```
library(bnlearn)
```

```
course.grades<-read.table("2020_bn_nb_data.txt",head=TRUE)
head(course.grades)
```

```
##    EC100 EC160 IT101 IT161 MA101 PH100 PH160 HS101 QP
## 1    BC    CC    BB    BC    CC    BC    AA    BB  y
## 2    CC    BC    BB    BB    CC    BC    AB    BB  y
## 3    AB    BB    AB    AB    BB    CC    BC    AB  y
## 4    BC    CC    BB    BB    BB    BB    BC    BB  y
## 5    BC    AB    CD    BC    BC    BC    BC    CD  y
## 6    DD    CC    DD    CD    CD    CC    BC    BC  n
```

```
course.grades<- lapply(course.grades,as.factor)
course.grades<- data.frame(course.grades)
course.grades.net<- hc(course.grades[,-9],score = 'k2')
plot(course.grades.net)
```
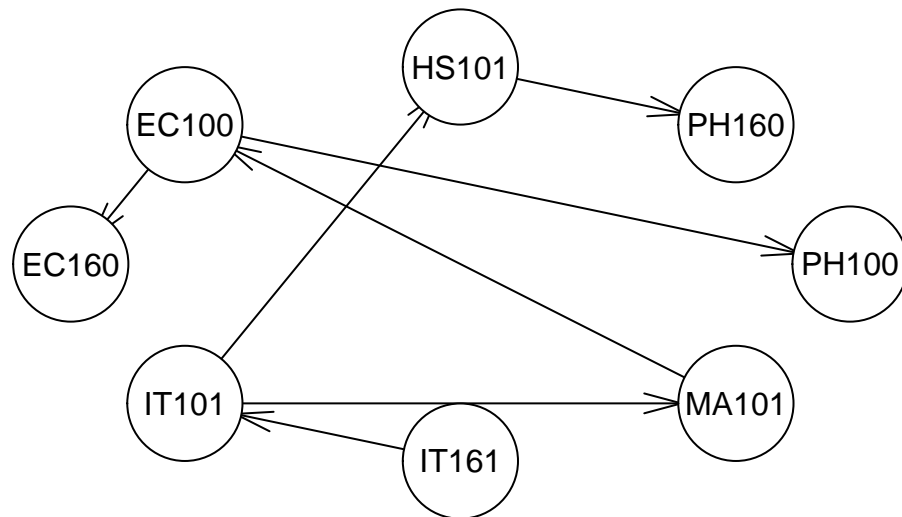


```
course.grades.net
```

```
##
##   Bayesian network learned via Score-based methods
##
```

1

```
##   model:
##    [IT161][IT101|IT161][MA101|IT101][HS101|IT101][EC100|MA101][PH160|HS101]
##    [EC160|EC100][PH100|EC100]
##   nodes:                               8
##   arcs:                                7
##     undirected arcs:                   0
##     directed arcs:                     7
##   average markov blanket size:         1.75
##   average neighbourhood size:          1.75
##   average branching factor:            0.88
##
##   learning algorithm:                  Hill-Climbing
##   score:                               Cooper & Herskovits' K2
##   tests used in the learning procedure: 105
##   optimized:                           TRUE
```

```
course.grades.fit <- bn.fit(course.grades.net,course.grades[,-9])
```

```
course.grades.fit$EC100
```

```
##
##   Parameters of node EC100 (multinomial distribution)
##
## Conditional probability table:
##
##      MA101
## EC100         AA         AB         BB         BC         CC         CD
##    AA 0.75000000 0.07692308 0.03846154 0.01851852 0.00000000 0.00000000
##    AB 0.00000000 0.46153846 0.25000000 0.05555556 0.00000000 0.00000000
##    BB 0.25000000 0.23076923 0.32692308 0.22222222 0.04081633 0.00000000
##    BC 0.00000000 0.15384615 0.28846154 0.27777778 0.32653061 0.00000000
##    CC 0.00000000 0.07692308 0.09615385 0.24074074 0.32653061 0.04166667
##    CD 0.00000000 0.00000000 0.00000000 0.12962963 0.26530612 0.33333333
##    DD 0.00000000 0.00000000 0.00000000 0.03703704 0.04081633 0.50000000
##    F  0.00000000 0.00000000 0.00000000 0.01851852 0.00000000 0.12500000
##      MA101
## EC100         DD          F
##    AA 0.00000000 0.00000000
##    AB 0.00000000 0.00000000
##    BB 0.00000000 0.00000000
##    BC 0.00000000 0.00000000
##    CC 0.00000000 0.00000000
##    CD 0.04761905 0.00000000
##    DD 0.19047619 0.00000000
##    F  0.76190476 1.00000000
```

```
course.grades.fit$EC160
```

```
##
##   Parameters of node EC160 (multinomial distribution)
##
## Conditional probability table:
```

```
## 
##        EC100
## EC160          AA          AB          BB          BC          CC          CD
##     AA 0.42857143 0.22727273 0.05714286 0.04166667 0.00000000 0.00000000
##     AB 0.42857143 0.22727273 0.08571429 0.04166667 0.08333333 0.00000000
##     BB 0.14285714 0.31818182 0.20000000 0.22916667 0.08333333 0.03448276
##     BC 0.00000000 0.22727273 0.42857143 0.43750000 0.36111111 0.17241379
##     CC 0.00000000 0.00000000 0.22857143 0.25000000 0.30555556 0.34482759
##     CD 0.00000000 0.00000000 0.00000000 0.00000000 0.11111111 0.27586207
##     DD 0.00000000 0.00000000 0.00000000 0.00000000 0.05555556 0.17241379
##     F  0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
##        EC100
## EC160          DD          F
##     AA 0.00000000 0.00000000
##     AB 0.00000000 0.00000000
##     BB 0.05000000 0.00000000
##     BC 0.00000000 0.00000000
##     CC 0.25000000 0.02857143
##     CD 0.55000000 0.40000000
##     DD 0.15000000 0.34285714
##     F  0.00000000 0.22857143
```

