BN - Assignment 2

Thomas Rost

Saturday, December 13, 2014

```
library(Rgraphviz)
```

```
## Loading required package: graph
## Loading required package: grid
```

```
library(gRain)
```

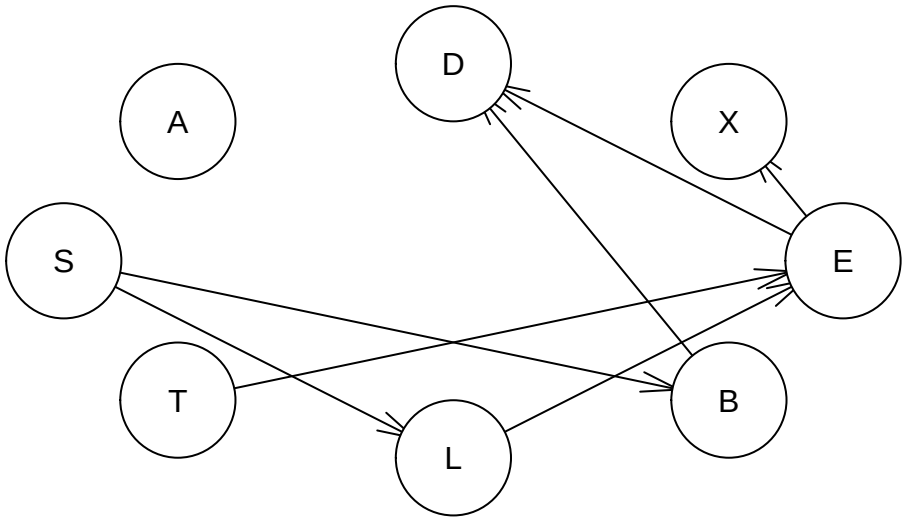
```
## Loading required package: gRbase
```

```
library(bnlearn)
```

```
##
## Attaching package: 'bnlearn'
##
## The following objects are masked from 'package:gRbase':
##
##     children, parents
```

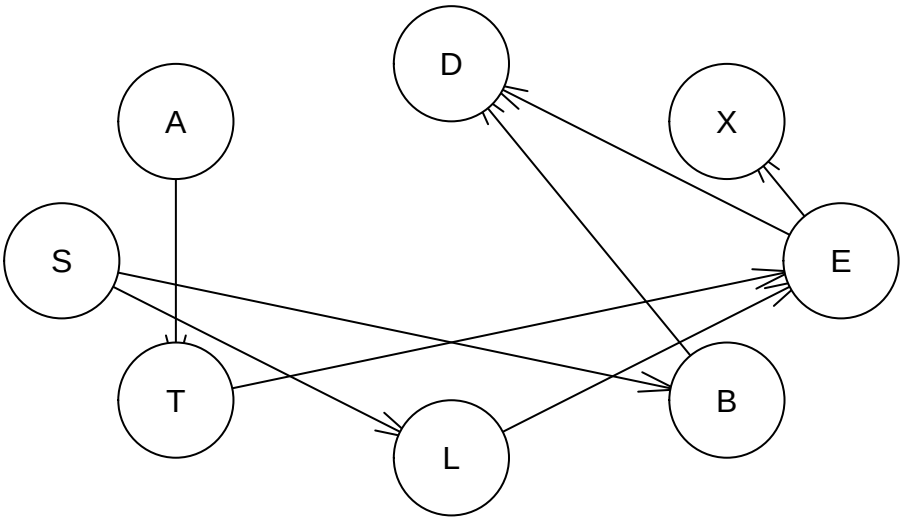
```
library(caTools)
data(asia)

net = hc(asia)
?hc
plot(net)
```




```
?plot
net <- set.arc(net,"A","T")
?set.arc

plot(net)
```




```
fitted = bn.fit(net,asia)
?bn.fit
fitted
```

```
##
##   Bayesian network parameters
##
##   Parameters of node A (multinomial distribution)
##
## Conditional probability table:
##
##     no    yes
## 0.9916 0.0084
##
##   Parameters of node S (multinomial distribution)
##
## Conditional probability table:
##
##    no   yes
## 0.497 0.503
##
##   Parameters of node T (multinomial distribution)
##
## Conditional probability table:
##
##      A
```

```
## T              no         yes
##   no  0.991528842 0.952380952
##   yes 0.008471158 0.047619048
##
##   Parameters of node L (multinomial distribution)
##
## Conditional probability table:
##
##      S
## L             no        yes
##   no  0.98631791 0.88230616
##   yes 0.01368209 0.11769384
##
##   Parameters of node B (multinomial distribution)
##
## Conditional probability table:
##
##      S
## B            no       yes
##   no  0.7006036 0.2823062
##   yes 0.2993964 0.7176938
##
##   Parameters of node E (multinomial distribution)
##
## Conditional probability table:
##
## , , L = no
##
##      T
## E     no yes
##   no   1   0
##   yes  0   1
##
## , , L = yes
##
##      T
## E     no yes
##   no   0   0
##   yes  1   1
##
##
##   Parameters of node X (multinomial distribution)
##
## Conditional probability table:
##
##      E
## X              no         yes
##   no  0.956587473 0.005405405
##   yes 0.043412527 0.994594595
##
##   Parameters of node D (multinomial distribution)
##
## Conditional probability table:
##
```

```
## , , E = no
##
##      B
## D             no        yes
##   no  0.90017286 0.21373057
##   yes 0.09982714 0.78626943
##
## , , E = yes
##
##      B
## D             no        yes
##   no  0.27737226 0.14592275
##   yes 0.72262774 0.85407725
```

```
score(net,asia)
```

