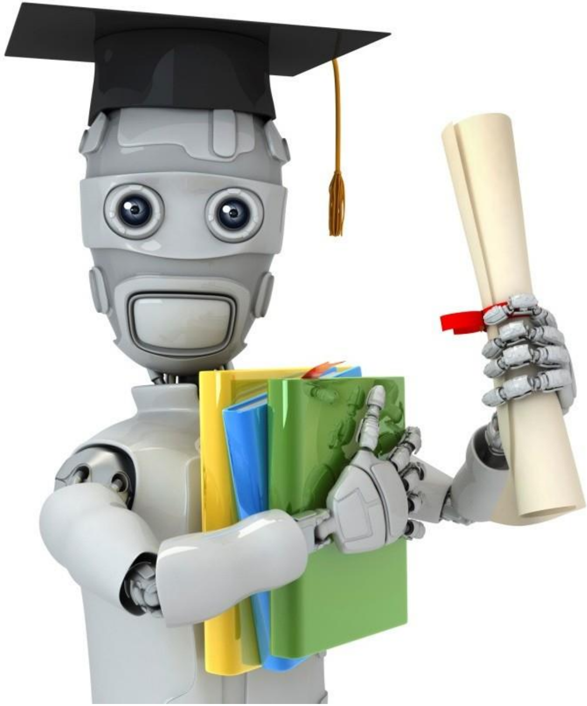
# Data Mining:
## Overfitting & Regularization

马锦华

数据科学与计算机学院
中山大学

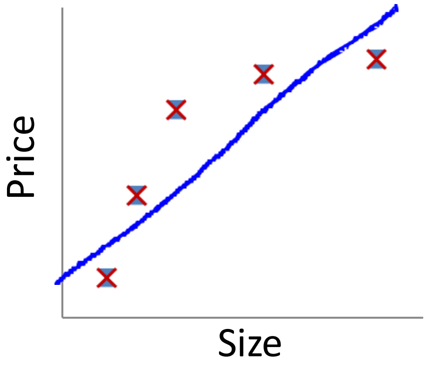


Machine Learning

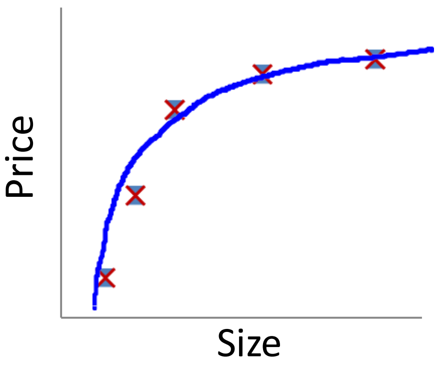
# Regularization

## The problem of overfitting

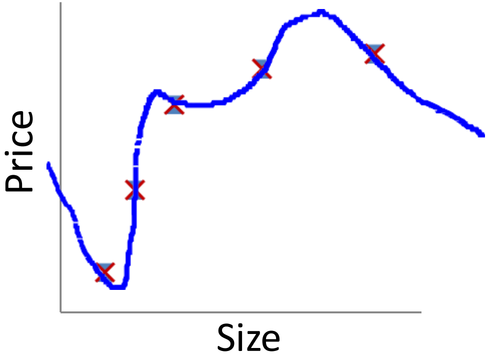
# Example: Linear regression (housing prices)




