Introduction to Machine Learning

Ronald J. Williams
CSG220, Spring 2007

What is learning?

- Learning => improving with experience
- Learning Agent = Performance Element
  + Learning Element
- Performance element decides what actions to take
  - e.g., identify this image
  - e.g., choose a move in this game
- Learning element modifies performance element so that it makes better decisions

Learning agent design

- Which components of the performance element are to be learned?
- What feedback is available to learn these components?
- What representation is to be used for the components?
- How is performance to be measured (i.e., what is meant by *better* decisions?)

Wide range of possible goals

- Given a set of data, find potentially predictive patterns in it
  - data mining
  - scientific discovery
- As a result of acquiring new data, gain knowledge allowing an agent to exploit its environment
  - robot navigation
  - acquisition of new knowledge may be passive or active (e.g., exploration or queries)

- Given experience in some problem domain, improve performance in it
  - game-playing
  - robotics
- Rote learning qualifies, but more interesting and challenging aspect is to be able to generalize successfully beyond actual experiences

Learning vs. programming

- Learning is essential for unknown environments, i.e., when designer lacks omniscience
- Learning is essential in changing environments
- Learning is useful as a system construction method
  - expose the agent to reality rather than trying to write it down

Application examples

- Robot control
- Playing a game
- Recognizing handwritten digits
- Various bioinformatics applications
- Filtering email (e.g., spam detection)
- Intelligent user interfaces

Relevant Disciplines

- Artificial intelligence
- Probability & statistics
- Control theory
- Computational complexity
- Philosophy
- Psychology
- Neurobiology

3 categories of learning problem

- Supervised learning
- Unsupervised learning
- Reinforcement learning

Not an exhaustive list

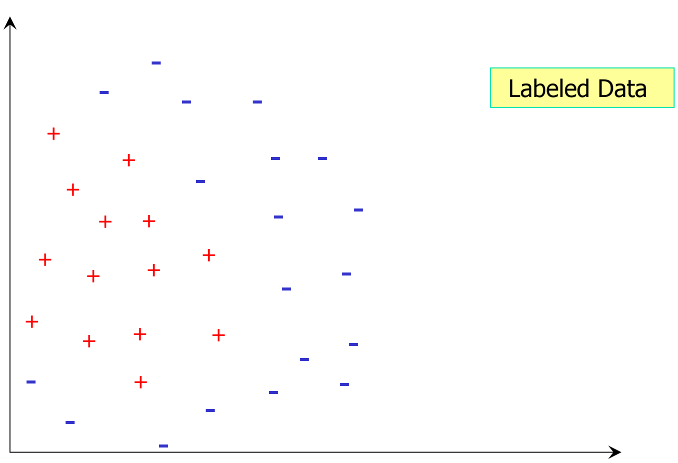
Not necessarily mutually exclusive

Supervised Learning

- Also called inductive inference
- Given training data $\{(x_i,y_i)\}$, where $y_i = f(x_i)$ for some unknown function f : X -> Y, find an approximation h to f
  - called a classification problem if Y is a small discrete set (e.g., {+, -})
  - called a regression problem if Y is a continuous set (e.g., a subset of R)
- More realistic, but harder: each observed y_i is a noise-corrupted approximation to $f(x_i)$

- X called the instance space
- Construct/adjust h to agree with f on training set
- h is *consistent* if it agrees with f on all training examples
  - inappropriate if noise assumed present
- If Y = {+, -}, define {x ε X | f(x)=+}, the set of all positive instances, to be a *concept*
- Thus 2-class classification problems may also be called *concept learning* problems

