Generating correlation matrix

Here I will explain how to use the functions

```
generateCorrelationMatrix(M,type,alphad=0.5,phi=0.5,sparse=F,thres=1e-4, delta = rep(1,M) )
generateCoVarianceMatrix(corMatrix, sigmas =  rep(1,ncol(corMatrix)))
```

where - M number of covariates - type "random" to generate a random correlation matrix using the onion method with parameter alphad. "AR" to generate an autorregressive correlation matrix AR(1), with entries $\frac{1}{1-\phi^2}\phi^{|\delta_{ij}|}$ - alphad parameter for the onion method, the smaller, the more correlated the variables - phi parameter for the AR matrix, the higher, the more correlated the entries - sparse if TRUE, entries of the matrix bellow the threshold are set to zero. - thres threshold for which lesser values are set to zero. - delta for the AR entries.

Examples

Here alphad is equal to 100, that means low correlation between variables

```
require(corrplot)
```

```
## Loading required package: corrplot

## Warning: package 'corrplot' was built under R version 3.4.2

## corrplot 0.84 loaded
```

```
source('./generateCorrelation.R')
```

```
## Loading required package: clusterGeneration

## Loading required package: MASS
```

```
corrplot(generateCorrelationMatrix(20,"random",alphad=100),tl.pos = "n")
```

Here alphad equals 0.001 that means high correlation between variables

```
corrplot(generateCorrelationMatrix(20,"random",alphad=0.001),tl.pos = "n")
```

Now we generate an AR(1) correlation matrix

```
corrplot(generateCorrelationMatrix(20,"AR"),tl.pos = "n")
```

We change phi to have more correlation

```
corrplot(generateCorrelationMatrix(20,"AR",phi = 0.9),tl.pos = "n")
```

We can also induce a block structure in the AR model by interwinding some delta > 1 at some points

```
block_delta <- c(rep(1,4),3,rep(1,9),2,rep(1,5)) # now you have steps of size 3 at pos. 5, and 2 at pos
corrplot(generateCorrelationMatrix(20,"AR",phi=0.8 ,delta=block_delta),tl.pos = "n")
```

Generating covariates

First we can generate a Covariance matrix from the correlation matrix, we can set the vector of standard deviations or use the default value of 1 for all covariates

