MDiNE Vignette

Kevin McGregor

2019-05-29

Microbiome **Di**fferential **N**etwork **E**stimation (**mdine**) allows the estimation of OTU co-occurrence networks within two separate groups, where the networks are defined through precision matrices. The difference between the two precision matrices is also estimated, along with corresponding interval estimates. This work was developed in the Greenwood Lab at McGill University.

Installation

mdine uses the package **rstan** to sample the model parameters. The first step to installing **mdine** is to install rstan along with the appropriate compiler. Currently, **mdine** is only available to install through github. To install, run:

```
if (!require(devtools)) {
  install.packages("devtools")
  library(devtools)
}

install_github("kevinmcgregor/mdine", dependencies=TRUE)
```

The model

The goal of **mdine** is to estimate a precision matrix-based taxa co-occurrence network within two groups. Here we describe the basic structure of the model being estimated. For more information consult McGregor, Labbe, and Greenwood (2019). Assume $\mathbf{Y}$ is an $n \times (J+1)$ matrix of counts of $J+1$ taxa in n samples. $z_i \in \{0,1\}$ indicates which group individual i belongs to, and this is the covariate that the co-occurrence network will vary over. Also, K additional covariates can be included in the model and are contained in the $(n \times (K+1))$ design matrix $\mathbf{X}$.

$$
\begin{aligned}
\mathbf{Y}_{i\cdot}|p_{i\cdot}, \mathbf{B}, \mathbf{W}_{i\cdot}, \mathbf{\Sigma}_0^{-1}, \mathbf{\Sigma}_1^{-1}, \lambda &\sim \text{Multinomial}(M_i, p_{i\cdot}) \\
\mathbf{W}_{i\cdot}|\mathbf{B}, \mathbf{\Sigma}_0^{-1}, \mathbf{\Sigma}_1^{-1}, \lambda &\sim \text{Normal}\left((\mathbf{X}_{i\cdot}\mathbf{B})^\top, z_i\mathbf{\Sigma}_1 + (1-z_i)\mathbf{\Sigma}_0\right) \\
s_{jj'}^{(z)}|\lambda &\sim \text{Laplace}\left(0, \lambda\right) \\
s_{jj}^{(z)}|\lambda &\sim \text{Exponential}\left(\lambda/2\right) \\
\lambda &\sim \text{Exponential}\left(\widehat{\lambda}_{init}^{-1}\right) \\
\mathbf{B}_{kj} &\sim \text{Normal}\left(0, 10000\right),
\end{aligned} \tag{1}
$$

for each $i \in \{1,\dots,N\}$, $j \in \{1,\dots,J\}$, $j' \in \{1,\dots,j-1\}$, $k \in \{1,\dots,K+1\}$, and $z \in \{0,1\}$.

The "true" OTU proportions are parameterized as:

$$
\left[\log\left(\frac{p_{i1}}{p_{i(J+1)}}\right), \dots, \left(\frac{p_{iJ}}{p_{i(J+1)}}\right)\right] = \mathbf{W}_{i\cdot} \tag{2}
$$

The $(J+1)^{th}$ OTU is considered to be the reference category, and will not be included in the networks. This could be a single OTU, or it could be the sum of two or more OTUs, e.g. the sum of all remaining OTUs not to be included in the networks.

The parameters in the $(K+1) \times J$ matrix $\mathbf{B}$ explain the effects of the covariates on the taxa abundances. The co-occurrence networks for individuals with $z_i = 0$ and $z_i = 1$ are defined through the two precision matrices $\mathbf{\Sigma}_0^{-1}$ and $\mathbf{\Sigma}_1^{-1}$, respectively. The value λ controls the amount of sparsity in $\mathbf{\Sigma}_0^{-1}$ and $\mathbf{\Sigma}_1^{-1}$ (though in the Bayesian context, values will not be set *exactly* to zero).

Using the mdine package

Arguments

The required arguments of the **mdine** function are:

- Y - The OTU counts. The last column contains the counts of the reference category. Usually, this would be the sum of the OTU columns that are not to be included in the networks.
- X - The design matrix including a column of ones for the intercept
- Z - The binary variable you want the network to vary over. This variable can also be included in the design matrix.

