HALMSTAD UNIVERSITY

Learning Systems (DT8008)

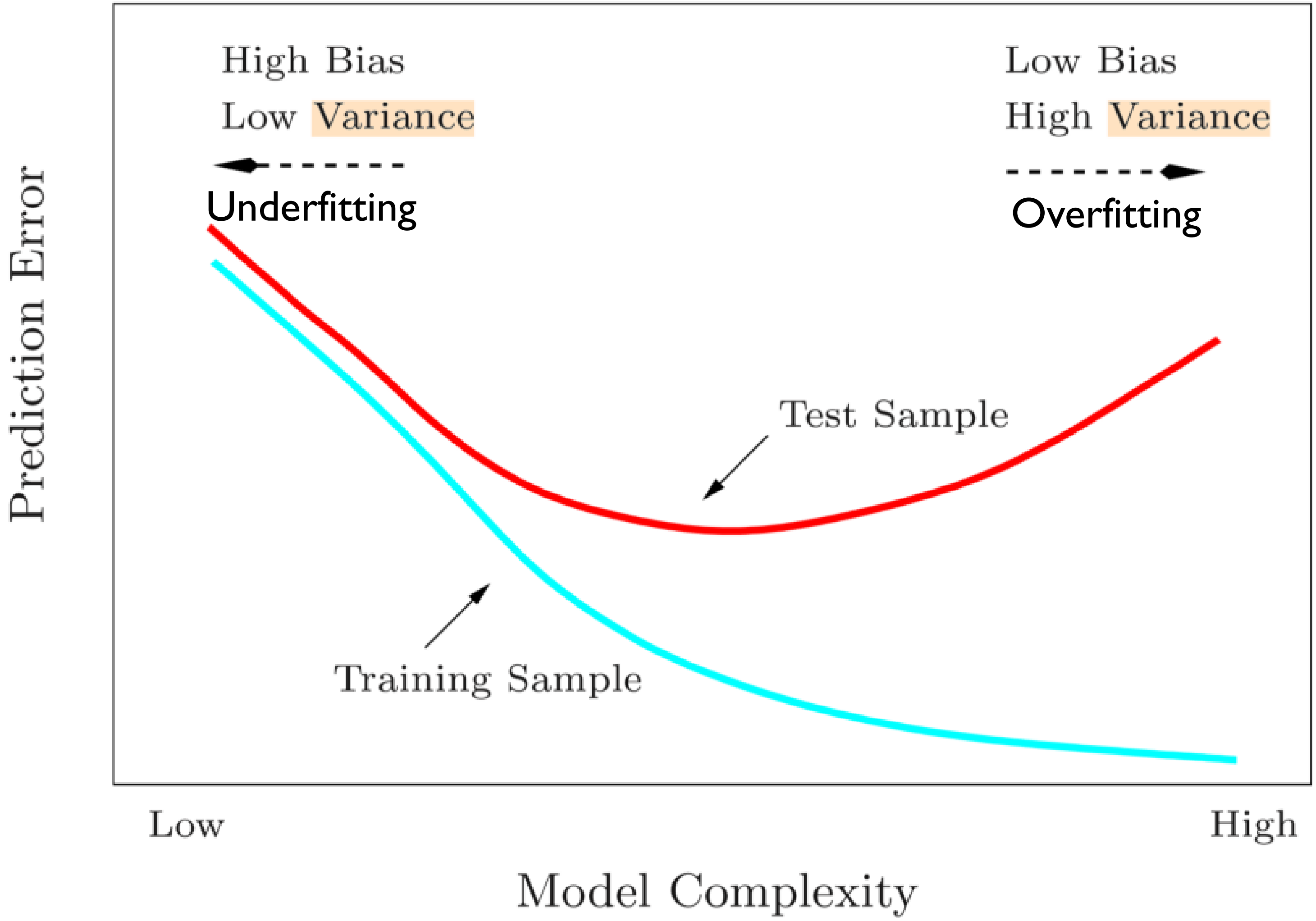
• Overfitting and Generalization
• Regularization

