Home range analysis for kākā at Orokonui Ecosanctuary

Scott Forrest

2022-11-17

Load packages and set working directory.

```
library(tidyverse)

#devtools::install_github("paleolimbot/ggspatial")

packages <- c("move", "lattice", "purrr", "here", "raster", "sf", "ggpubr",
              "ggspatial", "patchwork", "jtools", "MuMIn", "statmod")
walk(packages, require, character.only = T)

here::i_am(paste("./OneDrive - Queensland University of Technology/",
                 "MSc - Scott Forrest/DATA/Modelling/dBBMM/",
                 "Kaka home range analysis - Forrest, Recio and Seddon 20221110.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 an extent raster to be used for each individual. It is important to keep these the same size for the space use variability (SUV) incremental analysis, so it is important that the raster covers the extent of each individual's locations.

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

Warnings may arise in the following step about the datum - which can typically be ignored. Here they have been hidden them from the output.

The location error is the median error from Forrest, Recio, and Seddon (2022) who tested these particular devices.

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

for (i in 1:10) {

  all_tags_list[[i]]$DateTime <- as.POSIXct(all_tags_list[[i]]$DateTime, tz = "GMT")
  all_tags_dbbmm[[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")) %>%
    brownian.bridge.dyn(., raster = ext_raster, location.error = 8.35, margin = 3,
                        window.size = 15, time.step = 12)

}
```

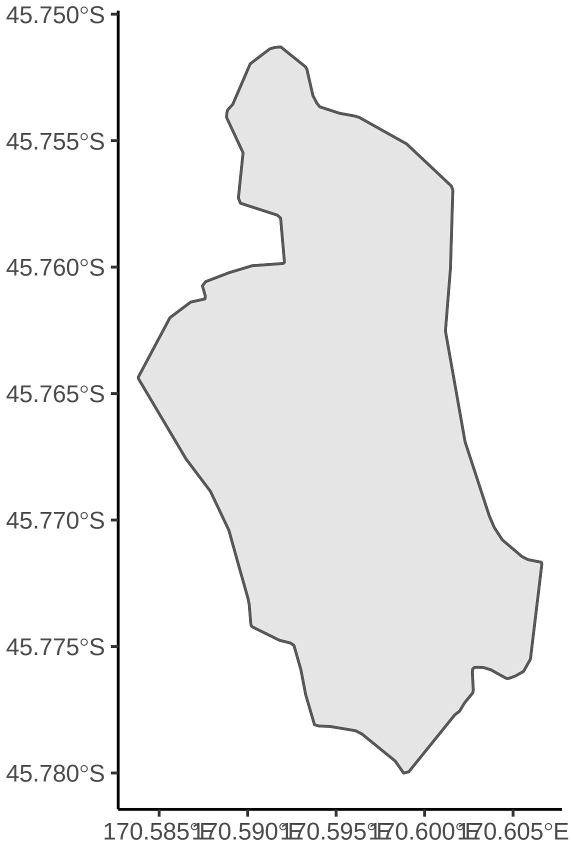
```
## Computational size: 1.2e+10
## Computational size: 1.2e+10

## Computational size: 1.4e+10
## Computational size: 1.4e+10

## Computational size: 8.8e+09

## Computational size: 9.4e+09

## Computational size: 1.4e+10

## Computational size: 1.3e+10
## Computational size: 1.3e+10

## Computational size: 1.2e+10
```

Import a spatial object of the Orokonui Ecosanctuary fence.

```r
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, Mappin
##   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
```

```
# check map
ggplot(OrokonuiFence) +
  geom_sf() +
  theme_classic()
```



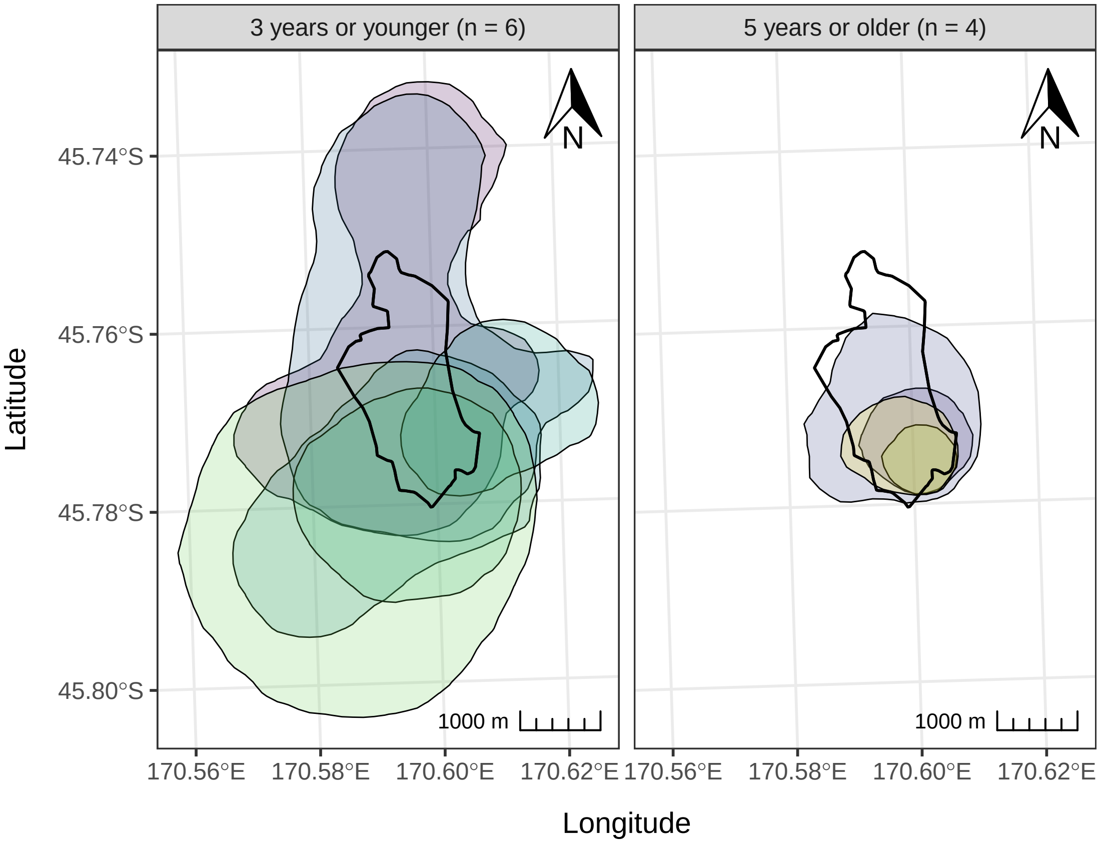
Extract contours for further analysis.

```
all_tags_contours_95 <- map(all_tags_dbbmm, raster2contour, levels = 0.95)

all_contours_95 <- vector(mode = "list", length = 10)

for(i in 1:10) {
all_tags_contours_95[[i]]@data$id <- rownames(all_tags_contours_95[[i]]@data)
all_contours_95[[i]] <- fortify(all_tags_contours_95[[i]], region = "id")
}

id <- 45505:45514
age <- c(1, 10, 5, 1, 3, 2, 2, 3, 10, 8)

all_contours_95_df <- bind_rows(all_contours_95) %>%
  mutate(id = rep(05:14, sapply(all_contours_95, nrow)),
         age = rep(age, sapply(all_contours_95, nrow)))

young <- all_contours_95_df[which(all_contours_95_df$age < 4),] %>%
  mutate(age_group = "3 years or younger (n = 6)")
```

```r
old <- all_contours_95_df[which(all_contours_95_df$age > 4),] %>%
  mutate(age_group = "5 years or older (n = 4)")

all_contours_95_df <- rbind(young, old)
```

Plot.

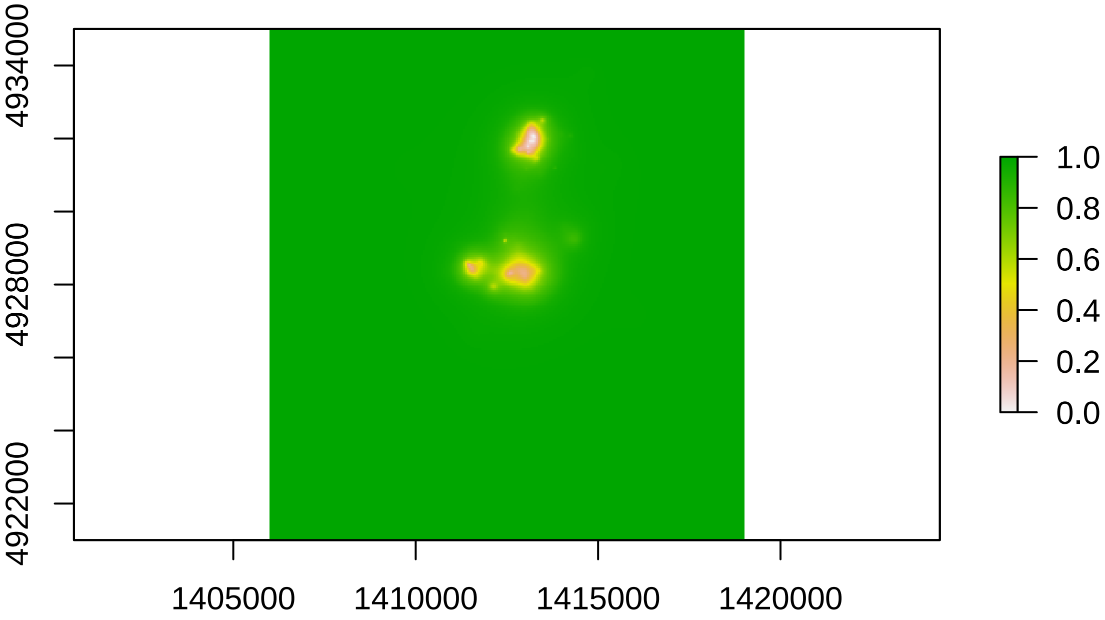
```r
ggplot() +
  geom_polygon(data = all_contours_95_df,
               aes(x = long, y = lat, fill = factor(id)),
               alpha = 0.2,
               colour = "black",
               size = 0.25) +
  geom_sf(data = OrokonuiFence, colour = "black", fill = NA) +
  coord_sf() +
  scale_y_continuous("Latitude", breaks = seq(-45.74, -45.80, by = -0.02)) +
  scale_x_continuous("Longitude", breaks = seq(170.56, 170.62, by = 0.02)) +
  scale_fill_viridis_d(name = "ID") +
  facet_wrap(vars(age_group)) +
  theme_bw() +
  theme(legend.position = "none",
        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))) +
  ggspatial::annotation_north_arrow(location = "tr",
                                    which_north = "true",
                                    height = unit(1, "cm"),
                                    width = unit(.75, "cm")) +
  ggspatial::annotation_scale(location = "br",
                              style = "ticks",
                              bar_cols = c("grey60", "white"),
                              height = unit(0.25, "cm"))
```

