Logistic Regression

Logistic Regression

Logistic regression is a classification algorithm used for binary classfication.

Binary classification means classification into categories such as :-

1/0 True/False Cat/Dog and so on

It gives the probability of binary outcome.

It uses sigmoid function which is an S-shaped curve used to map outputs between 0 and 1.

Model function

The model function for Logistic Regression is defined as :-

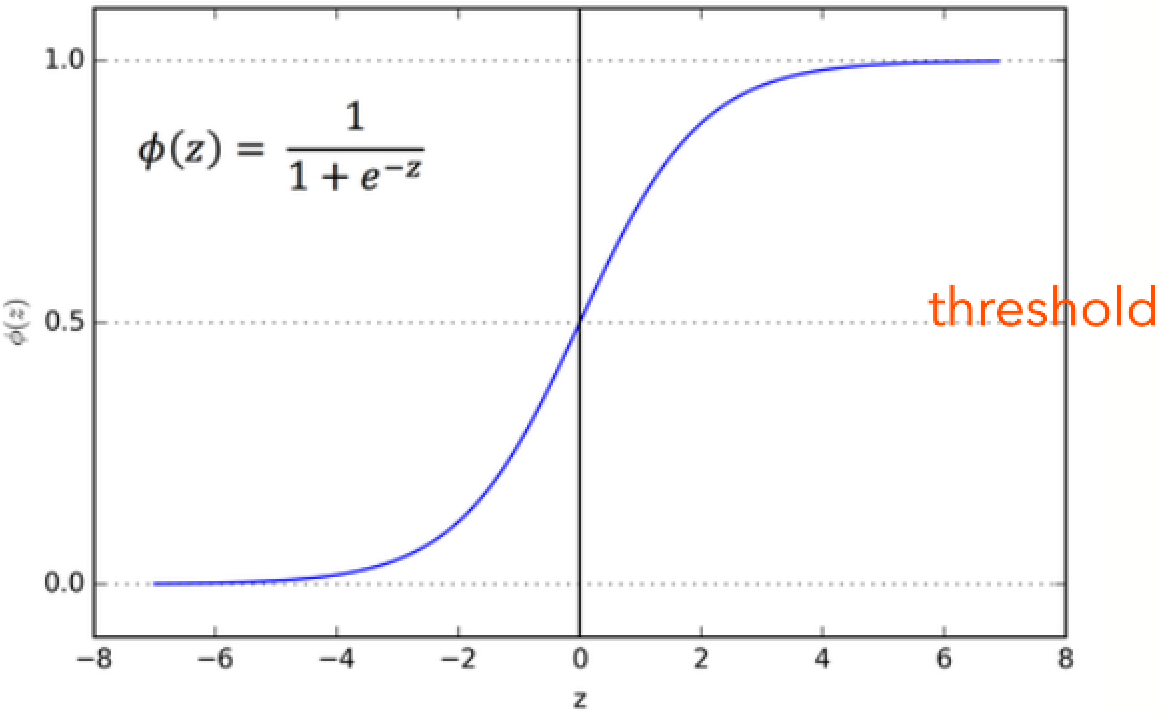
$$f_{w,b}(x) = g(z) = \frac{1}{1 + e^{-z}}$$

