

# Package ‘rebird’

July 9, 2015

**Version** 0.2

**Date** 2015-06-30

**Title** R Client for the eBird Database of Bird Observations

**Description** A programmatic client for the eBird database, including functions for searching for bird observations by geographic location (latitude, longitude), eBird hotspots, location identifiers, by notable sightings, by region, and by taxonomic name.

**License** MIT + file LICENSE

**URL** <http://github.com/ropensci/rebird>

**BugReports** <http://github.com/ropensci/rebird/issues>

**VignetteBuilder** knitr

**Imports** methods, utils, stats, jsonlite, httr, dplyr

**Suggests** knitr, testthat

**NeedsCompilation** no

**Author** Rafael Maia [aut],  
Scott Chamberlain [aut],  
Andy Teucher [aut],  
Sebastian Pardo [aut, cre]

**Maintainer** Sebastian Pardo <sebpardo@gmail.com>

**Repository** CRAN

**Date/Publication** 2015-07-09 14:47:54

## R topics documented:

rebird-package . . . . .	2
ebirdfreq . . . . .	2
ebirdgeo . . . . .	4
ebirdhotspot . . . . .	5
ebirdloc . . . . .	7
ebirdnotable . . . . .	9

ebirdregion . . . . .	11
ebirdregioncheck . . . . .	13
ebirdtaxonomy . . . . .	14
getlatlng . . . . .	15
nearestobs . . . . .	15

<b>Index</b>	<b>18</b>
--------------	-----------

---

rebird-package	<i>rebird: an R package for access to the eBird database.</i>
----------------	---

---

## Description

An R package to interface with the eBird database.

## Details

Package:	rebird
Type:	Package
Version:	0.1.2.99
Date:	2014-10-31
License:	MIT
LazyLoad:	yes

This package provides an R interface to the recent eBird database.

## Author(s)

Rafael Maia <rm72@zip.s.uakron.edu>, Scott Chamberlain <myrmecocystus@gmail.com>, Andy Teucher <andy.teucher@gmail.com>, Sebastian Pardo <sebpardo@gmail.com>

## Examples

#see functions.

---

ebirdfreq	<i>Download historical frequencies of bird observations from eBird</i>
-----------	--

---

## Description

Download historical frequencies of bird observations from eBird

## Usage

```
ebirdfreq(loctype, loc, startyear = 1900, endyear = format(Sys.Date(),
"%Y"), startmonth = 1, endmonth = 12, long = TRUE, ...)
```

**Arguments**

loctype	String with location type. Either "states", "counties", or "hotspots".
loc	String with location identifier. If querying states or provinces, the two letter country code followed by the two letter state code and separated by "-" (e.g. "US-NY"). If querying counties, is as in states/provinces, but appending county identifier after a dash. For counties in the US, the county codes is a 3-digit number specific to each state (e.g. Bronx County: "US-NY-005"). For counties in Canada, county codes are two-letter identifiers (e.g. Metro Vancouver: "CA-BC-GV"). If querying hotspots then the unique identifier is a 6-digit number prepended with an "L" (e.g. "L196159"). All these codes can be found by looking at the URL in the Explore Data eBird page ( <a href="http://ebird.org/GuideMe?cmd=changeLocation">http://ebird.org/GuideMe?cmd=changeLocation</a> ).
startyear	Starting year for query. Defaults to 1900.
endyear	Ending year for query. Defaults to current year specified by Sys.Date().
startmonth	Starting month for query as an integer (1-12). Defaults to January.
endmonth	Ending month for query as an integer (1-12). Defaults to December.
long	Logical, Should output be in long format? Defaults to TRUE. If FALSE then output will be in wide format.
...	Curl options passed on to <a href="#">GET</a>

**Value**

A data frame containing the collected information. If in long format:

"monthQt": month and week (eBird data divides each month by four weeks)

"comName": species common name

"frequency": proportion of times the species was seen in a specified week

"sampleSize" number of complete eBird checklists submitted for specified given week

If in wide format, then first column is the species list and all other columns are of individual weeks (four in each month). First row contains the number of complete checklists for each week.

