Problem 2 Bayes Nets (31 points)

GXXQXXR

Q has no dependencies:

Var Name	Var Name	Var Name	Cond Prob
			P(Q | _,_,_)
			1/3

G depends on Q only

Var Name	Var Name	Var Name	Cond Prob
Q			P(G | __,__,__)
T			0.95
F			0.5

R also depends on Q only

Var Name	Var Name	Var Name	Cond Prob
Q			P(R | __,__,__)
T			0.9
F			0.75

A depends on both G and R

Var Name	Var Name	Var Name	Cond Prob
G	R		P(A | __,__,__)
T	T		0.75
T	F		0.5
F	T		0.5
F	F		0.0

Part A (8 points)

The noisy-or model will work, as you can see by plugging numbers into the noisy-or formula.

It is especially convenient that P(A| G=F, R=F) = 0, for that tells us that the leak probability is zero. With a 0 leak, we can read the causal probabilities from the tables, without further computation.

P(A | G, R)

= 1 - (1 - P_C(A | G=T, R=F)) (1 - P_C(A | G=F, R=T)) (1 - P_C(A | leak)) =

1 - (1 - 0.5) (1 - 0.5) (1 - 0)

=

0.75

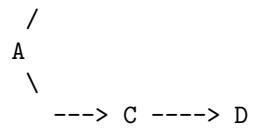
Then, note that P(A | G, R), given by the formula, is the same as the value given the table. Hence, noisy-or does model the situation.

Part B (6 points)

The following problems are independent of the ones above.

Given the following Bayes Net:

```
---> B
```



By inspection, from the diagram, the dependencies indicate:

8 P(A=a,B=b,C=c,D=d) = P(D=d| C=c) P(B=b| A=a) P(C=c| A=a) P(A=a)

Part C (8 points)

By the definition of conditional probability:

P(B=T,D=F | A=T,C=F) = P(A=T, B=T, C=F, D=F) / P(A=T, C=F)

But

P(A=T, C=F) = P(A=T, B=T, C=F, D=T) +P(A=T, B=F, C=F, D=T) +P(A=T, B=T, C=F, D=F) +P(A=T, B=F, C=F, D=F)

Hence, given

1	P(A=T,B=T,C=T,D=T)	9	P(A=F,B=T,C=T,D=T)
2	P(A=T,B=T,C=T,D=F)	10	P(A=F,B=T,C=T,D=F)
3	P(A=T,B=T,C=F,D=T)	11	P(A=F,B=T,C=F,D=T)
4	P(A=T,B=T,C=F,D=F)	12	P(A=F,B=T,C=F,D=F)
5	P(A=T,B=F,C=T,D=T)	13	P(A=F,B=F,C=T,D=T)
6	P(A=T,B=F,C=T,D=F)	14	P(A=F,B=F,C=T,D=F)
7	P(A=T,B=F,C=F,D=T)	15	P(A=F,B=F,C=F,D=T)
8	P(A=T,B=F,C=F,D=F)	16	P(A=F,B=F,C=F,D=F)

Then,

P(B=T,D=F | A=T,C=F) = 4 / (3 + 7 + 4 + 8)

Part D (4 points)

He is more likely to be reacting to the tree built for Group A, for which the number of samples is just about the number of terminals. Each sample has ended up in a group by itself; and because of that individualized handling of each sample, the overfitting label comes to mind.

Problem 4 How they work (23 points)

Part A Neural Nets

A.1 (5 points)

The training software is not likely to work with the mirror-image threshold function. When you derive the weight-change formulas using the mirror-image threshold function, you see that the incorrect sign in the exponent propagates through the derivation, producing a change in sign in the weight changes. That is, the original weight-change formulas will drive the network in the wrong direction, away from the desired outputs.

A.2 (5 points)

The training software may work with the altered training function, but behavior will certainly be different. When you derive the weight-change formulas using the threshold function with the altered exponent, you see that the multiplier of 10^6 propagates through the derivation, producing a multiplicative effect on the weight changes. That is, the original weight-change formulas will drive the network in the right direction, but it is likely that the rate constant would need adjustment to get reasonable behavior.

A.3 (5 points)

If you divide all the weights by 1000, the output of each sigmoid box will be unchanged; thus dividing all weights by 1000 compensates exactly for the multiplier of 1000 in the exponent.