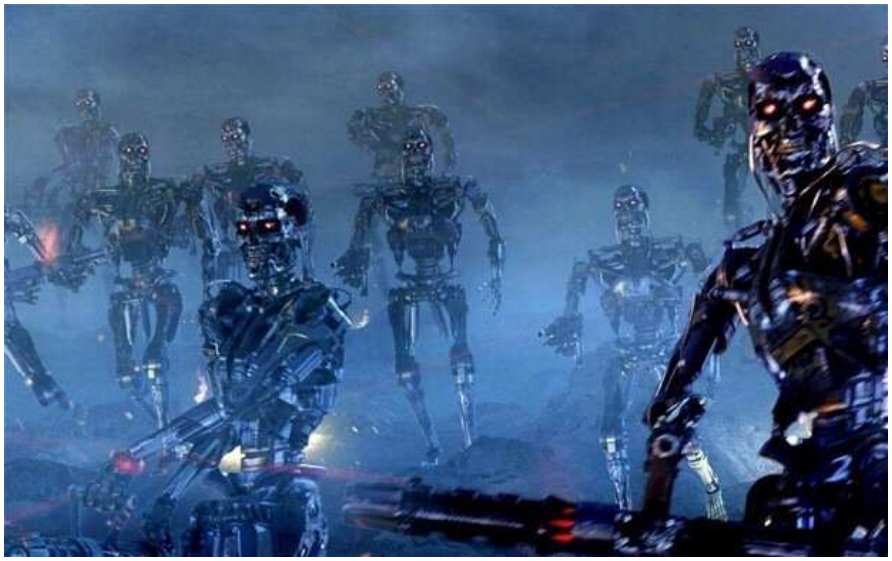
머신러닝 개념

목차

- 인공지능이란 무엇인가?
- 머신러닝 vs 딥러닝
- 딥러닝 멀리서 보기
- 딥러닝의 간단한 예시
- 언더 피팅 vs 오버 피팅
- 모델 선택 알고리즘
- 규제
- 머신러닝의 유형

인공지능이란 무엇인가?

- 인간에게 위협이 되는 존재?



https://www.bbc.com › korean › international-50231631

인공지능은 의도치 않게 인류를 파괴할 수 있다 - BBC

2019. 11. 3. — 우리 존재를 위협할 수 있는 것은 의식을 발전시킨 로봇이 아니다. ... 이 예는 AI가 인간이 미처 모든 걸 다 고려하지 못한 지시에 따라 행동할 때 ...

https://www.jobkorea.co.kr › goodjob › tip › view

[이슈&논술] 인공지능(AI)의 위협과 대응 방안 - 잡코리아

2020. 7. 27. — AI가 인간의 모든 지식 노동을 대체해 사람은 먹고 자는 것 이외에는 할 일이 없어진다는 것이다. 최근 AI의 발전 속도를 보면 이 예언은 더 빨리 적중할 ...

https://www.mk.co.kr › news › view › 2021/05

[위클리 스마트] 인공지능은 결국 인류를 위협할까 - 매일경제

2021. 5. 8. — 가장 높은 '용납될 수 없는 위험'은 인간의 안전·생계·권리에 위협이 되는 AI 시스템으로, 아예 개발 자체를 금지했다.

https://www.etri.re.kr › webzine › sub01

Special ____ 인공지능은 사람의 일자리를 위협할까?

인공지능이 발달하면서 "인간 중심으로 이뤄지던 일자리는 바뀌지 않을까?"란 우려는 어느 정도 생각해 볼 만 하다. 컴퓨터 보급으로 인한 단순 사무직 종사자나 대형마트 ...

인공지능이란 무엇인가?

- 인공지능에 대한 사전적 정의

: 인간의 학습능력, 추론능력, 지각능력, 그 외에 인공적으로 구현한 컴퓨터 프로그램 또는 이를 포함한 컴퓨터 시스템.

인공지능이란 무엇인가?

- 인공지능은 두가지로 분류됨

  - 강(强)인공지능: 터미네이터, 울트론처럼 사람의 도움 없이 스스로 사고하고 판단하며 학습하는 인공지능

  - 약(弱)인공지능: 사람이 제공한 데이터에 기반하며, 데이터로 학습한 후, 학습 데이터와 비슷한 입력에 대해 결과를 도출.

인공지능이란 무엇인가?

