Probabilitistic Graphical Models for Fraud and Anomaly Detection in Insurance

Mick Cooney
michael.cooney@applied.ai

6 July 2016

How to Build a Model with No Data and No Domain Knowledge...

Structure of Talk

- Conditional Dependence, Independence and Bayesian Networks
- The Sprinkler Network
- Medical Non-disclosure
- Building a Model
- Expanding the Model
- Beyond Bayesian Networks
- Summary

Conditional Probability

Probability of 2D6 totalling 11?

$$(5,6) \text{ or } (6,5)$$

$$P(T = 11) = \frac{2}{36} = 0.05556$$

Conditional Probability

Probability of 2D6 totalling 11 if first dice is 5?

$$(5, 6)$$

$$P(T = 11 | D_1 = 5) = \frac{5}{6} = 0.8333$$

Conditional Dependence and Independence

Three variables, A, B, C:

A and B are independent
C depends on A
C depends on B

What happens if we learn information about C?

A and B are *conditionally dependent* on C.

2D6 Example

Define variables D_1, D_2 and T.

D_1 and D_2 are independent, T depends on both

What happens to D_2 if $T = 7$, $D_1 = 4$?

$$P(D_2 = X) = \begin{cases} 1 \text{ iff } X = 3 \\ 0 \text{ otherwise} \end{cases}$$

2D6

$T = 9$

$P(D_2)$:

1	2	3	4	5	6
0	0	0.5	0.5	0	0

$P(D_1)$:

1	2	3	4	5	6
0	0	0	0	0.5	0.5

Conditional Independence

Now suppose we have T as before, but define

$$X_1 = \begin{cases} 1 \text{ iff } T \text{ even} \\ 0 \text{ otherwise} \end{cases}$$

and

$$X_2 = \begin{cases} 1 \text{ iff } T >= 9 \\ 0 \text{ otherwise} \end{cases}$$

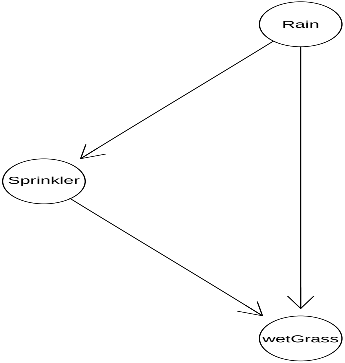
T NOT KNOWN, $X_1 \not\perp X_2$ T KNOWN, $X_1 \perp X_2$

X_1 and X_2 are *conditionally independent* on T

Probabilistic Graphical Models represent structural dependence amongst variables

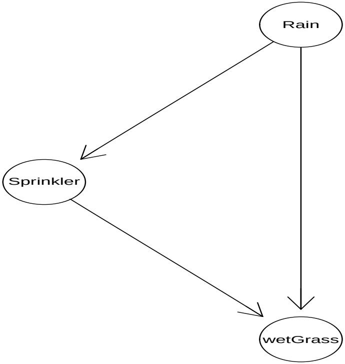
Bayesian Networks

PGM where graph is a *directed, acyclic graph* (DAG):



Dependencies represented via *Conditional Probability Tables* (CPTs)

The Sprinkler Network



Variables: (R)aining, (S)prinkler, wet(G)rass

The Sprinkler Network

```
print(sprinkler_grain$cptlist$Rain)

## Rain
## yes  no
## 0.2 0.8

ftable(sprinkler_grain$cptlist$Sprinkler, row.vars = 'Rain')

##      Sprinkler  yes   no
## Rain
## yes            0.01 0.99
## no             0.40 0.60

ftable(sprinkler_grain$cptlist$wetGrass, row.vars = c('Rain', 'Sprinkler'))

##                wetGrass  yes   no
## Rain Sprinkler
## yes  yes                0.99 0.01
##      no                 0.90 0.10
## no   yes                0.80 0.20
##      no                 0.00 1.00
```

Some Questions

What is the probability of the grass being wet?

```
querygrain(sprinkler_grain, nodes = 'wetGrass')$wetGrass

## wetGrass
##     yes      no
## 0.43618 0.56382
```

If the grass is wet, what is the probability that it is raining?

```
querygrain(sprinkler_grain, evidence = list(wetGrass = 'yes'), nodes = 'Rain')$Rain

## Rain
##      yes       no
## 0.413086 0.586914
```

If the grass is dry, what is the probability that the sprinker is on?

```
querygrain(sprinkler_grain, evidence = list(wetGrass = 'no'), nodes = 'Sprinkler')$Sprinkler

## Sprinkler
##      yes       no
## 0.113547 0.886453
```

Medical Non-disclosure

- Life/Health Insurance
- Questionnaire
- Disclosure / Non-disclosure
- Medical Examination

Focus exams on risky areas

Problems

Doing Outlier / Anomaly Detection:

- Data is sparse/missing
- Lack of output variables
- Low incidence rate
- Semi-supervised Learning

Anomaly/Outlier Detection

Anomaly/Outlier Detection

Anomaly/Outlier Detection

Anomaly/Outlier Detection

- Full automation difficult
- False positives
- Filter instead
- Human intuition

Build a Model

We want a model which, given the data observed in the policy application, allows us to estimate the probability of a subsequent medical exam changing the underwriting decision on the policy.

The model should incorporate our assumptions of the process and be as simple as possible.

