AI , Machine Learning, Deep Learning
Big Data

동아대학교 산업경영공학과 김해식

빅데이터의 개념

■ 빅데이터의 정의

- 디지털 환경에서 발생하는 대량의 모든 데이터
- 대규모의 데이터를 저장·관리·분석할 수 있는 하드웨어 및 소프트웨어 기술, 데이터를 유통·활용하는 모든 프로세스를 포함
- 빅데이터 플랫폼을 구성하는 하드웨어, 소프트웨어, 애플리케이션 간의 유기적 순환에 의해 가치를 창출

표 2-1 빅데이터의 정의

기관	정의
맥킨지	일반적인 데이터베이스 소프트웨어가 수집, 저장, 관리, 분석할 수 있는 범위를 초과하는 대규모의 데이터다.
가트너	향상된 시사점과 더 나은 의사결정을 위해 사용되는 것으로 비용 효율이 높고 혁신적이며 대용량 고속 및 다양성을 가지는 정보 자산이다.
위키피디아	기존 데이터베이스 관리 도구의 수집, 저장, 관리, 분석 역량을 넘어서는 대량의 정형 또는 비정형 데이터셋 및 이러한 데이터로부터 가치를 추출하고 결과를 분석하는 기술이다.
국가전략위원회	대용량 데이터를 활용 및 분석하여 가치 있는 정보를 추출하고 생성된 지식을 바탕으로 능동적으로 대응하거나 변화를 예측하기 위한 정보화 기술이다.
삼성경제연구소	기존의 관리 및 분석 체계로는 감당할 수 없을 정도의 기대한 데이터 집합으로 대규모 데이터와 관계된 기술 및 도구(수집, 저장, 검색, 공유, 분석, 시각화 등)를 모두 포함한다.
한국정보화진흥원	저장, 관리, 분석할 수 있는 범위를 초과하는 규모의 데이터와 이것을 저장, 관리, 분석할 수 있는 하드웨어 및 소프트웨어 기술, 데이터를 유통 및 활용하는 과정을 통틀어 나타낸다. 즉, 빅데이터를 구성하는 하드웨어, 소프트웨어 그리고 이를 포괄하는 모든 프로세스를 의미하는 거대 플랫폼이다.

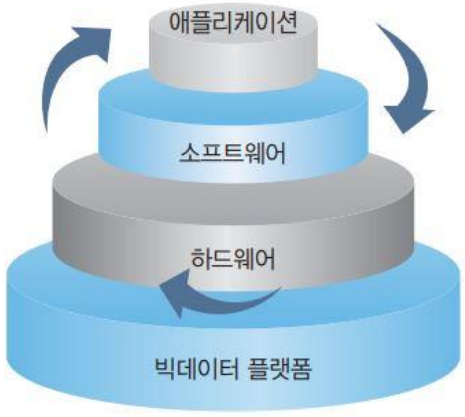


그림 2-1 빅데이터 플랫폼

빅데이터의 개념

■ 빅데이터의 출현

- 기술의 발달과 비용 저하, 소셜 네트워크 서비스 발달, 그림자 정보와 사물 정보 증가 등의 ICT 패러다임의 변화
- 빅데이터에 전문 역량과 기술을 더하여 전략적으로 활용할 방법이 주목됨
- 경제적 가치 창출, 사회 문제 해결, 새로운 ICT 패러다임 견인이라는 신가치 창출

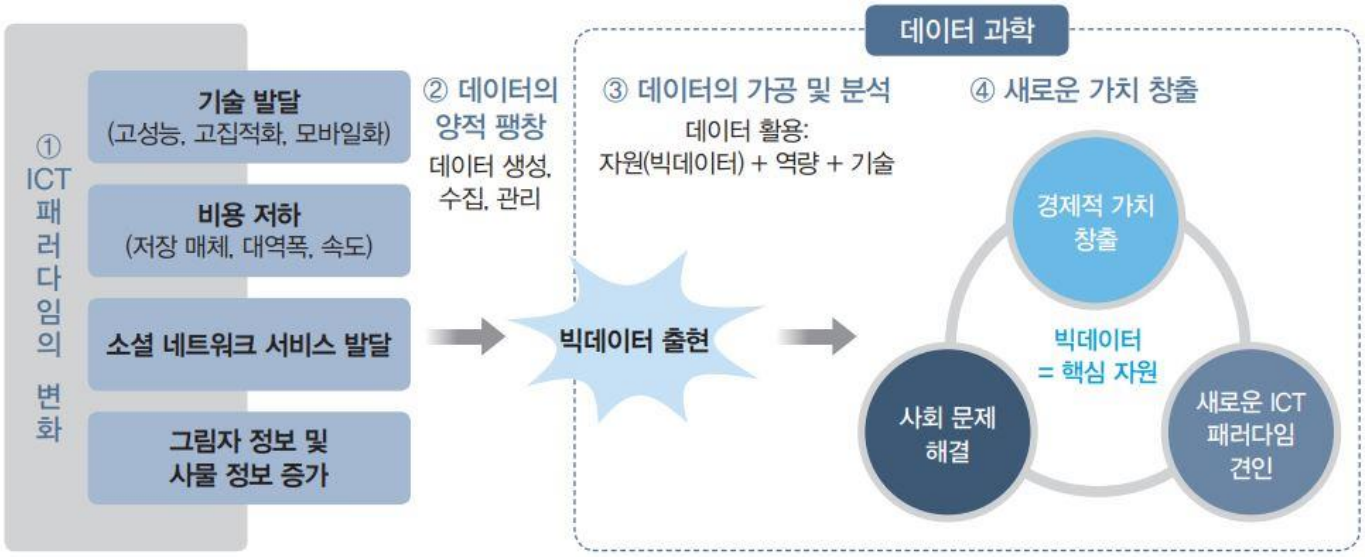


그림 2-2 빅데이터의 출현과 신가치 창출의 흐름

빅데이터의 개념

■ 빅데이터의 분류

- 정형 데이터
  - 일정한 규칙으로 체계적으로 정리된 것으로 그 자체로 해석이 가능하여 바로 활용할 수 있음(관계형데이터베이스, 스프레드시트등)
- 반정형 데이터
  - 고정된 필드에 저장되어 있지는 않지만 XML, HTML, JSON 등의 메타데이터와 스키마를 포함하는 것으로 파일 형태로 저장
- 비정형 데이터
  - 고정된 필드나 스키마가 없는 것
  - 스마트 기기에서 페이스북, 트위터, 유튜브 등으로 생성되는 소셜 데이터
  - IoT 환경에서 생성되는 위치 정보나 센서 데이터와 같은 사물 데이터 등

빅데이터의 개념

■ 빅데이터의 특징

▪ 데이터 측면

- 초기에는 빅데이터의 특징을 3V로 일컬어지는 규모, 다양성, 속도로 나타냄
- 빅데이터를 통한 가치 창출이 중요해지면서 정확성과 가치를 추가한 5V로 나타냄

표 2-3 빅데이터의 특징 – 데이터 측면

구분	특징	설명
1차 특징	규모	• ICT 기술의 발전으로 디지털 정보량이 기하급수적으로 폭증하여 제타바이트 시대로 진입
	다양성	• 데이터의 종류 증가: 로그 기록, 소셜/위치/소비/현실 데이터 등 • 데이터의 유형 다양화: 텍스트 외에 멀티미디어 등의 비정형 데이터 증가
	속도	• 센서, 모니터링 등의 사물 정보와 스트리밍 등의 실시간 정보가 증가하면서 데이터의 생성 및 이동(유통) 속도 증가 • 대규모 데이터를 처리하고 가치 있는 정보를 활용하기 위한 데이터 처리 및 분석 속도 증가
추가 특징	정확성	• 방대한 데이터를 기반으로 분석을 수행하므로 정확성 향상
	가치	• 빅데이터 분석으로 도출된 최종 결과물이 문제 해결을 위한 통찰력을 제공하므로 새로운 가치 창출 가능

빅데이터의 개념

■ 빅데이터의 특징

▪ 분석 환경 측면

- 데이터 분석 시스템의 구성 요소인 데이터, 하드웨어, 소프트웨어 분석 방법은 분석 환경에 따라 다른 특징을 나타냄

표 2-4 빅데이터의 특징 – 분석 환경 측면

요소	과거의 데이터 분석 환경	현재의 빅데이터 분석 환경
데이터	• 정형화된 수치 중심의 자료	• 비정형의 다양한 데이터 • 예: 문자 데이터(SMS, 검색어), 영상 데이터(CCTV, 동영상), 위치 데이터 등
하드웨어	• 고가의 저장 장치 • 데이터베이스 • 대규모 데이터웨어하우스	• 클라우드 컴퓨팅: 비용 대비 효율성 증대
소프트웨어 분석 방법	• 관계형 데이터베이스: RDBMS • 통계 패키지: SAS, SPSS • 데이터 마이닝 • 머신러닝 • 지식 발견	• 오픈 소스 형태의 무료 소프트웨어 • 오픈 소스 통계 솔루션: R • 텍스트 마이닝 • 오피니언 마이닝 • 감성 분석

빅데이터의 개념

■ 빅데이터의 특징

▪ 처리 방식 측면

- 빅데이터는 기존 데이터베이스 관리 시스템(DBMS)으로 처리하던 것에 비해 100배 이상 많은 정형, 비정형 데이터를 처리

표 2-5 빅데이터의 특징 – 처리 방식 측면

구분	이전의 데이터 처리 방식	빅데이터 처리 방식
데이터 트래픽	• 테라바이트 수준	• 페타바이트 수준: 최소 100테라바이트 이상 • 정보의 장기간 수집 및 분석 • 방대한 처리량
데이터 유형	• 정형 데이터 중심	• 비정형 데이터 비중이 높음: SNS 데이터, 로그 파일, 클릭스트림 데이터, 콜센터 로그 통신, CDR 로그 등 • 처리 복잡성 증대
프로세스 및 기술	• 단순한 프로세스 및 기술 • 정형화된 처리 및 분석 결과 • 원인 및 결과 규명 중심	• 다양한 데이터 소스와 복잡한 로직 처리 • 처리 복잡도가 높아 분산 처리 기술 필요 • 새롭고 다양한 처리 방법 필요: 정의된 데이터 모델/상관관계/절차 등이 없음 • 상관관계 규명 중심 • 하둡, NoSQL 등 개방형 소프트웨어 사용

빅데이터의 개념

■ 빅데이터의 가치

▪ 혁신과 창조의 도구

- 빅데이터 분석이 제공하는 스마트 서비스는 기존 비즈니스에 효율화, 개인화, 그리고 미래 예측력을 통한 혁신을 제공
- 단순히 새로운 기술이나 비즈니스 모델이 아니라 새로운 패러다임으로의 변화를 의미
- 빅데이터 자체부터 이를 활용한 사용자 애플리케이션까지 광범위하여 빅데이터 플랫폼과 에코시스템으로 확장

표 2-6 빅데이터 분석을 통한 비즈니스 혁신 방향

방향	설명
효율화	• 빅데이터를 이용해 과거 및 현재의 현상을 파악할 수 있다. • 물류, 재무, 기획, 마케팅 등 경영 전반의 데이터를 실시간으로 분석한 후 최선의 의사결정을 할 수 있다.
개인화	• 온라인 이용자의 활동 정보와 SNS 등으로 축적된 개인 정보를 결합하여 사용자에게 특화된 서비스를 제공할 수 있다. • 현재 개인 정보는 광고 분야에 활용 중인데 이를 넘어 의료, 교육 등 모든 분야로 확대가 가능하다.
미래 예측력	• 과거 및 실시간 데이터를 분석하여 축적한 개인 정보로 개인 또는 조직 전체의 행동 및 의사결정 패턴을 도출할 수 있다. • 미래에 적용 가능한 시나리오를 제시하고 예측 가능한 행동 및 발생 가능한 문제점을 사전에 방지하는 서비스가 가능하다.

