딥러닝의 통계적이해

2023 출석수업

한국방송통신대 이긍희 교수

1. 인공지능의 시대

인공지능

- 인간지능(Artificial Intelligence) : 인간의 지능을 컴퓨터로 구현
  - 추론·탐색, 지식의 연결, 머신러닝·딥러닝 순으로 발전

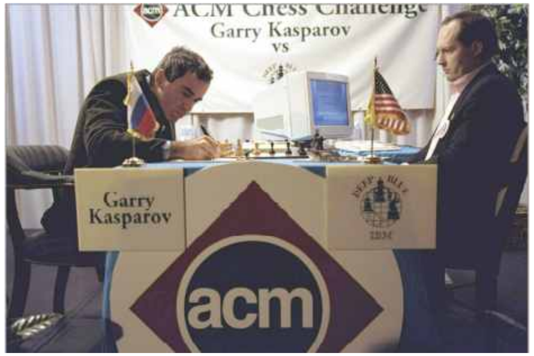
Dartmouth Conference(1956)

1. 인공지능의 시대

딥러닝의 통계적이해

2023 출석수업

Deep Blue(1997)



출처 : getty images

1. 인공지능의 시대

딥러닝의 통계적이해

2023 출석수업

IBM Watson(2011) : Jeopardy!

출처 : https://www.youtube.com/watch?v= P18EdAKu C1U

1. 인공지능의 시대

딥러닝의 통계적이해

2023 출석수업

AlphaGo(2016)

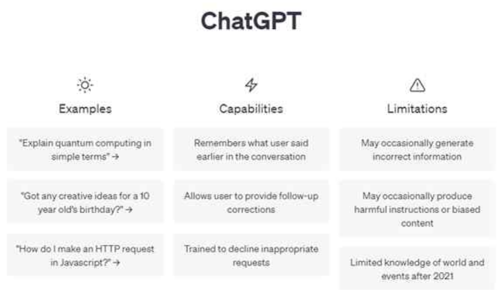
출처 : alphagomovie.com

1. 인공지능의 시대

딥러닝의 통계적이해

2023 출석수업

ChatGPT(2022)



딥러닝의 통계적이해
2023 출석수업

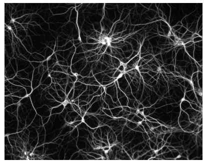
2. 신경망

2. 신경망

딥러닝의 통계적이해
2023 출석수업

인간의 뇌

- 인간의 뇌 : 1.5Kg, 860억개 뉴런, 시냅스로 연결
  - 뇌는 뉴런의 연결망



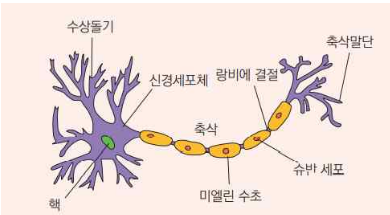
출처 : https://namu.wiki/w/%EB%89%B4%EB%9F%B0

2. 신경망

딥러닝의 통계적이해
2023 출석수업

뉴런

- 뉴런의 구조
  - 시냅스 : 축색말단과 다른 뉴런이 접하는 부분



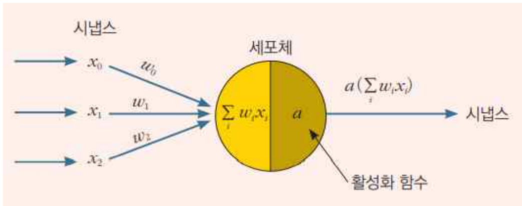
출처 : 위키피디아 <뉴런>

2. 신경망

딥러닝의 통계적이해
2023 출석수업

신경망의 구조

- 신경망 : 인간의 신경망을 모방



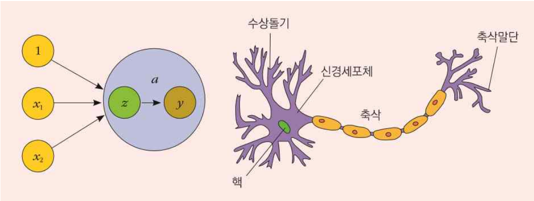
2. 신경망

딥러닝의 통계적이해

2023 출석수업

활성화 함수

- 활성화 함수(activation function) : 시냅스 구현 함수
  - 뉴런 정보가 시냅스로 연결 → 화학물질 이용, 전기 정보가 임계값을 넘었을 때 활성화 → 다른 뉴런 전달



2. 신경망

딥러닝의 통계적이해

2023 출석수업

활성화 함수의 종류

- 활성화 함수의 종류

항등함수 : $a(x)=x$

시그모이드 함수 : $a(x)=\dfrac{1}{1+e^{-x}}$

tanh 함수 : $a(x)=\dfrac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$

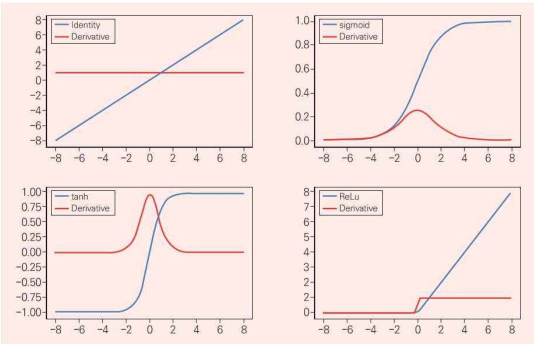
ReLU 함수 : $a(x)=\max(x,\ 0)$

2. 신경망

딥러닝의 통계적이해

2023 출석수업

활성화 함수의 종류



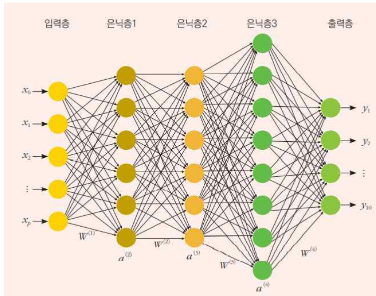
2. 신경망

딥러닝의 통계적이해

2023 출석수업

신경망의 구조

- 신경망은 뉴런과 네트워크(network)로 구성



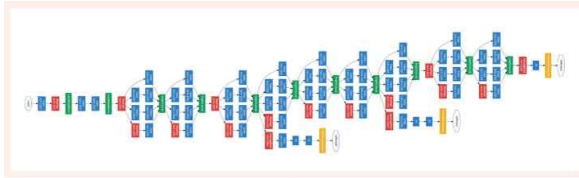
층(layer)

- 뉴런들이 모여서 층(layer)을 이룸
  - 입력층(input layer) : 정보가 입력되는 층
  - 출력층(output layer) : 예측하거나 분류하는 층
  - 은닉층(hidden layer) : 입력층과 출력층 사이의 층

- 네트워크 : 층 내의 뉴런은 서로 연결되지는 않고 아래층, 위층과 연결
- 가중치(weight) : 네트워크의 연결강도

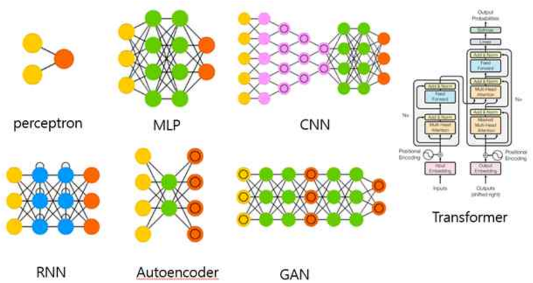
딥러닝 모형

- 딥러닝 모형 : 은닉층의 수가 많은 신경망



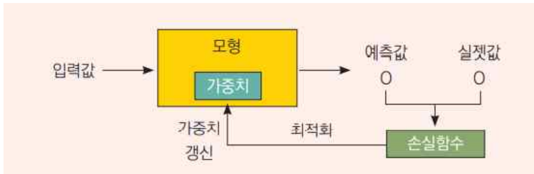
출처: Szegedy et al.(2015)