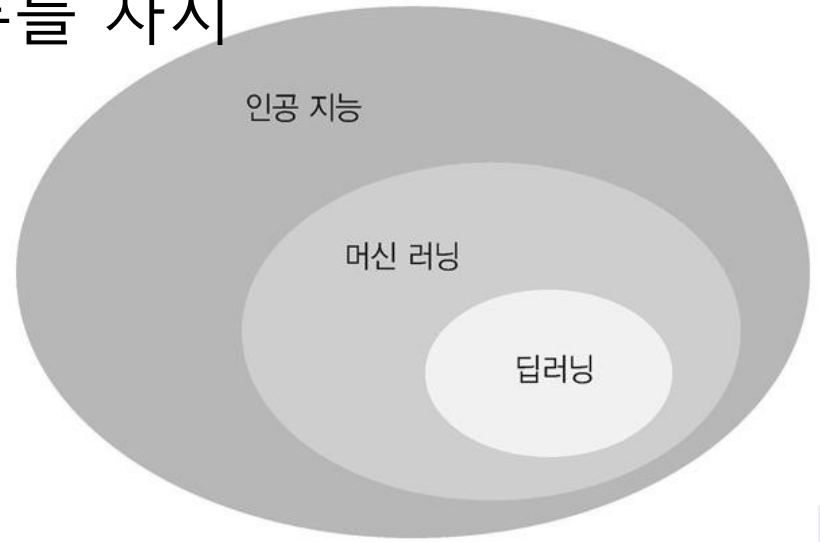
- 현대의 기술은 약인공지능을 구현하는 데 그쳐 있으며,
  이번 강의를 통해 간단하게 배워볼 것도 약인공지능에 해당.

머신러닝(ML) vs 딥러닝(DL)

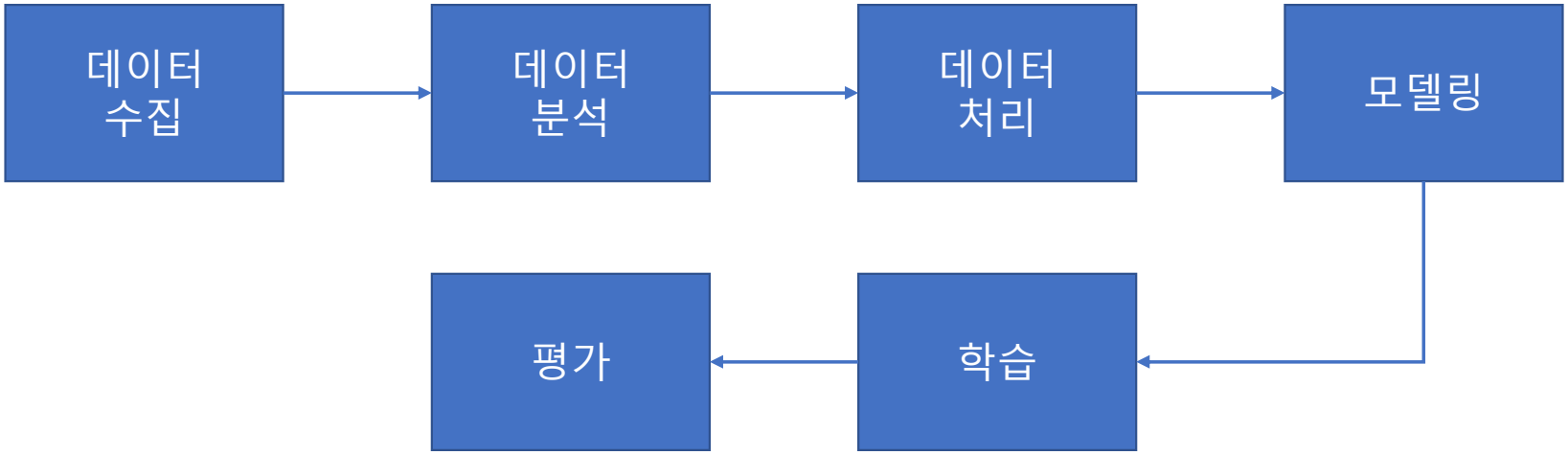
- 여러 매체를 접하다 보면 인공지능 관련 분야에서 머신러닝과 딥러닝을 혼용해서 사용하는 경우가 많음.

