Machine Learning

Logistic Regression Classifier

Dr. Mehran Safayani

safayani@iut.ac.ir

safayani.iut.ac.ir

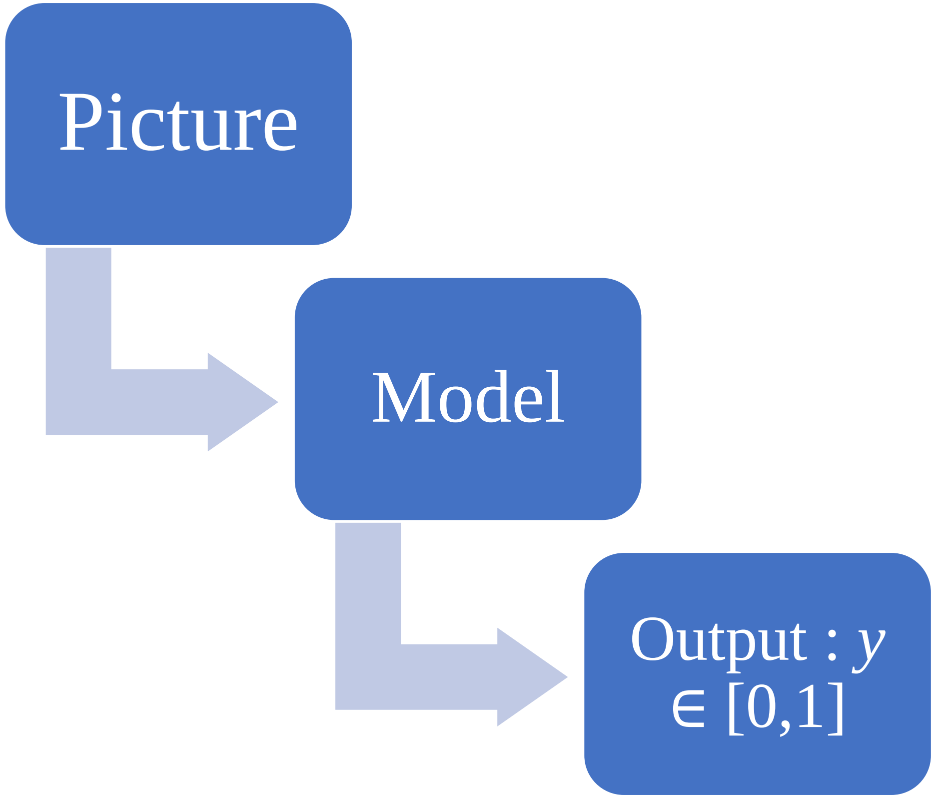
https://www.aparat.com/mehran.safayani

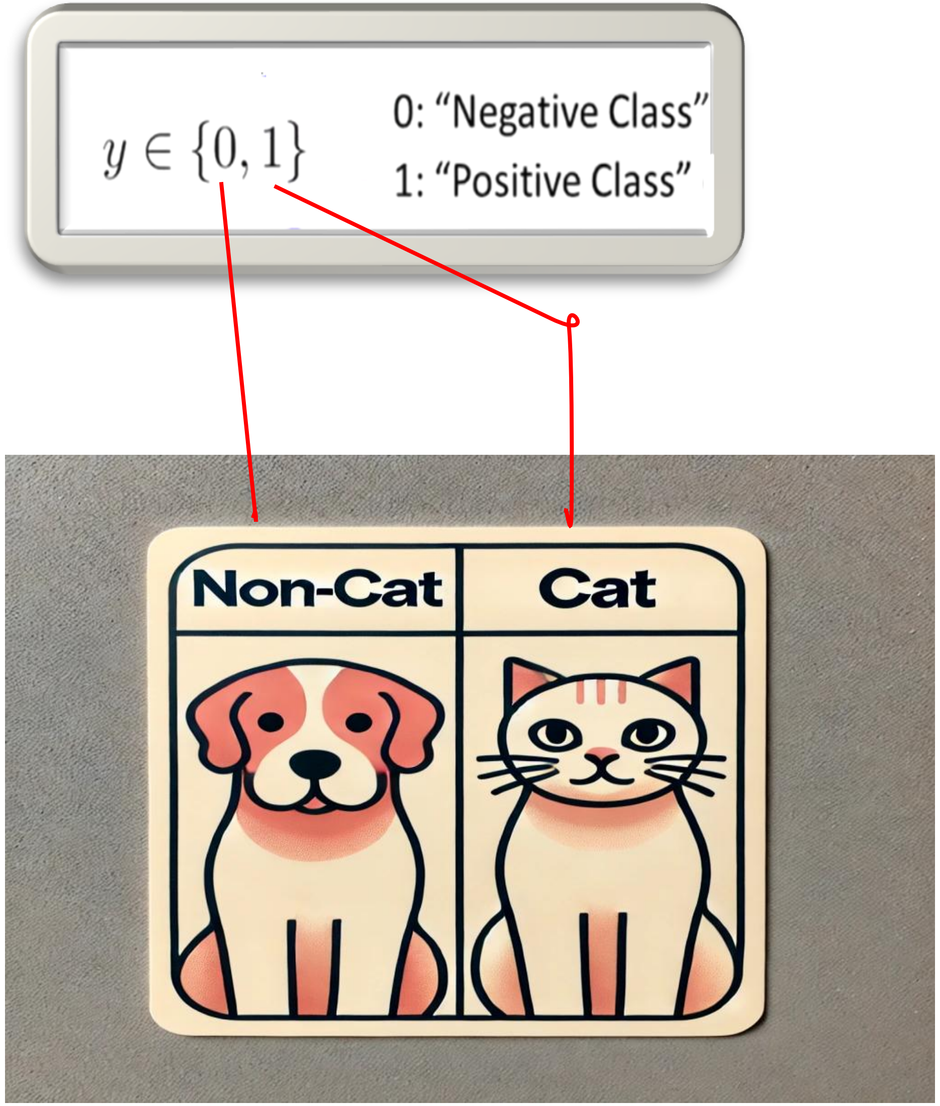
https://github.com/safayani/machine_learning_course

Department of Electrical and computer engineering, Isfahan university of technology, Isfahan, Iran

