Decompositions

Sophie Woodward

2023-06-08

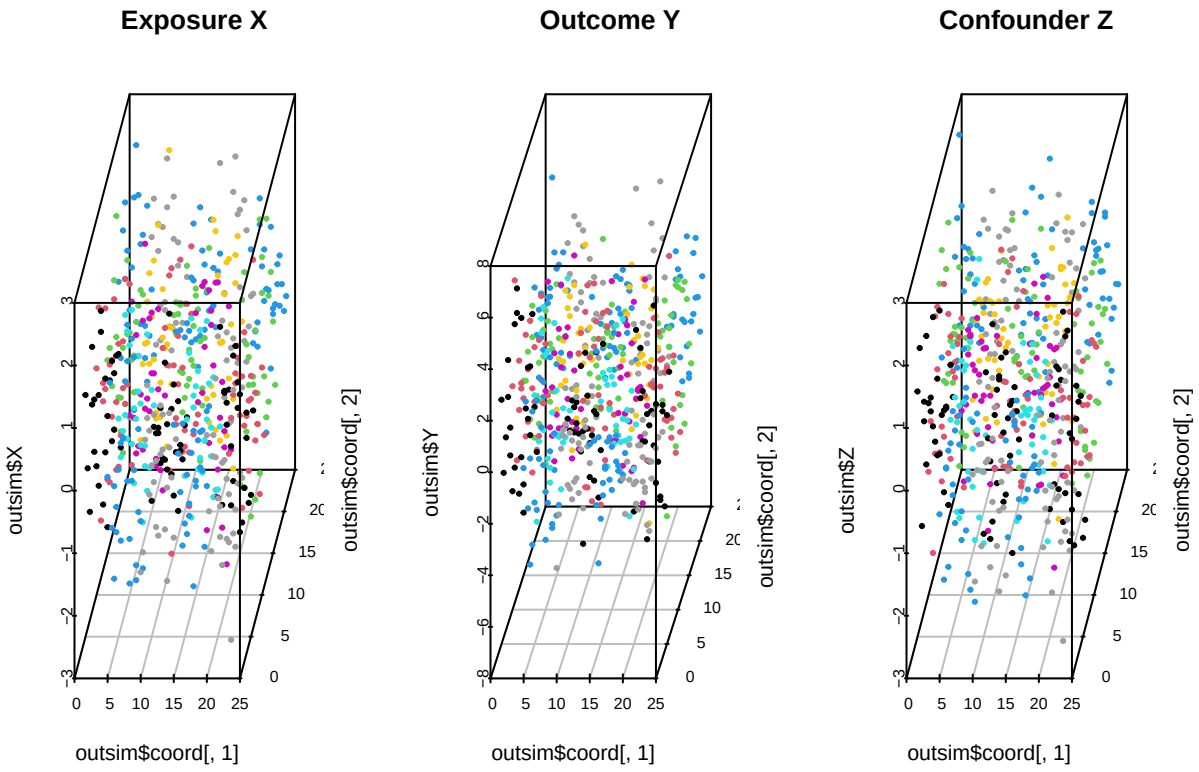
Generate Data

First let's generate some data using the spectral decomposition. Note that the covariance between X and Z is decreasing with spatial scale by construction (see rhox - things are flipped since spectral decomp projects in order of highest to lowest freq). We assume Y is a linear function of X and Z with coefficients 2 and -1 respectively.

```
outsim = sim(5, outcome = 'linear', rhox = seq(0,1, by = 1/(5^4-1)),
             decomposition = 'spectral')
```

```
## [1] "Creating coordinates and groups"
## [1] "Creating adjacency"
## [1] "Creating X and Z"
## [1] "Simulating outcome"
```

```
par(mfrow = c(1,3))
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$X,
              color = outsim$groups[,1], pch = 20, angle = 60, main = 'Exposure X')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Y,
              color = outsim$groups[,1], pch = 20, angle = 60, main = 'Outcome Y')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Z,
              main = 'Confounder Z',
              color = outsim$groups[,1], pch = 20, angle = 60)
```



Analysis

```
testnested = analysis(n = 5, A = outsim$A,
                X = outsim$X,
                Y = outsim$Y,
                Z = outsim$Z,
                groups = outsim$groups,
                decomposition = 'nested')
```

