Project 0

- Part 3 : may be easier with monks (2 classes)
- 80/20 for monks: combine train and test
  only do 30 for random (can also do for info gain)
- Can do standard deviation, standard error, or confidence intervals

Consistent Learners

- A learner *L* using a hypothesis *H* and training data *D* is said to be a consistent learner if it always outputs a hypothesis with zero error on *D* whenever *H* contains such a hypothesis.
- By definition, a consistent learner must produce a hypothesis in the version space for *H* given *D*.
- Therefore, to bound the number of examples needed by a consistent learner, we just need to bound the number of examples needed to ensure that the version-space contains no hypotheses with unacceptably high error.

ε-Exhausted Version Space

- The version space, $VS_{H,D}$, is said to be ***ε-exhausted*** iff every hypothesis in it has true error less than or equal to ε.

- One can never be sure that the version-space is ε-exhausted, but one can bound the probability that it is not.

- **Theorem** (Haussler, 1988): If the hypothesis space *H* is finite, and *D* is a sequence of $m \geq 1$ independent random examples for some target concept *c*, then for any $0 \leq \varepsilon \leq 1$, the probability that the version space $VS_{H,D}$ is ***not*** ε-exhausted is less than or equal to:

$$|H| e^{-\varepsilon m}$$

Proof

- Let $H_{bad}=\{h_1,\ldots h_k\}$ be the subset of H with error > ε. The VS is not ε-exhausted if any of these are consistent with all *m* examples.
- A single $h_i \in H_{bad}$ is consistent with ***one*** example with probability:

$$P(\text{consist}(h_i, e_j)) \leq (1-\varepsilon)$$

- A single $h_i \in H_{bad}$ is consistent with ***all*** *m* independent random examples with probability: ?

Proof

- Let H_{bad}=$\{h_1,\ldots h_k\}$ be the subset of H with error > ε. The VS is not ε-exhausted if any of these are consistent with all *m* examples.
- A single $h_i \in H_{bad}$ is consistent with ***one*** example with at most probability:

$$P(\text{consist}(h_i, e_j)) \leq (1-\varepsilon)$$

- A single $h_i \in H_{bad}$ is consistent with ***all*** *m* independent random examples with probability:

$$P(\text{consist}(h_i, D)) \leq (1-\varepsilon)^m$$

- The probability that ***any*** $h_i \in H_{bad}$ is consistent with all *m* examples is:

$$P(\text{consist}(H_{bad}, D)) = P(\text{consist}(h_1, D) \vee \cdots \vee \text{consist}(h_k, D))$$

Proof (cont.)

- What's an upper bound on the probability of a disjunction?

Proof (cont.)

- Since the probability of a disjunction of events is ***at most*** the sum of the probabilities of the individual events:

$$P(\text{consist}(H_{bad}, D)) \leq |H_{bad}|(1-\varepsilon)^m$$

- Since: $|H_{\text{bad}}| \leq |H|$ and $(1-\varepsilon)^m \leq e^{-\varepsilon m}$, $0 \leq \varepsilon \leq 1$, $m \geq 0$

