Linear / Logistic regression

index

Linear Regression

Logistic Regression

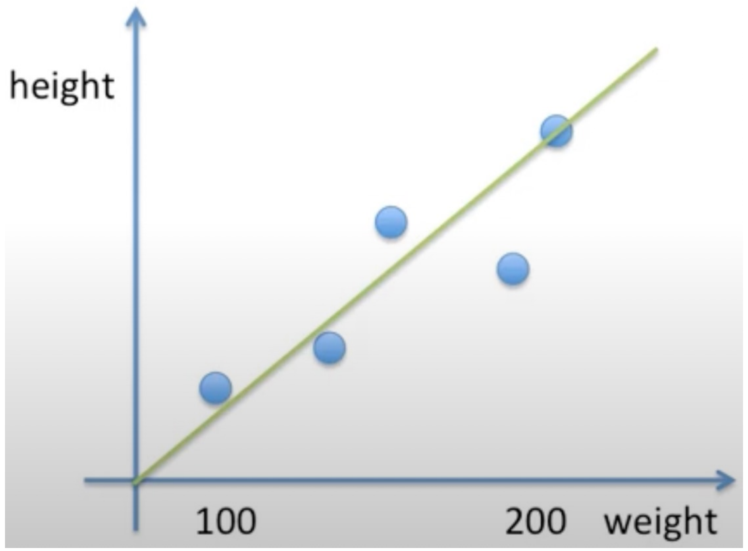
Entropy

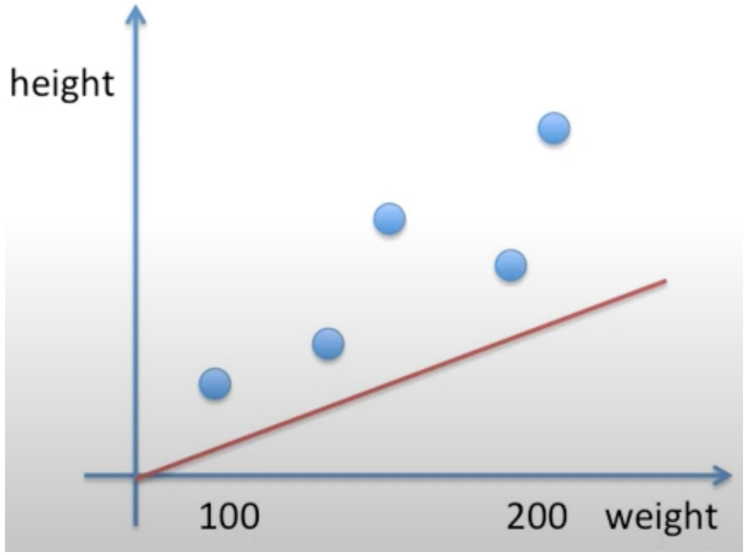
Linear Regression

선형회귀(Linear regression)는 종속 변수 y와 하나 이상의 독립 변수 x와의 선형 상관관계를 모델링하는 기법이다.

1)단순 선형 회귀(Simple Linear Regression)
y=Wx+b 의 식으로 나타내어지며, W을 가중치(weight) 상수항에 해당하는 b를 편향(bias)이라 한다. 그래프의 형태는 아래 그림과 같이 직선으로 나타나진다.

2)다중 선형 회귀(Multiple Linear Regression)
y=W1x1+W2x2+…+Wnxn+b 으로 나타나며 여러 독립변수에 의해 영향을 받는다. 그래프의 형태는 평면이다.





Cost function J(θ)

이상적인 선형회귀는 실제 데이터와의 오차가 가장 작아야 한다.

오차를 계산하기 위해서는 MSE, MAE, SSE등 여러 방식이 존재한다.

