

Temporal space use variability analysis for kākā at Orokonui Ecosanctuary

Scott Forrest

2022-11-17

Import relevant packages and set working directory.

```
library(tidyverse)

packages <- c("move", "lattice", "purrr", "here", "raster", "sf", "ggpubr", "ggspatial",
            "patchwork", "jtools", "MuMin", "statmod", "furrr", "lubridate", "scales",
            "heplots", "MASS", "broom", "ggridge", "ggridges", "gifski", "transformr")
walk(packages, require, character.only = T)

here::i_am(paste("./OneDrive - Queensland University of Technology/MSc - Scott Forrest/",
                  "DATA/Modelling/dBBMM/dBBMM markdown for paper 6thJuly2022.Rmd",
                  sep = ""))
```

Import data.

```
GPSfiles <- list.files(here(paste("OneDrive - Queensland University of Technology/",
                                 "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
                                 "CSV input data - dd_speed_6", sep = "")))

all_tags_list <- vector(mode = "list", length = length(GPSfiles))

for(i in 1:length(GPSfiles)){
  all_tags_list[[i]] <- read.csv(here(paste("OneDrive - Queensland University ",
                                         "of Technology/MSc - Scott Forrest/",
                                         "DATA/Modelling/dBBMM/",
                                         "CSV input data - dd_speed_6/",
                                         GPSfiles[[i]],
                                         sep = "")))
}
```

Create functions for the sweeping window.

```
window_function <- function(x, width, move_obj) {
  move_obj[x:(x+(width-1))]

subset_function <- function(df, width, inc) {
  c(seq(1, length(df) - width, inc)) %>% map(window_function, width, df)
}
```

Create an empty raster for the extent, which should be wider than the locations to provide the Brownian bridges with plenty of space. Can be trimmed off later. Create a simple function with preset parameters in the dBBMM model. Location error is the median error from Forrest, Recio, and Seddon (2022) who tested these particular devices. Reference paper for dBBMM method is Kranstauber et al. (2012).

```
ext_dbbmm <- as(extent(1390000, 1435000, 4910000, 4950000), 'SpatialPolygons')
crs(ext_dbbmm) <- "EPSG:2193"
ext_raster <- raster::raster(ext_dbbmm, res = 50)

dBMM_function <- function(x) {
  brownian.bridge.dyn(x, raster = ext_raster, location.error = 8.35, margin = 3,
                       window.size = 15, time.step = 12)
}
```

Run the above functions to create UDs for snapshots of the movement trajectory. Beware that this will take some time to run (~ 20-30 mins on my machine), and will create a large object (~ 8 GB for 240 locations per window at increments of 6 locations). Reducing the increment size will increase the time and memory requirements. This may also be possible with the `amt::rolling_od` (rolling occurrence) function in the ‘amt’ package (Signer, Fieberg, and Avgar 2019), which is a wrapper around `ctmm::occurrence` (Fleming et al. 2016). It should also be noted that this framework is not restricted to the dBBMM utilisation distribution estimation method. The for loop below simply splits up the trajectory into overlapping temporal snapshots, and the space use estimation function in `future_map(., dBMM_function)` can be substituted with any other method.

```
options(future.rng.onMisuse = "ignore")

plan(strategy = multisession) # set processing to parallel

all_tags_subsets <- vector(mode = "list", length = length(all_tags_list))

for (i in 1:length(all_tags_list)) {

  all_tags_list[[i]]$DateTime <- as.POSIXct(all_tags_list[[i]]$DateTime, tz = "GMT")

  all_tags_subsets[[i]] <- move(x = all_tags_list[[i]]$lon,
                                y = all_tags_list[[i]]$lat,
                                time = all_tags_list[[i]]$DateTime,
                                proj = CRS("+proj=longlat +ellps=WGS84"),
                                data = all_tags_list[[i]],
                                animal = all_tags_list[[i]]$id) %>%
    spTransform(., CRS(SRS_string = "EPSG:2193")) %>%
    subset_function(width = 240, inc = 6) %>%
    future_map(., dBMM_function)

}

plan(strategy = sequential) # set processing back to sequential (standard)
```

Calculate Bhattacharyya’s Affinity between UDs

Calculating Bhattacharyya’s Affinity between every UD for each individual. This is to create a similarity matrix, similar to the approach of Kranstauber et al. (2020), which allows for the identification of space use patterns.

This for loop calculates the BA for each pair, and creates the correlation plots (similarity matrix) for each individual.

```

corr.plots <- vector(mode = "list", length = length(all_tags_list))

for(x in 1:length(all_tags_list)) {

  dbbmms_240_8 <- all_tags_subsets[[x]]

  bbox_trim <- st_bbox(c(xmin = 1405000,
                         xmax = 1420000,
                         ymin = 4923000,
                         ymax = 4936000),
                         crs = 2193)
  dbbmms_240_8_trimmed <- map(dbbmms_240_8, crop, bbox_trim)

  BA <- c()

  for(i in 1:length(dbbmms_240_8_trimmed)) {
    for(j in 1:length(dbbmms_240_8_trimmed)) {

      BA <- c(BA, sum(sqrt(getValues(dbbmms_240_8_trimmed[[j]])) *
                    sqrt(getValues(dbbmms_240_8_trimmed[[i]]))))}

    }
  }

  index1 <- rep(1:length(dbbmms_240_8_trimmed),
                each = length(dbbmms_240_8_trimmed))

  index2 <- rep(1:length(dbbmms_240_8_trimmed),
                each = length(dbbmms_240_8_trimmed))

  mean.time <- .POSIXct(rep(NA, length(dbbmms_240_8)))
  min.time <- .POSIXct(rep(NA, length(dbbmms_240_8)))
  max.time <- .POSIXct(rep(NA, length(dbbmms_240_8)))

  for (i in 1:length(dbbmms_240_8)) {

    mean.time[[i]] <- mean(dbbmms_240_8[[i]]@DBMvar@timestamps)
    min.time[[i]] <- min(dbbmms_240_8[[i]]@DBMvar@timestamps)
    max.time[[i]] <- max(dbbmms_240_8[[i]]@DBMvar@timestamps)

  }

  mean.time1 <- rep(mean.time, each = length(dbbmms_240_8_trimmed))
  mean.time2 <- rep(mean.time, length(dbbmms_240_8_trimmed))

  min.time1 <- rep(min.time, each = length(dbbmms_240_8_trimmed))
  min.time2 <- rep(min.time, length(dbbmms_240_8_trimmed))

  max.time1 <- rep(max.time, each = length(dbbmms_240_8_trimmed))
  max.time2 <- rep(max.time, length(dbbmms_240_8_trimmed))
}

```

```

BA_df <- data.frame(index1, min.time1, mean.time1, max.time1,
                     index2, min.time2, mean.time2, max.time2, BA)

corr.plots[[x]] <- BA_df %>% ggplot(aes(x = mean.time1, y = mean.time2,
                                           colour = BA)) +
  geom_point(size = 10, shape = 15) +
  scale_x_datetime("2020/2021", labels = date_format("%b")) +
  scale_y_datetime("2020/2021", labels = date_format("%b")) +
  scale_colour_viridis_c(option = "B", limits = c(0.38,1), direction = -1) +
  coord_fixed() +
  theme_classic() +
  theme(legend.title.align = 0.5,
        legend.spacing.y = unit(0.5, "cm"),
        axis.title.y = element_text(margin = margin(r = 10)),
        axis.title.x = element_text(margin = margin(t = 10)))

}

```

Check the minimum BA for each individual.

```

for(i in 1:length(all_tags_list)) {
  print(min(corr.plots[[i]]$data$BA))
}

```

```

## [1] 0.4269132
## [1] 0.8527048
## [1] 0.8127294
## [1] 0.3818188
## [1] 0.8283545
## [1] 0.8140483
## [1] 0.7916728
## [1] 0.756685
## [1] 0.8969809
## [1] 0.9123947

```

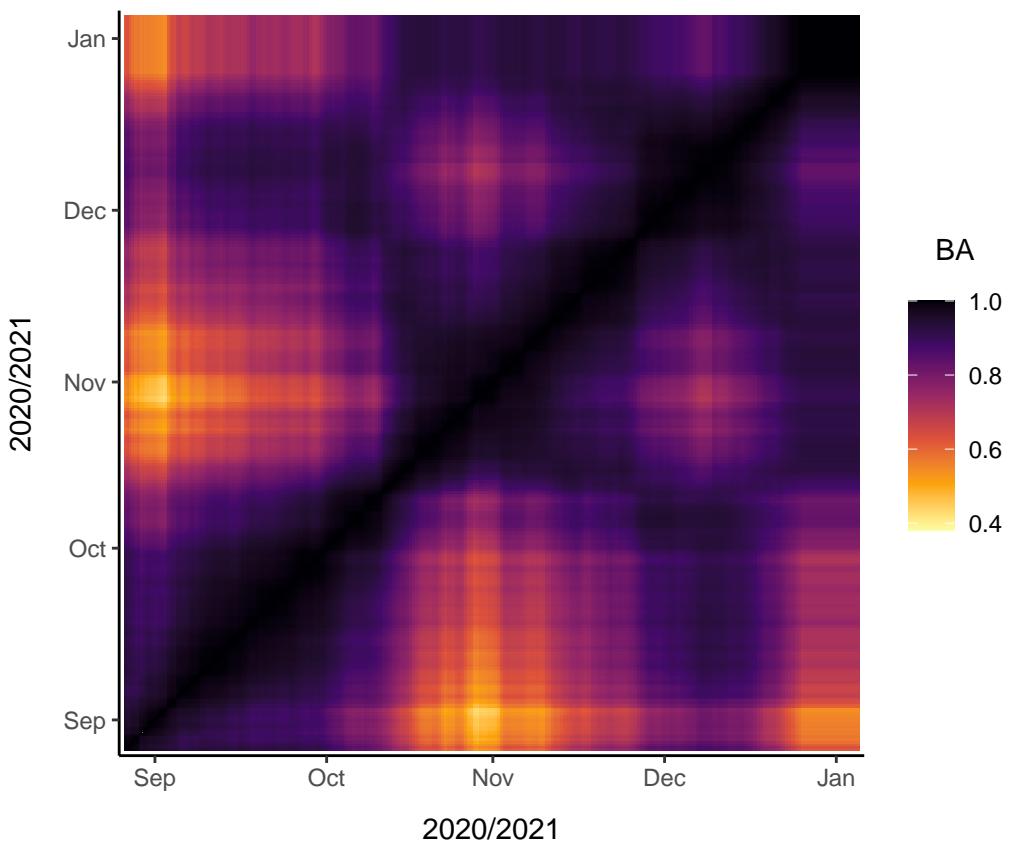
Plot the Bhattacharyya's Affinity (BA) similarity matrices for each individual.

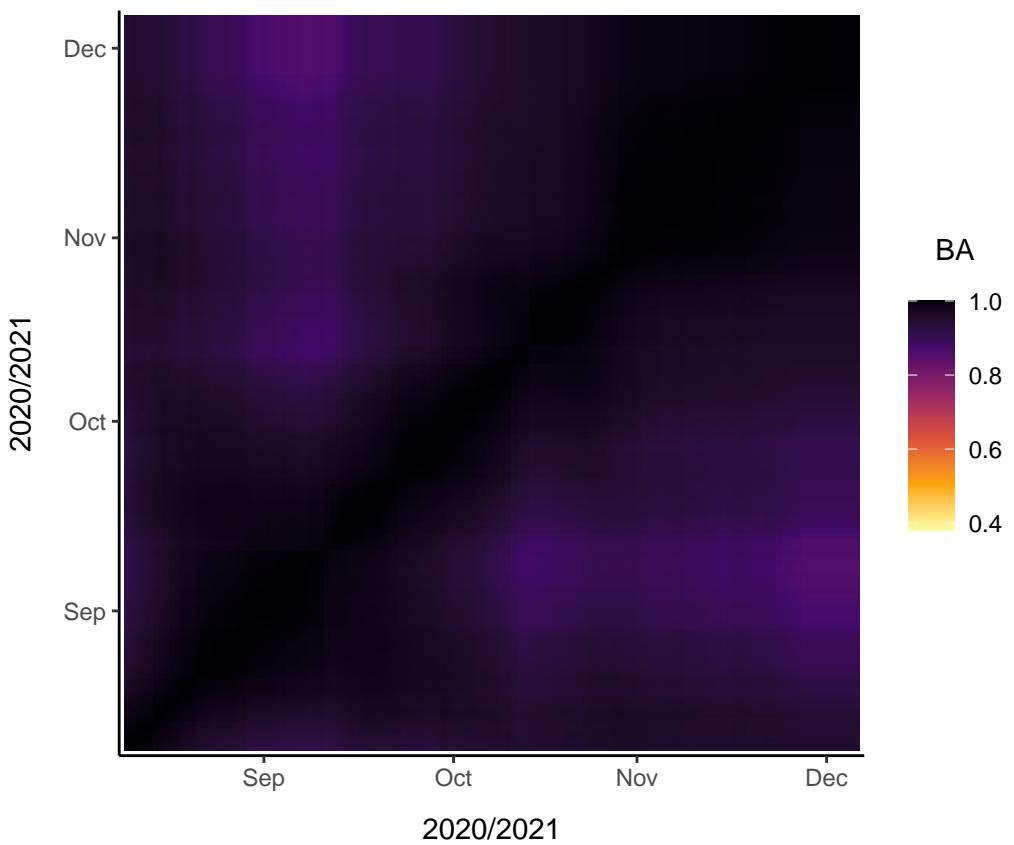
Caption from manuscript: Similarity matrices of space use calculated with Bhattacharyya's Affinity (BA) between all UDs for each individual throughout the study period. Darker values indicate higher similarity and lighter values indicate lower similarity. The brighter colours of panels A and D indicate the dissimilarity between the UDs of the juvenile individuals, which was largely due to positional changes of the UD, rather than expansion and contraction .

