IS 605 - Assignment 8

Dan Fanelli

March 22, 2016

Problem Set 1:

Your colleague either commutes by train or by the bus. 20 days of the month, she takes the train and the remaining 10 days she takes the bus. If she takes the train, she reaches work on time with a probability of 0.9. If she takes the bus, she frequently gets stuck in traffic and reaches work on time with a probability of 0.5. Given that she was on time today, what is the probability that she took the bus to work today?

```
prob_train <- (20/30)
prob_bus <- (10/30)
prob_ontime_given_bus <- 0.5
prob_ontime_given_train <- 0.9

prob_bus_given_on_time <- (prob_bus * prob_ontime_given_bus) /
  ((prob_bus * prob_ontime_given_bus) + (prob_train * prob_ontime_given_train))

prob_bus_given_on_time
```

```
## [1] 0.2173913
```

```
cat('Given you were on time, ', prob_bus_given_on_time * 100, '% chance you took the bus.')
```

```
## Given you were on time,  21.73913 % chance you took the bus.
```

Problem Set 2:

In the Grade Network that we looked at in the notes, what happens to the probability of Difficulty of Course when you present the evidence that the received recommendation letter was good?

Answer: The probabilities of difficulty of course should go up too - the same way we can derive the probabilities of the input based on the probabilities of the output, we can do the same for multiple node cause and effect graphs.

In addition, now present the evidence that both SAT scores were good and the letter of recommendation was good, What is the probability of the Difficulty of Course now? You should use the gRain package in R to build your network and perform these calculations.

You may need to install RBGL package from BioConductor in R to get gRain working. See http://www.bioconductor.org/packages/release/bioc/html/RBGL.html for instructions on RBGL.

```
library(gRain)
```

```
## Loading required package: gRbase
```

```
noyes <- c("no", "yes")
lowhi <- c("low", "high")

difficulty.node <- cptable(~difficulty, values=c(0.3,0.7), levels=noyes)
intelligence.node <- cptable(~intelligence, values=c(0.8, 0.2), levels=lowhi)
grade.node <- cptable(~grade|difficulty:intelligence,
    values=c(0.6, 0.01, 0.8, 0.1, 0.4, 0.99, 0.2, 0.9), levels=lowhi)

letter.node <- cptable(~letter|grade, values=c(0.9, 0.05, 0.1, 0.95), levels=lowhi)
sat.node <- cptable(~sat|intelligence, values=c(0.9, 0.2, 0.1, 0.8), levels=lowhi)

probs_list <- compileCPT(list(difficulty.node, intelligence.node, grade.node, letter.node, sat.node))

probs_list$difficulty
```

```
## difficulty
##  no yes
## 0.3 0.7
```

```
probs_list$grade
```

```
## , , intelligence = low
##
##       difficulty
## grade          no       yes
##   low  0.98360656 0.8888889
##   high 0.01639344 0.1111111
##
## , , intelligence = high
##
##       difficulty
## grade         no       yes
##   low  0.2877698 0.1818182
##   high 0.7122302 0.8181818
```

```
probs_list$letter
```

```
##       grade
## letter        low      high
##   low  0.94736842 0.0952381
##   high 0.05263158 0.9047619
```

Checkpoint:

What happens to the probability of Difficulty of Course when you present the evidence that the received recommendation letter was good?

We are now presented with the FACT that the reccomendation letter was good, so 1) confirm the base probability before the new evidence, 2) change the evidence, 3) requery the network:

```
net1 <- grain(probs_list)
net1
```

```
## Independence network: Compiled: FALSE Propagated: FALSE
##   Nodes: chr [1:5] "difficulty" "intelligence" "grade" "letter" ...
```

```
querygrain(net1, nodes=c("difficulty"))
```

```
## $difficulty
## difficulty
##  no yes
## 0.3 0.7
```

```
net2 <- setEvidence(net1, evidence=list(letter="high"))
querygrain(net2, nodes=c("difficulty"))
```

```
## $difficulty
## difficulty
##        no       yes
## 0.2286028 0.7713972
```

Conclusion:

The probability of Course Being Difficult went from **70%** to **77.1%** upon knowing that the reccomendation letter was good.