LGCP - Toy Example

Francesco Serafini

19/09/2021

```
knitr::opts_chunk$set(eval = TRUE)
```

Two ways of representing Point Patterns

Given a set of observations $\{\mathbf{y}_i : \mathbf{y}_i \in \mathcal{W}, i = 1, ..., n\}$ of a point process in a region $\mathcal{W}$ we can model this data in two ways: as a point process model or as a poisson counts model. The point process log-likelihood for this data is given by

$$\mathcal{L}_{PP} = -\int_{\mathcal{W}} \lambda(\mathbf{s})d\mathbf{s} + \sum_{i=1}^{n} \log \lambda(\mathbf{y}_i)$$

Where $\lambda(\cdot)$ is the intensity of the point process. The intensity calculated in a point $\mathbf{y}$ is equal to the limit of ratio between the probability of observing at least one event in a ball around $\mathbf{y}$ and the volume of this ball. The limit sends the volume of the ball to zero. Here, the data is composed by the actual observations $\mathbf{y}_1, ..., \mathbf{y}_n$

The second way (Poisson counts) relies on a discretization of the region $\mathcal{W}$. Suppose that the region $\mathcal{W}$ is divided in N bins $b_1, ..., b_K$ such that $\cup b_k = \mathcal{W}$ and $b_k \cap b_j = \emptyset$ for any $k \neq j$. For each bin b_k, N_k represents the number of observed points in b_k. Also, for each bin b_k, λ_k represents the expected number of points by the model in b_k. The log-likelihood for the Poisson counts model is given by

$$\mathcal{L}_{PC} = \sum_{k=1}^{K} -\lambda_k + \log(\lambda_k)N_k$$

$$= -\sum_{k=1}^{K} \lambda_k + \sum_{k=1}^{K} \log(\lambda_k)N_k$$

We have ignored N_k! because it is a known quantity. Here, the data is composed by the Poisson counts per bin $N_1, ..., N_K$.

What INLA sees

INLA is capable of dealing with Poisson models (as the Poisson counts model) but is not capable of dealing with Point process models directly. Thus, we need to approximate the Point Process likelihood with a Poisson counts likelihood. In other words we need $\mathcal{L}_{PP} \approx \mathcal{L}_{PC}$. In order to do that, we need,

$$\int_{\mathcal{W}} \lambda(\mathbf{s})d\mathbf{s} \approx \sum_{k=1}^{K} \lambda_k \qquad \sum_{i=1}^{n} \log \lambda(\mathbf{y}_i) \approx \sum_{k=1}^{K} \log(\lambda_k) N_k$$

The first bit regards the total number of points expected in the region $\mathcal{W}$. In fact, the integral of the intensity represents the expected value of the number of points in $\mathcal{W}$ and the sum of the expected number of points in each bin represents exactly the same thing. The sum of the intensity calculated at the observed points is a measure of "how likely" is to observe the present point patterns while the second summation is a measure of "how likely" is to observe the present counts.

We can reformulate the Poisson counts model in a way that is more convenient to approximate the Point Process model. Suppose that the intensity, in each bin b_k, is contant and equal to $\lambda(\mathbf{p}_k)$, where $\mathbf{p}_k$ is the centroid of the bin b_k. Suppose, also, that the bin b_k has volume (in 3D, area in 2D, length in 1D) E_k. E_k is also know as exposure of the bin b_k. In this case, the expected number of points in the bin b_k is given by the product between $\lambda(\mathbf{p}_k)$ and E_k:

$$\lambda_k = \lambda(\mathbf{p}_k) E_k$$

The likelihood of the Poisson Count model in this case becomes

$$\mathcal{L}_{PC} = -\sum_{k=1}^{K} \lambda(\mathbf{p}_k) E_k + \sum_{k=1}^{K} \log(\lambda(\mathbf{p}_k)) N_k$$

Where we have ignored the term $\log(E_k)N_k$ in the second summation because known quantity. So, we need that

$$\int_{\mathcal{W}} \lambda(\mathbf{s})d\mathbf{s} \approx \sum_{k=1}^{K} \lambda(\mathbf{p}_k) E_k \qquad \sum_{i=1}^{n} \log \lambda(\mathbf{y}_i) \approx \sum_{k=1}^{K} \log(\lambda(\mathbf{p}_k)) N_k$$

Essentially, we are approximating the continuos function $\lambda(\cdot)$ with a piecewise-constant version for which $\forall \mathbf{y} \in b_k$, $\lambda(\mathbf{y}) = \lambda(\mathbf{p}_k)$. The first bit know is just a discrete approximation of the integral. The intensity function $\lambda(\cdot)$ is approximated considering it, in each bin b_k, as contant and equal to the value at $\lambda(\mathbf{p}_k)$. In the second bit, the intensity in a point $\lambda(\mathbf{y}_i)$ is approximated by $\lambda(\mathbf{p}_k)$ where $\mathbf{p}_k$ is the centroid of the bin containing the observation $\mathbf{y}_i$.