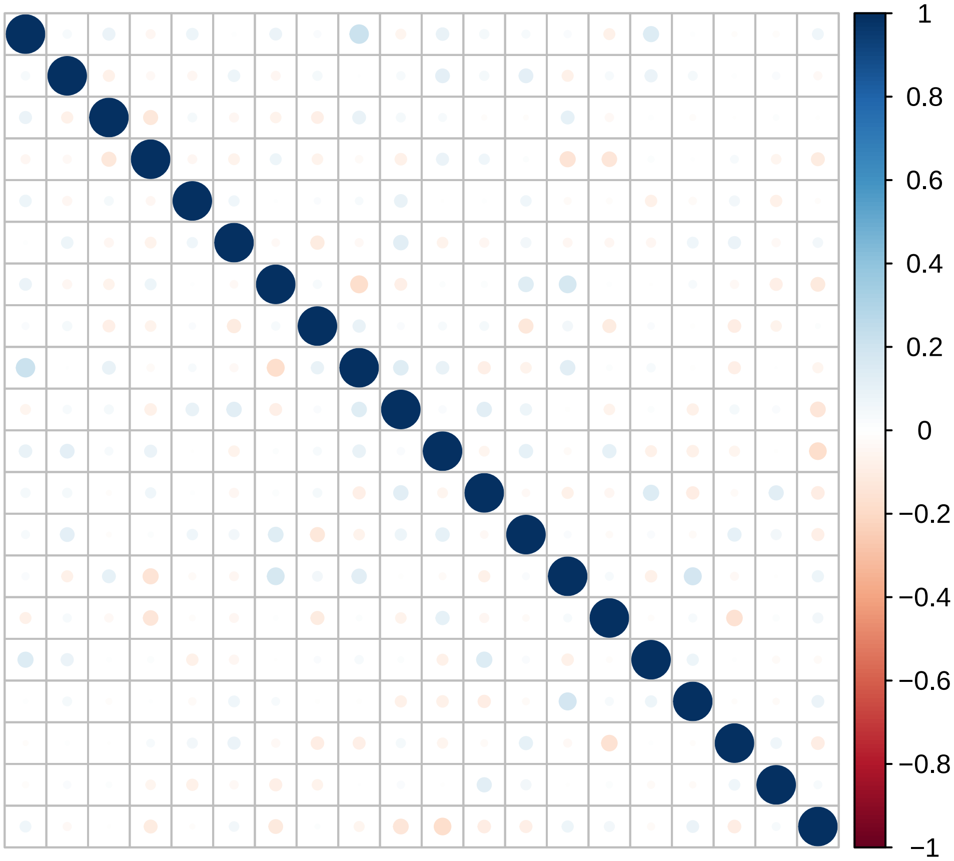

Figure 1: correlation matrix random alphad=100

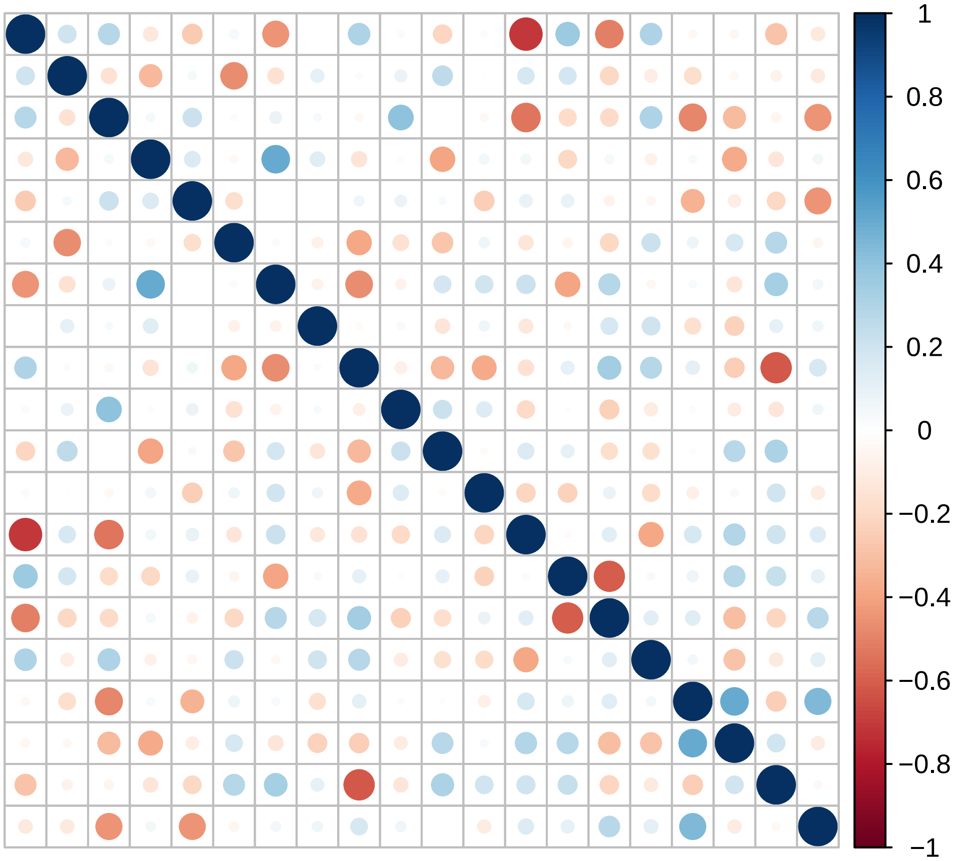

Figure 2: correlation matrix random alphad=0.001

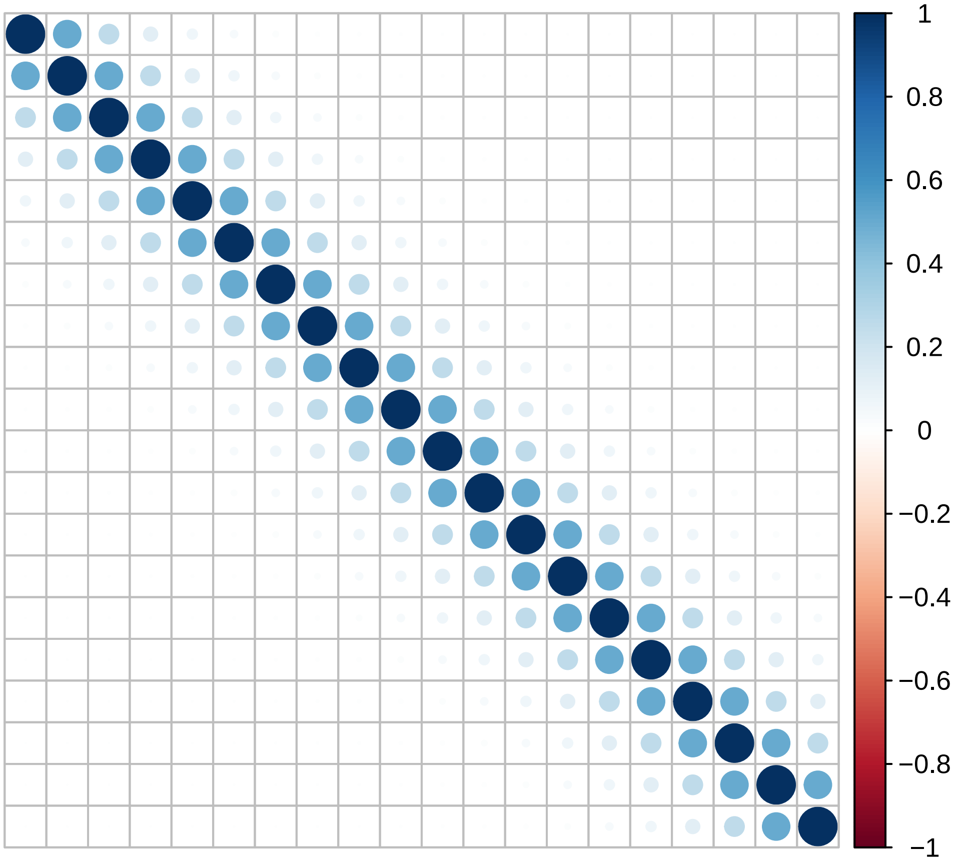

Figure 3: Correlation matrix AR

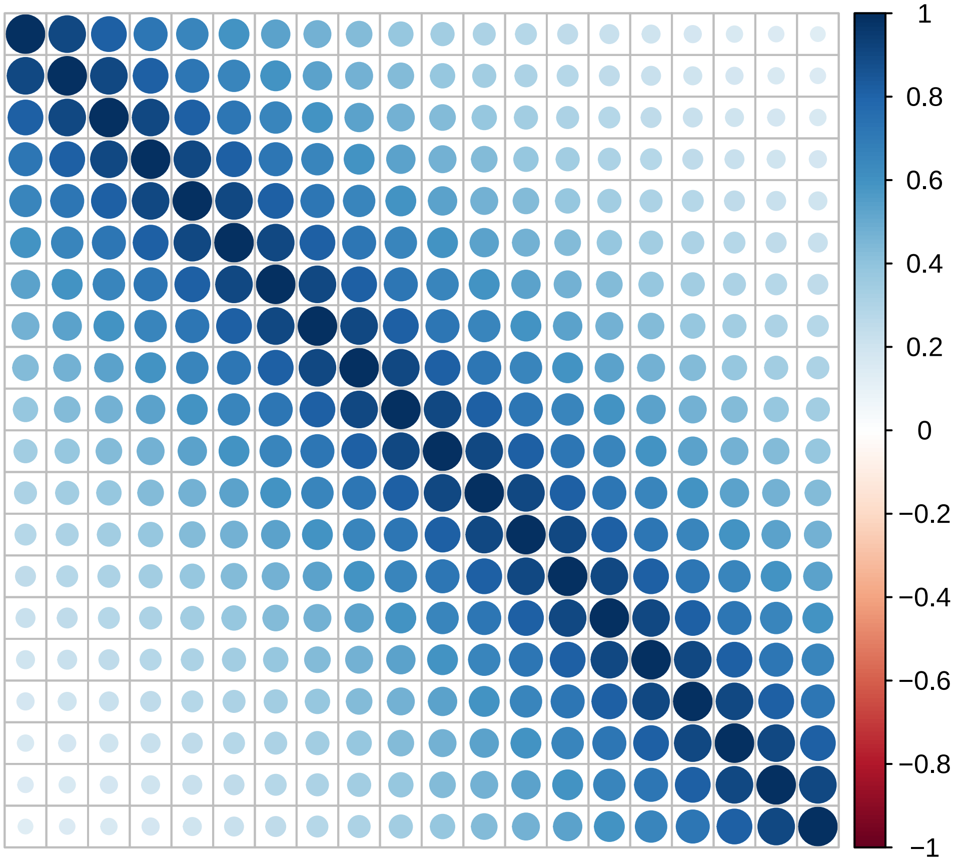

Figure 4: correlation matrix AR phi = 0.9