`course.grades.fit$IT101`

```
## 
##    Parameters of node IT101 (multinomial distribution)
## 
## Conditional probability table:
## 
##        IT161
## IT101          AA          AB          BB          BC          CC          CD
##     AA 0.35000000 0.08000000 0.05714286 0.02040816 0.00000000 0.00000000
##     AB 0.30000000 0.40000000 0.17142857 0.02040816 0.02380952 0.02857143
##     BB 0.25000000 0.40000000 0.31428571 0.14285714 0.00000000 0.02857143
##     BC 0.10000000 0.04000000 0.28571429 0.36734694 0.28571429 0.14285714
##     CC 0.00000000 0.08000000 0.14285714 0.32653061 0.33333333 0.11428571
##     CD 0.00000000 0.00000000 0.02857143 0.12244898 0.26190476 0.31428571
##     DD 0.00000000 0.00000000 0.00000000 0.00000000 0.04761905 0.34285714
##     F  0.00000000 0.00000000 0.00000000 0.00000000 0.04761905 0.02857143
##        IT161
## IT101          DD          F
##     AA 0.00000000 0.00000000
##     AB 0.00000000 0.00000000
##     BB 0.00000000 0.00000000
##     BC 0.04347826 0.00000000
##     CC 0.04347826 0.00000000
##     CD 0.21739130 0.33333333
##     DD 0.39130435 0.00000000
##     F  0.30434783 0.66666667
```

`course.grades.fit$IT161`

```
## 
```

```
##    Parameters of node IT161 (multinomial distribution)
##
## Conditional probability table:
##          AA         AB         BB         BC         CC         CD         DD
## 0.08620690 0.10775862 0.15086207 0.21120690 0.18103448 0.15086207 0.09913793
##           F
## 0.01293103
```

course.grades.fit$MA101

```
##
##    Parameters of node MA101 (multinomial distribution)
##
## Conditional probability table:
##
##      IT101
## MA101          AA         AB         BB         BC         CC         CD
##    AA 0.16666667 0.04000000 0.00000000 0.00000000 0.02380952 0.00000000
##    AB 0.25000000 0.20000000 0.02941176 0.08163265 0.00000000 0.00000000
##    BB 0.33333333 0.56000000 0.38235294 0.22448980 0.19047619 0.05714286
##    BC 0.16666667 0.16000000 0.29411765 0.36734694 0.23809524 0.22857143
##    CC 0.08333333 0.00000000 0.20588235 0.28571429 0.35714286 0.31428571
##    CD 0.00000000 0.04000000 0.08823529 0.02040816 0.16666667 0.11428571
##    DD 0.00000000 0.00000000 0.00000000 0.02040816 0.02380952 0.22857143
##    F  0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.05714286
##      IT101
## MA101          DD          F
##    AA 0.00000000 0.00000000
##    AB 0.00000000 0.00000000
##    BB 0.00000000 0.00000000
##    BC 0.08695652 0.00000000
##    CC 0.04347826 0.00000000
##    CD 0.30434783 0.08333333
##    DD 0.39130435 0.16666667
##    F  0.17391304 0.75000000
```

course.grades.fit$PH100

```
##
##    Parameters of node PH100 (multinomial distribution)
##
## Conditional probability table:
##
##      EC100
## PH100          AA         AB         BB         BC         CC         CD
##    AA 0.71428571 0.40909091 0.22857143 0.08333333 0.00000000 0.00000000
##    AB 0.14285714 0.31818182 0.20000000 0.18750000 0.05555556 0.00000000
##    BB 0.00000000 0.18181818 0.31428571 0.29166667 0.13888889 0.03448276
##    BC 0.14285714 0.04545455 0.14285714 0.22916667 0.33333333 0.13793103
##    CC 0.00000000 0.04545455 0.11428571 0.18750000 0.25000000 0.41379310
##    CD 0.00000000 0.00000000 0.00000000 0.02083333 0.19444444 0.31034483
##    DD 0.00000000 0.00000000 0.00000000 0.00000000 0.02777778 0.10344828
##    F  0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
```

```
##       EC100
## PH100         DD         F
##    AA 0.00000000 0.00000000
##    AB 0.00000000 0.00000000
##    BB 0.05000000 0.00000000
##    BC 0.00000000 0.00000000
##    CC 0.20000000 0.02857143
##    CD 0.45000000 0.11428571
##    DD 0.20000000 0.45714286
##    F  0.10000000 0.40000000
```