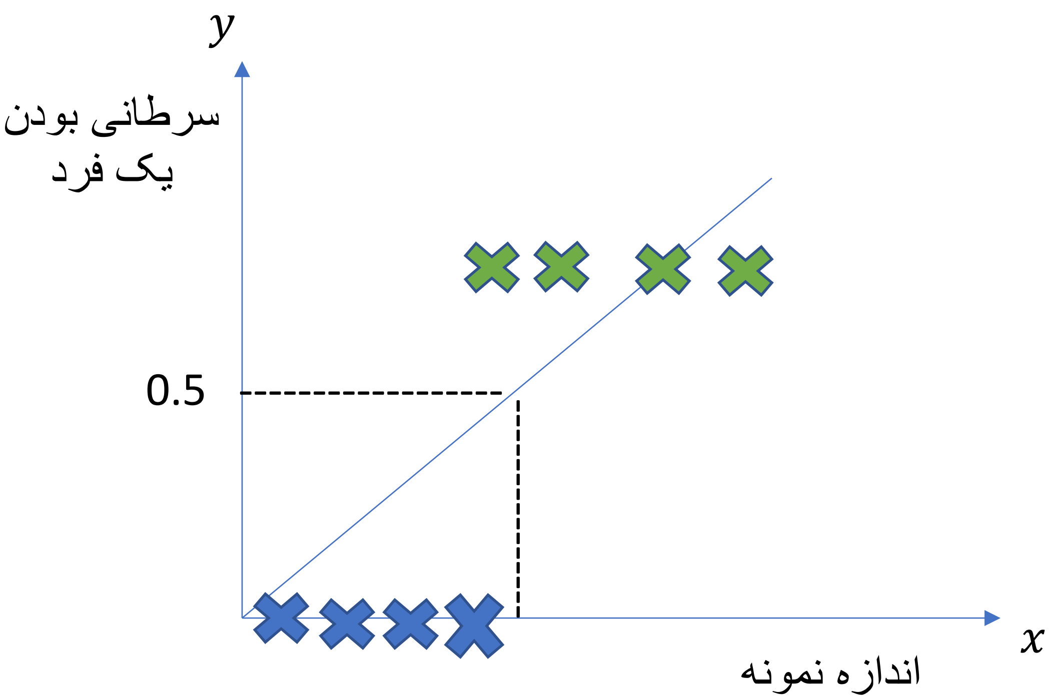
Classification

- Email: spam/not spam
- Animal: cat/non cat





Classification



$$If: h_\theta(x) \geq 0.5 \rightarrow predict, y = 1$$
$$If: h_\theta(x) \leq 0.5 \rightarrow predict, y = 0$$

$$0 \leq h_\theta(x) \leq 1$$
$$0 \leq h_\theta(x) = P(y = 1)|x) \leq 1$$

Logistic Regression

مبانی رگرسیون لجستیک:

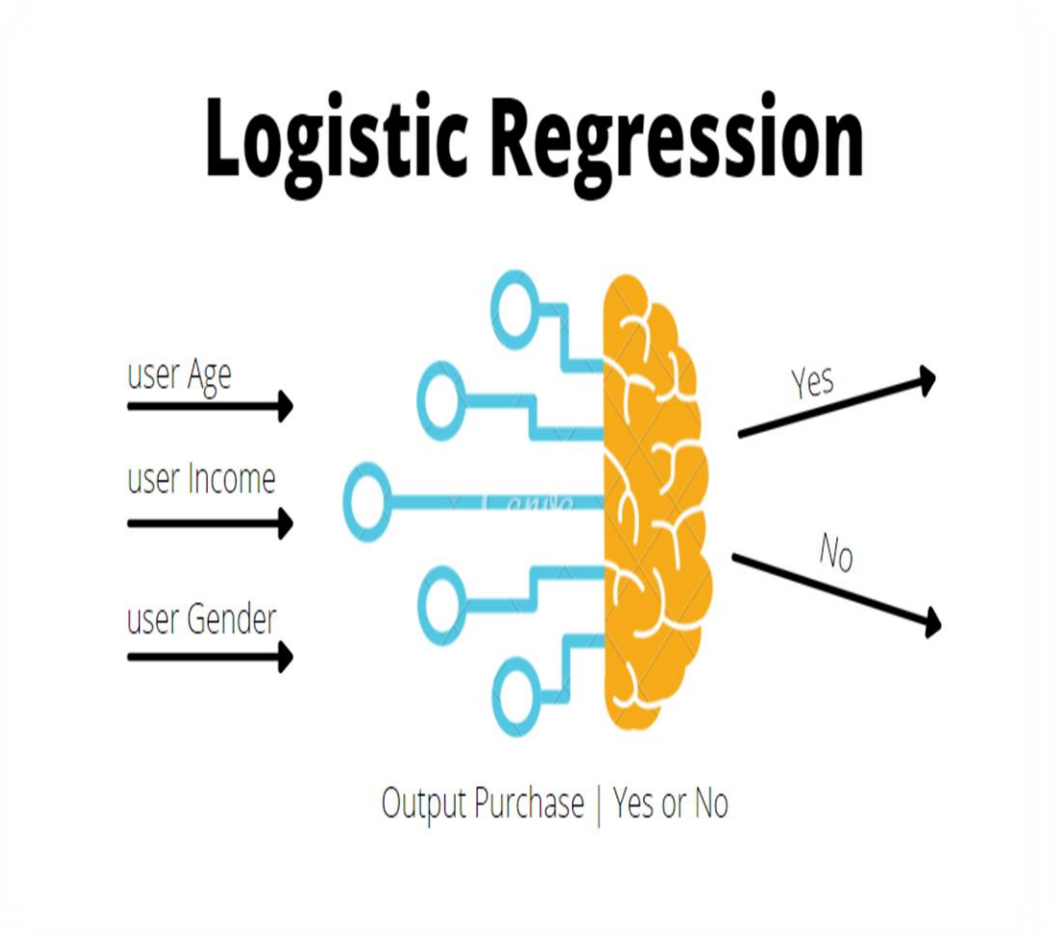
- عمدتاً برای مسائل طبقه‌بندی دودویی استفاده می‌شود.
- احتمال تعلق یک ورودی به یکی از دو کلاس را مدل‌سازی می‌کند.

