Classification Problem:

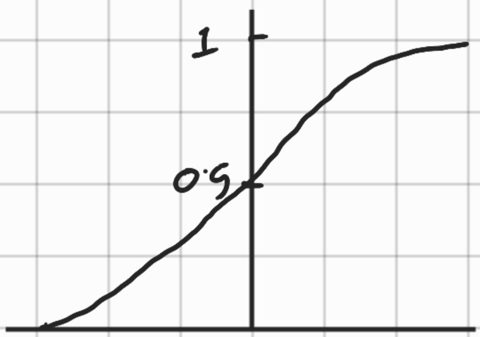
Email → Spam or Not

Online Transaction → fradulant or not.

Tumour → Malignant / Benign.

$y \in \{0, 1\}$

Logistic Regression



Logistic Regression:

To make $0 \leq h_\theta(x) \leq 1$

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

Sigmoid Function

Logistic Function

hypothesis probability:

$$h_\theta(x) = P(y=1 \mid x; \theta) = 1 - P(y=0 \mid x; \theta)$$

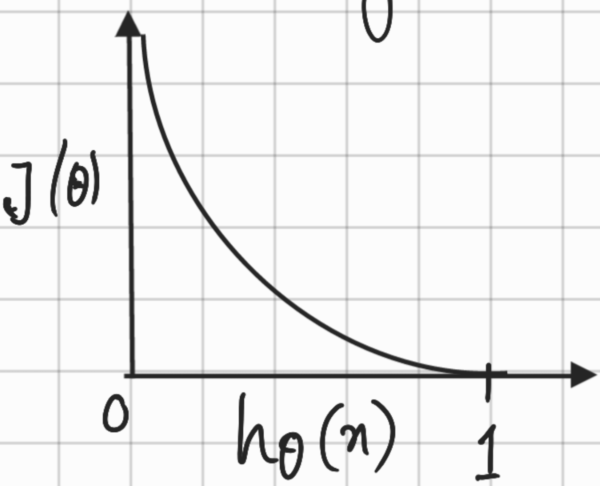
But, out exponential cost function is wavey (not convex) thus has a lot of local optimums. so to find a convex function we'll use logarithm.

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} Cost\left(h_\theta(x^{(i)}),\ y^{(i)}\right)$$

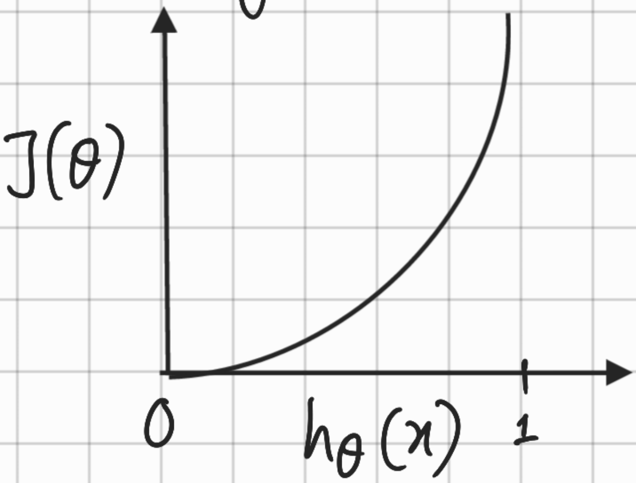
$$Cost(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{if } y=1 \\ -\log(1 - h_\theta(x)) & \text{if } y=0 \end{cases}$$

Now,

for $y=1$



$y=0$



y	$h_\theta(x)$	$Cost(h_\theta(x), y)$
$h_\theta(x)$	y	0
0	1	∞
1	0	∞

Now, the two equation can be combined into one,

$$\text{Cost}(h_\theta(x), y) = -y \log(h_\theta(x)) - (1-y) \log(1 - h_\theta(x))$$

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log\left(h_\theta(x^{(i)})\right) + (1-y) \log(1 - h_\theta(x^{(i)})) \right]$$

Vectorized implementation:

$$h = g(X\theta) \qquad g(z) = \frac{1}{1+e^{-z}}$$

$$J(\theta) = \frac{1}{m} \cdot \left(-y^T \log(h) - (1-y)^T \log(1-h) \right)$$

<u>Gradient descent:</u>

Repeat {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

}

Repeat {

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

}

$$\hookrightarrow \theta := \theta - \frac{\alpha}{m} X^T \left(g(X\theta) - \vec{y} \right)$$

implemented as,

$\theta := \frac{\alpha}{m} \times (g(X\theta) - y)^T X$

Advance Optimization :