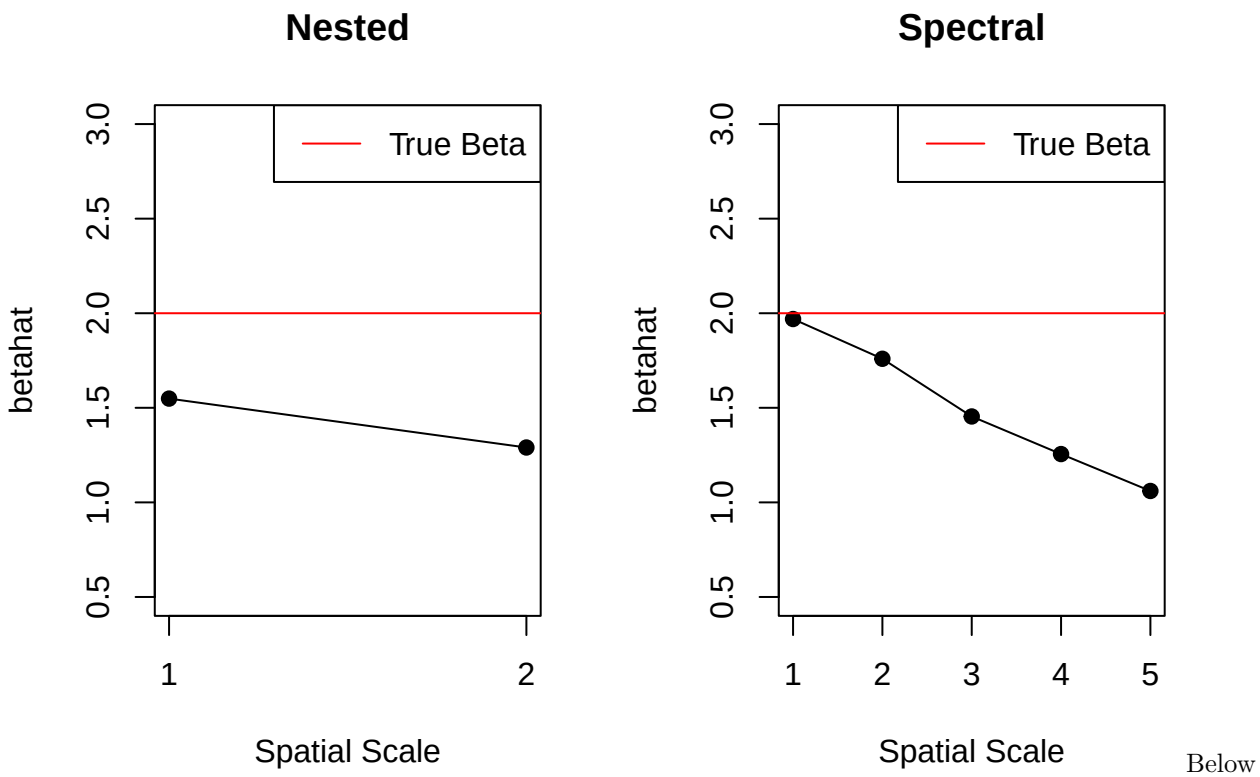
```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
testspectral = analysis(n = 5, A = outsim$A,
                X = outsim$X,
                Y = outsim$Y,
                Z = outsim$Z,
                groups = outsim$groups,
                decomposition = 'spectral')
```

```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
par(mfrow = c(1,2))
plot(1:2, testnested, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19, type = 'l',
     main = 'Nested', xaxp = c(1,2,1), ylim = c(0.5,3))
```

```
points(1:2, testnested, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')
plot(1:5, testspectral, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19,
     type = 'l', ylim = c(0.5,3), main = 'Spectral')
points(1:5, testspectral, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')
```



Below we see that the bias of the $\hat{\beta}_x$ estimates obtained from the spectral decomposition indeed decreases with smaller spatial scale, as desired. This is also true for the nested decomposition but the bias is pretty bad for both scales.

