

## Lecture 02 of 42

# Representation Bias vs. Search Bias and Intro to Decision Trees

Friday, 25 January 2008

William H. Hsu

Department of Computing and Information Sciences, KSU

<http://www.kddresearch.org>

<http://www.cis.ksu.edu/~bhsu>

Readings:

Chapter 2, Mitchell

Section 5.1.2, Buchanan and Wilkins



CIS 732: Machine Learning and Pattern Recognition

Kansas State University  
Department of Computing and Information Sciences

## Lecture Outline

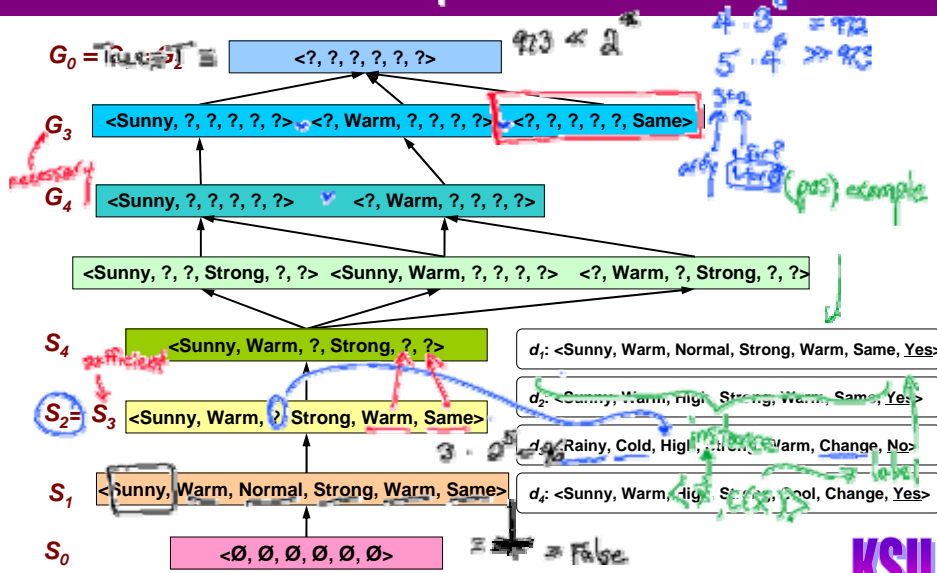
- **Read: Chapter 2, Mitchell; Section 5.1.2, Buchanan and Wilkins**
- **Suggested Exercises: 2.2, 2.3, 2.4, 2.6**
- **Taxonomy of Learning Systems**
- **Learning from Examples**
  - (Supervised) concept learning framework
  - Simple approach: assumes no noise; illustrates key concepts
- **General-to-Specific Ordering over Hypotheses**
  - Version space: partially-ordered set (poset) formalism
  - Candidate elimination algorithm
  - Inductive learning
- **Choosing New Examples**
- **Next Week**
  - The need for inductive bias: 2.7, Mitchell; 2.4.1-2.4.3, Shavlik and Dietterich
  - Computational learning theory (COLT): Chapter 7, Mitchell
  - PAC learning formalism: 7.2-7.4, Mitchell; 2.4.2, Shavlik and Dietterich



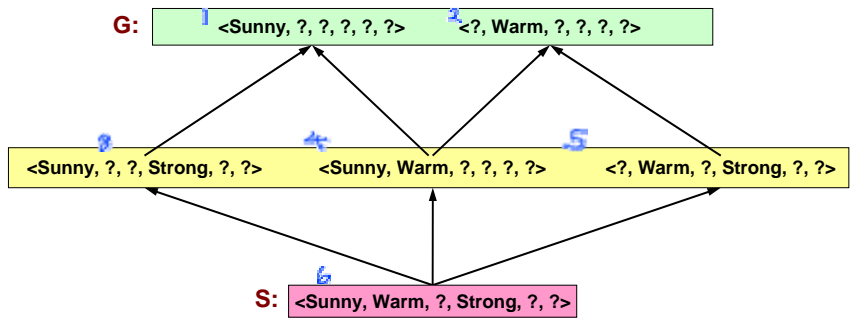
CIS 732: Machine Learning and Pattern Recognition

