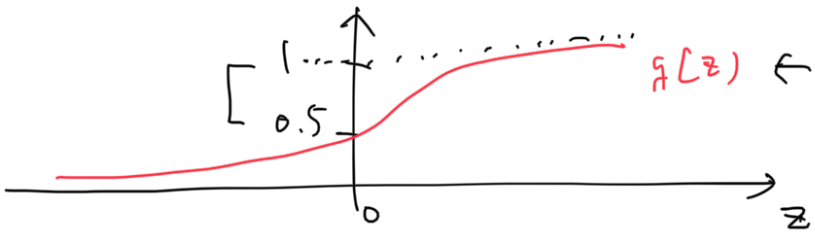
ML w3 logistic regression

I

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

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

$\rightarrow \quad g(z) = \dfrac{1}{1 + e^{-z}}$



$g(z)$ ← sigmoid function

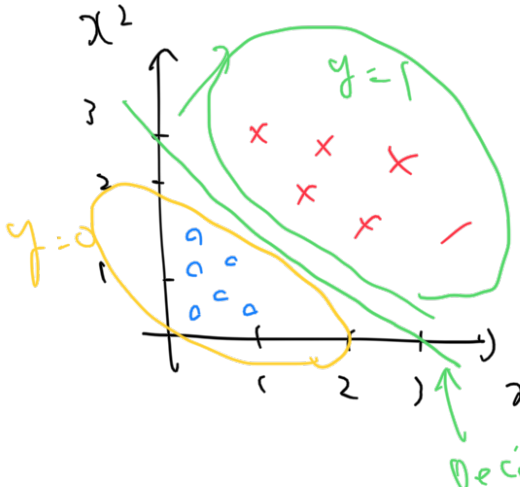
$h_\theta(x) = 0.7$ .. Tell patient that 70% chance of tumor being malignant.

Decision Boundary

$P(y=0|x;\theta) + P(y=1|x;\theta) = 1$

$y=1$ if $h_\theta(x) \geq 0.5$ (V)

$y=0$ if $h_\theta(x) < 0.5$



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

$\rightarrow h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$

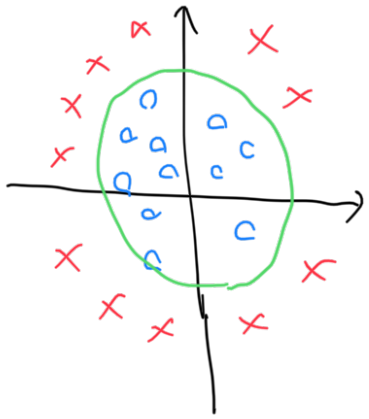
$-3 \quad 1 \quad 1$

Predict $y=1$ if $-3 + x_1 + x_2 \geq 0$

$x_1 + x_2 = 3$

(v) $h_\theta(x) = 0.5$

Non linear Decision Boundary



$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$$

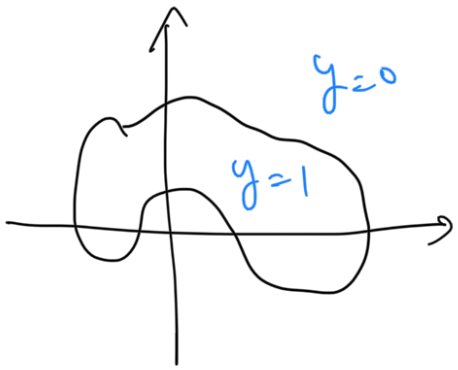
$\theta_0 = -1$, $\theta_1 = 0$, $\theta_2 = 0$, $\theta_3 = 1$, $\theta_4 = 1$

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

Predict $y = 1$ if $-1 + x_1^2 + x_2^2 \geq 0$

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

$$\boxed{x_1^2 + x_2^2 = 1}$$



$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_1^2 x_2 + \theta_5 x_1^2 x_2^2 + \theta_6 x_1^3 x_2 + \dots)$$

Cost Function

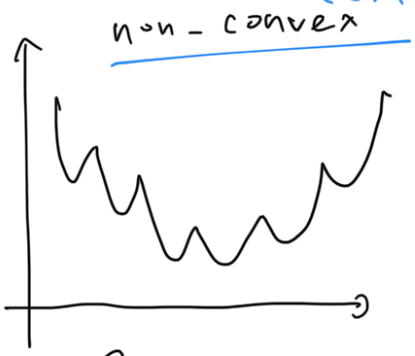
- Training Set: $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)})\}$
- m examples $x \in \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix} \mathbb{R}^{n+1}$ $x_0 = 1$, $y \in \{0, 1\}$

