Lecture 1

Outline of the Course

- **What is learning?**
- **Can we learn?**
- **How to do it?**
- **How to do it well?**

1. The Learning Problem
2. Is Learning Feasible?
3. The Linear Model
4. Error and Noise
5. Training versus Testing
6. Theory of Generalization
7. The VC Dimension
8. Bias-Variance Tradeoff
9. The Linear Model II
10. Neural Networks
11. Overfitting
12. Regularization
13. Validation
14. Support Vector Machines
15. Kernel Methods
16. Radial Basis Functions
17. Three Learning Principles
18. Epilogue

Lecture 1: The Learning Problem

Outline

- Example of machine learning
- Components of Learning
- A simple model
- Types of learning
- Puzzle

Example: Predicting how a viewer will rate a movie

- 10% improvement of recommendation system = 1 million dollar prize.
- **The essence of machine learning**:
  - A pattern exists
  - We cannot pin it down mathematically
  - We have data on it.

Movie Rating - A solution

Look at each viewer as a vector in some feature space.

For e.g. viewer1 = (likes comedy?, likes action?,...,likes Tom Cruise?)
movie1 = (comedy,not action,...., Tom Cruise is lead hero)
rating = f(viewer1 , movie1)

The Learning approach

$\{v_i\}_i$ - viewers in R^n.
$\{m_j\}_j$ - movies in R^n.

We **do not** know the coordinates of these vectors. But we know the `rating` given by viewer i for the movie j. A machine learning system will learn these vectors from these rating.

Components of Learning

- **Input**: $x \in \mathbb{R}^d$
- **Output**: y
- **Target function**: $f : X \to Y$

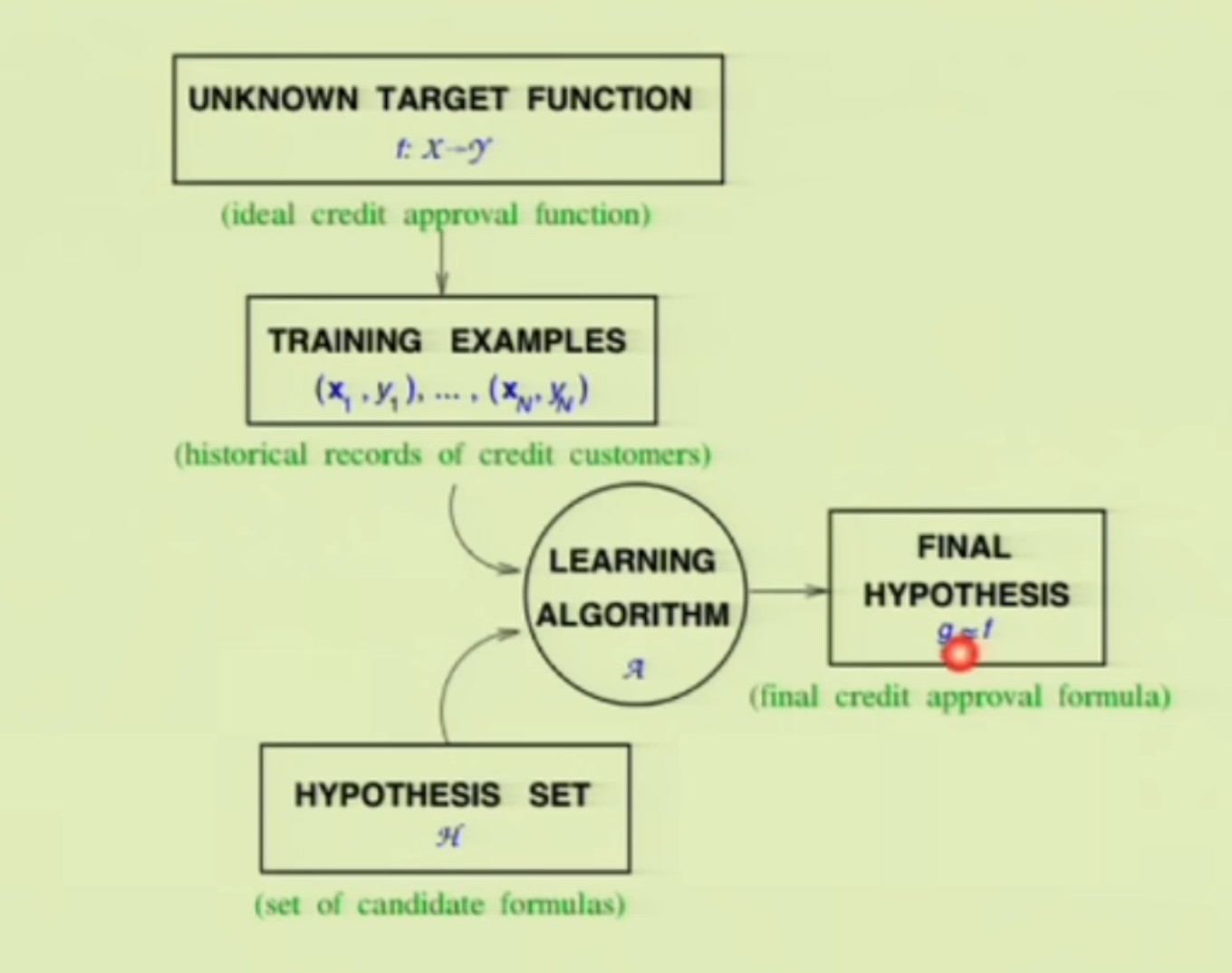
we do not know the target function.