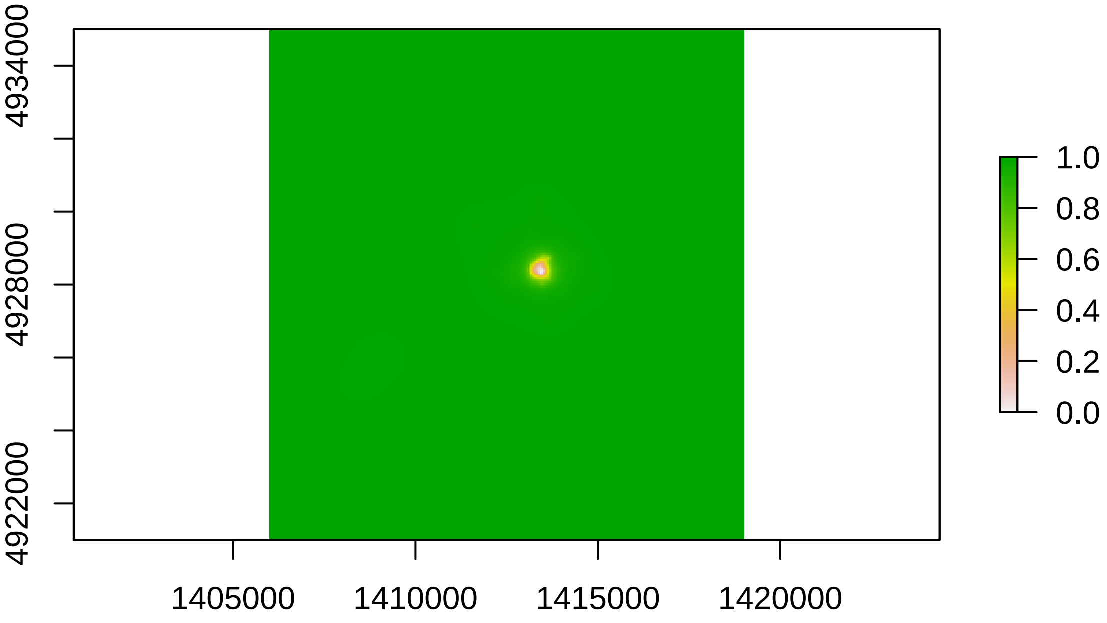


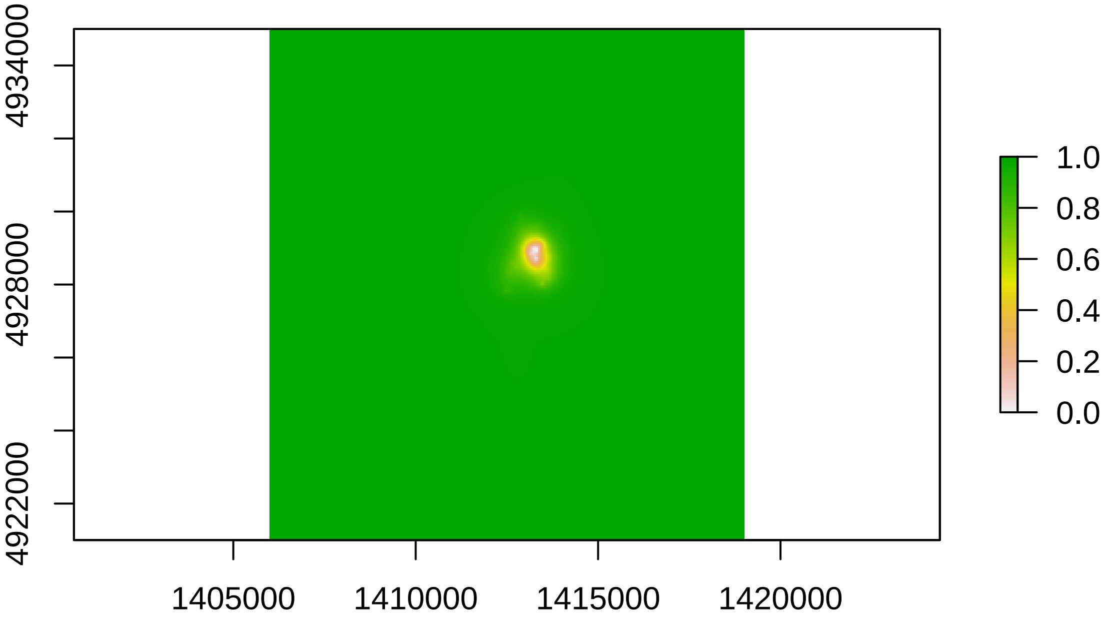
```
# ggsave(filename = here("OneDrive - Queensland University of Technology/
#                         MSc - Scott Forrest/DATA/Modelling/dBBMM/Graphical outputs/
#                         all_home_ranges_young_old_20221108.tiff"),
#        width = 160,
#        height = 125,
#        units = "mm",
#        dpi = 300)
```

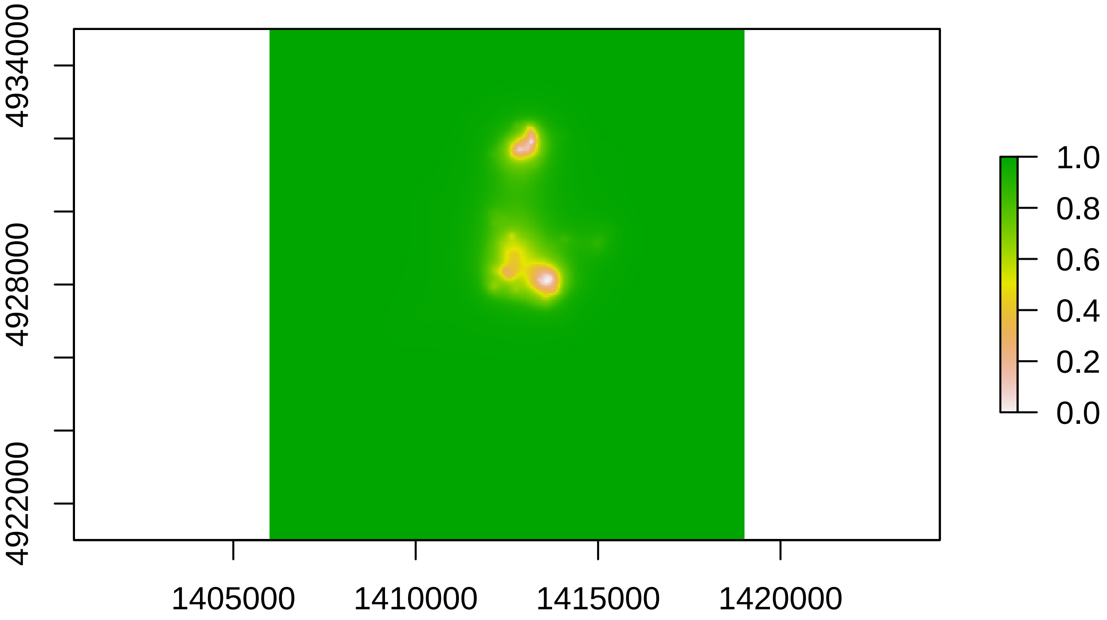
Convert to utilisation distributions (continuous probability surfaces).

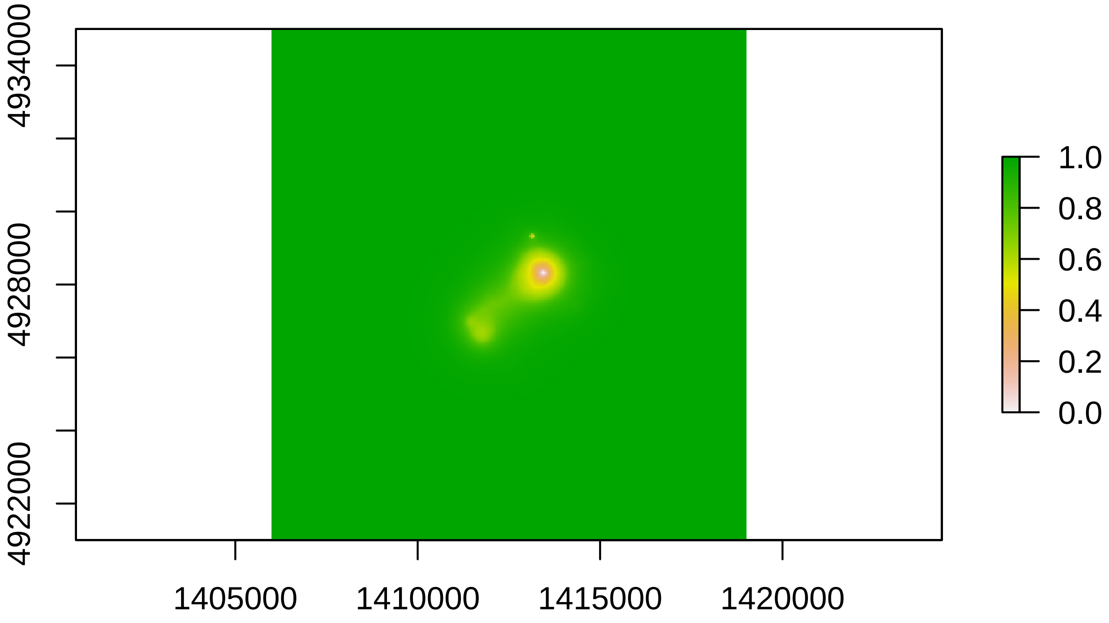
```
all_tags_uds <- map(all_tags_dbbmm, getVolumeUD)

# create an extent to trim away some low probability
bbox_trim <- st_bbox(c(xmin = 1406000, xmax = 1419000, ymin = 4921000, ymax = 4935000),
                     crs = 2193)

trimmed_uds <- map(all_tags_uds, crop, bbox_trim)
map(trimmed_uds, plot)
```

