Using the sampbias R package

#Installing sampbias To install the *sampbias* R-package, you can either download the latest build version from gitHub use the `devtools` package:

```
require(devtools)
install_github(repo = "azizka/sampbias")
library(sampbias)
library(maptools)
library(rgdal)
library(raster)
```

Input data

Sampbias calculates spatial bias in a species occurrence data set based on two input files:

1. A table of Species occurrence records
2. A set of geographical gazetteers

The example files for this tutorial are provided in the example_data folder and should be copied to the R working directory, before starting. You can find descriptions and help for all functions by typing `?Functionname`, for instance `?calculate_bias`.

Species occurrence records

Species occurrences can be provided to *sampbias* as an R `data.frame`, which can be loaded from any table format .txt file. The minimum input file needed to run *sampbias* is a table with species occurrences including three columns named "species", "decimallongitude", "decimallatitude". *Sampbias* can also directly work with data downloaded from the Global Biodiversity Information Facility data portal (GBIF). We will use the records of mammals from Borneo provided in the sampbias folder (GBIF, 2016) as example file throughout the tutorial. The file is also included as example data set in the package.

Text files can easily be loaded into R as data.frame using the `read.csv()` function.

```
#loading a text file to R
occ <-read.csv(system.file("extdata",
                           "mammals_borneo.csv",
                           package="sampbias"),
               sep = "\t")
```

Note: *Sampbias* evaluates the bias by comparing to a random sampling, meaning that the tool is not designed for single species data sets, as distribution of the records might then reflect ecological preferences, but rather for multi-species data sets. In general, the more species and the more records, the more reliable the results will be.

Geographic gazetteers

Sampbias evaluates the distribution of the sampled species occurrences in relation to geographic features that might bias sampling effort. These are generally related to accessibility or means of travel. *Sampbias* includes a set of default gazetteers for cities, airports, roads and rivers (Natural Earth, 2016), which can give fair estimates of bias for large and medium scale analyses. These are used by default, if no other gazetteers are

provided by the user. However, these defaults include major features only and if available high resolution user-provided gazetteers are preferable.

Any number of gazetteers can be provided to *sampbias* via the `gaz` argument as a list of `SpatialPointsDataFrame` and `SpatialLinesDataFrame` objects. These objects can easily be loaded into R from standard shape files using the `maptools` package:

```
cit <- readOGR(dsn = system.file("extdata", package="sampbias"),
               layer = "Borneo_major_cities")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "C:\Users\az64mycy\Dropbox (iDiv)\research_projects\16_spatial_bias_package\sampbias\inst\ex
## with 24 features
## It has 4 fields
```

```
roa <- readOGR(dsn = system.file("extdata", package="sampbias"),
               layer = "Borneo_major_roads")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "C:\Users\az64mycy\Dropbox (iDiv)\research_projects\16_spatial_bias_package\sampbias\inst\ex
## with 301 features
## It has 3 fields
```

```
gazetteers <- list(cities = cit,
                   roads = roa)
```

See here and `?SpatialPointsDataFrame` or `?SpatialLinesDataFrame` on how to create a Spatial Objects from tables of coordinates.

Running an analysis

A sampbias analyses is run in one line of code via the `calculate_bias` function:

```
bias.out <- calculate_bias(x = occ, gaz = gazetteers)
```

In addition to the input from above, `calculate_bias` offers a set of options to customize analyses, of which the most important ones are shown in Table 1. See `?calculate_bias` for a detailed description of all options.

Option	Description
res	the raster resolution for the distance calculation to the geographic features and the data visualization in decimal degrees. The default is to one degree, but higher resolution will be desirable for most analyses. Res together with the extent of the input data determine computation time and memory requirements.
restrict_sample	logical. a SpatialPolygons or SpatialPolygonDataframe. If provided the area for the bias test will be restricted to raster cells within these polygons (and the extent of the sampled points in x). Make sure to use adequate values for res. Default = NULL.
terrestrial	logical. If TRUE, the empirical distribution (and the output maps) are restricted to terrestrial areas. Default = TRUE.

The default MCMC runs for 100,000 generations with a 20% burn-in, which has proven sufficient for most analyses. We suggest to verify that the effective sample size of the posterior estimates is large enough (e.g. > 200) using the `ESS` function of the `LaplacesDemon` package (`LaplacesDemon::ESS(bias.out$bias_estimate)`).

Output

The output of `calculate_bias` is a list of different R objects.

Object	Description
summa	A list of summary statistics for the sampbias analyses, including the total number of occurrence points in x, the total number of species in x, the extent of the output rasters as well as the settings for res, binsize, and restric_area used in the analyses.
occurrences	a raster indicating occurrence records per grid cell, with resolution res.

