Department of Computing

UniS University of Surrey Guildford

CS364 Artificial Intelligence Machine Learning

Matthew Casey

m.casey@surrey.ac.uk

UniS University of Surrey Department of Computing

Learning Outcomes

UniS
University of Surrey
Department of Computing

- Describe methods for acquiring human knowledge
  - Through experience
- Evaluate which of the acquisition methods would be most appropriate in a given situation
  - Limited data available through example

Learning Outcomes

UniS
University of Surrey
Department of Computing

- Describe techniques for representing acquired knowledge in a way that facilitates automated reasoning over the knowledge
  - Generalise experience to novel situations
- Categorise and evaluate AI techniques according to different criteria such as applicability and ease of use, and intelligently participate in the selection of the appropriate techniques and tools, to solve simple problems
  - Strategies to overcome the 'knowledge engineering bottleneck'

Key Concepts

UniS
University of Surrey
Department of Computing

- Machines learning from experience…
  - Through examples, analogy or discovery
- Adapting…
  - Changes in response to interaction
- Generalising…
  - To use experience to form a response to novel situations

What is Learning?

UniS
University of Surrey
Department of Computing

- ‘The action of receiving instruction or acquiring knowledge’
- ‘A process which leads to the modification of behaviour or the acquisition of new abilities or responses, and which is additional to natural development by growth or maturation’

Oxford English Dictionary (1989). Learning, vbl. n. 2nd Edition.
http://dictionary.oed.com/cgi/entry/50131042?single=1&query_type=word&queryword=learning&first=1&max_to_show=10. [Accessed 16-10-06].

Machine Learning

UniS
University of Surrey
Department of Computing

- Negnevitsky:
  - ‘In general, machine learning involves adaptive mechanisms that enable computers to learn from experience, learn by example and learn by analogy’ (2005:165)
- Callan:
  - ‘A machine or software tool would not be viewed as intelligent if it could not adapt to changes in its environment’ (2003:225)
- Luger:
  - ‘Intelligent agents must be able to change through the course of their interactions with the world’ (2002:351)

