Assignment 9

Hui Jeong (HJ) Jung, Gudmundur Björgvin Magnusson, Jieran Sun

2023-04-29

Problem 23 d-separation

(i) According to the definition of d-separation, a path p between two nodes is d-separated by a set of nodes Z if and only if 1) p contains a chain of nodes s -> m -> t or s <- m <- t or a fork s <- m -> t such that the middle node m is in Z, or

2) p contains a collider s -> m <- t such that the middle fork m is not in Z and such that no descendant of m is in Z.

Thus, G follows the first definition of d-separation, and thus d-separated from A.

B is not d-separated. According to the graph, A -> D <- B, and this case A and B are d-separated if D is not in "Z" as mentioned above, but since it is, B is not separated. And accordingly F and E are also not d-separated.

(ii)

(a) False. $A \perp\!\!\!\perp B$ if we do not observe D, but given D $A \not\!\perp\!\!\!\perp B \mid D$. Since C is dependent on A, B is not conditionally dependent of C given D.
(b) False, G and E have a common ancestor B, and knowing D doesn't change the dependence that E and G have because E does not descend from D.
(c) False, $C \perp\!\!\!\perp D|A$, but $D \not\!\perp\!\!\!\perp F$ as they're both children of B. Thus given just A, C cannot be said to be conditionally independent of F.
(d) True, The markov blanket of C includes nodes A, D, G, and B. As mentioned in (c), $C \perp\!\!\!\perp F|A,B$. As A and B are both part of the blanket, C and F are independent given the blanket, and thus E will also be independent from C.

Problem 24 Testing for marignal correlation

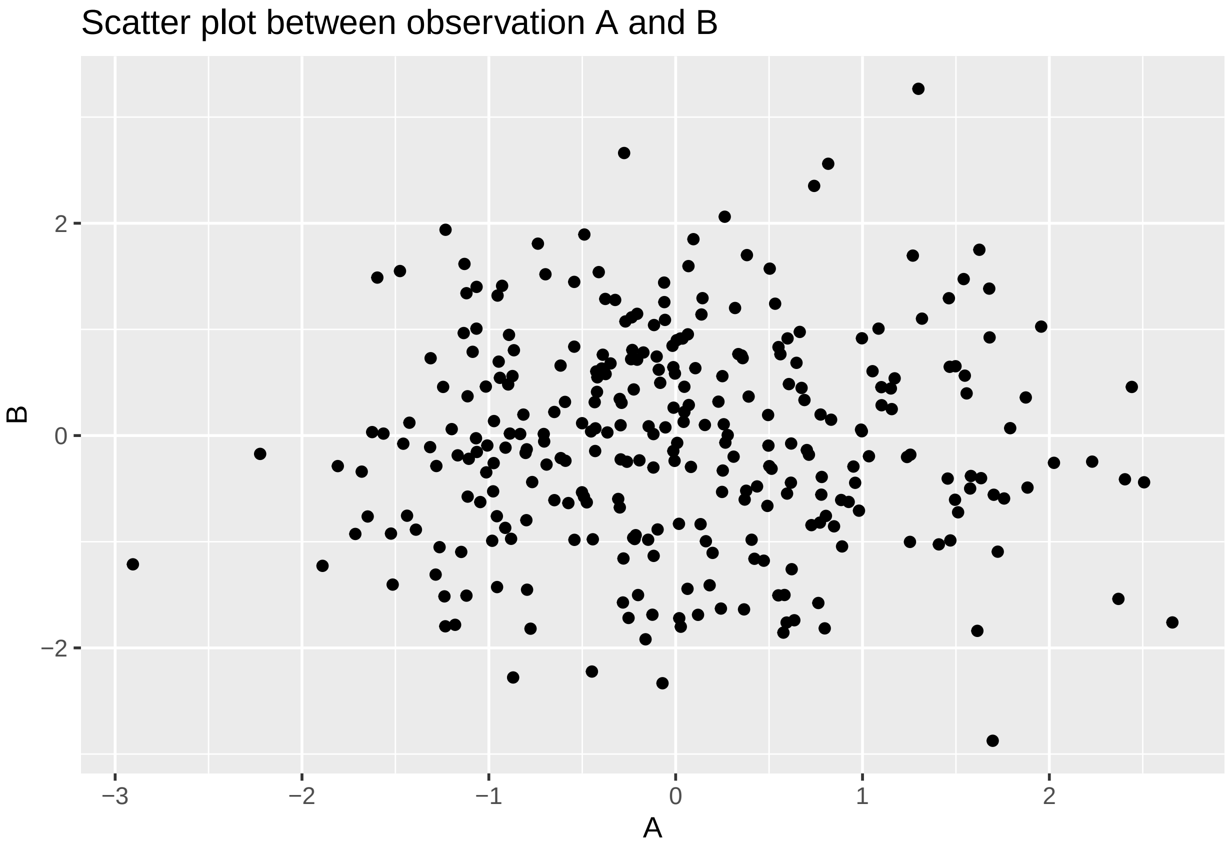
First we load the data and examine the data structure.

```r
nvmData <- readRDS("MVN_DAG.rds")
head(nvmData)
```

```
##              A           B          C           D           E          F
## 1 -0.294720447  0.09739665 -0.6693398 -1.83015111  0.41091561  0.8663284
## 2 -0.005767173 -0.23838695  1.3541249 -0.63615467 -0.62030667  0.5347883
## 3  2.404653389 -0.41182796 -0.6750873  2.40311737  0.25173361 -1.3942278
## 4  0.763593461 -1.57721805 -2.9092631 -0.52703829 -0.81604685  1.4960640
## 5 -0.799009249 -0.79727610 -2.5411415  0.18356375  0.07764537 -1.2424275
## 6 -1.147657009 -1.09623678 -0.8384432 -0.03142255 -0.23194797 -1.0266878
```

