Background of Logistic Regression

Saehwa Kim

Information and Communications Engineering
Hankuk University of Foreign Studies

Distributions

Trials

Outcomes	A Single	N independent
Binary	Bernoulli	Binomial
K possible	Categorical	Multinomial

Distributions: parameters | example

Trials

Outcomes	**A Single**	***n* independent**
Binary	Bernoulli p (probability of success) Flipping a single coin	Binomial n (# of trials) and p Counting # of heads in n coin Flips
***k* possible**	Categorical $p1, p2, \ldots, pk$ (p for each category) Rolling a die (k=6)	Multinomial n and ($p1, p2, \ldots, pk$) Counting each face in n die rolls

Distributions: Problem Type

Outcomes \ Trials	A Single	*n* independent
Binary	Bernoulli (Binary Classification)	Binomial (Multiple Binary Classification)
k possible	Categorical (Multi-Class Classification)	Multinomial (Multiple Multi-Class Classification)

Probability Mass Function (PMF)

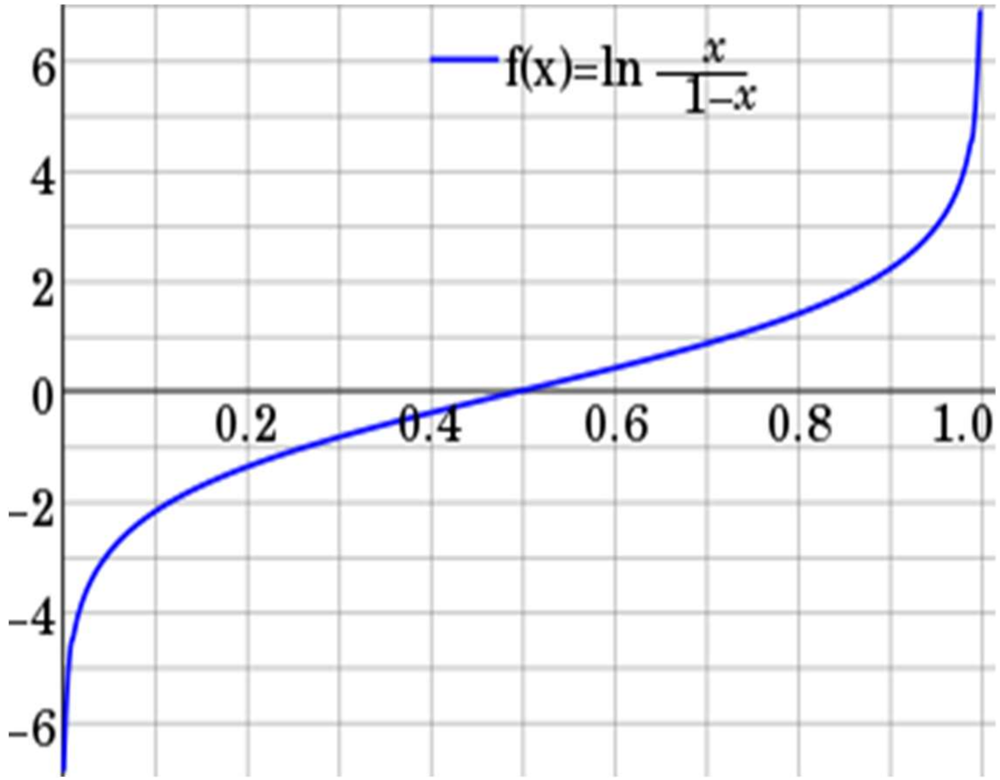
- 확률 질량 함수
- 이산 확률 변수에서 특정 값에 대한 확률
- 연속 확률 변수에서의 확률 밀도 함수와 대응
- Single Example Likelihood == Bernoulli probability mass function
  - When we have a single observation x, the likelihood is exactly the Bernoulli probability mass function
  - $L(p|x) = p^x * (1-p)^(1-x)$
- Multiple Independent Examples
  - For n independent observations, the likelihood is the product:
  - $L(p|x_1,...,x_n) = \prod(p^x_i * (1-p)^(1-x_i))$
- Log-Likelihood
  - $\log(L) = \sum(x_i \log(p) + (1-x_i)\log(1-p))$

