



Artificial Learning Models

Lecture 3 : Logistic Regression

By : Prof. Lamri SAYAD

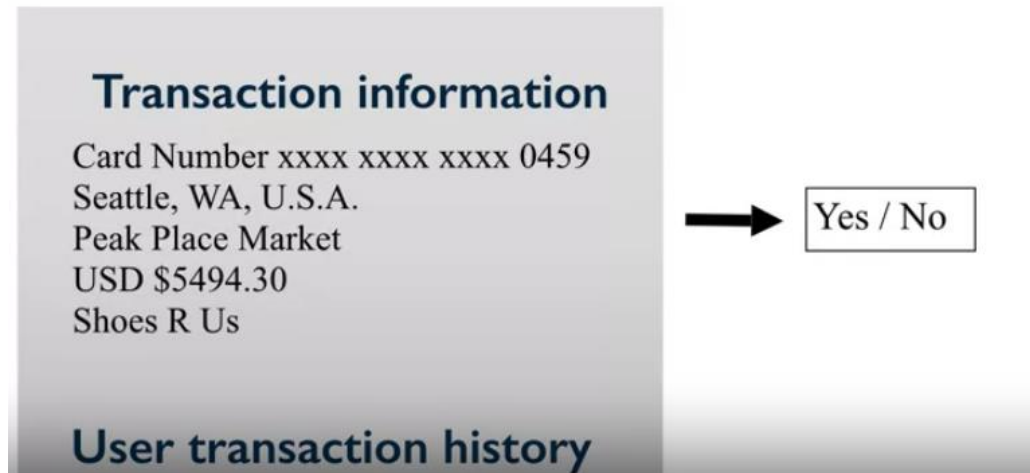
2024

Classification and Regression

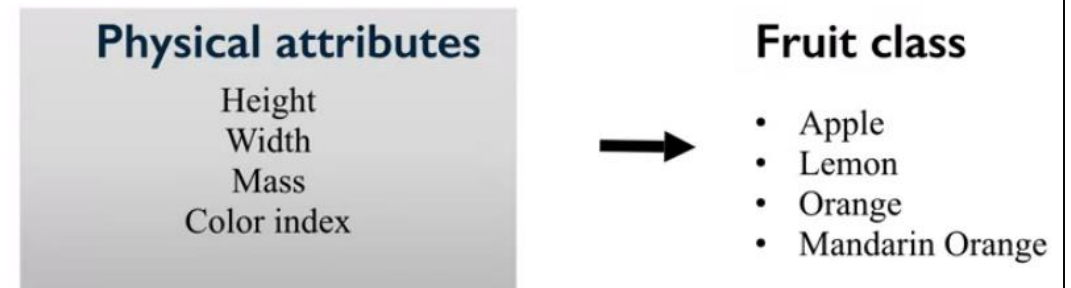
- Both classification and regression take a set of training instances and learn a mapping to a target value.
- For classification, the target value is a discrete class value
 - *Binary: target value is 0 (negative class) or 1 (positive class)*
 - e.g. detecting a fraudulent credit card transaction
 - *Multi-class: target value is one of a set of discrete values*
 - e.g. labelling the type of fruit from physical attributes

Binary classification vs Multi-class classification

- Binary classification : credit card fraud detection



- Multi-class classification: fruit recognition



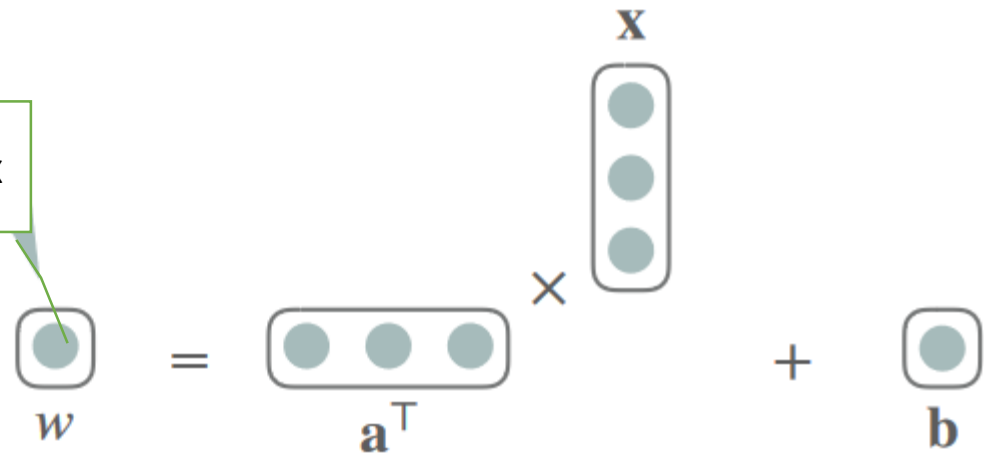
Binary Classification

- Input: $\mathbf{x} \in \mathbb{R}^d$
- Parameters: $\theta = \{\mathbf{a}, b\}$ avec $\mathbf{a} \in \mathbb{R}^d$, $b \in \mathbb{R}$
- Output: $y \in \{0,1\}$ ou $y \in \{-1,1\}$