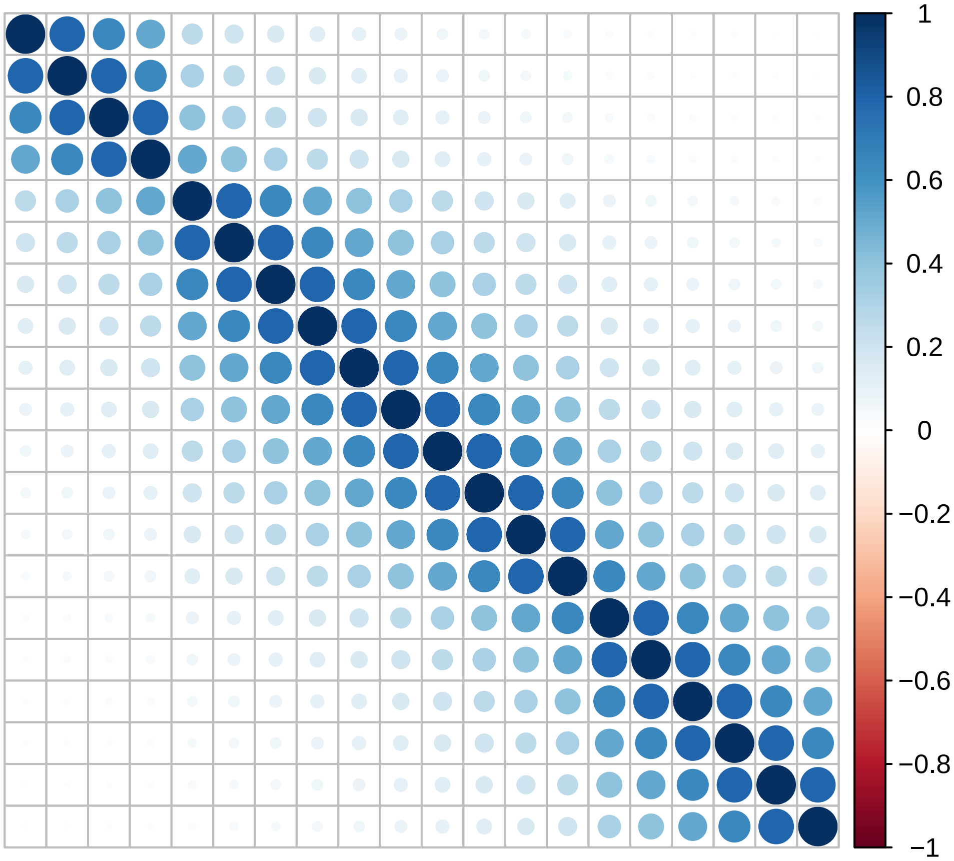

Figure 5: correlation matrix AR with block structure

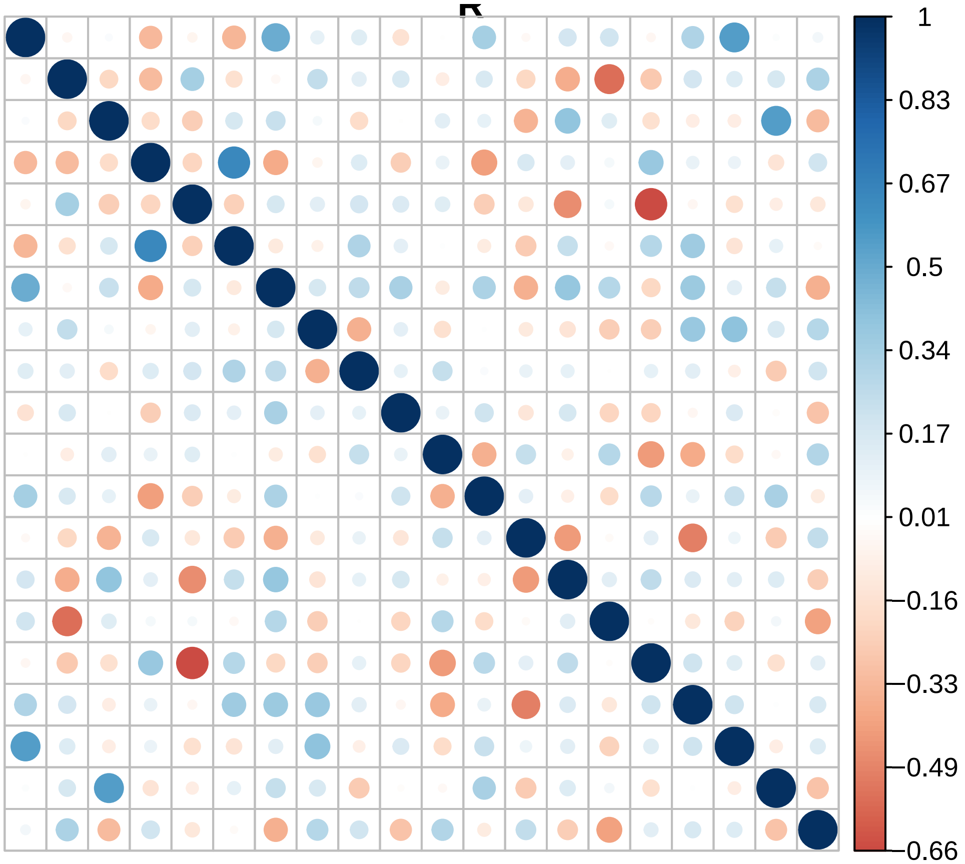

Figure 6: correlation and covariance matrix with column sds =1

```
R <- generateCorrelationMatrix(20,"random",alphad=0.001)
Sigma <- generateCoVarianceMatrix(R)
corrplot(R,is.corr = F,tl.pos = "n",title = "R")
```

```
corrplot(Sigma,is.corr = F,tl.pos = "n",title = "Sigma")
```

Now we can generate a set of covariates by drawing N samples from a multivariate normal, thus having our design matrix X

```
require(MASS)
M <- 20
N <- 10

R <- generateCorrelationMatrix(M,"random",alphad=0.001)
Sigma <- generateCoVarianceMatrix(R)
X <- mvrnorm(n = N, rep(0,M), Sigma)
corrplot(Sigma,is.corr = F, tl.pos = F, title = "Sigma")
```

We are going to plot the scatter matrix X^tX/N for different sample sizes, and scaled and unscaled values

```
par(mfrow=c(2,2))
corrplot( t(X) %*% X/N, is.corr=F, tl.pos =F, title = "un-scaled N=10")
corrplot( t(scale(X)) %*% scale(X)/N, is.corr=F, tl.pos =F, title = "scaled N=10")
```

