



핸즈온 머신러닝

챕터 1 - 한눈에 보는 머신러닝

22.04.06



1.1 머신러닝이란?

머신러닝은 데이터에서부터 학습하도록 컴퓨터를 프로그래밍하는 과학입니다.

- 컴퓨터가 학습하는 능력을 갖추게 하는 연구 분야



1.1 머신러닝이란?

- Training set: 시스템이 학습하는데 사용하는 샘플
- Training instance(sample): 훈련 데이터
- 작업: 새로운 메일이 스팸인지 구분하는 것
- 경험: 훈련 데이터
- 정확도: 성능 측정



1.2 왜 머신러닝을 사용하는가?

스팸 필터 만들기

1. 스팸에 주로 나타나는 단어 찾기
2. 각 패턴을 감지하는 알고리즘 작성 - 스팸으로 분류할 수 있도록
3. 프로그램을 테스트하고 충분한 성능이 나올 때까지 1,2단계 반복

문제점: 유지 보수

1.2 왜 머신러닝을 사용하는가?

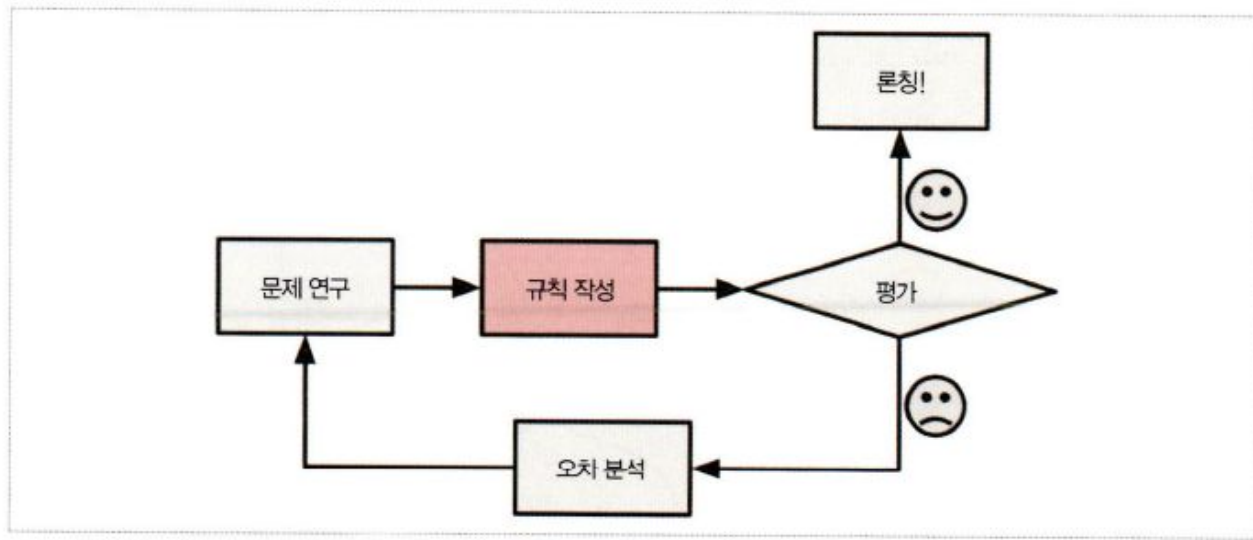


그림 1-1 전통적인 접근 방법

1.2 왜 머신러닝을 사용하는가?

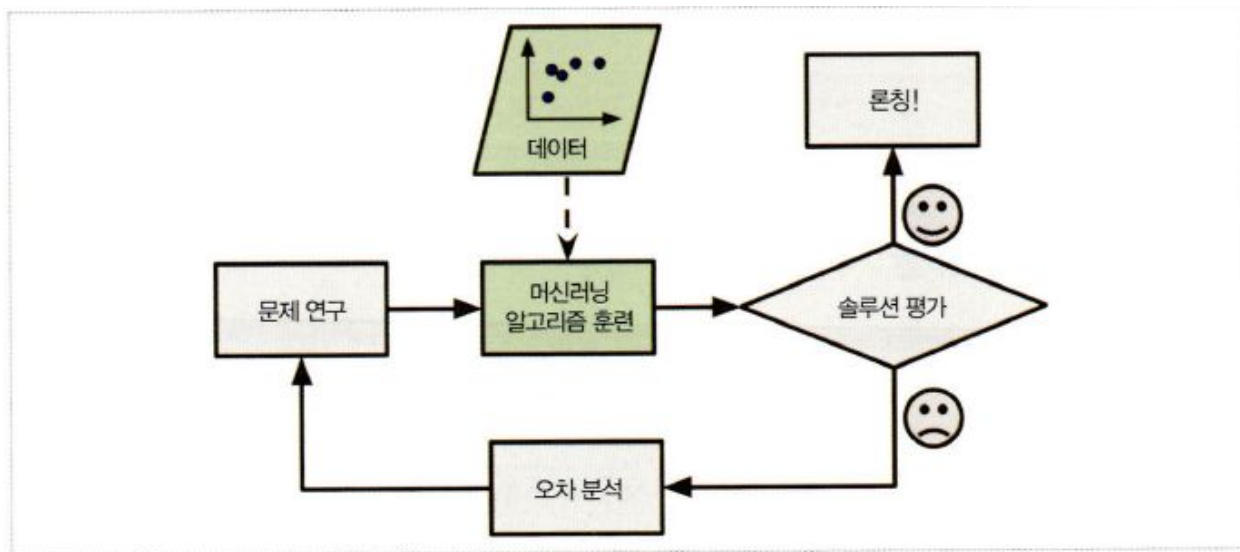
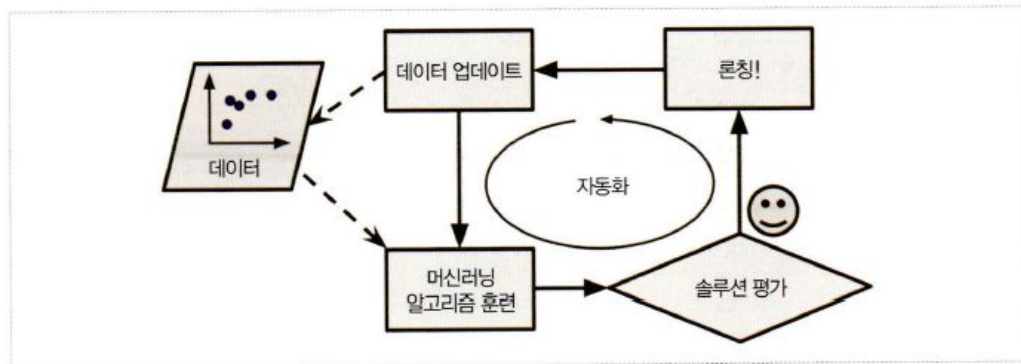


