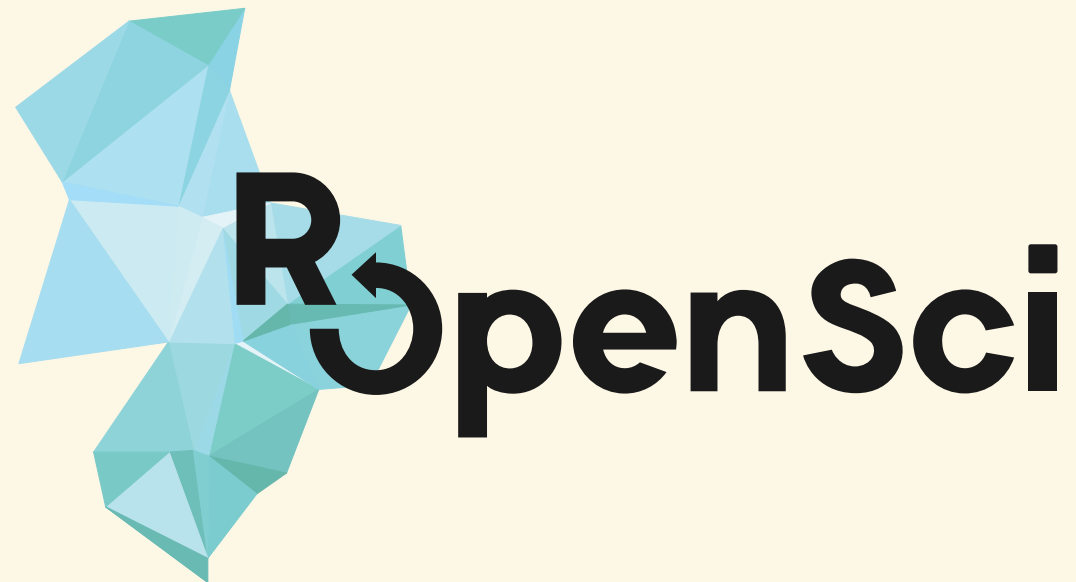


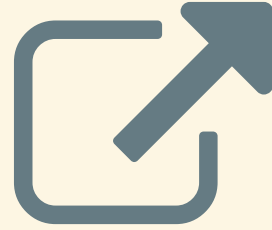
Biodiversity & Taxonomy Software Tools in R

Scott Chamberlain (🐦 [@sckottie](#)/[@ropensci](#))

UC Berkeley / rOpenSci



THE LEONA M. AND HARRY B.
HELMSLEY
CHARITABLE TRUST



scotttalks.info/bocc

pdf slides 960x720

pdf slides 1280x720

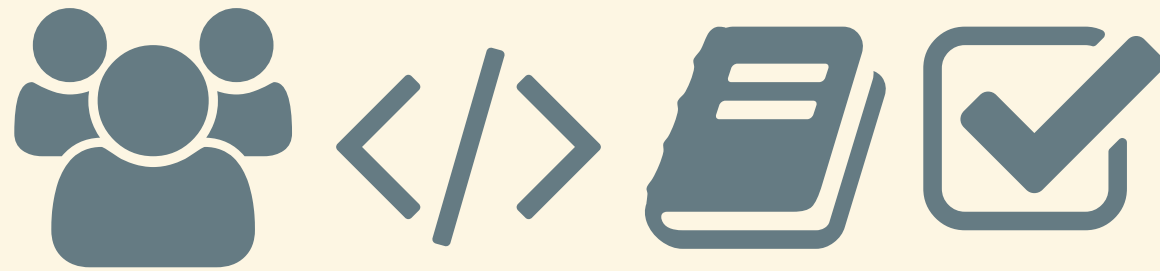
Keyboard shortcuts: press ?

