

Machine learning

- Automatically
 - finding knowledge in data - data mining
 - building models of processes/programs from data
 - adapting/customizing programs from user interaction
- Russell and Norvig, Chapters 18, 19

Mid-Semester Exam

- Average Mark = 14.0
- Best Mark = 19
- Best Answered Questions (>90%) = 5,7,14
 - Definition of completeness
 - Breadth First Search
 - MiniMax
- Worst Answered Questions (<50%) = 2,13,15,18
 - Negative attribute of Turing Test
 - Definition of effective branching factor
 - Alpha-beta pruning
 - Rational agent uses Decision Theory

Data mining

Patient103 time = 1 → Patient103 time = 2 → ... → Patient103 time = n

Age: 23	Age: 23	Age: 23
FirstPregnacy: no	FirstPregnacy: no	FirstPregnacy: no
Anemia: no	Anemia: no	Anemia: no
Diabetes: no	Diabetes: YES	Diabetes: no
PreviousPrematureBirth: no	PreviousPrematureBirth: no	PreviousPrematureBirth: no
Ultrasound: ?	Ultrasound: abnormal	Ultrasound: ?
Elective C-Section: ?	Elective C-Section: no	Elective C-Section: no
Emergency C-Section: ?	Emergency C-Section: ?	Emergency C-Section: Yes
...

Learned Rule:

If No previous vaginal delivery and Abnormal 2nd Trimester Ultrasound, and Malpresentation at admissions, and
Then Probability of Emrgency C-Section is 0.6

Training set accuracy: 26/41 - .63
Test set accuracy: 12/20 - .60

9714 pregnant women
215 attributes over time
(health, ultrasound, type of delivery, final health of mother and baby)
Predict features occurring late based on those occurring earlier, e.g. predict emergency C-section.

0.07 if rule is not used...

"Training" computer programs

30 x 32 Network Inputs

Learned Weights

624 grey-scale images of 20 different people were used to train a "neural network." Additional images were recognized with 90% accuracy. Face direction classified with 90% accuracy. Where's the program and who wrote it?

Typical Input Images

Self-customizing computer programs

True Rating	Predicted Rating					Skip	Total
	1	2	3	4	5		
1:	0	1	0	0	0	1	2
2:	1	15	6	4	0	15	41
3:	0	6	31	20	0	15	72
4:	0	6	8	42	0	20	76
5:	0	0	0	4	0	1	5
skip:	0	8	4	5	1	141	159

Automatic filtering of newsgroup articles.
Trained using user-expressed rankings (1-5).
Assembles top-20 list of articles for the user.
Based on a statistical analysis of text.

Machine learning: Concepts Overview: aims

- understand basic concepts used in machine learning, e.g. example, hypothesis, classification, regression, function, training and testing performance
- be aware of different types of feedback received by a learning agent, i.e. supervised and unsupervised
- understand what generalisation means and what specialisation means
- know of several machine learning techniques and representations
- know how to assess the performance of machine learning techniques
- know of several ways to improve machine learning techniques

Machine learning: Concepts

Overview: topics

- Learning (agent, element)
- Learning element (components, feedback, representation)
- Inductive learning, hypothesis space
- Information (theory, content, gain)
- Assessing performance of supervised learning algorithms
- Improving machine learning performance (overfitting, ensemble learning)
- Machine learning techniques (examples)

Nature of Task

- Classification:
 - output is discrete (nominal) – classify the input (percepts + internal state)
- Regression:
 - output is continuous – real-valued response

Classification

- Using a representation (function or process) and learning algorithm that *discriminates* between classes of inputs.
- Either discrete representations (e.g. logical expressions), or
- numeric representations that describe decision boundaries between classes in the input space.

Machine learning goals

- Building a model
- Data mining
- Saves us the hassle of building intelligent machines
- Allows the machine to acquire competence as it experiences its environment

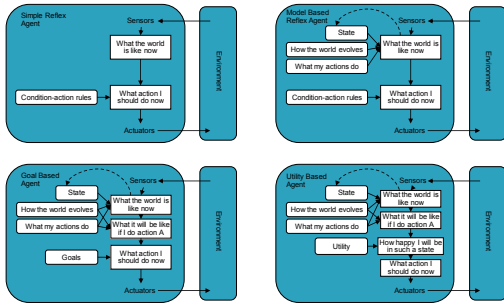
Applications

- Immense range – wherever you need adaptable software
 - Classification examples:
 - face recognition, speech recognition, patient diagnosis, email spam or not?, credit scoring
 - Regression examples:
 - predict \$A exchange rate, predict tomorrow's maximum temperature, estimate time to failure, estimate current location

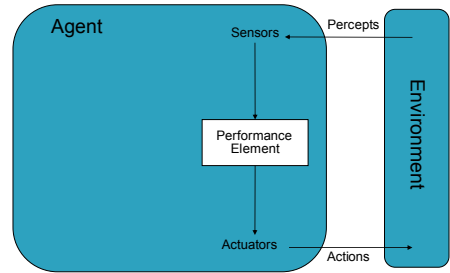
What is learning?