그림 1-2 머신러닝 접근 방법

1.2 왜 머신러닝을 사용하는가?

- 전통적 방식: 메일을 구분하기 위해 직접 지속적인 수정 필요
- 머신러닝 기반: 별도의 작업을 인식하지 않아도 자동으로 패턴을 인식하여 분류
 - 데이터를 업데이트 한다.





1.2 왜 머신러닝을 사용하는가?

데이터 마이닝(data mining)

머신러닝 기술을 적용해서 대용량의 데이터를 분석하면 겉으로는 보이지 않던 패턴을 발견하는 것

- 머신러닝 알고리즘이 학습한 것을 활용 할 수 있음



1.2 왜 머신러닝을 사용하는가?

데이터 마이닝(data mining) 활용 분야

- 기존 솔루션으로는 많은 수동 조정과 규칙이 필요한 문제
- 전통적 방식으로 해결 방법이 없는 복잡한 문제
- 유동적인 환경
- 복잡한 문제와 대량의 데이터에서 통찰이 필요할 때



1.3 애플리케이션 사례

p. 34-35 참고



1.4 머신러닝 시스템의 종류 ★

- 사람의 감독하에 훈련: 지도, 비지도, 준지도, 강화 학습
- 실시간으로 점진적인 학습 여부: 온라인 학습, 배치 학습
- 데이터 포인트 비교: 사례 기반 학습
- 훈련 데이터셋에서 패턴 발견하여 예측 모델 만들기: 모델 기반 학습

1.4.1 지도 학습과 비지도 학습

지도 학습(Supervised Learning)

알고리즘에 주입하는 훈련 데이터에 레이블(label)이라는 답이 포함됩니다.

1.4.1 지도 학습과 비지도 학습

지도 학습

- 분류(classification): 입력에 대해 순서가 없는 클래스(라벨)을 대응시키는 문제

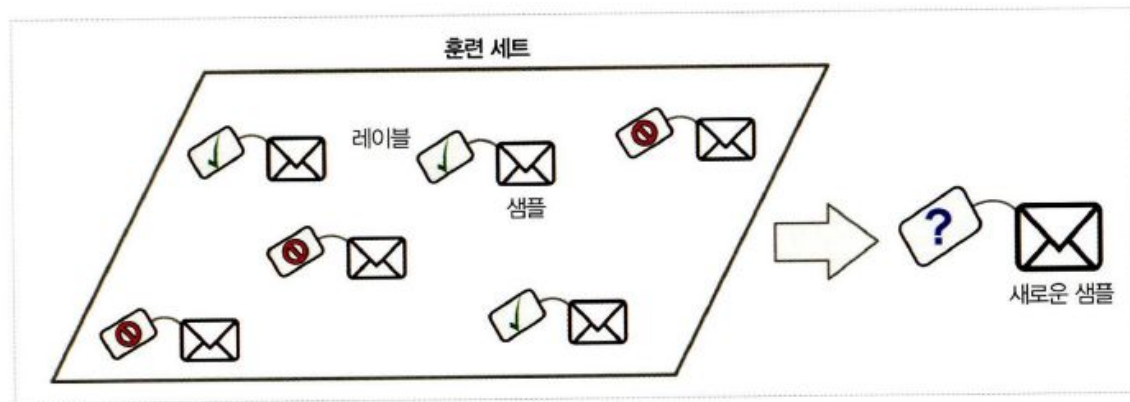


그림 1-5 스팸 분류를 위한 레이블된 훈련 세트(지도 학습의 예)

1.4.1 지도 학습과 비지도 학습

지도 학습

- 회귀(regression): 예측 변수(predictor variable)이라 부르는 특성(feature)을 사용해 타깃(target) 수치를 예측하는 문제

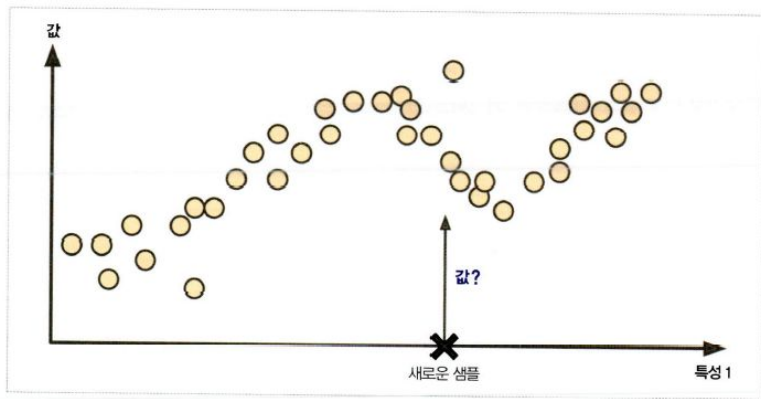


그림 1-6 회귀 문제: 주어진 입력 특성으로 값을 예측(일반적으로 입력 특성이 여러 개 있으며 이따금 값을 여러 개 출력하는 경우도 있습니다)

1.4.1 지도 학습과 비지도 학습



지도 학습 알고리즘

- K-최근접 이웃(k-nearest neighbors)
- 선형 회귀(linear regression)
- 로지스틱 회귀(logistic regression)
- 서포트 벡터 머신(support vector machine; SVM)
- 결정 트리(decision tree)와 랜덤 포레스트(random forest)
- 신경망(neural networks)

1.4.1 지도 학습과 비지도 학습

비지도 학습

훈련 데이터에 레이블이 없음. 아무런 도움 없이 학습해야 한다.