$$P(\text{consist}(H_{bad}, D)) \leq |H|e^{-\varepsilon m}$$

Q.E.D

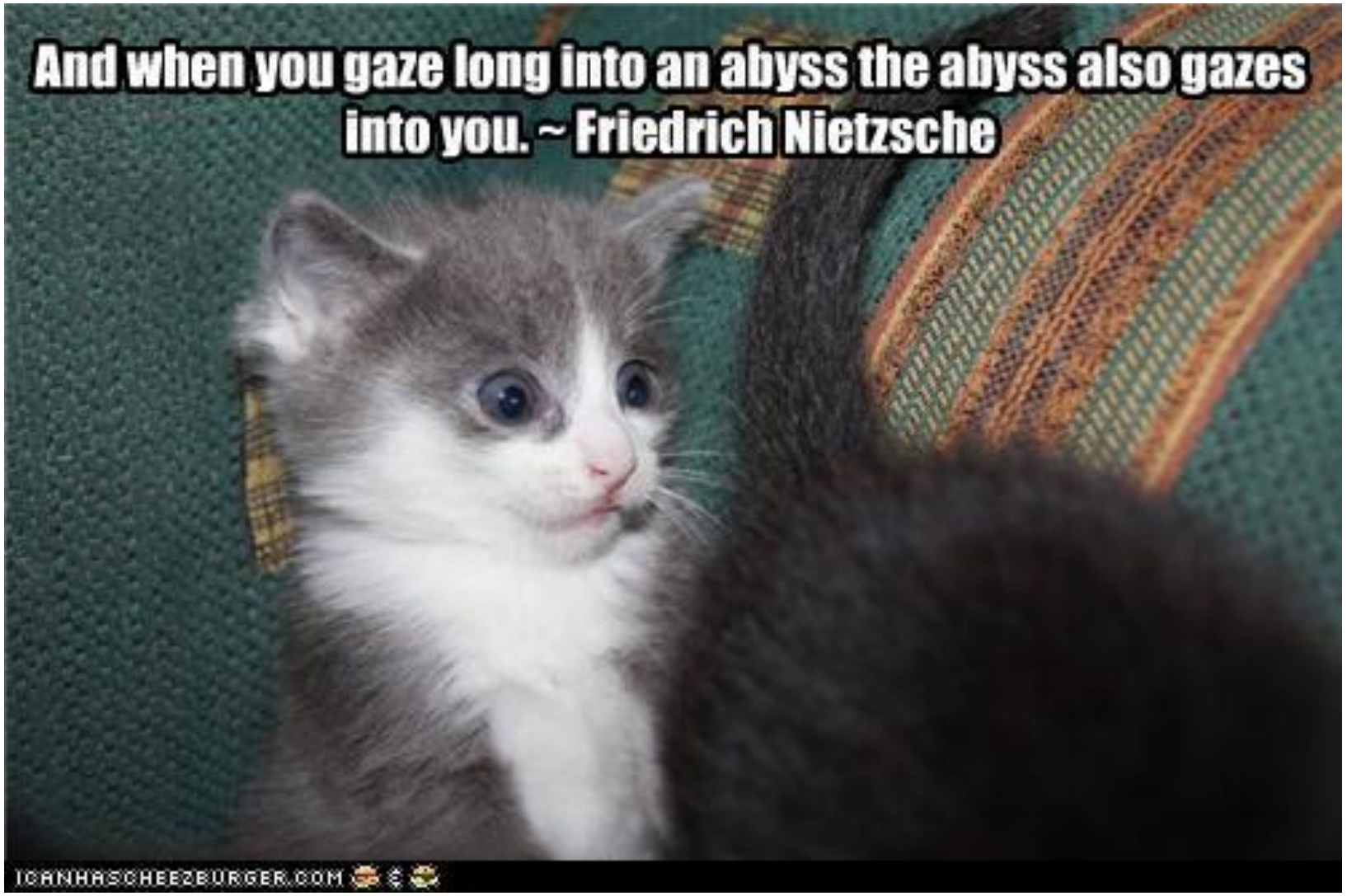
Sample Complexity Result

- Therefore, any consistent learner, given at least:

$$\left(\ln \frac{1}{\delta} + \ln |H| \right) / \varepsilon$$

  examples will produce a result that is PAC.
- Just need to determine the size of a hypothesis space to instantiate this result for learning specific classes of concepts.
- This gives a ***sufficient*** number of examples for PAC learning, but ***not*** a ***necessary*** number. Several approximations like that used to bound the probability of a disjunction make this a gross over-estimate in practice.

Let’s Work Through an Example



- Consider the class of concepts, *C*, consisting of axis-parallel hyper-rectangles in *n*-dimensional space.
  - Instances are described by *n* real-valued features and that an instance is classified as positive iff the value for each feature, x_i, falls in the range $(l_i <= x_i <= u_i)$ where l_i and u_i are separate lower and upper bounds specified for each feature.

- **C**onsider a discretized concept space where all bounds l_i and u_i must be integers in the interval (0, *m*), inclusive.
  - Zero-width hyper-rectangles along one or more dimensions are allowed since it is possible that $l_i = u_i$ for any feature.
- Using the size of this finite hypothesis space, give an upper bound on the number of randomly drawn training instances sufficient to assure that for any concept in *C*, any consistent learner using *H*=*C*, will, with probability at least 1-δ, output a hypothesis with error at most ε.