빅데이터의 개념

■ 빅데이터의 가치

▪ 사회·경제적 가치

- 빅데이터는 정치, 사회, 경제, 문화, 과학 기술 등 사회 전반에 걸쳐 가치 있는 정보를 제공
- 데이터의 도입과 활용은 산업 경쟁력 제고, 생산성 향상, 혁신을 위한 새로운 가치 창출을 할 것으로 기대

표 2-7 맥킨지가 제시한 빅데이터를 이용한 사회 · 경제적 가치 창출 방법

방법	설명
정보의 투명성	• 이해 관계자가 적시에, 보다 쉽게 빅데이터에 접근할 수 있게 하는 것만으로도 가치 창출이 가능하다. • 예: 공공 부문에서 분리된 부서가 관련 데이터에 보다 쉽게 접근할 수 있으면 데이터 검색과 처리 시간이 절감된다.
실험을 통한 소비자의 요구 발견, 트랜드 예측, 성과 관리	• 더 많은 거래 데이터를 디지털 형태로 축적함에 따라 더욱 정확하고 상세하게 소비자 요구를 발견하거나 트렌드 예측을 할 수 있다. • 예: 관리자가 빅데이터를 사용하여 자연스럽게 발생하거나 통제된 실험으로 일어나는 성과의 변동성을 분석하고 나아가서 근본적인 원인과 결과를 분석하면 더 높은 수준으로 성과를 관리할 수 있다.
소비자 맞춤 비즈니스를 위한 고객 세분화	• 빅데이터를 통해 더 구체적으로 고객을 세분화하여 고객의 요구에 맞는 더 정확한 맞춤형 서비스를 제공할 수 있다. • 예: 공공 부문에서 시민을 세분화하여 필요한 서비스를 제공할 수 있다.
자동화된 알고리즘을 통한 의사결정	• 빅데이터 기술을 사용하여 전체 데이터셋을 정교하게 분석함으로써 의사결정을 개선하고 위험을 최소화할 수 있으며 가치 있는 인사이트를 발굴할 수 있다. • 예: 판매 정보에 실시간 대응하여 재고 및 가격을 자동으로 조정하는 자동화 알고리즘은 소매업체의 의사결정을 최적화할 수 있다.
새로운 비즈니스 모델, 상품, 서비스의 혁신	• 빅데이터를 통해 새로운 상품 및 서비스를 개발하거나 기존 상품 및 서비스를 강화하여 완전히 새로운 비즈니스 모델을 개발할 수 있다. • 예: 실시간 위치 데이터를 이용하여 자동차를 운전하는 장소와 방법에 따라 내비게이션을 제공하고 상해보험 가격도 책정하는 완전히 새로운 위치 기반 서비스가 가능하다.

빅데이터의 활용

■ 빅데이터의 역할

- 미래 사회의 특성은 불확실성, 리스크, 스마트, 융합으로 대변됨
- 빅데이터를 활용해 여러 가지 가능성에 대한 시나리오 시뮬레이션을 하면 불확실한 상황 변화에 유연하게 대처 가능
- 빅데이터에 기반한 정보 패턴 분석으로 리스크에 대응할 수 있음
- 개인화 및 지능화된 서비스를 제공하여 삶의 질을 향상시킴

빅데이터의 활용

■ 빅데이터 활용 전략

▪ 빅데이터 처리 단계와 신기술 이해하기

- 빅데이터는 데이터의 생성 → 수집 → 저장 → 분석 → 표현의 단계를 거치며 세부 영역과 관련 기술이 개발
- 조직과 기업의 혁신 전략으로 적용할 수 있게 빅데이터 플랫폼, 빅데이터 분석 기법 및 기술에 대한 이해가 필요

- 분석 기술
  - 통계, 데이터 마이닝, 머신러닝, 딥러닝, 자연어 처리, 패턴 인식, 소셜 네트워크 분석, 비디오·오디오·이미지 프로세싱 등

- 빅데이터의 활용·분석·처리를 포함하는 인프라
  - BI, DW, 클라우드 컴퓨팅, 분산 데이터베이스 (NoSQL), 분산 병렬 처리, 분산 파일 시스템 등

- 빅데이터 관련 신기술
  - 대용량 데이터 처리를 위한 분산 처리 기술인 하둡과 인메모리, 의미 분석 기술인 데이터 마이닝, 자연어 처리, 머신러닝, 딥 러닝, 그리고 비정형 데이터 처리를 위한 NoSQL 기술

빅데이터의 활용

■ 빅데이터 처리 단계와 신기술 이해하기

표 2-10 빅데이터의 처리 단계별 기술 영역

단계	기술 영역	내용
데이터 소스	내부 데이터	데이터베이스, 파일 관리 시스템
	외부 데이터	파일, 멀티미디어, 스트리밍
수집	크롤링crawling	검색 엔진 로봇을 이용한 데이터 수집
	ETL: 추출Extraction, 변환Transformation, 적재Loading	소스 데이터의 추출, 전송, 변환, 적재
저장	데이터 관리: NoSQL	비정형 데이터 관리
	저장소	빅데이터 저장
	서버	초경량 서버
처리	맵리듀스mapReduce	데이터 추출
	작업 처리	다중 작업 처리
분석	신경 언어 프로그래밍NLP, Neuro Linguistic Programming	자연어 처리
	머신러닝	데이터 패턴 발견
	직렬화serialization	데이터 간 순서화
표현	시각화visualization	데이터를 도표나 그래픽으로 표현
	획득acquisition	데이터의 획득 및 재해석

빅데이터의 활용

■ 데이터 과학자 역량 강화하기

- 빅데이터 시대에는 데이터를 분석하고 관리할 수 있는 인력에 대한 중요성이 큼

- 존 라우치 : 데이터 과학자에게 필요한 6가지 기본 자질
  ① 수학 역량
  ② 공학 역량
  ③ 데이터를 분석할 때 필수적인 가설을 세우거나 검증할 때 필요한 비판적 시각
  ④ 이를 잘 작성할 수 있는 글쓰기 역량
  ⑤ 다른 사람에게 잘 전달할 수 있는 대화 능력
  ⑥ 호기심과 개인의 행복

- 데이터 과학자는 외부보다는 내부 인력으로 내재화하여 활용

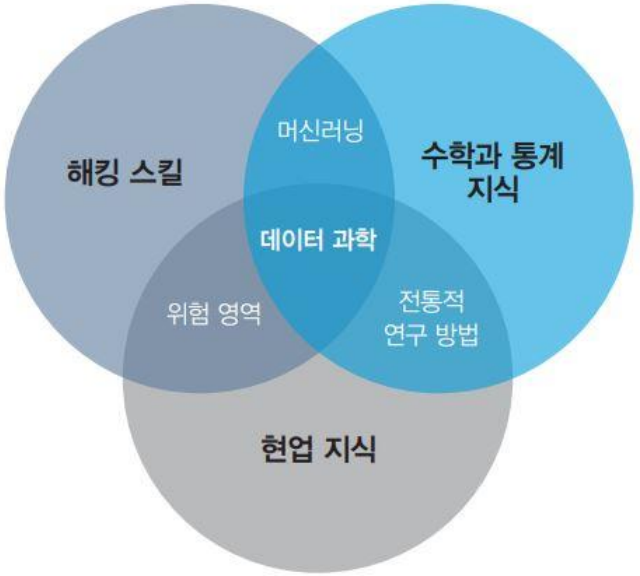


그림 2-5 데이터 과학자가 지녀야 할 역량 벤다이어그램

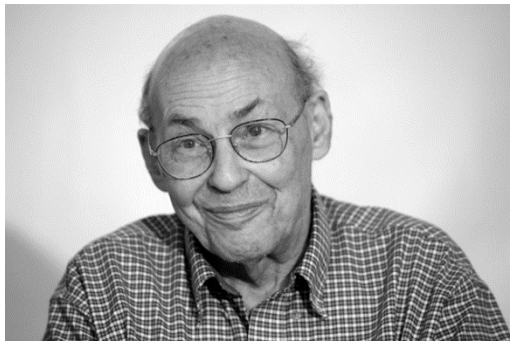
인공지능이란?

- 인공지능(AI)
  - 인공지능은 일반적으로 인간의 지능이 필요하거나 인간이 분석할 수 있는 것보다 규모가 큰 데이터를 포함하는 방식으로 추론, 학습 및 행동할 수 있는 컴퓨터 및 기계를 구축하는 것과 관련된 과학 분야이다.
  - AI는 컴퓨터 공학, 데이터 분석 및 통계, 하드웨어 및 소프트웨어 엔지니어링, 언어학, 신경 과학은 물론 철학과 심리학을 포함하여 여러 학문을 포괄하는 광범위한 분야이다.
  - 비즈니스의 운영 수준에서 AI는 주로 머신러닝과 딥 러닝을 기반으로 하는 기술 모음으로, 데이터 분석, 예상 및 예측, 객체 분류, 자연어 처리, 추천, 지능형 데이터 가져오기 등을 수행할 수 있다.