مرزهای تصمیم‌گیری خطی:

- رگرسیون لجستیک به‌طور معمول مرزهای تصمیم‌گیری خطی را در فرم استاندارد خود ایجاد می‌کند.

کاربردها:

- در مسائلی مانند طبقه‌بندی تصویر، تشخیص پزشکی و همچنین جایی که سطوح تصمیم پیچیده برای بهبود دقت ضروری هستند، استفاده می شود.

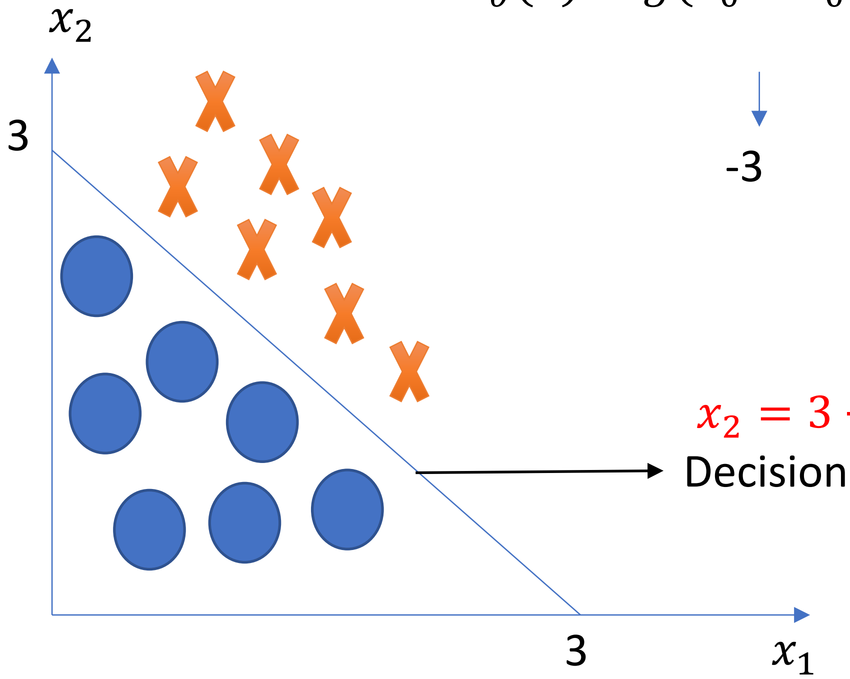


Decision Boundary

$$\theta = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

$$h_\theta(x) = g(\theta_0 + \theta_0 x_1 + \theta_2 x_2)$$

-3 1 1



$x_2 = 3 - x_1$

Decision Boundary

Predict $y = 1, if \underbrace{-3 + x_1 + x_2 \geq 0}_{\theta^T X}$

$x_1 + x_2 \geq 3$

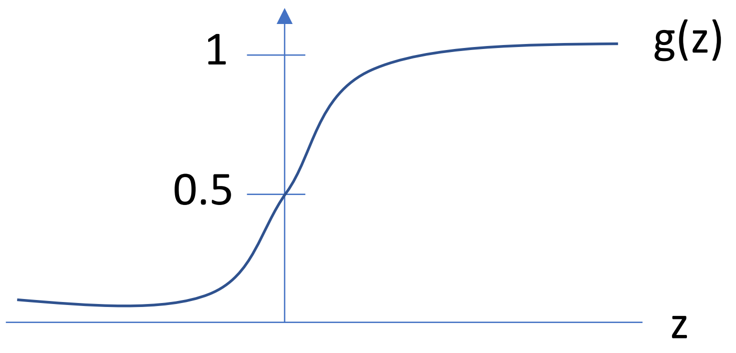
Sigmoid Function: Logistic Function

Want $0 \leq h_\theta(x) \leq 1$

$$h_\theta(x) = g(\theta^T x)$$

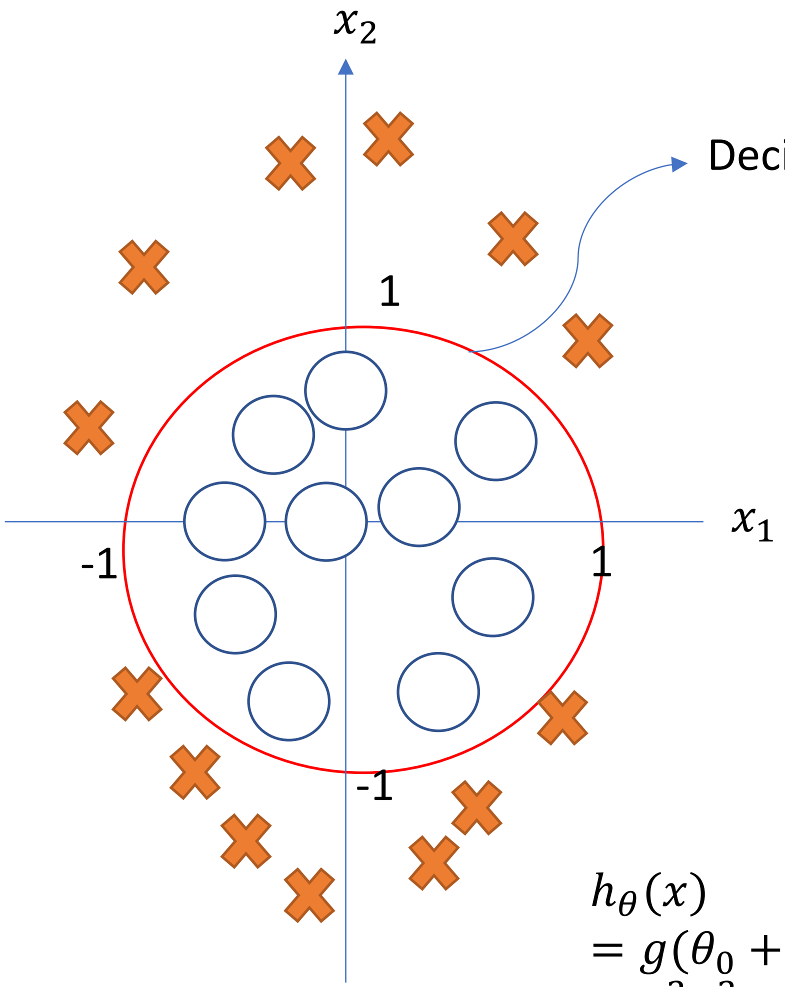
$$g(z) = \frac{1}{1 + e^{-z}}$$

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$



Model output example : $P(y = 1|x) = 0.8$

Non-Linear Decision Boundaries



Decision Boundary

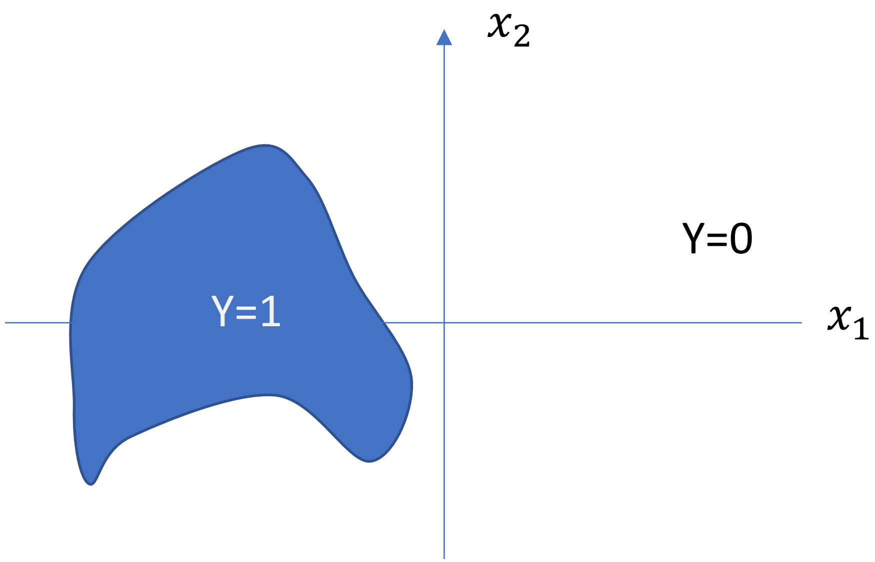
$$h_\theta(x) = g(\overset{-1}{\theta_0} + \overset{0}{\theta_0 x_1} + \overset{0}{\theta_2 x_2} + \theta_3 x_1^2 + \theta_4 x_2^2)$$

$$\theta = \begin{bmatrix} -1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