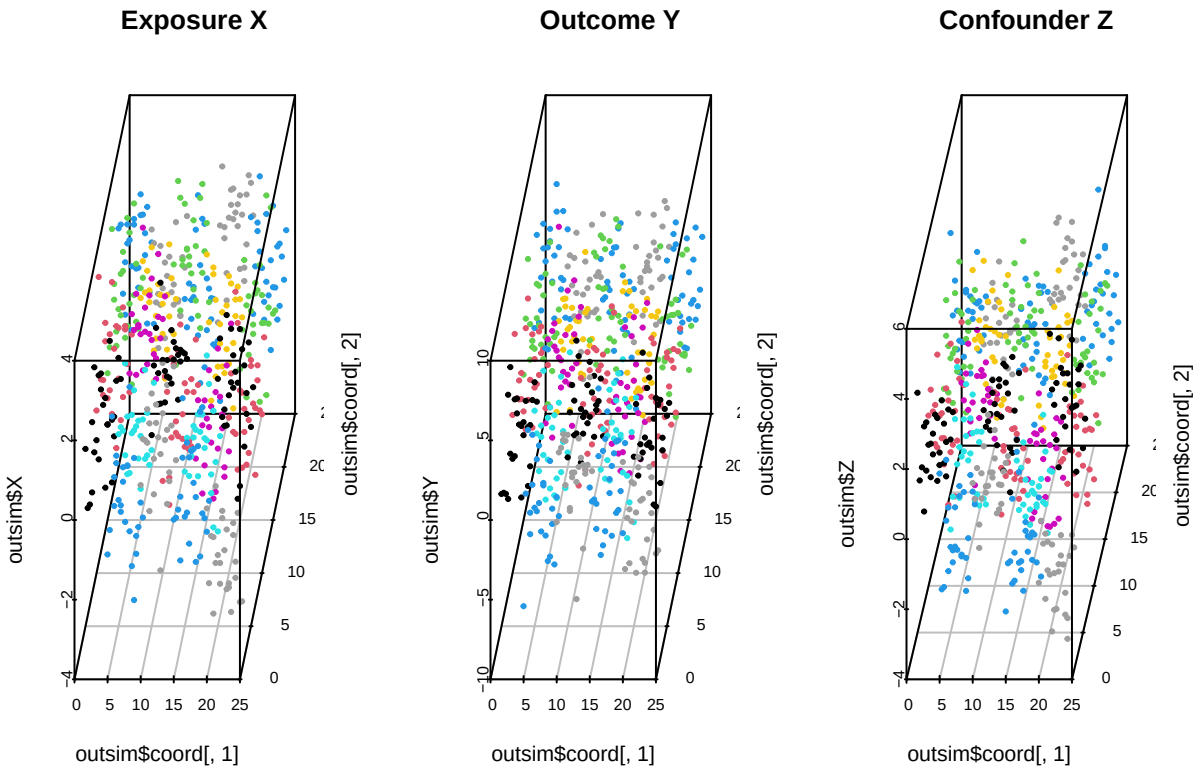
Repeat this with Nested DGM

Let's generate some data using the nested decomposition. The covariance between X and Z is decreasing with spatial scale by construction (see rhox). Note that we consider only two spatial scales for now. We assume Y is a linear function of X and Z with coefficients 2 and -1 respectively.

```
outsim = sim(n = 5, outcome = 'linear', rhox = c(0.9, 0.001),
             decomposition = 'nested')
```

```
## [1] "Creating coordinates and groups"
## [1] "Creating adjacency"
## [1] "Creating X and Z"
## [1] "Simulating outcome"
```

```
par(mfrow = c(1,3))
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$X,
              color = outsim$groups, pch = 20, angle = 60, main = 'Exposure X')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Y,
              color = outsim$groups, pch = 20, angle = 60, main = 'Outcome Y')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Z,
              main = 'Confounder Z',
              color = outsim$groups, pch = 20, angle = 60)
```



Repeat the analysis

Below we see that the bias of the $\hat{\beta}_x$ estimates obtained from the nested decomposition indeed decreases with smaller spatial scale, as desired. This is also true for the spectral decomposition.

```
par(mfrow = c(1,2))
testnested = analysis(n = 5, A = outsim$A,
                X = outsim$X,
                Y = outsim$Y,
                Z = outsim$Z,
                groups = outsim$groups,
                decomposition = 'nested')
```

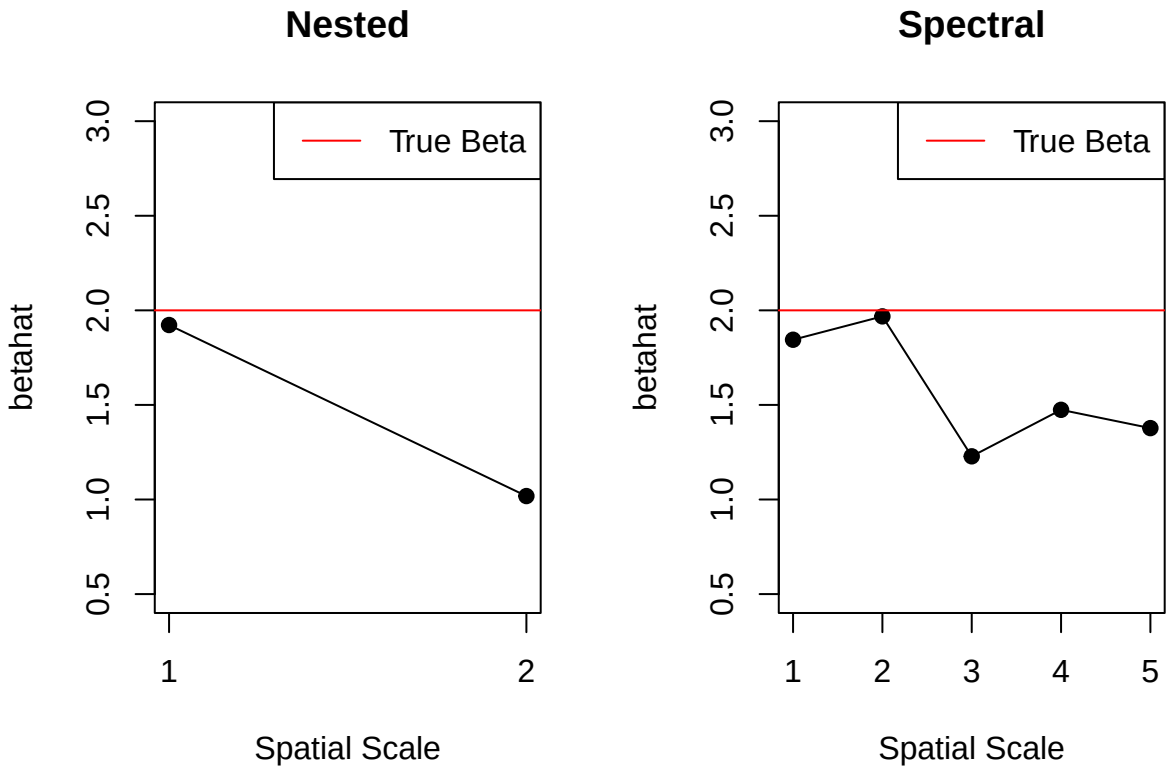
```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
plot(1:2, testnested, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19, type = 'l',
     main = 'Nested', xaxp = c(1,2,1), ylim = c(0.5,3))
points(1:2, testnested, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')

testspectral = analysis(n=5, A = outsim$A,
                X = outsim$X,
                Y = outsim$Y,
                Z = outsim$Z,
                groups = outsim$groups,
                decomposition = 'spectral')
```

```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
plot(1:5, testspectral, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19,
     type = 'l', ylim = c(0.5,3), main = 'Spectral')
points(1:5, testspectral, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')
```



