

R code used for Section 5 in the paper: ‘Some links between conditional and coregionalized multivariate Gaussian Random fields’

Miguel A. Martinez-Beneito

April, 17th 2018

Load libraries, cartography, data and related functions

```
# Load 'pbugs' Library (the library is installed if required)
if(!require(pbugs)) {
  if(!require(devtools)) {
    install.packages("devtools")
    devtools::install_github("fisabio/pbugs")
  }
  else {
    install_github("fisabio/pbugs")
  }
  library(pbugs)
}

# Load 'LaplacesDemon' Library for calculating Kullback-Leibler divergences (the library is installed i
if(!require(LaplacesDemon)) {
  install.packages("LaplacesDemon")
  library(LaplacesDemon)
}

## Loading required package: LaplacesDemon

# Load Valencian Region cartography data
load("VR.Rdata")

# Load expected cases
load("Exp.Rdata")
colnames(Exp)<-c("Cirrhosis", "Lung", "Oral")

#Load Observed cases
load("Obs.Rdata")
# Change the line above by the following in order to execute the document with the
# provided fake observed deaths
#load("ObsFake.Rdata")

# fake mortality data:
# This fake dataset has been generated to preserve the privacy of the original dataset
# while making possible to execute the whole code.
# The fake dataset has been generated by executing the following code which tries to
# mimic the original data. In the following code, Obs and Exp stand for the 540X3 matrix
# with the original Observed and Expected cases, respectively, per municipality.
# In the accompanying dataset ("ObsFake.Rdata" file), Obs has been substituted by the
```

```

# result of the following code:
# Fake<-matrix(nrow=540,ncol=3)
# for(i in 1:540){
#   dif.min<-10000
#   set.seed(1)
#   for(j in 1:ceiling(sum(Exp[i,]))){
#     generated<-rpois(3,as.vector(Exp[i,]))
#     dif.obs<-sum(abs(Obs[i,]-generated))
#     if(dif.obs<dif.min){
#       This<-generated
#       dif.min<-dif.obs
#     }
#   }
#   Fake[i,]<-This
# }
# Obs<-Fake
# save(Obs,file="ObsFake.Rdata")

colnames(Obs)<-c("Cirrhosis","Lung","Oral")

# Function for calculating DIC
getDIC<-function(Simu,O,E,save=FALSE){
  Mu<-t(t(Simu$sims.matrix[,which(substr(dimnames(Simu$sims.matrix)[[2]],1,2)=="RR")))*
    as.vector(t(E))
  D<-apply(Mu,1,function(x){-2*sum(dpois(as.vector(t(O)),x,log=T))})
  DMean<-mean(D)
  MuMean<-apply(Mu,2,mean)
  DInMean<- -2*sum(dpois(as.vector(t(O)),MuMean,log=T))
  if(save==TRUE){return(c(DMean,DMean-DInMean,2*DMean-DInMean))}
  cat("D=",DMean,"pD=",DMean-DInMean,"DIC=",2*DMean-DInMean,"\n")
}

#function for calculating CPD and PIT
getCPDandPIT<-function(i,j,Simu,O){
  Observed<-O[i,j]
  mu<-Simu$sims.list$mu[,i,j]
  P.it<-dpois(Observed,mu)
  CPD<-1/mean(1/P.it)
  #randomized PIT, as defined in Czado, Gneiting and Held (2009)
  w.it<-1/P.it/sum(1/P.it)
  aux<-vector()
  if(Observed>0){
    for(k in 0:(Observed-1)){
      aux[k+1]<-crossprod(dpois(k,mu),w.it)
    }
    aux[Observed+1]<-runif(1)*crossprod(dpois(Observed,mu),w.it)
  }
  else{
    aux[1]<-runif(1)*crossprod(dpois(Observed,mu),w.it)
  }
  PIT<-sum(aux)
  list(CPD=CPD,PIT=PIT,P.it=P.it)
}