LICENSE: CC-BY 4.0



ropensci.org 

rOpenSci Does



Questions addressed using our software

Comparative genomics in the Asteraceae reveals little evidence for parallel evolutionary change in invasive taxa

A BOTANICAL INVENTORY OF FOREST ON KARSTIC LIMESTONE AND METAMORPHIC SUBSTRATE IN THE CHIQUIBUL FOREST, BELIZE, WITH FOCUS ON WOODY TAXA

Retrieving taxa names from large biodiversity data collections using a flexible matching workflow

The changing patterns of plant naturalization in Australia

A quantitative synthesis of the importance of variables used in MaxEnt species distribution models

Australian acacias as invasive species: lessons to be learnt from regions with long planting histories[§]

Aligning marine species range data to better serve science and conservation

Refining area of occupancy to address the modifiable areal unit problem in ecology and conservation

Evolutionarily Stable Strategies for Fecundity and Swimming Speed of Fish

Citations of rOpenSci Biodiv Software

Taxonomy

package	citations
taxize	71
rentrez	13
ritis	1
taxa	1
worrms	1

Occurrences

package	citations
rgbif	51
rfishbase	24
spocc	8
rfisheries	2
rredlist	2
rvertnet	2
AntWeb	1
pangaeear	1
rotl	8

use case 1

*Hodgins, K. A., Turner, et al. (2015). Comparative genomics in the Asteraceae reveals little evidence for parallel evolutionary change in invasive taxa. Mol Ecol, 24(9), 2226–2240.
10.1111/mec.13026*

in the methods section:

... using the Encyclopedia of Life invasive species comprehensive list, which was accessed programmatically on August 12, 2014 using the `taxize` package in R .

use case 2

*Hodgins, K. A., Turner, et al. (2015). Comparative genomics in the Asteraceae reveals little evidence for parallel evolutionary change in invasive taxa. Mol Ecol, 24(9), 2226–2240.
[10.1111/mec.13026](https://doi.org/10.1111/mec.13026)*

in the methods section:

*... we used rOpenSci's **worms** package in R to standardize spellings of species names and synonyms ...*

Taxonomy

- [taxa](#) - Taxonomic classes and taxonomically aware data manipulation
- [taxize](#) - Taxonomic "toolbelt" - work w/ taxonomy web APIs
- [taxizedb](#) - taxize, but with local SQL databases
- [rentrez](#) - NCBI's Entrez, including taxonomy
- [worrms](#) - WORMS web service
- [ritis](#) - USGS's ITIS web service
- ... many others

Taxonomic data from >20 sources - taxize

always try to move from:

- taxonomic name -- to
- taxonomic ID -- to
- whatever other data
(e.g., synonyms, classifications, etc.)

Taxonomic data from >20 sources - taxize

Taxonomic hierarchies from NCBI/ITIS/COL/etc

```
library('taxize')  
id <- get_gbifid("Chironomus riparius")  
classification(id)
```

```
#> $`Chironomus riparius`  
#>      name      rank      id  
#> 1   Animalia kingdom      1  
#> 2   Arthropoda phylum    54  
#> 3     Insecta   class    216  
#> 4     Diptera   order    811  
#> 5   Chironomidae family   3343  
#> 6     Chironomus genus 1448033  
#> 7 Chironomus riparius species 1448237
```

Wrangling data paired with taxonomy - taxa

```
library('taxa')  
ex_taxmap
```

```
<Taxmap>  
17 taxa: b. Mammalia, c. Plantae, d. Felidae ... p. sapiens, q. lycopersicum, r. tuberosum  
17 edges: NA->b, NA->c, b->d, b->e, b->f, c->g, d->h, d->i ... g->l, h->m, i->n, j->o, k->p, l->q, l->r  
4 data sets:  
info:  
  # A tibble: 6 x 4  
  taxon_id name n_legs dangerous  
  <chr>    <chr> <dbl> <lgl>  
1 m      tiger     4 TRUE  
2 n      cat       4 FALSE  
3 o      mole     4 FALSE  
# ... with 3 more rows  
phylopic_ids: a named vector of 'character' with 6 items  
  m. e148eabb-f138-43c6-b1e4-5cda2180485a ... r. 63604565-0406-460b-8cb8-1abe954b3f3a  
foods: a list of 6 items named by taxa:  
  m, n, o, p, q, r  
abund:  
  # A tibble: 8 x 5  
  taxon_id code sample_id count taxon_index  
  <chr>    <fct> <fct>    <dbl>    <int>  
1 m      T      A         1         1  
2 n      C      A         2         2  
3 o      M      B         5         3
```