Some other optional arguments are:

- lambda - The penalization parameter. If not specified, then the value of λ is estimated according to the above model.
- offset - Offset term to include in the model
- mc.cores - Number of cores to use in MCMC sampling
- iter - Number of MCMC iterations. By default, the first half will be used as warmup.
- quant - Vector (length 2) specifying lower and upper quantiles for credible intervals.

Example

We apply **mdine** on a dataset containing samples from Chron's patients and controls (Gevers et al. (2014)). The dataset included in this package contains a subset of only 100 samples from the original dataset. The data come in the form of a list, where the first list element contains the covariates, and the second element contains the counts for 5 families, and a 6th "reference" category containing the sum of all remaining families.

```
library(mdine)
#> Loading required package: Rcpp
data(crohns)

# Covariate data
head(crohns$covars)
#>   disease      age    sex antibiotic
#> 1      CD 12.00000   male      false
#> 2      CD 11.33333 female       true
#> 3      no 14.16667   male      false
#> 4      no  9.25000   male      false
#> 5      no 12.66667 female      false
#> 6      CD  7.25000 female      false
# OTU table
head(crohns$otu.counts)
#>   f__Bacteroidaceae f__Ruminococcaceae f__Lachnospiraceae
#> 1              4578               2158                576
#> 2             24538              16843               4741
```

```
#> 3             22654               5043               5831
#> 4               340                  3                375
#> 5              3916               1848               2738
#> 6             10989               1581               4144
#>   f__Enterobacteriaceae f__Pasteurellaceae   ref
#> 1                  9887                 29  1365
#> 2                   295                 38 15801
#> 3                    83                119  6769
#> 4                    11                  0    52
#> 5                    59                138  3503
#> 6                  1713                 15  9928
```

First we'll prepare the model matrix. We'll only include disease status and an intercept:

```
X <- model.matrix(~disease, data=crohns$covars)
head(X)
#>   (Intercept) diseaseCD
#> 1           1         1
#> 2           1         1
#> 3           1         0
#> 4           1         0
#> 5           1         0
#> 6           1         1
```

Next, we'll run **mdine**:

```
# Running mdine
md.fit <- mdine(Y=crohns$otu.counts, X=X, Z=X[,2], mc.cores=4, iter=1000)
```

Looking at the estimated precision matrices:

```
# Estimated precision matrix for control samples (Z=0):
md.fit$post_mean$invsigma0
#>             [,1]         [,2]        [,3]         [,4]        [,5]
#> [1,]  1.51466148 -0.027291376 -0.33619647 -0.069585457 -0.11736847
#> [2,] -0.02729138  0.747170756 -0.02365023  0.004370145 -0.04459465
#> [3,] -0.33619647 -0.023650231  1.45287832 -0.016222227 -0.03780328
#> [4,] -0.06958546  0.004370145 -0.01622223  0.244850679 -0.09648859
#> [5,] -0.11736847 -0.044594652 -0.03780328 -0.096488593  0.24594660

# Estimated precision matrix for Crohn's samples (Z=1):
md.fit$post_mean$invsigma1
#>             [,1]        [,2]        [,3]        [,4]        [,5]
#> [1,]  0.72578749 -0.33657376 -0.13886681 -0.01876870 -0.01435105
#> [2,] -0.33657376  1.04369220 -0.58663907  0.03107528 -0.02216813
#> [3,] -0.13886681 -0.58663907  1.04972381 -0.05029655  0.04004395
#> [4,] -0.01876870  0.03107528 -0.05029655  0.16940092 -0.07987567
#> [5,] -0.01435105 -0.02216813  0.04004395 -0.07987567  0.24277103
```