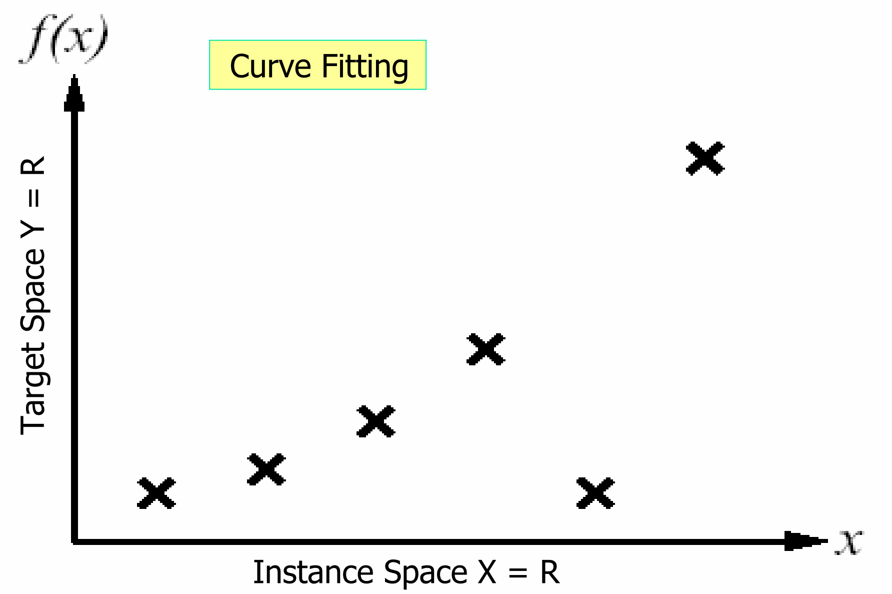
Classification (supervised)