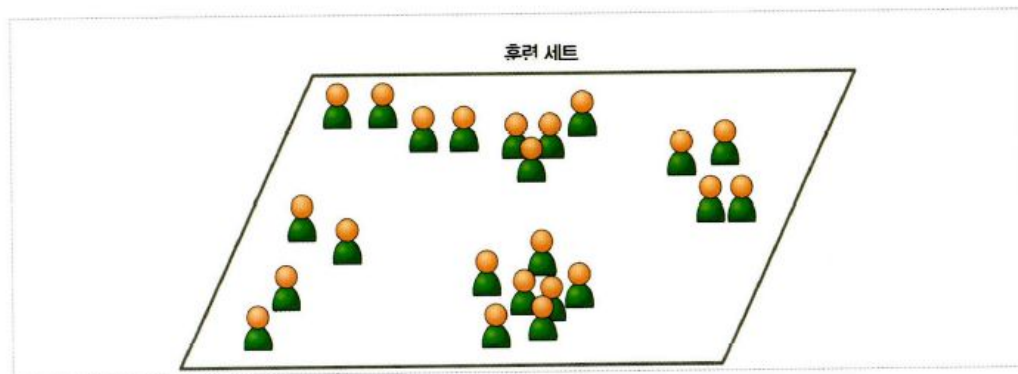


그림 1-7 비지도 학습에서 레이블 없는 훈련 세트

1.4.1 지도 학습과 비지도 학습



비지도 학습 알고리즘

군집(Clustering)

- k-평균(k-means)
- DBSCAN
- 계층 군집 분석(hierarchical cluster analysis; HCA)\
- 이상치 탐지(outlier detection)와 특이치 탐지(novelty detection)
- 원-클래스(one-class SVM)
- 아이솔레이션 포레스트(isolation forest)

1.4.1 지도 학습과 비지도 학습



비지도 학습 알고리즘

시각화(visualization)와 차원 축소(dimensionality reduction)

- 주성분 분석(principal component analysis; PCA)
- 커널(kernel) PCA
- 지역적 선형 임베딩(locally-linear embedding, LLE)
- t-SNE*t-distributed stochastic neighbor embedding)

1.4.1 지도 학습과 비지도 학습



비지도 학습 알고리즘

연관 규칙 학습(association rule learning)

- 어프라이어리(Apriori)
- 이클렛(Eclat)

1.4.1 지도 학습과 비지도 학습

비지도 학습

계층 군집(hierarchical clustering)

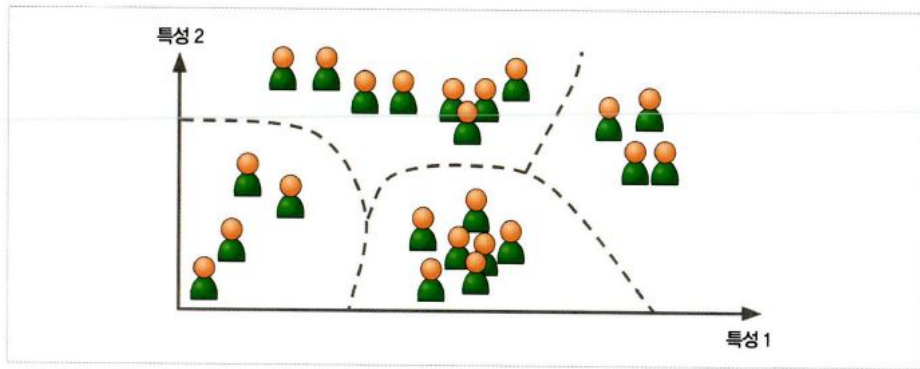


그림 1-8 군집

1.4.1 지도 학습과 비지도 학습

비지도 학습

시각화(visualization)

레이블이 없는 대규모의 고차원 데이터를 넣으면 2D, 3D로 도식화

차원 축소(dimensionality reduction)

정보를 잃지 않으면서 데이터를 간소화 하는 방법

특성 추출(feature extraction) - 상관 관계가 있는 여러 특성을 하나로 합치는 방법

1.4.1 지도 학습과 비지도 학습

비지도 학습

이상치 탐지(outlier detection)

- 학습 알고리즘에 주입하기 전에 데이터셋에서 이상한 값을 자동으로 제거하는 것
- 시스템은 훈련하는 동안 대부분 정상 샘플을 만나 이를 인식하도록 훈련

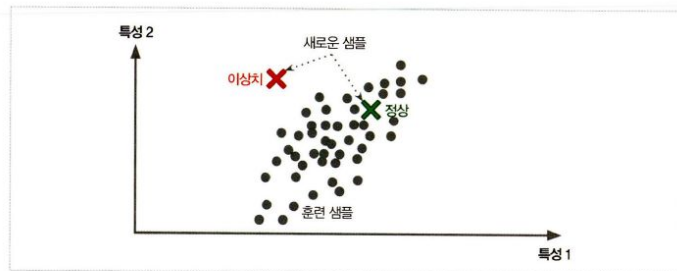


그림 1-10 이상치 탐지

1.4.1 지도 학습과 비지도 학습



비지도 학습

특이치 탐지(novelty detection)

- 훈련 세트에 있는 모든 샘플과 달라 보이는 새로운 샘플을 탐지하는 것이 목적

연관 규칙 학습(association rule learning)

