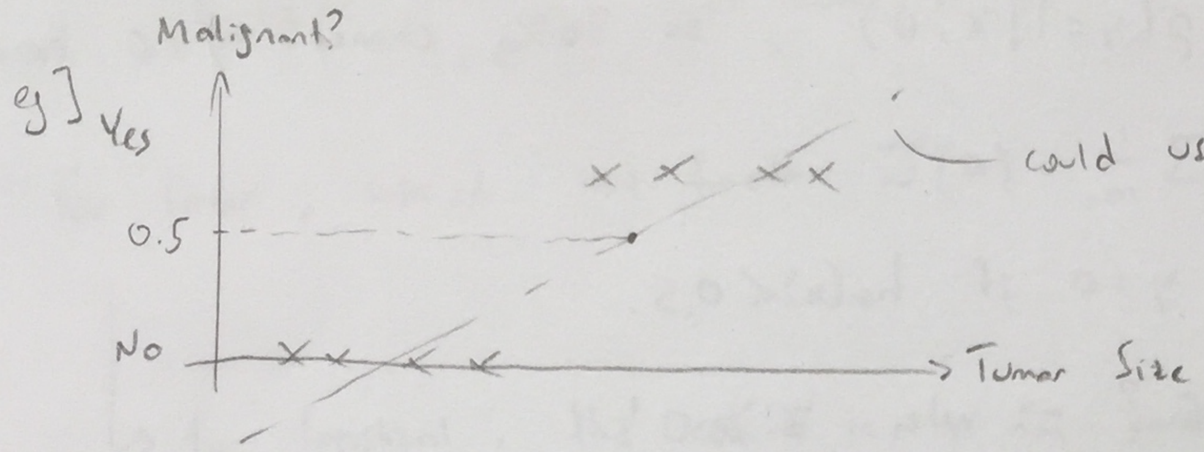
ML WEEK 3: LOGISTIC REGRESSION

LEO ROBINOVITCH

- Classification Problems => spam / not spam? Fraud / Not Fraud?

  ↳ $y \in \{0, 1\}$ → 0: negative class (eg. benign tumor)
  1: positive class (eg. malignant tumor)

  MORE CLASSES POSSIBLE



eg.] — could use regression technique, fit some $h_\theta(x)$!

W/ regression, say $h_\theta(x) \geq 0.5 \rightarrow y = 1$
$h_\theta(x) < 0.5 \rightarrow y = 0$

OFTEN BAD IDEA!

=> $h_\theta(x)$ not great predictor all the time.

↳ $h_\theta(x) > 1$ & $h_\theta(x) < 0$, too...

Logistic Regression: Hypothesis Representation:

=> previously, $h_\theta(x) = \theta^T x$ where $x = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}$ ($x_0 = 1$) $\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix}$, both $n+1 \times 1$ matrices where n is # features

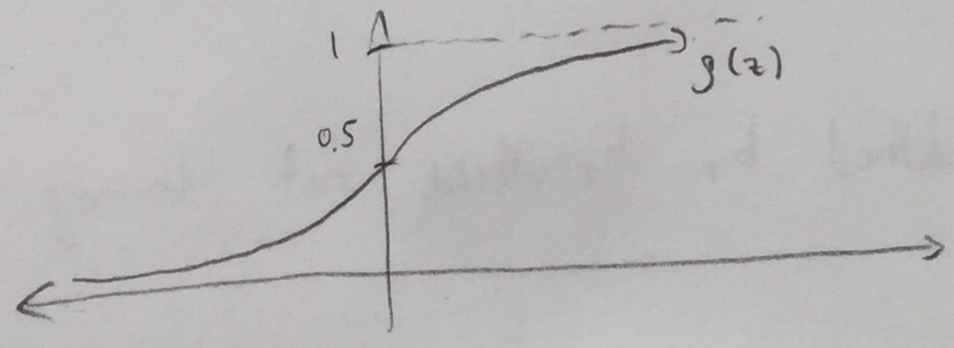
↳ in linear regression.