We are going to use the data to learn the target function

- **Data**$(x_1, y_1), \ldots, (x_N, y_N)$

Use data to learn the **Hypothesis** which supposedly approximates the target function.

- **Hypothesis**: $g : X \to Y$
- **Learning Algorithm**: Takes as input the data and outputs the hypothesis h from the hypothesis set H that best approximates the target function.



Solution Components

Things we have no control over: Target function and the training data.

What we can control: The Learning Model

- Hypothesis set $H = \{h\}$
- The Learning Algorithm A.

A simple hypothesis set - the 'perceptron'

For input $x = (x_1, \ldots, x_d)$,

output 1 if $\sum_i w_i x_i > t$
output -1 otherwise.

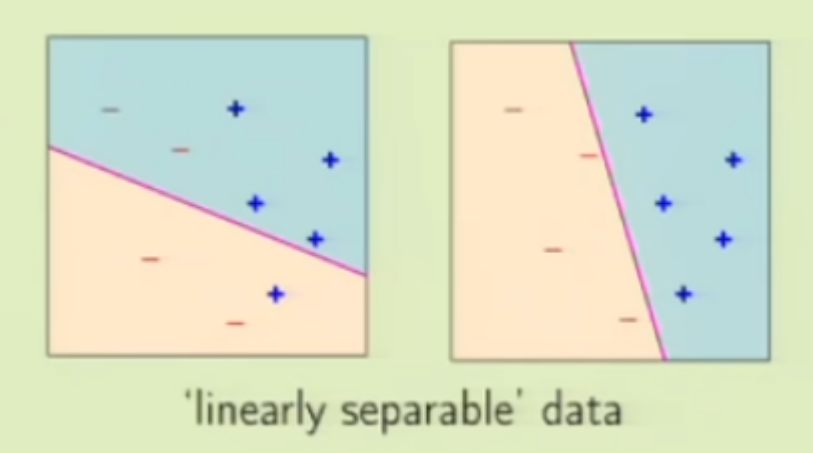
The linear formular $h \in H$ can be written as:

$$h(x) = \text{sign}(w^T x - t)$$

.

Each $h \in H$ is defined by the choice of the w_is and the threshold t.

Let us assume that the data is **linearly separable**.



To simplify notations, we introduce an artificial coordinate $x_0 = 1$ and $w_0 = -t$. Now h can be written as:

$$h(x) = \text{sign}(w^T x)$$

.

