Classification & Linear Discriminants

Dr. Fayyaz Minhas

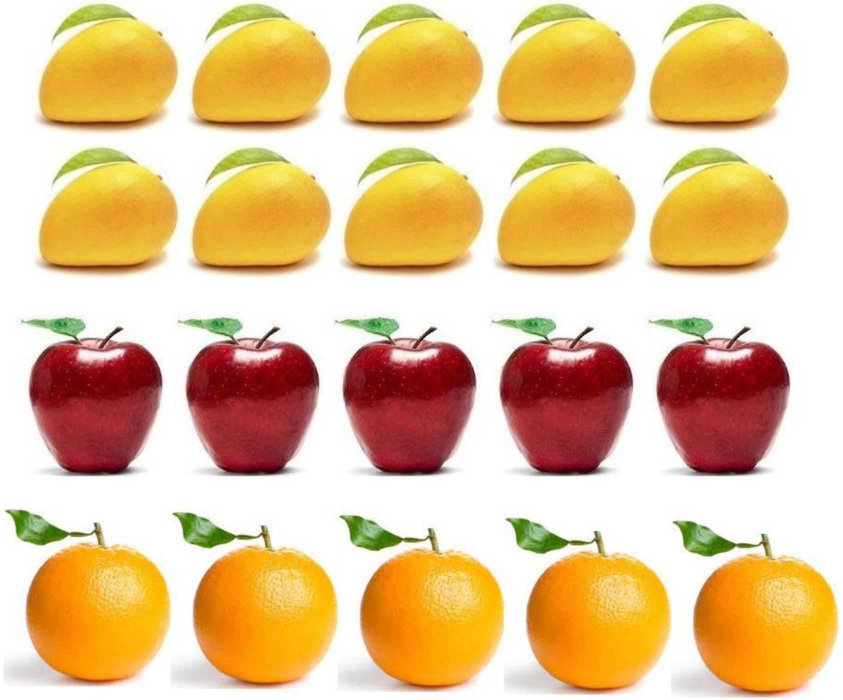
Department of Computer Science

University of Warwick

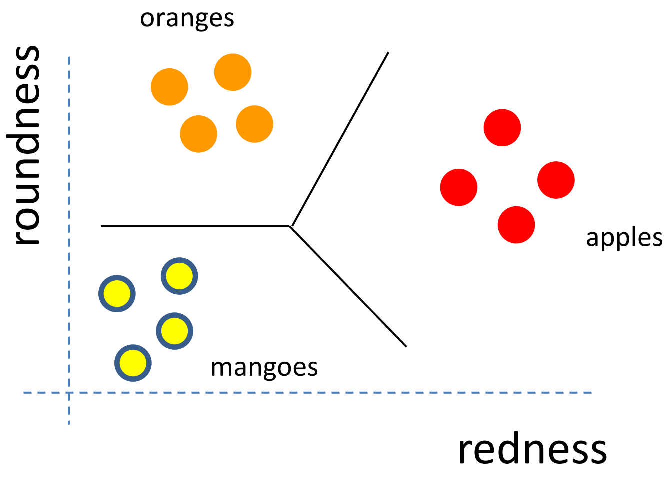
https://warwick.ac.uk/fac/sci/dcs/teaching/material/cs909/

Classification

- Given a set of data points (also called examples)
  - In solving a classification problem, the first step is to identify the unit of classification or "what is an example"
- Such that each example is represented by a feature vector
  - Representation of the example in terms of feature vector
- Assign a class label to each example
  - Such as apple, orange or mango



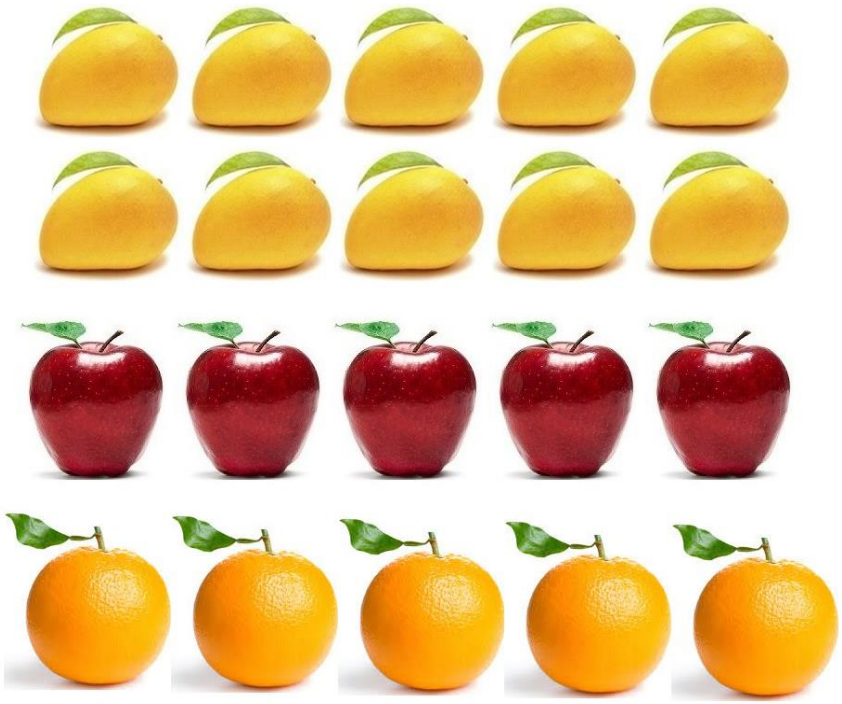
- Training Data
  - Set of examples for which both feature vectors and labels are available for "tuning"
    - Finding a mathematical function (called a classifier) that can be used to assign these labels
- Validation Data
  - Set of examples (with known labels) that are used to ensure that the classifier is expected to generalize to novel cases
- Test Data
  - (Ideally) Data for which the labels are not known and the ML model is used to find their labels



