Untitled

Stefano Mezzini

11/01/2022

```r
library('dplyr')     # for data wrangling
library('purrr')     # for functional programming
library('mgcv')      # for fitting GAMs
library('ctmm')      # for animal movement stuff
library('metafor')   # for meta-analysis
N <- 74 # number of tapirs

tapirs <- readRDS('../models/tapirs-final.rds') # tapir data

# categories with 1 individual -> group ages together
table(tapirs$age)
```

```
##
##     ADULT ADULT-OLD  JUVENILE SUB-ADULT
##        52         1         1        20
```

```r
table(tapirs$adult)
```

```
##
## No Yes
## 21  53
```

```r
# home range estimates
meta(tapirs$model, plot = FALSE) %>% round(2)
```

```
##                           <U+0394>AICc
## inverse-Gaussian          0.00
## Dirac-d                   17382.45
```

```
##              low  est  high
## mean (km²)  6.53 8.31 10.42
## CoV² (RVAR) 0.65 1.02  1.48
## CoV  (RSTD) 0.81 1.02  1.22
```

```r
d.hr <-
  escalc(measure = 'MN', # regression on the (untransformed) mean
         mi = area.est, # estimates
         sdi = map_dbl(1:N, # standard deviations
                       function(i)
                         ctmm:::area.covm(tapirs$model[[i]]$sigma)) %>% sqrt(),
         ni = map_dbl(tapirs$model, ctmm:::DOF.area), # degrees of freedom
         data = tapirs)
summary(rma(yi ~ 1, vi, data = d.hr, method = 'REML'))
```

```
##
```

```
## Random-Effects Model (k = 74; tau^2 estimator: REML)
##
##    logLik   deviance        AIC        BIC       AICc
## -335.9006   671.8011   675.8011   680.3820   675.9725
##
## tau^2 (estimated amount of total heterogeneity): 0 (SE = 52.1630)
## tau (square root of estimated tau^2 value):      0
## I^2 (total heterogeneity / total variability):   0.00%
## H^2 (total variability / sampling variability):  1.00
##
## Test for Heterogeneity:
## Q(df = 73) = 0.8684, p-val = 1.0000
##
## Model Results:
##
## estimate       se     zval     pval    ci.lb    ci.ub
##   3.5296   2.7285   1.2936   0.1958  -1.8181   8.8773
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The overall mean home range estimate is 8.31 km^2 with CI (6.53, 10.4) km^2.

```
range(tapirs$area.est) %>% round(digits = 1)
```

```
## [1]  1.0 29.7
```

The range in home range area estimates is 1-29.7 km^2.

```
round(range(tapirs$tau.position.est, na.rm = TRUE) / 60 / 60 / 24, 2)
```

```
## [1]  0.05 12.76
```

The range in range crossing time is 0.05-12.8 days.

Change mean estimates (now using GLMs instead of normal CIs)

```
est <- function(y, parameter = c('mean', 'lwr', 'upr')) {
  if(sum(!is.na(y)) > 2) { # if there's at least 2 observations
    # fit a Gamma GLM with only an intercept to estimate the group-level mean
    m <- gam(y ~ 1,
             family = Gamma(link = 'log'),
             method = 'REML')
    
    if(length(parameter) > 1) stop('Specify a single parameter.\n')
    
    pred <- predict(m, newdata = tibble(c = 1), se.fit = TRUE, scale = 'link') %>%
      as.data.frame() %>%
      mutate(lwr = exp(fit - 1.96 * se.fit),
             mean = exp(fit),
             upr = exp(fit + 1.96 * se.fit))
    pred[1, parameter]
  } else {NA_real_}
}