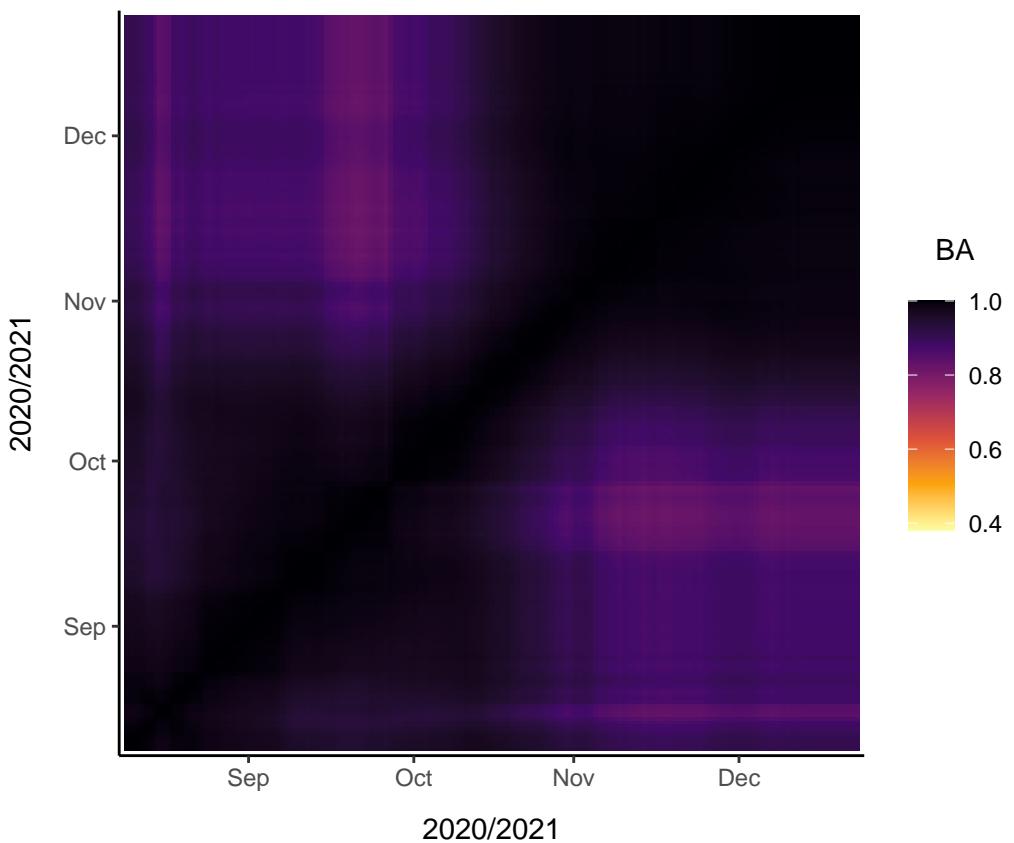
```

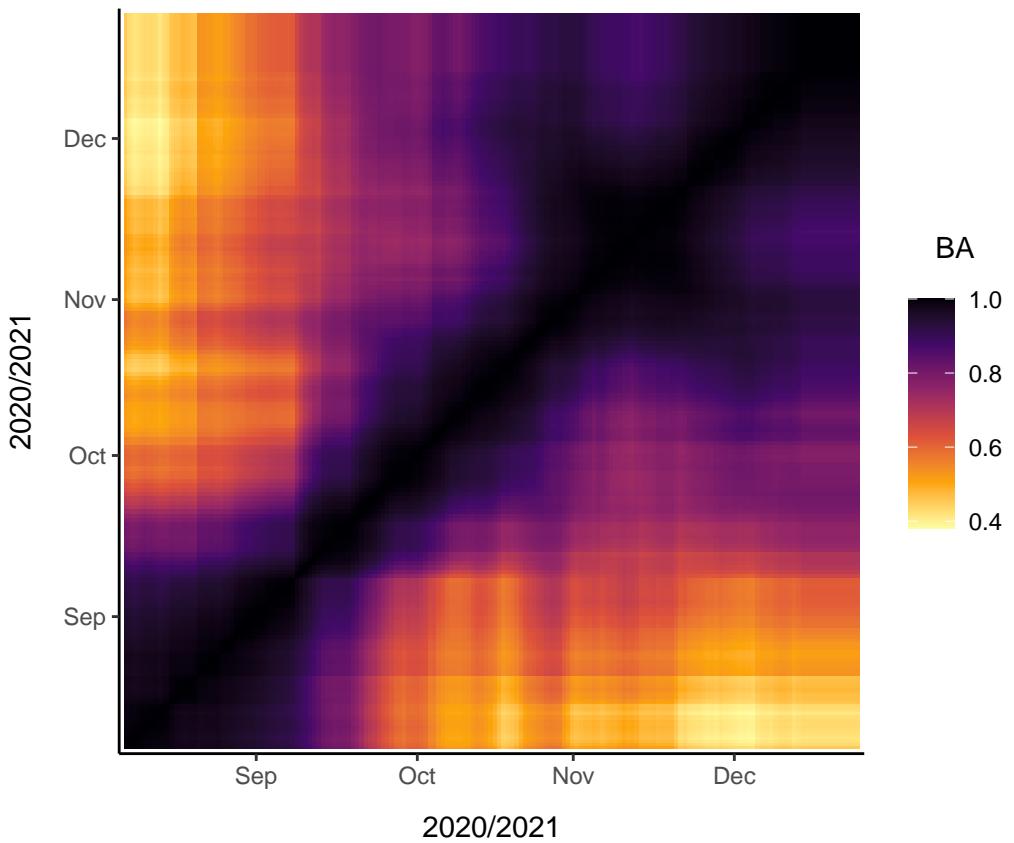
for(i in 1:length(all_tags_list)) {
  print(corr.plots[[i]])
}

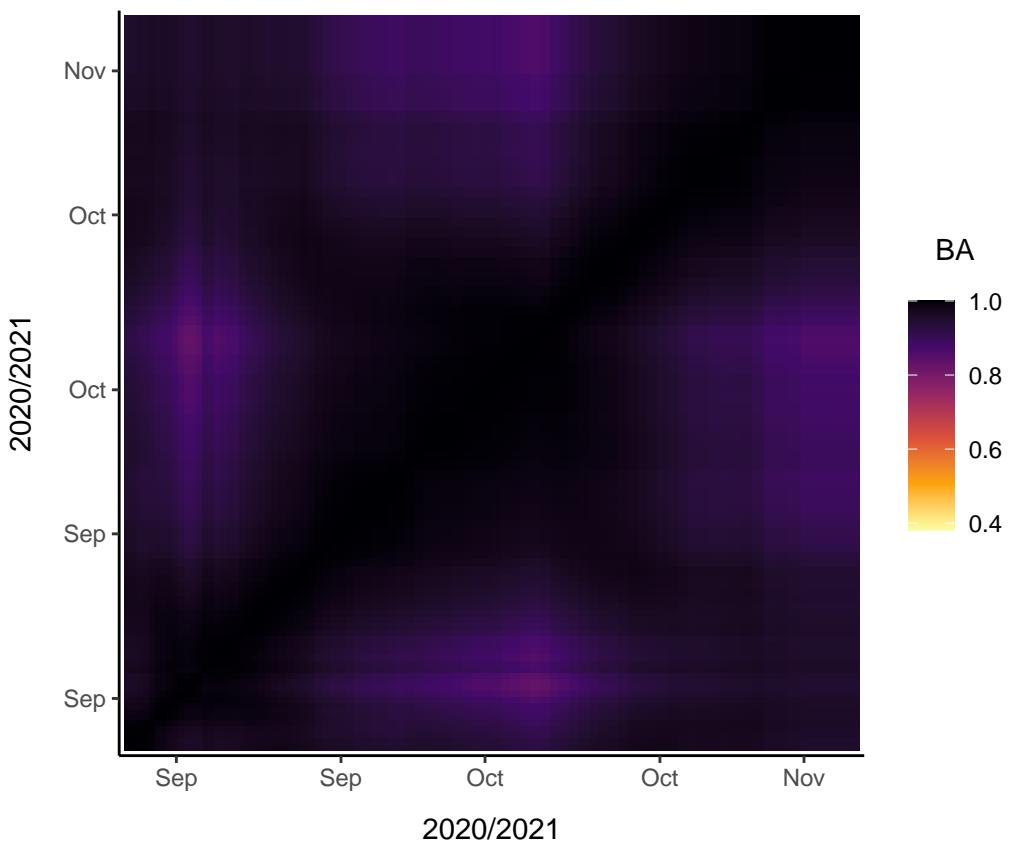
```

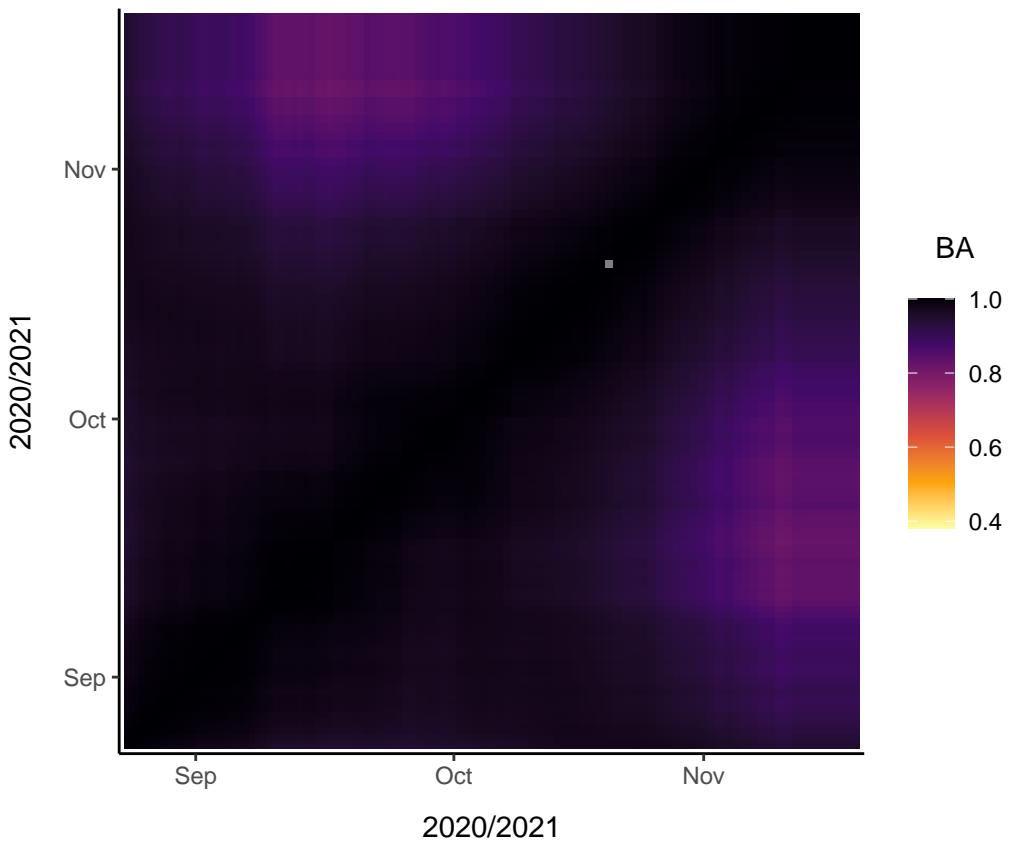


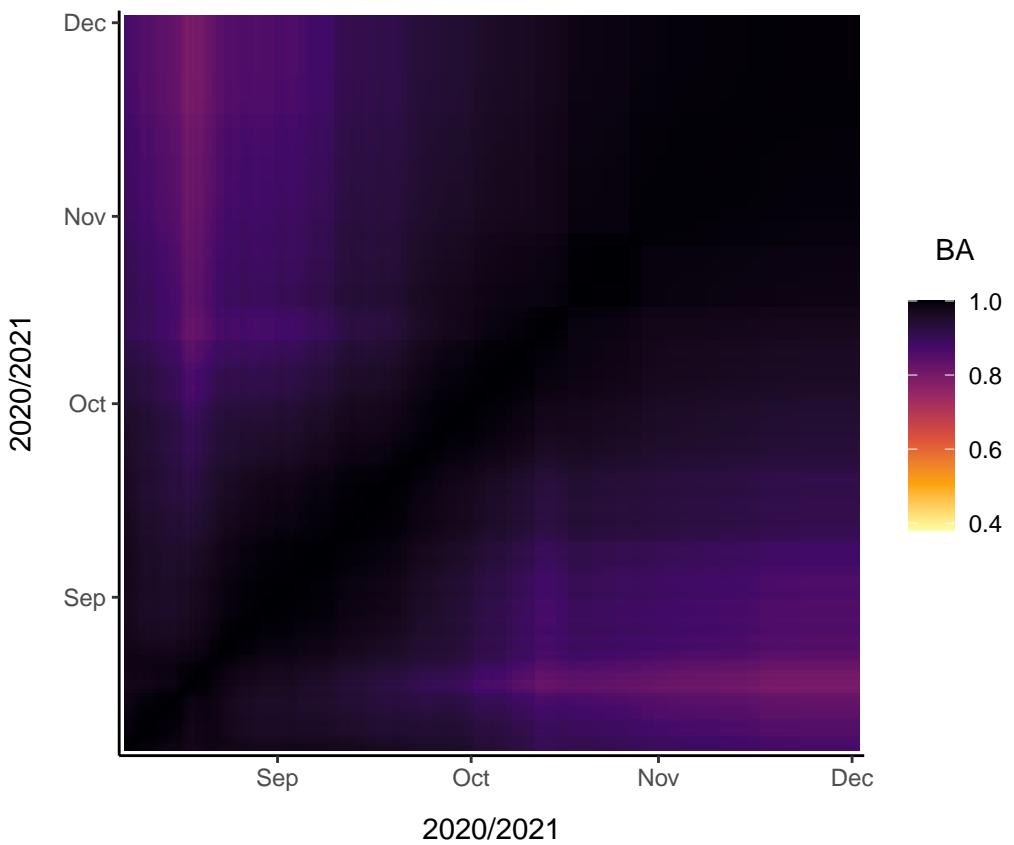


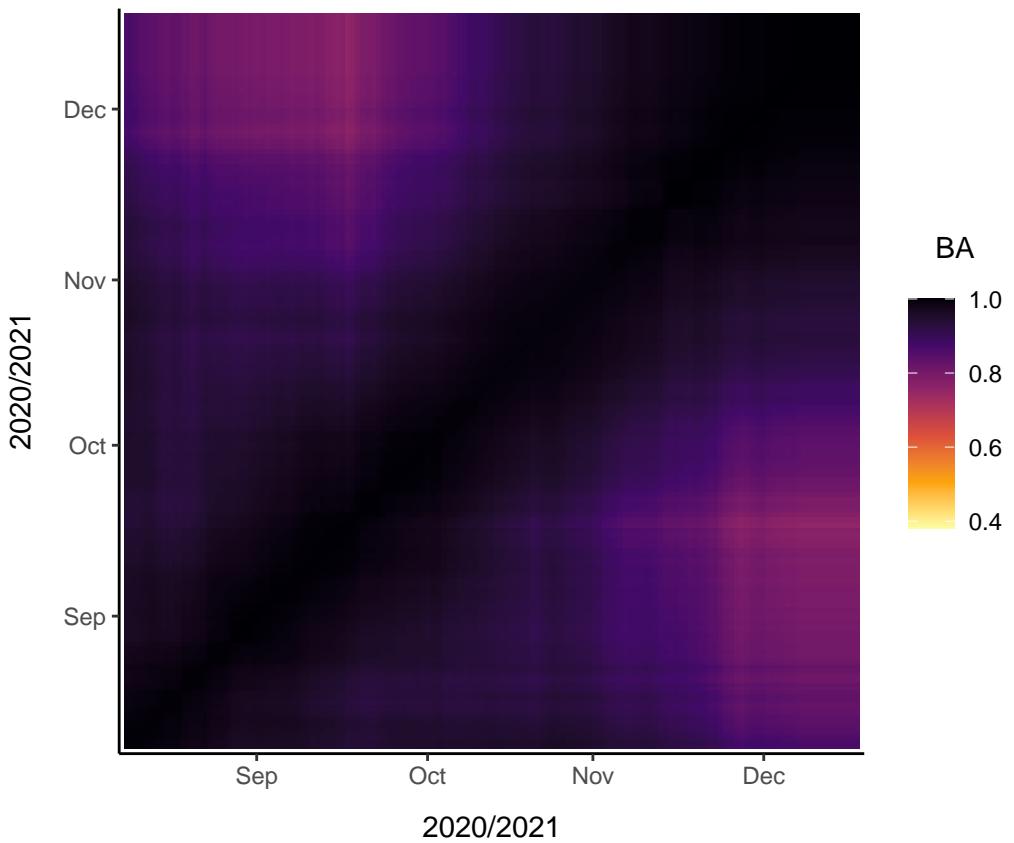


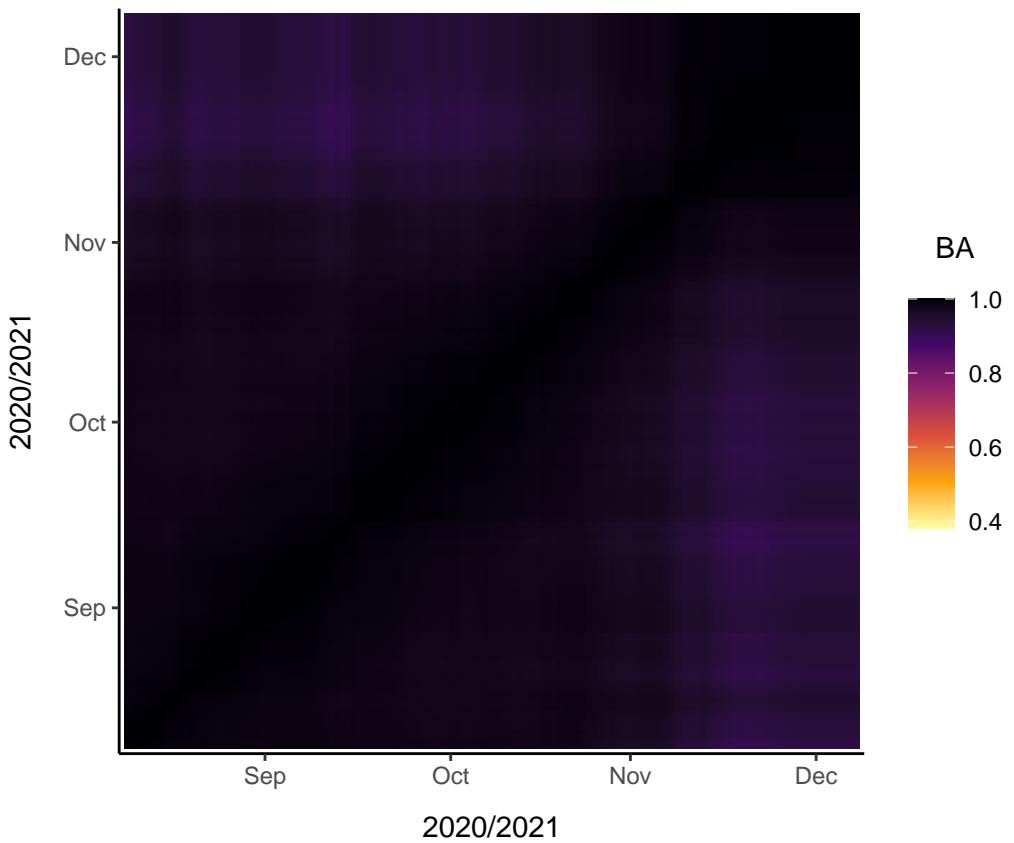


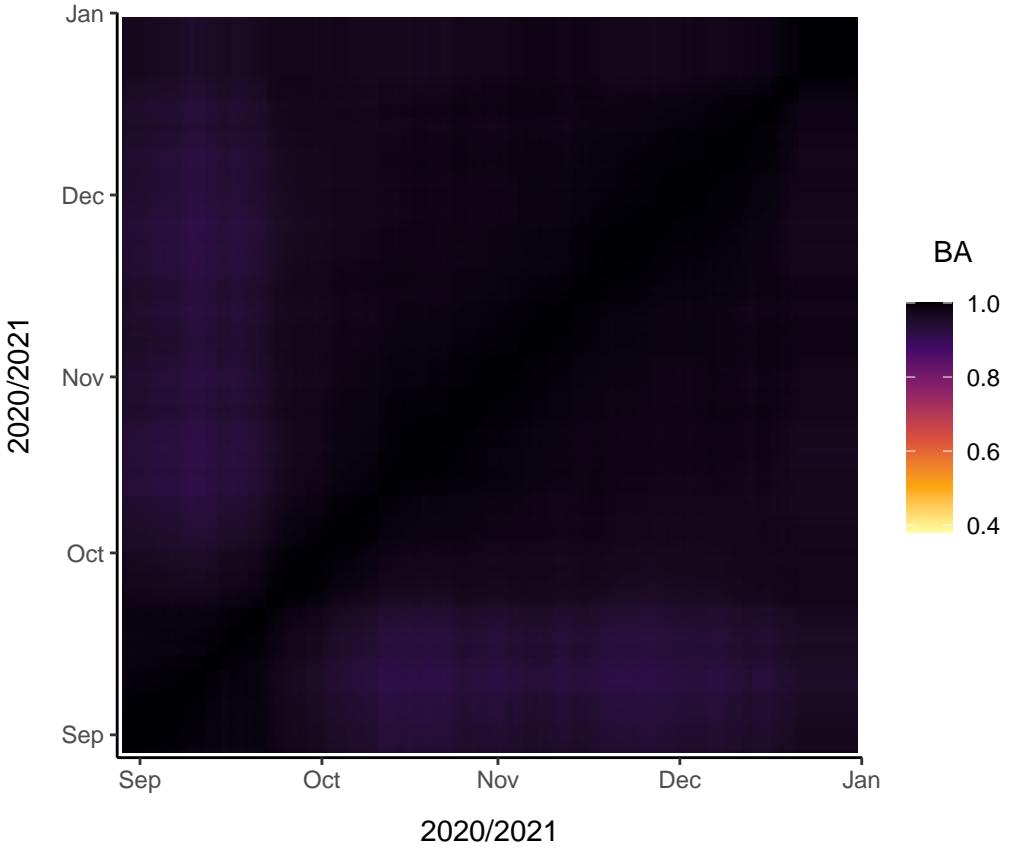












```
#to save plots individually
# for(i in 1:10) {
# ggsave(corr.plots[[i]],
#        filename = paste("correlation.plots240_8_0.38-1.",i,".png", sep = ""),
#        device = "png", path = here(paste("OneDrive - Queensland University of",
#                                         "Technology/MSc - Scott Forrest/DATA/",
#                                         "Modelling/dBBMM/Graphical outputs",
#                                         sep = "")),
#        width = 180, height = 150, dpi = 300, units = "mm")
# }
```