Predict $y = 1, if \underbrace{-3 + x_1^2 + x_2^2 \geq 0}_{\theta^T X}$

$${x_1}^2 + {x_2}^2 \geq 1$$

$$h_\theta(x) = g(\theta_0 + \theta_0 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_1^2 x_2 + \theta_5 x_1^2 x_2^2 + \theta_6 x_1^3 x_2 + \ldots)$$



Cost Function

Training Set: { $(x^1, y^1), (x^2, y^2), \ldots, (x^m, y^m)$ }

$$x \in \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}, \; x_0 = 1, \; y \in \{0,1\}$$

Linear Regression: J(θ) = $\frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^i) - y^i)^2$

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

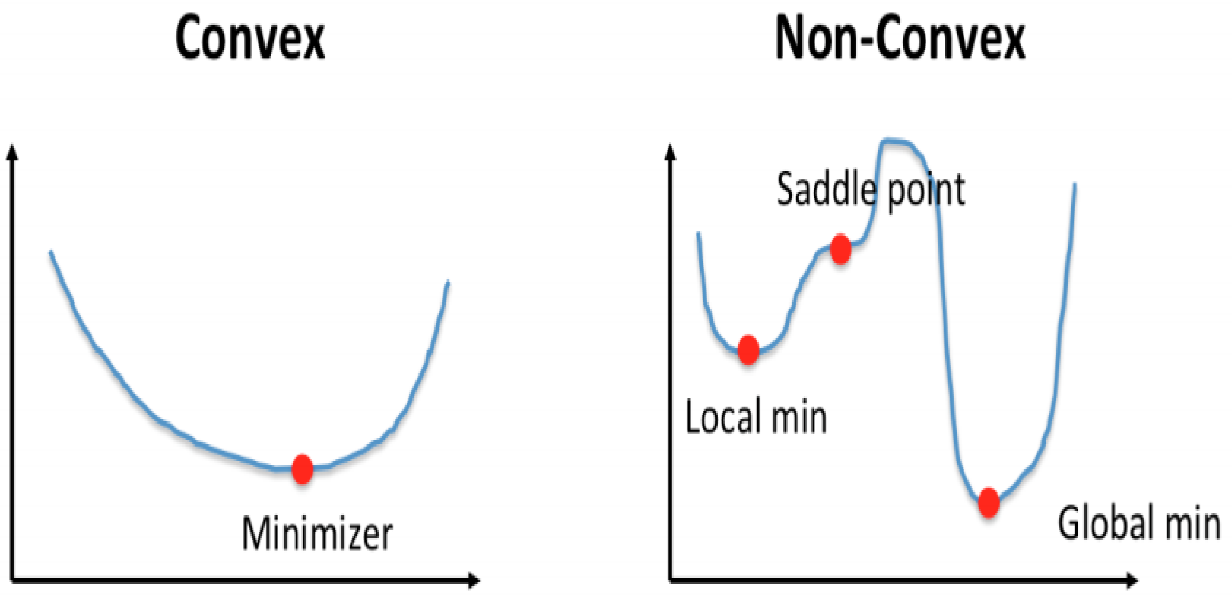
Logistic Regression: $h_\theta(x^i)$ = g($\theta^T x^i$)

MSE: J(θ) = $\frac{1}{2m}\sum_{i=1}^{m}(\mathrm{g}(\theta^T x^i) - y^i)^2$

Convex VS Non-Convex Cost Function

Convex Cost Function:
•Bowl-shaped with a single global minimum.
•Easier optimization, guarantees finding the global minimum.

Non-Convex Cost Function:
•Contains multiple local minima.
•**Challenges**: Optimization algorithms can get stuck in local minima.
•**Impact**: Risk of poor model performance, as finding the global minimum is difficult.