tap <-
```

```
bind_rows(
  bind_cols(
    name = c('Atlantic forest', 'Pantanal', 'Cerrado', 'Overall'),
    region.lab = c('Atlantic forest', 'Pantanal', 'Cerrado', 'Overall'),
    bind_rows(
      meta(tapirs$model[tapirs$region == 'atlantica'], plot = FALSE)[1, ],
      meta(tapirs$model[tapirs$region == 'pantanal'], plot = FALSE)[1, ],
      meta(tapirs$model[tapirs$region == 'cerrado'], plot = FALSE)[1, ],
      meta(tapirs$model, plot = FALSE)[1, ]) %>%
      rename(area.low = low, area.est = est, area.high = high),
    # regional estimates
    group_by(tapirs, region) %>%
      summarize(tau.position.e = est(tau.position.est, 'mean'),
                tau.position.low = est(tau.position.est, 'lwr'),
                tau.position.high = est(tau.position.est, 'upr'),
                tau.velocity.e = est(tau.velocity.est, 'mean'),
                tau.velocity.low = est(tau.velocity.est, 'lwr'),
                tau.velocity.high = est(tau.velocity.est, 'upr'),
                speed.e = est(speed.est, 'mean'),
                speed.low = est(speed.est, 'lwr'),
                speed.high = est(speed.est, 'upr')) %>%
      rename(tau.position.est = tau.position.e,
             tau.velocity.est = tau.velocity.e,
             speed.est = speed.e) %>%
      bind_rows(
        tapirs %>%
          summarize(tau.position.est = est(tapirs$tau.position.est, 'mean'),
                    tau.position.low = est(tapirs$tau.position.est, 'lwr'),
                    tau.position.high = est(tapirs$tau.position.est, 'upr'),
                    tau.velocity.est = est(tapirs$tau.velocity.est, 'mean'),
                    tau.velocity.low = est(tapirs$tau.velocity.est, 'lwr'),
                    tau.velocity.high = est(tapirs$tau.velocity.est, 'upr'),
                    speed.est = est(tapirs$speed.est, 'mean'),
                    speed.low = est(tapirs$speed.est, 'lwr'),
                    speed.high = est(tapirs$speed.est, 'upr'))))) %>%
  mutate(
    # convert from secods to days
    tau.position.est = tau.position.est / (60^2 * 24),
    tau.position.low = tau.position.low / (60^2 * 24),
    tau.position.high = tau.position.high / (60^2 * 24)) %>%
  select(region.lab, area.est, area.low, area.high, tau.position.est, tau.position.low,
         tau.position.high, tau.velocity.est, tau.velocity.low, tau.velocity.high, speed.est,
         speed.low, speed.high)
```

```
##                        <U+0394>AICc
## inverse-Gaussian          0.000
## Dirac-d                1310.217
##                        <U+0394>AICc
## inverse-Gaussian          0.000
## Dirac-d                8562.481
##                        <U+0394>AICc
## inverse-Gaussian          0.000
## Dirac-d                6961.898
##                        <U+0394>AICc
```

```
## inverse-Gaussian          0.00
## Dirac-d              17382.45
```

```
knitr::kable(t(tap)) # transpose to make readable
```

region.lab	Atlantic forest	Pantanal	Cerrado	Overall
area.est	10.285217	5.936261	13.383010	8.307124
area.low	4.381964	4.695496	7.652764	6.527058
area.high	20.741635	7.400507	21.797597	10.418790
tau.position.est	0.5393670	0.6555289	0.8000626	0.7227463
tau.position.low	0.1883971	0.3981069	0.3633637	0.4191364
tau.position.high	1.544168	1.079404	1.761596	1.246282
tau.velocity.est	NA	1441.737	1659.872	1579.777
tau.velocity.low	NA	1261.780	1343.525	1352.844
tau.velocity.high	NA	1647.360	2050.707	1844.776
speed.est	NA	12.14095	10.67056	11.17040
speed.low	NA	10.638837	9.325726	10.107009
speed.high	NA	13.85514	12.20933	12.34567

Can we use the estimates from above instead of using escalc?

```
# tau_p
d.tp <-
  escalc(measure = 'MN', # regression on the (untransformed) mean
         mi = tau.position.est,
         sdi = map_dbl(1:N,
                       function(i)
                         diag(tapirs$model[[i]]$COV)['tau position']) %>% sqrt(),
         ni = map_dbl(tapirs$model, ctmm:::DOF.area),
         data = tapirs)

# may not be able to obtain stable results due to small sample sizes
# estimates are in seconds instead of days
summary(rma(yi ~ region, vi, data = d.tp, method = 'REML'))
summary(rma(yi ~ sex, vi, data = d.tp, method = 'REML'))
summary(rma(yi ~ adult, vi, data = d.tp, method = 'REML'))
```

Broken:

```
# tau_v
d.tv <-
  escalc(measure = 'MN',
         mi = tau.velocity.est,
         sdi = map_dbl(1:N,
                       function(i)
                         speed(tapirs$model[[i]]$COV)['tau velocity']) %>%
           sqrt(),
         ni = map_dbl(1:N,
                      function(i)
                        summary(tapirs$model[[i]])$DOF['speed'],
         data = tapirs))
```

```r
summary(rma(yi ~ 1, vi, data = d.tv, method = 'REML'))