**Author(s)**

Andy Teucher <[andy.teucher@gmail.com](mailto:andy.teucher@gmail.com)>, Sebastian Pardo <[sebpardo@gmail.com](mailto:sebpardo@gmail.com)>

**References**

<http://ebird.org/>

**Examples**

```
## Not run:
ebirdfreq("states", "US-NY", 2014, 2014, 1, 12)
ebirdfreq("counties", "CA-BC-GV", 1900, 2015, 1, 3)
ebirdfreq("hotspots", "L196159", long=FALSE)

## End(Not run)
```

---

 ebirdgeo

*Sightings at location determined by latitude/longitude*


---

### Description

Returns the most recent sighting date and specific location for the requested species of bird reported within the number of days specified and reported in the specified area.

### Usage

```
ebirdgeo(species = NULL, lat = NULL, lng = NULL, dist = NULL,
         back = NULL, max = NULL, locale = NULL, provisional = FALSE,
         hotspot = FALSE, sleep = 0, ...)
```

### Arguments

species	Scientific name of the species of interest (not case sensitive). Defaults to NULL, so sightings for all species are returned. See eBird taxonomy for more information: <a href="http://ebird.org/content/ebird/about/ebird-taxonomy">http://ebird.org/content/ebird/about/ebird-taxonomy</a>
lat	Decimal latitude. value between -90.00 and 90.00, up to two decimal places of precision. Defaults to latitude based on IP.
lng	Decimal longitude. value between -180.00 and 180.00, up to two decimal places of precision. Defaults to longitude based on IP.
dist	Distance defining radius of interest from given lat/lng in kilometers (between 0 and 50, defaults to 25)
back	Number of days back to look for observations (between 1 and 30, defaults to 14).
max	Maximum number of result rows to return in this request (between 1 and 10000, defaults to all).
locale	Language/locale of response (when translations are available). See <a href="http://java.sun.com/javase/6/docs/api/j">http://java.sun.com/javase/6/docs/api/j</a> (defaults to en_US).
provisional	Should flagged records that have not been reviewed be included? (defaults to FALSE).
hotspot	Should results be limited to sightings at birding hotspots? (defaults to FALSE).
sleep	Time (in seconds) before function sends API call (defaults to zero. Set to higher number if you are using this function in a loop with many API calls).
...	Curl options passed on to <a href="#">GET</a>

### Value

A data.frame containing the collected information:

"comName": species common name

"howMany": number of individuals observed, NA if only presence was noted

"lat": latitude of the location  
 "lng": longitude of the location  
 "locID": unique identifier for the location  
 "locName": location name  
 "locationPrivate": TRUE if location is not a birding hotspot  
 "obsDt": observation date formatted according to ISO 8601 (e.g. 'YYYY-MM-DD', or 'YYYY-MM-DD hh:mm'). Hours and minutes are excluded if the observer did not report an observation time.  
 "obsReviewed": TRUE if observation has been reviewed, FALSE otherwise  
 "obsValid": TRUE if observation has been deemed valid by either the automatic filters or a regional viewer, FALSE otherwise  
 "sciName" species' scientific name

### Author(s)

Rafael Maia <rm72@zips.uakron.edu>

### References

<http://ebird.org/>

### Examples

```
## Not run:
ebirdgeo('spinus tristis', 42, -76)
ebirdgeo(lat=42, lng=-76, max=10, provisional=TRUE, hotspot=TRUE)
ebirdgeo('Anas platyrhynchos', 39, -121, max=5)
library('httr')
ebirdgeo('Anas platyrhynchos', 39, -121, max=5, config=verbose())
ebirdgeo('Anas platyrhynchos', 39, -121, max=5, config=user_agent("rebird"))
ebirdgeo('Anas platyrhynchos', 39, -121, max=5, config=progress())
# ebirdgeo('Anas platyrhynchos', 39, -121, max=5, config=timeout(0.1))

## End(Not run)
```

---

ebirdhotspot

*Recent observations at hotspots*

---

### Description

Returns the most recent sighting information reported in a given vector of hotspots.