신경망의 종류



신경망의 작성

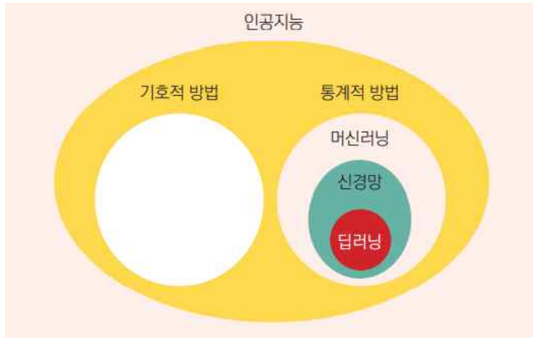
- 신경망의 작성과정은 머신러닝 모형과 동일



딥러닝의 통계적이해
2023 출석수업

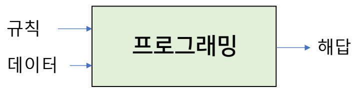
3. 머신러닝

머신러닝과 딥러닝



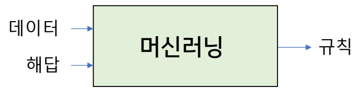
머신러닝의 정의

- 컴퓨터 : 문제를 수식화·논리화 → 규칙
  - 코딩을 통해 프로세서를 제어해서 문제를 해결
  - 규칙이 많아질수록 성능이 낮아지는 경향



머신러닝의 정의

- 머신러닝(기계학습) : 컴퓨터가 데이터로부터 스스로 학습하여 패턴(규칙)을 찾아내어 과제를 수행



머신러닝의 구성

- 머신러닝 : 과제, 데이터, 모형, 손실함수, 최적화 알고리즘

  - 과제 : 분류와 예측으로 구분
  - 데이터 : 입력 데이터와 출력데이터(label)로 구분
  - 모형 : 확률모형과 알고리즘 모형
  - 손실함수 : 머신러닝의 성과함수

3. 머신러닝

딥러닝의 통계적이해
2023 출석수업

머신러닝의 구성

- 최적화 알고리즘 : 최소제곱법, 최대가능도추정법, 경사하강법
  - · 학습(learning) : 손실함수가 최소가 되도록 데이터를 기반으로 최적화 알고리즘으로 머신러닝 모형을 지속적 수정

3. 머신러닝

딥러닝의 통계적이해
2023 출석수업

머신러닝의 학습방법

- 머신러닝의 대부분은 성과가 높은 지도 학습, 인간 지능의 대부분은 비지도 학습
  - 지도 학습 (supervised learning) : 예측
  - 비지도 학습(unsupervised learning) : 군집화
  - 강화 학습(reinforcement learning) : 보상액이 최대가 되도록 행동하고 시행착오를 통해 목표에 도달하는 학습

3. 머신러닝

딥러닝의 통계적이해
2023 출석수업

머신러닝의 구분

2. 신경망

딥러닝의 통계적이해
2023 출석수업

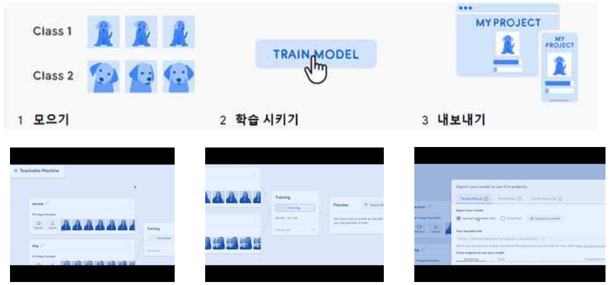
머신러닝과 딥러닝

- 딥러닝 모형은 데이터의 패턴을 계층적 네트워크로 이해할 수 있도록 설계된 신경망
  - 신경망의 층이 쌓아지면서 입력 데이터 보다 추상화

3. 머신러닝

딥러닝의 통계적이해
2023 출석수업

Teachable Machine



출처 : https://teachablemachine.withgoogle.com/

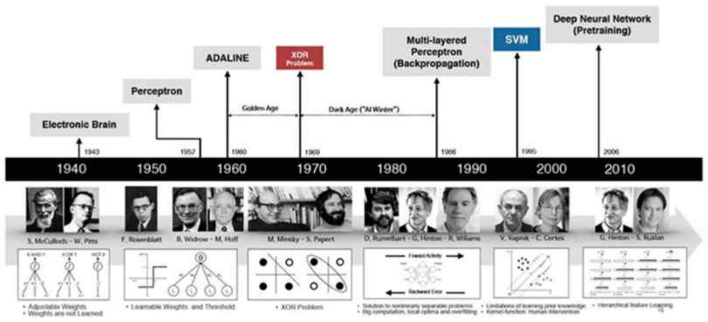
딥러닝의 통계적이해
2023 출석수업

4. 딥러닝의 역사

4. 딥러닝의 역사

딥러닝의 통계적이해
2023 출석수업

딥러닝 역사



https://www.futurespace.es/en/redes-neuronales-y-deep-learning-capitulo-1-preludio/

딥러닝 역사

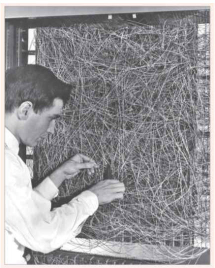
년도	내용	인물
1943	Neural networks	Pitts & Mcculloch
1956	Dartmouth Summer Research Project	AI
1957	Perceptron	Roseblatt & Wightman
1960	ADALINE	Widrow & Hoff
1969	XOR problem	Minsky & Papert
1986	backpropagation, RBM, RNN	Rumelhart, Hinton & Williams, Smolensky, Rumelhart
1989	CNN	LeCun
1997	LSTM	Hochreiter & Schmidhuber

딥러닝 역사

년도	내용	인물
2006	Deep Blief Network	Hinton, Osindo, Tech
2012	AlexNet	Krizhevsky 등
2014	DeepFace, GAN	FaceBook, Goodfellow 등
2016	AlphaGo	Deepmind
2017	Transformer	Vaswani 등
2018	BERT/GPT-1	Google/Open AI
2020	GPT-3	Open AI
2021	DALL·E	Open AI
2022	Stable Diffusion/Instruct GPT, ChatGPT,	CompVis/Open AI
2023	Bard/LLaMA/GPT-4	Meta/Google/Open AI

