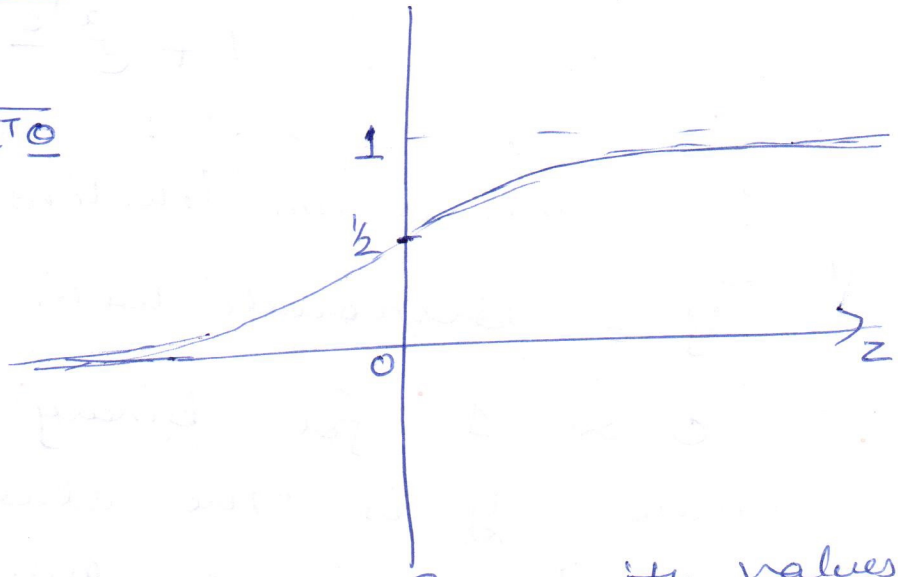
Logistic Regression

linear regression —

$$y = \underline{x}^T \underline{\theta} + \epsilon$$

logistic function is —

$$y = \frac{1}{1 + e^{-\underline{x}^T \underline{\theta}}}$$



let $\underline{x}^T \underline{\theta} = z$

$z = 0$, $y = \frac{1}{2}$

$z = \infty$, $y = 1$

$z = -\infty$, $y = 0$

∴ y looks like a prob. fn. with values between 0 & 1

Also, this function y is smooth & and hence differentiable, & also symmetric.

For binary classification,

$y \geq \frac{1}{2}$ ⟶ predict class-1

$y < \frac{1}{2}$ ⟶ " class-0.

$$\therefore p(y=1/\underline{x}) = \text{logistic}(\underline{x}, \underline{\theta})$$

$$= \frac{1}{1 + e^{-\underline{x}^T \underline{\theta}}}$$

$$= \frac{e^{\underline{x}^T \underline{\theta}}}{1 + e^{\underline{x}^T \underline{\theta}}}$$

$$\therefore \quad p(y=0/\underline{x}) = 1 - p(y=1/\underline{x})$$

$$= 1 - \frac{1}{1+e^{-\underline{x}^T\underline{\theta}}} = \frac{e^{-\underline{x}^T\underline{\theta}}}{1+e^{-\underline{x}^T\underline{\theta}}}$$

$$= \frac{1}{1+e^{\underline{x}^T\underline{\theta}}}$$

Use maximum likelihood estimator —

Actual $y \equiv$ Bernoulli with two values

0 & 1 for binary classification problem, where y is the class label.

Let there be n no. of independent observations of y for $i = 1, 2, \ldots, n$ generated by n feature vectors

$\underline{x}_1, \ldots \underline{x}_n$.

Conditional likelihood fn is —

$$p(\underline{y}/X, \underline{\theta}) = \prod_{i=1}^{n} p(y_i/\underline{x}_i, \underline{\theta}) = l(\underline{\theta})$$

max. the likelihood w.r.t $\underline{\theta}$ will provide its estimate

$$\hat{\underline{\theta}} = \max_{\underline{\theta}} \prod_{i=1}^{n} p(y_i/\underline{x}_i, \underline{\theta})$$

$$= \max_{\underline{\theta}} \prod_{i=1}^{n} \left(p(y_i=1/\underline{x}_i, \underline{\theta})\right)^{y_i} \left(p(y_i=0/\underline{x}_i, \underline{\theta})\right)^{1-y_i}$$

