GPCW: Gaussian Process Model for Critical Window Estimation

GPCW_Example

[1] Simulate data from the proposed model:

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

```
set.seed(8453)

library(GPCW)
library(mnormt)  #Multivariate normal distribution
library(boot)  #Inverse logit transformation
```

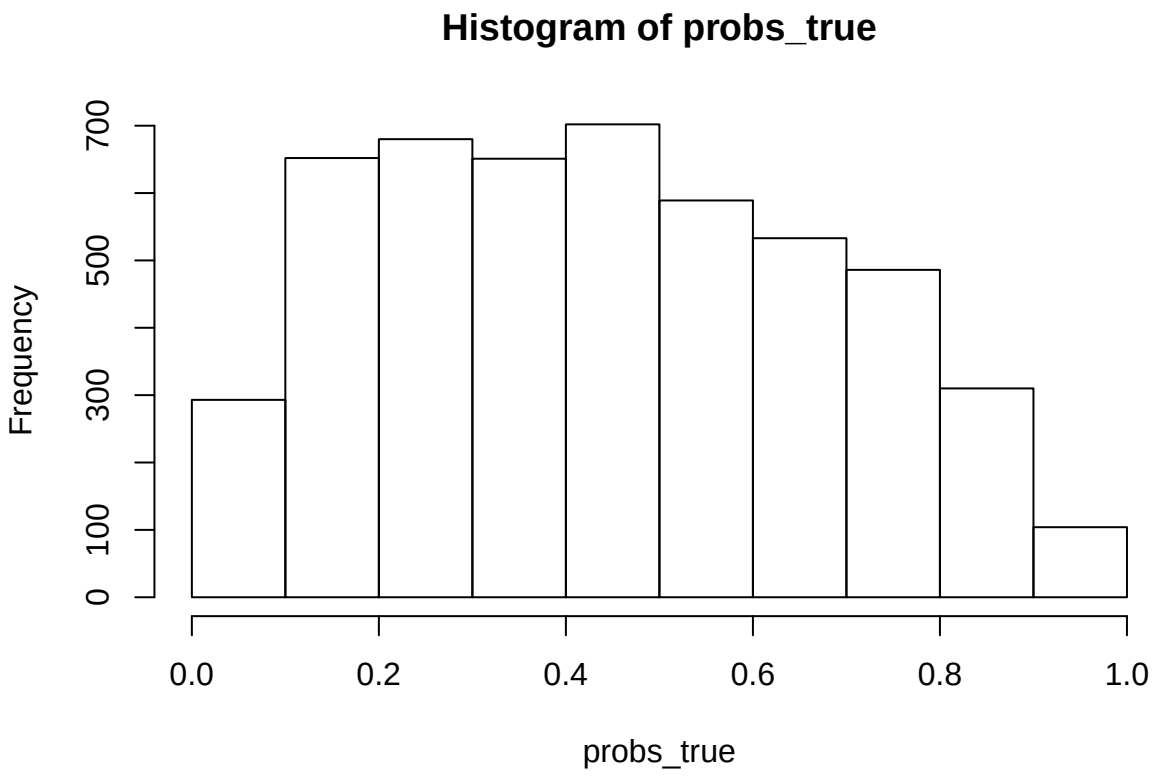
- Setting the global data values:

```
n<-5000  #Sample size
m<-36  #Number of exposure time periods
x<-matrix(1,
          nrow=n,
          ncol=1)  #Covariate design matrix
z<-matrix(rnorm(n=(n*m)),
          nrow=n,
          ncol=m)  #Exposure design matrix

for(j in 1:m){
   z[,j]<-(z[,j] - median(z[,j]))/IQR(z[,j])  #Data standardization (interquartile range)
   }
```

- Setting the values for the statistical model parameters:

```
beta_true<- -0.30
sigma2_theta_true<-0.50
phi_true<-0.01
Sigma_true<-sigma2_theta_true*chol2inv(chol(temporal_corr_fun(m,
                                                              phi_true)[[1]]))
theta_true<-rmnorm(n=1,
                   mean=rep(0, times=m),
                   varcov=Sigma_true)
theta_true<-theta_true - mean(theta_true)
logit_p_true<-x%*%beta_true +
              z%*%theta_true
probs_true<-inv.logit(logit_p_true)
hist(probs_true)
```



- Simulating the analysis dataset:

```
y<-rbinom(n=n,
          size=1,
          prob=probs_true)
```

[2] Fit GPCW to estimate critical windows of susceptibility:

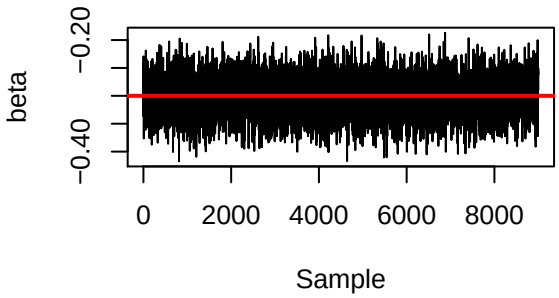
```
results<-GPCW(mcmc_samples = 10000,
              y = y, x = x, z = z,
              metrop_var_phi_trans = 1.15)
```

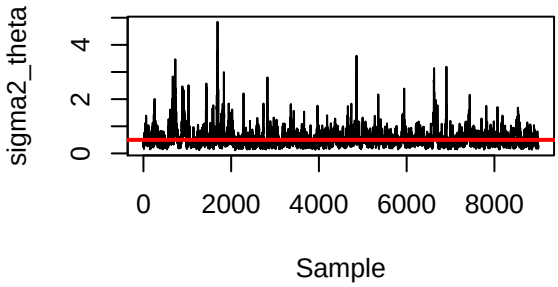
```
## Progress: 10%
## phi Acceptance: 30%
## ********************
## Progress: 20%
## phi Acceptance: 29%
## ********************
## Progress: 30%
## phi Acceptance: 29%
## ********************
## Progress: 40%
## phi Acceptance: 28%
## ********************
## Progress: 50%
## phi Acceptance: 28%
## ********************
## Progress: 60%
```

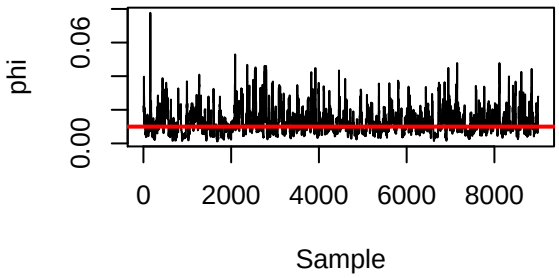
```
## phi Acceptance: 28%
## ********************
## Progress: 70%
## phi Acceptance: 28%
## ********************
## Progress: 80%
## phi Acceptance: 28%
## ********************
## Progress: 90%
## phi Acceptance: 28%
## ********************
## Progress: 100%
## phi Acceptance: 28%
## ********************
```

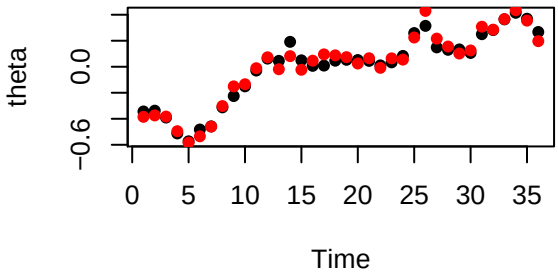
[3] Analyzing Output:

```
par(mfrow=c(2,2))
plot(results$beta[1, 1001:10000],
     type="l",
     ylab="beta",
     xlab="Sample")
abline(h=beta_true,
       col="red",
       lwd=2)  #True value
plot(results$sigma2_theta[1001:10000],
     type="l",
     ylab="sigma2_theta",
     xlab="Sample")
abline(h=sigma2_theta_true,
       col="red",
       lwd=2)  #True value
plot(results$phi[1001:10000],
     type="l",
     ylab="phi",
     xlab="Sample")
abline(h=phi_true,
       col="red",
       lwd=2)  #True value
plot(rowMeans(results$theta[,1001:10000]),
     pch=16,
     ylab="theta",
     xlab="Time")
points(theta_true,
       pch=16,
       col="red")  #True values
```


beta
−0.20
−0.40
0
2000
4000
6000
8000
Sample


sigma2_theta
0
2
4
0
2000
4000
6000
8000
Sample


phi
0.00
0.06
0
2000
4000
6000
8000
Sample


theta
−0.6
0.0
0
5
10
15
20
25
30
35
Time