Classification

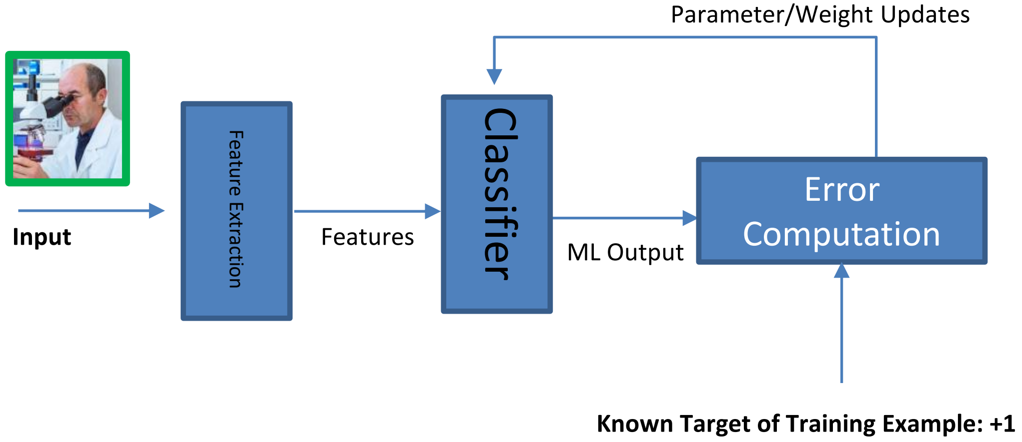
- The Objective of Classification is to assign class labels $y \in \{c_1, c_2, \ldots, c_M\}$
- to a given feature vector $\boldsymbol{x} = \begin{bmatrix} x^{(1)} \\ x^{(2)} \\ \vdots \\ x^{(d)} \end{bmatrix}$
- The classifier may use previously known and available training data
  - Good generalization, Good memorization
- The training data comprises of classified data points: $\mathrm{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}, \boldsymbol{x}_{i_{(d\times 1)}} = \begin{bmatrix} x_i^{(1)} \\ x_i^{(2)} \\ \vdots \\ x_i^{(d)} \end{bmatrix}$

$\mathrm{Y} = \{y_1, y_2, \ldots, y_N\}, y_i \in \{c_1, c_2, \ldots, c_M\}$

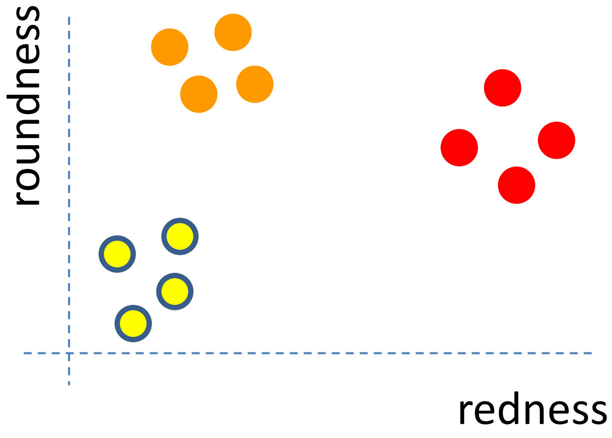


REO

- **R**epresentation
  - Represent examples in a feature space
    - Represent features: **x**
  - Define a classification function
    - Line: f(**x**;**w**) = $w_1x^{(1)}+w_2x^{(2)}+b = 0$



- **E**valuation
  - Define an error function
    - Misclassifications
- **O**ptimize
  - Reduce error by adjusting the parameters of the ML model (**w**)
- Real Test (Generalization)
  - How does it perform on unseen data?



