Spatial/diel partitioning of coocurrences

2020-05-28 14:35:50

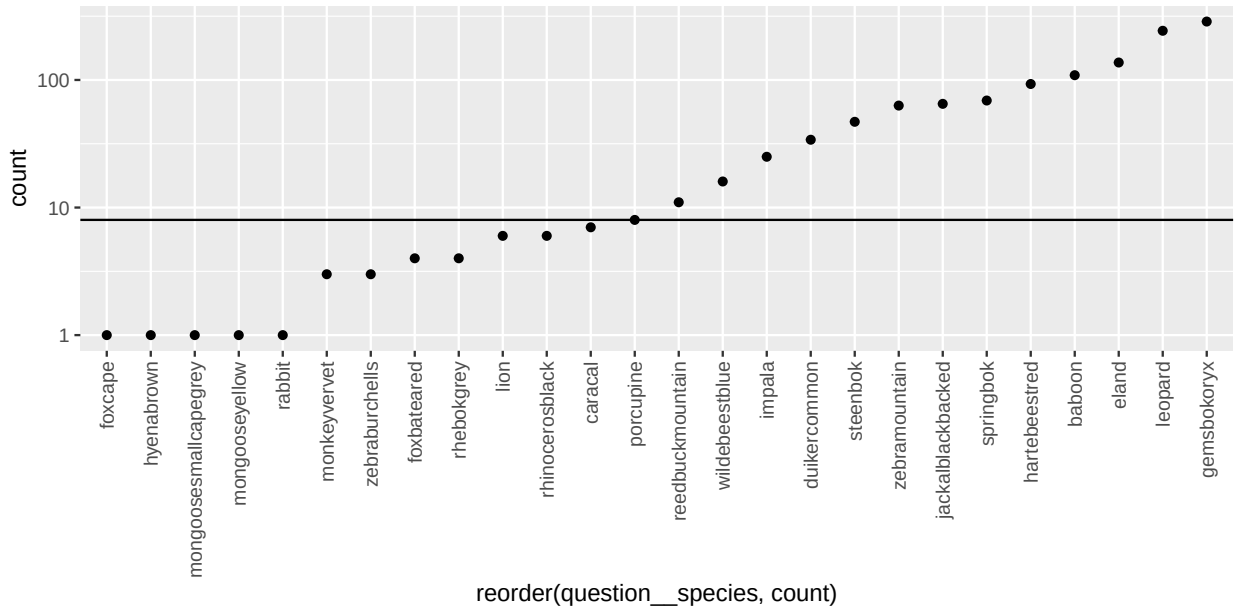
This is the report for file ../data/SAM_sample_data.csv with species filter set to 8.

```
params
```

```
## $file
## [1] "../data/SAM_sample_data.csv"
##
## $thr
## [1] 8
##
## $selcrit
## [1] "BIC"
##
## $durations
## [1] "864000s (~1.43 weeks)"  "1296000s (~2.14 weeks)" "1728000s (~2.86 weeks)"
##
## $codeloc
## [1] "SAM"
##
## $saving_folder
## [1] "output/SAM_thr8_habcov_BIC/"
##
## $use_default_offset
## [1] FALSE
##
## $use_habcov
## [1] TRUE
##
## $use_hour
## [1] FALSE
##
## $hour_slices
## [1] 00:00:00 05:00:00 09:30:00 14:30:00 19:00:00 23:59:59
##
## $spatial_scales
## [1] "cam"          "clust_fine"   "clust_coarse"
##
## $hour_offset
## [1] "none"
```

Import and prepare data

We determine the filtering thresholds as a compromise between data loss and noise. Therefore, we used curves plotted in quality_check.



Retrieve spatial clustering and covariates for our site only:

Extract appropriate clustering and glue it to the dataframe:

- clusters 1

```
## Warning: Column `site_ID` joining character vector and factor, coercing into character vector
```

Also define an appropriate offset because we merged sites and clusters are of inequal size.

```
##      Freq
## SAM1    4
## SAM2    3
## SAM3    4
## SAM4    3
```

- clusters 2

```
## Warning: Column `site_ID` joining character vector and factor, coercing into character vector
```

Also define an appropriate offset because we merged sites and clusters are of inequal size.

```
##      Freq
## SAM1    2
## SAM2    2
## SAM3    2
## SAM4    1
## SAM5    1
## SAM6    2
## SAM7    1
## SAM8    2
## SAM9    1
```