머신러닝(ML) vs 딥러닝(DL)

- 딥러닝은 인공지능 중 약인공지능,
  약인공지능 중 머신러닝의 여러 알고리즘 중 한 알고리즘.
  크기가 큰 데이터에 대해 높은 성능을 보이기 때문에
  현 인공지능 분야에서 주류를 차지



딥러닝 멀리서 보기



딥러닝 멀리서 보기

1. 데이터 수집: 학습에 필요한 데이터를 모으는 과정
2. 데이터 분석: 통계학적 방법이나 또는 직관 등을 이용해서 데이터의 구성을 파악하는 과정
3. 데이터 처리: 데이터를 학습에 적합한 형태로 변환하는 과정
4. 모델링: 데이터로 학습시킬 모델을 정의하는 과정
5. 학습: 데이터로 모델을 학습시키는 과정
6. 평가: 학습이 잘 되었는지 모델을 평가하는 과정

딥러닝의 간단한 예시

- 딥러닝이 하고자 하는 건 다음과 같다.
  아래와 같은 직선에서 x는 입력을 의미하고,
  y는 출력, W와 b는 매개변수를 의미한다.

$$y = Wx + b$$

x=[1,2,3,4], y=[3,5,7,9]가 주어졌을 때
x=[5,6]에 대해 y=[11,13]을 예측하는 직선을 찾아보자.

딥러닝의 간단한 예시

- 사실 눈으로 딱 보기에 $y = 2x + 1$이 정답이겠지만, 컴퓨터 입장에서 무작정 찾아보자.

W	b	x=[1,2,3,4], y=?
0	0	[0,0,0,0]
1	0	[1,2,3,4]
1	1	[2,3,4,5]
2	0	[2,4,6,8]
2	1	[3,5,7,9]

- W=2, b=1일 때 기존에 주어진 데이터 x=[1,2,3,4], y=[3,5,7,9]가 가장 잘 표현된다.

딥러닝의 간단한 예시

- 그렇다면 W=2, b=1일 때 처음에 우리가 원했던 x=[5,6]에 대해 y=[11,13]을 예측하는 지 확인해보자.

W	b	x=[5,6], y=?
2	1	[11,13]

- 정확하게 들어맞는다.

딥러닝의 간단한 예시

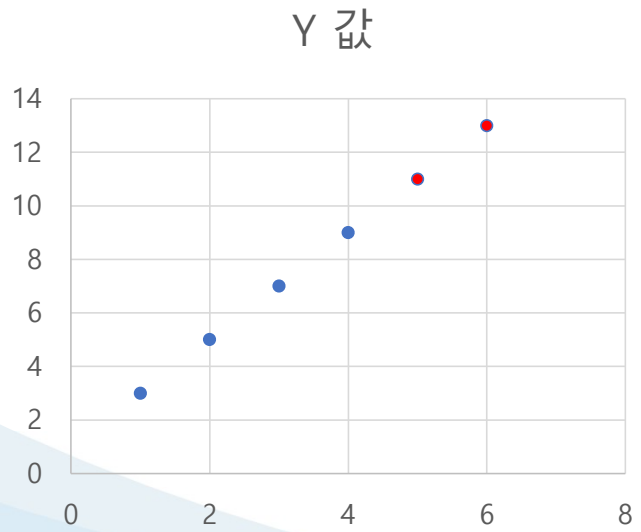
- 우리가 한 과정을 용어를 바꿔서 표현해 보자.

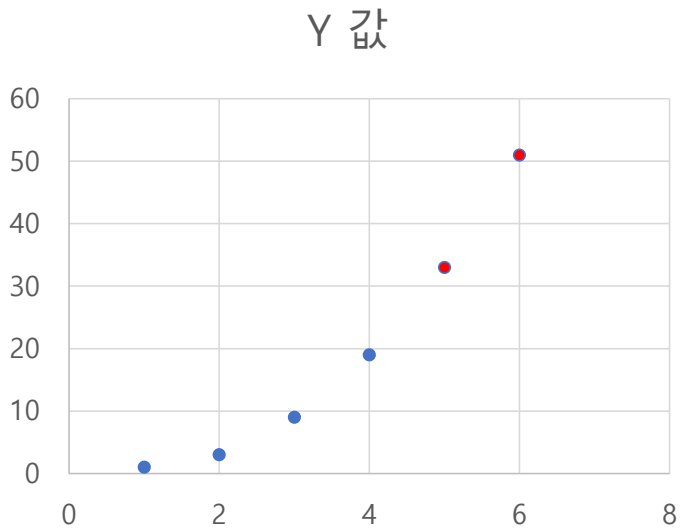
- 학습 데이터: x=[1,2,3,4], y=[3,5,7,9]
  모델: $y = Wx + b$
  매개변수: W, b
  테스트 데이터: x=[5,6], y=[11,13]

우리가 한 과정은 **테스트 데이터**를 가장 잘 표현하는 **모델**의 **매개 변수**를 **학습 데이터**를 이용하여 찾아낸 것이다.

