



Hands-On Machine Learning with Scikit-Learn & TensorFlow

2018. 09. 11

발표자 : 조 동 훈

1

머신러닝이란?

2

왜 머신러닝을 사용하는가?

3

머신러닝 시스템의 종류

4

머신러닝의 주요 도전 과제

5

테스트와 검증

1. 한눈에 보는 머신러닝

1.1 머신러닝이란?

데이터로 부터 학습하도록 컴퓨터를 프로그래밍하는 과학(또는 예술)이다

기계 학습(機械學習) 또는 머신 러닝(영어: machine learning)은 인공 지능의 한 분야로, 컴퓨터가 학습할 수 있도록 하는 알고리즘과 기술을 개발하는 분야를 말한다. 가령, 기계 학습을 통해서 수신한 이메일이 스팸인지 아닌지를 구분할 수 있도록 훈련할 수 있다.

<< 위키백과 : 기계학습 >>

1959년, 아서 사무엘은 기계 학습을 "기계가 일일이 코드로 명시하지 않은 동작을 데이터로부터 학습하여 실행할 수 있도록 하는 알고리즘을 개발하는 연구 분야"라고 정의하였다.

<< 위키백과 : 기계학습 정의 >>

어떤 **작업 T**에 대한 컴퓨터 프로그램의 **성능을 P**로 측정했을 때 **경험 E**로 인해 성능이 향상됐다면, 이 컴퓨터 프로그램은 작업 T와 성능 측정 P에 대해 E로 학습한 것이다.

<<핸즈온 머신러닝 : _토미첼, 1997 >>

구분	작업 T	경험 E	성능측정 P
스팸 메일 시스템	새로운 메일이 스팸인지 구분	훈련데이터	정확도

시스템이 학습하는 사용하는 샘플 : 훈련세트(Training Set)

각 훈련데이터 : 훈련 사례 (Training instance, 샘플)

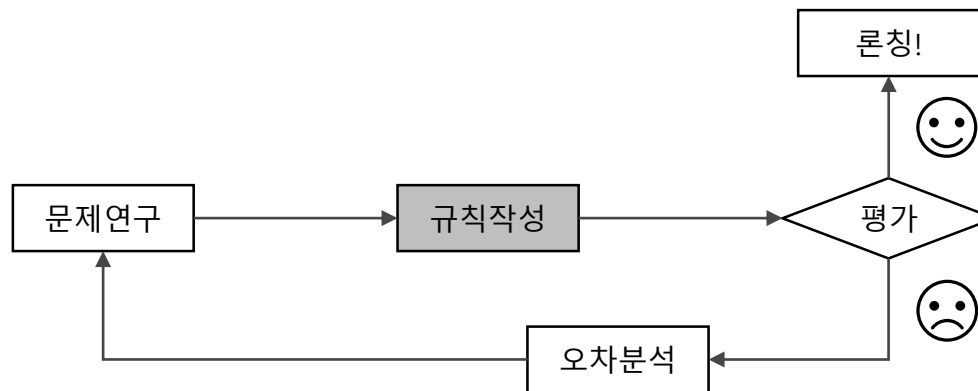
1. 한눈에 보는 머신러닝

1.2 왜 머신러닝을 사용하는가?

- 기존 솔루션으로는 많은 수동 조정과 규칙이 필요한 문제
- 전통적인 방법으로는 전혀 해결 방법이 없는 복잡한 문제
- 유동적인 환경에 적응하기 어려운 문제
- 대량의 데이터와 복잡한 문제들로 해결하기 어려운 문제