Also, if habitat covariates should be used:

```
##         cov_fac
## SAM_A01 COVSAM1
## SAM_A02 COVSAM2
## SAM_B01 COVSAM1
```

```
## SAM_B02 COVSAM1
## SAM_B03 COVSAM1
## SAM_C02 COVSAM1
## SAM_C03 COVSAM3
## SAM_D01 COVSAM1
## SAM_D03 COVSAM4
## SAM_D04 COVSAM1
## SAM_E01 COVSAM1
## SAM_E02 COVSAM5
## SAM_E03 COVSAM1
## SAM_F02 COVSAM3
```

```
##       clust1
## SAM1 COVSAM1
## SAM2 COVSAM2
## SAM3 COVSAM2
## SAM4 COVSAM3
```

```
##       clust2
## SAM1 COVSAM1
## SAM2 COVSAM2
## SAM3 COVSAM1
## SAM4 COVSAM3
## SAM5 COVSAM3
## SAM6 COVSAM4
## SAM7 COVSAM3
## SAM8 COVSAM3
## SAM9 COVSAM3
```

If we aggregated by hour, spans might be of unequal durations:

```
if(params$use_hour & params$hour_offset == "duration"){
  if(FALSE %in% (params$hour_slices == chron::times(c('00:00:00', '5:00:00', '9:30:00',
                                '14:30:00', '19:00:00','23:59:59')))){
    stop("Specified hour slice is not default, check hour slices names")
  }else{
    hour_ID <- c("19--05", "05--09", "09--14", "14--19")

    # Compute durations
    durations <- params$hour_slices[2:length(params$hour_slices)] -
      params$hour_slices[1:(length(params$hour_slices) -1)]

    # Merge first and last
    durations[1] <- durations[1] + durations[length(durations)]
    durations <- durations[1:(length(durations) - 1)]

    offset_h <- data.frame(hour_ID, durations)

    # Camera offset: easy
    offset_hourcam <- offset_h %>% rename("offset" = 'durations') %>%
      mutate(offset = as.numeric(offset, 'hours')) %>% column_to_rownames("hour_ID")
    offset_hourcam

    # Clusters offsets: multiple cases
    Freq1 <- rep(offset1$Freq, nrow(offset_h))
    clust1 <- rep(rownames(offset1), nrow(offset_h))
```

```
    df1 <- offset_h %>% slice(rep(1:n(), each =  nrow(offset1)))
    df2 <- data.frame(Freq1, clust1)

    offset1 <- bind_cols(df1, df2) %>% mutate(offset = Freq1*durations) %>%
      dplyr::select(-c(Freq1, durations)) %>% rename("cluster_ID" = 'clust1') %>%
      mutate(offset = as.numeric(offset, 'hours'))
    offset1

    cov.1.col <- cov.1 %>% rownames_to_column("cluster_ID")

    # Clusters offsets: multiple cases
    Freq2 <- rep(offset2$Freq, nrow(offset_h))
    clust2 <- rep(rownames(offset2), nrow(offset_h))

    df1 <- offset_h %>% slice(rep(1:n(), each = nrow(offset2)))
    df2 <- data.frame(Freq2, clust2)

    offset2 <- bind_cols(df1, df2) %>% mutate(offset = Freq2*durations) %>%
      dplyr::select(-c(Freq2, durations)) %>% rename("cluster_ID" = 'clust2') %>%
      mutate(offset = as.numeric(offset, 'hours'))
    offset2

    cov.2.col <- cov.2 %>% rownames_to_column("cluster_ID")

  }

}else if(params$hour_offset == "TSS"){
  cov.2.col <- cov.2 %>% rownames_to_column("cluster_ID")
  cov.1.col <- cov.1 %>% rownames_to_column("cluster_ID")
}
```

Spatial and temporal data exploration

We will now infer several networks for which the spatio-temporal aggregation scales differ, from coarser to finer.