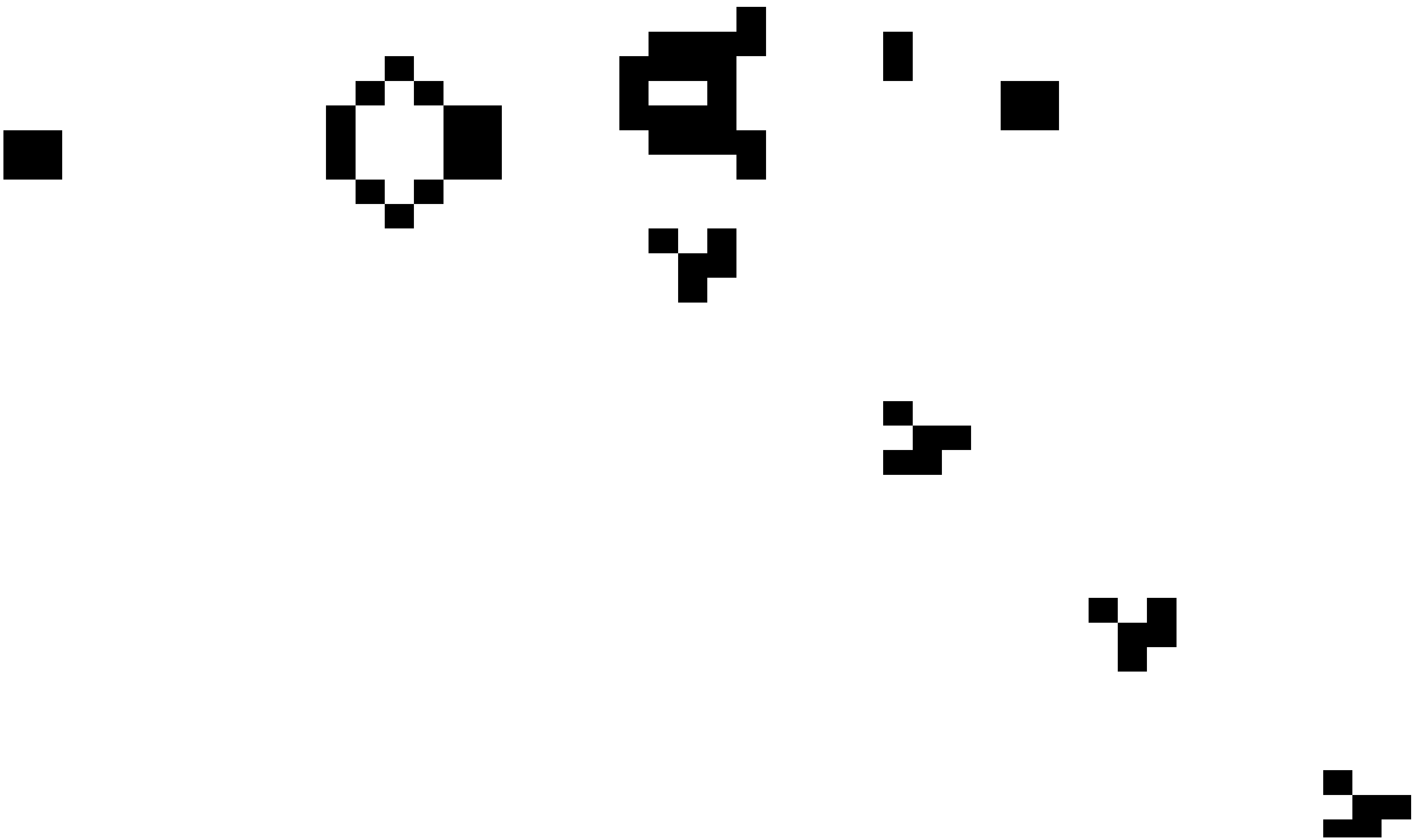
Types of Learning

UniS
University of Surrey
Department of Computing

- Inductive learning
  - Learning from examples
  - Supervised learning: training examples with a known classification from a teacher (ANN, Decision Tree Learning)
  - Unsupervised learning: no pre-classification of training examples (clustering, HMM, dimension reduction: PCA, rough set)
- Evolutionary/genetic learning
  - Shaping a population of individual solutions through survival of the fittest
  - Emergent/sudden behaviour/interaction: game of life

Game of Life

UniS
University of Surrey
Department of Computing



Wikipedia (2006). Image:Gospers glider gun.gif - Wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/Image:Gospers_glider_gun.gif. [Accessed 16-10-06].

Why need learning?

UniS
University of Surrey
Department of Computing

- Knowledge Engineering Bottleneck
  - ‘Cost and difficulty of building expert systems using traditional […] techniques’ (Luger 2002:351)
- Complexity of task / amount of data
  - Other techniques fail or are computationally expensive
- Problems that cannot be defined
  - Discovery of patterns / data mining

Example: Ice-cream

UniS
University of Surrey
Department of Computing

- When should an ice-cream seller attempt to sell ice-cream (Callan 2003:241)?
  - Could you write a set of rules?
  - How would you acquire the knowledge?
- You might learn by experience:
  - For example, experience of:
  - ‘Outlook’: Overcast or Sunny
  - ‘Temperature’: Hot, Mild or Cold
  - ‘Holiday Season’: Yes or No

Example: Ice-cream (contd...)

Randomly Ordered Data

Outlook	Temperature	Holiday Season	Result
Overcast	Mild	Yes	Don't Sell
Sunny	Mild	Yes	Sell
Sunny	Hot	No	Sell
Overcast	Hot	No	Don't Sell
Sunny	Cold	No	Don't Sell
Overcast	Cold	Yes	Don't Sell

Generalisation

UniS
University of Surrey
Department of Computing

- What should the seller do when:
  - ‘Outlook’: Sunny
  - ‘Temperature’: Hot
  - ‘Holiday Season’: Yes

Sell

- What about:
  - ‘Outlook’: Overcast
  - ‘Temperature’: Hot
  - ‘Holiday Season’: Yes

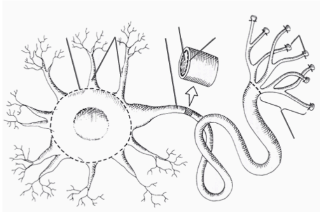
Sell

Can A Machine Learn?

- From a limited set of examples, you should be able to generalise
  - How did you do this?
  - How can we get a machine to do this?
- Machine learning is the branch of Artificial Intelligence concerned with building systems that generalise from examples

