♥ TITLE : 05_회귀 ♥ DATE : 2022.09.21

1 개요

- 여러 개의 독립변수(X, feature)와 한 개의 종속변수(Y, 결정 값) 간의 상관관계를 모델링하는 기법
- $Y = w_1X_1 + w_2X_2 + \cdots + w_nX_n$ (회귀계수)
- 유형 구분
  - i) 독립변수의 개수: 단일 회귀(1개) vs 다중 회귀(여러 개)
  - ii) 회귀계수의 결합: 선형 회귀(선형) vs 비선형 회귀(비선형)
- 예측값이 연속형 숫자 값
- **선형 회귀**
  - ↳ 규제 > 일반적인 선형 회귀의 과적합 문제를 해결하기 위해 회귀 계수에 패널티 값을 적용하는 것
  - ↳ 종류 > by 규제 방식
    - a. 일반 선형 회귀 [규제 적용 x (Residual Sum of Squares) 예측값과 실제 값의 RSS를 최소화
    - b. 릿지(Ridge): L_2 규제 적용
      - ↳ 상대적으로 큰 회귀 계수 값의 예측 영향도를 감소시키기 위해 회귀 계수값을 더 작게 만듦.
    - c. 라쏘(Lasso): L_1 규제 적용
      - ↳ 예측 영향력이 작은 피쳐의 회귀 계수를 0으로 만들어 회귀 예측 시 피처가 선택되서 않도록 하는 것
    - d. 엘라스틱넷(ElasticNet) [$L_1 + L_2$ 주로 피처가 많은 데이터 세트에 적용 ⇒ (L_1: 피처 개수 줄이기, L_2: 계수의 크기 조정

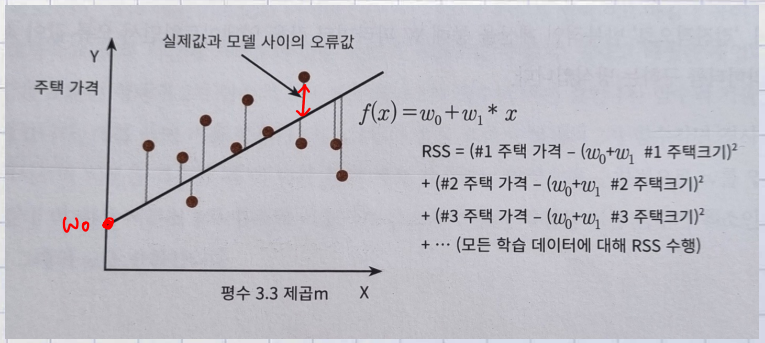
2 단순 선형 회귀

- 독립변수 1개, 종속변수 1개인 선형 회귀
- $\hat{Y} = w_0 + w_1 * X$ ($\hat{Y}$: 예측값, w_0: 절편, w_1: 회귀계수 (intercept))
- $Y_i = w_0 + w_1 * X + e_i$ (Y_i: 실제값, e_i: 오류값 (잔차))

① 최적의 회귀 모델
- 전체 데이터의 잔차의 합이 최소가 되는 모델
- 잔차의 합이 최소가 될 수 있는 최적의 회귀 계수를 가지는 모델 ⇒ RSS 최소화

② 오류 합의 계산
- MAE (Mean Absolute Error)
- RSS (Residual Sum of Square) ... $Error^2 = RSS$

③ 비용함수 (= 손실함수)
- $RSS(w_0, w_1) = \frac{\sum_{i=1}^{N} (y_i - (w_0 + w_1 x_i))^2}{N}$
- 머신러닝 회귀 알고리즘: 데이터를 계속 학습하면서 비용 함수가 반환하는 값을 지속해서 감소시키고 더 이상 감소하지 않는 최소의 오류 값 구하기

