Ican

Hands-On Machine Learning

Seokhun Ji's TIL (20.09.01)

1.1. 한눈에 보는 머신러닝

1. Machine Learning Preview

- 머신러닝이란?
- 왜 머신러닝을 사용하는가?
- 머신러닝의 시스템의 종류
- 머신러닝의 주요 도전 과제
- 테스트와 검증
- 연습문제

머신러닝이란?

일반적인 정의

명시적인 프로그래밍 없이 컴퓨터가 학습하는 능력을 갖추게 하는 연구분야다.
– Arthur Samuel, 1959

공학적인 정의

어떤 작업 T(Task)에 대한 컴퓨터 프로그램의 성능을 P(Performance)로 측정했을 때 경험 E(Experience)로 인해 성능이 향상됐다면, 이 컴퓨터 프로그램은 작업 T와 성능 P에 대해 경험 E로 학습한 것이다.
- Tom Mitchell, 1997

Example : 스팸필터 (T : 새 메일이 스팸인지 구분 | E : 훈련데이터 (Training data) | P : 정확히 분류된 메일의 비율, P 측정을 Accuracy라고도 부름)

왜 머신러닝을 사용하는가?

전통적인 프로그래밍 기법을 사용할 때
- 규칙이 복잡해지는 경우에는 유지보수에 어려움이 생김.

머신러닝 기반의 프로그램을 사용할 때
- (빈도 수 파악 등) 자동으로 규칙을 정립하여 별도의 작업없이 유지보수를 간결하게 함.

Example

1. 전통적 기법

ⓐ 스팸에 주로 사용되는 단어 분석

ⓑ 해당 패턴을 감지하는 알고리즘 작성

ⓒ 충분한 성능이 나올때까지 ⓐ,ⓑ를 반복
(규칙을 계속 추가해야하는 유지보수의 어려움 有)

2. 머신러닝 사용

사용자가 스팸으로 지정한 메일의 문자들을 분석하고 인식하여 자동으로 스팸 분류

+ 또한, 머신러닝 기술을 적용해서 대용량의 데이터를 분석하면 겉으로는 보이지않던 패턴을 발견할 수 있음.
= *데이터 마이닝*

왜 머신러닝을 사용하는가?

- *기존 솔루션으로 많은 수정과 규칙이 필요한 문제*
- *전통적 방식으로 해결법이 없는 복잡한 문제*
- *유동적인 환경*
- *복잡한 문제 및 대량의 데이터에서 통찰 얻기*

위와 같은 분야에 머신러닝이 뛰어나다고 말할 수 있음.

머신러닝 시스템의 종류

사람의 감독 하에 훈련하는가?
- 지도학습, 비지도학습, 준지도학습, 강화학습

입력 데이터의 스트림으로부터 (실시간으로) 점진적인 학습을 하는가?
- 온라인 학습, 배치 학습

단순 데이터 비교인가, 패턴 발견-예측모델 제작인가(어떻게 일반화 되는가)?
-사례 기반 학습, 모델 기반 학습

위는 배타적이지 않고 원하는 대로 연결 가능함.

지도학습 (Supervised Learning)

훈련 데이터에 레이블(label, 정답)이 포함 되는 경우
- 분류 (Classification)
- 회귀 (Regression, feature(Predictor Variable)를 사용해 타깃 수치를 예측)

지도학습 알고리즘 내역
- K-Nearest Neighbors (k-최근접 이웃)
- Linear Regression (선형 회귀)
- Logistic Regression (로지스틱 회귀)
- Support Vector Machines (서포트 벡터 머신)
- Decision Tree, Random Forests (결정트리, 랜덤 포레스트)
- Neural Networks (신경망)

비지도학습 (Unsupervised Learning)