Repeat with multiple levels

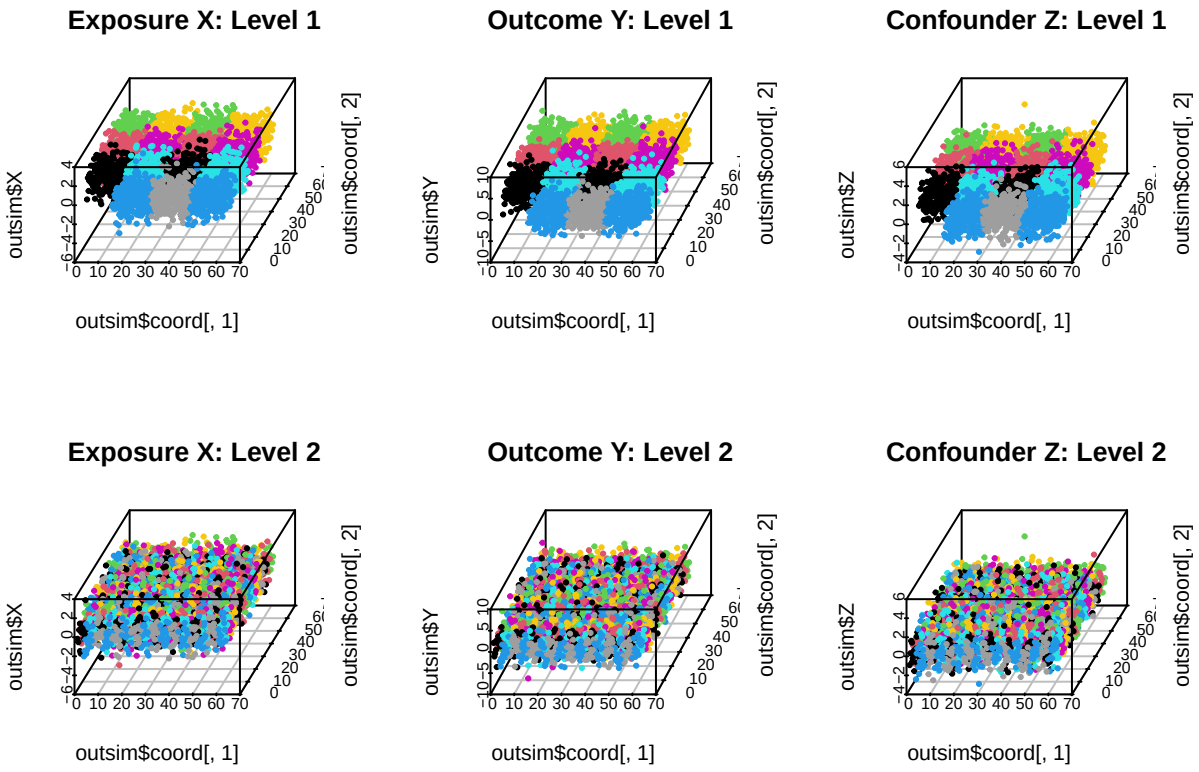
We will now repeat the analysis using 4 subgroups within a group and 3 nested levels ($4^{2\times3}$ units total).

Generate Data

```
outsim = sim(n=4,l=3,outcome = 'linear', rhox = seq(0,1, by = 1/(4^(2*3)-1)),
             decomposition = 'spectral')
```

```
## [1] "Creating coordinates and groups"
## [1] "Creating adjacency"
## [1] "Creating X and Z"
## [1] "Simulating outcome"
```

```
par(mfrow = c(2,3))
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$X,
              color = outsim$groups[,1], pch = 20, angle = 60,
              main = 'Exposure X: Level 1')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Y,
              color = outsim$groups[,1], pch = 20, angle = 60, main = 'Outcome Y: Level 1')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Z,
              main = 'Confounder Z: Level 1',
              color = outsim$groups[,1], pch = 20, angle = 60)
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$X,
              color = outsim$groups[,2], pch = 20, angle = 60, main = 'Exposure X: Level 2')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Y,
              color = outsim$groups[,2], pch = 20, angle = 60, main = 'Outcome Y: Level 2')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Z,
              main = 'Confounder Z: Level 2',
              color = outsim$groups[,2], pch = 20, angle = 60)
```



Analysis

```
testnested = analysis(n = 4, A = outsim$A,
                X = outsim$X,
                Y = outsim$Y,
                Z = outsim$Z,
                groups = outsim$groups,
                decomposition = 'nested')
```

```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
testspectral = analysis(n = 4, A = outsim$A,
                X = outsim$X,
                Y = outsim$Y,
                Z = outsim$Z,
                groups = outsim$groups,
                decomposition = 'spectral')
```

