GenePair: Statistical Methods for Modeling Spatially-Referenced Paired Genetic Relatedness Data

Patristic_Example

[1] Simulate data from the proposed model:

- Setting the reproducibility seed and initializing packages for data simulation:

```
set.seed(9722)

library(GenePair)
library(mnormt)  #Multivariate normal distribution
```

- Setting the global data values:

```
n<-100  #Number of individuals
n_star<-0  #Number of paired responses
for(j in 1:(n - 1)){
   for(k in (j + 1):n){
      n_star<-n_star +
             1
      }
   }

#Unique locations (one for each individual)
locs<-matrix(runif((2*n),
                   min = -1.00,
                   max =  1.00),
             nrow = n,
             ncol = 2)
plot(locs)
```
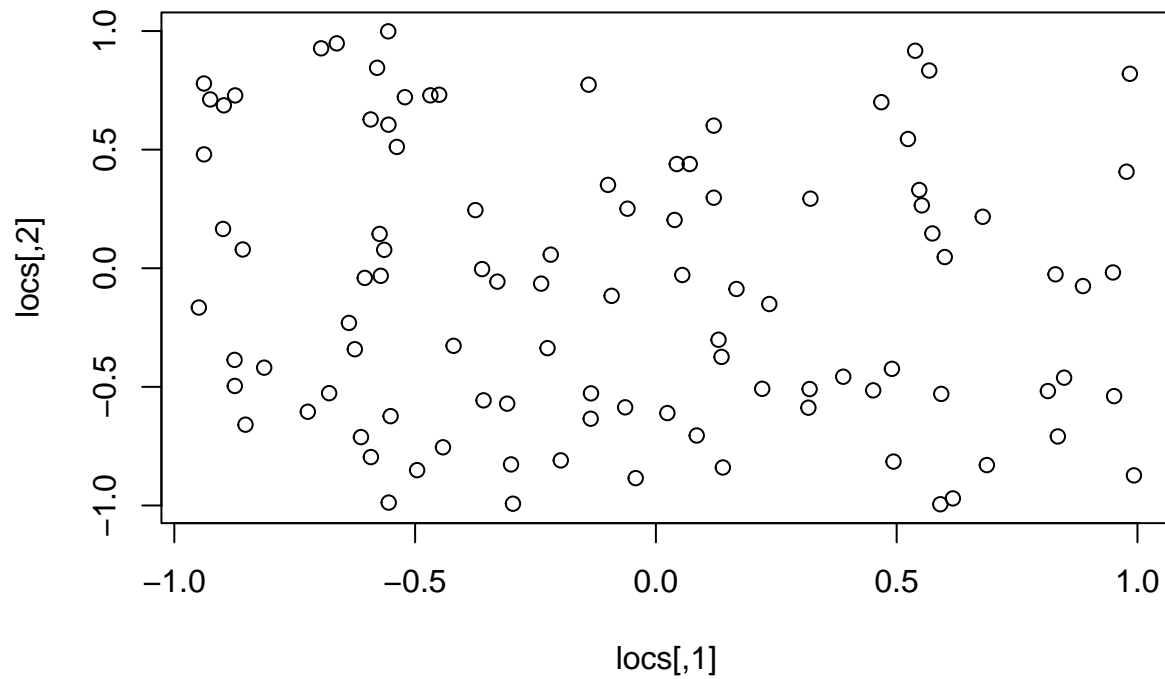
```r
v<-diag(n)

spatial_dists<-as.matrix(dist(locs,
                              diag = TRUE,
                              upper = TRUE))

#x_pair
x_pair<-matrix(1,
               nrow = n_star,
               ncol = 2)
x_pair[,2]<-rnorm(n = n_star)
x_pair[,2]<-scale(x_pair[,2])

#x_ind
x1<-rnorm(n = n)
x2<-rnorm(n = n)
x3<-rnorm(n = n)
x_ind<-matrix(1,
              nrow = n_star,
              ncol = 3)
counter<-1
for(j in 1:(n - 1)){
  for(k in (j + 1):n){

    x_ind[counter, 1]<-abs(x1[j] - x1[k])
    x_ind[counter, 2]<-abs(x2[j] - x2[k])
```

```
      x_ind[counter, 3]<-abs(x3[j] - x3[k])
      counter<-counter +
              1

    }
  }
x_ind<-scale(x_ind)

#z
z<-matrix(0,
          nrow = n_star,
          ncol = n)
counter<-1
for(j in 1:(n - 1)){
   for(k in (j + 1):n){

      z[counter, j]<-1
      z[counter, k]<-1
      counter<-counter +
              1

    }
  }
```

- Setting the values for the statistical model parameters:

```
beta_true<- c(1.00,
              -0.30)
gamma_true<-c(2.10,
              -1.75,
              0.00)

tau2_true<-2.00
sigma2_zeta_true<-0.01

phi_true<-5.00
Sigma_true<-tau2_true*exp(-phi_true*spatial_dists)
eta_true<-rmnorm(n = 1,
                 mean = rep(0.00,
                            times = n),
                 varcov = Sigma_true)

theta_true<-eta_true +
           rnorm(n = n,
                 mean = 0.00,
                 sd = sqrt(sigma2_zeta_true))
theta_true<-theta_true -
           mean(theta_true)

mu_true<-x_pair%*%beta_true +
         x_ind%*%gamma_true +
         z%*%theta_true

sigma2_epsilon_true<-0.10
```
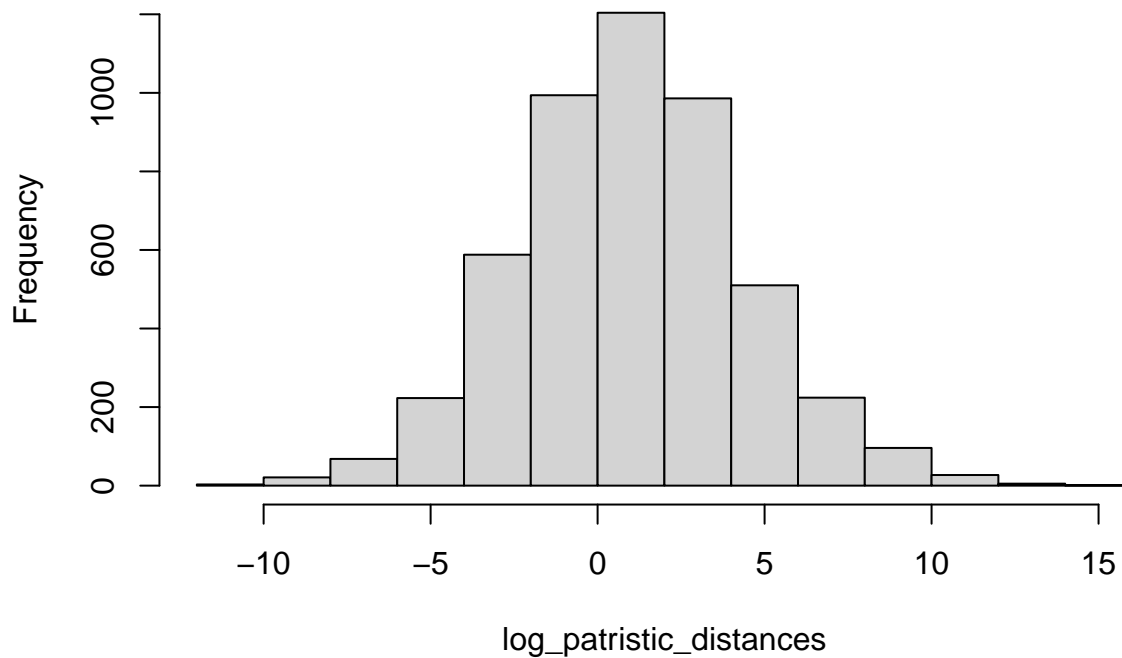
```
log_patristic_distances<-mu_true +
                          rnorm(n = n_star,
                                mean = 0.00,
                                sd = sqrt(sigma2_epsilon_true))

hist(log_patristic_distances)
```

Histogram of log_patristic_distances

log_patristic_distances

[2] Apply the Patristic Function within the GenePair Package:
```
results<-Patristic(mcmc_samples = 20000,
                   log_patristic_distances =  log_patristic_distances,
                   x_pair = x_pair,
                   x_ind = x_ind,
                   z = z,
                   spatial_dists = spatial_dists,
                   v = v,
                   metrop_var_phi_trans = 0.75)
```

```
## Progress: 10%
## phi Acceptance: 25%
## ******************
## Progress: 20%
## phi Acceptance: 24%
## ******************
## Progress: 30%
## phi Acceptance: 24%
## ******************
```

```
## Progress: 40%
## phi Acceptance: 24%
## ******************
## Progress: 50%
## phi Acceptance: 24%
## ******************
## Progress: 60%
## phi Acceptance: 24%
## ******************
## Progress: 70%
## phi Acceptance: 24%
## ******************
## Progress: 80%
## phi Acceptance: 24%
## ******************
## Progress: 90%
## phi Acceptance: 24%
## ******************
## Progress: 100%
## phi Acceptance: 24%
## ******************
```

[3] Analyzing Output:

```
#Compare random effects
plot(rowMeans(results$theta[,10001:20000]),
     theta_true)
abline(0,1)
```
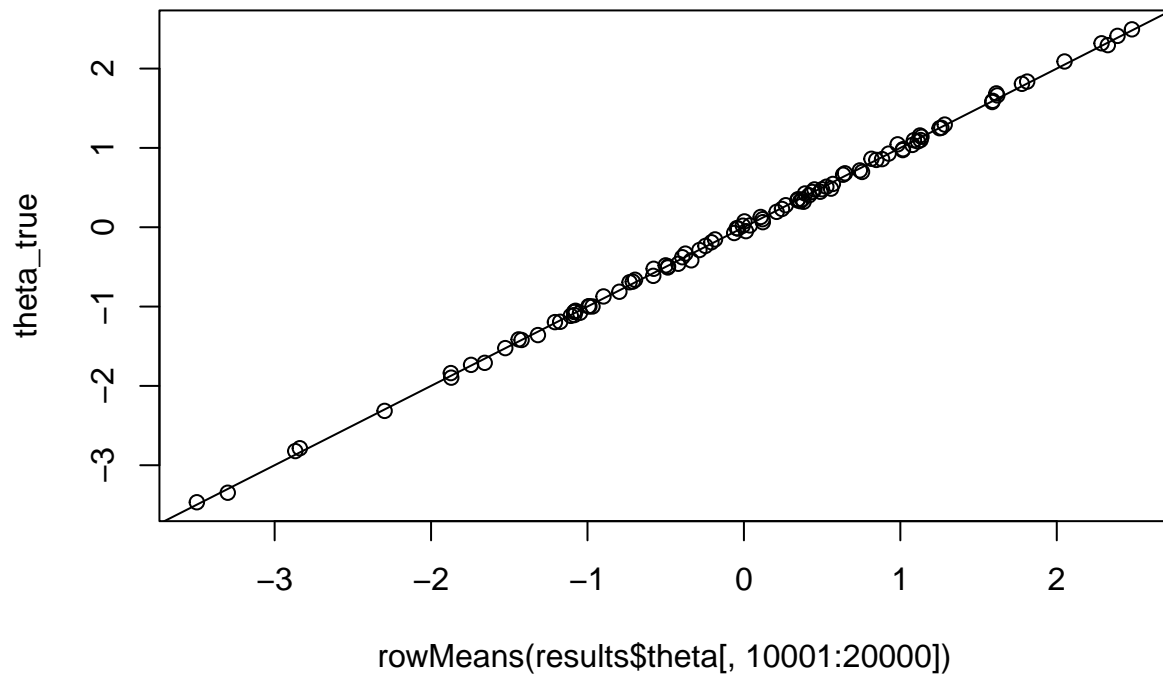
```
#Compare \beta
round(rowMeans(results$beta[,10001:20000]), 2)
```

```
## [1]  0.99 -0.29
```

```
round(beta_true, 2)
```

```
## [1]  1.0 -0.3
```

```
#Compare \gamma
round(rowMeans(results$gamma[,10001:20000]), 2)
```

```
## [1]  2.11 -1.75  0.00
```

```
round(gamma_true, 2)
```

```
## [1]  2.10 -1.75  0.00
```

```
#Compare \sigma2_epsilon
round(mean(results$sigma2_epsilon[10001:20000]), 2)
```

```
## [1] 0.1
```

```
sigma2_epsilon_true
```

```
## [1] 0.1
```