Eligiendo un modelo de Machine Learning

¿Y ahora qué?

Depurando un algoritmo de aprendizaje:
Supongamos que usted ha implementado un programa de regresión lineal regularizada para predecir el precio de las casas.

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

Sin embargo, cuando aplica testea su hipótesis a nuevas casas, resulta que encuentra errores demasiado grandes. ¿que debería hacer en tal caso

- Obtener mas ejemplos de entrenamiento
- Usar un set menor de features
- Usar features adicionales
- Adicionar features polinomiales $(x_1^2, x_2^2, x_1x_2, \text{etc.})$
- Intentar decrementar λ
- Intentar incrementar λ

El chequeo en el aprendizaje de maquina:

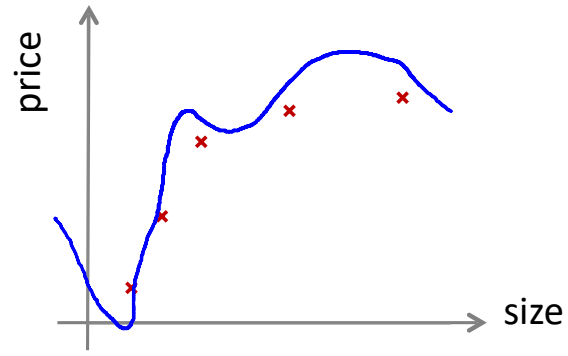
Diagnóstico: Una guía para saber si algo esta trabajando (o no) en su algoritmo de aprendizaje es ver su performance.

Un diagnostico puede tomar tiempo para implementar, pero puede ser un buen uso del tiempo.

Eligiendo un modelo de aprendizaje de máquina

Evaluando una hipotesis

Evaluando su hipotesis



$$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

Falla en aplicar hipotesis en nuevos ejemplos

$x_1 =$ Tamaño de casa
$x_2 =$ Nro de dormitorios
$x_3 =$ Nro de pisos
$x_4 =$ Edad de casa
$x_5 =$ Ingreso promedio de vecindad
$x_6 =$ Tamaño de cocina
$\vdots$
x_{100}

Evaluando su hipotesis

Dataset:

Size	Price
2104	400
1600	330
2400	369
1416	232
3000	540
1985	300
1534	315
1427	199
1380	212
1494	243

$$(x^{(1)}, y^{(1)})$$
$$(x^{(2)}, y^{(2)})$$
$$\vdots$$
$$(x^{(m)}, y^{(m)})$$

$$(x_{test}^{(1)}, y_{test}^{(1)})$$
$$(x_{test}^{(2)}, y_{test}^{(2)})$$
$$\vdots$$
$$(x_{test}^{(m_{test})}, y_{test}^{(m_{test})})$$

Procedimiento de training/testing para regresion lineal

- Aprender parametro θ de data de entrenamiento (minimizar error de entrenamiento $J(\theta)$)

- Computar error de set de test:

$$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} \left(h_\theta(x_{test}^{(i)}) - y_{test}^{(i)} \right)^2$$

Procedimiento de training/testing para regresion logistica

- Aprender parametro θ de data de entrenamiento.

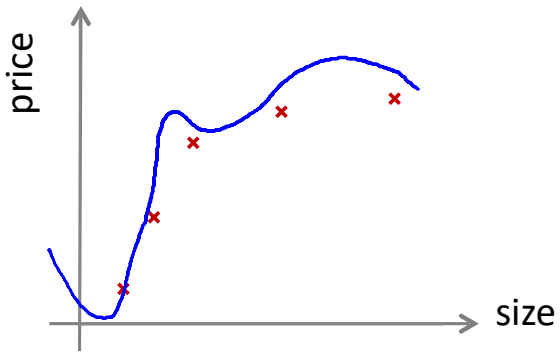
- Computar error de set de testeo:

$$J_{test}(\theta) = -\frac{1}{m_{test}} \sum_{i=1}^{m_{test}} y_{test}^{(i)} \log h_\theta(x_{test}^{(i)}) + (1 - y_{test}^{(i)}) \log h_\theta(x_{test}^{(i)})$$

Eligiendo un modelo de aprendizaje de máquina

Seleccion de modelo y sets de training/validation/test

Ejemplo de overfitting



$$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

Dado que $\theta_0, \theta_1, \ldots, \theta_4$ fueron ajustados a un set de datos particular, el error de los parametros medidos en tales datos (training error $J(\theta)$) es probable ser menor que el error real de generalizacion.