인공지능의 역사

- 1950년 앨런 튜링의 튜링 기계와 이미테이션 게임에서 컴퓨터 기계와 지능에 대한 논의가 시작
- 1956년에는 다트머스대학교의 하계 컨퍼런스에서 Artificial Intelligence 라는 용어가 처음 사용(1차 전성기)
- 1970년에 들어서자 컴퓨터의 계산 기능과 논리 체계의 한계로 인공지능 이론 구현에 실패(1차 인공지능 겨울)
- 1980년대에는 신경망 다층 퍼셉트론이 개발(2차 전성기)
- 신경망의 성능을 높이기 위해 필요한 데이터가 부족하고 프로세서의 계산 능력이 한계에 도달(2차 인공지능 겨울)
- 2000년대에 들어서면서 메모리, CPU, GPU 등의 하드웨어와 네트워크의 성능 향상으로 신경망 연구가 다시 활발해짐
- 빅데이터가 출현하면서 딥러닝의 성능 향상이 가속, 구글 딥마인드의 알파고가 바둑대회에서 우승(3차 전성기)

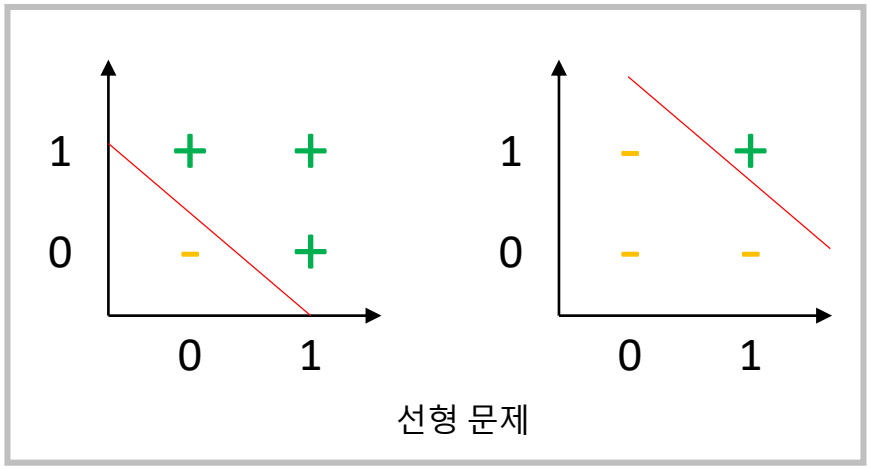
인공신경망의 첫번째 위기 – XOR 문제



Marvin Minsky

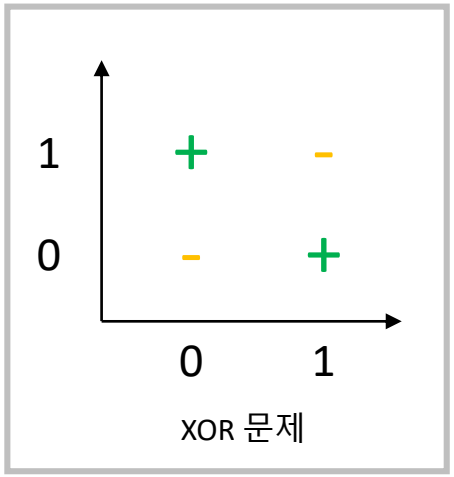
책 “Perceptrons: an introduction to computational geometry”

“Perceptrons은 단순한 선형 분류기이다.
비선형 문제를 해결할 수 없다.
예를 들어 XOR 문제가 바로 그것이다.”



선형 문제

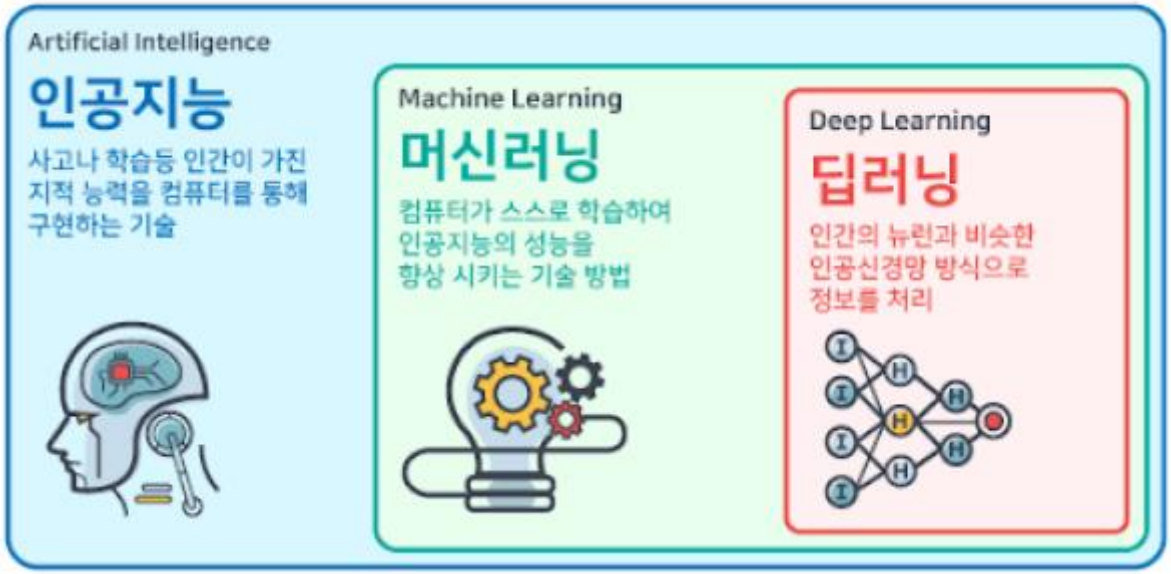
퍼셉트론이 해결 가능



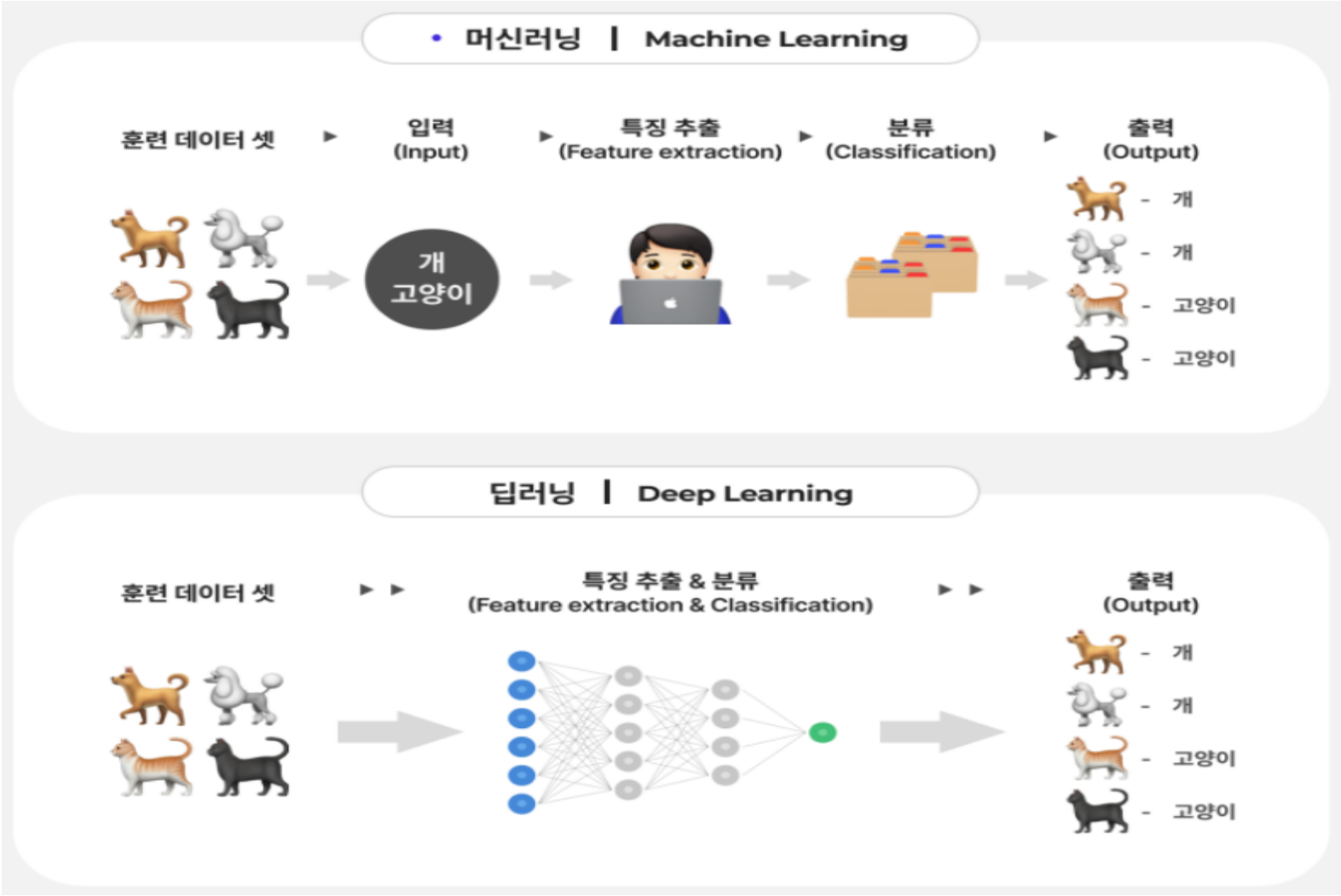
XOR 문제

퍼셉트론이 해결 불가능

AI와 머신러닝과 딥러닝의 관계



머신러닝과 딥러닝의 차이점



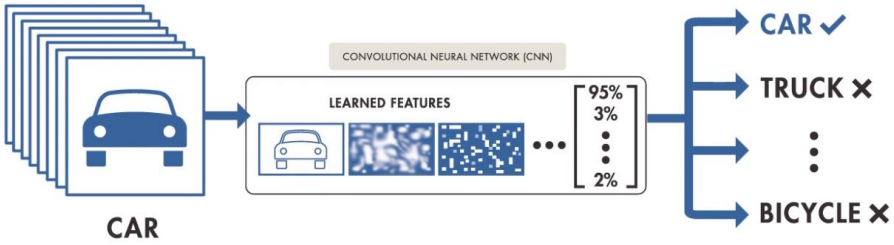
머신러닝과 딥러닝

머신러닝

1. 우선 이미지에서 관련 특징들을 추출한다.
2. 물체의 특징을 예측하는 모델을 만든다.

✓ 반면에 딥러닝은 수동으로 이미지에서 특징 추출을 할 필요가 없다.