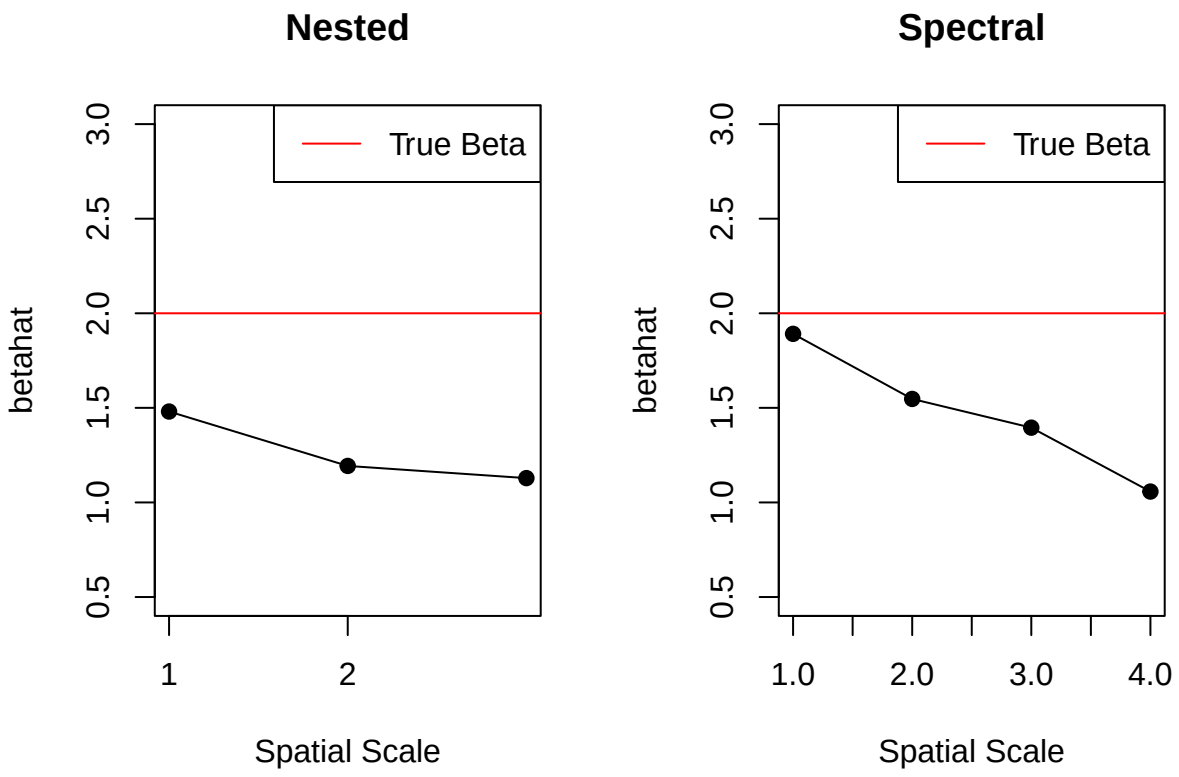
```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
par(mfrow = c(1,2))

plot(1:3, testnested, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19, type = 'l',
```

```
     main = 'Nested', xaxp = c(1,2,1), ylim = c(0.5,3))
points(1:3, testnested, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')

plot(1:4, testspectral, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19,
     type = 'l', ylim = c(0.5,3), main = 'Spectral')
points(1:4, testspectral, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')
```



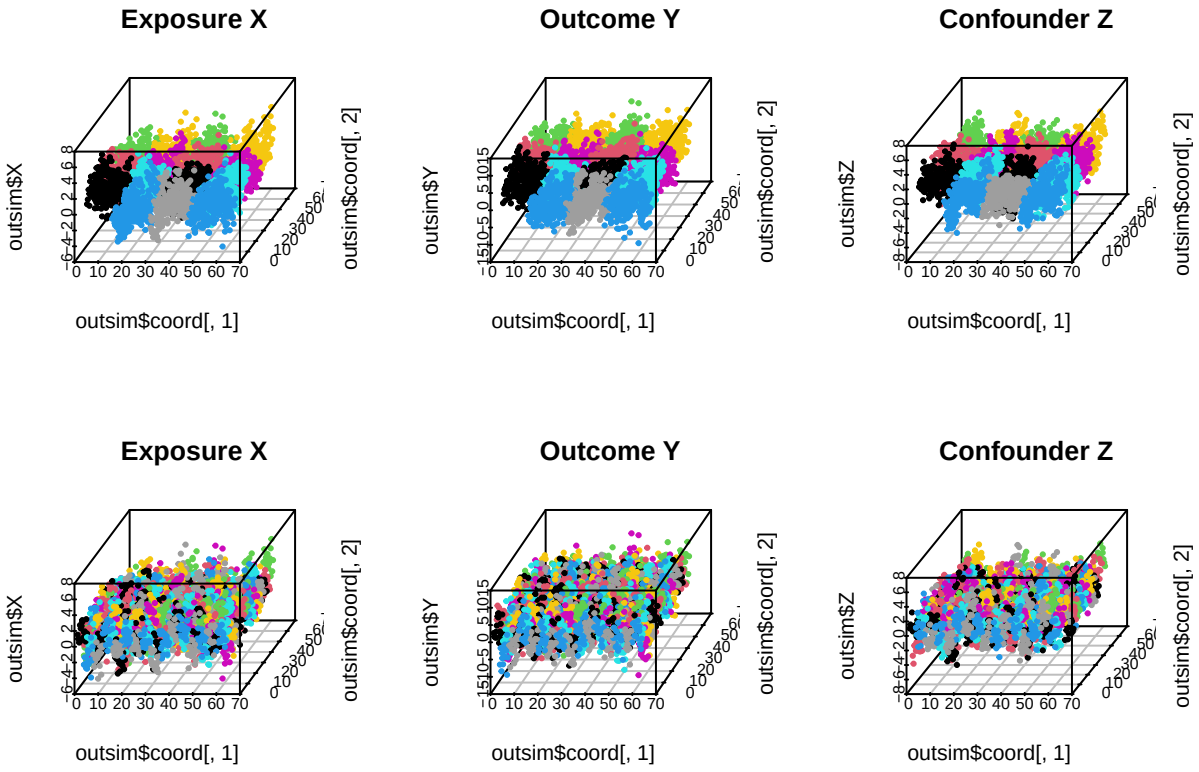
Repeat this with Nested DGM

```
outsim = sim(n = 4, l=3, outcome = 'linear', rhox = c(0.9, 0.5, 0.001),
             decomposition = 'nested')
```

```
## [1] "Creating coordinates and groups"
## [1] "Creating adjacency"
## [1] "Creating X and Z"
## [1] "Simulating outcome"
```

```
par(mfrow = c(2,3))
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$X,
              color = outsim$groups[,1], pch = 20, angle = 60, main = 'Exposure X')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Y,
              color = outsim$groups[,1], pch = 20, angle = 60, main = 'Outcome Y')
```

```
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Z,
              main = 'Confounder Z',
              color = outsim$groups[,1], pch = 20, angle = 60)
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$X,
              color = outsim$groups[,2], pch = 20, angle = 60, main = 'Exposure X')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Y,
              color = outsim$groups[,2], pch = 20, angle = 60, main = 'Outcome Y')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Z,
              main = 'Confounder Z',
              color = outsim$groups[,2], pch = 20, angle = 60)
```



