

get Markov Equivalent Class

Obtaining the Markov equivalence class using pcalg

A quick document to check if my version of R and pcalg (or other dependencies) is causing a problem. First I create a non-collider 1->2->3 dag structure and a collider structure 1->2<-3.

```
library(pcalg)

p <- 3
# non-collider
(dag_NC_mat <- matrix(c(0,0,0,1,0,0,0,1,0),p,p))
```

```
##      [,1] [,2] [,3]
## [1,]    0    1    0
## [2,]    0    0    1
## [3,]    0    0    0
```

```
dag_NC <- as(dag_NC_mat, "graphNEL")
# collider
(dag_C_mat <- matrix(c(0,0,0,1,0,1,0,0,0),p,p))
```

```
##      [,1] [,2] [,3]
## [1,]    0    1    0
## [2,]    0    0    0
## [3,]    0    1    0
```

```
dag_C <- as(dag_C_mat, "graphNEL")
```

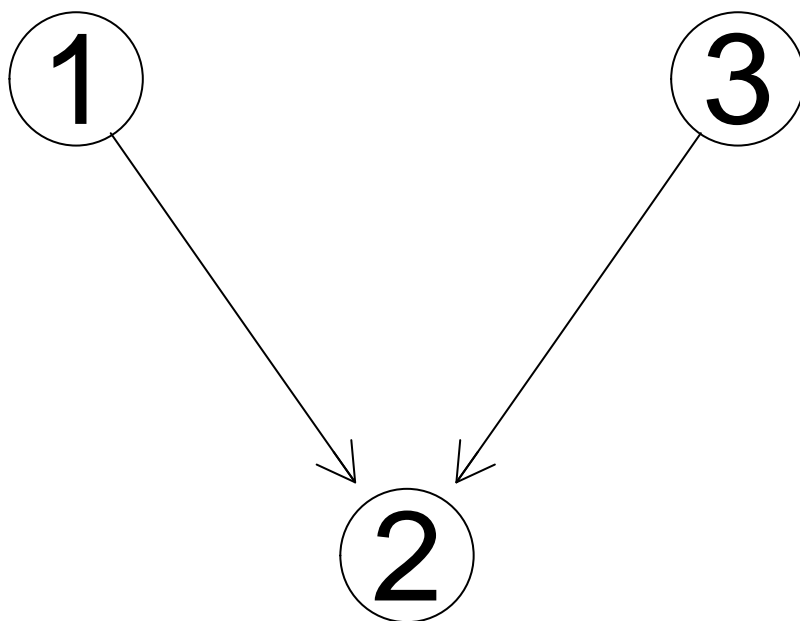
Lets look at the graphs:

```
plot(dag_NC)
```

```
##
## Attaching package: 'BiocGenerics'
##
## The following objects are masked from 'package:parallel':
##
##   clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##   clusterExport, clusterMap, parApply, parCapply, parLapply,
##   parLapplyLB, parRapply, parSapply, parSapplyLB
##
## The following objects are masked from 'package:stats':
##
##   IQR, mad, sd, var, xtabs
##
## The following objects are masked from 'package:base':
##
##   anyDuplicated, append, as.data.frame, cbind, colMeans,
##   colnames, colSums, do.call, duplicated, eval, evalq, Filter,
##   Find, get, grep, grepl, intersect, is.unsorted, lapply,
##   lengths, Map, mapply, match, mget, order, paste, pmax,
##   pmax.int, pmin, pmin.int, Position, rank, rbind, Reduce,
##   rowMeans, rownames, rowSums, sapply, setdiff, sort, table,
##   tapply, union, unique, unsplit, which, which.max, which.min
```



```
plot(dag_C)
```



Obtain cpdags corresponding to this generating dags, and then obtain all generated dags corresponding to cpdags.