Define log likelihood = $L(\underline{\theta}) = \ln\left(l(\underline{\theta})\right)$

maximizing the $\ln(l(\underline{\theta}))$ will give the same result as of maximizing the $l(\underline{\theta})$ w.r.t. $\underline{\theta}$ because ln is a monotonic fn. —

$$\therefore \hat{\underline{\theta}} = \max_{\underline{\theta}} L(\underline{\theta})$$

$$= \max_{\underline{\theta}} \left\{ \sum_{i=1}^{n} y_i \ln\left(p(y_i=1/\underline{x}_i, \underline{\theta})\right) + \sum_{i=1}^{n} (1-y_i) \ln\left(p(y_i=0/\underline{x}_i, \underline{\theta})\right) \right\}$$

Define loss fn. as —

$$J(\underline{\theta}) = -L(\underline{\theta})$$

maximizing $L(\underline{\theta})$ ≡ minimizing $J(\underline{\theta})$

$$\therefore \hat{\underline{\theta}} = \min_{\underline{\theta}} J(\underline{\theta}) = \min_{\underline{\theta}} -L(\underline{\theta})$$

$$= \min_{\underline{\theta}} \left\{ -\sum_{i=1}^{n} y_i \ln\left(p(y_i=1/\underline{x}_i, \underline{\theta})\right) - \sum_{i=1}^{n} (1-y_i) \ln\left(p(y_i=0/\underline{x}_i, \underline{\theta})\right) \right\}$$

Binary Cross Entropy loss fn. (BCE)

×

Entropy = $-\sum_i p_i \ln p_i$

Cross entropy = $-\sum_i q_i \ln p_i$

$$J(\underline{\theta}) = -\sum_{i=1}^{n} y_i \ln\left(\frac{e^{z_i}}{1+e^{z_i}}\right) - \sum_{i=1}^{n} (1-y_i) \ln\left(\frac{1}{1+e^{z_i}}\right)$$

where $z_i = \underline{x}_i^T \underline{\theta}$

~~minimize $J(\underline{\theta})$ by~~ Taking the derivative of $J(\underline{\theta})$ & computing the gradient of $J(\underline{\theta})$:

$$\frac{\partial}{\partial \underline{\theta}} J(\underline{\theta}) = \frac{\partial}{\partial \underline{\theta}} \left[\sum_{i=1}^{n} y_i \ln(e^{z_i}) + \sum_{i=1}^{n} y_i \ln(1+e^{z_i}) - \sum_{i=1}^{n} \ln(1+e^{z_i}) + \sum_{i=1}^{n} y_i \ln(1+e^{z_i}) \right]$$

$$= \frac{-\partial}{\partial \underline{\theta}} \left[\sum_{i=1}^{n} \left(y_i \ln e^{\underline{x}_i^T \underline{\theta}} - \ln\left(1+e^{\underline{x}_i^T \underline{\theta}}\right) \right) \right]$$

$$= -\sum_{i=1}^{n} \left(\frac{y_i \underline{x}_i e^{\underline{x}_i^T \underline{\theta}}}{e^{\underline{x}_i^T \underline{\theta}}} - \frac{\underline{x}_i e^{\underline{x}_i^T \underline{\theta}}}{1+e^{\underline{x}_i^T \underline{\theta}}} \right)$$

$$= -\sum_{i=1}^{n} \underline{x}_i \underbrace{\left(y_i - p\left(y_i = 1 / \underline{x}_i, \underline{\theta}\right) \right)}_{e_i}$$

prediction error

gradient looks like $-\sum_{i=1}^{n} \underline{x}_i e_i$

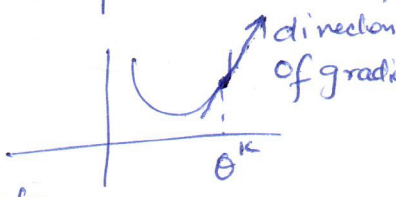
if $y_i = 1 \rightarrow$ we would like $p(y_i = 1 / \underline{x}_i, \underline{\theta}) \rightarrow 1$

if $y_i = 0 \rightarrow$ " " " $p(y_i = 1 / \underline{x}_i, \underline{\theta}) \rightarrow 0$

to keep the error small.