Logistic Regression

MSE Cost Function:

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}((h_\theta(x^{(i)}) - y^{(i)})^2)$$

Binary Cross Entropy cost function:

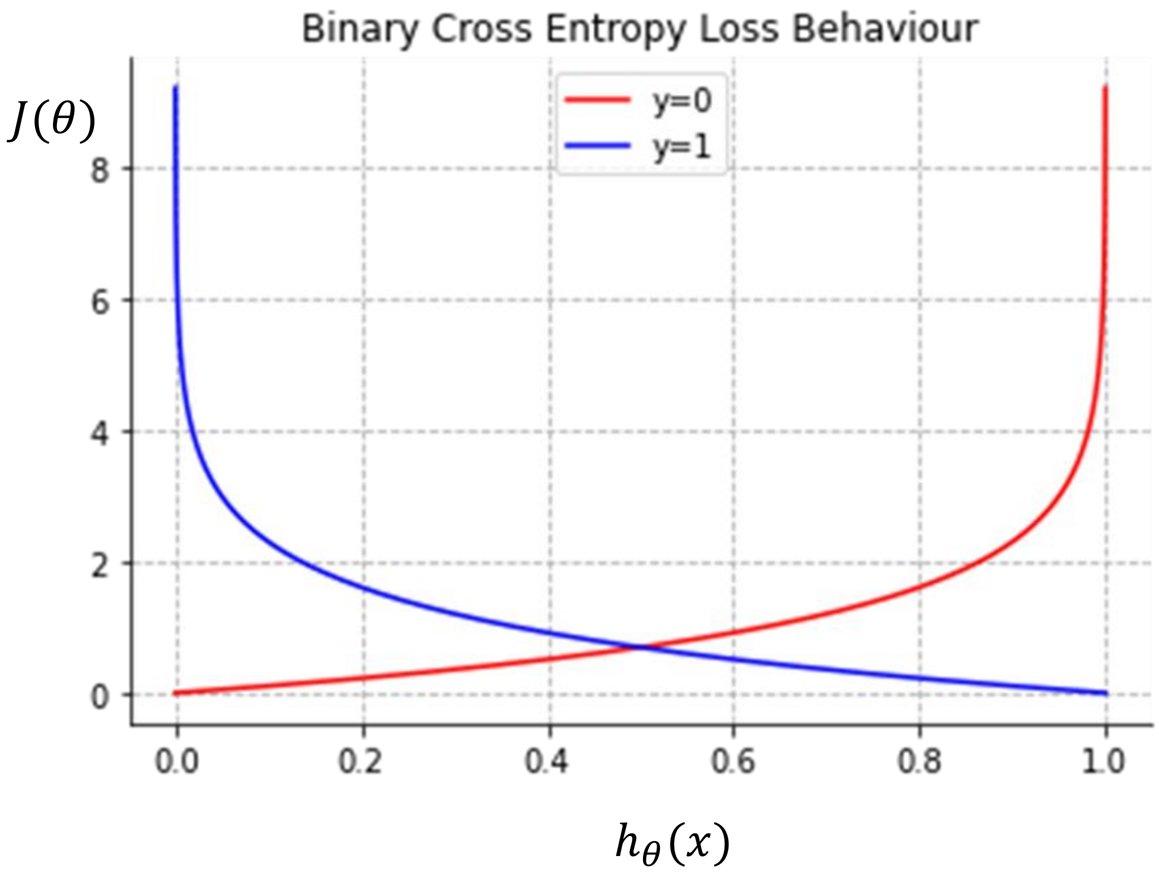
$$cost(h_\theta(x), y) = \begin{cases} -log(h_\theta(x)), & if \quad y = 1 \\ -log(1 - h_\theta(x)), & if \quad y = 0 \end{cases}$$

if y=1 and $h_\theta(x) = 1 \Rightarrow cost = 0;$

if y=1 and $h_\theta(x) = 0 \Rightarrow cost = \infty;$

if y=0 and $h_\theta(x) = 0 \Rightarrow cost = 0;$

if y=0 and $h_\theta(x) = 1 \Rightarrow cost = \infty;$



Logistic Regression

$$cost(h_\theta(x), y) = \begin{cases} -log(h_\theta(x)), & if \quad y = 1 \\ -log(1 - h_\theta(x)), & if \quad y = 0 \end{cases}$$

$$cost(h_\theta(x), y) = -ylog\big(h_\theta(x)\big) - (1 - y)log(1 - h_\theta(x))$$

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} \underbrace{-y^i log\left(h_\theta\left(x^i\right)\right) - (1 - y^i)log(1 - h_\theta(x^i))}_{T_i}$$

$$\min_\theta J(\theta)$$

Gradient Descent: Repeat Unit Convergence :

repeat{

$$\theta_j = \theta_j - \alpha\frac{dJ(\theta)}{d_\theta} \quad J = 0, \ldots, n$$

}until convergence

$$\frac{dJ(\theta)}{d_\theta} = ? \qquad J(\theta) = \frac{1}{m}\sum_{i=1}^{m} T_i \qquad \frac{dJ(\theta)}{d_\theta} = \frac{1}{m}\sum_{i=1}^{m} \frac{dT_i}{di}$$

$$T_i = -[y^i \log h_\theta(x^i) + (1-y^i)\log(1-h_\theta(x^i))] \qquad h_\theta(x^i) = \sigma(\theta^T x^i) = \sigma(z^i) = \frac{1}{1+e^{-z^i}}$$

$$T_i = -[y^i \log \sigma(z^i) + (1-y^i)\log(1-\sigma(z^i)]$$

$$(1)\ \frac{dT_i}{d\sigma(z^i)} = -\left[\frac{y^i}{\sigma(z^i)} + (1-y^i).\frac{-1}{1-\sigma(z^i)}\right] = -\left[\frac{y^i}{\sigma(z^i)} - \frac{1-y^i}{1-\sigma(z^i)}\right]$$

$$(2)\ \frac{d\sigma(z^i)}{dz^i} = \frac{e^{-z^i}}{(1+e^{-z^i})^2} = \frac{1}{1+e^{-z^i}}.\frac{e^{-z^i}}{1+e^{-z^i}} = \sigma(z^i).(1-\sigma(z^i)) \qquad ,$$

$$(3)\ \frac{dz^i}{d\theta_j} = x_j^i$$

$$z^i = \theta^T x^i = \theta_0 x_0 + \theta_1 x_1 + \ldots \theta_n x_n$$

$$1-\sigma(z^i) = 1 - \frac{1}{1+e^{-z^i}} = \frac{1+e^{-z^i}-1}{1+e^{-z^i}} = \frac{e^{-z^i}}{1+e^{-z^i}}$$