=> Logistic Regression → SIGMOID FUNCTION:

$$g(z) = \frac{1}{1 + e^{-z}} \quad \text{where} \quad h_\theta(x) = g(\theta^T x)$$

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

$g(z)$ sigmoid or logistic fnt.



Interpretation of $h_\theta(x)$ → estimated probability that $y=1$ on input x.

↳ eg. $x = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ \text{tumor size} \end{bmatrix} \longrightarrow h_\theta(x) = 0.7$, 70% chance tumor malignant

↳ $h_\theta(x) = p(y=1 \mid x; \theta)$ "probability that $y=1$, given x, parameterized by θ"

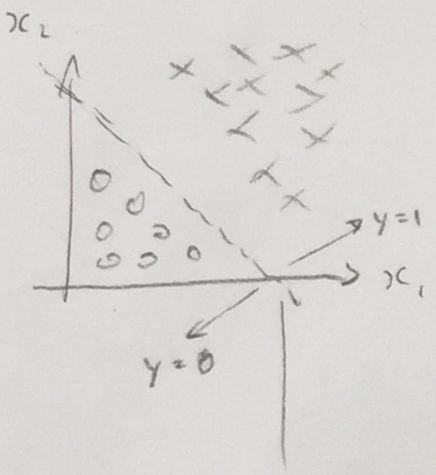
* note that $p(y=0 \mid x;\theta) = 1 - p(y=1 \mid x;\theta)$, so 30% chance $y=0$ here

Decision Boundary:

Suppose $y=1$ if $h_\theta(x) \geq 0.5$, $y=0$ if $h_\theta(x) < 0.5$.

↳ when will $h_\theta(x)$ be > 0.5? → when $z \geq 0$!

$z \geq 0 \rightarrow g(\theta^T x) \geq 0.5$ when $\underline{\theta^T x \geq 0}$!

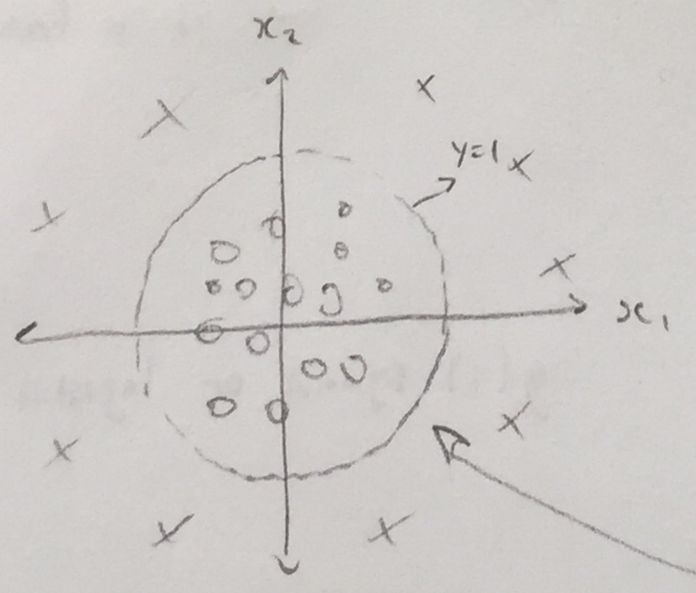
eg) 

$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$ where $\theta = [-3, 1, 1]$ (found)

↳ predict $y=1$ if $\underbrace{-3 + x_1 + x_2}_{\theta^T x} \geq 0$!

$\longrightarrow x_1 + x_2 \geq 3$

Non-Linear Decision Boundaries:



$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 \underbrace{x_1^2}_{x_3 = x_1^2} + \theta_4 \underbrace{x_2^2}_{x_4 = x_2^2})$

found $\theta = [-1, 0, 0, 1, 1]$

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

→ decision boundary is defined by hypothesis, not training set!

