

Ouro Preto corridors - overlap between corridors with and without urban zoning

Bernardo Niebuhr

Sat May 4 18:43:49 2019

Rationale

My idea here is to develop a way of calculating the overlap between corridors in different simulation scenarios. The general idea is that we have two runs of corridors and we want to measure how much the runs overlap between each other - this includes not only IF corridors overlap, but also how many corridors from each run cross pixels where there is overlap.

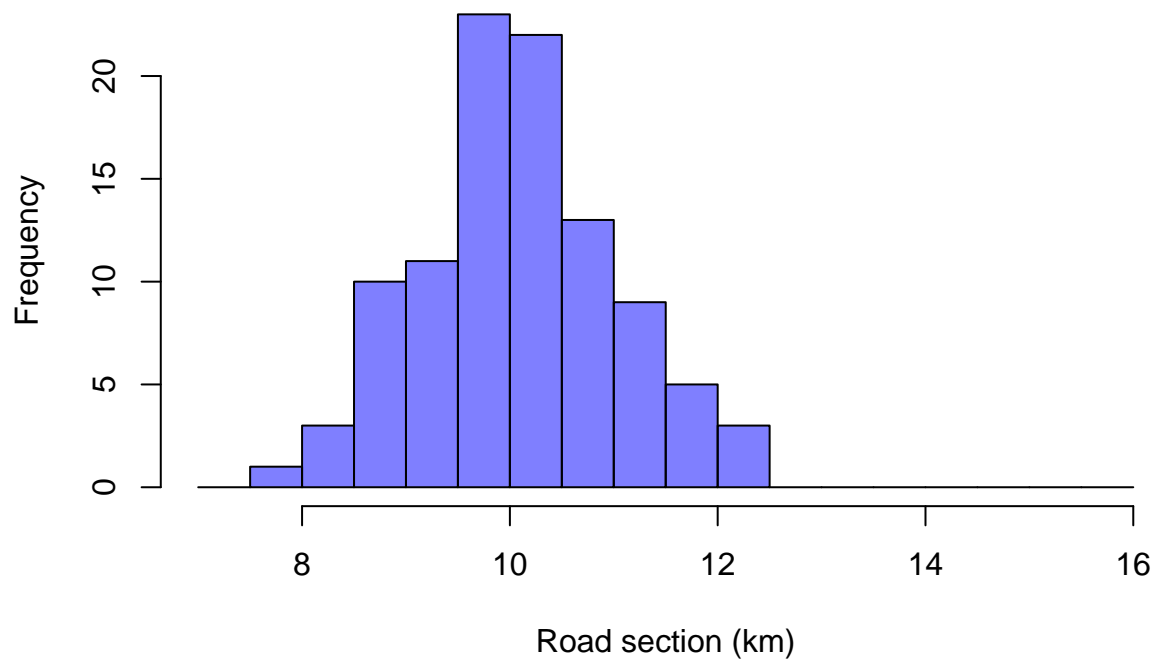
A simple example

First, let's think about a 1 dimensional example. Imagine that, instead of corridors simulations in a two dimensional space, we have the number of animals that cross a given section of a road (so, a 1D problem). It is still an issue of landscape permeability, but simplified for illustration purposes. Let's say the number of animals in a given scenario (e.g., with no road passages) is given by a Gaussian distribution with $\mu = 10$ and $\sigma = 1$. I'll sample 100 points where animals crossed. This mean most