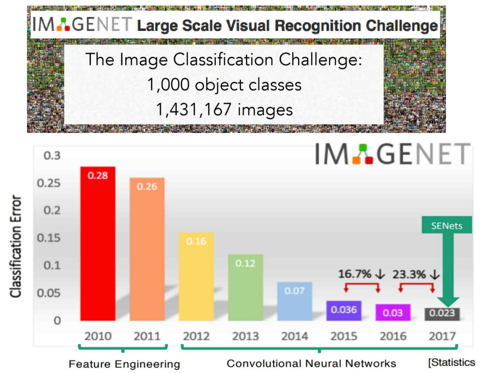
퍼셉트론

- 1958년 로젠블랫(F. Rosenblatt) : 이미지 인식 기계인 퍼셉트론



출처 : getty images

IMAGENET의 경진대회

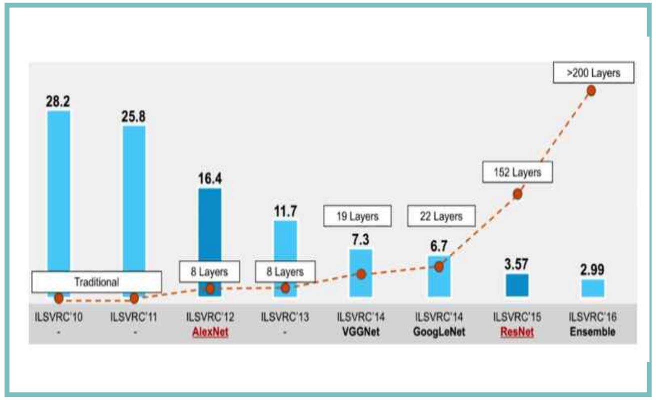


4. 딥러닝의 역사

딥러닝의 통계적이해

2023 출석수업

IMAGENET의 경진대회

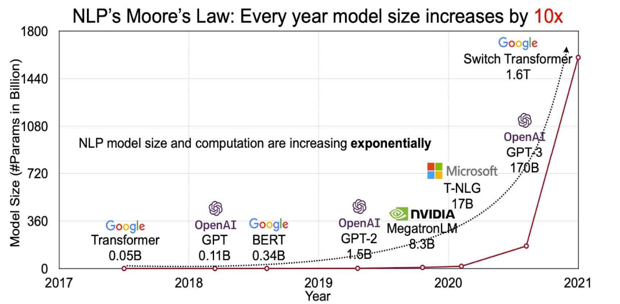


4. 딥러닝의 역사

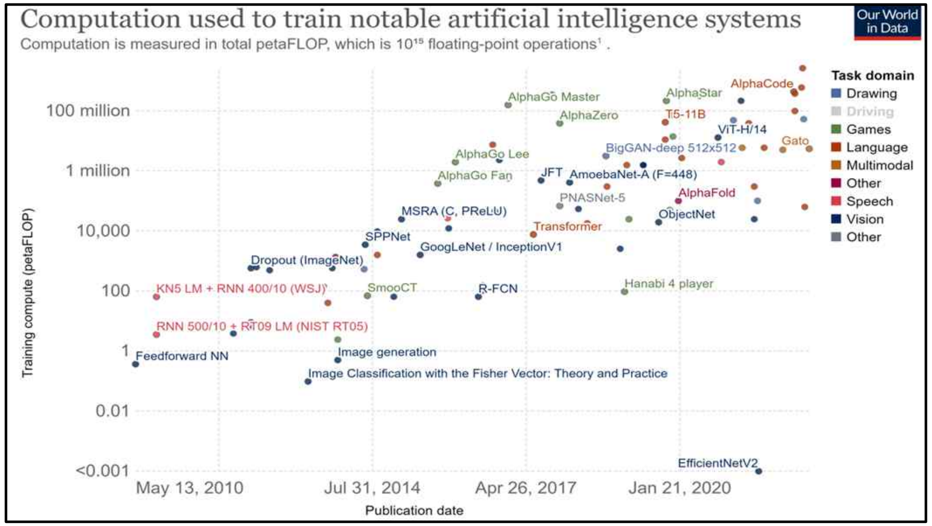
딥러닝의 통계적이해

2023 출석수업

LLM



출처 : https://indiaai.gov.in/article/the-future-of-large-language-models-llms-scope-opportunities-and-challenges

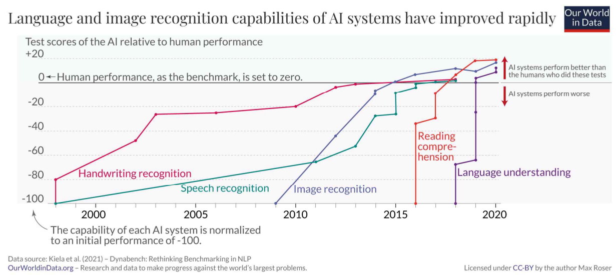


4. 딥러닝의 역사

딥러닝의 통계적이해

2023 출석수업

I



출처 : https://ourworldindata.org/uploads/2022/12/AI-performance_Dynabench-paper-2048x921.png

딥러닝의 통계적이해
2023 출석수업

5. 통계학과 딥러닝

통계학

- 통계학 : 데이터를 통해 배우는 과학
  - 통계적 추론 : 작은 수의 데이터로 모집단 일반화
  - 과학연구, 신약개발, 여론조사, 품질관리 등 성공
- 레이블(답)이 있는 데이터가 많아지고 컴퓨팅 능력이 향상
  - 수학 중심 통계학 → 알고리즘 기반 머신러닝

머신러닝

- 레이블 + 입력 데이터 → 머신러닝 모형 → 적절한 절차
  - 절차를 기반으로 새로운 데이터로 분류 또는 예측
- 인공지능 : 1980년대 후반 이후 통계학의 방법론을 활용
  - 컴퓨팅 능력과 데이터가 부족
    → 딥러닝·머신러닝 : 확률분포, 통계학을 활용
  - 이후 데이터 기반 머신러닝, 딥러닝으로 진화

용어 비교

통계학	딥러닝·머신러닝
모수(parameter)	가중치(weights)
추정(estimation) 적합(fitting)	학습(learning)
회귀 또는 분류	지도학습
군집화, 분포 추정	비지도학습
독립(설명)변수	특징
종속(반응)변수	레이블

통계모형과 딥러닝 모형

- 통계모형 : 확률분포 기반으로 추정과 가설검정을 통해 결과의 원인을 설명하는 것이 주목적
- 딥러닝 모형 : 블랙박스(black box) → 결과의 예측 목적

딥러닝의 통계적이해
2023 출석수업

통계모형과 딥러닝 모형

	전통적 통계모형	딥러닝 모형
데이터 크기	소규모	대규모
모형의 구조	데이터 = 생성구조 + 오차	오차가 주요 문제가 아님
모형의 크기	데이터 생성구조를 저차원 모형으로 파악	데이터 생성구조가 복잡 → 고차원 모형(블랙박스)
모형의 평가	적합도, 유의성	예측력
모형의 작성	생성구조를 오차에서 분리	학습을 통해 데이터의 복잡한 특성을 이해

딥러닝의 통계적이해
2023 출석수업

6. 회귀모형

딥러닝의 통계적이해
2023 출석수업

선형 회귀모형

- 설명변수들로 종속변수를 선형적으로 예측하는 모형

  - $y_i = w_0 + w_1 x_i + \varepsilon_i, \quad \varepsilon_i \sim N(0, \sigma^2)$
  - w_0, w_1 : 최소제곱법과 최대가능도추정법으로 추정

최소제곱법

- 손실함수 최소화하는 w_0, w_1를 구함
  - 손실 함수 : $J(w) = \frac{1}{2n}\sum_{i=1}^{n}(w_0 + w_1 x_i - y_i)^2$
  - w_0와 w_1에 대해 미분

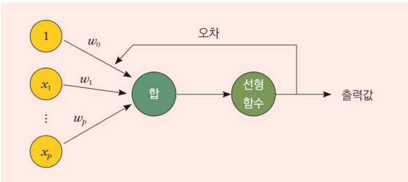
$$\rightarrow \widehat{w}_1 = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sum_{i=1}^{n}(x_i-\bar{x})^2}, \quad \widehat{w}_0 = \bar{y} - \widehat{w}_1\bar{x}$$

최대가능도법

- 가능도 함수를 최대화하는 w_0, w_1를 구함
  - 가능도 함수 : $L(w) = \prod_{i=1}^{n}\frac{1}{\sqrt{2\pi}\sigma}\exp\left[-\frac{(y_i-w_0-w_1x_i)^2}{2\sigma^2}\right]$
  - 최대가능도추정법 추정결과 : 최소제곱법과 동일