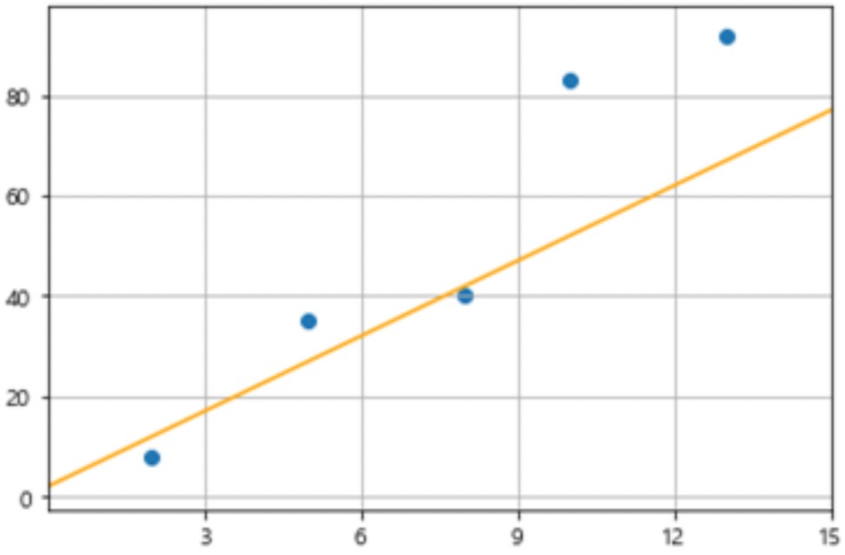
보통은 MSE를 많이 사용한다.

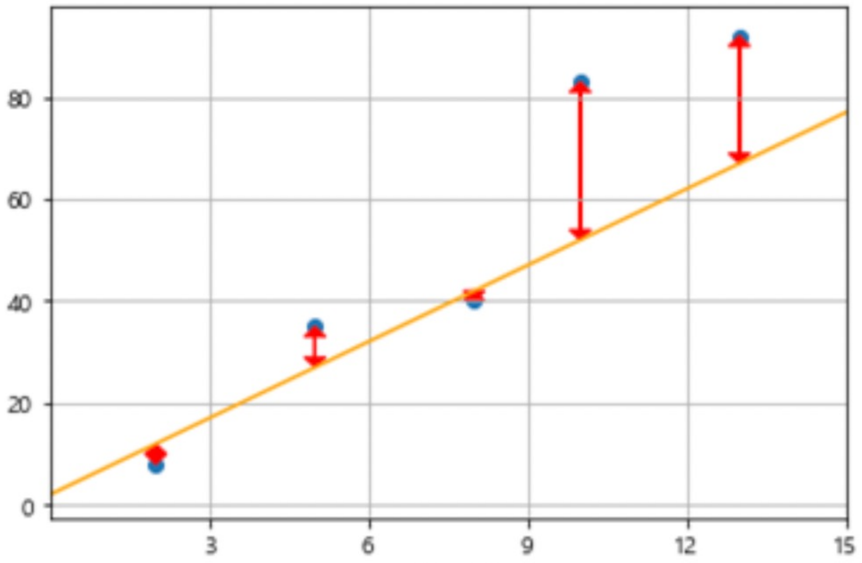
MSE를 이용하면 제곱을 이용하기에 큰 오차가 발생했을 때 더 큰 페널티를 부여하는 장점이 존재한다.