Discriminant based classification

- In this type of classification, the objective is to learn a function or, in the case of more than 2 classes, a set of functions from training data which can generate decisions for test data such that the classes in the data can be separated
  - Class label assignment: $c(\boldsymbol{x}) = argmax_{k=1,\ldots,M}\ f_k(\boldsymbol{x})$
    - $f_k(\boldsymbol{x})$ tells you the '*k-classiness*' of an example $\boldsymbol{x}$
  - If M = 2
    - Choose class-1 if $f_1(\boldsymbol{x}) \geq f_2(\boldsymbol{x})$, i.e., $f_1(\boldsymbol{x}) - f_2(\boldsymbol{x}) \geq 0$
    - Otherwise assign it to class-2, i.e., $f_1(\boldsymbol{x}) - f_2(\boldsymbol{x}) < 0$
    - We can thus replace the two functions with a single function

$$f(\boldsymbol{x}) = f_1(\boldsymbol{x}) - f_2(\boldsymbol{x})$$

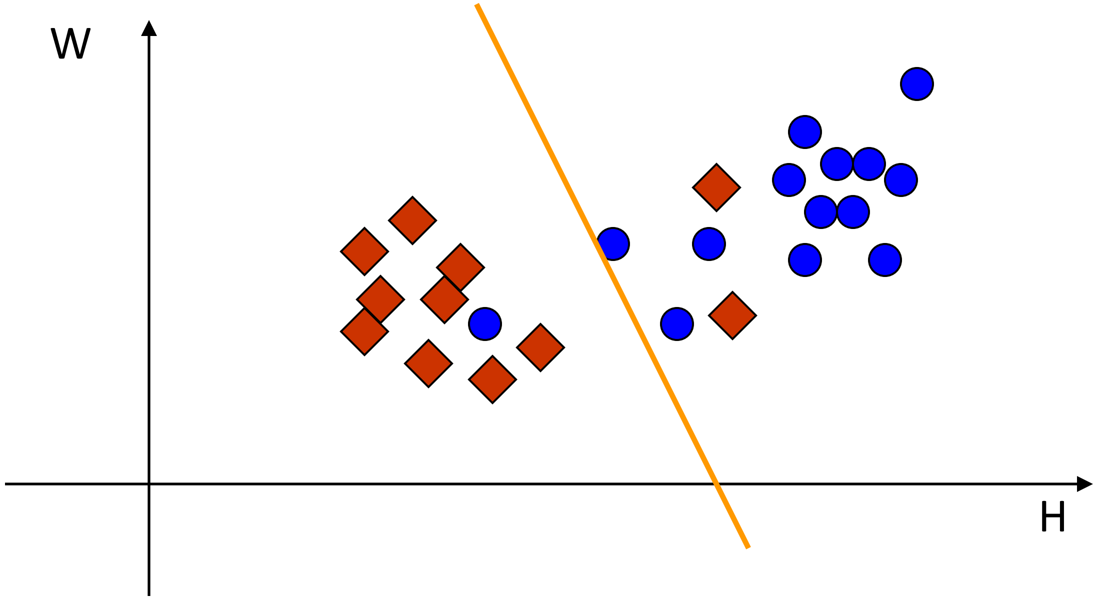
Assign to positive class if $f(\boldsymbol{x}) \geq 0$, otherwise negative