Logit

- A function that maps probabilities [0, 1] to R [-inf, inf]

$$L = \ln \frac{p}{1-p} \qquad p = \frac{1}{1+e^{-L}}$$

- Inverse of the logistic function
- Probability 0.5 → Logit 0
- Probability < 0.5 → Negative logit
- Probability > 0.5 → Positive logit



Probability, Odds, Logit, Logistic, and Sigmoid 승산

Let p a probability.

$$\text{odds}(p) = \frac{p}{1-p}$$

$$\left(e^{\alpha} = \frac{p}{1-p}, \quad e^{\alpha} - p \cdot e^{\alpha} = p, \quad p = \frac{e^{\alpha}}{1+e^{\alpha}} \right)$$

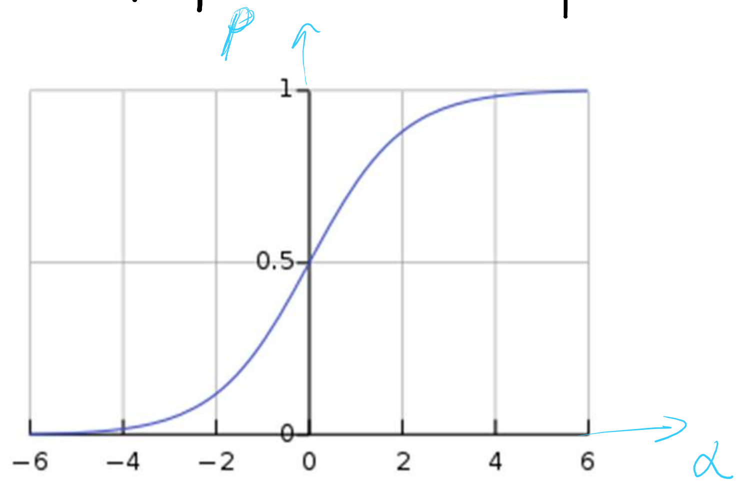
$$\alpha = \text{logit}(p) = \log(\text{odds}(p)) = \log\left(\frac{p}{1-p}\right) = -\log\left(\frac{1}{p} - 1\right)$$

$$\text{logistic}(\alpha) = \text{logit}^{-1}(\alpha)$$

-1 ← inverse

$$= \frac{e^{\alpha}}{1+e^{\alpha}} = \frac{1}{e^{-\alpha}+1}$$

$$= \text{sigmoid}(\alpha)$$



https://en.wikipedia.org/wiki/Sigmoid_function

Likelihood (가능도)

- A measure of how well a statistical model explains observed data
- A function of the model parameters, treating the observed data as fixed
- $L(\theta|Y) = P(Y|\theta)$
  - L: the likelihood function
  - θ (theta): model parameters
  - Y: observed data
  - $P(Y|\theta)$: the probability of seeing data Y given parameters θ
  - $P(Y|X)$로 표기할 경우에는 X가 θ임(weight값)
- Probability와의 비교
  - Probability: Fix parameters, look at different possible data
  - Likelihood: Fix data, look at different possible parameters
- Ex1: Coin Flipping (Binary)
  - Data: HTTHTH (H=heads, T=tails)
  - Parameter: p (probability of heads)
  - Likelihood: $L(p) = p(1\text{-}p)p(1\text{-}p)p(1\text{-}p) = p^3(1\text{-}p)^3$
- Ex 2: Die Rolling
  - Data: [2,6,6,3]
  - Parameters: $p_1,\ldots,p_6$ (probabilities for each face)
  - Likelihood: $L(p_1,\ldots,p_6) = p_2 p_6 p_6 p_3$

Likelihood (가능도)

- Not a probability (doesn't need to sum to 1)
- Can be very small with many data points
- Often work with log-likelihood (turns products into sums)
- Maximum Likelihood Estimation (MLE) finds parameters that maximize likelihood
- Training maximizes likelihood of seeing the training data
- Cross-entropy loss is derived from negative log-likelihood
- Different problems assume different probability distributions:
  - Binary classification: Bernoulli
  - Multi-class: Categorical
  - Regression: Often Gaussian