Function -> getting $\underline{\theta}$

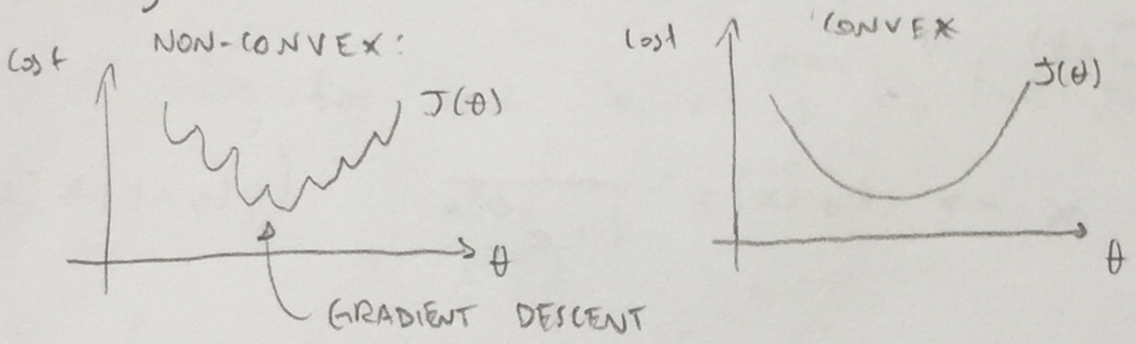
ex] Training set (m-examples) $\rightarrow \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}) \dots (x^{(m)}, y^{(m)})\}$

$x = \begin{bmatrix} x_0 \leftarrow =1 \\ x_1 \\ \vdots \\ x_n \end{bmatrix} \in \mathbb{R}^{n+1}$ where n # features, $y \in \{0,1\}$

$h_\theta(x) = \frac{1}{1+e^{-\theta^T x}}$ -> how to set $\underline{\theta}$?

=> for linear, used cost function $J(\theta) = \frac{1}{m} \Sigma \underbrace{\frac{1}{2}(h_\theta(x)^{(i)} - y^{(i)})^2}_{\text{cost}(h_\theta(x)^{(i)} - y^{(i)})}$

↳ for logistical, this results in a non-convex result



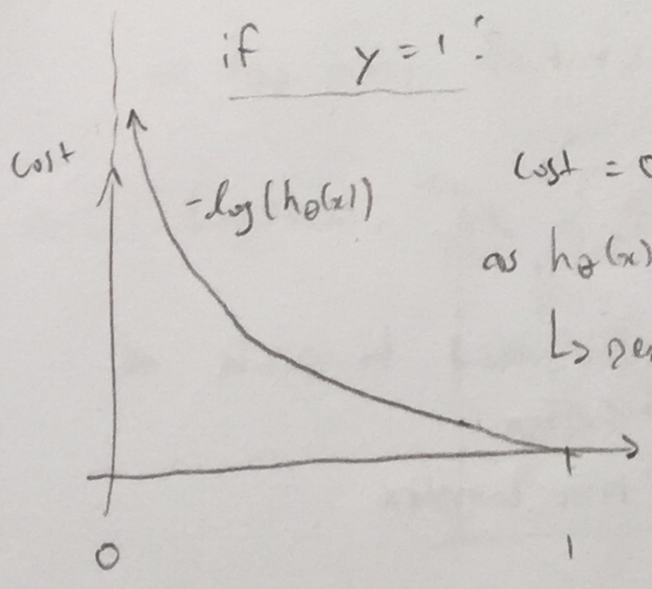
GLOBAL MIN NOT GUARANTEED!