딥러닝의 간단한 예시

- 방금의 학습 데이터는 좌측과 같이 충분히 직선으로 표현할 수 있었다. 그러나 우측의 경우 직선으로 표현하긴 무리가 있다.





AIoT

딥러닝의 간단한 예시

- 학습 데이터 x=[1,2,3,4], y=[1,3,9,19]를 더 잘 표현하기 위해 다음과 같은 2차원 모델을 정의하자.
  이번 테스트 데이터는 x=[5,6], y=[33,51]이다.

$$y = W_1 x^2 + W_2 x + b$$

마찬가지로 해야 할 일은 학습 데이터를 사용하여 테스트 데이터를 가장 잘 표현하는 모델의 매개변수 (W_1,W_2,b)를 찾는 것이다.
직선 모델에 비해 매개변수가 하나 늘었다.

딥러닝의 간단한 예시

- 매개변수를 찾아보면 아래 방정식이 최적의 해가 된다.

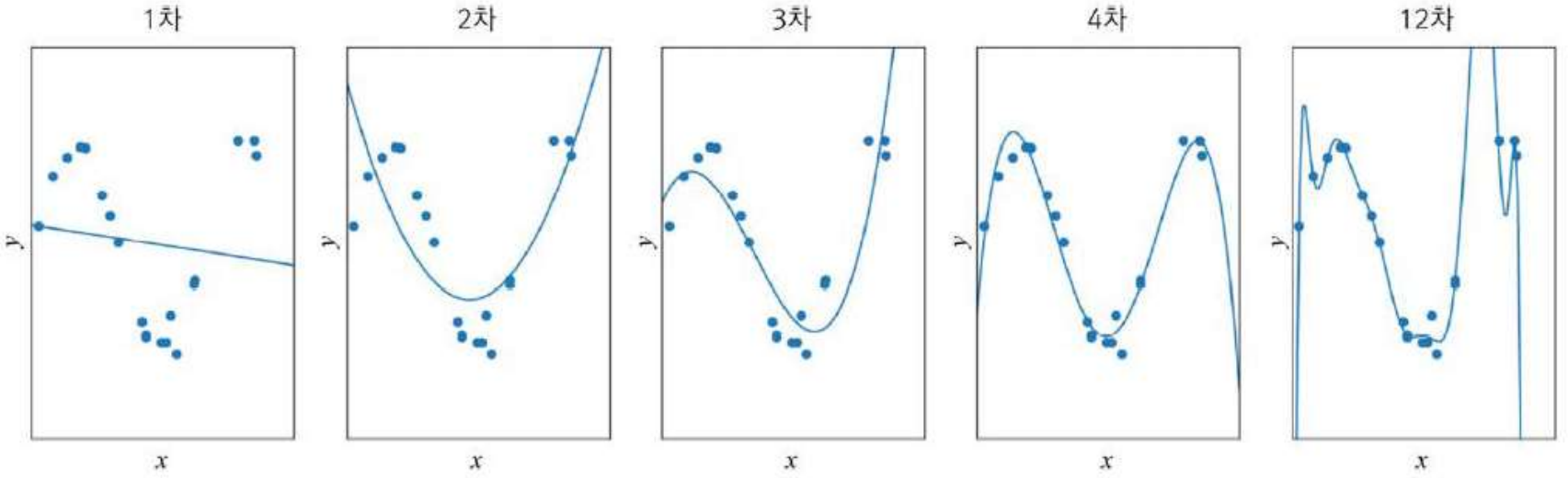
$$y = 2x^2 - 4x + 3$$

이처럼 모델이 복잡해 질수록(차수가 높아질 수록)
더욱더 최적의 해를 찾기 쉬워진다.

그렇다면 무작정 모델을 복잡하게 만들면 되지 않을까?

