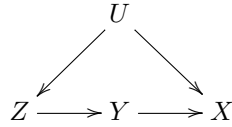


BACKDOOR PATH WITH A COLLIDER AND UNMEASURED CONFOUNDING



The corresponding probability model factorizes as

$$f_{U,X,Y,Z}(u, x, y, z) = f_U(u) f_{Z|U}(z|u) f_{Y|Z}(y|z) f_{X|U,Y}(x|u, y)$$

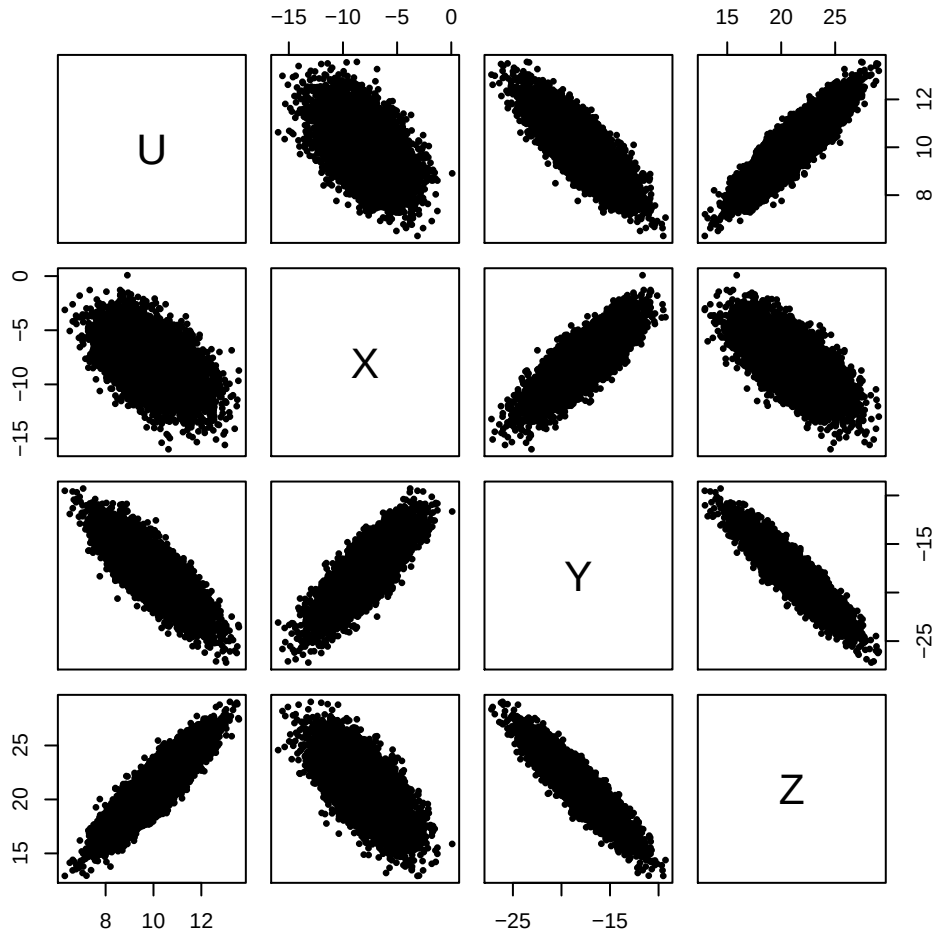
In this graph, we have two paths from Z to Y

- Path (Z, Y) this is a directed path;
- Path (Z, U, X, Y) : this is an undirected path that is also a backdoor path.

However the second path is blocked at the collider X , so there is no open backdoor path, and thus the effect of Z on Y is only found through the first.

```

set.seed(2384)
n<-10000
U<-rnorm(n,10,1)
Z<-rnorm(n,2*U+1,1)
Y<-rnorm(n,-Z+3,1)
X<-rnorm(n,Y+U,1)
data1<-data.frame(U,X,Y,Z);pairs(data1,pch=19,cex=0.5)
  
```



If we regress Y on Z , then the correct relationship is recovered.

```
coef(summary(lm(Y~Z)))
```

	Estimate	Std. Error	t value	Pr(> t)
+ (Intercept)	2.9619940	0.094021847	31.50325	8.955852e-208
+ Z	-0.9980352	0.004447717	-224.39271	0.000000e+00

