* **Logistic Regression**

분류 -> Training data 특성과 관계 등을 파악 한 후에, 미지의 입력 데이터에 대해서 결과가 어떤 종류의 값으로 분류될 수 있는지 예측하는 것

지도이(가) 표시된 사진

자동 생성된 설명

최적의 직선을 training data로 찾고 -> 그 직선을 기준으로 1 or 0으로 분류

정확도가 높고 deep learning에서 기본 component로 사용되고 있음

스크린샷이(가) 표시된 사진

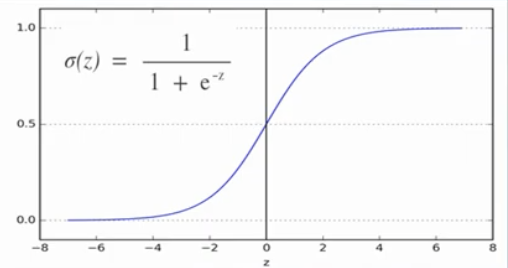
자동 생성된 설명

sigmoid 함수를 이용해

개체, 시계이(가) 표시된 사진

자동 생성된 설명 -> 시계, 측정기이(가) 표시된 사진

자동 생성된 설명



아래와 같이 log 들어간 저 식 E(W,b) -> classification의 손실함수 -> cross-entropy

스크린샷이(가) 표시된 사진

자동 생성된 설명

Cross-entropy를 이용해서 linear regression에서와 같이 W,b를 구할 수 있다 구할 수 있다

스크린샷이(가) 표시된 사진

자동 생성된 설명

* 지금 하는 과정이 Loss function의 값을 낮추기 위해 최적의 W,b를 구하기 위한 것이다!

입력 -> 임의의 W,b를 가정하여 trainig data를 이용하여 손실함수[ E(W,b) ]를 계산한다 -> 손실함수의 값이 최소값이라면 학습 종료

스크린샷이(가) 표시된 사진

자동 생성된 설명

훈련: 입력 데이타(10\*1)와 W를 행렬곱해서 바이어스 b를 더해서 Z를 만들고 sig(z)해서 Y값을 만들고 정답과 비교하면 오차를 줄이는것

스크린샷이(가) 표시된 사진

자동 생성된 설명

훈련: 입력 데이타(9x2)가 2개라 W가 2x1행렬로 됨 이거 행렬곱해서 b더해준 후 임의의 직선 값 Z를 만들어서 값을 합쳐주고 그 Z를 sig(Z)해서 Y값을 구하고 그 Y값과 정답을 비교한다. 손실함수 값이 가장 낮은 W,b를 찾는 것.

W1(예습), W2(복습) -> W1 > W2라면 예습시간이 합격률에 더 영향을 많이 미침