$\theta_0 + \theta_1 x$




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

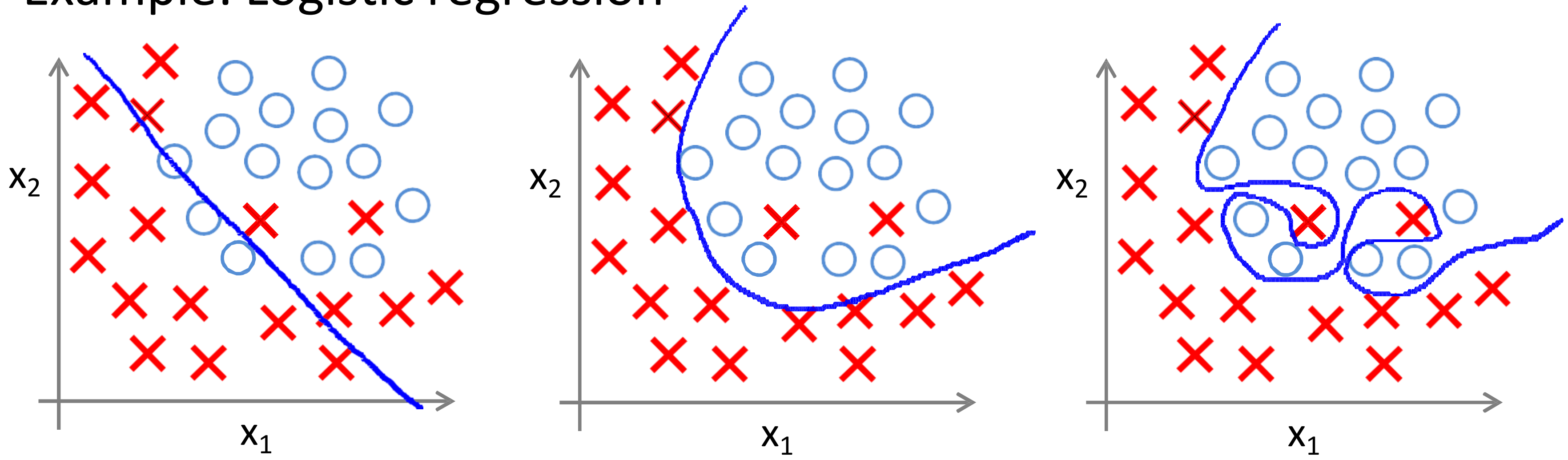



$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

**Overfitting:** If we have too many features, the learned hypothesis may fit the training set very well ($J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 \approx 0$), but fail to generalize to new examples (predict prices on new examples).

Andrew Ng

# Example: Logistic regression




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

( $g$ = sigmoid function)

$$g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2)$$

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

Andrew Ng

Consider the medical diagnosis problem of classifying tumors as malignant or benign. If a hypothesis $h_\theta(x)$ has overfit the training set, it means that:

- It makes accurate predictions for examples in the training set and generalizes well to make accurate predictions on new, previously unseen examples.
- It does not make accurate predictions for examples in the training set, but it does generalize well to make accurate predictions on new, previously unseen examples.
- It makes accurate predictions for examples in the training set, but it does not generalize well to make accurate predictions on new, previously unseen examples.
- It does not make accurate predictions for examples in the training set and does not generalize well to make accurate predictions on new, previously unseen examples.

Consider the medical diagnosis problem of classifying tumors as malignant or benign. If a hypothesis $h_\theta(x)$ has overfit the training set, it means that:

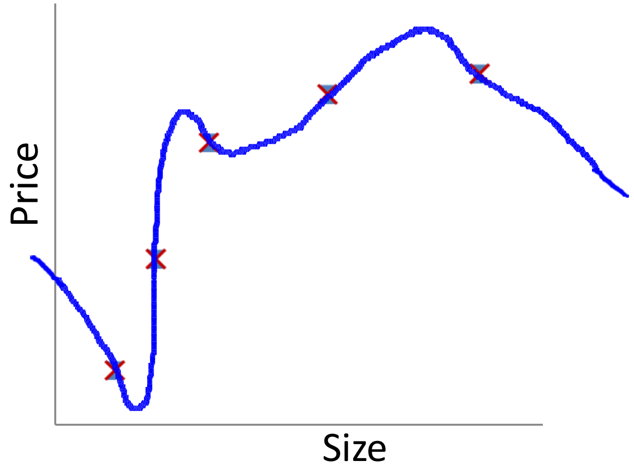
- It makes accurate predictions for examples in the training set and generalizes well to make accurate predictions on new, previously unseen examples.
- It does not make accurate predictions for examples in the training set, but it does generalize well to make accurate predictions on new, previously unseen examples.
- It makes accurate predictions for examples in the training set, but it does not generalize well to make accurate predictions on new, previously unseen examples.
- It does not make accurate predictions for examples in the training set and does not generalize well to make accurate predictions on new, previously unseen examples.

## Addressing overfitting:

$x_1 =$ size of house
$x_2 =$ no. of bedrooms
$x_3 =$ no. of floors
$x_4 =$ age of house
$x_5 =$ average income in neighborhood
$x_6 =$ kitchen size