From (1), (2) and (3):

$$\frac{dT_i}{d\theta_j} = -\left[\frac{y^i}{\sigma(z^i)} - \frac{1-y^i}{1-\sigma(z^i)}\right]\sigma(z^i).\,(1-\sigma(z^i))x_j^i$$

$$= -\left[y^i.\left(1-\sigma(z^i)\right) - \left(1-y^i\right).\sigma(z^i)\right]x_j^i$$

$$= -\left[y^i - \sigma(z^i)\right]x_j^i = \left[\sigma(z^i) - y^i\right]x_j^i$$

$$\frac{dy(\theta)}{d\theta_j} = \frac{1}{m}\sum_{i=1}^{m}\frac{dT_i}{d\theta_j} = \frac{1}{m}\sum_{i=1}^{m}\frac{dT_i}{dz^i}.\frac{dz^i}{d\theta_j} = \frac{1}{m}\sum_{i=1}^{m}\left(\underbrace{\sigma(z^i)}_{h_\theta(x^i)} - y^i\right).x_j^i$$

GD: $\mathrm{R}peatUntiConvergance\{$

$$\theta_j = \theta_j - \alpha\sum_{i=1}^{m}\left(h_\theta(x^i) - y^i\right).x_j^i$$

$\}$

Logistic regression on m examples

$\theta_1 \leftarrow \text{random} \quad \theta_2 \leftarrow \text{random} \quad b \leftarrow \text{random}$

$$\theta^t = \begin{bmatrix} \theta_1^t \\ \theta_2^t \\ b^t \end{bmatrix} \quad \theta^{t+1} = \begin{bmatrix} \theta_1^{t+1} \\ \theta_2^{t+1} \\ b^{t+1} \end{bmatrix}$$

Repeat{

$J = 0; \quad d\theta_1 = 0; \quad d\theta_2 = 0; \quad db = 0;$

$$\|\theta^{t+1} - \theta^t\|_2 \le \varepsilon$$

For i=1 to m

$z^{(i)} = \theta^T x^{(i)} + b$

$a^{(i)} = \sigma\left(z^{(i)}\right)$

$J \mathrel{+}= \left[y^{(i)} Log a^{(i)} + \left(1 - y^{(i)}\right) Log(1 - a^{(i)})\right]$

$dz^{(i)} = a^{(i)} - y^{(i)}$

$d\theta_1 \mathrel{+}= x_1^{(i)}\, dz^{(i)} \quad d\theta_2 \mathrel{+}= x_2^{(i)}\, dz^{(i)} \quad db \mathrel{+}= dz^{(i)}$

$$d\theta = \begin{bmatrix} d\theta_1 \\ d\theta_2 \\ db \end{bmatrix}$$

$$\|d\theta\| \le \varepsilon = 10^{-4}$$

$J /= m; \quad d\theta_1 /= m; \quad d\theta_2 /= m; \quad db /= m;$

$\theta_1 = \theta_1 - \alpha\, d\theta_1 \quad \theta_2 = \theta_2 - \alpha\, d\theta_2 \quad b = b - \alpha\, db$

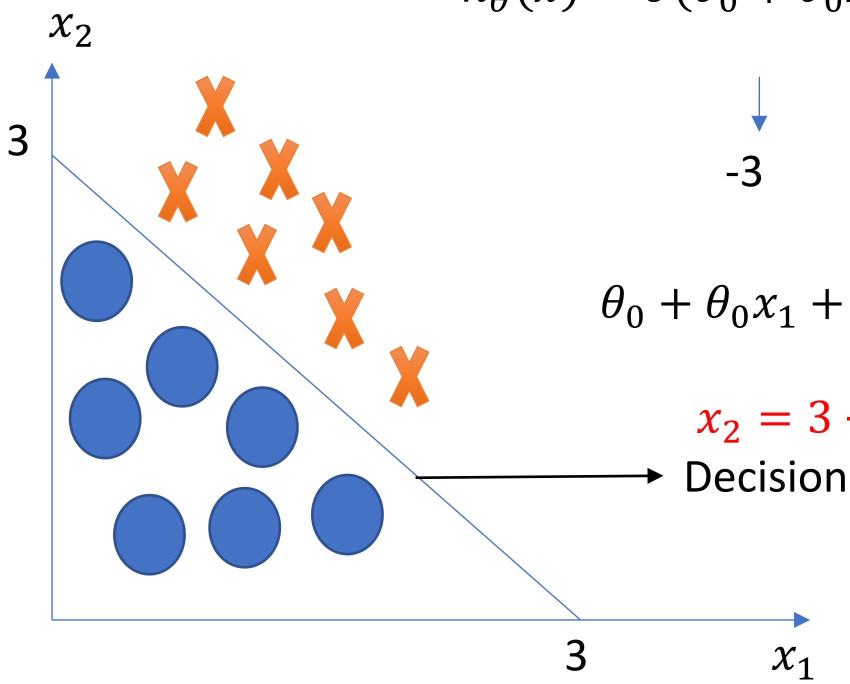
} until convergence

Decision Boundary of logistic regression

$$\theta = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

$$h_\theta(x) = \sigma(\theta_0 + \theta_0 x_1 + \theta_2 x_2)$$

-3 1 1



$\theta_0 + \theta_0 x_1 + \theta_2 x_2$=-3+ $x_1 + x_2 = 0$

$x_2 = 3 - x_1$

Decision Boundary

تعبیر احتمالاتی رگرسیون لاجستیک

$$p(y, \mathbf{X}|\theta) = p(\mathbf{X}|\theta)p(y|\mathbf{X}, \theta) = p(\mathbf{X})p(y|\mathbf{X}, \theta)$$

$$L = p(Y|X, \theta) = \prod_{i=1}^{m} p(y_i|x_i)$$
$$= \prod_{n:y_n=1} \underbrace{p(y_i = 1|x_i)}_{\mu_i} \prod_{n:y_i=0} \underbrace{p(y_i = 0|x_i)}_{1-\mu_i}$$
$$= \prod_{i=1}^{m} \left[\underbrace{\sigma(x_i^T\theta)}_{\mu_i}\right]^{y_i} \left[\underbrace{1 - \sigma(x_i^T\theta)}_{1-\mu_i}\right]^{1-y_i}$$

Bernoulli distribution

$$p(y|x) = f(x) = \begin{cases} \mu, & y = 1 \\ 1 - \mu, & y = 0 \end{cases}$$
$$\mu_i = h_\theta(x_i) = \sigma(x_i^T\theta)$$

$$LL = \sum_{i=1}^{m} y_i log\big(h_\theta(x_i)\big) + (1 - y_i)log(1 - h_\theta(x_i))$$
$$j(\theta) = -LL$$

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}}\, j(\theta)$$

برای ساده کردن رابطه و هم چنین مسائل محاسباتی ازتابع log میگیریم. هم چنین یک منفی در آن ضرب می کنیم تا تابع را کمینه کنیم وحالت تابع هزینه پیدا کند.