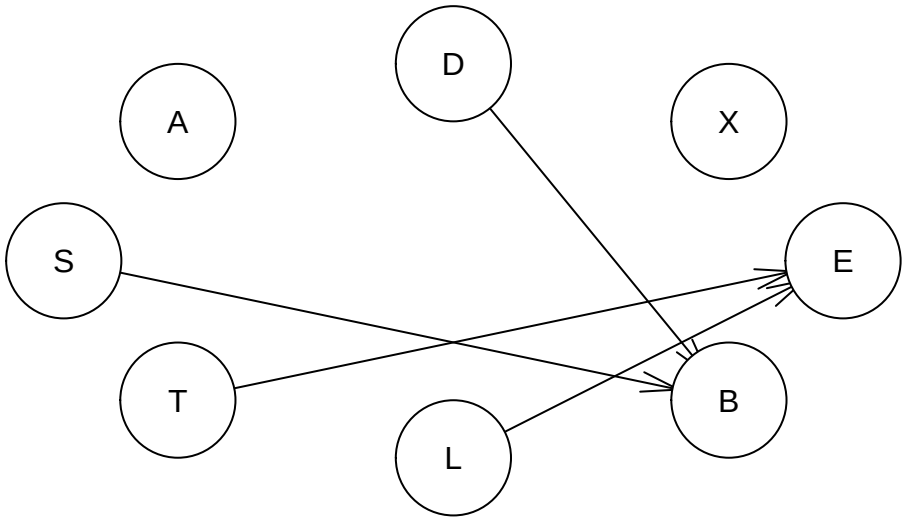
```
## [1] -11109.74
```

```
?score

netgs = gs(asia)
netgs_directed = cextend(netgs)
netgs_directed
```

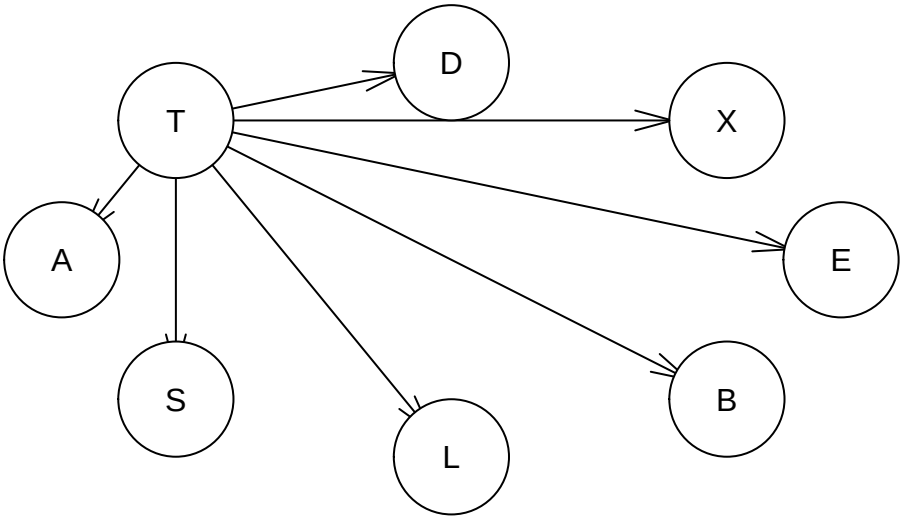
```
##
##   Bayesian network learned via Constraint-based methods
##
##   model:
##    [A][S][T][L][X][D][B|S:D][E|T:L]
##   nodes:                                8
##   arcs:                                 4
##     undirected arcs:                    0
##     directed arcs:                      4
##   average markov blanket size:          1.50
##   average neighbourhood size:           1.00
##   average branching factor:             0.50
##
##   learning algorithm:                   Grow-Shrink
##   conditional independence test:        Mutual Information (disc.)
##   alpha threshold:                      0.05
##   tests used in the learning procedure: 105
##   optimized:                            TRUE
```

```
plot(netgs)
```




```
plot(netgs_directed)


netnb = naive.bayes(asia,"T",names(asia)[c(1:2,4:8)])
plot(netnb)
```


D
T
X
A
E
S
B
L