$\vdots$

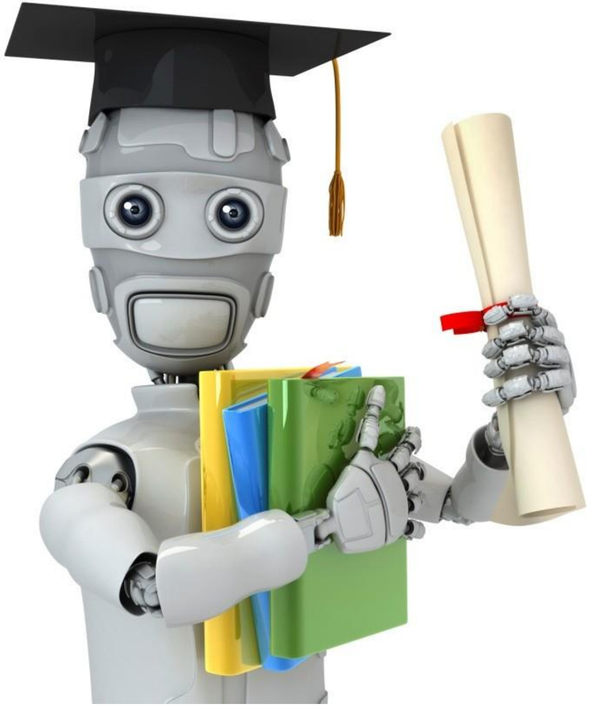
$x_{100}$



## Addressing overfitting:

Options:

1. Reduce number of features.
   - Manually select which features to keep.
   - Model selection algorithm (later in course).
2. Regularization.
   - Keep all the features, but reduce magnitude/values of parameters $\theta_j$.
   - Works well when we have a lot of features, each of which contributes a bit to predicting $y$.

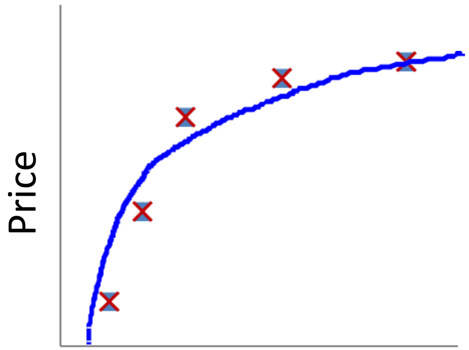


Machine Learning

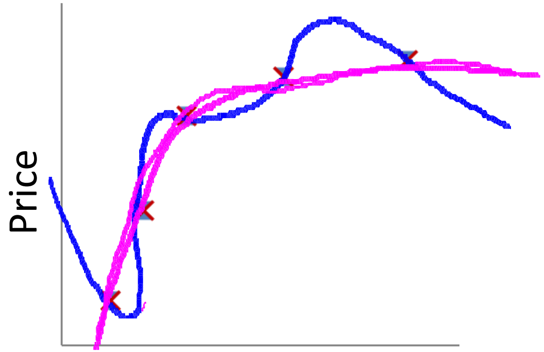
# Regularization

## Cost function

# Intuition




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




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

Suppose we penalize and make $\theta_3, \theta_4$ really small.

$$\min_\theta \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 + 1000\,\theta_3^2 + 1000\,\theta_4^2$$

# Regularization.

Small values for parameters $\theta_0, \theta_1, \ldots, \theta_n$

— "Simpler" hypothesis

— Less prone to overfitting

Housing:

— Features: $x_1, x_2, \ldots, x_{100}$

— Parameters: $\theta_0, \theta_1, \theta_2, \ldots, \theta_{100}$

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

## Regularization.

Small values for parameters $\theta_0, \theta_1, \ldots, \theta_n$

- "Simpler" hypothesis
- Less prone to overfitting

Housing:

- Features: $x_1, x_2, \ldots, x_{100}$
- Parameters: $\theta_0, \theta_1, \theta_2, \ldots, \theta_{100}$

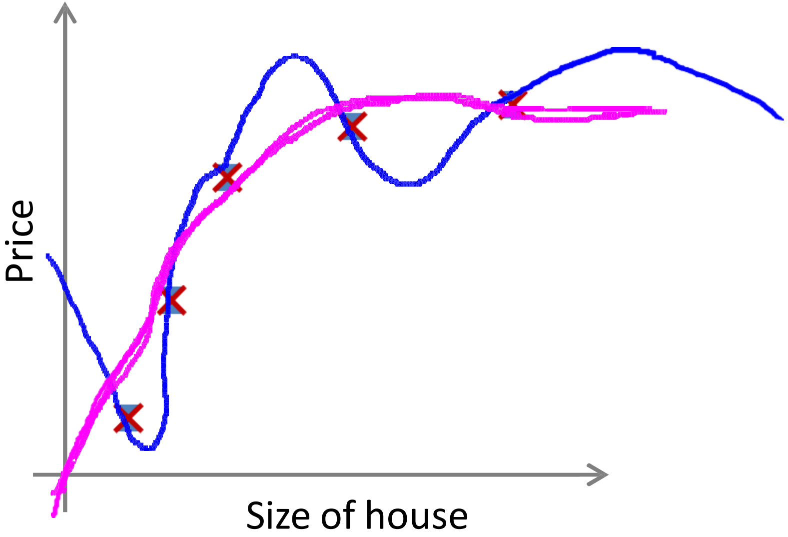
$$J(\theta) = \frac{1}{2m} \left[ \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^{n} \theta_j \right]$$

# Regularization.

Positive regularization parameter

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

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



In regularized linear regression, we choose $\theta$ to minimize

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

What if $\lambda$ is set to an extremely large value (perhaps for too large for our problem, say $\lambda = 10^{10}$)?