- 학습 알고리즘에 주입하기 전에 데이터셋에서 이상한 값을 자동으로 제거하는 것
- 시스템은 훈련하는 동안 대부분 정상 샘플을 만나 이를 인식하도록 훈련

1.4.1 지도 학습과 비지도 학습

준지도 학습(semi-supervised learning)

일부만 레이블이 있는 데이터

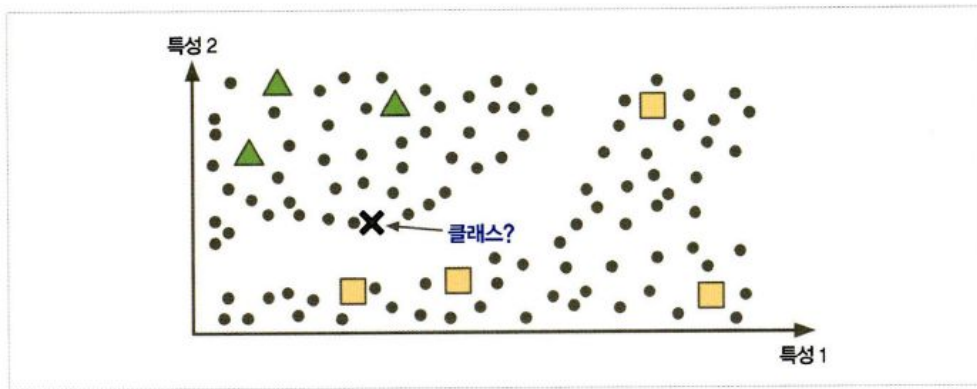


그림 1-11 두 개의 클래스(삼각형과 사각형)를 사용한 준지도 학습: 새로운 샘플(곱셈 기호)이 레이블이 있는 사각형 클래스에 더 가깝지만 레이블이 없는 샘플(원)이 이 샘플을 삼각형 클래스로 분류하는 데 도움을 줍니다.

1.4.1 지도 학습과 비지도 학습

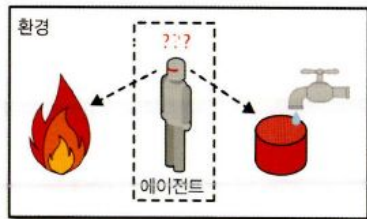


강화 학습(reinforcement learning)

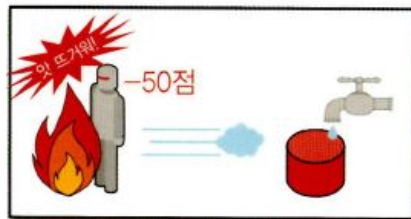
- 에이전트(agent): 학습하는 시스템
- 환경(environment)을 관찰해서 행동(action)을 실행하고 그 결과로 보상(reward) 또는 부정적인 보상에 해당하는 벌점(penalty)을 받습니다.
- 시간이 지나며 가장 큰 보상을 얻기 위해 스스로 최상의 전략인 정책(policy)을 학습함

1.4.1 지도 학습과 비지도 학습

강화 학습



- 1 관찰
- 2 정책에 따라 행동을 선택



- 3 행동 실행!
- 4 보상이나 벌점을 받음



- 5 정책 수정(학습 단계)
- 6 최적의 정책을 찾을 때까지 반복

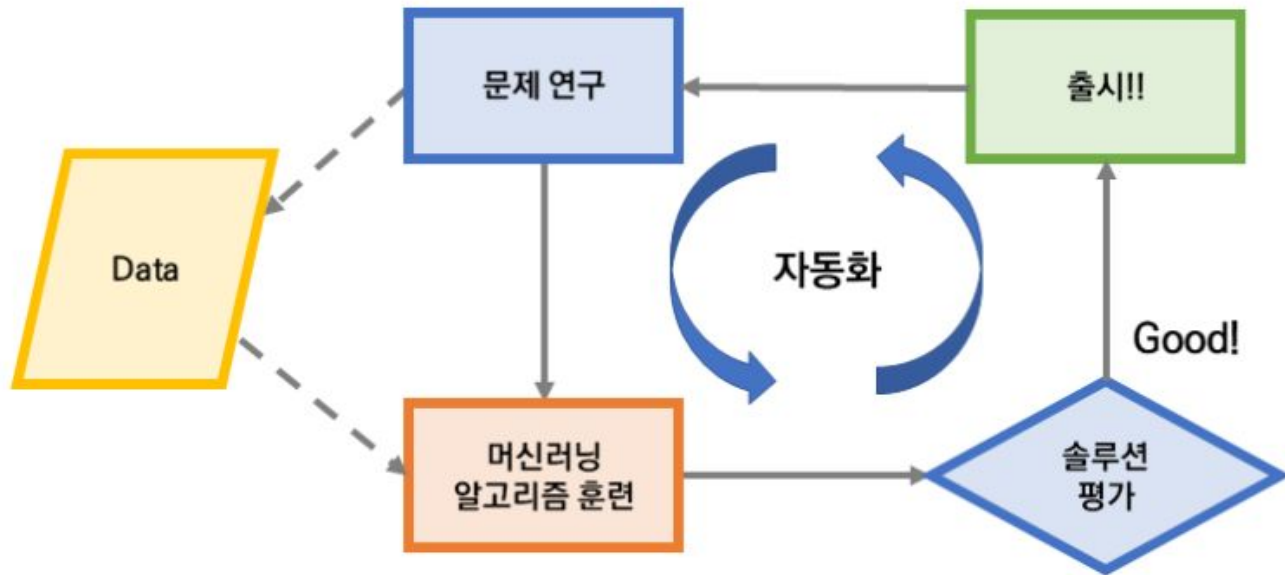
1.4.2 배치 학습과 온라인 학습

배치 학습(batch learning)