Common Techniques

UniS
University of Surrey
Department of Computing



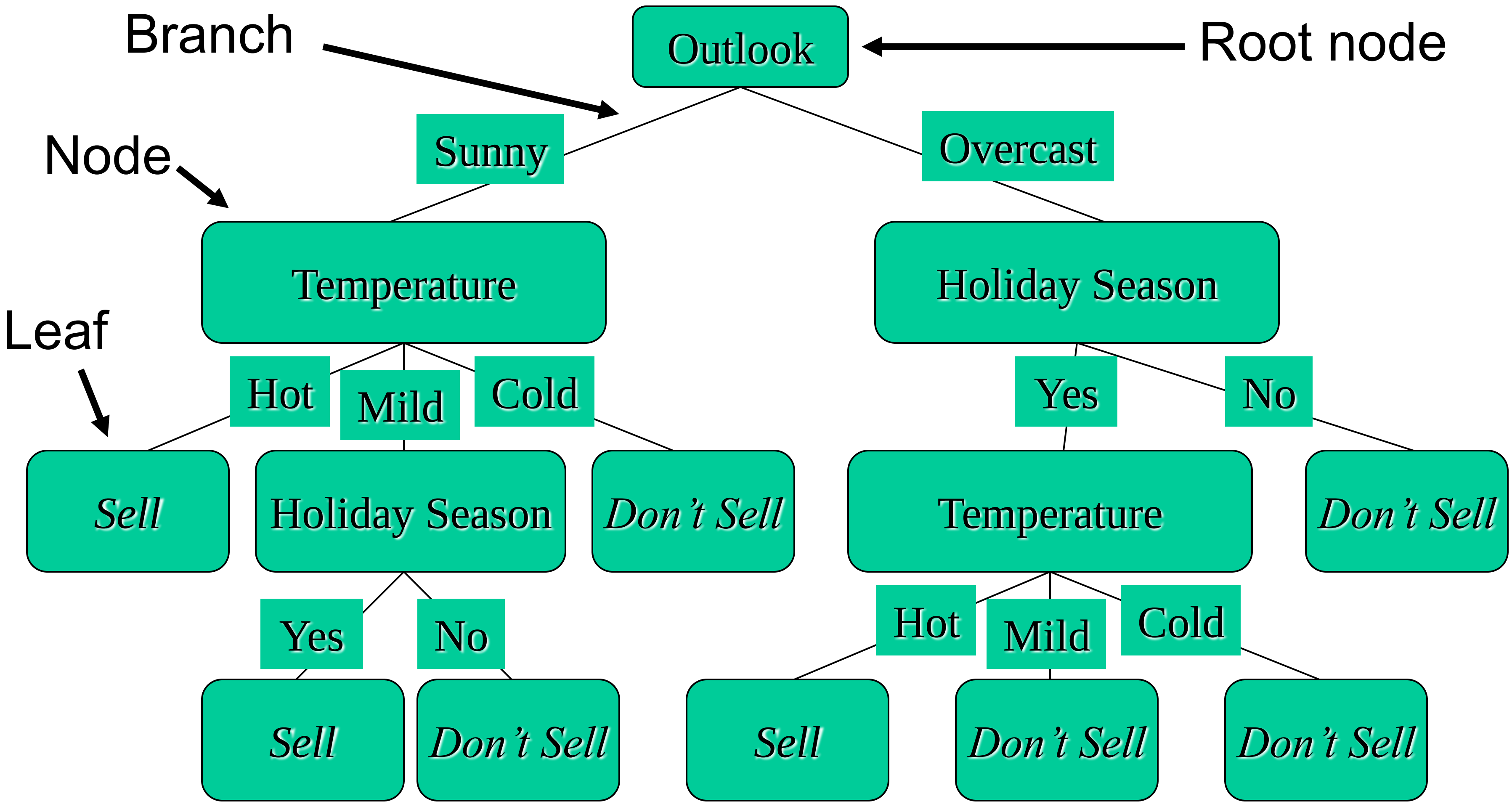
- Decision trees
- Neural networks
  - Developed from models of the biology of behaviour: parallel processing in neurons
  - Human brain contains of the order of 10^{10} neurons, each connecting to 10^4 others
- Genetic algorithms
  - Evolving solutions by 'breeding'
  - Generations assessed by fitness function

Decision Trees

- A map of the reasoning process, good at solving classification problems (Negnevitsky, 2005)
- A decision tree represents a number of different attributes and values
  - Nodes represent attributes
  - Branches represent values of the attributes
- Path through a tree represents a decision
- Tree can be associated with rules

Example 1

UniS
University of Surrey
Department of Computing



Construction

UniS
University of Surrey
Department of Computing

- Concept learning:
  - Inducing concepts from examples
- Different algorithms used to construct a tree based upon the examples
  - Most popular ID3 (Quinlan, 1986)
- But:
  - Different trees can be constructed from the same set of examples
  - Real-life is noisy and often contradictory

Ambiguous Trees

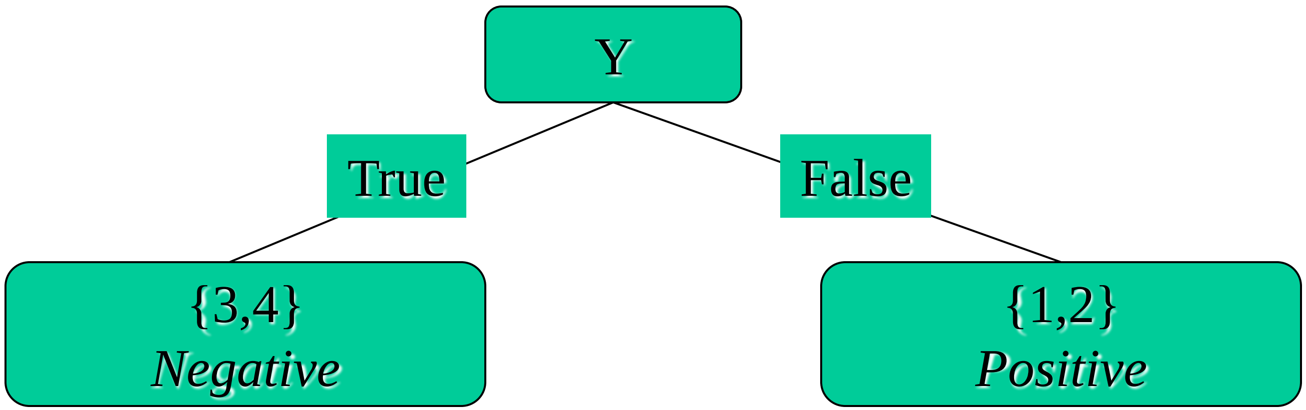
UniS
University of Surrey
Department of Computing