Dr. Mohamed-Rafik Bouguelia
mohamed-rafik.bouguelia@hh.se

Halmstad University

Quick reminder about overfitting

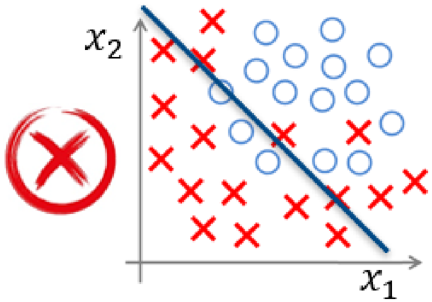
The problem of overfitting



HALMSTAD UNIVERSITY

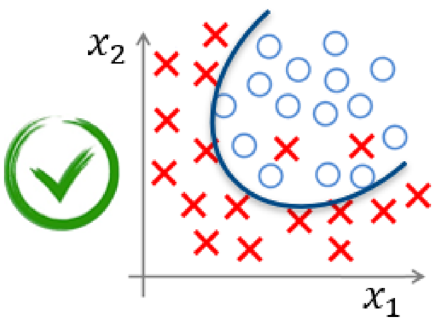
The problem of overfitting

Classification



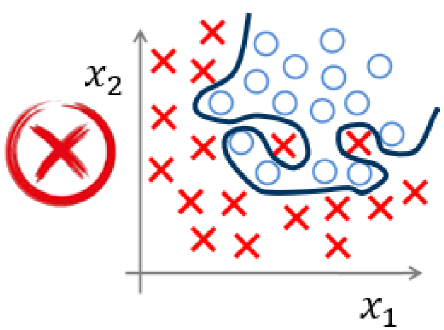
Simple model

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



More complex model

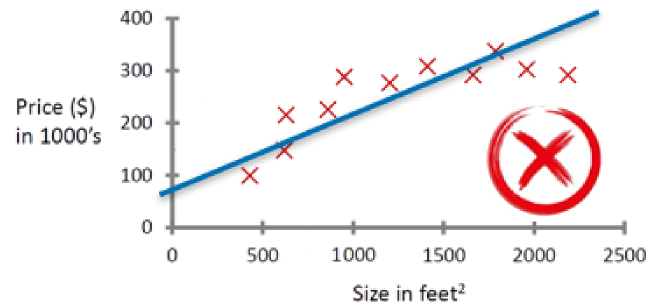
$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5\, x_1 x_2)$$



Much more complex model

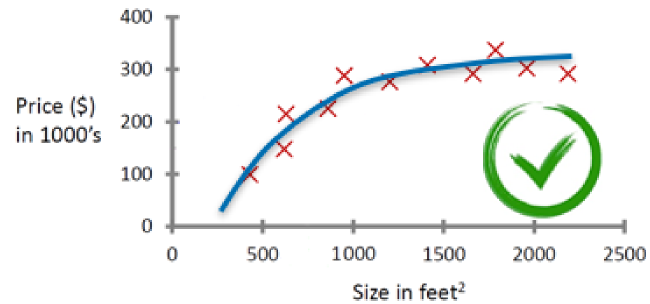
$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^2 x_2 + \theta_4\, x_1^2 x_2^2 + \theta_5 x_1^2 x_2^3 + \theta_6 x_1^3 x_2 + \cdots)$$