- Any change in a system that allows it to perform better the second time on repetition on the same or on another task drawn from the same population
(Simon, 1983).
- Compare with human learning, natural evolution, etc.

Types of Agents



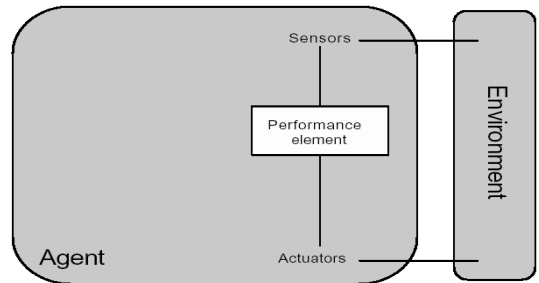
A Basic Agent



The learning agent

- Learning allows agents to operate in unknown environments
- Learning agents can become more competent than their initial knowledge base allows
- Percepts can be used for deciding which actions to perform, and also for improving the ability to act in the future

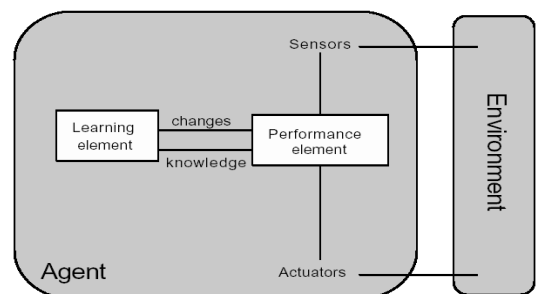
The learning agent



The learning agent: Performance element

- Responsible for selecting external actions
- All of the components of previously considered agent types (simple reflex, model based reflex, goal based, utility) are part of the performance element

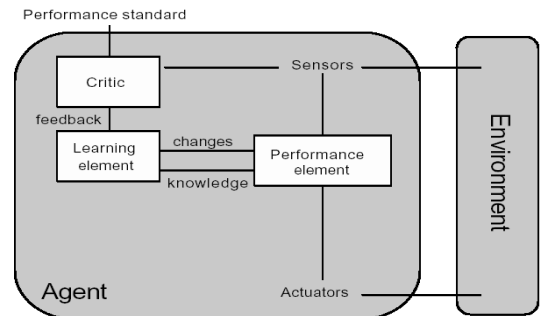
The learning agent



The learning agent: Learning element

- Responsible for making improvements
- Modifies the performance element so that it makes better decisions
- Makes changes to the 'knowledge' components of the previously considered agent types
 - How the world evolves
 - What my actions do

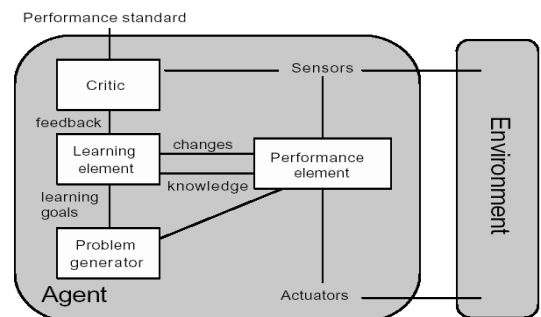
The learning agent



The learning agent: Critic

- Provides feedback to the agent based on a fixed performance standard
- Determines how the performance element should be modified

The learning agent



The learning agent: Problem generator

- Suggests actions that will lead to new, informative experiences
- Exploratory actions
- Enables the agent to potentially discover better actions

The learning agent: Learning Element

How can the agent learn?

- Components
 - Of the performance element that can be learned
- Feedback
 - About the components to aid learning
- Representation
 - Used for the components

Learning Element: Performance Element Components

What can the agent learn?

- Conditions -> Actions
- Percepts -> Inferred World Properties
- How the world evolves
- Actions -> World Changes
- Utility of world states
- Action-values
- Goal states

Learning Element: Type of Feedback

What feedback can the agent receive?

