UNC CHARLOTTE

The WILLIAM STATES LEE COLLEGE of ENGINEERING

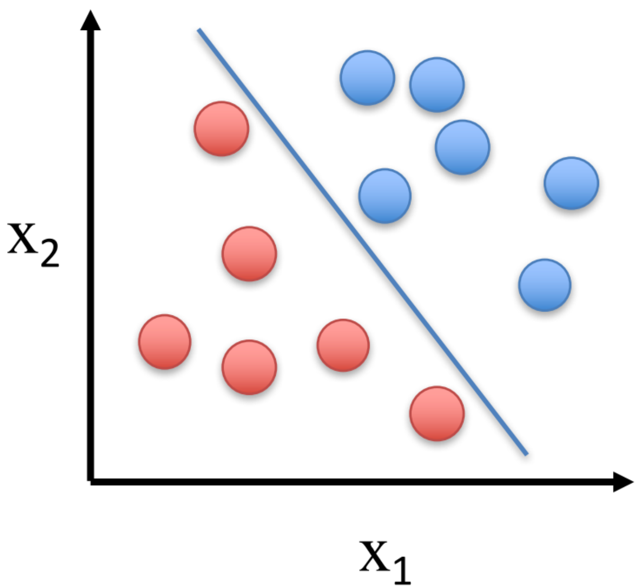
Introduction to ML
Lecture 4: Logistic Regression (Linear Classifier)

Hamed Tabkhi
Department of Electrical and Computer Engineering,
University of North Carolina Charlotte (UNCC)
htabkhiv@uncc.edu

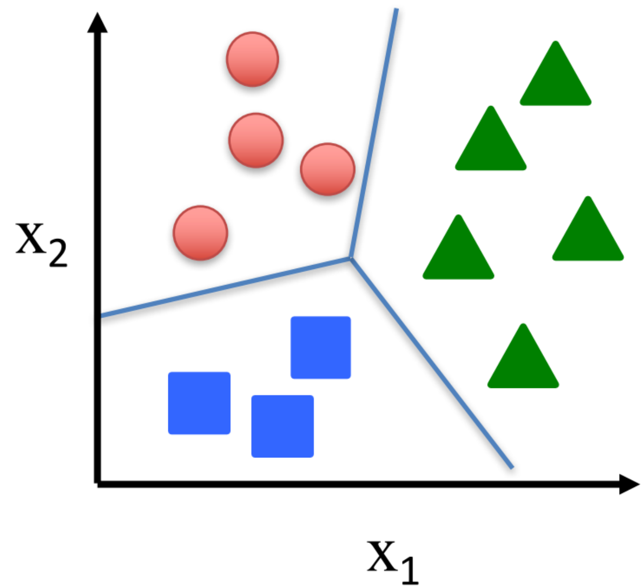
UNC CHARLOTTE

Multi-classification (e.g. two explanatory variables)

Binary classification:



Multi-class classification:



Disease diagnosis: healthy / cold / flu / pneumonia

Object classification: desk / chair / monitor / bookcase

Linear regression for classification

- We have discussed about regression
  - Output real value prediction

Classification

Email: Spam / Not Spam?
Online Transactions: Fraudulent (Yes / No)?
Tumor: Malignant / Benign ?

$y \in \{0, 1\}$

0: "Negative Class" (e.g., benign tumor)
1: "Positive Class" (e.g., malignant tumor)

Classification based on probability

- Instead of just predicting the class, give the probability of the instance being that class
  - i.e., learn $p(y \mid \boldsymbol{x})$

- Recall that:

$$0 \leq p(\text{event}) \leq 1$$

$$p(\text{event}) + p(\neg\text{event}) = 1$$

Not

Note: Although the name says "regression", but logistic regression is a classification approach

UNC CHARLOTTE

Logistic regression

- Why sigmoid function

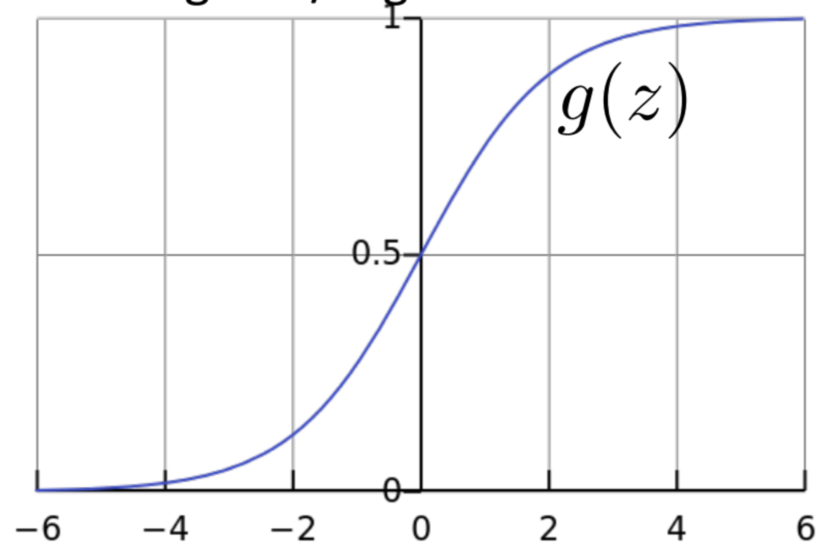
$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g\left(\boldsymbol{\theta}^\intercal \boldsymbol{x}\right)$$