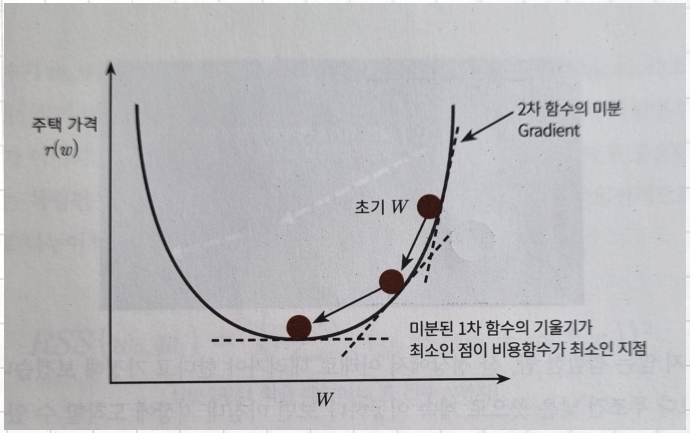


♥ TITLE : 5_회귀 ♥ DATE : 2022.09.22

3 경사 하강법

- 비용 함수가 최소가 되는 W 파라미터 구하기
- 점진적으로 반복적인 계산을 통해 W 파라미터 값을 업데이트하면서 오류 값이 최소가 되는 W 파라미터 구하기
  (e.g. |예측값 - 실제값|)
- 오류 값이 더 이상 작아지지 않으면 그 오류 값을 최소 비용으로 판단하고 그때의 W 값을 최적 파라미터로 반환
- 단계>
  i) w_0, w_1을 임의의 값으로 설정하고 첫 비용 함수의 값 계산 … RSS(w_0, w_1) 편미분
  ii) w_1을 $w_1 + \eta \times \frac{2}{N}\sum x_i \times (\text{실제} - \text{예측})$,
  w_0을 $w_0 + \eta \times \frac{2}{N}\sum (\text{실제} - \text{예측})$
  (η: 보정 계수 (learning rate))
  iii) 비용 함수의 값이 감소했다면 다시 Step2 반복
  → 주> 감소 X : 그때의 w_0, w_1을 구하고 중지

- 입력 배열 X에 대한 예측 배열 y_pred는 np.dot(X, w1.T) + w0 으로 구함 (e.g. "내적")
- 일반적으로 경사 하강법은 모든 학습 데이터에 대해 반복적으로 비용함수 최소화를 위한 값을 업데이트 ⇒ 수행 시간이 매우 오래 걸림
  ↳ 일부 데이터만 이용해 w가 업데이트되는 값을 계산하는 확률적 경사 하강법을 이용
  batch_size
- feature의 개수가 M개이면 그에 따른 회귀 계수는 M+1개이다.



$$\hat{Y} \quad X_{mat}$$

Feature 1 Feature 2 … Feature M

$$\begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} * \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}^T + w_0$$

내적 $[w_1 \ w_2 \ \cdots \ w_m]$

set a record

♥ TITLE : 5_회귀 ♥ DATE : 2022.09.25

4 Linear Regression 클래스

① Ordinary Least Squares

- 예측값과 실제 값의 RSS(Residual Sum of Squares)를 최소화해 OLS(Ordinary Least Squares) 추정 방식으로 구현한 클래스
- LinearRegression 클래스는 fit() 메서드로 X, y 배열을 입력 받으면 회귀 계수인 W를 coef_ 속성에 저장

i) 입력 파라미터

- **fit_intercept** : intercept(절편) 계산 여부 결정, boolean 값, default=True
  - False로 지정 시 intercept가 사용되지 않고 0으로 지정됨
- **normalize** : 회귀 수행 전 입력 데이터 세트 정규화, default = False
  - fit_intercept = False인 경우 해당 파라미터가 무시됨

ii) 속성

- **coef_** : fit() 메서드 수행 시 회귀 계수가 배열 형태로 저장하는 속성
  - shape = (target 개수, feature 개수)
- **intercept_** : intercept 값

