Appendix S3: Inferring move persistence from southern elephant seal Argos tracking data

Jonsen et al

2022-11-30

We draw on IMOS southern elephant seal tracking data from Iles Kerguelen to infer move persistence, γ_t, along four tracks. We fitted the `mp` SSM with a 12-h prediction interval with `fit_ssm()`, using the Argos Kalman filter-derived error ellipse information provided with each observation (Jonsen et al. 2020). The `mp` SSM simultaneously estimates locations and γ_t, and their uncertainties. We then assess how γ_t changes along the seals' tracks to infer regions where the seals spend disproportionately more or less time during their foraging trips.

This example application also illustrates how the SSM can be fit using parallel processing, leveraging the `future` and `furrr` R packages, to reduce computation time. This approach is not typically necessary for a data set consisting of only 4 individual tracks, but is a valuable time-saver for larger data sets.

The three southern elephant seals on foraging trips to the Antarctic shelf region all engaged in spatially constrained bouts of low move persistence while in the shelf region (Fig. 1 a,c,d; blue circles in e). Without additional data it is unclear exactly what these bouts of low horizontal move persistence represent. They could result from area-restricted search and foraging within dense prey aggregations, physical constraints of dense ice on horizontal movements, haulout (resting) on sea-ice, or some combination of these. Conversely, the seal on a pelagic foraging trip engaged in slower, more meandering movements with less spatially constrained bouts of lower move persistence (Fig. 1 b, e). This general movement pattern may be consistent with searching for suitable foraging resources within the highly variable eddy fields between the Subantarctic and Polar Fronts (Jonsen et al. 2019).
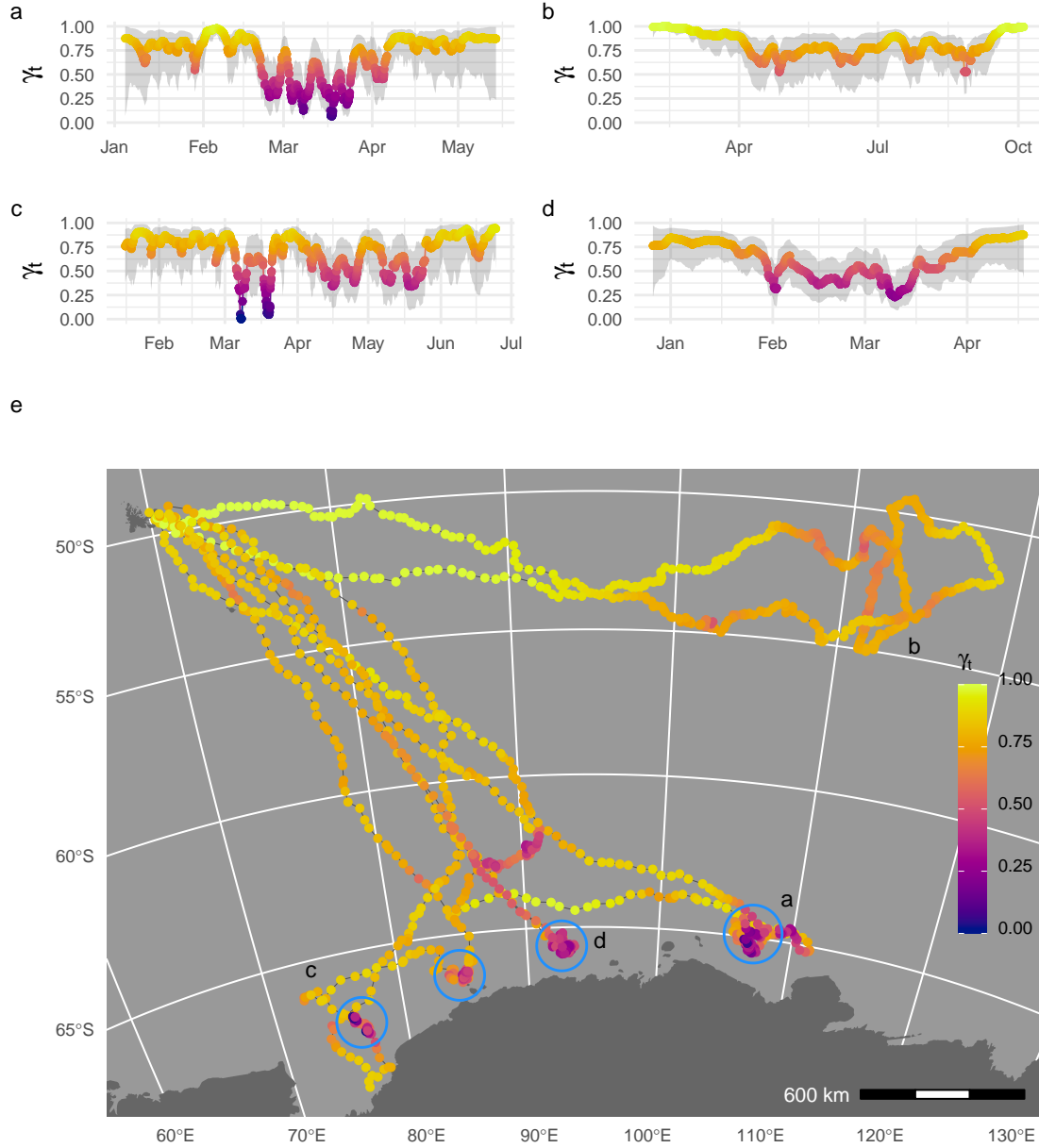
Figure 1: Inferred move persistence, γ_t, time-series for four southern elephant seals (a-d; grey envelopes are 95 % CI's, note differing date ranges on the x axes), and along their 2-D tracks (e; track labels, a-d, correspond to the time-series plots). Locations associated with low γ_t (purple) are indicative of slow, undirected movements, whereas high γ_t (yellow) is indicative of faster, directed movements. Blue circles highlight bouts of spatially constrained low γ_t.