훈련 데이터에 레이블(label)이 포함 되지 않는 경우

비지도학습 알고리즘 내역

- 군집 (Clustering)
  - k-Means (k-평균)
  - Hierarchical Cluster Analysis (HCA, 계층 군집 분석)
  - Expectation Maximization (기댓값 최대화)
- 시각화와 차원축소 (Visualization and Dimensionality Reduction)
  - Principal Component Analysis (PCA, 주성분 분석)
  - Kernel (커널)
  - Locally-Linear Embedding (LLE, 지역적 선형 임베딩)
  - t-SNE (t-distributed Stochastic Neighbor Embedding)
- 연관 규칙 학습 (Association rule learning)
  - Apriori (어프라이어리)
  - Eclat (이클렛)

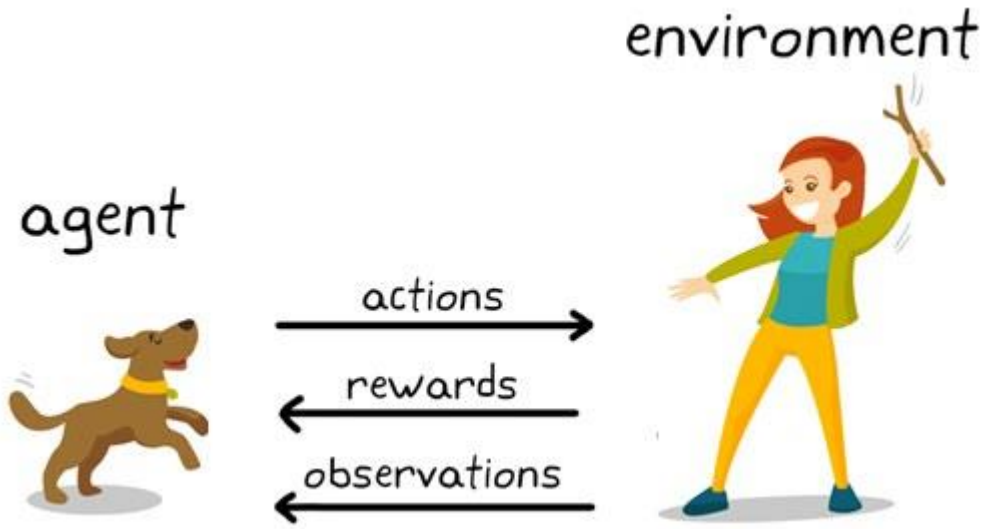
준지도학습 (Semisupervised Learning)

훈련 데이터에 레이블(label)이 일부만 있는 경우 (지도 학습과 비지도 학습의 조합으로 이뤄짐)

- Deep Belief Network(DBN, 심층 신뢰 신경망)
  - Restricted Boltzmann Machine(제한된 볼츠만 머신, 비지도학습)에 기초한 후 전체 시스템이 지도 학습 방식으로 조정 됨

강화학습 (Reinforcement Learning)

Agent(에이전트)가 Environment(환경)를 관찰해서 Action(행동)을 실행하고 결과로 Reward(보상) 혹은 Penalty(벌)를 받는 알고리즘
example) Alphago



https://www.youtube.com/watch?v=eVccQ82BekI

배치학습 (Batch Learning)

가용한 데이터를 모두 사용해 훈련시킴
점진적 학습이 불가능 (학습한 것을 단지 적용)
= Offline Learning (오프라인 학습)

- 배치 학습 시스템은 새 데이터에 대해 학습하려면 새 시스템 버전을 처음부터 다시 훈련해야 함.
- 변화에 적응할 순 있지만, 시스템이 빠르게 변하는 데이터에 적응해야 할 경우 좀 더 능동적인 방법이 필요

온라인학습 (Online Learning)

데이터를 순차적으로 한 개씩 (미니배치 단위로) 주입하여 시스템을 훈련
=> 연속적으로 데이터를 받거나 빠른 변화에 적응해야 하는 시스템에 적합 (ex 주식)

