

선형 회귀

다중 선형 회귀(Multiple Linear Regression)

수치형(실수형) 설명변수 X 와 연속형 숫자로 이뤄진 종속변수 Y 의 관계를 선형으로 가정하고 이를 잘 표현할 수 있는 회귀계수 β 를 데이터로 추정하는 모델

다중 선형 회귀 모델

$$y = f(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n + \varepsilon$$

ε : 오차항, β : 회귀계수(찾아야 하는 것)

다중 선형 회귀 모델 방정식

n 개의 데이터, k 개의 설명변수(x_{11}, \dots, x_{nk}), k 개의 회귀계수(β_0, \dots, β_k)

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & \dots & x_{1k} \\ 1 & x_{21} & \dots & x_{2k} \\ \vdots & \vdots & & \vdots \\ 1 & x_{n1} & \dots & x_{nk} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

$$\vec{y} = X\vec{\beta} + \vec{\varepsilon}, \vec{\varepsilon} \sim N(E(\vec{\varepsilon}), V(\vec{\varepsilon})), E(\vec{\varepsilon}) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, V(\vec{\varepsilon}) = \sigma^2 I$$

\vec{y} : 라벨, X : 입력, $\vec{\beta}$: 회귀계수, $\vec{\varepsilon}$: 오차항

회귀 계수 결정법

1. Direct Solution

오차 제곱합: 실제값(Y)과 모델 예측값(\hat{Y})의 차이를 최소로 하는 값을 회귀 계수로 선정 $error = loss = \|Y - \hat{Y}\|_2$ 작도록 하는 값 $\beta_0 \rightarrow \beta_k$

Sol. 회귀 계수에 대해 미분한 식 = $O(\text{극소점})$ 을 놓고 풀면 최적 계수들의 명시적인 해 구할 수 있다. $error$ 식의 미분값 = 0

X 와 Y 데이터만으로 회귀계수 구할 수 있다: $\vec{\beta} = (X^T X)^{-1} X^T \vec{y}$

2. Numerical Search

경사하강법(Gradient Descent)

같은 방식을 반복하여 선형 회귀 계수를 구한다

목표: 어떤 함수 값(목적함수=비용함수=에러값)을 최소화

방법: 임의의 시작점 잡고, 해당 지점에서의 gradient를 구하고, gradient의 반대 방향으로 조금씩 이동하는 과정을 반복한다

조금씩 이동하는 과정의 정도: learning rate

경사하강법의 종류

1. Batch Gradient Descent(GD) = Vanilla Gradient Descent

파라미터 업데이트할 때마다 모든 학습 데이터를 사용해서 cost function의 gradient 계산

단점: 매우 낮은 학습 효율 보일 수 있다.

2. Stochastic Gradient Descent(SGD)

파라미터 업데이트할 때, 무작위로 샘플링된 학습 데이터를 하나씩만 사용해서 cost function의 gradient 계산

장점: 모델을 자주 업데이트하며, 성능 개선 정도를 빠르게 확인 가능
local minima에 빠질 가능성 줄일 수 있음

단점: 최소 cost에 수렴했는지 판단이 상대적으로 어려움

3. Mini Batch Gradient Descent

파라미터 업데이트할 때마다 일정량의 데이터를 무작위로 뽑아 cost function의 gradient 계산(GD와 SGD 개념의 혼합)

장점: SGD의 노이즈를 줄이면서, GD의 전체 배치보다 효율적

정규화

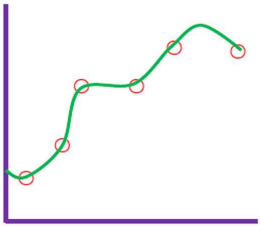
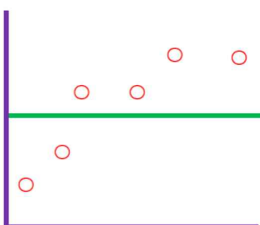
: 회귀계수가 가질 수 있는 값에 제약조건 부여하여 미래 데이터에 대한 오차 기대

