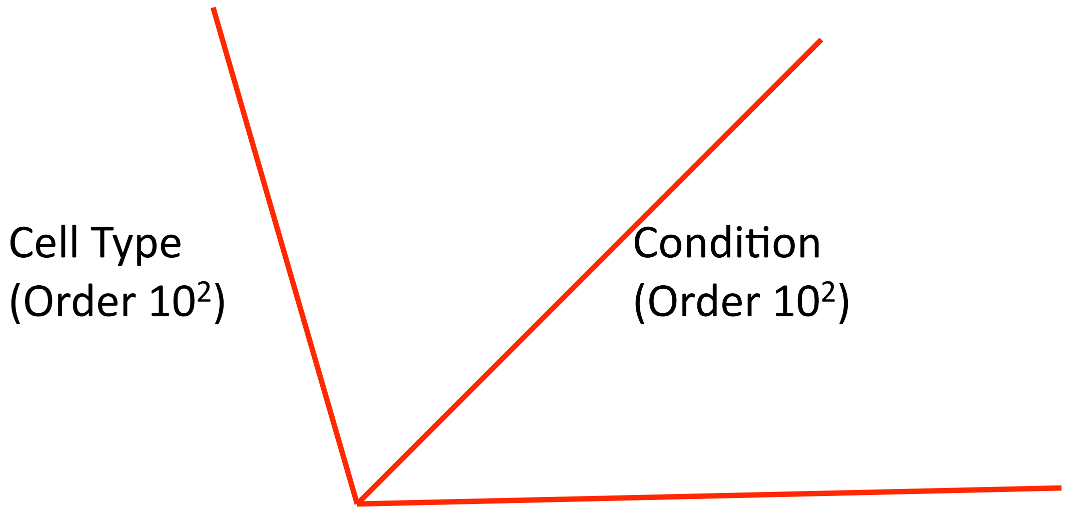
Machine Learning Approaches to Biological Research: Bioimage Informatics and Beyond

Lecture 4: Active learning

Robert F. Murphy
External Senior Fellow, Freiburg Institute for Advanced Studies
Ray and Stephanie Lane Professor of Computational Biology, Carnegie Mellon University

September 29-October 1, 2009

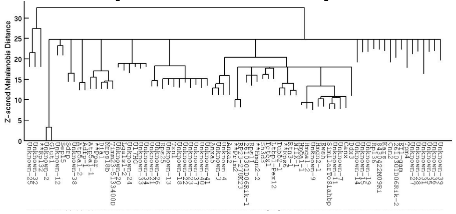
The problem of subcellular location analysis



Cell Type (Order 10^2)

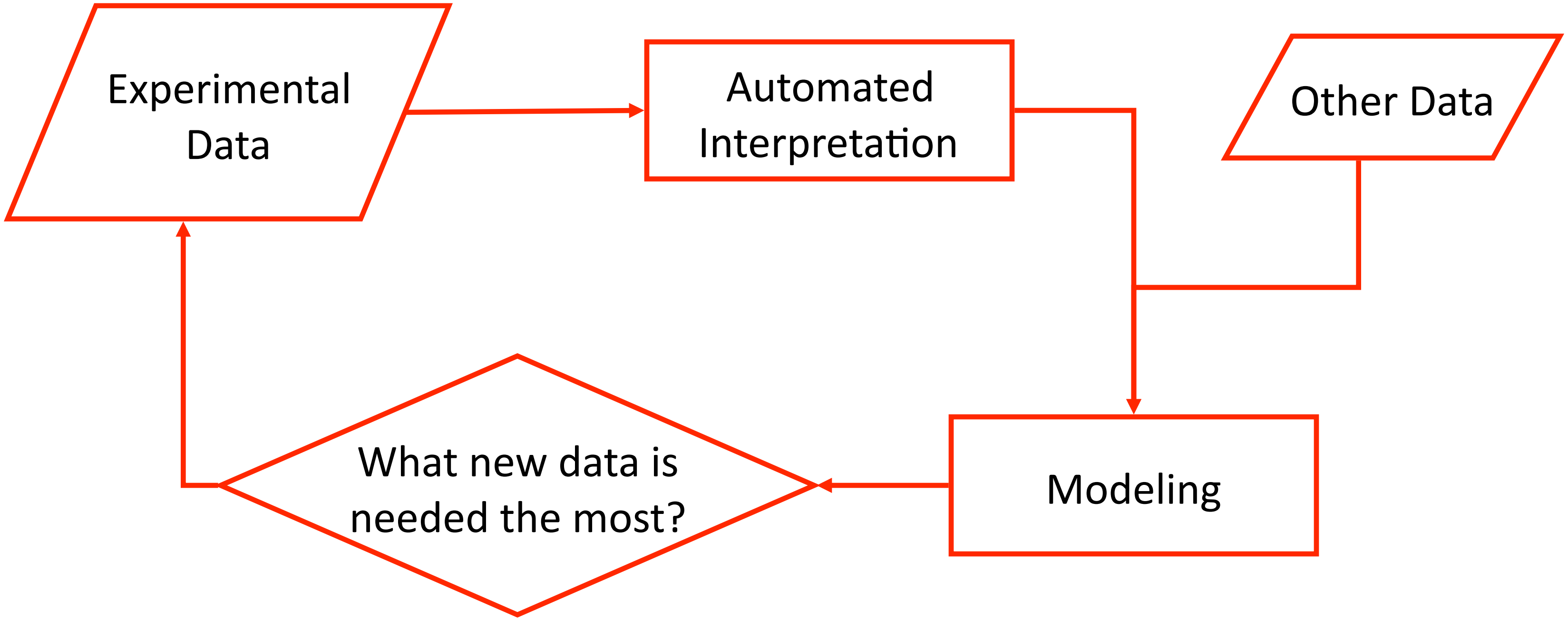
Condition (Order 10^2)

Protein (Order 10^4)