Consider the following data:

Item	X	Y	Class
1	False	False	+
2	True	False	+
3	False	True	-
4	True	True	-

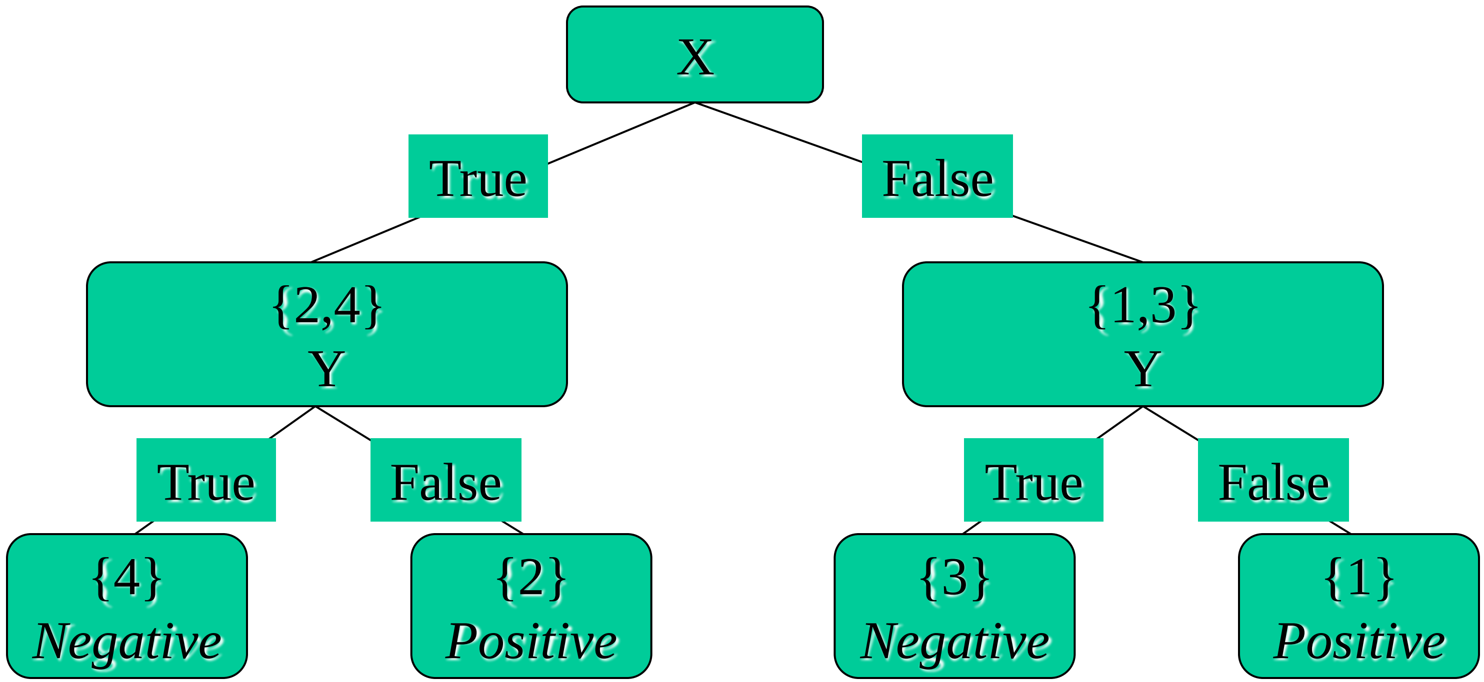
Ambiguous Trees

UniS
University of Surrey
Department of Computing



Ambiguous Trees

UniS
University of Surrey
Department of Computing



Which tree is the best?

- Based upon choice of attributes at each node in the tree
- A split in the tree (branches) should correspond to the predictor with the maximum separating power

Information Theory

UniS
University of Surrey
Department of Computing

- We can use Information Theory to help us understand:
  - Which attribute is the best to choose for a particular node of the tree
  - This is the node that is the best at separating the required predictions, and hence which leads to the best (or at least a good) tree
- 'Information Theory address both the *limitations* and the *possibilities* of communication' (MacKay, 2003:16)
  - Measuring information content
  - Probability and entropy: measure of disorder

MacKay, D.J.C. (2003). Information Theory, Inference, and Learning Algorithms. Cambridge, UK: Cambridge University Press.