■ 전통적인 접근 방법

문제가 단순하지 않아 규칙이 점점 길고 복잡해지므로 유지 보수가 매우 힘들

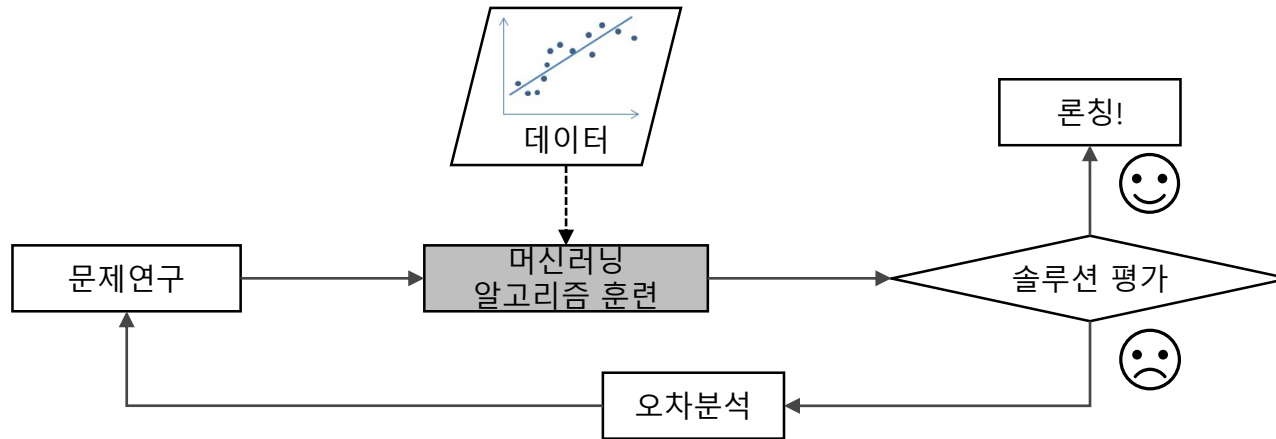


규칙(4U, 신용카드, 무료, 대출, 광고, 대행 등)을 분석, 패턴을 감지하는 알고리즘 작성 후 테스트/적용

1. 한눈에 보는 머신러닝

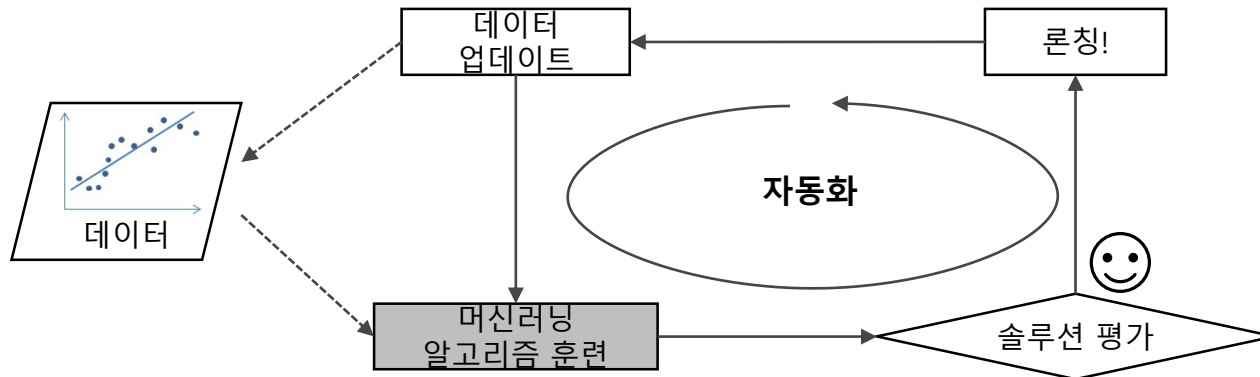
■ 머신러닝 접근 방법

문제에 대한 패턴을 인지하고 학습하여 프로그램이 짧아지고 유지보수가 쉬우며 정확도를 높임



■ 자동으로 변화에 적응함

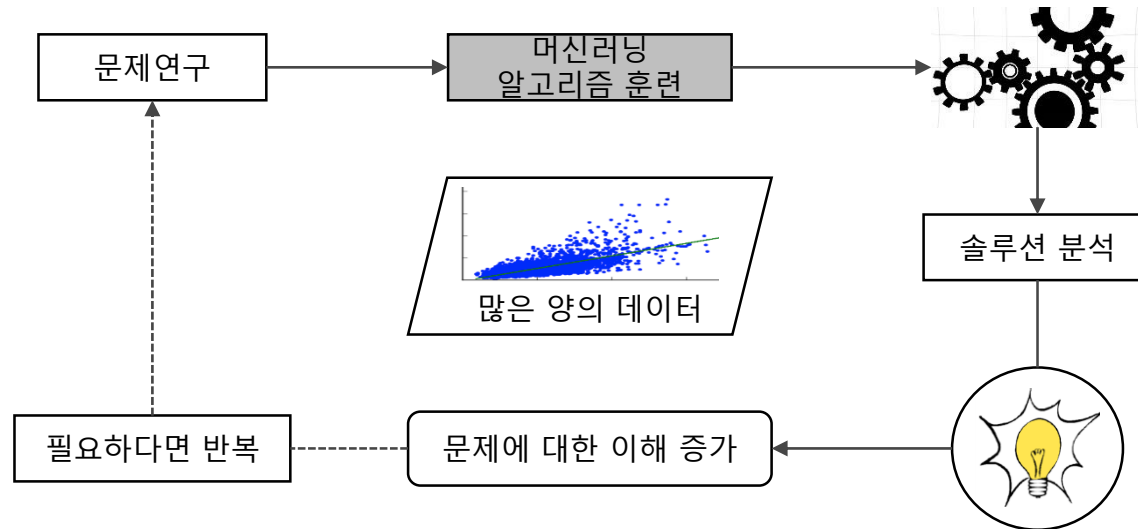
사용자가 지정한 데이터에서 특정 패턴을 자동으로 인식하고 별도의 작업이 없어도 분류



1. 한눈에 보는 머신러닝

■ 머신러닝을 통해 배우기

대용량의 데이터를 분석하면 겉으로는 보이지 않던 패턴을 발견 → 데이터 마이닝(Data Mining)



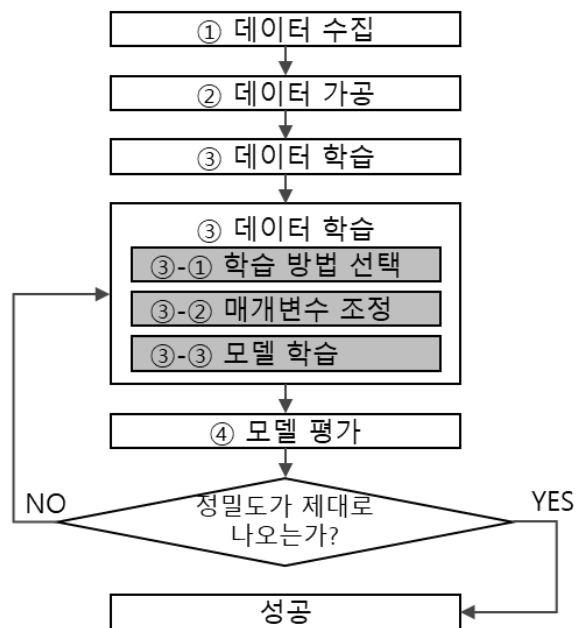
1. 한눈에 보는 머신러닝

1.3 머신러닝 시스템의 종류

머신러닝 시스템의 종류는 굉장히 많으며 아래와 같이 크게 3가지로 분류할 수 있음

- 사람의 감독하에 훈련하는 것인지 그렇지 않은 것인지 (지도학습과 비지도 학습)
- 실시간으로 점진적인 학습을 하는지 아닌지 (온라인 학습과 배치 학습)
- 단순 알고 있는 데이터 포인트와 새 데이터 포인트를 비교하는지 아니면 훈련 데이터셋에서 과학자들처럼 패턴을 예측하여 예측 모델을 만드는지 (사례 기반 학습과 모델 기반 학습)

● 머신러닝의 과정



③-① 학습방법선택 : 알고리즘 선택 (SVM, K-means 등)

③-② 매개변수 조정 : 데이터와 알고리즘에 맞게 변수 조정

④ 모델평가 : 테스트 데이터 활용해 정밀도 확인

정밀도가 나오지 않으면 매개변수 수정 또는 알고리즘 변경
이후 반복 수행

1. 한눈에 보는 머신러닝

1.3.1 지도 학습과 비지도 학습

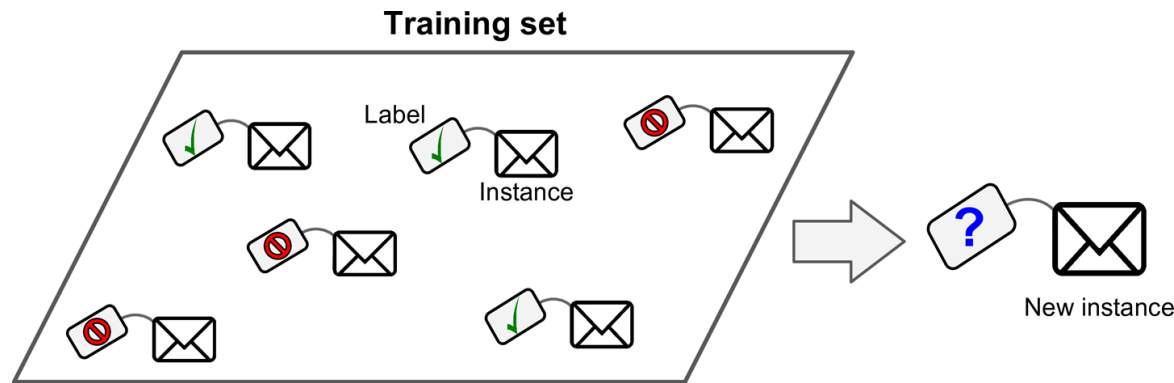
학습하는 동안의 감독 형태나 정보량에 따라 지도학습, 비지도학습, 준지도학습, 강화학습로 구분

종류	내용	알고리즘
지도 학습	데이터와 정답을 함께 입력	<ul style="list-style-type: none"> • K-최근접 이웃 • 선형회귀 • 로지스틱 회귀 • 서포트 벡터 머신 • 결정트리와 랜덤 포레스트 • 신경망
	다른 데이터의 정답을 예측	
비지도 학습	데이터는 입력하지만 정답은 미입력	<ul style="list-style-type: none"> • 군집 • 시각화와 차원 축소 • 연관 규칙 학습
	다른 데이터의 규칙성 찾을	
강화 학습	부분적으로 정답을 입력	
	데이터를 기반으로 최적의 정답을 찾을	

1. 한눈에 보는 머신러닝

■ 지도 학습

알고리즘에 주입하는 훈련데이터에 **레이블(LABEL)**이라는 정답이 포함



지도 학습의 주요 작업은 **분류(Classfication)**이며 예를 들어 스팸필터 작업 (많은 양의 메일, 스팸 유무)

예제 1) 중고차 가격 (타겟)

- **예측 변수 (predictor variable)** 또는 **특성 (feature)**을 사용해 타겟을 예측하는 것
- Feature : 주행거리, 연식, 브랜드, 사고유무 등
- Feature와 Label (중고차 가격)이 포함된 많은 데이터가 필요
- 이러한 작업을 **회귀(Regression)**이라 함

