Supporting information for ‘The Point Process Framework for Integrated Modelling of Biodiversity Data’

Kwaku Peprah Adjei, Philip Mostert, Jorge Sicacha Parada, Emma Skarstein, Robert B. O’Hara

Last updated: June 20 2024 at 07:29:28.

Setup

These are the packages we will be using:

```
#devtools::install_github("PhilipMostert/PointedSDMs")
library(PointedSDMs) # model fitting
library(ggplot2)     # plotting
library(raster)      # plotting
library(mapproj)     # map options for plotting
library(INLA)        # functions for specifying mesh
library(dplyr)       # data handling
library(sf)          # spatial stuff
library(showtext)    # font for plot
library(giscoR)      # Polygon object
library(patchwork)   # combining figures
```

Downloading data

Before you run this file, make sure you have the following files in the given locations:

- `data/environmental_covariates.rds`
  - should be in the data folder already.
- `data/Norwegian_lakes.rds`
  - can be downloaded from https://bird.unit.no/resources/9b27e8f0-55dd-442c-be73-26781dad94c8/content, click on “Innhold”-tab at the bottom of the page and select “Norwegian_lakes.rds”.
- `data/artsobs_clean.rds`
  - created from running the script `R/data_preparation.R`, see further instructions there.
- `data/survey_clean.rds`
  - also created from running the script `R/data_preparation.R`, see further instructions there.

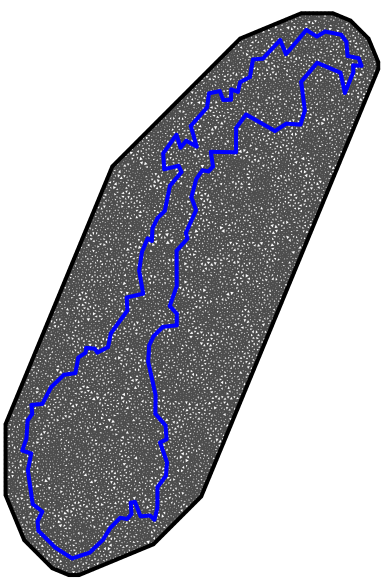
Loading Norway map and making mesh

We begin by making a spatial mesh out of a map of Norway.

```
#proj <- '+proj=utm +zone=32 +datum=WGS84 +units=km +no_defs'#m
proj <- '+proj=tmerc +lat_0=58 +lon_0=6.05625 +k=1 +x_0=0 +y_0=0 +a=6377492.018 +units=km +no_defs +typ
norway.poly <- giscoR::gisco_get_countries(year = 2020, country = 'Norway', resolution = 60)
norway.poly <- st_transform(norway.poly, proj)
norway.poly <- st_cast(st_as_sf(norway.poly), 'POLYGON')
norway.poly <- norway.poly[which.max(st_area(norway.poly)),]

norway.poly.simp <- rmapshaper::ms_simplify(norway.poly, keep = 0.8)
```

Adjusting the mesh to be coarser is the easiest way to decrease the run-time for the model. With the following mesh, the model estimation takes a long time to complete, but feel free to change the `max.edge` or `cutoff` to get a coarser mesh.

```
mesh <- inla.mesh.2d(boundary = inla.sp2segment(norway.poly.simp),
                     cutoff = 0.3 * 30, # 30smallest allowed distance between points
                     max.edge = c(6, 3) * 7.5, # 10decrease this for more int. points
                     offset = c(1, 1) * 50, #50
                     crs = st_crs(proj))

ipoints = fm_int(mesh, norway.poly.simp)

plot(mesh)
```



Setting up covariate data

Next we load the environmental data, which will be used as covariates in our model.

```
covariates_raw <- readRDS("data/environmental_covariates.RDS")

covariates <- covariates_raw %>%
  # Log-transform area of lake
  dplyr::mutate(log_area = log(area_km2)) %>%
  # Log-transform catchment area of lake
  dplyr::mutate(log_catchment = log(catchment_area_km2)) %>%
  # Remove some uninformative variables
```

```r
  dplyr::select(-c(ebint, no_vatn_lnr, eb_waterregionID))

# Choose from
# "decimalLatitude", "decimalLongitude",
# "log_area", "perimeter_m", "distance_to_road",
# "eurolst_bio10", "catchment_area_km2", "SCI", "HFP"

Use <- c("SCI", 'HFP')

cov_pixel <- SpatialPixelsDataFrame(
  points = covariates[,c("decimalLongitude","decimalLatitude")],
  data = data.frame(covariates[,Use]),
  proj4string = CRS('+proj=lonlat +zone=32 +datum=WGS84 +units=m +no_defs'),
  tol = 0.99)
if (length(Use) == 1) names(cov_pixel@data) <- Use
# Scale covariates and convert to terra::rast
cov_raster <- scale(terra::project(terra::rast(cov_pixel), proj))
```

Observation data

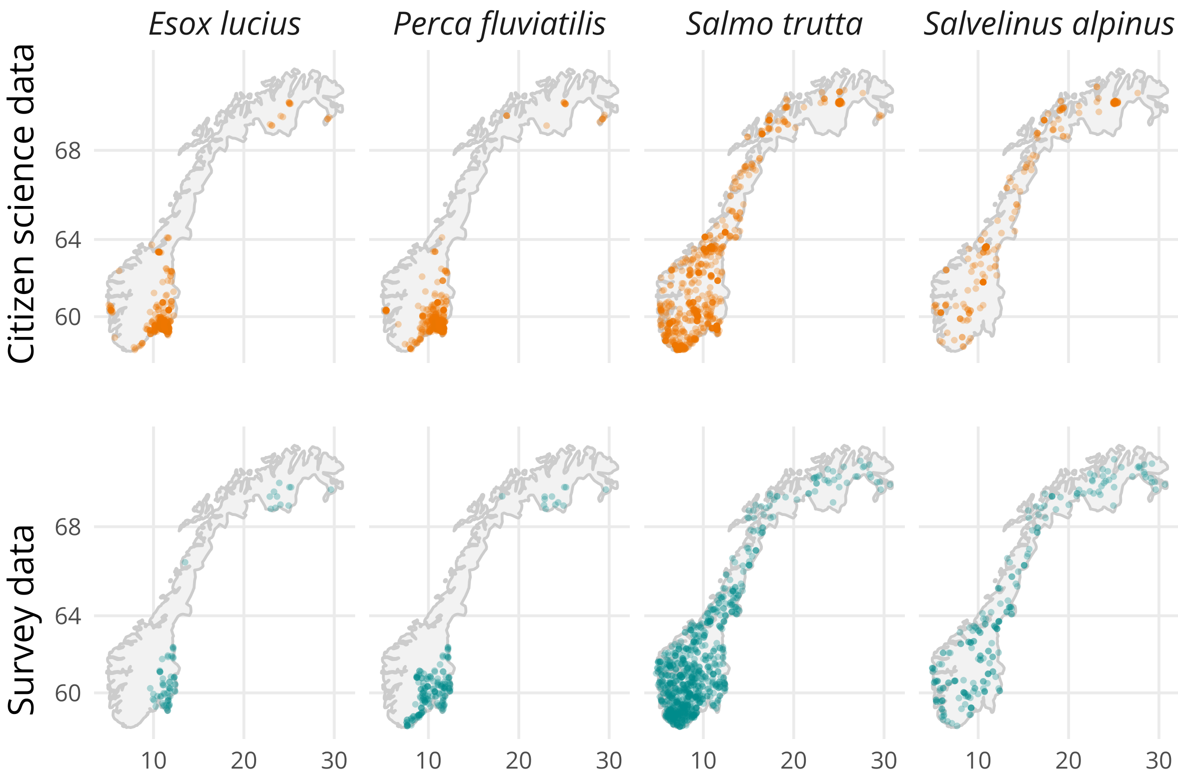
For this model, we have two observation sets, one which is downloaded from GBIF and one that is a survey dataset (see separate document for download instructions).

```r
fishes <- c("Esox_lucius", "Perca_fluviatilis", "Salmo_trutta", "Salvelinus_alpinus")

survey <- readRDS("data/survey_clean.rds") %>%
  filter(species %in% fishes) %>%
  st_as_sf(coords =c("decimalLongitude", "decimalLatitude"),
           crs = '+proj=lonlat +zone=32 +datum=WGS84 +units=m +no_defs') %>%
  mutate(decimalLongitude = st_coordinates(.)[,1], decimalLatitude = st_coordinates(.)[,2]) %>%
  st_transform(proj) %>%
st_intersects(., norway.poly.simp, sparse = FALSE)[,1]
artsobs <- readRDS("data/artsobs_clean.rds") %>%
  filter(species %in% fishes) %>%
  st_as_sf(coords =c("decimalLongitude", "decimalLatitude"),
           crs = '+proj=lonlat +zone=32 +datum=WGS84 +units=m +no_defs') %>%
  mutate(decimalLongitude = st_coordinates(.)[,1], decimalLatitude = st_coordinates(.)[,2]) %>%
  st_transform(proj) %>%
  st_intersects(., norway.poly.simp, sparse = FALSE)[,1]
```