# speed
d.speed <-
  escalc(measure = 'MN',
         mi = speed.est,
         sdi = map_dbl(1:N,
                       function(i) i) %>% sqrt(),
         ni = map_dbl(1:N,
                      function(i)
                        summary(tapirs$model[[i]])$DOF['speed'],
         data = tapirs))

summary(rma(yi ~ 1, vi, data = d.speed, method = 'REML'))
summary(rma(yi ~ adult, vi, data = d.speed, method = 'REML'))
summary(rma(yi ~ sex, vi, data = d.speed, method = 'REML'))
```

This part works:

```r
# same tests with mgcv::gam ###################################################
m.speed.sex <- gam(speed.est ~ sex,
                   family = Gamma('log'),
                   data = tapirs,
                   method = 'REML')
m.speed.age <- gam(speed.est ~ adult,
                   family = Gamma('log'),
                   data = tapirs,
                   method = 'REML')
# model summaries
summary(m.speed.sex)
```

```
##
## Family: Gamma
## Link function: log
##
## Formula:
## speed.est ~ sex
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.35276    0.06872  34.235   <2e-16 ***
## sexMALE      0.12420    0.10006   1.241     0.22
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## R-sq.(adj) =  0.00946   Deviance explained = 2.16%
## -REML = 156.04  Scale est. = 0.13224   n = 53
```

```r
summary(m.speed.age)
```

```
##
## Family: Gamma
```

```
## Link function: log
##
## Formula:
## speed.est ~ adult
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.2532     0.0928  24.281   <2e-16 ***
## adultYes      0.2168     0.1096   1.978   0.0533 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## R-sq.(adj) =  0.0454   Deviance explained = 5.18%
## -REML = 155.11  Scale est. = 0.12917   n = 53
```

```r
# predictions
tibble(Sex = c('MALE', 'FEMALE'),
       bind_cols(predict(m.speed.sex, tibble(sex = Sex), se.fit = TRUE))) %>%
  mutate(est = round(exp(fit), 2),
         lwr = round(exp(fit - 1.96 * se.fit), 2),
         upr = round(exp(fit + 1.96 * se.fit), 2))
```

```
## # A tibble: 2 x 6
##   Sex      fit se.fit   est   lwr   upr
##   <chr>  <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 MALE    2.48 0.0727  11.9  10.3  13.7
## 2 FEMALE  2.35 0.0687  10.5  9.19  12.0
```

```r
tibble(Adult = c('Yes', 'No'),
       bind_cols(predict(m.speed.age, tibble(adult = Adult), se.fit = TRUE))) %>%
  mutate(est = round(exp(fit), 2),
         lwr = round(exp(fit - 1.96 * se.fit), 2),
         upr = round(exp(fit + 1.96 * se.fit), 2))
```

```
## # A tibble: 2 x 6
##   Adult  fit se.fit   est   lwr   upr
##   <chr> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 Yes    2.47 0.0583  11.8  10.6  13.2
## 2 No     2.25 0.0928  9.52  7.94  11.4
```

Estimate differences in home range sizes using `ctmm::meta()`:

```r
# by sex
meta(list(female = filter(tapirs, sex == 'FEMALE')$akde,
          male = filter(tapirs, sex == 'MALE')$akde),
     plot = FALSE)
```

```
## * Sub-population female
##
##                       <U+0394>AICc
## inverse-Gaussian         0.000
## Dirac-d               7367.412
##
## * Sub-population male
##
##                       <U+0394>AICc
## inverse-Gaussian         0.000
```

```
## Dirac-d              7240.514

## * Joint population

##                      <U+0394>AICc
## inverse-Gaussian        0.00
## Dirac-d             14778.71

## * Joint population versus sub-populations (best models)

##                      <U+0394>AICc
## Joint population     0.000000
## Sub-population       4.035555

## , , low
##
##           /female     /male
## female/ 1.0000000 0.7026962
## male/   0.5608838 1.0000000
##
## , , est
##
##          /female     /male
## female/ 1.000000 1.094326
## male/   0.873601 1.000000
##
## , , high
##
##          /female     /male
## female/ 1.000000 1.629906
## male/   1.300086 1.000000
```

```r
meta(filter(tapirs, sex == 'FEMALE')$akde, plot = FALSE) # CIs for females
```

```
##                      <U+0394>AICc
## inverse-Gaussian        0.000
## Dirac-d              7367.412

##                  low       est      high
## mean (km²)  4.5262477 6.1131618 8.068985
## CoV² (RVAR) 0.4615141 0.8615983 1.384492
## CoV  (RSTD) 0.6857438 0.9369617 1.187722
```

```r
meta(filter(tapirs, sex == 'MALE')$akde, plot = FALSE) # CIs for males
```

```
##                      <U+0394>AICc
## inverse-Gaussian        0.000
## Dirac-d              7240.514