$$MSE = \frac{1}{n}\sum(y_i - \hat{y})^2$$

$$MAE = \frac{1}{n}\sum|y_i - \hat{y}|$$

$$SSE = \sum(y_i - \hat{y})^2$$





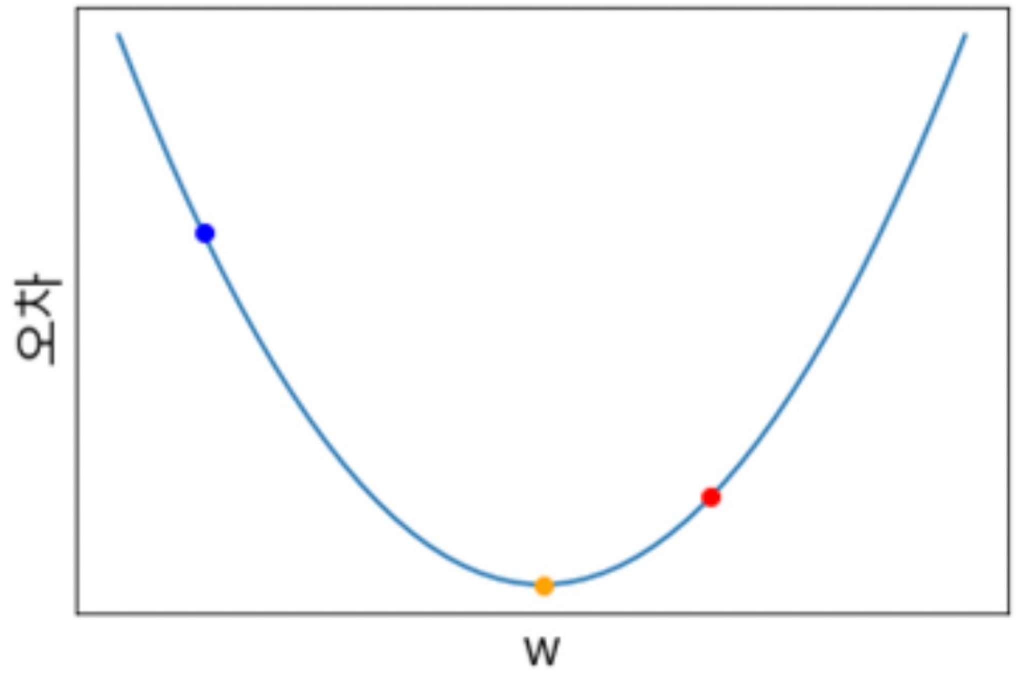
Optimizer: Gradient Descent(경사하강법)

Optimizer : cost function을 최소화 하는 w와 b를 구하는 것.

Gradient Descent : 비용함수의 기울기가 작아지는 방향으로 w와 b를 갱신하는 방식

오차는 거리의 제곱으로 나타내지기에 MSE(mean square error)의 그래프는 다음과 같이 이차함수 형태로 나타난다.

아래 그림의 노란색 점에 위치할 때 오차의 값이 최소값이 되므로 최적화 모델이 된다.



Gradient decent

파라미터를 임의로 정한 다음에 조금씩 변화시켜가며 손실을 점점 줄여가는 방법으로 최적의 파라미터를 찾아간다

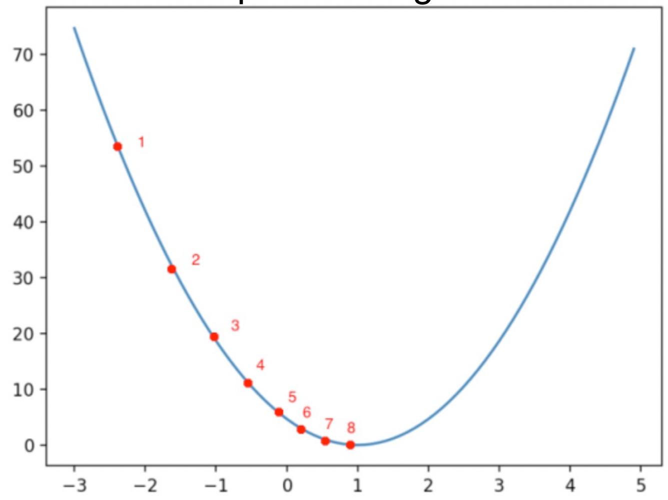
- 기울기가 음수일때는 양의 방향으로, 기울기가 양수일때는 음의 방향으로 이동한다.