$$g(z) = \frac{1}{1+e^{-z}}$$

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

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

Logistic / Sigmoid Function



Logistic regression

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g\left(\boldsymbol{\theta}^\top \boldsymbol{x}\right)$$

$$g(z) = \frac{1}{1 + e^{-z}}$$

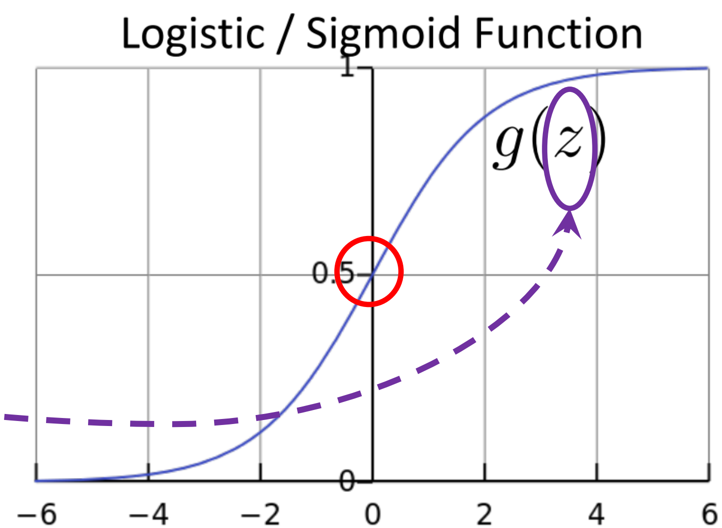
$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

- Assume a threshold and...
  - Predict y = 1 if $\boxed{h_{\boldsymbol{\theta}}(\boldsymbol{x}) \geq 0.5}$

    $\rightarrow \boldsymbol{\theta}^\top \boldsymbol{x} \geq 0$

  - Predict y = 0 if $\boxed{h_{\boldsymbol{\theta}}(\boldsymbol{x}) < 0.5}$

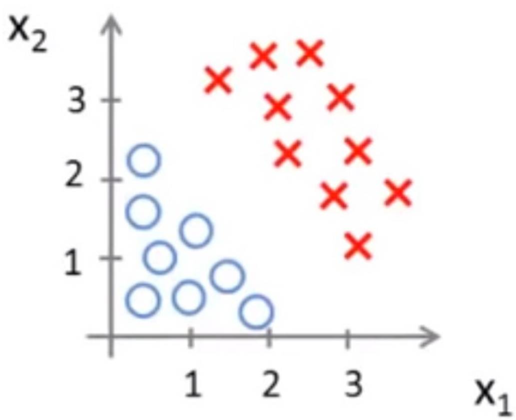
    $\rightarrow \boldsymbol{\theta}^\top \boldsymbol{x} < 0$



Logistic regression

- Example (let's see how the hypothesis is used to make predictions)

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

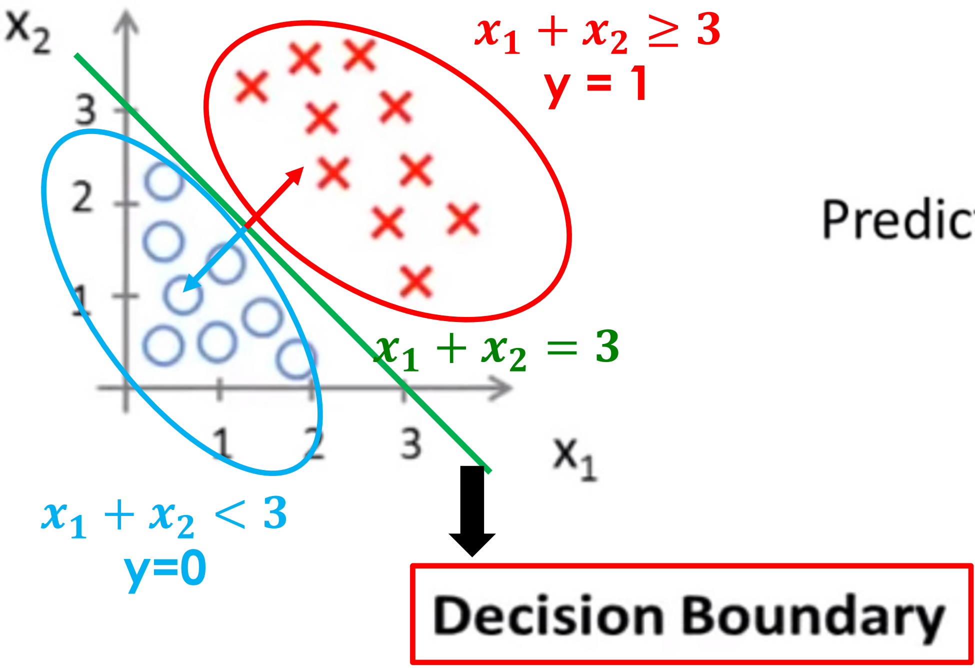


$$h_\theta(x) = g(\underline{\theta_0}_{-3} + \underline{\theta_1}_{1} x_1 + \underline{\theta_2}_{1} x_2)$$

$$\Downarrow$$

$$\boldsymbol{\theta}^\mathsf{T}\boldsymbol{x} = -3 + x_1 + x_2$$

Logistic regression



Predict "$y = 1$" if $-3 + x_1 + x_2 \geq 0$

$$x_1 + x_2 \geq 3$$

Predict "$y = 0$" if $x_1 + x_2 < 3$

Logistic regression – fit parameters

- Given $\left\{\left(\boldsymbol{x}^{(1)}, y^{(1)}\right), \left(\boldsymbol{x}^{(2)}, y^{(2)}\right), \ldots, \left(\boldsymbol{x}^{(m)}, y^{(m)}\right)\right\}$ m training samples

  where $\boldsymbol{x}^{(i)} \in \mathbb{R}^n,\ y^{(i)} \in \{0, 1\}$

- Model: $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g\left(\boldsymbol{\theta}^\top \boldsymbol{x}\right)$

$$g(z) = \frac{1}{1 + e^{-z}}$$ scales $\boldsymbol{\theta}^\top \boldsymbol{x}$ to [0, 1]

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} \qquad \boldsymbol{x}^\top = \begin{bmatrix} 1 & x_1 & \ldots & x_n \end{bmatrix}$$

How to choose parameter $\boldsymbol{\theta}$?

