

# 2022학년도 1학기 컴퓨터언어학

## 제4강 로지스틱 회귀분석 (2)

박수지

서울대학교 인문대학 언어학과

2022년 3월 16일 수요일

## 오늘의 목표

- 1 경사하강법을 적용하여 로지스틱 회귀분석의 모형 매개변수를 학습할 수 있다.
- 2 학습된 모형 매개변수를 해석할 수 있다.
- 3 부류가 세 가지 이상일 때 분류함수로 사용되는 소프트맥스 함수의 정의와 특징을 설명할 수 있다.

## 권장하는 학습 방법

- 복습을 합시다.
- 공부하는 혼자 하기보다 떼로 합시다.

## 통계적 분류기로서 로지스틱 회귀분석의 작동 과정

- 1 훈련 집합의 각 문서를 특성값들의 벡터  $\vec{x}$ 로 나타낸다.
- 2 분류기가  $\vec{x}$ 를 1로 분류할 확률  $\hat{y} = P(y = 1|\vec{x})$ 를  $\sigma(\vec{w} \cdot \vec{x} + b)$ 로 나타낸다.
- 3 확률 추정값  $\hat{y}$ 과 실제 정답  $y$  사이의 “거리”를 교차엔트로피 손실 함수로 정의한다.
- 4 교차엔트로피 손실 함수의 값을 최소로 만들기 위해 편도함수의 값이 0이 될 때의 모형 매개변수  $\vec{w}$ ,  $b$ 의 값을 계산한다.

## 남은 문제

- 1 텍스트를 어떻게 수치화된 벡터  $\vec{x}$ 로 나타내는가? — Feature Engineering
- 2 부류가 0, 1 이외에 세 개 이상 존재하는 경우 확률을 어떻게 추정하는가? — Multinomial logistic regression
- 3 교차엔트로피 손실 함수의 편도함수가 0이 되는 지점을 어떻게 찾는가? — Gradient Descent

# 오늘의 재료 (1)

## 데이터

### 주어진 데이터

텍스트 “컴퓨터언어학 재밌어요!!!”

정답  $y \in \{0, 1\}$

- 주로 사람의 판단(주석)을 따름
- 모형과 상관없이 정해져 있음

### 관측의 특성값 추출

입력  $\vec{x} = [x_1, x_2, \dots, x_f] \in \mathbb{R}^f$

- 모형에 따라 선택이 달라짐

## 오늘의 재료 (2)

### 로지스틱 회귀분석

#### 모형 매개변수

가중치  $\vec{w} = [w_1, w_2, \dots, w_f] \in \mathbb{R}^f$

편향  $b \in \mathbb{R}$

- 가중치의 개수는 특성값의 개수와 같음
- 분류함수에 따라 매개변수의 종류가 달라짐

#### 확률 추정

$$\text{추정치 } \hat{y} = P(y = 1 | \vec{x}) = \sigma(\vec{w} \cdot \vec{x} + b) = \frac{1}{1 + \exp(-[\vec{w} \cdot \vec{x} + b])}$$

# 오늘의 재료 (3)

교차엔트로피 손실함수

## 분류기의 손실 계산

$$\begin{aligned} L_{CE}(\hat{y}, y) &= -\log p(y|\vec{x}) \\ &= -[y \log \sigma(\vec{w} \cdot \vec{x} + b) + (1 - y) \log (1 - \sigma(\vec{w} \cdot \vec{x} + b))] \end{aligned}$$

- $\vec{x}$ 와  $y$ 는 훈련 집합의 데이터 및 특성값 추출로부터 주어지는 값임
- 모형 매개변수  $\vec{w}$ 와  $b$ 가 위 함수의 변수가 됨

## 손실의 최소화

방정식  $\frac{\partial}{\partial w_1} L_{CE} = 0, \dots, \frac{\partial}{\partial w_f} L_{CE} = 0, \frac{\partial}{\partial b} L_{CE} = 0$ 을 만족하는  $\vec{w}, b$ 의 값을 구하기

## 최적화의 전제

사실1 미분가능한 함수의 값이 최소가 될 때 접선의 기울기(=도함수의 값)는 0이다.

주의 접선의 기울기가 0이라고 해서 함수의 값이 최소가 된다고 보장할 수 있는가?

