Classification : Regression Approaches

(x_i, y_i), $x_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}^k$ (k-class classification), $i = 1, 2, \ldots N$

$$x = \underbrace{\begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(d) \end{bmatrix}}_{d}, \quad y_i = \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ 0 \\ \vdots \end{bmatrix}}_{k} \rightarrow \text{j-th position} \quad (\text{encodes } x_i \in C_j)$$

$\rightarrow$ j-th column of the identity matrix

$y(x) = w_0 + w_1 x(1) + w_2 x(2) + \ldots w_d x(d)$ — Linear fit

Linear Discriminants : $\bar{w}_j^T x + w_{j0}$ for j-th class

$$\begin{array}{l} C_1 \\ \\ C_2 \\ \\ C_k \end{array} \underbrace{\begin{bmatrix} 1 & x_1(1) & x_1(2) & \ldots & x_1(d) \\ & & \vdots & & \\ 1 & x_i(1) & x_i(2) & \ldots & x_i(d) \\ & & \vdots & & \\ 1 & x_N(1) & x_N(2) & \ldots & x_N(d) \end{bmatrix}}_{X} \; \underbrace{\overbrace{\begin{bmatrix} w_{10} & w_{20} & \ldots & w_{k0} \\ \bar{w}_1 & \bar{w}_2 & \ldots & \bar{w}_k \end{bmatrix}}^{k}}_{W} \simeq \underbrace{\overbrace{\begin{bmatrix} 1 & 0 & 0 & \ldots & 0 \\ 1 & 0 & 0 & \ldots & 0 \\ 1 & 0 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ & & \vdots & & \\ 0 & 0 & \ldots & & 1 \\ 0 & 0 & \ldots & & 1 \end{bmatrix}}^{k}}_{Y}$$

$$XW \simeq Y \Rightarrow \min_W \frac{1}{2} \|XW - Y\|_F^2 \qquad W \in \mathbb{R}^{(d+1) \times k}.$$

$$(X^T X) W^* = X^T Y$$

$$\boxed{W^* = (X^T X)^{-1} X^T Y}$$

XW^* is prediction on training data

$x \in \mathbb{R}^{d+1} = \begin{bmatrix} 1 \\ x(1) \\ \vdots \\ x(d) \end{bmatrix}$

$$x^T W^* = \underbrace{[0.01 \quad 0.02 \ldots \quad \overset{\text{j-th position}}{0.98} \ldots]}_{\text{k-dimensional}} \quad \text{if } x_i \in C_j$$

However, the above W^* can lead to

$$x^T W^* \simeq [-5.1 \quad -3.1 \quad 0 \quad 3.2 \quad 3.1 \ldots 0.9]$$

and clearly, this is not a good approximation to an indicator vector → [0 . . 0 1 . . 0] (1-hot vector)

Just Least Squares for Classification has obvious drawbacks

Logistic Regression

We had modeled each class as a Gaussian with covariance Σ:

$$\log \frac{p(C_i|x)}{p(C_j|x)} = \underbrace{\log \frac{p(C_i)}{p(C_j)} - \frac{1}{2}(m_i+m_j)^T \Sigma^{-1}(m_i-m_j)}_{w_0} + \underbrace{x^T \Sigma^{-1}(m_i-m_j)}_{w^T x}$$

$$\boxed{w_0 + w^T x}$$

K-class problem

$$\log\left(\frac{p(C_1|x)}{p(C_K|x)}\right) = w_0 + w_1^T x = w_1^T x \quad — (1)$$

$$\log\left(\frac{p(C_2|x)}{p(C_K|x)}\right) = w_2^T x \quad — (2)$$

, $w_i \in \mathbb{R}^{d+1}$
$x \in \mathbb{R}^{d+1}$

$$\vdots \qquad \vdots \qquad \vdots$$

$$\log \frac{p(C_{K-1}|x)}{p(C_K|x)} = w_{K-1}^T x \quad — (K-1)$$

Write $p(C_i|x)$ as p_i

$$\log \frac{p_1}{p_K} = w_1^T x \Rightarrow \frac{p_1}{p_K} = e^{w_1^T x} \quad — (1)$$

$$\frac{p_2}{p_K} = e^{w_2^T x} \quad — (2)$$

$$\vdots$$

$$\frac{p_{K-1}}{p_K} = e^{w_{K-1}^T x} \quad — (K-1)$$

(1) + (2) + . . . (K-1)

$$\frac{p_1}{p_K} + \frac{p_2}{p_K} + \dots + \frac{p_{K-1}}{p_K} = e^{w_1^T x} + e^{w_2^T x} + \dots e^{w_{K-1}^T x}$$

$$\Rightarrow \frac{p_1 + p_2 + \cdots + p_{k-1}}{p_k} = \sum_{i=1}^{k-1} e^{w_i^T x}$$

$$\frac{1-p_k}{p_k} = \sum_{i=1}^{k-1} e^{w_i^T x}$$

$$1 - p_k = p_k \sum_{j=1}^{k-1} e^{w_j^T x}$$

$$1 = p_k \left(1 + \sum_{j=1}^{k-1} e^{w_j^T x}\right)$$

$$\Rightarrow \boxed{p_k = \frac{1}{1 + \sum_{j=1}^{k-1} e^{w_j^T x}}}$$

$$\frac{p_i}{p_k} = e^{w_i^T x}$$

$$\Rightarrow \boxed{p_i = \frac{e^{w_i^T x}}{1 + \sum_{j=1}^{k-1} e^{w_j^T x}}}, \quad i = 1, 2, \ldots k-1$$

$w_1, w_2, w_3, \ldots, w_{k-1}$ are all parameters.