We can plot the observed data points:

```r
showtext_auto()
showtext_opts(dpi = 300)
f1 <- "Open sans"
font_add_google(f1, f1)

norway <- ggplot2::map_data("world", region = "Norway(?!:Svalbard)")
norway <- setdiff(norway, dplyr::filter(norway, subregion == "Jan Mayen"))

p_artsobs <- ggplot(artsobs, aes(x = decimalLongitude, y = decimalLatitude)) +
  geom_polygon(data = norway, aes(long, lat, group = group),
               color="grey80", fill = "grey95") +
  geom_point(color = "darkorange2", size = 0.5, alpha = 0.3) +
```

```r
  facet_wrap(~species, nrow = 1,
             labeller = labeller(species = function(string) sub("_", " ", string))) +
  coord_map() +
  labs(tag = "Citizen science data") +
  theme_minimal() +
  theme(text = element_text(family = f1),
        strip.text = element_text(family = f1, size = 12, face = "italic"),
        plot.tag = element_text(angle = 90, hjust = 0.5),
        plot.tag.position = c(-0.03, 0.45),
        legend.position = "none",
        axis.title = element_blank(),
        axis.text.x = element_blank())

p_survey <- ggplot(survey %>% filter(occurrenceStatus == 1),
       aes(x = decimalLongitude, y = decimalLatitude)) +
  geom_polygon(data = norway, aes(long, lat, group = group),
               color="grey80", fill = "grey95") +
  geom_jitter(color = "darkcyan", size = 0.5, alpha = 0.3) +
  facet_wrap(~species, nrow = 1) +
  coord_map() +
  xlab("Longitude") +
  ylab("Latitude") +
  labs(tag = "Survey data") +
  theme_minimal() +
  theme(text = element_text(family = f1),
        plot.tag = element_text(angle = 90, hjust = 0.5),
        plot.tag.position = c(-0.03, 0.45),
        legend.position = "none",
        axis.title = element_blank(),
        strip.text = element_blank(),
        plot.margin = margin(l = 30))

p_artsobs / p_survey
```




```
ggsave("figures/presence_points.pdf", height = 5, width = 8)
ggsave("figures/presence_points.png", height = 5, width = 8)
```

Separate models for the two datasets

For the presence/absence survey data, we use a Bernoulli distribution, where the presence probability for species $j \in \{$*Salmo trutta, Perca fluviatilis, Esox lucius, Salvelinus alpinus*$\}$ depends on some covariates $x(s)$, along with a spatial field $\xi_j(s)$:

$$Y_{PA,j}(s_i) \sim \text{Bernoulli}(p_{PA,j}(s_i))$$
$$\text{cloglog}(p_{PA,j}(s_i)) = \alpha_{PA,j} + x(s_i)^T \beta_j + \xi_j(s_i).$$

We first prepare the model using the `startSpecies` function.

```
surveyModel <- startSpecies(
  survey,                              # Survey data
  Boundary = norway.poly,              # Boundary
  IPS = ipoints,                       # integration points
  spatialCovariates = cov_raster,      # Covariates
  speciesName = "species",             # The column containing species name
  speciesSpatial = 'copy',             # Copies across the species
  speciesIntercept = FALSE,            # Turn random intercept off for species
  pointsIntercept = FALSE,             # Turn dataset intercept off
  responsePA = "occurrenceStatus",     # Name of response column
  pointsSpatial = NULL,                # NULL since we use speciesSpatial
  Mesh = mesh,                         # inla mesh object
  Projection = proj                    # CRS for points and covariates
  )
```

We then specify priors for the spatial effects using `.$specifySpatial` and for the intercepts using `.$priorsFixed`.

```
for (fish in fishes) {

  surveyModel$specifySpatial(Species = fish,
                             prior.range = c(50, 0.01),
                             prior.sigma = c(0.5, 0.01),
                             constr = FALSE)

  surveyModel$priorsFixed(Effect = 'intercept',
                          Species = fish,
                          mean.linear = 0,
                          prec.linear = 10)
}

surveyModel$priorsFixed(Effect = 'HFP', mean.linear = 0, prec.linear = 0.1)
surveyModel$priorsFixed(Effect = 'SCI', mean.linear = 0, prec.linear = 0.1)
```

And fit the model using `fitISDM`.

```
surveyFit <- fitISDM(surveyModel,
                     options = list(num.threads = 4,
                       control.inla = list(int.strategy = 'ccd',
                                           cmin = 0,
                                           control.vb=list(enable=FALSE),
                                           diagonal = 1e-3,
                                           strategy = 'adaptive'),
                       safe = TRUE,
                       inla.mode = 'experimental'))

summary(surveyFit)
```

```
## Summary of 'modSpecies' object:
##
## inlabru version: 2.10.1
## INLA version: 24.05.13-1
##
## Types of data modelled:
##
## survey                Present absence
##
## Summary of the fixed effects for the species:
##
## Summary for Perca_fluviatilis:
##                                    mean        sd 0.025quant     0.5quant
## Perca_fluviatilis_intercept -0.71421103 0.2629424 -1.2294774 -0.714332836
## Perca_fluviatilis_HFP        0.00349559 0.1183907 -0.2286810  0.003502048
## Perca_fluviatilis_SCI        0.03000720 0.1140735 -0.1937173  0.030019243
##                             0.975quant        mode          kld
## Perca_fluviatilis_intercept -0.1982521 -0.71433370 9.839339e-11
## Perca_fluviatilis_HFP        0.2356355  0.00350209 3.866160e-11
## Perca_fluviatilis_SCI        0.2536630  0.03001929 4.771673e-11
##
## Summary for Salmo_trutta:
```

```
##                               mean         sd  0.025quant    0.5quant
## Salmo_trutta_intercept  0.46272660 0.21521212  0.03697789  0.46397094
## Salmo_trutta_HFP       -0.02749545 0.06512274 -0.15518116 -0.02750344
## Salmo_trutta_SCI        0.01465864 0.06805288 -0.11879776  0.01466073
##                        0.975quant        mode          kld
## Salmo_trutta_intercept  0.8814444  0.46398964 8.366963e-09
## Salmo_trutta_HFP        0.1002357 -0.02750350 1.803188e-11
## Salmo_trutta_SCI        0.1481032  0.01466073 2.858567e-11
##
## Summary for Salvelinus_alpinus:
##                                     mean         sd  0.025quant    0.5quant
## Salvelinus_alpinus_intercept -0.54735946 0.18377892 -0.90625112 -0.54792610
## Salvelinus_alpinus_HFP        0.12237216 0.09044759 -0.05497833  0.12236533
## Salvelinus_alpinus_SCI        0.09383392 0.09004905 -0.08273927  0.09382917
##                              0.975quant        mode          kld
## Salvelinus_alpinus_intercept -0.1852481 -0.54793002 2.471222e-09
## Salvelinus_alpinus_HFP        0.2997615  0.12236531 1.935013e-11
## Salvelinus_alpinus_SCI        0.2704342  0.09382915 2.279340e-11
##
## Summary for Esox_lucius:
##                             mean        sd 0.025quant    0.5quant 0.975quant
## Esox_lucius_intercept -0.8441590 0.2620239 -1.3576072 -0.84428575 -0.3299902
## Esox_lucius_HFP        0.4628622 0.1115507  0.2441337  0.46285409  0.6816371
## Esox_lucius_SCI        0.0199060 0.1459911 -0.2663721  0.01990682  0.3061794
##                              mode          kld
## Esox_lucius_intercept -0.84428673 1.073742e-10
## Esox_lucius_HFP        0.46285404 2.329318e-11
## Esox_lucius_SCI        0.01990683 5.073783e-11

## Time used:
##     Pre = 1.7, Running = 52.5, Post = 4.01, Total = 58.2
## Random effects:
##   Name     Model
##     Perca_fluviatilis_survey_spatial SPDE2 model
##    Salmo_trutta_survey_spatial SPDE2 model
##    Salvelinus_alpinus_survey_spatial SPDE2 model
##    Esox_lucius_survey_spatial SPDE2 model
##
## Model hyperparameters:
##                                          mean      sd 0.025quant 0.5quant
## Range for Perca_fluviatilis_survey_spatial  307.296  57.126    211.672  301.623
## Stdev for Perca_fluviatilis_survey_spatial    1.663   0.190      1.319    1.652
## Range for Salmo_trutta_survey_spatial       457.563 141.786    247.078  435.131
## Stdev for Salmo_trutta_survey_spatial         0.606   0.117      0.408    0.596
## Range for Salvelinus_alpinus_survey_spatial  97.896  17.931     68.024   96.069
## Stdev for Salvelinus_alpinus_survey_spatial   1.309   0.147      1.039    1.302
## Range for Esox_lucius_survey_spatial        487.751 108.441    312.264  475.049
## Stdev for Esox_lucius_survey_spatial          1.288   0.182      0.966    1.276
##                                          0.975quant    mode
## Range for Perca_fluviatilis_survey_spatial     435.805 289.852
## Stdev for Perca_fluviatilis_survey_spatial       2.068   1.632
## Range for Salmo_trutta_survey_spatial          799.456 392.176
## Stdev for Salmo_trutta_survey_spatial            0.866   0.575
## Range for Salvelinus_alpinus_survey_spatial    138.380  92.174
```

```
## Stdev for Salvelinus_alpinus_survey_spatial      1.619   1.290
## Range for Esox_lucius_survey_spatial           737.047 449.310
## Stdev for Esox_lucius_survey_spatial             1.681   1.251
##
## Deviance Information Criterion (DIC) ...............: 1608.50
## Deviance Information Criterion (DIC, saturated) ....: 1605.78
## Effective number of parameters .....................: 188.51
##
## Watanabe-Akaike information criterion (WAIC) ...: 1587.73
## Effective number of parameters .................: 141.91
##
## Marginal log-Likelihood:  -974.40
##  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)')
```

```
saveRDS(surveyFit, "results/surveyModel.rds")
```