사실2 교차엔트로피 함수의 경우에는 보장할 수 있다. (아래로 볼록한 함수이기 때문에)

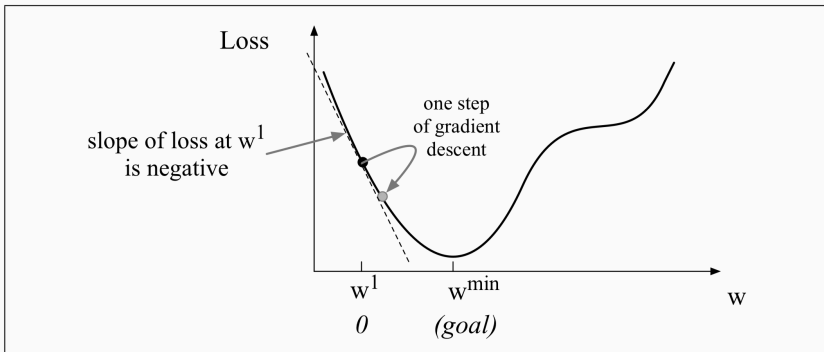
## $L_{CE}$ 의 편도함수

- $\frac{\partial}{\partial w_j} L_{CE}(\hat{y}, y) = (\hat{y} - y)x_j$
- $\frac{\partial}{\partial b} L_{CE}(\hat{y}, y) = (\hat{y} - y)$

증명: SLP3 Ch.5 부록

편도함수의 식은 간단하지만 언제 0이 되는지 알기는 어렵다.

## 경사 하강법



**Figure 5.4** The first step in iteratively finding the minimum of this loss function, by moving  $w$  in the reverse direction from the slope of the function. Since the slope is negative, we need to move  $w$  in a positive direction, to the right. Here superscripts are used for learning steps, so  $w^1$  means the initial value of  $w$  (which is 0),  $w^2$  at the second step, and so on.



## 경사 하강법

접선의 기울기(편도함수의 값)가 감소하는 방향으로  $\eta$ 만큼 움직인다.

$$w_j^{(t+1)} \leftarrow w_j^{(t)} - \eta \frac{\partial}{\partial w_j} L_{CE}$$

$t$  시간

$w_j^{(t)}$  현재 단계의  $w_j$  값

$w_j^{(t+1)}$  다음 단계의  $w_j$  값

$\eta$  움직이는 정도. 학습률 (learning rate).

로지스틱 회귀분석 모형의 매개변수를  $\theta = (w_1, w_2, \dots, w_f, b)$  라고 할 때

$$\theta^{(t+1)} \leftarrow \theta^{(t)} - \eta \nabla_{\theta} L_{CE}$$

$$\begin{bmatrix} w_1^{(t+1)} \\ w_2^{(t+1)} \\ \vdots \\ w_f^{(t+1)} \\ b^{(t+1)} \end{bmatrix} \leftarrow \begin{bmatrix} w_1^{(t)} \\ w_2^{(t)} \\ \vdots \\ w_f^{(t)} \\ b^{(t)} \end{bmatrix} - \eta \begin{bmatrix} (\hat{y} - y) x_1 \\ (\hat{y} - y) x_2 \\ \vdots \\ (\hat{y} - y) x_f \\ (\hat{y} - y) \end{bmatrix}$$

$$\text{단, } \hat{y} = \sigma(\vec{w}^{(t)} \cdot \vec{x} + b^{(t)}) = \frac{1}{1 + \exp(-(\vec{w}^{(t)} \cdot \vec{x} + b^{(t)}))} = \frac{1}{1 + \exp\left(-\left[w_1^{(t)} x_1 + w_2^{(t)} x_2 + \dots + w_f^{(t)} x_f + b^{(t)}\right]\right)}$$