- Scoring function

$$s_{\theta}(\mathbf{x}) = s(\mathbf{x}; \theta) = \langle \mathbf{a}, \mathbf{x} \rangle + b$$

Score associated to x

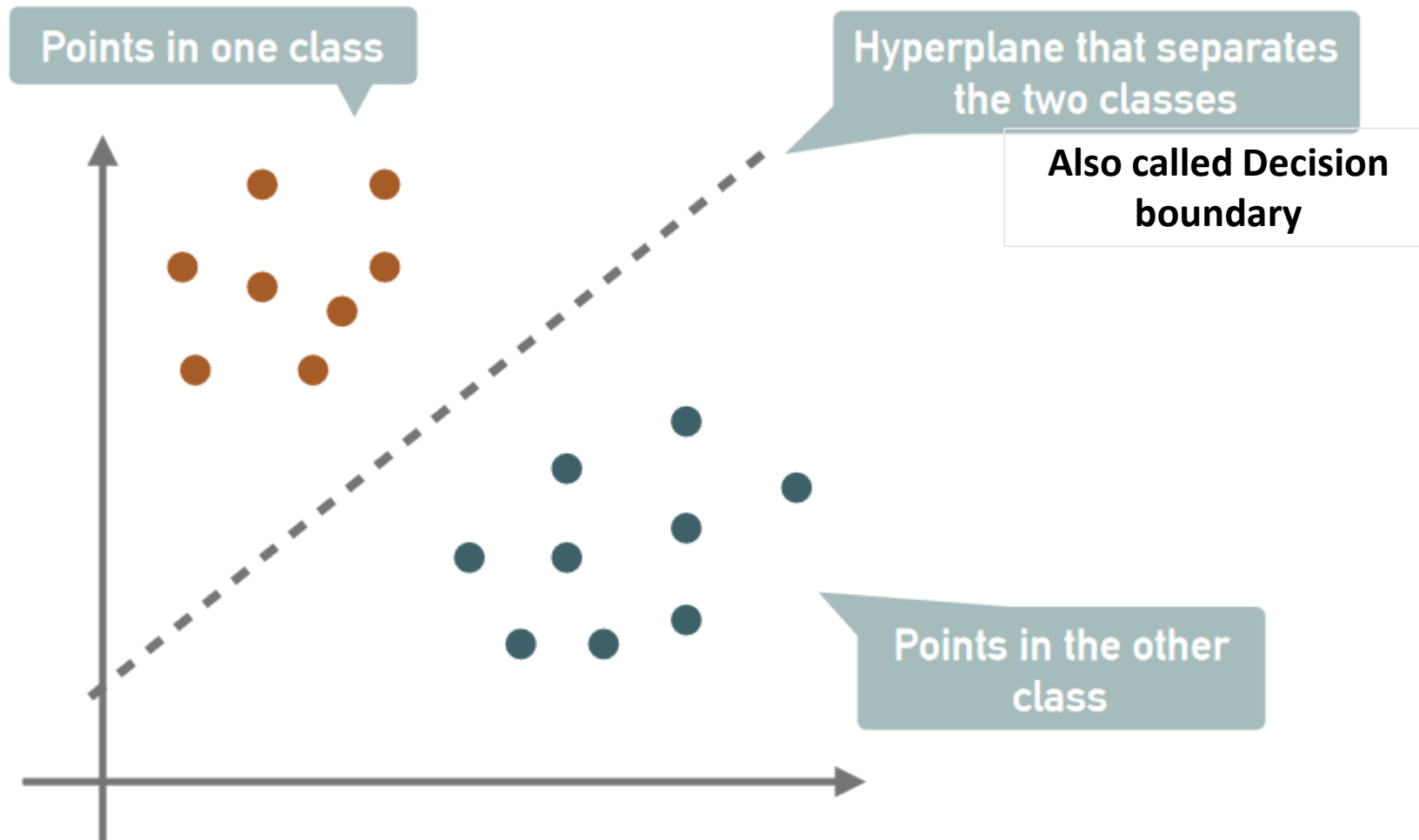


Binary Classification

- Prediction function

$$\hat{y}(w) = \begin{cases} 1 & \text{if } w \geq 0, \\ 0 & \text{otherwise.} \end{cases} \quad \text{ou} \quad \hat{y}(w) = \begin{cases} 1 & \text{if } w \geq 0, \\ -1 & \text{otherwise.} \end{cases}$$

