data simulation1

ss5929

7/30/2020

Exposure

For the exposure, in this scenario, I will only use $x_{t(j)}$ and $x_{t(j-1)}$ to predict $x_{t(j+1)}$ and then the data generation process(DGP) is:

$$x_{t(j+1)} = g^X(x_{t(j)}, x_{t(j-1)})$$

for each time point, $x_{t(j)}$ will follow a Bernoulli distribution and the data generation model is(DGM):

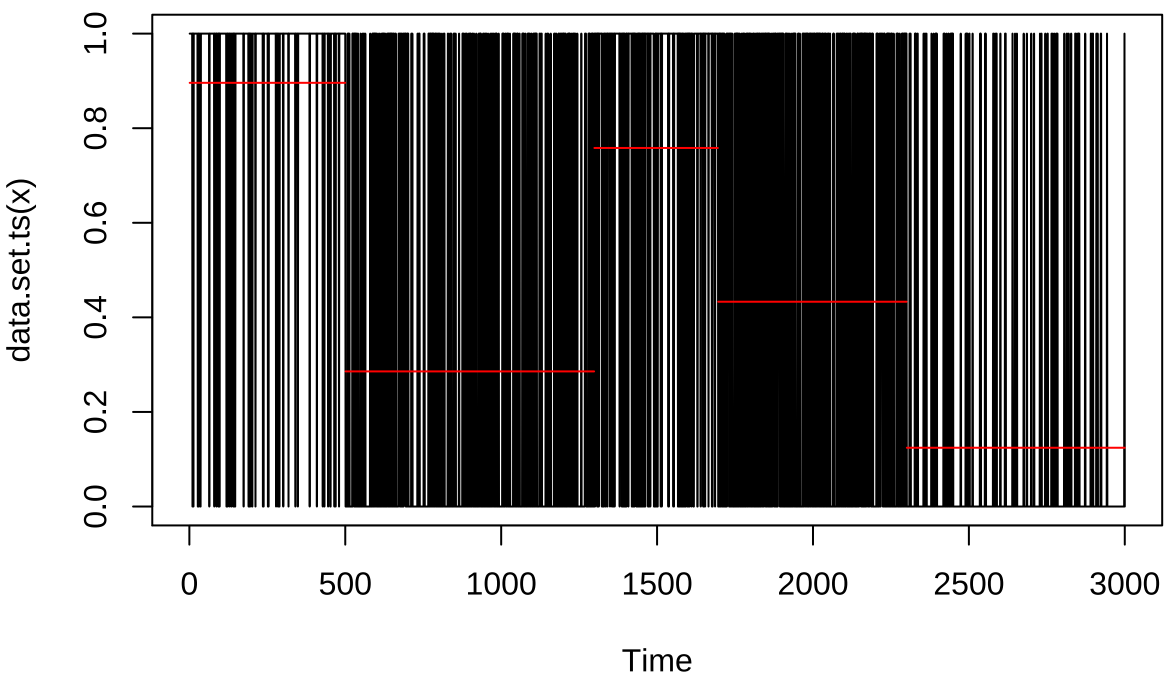
$$X_{t(j+1)} \sim bernoulli(logit(p_{t(j+1)}))$$
$$logit(p_{t(j+1)}) = \nu_t + \phi_{1,t}X_{t(j)} + \phi_{2,t}X_{t(j-1)}$$

Here, ν_t means fixed effect for x at different time periods which is decided by the probability of having high count/texts in each time period. To make it stationary and close to the fact the first lag will have more effect to the current day than the second lag, ϕ_1 and ϕ_2 are restricted as:

