01-SPDEtoy-INLA

Frances Lin

3/28/2022

Explanations are a bit confusing but we will follow the example for the purpose of learning the **R** package `R-INLA`.

Load packages

```
library(INLA)
```

```
## Loading required package: Matrix
```

```
## Loading required package: foreach
```

```
## Loading required package: parallel
```

```
## Loading required package: sp
```

```
## This is INLA_22.03.16 built 2022-03-16 13:24:07 UTC.
##  - See www.r-inla.org/contact-us for how to get help.
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x purrr::accumulate() masks foreach::accumulate()
## x tidyr::expand()     masks Matrix::expand()
## x dplyr::filter()     masks stats::filter()
## x dplyr::lag()        masks stats::lag()
## x tidyr::pack()       masks Matrix::pack()
## x tidyr::unpack()     masks Matrix::unpack()
## x purrr::when()       masks foreach::when()
```

```
library(pander)
library(ggplot2)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

Load data

```
class(SPDEtoy)
```

```
## [1] "data.frame"
```

```
SPDEtoy %>% head %>% pander
```

s1	s2	y
0.08266	0.05641	11.52
0.6123	0.9168	5.278
0.162	0.357	6.903
0.7526	0.2576	13.18
0.851	0.1541	14.6
0.001806	0.7353	9.78

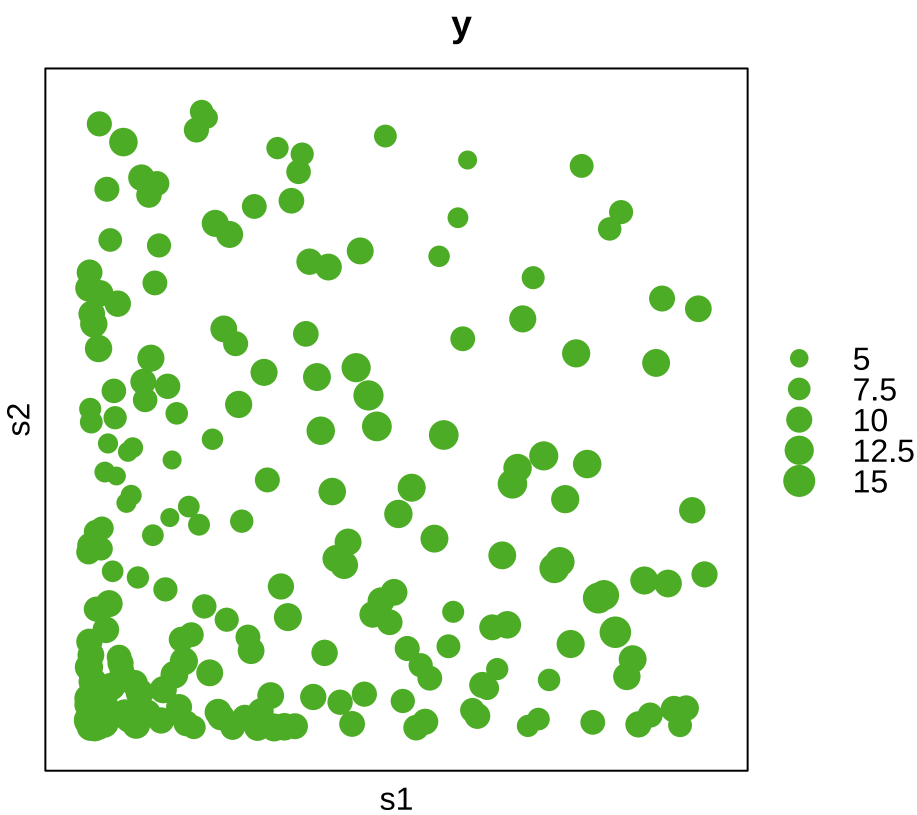
Convert to a SpatialPointsDataFrame

```
SPDEtoy.sp <- SPDEtoy
coordinates(SPDEtoy.sp) <- ~ s1 + s2 # Isn't this from the package sp? Yes.
```

Plot it

`s1` and `s2` are x- and y-coordinate and `y` is the simulated observations at the locations.