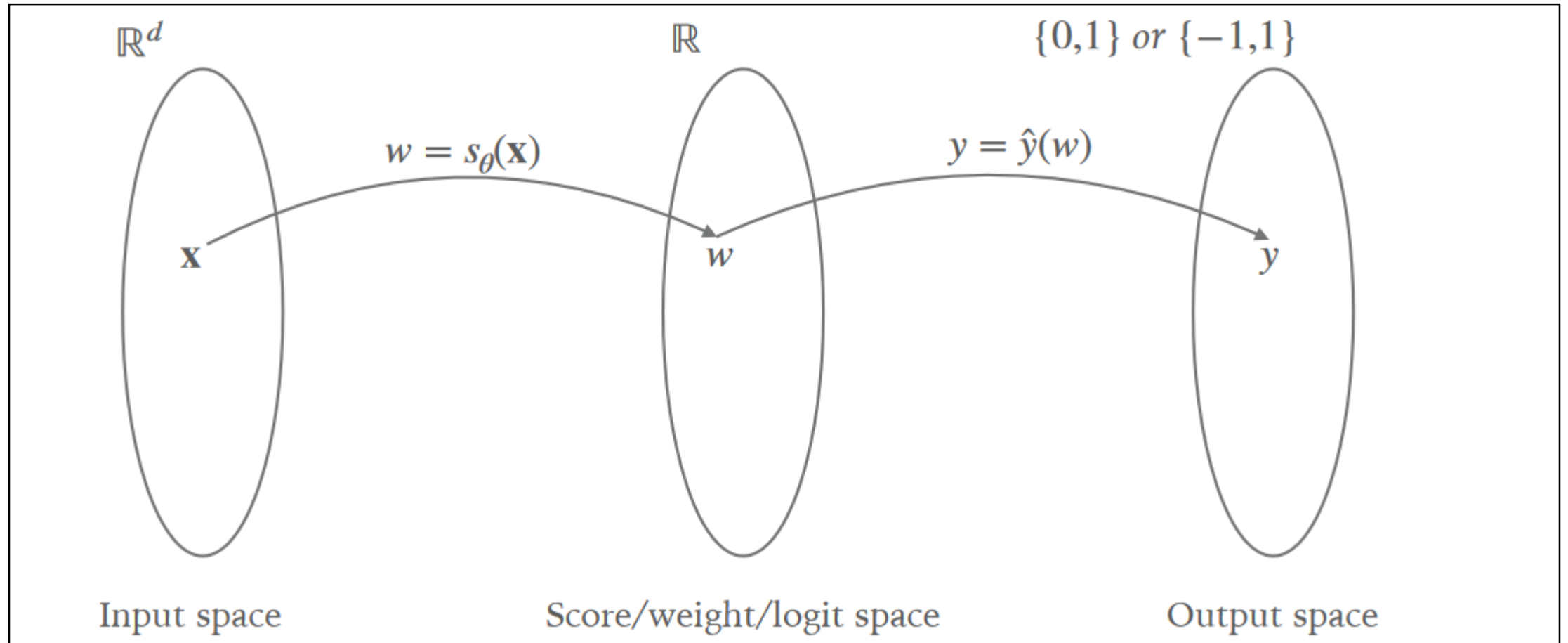
Binary Linear Classifier : Intuition



Binary Linear Classifier Definition

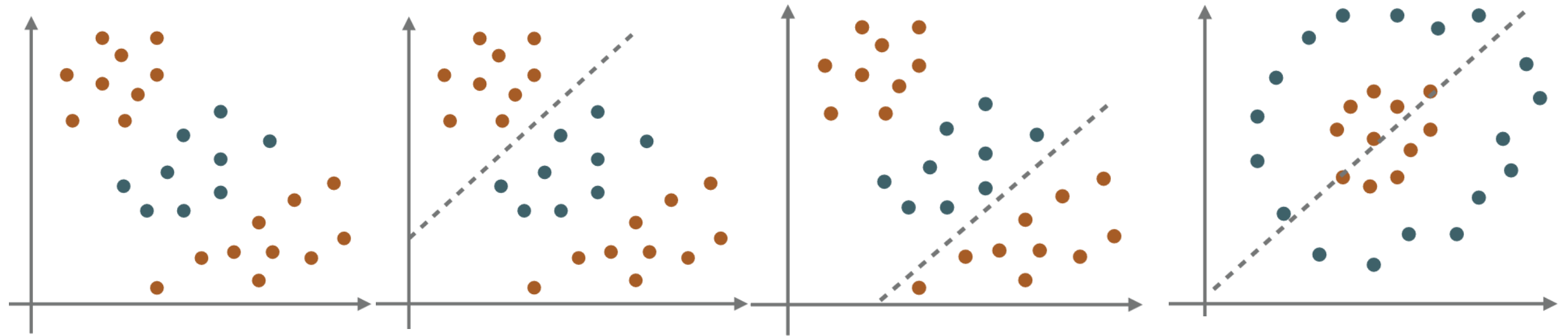


Binary Classification

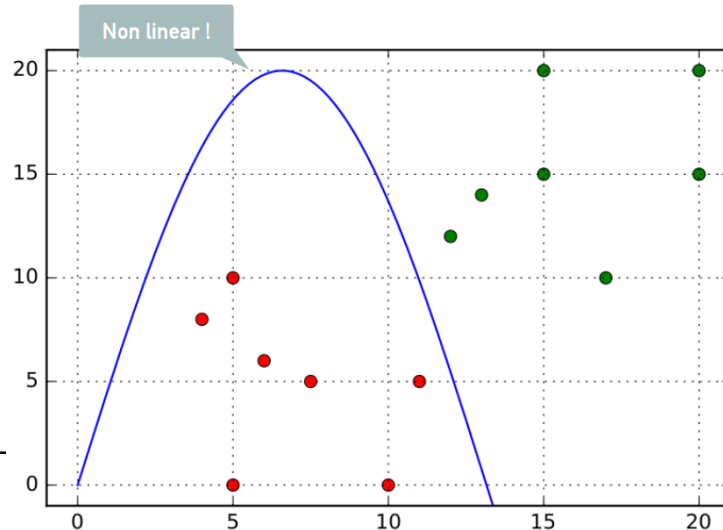
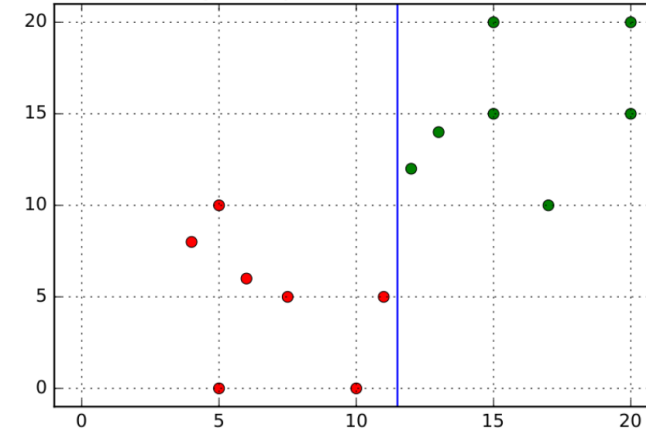
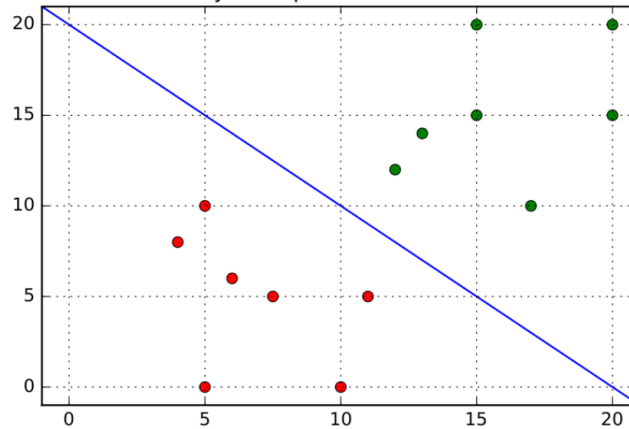
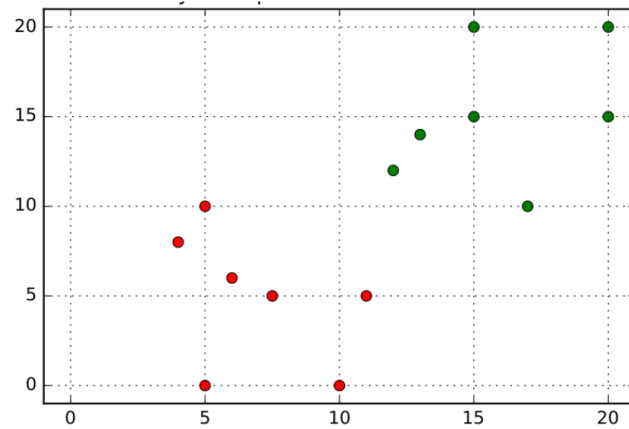


Problematic Cases

- Can we always find a hyperplane that separates classes? NO
- Can we characterize formally in which cases we can? YES



How to separate the data ?



In practice

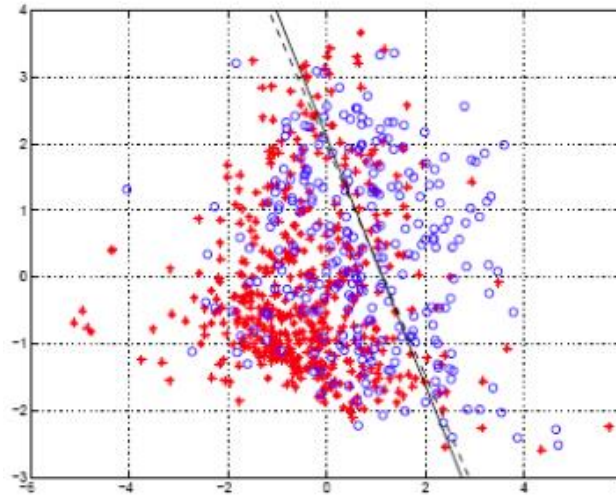
➤ Data is not linearly separable (i.e. such a hyperplane does not exist)

Regression or classification ?

