# Project 9

## Richard Affolter

### 01 Mai, 2021

Difficulty of this project: **2**

## Problem 23: d-separation

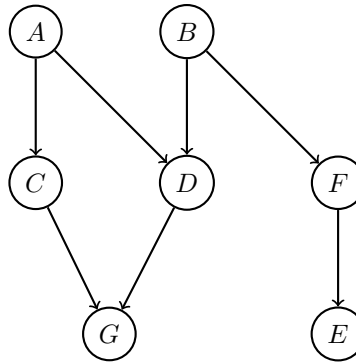For the following Bayesian network



Figure 1

**(i) Write down all the variables that are d-separated from $A$ given $\{C, D\}$.**

- $A \perp G \mid \{C, D\}$

**(ii) Which of the following statements are true? If false, please explain why**

- (a) B is conditionally independent of C given D
  **FALSE**, B is not d-separated from C given D. We have an active path C-A-D-B.
  Also we have
  $$P(B, C|D) = \frac{P(B, C, D)}{P(D)}$$
  $$= \frac{\sum_A P(A) \cdot P(B) \cdot P(C|A) \cdot P(D|A, B)}{\sum_A \sum_B P(D|A, B)}$$

  in general this does not factorize to the product $P(B|D) \cdot P(C|D)$.

- (b) G is conditionally independent of E given D
  **FALSE** we have an active path G-C-A-D-B-F-E

- (c) C is conditionally independent of F given A.
  **TRUE** C is d-separated from F given A.

- (d) C is conditionally independent of E given its Markov blanket (of C).
  **TRUE** we have $MB(C) = \{A, D, G\}$, adn C is d-sepatated from E given $\{A, D, G\}$.

## Problem 24: Testing for marginal correlation

The covariance between two random variables $X$ and $Y$ captures their linear relationship, and is defined as $\mathrm{Cov}(X, Y) := \mathbb{E}\left[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])\right]$. Their correlation $\rho_{X,Y} := \frac{\mathrm{Cov}(X,Y)}{\sqrt{\mathrm{Var} X \mathrm{Var} Y}}$ is merely their covariance scaled by the product of their respective standard deviations. Note that for a multivariate normal distribution, uncorrelated variables are independent. However, it is important to keep in mind that this implication does not hold in general
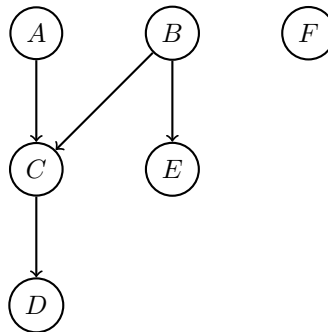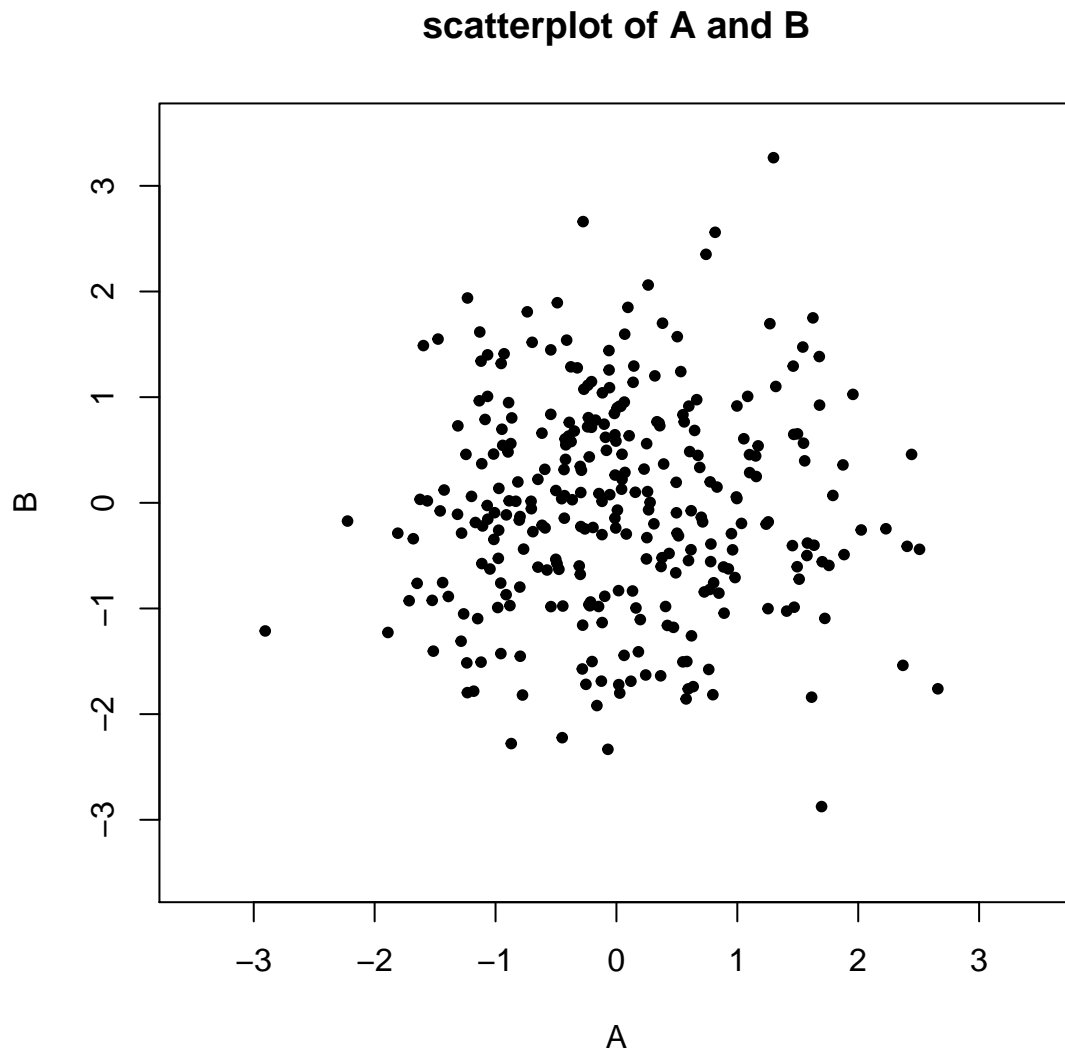