- None -> **unsupervised** learning – e.g. find “clusters” in data
- Good/bad feedback -> **reinforcement** learning – aim for good results
- Right answers -> **supervised** learning – system should produce similar answers on examples and generalise to new cases
 - Very common – we focus on this

Learning Element: Feedback – Unsupervised

- The learning element aims to learn patterns in the input when no feedback is supplied
- E.g. taxi agent gradually develops a concept of ‘good traffic days’ and ‘bad traffic days’



Photo from: commons.wikimedia.org

Learning Element: Feedback – Reinforcement

- The learning agent learns from reward / punishment about their behaviour
- E.g. playing a game, after 100 moves being told ‘you lose’



Photo from: commons.wikimedia.org

Learning Element: Feedback – Supervised

- A teacher provides the correct output for a set of examples, the learning agent learns to associate the input pattern with these outputs
- E.g. seeing many camera images, being told they contain buses, learning to recognise buses



Photo from: commons.wikimedia.org

Supervised learning Concepts, notation

- Input space X with examples $x_i \in X, i=1, \dots, m$.
- Assume that the true (or best) function mapping input to output is $f(\cdot)$
- Output space Y with corresponding examples $y_i = f(x_i) \in Y, i=1, \dots, m$.
- We hope to approximate $f(\cdot)$ via a Hypothesis function $h(\cdot)$
- Examples: input/output pairs: $d_i = \{x_i, y_i\}$,
- Whole dataset $D = \{d_i, i=1, \dots, m\}$

Learning Element: Representation

How can the knowledge be represented?

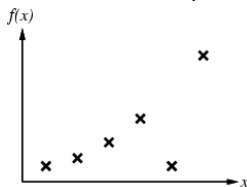
- Propositional logic (Chapter 18)
 - (Decision Trees)
- First-order logic (Chapter 19)
 - (Current Best Learning)
- Bayesian networks (Chapter 20)
 - (Naïve Bayes' Classifier)
- Neural networks (Chapter 20)
 - (Feed-forward networks)

Learning Element: Availability of prior knowledge

- Majority of learning in AI involves agents starting with no knowledge
- Most human learning has context of background knowledge
- Prior knowledge can help in learning

Inductive learning (1)

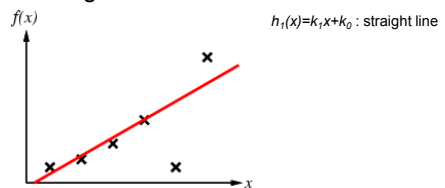
- Given a collection of examples of f , return a function h that approximates f
- h is a hypothesis, and is consistent if it agrees with f on all examples



Figures from: <http://aima.eecs.berkeley.edu/slides-ppt/>

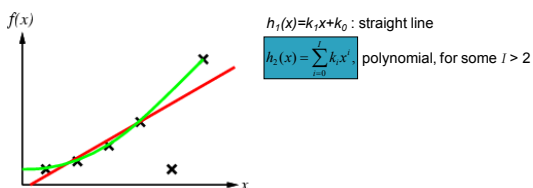
Inductive learning (2)

- The problem of induction is to find a good hypothesis that will generalise well
- A straight line?



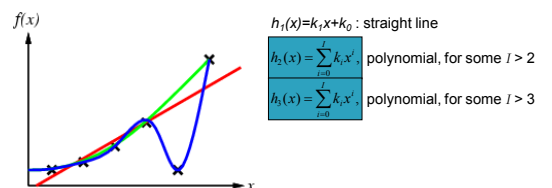
Inductive learning (3)

- Curve?



Inductive learning (4)

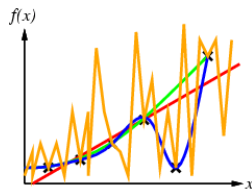
- Polynomial?



Inductive learning (5)

- More complex ...

Ockham's razor: prefer the simplest hypothesis consistent with data



Complexity?
Error?
Generalisation?