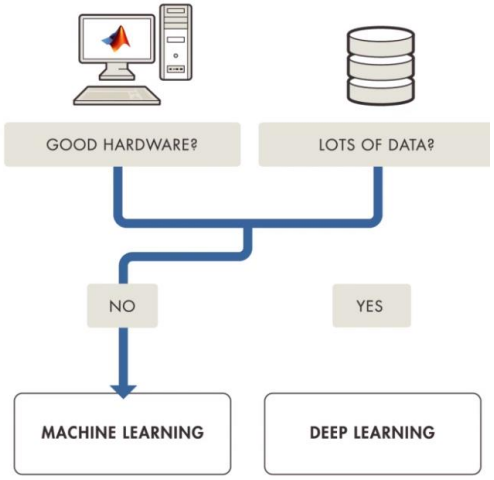
딥러닝



1. 이미지 자체를 딥러닝 알고리즘에 제공함으로써 예측이 가능하다.
   즉, 특징 추출이 필요 없다.

딥러닝이 유리한 경우

1. 데이터가 많을 때
2. 컴퓨팅 파워가 좋을 때(특히 GPU-그래픽카드 좋은 것, 램 용량 큰 것 등등)



	Machine Learning	Deep Learning
Training dataset	Small	Large
Choose your own features	Yes	No
# of classifiers available	Many	Few
Training time	Short	Long

GPU Nvidia RTX2070 8G (61만)
RTX2080Ti 11G (151만)
GTX1060 6G (35만)

성능 : 2080Ti 〉 2070 〉 1080Ti 〉 1070Ti
가성비: 2070 〉 1060 〉 1070Ti 〉 1080Ti

● 딥러닝은 수많은 뉴런들이 동시에 연산을 한다. 그래서 병렬 처리가 중요!

● GPU(Graphics processing unit) : 그래픽 처리 장치 원래 모니터에서 일어나는 수많은 모니터 픽셀 연산을 병렬 처리하는데 사용하는 장치였는데 최근 들어 비트코인 채굴, 딥러닝에 사용하고 있다.

●노트북으로도 딥러닝이 가능하지만, 한계가 있고 본격적으로 딥러닝 하려면 GPU 컴퓨터를 사는 게 낫다.

머신러닝 개요

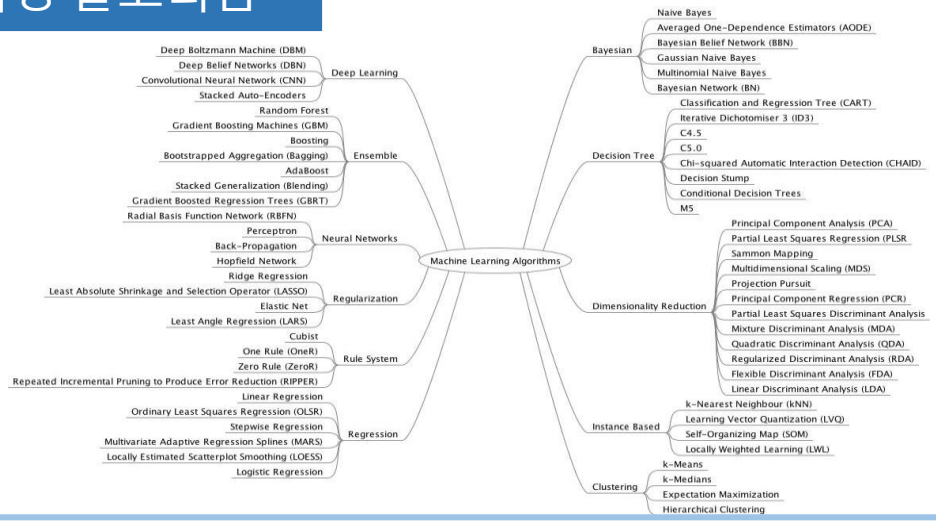
머신러닝 프레임워크

① 프로그래밍 언어 : Python, C/C++, JAVA, MATLAB 등등
② 프레임워크 : Scikit-Learn, TensorFlow, Pytorch, Caffe, theano

머신러닝 분류

① 지도학습 : 분류, 회귀, 시각/음성감지/인지, 텍스트 분석 등
② 비지도학습 : 클러스터링, 차원축소
③ 강화학습

머신러닝 알고리즘

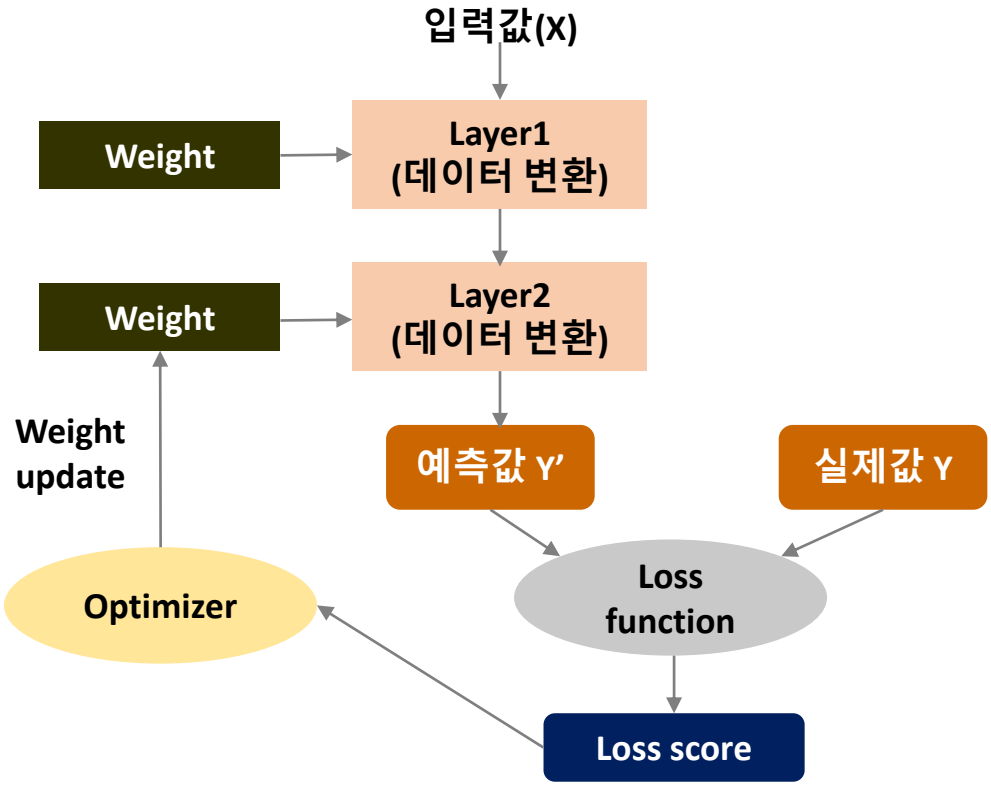


딥러닝 모델(머신러닝 포함) 적용 프로세스 정리

1. 데이터 수집& 확인	2. 데이터 전처리	3. 딥러닝 구조	4. 컴파일	5. 학습	6. 검증
<보유 데이터> - DB - 클라우드 <크롤링> - 셀레늄 - Request, 뷰티풀숲 - Rest API	<데이터 벡터화> - 텍스트 -> 정수화 - 이미지 -> 원-핫 <정규화> - 0~1 값 - 평균0, 표준편차1 - 특성 균일화 <누락 값 처리> - 누락값 0으로 <특성 공학> - 딥러닝은 특성공학 X - NLP n-그램, BOW	<activation> - Sigmoid, ReLU, Softmax, etc. <과대적합 방지> - 가중치 규제(L1, L2) - Dropout <하이퍼 파라미터 튜닝> - 층의 노드 수 - optimizer 학습률	<optimizer> - Adam, RMSProp 등 <loss 함수> - 회귀: MSE - 분류: crossentropy (ROC, AUC과 반비례)	<학습 파라미터> - epoch - batch_size	<모니터링> - accuracy - (train) loss - val_loss

딥러닝 학습 프로세스

입력값이 네트워크 층을 거치면 예측값이 나오고, 이를 실제값과 비교해서 Loss score를 계산한 후에 Optimizer를 통해 Weight를 업데이트 한다.



딥러닝 학습 프로세스 – 폐암 환자의 생존율 예측

폐암 환자 470명 데이터 살펴보기(2013년 폴란드 브로츠와프 의과대학)

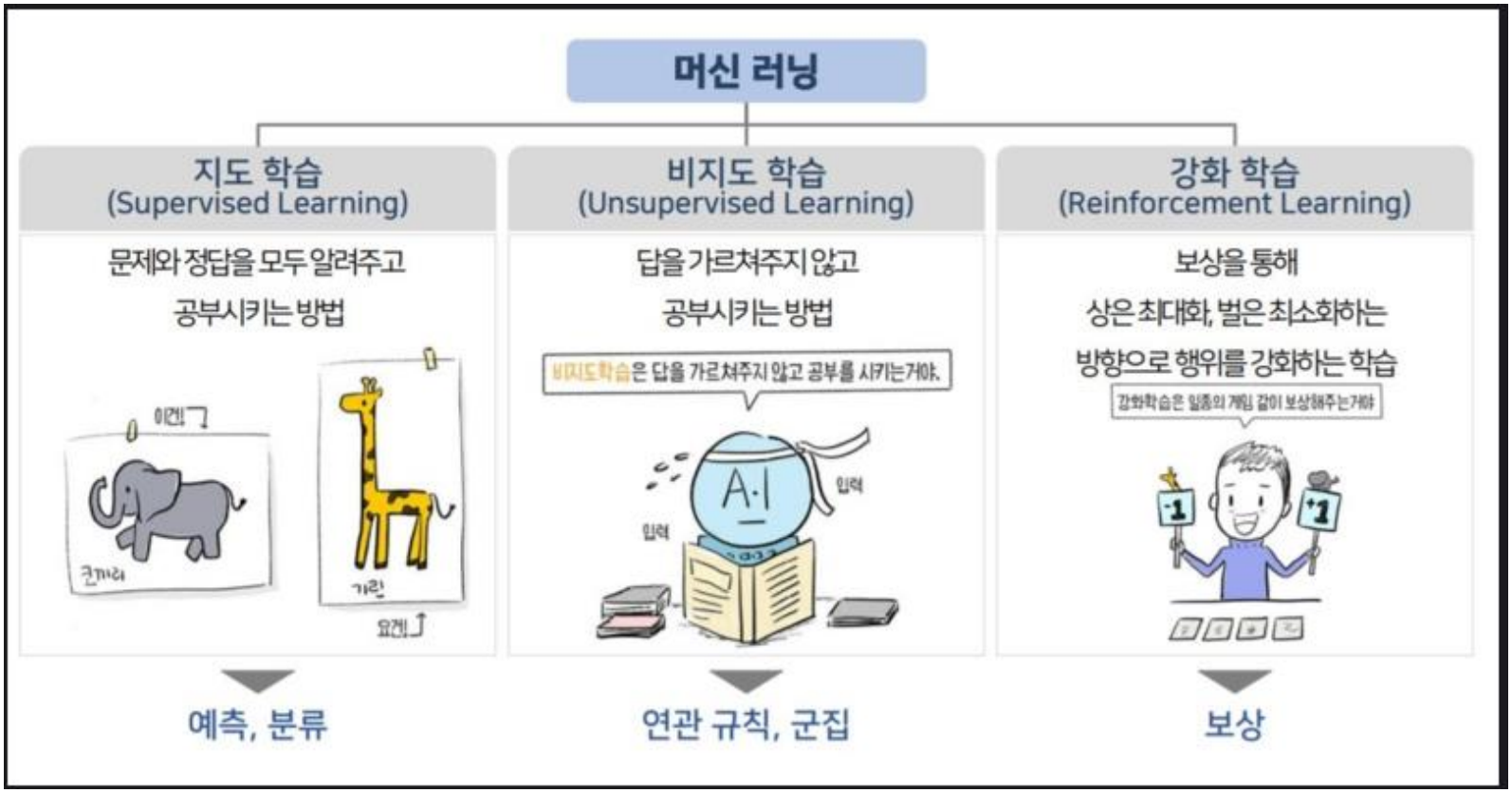
Feature(수술 전 진단데이터)
(종양 유형, 폐활량, 호흡곤란 여부, 고통 정도, 기침, 흡연, 천식 여부 등 17가지)