Approximating the Integral

In order to provide a better approximation of the integral of the intensity is convenient to base the approximation on a tringulation (or mesh) of the region $\mathcal{W}$. Let's call $\mathbf{s}_1, ..., \mathbf{s}_J$ the mesh points with weights $w_1, ..., w_J$. Then, the integral is approximated by:

$$\int_{\mathcal{W}} \lambda(\mathbf{s})d\mathbf{s} \approx \sum_{j=1}^{J} \lambda(\mathbf{s}_j) w_j$$

Essentially is like considering a Poisson counts model with J bins defined by the triangulation. Here, $\mathbf{s}_j$ is the centroid of the bin b_j and w_j is its exposure. The likelihood of such model would be

$$\mathcal{L}_{int} = -\sum_{j=1}^{J} \lambda(\mathbf{s}_j) w_j + \sum_{j=1}^{J} \log(\lambda(\mathbf{s}_j)) N_j$$

Considering $\forall j,\ N_j = 0$ we have that

$$\mathcal{L}_{int} = -\sum_{j=1}^{J} \lambda(\mathbf{s}_j) w_j$$

Approximating the Summation

To approximate the summation is convenient to consider as centroids the observed points $\mathbf{y}_1, ..., \mathbf{y}_n$. In this way the summation is approximated by:

$$\sum_{i=1}^{n} \log \lambda(\mathbf{y}_i) \approx \sum_{i=1}^{n} \log(\lambda(\mathbf{y}_i)) N_i$$

The associated Poisson counts model log-likelihood is given by:

$$\mathcal{L}_{sum} = -\sum_{i=1}^{n} \lambda(\mathbf{y}_i) w_i + \sum_{i=1}^{n} \log(\lambda(\mathbf{y}_i)) N_i$$

Considering $\forall i,\ w_i = 0, N_i = 1$ we have that

$$\mathcal{L}_{sum} = \sum_{i=1}^{n} \log(\lambda(\mathbf{y}_i))$$

Putting all together

In order to provide a reliable approximation we need to put together the approximation of the integral and the approximation of the summation in a single Poisson counts model. Using the terminology used before, we need to specify a set of centroids $\mathbf{p}_1, ..., \mathbf{p}_P$ representing the bins, a vector of exposures $E_1, ..., E_P$ representing the "size" of the bins and a vector of counts $N_1, ..., N_P$ representing the number of observed events in each bin.

The dimension P is given by the sum of the number of mesh points J and the observations n, so $P = J + n$. We can use the specifications used in the previous two sections, the three vectors are specified as follows:

$$\text{centroids} = \begin{pmatrix} \mathbf{s}_1 \\ \vdots \\ \mathbf{s}_J \\ \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_n \end{pmatrix} \quad \text{exposures} = \begin{pmatrix} w_1 \\ \vdots \\ w_J \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad \text{counts} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$$

The resulting Poisson counts model has log-likelihood given by

$$\mathcal{L}_{PC} = \mathcal{L}_{int} + \mathcal{L}_{sum}$$

$$= -\sum_{j=1}^{J} \lambda(\mathbf{s}_j) w_j + \sum_{i=1}^{n} \log(\lambda(\mathbf{y}_i))$$

Therefore, if $\lambda(\cdot)$ does not require another approximation, the difference between the log-likelihood of the Point Process model and the Poisson counts model depends only on how well we approximate the integral.