A simple learning algorithm - PLA

The perceptron implements

$$h(x) = \text{sign}(w^T x)$$

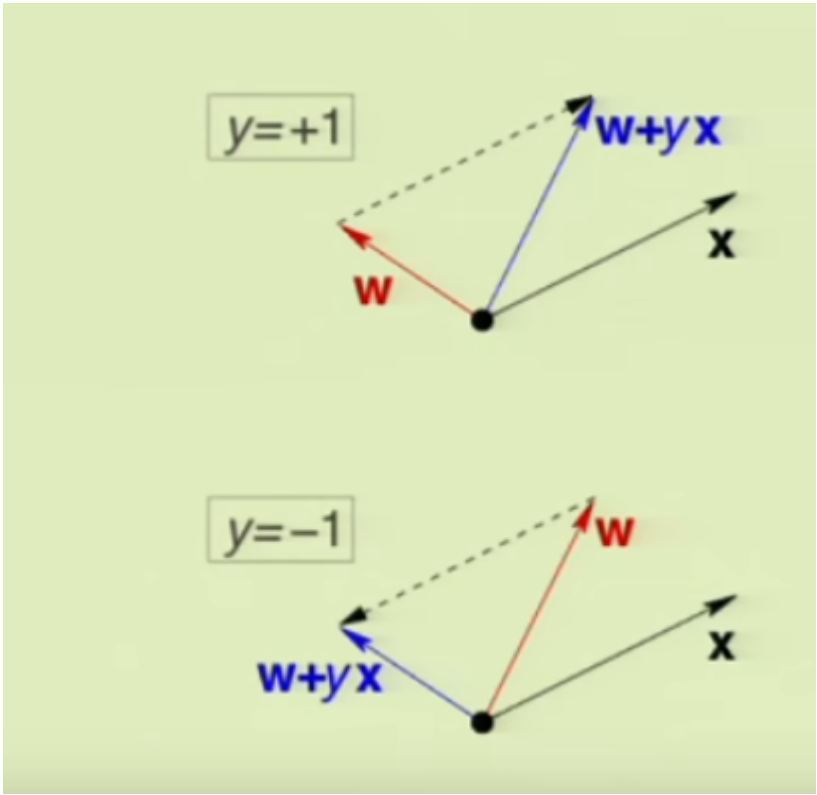
Given the training set:

$$(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$$

.

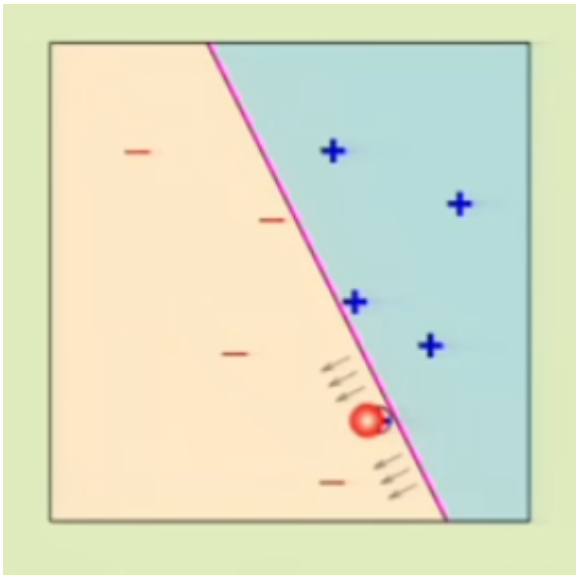
pick a **misclassified** point x_i, i.e $\text{sign}(w^T x_i) \neq y_i$ and adjust w so that this particular misclassified point is correctly classified.

$$w \leftarrow w + y_i x_i$$



Iterations of PLA

- One iteration of PLA



- for iteration $t = 1, 2, 3, \ldots$ pick a misclassified from the training data and run PLA on it.

Questions:

1. Will we converge?
2. If yes, when?

Basic Premise of learning

"using a set of observations to uncover an underlying process"

Statistics:

- underlying process: Probability Distribution
- observations: Samples generated by the Distribution.

