Basic concept of Machine Leanring

Jeonghun Yoon

Machine learning

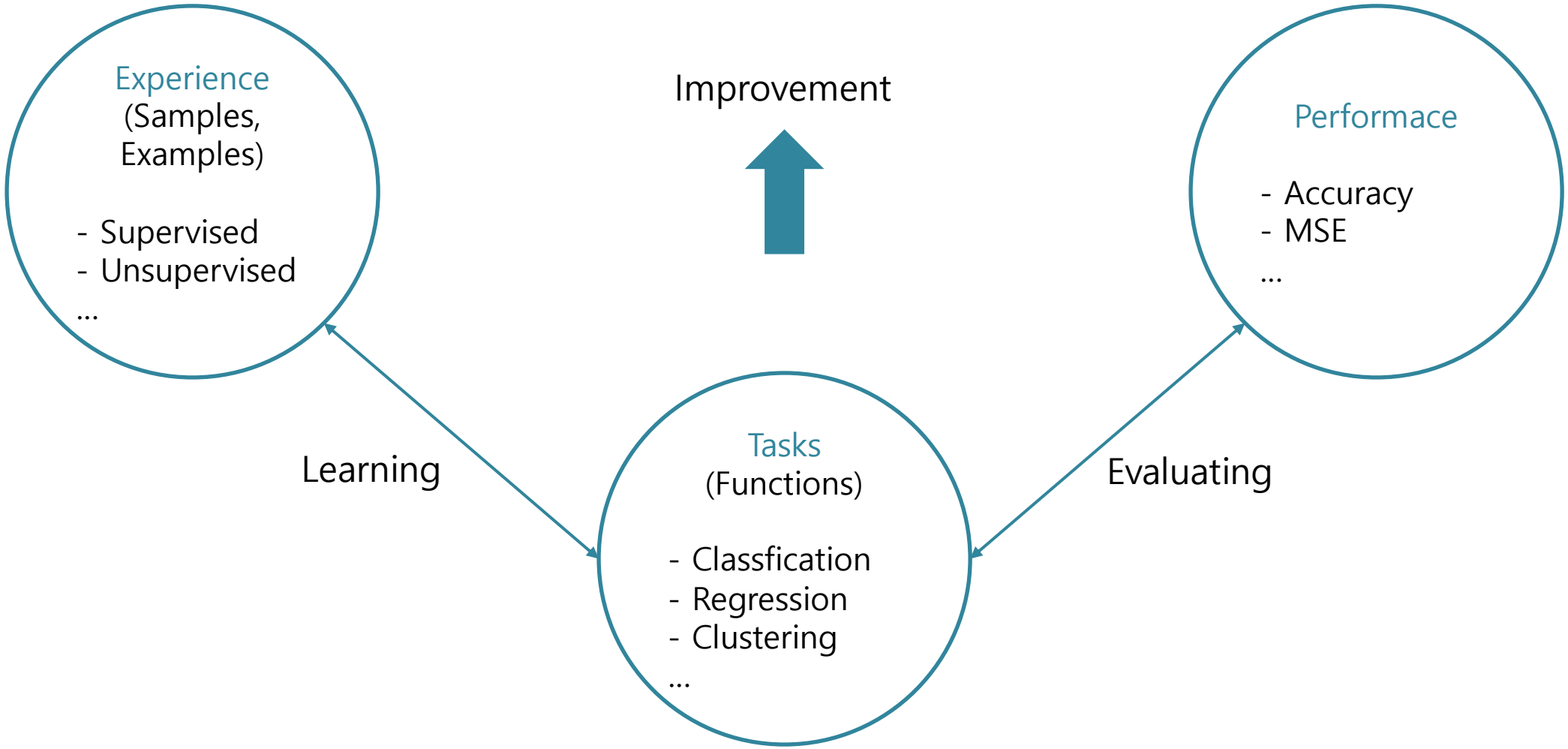
A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, imporves with experience E.

Machine Learing by Tom M.Mitchell

컴퓨터 프로그램은 다양한 기능을 수행한다.
그 기능을 잘 수행했는지 그러지 못했는지에 대해 성능을 평가 할 수 있다.