References

Jonsen I, McMahon C, Patterson T, Auger-Methe M, Harcourt R, Hindell M, Bestley S. (2019). Movement behaviour responses to environment: fast inference of individual variation with a mixed effects model. *Ecology*, 100:e02566.

Jonsen ID, Patterson TA, Costa DP, Doherty PD, Godley BJ, Grecian WJ, Guinet C, Hoenner X, Kienle SS, Robinson PW, Votier SC. (2020). A continuous-time state-space model for rapid quality control of Argos locations from animal-borne tags. *Movement Ecology*, 8:1-13.

Code for fitting SSM & rendering figure

Load required R packages

```
require(tidyverse)
require(ggspatial)
require(ggforce)
require(patchwork)
require(sf)
require(foieGras)
```

Fit `mp` SSM with 12-h `time.step` to infer movement persistence along 4 southern elephant seal tracks. Use `future` and `furrr` R packages to fit SSM to individual seal tracks using parallel processing for computation speed-up.

```
## Load data from .csv file
sese <- read.csv("../data/sese_extra.csv")

## invoke parallel multisession
future::plan("multisession")

## split individual fit_ssm jobs across processors
## fit `mp` SSM, using speed filter of 4 m/s to exclude extreme observations
fit <- sese %>%
  split(.$id) %>%
  furrr::future_map(~ try(fit_ssm(
      x = .x,
      vmax = 4,
      model = "mp",
      time.step = 12,
      control = ssm_control(verbose = 0)
    ), silent = TRUE),
    .progress = FALSE,
    .options = furrr::furrr_options(seed = TRUE)
    ) %>%
    bind_rows(.)

## close multisession
future::plan("sequential")
```

Plot move persistence estimates as time-series, 1 panel per seal

```r
## Normalise the gamma_t estimates so they span the interval 0, 1, grouping all
##  seals together so estimates are comparable among seals.
p1 <- plot(fit,
           type = 3,
           pages = 1,
           ncol = 2,
           normalise = TRUE,
           group = TRUE,
           pal = "Plasma") &
  ylim(0, 1) &
  theme(legend.position = "none",
        axis.text = element_text(size = 7))
```

Map move persistence estimates along the SSM-estimated seal tracks

```r
## customise mapping aesthetics
my.aes <- aes_lst(conf = FALSE,
                  line = TRUE)
my.aes$df$size[1] <- 1.5
my.aes$df$size[3] <- 0.1
my.aes$df$col[3] <- grey(0.3)
my.aes$df$fill[5] <- grey(0.4)
my.aes$df$fill[6] <- grey(0.6)

## project map, add scale bar using ggspatial::annotation_scale
m1 <- foieGras::map(fit,
           what = "p",
           aes = my.aes,
           crs = "+proj=stere +lon_0=95 +units=km +datum=WGS84 +no_defs",
           normalise = TRUE,
           group = TRUE,
           silent = TRUE) +
  xlab(element_blank()) +
  ylab(element_blank()) +
  ggspatial::annotation_scale(height = unit(1.25, "mm"),
                              aes(width_hint = 0.2,
                                  location = "br",
                                  text_col = "white")) +
  theme(legend.position = c(0.95,0.5),
        legend.direction = "vertical",
        legend.key.width = unit(4, "mm"),
        legend.key.height = unit(7, "mm"),
        legend.title = element_text(size = 9),
        legend.text = element_text(size = 7),
        axis.text = element_text(size = 7),
        panel.grid = element_line(colour = "white"))

## define bounding box based on SSM-predicted locations - for map annotations
bb <- grab(fit, what = "p", as_sf = TRUE) %>%
  sf::st_transform(., crs = m1$coordinates$crs) %>%
  sf::st_bbox()
```

```r
## create track labels & highlight circles for map annotations
label.df <- data.frame(tag = c("a","b","c","d"),
                       x = c(0.75, 0.9, 0.19, 0.53) *
                        (bb["xmax"] - bb["xmin"]) + bb["xmin"],
                       y = c(0.32, 0.75, 0.2, 0.25) *
                        (bb["ymax"] - bb["ymin"]) + bb["ymin"])
circle.df <- data.frame(x = c(0.25, 0.365, 0.485, 0.71) *
                         (bb["xmax"] - bb["xmin"]) + bb["xmin"],
                       y = c(0.11, 0.19, 0.24, 0.26) *
                        (bb["ymax"] - bb["ymin"]) + bb["ymin"],
                       r = c(125,125,125,145)
                       )


## add track labels & highlight circles to map
m1 <- m1 +
  geom_text(data = label.df,
            aes(x, y, label=tag),
            size = 3) +
  ggforce::geom_circle(aes(x0 = circle.df$x, y0 = circle.df$y, r = circle.df$r),
             fill = NA,
             size = 0.8,
             colour = "dodgerblue")

## arrange panels & render figure using the `patchwork` package
(p1 | m1) +
  patchwork::plot_layout(nrow = 2,
              heights = c(2, 5),
              guides = "keep") +
  patchwork::plot_annotation(tag_levels = "a") &
  theme(plot.title = element_blank(),
        plot.tag = element_text(size = 9, colour = "black"))
```