Example

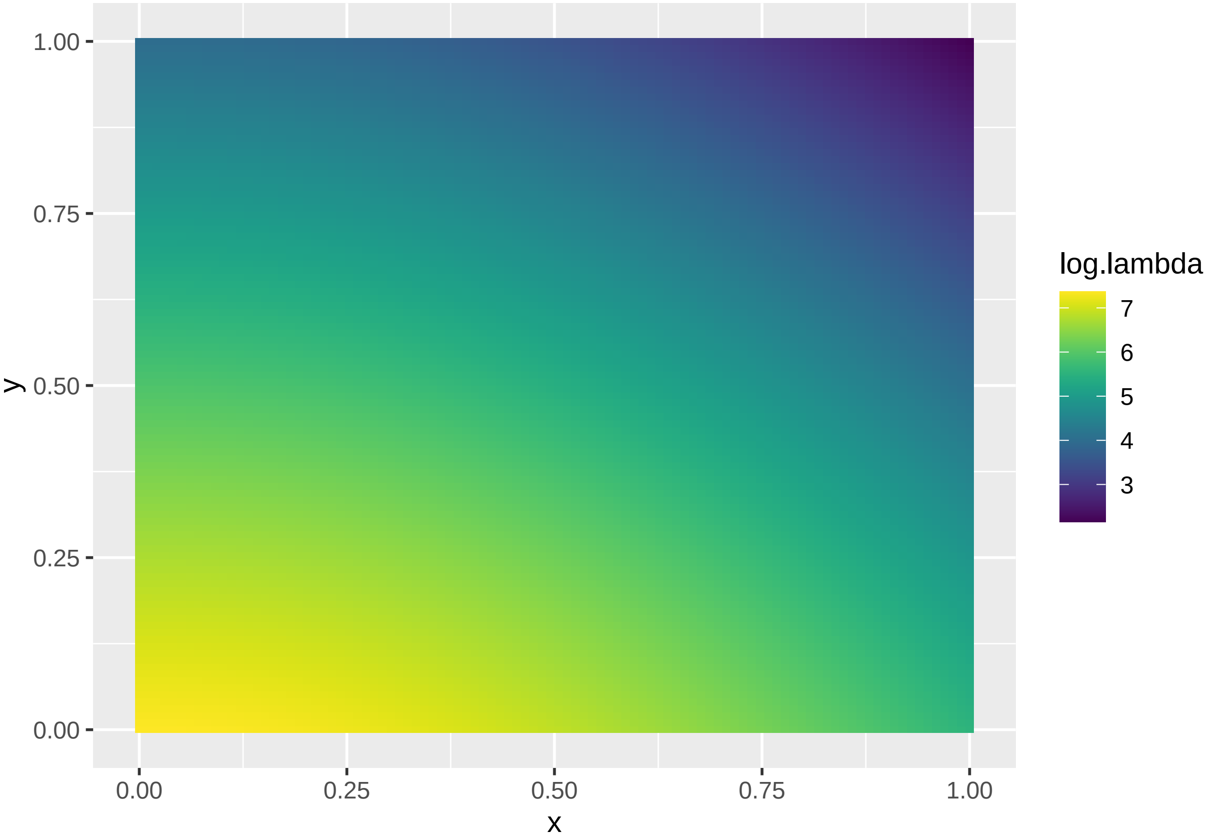
Here, we give a very simple example. We fit a model using the `lgcp` Inlabru function and we fit the corresponding Poisson model using the `bru` function. The results will be exactly the same.

For this example, $\mathbf{y} = (x, y) \in \mathbb{R}^2$ and $\mathcal{W} = (0, 1) \times (0, 1)$. The log-intensity is given by:

$$\log \lambda(x, y) = \theta(\cos(x) - \sin(y - 1))$$

The only parameter of the model is $\theta = 4$ and the log-intensity is a linear function of the spatially varying covariate $\cos(x) - \sin(y - 1)$.