Add the correlation plots to a single plot for the manuscript.

```
# for all BA similarity matrices

# patchwork <- corr.plots[[1]] + corr.plots[[2]] + corr.plots[[3]] +
#   corr.plots[[4]] + corr.plots[[5]] + corr.plots[[6]] + corr.plots[[7]] +
#   corr.plots[[8]] + corr.plots[[9]] + corr.plots[[10]] +
#   plot_layout(guides = 'collect', nrow = 2)

# omitting the final BA similarity matrix for clarity

patchwork <- corr.plots[[1]] + corr.plots[[2]] + corr.plots[[3]] +
  corr.plots[[4]] + corr.plots[[5]] + corr.plots[[6]] + corr.plots[[7]] +
  corr.plots[[8]] + corr.plots[[9]] +
  plot_layout(guides = 'collect', nrow = 3)
```

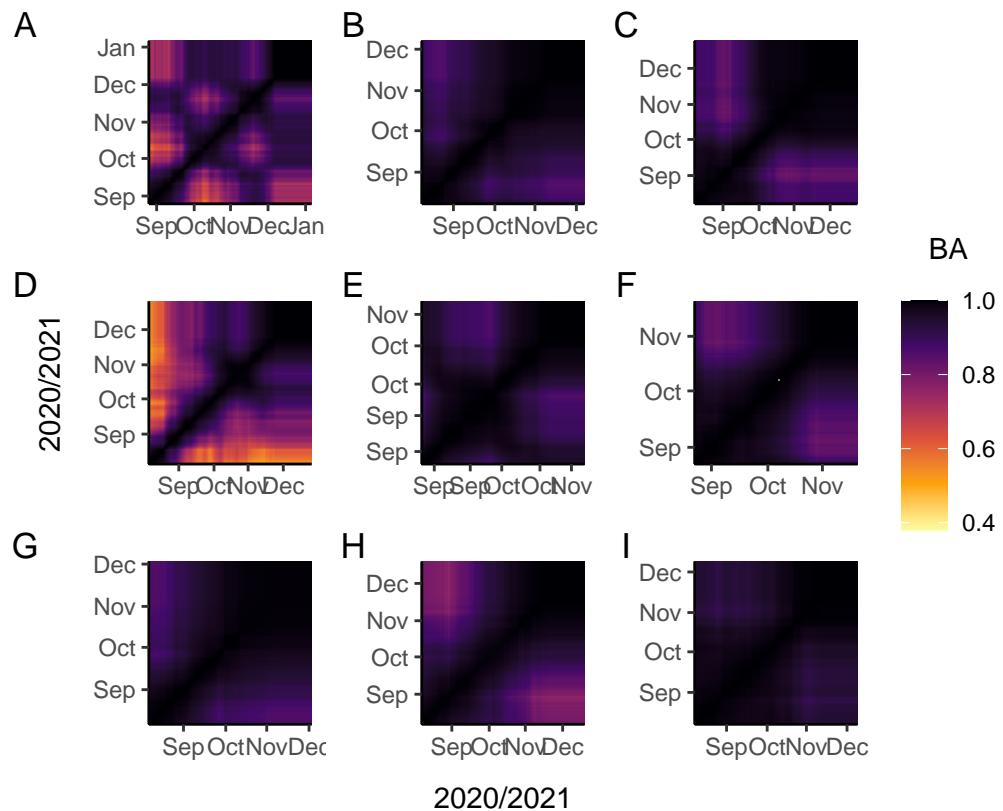
```

for (i in c(1:3, 5:9)) {
patchwork[[i]] <- patchwork[[i]] + theme(axis.title.y = element_blank())
}

for(i in c(1:6,7,9)){
  patchwork[[i]] <- patchwork[[i]] + theme(axis.title.x = element_blank())
}

patchwork + plot_annotation(tag_levels = 'A')

```



```

# ggsave("all_tags_240_6_square_BA_similarity_matrix_20221110.tiff",
#        path = here(paste("OneDrive - Queensland University of Technology/",
#                           "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
#                           "Graphical outputs", sep = "")),
#        width = 200, height = 220, dpi = 300, units = "mm")

```

Utilisation distribution area

Calculate the area of the UD₅₀ and UD₉₅ isopleths for each individual.

```

all_areas_50 <- vector(mode = "list", length = length(all_tags_list))
all_areas_95 <- vector(mode = "list", length = length(all_tags_list))
all_mean_times <- vector(mode = "list", length = length(all_tags_list))

```

```

plan(multisession)

for(j in 1:length(all_tags_list)) {

all_areas_50[[j]] <- all_tags_subsets[[j]] %>%
  future_map(., raster2contour, levels = 0.5) %>%
  future_map(., st_as_sf) %>%
  future_map(., st_polygonize) %>%
  future_map(., st_area) %>%
  unlist()

all_areas_95[[j]] <- all_tags_subsets[[j]] %>%
  future_map(., raster2contour, levels = 0.95) %>%
  future_map(., st_as_sf) %>%
  future_map(., st_polygonize) %>%
  future_map(., st_area) %>%
  unlist()

for (a in 1:length(all_tags_subsets[[j]])) {
  all_mean_times[[j]][[a]] <- mean(all_tags_subsets[[j]][[a]]@DBMvar@timestamps)
}

plan(sequential)

```

Calculating UD centroids

Calculate the centroid location of the UD₉₅ areas. If the UD is multimodal and has separate areas, this code will average the location between them, but there is also an option ('of_largest_polygon = TRUE' in the st_centroid function) to use the centroid location of only the largest area.

```

plan(multisession)

all_centroids <- vector(mode = "list", length = length(all_tags_list))

for(i in 1:length(all_centroids)) {
  all_centroids[[i]] <- all_tags_subsets[[i]] %>%
    future_map(., raster2contour, levels = 0.95) %>%
    future_map(., st_as_sf) %>%
    future_map(., st_polygonize) %>%
    future_map(., st_centroid)
}

all_tags_centerX <- vector(mode = "list", length = length(all_tags_list))
all_tags_centerY <- vector(mode = "list", length = length(all_tags_list))

for(i in 1:length(all_centroids)) {
  for(j in 1:length(all_centroids[[i]])){
    all_tags_centerX[[i]][[j]] <- st_coordinates(all_centroids[[i]][[j]])[1]
  }
}

```

```

    all_tags_centY[[i]][[j]] <- st_coordinates(all_centroids[[i]][[j]])[2]
  }
}

plan(sequential)

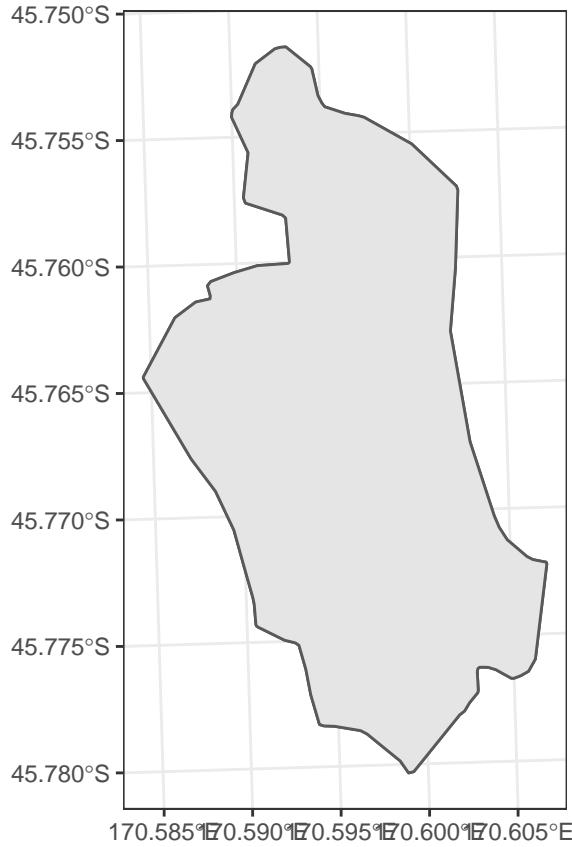
```

Import Orokonui Ecosanctuary fence spatial object to calculate overlap for each UD to assess risk exposure throughout the tracking period.

```
OrokonuiFence <- st_read(here(paste("OneDrive - Queensland University of Technology/",
                                     "MSc - Scott Forrest/GIS, Mapping/",
                                     "OrokonuiFence.shp", sep = "")))
```

```
## Reading layer 'OrokonuiFence' from data source
##   'C:\Users\n11207361\OneDrive - Queensland University of Technology\MSc - Scott Forrest\GIS, Mapping'
##   using driver 'ESRI Shapefile'
## Simple feature collection with 2 features and 1 field
## Geometry type: POLYGON
## Dimension:      XY
## Bounding box:  xmin: 1412176 ymin: 4927554 xmax: 1413950 ymax: 4930746
## Projected CRS: NZGD2000 / New Zealand Transverse Mercator 2000
```

```
OrokonuiFence %>% ggplot() +
  geom_sf() +
  theme_bw()
```



Proportion of UD overlap with risk area

In this case it is the inside of Orokonui Ecosanctuary fence (although this could be any area of risk/safety). This is not the proportion of UD₉₅ area, but the the proportion of the probability surface (i.e. weighted by the probability), which represents the time spent in that area.

```
all_tags_inside_overlap <- vector(mode = "list", length = length(all_tags_list))

for (i in 1:length(all_tags_subsets)){
  for (j in 1:length(all_tags_subsets[[i]])){
    all_tags_inside_overlap[[i]][[j]] <- cellStats(mask(all_tags_subsets[[i]][[j]],
                                                       OrokonuiFence),
                                                 sum)
  }
}
```

Generating UD contours to create animations

```
plan(multisession)
```

```

all_uds_95 <- vector(mode = "list", length = length(all_tags_subsets))
all_uds_90 <- vector(mode = "list", length = length(all_tags_subsets))
all_uds_75 <- vector(mode = "list", length = length(all_tags_subsets))
all_uds_50 <- vector(mode = "list", length = length(all_tags_subsets))

for(i in 1:length(all_tags_subsets)) {

  all_uds_95[[i]] <- all_tags_subsets[[i]] %>%
    future_map(., raster2contour, levels = 0.95) %>%
    future_map(., st_as_sf)

  all_uds_90[[i]] <- all_tags_subsets[[i]] %>%
    future_map(., raster2contour, levels = 0.90) %>%
    future_map(., st_as_sf)

  all_uds_75[[i]] <- all_tags_subsets[[i]] %>%
    future_map(., raster2contour, levels = 0.75) %>%
    future_map(., st_as_sf)

  all_uds_50[[i]] <- all_tags_subsets[[i]] %>%
    future_map(., raster2contour, levels = 0.50) %>%
    future_map(., st_as_sf)

}

plan(sequential)

all_uds_95_bound <- do.call(bind_rows, all_uds_95)
all_uds_90_bound <- do.call(bind_rows, all_uds_90)
all_uds_75_bound <- do.call(bind_rows, all_uds_75)
all_uds_50_bound <- do.call(bind_rows, all_uds_50)

```

Creating a data frame that can be used in further analyses.

```

id <- 05:14
sex <- c("M", "M", "M", "F", "F", "F", "M", "M", "F")
age <- c(1, 10, 5, 1, 3, 2, 2, 3, 10, 8)
age_group <- c(1, 2, 2, 1, 1, 1, 1, 2, 2)
breeding <- c("N", "N", "N", "N", "Y", "N", "Y", "N", "N", "N")

all_areas_dfs <- vector(mode = "list", length = length(all_tags_subsets))

# all_tags_df <- NULL

for (k in 1:length(all_tags_list)) {

  all_areas_dfs[[k]] <- as.data.frame(rep(id[[k]], length(all_areas_50[[k]])))
  all_areas_dfs[[k]]$index <- 1:length(all_areas_50[[k]])
  all_areas_dfs[[k]]$sex <- rep(sex[[k]], length(all_areas_50[[k]]))
  all_areas_dfs[[k]]$age <- rep(age[[k]], length(all_areas_50[[k]]))
  all_areas_dfs[[k]]$age_group <- rep(age_group[[k]], length(all_areas_50[[k]]))
  all_areas_dfs[[k]]$breeding <- rep(breeding[[k]], length(all_areas_50[[k]]))
}

```

```

all_areas_dfs[[k]]$time <- .POSIXct(unlist(all_mean_times[[k]]))
all_areas_dfs[[k]]$area_50 <- all_areas_50[[k]]
all_areas_dfs[[k]]$area_95 <- all_areas_95[[k]]
all_areas_dfs[[k]]$centx <- unlist(all_tags_centX[[k]])
all_areas_dfs[[k]]$centy <- unlist(all_tags_centY[[k]])
all_areas_dfs[[k]]$inside <- unlist(all_tags_inside_overlap[[k]])
all_areas_dfs[[k]]$outside <- 1 - unlist(all_tags_inside_overlap[[k]])
all_areas_dfs[[k]] <- all_areas_dfs[[k]] %>%
  mutate(mean_50 = mean(area_50),
        mean_95 = mean(area_95),
        diff_50 = (area_50-mean_50)/mean_50,
        diff_95 = (area_95-mean_95)/mean_95)

colnames(all_areas_dfs[[k]]) <- c("tag", "index", "sex", "age", "age_group", "breeding", "time", "area50")
all_areas_dfs[[k]]$tag <- as.factor(all_areas_dfs[[k]]$tag)
all_areas_dfs[[k]]$sex <- as.factor(all_areas_dfs[[k]]$sex)
all_areas_dfs[[k]]$breeding <- as.factor(all_areas_dfs[[k]]$breeding)

}

all_tags_df <- bind_rows(all_areas_dfs)
all_tags_df <- all_tags_df %>% mutate(diff_95_perc = diff_95 * 100,
                                         area_95_km2 = area_95/1000000,
                                         ud95 = all_uds_95_bound$geometry,
                                         ud90 = all_uds_90_bound$geometry,
                                         ud75 = all_uds_75_bound$geometry,
                                         ud50 = all_uds_50_bound$geometry,
                                         time_rounded = round_date(time,
                                         unit = "day"))

all_tags_df$age_group <- factor(all_tags_df$age_group, levels = c(1, 2),
                                 labels = c("3 years or younger (n = 6)",
                                           "5 years or older (n = 4)"))

all_tags_df %>% summarise(max_diff_95 = max(diff_95_perc),
                           min_diff_95 = min(diff_95_perc))

##   max_diff_95 min_diff_95
## 1    119.9192     -70.4403

```

Animating space use changes for each individual

Generating animations for all individuals. (eval set to `eval = FALSE` for PDF - html files at GitHub repo at https://github.com/swforrest/kaka_HRandSUV)

```

animation_95 <- ggplot() +
  geom_sf(data = all_tags_df,
          aes(geometry = ud95, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df,
          aes(geometry = ud90, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df,
          aes(geometry = ud75, colour = tag, group = tag)) +