미래에 대한 오차 기댓값 = 모델의 Bias와 Variance

정규화: variance를 감소시켜 일반화 성능 높이는 기법

(단, 이 과정에서 bias 증가할 수 있다)

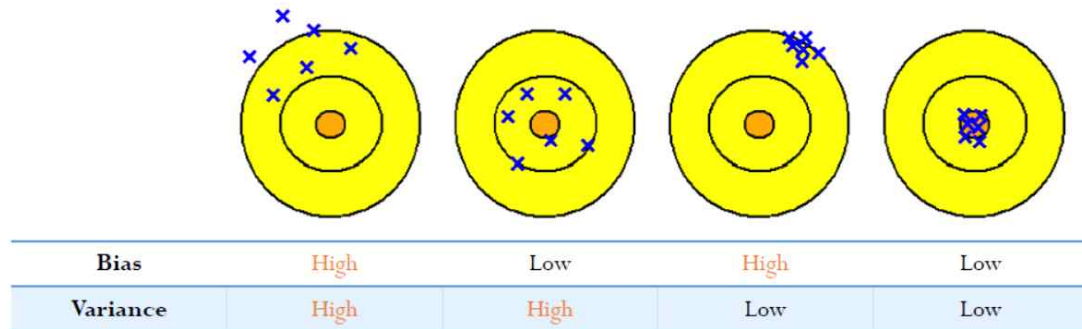
정규화 정도: $c = \frac{1}{\lambda}$

| | |
|---|---|
|  | Overfitting 학습 데이터는 잘 맞추지만, 미래 데이터가 조금만 바뀌어도 예측값이 들쭉날쭉할 수 있다. |
|  | 정규화의 결과로 학습 데이터에 대한 설명력을 다소 포기한 대신 미래 데이터 변화에 상대적으로 안정적인 결과를 나타낼 수 있다. |

Bias-Variance Decomposition

일반화(generalization) 성능을 높이는 정규화(regularization), 앙상블(ensemble) 기법의 이론적 배경

: 학습에 쓰지 않은 미래 데이터에 대한 오차의 기댓값을 모델의 bias와 variance로 분해하자



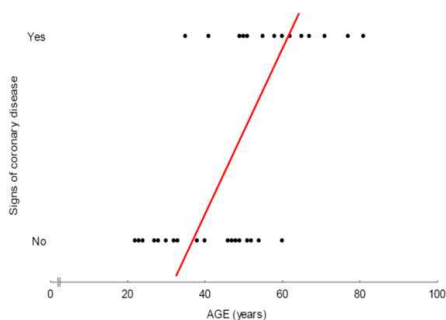
부스팅(Boosting): bias를 줄여 성능을 높이는 기법

라쏘 회귀(Lasso regression): variance를 줄여 성능을 높이는 기법

| Bias | Variance | | |
|------|----------|--|------------------------|
| Low | High | NN, SVM, kNN(small k) | 튜닝만 잘 하면 과녁을 제대로 맞출 모델 |
| High | Low | Logistic Repression, LDA, kNN(large k) | 데이터 노이즈에 강인한 모델 |

Why 선형 회귀는 선형 분류 문제 풀 수 X?

선형 회귀(수치형x연속형: 나이와 혈압)에서 선형 분류(수치형x범주형: 나이와 암 발병 여부)로 넘어가는 이유



: 범주형 숫자는 연속형 숫자와 달리

의미를 지니지 않는다

= 0(음성)과 1(양성)을 바꿔서

0(양성)과 1(음성)으로 해도 상관없다

∴ 분류 문제에 다중 선형 회귀 모델을 적용하지 못한다.

해결 방법: 범주형 숫자에 로지스틱 회귀 모델을 적용할 수 있다.

