Tree based correlation model implementation

Elias T Krainski, Denis Rustand, Anna Freni-Sterrantino, Janet van Niekerk, and Håvard Rue

Started in November 2023, updated in January, 2026

Abstract

In this vignette we show how to work a tree based correlation model proposed in Freni-Sterrantino et al. (2025). We recommend to look at the 'tree_model' vignette for details.

The model definition

A directed acyclic graph is defined from a set of nodes and directed edges linking them. Because the edges are directed, there is no closed loop on it. A tree is a graph with only one (directed) path between a pair of nodes. We use this fact to define correlation model. The nodes hierarchy will imposes a hierarchy on the correlation as well.

The definition start by considering two kind of nodes: parent or children. The m variables of interest, those for we want to model their correlation, will be classified as children variables. They are labeled as c_i, $i \in \{1, \ldots, m\}$. The nodes to represent children variables are leafs of the tree, having always an ancestor (parent) but with no children. Each children node has a directed edge from its parent. The path between each c_i goes through a set of k parent variables, labeled as p_j, $j \in \{1, \ldots, k\}$. The m parent variables are nodes with children nodes. and some may have parent but will still be classified as parent because they have children.

In the **R** environment we represent the parent variables with the letter `p` along with an integer number (`p1`, …, `pk`) and the children variables with the letter `c` along with an integer number (`c1`, …, `cm`). We adopt a simple way to specify the parent children representation. We consider the `~` (tilde) to represent the directed link and `+` (plus) or `-` (minus) to append the descendant to a parent. E.g.: `p1 ~ p2 + c1 + c2, p2 ~ c3 - c4 .`

Intial example

Let us consider a correlation model for three variables, with one parameter, that is the same absolute correlation between each pair but the sign may differ. This can be the case when these variables share the same latent factor. The parent is represented as `p1`, and the children variables as `c1`, `c2` and `c3`. We consider that `c3` will be negatively correlated with `c1` and `c2`. For this case we define the tree as

```
tree1 <- treepcor(p1 ~ c1 + c2 - c3)
tree1
```

```
## treepcor for 3 children and 1 parent variables
## p1 ~ + c1 + c2 - c3
```

```
summary(tree1)
```

```
##    p1
## c1  1
## c2  1
## c3 -1
```

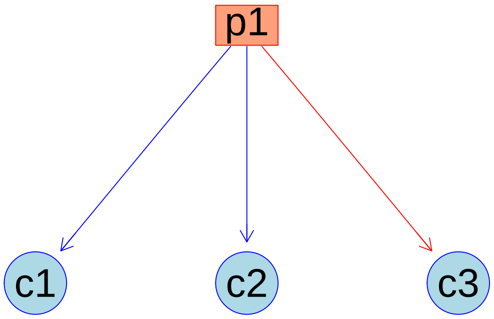
where the summary shows their relationship. The number of children and parent variables are obtained with

```
dim(tree1)
```

```
## children   parent
##        3        1
```

This tree can be visualized with

```
plot(tree1)
```



From this model definition we will use the methods to build the correlation matrix. First, we build the precision matrix structure (that is not yet a precision matrix):

```
prec(tree1)
```

```
##    c1 c2 c3 p1
## c1  1  0  0 -1
## c2  0  1  0 -1
## c3  0  0  1  1
## p1 -1 -1  1  3
```

and we can use inform the log of γ_1, which is the standard error for p_1, with:

```
q1 <- prec(tree1, theta = 0)
q1
```

```
##    c1 c2 c3 p1
## c1  1  0  0 -1
## c2  0  1  0 -1
## c3  0  0  1  1
## p1 -1 -1  1  4
```

We can obtain the correlation matrix, which is our primarily interest, from the precision matrix. However, also have a covariance method to be directly applied with

```
vcov(tree1) ## assume theta = 0 (\gamma_1 = 1)
```

```
##      [,1] [,2] [,3]
## [1,]  1.0  0.5 -0.5
## [2,]  0.5  1.0 -0.5
## [3,] -0.5 -0.5  1.0
```

```
vcov(tree1, theta = 0.5) # \gamma_1^2 = exp(2 * 0.5) = exp(1)
```

```
##            [,1]       [,2]       [,3]
## [1,]  1.0000000  0.7310586 -0.7310586
## [2,]  0.7310586  1.0000000 -0.7310586
## [3,] -0.7310586 -0.7310586  1.0000000
```

```
cov1a <- vcov(tree1, theta = 0)
cov1a
```

```
##      [,1] [,2] [,3]
## [1,]  1.0  0.5 -0.5
## [2,]  0.5  1.0 -0.5
## [3,] -0.5 -0.5  1.0
```

from where we obtain the desired matrix with

```
c1 <- cov2cor(cov1a)
round(c1, 3)
```

```
##      [,1] [,2] [,3]
## [1,]  1.0  0.5 -0.5
## [2,]  0.5  1.0 -0.5
## [3,] -0.5 -0.5  1.0
```