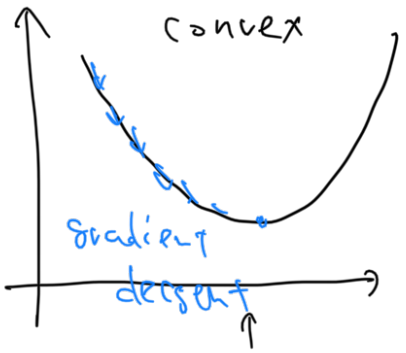
$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

(Linear Regression model)

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

$\text{Cost}(h_\theta(x^{(i)}), y^{(i)})$ ← squared error





(Logistic Regression)

$$\text{Cost}(h_\theta(x), y^{(i)})$$

$$\uparrow$$

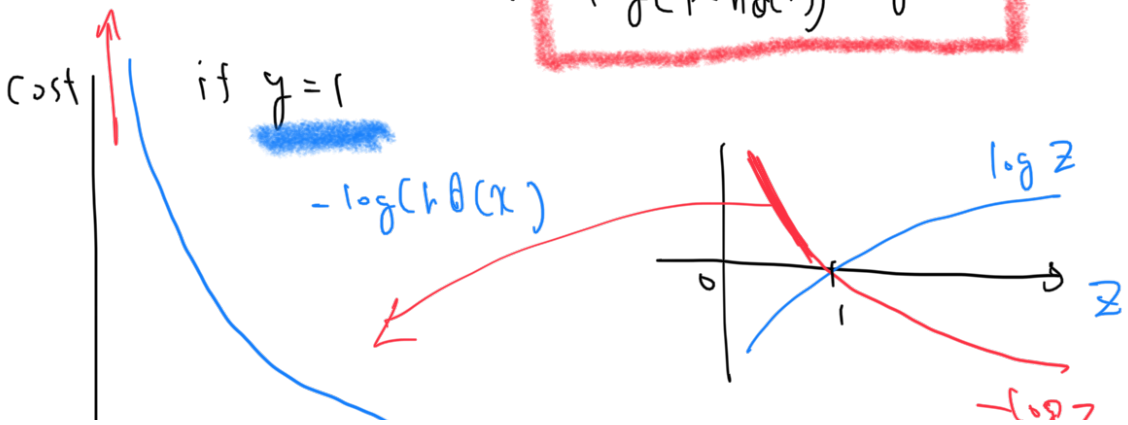
$$\frac{1}{1 + e^{-\theta^T x}}$$

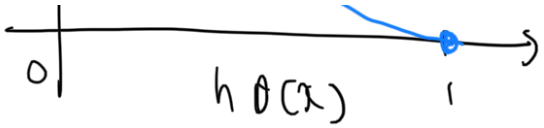
⇒ non-convex function
非凸関数
非線形，
局所最適.

⇕

convex, gradient decent で使用するモデルをつくる

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



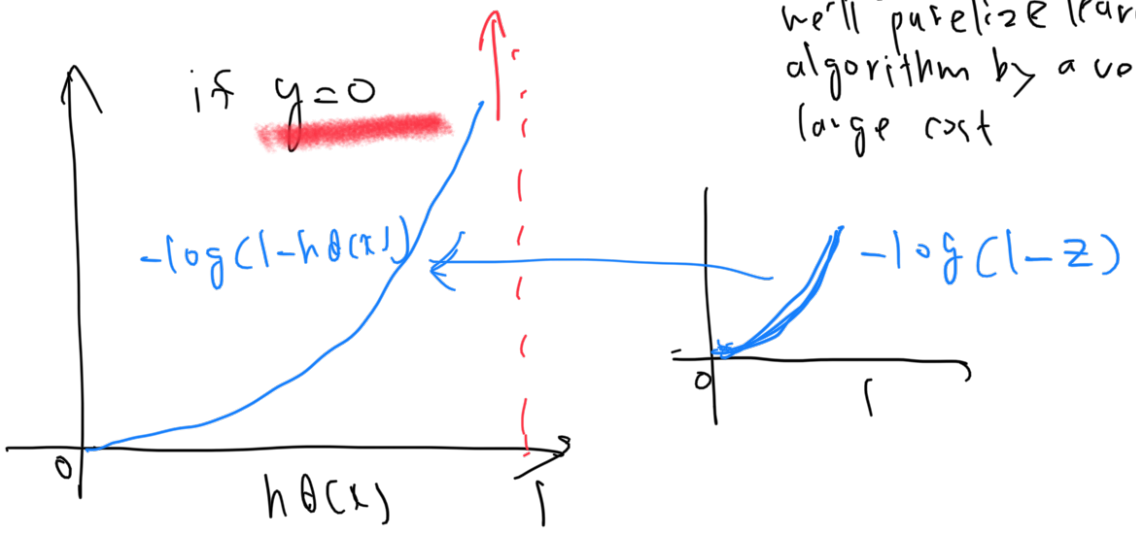


cost = 0 if $y=1$, $h_\theta(x)=1$

But as $h_\theta(x) \to 0$

cost $\to \infty$

if $h(\theta) \to 0$,
(predict $P(y=1|x;\theta)=0$)
but $y=1$,
we'll penalize learning algorithm by a very large cost