머신러닝은 컴퓨터가 수행하는 기능을, dataset을 통하여 더 나은 성능을 낼 수 있도록 개선하는 것이다.

Machine learning



Machine learning

(Easy version)

머신 러닝은 과거 경험에서 학습을 통해 얻은 지식을 미래의 결정에 이용하는 전산학의 한 분야이다.

머신 러닝의 목표는 관측된 패턴을 일반화하거나 주어진 예제를 통해 새로운 규칙을 생성해내는 것이다.

머신러닝 학습 알고리즘의 카테고리

머신러닝은 크게 세 가지 범주로 분류된다.

- Supervised Learning algorithm
- Unsupervised Learning algorithm
- Reinforcement Learning algorithm

머신러닝 학습 알고리즘의 카테고리

Supervised Learning algorithm

- Random sample vectors $\mathbb{x}'s$ 와 각각에 대응하는 t가 관찰되었을 때, 제공된 dataset을 사용하여 모든 $\mathbb{x}$에 대해 정답인 t를 유추해 낼 수 있도록 일반화시키는 알고리즘(예제를 통한 학습 알고리즘)
- Roughly $P(t|\mathbb{x})$
- Classification, Regression, ..
- Input, target

머신러닝 학습 알고리즘의 카테고리

Unsupervised Learning algorithm

- Random sample vectors $\mathbb{x}'s$가 관찰되었을 때, $\mathbb{x}$의 distribution or dataset 구조의 useful properties를 찾고자 하는 알고리즘
- Roughly $P(\mathbb{x})$
- Density estimation, Clustering, Dimensional reduction
- Input

머신러닝 학습 알고리즘의 카테고리

Reinforcement Learning algorithm

- Machine 또는 agent가 주변 환경의 feedback으로부터 행동을 학습하는 알고리즘
- 강화학습에서 agent는 지도를 받는 것이 아니라 스스로 일련의 결정을 내린 후, 그 결과에 따라 마지막에 +1 이나 -1의 보상 reward를 받음
- 보편적인 머신러닝 알고리즘보다는 인공지능 기법에 더 가깝다.

통계 모델링과 머신 러닝의 차이점

통계 모델링과 머신 러닝 간에는 근본적인 유사점이 있지만, 실제 적용에 있어서는 때때로 그 점이 분명하지 않을 때가 있다.

통계 모델링	머신 러닝
변수 간의 관계를 수학식을 통해 정량화	규칙 기반 프로그래밍(rule-based programming)에 의존하지 않고 데이터로부터 학습 가능한 알고리즘
데이터에 맞는 모델 적합화를 수행하기 전 미리 곡선의 형태를 가정해야 함(예: 선형, 다항 등)	머신 러닝 알고리즘은 주어진 데이터로부터 복잡한 패턴을 스스로 학습하는 알고리즘이므로 곡선의 형태를 미리 가정할 필요가 없음
통계 모델은 85%의 정확도와 90%의 신뢰 수준으로 결과를 예측함	머신 러닝은 결과를 정확도 85%로 예측함
통계 모델링에서는 P value 같은 다양한 매개변수 진단이 수행됨	머신 러닝 모델은 어떠한 통계적 유의성 진단도 수행하지 않음

통계 모델링과 머신 러닝의 차이점

통계 모델링	머신 러닝
데이터는 70:30 (때에 따라 다르게 split 가능)으로 나뉘어 각각 훈련 집합과 테스트 집합이 됨. 모델은 훈련 집합에서 개발되고 테스트 집합으로 테스트함.	데이터는 50:25:25(때에 따라 다르게 split 가능)으로 나뉘어 각각 훈련 데이터, 검증 데이터, 테스트 데이터가 됨. 모델은 훈련 데이터에서 개발되고, 초매개변수 hyperparameter는 검증 데이터를 통해 튜닝되고 최종적으로 테스트 데이터에 관해 평가함.
통계 모델은 훈련 데이터라 불리는 단일 데이터만으로도 개발 가능함. 진단은 전체 정확도 뿐만 아니라 개별 변수 단위에 관해서도 수행되기 때문임.	변수에 관한 진단이 없기 때문에 머신 러닝 알고리즘은 이중 검증을 위해 훈련 데이터와 검증 데이터라 불리는 두 데이터 세트에 관해 학습이 이뤄짐.
통계 모델링은 보토 연구 목적으로 사용됨.	머신 러닝은 실제 환경에서 구현하기 적합함

