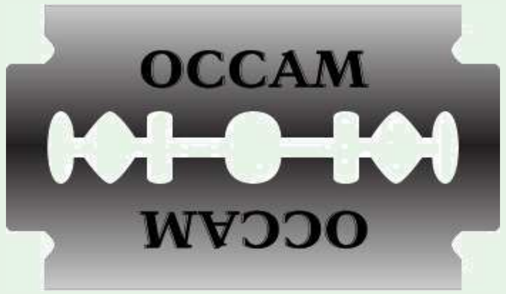
Review of Lecture 17

- Occam's Razor

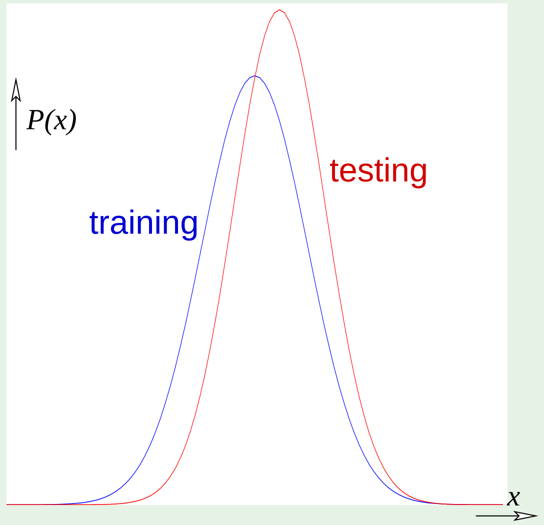
The simplest model that fits the data is also the most plausible.



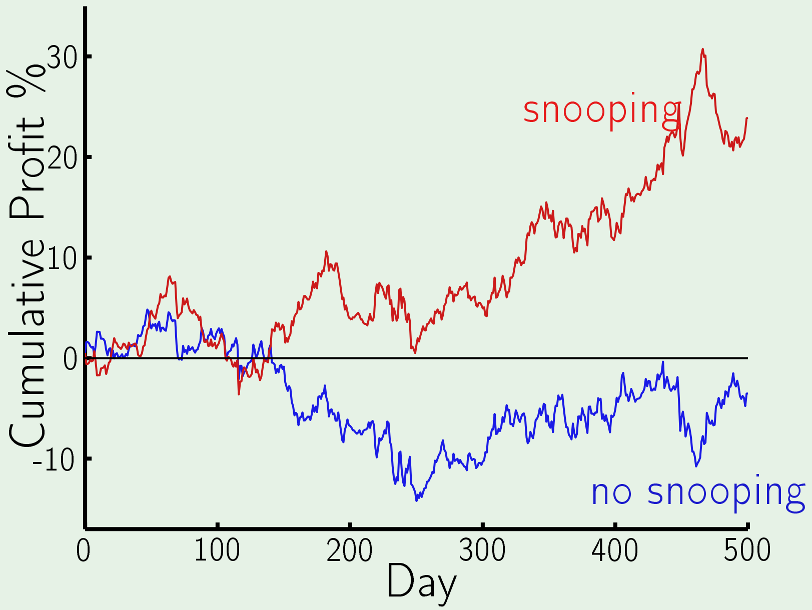
complexity of h $\longleftrightarrow$ complexity of $\mathcal{H}$

unlikely event $\longleftrightarrow$ significant if it happens

- Sampling bias



- Data snooping



Learning From Data

Yaser S. Abu-Mostafa
California Institute of Technology

Lecture 18: Epilogue

Sponsored by Caltech's Provost Office, E&AS Division, and IST • Thursday, May 31, 2012

Outline

- The map of machine learning
- Bayesian learning
- Aggregation methods
- Acknowledgments

It's a jungle out there

semi−supervised learning
overfitting
stochastic gradient descent
SVM
Q learning
Gaussian processes
deterministic noise
data snooping
distribution−free
linear regression
VC dimension
learning curves
collaborative filtering
mixture of expe
sampling bias
neural networks
decision trees
nonlinear transformation
no free
RBF
training versus testing
noisy targets
Bayesian prior
active learning
linear models
bias−variance tradeoff
weak learners
ordinal regression
cross validation
logistic regression
data contamination
ensemble learning
types of learning
perceptrons
hidden Markov mo
error measures
kernel methods
xploration versus exploitation
graphical models
is learning feasible?
soft−order constraint
clustering
weight decay
Boltzmann mach
regularization
Occam's razor