$f(\boldsymbol{x}) = 0$ separates the two classes and is called the discriminant

If the function(s) are linear, the classifier is called a linear discriminant

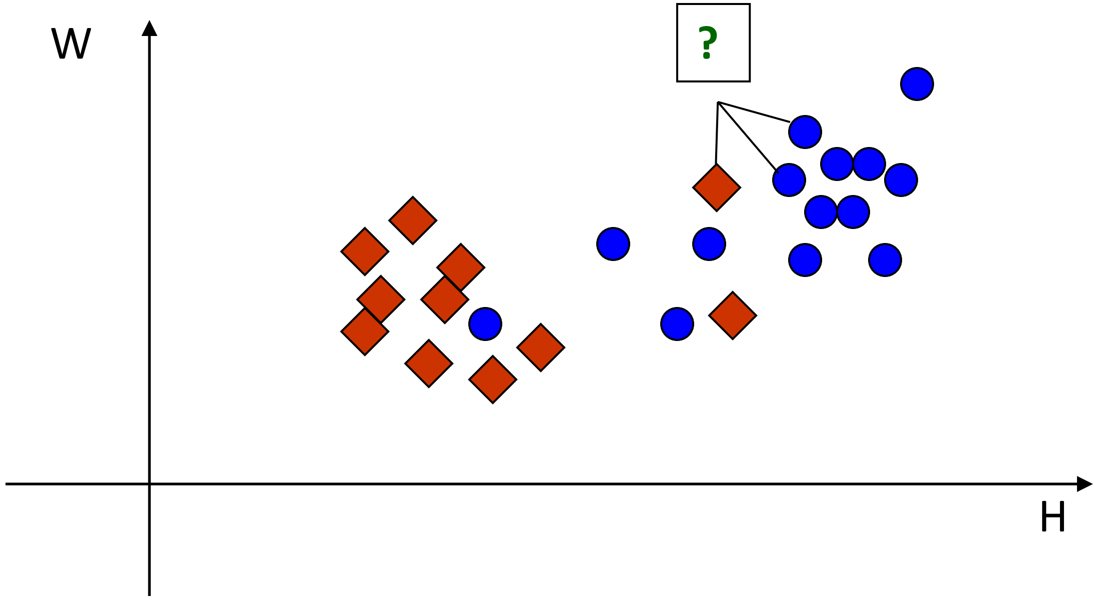
Classification Approaches: Supervised…

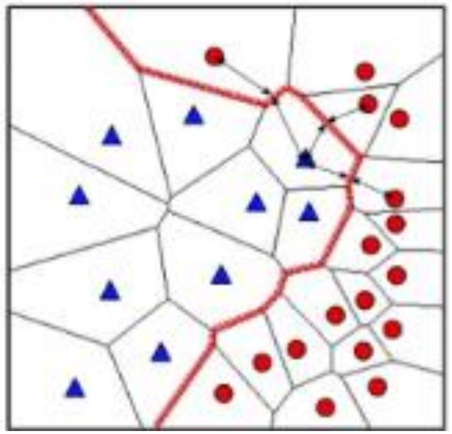
- Linear Classifier



Classification Approaches: Supervised

- Example (k=3)-Nearest Neighbor Classification

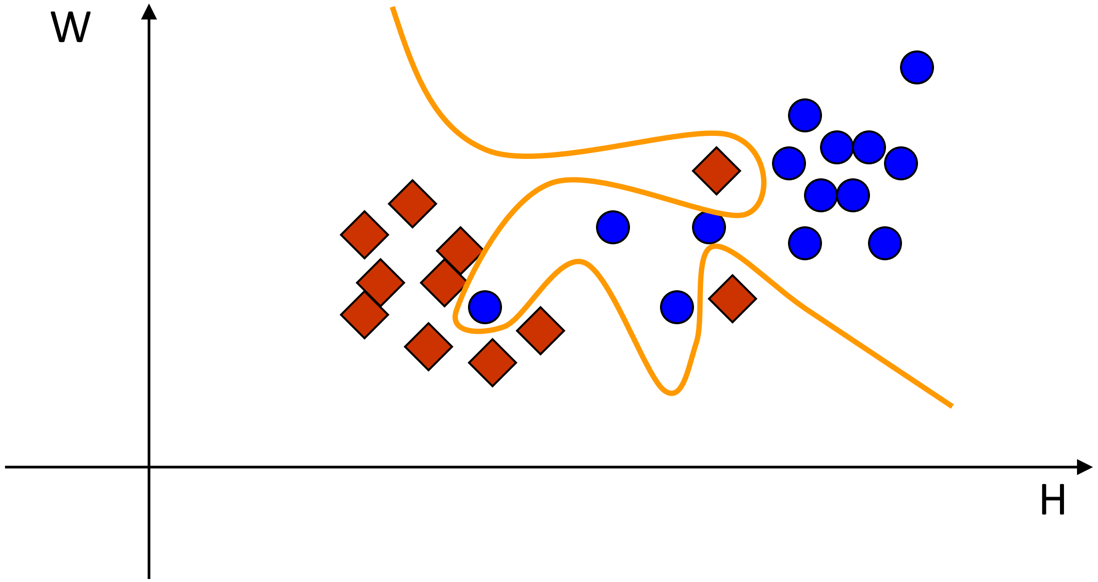




Demo: http://vision.stanford.edu/teaching/cs231n-demos/knn/

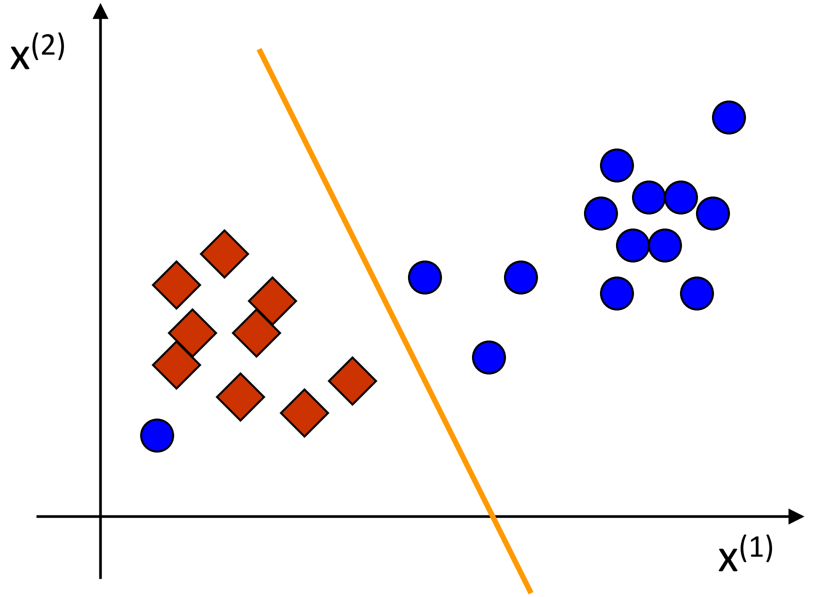
Classification Approaches: Supervised…

- Nonlinear Classification boundary



Linear Separability

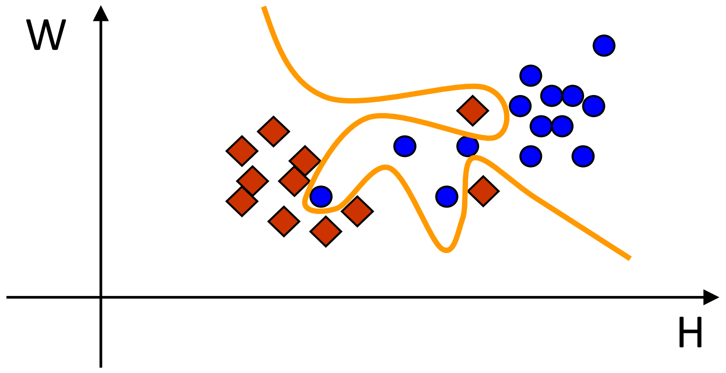
