

# Package ‘rAvis’

August 29, 2016

**Encoding** UTF-8

**Type** Package

**Title** Interface to the Bird-Watching Dataset Proyecto AVIS

**Version** 0.1.4

**Date** 2015-06-20

**BugReports** <https://github.com/ropensci/rAvis/issues>

**Author** Javier González Hernández  
<javigzz@yahoo.es>, Sara Varela <svarela@paleobiogeography.org>

**Maintainer** Sara Varela <svarela@paleobiogeography.org>

**Description** Interface to <<http://proyectoavis.com>> database.  
It provides means to download data filtered by species, order,  
family, and several other criteria. Provides also basic functionality to  
plot exploratory maps of the datasets.

**License** GPL-2

**URL** <https://github.com/ropensci/rAvis>

**LazyData** true

**Imports** stringr, XML, RCurl, scrapeR, gdata, scales, rgdal, maptools,  
raster, sp, tools

**Suggests** testthat, roxygen2

**Depends** R (>= 2.10)

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2015-06-22 01:44:46

## R topics documented:

avisAllSpecies . . . . .	2
avisContributorAggregatedObservations . . . . .	3
avisContributorsSummary . . . . .	3

avisHasSpecies . . . . .	4
avisMap . . . . .	5
avisMapSpecies . . . . .	5
avisQuery . . . . .	6
avisQueryContributor . . . . .	8
avisQuerySpecies . . . . .	9
avisSetup . . . . .	9
avisSpeciesId . . . . .	10
avisSpeciesSummary . . . . .	10
canarias . . . . .	11
peninsula . . . . .	11
rAvis . . . . .	12
ravisUTMLatLong . . . . .	13
ravis_credentials . . . . .	13
ravis_shape_spain . . . . .	13
<b>Index</b>	<b>14</b>

---

avisAllSpecies	<i>avisAllSpecies</i>
----------------	-----------------------

---

## Description

Returns a vector with the ids of the species in Proyecto AVIS

## Usage

```
avisAllSpecies()
```

## Value

returns a vector

## Note

This functions does not allow arguments

## Examples

```
## Not run:
avisAllSpecies()

## End(Not run)
```

---

```
avisContributorAggregatedObservations
  avisContributorAggregatedObservations
```

---

**Description**

A function to download the information about the observations of a birdwatcher.

**Usage**

```
avisContributorAggregatedObservations(contributor_id)
```

**Arguments**

`contributor_id` a number setting the id of the birdwatcher (see `avisContributorSummary`)

**Value**

This function returns a dataframe

**Examples**

```
# Explore the contributions of Colectivo Ornitologico Ciguena Negra
## Not run:
avisContributorAggregatedObservations (370)

## End(Not run)
```

---

```
avisContributorsSummary
  avisContributorsSummary
```

---

**Description**

Returns a table with the observations aggregated by birdwatcher.

**Usage**

```
avisContributorsSummary()
```

**Value**

This function returns a matrix

**Note**

This function does not allow arguments

## Examples

```
## Not run:
birdwatchers<- avisContributorsSummary()
par (mfrow =c(2,2))
plot (birdwatchers[,2],birdwatchers[,3], xlab=colnames (birdwatchers)[2],
ylab=colnames (birdwatchers)[3], pch=19)
plot (birdwatchers[,2],birdwatchers[,4], xlab=colnames (birdwatchers)[2],
ylab=colnames (birdwatchers)[4], pch=19)
plot (birdwatchers[,2],birdwatchers[,5], xlab=colnames (birdwatchers)[2],
ylab=colnames (birdwatchers)[5], pch=19)
plot (birdwatchers[,2],birdwatchers[,6], xlab=colnames (birdwatchers)[2],
ylab=colnames (birdwatchers)[6], pch=19)

## End(Not run)
```

---

avisHasSpecies	<i>avisHasSpecies</i>
----------------	-----------------------

---

## Description

check if a species name exists in Proyecto AVIS.