=> b/c sigmoid $\frac{1}{1+e^{-z}}$ is nonlinear -> non-convex

=> COST FUNCTION FOR LOGISTICAL REG:

$$\text{cost}(h_\theta(x), y) = \begin{cases} -\log h_\theta(x) & \text{if } y=1 \\ -\log(1-h_\theta(x)) & \text{if } y=0 \end{cases} \Big\} \text{GIVES CONVEX COST FNT.}$$

if $y=1$:



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

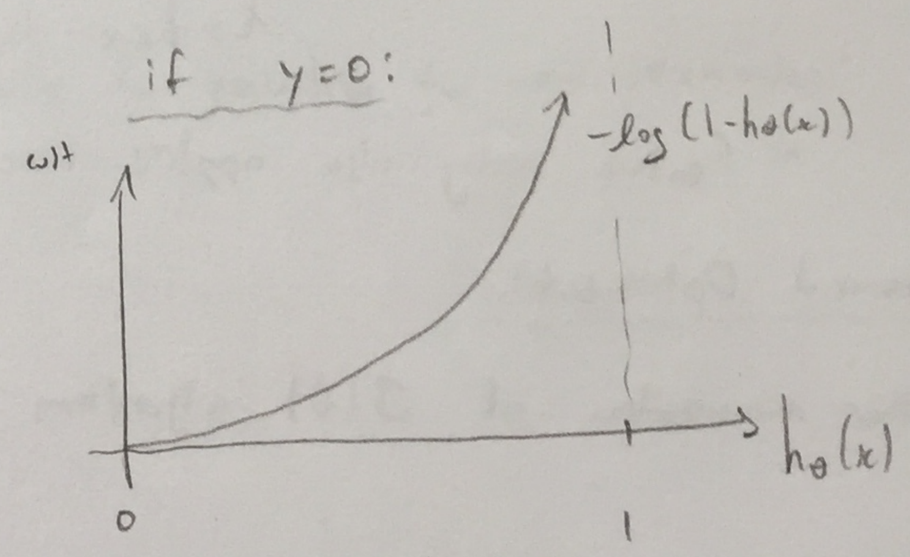
as $h_\theta(x) \rightarrow 0$, cost $\rightarrow \infty$

↳ penalize incorrect classification by a lot!

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

very unlikely

if $y=0$:



cost = 0 if $h_\theta(x) = 0$

cost v. high if $h_\theta(x) = 1$

↳ $p(y=1|x;\theta) = 0$

very unlikely

Simplified Cost Function & Gradient Descent

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} \text{cost}(h_\theta(x)^{(i)}, y^{(i)}) \quad \text{where} \quad \text{cost} = \begin{cases} -\log(h_\theta(x)) & \text{if } y=1 \\ -\log(1-h_\theta(x)) & \text{if } y=0 \end{cases}$$

Note: $y=0$ or 1 always.

↳ RE-WRITE: $\text{cost}(h_\theta(x), y) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x))$ — b/c y always 0 or 1.

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

↳ convex & derived from stats (principal of max likel. est.)

↳ find θ that minimizes $J(\theta)$

⇒ to make prediction given new x → $h_\theta(x) = \frac{1}{1+e^{-\theta^T x}}$ $\quad (p(y=1|x;\theta))$

Gradient Descent:

repeat $\theta_j := \theta_j - \alpha \frac{\partial}{\partial\theta_j} J(\theta)$ — simultaneous update!

$$\Rightarrow \frac{\partial}{\partial\theta_j} J(\theta) = \sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)} \quad \text{as before}$$

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

* Feature scaling also applies here.

Advanced Optimization

Other minimization of $J(\theta)$ algorithms:
- Conjugate Gradient
- BFGS
- L-BFGS

(· No need to pick α · Faster · More complex)

ex.) $\theta = \begin{bmatrix}\theta_1\\ \theta_2\end{bmatrix}$ $\quad J(\theta) = (\theta_1-5)^2 + (\theta_2-5)^2$

$\frac{\partial}{\partial\theta_1} J(\theta) = 2(\theta_1-5)$ $\quad \frac{\partial}{\partial\theta_2} J(\theta) = 2(\theta_2-5)$