Repeat the analysis

```
testnested = analysis(n = 4, A = outsim$A,
                      X = outsim$X,
                      Y = outsim$Y,
                      Z = outsim$Z,
                      groups = outsim$groups,
                      decomposition = 'nested')
```

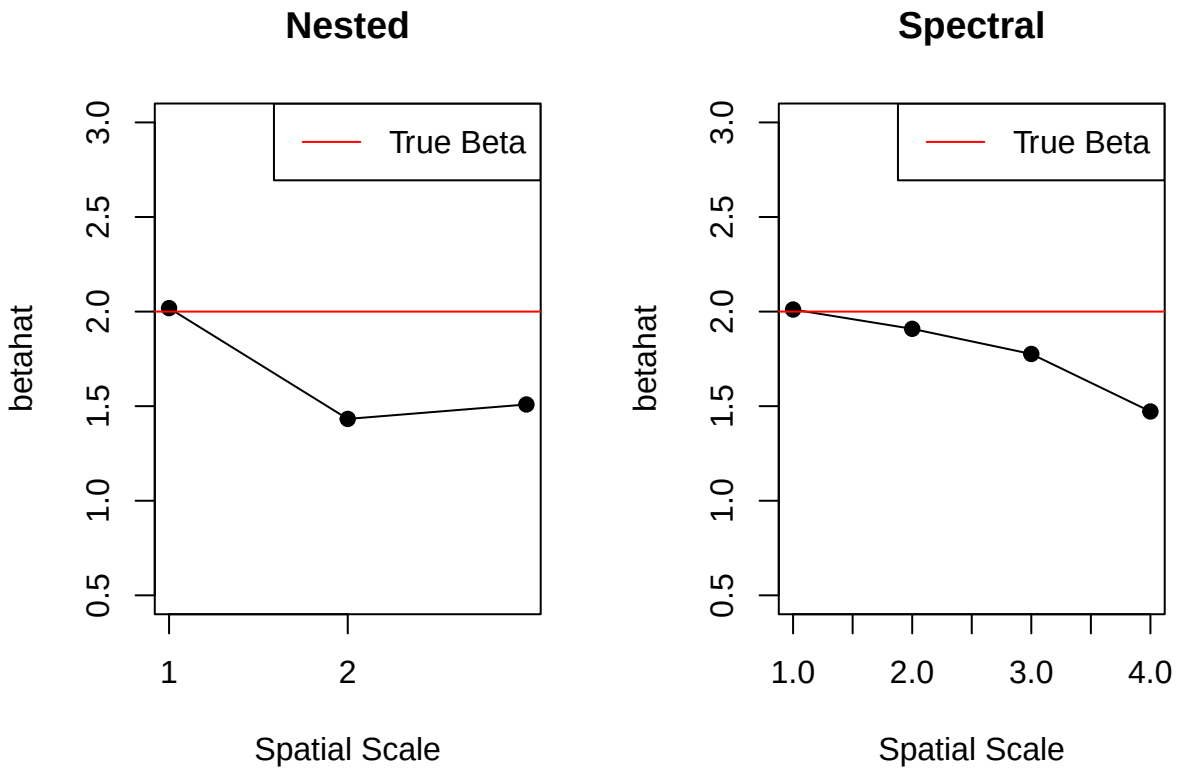
```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
testspectral = analysis(n = 4, A = outsim$A,
                      X = outsim$X,
```

```
                Y = outsim$Y,
                Z = outsim$Z,
                groups = outsim$groups,
                decomposition = 'spectral')
```

```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
par(mfrow = c(1,2))
plot(1:3, testnested, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19, type = 'l',
     main = 'Nested', xaxp = c(1,2,1), ylim = c(0.5,3))
points(1:3, testnested, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')
plot(1:4, testspectral, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19,
     type = 'l', ylim = c(0.5,3), main = 'Spectral')
points(1:4, testspectral, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')
```