Kansas State University  
Department of Computing and Information Sciences

## Review: Example Trace



## Review: What Next Training Example?

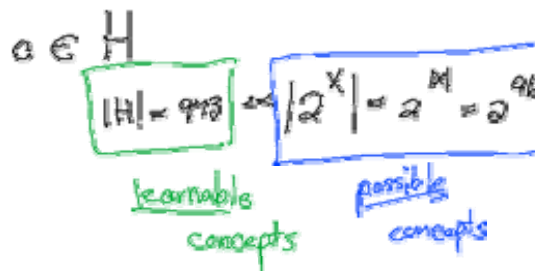


- Active Learning: What Query Should The Learner Make Next?
- How Should These Be Classified?
  - \*  $\langle \text{Sunny}, \text{Warm}, \text{Normal}, \text{Strong}, \text{Cool}, \text{Change} \rangle$  +
  - \*  $\langle \text{Rainy}, \text{Cold}, \text{Normal}, \text{Light}, \text{Warm}, \text{Same} \rangle$  -
  - \*  $\langle \text{Sunny}, \text{Warm}, \text{Normal}, \text{Light}, \text{Warm}, \text{Same} \rangle$  ? don't know - 3,5,6



## What Justifies This Inductive Leap?

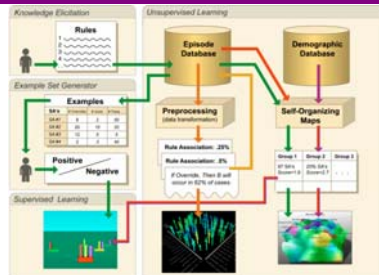
- **Example: Inductive Generalization**
  - Positive example: <Sunny, Warm, Normal, Strong, Cool, Change, Yes>
  - Positive example: <Sunny, Warm, Normal, Light, Warm, Same, Yes>
  - Induced S: <Sunny, Warm, Normal, ?, ?, ?>
- **Why Believe We Can Classify The Unseen?**
  - e.g., <Sunny, Warm, Normal, Strong, Warm, Same>
  - When is there enough information (in a new case) to make a prediction?



CIS 732: Machine Learning and Pattern Recognition

**KSU**  
Kansas State University  
Department of Computing and Information Sciences

## Interesting Applications



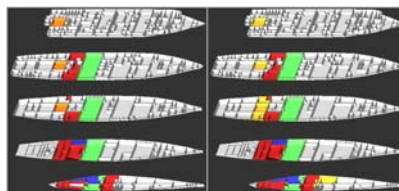
NCSA D2K - <http://alg.ncsa.uiuc.edu>

Database Mining



Cartia ThemeScapes - <http://www.cartia.com>

Reasoning (Inference, Decision Support)



DC-ARM - <http://www.kbs.ai.uiuc.edu>

Planning, Control

CIS 732: Machine Learning and Pattern Recognition

**KSU**  
Kansas State University  
Department of Computing and Information Sciences

## An Unbiased Learner

- **Example of A Biased  $H$** 
  - *Conjunctive* concepts with don't cares
  - What concepts can  $H$  not express? (Hint: what are its syntactic limitations?)
- **Idea**
  - Choose  $H'$  that expresses every teachable concept
  - i.e.,  $H'$  is the power set of  $X$
  - Recall:  $|A \rightarrow B| = |B|^{|A|}$  ( $A = X$ ;  $B = \{\text{labels}\}$ ;  $H' = A \rightarrow B$ )
  - $\{\{\text{Rainy, Sunny}\} \times \{\text{Warm, Cold}\} \times \{\text{Normal, High}\} \times \{\text{None, Mild, Strong}\} \times \{\text{Cool, Warm}\} \times \{\text{Same, Change}\}\} \rightarrow \{0, 1\}$
- **An Exhaustive Hypothesis Language**
  - Consider:  $H' =$  disjunctions ( $\vee$ ), conjunctions ( $\wedge$ ), negations ( $\neg$ ) over previous  $H$
  - $|H'| = 2^{(2 \cdot 2 \cdot 2 \cdot 3 \cdot 2 \cdot 2)} = 2^{96}$ ;  $|H| = 1 + (3 \cdot 3 \cdot 3 \cdot 4 \cdot 3 \cdot 3) = 973$
- **What Are  $S, G$  For The Hypothesis Language  $H'$ ?**
  - $S \leftarrow$  disjunction of all positive examples
  - $G \leftarrow$  conjunction of all negated negative examples

