Week 1 1/12

→ linear regression, cont...

→ gradient descent

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

to define notion of convergence

to ensure for future x_i by the gap angle $< \epsilon$ θ

to an example by moving...

x_i or $\frac{x_i - \mu_i}{\sigma_i}$

to $\text{lr} \leftarrow \begin{cases} \text{lr} \cdot 2 & \text{if } \cos\theta > 0 \\ \text{lr} \cdot .5 & \text{if } \cos\theta < 0 \end{cases}$

to changing gradient descent

right $J(\theta)$ vs θ shape

low cost

cross product

→ Fitting w/ Polynomial Regression

to can boost our fitting w/ number of newly data

to can fit additional kinds of functions than just linear fitting

by adding more features, thus using the same machinery!

to e.g., $x_i' = (x_i)^2$

→ Normal Equation for Linear Regression

for $h_\theta(x) = \theta_0 + \theta_1 x_1 + \dots + \theta_n x_n$

∴ output $y^{(i)}$

the

$$\begin{bmatrix} x_1^{(1)} & \dots & x_n^{(1)} \\ x_1^{(2)} & \dots & x_n^{(2)} \\ \vdots & & \vdots \\ x_1^{(m)} & \dots & x_n^{(m)} \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix}$$

$\Rightarrow$ $X\theta = \vec{y}$

$X^T X \theta = X^T \vec{y}$

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

$\theta = \boxed{(X^T X)^{-1} X^T y}$ → and to be sure this is invertible, we need to:

to guarantee linear linearly dependent features

to $(\# \text{ features}) \geq (\# \text{ data points})$

to use other feature fitting or use regularization

to when do you use gradient descent vs normal method?

to computing $\boxed{(X^T X)^{-1}}$ is expensive → runtime scales $O(n^3)$

to use adaptive based on size of n →

$n \leq 10^4$ → normal equation

$n \geq 10^6$ → gradient descent

programming - week 2

$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta)$

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

$h_\theta(x) = \theta_0^{(i)} + \theta_1^{(i)} x$

$\frac{\partial}{\partial \theta_0} J(\theta) = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})$

$\frac{\partial}{\partial \theta_1} J(\theta) = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x^{(i)}$

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

$\theta_1 := \theta_1 - \frac{\alpha}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x^{(i)}$

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

Week 3

→ neural nets

to activation units = nodes, i.e. "bits", more or less

to each node has $\sigma(\sum \text{(weight} \cdot \text{(previous output))})$

to σ is activation function

usually logistic, $\sigma(z) = \frac{1}{1+e^{-z}}$

to see from logic gates w/ neural nets!

→ classification ∴ decision boundary

to want distinct outputs

to have continuous outputs

to also, put the line through this

→ logistic regression

to goal for nonlinear classification boundary

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

to $\sigma(z) = h_\theta(x) = P(y=1)$ w/ $\sigma(z) = \frac{1}{1+e^{-z}}$

to cost function?

to want just that squaring, since that is convex

to this will make gradient descent still work

to instead: $\text{cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & y=1 \\ -\log(1-h_\theta(x)) & y=0 \end{cases}$

to $\text{lr} \leftarrow \frac{\text{lr}}{2}$

if $y=1$, positive probability 0

to $\sigma(z) = \frac{1}{1+e^{-z}}$, taking derivative

find $\sigma'(z) = \sigma(z)(1-\sigma(z))$

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

to for our gradient descent...

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

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

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

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

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

$\Rightarrow \frac{\partial}{\partial \theta_j} h_\theta(x^{(i)}) = \frac{\partial}{\partial \theta_j} \left(\frac{1}{1+e^{-\theta^T x^{(i)}}} \right)$

w/ chain rule: $\frac{\partial}{\partial \theta_j} \frac{1}{1+e^{-\theta^T x^{(i)}}} = \frac{\partial}{\partial \theta_j} \frac{1}{1+e^{-z}}$

$= -\left(\frac{1}{1+e^{-\theta^T x^{(i)}}} \right)^2 \left(-e^{-\theta^T x^{(i)}} \right) \cdot \frac{\partial}{\partial \theta_j} \left(-\theta^T x^{(i)} \right)$

$= \frac{e^{-\theta^T x^{(i)}}}{(1+e^{-\theta^T x^{(i)}})^2} \cdot x_j^{(i)}$

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

w/ chain rule: $\frac{\partial}{\partial \theta_j} \log(h_\theta(x^{(i)})) = \frac{1}{h_\theta(x^{(i)})} \cdot \frac{\partial}{\partial \theta_j} h_\theta(x^{(i)})$

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

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

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

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

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

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