

- KNN (K-Nearest Neighbors)
- 선형회귀 (Linear Regression)
- 로지스틱 회귀 (Logistic Regression)
- 서포트 벡터 머신 (Support Vector Machine 혹은 SVM)
- 결정트리 (Decision Tree)와 랜덤 포레스트 (Random Forest)
- 신경망 (Neural Network 혹은 NN)

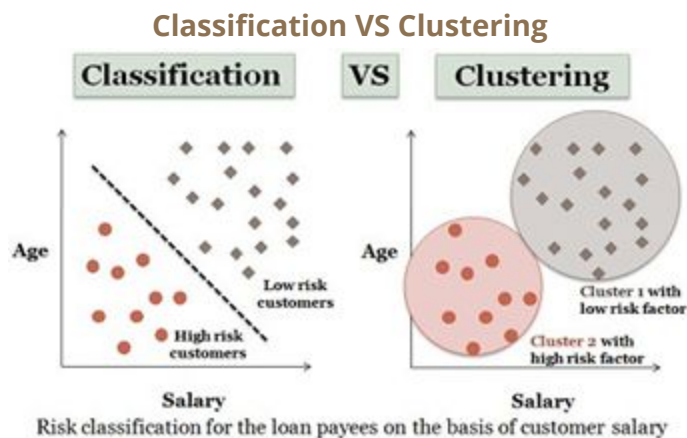
### 2.2 비지도학습 (Unsupervised learning)

비지도학습에는 말 그대로 훈련 데이터에 레이블이 없다. 시스템은 y에 대한 정보는 전무한 상태로 X정보만 가지고 학습해야 한다. 비지도학습의 알고리즘을 간략히 살펴보자.

### 2.2.1 군집화(Clustering)

- K평균 군집화 (K-Means Clustering)
- 계층적 군집 분석 (Hierarchical Cluster Analysis)
- 밀도 기반 군집화 (DBSCAN, Density-Based Spatial Clustering of Applications with Noise)

학습데이터의 다양한 변수들을 기준으로 여러 개의 군집을 나누는 것이다. 레이블이 주어지지 않았기 때문에 각 군집 별 특성을 임의로 지정해주고 이를 각 군집의 레이블처럼 사용한다.



Classification에는 이미 Low Risk와 High Risk라는 이름이 정해져 있다. 이를 통해 변수들이 (Age와 Salary) 두 집단을 적절하게 나누는가를 판단할 수 있다. 반면에 Clustering은 변수들을 통해서 군집을 형성하고 이 군집들이 서로 어떤 특징을 가질까에 대해 분석하는 과정을 거친다. 연구를 통해 연구자들은 두 군집이 Risk라는 척도에 따라 혹은 다른 어떠한 척도에 따른 다른 특징을 갖는다는 것을 알아낼 수 있다.

### 2.2.2 시각화(Visualization) / 차원축소(Dimension Reduction)

- t-SNE (t-distribution Stochastic Neighbor Embedding)
- 지역적 선형 임베딩(Locally-Linear Embedding 혹은 LLE)
- 주성분 분석 (Principal Component Analysis 혹은 PCA)
- 커널 PCA (Kernel PCA)

시각화 역시 비지도 학습 알고리즘의 좋은 예이다. 레이블이 없는 대규모의 고차원 데이터를 넣으면 도식화가 가능한 2D나 3D 표현을 만들어 준다. 이런 알고리즘은 가능한 한 구조를 그대로 유지하려 하므로(예를 들어 입력공간에서 떨어져 있던 클러스터는 시각화된 그래프에서

겹쳐지지 않게 유지된다.) 데이터가 어떻게 조직되어 있는지 이해할 수 있고 예상치 못한 패턴을 발견할 수도 있다.

차원축소는 비슷한 작업으로 낭비되는 효율성을 제고하며 데이터를 간소화하는 작업이다. 대표적으로 주어진 많은 X변수들의 많은 부분을 설명하면서 기존의 X변수보다 적은 수의 새로운 변수(주성분 Principal Component)를 만들어내는 주성분분석(PCA)이 있다.