$$|\phi_1 + \phi_2| < 1$$
$$|\phi_2 - \phi_1| < 1$$
$$|\phi_1| < 1$$
$$|\phi_2| < 1$$
$$|\phi_2| < |\phi_1|$$

```
#x = c(rbinom(500,1,6/7),rbinom(800,1,2/7),rbinom(400,1,5/7),rbinom(600,1,3/7),rbinom(700,1,1/7))
seed = 1234
# simulate x
x = NULL
phi1 = c(1/7,1/20,1/8,1/10,1/30) # set up phi1
phi2 = c(1/8,1/30,1/9,1/12,1/40) # set up phi2
gama = c(1.8,-0.9,0.9,-0.3,-1.8) # define gama(fixed value)
num = c(500,800,400,600,700)  # define the number of time points in every period
x_test_result = NULL


for(j in 1:5)
{
  set.seed(3399)
  p.treat = NULL
  p = NULL
  a = NULL
  a[1] = ifelse(j==1,1,ifelse(j==2,0,ifelse(j==3,0,ifelse(j==4,1,ifelse(j==5,0,NULL)))))
  a[2] = ifelse(j==1,1,ifelse(j==2,0,ifelse(j==3,1,ifelse(j==4,0,ifelse(j==5,0,NULL))))) # set up the f
  for(i in 3:num[j])
  {
    p[i] = gama[j]+phi1[j]*a[i-1]+phi2[j]*a[i-2]
    p.treat= exp(p[i])/(1+exp(p[i]))
```

```r
    a[i] =sample(rbinom(1000,1,p.treat),size=1)

  }
  adf = adf.test(a)
  x_test_result[j] = adf$p.value
  x=c(x,a)

}

plot(cpt.mean(x,penalty='Manual',pen.value = 2,method='PELT'))
```




```r
mean_x = NULL
variance_x = NULL
idx = 0
for(i in 1:5)
{
  mean_x[i] = mean(x[idx:(idx+num[i])])
  variance_x[i] = var(x[idx:(idx+num[i])])
  idx = idx+num[i]
}
```

The parameters for the exposure are shown as below:

		Parameters of Exposure	
time.period	num	gama	probability
1	500	1.8	0.8571429
2	800	-0.9	0.2857143
3	400	0.9	0.7142857
4	600	-0.3	0.4285714
5	700	-1.8	0.1428571

Outcome without confounder

The DGP is:

$$Y_{t(j)} = g^Y(X_{t(j)}, Y_{t(j-1)}, U_{t(j)})$$

The DGM is:

$$Y_{t(j)} = \zeta_t + \alpha_t X_{t(j)} + \phi_t Y_{t(j-1)} + \eta_{t(j)} U_{t(j)} + \varepsilon_{t(j)}$$

ζ_t: fixed effect for Y at different time period

$U_{t(j)}$: other nonexposure covariates

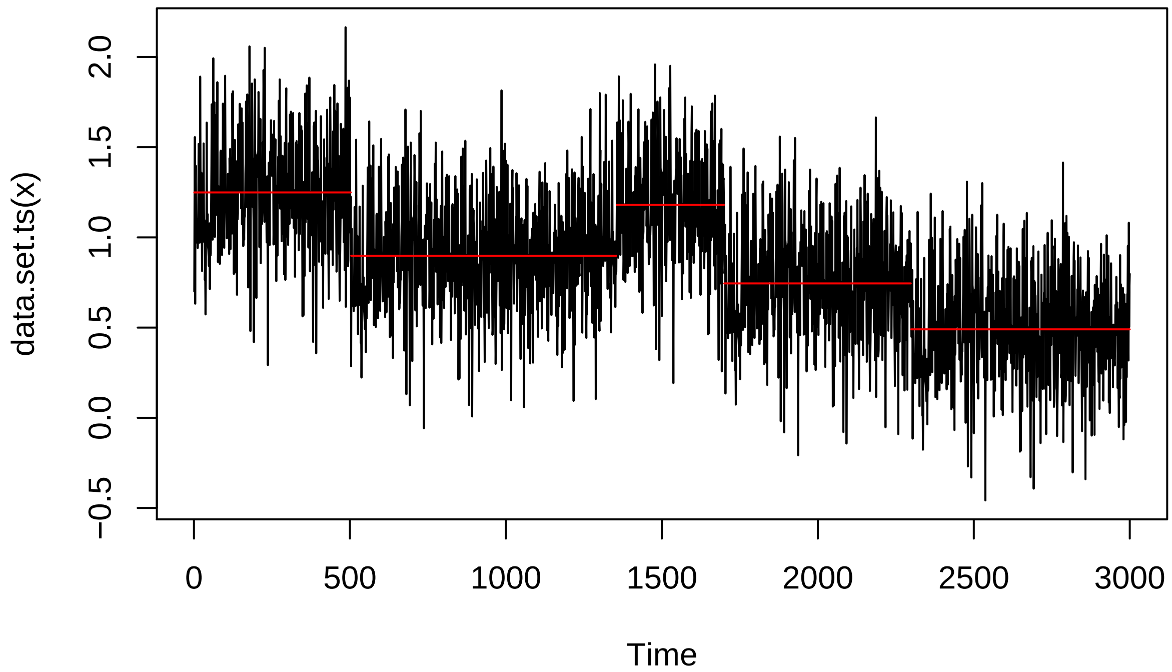
$\varepsilon_{t(j+1)}$ $N(0, \sigma_t^2)$: error term