animals tend to cross the road at the kilometer 10, but there is some variation and animals also crossing at other places. One possibility of explanation of this pattern may be that, at the kilometer 10, there are habitat patches contiguous to both sides of the road, so animals are prone to approach and cross the road at this place). Take a look at the distribution below:

```
# seed to keep the random numbers fixed
set.seed(123)

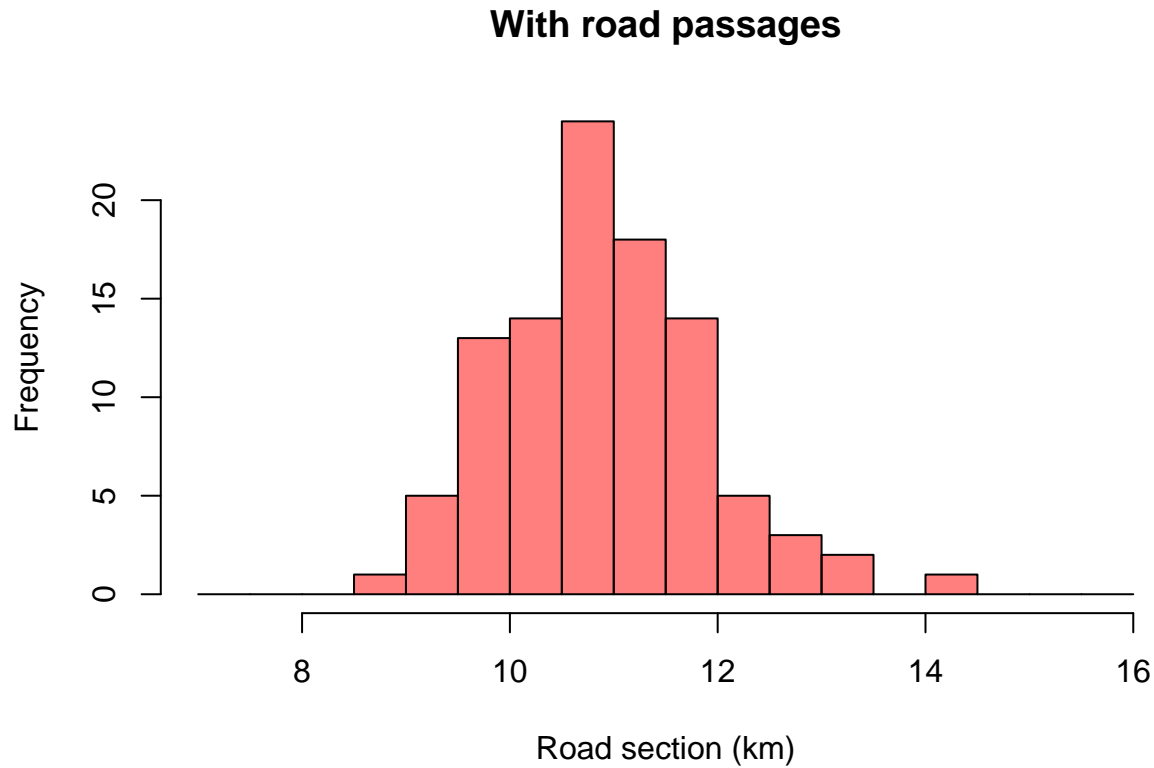
no.passages <- rnorm(100, 0, 1) + 10
breaks <- seq(7, 16, 0.5)
hist(no.passages, breaks = breaks, col = rgb(0, 0, 1, alpha = 0.5),
     xlab = 'Road section (km)', main = 'No road passages')
```

No road passages



Now I'll do the same after a road passage is built at the kilometer 11 (so I'll suppose most animals cross the road at this new point, but again with the some variation).

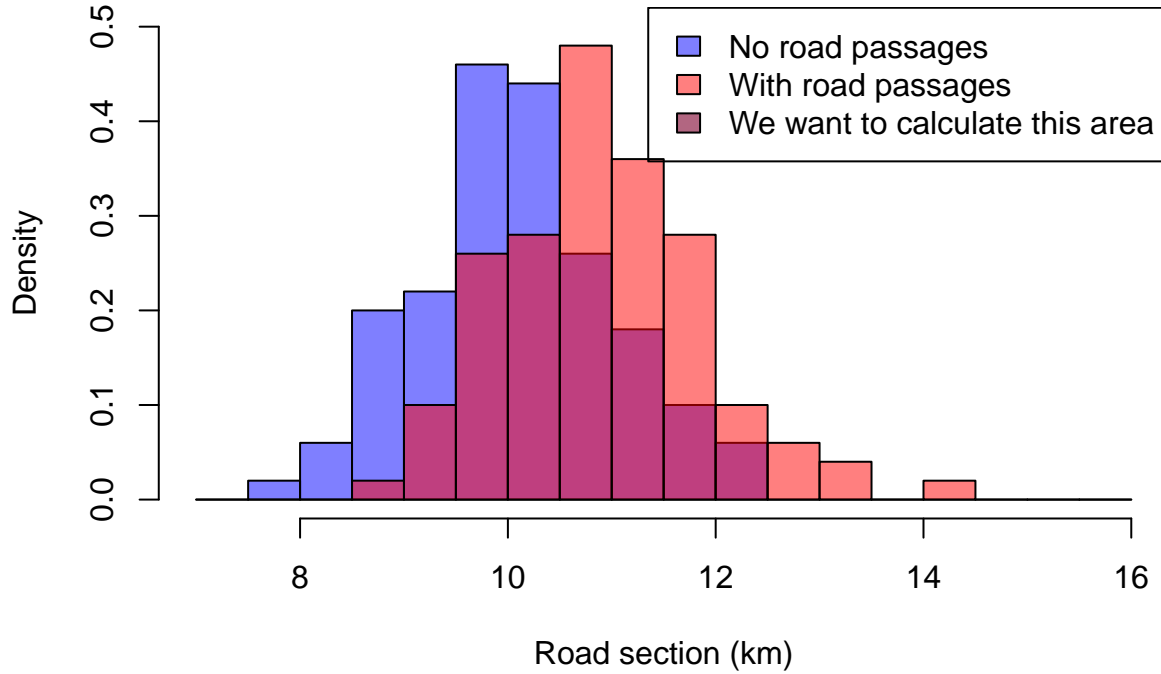
```
passages <- rnorm(100, 11, 1)
hist(passages, breaks = breaks, col = rgb(1, 0, 0, alpha = 0.5),
      xlab = 'Road section (km)', main = 'With road passages')
```