The presence-only data is fitted with a Poisson point process model, where the intensity depends on the same covariates $x(s)$ and the same spatial field $\xi_j(s)$. In the ISDM, we will add an additional spatial field $\xi_{\text{bias}}(s)$ that is unique to the citizen science data, but shared across all fish species:

$$Y_{PO,j}(s_i) \sim \text{Poisson}(e^{\eta_{PO,j}(s_i)})$$
$$\eta_{PO,j}(s_i) = \alpha_{PO,j} + x(s)^T\beta_j + \xi_j(s_i) + \xi_{\text{bias}}(s_i).$$

```
artsobsModel <- startSpecies(
  artsobs,                          # Citizen science data
  Boundary = norway.poly,           # Boundary
  IPS = ipoints,                    # integration points
  spatialCovariates = cov_raster,   # Covariates
  speciesName = "species",          # The column containing species name
  speciesSpatial = 'copy',          # Copies across the species
  speciesIntercept = FALSE,         # Turn random intercept off for species
  pointsIntercept = FALSE,          # Turn dataset intercept off
  pointsSpatial = NULL,             # NULL since we use speciesSpatial
  Mesh = mesh,                      # inla mesh object
  Projection = proj                 # CRS for points and covariates
  )
```

```
for (fish in fishes) {

  artsobsModel$specifySpatial(Species = fish,
                              prior.range = c(50, 0.01),
                              prior.sigma = c(0.5, 0.01),
                              constr = FALSE)

  artsobsModel$priorsFixed(Effect = 'intercept',
                           Species = fish,
                           mean.linear = 0,
                           prec.linear = 10)
}

artsobsModel$priorsFixed(Effect = 'HFP', mean.linear = 0, prec.linear = 0.1)
artsobsModel$priorsFixed(Effect = 'SCI', mean.linear = 0, prec.linear = 0.1)
```

```r
artsobsFit <- fitISDM(artsobsModel,
                      options = list(num.threads = 4,
                        control.inla = list(int.strategy = 'ccd',
                                            cmin = 0,
                                            control.vb=list(enable=FALSE),
                                            diagonal = 1e-3,
                                            strategy = 'gaussian'),
                        safe = TRUE,
                        inla.mode = 'experimental'))

summary(artsobsFit)
```

```
## Summary of 'modSpecies' object:
##
## inlabru version: 2.10.1
## INLA version: 24.05.13-1
##
## Types of data modelled:
##
## artsobs                   Present only
##
## Summary of the fixed effects for the species:
##
## Summary for Salvelinus_alpinus:
##                                  mean         sd 0.025quant   0.5quant
## Salvelinus_alpinus_intercept -2.2986024 0.27078963 -2.8294055 -2.2986683
## Salvelinus_alpinus_HFP       -1.0152689 0.14737791 -1.3072065 -1.0142352
## Salvelinus_alpinus_SCI        0.2695658 0.07027954  0.1318143  0.2695432
##                              0.975quant       mode          kld
## Salvelinus_alpinus_intercept -1.7674243 -2.2986687 7.193901e-11
## Salvelinus_alpinus_HFP       -0.7291660 -1.0142164 1.219829e-08
## Salvelinus_alpinus_SCI        0.4074458  0.2695431 5.979181e-11
##
## Summary for Esox_lucius:
##                            mean         sd 0.025quant   0.5quant  0.975quant
## Esox_lucius_intercept -2.4111020 0.27242719 -2.9451887 -2.4111427 -1.87678364
## Esox_lucius_HFP        0.6733824 0.05606802  0.5639689  0.6731929  0.78386861
## Esox_lucius_SCI       -0.2291887 0.07848890 -0.3833347 -0.2291094 -0.07549434
##                            mode          kld
## Esox_lucius_intercept -2.4111430 6.129350e-11
## Esox_lucius_HFP        0.6731905 2.898393e-09
## Esox_lucius_SCI       -0.2291090 2.763500e-10
##
## Summary for Salmo_trutta:
##                              mean         sd  0.025quant    0.5quant
## Salmo_trutta_intercept -2.19819141 0.26549284 -2.71838432 -2.19833557
## Salmo_trutta_HFP        0.15301392 0.04222593  0.07020865  0.15301483
## Salmo_trutta_SCI        0.04398188 0.04395481 -0.04221223  0.04398257
##                        0.975quant        mode          kld
## Salmo_trutta_intercept -1.6771802 -2.19833699 1.348579e-10
## Salmo_trutta_HFP        0.2358140  0.15301484 3.989725e-11
## Salmo_trutta_SCI        0.1301721  0.04398257 4.585826e-11
##
## Summary for Perca_fluviatilis:
```

```
##                                    mean         sd 0.025quant     0.5quant
## Perca_fluviatilis_intercept -2.408427304 0.27235041 -2.9423561 -2.408470618
## Perca_fluviatilis_HFP        0.671372335 0.05078751  0.5722838  0.671193704
## Perca_fluviatilis_SCI        0.002112146 0.06141227 -0.1183818  0.002134226
##                             0.975quant         mode          kld
## Perca_fluviatilis_intercept -1.8742521 -2.408470892 6.200760e-11
## Perca_fluviatilis_HFP        0.7714728  0.671191413 3.140689e-09
## Perca_fluviatilis_SCI        0.1224803  0.002134311 5.084684e-11

## Time used:
##     Pre = 1.98, Running = 104, Post = 6.83, Total = 113
## Random effects:
##   Name     Model
##     Salvelinus_alpinus_artsobs_spatial SPDE2 model
##    Esox_lucius_artsobs_spatial SPDE2 model
##    Salmo_trutta_artsobs_spatial SPDE2 model
##    Perca_fluviatilis_artsobs_spatial SPDE2 model
##
## Model hyperparameters:
##                                               mean     sd 0.025quant 0.5quant
## Range for Salvelinus_alpinus_artsobs_spatial 189.72 15.867     160.56   189.01
## Stdev for Salvelinus_alpinus_artsobs_spatial   3.41  0.223       2.99     3.40
## Range for Esox_lucius_artsobs_spatial        245.80 20.765     207.59   244.88
## Stdev for Esox_lucius_artsobs_spatial          3.23  0.236       2.79     3.22
## Range for Salmo_trutta_artsobs_spatial       209.31 17.385     177.82   208.38
## Stdev for Salmo_trutta_artsobs_spatial         2.71  0.192       2.36     2.71
## Range for Perca_fluviatilis_artsobs_spatial  248.06 22.458     207.64   246.76
## Stdev for Perca_fluviatilis_artsobs_spatial    3.21  0.223       2.79     3.20
##                                             0.975quant   mode
## Range for Salvelinus_alpinus_artsobs_spatial     223.00 187.48
## Stdev for Salvelinus_alpinus_artsobs_spatial       3.87   3.39
## Range for Esox_lucius_artsobs_spatial            289.29 242.96
## Stdev for Esox_lucius_artsobs_spatial              3.71   3.20
## Range for Salmo_trutta_artsobs_spatial           246.22 206.06
## Stdev for Salmo_trutta_artsobs_spatial             3.11   2.69
## Range for Perca_fluviatilis_artsobs_spatial      295.99 243.60
## Stdev for Perca_fluviatilis_artsobs_spatial        3.67   3.18
##
## Deviance Information Criterion (DIC) ...............: -37108.25
## Deviance Information Criterion (DIC, saturated) ....: NA
## Effective number of parameters .....................: -41986.51
##
## Watanabe-Akaike information criterion (WAIC) ...: 10719.10
## Effective number of parameters .................: 2946.19
##
## Marginal log-Likelihood:  -25078.88
##  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)')
```

```r
saveRDS(artsobsFit, "results/artsObsModel.rds")
```