Class(생존 결과)
(생존:1, 사망:0)

	줄 / 항목	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
환자1	1	293	1	3.8	2.8	0	0	0	0	0	0	12	0	0	0	1	0	62	0
환자2	2	1	2	2.88	2.16	1	0	0	0	1	1	14	0	0	0	1	0	60	0
환자3	3	8	2	3.19	2.5	1	0	0	0	1	0	11	0	0	1	1	0	66	1
	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
환자470	470	447	8	5.2	4.1	0	0	0	0	0	0	12	0	0	0	0	0	49	0

feature: X = Data_set[:, 0:17]
Class: Y = Data_set[:, 17]

머신러닝 개요



출처 : https://busy.org/@urobotics/5bksow

머신러닝 개요

- 지도학습(Supervised Learning) : 지도 학습은 레이블(Label), 즉 명시적인 정답이 있는 데이터가 주어진 상태에서 학습하는 방식이다. 입력 값(X data)이 주어지면 입력값에 대한 Label(Y data)를 주어 학습시키며 대표적으로 분류, 회귀 문제가 있다.

  - 분류(Classification)
    - 분류는 주어진 데이터를 정해진 카테고리(라벨)에 따라 분류하는 문제를 말한다. 분류는 이진 분류 문제 또는 다중 분류 문제가 있다.
    - 즉,기존 데이터가 어떤 레이블에 속하는지 패턴을 알고리즘으로 인지한 뒤에 새롭게 관측된 데이터에 대한 레이블을 판별하는 것
  - 회귀(Regression)
    - 회귀는 여러 개의 독립변수(feature)와 한 개의 종속변수(label)간의 상관관계를 모델링하는 기법을 통칭
    - 예를 들어 아파트의 방 개수, 방 크기, 주변 학군등 여러 개의 독립변수에 따라 아파트 가격이라는 종속변수가 어떤 관계를 나타내는지를 모델링하고 예측하는 것이다.

머신러닝 개요

- 비지도학습(Unsupervised Learning)
  - 지도 학습과는 달리 데이터에 레이블이 없어 비슷한 특징끼리 군집화하여 새로운 데이터에 대한 결과를 예측하는 방식이다. 시스템이 아무런 도움없이 학습하여야 한다.
  - 비지도학습의 대표적인 종류는 군집(clustering)이 있다. 이 외에도 시각화와 차원축소, 연관규칙 학습등이 있다. 예를 들어 블로그 방문자에 대한 데이터가 많이 있을 때 비슷한 방문자들을 그룹으로 묶는 군집알고리즘을 생각해 볼 수 있다. 여기서 방문자가 어떤 그룹에 속하는지 알고리즘에 알려줄 데이터는 없다. 따라서 알고리즘이 방문자 사이의 연결고리를 찾아야 한다.

머신러닝 개요

- 강화학습(Reinforcement Learning)
  - 매우 다른 종류의 알고리즘으로 여기서는 학습하는 시스템을 에이전트라 부르며 환경을 관찰해서 행동을 실행하고 그 결과로 보상 또는 벌점을 받는 방식이다.
  - 시간이 지나면서 가장 큰 보상을 얻기 위해 정책이라고 부르는 최상의 전략을 스스로 학습한다. 정책은 주어진 상황에서 에이전트가 어떤 행동을 선택해야 할 지 정의한다.
  - 대표적인 예가 알파고, 보행로봇 등

머신러닝 개요

- 주요 지도학습 알고리즘(분류)
  - 베이즈 통계와 생성모델에 기반한 나이브 베이즈(Naïve Bayes)
  - 독립변수/종속변수 선형관계성에 기반한 로지스틱 회귀(Logistic Regression)
  - 데이터 균일도에 따른 규칙 기반의 결정트리(Decision Tree)
  - 개별 클래스간의 최대 분류 마진을 효과적으로 찾아주는 서포트 벡터 머신(SVM : Support Vector Machines)
  - 근접거리를 기준으로 하는 K-최근접 이웃(k-Nearest Neighbors)
  - 심층 연결 기반의 신경망(Neural networks)
  - 서로 다른(또는 같은)머신 러닝 알고리즘을 결합한 앙상블(Ensemble)

머신러닝 개요

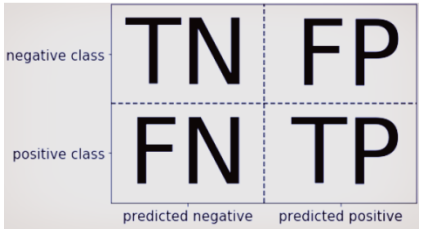
- 주요 지도학습 알고리즘(회귀)
  - 일반선형회귀
  - 릿지(Ridge)
  - 라쏘(Lasso)
  - 엘라스틱넷(ElasticNet)
  - 로지스틱 회귀(Logistic Regression)

머신러닝 개요

- 주요 비지도학습 알고리즘
  - 군집(clustering)
    - K평균(k-means)
    - 계층군집분석(HCA : hierarchical Cluster Analysis)
    - 기대값 최대화(Expectation Maximization)
  - 시각화(Visualization)와 차원축소(Dimensionality reduction)
    - 주성분분석(PCA : Principal Component Analysis)
    - 커널(Kernel) PCA
    - 지역적 선형 임베딩(LLE : Locally-Linear Embedding)
    - t-SNE(t-distributed Stochastic Neighbor Embedding)
  - 연관규칙학습(Association rule learning)
    - 어프라이어리(Apriori)
    - 아클렛(Eclat)

머신러닝 개요

- 머신러닝 모델의 예측성능에 대한 주요 평가 지표(분류)
  - 정확도 = $\frac{\text{예측결과가 동일한 데이터 건수}_{(TN+TP)}}{\text{전체 예측 데이터 건수}_{(TN+FN+FP+TP)}}$
    - 데이터 구성에 따라 ML 모델의 성능을 왜곡할 수 있기 때문에 이것만 가지고 평가해서는 안됨
  - 오차행렬
    - 이진분류에서 성능지표로 활용, 학습된 분류모델이 예측을 수행하면서 얼마나 헷갈리고(confused) 있는지도 함께 보여주는 지표



negative(0)

positive(1)

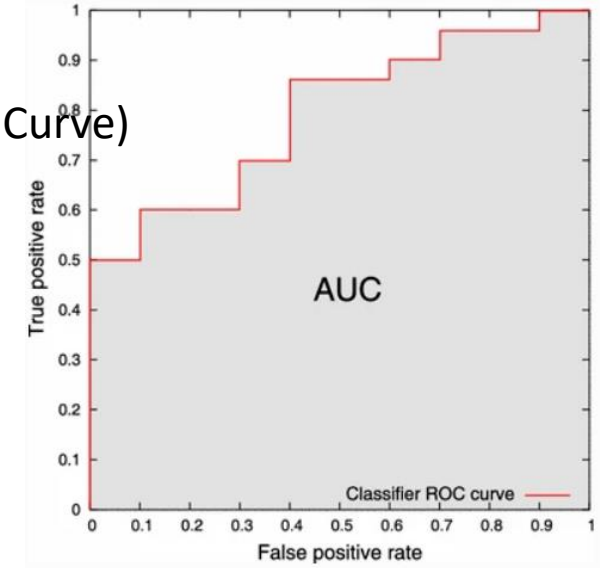
예측 **negative(0)** **positive(1)**

머신러닝 개요

- 머신러닝 모델의 예측성능에 대한 주요 평가 지표(분류)
  - 정밀도 = $\frac{TP}{(\boldsymbol{FP}+TP)}$
    - 예측을 positive 로 한 대상 중에 예측과 실제값이 positive 로 일치한 데이터의 비율을 의미(TP를 높이는 거 외에 FP 를 낮추는 초점을 맞춤)
    - positive 예측성능을 더욱 정밀하게 측정하기 위한 평가지표로 실제 negative 인 데이터의 예측을 positive 로 잘못 판단하게 될 경우 업무상 영향이 큰 경우
    - 스팸메일을 필터링 하는 경우(일반 메일을 스팸으로 분류할 경우, 메일 X)
  - 재현율(민감도) = $\frac{TP}{(\boldsymbol{FN}+TP)}$
    - 실제값이 positive 인 대상 중에 예측과 실제값이 positive로 일치한 데이터의 비율을 의미(TP를 높이는 거 외에 FN을 낮추는데 초점을 맞춤)
    - 재현율이 중요 지표인 경우, 실제 positive 양성 데이터를 negative 로 잘못 판단하게 되면 업무상 큰 영향이 발생하는 경우(예. 암 판단모델, 금융사기 적발 모델등)

머신러닝 개요

- 머신러닝 모델의 예측성능에 대한 주요 평가 지표(분류)
  - F1 스코어
    - 정밀도와 재현율을 결합한 지표, 재현율(R) 과 정밀도(P)의 조화평균
    - $F1 = \frac{2PR}{P+R} = \frac{2TP}{총샘플수 + TP - TN}$
    - 정밀도와 재현율이 어느 한쪽으로 치우치지 않는 수치를 보일 때 높은 값을 상대적으로 가짐.
  - ROC(Receiver OC Curve) 와 AUC(Area Under the Curve)
    - ROC-AUC 곡선은 다양한 임계값에서 모델의 분류성능에 대한 측정 그래프
    - AUC 가 높다는 사실은 **클래스를 구별**하는 모델의 성능이 훌륭하다는 것을 의미