محاسبات رگرسیون لاجستیک به صورت برداری

$$j(\theta) = -\sum_{i=1}^{m} y_i ln\left(\sigma\left(x_i^T\theta\right)\right) + (1-y_i)ln(1-\sigma\left(x_i^T\theta\right))$$

می توان نشان داد:

$$j(\theta) = \sum_{i=1}^{m} ln\left[1+\exp\left(x_i^T\theta\right)\right] - y_i x_i^T\theta$$

می توان نشان داد:

$$\frac{\partial \ln[1+\exp(x)]}{\partial x} = \sigma(x).$$

$$\frac{d \ln\left[1+\exp(\overbrace{x_i^T\theta}^{z})\right]}{\partial\theta} = d\frac{ln[1+\exp(z)]}{\partial z}.\frac{\partial z}{\partial\theta} = \sigma(z)\frac{\partial x_i^T\theta}{\partial\theta} = \sigma(z)x_i$$

بنابراین:

$$\underbrace{\nabla L(\theta)}_{P\times 1} = \sum_{i=1}^{m} x_i\left(\sigma(x_i^T\theta) - y_i\right) = \underbrace{X^T}_{P\times m}\left[\underbrace{\sigma(X\theta)}_{m\times 1} - \underbrace{y}_{m\times 1}\right].$$

$$\alpha_1\overrightarrow{x_1} + \alpha_2\overrightarrow{x_2} + \cdots + \alpha_m\overrightarrow{x_m}$$

بهینه سازی به روش نیوتن

$$\underbrace{\nabla L(\theta)}_{P\times 1}= \underbrace{X^T}_{P\times m}\left[\underbrace{\sigma(X\theta)}_{m\times 1} - \underbrace{y}_{m\times 1}\right]$$

نمی توانیم رابطه بالا را برابر با صفر قرار دهیم و نسبت به w مسئله را حل کنیم بنابراین از روش نزول گرادیانی استفاده می کنیم.

$$\theta^{(t+1)} = \theta^{(t)} - \alpha\nabla L(\theta^{(t)}),$$

روش نیوتن:

از مشتق مرتبه دوم استفاده میکند و در تعداد گام های کمتری همگرا می شود:

ماتریس هسین (Hessian) را به صورت زیر تعریف می کنیم:

$$\mathbf{H}_f = \begin{bmatrix} \frac{\partial^2 f}{\partial\theta_1^2} & \frac{\partial^2 f}{\partial\theta_1\partial\theta_2} & \cdots & \frac{\partial^2 f}{\partial\theta_1\partial\theta_n} \\ \frac{\partial^2 f}{\partial\theta_2\partial\theta_1} & \frac{\partial^2 f}{\partial\theta_2^2} & \cdots & \frac{\partial^2 f}{\partial\theta_2\partial\theta_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial\theta_n\partial\theta_1} & \frac{\partial^2 f}{\partial\theta_n\partial\theta_2} & \cdots & \frac{\partial^2 f}{\partial\theta_n^2} \end{bmatrix}$$

$$(\mathbf{H}_f)_{i,j} = \frac{\partial^2 f}{\partial\theta_i\partial\theta_j}$$

$$\frac{\partial^2 f}{\partial\vec{\theta}\partial\vec{\theta}} = \frac{\partial}{\partial\vec{\theta}}\left(\frac{\partial f}{\partial\vec{\theta}}\right)^T$$

بهینه سازی به روش نیوتن

اگر θ، P بعدی باشه H یک ماتریس $P \times P$است قبلا گرادیان$\frac{dL(\theta)}{\partial\theta}$ را حساب کردیم:

$$\underbrace{\nabla L(\theta)}_{P\times1}= x_i(\sigma(x_i^T\theta) - y_i)$$

نشان دهید اگر از ترانهاده رابطه بالا نسبت به θ یک بار دیگر مشتق بگیریم داریم :

$$\underbrace{x_i}_{P\times1}\ \overbrace{x_i^T}^{1\times P}\ \overbrace{\underbrace{\sigma(x_i^T\theta)}_{scalar}\underbrace{(1-\sigma(x_i^T\theta))}_{scalar}}^{\alpha}$$

$$\underbrace{H(\theta)}_{P\times P} = \sum_{i=1}^{m} \underbrace{x_i}_{P\times1}\ \overbrace{x_i^T}^{1\times P}\ \overbrace{\underbrace{\sigma(x_i^T\theta)}_{scalar}\underbrace{(1-\sigma(x_i^T\theta))}_{scalar}}^{\alpha}$$

بهینه سازی به روش نیوتن

می توان نشان داد برای m نمونه داریم :

$$\underbrace{H(\theta)}_{P\times P} = \sum_{i=1}^{m} \underbrace{x_i}_{P\times 1} \overbrace{x_i^T}^{1\times P} \overbrace{\underbrace{\sigma(x_i^T\theta)}_{scalar} \underbrace{(1-\sigma(x_i^T\theta))}_{scalar}}^{\alpha}$$

$$H(\theta) = X^T S X,$$

S یک ماتریس قطری m×m است.

$$S_{ii} := \sigma(x_i^T\theta)\left[1-\sigma(x_i^T\theta)\right]$$

به روز رسانی به روش نیوتن

$$\theta^{t+1} = \theta^{(t)} - \alpha(H^{(t)})^{-1}\nabla L(\theta^{(t)}).$$

بهینه سازی به روش نیوتن

$$\theta^{t+1} = \theta^{(t)} - \alpha (H^{(t)})^{-1} \nabla L(\theta^{(t)}).$$

این رابطه چگونه بدست آمد :