Joint model for four fish species

Now we will fit two IDMS, one with the two datasets combined, and another with an additional spatial field for the citizen science data. Both these models will have four shared fields, shared across the data sets (survey/citizen science), but separate for each fish species. Since we have two data sets and four species, that means that we in total have eight sub-models.

```
ISDMsetup <- startSpecies(
  survey,                              # Survey data
  artsobs,                             # Citizen science data
  Boundary = norway.poly,              # Boundary
  IPS = ipoints,                       # integration points
  spatialCovariates = cov_raster,      # Covariates
  speciesName = "species",             # The column containing species name
  speciesSpatial = 'copy',             # Copies across the species
  speciesIntercept = FALSE,            # Turn random intercept off for species
  pointsIntercept = FALSE,             # Turn dataset intercept off
  responsePA = "occurrenceStatus",     # Name of response column
  pointsSpatial = NULL,                # NULL since we use speciesSpatial
  Mesh = mesh,                         # inla mesh object
  Projection = proj                    # CRS for points and covariates
  )
```

```
for (fish in fishes) {

  ISDMsetup$specifySpatial(Species = fish,
                           prior.range = c(50, 0.01),
                           prior.sigma = c(0.5, 0.01),
                           constr = FALSE)

  ISDMsetup$priorsFixed(Effect = 'intercept',
                        Species = fish,
                        mean.linear = 0,
                        prec.linear = 10)

}

ISDMsetup$priorsFixed(Effect = 'HFP', mean.linear = 0, prec.linear = 0.1)
ISDMsetup$priorsFixed(Effect = 'SCI', mean.linear = 0, prec.linear = 0.1)
```

For the species specific fields, the default in PointedSDMs is that these are allowed to be different up to a scaling factor (named beta in INLA) across the data sets. That means that for instance the trout-specific spatial field for the citizen science data set is equal to beta times the trout-specific spatial field for the survey data. In practice, this has to do with the copy-option in INLA. But in our model we want them to be the same, not to vary by a factor. So we manually change this using the `$specifyRandom` function, by setting `hyper = list(beta = list(fixed = TRUE))` for each of the four citizen science fields.

```
ISDMsetup$specifyRandom(copyModel = list(beta = list(fixed = TRUE)))
ISDMsetup$changeComponents()
```

```
## Model components:
## ~-1 + Perca_fluviatilis_survey_spatial(main = geometry, model = Perca_fluviatilis_survey_field) +
##     Salmo_trutta_survey_spatial(main = geometry, model = Salmo_trutta_survey_field) +
##     Salvelinus_alpinus_survey_spatial(main = geometry, model = Salvelinus_alpinus_survey_field) +
##     Esox_lucius_survey_spatial(main = geometry, model = Esox_lucius_survey_field) +
##     Perca_fluviatilis_artsobs_spatial(main = geometry, copy = "Perca_fluviatilis_survey_spatial",
```

```
##         hyper = list(beta = list(fixed = FALSE))) + Salmo_trutta_artsobs_spatial(main = geometry,
##     copy = "Salmo_trutta_survey_spatial", hyper = list(beta = list(fixed = FALSE))) +
##     Salvelinus_alpinus_artsobs_spatial(main = geometry, copy = "Salvelinus_alpinus_survey_spatial",
##         hyper = list(beta = list(fixed = FALSE))) + Esox_lucius_artsobs_spatial(main = geometry,
##     copy = "Esox_lucius_survey_spatial", hyper = list(beta = list(fixed = FALSE))) +
##     Esox_lucius_intercept(1, mean.linear = 0, prec.linear = 10) +
##     Perca_fluviatilis_intercept(1, mean.linear = 0, prec.linear = 10) +
##     Salmo_trutta_intercept(1, mean.linear = 0, prec.linear = 10) +
##     Salvelinus_alpinus_intercept(1, mean.linear = 0, prec.linear = 10) +
##     Perca_fluviatilis_HFP(main = Perca_fluviatilis_HFP, model = "linear",
##         mean.linear = 0, prec.linear = 0.1) + Salmo_trutta_HFP(main = Salmo_trutta_HFP,
##     model = "linear", mean.linear = 0, prec.linear = 0.1) + Salvelinus_alpinus_HFP(main = Salvelinus_
##     model = "linear", mean.linear = 0, prec.linear = 0.1) + Esox_lucius_HFP(main = Esox_lucius_HFP,
##     model = "linear", mean.linear = 0, prec.linear = 0.1) + Perca_fluviatilis_SCI(main = Perca_fluvia
##     model = "linear", mean.linear = 0, prec.linear = 0.1) + Salmo_trutta_SCI(main = Salmo_trutta_SCI
##     model = "linear", mean.linear = 0, prec.linear = 0.1) + Salvelinus_alpinus_SCI(main = Salvelinus_
##     model = "linear", mean.linear = 0, prec.linear = 0.1) + Esox_lucius_SCI(main = Esox_lucius_SCI,
##     model = "linear", mean.linear = 0, prec.linear = 0.1)
## <environment: 0x55b382ddb370>
```

We may look at which terms are included in each of the eight sub-models by calling `$updateFormula` with the data sets as the arguments.

```
ISDMsetup$updateFormula(datasetName = "survey")
```

```
## $Perca_fluviatilis
## occurrenceStatus ~ Perca_fluviatilis_SCI + Perca_fluviatilis_HFP +
##     Perca_fluviatilis_survey_spatial + Perca_fluviatilis_intercept
## <environment: 0x55b3e6fbdea0>
##
## $Salmo_trutta
## occurrenceStatus ~ Salmo_trutta_SCI + Salmo_trutta_HFP + Salmo_trutta_survey_spatial +
##     Salmo_trutta_intercept
## <environment: 0x55b3e6fbdea0>
##
## $Salvelinus_alpinus
## occurrenceStatus ~ Salvelinus_alpinus_SCI + Salvelinus_alpinus_HFP +
##     Salvelinus_alpinus_survey_spatial + Salvelinus_alpinus_intercept
## <environment: 0x55b3e6fbdea0>
##
## $Esox_lucius
## occurrenceStatus ~ Esox_lucius_SCI + Esox_lucius_HFP + Esox_lucius_survey_spatial +
##     Esox_lucius_intercept
## <environment: 0x55b3e6fbdea0>
```

```
ISDMsetup$updateFormula(datasetName = "artsobs")
```

```
## $Salvelinus_alpinus
## geometry ~ Salvelinus_alpinus_SCI + Salvelinus_alpinus_HFP +
##     Salvelinus_alpinus_artsobs_spatial + Salvelinus_alpinus_intercept
## <environment: 0x55b3e6fbdea0>
##
## $Esox_lucius
## geometry ~ Esox_lucius_SCI + Esox_lucius_HFP + Esox_lucius_artsobs_spatial +
##     Esox_lucius_intercept
## <environment: 0x55b3e6fbdea0>
```

```
##
## $Salmo_trutta
## geometry ~ Salmo_trutta_SCI + Salmo_trutta_HFP + Salmo_trutta_artsobs_spatial +
##     Salmo_trutta_intercept
## <environment: 0x55b3e6fbdea0>
##
## $Perca_fluviatilis
## geometry ~ Perca_fluviatilis_SCI + Perca_fluviatilis_HFP + Perca_fluviatilis_artsobs_spatial +
##     Perca_fluviatilis_intercept
## <environment: 0x55b3e6fbdea0>
```

```
ISDM <- fitISDM(ISDMsetup,
                      options = list(num.threads = 4,
                        control.inla = list(int.strategy = 'ccd',
                                            cmin = 0,
                                            control.vb=list(enable=FALSE),
                                            diagonal = 1e-3,
                                            strategy = 'gaussian'),
                        safe = TRUE,
                        inla.mode = 'experimental'))
```