Plot the scatter plot for the observation A and B

```
ggplot(nvmData, aes(x = A, y = B)) + geom_point() +
  ggtitle("Scatter plot between observation A and B")
```



It can be inferred from the scatter plot that the correlation between observation A and B are merely nonexistent, A and B are not correlated with each other. This observation from the scatter plot also agree with what we have observed on the DAG graph where A and B are independent of each other.

Now we test the independence hypothesis using `cor.test()`

```
cor.test(nvmData$A, nvmData$B)
```

```
##
##  Pearson's product-moment correlation
##
## data:  nvmData$A and nvmData$B
## t = 0.20194, df = 298, p-value = 0.8401
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1016784  0.1247727
## sample estimates:
##        cor
## 0.01169715
```

Since the p value is way larger than 0.05, we accept the null hypothesis and confirm that A and B are independent from each other.

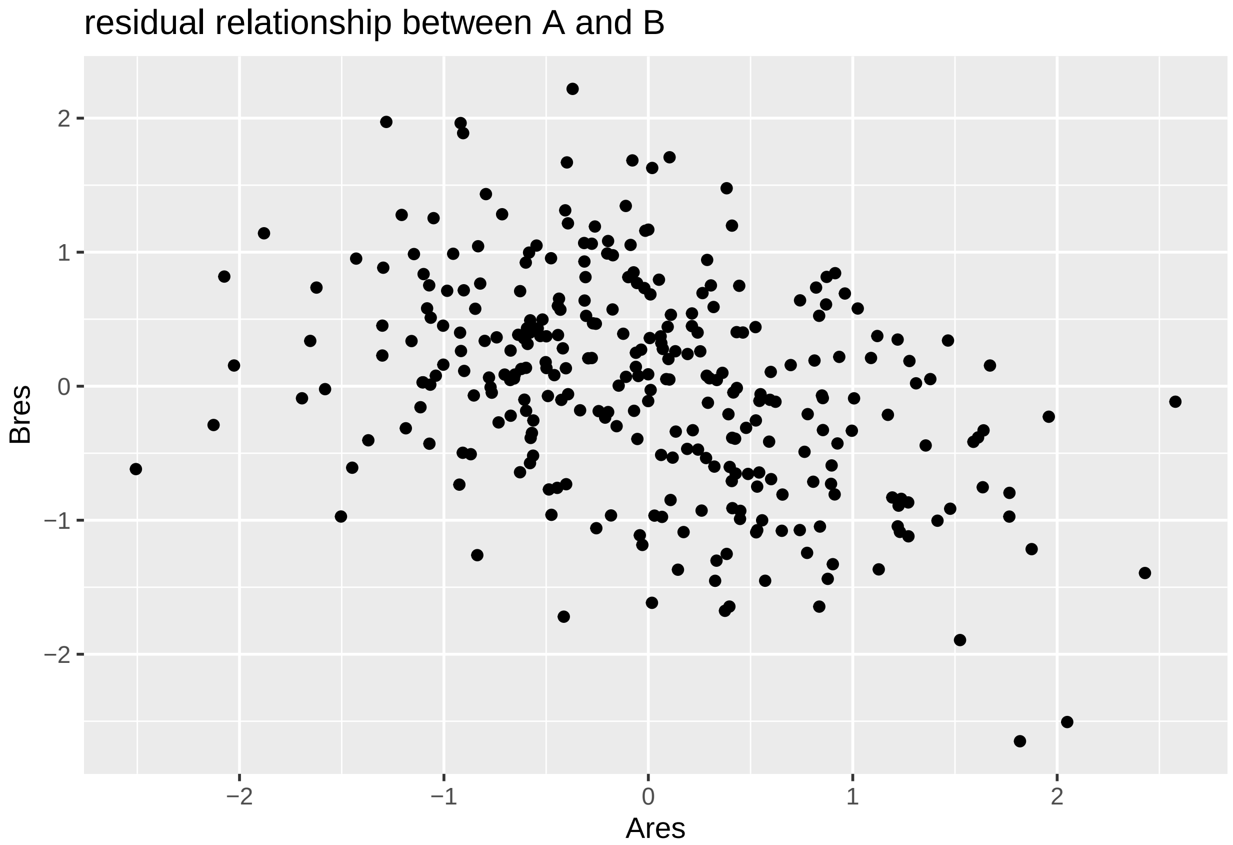
25 partial correlation

First, we find the residual of A and B by fitting them into a linear regression on C

