**머신러닝**

: 데이터를 기반으로 패턴을 학습하고, 이를 통해 미래의 데이터를 예측하거나 의사 결정을 자동화하는 인공지능(AI)의 한 분야. 대량의 데이터를 분석하여 알고리즘이 스스로 규칙을 찾아내는 방식으로 동작한다.

-> Data. Model. 성능지표

-전통적인 접근 방법 vs 머신러닝 접근 방법

|  |  |  |
| --- | --- | --- |
|  | 전통적 | 머신러닝 |
| 문제 연구 | 해결책을 찾음 | 데이터의 특징을 찾음 |
| 접근 방식 | 결정된 규칙으로 프로그램을 작성 | 주어진 데이터를 바탕으로 훈련 |
| 평가 | 만들어진 프로그램을 테스트 | 문제가 있을 시 오차 분석 후 처음부터 다시 실시 |
| 차이 | 새로운 규칙이 생겼을 때 사용자가 매번 업데이트해야함  -> 유지/보수가 어려움 | 별도의 작업을 하지 않아도 자동으로 인식/업데이트 |

=> 머신러닝은

1)기존 솔루션으로 많은 수동 조정과 규칙이 필요한 문제

2)전통적인 방식으로 해결 방법이 없는 복잡한 문제

3)유동적인 환경(새로운 데이터)

4)복잡한 문제, 대량의 데이터에서 통찰 얻기

에서 뛰어나다.

**머신러닝 시스템의 종류**

**1. 지도학습과 비지도학습**

**지도학습** : 데이터가 많은 경우. 훈련 데이터에 답을 포함

Ex)

분류 – 특성을 사용하여 데이터를 분류하는 문제. y값이 범주형 데이터

회귀 – 특성을 사용해 타깃 수치를 예측하는 문제. y값이 연속형 데이터

\*일부 회귀 알고리즘을 분류에 사용할 수 있고, 일부 분류 알고리즘을 회귀에 사용할 수 있다.

**비지도 학습** : 레이블이 없는 훈련 데이터를 이용하여 시스템 스스로 학습.

Ex)

군집화(clustering) : 데이터를 비슷한 특징을 가진 몇 개의 그룹으로 나누는 것

시각화와 차원축소 : 다차원 특성을 가진 데이터셋을 2D 또는 3D로 표현.

-> 상관관계가 있는 여러 특성을 하나로 합쳐 데이터 특성의 수를 줄임

-> 시각화를 하기 위해서는 데이터의 특성을 2가지로 줄여야 함

-> 차원 축소 > 시각화

-> 다중공산성: 상관관계가 큰 특성들 -> 해석이 어렵다(어떤 요소 때문인지 알기 어려움)

-> 차원축소의 저주: 차원은 작을수록 좋다(군집화를 해야하기 때문, 연산양/속도 이슈)

이상치 탐지 : 정상 샘플을 이용하여 훈련 후 특정한 기준을 바탕으로 입력 샘플의 정상여부 판단.

특이치 탐지 : 전혀 오염되지 않은(이상치가 없다고 가정한) 훈련 세트 활용, 그 경계를기준으로 훈련 세트에 포함된 데이터와 다른 데이터를 감지

**준지도 학습** : 적은 수의 샘플에 레이블을 적용하여 비지도 학습을 통해 군집을 분류한 후, 샘플들을 활용해 지도 학습에 활용함. (y값이 없는 데이터들에게 예측한 y값을 넣고 이를 가지고 학습(라벨링))

**강화 학습** : 에이전트가 환경을 관찰하여 행동을 실행하고 그 결과로 보상 혹은 벌점을받는다. -> 최종 결과물이 정책(최상의 전략)

**2. 배치 학습과 온라인 학습**

**배치 학습** : 주어진 훈련 세트 전체를 사용하여 오프라인 훈련.

-> 컴퓨팅 자원이 충분한 경우에만 사용 가능

-> 새로운 데이터가 들어오면 처음부터 새롭게 학습해야 함

**온라인 학습** : 적은 양의 데이터(미니배치)를 사용해 점진적으로 훈련.

-> 실시간으로 데이터가 들어옴.