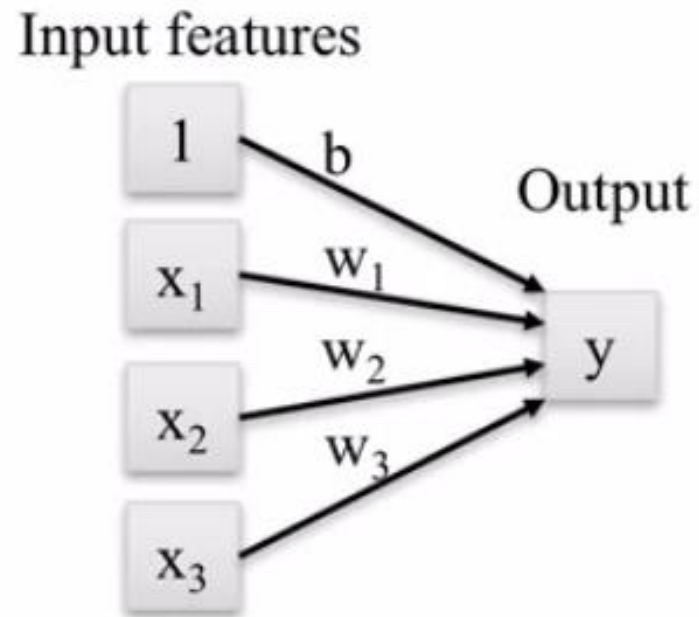
- Logistic regression is not a regression task
 - It is a classification task
- used when the dependent variable(target) is categorical.
- example,
 - To predict whether an email is spam (1) or (0)
 - Whether the tumor is malignant (1) or not (0)

Classification Based on Probability

- Takes a probabilistic approach to learning discriminative functions (i.e., a classifier)
- Instead of just predicting the class, give the probability of the instance being that class
 - i.e., learn $p(y \mid x)$



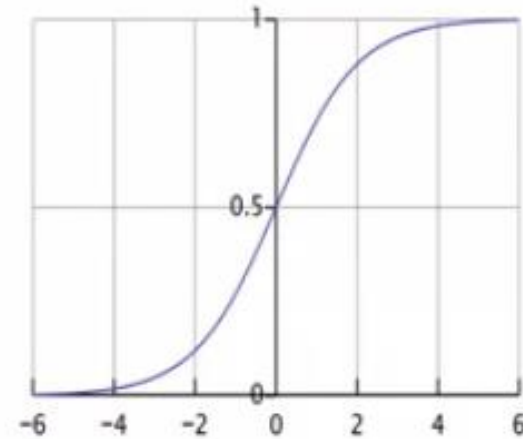
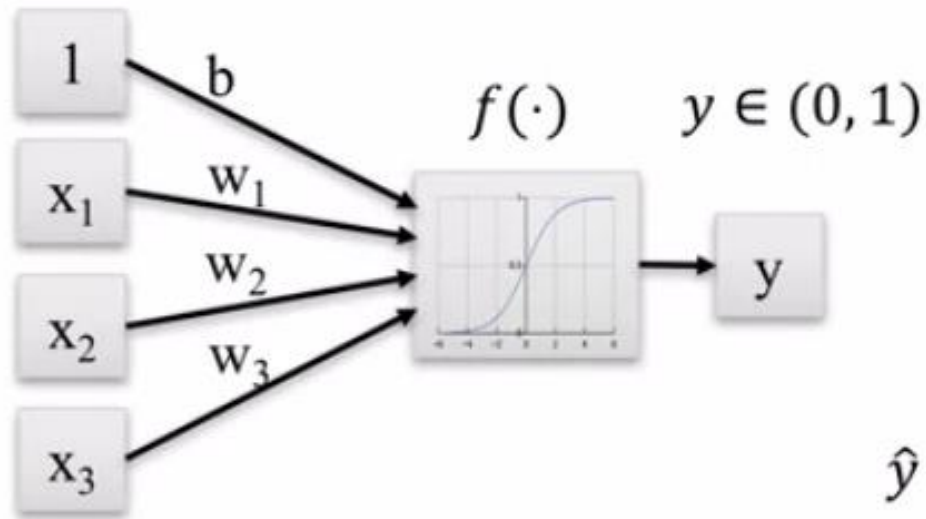
Linear Regression : Reminder



$$\hat{y} = \hat{b} + \hat{w}_1 \cdot x_1 + \cdots \hat{w}_n \cdot x_n$$

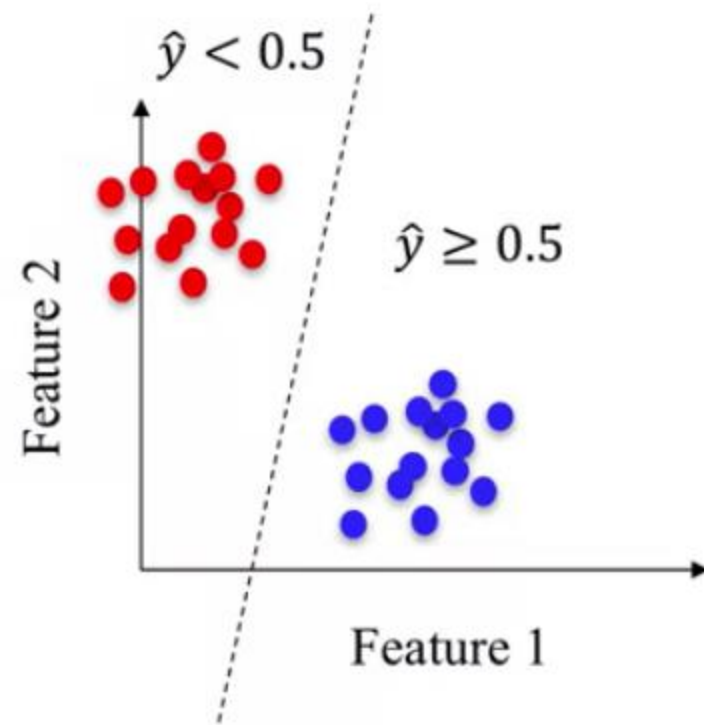
Linear models for classification : Logistic Regression