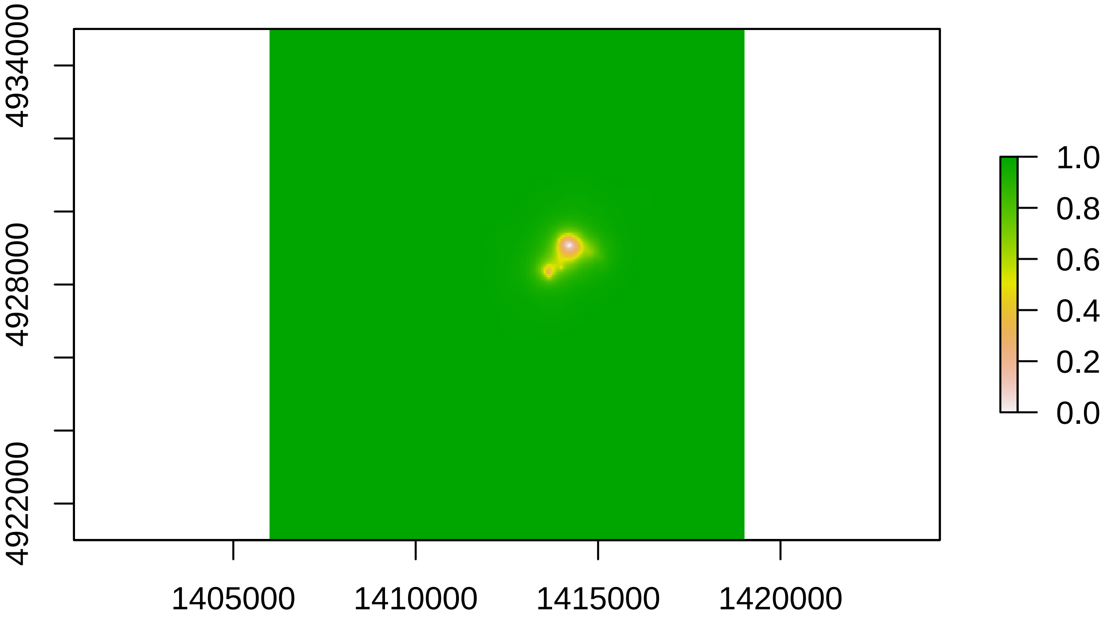

4934000
4928000
4922000
1405000
1410000
1415000
1420000
1.0
0.8
0.6
0.4
0.2
0.0

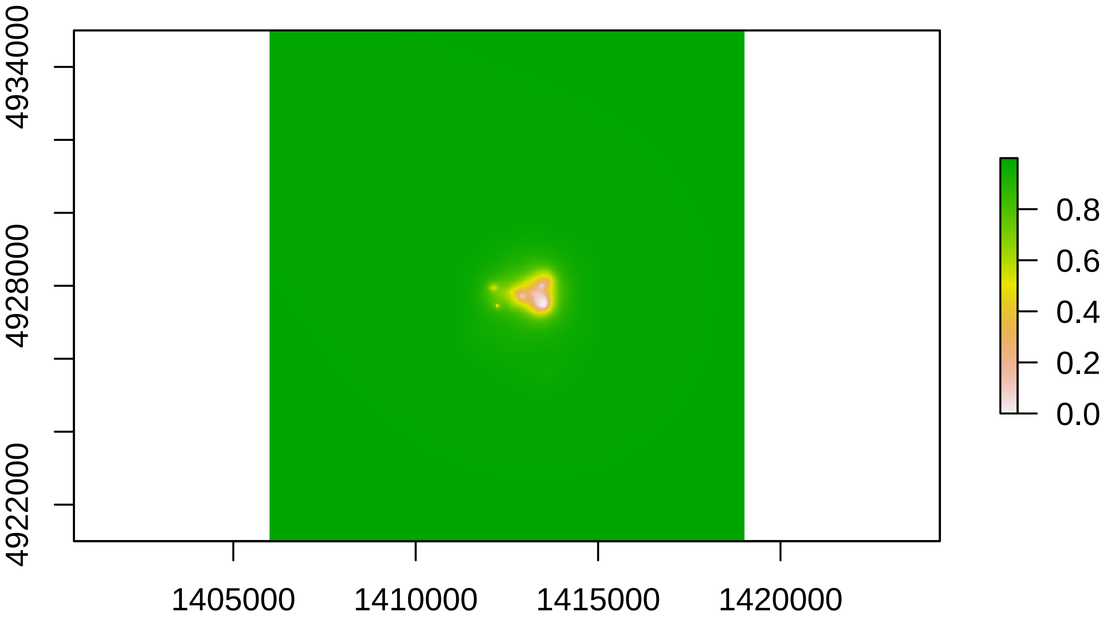

4934000
4928000
4922000
1405000
1410000
1415000
1420000
1.0
0.8
0.6
0.4
0.2
0.0

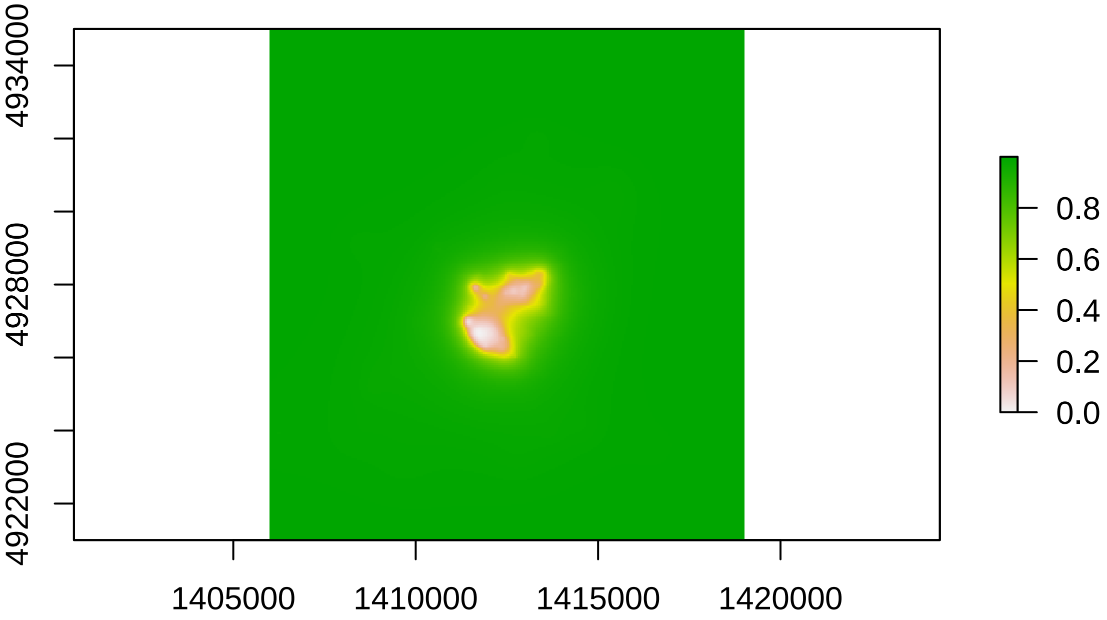

4934000
4928000
4922000
1405000
1410000
1415000
1420000
1.0
0.8
0.6
0.4
0.2
0.0

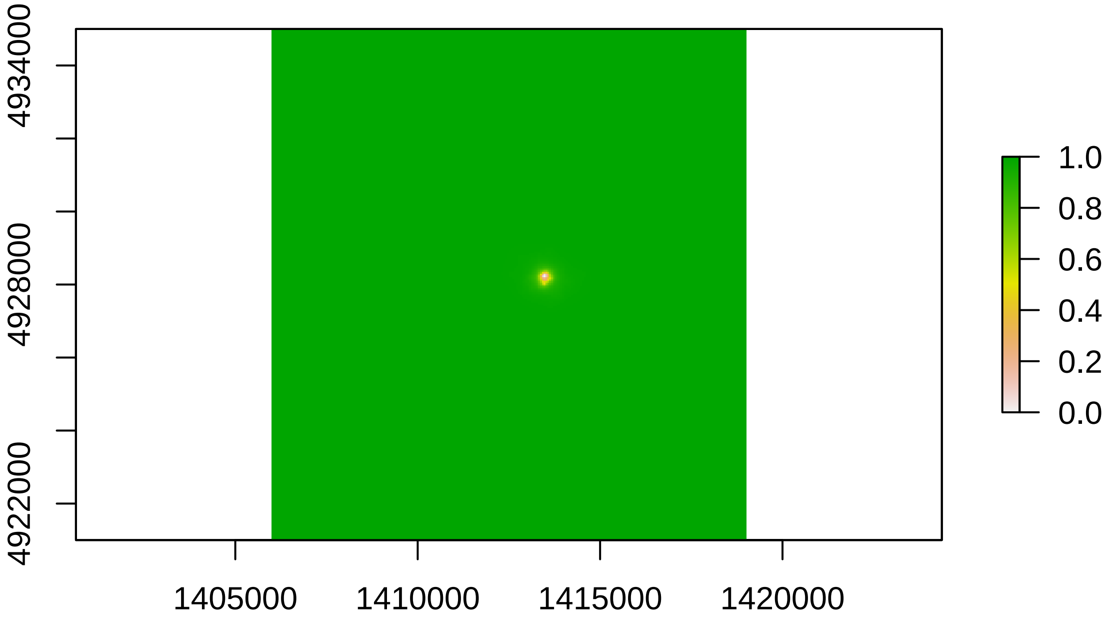

4934000
4928000
4922000
1405000
1410000
1415000
1420000
1.0
0.8
0.6
0.4
0.2
0.0

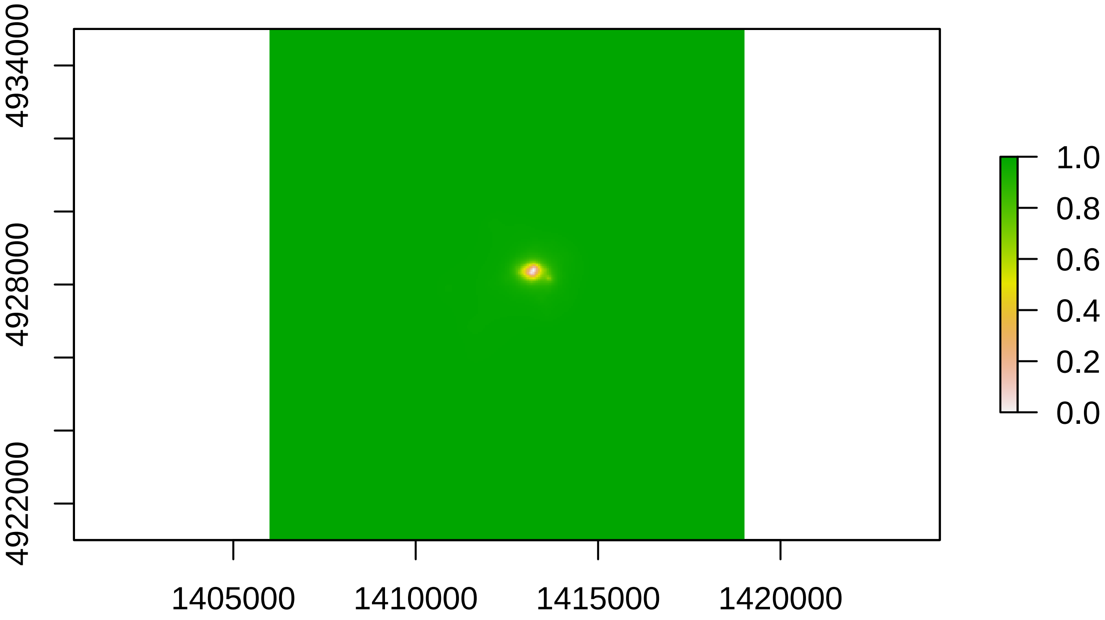

4934000
4928000
4922000
1405000
1410000
1415000
1420000
1.0
0.8
0.6
0.4
0.2
0.0


4934000
4928000
4922000
1405000
1410000
1415000
1420000
1.0
0.8
0.6
0.4
0.2
0.0


4934000
4928000
4922000
1405000
1410000
1415000
1420000
0.8
0.6
0.4
0.2
0.0


4934000
4928000
4922000
1405000
1410000
1415000
1420000
0.8
0.6
0.4
0.2
0.0


4934000
4928000
4922000
1405000
1410000
1415000
1420000
1.0
0.8
0.6
0.4
0.2
0.0



```
## [[1]]
## NULL
##
## [[2]]
## NULL
##
## [[3]]
## NULL
##
## [[4]]
## NULL
##
## [[5]]
## NULL
##
## [[6]]
## NULL
##
## [[7]]
## NULL
##
## [[8]]
## NULL
##
## [[9]]
## NULL
```

```
##
## [[10]]
## NULL
```