## Usage

```
avisHasSpecies(nameraw)
```

## Arguments

nameraw            scientific name of the species (e.g. "Pica pica")

## Value

Logical: returns TRUE for species with observations in the database and FALSE otherwise

## Examples

```
## Not run:
avisHasSpecies("Pica pica")
avisHasSpecies("Pica pic")

## End(Not run)
```

---

avisMap	<i>Renders a map for the observations provided in 'obs'</i>
---------	-------------------------------------------------------------

---

**Description**

This function should be used with `avisQuerySpecies`, to set a particular query (with or without filters) and get the observations that we want to map. It just allow to map one species. See `avisMapSpecies` for multiple maps.

**Usage**

```
avisMap(obs, label = "", maptype = "admin")
```

**Arguments**

obs	set of observations returned by any of the <code>avisQueryXXX</code> functions
label	label for the map. E.g. "Occurrences of <i>Pica pica</i> in Proyecto AVIS"
maptype	Available types of map are 'admin', administrative provinces of Spain (by default) or 'phys', physical map of Spain.

**Value**

a plot with the occurrences of the species in the Iberian Peninsula. Maps have high resolution, so they could be printed.

**Examples**

```
## Not run:
obs<- avisQuerySpecies ("Pica pica", args = list(habitat = "bosque"))
avisMap(obs, label = "Pica pica")
avisMap(obs, label = "Pica pica", maptype = "phys")

## End(Not run)
```

---

avisMapSpecies	<i>Renders a map for each of the species provided in names</i>
----------------	----------------------------------------------------------------

---

**Description**

This function map the species occurrences in the Iberian Peninsula.

**Usage**

```
avisMapSpecies(names, maptype = "admin", ...)
```

**Arguments**

names	scientific name of the species (it could be a list of scientific names). E.g. "Pica pica"
maptype	Available types of map are 'admin', administrative provinces of Spain (by default) or 'phys', physical map of Spain.
...	other filters passed to the observations query with <code>avisQuerySpecies</code>

**Details**

For constructing these maps we used free online map repositories. We downloaded the Spanish administrative map from <http://www.diva-gis.org/> and the Spanish physical map of <http://www.openstreetmap.org/> using the R- library `OpenStreetMap`.

**Value**

a plot with the occurrences of the species in the Iberian Peninsula. Maps have high resolution, so they could be printed.

**Examples**

```
## Not run:

avisMapSpecies("Bubo bubo", "phys")

# if interested in several species, you can explore the database using avisMapSpecies
avisMapSpecies (list("Tyto alba", "Athene noctua", "Bubo bubo", "Strix aluco"),
               maptype="phys")

# and you can save those maps individually using the tiff function

directory<- "C:/your_directory"
species<- list("Tyto alba", "Athene noctua", "Bubo bubo", "Strix aluco")
for (x in species){
  tiff (file.path (directory, paste ("/", x, ".tiff", sep="")))
  avisMapSpecies (x)
  dev.off()
}

## End(Not run)
```

---

 avisQuery

*avisQuery*


---

**Description**

General function for querying the database using several filters, like order, family, species, age, sex, habitat, etc.

**Usage**

```
avisQuery(id_species = "", species = "", family = "", order = "",
age = "", sex = "", breeding = "", habitat = "", month = "", year = "",
args = list())
```

**Arguments**

id_species	a number setting the id of the species according to proyectoavis.com database. You may get the id of a species with <a href="#">avisSpeciesId</a>
species	scientific name of the species (one single species): e.g. "Passer domesticus"
family	To filter the data by family: e.g. "Passeridae", "Falconidae", etc.
order	To filter the data by Order: e.g. "Passeriformes", "Falconiformes", etc.
age	To filter the data by age: "pollo", "juvenil", "adulto", "indeterminado".
sex	To filter the data by sex: "macho", "hembra", "indeterminado", "pareja", "machos y hembras"
breeding	To filter the data by breeding-migratory status: "reproducción posible", "reproducción probable", "reproducción segura", "migración", "invernada"
habitat	Filter by habitat: "bosque", "matorral", "pastizales", "terrenos agrícolas", "zonas humanizadas", "zonas húmedas interiores", "roquedos de interior", "costas", "otros"
month	Filter by month: 1 to 12
year	Filter by year: e.g. 2001
args	List of arguments accepted by <a href="#">www.proyectoavis.com</a> endpoint. You may use this list to set the arguments of the function (species, sex, breeding...), or you may also set all the parameters supported by the endpoint, but not normalized for its use in this package. These arguments are: id_ca, id_provincia, dia_ini, mes_ini, ano_ini, dia_fin, mes_fin, ano_fin, usu, plazo, hora_ini, minuto_ini, hora_fin, minuto_fin, codigo_habitat, gr, cf, utm_10, utm_1 (see <a href="#">www.proyectoavis.com</a> )