Regression



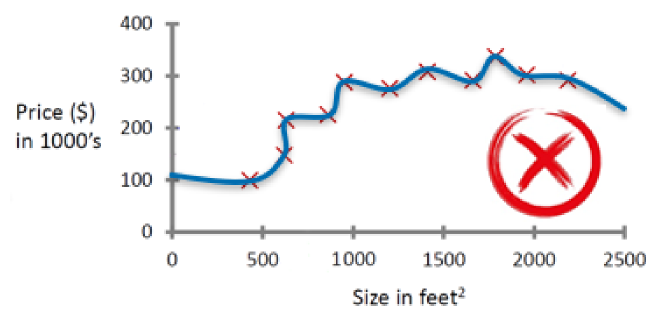
Simple model

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



More complex model

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



Much more complex model

$$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4 + \theta_5 x^5 + \theta_6 x^6 + \theta_7 x^7 + \theta_8 x^8 + \theta_9 x^9 + \theta_{10} x^{10}$$

Addressing overfitting

1. Model selection **(previous lecture)**
   – You can try various models (of different complexity) and compute the generalization error (as explained previously), and keep the best model.

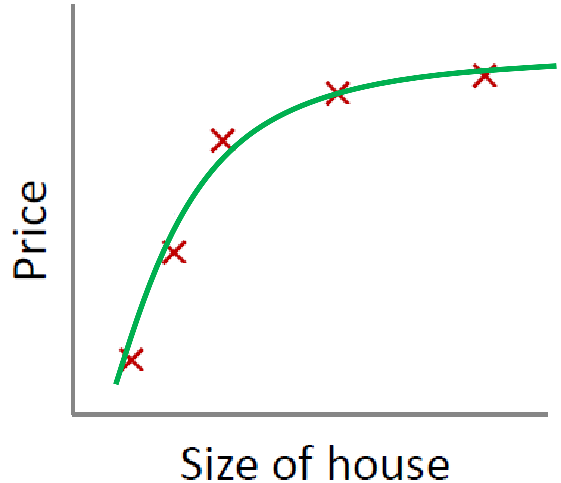
2. Reducing the number of features **(previous lecture)**
   – We are more likely to overfit when the number of features is high (relatively to the size of the dataset).
     - Manually select which features to keep / remove
     - Or using feature selection algorithms

3. Using an ensemble method **(previous lecture)**

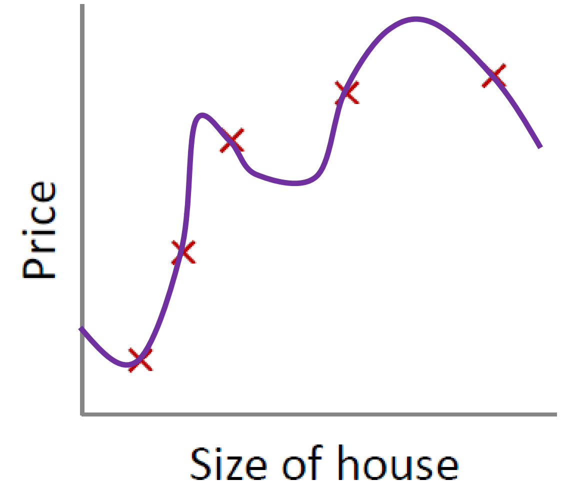
4. Using **regularization** **(this lecture)**
   – Keep all features, but reduce the magnitude / values of parameters θ_j
   – Works well when we have a lot of features, and each feature contributes a bit to predicting y

Regularization

Regularization - Motivation



$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2$$

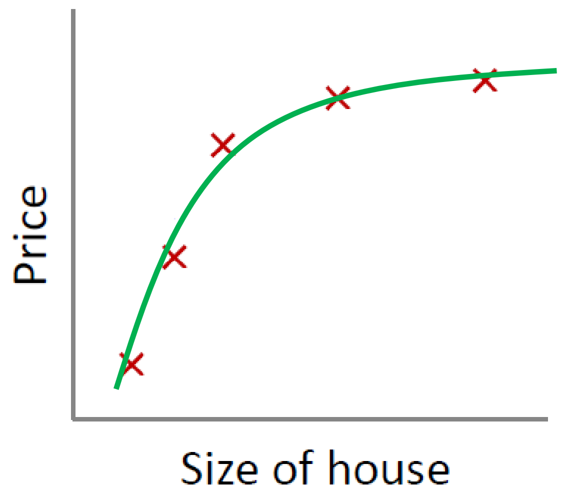


$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \boldsymbol{\theta_3 x_1^3} + \boldsymbol{\theta_4 x_1^4}$$