gives probability that class is 1

where :-
w,b are parameters of model
$z = w.x + b$

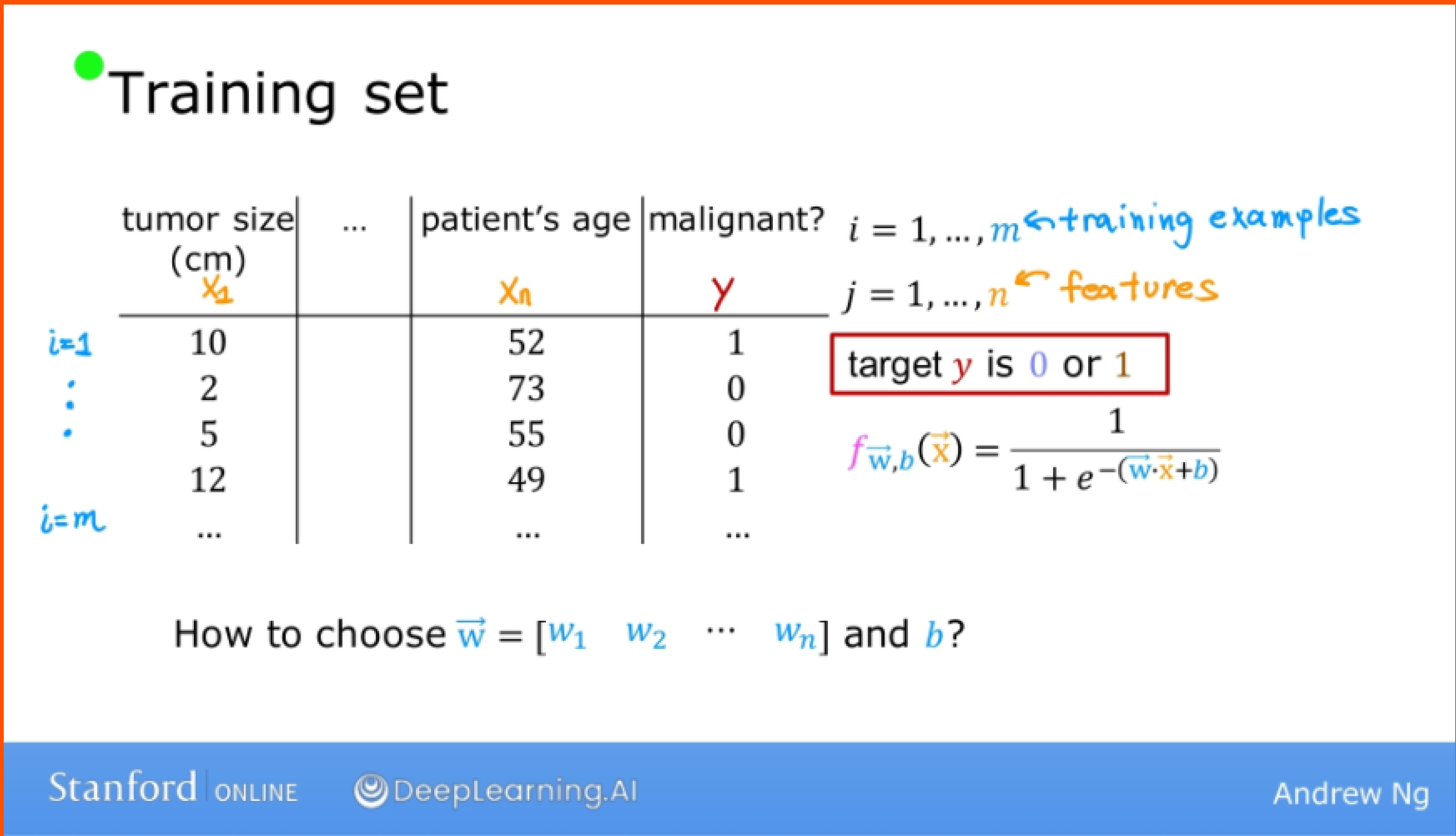
Sigmoid Fuction

It is also called as Logistic Function



outputs between 0 and 1

0 < φ(z) < 1



$x_{i,j}$ represents the value of jth feature of ith data point
example $x_{1,2} = 2$

Loss function

The loss function for one training example is

$$L(f_{w,b}(x^{(i)},y^{(i)}) = \begin{cases} -\log(f_{w,b}(x^{(i)})) & \text{if } y^{(i)}=1 \\ -(\log(1 - f_{w,b}(x^{(i)})) & \text{if } y^{(i)}=0 \end{cases}$$