언더피팅 vs 오버피팅

- 언더피팅(Underfitting, 과소적합): 모델의 용량이 작아(단순하여) 데이터를 효과적으로 표현하지 못함.
  모델의 용량을 크게(복잡하게) 구현하면 해결

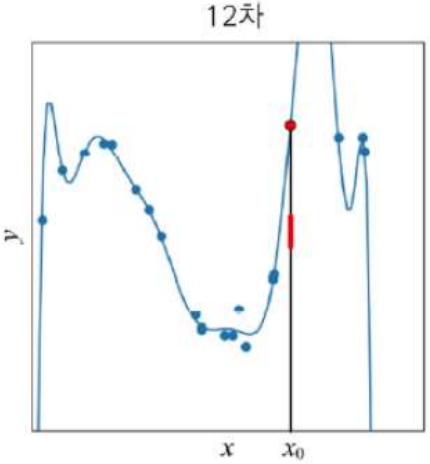


AIoT

언더피팅 vs 오버피팅

- 오버피팅(Overfitting, 과잉적합): 실제 세계의 데이터는 이전의 *딥러닝의 간단한 예시*에서 처럼 딱 떨어지지 않음.
  모델을 복잡하게 구성하여 학습 데이터는 정확하게 표현하지만 테스트 데이터에 대해서는 큰 오차가 발생.

  오른쪽 사진에서 테스트 데이터(x_0)에대해 빨간 막대를 예측해야 하지만 빨간 점을 예측해 오차가 크게 발생.



AIoT

언더피팅 vs 오버피팅

- 실제 학습을 진행하면 언더피팅 보다는 오버피팅이 자주 발생. 오버피팅을 방지하기 위한 여러가지 방법 적용 필요.

- 모델 선택 알고리즘
  - 검증
  - 교차 검증
- 규제
  - 데이터 증강
  - 가중치 감쇠

모델 선택 알고리즘

- 검증(Validation)
  학습 데이터의 일부를 검증 데이터로 나누어
  학습이 끝나고 모델을 검증 데이터로 평가.
  검증 데이터 기준에서 가장 학습이 잘 된 모델을 최종 선택.

모델 선택 알고리즘

학습 데이터

테스트 데이터

학습 데이터	검증 데이터	테스트 데이터
1	2	3

validation

```
best_model=None
best_score=-infinity
for model in models:
  1로 model 학습
  2로 model 평가
  if model의 score > best_score:
    best_score=model의 score
    best_model=model
3으로 best_model 평가
```

AIoT

모델 선택 알고리즘

- 검증방법의 문제
  테스트 데이터의 분포가 검증 데이터와 다르면
  검증 데이터의 평가 결과가 테스트 데이터와 연관 있다고 볼 수 없음.
  => 교차 검증 사용

모델 선택 알고리즘

- 교차 검증(Cross Validation)

cross validation

```
best_model=None
best_score=-infinity
for model in models:
  k=5
  model_scores=[]
  for i in [1,2,3,...,k]:
    i를 검증 데이터, 나머지를 학습 데이터로 지정
    학습 데이터로 model 학습
    model_score=i로 model 평가
    model_scores.append(model_score)
  if mean(model_scores)>best_score:
    best_model=model
    best_score=mean(model_scores)
6으로 best_model 평가
```

AIoT

규제

- 가중치 감쇠(Weight Decay): 한 가중치가 너무 커져버리면 모델의 출력이 입력의 특정 형태에 의존하게 되는 경향이 발생. 아래 수식에서 다음과 같은 결과가 발생 => 가중치를 되도록 작게

$$y = W_1 x^2 + W_2 x$$

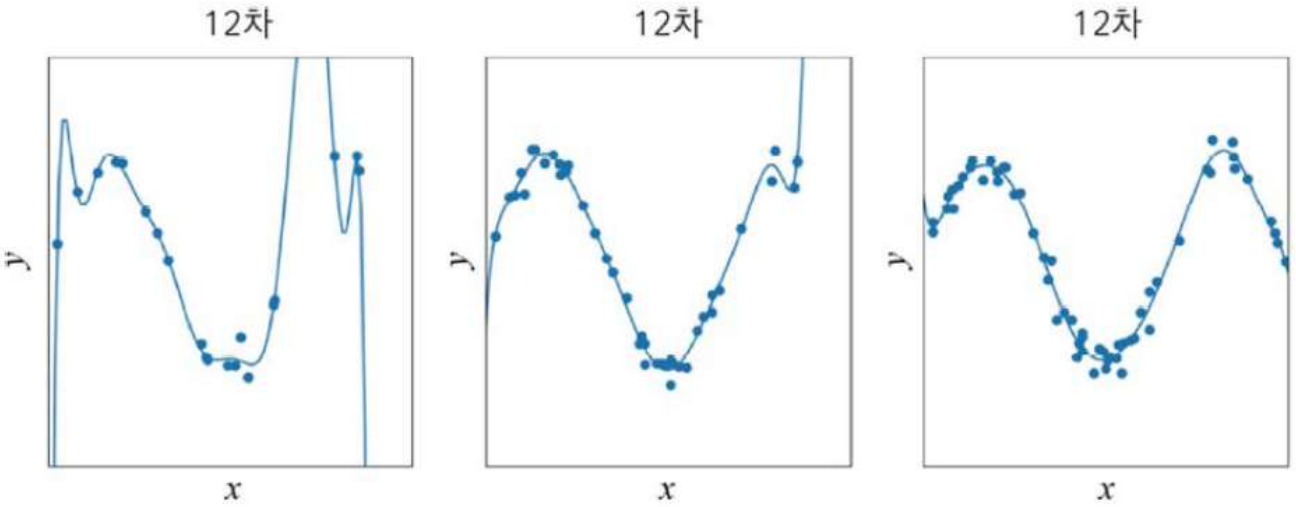
입력 X=[-2,-1,0,1,2]