Input features

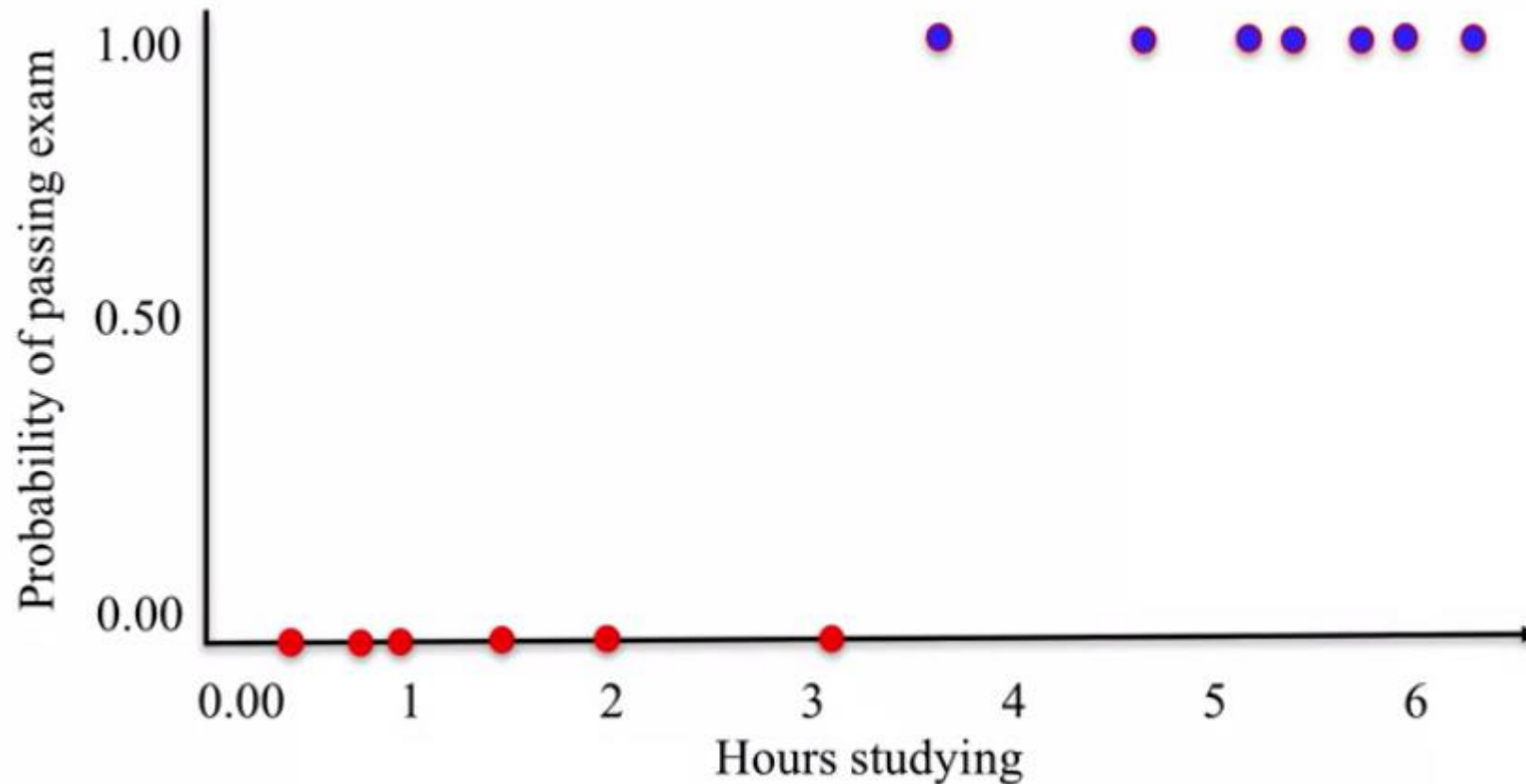


$$\hat{y} = \text{logistic}(\hat{b} + \hat{w}_1 \cdot x_1 + \cdots \hat{w}_n \cdot x_n)$$

$$= \frac{1}{1 + \exp[-(\hat{b} + \hat{w}_1 \cdot x_1 + \cdots \hat{w}_n \cdot x_n)]}$$



Linear models for classification : Logistic Regression (Example)



Logistic Regression: The model

- Takes a probabilistic approach to learning discriminative functions (i.e., a classifier)

