# AI: ML

* Given: Training examples <x, f(x)> for some unknown function f
* Find: Either f(precisely) or a good approx. of f

Examples:

1. Detecting objects in image
2. Face recognition
3. Categorization into different classes (eg. A taxonomy on flowers)
4. Credit risk assessment
5. Medical diagnosis

When is it a good idea?

1. Lots of data
   * Takes a lot of time for human
   * Many different examples/ corner cases
   * That we hope they will help us
2. Situations where there are no human experts
3. Situations where human can perform the task but cannot really explain how they do it
4. Character recognition
   * Image
5. Situations where each user needs
   * A customized spam/ham email
   * Amazon recommendations

“No Free-lunch” Theories

WE can’t expect to generalize well in any problems given to us

We need to be biased for the solutions somehow

THM: Consider function f: {0,1}n->{0,1} (f belongs to F)

For any feature L,

F

Where Accu(L) is generalization accuracy of L, accuracy of L on training examples.

F is all Boolean functions, |F| = 2^(2^n)

…. Missing parts

Corollary:

For any two learners L1, L2: if there exists problem P ehrtr where the generalization accuracy AccG(L1) > AccG(L2) then there exists another problem P’ s.t. AccG(L1) > AccG(L2)

Version Spaces:

Conjunction of n literals (var or its negation)

|# of conjunctions| = 3^n + 1(constant )

(true false or not used)

max len of a conjunction is 2n

Representation: Set of truth assignments (inefficient)

Exploit some compact representations:

Eg. Conjunction as strings of length n