Figure 2

Using the data from `MVN_DAG.rds`, display the observations of $A$ and $B$ in a scatterplot. **What does the plot suggest about their (marginal) correlation? Does it agree with Figure 2? Use the function `cor.test()` to test the null hypothesis of no correlation between $A$ and $B$. What is your conclusion?**

```
data = readRDS("MVN_DAG.rds")
#par(pty="s")
plot(data$A, data$B, xlab = "A", ylab = "B", main = "scatterplot of A and B",
     xlim = c(-3.5,3.5), ylim = c(-3.5,3.5), pch=20)
```

## scatterplot of A and B



The scatterplot suggests that there is no marginal covariance and therefore also no marginal correlation between A and B. This is also in agreement with Figure 2, since there is also marginal independence between A and B.

We can test this null-hypothesis (no correlation between A and B) with `cor.test()`:

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

```
##
##  Pearson's product-moment correlation
##
## data:  data$A and data$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
```

We can see that our p-value is over the significance level $\alpha = 0.05$. Therefore we cannot reject the null-hypothesis.

So my conclusion is that A and B are (marginally) independent.

## Problem 25: Testing for partial correlation

The partial correlation between two random variables $X$ and $Y$ given a random variable $Z$ is

$$\rho_{X,Y|Z} = \frac{\rho_{X,Y} - \rho_{X,Z}\rho_{Y,Z}}{\sqrt{(1 - \rho_{X,Z}^2)(1 - \rho_{Y,Z}^2)}}$$

Alternatively, the partial correlation $\rho_{X,Y|Z}$ equals the correlation between residuals from the linear regressions of $X$ on $Z$, and $Y$ on $Z$, respectively. We will now compute the partial correlation $\rho_{A,B|C}$ to assess the association between $A$ and $B$ given $C$ as follows:

- Linearly regress $A$ on $C$ (that is, with $A$ as the response variable and $C$ as the explanatory variable). Compute and store the residuals.

```
lmFit = lm(A ~ C, data = data)
res_AC= residuals(lmFit)
```