$L(\underline{\theta})$ is a concave fn. in $\underline{\theta}$

& $J(\underline{\theta})$ is " convex fn. in $\underline{\theta}$



$\therefore$ max. $L(\underline{\theta})$ ≡ gradient ascent

& min. ~~$J(\underline{\theta})$~~ $J(\underline{\theta})$ ≡ gradient descent.

will converge to global min.

if we move iteratively as - $\underline{\theta}^{(k+1)} = \underline{\theta}^{(k)} - \eta \left(+ \frac{\partial J(\underline{\theta})}{\partial \underline{\theta}} \right)$ ⇐ Gradient Descent Method

$\underline{\theta}^{(k)}$ ≡ current value of $\underline{\theta}$ in the k^{th} iteration

$\underline{\theta}^{(k+1)}$ ≡ value of $\underline{\theta}$ in the $(k+1)^{th}$ iteration

η = learning rate

$-\frac{\partial J(\underline{\theta})}{\partial \underline{\theta}}$ ≡ move in the direction opposite to the direction of gradient to reach to min.

$\therefore \underline{\theta}^{(k+1)} = \underline{\theta}^{(k)} - \eta \frac{\partial J(\underline{\theta})}{\partial \underline{\theta}}$

$= \underline{\theta}^{(k)} + \eta \sum_{i=1}^{n} \underline{x}_i e_i$

$$+\frac{\partial J(\underline{\theta})}{\partial \underline{\theta}} = -\sum_{i=1}^{n} \underline{x}_i \left(y_i - b(y_i = 1 / \underline{x}_i, \underline{\theta}) \right)$$

$$= -\underset{p \times n}{X^T} \underbrace{\left(\underline{y} - \underline{b} \right)}_{n \times 1} = -X^T \underline{e}$$

$$= -\left(\underline{x}_1 e_1 + \underline{x}_2 e_2 + \cdots + \underline{x}_n e_n \right)$$

Since $X = \begin{bmatrix} \underline{x}_1^T \\ \underline{x}_2^T \\ \vdots \end{bmatrix}$

$\underline{\theta} = p \times 1$ vector

$$\therefore X^T \underline{e} = \underset{p \times n}{\begin{bmatrix} \underline{x}_1 & \underline{x}_2 & \cdots & \underline{x}_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}} \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$

$$= \underline{x}_1 e_1 + \cdots + \underline{x}_n e_n$$

$$\underline{y} = \underset{n \times 1}{\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}} \quad ; \quad \underline{b} = \underset{n \times 1}{\begin{bmatrix} b(y_1 = 1 / \underline{x}_1, \underline{\theta}) \\ \vdots \\ b(y_n = 1 / \underline{x}_n, \underline{\theta}) \end{bmatrix}}$$

$$\frac{\partial^2 J(\underline{\theta})}{\partial \underline{\theta} \, \partial \underline{\theta}^T} = \text{Hessian matrix}$$

$$= \underbrace{\frac{\partial}{\partial \underline{\theta}^T} \Big(\underbrace{-X^T (\underline{y} - \underline{b})}_{p \times 1} \Big)}_{p \times p \text{ matrix}}$$

$$\frac{\partial^2 J(\underline{\theta})}{\partial \underline{\theta}\, \partial \underline{\theta}^T} = + \frac{\partial}{\partial \underline{\theta}^T}\left(-X^T(\underline{y} - \underline{p})\right)$$

$$= \frac{\partial}{\partial \underline{\theta}^T}\left(-X^T\underline{y} + X^T\begin{pmatrix} \frac{1}{1+e^{-\underline{x}_1^T\underline{\theta}}} \\ \frac{1}{1+e^{-\underline{x}_2^T\underline{\theta}}} \\ \vdots \\ \frac{1}{1+e^{-\underline{x}_n^T\underline{\theta}}} \end{pmatrix}\right)$$

$-X^T\underline{y}$: not a fn. of $\underline{\theta}$

$$= \frac{\partial}{\partial \underline{\theta}^T}\left[X^T\begin{pmatrix} \frac{1}{1+e^{-\underline{x}_1^T\underline{\theta}}} \\ \vdots \\ \frac{1}{1+e^{-\underline{x}_n^T\underline{\theta}}} \end{pmatrix}\right] = \frac{\partial}{\partial \underline{\theta}^T}\left(X^T\underline{p}\right)$$

$$= X^T\left[\frac{\partial}{\partial \underline{\theta}^T}\underline{p}\right]$$

