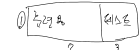


머신러닝?

2022년 11월 2일 월요일 오전 10:38



훈련 → model → 평가
test

회귀분석 (Regression) VS
분류 (Classification)



W := $\alpha \frac{\text{cost}}{\text{data}}$

평가치: α : 하이퍼 파라미터
W, b : 값을 update (cost)

회귀분석 (Regression)
- 지도학습 - labeling 가능 (예: 가격)
- 비지도학습 - labeling 불가능 (예: 패턴)

독립변수 X가 여러개 → 공동변수 Y
함수 X → $ax + b$

confusion matrix

	실제 True	실제 False
예측 True	TP (True Positive)	FP (False Positive)
예측 False	FN (False Negative)	TN (True Negative)

정확도: 모든의 정확성 비율
→ 분류한 모든의 비율

정밀도: Precision
TP / (TP + FP)

재현율: Recall
TP / (TP + FN)

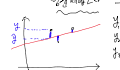
F1 점수: F1 Score
2 * (Precision * Recall) / (Precision + Recall)

가설함수 (hypothesis function)

비용함수 (cost function)

경사하강법 (gradient descent)

경사하강법 (gradient descent)
- $J = \frac{1}{2} \sum (y_i - \hat{y}_i)^2$
- $\hat{y}_i = w_0 + w_1 x_i$
- $J = \frac{1}{2} (w_0^2 + 2w_0 w_1 x_i + w_1^2 x_i^2 - 2w_0 y_i - 2w_1 x_i y_i + y_i^2)$
- $\frac{\partial J}{\partial w_0} = w_0 - y_i$
- $\frac{\partial J}{\partial w_1} = w_1 x_i - y_i x_i$
- $w_0 = \frac{1}{n} \sum (y_i - w_1 x_i)$
- $w_1 = \frac{\sum (x_i y_i - w_0 x_i)}{\sum x_i^2}$



Keras

SEQUENCE를 처리 할때 사용
→ "sequential" 구조는 가장 쉽다
(순차적인 layer를 쌓아서 학습시킬 것임)
학습 샘플이 2차 이상 → deep learning
입력데이터가 1차원인 경우라면 1차 순서만 리
앙그램, 임베딩, 어텐션, 트랜스포머

deep learning intro.

- 1) cost의 생성
 - 훈련 / 검증 / 테스트 세트 생성
 - 데이터셋의 분할 (랜덤 샘플링)
- 2) 층의 구조 (순서)
 - sequential 구조 이용하여 층 (layer) 생성
 - layer 추가
 - hidden layer의 층의 수를 늘려서
- 3) 학습률 설정
 - cost의 값이 줄어, 학습률 조정 (learning rate)
 - compile layer
- 4) 모델 평가
 - 훈련 → 모델 평가
 - 평가 결과
- 5) 하이퍼파라미터 튜닝
 - 하이퍼파라미터 튜닝 cost 관련
 - accuracy 향상
 - 반복학습 증가
- 6) 결과 평가
 - 테스트 데이터셋 평가
 - performance 평가
- 7) 결과 분석
 - 결과 분석하여 → 모델 → 평가
 - predict 결과