머신러닝 개요

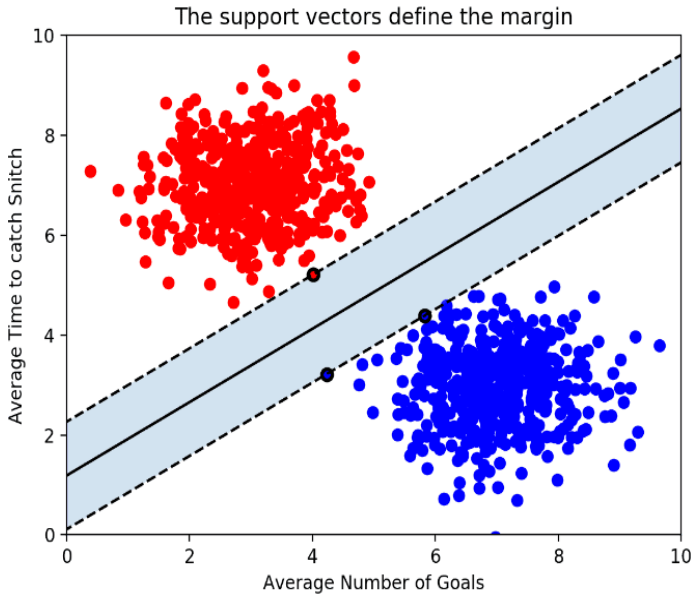
- 머신러닝 모델의 예측성능에 대한 주요 평가 지표(회귀)
  - MAE(Mean Absolute Error) = $\frac{1}{N}\sum_{i}^{N}|y_i - \hat{y}_i|$
  - MSE(Mean Squared Error) = $\frac{1}{N}\sum_{i}^{N}(y_i - \hat{y}_i)^2$
  - RMSE(Root Mean Squared Error)
  - $R^2 = \frac{\text{예측값 분산}}{\text{실제값 분산}}$

주요 알고리즘 소개(분류)

- 나이브 베이즈
  - 나이브 베이즈 분류는 베이즈 정리에 기반한 통계적 분류 기법이다. 가장 단순한 지도 학습 중 하나로써, 빠르고, 정확하며, 믿을만한 알고리즘이다.
  - 정확성도 높고 대용량 데이터에 대해 속도도 빠르며, 주로 스팸 메일 필터, 텍스트 분류, 감정 분석, 추천 시스템 등에 광범위하게 활용되는 분류 기법
  - 단, feature 간의 독립성(feature간에 서로 상관관계가 없어야 함)이 있어야 한다. 하지만 실제 데이터에서 모든 feature가 독립인 경우는 드물다. 장점이 많지만 feature가 서로 독립이어야 한다는 단점이 있다.

주요 알고리즘 소개(분류)

- SVM(Support Vector Machine)
  - 선형이나 비선형 분류, 회귀, 이상치 탐색에도 사용할 수 있는 다목적 모델이며 , 복잡한 분류문제에 잘 들어맞으며 작거나 중간크기의 데이터셋에 적합함.
  - 기본 아이디어는 Margin(선과 가장 가까운 양 옆 데이터와의 거리)을 최대화하는 것. 이 때 선과 가장 가까운 포인트를 서포트 벡터(Support vector)라고 한다
  - **결정 경계(Decision Boundary)**, 즉 분류를 위한 기준선을 정의하는 모델이다. 그래서 분류되지 않은 새로운 점이 나타나면 경계의 어느 쪽에 속하는지 확인해서 분류 과제를 수행할 수 있게 된다.



주요 알고리즘 소개(분류)

- SVM(Support Vector Machine)
  - 장점은 범주나 수치예측문제에 사용 가능하며, 오류 데이터에 대한 영향이 없다는 점과 과적합이 되는 경우가 적고 신경망보다 사용하기 쉬운 장점이 있다.
  - 단점은 최적 모델을 찾기 위해서 커널과 모델에서 다양한 테스트가 필요하다. 따라서 연산이 필요하고 입력데이터 셋이 많을 경우에 학습 속도가 느리다.

주요 알고리즘 소개(분류)

- 과소적합과 과대적합(Under-fitting and Over-fitting)
  - 과대적합(overfitting)은 모델이 훈련 데이터에 너무 잘 맞지만 일반성이 떨어진다는 의미이다.
  - 과소적합(underfitting)은

과대적합의 반대의미로

모델이 너무 단순해서

데이터의 내재된 구조를

학습하지 못할 때 발생한다.

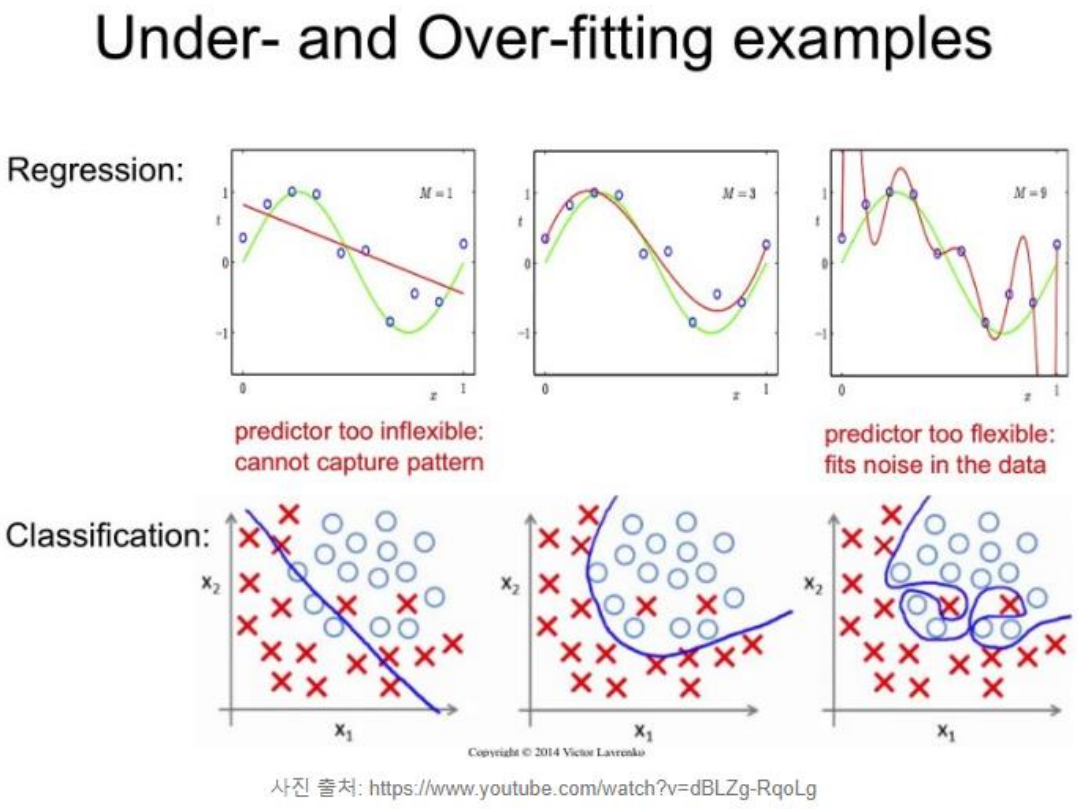
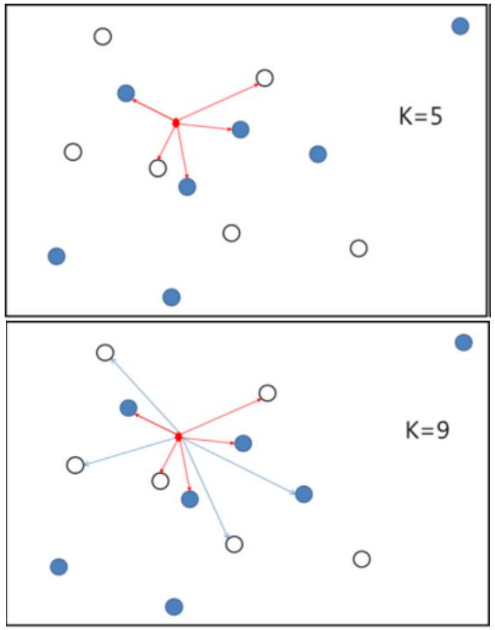


사진 출처: https://www.youtube.com/watch?v=dBLZg-RqoLg

주요 알고리즘 소개(분류)

- *k*-최근접 이웃 알고리즘
  - 분류와 회귀 문제를 해결할 수 있는 알고리즘으로 특정 입력 데이터에 대해, 주변 k개의 데이터 클래스 중 가장 많은 클래스로 특정 자료를 분류하는 방식이다.
  - **장점**
    - 비모수적 방법이기 때문에 어떤 분포든 상관 없음
    - 쉽고 이해하기 직관적
    - 샘플 수가 많을 때 좋은 분류법이다.
  - **단점**
    - 최적의 k를 선택하기가 어렵다.
    - 데이터가 많을 때 분석속도가 느릴 수 있음.
    - 특정분포를 가정하지 않기 때문에 샘플수가 많이 있어야 정확도가 좋다.



주요 알고리즘 소개(분류)

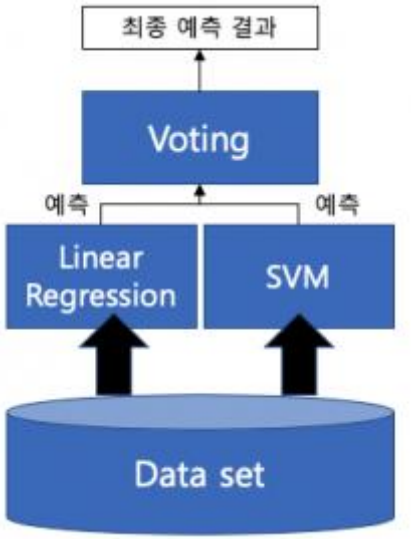
- 결정트리 (Decision Tree)
  - 결정 트리는 분류와 회귀 모두 가능한 모델 중 하나
  - 규칙이 많으면 분류를 결정하는 방식이 복잡해지며, 이는 곧 과적합으로 이어짐. 즉 트리의 깊이가 깊어질수록 예측성능이 저하될 수 있음.
  - 가능한 적은 결정노드로 높은 예측 정확도를 가지려면 데이터 분류시 최대한 많은 데이터 셋이 해당 분류에 속할 수 있도록 규칙이 정해질 필요가 있음.(정보의 균일도 고려)
  - 결정트리 모델 구현시 여러 하이퍼 파라미터 고려

주요 알고리즘 소개(분류)

- 앙상블 학습(Ensemble Learning)
  - 어떤 데이터의 값을 예측한다고 할 때, 보통 하나의 모델을 활용하지만 여러 개의 모델을 조화롭게 학습시켜 그 모델들의 예측 결과들을 이용한다면 더 정확한 예측값을 구할 수 있다.
  - 앙상블 학습은 여러 개의 결정 트리를 결합하여 하나의 결정 트리보다 더 좋은 성능을 내는 머신러닝 기법이다. 앙상블 학습의 핵심은 여러 개의 약 분류기(Weak Classifier : 예측성능이 조금 낮은)를 결합하여 강 분류기(Strong Classifier)를 만드는 것이며, 이에 따라 모델의 정확성이 향상된다.
  - 앙상블 학습의 유형은 보팅(Voting), 배깅(Bagging), 부스팅(Boosting)의 세 가지로 구분할 수 있다.