UNC CHARLOTTE

Logistic regression

Logistic regression cost function

$$\text{cost}\left(h_{\boldsymbol{\theta}}(\boldsymbol{x}), y\right) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})) & \text{if } y = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})) & \text{if } y = 0 \end{cases}$$

This cost function is convex

Logistic regression

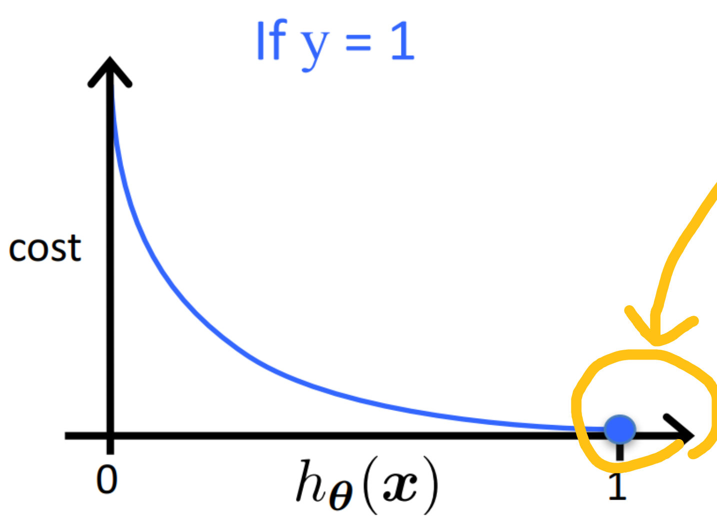
$$\text{cost}\,(h_{\boldsymbol{\theta}}(\boldsymbol{x}), y) = \begin{cases} \boxed{-\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})) \quad \text{if } y = 1} \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})) \quad \text{if } y = 0 \end{cases}$$

$$0 \leq h_{\boldsymbol{\theta}}(\boldsymbol{x}) \leq 1$$

Intuition behind the Objective

If y = 1

- Cost = 0 if prediction is correct



Probability = 1 (100% that the label is 1, which is the ground truth), so let's don't impose any cost

e.g.: probability = 0.8 (80% that the label is 1, which is a good prediction, but not perfect → add a small cost)

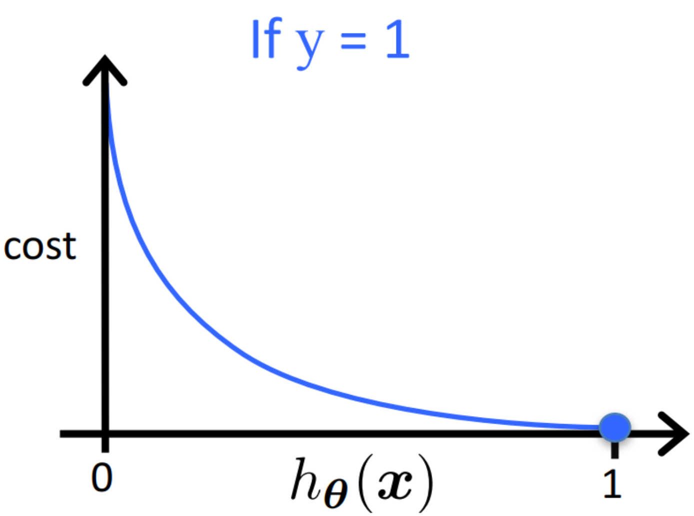
Logistic regression

$$\text{cost}\,(h_{\boldsymbol{\theta}}(\boldsymbol{x}), y) = \begin{cases} \boxed{-\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})) \quad \text{if } y = 1} \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})) \quad \text{if } y = 0 \end{cases}$$

$$0 \leq h_{\boldsymbol{\theta}}(\boldsymbol{x}) \leq 1$$

Intuition behind the Objective

If y = 1

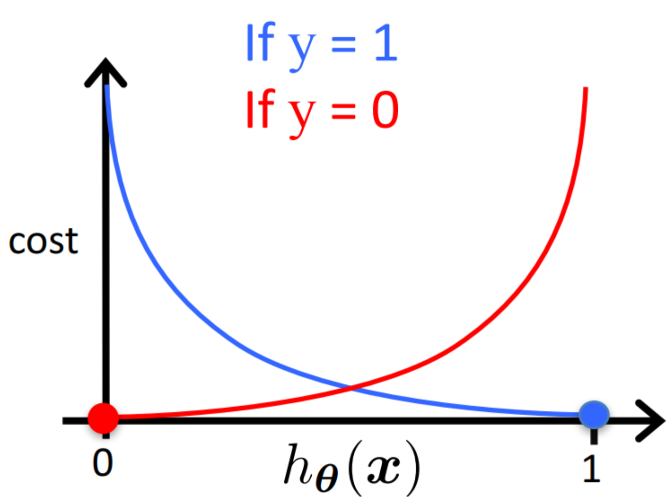


- Cost = 0 if prediction is correct
- As $h_{\boldsymbol{\theta}}(\boldsymbol{x}) \to 0, \text{cost} \to \infty$
- Captures intuition that larger mistakes should get larger penalties
  - e.g., predict $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0$, but y = 1