- Algorithm works fine; setting $\lambda$ to be very large can’t hurt it
- Algortihm fails to eliminate overfitting.
- Algorithm results in underfitting. (Fails to fit even training data well).
- Gradient descent will fail to converge.

In regularized linear regression, we choose $\theta$ to minimize

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

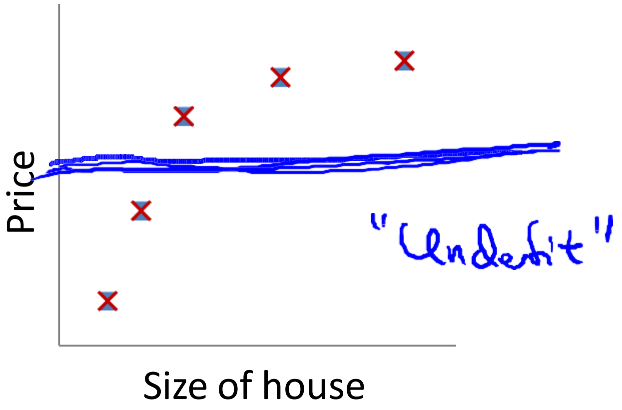
What if $\lambda$ is set to an extremely large value (perhaps for too large for our problem, say $\lambda = 10^{10}$)?

- Algorithm works fine; setting $\lambda$ to be very large can't hurt it
- Algortihm fails to eliminate overfitting.
- Algorithm results in underfitting. (Fails to fit even training data well).
- Gradient descent will fail to converge.

In regularized linear regression, we choose $\theta$ to minimize

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

What if $\lambda$ is set to an extremely large value (perhaps for too large for our problem, say $\lambda = 10^{10}$)?




$\underline{\theta_1}, \underline{\theta_2}, \underline{\theta_3}, \underline{\theta_4}$

$\theta_1 \approx 0, \theta_2 \approx 0$

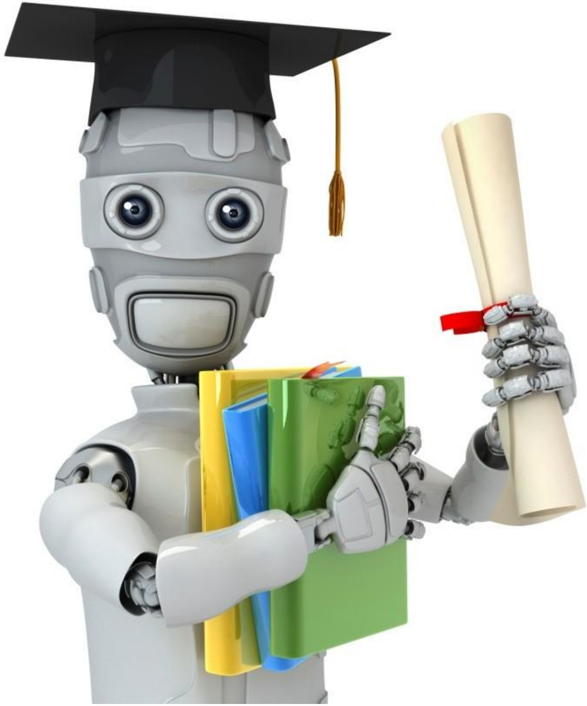
$\theta_3 \approx 0, \theta_4 \approx 0$

$$\boxed{h_\theta(x) = \theta_0}$$

$h_\theta(x)$

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

Andrew Ng



Machine Learning

# Regularization

## Regularized linear regression

# Regularized linear regression

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

$$\min_\theta \underline{J(\theta)}$$

# Gradient descent

Repeat $\{$

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

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

$$(j = \cancel{0}, \underline{1, 2, 3, \ldots, n})$$

$\}$

$J(\theta)$

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

# Normal equation

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

$m \times (n+1)$

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

$$\theta = \Big( X^T X + \lambda \underbrace{\begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}}_{(n+1)\times(n+1)} \Big)^{-1} X^T y$$

E.g. $n = 2$ $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

# Non-invertibility (optional/advanced).

Suppose $m \leq n$,
(#examples) (#features)