Let's compare both now. But, instead of showing the absolute frequency of records at each location, I'll show the relative frequency of records at each location. This means that, if we multiply the width and the height of each bar in this histogram, I'll get the probability of observing an animal crossing at each road section. The probability density in the y axis is then the probability per unit of length of the road section.

What we want, at the end, is to calculate the area under both curves, where there is an overlap between both probability curves. This will tell us how much consistency there is between the two scenarios (without and with road passages), a value between 0 (no overlap at all) and 1 (complete overlap, the number of crossing animals in each road section is exactly the same in both scenarios).

```
hist(no.passages, breaks = breaks, col = rgb(0, 0, 1, alpha = 0.5), prob = T,
     xlab = 'Road section (km)', main = '',
     ylim = c(0, 0.5), xlim = c(7,16))
hist(passages, breaks = breaks, col = rgb(1, 0, 0, alpha = 0.5), prob = T, add = T)
legend('topright', legend = c('No road passages', 'With road passages', 'We want to calculate this area'),
     fill = c(rgb(0, 0, 1, alpha = 0.5), rgb(1, 0, 0, alpha = 0.5), rgb(0.5, 0, 0.18, alpha = 0.6)))
```



If we would calculate the overlap between continuous probability distributions, we would calculate the area that falls below both curves, using the formula:

$$\int_x \min(p_1(x), p_2(x)) dx \quad (1)$$

where $p_1(x)$ and $p_2(x)$ are the two probability functions. As we're using samples of a distribution, we can approximate that for:

$$\sum_x \min(p_1(x), p_2(x)) \Delta x \quad (2)$$

and Δ can be as small as one wants. We'll do that for our case.

First, we'll count the number of crossing records at each road section (here, road sections of 0.5 km) and each scenario.

```
breaks <- seq(7, 16, 0.5)
(np.tab <- cut(no.passages, breaks) %>% table)
```

```
## .
## (7,7.5] (7.5,8] (8,8.5] (8.5,9] (9,9.5] (9.5,10] (10,10.5]
##      0      1      3     10     11     23     22
## (10.5,11] (11,11.5] (11.5,12] (12,12.5] (12.5,13] (13,13.5] (13.5,14]
##     13      9      5      3      0      0      0
## (14,14.5] (14.5,15] (15,15.5] (15.5,16]
##      0      0      0      0
```

```
(p.tab <- cut(passages, breaks) %>% table)
```

```
## .
## (7,7.5] (7.5,8] (8,8.5] (8.5,9] (9,9.5] (9.5,10] (10,10.5]
##      0      0      0      1      5     13     14
## (10.5,11] (11,11.5] (11.5,12] (12,12.5] (12.5,13] (13,13.5] (13.5,14]
```

```
##          24          18          14          5          3          2          0
## (14,14.5] (14.5,15] (15,15.5] (15.5,16]
##          1          0          0          0
```

Now we'll combine both counts in a `data.frame`, calculate the probability of crossing at each road section, and the calculus of $\min(p_1(x), p_2(x))$ (as in eq. 2 above) for each road section.

```
(df <- data.frame(cbind(np.tab, p.tab)) %>%
  mutate(np.prob = np.tab/sum(np.tab),
         p.prob = p.tab/sum(p.tab),
         overlap_per_road_section = pmin(np.prob, p.prob)))
```

```
##      np.tab p.tab np.prob p.prob overlap_per_road_section
## 1         0     0    0.00  0.00                0.00
## 2         1     0    0.01  0.00                0.00
## 3         3     0    0.03  0.00                0.00
## 4        10     1    0.10  0.01                0.01
## 5        11     5    0.11  0.05                0.05
## 6        23    13    0.23  0.13                0.13
## 7        22    14    0.22  0.14                0.14
## 8        13    24    0.13  0.24                0.13
## 9         9    18    0.09  0.18                0.09
## 10         5    14    0.05  0.14                0.05
## 11         3     5    0.03  0.05                0.03
## 12         0     3    0.00  0.03                0.00
## 13         0     2    0.00  0.02                0.00
## 14         0     0    0.00  0.00                0.00
## 15         0     1    0.00  0.01                0.00
## 16         0     0    0.00  0.00                0.00
## 17         0     0    0.00  0.00                0.00
## 18         0     0    0.00  0.00                0.00
```