We may then examine the model summary and save the model for future use.

```
summary(ISDM)
```

```
## Summary of 'modSpecies' object:
##
## inlabru version: 2.10.1
## INLA version: 24.05.13-1
##
## Types of data modelled:
##
## survey                 Present absence
## artsobs                   Present only
##
## Summary of the fixed effects for the species:
##
## Summary for Perca_fluviatilis:
##                                    mean         sd 0.025quant    0.5quant
## Perca_fluviatilis_intercept  0.76532600 0.12907567  0.5126458  0.76517825
## Perca_fluviatilis_HFP        0.54254319 0.04117692  0.4618744  0.54251621
## Perca_fluviatilis_SCI       -0.01856535 0.05163308 -0.1198314 -0.01855939
##                             0.975quant        mode          kld
## Perca_fluviatilis_intercept 1.01884360  0.76517642 4.048103e-10
## Perca_fluviatilis_HFP       0.62336537  0.54251602 1.514439e-10
## Perca_fluviatilis_SCI       0.08266671 -0.01855936 4.688429e-11
##
## Summary for Salmo_trutta:
##                              mean         sd  0.025quant    0.5quant
## Salmo_trutta_intercept -1.28996362 0.04569968 -1.37842376 -1.29002936
## Salmo_trutta_HFP        0.12768190 0.02730173  0.07414194  0.12768322
## Salmo_trutta_SCI        0.01514846 0.03390182 -0.05133040  0.01514863
##                        0.975quant        mode          kld
## Salmo_trutta_intercept -1.20124643 -1.29007059 7.229895e-08
## Salmo_trutta_HFP        0.18121433  0.12768323 5.292562e-11
```

```
## Salmo_trutta_SCI        0.08162637  0.01514863 5.216595e-11
##
## Summary for Salvelinus_alpinus:
##                                   mean         sd  0.025quant   0.5quant
## Salvelinus_alpinus_intercept -3.2923239 0.06385989 -3.41726548 -3.2923067
## Salvelinus_alpinus_HFP       -0.2209171 0.06219484 -0.34293195 -0.2208977
## Salvelinus_alpinus_SCI        0.1699334 0.04836064  0.07510407  0.1699328
##                               0.975quant       mode         kld
## Salvelinus_alpinus_intercept -3.16749223 -3.2923108 7.279112e-09
## Salvelinus_alpinus_HFP       -0.09901261 -0.2208975 7.006881e-11
## Salvelinus_alpinus_SCI        0.26476565  0.1699328 5.234281e-11
##
## Summary for Esox_lucius:
##                             mean         sd 0.025quant   0.5quant  0.975quant
## Esox_lucius_intercept -0.8293970 0.18753973 -1.1967987 -0.8295209 -0.46129183
## Esox_lucius_HFP        0.6444211 0.04777995  0.5508690  0.6443713  0.73825632
## Esox_lucius_SCI       -0.2074947 0.07083151 -0.3464598 -0.2074708 -0.06866574
##                             mode          kld
## Esox_lucius_intercept -0.8295220 1.543471e-10
## Esox_lucius_HFP        0.6443709 3.191612e-10
## Esox_lucius_SCI       -0.2074707 6.734711e-11

## Time used:
##     Pre = 2.65, Running = 614, Post = 26.4, Total = 643
## Random effects:
##   Name     Model
##     Perca_fluviatilis_survey_spatial SPDE2 model
##    Salmo_trutta_survey_spatial SPDE2 model
##    Salvelinus_alpinus_survey_spatial SPDE2 model
##    Esox_lucius_survey_spatial SPDE2 model
##    Salvelinus_alpinus_artsobs_spatial Copy
##    Esox_lucius_artsobs_spatial Copy
##    Salmo_trutta_artsobs_spatial Copy
##    Perca_fluviatilis_artsobs_spatial Copy
##
## Model hyperparameters:
##                                               mean     sd 0.025quant 0.5quant
## Range for Perca_fluviatilis_survey_spatial  369.98 40.911     297.68   367.18
## Stdev for Perca_fluviatilis_survey_spatial    1.29  0.127       1.06     1.28
## Range for Salmo_trutta_survey_spatial       244.72 22.215     204.68   243.46
## Stdev for Salmo_trutta_survey_spatial         1.34  0.119       1.11     1.33
## Range for Salvelinus_alpinus_survey_spatial 204.76 18.720     171.16   203.66
## Stdev for Salvelinus_alpinus_survey_spatial   1.44  0.126       1.21     1.43
## Range for Esox_lucius_survey_spatial        317.75 33.853     257.86   315.45
## Stdev for Esox_lucius_survey_spatial          1.29  0.130       1.05     1.28
## Beta for Salvelinus_alpinus_artsobs_spatial  -2.04  0.145      -2.33    -2.04
## Beta for Esox_lucius_artsobs_spatial          3.18  0.156       2.87     3.18
## Beta for Salmo_trutta_artsobs_spatial        -2.25  0.143      -2.54    -2.25
## Beta for Perca_fluviatilis_artsobs_spatial    3.61  0.135       3.35     3.61
##                                             0.975quant   mode
## Range for Perca_fluviatilis_survey_spatial      458.56 360.52
## Stdev for Perca_fluviatilis_survey_spatial        1.56   1.26
## Range for Salmo_trutta_survey_spatial           292.07 240.41
## Stdev for Salmo_trutta_survey_spatial             1.58   1.32
```

```
## Range for Salvelinus_alpinus_survey_spatial     244.81 200.92
## Stdev for Salvelinus_alpinus_survey_spatial       1.71   1.42
## Range for Esox_lucius_survey_spatial            391.00 309.88
## Stdev for Esox_lucius_survey_spatial              1.56   1.26
## Beta for Salvelinus_alpinus_artsobs_spatial      -1.76  -2.04
## Beta for Esox_lucius_artsobs_spatial              3.49   3.18
## Beta for Salmo_trutta_artsobs_spatial            -1.98  -2.24
## Beta for Perca_fluviatilis_artsobs_spatial        3.88   3.61
##
## Deviance Information Criterion (DIC) ...............: -35106.59
## Deviance Information Criterion (DIC, saturated) ....: NA
## Effective number of parameters .....................: -42302.23
##
## Watanabe-Akaike information criterion (WAIC) ...: 12757.62
## Effective number of parameters .................: 2805.11
##
## Marginal log-Likelihood:  -26622.83
##  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)')
```

```r
saveRDS(ISDM, "results/ISDMModel.rds")
```