```

```

geom_sf(data = all_tags_df,
        aes(geometry = ud50, colour = tag, group = tag)) +
scale_color_viridis_d(option = "D", direction = 1) +
labs(title = "{closest_state}") +
transition_states(states = time_rounded, transition_length = 5,
                  state_length = 1) +
shadow_wake(wake_length = 0.1, wrap = FALSE) +
geom_sf(aes(geometry = geometry), data = OrokonuiFence,
        colour = "black", fill = "NA") +
theme_bw() +
theme(legend.position = "none")

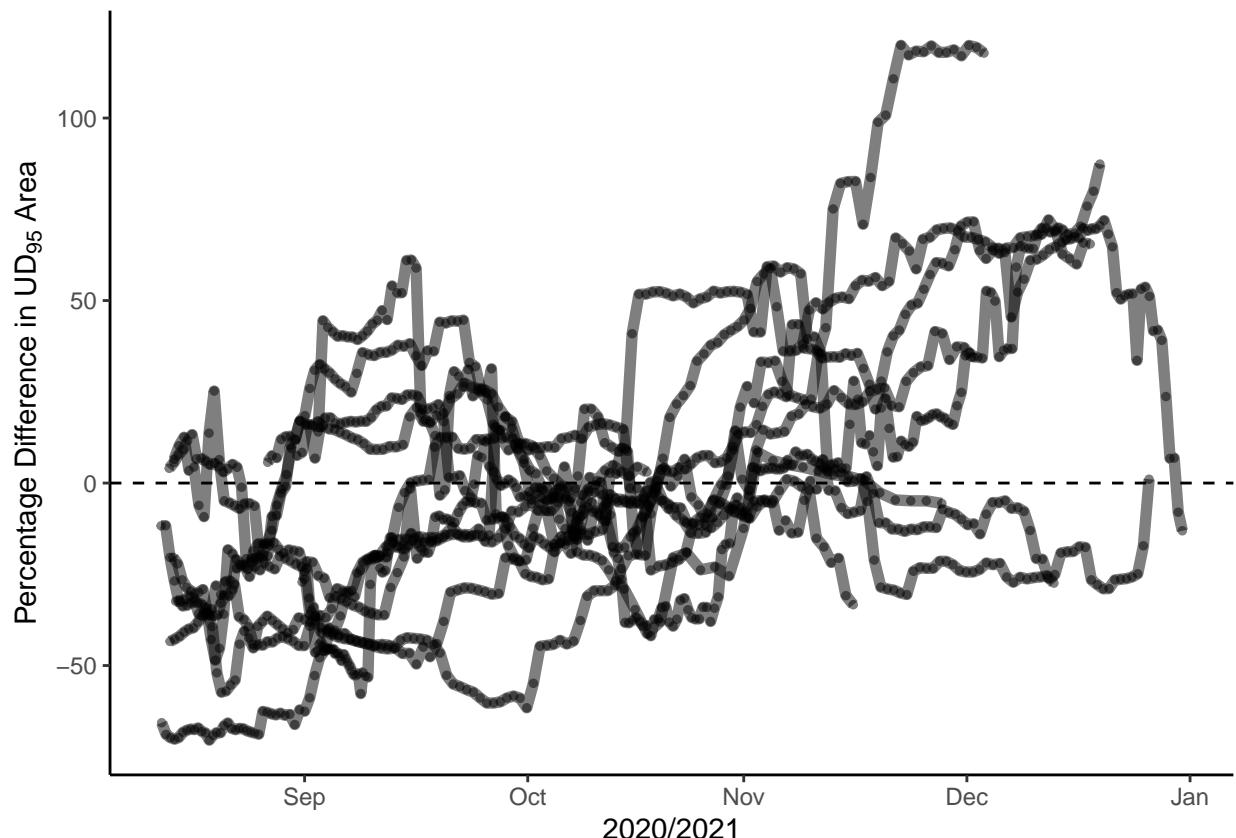
animate(plot = animation_95,
       fps = 10)

```

```

all_tags_df %>% ggplot(aes(x = time, y = diff_95_perc)) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_point(aes(group = tag), alpha = 0.5, size = 1) +
  geom_path(aes(group = tag), alpha = 0.5, size = 2) +
  #geom_smooth(method = "gam", se = T, alpha = 0.25, level = 0.99) +
  scale_y_continuous(expression("Percentage Difference in UD"[95]*" Area")) +
  scale_x_datetime("2020/2021", labels = date_format("%b")) +
  scale_color_viridis_d("ID", option = "D", direction = 1) +
  theme_classic()

```



```

# Generating animations for all young individuals (3 years and younger, n = 6).

animation_95 <- ggplot() +
  geom_sf(data = all_tags_df %>% filter(age <= 4),
          aes(geometry = ud95, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df %>% filter(age <= 4),
          aes(geometry = ud90, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df %>% filter(age <= 4),
          aes(geometry = ud75, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df %>% filter(age <= 4),
          aes(geometry = ud50, colour = tag, group = tag)) +
  scale_color_viridis_d(option = "D", direction = 1) +
  labs(title = "{closest_state}") +
  transition_states(states = time_rounded, transition_length = 5,
                    state_length = 1) +
  shadow_wake(wake_length = 0.1, wrap = FALSE) +
  geom_sf(aes(geometry = geometry), data = OrokouuiFence,
          colour = "black", fill = "NA") +
  theme_bw() +
  theme(legend.position = "none")

animate(plot = animation_95,
        fps = 10)

```

```

# Generating animations for all older individuals (5 years and older, n = 4).

animation_95 <- ggplot() +
  geom_sf(data = all_tags_df %>% filter(age >= 5),
          aes(geometry = ud95, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df %>% filter(age >= 5),
          aes(geometry = ud90, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df %>% filter(age >= 5),
          aes(geometry = ud75, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df %>% filter(age >= 5),
          aes(geometry = ud50, colour = tag, group = tag)) +
  scale_color_viridis_d(option = "D", direction = 1) +
  labs(title = "{closest_state}") +
  transition_states(states = time_rounded, transition_length = 5,
                    state_length = 1) +
  shadow_wake(wake_length = 0.1, wrap = FALSE) +
  geom_sf(aes(geometry = geometry), data = OrokouuiFence,
          colour = "black", fill = "NA") +
  theme_bw() +
  theme(legend.position = "none")

animate(plot = animation_95,
        fps = 10)

```

```

# Generating animations for the two juvenile individuals.
# The UDs closely overlap in the latter half of the study.

```

```

animation_95 <- ggplot() +
  geom_sf(data = all_tags_df %>% filter(age == 1),
          aes(geometry = ud95, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df %>% filter(age == 1),
          aes(geometry = ud90, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df %>% filter(age == 1),
          aes(geometry = ud75, colour = tag, group = tag)) +
  geom_sf(data = all_tags_df %>% filter(age == 1),
          aes(geometry = ud50, colour = tag, group = tag)) +
  scale_color_viridis_d(option = "D", direction = 1) +
  labs(title = "{closest_state}") +
  transition_states(states = time_rounded, transition_length = 5,
                    state_length = 1) +
  shadow_wake(wake_length = 0.1, wrap = FALSE) +
  geom_sf(aes(geometry = geometry), data = OrokouuiFence,
          colour = "black", fill = "NA") +
  theme_bw() +
  theme(legend.position = "none")

animate(plot = animation_95,
        fps = 10)

```

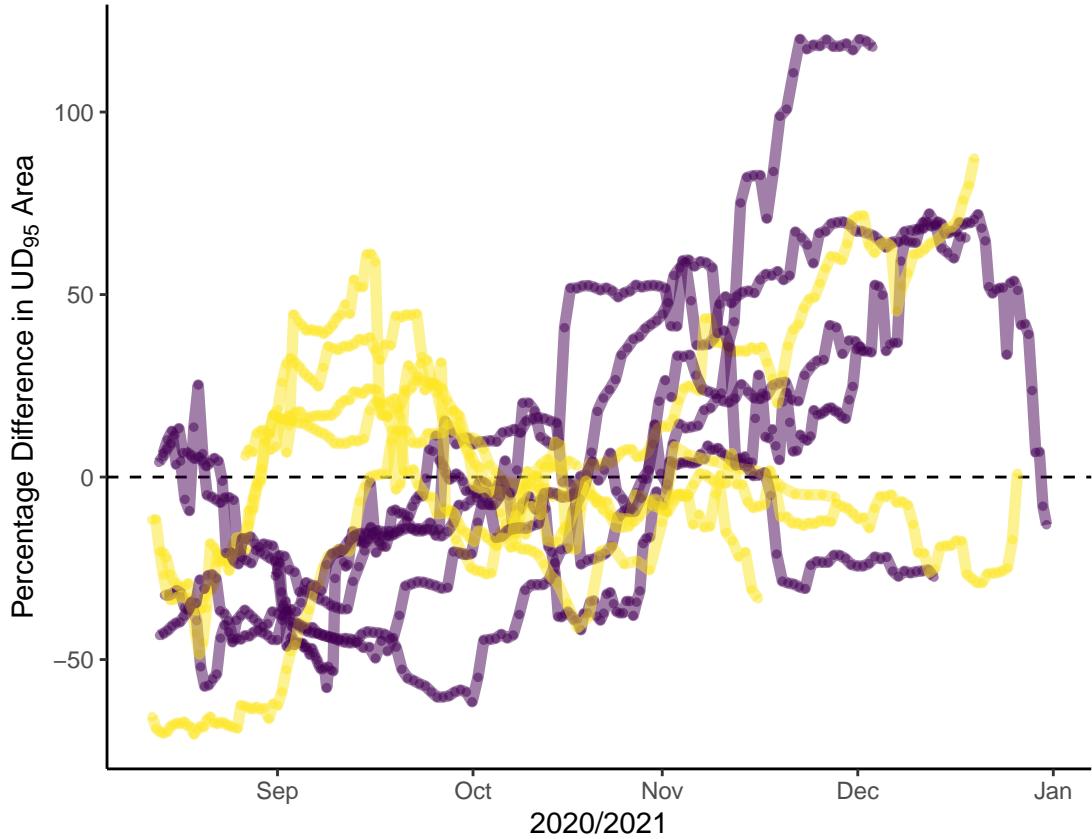
Plotting the expansion and contraction of the UD₉₅ area, relative to *each individual's* average.

Caption from manuscript: Percentage difference compared to average UD95 area for each individual kākā tracked at Orokouui Ecosanctuary in Dunedin. Positive values are above average UD95 area compared to the monthly average for that individual, whereas negative values are below average .

```

all_tags_df %>% ggplot(aes(x = time, y = diff_95_perc, colour = sex)) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_point(aes(group = tag), alpha = 0.5, size = 1) +
  geom_path(aes(group = tag), alpha = 0.5, size = 2) +
  #geom_smooth(method = "gam", se = T, alpha = 0.25, level = 0.99) +
  scale_y_continuous(expression("Percentage Difference in UD"[95]*" Area")) +
  scale_x_datetime("2020/2021", labels = date_format("%b")) +
  scale_color_viridis_d("Sex", option = "D", direction = 1) +
  theme_classic()

```



```
# ggsave("all_tags_increment_area_240_6_20221110.tiff",
#        path = here(paste("OneDrive - Queensland University of Technology/",
#                           "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
#                           "Graphical outputs", sep = "")),
#        width = 180, height = 100, units = "mm", dpi = 300, scale = 1)
```

Plot the location changes of the centroid throughout the tracking period, coloured by age. In this plot it is difficult to discern between the two 1 year old individuals, which both move substantially and are both coloured yellow.

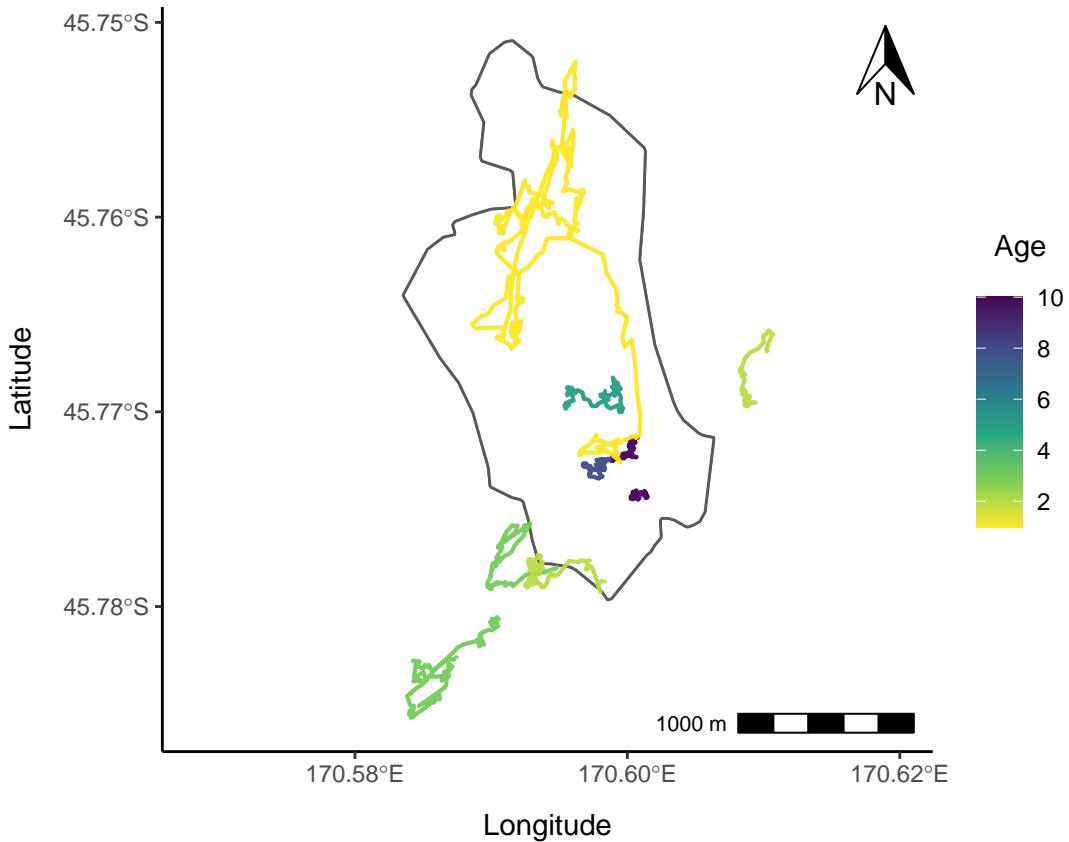
```
drift_xy <- all_tags_df %>% ggplot() +
  geom_sf(aes(geometry = geometry), data = OrokouuiFence, fill = "white") +
  geom_path(aes(x = centX, y = centY, colour = age, group = tag),
            data = all_tags_df, alpha = 0.95, size = 0.75) +
  scale_y_continuous("Latitude",
                     breaks = c(-45.77, -45.78, -45.76, -45.75, -45.79)) +
  scale_x_continuous("Longitude",
                     breaks = c(170.58, 170.60, 170.62),
                     limits = c(1411000, 1415000)) +
  annotation_scale(location = "br") +
  annotation_north_arrow(location = "tr", width = unit(0.75, "cm"),
                         height = unit(1, "cm")) +
```

```

theme_classic() +
theme(legend.title.align = 0.5,
      legend.spacing.y = unit(0.5, "cm"),
      axis.title.y = element_text(margin = margin(t = 0, r = 10, b = 0, l = 0)),
      axis.title.x = element_text(margin = margin(t = 10, r = 0, b = 0, l = 0))) +
scale_colour_viridis_c(breaks = c(0,2,4,6,8,10), name = "Age", direction = -1)

drift_xy

```



```

# ggsave("all_tags_increment_drift_xy_age_240_6_16thDec21.tiff",
#        path = here(paste("OneDrive - Queensland University of Technology/",
#                           "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
#                           "Graphical outputs", sep = "")),
#        width = 180, height = 100, units = "mm", dpi = 300)

```

Plot the temporal UD overlap with Orokonui Ecosanctuary. In this case it is the proportion **outside** of the fence, to indicate risk exposure.

```

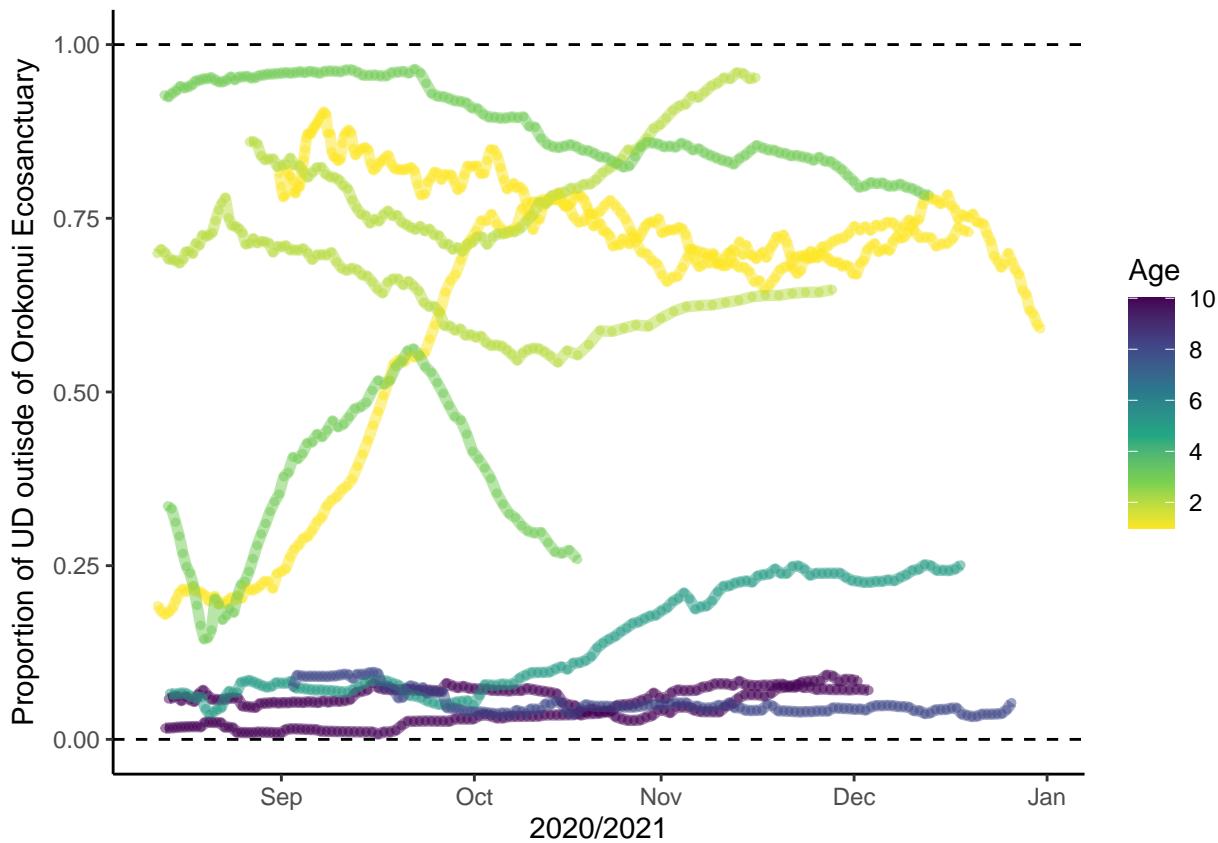
all_tags_df %>% ggplot(aes(x = time, y = outside, colour = age)) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_hline(yintercept = 1, linetype = "dashed") +
  geom_point(aes(group = tag), alpha = 0.5, size = 1) +