$h_{\theta}(x)$ should give $p(y = 1 \mid x; \theta)$

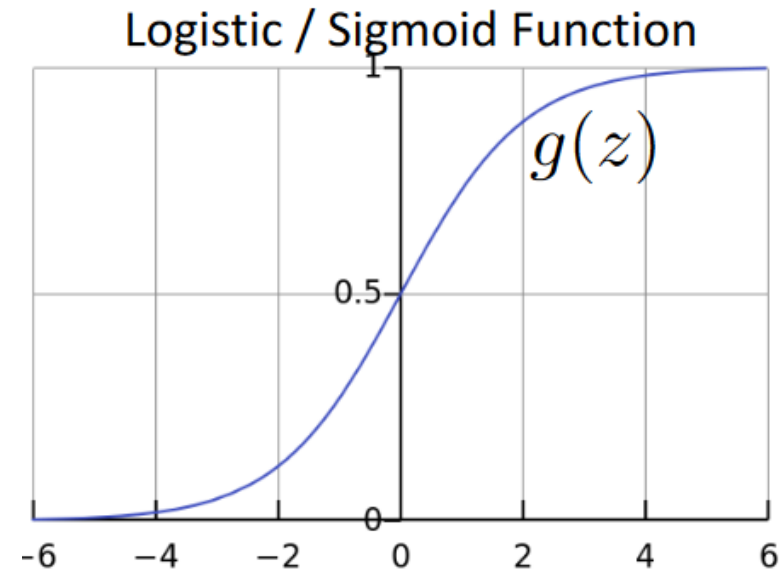
– Want $0 \leq h_{\theta}(x) \leq 1$

- Logistic regression model:

$$h_{\theta}(x) = g(\theta^T x)$$

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

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



Interpretation of Hypothesis Output

$$h_{\theta}(\mathbf{x}) = \text{estimated } p(y = 1 \mid \mathbf{x}; \theta)$$

Example: Cancer diagnosis from tumor size

$$\mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ \text{tumorSize} \end{bmatrix}$$

$$h_{\theta}(\mathbf{x}) = 0.7$$

→ Tell patient that 70% chance of tumor being malignant

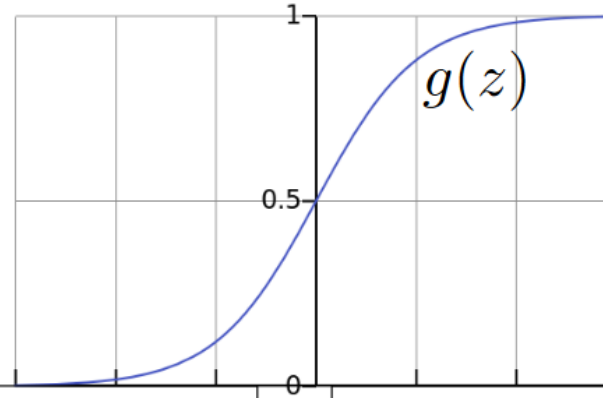
Note that: $p(y = 0 \mid \mathbf{x}; \theta) + p(y = 1 \mid \mathbf{x}; \theta) = 1$

Therefore, $p(y = 0 \mid \mathbf{x}; \theta) = 1 - p(y = 1 \mid \mathbf{x}; \theta)$

Logistic Regression

$$h_{\theta}(\mathbf{x}) = g(\theta^{\top} \mathbf{x})$$

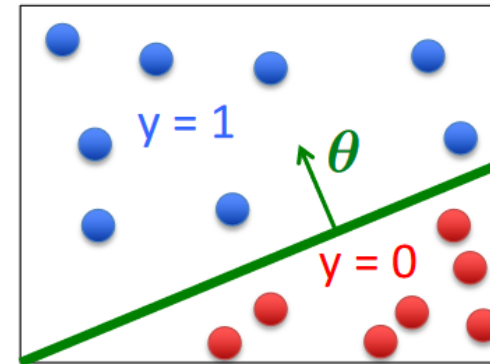
$$g(z) = \frac{1}{1 + e^{-z}}$$



$\theta^{\top} \mathbf{x}$ should be large negative values for negative instances

$\theta^{\top} \mathbf{x}$ should be large positive values for positive instances

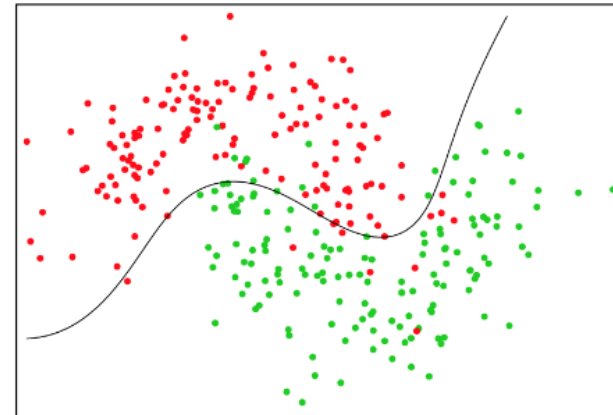
- Assume a threshold and...
 - Predict $y = 1$ if $h_{\theta}(\mathbf{x}) \geq 0.5$
 - Predict $y = 0$ if $h_{\theta}(\mathbf{x}) < 0.5$



Logistic Regression: Non-Linear decision boundary

- Can apply basis function expansion to features, same as with linear regression

$$\mathbf{x} = \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix} \rightarrow \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1 x_2 \\ x_1^2 \\ x_2^2 \\ x_1^2 x_2 \\ x_1 x_2^2 \\ \vdots \end{bmatrix}$$