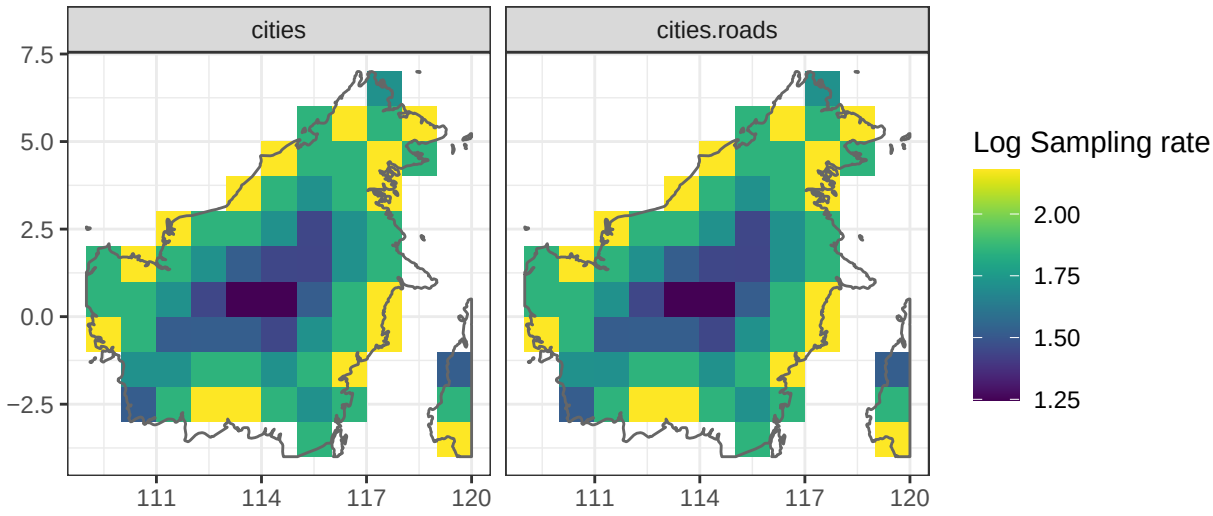
The package includes summary and plot methods for an easy exploration of the results:

```
summary(bias.out)
plot(bias.out)
```

The plot method generates a boxplot of the posterior estimates of the weights for each biasing factor.

As the last step, it is possible to project the bias effects into space and amp them, to identify areas with particular high bias, for instance, to design future field campaigns.

```
proj <- project_bias(bias.out)
map_bias(proj)
```



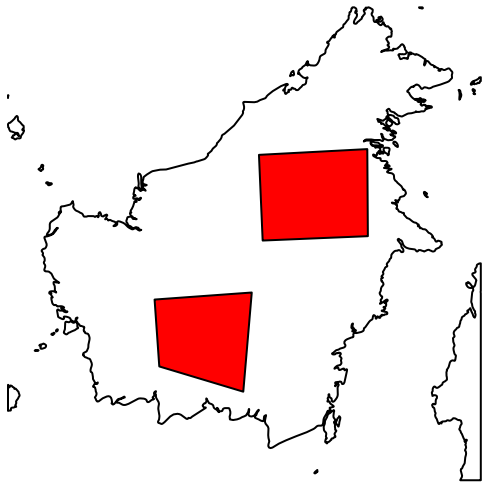
Major modification to a standard sampbias analyses

Default study extent

By default sampbias uses a rectangle defined by the minimum and maximum longitude and latitude values in the input point records to define the study area. It may be desirable to change this, for instance if this rectangle comprises a set of very different habitats for instance rainforest and desert. Samp bias enables a user-defined study area via the `restrict_sample` option of the calculate_bias function. This option takes any object of the class , or SpatialPolygons. This could be simple custom polygons as provided as example in the area_example dataset:

```
data(area_example)
borneo <- crop(sampbias::landmass, extent(108, 120, -5, 7))

plot(borneo)
plot(area_example, col = "red", add = TRUE)
```

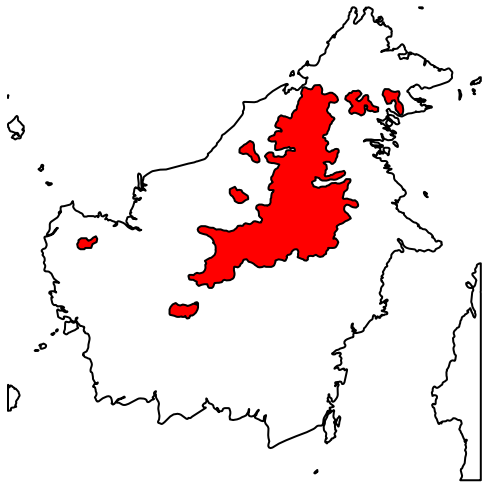


```
bias.out <- calculate_bias(x = occ,
                           gaz = gazetteers,
                           restrict_sample = area_example)
```

