<신경망이란?>

뉴런(마치 레고 블록)들로 구성.

신경망 은닉 유닛이 입력을 받고 (노드), input layer이 입력과 출력을 연결.

주택의 가격에 영향을 미치는 것을 예시로 하면 가족의 수를 결정하기 위해서는 우편번호, 침실의 수 등이 영향을 미침

뉴런이나 예측기를 쌓으면서 신경망의 크기를 키움. 훈련세트에서 u 출력 y

<지도학습> - 머신러닝의 한 기법

x와 y에 매핑되는 함수를 학습. (입력은 주택관련 특성, 출력은 주택의 가격)

(딥러닝 응용분야)

- 온라인 광고광고를 클릭할 것인지 아닌지 예측하는 것. 클릭릭할 만한 광고를 보여주는 것.
- 사진 태깅, 음성인식(음성파일을 넣으면 텍스트파일로 출력), machine translation 자율주행 차들의 위치 정보 학습.

응용분야에 각각 적절한 다른 신경망들을 사용해야한다.

-부동산 어플리케이션: 표준 신경망 구조 사용

-온라인 광고: 표준신경망

-이미지분야: CNN 합성곱 신경망

-음성데이터(1차원 시계열 시퀀스 데이터): RNN

-언어 : 시퀀스데이터. RNN

-자율주행 : CNN, 하이브리드 신경망 구조

<구조적/비구조적 데이터>

- 구조적: 데이터 베이스로 표현되는 데이터

(나이, 광고정보 침실의 개수.. 등등)

사용자 맞춤 . 정확한 예측.

- 비구조적: 음성파일, 이미지, 텍스트.

  구조적 데이터보다 작업이 어려움. 딥러닝으로 비구조적 데이터 분석 수월해짐. 음성인식, 자연언어처리 같은 응용 분야 생김

<왜 딥러닝이 뜨고 있을까?>

전통적인 학습 알고리즘(로지스틱 회귀분석, 서포트 벡터 머신) 은 데이터 양에 따라 performance 가 향상되지만 성능이 정체기에 이름.

<u>좋은 성능 발휘 조건</u>

1. 많은 데이터를 이용하기 위한 큰 신경망 필요.

2. 많은 데이터 필요

많은 은닉 유닛. 많은 파라미터, 많은 데이터.

-데이터와 계산.

시그모이드 함수를 사용하면 경사가 0인 부분에서 문제가 발생한다. 활성화 함수를 relu로 바꾸면 양수에서 경사가 다 1로 같으므로 경사가 0으로 수렴할 가능성이 적음.

-빠른 계산

<이진분류>- 신경프로그래밍의 기초

신경망을 학습하는 계산 과정: 정방향패스/ 역전파

- 로지스틱회귀 이용. : 이진분류를 위한 알고리즘

ex) 이미지 데이터가 주어졌을 때 rgb 3개의 픽셀 강도값을 나타내는 행렬을 한 열로 나타내기위해 특성 벡터 x를 정의하고, 특성벡터 x로 y가 0인지 1인지 예측.

<변수 설명>

(x,y)에서 x는 n_x차원 상의 특성벡터, y는 0or1 , m개의 훈련샘플

x 행렬은 x1을 한 칼럼으로, x2를 하나의 칼럼으로.. xm까지.

x는 n_x * m 행렬 (python에서는 X.shape로 알 수 있음)

y 값도 칼럼으로. y1, y2, .. 해서 1*m 행렬

(이처럼 각각의 열로 놓는 것이 유용.)

<로지스틱 회귀분석>: 출력 레이블이 0or 1일 때 (이진분류)

- y hat 을 알려주는 알고리즘.

(항상 0과 1 사이어야함, 그래서 선형회귀로 할 수 없음(전치행렬을 곱하면 1보다 큰 결과가 나오는 경우 많음(??))

- 시그모이드 함수를 적용함. 0부터 1까지 매끈한 s자형의 모형. 1/(1+e^(-z))

z가 아주 크면 시그모이드는 1에 가깝고, z가 아주 작거나 큰 음수면 시그모이드는 0에 수렴.

- y가 1일 확률을 잘 예측하게 하기 위해 파라미터 w와 b를 학습

(b는 인터셉트, b와 w를 분리하는 것이 편함)

<비용함수와 손실함수>

·◌ "y hat과 y 의 제곱오차의 반" 공식은 최적화 함수가 볼록하지 않아서 여러 개의 지역 최적값을 가지고 있어서 경사하강법이 전역최소값을 찾지 못할 수도 있기 때문에 로지스틱에서는 잘 사용하지 않음.

손실함수

★ 로지스틱 회귀에서는 –(y*log(y의 예측값)+(1-y)*log(1-y의 예측값))라는 손실함수 이용

1. y 가 1일 때: -log yhat : log(y hat)이 커져야 loss가 작아지므로 y hat이 1에 수렴하길 원함
2. y가 0일 때 : (1-y)log(1-y hat) : log(1-y hat)이 커야 loss가 작아지므로 y hat이 0에 수렴하길 원함

비용함수: 훈련세트 전체에 대해 얼마나 잘 추측되었는지 측정하는 함수

J= 손실함수를 각각의 훈련샘플에 적용한 값을 합들의 평균(m)으로 나눈 값

손실함수가 하나의 훈련 샘플에 적용. 비용함수는 매개변수의 비용처럼 작용.

목적 : 손실함수를 최소화하는 매개변수 w와 b를 찾는 것

비용함수를 가장 작게 만드는 w와 b를 찾는 것

<경사하강법>

볼록하지 않은 함수는 지역 최적값이 여러 개이기 때문에 볼록한 비용함수를 로지스틱 회귀에 사용한다.

가장 가파른 내리막 방향으로 한 단 계 내려간다. 이 과정을 반복하여 전역 최적값에 도달하게 된다.

계속해서 w-알파*J(w)의 미분계수(함수의 기울기)(dw)를 곱한 값으로 w를 갱신한다.

알파: 학습률(한 단계의 크기를 결정)

w-알파*dw

b-알파*dw

<계산그래프>

특정한 출력값 변수를 최적화하고싶을 때 유용하다.

도함수 계산할 때는 반대로 (오른쪽에서 왼쪽으로) 계산한다.

<계산그래프로 미분>

연쇄법칙

-역방향으로 도함수 계산.

-정방향 계산으로 비용함수 계산

<로지스틱 회귀에서 미분>

da=-y/a+(1-y)/(1-a)

dz=dL/dz=a-y=dL/da * da/dz

w와b갱신

w1=w1-alpha*dw1 로 갱신

m개의 훈련 샘플에 적용

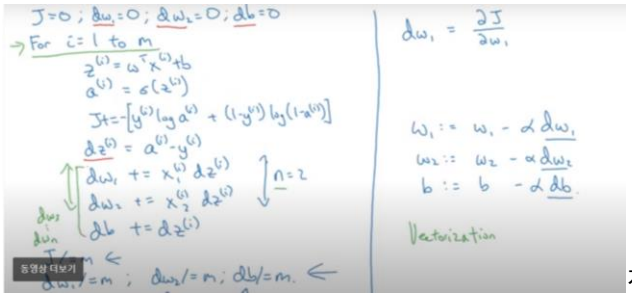
w1에 대한 전체 비용함수의 도함수는 w1의 각 손실 항 도함수의평균

⇨ 전체적인 경사를 구할 수 있음

하지만 for 문 2개 만들어야함

n개의 특성을 반복하는 for 문 필요. 비효율적

⇨ 벡터화로 for 문 제거



전