course.grades.fit$PH160

```
##
##    Parameters of node PH160 (multinomial distribution)
##
## Conditional probability table:
##
##       HS101
## PH160         AA         AB         BB         BC         CC         CD
##    AA 0.23809524 0.17647059 0.05000000 0.11111111 0.07692308 0.10000000
##    AB 0.23809524 0.11764706 0.15000000 0.13888889 0.07692308 0.10000000
##    BB 0.16666667 0.26470588 0.17500000 0.16666667 0.00000000 0.00000000
##    BC 0.21428571 0.32352941 0.45000000 0.22222222 0.50000000 0.30000000
##    CC 0.09523810 0.08823529 0.12500000 0.30555556 0.15384615 0.45000000
##    CD 0.04761905 0.02941176 0.02500000 0.05555556 0.11538462 0.05000000
##    DD 0.00000000 0.00000000 0.02500000 0.00000000 0.07692308 0.00000000
##    F  0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
##       HS101
## PH160         DD         F
##    AA 0.03448276 0.00000000
##    AB 0.10344828 0.00000000
##    BB 0.00000000 0.20000000
##    BC 0.10344828 0.00000000
##    CC 0.24137931 0.00000000
##    CD 0.37931034 0.00000000
##    DD 0.13793103 0.40000000
##    F  0.00000000 0.40000000
```
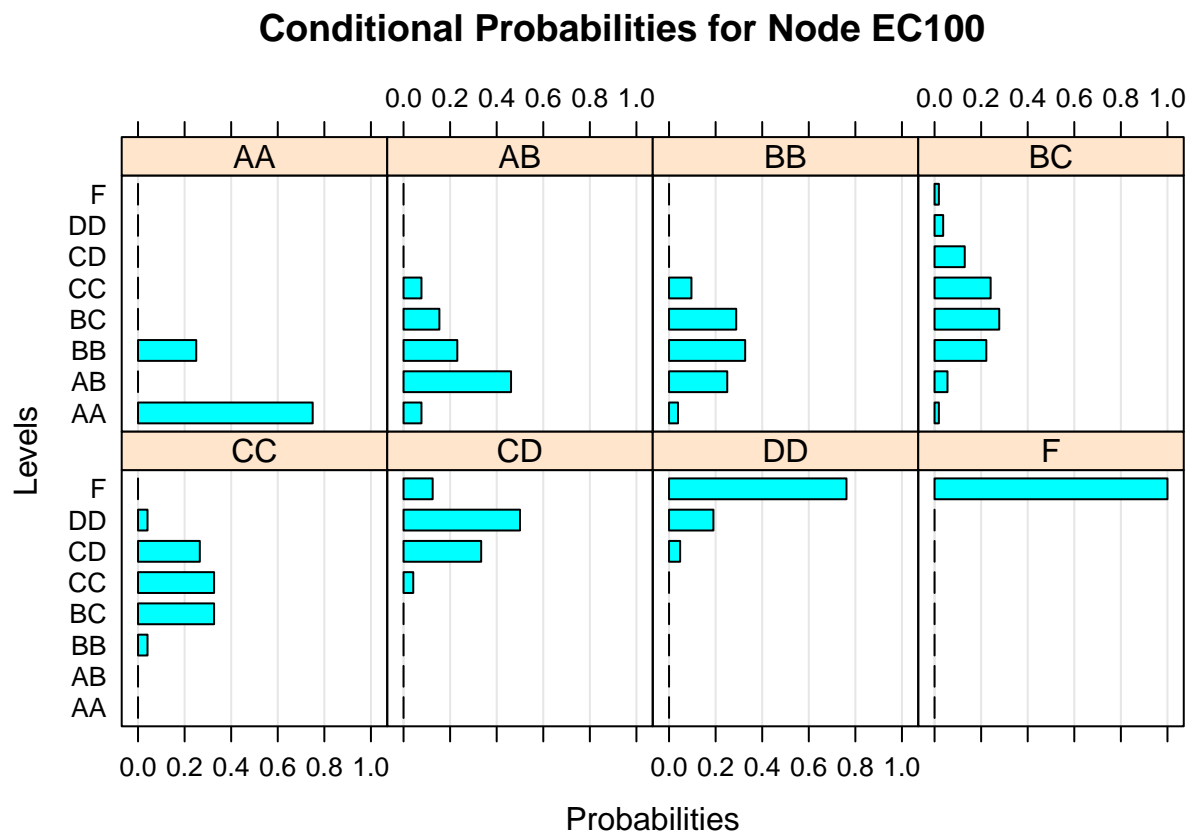
course.grades.fit$HS101

```
##
##    Parameters of node HS101 (multinomial distribution)
##
## Conditional probability table:
##
##       IT101
## HS101         AA         AB         BB         BC         CC         CD
##    AA 0.58333333 0.56000000 0.32352941 0.10204082 0.07142857 0.05714286
##    AB 0.33333333 0.24000000 0.11764706 0.22448980 0.14285714 0.08571429
##    BB 0.00000000 0.12000000 0.26470588 0.26530612 0.26190476 0.11428571
##    BC 0.08333333 0.08000000 0.08823529 0.24489796 0.23809524 0.20000000
##    CC 0.00000000 0.00000000 0.11764706 0.12244898 0.14285714 0.11428571
```

```
##      CD  0.00000000  0.00000000  0.05882353  0.02040816  0.14285714  0.20000000
##      DD  0.00000000  0.00000000  0.02941176  0.02040816  0.00000000  0.22857143
##      F   0.00000000  0.00000000  0.00000000  0.00000000  0.00000000  0.00000000
##        IT101
## HS101         DD          F
##      AA  0.00000000  0.00000000
##      AB  0.00000000  0.00000000
##      BB  0.00000000  0.00000000
##      BC  0.04347826  0.00000000
##      CC  0.26086957  0.00000000
##      CD  0.13043478  0.08333333
##      DD  0.52173913  0.58333333
##      F   0.04347826  0.33333333
```

```
bn.fit.barchart(course.grades.fit$EC100)
```

Conditional Probabilities for Node EC100


```
bn.fit.barchart(course.grades.fit$EC160)
```