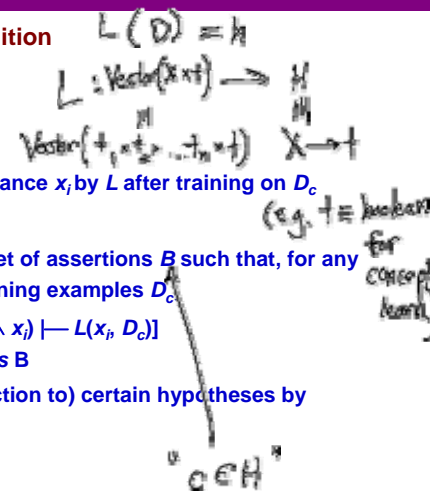


## Inductive Bias

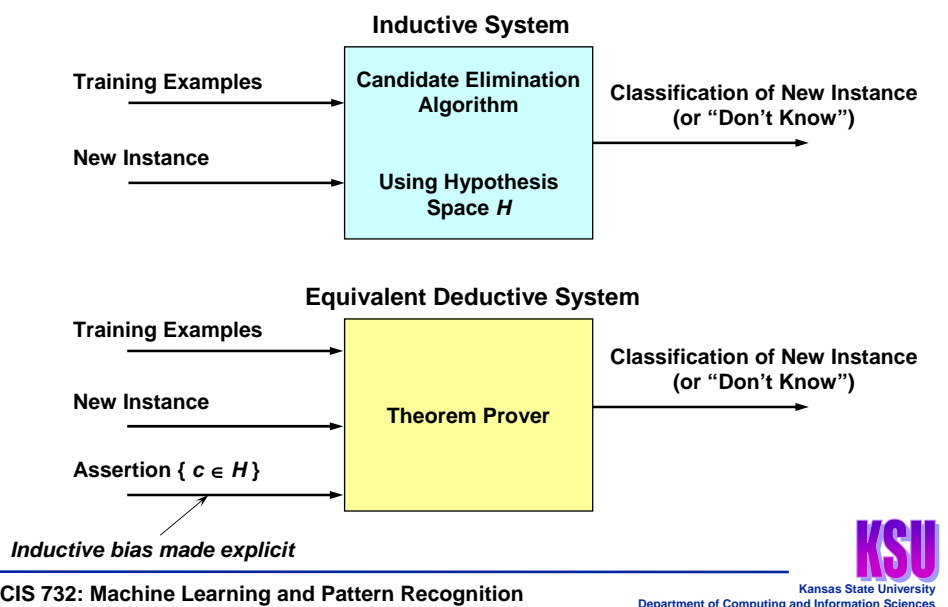
- **Components of An Inductive Bias Definition**
  - Concept learning algorithm  $L$
  - Instances  $X$ , target concept  $c$
  - Training examples  $D_c = \{ \langle x, c(x) \rangle \}$
  - $L(x_i, D_c)$  = classification assigned to instance  $x_i$  by  $L$  after training on  $D_c$
- **Definition**
  - The inductive bias of  $L$  is any minimal set of assertions  $B$  such that, for any target concept  $c$  and corresponding training examples  $D_c$ 

$$\forall x_i \in X. [(B \wedge D_c \wedge x_i) \vdash L(x_i, D_c)]$$

where  $A \vdash B$  means  $A$  *logically entails*  $B$
  - Informal idea: preference for (i.e., restriction to) certain hypotheses by structural (syntactic) means
- **Rationale**
  - Prior assumptions regarding target concept
  - Basis for inductive generalization

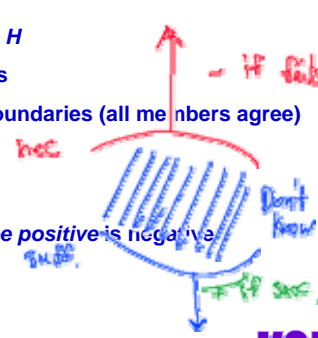


## Inductive Systems and Equivalent Deductive Systems



## Three Learners with Different Biases

- **Rote Learner**
  - Weakest bias: anything seen before, i.e., no bias
  - Store examples
  - Classify  $x$  if and only if it matches previously observed example
- **Version Space Candidate Elimination Algorithm**
  - Stronger bias: concepts belonging to conjunctive  $H$
  - Store extremal generalizations and specializations
  - Classify  $x$  if and only if it "falls within"  $S$  and  $G$  boundaries (all members agree)
- **Find-S**
  - Even stronger bias: most specific hypothesis
  - Prior assumption: any instance *not observed to be positive* is negative
  - Classify  $x$  based on  $S$  set



