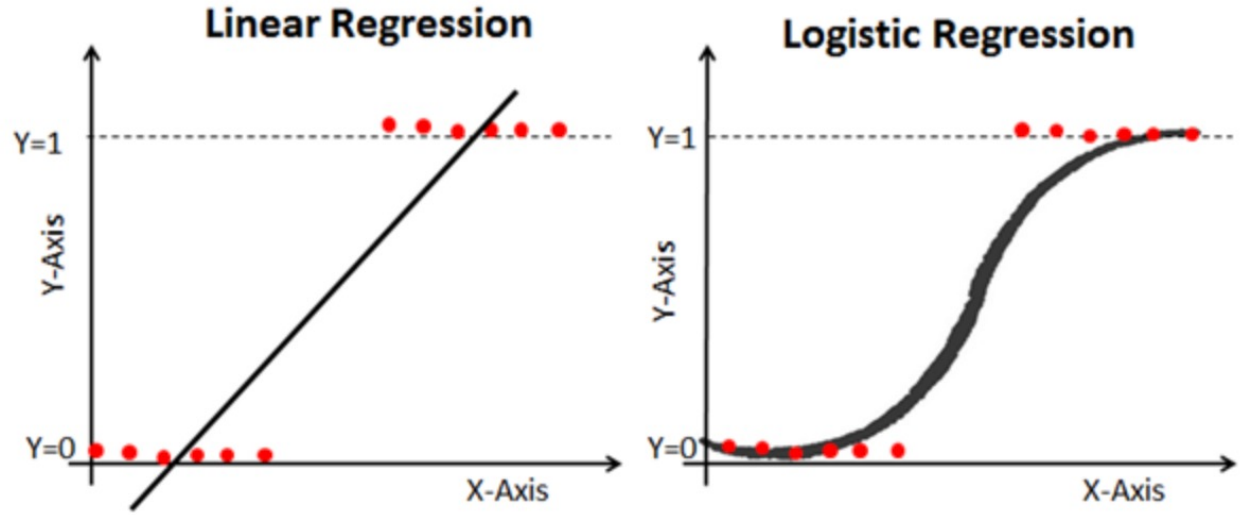
python™

로지스틱 회귀 (Logistic Regression)