머신 러닝 모델 개발 순서

1. 해결하고자 하는 문제의 정의
2. 데이터 수집
3. 데이터 준비와 결측 값 / 이상 값 처리
4. 데이터 분석(Exploratory Data Analysis)과 feature engineering
5. 모델 선택
6. 훈련 및 검증 데이터에 이용하여 모델 훈련 및 Hyperparameter 튜닝
7. 테스트 데이터를 사용하여 모델의 최종 테스트
8. 모델 배포 및 서비스에 활용

머신 러닝 모델 개발 순서

데이터 수집

- 머신 러닝 데이터는 구조화된 소스, 웹 스크래핑, API, 채팅 등을 통해 집적 수집한다. 머신 러닝은 구조화된 데이터(table data, DB data 등)와 비구조화 데이터(음성, 이미지, 텍스트)를 모두 처리 할 수 있다.

데이터 준비와 결측 값 / 이상 값 처리

- 데이터를 모델에서 사용할 수 있도록 알맞게 가공한다. 결측 값missing value, null value 또는 이상 값abnormal value는 데이터 및 해결하고자 하는 문제의 성질에 맞게 평균 값, 중간 값 등으로 대체한다.

데이터 분석(EDA)과 feature engineering

- 변수들 사이에 숨겨진 패턴 및 관계를 찾아내려면 데이터를 분석해야 한다. 올바른 feature engineering과 적절한 비지니스 지식을 동원한다면 많은 문제를 해결할 수 있고, 모델이 잘 학습되는 데에 큰 도움이 된다.

머신 러닝 모델 개발 순서

훈련 및 검증 데이터에 이용하여 모델 훈련

- Feature engineering이 끝난 후 모델을 학습하는 단계이다. 데이터를 3개 집합(훈련, 검증, 테스트)로 분리한다. 훈련 데이터에서 머신 러닝 모델을 훈련 시키고, overfitting$_{과적합}$을 피하기 위해 검증 데이터를 대상으로 모델의 hyperparameter를 튜닝하는 과정을 거친다.

테스트 데이터를 사용한 모델 테스트

- 모델이 훈련 데이터와 검증 데이터를 상대로 충분히 좋은 성능을 발휘하면 새로운$_{unseen}$ 테스트 데이터를 대상으로 성능 점검을 수행한다. 이 테스트에서 충분히 성능이 인정되면 마지막 단계로 넘어간다.

모델 배포 및 서비스에 활용

- 학습된 머린 러닝 모델을 배포하고, 학습된 머신 러닝 알고리즘을 서비스에 적용한다.

머신 러닝에서 사용하는 기본적인 용어들

Input / Input vector / Features / Feature vectors / 입력 / 특성

- 입력 벡터는 머신러닝 알고리즘의 입력으로 주어진 데이터를 의미한다. Input data가 총 m차원이면, $\mathbb{x} = (x_1, x_2, \ldots, x_m)$라고 표현한다. 벡터의 요소인 x_i는 entity이다.

Output / Output vector / 출력값

- 출력 벡터는 머신러닝 알고리즘의 출력 결과 데이터를 의미한다. 출력 벡터의 각 entity는 알고리즘의 종류에 따라서 연속적인 실수값이 될 수 있고, 불연속적인 정수값이 될 수 있다. 또한 1차원이 될 수도 있고 다차원이 될 수도 있다. 출력은 회귀모형을 설명할 때는 $\hat{y}$라고 표현하겠다.

