

R code - models 1-8

```
1  ## Set up
2  library(sp)
3  library(INLA)
4  library(inlabru)
5  library(raster)
6  library(dplyr)
7
8  ## "pts" is a SpatialPointsDataFrame containing the locations of the
9  ## university halls of residence and additional information about each
10 ## hall.
11
12 ## "boundary" is a SpatialPolygonsDataFrame defining the boundary of the
13 ## area of interest.
14
15 ## "comcov_pxl_spatial" is a SpatialPixelsDataFrame containing
16 ## information on the community level of COVID-19 at each Intermediate
17 ## Zone. "comcov_pxl" is a spatiotemporal version of this
18 ## SpatialPixelsDataFrame, covering 3 months.
19
20 t.proj <- CRS("+proj=tmerc +lat_0=49 +lon_0=-2 +k=0.9996012717
21             +x_0=400000 +y_0=-100000 +ellps=airy +units=km +no_defs")
22 ## names target projection as British National Grid but units=km
23
24 bru_options_set(bru_verbose=3,
25               ## info printed while model is running...
26               control.compute=list(dic=TRUE,waic=TRUE,po=TRUE,cpo=TRUE))
27 ## Compute scores for model assessment...
28
29 ## Mesh
30 mesh <- inla.mesh.2d(boundary=boundary,
31                     max.edge=c(0.1,2),
32                     min.angle=c(30,21),
33                     max.n=c(48000,16000),
34                     max.n.strict=c(128000,128000),
35                     cutoff=0.03,
36                     offset=c(0.1,2),
37                     crs=t.proj)
38
39 ## SPDE priors
40 point_matern <- inla.spde2.pcmatern(mesh,
41                                   prior.range=c(0.2,0.01),
42                                   prior.sigma=c(1,0.01))
43
44 mark_matern <- inla.spde2.pcmatern(mesh,
45                                   prior.range=c(0.2,0.01),
46                                   prior.sigma=c(1,0.01))
47
48 ## Define function for spatiotemporal community covariate
```

```

49 f.comcov <- function(x,y,ti) {
50   spp <- SpatialPoints(data.frame(x=x,y=y),
51                         proj4string=fm_sp_get_crs(comcov_pxl))
52   proj4string(spp) <- fm_sp_get_crs(comcov_pxl)
53   v <- over(spp,comcov_pxl[which(comcov_pxl$ti==ti),])
54   if (any(is.na(v$count_per_pop))) {
55     v$count_per_pop <- bru_fill_missing(count_per_pop,spp,v$count_per_pop)
56   }
57   return(v$count_per_pop)
58 }
59
60 ## Model 1
61 cmp_1 <-
62   ~ -1 +
63   point_field(coordinates,model=point_matern) +
64   mark_field(coordinates,model=mark_matern) +
65   Inter_point(1) + Inter_mark(1) +
66   scaling_latent()
67 point_lik_1 <- like("cp",
68                   formula=coordinates ~ point_field + Inter_point,
69                   include=c("point_field","Inter_point"),
70                   data=pts,
71                   domain=list(coordinates=mesh),
72                   samplers=boundary)
73 mark_lik_1 <- like("poisson",
74                  formula=count ~ Inter_mark +
75                  point_field*scaling_latent + mark_field,
76                  include=c("Inter_mark","point_field","mark_field",
77                            "scaling_latent"),
78                  data=pts)
79 fit_1 <- bru(cmp_1,
80             point_lik_1,
81             mark_lik_1,
82             options=list(verbose=TRUE,E=pts$total_rooms))
83
84 ## Model 2
85 cmp_2 <-
86   ~ -1 +
87   point_field(coordinates,model=point_matern) +
88   mark_field(coordinates,model=mark_matern) +
89   Inter_point(1) + Inter_mark(1) +
90   scaling_latent() +
91   count_per_pop(comcov_pxl_spatial,model="linear")
92 point_lik_2 <- like("cp",
93                   formula=coordinates ~ point_field + Inter_point,
94                   include=c("point_field","Inter_point"),
95                   data=pts,
96                   domain=list(coordinates=mesh),
97                   samplers=boundary)
98 mark_lik_2 <- like("poisson",
99                  formula=count ~ Inter_mark +
100                  point_field*scaling_latent + mark_field +
101                  count_per_pop,
102                  include=c("Inter_mark","point_field","mark_field",
103                            "count_per_pop","scaling_latent"),