# 확률적 경사 하강법 (SGD)

```

function STOCHASTIC GRADIENT DESCENT( $L()$ ,  $f()$ ,  $x$ ,  $y$ ) returns  $\theta$ 
  # where:  $L$  is the loss function
  #    $f$  is a function parameterized by  $\theta$ 
  #    $x$  is the set of training inputs  $x^{(1)}, x^{(2)}, \dots, x^{(m)}$ 
  #    $y$  is the set of training outputs (labels)  $y^{(1)}, y^{(2)}, \dots, y^{(m)}$ 

   $\theta \leftarrow 0$ 
  repeat til done # see caption
    For each training tuple  $(x^{(i)}, y^{(i)})$  (in random order)
      1. Optional (for reporting): # How are we doing on this tuple?
        Compute  $\hat{y}^{(i)} = f(x^{(i)}; \theta)$  # What is our estimated output  $\hat{y}$ ?
        Compute the loss  $L(\hat{y}^{(i)}, y^{(i)})$  # How far off is  $\hat{y}^{(i)}$  from the true output  $y^{(i)}$ ?
      2.  $g \leftarrow \nabla_{\theta} L(f(x^{(i)}; \theta), y^{(i)})$  # How should we move  $\theta$  to maximize loss?
      3.  $\theta \leftarrow \theta - \eta g$  # Go the other way instead
  return  $\theta$ 

```

**Figure 5.6** The stochastic gradient descent algorithm. Step 1 (computing the loss) is used mainly to report how well we are doing on the current tuple; we don't need to compute the loss in order to compute the gradient. The algorithm can terminate when it converges (or when the gradient norm  $< \epsilon$ ), or when progress halts (for example when the loss starts going up on a held-out set).

# 경사 하강법 적용 예시

## 설정

- 특성값  $\vec{x} = [x_1, x_2]$ 
  - $x_1$  긍정적인 단어 개수
  - $x_2$  부정적인 단어 개수
- 부류  $y$  인코딩
  - 1 긍정
  - 0 부정

## 훈련집합 (가상의 예시)

$\{([3, 2], 1), ([1, 4], 0), ([3, 0], 1), ([2, 3], 0)\}$

## 모형 매개변수 초기값

$w_1 = 0, w_2 = 0, b = 0$

## 모형 초매개변수

학습률  $\eta = 0.1$

## 훈련집합

$\{([3, 2], 1), ([1, 4], 0), ([3, 0], 1), ([2, 3], 0)\}$

	$w_1^{(t)}$	$w_2^{(t)}$	$b^{(t)}$
$t = 0$	0	0	0

$$\begin{bmatrix} w_1^{(1)} \\ w_2^{(1)} \\ b^{(1)} \end{bmatrix} \leftarrow \begin{bmatrix} w_1^{(0)} \\ w_2^{(0)} \\ b^{(0)} \end{bmatrix} - \eta (\hat{y} - y) \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} - 0.1 (0.5 - 1) \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.15 \\ 0.1 \\ 0.05 \end{bmatrix}$$

$$([x_1, x_2], y) = ([3, 2], 1)$$

$$\begin{aligned} \hat{y} &= \sigma(\vec{w}^{(0)} \cdot \vec{x} + b^{(0)}) \\ &= \sigma([0, 0] \cdot [3, 2] + 0) \\ &= \sigma(0) \\ &= 0.5 \end{aligned}$$

## 훈련집합

$\{([3, 2], 1), ([1, 4], 0), ([3, 0], 1), ([2, 3], 0)\}$

	$w_1^{(t)}$	$w_2^{(t)}$	$b^{(t)}$
$t = 1$	0.15	0.1	0.05

$$([x_1, x_2], y) = ([1, 4], 0)$$

$$\hat{y} = \sigma(\vec{w}^{(1)} \cdot \vec{x} + b^{(1)})$$

$$= \sigma([0.15, 0.1] \cdot [1, 4] + 0.05)$$

$$= \sigma(0.6)$$

$$\approx 0.6457$$

$$\begin{bmatrix} w_1^{(2)} \\ w_2^{(2)} \\ b^{(2)} \end{bmatrix} \leftarrow \begin{bmatrix} w_1^{(1)} \\ w_2^{(1)} \\ b^{(1)} \end{bmatrix} - \eta (\hat{y} - y) \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.15 \\ 0.1 \\ 0.05 \end{bmatrix} - 0.1 (0.6457 - 0) \begin{bmatrix} 1 \\ 4 \\ 1 \end{bmatrix} \approx \begin{bmatrix} 0.0854 \\ -0.1583 \\ -0.0146 \end{bmatrix}$$