Logistic Regression: Cost (Objective) Function

- Shouldn't use squared loss as in linear regression:

$$J(\boldsymbol{\theta}) = \frac{1}{2n} \sum_{i=1}^n \left(h_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) - y^{(i)} \right)^2$$

Logistic regression objective:

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

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

Logistic Regression: Cost (Objective) Function

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

- Cost of a single instance:

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

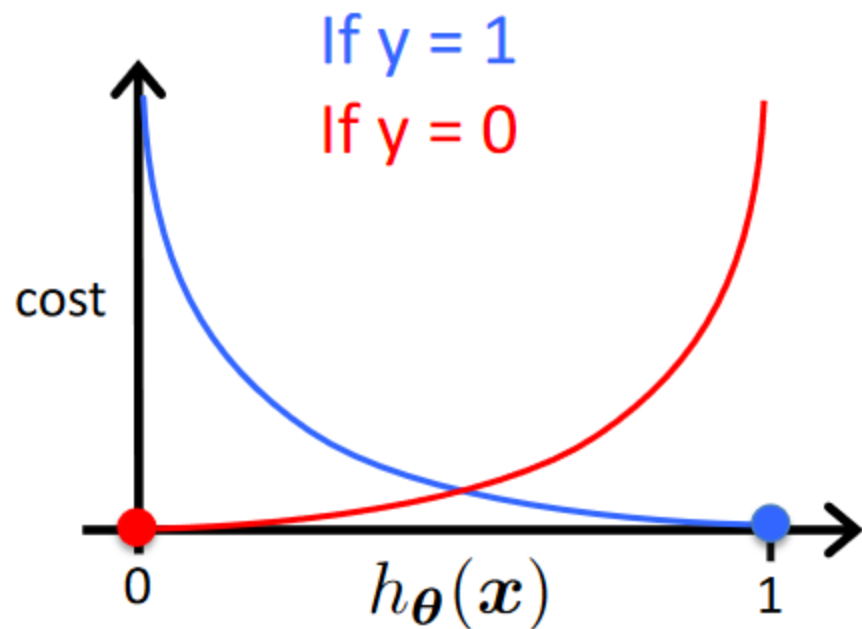
- Can re-write objective function as

$$J(\boldsymbol{\theta}) = \sum_{i=1}^n \text{cost}(h_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}), y^{(i)})$$

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

If $y = 0$

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

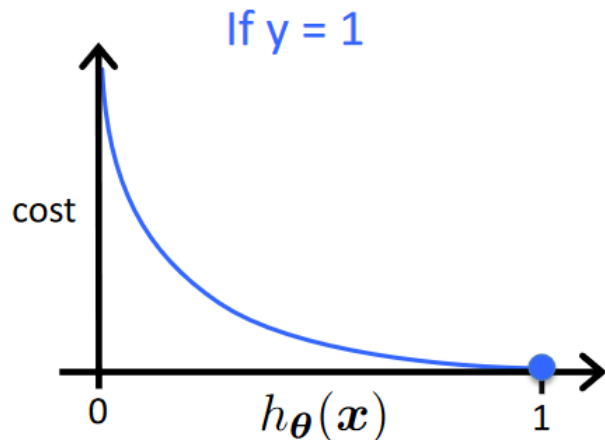


Logistic Regression: Cost (Objective) Function

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

If $y = 1$

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

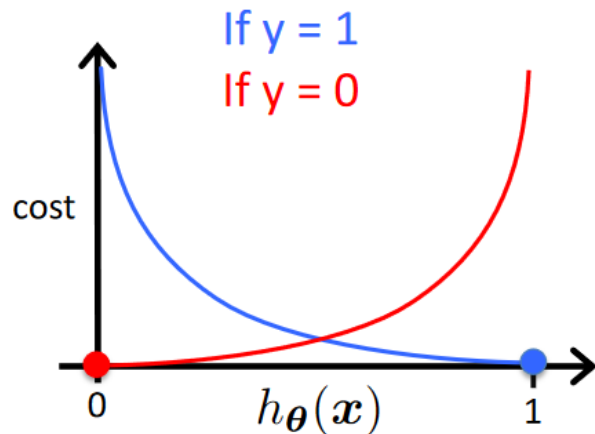


Logistic Regression: Cost (Objective) Function

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

If $y = 0$

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



Logistic Regression: Gradient descent (training)

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

- Initialize θ
- Repeat until convergence

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

simultaneous update
for $j = 0 \dots d$

Logistic Regression: Gradient descent (training)

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

- Initialize θ
- Repeat until convergence (simultaneous update for $j = 0 \dots d$)

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

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

Logistic Regression: Odds

- The odds are the ratio of the proportions for the two possible outcomes.

$$\text{odds} = \frac{p}{1 - p} = \frac{\text{probability of success}}{\text{probability of failure}}$$

- **log odds** or **logit**

Simple Logistic Regression Model

The statistical model for simple logistic regression is

$$\log\left(\frac{p}{1 - p}\right) = \beta_0 + \beta_1 x$$

where p is a binomial proportion and x is the explanatory variable. The parameters of the logistic model are β_0 and β_1 .