<chapter 5>

1. 회귀소개

- 회귀 분석이란?

데이터 값이 평균과 같은 일정한 값으로 돌아가려는 경향을 이용한 통계학 기법

여러 개의 독립변수와 한 개의 종속변수 간의 상관관계를 모델링하는 기법을 통칭함

독립변수의 값에 영향을 미치는 것을 회귀 계수

머신러닝 회귀 예측의 핵심은 주어진 피처와 결정 값 데이터 기반에서 학습을 통해 최적의 회귀 계수를 찾아내는 것

독립변수 개수	회귀 계수의 결합
1개: 단일 회귀	선형: 선형 회귀
여러 개: 다중 회귀	비선형: 비선형 회귀

〈 회귀 유형 구분 〉

선형 회귀가 가장 많이 사용되는데, 실제 값과 예측 값의 차이를 최소화하는 직선형 회귀선을 최적화하는 방식

2. 단순 선형 회귀를 통한 회귀 이해

단순 선형 회귀: 독립변수도 하나, 종속변수도 하나인 선형 회귀. 1차 함수식으로 모델링 가능

실제 값과 회귀 모델의 차이에 따른 오류 값을 남은 오류, 잔차라고 부름

최적 회귀 모델을 만든다는 것 = 전체 데이터의 잔차 합이 최소가 되는 모델을 만든다

오류 값은 +나 -로, 합을 구했다가 오류 합이 크게 줄어들 수 있으므로 합을 구할 때는 절대값을 취하거나(Mean Absolute Error), 오류 값의 제곱을 구해서 더하는 방식(RSS)을 취함

미분과 같은 계산을 편리하게 하기 위해선 RSS 방식으로 구함

3. 비용 최소화하기 – 경사하강법

비용함수가 최소가 되는 w 파라미터를 구하는 방법

점진적으로 반복적인 계산을 통해 w 파라미터 값을 업데이트하면서 오류 값이 최소가 되는 w 파라미터를 구하는 방식

- Step 1: w_1, w_0를 임의의 값으로 설정하고 첫 비용 함수의 값을 계산합니다.
- Step 2: w_1을 $w_1 + \eta \frac{2}{N} \sum_{i=1}^{N} x_i * (\text{실제값}_i - \text{예측값}_i)$, w_0을 $w_0 + \eta \frac{2}{N} \sum_{i=1}^{N} (\text{실제값}_i - \text{예측값}_i)$으로 업데이트한 후 다시 비용 함수의 값을 계산합니다.
- Step 3: 비용 함수의 값이 감소했으면 다시 Step 2를 반복합니다. 더 이상 비용 함수의 값이 감소하지 않으면 그때의 w_1, w_0를 구하고 반복을 중지합니다.

경사 하강법은 모든 학습 데이터에 대해 반복적으로 비용함수 최소화를 위한 값을 업데이트 하기 때문에 수행 시간이 오래 걸린다는 단점이 존재함. 때문에 실전에서는 확률적 경사 하강법을 이용함

확률적 경사 하강법: 전체 입력 데이터로 w가 업데이트되는 값을 계산하는 것이 아니라 일부 데이터만을 이용해 w가 업데이트되는 값을 계산하므로 경사 하강법에 비해서 빠른 속도를 보장

4. 회귀 평가 지표

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'

5. 다항 회귀와 과적합/과소적합

- 다항 회귀

모든 독립변수와 종속변수의 관계가 일차 방정식 형태로 표현된 회귀식은 아님

회귀가 독립 변수의 단항식이 아닌 2차, 3차 방정식과 같은 다항식을 표현되는 것이 다항회귀

사이킷런은 다항 회귀를 위한 클래스를 명시적으로 제공하지 않음

대신 다항 회귀 역시 선형 회귀이기 때문에 비선형 함수를 선형 모델에 적용시키는 방법을 사용해 구현

- 다항 회귀를 이용한 과소적합 및 과적합 이해

다항 회귀는 피처의 직선적 관계가 아닌 복잡한 다항 관계를 모델링할 수 있음

다항식의 차수가 높아질수록 매우 복잡한 피처 간의 관계까지 모델링이 가능

다항 회귀의 차수를 높일수록 학습 데이터에만 너무 맞춘 학습이 이뤄져서 정작 테스트 데이터 환경에서는 예측 정확도가 떨어짐 -> 차수가 높아질수록 과적합 문제 크게 발생

좋은 예측 모델: 학습 데이터의 패턴을 지나치게 단순화한 과소적합 모델도 아니고 모든 학습 데이터의 패턴을 하나하나 감안한 지나치게 복잡한 과적합 모델도 아닌, 학습 데이터의 패턴을 잘 반영하면서도 복잡하지 않은 균형 잡힌 모델

- 편향-분산 트레이드 오프

과소적합된 모델: 고편향성을 가짐

과적합된 모델: 고분산성을 가짐

일반적으로 편형과 분산은 한쪽이 높으면 한쪽이 낮아지는 경향이 있음.

편향과 분산이 서로 트레이드오프를 이루면서 오류 cost 값이 최대로 낮아지는 모델을 구축하는 것이 가장 효율적인 머신러닝 예측 모델을 만드는 방법

6. 규제 선형 모델 – 릿지, 라쏘, 엘라스틱넷

- 규제 선형 모델

이전까지 선형 모델의 비용 함수는 RSS를 최소화하는 것에만 고려함. 그러다 보니 학습 데이

터에 지나치게 맞추게 되고, 회귀 계수는 쉽게 커짐.

이를 반영해 비용함수는 학습 데이터의 잔차 오류 값을 최소로 하는 RSS 최소화 방법과 과적합을 방지하기 위해 회귀 계수 값이 커지지 않도록 하는 방법이 균형을 이뤄야 함

- 선형 회귀 모델을 위한 데이터 변환

선형 모델은 일반적으로 피처와 타깃 값 간에 선형의 관계가 있다고 가정하고, 최적의 선형 함수를 찾아내 결과값을 예측함

피처 값과 타깃 값의 분포가 정규분포 형태를 매우 선호함

특히 타깃 값의 경우 정규 분포 형태가 아니라 특정 값의 분포가 치우친 왜곡된 형태의 분포일 경우 예측 성능에 부정적인 영향을 미칠 수 있음

따라서 선형 회귀 모델을 적용하기 전에 데이터에 대한 스케일링/정규화 작업을 수행하는 것이 일반적

1) Standardscaler or minmaxscaler 이용하기

2) 1번을 통해 예측 성능 향상이 없을 경우 다항 특성을 적용해서 변환

3) 로그 변환

타깃 값의 경우 일반적으로 로그 변환을 적용해서 예측 성능을 향상시킴

7. 로지스틱 회귀

선형 회귀 방식을 분류에 적용한 알고리즘

학습을 통해 선형 함수의 회귀 최적선을 찾는 것이 아니라 시그모이드 함수 최적선을 찾고 이 시그모이드 함수의 변한 값을 확률로 간주해 확률에 따라 분류를 결정하는 것

8. 회귀 트리

결정 트리와 같이 트리를 기반으로 하는 회귀 방식

분류 트리와 크게 다르지 않고 리프 노드에서 예측 결정 값을 만드는 과정에서 차이가 있는데, 분류 트리가 특정 클래스 레이블을 결정하는 것과는 달리 회귀 트리는 리프 노드에 속한 데이터 값의 평균값을 구해 회귀 예측 값을 계산함