However, if we condition on X in the regression, we see that bias is introduced in the estimation of the coefficient.

```
coef(summary(lm(Y~Z+X)))
```

	Estimate	Std. Error	t value	Pr(> t)
+ (Intercept)	0.9038038	0.072623021	12.44514	2.711881e-35
+ Z	-0.7273139	0.004393444	-165.54528	0.000000e+00
+ X	0.4535046	0.004916645	92.23862	0.000000e+00

If we condition on U only, then the direct effect of Z on Y is correctly captured, as the path is still blocked at X

```
coef(summary(lm(Y~Z+U)))
```

	Estimate	Std. Error	t value	Pr(> t)
+ (Intercept)	2.93204733	0.10038171	29.2089797	4.719408e-180
+ Z	-1.00567039	0.01000684	-100.4983396	0.000000e+00
+ U	0.01904492	0.02235960	0.8517557	3.943701e-01

If we condition on U and X , then the direct effect of Z on Y is also not captured.

```
coef(summary(lm(Y~Z+X+U)))
```

	Estimate	Std. Error	t value	Pr(> t)
+ (Intercept)	1.4741443	0.072171300	20.42563	6.981817e-91
+ Z	-0.5022097	0.008639794	-58.12751	0.000000e+00
+ X	0.4989366	0.004951508	100.76459	0.000000e+00
+ U	-0.4938430	0.016551569	-29.83663	1.837148e-187

This is due to *selection bias*: conditioning on a descendant of Y will lead to bias in most circumstances. Consider the simple chain graph

$$Z \longrightarrow Y \longrightarrow X$$

The corresponding probability model factorizes as

$$f_{X,Y,Z}(x,y,z) = f_Z(z)f_{Y|Z}(y|z)f_{X|Y,Z}(x|y,z).$$

Clearly we can integrate out x from the joint density to leave

$$f_{Y,Z}(y,z) = f_Z(z)f_{Y|Z}(y|z)$$

leaving the (Z, Y) relationship unchanged. However, we have that

$$f_{Y|X,Z}(y|x,z) = \frac{f_{X,Y,Z}(x,y,z)}{f_{X,Z}(x,z)} = \frac{f_Z(z)f_{Y|Z}(y|z)f_{X|Y,Z}(x|y,z)}{f_Z(z)f_{X|Z}(x|z)} = \frac{f_{X|Y,Z}(x|y,z)}{f_{X|Z}(x|z)}f_{Y|Z}(y|z)$$

and in general

$$\frac{f_{X|Y,Z}(x|y,z)}{f_{X|Z}(x|z)} \neq 1$$

so

$$f_{Y|X,Z}(y|x,z) \neq f_{Y|Z}(y|z).$$

Notice, however, that we can change the data generating model to make the analyses agree. If the conditional model for Y given Z is instead