```

```
}
```

M-model

```
M.model<-function(){  
  # Likelihood  
  for (i in 1:Nareas) {  
    for (k in 1:Ndiseases) {  
      Y[i, k] ~ dpois(mu[i, k])  
      log(mu[i, k]) <- log(E[i, k]) + alpha[k] + S[i,k]  
      RR[i,k] <- exp(alpha[k] + S[i,k])  
    }  
  }  
  
  for (i in 1:Nareas){  
    for(k in 1:Ndiseases){  
      S[i,k]<-inprod2(tDelta[,i],SM[,k])  
    }  
  }  
  for(j in 1:Ndiseases){  
    tDelta[j, 1:Nareas]~car.proper(zeros[],C[],adj[],num[],M[],1,gamma[j])  
    gamma[j]~dunif(gamma.inf,gamma.sup)  
  }  
  for(i in 1:Nareas){zeros[i]<-0}  
  gamma.inf<-min.bound(C[],adj[],num[],M[])  
  gamma.sup<-0.99  
  
  # Square matrix of dimension Ndiseases x Ndiseases  
  for (i in 1:Ndiseases){  
    for (j in 1:Ndiseases){  
      SM[i,j] ~ dnorm(0,prec)  
    }  
  }  
  prec<-pow(sdstruct,-2)  
  sdstruct~dunif(0,10)  
  
  # Other priors  
  for (k in 1:Ndiseases){  
    alpha[k] ~ dflat()  
  }  
}  
  
# Run M-model  
parameters<-c("mu","alpha","RR","gamma","SM","sdstruct")  
  
data<-list(Y=Obs,E=Exp, adj=VR.wb$adj, num=VR.wb$num, C=rep(1/VR.wb$num,VR.wb$num),  
           M=1/VR.wb$num, Nareas=540, Ndiseases=3)  
  
inits<-function(){  
  list(SM=matrix(rnorm(3*3,0,1),nrow=3), tDelta=matrix(rnorm(540*3),nrow=3),  
        alpha=runif(3,-0.5,0), sdstruct=runif(1,0,5))  
}
```

```

}

set.seed(1)
res.M<-pbugs(data=data, inits=inits, par=parameters, model=M.model, n.iter=20000,
             n.burnin = 5000, n.sims=3000, n.chains = 3, bugs.seed=1)
res.M$exec_time

```

Time difference of 10.65048 mins

permuted QsR-models

```

pQsR.model<-function(){
  # Likelihood
  for (i in 1:Nareas) {
    for (k in 1:Ndiseases) {
      Y[i, k] ~ dpois(mu[i, k])
      log(mu[i, k]) <- log(E[i, k]) + alpha[k] + S[i,k]
      RR[i,k] <- exp(alpha[k] + S[i,k])
    }
  }

  for (i in 1:Nareas){
    for(k in 1:Ndiseases){
      S[i,k]<-inprod2(tDelta[,i], SM.star[i,,k])
    }
  }
  for(j in 1:Ndiseases){
    tDelta[j, 1:Nareas]~car.proper(zeros[], C[], adj[], num[], M[], 1, gamma[j])
    gamma[j]~dunif(gamma.inf, gamma.sup)
  }
  for(i in 1:Nareas){zeros[i]<-0}
  gamma.inf<-min.bound(C[], adj[], num[], M[])
  gamma.sup<-0.99

  for(i in 1:Nareas){
    for(j in 1:Ndiseases){
      for(k in 1:Ndiseases){
        SM.star[i,j,k]<-SM[perm[i,j],k]
      }
    }
  }

  for(i in 1:Nareas){
    for(j in 1:Ndiseases){
      perm[i,j]<-permutas[cual[i],j]
    }
    cual[i]~dcat(p[1:ncual])
  }
  for(i in 1:ncual){p[i]<-1/ncual}

  # Square matrix Ndiseases x Ndiseases
  for (i in 1:Ndiseases){