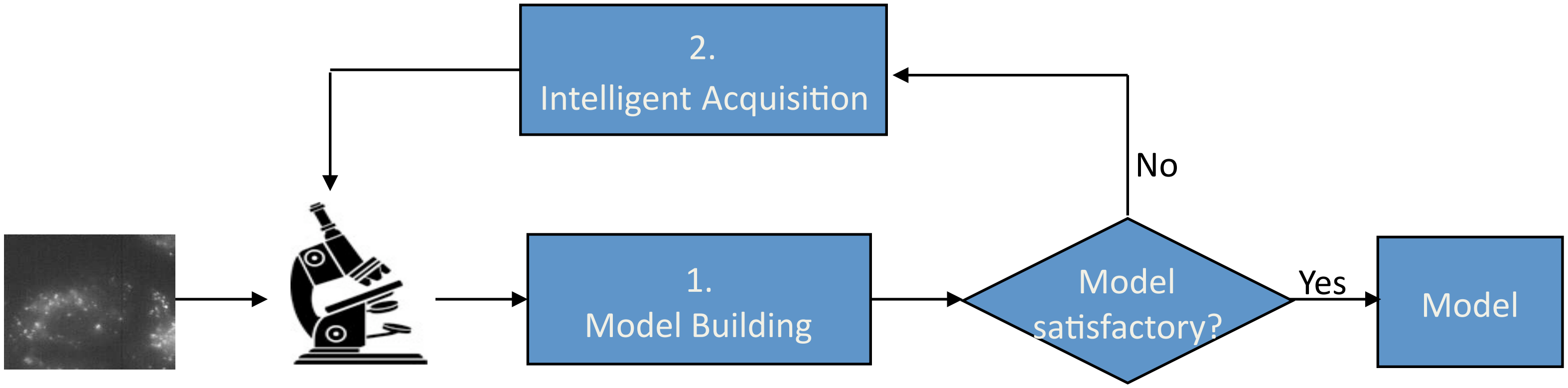


Plus: Time scale from subsecond to years

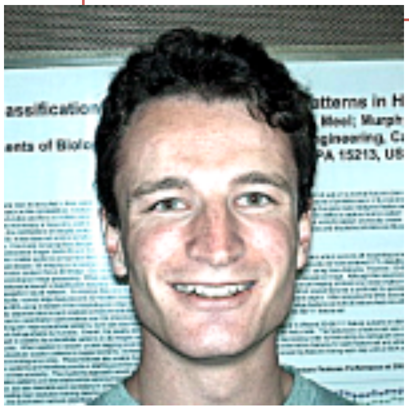
Automated Science (Active Learning)



Efficient Acquisition and Learning of Fluorescence Microscope Data Models



Develop a mathematical framework and algorithms
to build accurate models of fluorescence microscope data sets
as well as design intelligent acquisition systems based on those models



1. Use all the input from the microscope to model the data set

2. Choose acquisition requests that allow us to construct an accurate model in the shortest amount of time

Charles Jackson & Jelena Kovacevic

Active Learning

Slides from Irina Rish and Barbara Engelhardt

Problem Setup

- Unlabeled data available but labels are expensive

- I would like to choose which data to label
  - to maximize the “value” of that data to my problem
  - to minimize the “cost” of labeling

Toy Example: threshold function

Unlabeled data: labels are all 0 then all 1 (left to right)

Classifier is threshold function:

$$h_w(x) = 1 \text{ if } x > w \text{ (}0\text{ otherwise)}$$

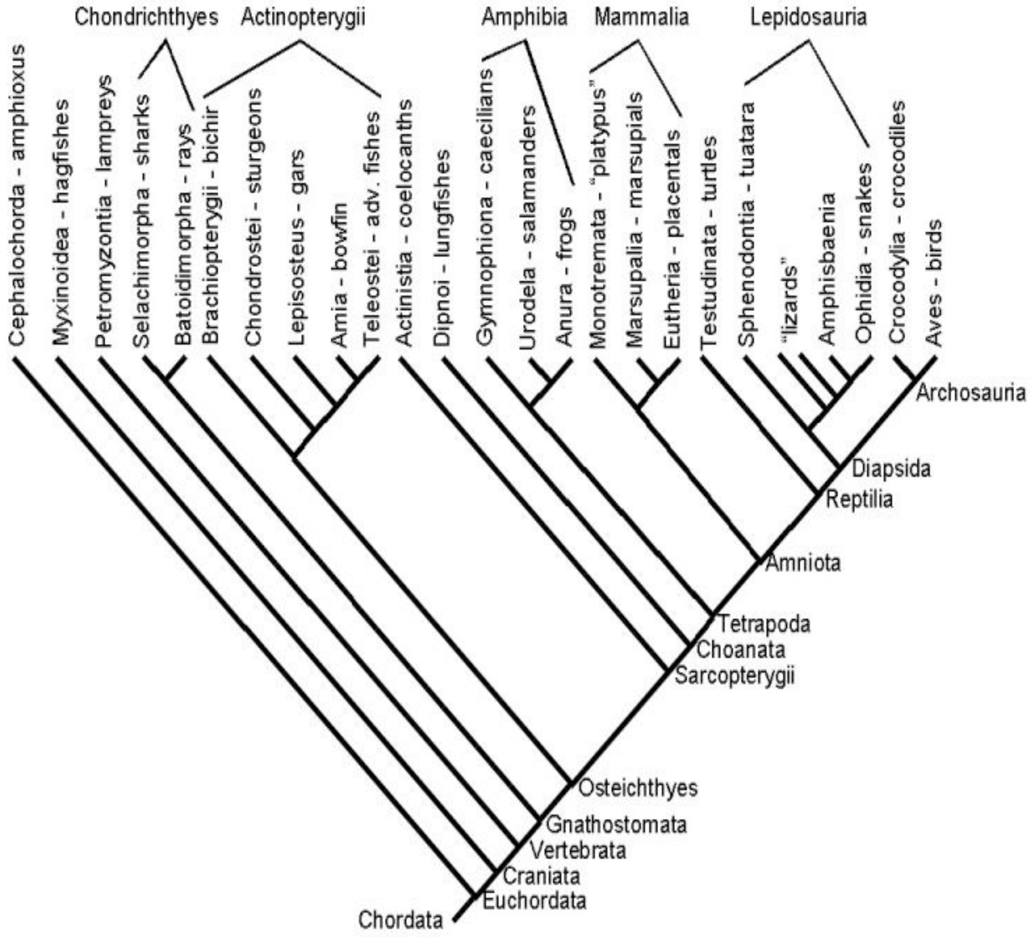
Goal: find transition between 0 and 1 labels in minimum steps

Naïve method: choose points to label at random on line

Better method: binary search for transition between *0* and *1*

Example: Sequencing genomes

- What genome should be sequenced next?
- Criteria for selection?
- Optimal species to detect functional elements across genomes
- Breadth of species encompassing biological phenomena of interest
- (Not the same as the most diverged set of species)
- Marsupials should be sequenced next



[McAuliffe et al., 2004]

Example: collaborative filtering

- Users rate only a few movies usually; ratings "expensive"
- Which movies do you show users to best extrapolate movie preferences?
- Also known as *questionnaire design*
- Baseline questionnaires:
  - Random: *m* movies randomly
  - Most Popular Movies: *m* most frequently rated movies
- Most popular movies is **not** better than random design!
- Popular movies rated highly by all users; do not discriminate tastes

If You Like I, Robot [WS]...