The map

THEORY

- VC
- bias−variance
- complexity
- bayesian

TECHNIQUES

models

- linear
- neural networks
- SVM
- nearest neighbors
- RBF
- gaussian processes
- SVD
- graphical models

methods

- regularization
- validation
- aggregation
- input processing

PARADIGMS

- supervised
- unsupervised
- reinforcement
- active
- online

Outline

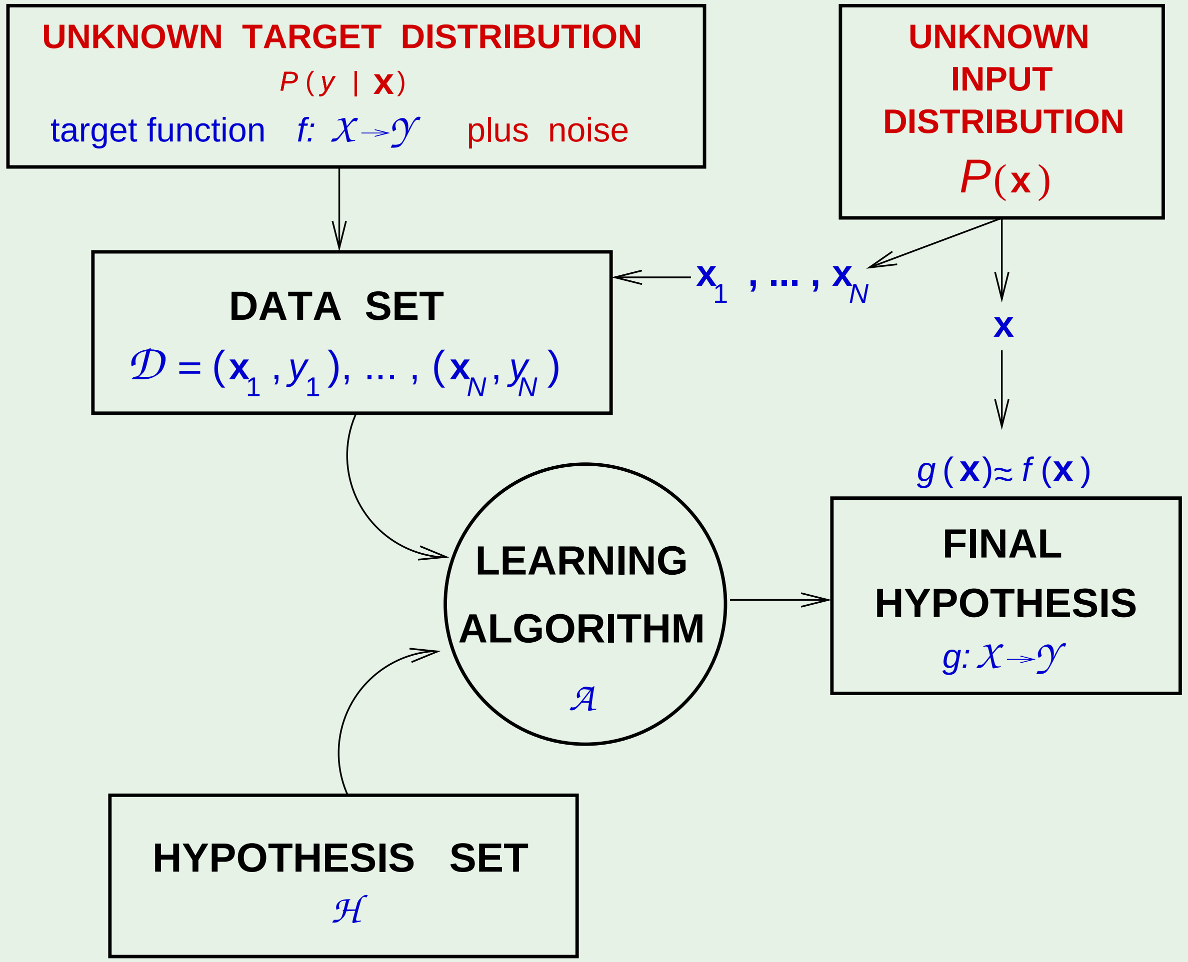
- The map of machine learning
- Bayesian learning
- Aggregation methods
- Acknowledgments

Probabilistic approach

Extend probabilistic role to all components

$P(\mathcal{D} \mid h = f)$ decides which h (likelihood)

How about $P(h = f \mid \mathcal{D})$?