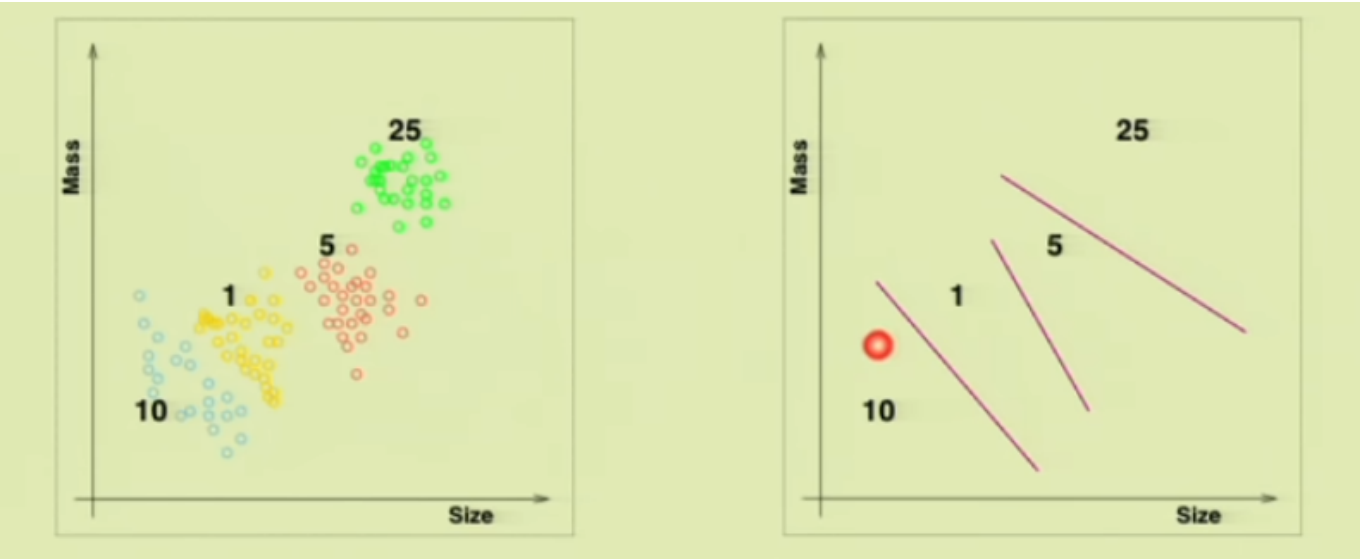
Types of Learning

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

Supervised Learning

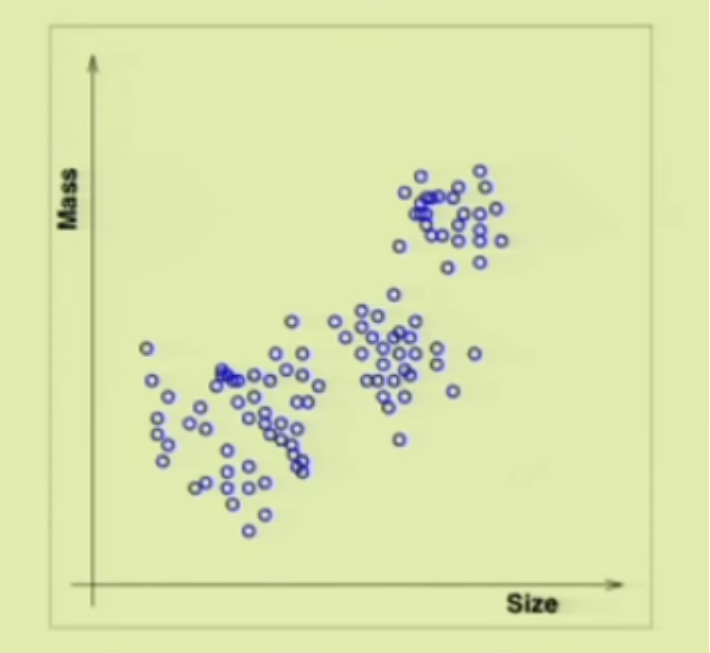
- Training Data is given.

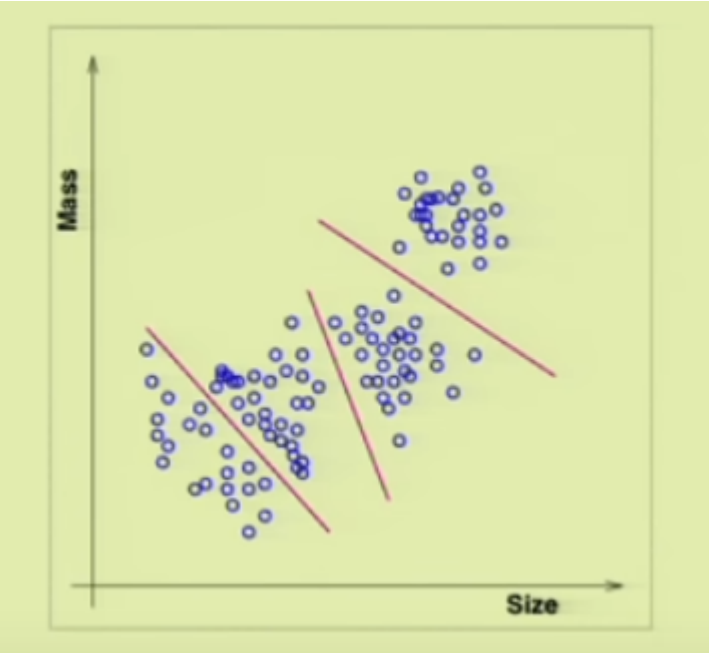
Example from vending machines - coin recognition.