Target / Label / 목표값 / 라벨

- 목표값은 입력 벡터와 함께 pair로 주어진, 즉 $\mathbb{x}$에 associate / assign 된 데이터를 의미한다. 실수값, 정수값을 가질 수 있다. 타겟은 회귀모형을 설명할 때는 y라고 표현하겠다. Supervised learning에서는 $(\mathbb{x}_1, y_1), (\mathbb{x}_2, y_2), \ldots, (\mathbb{x}_n, y_n)$이 모델의 학습에 사용된다.

머신 러닝에서 사용하는 기본적인 용어들

Training set

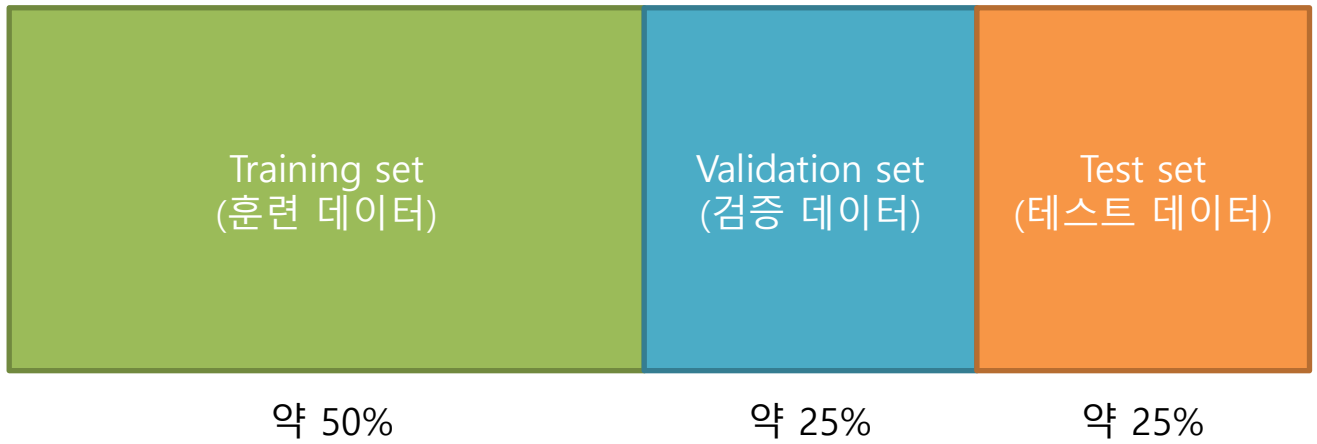
- 알고리즘을 학습할 때 사용되는 input dataset (supervised learning의 경우는 target도 training set에 포함)

Validation set

- 알고리즘을 학습할 때, 알고리즘의 성능을 (중간에) 측정하여 성능이 나쁜 알고리즘을 구별해내고(prune) 좋은 알고리즘을 선택할 수 있도록 하는 과정에서 사용되는 dataset

Test set

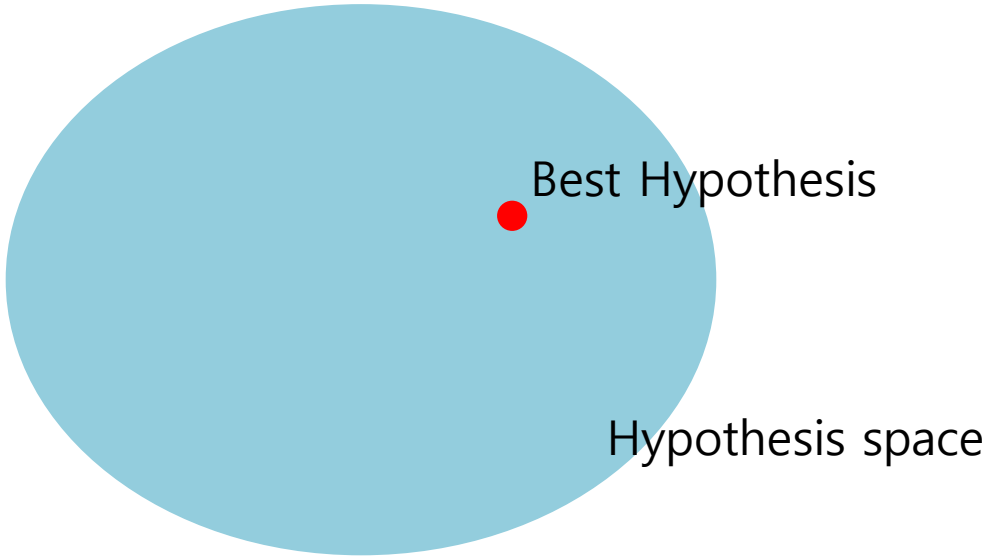
- 학습된 알고리즘을 최종 테스트하여 알고리즘의 성능을 평가할 때 사용하는 dataset