### Usage

```
ebirdhotspot(locID, species = NULL, back = NULL, max = NULL,
  locale = NULL, provisional = FALSE, sleep = 0, ...)
```

**Arguments**

locID	(required) Vector containing code(s) for up to 10 regions of interest; here, regions are the locIDs of hotspots. Values that are not valid or are not hotspots are ignored.
species	Scientific name of the species of interest (not case sensitive). Defaults to NULL, in which case sightings for all species are returned. See eBird taxonomy for more information: <a href="http://ebird.org/content/ebird/about/ebird-taxonomy">http://ebird.org/content/ebird/about/ebird-taxonomy</a>
back	Number of days back to look for observations (between 1 and 30, defaults to 14).
max	Maximum number of result rows to return in this request (between 1 and 10000, defaults to all)
locale	Language/locale of response (when translations are available). See <a href="http://java.sun.com/javase/6/docs/api/">http://java.sun.com/javase/6/docs/api/</a> (defaults to en_US)
provisional	Should flagged records that have not been reviewed be included? (defaults to FALSE)
sleep	Time (in seconds) before function sends API call (defaults to zero. Set to higher number if you are using this function in a loop with many API calls).
...	Curl options passed on to <a href="#">GET</a>

**Value**

A data.frame containing the collected information:

"comName": species common name

"howMany": number of individuals observed, NA if only presence was noted

"lat": latitude of the location

"lng": longitude of the location

"locID": unique identifier for the location

"locName": location name

"locationPrivate": TRUE if location is not a birding hotspot

"obsDt": observation date formatted according to ISO 8601 (e.g. 'YYYY-MM-DD', or 'YYYY-MM-DD hh:mm'). Hours and minutes are excluded if the observer did not report an observation time.

"obsReviewed": TRUE if observation has been reviewed, FALSE otherwise

"obsValid": TRUE if observation has been deemed valid by either the automatic filters or a regional viewer, FALSE otherwise

"sciName" species' scientific name

**Author(s)**

Rafael Maia <[rm72@zips.uakron.edu](mailto:rm72@zips.uakron.edu)>

**References**

<http://ebird.org/>

**Examples**

```
## Not run:
ebirdhotspot(locID=c('L99381','L99382'), species='larus delawarensis')
ebirdhotspot('L99381', max=10, provisional=TRUE)

## End(Not run)
```

---

ebirdloc	<i>Recent observations at a locality</i>
----------	--

---

**Description**

Returns the most recent sighting information reported in a given vector of locations (including non-hotspots).

**Usage**

```
ebirdloc(locID, species = NULL, back = NULL, max = NULL, locale = NULL,
         provisional = FALSE, simple = TRUE, sleep = 0, ...)
```

**Arguments**

locID	(required) Vector containing code(s) for up to 10 regions of interest; here, values that are not hotspots are returned. Values that are not valid are ignored.
species	Scientific name of the species of interest (not case sensitive). Defaults to NULL, in which case sightings for all species are returned. See eBird taxonomy for more information: <a href="http://ebird.org/content/ebird/about/ebird-taxonomy">http://ebird.org/content/ebird/about/ebird-taxonomy</a>
back	Number of days back to look for observations (between 1 and 30, defaults to 14).
max	Maximum number of result rows to return in this request (between 1 and 10000, defaults to all)
locale	Language/locale of response (when translations are available). See <a href="http://java.sun.com/javase/6/docs/api/j">http://java.sun.com/javase/6/docs/api/j</a> (defaults to en_US)
provisional	Should flagged records that have not been reviewed be included? (defaults to FALSE)
simple	Logical. Whether to return a simple (TRUE, default) or detailed (FALSE) set of results fields.
sleep	Time (in seconds) before function sends API call (defaults to zero. Set to higher number if you are using this function in a loop with many API calls).
...	Curl options passed on to <a href="#">GET</a>

**Value**

A data.frame containing the collected information:

"comName": species common name

"howMany": number of individuals observed, NA if only presence was noted

"lat": latitude of the location

"lng": longitude of the location

"locID": unique identifier for the location

"locName": location name

"locationPrivate": TRUE if location is not a birding hotspot

"obsDt": observation date formatted according to ISO 8601 (e.g. 'YYYY-MM-DD', or 'YYYY-MM-DD hh:mm'). Hours and minutes are excluded if the observer did not report an observation time.