Simplified loss function

ith target

$$= -y^{(i)}(\log(f_{w,b}(x^{(i)}))) - (1-y^{(i)})\log(1 - f_{w,b}(x^{(i)}))$$

model's prediction

In logistic regression, loss function is called binary cross entropy or log loss

Cost function

Cost function is the average of binary cross entropy over all training examples

$$J(w,b) = \frac{1}{m}\sum_{i=1}^{m} L(f_{w,b}(x^{(i)}), y^{(i)})$$

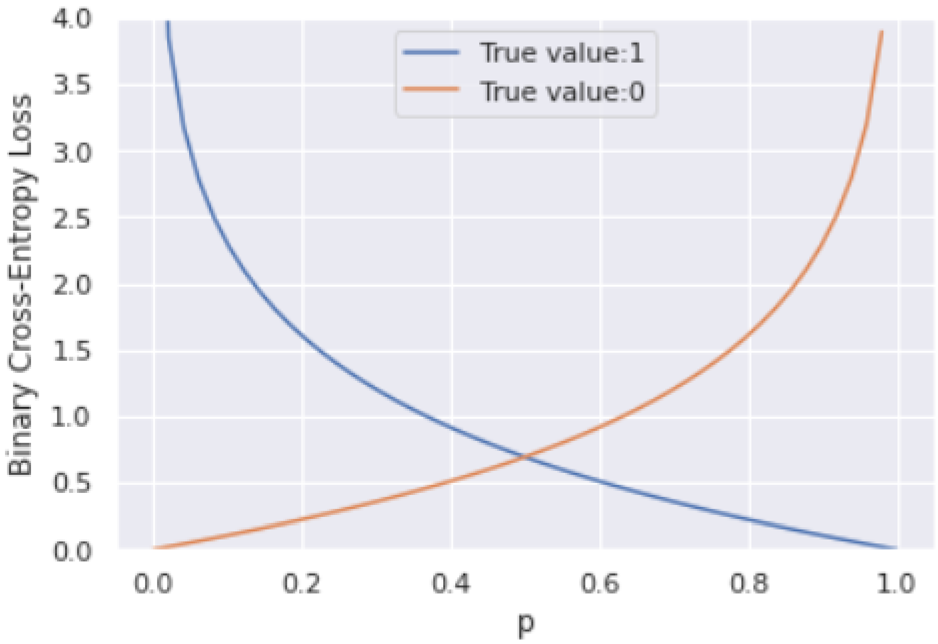
$$L(f_{w,b}(x^{(i)}), y^{(i)}) \begin{cases} -\log(f_{w,b}(x^{(i)})) & \text{if } y^{(i)} = 1 \\ -\log(1 - f_{w,b}(x^{(i)})) & \text{if } y^{(i)} = 0 \end{cases}$$

Simplified cost function

$$= \frac{-1}{m}\sum_{i=1}^{m} [y^{(i)}\log(f_{w,b}(x^{(i)})) + (1 - y^{(i)})\log(1 - f_{w,b}(x^{(i)}))]$$

$y^{(i)}$ is the ith target value
m = # of training examples

Properties of losss function



1) It is convex which means that is has single global minimum which makes gradient descent easier

2) Differentiable which means gradient descnet can be used to update parameters

$$-y^{(i)}(\log(f_{w,b}(x^{(i)}))) - (1-y^{(i)})\log(1 - f_{w,b}(x^{(i)}))$$

when y(i) is 1 this corresponds to the function log(f) i.e. True value plot and when y(i) is 0 it coresponds to the False value plot

Gradient Descent

To find the optimal parameters of the model we use the Gradient Descent algorithm.

This is similar to linear regression except that model function is defined differently.