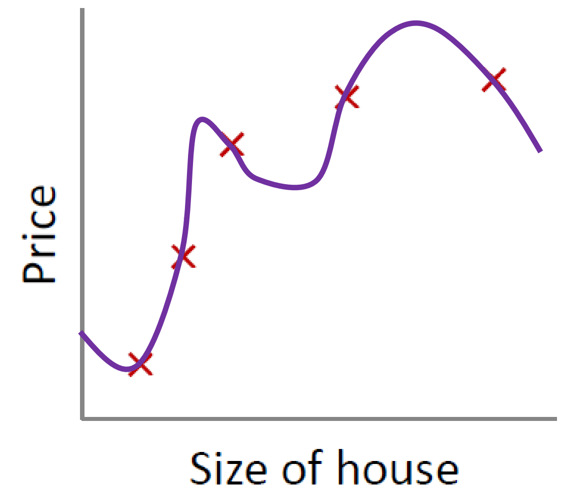
We added more features, e.g. $\boldsymbol{x_1^3}$ and $\boldsymbol{x_1^4}$

Overfits the data poorly and does not generalize well ☹

Regularization - Motivation



$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2$$



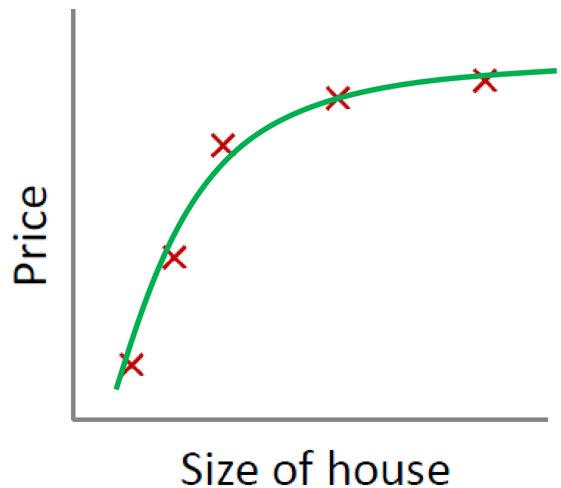
$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \boldsymbol{\theta_3 x_1^3} + \boldsymbol{\theta_4 x_1^4}$$

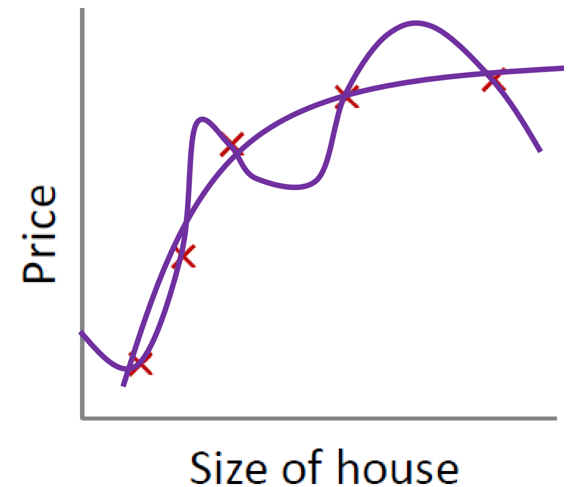
Suppose that we penalize and make θ_3, θ_4 really small.

$$\min_\theta \frac{1}{2n} \sum_{i=1}^{n} \left[h_\theta\left(x^{(i)}\right) - y^{(i)}\right]^2 + \mathbf{1000}\,\boldsymbol{\theta_3^2} + \mathbf{1000}\,\boldsymbol{\theta_4^2}$$