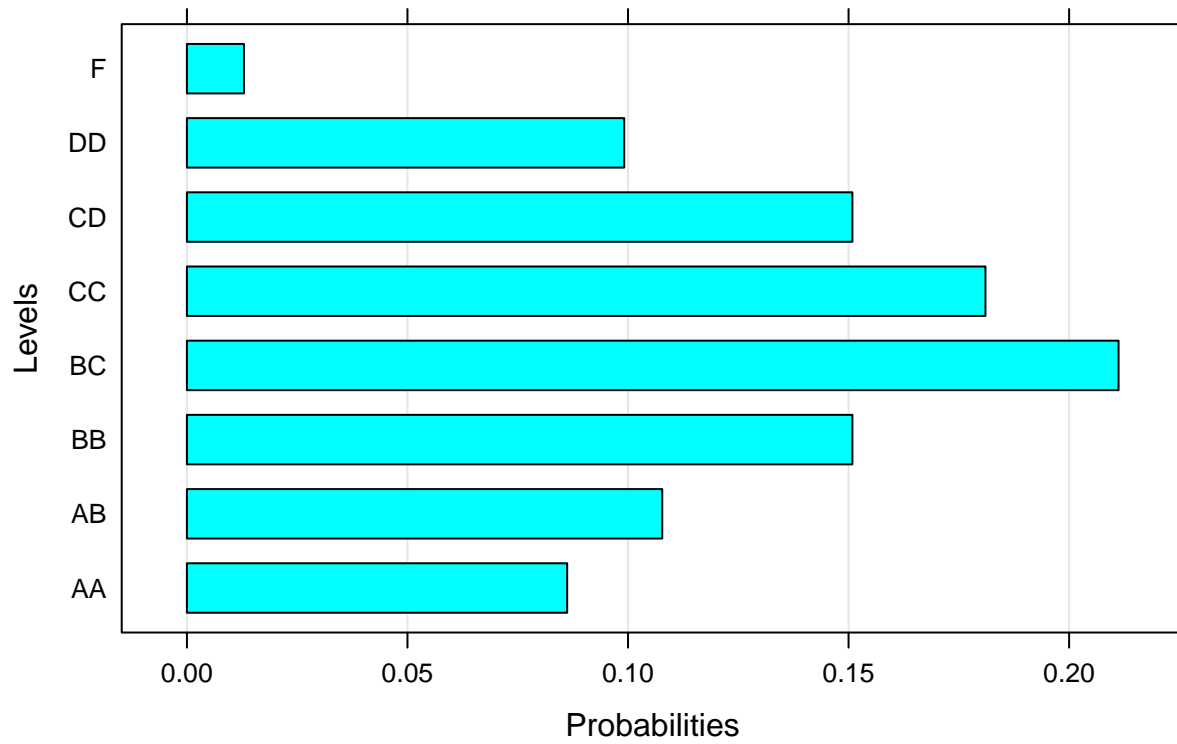# Conditional Probabilities for Node EC160
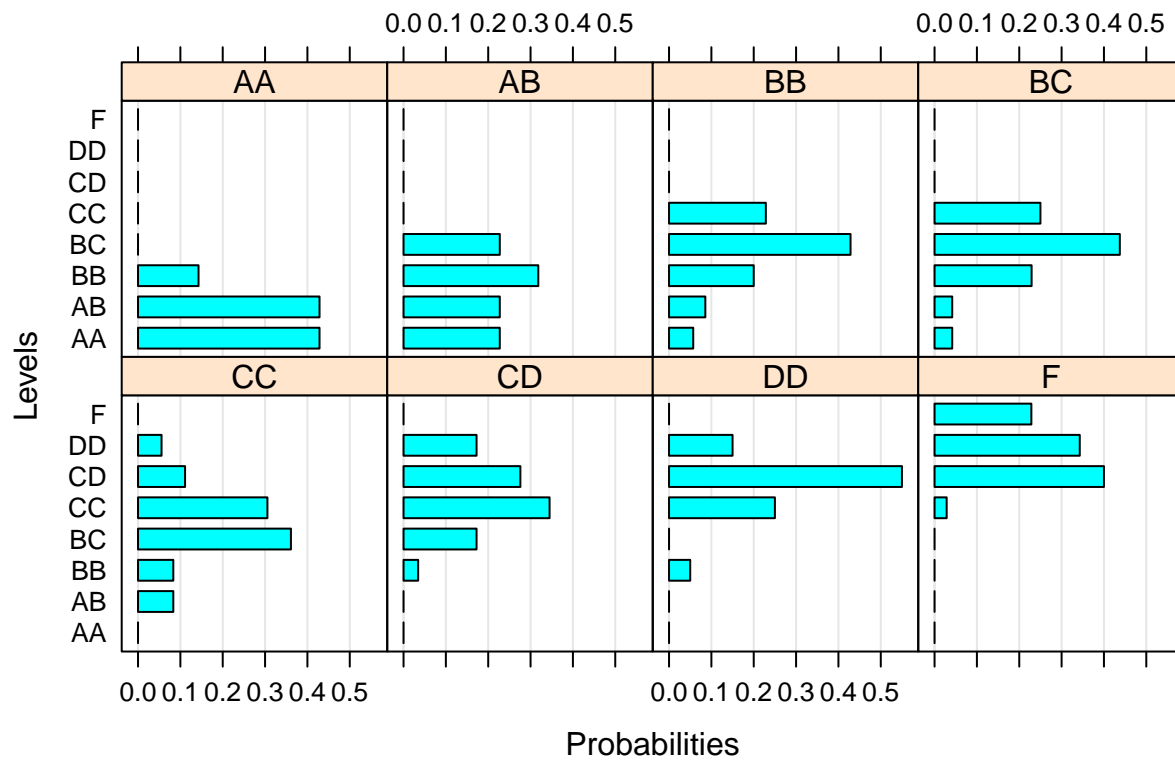

```
bn.fit.barchart(course.grades.fit$IT101)
```