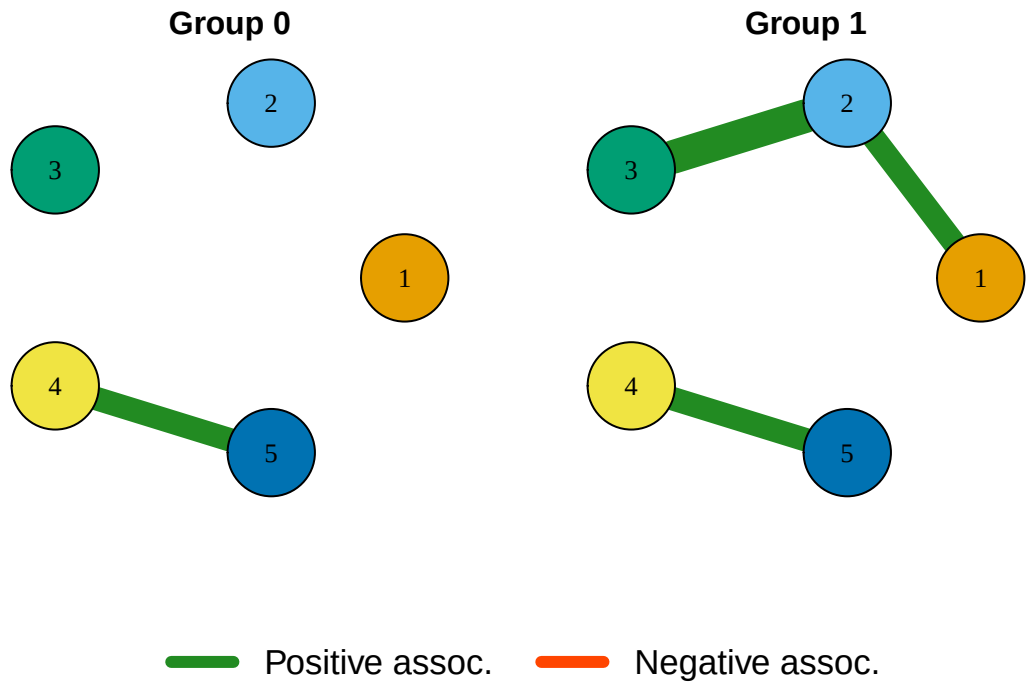
```
# Weighted adjacency matrices based on each precision matrix
adj <- ci2adj(md.fit, weighted = TRUE)
adj
#> $adj0
#>      [,1] [,2] [,3]      [,4]      [,5]
#> [1,]    0    0    0 0.0000000 0.0000000
#> [2,]    0    0    0 0.0000000 0.0000000
```

```
#> [3,]    0    0    0 0.0000000 0.0000000
#> [4,]    0    0    0 0.0000000 0.3931922
#> [5,]    0    0    0 0.3931922 0.0000000
#>
#> $adj1
#>           [,1]      [,2]      [,3]      [,4]      [,5]
#> [1,] 0.0000000 0.3867135 0.0000000 0.0000000 0.0000000
#> [2,] 0.3867135 0.0000000 0.5604634 0.0000000 0.0000000
#> [3,] 0.0000000 0.5604634 0.0000000 0.0000000 0.0000000
#> [4,] 0.0000000 0.0000000 0.0000000 0.0000000 0.3938748
#> [5,] 0.0000000 0.0000000 0.0000000 0.3938748 0.0000000
```

Plotting resulting networks

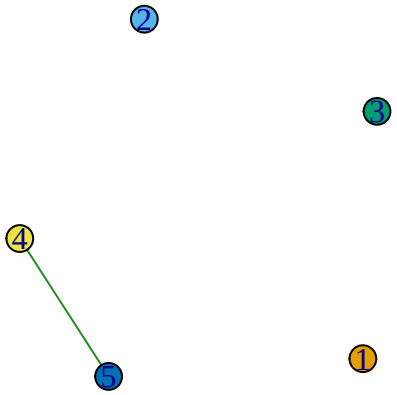
A function (with limited ability) is provided to plot the networks for the two groups based on which edges are "significant" according to the credible intervals caluclated in **mdine**.

```
# Plotting the two networks
plot_networks(md.fit)
```



The function *plot_networks()* is meant only as a way to quickly visualize the networks corresponding to two groups; its functionality is rather limited. However, this package also contains a function to convert a weighted adjacency matrix to an *igraph* object for use in more sophisticated figures using *plot.igraph()*.

```
# Weighted adjacency matrices based on each precision matrix
ig0 <- adj2ig(adj$adj0)
igraph::plot.igraph(ig0)
```



References

Gevers, Dirk, Subra Kugathasan, Lee A Denson, Yoshiki Vázquez-Baeza, Will Van Treuren, Boyu Ren, Emma Schwager, et al. 2014. "The Treatment-Naive Microbiome in New-Onset Crohn's Disease." *Cell Host & Microbe* 15 (3). Elsevier: 382–92.

McGregor, Kevin, Aurélie Labbe, and Celia MT Greenwood. 2019. "MDiNE: A Model to Estimate Differential Co-Occurrence Networks in Microbiome Studies." *bioRxiv*. Cold Spring Harbor Laboratory, 544122.