How I would determine causal relationships

```
for i, var1 in enumerate(data):
    for j, var2 in enumerate(data):
        if i == j:
            cm[i][j] = 1.0
        else:
            r = Σ((d_var1)(d_var2)) / sd_var1 * sd_var2)
            cm[i][j] = r
            cm[j][i] = r
```

```
for i, var1 in enumerate(data):
    for j, var2 in enumerate(data):
        if abs(cm[i][j]) > some_threshold
            add_undirected_edge(var1, var2)
```

Search for V structures

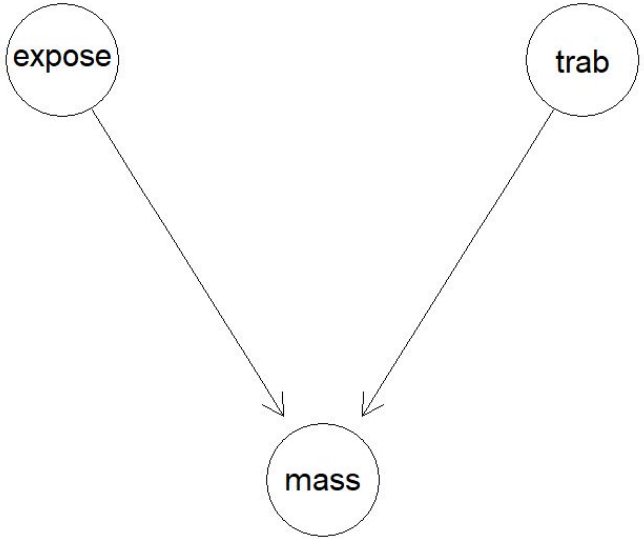
If **A** ⊥ **B** | **C**,

Orient edges **A** ← **C** → **B**

Else

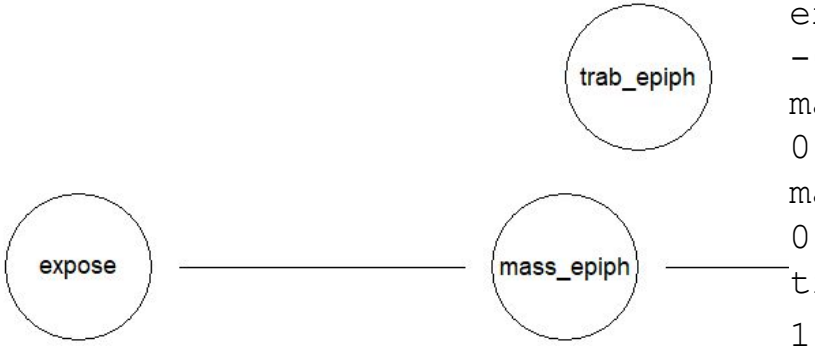
Orient edges **A** → **C** ← **B**

Alwood

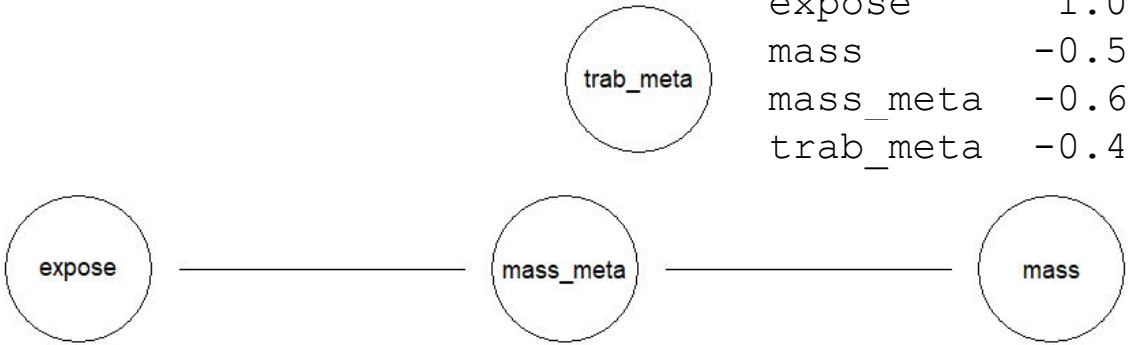


	expose	trab	mass
expose	1.00	-0.26	-0.48
trab	-0.26	1.00	0.77
mass	-0.48	0.77	1.00

GLDS

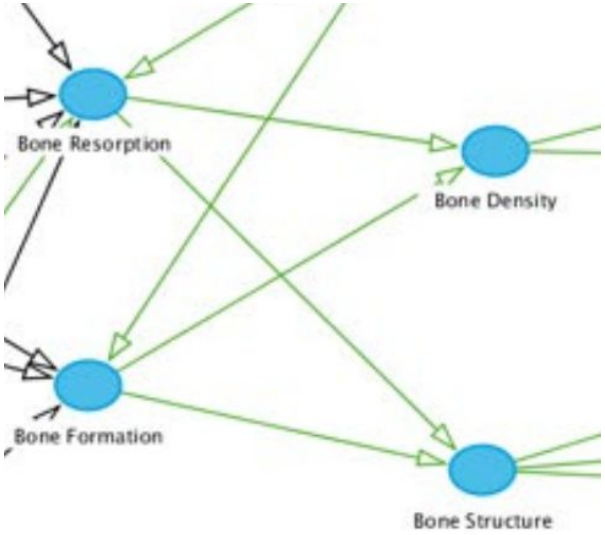


	expose	mass	mass_epiph	trab_epiph
expose	1.00	-0.56	-0.91	-0.80
mass	-0.56	1.00	0.73	0.58
mass_epiph	-0.91	0.73	1.00	0.81
trab_epiph	-0.80	0.58	0.81	1.00



	expose	mass	mass_meta	trab_meta
expose	1.00	-0.56	-0.67	-0.48
mass	-0.56	1.00	0.75	0.31
mass_meta	-0.67	0.75	1.00	0.44
trab_meta	-0.48	0.31	0.44	1.00

Turner



	expose	mass	resorp	form
expose	1.00	-0.63	0.72	-0.42
mass	-0.63	1.00	-0.78	0.56
resorp	0.72	-0.78	1.00	-0.29
form	-0.42	0.56	-0.29	1.00

Ko

	unload	dur	expose	mass	trab	stren	resorp	form
unload	1.00	-0.02	0.73	-0.41	-0.26	-0.26	NA	NA
dur	-0.02	1.00	0.53	-0.06	-0.10	0.19	NA	NA
expose	0.73	0.53	1.00	-0.34	-0.25	-0.07	NA	NA
mass	-0.41	-0.06	-0.34	1.00	0.80	0.26	NA	NA
trab	-0.26	-0.10	-0.25	0.80	1.00	0.19	NA	NA
stren	-0.26	0.19	-0.07	0.26	0.19	1.00	NA	NA
resorp	NA	NA	NA	NA	NA	NA	1	NA
form	NA	NA	NA	NA	NA	NA	NA	1