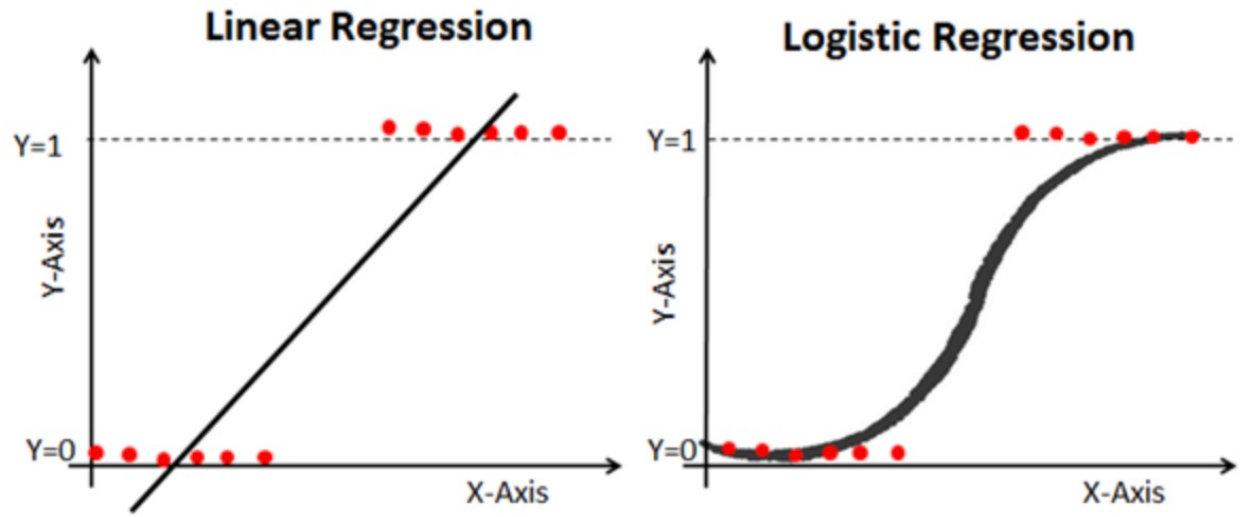


- Regression을 이용한 Classification 모델 (목적은 분류)
- 보통은 이항 로지스틱 회귀를 지칭 (결과값이 2개, 다항도 있기는 함)
- 독립변수는 연속형, 범주형 모두 가능하지만 종속 변수는 범주형 (0 or 1)
  종속변수의 예측 값은 0~1 사이의 값, 특정 값을 기준으로 0과 1로 종속변수가 결정

python™

로지스틱 회귀 (Logistic Regression)

- 선형회귀로 나타낼 경우 특정 값에 대해선 0 미만으로, 그리고 1을 초과하는 값을 가진다. 확률적으로 이는 말이 안되는 경우이기 때문에 우리는 로지스틱 회귀를 사용한다.



로지스틱 회귀 (Logistic Regression)

- HOW?
  1. 모든 속성(feature)들의 계수(coefficient)와 절편(intercept)을 0으로 초기화한다.
  2. 각 속성들의 값(value)에 계수를 곱하여 log-odds(=logit)를 확인한다.
  3. log-odds를 sigmoid 함수에 넣어서 [0,1] 범위의 확률을 구한다.

$$odds = \frac{P}{1-P} \quad logit = \log(odds) = \log\frac{P}{1-P}$$

$$sigmoid = \frac{1}{1+e^{-z}} \quad where \quad z = b_0 + b_1 x_1 + \cdots + b_p x_p$$

- 확률을 odds로 나타내는 이유는? 그리고 logit을 sigmoid 함수에 넣는 이유는?

  그냥 로그에 넣으면 x값이 0~1인 것만 나오겠지만
  odds로 바꿔서 넣어주면 음의무한대부터 양의무한대까지의 값을 얻을 수 있음.
  시그모이드에 넣는 이유는 음의무한대부터 양의무한대까지의 값을
  0부터 1사이의 값으로 만들어주기 위함임.

로지스틱 회귀 (Logistic Regression)

odds = 상대가 1번 할때 우리가 몇번 하냐

- odds : 사건이 발생할 확률 / 사건이 발생하지 않을 확률
  확률 값을 0부터 양의 무한대의 범위를 갖게 한다.
  P : [0,1] odds: [0, +∞]
  예) vs LG에 대한 두산의 승률 odds가 2 → LG가 1승 할 때 두산이 2승

- logit : odds로 나타낸 확률 값을 [-∞, +∞]의 범위를 갖게 한다.
  로지스틱 회귀분석에서는 **선형회귀식이** 들어간다!

- sigmoid : logit으로 나타낸 확률 값을 [0, +1]의 범위를 갖게 한다.

python™

로지스틱 회귀 (Logistic Regression)

- 로지스틱 회귀의 장점
  - 회귀계수의 해석이 가능하다 (통계적 설명이 가능하다)
  - class에 속할 확률을 추정한다
    (분류 자체보다 확률에 관심이 많을 경우 유용)
  - 낮은 차원에서 overfitting이 적음

b1이 증가할때 y가 얼마나 증가한다
b2가 증가할때 y가 얼마나 증가한다

- 로지스틱 회귀의 단점
  - Multiple 분류에 대해서는 다른 분류기에 밀리는 경향이 있다
  - outlier에 민감하다

나이브 베이즈 (Naive Bayes)

- Discrete Naïve Bayes
  Discrete : Feature가 모두 discrete 확률변수
  Naïve : Feature는 같은 class 내에서 조건부 독립
  Bayes : Bayes 정리를 이용하여 추정모형 단순화

- Gaussian Naïve Bayes
  Gaussian : Feature가 모두 정규분포를 따르는 확률변수
  Naïve : Feature는 같은 class 내에서 조건부 독립
  Bayes : Bayes 정리를 이용하여 추정모형 단순화

나이브 베이즈 (Naive Bayes)

- Bayes' Theorem을 이용한 Classification 모형

  - 베이즈 정리란?