بسط تابع تیلور تابع L(θ) را در نقطه θ^* می نویسیم :

$$L(\theta) \approx L(\theta^*) + \nabla L(\theta^*)^T (\theta - \theta^*) + \frac{1}{2} (\theta - \theta^*)^T H(\theta^*) (\theta - \theta^*).$$

سمت راست تابع یک تخمین محلی درجه دوم از تابع L(θ) در نقطه θ^*

نقطه کمینه این تخمین محلی را محاسبه می کنیم خوشبختانه رابطه بالا یک راه حل بسته دارد

بهینه سازی به روش نیوتن

$$L(\theta) \approx L(\theta^*) + \nabla L(\theta^*)^T(\theta - \theta^*) + \frac{1}{2}(\theta - \theta^*)^T H(\theta^*)(\theta - \theta^*).$$

$$\frac{\partial L(\theta)}{\partial \theta} = 0$$

$$\frac{\partial (\theta^T) A \theta}{\partial \theta} = (A + A^T)\theta$$

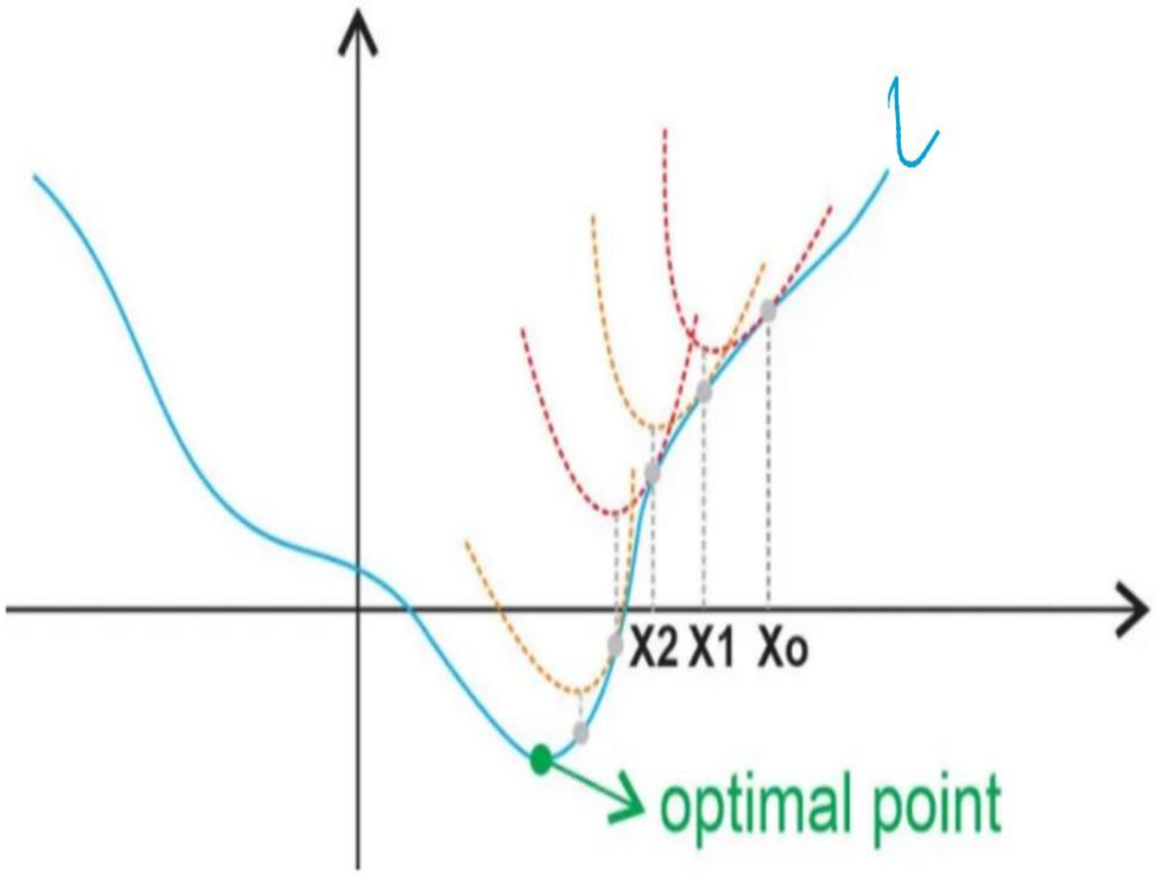
$$\nabla \mathcal{L}(\theta^\star) + \mathbf{H}(\theta^\star)(\theta - \theta^\star) = 0.$$

$$\theta = \theta^\star - \mathbf{H}(\theta^\star)^{-1}\nabla \mathcal{L}(\theta^\star).$$

$$H(\theta^*)\theta = -\nabla \mathcal{L}(\theta^\star) + H(\theta^*)\theta^*$$
$$\theta = H(\theta^*)^{-1}(-\nabla \mathcal{L}(\theta^\star) + H(\theta^*)\theta^*)$$
$$\theta = \theta^* - H(\theta^*)^{-1}\nabla \mathcal{L}(\theta^\star)$$

با توجه به اینکه تابع یک تخمین از تابع اصلی است بهتر است ضریب α هم در فرمول لحاظ شود و در چند گام به جواب برسیم

بهینه سازی به روش نیوتن



Regularized Logistic Regression

اگرچه حد پایین تابع هزینه رگرسیون لاجیستیک صفر است. ولی در حالتی که داده ها خطی تفکیک نشده باشند هیچ θ محدود این حداقل را به ما نمی دهد. و اگر بهینه سازی را ادامه دهیم θ به بی نهایت میل میکند
(این را نشان دهید) (به عبارتی مسئله ما بیش برازش می شود)
برای اجتناب از این مسئله میتوانیم یک ترم جریمه به تابع هزینه اضافه کنیم :

$$argmin_{\theta}\left(-\sum_{n=1}^{N}\ln p(y_n|x_n^T\theta)+\frac{\boldsymbol{\lambda}}{2}||\theta||^2\right)$$

این کار از بی نهایت شدن θ ممانعت می کند.