수치해석적 방법

- 선형 회귀모형은 g가 선형인 신경망으로 표현
  - $\hat{y} = g\left(\sum_{j=0}^{p} w_j x_j\right) = g(\boldsymbol{x}^T\boldsymbol{w})$



7. 최적화 방법

머신러닝과 최적화

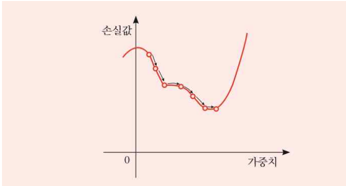
- 머신러닝 : 손실함수를 최소화하여 모형의 모수를 정함
  → 최적화(optimization) 문제
- 손실함수 미분해서 최적해를 구하기 어려운 경우
  - 수치해석 최적화 방법 : 뉴턴(Newton – Raphson) 방법과 경사하강법(Gradient Descent Algorithm)

7. 최적화 방법

딥러닝의 통계적이해
2023 출석수업

경사하강법

- 함수 1차 미분 가능, 볼록 함수가 아닌 복잡한 모양
  - 함수의 현재 위치에서 조금씩 이동 → 최솟값에 접근
  - 손실함수를 줄이는 경사를 따라 조금씩 가중치 갱신



7. 최적화 방법

딥러닝의 통계적이해
2023 출석수업

경사하강법

- 손실함수 $J(w)$ 최소점으로 가는 w를 찾는 법

$$w \coloneqq w - \eta \frac{\partial}{\partial w} J(w)$$

- 선형 회귀모형

$$w_0 \coloneqq w_0 - \eta \frac{1}{n} \sum_{i=1}^{n} (w_0 + w_1 x_i - y_i)$$

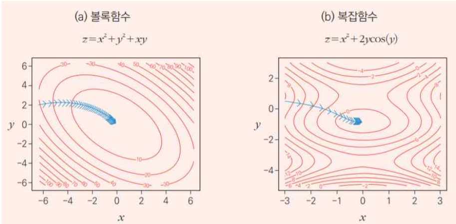
$$w_1 \coloneqq w_1 - \eta \frac{1}{n} \sum_{i=1}^{n} (w_0 + w_1 x_i - y_i) x_i$$

7. 최적화 방법

딥러닝의 통계적이해
2023 출석수업

경사하강법의 경로

- 손실함수가 복잡한 형태 → 국지 최솟값

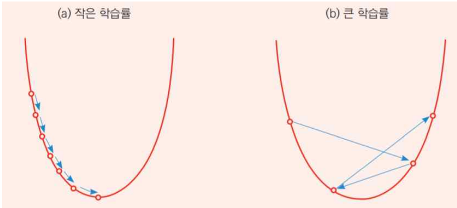


7. 최적화 방법

딥러닝의 통계적이해
2023 출석수업

경사하강법

- 경사하강법에서 학습률 η에 따라 수렴속도가 달라짐



7. 최적화 방법

딥러닝의 통계적이해
2023 출석수업

확률적 경사하강법(SGD)

- 데이터 임의로 한 개 선택 → $\frac{\partial}{\partial w}J(w)$ 구하고, 가중치 갱신
- 미니배치 경사하강법 : 일부 데이터를 이용하여 경사하강법 적용

7. 최적화 방법

딥러닝의 통계적이해
2023 출석수업

에포크(epoch)

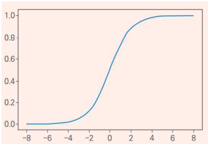
- 1 에포크(epoch) : 훈련 데이터 전체를 학습하는 것
  - 배치(batch)는 전체 훈련데이터를 몇 개로 나눈 것
  - 에포크(epoch)의 수는 전체 데이터를 학습한 횟수

딥러닝의 통계적이해
2023 출석수업

8. 로지스틱 회귀모형과 소프트맥스 회귀모형

로지스틱 회귀모형

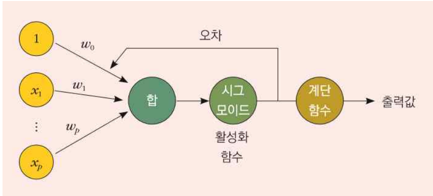
- 출력값은 2개 범주
  - 활성화 함수 : $g(x) = \frac{1}{1+e^{-x}}$



  - $\pi(\boldsymbol{x}) = P(y = 1|\boldsymbol{x})$
    - · $\pi(\boldsymbol{x}) > 0.5 \rightarrow y = 1$로 판단

로지스틱 회귀모형

- $\pi(\boldsymbol{x}) = g\left(\sum_{i=0}^{p} w_i x_i\right) = \frac{1}{1+\exp\left(-\sum_{i=0}^{p} w_i x_i\right)}$



로지스틱 회귀모형

- $\pi_i = \pi_i(\boldsymbol{x})$인 베르누이 분포
  - 가능도 함수 : $L(w) = \prod_{i=1}^{n} \pi_i^{y_i}(1-\pi_i)^{1-y_i}$
  - 로그가능도 함수 :

$$\log[L(w)] = \sum_{i=1}^{n}[y_i \log(\pi_i) + (1-y_i)\log(1-\pi_i)]$$

→ 경사하강법으로 w를 구함

로지스틱 회귀모형

- 로그가능도함수 최대화 = 손실함수 최소화

$$J(w) = -\log[L(w)] = \sum_{i=1}^{n} J_i(w)$$

$$J_i(w) = -[y_i \log(\pi_i) + (1-y_i)\log(1-\pi_i)]$$

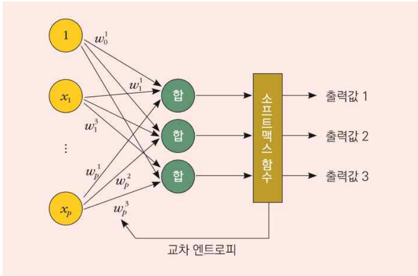
  - 경사하강법 : $w_j := w_j - \eta \frac{\partial}{\partial w_j} J(w)$

소프트맥스 회귀모형

◆ 여러 개중 하나를 분류할 때의 모형

- 소프트맥스 함수 : $g(x_i) = \frac{\exp(x_i)}{\sum_i \exp(x_i)}$
- $\pi_i^k = P(y_i = k | \boldsymbol{x}, \boldsymbol{w}^k) = \frac{\exp(x w^k)}{\sum_{j=1}^{K} \exp(x w^j)}$

소프트맥스 회귀모형



소프트맥스 회귀모형

◆ 멀티누이분포 → 손실함수 : 교차 엔트로피

$$J(w) = -\sum_i \sum_k y_i^k \log(\pi_i^k)$$

- 경사하강법 : 가중치 갱신

$$w_j^k := w_j^k - \eta \frac{\partial J(w)}{\partial w_j^k}$$

활성화 함수와 손실함수

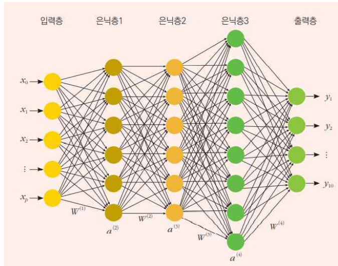
	출력변수	활성화 함수	손실함수
이산형	베르누이 분포	시그모이드	이진 엔트로피
이산형	멀티누이 분포	소프트맥스	교차 엔트로피
연속형	정규분포	선형	평균제곱오차