```
bubble(SPDEtoy.sp, "y", key.entries = c(5, 7.5, 10, 12.5, 15),
       maxsize = 2, xlab = "s1", ylab = "s2")
```



Fit a MLR model

By default, the prior on the intercept α is a uniform; the prior on the coefficients is a Gaussian with mean 0 and precision ($variance^{-1}$) 0.001, and the prior on the precision τ is a Gamma with parameters 1 and 0.000005. These priors can be adjusted.

The model is as follows:

$$y_i \sim N(\mu_i, \tau^{-1}), i = 1, ..., 200$$
$$\mu_i = \alpha + \beta_1 s_{1i} + \beta_2 s_{2i}$$
$$\alpha \sim Uniform$$
$$\beta_j \sim N(0, 0.001^{-1}), j = 1, 2$$
$$\tau \sim Gamma(1, 0.00005).$$

```
m0 <- inla(y ~ s1 + s2, data = SPDEtoy)
summary(m0)
```

```
##
## Call:
##    c("inla.core(formula = formula, family = family, contrasts = contrasts,
##    ", " data = data, quantiles = quantiles, E = E, offset = offset, ", "
##    scale = scale, weights = weights, Ntrials = Ntrials, strata = strata,
##    ", " lp.scale = lp.scale, link.covariates = link.covariates, verbose =
##    verbose, ", " lincomb = lincomb, selection = selection, control.compute
```

```
##    = control.compute, ", " control.predictor = control.predictor,
##    control.family = control.family, ", " control.inla = control.inla,
##    control.fixed = control.fixed, ", " control.mode = control.mode,
##    control.expert = control.expert, ", " control.hazard = control.hazard,
##    control.lincomb = control.lincomb, ", " control.update =
##    control.update, control.lp.scale = control.lp.scale, ", "
##    control.pardiso = control.pardiso, only.hyperparam = only.hyperparam,
##    ", " inla.call = inla.call, inla.arg = inla.arg, num.threads =
##    num.threads, ", " blas.num.threads = blas.num.threads, keep = keep,
##    working.directory = working.directory, ", " silent = silent, inla.mode
##    = inla.mode, safe = FALSE, debug = debug, ", " .parent.frame =
##    .parent.frame)")
## Time used:
##     Pre = 2.7, Running = 0.263, Post = 0.0265, Total = 2.99
## Fixed effects:
##               mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept) 10.132 0.242      9.656   10.132     10.608 10.132   0
## s1           0.762 0.429     -0.081    0.762      1.605  0.762   0
## s2          -1.584 0.429     -2.427   -1.584     -0.741 -1.584   0
##
## Model hyperparameters:
##                                          mean   sd 0.025quant 0.5quant
## Precision for the Gaussian observations 0.308 0.03      0.252    0.307
##                                         0.975quant  mode
## Precision for the Gaussian observations      0.371 0.305
##
## Marginal log-Likelihood:  -423.18
##  is computed
## Posterior summaries for the linear predictor and the fitted values are computed
## (Posterior marginals needs also 'control.compute=list(return.marginals.predictor=TRUE)')
```

Plot the marginal density for the MLR model