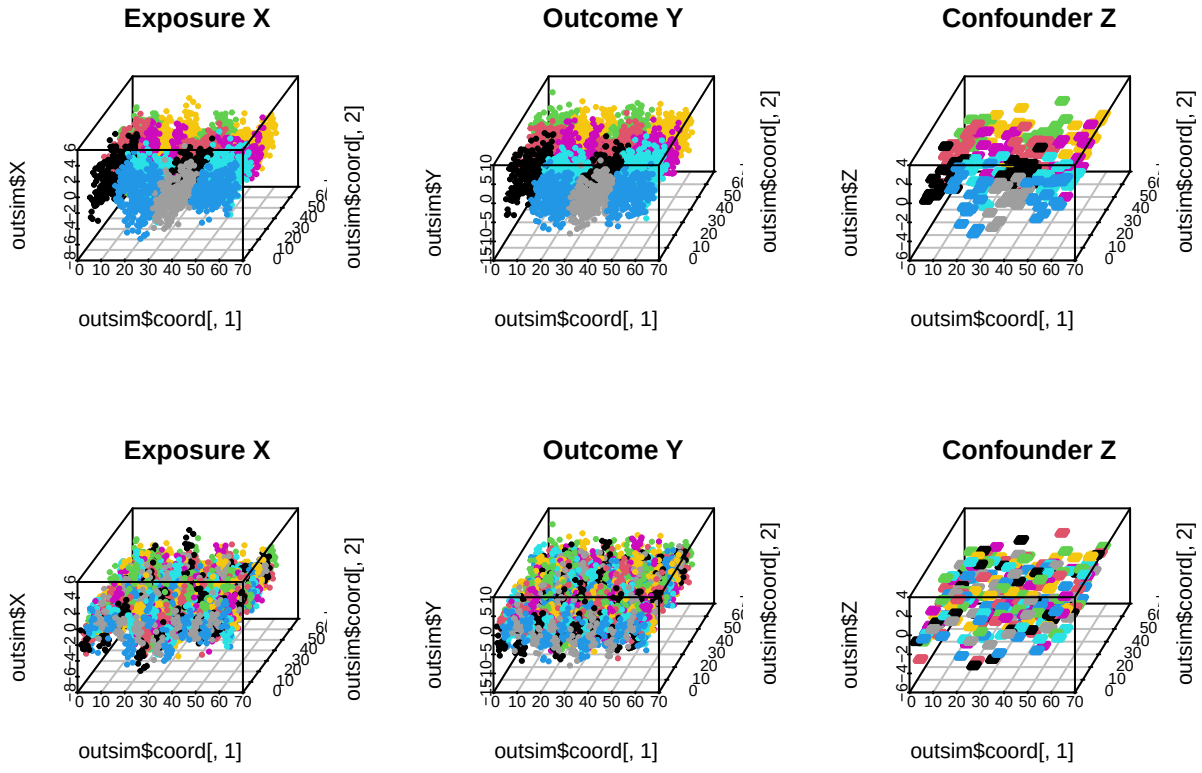
Try truncation

Here we generate data but remove the variation of Z in the nested decomposition at the smallest grid levels.

```
outsim = sim(n = 4, l=3, truncate = 2, outcome = 'linear', rhox = c(0.9, 0.5, 0.001),
             decomposition = 'nested')
```

```
## [1] "Creating coordinates and groups"
## [1] "Creating adjacency"
## [1] "Creating X and Z"
## [1] "Simulating outcome"
```

```
par(mfrow = c(2,3))
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$X,
              color = outsim$groups[,1], pch = 20, angle = 60, main = 'Exposure X')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Y,
              color = outsim$groups[,1], pch = 20, angle = 60, main = 'Outcome Y')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Z,
              main = 'Confounder Z',
              color = outsim$groups[,1], pch = 20, angle = 60)
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$X,
              color = outsim$groups[,2], pch = 20, angle = 60, main = 'Exposure X')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Y,
              color = outsim$groups[,2], pch = 20, angle = 60, main = 'Outcome Y')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Z,
              main = 'Confounder Z',
              color = outsim$groups[,2], pch = 20, angle = 60)
```



Analysis

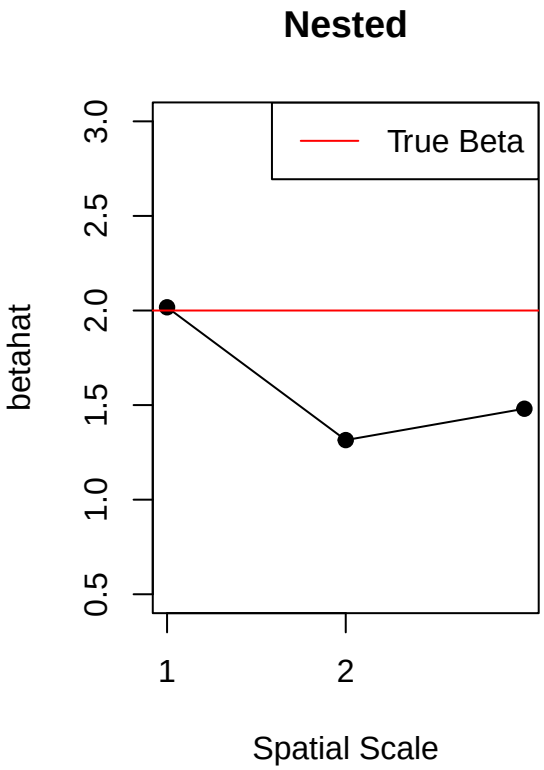
```
testnested = analysis(n = 4, A = outsim$A,
                X = outsim$X,
                Y = outsim$Y,
                Z = outsim$Z,
                groups = outsim$groups,
                decomposition = 'nested')
```

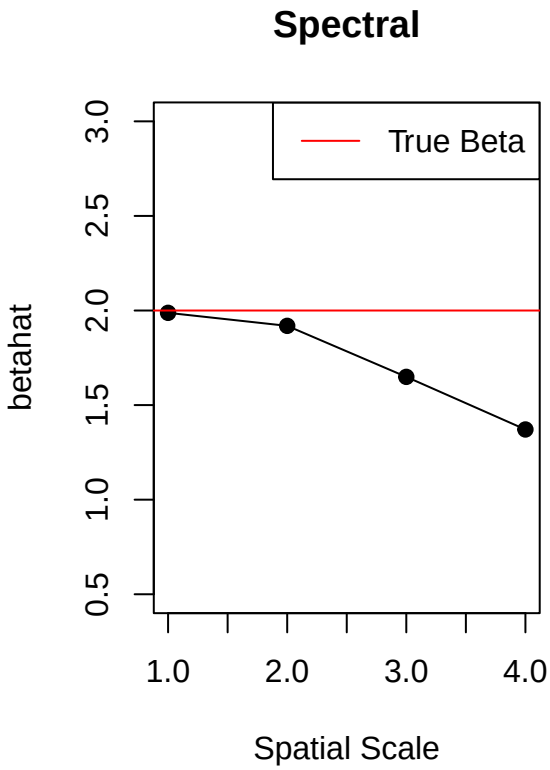
```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
testspectral = analysis(n = 4, A = outsim$A,
                X = outsim$X,
                Y = outsim$Y,
                Z = outsim$Z,
                groups = outsim$groups,
                decomposition = 'spectral')
```

```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
par(mfrow = c(1,2))
plot(1:3, testnested, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19, type = 'l',
     main = 'Nested', xaxp = c(1,2,1), ylim = c(0.5,3))
points(1:3, testnested, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')
plot(1:4, testspectral, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19,
     type = 'l', ylim = c(0.5,3), main = 'Spectral')
points(1:4, testspectral, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')
```