딥러닝의 통계적이해
2023 출석수업

9. 다층신경망

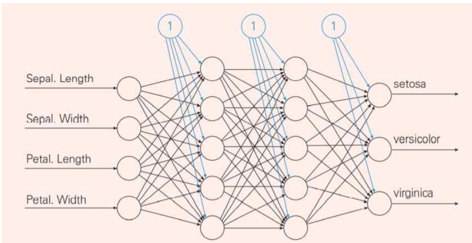
다층신경망

- ◆ 입력층, 출력층 사이 은닉층이 포함된 신경망
  → 다층 퍼셉트론(MLP) → 딥러닝(deep learning)



다층신경망의 예

- ◆ 붓꽃 종의 분류를 다층신경망을 이용
  - 동그란 부분이 뉴런(또는 노드), 연결된 선이 네트워크



순방향신경망

- ◆ 순방향신경망(Feedforward Neural Network, FNN) : 같은 층 내에서는 연결되지 않고 앞의 층으로만 연결

  - 뉴런이 위층의 뉴런과 모두 연결됨 → 완전연결망 (Fully Connected Network, FCN)이라고 부름

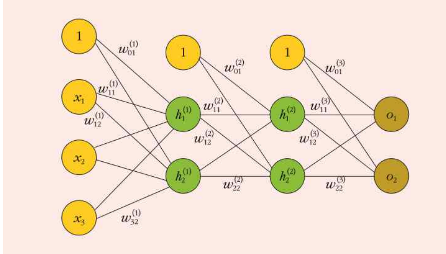
다층신경망과 합성함수

- 다층신경망 : 합성함수로 표현
  - 함수의 합성이 반복 → 신경망의 목적 함수의 표현력이 좋아짐

$$y = f^{(3)}(f^{(2)}(f^{(1)}(x)))$$

다층신경망의 표현

- 뉴런이 3개인 입력층, 뉴런이 2개인 은닉층이 2층, 뉴런이 2개인 출력층으로 구성된 다층 신경망



다층신경망의 표현

- 뉴런이 3개인 입력층, 뉴런이 2개인 은닉층이 2층, 뉴런이 2개인 출력층으로 구성된 다층 신경망

가중합 : $z_j^{(1)} = w_{0j}^{(1)} + w_{1j}^{(1)}x_1 + w_{2j}^{(1)}x_2 + w_{3j}^{(1)}x_3 = \sum_{i=0}^{3} w_{ij}^{(1)}x_i$

활성화 함수 적용 : $h_j^{(1)} = a(z_j^{(1)}) = a(\sum_{i=0}^{3} w_{ij}^{(1)}x_i)$

다층신경망의 표현

- 뉴런이 3개인 입력층, 뉴런이 2개인 은닉층이 2층, 뉴런이 2개인 출력층으로 구성된 다층 신경망

제1 은닉층 : $h_{j_1}^{(1)} = a(\sum_{i=0} w_{ij_1}^{(1)}x_i),\ j_1 = 1,\ 2$

제2 은닉층 : $h_{j_2}^{(2)} = a(\sum_{j_1=0}^{2} w_{j_1j_2}^{(2)}h_{j_1}^{(1)}),\ j_2 = 1,\ 2$

출력층 : $o_{j_3} = a_o(\sum_{j_2=0}^{2} w_{j_2j_3}^{(3)}h_{j_2}^{(2)}),\ j_3 = 1,\ 2$

출력층의 활성화 함수

- 회귀모형 : 항등함수
- 이진분류 : 시그모이드 함수
- 다중분류 : 소프트맥스 함수

신경망의 구조 설계

- 은닉층의 수와 각 층별 뉴런의 수를 정하는 것
  - 층이 깊어질수록 신경망이 보다 추상적으로 입력 데이터를 파악→ 데이터의 표현력이 좋아짐
  - 층별 뉴런의 수를 늘려도 신경망의 성과가 크게 높아지지는 않음

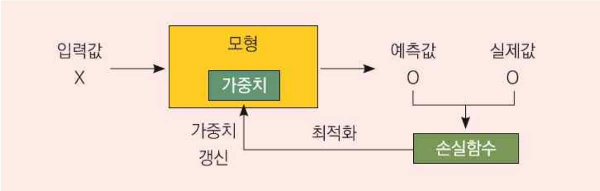
일반근사정리

- 충분한 크기의 뉴런을 가진 은닉층 하나 이상의 다층 신경망이 모든 유계 연속함수를 근사
  - 충분히 많은 뉴런을 가진 은닉층 1개인 신경망으로도 모든 함수를 표현
  - 신경망이 딥러닝 모형으로 발전하는데 방해 요인

10. 신경망의 학습

신경망의 작성과정

◆ 신경망의 작성과정



10. 신경망의 학습
딥러닝의 통계적이해
2023 출석수업

신경망의 작성과정

◆ 순방향 : 입력층 뉴런 → 은닉층 뉴런 → … 출력 뉴런

- 역방향 : 손실함수 $J(w)$ 기반 경사하강법을 통해 반복적으로 가중치들 갱신

$$w := w - \eta \frac{\partial}{\partial w} J(w)$$

- 가중치들이 갱신되더라도 손실함수가 더 이상 줄지 않는다면 그 가중치들이 최적값 판단
  → 그 가중치들의 신경망을 활용하여 예측

10. 신경망의 학습
딥러닝의 통계적이해
2023 출석수업

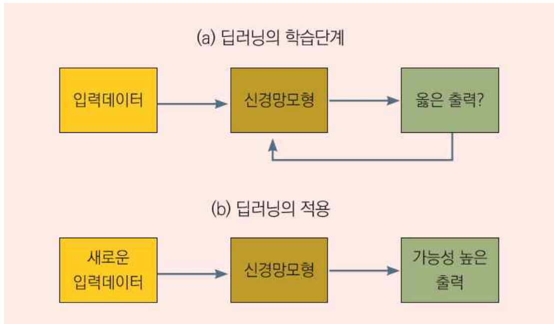
신경망의 학습 데이터

◆ 전체 데이터를 훈련(training), 검증(validation), 시험(test) 데이터를 나눔

- 훈련 데이터 : 신경망의 학습
- 검증 데이터 : 신경망의 선택
- 시험 데이터 : 신경망의 성과 확인
- 60%, 20%, 20%로 분할

10. 신경망의 학습
딥러닝의 통계적이해
2023 출석수업

신경망의 학습과 적용



10. 신경망의 학습

딥러닝의 통계적이해

2023 출석수업

손실함수

◆ 손실함수 $J(w)$: 출력층 값과 실제 값을 비교해서 구함

$$J(w)=\frac{1}{2n}\sum_{i=1}^{n}(y_i-o_i)^2$$

$$J(w)=-\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}y_{ik}\log(o_{ik})$$

10. 신경망의 학습

딥러닝의 통계적이해

2023 출석수업

학습(learning)

◆ 손실함수를 경사하강법을 통해 경사에 따라 일정한 학습률로 가중치를 갱신시켜나가는 것
- 가중치의 초깃값을 정하는 것이 학습속도를 높이는데 중요
  - · 초깃값은 주로 0 근처의 작은 값에서 시작
  - · 초깃값을 너무 큰 값
    - → 활성화함수가 제대로 작동하지 않음

10. 신경망의 학습

딥러닝의 통계적이해

2023 출석수업

학습(learning)

◆ 1 에포크(epoch) : 한 데이터셋 전체를 1회 학습
- 전체 훈련 데이터를 미니배치로 나누고, 이 조각을 모두 합쳐서 1번 훈련시킨 것도 1 에포크
- 에포크 수 증가
  - → 훈련 데이터의 정확도 증가, 손실함수 감소