```

```

geom_path(aes(group = tag), alpha = 0.5, size = 2) +
#geom_smooth(method = "gam", se = T, alpha = 0.25, level = 0.99) +
#scale_colour_manual(values = c("orange", "skyblue")) + # name = "Breeding",
scale_y_continuous(expression("Proportion of UD outside of Orokonui Ecosanctuary")) +
scale_x_datetime("2020/2021", labels = date_format("%b")) +
#scale_color_viridis_d("Individual", option = "C", direction = 1) +
scale_color_viridis_c(breaks = c(0,2,4,6,8,10), name = "Age", direction = -1) +
theme_classic()

```



```

# ggsave("all_tags_increment_overlap_tag_240_6_20221110.tiff",
#        path = here(paste("OneDrive - Queensland University of Technology/",
#                           "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
#                           "Graphical outputs", sep = "")),
#        width = 180, height = 100, units = "mm", dpi = 300)

```

Compare the similarity matrices, expansion and contraction relative the their average area (UD_{95}) and centroid drift of two individuals, T08 and T14, who are a 1-year-old female and 10-year-old male, respectively. Centroid drift and initial and final UDs.

```

# to show viridis colour codes to manually add to plot
#show_col(viridis_pal()(10))

# create UD contours
t08_1st_95 <- raster2contour(all_tags_subsets[[4]][[1]], levels = 0.95)

```

```

t08_last_95 <- raster2contour(all_tags_subsets[[4]][[127]], levels = 0.95)

t08_1st_90 <- raster2contour(all_tags_subsets[[4]][[1]], levels = 0.9)
t08_last_90 <- raster2contour(all_tags_subsets[[4]][[127]], levels = 0.9)

t08_1st_75 <- raster2contour(all_tags_subsets[[4]][[1]], levels = 0.75)
t08_last_75 <- raster2contour(all_tags_subsets[[4]][[127]], levels = 0.75)

t08_1st_50 <- raster2contour(all_tags_subsets[[4]][[1]], levels = 0.50)
t08_last_50 <- raster2contour(all_tags_subsets[[4]][[127]], levels = 0.50) %>%
  st_as_sf() %>%
  st_polygonize()

cent_08_plot <- ggplot() +

  geom_polygon(data = t08_last_95,
               aes(x = long, y = lat),
               alpha = 0.25,
               fill = "#FDE725FF",
               colour = "black",
               #linetype = "dashed",
               size = 0.15) +

  geom_polygon(data = t08_last_90,
               aes(x = long, y = lat),
               alpha = 0.25,
               fill = "#FDE725FF",
               #colour = "black",
               size = 0.15,
               linetype = "dashed") +

  geom_polygon(data = t08_last_75,
               aes(x = long, y = lat),
               alpha = 0.25,
               fill = "#FDE725FF",
               #colour = "black",
               size = 0.15,
               linetype = "dashed") +

# geom_polygon(data = t08_last_50,
#               aes(x = long, y = lat),
#               alpha = 0.95,
#               fill = "#FDE725FF",
#               #colour = "black",
#               size = 0.15,
#               linetype = "dashed") +

  geom_sf(data = t08_last_50, # to replace the above component
          alpha = 0.95,
          fill = "#FDE725FF",
          colour = NA) +

  geom_polygon(data = t08_1st_95,

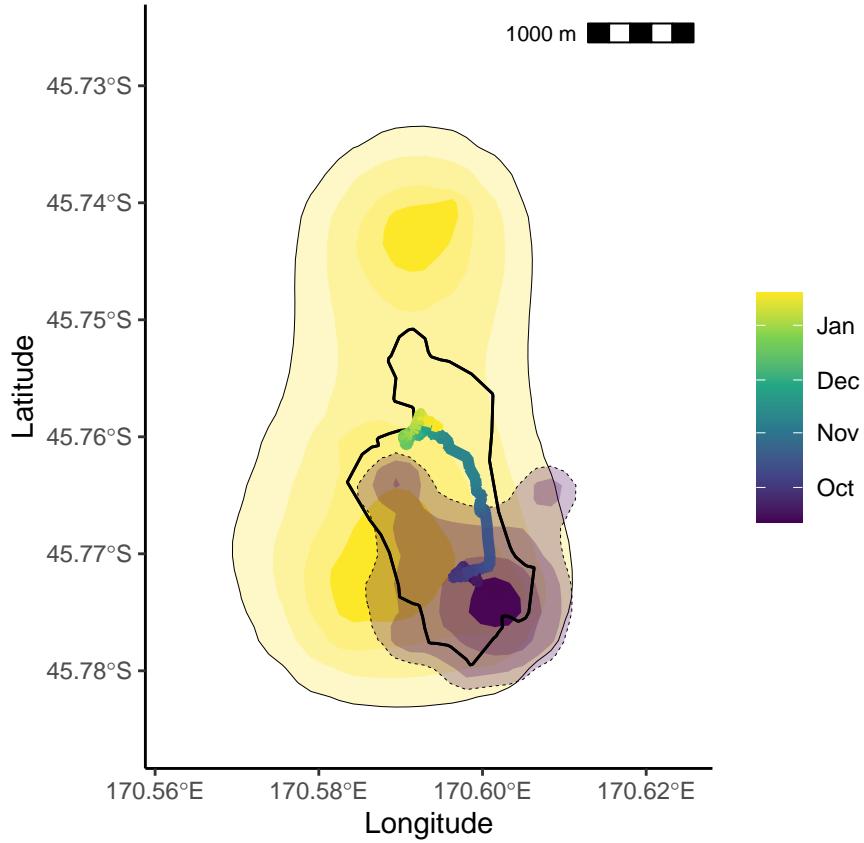
```

```

aes(x = long, y = lat),
alpha = 0.25,
fill = "#440154FF",
colour = "black",
size = 0.15,
linetype = "dashed") +
geom_polygon(data = t08_1st_90,
aes(x = long, y = lat),
alpha = 0.25,
fill = "#440154FF",
#colour = "black",
size = 0.15,
linetype = "dashed") +
geom_polygon(data = t08_1st_75,
aes(x = long, y = lat),
alpha = 0.25,
fill = "#440154FF",
#colour = "black",
size = 0.15,
linetype = "dashed") +
geom_polygon(data = t08_1st_50,
aes(x = long, y = lat),
alpha = 0.95,
fill = "#440154FF",
#colour = "black",
size = 0.15,
linetype = "dashed") +
geom_sf(aes(geometry = geometry), data = OrokonuiFence,
colour = "black", fill = "NA") +
geom_point(data = all_tags_df %>% filter(tag == 8),
aes(x = centX, y = centY, colour = time, group = tag),
alpha = 1, size = 1) +
geom_path(data = all_tags_df %>% filter(tag == 8),
aes(x = centX, y = centY, colour = time, group = tag),
alpha = 1, size = 2) +
scale_colour_viridis_c(name = "", trans = "time") +
scale_y_continuous("Latitude", limits = c(4926900, 4933500)) +
scale_x_continuous("Longitude", limits = c(1410500, 1415400),
breaks = seq(170.56, 170.62, by = 0.02)) +
annotation_scale(location = "tr") + # from ggspatial package
theme_classic()

cent_08_plot

```



```
t14_1st_95 <- raster2contour(all_tags_subsets[[10]][[1]], levels = 0.95)
t14_last_95 <- raster2contour(all_tags_subsets[[10]][[111]], levels = 0.95)

t14_1st_90 <- raster2contour(all_tags_subsets[[10]][[1]], levels = 0.9)
t14_last_90 <- raster2contour(all_tags_subsets[[10]][[111]], levels = 0.9)

t14_1st_75 <- raster2contour(all_tags_subsets[[10]][[1]], levels = 0.75)
t14_last_75 <- raster2contour(all_tags_subsets[[10]][[111]], levels = 0.75)

t14_1st_50 <- raster2contour(all_tags_subsets[[10]][[1]], levels = 0.50)
t14_last_50 <- raster2contour(all_tags_subsets[[10]][[111]], levels = 0.50)

cent_14_plot <- all_tags_df %>% filter(tag == "45514") %>% ggplot() +
  geom_polygon(data = t14_last_95,
               aes(x = long, y = lat),
               alpha = 0.25,
               fill = "#FDE725FF",
               colour = "black",
               #linetype = "dashed",
               size = 0.15) +
  geom_polygon(data = t14_last_90,
               aes(x = long, y = lat),
               alpha = 0.25,
```

```

    fill = "#FDE725FF",
    #colour = "black",
    size = 0.15,
    linetype = "dashed") +
geom_polygon(data = t14_last_75,
             aes(x = long, y = lat),
             alpha = 0.25,
             fill = "#FDE725FF",
             #colour = "black",
             size = 0.15,
             linetype = "dashed") +
geom_polygon(data = t14_last_50,
             aes(x = long, y = lat),
             alpha = 0.95,
             fill = "#FDE725FF",
             #colour = "black",
             size = 0.15,
             linetype = "dashed") +
geom_polygon(data = t14_1st_95,
             aes(x = long, y = lat),
             alpha = 0.25,
             fill = "#440154FF",
             colour = "black",
             size = 0.15,
             linetype = "dashed") +
geom_polygon(data = t14_1st_90,
             aes(x = long, y = lat),
             alpha = 0.25,
             fill = "#440154FF",
             #colour = "black",
             size = 0.15,
             linetype = "dashed") +
geom_polygon(data = t14_1st_75,
             aes(x = long, y = lat),
             alpha = 0.25,
             fill = "#440154FF",
             #colour = "black",
             size = 0.25,
             linetype = "dashed") +
geom_polygon(data = t14_1st_50,
             aes(x = long, y = lat),
             alpha = 0.95,
             fill = "#440154FF",
             #colour = "black",
             size = 0.15,
             linetype = "dashed") +

```

```

geom_sf(aes(geometry = geometry), data = OrokonuiFence,
        colour = "black", fill = "NA") +

geom_point(data = all_tags_df %>% filter(tag == 14),
            aes(x = centX, y = centY, colour = time, group = tag),
            alpha = 1, size = 1) +

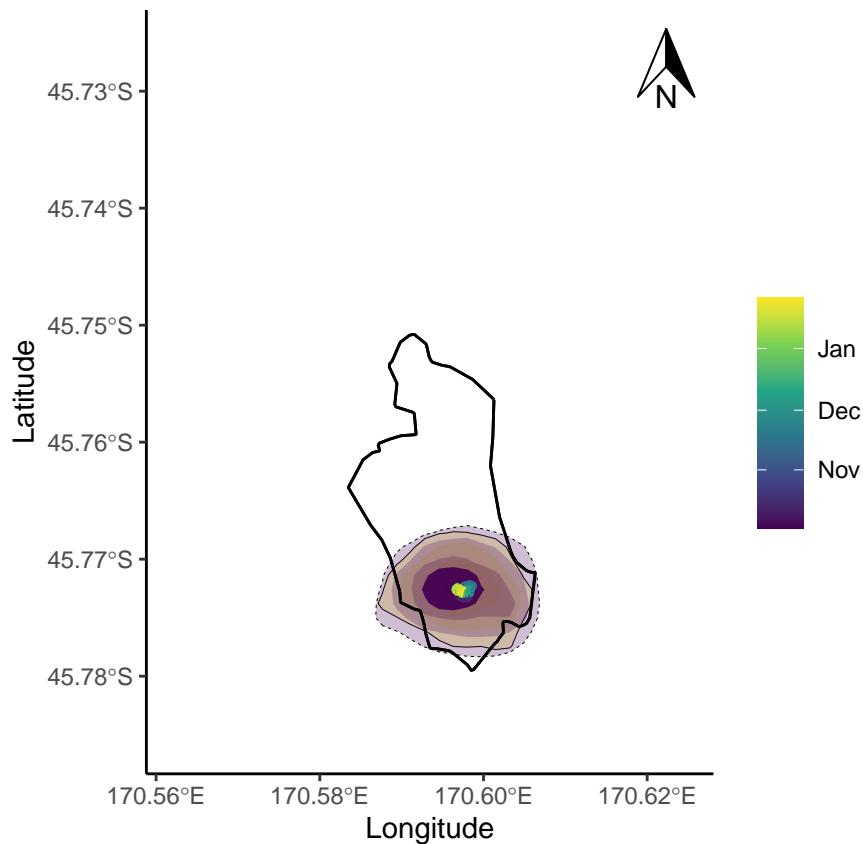
geom_path(data = all_tags_df %>% filter(tag == 14),
            aes(x = centX, y = centY, colour = time, group = tag),
            alpha = 1, size = 2) +

scale_colour_viridis_c(name = "", trans = "time") +
scale_y_continuous("Latitude", limits = c(4926900, 4933500)) +
scale_x_continuous("Longitude", limits = c(1410500, 1415400),
                   breaks = seq(170.56, 170.62, by = 0.02)) +

annotation_north_arrow(location = "tr", width = unit(0.75, "cm"),
                        height = unit(1, "cm")) +
theme_classic()

cent_14_plot

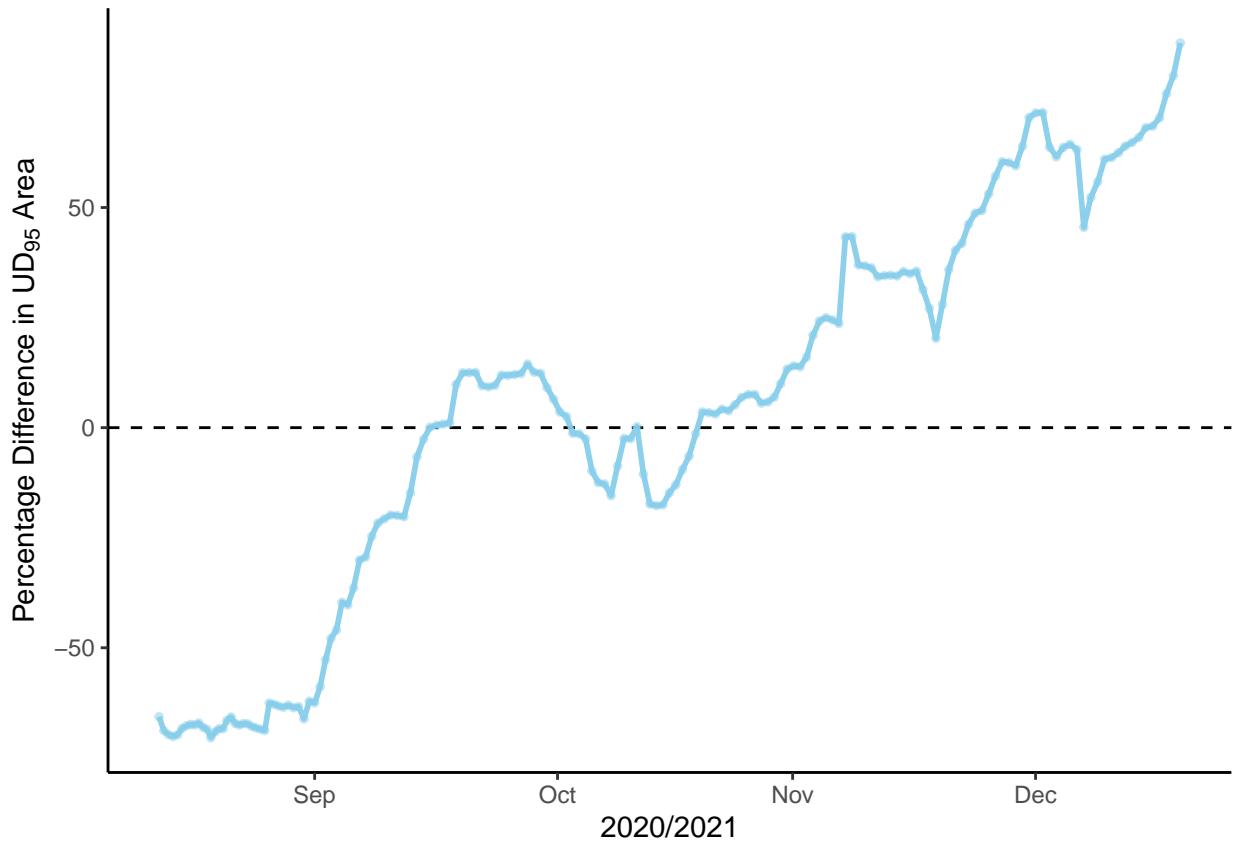
```



Expansion/contraction.