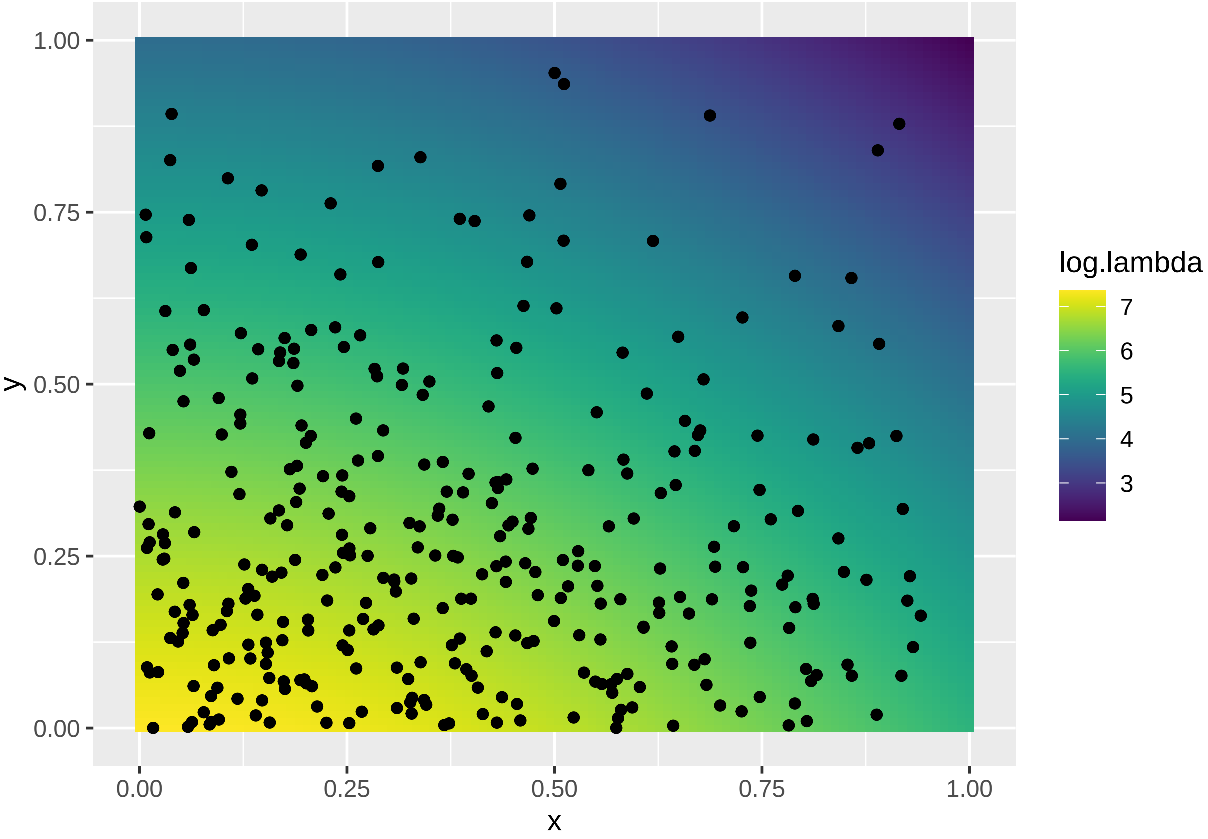
```
# create sequences for plotting
xx <- seq(0,1,length.out = 100)
yy <- seq(0,1,length.out = 100)
pp <- expand.grid(xx, yy)
# fix theta
theta <- 4
# calculate log-intensity
pp$log.lambda <- theta*(cos(pp[,1]) - sin(pp[,2]-1))
colnames(pp) <- c('x', 'y', 'log.lambda')
# plot
ggplot(pp, aes(x = x, y = y, fill = log.lambda)) + geom_tile() + scale_fill_viridis()
```



The first step is to build a mesh and to generate a sample. We have generated a sample considering containing 345 observations.

```
# create a boundary box to build a mesh
b.coords <- cbind(c(0, 0, 1, 1, 0),
                  c(0, 1, 1, 0, 0))
poly1 <- Polygon(b.coords)
poly2 <- Polygons(list(poly1), 'a')
bound <- SpatialPolygons(list(poly2))

# build a mesh
mesh = inla.mesh.2d(boundary = bound, max.edge = c(0.05, 0.2))
# calculate the log-intensity at the mesh location
mesh.logl <- theta*(cos(mesh$loc[,1]) - sin(mesh$loc[,2] - 1))
# sample from the corresponding point process model
set.seed(12)
sample1 <- sample.lgcp(mesh, mesh.logl, samplers = bound)
# plot
ggplot(pp, aes(x = x, y = y, fill = log.lambda)) + geom_tile() +
  geom_point(data = data.frame(sample1@coords),
             mapping = aes(x = x, y = y, fill = NULL)) +
  scale_fill_viridis()
```



Now that we have our observed sample we can fit the models and check that they produce exactly the same results.

```
# LGCP model
# components is the same for both models
cmp <- ~ -1 + thetap(1, model = 'linear')
```

```r
# formula for lgcp
frm.lgcp <- coordinates ~ thetap*(cos(x) - sin(y - 1))

# fit lgcp
fit.lgcp <- lgcp(components = cmp,
                 formula = frm.lgcp,
                 data = sample1,
                 domain = list(coordinates = mesh))

# calculate weights associated with mesh points
ips <- ipoints(domain = mesh)

# build the dataset
data.pois <- data.frame(
  # x,y location (centroids)
  xx = c(mesh$loc[,1], sample1@coords[,1]),
  yy = c(mesh$loc[,2], sample1@coords[,2]),
  # exposures
  exposures = c(ips$weight[,1], rep(0, length(sample1))),
  # observed counts
  obs.c = c(rep(0, mesh$n), rep(1, length(sample1))))

# formula poisson counts model
frm.pois <- obs.c ~ thetap*(cos(xx) - sin(yy - 1))
# poisson fit
fit.pois <- bru(components = cmp,
                formula = frm.pois,
                data = data.pois,
                family = 'poisson',
                options = list(E = data.pois$exposures))
# check the results
rbind(fit.lgcp$summary.fixed, fit.pois$summary.fixed)
```

```
##             mean         sd 0.025quant 0.5quant 0.975quant     mode
## thetap  3.477382 0.03364984   3.410237 3.477754   3.542401 3.478495
## thetap1 3.477382 0.03364984   3.410237 3.477754   3.542401 3.478495
##                  kld
## thetap  3.792323e-08
## thetap1 3.792323e-08
```