1. 한눈에 보는 머신러닝

■ 비지도 학습

훈련데이터에 레이블이 없는 것을 말하며, 최종적으로 내야하는 답이 정해져 있지 않는 것이 특징
대표적인 비지도 학습 알고리즘은 아래와 같음

■ 군집 (Clustering)

- K-평균 알고리즘 K-means
- 위계적 군집 분석 Hierarchical Cluster Analysis (HCA)
- 기대치 최대화 알고리즘 Expectation Maximization

■ 시각화 및 차원 축소법 (Visualization and Dimensionality Reduction)

- 주성분(최적치) 알고리즘 Principal Component Algorithm (PCA)
- 커널 기법 주성분(최적치) 알고리즘 Kernel PCA
- 지역적 선형 위상 배치법 Locally-Linear Embedding
- T개의 분산 기반 확률적 근접 위상 배치법 T-distributed Stochastic Neighbor Embedding (t-SNE)

■ 연관 규칙 학습 (Association rule learning)

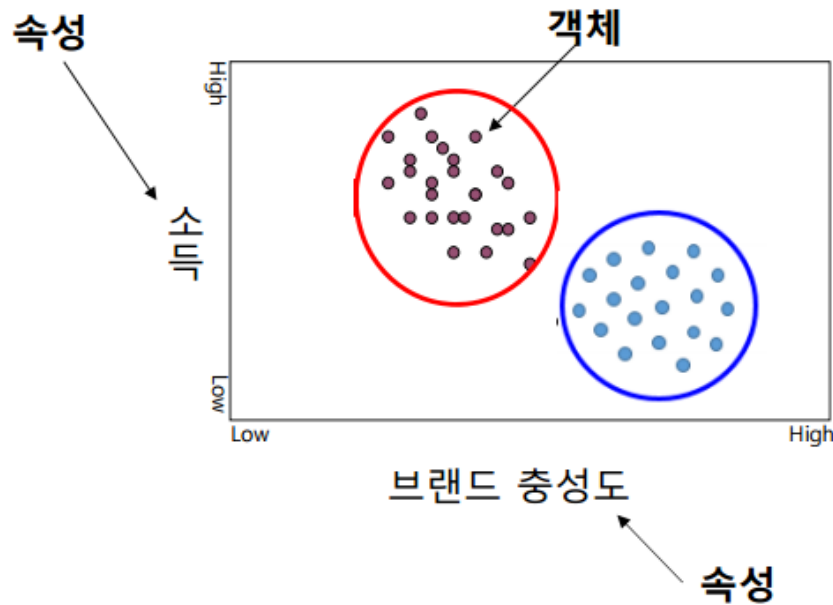
- 선형(아프리오리) Apriori
- 화려함(에플라) Eclat

1. 한눈에 보는 머신러닝

■ 비지도 학습

- 군집 (Clustering) : 유사한 속성을 객체들을 군집(Cluster)으로 나누거나 묶어주는 데이터마이닝 기법

예제) 고객들의 구매패턴을 반영하는 속성들에 관한 데이터가 수집된다고 할 때

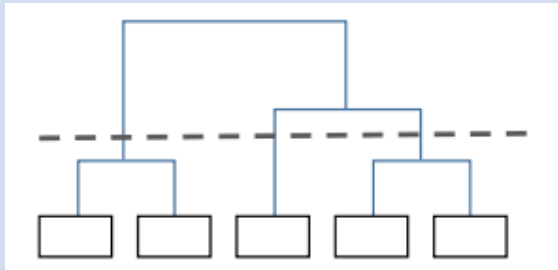
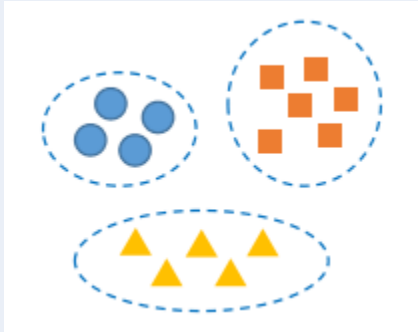


군집 분석을 통해 유사한 구매패턴을 보이는
고객들을 군집화하고 판매전략을 도출

1. 한눈에 보는 머신러닝

■ 비지도 학습

- 군집 분석의 방법은 '계층적 방법'과 '비계층적 방법'으로 구분

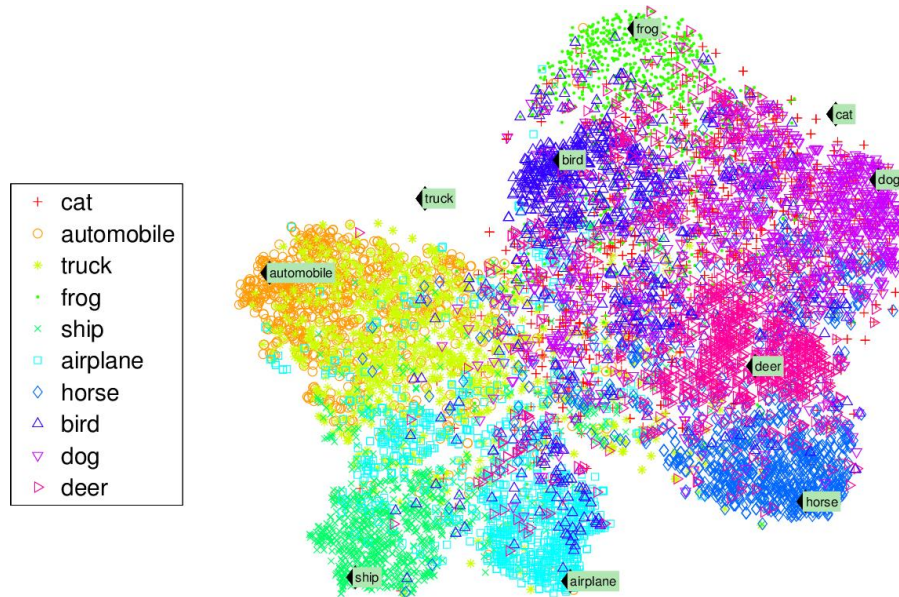
종류	내용	도식화
비계층적 군집 (Non-hierarchical Clustering)	사전에 군집 수 k 를 정한 후 입력 데이터를 k 개 중 하나의 군집에 배정	
계층적 군집 (Hierarchical Clustering)	사전에 군집 수 k 를 정하지 않고 단계적으로 군집 트리를 제공	

1. 한눈에 보는 머신러닝

■ 비지도 학습

● 시각화와 차원 축소 (Visualization and Dimensionality Reduction)