- Calculate a specific number of sufficient examples when *n*=3 (axis-parallel boxes in 3-D), *m*=10, and δ=ε= 0.01.

Since $l_i \leq u_i$ for each feature x_i, there are the following ranges on x_i:

- For $l_i = s$ there are $m + 1 - s$ values for u_i: $s, s + 1, s + 2, ...m$
- Therefore the total number of ranges on x_i is

$$\sum_{s=0}^{m}(m + 1 - s) = \qquad = \frac{(m + 1)(m + 2)}{2}$$

Since the range along each dimension can be selected independently, there are

$$|H| = \left(\frac{(m + 1)(m + 2)}{2}\right)^n$$

total possible hyper-rectangles.
Therefore

$$m' \geq \frac{1}{\epsilon}\left(\ln\frac{1}{\delta} + \ln\left(\frac{(m + 1)(m + 2)}{2}\right)^n\right)$$

$$m' \geq \frac{1}{\epsilon}\left(\ln\frac{1}{\delta} + n\ln\left(\frac{(m + 1)(m + 2)}{2}\right)\right)$$

examples are sufficient.
For $n = 3, m = 10, \delta = \epsilon = 0.01$

$$m' \geq 1718$$

Other Concept Classes

- *k*-term DNF: Disjunctions of at most *k* unbounded conjunctive terms: $T_1 \vee T_2 \vee \cdots \vee T_k$
  - ln(|*H*|)=O(*kn*)
- *k*-DNF: Disjunctions of any number of terms each limited to at most *k* literals: $((L_1 \wedge L_2 \wedge \cdots \wedge L_k) \vee (M_1 \wedge M_2 \wedge \cdots \wedge M_k) \vee \cdots$
  - ln(|*H*|)=O(n^k)
- *k*-clause CNF: Conjunctions of at most *k* unbounded disjunctive clauses: $C_1 \wedge C_2 \wedge \cdots \wedge C_k$
  - ln(|*H*|)=O(*kn*)
- *k*-CNF: Conjunctions of any number of clauses each limited to at most *k* literals:$((L_1 \vee L_2 \vee \cdots \vee L_k) \wedge (M_1 \vee M_2 \vee \cdots \vee M_k) \wedge \cdots$
  - ln(|*H*|)=O(n^k)

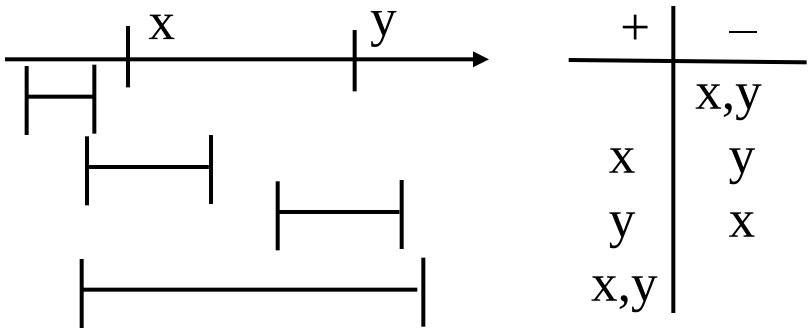
Therefore, all of these classes have polynomial sample complexity given a fixed value of *k*.

Infinite Hypothesis Spaces

- The preceding analysis was restricted to finite hypothesis spaces.
- Some infinite hypothesis spaces (such as those including real-valued thresholds or parameters) are more expressive than others.
  - Compare a rule allowing one threshold on a continuous feature (length<3cm) vs one allowing two thresholds (1cm<length<3cm).
- Need some measure of the expressiveness of infinite hypothesis spaces.
- The ***Vapnik-Chervonenkis*** (***VC***) ***dimension*** provides just such a measure, denoted VC(H).
- Analagous to $\ln|H|$, there are bounds for sample complexity using VC(H).