W1	W2	X=[-2,-1,0,1,2], y=?	
1	1	[2,0,0,2,6]	
1	-10	[24,11,0,-9,-16]	입력의 부호에 따라 크게 영향
1	10	[-16,-9,0,11,24]	입력의 부호에 따라 크게 영향
5	5	[10,0,0,10,30]	입력의 절대값에 따라 크게 영향
1000	1000	[2000,0,0,2,6000]	입력의 절대값에 따라 크게 영향

AIoT

규제

- 아래의 그림처럼 데이터가 많아지면 오버피팅을 많이 줄일 수 있음.



(a) 훈련집합의 크기=20 (b) 훈련집합의 크기=40 (c) 훈련집합의 크기=60

규제

- 데이터 증강(Data Augmentation): 비용의 문제로 데이터를 추가하는 데는 한계가 있기 때문에 기존의 데이터를 일부 변형하여 새로운 데이터 생성.

AIoT

머신러닝의 유형

- 지도 방식에 따라
  지도 학습, 비지도 학습, 준지도 학습, 강화 학습으로 분류

- 지도 학습(Supervised Learning): X와 Y가 존재
- 비지도 학습(Unsupervised Learning): X만 존재
- 준지도 학습(Semi-supervised Learning):
  일부 데이터는 X와 Y 둘다 존재, 나머지 일부는 X만 존재
- 강화 학습: 특정 환경에서 보상을 최대화하는 쪽으로 학습

실습 환경 소개 – Google Colab

- 구글에서 제공하는 클라우드 기반의
  무료 Jupyter Notebook 개발 환경.

- 무료로 고사양의 GPU를 사용가능, 딥러닝 관련 라이브러리
  (Pytorch, Tensorflow, Keras, Scikit-Learn 등)가 설치 되어있음.

- 구글 드라이브를 간편하게 마운트(연결)하여 사용할 수 있음

실습 환경 소개 – Google Colab

- 무료로 사용하는 만큼 여러 단점도 존재.
- 세션(런타임)의 최대 유지시간은 12시간이기 때문에 세션이 끊기면 로컬 변수나 드라이브에 저장되지 않은 파일은 사라짐
- 학습을 진행하다 보면 몇시간은 기본으로 넘는 경우가 많은데, 가끔 할당된 GPU 사용 가능 시간이 학습 시간을 넘지 못해 학습 중간에 CPU 모드로 바뀌곤 함 -> 울며 겨자먹기로 유료플랜

Review

- 인공지능이란 무엇인가?
  강인공지능 vs 약인공지능
- 머신러닝 vs 딥러닝
  딥러닝은 머신러닝의 한 분야
- 딥러닝 멀리서 보기
  데이터 수집 및 가공, 학습, 평가
- 딥러닝의 간단한 예시
  용어
  모델의 복잡도와 학습 정확도
- 언더 피팅 vs 오버 피팅
  오버피팅을 해결하는 것이 관건
- 모델 선택 알고리즘
  검증, 교차 검증
- 규제
  가중치 감쇠, 데이터 증강
- 머신러닝의 유형
  지도, 비지도, 준지도, 강화

2장 Preview

- 데이터를 컴퓨터에서 표현하는 방법 – 벡터, 행렬, 텐서
- 무작정 대입하지 않고 매개변수를 찾는 효과적인 방법 – 최적화

수고하셨습니다!