Wrangling data paired with taxonomy - taxa

```
filter_taxa(ex_taxmap, startsWith(taxon_names, "t")) # filter
```

```
<Taxmap>
  3 taxa: m. tigris, o. typhlops, r. tuberosum
  3 edges: NA->m, NA->o, NA->r
  4 data sets:
  info:
    # A tibble: 3 x 4
      taxon_id name    n_legs dangerous
    <chr>    <chr>    <dbl> <lgl>
  1 m      tiger      4 TRUE
  2 o      mole      4 FALSE
  3 r      potato    0 FALSE
  phylopic_ids: a named vector of 'character' with 3 items
    m. e148eabb-f138-43c6-b1e4-5cda2180485a, o. 11b783d5-af1c-4f4e-8ab5-a51470652b47, r. 63604565-0
be954b3f3a
  foods: a list of 3 items named by taxa:
    m, o, r
  abund:
    # A tibble: 4 x 5
      taxon_id code  sample_id count taxon_index
    <chr>    <fct> <fct>    <dbl>    <int>
  1 m      T     A         1         1
  2 o      M     B         5         3
  3 m      T     A         6         1
  # ... with 1 more row
  1 functions:
  reaction
```

ENTREZ in R - `rentrez`

Retrieve downstream taxonomy from a given taxon

```
library(rentrez)
x <- entrez_search(db = "taxonomy", term = "Satyrium[Next Level]", retmax = 10)
z <- entrez_summary(db = "taxonomy", id = x$sids)

data.frame(t(
  extract_from_esummary(z, c("uid", "scientificname", "rank"))
))

#> uid          scientificname      rank
#> 2023262      Satyrium liltvedianum species
#> 1888668          Satyrium calanus species
#> 1847691 Satyrium sp. BOLD:ABX6433 species
#> 1825277      Satyrium sp. XQX-2016 species
#> 1824831          Satyrium grandis species
#> 1430660          Satyrium favonius species
#> 1423409          Satyrium sylvinus species
#> 1405335          Satyrium mera species
#> 1405334          Satyrium iyonis species
#> 985828      Satyrium situsanguinum species
```

