Practical 5: Bayesian Networks (cont)

I. Learning local causal structures from data

In this section, we use the PC-select function (PC-simple algorithm) from the *pcalg* package to learn the local network structure around one node from data. Please refer to the user manual of *pcalg* for more details https://cran.r-project.org/web/packages/pcalg/pcalg.pdf

Following example is performed with *pcalg* version 2.7.3. A different version can cause randomly generated graph to change, please install this version of *pcalg* or interpret your results accordingly to the graph.

1. Check pcalg version

   ```
   packageVersion("pcalg")
   ```

2. Generate and draw random DAG with 10 nodes

   ```
   p <- 10
   set.seed(10)
   myDAG <- randomDAG(p, prob = 0.25)
   if (require(Rgraphviz))
   { plot(myDAG, main = "randomDAG(10, prob = 0.25)") }
   ```

3. Generate 10000 samples of the DAG using standard normal error distribution

   ```
   n <- 10000
   d.mat <- rmvDAG(n, myDAG, errDist = "normal")
   ```

4. Learn the causal structure around node 10th, i.e. which of the first 9 variables "cause" the tenth variable?

   ```
   pcS <- pcSelect(d.mat[,10], d.mat[,-10], alpha=0.05)
   pcS
   ```

You can see from the result that variables 1,2,3,4 are the causes of the target (the variable 10).
By inspecting zMin, you can also see that the influence of variable 1 is the most evident from the data (The larger the number, the more consistent is the edge with the data.)

5. Apply PC algorithm to the **d.mat** dataset you just create (recall Practical 3). What are the causes of the node 10 based on PC algorithm?

II. Finding Parent and Children Set of a Node with HITON-PC

The function learn.nbr in bnlearn is implemented to learn the local causal structure around a target node. This function can be used with different local causal structure learning algorithms, including HITON-PC. In this example, we use the built-in asia data set from the bnlearn package to demonstrate the usage of HITON-PC in local causal discovery. The asia data set contains eight binary variables, D (dyspnoea), T (tuberculosis), L (lung cancer), B (bronchitis), A (visit to Asia), S (smoking), X (chest X-ray), and E (tuberculosis versus lung cancer/bronchitis).

We firstly use the function si.hiton.pc for learning the global causal structure from the data set. The following codes show how to learn the global causal structure from the asia data set.

```
library(bnlearn)
data(asia)
global.network = si.hiton.pc(asia, alpha=0.01)
plot(global.network)
```

We now assume that node E is the target variable, and we apply HITON-PC to learn the parents and children set of E.

```
HITON.PC.E = learn.nbr(asia, "E", method="si.hiton.pc", alpha=0.01)
```

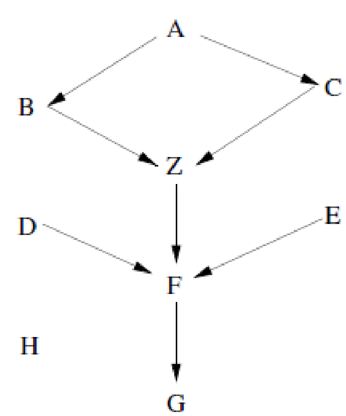
1. Is the parent and children set of E consistent with that in the global network?

In the bnlearn package, mutual information test is set as the default conditional independence test for binary variables. However, we can specify a different type of conditional independence test for HITON-PC, e.g. Chi-square (denoted as "x2" in bnlearn) as follows:

```
HITON.PC.E = learn.nbr(asia, "E", method="si.hiton.pc"
                       , test="x2", alpha=0.01)
HITON.PC.E
```

III. Finding Markov Blanket of a Node

1. Given a Bayesian network as in the following figure, what is the Markov Blanket (MB) of node Z? (Hint: Let's google it)



2. Download "Example21" dataset. It has the same dependence relationships as the above network.

3. Learn the MB of Z from data using the IAMB algorithm from bnlearn.

```
library(bnlearn)
```

Assuming the Example21.csv has been placed in the working directory, you can read it using read.csv

```
data=read.csv("Example21.csv", header=TRUE, sep=",")
data[1:5,]
```

bnlearn requires numeric or factor data types. Convert data of the nine variables (nine columns) in the data set to factor data types.

```
nvar <- ncol(data)
for(i in 1:nvar){data[,i] = as.factor(data[,i]) }
```

learn the markov blanket

```
MB.Z=learn.mb(data, "Z", method="iamb", alpha=0.01)
MB.Z
```

IV. Estimating causal effect of a variable on another with IDA

Given a Bayesian network, we can estimate the causal effect that a node has on another. In this example, we re-use the dataset **d.mat** from Section II, and apply *ida* and *idaFast* functions from pcalg package to estimate the causal effects.

1. Learn the causal structure from data.

```
suffStat <- list(C = cor(d.mat), n = nrow(d.mat))
pc.fit <- pc(suffStat, indepTest = gaussCItest
             , p=ncol(d.mat), alpha = 0.01)
plot(pc.fit@graph)
```

2. Estimate the causal effect of node 2 on node 10.

```
ida(2, 10, cov(d.mat), pc.fit@graph)
```

3. Estimate the causal effect of node 4 on nodes 10 and 6.

```
ida(4, c(10,6), cov(d.mat), pc.fit@graph)
idaFast(4, c(10,6), cov(d.mat), pc.fit@graph)
```

If the equivalence class contains **k** DAGs, this will yield **k** estimated total causal effects.

4. Estimate the causal effect of node 5 on node 7.

```
ida(5,7, cov(d.mat), pc.fit@graph)
idaFast(5,7, cov(d.mat), pc.fit@graph)
```

5. Calculate the causal effect of node 3 on nodes 6, 10.
6. Calculate the causal effect of node 2 on node 10 and node 8 on nodes 7, 9.

V. Summary of Bayesian Networks

1. Generate and draw random DAG with 10 nodes (set seed to 50 and prob to 0.2)

   ```
   p <- 10
   set.seed(50)
   myDAG <- randomDAG(p, prob = 0.2)
   if (require(Rgraphviz))
   { plot(myDAG, main = "randomDAG(10, prob = 0.2)") }
   ```

2. Generate 10000 samples of the DAG using standard normal error distribution

   ```
   n <- 10000
   mydataset <- rmvDAG(n, myDAG, errDist = "normal")
   ```

3. Use PC algorithm to learn the causal structure of the dataset.

4. Estimate the causal effects of node 2 on nodes 5,9.

5. Find the parent and children set of node 7 using pcSelect (the PC-Simple algorithm)

Note: bnlearn requires the input dataset in **dataframe** format. Use dataset=data.frame(mydataset) to convert the dataset to **dataframe** format. Also, check variables names returned after use data.frame

6. Find the parent and children set of node 7 using HITON-PC

7. Learn the Markov blanket of node 7 from data