The prior

$P(h = f \mid \mathcal{D})$ requires an additional probability distribution:

$$P(h = f \mid \mathcal{D}) \;=\; \frac{P(\mathcal{D} \mid h = f)\; P(h = f)}{P(\mathcal{D})} \;\propto\; P(\mathcal{D} \mid h = f)\; P(h = f)$$

$P(h = f)$ is the **prior**

$P(h = f \mid \mathcal{D})$ is the **posterior**

Given the prior, we have the full distribution

Example of a prior

Consider a perceptron: h is determined by $\mathbf{w} = w_0, w_1, \cdots, w_d$

A possible prior on $\mathbf{w}$: Each w_i is independent, uniform over $[-1, 1]$

This determines the prior over h - $P(h = f)$

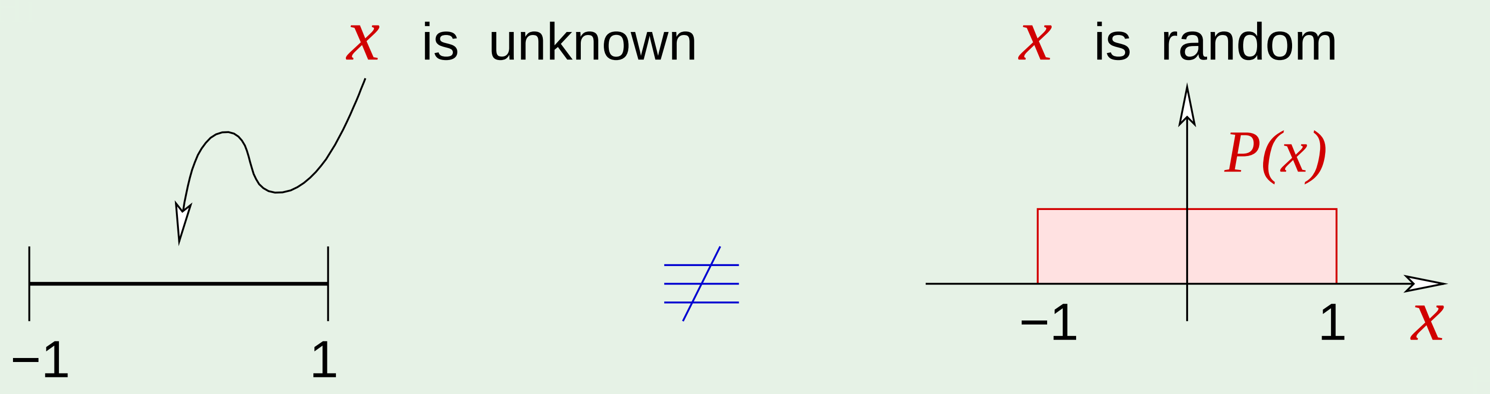
Given $\mathcal{D}$, we can compute $P(\mathcal{D} \mid h = f)$

Putting them together, we get $P(h = f \mid \mathcal{D})$

$$\propto \quad P(h = f)P(\mathcal{D} \mid h = f)$$

A prior is an assumption

Even the most "neutral" prior:



The true equivalent would be:

If we knew the prior

... we could compute $P(h = f \mid \mathcal{D})$ for every $h \in \mathcal{H}$

$\Longrightarrow$ we can find the most probable h given the data

we can derive $\mathbb{E}(h(\mathbf{x}))$ for every $\mathbf{x}$

we can derive the **error bar** for every $\mathbf{x}$

we can derive everything in a principled way

When is Bayesian learning justified?

1. The prior is **valid**

   trumps all other methods

2. The prior is **irrelevant**

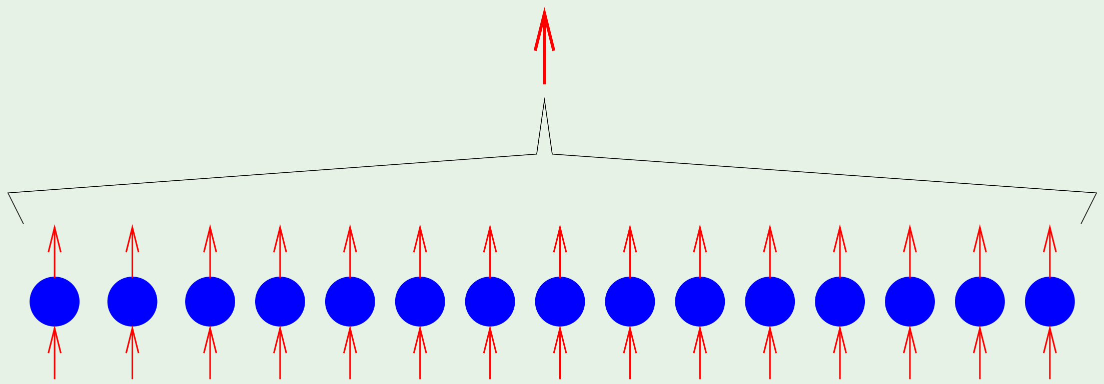
   just a computational catalyst

Outline

- The map of machine learning
- Bayesian learning
- Aggregation methods
- Acknowledgments

What is aggregation?