- 한 번에 모든 훈련 데이터를 학습 시키는 방법
- 시간과 자원을 많이 소모하여 오프라인 환경에서 수행함
- 오프라인 학습(**offline learning**)이라고도 한다.
- 학습은 런칭 전에만 진행하고, 제품에 학습된 내용을 적용하면 더 이상의 학습 없이 사용된다.
- 새로운 데이터가 등장해서 머신을 재학습 하고자 하는 경우, 이전 데이터에 새로운 데이터를 포함한 전체 데이터를 학습시키고, 학습된 새로운 모델을 사용해야 함

1.4.2 배치 학습과 온라인 학습

배치 학습(batch learning)



1.4.2 배치 학습과 온라인 학습



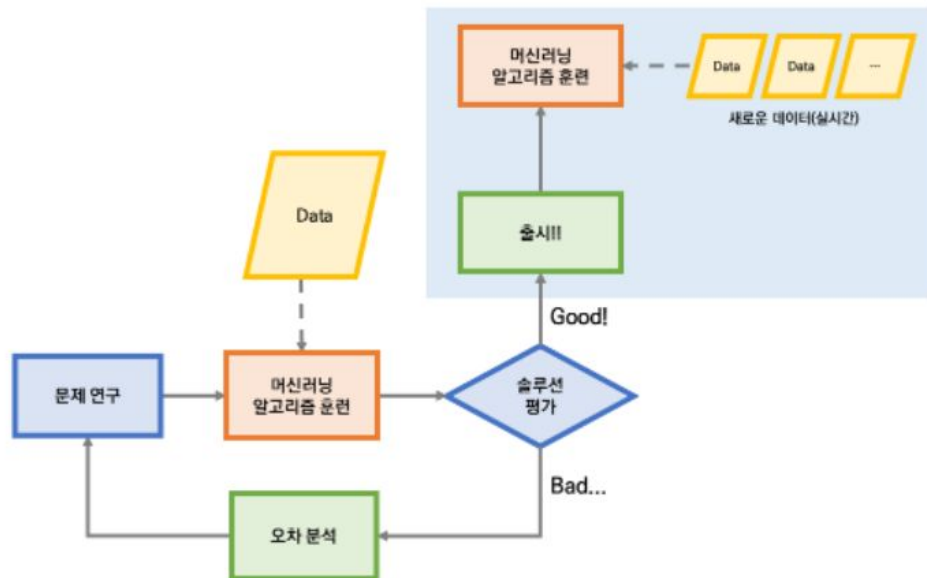
온라인 학습(Online learning)

데이터를 순차적으로 한 개씩 또는 미니배치(mini-batch)라 부르는 작은 묶음 단위로 주입하여 시스템을 훈련 시킨다.

- 미니 배치의 크기가 작기 때문에 학습 단계가 빠르고 비용이 적게 들어 모델은 데이터가 도착하는 대로 즉시 학습을 할 수 있다.
- 점진적 학습(Incremental learning)이라고도 함
- 학습률(learning rate)이 중요한 파라미터이다.
 - 학습률이 높으면 데이터에 빠르게 적응
 - 학습률이 낮으면 새로운 데이터에 있는 잡음이나 대표성 없는 데이터 포인트에 덜 민감해짐
- 이전 데이터는 더 필요하지 않으므로 저장 공간을 아낄 수 있음

1.4.2 배치 학습과 온라인 학습

온라인 학습(Online learning)



1.4.3 사례 기반 학습과 모델 기반 학습



일반화(generalize)

머신러닝 작업은 **예측**을 만드는 것

주어진 훈련 데이터로 학습하고 훈련 데이터에서 본 적 없는

새로운 데이터에서 좋은 예측을 만드는 것이 일반화의 의미이다.

훈련 데이터에서 높은 성능을 내는 것이 중요한 것이 아니라

새로운 샘플에 잘 작동하는 모델을 목표로 합니다.

1.4.3 사례 기반 학습과 모델 기반 학습



사례 기반 학습

- 시스템이 훈련 샘플을 기억함으로써 학습
- 유사도(similarity) 측정을 통해 새로운 데이터와 학습한 샘플을 비교하는 식으로 일반화

1.4.3 사례 기반 학습과 모델 기반 학습

사례 기반 학습

- 아래 그림에서 새로운 샘플은 가장 비슷한 샘플 중 다수가 삼각형이므로 클래스로 분류될 것임

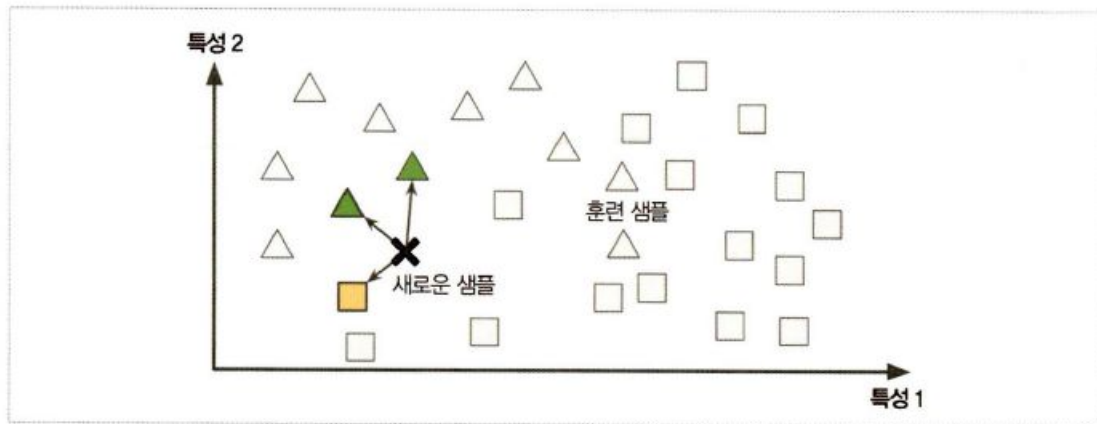


그림 1-15 사례 기반 학습

1.4.3 사례 기반 학습과 모델 기반 학습

모델 기반 학습

샘플들의 모델을 만들어 예측(prediction)에 사용하는 것을 모델 기반 학습(model-based learning)이라고 한다.