Other covariates and $\eta_{t(j)}$ will be updated evey time when a new observation is added:

$U_{t(j+1)} = U_t^{\triangle} + \rho_{1,t}(U_{t(j)} - U_t^{\triangle}) + w_{1,t(j)}$ $w_{1,t(j)} \sim N(0, \sigma_{1,w}^2)$

$\eta_{t(j+1)} = \eta_t^{\triangle} + \rho_{2,t}(\eta_{t(j)} - \eta_t^{\triangle}) + w_{2,t(j)}$ $w_{1,t(j)} \sim N(0, \sigma_{2,w}^2)$

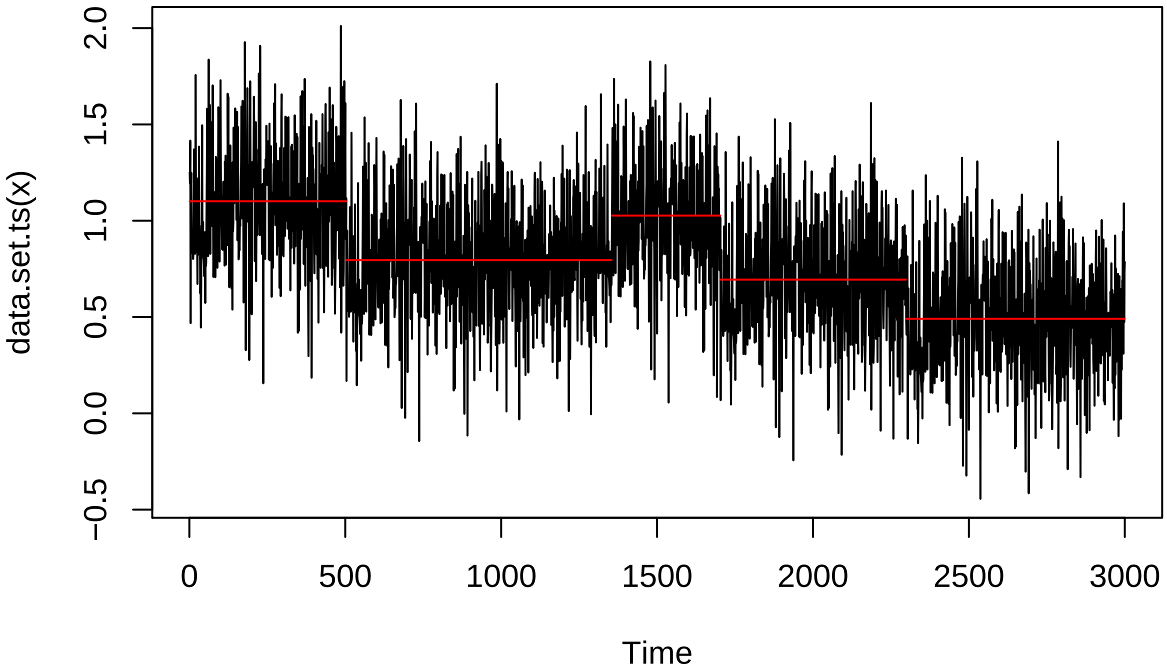
```
# simulate nonexposure covariates U
u = NULL
var_u = 0.08
u0 = c(2.5,1.8,2.3,1.5,1)/2
rho1 = c(0.3,0.2,0.3,0.2,0.1)/2


first_c = c(0.7,1.2,1.8,0.8,0.5)
for(j in 1:5)
{
  set.seed(seed)
  c=NULL
  c[1]=first_c[j]
  error_u = rnorm(num[j],0,sqrt(var_u))
  for(i in 2:num[j])
  {
    c[i] = u0[j] + rho1*(c[i-1]-u0[j])+error_u[i]
  }
  u = c(u,c)
  u_error = NULL
}


plot(cpt.mean(u,penalty='Manual',pen.value = 2,method='PELT'))
```




```
mean_u = NULL
variance_u = NULL
truevar_u = NULL
truemean_u = NULL
idx = 0
for(i in 1:5)
{
  mean_u[i] = mean(u[idx:(idx+num[i])])
  variance_u[i] = var(u[idx:(idx+num[i])])
  idx = idx+num[i]
  truevar_u[i]  = (var_u/(1-rho1[i]^2))
  truemean_u[i] = (1-rho1[i])*u0[i]
}
```

```
# simulate eta
eta = NULL
var_eta = 0.08
eta0 = c(2.2,1.6,2,1.4,1)/2
rho2 = c(0.2,0.1,0.2,0.14,0.08)/2

first_eta = c(1.25,1.05,1.15,0.95,0.55)
for(j in 1:5)
{
  set.seed(seed)
  d=NULL
  d[1]= first_eta[j]
  error_eta = rnorm(num[j],0,sqrt(var_eta))
  for(i in 2:num[j])
  {
    d[i] = eta0[j] + rho2*(d[i-1]-eta0[j])+error_eta[i]
  }
  eta = c(eta,d)
  error_eta = NULL
```