Tradeoff between the complexity of the hypothesis and the degree of fit to the data

Hypothesis space

- **H** is the set of hypotheses considered
- A learning problem is **realisable** if **H** contains the true function
- A learning problem is **unrealisable** if **H** does not contain the true function

Searching the hypothesis space (1)

- Alternative to searching hypothesis space to find a hypothesis that matches the current knowledge is:
- Taking advantage of prior knowledge of the world, and allowing the incremental construction of hypotheses

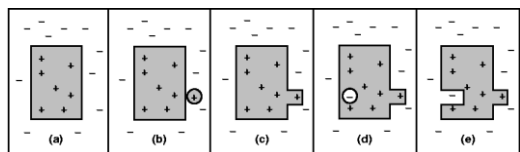
Searching the hypothesis space (2)

- Assume representation is predicate-based
- A hypothesis defines a goal predicate
 - which is true for all "positives" and false for all "negatives"
- The hypothesis predicts a set of examples as "positives" – the extension of the hypothesis

Searching the hypothesis space (3)

- A hypothesis that does not agree with observed examples may be:
 - False positives (hypothesis says the example should be positive, but it is negative)
 - False negatives (hypothesis says the example should be negative, but it is positive)

Searching the hypothesis space (4)



Training examples: H_f is illustrated as the boundary between its positives (extension) and its negatives. (a) consistent, (b) one false negative, (c) H_f is generalized, (d) one false positive, (e) H_f is specialized.

Searching the hypothesis space (5)

- Generalisation
 - The new hypothesis is more general than the existing hypothesis
 - More examples are classified as positive
 - The extension of the hypothesis is increased
- Specialisation
 - The new hypothesis is more specific than the existing hypothesis
 - Fewer examples are classified as positive
 - The extension of the hypothesis is decreased

Tossing a coin...



Information theory (1)

- Imagine we want to send a signal along a telegraph line via a binary encoding
- Call the two possible symbols A and B
- For the type of information we wish to send, there will be a certain probabilities $P(A)$ and $P(B) = 1 - P(A)$
- Claude Shannon worked out that the number of bits of information one gets from each received symbol (either an A or a B) depends upon $P(A)$ and $P(B)$

Information theory (2)

- $I[P(A), P(B)] = -P(A)\log_2 P(A) - P(B)\log_2 P(B)$
- The information content per received symbol is maximised when $P(A) = P(B)$
- Basis of many coding schemes, e.g. zip
- The theory was easily generalised to an arbitrary number of symbols: m

$$I(P(v_1), \dots, P(v_m)) = \sum_{i=1}^m -P(v_i) \log_2 P(v_i)$$

Information theory (3)

Information content of answer in bits, given probabilities of all possible answers v_i

$$I(P(v_1), \dots, P(v_m)) = \sum_{i=1}^m -P(v_i) \log_2 P(v_i)$$

Estimate of information content of a positive answer in a binary classification:

$$I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) = -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n}$$

where p is the number of positive examples and n is the number of negative examples.

$$I\left(\frac{1}{2}, \frac{1}{2}\right) = -0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1 \text{ bit}$$

Information theory (4)

$$I(P(v_1), \dots, P(v_m)) = \sum_{i=1}^m -P(v_i) \log_2 P(v_i)$$

$$I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) = -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n}$$

Convention: $0 \log 0 = 0$. Why? L'Hôpital's rule on $x \log x$:

$$\lim_{x \rightarrow 0} \log x / (1/x) = \lim_{x \rightarrow 0} (-x) = 0$$

Also remember: $\log 1 = 0$; $\log 0$ generally undefined (approaches negative infinity).

1 0 1 0 0 1 0 1 1 0 0 1 1 $-0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1 \text{ bit}$

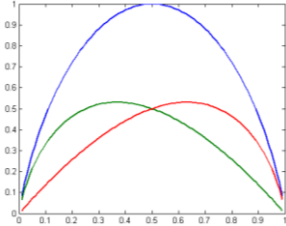
1 1 1 0 1 1 0 1 1 1 0 1 1 $-\frac{9}{12} \log_2 \frac{9}{12} - \frac{3}{12} \log_2 \frac{3}{12} = 0.81 \text{ bits}$

0 0 1 0 0 0 0 1 0 0 0 0 0 $-\frac{2}{12} \log_2 \frac{2}{12} - \frac{10}{12} \log_2 \frac{10}{12} = 0.65 \text{ bits}$

Entropy

$$\sum_{i=1}^m -p_i \log_2 p_i$$

Measure of the information contained in a message, as opposed to the portion of the message that is strictly determined (hence predictable) by inherent structures.
(from Wikipedia, www.wikipedia.org)

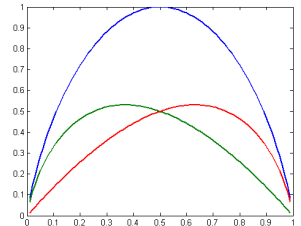


Information theory:

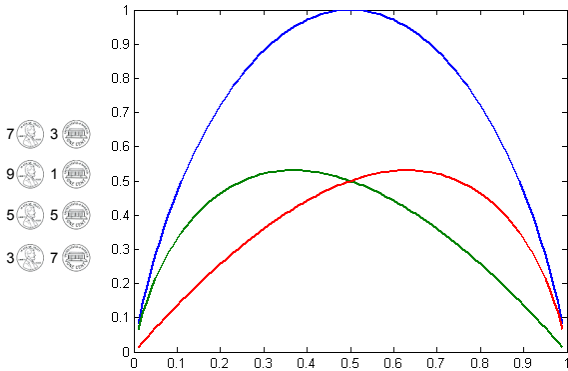
Example

$$I(P(v_1), \dots, P(v_m)) = \sum_{i=1}^m -P(v_i) \log_2 P(v_i)$$

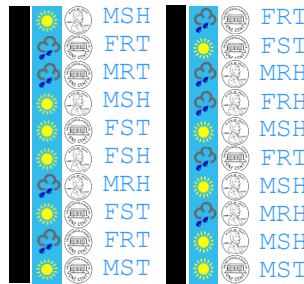
$$I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) = -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n}$$



$$I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) = -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n}$$



Tossing a coin...



Can we use Gender or Weather to help us predict the result?

Symbols from: commons.wikimedia.org

Information gain for choosing an attribute

$$Gain(A) = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) - Remainder(A)$$

$$Remainder(A) = \sum_{i=1}^v \frac{p_i + n_i}{p+n} I\left(\frac{p_i}{p_i + n_i}, \frac{n_i}{p_i + n_i}\right)$$

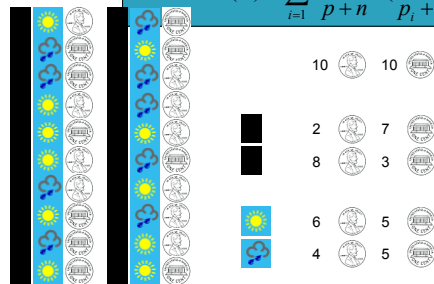
- Information gain is the difference between the original information requirement and the new requirement

Information gain for choosing an attribute:

Example (1)

$$Gain(A) = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) - Remainder(A)$$

$$Remainder(A) = \sum_{i=1}^v \frac{p_i + n_i}{p+n} I\left(\frac{p_i}{p_i + n_i}, \frac{n_i}{p_i + n_i}\right)$$



Information gain for choosing an attribute:
Example

$$\begin{aligned} \text{Remainder}(\text{Gender}) &= \frac{2+7}{10+10} I\left(\frac{2}{2+7}, \frac{7}{2+7}\right) + \frac{8+3}{10+10} I\left(\frac{8}{8+3}, \frac{3}{8+3}\right) \\ &= \frac{9}{20} I\left(\frac{2}{9}, \frac{7}{9}\right) + \frac{11}{20} I\left(\frac{8}{11}, \frac{3}{11}\right) \\ &= \frac{9}{20} \left(-\frac{2}{9} \log_2 \frac{2}{9} - \frac{7}{9} \log_2 \frac{7}{9}\right) + \frac{11}{20} \left(-\frac{8}{11} \log_2 \frac{8}{11} - \frac{3}{11} \log_2 \frac{3}{11}\right) \\ &= \frac{9}{20} (0.7642) + \frac{11}{20} (0.8454) \\ &= 0.8089 \end{aligned}$$

$$\begin{aligned} \text{Remainder}(A) &= \sum_{i=1}^v \frac{p_i + n_i}{p+n} I\left(\frac{p_i}{p_i + n_i}, \frac{n_i}{p_i + n_i}\right) \\ I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) &= -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n} \end{aligned}$$

Information gain for choosing an attribute:
Example

$$\begin{aligned} \text{Remainder}(\text{Weather}) &= \frac{6+5}{10+10} I\left(\frac{6}{6+5}, \frac{5}{6+5}\right) + \frac{4+5}{10+10} I\left(\frac{4}{4+5}, \frac{5}{4+5}\right) \\ &= \frac{11}{20} I\left(\frac{6}{11}, \frac{5}{11}\right) + \frac{9}{20} I\left(\frac{4}{9}, \frac{5}{9}\right) \\ &= \frac{11}{20} \left(-\frac{6}{11} \log_2 \frac{6}{11} - \frac{5}{11} \log_2 \frac{5}{11}\right) + \frac{9}{20} \left(-\frac{4}{9} \log_2 \frac{4}{9} - \frac{5}{9} \log_2 \frac{5}{9}\right) \\ &= \frac{11}{20} (0.9940) + \frac{9}{20} (0.9911) \\ &= 0.9927 \end{aligned}$$

$$\begin{aligned} \text{Remainder}(A) &= \sum_{i=1}^v \frac{p_i + n_i}{p+n} I\left(\frac{p_i}{p_i + n_i}, \frac{n_i}{p_i + n_i}\right) \\ I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) &= -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n} \end{aligned}$$