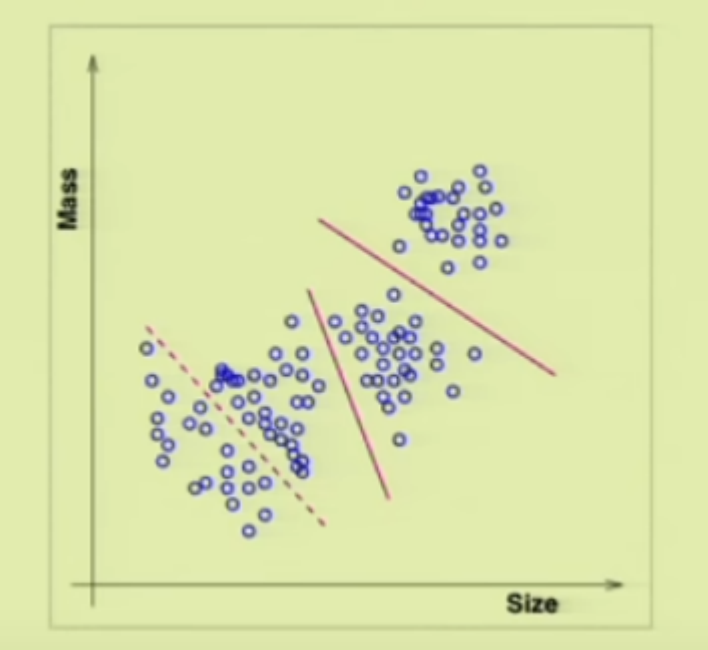
Unsupervised learning

Instead of (input,correct output), we get (input,?)





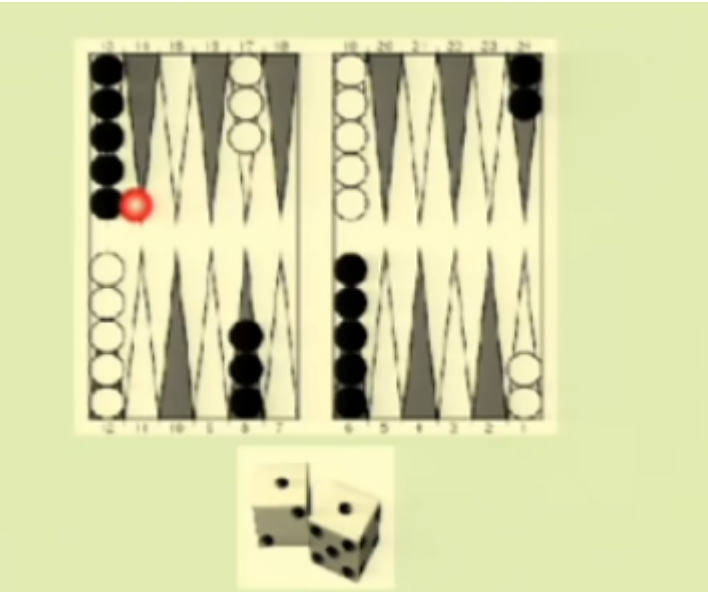
Sometimes there need not be clear cut distinction between clusters.



Unsupervised learning is a way to get a **higher level** representation of the input.