## 훈련집합

$\{([3, 2], 1), ([1, 4], 0), ([3, 0], 1), ([2, 3], 0)\}$

	$w_1^{(t)}$	$w_2^{(t)}$	$b^{(t)}$
$t = 2$	0.0854	-0.1583	-0.0146

$$([x_1, x_2], y) = ([3, 0], 1)$$

$$\begin{aligned}\hat{y} &= \sigma(\vec{w}^{(2)} \cdot \vec{x} + b^{(2)}) \\ &= \sigma([0.0854, -0.1583] \cdot [3, 0] + 0.05) \\ &\approx \sigma(2.612) \\ &\approx 0.9316\end{aligned}$$

$$\begin{bmatrix} w_1^{(3)} \\ w_2^{(3)} \\ b^{(3)} \end{bmatrix} \leftarrow \begin{bmatrix} w_1^{(2)} \\ w_2^{(2)} \\ b^{(2)} \end{bmatrix} - \eta (\hat{y} - y) \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.0854 \\ -0.1583 \\ -0.0146 \end{bmatrix} - 0.1 (0.9316 - 1) \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix} \approx \begin{bmatrix} 0.1059 \\ -0.1583 \\ 0.0214 \end{bmatrix}$$

## 훈련집합

$\{([3, 2], 1), ([1, 4], 0), ([3, 0], 1), ([2, 3], 0)\}$

	$w_1^{(t)}$	$w_2^{(t)}$	$b^{(t)}$
$t = 3$	0.1059	-0.1583	0.0214

$$([x_1, x_2], y) = ([2, 3], 0)$$

$$\hat{y} = \sigma(\vec{w}^{(3)} \cdot \vec{x} + b^{(3)})$$

$$= \sigma([0.1059, -0.1583] \cdot [2, 3] + 0.0568)$$

$$\approx \sigma(-0.2063)$$

$$\approx 0.4486$$

$$\begin{bmatrix} w_1^{(4)} \\ w_2^{(4)} \\ b^{(4)} \end{bmatrix} \leftarrow \begin{bmatrix} w_1^{(3)} \\ w_2^{(3)} \\ b^{(3)} \end{bmatrix} - \eta (\hat{y} - y) \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.1059 \\ -0.1583 \\ 0.0214 \end{bmatrix} - 0.1 (0.4486 - 0) \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix} \approx \begin{bmatrix} 0.0162 \\ -0.2929 \\ -0.0235 \end{bmatrix}$$



## 학습된 모델 매개변수

$$\vec{w} = [0.0162, -0.2929], b = -0.0235$$

## 시험집합

$$\{\vec{x}\} = \{[2, 1]\}$$

## 예측

$\sigma(\vec{w} \cdot \vec{x} + b) = \sigma([0.0162, -0.2929] \cdot [2, 1] - 0.0235) = \sigma(-0.2843) < 0.5$ 이므로  
시험집합의 관측은 0(부정)으로 예측된다.



# 소프트맥스 함수

관측을  $K > 2$ 개 범주로 분류하기

## 정의

$\vec{z} = (z_1, z_2, \dots, z_K) \in \mathbb{R}^K$ 일 때

$$\text{softmax}(z_i) := \frac{\exp(z_i)}{\exp(z_1) + \exp(z_2) + \dots + \exp(z_K)}$$

## 특징

- 1 모든  $i \in \{1, 2, \dots, K\}$ 에 대하여  $0 < \text{softmax}(z_i) < 1$ 이 성립한다.
  - 2  $\text{softmax}(z_1) + \text{softmax}(z_2) + \dots + \text{softmax}(z_K) = 1$ 이 성립한다.
- ⇒ 확률의 속성을 만족한다.

## 소프트맥스 회귀분석과 로지스틱 회귀분석의 차이: 정답 표현 방법

로지스틱 회귀분석 정답이 하나의 스칼라  $y \in \{0, 1\}$ 로 표현된다.

소프트맥스 회귀분석 K개의 부류가 존재할 때 정답이 K차원 벡터  $\vec{y}$ 로 표현된다.

- 1번 부류가 정답인 경우  $\vec{y} = (1, 0, 0, \dots, 0, 0) \in \{0, 1\}^K$
- 2번 부류가 정답인 경우  $\vec{y} = (0, 1, 0, \dots, 0, 0) \in \{0, 1\}^K$
- K번 부류가 정답인 경우  $\vec{y} = (0, 0, 0, \dots, 0, 1) \in \{0, 1\}^K$

이러한 벡터 표현을 원-핫 인코딩(one-hot encoding)이라고 한다.