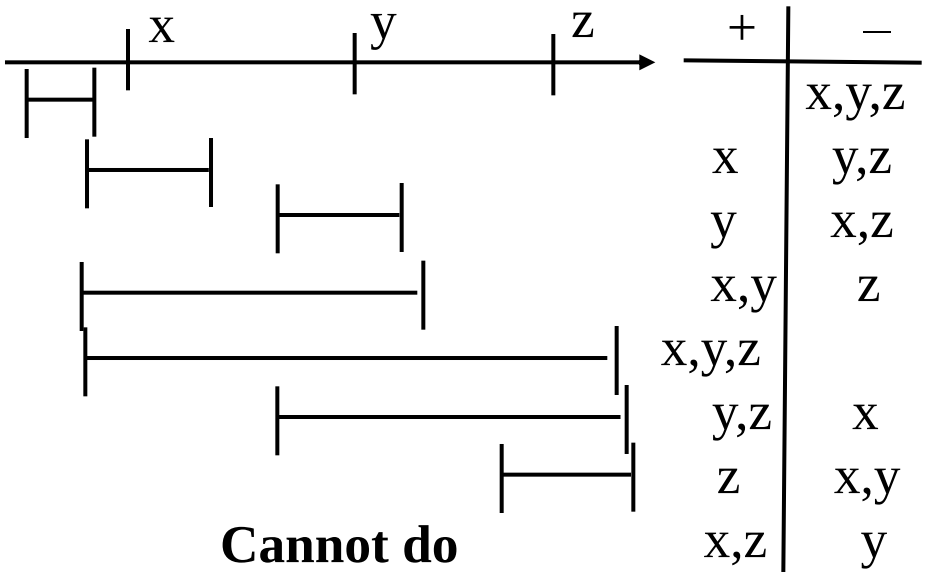
Shattering Instances

- A hypothesis space is said to shatter a set of instances iff for every partition of the instances into positive and negative, there is a hypothesis that produces that partition.
- For example, consider 2 instances described using a single real-valued feature being shattered by a single interval.



Shattering Instances (cont)

- But 3 instances cannot be shattered by a single interval.



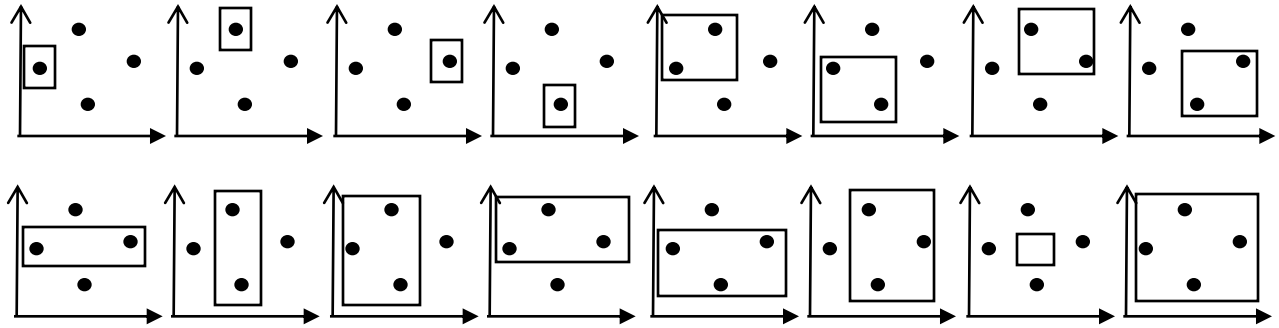
- Since there are 2^m partitions of m instances, in order for H to shatter instances: $|H| \geq 2^m$.

VC Dimension

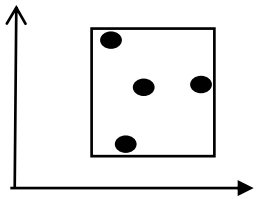
- An unbiased hypothesis space shatters the entire instance space.
- The larger the s d, the more expressive the h biased.
- The Vapnik-Che hypothesis space *H* defined ze of the largest finite subset of *X* arge finite subsets of *X* can
- If there exists at that can be shattered then V can be shattered, then
- For a single inte 2 instances can be shattered, bu VC(*H*) = 2.
- Since $|H| \geq 2^m$, $\log_2|H|$

VC Dimension Example

- Consider axis-parallel rectangles in the real-plane, i.e. conjunctions of intervals on two real-valued features. Some 4 instances can be shattered.