## Views of Learning

- **Removal of (Remaining) Uncertainty**
  - Suppose unknown function was *known* to be *m-of-n* Boolean function
  - Could use training data to infer the function
- **Learning and Hypothesis Languages**
  - Possible approach to *guess a good, small hypothesis language*:
    - Start with a very small language
    - Enlarge until it contains a hypothesis that fits the data
  - Inductive bias
    - Preference for certain languages
    - Analogous to data compression (removal of redundancy)
    - Later: coding the “model” versus coding the “uncertainty” (error)
- **We Could Be Wrong!**
  - Prior knowledge could be wrong (e.g.,  $y = x_4 \wedge$  one-of  $(x_1, x_3)$  also consistent)
  - If guessed language was wrong, errors will occur on new cases



CIS 732: Machine Learning and Pattern Recognition

Kansas State University  
Department of Computing and Information Sciences

## Two Strategies for Machine Learning

- **Develop Ways to Express Prior Knowledge**
  - Role of prior knowledge: guides search for hypotheses / hypothesis languages
  - Expression languages for prior knowledge
    - Rule grammars; stochastic models; etc.
    - Restrictions on computational models; other (formal) specification methods
- **Develop Flexible Hypothesis Spaces**
  - Structured collections of hypotheses
    - Agglomeration: nested collections (hierarchies)
    - Partitioning: decision trees, lists, rules
    - Neural networks; cases, etc.
  - Hypothesis spaces of adaptive size
- **Either Case: Develop Algorithms for Finding A Hypothesis That Fits Well**
  - Ideally, will generalize well
- **Later: Bias *Optimization* (Meta-Learning, Wrappers)**



CIS 732: Machine Learning and Pattern Recognition

Kansas State University  
Department of Computing and Information Sciences

## Computational Learning Theory

- What General Laws Constrain Inductive Learning?
- What Learning Problems Can Be Solved?
- When Can We Trust The Output of A Learning Algorithm?
- We Seek Theory To Relate:
  - Probability of successful learning
  - Number of training examples
  - Complexity of hypothesis space
  - Accuracy to which target concept is approximated
  - Manner in which training examples are presented



## Prototypical Concept Learning Task

- Given
  - Instances  $X$ : possible days, each described by attributes *Sky, AirTemp, Humidity, Wind, Water, Forecast*
  - Target function  $c \equiv \text{EnjoySport}: X \rightarrow \mathbf{H}$
  - Hypotheses  $H$ : conjunctions of literals, e.g.,  
 $\langle ?, \text{Cold}, \text{High}, ?, ?, ? \rangle$
  - Training examples  $D$ : positive and negative examples of the target function  
 $\langle x_1, c(x_1) \rangle, \langle x_2, c(x_2) \rangle, \dots, \langle x_m, c(x_m) \rangle$
- Determine
  - A hypothesis  $h$  in  $H$  such that  $h(x) = c(x)$  for all  $x$  in  $D$ ?
  - A hypothesis  $h$  in  $H$  such that  $h(x) = c(x)$  for all  $x$  in  $X$ ?



## Sample Complexity

- **How Many Training Examples Sufficient To Learn Target Concept?**
- **Scenario 1: Active Learning**
  - Learner proposes instances, as *queries* to teacher
  - Query (learner): instance  $x$
  - Answer (teacher):  $c(x)$
- **Scenario 2: Passive Learning from Teacher-Selected Examples**
  - Teacher (who knows  $c$ ) provides training examples
  - Sequence of examples (teacher):  $\{<x_i, c(x_i)>\}$
  - Teacher *may or may not* be helpful, optimal
- **Scenario 3: Passive Learning from Teacher-Annotated Examples**
  - Random process (e.g., nature) proposes instances
  - Instance  $x$  generated randomly, teacher provides  $c(x)$