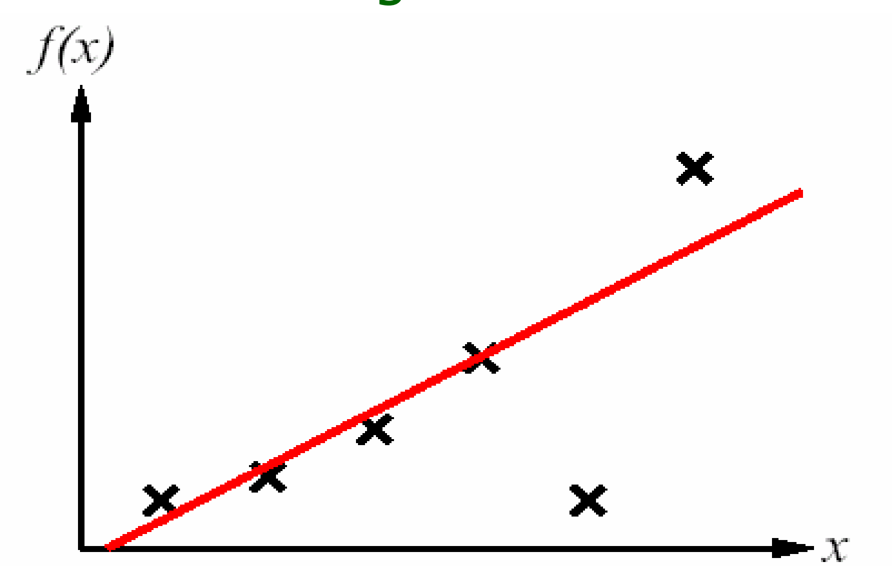
Instance Space X = R^2

Target Space Y = {+, -}

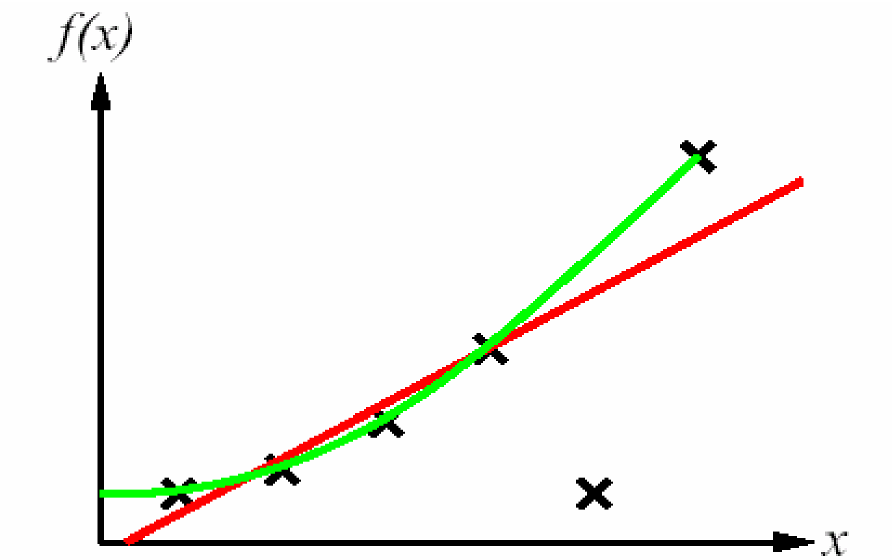
Regression (supervised)