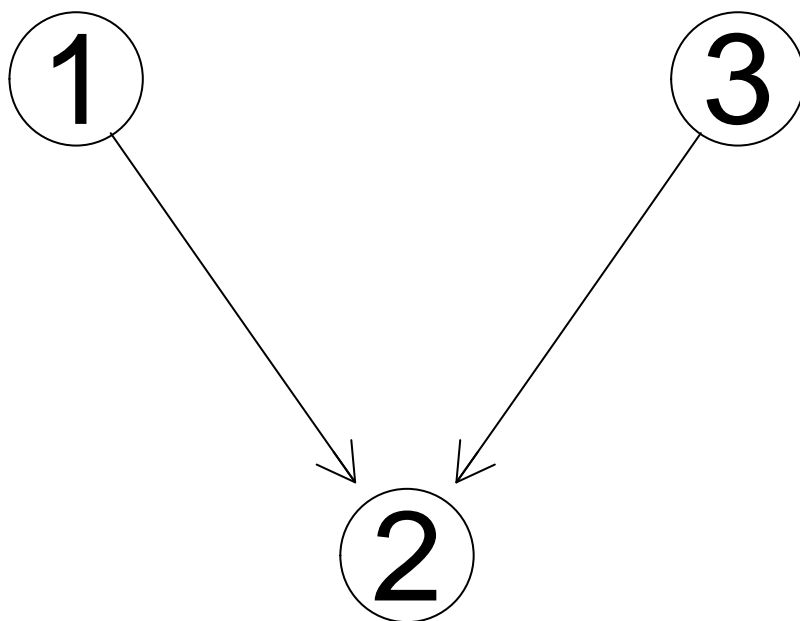
```
cpdag_NC <- dag2cpdag(dag_NC)
cpdag_C <- dag2cpdag(dag_C)
```

Lets plot the cpdags.

```
plot(cpdag_NC)
```



```
plot(cpdag_C)
```



Convert to matrices, obtain markov equivalent dags from each.

```

# non-collider
cpdag_NC_Mat <- as(cpdag_NC, "matrix")
(dags_NC_Mat <- pdag2allDags(t(cpdag_NC_Mat))$dags)

```

```

##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
## [1,]    0    1    0    0    0    1    0    0    0
## [2,]    0    1    0    0    0    0    0    1    0
## [3,]    0    0    0    1    0    0    0    1    0

```

```
dags_NC_aMat_1 <- matrix(dags_NC_Mat[1,],p,p,byrow=T)
dags_NC_aMat_2 <- matrix(dags_NC_Mat[2,],p,p,byrow=T)
dags_NC_aMat_3 <- matrix(dags_NC_Mat[3,],p,p,byrow=T)
# collider
cpdag_C_Mat <- as(cpdag_C, "matrix")
(dags_C_Mat <- pdag2allDags(t(cpdag_C_Mat))$dags)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
## [1,]    0    0    0    1    0    1    0    0    0

dags_C_aMat_1 <- matrix(dags_C_Mat[1,],p,p,byrow=T)
```

As is specified in ?pdag2allDags pdag2allDags returns an adjacency of type “amat.cpdag” which as is specified in ?amatType is coded so that $\text{mat}[a,b]=0$, $\text{mat}[b,a]=1$ implies $a \rightarrow b$, so that “from” of arrow is in columns and “to” in rows. This means we have to transpose before converting to graph.

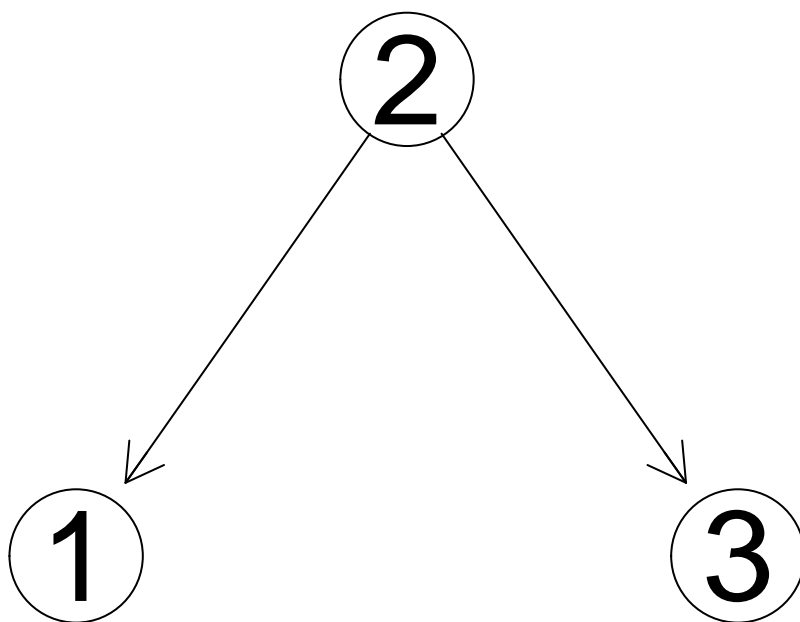
```
# non-collider
dags_NC_1 <- as(t(dags_NC_aMat_1), "graphNEL")
dags_NC_2 <- as(t(dags_NC_aMat_2), "graphNEL")
dags_NC_3 <- as(t(dags_NC_aMat_3), "graphNEL")
# collider
dags_C_1 <- as(t(dags_C_aMat_1), "graphNEL")
```