## Sample Complexity: Scenario 1

- **Learner Proposes Instance  $x$**
- **Teacher Provides  $c(x)$** 
  - *Comprehensibility*: assume  $c$  is in learner's hypothesis space  $H$
  - A form of inductive bias (sometimes nontrivial!)
- **Optimal Query Strategy: Play 20 Questions**
  - Pick instance  $x$  such that half of hypotheses in  $V_S$  classify  $x$  positive, half classify  $x$  negative
  - When this is possible, need  $\lceil \log_2 |H| \rceil$  queries to learn  $c$
  - When not possible, need even more



## Sample Complexity: Scenario 2

- **Teacher Provides Training Examples**
  - Teacher: agent who knows  $c$
  - Assume  $c$  is in learner's hypothesis space  $H$  (as in Scenario 1)
- **Optimal Teaching Strategy: Depends upon  $H$  Used by Learner**
  - Consider case:  $H =$  conjunctions of up to  $n$  boolean literals and their negations
  - e.g.,  $(AirTemp = Warm) \wedge (Wind = Strong)$ , where  $AirTemp, Wind, \dots$  each have 2 possible values
  - Complexity
    - If  $n$  possible boolean attributes in  $H$ ,  $n + 1$  examples suffice
    - Why?



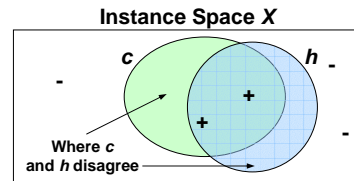
## Sample Complexity: Scenario 3

- **Given**
  - Set of instances  $X$
  - Set of hypotheses  $H$
  - Set of possible target concepts  $C$
  - Training instances generated by a *fixed*, unknown probability distribution  $D$  over  $X$
- **Learner Observes Sequence  $D$** 
  - $D$ : training examples of form  $\langle x, c(x) \rangle$  for target concept  $c \in C$
  - Instances  $x$  are drawn from distribution  $D$
  - Teacher provides target value  $c(x)$  for each
- **Learner Must Output Hypothesis  $h$  Estimating  $c$** 
  - $h$  evaluated on performance on subsequent instances
  - Instances still drawn according to  $D$
- **Note: Probabilistic Instances, Noise-Free Classifications**

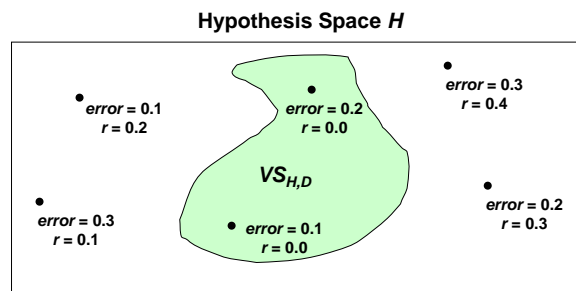


# True Error of A Hypothesis

- **Definition**
  - The **true error** (denoted  $error_D(h)$ ) of hypothesis  $h$  with respect to target concept  $c$  and distribution  $D$  is the probability that  $h$  will misclassify an instance drawn at random according to  $D$ .
  - $error_D(h) \equiv Pr_{x \in D}[c(x) \neq h(x)]$
- **Two Notions of Error**
  - **Training error** of hypothesis  $h$  with respect to target concept  $c$ : how often  $h(x) \neq c(x)$  over training instances
  - **True error** of hypothesis  $h$  with respect to target concept  $c$ : how often  $h(x) \neq c(x)$  over future random instances
- **Our Concern**
  - Can we bound true error of  $h$  (given training error of  $h$ )?
  - First consider when training error of  $h$  is zero (i.e,  $h \in VS_{H,D}$ )



# Exhausting The Version Space



- **Definition**
  - The version space  $VS_{H,D}$  is said to be  **$\epsilon$ -exhausted** with respect to  $c$  and  $D$ , if every hypothesis  $h$  in  $VS_{H,D}$  has error less than  $\epsilon$  with respect to  $c$  and  $D$ .
  - $\forall h \in VS_{H,D} \cdot error_D(h) < \epsilon$



