

Extracting data from the NBN Gateway into R

Stuart Ball, JNCC
Tom August, CEH

March 24, 2014

1 Introduction

The National Biodiversity Network (NBN) is an on-line repository for biodiversity data from the UK. At the time of writing, it contains over 90 million species records in over 900 datasets. Data can be accessed via web-services provided by the Gateway web-site (for documentation see https://data.nbn.org.uk/Documentation/Web_Services/Web_Services-REST/resources/restapi/index.html).

This package provides methods to interact with the NBN's web services and get species records and other supporting information. The functions fall into two groups:

Functions that access a particular service and return a JSON object

dataProviders get information on the organisations that have contributed data to specified datasets

datasetTaxa gets a list of taxa that are included in a given dataset or list of datasets.

getFeature get information about a "feature" (a location at which occurrences have been recorded) given its featureID.

getGroupSpeciesTVKs given the name of a group (see `listGroups`) this function returns the pTVKs (preferred taxon version keys) for all members of that group. This is currently restricted to returning up to 5000 results.

getOccurrences get occurrences for a particular taxa, grid cell or species group and returns a data.frame containing the occurrences. Optionally, the results can be filtered by dataset, date and vicecounty.

getTaxonomy given a TVK, this function gets details of the taxonomical heirarchy of a taxon.

getTVKQuery given a search term this function returns species information, including the TVK, for the first 25 taxa that match that search on the NBN.

listDatasets returns a dataframe of the datasets available from the NBN Gateway for reference.

listGroups returns a dataframe of the group definitions from the NBN Gateway for reference.

listOrganisations returns a dataframe of the organisation definitions from the NBN Gateway for reference.

listVCs returns a dataframe of the Watsonian vice-counties and their keys for reference.

nbnLogin takes a username and password and logs the user into the NBN gateway

Utility functions which manipulate grid reference and date information returned by the NBN Gateway

datePart takes the vague date information, returned in three fields (startDate, endDate and dateTypeKey) from the NBN Gateway and extracts elements of the date like the year or week, whilst properly taking into account the type of vague date.

gridCoords takes a grid reference string (OSGB or OSNI) and calculates the x,y coordinates of the bottom, left-hand corner of the grid square.

gridRef takes a grid reference string (OSGB or OSNI) and extracts grid references at other precisions. For example, extract 10km square grid refs from the grid references returned from the Gateway.

gr2gps_latlon takes a grid reference string (OSGB or OSNI) and calculates the latitude and longitude of the centre or bottom left corner.

2 Registering with the NBN gateway and logging in

To use data from the NBN gateway you must first register. This is an easy process and can be done by visiting <https://data.nbn.org.uk/User/Register>. Once registered you will be sent an email to verify your address, once verified you are ready to use **rnbn**.

When using **rnbn** you will be asked to login the first time you attempt to access occurrence data. Once logged in cookies are saved in your working directory and will be used in the future preventing the need to log in repeatedly.

3 Getting species occurrence records

The **getOccurrences** function gets a data.frame of species occurrence records from the NBN Gateway. Columns include name, TVK, date and location of the observation as a minimum, and may include other columns depending what has been submitted by the data providers and what access they allow. The

first time this function is used, or if your cookies expire, you will be asked for your username and password. An alternative method for logging in is to use the `nbnLogin` function (see below)

The minimum information required to request species occurrences from the NBN Gateway is one of the following: a Taxon Version Key (TVK), a grid reference or the name of a species group.

Independent of which method you use there are three messages that will appear in your console:

```
# Load the package
library(rnbn)

## Loading required package: RCurl
## Loading required package: bitops
## Loading required package: RJSONIO
## Loading required package: tcltk

# I could log in like this...
# nbnLogin(username = 'myUsername', password = 'myPassword')
# ...or let getOccurrences prompt me. The latter is more
# secure as I dont have to include my password in my scripts

# Request occurrence data using taxon version key
occ <- getOccurrences(tvks = "NBNSYS0000002010")

## Requesting batch 1 of 1
## Requesting data providers information

## IMPORTANT: https://data.nbn.org.uk/Terms By using this package you
are agreeing to the Gateway Terms & Conditions and Privacy Policy.
This message can be suppressed using the acceptTandC argument
```

The first message returned to console details the batch number being processed. `rnbn` breaks down a data request into batches so that it does not overload the system. This is also useful for monitoring progress. The second message tells us that the function is retrieving the data providers for the data it just collected. These can be silenced by setting `silent = TRUE`. The third message is a warning that highlights the terms and conditions associated with using data from the NBN gateway. It is important that you read these terms and conditions since by using the `rnbn` package you are accepting them. This warning can be silenced by setting `acceptTandC = TRUE`.