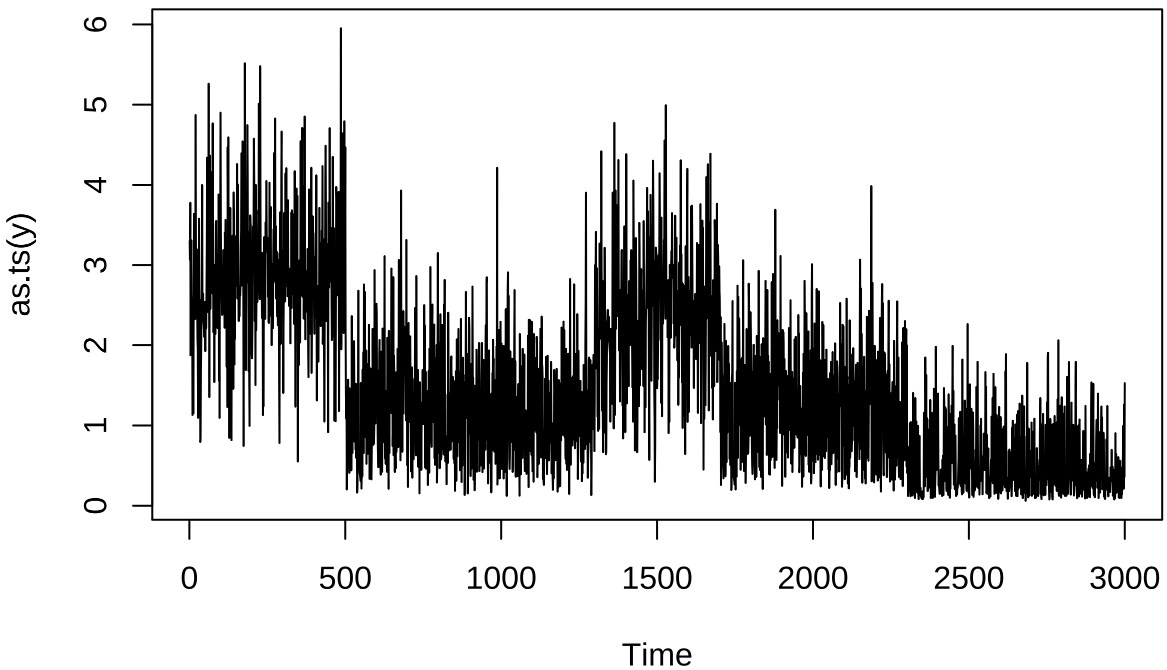
```
}

plot(cpt.mean(eta,penalty='Manual',pen.value = 2,method='PELT'))
```