```
Ares <- residuals(lm(A ~ C, data = nvmData))
Bres <- residuals(lm(B ~ C, data = nvmData))
```

Then similarly we plot the residual relationship

```
ggplot() + geom_point(aes(x = Ares, y = Bres)) +
  ggtitle("residual relationship between A and B")
```



Comparing to the results shown in question 24, this plots show us a stronger correlation between the residual of A and B observations, indicating that A and B are not conditionally independent given C.

Now let's test the null hypothesis that A and B are conditionally independent given C.

```
cor.test(Ares, Bres)
```

```
##
##  Pearson's product-moment correlation
##
## data:  Ares and Bres
## t = -7.5173, df = 298, p-value = 6.6e-13
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
```

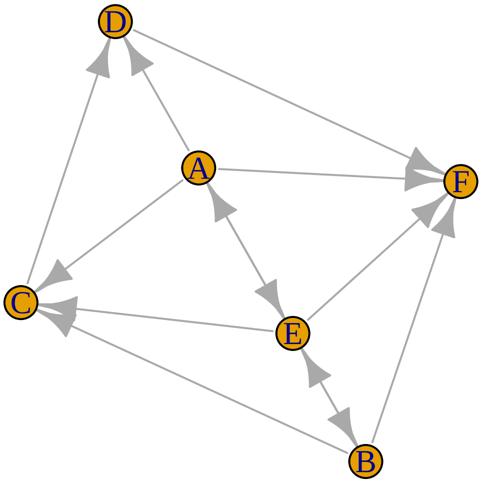
```
##  -0.4903245 -0.2995546
## sample estimates:
##        cor
## -0.3992521
```

Given that p value is way smaller than 0.05, we reject the null hypothesis and state that A and B are not conditionally independent given C. This result also agree with what we observed from Figure 2.

