머신러닝 = 기계가 스스로 학습하게 하는 것

머신러닝 종류

supervised learning : 데이터에 정답이 있고 예측 모델을 만들 떄

unsupervised learning : 데이터에 정답이 없는 경우 > 비슷한 것끼리 분류

Reinforcement Learning : 게임가르친다 벽돌깨기 하나 깨면 +1 공 잃으면 -10 점수 높이

는 방향으로학습

적절한 가중치와 편향 값를 구해가는 과정 > 머신러닝

가중치(weight) 편향(bias)

딥러닝 > 신경망 구성

데이터와 예측값 사이에 hidden layer만들기

데이터와 hidden layer node를 묶어 perception

스크린샷, 텍스트, 원, 도표이(가) 표시된 사진

자동 생성된 설명  
layer : 생각을 저장하는 공간들

자료들의 특징을 구분해 처리할 수 있음.

텍스트, 스크린샷, 차량, 육상 차량이(가) 표시된 사진

자동 생성된 설명

딥러닝 : 사람의 가이드 없이 차의 특징 추출, 이미지를 잘게 잘라서 특성들을 추출한 후 구분

텍스트, 폰트, 그래픽, 스크린샷이(가) 표시된 사진

자동 생성된 설명

평균제곱오차의 수식(loss function)

텍스트, 폰트, 타이포그래피, 디자인이(가) 표시된 사진

자동 생성된 설명

파이썬의 tensorflow활용

위의 수능성적 예측 모델은 결국 데이터와 가중치의 곱이므로 hidden layer가 없어도 결과가 동일

이를 해결하기 위해 활성함수 (activation function) 활용 >> 활성함수가 없으면 딥러닝 x

ex) sigmoid 1/(1+e^(-x))

텍스트, 스크린샷, 라인, 도표이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 스크린샷, 번호이(가) 표시된 사진

자동 생성된 설명 텍스트, 폰트, 스크린샷, 그래픽이(가) 표시된 사진

자동 생성된 설명 여러 활성함수들

>> 비선형적 예측 가능하게함

텍스트, 스크린샷, 라인, 도표이(가) 표시된 사진

자동 생성된 설명

비선형성을 예측한 경우

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

loss (E) 를 최소화 하는 가중치(w) 찾기 > 경사 하강법

경사하강법 : 현재 w1값에서의 3접선의 기울기를 w1에서 빼기

폰트, 텍스트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

스크린샷, 도표, 그래프, 라인이(가) 표시된 사진

자동 생성된 설명

running rate 의 역할 local minima를 피해 최소값 찾기 >> 실험적으로 찾아야함

learning rage (a) optimizer >> 실험적으로 결정 주로 Adam 사용

* Momentum 가속도를 유지하자
* AdaGrad : 자주변하는 w는 작게, 자주 변하면 크게
* RMSProp : AdaGrad인데 제곱
* AdaDelta : Adagrad인데 a너무 작아져서 학습 안되는거 방지해줌
* Adam : RMSProp + Momentum

딥러닝 예제 정리

딥러닝 과정

1. 모델 만들기

2. 데이터처리

3. 학습

4. 결과 확인 후 여러 요인(활성함수, 손실함수, iteration횟수 등)변경하며 성능향상

활성함수

Tanh, sigmoid, relu, softmax,leakRelu

ReLU는 f(x) = max(0, x)

Leakrelu는 relu의 dead neurons problems를 해결하기 위해 음수에서도 약간의 기울기(~0.01)를 유지함

Softmax는 분류문제의 출력층에서 사용 벡터를 입력으로 받아 각 클래스에 대하여 확률분포 출력 e^x함수에 출력값들 넣은 뒤 각 비율 출력

Optimizer(learning rate 수정 함수)

adam,adagrad,agagelta,rmsprop,sgd ...

Adam: Adaptive Moment Estimation의 약자로, 모멘텀과 RMSprop의 장점을 결합한 알고리즘입니다. 학습률을 각 매개변수에 독립적으로 조정합니다.

Adagrad: 각 매개변수에 대해 개별적인 학습률 조정을 수행합니다. 크게 업데이트되지 않은 매개변수에는 더 큰 학습률을 적용합니다.

Adadelta: Adagrad의 확장으로, 과거 모든 그래디언트를 누적하는 대신 고정 크기의 윈도우를 사용하여 그래디언트의 누적을 제한합니다.

RMSprop: 가중치 갱신 시 이전 그래디언트가 얼마나 영향을 미칠지를 조정하여 학습 경로를 부드럽게 합니다.

SGD (Stochastic Gradient Descent): 매 반복마다 임의의 데이터를 사용하여 그래디언트를 계산합니다. 기본적이지만 여전히 효과적인 방법입니다.