With coarse clusters

```
## Warning: Column `cov_ID` joining character vector and factor, coercing into character vector
```

```
## Warning: Column `offset_ID` joining character vector and factor, coercing into character vector
```

```
##
##  Initialization...
##  Adjusting 30 PLN with sparse inverse covariance estimation
##  Joint optimization alternating gradient descent and graphical-lasso
##  sparsifying penalty = 0.9534018     sparsifying penalty = 0.8806295     sparsifying penalty = 0.813
##  Post-treatments
##  DONE!
```

```
## Empty graph
```

```
## Warning: Column `cov_ID` joining character vector and factor, coercing into character vector

## Warning: Column `offset_ID` joining character vector and factor, coercing into character vector

##
##  Initialization...
##  Adjusting 30 PLN with sparse inverse covariance estimation
##  Joint optimization alternating gradient descent and graphical-lasso
##  sparsifying penalty = 1.104336  sparsifying penalty = 1.020043  sparsifying penalty = 0.9421839
##  Post-treatments
##  DONE!

## File saved

## Warning: Column `cov_ID` joining character vector and factor, coercing into character vector

## Warning: Column `offset_ID` joining character vector and factor, coercing into character vector

##
##  Initialization...
##  Adjusting 30 PLN with sparse inverse covariance estimation
##  Joint optimization alternating gradient descent and graphical-lasso
##  sparsifying penalty = 0.4220524     sparsifying penalty = 0.3898375     sparsifying penalty = 0.360
##  Post-treatments
##  DONE!
```

With fine clusters

```
## Warning: Column `cov_ID` joining character vector and factor, coercing into character vector

## Warning: Column `offset_ID` joining character vector and factor, coercing into character vector

##
##  Initialization...
##  Adjusting 30 PLN with sparse inverse covariance estimation
##  Joint optimization alternating gradient descent and graphical-lasso
##  sparsifying penalty = 0.88773   sparsifying penalty = 0.8199703     sparsifying penalty = 0.7573827
##  Post-treatments
##  DONE!

## File saved

## Warning: Column `cov_ID` joining character vector and factor, coercing into character vector

## Warning: Column `offset_ID` joining character vector and factor, coercing into character vector

##
##  Initialization...
##  Adjusting 30 PLN with sparse inverse covariance estimation
##  Joint optimization alternating gradient descent and graphical-lasso
##  sparsifying penalty = 0.7021366     sparsifying penalty = 0.6485431     sparsifying penalty = 0.599
##  Post-treatments
##  DONE!

## File saved

## Warning: Column `cov_ID` joining character vector and factor, coercing into character vector

## Warning: Column `offset_ID` joining character vector and factor, coercing into character vector
```

```
##
##  Initialization...
##  Adjusting 30 PLN with sparse inverse covariance estimation
##  Joint optimization alternating gradient descent and graphical-lasso
##  sparsifying penalty = 1.023789  sparsifying penalty = 0.9456442     sparsifying penalty = 0.873464
##  Post-treatments
##  DONE!
```

```
## File saved
```

By camera

```
##
##  Initialization...
##  Adjusting 30 PLN with sparse inverse covariance estimation
##  Joint optimization alternating gradient descent and graphical-lasso
##  sparsifying penalty = 0.7312269     sparsifying penalty = 0.675413  sparsifying penalty = 0.6238593
##  Post-treatments
##  DONE!
```

```
## File saved
```

```
##
##  Initialization...
##  Adjusting 30 PLN with sparse inverse covariance estimation
##  Joint optimization alternating gradient descent and graphical-lasso
##  sparsifying penalty = 0.9630822     sparsifying penalty = 0.889571  sparsifying penalty = 0.8216708
##  Post-treatments
##  DONE!
```

```
## Empty graph
```

```
##
##  Initialization...
##  Adjusting 30 PLN with sparse inverse covariance estimation
##  Joint optimization alternating gradient descent and graphical-lasso
##  sparsifying penalty = 1.112301  sparsifying penalty = 1.0274    sparsifying penalty = 0.9489791
##  Post-treatments
##  DONE!
```