##                  low       est      high
## mean (km²)  4.0316215 5.4605393 7.232391
## CoV² (RVAR) 0.3590258 0.7091030 1.176317
## CoV  (RSTD) 0.6057262 0.8512716 1.096417
```

```r
# by age group
meta(list(female = filter(tapirs, adult == 'No')$akde,
          male = filter(tapirs, adult == 'Yes')$akde),
     plot = FALSE)
```

```
## * Sub-population female
```

```
##                            <U+0394>AICc
## inverse-Gaussian           0.000
## Dirac-d                 4785.763

## * Sub-population male

##                            <U+0394>AICc
## inverse-Gaussian           0.000
## Dirac-d                 9841.229

## * Joint population

##                            <U+0394>AICc
## inverse-Gaussian            0.00
## Dirac-d                 14778.71

## * Joint population versus sub-populations (best models)

##                            <U+0394>AICc
## Sub-population          0.00000
## Joint population        3.08865

## , , low
##
##          /female     /male
## female/ 1.0000000 0.6457535
## male/   0.3654871 1.0000000
##
## , , est
##
##          /female    /male
## female/ 1.0000000 1.28781
## male/   0.7053567 1.00000
##
## , , high
##
##          /female    /male
## female/ 1.000000 2.179039
## male/   1.321561 1.000000
```

```r
meta(filter(tapirs, adult == 'No')$akde, plot = FALSE) # CIs for sub-adults
```

```
##                            <U+0394>AICc
## inverse-Gaussian           0.000
## Dirac-d                 4785.763

##                  low      est       high
## mean (km²)   3.869323 6.985186 11.648677
## CoV² (RVAR) 0.576386 1.674820  3.348701
## CoV  (RSTD) 0.776464 1.323575  1.871556
```

```r
meta(filter(tapirs, adult == 'Yes')$akde, plot = FALSE) # CIs for adults
```

```
##                            <U+0394>AICc
## inverse-Gaussian           0.000
## Dirac-d                 9841.229

##                   low       est      high
## mean (km²)   4.3936513 5.3698378 6.4954740
## CoV² (RVAR) 0.3073040 0.5063389 0.7542647
```

```
## CoV  (RSTD) 0.5578842 0.7161114 0.8740209
```

Difference in home range estimates by sex and age

```r
m.hr.sex <- gam(area.est ~ sex,
                family = Gamma('log'),
                data = tapirs,
                method = 'REML')
m.hr.age <- gam(area.est ~ adult,
                family = Gamma('log'),
                data = tapirs,
                method = 'REML')
# model summaries
summary(m.hr.sex)
```

```
##
## Family: Gamma
## Link function: log
##
## Formula:
## area.est ~ sex
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.8360     0.1542  11.910   <2e-16 ***
## sexMALE      -0.1441     0.2344  -0.615    0.541
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## R-sq.(adj) =  -0.00882   Deviance explained = 0.758%
## -REML = 202.62  Scale est. = 0.9982    n = 74
```

```r
summary(m.hr.age)
```

```
##
## Family: Gamma
## Link function: log
##
## Formula:
## area.est ~ adult
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.9477     0.2117   9.198 8.94e-14 ***
## adultYes     -0.2484     0.2502  -0.993    0.324
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## R-sq.(adj) =  0.000201   Deviance explained = 1.94%
## -REML = 202.04  Scale est. = 0.94157    n = 74
```

```
# predictions
tibble(Sex = c('MALE', 'FEMALE'),
       bind_cols(predict(m.hr.sex, tibble(sex = Sex), se.fit = TRUE))) %>%
  mutate(est = round(exp(fit), 2),
         lwr = round(exp(fit - 1.96 * se.fit), 2),
         upr = round(exp(fit + 1.96 * se.fit), 2))
```

```
## # A tibble: 2 x 6
##    Sex       fit se.fit   est   lwr   upr
##    <chr>   <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 MALE     1.69  0.177  5.43  3.84  7.68
## 2 FEMALE   1.84  0.154  6.27  4.64  8.48
```

```
tibble(Adult = c('Yes', 'No'),
       bind_cols(predict(m.hr.age, tibble(adult = Adult), se.fit = TRUE))) %>%
  mutate(est = round(exp(fit), 2),
         lwr = round(exp(fit - 1.96 * se.fit), 2),
         upr = round(exp(fit + 1.96 * se.fit), 2))
```

```
## # A tibble: 2 x 6
##    Adult   fit se.fit   est   lwr   upr
##    <chr> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 Yes    1.70  0.133  5.47  4.21   7.1
## 2 No     1.95  0.212  7.01  4.63  10.6
```