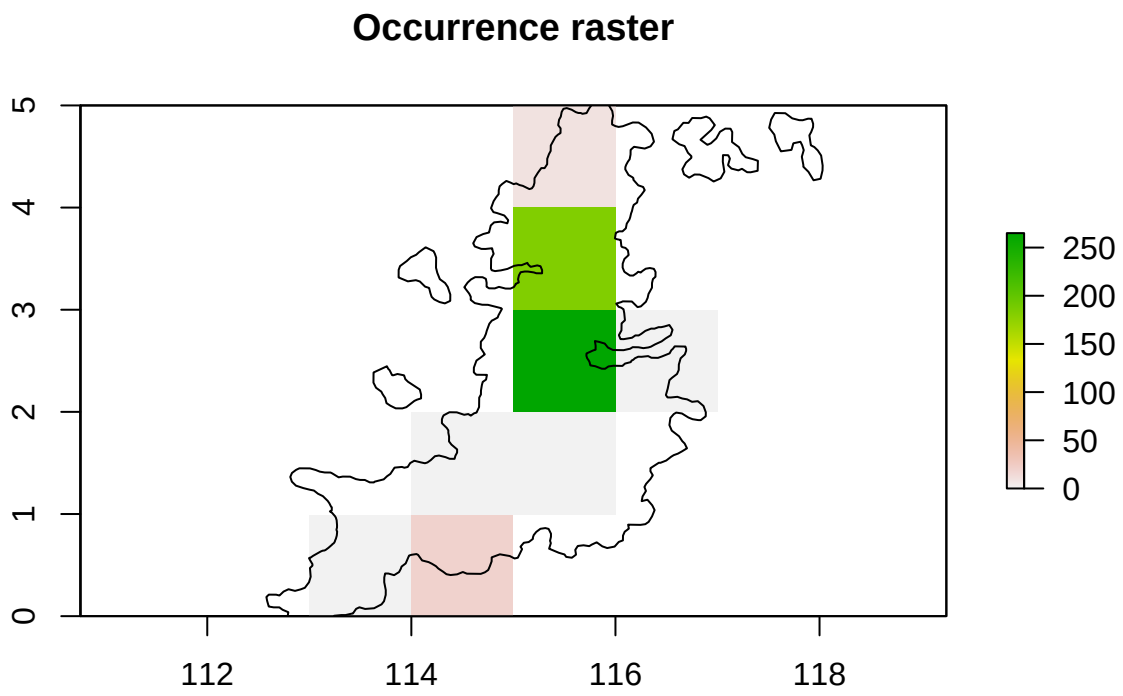
Note the rough resolution of the raster, for a full analyses a much higher resolution will be needed. More complex restrictions are possible, for instance limiting the study area to the ecoregion "Borneo montane rain forests" (Olson, et al. 2001). For diagnostics the sample raster can be plotted using the `plot_raster` argument.

```
data(ecoregion_example)

plot(borneo)
plot(ecoregion_example, col = "red", add = TRUE)
```



```
bias.out <- calculate_bias(x = occ,
                           gaz = gazetteers,
                           restrict_sample = ecoregion_example,
                           plot_raster = TRUE)
```



Equal area rasters

By default sampbias uses a lat/lon projected raster. This is a reasonable approximation for local and regional studies. However, since the area of the cells in such a raster differs with longitude, for large-scale studies an equal area projected raster—for instance using a cylindrical behrmann projection—is a better choice. When using an equal area raster, it is necessary to reproject the input occurrence records as well as the gazetteer files to the new coordinate reference system.

```
# an example for an equal area raster
data(ea_raster)

# reproject the occurrence coordinate
## select coordinates from the occ data.frame and create spatial object
ea_occ <- SpatialPoints(occ[,c(3,2)],
                        proj4string = CRS("+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs"))
## transform to the same CRS as the raster
ea_occ <- spTransform(ea_occ, CRSobj = proj4string(ea_raster))
## retransform into a data.frame
ea_occ <- data.frame(species = occ[,1], coordinates(ea_occ))

# reproject gazetteers
## set the CRS in case it is not defined. Make sure to know the correct CRS.
proj4string(gazetteers[[1]]) <- proj4string(gazetteers[[2]]) <- CRS("+proj=longlat +ellps=WGS84 +datum=

#transform to the new crs
ea_gaz <- lapply(gazetteers, "spTransform", CRSobj = proj4string(ea_raster))
```

```r
# run sampbias
ea_bias <- calculate_bias(x = ea_occ,
                          gaz = ea_gaz,
                          inp_raster = ea_raster)
summary(ea_bias)
```

```
## Number of occurences:  6262
## Raster resolution:  1
## Convexhull:
## Geographic extent:
## class      : Extent
## xmin       : -17367529
## xmax       : 17367529
## ymin       : -6356742
## ymax       : 7348382
## Bias weights:
##          bias_weight      std_dev
## w_cities 0.009231803 0.0002078387
## w_roads  0.001314693 0.0003980419
```