- Linearly regress $B$ on $C$. Compute and store the residuals.

```
lmFit = lm(B ~ C, data = data)
res_BC= residuals(lmFit)
```

- Plot the residuals of $A$ (regressed on $C$) against the residuals of $B$ (regressed on $C$). What do you see?

```
plot(res_AC, res_BC, xlab = "residuals A~C", ylab = "residuals B~C", main = "residuals of A (regressed
      xlim = c(-2.7,2.7), ylim = c(-2.7,2.7), pch=16)
```

**residuals of A (regressed on C)
against the residuals of B (regressed on C)**
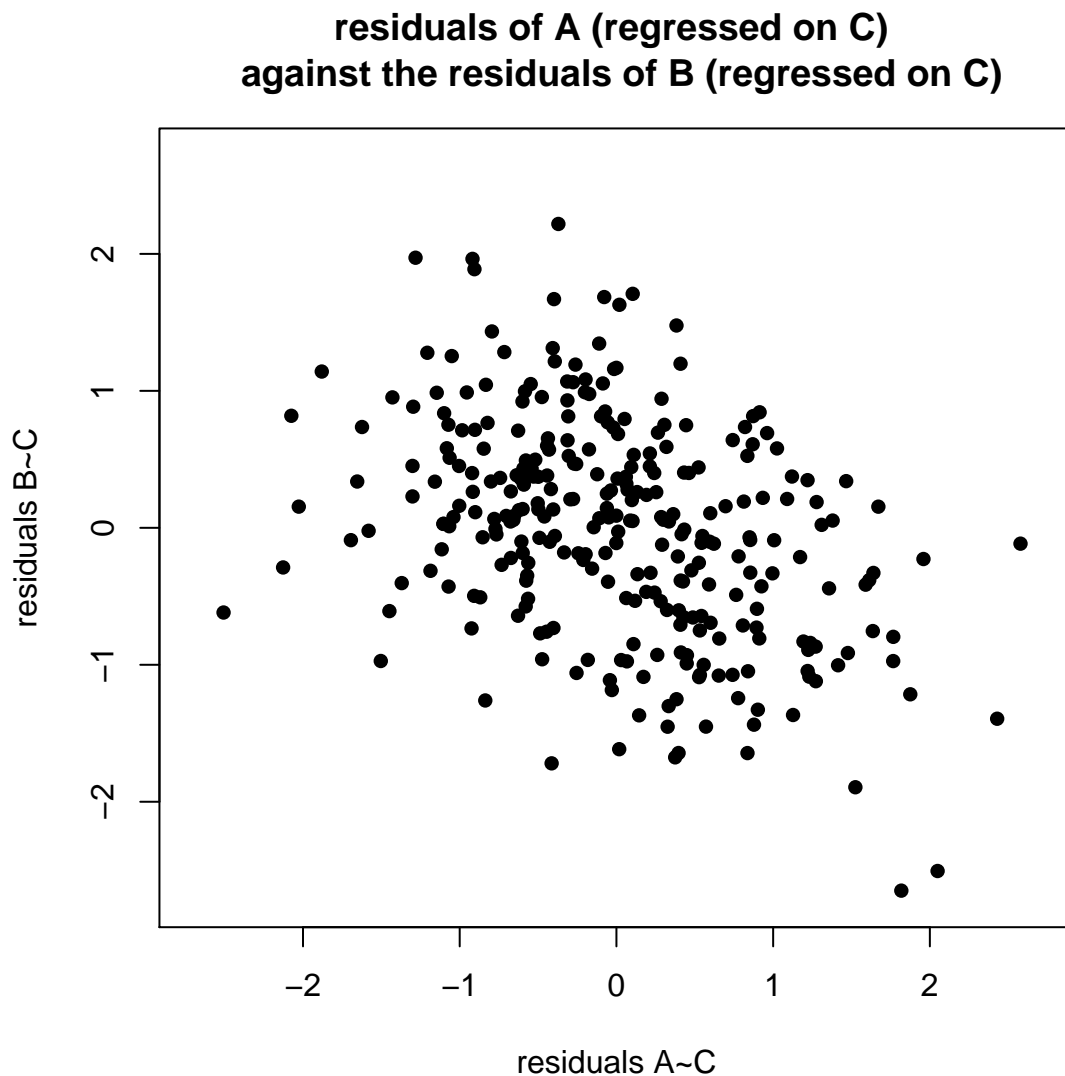


There seems to be a negative correlation between he residuals of $A$ (regressed on $C$) against the residuals of $B$ (regressed on $C$). Higher residuals of $A$ tend to be lower residuals of $B$. This indicates that the partial correlation $\rho_{A,B|C}$ is negative as well.