Lets plot non-collider generated dags.

```
plot(dags_NC_1)
```



```
plot(dags_NC_2)
```

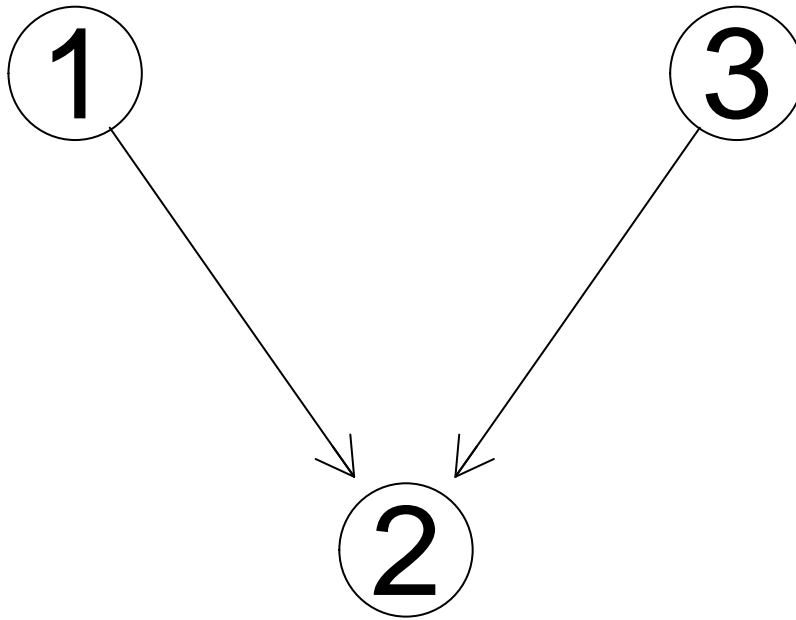


```
plot(dags_NC_3)
```



This is fine! Now lets plot collider generated dags.

```
plot(dags_C_1)
```



The collider structure was converted into a non-collider structure in the process.

This code, using the following packages, is returning a v-structure as one of the dags belonging to a Markov Equivalence class that was generated from a non-v-structure.

```
sessionInfo()
```

```
## R version 3.4.3 (2017-11-30)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 16.04.1 LTS
##
## Matrix products: default
## BLAS: /home/soulivanh/R-3.4.3/lib/libRblas.so
## LAPACK: /home/soulivanh/R-3.4.3/lib/libRlapack.so
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8      LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8      LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8  LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8     LC_NAME=C
##  [9] LC_ADDRESS=C             LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] grid      parallel  stats      graphics  grDevices  utils      datasets
## [8] methods  base
##
## other attached packages:
## [1] Rgraphviz_2.22.0    graph_1.56.0      BiocGenerics_0.24.0
## [4] pcalg_2.6-0
##
## loaded via a namespace (and not attached):
## [1] igraph_1.2.2      Rcpp_1.0.0        cluster_2.0.6
## [4] knitr_1.20        magrittr_1.5      MASS_7.3-48
## [7] sfsmisc_1.1-1     clue_0.3-54       ggm_2.3
```

```

## [10] dagitty_0.2-2      stringr_1.3.1      tools_3.4.3
## [13] corpcor_1.6.9      htmltools_0.3.6    abind_1.4-5
## [16] yaml_2.2.0         RBGL_1.54.0        rprojroot_1.3-2
## [19] digest_0.6.18      bdsmatrix_1.3-3     curl_3.2
## [22] robustbase_0.93-3  evaluate_0.11       rmarkdown_1.10
## [25] V8_1.5             fastICA_1.2-1       stringi_1.2.4
## [28] compiler_3.4.3     DEoptimR_1.0-8      backports_1.1.2
## [31] boot_1.3-20        stats4_3.4.3        jsonlite_1.5
## [34] pkgconfig_2.0.2

```