This model has the species specific spatial fields, but we also want a bias field that is shared across the species. We add this using `$addBias`.

```r
ISDMsetup$addBias("artsobs", copyModel = FALSE)
ISDMsetup$specifySpatial(Bias = 'artsobs',
                         prior.range = c(50, 0.01),
                         prior.sigma = c(0.5, 0.01))
```

```r
ISDMbias <- fitISDM(ISDMsetup,
                    options = list(num.threads = 4,
                      control.inla = list(int.strategy = 'ccd',
                                          cmin = 0,
                                          control.vb=list(enable=FALSE),
                                          diagonal = 1e-3,
                                          strategy = 'gaussian'),
                      safe = TRUE,
                      inla.mode = 'experimental'))

summary(ISDMbias)
```

```
## Summary of 'modSpecies' object:
##
## inlabru version: 2.10.1
## INLA version: 24.05.13-1
##
## Types of data modelled:
##
## survey                 Present absence
## artsobs                   Present only
##
## Summary of the fixed effects for the species:
##
## Summary for Perca_fluviatilis:
```

```
##                                    mean         sd 0.025quant    0.5quant
## Perca_fluviatilis_intercept -1.4496786 0.20544723 -1.8523350 -1.44976123
## Perca_fluviatilis_HFP        0.6851287 0.04507680  0.5968613  0.68508443
## Perca_fluviatilis_SCI       -0.0457123 0.05594054 -0.1554272 -0.04570534
##                              0.975quant        mode          kld
## Perca_fluviatilis_intercept -1.04655159 -1.44976154 5.511039e-11
## Perca_fluviatilis_HFP        0.77364782  0.68508415 2.861709e-10
## Perca_fluviatilis_SCI        0.06396302 -0.04570531 5.080971e-11
##
## Summary for Salmo_trutta:
##                              mean         sd  0.025quant   0.5quant 0.975quant
## Salmo_trutta_intercept 0.01132920 0.12721862 -0.24025430 0.01204810 0.25883250
## Salmo_trutta_HFP       0.12541103 0.03414523  0.05847313 0.12540470 0.19238496
## Salmo_trutta_SCI       0.02048531 0.03728854 -0.05263589 0.02048593 0.09360301
##                              mode          kld
## Salmo_trutta_intercept 0.01205462 8.004471e-09
## Salmo_trutta_HFP       0.12540467 5.853385e-11
## Salmo_trutta_SCI       0.02048593 5.211818e-11
##
## Summary for Salvelinus_alpinus:
##                                    mean         sd  0.025quant   0.5quant
## Salvelinus_alpinus_intercept -0.8170756 0.15155506 -1.11379618 -0.8172576
## Salvelinus_alpinus_HFP       -0.3408945 0.07736402 -0.49280822 -0.3408224
## Salvelinus_alpinus_SCI        0.1729362 0.05538737  0.06433409  0.1729334
##                              0.975quant       mode          kld
## Salvelinus_alpinus_intercept -0.5193193 -0.8172585 3.693469e-10
## Salvelinus_alpinus_HFP       -0.1893911 -0.3408219 2.551701e-10
## Salvelinus_alpinus_SCI        0.2815540  0.1729334 5.033798e-11
##
## Summary for Esox_lucius:
##                             mean         sd 0.025quant   0.5quant  0.975quant
## Esox_lucius_intercept -1.6637612 0.22187841 -2.0975707 -1.6642170 -1.22736291
## Esox_lucius_HFP        0.8036487 0.05523057  0.6955821  0.8035645  0.91219453
## Esox_lucius_SCI       -0.1988648 0.07318098 -0.3424467 -0.1988375 -0.05543812
##                             mode          kld
## Esox_lucius_intercept -1.6642209 1.098659e-09
## Esox_lucius_HFP        0.8035639 6.247668e-10
## Esox_lucius_SCI       -0.1988374 7.352872e-11

## Time used:
##     Pre = 2.62, Running = 10952, Post = 52.6, Total = 11007
## Random effects:
##   Name     Model
##     Perca_fluviatilis_survey_spatial SPDE2 model
##    Salmo_trutta_survey_spatial SPDE2 model
##    Salvelinus_alpinus_survey_spatial SPDE2 model
##    Esox_lucius_survey_spatial SPDE2 model
##    artsobs_biasField SPDE2 model
##    Salvelinus_alpinus_artsobs_spatial Copy
##    Esox_lucius_artsobs_spatial Copy
##    Salmo_trutta_artsobs_spatial Copy
##    Perca_fluviatilis_artsobs_spatial Copy
##
## Model hyperparameters:
```

```
##                                            mean     sd 0.025quant 0.5quant
## Range for Perca_fluviatilis_survey_spatial  196.531 31.514    142.227  193.909
## Stdev for Perca_fluviatilis_survey_spatial    1.876  0.190      1.524    1.869
## Range for Salmo_trutta_survey_spatial       453.252 97.887    296.234  441.362
## Stdev for Salmo_trutta_survey_spatial         0.720  0.123      0.506    0.711
## Range for Salvelinus_alpinus_survey_spatial  77.111 10.126     58.955   76.505
## Stdev for Salvelinus_alpinus_survey_spatial   1.244  0.124      1.018    1.238
## Range for Esox_lucius_survey_spatial        162.180 25.300    118.383  160.141
## Stdev for Esox_lucius_survey_spatial          1.412  0.153      1.133    1.404
## Range for artsobs_biasField                 226.693 15.765    197.235  226.149
## Stdev for artsobs_biasField                   3.568  0.213      3.169    3.562
## Beta for Salvelinus_alpinus_artsobs_spatial   0.883  0.101      0.684    0.884
## Beta for Esox_lucius_artsobs_spatial          1.209  0.117      0.980    1.208
## Beta for Salmo_trutta_artsobs_spatial         0.357  0.097      0.165    0.358
## Beta for Perca_fluviatilis_artsobs_spatial    0.633  0.067      0.501    0.632
##                                             0.975quant    mode
## Range for Perca_fluviatilis_survey_spatial     265.982 188.531
## Stdev for Perca_fluviatilis_survey_spatial       2.270   1.859
## Range for Salmo_trutta_survey_spatial          679.683 416.165
## Stdev for Salmo_trutta_survey_spatial            0.989   0.694
## Range for Salvelinus_alpinus_survey_spatial     98.751  75.401
## Stdev for Salvelinus_alpinus_survey_spatial      1.506   1.226
## Range for Esox_lucius_survey_spatial           217.750 155.970
## Stdev for Esox_lucius_survey_spatial             1.736   1.390
## Range for artsobs_biasField                    259.277 225.074
## Stdev for artsobs_biasField                      4.006   3.548
## Beta for Salvelinus_alpinus_artsobs_spatial      1.081   0.886
## Beta for Esox_lucius_artsobs_spatial             1.440   1.205
## Beta for Salmo_trutta_artsobs_spatial            0.547   0.360
## Beta for Perca_fluviatilis_artsobs_spatial       0.767   0.630
##
## Deviance Information Criterion (DIC) ...............: -36161.52
## Deviance Information Criterion (DIC, saturated) ....: NA
## Effective number of parameters .....................: -42102.75
##
## Watanabe-Akaike information criterion (WAIC) ...: 12261.95
## Effective number of parameters .................: 3180.64
##
## Marginal log-Likelihood:  -25470.34
##  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)')
```

```r
saveRDS(ISDMbias, "results/ISDMBiasModel.rds")
```

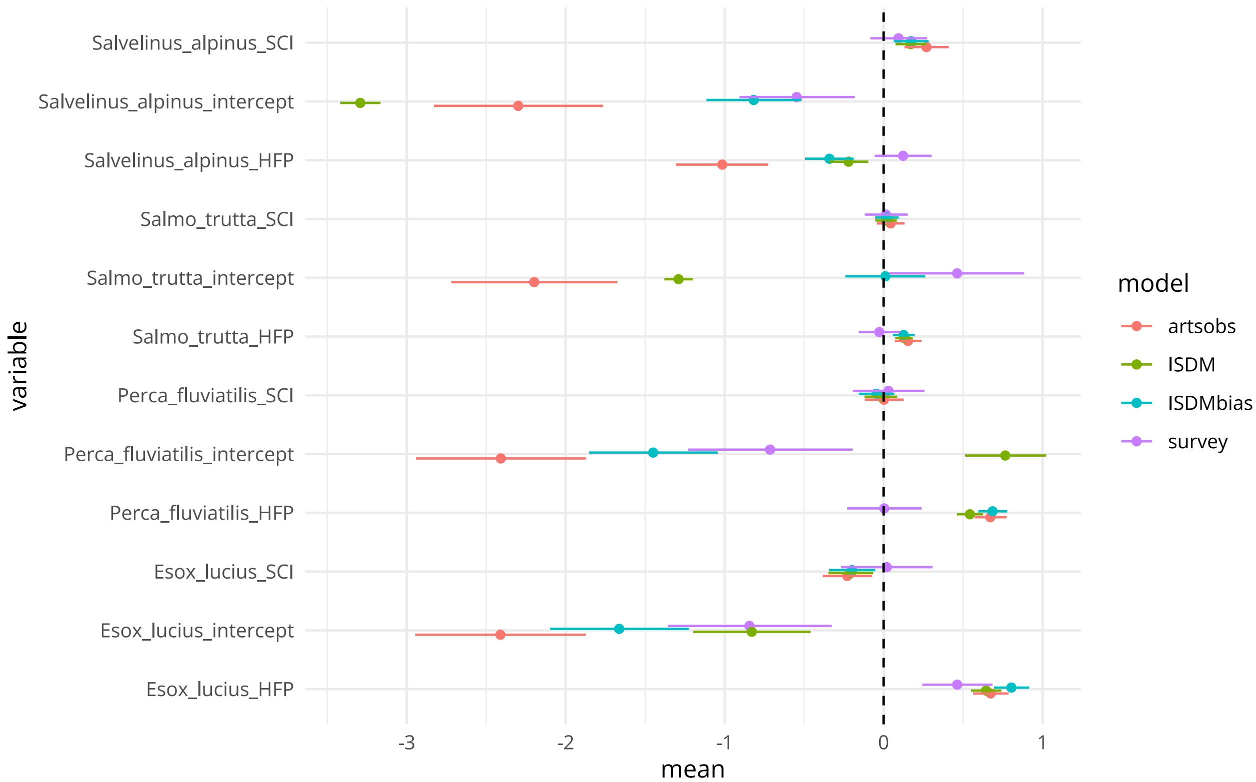
Predictions and plots

Once the model has been fit, we can look at the predictions from the species-specific shared fields and the bias field.