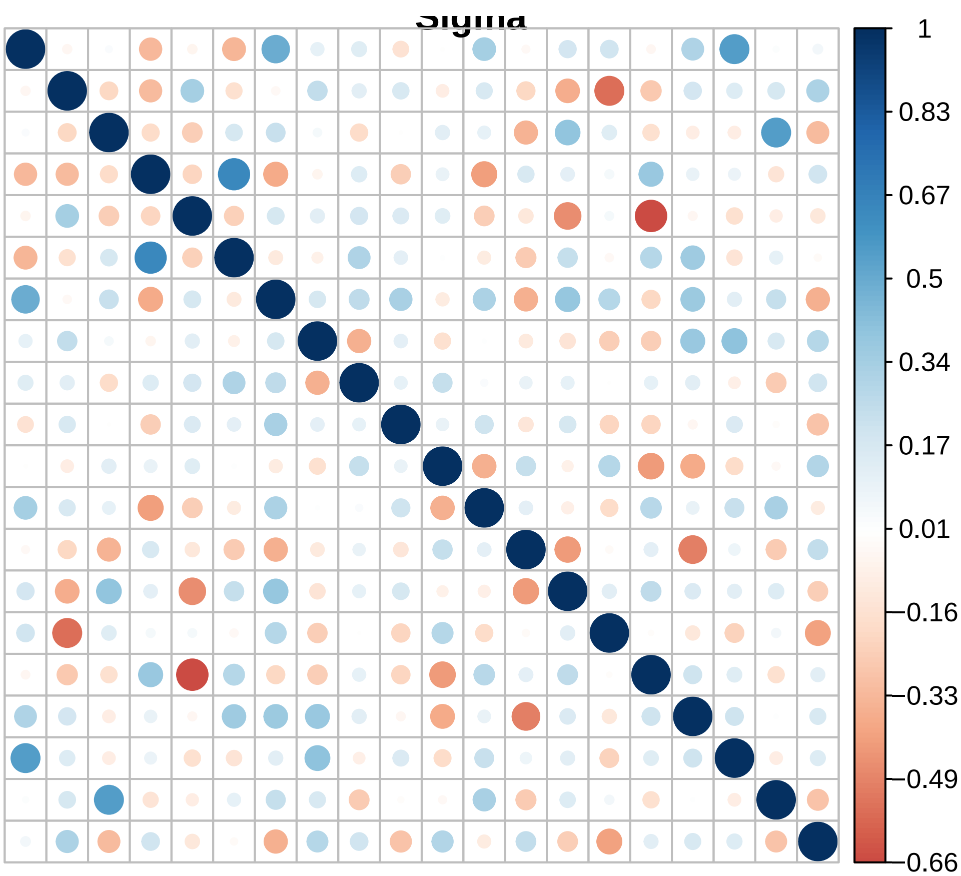

Figure 7: correlation and covariance matrix with column sds =1

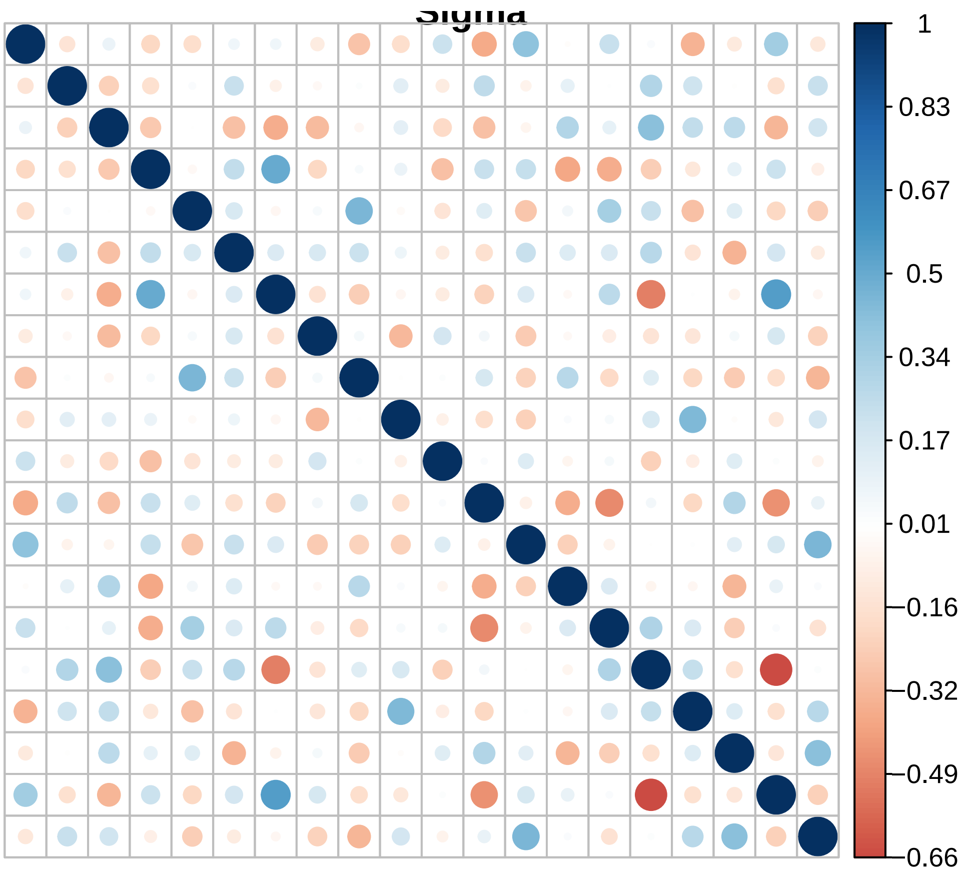

Figure 8: original covariance matrix

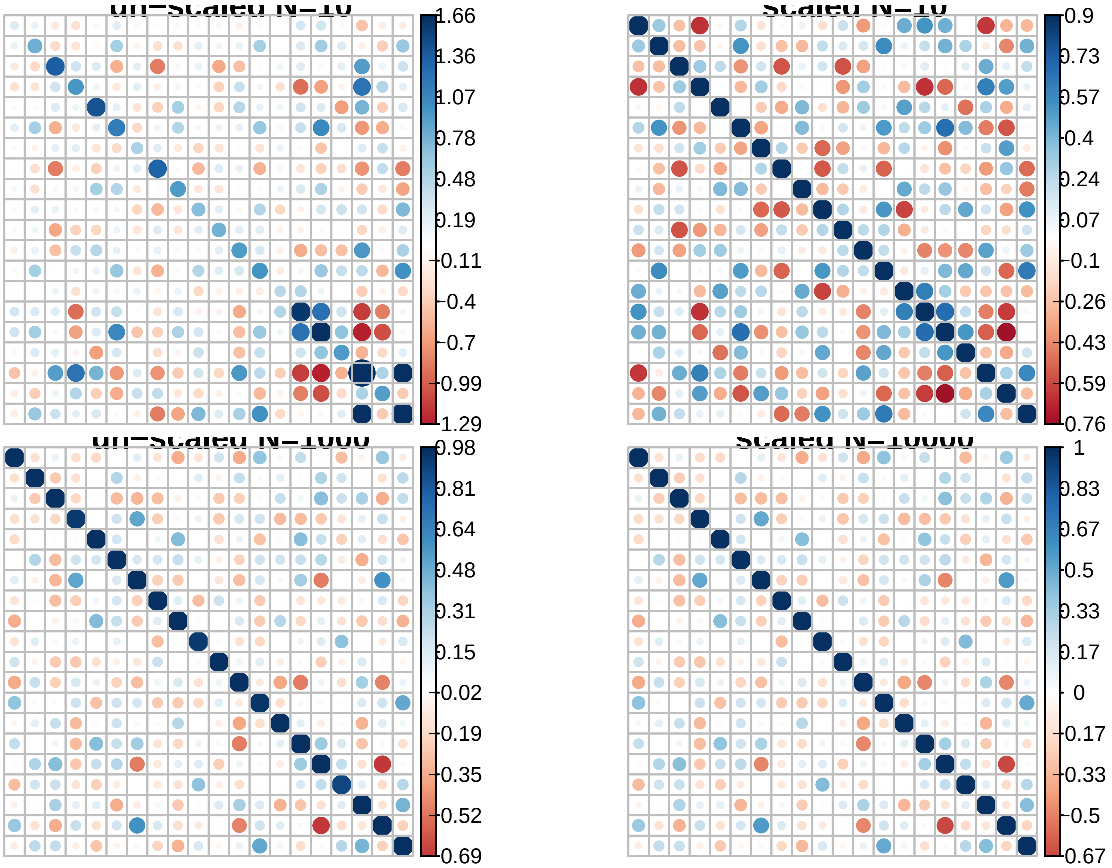

Figure 9: XtX for scaled and unscaled matrix X for N=10 and N=1000

```
N<-1000
X <- mvrnorm(n = N, rep(0,M), Sigma)
corrplot( t(X) %*% X/N, is.corr=F, tl.pos =F, title ="un-scaled N=1000")
corrplot( t(scale(X)) %*% scale(X)/N, is.corr=F, tl.pos =F, title = "scaled N=10000")
```

Now for an AR(1) process

```
M <- 20
N <- 10
block_delta <- c(rep(1,4),3,rep(1,9),2,rep(1,5))
R <- generateCorrelationMatrix(M,"AR",phi = 0.8 ,delta = block_delta)
par(mfrow=c(1,1))
Sigma <- generateCoVarianceMatrix(R)
corrplot(Sigma,is.corr = F, tl.pos = F,title = "Sigma")
```

```
X <- mvrnorm(n = N, rep(0,M), Sigma)
par(mfrow=c(2,2))

corrplot( t(X) %*% X/N, is.corr=F, tl.pos =F, title = "un-scaled N=10")
corrplot( t(scale(X)) %*% scale(X)/N, is.corr=F, tl.pos =F, title = "scaled N=10")

N<-1000
X <- mvrnorm(n = N, rep(0,M), Sigma)
corrplot( t(X) %*% X/N, is.corr=F, tl.pos =F, title ="un-scaled N=1000")
```