Finally, we calculate the total overlap (the area below both curves) summing the overlap column to get the sum in eq (2):

```
sum(df$overlap_per_road_section)
```

```
## [1] 0.63
```

Overlap between ecological corridors simulated in two different scenarios

Now we're going to extend this approach to a two dimensional example. We simulated ecological corridors in two scenarios (considering (a) land use only and (b) land use + urban zoning) to define the permeability of animals moving throughout the landscape. We use the Route Selection Frequency Index of each scenario (RSFI), which measures the number of corridors crossing each pixel in space, and is therefore a proxy for the frequency distribution in 1D shown above. We'll transform this RSFI map into a map of probability of movement (specifically, a probability per unit of area) by dividing the RSFI by the sum of all RSFI pixel values. Then, we'll proceed with the same calculus of eq. 2, but applied to the 2D space.

Let's first load the RSFI maps.

```
# folder where RSFI maps are
corrdir <- '/home/bniebuhr/Documents/00_academico/01_artigos/ms_Pena_et_al_corredores_OuroPreto/corridors'
```

```

# Load data
setwd(corrdir)

# Corridors without urban zoning
files <- list.files('corredores_sem_zona/', pattern = 'RSFI.tif', include.dirs = T)
files_tif <- files[endsWith(files, '.tif')]

corridors_no_zone <- list()
for(i in 1:length(files_tif)) {
  corridors_no_zone[[i]] <- raster(paste0('corredores_sem_zona/', files_tif[i]))
}

# Corridors with urban zoning
files <- list.files('corredores_com_zona_2018_04_d09/', pattern = 'RSFI.tif',
                    include.dirs = T)
files_tif <- files[endsWith(files, '.tif')]

corridors_zone <- list()
for(i in 1:length(files_tif)) {
  corridors_zone[[i]] <- raster(paste0('corredores_com_zona_2018_04_d09/', files_tif[i]))
}

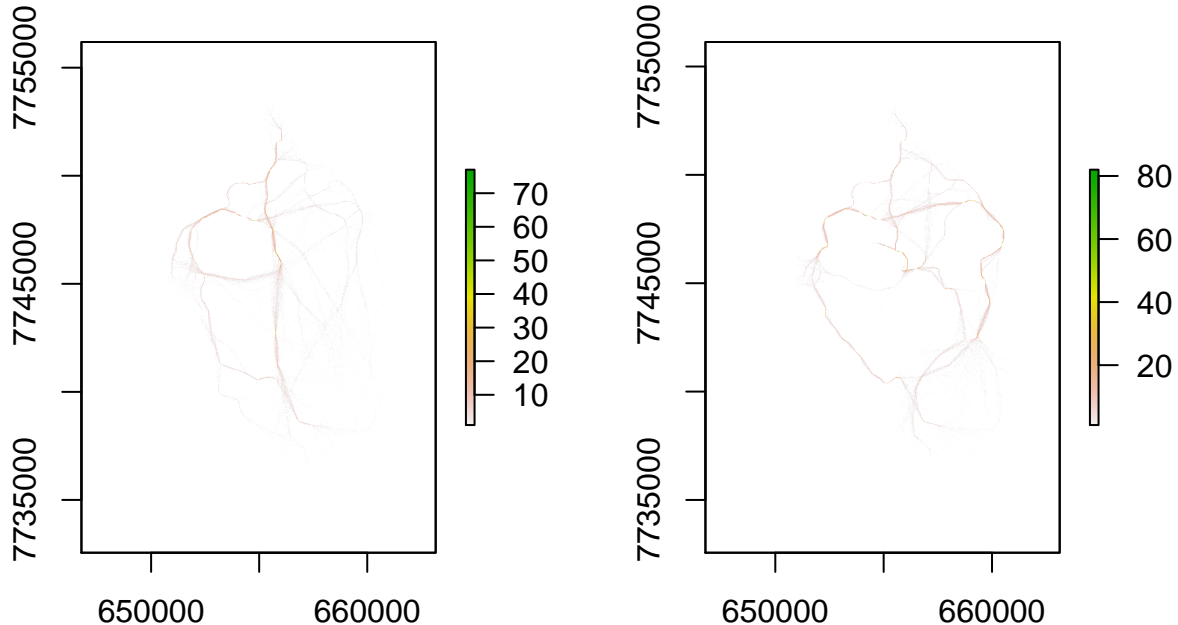
```