Choosing Attributes

UniS
University of Surrey
Department of Computing

- Entropy:
  - Measure of disorder/unexpected (high is bad)
- For *c* classification categories
- Attribute *a* that has value *v*
- Probability of *v* being in category *i* is p_i
- Entropy *E* is:

$$E(a=v)=\sum_{i=1}^{c}-p_i\log_2 p_i$$

Example

University of Surrey
Department of Computing

- Callan (2003:242-247)
  - Locating a new bar

Entropy Example

- Choice of attributes:
  - City/Town, University, Housing Estate, Industrial Estate, Transport and Schools
- City/Town: is either Y or N
- For Y: 7 *positive* examples, 3 *negative*
- For N: 4 *positive* examples, 6 *negative*

Entropy Example

UniS
University of Surrey
Department of Computing

- City/Town as root node:
  - For $c=2$ (*positive* and *negative*) classification categories
  - Attribute a=City/Town that has value v=Y
  - Probability of v=Y being in category *positive*
    - $p_{\text{i=positive}} = 7/10$
  - Probability of v=Y being in category *negative*
    - $p_{\text{i=negative}} = 3/10$

Entropy Example

UniS
University of Surrey
Department of Computing

- City/Town as root node:
  - For *c=2* (*positive* and *negative*) classification categories
  - Attribute *a*=City/Town that has value *v*=Y
  - Entropy *E* is:

$$E(\text{City/Town} = \text{Y}) = -7/10 \text{ x } \log_2 7/10 - 3/10 \text{ x } \log_2 3/10$$
$$= 0.881$$

Entropy Example

UniS
University of Surrey
Department of Computing

- City/Town as root node:
  - For $c=2$ (*positive* and *negative*) classification categories
  - Attribute a=City/Town that has value v=N
  - Probability of v=N being in category *positive*
    - $p_{\text{i=positive}} = 4/10$
  - Probability of v=N being in category *negative*
    - $p_{\text{i=negative}} = 6/10$

Entropy Example

University of Surrey
Department of Computing

- City/Town as root node:
  - For *c=2* (*positive* and *negative*) classification categories
  - Attribute *a*=City/Town that has value *v*=N
  - Entropy *E* is:

$$E(\text{City/Town} = \text{N}) = -4/10 \text{ x } \log_2 4/10 - 6/10 \text{ x } \log_2 6/10$$
$$= 0.971$$

Choosing Attributes

University of Surrey
Department of Computing

- Information gain:
  - Expected reduction in entropy (high is good)
- Entropy of whole example set T is $E(T)$
- Examples with $a=v$, v is j^{th} value are T_j,
- Entropy $E(a=v_j)=E(T_j)$
- Gain is:

$$Gain(T, a) = E(T) - \sum_{j=1}^{V} \frac{|T_j|}{|T|} E(T_j)$$

- T = total samples = 20
- T_j = number of samples with value j (Y/N values)

Information Gain Example

UniS
University of Surrey
Department of Computing

- For root of tree there are 20 examples:
  - For *c=2* (*positive* and *negative*) classification categories
  - Probability of being *positive* with 11 examples
    - $p_{\text{i=positive}} = 11/20$
  - Probability of being *negative* with 9 examples
    - $p_{\text{i=negative}} = 9/20$

Information Gain Example

UniS
University of Surrey
Department of Computing

- For root of tree there are 20 examples:
  - For *c=2* (*positive* and *negative*) classification categories
  - Entropy of all training examples *E(T)* is:

$$|T| = 20$$

$$E(\mathrm{T}) = -11/20 \times \log_2 11/20 - 9/20 \times \log_2 9/20$$

$$= 0.993$$

Information Gain Example

UniS
University of Surrey
Department of Computing

- City/Town as root node:
  - 10 examples for *a*=City/Town and value *v*=Y
    - $|T_{j=Y}| = 10$ $\mathrm{E}(T_{j=Y}) = 0.881$
  - 10 examples for *a*=City/Town and value *v*=N
    - $|T_{j=N}| = 10$ $E(T_{j=N}) = 0.971$

$$Gain(T, City/Town) = 0.993 - ((10/20 \text{ x } 0.881) + (10/20 \text{ x } 0.971))$$
$$= 0.067$$

Example

UniS
University of Surrey
Department of Computing

- Calculate the information gain for the Transport attribute

Information Gain Example

UniS
University of Surrey
Department of Computing

	All	City/Town		University		Housing-Estate			
		Y	N	Y	N	L	M	S	N
Pr(Class=positive)	0.550	0.700	0.400	0.600	0.533	0.600	0.500	0.000	0.833
Pr(Class=negative)	0.450	0.300	0.600	0.400	0.467	0.400	0.500	1.000	0.167
Entropy	0.993	0.881	0.971	0.971	0.997	0.971	1.000	0.000	0.650
Information Gain Gain(T,a)		0.067		0.002		0.255			