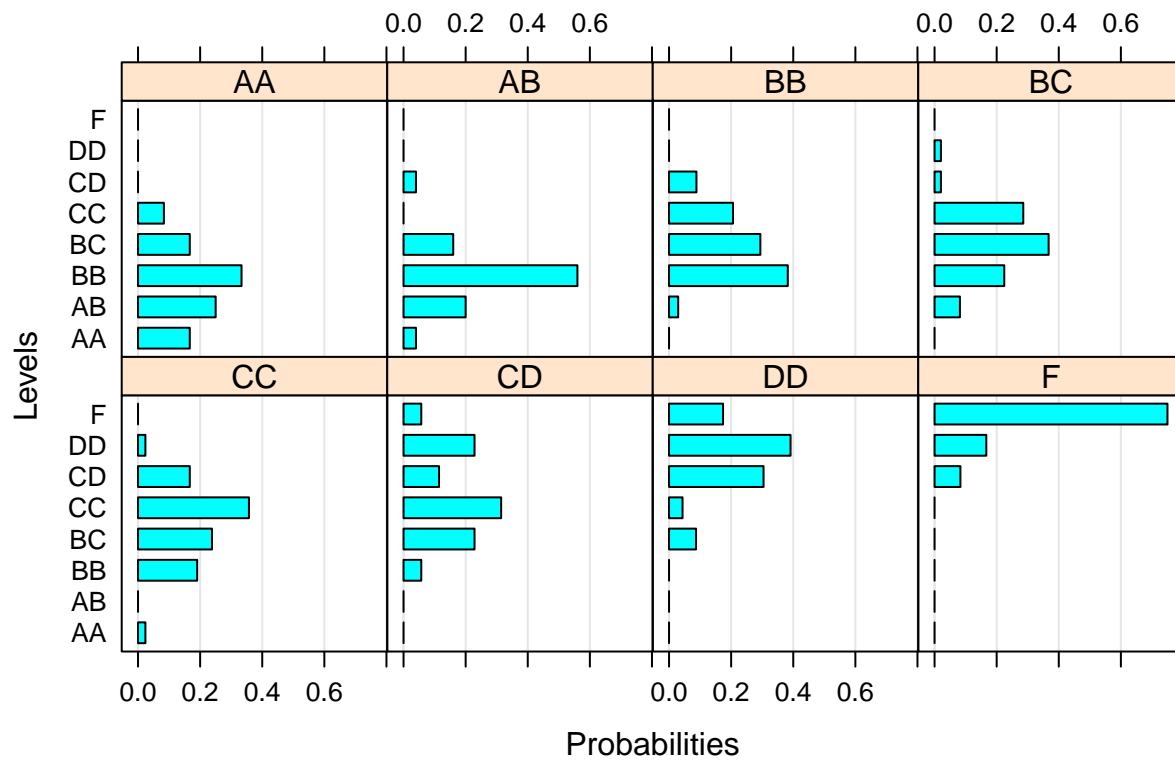## Conditional Probabilities for Node IT101
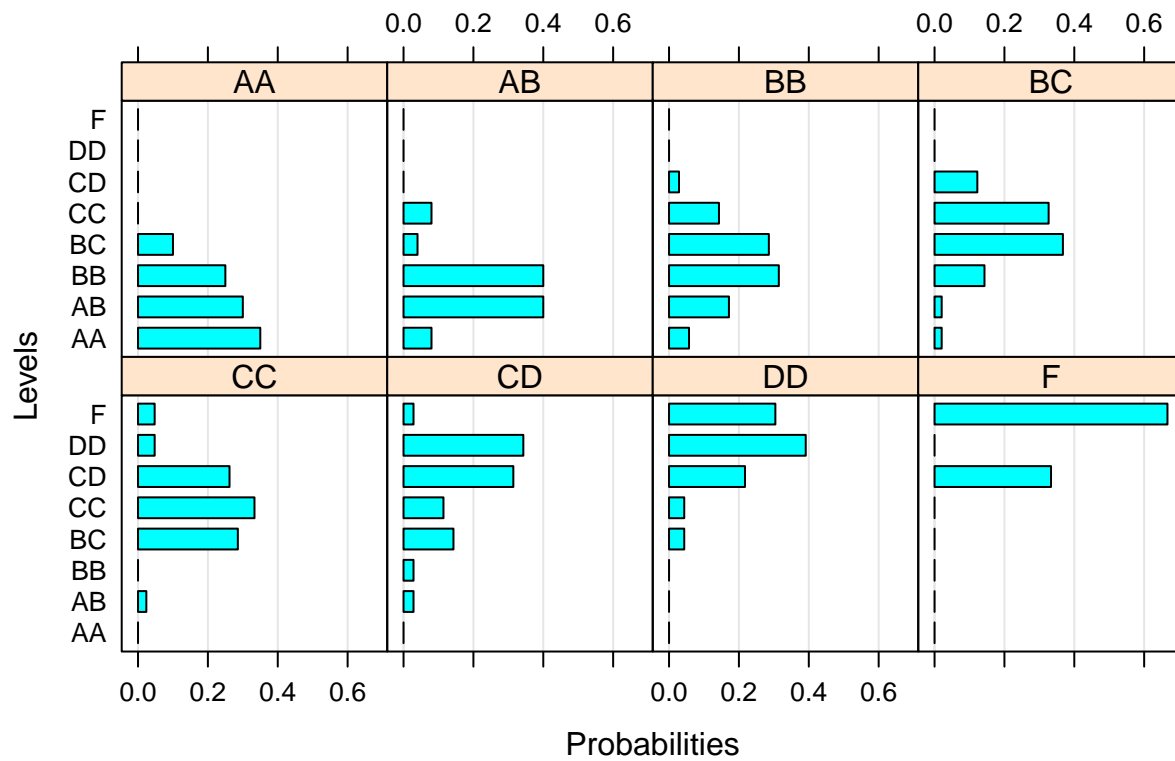

```
bn.fit.barchart(course.grades.fit$IT161)
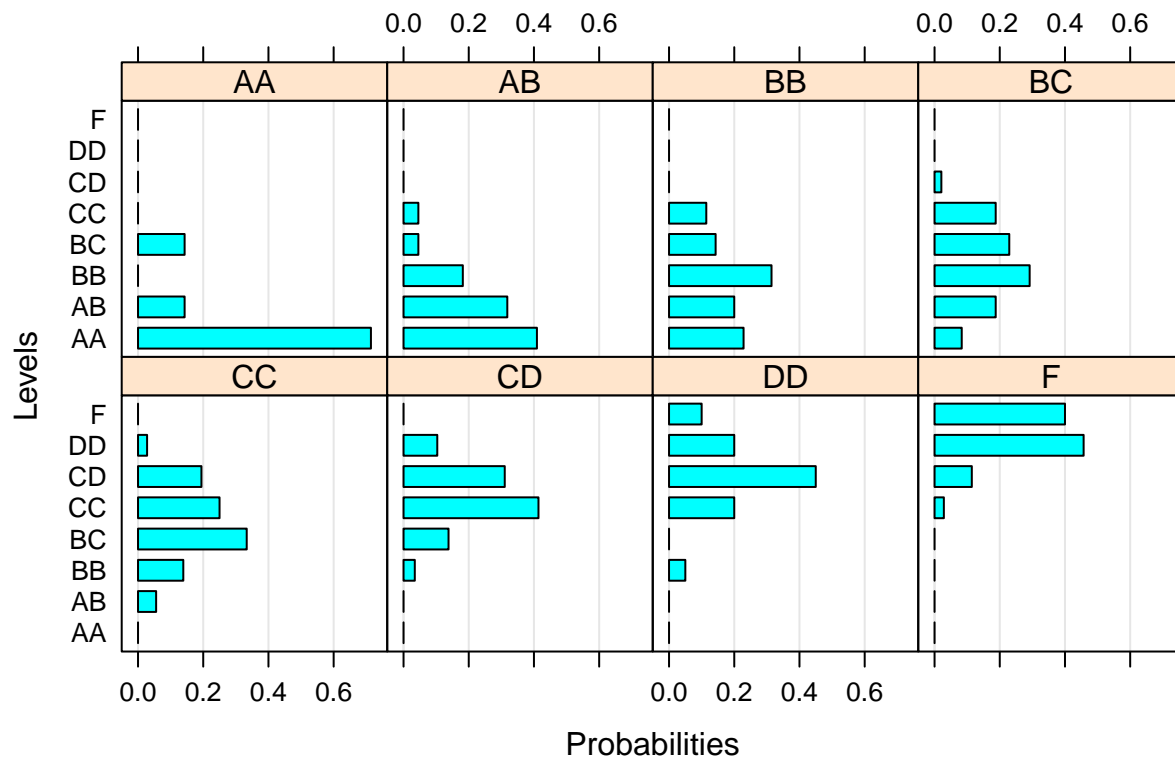```

## Conditional Probabilities for Node IT161