$$\Pr(A|B) = \frac{\Pr(B|A)\Pr(A)}{\Pr(B)}$$

- 종속변수, 독립변수, 파라미터 모두 확률변수로 취급
- 다항분류도 가능. feature가 이산확률변수일 경우 성능이 좋음
- 훈련 시간이 극히 짧기 때문에 표본이 큰 데이터에 쓰기 좋음

  → 자연어처리 추천시스템 등 feature가 많고 표본이 큰 데이터에 많이 쓰임

나이브 베이즈 (Naive Bayes)

- 베이즈 정리란?

$$\Pr(A|B) = \frac{\Pr(B|A)\Pr(A)}{\Pr(B)}$$

사전확률 = P(A) & P(B), 조건부확률 = P(BIA), 사후확률 = P(AIB)
사전확률과 조건부확률을 통하여 사후확률을 구하는 정리

나이브 베이즈 (Naive Bayes)

Frequency Table		Buy	
		Yes	No
Discount	Yes	19	1
	No	5	5

Frequency Table		Buy	
		Yes	No
Free Delivery	Yes	21	2
	No	3	4

Frequency Table		Buy	
		Yes	No
Day	Weekday	9	2
	Workday	7	1
	Holiday	8	3

- 예측하고 싶은 것 : Pr(Buy | Holiday, Discount, Free Delivery)
- Buy → A // Discount, Free Delivery, Day → B

나이브 베이즈 (Naive Bayes)

Frequency Table		Buy		
		Yes	No	
Day	Weekday	9	2	11
	Workday	7	1	8
	Holiday	8	3	11
		24	6	30

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Workday	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	30/30

- $P(B) = P(\text{Weekday}) = \frac{11}{30} = 0.37$
- $P(A) = P(\text{No Buy}) = \frac{6}{30} = 0.2$
- $P(B|A) = P(\text{Weekday} \mid \text{No Buy}) = \frac{2}{6} = 0.33$

나이브 베이즈 (Naive Bayes)

Frequency Table		Buy		
		Yes	No	
Day	Weekday	9	2	11
	Workday	7	1	8
	Holiday	8	3	11
		24	6	30

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Workday	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	30/30

- 사전확률
  - $\mathrm{P(A)} = \mathrm{P(No\ Buy)} = \frac{6}{30} = 0.2$
  - $\mathrm{P(B)} = \mathrm{P(Weekday)} = \frac{11}{30} = 0.37$

나이브 베이즈 (Naive Bayes)

Frequency Table		Buy		
		Yes	No	
Day	Weekday	9	2	11
	Workday	7	1	8
	Holiday	8	3	11
		24	6	30

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Workday	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	30/30

- 조건부확률
  - $P(B|A) = P(\text{Weekday} \mid \text{No Buy}) = \frac{2}{6} = 0.33$

나이브 베이즈 (Naive Bayes)

Frequency Table		Buy		
		Yes	No	
Day	Weekday	9	2	11
	Workday	7	1	8
	Holiday	8	3	11
		24	6	30

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Workday	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	30/30

- 사후확률
  - $P(A|B) = P(\text{No Buy} \mid \text{Weekday}) = \frac{P(\text{Weekday} \mid \text{No Buy}) \times P(\text{No Buy})}{P(\text{Weekday})} = \frac{0.33 \times 0.2}{0.37} = 0.18$

→ Weekday일 때 물건을 구매하지 않을 확률 = 0.18

나이브 베이즈 (Naive Bayes)

Frequency Table		Buy		
		Yes	No	
Discount	Yes	19	1	20
	No	5	5	10
		24	6	30

Likelihood Table		Buy		
		Yes	No	
Discount	Yes	19/24	1/6	20/30
	No	5/24	5/6	10/30
		24/30	6/30	30/30

- 사전확률
  - $P(A) = P(Buy) = ?$
  - $P(B) = P(Discount) = ?$
- 조건부확률 & 사후확률
  - $P(B|A) = P(Discount \mid Buy) = ?$
  - $P(A|B) = P(Buy \mid Discount) = ?$

나이브 베이즈 (Naive Bayes)