"obsReviewed": TRUE if observation has been reviewed, FALSE otherwise

"obsValid": TRUE if observation has been deemed valid by either the automatic filters or a regional viewer, FALSE otherwise

"sciName" species' scientific name

"subnational2Code": county code (returned if simple=FALSE)

"subnational2Name": county name (returned if simple=FALSE)

"subnational1Code": state/province ISO code (returned if simple=FALSE)

"subnational1Name": state/province name (returned if simple=FALSE)

"countryCode": country ISO code (returned if simple=FALSE)

"countryName": country name (returned if simple=FALSE)

"userDisplayName": first and last name of the observer (returned if simple=FALSE)

"firstName": observer's first name (returned if simple=FALSE)

"lastName": observer's last name (returned if simple=FALSE)

"subID": submission ID (returned if simple=FALSE)

"obsID": observation ID (returned if simple=FALSE)

"checklistID": checklist ID (returned if simple=FALSE)

"presenceNoted": 'true' if user marked presence but did not count the number of birds. 'false' otherwise (returned if simple=FALSE)

**Author(s)**

Rafael Maia <rm72@zips.uakron.edu>

**References**

<http://ebird.org/>

**Examples**

```
## Not run:
ebirdloc(locID = c('L99381', 'L99382'))
ebirdloc('L99381', 'Branta canadensis', provisional=TRUE)

## End(Not run)
```

---

ebirdnotable	<i>Notable sightings</i>
--------------	--------------------------

---

**Description**

Returns the most recent notable observations by either latitude/longitude, hotspot or location ID, or particular region.

**Usage**

```
ebirdnotable(lat = NULL, lng = NULL, dist = NULL, locID = NULL,
             region = NULL, regtype = NULL, back = NULL, max = NULL,
             locale = NULL, provisional = FALSE, hotspot = FALSE, simple = TRUE,
             sleep = 0, ...)
```

**Arguments**

lat	Decimal latitude. value between -90.00 and 90.00, up to two decimal places of precision.
lng	Decimal longitude. value between -180.00 and 180.00, up to two decimal places of precision.
dist	Distance defining radius of interest from given lat/lng in kilometers (between 0 and 50, defaults to 25)
locID	Vector containing code(s) for up to 10 locations of interest.
region	Region code corresponding to selected region type. For supported region and coding, see <a href="https://confluence.cornell.edu/display/CLOISAPI/eBird-1.1-RegionCodeReference">https://confluence.cornell.edu/display/CLOISAPI/eBird-1.1-RegionCodeReference</a>
regtype	Region type you are interested in. can be "country" (e.g. "US"), "subnational1" (states/provinces, e.g. "US-NV") or "subnational2" (counties, not yet implemented, e.g. "US-NY-109"). Default behavior is to try and match according to the region specified.
back	Number of days back to look for observations (between 1 and 30, defaults to 14).
max	Maximum number of result rows to return in this request (between 1 and 10000, defaults to all).
locale	Language/locale of response (when translations are available). See <a href="http://java.sun.com/javase/6/docs/api/j">http://java.sun.com/javase/6/docs/api/j</a> (defaults to en_US).
provisional	Should flagged records that have not been reviewed be included? (defaults to FALSE)

hotspot	Should results be limited to sightings at birding hotspots? (defaults to FALSE).
simple	Logical. Whether to return a simple (TRUE, default) or detailed (FALSE) set of results fields.
sleep	Time (in seconds) before function sends API call (defaults to zero. Set to higher number if you are using this function in a loop with many API calls).
...	Curl options passed on to <a href="#">GET</a>

### Value

A data.frame containing the collected information:

"comName": species common name

"howMany": number of individuals observed, NA if only presence was noted

"lat": latitude of the location

"lng": longitude of the location

"locID": unique identifier for the location

"locName": location name

"locationPrivate": TRUE if location is not a birding hotspot

"obsDt": observation date formatted according to ISO 8601 (e.g. 'YYYY-MM-DD', or 'YYYY-MM-DD hh:mm'). Hours and minutes are excluded if the observer did not report an observation time.