**Details**

In case you set a query parameter by its name (eg: `avisQuery (species="Bubo bubo")`) and also you set it inside the 'args' parameter (eg: `avisQuery (species="Bubo bubo", args=list(species="Tyto alba"))`), the value setted by its name will prevail (in the example, "Bubo bubo" will apply).

**Value**

a dataframe with the results of your specific query to Proyecto AVIS database.

**Examples**

```
## Not run:
# get all the observations of the species of the Order Falconiformes
avisQuery (order = "Falconiformes")
# get all the observations of the species of the Family Falconidae
avisQuery(family = "Falconidae")
```

```
# get the observations of immatures of Iberian Imperial Eagle
avisQuery (species= "Aquila adalberti", age = "juvenil")

## End(Not run)
```

---

`avisQueryContributor` *avisQueryContributor*

---

### **Description**

Is a wrapper for `avisQuery` that allows you to perform a query for more than one contributor at once.

### **Usage**

```
avisQueryContributor(contributor_ids, args = list())
```

### **Arguments**

<code>contributor_ids</code>	must be either an integer or a list of contributors ids (integers)
<code>args</code>	A list of normalized parameters to add filters to the query. Currently in Spanish, but this might become outdated. See <code>avisQuery</code> .

### **Value**

a dataframe with the results of your specific query to Proyecto AVIS database

### **See Also**

`avisContributorsSummary`

### **Examples**

```
## Not run:
avisQueryContributor(370)
avisQueryContributor(list(370, 399), args = list(year = 2002))

## End(Not run)
```



---

avisQuerySpecies	<i>avisQuerySpecies</i>
------------------	-------------------------

---

**Description**

Is a wrapper for avisQuery that allows you to perform a query for more than one species at once. 'names' must be either a string or a list of species names, 'args' is a list of query parameters (see avisQuery) that adds further filters to the query.

**Usage**

```
avisQuerySpecies(names, args = list())
```

**Arguments**

names	Must be either a string or a list of scientific names
args	A list of normalized parameters to add filters to the query. Currently in Spanish, but this might become outdated. See avisQuery.

**Value**

a dataframe with the results of your specific query to Proyecto AVIS database

**Examples**

```
## Not run:
avisQuerySpecies("Bubo bubo")
avisQuerySpecies(list("Bubo bubo", "Tyto alba"), args = list(year = 2012))

## End(Not run)
```

---

avisSetup	<i>avisSetup</i>
-----------	------------------

---

**Description**

Sets up settings that apply to the behaviour of the package Allow users to turn off the information messages of the functions.

**Usage**

```
avisSetup (...)
```

**Arguments**

...	Package settings parameters. Available params: verbose = TRUE/FALSE
-----	---------------------------------------------------------------------

**Examples**

```
## Not run:  
avisSetup(verbose=FALSE)  
  
## End(Not run)
```

---

avisSpeciesId	<i>avisSpeciesId</i>
---------------	----------------------

---

**Description**

Returns the id of the selected species

**Usage**

```
avisSpeciesId(nameraw)
```

**Arguments**

nameraw            scientific name of the species (e.g. "Pica pica")

**Value**

an integer

**Examples**

```
## Not run:  
avisSpeciesId("Pica pica")  
  
## End(Not run)
```

---

avisSpeciesSummary	<i>avisSpeciesSummary</i>
--------------------	---------------------------

---

**Description**

Download a table with a summary of the records stored in Proyecto AVIS (<http://proyectoavis.com>) aggregated by species; number of observations of each species, number of individuals recorded, number of different UTM's (10x10km) with observations, number of birdwatchers that recorded the species