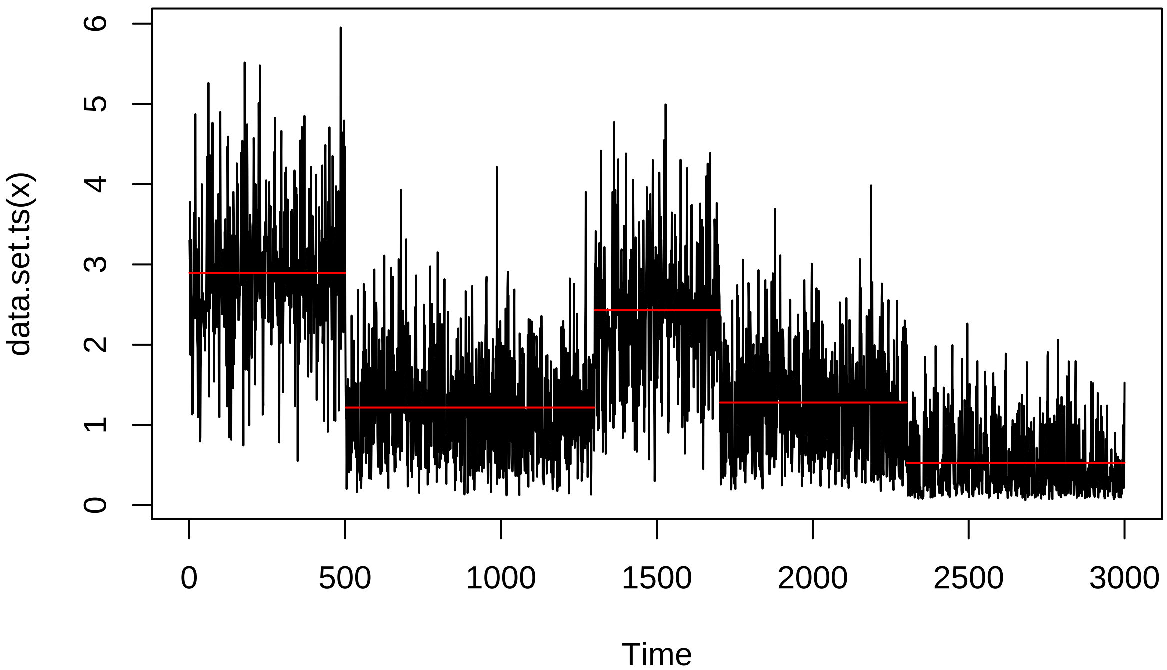

```
mean_eta = NULL
variance_eta = NULL
truemean_eta = NULL
truevar_eta = NULL
idx = 0
for(i in 1:5)
{
  mean_eta[i] = mean(eta[idx:(idx+num[i])])
  variance_eta[i] = var(eta[idx:(idx+num[i])])
  idx = idx+num[i]
  truevar_eta[i]  = (var_eta/(1-rho2[i]^2))
  truemean_eta[i] = (1-rho2[i])*eta0[i]
}
```

```
#simulate Y
zeta = c(2,1,2,1.5,0.8)/10 #fixed effect for y at different time period
phi = c(0.8,0.5,0.7,0.6,0.4)/15
alpha = c(2.4,2,2.3,2.2,1.8)/2
var_y = 0.0001

y_test_result=NULL
y=NULL
id=0
for(j in 1:5)
{
  set.seed(seed)
  e =NULL
  e[1] = ifelse(j==1,3.3,ifelse(j==2,2.3,ifelse(j==3,3,ifelse(j==4,2.6,ifelse(j==5,2,NULL)))))
  error_y = rnorm(num[j],0,sqrt(var_y))
```

```r
  for(i in 2:num[j])
  {
    e[i] = zeta[j]+alpha[j]*x[id:(id+num[j])][i]+phi[j]*e[i-1]+
      eta[id:(id+num[j])][i]*u[id:(id+num[j])][i]+error_y[i]
  }
  adf = adf.test(e)
  y_test_result[j] = adf$p.value
  y = c(y,e)
  error_y =NULL
  id=id+num[j]

}
plot(as.ts(y))
```




```r
plot(cpt.mean(y,penalty='Manual',pen.value = 12,method='PELT'))
```




```
mean_y = NULL
variance_y = NULL
idx = 0
for(i in 1:5)
{
  mean_y[i] = mean(y[idx:(idx+num[i])])
  variance_y[i] = var(y[idx:(idx+num[i])])
  idx = idx+num[i]
}
```