Combining different solutions $h_1, h_2, \cdots, h_T$ that were trained on $\mathcal{D}$:



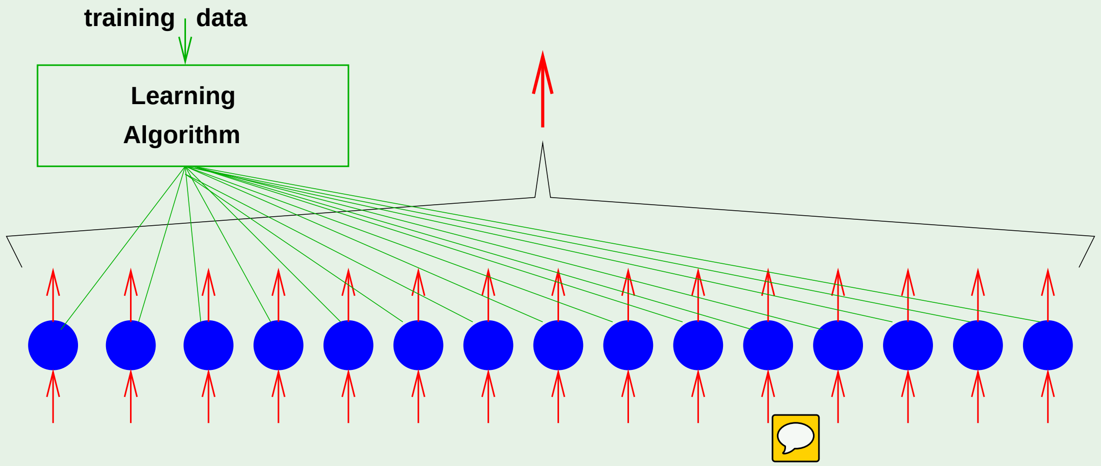
Regression: take an average

Classification: take a vote

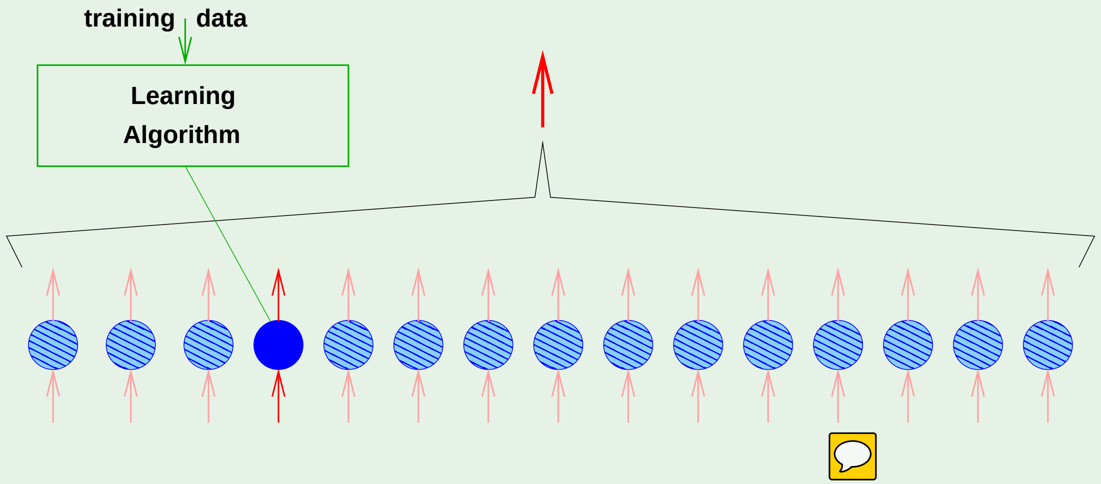
a.k.a. *ensemble learning* and *boosting*

Different from 2-layer learning

In a 2-layer model, all units learn **jointly**:



In aggregation, they learn **independently** then get combined:



