

1주차

📅 날짜	@2024년 3월 12일
☰ 과제	강의 요약 출석 퀴즈
☰ 세부내용	[딥러닝 1단계] 1. 딥러닝 소개 2-1. 신경망과 로지스틱회귀 (이진 분류~더 많은 미분예제)
📎 자료	1주차 출석 퀴즈 C1_W1_note.pdf

딥러닝 소개

1 환영합니다!

인공지능은 새로운 전기

- 전기가 산업적으로 많은 변화를 가져온 것처럼 인공지능도 큰 변화를 보여주고 있음
- 딥러닝 : 인공지능의 한 부분

강의 목차

1. NN과 DL의 기초 → 고양이 인식기 생성
2. DNN 성능 개선 : 하이퍼파라미터 튜닝, 규제, 최적화
3. 머신러닝 프로젝트
4. CNN 구축 방법
5. 시퀀스 모델(RNN, LSTM)과 자연어 처리

2 신경망은 무엇인가?

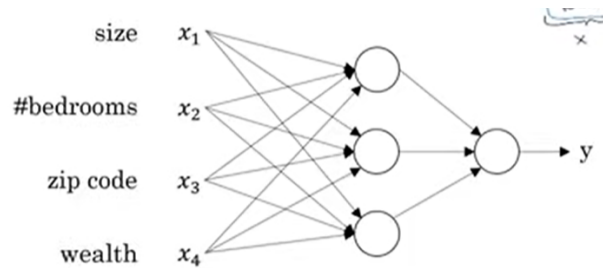
주택 가격 예측 예제 (1)

: 주택의 크기와 가격에 대한 6개의 데이터를 가지고 가격을 예측할 수 있는 함수 생성하기

- 선형회귀를 적용하면?
 - 점들을 잘 설명할 수 있는 하나의 직선을 그리고,
가격은 음수일 수 없으므로 직선이 0에서 끝나도록 꺾어서 그림
- 간단한 신경망으로 생각해보면?
 - $\text{Size}(X) \rightarrow O(\text{Neuron}) \rightarrow \text{Price}(Y)$: 하나의 작은 신경망
 - Neuron의 역할 : Size를 입력으로 받아서 선형 함수를 계산하고 $\max(\text{결과값}, 0)$ 을 Price로 예측
 - ReLU(Rectified Linear Unit) : 0으로 유지되다가 직선으로 증가하는 형식의 함수
- 하나의 작은 신경망을 쌓아 더 큰 신경망을 만들 수 있음

주택 가격 예측 예제 (2)

- 주택의 크기, 침대의 수 $\rightarrow O \rightarrow$ 가족의 크기 \searrow
- 우편번호 $\rightarrow O \rightarrow$ walkability \rightarrow 주택 가격(Price)
- 우편번호, 재력 $\rightarrow O \rightarrow$ 학교의 질 \nearrow



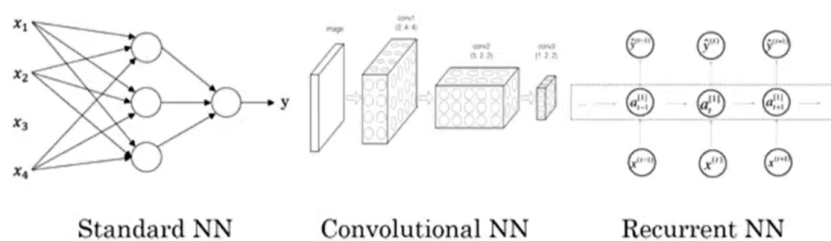
- X : 입력값 / O : 신경망의 은닉 유닛 / y : 예측하고자 하는 값

3 신경망을 이용한 지도학습

지도학습(Supervised Learning)

Input(x)	Output(y)	Application	Neural Network
Home features	Price	Real Estate	Standard NN
Ad, user info	Click on ad? (0/1)	Online Advertising	Standard NN
Image	Object (1,...,1000)	Photo tagging	CNN
Audio	Text transcript	Speech recognition	RNN
English	Chinese	Machine translation	RNN
Image, Radar info	Position of other cars	Autonomous driving	Custom/Hybrid NN

Neural Network Examples



정형 데이터와 비정형 데이터

- 정형 데이터(Structured Data) : 행과 열을 가진 데이터베이스
- 비정형 데이터(Unstructured Data) : 오디오, 이미지, 텍스트, ...

4 왜 딥러닝이 뜨고 있을까요?

Scale drives deep learning progress

- 전통적인 알고리즘 : 어느 지점에 도달하면 성능이 정체됨
- 현재 : 전통적인 알고리즘이 다룰 수 없을 정도로 데이터 양이 많아짐
- 신경망의 규모가 커질수록 성능이 향상됨
 - ⇒ 높은 성능 발휘를 위해 1) 많은 양의 데이터를 다룰 수 있는 큰 신경망과 2) 많은 데이터가 필요
- 규모 : 신경망의 크기, 많은 은닉층, 많은 연결, 많은 파라미터, 많은 데이터
- small training set : 구현 방법에 따라 성능이 결정됨
- large training set : 큰 신경망의 성능이 압도적