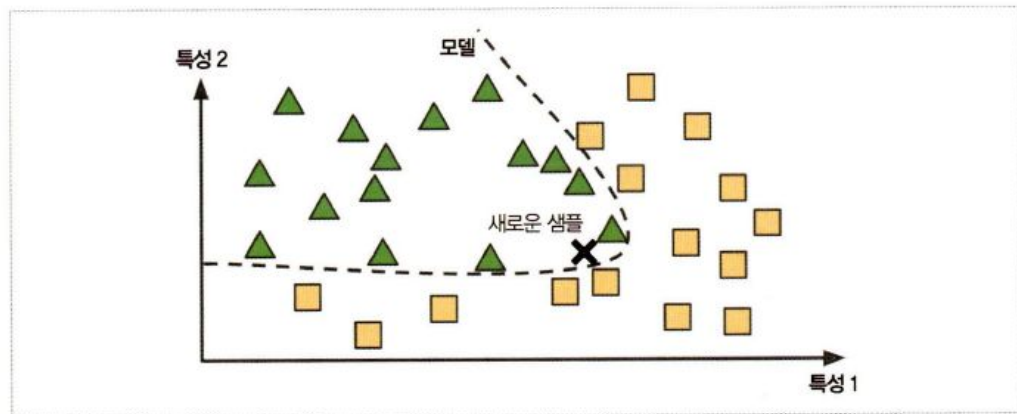


그림 1-16 모델 기반 학습

1.4.3 사례 기반 학습과 모델 기반 학습

모델 기반 학습 예시

- 삶의 만족도는 국가의 1인당 GDP가 올라갈수록 선형으로 올라가는 경향을 볼 수 있음

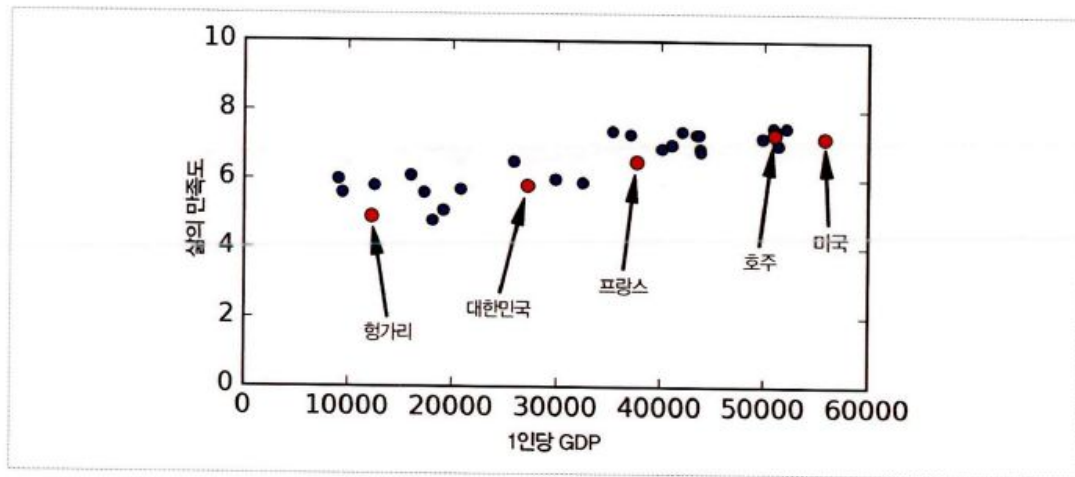


그림 1-17 어떤 경향이 보이나요?

1.4.3 사례 기반 학습과 모델 기반 학습

모델 기반 학습 예시

- 1인당 GDP의 선형 함수로 삶의 만족도를 모델링
 - 이 단계를 모델 선택(model selection)이라고 함
 - 1인당 GDP라는 feature 하나를 가진

삶의 만족도에 대한 선형 모델(linear model)이다.

- 삶의 만족도 = $\theta_0 + \theta_1 * GDP$
- 이 모델은 두 개의 모델 파라미터 θ_0, θ_1 를 가짐

1.4.3 사례 기반 학습과 모델 기반 학습

모델 기반 학습 예시

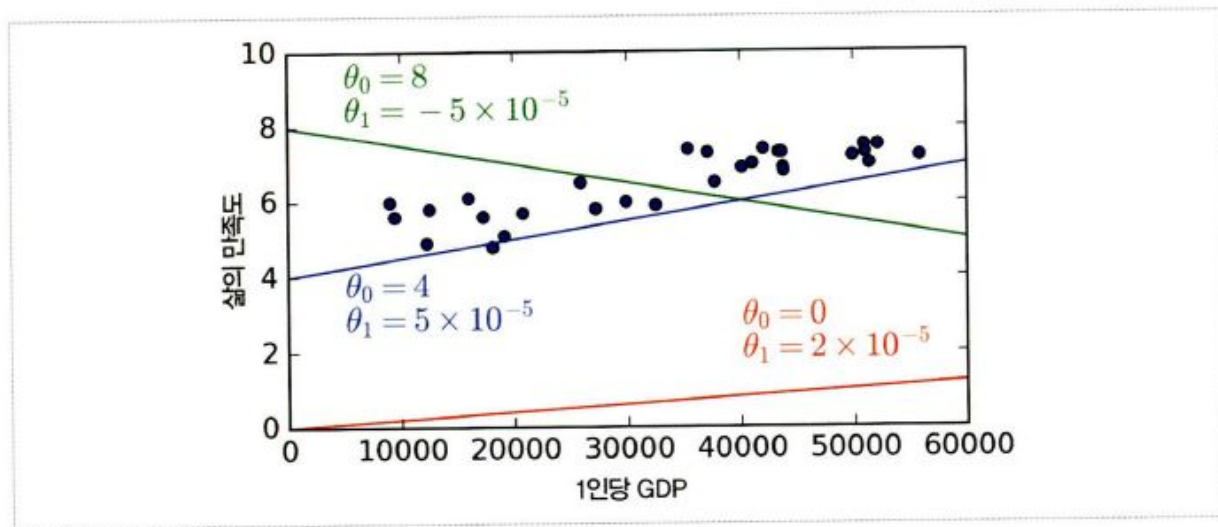


그림 1-18 가능한 몇 개의 선형 모델

1.4.3 사례 기반 학습과 모델 기반 학습



모델 기반 학습 예시

- 효용 함수(utility function): 모델이 얼마나 좋은지 측정하는 함수
- 비용 함수(cost function): 모델이 얼마나 나쁜지 측정하는 함수
 - 선형 회귀에서는 선형 모델의 예측과 훈련 데이터 사이의 거리를 재는 비용 함수를 사용하고, 이를 최소화하는 것이 목적

