LGCP with PC priors

Patrick Brown

September 18, 2024

The data

```
library("geostatsp")
data('murder')
data('torontoPop')
murder = unwrap(murder)
torontoBorder = unwrap(torontoBorder)
torontoPdens = unwrap(torontoPdens)
torontoIncome = unwrap(torontoIncome)
```

covariates

```
theCrs = paste0("+proj=omerc +lat_0=43.7117469868935 +lonc=-79.3789787759006",
 " +alpha=-20 +gamma=0 +k=1 +x_0=0 +y_0=0 +datum=WGS84 +units=m +no_defs")
murderT = project(murder, theCrs)
borderT = project(torontoBorder, crs(murderT))
borderC = crop(borderT, ext(-12700, 7000, -7500, 3100))

covList = list(
  pop=torontoPdens,
  inc = log(torontoIncome) )

formulaHere = ~ inc + offset(pop, log=TRUE)
```

LGCP with priors given by quantiles

gamma priors.

```
resG=lgcp(
 formula = formulaHere, data=murderT,
 grid=squareRaster(borderC, 30), covariates=covList,
 border=borderC, buffer=2000,
 prior = list(
```

```
    sd = c(lower = 0.2, upper = 2),
    range = c(lower = 2, upper=20)*1000),
 control.inla=list(strategy='gaussian'))

if(!is.null(resG$parameters)) {
 knitr::kable(resG$parameters$summary, digits=3)
}
```

LGCP with penalised complexity prior

$pr(sd > 1) = 0.05$ and $pr(phi < 0.2) = 0.95$

```
resP=lgcp(formulaHere, data=murderT,
   grid=squareRaster(borderC, 30),
   covariates=covList,
   border=borderC, buffer=2000,
   prior = list(
    sd = c(u=0.5, alpha=0.05),
    range = c(u=10*1000, alpha = 0.4)),
   control.inla = list(strategy='gaussian')
 )

if(!is.null(resP$parameters)) {
 knitr::kable(resP$parameters$summary, digits=3)
}
```

LGCP with table priors

```
sdSeq = seq(0,4,len=501)
rangeSeq = seq(0,15*1000, len=501)
resT=lgcp(formulaHere,
   data=murderT,
   grid=squareRaster(borderC, 30),
   covariates=covList,
   border=borderC, buffer=2000,
   prior = list(
    sd = cbind(sdSeq, dexp(sdSeq, 2)),
    range = cbind(rangeSeq, dexp(rangeSeq, 1/5000))),
   control.inla = list(strategy='gaussian')
)

if(!is.null(resT$parameters)) {
 knitr::kable(resT$parameters$summary, digits=3)
}
```
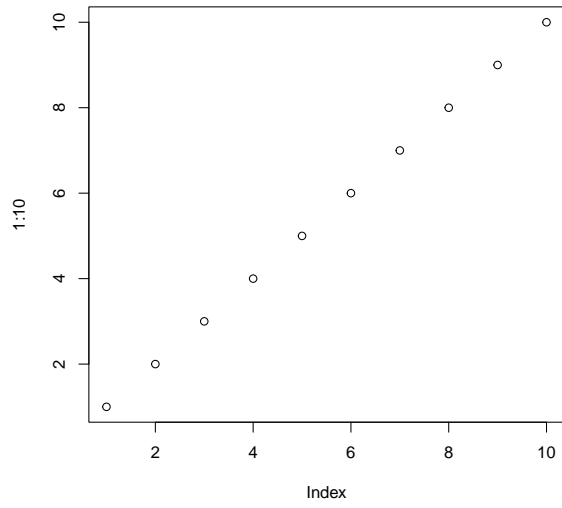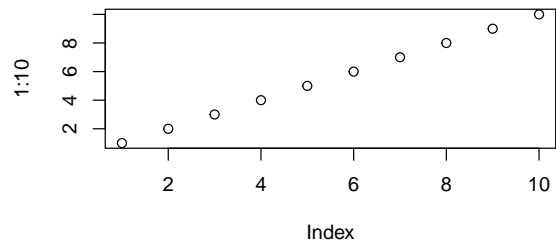
Figure 1: Priors and posteriors

Figure 2: Random effects and fitted values

Maps