- If data points can be separated by a linear discriminant then that dataset/classification problem is called "linearly separable"



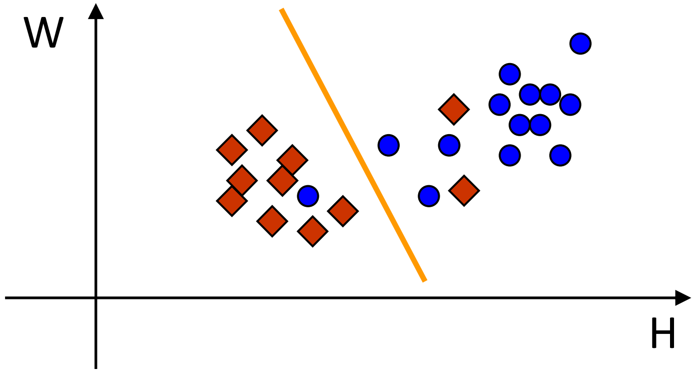
- Mathematically,
  - If there exists a linear function $f(\boldsymbol{x};\boldsymbol{w}) = w_1 x^{(1)} + w_2 x^{(2)} + \cdots + w_d x^{(d)} + b = 0$ such that
    - If $y_i = +1$, then $f(\boldsymbol{x_i};\boldsymbol{w}) > 0$
    - And If $y_i = -1$, then $f(\boldsymbol{x_i};\boldsymbol{w}) < 0$

Classification Approaches: Supervised…

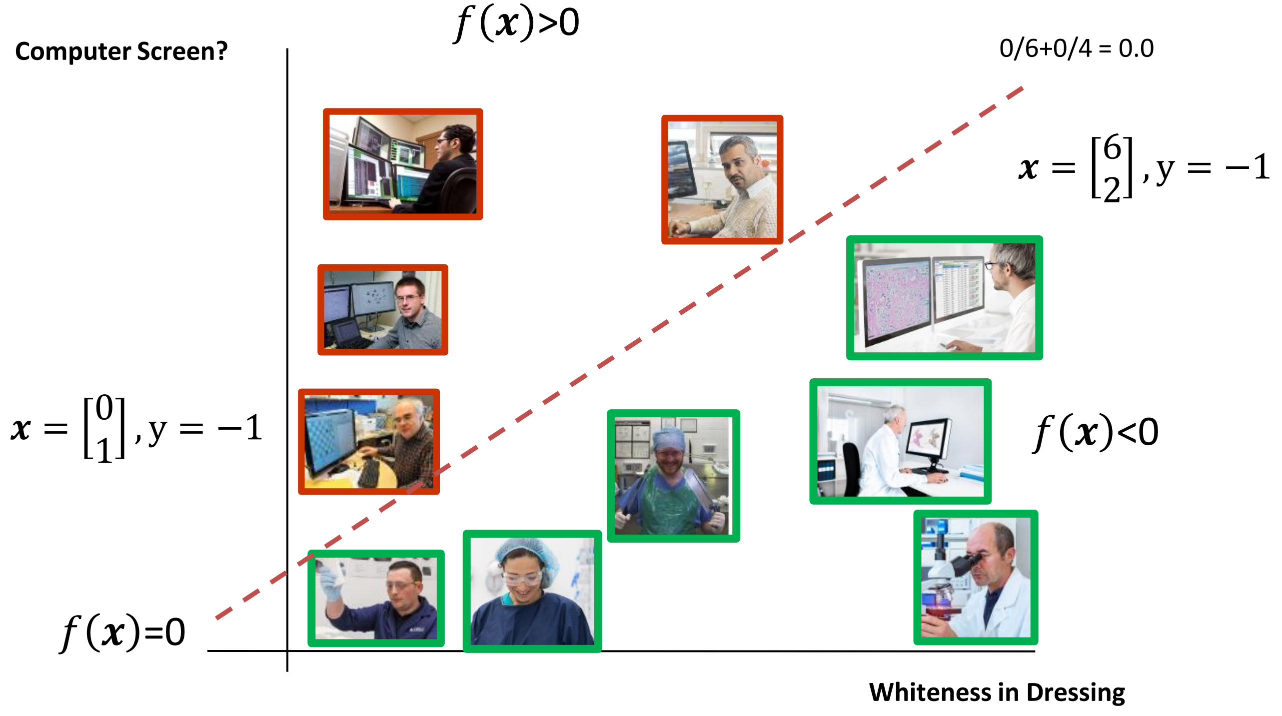
- Generalization vs. Memorization
  - A particular issue in classification is the tradeoff between memorization vs. generalization
    - Remembering everything is not learning
    - The true test of learning is handling similar but unseen cases



