LOGISTIC REGRESSION

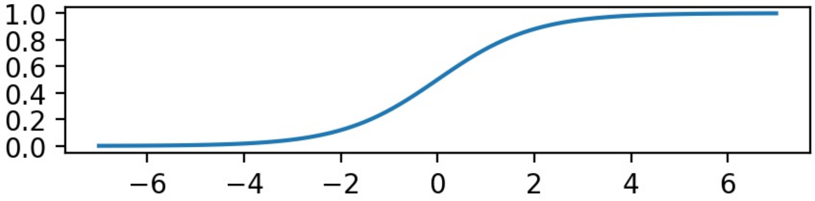
SIEYE RYU

1. Logistic Function

The function g defined by

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

is called the *logistic function.* The logistic function is a common example of a sigmoid function. In general, a *sigmoid function* is a bounded, differentiable real function defined on the real line such that it has a non-negative derivative at each point and exactly one inflection point. The following is the graph of the logistic function:



The point $(0, 0.5)$ is the unique inflection point of the logistic function.

2. Logistic Regression

Suppose that we have m pairs of data $\{(x_i, y_i) \in \mathbb{R} \times \{0, 1\} : i = 1, \cdots, m\}$.

x	y
x_1	y_1
x_2	y_2
$\vdots$	$\vdots$
x_m	y_m

If $t = (t_0, t_1) \in \mathbb{R}^2$, we define the cost function $\text{cost}(h_t(x_i), y_i)$ by

$$\text{cost}(h_t(x_i), y_i) = -y_i \ln(h_t(x_i)) - (1 - y_i) \ln(1 - h_t(x_i)) \tag{2.1}$$

for each $i = 1, \cdots, m$. Suppose that $\theta = (\theta_0, \theta_1)$ minimizes the sum of costs

$$J(t_0, t_1) = \frac{1}{m} \sum_{i=1}^{m} \text{cost}(h_t(x_i), y_i).$$

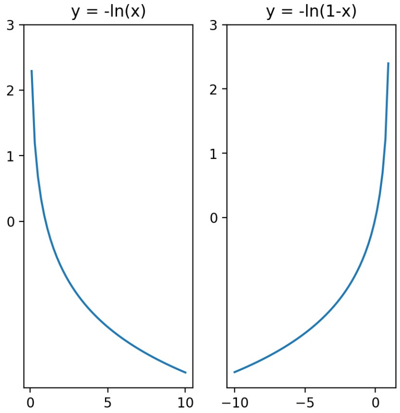
The logistic regression predictor is defined by

$$h_\theta(x) = g(\theta_0 + \theta_1 x) = \frac{1}{1 + \exp(-(\theta_0 + \theta_1 x))}.$$

We note that $h_t(x_i)$ always satisfies $0 \leq h_t(x_i) \leq 1$ for any $i = 1, \cdots, m$ and $t \in \mathbb{R}^2$. We also note that the cost function for the logistic regression does not get involved with the least square method. In fact, the least square method does not make sense in the logistic regression, since

$$\sum_{i=1}^{m} (h_t(x_i) - y_i)^2$$

is not convex. Meanwhile, the function $J(t_0, t_1)$ is convex since $y = -\ln(x)$ and $y = -\ln(1-x)$ are convex and $y_i \in \{0, 1\}$ for each $i = 1, \cdots, m$. Thus, $J(t_0, t_1)$ has the minimum. The following are graphs of $y = -\ln(x)$ and $y = -\ln(1-x)$.



We can split (2.1) as follows:

$$\operatorname{cost}(h_t(x), y) = \begin{cases} -\ln(h_t(x)) & \text{if } y = 1 \\ -\ln(1 - h_t(x)) & \text{if } y = 0 \end{cases}$$

If $y = 1$ and $h_t(x) = 1$, then $\operatorname{cost}(h_t(x), y) = 0$. However, if $y = 1$ and $h_t(x)$ goes to 0, then the cost goes to ∞. Similarly, if $y = 0$ and $h_t(x) = 0$, then $\operatorname{cost}(h_t(x), y) = 0$. If $y = 0$ and $h_t(x)$ goes to 1, then the cost goes to ∞.

In order to obtain the minimum of $J(t_0, t_1)$, we partial differentiate the function

$$\begin{aligned} J(t_0, t_1) \quad = \quad & -\frac{1}{m} \sum_{i=1}^{m} \left[y_i \ln\left(\frac{1}{1 + \exp(-(t_0 + t_1 x_i))} \right) \right. \\ & \left. +(1 - y_i) \ln\left(1 - \frac{1}{1 + \exp(-(t_0 + t_1 x_i))} \right) \right]. \end{aligned}$$

By the chain rule, we have

$$\begin{aligned} & \frac{\partial}{\partial t_0} \left(y_i \ln\left(\frac{1}{1 + \exp(-(t_0 + t_1 x_i))} \right) \right) (t_0, t_1) \\ = \quad & y_i (1 + \exp(-(t_0 + t_1 x_i))) \frac{-1}{(1 + \exp(-(t_0 + t_1 x_i)))^2} \exp(-(t_0 + t_1 x_i))(-1) \\ = \quad & \frac{y_i \exp(-(t_0 + t_1 x_i))}{1 + \exp(-(t_0 + t_1 x_i))} \end{aligned}$$

and

$$\begin{aligned} & \frac{\partial}{\partial t_0}\left((1-y_i)\ln\left(1-\frac{1}{1+\exp(-(t_0+t_1x_i))}\right)\right)(t_0,t_1) \\ = \quad & (1-y_i)\frac{-1-\exp(-(t_0+t_1x_i))}{\exp(-(t_0+t_1x_i))}\frac{-1}{(1+\exp(-(t_0+t_1x_i)))^2}\exp(-(t_0+t_1x_i))(-1) \\ = \quad & \frac{-(1-y_i)}{(1+\exp(-(t_0+t_1x_i)))}. \end{aligned}$$