First we create a plot to show the coefficients and the associated credibility intervals for the four different models considered. The estimates and standard errors for the four models are relatively similar.

```r
makeData <- function(data, name) {
```

```
 data$summary.fixed$variable <- rownames(data$summary.fixed)
 data$summary.fixed$min <- data$summary.fixed$`0.025quant`
 data$summary.fixed$max <- data$summary.fixed$`0.975quant`
 data$summary.fixed$model <- name
 data$summary.fixed

}

plotData <- rbind(makeData(surveyFit, 'survey'),
                  makeData(artsobsFit, 'artsobs'),
                  makeData(ISDM, 'ISDM'),
                  makeData(ISDMbias, 'ISDMbias'))

ggplot(plotData, aes(x = mean, y = variable, col = model)) +
  geom_point(position=position_dodge(width=0.2)) +
  geom_errorbar(aes(x = mean, y = variable, xmin = min, xmax = max),
                position=position_dodge(width=0.2), width = 0) +
  geom_vline(xintercept = 0, lty = 2) +
  theme_minimal() +
  theme(text = element_text(family = f1),
        strip.text = element_text(family = f1, size = 12, face = "italic"),
        plot.tag = element_text(angle = 90, hjust = 0.5),
        plot.tag.position = c(-0.03, 0.45))
```




```
ggsave("figures/CIplots.png")
```

```
## Saving 8 x 5 in image
```

```
ggsave("figures/CIplots.pdf")
```

```
## Saving 8 x 5 in image
```

We define a function that will do species-specific predictions for the ISDM with a bias field, and save the species predictions, since these take a little time to compute.

```
predict_species <- function(model, species, predict_data){
  sharedfield <- predict(model,
                         data = predict_data,
                         spatial = TRUE,
                         species = species,
                         n.samples = 1000)
  file_name <- paste0("results/sharedfield_", species, ".rds")
  saveRDS(sharedfield, file_name)
  return(sharedfield)
}
```

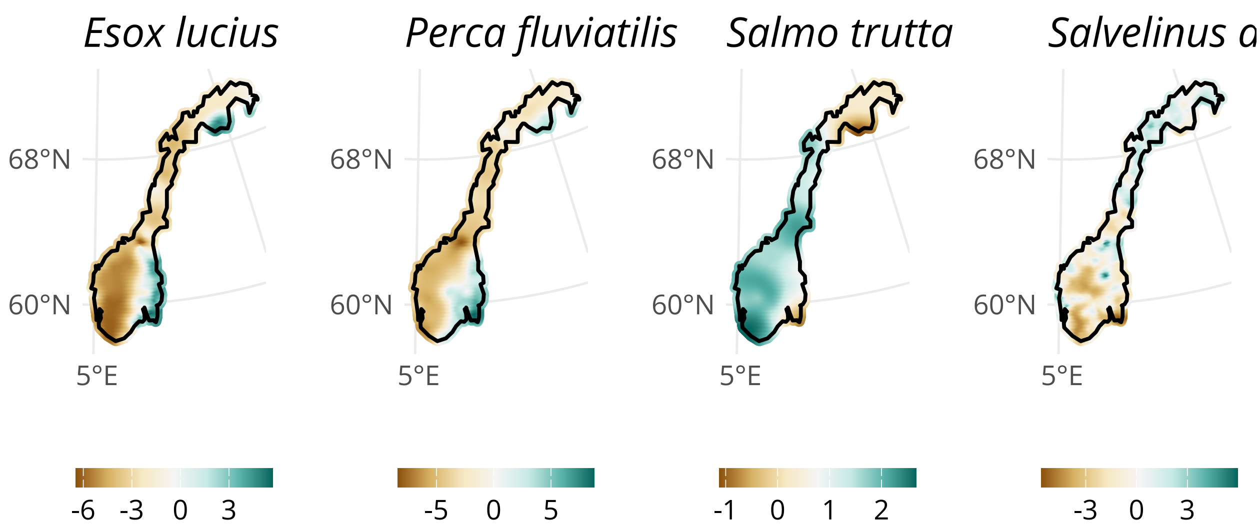
```
predData <- fm_pixels(mesh = mesh, mask = norway.poly, dims = c(450, 450))
prediction_list <- list()
for(fish in fishes) {
  prediction_list[[fish]] <- predict_species(
    model = ISDMbias,
    predict_data = predData,
    species = fish
  )
}

saveRDS(prediction_list, "results/Predictionlist.rds")
```

Once we have the predictions, we can make some plots. We similarly define a function that makes a plot for one species, and then run this for all four species.

```
NO <- st_boundary(st_as_sf(norway.poly))

plot_preferences <- list(scale_color_distiller(palette = "BrBG", direction = 1),
                         coord_sf(),
                         xlab(""), ylab(""),
                         scale_x_continuous(breaks = c(5, 25)),
                         scale_y_continuous(breaks = c(60, 68)),
                         theme_minimal(),
                         theme(text = element_text(family = f1),
                               #legend.title = element_text(size = 6),
                               axis.text=element_text(size=12),
                               legend.text = element_text(size = 12),
                               title = element_text(family = f1, size = 15, face = "italic"),
                               legend.key.height = unit(0.3, "cm"),
                               legend.title = element_blank(),
                               legend.position = "bottom")
)

plot_species <- function(predictions, species_to_plot, plot_preferences){

  p <- ggplot() +
    gg(predictions$speciesPredictions[[species_to_plot]],
       aes(col = mean)) +
```

```
    gg(NO, lwd = 0.8) +
    labs(title = sub("_", " ", species_to_plot)) +
    plot_preferences
  return(p)
}
```

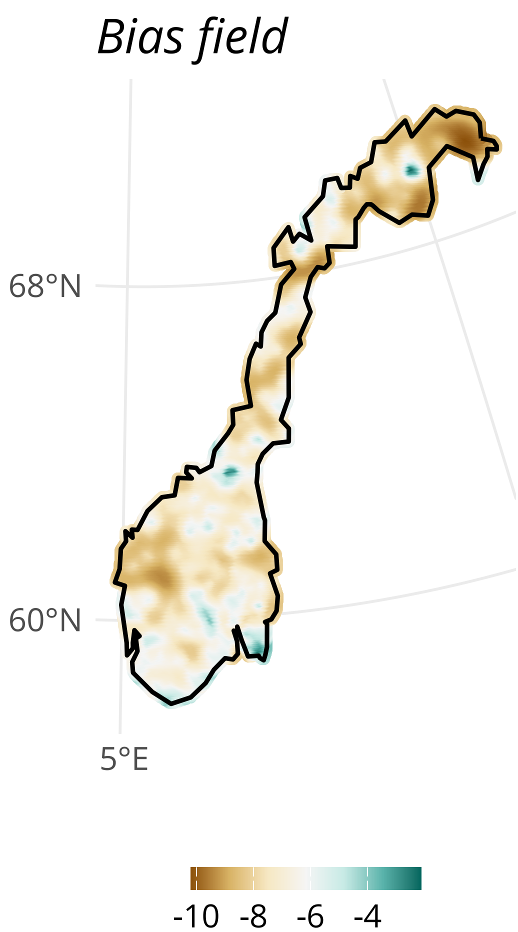
```
plot_list <- list()
for(fish in fishes) {
  plot_list[[fish]] <- plot_species(
    predictions = readRDS(paste0("results/sharedfield_", fish, ".rds")),
    species = fish,
    plot_preferences = plot_preferences)
}

patchwork::wrap_plots(plot_list, nrow = 1)
```




```
ggsave("figures/fishplot_four_species.pdf", height = 3*2, width = 8*2)
ggsave("figures/fishplot_four_species.png", height = 3*2, width = 8*2)
```

And finally we predict and plot the bias field, which is shared between all the fish, as it describes the human sampling more than the distribution of the fish.

```
fish_biasfield <- predict(ISDMbias,
                          data = predData,
                          bias = TRUE,
                          n.samples = 1000)

saveRDS(fish_biasfield, "results/ISDMbiasfield.rds")
fish_biasfield <- readRDS("results/ISDMbiasfield.rds")

ggplot() +
  gg(fish_biasfield$biasFields$artsobs,
     aes(col = mean)) +
  gg(NO, lwd = 0.8) +
  labs(title = "Bias field") +
  plot_preferences
```




```r
ggsave("figures/ISDMbiasfield.pdf", width = 3*2, height = 3*2)
```

Session info and runtime

This document took 8.26 hours to compile.

