Animal movement simulation

Pierre Cottais & An Hoàng

30/11/2021

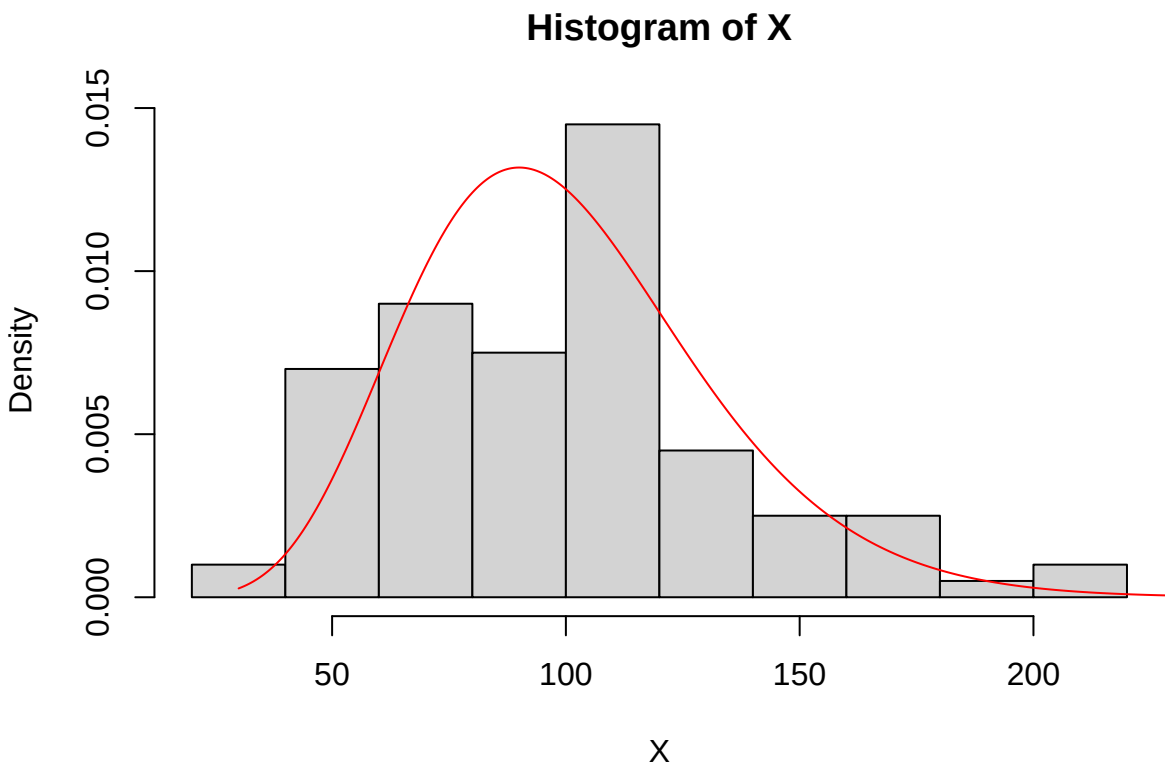
Contents

1 Generating individual animal step lengths data (utilization distribution)

Probability of obtaining a sample at some distance, $l'_{t,i}$ from the previous observed point ($l'_{t,i} = ||x'_{t,i} - x$ is givenby the gammaPDF:

$$g(l'_{t,i}|b_1, b_2) = \frac{1}{\Gamma(b_1).b1^{b2}}.l'^{b_1-1}_{t,i}.e^{-\frac{l'_{t,i}}{b_2}}$$

```
b1 <- 10 ; b2 <- 10
n <- 100
X <- rgamma(n, scale = b1, shape = b2)
hist(X, freq=F)
x <- seq(30, 300, by = 0.1)
lines(x, dgamma(x, scale = b1, shape = b2), col = "red")
```