26 run PC algorithm

Given the instruction, we run the pc function and plot the resulting DAG

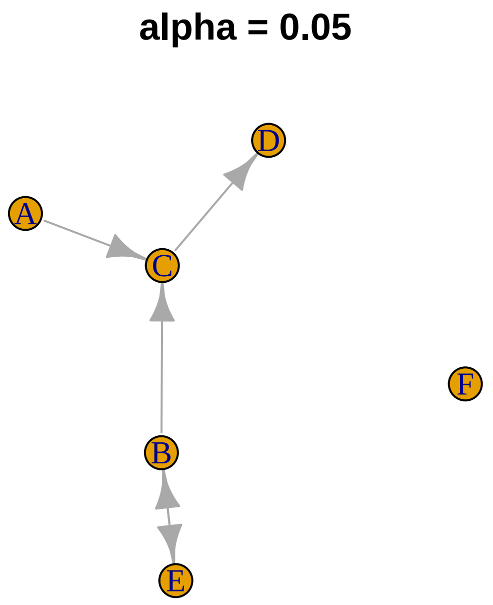
```
statList <- list(C = cor(nvmData), n = nrow(nvmData))

pcResults <- pc(suffStat = statList,
                indepTest = gaussCItest,
                alpha = 0.9,
                labels = colnames(nvmData))

iplotPC(pcResults)
```



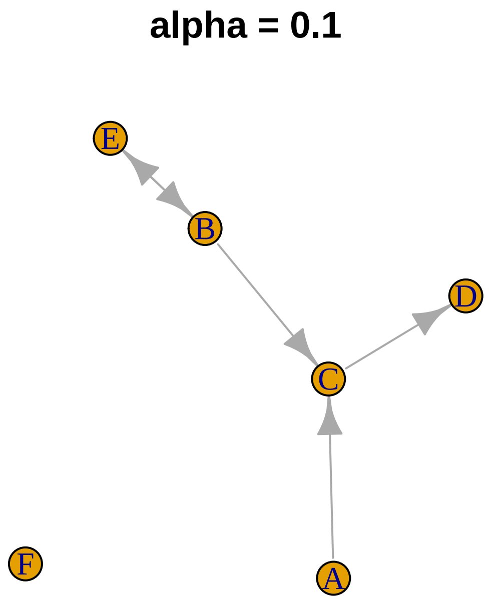
## 27 partition MCMC algorithm

First we set up the score and run the MCMC algorithm

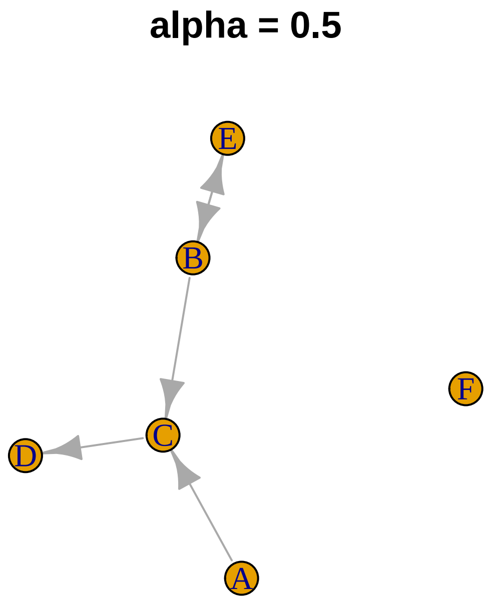
```r
# initialize the parameters
score <- scoreparameters("bge", nvmData)

# run iterative MCMC algorithm
maxBNalpha05 <- learnBN(score, algorithm = "orderIter", alpha = 0.05)
# plot the maxBN DAG results
plot.igraph(graph_from_adjacency_matrix(maxBNalpha05$CPDAG, mode= "directed"),
            main = "alpha = 0.05")
```



```r
# set alpha to 0.1
maxBNalpha10 <- learnBN(score, algorithm = "orderIter", alpha = 0.1)
# plot the maxBN DAG results
plot.igraph(graph_from_adjacency_matrix(maxBNalpha10$CPDAG, mode= "directed"),
            main = "alpha = 0.1")
```