## An Unbiased Learner

- **Example of A Biased  $H$** 
  - *Conjunctive* concepts with don't cares
  - What concepts can  $H$  not express? (Hint: what are its syntactic limitations?)
- **Idea**
  - Choose  $H'$  that expresses every teachable concept
  - i.e.,  $H'$  is the power set of  $X$
  - Recall:  $|A \rightarrow B| = |B|^{|A|}$  ( $A = X$ ;  $B = \{\text{labels}\}$ ;  $H' = A \rightarrow B$ )
  - $\{\{\text{Rainy, Sunny}\} \times \{\text{Warm, Cold}\} \times \{\text{Normal, High}\} \times \{\text{None, Mild, Strong}\} \times \{\text{Cool, Warm}\} \times \{\text{Same, Change}\}\} \rightarrow \{0, 1\}$
- **An Exhaustive Hypothesis Language**
  - Consider:  $H' =$  disjunctions ( $\vee$ ), conjunctions ( $\wedge$ ), negations ( $\neg$ ) over previous  $H$
  - $|H'| = 2^{(2 \cdot 2 \cdot 2 \cdot 3 \cdot 2 \cdot 2)} = 2^{96}$ ;  $|H| = 1 + (3 \cdot 3 \cdot 3 \cdot 4 \cdot 3 \cdot 3) = 973$
- **What Are S, G For The Hypothesis Language  $H'$ ?**
  - $S \leftarrow$  *disjunction of all positive examples*
  - $G \leftarrow$  *conjunction of all negated negative examples*



CIS 732: Machine Learning and Pattern Recognition

Kansas State University  
Department of Computing and Information Sciences

## Inductive Bias

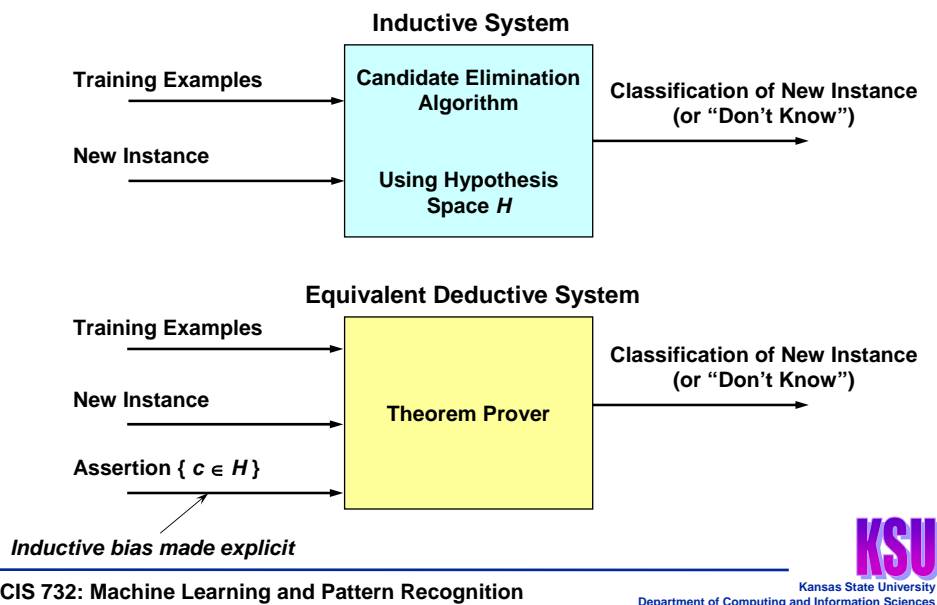
- **Components of An Inductive Bias Definition**
  - Concept learning algorithm  $L$
  - Instances  $X$ , target concept  $c$
  - Training examples  $D_c = \{ \langle x, c(x) \rangle \}$
  - $L(x_i, D_c)$  = classification assigned to instance  $x_i$  by  $L$  after training on  $D_c$
- **Definition**
  - The inductive bias of  $L$  is any minimal set of assertions  $B$  such that, for any target concept  $c$  and corresponding training examples  $D_c$ ,
    - $\forall x_i \in X. [(B \wedge D_c \wedge x_i) \vdash L(x_i, D_c)]$
    - where  $A \vdash B$  means  $A$  *logically entails*  $B$
  - Informal idea: preference for (i.e., restriction to) certain hypotheses by structural (syntactic) means
- **Rationale**
  - Prior assumptions regarding target concept
  - Basis for inductive generalization