1.4.3 사례 기반 학습과 모델 기반 학습



모델 학습 작업 요약

- 데이터를 분석합니다
- 모델을 선택합니다
- 훈련 데이터로 모델을 훈련시킵니다.

(즉, 학습 알고리즘이 비용 함수를 최소화하는 모델 파라미터를 찾습니다.)

- 마지막으로 새로운 데이터에 모델을 적용해 예측을 하고, 일반화 되게 한다.


1.4.3 사례 기반 학습과 모델 기반 학습



모델 기반 학습 예시

- 효용 함수(utility function): 모델이 얼마나 좋은지 측정하는 함수
- 비용 함수(cost function): 모델이 얼마나 나쁜지 측정하는 함수
 - 선형 회귀에서는 선형 모델의 예측과 훈련 데이터 사이의 거리를 재는 비용 함수를 사용하고, 이를 최소화하는 것이 목적

1.5 머신러닝의 주요 도전 과제



1.5.1 충분하지 않은 양의 훈련 데이터

- 머신러닝 알고리즘이 잘 작동하려면 데이터가 많아야 한다.
- 아주 간단한 문제에서조차도 수천 개의 데이터가 필요하고,
이미지나 음성 인식 같은 복잡한 문제라면 수백만 개가 필요함

1.5 머신러닝의 주요 도전 과제

1.5.1 충분하지 않은 양의 훈련 데이터

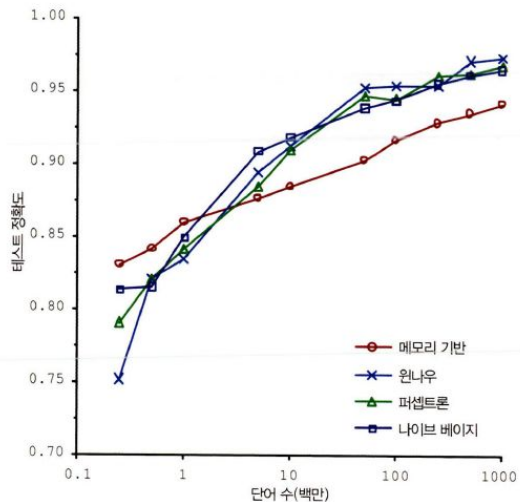



그림 1-20 알고리즘 대비 데이터의 중요성¹⁹⁾


1.5 머신러닝의 주요 도전 과제



1.5.2 대표성 없는 훈련 데이터

- 일반화가 잘 되려면 일반화하기 원하는 새로운 사례를 훈련 데이터가 잘 대표하는 것이 중요
- 샘플이 작으면 샘플링 잡음(sampling noise)이 생기고, 매우 큰 샘플도 표본 추출 방법이 잘못되면 대표성을 띠지 못함
 - 이를 샘플링 편향(sampling bias)라고 함


1.5 머신러닝의 주요 도전 과제



1.5.3 낮은 품질의 데이터

- 훈련 데이터가 error, outlier, noise로 가득하다면 머신러닝 시스템이 내재된 패턴을 찾기 어려워 잘 작동하지 않을 것
- 훈련 데이터 정제에 시간을 투자해야 함


1.5 머신러닝의 주요 도전 과제



1.5.4 관련 없는 특성

- 훈련 데이터에 관련 없는 특성이 적고 관련 있는 특성이 충분해야 시스템이 학습할 수 있다.
- 머신러닝 프로젝트의 핵심 요소는 **훈련에 사용할 좋은 특성을 찾는** 것이다.
 - 이 과정을 특성 공학(feature engineering)이라 함

1.5 머신러닝의 주요 도전 과제



1.5.4 관련 없는 특성

- 특성 선택(feature selection): 가지고 있는 특성 중에서 훈련에 가장 유용한 특성 선택
- 특성 추출(feature extraction): 특성을 결합하여 더 유용한 특성을 만듦. 차원 축소 알고리즘 등이 있다.
- 새로운 데이터를 수집해 새 특성을 만듦