$$\frac{\partial}{\partial \underline{\theta}^T}\underline{p} = \left[\frac{\partial}{\partial \theta_1}\underline{p} \quad \frac{\partial}{\partial \theta_2}\underline{p} \quad \cdots \quad \frac{\partial}{\partial \theta_p}\underline{p}\right]$$

$$\frac{\partial}{\partial \theta_1}\underline{p} = \frac{\partial}{\partial \theta_1}\begin{bmatrix} \frac{1}{1+e^{-\underline{x}_1^T\underline{\theta}}} \\ \vdots \\ \frac{1}{1+e^{-\underline{x}_n^T\underline{\theta}}} \end{bmatrix} = \begin{bmatrix} x_{11}\,\frac{e^{-\underline{x}_1^T\underline{\theta}}}{\left(1+e^{-\underline{x}_1^T\underline{\theta}}\right)^2} \\ \vdots \\ x_{n1}\,\frac{e^{-\underline{x}_n^T\underline{\theta}}}{\left(1+e^{-\underline{x}_n^T\underline{\theta}}\right)^2} \end{bmatrix}$$

$$= \begin{bmatrix} x_{11}\,p(y_1=1/\underline{x}_1,\underline{\theta})\left(1-p(y_1=1/\underline{x}_1,\underline{\theta})\right) \\ \vdots \\ x_{n1}\,p(y_n=1/\underline{x}_n,\underline{\theta})\left(1-p(y_n=1/\underline{x}_n,\underline{\theta})\right) \end{bmatrix}$$

$$\frac{\partial \underline{p}}{\partial \theta_1} = \begin{bmatrix} x_{11}\, p_1(1-p_1) \\ \cdot \\ \cdot \\ \cdot \\ x_{n1}\, p_n(1-p_n) \end{bmatrix}$$

$$= \begin{bmatrix} p_1(1-p_1) & 0 & \cdots & 0 \\ 0 & p_2(1-p_2) & \cdots & 0 \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdots & p_n(1-p_n) \end{bmatrix} \begin{bmatrix} x_{11} \\ \cdot \\ \cdot \\ \cdot \\ x_{n1} \end{bmatrix}$$

$n \times n$ diagonal matrix

$$= W \begin{bmatrix} x_{11} \\ \cdot \\ \cdot \\ \cdot \\ x_{n1} \end{bmatrix}$$

Similarly,

$$\frac{\partial \underline{p}}{\partial \theta_2} = W \begin{bmatrix} x_{12} \\ \cdot \\ \cdot \\ \cdot \\ x_{n2} \end{bmatrix}$$

$$\frac{\partial \underline{p}}{\partial \theta_p} = W \begin{bmatrix} x_{1n} \\ \cdot \\ \cdot \\ \cdot \\ x_{nn} \end{bmatrix}$$

$$\therefore \begin{bmatrix} \frac{\partial \underline{p}}{\partial \theta_1} & \cdots & \frac{\partial \underline{p}}{\partial \theta_p} \end{bmatrix} = W \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ x_{n1} & & x_{nn} \end{bmatrix}$$

$$= WX$$

$$\frac{\partial^2 J(\underline{\theta})}{\partial \underline{\theta}\, \partial \underline{\theta}^T} = X^T \left[\frac{\partial}{\partial \underline{\theta}} \underline{b} \right]$$

$$= X^T \, I_{d} \, X$$

Newton's algorithm of optimization —

This can be used on a twice differentiable fn. & is based on the quadratic Taylor's expansion.

Let us consider the minimization of a fn. of one variable — say global min. is at θ^*.

Guess an initial $\theta = \theta^{(0)}$ ← can be zero.

Use 2nd order Taylor's series expansion in the neighborhood of $\theta^{(0)}$ —

$$f(\theta) \approx f(\theta^{(0)}) + (\theta - \theta^{(0)}) \frac{df}{d\theta}\bigg|_{\theta=\theta^{(0)}} + \frac{1}{2}(\theta - \theta^{(0)})^2 \frac{d^2 f}{d\theta^2}\bigg|_{\theta=\theta^{(0)}}$$

$$= f(\theta^{(0)}) + (\theta - \theta^{(0)}) f'(\theta^0) + \frac{1}{2}(\theta - \theta^{(0)})^2 f''(\theta^0)$$