Then, the only way to make this new cost function small is if θ_3 and θ_4 are small

Regularization - Motivation





$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2$$

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \underbrace{\cancel{\boldsymbol{\theta_3 x_1^3}}}_{\approx \mathbf{0}} + \underbrace{\cancel{\boldsymbol{\theta_4 x_1^4}}}_{\approx \mathbf{0}}$$

Suppose that we penalize and make θ_3, θ_4 really small.

$$\min_\theta \frac{1}{2n} \sum_{i=1}^{n} \left[h_\theta\left(x^{(i)}\right) - y^{(i)} \right]^2 + \mathbf{1000}\ \boldsymbol{\theta_3^2} + \mathbf{1000}\ \boldsymbol{\theta_4^2}$$

Then, the only way to make this new cost function small is if θ_3 and θ_4 are small

Regularization

- Small values for parameters $\theta_0, \theta_1, \dots, \theta_p$
  - Implies a simpler hypothesis
  - Less prone to overfitting

- So we just modify our cost function as follows

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

λ = Regularization parameter
(it's a hyper-parameter)

Regularization

- Small values for parameters $\theta_0, \theta_1, \ldots, \theta_p$
  - Implies a simpler hypothesis
  - Less prone to overfitting

- So we just modify our cost function as follows

$$E(\theta) = \frac{1}{2n}\left[\underbrace{\sum_{i=1}^{n}[h_\theta(x^{(i)}) - y^{(i)}]^2}_{\text{Objective 1}} + \lambda \underbrace{\sum_{j=1}^{p}\theta_j^2}_{\text{Objective 2}}\right]$$

λ controls the trade-off between two objectives:

Objective 1:

- Fit the training dataset well

Objective 2:

- Keep the parameters small

Regularization

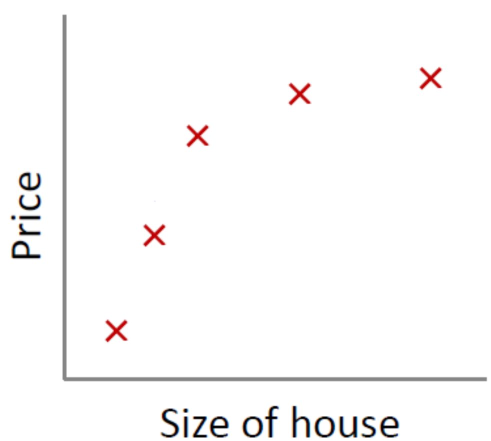
$$E(\theta) = \frac{1}{2n}\left[\sum_{i=1}^{n}[h_\theta(x^{(i)}) - y^{(i)}]^2 + \lambda \sum_{j=1}^{p} \theta_j^2\right]$$

What happens if λ is set to zero ?

- This becomes our original cost function. **Overfitting** can happen.

What happens if λ is set to an extremely large value**?**

- The algorithm might result in underfitting.
- Example for Linear Regression:



Suppose:

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^3 + \theta_4 x_4^4$$

HALMSTAD UNIVERSITY

Regularization

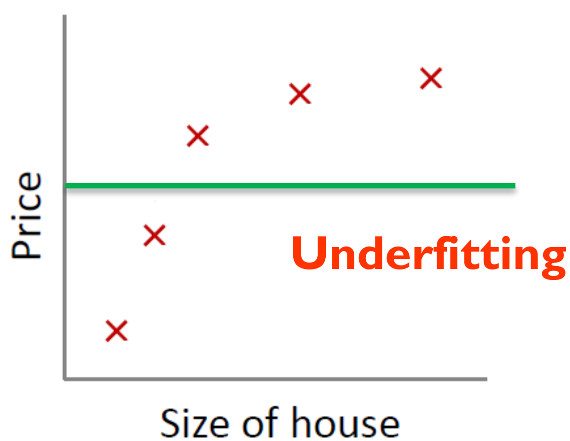
$$E(\theta) = \frac{1}{2n}\left[\sum_{i=1}^{n}[h_\theta(x^{(i)}) - y^{(i)}]^2 + \lambda \sum_{j=1}^{p} \theta_j^2\right]$$

What happens if λ is set to zero ?