- 레이블이 없는 대규모의 고차원 데이터를 2D나 3D로 표현함
- 데이터가 어떻게 조직되어 있는지 이해할 수 있고 예상 못한 패턴 발견 가능



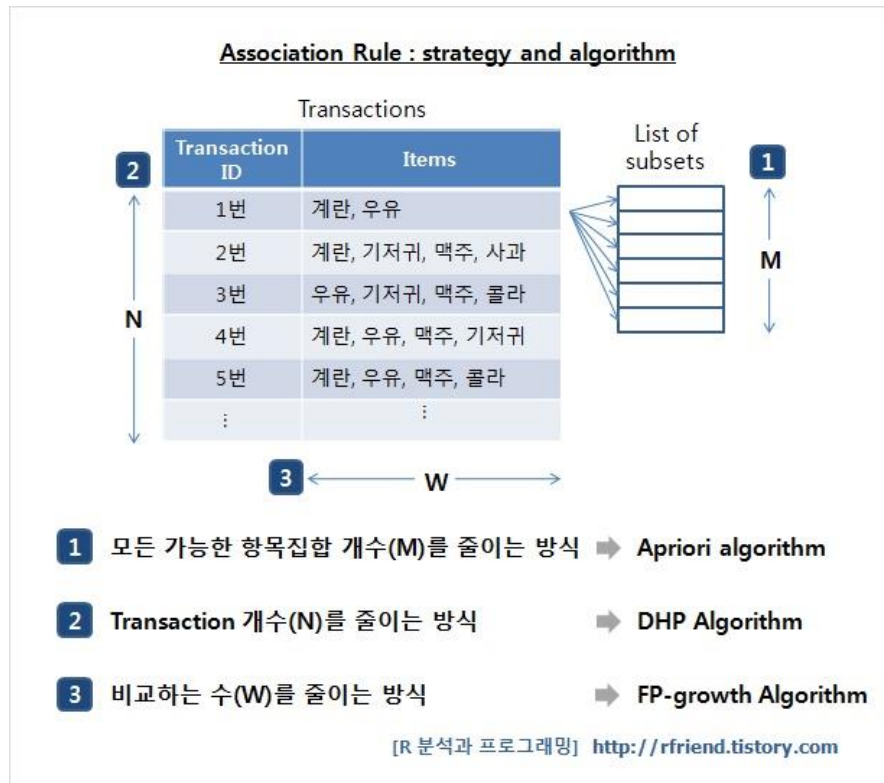
- 차원 축소(Dimensionality Reduction)는 너무 많은 정보를 잃지 않으면서 데이터를 간소화하며, 하나의 특성으로 합침 → 특성 추출(Feature Extraction)

1. 한눈에 보는 머신러닝

■ 비지도 학습

● 시각화와 차원 축소 (Visualization and Dimensionality Reduction)

- 연관 규칙 학습(Association rule learning) : 동시 발생의 규칙을 이용해 특성 간의 관계 탐구로 데이터의 패턴을 분석
- 연관 규칙 분석 : 군집 분석 이후에 각 그룹의 특성을 분석하기 위함
- 예제) 바비큐 소스와 감자를 구매한 사람 → 스테이크 구매



1. 한눈에 보는 머신러닝

■ 준지도 학습

보통 레이블이 없는 데이터가 많고 레이블이 있는 데이터는 아주 조금인 경우를 말함

- 사용 이유 : 목표 값을 포함한 데이터를 얻기 위해서는 비용이 감당할 수 없을 경우 사용
- 최초로 제안된 준지도 학습의 기술 방법 : GAN(Generative Adversarial Networks)를 활용한 방법
GAN은 대립하는(Adversarial) 두 개의 신경망(Networks)을 활용하여 이미지를 생성(Generative)하는 기술로 최신 딥러닝 기술 중 하나
- GAN을 활용한 예제 1

입력으로 '일정한 길이의 숫자 벡터'를 받아 새로운 이미지를 생성하는 GAN의 경우 (128*128 사이즈 이미지)



- GAN을 활용한 예제 2

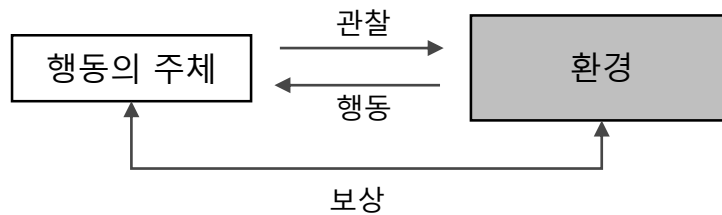
입력으로 '특정 이미지'를 받아 실제 핸드백 이미지와 신발 이미지를 학습 후, 새로운 '실제 핸드백 이미지' 생성



1. 한눈에 보는 머신러닝

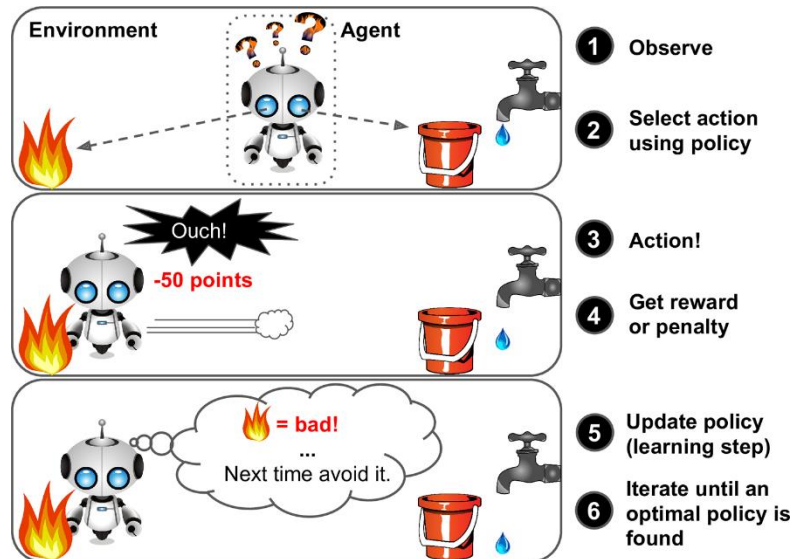
■ 강화 학습

- 현재 상태를 관찰해서 어떻게 대응해야 할지와 관련된 문제를 다룬
- 행동의 주체, 환경(상황 또는 상태), 보상 등으로 구성되어 있음.



- 행동의 주체 : 고양이
- 환경 : 자동으로 먹이주는 기계

- 학습하는 시스템을 에이전트라고 하며, 정책에 따라 보상과 벌점을 받음



1. 한눈에 보는 머신러닝

1.3.2 배치 학습과 온라인 학습

머신러닝 시스템을 분류하는데 다른 기준은 입력 데이터의 스트림에서 점진적인 학습 가능 여부

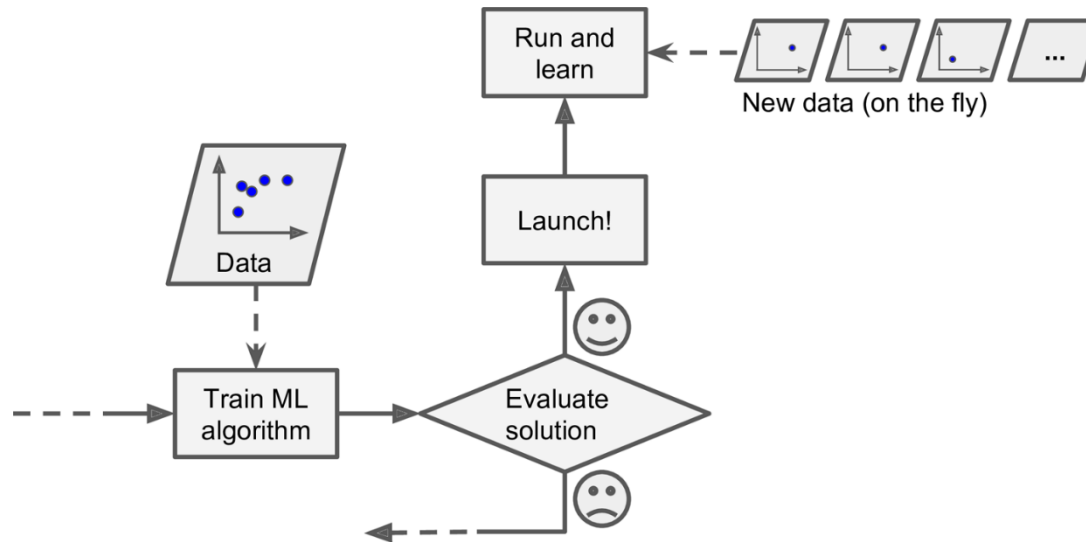
■ 배치 학습 (Batch Learning)