- 다중 공선성
  - ↳ OLS 기반 회귀 계수 계산은 입력 피처의 독립성에 많은 영향을 받음
  - ↳ 해결>
    - i) 독립적인 주요 피처만 남기고 제거 or 규제 적용
    - ii) 매우 많은 피처가 다중 공선성 문제를 지닌 경우 PCA를 통한 차원 축소 수행도 고려할 수 O

② 회귀 평가 지표

- 실제 값과 예측값의 차이를 그냥 더하면 (+)와 (-)가 섞여 오류가 상쇄됨
  ⇒ 오류의 절댓값 평균, 제곱, 또는 제곱한 뒤 다시 루트를 씌운 평균값 이용

평가 지표	설명	수식
MAE	Mean Absolute Error(MAE)이며 실제 값과 예측값의 차이를 절댓값으로 변환해 평균한 것입니다.	$MAE = \frac{1}{n}\sum_{i=1}^{n}\lvert Yi - \hat{Y}i\rvert$
MSE	Mean Squared Error(MSE)이며 실제 값과 예측값의 차이를 제곱해 평균한 것입니다.	$MSE = \frac{1}{n}\sum_{i=1}^{n}(Yi - \hat{Y}i)^2$
RMSE	MSE 값은 오류의 제곱을 구하므로 실제 오류 평균보다 더 커지는 특성이 있으므로 MSE에 루트를 씌운 것이 RMSE(Root Mean Squared Error)입니다.	$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Yi - \hat{Y}i)^2}$
R^2	분산 기반으로 예측 성능을 평가합니다. 실제 값의 분산 대비 예측값의 분산 비율을 지표로 하며, 1에 가까울수록 예측 정확도가 높습니다.	$R^2 = \frac{\text{예측값 } Variance}{\text{실제값 } Variance}$

평가 방법	사이킷런 평가 지표 API	Scoring 함수 적용 값
MAE	metrics.mean_absolute_error	'neg_mean_absolute_error'
MSE	metrics.mean_squared_error	'neg_mean_squared_error'
R^2	metrics.r2_score	'r2'

작은 오류 값이 더 큰 숫자로 인식되도록 -1을 원래의 평가 지표에 곱함

set a record

♥ TITLE : 5_회귀 ♥ DATE : 2022.09.22

5 다항 회귀와 과대/과소적합

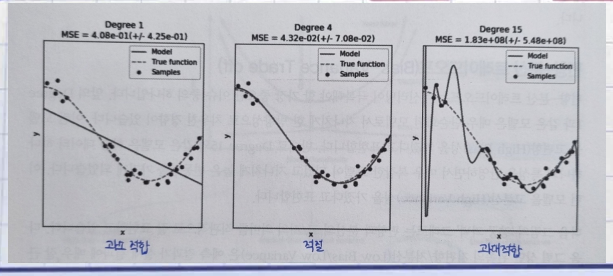
① 다항 회귀 이해

- 회귀가 독립변수의 단항식이 아닌 2차, 3차 방정식과 같은 다항식으로 표현되는 것
- ex> $y = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1 * x_2 + w_4 x_1^2 + w_5 x_2^2$
- ★ 다항 회귀는 선형 회귀 (≠비선형)
- Scikit-learn에는 다항 회귀를 위한 별도의 클래스는 X
  ⇒ 비선형 함수를 선형 모델에 적용시키는 방법을 이용 → PolynomialFeatures 클래스

- PolynomialFeature 클래스
  ↳ degree 파라미터를 통해 입력 받은 단항식 feature를 degree에 해당하는 다항식 feature로 변환 ⇒ fit( ) (적합), transform( ) (변형) 이용
  ↳ 변환된 feature에 선형 회귀 적용
  LinearRegression 클래스