Seleccion de modelo

1. $h_\theta(x) = \theta_0 + \theta_1 x$
2. $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$
3. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3$

$\vdots$

10. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10}$

Escoger $\theta_0 + \ldots \theta_5 x^5$

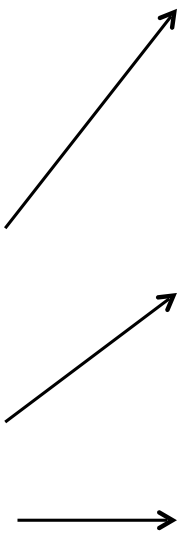
¿Cuan bien este modelo generaliza? Se debe reportar el error de test $J_{test}(\theta^{(5)})$

Problema: $J_{test}(\theta^{(5)})$ es probablemente una estimacion optimista del error. I.e. el parametro extra (d = grado de polinomio) se usa para hacer fit en el test.

Evaluando la hipotesis

Dataset:

Size	Price
2104	400
1600	330
2400	369
1416	232
3000	540
1985	300
1534	315
1427	199
1380	212
1494	243



$(x^{(1)}, y^{(1)})$
$(x^{(2)}, y^{(2)})$
$\vdots$
$(x^{(m)}, y^{(m)})$

$(x_{cv}^{(1)}, y_{cv}^{(1)})$
$(x_{cv}^{(2)}, y_{cv}^{(2)})$
$\vdots$
$(x_{cv}^{(m_{cv})}, y_{cv}^{(m_{cv})})$

$(x_{test}^{(1)}, y_{test}^{(1)})$
$(x_{test}^{(2)}, y_{test}^{(2)})$
$\vdots$
$(x_{test}^{(m_{test})}, y_{test}^{(m_{test})})$

Train/validation/test error

Training error:

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

Validation error:

Test error:

$$J_{test}(\theta) = \frac{1}{2m_{test}}\sum_{i=1}^{m_{test}}(h_\theta(x^{(i)}_{test}) - y^{(i)}_{test})^2$$

Seleccion de modelo

1. $h_\theta(x) = \theta_0 + \theta_1 x$
2. $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$
3. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3$

$\vdots$

10. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10}$

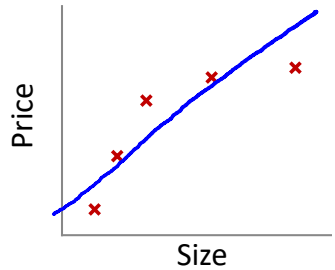
Escoger $\theta_0 + \theta_1 x_1 + \cdots + \theta_4 x^4$

Estimar error de generalizacion para set de test $J_{test}(\theta^{(4)})$

Eligiendo un modelo de aprendizaje de máquina

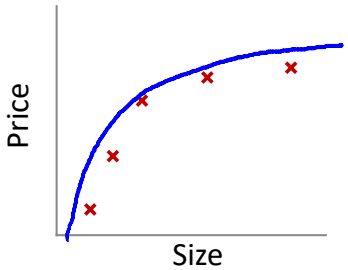
Bias vs. Variance

Bias/variance



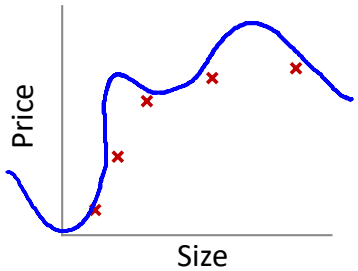
$\theta_0 + \theta_1 x$

Bias alto (underfit)