UNC CHARLOTTE

Logistic regression

$$\text{cost}\left(h_{\boldsymbol{\theta}}(\boldsymbol{x}), y\right) = \begin{cases} \boxed{-\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})) \quad \text{if } y = 1} \\ \boxed{-\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})) \quad \text{if } y = 0} \end{cases}$$



If y = 0

- Cost = 0 if prediction is correct
- As $(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})) \rightarrow 0, \text{cost} \rightarrow \infty$
- Captures intuition that larger mistakes should get larger penalties

The cost function of logistic regression

Logistic regression cost function

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \mathrm{Cost}(h_\theta(x^{(i)}), y^{(i)})$$

$$\mathrm{Cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{if } y = 1 \\ -\log(1 - h_\theta(x)) & \text{if } y = 0 \end{cases}$$

Note: $y = 0$ or 1 always

Compact form:

$$\boxed{Cost\ (h_\theta(x), y) = -y \log\big(h_\theta(x)\big) - (1 - \mathrm{y}) \log(1 - h_\theta(x))}$$

UNC CHARLOTTE

Logistic regression

Find the parameters using Gradient descent

Gradient Descent

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

Want $\min_\theta J(\theta)$:

Repeat $\{$

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

(simultaneously update all θ_j)

$\}$

Logistic regression

$$\frac{\partial}{\partial \theta_j} J(\theta) = ?$$

- Logistic regression model:

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g\left(\boldsymbol{\theta}^{\mathsf{T}} \boldsymbol{x}\right)$$

$$g(z) = \frac{1}{1+e^{-z}}$$

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

$$\begin{aligned} g'(z) &= \frac{d}{dz} \frac{1}{1+e^{-z}} \\ &= \frac{1}{(1+e^{-z})^2}\left(e^{-z}\right) \\ &= \frac{1}{(1+e^{-z})} \cdot \left(1 - \frac{1}{(1+e^{-z})}\right) \\ &= g(z)(1-g(z)). \end{aligned}$$

See here: https://towardsdatascience.com/derivative-of-the-sigmoid-function-536880cf918e

You can do the math yourself, if you are interested!

Logistic regression

Gradient Descent

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

Want $\min_\theta J(\theta)$:

This looks IDENTICAL to linear regression!!!

Repeat $\{$

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

(simultaneously update all θ_j)

$\}$

Another good reason of using Sigmoid function: mathematical convenient when computing the derivative

- However, the form of the model is very different:

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

Gradient descent for Linear Regression

Repeat {

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

}

Gradient descent for Logistic Regression

Repeat {

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right) x_j^{(i)} \qquad \boxed{h_\theta(x) = \frac{1}{1 + e^{-\theta^\top x}}}$$

}

UNC CHARLOTTE

Logistic regression

We can use gradient descent to learn parameter values, and hence compute the prediction for a new input.

To make a prediction given new x :

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

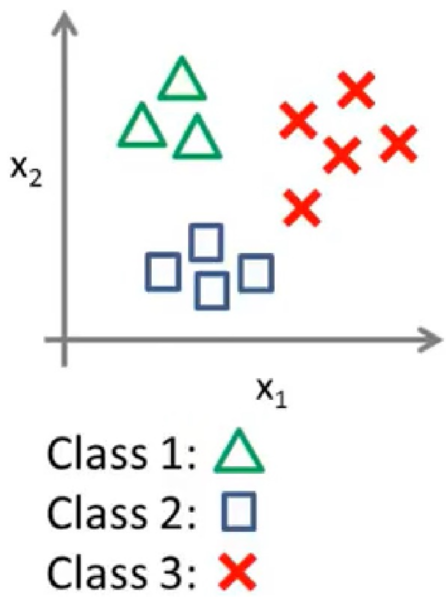
= estimated probability that y = 1 on input x