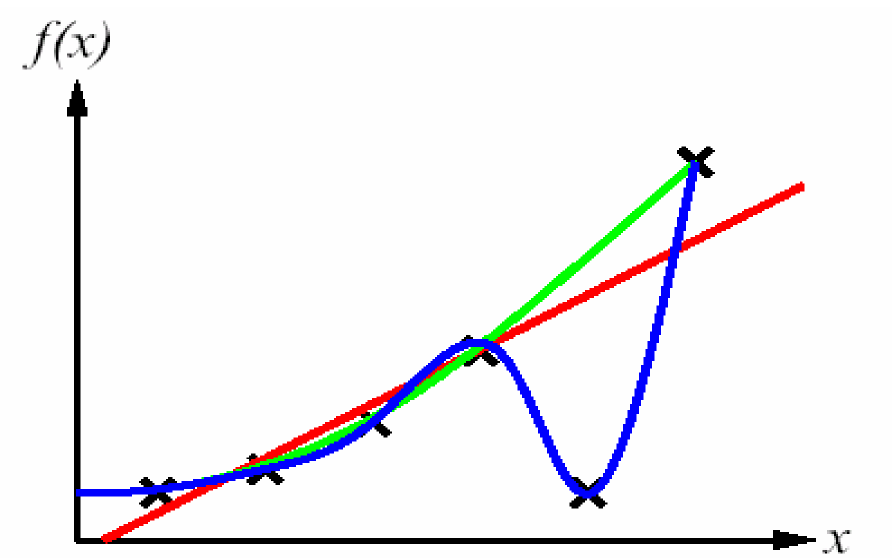
Regression



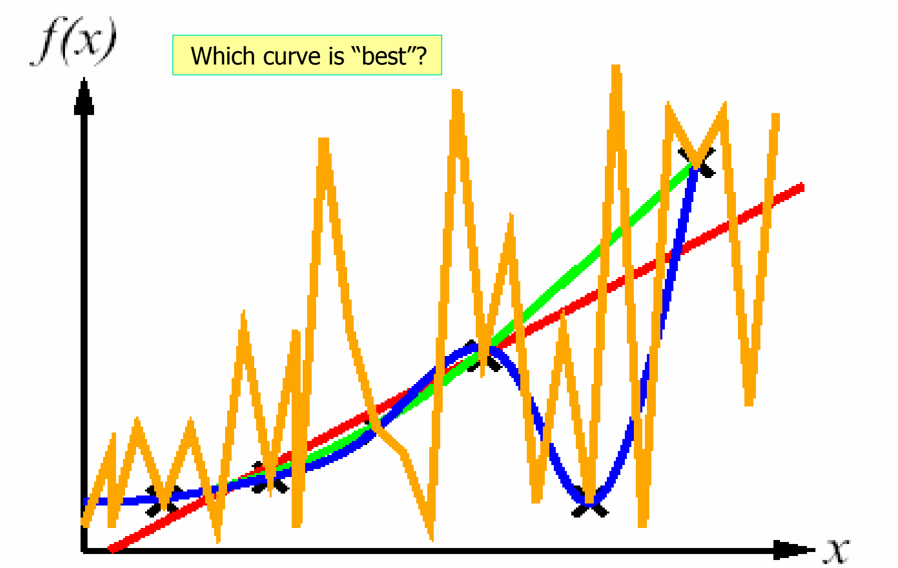
Regression



Regression



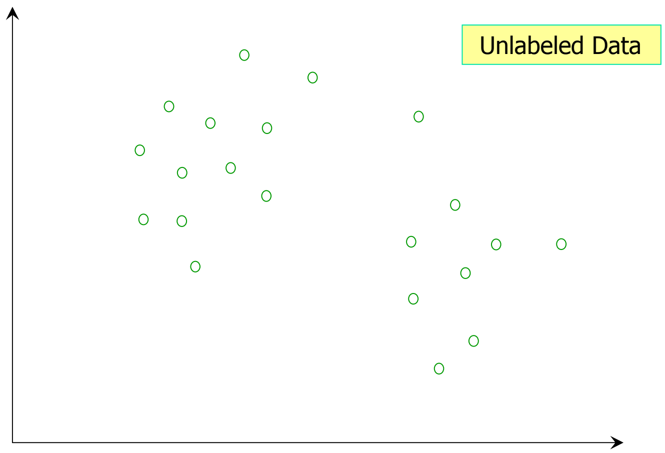
Regression



Unsupervised Learning

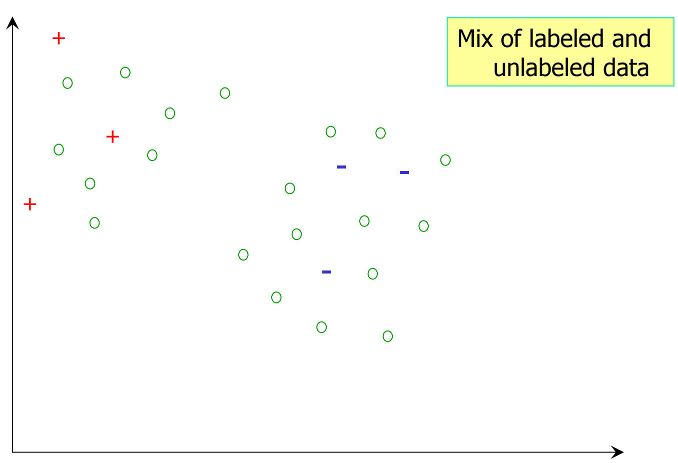
- Have an instance space X
- Possible objectives
  - clustering
  - characterize distribution
  - principal component analysis
- One possible use: novelty detection
  "This newly observed instance is different"
- Also includes such things as association rules in data mining
  "People who buy diapers tend to also buy beer"

Unsupervised



Instance Space X = R^2

But also ...