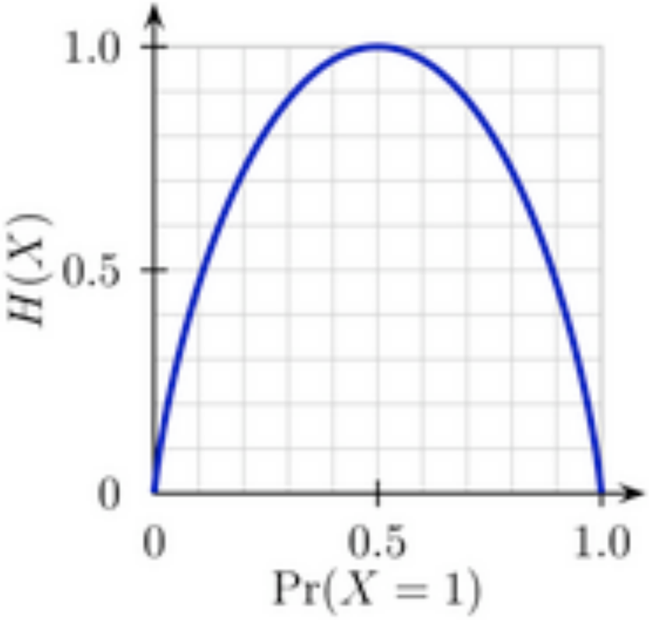
[Yu et al. 2006]

Entropy Function

- A measure of information in random event *X* with possible outcomes *{x_1,…,x_n}*

  $H(x) = - \Sigma_i \ p(x_i) \log_2 p(x_i)$

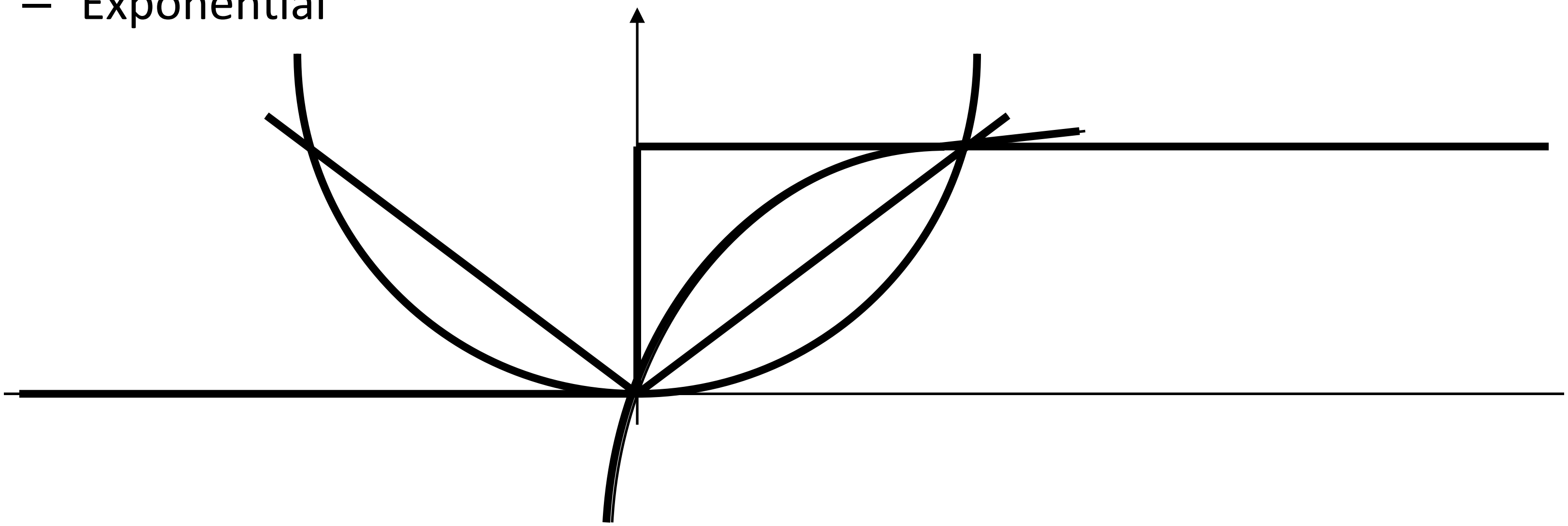
- Comments on entropy function:
  - Entropy of an event is zero when the outcome is known
  - Entropy is maximal when all outcomes are equally likely
- The average minimum yes/no questions to answer some question (connection to binary search)



[Shannon, 1948]

Loss Functions

- A function *L* that maps an event to a real number, representing cost or regret associated with event
- E.g., in regression problems, $L(y, \theta^T f(x))$ maps to reals
- Examples:
  - Quadratic (least squares) loss
  - Linear (absolute value) loss
  - 0-1 (binary) loss
  - Exponential



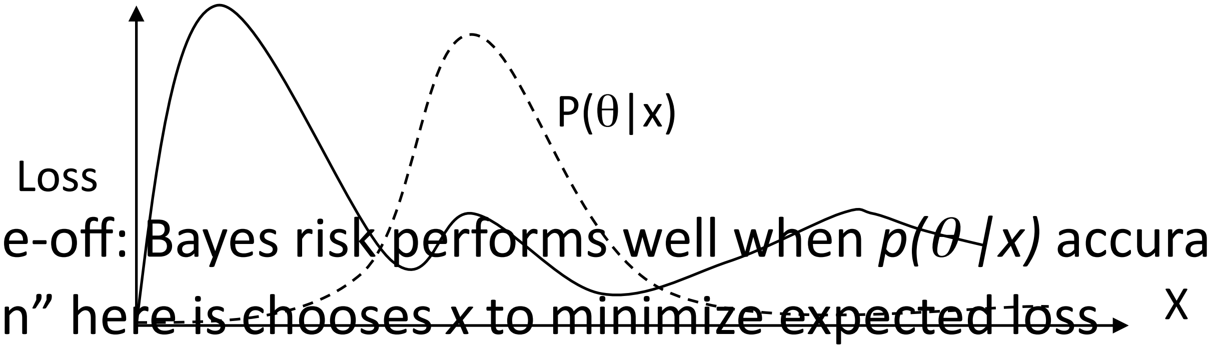
Risk Function

- *Risk* is also known as *expected loss*
- The (frequentist) risk function is explicitly expected loss

$$R(\Theta, X) = \Sigma_x L(\theta, x)\ p(x|\theta)$$

- Bayes risk is defined as posterior expected loss:

$$R(\Theta, X) = \Sigma_\theta L(\theta, x)\ p(\theta|x)$$



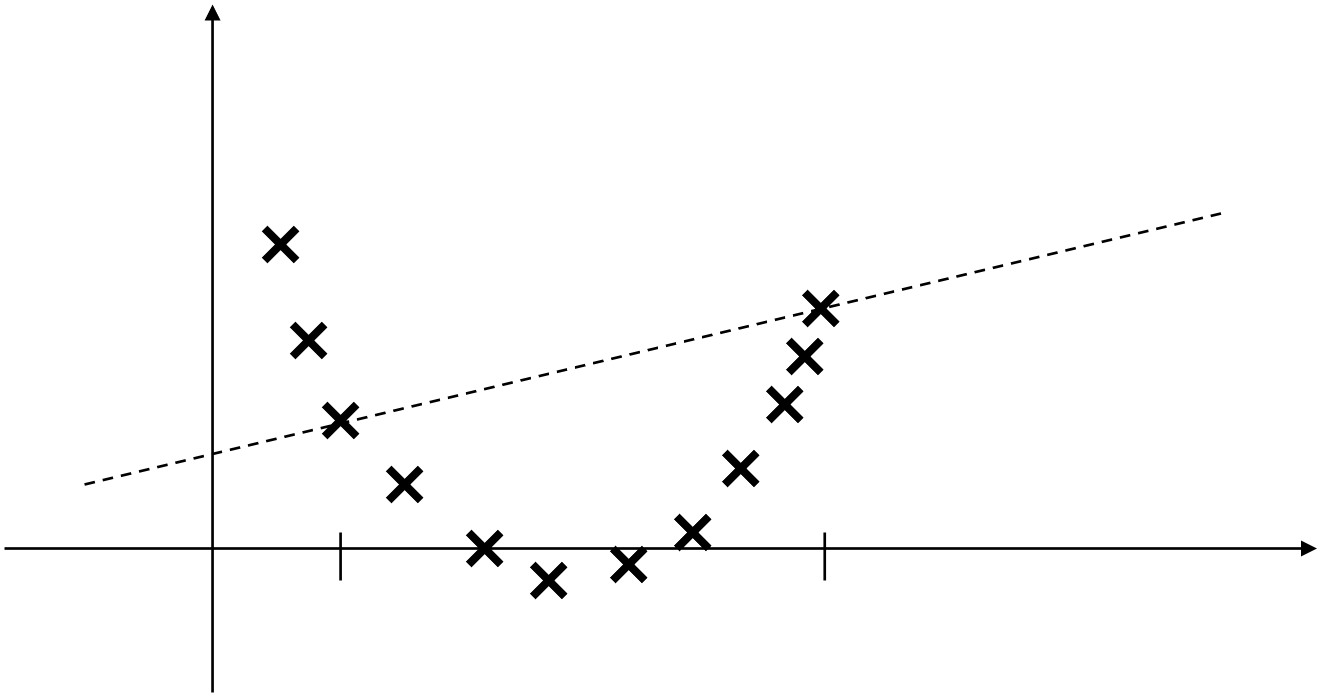
- Trade-off: Bayes risk performs well when *p(θ |x)* accurate
- “Gain” here is chooses *x* to minimize expected loss

What is Active Learning?

- Unlabeled data are readily available; labels are expensive
- Want to use adaptive decisions to choose which labels to acquire for a given dataset
- Goal is accurate classifier with minimal cost

Active learning warning

- Choice of data is only as good as the model itself
- Assume a linear model, then two data points are sufficient
- What happens when data are not linear?



Active Learning

- *Active learner* is able to query world and receive a response before outputting a classifier
- Learner selects queries (but cannot impact response)
- Two general methods:
  - Select "most uncertain" data given model and parameters
  - Select "most informative" data to optimize expected gain
- Given model *M* with parameters θ and loss function *L*
- Query *q* with response *x* updates the model posterior θ'

$$L(\theta', X) = \mathrm{E}_x L(\theta')$$

Active Learning Approaches

- Membership queries
- Uncertainty Sampling
- Query by committee

Membership queries

Earliest model of active learning in theory work [Angluin 1992]

X = space of possible inputs, like $\{0,1\}^n$
H = class of hypotheses

Target concept $h^* \in H$ to be identified *exactly*.
You can ask for the label of any point in X: *no unlabeled data*.

```
$H_0 = H$
For $t = 1,2,\ldots$
        pick a point $x \in X$ and query its label $h^*(x)$
        let $H_t$ = all hypotheses in $H_{t-1}$ consistent with $(x, h^*(x))$
```

What is the minimum number of “membership queries” needed to reduce H to just $\{h^*\}$?

Slide credit: S. Dasgupta

Membership queries: example

$X = \{0,1\}^n$

H = AND-of-positive-literals, like $x_1 \wedge x_3 \wedge x_{10}$

```
$S = \{ \}$  (set of AND positions)
For $i = 1$ to $n$:
    ask for the label of $(1,\ldots,1,0,1,\ldots,1)$ [0 at position $i$]
    if negative: $S = S \cup \{i\}$
```

Total: n queries

General idea: synthesize highly informative points.

Each query cuts the *version space* -- the set of consistent hypotheses -- in half.

Slide credit: S. Dasgupta

Problem

Many results in this framework, even for complicated hypothesis classes.

[Baum and Lang, 1991] tried fitting a neural net to handwritten characters.
Synthetic instances created were incomprehensible to humans!

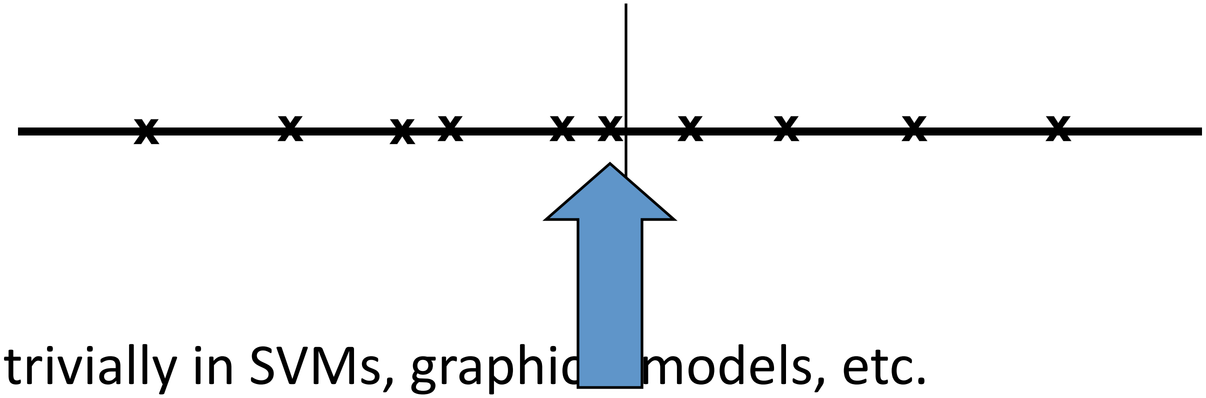
[Lewis and Gale, 1992] tried training text classifiers.
“an artificial text created by a learning algorithm is unlikely to be a legitimate natural language expression, and probably would be uninterpretable by a human teacher.”