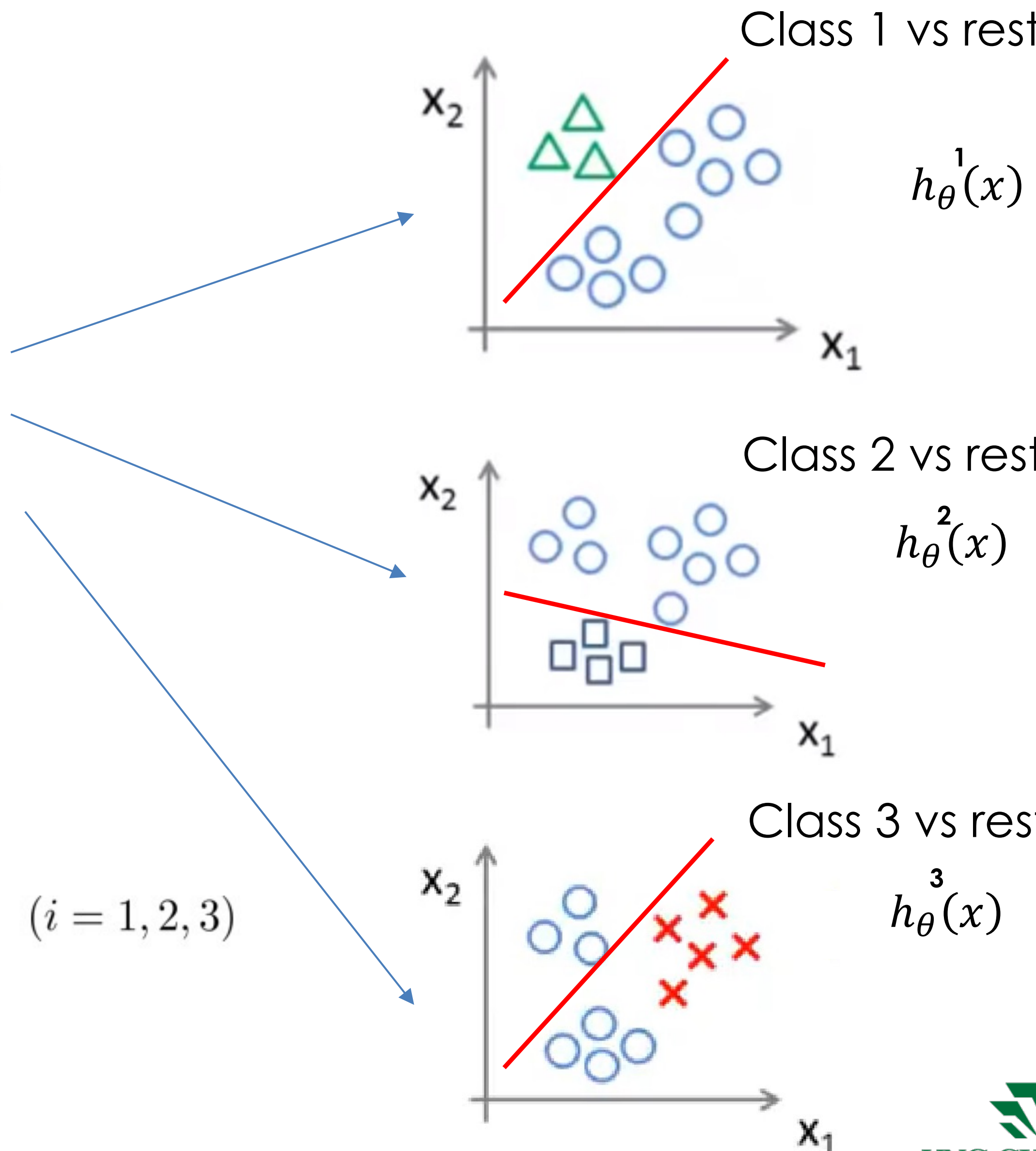
Logistic regression

- How to solve multi-class classification?

Logistic regression

One-vs-all (one-vs-rest):





$$h_\theta^{(i)}(x) = P(y = i|x;\theta) \qquad (i = 1, 2, 3)$$

Logistic regression

One-vs-all

Train a logistic regression classifier $h_\theta^{(i)}(x)$ for each class i to predict the probability that $y = i$.

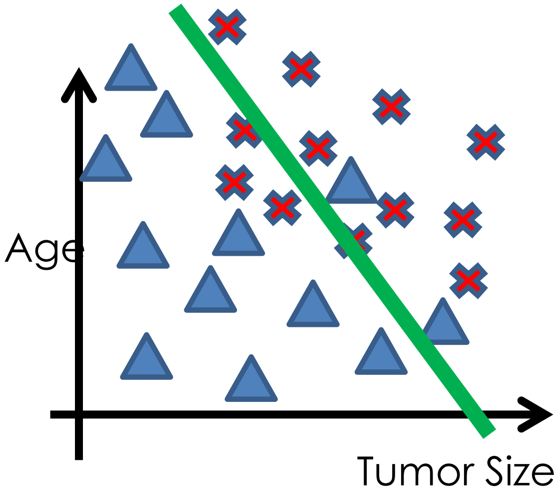
On a new input x, to make a prediction, pick the class i that maximizes

$$\max_i \underbrace{h_\theta^{(i)}(x)}_{\text{Probability score}}$$

Logistic regression

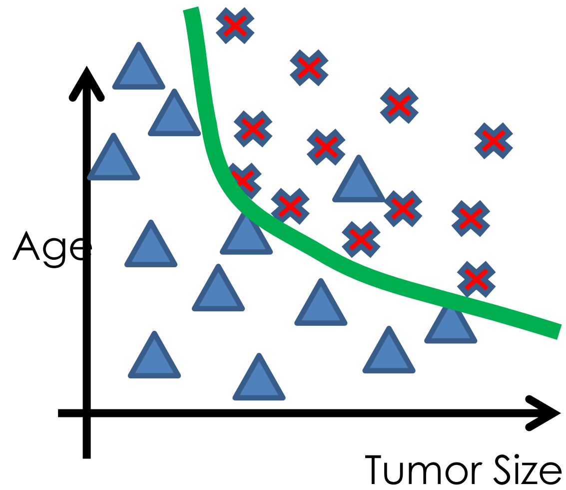
- How to perform regularization in logistic regression?