Frequency Table		Buy		
		Yes	No	
Free Delivery	Yes	21	2	23
	No	3	4	7
		24	6	30

Likelihood Table		Buy		
		Yes	No	
Free Delivery	Yes	21/24	2/6	23/30
	No	3/24	4/6	7/30
		24/30	6/30	30/30

- 사전확률
  - $P(A) = P(\text{Buy}) = ?$
  - $P(B) = P(\text{Free Delivery}) = ?$
- 조건부확률 & 사후확률
  - $P(B|A) = P(\text{Free Delivery} \mid \text{Buy}) = ?$
  - $P(A|B) = P(\text{Buy} \mid \text{Free Delivery}) = ?$

나이브 베이즈 (Naive Bayes)

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Workday	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	30/30

Likelihood Table		Buy		
		Yes	No	
Discount	Yes	19/24	1/6	20/30
	No	5/24	5/6	10/30
		24/30	6/30	30/30

Likelihood Table		Buy		
		Yes	No	
Free Delivery	Yes	21/24	2/6	23/30
	No	3/24	4/6	7/30
		24/30	6/30	30/30

- A=Buy
  B=Holiday, Discount, Free Delivery
- A=No Buy
  B=Holiday, Discount, Free Delivery

나이브 베이즈 (Naive Bayes)

- $P(A|B) = P(\text{Buy} \mid \text{Holiday}, \text{Discount}, \text{Free Delivery})$

$$= \frac{P(\text{Holiday} \mid \text{Buy}) \times P(\text{Discount} \mid \text{Buy}) \times P(\text{Free Delivery} \mid \text{Buy}) \times P(\text{Buy})}{P(\text{Holiday}) \times P(\text{Discount}) \times P(\text{Free Delivery})}$$

$$= \frac{\frac{19}{24} \times \frac{21}{24} \times \frac{8}{24} \times \frac{24}{30}}{\frac{23}{30} \times \frac{20}{30} \times \frac{11}{30}}$$

$$= 0.986$$

- 구매 확률 : 0.986

나이브 베이즈 (Naive Bayes)

- $P(A|B) = P(\text{No Buy} \mid \text{Holiday, Discount, Free Delivery})$

$$= \frac{P(\text{Holiday} \mid \text{No Buy}) \times P(\text{Discount} \mid \text{No Buy}) \times P(\text{Free Delivery} \mid \text{No Buy}) \times P(\text{No Buy})}{P(\text{Holiday}) \times P(\text{Discount}) \times P(\text{Free Delivery})}$$

$$= \frac{\frac{2}{6} \times \frac{1}{6} \times \frac{3}{6} \times \frac{6}{30}}{\frac{23}{30} \times \frac{20}{30} \times \frac{11}{30}}$$

$$= 0.178$$

- 구매하지 않을 확률 : 0.178

나이브 베이즈 (Naive Bayes)

- 구매확률 = 0.986
  구매하지 않을 확률 = 0.178

어떻게 더해서 1이 안나올수가 있는걸까

- 확률의 합은 1이 될 수 없으므로 정규화

$$\frac{0.986}{0.986 + 0.178} = 84.71\% > \frac{0.178}{0.986 + 0.178} = 15.29\%$$

- 즉, 고객은 휴일에, 할인 및 무료배송으로 상품을 구매한다

나이브 베이즈 (Naive Bayes)

- 나이브 베이즈의 장점
  - 간단하고 구현하기 쉽다
  - 연속 데이터 / 이산 데이터를 모두 처리할 수 있다 (이산 데이터가 더 성능이 좋음)
  - 계산 시간이 매우 빠르다 (큰 데이터 셋에 적합)
  - 관련 없는 feature에 민감하지 않는다
  - 다중분류를 위해서도 사용할 수 있다

- 나이브 베이즈의 단점
  - feature 간의 독립성이 전제되어야 한다
    (다만 실제 데이터에서는 모든 feature가 독립인 경우는 드물다)

나이브 베이즈 (Naive Bayes)

- 나이브 베이즈가 많이 쓰이는 영역

  - 얼굴 인식
  - 날씨 예측
  - 의료 진단
  - 뉴스 분류 등...

python™