```

```

    for (j in 1:Ndiseases){
      SM[i,j] ~ dnorm(0,prec)
    }
  }
prec<-pow(sdstruct,-2)
sdstruct~dunif(0,10)

# Other priors
for (k in 1:Ndiseases){
  alpha[k] ~ dflat()
}
}

# Run permuted QsR-model

# Full-QsR
param<-c("mu","alpha","RR","gamma","SM","sdstruct","perm","cual")

data<-list(Y=Obs,E=Exp,adj=VR.wb$adj,num=VR.wb$num,ncual=6,
          C=rep(1/VR.wb$num,VR.wb$num),M=1/VR.wb$num,Nareas=540,Ndiseases=3,
          permutas=matrix(c(1,2,3,1,3,2,2,1,3,2,3,1,3,1,2,3,2,1),ncol=3,byrow=T))

inits<-function(){
  list(SM=matrix(rnorm(3*3,0,1),nrow=3), tDelta=matrix(rnorm(540*3),nrow=3),
       alpha=runif(3,-0.5,0), sdstruct=runif(1,0,5), cual=sample(1:6,540,replace=T))
}

set.seed(1)
res.QsR.full<-pbugs(data=data,inits=inits,par=param,model=pQsR.model,n.iter=20000,
                   n.burnin = 5000,DIC=F,n.sims=3000,n.chains = 3,bugs.seed=1)
res.QsR.full$exec_time

```

Time difference of 13.98816 mins

```

# (2->3) QsR-model
data<-list(Y=Obs,E=Exp,adj=VR.wb$adj,num=VR.wb$num,ncual=2,
          C=rep(1/VR.wb$num,VR.wb$num),M=1/VR.wb$num,Nareas=540,Ndiseases=3,
          permutas=matrix(c(1,2,3,1,3,2),ncol=3,byrow=T))

inits<-function(){
  list(SM=matrix(rnorm(3*3,0,1),nrow=3),
       tDelta=matrix(rnorm(540*3),nrow=3),
       alpha=runif(3,-0.5,0),
       sdstruct=runif(1,0,5),cual=sample(1:2,540,replace=T))
}

set.seed(1)
res.QsR.23<-pbugs(data=data,inits=inits,par=param,model=pQsR.model,n.iter=20000,
                  n.burnin = 5000,n.sims=3000,DIC=F,n.chains = 3, bugs.seed=1)
res.QsR.23$exec_time

```

Time difference of 12.0703 mins

```

# (3->1->2) QsR-model
data<-list(Y=Obs,E=Exp,adj=VR.wb$adj,num=VR.wb$num,ncual=2,

```

```

C=rep(1/VR.wb$num,VR.wb$num),M=1/VR.wb$num,Nareas=540,Ndiseases=3,
permutas=matrix(c(1,2,3,3,1,2),ncol=3,byrow=T))

set.seed(1)
res.QsR.132<-pbugs(data=data,init=init,par=param,model=pQsR.model,n.iter=20000,
n.burnin = 5000,n.sims=3000,DIC=F,n.chains = 3, bugs.seed=1)
res.QsR.132$exec_time

## Time difference of 12.26749 mins

```

Model selection and validation

```

#DICs comparison
getDIC(O = Obs,E = Exp,Simu = res.M)

## D= 6968.072 pD= 419.8253 DIC= 7387.898

getDIC(O=Obs,E=Exp,Simu=res.QsR.full)

## D= 6962.454 pD= 433.0768 DIC= 7395.531

getDIC(O=Obs,E=Exp,Simu=res.QsR.23)

## D= 6955.077 pD= 419.8564 DIC= 7374.933

getDIC(O=Obs,E=Exp,Simu=res.QsR.132)

## D= 6951.292 pD= 436.6236 DIC= 7387.916

#CPOs and PITs for the M-model
PITS.M<-matrix(nrow=540,ncol=3)
CPO.M<-matrix(nrow=540,ncol=3)
for(i in 1:540){
  for(j in 1:3){
    aux<-getCPOandPIT(i,j,res.M,Obs)
    PITS.M[i,j]<-aux$PIT
    CPO.M[i,j]<-aux$CPO
  }
}

#CPOs and PITs for the full-QsR model
PITS.full<-matrix(nrow=540,ncol=3)
CPO.full<-matrix(nrow=540,ncol=3)
for(i in 1:540){
  for(j in 1:3){
    aux<-getCPOandPIT(i,j,res.QsR.full,Obs)
    PITS.full[i,j]<-aux$PIT
    CPO.full[i,j]<-aux$CPO
  }
}

#CPOs and PITs for the (1->3->2)-QsR model
PITS.132<-matrix(nrow=540,ncol=3)
CPO.132<-matrix(nrow=540,ncol=3)
for(i in 1:540){