-> 나쁜 데이터가 주입되는 경우 시스템 성능이 점진적으로 떨어질 수 있음.

지속적인 시스템 모니터링 필요

\*일반화 : 새로운 데이터에 대한 예측을 잘한다는 의미

**사례 기반 학습** : 샘플을 기억하는 것이 훈련의 전부. 예측을 위해 기존 샘플과의 유사 도

측정

-> 함수가 아닌 데이터 자체로만 비교.

-> 데이터가 많을수록 성능 높아짐. 고차원 데이터에서 성능 안좋음.

선형 모델 : 데이터를 대표하는 하나의 직선(선형 모델)을 찾기

**모델 기반 학습** : 모델을 미리 지정한 후 훈련 세트를 사용하여 모델을 훈련시킴(본인이 함수를 결정해야함.). 훈련된 모델을 사용해 새로운 데이터에 대한 예측 실행

효용 함수 : 모델이 얼마나 좋은지 측정

비용 함수 : 모델이 얼마나 나쁜지 측정

**머신러닝의 주요 도전 과제**

**1. 충분하지 않은 양의 훈련 데이터**

간단한 문제여도 수천 개의 데이터가 필요함.

-> 데이터가 부족하면 알고리즘 성능 향상 어려움

**2. 대표성 없는 훈련 데이터**

샘플링 잡음 : 우연에 의해 대표성이 없는 데이터. 데이터가 많을수록 잡음이 줄어듦

샘플링 편향 : 표본 추출 과정에서 잘못된 대표성이 없는 데이터

**3. 낮은 품질의 데이터**

이상치 샘플이라면

-> 고치거나 무시

특성이 누락되었다면

-> 해당 특성을 제외

-> 해당 샘플을 제외

-> 누락된 값을 채움

-> 해당 특성을 넣은 경우와 뺀 경우 각기 모델을 훈련

**4. 관련이 없는 특성**

특성 공학 : 풀려는 문제에 관련이 높은 특성 찾기.

특성 선택 : 준비되어 있는 특성 중 가장 유용한 특성을 찾음

특성 추출 : 특성을 조합하여 새로운 특성을 만듦

**5. 훈련 데이터 과대적합**

훈련 세트에 특화되어 일반화 성능이 떨어지는 현상

**6. 훈련 데이터 과소적합**

모델이 너무 단순해서 훈련세트를 잘 학습하지 못함

-> 해결 방법 : 모델 파라미터가 더 많은 복잡한 모델을 사용/특성 공학으로 더 좋은 특성을 찾음/규제의 강도를 줄임

**테스트와 검증**

**1. 검증**

훈련된 모델의 성능 평가 : 테스트 세트 활용

검증 기준 : 일반화 오차(테스트 데이터 정확도)

-> 새로운 샘플에 대한 오류 비율

-> 학습된 모델의 일반화 성능의 기준

과대 적합 : 훈련 오차에 비해 일반화 오차가 높은 경우

**2. 하이퍼파라미터 튜닝과 모델 선택**

하이퍼파라미터 : 알고리즘 학습 모델을 정의하는데 사용되는 파라미터. 함수를 골랐을 때 함수의 특성을 부여하는 것 (ax+b 에서 a,b)

->훈련 과정에서 변하는 파라미터가 아님

->하이퍼파라미터를 조절하면서 가장 좋은 성능의 모델을 선정

**3. 교차검증**

검증 세트(홀드아웃 검증) : 훈련 세트의 일부로 만들어진 데이터셋

-> 다양한 하이퍼파라미터 값을 후보 모델 평가용으로 예비표본을 검증세트로 활용하는 기법

교차검증 : 여러 개의 검증세트를 사용한 반복적인 예비표본 검증 적용 기법. 일반화된 모델.

-> 교차 검증 후 모든 모델의 평가를 평균하면 훨씬 정확한 성능 측정 가능

-> 훈련 시간이 검증 세트의 개수에 비례해 늘어남. 연산량 많음.