$$w := w - \alpha \frac{\partial}{\partial w} cost(w, b)$$

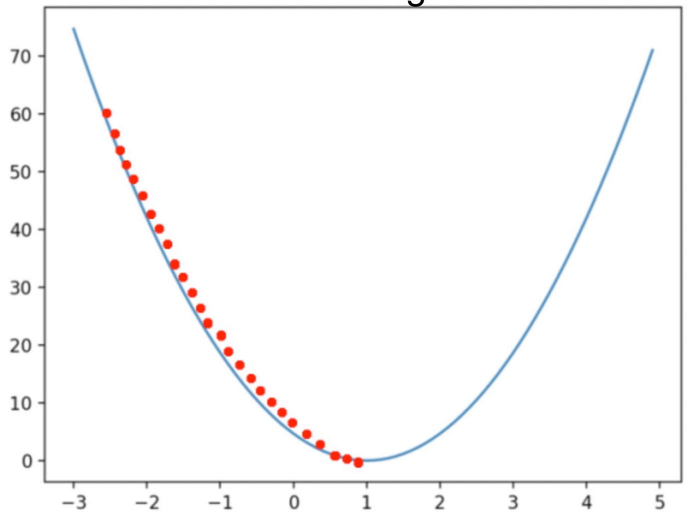
$\frac{\partial}{\partial w} cost(w, b)$는 접선의 기울기이다.

α 는 학습률(**learning rate**)이다. α 를 이용하여 $\boldsymbol{w}$의 값을 갱신한다.

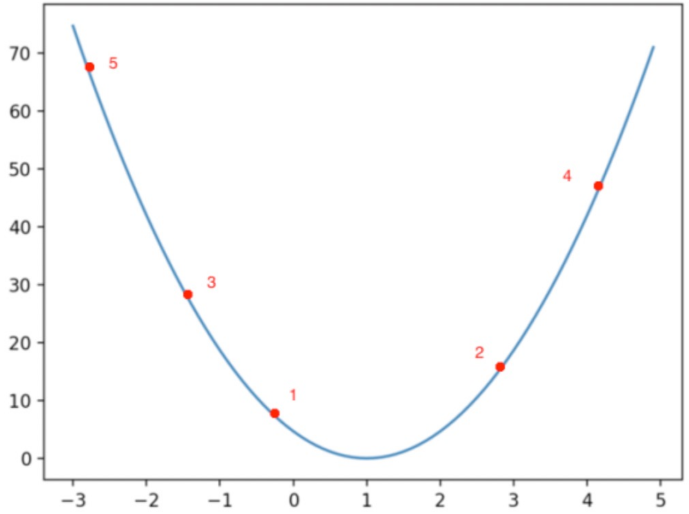
Proper learning rate



small learning rate



big learning rate



Ordinary Least Square(최소자승법)

OLS는 RSS(Residual Sum of Square)을 최소화 하는 가중치 벡터를 행렬 미분으로 구하는 방법인데, 이는 SSE(Sum of Squared Errors)와 동일하다.

Y=Wx + b는 다음과 같이 벡터로 표시할 수 있다.

$$\hat{Y} = X\theta \;\; (\text{단}, \hat{Y} = \begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \hat{y}_3 \\ \hat{y}_4 \\ \hat{y}_5 \end{pmatrix}, X = \begin{pmatrix} 1x_1 \\ 1\; x_2 \\ 1\; x_3 \\ 1\; x_4 \\ 1\; x_5 \end{pmatrix}, \theta = \begin{pmatrix} b \\ w \end{pmatrix})$$

이때 실제값과 예측값의 오차 벡터 e는 다음과 같다.

$$e = Y - \hat{Y} = Y - X\theta$$

이를 통하여 SSE를 구하고 이를 미분하여 비용함수의 기울기의 최솟값을 구할 수 있다,.

$$\begin{aligned} SSE &= e^T e \\ &= (Y - X\theta)^T (Y - X\theta) \\ &= Y^T Y - \theta^T X^T Y - Y^T X\theta + \theta^T X^T X\theta \\ &= Y^T Y - 2Y^T X\theta^T + \theta^T X^T X\theta \end{aligned}$$

$$\begin{aligned} \nabla\theta(Y^T Y - 2Y^T X\theta + \theta^T X^T X\theta) &= 0 \\ &= 2X^T X\theta - 2X^T Y \end{aligned}$$

Logistic Regression

독립변수 x의 선형결합을 통해 사건의 발생가능성을 예측하는 기법

- Linear Regression과 달리 Trend를 찾는 것이 아닌 사건의 발생가능성 즉, 확률을 도출함
- 연결함수(link function)을 통해 선형 방정식의 값을 [0, 1]구간의 확률로 mapping