Information gain for choosing an attribute:
Example

$$\begin{aligned} \text{Gain}(\text{Gender}) &= I\left(\frac{10}{10+10}, \frac{10}{10+10}\right) - \text{Remainder}(\text{Gender}) \\ &= I\left(\frac{1}{2}, \frac{1}{2}\right) - 0.8089 \\ &= 0.1911 \\ \text{Gain}(\text{Weather}) &= I\left(\frac{10}{10+10}, \frac{10}{10+10}\right) - \text{Remainder}(\text{Weather}) \\ &= I\left(\frac{1}{2}, \frac{1}{2}\right) - 0.9927 \\ &= 0.0073 \end{aligned}$$

The gain from choosing gender to predict the result is greater than the gain from choosing weather to predict the result

A method for assessing performance of any supervised learning algorithm (1)

- A learning algorithm is good if the produced hypotheses can predict classifications of unseen examples.

A method for assessing performance of any supervised learning algorithm (2)

- Collect a large set of examples
- Divide them into
 - a training set
 - a test set
- Use training set to generate a hypothesis function: h (assume classification task)
 - h takes any valid input vector and gives a predicted output
- Measure the proportion of examples in the test set that are correctly classified by h
 - This is the estimated accuracy of h

Factors affecting performance

- Inconsistent examples
- Vital information missing
- Irrelevant attributes used

Overfitting

- Large set of possible hypotheses, algorithm can find meaningless 'regularity' in the data
- Solutions
 - Cross-validation (K-fold cross-validation)
 - Technique specific (e.g. Decision tree pruning, Neural network early stopping)

Overfitting: Decision tree pruning (1)

- Don't consider attributes that are not relevant
- Only consider attributes that provide a statistically significant information gain

Overfitting: Neural network early stopping

- If neural networks are trained for 'too long' they can become too specified for the training examples they have been given
- Particularly a problem if there are not enough training examples
- One solution is to stop training when the network is still general

Overfitting: Cross-validation

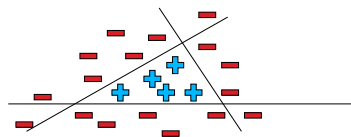
- Any learning algorithm
- Estimate how well each hypothesis will predict unseen data
- K-fold cross validation:
 - Run k experiments, testing on a different $1/k$ of the data each time
 - $k=5, k=10$
 - $k=n$ is leave-one-out cross-validation

Overfitting: Decision tree pruning (2)

- Probability that, given there is no underlying pattern, a sample of size v would exhibit the observed deviation from the expected distribution of positive and negative examples
- χ^2 pruning
- Comparing the actual numbers of positive and negative examples in a subset to the expected numbers of positive and negative examples

Ensemble Learning

- Select an ensemble of hypotheses from the hypothesis space
- Combine their predictions
- E.g. 100 classifiers that vote on the best classification for a new example
- Can result in a more expressive hypothesis space without much more complexity



Ensemble Learning: Boosting

- A widely used ensemble method
 - First trained on all examples equally
 - Second trained weakly on those classified well by 1 and strongly on those not classified well by 1
 - Third trained weakly on those classified well by 1 and 2 and strongly on those not classified well by 1 and 2
 - ...
- Final ensemble hypothesis is a weighted-majority combination of all hypothesis, weighted according to performance on training set

Types of machine learning techniques

Technique	Style of data	Type of learning	Representation/process
K-means	Numeric	Unsupervised, classification	Numeric (means of subsets of data)
Version-space learning	Nominal	Supervised, classification	Logical rules
ID3	Nominal	Supervised, classification	Decision tree
Neural networks	Numeric	Mainly supervised, both classification and regression	Numeric (weights and a network topology)
Current best learning	Nominal	Supervised, classification	Logical rules
Naïve Bayes' Classifier	Numeric	Supervised, classification	Numeric (probabilities)

Decision Tree

- Symbolic representation of acquired knowledge
- One of the simplest, most successful forms of learning algorithm
- Construction: input of a set of examples that are made up of attributes and an output
- Prediction: input of a situation made up of attributes, the tree returns the predicted output value

