Typical workflow for using the sparse.inv.cov R library

It is recommended that the reader runs the code here in R and also consults the help files in the library.

1. Get data

For illustrative purposes here we simulate some data with known structure

```
#generate sigma.inv
sinv<-diag(5)/2
sinv[1,2]=sinv[2,3]=sinv[3,4]=sinv[4,5]=0.1
sinv<-sinv+t(sinv)
sinv<-as(sinv,"dsCMatrix")
# generate data with mean zero and inverse covariance matrix sinv
res<-Simulate(sigma.inv=sinv,M=NULL,nsamp=1000,nsim=1,seed=1)
X<-res$data[[1]]
# mean correct each column
X<-scale(X,center=TRUE,sc=FALSE)
# more generally regress each column of X on the design matrix and replace it with
# the residuals

# check means
colMeans(X)
# check structure in observed sample inverse
solve(crossprod(X)/1000)
# true inverse
sinv
```

2. Find zero pattern

```
# there can be at most 4 neighbours for each variable here
# generally want m/n <1/20 where m is the number of neighbours of a variable
res<-get.neighbs(X,kmax=4)
# for a discussion of the parameter kmax see Kiiveri(2011).
# example of splitting  the job over two processors to speed things up
# on processor 1
res1<-get.neighbs(X,kmax=4,cols=1:2)
# save(res1,file="mydir/res1.RData")
#on processor 2
res2<-get.neighbs(X,kmax=4,cols=3:5)
# save(res2,file="mydir/res2.RData")
# then combine results
# load("mydir/res1.RData")
# load("mydir/res2.RData")
# get incidence matrix for all variables
a00<-res1$a00+res2$a00  # similar results apply for a05 a01 etc
# the nonzero entries in a00 could be reset to 1 but typically it doesn't matter for
# further processing
# the generalisation to more than 2 processors should be clear
```

3. Fit covariance selection model using default parameters

```
# use res$a00 here (the usual BIC) because n is much greater than p. When p>>n use
# res$a01 from the previous step.
tmp<-hd.covsel(X, res$a00, nsamp=nrow(X),corr=FALSE)
# look at fitted inverse matrix
tmp$sigma.inv
# look at fitted covariance matrix in positions where sigma.inv is not zero
tmp$sigma
```

Reference

Kiiieri(2011) Multivariate analysis of microarray data: differential expression and differential connection. BMC Bioinformatics