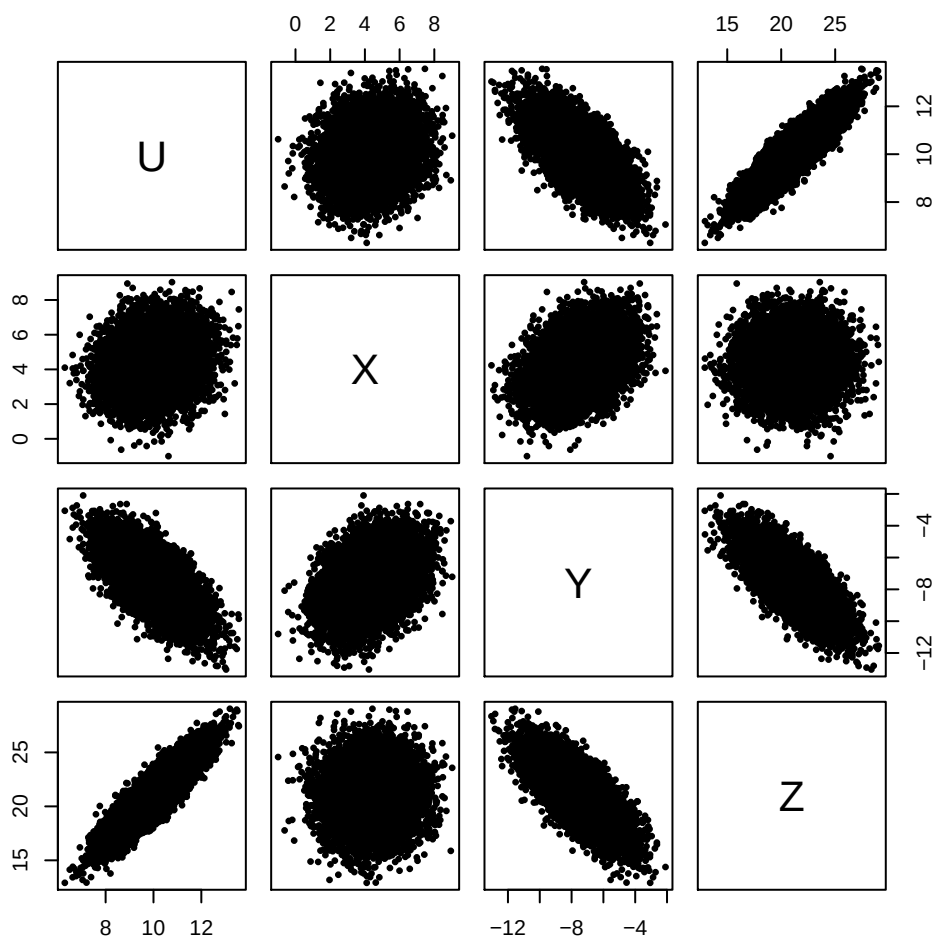
$$Y = -0.5Z + \epsilon$$

and then

$$X = 0.75Y + Z + \epsilon$$

then the effect of conditioning changes.

```
set.seed(2384)
n<-10000
U<-rnorm(n,10,1)
Z<-rnorm(n,2*U+1,1)
Y<-rnorm(n,-0.5*Z+3,1)
X<-rnorm(n,0.75*Y+U,1)
data2<-data.frame(U,X,Y,Z);pairs(data2,pch=19,cex=0.5)
```



If we regress Y on Z , then the correct relationship is recovered.

```
coef(summary(lm(Y~Z)))
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.9619940	0.094021847	31.50325	8.955852e-208
+ Z	-0.4980352	0.004447717	-111.97548	0.000000e+00

Now, if we condition on X in the regression, we see that bias is still not present, even though there is an open, biasing path.

```
coef(summary(lm(Y~Z+X)))
```

	Estimate	Std. Error	t value	Pr(> t)
+ (Intercept)	1.342382	0.080650106	16.64451	2.213602e-61
+ Z	-0.509786	0.003656305	-139.42652	0.000000e+00
+ X	0.426448	0.006136557	69.49304	0.000000e+00

If we condition on U only, then the direct effect of Z on Y is correctly captured, as the path is still blocked at X

```
coef(summary(lm(Y~Z+U)))
```

	Estimate	Std. Error	t value	Pr(> t)
+ (Intercept)	2.93204733	0.10038171	29.2089797	4.719408e-180
+ Z	-0.50567039	0.01000684	-50.5324956	0.000000e+00
+ U	0.01904492	0.02235960	0.8517557	3.943701e-01

However, if we condition on U and X , then the direct effect of Z on Y is also not captured.

```
coef(summary(lm(Y~Z+X+U)))
```

	Estimate	Std. Error	t value	Pr(> t)
+ (Intercept)	1.8803217	0.081180644	23.16219	1.173944e-115
+ Z	-0.3218234	0.008334229	-38.61466	3.520676e-304
+ X	0.4804575	0.006336652	75.82198	0.000000e+00
+ U	-0.4725596	0.018959792	-24.92430	4.436511e-133

As a final summary, we can inspect the **inverse** of the sample correlation matrices:

```
round(solve(cor(data1)),6)
```

	U	X	Y	Z
+ U	6.088401	-2.031254	2.461387	-4.562629
+ X	-2.031254	4.020629	-4.965237	-0.029805
+ Y	2.461387	-4.965237	12.168425	5.592981
+ Z	-4.562629	-0.029805	5.592981	10.175987

```
round(solve(cor(data2)),6)
```

	U	X	Y	Z
+ U	6.088401	-1.354396	1.124682	-4.552405
+ X	-1.354396	1.787544	-1.522344	-0.004926
+ Y	1.124682	-1.522344	3.550755	1.711425
+ Z	-4.552405	-0.004926	1.711425	6.354784

Note that in both cases, the entry in the position relating X and Z is almost zero. This is an indication that conditional on the other variables, X and Z are uncorrelated (actually independent here in this Gaussian case). This entry corresponds to the partial correlation between the two variables.