머신 러닝에서 사용하는 기본적인 용어들

Hypothesis

- 우리가 찾고자하는 알고리즘을 의미한다. 머신러닝 알고리즘을 학습하는 것은, 결국 머신러닝 알고리즘이 존재하는 전체 셋(hypothesis set)에서 최상의 머신러닝 알고리즘(hypothesis)을 찾는 것이다.

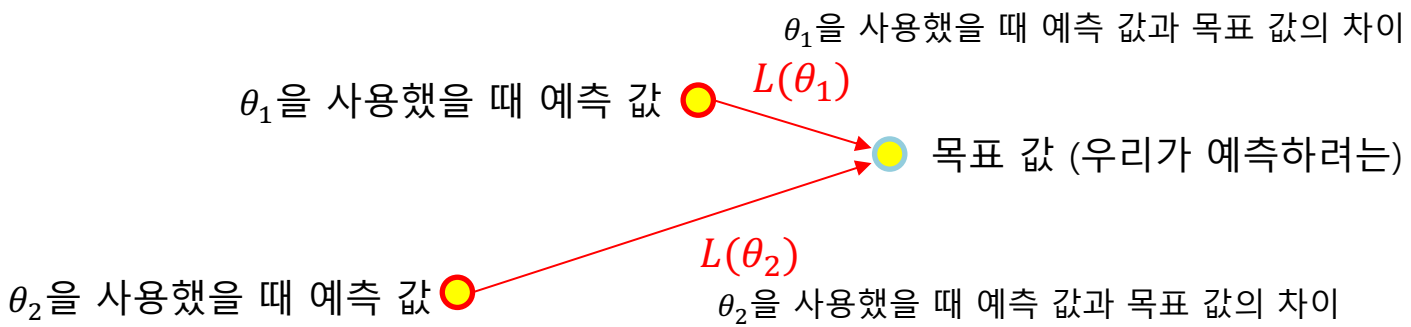


Model의
학습 정도를 표현할 때,
Hypothesis를 사용하여
수학적으로 표기하는 경우가 많다.

머신 러닝에서 사용하는 기본적인 용어들

손실 함수loss function 또는 비용 함수cost function

- 모수(θ)의 값을 실수$L(\theta)$로 mapping 하는 함수이다. 구하고자 하는 모수(파라미터)를 θ라고 할 때, loss function 은 $L(\cdot)$로 주어질 것이다.
- 모수에 대응되는 실수는, 우리가 모수를 사용했을 때 예측 값과 실제 값과의 거리를 나타낸다고 생각하면 된다. (정확히는 거리가 아니지만, 그렇게 생각하면 이해하기 쉽다.) 즉, 모수를 선정할 때 기준이 되는 값이다.



θ_1 을 사용했을 때
손실 함수의 값 L이 더 작다.
따라서 θ_1 을 사용한다.

머신 러닝에서 사용하는 기본적인 용어들

대표적인 손실 함수

- 제곱 손실Squared loss : 회귀에서 많이 사용함
  - MSE
  - RMSE
- Cross entropy : 분류모델에서 많이 사용함
- 힌지 손실Hinge loss : SVM(분류 모델)에서 사용함

머신 러닝 모델 summary

지도 학습 (입력 변수와 목표 변수가 주어진다.)