```
bn.fit.barchart(course.grades.fit$MA101)
```

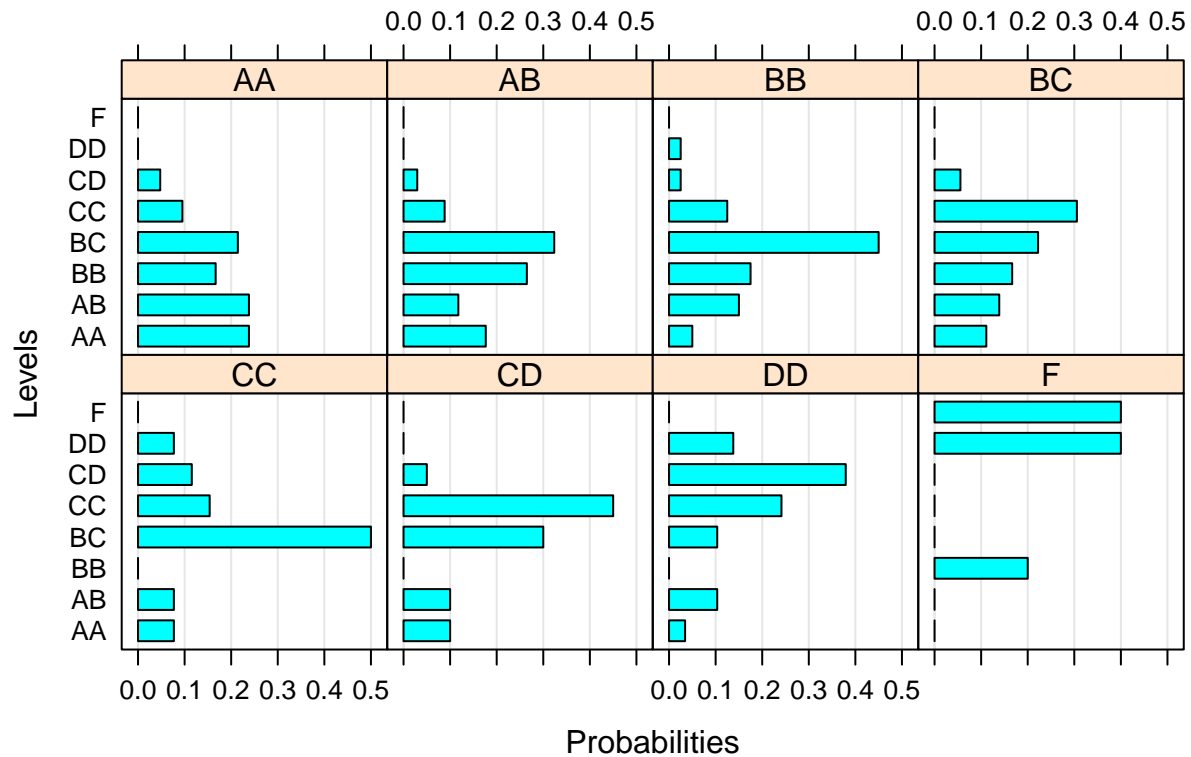# Conditional Probabilities for Node MA101


```
bn.fit.barchart(course.grades.fit$PH100)
```

Conditional Probabilities for Node PH100

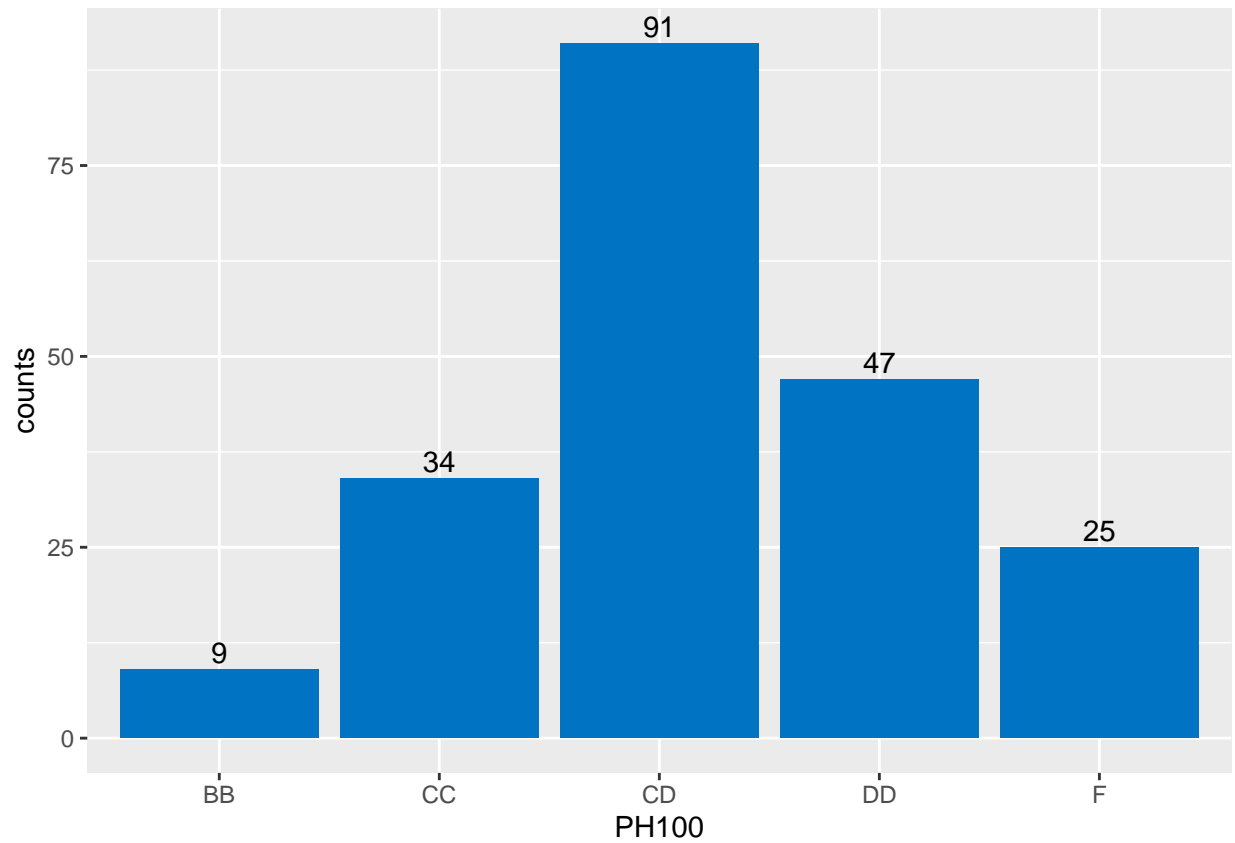
```
bn.fit.barchart(course.grades.fit$PH160)
```

Conditional Probabilities for Node PH160