Calculating the area contained within UD isopleths, which will be in m^2.

```
UD50_areas <- vector(mode = "numeric", length = 10)
UD95_areas <- vector(mode = "numeric", length = 10)

for(i in 1:10) {

  UD50_areas[[i]] <- all_tags_dbbmm[[i]] %>% raster2contour(levels = 0.5) %>%
    st_as_sf() %>% st_polygonize() %>% st_area()

  UD95_areas[[i]] <- all_tags_dbbmm[[i]] %>% raster2contour(levels = 0.95) %>%
    st_as_sf() %>% st_polygonize() %>% st_area()

}

UD50_areas
```

```
##  [1]  963379.69  173496.56  407912.25 1434784.92  422480.46  373071.00
##  [7]  763871.09 2972991.17   57841.17  127631.59
```

```
UD95_areas
```

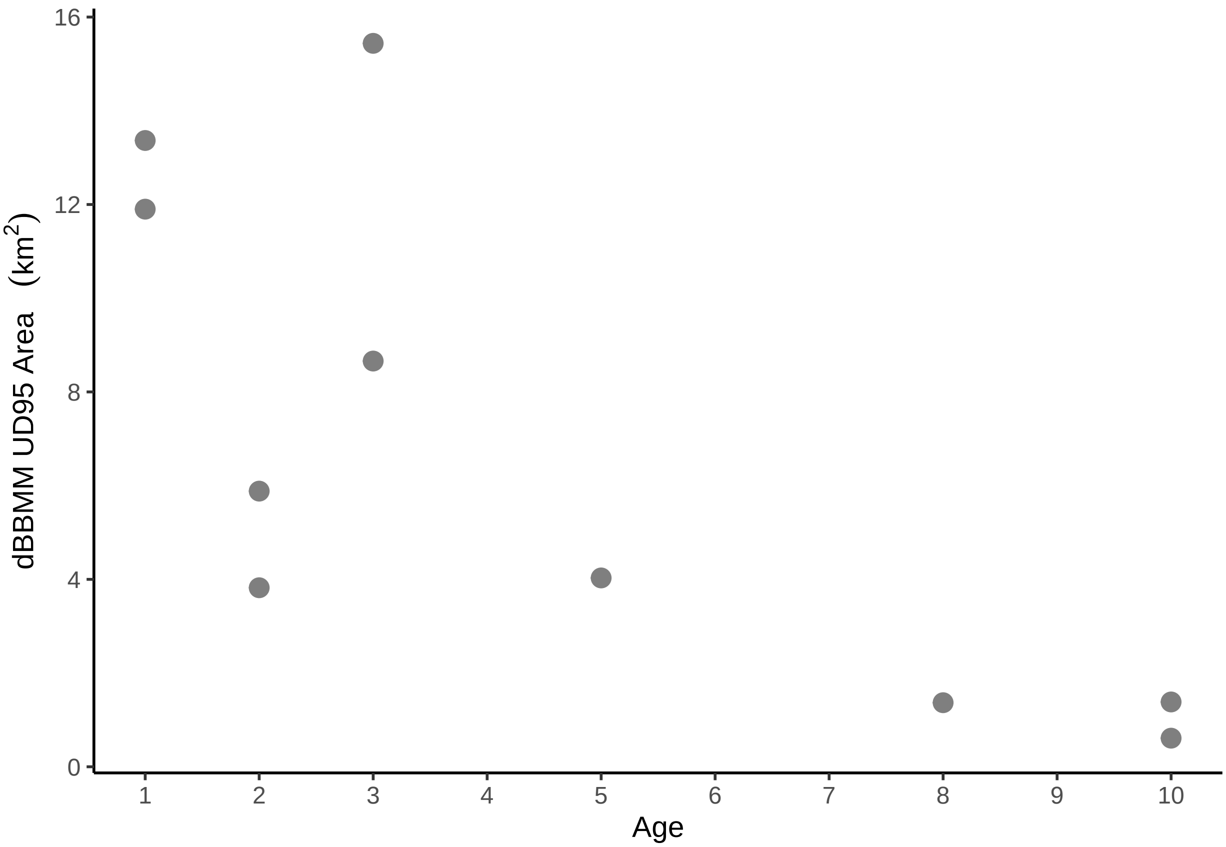
```
##  [1] 11900491.4  1383199.4  4030223.6 13366247.4  8659034.8  3820982.4
##  [7]  5882845.6 15439282.8   610797.9  1367768.4
```

Add in individual-level covariates and create a data frame.

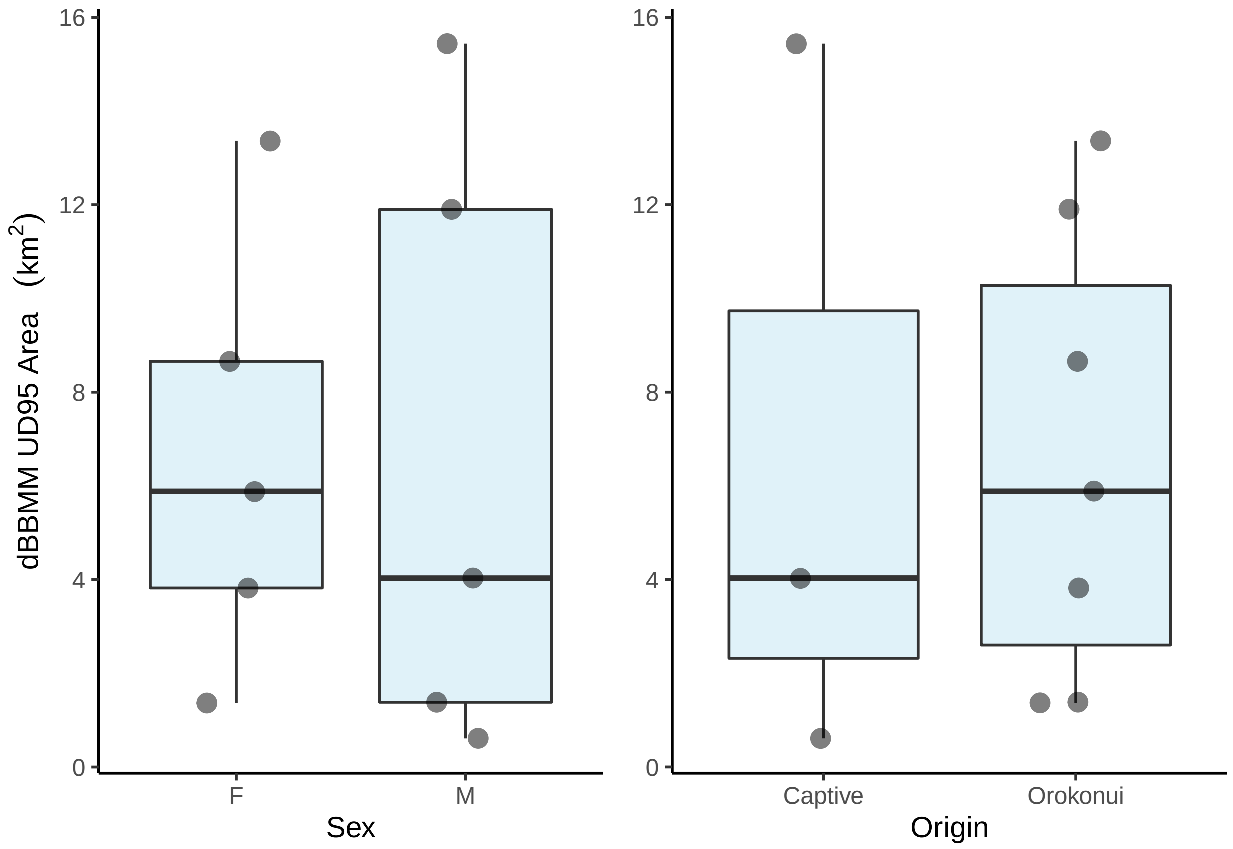
```
id <- 45505:45514
sex <- c("M", "M", "M", "F", "F", "F", "F", "M", "M", "F")
age <- c(1, 10, 5, 1, 3, 2, 2, 3, 10, 8)
origin <- c("Orokonui", "Orokonui", "Captive", "Orokonui", "Orokonui", "Orokonui",
            "Orokonui",  "Captive", "Captive", "Orokonui")

all_tags_summary <- data.frame(id, sex, age, origin, UD50_areas, UD95_areas)

dBBMM <- all_tags_summary %>% mutate(Sex = factor(sex),
                                     Origin = factor(origin),
                                     Age = age,
                                     UD50km2 = UD50_areas/1000000,
                                     UD95km2 = UD95_areas/1000000)
```

Plot.

```
dBBMM %>% ggplot(aes(Age, UD95km2)) +
  geom_point(alpha = 0.5, size = 3) +
  scale_x_continuous(breaks = c(1:10)) +
  labs(y = expression(paste("dBBMM UD95 Area   ", (km^2)))) +
  theme_classic()
```



```
sexboxplot <- dBBMM %>% ggplot(aes(Sex, UD95km2)) +
  geom_boxplot(alpha = 0.25, fill = "skyblue") +
  geom_jitter(width = 0.15, size = 3, alpha = 0.5) +
  labs(y = expression(paste("dBBMM UD95 Area   ", (km^2)))) +
  theme_classic()

originboxplot <- dBBMM %>% ggplot(aes(Origin, UD95km2)) +
  geom_boxplot(alpha = 0.25, fill = "skyblue") +
  geom_jitter(width = 0.15, size = 3, alpha = 0.5) +
  theme_classic()

ggarrange(sexboxplot, originboxplot + rremove("ylab"))
```