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

Modelo correcto



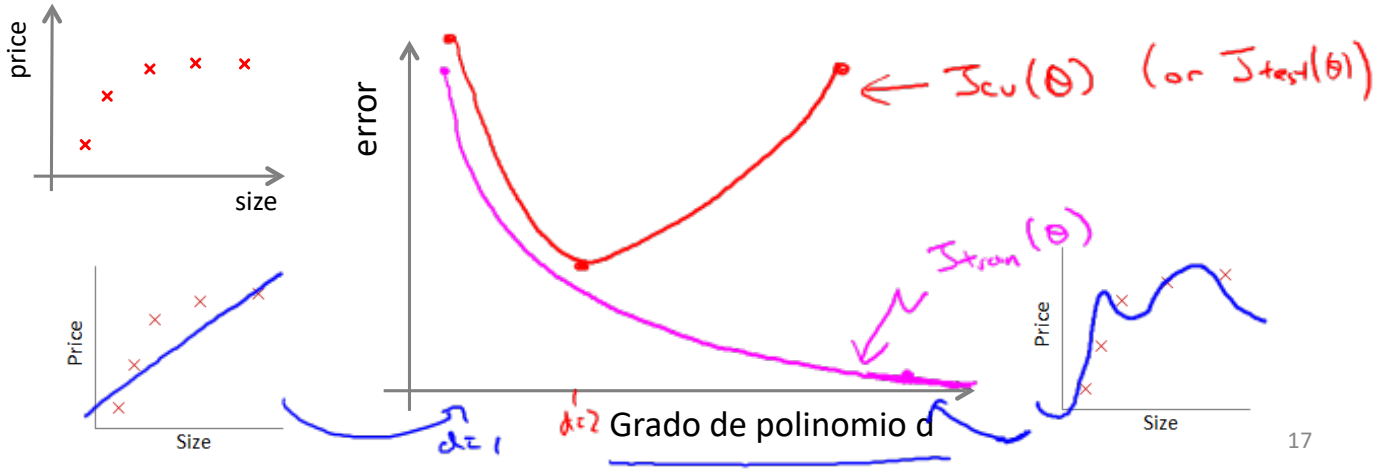
$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

Variance alta (overfit)

Bias/variance

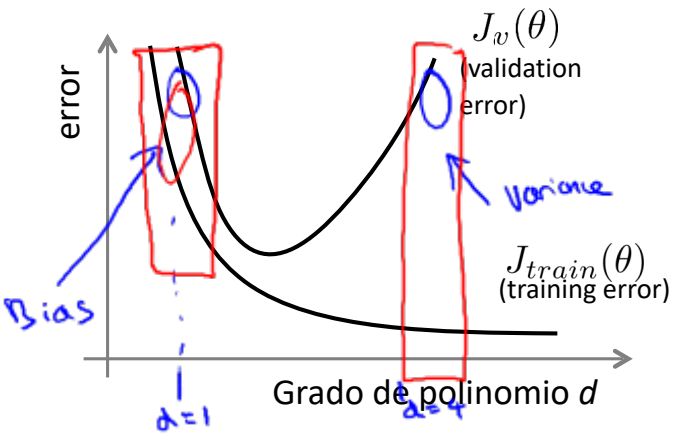
Training error: $J_{train}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$

Validation error:



¿Bias o variance?

Suponga que su algoritmo de aprendizaje esta andando mucho menos de lo esperaba ($J_v(\theta)$ o $J_{test}(\theta)$ es alta.) ¿es un problema de bias o variance?



Bias (underfit):

Variance (overfit):

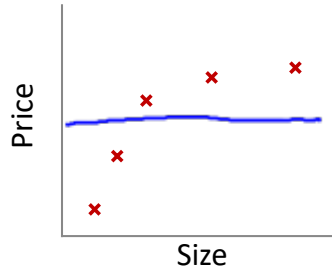
Eligiendo un modelo de aprendizaje de máquina

Regularizacion y bias/variance

Regresion lineal con regularizacion

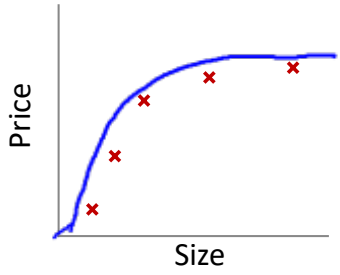
Modelo: $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

