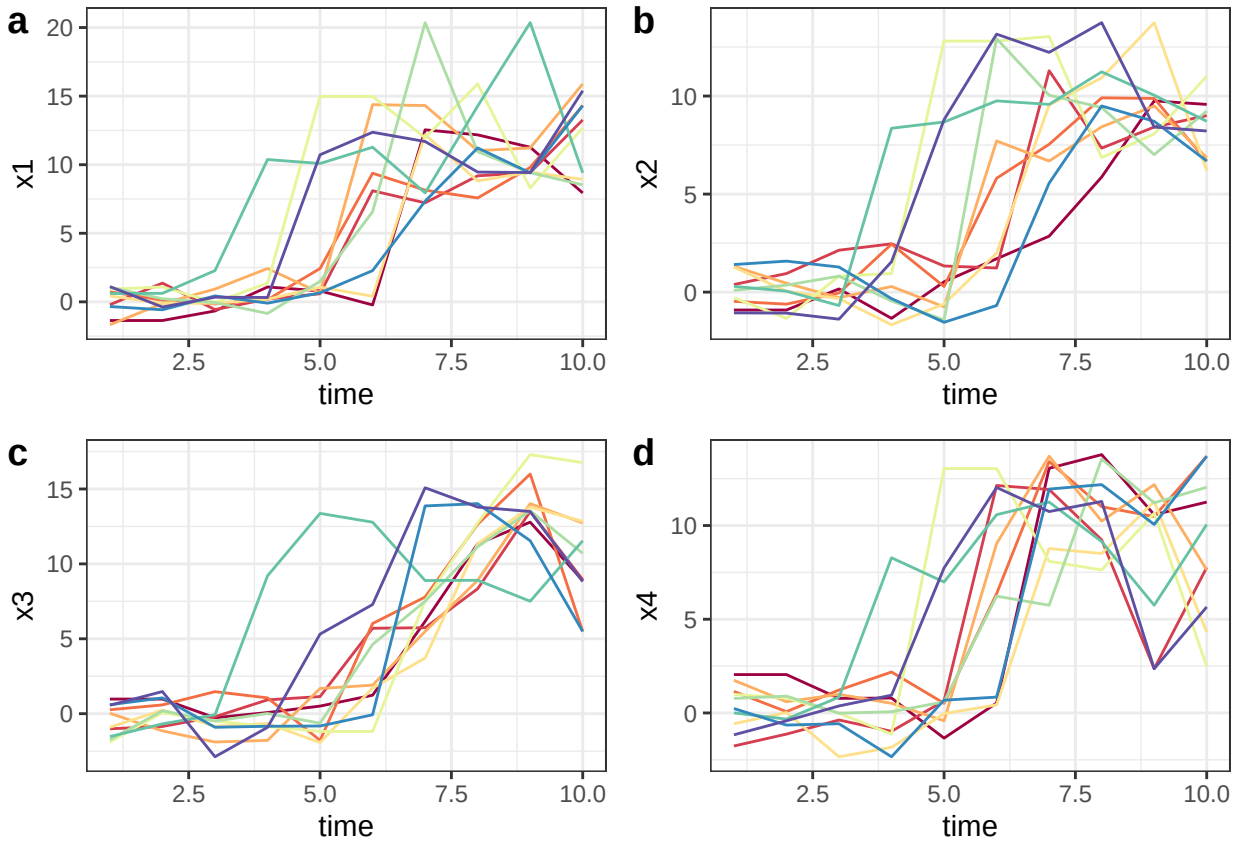
Report - Pseudo Time Series Simulation

Han Chen

2023-04-16

Results

For the simulation, I randomly draw two classes of r.v.'s from multivariate normal distribution, where $X_1 \sim \mathcal{N}_4(0, \mathbf{I}_4)$, and $X_2 \sim \mathcal{N}_4(10, 3\mathbf{I}_4)$, each with 100 samples. The constructed pseudo time series are shown below. Each color represents one time of resampling and sorting, and the x-axis represents pseudo time, the y-axis the value of covariates x_1, x_2, x_3, x_4.



Code

```
## Functions
boot = function(x,y,t){ # bootstrap, choose start and end points
  n = nrow(x)

  repeat{ # sampling
```

```
    id = sample(1:n, t, replace=T)
    s = sum(y[id])
    if(s>0 & s<t) break
  }

  repeat{ # select starting point
    start = sample(id,1)
    if(y[start]==0) break
  }
  #print(start)

  repeat{ # select end point
    end = sample(id,1)
    if(y[end]==1) break
  }
  #print(end)

  res = list(id=id,start=start,end=end)
  return(res)
}

Dist = function(x){ # calculate euclidean distance
  n = nrow(x) # number of observations
  res = matrix(rep(0,n*n),nrow=n) # euclidean distance

  for(i in 1:n){
    for(j in 1:n){
      if(j==i) next
      res[i,j] = norm(t(x[i,]-x[j,]))
    }
  }
  return(res)
}

Path = function(x,start,end){ # forming shortest path
  n = nrow(x)
  id = rep(0,n)

  id[1] = start
  id[n] = end

  m = Dist(x) # the distance matrix
  M = max(m)
  m = m + diag(rep(M,n))
  m[,id[1]] = rep(M,n)
  m[,id[n]] = rep(M,n)

  for(i in 2:(n-1)){
    id[i] = which.min(m[id[i-1],])
    m[,id[i]] = rep(M,n)
  }

  #print(id)
```

```
  return(id)
}

pts =  function(x,y,t){ # pseudo time construction
  sp = boot(x,y,t) # bootstrap
  sp_df = as.data.frame(x[sp$id,])
  sp_start = min(which(sp$id==sp$start)) # starting point
  #print(sp_start)
  #print(sp$id)
  sp_end = max(which(sp$id==sp$end)) # end point
  sp_ordered = Path(sp_df,sp_start,sp_end) # within group order
  sp_df$s = c(1:t)
  df_ord = sp_df[match(sp_ordered,sp_df$s),] # within group ordered data

  return(df_ord)
}

## Simulation
set.seed(20230416)
x1 = matrix(rep(0,4*100),ncol=4)
x2 = x1
y1 = rep(0,100)
y2 = y1 + 1

for(i in 1:4){
  x1[,i] = rnorm(100,0,1) # class 1
  x2[,i] = rnorm(100,10,3) # class 2
}

x = rbind(x1,x2) # covariates
y = c(y1,y2) # labels

t=10 # number of sampling each time

df = NULL # sampling matrix
p = 10 # times of sampling

for(i in 1:p){ # output combination
  sp = pts(x,y,t)
  df = rbind(df,sp)
}
```