Statistical analysis of home range area

```r
area_age_glm <- glm(UD95km2 ~ age, data = dBBMM, family = Gamma(link = "log"))

summary(area_age_glm)
```

```
##
## Call:
## glm(formula = UD95km2 ~ age, family = Gamma(link = "log"), data = dBBMM)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.80980  -0.39729  -0.05477   0.13750   0.85191
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.84677    0.27275  10.437 6.16e-06 ***
## age         -0.28538    0.04844  -5.891 0.000365 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for Gamma family taken to be 0.2687004)
##
##     Null deviance: 8.6490  on 9  degrees of freedom
## Residual deviance: 2.0193  on 8  degrees of freedom
## AIC: 47.948
##
## Number of Fisher Scoring iterations: 5
```

```
AIC(area_age_glm) # for if including other variables
```

```
## [1] 47.94792
```

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

```
## Analysis of Deviance Table
##
## Model: Gamma, link: log
##
## Response: UD95km2
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev      F   Pr(>F)
## NULL                     9     8.6490
## age   1   6.6297         8     2.0193 24.673 0.001097 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
r.squaredLR(area_age_glm)
```

```
## [1] 0.7903232
## attr(,"adj.r.squared")
## [1] 0.792829
```

```
confint(area_age_glm)
```

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

```
##                  2.5 %     97.5 %
## (Intercept)  2.3284201  3.4224263
## age         -0.3801093 -0.1830295
```

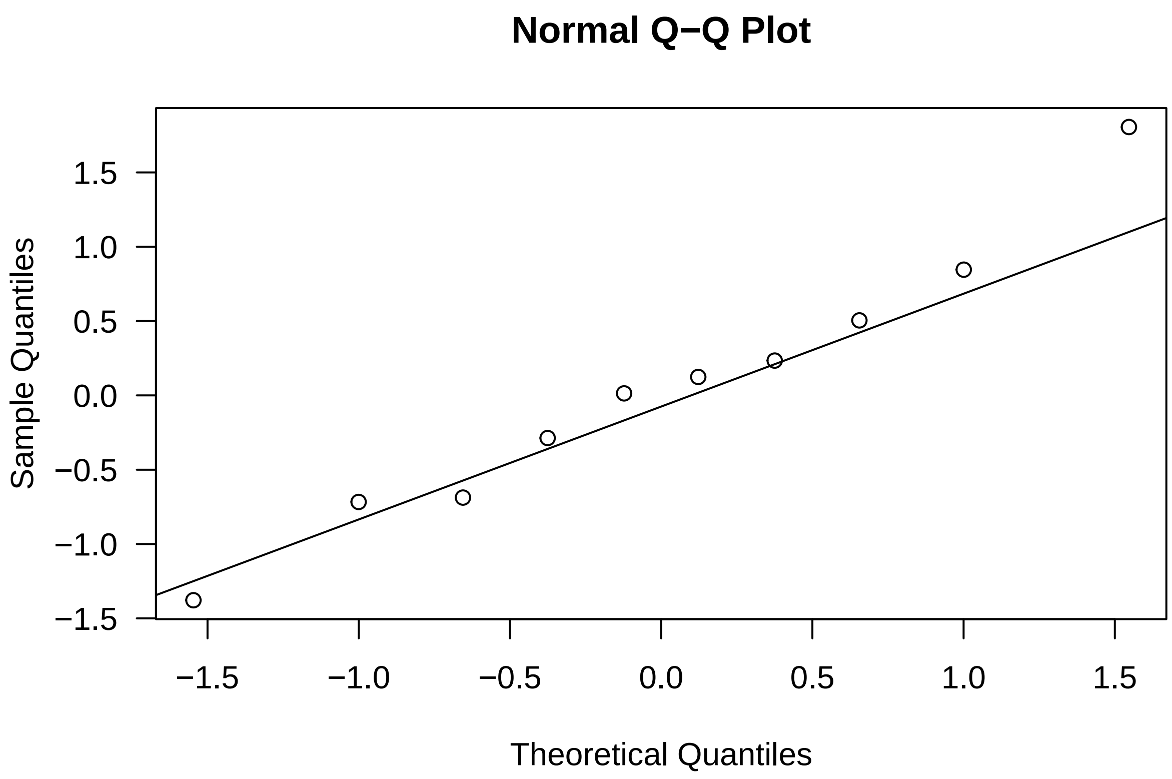
```
dfun <- function(object) {
  with(object,sum((weights * residuals^2)[weights > 0])/df.residual)
}

dfun(area_age_glm)
```

```
## [1] 0.2687004
```

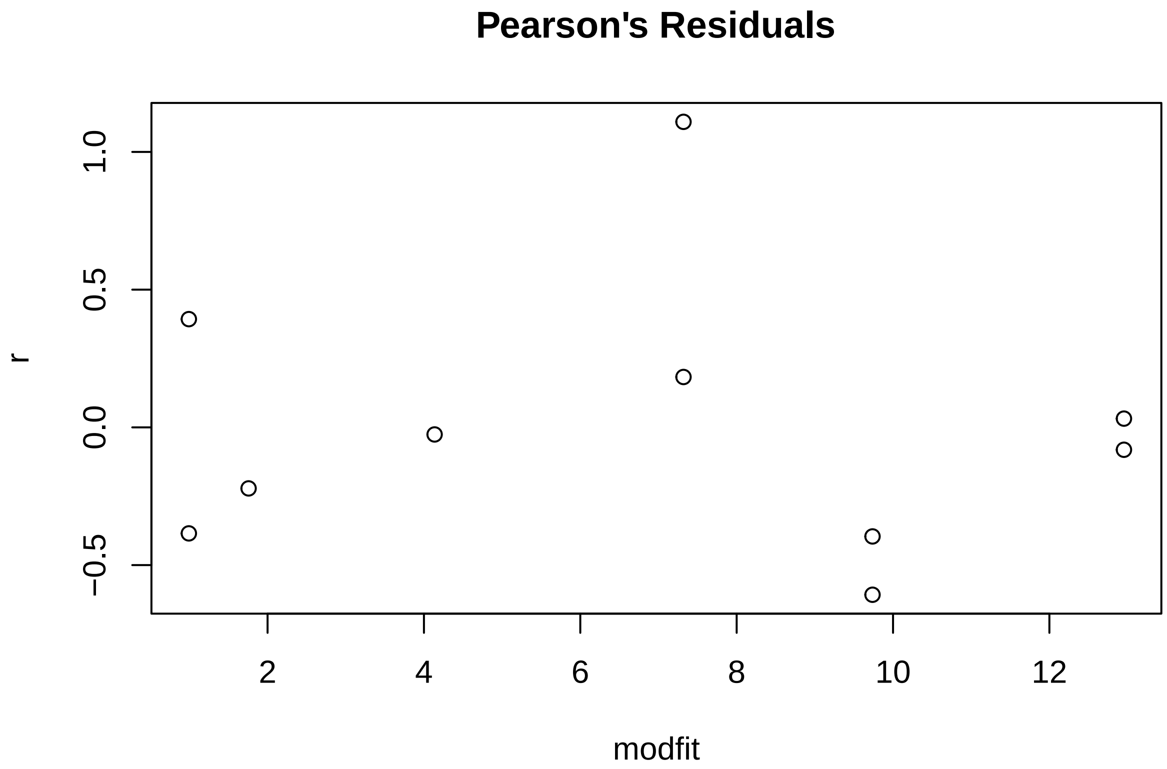
```
pseudoR2 <- 1 - (area_age_glm$deviance / area_age_glm$null.deviance)
pseudoR2
```

```
## [1] 0.7665318
```

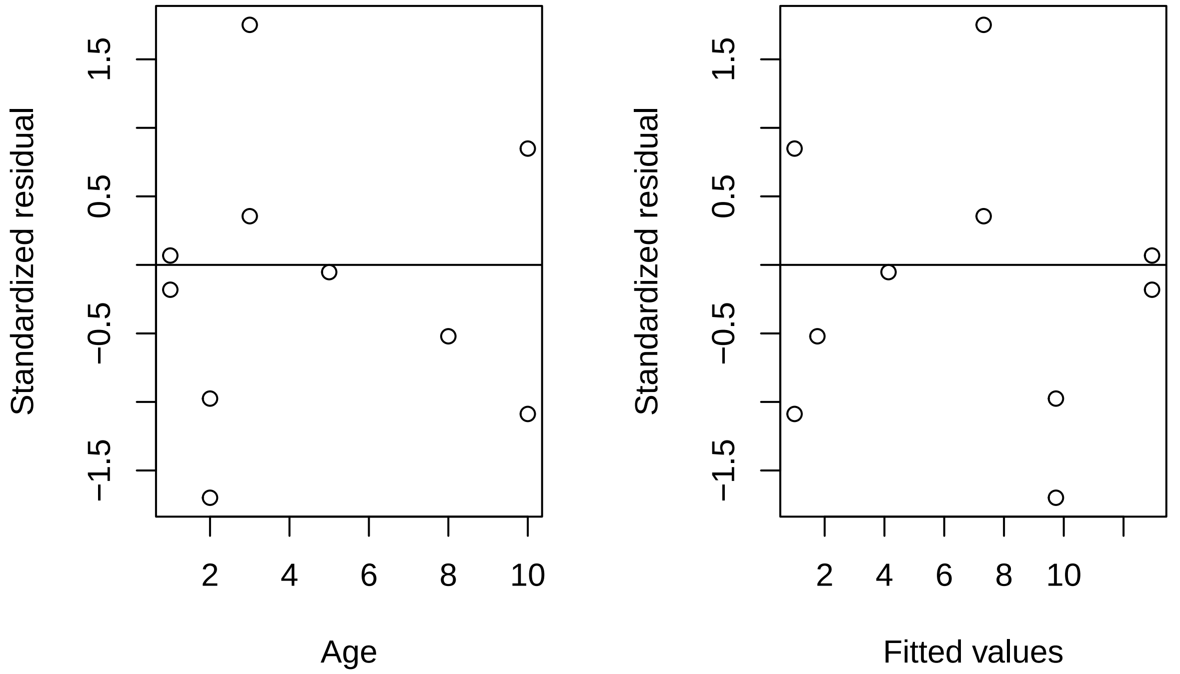
```
qr.area_age_glm <- qresid(area_age_glm)
qqnorm(qr.area_age_glm, las = 1)
qqline(qr.area_age_glm)
```