"obsReviewed": TRUE if observation has been reviewed, FALSE otherwise

"obsValid": TRUE if observation has been deemed valid by either the automatic filters or a regional viewer, FALSE otherwise

"sciName" species' scientific name

"subnational2Code": county code (returned if simple=FALSE)

"subnational2Name": county name (returned if simple=FALSE)

"subnational1Code": state/province ISO code (returned if simple=FALSE)

"subnational1Name": state/province name (returned if simple=FALSE)

"countryCode": country ISO code (returned if simple=FALSE)

"countryName": country name (returned if simple=FALSE)

"firstName": observer's first name (returned if simple=FALSE)

"lastName": observer's last name (returned if simple=FALSE)

"subID": submission ID (returned if simple=FALSE)

"obsID": observation ID (returned if simple=FALSE)

"checklistID": checklist ID (returned if simple=FALSE)

"presenceNoted": 'true' if user marked presence but did not count the number of birds. 'false' otherwise (returned if simple=FALSE)

**Note**

ebirdnotable requires that either latitude/longitude, location ID, or region be passed to the function. Multiple entries will result in the most specific being used. If none is supplied, defaults to lat/lng based on your IP.

**Author(s)**

Rafael Maia <rm72@zips.uakron.edu>

**References**

<http://ebird.org/>

**Examples**

```
## Not run:
ebirdnotable(lat=42, lng=-70)
ebirdnotable(region='US', max=10)
ebirdnotable(region='US-OH', regtype='subnational1')

## End(Not run)
```

---

 ebirdregion

*Recent observations at a region*


---

**Description**

Returns the most recent sighting information reported in a given region.

**Usage**

```
ebirdregion(region, species = NULL, regtype = NULL, back = NULL,
  max = NULL, locale = NULL, provisional = FALSE, hotspot = FALSE,
  sleep = 0, ...)
```

**Arguments**

region	(required) Region code corresponding to selected region type. For supported region and coding, see <a href="https://confluence.cornell.edu/display/CLOISAPI/eBird-1.1-RegionCodeReference">https://confluence.cornell.edu/display/CLOISAPI/eBird-1.1-RegionCodeReference</a>
species	scientific name of the species of interest (not case sensitive). Defaults to NULL, in which case sightings for all species are returned. See eBird taxonomy for more information: <a href="http://ebird.org/content/ebird/about/ebird-taxonomy">http://ebird.org/content/ebird/about/ebird-taxonomy</a>
regtype	Region type you are interested in. can be "country" (e.g. "US"), "subnational1" (states/provinces, e.g. "US-NV") or "subnational" (counties, not yet implemented, e.g. "US-NY-109"). Default behavior is to try and match according to the region specified.

back	Number of days back to look for observations (between 1 and 30, defaults to 14).
max	Maximum number of result rows to return in this request (between 1 and 10000, defaults to all)
locale	Language/locale of response (when translations are available). See <a href="http://java.sun.com/javase/6/docs/api/j">http://java.sun.com/javase/6/docs/api/j</a> (defaults to en_US)
provisional	Should flagged records that have not been reviewed be included? (defaults to FALSE)
hotspot	Should results be limited to sightings at birding hotspots? (defaults to FALSE).
sleep	Time (in seconds) before function sends API call (defaults to zero. Set to higher number if you are using this function in a loop with many API calls).
...	Curl options passed on to <a href="#">GET</a>

**Value**

A data.frame containing the collected information:

"comName": species common name

"howMany": number of individuals observed, NA if only presence was noted

"lat": latitude of the location

"lng": longitude of the location

"locID": unique identifier for the location

"locName": location name

"locationPrivate": TRUE if location is not a birding hotspot

"obsDt": observation date formatted according to ISO 8601 (e.g. 'YYYY-MM-DD', or 'YYYY-MM-DD hh:mm'). Hours and minutes are excluded if the observer did not report an observation time.