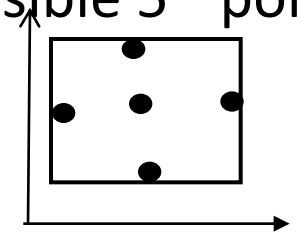


Some 4 instances cannot be shattered:



VC Dimension Example (cont)

- No five instances can be shattered since there can be at most 4 distinct extreme points (min and max on each of the 2 dimensions) and these 4 cannot be included without including any possible 5^{th} point.



- Therefore VC(*H*) = 4
- Generalizes to axis-parallel hyper-rectangles (conjunctions of intervals in *n* dimensions): VC(*H*)=2*n*.

Upper Bound on Sample Complexity with VC

- Using VC dimension as a measure of expressiveness, the following number of examples have been shown to be sufficient for PAC Learning (Blumer *et al.*, 1989).

$$\frac{1}{\varepsilon}\left(4\log_2\left(\frac{2}{\delta}\right)+8VC(H)\log_2\left(\frac{13}{\varepsilon}\right)\right)$$

- Compared to the previous result using $\ln|H|$, this bound has some extra constants and an extra $\log_2(1/\varepsilon)$ factor. Since $VC(H) \leq \log_2|H|$, this can provide a tighter upper bound on the number of examples needed for PAC learning.

Conjunctive Learning with Continuous Features

- Consider learning axis-parallel hyper-rectangles, conjunctions on intervals on *n* continuous features.
  - $1.2 \leq \text{length} \leq 10.5 \wedge 2.4 \leq \text{weight} \leq 5.7$
- Since VC(*H*)=2*n* sample complexity is

$$\frac{1}{\varepsilon}\left(4\log_2\left(\frac{2}{\delta}\right)+16n\log_2\left(\frac{13}{\varepsilon}\right)\right)$$

- Since the most-specific conjunctive algorithm can easily find the tightest interval along each dimension that covers all of the positive instances ($f_{\min} \leq f \leq f_{\max}$) and runs in linear time, O(|*D*|*n*), axis-parallel hyper-rectangles are PAC learnable.

Sample Complexity Lower Bound with VC

- There is also a general lower bound on the minimum number of examples necessary for PAC learning (Ehrenfeucht, *et al*., 1989):

  Consider any concept class *C* such that VC(*H*)≥2 any learner *L* and any 0<ε<1/8, 0<δ<1/100. Then there exists a distribution *D* and target concept in *C* such that if *L* observes fewer than:

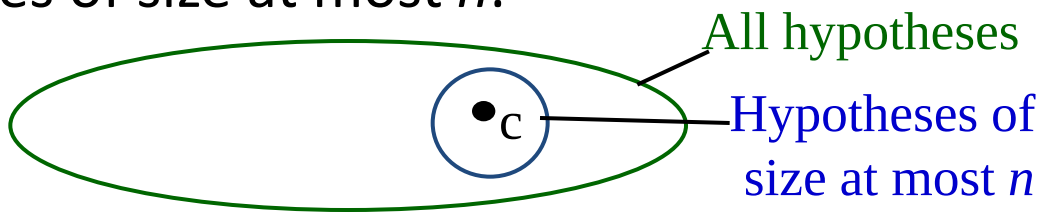
$$\max\left(\frac{1}{\varepsilon}\log_2\left(\frac{1}{\delta}\right), \frac{VC(C)-1}{32\varepsilon}\right)$$

  examples, then with probability at least δ, *L* outputs a hypothesis having error greater than ε.

- Ignoring constant factors, this lower bound is the same as the upper bound except for the extra $\log_2(1/\varepsilon)$ factor in the upper bound.