Current session info

```
- Session info ---------------------------------------------------------------------------------------
 setting  value
 version  R version 4.4.1 (2024-06-14)
 os       Ubuntu 20.04.1 LTS
 system   x86_64, linux-gnu
 ui       X11
 language (EN)
 collate  C.UTF-8
 ctype    C.UTF-8
 tz       Europe/Oslo
 date     2024-06-20
 pandoc   2.5 @ /usr/bin/ (via rmarkdown)

- Packages -------------------------------------------------------------------------------------------
 package      * version    date (UTC) lib source
 base64enc      0.1-3      2015-07-28 [1] CRAN (R 4.3.2)
```

```
 blockCV          3.1-3      2023-06-04 [1] CRAN (R 4.3.1)
 class            7.3-21     2023-01-23 [1] CRAN (R 4.2.2)
 classInt         0.4-7      2022-06-10 [1] CRAN (R 4.2.1)
 cli              3.6.1      2023-03-23 [1] CRAN (R 4.3.1)
 clipr            0.8.0      2022-02-22 [1] CRAN (R 4.3.2)
 codetools        0.2-20     2024-03-31 [1] CRAN (R 4.4.0)
 colorspace       2.0-3      2022-02-21 [1] CRAN (R 4.1.2)
 countrycode      1.6.0      2024-03-22 [1] CRAN (R 4.4.0)
 curl             4.3.3      2022-10-06 [1] CRAN (R 4.2.2)
 DBI              1.2.3      2024-06-02 [1] CRAN (R 4.4.0)
 desc             1.4.3      2023-12-10 [1] CRAN (R 4.3.2)
 details          0.3.0      2022-03-27 [1] CRAN (R 4.3.2)
 digest           0.6.31     2022-12-11 [1] CRAN (R 4.2.2)
 dplyr          * 1.1.3      2023-09-03 [1] CRAN (R 4.3.1)
 e1071            1.7-11     2022-06-07 [1] CRAN (R 4.2.1)
 evaluate         0.24.0     2024-06-10 [1] CRAN (R 4.4.0)
 fansi            1.0.2      2022-01-14 [1] CRAN (R 4.1.2)
 farver           2.1.1      2022-07-06 [1] CRAN (R 4.2.1)
 fastmap          1.1.1      2023-02-24 [1] CRAN (R 4.3.1)
 fmesher        * 0.1.5      2023-12-20 [1] CRAN (R 4.3.2)
 generics         0.1.3      2022-07-05 [1] CRAN (R 4.3.2)
 geojsonsf        2.0.3      2022-05-30 [1] CRAN (R 4.3.2)
 ggplot2        * 3.5.1      2024-04-23 [1] CRAN (R 4.4.0)
 giscoR         * 0.5.0      2024-05-29 [1] CRAN (R 4.4.0)
 glue             1.6.2      2022-02-24 [1] CRAN (R 4.2.1)
 gtable           0.3.5      2024-04-22 [1] CRAN (R 4.4.0)
 highr            0.11       2024-05-26 [1] CRAN (R 4.4.0)
 htmltools        0.5.4      2022-12-07 [1] CRAN (R 4.2.2)
 httr             1.4.7      2023-08-15 [1] CRAN (R 4.3.1)
 INLA           * 24.05.13-1 2024-05-13 [1] local
 inlabru        * 2.10.1     2023-12-21 [1] CRAN (R 4.3.2)
 jsonlite         1.7.3      2022-01-17 [1] CRAN (R 4.1.2)
 KernSmooth       2.23-22    2023-07-10 [1] CRAN (R 4.3.1)
 knitr            1.45       2023-10-30 [1] CRAN (R 4.3.2)
 labeling         0.4.3      2023-08-29 [1] CRAN (R 4.3.1)
 lattice          0.20-45    2021-09-22 [1] CRAN (R 4.2.1)
 lifecycle        1.0.4      2023-11-07 [1] CRAN (R 4.3.2)
 magrittr         2.0.3      2022-03-30 [1] CRAN (R 4.4.0)
 mapproj        * 1.2.8      2022-01-12 [1] CRAN (R 4.1.2)
 maps           * 3.4.0      2021-09-25 [1] CRAN (R 4.1.2)
 Matrix         * 1.7-0      2024-04-26 [1] CRAN (R 4.4.0)
 MatrixModels     0.5-3      2023-11-06 [1] CRAN (R 4.4.0)
 mnormt           2.1.0      2022-06-07 [1] CRAN (R 4.2.1)
 munsell          0.5.1      2024-04-01 [1] CRAN (R 4.3.3)
 numDeriv         2016.8-1.1 2019-06-06 [1] CRAN (R 4.3.2)
 patchwork      * 1.2.0      2024-01-08 [1] CRAN (R 4.3.2)
 pillar           1.9.0      2023-03-22 [1] CRAN (R 4.3.1)
 pkgconfig        2.0.3      2019-09-22 [1] CRAN (R 4.3.2)
 plyr             1.8.8      2022-11-11 [1] CRAN (R 4.2.2)
 png              0.1-8      2022-11-29 [1] CRAN (R 4.2.2)
 PointedSDMs    * 2.0.0      2024-06-19 [1] Github (PhilipMostert/PointedSDMs@f3749fb)
 proxy            0.4-27     2022-06-09 [1] CRAN (R 4.2.1)
 R.devices        2.17.2     2024-01-29 [1] CRAN (R 4.3.2)
```

```
 R.methodsS3     1.8.2     2022-06-13 [1] CRAN (R 4.3.2)
 R.oo            1.26.0    2024-01-24 [1] CRAN (R 4.3.2)
 R.utils         2.12.3    2023-11-18 [1] CRAN (R 4.3.2)
 R6            * 2.5.1     2021-08-19 [1] CRAN (R 4.3.2)
 ragg            1.2.7     2023-12-11 [1] CRAN (R 4.3.2)
 rappdirs        0.3.3     2021-01-31 [1] CRAN (R 4.3.2)
 raster        * 3.5-11    2021-12-23 [1] CRAN (R 4.1.2)
 RColorBrewer    1.1-3     2022-04-03 [1] CRAN (R 4.3.2)
 Rcpp            1.0.8     2022-01-13 [1] CRAN (R 4.1.2)
 rlang           1.1.1     2023-04-28 [1] CRAN (R 4.3.1)
 rmarkdown       2.27      2024-05-17 [1] CRAN (R 4.4.0)
 s2              1.1.6     2023-12-19 [1] CRAN (R 4.3.2)
 scales          1.3.0     2023-11-28 [1] CRAN (R 4.3.2)
 sessioninfo     1.2.2     2021-12-06 [1] CRAN (R 4.3.2)
 sf            * 1.0-9     2022-11-08 [1] CRAN (R 4.2.2)
 showtext      * 0.9-7     2024-03-02 [1] CRAN (R 4.4.0)
 showtextdb    * 3.0       2020-06-04 [1] CRAN (R 4.3.1)
 sn              2.1.1     2023-04-04 [1] CRAN (R 4.3.1)
 sp            * 2.0-0     2023-06-22 [1] CRAN (R 4.3.1)
 sysfonts      * 0.8.9     2024-03-02 [1] CRAN (R 4.4.0)
 systemfonts     1.1.0     2024-05-15 [1] CRAN (R 4.4.0)
 terra           1.6-47    2022-12-02 [1] CRAN (R 4.2.2)
 textshaping     0.4.0     2024-05-24 [1] CRAN (R 4.4.0)
 tibble          3.2.1     2023-03-20 [1] CRAN (R 4.3.1)
 tidyselect      1.2.1     2024-03-11 [1] CRAN (R 4.3.3)
 units           0.8-0     2022-02-05 [1] CRAN (R 4.1.2)
 utf8            1.2.2     2021-07-24 [1] CRAN (R 4.2.1)
 vctrs           0.6.4     2023-10-12 [1] CRAN (R 4.3.1)
 withr           3.0.0     2024-01-16 [1] CRAN (R 4.3.2)
 wk              0.6.0     2022-01-03 [1] CRAN (R 4.1.2)
 xfun            0.40      2023-08-09 [1] CRAN (R 4.3.1)
 xml2            1.3.3     2021-11-30 [1] CRAN (R 4.1.2)
 yaml            2.3.7     2023-01-23 [1] CRAN (R 4.2.2)

 [1] /home/ahomec/p/philism/R/x86_64-pc-linux-gnu-library/4.1
 [2] /usr/local/lib/R/site-library
 [3] /usr/lib/R/site-library
 [4] /usr/lib/R/library

------------------------------------------------------------------------------------------------------------
```