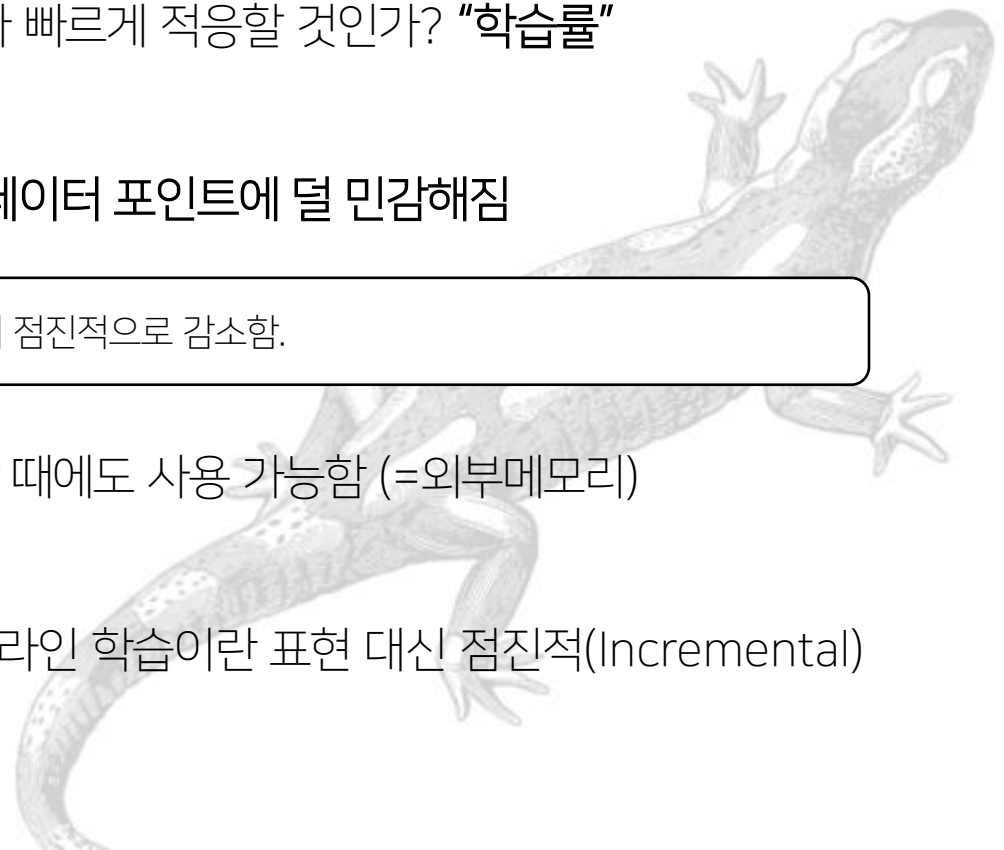
온라인 학습 시스템에서 중요한 파라미터 -> 변화하는 데이터에 얼마나 빠르게 적응할 것인가? **"학습률"**

If 학습률 ↑, 시스템이 데이터에 빠르게 적응, 예전 데이터를 금방 잊음
If 학습률 ↓, 시스템의 관성이 커져 느린 학습, 잡음 및 대표성이 없는 데이터 포인트에 덜 민감해짐

시스템에 나쁜 데이터가 주입될 경우 시스템 성능이 점진적으로 감소함.

+ 컴퓨터 한 대의 메인 메모리에 들어갈 수 없는 큰 데이터 셋을 학습할 때에도 사용 가능함 (=외부메모리)

CAUTION
- 이 외부메모리의 경우 전체 프로세스는 오프라인으로 실행되므로, 온라인 학습이란 표현 대신 점진적(Incremental) 학습이라 생각하는 게 좋음