- Use the function `cor.test()` to test the null hypothesis of no correlation between the residuals of $A$ (regressed on $C$) and the residuals of $B$ (regressed on $C$). What is your conclusion? Does this agree with your expectation based on the underlying DAG in Figure 2?

```
cor.test(res_AC, res_BC)
```

```
##
##  Pearson's product-moment correlation
##
## data:  res_AC and res_BC
## 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
```

We can see that our p-value is smaller than the significance level $\alpha = 0.05$. Therefore we reject the null-hypothesis. There is a negative correlation between the residuals. This leads to the conclusion, that there is a partial correlation $\rho_{A,B|C}$. This is also in agreement with figure 2. We can see that $A$ is not d-separated from $B$ given $C$.

## Problem 26: Running the PC algorithm

Install and load the R package pcalg. Use the function pc() to run the PC algorithm on the data in MVN_DAG.rds, and plot the result. Does the algorithm successfully learn the structure of the data-generating graph in Figure 2? How is the result affected by the significance level $\alpha$ for the conditional independence tests?

```
library(pcalg)
C = cor(data)
n = dim(data)[1]
pc.fit = pc(suffStat = list(C=C,n=n), indepTest = gaussCItest, alpha = 0.05,
    labels = colnames(data))

plot(pc.fit, main = "Estimated CPDAG")
```