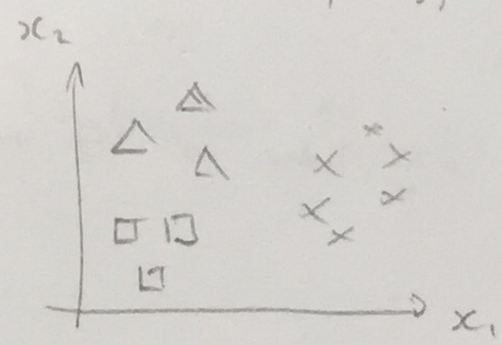
↳ see video for Octave implementation

multiclass Classification:

eg) email → work (1), friends (2), family (3)

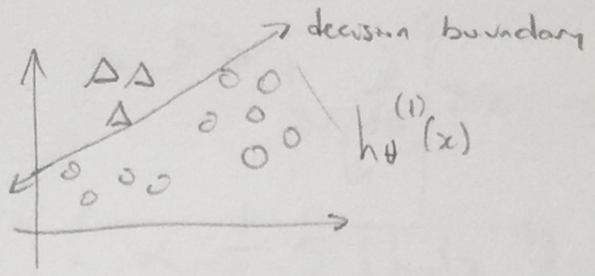
not ill, cold, flu

sunny, rainy, cloudy, snow etc.

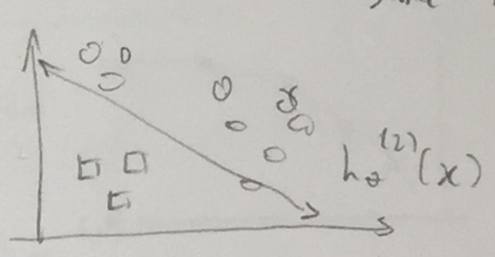


⇒ One-vs-all (one vs rest)

Class 1) △:



⇒



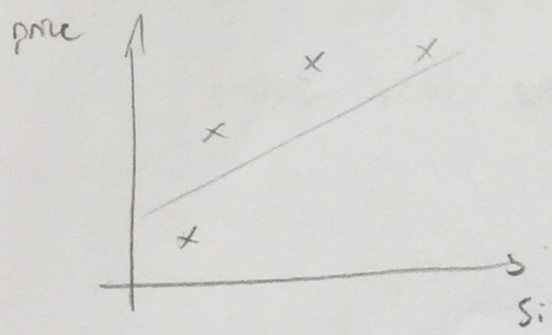
Same for x

$\Rightarrow h_\theta^{(i)}(x) = p(y=i \mid x;\theta)$, $i=1,2,3$

Don't add to 1 necessarily

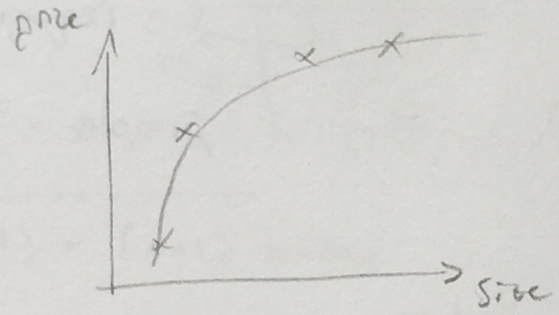
↳ for new x, $\max_i h_\theta^{(i)}(x)$ gives prediction (eg 60%, 5%, 5%)

Overfitting:



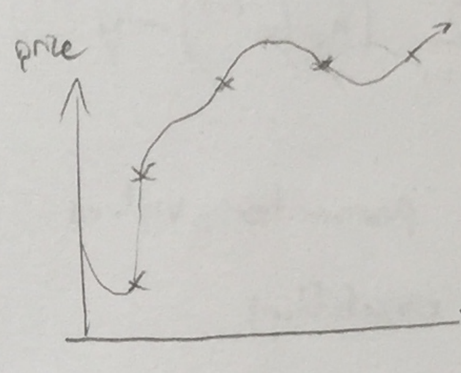
$\theta_0 + \theta_1 x$

UNDERFIT (high bias)



$\theta_0 + \theta_1 x + \theta_2 x^2$

Pretty Good!