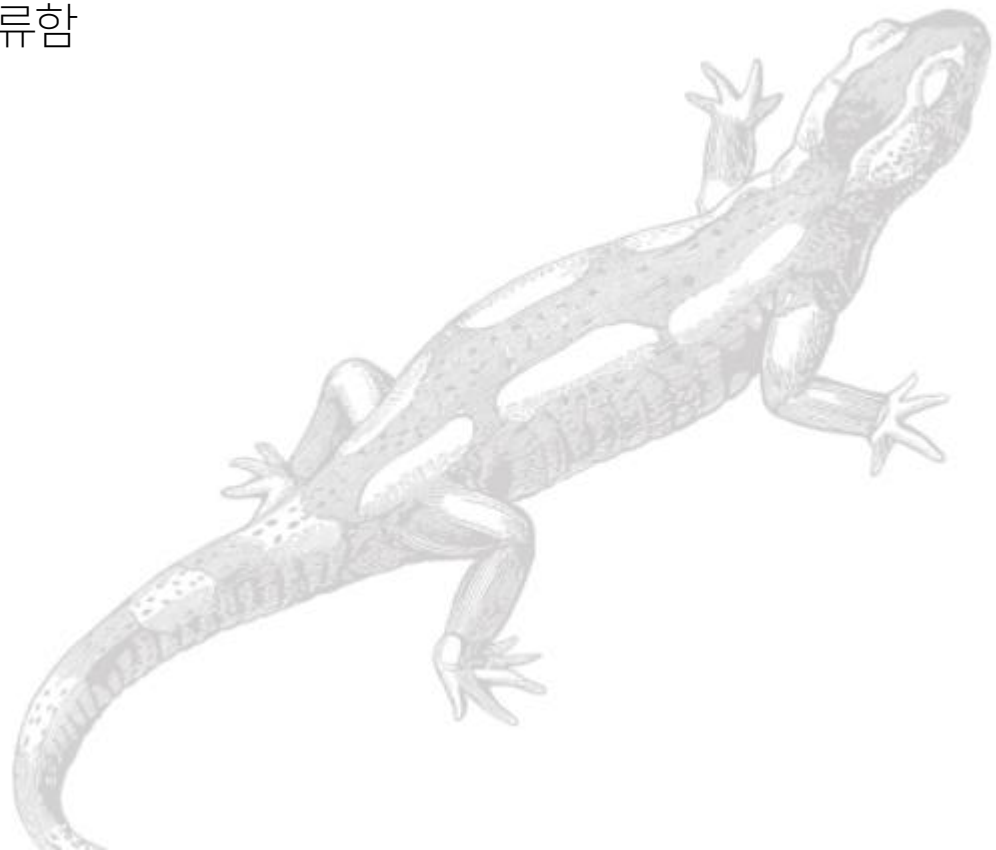
사례기반학습 (Instance-based Learning)

시스템이 사례를 기억함으로써 학습
유사도 측정을 사용해 새 데이터에 일반화 함

ex) 사용자가 스팸이라 지정한 메일과 동일한 모든 메일을 스팸으로 분류함

종류 내역

- Rote Learning
- k Nearest-Neighbor Classification
  - Prediction, Weighted Prediction
  - choosing k
  - feature weighting (RELIEF)
  - instance weighting (PEBLS)
  - efficiency
  - kD-trees
- IBL and Rule Learning
  - EACH: Nearest Nested Hyper-Rectangles
  - RISE



모델기반학습 (Model-based Learning)

샘플들의 모델을 만들어 예측에 사용함
모델 파라미터, 비용함수, 훈련 등의 개념 사용

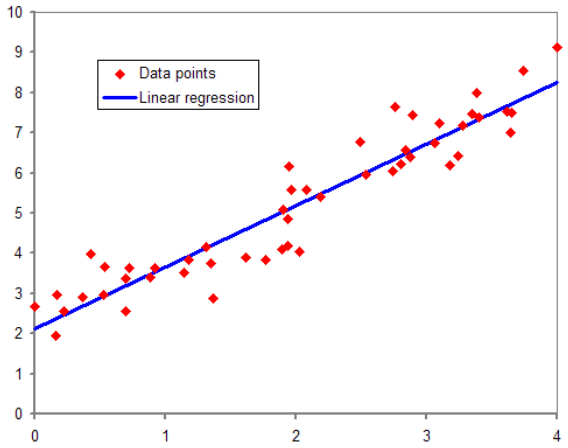
데이터의 생김새를 가정 (데이터 분석)

모델의 학습목표 수식화 하기 (모델 선택)

실제 데이터로 모델 학습하기
(최적화, 훈련데이터로 모델 훈련)