```
course.grades.PH100 <- data.frame( cpdist(course.grades.fit, nodes = c("PH100"),evidence = ( (EC100=="D
```

```
library(dplyr)
df <- course.grades.PH100 %>%
  group_by(PH100) %>%
  summarise(counts = n())

library(ggplot2)
ggplot(df, aes(x = PH100, y = counts)) +
  geom_bar(fill = "#0073C2FF", stat = "identity") +
  geom_text(aes(label = counts), vjust = -0.3)
```
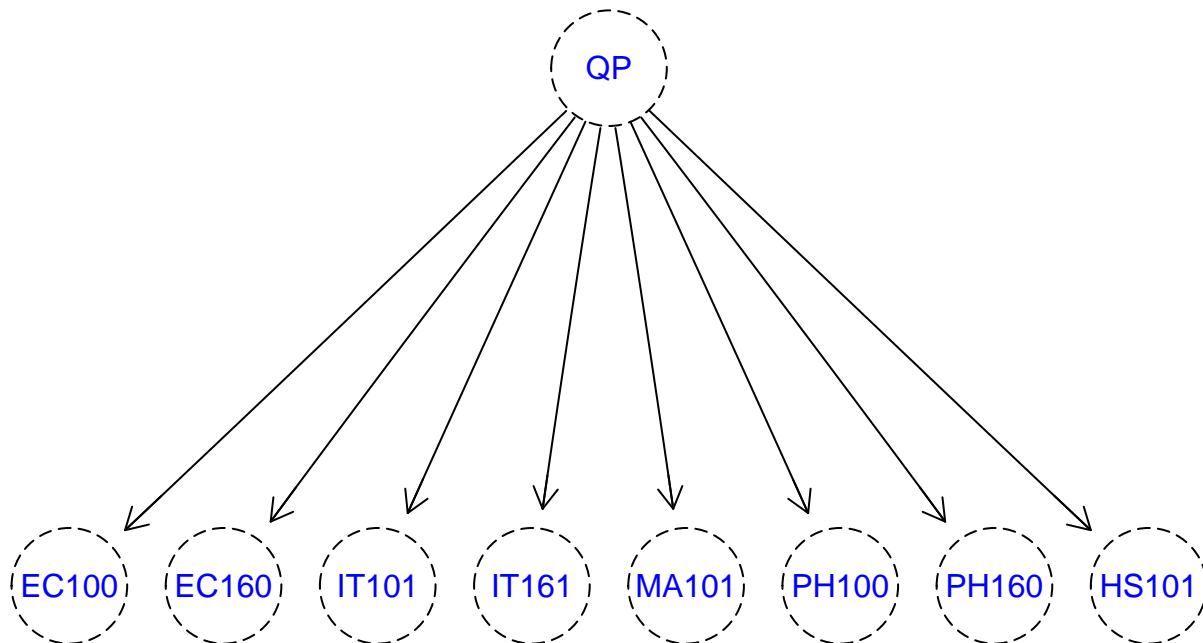
```
library(bnclassify)
```

```
set.seed(101)
sample <-  sample.int(n = nrow(course.grades), size = floor(.75*nrow(course.grades)), replace = F)
course.grades.train <-course.grades[sample,]
course.grades.test<- course.grades[-sample,]
```

```
nb.grades <- nb(class = "QP",dataset= course.grades.train)
```

```
plot(nb.grades)
```

```
nb.grades <- lp(nb.grades,course.grades.train,smooth=0)
nb.grades$.params
```

```
## $EC100
##       QP
## EC100          n          y
##     AA 0.00000000 0.04310345
##     AB 0.00000000 0.13793103
##     BB 0.00000000 0.23275862
##     BC 0.00000000 0.28448276
##     CC 0.06896552 0.18103448
##     CD 0.17241379 0.11206897
##     DD 0.31034483 0.00862069
##     F  0.44827586 0.00000000
##
## $EC160
##       QP
## EC160          n          y
##     AA 0.00000000 0.07758621
##     AB 0.00000000 0.08620690
##     BB 0.01724138 0.18103448
##     BC 0.01724138 0.38793103
##     CC 0.15517241 0.21551724
##     CD 0.46551724 0.03448276
##     DD 0.24137931 0.01724138
```