- This becomes our original cost function. **Overfitting** can happen.

What happens if λ is set to an extremely large value**?**

- The algorithm might result in **underfitting**.
- Example for Linear Regression:



Suppose:

$$h_\theta(x) = \boldsymbol{\theta_0} + \cancel{\theta_1} x_1 + \cancel{\theta_2} x_1^2 + \cancel{\theta_3} x_1^3 + \cancel{\theta_4} x_4^4$$

We will end up penalizing $\theta_1, \theta_2, \theta_3, \theta_4$ (their value will be close to 0)

Regularization

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

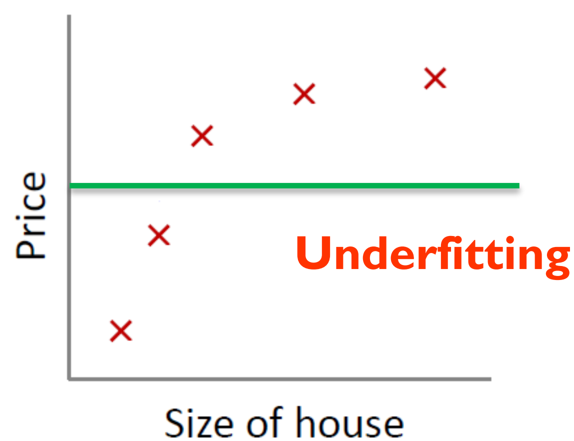
What happens if λ is set to zero ?

- This becomes our original cost function. **Overfitting** can happen

What happens if λ is set to an extremely large value**?**

- The algorithm might result in **underfitting**.
- Example for Linear Regression:

So, it's good to try several values for λ and estimate the generalization error each time ...



Suppose:

$$h_\theta(x) = \boldsymbol{\theta_0} + \cancel{\theta_1 x_1} + \cancel{\theta_2 x_1^2} + \cancel{\theta_3 x_1^3} + \cancel{\theta_4 x_4^4}$$

We will end up penalizing $\theta_1, \theta_2, \theta_3, \theta_4$ (their value will be close to 0)

Regularized Linear Regression

HALMSTAD UNIVERSITY

Regularized Linear Regression

We minimize:

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

where $h_\theta(x) = \theta^T x = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_d x_d$

- By the way, how can you write $E(\theta)$ in a more compact way, using vectors/matrices?

?

HALMSTAD UNIVERSITY

Regularized Linear Regression

We minimize:

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

where $h_\theta(x) = \theta^T x = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_d x_d$

- By the way, how can you write $E(\theta)$ in a more compact way, using vectors/matrices?

$$E(\theta) = \|\underbrace{X\theta}_{\text{vector of predictions}} - \underbrace{y}_{\text{vector of true outputs}}\|_2^2 + \lambda\left\|\underbrace{\hat{\theta}}_{\text{vector of parameters } \theta_1, \theta_2, \ldots, \theta_d}\right\|_2^2$$

vector of predictions

vector of true outputs

vector of parameters $\theta_1, \theta_2, \ldots, \theta_d$

Regularized Linear Regression

Gradient Descent

Repeat until convergence {

$$\theta_0 \leftarrow \theta_0 - \alpha \frac{1}{n} \sum_{i=0}^{n} [h_\theta(x^{(i)}) - y^{(i)}] x_0^{(i)}$$

$$\theta_j \leftarrow \theta_j - \alpha \left[\frac{1}{n} \sum_{i=0}^{n} [h_\theta(x^{(i)}) - y^{(i)}] x_j^{(i)} + \frac{\lambda}{n} \theta_j \right]$$