Has great memorization but may generalize poorly



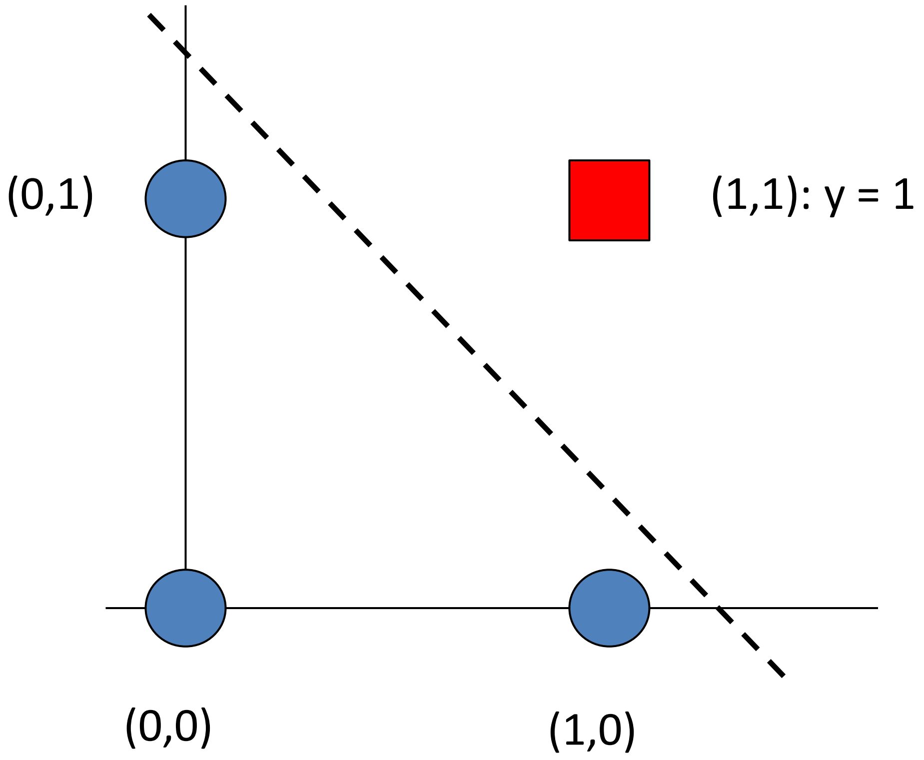
Has lesser memorization but may generalize better



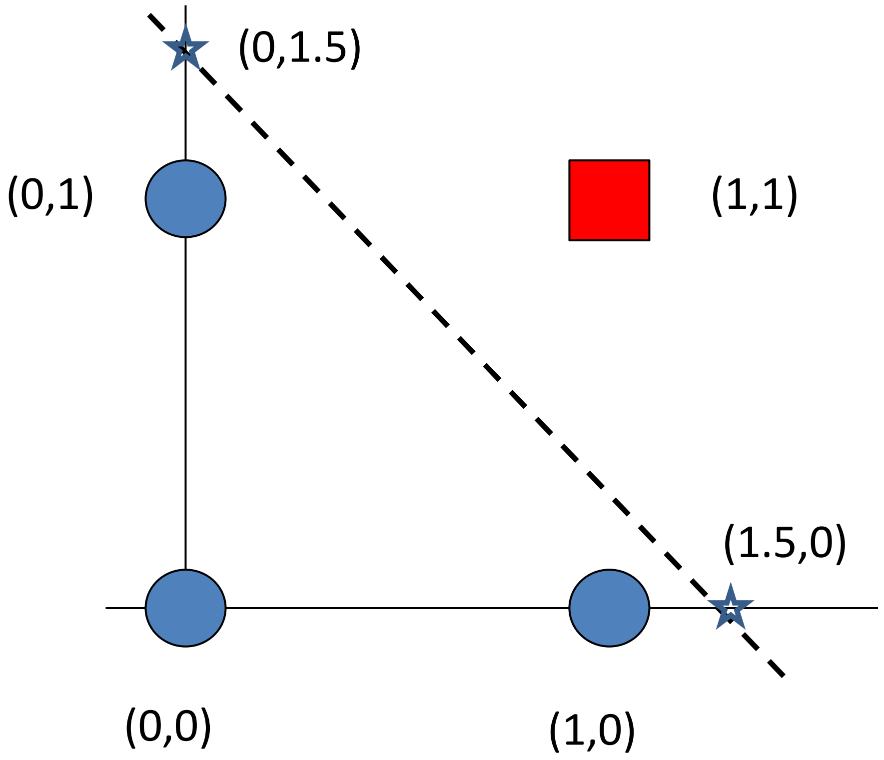
$$\boldsymbol{x} = \begin{bmatrix} x^{(1)} \\ x^{(2)} \end{bmatrix}$$