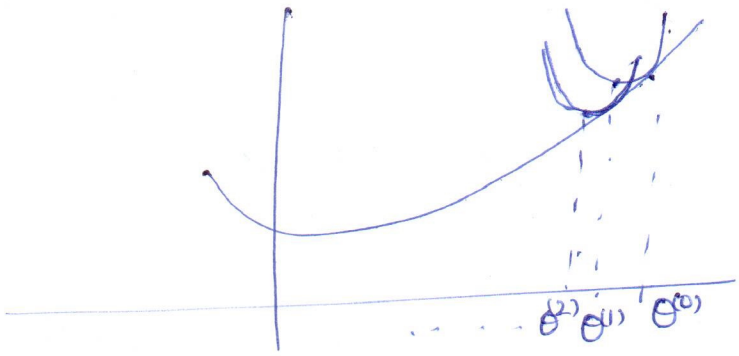
Take the derivative & equate to zero —

$$f'(\theta) = 0 = f'(\theta^{(0)}) + \frac{1}{2} \cdot 2 (\theta^{(1)} - \theta^{(0)}) f''(\theta^{(0)})$$

$$\Rightarrow \quad \theta^{(1)} - \theta^{(0)} = \frac{-f'(\theta^0)}{f''(\theta^0)}$$

$$\Rightarrow \quad \theta^{(1)} = \theta^{(0)} - \frac{f'(\theta^{(0)})}{f''(\theta^{(0)})}$$

$$\theta^{(1)} = \theta^{(0)} - \left(\nabla^2 f(\theta^{(0)})\right)^{-1} \nabla f(\theta^{(0)})$$



Newton's method of moving to the bottom of the curve.

converges faster than the gradient descent method -

if we apply this to our problem -

$$\underline{\theta}^{new} = \underline{\theta}^{old}$$

$$\underline{\theta}^{(1)} = \underline{\theta}^{(0)} - \left(\nabla^2 J(\underline{\theta})\right)^{-1} \nabla\left(J(\underline{\theta})\right)$$

$$= \underline{\theta}^{(0)} - \left(X^T W^{(0)} X\right)^{-1} \left(-X^T (\underline{y} - \underline{p}^{(0)})\right)$$

$$= \underline{\theta}^{(0)} + \left(X^T W^{(0)} X\right)^{-1} \left(X^T (\underline{y} - \underline{p}^{(0)})\right)$$

$$\left(X^T W^{(0)} X\right) \underline{\theta}^{(1)} = \left(X^T W^{(0)} X\right) \underline{\theta}^{(0)} + X^T (\underline{y} - \underline{p}^{(0)})$$

$$= X^T W^{(0)} \underbrace{\left[X \underline{\theta}^{(0)} + W^{(0)^{-1}} (\underline{y} - \underline{p}^{(0)})\right]}_{\underline{z}^{(0)}_{p \times 1}}$$

$$= X^T W^{(0)} \underline{z}^{(0)}$$

$$(X^T W^{(0)} X)\ \underline{\theta}^{(1)} = X^T W^{(0)} \underline{z}^{(0)}$$

$$\underline{\theta}^{(1)} = (X^T W^{(0)} X)^{-1} X^T W^{(0)} \underline{z}^{(0)}$$

$$\underline{\theta}^{(k+1)} = (X^T W^{(k)} X)^{-1} X^T W^{(k)} \underline{z}^{(k)} \quad \text{—— (1)}$$

This update solution is also called as the Iteratively reweighted least squares (IRLS) method.

remember least squares soln —

$$\boxed{\underline{\theta} = (X^T X)^{-1} X^T \underline{y}} \quad \text{—— (2)}$$

Compare (1) & (2) —

there is a diagonal weight matrix in (1)

$$w_1^{(k)} = \frac{e^{-x_1^T \underline{\theta}^{(0)}}}{\left(1 + e^{-x_1^T \underline{\theta}^{(0)}}\right)^2}$$

$$w_n^{(k)} = \frac{e^{-x_n^T \underline{\theta}^{(0)}}}{\left(1 + e^{-x_n^T \underline{\theta}^{(0)}}\right)^2}$$

$$w_1^{(k)} = \frac{e^{-x_1^T \underline{\theta}^{(k)}}}{\left(1 + e^{-x_1^T \underline{\theta}^{(k)}}\right)^2}$$

$$w_n^{(k)} = \frac{e^{-x_n^T \underline{\theta}^{(k)}}}{\left(1 + e^{-x_n^T \underline{\theta}^{(k)}}\right)^2}$$