Estimated CPDAG
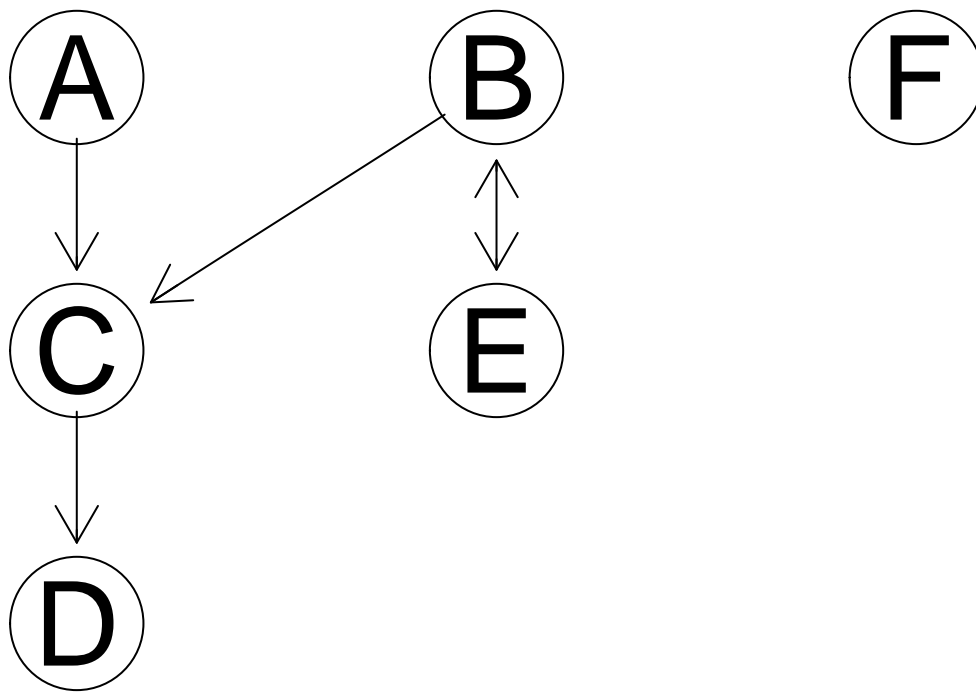
We can see that the algorithm successfully learns the structure of the data-generating graph in figure 2. The only discrepancy is that the edge between B and E is now undirected.
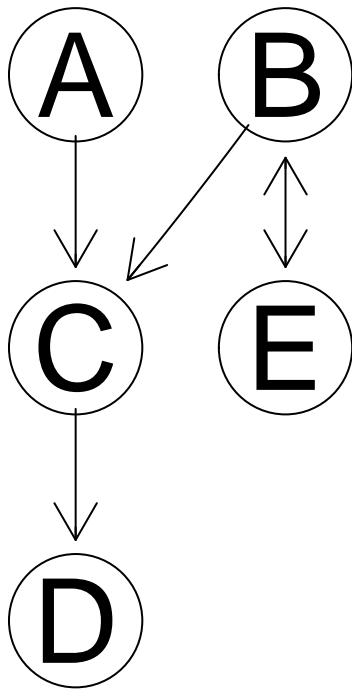
```
# popular levels of significance
alphas = c(0.1, 0.05, 0.01, 0.005, 0.001, 0.0001)

plot_list=sapply(alphas, FUN=pc, suffStat = list(C=C,n=n), indepTest = gaussCItest, labels = colnames(da

par(mfrow=c(1,2))
invisible(mapply(function(x,y) plot(x, main=bquote(alpha==.(y))),x=plot_list,y=alphas))
```
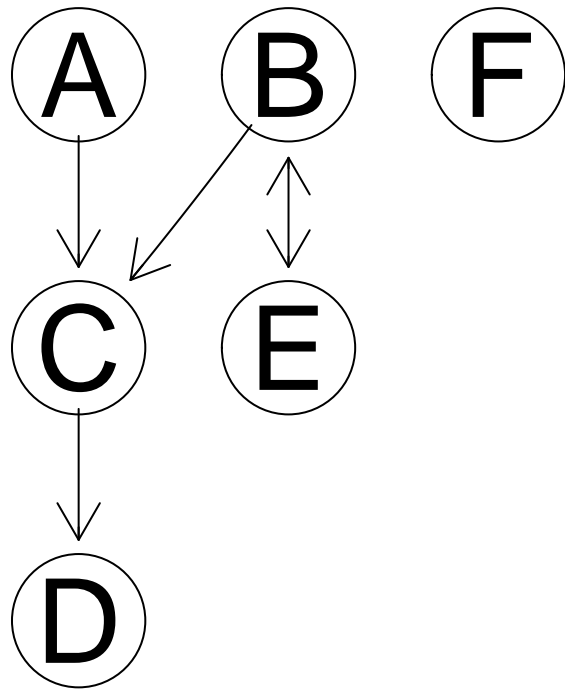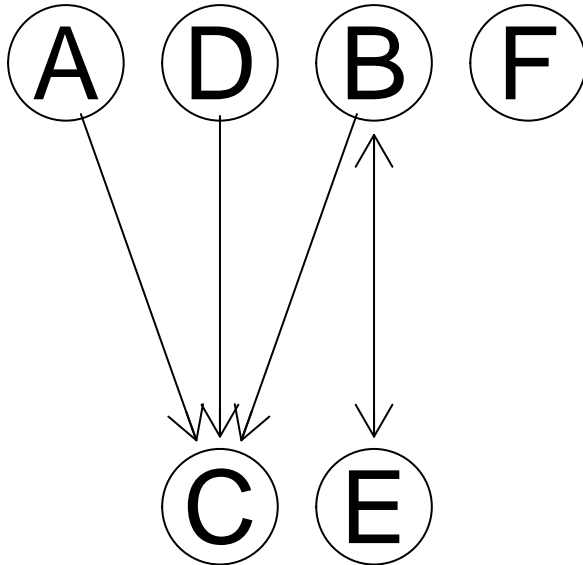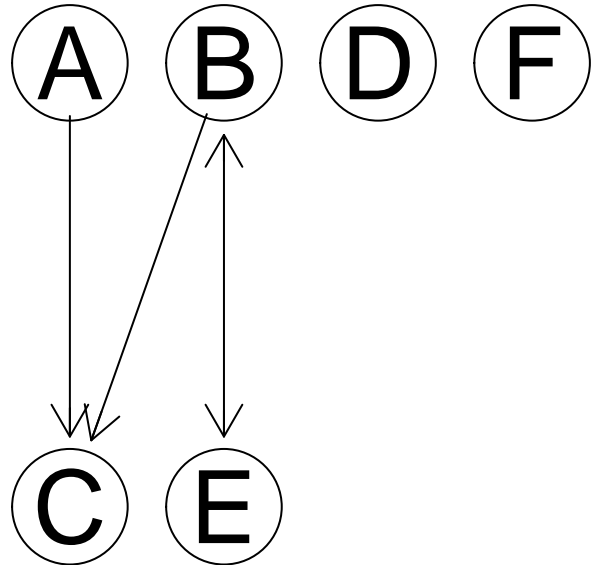
$$\alpha = 0.1$$ $$\alpha = 0.05$$