Nested
True Beta
betahat
0.5
1.0
1.5
2.0
2.5
3.0
1
2
Spatial Scale

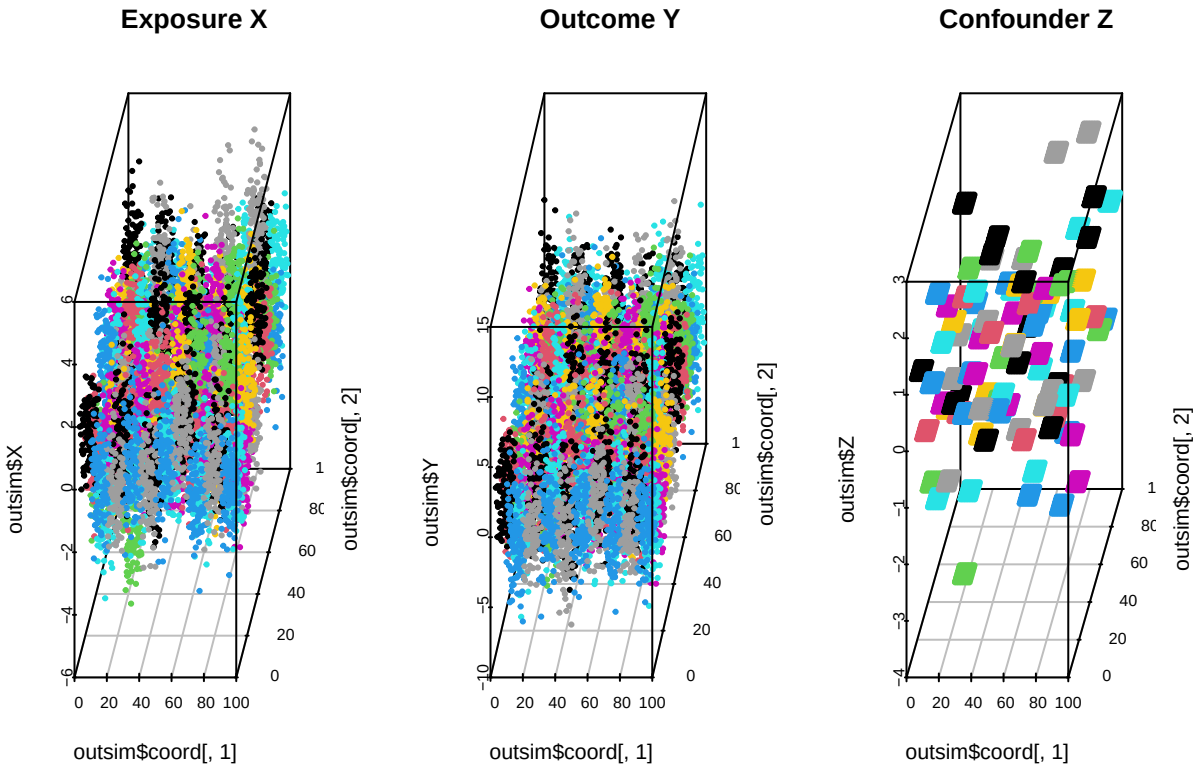

Spectral
True Beta
betahat
0.5
1.0
1.5
2.0
2.5
3.0
1.0
2.0
3.0
4.0
Spatial Scale

Trying exactly what Francesca wanted

```
outsim = sim(n = 10, l=2, truncate = 1, outcome = 'linear', rhox = c(0.9, 0.5, 0.001),
             decomposition = 'nested')
```

```
## [1] "Creating coordinates and groups"
## [1] "Creating adjacency"
## [1] "Creating X and Z"
## [1] "Simulating outcome"
```

```
par(mfrow = c(1,3))
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$X,
              color = outsim$groups[,1], pch = 20, angle = 60, main = 'Exposure X')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Y,
              color = outsim$groups[,1], pch = 20, angle = 60, main = 'Outcome Y')
scatterplot3d(x = outsim$coord[,1], outsim$coord[,2], outsim$Z,
              main = 'Confounder Z',
              color = outsim$groups[,1], pch = 20, angle = 60)
```




```
testnested = analysis(n = 10, A = outsim$A,
                      X = outsim$X,
                      Y = outsim$Y,
                      Z = outsim$Z,
                      groups = outsim$groups,
                      decomposition = 'nested')
```

```
## [1] "Perform decomposition"
## [1] "Calculate betahats"
```

```
plot(1:2, testnested, xlab = 'Spatial Scale', ylab = 'betahat', pch = 19, type = 'l',
     main = 'Nested', xaxp = c(1,2,1), ylim = c(0.5,3))
points(1:2, testnested, pch = 19)
legend('topright', legend = 'True Beta', col = 'red', lty = 1)
abline(h = 2, col = 'red')
```