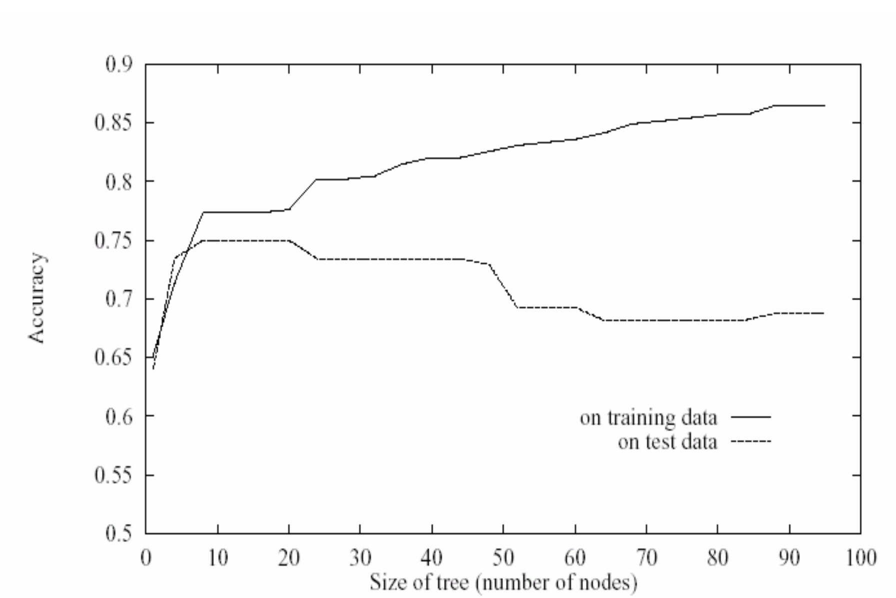


Instance Space X = R^2

Target Space Y = {+, -}

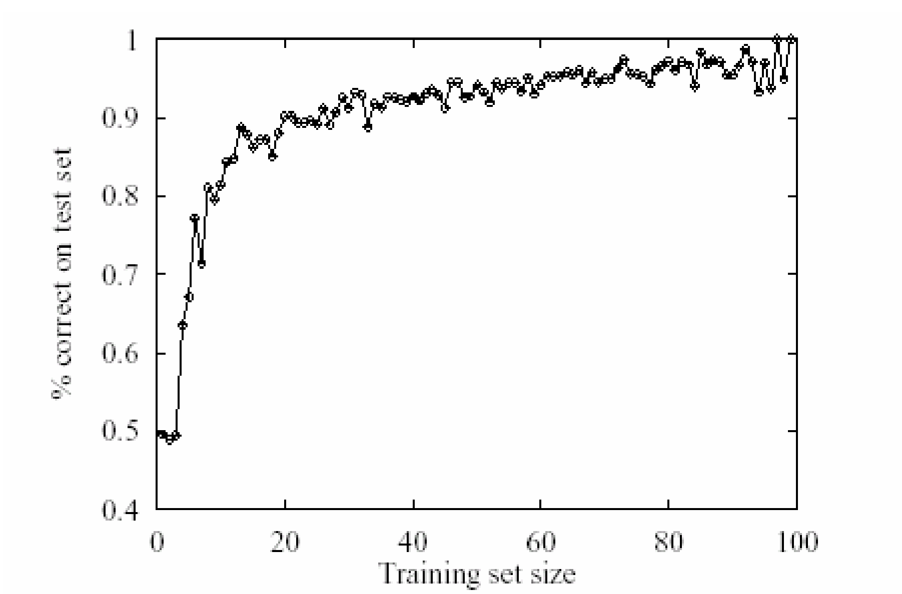
Overfitting

- Especially applicable in supervised learning, but may also appear in other types of learning problems
- Often manifests itself by having the learner perform worse on test data even as it gets better at fitting the training data
- There are practical techniques as well as theoretical approaches for trying to avoid this problem