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

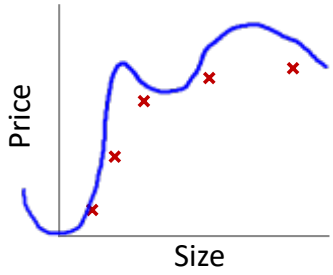


λ largo
Bias alto (underfit)

$\lambda = 10000.\ \theta_1 \approx 0, \theta_2 \approx 0, \ldots$
$h_\theta(x) \approx \theta_0$



λ intermedio
"correcto"



λ pequeño
variance alta (overfit)

Escogiendo el parametro de regularizacion λ

Modelo: $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

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

1. Probar $\lambda = 0$
2. Probar $\lambda = 0.01$
3. Probar $\lambda = 0.02$
4. Probar $\lambda = 0.04$
5. Probar $\lambda = 0.08$

   ⋮

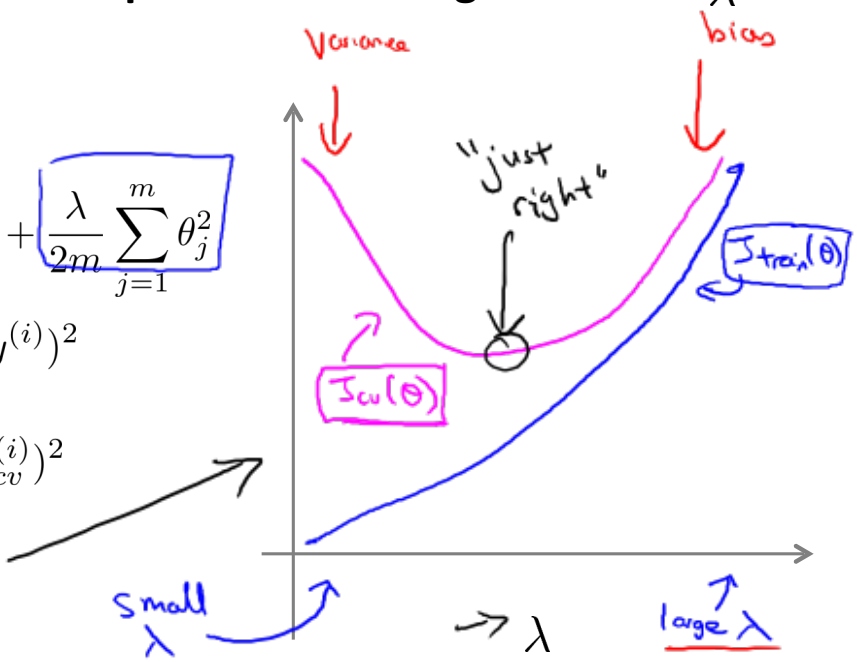
12. Probar $\lambda = 10$ Escoger (ejm) $\theta^{(5)}$. Test error:

Bias/variance como funcion de parametro de regularizacion λ

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

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

$$J_{cv}(\theta) = \frac{1}{2m_{cv}}\sum_{i=1}^{m_{cv}}(h_\theta(x_{cv}^{(i)}) - y_{cv}^{(i)})^2$$



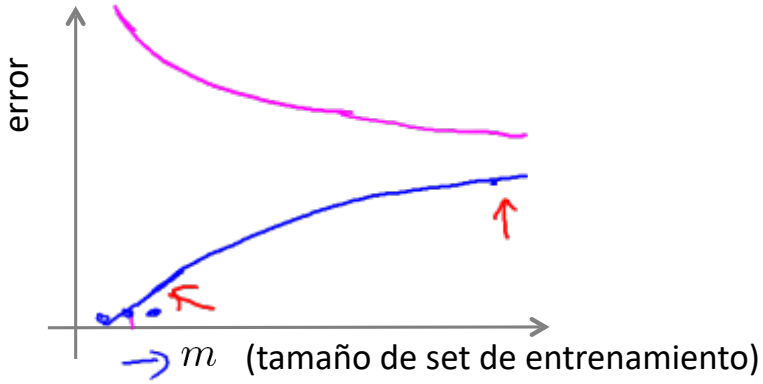
Eligiendo un modelo de aprendizaje de máquina