```
marginal_fixed <- m0$marginals.fixed
#m0$marginals.fixed$`(Intercept)`
```

```
# Plot the intercept using ggplot2
# https://www.paulamoraga.com/book-geospatial/sec-inla.html
#m0$marginals.fixed
alpha <- m0$marginals.fixed$`(Intercept)`
ggplot(data.frame(inla.smarginal(alpha)), aes(x, y)) +
  geom_line() +
  labs(x = expression(alpha), y = "density") -> p_alpha
```

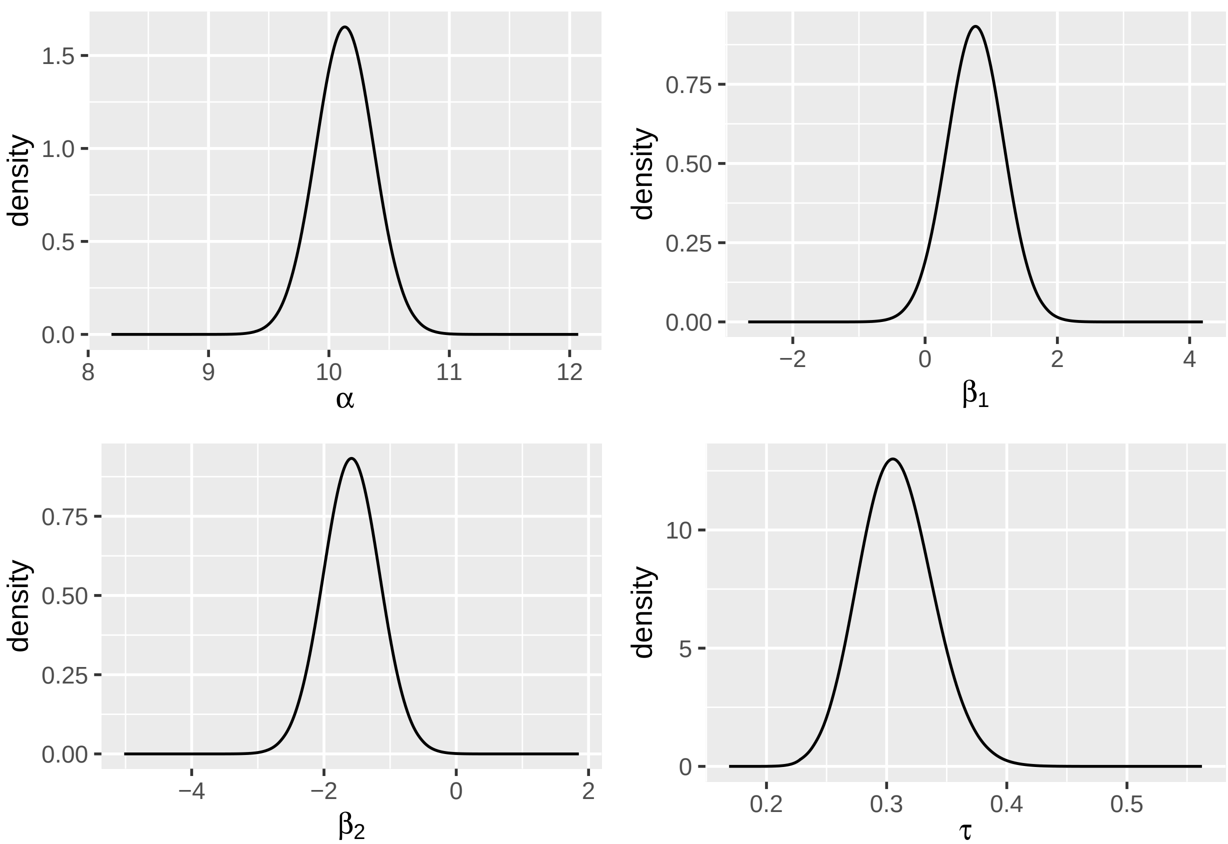
```
# Plot b1
#m0$marginals.fixed
s1 <- m0$marginals.fixed$s1
ggplot(data.frame(inla.smarginal(s1)), aes(x, y)) +
  geom_line() +
  labs(x = expression(beta[1]), y = "density") -> p_beta_1
```

```
# Plot b2
#m0$marginals.fixed
s2 <- m0$marginals.fixed$s2
ggplot(data.frame(inla.smarginal(s2)), aes(x, y)) +
  geom_line() +
  labs(x = expression(beta[2]), y = "density") -> p_beta_2
```

```
#marginal_random <- m0$marginals.random # no random effects here
```

```
# Plot tau
#m0$marginals.hyperpar
tau <- m0$marginals.hyperpar$`Precision for the Gaussian observations`
ggplot(data.frame(inla.smarginal(tau)), aes(x, y)) +
  geom_line() +
  labs(x = expression(tau), y = "density") -> p_tau
```

```
# Combine plots
grid.arrange(p_alpha, p_beta_1, p_beta_2, p_tau, ncol = 2)
```



Fit another model

Reference

1.3 A simple example.