# Spatial analyses tasks

Søren Faurby

23/9/2022

Dependencies:

```
library(fasterize)
library(geosphere )
library(ggplot2)
library(MCMCglmm)
library(ncf)
library(phytools)
library(raster)
library(sf)
library(spatialreg)
library(spdep)
library(wiqid)
```

Today we will be pretending to do real dataanalyses as we often would do for scientific studies. This at times means a lot of computational time. R by default only uses one processor and it may be beneficial to start thinking and potentially do the first analyses for the next exercises in another R window whenever you are waiting long for a result. Alternatively we will employ a "tv-kitchen approach" and whenever you are waiting very long for a result, call us and we will give you a file as they would look like if created by the code.

## 1 Test Bergmanns rule on pidgeons

For this task you should try to incorporate an additional spatial predictor into the analyses of pidgeons from yesterday. More specifically you should test if it changes the conclusions on the analyses of the relationship between tarsus and body size if you include mean annual temperature for all species into the analyses.

For this you first have to download the ranges of all pigeons. This can be done by going to the IUCN download page and clicking though Animalia -> Chordata -> Aves before finally ticking Columbiformes. Load the polygons into R and transform it to cylindrical equal area. You should not transform it to spatial using "as(..., 'Spatial') as you did last time since we want the data in the sf format for later.

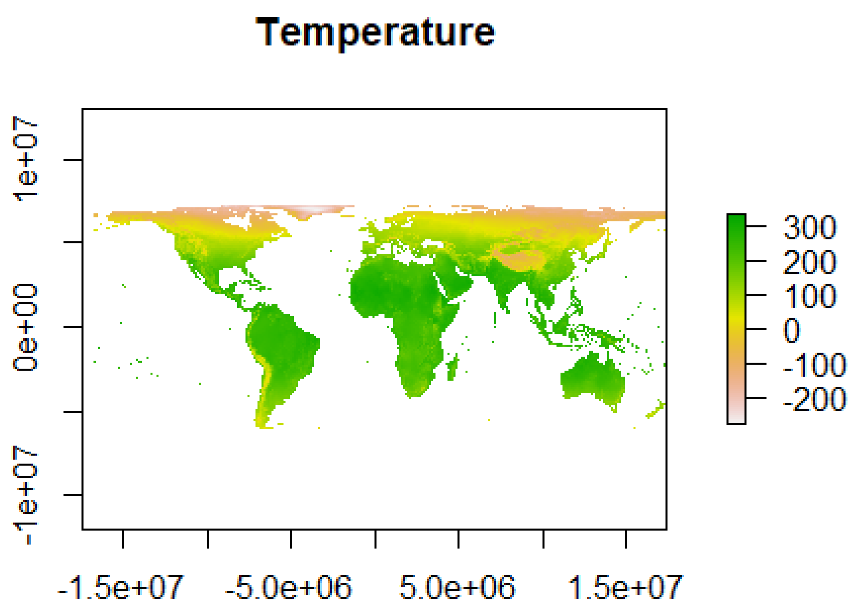
```
## Reading layer `data_0' from data source
`C:\\Users\\xfauso\\Documents\\Teaching\\2022\\PhDR\\Other_data\\Pidgeon\\dat
a_0.shp' using driver `ESRI Shapefile'
## Simple feature collection with 484 features and 15 fields
```

```
## Geometry type: MULTIPOLYGON
## Dimension:      XY
## Bounding box:   xmin: -179.999 ymin: -55.72332 xmax: 179.999 ymax: 70.13647
## Geodetic CRS:   WGS 84
```

Data from IUCN has a whitespace between genus and species name. The dataset from yesterday has an underscore instead. We can change the whitespaces to underscores (I am here calling the object with pidgeon ranges "Polygon")

```
Polygon$BINOMIAL=gsub(" ", "_", Polygon$BINOMIAL)
```