α = 0.01

α = 0.005
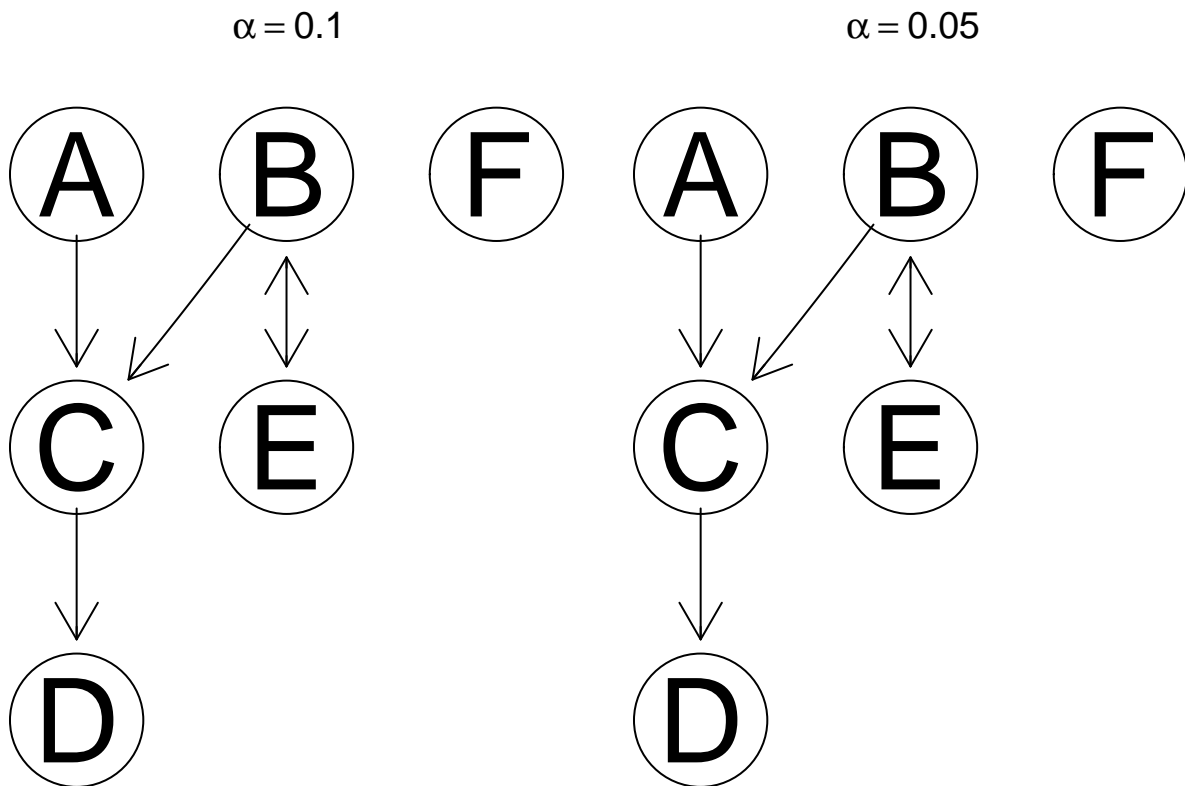
We see that for very small significance levels $\alpha$ the position of node $D$ gets unstable. The relationship between the other nodes stays stable.