1.2 수식

1.2.1 평균 제곱 오차(MSE) 도출

회귀 문제에서 모델이 얼마나 잘 예측했는지를 수치로 평가하려면, 예측값과 실제값의 차이를 계산해야 합니다.

이 차이를 바탕으로 정의되는 가장 널리 쓰이는 손실 함수는 **평균 제곱 오차 (Mean Squared Error)** 입니다.

1.2.1.1 목적: 오차의 제곱을 평균 내자

각 샘플 $x^{(i)}$ 에 대해 예측값은 $h_\theta(x^{(i)}) = \theta^T x^{(i)}$, 실제값은 $y^{(i)}$입니다.

따라서 예측 오차는 $\theta^T x^{(i)} - y^{(i)}$이며, 이를 제곱해서 더한 뒤 평균을 내면 전체 오차를 대표할 수 있습니다.

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left(\theta^T x^{(i)} - y^{(i)}\right)^2$$

1.2.2 경사 하강법을 위한 미분 유도

모델의 학습이란 결국 이 비용 함수 $J(\theta)$를 최소화하는 θ를 찾는 것입니다.

이를 위해 우리는 **경사 하강법 (Gradient Descent)** 알고리즘을 적용합니다.

그 핵심은 각 파라미터 θ_j에 대해 **기울기(gradient)** 를 구하는 것입니다.

1.2.2.1 비용 함수에 대한 편미분

θ_j대해 편미분하면:

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m} \sum_{i=1}^{m} (\theta^T x^{(i)} - y^{(i)}) \cdot x_j^{(i)}$$

1.2.2.2 경사 하강법 업데이트 식 유도

$$\theta_j := \theta_j - \alpha \cdot \frac{\partial J(\theta)}{\partial \theta_j}$$

즉, 각 파라미터는 그에 대한 비용 함수의 기울기만큼, **오차가 줄어드는 방향으로** 이동합니다.

> 학습률 α 는 너무 크면 발산, 너무 작으면 수렴이 느려서, 적절한 값 설정이 중요합니다.

1.2.3 전체 수식 요약

비용함수

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (\theta^T x^{(i)} - y^{(i)})^2$$

기울기

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m} \sum_{i=1}^{m} (\theta^T x^{(i)} - y^{(i)}) x_j^{(i)}$$

경사 하강법 업데이트 식

$$\theta_j := \theta_j - \alpha \cdot \frac{\partial J(\theta)}{\partial \theta_j}$$

1.2.4 정규화 회귀 – 미분 관점에서의 수식 유도

머신러닝에서 모델의 복잡도가 지나치게 커져 학습 데이터에 과적합(overfitting)되는 것을 막기 위해 **정규화(regularization)** 를 도입합니다.

정규화는 손실 함수에 **패널티 항**을 추가하여 모델이 너무 큰 계수를 갖지 못하게 억제합니다.

1.2.4.1 Ridge 회귀 (L2 정규화)

Ridge 회귀는 제곱합을 기반으로 정규화하는 방법입니다.

비용 함수

$$J_{\text{ridge}}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(\theta^T x^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$$

편미분 (Gradient)

$$\frac{\partial J}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}(\theta^T x^{(i)}) - y^{(i)})x_j^{(i)} + \frac{\lambda}{m}\theta_j$$

- 기존의 MSE 오차 기울기에 **정규화 항의 기울기** $\frac{\lambda}{m}\theta_j$ 가 추가됩니다.
- 이 항은 θ_j의 절대 크기에 비례하여, θ_j를 **0 방향으로 부드럽게 수축**시킵니다.
- Ridge는 **모든 파라미터를 조금씩 줄이며**, 과도하게 커진 계수를 억제하는 방식입니다.
- 하지만 $\theta_j = 0$이 되도록 강제하지는 않기 때문에 **희소성(sparsity)은** 유도하지 않습니다.