You now need to calculate mean annual temperature within the range of each species. This is easiest done by first creating a raster of annual temperature in a cylindrical equal area projection. You can use a Behrmann projection as we have used earlier in the week but in a higher resolution of 0.2 by 0.2 degree equivalent. This is the very similar to what you did Wednesay




You can check the dimensions of the raster

```
dim(Proj_Temp)

## [1]  710 1800    1
```

You should now load the data you used yesterday (I am here calling the pigeon dataset DATA) and create a new column called "Temperature" which should be replaced with the median temperature for each species. Below I will highlight how this can be done for the first species. You can do this for the rest using a for loop. We will rasterize using the package fasterize instead of raster as we used earlier since fasterize is substantially faster.

```
library(fasterize)
i=1
Bird_Raster=fasterize(Polygon[which(Polygon$BINOMIAL==DATA$species[i]),],
Proj_Temp) #Create a raster for the species. It takes the value 1 for a cell
where it is found in the midpoint and NA for all other cells
Temp_Values=values(Bird_Raster*Proj_Temp) #Since Bird_Raster takes the value
1 for cells with the bird and NA for others, this raster is identical to
temperature for cells with the bird and NA for all other cells.
DATA$Temperature[i]=median(Temp_Values, na.rm=T)/10 #The units in the raster
were as we have discussed before in degrees times 10. We therefore divide by
ten here to get units in degrees.
```

Now you just need to set this up as a for loop across all species.

If successful you should get something like this.

```
summary(DATA$Temperature)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  -1.095  21.648  24.774  22.399  25.488  27.818       1
```

As you will see the procedure gave NA for a single species. This is because this species the socorro dove is endemic to a tiny island. The rasterization procedure only looks at cells where the midpoint of the cell is within the range of a specis. The island the socorro dove is found in does not cross any such midpoints. You can instead identify the temperature of the centroid of the species polygon using extract() from the library raster

```
library(geosphere )
NA_SPECIES=which(is.na(DATA$Temperature))
CENTROID=centroid(as(Polygon[which(Polygon$BINOMIAL==DATA$species[NA_SPECIES]
),], "Spatial"))
extract(Proj_Temp, CENTROID)

## [1] 240.6872
```

Replace the NA with this value and scale temperautre to have a mean of 0 and a standard deviation of 1.

While not directly related to the topic here, this is also a good chance to look at issues of taxonomy. Many sources use different taxonomy as we discussed Tuesday and we generally cannot know if the size measurements are using the same taxonomy as IUCN. There is in fact at least one case where we can nearly guarantee that they did not. The Nicobar Imperial-pigeon (*Ducula nicobarica*) were very recently split from the Green Imperial-pigeon (*Ducula aenea*) and the measurements with near guarantee treated them as the same species. We will ignore this issue here and just note that if we wanted to address it, the easiest way to do so would be to change the IUCN polygons names to match the taxonomy of the measurements.

```
Polygon$BINOMIAL[Polygon$BINOMIAL=="Ducula_nicobarica"]="Ducula_ aenea"
```

We will not do so here because doing so properly would require a systematic comparisons of the taxonomy used between sources. A partial correction as with the line above may at least theoretically create biased results even if they are somewhat more precise than the non-corrected.

Now the goal is testing if the temperature that pidgeons live in influences tarsus size by building and asessing an MCMCglmm model with “foraging”, “log(body.g)” and “Temperature” as main effects and phylogeny and measurement as random effects. Below I have called this model “Model_T”. I have here just used the defaults priors, if you set priors as you did Wednesday you will get marginally different results.

We can look at the results which should look something like this.