```
##     F   0.10344828 0.00000000
##
## $IT101
##       QP
## IT101          n          y
##    AA 0.00000000 0.06034483
##    AB 0.00000000 0.15517241
##    BB 0.05172414 0.22413793
##    BC 0.01724138 0.27586207
##    CC 0.13793103 0.20689655
##    CD 0.32758621 0.07758621
##    DD 0.32758621 0.00000000
##    F   0.13793103 0.00000000
##
## $IT161
##       QP
## IT161          n          y
##    AA 0.00000000 0.12068966
##    AB 0.01724138 0.13793103
##    BB 0.01724138 0.20689655
##    BC 0.01724138 0.30172414
##    CC 0.22413793 0.17241379
##    CD 0.43103448 0.05172414
##    DD 0.24137931 0.00862069
##    F   0.05172414 0.00000000
##
## $MA101
##       QP
## MA101          n          y
##    AA 0.00000000 0.02586207
##    AB 0.00000000 0.07758621
##    BB 0.00000000 0.31896552
##    BC 0.10344828 0.30172414
##    CC 0.15517241 0.25000000
##    CD 0.27586207 0.02586207
##    DD 0.29310345 0.00000000
##    F   0.17241379 0.00000000
##
## $PH100
##       QP
## PH100          n          y
##    AA 0.00000000 0.15517241
##    AB 0.00000000 0.16379310
##    BB 0.01724138 0.23275862
##    BC 0.03448276 0.21551724
##    CC 0.13793103 0.17241379
##    CD 0.24137931 0.06034483
##    DD 0.36206897 0.00000000
##    F   0.20689655 0.00000000
##
## $PH160
##       QP
## PH160          n          y
##    AA 0.05172414 0.17241379
```

```
##     AB 0.10344828 0.15517241
##     BB 0.01724138 0.15517241
##     BC 0.20689655 0.32758621
##     CC 0.31034483 0.14655172
##     CD 0.15517241 0.04310345
##     DD 0.12068966 0.00000000
##     F  0.03448276 0.00000000
##
## $HS101
##       QP
## HS101          n          y
##     AA 0.00000000 0.23275862
##     AB 0.00000000 0.23275862
##     BB 0.05172414 0.23275862
##     BC 0.13793103 0.14655172
##     CC 0.17241379 0.08620690
##     CD 0.22413793 0.05172414
##     DD 0.34482759 0.01724138
##     F  0.06896552 0.00000000
##
## $QP
## QP
##         n         y
## 0.3333333 0.6666667
```
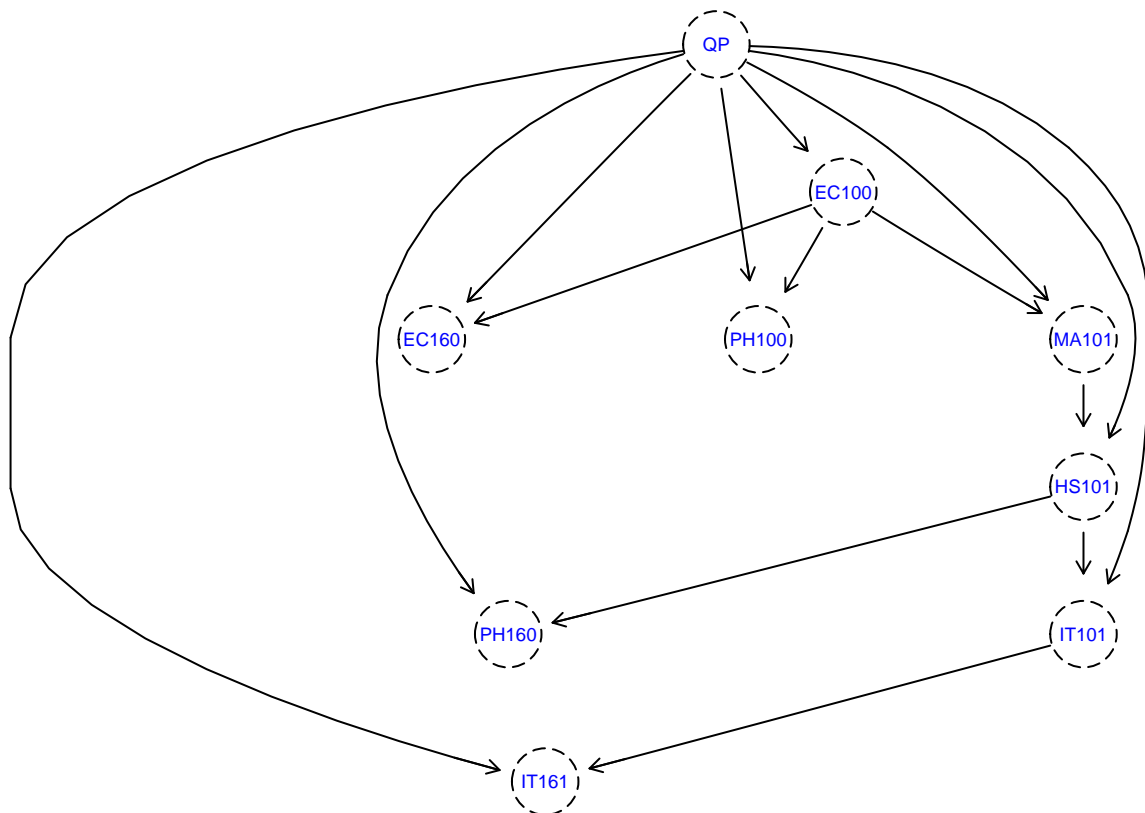
```
p <-predict(nb.grades,course.grades.test)
cm <- table(predicted_on_test_data = p , true = course.grades.test$QP)
cm
```

```
##                    true
## predicted_on_test_data  n  y
##                   n 13  1
##                   y  1 43
```

```
bnclassify:::accuracy(p, course.grades.test$QP)
```

```
## [1] 0.9655172
```

```
tn <- tan_cl('QP', course.grades.train)
tn <- lp(tn,course.grades.train, smooth = 1)
plot(tn)
```

```
p <- predict(tn, course.grades.test)
cm <- table(predicted_on_test_data = p , true = course.grades.test$QP)
cm
```

```
##                    true
## predicted_on_test_data  n  y
##                    n 12  0
##                    y  2 44
```

```
bnclassify:::accuracy(p, course.grades.test$QP)
```

```
## [1] 0.9655172
```