Overfitting

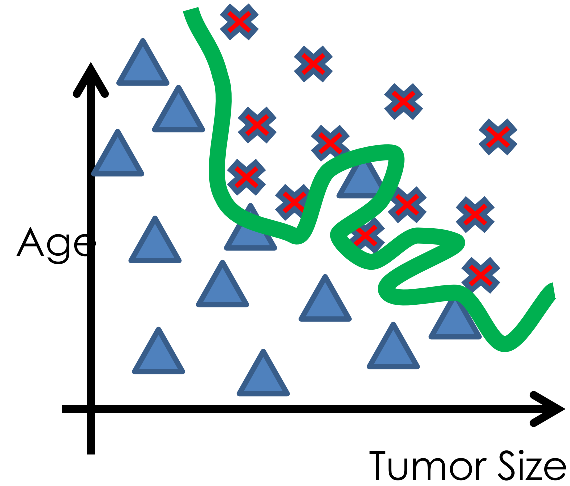


$h_\theta(x) = g(\theta_0 + \theta_1 x + \theta_2 x_2)$

Underfitting



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



$h_\theta(x) = g(\theta_0 + \theta_1 x + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2 + \theta_6 x_1^3 x_2 + \theta_7 x_1 x_2^3 + \cdots)$

Overfitting

- Learning the training data too precisely usually leads to poor classification results on new data.
- Classifier has to have the ability to generalize.

Recap: Regularized linear regression

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

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

n: Number of features

θ_0 is not panelized

Regularized logistic regression

- Regularized Logistic Regression

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

$$J_{\text{regularized}}(\boldsymbol{\theta}) = J(\boldsymbol{\theta}) + \frac{\lambda}{2m} \sum_{j=1}^{n} \theta_j^2$$

UNC CHARLOTTE

Regularized logistic regression

- Regularized Logistic Regression

$$J_{\text{regularized}}(\boldsymbol{\theta}) = J(\boldsymbol{\theta}) + \frac{\lambda}{2m} \sum_{j=1}^{n} \theta_j^2$$

Gradient decent update

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

This looks IDENTICAL to linear regression

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

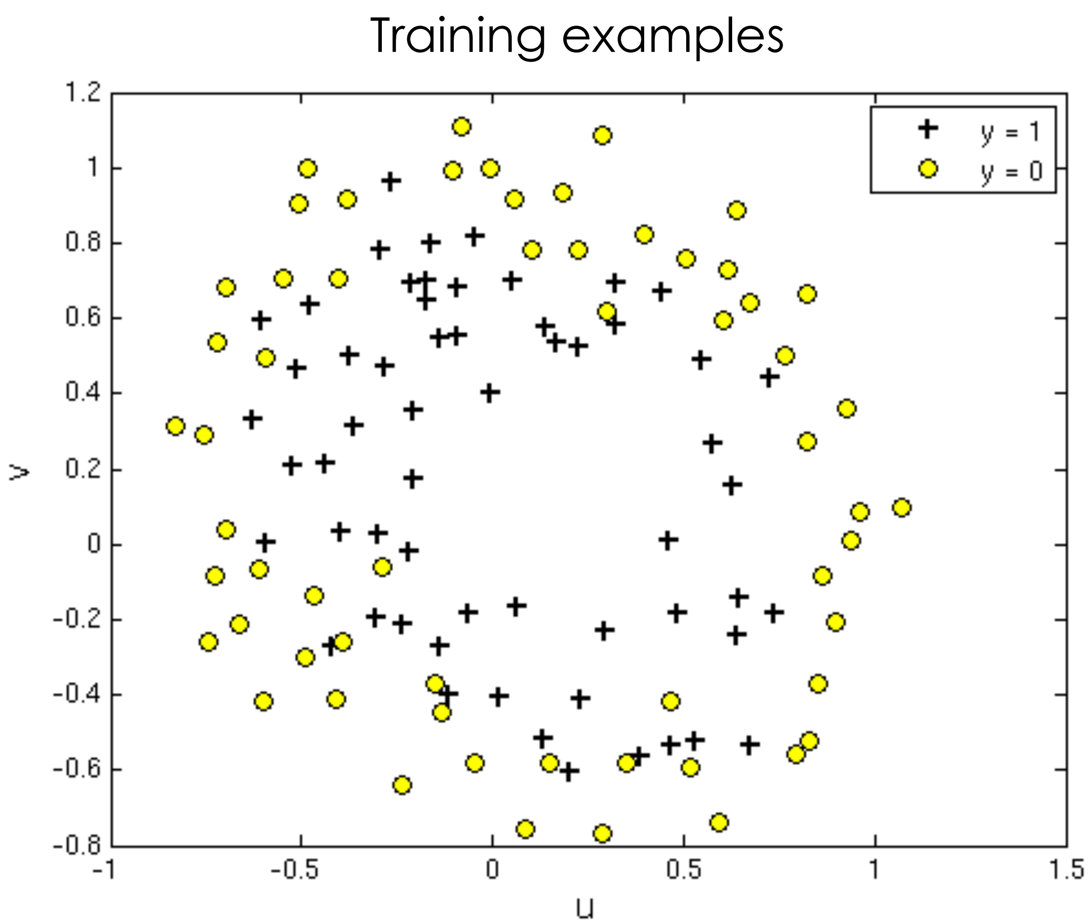
However, the form of the model is very different:

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

UNC CHARLOTTE

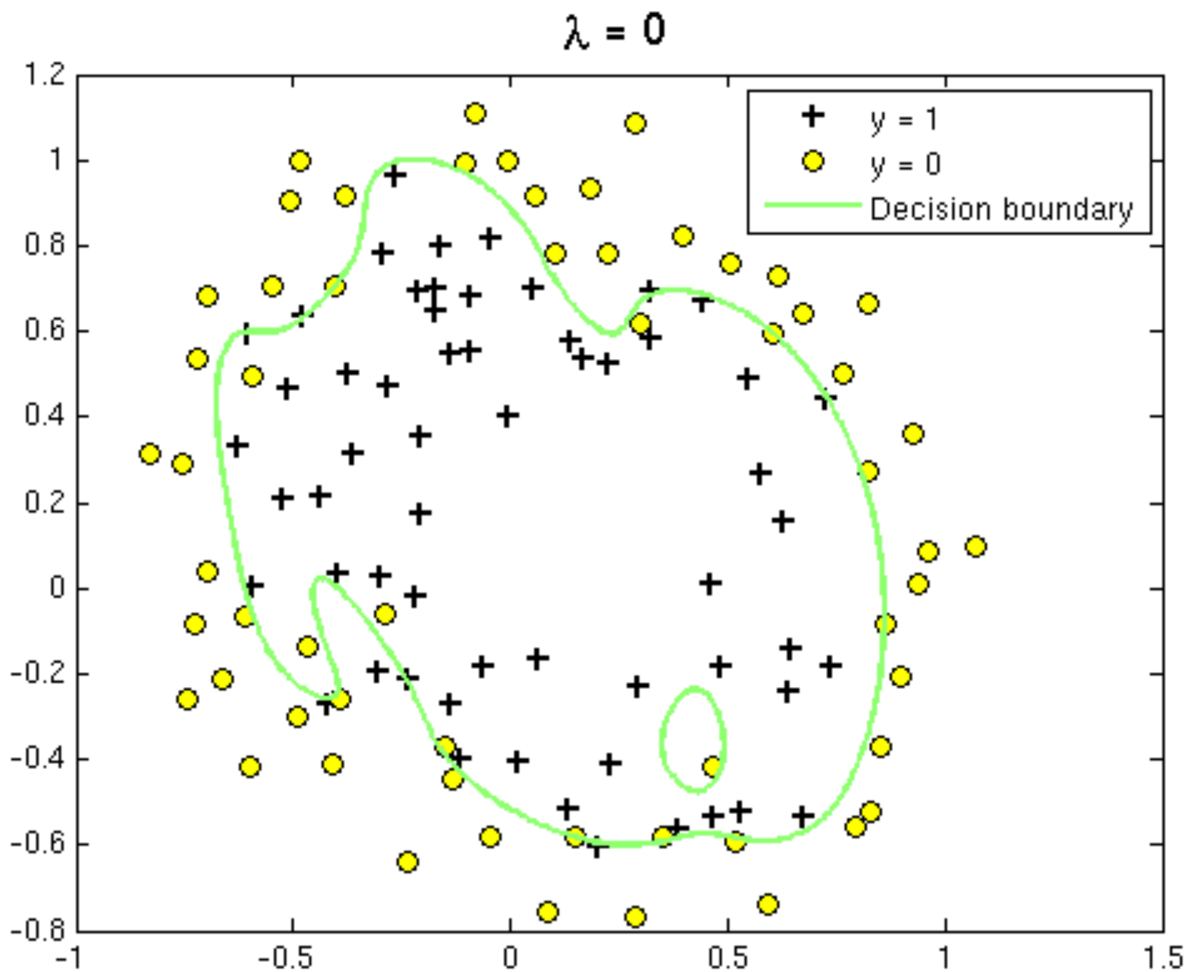
Regularization

- Example

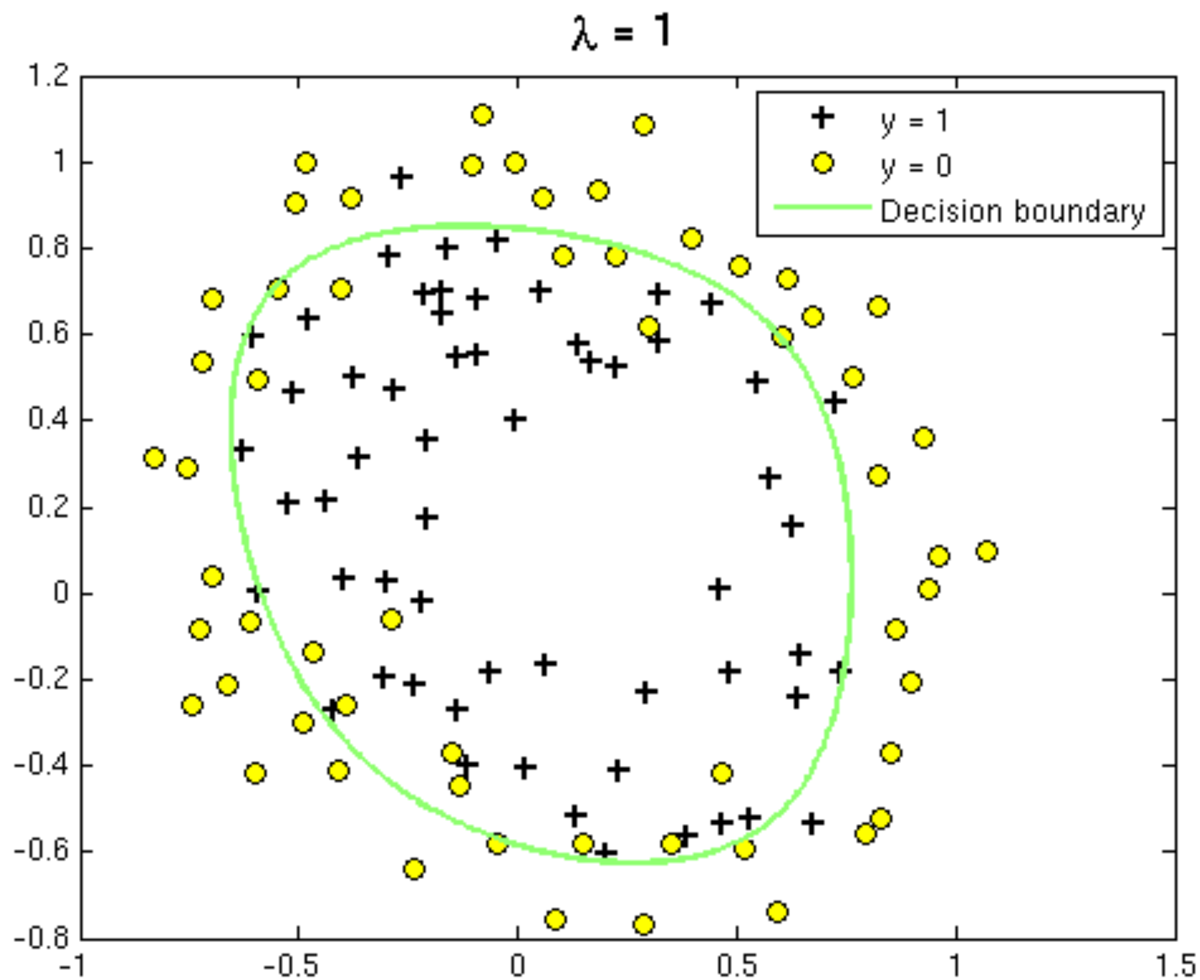