Industrial-Estate		Transport			Schools		
Y	N	A	P	G	L	M	S
1.000	0.438	0.429	0.375	1.000	0.714	0.500	0.429
0.000	0.563	0.571	0.625	0.000	0.286	0.500	0.571
0.000	0.989	0.985	0.954	0.000	0.863	1.000	0.985
0.202		0.266			0.046		

Choosing Attributes

UniS
University of Surrey
Department of Computing

- Chose root node as the attribute that gives the highest Information Gain
  – In this case attribute Transport with 0.266
- Branches from root node then become the values associated with the attribute
  – Recursive calculation of attributes/nodes
  – Filter examples by attribute value

Recursive Example

UniS
University of Surrey
Department of Computing

- With Transport as the root node:
  - Select examples where Transport is Average
  - (1, 3, 6, 8, 11, 15, 17)
  - Use only these examples to construct this branch of the tree
  - Repeat for each attribute (Poor, Good)

Choosing child node

UniS
University of Surrey
Department of Computing

Choice of attributes where Transport is Average {1, 3, 6, 8, 11, 15, 17}:

– City/Town, University, Housing Estate, Industrial Estate, and Schools (w/o Transport)
– City/Town: is either Y or N
– For Y: 2 pos examples {1,3}, 2 neg {8,15}
– For N: 1 pos {11}, 2 neg {6,17}
– Attribute *a*=City/Town that has value *v*=Y
– Prob of *v*=Y being *positive* $p_{i=pos} = 2/4$
– Prob of *v*=Y being *negative* $p_{i=neg} = 2/4$

Choosing child node

UniS
University of Surrey
Department of Computing

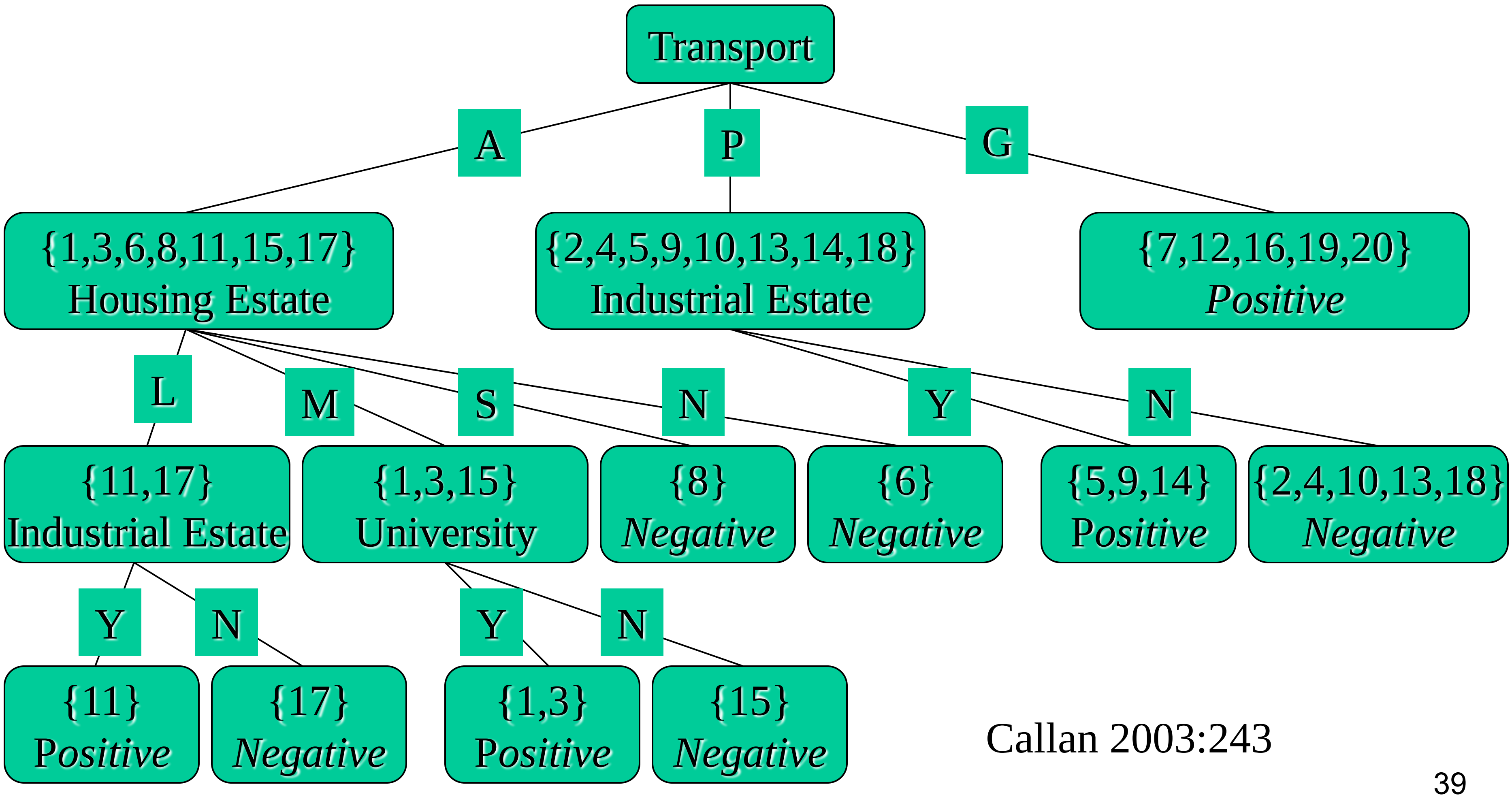
- $E(T_1 = City/Town = \mathrm{Y}) = -2/4 \log_2 2/4 - 2/4 \log_2 2/4 = 1$
- Attribute $a = City/Town$ that has value $v = \mathrm{N}$
- Prob of $v = \mathrm{N}$ being *positive* $p_{i=pos} = 1/3$
- Prob of $v = \mathrm{N}$ being *negative* $p_{i=neg} = 2/3$
- $\mathrm{E}(T_2 = City/Town = \mathrm{N}) = -1/3 \log_2 1/3 - 2/3 \log_2 2/3 = 0.918$