Entendiendo curvas

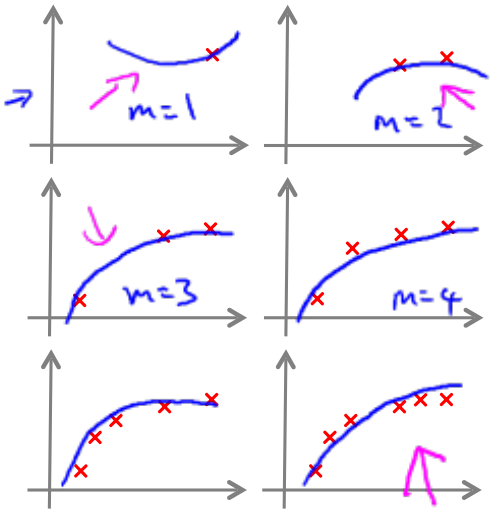
Entendiendo curvas

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

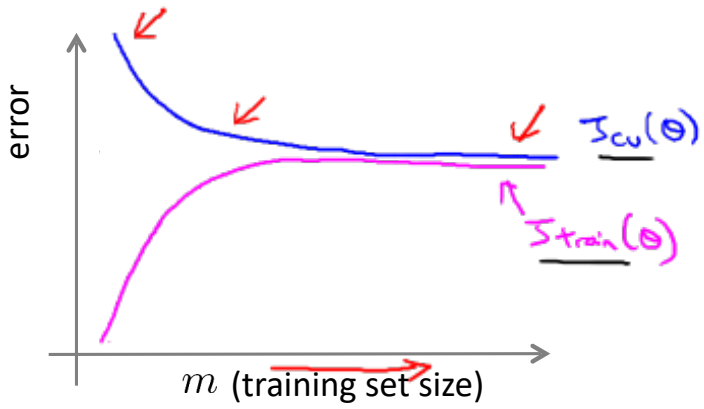
$$J_{cv}(\theta) = \frac{1}{2m_{cv}}\sum_{i=1}^{m_{cv}}(h_\theta(x_{cv}^{(i)}) - y_{cv}^{(i)})^2$$



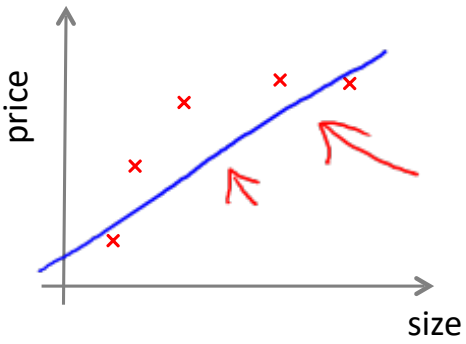
$$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$$