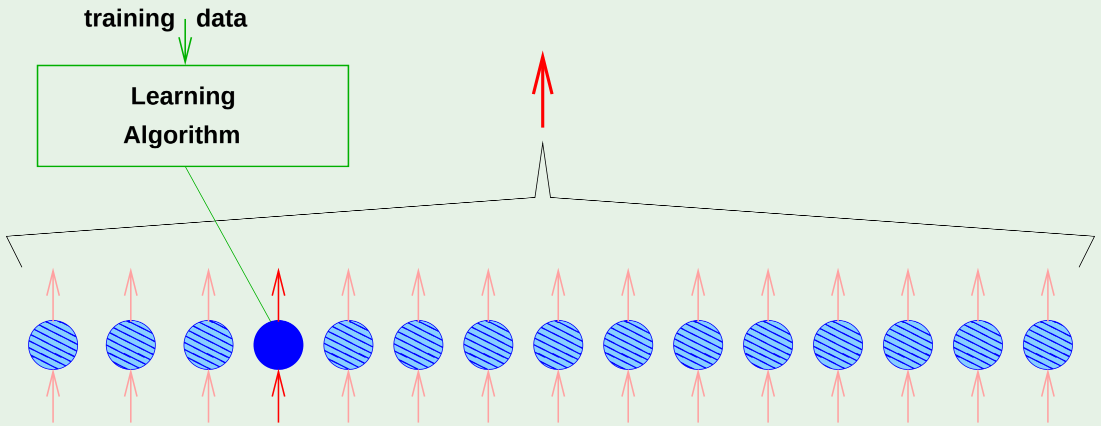
Two types of aggregation

1. **After the fact:** combines existing solutions

Example. Netflix teams merging "blending"

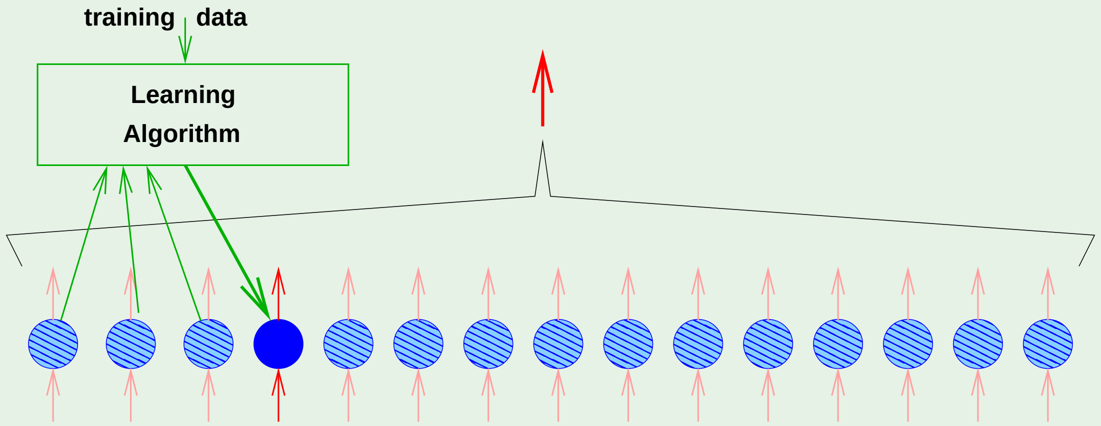
2. **Before the fact:** creates solutions to be combined

Example. Bagging - resampling $\mathcal{D}$



Decorrelation - boosting

Create $h_1, \cdots, h_t, \cdots$ sequentially: Make h_t decorrelated with previous h's:



Emphasize points in $\mathcal{D}$ that were misclassified

Choose weight of h_t based on $E_{\text{in}}(h_t)$

Blending - after the fact

For regression, $h_1, h_2, \cdots, h_T \quad \longrightarrow \quad g(\mathbf{x}) = \sum_{t=1}^{T} \alpha_t \, h_t(\mathbf{x})$

Principled choice of α_t's: minimize the error on an "aggregation data set" pseudo-inverse

Some α_t's can come out negative

Most valuable h_t in the blend?

Outline

- The map of machine learning
- Bayesian learning
- Aggregation methods
- Acknowledgments

Course content

Professor **Malik Magdon-Ismail**, *RPI*

Professor **Hsuan-Tien Lin**, *NTU*

Course staff

Carlos Gonzalez (*Head TA*)

Ron Appel

Costis Sideris

Doris Xin

Filming, production, and infrastructure

Leslie Maxfield and the **AMT** staff

Rich Fagen and the **IMSS** staff

Caltech support

IST - Mathieu Desbrun

E&AS Division - Ares Rosakis and Mani Chandy

Provost's Office - Ed Stolper and Melany Hunt

Many others

Caltech TA's and staff members

Caltech alumni and Alumni Association

Colleagues all over the world

To the fond memory of

Faiza A. Ibrahim