LGCP with PC priors

Patrick Brown

September 21, 2023

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

```
if(requireNamespace("INLA", quietly=TRUE) ) {
  INLA::inla.setOption(num.threads=2)
  # not all versions of INLA support blas.num.threads
  try(INLA::inla.setOption(blas.num.threads=2), silent=TRUE)
}
```

```
        theCrs = "+proj=omerc +lat_0=43.7117469868935 +lonc=-79.3789787759006 +alpha=-20
        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

```
if(requireNamespace("INLA", quietly=TRUE)) {
    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')
    )
} else {
    resG = NULL
}

## Warning:  [`[`] using the first column of i to subset x
```

LGCP with penalised complexity prior

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

```
if(requireNamespace("INLA", quietly=TRUE)) {
    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')
    )
} else {
    resP = NULL
}

## Warning:  [`[`] using the first column of i to subset x
```

LGCP with table priors

```
sdSeq = seq(0,4,len=501)
rangeSeq = seq(0,15*1000, len=501)
if( requireNamespace("INLA", quietly=TRUE)) {
        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')
        )
} else {
        resT = NULL
}

## Warning:  [‘[‘] using the first column of i to subset x
```

Parameters

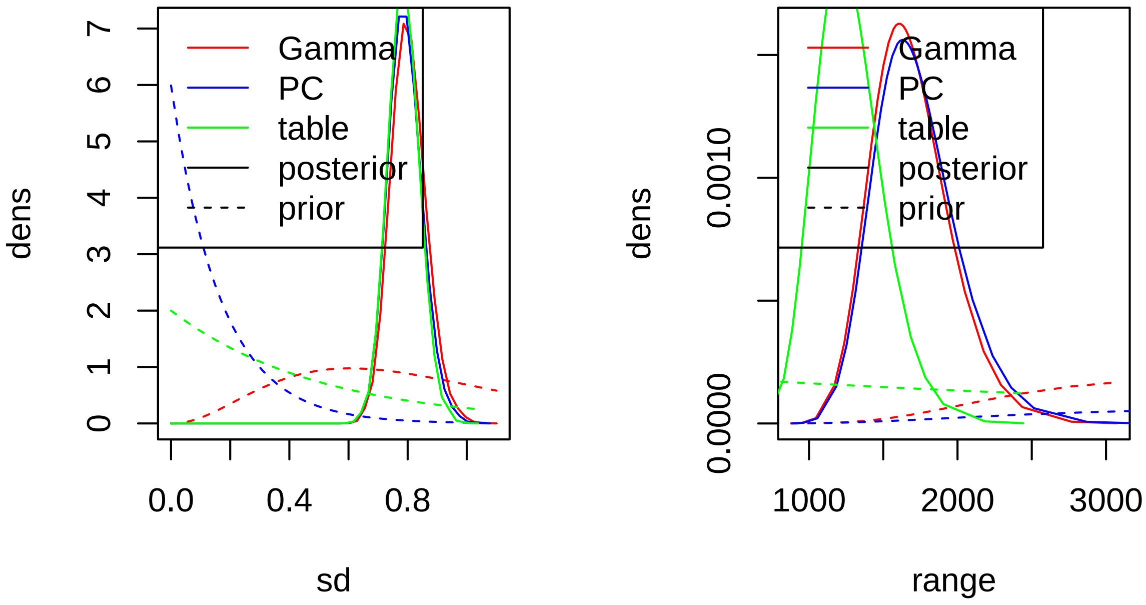
```
if(!is.null(resG$parameters))
        knitr::kable(resG$parameters$summary[,c(1,3,5)], digits=3)
```

	mean	0.025quant	0.975quant
(Intercept)	-3.178	-10.142	3.801
inc	-1.266	-1.911	-0.623
range/1000	1.691	1.237	2.292
sd	0.836	0.695	0.926

```
if(!is.null(resP$parameters))
        knitr::kable(resP$parameters$summary[,c(1,3,5)], digits=3)
```

	mean	0.025quant	0.975quant
(Intercept)	-3.296	-10.220	3.646
inc	-1.255	-1.896	-0.615
range/1000	1.724	1.252	2.361
sd	0.825	0.687	0.910

```
if(!is.null(resT$parameters))
        knitr::kable(resT$parameters$summary[,c(1,3,5)], digits=3)
```



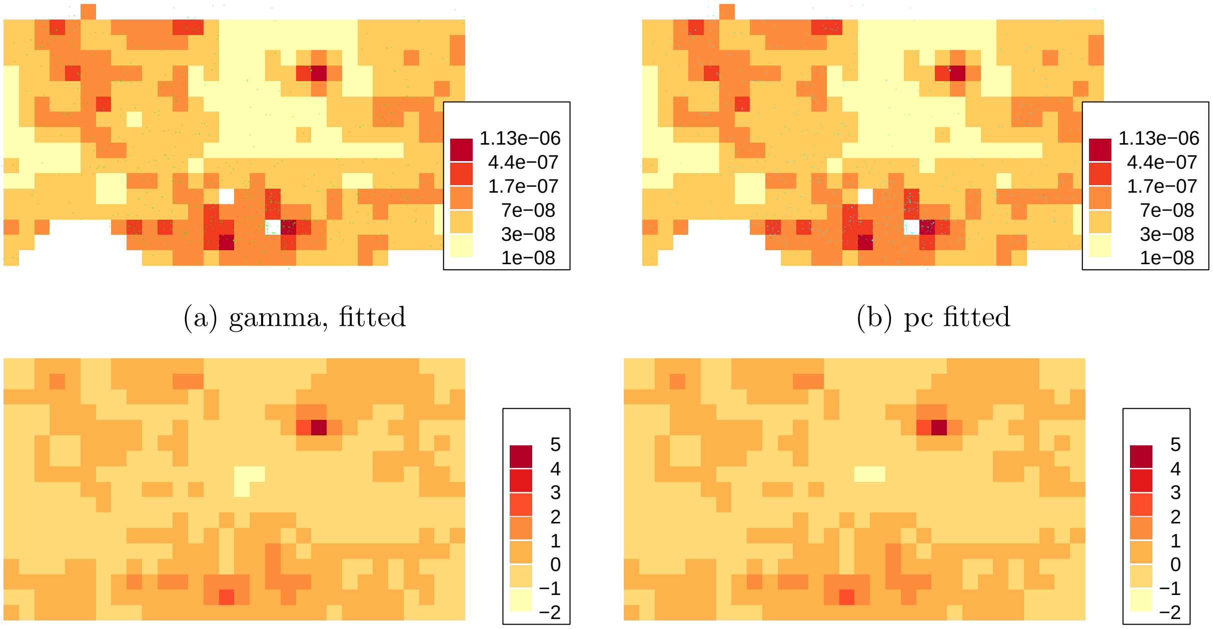
(a) sd

(b) range

Figure 1: Priors and posteriors

	mean	0.025quant	0.975quant
(Intercept)	-2.449	-9.012	4.073
inc	-1.333	-1.936	-0.726
range/1000	1.276	0.886	1.783
sd	0.820	0.686	0.894

Maps



(a) gamma, fitted

(b) pc fitted

(c) gamma random

(d) pc random

Figure 2: Random effects and fitted values