

# Get occurrences from GBIF

---

**GBIF** (Global Biodiversity Information Facility) is an international open-data infrastructure that provides free and open access to biodiversity data. It aggregates species occurrence records (e.g., observations, museum specimens, and monitoring data) from around the world, making them available for research, conservation, and policy-making. Although you can download data directly from the GBIF website, you can also retrieve occurrences programmatically using R. Let's see how to do it.

First, the package that allows us to access GBIF data is *rgbif*. You'll need to install it along with its required dependencies.

```
install.packages("rgbif", dependencies = TRUE)
```

Now, with the package installed, you need to load the library and begin your search. For example, you can use the function *occ\_search()*, which is the simplest and most straightforward option. In this example, you'll download occurrences of *Marmota monax* (groundhog).

Since you'll likely use spatial data for analysis, you'll want only records with coordinates—GBIF also includes non-georeferenced records. To filter these, set the parameter *hasCoordinate = TRUE*.

```
library(rgbif)

occ_data <- occ_search(
  scientificName = "Marmota monax",
  hasCoordinate = TRUE # Only records with coordinates
)
```

Now, let's explore the results. The *occ\_data* object is a structured list containing different types of data. An easy way to inspect its contents is using the *str()* function.

As you'll see, *occ\_data* consists of several components:

- **meta**: Metadata about your query
- **hierarchy**: Taxonomic classification details
- **data**: The core occurrence records (as a tibble)
- **media**: Associated multimedia information
- **facets**: Aggregated statistics (if requested)

The **data** component contains all the key information - not just coordinates, but also year, date, taxonomy, observation type, and more.

```
str(occ_data, max.level = 1)
```

```
## List of 5
## $ meta      :List of 4
```

```
## $ hierarchy:List of 1
## $ data      : tibble [500 x 103] (S3: tbl_df/tbl/data.frame)
## $ media     :List of 500
## $ facets    : Named list()
## - attr(*, "class")= chr "gbif"
## - attr(*, "args")=List of 6
## - attr(*, "type")= chr "single"
```

One limitation of the `*occ_search()*` function is its default return **limit** of 500 occurrence records. Fortunately, you can adjust this by specifying the limit parameter, and further refine your search using additional filters like year ranges, country codes, species specifications, and spatial precision requirements.

```
dim(occ_data$data) #Check the dimensions of the table

# Tuning Your GBIF Search Parameters

occ_data <- occ_search(
  scientificName = "Marmota monax",
  country = "US", # Country code
  year = "2000,2023", # Year range
  basisOfRecord = "HUMAN_OBSERVATION", # Only human observations
  hasCoordinate = TRUE,
  limit=90000
)

dim(occ_data$data) #Check the dimensions of the table
```

This function offers numerous parameters, allowing you to refine your search with precision. To view all available options, simply run:

```
?occ_search
```

Here is the final code with optimized parameters for this exercise. You can save your data in either CSV or RDS format - both are good options. If you need to work with the coordinates in other software, we recommend using the CSV format.

```
occ_data <- occ_search(
  scientificName = "Marmota monax",
  country = "US", # Country code
  hasCoordinate = TRUE,
  limit=90000
)

dim(occ_data$data) #Check the dimensions of the table

# Save as CSV
write.csv(occ_data$data, "M_monax_gbif_occurrences.csv", row.names = FALSE)

# Save as RDS (preserves data types)
saveRDS(occ_data$data, "M_monax_gbif_occurrences.rds")
```