10. 신경망의 학습

딥러닝의 통계적이해

2023 출석수업

오차역전파법

◆ 경사하강법에서 연쇄 미분을 통해 손실함수의 경사를 효율적으로 구하여 신경망의 가중치를 갱신하는 것
- Rumelhart, Hinton, and Williams (1986)
  - → 연산은 순방향과 역방향으로 구분

10. 신경망의 학습

딥러닝의 통계적이해
2023 출석수업

오차역전파법

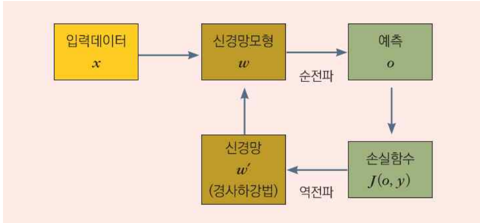
- 순방향 연산 : 가중치의 초깃값을 정한 후 [입력데이터 → 신경망모형 → 예측] 순, 누적적·순차적으로 가중합과 비선형 연산 → 예측 오차로 손실함수 값을 구함
- 역방향 연산 : 손실함수 값 줄이도록 가중치 갱신
  - 순방향 연산의 역 방향으로 경사(기울기, 미분값)을 구함

10. 신경망의 학습

딥러닝의 통계적이해
2023 출석수업

오차역전파법의 연산

- 순방향 연산시 미분을 저장해 두었다가 나중에 역전파 연산시 이를 이용 → 합성함수 미분 이용, 계산 그래프 이용



딥러닝의 통계적이해
2023 출석수업

11. 딥러닝 학습의 제 문제

11. 딥러닝 학습의 제 문제

딥러닝의 통계적이해
2023 출석수업

딥러닝 모형 학습시 문제

- 경사소실 : 활성화 함수의 문제로 경사하강법이 제대로 작동되지 않아 가중치가 갱신되지 않는 현상
- 초깃값 설정 : 초깃값을 잘못 설정하는 경우 손실함수가 국지적 최솟값에 머뭄
- 과대적합 : 훈련데이터에서는 딥러닝 모형의 적합도가 높지만 새로운 데이터에 대한 예측이 제대로 되지 않음
- 경사하강법에서 학습시간 과다 및 국지적 최적화

11. 딥러닝 학습의 제 문제

딥러닝의 통계적이해

2023 출석수업

경사소실의 원인

- 활성화 함수의 미분과 경사하강법
  - 시그모이드 함수 미분 : 최댓값 0.25, 0에서 멀어질수록 작은 값
  - tanh 함수 미분 : 최댓값 1, 0에서 멀어질수록 0에 가까움
  - ReLU 함수 : 0보다 큰 값에서 미분값이 1 경사소실 미발생하여 주로 이용됨

11. 딥러닝 학습의 제 문제

딥러닝의 통계적이해

2023 출석수업

붓꽃 데이터의 분류

- 은닉층 2개인 신경망
  - 시그모이드 함수와 ReLU 함수, 500회 학습

	훈련 데이터		시험 데이터	
	손실	정확도	손실	정확도
시그모이드 함수	0.8528	0.7083	0.8874	0.6000
ReLU 함수	0.2703	0.8917	0.2630	0.9000

11. 딥러닝 학습의 제 문제

딥러닝의 통계적이해

2023 출석수업

심층신뢰망

- 2006년 힌튼 교수 연구팀이 제시한 모형
- 제한된 볼츠만 머신(RBM)을 적층 신경망을 깊게 쌓는 방법
  - RBM은 출력층 없이 입력층(가시층)과 은닉층만 있는 신경망
- 다층 신경망을 딥러닝이라 부름

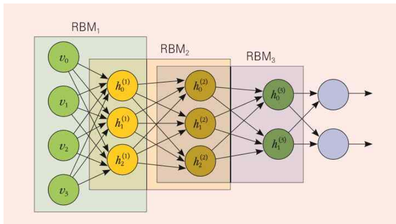
11. 딥러닝 학습의 제 문제

딥러닝의 통계적이해

2023 출석수업

심층신뢰망

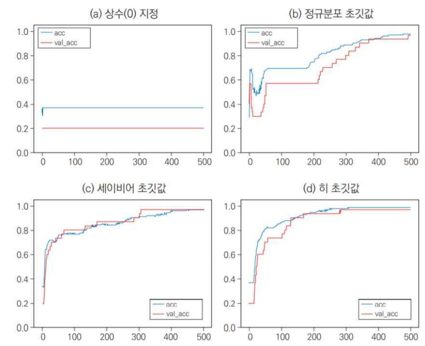
- 심층신뢰망은 다층 신경망과 동일한 구조처럼 보이나 실제로는 RBM들을 쌓아가면서 연결한 모형



초깃값의 선택

- 초깃값 일반적 방법 : 평균 0, 작은 분산 정규분포로부터 난수
  - 가장 쉬운 방법 ; 모두 같은 값 0으로 지정
    → 오차역전파법 적용시 가중치가 같은 값으로 갱신
- 세이비어(Xavier) 초깃값과 히(He) 초깃값
  - 시그모이드 함수 : 세이비어 초깃값 $N\left(0, \frac{2}{n_{input}+n_{output}}\right)$
  - ReLU 함수 : 히 초깃값 $N\left(0, \frac{4}{n_{input}+n_{output}}\right)$

초깃값의 비교



확률적 경사하강법

- 손실함수의 최솟값으로 갈 때 데이터를 임의로 뽑아서 진행 → 손실함수의 최솟값으로 진동하면서 내려감
  - 확률적 경사하강법 적용시 이전의 미분값(경사)을 기억하지 않고 진행

$$w_{ij}^{(l)} := w_{ij}^{(l)} - \eta \frac{\partial J(w)}{\partial w_{ij}^{(l)}}$$

최적화 방법 비교학습률

- 최적화 방법과 별도로 학습률을 학습에 따라 바꾸어 진행
  - 학습률 $\eta = \eta_0 10^{-\frac{t}{n}}$로 지정
    → 학습이 진행됨에 따라 학습률을 조금씩 줄여주는 것

모멘텀(Momentum) 방법

- 확률적 경사하강법의 손실함수 감소 경로에서 관성을 이용하여 평활하게 움직이도록 하는 방법
  - 이전의 미분값(경사)에 β를 곱해줘서 누적 갱신
  - β값이 0.9라면 $\frac{\eta}{1-0.9} = 10\eta$의 속도로 최적점에 접근

$$m_{t+1} = \beta m_t + \eta \frac{\partial J(w)}{\partial w_{ij}^{(l)}}$$

$$w_{ij}^{(l)} := w_{ij}^{(l)} - m_{t+1}$$

AdaGrad 방법

- 최저점에 가까워질수록 학습률이 감소하는 방법
  - 가중치 갱신이 천천히 이루어져서 최저점에 도달하기도 전에 학습이 끝나는 문제

$$s_0 = 0$$

$$s_{t+1} = s_t + \left(\frac{\partial J(w)}{\partial w_{ij}^{(l)}}\right)^2$$

$$w_{ij}^{(l)} := w_{ij}^{(l)} - \eta \frac{\partial J(w)}{\partial w_{ij}^{(l)}} / \sqrt{s_{t+1} + \varepsilon}$$

  - ε : 매우 작은 값 (ex. $\varepsilon = 10^{-10}$)

RMSprop 방법

- AdaGrad 방법 지수평활법을 적용하여 성능 개선, $\alpha = 0.9$

$$s_0 = 0$$

$$s_{t+1} = \alpha s_t + (1-\alpha)\left(\frac{\partial J(w)}{\partial w_{ij}^{(l)}}\right)^2$$

$$w_{ij}^{(l)} := w_{ij}^{(l)} - \eta \frac{\partial J(w)}{\partial w_{ij}^{(l)}} / \sqrt{s_{t+1} + \varepsilon}$$