모델 적용하여 예측 및 평가하기

대표적 예시 - 선형회귀



머신러닝의 주요 도전과제

충분하지 않은 양의 훈련 데이터

대표성 없는 훈련 데이터

낮은 품질의 데이터 (정제)

관련 없는 특성 (특성 공학 – 선택 및 추출)

훈련 데이터 과대적합 (규제와 자유도) / 과소적합

한걸음 물러서서

정리를 하자면…

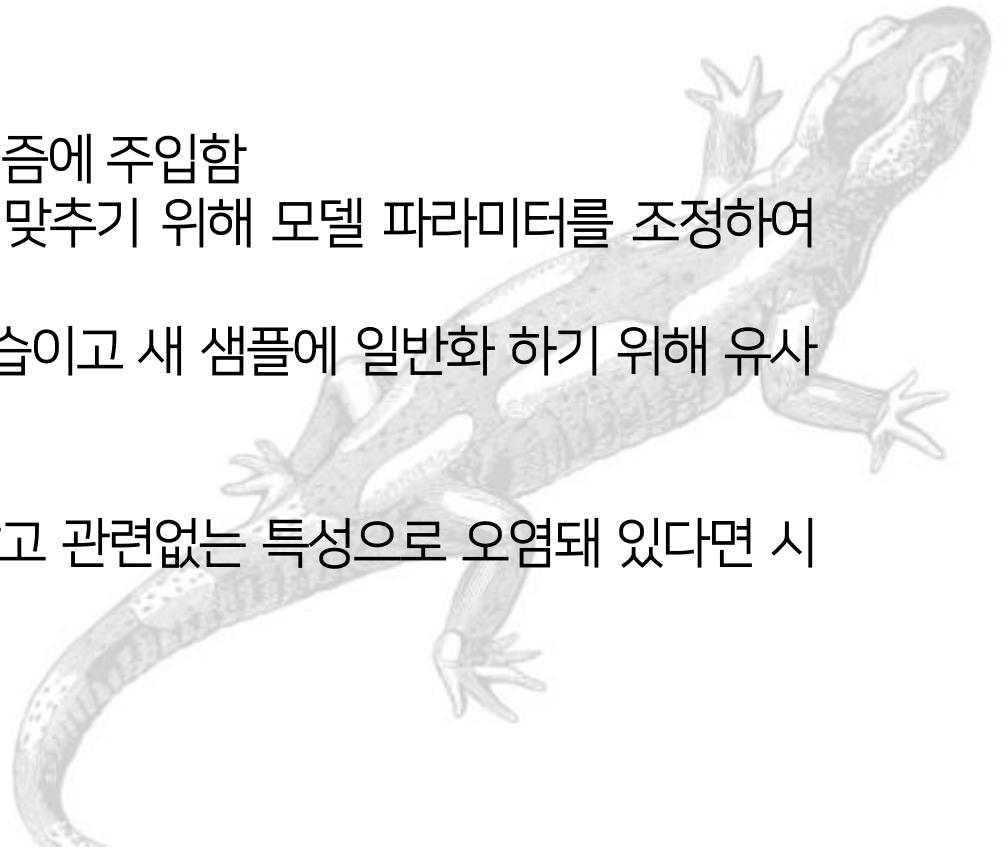
- 머신러닝은 기계가 데이터로부터 학습하여 어떤 작업을 더 잘하도록 만드는 것임

- 여러 종류의 머신러닝 시스템이 있음

- 머신러닝 프로젝트에선 훈련 세트에 데이터를 모아 학습 알고리즘에 주입함
->학습 알고리즘이 모델 기반이면 훈련세트에 모델을 맞추기 위해 모델 파라미터를 조정하여 새 데이터에서도 좋은 예측을 만들거라 기대함.
->알고리즘이 사례 기반이면 샘플을 기억하는 것이 학습이고 새 샘플에 일반화 하기 위해 유사도 측정을 사용함