### 예시: 원-핫 인코딩

문장쌍을 Entailment/Contradiction/Neutral 중 하나로 분류하는 자연어 추론 ( $K = 3$ )

- 1 Entailment가 정답인 경우:  $\vec{y} = (1, 0, 0)$
- 2 Contradiction이 정답인 경우:  $\vec{y} = (0, 1, 0)$
- 3 Neutral이 정답인 경우:  $\vec{y} = (0, 0, 1)$

## 소프트맥스 회귀분석과 로지스틱 회귀분석의 차이: 모형 매개변수의 형상

로지스틱 회귀분석 관측을  $f$ 개의 특성값으로 표현할 때 ( $\vec{x} \in \mathbb{R}^f$ )  
가중치는  $f$ 차원 벡터  $\vec{w}$ , 편향은 스칼라(실수)  $b$ 가 된다.

$$\begin{aligned}\vec{x} &\mapsto z = \vec{w} \cdot \vec{x} + b \in \mathbb{R} \quad (\text{선형변환}) \\ &\mapsto \hat{y} = \sigma(z) \quad (\text{정답이 1일 확률 추정치})\end{aligned}$$

소프트맥스 회귀분석  $K$ 개의 부류가 존재하고  
관측을  $f$ 개의 특성값으로 표현할 때 ( $\vec{x} \in \mathbb{R}^f$ )  
가중치는  $[K \times f]$ 차원 행렬  $\mathbf{W}$ , 편향은  $K$ 차원 벡터  $\vec{b}$ 가 된다.

$$\begin{aligned}\vec{x} &\mapsto \vec{z} = \mathbf{W} \cdot \vec{x} + \vec{b} \in \mathbb{R}^K \quad (\text{선형변환}) \\ &\mapsto \hat{\mathbf{y}} = \text{softmax}(\vec{z}) \quad (\hat{y}_k: \text{정답이 } k \text{번 부류일 확률 추정치})\end{aligned}$$

## 로지스틱 함수와 소프트맥스 함수의 관계

$\vec{z} = (z_1, z_2) \in \mathbb{R}^2$  일 때

$$\begin{aligned}\text{softmax}(z_1) &= \frac{\exp(z_1)}{\exp(z_1) + \exp(z_2)} \\&= \frac{\exp(z_1) / \exp(z_1)}{[\exp(z_1) + \exp(z_2)] / \exp(z_1)} \\&= \frac{1}{1 + \exp(z_2) / \exp(z_1)} \\&= \frac{1}{1 + \exp(z_2 - z_1)} \\&= \frac{1}{1 + \exp(-(z_1 - z_2))} \\&= \sigma(z_1 - z_2)\end{aligned}$$

## 로지스틱 함수와 소프트맥스 함수의 관계

$\vec{z} = (z_1, z_2) \in \mathbb{R}^2$  일 때  $\text{softmax}(z_1) = \sigma(z_1 - z_2)$  이므로

- $z_1 > z_2$  이면  $z_1 - z_2 > 0$  이므로  $\sigma(z_1 - z_2) > 0.5$  이다.  $\Rightarrow$  1번 부류(1)로 예측한다.
- $z_1 < z_2$  이면  $z_1 - z_2 < 0$  이므로  $\sigma(z_1 - z_2) < 0.5$  이다.  $\Rightarrow$  2번 부류(0)로 예측한다.

$K = 2$  일 때 소프트맥스 함수를 사용하는 것은 사실 로지스틱 함수를 사용하는 것과 같다!

## 일반화: 신경망의 1개 “층”(layer)

입력  $\vec{x}$   $\mapsto$  선형변환  $z = \vec{w} \cdot \vec{x} + b$   $\mapsto$  분류함수 적용  $\hat{y} = \sigma(z)$   $\mapsto$  출력

Shallow learning 입력  $\rightarrow$  출력

Deep learning 입력  $\rightarrow$  은닉  $\rightarrow$  출력

다음 시간에 할 일

SLP Ch. 6 읽어 오기