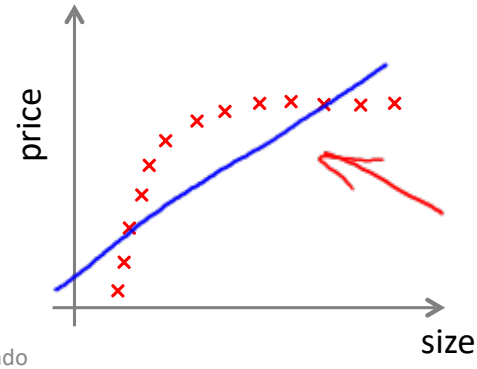


Alto bias



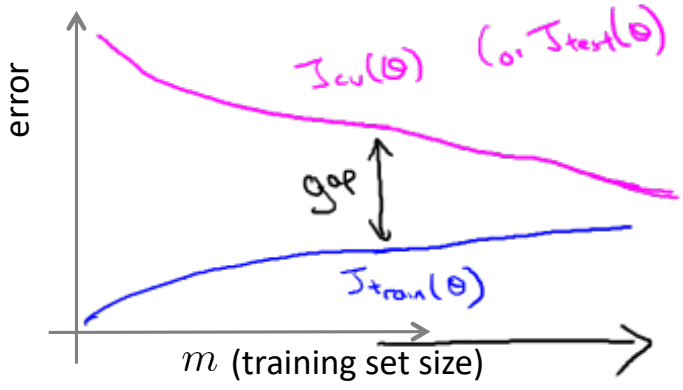
$h_\theta(x) = \theta_0 + \theta_1 x$



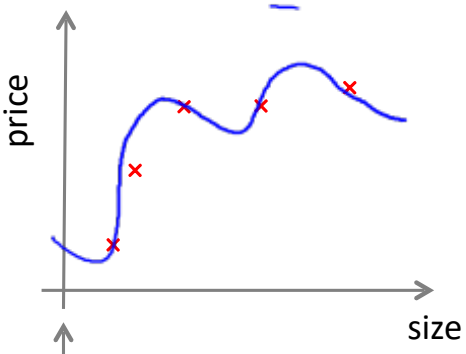


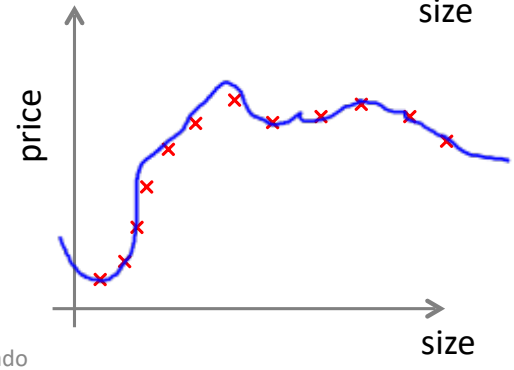
Si un algoritmo de aprendizaje esta sufriendo de alto bias, mas data de entrenamiento no ayudara mucho.

Alta varianza



$h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{100} x^{100}$
(and pequeño λ)





Si un algoritmo esta sufriendo de alta varianza, obtener mas datos de entrenamiento probablemente ayude.

Eligiendo un modelo de aprendizaje de máquina

¿Y ahora que? (soluciones)

Depurando un algoritmo de aprendizaje:
Supongamos que usted ha implementado un programa de regresion lineal regularizadapara predecir el precio de las casas.

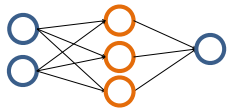
$$J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda\sum_{j=1}^{m}\theta_j^2\right]$$

Sin embargo, cuando aplica testea su hipotesis a nuevas casas, resulta que halla errores demasiado grandes. ¿que deberia hacer en tal caso

- Usar un set menor de features (vs alta varianza)
- Obtener features adicionales (vs alto bias)
- Adicionar features polinomiales $(x_1^2, x_2^2, x_1x_2, \text{etc.})$ (vs alto bias)
- Tratar decrementando λ (vs alto bias)
- Tratar incrementando λ (vs alta varianza)
- Obtener mas ejemplos de entrenamiento (vs alta varianza)

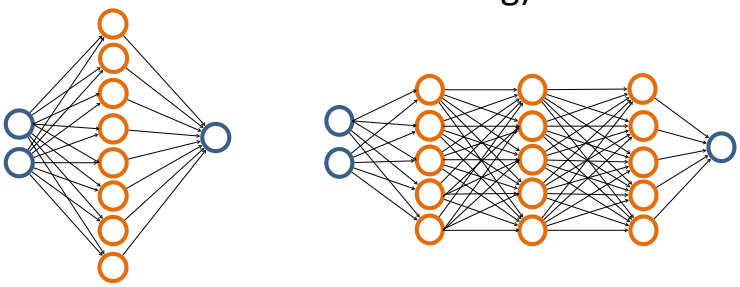
Redes neuronales y overfitting

Redes neuronales pequeñas (menos parametros, mas tendencia de underfitting)



Computacionalmente baratas

Redes neuronales grandes (mas parametros; mas tendencia de overfitting)



Computacionalmente mas costosa

Usar regularizacion (λ) para enfrentar overfitting.