Adam 방법

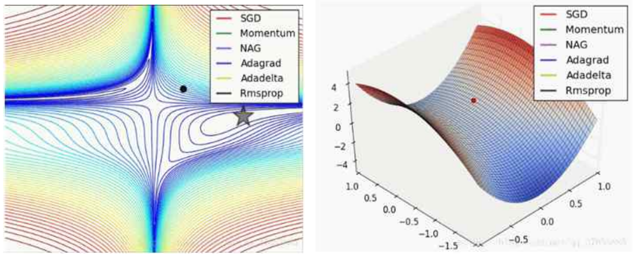
- 모멘텀 방법과 미분값(경사)의 지수평활 값과 RMSProp방법에서의 미분값(경사) 제곱의 지수평활 값을 이용 →가중치 갱신
  - $\beta_1 = 0.9,\ \beta_2 = 0.999,\ \ \varepsilon = 10^{-8}$
  - Adam방법이 딥러닝 모형의 학습에서 자주 이용됨

Adam 방법

$$\mathrm{m}_0 = 0,\ s_0 = 0$$
$$\mathrm{m}_{t+1} = \beta_1 m_t + (1-\beta_1)\frac{\partial J(w)}{\partial w_{ij}^{(l)}}$$
$$s_{t+1} = \beta_2 s_t + (1-\beta_2)\left(\frac{\partial J(w)}{\partial w_{ij}^{(l)}}\right)^2$$
$$w_{ij}^{(l)} := w_{ij}^{(l)} - \eta m_{t+1} / \sqrt{s_{t+1} + \varepsilon}$$

https://ruder.io/content/images/2016/09/saddle_point_evaluation_optimizers.gif

최적화 방법의 비교

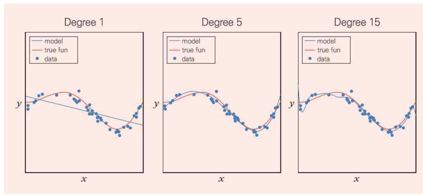


https://ruder.io/optimizing-gradient-descent/

12. 편의와 분산

과소적합과 과대적합

- 산점도의 데이터를 p를 1, 5, 15로 달리하여 다항 회귀모형으로 추정



12. 편의와 분산

딥러닝의 통계적이해

2023 출석수업

과소적합과 과대적합

- 훈련데이터와 검증데이터 분류 오류율을 비교하여 모형의 과대적합 여부 파악

	훈련 데이터	검증 데이터	판단
분류 오류율	낮음	보통	과대적합(고분산)
	보통	보통	과소적합(고편의)
	보통	높음	고편의, 고분산
	낮음	낮음	저편의, 저분산

12. 편의와 분산

딥러닝의 통계적이해

2023 출석수업

모형의 선택

- 머신러닝, 딥러닝 모형 성과가 비슷 → 간단한 모형 선택
  - 간단한 모형이 계산량도 적고, 분산도 작기 때문
- 딥러닝 모형에서 성과가 나쁜 경우 과대적합(고분산) 상황
  - 훈련데이터를 추가하거나 모형을 단순화

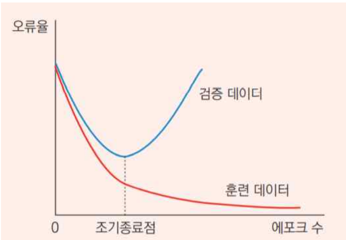
12. 편의와 분산

딥러닝의 통계적이해

2023 출석수업

조기 학습종료

- 조기학습 종료 : 검증데이터의 손실함수가 감소하다가 증가하는 부분에서 학습을 중단하고 그 가중치를 이용



12. 편의와 분산

딥러닝의 통계적이해

2023 출석수업

정칙화

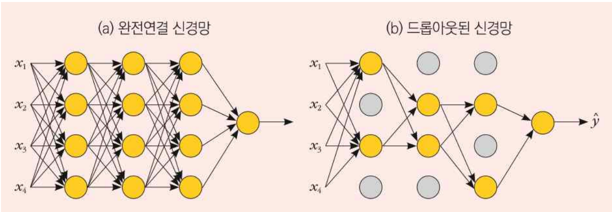
- 정칙화(regularization) : 적합부분 이루어진 기존 손실함수에 가중치 벌칙항을 더해 손실함수 정의, 가중치 구함
  - L_2 정칙화 : 적절한 수준에서 적합되는 가중치값 선택
  - L_1 정칙화 : 많은 가중치들이 0이 됨

12. 편의와 분산

딥러닝의 통계적이해
2023 출석수업

드롭아웃

◆ 드롭아웃(Dropout) : 입력층부터 은닉층 사이의 뉴런 간 네트워크를 임의로 없애고 학습하는 것



12. 편의와 분산

딥러닝의 통계적이해
2023 출석수업

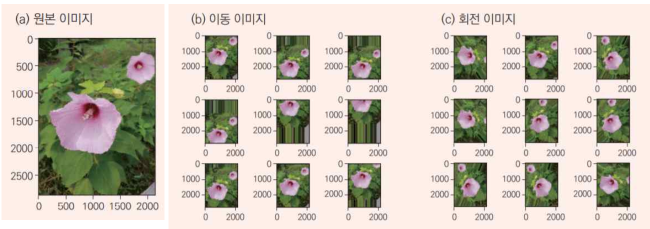
데이터 증식

◆ 데이터 증식(data augmentation) : 데이터의 수 증가
- 딥러닝 모형은 추정해야 할 네트워크 가중치가 많아서 매우 많은 데이터가 필요, 실제로는 부족
- 이미지 한 장을 바탕으로 수 백, 수 천개를 생성 :
  - 수평이동, 대칭이동, 정규분포, 이항분포를 따르는 노이즈를 추가, 이미지의 일부 확대 등

12. 편의와 분산

딥러닝의 통계적이해
2023 출석수업

데이터 증식



딥러닝의 통계적이해
2023 출석수업

13. 배치 정규화

경사하강법과 활성화 함수

- 경사하강법에서 활성화 함수에 의존하지 않는 방법이 필요
  - 입력층, 은닉층의 데이터 값을 표준화 → 활성화 함수 적용 전 값들이 0 근처의 값으로 분포
    → 딥러닝 학습에서 활성화 함수의 영향력을 줄임

배치 정규화

- 아래 과정을 미니 배치 단위로 반복
  ① 표준화 : 분산은 1, 평균은 0을 만든 다음 $z_{norm}^{(l)}$ 을 구함
  • γ와 β를 더해서 새로운 입력값 $\tilde{z}^{(l)}$을 만듬

$$\hat{\mu}^{(l)} = \frac{1}{m}\sum_i z_i^{(l)}, \quad \hat{\sigma}^{2(l)} = \frac{1}{m}\sum_i \left(z_i^{(l)} - \hat{\mu}^{(l)}\right)^2$$

$$z_{norm}^{(l)} = \frac{z^{(l)} - \mu^{(l)}}{\sqrt{\hat{\sigma}^{2(l)} + \varepsilon}}, \quad \tilde{z}^{(l)} = \gamma z_{norm}^{(l)} + \beta$$

배치 정규화

② $\tilde{z}^{(l)}$에 활성화 함수를 적용
  • γ와 β 등을 미니배치에 대해 경사하강법을 적용

$$x \xrightarrow{w^{(1)}} z^{(1)} \xrightarrow{\beta,\gamma} \tilde{z}^{(1)} \to h^{(1)} = a\left(\tilde{z}^{(1)}\right) \xrightarrow{w^{(2)}} z^{(2)} \ldots$$