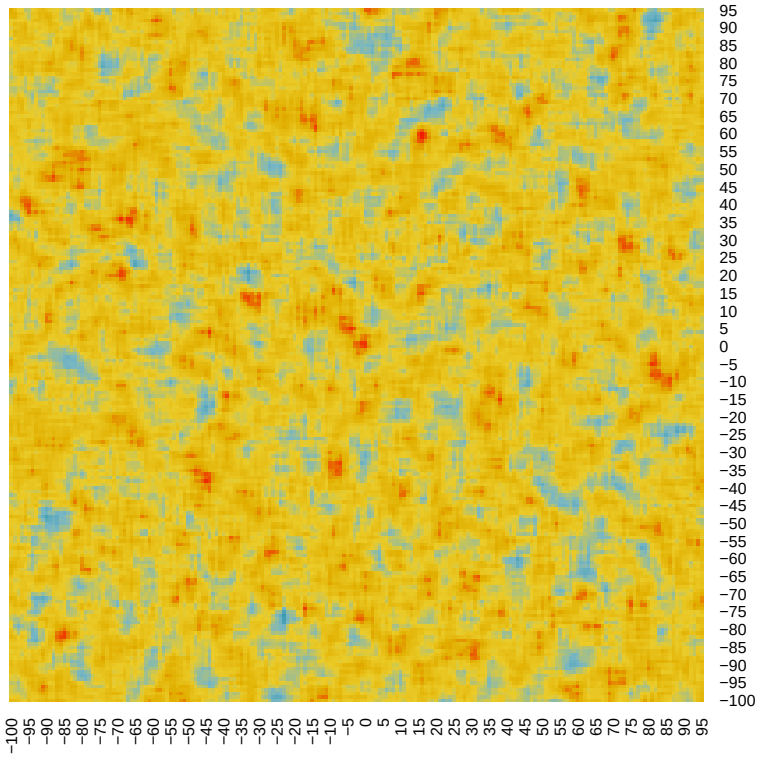

```
s <- rep_len(1:3, 200000)
L <- c(0.01, 0.05, 0.1)
nimporte <- case_when(s == 1 ~ L[1],
                      s == 2 ~ L[2],
                      s == 3 ~ L[3])
fin <- Sys.time()
```

```
# Environment grid ver2
lat <- seq(-100,100, 1)
long <- seq(-100,100, 1)
n <- length(lat)
m <- length(long)
envi <- matrix(data = runif(n*m),nrow = n, ncol = m)
colnames(envi) <- lat
row.names(envi) <- long
```

```
#autocorection
p <- 5 #spatial unit

#sliding window
for (i in 1:(n-p)){
  for (j in 1:(m-p)){
    sub <- envi[i:(i+p),j:(j+p)]
    envi[i,j] = mean(sub)
  }
```

```
}
envi <- envi[1:(n-p),1:(m-p)]
```

```
pal <- wes_palette(40401,name = "Zissou1", type = "continuous")
heatmap(envi, Rowv = NA, Colv = NA, col = pal)
```