- → Conjugate gradient
- → BFGS
- → L-BFGS

→ No need to pick manually α

→ Sometimes faster than gradient descent

```
function [jval, gradient] = costFunction(theta)
    jval = ----code to compute J(θ)---
    gradient = ----code to compute ∂J(θ)/∂θ ---
end
```

```
options = optimset('Gradobj','on','MaxIter',100);
initialTheta = zeros(2,1);
  [optTheta, functionVal, exitFlag] = fminunc
      (@costFunction, initialTheta, options);
```

Multiclass Classification:

→ Email Foldering/tagging: work, Fiends, Family,
$y=1$ $y=2$ $y=3$

→ Medical diagrams: Not ill, Cold, Flu
$y=1$ 2 3

→ Weather: Sunny, Cloudy, Rain, Snow
$y=1$ 2 3 4

Solving method: One vs all.

$$y=\{0, 1, \ldots n\}$$

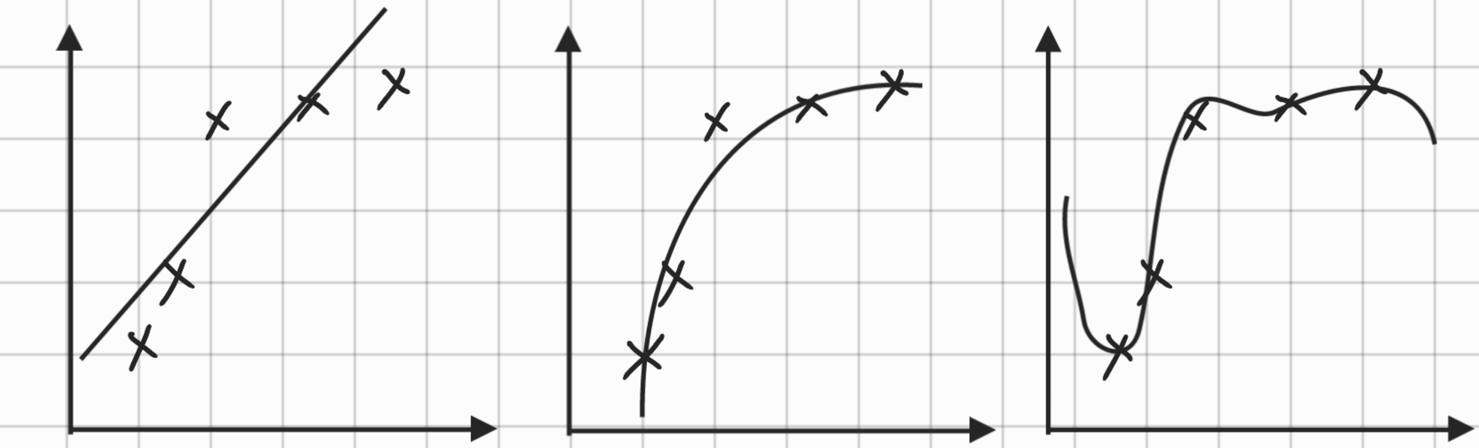
$$h_\theta^0(x) = P(y=0|x;\theta)$$

$$\vdots$$

$$h^n(x) = P(y=n|x;\theta)$$

$$\text{prediction} = \max_i \left(h_\theta^i(x)\right)$$

Overfitting:



→ Underfit → Just right → Overfit

→ High bias → High Variance

Overfitting : The curve would pass through the all datapoints of the training set but may not predict accurately when a new example needs to be predicted.

"Not A Good Predictor"

Soln:

→ Reduce the number of features

→ Regularization

Regularization:

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta (x^{(i)}) - y^{(i)}\right)^2 + \lambda \sum_{j=1}^{n} \theta_j^2$$

$\lambda \rightarrow$ regularization parameter

↳ becomes extremly high $\approx 10^{10}$, then underfit

↳ becomes extremly small ≈ 0, no regularization

↓ again overfitting

Gradient descent

Repeat {

$$\theta_0 = \theta_0 - \frac{\alpha}{m} \sum_{i=1}^{m} \left(h_\theta (x^{(i)}) - y^{(i)}\right) x_0^{(i)}$$

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

}

↳ Always less than 1

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

Normal Equation

$$\theta = \left(X^T X + \lambda . L\right)^{-1} X^T y$$

$$L = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & & & 0 \\ 0 & & 1 & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}_{(n+1) \times (n+1)}$$

For Logistic Regression:

$$J(\theta) = -\left[\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)}))\right] + \frac{\lambda}{2m}\sum_{j=1}^{n} \theta_j^2$$