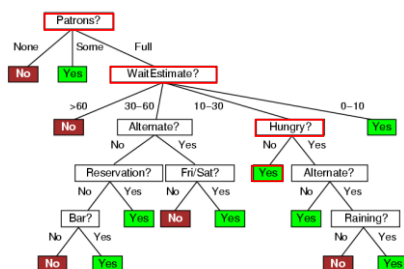
Decision tree: Algorithm

```

function DECISION-TREE-LEARNING(examples, attributes, default) returns a decision tree
  inputs: examples, set of examples
         attributes, set of attributes
         default, default value for the goal predicate

  if examples is empty then return default
  else if all examples have the same classification then return the classification
  else if attributes is empty then return MAJORITY-VALUE(examples)
  else
    best ← CHOOSE-ATTRIBUTE(attributes, examples)
    tree ← a new decision tree with root test best
    for each value vi of best do
      examplesi ← {elements of examples with best = vi}
      subtree ← DECISION-TREE-LEARNING(examplesi, attributes - best,
                                       MAJORITY-VALUE(examplesi))
      add a branch to tree with label vi and subtree subtree
    end
  return tree
    
```

Decision tree: Example



$\forall r \text{ Patrons}(r, \text{Full}) \wedge \text{WaitEstimate}(r, >10-30) \wedge \text{Hungry}(r, \text{No}) \Rightarrow \text{WillWait}(r)$

Current best learning

- Given a set of examples, find the best hypothesis / set of hypotheses that matches, with an appropriate degree of generalisation vs specialisation
- Maintain a single hypothesis, adjust as new examples are available by
 - generalising or
 - specialising

Current best learning: Algorithm

```

function CURRENT-BEST-LEARNING(examples) returns a hypothesis
   $H \leftarrow$  any hypothesis consistent with the first example in examples
  for each remaining example in examples do
    if  $e$  is false positive for  $H$  then
       $H \leftarrow$  choose a specialization of  $H$  consistent with examples
    else if  $e$  is false negative for  $H$  then
       $H \leftarrow$  choose a generalization of  $H$  consistent with examples
    if no consistent specialization/generalization can be found then fail
  end
  return  $H$ 
    
```

Current best learning: Example

Alternate	Bar	Fri/Sat	Hungry	Patrons	Price	Rain	Reservation	Type	Est time	Will wait?
TRUE	FALSE	FALSE	TRUE	Some	\$\$\$	FALSE	TRUE	French	0-10	TRUE
TRUE	FALSE	FALSE	TRUE	Full	\$	FALSE	FALSE	Thai	30-60	FALSE
FALSE	TRUE	FALSE	FALSE	Some	\$	FALSE	FALSE	Burger	0-10	TRUE
TRUE	FALSE	TRUE	TRUE	Full	\$	FALSE	FALSE	Thai	10-30	TRUE
TRUE	FALSE	TRUE	FALSE	Full	\$\$\$	FALSE	TRUE	French	>60	FALSE
FALSE	TRUE	FALSE	TRUE	Some	\$\$	TRUE	TRUE	Italian	0-10	TRUE
FALSE	TRUE	FALSE	FALSE	None	\$	TRUE	FALSE	Burger	0-10	FALSE
FALSE	FALSE	FALSE	TRUE	Some	\$\$	TRUE	TRUE	Thai	0-10	TRUE
FALSE	TRUE	TRUE	FALSE	Full	\$	TRUE	FALSE	Burger	>60	FALSE
TRUE	TRUE	TRUE	TRUE	Full	\$\$\$	FALSE	TRUE	Italian	10-30	FALSE
FALSE	FALSE	FALSE	FALSE	None	\$	FALSE	FALSE	Thai	0-10	FALSE
TRUE	TRUE	TRUE	TRUE	Full	\$	FALSE	FALSE	Burger	30-60	TRUE

$$h_1: \forall x \text{ WillWait}(x) \Leftrightarrow \text{Alternate}(x)$$

$$h_2: \forall x \text{ WillWait}(x) \Leftrightarrow \text{Alternate}(x) \wedge \text{Patrons}(x, \text{Some})$$

$$h_3: \forall x \text{ WillWait}(x) \Leftrightarrow \text{Patrons}(x, \text{Some})$$

$$h_4: \forall x \text{ WillWait}(x) \Leftrightarrow \text{Patrons}(x, \text{Some}) \vee [\text{Patrons}(x, \text{Full}) \wedge \text{Fri/Sat}(x)]$$

...