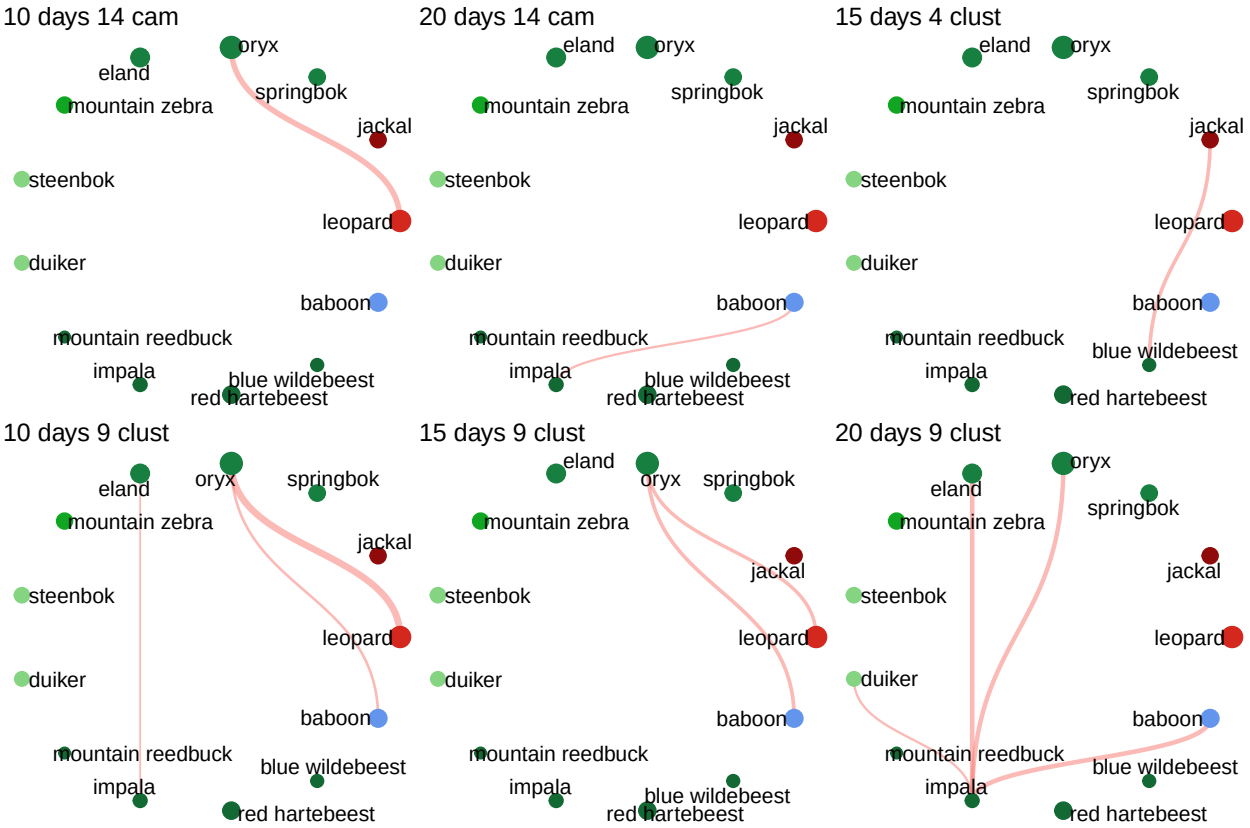
```
## File saved
```

Recap

Saved graphs:

```
## [1] "SAM_cooc_10d_cam.rds"    "SAM_cooc_10d_clust2.rds" "SAM_cooc_15d_clust1.rds" "SAM_cooc_15d_clust
## [5] "SAM_cooc_20d_cam.rds"    "SAM_cooc_20d_clust2.rds"
```

```
## Warning: Column `n`/`spp_mysites` joining factors with different levels, coercing to character vecto
```


10 days 14 cam
oryx
eland
springbok
mountain zebra
jackal
steenbok
leopard
duiker
baboon
mountain reedbuck
impala
blue wildebeest
red hartebeest
20 days 14 cam
eland
oryx
springbok
mountain zebra
jackal
steenbok
leopard
duiker
baboon
mountain reedbuck
impala
blue wildebeest
red hartebeest
15 days 4 clust
eland
oryx
springbok
mountain zebra
jackal
steenbok
leopard
duiker
baboon
mountain reedbuck
blue wildebeest
impala
red hartebeest
10 days 9 clust
eland
oryx
springbok
mountain zebra
jackal
steenbok
leopard
duiker
baboon
mountain reedbuck
blue wildebeest
impala
red hartebeest
15 days 9 clust
eland
oryx
springbok
mountain zebra
jackal
steenbok
leopard
duiker
baboon
mountain reedbuck
impala
blue wildebeest
red hartebeest
20 days 9 clust
oryx
eland
springbok
mountain zebra
jackal
steenbok
leopard
duiker
baboon
mountain reedbuck
blue wildebeest
impala
red hartebeest