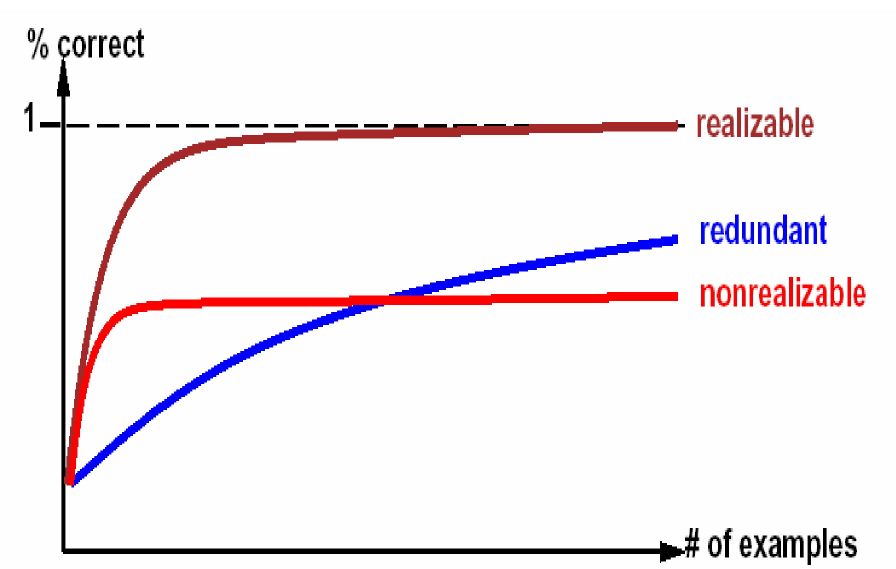


Performance measurement for supervised learning

- How do we know that h ≈ f? (Hume's Problem of Induction)
  - use theorems of computational/statistical learning theory, or
  - try h on a new test set of examples (using same distribution over instance space as training set)
- Learning curve = % correct on test set as a function of training set size



- Learning curve depends on
  - realizable (can express target function) vs. non-realizable
    - non-realizability can be due to missing attributes or restricted hypothesis class that excludes true hypothesis (called *selection bias*)
  - redundant expressiveness (e.g., many irrelevant attributes)
  - size of hypothesis space



- Occam's razor: maximize a combination of consistency and simplicity
  - in this form, just an informal principle
  - involves a trade-off
- Attempts to formalize this
  - penalize "more complex" hypotheses
  - Minimum Description Length
  - Kolmogorov complexity
- Alternative: Bayesian approach
  - start with *a priori* distribution over hypotheses

Reinforcement Learning

- Applies to choosing sequences of actions to obtain a good long-term outcome
  - game-playing
  - controlling dynamical systems
- Key feature is that system not told directly how to behave, only given a performance score

  "That was good/bad"

  "That was worth a 9.5"

Issues for a learning system designer

- How to represent performance element inputs and outputs
  - symbolic
  - logical expressions
  - numerical
  - attribute vectors
- How to represent the input/output mapping
  - artificial neural network
  - decision tree
  - Bayes network
  - general computer program
- What kind of prior knowledge to use and how to represent it and/or take advantage of it during learning

- Contrasting representations of X (and Y and h, if applicable)
  - symbolic, with logical rules (e.g., X = shapes with size and color specified)
    - e.g., instances:
      (Shape=circle)^(Size=large)^(Color=red)
    - e.g., rules:
      IF (shape=circle)^(size=large) THEN (interesting=yes)
      IF (shape=square)^(color=green) THEN (interesting = no)
  - numeric
    - e.g., points in Euclidean space R^n

Learning as search through a hypothesis space H

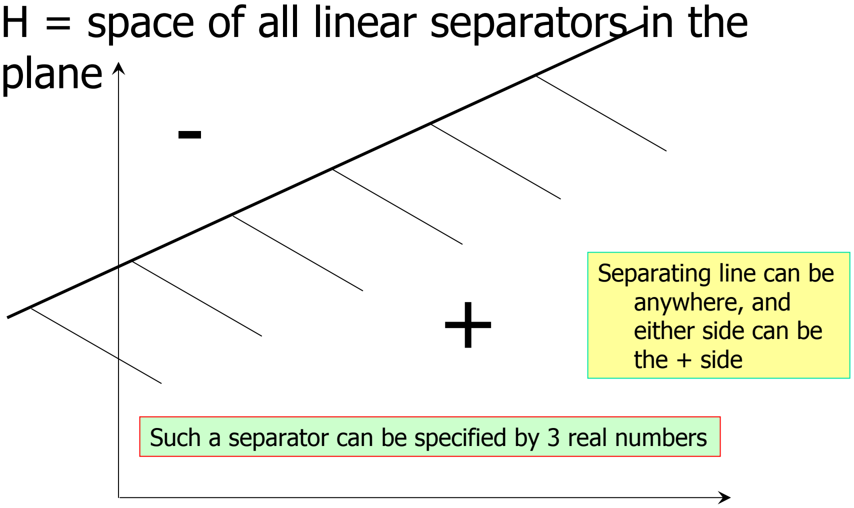
- Inductive bias
  - Selection bias: only hypotheses h ε H are allowed
  - Preference bias: H includes all possible hypotheses, but if more than one fits the data, choose the "best" among these (e.g., Occam's razor: simpler hypotheses are better)
- Selection bias leads to less of an overfitting problem, but runs the risk of eliminating the true hypothesis (i.e., true hypothesis is unrealizable in H)