How do we find them?

Parameters in logistic regression $\{w_i \in \mathbb{R}^{d+1}\}_{i=1}^{k}$ are usually fit by maximum likelihood.

Consider the 2-class problem

$p(C_1|x) = p$ $\qquad$ $p(C_1|x) + p(C_2|x) = 1$

$p(C_2|x) = 1 - p$ $\qquad$ $p = f(w)$

(x_i, y_i) is training data, $i = 1, 2, \ldots N$

Let $y_i = 1$ when $x_i \in C_1$

$y_i = 0$ when $x_i \in C_2$

$$\boxed{\text{Data Likelihood} : \prod_{i=1}^{N} p^{y_i} (1-p)^{1-y_i}}$$

(x_i, y_i)

If $x_i \in C_i$, $y_i = 1$

$p^{y_i} (1-p)^{1-y_i}$

$$= p^1(1-p)^{1-1} = p.(1-p)^0 = p$$

Log-likelihood $\ell(w)$

$$\ell(w) = \sum_{i=1}^{N} \log\left(p^{y_i}(1-p)^{1-y_i}\right)$$

$$= \sum_{i=1}^{N} \log p^{y_i} + \log(1-p)^{1-y_i}$$

$$= \sum_{i=1}^{N} \left[y_i \log p + (1-y_i)\log(1-p)\right]$$

Maximum log-likelihood

$p = p(w)$

$$\max_w \ell(w) = \boxed{\max_w} \boxed{\left[\sum_{i=1}^{N}\left\{y_i \log p + (1-y_i)\log(1-p)\right\}\right]}$$

$$p = p(w) = \frac{e^{w^T x_i}}{1+e^{w^T x_i}}, \quad 1-p = 1 - \frac{e^{w^T x_i}}{1+e^{w^T x_i}} = \frac{1}{1+e^{w^T x_i}}$$

$$\log p = w^T x_i - \log\left(1+e^{w^T x_i}\right) \qquad \log(1-p) = -\log\left(1+e^{w^T x_i}\right)$$

$$\underbrace{\nabla_w \ell(w)}_{} = 0$$

will involve $\nabla_w \log p$ & $\nabla_w \log(1-p)$

$$\boxed{\nabla_w \log p} = x_i - \frac{1}{1+e^{w^T x_i}} e^{w^T x_i} x_i = \boxed{x_i\left(1 - \frac{e^{w^T x_i}}{1+e^{w^T x_i}}\right)}$$

$$\boxed{\nabla_w \log(1-p)} = \boxed{-\frac{e^{w^T x_i}}{1+e^{w^T x_i}} x_i}$$

$$\nabla_w \ell(w) = \sum_{i=1}^{N}\left[y_i \nabla_w \log p + (1-y_i)\nabla_w \log(1-p)\right]$$

$$= \sum_{i=1}^{N}\left[y_i x_i\left(1 - \frac{e^{w^T x_i}}{1+e^{w^T x_i}}\right) + (1-y_i)x_i\left(-\frac{e^{w^T x_i}}{1+e^{w^T x_i}}\right)\right]$$

$$= \sum_{i=1}^{N} x_i\left[y_i - \cancel{y_i\frac{e^{w^T x_i}}{1+e^{w^T x_i}}} - \frac{e^{w^T x_i}}{1+e^{w^T x_i}} + \cancel{y_i\frac{e^{w^T x_i}}{1+e^{w^T x_i}}}\right]$$

$$\Rightarrow \nabla_w \ell(w) = \sum_{i=1}^{N} x_i\left[y_i - \frac{e^{w^T x_i}}{1+e^{w^T x_i}}\right] \rightarrow p(C_1|x)$$

To find w that maximizes $l(w)$

$$\nabla_w l(w) = 0$$

$$\boxed{\nabla_w l(w) = \sum_{i=1}^{N} x_i \left[y_i - \frac{e^{w^T x_i}}{1+e^{w^T x_i}} \right] = 0}, \quad x_i \in \mathbb{R}^{d+1}, \quad w \in \mathbb{R}^{d+1}$$

$(d+1)$ parameters

$(d+1)$ equations since $x_i \in \mathbb{R}^{d+1}$

nonlinear equations

There is no closed form solution for above

How do we solve for w?

Need to use optimization methods to solve for these parameters

- Gradient Descent / Ascent
- Newton's Method

$w_{j+1} \leftarrow w_j - \eta \nabla_w l(w)$ — Gradient Descent

$w_{j+1} \leftarrow w_j - \eta \left(\nabla^2 l(w)\right)^{-1} \nabla l(w)$ — Newton's Method

Drawback of these methods (Gradient Descent, Newton) is that each step requires $O(N)$ computation

Not feasible when N is very large

Stochastic Gradient Descent (SGD)

Regularization: $\lambda \|w\|_2^2$ or $\lambda \|w\|_1$ should be used