Massachusetts Institute of Technology

6.034 Artificial Intelligence

Solutions # **Final96** 6034 Item # **33**

Problem 1 Rules

Part A

```
Step    Ready to fire    Selected Rule    Assertion Added
1       R1 R3 R7         R1               Fuzzy is a mammal
2       R5 R7            R5               Fuzzy is a carnivore
3       R7               R7               Fuzzy is a feline
4       R9               R9               Fuzzy is a tiger
```

(R3's antecedents are in the database in each of the steps 2,3,4 but it is not triggered because its conclusion is redundant with R1.)

Part B

Here, lying on its "side" is the complete tree that would be sprouted by backward chaining.

```
R9:tiger
    carnivore
        R4: mammal
                    R1: milk
                    R2: warm blooded

                    R3: milk
                        doesn't lay eggs
                        live young
            meat
        R5: mammal
                    R1: milk
                    R2: warm blooded
                    R3: milk
                        doesn't lay eggs
                        live young
            pointed teeth
```

```
        forward looking eyes
    R6: mammal
                R1: milk
                R2: warm blooded
                R3: milk
                    doesn't lay eggs
                    live young
        stealthy hunter

feline
    R7: whiskers
        claws

tawny

stripes
    R8: light fur
        vertical bands
```

The questions asked in depth-first order, and their answers, are:

```
milk?           yes
meat?           no
pointed teeth?  yes
forward eyes?   yes
whiskers?       yes
claws           yes
tawny?          yes
stripes?        yes
```

Part C

In breadth-first order the questions are:

```
tawny?          yes
meat?           no
pointed teeth?  yes
forward eyes?   yes
stealthy?       no
whiskers?       yes
claws?          yes
light fur?      no
vertical bands  no
milk?           yes
stripes?        yes     [having failed to infer stripes, the system eventually asks]
```

A backward chaining system asks only when it can not infer. That in turn happens when either there are no rules by which it can infer something, or it has tried all the rules and they didn't work. Hence in the example above the first thing asked is "tawny" because that's the thing nearest the top of the tree that has no rules by which it can be inferred.

Part D

Answer 1 is correct. Depth first is preferable since it produces a more coherent set of questions.

Problem 2 Constraints

Part A

Eiher (3) "all stations for a trip" or (4) "the track segments for a trip" are reasonable answers. The key observation is that the variables are related to the segments of EACH trip. Below, we assume that the answer (3) is chosen.

Part B

A sample variable is A_1, i.e. the station A as part of Trip 1.

Part C

The varible domains are (2) "all the possible times that a train can depart a station, that are earlier than the maximum time of the trip." One could actually refine this more by using the minimal duration of the trip from that station, i.e. by eliminating times that would not enable to reach the end of the trip before the required final time.

Part D

$$A_1 \in \{12:00am, 12:05am, ..., 1:20pm, 1:25pm\}$$

Part E

The constraints are:

1. The departure times have to be ordered consistently and there must be time to reach the next station before the scheduled departure time from that station. For example, the departure

time from B, in trip 1, has to be at least 35 minutes earlier than the departure time from E, in trip 1.

2. Two trains moving in opposite directions cannot occupy the same track. One could even be more conservative and simply require that track segments not be shared, even by trains traveling in the same direction. For example, the leg of trip 1 that goes from E to F cannot overlap in time with the leg of trip 2 that goes from F to E.

Part F

Here's a constraint graph with the constraints labelled by the type of constraint (1 or 2 from above).

```
      1        1        1        1
A1 ----- B1 ----- E1 ----- F1 ----- G1
                   |        |
          ------------------
          |2       |2
         G2 ----- F2 ----- E2 ----- C2
              1        1        1
```

Part G

We did not go into the Min-conflict Hill Climbing procedure much this term, but here are the answers.

Algorithm	ALL	ONE	UNL	LGE	INI
Backtracking	Yes	Yes			
AC-3			Yes		
Backtracking + Forward Check	Yes	Yes			
Backtracking + AC-3	Yes	Yes			
Min-conflict Hill Climb (with conflict weights)				Yes	Yes

Backtracking is, in principle, guarateed to find an answer if it exists and by continuing to run it one could get all the answers. Adding constraint propagation to backtracking preserves this property. The constraint propagation by itself is unlikely to solve complex problems. The MC HillClimbing is more likely to work on large problems with large domains; it needs an initial (usually random) assignment to get started.

Problem 3 Frames and Language

Please ignore Problem 3 – we did not cover most of this.

Problem 4 Neural Nets

Part A

1. A step threshold activation function: $\sum_i w_i x_i > 0$. The weights should be: $W_1 = 1/2$ and $W_2 = 1$.
2. A line going through the origin that passes through (-2, 1) and (2, -1).
3. Add a new input to the unit that is connected to a constant -1 and has a separate weight, call it W_0.
4. $W_0 = 2$, $W_1 = 1/2$ and $W_2 = 1$.
5. A line going through (0, 2) and (4, 0).

Part B

Backprop is just gradient descent on the error $1/2(d-y)^2$, so in this case we get:

$$\Delta W_i = -r(d-y)dy/dW_i$$

with

$$dy/dW_i = -y(1-y)x_i$$

so

$$\Delta W_i = r(d-y)y(1-y)x_i$$

Part C

1. There are 70 inputs to the net.
2. Each input represents a pixel in one of the images.
3. Have 10 binary outputs, each representing the likelihood of one of the digits. Each output will be produced by a sigmoid and is, therefore, a continuous value in the range from 0 to 1. We can decide what digit it is by picking the digit having the maximal output value (highest likelihood). One could also have 4 outputs that encode the answer digit in binary, but that would be harder to learn.

Part D

Methods 2, 6 and 8 all help combat overfitting. That is, use smaller nets, don't overtrain and (best of all) use a validation set to decide when to stop training.

Problem 5 Learning

Part A

The weighted disorder of the top node is:

$$3/6 * (-1/3 \log(1/3) - 2/3 \log(2/3)) + 3/6(-1log(1) - 0log(0)) = 0.46$$

Part B

digit	prediction
0	even
1	even
5	odd
7	odd

Part C

Note the typo in the figure caption. This is the description for digit 8.

digit	induction heuristics	affected node
8	initial model	–
9	require	S5
3	ignore	
2	Drop-link	S2, S6
4	Drop-link	S1, S5, S7
6	Drop-link	S3

The resulting model just has S4 being 1.

Part D

The result from Procedure W and the decision tree do not agree. In fact, in this context, Proc. W cannot ever learn a description like that represented by the tree since the tree concept involves an OR. There are two ways for a digit to be even. Procedure W (without the enlarge-set heuristic) will look for a single AND description.

Part E

digit	prediction
0	even (closest to 8)
1	even (closest to 2)
5	even (closest to 6)
7	odd (closest to 3)

Part F

The performance of an identification tree system would probably be unaffected by the extraneous segments, since they are uncorrelated with the desired output, they would not be chosen.

The performance of Procedure W would probably not be affected as long as a teacher chose an order of presentation which would cause the system to drop all the useless segments.

The performance of the nearest neighbor algorithm would substantially decrease since it has no direct mechanism for ignoring the extraneous segments. The Hamming distance computation would be swamped by differences in the irrelevant segments and tend to produce random results.

Problem 6 Miscellaneous

Many of these reflect material that was not covered this term, e.g. CLOS.

1. True
2. True
3. True
4. False
5. False (not covered this term)
6. False (not covered this term)
7. True
8. False
9. False (not covered this term)
10. True