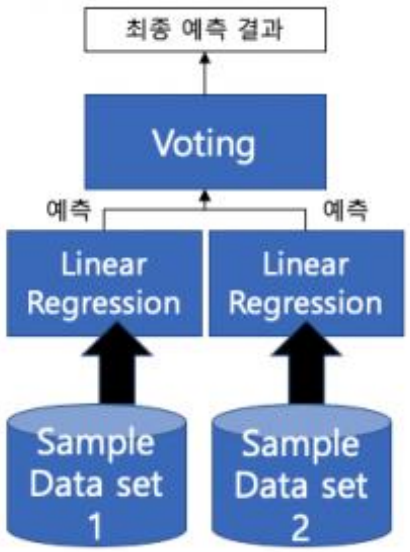
주요 알고리즘 소개(분류)

- **보팅(Voting)**
  - 여러 개의 분류기가 투표를 통해 최종 예측 결과를 결정하는 방식
  - 서로 다른 알고리즘을 여러 개 결합하여 사용
  - 보팅 방식
    - **하드 보팅(Hard Voting)**
      - 다수의 분류기가 예측한 결과값을 최종 결과로 선정
    - **소프트 보팅(Soft Voting)**
      - 모든 분류기가 예측한 레이블 값의 결정 확률
      - 평균을 구한 뒤 가장 확률이 높은 레이블 값을 최종 결과로 선정



주요 알고리즘 소개(분류)

- **배깅(Bootstrap AGGregating, Bagging)**
  - 데이터 샘플링(Bootstrap) 을 통해 모델을 학습시키고 결과를 집계하는 방법으로 모두 같은 유형의 알고리즘 기반의 분류기를 사용
  - 데이터 분할 시 중복을 허용
  - 범주형 데이터 : 다수결 투표 방식으로 결과 집계
  - 연속형 데이터 : 평균값 집계
  - 과적합(Overfitting) 방지에 효과적
  - 대표적인 배깅 방식 : 랜덤 포레스트 알고리즘



주요 알고리즘 소개(분류)

- **부스팅(Boosting)**
  - 여러개의 분류기가 순차적으로 학습을 수행
  - 이전 분류기가 예측이 틀린 데이터에 대해서 올바르게 예측할 수 있도록 다음 분류기에게 가중치(weight)를 부여하면서 학습과 예측을 진행
  - 계속하여 분류기에게 가중치를 부스팅하며 학습을 진행하기에 부스팅 방식이라고 불림
  - 예측 성능이 뛰어나 앙상블 학습을 주도
  - 대표적인 부스팅 모듈 – XGBoost, LightGBM
  - 보통 부스팅 방식은 배깅에 비해 성능이 좋지만, 속도가 느리고 과적합이 발생할 가능성이 존재하므로 상황에 따라 적절하게 사용해야 함.

주요 알고리즘 소개(회귀)

- 일반선형회귀
  - 예측값과 실제값의 RSS(Residual Sum of Squares)를 최소화할 수 있도록 회귀계수를 최적화하며, 규제를 적용하지 않은 모델
  - 다항회귀, 다중회귀 가능하며, 과소적합과 과대적합 문제 발생할 수 있음.
- 릿지(Ridge) 회귀
  - 선형회귀에 L2 규제를 추가한 회귀 모델. L2 규제는 상대적으로 큰 회귀계수값의 예측 영향도를 감소시키기 위해서 회귀계수값을 더 작게 만드는 규제모델임.
  - 이 회귀방법은 일반적으로 영향을 거의 미치지 않는 특성에 대하여 0에 가까운 가중치를 주게 된다.
- 라쏘(Lasso) 회귀
  - 선형회귀에 L1 규제를 추가한 회귀 모델. L2 규제가 회귀계수값의 크기를 줄이는 데 반해 L1 규제는 예측 영향력이 작은 피처의 회귀계수를 0으로 만들어 예측시 피처가 선택되지 않게 하는 규제로 L1 규제는 피처 선택 기능으로도 불리움.

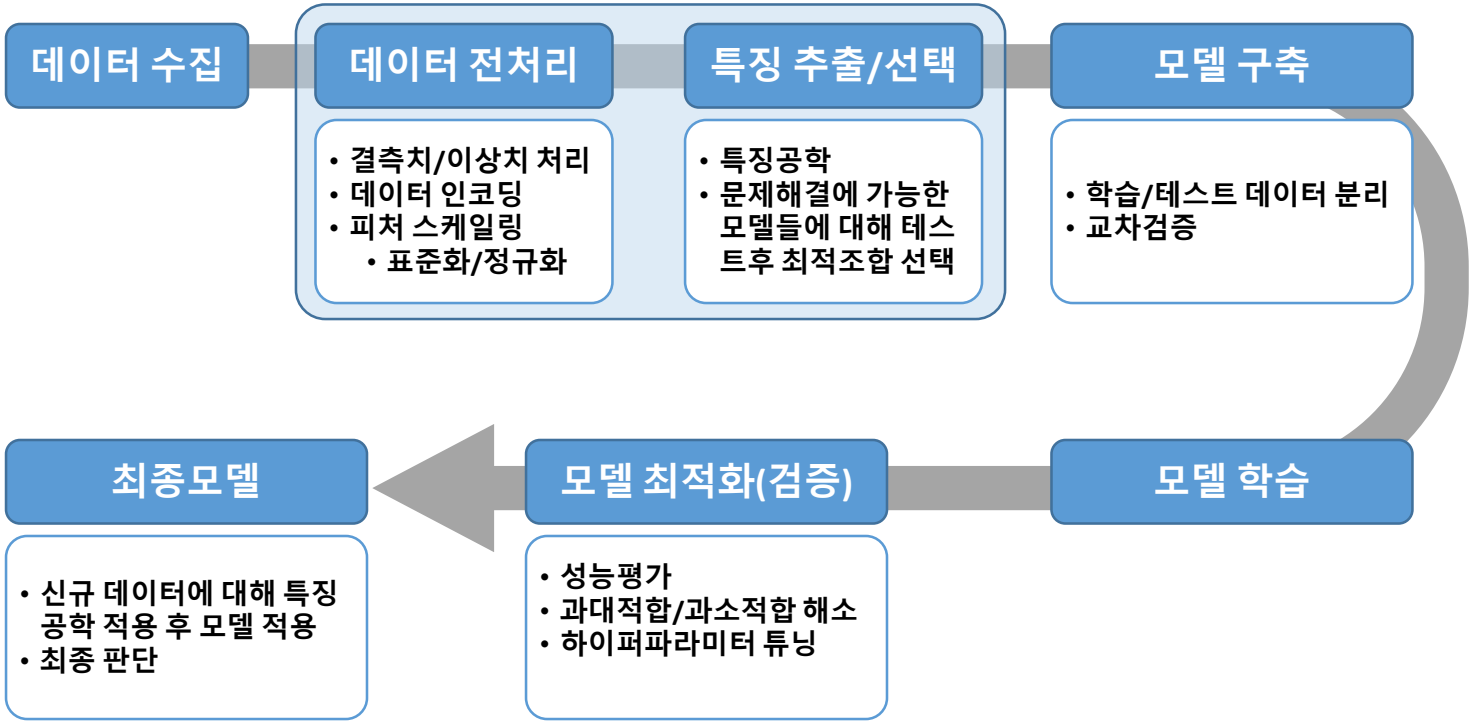
주요 알고리즘 소개(회귀)

- 엘라스틱넷(ElasticNet) 회귀
  - L1, L2 규제를 함께 결합한 모델이며, 주로 피처가 많은 데이터 셋에서 적용되며, L1 규제로 피처의 개수를 줄임과 동시에 L2 규제로 계수 값의 크기를 조정한다.
- 로지스틱 회귀
  - 회귀라는 이름이 붙었지만, 사실은 분류에 사용되는 선형모델이다. 로지스틱 회귀는 매우 강력한 분류알고리즘이며, 이진 분류뿐만 아니라 텍스트 분류와 같은 영역에서도 좋은 예측 성능을 보인다.

머신러닝의 또 다른 기준에 의한 분류

- 입력데이터로부터 점진적인 학습 가능여부
  - 배치학습 : 점진적인 학습이 아니라 가용한 모든 데이터를 사용해 훈련하는 방식으로 훈련후 제품 시스템에 적용하면 더 이상의 학습없음. 즉 학습한 것을 단지 적용만 함(오프라인 학습)
  - 온라인 학습 : 데이터를 순차적으로 한 개씩 또는 미니배치 크기로 시스템을 훈련시킴(주식가격 예측). 학습률, 모니터링 중요(나쁜 데이터가 주입되었을 때, 경우에 따라서는 학습을 중단시킬 필요 있음.
- 어떻게 일반화되는가?
  - 사례기반학습 : 예를들어 스팸메일 예측할 때, 스팸메일과 유사한 메일을 구분지어 기억시키는 방식으로 예측(유사성 : 공통 포함 단어 등)
  - 모델기반학습 : 샘플들의 모델(선형모델 등)을 만들어 예측

머신러닝 모델 적용 프로세스



인공지능, 머신러닝, 딥러닝

- Feature
  - 머신러닝은 어떤 데이터를 분류하거나, 값을 예측(회귀)하는 것이다.
  - 이렇게 데이터의 값을 잘 예측하기 위한 데이터의 특징들을 머신러닝/딥러닝에서는 "Feature"라고 부르며, 지도, 비지도, 강화학습 모두 적절한 feature를 잘 정의하는 것이 핵심이다.
  - 엑셀에서 attribute(column)라고 불려지던 것을 머신러닝에서는 통계학의 영향으로 feature라고 부른다.
  - 예를 들어 고양이, 강아지 사진은 분류한다고 하면 고양이는 귀가 뾰족하다 거나 눈코입의 위치, 무늬 등이 피처가 됩니다. 키와 성별을 주고 몸무게를 예측한다고 하면 키와 성별이 피처가 됩니다.
  - Feature는 Label, Class, Target, Response, Dependent variable 등으로 불려진다.