1.2.4.2 Lasso 회귀 (L1 정규화)

Lasso 정규화는 계수의 **절댓값 합**을 패널티로 더함으로써,

일부 계수가 **완전히 0이 되도록** 만들어 **특성 선택(feature selection)** 효과를 냅니다.

비용 함수

$$J_{textlasso}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(\theta^T x^{(i)} - y^{(i)})^2 + \frac{\lambda}{m}\sum_{j=1}^{n}|\theta_j|$$

편미분 (서브그라디언트)

L1 정규화는 절댓값이 포함되어 있어 $\theta_j = 0$에서 미분이 정의되지 않으므로,

서브그라디언트(subgradient) 를 사용합니다.

$$\frac{\partial J}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}(\theta^T x^{(i)} - y^{(i)})x_j^{(i)} + \frac{\lambda}{m}\cdot \text{sign}(\theta_j)$$

- $sign(\theta_j)$는 $\theta_j > 0$일 때 $+1$, $\theta_j < 0$일 때 -1, $\theta_j = 0$일 땐 $[-1, +1]$ 범위입니다.
- 이 항은 일정한 크기의 상수 힘으로 θ_j를 **0 방향으로 끌어당기며**, θ_j가 작을수록 쉽게 0이 되도록 유도합니다.
- 결과적으로 Lasso는 일부 계수를 완전히 0으로 만들어, **희소한 모델(sparse model)** 을 형성합니다.

1.2.5 정규화 회귀 – 기하학적 관점에서의 해석

정규화 회귀는 **최적화 문제에 제약 조건을 추가한 형태**로 이해할 수 있으며, 이를 통해 Ridge와 Lasso의 동작 차이를 **기하학적으로 직관적**으로 이해할 수 있습니다.

1.2.5.1 제약 조건 최적화 형태로의 변환

Ridge나 Lasso에서 손실 함수에 정규화 항을 더하는 방식은, 아래와 같이 **제약 조건을 둔 최적화 문제**와 동치입니다.

> $L(\theta)$가 가장 작아지도록 하는 파라미터 θ 를 찾아라. 동시에 $R(\theta) \le t$ 라는 제약 조건을 만족해야 한다.

$$\min_{\theta} \ L(\theta) \quad \text{subject to} \quad R(\theta) \le t$$

- $L(\theta)$: 손실 함수 (예: 평균제곱오차)
- $R(\theta)$: 정규화 항 (규제 함수) $\|\theta\|_2^2$, $\|\theta\|_1$
- t: 제약의 강도를 조절하는 상수, 허용 가능한 복잡도의 상한선
- 해를 찾을 때, 반드시 $R(\theta) \le t$ 범위 안에서만 θ를 탐색합니다.

Ridge의 제약 영역

$$R(\theta) = \sum_{j=1}^{n} \theta_j^2 \quad \Rightarrow \quad \text{제약 영역은 원 또는 구의 형태}$$

Lasso의 제약 영역

$$R(\theta) = \sum_{j=1}^{n} |\theta_j| \quad \Rightarrow \quad \text{제약 영역은 마름모 또는 정육면체 형태}$$

1.2.5.2 손실 함수 등고선과 제약 영역의 교점

손실 함수 $L(\theta)$는 일반적으로 **타원 형태의 등고선**을 갖습니다. 최적해는 이 등고선이 **제약 영역의 경계**와 처음 **접하는 지점**에서 결정됩니다.

Ridge의 경우

- 원형 제약 영역은 매끄럽기 때문에 등고선이 일반적으로 중심 근처 **곡면**과 접함
- 결과적으로 대부분의 θ_j는 **작지만 0이 아님**

Lasso의 경우

- 마름모 제약 영역은 **꼭짓점이 축 방향 위에 있음**
- 등고선이 꼭짓점과 자주 접하게 되어, 일부 $\theta_j = 0$이 되는 **희소 해(sparse solution)** 가 도출됨