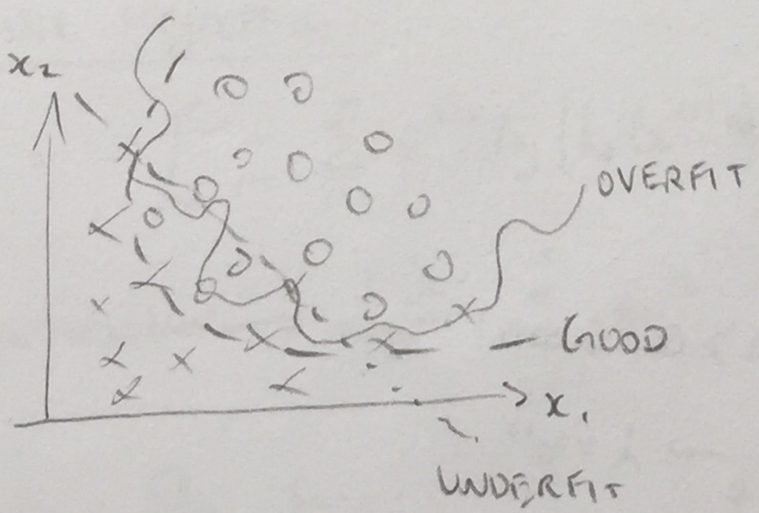


$\theta_0 + \theta_1 x^1 + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

OVERFIT (high variance)

⇒ too many features → cost ~ 0, but fails to generalize to new examples!

ex2: Logistical:



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

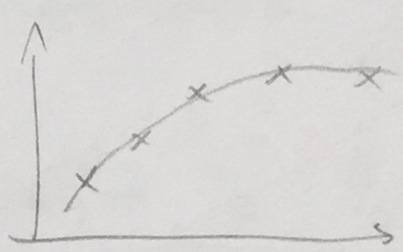
=> we may just have a lot of features -> if we have too many & not a lot of training data, may overfit!

↳ SOLTNS: -> reduce # features

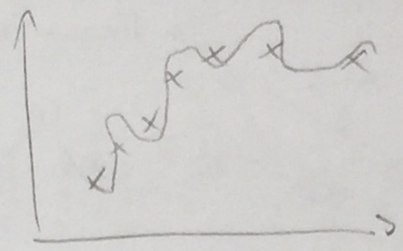
-> model select which features to keep / throw out

-> Regularization: keep all features, but change magnitudes of θ vals!

Regularization + Cost Function:



$\theta_0 + \theta_1 x + \theta_2 x^2$ (Quadratic)



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

↳ penalize θ_3 & θ_4, make v. small!

- $\min_\theta \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2 + \underbrace{1000\,\theta_3^2 + 1000\,\theta_4^2}$ require θ_3 & θ_4 to be small!

↳ small parameter values $\theta_0, \theta_1 \dots \theta_n$, simpler hypothesis less prone to overfitting