Another example

$f(\mathbf{x};\mathbf{w}) = w_1 x^{(1)} + w_2 x^{(2)} + b = 0$



Example (Graphical Approach)



$f(\mathbf{x};\mathbf{w}) = w_1x^{(1)}+w_2x^{(2)}+b = 0$

$w_1(1.5)+w_2(0)+b = 0$

$b = -1.5w_1$

$w_1(0)+w_2(1.5)+b = 0$

$b = -1.5w_2$

If I set $w_1 = 1.0$

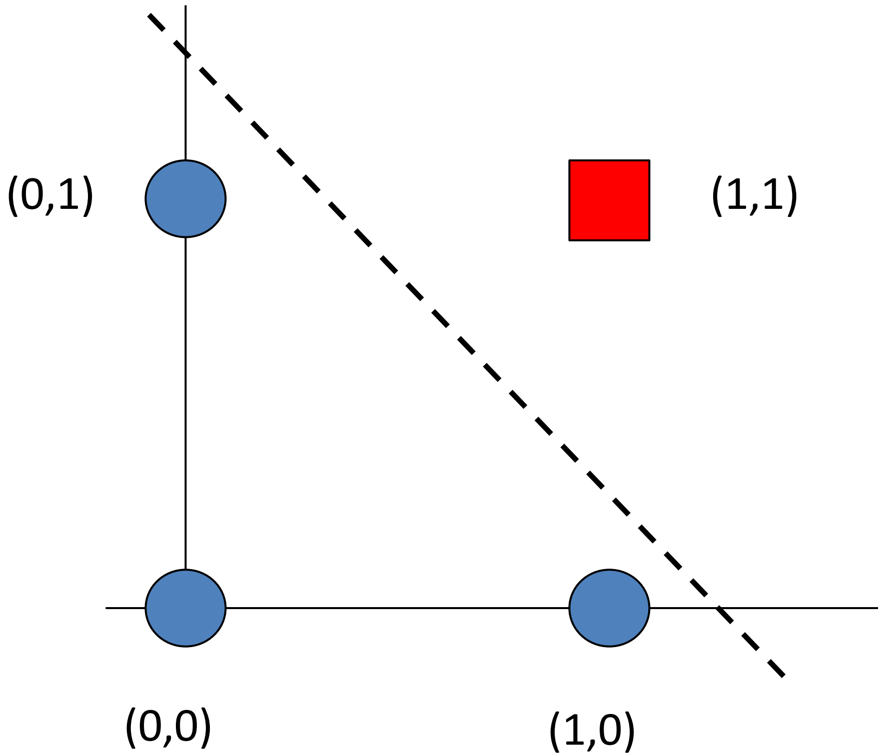
$b = -1.5$

$w_2 = 1.0$

$f(\mathbf{x};\boldsymbol{\theta}) = x^{(1)}+x^{(2)} -1.5$

(1,1)→0.5
(1,0) →-0.5, (0,1) →-0.5, (0,0) →-1.5

Example: Another Way (Algebraic Constraint Satisfaction)