배치 정규화

- 배치 정규화는 양쪽 극단 값이 덜 발생 → 학습이 잘 이루어지도록 하고 경사소실 문제 해소
  - 은닉층 입력값들이 제대로 분포 → 학습에서 초깃값의 의존성이 줄어들고 과대적합을 억제 (드롭아웃과 정칙화 불필요)
  - γ와 β같이 추가적 추정 모수 증가 → 모형이 더 복잡, 추가적 학습시간 소요

딥러닝의 통계적이해
2023 출석수업

14. 하이퍼파라미터의 선택과 전이학습

하이퍼파라미터

- 신경망 학습 전에 미리 값을 정하는 모수
  - 경사하강법의 학습률 η와 최적화방법, 미니배치의 크기, 딥러닝 모형의 설계 등

하이퍼파라미터의 탐색

- 임의로 탐색한 후 좋은 결과를 보이는 범위를 정하고 그 범위 내에서 더 세밀하게 하이퍼파라미터를 탐색
  - 탐색이 환경상 불가능하다면 좋은 성과를 보였다고 알려진 논문의 하이퍼파라미터를 이용

구글 텐서플로우 플레이그라운드

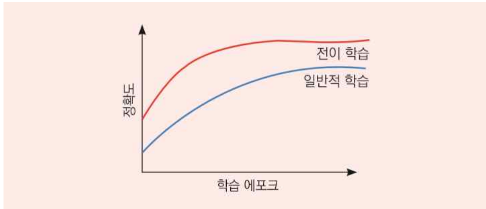
- 하이퍼파라미터의 효과를 파악하기 위한 도구

전이학습

- 전이학습(transfer learning) : 이미 훈련된 신경망으로 신경망을 학습
  - 데이터와 컴퓨팅 환경이 충분하지 않다면 이미지넷 경진대회에서 우승했던 모형들의 공개된 가중치 그대로 이용

전이학습

- 전이학습을 이용하여 신경망 모형을 학습하면 좋은 초깃값



전이학습

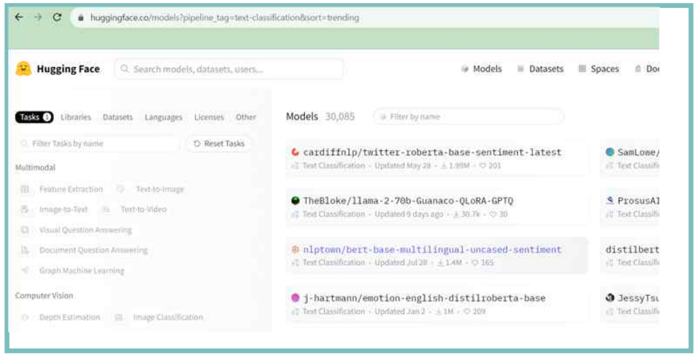
- 데이터 크기와 유사성에 따른 전이학습 방안

		데이터의 유사성	
		크다	작다
데이터	소규모	기존 모형	학습 어려움
	대규모	일부 층 조정	많은 층 조정

거대 딥러닝 모형의 학습

구분	정의	예시
Zero-shot learning	학습데이터 없이 패턴 인식하는 방법	구글번역
Few-shot learning	소량의 학습 데이터만으로 학습	고양이와 호랑이 사진을 몇 개씩 보여주고, 동물들 중에서 고양이와 호랑이 구분.
One-shot learning	하나의 학습 데이터로 학습	고양이 사진 하나를 보여주고, 동물들 중에서 고양이 분류.

Hugging Face



https://huggingface.co

딥러닝의 통계적이해
2023 출석수업

15. 구글 Colaboratory

딥러닝의 통계적이해
2023 출석수업

컴퓨팅 환경

- 딥러닝 모형 구현 컴퓨팅 환경 → 개별 PC에서는 복잡
  - GPU, Python, Tensorflow 등 딥러닝 프레임워크
  → 딥러닝 개발 환경 : 구글 Colaboratory(Colab)

15. 구글 Colaboratory

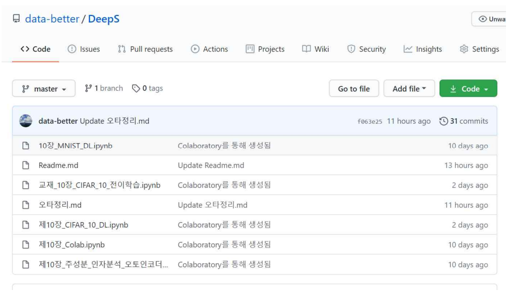
딥러닝의 통계적이해
2023 출석수업

Colab의 개요

- 구글 Colab은 구글 클라우드의 가상 리눅스(Linux) 서버 기반 주피터 노트북(Jupyter notebook)
  - Python, Tensorflow, Keras, Numpy 등이 설치
  - GPU와 TPU 이용 가능
  - Python과 파일은 구글 드라이브에 저장, 이용되지 않는 중 가상머신이 중지되어 초기화

Colab의 실습코드

https://github.com/data-better/DeepS



완전연결 신경망

- MNIST 데이터를 이용하여 손글씨를 식별하는 은닉층 1개의 완전연결 신경망

```
import tensorflow as tf
from tensorflow.keras import datasets, models
from tensorflow.keras.layers import Flatten, Dense, Dropout, Conv2D, MaxPool2D
from tensorflow.keras.utils import plot_model
print(tf.__version__)
```


완전연결 신경망

```
mnist = datasets.mnist
(train_x, train_y),(test_x, test_y)=mnist.load_data()
train_x, test_x = train_x / 255.0, test_x / 255.0
import matplotlib.pyplot as plt
for col1 in range(16):
   plt.subplot(4,4,col1+1)
   plt.imshow(train_x[col1].reshape(28,28), cmap=plt.cm.binary)
plt.show()
```


완전연결 신경망

```
digit =  train_x[0]
print("digit :", digit.shape)
print("train images :", train_x.shape)
print("test  images :", test_x.shape)
```

완전연결 신경망

```
model1 = models.Sequential([
    Flatten(input_shape=(28, 28)),
    Dense(512, activation='relu'),
    Dense(10, activation='softmax')
])
model1.summary()
plot_model(model1, to_file="model1_mnist.png",
show_shapes=True)
```


완전연결 신경망

```
model1.compile(optimizer='adam',
 loss='sparse_categorical_crossentropy',
 metrics=['accuracy'])

hist = model1.fit(train_x, train_y, epochs=12,
 batch_size=256, validation_split=0.25)
```


완전연결 신경망

```
plt.plot(hist.history['accuracy'], 'b-')
plt.plot(hist.history['val_accuracy'], 'r--')
plt.legend(['train', 'test'], loc='upper left')
plt.ylim([0.94,1.005])
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.show()
sc = model1.evaluate(test_x, test_y)
print("accuracy : ", sc[1], "  loss : ", sc[0])
```


16. 딥러닝의 문제점

16. 딥러닝의 문제점

딥러닝의 통계적이해

2023 출석수업

문제점

- 고용 및 소득 불평등 문제 : 일자리 대체 등
- 사회적 억압과 사생활 침해 : 얼굴 인식, 사생활 추적
- 사회적 조작 : 가짜뉴스 문제
- 환경 문제 : 과도한 전력 소비
- 저작권 문제: 인간 저자 학습한 후 생성
- 자율 무기

16. 딥러닝의 문제점

딥러닝의 통계적이해

2023 출석수업

가짜뉴스

- 2023년5월22일(미국) 트위터 가짜뉴스와 주가하락

출처 : https://twitter.com/WGMImedia/status/1661294126099480583