**Usage**

```
avisSpeciesSummary()
```

**Value**

returns a dataframe

**Note**

This functions does not allow arguments

**Examples**

```
## Not run:
avis_summary<- avisSpeciesSummary()
# general overview of the data aggregated by species
par (mfrow =c(2,2))
hist (avis_summary$Observations, col="red", border=FALSE, main=NULL)
hist (avis_summary$Individuals, col="red", border=FALSE, main=NULL)
hist (avis_summary$UTM.10x10, col="red", border=FALSE, main=NULL)
hist (avis_summary$Birdwatchers, col="red", border=FALSE, main=NULL)

## End(Not run)
```

---

canarias

*A physical map of the Canary Islands*

---

**Description**

A tif image downloaded from <http://www.openstreetmap.org/> using the R- library OpenStreetMap

**Format**

tif image

**Source**

<http://www.openstreetmap.org/>

---

peninsula

*A physical map of the Iberian Peninsula*

---

**Description**

A tif image downloaded from <http://www.openstreetmap.org/> using the R- library OpenStreetMap

**Format**

tif image

**Source**

<http://www.openstreetmap.org/>

---

rAvis

*rAvis: An R-package to download the information stored in Proyecto AVIS, a citizen science bird project.*

---

## Description

We developed several functions to explore and download the information stored in ProyectoAVIS database ([www.proyectoavis.com](http://www.proyectoavis.com)), in an easy and visual way.

## Details

We programmed two main functions to set flexible queries about the species occurrences and the birdwatcher observations: `avisQuerySpecies` and `avisQueryContributor`. Besides, there are also general functions to explore the database, like `avisMapSpecies`.

Package: rAvis  
Type: Package  
Version: 0.1.3  
Date: 2013-11-24  
License: GPL-2

## Author(s)

Javier Gonzalez Hernandez <[javigzz@yahoo.es](mailto:javigzz@yahoo.es)>

Sara Varela <[svarela@paleobiogeography.org](mailto:svarela@paleobiogeography.org)>

## References

Varela S, Gonzalez-Hernandez J, Casabella E, Barrientos R (2014) rAvis: An R-Package for Downloading Information Stored in Proyecto AVIS, a Citizen Science Bird Project. PLoS ONE 9(3): e91650. doi: 10.1371/journal.pone.0091650

## See Also

<http://proyectoavis.com>

## Examples

```
## Not run:  
avisSpeciesSummary()  
  
avisMapSpecies ("Pica pica", matype="phys")  
  
avisQuerySpecies(list("Bubo bubo", "Tyto alba"), args = list(year = 2012))  
  
## End(Not run)
```

---

ravisUTMLatLong	<i>UTM-Latlong</i>
-----------------	--------------------

---

**Description**

Geographic coordinates (lat-long) of the centroids of the Spanish UTM squares

**Format**

matrix

---

ravis_credentials	<i>ravis_credentials</i>
-------------------	--------------------------

---

**Description**

Credentials for API

**Format**

string

---

ravis_shape_spain	<i>A Spanish administrative map</i>
-------------------	-------------------------------------

---

**Description**

A shapefile downloaded from <http://www.diva-gis.org/>

**Format**

shapefile

**Source**

<http://www.diva-gis.org/>

# Index

## \*Topic **datasets**

- canarias, [11](#)
- peninsula, [11](#)
- ravis\_credentials, [13](#)
- ravis\_shape\_spain, [13](#)
- ravisUTMLatLong, [13](#)

## \*Topic **package**

- rAvis, [12](#)

- avisAllSpecies, [2](#)
- avisContributorAggregatedObservations, [3](#)

- avisContributorsSummary, [3](#)

- avisHasSpecies, [4](#)

- avisMap, [5](#)

- avisMapSpecies, [5](#)

- avisQuery, [6](#)

- avisQueryContributor, [8](#)

- avisQuerySpecies, [9](#)

- avisSetup, [9](#)

- avisSpeciesId, [7](#), [10](#)

- avisSpeciesSummary, [10](#)

- canarias, [11](#)

- peninsula, [11](#)

- rAvis, [12](#)

- rAvis-package (rAvis), [12](#)

- ravis\_credentials, [13](#)

- ravis\_shape\_spain, [13](#)

- ravisUTMLatLong, [13](#)