3.1 Using Taxon Version Keys (TVKs)

TVKs are 16-character strings of (usually, upper-case) letters and numbers. For example, “NBNSYS0000007111”.

TVKs can be found using the function `getTVKQuery`. This function will take the name of a species and attempt to match it to a TVK using the NBN's search feature. For example if we wanted the TVK for 'badger' (*Meles meles*):

```
# Search for taxon information using the query 'badger'
dt <- getTVKQuery(query = "badger")
# Display two columns of the data 'ptaxonVersionKey' and 'name'
dt[, c("ptaxonVersionKey", "name")]

##      ptaxonVersionKey      name
## 1 NHMSYS0000080191      Badger
## 2 NBNSYS0000013055    Badger Flea
## 3 NHMSYS0000545919    a Badger flea
## 4 NHMSYS0000080191 Eurasian Badger
```

You will notice that 'Badger' and 'Eurasian Badger' have the same 'ptaxonVersionKey' (the 'p' stands for preferred). This is because the terms are synonyms, both referring to *Meles meles* (which would also share the same ptaxonVersionKey). By using this TVK in the `getOccurrence` function it ensures that you get data for all synonyms. If you don't wish to include synonyms you can instead use the TVK given in the column 'taxonVersionKey'.

The following example will get all publicly available observations of *Tropidia scita* from all datasets and for any date:

```
library(rnbn)
# Get species TVK
dt <- getTVKQuery(query = "Tropidia scita") #returns one row
# Retrieve data from NBN using a TVK
occ <- getOccurrences(tvks = dt$ptaxonVersionKey, acceptTandC = TRUE)

## Requesting batch 1 of 1
## Requesting data providers information

# Print the first few rows and a selection of columns
occ[1:10, c("pTaxonName", "startDate", "latitude", "longitude")]

##      pTaxonName startDate latitude longitude
## 1 Tropidia scita 1985-01-01    51.38   -1.2111
## 2 Tropidia scita 2005-06-28    52.73    1.6216
## 3 Tropidia scita 2005-06-27    52.73    1.6159
## 4 Tropidia scita 2005-06-28    52.72    1.6198
## 5 Tropidia scita 2005-06-27    52.73    1.6114
## 6 Tropidia scita 1985-07-01    51.14   -1.4653
## 7 Tropidia scita 1986-09-01    52.24    0.2385
## 8 Tropidia scita 2006-06-29    52.24    0.2413
## 9 Tropidia scita 1960-01-01    52.24    0.2417
## 10 Tropidia scita 1996-06-25    57.89   -4.2439
```

TVKs can also be found on the NBN gateway at <https://data.nbn.org.uk/Taxa>. Navigating to a species reveals additional information including the 'Taxon Version Key'

Occurrences for more than one species can be obtained by passing a list of TVKs. Such lists can be created in two ways:

```
# List TVKs manually
tvks <- c("NHMSYS0000530420", "NHMSYS0000530658")
tvks

## [1] "NHMSYS0000530420" "NHMSYS0000530658"

# Retrieve a list of TVKs using the NBN search
species <- getTVKQuery("grouse")
tvks <- unique(species$ptaxonVersionKey)
tvks

## [1] "NHMSYS0000530420" "NHMSYS0000530658"
```

3.2 Using grid references

Data can be retrieved by specifying a grid reference in which to search:

```
# Retrieve data from NBN using a gridreference
occ <- getOccurrences(gridRef='TL3490', acceptTandC=TRUE)
```

This search will work with a range of grid reference resolutions and for grid references in OSNI and OSGB format.

3.3 Using species group

Data can be retrieved by specifying a species group. Species groups are taxonomic groups that are predefined by the NBN. A list of available groups can be found using the `listGroups` function.

```
# View some of the groups available
groups <- listGroups()
head(groups)

##              name              key
## 1      acarine (Acari) NHMSYS0000629148
## 2 acorn worm (Hemichordata) NHMSYS0000080031
## 3              alga NHMSYS0000080032
## 4      amphibian NHMSYS0000080033
## 5          annelid NHMSYS0000080034
## 6      archaean NHMSYS0000629143
```

Once you have decided which group you require the name is passed to `getOccurrences` in the following manner.

```
# Retrieve data from NBN using a species group
# Note this can take some time depending on the size of the species group
occ <- getOccurrences(group='quillwort')
```