Uncertainty Sampling

[Lewis & Gale, 1994]

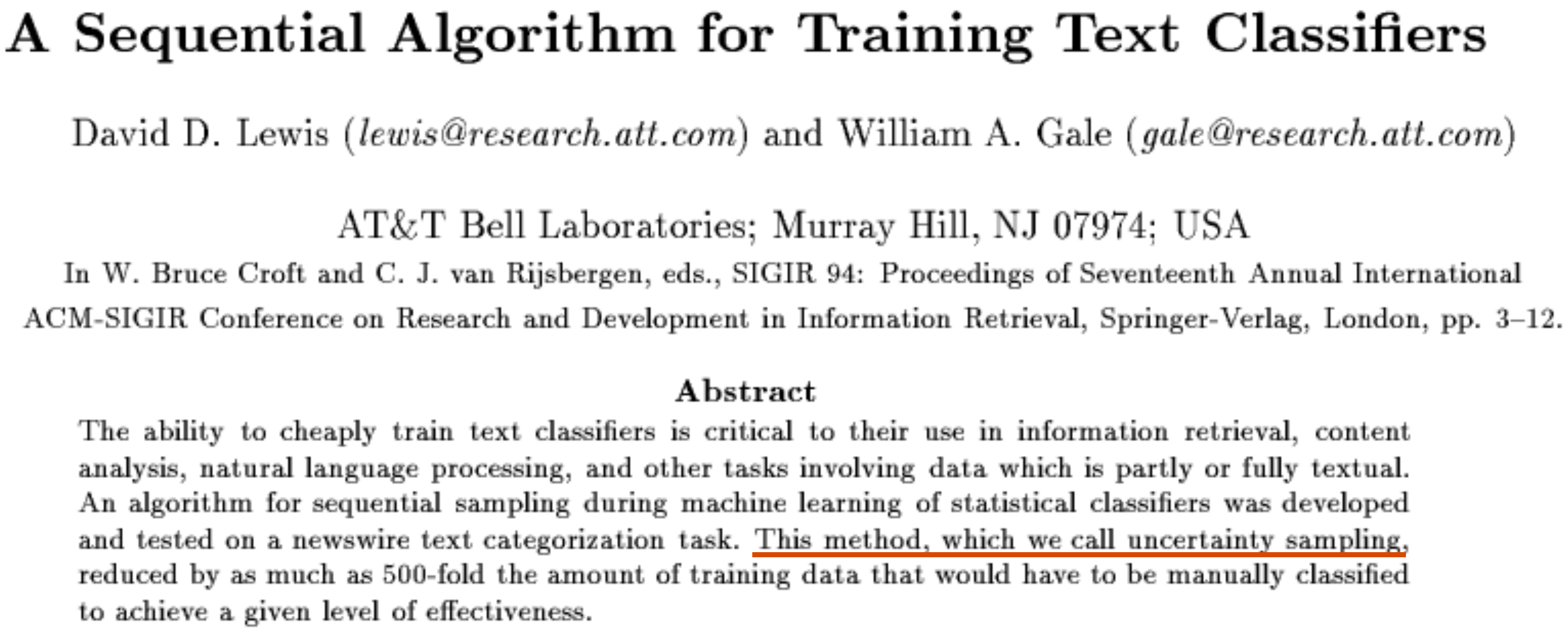
- Query the event that the current classifier is most uncertain about

If uncertainty is measured in Euclidean distance:



- Used trivially in SVMs, graphic models, etc.

1994



A Sequential Algorithm for Training Text Classifiers

David D. Lewis (*lewis@research.att.com*) and William A. Gale (*gale@research.att.com*)

AT&T Bell Laboratories; Murray Hill, NJ 07974; USA

In W. Bruce Croft and C. J. van Rijsbergen, eds., SIGIR 94: Proceedings of Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, Springer-Verlag, London, pp. 3–12.

Abstract

The ability to cheaply train text classifiers is critical to their use in information retrieval, content analysis, natural language processing, and other tasks involving data which is partly or fully textual. An algorithm for sequential sampling during machine learning of statistical classifiers was developed and tested on a newswire text categorization task. This method, which we call uncertainty sampling, reduced by as much as 500-fold the amount of training data that would have to be manually classified to achieve a given level of effectiveness.

Score Function

$$\text{score}_{uncert}(S_t) = \text{uncertainty}(P(S_t \mid O_t))$$

$$= H(S_t)$$

$$= \sum_i P(S_t = i) \log P(S_t = i)$$

Uncertainty Sampling Example

t	Sex	Age	Test A	Test B	Test C	S_t	$P(S_t)$	$H(S_t)$
1	M	20-30	0	1	1	?	0.02	0.043
2	F	20-30	0	1	0	?	0.01	0.024
3	F	30-40	1	0	0	?	0.05	0.086
4	F	60+	1	1	0	FALSE	0.12	0.159
5	M	10-20	0	1	0	?	0.01	0.024
6	M	20-30	1	1	1	?	0.96	0.073

Uncertainty Sampling Example

t	Sex	Age	Test A	Test B	Test C	S_t	$P(S_t)$	$H(S_t)$
1	M	20-30	0	1	1	?	0.01	0.024
2	F	20-30	0	1	0	?	0.02	0.043
3	F	30-40	1	0	0	?	0.04	0.073
4	F	60+	1	1	0	FALSE	0.00	0.00
5	M	10-20	0	1	0	TRUE	0.06	[✓ 0.112]
6	M	20-30	1	1	1	?	0.97	0.059

Uncertainty Sampling

GOOD: couldn’t be easier

GOOD: often performs pretty well

BAD: $H(S_t)$ measures information gain about the ***samples***, not the ***model***