로지스틱 회귀(Logistic Regression)

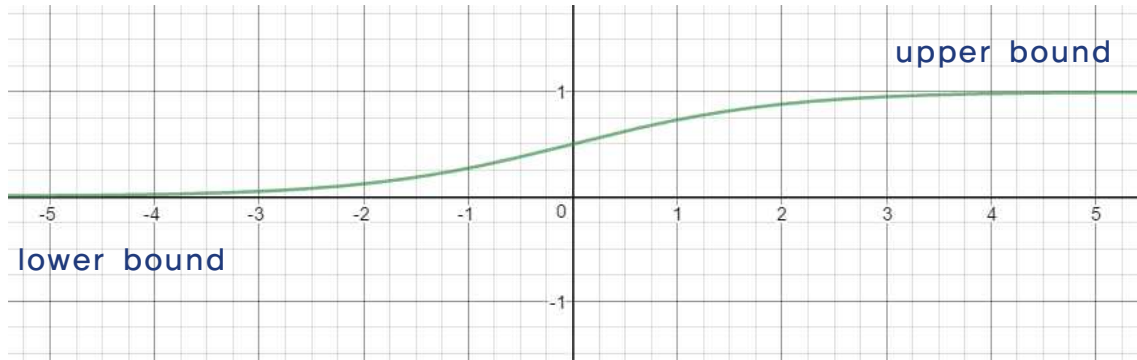
로지스틱 함수(Logistic function) = 시그모이드 함수(Sigmoid func.)

: x값으로 어떤 값이든 받을 수 있지만

출력 결과 y는 항상 0과 1 사이의 값이 된다.

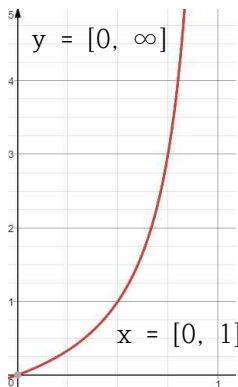
$$y = \frac{1}{1 + e^{-x}}$$

확률 밀도 함수(probability density function) 요건 충족



승산(Odds)

: 임의의 사건 A가 발생하지 않을 확률 대비 일어날 확률의 비율



$$odds = \frac{P(A)}{P(A^c)} = \frac{P(A)}{1 - P(A)}$$

P(A)가 1에 가까울수록 승산이 커지고,
반대로 P(A)가 0이라면 승산은 0이다.

이항 로지스틱 회귀

Y가 범주형일 경우, 다중선형회귀 모델 적용 불가

다중선형회귀모델: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n = \vec{\beta}^T \vec{x}$

라벨: 연속형 Y → 범주형 Y [0, 1]로 하기 위해

1. Y를 확률식으로 바꿔보면(좌변은 그대로) $P(Y=1|X=\vec{x}) = \vec{\beta}^T \vec{x}$

범위: 좌변 [0, 1], 우변 $[-\infty, \infty]$ 으로 좌변과 우변의 레벨이 맞지 않다

2. Y를 승산으로 바꿔보면(좌변은 그대로) $\frac{P(Y=1|X=\vec{x})}{1-P(Y=1|X=\vec{x})} = \vec{\beta}^T \vec{x}$

범위: 좌변 $[0, \infty]$, 우변 $[-\infty, \infty]$ 으로 좌변과 우변의 레벨이 맞지 않다

3. Y 승산에 로그를 취하면 $\log_e\left(\frac{P(Y=1|X=\vec{x})}{1-P(Y=1|X=\vec{x})}\right) = \vec{\beta}^T \vec{x}$

범위: 좌변 $[-\infty, \infty]$, 우변 $[-\infty, \infty]$ 으로 좌변과 우변의 레벨이 맞다

4. x가 주어졌을 때 범주 1일 확률 = $p(x)$, 우변 $\vec{\beta}^T \vec{x} = a$ 로 치환하면

$$\begin{aligned}\log_e\left(\frac{p(x)}{1-p(x)}\right) &= a \\ \frac{p(x)}{1-p(x)} &= e^a \\ p(x) &= e^a\{1-p(x)\} = e^a + e^a p(x) \\ p(x)(1+e^a) &= e^a \\ p(x) &= \frac{e^a}{1+e^a} = \frac{1}{1+e^{-a}}\end{aligned}$$

$$\therefore P(Y=1|X=\vec{x}) = \frac{1}{1+e^{-\vec{\beta}^T \vec{x}}} \Rightarrow \text{로지스틱 함수(시그모이드 함수)}$$