```

```

for(j in 1:3){
  aux<-getCPOandPIT(i,j,res.QsR.132,Obs)
  PITS.132[i,j]<-aux$PIT
  CPO.132[i,j]<-aux$CPO
}
}

#CPOs and PITs for the (2->3)-QsR model
PITS.23<-matrix(nrow=540,ncol=3)
CPO.23<-matrix(nrow=540,ncol=3)
for(i in 1:540){
  for(j in 1:3){
    aux<-getCPOandPIT(i,j,res.QsR.23,Obs)
    PITS.23[i,j]<-aux$PIT
    CPO.23[i,j]<-aux$CPO
  }
}

# CPOs comparisons
c(sum(log(CPO.M)),sum(log(CPO.full)),sum(log(CPO.23)),sum(log(CPO.132)))

## [1] -3776.530 -3768.170 -3759.138 -3772.686

# CPOs for cirrhosis
c(sum(log(CPO.M[,1])),sum(log(CPO.full[,1])),sum(log(CPO.23[,1])),sum(log(CPO.132[,1]))))

## [1] -1252.803 -1235.312 -1238.709 -1244.080

# CPOs for lung cancer
c(sum(log(CPO.M[,2])),sum(log(CPO.full[,2])),sum(log(CPO.23[,2])),sum(log(CPO.132[,2]))))

## [1] -1652.751 -1648.987 -1649.240 -1651.308

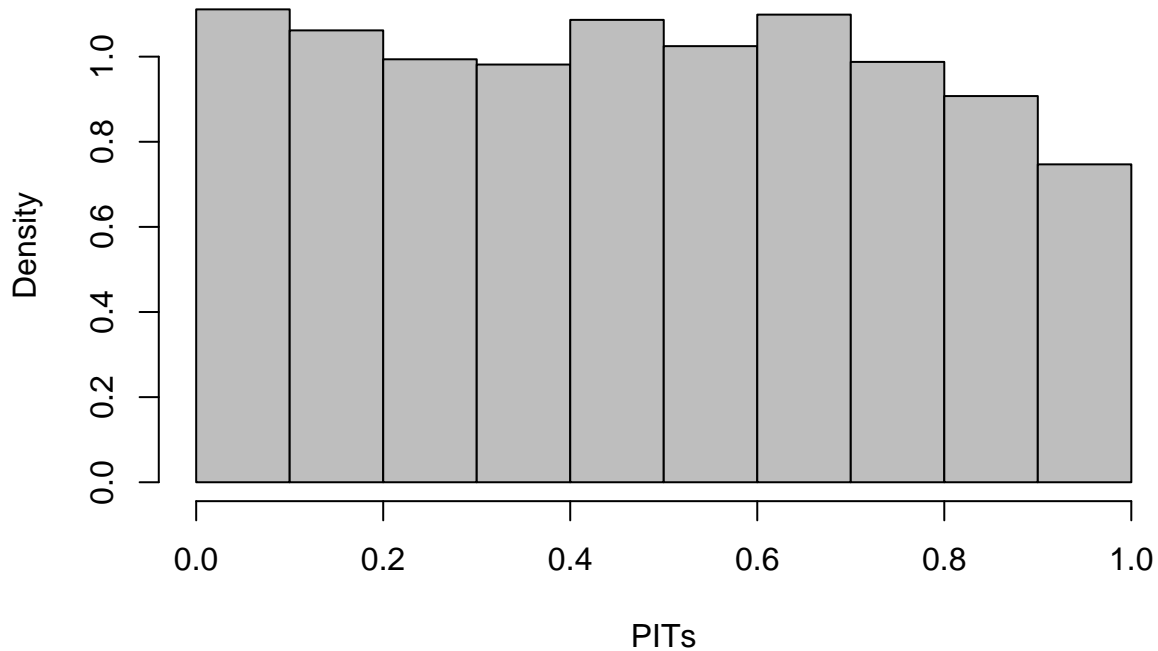
# CPOs for oral cancer
c(sum(log(CPO.M[,3])),sum(log(CPO.full[,3])),sum(log(CPO.23[,3])),sum(log(CPO.132[,3]))))

## [1] -870.9756 -883.8708 -871.1890 -877.2986

#PITs histograms for the 2->3 model
#pdf("PITs.pdf")
hist(PITS.23,main="PITs for all 3 diseases (2->3 model)",xlab="PITs",probability = TRUE, col="gray")