- 가용한 데이터를 모두 사용하여 훈련하여 시간과 자원을 많이 소모하는 것이 특징
- 먼저 시스템을 훈련시키고 제품에 적용하여 더 이상의 학습 없이 실행
- 즉 학습한 것을 단지 적용하여 이를 오프라인 학습(Offline Learning)이라 함
- 문제점 :
 - ① 새로운 데이터를 학습하려면 전체 데이터를 사용하여 처음부터 다시 훈련해야 함
 - ② 전체 데이터를 사용하여 훈련하는데 많은 시간이 소요될 수 있음
 - ③ 시스템 자원을 많이 소모
 - ④ 자원이 제한된 시스템에서 많은 양의 훈련데이터를 나르고 학습을 위해 자원 사용하는 경우

1. 한눈에 보는 머신러닝

■ 온라인 학습 (Online Learning)

- 데이터를 순차적으로 한 개씩 또는 미니배치(Mini-Batch)라 부르는 작은 단위로 입력하여 훈련
- 비용이 적게 들어 시스템은 데이터가 도착하는 대로 즉시 학습 수행

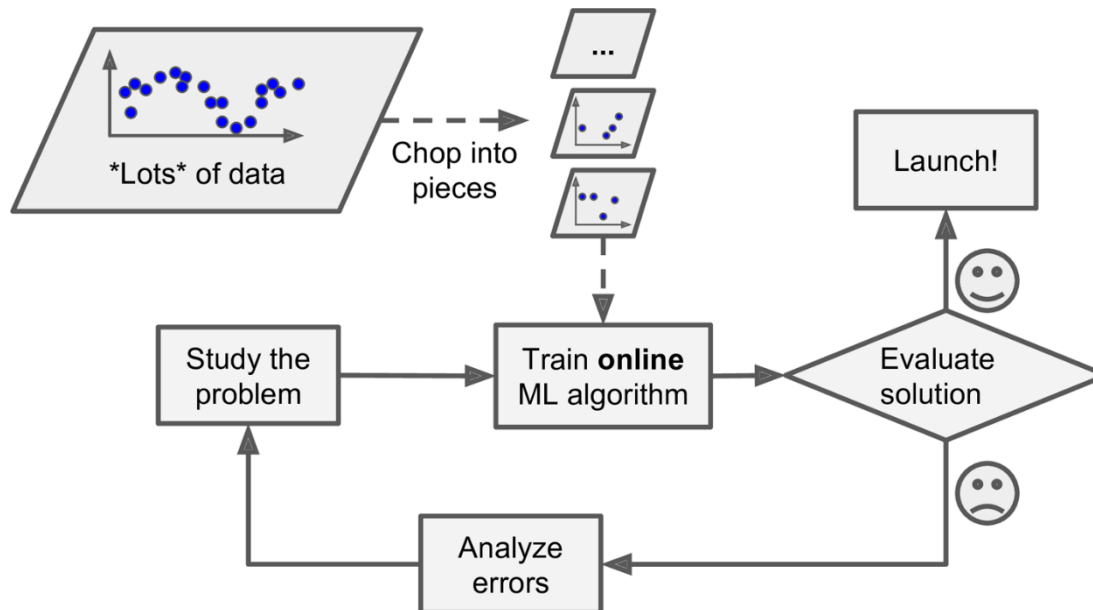


- 연속적으로 데이터를 받고 빠른 변화에 스스로 적응해야 하는 시스템에 적합 (주식 가격)
- 큰 데이터 셋을 학습하는 시스템에도 가능 ➔ 이를 외부 메모리(out-of-core) 학습이라 함

1. 한눈에 보는 머신러닝

■ 온라인 학습 (Online Learning)

- 변화하는 데이터에 얼마나 빠르게 적응할 것인가, 학습률(Learning Rate)로 결정
- 학습률을 높게 하면 데이터에 빠르게 적응하지만, 예전 데이터를 금방 잊음
- 학습률이 낮으면 시스템의 관성으로 더 느리게 학습, 데이터의 잡음에 덜 민감함
- 문제점 : 나쁜 데이터가 주입 되면 시스템 성능이 점진적으로 감소 (면밀한 감시가 필요)



1. 한눈에 보는 머신러닝

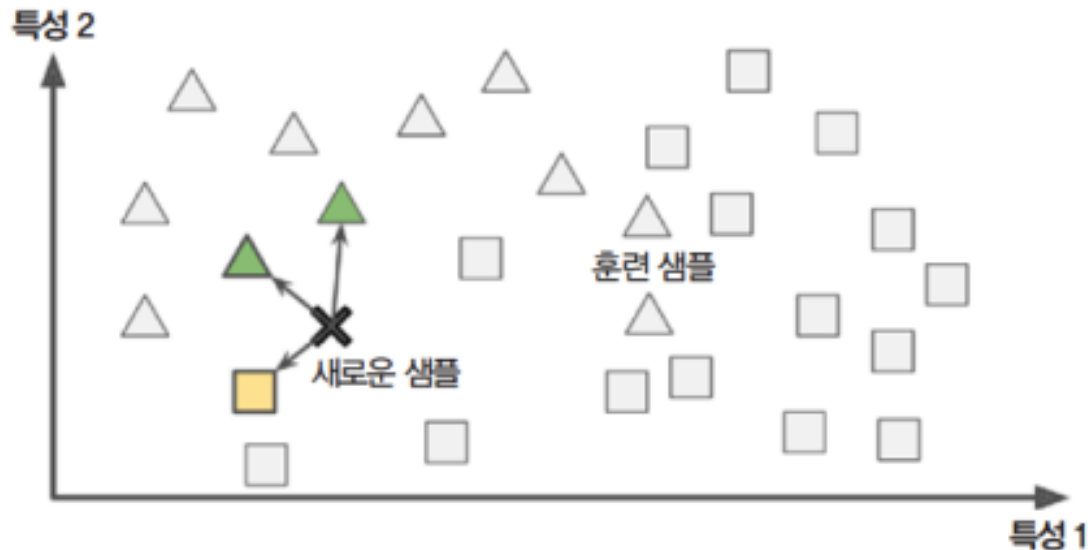
1.3.3 사례기반 학습과 모델기반 학습

머신러닝 시스템은 어떻게 일반화 되는가에 따라 분류할 수 있음

일반화를 위한 두 가지 접근법인 "사례기반 학습"과 "모델기반 학습"이 있음

■ 사례기반 학습 (Instance-Based Learning)

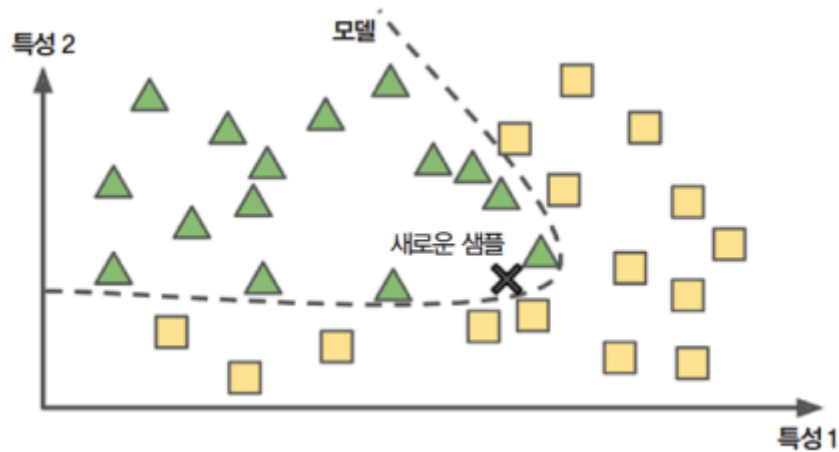
- 시스템이 사례를 기억함으로써 학습하고, 유사도 측정을 통해 새로운 샘플을 일반화 함
- 아래는 저장된 훈련 데이터에서 가장 가까운 샘플을 찾는 과정.(유사도 측정)



