Reproducing the fitted inverse covariance matrix used in the paper

1. **Do the following to create the R workspace smoking.RData.**

```
# The data used to illustrate the method implemented in mvama was discussed in:
# Spira, A. et al. (2004). Effects of cigarette smoke on the human
# airway epithelial cell transcriptome. Proc. Natl Acad. Sci. USA,
# 101, 10143-10148.

# The raw data may be obtained from the NCBI at
# http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE994. Only CEL
# files corresponding to 'smokers' and 'never smokers were used in the
# paper; these correspond to Accession IDs GSM15684-15717
# and GSM15718-15740, respectively.

# Once the compressed CEL files (GSM15XYZ.CEL.gz) have been downloaded
# to a directory, they can be read in and rma-corrected as follows:

# Install the following Bioconductor (http://www.bioconductor.org)
# packages if required:
require(affy)
require(limma)
require(hgu133acdf)

data <- ReadAffy(celfile.path = "/path_to_CEL_files", compress = TRUE)

# Carry out rma correction
rma.data <- rma(data)

# Note that this is equivalent to using the function 'expresso' as
# follows:
#rma.data <- expresso(data,
#                   bgcorrect.method = "rma",
#                   normalize.method = "quantiles",
#                   pmcorrect.method = "pmonly",
#                   summary.method = "medianpolish")

# And finally,
rma.data <- t(exprs(rma.data))
# which gives a 57 x 22283 matrix where the first 34 rows correspond
# to 'smokers', and the next 23 correspond to 'never
# nmokers'. Consequently, the vector of class labels y can be
# generated as
y <- rep(1:2, times = c(34, 23))
# generate the design matrix
D<-matrix(0,nrow=57,ncol=2)
D[,1]<-as.numeric(y==2)
D[,2]<-as.numeric(y==1)
```

```
# save to smoking.RData
x<-rma.data
save(x,y,D,file=”smoking.RData”)
```

The first 5 rows and columns of x should be

```
     X1007_s_at X1053_at  X117_at  X121_at X1255_g_at
[1,]   9.729530 4.169440 5.599752 8.113206   4.256429
[2,]  10.172280 4.381633 5.166447 8.025542   3.807625
[3,]   9.801157 4.335081 5.696982 8.079880   3.982507
[4,]  10.084170 4.025541 5.193149 7.842162   3.951598
[5,]   9.765211 4.142204 5.378281 7.858246   3.582457
```

2**. Compute the residual matrix after removing mean structure.**

```
load(“smoking.RData”)
# get residual matrix
R<-lm(x~D-1)$residuals
# save R in smoking.RData
save(x,y,R,file=’’smoking.RData”)
```

3. **Determine zero pattern**

```
library(Matrix)
library(spars.inv.cov)
# single run if you have the time and patience
res<-get.neighbs(R,kmax=3)
a<-res$a01 # use modified BIC with g=1

# or split it over more processors, in this case 5

# Source the scripts nb.script1.r to nb.script5.r on separate processors.
# These scripts may need to be edited so that the load and save statements
# read and write from a specified directory. After these jobs have finished, running R
#  in the common directory and sourcing the script nb.collate.r produces the list
#  bres10. From this set
# a<-bres10$a01
```

4. **Fit the inverse covariance matrix**

```
res<-hd.covsel(R,a,nsamp=57,corr=TRUE,eps=.01,m=200)
# depending on you processor speed the fit could take between 1 and 2 days
# larger values of m might reduce this

#compute inverse covariance matrix from fitted inverse correlation matrix
# assumes residual matrix R is available as well as Matrix library
sd<-apply(R,2,var)*56/57
sd<-sd^0.5
d<-Diagonal(length(sd),1/sd)
sinv<-d%*%res$si%*%d
```