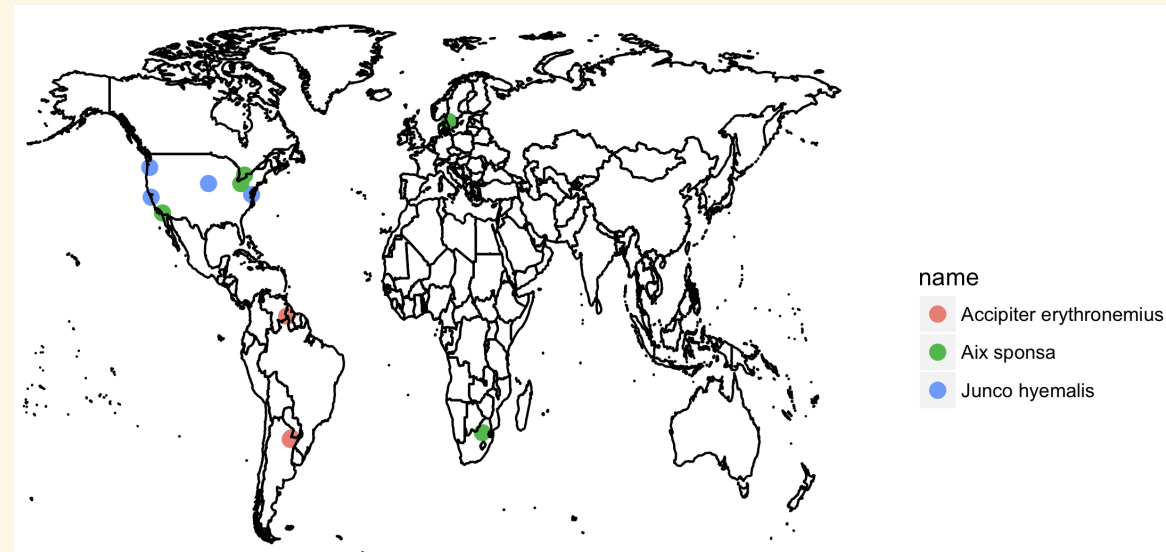

Occurrence data

- `rgbif` - GBIF
- `rbison` - USGS's BISON
- `rvertnet` - VertNet
- `rebird` - eBird (see also `auk`)
- `spocc` - one stop shop (of all above)
- `finch` - parse GBIF bulk data
 - `EML` - read and create EML
- `ridigbio` `ALA4R` `robis` *

GBIF data - rgbif

```
library(rgbif)
spp <- c('Accipiter erythronemius', 'Junco hyemalis', 'Aix sponsa')
keys <- unname(vapply(spp, function(x) name_backbone(name=x)$speciesKey, 1))
dat <- occ_search(taxonKey=keys, limit=5, hasCoordinate=TRUE)

library(mapr)
map_ggplot(dat)
```



GBIF p.s.

we also maintain GBIF clients in [Python](#) and [Ruby](#)

one stop shop for occurrence data - spocc

```
library('spocc')
gbifopts <- list(country = 'US')
idigbioopts <- list(fields = "scientificname")

out <- occ(query = 'Setophaga caerulescens',
           from = c('gbif','bison','inat','idigbio'), limit = 50,
           gbifopts = gbifopts, idigbioopts = idigbioopts)
dat <- occ2df(out)
head(dat); tail(dat)
#>   name                longitude  latitude  prov  date        key
#> 1 Setophaga caerulescens -80.347459 25.743763 gbif 2018-01-20 1806338790
#> 2 Setophaga caerulescens -80.342233 25.77536  gbif 2018-01-19 1805421161
#> 3 Setophaga caerulescens -81.355815 28.569623 gbif 2018-03-14 1837766480
```

Standard interface to varied user inputs to the same things

- **Pagination:** `limit`, `start`, `page`
- **Spatial search:** `geometry`
- **Records w/ coordinates:** `has_coords`

future work /
hard problems

taxonomy tools: in the works

- taxonomic name parsing: *fast & platform independent for other R tool builders & tools for R users* (see [pegax poster](#))
- package [taxa](#): needs more user testing - feedback plz!
- package [taxadc](#): serialize R taxonomic data to Darwin Core- in early development
- package [taxizedb](#) - hard to a) make similar interface to SQL DB's as web services & 2) simplify varied database installs
- package [taxview](#) - summarise and visualize data sets from with respect to taxonomy

Summarise/visualize data sets by taxonomy - taxview

```
library('taxview')
x <- system.file("examples/plant_spp.csv", package = "taxview")
```

prepare data: clean, etc.

```
(dat <- tibble::as_tibble(
  data.table::fread(x, stringsAsFactors = FALSE,
    data.table = FALSE)))
are> dat
#> # A tibble: 130 x 2
#>   name          id
#>   <chr>         <int>
#> 1 Dianthus engleri 1531994
#> 2 Anacolosa frutescens 1618138
#> 3 Hymenophyllum plicatum 638568

(dat_clean <- tv_clean_ids(x, ids = dat$id, db = "ncbi"))
#>   name          rank      id      query
#>   <chr>         <chr>    <chr> <chr>
#> 1 cellular organisms no rank 131567 1531994
#> 2 Eukaryota        superkingdom 2759 1531994
#> 3 Viridiplantae    kingdom 33090 1531994
```