1. 한눈에 보는 머신러닝

■ 모델기반 학습 (Model-based Learning)

- 샘플로부터 일반화 시키는 방법 중 모델을 만들어 예측에 사용

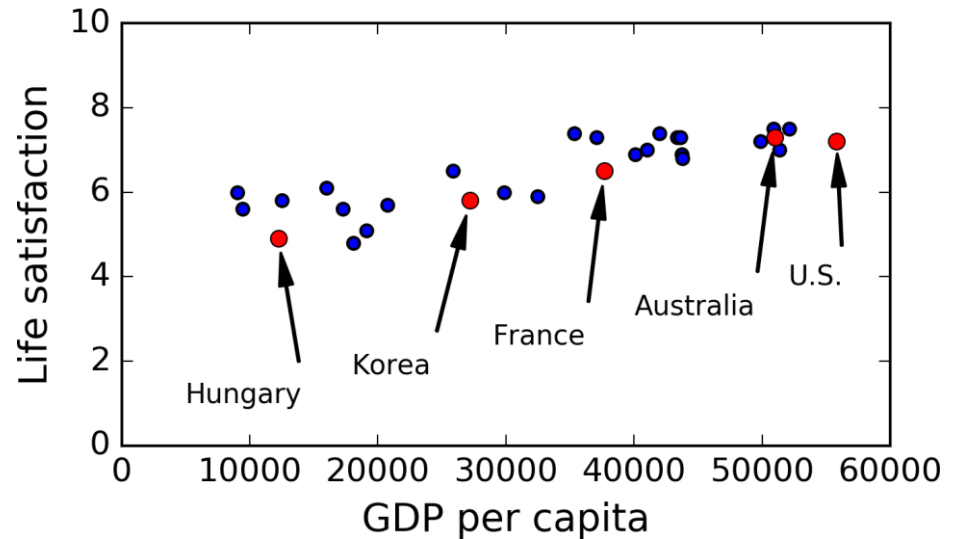


1. 한눈에 보는 머신러닝

■ 모델기반 학습 (Model-based Learning)

- 예제) 1인당 GDP에 대한 삶의 만족도

Country	GDP per Capita (USD)	Life satisfaction
Hungary	12,240	4.9
Korea	27,195	5.8
France	37,675	6.5
Australia	50,962	7.3
United States	55,805	7.2



- 1인당 GDP가 증가할수록 선형으로 같이 올라감 → 선형함수를 표현하며 선형모델을 얻음

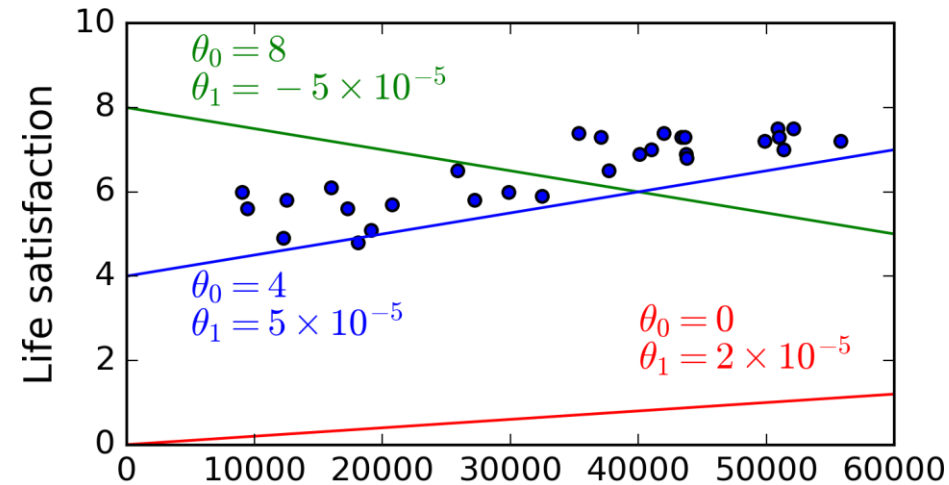
$$\text{삶의만족도} = \theta_0 + \theta_1 \times \text{1인당 GDP}$$

1. 한눈에 보는 머신러닝

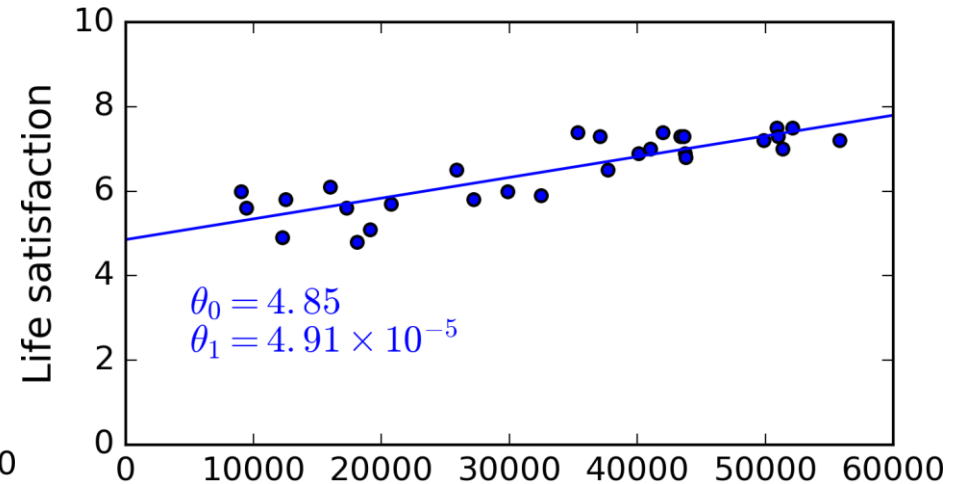
■ 모델기반 학습 (Model-based Learning)