분류모델로의 활용

- 특정 집단(class)에 속하는 사건을 고려

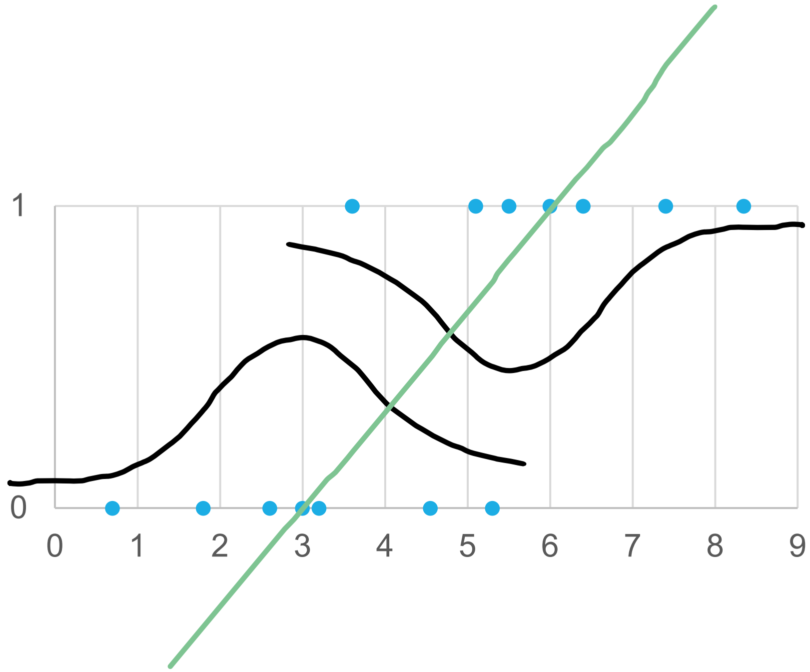
Logistic Regression Classification

아래와 같은 범주형 데이터를 고려

- Class 0(μ=3), Class 1(μ=6)
- Feature의 선형결합을 통해 Class 1에 속할 확률을 구한다

$$P(Y=1|X=\vec{x}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 = \vec{\beta}^T \cdot \vec{x}$$

- 연결함수 g(X)를 통해 우변의 구간 (-∞, +∞)을 [0,1]으로 mapping 해야 한다



Logistic Regression Classification (Cont'd)

사건의 확률이 아니라 승산(odds)를 구하는 문제로 바꿔본다면

$\frac{P(Y=1|X=\vec{\mathrm{x}})}{1-P(Y=1|X=\vec{\mathrm{x}})} = \vec{\beta}^{\mathrm{T}} \cdot \vec{\mathrm{x}}$ ← 여전히 좌항[0, ∞)과 우항(-∞, ∞)의 범위가 맞지 않음

좌변에 로그를 취하면

$\log \frac{P(Y=1|X=\vec{\mathrm{x}})}{1-P(Y=1|X=\vec{\mathrm{x}})} = \vec{\beta}^{\mathrm{T}} \cdot \vec{\mathrm{x}}$ ← 양변의 범위가 (-∞, ∞)으로 일치한다

Logistic Regression Classification (Cont'd)

원래의 문제로 돌려놓자

- **주어진 $\vec{x}$가 Y label값 1에 속할 확률을 구하는 문제**

Notation

- 로지스틱 함수(logistic function) $= \frac{\mathbf{1}}{(\mathbf{1}+e^{-\vec{\beta}^T\cdot\vec{x}})}$
- 로짓(logit) $= \log\frac{\mathrm{P}}{1-\mathrm{P}}$
- 승산(odds) $= \frac{\mathrm{P}}{1-\mathrm{P}}$

$$\log\frac{P(Y=1|X=\vec{x})}{1-P(Y=1|X=\vec{x})} = \vec{\beta}^T\cdot\vec{x}$$

$$\rightarrow \quad \frac{P(Y=1|X=\vec{x})}{1-P(Y=1|X=\vec{x})} = e^{\vec{\beta}^T\cdot\vec{x}}$$

$$\rightarrow \quad P(Y=1|X=\vec{x}) = e^{\vec{\beta}^T\cdot\vec{x}} - P(Y=1|X=\vec{x})e^{\vec{\beta}^T\cdot\vec{x}}$$