### 2.2.3 연관규칙 학습(Association Rule)

연관규칙 학습에서는 대량의 데이터에서 특성 간의 흥미로운 관계를 찾는다. 예를 들어 슈퍼마켓 판매 데이터를 분석한다 가정하자. 판매 기록에 연관 규칙을 적용하면 바비큐 소스와 감자를 구매한 사람이 스테이크도 구매하는 경향이 있다는 것을 찾을지도 모른다.

연관규칙 학습은 조건부 확률에 기반하여 아이템 간의 규칙을 발견한다. 대표적으로 지지도(Support), 신뢰도(Confidence), 향상도(Lift)와 같은 여러 방식을 사용한다.

$$\text{지지도(support)} : \Pr(A \cap B) = \frac{A \text{와 } B \text{ 둘다 고른 사람 수}}{\text{전체 분석하는 총 인원 수}}$$

$$\text{신뢰도(confidence)} : \Pr(A | B) = \frac{A \text{와 } B \text{ 둘다 고른 사람 수}}{B \text{를 고른 사람 수}}$$

$$\text{향상도(lift)} \quad \frac{\Pr(A | B)}{\Pr(B)} = \frac{\Pr(A \cap B)}{\Pr(A) \star \Pr(B)}$$

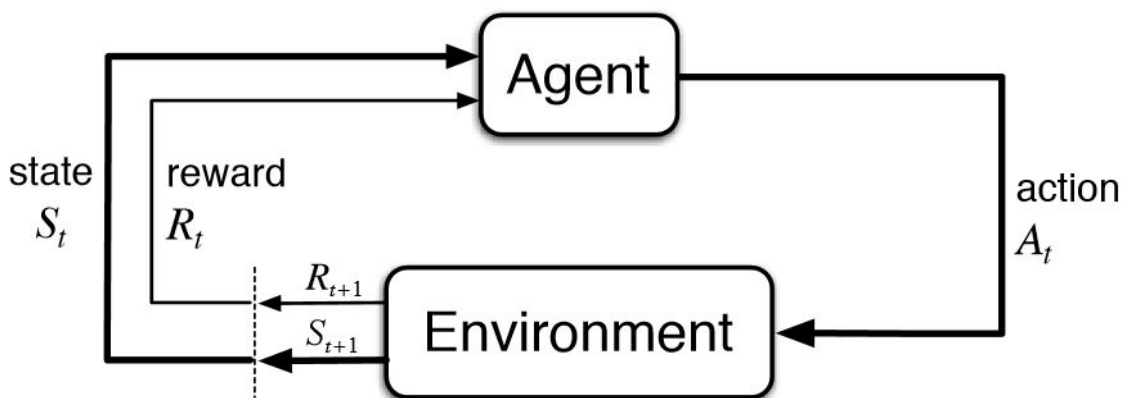
각 방식에 임의의 기준을 정하고 이를 기반으로 적당한 연관성을 찾는 방식이다. 예를 들어 '지지도는 0.6 이상, 신뢰도는 0.7 이상 중에서 향상도가 제일 높은 것은 무엇일까?' 와 같은 방식이다.

## 2.3 준지도 학습 (Semi-Supervised Learning)

어떤 알고리즘은 레이블이 일부만 있는 데이터도 다룰 수 있다. 보통은 레이블이 없는 데이터가 많고 레이블이 있는 데이터가 조금인 경우가 많다. 이를 준지도 학습이라 한다.

## 2.4 강화 학습 (Reinforcement Learning)

강화학습은 매우 다른 종류의 알고리즘이다. 여기서는 학습하는 시스템을 에이전트(Agent)라고 부르며 환경(Environment)을 관찰해서 행동(Action)을 실행하고 그 결과로 보상(Reward) 또는 벌점(Penalty ; 일종의 부정적인 보상)를 받는다. 시간이 지나면서 가장 큰 보상을 얻기 위해 정책(Policy)이라 부르는 최상의 전략을 스스로 학습한다. 정책은 주어진 상황에서 에이전트가 어떤 행동을 선택해야 할지 정의한다.



가장 대표적인 예가 google의 DeepMind Challenge의 알파고(AlphaGo)프로그램이다. 알파고는 수백만 개의 게임을 분석해서 승리에 대한 전략을 학습하며 자기 자신과도 계속해서 많은 게임을 한다. 알파고가 실제로 바둑기사와 게임을 할 때는 학습 기능을 끄고 그 동안 학습했던 전략을 적용한다.