```
## Load libraries
library(splines)
library(MASS)
library(mvtnorm)

## Warning:  package 'mvtnorm' was built under R version 3.3.2

library(Matrix) ##for the bdiag function
library(doParallel) ##to make cluster (on Windows)

## Warning:  package 'doParallel' was built under R version 3.3.2
## Loading required package:  foreach
## Loading required package:  iterators
## Loading required package:  parallel

library(foreach) ##to use foreach function that does the parallel processing
library(doRNG) ##for reproducible seeds when doing parallel processing

## Loading required package:  rngtools
## Loading required package:  pkgmaker
## Loading required package:  registry
##
## Attaching package:  'pkgmaker'
## The following object is masked from 'package:base':
##
##     isNamespaceLoaded

## Define the number of tests
ntest <- 1000

## Set nuber of simulations
nSims <- 10000

##second shape parameter for beta distribution
shape2 <- 2
```

Function to generate p-values from correlated binary data + helper function:

```
##this is based on ra2ba from the bindata package, but returns a logical rather than a numer
ra2baLogic <- function (x)
{
  retval <- x > 0
  dim(retval) <- dim(x)
  retval
}
```

```
##function to generate p-values
genPvals <- function(pi0, shape2, Sigma)
{
  ntest <- length(pi0)

  nullIlist <- rmvnorm(1, qnorm(pi0), Sigma)
  nullI <- ra2baLogic(nullIlist)[1,]

  ##nullI <- rbinom(ntest,prob=pi0,size=1)> 0

  pValues <- rep(NA,ntest)
  pValues[nullI] <- runif(sum(nullI))
  pValues[!nullI] <- rbeta(sum(!nullI),1,shape2)

  pValues
}
```
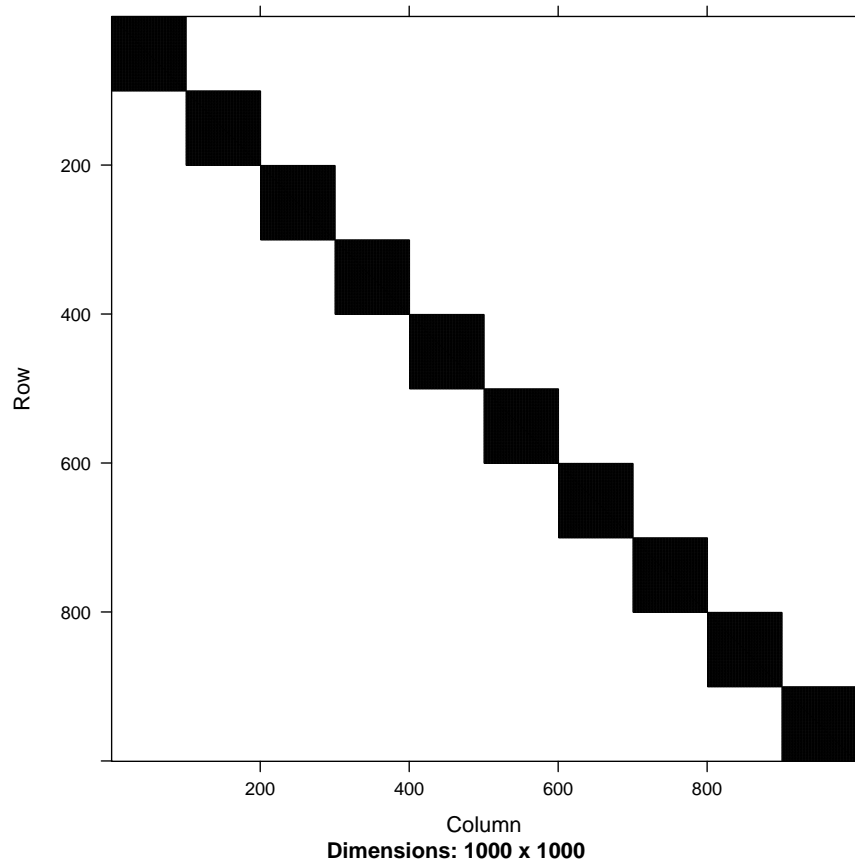
Create block diagonal correlation matrix:

```
##create correlation matrix with 10 blocks of 100 rvs
rho <- 0.9
sizeBlock <- 100
nrBlocks <- ntest/sizeBlock

block <- matrix(rho, sizeBlock, sizeBlock)
diag(block) <- 1
blockList <- list()
for(i in 1:nrBlocks)
{
  blockList[[i]] <- block
}
Sigma <- bdiag(blockList)
image(Sigma)
Sigma <- as.matrix(Sigma)
```

Row

200   400   600   800

Column

Dimensions: 1000 x 1000

1 Probability of being a false positive as a linear function of time

```
## Set up the time vector and the probability of being null
tme <- seq(-1,2,length=ntest)
pi0 <- 1/4*tme+1/2

cl<-makeCluster(8) ##specify number of cores less than or equal to number of cores on your o
registerDoParallel(cl)

set.seed(1345)

pValuesSims <- foreach(sim=1:nSims, .packages=c("Matrix","mvtnorm")) %dorng% {
  genPvals(pi0, shape2, Sigma)
```

```
}

##close the cluster
stopCluster(cl)

length(pValuesSims)

## [1] 10000

length(pValuesSims[[1]])

## [1] 1000

##save results
save(file="simResults_1.RData",
     list=c("pValuesSims"))
```

2   Probability of being a false positive as a smooth function of time

```
set.seed(1345)

## Set up the time vector and the probability of being null
tme <- seq(-1,2,length=ntest)
pi0 <- pnorm(tme)

splineMat <- ns(tme,df=3)

cl<-makeCluster(8) ##specify number of cores less than or equal to number of cores on your
registerDoParallel(cl)

set.seed(1345)

pValuesSims <- foreach(sim=1:nSims, .packages=c("Matrix","mvtnorm")) %dorng% {
  genPvals(pi0, shape2, Sigma)
}

##close the cluster
stopCluster(cl)

length(pValuesSims)

## [1] 10000
```

```r
length(pValuesSims[[1]])

## [1] 1000

##save results
save(file="simResults_2.RData",
     list=c("pValuesSims"))
```

3 Probability of being a false positive as a sine + step function

```r
set.seed(1345)

## Set up the time vector and the probability of being null
tme1 <- seq(-1*pi,2*pi,length=ntest)
tme2 <- rep(1:0, each=ntest/2)
pi0 <- 1/4*sin(tme1) + tme2/4 + 1/2
range(pi0)

## [1] 0.2500028 0.9999972

splineMat3 <- cbind(ns(tme1,df=3), tme2)
splineMat20 <- cbind(ns(tme1,df=20), tme2)

cl<-makeCluster(8) ##specify number of cores less than or equal to number of cores on your
registerDoParallel(cl)

set.seed(1345)

pValuesSims <- foreach(sim=1:nSims, .packages=c("Matrix","mvtnorm")) %dorng% {
  genPvals(pi0, shape2, Sigma)
}

##close the cluster
stopCluster(cl)

length(pValuesSims)

## [1] 10000

length(pValuesSims[[1]])

## [1] 1000

save(file="simResults_3.RData",
     list=c("pValuesSims"))
```

Session info:

```
devtools::session_info()

## Session info ------------------------------------------------

##  setting  value
##  version  R version 3.3.1 (2016-06-21)
##  system   x86_64, mingw32
##  ui       RTerm
##  language (EN)
##  collate  English_United States.1252
##  tz       America/New_York
##  date     2017-01-01

## Packages ----------------------------------------------------

##  package    * version date       source
##  codetools    0.2-14  2015-07-15 CRAN (R 3.3.1)
##  devtools     1.12.0  2016-06-24 CRAN (R 3.3.1)
##  digest       0.6.9   2016-01-08 CRAN (R 3.3.1)
##  doParallel * 1.0.10  2015-10-14 CRAN (R 3.3.2)
##  doRNG      * 1.6     2014-03-07 CRAN (R 3.3.1)
##  evaluate     0.10    2016-10-11 CRAN (R 3.3.2)
##  foreach    * 1.4.3   2015-10-13 CRAN (R 3.3.1)
##  highr        0.6     2016-05-09 CRAN (R 3.3.1)
##  iterators  * 1.0.8   2015-10-13 CRAN (R 3.3.0)
##  knitr      * 1.15.1  2016-11-22 CRAN (R 3.3.2)
##  lattice      0.20-33 2015-07-14 CRAN (R 3.3.1)
##  magrittr     1.5     2014-11-22 CRAN (R 3.3.1)
##  MASS       * 7.3-45  2016-04-21 CRAN (R 3.3.1)
##  Matrix     * 1.2-6   2016-05-02 CRAN (R 3.3.1)
##  memoise      1.0.0   2016-01-29 CRAN (R 3.3.1)
##  mvtnorm    * 1.0-5   2016-02-02 CRAN (R 3.3.2)
##  pkgmaker   * 0.22    2014-05-14 CRAN (R 3.3.1)
##  registry   * 0.3     2015-07-08 CRAN (R 3.3.1)
##  rngtools   * 1.2.4   2014-03-06 CRAN (R 3.3.1)
##  stringi      1.1.1   2016-05-27 CRAN (R 3.3.0)
##  stringr      1.0.0   2015-04-30 CRAN (R 3.3.1)
##  withr        1.0.2   2016-06-20 CRAN (R 3.3.1)
##  xtable       1.8-2   2016-02-05 CRAN (R 3.3.1)
```