파이썬 머신러닝 생태계를 구성하는 주요 패키지

- 머신러닝 패키지
  - Scikit-Learn (데이터 마이닝 기반의 머신러닝), TensorFlow , Pytorch 등
- 행렬/선행대수/통계 패키지
  - numpy, Scipy
- 데이터 핸들링
  - Pandas : 2차원 데이터 처리에 특화되어 있음.
- 시각화
  - Matplotlib, Seaborn
- 주피터 노트북

Scikit-learn

Scikit-learn 라이브러리

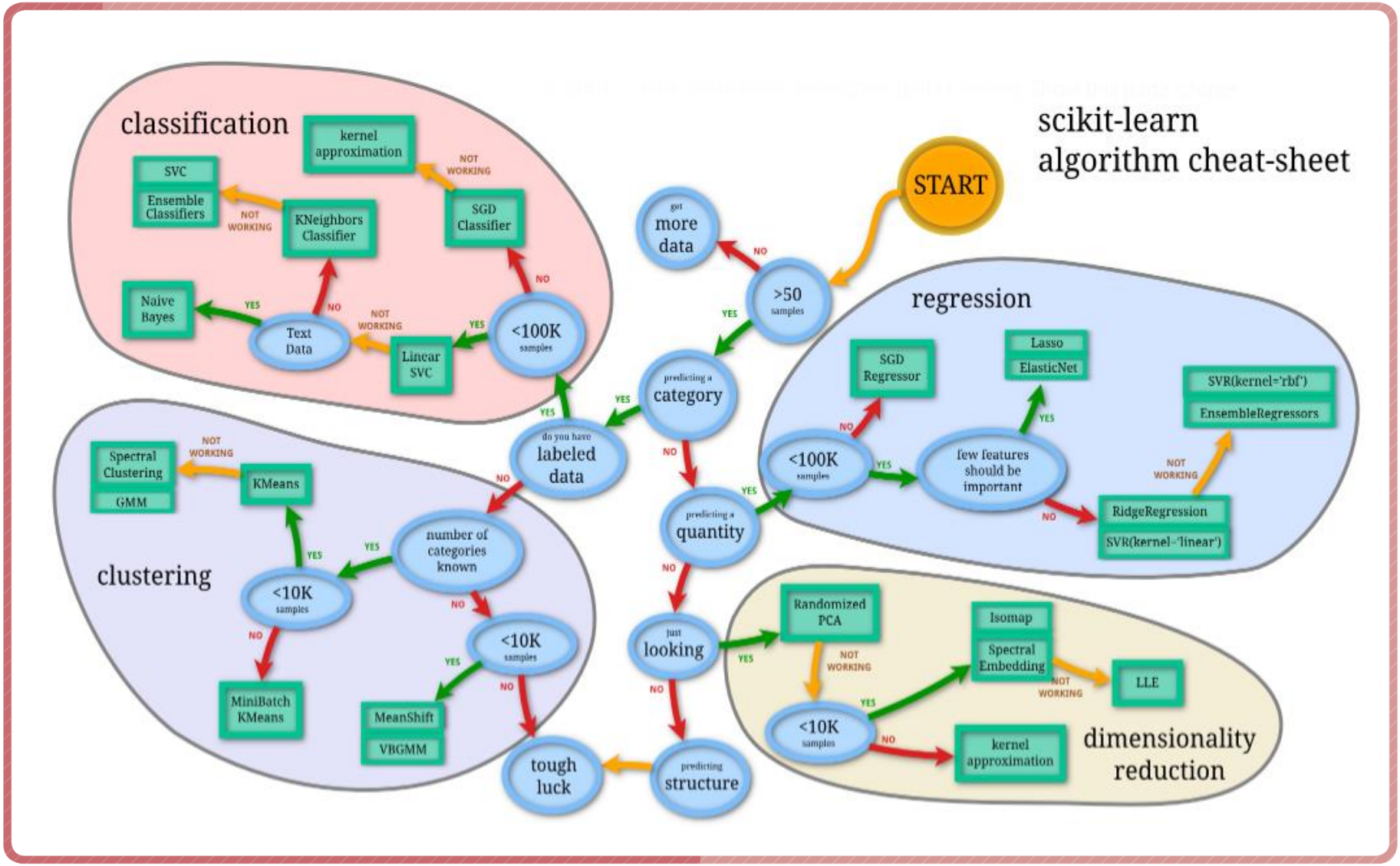
- Scikit-learn 은 머신러닝 알고리즘을 구현한 오픈소스 라이브러리 중 가장 범용적이며 유명한 라이브러리 중 하나
- 일관되고 간결한 API 가 강정이며, 문서화가 잘되어 있음
- 알고리즘은 파이썬 클래스로 구현되고 데이터셋은 numpy 배열, pandas DataFrame, SciPy 희소행렬을 사용할 수 있음.
- Tensorflow, pytorch 는 딥러닝분야에 주로 사용

Scikit-learn 라이브러리

- Scikit-learn 에서 제공되는 주요 데이터 셋

데이터셋	샘플갯수	독립변수	종속변수	데이터로드 함수
보스턴 주택가격 데이터	506	13	주택가격	load_boston()
붓꽃 데이터	150	4	붓꽃종류	load_iris()
당뇨병 환자 데이터	442	10	당뇨병수치	load_diabetes()
숫자 0~9를 손으로 쓴 흑백 데이터	1797	64	숫자	load_digits()
와인의 화학 성분 데이터	178	13	와인종류	load_wine()
체력검사 데이터	20	3	체력검사 점수	load_linnerud()
유방암 진단 데이터	569	30	약성/양성	load_breast_cancer()

Scikit-learn algorithm cheat-sheet



Scikit-learn 의 데이터 표현방식

- 특징행렬(Feature Matrix)
  - 표본은 데이터셋이 설명하는 개별 객체를 나타냄
  - 특징(feature)은 각 표본을 연속적인 수치값, 부울값, 이산값으로 표현하는 개별 관측치를 의미
  - 표본 : 행렬의 행
  - 행의 개수 : n_samples
  - 피쳐 : 행렬의 열
  - 열의 개수 : n_features
  - 관례적으로 피쳐행렬은 변수 X 에 저장
  - [n_samples, n_features] 형태의 2차원 배열 구조를 사용
    - (주로 numpy 배열, pandas DataFrame, SciPy 희소행렬을 사용)

Scikit-learn 의 데이터 표현방식

- 대상벡터(Target Vector)
  - 연속적인 수치값, 이산 클래스/레이블을 가짐
  - 길이 : n_samples
  - 관례적으로 대상벡터는 변수 y에 저장
  - 1차원 배열 구조를 사용(주로 numpy 배열, pandas Series 를 사용)
  - 피쳐행렬로 부터 예측하고자 하는 값의 벡터
  - 종속변수, 출력변수, 결과변수, 반응변수라고 함.

Scikit-learn Estimator API 기본 활용절차

1) 데이터 준비
2) 모델 클래스 선택
3) 모델 인스턴스 생성과 하이퍼파라미터 선택
4) 특징행렬과 대상벡터 준비
5) 모델을 데이터에 적합
6) 새로운 데이터를 이용해 예측
7) 모델 평가

훈련데이터와 테스트 데이터

- 훈련데이터와 테스트 데이터의 분리
  - 머신러닝 모델을 만들 때 사용한 데이터는 모델의 성능측정용으로 사용하지 않음. -> 일반화 문제
  - 훈련데이터
    - 머신러닝 모델을 만들 목적으로 사용
  - 테스트 데이터
    - 머신러닝 모델이 잘 작동하는지를 측정할 목적으로 사용
  - Scikit-learn 의 train_test_split 함수를 주로 사용
    - (기본적으로 훈련용 75%, 테스트용 25%로 구성)

Scikit learn 에서 제공되는 주요 모듈

분류	모듈명	설명
예제 데이터	sklearn.datasets	사이킷런에 내장되어 예제로 제공하는 데이터 세트
피처처리	sklearn.preprocessing	데이터 전처리에 필요한 다양한 가공 기능 제공(문자열을 숫자형 코드 값으로 인코딩, 정규화, 스케일링 등)
	sklearn.feature_selection	알고리즘에 큰 영향을 미치는 피처를 우선순위대로 셀렉션 작업을 수행하는 다양한 기능 제공
	sklearn.feature_extraction	텍스트 데이터나 이미지 데이터의 벡터화된 피처를 추출하는데 사용됨. 예를 들어 텍스트 데이터에서 Count Vectorizer나 Tf-Idf Vectorizer 등을 생성하는 기능 제공. 텍스트 데이터의 피처 추출은 sklearn.feature_extraction.text 모듈에, 이미지 데이터의 피처 추출 은 sklearn.feature_extraction.image 모듈에 지원 API가 있음
피처 처리 & 차원 축소	sklearn.decomposition	차원 축소와 관련한 알고리즘을 지원하는 모듈이다. PCA, NMF, Truncated SVD 등을 통해 차원 축소 기능을 수행할 수 있다.
데이터 분리, 검증 & 파라미터 튜닝	sklearn.model_selection	교차 검증을 위한 학습용/테스트용 분리, 그리드 서치(Grid Search)로 최적 파라미터 추출 등의 API 제공

Scikit learn 에서 제공되는 주요 모듈

분류	모듈명	설명
평가	sklearn.metrics	분류, 회귀, 클러스터링, 페어와이즈(Pairwise)에 대한 다양한 성능 측정 방법 제공 Accuracy, Precision, Recall, ROC-AUC, RMSE 등 제공
ML 알고리즘	sklearn.ensemble	앙상블 알고리즘 제공 랜덤 포레스트, 에이다 부스트, 그래디언트 부스팅 등을 제공
	sklearn.linear_model	주로 선형 회귀, 릿지(Ridge), 라쏘(Lasso) 및 로지스틱 회귀 등 회귀 관련 알고리즘을 지원. 또한 SGD(Stochastic Gradient Desccent) 관련 알고리즘도 제공
	sklearn.naïve_bayes	나이브 베이즈 알고리즘 제공. 가우시안 NB. 다항 분포 NB 등
	sklearn.neighbors	최근접 이웃 알고리즘 제공. K-NN(K-Nearest Neighborhood) 등
	sklearn.svm	서포트 벡터 머신 알고리즘 제공
	sklearn.tree	의사 결정 트리 알고리즘 제공
	sklearn.cluster	비지도 클러스터링 알고리즘 제공 (K-평균, 계층형, DBSCAN 등)
유틸리티	sklearn.pipeline	피처 처리 등의 변환과 ML 알고리즘 학습, 예측 등을 함께 묶어서 실행할 수 있는 유틸리티 제공