- 훈련세트가 너무 작거나, 대표성이 없는 데이터거나, 잡음이 많고 관련없는 특성으로 오염돼 있다면 시스템이 잘 작동하지 않음

- 모델이 너무 단순하거나 복잡하지 않아야 함



테스트와 검증

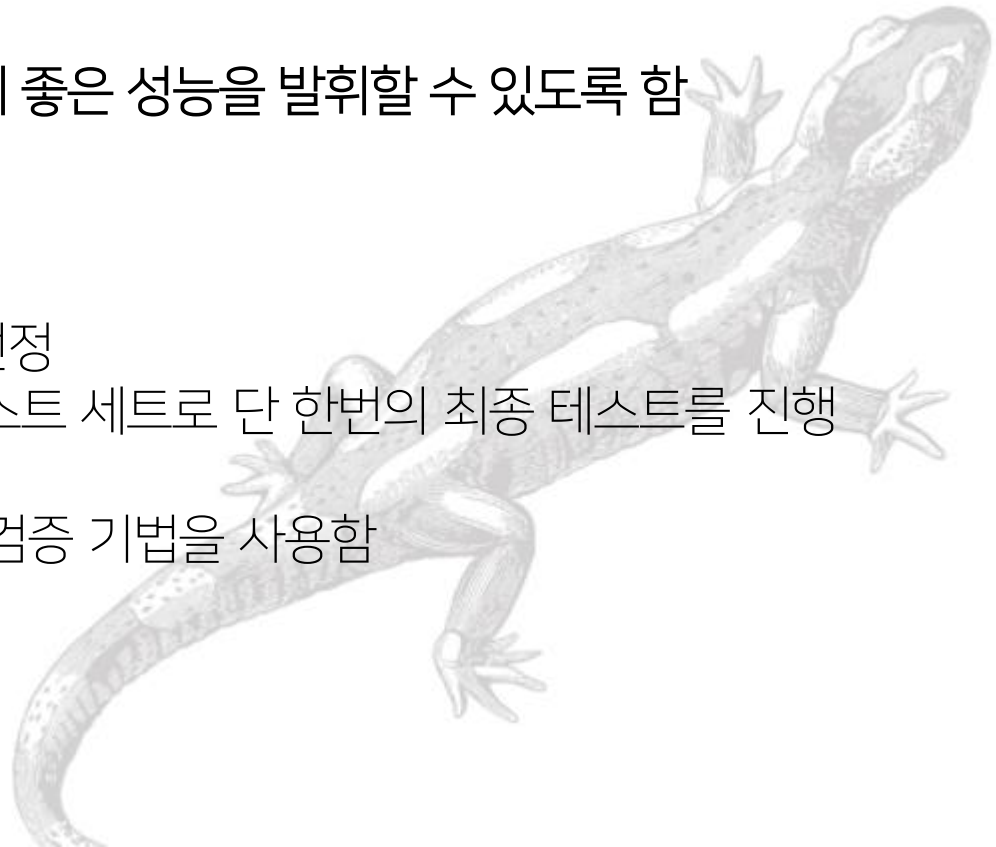
훈련데이터를 훈련 세트와 테스트 세트로 나누어 모델이 새 샘플에 얼마나 잘 일반화 될지 알아봄

- 새 샘플에 대한 오류 비율을 일반화 오차라 하며 테스트 세트에서 모델을 평가함으로써 이 오차에 대한 추정값을 얻음 (보통 80% 훈련용, 20% 테스트용)

또한 검증 세트(2nd holdout)를 만들어 실제 서비스 중인 모델에 좋은 성능을 발휘할 수 있도록 함

과정)
1.훈련 세트를 사용하여 여러 모델을 훈련시킴
2.검증 세트에서 최상의 성능을 내는 모델과 하이퍼파라미터를 선정
3.만족스런 모델을 찾으면 일반화 오차의 추정값을 얻기 위해 테스트 세트로 단 한번의 최종 테스트를 진행

- 이때 검증 세트로 너무 많은 훈련데이터를 뺏기지 않고자 교차 검증 기법을 사용함



Thanks