Selection bias example

- H = pure conjunctive concepts in some attribute/value description language
  - (Shape=square)^(Size=large)
  - (Shape=circle)^(Size=small)^(Color=red)
  - Boolean: A^~B (equivalent to (A=true)^(B=false)
- Description of all positive instances restricted to have this form

Another selection bias example

- H = space of all linear separators in the plane



Hypothesis space size

How many distinct Boolean functions of n Boolean attributes?

Hypothesis space size

How many distinct Boolean functions of n Boolean attributes?

= number of distinct truth tables with 2^n rows = 2^{2^n}

Hypothesis space size

How many distinct Boolean functions of n Boolean attributes?

= number of distinct truth tables with 2^n rows = 2^{2^n}

E.g., with 6 Boolean attributes, there are 18,446,744,073,709,551,616 different possible Boolean hypotheses

Hypothesis space size

How many distinct Boolean functions of n Boolean attributes?

= number of distinct truth tables with 2^n rows = 2^{2^n}

E.g., with 6 Boolean attributes, there are 18,446,744,073,709,551,616 different possible Boolean hypotheses

How many purely conjunctive concepts over n Boolean attributes?

Hypothesis space size

How many distinct Boolean functions of n Boolean attributes?

= number of distinct truth tables with 2^n rows = 2^{2^n}

E.g., with 6 Boolean attributes, there are 18,446,744,073,709,551,616 different possible Boolean hypotheses

How many purely conjunctive concepts over n Boolean attributes?

Each attribute can be required to be true, required to be false, or ignored, so 3^n distinct purely conjunctive hypotheses

Hypothesis space size

How many distinct Boolean functions of n Boolean attributes?

= number of distinct truth tables with 2^n rows = 2^{2^n}

E.g., with 6 Boolean attributes, there are 18,446,744,073,709,551,616 different possible Boolean hypotheses

How many purely conjunctive concepts over n Boolean attributes?

Each attribute can be required to be true, required to be false, or ignored, so 3^n distinct purely conjunctive hypotheses

More expressive hypothesis space

- increases chance that target function can be expressed
- increases number of hypotheses consistent w/ training set so may get worse predictions

Hypothesis space size (cont.)

How many distinct linear separators in n-dimensional Euclidean space?

Hypothesis space size (cont.)

How many distinct linear separators in n-dimensional Euclidean space?

Infinitely many

Hypothesis space size (cont.)

How many distinct linear separators in n-dimensional Euclidean space?

Infinitely many

How many distinct quadratic separators in n-dimensional Euclidean space (e.g., with quadratic curves as separators in R^2)?

Hypothesis space size (cont.)

How many distinct linear separators in n-dimensional Euclidean space?

Infinitely many

How many distinct quadratic separators in n-dimensional Euclidean space (e.g., with quadratic curves as separators in R^2)?

Infinitely many

Hypothesis space size (cont.)

How many distinct linear separators in n-dimensional Euclidean space?

Infinitely many

How many distinct quadratic separators in n-dimensional Euclidean space (e.g., with quadratic curves as separators in R^2)?

Infinitely many

It's clear that allowing more complex separators gives rise to a more expressive hypothesis space, but a simple count of hypotheses doesn't measure it

Hypothesis space size (cont.)

How many distinct linear separators in n-dimensional Euclidean space?

Infinitely many

How many distinct quadratic separators in n-dimensional Euclidean space (e.g., with quadratic curves as separators in R^2)?

Infinitely many

Preview of coming attractions

It's clear that allowing more co… …ives rise to a more expressive hypothes… …simple count of hypotheses doesn't measure i…

But there is a measure of hypothesis space size that applies even to these infinite hypothesis spaces: *Vapnik-Chervonenkis (VC) dimension*