CIS 732: Machine Learning and Pattern Recognition

Kansas State University  
Department of Computing and Information Sciences

## Inductive Systems and Equivalent Deductive Systems



## Three Learners with Different Biases

- **Rote Learner**
  - Weakest bias: anything seen before, i.e., no bias
  - Store examples
  - Classify  $x$  if and only if it matches previously observed example
- **Version Space Candidate Elimination Algorithm**
  - Stronger bias: concepts belonging to conjunctive  $H$
  - Store extremal generalizations and specializations
  - Classify  $x$  if and only if it "falls within"  $S$  and  $G$  boundaries (all members agree)
- **Find-S**
  - Even stronger bias: most specific hypothesis
  - Prior assumption: any instance *not observed to be positive* is negative
  - Classify  $x$  based on  $S$  set

## Number of Examples Required to Exhaust The Version Space

- **How Many Examples Will  $\epsilon$ -Exhaust The Version Space?**
- **Theorem [Haussler, 1988]**
  - If the hypothesis space  $H$  is finite, and  $D$  is a sequence of  $m \geq 1$  independent random examples of some target concept  $c$ , then for any  $0 \leq \epsilon \leq 1$ , the probability that the version space with respect to  $H$  and  $D$  is not  $\epsilon$ -exhausted (with respect to  $c$ ) is less than or equal to
 
$$|H| e^{-\epsilon m}$$
- **Important Result!**
  - *Bounds the probability* that any consistent learner will output a hypothesis  $h$  with  $error(h) \geq \epsilon$
  - Want this probability to be below a specified threshold  $\delta$ 

$$|H| e^{-\epsilon m} \leq \delta$$
  - To achieve, solve inequality for  $m$ : let
 
$$m \geq 1/\epsilon (\ln |H| + \ln (1/\delta))$$
  - Need to see *at least* this many examples



## Learning Conjunctions of Boolean Literals

- **How Many Examples Are Sufficient?**
  - Specification - ensure that with probability *at least*  $(1 - \delta)$ :
 
$$\text{Every } h \text{ in } VS_{H,D} \text{ satisfies } error_D(h) < \epsilon$$
  - “The probability of an  $\epsilon$ -bad hypothesis ( $error_D(h) \geq \epsilon$ ) is no more than  $\delta$ ”
  - Use our theorem:
 
$$m \geq 1/\epsilon (\ln |H| + \ln (1/\delta))$$
  - $H$ : conjunctions of constraints on up to  $n$  boolean attributes ( $n$  boolean literals)
  - $|H| = 3^n$ ,  $m \geq 1/\epsilon (\ln 3^n + \ln (1/\delta)) = 1/\epsilon (n \ln 3 + \ln (1/\delta))$
- **How About *EnjoySport*?**
  - $H$  as given in *EnjoySport* (conjunctive concepts with don't cares)
    - $|H| = 973$
    - $m \geq 1/\epsilon (\ln |H| + \ln (1/\delta))$
  - Example goal: probability  $1 - \delta = 95\%$  of hypotheses with  $error_D(h) < 0.1$
  - $m \geq 1/0.1 (\ln 973 + \ln (1/0.05)) \approx 98.8$



## PAC Learning

- **Terms Considered**
  - Class  $C$  of possible concepts
  - Set of instances  $X$
  - Length  $n$  (in attributes) of each instance
  - Learner  $L$
  - Hypothesis space  $H$
  - Error parameter (error bound)  $\epsilon$
  - Confidence parameter (excess error probability bound)  $\delta$
  - $size(c)$  = the encoding length of  $c$ , assuming some representation
- **Definition**
  - $C$  is **PAC-learnable** by  $L$  using  $H$  if for all  $c \in C$ , distributions  $D$  over  $X$ ,  $\epsilon$  such that  $0 < \epsilon < 1/2$ , and  $\delta$  such that  $0 < \delta < 1/2$ , learner  $L$  will, with probability at least  $(1 - \delta)$ , output a hypothesis  $h \in H$  such that  $error_D(h) \leq \epsilon$
  - $C$  is **efficiently PAC-learnable** if  $L$  runs in time polynomial in  $1/\epsilon, 1/\delta, n, size(c)$



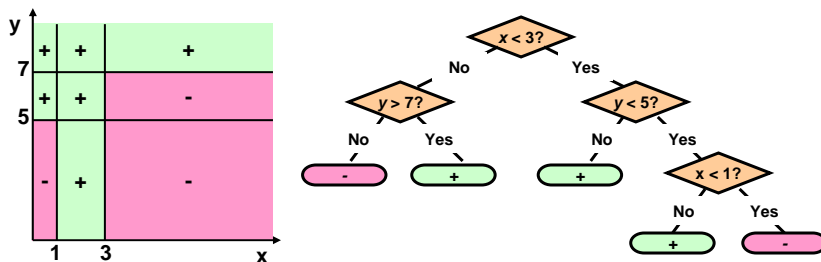
## When to Consider Using Decision Trees