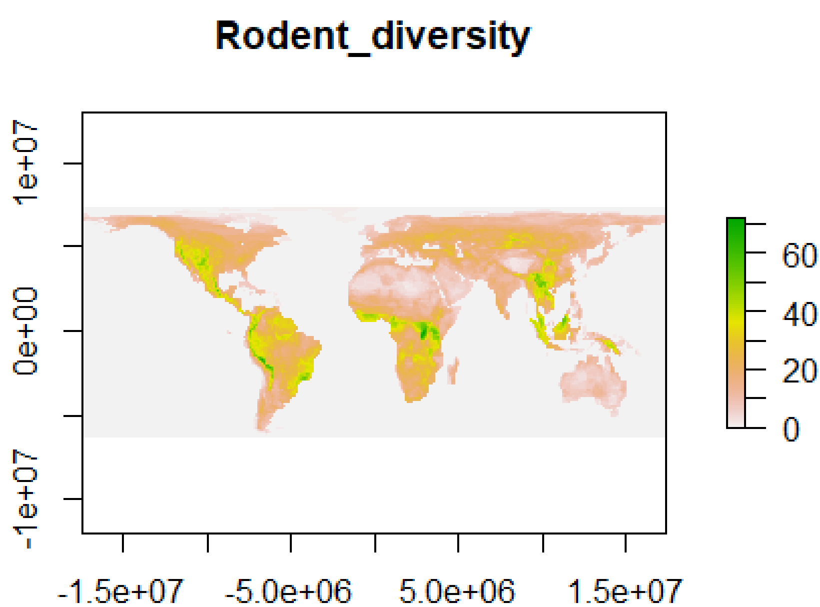
```
summary(Model_T)

##
##  Iterations = 101:1091
##  Thinning interval  = 10
##  Sample size  = 100
##
##  DIC: -1154.285
##
##  G-structure:  ~animal
##
##        post.mean l-95% CI u-95% CI eff.samp
## animal   0.06588  0.05182  0.08808    70.33
##
##                ~measure
##
##         post.mean  l-95% CI u-95% CI eff.samp
## measure   0.01051 6.951e-17  0.04365      100
##
##  R-structure:  ~units
##
##       post.mean  l-95% CI u-95% CI eff.samp
## units  0.000846 1.091e-07 0.006045    11.08
##
##  Location effects: log(tarsus.mm) ~ foraging + log(body.g) + Temperature
##
##                        post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)              0.64742  0.24963  0.91127    100.0 <0.01 **
## foragingterrestrial      0.25317  0.15593  0.34364    100.0 <0.01 **
```

```
## log(body.g)           0.45905  0.40815  0.52276    100.0 <0.01 **
## Temperature           0.02803 -0.01182  0.07884    151.8  0.14
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

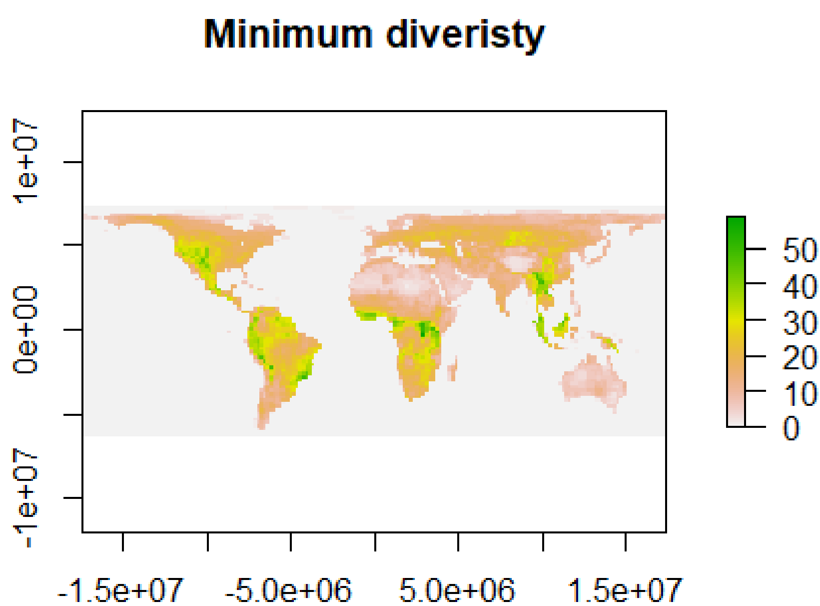
## 2 Predictors of rodent diversity

For the task, you should investigate predictors of the diversity of rodents across the globe. More specifically you should investigate using a SAR regression with an optimal neighborhood if the diversity of rodents within each grid cells is best explained by precipitation, the difference in elevation within the cell or both. You can calculate the diversity of rodents based on the present-natural data from Phylcacine.If you do this correctly you should get something looking like this.

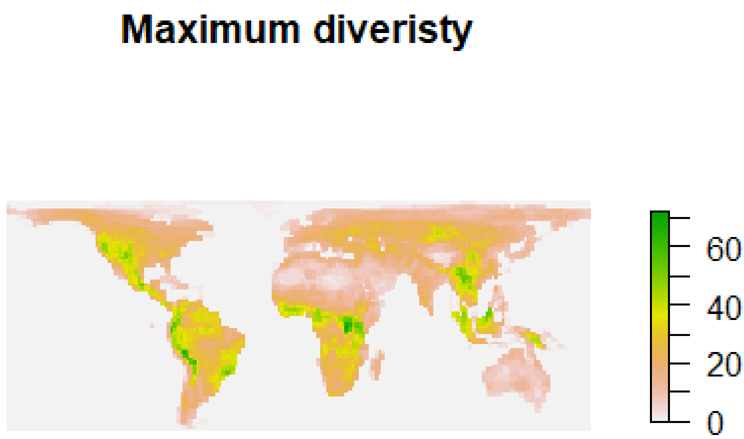