```

```

104         data=pts)
105 fit_2 <- bru(cmp_2,
106             point_lik_2,
107             mark_lik_2,
108             options=list(verbose=TRUE,E=pts$total_rooms))
109
110 ## Model 3
111 cmp_3 <-
112   ~ -1 +
113   point_field(coordinates,model=point_matern) +
114   mark_field(coordinates,
115             group=ti,
116             group_mapper=bru_mapper_index(3),
117             model=mark_matern,
118             control.group=list(model="ar1")) +
119   Inter_point(1) + Inter_mark(1) +
120   scaling_latent()
121 point_lik_3 <- like("cp",
122                   formula=coordinates ~ point_field + Inter_point,
123                   include=c("point_field","Inter_point"),
124                   data=pts,
125                   domain=list(coordinates=mesh),
126                   samplers=boundary)
127 mark_lik_3 <- like("poisson",
128                   formula=count ~ Inter_mark +
129                   point_field*scaling_latent + mark_field,
130                   include=c("Inter_mark","point_field","mark_field",
131                             "scaling_latent"),
132                   data=pts)
133 fit_3 <- bru(cmp_3,
134             point_lik_3,
135             mark_lik_3,
136             options=list(verbose=TRUE,E=pts$total_rooms))
137
138 ## Model 4
139 cmp_4 <-
140   ~ -1 +
141   point_field(coordinates,model=point_matern) +
142   mark_field(coordinates,
143             group=ti,
144             group_mapper=bru_mapper_index(3),
145             model=mark_matern_A,
146             control.group=list(model="ar1")) +
147   Inter_point(1) + Inter_mark(1) +
148   scaling_latent() +
149   count_per_pop(f.comcov(x,y,ti),model="linear")
150 point_lik_4 <- like("cp",
151                   formula=coordinates ~ point_field + Inter_point,
152                   include=c("point_field","Inter_point"),
153                   data=pts,
154                   domain=list(coordinates=mesh),
155                   samplers=boundary)
156 mark_lik_4 <- like("poisson",
157                   formula=count ~ Inter_mark +
158                   point_field*scaling_latent + mark_field +

```

```

159         count_per_pop,
160         include=c("Inter_mark", "point_field", "mark_field",
161                 "count_per_pop", "scaling_latent"),
162         data=pts)
163 fit_4 <- bru(cmp_4,
164             point_lik_4,
165             mark_lik_4,
166             options=list(verbose=TRUE, E=pts$total_rooms))
167
168 ## Model 5
169 cmp_5 <-
170   ~ -1 +
171   point_field(coordinates, model=point_matern) +
172   mark_field(coordinates,
173             group=ti,
174             group_mapper=bru_mapper_index(3),
175             model=mark_matern,
176             control.group=list(model="ar1")) +
177   Inter_point(1) + Inter_mark(1) +
178   scaling_latent() +
179   count_per_pop(f.comcov(x,y,ti), model="linear") +
180   prop_ensuite
181 point_lik_5 <- like("cp",
182                   formula=coordinates ~ point_field + Inter_point,
183                   include=c("point_field", "Inter_point"),
184                   data=pts,
185                   domain=list(coordinates=mesh),
186                   samplers=boundary)
187 mark_lik_5 <- like("poisson",
188                  formula=count ~ Inter_mark +
189                  point_field*scaling_latent + mark_field +
190                  count_per_pop + prop_ensuite,
191                  include=c("Inter_mark", "point_field", "mark_field",
192                          "count_per_pop", "prop_ensuite",
193                          "scaling_latent"),
194                  data=pts)
195 fit_5 <- bru(cmp_5,
196             point_lik_5,
197             mark_lik_5,
198             options=list(verbose=TRUE, E=pts$total_rooms))
199
200 ## Model 6
201 cmp_6 <-
202   ~ -1 +
203   point_field(coordinates, model=point_matern) +
204   mark_field(coordinates,
205             group=ti,
206             group_mapper=bru_mapper_index(3),
207             model=mark_matern,
208             control.group=list(model="ar1")) +
209   Inter_point(1) + Inter_mark(1) +
210   scaling_latent() +
211   count_per_pop(f.comcov(x,y,ti), model="linear") +
212   num_fines
213 point_lik_6 <- like("cp",