이항 로지스틱 회귀의 결정 경계

이항 로지스틱 모델에 입력 벡터 x를 넣으면 범주 1에 속할 확률을 반환한다

$$\text{판단 기준: } P(Y=1|X=\vec{x}) > P(Y=0|X=\vec{x})$$

범주가 2개 뿐이므로 $P(Y=1|X=\vec{x}) = p(x)$ 로 치환하면

$$\begin{aligned}p(x) &> 1-p(x) \\ \frac{p(x)}{1-p(x)} &> 1 \\ \log \frac{p(x)}{1-p(x)} &> 0\end{aligned}$$

$$\therefore \vec{\beta}^T \vec{x} > 0$$

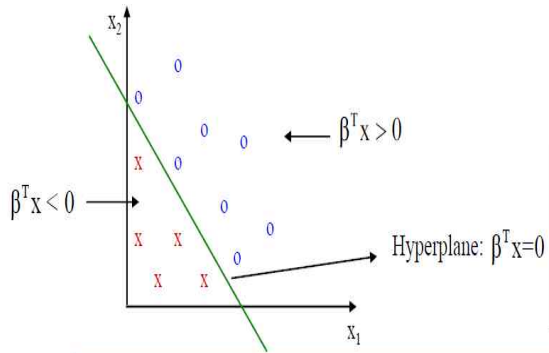
마찬가지로 $\vec{\beta}^T \vec{x} < 0$ 이면 데이터 범주를 0으로 분류한다.

로지스틱 결정 경계(decision boundary)

: $\vec{\beta}^T \vec{x} = 0$ 인 하이퍼플레인(hyperplane)

$$\vec{\beta}^T \vec{x} > 0 \Rightarrow 1, \vec{\beta}^T \vec{x} < 0 \Rightarrow 0$$

$$\vec{\beta}^T \vec{x} = 0 \Rightarrow \text{로지스틱 결정 경계} = \text{하이퍼프레인}$$



Classifier

$$y = \frac{1}{1 - e^{-\vec{\beta}^T \vec{x}}} \begin{pmatrix} y \rightarrow 1 & \text{if } \vec{\beta}^T \vec{x} \rightarrow \infty \\ y = \frac{1}{2} & \text{if } \vec{\beta}^T \vec{x} = 0 \\ y \rightarrow 0 & \text{if } \vec{\beta}^T \vec{x} \rightarrow -\infty \end{pmatrix}$$

다항 로지스틱 회귀

이항 로지스틱 회귀 모델을 이용해서 다항 로지스틱 회귀 문제 풀기

$$\log \frac{P(Y=1|X=\vec{x})}{P(Y=3|X=\vec{x})} = \beta_1^T \vec{x}$$

$$\log \frac{P(Y=2|X=\vec{x})}{P(Y=3|X=\vec{x})} = \beta_2^T \vec{x}$$

(범주 3에 속할 확률) = 1 - (범주 1에 속할 확률) - (범주 2에 속할 확률)

$$P(Y=1|X=\vec{x}) = \frac{e^{\beta_1^T \vec{x}}}{1 + e^{\beta_1^T \vec{x}} + e^{\beta_2^T \vec{x}}}$$

$$P(Y=2|X=\vec{x}) = \frac{e^{\beta_2^T \vec{x}}}{1 + e^{\beta_1^T \vec{x}} + e^{\beta_2^T \vec{x}}}$$

$$P(Y=3|X=\vec{x}) = \frac{1}{1 + e^{\beta_1^T \vec{x}} + e^{\beta_2^T \vec{x}}}$$

k개 범주를 분류하는 다항 로지스틱 회귀 모델의 입력 벡터 x가 각 클래스로 분류될 확률

$$P(Y=1|X=\vec{x}) = \frac{e^{\beta_k^T \vec{x}}}{1 + \sum_{i=1}^{K-1} e^{\beta_i^T \vec{x}}} \quad (k = 0, 1, \dots, K-1)$$

$$P(Y=K|X=\vec{x}) = \frac{1}{1 + \sum_{i=1}^{K-1} e^{\beta_i^T \vec{x}}}$$

소프트맥스(Softmax function)