Simplified Cost Function and Gradient Decent

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

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

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

If $y=0$: $\text{Cost}(h_\theta(x), y) = -\log(1-h_\theta(x))$

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

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

Minimize

$\min_\theta J(\theta)$

$h_\theta(x) = \theta^T x$ ← Linear Regression

Repeat {

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

simultaneously update

}

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

↳ Logistic Regression

Vectorized Implementation

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

Advanced Optimization

Gradient Decent General form

Repeat {

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

}

[Conjugate cl BFGS ...] ⟷

pros

No need to manually pick α

Example

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

→ $J(\theta) = (\theta_1 - 5)^2 + (\theta - 5)^2$

```
function [jVal, gradient]
    = costFunction(theta)
jVal = (theta(1) - 5)^2
...
```

$$\frac{\partial}{\partial \theta_1} J(\theta) = 2(\theta_1 - 5)$$

$$\frac{\partial}{\partial \theta_2} J(\theta) = 2(\theta_2 - 5)$$

```
(theta(1) - 5)^2 .
gradient = zeros(2, 1)
gradient(1) = 2*(theta(1) - 5);
gradient(2) = 2*(theta(2) - 5);
```

... advanced: ex: fminunc(@costFunction,
initial Theta,
options).

$$\text{theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} \quad \begin{matrix} \text{theta}(1) \\ \text{theta}(2) \\ \vdots \\ \text{theta}(n) \end{matrix}$$

function [jVal, gradient] = costFunction(theta).

gradient(1) = (1)

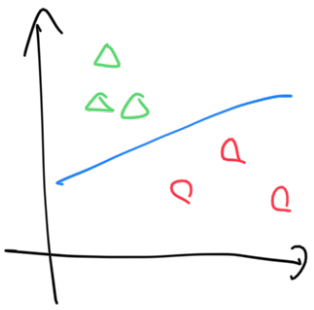
:

= (n+1)

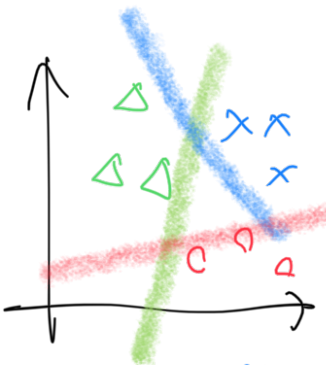
Multiclass Classification — One-vs-all

e.g. Sunny, Cloudy, Rainy ...

$y = 1 \quad 2 \quad 3$



Binary



Multi Class

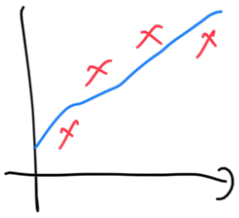
$$h_\theta^{(i)}(x) = P(y = i \mid x; \theta)$$

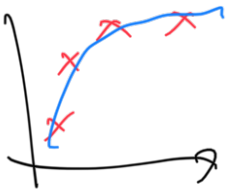
Train a logistic regression classifier $h_\theta^{(i)}(x)$ for each class i to predict the probability that $y = i$

On a new input x, to make a prediction, pick the class i that maximizes

$$\max_i h_\theta^{(i)}(x)$$

Regularization — solving the problem of overfitting.





overfitting

Options

1. Reduce number of features
2. Regularization

$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

penalize $\theta_3 \to \theta_4$ because x^3 and x^4 is very small.

Cost Function

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

Regularization parameter ← set not too large value.

$$h_\theta(x) = \theta_0$$

Regularized Linear Regression

Gradient descent

Repeat {

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

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

}

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

0.99...

$$1 - \alpha \frac{\lambda}{m} < 1$$

Normal Equation

$$X = \begin{bmatrix} (x^{(1)})^T \\ \vdots \\ (x^{(m)})^T \end{bmatrix} \qquad y = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(m)} \end{bmatrix}$$

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

$$\rightarrow \theta = \left(X^T X + \lambda \begin{bmatrix} 0 & & & \\ & 1 & & \\ & & 1 & \\ & & & \ddots \end{bmatrix}\right)^{-1} X^T y$$

$(n+1)$ by $(n+1)$ matrix

Non-invertibility (optional / advanced)

Suppose $(m \leq n)$

$$\theta = (X^T X)^{-1} X^T y \qquad \text{pinv}$$

if $\lambda > 0$,

$$\theta = \left(X^T X + \lambda \begin{bmatrix} 0 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix} \right)^{-1} X^T y$$

↑ non invertible

Regularized logistic Regression

- Cost function (see above)
- Gradient Descent
- Advanced Optimization.