```

PITs for all 3 diseases (2->3 model)



```
#dev.off()

#KL divergence between the PITs of each model and a uniform(0,1) distribution
set.seed(1)
KLD.M<-KLD.full<-KLD.132<-KLD.23<-0
for(i in 1:1000){
  x<-runif(540*3)
  KLD.M<-KLD.M+KLD(PITS.M,x)$mean.sum.KLD
  KLD.full<-KLD.full+KLD(PITS.full,x)$mean.sum.KLD
  KLD.132<-KLD.132+KLD(PITS.132,x)$mean.sum.KLD
  KLD.23<-KLD.23+KLD(PITS.23,x)$mean.sum.KLD
}
c(KLD.M,KLD.full,KLD.132,KLD.23)/1000

## [1] 0.5180522 0.5127370 0.4966165 0.5118516
```

Some results for the (2->3) QsR-model

Proportion of posterior samples that each Valencia neighbour's M matrix is flipped in the same manner as for Valencia (Table 2 in the paper)

```
# Valencia city Neighbours (Valencia is the 526th municipality)
neigh<-VR.nb[[526]]

# Probability of having the same M matrix as Valencia for each of its neighbours.
P.equal.val.neigh<-apply(res.QsR.23$sims.list$cual[,neigh],2,
  function(x){
```



```

        mean(x==res.QsR.23$sims.list$cual[,526])
    })
# Minimum and maximum probability of having the same M matrix as Valencia for each of its neighbours.
c(P.equal.val.neigh[which.min(P.equal.val.neigh)],P.equal.val.neigh[which.max(P.equal.val.neigh)])

## [1] 0.03033333 0.99833333

# Observed cases for each disease in Valencia city
Obs[526,]

## Cirrhosis      Lung      Oral
##      2722      7938      734

# Observed cases for Valencia neighbours with M matrix usually equal to that of Valencia city
apply(Obs[neigh[which(P.equal.val.neigh>0.8)],],2,sum)

## Cirrhosis      Lung      Oral
##      462      1187      106

# Observed cases for Valencia neighbours with M matrix usually different to that of Valencia city
apply(Obs[neigh[which(P.equal.val.neigh<0.2)],],2,sum)

## Cirrhosis      Lung      Oral
##      120      641      65

```

Cross-covariance matrices for Valencia and some of its neighbours (Table 3 in the paper)

```

cual.max<-vector()
neigh<-VR.nb[[526]]
perm23<-matrix(c(1,2,3,1,3,2),ncol=3,byrow=T)
n.it<-dim(res.QsR.23$sims.list$SM)[1]
Cov<-array(dim=c(n.it,3,3))
Covs.mean<-list()
for(j in 1:length(neigh)){
  for(i in 1:n.it){
    Cov[i,]<-t(res.QsR.23$sims.list$SM[i,perm23[res.QsR.23$sims.list$cual[i,526],],])%*%
      diag(res.QsR.23$sims.list$gamma[i,])%*%
      (res.QsR.23$sims.list$SM[i,perm23[res.QsR.23$sims.list$cual[i,neigh[j]],],])
  }
  Covs.mean[[j]]<-apply(Cov,c(2,3),mean)
}

which.max(sapply(Covs.mean,function(x){y<-x-t(x);sum(abs(y))/2}))

## [1] 29

which.min(P.equal.val.neigh)

## [1] 29

# The more asymmetric cross-covariance matrix is that of the neighbouring municipality
# with lowest probability of having the same M matrix as Valencia.

# Cross-covariance for Valencia and its neighbour with lowest probability of having