For all 7 examples {1, 3, 6, 8, 11, 15, 17}

- Prob of being pos $p_{i=pos} = 3/7$
- Prob of being pos $p_{i=neg} = 4/7$
- $E(T) = -3/7 \log_2 3/7 - 4/7 \log_2 4/7 = -0.43 \log 0.43/\log 2 - 0.57 \log 0.57/\log 2 = 0.985$
- $\mathrm{Gain}(T, City/Town) = E(T) - T_1/T * E(T_1) - T_2/T * E(T_2)$
  - » $= 0.985 - 4/7 * 1 - 3/7 * 0.918 = 0.020$

Final Tree

UniS
University of Surrey
Department of Computing



Callan 2003:243

UniS
University of Surrey
Department of Computing

ID3

- Procedure Extend(Tree *d*, Examples T)
  - Choose best attribute *a* for root of *d*
    - Calculate *E(a=v)* and *Gain(T,a)* for each attribute
    - Attribute with highest *Gain(T,a)* is selected as best
  - Assign best attribute *a* to root of *d*
  - For each value *v* of attribute *a*
    - Create branch for *v=a* resulting in sub-tree d_j
    - Assign to T_j training examples from *T* where *v=a*
    - Recurse sub-tree with Extend(d_j, T_j)

Data Issues

UniS
University of Surrey
Department of Computing

- Use prior knowledge where available
- Understand the data
  - Examples may be noisy
  - Examples may contain irrelevant attributes
  - For missing data items, substitute appropriate values or remove examples
  - Check the distribution of attributes across all examples and normalise where appropriate
- Where possible, split the data
  - Use a training, validation and test data set
  - Helps to construct an appropriate system and test generalisation
  - Validation data can be used to limit tree construction/prune the tree to achieve a desired level of performance