Thus,

$$\begin{aligned} \frac{\partial J}{\partial t_0}(t_0,t_1) &= -\frac{1}{m}\sum_{i=1}^{m}\left[\frac{y_i\exp(-(t_0+t_1x_i))}{1+\exp(-(t_0+t_1x_i))}-\frac{1-y_i}{(1+\exp(-(t_0+t_1x_i)))}\right] \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left[\frac{y_i\,(1+\exp(-(t_0+t_1x_i)))-1}{(1+\exp(-(t_0+t_1x_i)))}\right] \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left[y_i-\frac{1}{(1+\exp(-(t_0+t_1x_i)))}\right] \\ &= \frac{1}{m}\sum_{i=1}^{m}\left(h_t(x_i)-y_i\right). \end{aligned}$$

Similarly, we have

$$\frac{\partial J}{\partial t_1}(t_0,t_1)=\frac{1}{m}\sum_{i=1}^{m}\left(h_t(x_i)-y_i\right)x_i.$$

Since $J(t_0,t_1)$ is convex, if (θ_0,θ_1) is the point satisfying

$$\nabla J(\theta_0,\theta_1)=\left(\frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x_i)-y_i\right),\frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x_i)-y_i\right)x_i\right)=(0,0),$$

then (θ_0,θ_1) minimizes J.

We can find the minimum of J by *gradient descent* as well. We first choose $(\theta_0,\theta_1)\in\mathbb{R}^2$ and set

$$\theta_j^{(0)}=\theta_j$$

and

$$\theta_j^{(r)}=\theta_j^{(r-1)}-\alpha\frac{\partial J}{\partial\theta_j^{(r-1)}}(\theta_0^{(r-1)},\theta_1^{(r-1)})\qquad(j=0,1;r=1,2,\cdots).$$

For sutible α, $(\theta_0^{(r)},\theta_1^{(r)})$ converges to the minimum point as r goes to the infinity.

Now, we consider the case where there are n features $x^{(1)},x^{(2)},\cdots,x^{(n)}$ in the data set.

$x^{(1)}$	$\cdots$	$x^{(n)}$	y
$x_1^{(1)}$	$\cdots$	$x_1^{(n)}$	y_1
$x_2^{(1)}$	$\cdots$	$x_2^{(n)}$	y_2
$\vdots$		$\vdots$	$\vdots$
$x_m^{(1)}$	$\cdots$	$x_m^{(n)}$	y_m

We set $x_i^{(0)} = 1$ and let

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}, \quad t = \begin{bmatrix} t_0 \\ t_1 \\ \vdots \\ t_n \end{bmatrix}, \quad x_i = \begin{bmatrix} x_i^{(0)} \\ x_i^{(1)} \\ \vdots \\ x_i^{(n)} \end{bmatrix}$$

for each $i = 1, \cdots, m$. We define $h_t(x_i)$ by

$$h_t(x_i) = g(t^\mathsf{T} x_i)$$

so that the cost function $\mathrm{cost}(h_t(x_i, y_i))$ becomes

$$\mathrm{cost}(h_t(x_i, y_i)) = -y_i \ln(h_t(x_i), y_i) - (1 - y_i) \ln(h_t(x_i, y_i))$$

for each $i = 1, \cdots, m$. In this case, we can similarly define the function $J(t_0, \cdots, t_n)$:

$$\begin{aligned} J(t_0, \cdots, t_n) &= \frac{1}{m} \sum_{i=1}^{m} \mathrm{cost}(h_t(x_i), y_i) \\ &= -\frac{1}{m} \sum_{i=1}^{m} \left[y_i \ln(h_t(x_i)) + (1 - y_i) \ln(1 - h_t(x_i)) \right]. \end{aligned}$$

The partial differentiations can be obtained similarly:

$$\frac{\partial J}{\partial t_j}(t_0, \cdots, t_n) = \frac{1}{m} \sum_{i=1}^{m} (h_t(x_i) - y_i) x_i^{(j)}$$

for each $j = 0, 1, \cdots, n$. If we want to find the minimum point of J by gradient descent, the same argument we used for a one feature case can be applied.

3. Decision Boundary

Suppose that we have m training sets and n features.

$x_1^{(1)}$	$\cdots$	$x_1^{(n)}$	y_1
$x_2^{(1)}$	$\cdots$	$x_2^{(n)}$	y_2
$\vdots$		$\vdots$	$\vdots$
$x_m^{(1)}$	$\cdots$	$x_m^{(n)}$	y_m

Suppose that we want to predict the outcome of $(x^{(1)}, \cdots, x^{(n)})$ based on logistic regression predictor of the given training data set. We set

$$x = \begin{bmatrix} 1 \\ x^{(1)} \\ \vdots \\ x^{(n)} \end{bmatrix}$$

and compute the value $h_\theta(x)$, where θ minimizes $J(t)$. If $h_\theta(x) \geq 0.5$, then we predict that the outcome of x is 1 and if $h_\theta(x) < 0.5$, then we predict that the outcome of x is 0. Since $\theta^\mathsf{T} x \geq 0$ implies $h_\theta(x) \geq 0.5$ and $\theta^\mathsf{T} x < 0$ implies $h_\theta(x) < 0.5$, we call $\theta^\mathsf{T} x = 0$ the *decision boundary.*

References

[1] Andrew Ng, Machine Learning tutorial, Stanford University
https://www.youtube.com/playlist?list=PLLssT5z_DsK-h9vYZkQkYNWcItqhlRJLN