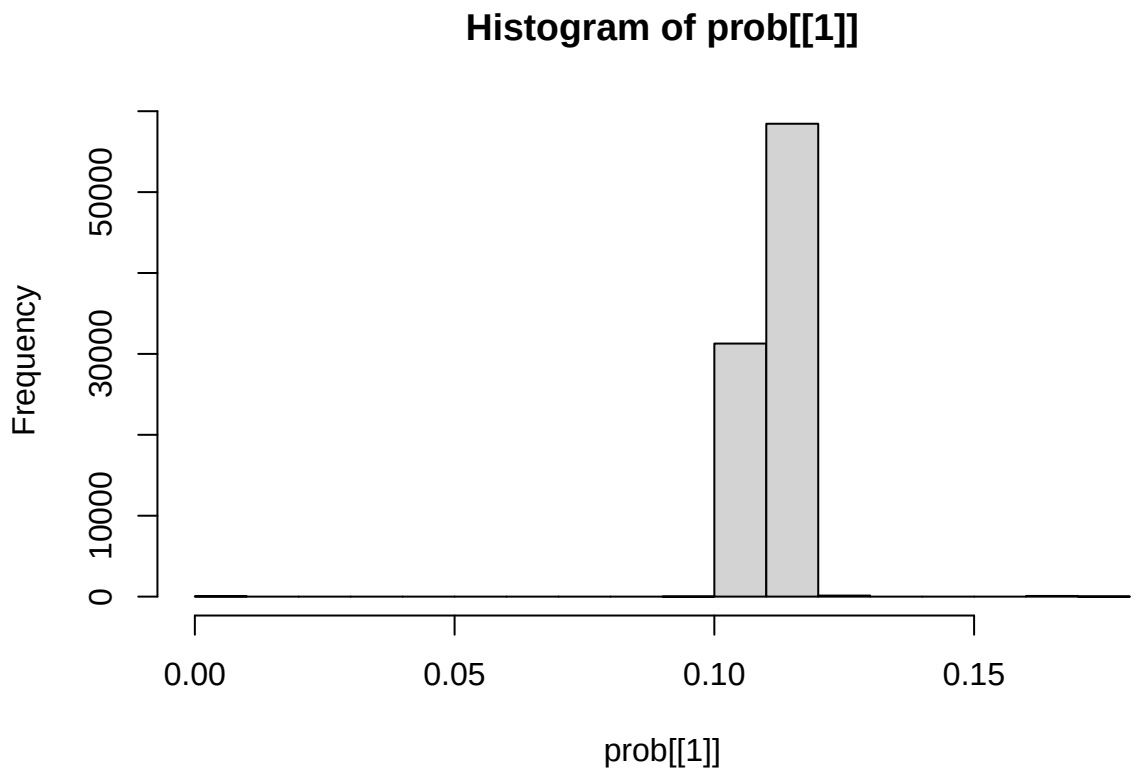

```
# habitat selection step
mu <- 1.7918
# chose moving length always || x- x(t) || = 1
omega <- 1
# #local habitat quality
envi_dt <- as.data.frame(envi)

# apply truncated redistribution calculus kernel to the whole grid-landscape
envi_exp <- exp(omega*envi-mu)

# adding two null columns at the "borders"
envi_exp <- cbind(rep(0, times = nrow(envi_exp)),
                  envi_exp,
                  rep(0, times = nrow(envi_exp)))
# adding two null rows at the "borders"
# adding two null columns at the "borders"
envi_exp <- rbind(rep(0, times = ncol(envi_exp)),
                  envi_exp,
                  rep(0, times = ncol(envi_exp)))
```

```r
envi_exp <- as.data.frame(envi_exp)

# number of burn-in steps
step1 <- 2*length(envi)^2
# number of simulate steps
step2 <- 10^5
# # starting point
# bird <- which(envi == max(envi), arr.ind = TRUE)
```

```r
moves <- function(steps){
  # matrix of avaible movements
  code <- cbind(rep(c(1, 0, -1), 3), rep(c(-1, 0, 1), each = 3))

  # starting point (and   )
  coordinates <- which(envi_exp == max(envi_exp), arr.ind = TRUE)
  latitude <- coordinates[1,1]
  longitude <- coordinates[1,2]
  kernel <- envi_exp[(latitude-1):(latitude+1),
                     (longitude-1):(longitude+1)]

  # truncated redistribution kernel calculus
  prob_avail <- kernel/sum(kernel)
  prob_avail <- prob_avail %>% as_vector()
  names(prob_avail) <- 1:9
  num_cell <- which(rmultinom(1, 1, prob_avail)==1)
  move <- code[num_cell,]
  coordinates <- rbind(coordinates,
                       c(latitude+move[1], longitude+move[2]))
  list_avail <- list(prob_avail, num_cell)

  for(i in 2:steps){
    latitude <- coordinates[i,1]
    longitude <- coordinates[i,2]

    # # dealing with "out of bounds" steps
    # if (latitude==nrow(envi_dt) | longitude==ncol(envi_dt) |
    #     latitude==1 | longitude==1){
    #   warning(paste0('stopping the algorithm at the ',i,'th step:
    #          animal reached the border of the "landscape"'))
    #   coord_dt <- as.data.frame(coordinates, row.names = FALSE)
    #   return(list(coord_dt, list_avail))
    # }else{
    kernel <- envi_exp[(latitude-1):(latitude+1),
                       (longitude-1):(longitude+1)]
    # }
    # truncated redistribution kernel calculus
    # sub <- exp(omega*kernel-mu)  # done earlier outside the function
    prob_avail <- kernel/sum(kernel)
    prob_avail <- prob_avail %>% as_vector()
    names(prob_avail) <- 1:9
    num_cell <- which(rmultinom(1, 1, prob_avail)==1)
    move <- code[num_cell,]
    coordinates <- rbind(coordinates,
```