Naïve Bayes Classifier

- Calculates the probability of each available hypothesis (model or input to output function), given the data
- Predicts output using *all* available hypotheses, weighted by their probabilities – not just using a *single* hypothesis
- Learning is thus reduced to probabilistic inference
- A limited set of hypotheses are considered

Naïve Bayes Classifier: Equations

- D = all data
- d = observed value
- d_j = random variable j
- h_i = hypothesis i
- C = class
- X = variables

$$P(\mathbf{d} | h_i) = \prod_j P(d_j | h_i)$$

$$P(d' | \mathbf{d}) \propto \sum_i P(d' | h_i) P(h_i) \prod_{j=1}^n P(d_j | h_i)$$

$$P(C | X_1, X_2, \dots, X_m) = \alpha P(C) \prod_{i=1}^m P(X_i | C)$$

Naïve Bayes Classifier: Example (1)

- Very large candy bags, either
 - h_1 : 100% cherry
 - h_2 : 75% cherry and 25% lime
 - h_3 : 50% cherry and 50% lime
 - h_4 : 25% cherry and 75% lime
 - h_5 : 100% lime

$$\begin{aligned}
 P(\mathbf{d} | h_1) &= [P(\text{lime} | h_1)]^5 = 0.00 \\
 P(\mathbf{d} | h_2) &= [P(\text{lime} | h_2)]^5 = 0.00 \\
 P(\mathbf{d} | h_3) &= [P(\text{lime} | h_3)]^5 = 0.03 \\
 P(\mathbf{d} | h_4) &= [P(\text{lime} | h_4)]^5 = 0.24 \\
 P(\mathbf{d} | h_5) &= [P(\text{lime} | h_5)]^5 = 1.00
 \end{aligned}$$



Naïve Bayes Classifier: Example (2)

- Remember the hypothesis prior...
- Note that all hypotheses are being used to predict

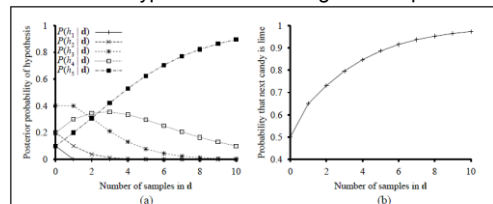
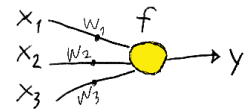


Figure 20.1 (a) Posterior probabilities $P(h_i | d_1, \dots, d_N)$ from Equation (20.1). The number of observations N ranges from 1 to 10, and each observation is of a lime candy. (b) Bayesian prediction $P(d_{N+1} = \text{lime} | d_1, \dots, d_N)$ from Equation (20.2).

Neural networks

- Computational models, consisting of simple processing elements, for representing and learning functions and procedures from examples
- Crude mathematical descriptions of biological neural circuitry and functionality
- Tools for modelling cognitive phenomena
- Tools for statistical classification and regression applications

Neural networks: Equations



Inputs Output

Activation (Sigmoid or Threshold):

$$net = \sum_{i=1}^n w_i x_i$$

$$f(net) = \frac{1}{1 + e^{-net}}$$

$$f(net) = \begin{cases} 0 & \text{if } net < 0 \\ 1 & \text{if } net \geq 0 \end{cases}$$

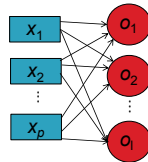
Update (backpropagation):

$$E = \sum_{i=1}^{n_{in}} \sum_{k=1}^m E_{ik}$$

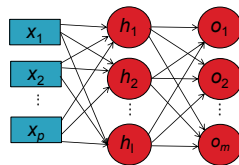
$$\Delta w_i \propto - \frac{\partial E}{\partial w_i}$$

Neural networks: Example networks

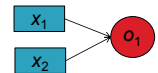
- Single layer networks



- Multi-layer networks



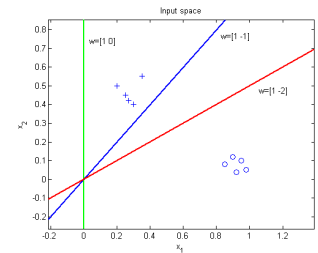
Neural networks: Example



- A network with two inputs: x1 and x2
- and one output: 0 or 1
- bias $\beta=0$

$$net = \sum_{i=1}^n x_i w_i + \beta$$

$$f(net) = \begin{cases} 0 & \text{if } net < 0 \\ 1 & \text{if } net \geq 0 \end{cases}$$



Summary

- Machine Learning
- Learning Agent
- Learning
- Inductive learning
- Hypothesis space
- Information Theory
- Performance of supervised learning algorithms
- Machine learning techniques