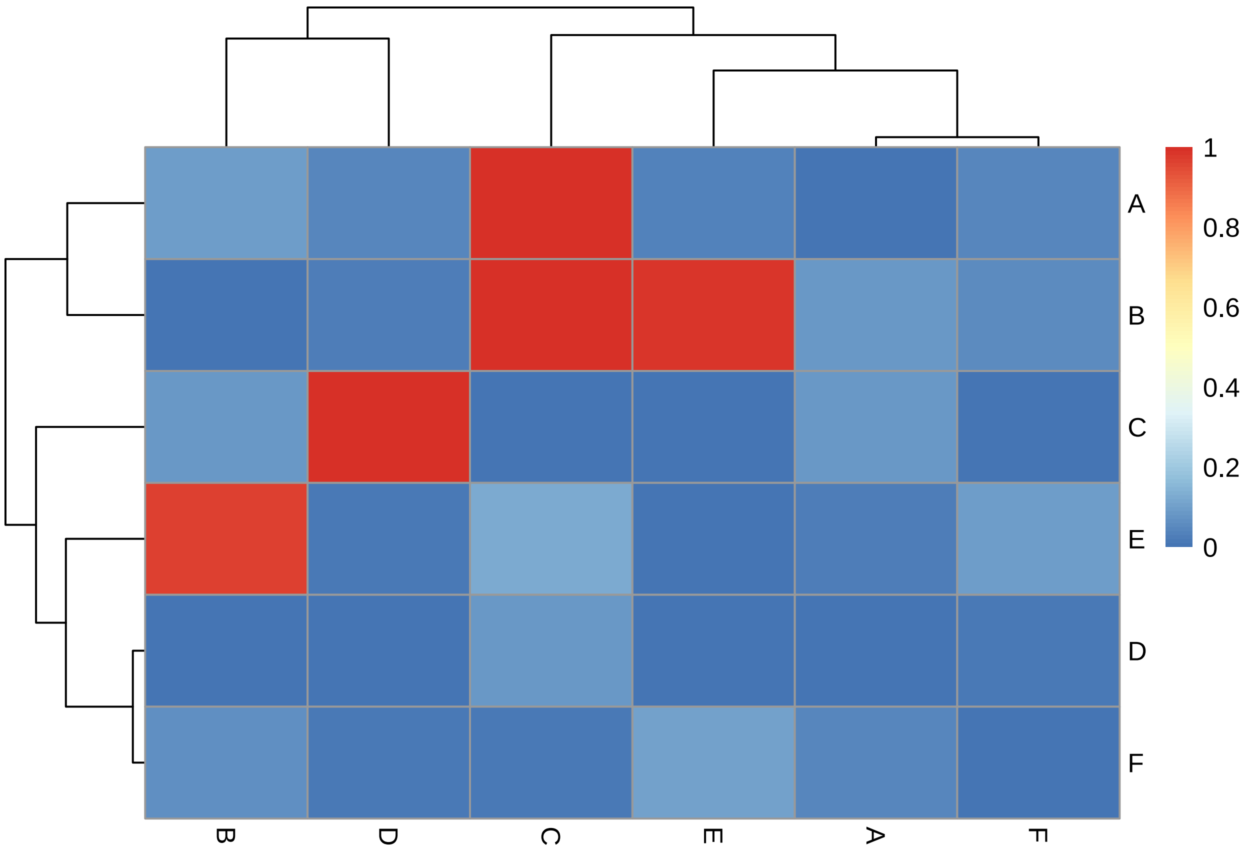
```
# set alpha to 0.2
maxBNalpha50 <- learnBN(score, algorithm = "orderIter", alpha = 0.5)
# plot the maxBN DAG results
plot.igraph(graph_from_adjacency_matrix(maxBNalpha50$CPDAG, mode= "directed"),
            main = "alpha = 0.5")
```



From the list of graphs above we can see the hyperparameter alpha does not affect the final DAG connectivity and shape.

Then we run the MCMC partition algorithm, extract the marginal probability for the edges and plot them into a heatmap

```
partitionSample <- sampleBN(score,
                            algorithm = "partition",
                            startspace = maxBNalpha05$endspace)

# compute the marginal posterior probabilities
edgesPosterior <- edgep(partitionSample, pdag = TRUE)

# plot the heatmap
pheatmap(edgesPosterior)
```


1
0.8
0.6
0.4
0.2
0
A
B
C
E
D
F
B
D
C
E
A
F