↓

Sensitive to noisy samples

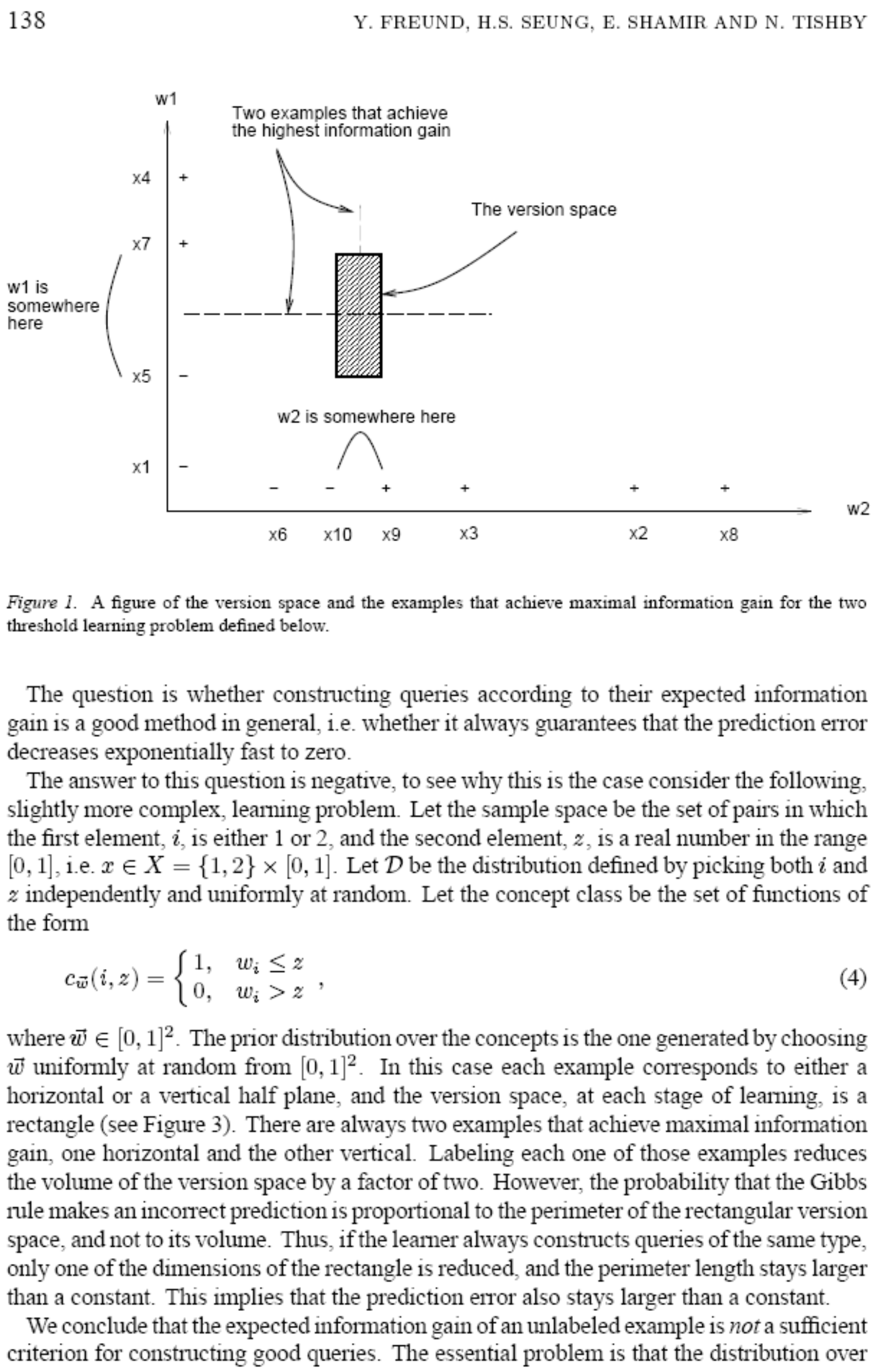

w1
Two examples that achieve the highest information gain
The version space
x4
x7
w1 is somewhere here
x5
w2 is somewhere here
x1
w2
x6
x10
x9
x3
x2
x8
+
+
−
−
−
−
+
+
+
+

Figure 1. A figure of the version space and the examples that achieve maximal information gain for the two threshold learning problem defined below.

The question is whether constructing queries according to their expected information gain is a good method in general, i.e. whether it always guarantees that the prediction error decreases exponentially fast to zero.

The answer to this question is negative, to see why this is the case consider the following, slightly more complex, learning problem. Let the sample space be the set of pairs in which the first element, i, is either 1 or 2, and the second element, z, is a real number in the range $[0,1]$, i.e. $x \in X = \{1,2\} \times [0,1]$. Let $\mathcal{D}$ be the distribution defined by picking both i and z independently and uniformly at random. Let the concept class be the set of functions of the form

$$c_{\vec{w}}(i,z) = \begin{cases} 1, & w_i \leq z \\ 0, & w_i > z \end{cases}, \tag{4}$$

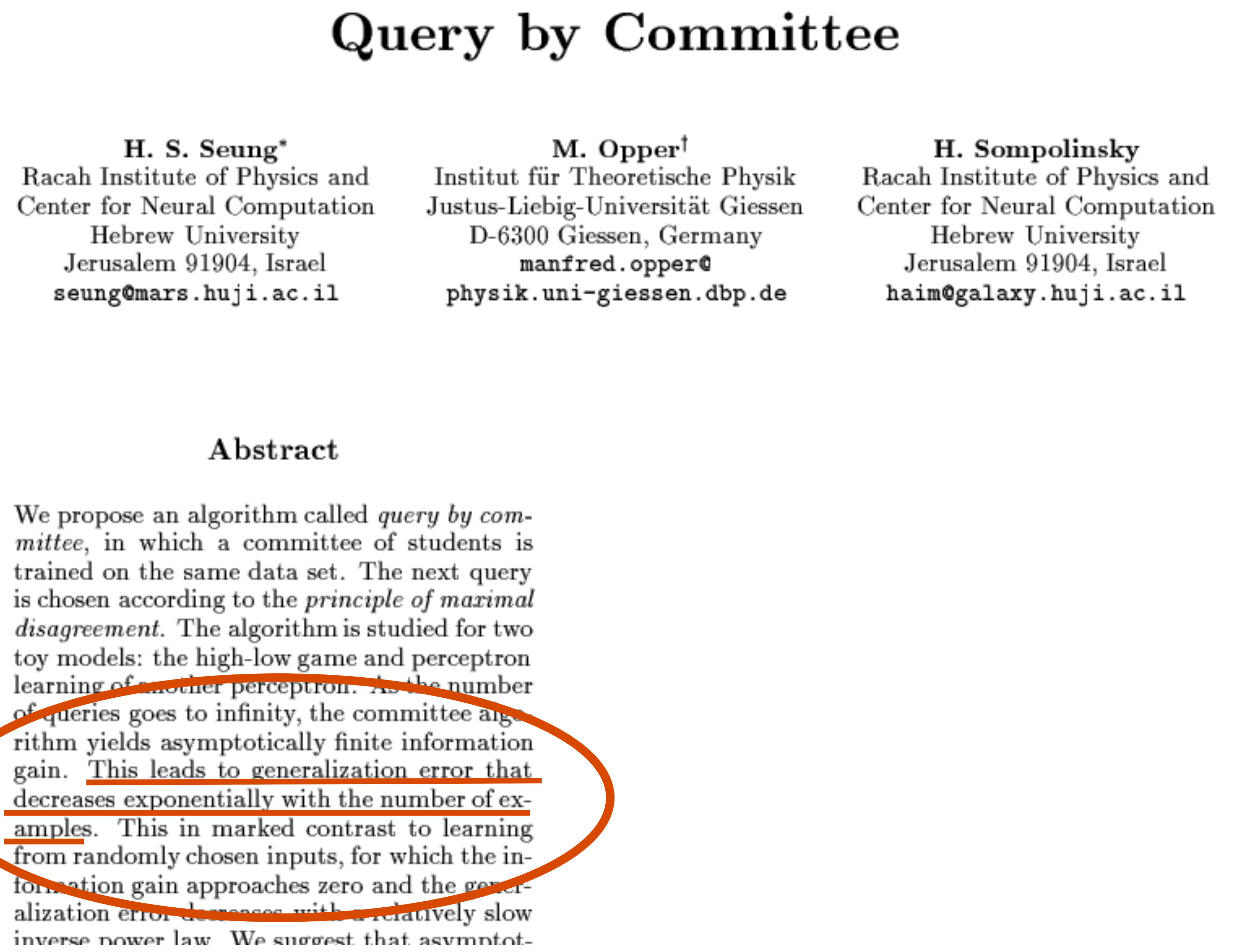
where $\vec{w} \in [0,1]^2$. The prior distribution over the concepts is the one generated by choosing $\vec{w}$ uniformly at random from $[0,1]^2$. In this case each example corresponds to either a horizontal or a vertical half plane, and the version space, at each stage of learning, is a rectangle (see Figure 3). There are always two examples that achieve maximal information gain, one horizontal and the other vertical. Labeling each one of those examples reduces the volume of the version space by a factor of two. However, the probability that the Gibbs rule makes an incorrect prediction is proportional to the perimeter of the rectangular version space, and not to its volume. Thus, if the learner always constructs queries of the same type, only one of the dimensions of the rectangle is reduced, and the perimeter length stays larger than a constant. This implies that the prediction error also stays larger than a constant.