Probability, Information, Entropy, and Cross Entropy

- Let p a probability
- Information (정보량): 확률의 음의 로그: 단위는 bits (비트의 수)
  - $-\log_2 (p) = \log_2(1/p)$
  - 확률이 작을수록 정보량이 더 커짐
  - (예) 상금이 큰 로또 번호의 정보량
- Entropy (평균 정보량): 정보량의 평균
  - $E(-\log(p)) = - p^T * \log(p)$
- Cross entropy (평균 예측 정보량): 예측 정보량의 평균
  - 예측 정보량: 예측 확률 q의 음의 로그
  - 예측 확률 q의 음의 로그에 실제 확률 p를 곱해 적분/합
  - $E(-\log(q)) = - p^T * \log(q)$
- K-L divergence (Kullback–Leibler 쿨벡-라이블러 발산): relative entropy
  - Cross entropy – entropy
    - ⇔ Cross entropy = K-L divergence + Entropy
      - Loss로 cross entropy를 많이 쓰는데 이는 사실 K-L divergence를 최소화하고자 하는 것임 (entropy는 고정값이니까)
  - $E(-\log(q)) - E(-\log(p)) = E(-\log(q/p)) = - p^T * \log(q/p)$

More on Cross Entropy

- Cross Entropy == Negative Log Likelihood (NLL)
- Cross Entropy Minimization ==
  - NLL Minimization
  - Maximizing Likelihood
  - Maximum Likelihood Estimation (MLE)

- If we use one hot encoding
  - Entropy == 0
    - $- p^T * \log(p)$에서 p_i가 1이면 log(p)가 0, 나머지 p_i는 0
  - K-L divergence == Cross Entropy (- Entropy)
  - Cross Entropy == K-L divergence (+ Entropy)

Cross Entropy Example

- 매 주기 날씨 정보를 전송
- Weather information: 8 options (sunny, rainy, etc.)
- A Naive method: Using 3 bits
- Cross entropy measures the average number of bits you actually send per option.
- 예측 확률이 원 확률과 동일하면, KL-divergence = 0
  - cross entropy == entropy
- Cross entropy = entropy + KL-divergence
- 한 option의 확률이 압도적으로 클 경우(예: almost sunny):
  - sunny는 0으로 1 bit 사용
  - 나머지는 1xxx로 총 4 bits 사용
  - (예1) p(sunny) = 0.8 인 경우 평균 전송 비트 수: 0.8 * 1 + 0.2 * 4 = 1.6 bits
  - (예2) p(sunny) = 0.5 인 경우 평균 전송 비트 수: 0.5 * 1 + 0.5 * 4 = 2.5 bits