1.5 머신러닝의 주요 도전 과제

1.5.5 훈련 데이터 과대 적합

- 모델이 훈련 데이터에 너무 잘 맞지만 일반성이 떨어진다는 뜻

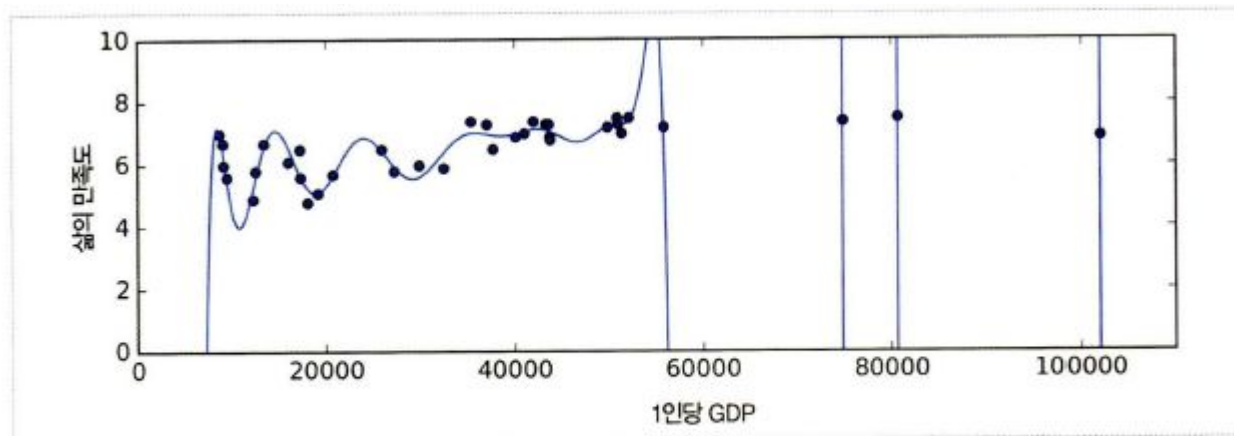



그림 1-22 훈련 데이터에 과대적합


1.5 머신러닝의 주요 도전 과제



1.5.5 훈련 데이터 과대 적합

- 과대적합은 훈련 데이터에 있는 잡음의 양에 비해 모델이 너무 복잡할 때 일어남
- 파라미터 수가 적은 모델을 선택하거나 (예를 들면 고차원 다항 모델보다 선형모델), 훈련 데이터에 있는 특성 수를 줄이거나, 모델에 제약을 가하여 단순화시킨다.
- 훈련 데이터를 더 많이 모은다.
- 훈련 데이터의 잡음을 줄인다(예를 들면 오류 데이터 수정과 이상치 제거)

1.5 머신러닝의 주요 도전 과제



1.5.5 훈련 데이터 과대 적합

- 규제(regularization):


모델을 단순화하기 위해 모델에 제약을 가하는 것

- 자유도(degree of freedom) ? - p.60

자유도란 어떤 제약으로 인해 그 제약을 제외한 자유 의지를 가질 수 있는
남은 개수

최소한의 독립된 변수의 수

1.5 머신러닝의 주요 도전 과제



1.5.6 훈련 데이터 과소적합

- 과소적합(underfitting)은 모델이 너무 단순해서 데이터의 내재된 구조를 학습하지 못할 때 일어남
 - 모델 파라미터가 더 많은 강력한 모델 선택
 - 학습 알고리즘에 더 좋은 특성을 제공(특성 공학)
 - 모델의 제약을 줄임(규제 하이퍼파라미터 감소)



1.5.7 한걸음 물러서서

1. 머신러닝? 기계가 데이터로부터 학습하여 어떤 작업을 더 잘하도록 만드는 것
2. 지도 학습, 비지도 학습, 배치 학습, 온라인 학습, 사례 기반 학습, 모델 기반 학습
3. 학습 알고리즘이 모델 기반이면 훈련 세트에 모델을 맞추기 위해 모델 파라미터를 조정, 새로운 데이터에서도 좋은 예측을 만들 거라 기대
4. 사례 기반이면 샘플을 기억하는 것이 학습이고 측정을 사용해 학습 샘플과 새로운 샘플을 비교하는 식으로 새로운 샘플에 일반화한다.
5. 훈련 세트에 따른 결과
6. 과대적합, 과소적합



1.6 테스트와 검증

- 훈련 데이터를 훈련 세트와 테스트 세트 두 개로 나눔
- 일반화 오차(**generalization error**): 새로운 샘플에 대한 오류 비율
- 테스트 세트에서 모델을 평가함으로써 이 오차에 대한 추정값 (**estimation**)을 얻음
- 이 값은 새로운 샘플에 모델이 얼마나 잘 작동할지 알려줌
- 훈련 오차가 낮지만 일반화 오차가 높다면 이는 모델이 훈련 데이터에 과대적합되었다는 뜻



1.6.1 하이퍼파라미터 튜닝과 모델 선택

- 모델 평가는 테스트 세트를 사용
- 선형 모델과 다항 모델이 있는 경우, 훈련 세트로 훈련하고 테스트 세트를 사용해 얼마나 잘 일반화 했는지 비교



1.6.1 홀드아웃 검증(holdout validation)

- 훈련 세트의 일부를 떼어내어 여러 후보 모델을 평가하고 가장 좋은 하나를 선택
- 홀드아웃 세트를 **검증 세트(validation set)**이라 부름
- 훈련 세트에서 다양한 하이퍼파라미터 값을 가진 여러 모델을 훈련하고, 가장 높은 성능을 내는 모델 선택
- 검증 과정이 끝나면 최선의 모델을 전체 훈련 세트에서 다시 훈련하여 최종 모델을 만듦
- 최종 모델을 테스트 세트에서 평가하여 일반화 오차를 추정함



1.6.1 교차 검증(cross validation)

- 작은 검증 세트를 여러 개 사용해 반복적인 교차 검증(cross-validation)을 수행하는 것
- 검증 세트마다 나머지 데이터에서 훈련한 모델을 해당 검증 세트에서 평가
- 모델의 평가를 평균하면 훨씬 더 정확한 성능을 측정할 수 있음
- 단점은 훈련 시간이 검증 세트 개수에 비해 늘어남
- 검증 세트 개수를 k 개로 지정한 교차 검증을 **K-fold Cross Validation**이라 함



1.6.1 데이터 불일치

훈련-개발 세트

- 검증 세트와 테스트 세트가 실전에서 기대하는 데이터를 가능한 잘 대표해야 함
- 훈련 세트 중 일부를 떼어내서 또 다른 세트를 만드는 것
- 모델을 훈련 세트에서 훈련 시킨다음, 훈련-개발 세트에서 평가한다.
- 이 모델이 검증 세트에서 좋지 않은 성능을 보인다면, 데이터 불일치로 인해 발생한 문제이다.
- 즉, 훈련-개발 세트와 검증 세트의 오차 차이가 크다면, 데이터 분포의 차이 문제이다.



1.7 연습 문제

p. 65~66 참고