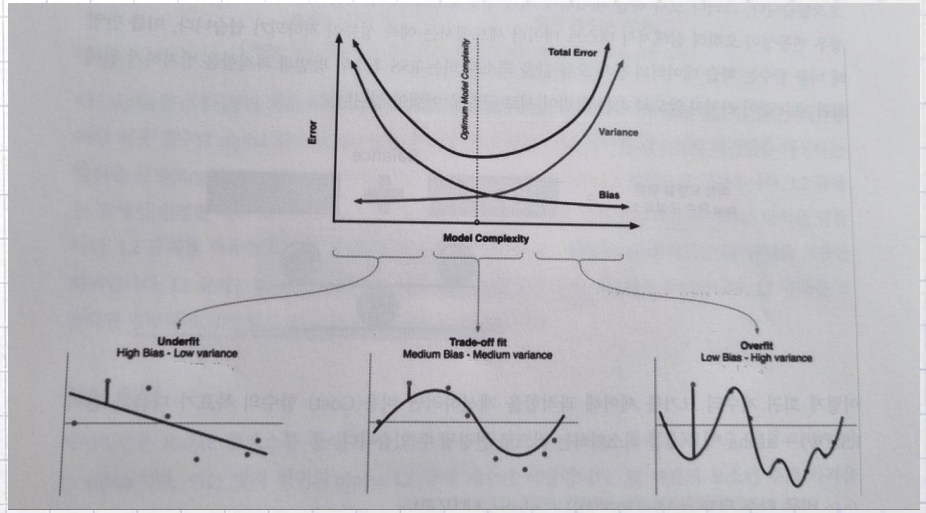
② 과소적합/과적합 이해

- 다항 회귀의 차수(degree)를 높일수록 학습 데이터에만 너무 맞춘 학습이 이뤄져서 테스트 데이터에서는 오히려 정확도가 떨어질 수도 O. ⇒ 과적합 문제
- 예측 곡선이 너무 단순하면(낮은 degree) 학습 데이터의 패턴을 제대로 반영하지 못함 ⇒ 과소적합 문제
- 좋은 예측 모델: 학습 데이터의 패턴을 잘 반영(과소적합(X))하면서도 복잡하지 않은(과대적합(X)) 균형잡힌 모델



③ 편향-분산 Trade off

- 편향과 분산은 한쪽이 높아지면 한 쪽이 낮아지는 경향이 O … trade-off
- 고편향 • 저분산 : 과소적합 가능성↑
  저편향 • 고분산 : 과적합 가능성↑



set a record

♥ TITLE : 5_회귀 ♥ DATE : 2022.09.23

6 규제 선형 모델

① 개요

i) 비용 함수의 목표
- RSS 최소화: 학습 데이터의 잔차 오류 최소화
- 회귀 계수의 크기 제어

⇒ $Min(RSS(W)) + alpha * \|W\|_2^2$

ii) alpha
- 학습 데이터 적합 정도와 회귀 계수 값의 크기 제어를 수행하는 튜닝 파라미터
- alpha = 0 : W가 커도 $alpha * \|W\|_2^2$ 가 0이 되어 비용 함수는 $Min(RSS(W))$
- alpha = ∞ : $alpha * \|W\|_2^2$ 도 무한대 ⇒ 비용 함수: W 최소화 하도록!

⇒ alpha를 0에서부터 지속적으로 증가시키면 회귀 계수의 값을 감소시킬 수 O → 규제

iii) 규제의 종류
- a. L2 규제 (↗ 릿지): W의 제곱에 대해 페널티 부여 ⇒ 회귀 계수의 크기 감소
- b. L1 규제 (↘ 라쏘): W의 절댓값에 대해 페널티 부여
  - 불필요한 회귀 계수를 급격히 감소시켜 0으로 만들고 제거 ⇒ 피처 선택

② 릿지 회귀
- Sklearn의 linear_model 패키지의 Ridge 이용
- ex> from sklearn.linear_model import Ridge
- 회귀 수행: ridge = Ridge(alpha = 10) # 릿지 객체 생성
- 회귀계수(w): Ridge.coef_
- alpha 값 증가 시 회귀 계수가 지속적으로 감소
  but 회귀 계수가 0이 되지는 X