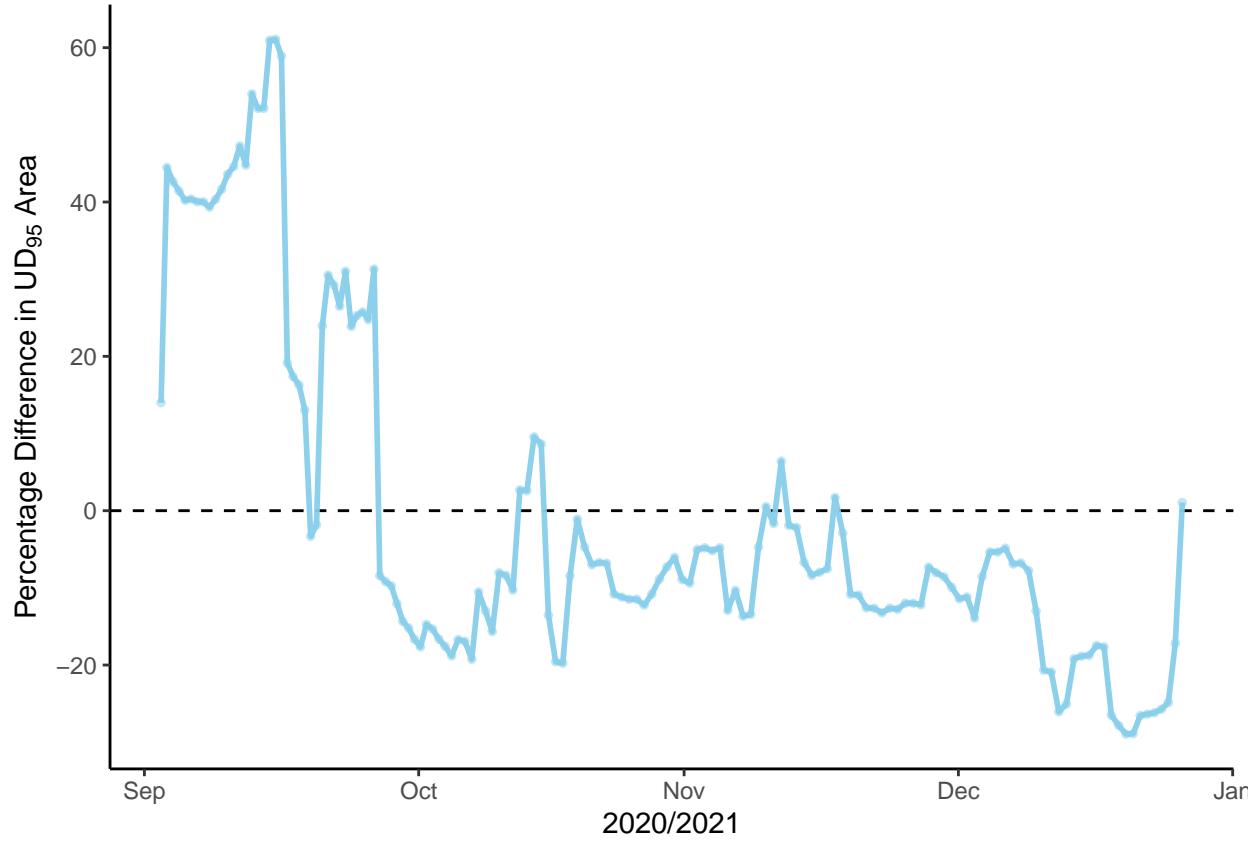
```
t08_ud95_area_plot <- all_tags_df %>% filter(tag == "8") %>%
  ggplot(aes(x = time, y = diff_95_perc)) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_point(aes(group = tag), alpha = 0.5, size = 1, colour = "skyblue") +
  geom_path(aes(group = tag), alpha = 0.95, size = 1, colour = "skyblue") +
  scale_colour_manual(name = "Sex", values = c("skyblue", "orange")) +
  scale_y_continuous(expression("Percentage Difference in UD"[95]*" Area")) +
  scale_x_datetime("2020/2021", labels = date_format("%b")) +
  theme_classic()

t08_ud95_area_plot
```



```
t14_ud95_area_plot <- all_tags_df %>% filter(tag == "14") %>%
  ggplot(aes(x = time, y = diff_95_perc)) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_point(aes(group = tag), alpha = 0.5, size = 1, colour = "skyblue") +
  geom_path(aes(group = tag), alpha = 0.95, size = 1, colour = "skyblue") +
  scale_colour_manual(name = "Sex", values = c("skyblue", "orange")) +
  scale_y_continuous(expression("Percentage Difference in UD"[95]*" Area")) +
  scale_x_datetime("2020/2021", labels = date_format("%b")) +
  theme_classic()

t14_ud95_area_plot
```

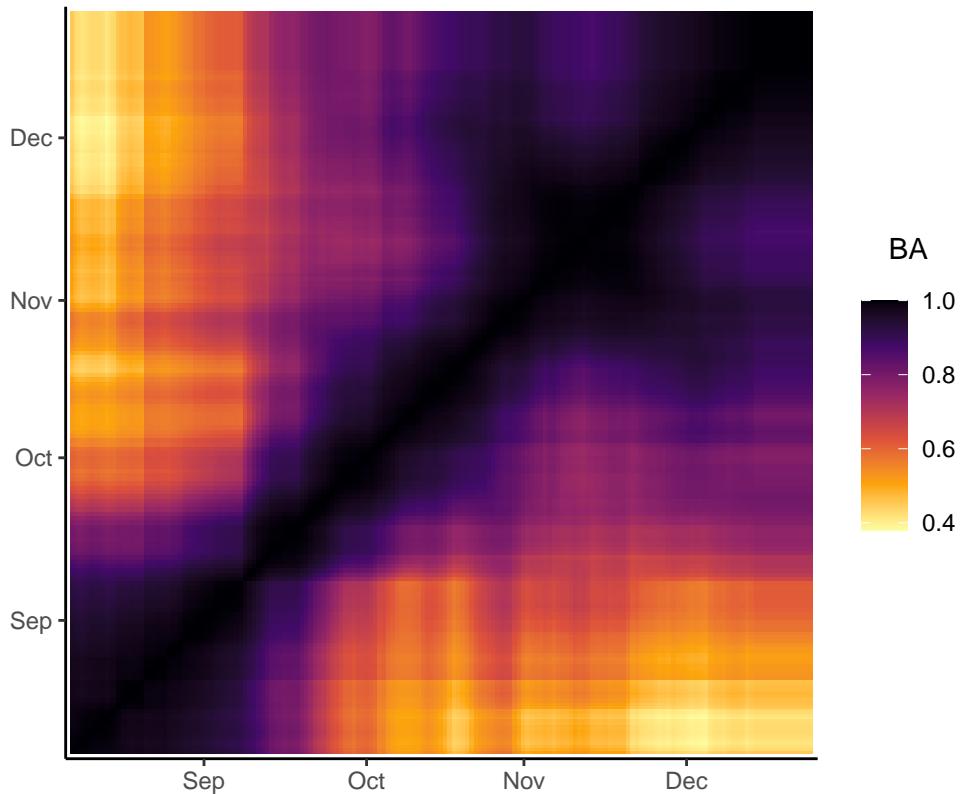


Bhattacharyya's Affinity similarity matrices for T08 and T14 plots.

```
t08 <- corr.plots[[4]] + theme(axis.title.x = element_blank(),
                               axis.title.y = element_blank()) +
  ggtitle("Female, 1 year old")

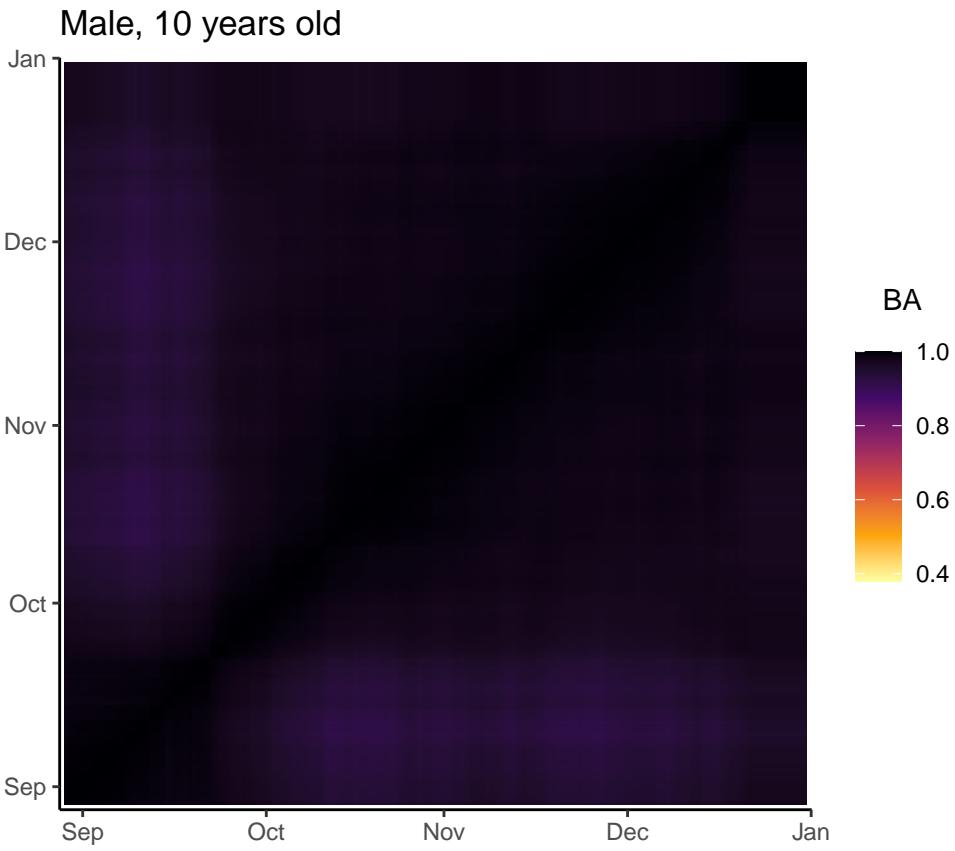
t08
```

Female, 1 year old



```
t14 <- corr.plots[[10]] + theme(axis.title.x = element_blank(),
                                axis.title.y = element_blank()) +
  ggtitle("Male, 10 years old")
```

```
t14
```



Combining the plots for the manuscript.

Caption from manuscript: The left column represents the temporal space use variability (SUV) of an individual female kākā of age 1 year, and the right column represents the temporal space use variability (SUV) of a 10-year-old male kākā. The upper panels show the similarity matrices of Bhattacharyya's Affinity (BA) throughout the study period. The middle panels show the percentage difference in UD95 area, compared to each individual's average UD95 area for the full study period. The lower panels show the drift of UD95 centroid on the same scale, with the fence of Orokonui Ecosanctuary represented as the black outline. The lower panels also show the initial (purple) and final (yellow) UDs for each individual.

```
t08_ud95 <- t08_ud95_area_plot +
  scale_y_continuous(expression("Percentage Difference in UD"[95]*" Area"),
    limits = c(-100, 100))
```

```
## Scale for 'y' is already present. Adding another scale for 'y', which will
## replace the existing scale.
```

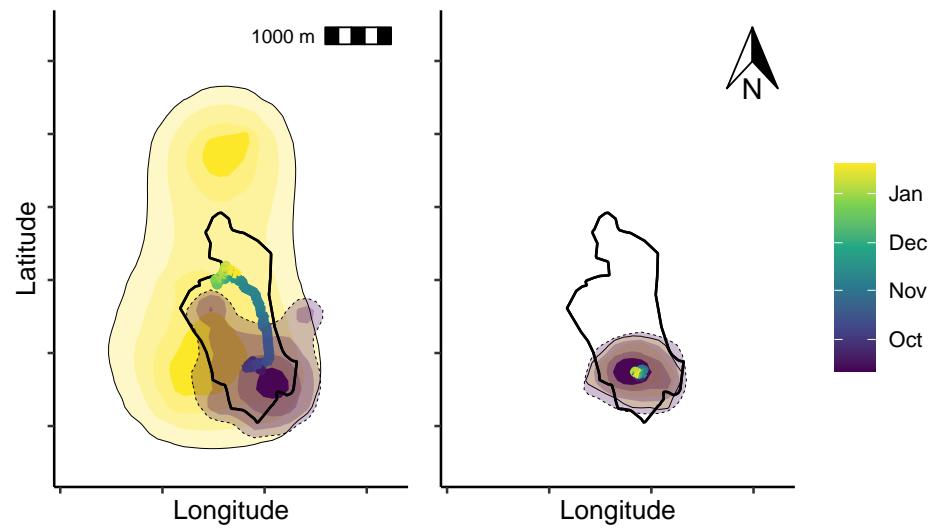
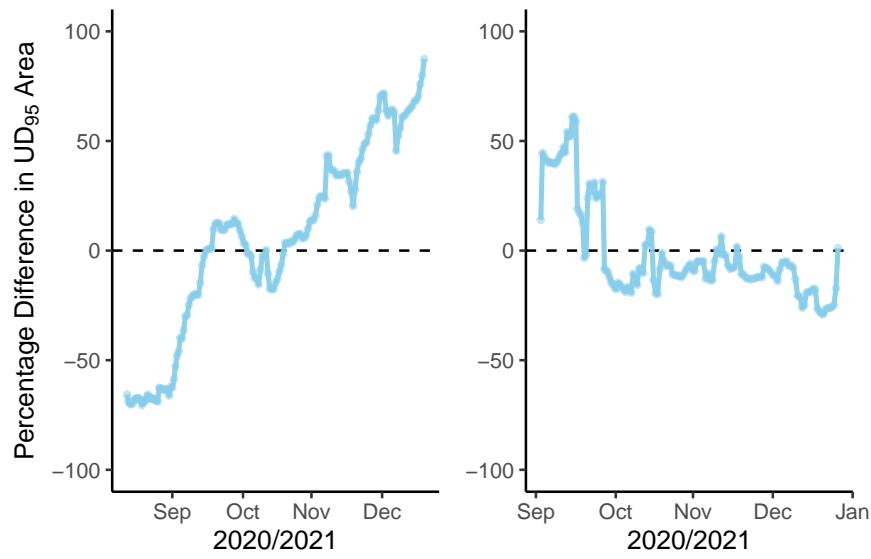
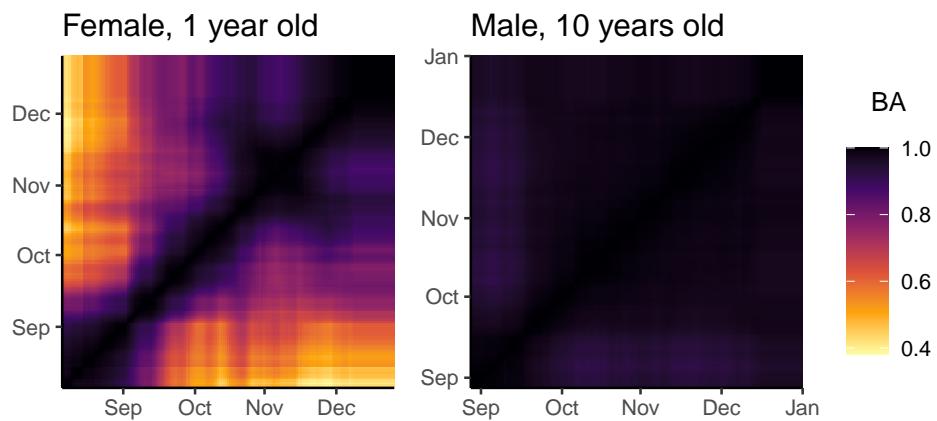
```
t14_ud95 <- t14_ud95_area_plot +
  scale_y_continuous(limits = c(-100, 100)) +
  theme(axis.title.y = element_blank())
```

```
## Scale for 'y' is already present. Adding another scale for 'y', which will
## replace the existing scale.
```

```
t08_cent <- cent_08_plot + theme(axis.text.x = element_blank(),
                                   axis.text.y = element_blank())

t14_cent <- cent_14_plot + theme(axis.text.x = element_blank(),
                                   axis.title.y = element_blank(),
                                   axis.text.y = element_blank(),
                                   legend.position = "none")

(t08 + t14 + plot_layout(guides = "collect")) /
  (t08_ud95 + t14_ud95) /
  (t08_cent + t14_cent + plot_layout(guides = "collect")) +
  theme(plot.margin=grid::unit(c(0,0,0,0), "mm"))
```



```
# ggsave("t08_t14_tags_dist-area-centroid_increment_240_6_20221110.tiff",
#        path = here(paste("OneDrive - Queensland University of Technology/",
#                           "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
#                           "Graphical outputs", sep = "")),
#        width = 160, height = 240, units = "mm", dpi = 300)
```