UNC CHARLOTTE

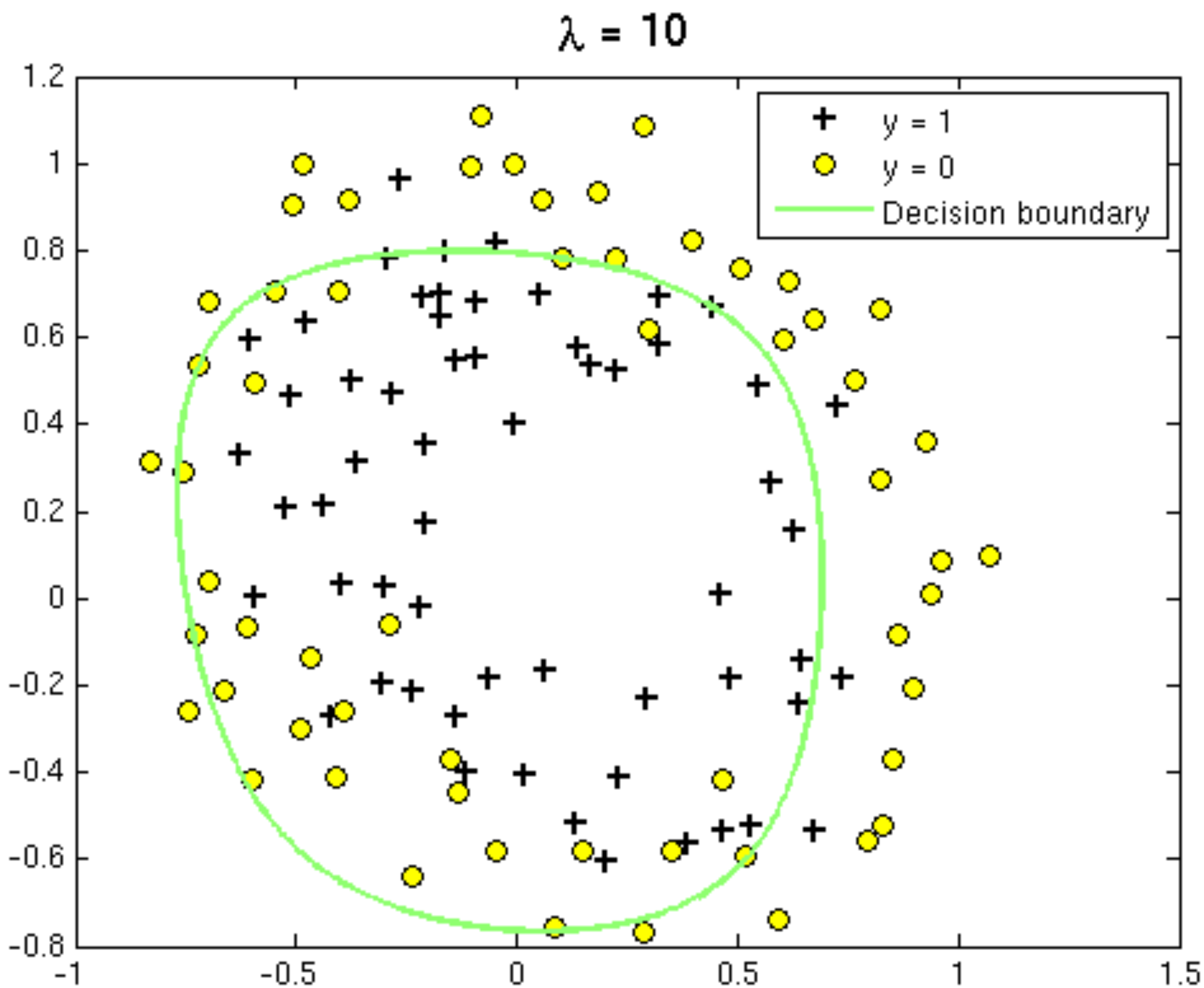
Regularization



Regularization



Regularization



Maybe too much regularization

UNC CHARLOTTE