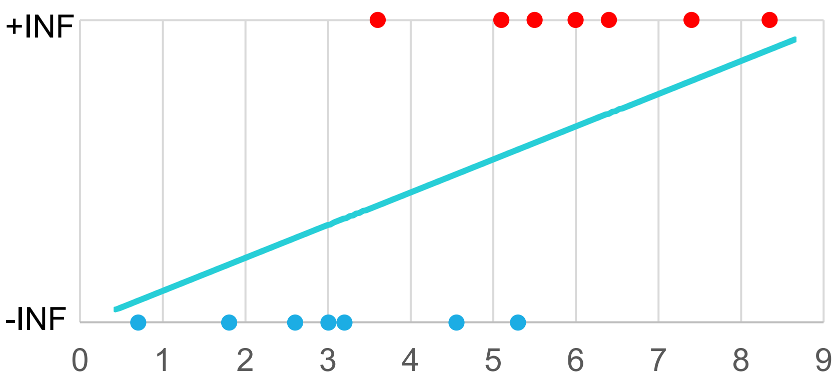
$$\rightarrow \quad P(Y=1|X=\vec{x})(1-e^{\vec{\beta}^T\cdot\vec{x}}) = e^{\vec{\beta}^T\cdot\vec{x}}$$

$$\therefore P(Y=1|X=\vec{x}) = \frac{e^{\vec{\beta}^T\cdot\vec{x}}}{(1-e^{\vec{\beta}^T\cdot\vec{x}})} = \frac{1}{(1+e^{-\vec{\beta}^T\cdot\vec{x}})}$$

Logistic Regression Model Fitting

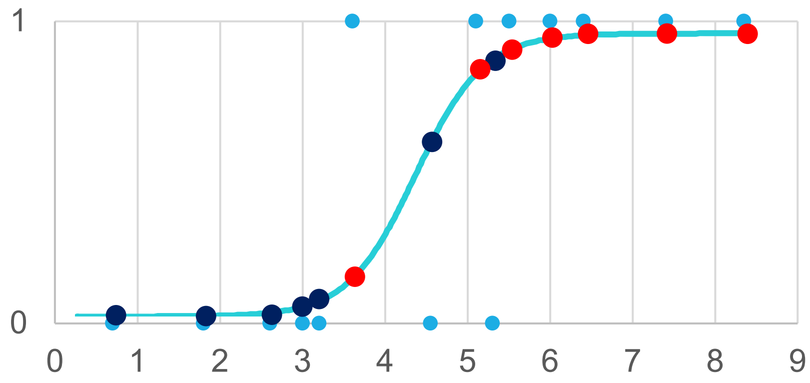
Y축이 logit인 linear regression 문제를 풀면 된다?

- MSE가 ∞
- 따라서, 각 점을 trend-line에 사영(project)하여 S-curve로 가져간다



이제, 최대우도법(maximum likelihood method)를 적용한다

- Training dataset이 independent하다면,
- $\mathcal{L}(\theta) = \prod_{\mathrm{i}} \mathrm{P}(Y = 1 | X = \vec{\mathrm{x}}_{\mathrm{i}}) \quad \rightarrow \quad \mathcal{L}^{*}(\theta) = \sum_{\mathrm{i}} \log \mathrm{P}(Y = 1 | X = \vec{\mathrm{x}}_{\mathrm{i}})$[1]
- $\ell(\theta) = -\frac{1}{\mathrm{m}} \sum_{\mathrm{i}}^{\mathrm{m}} [\mathrm{y}_{\mathrm{i}} \log \hat{\mathrm{p}}_{\mathrm{i}} + (\mathrm{y}_{\mathrm{i}} - 1) \log(1 - \hat{\mathrm{p}}_{\mathrm{i}})]$[2]