- **Instances Describable by Attribute-Value Pairs**
- **Target Function Is Discrete Valued**
- **Disjunctive Hypothesis May Be Required**
- **Possibly Noisy Training Data**
- **Examples**
  - Equipment or medical diagnosis
  - Risk analysis
    - Credit, loans
    - Insurance
    - Consumer fraud
    - Employee fraud
  - Modeling calendar scheduling preferences (predicting quality of candidate time)



## Decision Trees and Decision Boundaries

- **Instances Usually Represented Using Discrete Valued Attributes**
  - Typical types
    - Nominal ({red, yellow, green})
    - Quantized ({low, medium, high})
  - Handling numerical values
    - Discretization, a form of vector quantization (e.g., histogramming)
    - Using thresholds for splitting nodes
- **Example: Dividing Instance Space into Axis-Parallel Rectangles**

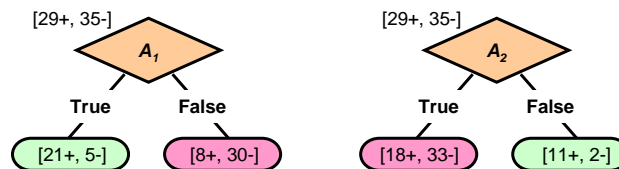


CIS 732: Machine Learning and Pattern Recognition

Kansas State University  
Department of Computing and Information Sciences

## Decision Tree Learning: Top-Down Induction (ID3)

- **Algorithm Build-DT (Examples, Attributes)**
  - IF all examples have the same label THEN RETURN (leaf node with label)
  - ELSE
  - IF set of attributes is empty THEN RETURN (leaf with majority label)
  - ELSE
  - Choose best attribute  $A$  as root
  - FOR each value  $v$  of  $A$ 
    - Create a branch out of the root for the condition  $A = v$
    - IF  $\{x \in \text{Examples}: x.A = v\} = \emptyset$  THEN RETURN (leaf with majority label)
    - ELSE Build-DT ( $\{x \in \text{Examples}: x.A = v\}$ , Attributes  $\sim \{A\}$ )
- **But Which Attribute Is Best?**



CIS 732: Machine Learning and Pattern Recognition

Kansas State University  
Department of Computing and Information Sciences

## Terminology

- **Inductive Bias**
  - Strength of inductive bias: how *few* hypotheses?
  - Specific biases: based on specific languages
- **Hypothesis Language**
  - “Searchable subset” of the space of possible descriptors
  - *m-of-n*, conjunctive, disjunctive, clauses
  - Ability to represent a concept
- **PAC Learning**
  - Probably Approximately Correct
  - Computational Learning Theory (COLT)
  - True error versus training error
  - Notation: distribution  $D$ ,  $error_D(h)$ ,  $\epsilon$ -bad with probability  $\delta$
  - $\epsilon$ -exhaustion: every hypothesis in  $VS_{H,D}$  has  $error_D(h) < \epsilon$
  - PAC-learnability: for  $c \in C, X, n, L, H, \epsilon, \delta$



Kansas State University  
Department of Computing and Information Sciences

CIS 732: Machine Learning and Pattern Recognition

## Summary Points

- **Inductive Leaps Possible Only if Learner Is Biased**
  - Futility of learning without bias
  - Strength of inductive bias: proportional to restrictions on hypotheses
- **Modeling Inductive Learners with Equivalent Deductive Systems**
  - Representing inductive learning as theorem proving
  - Equivalent learning and inference problems
- **Syntactic Restrictions**
  - Example: *m-of-n* concept
- **Views of Learning and Strategies**
  - Removing uncertainty (“data compression”)
  - Role of knowledge
- **Introduction to Computational Learning Theory (COLT)**
  - Things COLT attempts to measure
  - Probably-Approximately-Correct (PAC) learning framework
- **Next Lecture: Occam’s Razor, VC Dimension, and Error Bounds**



Kansas State University  
Department of Computing and Information Sciences

CIS 732: Machine Learning and Pattern Recognition