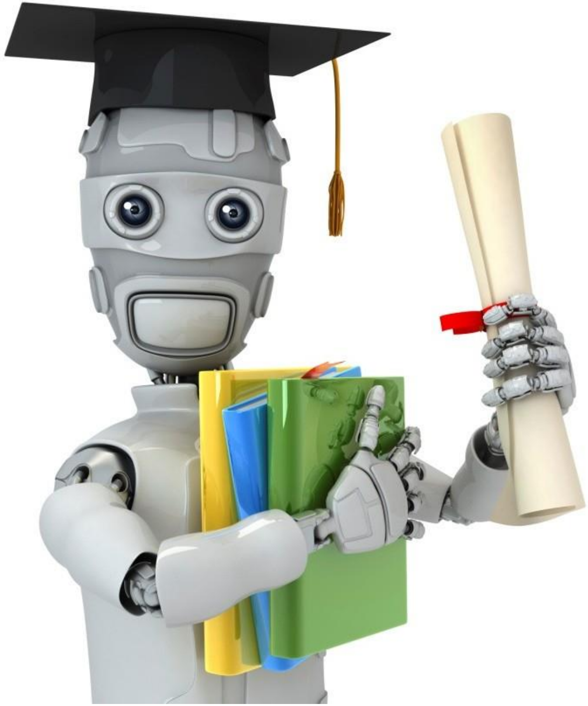
$$\theta = (X^T X)^{-1} X^T y$$

Non-invertible or singular

If $\lambda > 0$,

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

Invertible

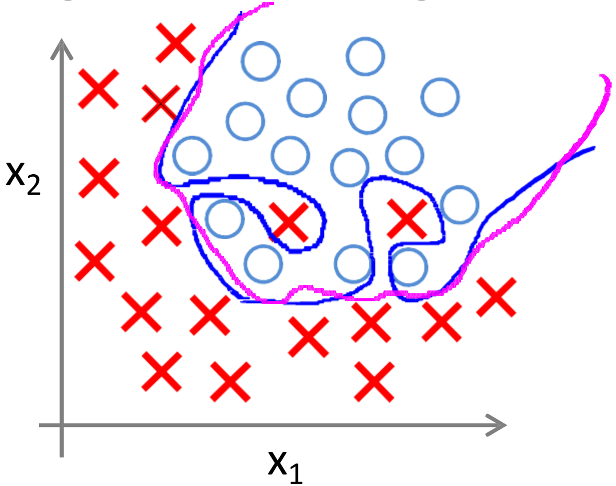


Machine Learning

# Regularization

## Regularized logistic regression

# Regularized logistic regression.




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

Cost function:

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

$$+ \frac{\lambda}{2m} \sum_{j=1}^{n} \theta_j^2$$

$\boxed{\theta_1, \theta_2, \dots, \theta_n}$

# Gradient descent

Repeat $\{$

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

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

$$(j = \cancel{0}, \underbrace{1, 2, 3, \ldots, n}_{\theta_1 \ldots \theta_n})$$

$\}$

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

# Unified Formulation

- General cost function

$$J(\boldsymbol{\theta}) = \lambda R(\boldsymbol{\theta}) + \frac{1}{m}\sum_{i=1}^{m} L\left(h_{\boldsymbol{\theta}}\left(\boldsymbol{x}^{(i)}\right), y^{(i)}\right)$$

  - Loss function $L$ measures the cost to pay for inaccuracy prediction, e.g.,
    - Square loss, Logistic loss, Hinge loss, etc.
  - Regularization function $R$ to make the prediction model simple or prevent overfitting, e.g.,
    - L2 norm (Tikhonov regularization): $R(\boldsymbol{\theta}) = \frac{1}{2}\|\boldsymbol{\theta}\|_2^2$
    - L1 norm (LASSO): $R(\boldsymbol{\theta}) = \|\boldsymbol{\theta}\|_1$
    - Elastic net: $R(\boldsymbol{\theta}) = \frac{\alpha}{2}\|\boldsymbol{\theta}\|_2^2 + (1-\alpha)\|\boldsymbol{\theta}\|_1, 0 < \alpha < 1$

# References


- Andrew Ng, Coursera: Machine Learning, https://www.coursera.org/learn/machine-learning
- Andrew Ng, "Feature selection, L 1 vs. L 2 regularization, and rotational invariance." ICML, 2004.
- Robert Tibshirani, "Regression shrinkage and selection via the lasso." Journal of the Royal Statistical Society. Series B (Methodological) (1996): 267-288.
- Hui Zou, and Trevor Hastie. "Regularization and variable selection via the elastic net." Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67.2 (2005): 301-320.