We conclude that the expected information gain of an unlabeled example is *not* a sufficient criterion for constructing good queries. The essential problem is that the distribution over

If our objective is to reduce the prediction error, then

“the expected information gain of an unlabeled sample is NOT a sufficient criterion for constructing good queries”

1992



Query by Committee

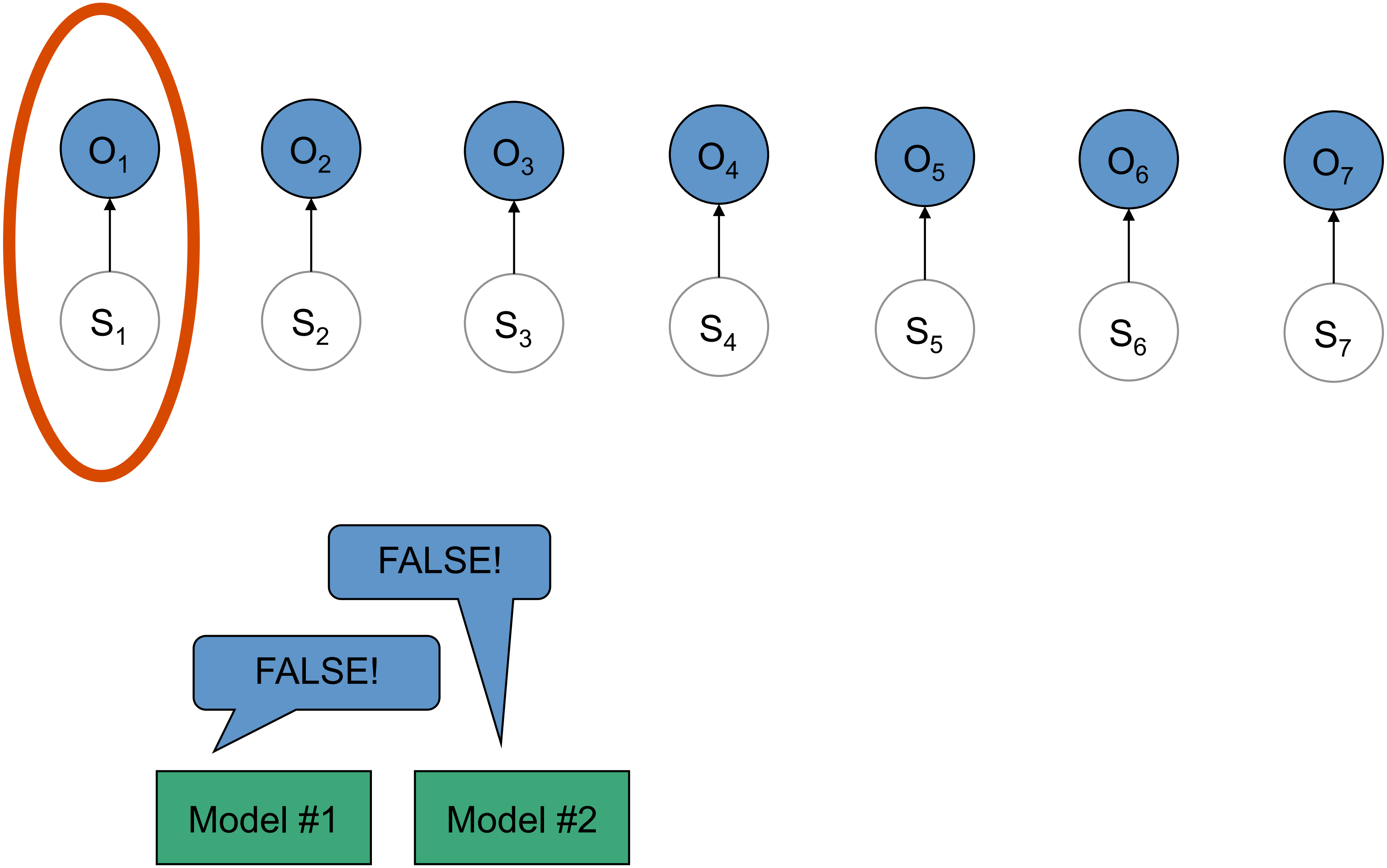
H. S. Seung[*]
Racah Institute of Physics and
Center for Neural Computation
Hebrew University
Jerusalem 91904, Israel
seung@mars.huji.ac.il

M. Opper[†]
Institut für Theoretische Physik
Justus-Liebig-Universität Giessen
D-6300 Giessen, Germany
manfred.opper@
physik.uni-giessen.dbp.de

H. Sompolinsky
Racah Institute of Physics and
Center for Neural Computation
Hebrew University
Jerusalem 91904, Israel
haim@galaxy.huji.ac.il

Abstract

We propose an algorithm called *query by committee*, in which a committee of students is trained on the same data set. The next query is chosen according to the *principle of maximal disagreement*. The algorithm is studied for two toy models: the high-low game and perceptron learning of another perceptron. As the number of queries goes to infinity, the committee algorithm yields asymptotically finite information gain. This leads to generalization error that decreases exponentially with the number of examples. This in marked contrast to learning from randomly chosen inputs, for which the information gain approaches zero and the generalization error decreases with a relatively slow inverse power law. We suggest that asymptot-