```

```

214         formula=coordinates ~ point_field + Inter_point,
215         include=c("point_field", "Inter_point"),
216         data=pts,
217         domain=list(coordinates=mesh),
218         samplers=boundary)
219 mark_lik_6 <- like("poisson",
220                 formula=count ~ Inter_mark +
221                 point_field*scaling_latent + mark_field +
222                 count_per_pop + num_fines,
223                 include=c("Inter_mark", "point_field", "mark_field",
224                           "count_per_pop", "num_fines",
225                           "scaling_latent"),
226                 data=pts)
227 fit_6 <- bru(cmp_6,
228             point_lik_6,
229             mark_lik_6,
230             options=list(verbose=TRUE, E=pts$total_rooms))
231
232 ## Model 7
233 cmp_7 <-
234   ~ -1 +
235   point_field(coordinates,
236               model=point_matern) +
237   mark_field(coordinates,
238             group=ti,
239             group_mapper=bru_mapper_index(3),
240             model=mark_matern,
241             control.group=list(model="ar1")) +
242   Inter_point(1) + Inter_mark(1) +
243   scaling_latent() +
244   count_per_pop(f.comcov(x,y,ti), model="linear") +
245   unnamed_hall
246 point_lik_7 <- like("cp",
247                   formula=coordinates ~ point_field + Inter_point,
248                   include=c("point_field", "Inter_point"),
249                   data=pts,
250                   domain=list(coordinates=mesh),
251                   samplers=boundary)
252 mark_lik_7 <- like("poisson",
253                 formula=count ~ Inter_mark +
254                 point_field*scaling_latent + mark_field +
255                 count_per_pop + unnamed_hall,
256                 include=c("Inter_mark", "point_field", "mark_field",
257                           "count_per_pop", "unnamed_hall",
258                           "scaling_latent"),
259                 data=pts)
260 fit_7 <- bru(cmp_7,
261             point_lik_7,
262             mark_lik_7,
263             options=list(verbose=TRUE, E=pts$total_rooms))
264
265 ## Model 8
266 cmp_8 <-
267   ~ -1 +
268   point_field(coordinates, model=point_matern) +

```

```

269   mark_field(coordinates,
270             group=ti,
271             group_mapper=bru_mapper_index(3),
272             model=mark_matern,
273             control.group=list(model="ar1")) +
274   Inter_point(1) + Inter_mark(1) +
275   scaling_latent() +
276   count_per_pop(f.comcov(x,y,ti),model="linear") +
277   unnamed_hall +
278   prop_ensuite
279 point_lik_8 <- like("cp",
280                   formula=coordinates ~ point_field + Inter_point,
281                   include=c("point_field","Inter_point"),
282                   data=pts,
283                   domain=list(coordinates=mesh),
284                   samplers=boundary)
285 mark_lik_8 <- like("poisson",
286                  formula=count ~ Inter_mark +
287                  point_field*scaling_latent + mark_field +
288                  count_per_pop + unnamed_hall + prop_ensuite,
289                  include=c("Inter_mark","point_field","mark_field",
290                            "count_per_pop","unnamed_hall",
291                            "prop_ensuite","scaling_latent"),
292                  data=pts)
293 fit_8 <- bru(cmp_8,
294            point_lik_8,
295            mark_lik_8,
296            options=list(verbose=TRUE,E=pts$total_rooms))
297
298 ## Make Predictions
299 mark_field_1 <- predict(fit_1,pxl,~ mark_field,n.samples=1000)
300 mark_field_2 <- predict(fit_2,pxl,~ mark_field,n.samples=1000)
301 mark_field_3 <- predict(fit_3,pxl,~ mark_field,n.samples=1000)
302 mark_field_4 <- predict(fit_4,pxl,~ mark_field,n.samples=1000)
303 mark_field_5 <- predict(fit_5,pxl,~ mark_field,n.samples=1000)
304 mark_field_6 <- predict(fit_6,pxl,~ mark_field,n.samples=1000)
305 mark_field_7 <- predict(fit_7,pxl,~ mark_field,n.samples=1000)
306 mark_field_8 <- predict(fit_8,pxl,~ mark_field,n.samples=1000)
307
308 ## pxl is an empty SpatialPixelsDataFrame with a column for each month

```