「로그 승산」으로 된 좌변을 「로그확률」로 변경하기

$$\begin{aligned}
 P(Y=1|X=\vec{x}) &= \frac{e^{\vec{\beta}_1^T \vec{x}}}{1 + \sum_{i=1}^{K-1} e^{\vec{\beta}_i^T \vec{x}}} & Z &= 1 + e^{\vec{\beta}_1^T \vec{x}} + e^{\vec{\beta}_2^T \vec{x}} + \dots = 1 + \sum_{i=1}^{K-1} e^{\vec{\beta}_i^T \vec{x}} \\
 P(Y=2|X=\vec{x}) &= \frac{e^{\vec{\beta}_2^T \vec{x}}}{1 + \sum_{i=1}^{K-1} e^{\vec{\beta}_i^T \vec{x}}} & \Rightarrow \log P(Y=1|X=\vec{x}) &= \vec{\beta}_1^T \vec{x} - \log Z \\
 & & \log P(Y=2|X=\vec{x}) &= \vec{\beta}_2^T \vec{x} - \log Z \\
 & & \vdots & \\
 P(Y=K|X=\vec{x}) &= \frac{1}{1 + \sum_{i=1}^{K-1} e^{\vec{\beta}_i^T \vec{x}}} & \log P(Y=K|X=\vec{x}) &= \vec{\beta}_K^T \vec{x} - \log Z
 \end{aligned}$$

로그 성질을 이용해 범주 c에 속할 확률을 기준으로 식을 정리하기

$$\begin{aligned}
 \log P(Y=c) + \log Z &= \vec{\beta}_c^T \vec{x} \\
 \log \{P(Y=c) \times Z\} &= \vec{\beta}_c^T \vec{x} \\
 P(Y=c) \times Z &= e^{\vec{\beta}_c^T \vec{x}} \\
 P(Y=c) &= \frac{1}{Z} e^{\vec{\beta}_c^T \vec{x}}
 \end{aligned}
 \quad \left| \begin{array}{l} \text{plus. 전체 확률의 합} = 1 \\ 1 = \sum_{k=1}^K P(Y=k) = \sum_{k=1}^K \frac{1}{Z} e^{\vec{\beta}_k^T \vec{x}} = \frac{1}{Z} \sum_{k=1}^K e^{\vec{\beta}_k^T \vec{x}} \\ \therefore Z = \sum_{k=1}^K e^{\vec{\beta}_k^T \vec{x}} \end{array} \right.$$

$$\Rightarrow P(Y=c) = \frac{e^{\vec{\beta}_c^T \vec{x}}}{\sum_{k=1}^K e^{\vec{\beta}_k^T \vec{x}}} : \text{소프트맥스 함수}$$

정리

선형 회귀 → 이항 로지스틱 회귀 → 다항 로지스틱 회귀

└ 시그모이드 함수 └ 소프트맥스 함수

로지스틱 회귀 api

[sklearn.linear_model.LogisticRegression\(penalty='l2', C=1.0, random_state=None\)](#)

→ $C = \frac{1}{\lambda}$ 로 규제화의 정도를 조절: c값이 클수록 규제화 강도가 줄어든다

규제 강도에 따른 실험

1. L1, L2 규제화

penalty: {'l1', 'l2', 'elasticnet', 'none'}, default = 'l2'

2. $C = \frac{1}{\lambda}$ 로 규제화

→ 테스트 데이터의 라벨을 알 수 없을 경우,

학습 데이터의 일부를 검증 데이터(validation data)로 구성하여 테스트

규제 강도에 따른 초정계수(회귀계수) 실험

- 규제 강도 $\uparrow \rightarrow$ 추정된 계수들의 크기(절댓값) \downarrow

L1 규제화: 규제 강도 $\uparrow \rightarrow$ 계수에 0이 많아진다

계수에 대응하는 특성 변수(설명 변수 = x)를 제거하는 역할이다.

L2: β 거의 변화 없이 그대로

L1: $\beta \rightarrow 0, \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots$