# 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:

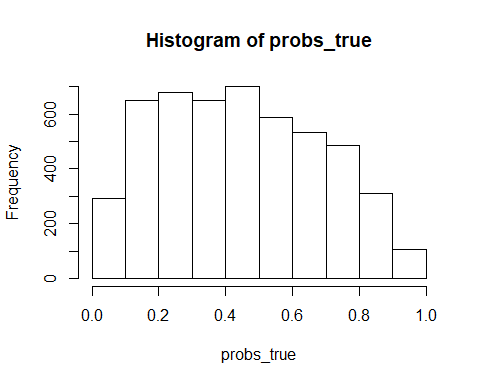
set.seed(8453)  
  
library(GPCW)   
library(mnormt) #Multivariate normal distribution  
library(boot) #Inverse logit transformation

* 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: 27%  
## \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
## Progress: 20%  
## phi Acceptance: 27%  
## \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
## Progress: 30%  
## phi Acceptance: 28%  
## \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
## Progress: 40%  
## phi Acceptance: 28%  
## \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
## Progress: 50%  
## phi Acceptance: 29%  
## \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
## 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