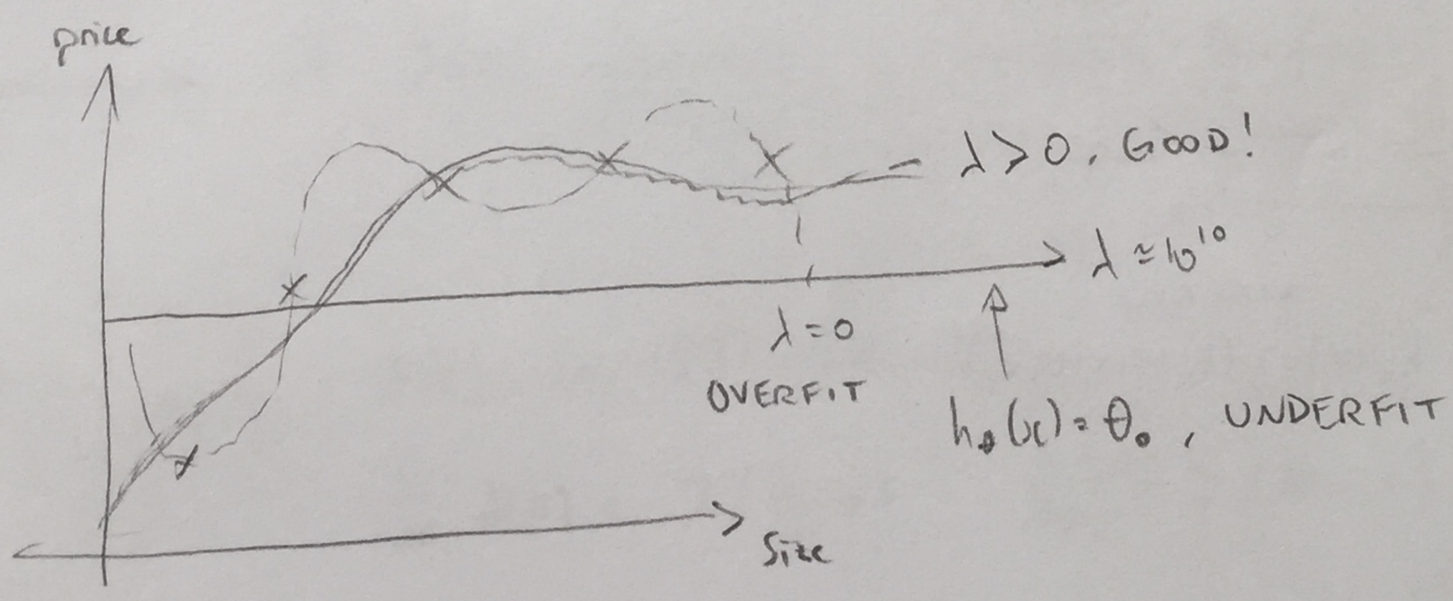
eg] Housing: $x_1, x_2, \dots x_{100}$ # Features

$\theta_0, \theta_1, \theta_2 \dots \theta_{100}$ Parameters

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

↳ not penalizing θ_0 by convention

↳ λ: regularization parameter



...rized Linear Regression:

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

↳ Gradient Descent, $\theta_0 := \theta_0 - \alpha\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_0^{(i)}$ ← don't penalize θ_0

$$\theta_j := \theta_j - \alpha\underbrace{\left[\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)} + \frac{\lambda}{m}\theta_j\right]}_{\frac{\partial}{\partial\theta_j}J(\theta)\text{ regularized}} \quad j = 1, 2, \dots n$$

$$\hookrightarrow \theta_j := \theta_j\underbrace{(1 - \alpha\frac{\lambda}{m})}_{1 - \alpha\frac{\lambda}{m} < 1} - \alpha\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}$$

Normal Equation for Regularization:

$$\theta = \left(X^TX + \lambda\underbrace{\begin{bmatrix}0 & & & \\ & 1 & & \\ & & 1 & \\ & & & \ddots \\ & & & & 1\end{bmatrix}}_{(n+1)\times(n+1)\text{ matrix}}\right)^{-1}X^Ty$$

recall if $m \leq n \rightarrow X^TX$ non-invertible (degenerate, singular)

↳ luckily, if $\lambda > 0$, $X^TX + \lambda\left[\dots\right]$ → always invertible!

Regularized Logistic Regression:

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

regularization → $+\frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$ ← $\theta_1, \theta_2 \dots \theta_n$

Gradient Descent: θ_0 same

$$\theta_j := \theta_j - \alpha\underbrace{\left[\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)} + \frac{\lambda}{m}\theta_j\right]}_{\frac{\partial}{\partial\theta_j}J(\theta)}, \quad j = 1, 2 \dots n$$

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