Summarise/visualize data sets by taxonomy - taxview

```
(sumdat <- tv_summarise(dat_clean))
```

```
#> <tv_summary>  
#> no. taxa: 129  
#> by rank: N (21)  
  
#> by rank name: N (594)  
#> within ranks: N (20)
```

```
sumdat$by_within_rank  
#> $superkingdom  
#> name      count percent  
#> 1 Eukaryota  127      98  
#> 2 Bacteria    2        2  
  
sumdat$by_within_rank$subphylum  
#> name      count percent  
#> 1 Streptophytina  119      96  
#> 2 Craniata         2         2  
#> 3 Hexapoda         2         2  
#> 4 Chelicerata     1         1
```


Summarise/visualize data sets by
taxonomy

coming ...

occurrence tools: in the works

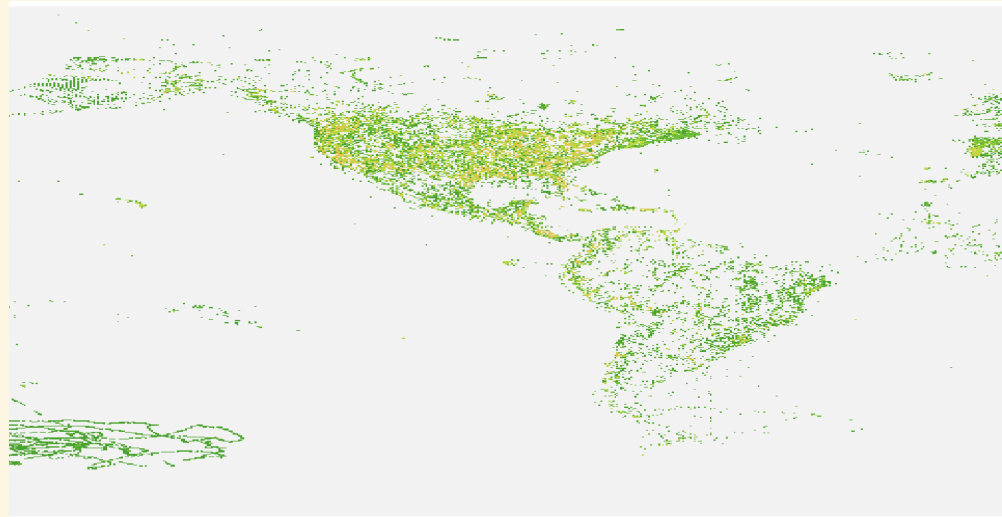
- taking the pain out of GBIF downloads:
[ropensci/rgbif#266](#): queuing tool for GBIF downloads
- *hard problem*
- occurrence cleaning in R: *hard problem!* A few efforts:
 - package [scrubr](#)
 - package [CoordinateCleaner](#)
 - deduplication badly needed: aggregation up the data provider ladder

occurrence tools: questions

- DOIs for GBIF search service (and other services)?
- related to above: Sharing dataset associated with paper
- Visualizing huge occurrence datasets? (GBIF map API now in dev ver of `rgbif`)

GBIF map service (rasters) ~ static

```
library(rgbif) # development version remotes::install_github("ropensci/rgbif")
library(raster)
x <- map_fetch(search = "taxonKey", id = 3118771, year = 2010)
plot(x)
```



Perhaps scientists can use these rasters for analysis?

GBIF map service (rasters) ~ interactive

```
library(leaflet)
pal <- colorNumeric(c("#0C2C84", "#41B6C4", "#FFFFCC"), values(x), na.color = "tr
leaflet() %>%
  addTiles() %>%
  addRasterImage(x, colors = pal, opacity = 0.8) %>%
  addLegend(pal = pal, values = values(x),
            title = "Occurrences")
```



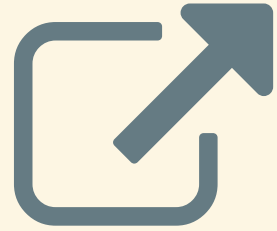
Hopefully use GBIF's Map Vector Tile (MVT) soon

talk to us 

what would you like to see?

discussion forum: discuss.ropensci.org

submit a package/review a package:
github.com/ropensci/onboarding



scotttalks.info/bocc

Made w/: [reveal.js v3.2.0](#)

Some Styling: [Bootstrap v3.3.5](#)

Icons by: [FontAwesome v4.4.0](#)