}

Update $\theta_0, \theta_1, \ldots, \theta_d$ <u>simultaneously</u>

same as

$$\theta_j \leftarrow \underbrace{\theta_j \left(1 - \alpha \frac{\lambda}{n}\right)}_{\text{Some ratio times current } \theta_j} - \underbrace{\alpha \frac{1}{n} \sum_{i=0}^{n} [h_\theta(x^{(i)}) - y^{(i)}] x_j^{(i)}}_{\text{This term is same as what we had previously in GD.}}$$

Regularized Linear Regression

Normal equation

- Previously (in the lecture about linear regression), when we computed the derivative of the cost function (without the regularization term) and set it equal to 0 (to find optimal θ), we found that the solution is:

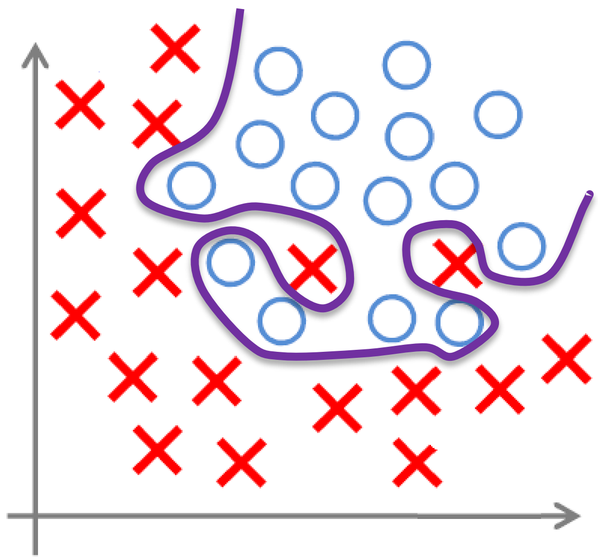
$$\theta = \left(X^T X \right)^{-1} X^T y$$

- If we do the same while including the regularization term in our cost function, then the solution would be:

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

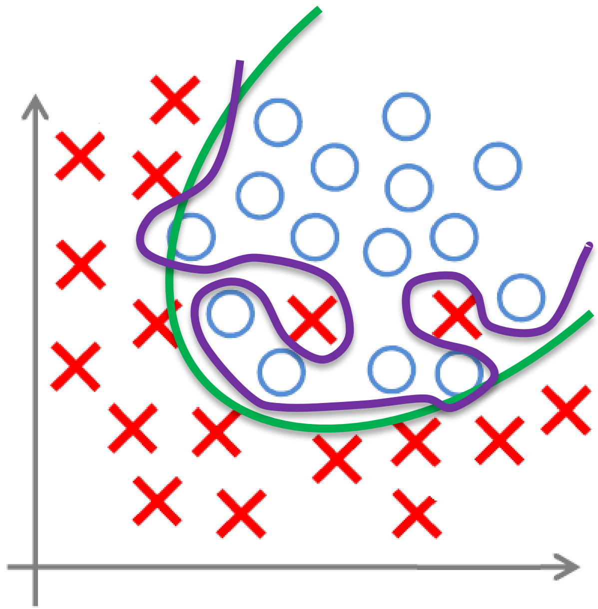
Regularized Logistic Regression (for classification)

Regularized Logistic Regression



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

Regularized Logistic Regression



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

$$E(\theta) = -\frac{1}{n}\left[\sum_{i=1}^{n}\left[y^{(i)}\ \log\left(h_\theta(x^{(i)})\right) + (1-y^{(i)})\ \log\left(1-h_\theta(x^{(i)})\right)\right] \underbrace{+ \lambda \sum_{j=1}^{p} \theta_j^2}_{\text{Regularization term}}\right]$$

Regularized Logistic Regression

Gradient Descent

Repeat until convergence {

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

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

}

Simultaneously update all parameters $\theta_0, \theta_1, \dots, \theta_p$

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

$$\frac{\partial E}{\partial \theta_j}$$