Chap 4. Logistic Regression

Binary Classification

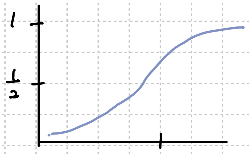
Bayes' rule을 이용함

$$P(C_1|x) = \frac{P(x/C_1)P(C_1)}{P(x)} = \frac{P(x/C_1)P(C_1)}{P(x/C_1)P(C_1) + P(x/C_2)P(C_2)}$$

logistic function

$$P(C_1/x) = \frac{1}{1+e^{-\varepsilon}}$$

$$\varepsilon = \underbrace{\log \frac{P(x/C_1)}{P(x/C_2)}}_{\text{likelihood ratio}} + \underbrace{\log \frac{P(C_1)}{P(C_2)}}_{\text{Prior ratio}}$$



Maximum likelihood Function

Logistic Regression : MLE

① Modelling

$$P(Y/X,w) = \prod_{n=1}^{N} P(y_n=1/x_n)^{y_n}(1-P(y_n=1/x_n))^{1-y_n}$$
$$= \prod_{n=1}^{N} \sigma(w^T x_n)^{y_n}(1-\sigma(w^T x_n))^{1-y_n}$$

② w maximize

log를 취함 $L = \sum_{n=1}^{N} \log P(y_n/x_n) = \sum_{n=1}^{N}(y_n \log \hat{y}_n + (1-y_n)\log(1-\hat{y}_n))$

③ optimization

IRLS(iterative re-weighted least square) by Newton's method.

Newton's Method

- gradient

$f(x): R^D \to R$ (즉, $x \in \mathbb{R}^D$)

$\nabla f(x) = [\frac{\partial f}{\partial x_1} \cdots \frac{\partial f}{\partial x_D}]^T$ (즉 $\nabla f(x) \in \mathbb{R}^D$)

- Hessian Maxtrix

x가 Scalar 일때 $\nabla^2 f(x)$ $\begin{cases} x>0 & \text{최소점} \\ x<0 & \text{최대점} \end{cases}$

$x \in \mathbb{R}^D$

H는 positive definite

if $z^T H z > 0$ $\forall z \neq 0$ $\Leftrightarrow$ All eigen value of H positive.

- Gradient $\underbrace{\text{descent}}_{\text{minimize}}$ / $\underbrace{\text{Ascent}}_{\text{maximize.}}$ 전제 Learning rate(step size $\eta > 0$).

→ simple iterative method

- Newton's Method.

$$w^{(k+1)} = w^{(k)} - [\underbrace{\nabla^2 J(w^{(k)})}_{\text{positive definite.}}]^{-1} \nabla J(w^{(k)})$$

convex . non convex .. ?

Logistic Regression : IRLS Algorithm

- $J(w)$ as negative log-likelihood

$$J(w) = -L(w) = -\sum_{n=1}^{N}[y_n \log \hat{y}_n + (1-y_n)\log(1-\hat{y}_n)]$$

gradient

$$\nabla J(w) = -\sum_{n=1}^{N}(y_n - \hat{y}_n)x_n$$

Hessian

$$\nabla^2 J(w) = \sum_{n=1}^{N} \hat{y}_n(1-\hat{y}_n)x_n x_n^T$$

- Logistic Regression : IRLS

$\Delta w =$ inverse of Hessian + gradient
$[\nabla^2 J(w)]^{-1}$ $\nabla J(w)$

$$w \leftarrow w + (XSX^T)^{-1}XSb$$

Softmax Regression

공식) $P(y=k/x) = \dfrac{e^{w_k^T x}}{e^{w_1^T x} + \cdots e^{w_k^T x}}$

Cross Entropy Error

- Information

$I = \log_2 \frac{1}{P(x)}$

Entropy = Average Information

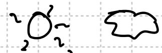
$= \mathbb{E}[-\log P(x)]$

Cross-entropy

$H(\underbrace{p}_{\text{실제값}}, \underbrace{q}_{\text{예측값.}})$

$= \mathbb{E}_p[-\log_2 q]$

예시)



75% ($\frac{3}{4}$) 25% ($\frac{1}{4}$)

$I(\text{sunny}) = \log_2 \frac{4}{3}$

$I(\text{rainny}) = \log_2 4$

$\text{Entropy} = \frac{3}{4}\log_2 \frac{4}{3} + \frac{1}{4}\log_2 4$

$\text{Cross-Entropy} = 1 + \log \frac{4}{3} + \log \sim$

- Cross Entropy Error

= negative conditional Bernoulli likelihood

$$J = -\sum_{n=1}^{N} y_n \log \hat{y}_n - \sum_{n=1}^{N}(1-y_n)\log(1-\hat{y}_n)$$

- Linear Regression과 Logistic Regression 차이

Linear Regression : $y \in \mathbb{R}$
Logistic Regression $y \in \{0, 1\}$

Multi-Label learning

여러레이블로 예측

label dependency

x_1 $y_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$

x_2 $y_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$

Target Matrix Y

$$\begin{array}{cc} \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ & & \vdots & & \end{bmatrix} & \begin{array}{l} \text{Cat} \\ \text{Dog} \\ \vdots \end{array} \end{array}$$