Reinforcement learning

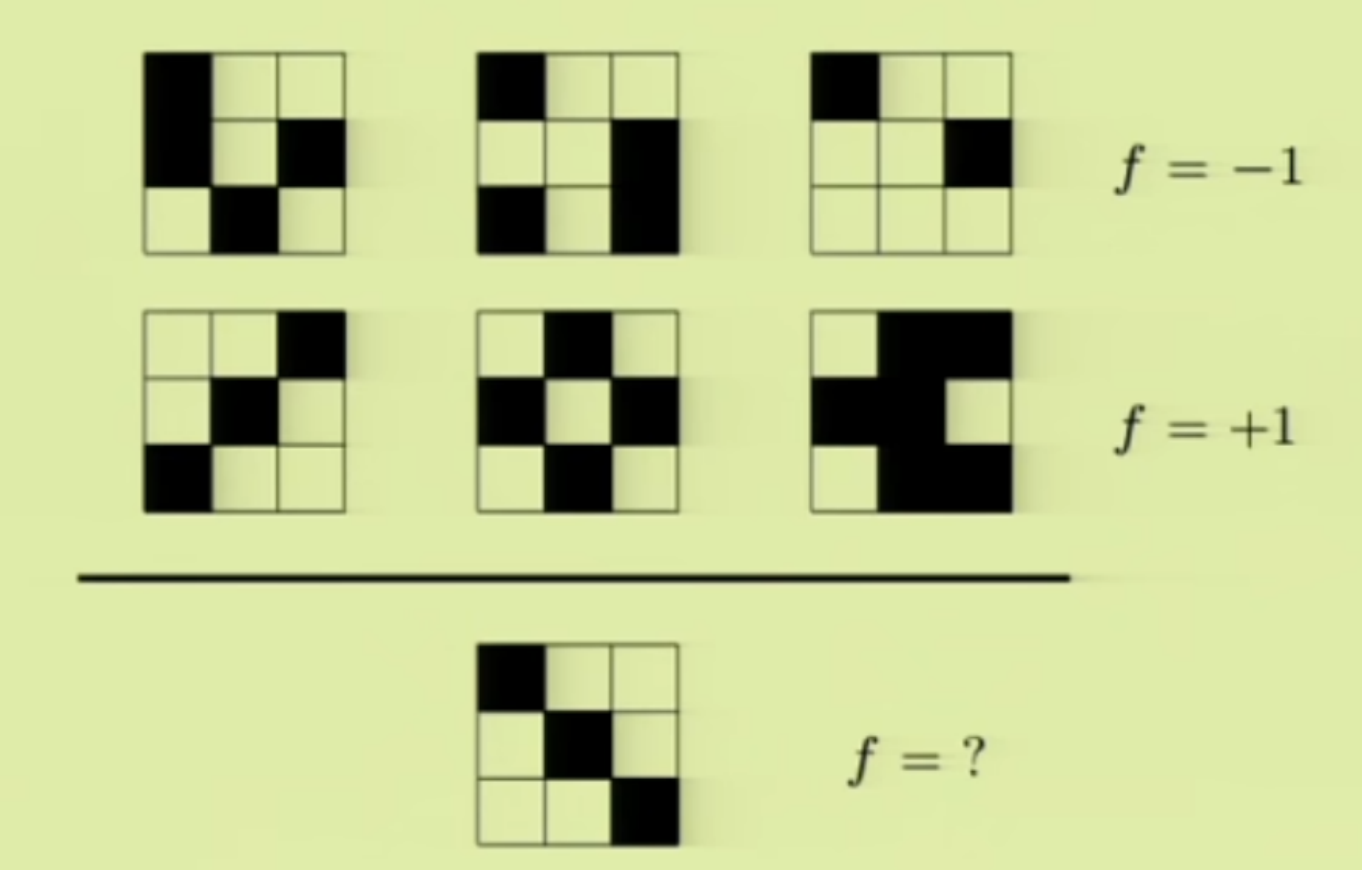
(input,*some* output,grade for this output).



Backgammon - What is the best move given the state?

- Make a random move m.
- Play and see what happens *eventually*
- Assign a credit for move m.

Learning Puzzle



It is impossible to say with absolute certainity if f for the example is 1 or -1 because the target function is unknown.

Q and A:

1. How do you determine if the points are linearly separable?

- The perceptron algorithm does not converge!
  The pocket algorithm (modification of perceptron algorithm for the non linearly separable case)

2. How does the rate of convergence of PLA changes with the dimensionality of data?

- Badly :)

3. How do you know if there is a pattern or not?

- We don't :) Is learning Feasible?

4. Statistics vs ML

- ML - try to make fewer assumptions
  Statistics - make more precise assumptions