[1] conditional log-likelihood
[2] logistic regression 손실함수

Entropy

정보 전달의 기대되는 정보량(=최소 정보량)

- 특정 사건이 일어날 확률의 기댓값 과 반비례

정보량 : 일어날 확률 p(x)

- 이산변수: –log2
- 연속변수: –ln

정보 엔트로피 : 정보량의 기댓값

$$H(X) = E[I(X)] = -\sum_{i=1}^{n} P(x_i)\log_b\big(P(x_i)\big)$$

$$I(x) = -\log_b\big(P(X)\big)$$

Entropy - Example

1) 동전의 앞, 뒷면 이 나올 확률 각 [0.5, 0.5]

$$0.5 \cdot (-\log_2 0.5) + 0.5 \cdot (-\log_2 0.5) = 1$$

2) 동전의 앞, 뒷면 이 나올 확률 각 [0.3, 0.7]

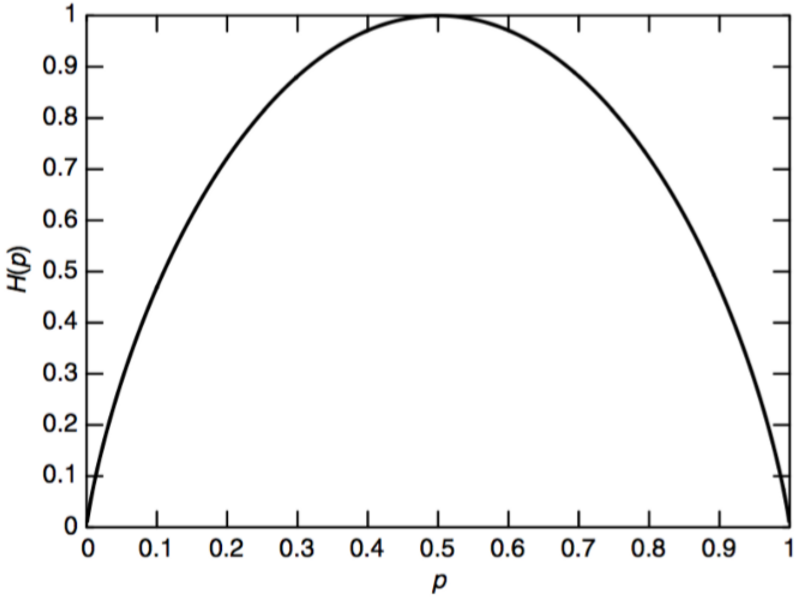
$$0.3 \cdot (-\log_2 0.3) + 0.7 \cdot (-\log_2 0.7) = 0.5211 + 0.3605 = 0.8816$$

3) 동전의 앞, 뒷면 이 나올 확률 각 [1, 0]

$$1 \cdot (-\log_2 1) + 0 \cdot (-\log_2 0) = 0$$

- 항상 앞면이 나오는 확률의 경우 필요한 정보량이 0

$$H(X) = -p \log p - (1-p)\log(1-p) \stackrel{\text{def}}{=} H(p)$$



uniform한 확률 변수를 가질 때 정보량이 최대

즉, 확률이 높은(자주 일어날 수 있는) 사건은 정보량 적음
엔트로피 (= 불확실성)

Cross-entropy

cross-entropy : 예측확률로 불확실성 추측

딥러닝 예측 시 정답과 얼마나 근사한지

Cross-entropy loss : classification 모델이 학습이 잘 되었는지 측정 가능

MSE (직관적 loss의 거리) 와 달리 데이터 분포를 학습하기 유용함

MLE 학습으로 데이터 분포를 학습

p : prediction prob

y : 실제 지표

$$-(y\log(p) + (1-y)\log(1-p))$$

실제 지표가 [1 , 0]일때 예측값은 [0.6 , 0.4]

Entropy : -1 x log2(1) – 0 x log2(0) = 0

Cross-entropy : -1 x log2(0.6) – 0 x log2(0.4) = 0.74

Entropy와 달리 정답을 추측