- 회귀 모형 (목표 변수가 연속적인 값이다.)
  - 단순 선형 회귀 모형
  - Lasso, Ridge 회귀
  - 의사결정 트리 회귀
  - Bagging 회귀
  - Random forest 회귀
  - Boosting 회귀
  - SVM 회귀

머신 러닝 모델 summary

지도 학습 (입력 변수와 목표 변수가 주어진다.)

- 분류 모형 (목표 변수가 이산값을 가진다.)
  - 로지스틱 회귀
  - 의사결정 트리(분류기 트리)
  - Bagging 분류
  - Random forest 분류
  - 부스팅 분류(Adaboost, Gradient boost, Xgboost)
  - SVM 분류
  - Perceptron

머신 러닝 모델 summary

비지도 학습 (목표 변수가 주어지지 않는다.)

- 주성분 분석(PCA)
- K-means clustering

강화 학습

- 마르코프 결정 프로세스
- 몬테카를로 기법
- 시간차 학습

머신 러닝 모델 summary

선형 회귀

- 이 방법은 고객의 소득 같은 연속 변수의 예측에 사용된다. 모델은 최적의 선을 fitting하기 위해, 손실 함수의 값이 선형 방적식의 계수들 $\beta_i's$에 관해 최소화되도록 한다. 선형 회귀는 높은 bias와 낮은 variance 오류의 특징을 가진다.
- 선형 방정식의 기본적인 식 : $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$

Lasso / Ridge 회귀

- 이 기법은 선형 방정식의 계수에 penalty를 적용한 규제화 과정을 통해 과적합 문제를 통제한다. Ridge 회귀는 계수의 제곱의 합, Lasso는 계수의 절대값에 penalty 를 적용한다. 또한 penalty 의 양을 조절할 수 있다. Ridge 회귀는 계수의 크기를 최소화하는 반면, Lasso 회귀는 계수를 제거하려고 노력한다. 즉 계수의 값이 0이 나올 수 도 있다.

머신 러닝 모델 summary

의사결정 트리 분류

- 각 레벨에는 이진 분할을 적용한다. 분할은 각 레벨의 부류가 최대한 순수한 부류$_{class}$만 남을 때까지 반복한다. 분류 오류율은 단순히 그 구역의 훈련 관측 값 중 가장 일반적인 부류에 속하지 않는 관측 값들의 비율이다. 즉 그 지역의 훈련 관측 값 중 많았던 부류가 A 였다면, 그 지역의 관측 값 중 A 부류가 아닌 부류의 비율이다. 의사결정 트리는 fitting 과정에서 높은 분산으로 인한 overfitting 문제를 겪는다. 가지치기(pruning)을 통해 overfitting 문제를 감소시킬 수 있다. 의사결정 트리는 낮은 bias와 높은 variance 오류의 특징을 가진다.

머신 러닝 모델 summary

Bagging 분류 / 회귀

- Bagging은 의사결정 트리에 앙상블을 적용해 variance 오류를 최소화하는 동시에 bias에 의한 오류 성분이 증가하지 않도록 하는 기술이다. Bagging은 복원 추출을 통해 표본을 선택하고, **모든 변수**(열)은 각 표본에 관해 의사결정 트리를 개별적으로 fitting한다. 그 후 투표(회귀의 경우, 출력값의 평균)를 통해 최종 결과를 ensemble한다.

Random Forest

- 이 방법은 Bagging과 흡사한데, 한 가지 차이가 있다. Bagging은 각 표본 별로 모든 변수(열)를 선택하지만 Random forest는 **몇 개의 열만 선택**한다. 전체 변수를 선택하지 않고 일부만 선택하는 이유는 각 개별 트리의 표본을 추출할 때 많은 수의 변수가 항상 트리의 상위 계층에 나타나 결과적으로 분할 후에도 모든 트리가 유사하게 자라게 돼 앙상블의 취지에 반하기 때문이다. 이 방법은 bias와 variance 오류가 모두 낮은 특징이 있다.