- 삶의 만족도 파라미터를 조절하면 아래와 같은 선형모델을 얻을 수 있다

$$\text{삶의만족도} = \theta_0 + \theta_1 \times \text{인당 GDP}$$



<< 가능한 몇 개의 선형 모델 >>

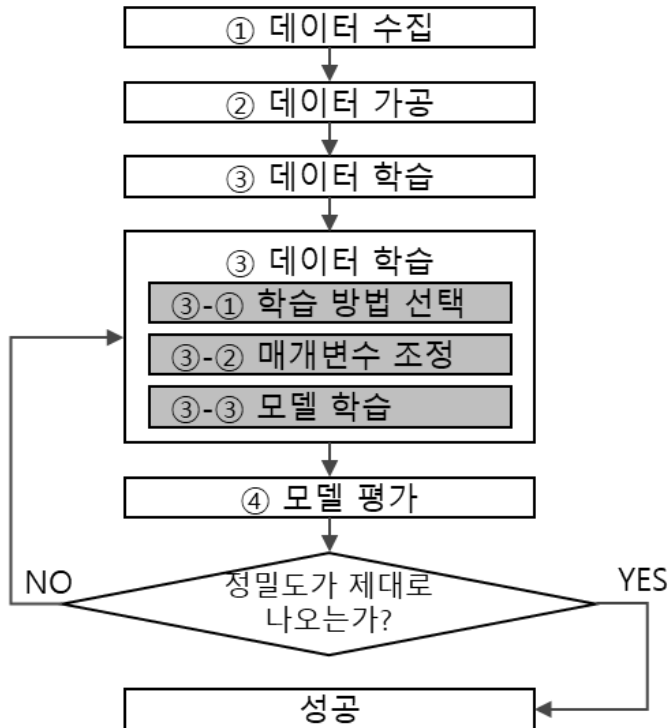


<< 훈련데이터에 최적인 선형 모델 >>

- 선형모델은 좋은지 나쁜지 판단을 위해 비용함수를 사용함
- 훈련과 예측데이터의 거리를 측정하여 최소화하는 것이 목표임

1. 한눈에 보는 머신러닝

■ 머신러닝 시스템 작업 요약

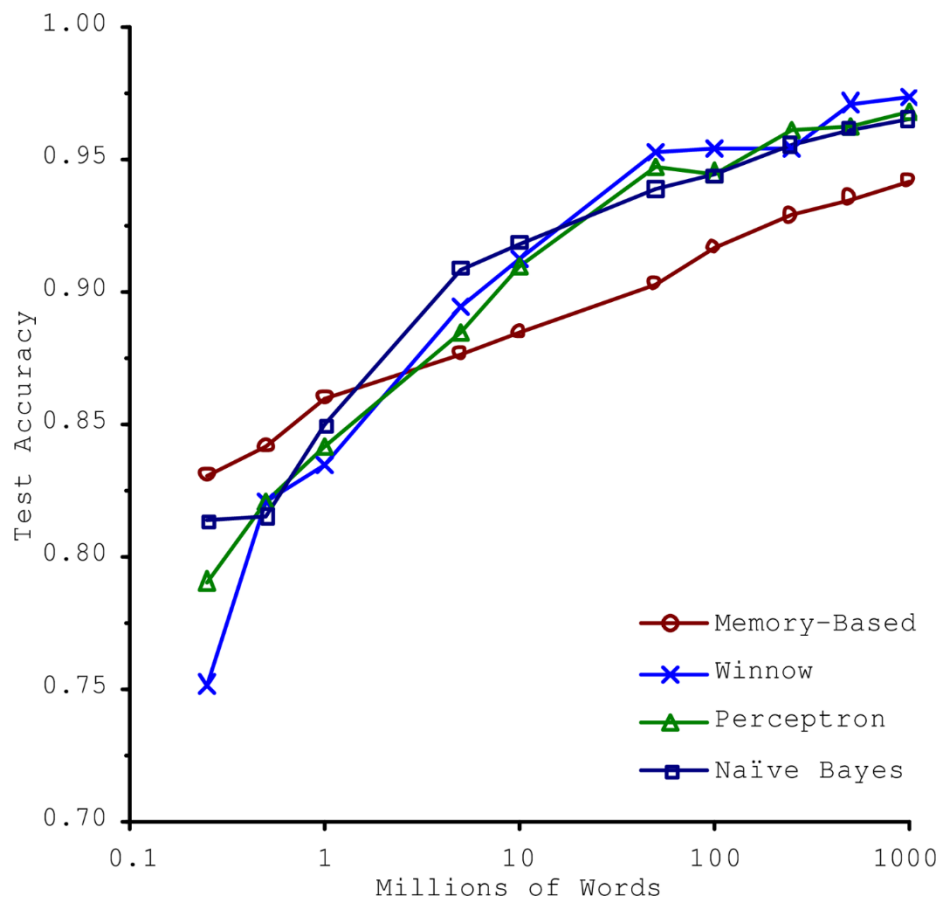


1. 데이터를 분석합니다.
2. 모델을 선택합니다.
3. 훈련데이터로 모델을 훈련시킵니다.
(비용함수가 최소인 모델 파라미터를 찾습니다.)
4. 새로운 데이터에 모델을 적용해 예측, 모델의 일반화를 기대합니다.

1. 한눈에 보는 머신러닝

1.4.1 충분하지 않은 양의 훈련 데이터

- 어린 아이는 사과에 대해 알려주면 모든 종류의 사과를 쉽게 일반화 함



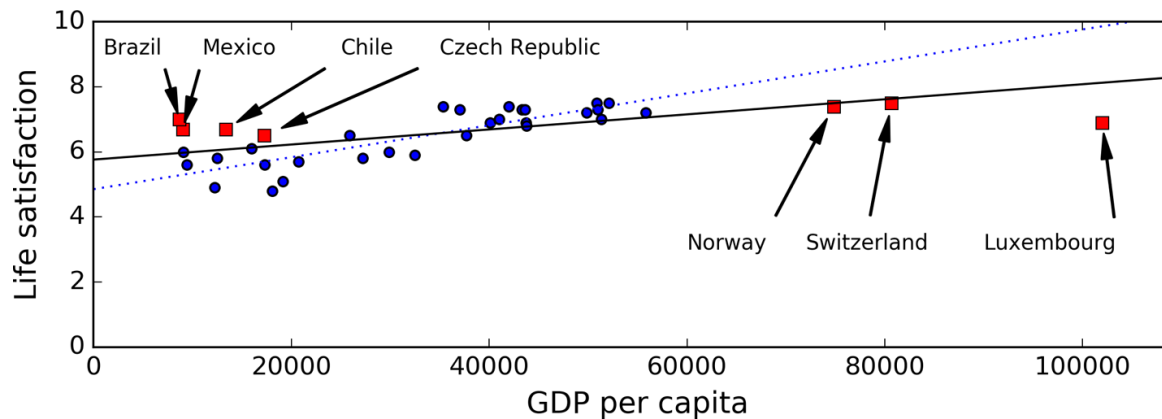
사례1) 알고리즘과 말뭉치 사이의 트레이드오프
2001년 마이크로소프트의 미셜반코와 에릭브릭은 머신러닝 알고리즘에 충분한 데이터가 주어지면 자연어 처리문제를 거의 비슷하게 처리 한다

사례2) 논문 "The Unreasonable Effectiveness of Data"
복잡한 문제에서 알고리즘보다 데이터가 더 중요하다

1. 한눈에 보는 머신러닝

1.4.2 대표성 없는 훈련데이터

- 일반화가 잘되려면 원하는 새로운 사례를 훈련데이터가 잘 대표하는 것이 중요함
- 대표성이 없는 훈련데이터를 추가하면 그래프가 달라짐



- 샘플링 잡음 (Sampling Noise) : 샘플이 작거나 대표성이 없는 데이터
- 샘플링 편향 (Sampling bias) : 표본 추출 방법이 잘못된 경우 → 대표성이 없음

1. 한눈에 보는 머신러닝

1.4.3 낮은 품질의 데이터

- 훈련데이터 정제에 시간을 투자할 만한 가치는 충분함
- 일부 샘플이 이상치라는게 명확하면 무시하거나 수동으로 고치는 것이 좋음
- 일부 샘플에서 특성이 몇 개 빠져 있다면, 이 특성을 넣은 것과 빼 것을 따로 훈련할지 정해야 함