Let's compare how the corridors look like in both scenarios for one species, **A. leucophthalmus**.

```

par(mfrow = c(1, 2))
plot(corridors_no_zone[[1]])
plot(corridors_zone[[1]])

```



```

par(mfrow = c(1,1))

```

Now I'll transform both maps into probability maps, as mentioned above:

```

nz.probab <- corridors_no_zone[[1]]/cellStats(corridors_no_zone[[1]], sum)
sum(nz.probab[], na.rm = T) # ok, sum = 1

```

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

```
# plot(nz.prob)
```

```
z.prob <- corridors_zone[[1]]/cellStats(corridors_zone[[1]], sum)  
sum(z.prob[], na.rm = T) # ok, sum = 1
```

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

The next step is to calculate the overlap at each pixel, but before that we have to make sure both maps have exactly the same extent. Finally, we'll sum the values of the overlap layer to get a measure of overlap or consistency between the two scenarios, and we'll also plot that to visualize the locations where there was more overlap.

```
# Force that the no zone probability map has the same extent as the zone map, by extending and cropping  
# its extension.
```

```
nz.prob2 <- crop(extend(nz.prob, z.prob), z.prob)  
#plot(nz.prob2)
```

```
# Calculate the overlap at each pixel
```

```
overlap <- z.prob
```

```
overlap[] <- pmin(nz.prob2[], z.prob[])
```

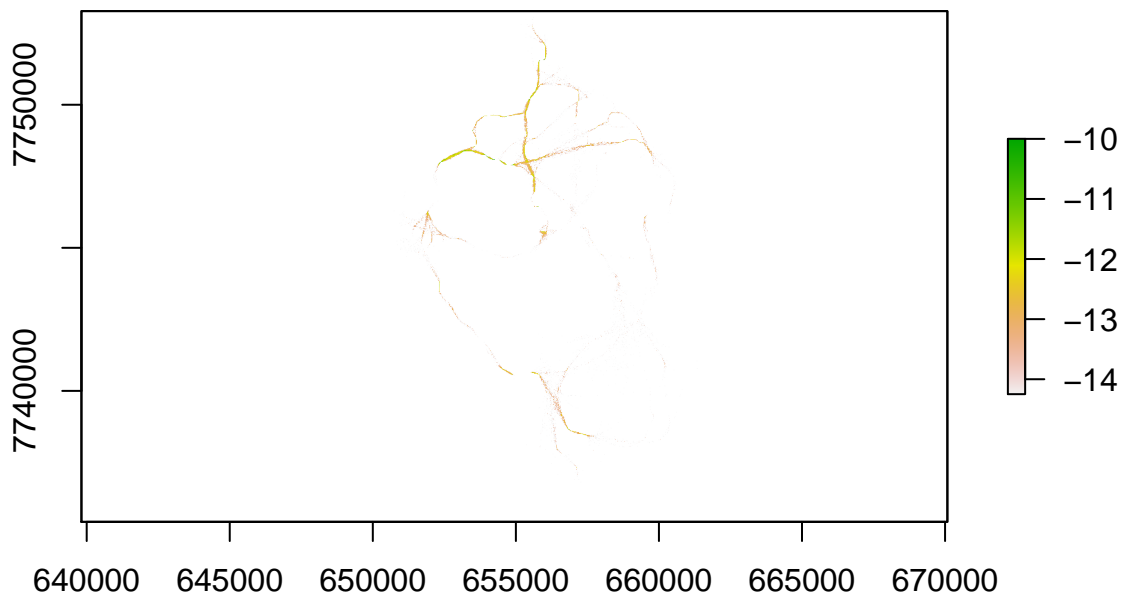
```
# Total overlap value between scenarios
```

```
cellStats(overlap, sum)
```

```
## [1] 0.2877443
```

```
# Plot overlap layer in log scale, since the probability values per unit area are too small.
```

```
plot(log(overlap))
```



Concluding remarks

Preguiça de continuar a escrever em inglês. Rsss. O legal dessa abordagem é que, para cada espécie, podemos ter uma medida de overlap entre os cenários. Porém, também podemos ter uma medida de overlap entre cada par de espécies, e visualizar no mapa os locais com mais overlap.

Daria ainda para calcular o quanto foi “exclusivo” de cada espécie numa abordagem similar, mas eu ainda não pensei em como fazer isso, matematicamente. Mas acho que pode ser possível, se você acharem que vale a pena.