Outline

- What is a well defined learning problem?
- An example: Qubic
- What questions should we ask about machine learning?

Copyright Tom Mitchell, 14 September 1994

Why Study Machine Learning?

Save us from writing 10^9 programs

Understand human learning and teaching

Good rate of recent progress:

- Real applications: robot driving, speech recognition, loan risk evaluation
- Theory: confidence = f(|training data|, |hypothesis space|)
- Human learning: better cognitive and neural models

Copyright Tom Mitchell, 14 September 1994

What is a Well-Defined Learning Task?

Learning = Improve with experience at some task

- Improve over task class T,
- with respect to performance measure P,
- based on experience E.

E.g., Qubic (4x4x4 TicTacToe):

- T: All Qubic games against Merrick
- P: % of games won
- E: opportunity to play against Merrick for 100 hours

Copyright Tom Mitchell, 14 September 1994

More Learning Tasks

Recognizing handwritten mail addresses

- T: Recognize destination city in handwritten addresses
- P: % of mail sorted to correct city
- E: 100,000 handwritten addresses with known destination city

Driving a robot vehicle

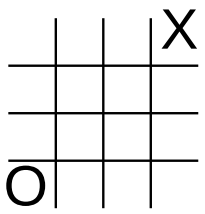
- T: Drive on public 4-lane highways
- P: Speed/[(1+Crashes)^{100000}]
- E: 20 minutes of training camera images and steering commands

Database

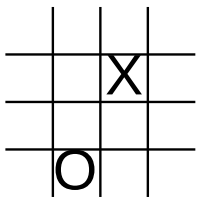
- T: Answer database queries
- P: % of queries correctly answered
- E: Database updates

Copyright Tom Mitchell, 14 September 1994

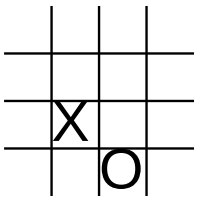
An Example: Qubic



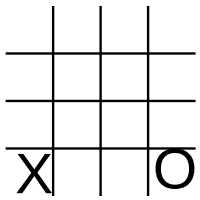
Level 1



Level 2



Level 3



Level 4

A learning problem:

- T: All Qubic games against Merrick
- P: % of games won
- E: opportunity to play against Merrick for 100 hours

Exactly what should be learned?

How should it be represented?

How shall we assign credit/blame given delayed feedback?

Copyright Tom Mitchell, 14 September 1994

Qubic (4x4x4 TicTacToe)

Qubic:

- T: All Qubic games against Merrick
- P: % of games won
- E: opportunity to play against Merrick for 100 hours
- Target function: $Eval^*$

  $Eval^*: Board \rightarrow [-100, 100]$

  where
  $Board$ is the set of legal board positions
  $Eval^*(b)$ = +100 if b has 4 X's in a row
  $Eval^*(b)$ = -100 if b has 4 O's in a row
  $Eval^*(b)$ = 0 if b is drawn
  $Eval^*(b)$ = expected final board value under the optimal strategy

- $Eval$ approximation to $Eval^*$: $Eval = w_0 + w_1x_1 + w_2x_2 + ... + w_{10}x_{10}$
  - $x_1(x_6)$: # of live lines containing 1 X (O)
  - $x_2(x_7)$: # of live lines containing 2 Xs (Os)
  - $x_3(x_8)$: # of live lines containing 3 Xs (Os)
  - $x_4(x_9)$: # of line intersections owned by X (O)
  - $x_5(x_{10})$: # of line intersections owned by X with 2 or more X's per line (O's)

Copyright Tom Mitchell, 14 September 1994

Learning the Coefficients to Approximate a Function

Want to learn $Eval$ that "best" approximates true $Eval^*$

- "best" = Minimize squared error E

$$E=\sum_{i=1}^{n}[Eval^*(x_i)-Eval(x_i)]^2$$

Gradient descent search:

1. Initialize all weights w_i to 0
2. For each training example, apply LMS rule

LMS Weight tuning rule:

$$\vec{w}_{t+1} \leftarrow \vec{w}_t + \frac{c\,2e_t\vec{x}_t}{|\vec{x}_t|^2}$$

where $e_t=Eval^*(x_t)-Eval(x_t)$

$0<c<1$ is the learning rate

Note $Eval(\vec{x}) = \vec{w}\vec{x}$, $\frac{dE}{d\vec{w}}=\sum_{i=1}^{n}2e_i\vec{x}_i$

Copyright Tom Mitchell, 14 September 1994

Obtaining Training Instances

How can we obtain training values of $Eval^*(x)$???

Could estimate them using $Eval$ and lookahead search:

- $Eval\ (x) \leftarrow Max(i)\ [Eval(Successor_i(x)]$
- The better $Eval$ gets, the better this estimate
- Does this scheme converge??
- How many steps ahead should we look?

Copyright Tom Mitchell, 14 September 1994

Questions about Learning Eval

Will this function approximation algorithm converge?

- Under what conditions?
- After how many examples?
- After how many errors?
- After how many CPU cycles?
- Even if $Eval^*$ cannot be represented as linear function?

How does indirect estimation of training values affect convergence?

Can clever experimentation improve correctness/speed of learning?

How can learner improve representation?

Is this how humans learn?

Copyright Tom Mitchell, 14 September 1994

The Central Issue: Generalizing to New Cases

Inductive Learning:

- Find syntactic regularities among training examples

Analytical Learning:

- Use prior knowledge to identify relevant features

Copyright Tom Mitchell, 14 September 1994

Qubic (4x4x4 TicTacToe)

Qubic:

- T: All Qubic games against Merrick
- P: % of games won
- E: opportunity to play against Merrick for 100 hours
- Target function:

  Win-In-3-Moves(b) ≡
  $(\exists m1)(\forall m2)(\exists m3)Won(m3(m2(m1(b))))$
  $\wedge \neg(Win\text{–}In\text{–}2\text{–}Moves(b))$

- Target representation: Rules, e.g.,

  IF InLine(?l9,?p5,?p6,?p3,?p8)
  InLine(?l2,?p1,?p2,?p3,?p4)
  Open(?p3), Open(?p2), Open(?p6)
  X(?p1),X(?p4),X(?p5),X(?p6)
  THEN Win-In-3-Moves=T

- Approach: Explanation-based learning

Copyright Tom Mitchell, 14 September 1994

Explanation Based Learning

Key idea: Rely on prior knowledge to replace need for (some) data

Given:

- Target boolean function: Win-In-3-Moves
- Domain knowledge:

  Won ← ...
  Win-In-3-Moves(b) ← ...
  Legal Moves
  ...

- Training examples:

Determine:

- Justifiable generalization of training example

Explanation-Based Learning Algorithm

1. Explain (prove) that observed examples satisfies target function
2. Generalization $\leftarrow$ weakest preconditions of explanation

Generalization guided by initial knowledge rather than syntactic similarity

Can construct new features

Copyright Tom Mitchell, 14 September 1994

Questions about Explanation-based Learning

- How does prior knowledge affect the complexity of learning?
- How can one use *imperfect* prior knowledge?
- What if explanations are intractable?
- Can we unify inductive and analytical methods?

Copyright Tom Mitchell, 14 September 1994

Some Questions about Machine Learning

What is the best way to reduce the learning task to function approximation?

What are good algorithms for function approximation?

What is the relationship between

- Confidence that learned hypothesis is correct
- Number of training examples
- Amount of prior knowledge
- Nature of hypothesis space
- Nature of training examples provided

When and what should an agent try to learn?

How can new representations be learned?

Copyright Tom Mitchell, 14 September 1994