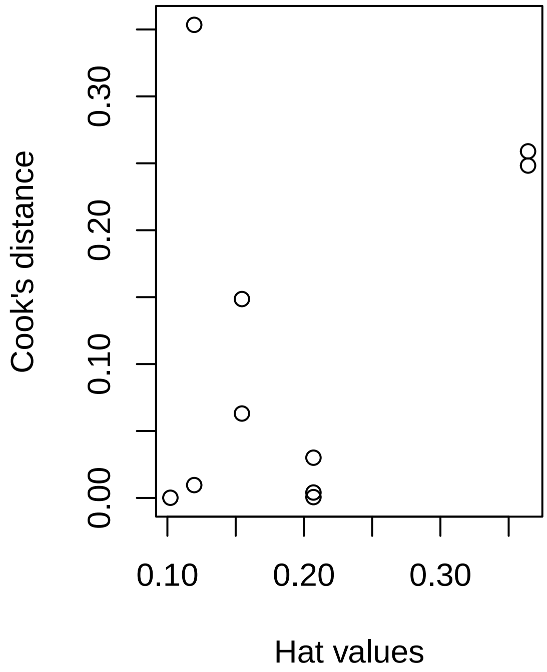
```
r<-residuals(area_age_glm,type="pearson")
modfit<-fitted(area_age_glm)

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



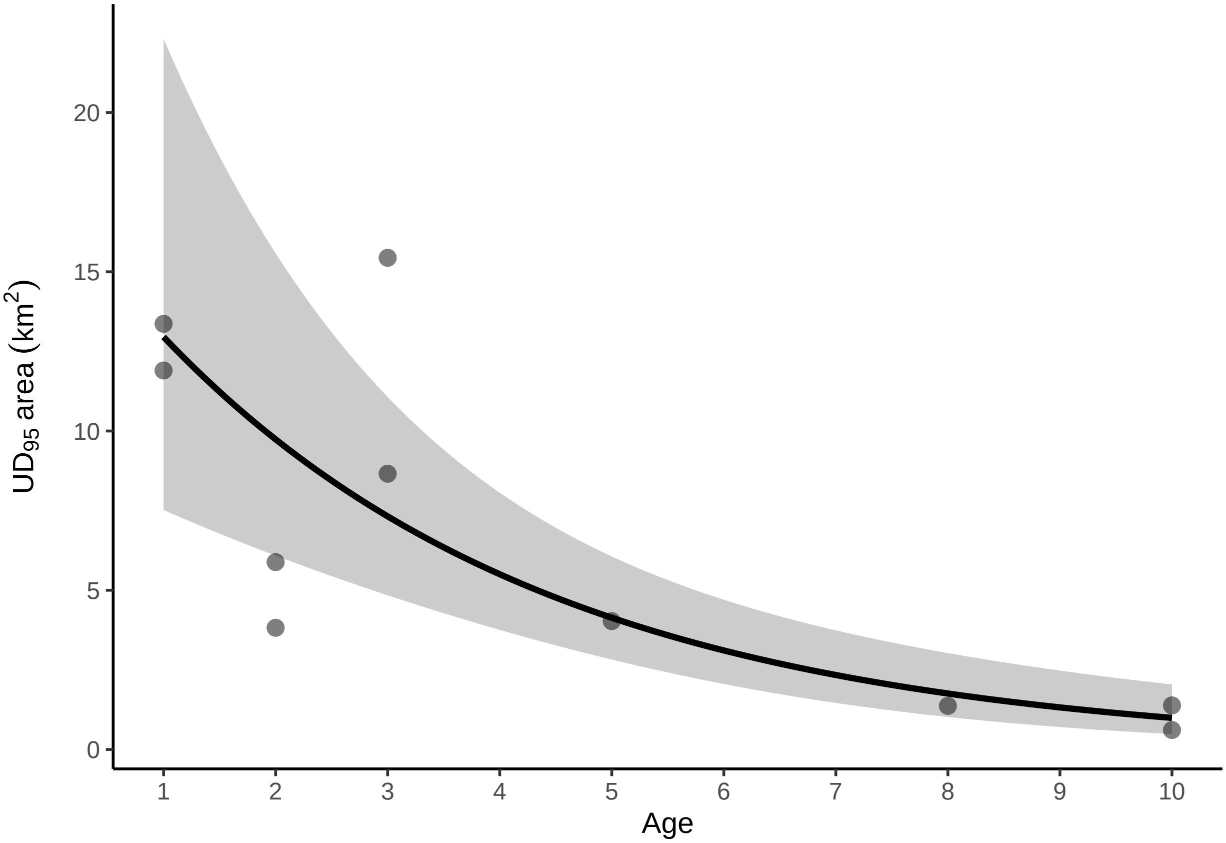
```
par(mfrow=c(1,2)) # change plot window to accommodate 2 plots

r = rstandard(area_age_glm)
lf <- fitted(area_age_glm)
plot(dBBMM$Age, r,xlab="Age",
     ylab="Standardized residual")
abline(h=0)

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



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

par(mfrow=c(1,1)) # return to single plotting
```



```
area_age_glm_plot <- effect_plot(area_age_glm,
                                 pred = age,
                                 plot.points = T,
                                 interval = T,
                                 point.size = 2.5,
                                 point.alpha = 0.5) +
  scale_x_continuous("Age", breaks = c(1:10)) +
  ylab(expression(UD[95]~area~(km^2))) +
  theme_classic() +
  theme(axis.title.y = element_text(margin = margin(r = 10)))

area_age_glm_plot
```



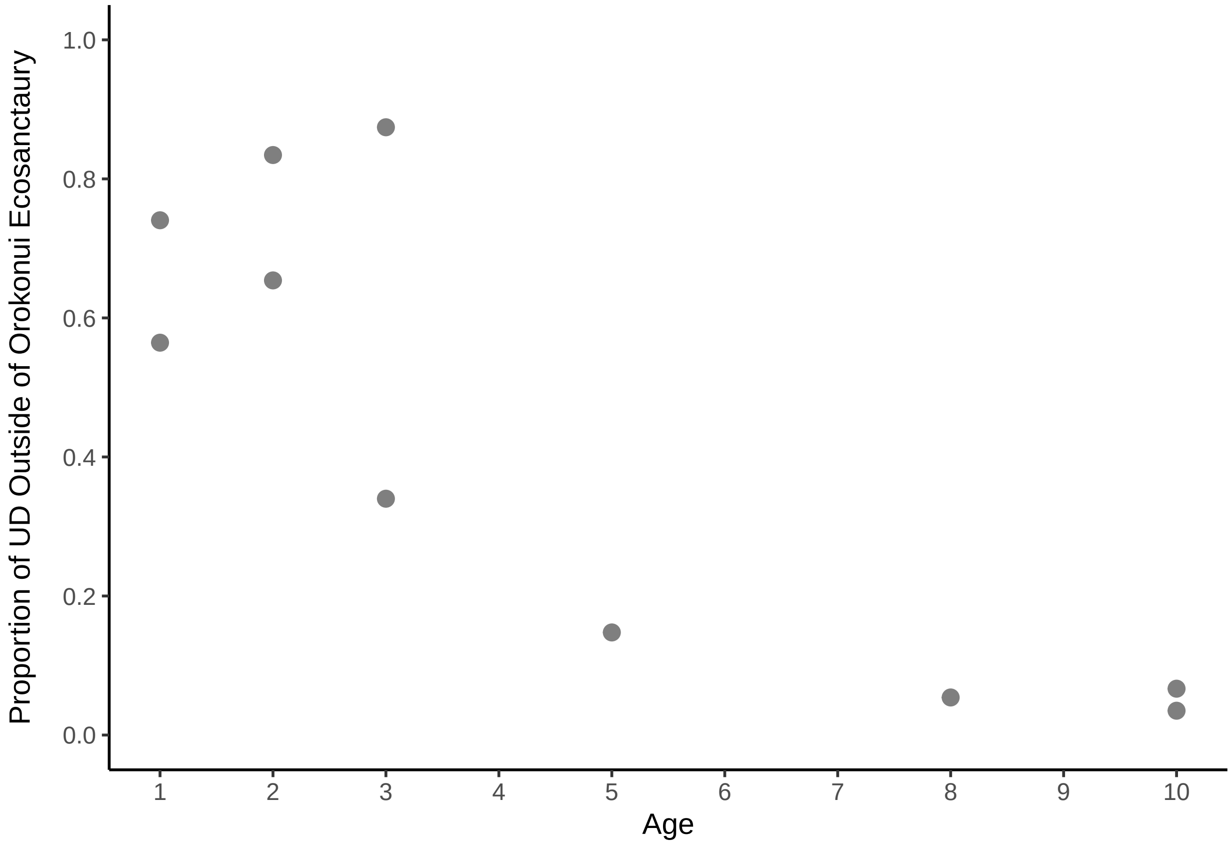
For checking the overlap between the fence of Orokonui and each UD. This calculation is a summation of the values inside the cells (i.e. the probability), rather than the number of cells, which therefore represents the time spent in that area.

```
inside_overlap <- vector(mode = "numeric", length = 10)

for (i in 1:10){
inside_overlap[[i]] <- cellStats(mask(all_tags_dbbmm[[i]], OrokonuiFence), sum)
}

all_tags_summary <- all_tags_summary %>%
  mutate(inside = inside_overlap, outside = 1 - inside_overlap)

head(all_tags_summary)
```

```
##      id sex age   origin UD50_areas UD95_areas    inside    outside
## 1 45505   M   1 Orokonui   963379.7   11900491 0.2595451 0.74045488
## 2 45506   M  10 Orokonui   173496.6    1383199 0.9333076 0.06669243
## 3 45507   M   5  Captive   407912.2    4030224 0.8524026 0.14759741
## 4 45508   F   1 Orokonui  1434784.9   13366247 0.4356122 0.56438784
## 5 45509   F   3 Orokonui   422480.5    8659035 0.6600401 0.33995987
## 6 45510   F   2 Orokonui   373071.0    3820982 0.1656911 0.83430893
```

Plotting home range area, individual-level covariates and overlap with Orokonui Ecosanctuary

```
all_tags_summary %>% ggplot(aes(x = age, y = outside)) +
  geom_point(size = 2.5, alpha = 0.5) +
  scale_x_continuous(breaks = c(1:10), "Age") +
  scale_y_continuous(breaks = seq(0,1,0.2), limits = c(0,1),
                     "Proportion of UD Outside of Orokonui Ecosanctaury") +
  theme_classic() +
  theme(axis.title.y = element_text(margin = margin(r = 10)))
```



Statistical analysis using the same methodology as above, for the proportion of area that lies outside of the Orokonui Ecosanctuary fence.