③ 라쏘 회귀

④ 엘라스틱넷 회귀
- L2 규제 (↗ 릿지) ⊕ L1 규제 (↘ 라쏘) ⇒ 수행시간이 상대적으로 오래 걸린다.
- 규제: $a * L1 + b * L2$
- alpha = a+b , l1_ratio: a/(a+b)
- ElasticNet 클래스를 통해서 구현

⑤ 선형 회귀 모델을 위한 데이터 변환
- 선형 모델
  - 일반적으로 feature와 target 간에 선형 관계가 있다고 가정하고, 최적의 선형함수를 찾아내 결과 예측
  - feature와 target 데이터의 분포가 정규 분포 형태이기를 기대

⇒ 데이터에 대한 스케일링/정규화 작업 수행

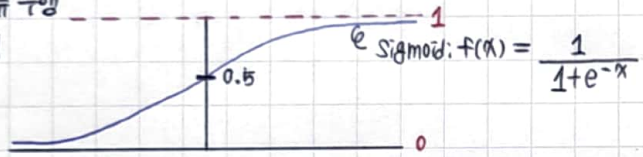
i) feature data 변환 (3가지 방법)
- a. StandardScaler 클래스를 이용하여 평균이 0, 분산이 1인 표준정규분포를 가진 데이터 세트로 변환 or MinMaxScaler 클래스를 이용하여 최솟값이 0이고 최댓값이 1인 값으로 정규화 수행
- b. 스케일링/정규화를 적용한 데이터 세트에 다시 다항 특성을 적용하여 변환
  - ↳ if> 피처의 개수가 수↓: 다항 변환으로 생성되는 feature의 개수↑ : 과적합 이슈
- c. 로그(log) 변환

set a record ii) target data : 로그 변환

♥ TITLE : 5_회귀 ♥ DATE : 2022.09.24

7 로지스틱 회귀

- 선형 회귀 방식을 분류에 적용한 알고리즘
- 학습을 통해 시그모이드 함수 최적선을 찾고 이 함수의 반환 값을 확률로 간주해 확률에 따른 분류 수행
- S자 커브 형태



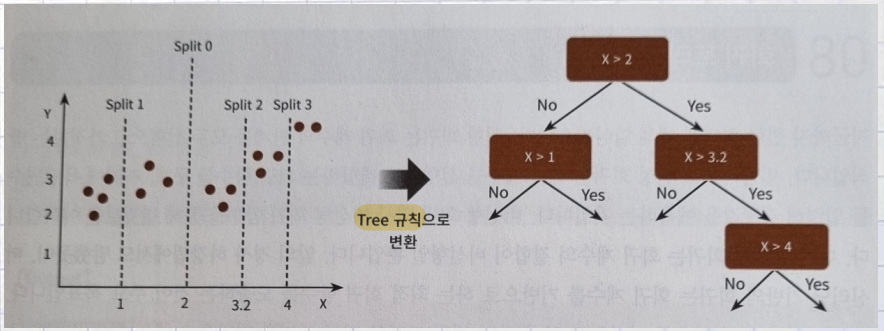
- 데이터의 정규 분포도에 따라 예측 성능 영향을 받을 수 O ⇒ 표준 스케일링 적용
- 사이킷런 LogisticRegression 클래스의 주요 파라미터
  - i) penalty : 규제의 유형 (default = 'l2')
  - ii) C : 규제 강도 제어, $\frac{1}{alpha}$
- 이진 분류 예측 성능 good, 희소한 데이터 세트 분류에도 뛰어난 성능

8 회귀 트리

- 트리 기반 회귀 방식 ⇒ 회귀를 위한 트리를 생성하고 이를 기반으로 회귀 예측 수행
- 리프 노드에 속한 데이터 값의 평균을 구해 회귀 예측값 계산
- 단계>
  - i) X값의 균일도를 반영한 지니 계수에 따라 분할
  - ii) 트리 분할이 완료되면 리프 노드에 소속된 데이터 값의 평균값을 구해 최종 결정 값으로 할당
- CART (Classification And Regression Trees) 알고리즘 기반

p. 337 ㉮

i) 단계 1



ii) 단계 2