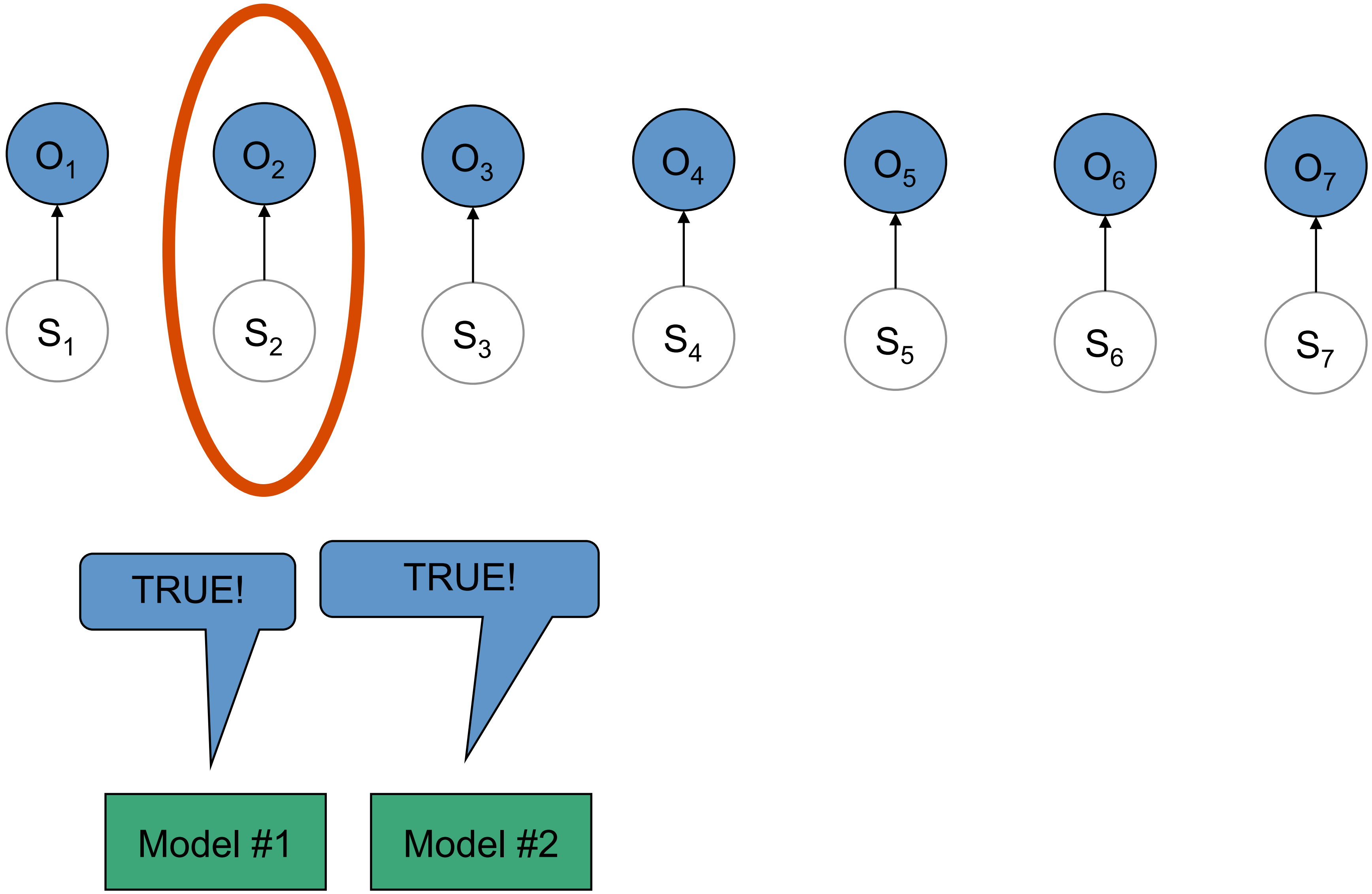


FALSE!

FALSE!

Model #1

Model #2

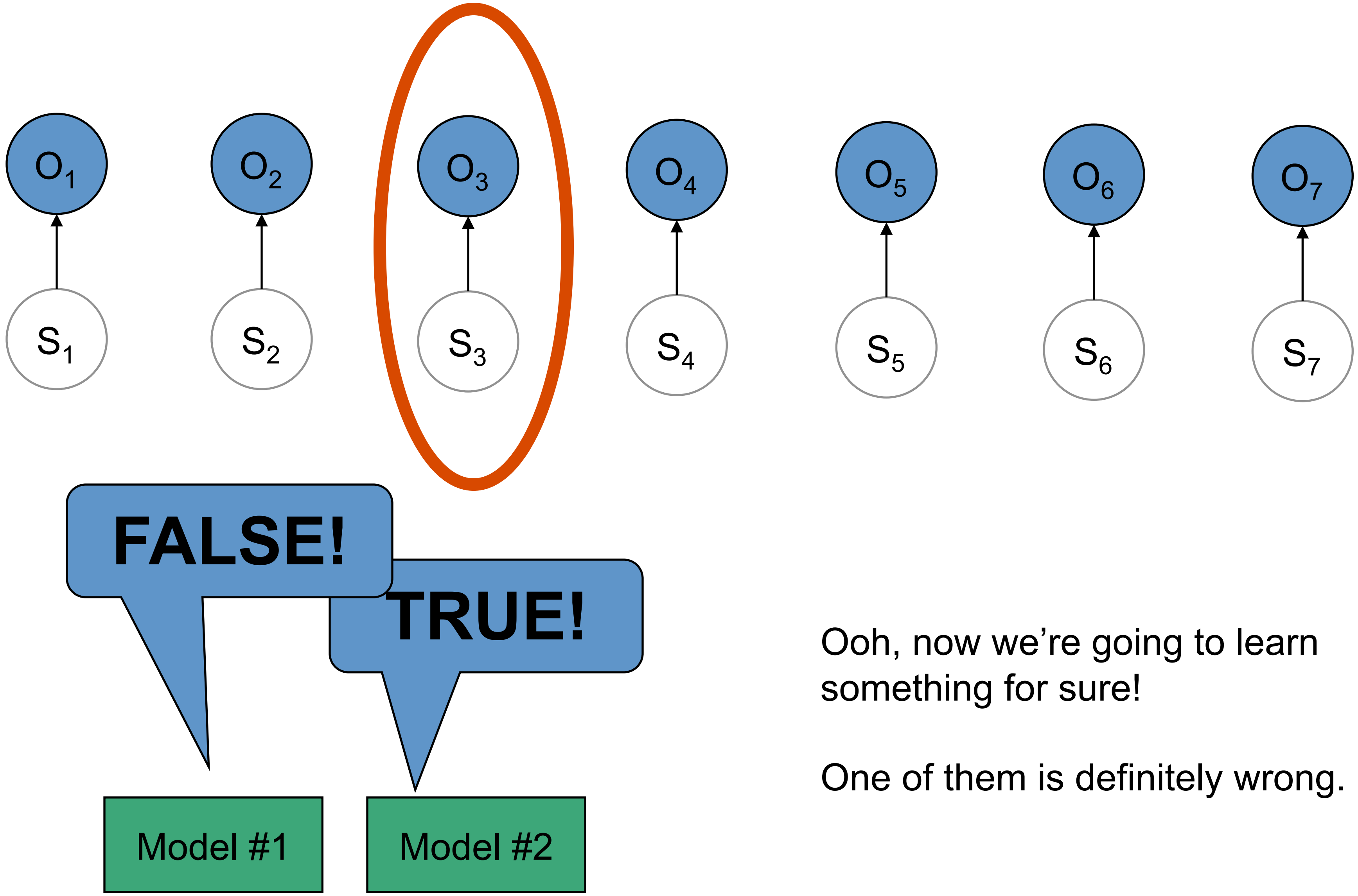


TRUE!

TRUE!

Model #1

Model #2



O_1 O_2 O_3 O_4 O_5 O_6 O_7

S_1 S_2 S_3 S_4 S_5 S_6 S_7

FALSE!

TRUE!

Model #1

Model #2

Ooh, now we’re going to learn something for sure!

One of them is definitely wrong.

The Original QBC Algorithm

As each example arrives…

1. Choose a committee, *C*, (usually of size 2) randomly from Version Space

2. Have each member of *C* classify it

3. If the committee disagrees, select it.