For precipitation you should process the data similar to the process of temperature from yesterday with one main difference. For precipitation it generally makes most sense to analyze log (precipitation) rather than raw precipitation (I briefly touched upon this at my presentation yesterday). Since there is a few cells without any precipitation add 1 to each cell before log transforming. The input data for elevation variation can be found here. **More specifically I have used the file wc2.1_2.5m_elev.tif, which is the file listed under a 2.5 minute resolution. This is a decent trade-off between computational speed and acuracy. Had this been for a paper purpose you could have considered using the finer scale 30 second one.** You need to do a little work to get the elevational range within each cell. The easiest way to deal with this utilizes the fact that the aggregate function can aggregate based on multiple different functions. Similar to precipitation it it desirable to log-transform this variable after it is calculated. So far we have only been aggregating based on mean but you can also aggregate based to find the smallest value or the largest value in a the cells you aggregate". To see how this works we can aggreagte to find the smallest and largest diversity of rodents while aggregating by a factor of 2 (note that I here refer to a *Diversity_raster*, which was the name I gave to the diversity of rodents I calculated above, if you want to redo this you may need to change the name).

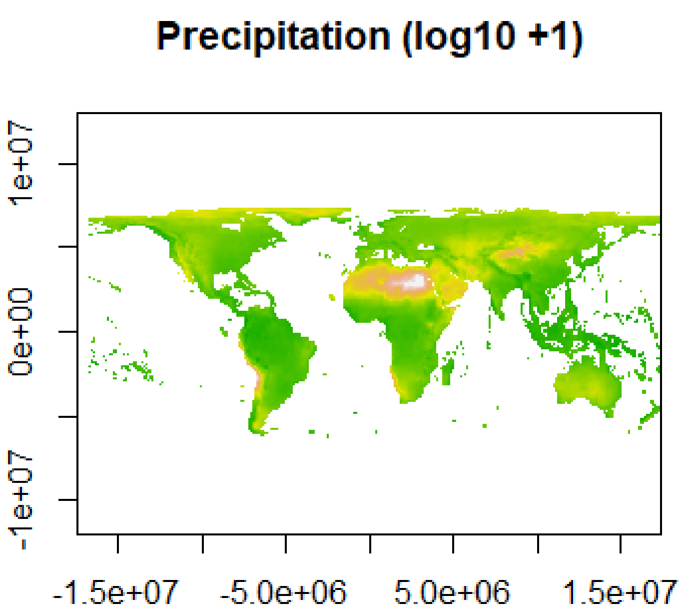
```
Min_Div=aggregate(Diversity_raster, 2, min)
plot(Min_Div, main="Minimum diveristy")
```



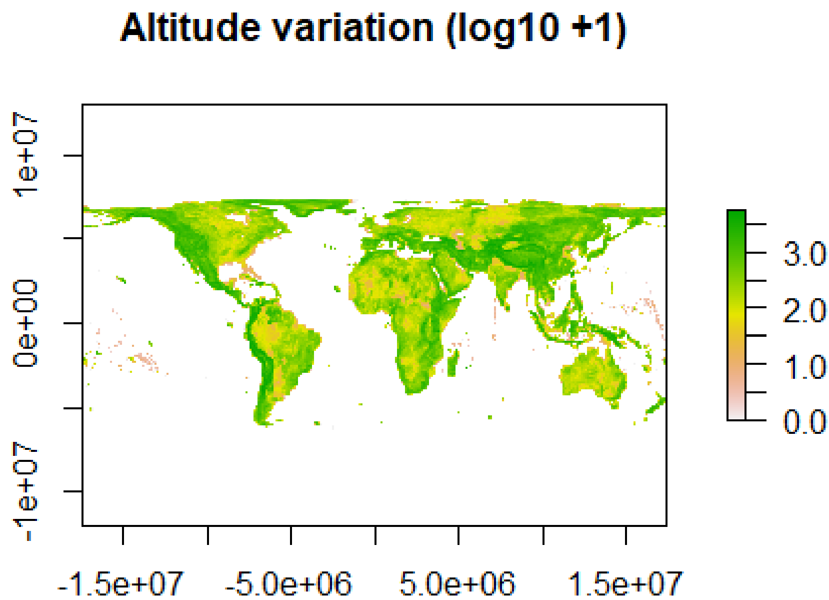