Analyzing a Preference Bias

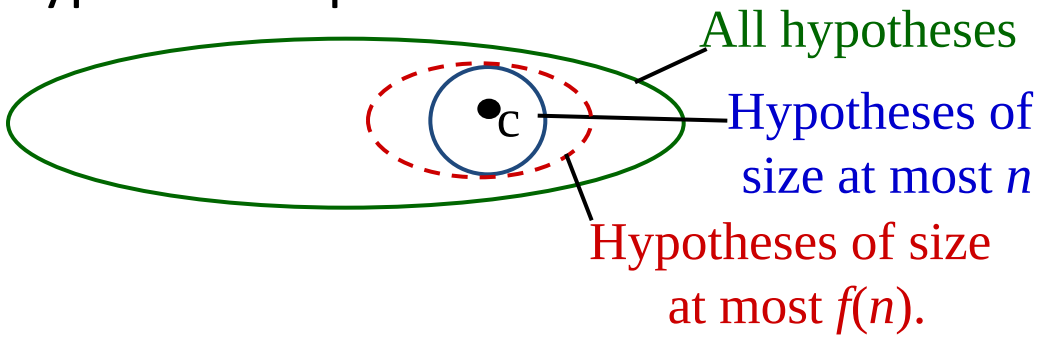
- Unclear how to apply previous results to an algorithm with a preference bias such as simplest decisions tree or simplest DNF.
- If the size of the correct concept is *n*, and the algorithm is guaranteed to return the minimum sized hypothesis consistent with the training data, then the algorithm will always return a hypothesis of size at most *n*, and the effective hypothesis space is all hypotheses of size at most *n*.



- Calculate $|H|$ or VC(H) of hypotheses of size at most *n* to determine sample complexity.

Computational Complexity and Preference Bias

- However, finding a minimum size hypothesis for most languages is computationally intractable.
- If one has an approximation algorithm that can bound the size of the constructed hypothesis to some polynomial function, $f(n)$, of the minimum size n, then can use this to define the effective hypothesis space.



- However, no worst case approximation bounds are known for practical learning algorithms (e.g. ID3).

"Occam's Razor" Result (Blumer *et al*., 1987)

- Assume that a concept can be represented using at most *n* bits in some representation language.
- Given a training set, assume the learner returns the consistent hypothesis representable with the least number of bits in this language.
- Therefore the effective hypothesis space is all concepts representable with at most *n* bits.
- Since *n* bits can code for at most 2^n hypotheses, $|H|=2^n$, so sample complexity if bounded by:

$$\left(\ln\frac{1}{\delta}+\ln 2^n\right)/\varepsilon=\left(\ln\frac{1}{\delta}+n\ln 2\right)/\varepsilon$$

- This result can be extended to approximation algorithms that can bound the size of the constructed hypothesis to at most n^k for some fixed constant *k* (just replace *n* with n^k)

Interpretation of “Occam’s Razor” Result

- Since the encoding is unconstrained it fails to provide any meaningful definition of “simplicity.”
- Hypothesis space could be any sufficiently small space, such as “the 2^n most complex boolean functions, where the complexity of a function is the size of its smallest DNF representation”
- Assumes that the correct concept (or a close approximation) is actually in the hypothesis space, so assumes *a priori* that the concept is simple.
- Does not provide a theoretical justification of Occam’s Razor as it is normally interpreted.

Mistake Bound

- How many mistakes before PAC?
- How many mistakes before exactly learning c?
- Optimal mistake bound (over *all* learning algos)?

- Similar to idea of *regret*

COLT Conclusions

- The PAC framework provides a theoretical framework for analyzing the effectiveness of learning algorithms.
- The sample complexity for any consistent learner using some hypothesis space, H, can be determined from a measure of its expressiveness $|H|$ or VC(H), quantifying bias and relating it to generalization.
- If sample complexity is tractable, then the computational complexity of finding a consistent hypothesis in H governs its PAC learnability.
- Constant factors are more important in sample complexity than in computational complexity, since our ability to gather data is generally not growing exponentially.
- Experimental results suggest that theoretical sample complexity bounds over-estimate the number of training instances needed in practice since they are worst-case upper bounds.

COLT Conclusions (cont)

- Additional results produced for analyzing:
  - Learning with queries
  - Learning with noisy data
  - Average case sample complexity given assumptions about the data distribution.
  - Learning finite automata
  - Learning neural networks
- Analyzing practical algorithms that use a preference bias is difficult.
- Some effective practical algorithms motivated by theoretical results:
  - Boosting
  - Support Vector Machines (SVM)

Example result

- http://www.cis.udel.edu/~case/slides/nugget-ushape.pdf