https://colah.github.io/posts/2015-09-Visual-Information/

출처: Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition, O'Reilly Media

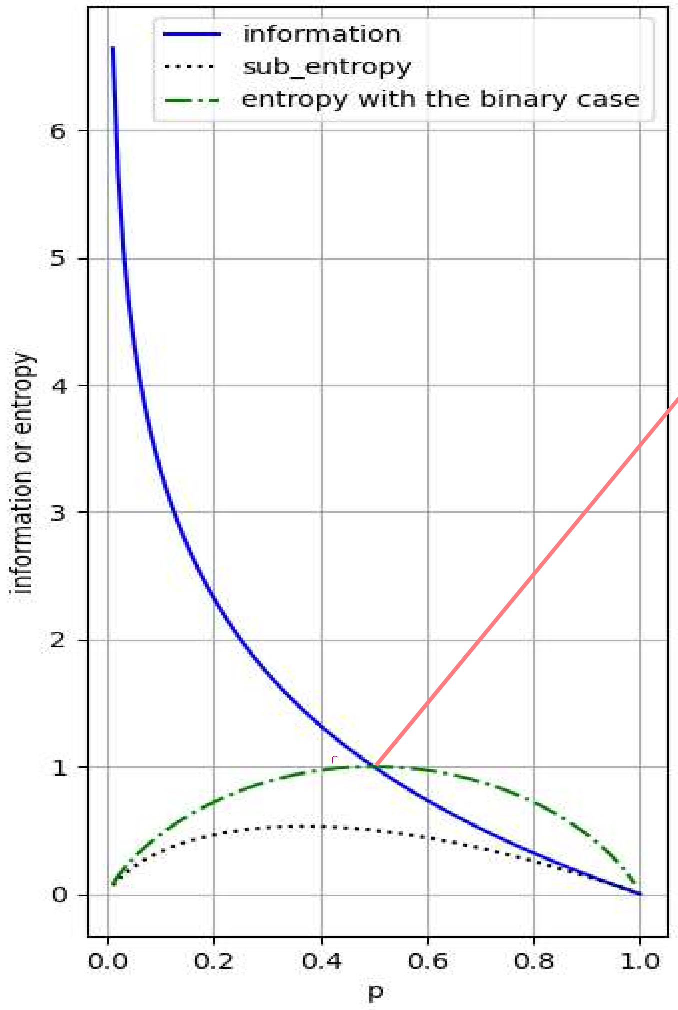
Information and Entropy

- Information: $-\log_2(p)$
- Entropy: $-p^T * \log(p)$
- 동전 던지기

동전 A	앞면	뒷면
확률	0.5	0.5
Inform.	1	1
Entropy	0.5 + 0.5 = 1	

동전 B	앞면	뒷면
확률	0.75	0.25
Inform.	0.42	2
Entropy	0.31 + 0.5 = 0.81	

동전 C	앞면	뒷면
확률	0.9	0.1
Inform.	0.15	3.32
Entropy	0.13 + 0.33 = 0.47	



Uniform distribution 일 때(random한 정도가 가장 큰 경우) entropy가 가장 큼 (random한 정도가 entropy)

(0.37, 0.53)

Entropy and Cross Entropy

- Cross entropy: $- p^T * \log_2(q)$
  - p가 실제 확률, q가 예측 확률
- 동전 던지기

동전 A	앞면	뒷면
확률	0.5	0.5
Inform.	1	1
Entropy	0.5 + 0.5 = 1	

동전 B	앞면	뒷면
확률	0.75	0.25
Inform.	0.42	2
Entropy	0.31 + 0.5 = 0.81	

동전 C	앞면	뒷면
확률	0.9	0.1
Inform.	0.15	3.32
Entropy	0.13 + 0.33 = 0.47	

동전 A	앞면	뒷면
예측 확률 1	0.75	0.25
예측 확률 2	0.9	0.1
Cross Entropy 1	[0.5, 0.5] * [0.42, 2] = 1.21	
Cross Entropy 2	[0.5, 0.5] * [0.15, 3.32] = 1.735	

동전 B	앞면	뒷면
예측 확률 1	0.5	0.5
예측 확률 2	0.9	0.1
Cross Entropy 1	[0.75, 0.25] * [1, 1] = 1	
Cross Entropy 2	[0.75, 0.25] * [0.15, 3.32] = 0.94	

동전 A	앞면	뒷면
예측 확률 1	0.5	0.5
예측 확률 2	0.75	0.25
Cross Entropy 1	[0.9, 0.1] * [1, 1] = 1	
Cross Entropy 2	[0.9, 0.1] * [0.42, 2] = 0.58	

Probability, Odds, Logit, Logistic, Information, Entropy, and Cross Entropy

- Probability 1/5 → Odds 1/4
- Probability 3/5 → Odds 3/2
- Probability 1/e^8 → Information 8
- Probability e / (e + 1) → Logit 1 (Odds e)
- Probability e^3 / (e^3 + 1) → Logit 3 (Odds e^3)
- 승산이 1 이상 되기 위한 최소 확률: 1/2

Softmax Regression

- Generalization of the logistic function to multiple dimensions
- Also called multinomial logistic regression

The standard (unit) softmax function $\sigma : \mathbb{R}^K \to \mathbb{R}^K$ is defined by the formula

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \text{ for } i = 1, \dots, K \text{ and } \mathbf{z} = (z_1, \dots, z_K) \in \mathbb{R}^K$$

https://en.wikipedia.org/wiki/Softmax_function

- we apply the standard exponential function to each element z_i of the input vector z
- and normalize these values
  - by dividing by the sum of all these exponentials
  - this normalization ensures that the sum of the components of the output vector is 1.