```
Max_Div=aggregate(Diversity_raster, 2, max)
plot(Max_Div, main="Maximum diveristy", axes=F,box=F)
```




Now you have all the skills to produce raster for precipitaion and altitude_variaiton. If you succeed precipitation (log-transformed) should look something like this. (For precipitation it is likely desirable to log-transform data before aggregation).



And altitude variaiton (log-transformed) should look something like this. You should again add 1 to before logtransforming becuase there might be cells without elevational variation-.




And now you have all the data needed to to a SAR regression based on what you learned yesterday but testing both predictors as once. To reduce time you can use a subample of 1000 cells again. If you succeed you should get a model and a null-model similar to this (exact results will vary slightly since you will analyze different sets of cells than me)

```
summary(sar_model, Nagelkerke = TRUE)

##
## Call:spatialreg::errorsarlm(formula = formula, data = data, listw = listw,
##     na.action = na.action, Durbin = Durbin, etype = etype, method =
method,
##     quiet = quiet, zero.policy = zero.policy, interval = interval,
##     tol.solve = tol.solve, trs = trs, control = control)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -2.7656742 -0.2424164  0.0044875  0.2459231  2.2872169
##
## Type: error
## Coefficients: (asymptotic standard errors)
##              Estimate Std. Error z value  Pr(>|z|)
## (Intercept) -0.162550   0.079824 -2.0364   0.04171
## Precip       0.283289   0.038478  7.3623 1.807e-13
## Alt          0.296693   0.019744 15.0267 < 2.2e-16
```