```
outsideglm <- glm(outside ~ age,
                  data = all_tags_summary,
                  family = Gamma(link = "log"))

summary(outsideglm)
```

```
##
## Call:
## glm(formula = outside ~ age, family = Gamma(link = "log"), data = all_tags_summary)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.4393  -0.3846  -0.2046   0.2206   0.7292
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.21058    0.24472   0.860    0.415
## age         -0.33177    0.04346  -7.633 6.11e-05 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for Gamma family taken to be 0.2163081)
##
##     Null deviance: 10.3345  on 9  degrees of freedom
## Residual deviance:  1.5161  on 8  degrees of freedom
## AIC: -11.397
##
## Number of Fisher Scoring iterations: 4
```

```
AIC(outsideglm)
```

```
## [1] -11.39746
```

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

```
## Analysis of Deviance Table
##
## Model: Gamma, link: log
##
## Response: outside
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev      F    Pr(>F)
## NULL                     9    10.3345
## age   1   8.8184         8     1.5161 40.768 0.0002125 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
r.squaredLR(outsideglm)
```

```
## [1] 0.8727209
## attr(,"adj.r.squared")
## [1] 3.173168
```

```
confint(outsideglm)
```

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

```
##                  2.5 %     97.5 %
## (Intercept) -0.2470777  0.7136652
## age         -0.4141895 -0.2431403
```

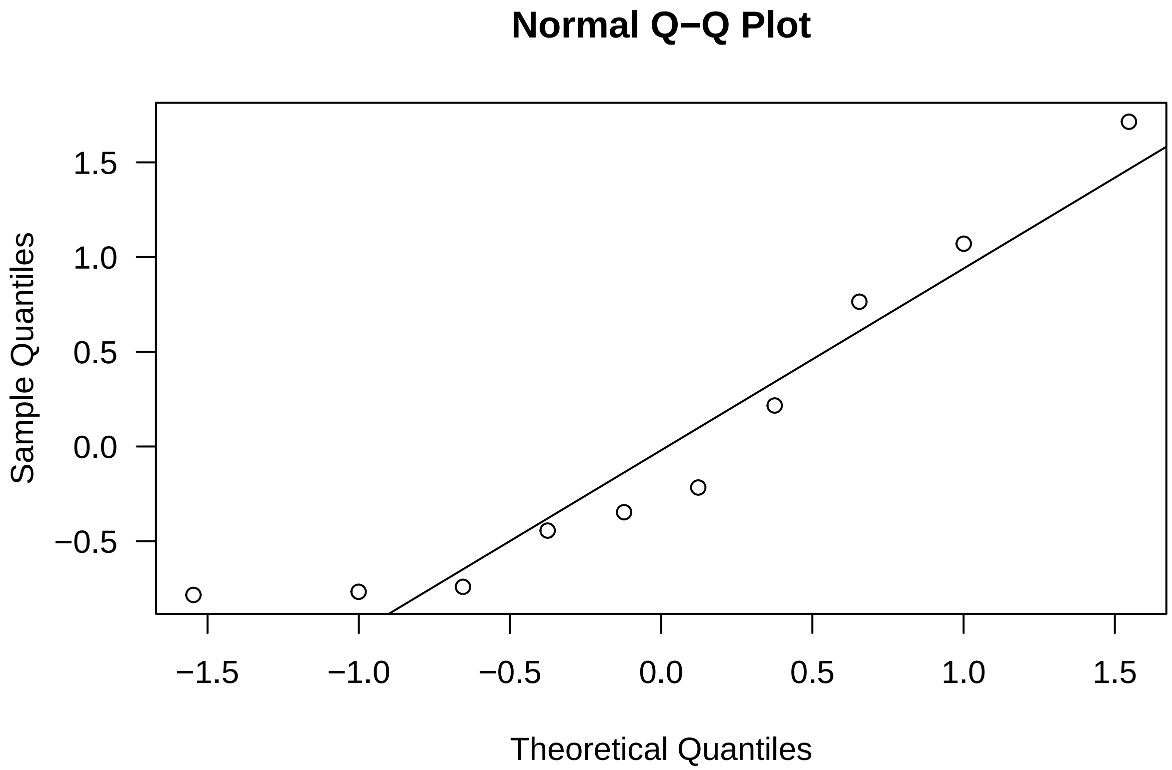
```
dfun <- function(object) {
  with(object,sum((weights * residuals^2)[weights > 0])/df.residual)
}

dfun(outsideglm)
```

```
## [1] 0.2163081
```

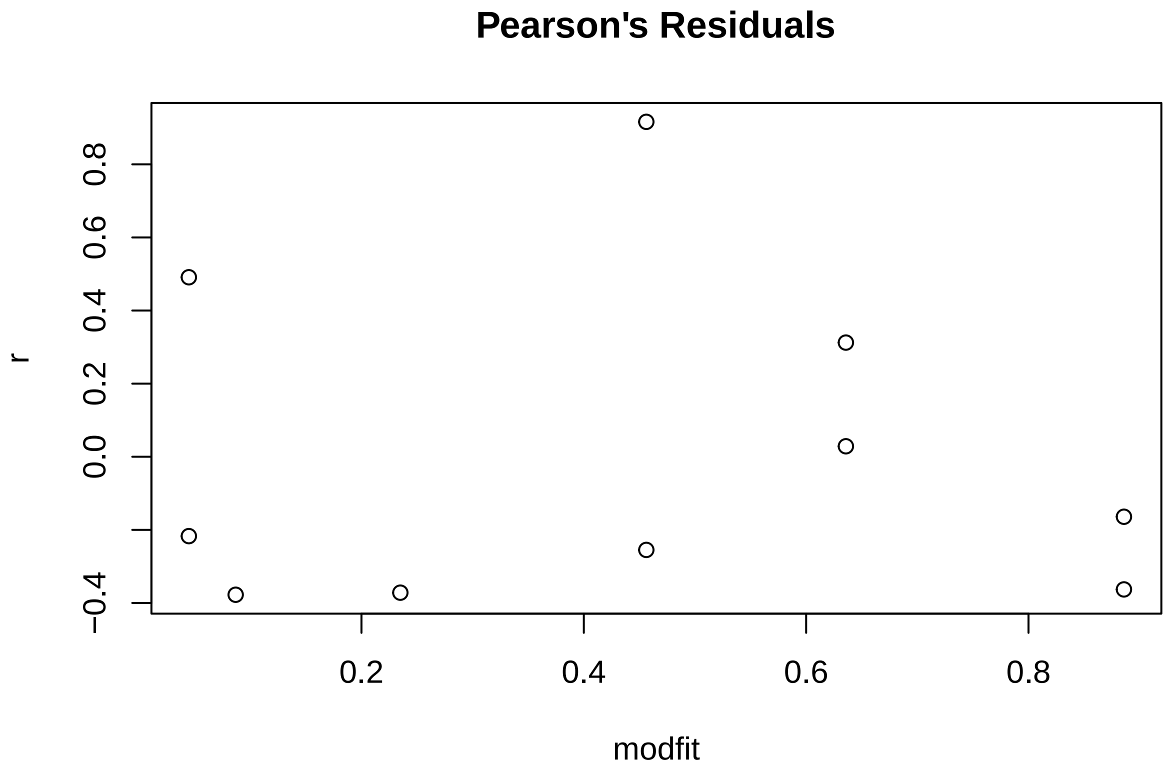
```
pseudoR2 <- 1 - (outsideglm$deviance / outsideglm$null.deviance)
pseudoR2
```

```
## [1] 0.8532969
```

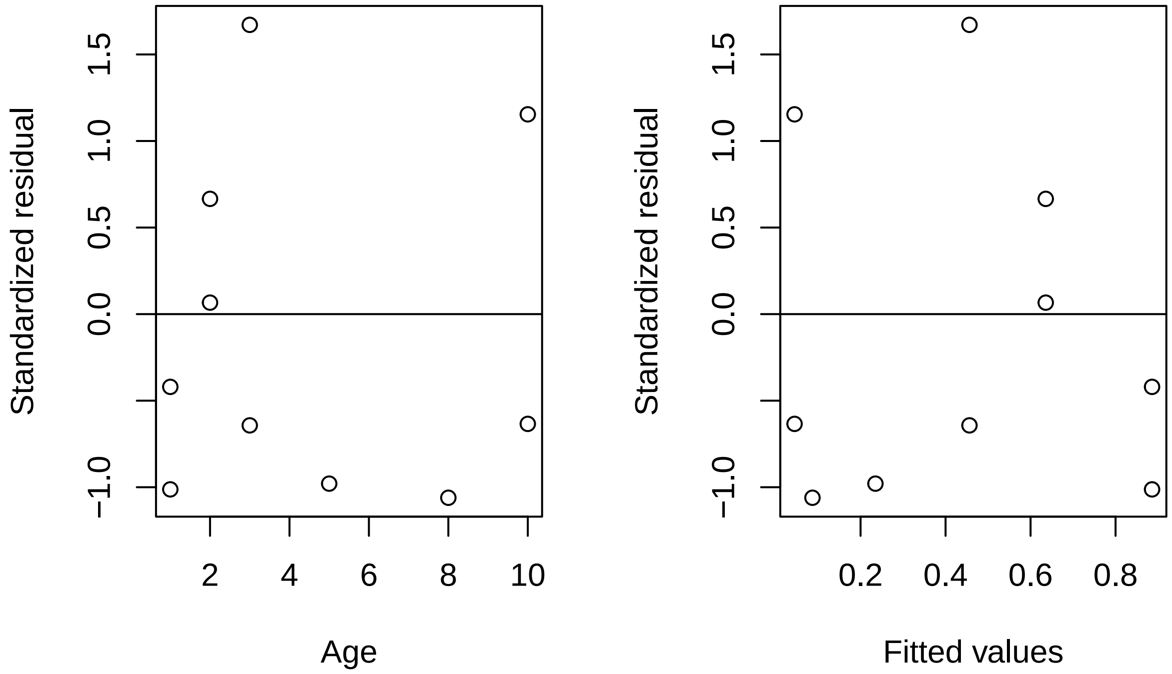
```
qr.outsideglm <- qresid(outsideglm)
qqnorm(qr.outsideglm, las = 1)
qqline(qr.outsideglm)
```



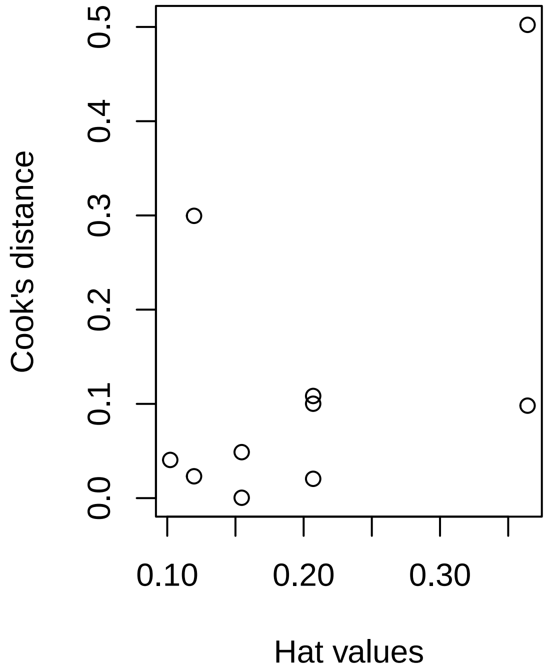
```
r<-residuals(outsideglm,type="pearson")
modfit<-fitted(outsideglm)