Correlation matrix with two parameters

In this example, we model the correlation between four variables using two parameters. We consider `c1` and `c2` having the same parent, `p1` and `c3` and `c4` having the second parent as parent. We want to have the correlation between `c3` and `c4` higher than the correlation between `c1` and `c3`. This requires `p2` to be children of `p1`. The tree for this is set by

```
tree2 <- treepcor(
  p1 ~ p2 + c1 + c2,
  p2 ~ c3 - c4)
dim(tree2)
```

```
## children   parent
##        4        2
```

```
tree2
```

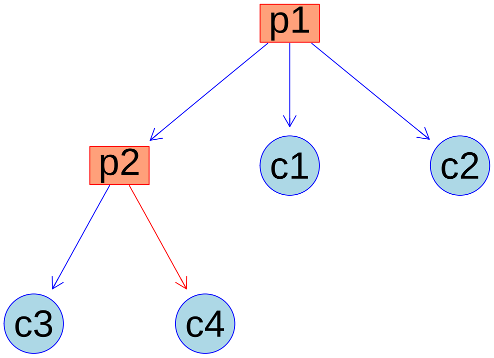
```
## treepcor for 4 children and 2 parent variables
## p1 ~ + p2 + c1 + c2
## p2 ~ + c3 - c4
```

```
summary(tree2)
```

```
##    p1 p2
## c1  1  0
## c2  1  0
## c3  0  1
## c4  0 -1
## p2  1  0
```

which can be visualized by

```
plot(tree2)
```



We can drop the last parent with

```
drop(tree2)
```

```
## treepcor for 4 children and 2 parent variables
## p1 ~ + p2 + c1 + c2
## p2 ~ + c3 - c4
```

We now have two parameters: γ_1^2 the variance of p_1 and γ_2^2 the conditional variance of p_2. For $\gamma_1 = \gamma_2 = 1$, the precision matrix can be obtained with:

```
q2 <- prec(tree2, theta = c(0, 0))
q2
```

```
##    c1 c2 c3 c4 p1 p2
## c1  1  0  0  0 -1  0
## c2  0  1  0  0 -1  0
## c3  0  0  1  0  0 -1
## c4  0  0  0  1  0  1
## p1 -1 -1  0  0  4 -1
## p2  0  0 -1  1 -1  3
```

The correlation matrix can be obtained with

```
cov2 <- vcov(tree2, theta = c(0, 0))
cov2
```

```
##            [,1]       [,2]       [,3]       [,4]
## [1,]  1.0000000  0.5000000  0.4082483 -0.4082483
## [2,]  0.5000000  1.0000000  0.4082483 -0.4082483
## [3,]  0.4082483  0.4082483  1.0000000 -0.6666667
## [4,] -0.4082483 -0.4082483 -0.6666667  1.0000000
```

```
c2 <- cov2cor(cov2)
round(c2, 3)
```

```
##        [,1]   [,2]   [,3]   [,4]
## [1,]  1.000  0.500  0.408 -0.408
## [2,]  0.500  1.000  0.408 -0.408
## [3,]  0.408  0.408  1.000 -0.667
## [4,] -0.408 -0.408 -0.667  1.000
```

Playing with sign

We can change the sign at any edge of the graph. The change in the edge of parent to children is simpler to interpret, as we can see in the covariance/correlation from the two examples.

Let us consider the second example but change the sign between the parents and swap the sign in both terms of the second equation:

```
tree2b <- treepcor(
  p1 ~ -p2 + c1 + c2,
  p2 ~ -c3 + c4)
tree2b
```

```
## treepcor for 4 children and 2 parent variables
## p1 ~ - p2 + c1 + c2
## p2 ~ - c3 + c4
```

```
summary(tree2b)
```

```
##    p1 p2
## c1  1  0
## c2  1  0
## c3  0 -1
## c4  0  1
## p2 -1  0
```

This gives the precision matrix as

```
q2b <- prec(tree2b, theta = c(0, 0))
q2b
```

```
##    c1 c2 c3 c4 p1 p2
## c1  1  0  0  0 -1  0
## c2  0  1  0  0 -1  0
## c3  0  0  1  0  0  1
## c4  0  0  0  1  0 -1
## p1 -1 -1  0  0  4  1
## p2  0  0  1 -1  1  3
```

The covariance computed from the full precision (and the correlation) between children is the same as before

```
all.equal(solve(q2)[1:4, 1:4],
          solve(q2b)[1:4, 1:4])
```

```
## [1] TRUE
```

Therefore, allowing flexibility in an edge of parent to another parent is not useful and will only imply more complexity. Therefore we will not consider it in the `vcov` method. NOTE: The `vcov` of a `treepcor` does not takes into account the sing between parent variables! So, please use it with care.

References

Freni-Sterrantino, Anna, Denis Rustand, Janet van Niekerk, Elias T. Krainski, and Håvard Rue. 2025. "A Graphical Framework for Interpretable Correlation Matrix Models." *Statistical Methods & Applications*. https://doi.org/10.1007/s10260-025-00788-y.