```
##
## Lambda: 0.79903, LR test value: 835.27, p-value: < 2.22e-16
## Asymptotic standard error: 0.017684
##     z-value: 45.185, p-value: < 2.22e-16
## Wald statistic: 2041.7, p-value: < 2.22e-16
##
## Log likelihood: -826.0862 for error model
## ML residual variance (sigma squared): 0.25709, (sigma: 0.50704)
## Nagelkerke pseudo-R-squared: 0.68053
## Number of observations: 1000
## Number of parameters estimated: 5
## AIC: 1662.2, (AIC for lm: 2495.4)
```

```
summary(sar_model_NULL, Nagelkerke = TRUE)
```

```
##
## Call:spatialreg::errorsarlm(formula = formula, data = data, listw = listw,
##     na.action = na.action, Durbin = Durbin, etype = etype, method =
method,
##     quiet = quiet, zero.policy = zero.policy, interval = interval,
##     tol.solve = tol.solve, trs = trs, control = control)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -2.7945515 -0.2482898 -0.0097259  0.2341951  2.9329672
##
## Type: error
## Coefficients: (asymptotic standard errors)
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.172436   0.091791 -1.8786   0.0603
##
## Lambda: 0.8019, LR test value: 887.46, p-value: < 2.22e-16
## Asymptotic standard error: 0.017486
##     z-value: 45.859, p-value: < 2.22e-16
## Wald statistic: 2103.1, p-value: < 2.22e-16
##
## Log likelihood: -952.8967 for error model
## ML residual variance (sigma squared): 0.33065, (sigma: 0.57503)
## Nagelkerke pseudo-R-squared: 0.5883
## Number of observations: 1000
## Number of parameters estimated: 3
## AIC: 1911.8, (AIC for lm: 2797.3)
```

## 3 Test if climatic generalists have different tarsus size than climatic specialists.

For this last task you should see the tarsal length it influenced by climatic heterogeneity (i.e. if climatic generalists are consistently different from climatic specialists). You should estimate this based on the variaiton of the (log-transformed) annual precipitation they are found in. Use the same resolution of the raster as you did for median temperature. **This should not be based the difference between minimum and maximum becuase this is**

**logically connected to the range size but should instead be based on the difference bteewen the 10 and 90% quantiles.** In R these quantiles for a vector *V* can be calcualted as quantile(V, probs =c(0.1, 0.9), type=8). The underlying effect of differnet algoritms for quantiles which are specified with R can be seen in the helpfile which you get by writing help(quantile) and is also discussed in the appendix of one of my recent papers. You should be able to do this on you have learned so far.

If successful calling summary on the climate variation should be something like this (after you have scaled it to have a mean of 0 and a sd of 1)

```
summary(DATA$Climate_Var)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
## -1.4057 -0.6104 -0.3669  0.0000  0.4201  3.9474
```

If successful you should get something like this. Again this is based on default priors and your results will be marginally different if you set priors as you did yesterday.

```
summary(Model_P)

## 
##  Iterations = 101:1091
##  Thinning interval  = 10
##  Sample size  = 100 
## 
##  DIC: -9.972603 
## 
##  G-structure:  ~animal
## 
##        post.mean l-95% CI u-95% CI eff.samp
## animal   0.01304 0.002593  0.03954    15.28
## 
##                ~measure
## 
##          post.mean  l-95% CI u-95% CI eff.samp
## measure  0.006871 5.446e-15  0.03564      100
## 
##  R-structure:  ~units
## 
##       post.mean l-95% CI u-95% CI eff.samp
## units   0.04479  0.02019  0.06797    31.93
## 
##  Location effects: log(tarsus.mm) ~ foraging + log(body.g) + Climate_Var - 
1 
## 
##                       post.mean l-95% CI u-95% CI eff.samp pMCMC   
## foragingarboreal        0.67811  0.38083  1.06031    100.0 <0.01 **
## foragingterrestrial     0.98271  0.66206  1.25576    100.0 <0.01 **
## log(body.g)             0.44567  0.38357  0.50547    141.5 <0.01 **
## Climate_Var            -0.09072 -0.14238 -0.05642    100.0 <0.01 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```