Parameters for the outcome are shown as below:

		other covariates			eta			final outcome		
time.period	num	var_u	u0	rho1	var_eta	eta0	rho2	alpha	zeta	phi
1	500	0.08	1.25	0.15	0.08	1.1	0.10	1.20	0.20	0.05
2	800	0.08	0.90	0.10	0.08	0.8	0.05	1.00	0.10	0.03
3	400	0.08	1.15	0.15	0.08	1.0	0.10	1.15	0.20	0.05
4	600	0.08	0.75	0.10	0.08	0.7	0.07	1.10	0.15	0.04
5	700	0.08	0.50	0.05	0.08	0.5	0.04	0.90	0.08	0.03

```
# generate final data frame
final1 = as.data.frame(cbind(x,y,u))
write.csv(final1,file = "final1.csv")
```

outcome with confounder

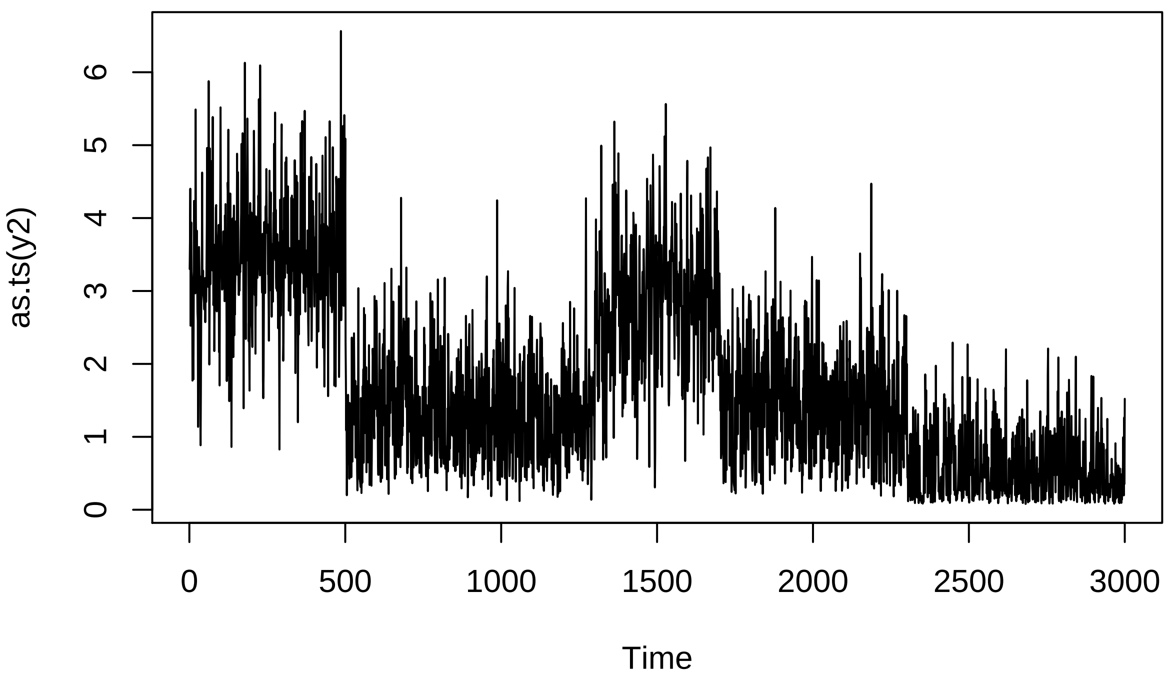
The data generation process(DGP) will be:

$$Y_{t(j)} = g^{Y}(X_{t(j)}, X_{t(j-1)}, Y_{t(j-1)}, U_{t(j)})$$

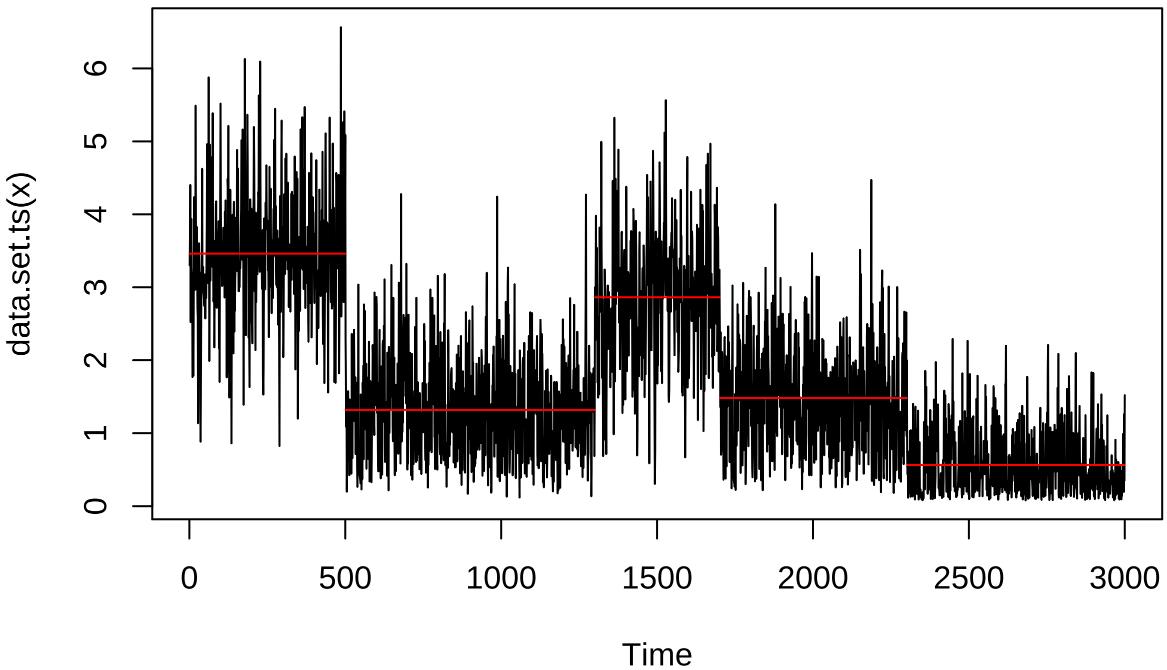
The data generation model(DGM) will be:

$$Y_{t(j)} = \zeta_t + \alpha_t X_{t(j)} + \beta_{1,t} X_{t(j-1)} + \phi_t Y_{t(j-1)} + \eta_{t(j)} U_{t(j)} + \varepsilon_{t(j)}$$

```r
# simulate y with confounder
beta1 = c(1.2,0.7,1.1,0.9,0.6)/2
var_y2 = 0.00001

y2_test_result=NULL
y2=NULL
id=0
for(j in 1:5)
{
  set.seed(seed)
  e =NULL
  e[1] = ifelse(j==1,3.3,ifelse(j==2,2.3,ifelse(j==3,3,ifelse(j==4,2.6,ifelse(j==5,2,NULL)))))
  error_y2 = rnorm(num[j],0,sqrt(var_y2))
  for(i in 2:num[j])
  {
    e[i] = zeta[j]+alpha[j]*x[id:(id+num[j])][i]+phi[j]*e[i-1]+
      eta[id:(id+num[j])][i]*u[id:(id+num[j])][i]+error_y2[i]+beta1[j]*x[id:(id+num[j])][i-1]
  }
  adf = adf.test(e)
  y2_test_result[j] = adf$p.value
  y2 = c(y2,e)
  error_y2 =NULL
  id=id+num[j]

}

plot(as.ts(y2))
```




```r
plot(cpt.mean(y2,penalty='Manual',pen.value = 12,method='PELT'))
```




```
mean_y2 = NULL
variance_y2 = NULL
idx = 0
for(i in 1:5)
{
  mean_y2[i] = mean(y2[idx:(idx+num[i])])
  variance_y2[i] = var(y2[idx:(idx+num[i])])
  idx = idx+num[i]
}
```

distributions

	exposure		outcome without condounder		outcome with confounder	
time_period	mean_x	variance_x	mean_y	variance_y	mean_y2	variance_y2
1	0.90	0.09	2.90	0.71	3.46	0.76
2	0.29	0.21	1.22	0.45	1.33	0.48
3	0.75	0.19	2.43	0.69	2.86	0.78
4	0.43	0.25	1.28	0.49	1.49	0.55
5	0.13	0.11	0.53	0.18	0.57	0.20

	other covariates			
time_period	mean_u	variance_u	truemean_u	truevar_u
1	1.2499358	0.0901623	1.0625	0.0818414
2	0.8931374	0.0842817	0.8100	0.0808081
3	1.1539272	0.0883256	0.9775	0.0818414
4	0.7440270	0.0867324	0.6750	0.0808081
5	0.4901938	0.0838438	0.4750	0.0802005

	parameter of other covariates			
time_period	mean_eta	variance_eta	truemean_eta	truevar_eta
1	1.1015562	0.0877958	0.990	0.0808081
2	0.7933991	0.0826377	0.760	0.0802005
3	1.0023681	0.0851993	0.900	0.0808081
4	0.6945419	0.0850401	0.651	0.0803939
5	0.4907813	0.0823075	0.480	0.0801282