repeat until convergence{

$$w_j = w_j - \alpha \frac{\partial J(w,b)}{\partial w_j} \qquad j=0,1...n$$

$$b = b - \alpha \frac{\partial J(w,b)}{\partial b}$$

}

where

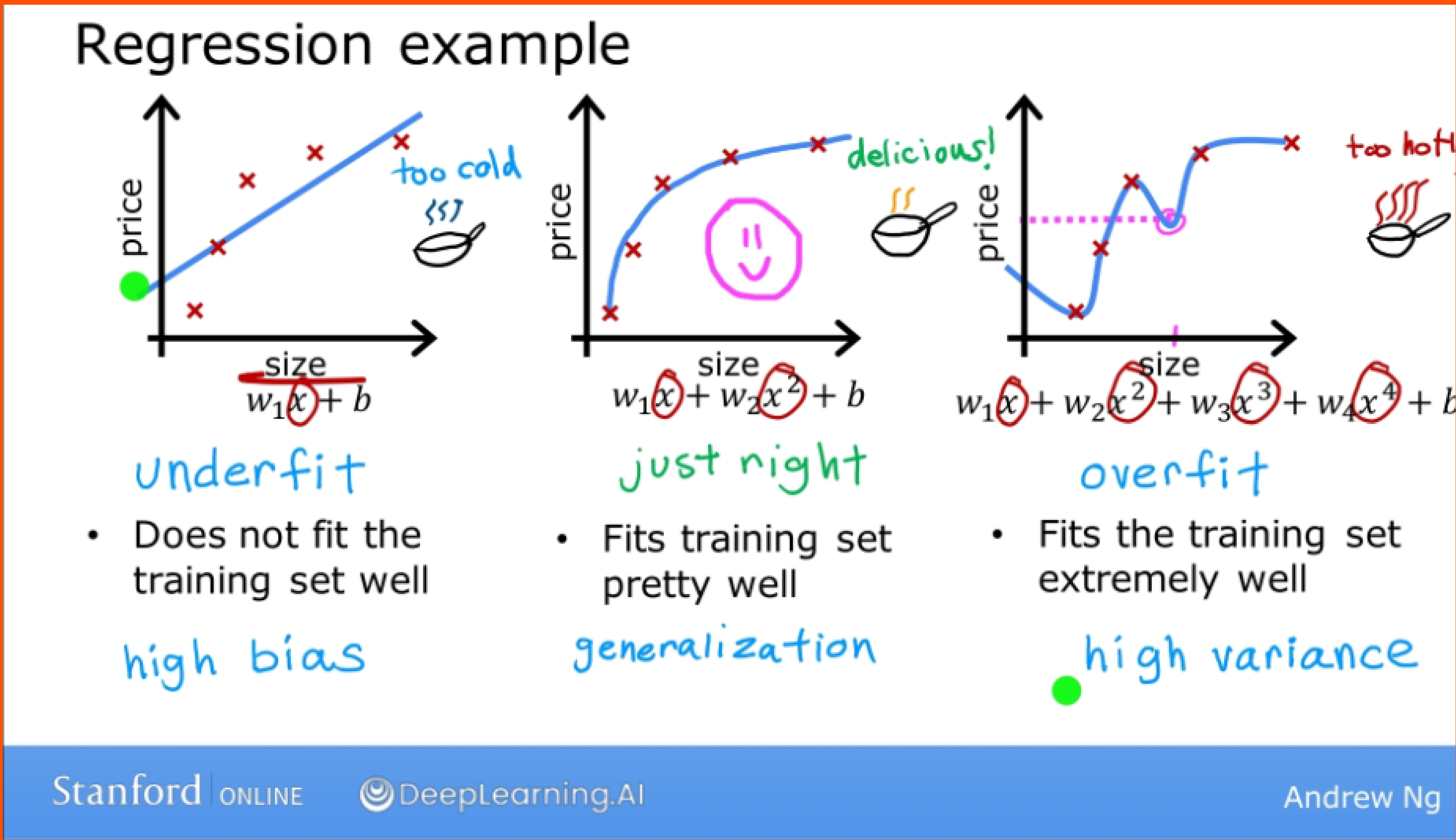
$$\frac{\partial J(w,b)}{\partial w_j} = \frac{1}{m} \sum_{i=1}^{m} (f_{w,b}(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

$$\frac{\partial J(w,b)}{\partial b} = \frac{1}{m} \sum_{i=1}^{m} (f_{w,b}(x^{(i)}) - y^{(i)})$$

α = learning rate

$x_j^{(i)}$ is the jth feature of ith data point

Overfitting



Over fitting is addressed by

1) Collecting more training data

2) Select features to include or exclude
(above is called feature selction)

3) Regularisation
(to reduce size of parameters w)

Regularised Logistic Regression

Cost function

$$= \frac{-1}{m}\sum_{i=1}^{m}[y^{(i)}\log(f_{w,b}(x^{(i)})) + (1 - y^{(i)})\log(1 - f_{w,b}(x^{(i)})) + \underbrace{\frac{\lambda}{2m}\sum_{j=1}^{n} w_j^{\,2}}_{\text{regularisation term}}$$

λ = regularisation parameter
λ > 0 (always)
w_j = weights
n = # of features

Gradient Descent

repeat until convergence{

$$w_j = w_j - \alpha \frac{\partial J(w,b)}{\partial w_j}$$

$$b = b - \alpha \frac{\partial J(w,b)}{\partial b}$$

}

where

$$\frac{\partial J(w,b)}{\partial w_j} = \frac{1}{m} \sum_{i=1}^{m} (f_{w,b}(x^{(i)}) - y^{(i)}) x_j^{(i)} + \frac{\lambda}{m} w_j$$

$$\frac{\partial J(w,b)}{\partial b} = \frac{1}{m} \sum_{i=1}^{m} (f_{w,b}(x^{(i)}) - y^{(i)})$$

α = learning rate

}

Follow me

for more

REPOST IF YOU LIKED IT