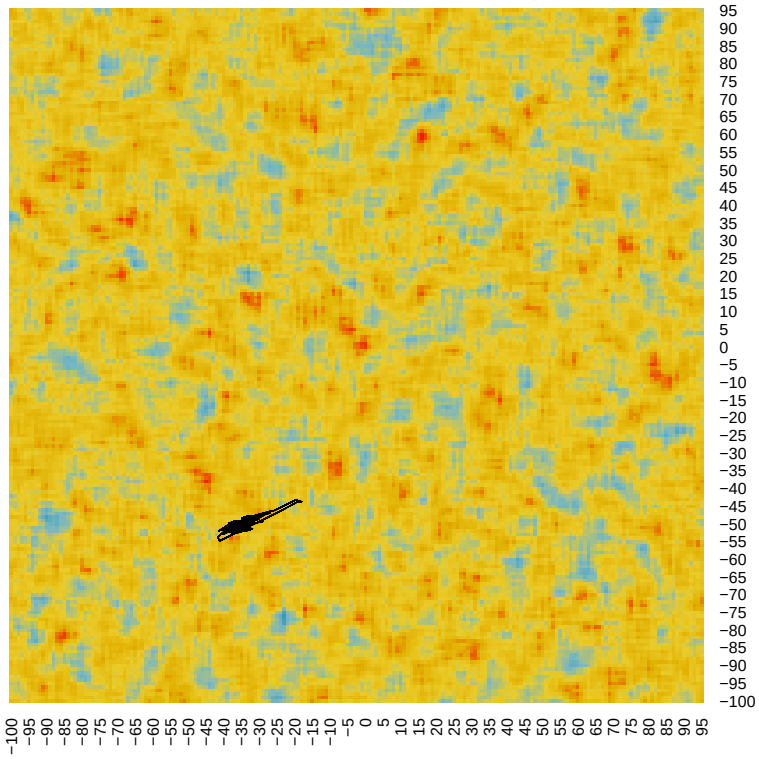
```
                          c(latitude+move[1], longitude+move[2]))
    list_avail[[1]] <- rbind(list_avail[[1]], prob_avail)
    list_avail[[2]] <- rbind(list_avail[[2]], num_cell)
  }
  coord_dt <- as.data.frame(coordinates, row.names = FALSE)
  return(list(coord_dt, list_avail))
}
```

```
res <- moves(10000)
coord <- res[[1]]
prob <- res[[2]]

hist(prob[[1]])
```




```
# sub <- which(envi == max(envi), arr.ind = TRUE)
# sub <- exp(omega*sub-mu)
# prob <- sub/sum(sub)
# prob_vect <- envi[1:3, 1:3] %>% as.data.frame() %>% as_vector()
# names(prob_vect) <- 1:9
# num_cell <- which(rmultinom(1, 1, prob_vect)==1) ; num_cell
```

```
heatmap(envi, Rowv = NA, Colv = NA, col = pal)
lines(envi_exp[coord$row, coord$col], lwd = 0.2)
```




```
# points(envi_exp[coord$row, coord$col], lwd = 0.1, pch = 3, cex = 0.2)
```

```
# seal <- read.csv("https://www.datarepository.movebank.org/bitstream/handle/10255/move.451/Grey%20seal
```

```
# seal %>%
#   mutate(timestamp = ymd_hms(timestamp)) %>%
#   select(timestamp)
```

```
# seal %>%
#   filter(tag.local.identifier == 106705) %>%
#   ggplot() + aes(x = location.long, location.lat) +
#   geom_point() +
#   geom_line()
```