머신 러닝 모델 summary

Boosting 분류 / 회귀

- Boosting 은 순차 알고리즘으로 결정 그루터기(단일 레벨 결정 트리 또는 하나의 root와 2개의 단말 노드를 가진 트리) 같은 약한 분류기에 적용한 후 그 결과를 앙상블함으로써 강력한 분류기로 탈바꿈하기 위해 적용한다. 알고리즘은 모든 관측값에 동일한 가중값을 할당하고 시작한다. 그 후 후속 반복 작업에서 잘못 분류된 관측값에는 가중값을 증가시키고 제대로 분류된 관측값은 가중값을 감소시킨다. 결국 모든 개별 분류기는 더 강력한 분류기로 합쳐진다. Boosting은 overffiting 문제가 있을 수 있지만, 매개변수를 세밀히 튜닝함으로써 스스로 학습하는 최고의 머신 러닝 모델을 얻을 수 있다.

머신 러닝 모델 summary

SVM

- 이 방법은 부류 사이의 경계를 가장 넓게 형성하는 초평면을 학습해서 경계 간의 간격(margin)이 최대가 되게 한다. 비선형으로 분리 가능한 부류의 경우 관측값을 높은 차원의 공간으로 옮기기 위해 커널(kernel)을 사용한다. 그 후 초평면을 이용해 이를 선형 분리한다.

추천 엔진

- 이 방법은 협업 필터링 알고리즘(Collaborative filtering algorithm)을 사용해서 특정 고객이 과거에 사용한 적이 없는 상품 중 구매 확률이 높은 아이템을 찾아내기 위해 비슷한 성향의 고객이 가진 구매 취향을 파악하는 기법이다. 이 문제를 풀기 위해 교대 최소 자승법(Alternating least squares, ALS)을 활용한다.

머신 러닝 모델 summary

PCA

- 이 기법은 주성분을 원래 변수 대신 계산해 전체 차원을 낮추는 기술이다. 무엇을 주성분으로 할 것인지는 데이터의 분산이 최대값이 되는 것으로 판단한다. 결과적으로 전체 분산의 x%를 차지하는 최상위 n개 성분을 선택해 다음 단계 모델링 프로세스에서 사용하거나, 비지도학습으로 탐색적 요인 분석(Exploratory Data Analysis)를 수행한다.

K-means clustering

- 이 기법은 비지도학습 알고리즘의 하나로, 주로 분할에 활용된다. K-means clustering은 주어진 데이터를 k 개의 군집으로 분류하는데, 각 군집 내의 분산은 최소화시키되, 군집 간의 분산은 최대화시키는 것이 목표다.

머신 러닝 모델 summary

Markov decision process

- Markov decision process(MDP)는 강화학습에서 결과가 통제 되지 않은 임의의 영향을 부분적으로 받는 환경이나 상황에서 에이전트의 결정을 모델링하기 위한 수학적 프레임워크를 제공한다. 이 모델은 시스템을 통제하기 위해 에이전트가 취할 수 있는 상태와 행동의 집합으로 환경을 모델링한다. 에이전트의 총 수익이 최대화되도록 시스템을 통제하는 것이 이 모델의 목표다

Monte Carlo method

- Monte Carlo method는 환경에 관한 모든 지식을 필요로 하지 않는다. 이 기법은 실 데이터나 시뮬레이션 환경에서 추출된 표본의 상태 순서, 행동, 보상 데이터를 기반으로 한다. 이 기법은 주어진 표본에서 순차적으로 마지막 결과까지 공간을 탐색하고 해당 값을 갱신한다.

머신 러닝 모델 summary

시간차 학습

- Temporal difference learning은 강화학습의 핵심이다. 시간차 학습은 몬테카를로와 동적 프로그래밍 아이디어를 조합한 것이다. 시간차 학습은 몬테카를로와 비슷하게 환경의 역학을 모델링하지 않고도 원시 경험으로부터 바로 학습 할 수 있다. 또 동적 프로그래밍처럼 이 기법은 최종 결과를 기다리지 않고 다른 학습 결과에 일부 의존해 계산을 갱신한다. 이 방법은 통계나 머신 러닝을 통틀어 좋은 방법이고, 알파고 같은 게임에 주로 이용된다.