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



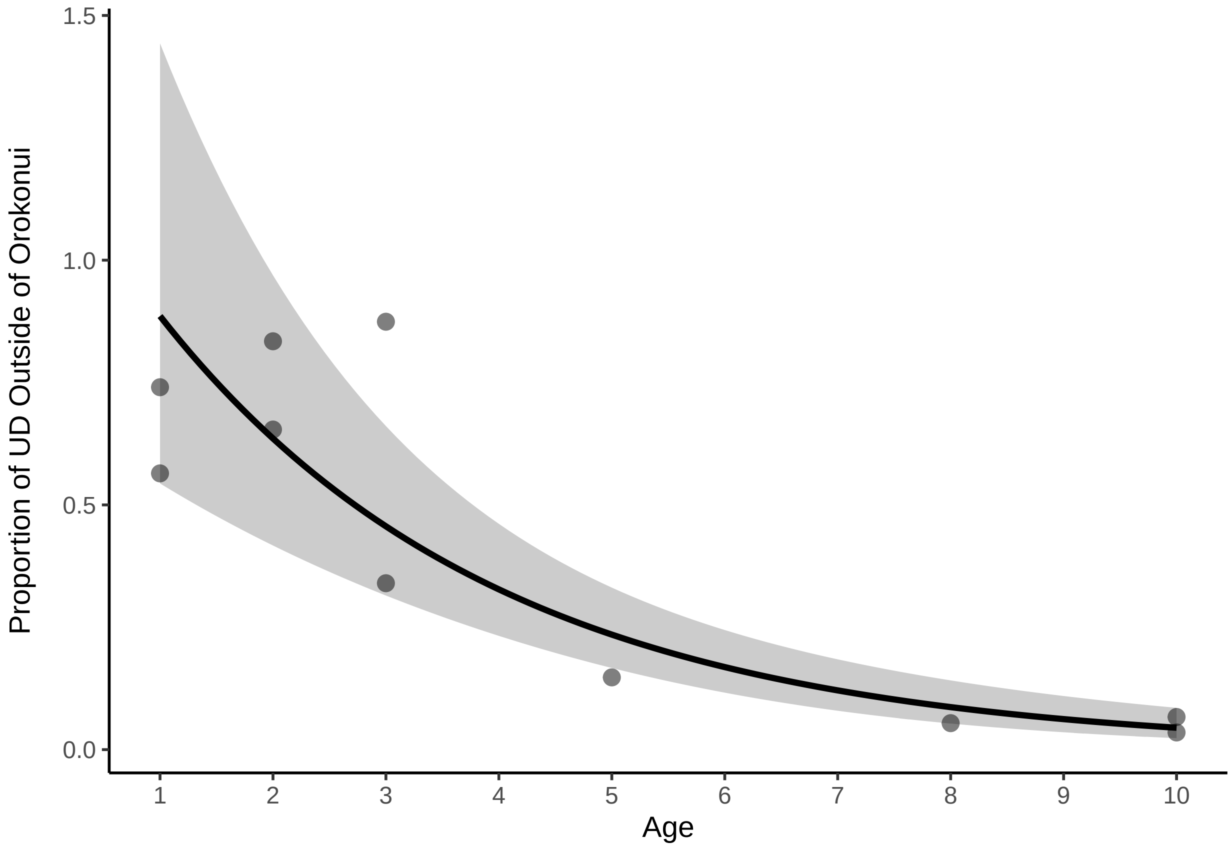
```
par(mfrow=c(1,2)) # change plot window to accommodate 2 plots

r = rstandard(outsideglm)
lf <- fitted(outsideglm)
plot(dBBMM$Age, r,xlab="Age",
     ylab="Standardized residual")
abline(h=0)

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



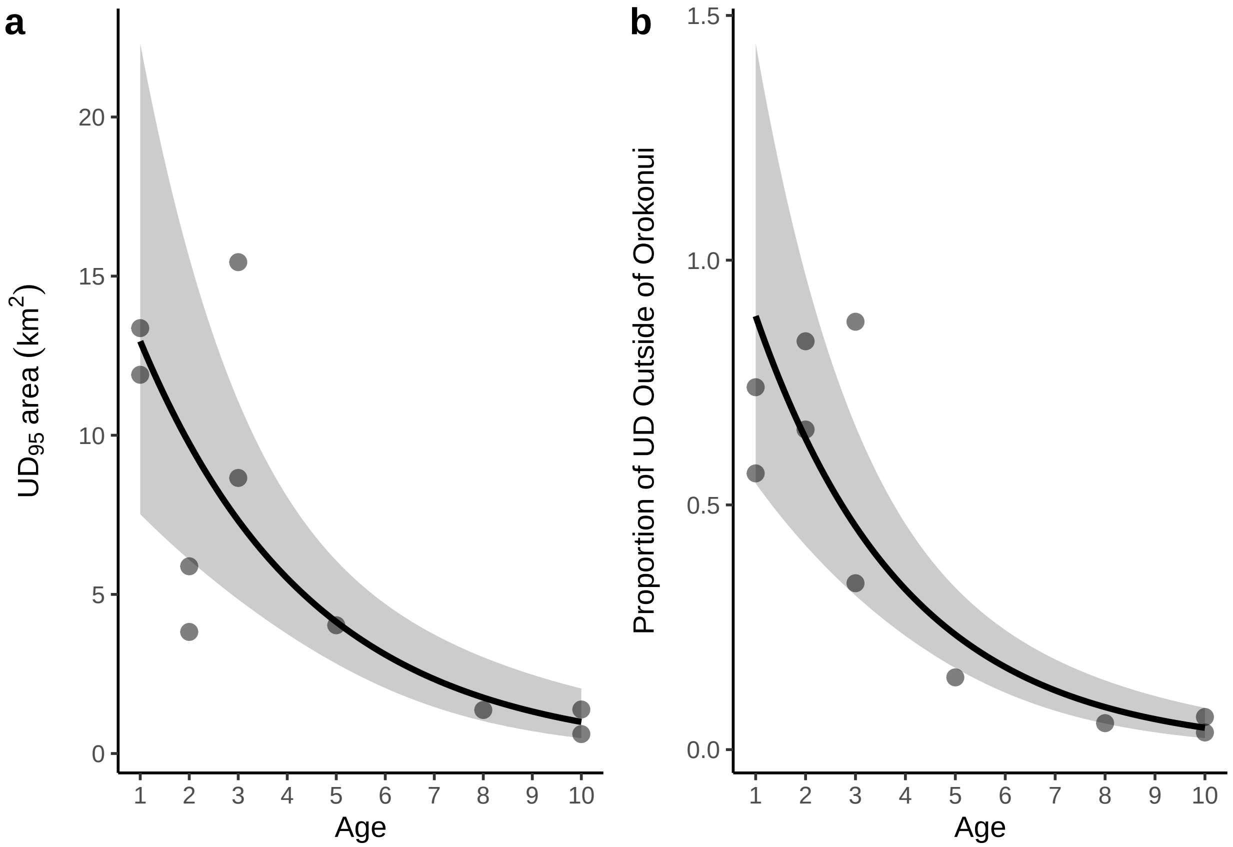
```
h <- hatvalues(outsideglm)
cd <- cooks.distance(outsideglm)
plot(h,cd,xlab="Hat values",ylab="Cook's distance")

par(mfrow=c(1,1)) # return to single plotting
```



```
outsideglmplot <- effect_plot(outsideglm,
                              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 = "Proportion of UD Outside of Orokonui") +
  theme_classic() +
  theme(axis.title.y = element_text(margin = margin(r = 10)))

outsideglmplot
```



```
ggarrange(area_age_glm_plot, outsideglmplot, labels = "auto")
```



References

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.

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] statmod_1.4.37   MuMIn_1.47.1     jtools_2.2.0     patchwork_1.1.2
##  [5] ggspatial_1.1.6  ggpubr_0.4.0     sf_1.0-8         here_1.0.1
##  [9] lattice_0.20-45  move_4.1.8       rgdal_1.5-32     raster_3.5-29
## [13] sp_1.5-0         geosphere_1.5-14 forcats_0.5.2    stringr_1.4.1
## [17] dplyr_1.0.9      purrr_0.3.4      readr_2.1.2      tidyr_1.2.0
## [21] tibble_3.1.8     ggplot2_3.3.6    tidyverse_1.3.2  knitr_1.40
##
## 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        fansi_1.0.3         lubridate_1.8.0
## [13] xml2_1.3.3          codetools_0.2-18    cachem_1.0.6
## [16] jsonlite_1.8.0      broom_1.0.1         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] htmltools_0.5.3     tools_4.2.1         gtable_0.3.1
## [31] glue_1.6.2          wk_0.6.0            Rcpp_1.0.9
## [34] carData_3.0-5       cellranger_1.1.0    vctrs_0.4.1
## [37] nlme_3.1-157        xfun_0.32           rvest_1.0.3
## [40] lifecycle_1.0.3     rstatix_0.7.0       googlesheets4_1.0.1
## [43] terra_1.6-7         MASS_7.3-57         scales_1.2.1
## [46] hms_1.1.2           parallel_4.2.1      yaml_2.3.5
## [49] memoise_2.0.1       pander_0.6.5        stringi_1.7.8
## [52] highr_0.9           e1071_1.7-11        rlang_1.0.6
## [55] pkgconfig_2.0.3     evaluate_0.16       labeling_0.4.2
## [58] cowplot_1.1.1       tidyselect_1.1.2    magrittr_2.0.3
## [61] R6_2.5.1            generics_0.1.3      DBI_1.1.3
## [64] pillar_1.8.1        haven_2.5.1         withr_2.5.0
## [67] units_0.8-0         abind_1.4-5         modelr_0.1.9
## [70] crayon_1.5.1        car_3.1-0           KernSmooth_2.23-20
## [73] utf8_1.2.2          tzdb_0.3.0          rmarkdown_2.16
## [76] grid_4.2.1          readxl_1.4.1        reprex_2.0.2
## [79] digest_0.6.29       classInt_0.4-7      stats4_4.2.1
## [82] munsell_0.5.0       viridisLite_0.4.1
```