딥러닝 성능 향상의 요인

- Data 규모 증가
- Computation 능력 향상 : CPU, GPU에서 훈련이 가능하게 됨
- Algorithms의 개선 : Sigmoid 함수에서 ReLU 함수로의 변화로 학습 속도 개선
 - Sigmoid 함수 : 기울기가 0에 가까운 곳에서 학습이 매우 느리게 일어남
 - ReLU 함수 : 입력값이 양수인 경우 기울기가 1이므로 Sigmoid 함수의 기울기 소실 문제 보완
- 신경망 학습 과정이 반복적이기 때문에 효율적인(빠른) 신경망을 만드는 것이 필요함

신경망과 로지스틱회귀

1 이진 분류(Binary Classification)

이진 분류의 예

- 고양이 사진(64x64) → 3개(RGB)의 64x64 행렬 → 입력 특성 벡터 X 생성(64x64x3) → y를 0(고양이가 아님) 또는 1(고양이)로 예측하는 분류기 생성

Notation

- (x, y) $x \in \mathbb{R}^{n_x}, y \in \{0, 1\}$
- m개의 training sample : $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)})\}$
- m_{train} : # of training sample
- m_{test} : # of test sample
- $X = \begin{bmatrix} x^{(1)} & x^{(2)} & \dots & x^{(m)} \end{bmatrix}, X \in \mathbb{R}^{n_x \times m}$
- $Y = \begin{bmatrix} y^{(1)} & y^{(2)} & \dots & y^{(m)} \end{bmatrix}, Y \in \mathbb{R}^{1 \times m}$

2 로지스틱 회귀(Logistic Regression)

로지스틱 회귀

- 목표 : X 가 주어졌을 때, $\hat{y} = P(y = 1|x)$ 를 구하는 것
- Parameters : $W \in \mathbb{R}^{n_x}, b \in \mathbb{R}$
- Output : $0 \leq \hat{y} = \sigma(W^T X + b) = \sigma(z) \leq 1$
 - 예측값은 y 가 1일 확률이므로 0과 1 사이의 수여야 하므로 시그모이드 함수를 적용
- 시그모이드 함수 : $\text{sigmoid}(z) = \frac{1}{1+e^{-z}}$
 - 만약 z 가 크면 $\sigma(z) \approx 1$, z 가 작으면 $\sigma(z) \approx 0$

신경망 구현 시 파라미터 표기법

- $\hat{y} = \sigma(\theta^T X)$
- $\theta = [\theta_0 \ \theta_1 \ \dots \ \theta_{n_x}]^T$
 - $\theta_0 = b$
 - $\theta_1, \dots, \theta_{n_x} = W$

3 로지스틱 회귀의 비용함수

손실함수(Loss function)

- X 에 대한 실제값(y)과 예측값(\hat{y})의 오차를 계산하는 함수
- 일반적인 손실함수 : $L(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$
 - 여러 개의 지역 최적값을 가지는 문제로 로지스틱 회귀에서는 사용하지 않음

로지스틱 회귀의 손실함수

- $L(\hat{y}, y) = -(y \log \hat{y} + (1 - y) \log (1 - \hat{y}))$
- $y=1$ $L(\hat{y}, y) = -\log \hat{y}$ 가 0에 가까울수록 좋음 → \hat{y} 가 1에 수렴해야 함
- $y=0$ $L(\hat{y}, y) = -\log(1 - \hat{y})$ 가 0에 가까울수록 좋음 → \hat{y} 는 0에 수렴해야 함

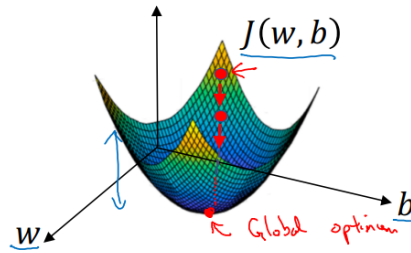
비용함수(Cost function)

- 훈련 세트 전체에 대해 얼마나 잘 예측되었는지 측정해주는 함수
- $J(W, b) = \frac{1}{m} \sum_{i=1}^m L(\hat{y}^{(i)}, y^{(i)}) = -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})]$

⇒ 로지스틱 회귀 모델을 학습하는 것은 비용함수 J 를 최소화하는 매개변수 W 와 b 를 찾는 것 !

4 경사하강법(Gradient Descent)

경사하강법이란?



- 비용 함수 볼록한 함수
- w, b 의 초기값을 설정 → 가장 가파른 방향을 따라 파라미터 업데이트 → Global optimum에 도착

경사하강법 알고리즘

- $w := w - \alpha \frac{\partial J(w, b)}{\partial w}$ 과 $b := b - \alpha \frac{\partial J(w, b)}{\partial b}$ 의 갱신 작업을 반복
- α : 학습률(learning rate)
- $dw = \frac{dJ(w, b)}{dw}$: 미분계수(=임의의 점에서의 기울기)

5 미분(Derivatives)

일차함수에서의 미분

- 도함수(=기울기) : 삼각형의 높이를 밑변의 너비로 나눈 값
→ 변수 a 가 아주 조금만 변화했을 때, 함수 $f(a)$ 값이 얼마나 변하는지 측정
- 도함수 표기 : $\frac{d}{da} f(a) = \frac{df(a)}{da}$

6 더 많은 미분 예제

함수 : $f(a)$	a^2	a^3	$\ln(a)$
도함수 : $\frac{d}{da} f(a)$	$2a$	$3a^2$	$\frac{1}{a}$

⇒ 함수의 기울기는 함수의 위치에 따라 달라질 수 있음