Learning

A learning program is said to *learn* from *experience* E on *task* T with respect to *performance measure* P, if its performance on T improves with experience E.

A learning program produces a *representation* R (often called a hypothesis h) of what it has learned. Another program can use R to perform T.

A learning program uses a *learning algorithm* A to produce R from E. Many different algorithms can be used produce the same type of representation.

Example: Learning Checkers

Task T: playing checkers

Performance measure P: percent of games won against a set of players.

Training experience E: games played against the other players.

Representation R: an evaluation function that measures the goodness of the board.

Learning algorithm A: a variation of temporal difference.

Example: Learning Handwriting

Task T: recognize handwritten letters.

Performance measure P: error rate on sample handwriting.

Training experience E: grayscale images of sample handwriting, all identified in advance.

Representation R: support vector machine, where each pixel is a separate attribute.

Learning algorithm A: SVMlight.

Supervised and Concept Learning

Supervised Learning

X is the set of all possible instances (examples).
Y is the set of all possible outcomes (classes).
(x, y) is a labeled example, $x \in X$ and $y \in Y$.
D is the dataset, a set of labeled examples.
H is the hypothesis space.
A hypothesis $h \in H$ is a function from X to Y.
The target hyp. c is a function from X to Y.
c may or may not be in H.

Concept Learning. $Y = \{-1, 1\}$ or $\{0, 1\}$.
Examples are positive or negative.

Attribute Vectors. Each $x \in X$ is a vector of values, e.g., (sunny, hot, high, false).

Inductive Learning Assumption
Study the past if you would divine the future. - Confucius
"Small" hypotheses that do well on the training examples will do well on unobserved examples.

Hypothesis Space Assumption: usually made
Assume $c \in H$ or that c can be approximated by $h \in H$.

No Noise Assumption: usually not made
Assume $c(x) = y$ for all $(x, y) \in D$.

No.	Attributes				Class
	Outlook	Temp	Humidity	Windy	
1	sunny	hot	high	false	neg
2	sunny	hot	high	true	neg
3	overcast	hot	high	false	pos
4	rain	mild	high	false	pos
5	rain	cool	normal	false	pos
6	rain	cool	normal	true	neg
7	overcast	cool	normal	true	pos
8	sunny	mild	high	false	neg
9	sunny	cool	normal	false	pos
10	rain	mild	normal	false	pos
11	sunny	mild	normal	true	pos
12	overcast	mild	high	true	pos
13	overcast	hot	normal	false	pos
14	rain	mild	high	true	neg

Example: Concept Learning by class

Task T: distinguish "positive" days from "negative" days.

Performance measure P: error rate on test examples.

Training experience E: training examples.

Representation H: if (conjunction of values) then class else other class.

Learning algorithm A: find most accurate h via informal search.

Inductive Bias

Suppose h_1 and h_2 are equally accurate on D.

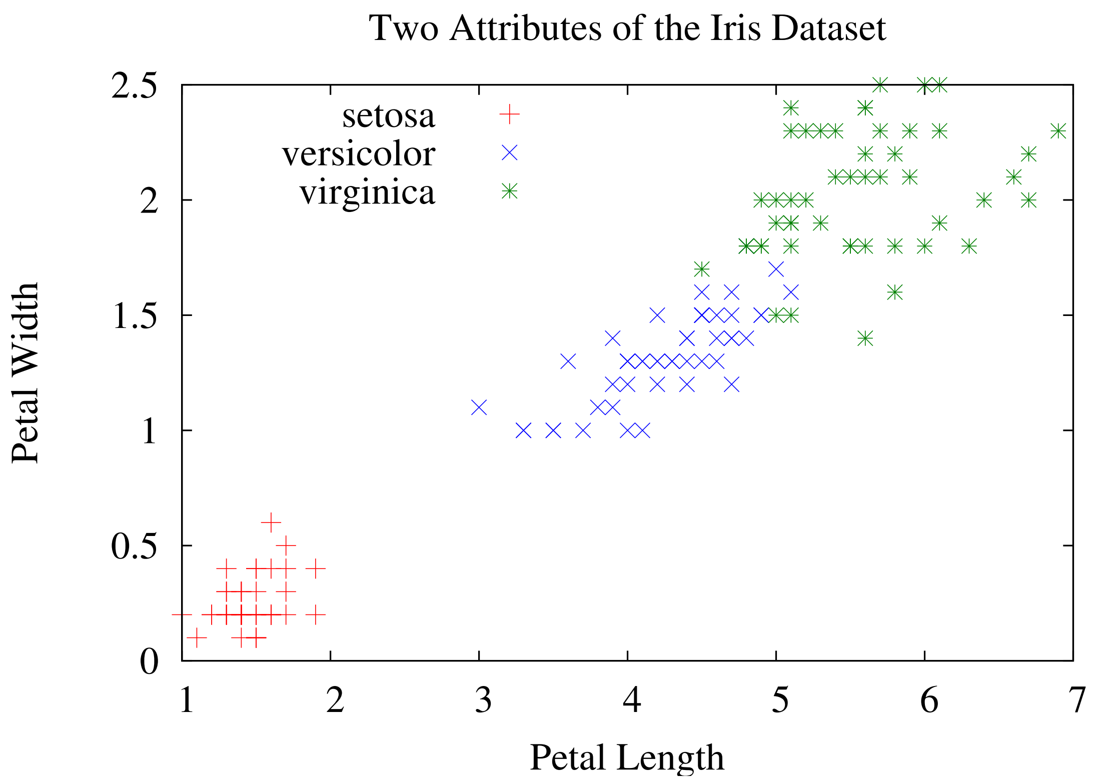
The inductive bias of a learning alg. is preferring one hypothesis over equally good or better ones.

Types of inductive bias include: restrictive hypothesis space, and preference for smaller hypotheses.

Inductive bias is necessary for generalizing to unobserved instances.

Consider a learning alg. with no inductive bias. With $|X|$ instances, there are $2^{|X|}$ hypotheses. How many hypotheses are consistent with D? How many classify an unobserved $x \in X$ positive? negative?

Iris Dataset



Glass Dataset