Consequences

- Applicant may be unaware
- Is the nondisclosure relevant?
- Is the juice worth the squeeze?

Consequences

Consider 3 conditions:

(S)moker: Smoker, Quitter, Non-smoker

(B)MI: Normal, Overweight, Obese

Family (H)istory: None, HeartDisease

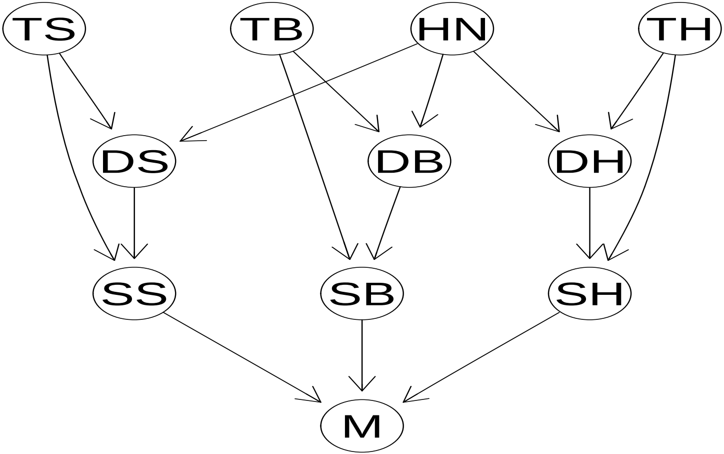
Conditions have related aspects:

T True state

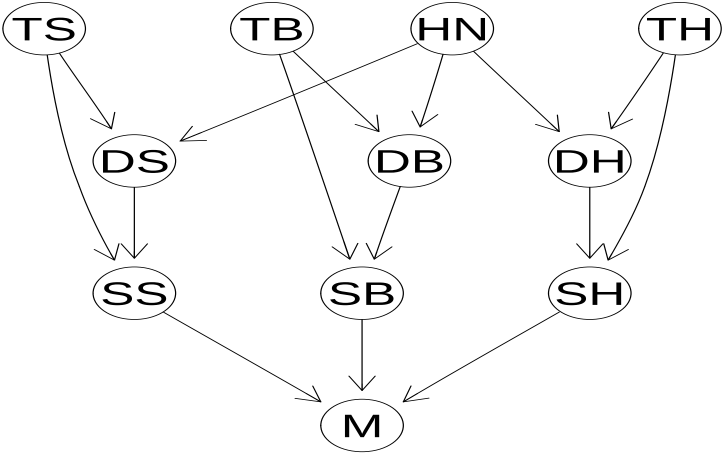
D Declared state

S Seriousness of condition's impact on decision

Medical Exam Network



Bad Teacher Syndrome


TS
TB
HN
TH
DS
DB
DH
SS
SB
SH
M

```
print(underwriting_grain$cptlist$TH)

## TH
##          None HeartDisease
##          0.95         0.05

ftable(underwriting_grain$cptlist$DH, row.vars = c('HN', 'TH'))

##                           DH None HeartDisease
## HN        TH
## Dishonest None                0.9          0.1
##           HeartDisease        0.5          0.5
## Honest    None                0.9          0.1
##           HeartDisease        0.1          0.9

ftable(underwriting_grain$cptlist$SH, row.vars = c('TH', 'DH'))

##                              SH Serious NotSerious
## TH           DH
## None         None                  0.01       0.99
##              HeartDisease          0.20       0.80
## HeartDisease None                  0.60       0.40
##              HeartDisease          0.10       0.90
```

Medical Exam

```
ftable(underwriting_grain$cptlist$M, row.vars = c('SS', 'SB', 'SH'))

##                                  M Medical NoMedical
## SS         SB         SH
## Serious    Serious    Serious         0.99      0.01
##                       NotSerious      0.85      0.15
##            NotSerious Serious         0.95      0.05
##                       NotSerious      0.60      0.40
## NotSerious Serious    Serious         0.90      0.10
##                       NotSerious      0.60      0.40
##            NotSerious Serious         0.85      0.15
##                       NotSerious      0.10      0.90
```

What is the unconditional probability of a medical exam finding something?

```
querygrain(underwriting_grain, nodes = 'M')$M

## M
##   Medical NoMedical
##  0.177515  0.822485
```

Too high?

Ignores business processes — may be reasonable

Lack of domain knowledge → probably flawed

Assess the Model

Declares a clean bill of health (DS = Nonsmoker, DB = Normal, DH = None)?

```
querygrain(underwriting_grain, nodes = 'M'
          ,evidence = list(DS = 'Nonsmoker'
                          ,DB = 'Normal'
                          ,DH = 'None'))$M
```

```
## M
##   Medical NoMedical
##  0.146951  0.853049
```

Declares history of heart disease? (DH = HeartDisease)?

```
querygrain(underwriting_grain, nodes = 'M'
          ,evidence = list(DS = 'Nonsmoker'
                          ,DB = 'Normal'
                          ,DH = 'HeartDisease'))$M
```

```
## M
##   Medical NoMedical
##  0.257899  0.742101
```

Expanding the Model

Beyond Bayesian Networks

Conclusions

Get In Touch

Mick Cooney
michael.cooney@applied.ai

Slides and code available on GitHub:
`https://www.github.com/kaybenleroll/dublin_r_workshops`