3.4 Filtering results

3.4.1 By Dataset

Observations can be filtered so that they come only from datasets you trust by passing one or more dataset key to the `datasets` parameter. Dataset keys can be found using the `listDatasets` function:

```
# View some of the datasets available
datasets <- listDatasets()
head(datasets[45:50, ]) # I select a group with short titles

##               title      key
## 46 Bedfordshire Diplopoda (BNHS) - 1975-1985 GA000675
## 47 Bedfordshire Dormice (BNHS/BDG) - 2000-2012 GA000703
## 48 Bedfordshire Fish (BNHS) - 1800-2011 GA000704
## 49 Bedfordshire Flora (BNHS/BSBI) - 1904-2012 GA000482
## 50 Bedfordshire Herpetofauna (BNHS/BRAG) - 1973-2013 GA000458
## 51 Bedfordshire Himalayan Balsam Surveys (WT) - 2010-2012 GA001205
```

A list of datasets can be passed in a similar way to a list of species keys.

```
# Specify dataset keys
datasets <- c("SGB00001", "GA000483")
# Retrieve data
occ <- getOccurrences(tvk='NBNSYS0000007111', datasets=datasets)
```

Dataset keys can also be found on the NBN gateway at <https://data.nbn.org.uk/Datasets>. Clicking on a dataset reveals metadata for that dataset including the key, named 'Permanent key'.

3.4.2 By Year

The range of years for which you want to extract data can be specified using the `startYear` and/or `endYear` parameters:

```
# Get data for a specified species, from a specified dataset over
# a specified time period
dt <- getOccurrences(tvks="NBNSYS0000007111", datasets="SGB00001",
                    startYear=1990, endYear=2006)
```

3.4.3 By Vice-county

If data from a specific vice-county is required then the VC argument can be used. This takes the name of a vicecounty, a list of which can be found using `listVCs`:

```
# View some of the vice-counties available
VCs <- listVCs()
head(VCs)

##           name identifier featureID
## 1   Anglesey  GA00034452   2583220
## 2 Angus (Forfar) GA00034490   2583258
## 3   Ayrshire  GA00034475   2583243
## 4   Banffshire GA00034494   2583262
## 5 Bedfordshire GA00034430   2583198
## 6   Berkshire GA00034422   2583190
```

Once you have decided the vice-county you wish to search within you can use the `getOccurrences` function like this:

```
# Request data for one species from East Suffolk
occ <- getOccurrences(tvk='NBNSYS0000007111',VC='East Suffolk')
```

3.5 Dataset Information

Two functions allow access to additional information about datasets.

3.5.1 Data providers

For many uses of data from the NBN it is necessary to get permission from data owners. This function returns the contact information for a given dataset:

```
# Get contact details for two datasets
providers <- dataProviders(c("GA000426", "GA000832"))
# A range of details are provided
names(providers)

## [1] "id"           "name"           "address"         "postcode"
## [5] "contactName"  "contactEmail"   "website"

# This function is used internally to provide contact information for
# getOccurrences searches
occ <- getOccurrences(gridRef = "TL3490", acceptTandC = TRUE)

## Requesting batch 1 of 1
## Requesting data providers information
```

```
# The information is returned as an attribute 'providers'
providers <- attr(occ, "providers")
# A row is given for each data provider
nrow(providers)

## [1] 8
```

3.5.2 Taxa list

It can be sometimes be helpful to have a list of taxa that are recorded in a given dataset here is an example of how this can be done:

```
# Get taxa list for the ladybird survey
taxalist <- datasetTaxa("GA000312")
# A range of details are provided
names(taxalist)

## [1] "taxonVersionKey"      "name"                  "authority"
## [4] "languageKey"          "taxonOutputGroupKey"   "taxonOutputGroupName"
## [7] "commonName"           "gatewayRecordCount"    "href"
## [10] "observationCount"     "datasetKey"            "ptaxonVersionKey"

# Here are some of those species
head(taxalist$commonName)

## [1] "2-spot Ladybird"      "10-spot Ladybird"      "Eyed Ladybird"
## [4] "Water Ladybird"       "Larch Ladybird"        "Cream-spot Ladybird"
```

References

- Hijmans, R., Phillips, S., Leathwick, J. & Elith, J., 2013. dismo: Species distribution modeling. r package version 0.8-11.
URL <http://CRAN.R-project.org/package=dismo>
- Hill, M.O., 2012. Local frequency as a key to interpreting species occurrence data when recording effort is not known. *Methods in Ecology and Evolution*, **3**, 195–205.
URL <http://onlinelibrary.wiley.com/doi/10.1111/j.2041-210X.2011.00146.x/pdf>