$f(\mathbf{x};\mathbf{w}) = w_1x^{(1)}+w_2x^{(2)}+b = 0$

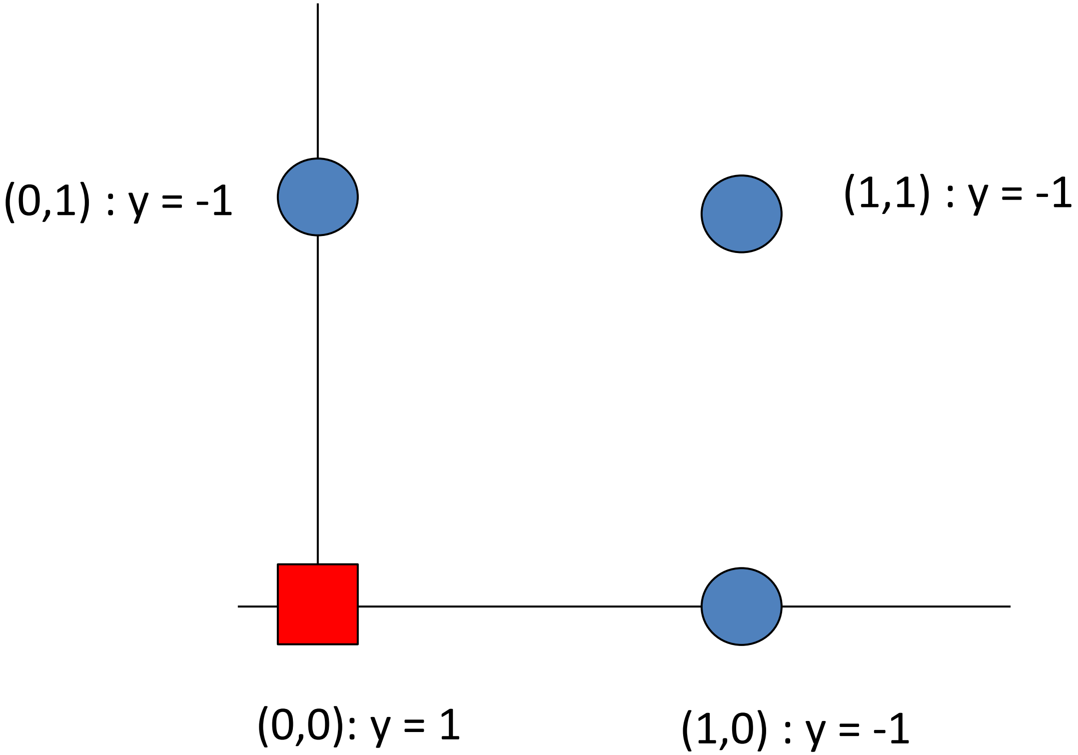
(1,1): $w_1(1.0)+w_2(1.0)+b > 0$
(1,0): $w_1(1.0)+w_2(0.0)+b < 0$
(0,1): $w_1(0.0)+w_2(1.0)+b < 0$
(0,0): $w_1(0.0)+w_2(0.0)+b < 0$

$w_1+w_2+b > 0$
$w_1+b < 0$
$w_2+b < 0$
$b < 0$

$b = -1.5$
$w_1= 1.0$
$w_2= 1.0$

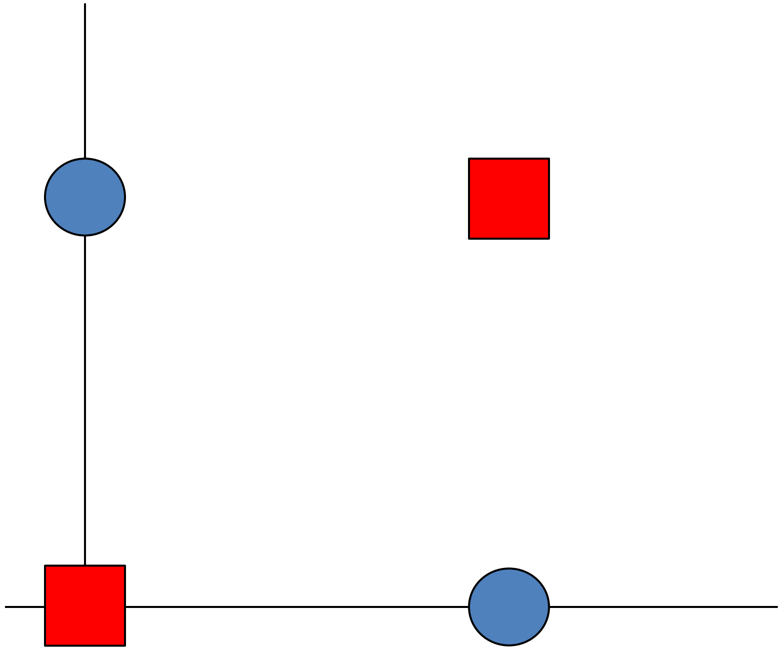
Exercise

- Is this problem linearly separable?



Let's talk about: Linear Separability

- What about this one?



(1,1): $w_1(1.0)+w_2(1.0)+b > 0$
(1,0): $w_1(1.0)+w_2(0.0)+b < 0$
(0,1): $w_1(0.0)+w_2(1.0)+b < 0$
(0,0): $w_1(0.0)+w_2(0.0)+b > 0$

What about this one?

- (0,0,0): -1
- (1,0,0): +1
- (0,1,0): -1
- (0,0,1):+1
- (1,0,1): +1
- (1,1,0): +1
- (0,1,1): +1
- (1,1,1): +1

$f(\mathbf{x};\mathbf{w}) = w_1x^{(1)}+w_2x^{(2)}+w_3x^{(3)}+b = 0$

End of Lecture

We want to make a machine that will be proud of us.

- Danny Hillis