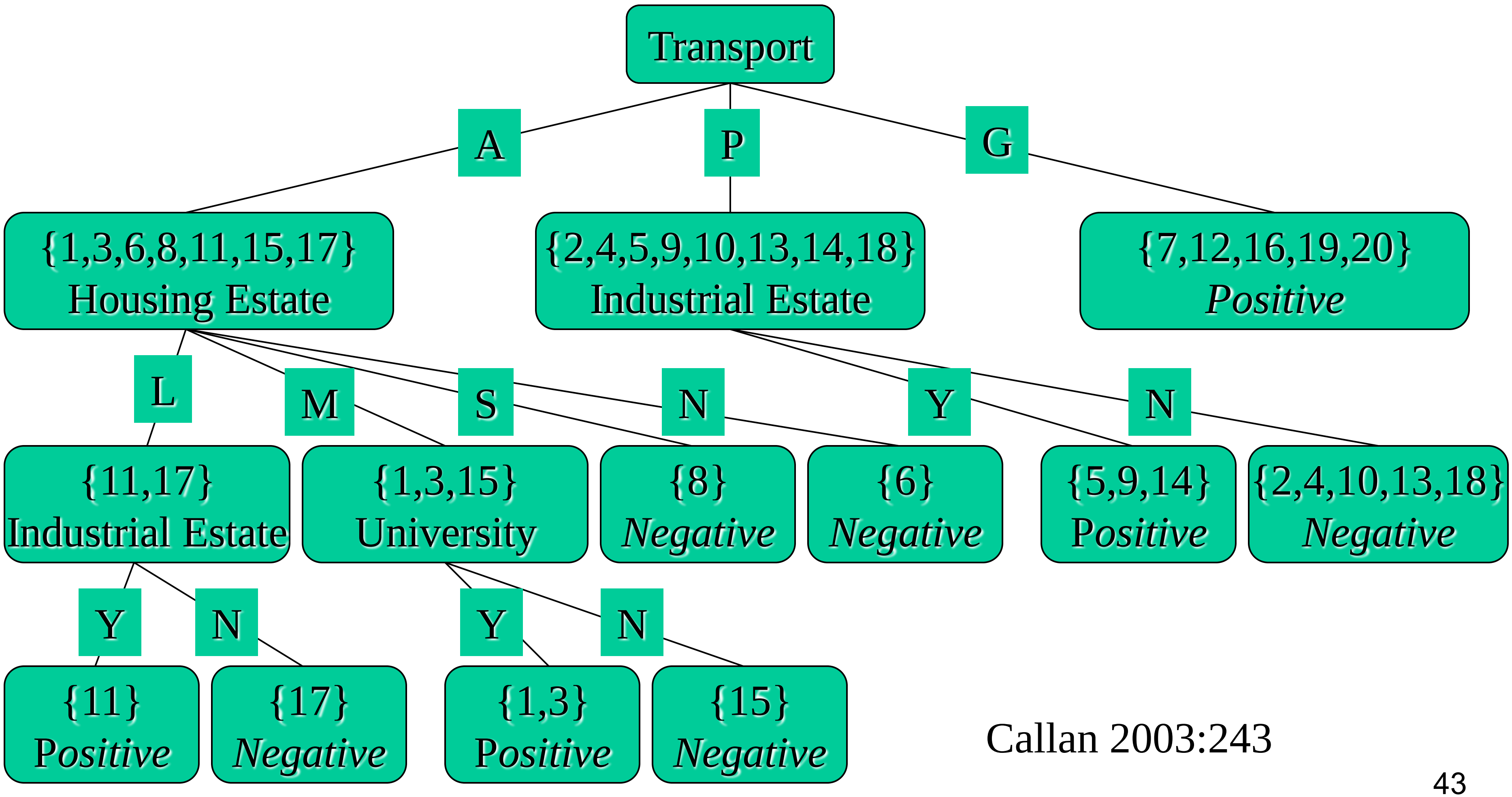
Extracting Rules

UniS
University of Surrey
Department of Computing

- We can extract rules from decision trees
  - Create one rule for each root-to-leaf path
  - Simplify by combining rules
- Other techniques are not so transparent:
  - Neural networks are often described as ‘black boxes’ – it is difficult to understand what the network is doing
  - Extraction of rules from trees can help us to understand the decision process

Rules Example

UniS
University of Surrey
Department of Computing



Callan 2003:243

Rules Example

UniS
University of Surrey
Department of Computing

- IF Transport is Average
  AND Housing Estate is Large
  AND Industrial Estate is Yes
  THEN *Positive*
- …
- IF Transport is Good
  THEN *Positive*

Summary

UniS
University of Surrey
Department of Computing

- What are the benefits/drawbacks of machine learning?
  - Are the techniques simple?
  - Are they simple to implement?
  - Are they computationally cheap?
  - Do they learn from experience?
  - Do they generalise well?
  - Can we understand how knowledge is represented?
  - Do they provide perfect solutions?

Key Concepts

UniS
University of Surrey
Department of Computing

- Machines learning from experience…
  - Through examples, analogy or discovery
  - But real life is imprecise – how do you know which data is valid and collect (enough of) it?
- Adapting…
  - Changes in response to interaction
  - But you only want to learn what's 'correct' – how do you know this (you don't know the solution)?
- Generalising…
  - To use experience to form a response to novel situations
  - How do you know the solution is accurate?

Source Texts

University of Surrey
Department of Computing

- Negnevitsky, M. (2005). *Artificial Intelligence: A Guide to Intelligent Systems*. 2nd Edition. Essex, UK: Pearson Education Limited.
  - Chapter 6, pp. 165-168, chapter 9, pp. 349-360.
- Callan, R. (2003). *Artificial Intelligence*, Basingstoke, UK: Palgrave MacMillan.
  - Part 5, chapters 11-17, pp. 225-346.
- Luger, G.F. (2002). *Artificial Intelligence: Structures & Strategies for Complex Problem Solving.* 4th Edition. London, UK: Addison-Wesley.
  - Part IV, chapters 9-11, pp. 349-506.

Journals

University of Surrey
Department of Computing

- Artificial Intelligence
  - http://www.elsevier.com/locate/issn/00043702
  - http://www.sciencedirect.com/science/journal/00043702

Articles

University of Surrey
Department of Computing

- Quinlan, J.R. (1986). Induction of Decision Trees. *Machine Learning*, vol. 1, pp.81-106.
- Quinlan, J.R. (1993). *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann Publishers.

Websites

UniS
University of Surrey
Department of Computing

- UCI Machine Learning Repository
  - Example data sets for benchmarking
  - http://www.ics.uci.edu/~mlearn/MLRepository.html
- Wonders of Math: Game of Life
  - Game of life applet and details
  - http://www.math.com/students/wonders/life/life.html