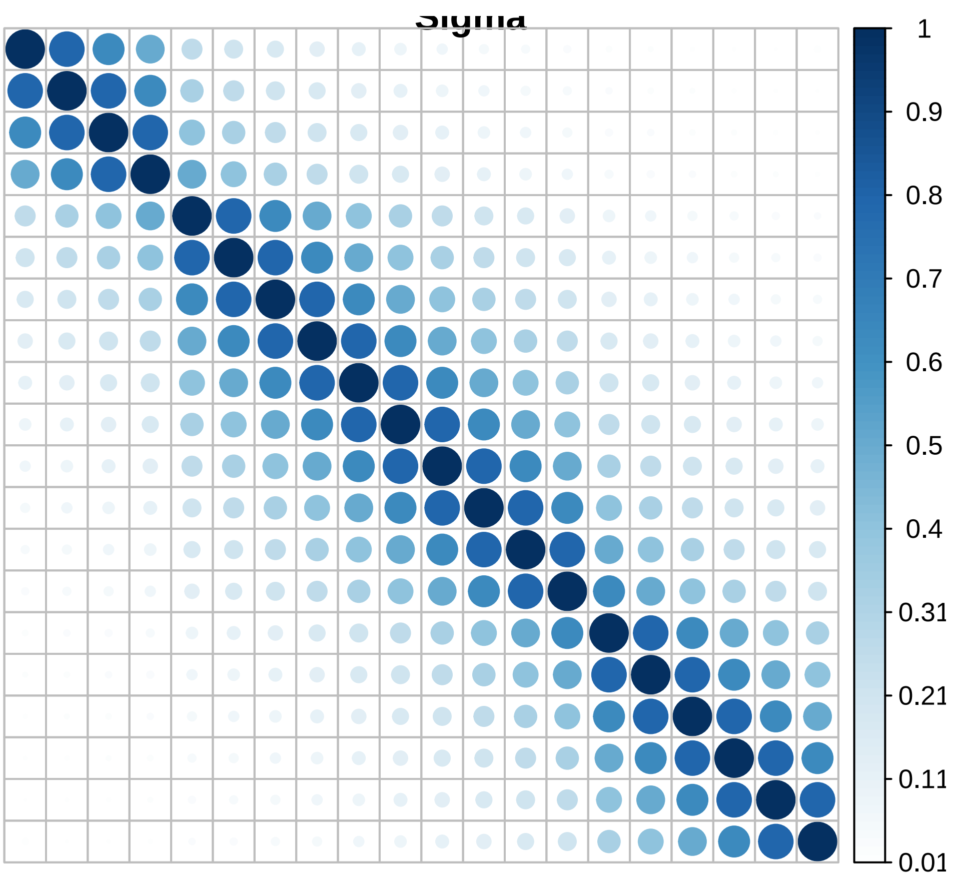

Figure 10: original covariance matrix

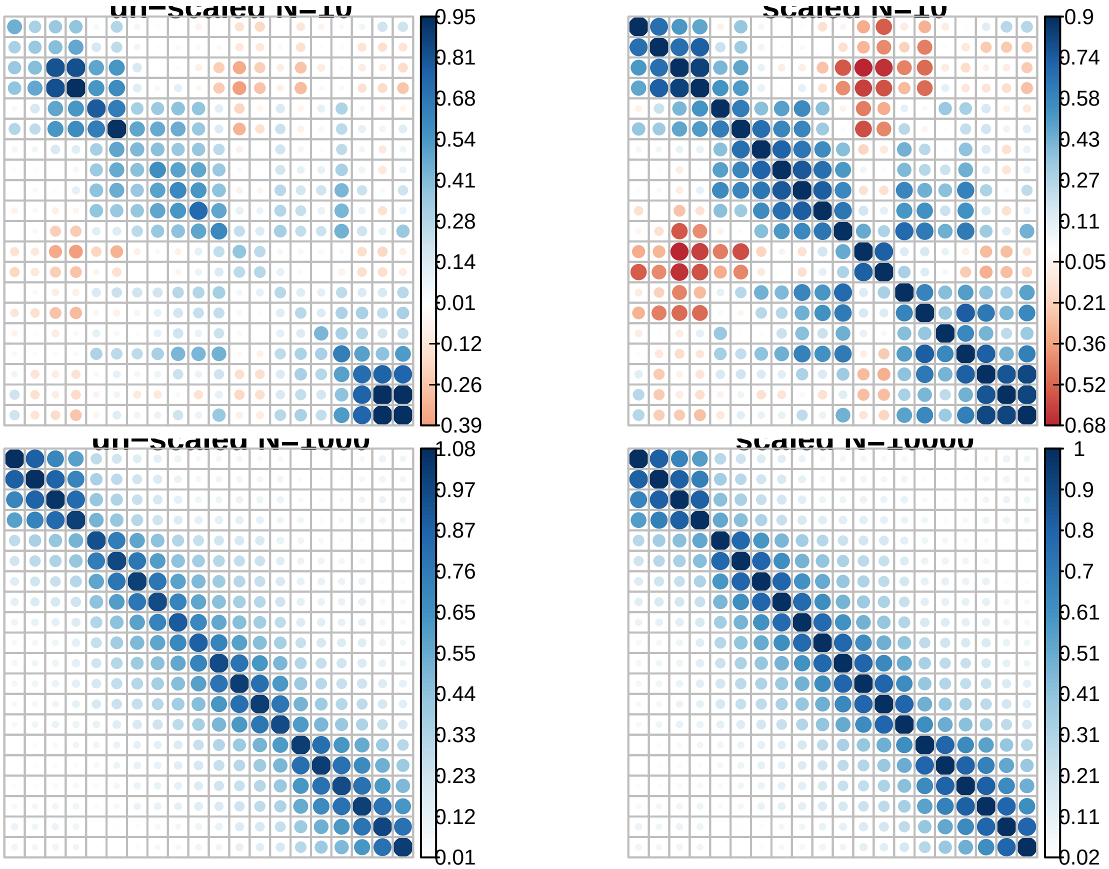

Figure 11: Sigma, XtX for scaled and unscaled matrix X for N=10 and N=1000

```
corrplot( t(scale(X)) %*% scale(X)/N, is.corr=F, tl.pos =F, title = "scaled N=10000")
```

```
dev.off()
```

```
## null device
##           1
```