Drift (displacement between successive centroids)

```
#create empty vector
d <- c()

for (i in 1:length(all_areas_dfs)) {
  for (j in 1:nrow(all_areas_dfs[[i]])) {

    d <- c(d, sqrt((all_areas_dfs[[i]]$centX[[j]] -
                    all_areas_dfs[[i]]$centX[[max(j-1, 1)]]))^2 +
            (all_areas_dfs[[i]]$centY[[j]] -
             all_areas_dfs[[i]]$centY[[max(j-1, 1)]]))^2))

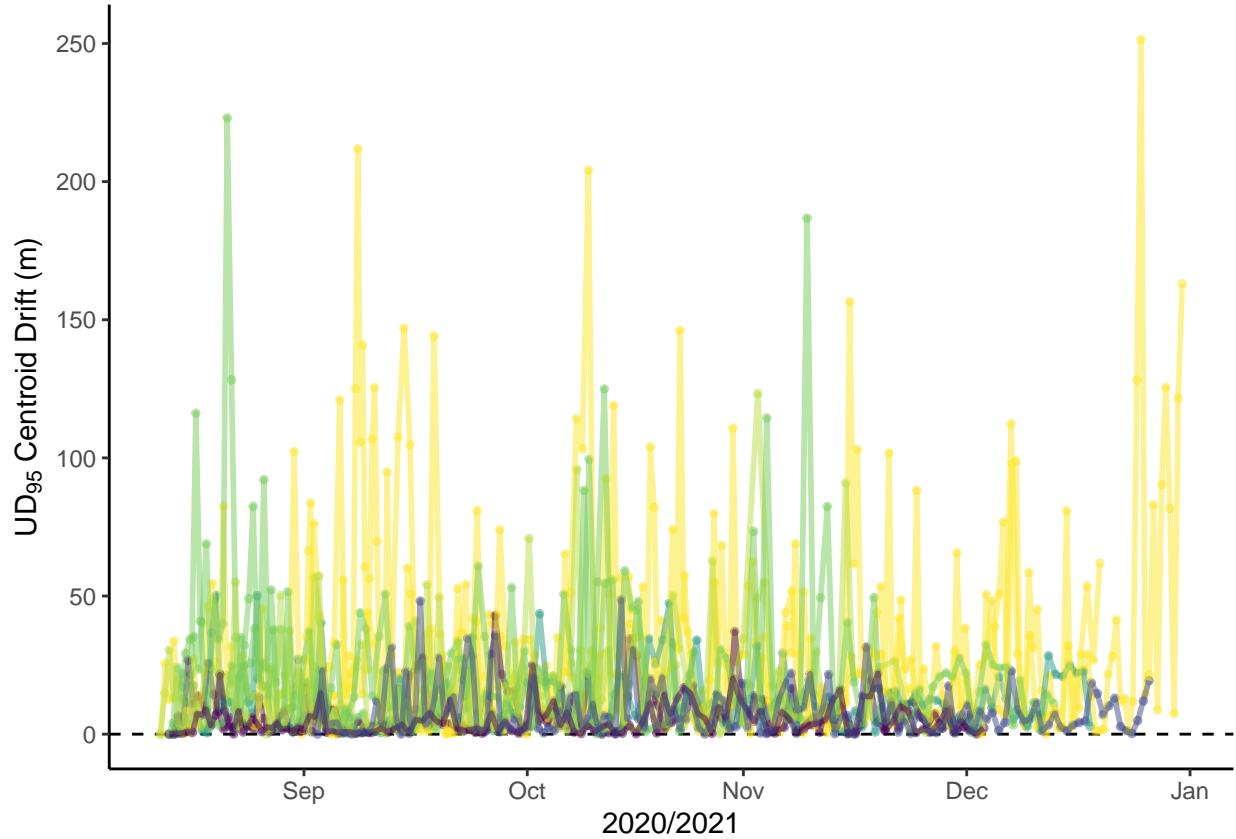
  }
}

#add to data frame
all_tags_df$drift <- d
```

Plotting the displacement throughout time. Assessing whether there is a temporal pattern to space use positional changes.

```
drift_m <- all_tags_df %>% ggplot(aes(x = time, y = drift, colour = age)) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_point(aes(group = tag), alpha = 0.5, size = 1) +
  geom_path(aes(group = tag), alpha = 0.5, size = 1) +
  #geom_smooth(method = "gam", se = T, alpha = 0.25, level = 0.99) +
  #scale_colour_manual(values = c("red", "green")) + # name = "Breeding",
  scale_y_continuous(expression("UD"[95]*" Centroid Drift (m)")) +
  scale_x_datetime("2020/2021", labels = date_format("%b")) +
  scale_colour_viridis_c(breaks = c(0,2,4,6,8,10), name = "Age", direction = -1) +
  theme_classic() +
  theme(legend.position = "none")

drift_m
```

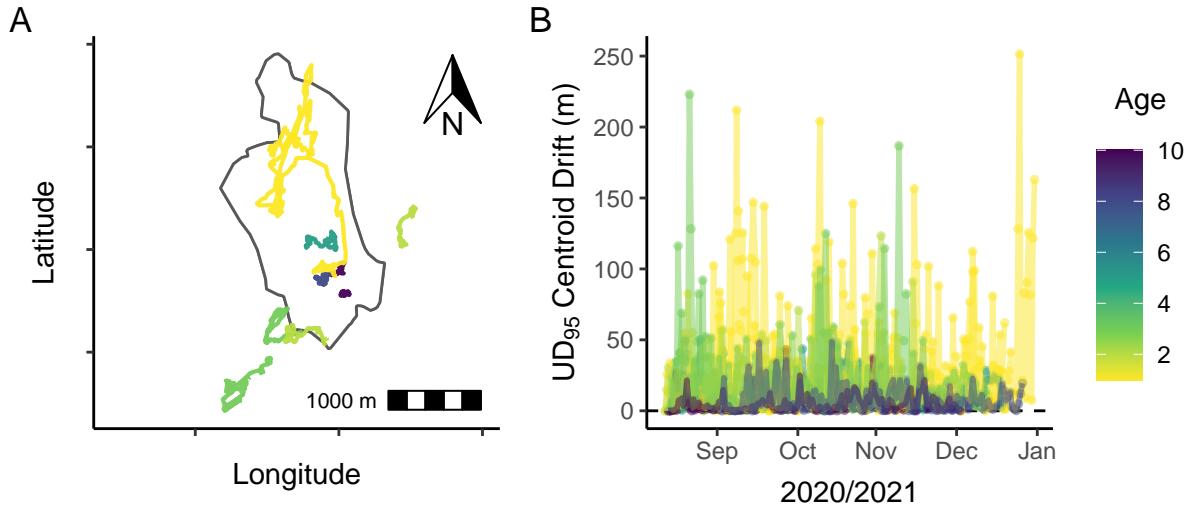


```
# ggsave("all_tags_increment_drift_m_age_240_6_20221110.tiff",
#        path = here(paste("OneDrive - Queensland University of Technology/",
#                           "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
#                           "Graphical outputs", sep = "")),
#        width = 180, height = 100, units = "mm", dpi = 300)
```

Combining plots for manuscript.

Caption from manuscript: A) Position of the UD₉₅ centroid for each individual kākā throughout the study period, in relation to age (colour – legend to the right). The fence of Orokouui Ecosanctuary is shown in black. The scale bar is in 200m increments and is 1000m in total. B) Drift distance of UD₉₅ centroid between space use increments, there were no discernible temporal trends in space use drift.

```
drift_xy + theme(axis.text.x = element_blank(), axis.text.y = element_blank()) +
  drift_m + plot_layout(guides = "collect") + plot_annotation(tag_levels = 'A')
```



```
# ggsave("drift_xy_drift_m_240_6_20221110.tiff",
#        path = here(paste("OneDrive - Queensland University of Technology/",
#                           "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
#                           "Graphical outputs", sep = "")),
#        width = 180, height = 80, units = "mm", dpi = 300)
```

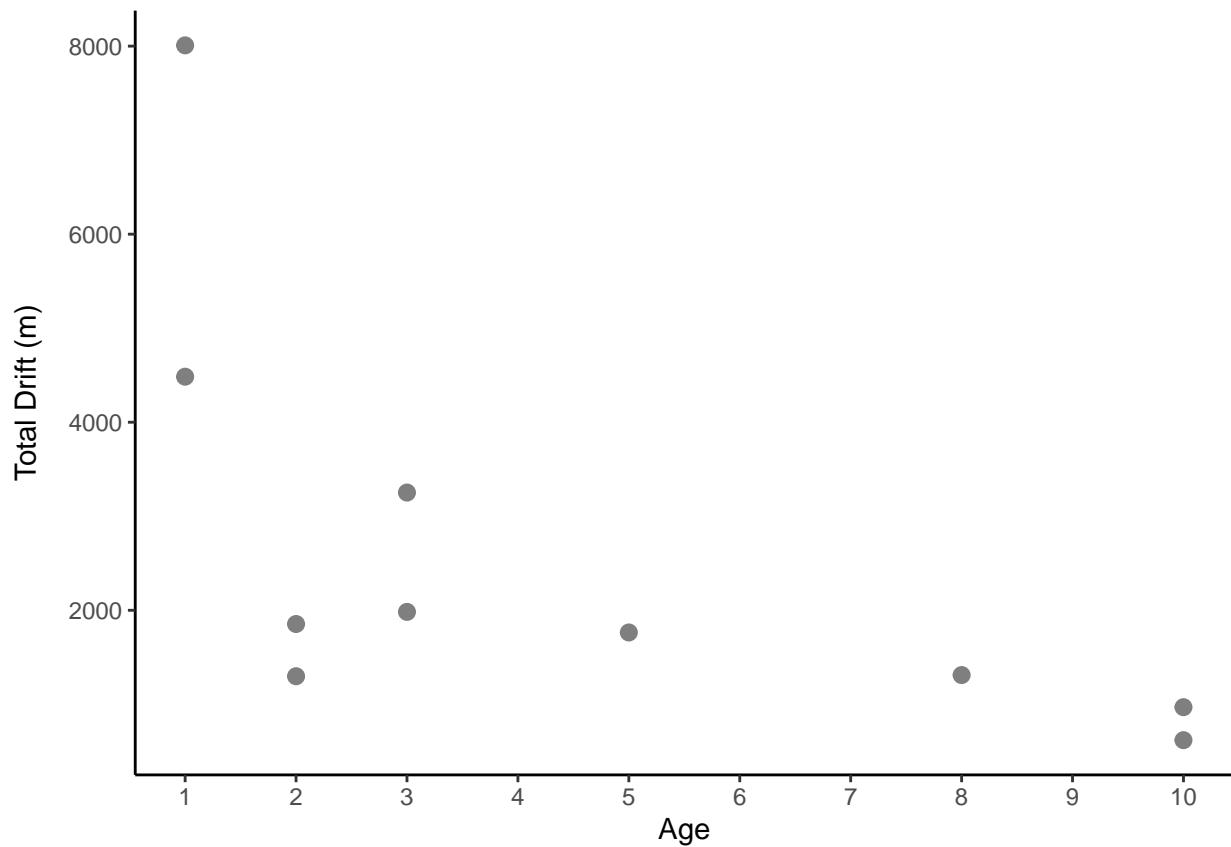
Assessing total and average drift against age. Substantial age-related variation present.

```
drift_by_tag <- all_tags_df %>% group_by(tag) %>%
  summarise(tot_drift = sum(drift), n = n(), ave_drift = sum(drift)/n())

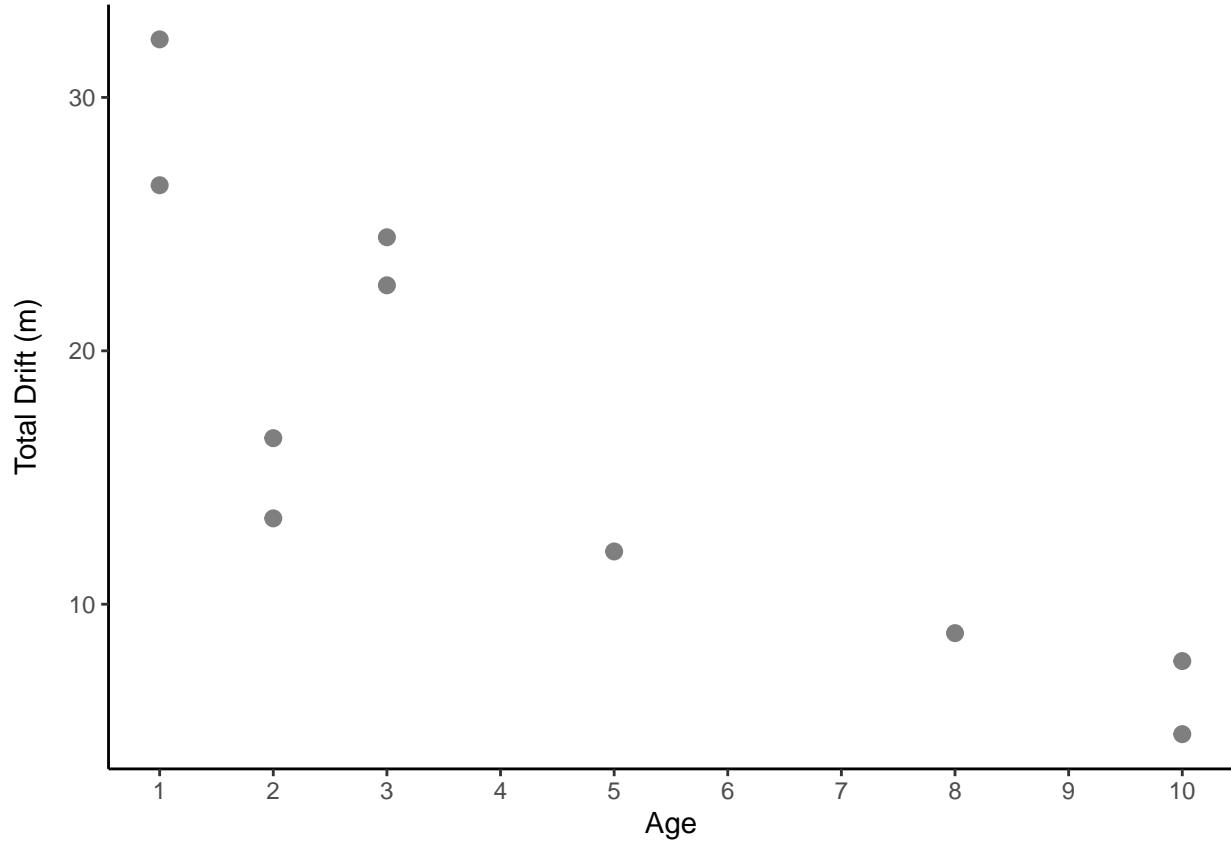
id <- 45505:45514
sex <- c("M", "M", "M", "F", "F", "F", "M", "M", "F")
age <- c(1, 10, 5, 1, 3, 2, 2, 3, 10, 8)
breeding <- c("N", "N", "N", "N", "Y", "N", "Y", "N", "N", "N")

drift_by_tag <- drift_by_tag %>% mutate(sex = sex, age = age)

drift_by_tag %>% ggplot(aes(x = age, y = tot_drift)) +
  geom_point(size = 2.5, alpha = 0.5) +
  scale_y_continuous("Total Drift (m)") +
  scale_x_continuous("Age", breaks = c(1:10)) +
  theme_classic() +
  theme(axis.title.y = element_text(margin = margin(r = 10)))
```



```
# use average drift as the number of UDs calculated are not equivalent
drift_by_tag %>% ggplot(aes(x = age, y = ave_drift)) +
  geom_point(size = 2.5, alpha = 0.5) +
  scale_y_continuous("Total Drift (m)") +
  scale_x_continuous("Age", breaks = c(1:10)) +
  theme_classic() +
  theme(axis.title.y = element_text(margin = margin(r = 10)))
```



Saving images and csv files to easily return to the analysis without having to re-run slow functions of creating UDs etc.

```
# save.image(here(paste("OneDrive - Queensland University of Technology/",
#                   "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
#                   "R objects/image3_20221110.RData", sep = "")))
#
# load(here(paste("OneDrive - Queensland University of Technology/",
#                 "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
#                 "R objects/image3_20221110.RData", sep = "")))
#
# write_csv(all_tags_df,
#           here(paste("OneDrive - Queensland University of Technology/",
#                     "MSc - Scott Forrest/DATA/Modelling/dBBMM/CSV outputs/",
#                     "all_tags_df_window240_increment6_with_diff_20221110.csv",
#                     sep = "")))
#
# all_tags_df <- read_csv(here(paste("OneDrive - Queensland University of Technology/",
#                                 "MSc - Scott Forrest/DATA/Modelling/dBBMM/CSV outputs/",
#                                 "all_tags_df_window240_increment6_with_diff_20221110.csv",
#                                 sep = "")))
#
# all_tags_df <- all_tags_df %>% mutate(tag = factor(tag), sex = factor(sex))
```

Drift statistics (GLM)

Fitting the model.

```
driftglm <- glm(ave_drift ~ age, data = drift_by_tag, family = Gamma(link = "log"))

summary(driftglm)
```

```
## 
## Call:
## glm(formula = ave_drift ~ age, family = Gamma(link = "log"),
##      data = drift_by_tag)
## 
## Deviance Residuals:
##       Min        1Q     Median        3Q       Max
## -0.46561  -0.21916   0.02375   0.20559   0.26732
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 3.41334   0.13476  25.329 6.32e-09 ***
## age         -0.15713   0.02393  -6.565 0.000176 ***  
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for Gamma family taken to be 0.06559517)
## 
## Null deviance: 3.04493  on 9  degrees of freedom
## Residual deviance: 0.57617  on 8  degrees of freedom
## AIC: 59.484
## 
## Number of Fisher Scoring iterations: 4
```

```
AIC(driftglm) # if comparing to other models
```

```
## [1] 59.48413
```

```
Anova(driftglm)
```

```
## Analysis of Deviance Table (Type II tests)
## 
## Response: ave_drift
##          LR Chisq Df Pr(>Chisq)
## age     37.636  1  8.524e-10 ***
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(driftglm,test="F")
```

```
## Analysis of Deviance Table
## 
## Model: Gamma, link: log
```

```

## 
## Response: ave_drift
## 
## Terms added sequentially (first to last)
## 
##          Df Deviance Resid. Df Resid. Dev      F    Pr(>F)
## NULL             9   3.04493
## age     1   2.4688      8   0.57617 37.636 0.0002786 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
r.squaredLR(driftglm)
```

```

## [1] 0.8183769
## attr(),"adj.r.squared")
## [1] 0.8190844

```

```
confint(driftglm)
```

```
## Waiting for profiling to be done...
```

```

##           2.5 %    97.5 %
## (Intercept) 3.1566983 3.6845807
## age         -0.2032149 -0.1094599

```

Model checking.

```

dfun <- function(object) {
  with(object,sum((weights * residuals^2)[weights > 0])/df.residual)
}

dfun(driftglm)

```

```
## [1] 0.06559517
```

```

pseudoR2 <- 1 - (driftglm$deviance / driftglm$null.deviance)
pseudoR2