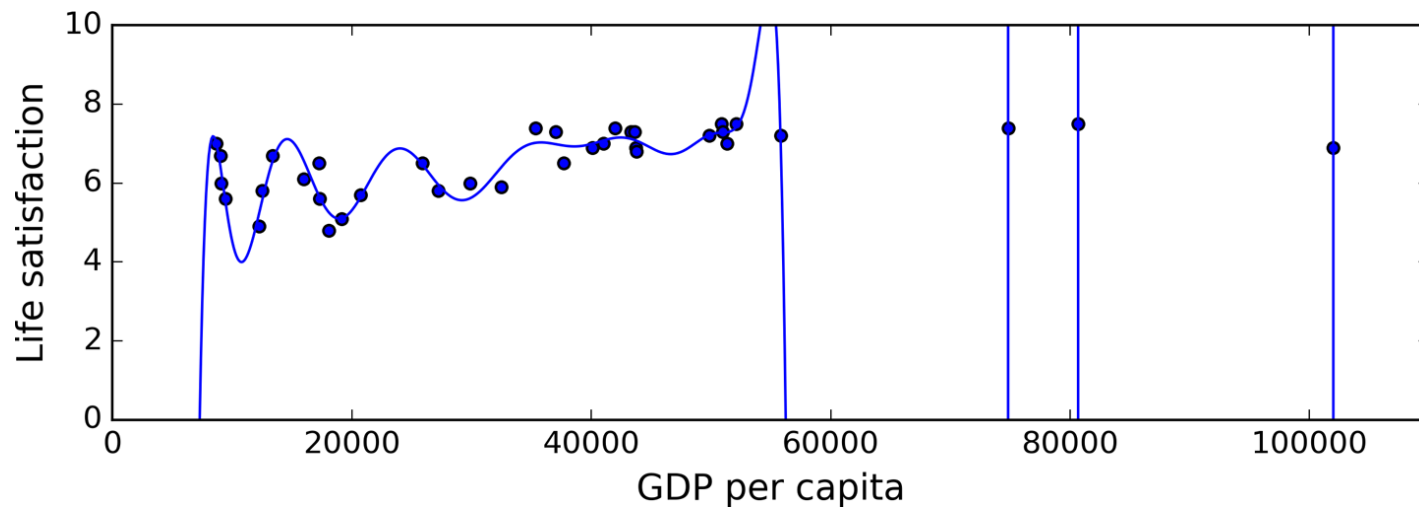
1.4.4 관련 없는 특성

- 성공적인 머신러닝 핵심요소는 훈련에 사용할 좋은 특성을 찾는 것 (특성공학(Feature Engineering))
- 특성 선택 : 가지고 있는 특성 중에서 가장 유용한 특성을 선택
- 특성 추출 : 특성을 결합하여 더 유용한 특성을 만듦 (차원 축소 등)
- 새로운 데이터로 새로운 특성 만듦

1. 한눈에 보는 머신러닝

1.4.5 훈련데이터 과대적합

- 해외여행 중 택시운전사가 내 물건을 훔쳤다 가정하면 그 나라 모든 택시운전사가 도둑이라 생각
→ 일반화의 오류이며, 머신러닝에선 과대적합(Overfitting)이라 함
- 복잡한 모델이 훈련데이터에만 맞는 경우 → 과대적합
- 잡음이 많거나 샘플이 작으면 일반화의 원칙에 부적합

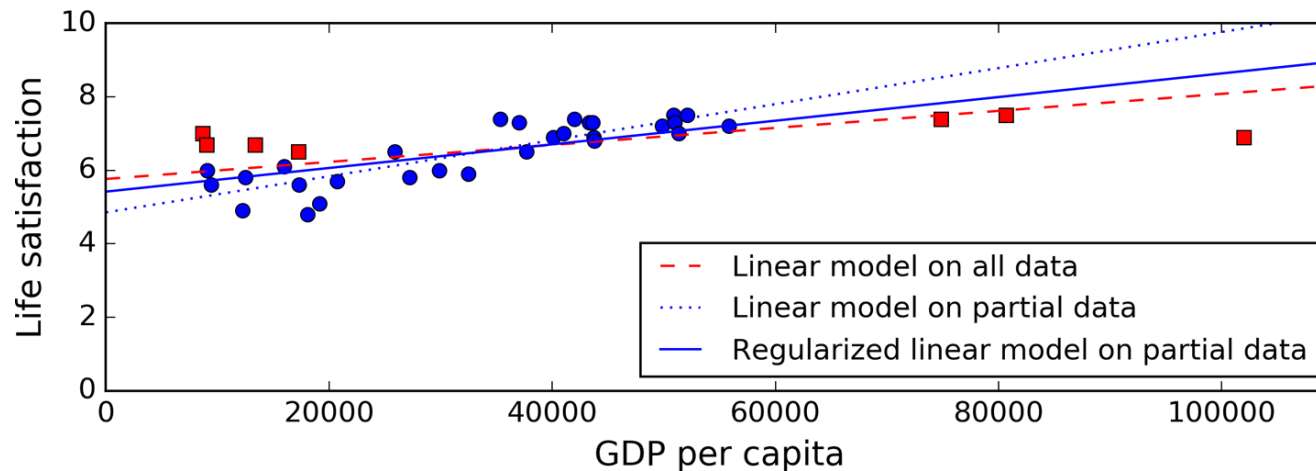


- 고차원의 다항 회귀모델이 삶의 만족도 훈련데이터에 크게 과대 적합된 사례임

1. 한눈에 보는 머신러닝

■ 1.4.5 훈련데이터 과대적합

- 과대적합 해결방법
 - . 파라미터 수가 적은 모델을 선택하거나 훈련데이터에 있는 특성 수를 줄이거나, 모델의 단순화
 - . 훈련데이터를 더 많이 모음
 - . 훈련데이터의 잡음을 줄임 (오류데이터 수정, 이상치 제거 등)



- 규제 과대적합 위험 감소를 위해 모델 파라미터(θ_1)의 값을 작게함 (하이퍼파라미터)

1. 한눈에 보는 머신러닝

1.5 테스트와 검증

- 훈련데이터를 훈련세트와 테스트세트로 구분하여 사용
- 훈련세트를 사용해 모델을 훈련시키고, 테스트 세트를 사용해 모델을 테스트 함
- 새로운 샘플의 오류 비율을 일반화 오차라 하며 테스트 세트에서 모델을 평가, 이 오차에 대한 추정값 획득 (추정값 얻음)
- 보통데이터의 80%를 훈련에, 20%는 테스트용으로 사용
- 훈련세트, 검증세트, 테스트 세트로 나누고 검증세트로 하이퍼파라미터를 조정
- 훈련데이터에서 검증세트로 너무 많은 양의 데이터를 뺏기지 않기 위해 교차검증 기법을 사용

1. 한눈에 보는 머신러닝

1.4.7 한걸음 물러서서

- 머신러닝은 명시적인 규칙을 코딩하지 않고 데이터로부터 학습하여 어떤 작업을 마치는 것
- 머신러닝 시스템은 지도학습과 비지도학습, 배치학습과 온라인학습, 사례기반학습과 모델기반학습 등이 있음
- 학습알고리즘이 모델기반이면 훈련세트에 모델을 맞추기 위해 파라미터를 조정하고 사례기반이면 샘플을 기억하는 것이 학습이고 새로운 샘플에 일반화하기 위해 유사도 측정을 사용
- 훈련세트가 너무 작거나 대표성이 없거나, 잡음이 많고 관련 없는 특성으로 오염되어 있다면 시스템이 잘 작동하지 않음
- 모델이 너무 단순하거나(과소적합) 너무 복잡(과대적합)하지 않아야 함

A photograph of a railway track receding into the distance under a bright, hazy sky with a sun flare.

Thank you for your attention