```

```

# the same M matrix as Valencia
Covs.mean[[which.min(P.equal.val.neigh)]]

##           [,1]      [,2]      [,3]
## [1,] -0.05005174 0.1118775 0.1062521
## [2,]  0.08888649 0.1538258 0.1628434
## [3,]  0.10281181 0.1673209 0.2437983

# Cross-covariance for Valencia and its neighbour with highest probability of having
# the same M matrix as Valencia
Covs.mean[[which.max(P.equal.val.neigh)]]

##           [,1]      [,2]      [,3]
## [1,]  0.2292902 0.1223473 0.1630588
## [2,]  0.1223429 0.1563071 0.1709370
## [3,]  0.1630728 0.1709375 0.2609292

```

5 diseases analysis

```

# data not provided since this section is just for computational goals
load("ObsExp5.Rdata")

```

M-model

```

# M model
data<-list(Y=Obs5,E=Exp5, adj=VR.wb$adj, num=VR.wb$num, C=rep(1/VR.wb$num,VR.wb$num),
           M=1/VR.wb$num, Nareas=540, Ndiseases=5)

inits<-function(){
  list(SM=matrix(rnorm(5*5,0,1),nrow=5), tDelta=matrix(rnorm(540*5),nrow=5),
        alpha=runif(5,-0.5,0), sdstruct=runif(1,0,5))
}

set.seed(1)
res.M.5<-pbugs(data=data,inits=inits,par=parameters,model=M.model, n.iter=20000,
               n.burnin = 5000, n.sims=3000, n.chains = 3,bugs.seed=1)
res.M.5$exec_time

## Time difference of 27.39597 mins

```

full-QsR model

```

param<-c("mu","alpha","RR","gamma","SM","sdstruct","perm","cual")

data<-list(Y=Obs5,E=Exp5,adj=VR.wb$adj,num=VR.wb$num,ncual=120,
           C=rep(1/VR.wb$num,VR.wb$num),M=1/VR.wb$num,Nareas=540,Ndiseases=5,
           permutas=gtools::permutations(5,5))

inits<-function(){
  list(SM=matrix(rnorm(5*5,0,1),nrow=5), tDelta=matrix(rnorm(540*5),nrow=5),

```

```

alpha=runif(5,-0.5,0), sdstruct=runif(1,0,5), cual=sample(1:120,540,replace=T))
}

set.seed(1)
res.QsR.full.5<-pbugs(data=data, inits=inits, par=param, model=pQsR.model, n.iter=20000,
                      n.burnin = 5000, DIC=F, n.sims=3000, n.chains = 3, bugs.seed=1)
res.QsR.full.5$exec_time

## Time difference of 2.017833 hours

```

(2->3) QsR-model

```

data<-list(Y=Obs5, E=Exp5, adj=VR.wb$adj, num=VR.wb$num, ncual=2,
           C=rep(1/VR.wb$num, VR.wb$num), M=1/VR.wb$num, Nareas=540, Ndiseases=5,
           permutas=matrix(c(1,2,3,4,5,1,2,3,5,4), ncol=5, byrow=T))

inits<-function(){
  list(SM=matrix(rnorm(5*5,0,1), nrow=5),
       tDelta=matrix(rnorm(540*5), nrow=5),
       alpha=runif(5,-0.5,0),
       sdstruct=runif(1,0,5), cual=sample(1:2,540,replace=T))
}

set.seed(1)
res.QsR.23.5<-pbugs(data=data, inits=inits, par=param, model=pQsR.model, n.iter=20000,
                    n.burnin = 5000, n.sims=3000, DIC=F, n.chains = 3, bugs.seed=1)
res.QsR.23.5$exec_time

## Time difference of 37.62758 mins

```

(1->3->2) QsR-model

```

data<-list(Y=Obs5, E=Exp5, adj=VR.wb$adj, num=VR.wb$num, ncual=2,
           C=rep(1/VR.wb$num, VR.wb$num), M=1/VR.wb$num, Nareas=540, Ndiseases=5,
           permutas=matrix(c(1,2,3,4,5,3,1,2,4,5), ncol=5, byrow=T))

set.seed(1)
res.QsR.132.5<-pbugs(data=data, inits=inits, par=param, model=pQsR.model, n.iter=20000,
                    n.burnin = 5000, n.sims=3000, DIC=F, n.chains = 3, bugs.seed=1)
res.QsR.132.5$exec_time

## Time difference of 42.41596 mins

```