```

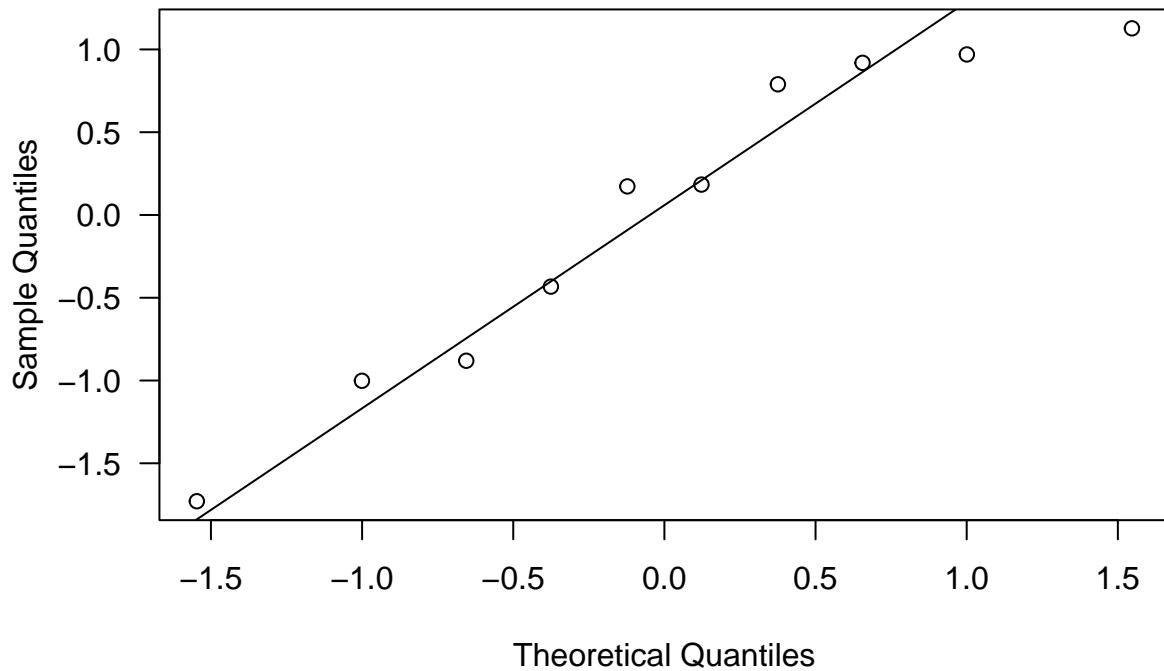
```
## [1] 0.8107768
```

```

qr.driftglm <- qresid(driftglm)
qqnorm(qr.driftglm, las = 1)
qqline(qr.driftglm)

```

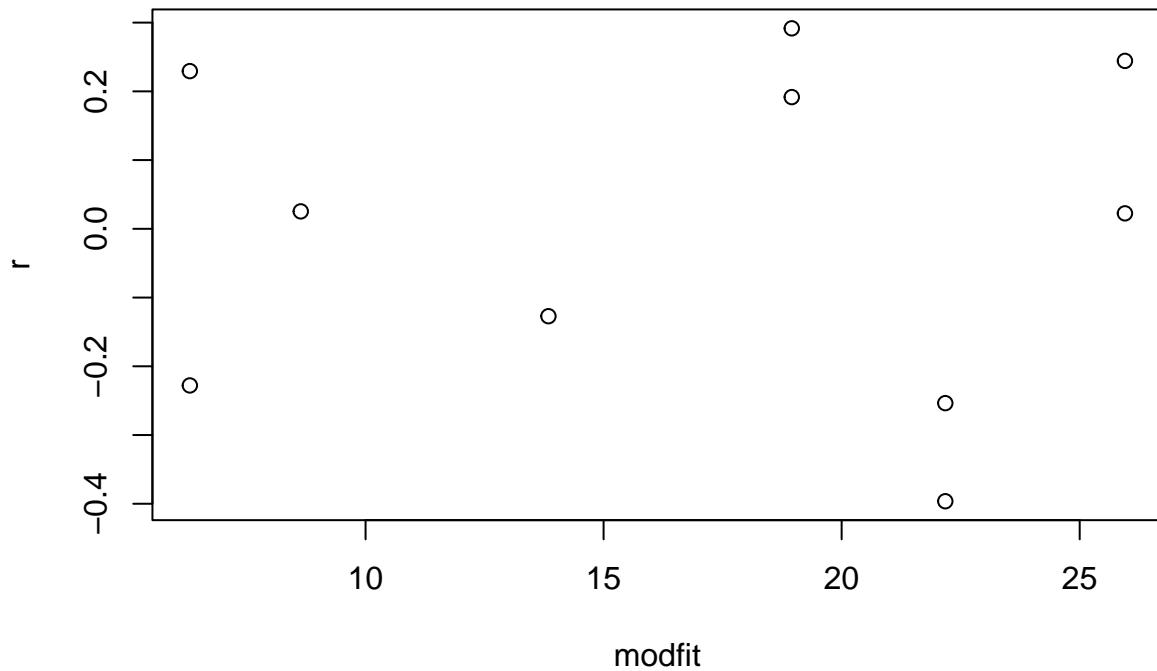
Normal Q-Q Plot



```
r<-residuals(driftglm, type="pearson")
modfit<-fitted(driftglm)

plot(r~modfit, main = "Pearson's Residuals")
```

Pearson's Residuals

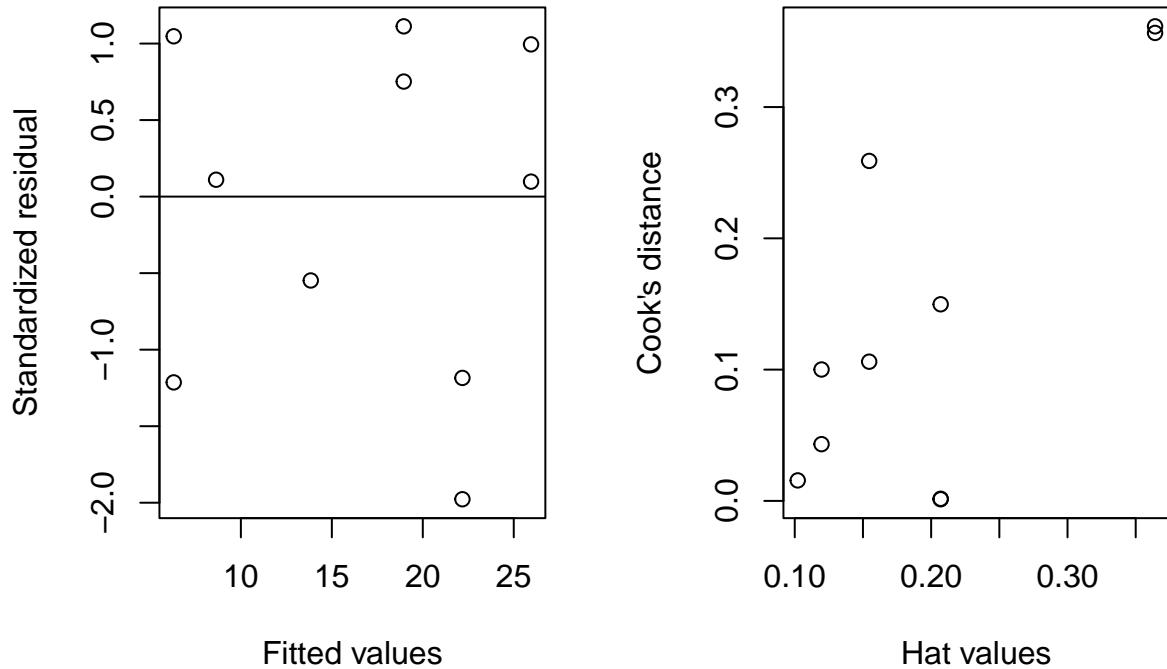


```
par(mfrow=c(1,2))

r = rstandard(driftglm)
lf <- fitted(driftglm)

plot(lf, r,xlab="Fitted values",ylab="Standardized residual")
abline(h=0)

h <- hatvalues(driftglm)
cd <- cooks.distance(driftglm)
plot(h,cd,xlab="Hat values",ylab="Cook's distance")
```

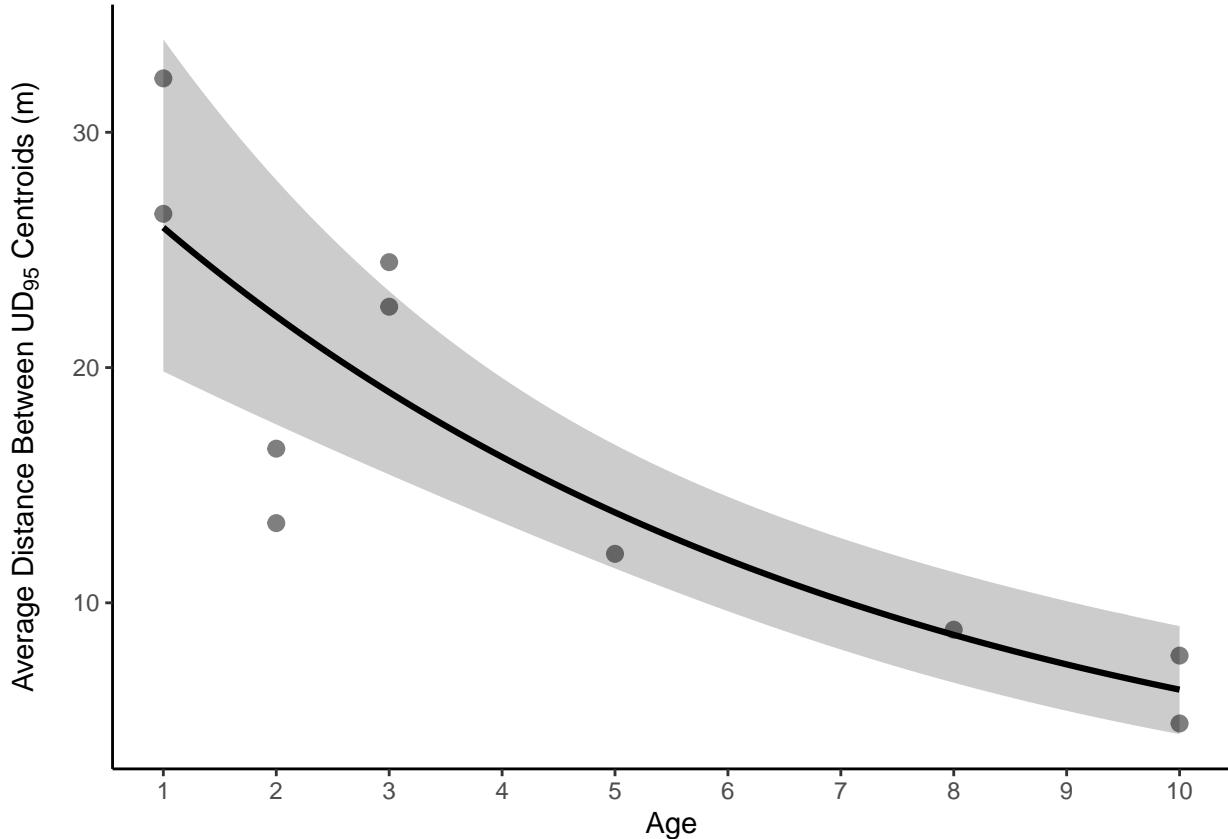


```
par(mfrow=c(1,1))
```

Plotting the model predictions.

Caption from manuscript: Average distance between centroids of consecutive UD95 snapshots from the sweeping window approach to quantify space use variability, which is plotted as a function of age. The increments between space use estimations were approximately one day. Larger values indicate that the UD95 centroid changed position more than for smaller values, suggesting positional changes of space use. Line estimated from generalised linear model using Gamma distribution with logarithmic link, which was fitted with age as a predictor. The ribbon is the 95% confidence interval. Pseudo-R2 based on likelihood ratio = 0.82.

```
effect_plot(driftglm,
            pred = age,
            plot.points = T,
            interval = T,
            point.size = 2.5,
            point.alpha = 0.5) +
scale_x_continuous(breaks = c(1:10)) +
labs(x = "Age",
     y = expression("Average Distance Between UD"[95]*" Centroids (m)")) +
theme_classic() +
theme(axis.title.y = element_text(margin = margin(r = 10)))
```



```
# ggsave("ave_drift_glm_gamma_log_240_6_20221110.tiff",
#       path = here(paste("OneDrive - Queensland University of Technology/",
#                         "MSc - Scott Forrest/DATA/Modelling/dBBMM/Graphical outputs",
#                         sep = "")),
#       width = 180, height = 100, units = "mm", dpi = 300)
```

References

- Fleming, C H, W F Fagan, T Mueller, K A Olson, P Leimgruber, and J M Calabrese. 2016. “Estimating Where and How Animals Travel: An Optimal Framework for Path Reconstruction from Autocorrelated Tracking Data.” *Ecology* 97 (3): 576–82. <https://doi.org/10.1890/15-1607>.
- Forrest, S W, M R Recio, and P J Seddon. 2022. “Moving Wildlife Tracking Forward Under Forested Conditions with the SWIFT GPS Algorithm.” *Animal Biotelemetry* 2022 10:1 10 (1): 1–11. <https://doi.org/10.1186/S40317-022-00289-9>.
- Kranstauber, Bart, Gabriella E C Gall, Tim Vink, Tim Clutton-Brock, and Marta B Manser. 2020. “Long-Term Movements and Home-Range Changes: Rapid Territory Shifts in Meerkats.” Edited by Jason Chapman. *The Journal of Animal Ecology* 89 (3): 772–83. <https://doi.org/10.1111/1365-2656.13129>.
- Kranstauber, Bart, Roland Kays, Scott D Lapoint, Martin Wikelski, and Kamran Safi. 2012. “A Dynamic Brownian Bridge Movement Model to Estimate Utilization Distributions for Heterogeneous Animal Movement.” *The Journal of Animal Ecology* 81 (4): 738–46. <https://doi.org/10.1111/j.1365-2656.2012.01955.x>.
- Signer, Johannes, John Fieberg, and Tal Avgar. 2019. “Animal Movement Tools (Amt): R Package for Managing Tracking Data and Conducting Habitat Selection Analyses.” *Ecology and Evolution* 9 (2): 880–90. <https://doi.org/10.1002/ece3.4823>.

Session info

```
sessionInfo()

## R version 4.2.1 (2022-06-23 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19044)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_New Zealand.utf8  LC_CTYPE=English_New Zealand.utf8
## [3] LC_MONETARY=English_New Zealand.utf8 LC_NUMERIC=C
## [5] LC_TIME=English_New Zealand.utf8
##
## attached base packages:
## [1] stats      graphics   grDevices utils      datasets   methods    base
##
## other attached packages:
## [1] transformr_0.1.4 gifski_1.6.6-1   gganimate_1.0.8  broom_1.0.1
## [5] MASS_7.3-57       heplots_1.3-9    car_3.1-0      carData_3.0-5
## [9] scales_1.2.1     lubridate_1.8.0  furrr_0.3.1    future_1.28.0
## [13] statmod_1.4.37   MuMIN_1.47.1   jtools_2.2.0   patchwork_1.1.2
## [17] ggspatial_1.1.6  ggpubr_0.4.0   sf_1.0-8      here_1.0.1
## [21] lattice_0.20-45 move_4.1.8     rgdal_1.5-32   raster_3.5-29
## [25] sp_1.5-0        geosphere_1.5-14 forcats_0.5.2  stringr_1.4.1
## [29] dplyr_1.0.9     purrrr_0.3.4   readr_2.1.2   tidyverse_1.2.0
## [33] tibble_3.1.8    ggplot2_3.3.6  tidyverse_1.3.2
##
## loaded via a namespace (and not attached):
## [1] googledrive_2.0.0  colorspace_2.0-3   ggsignif_0.6.3
## [4] ellipsis_0.3.2    class_7.3-20     rprojroot_2.0.3
## [7] fs_1.5.2          rstudioapi_0.14  proxy_0.4-27
## [10] farver_2.1.1     listenv_0.8.0   fansi_1.0.3
## [13] xml2_1.3.3       codetools_0.2-18  cachem_1.0.6
## [16] knitr_1.40        jsonlite_1.8.0  dbplyr_2.2.1
## [19] compiler_4.2.1   httr_1.4.4     backports_1.4.1
## [22] assertthat_0.2.1 Matrix_1.4-1   fastmap_1.1.0
## [25] gargle_1.2.0     cli_3.3.0     s2_1.1.0
## [28] tweenr_2.0.2     prettyunits_1.1.1 htmltools_0.5.3
## [31] tools_4.2.1      gtable_0.3.1   glue_1.6.2
## [34] wk_0.6.0         Rcpp_1.0.9    cellranger_1.1.0
## [37] vctrs_0.4.1      nlme_3.1-157  xfun_0.32
## [40] globals_0.16.1   rvest_1.0.3   lpSolve_5.6.16
## [43] lifecycle_1.0.3   rstatix_0.7.0  googlesheets4_1.0.1
## [46] terra_1.6-7      hms_1.1.2     parallel_4.2.1
## [49] yaml_2.3.5       memoise_2.0.1  pander_0.6.5
## [52] stringi_1.7.8   highr_0.9    e1071_1.7-11
## [55] rlang_1.0.6      pkgconfig_2.0.3 evaluate_0.16
## [58] labeling_0.4.2   tidyselect_1.1.2 parallelly_1.32.1
## [61] magrittr_2.0.3   R6_2.5.1     generics_0.1.3
## [64] DBI_1.1.3       pillar_1.8.1  haven_2.5.1
## [67] withr_2.5.0     units_0.8-0  abind_1.4-5
```

```
## [70] modelr_0.1.9      crayon_1.5.1       KernSmooth_2.23-20
## [73] utf8_1.2.2        tzdb_0.3.0        rmarkdown_2.16
## [76] progress_1.2.2     grid_4.2.1        readxl_1.4.1
## [79] reprex_2.0.2       digest_0.6.29    classInt_0.4-7
## [82] stats4_4.2.1       munsell_0.5.0    viridisLite_0.4.1
```