"obsReviewed": TRUE if observation has been reviewed, FALSE otherwise

"obsValid": TRUE if observation has been deemed valid by either the automatic filters or a regional viewer, FALSE otherwise

"sciName" species' scientific name

**Author(s)**

Rafael Maia <rm72@zips.uakron.edu>

**References**

<http://ebird.org/>

**Examples**

```
## Not run:
ebirdregion(region = 'US', species = 'Setophaga caerulescens')
ebirdregion('US-OH', max=10, provisional=TRUE, hotspot=TRUE)

## End(Not run)
```

---

ebirdregioncheck	<i>Check if a region type is valid</i>
------------------	--

---

**Description**

Check if a region type is valid

**Usage**

```
ebirdregioncheck(loctype, loc, ...)
```

**Arguments**

loctype	One of: 'country', 'states', 'counties'
loc	The location code to be checked. A single location only, unless loctype = 'country'.
...	Curl options passed on to <a href="#">GET</a>

**Value**

A logical vector of the same length as loc.

**Author(s)**

Sebastian Pardo <sebpardo@gmail.com>, Andy Teucher <andy.teucher@gmail.com>

**References**

<http://ebird.org/>

**Examples**

```
## Not run:  
ebirdregioncheck("country", c("US", "CA"))  
ebirdregioncheck("states", "CA-BC")  
ebirdregioncheck("counties", "CA-BC-GV")  
  
## End(Not run)
```

---

ebirdtaxonomy

*eBird Taxonomy*

---

### Description

Returns a data.frame of all species in the eBird taxonomy for the given combination of categories. The default category is "species". Any taxon with the category of 'species' may be used as a parameter in service calls that take a scientific name. Any taxon not in this category will be rejected by these services at this time.

### Usage

```
ebirdtaxonomy(cat = NULL, locale = NULL, ...)
```

### Arguments

cat	Species category. String or character vector with one of more of: "domestic", "form", "hybrid", "intergrade", "issf", "slash", "species", "spuh". For more info about the meaning of species categories, see <a href="http://ebird.org/content/ebird/about/ebird-taxonomy">http://ebird.org/content/ebird/about/ebird-taxonomy</a>
locale	Language/locale of response (when translations are available). See <a href="http://java.sun.com/javase/6/docs/api/java/util/Locale.html">http://java.sun.com/javase/6/docs/api/java/util/Locale.html</a> (defaults to en_US).
...	Curl options passed on to <a href="#">GET</a>

### Value

A data.frame containing the collected information:

"comName": species' common name

"sciName": species' scientific name

"taxonID": Taxonomic Concept identifier, note this is currently in test

### Author(s)

Andy Teucher <[andy.teucher@gmail.com](mailto:andy.teucher@gmail.com)>

### References

<http://ebird.org/>

### Examples

```
## Not run:  
ebirdtaxonomy()  
ebirdtaxonomy(cat=c("spuh", "slash"))  
  
## End(Not run)
```

---

getlatlng	<i>get latitude and longitude from ip address</i>
-----------	---

---

**Description**

Returns the most recent and nearest reported sighting information with observations of a species.

**Usage**

```
getlatlng()
```

**Value**

a vector of length 2 with lat, lng in that order

**Author(s)**

Andy Teucher <andy.teucher@gmail.com>

**References**

<http://ipinfo.io>

**Examples**

```
## Not run:  
getlatlng()  
  
## End(Not run)
```

---

nearestobs	<i>Nearest species sightings</i>
------------	----------------------------------

---

**Description**

Returns the most recent and nearest reported sighting information with observations of a species.

**Usage**

```
nearestobs(species, lat = NULL, lng = NULL, back = NULL, max = NULL,  
           locale = NULL, provisional = FALSE, hotspot = FALSE, sleep = 0, ...)
```

**Arguments**

species	(required) Scientific name of the species of interest (not case sensitive). See eBird taxonomy for more information: <a href="http://ebird.org/content/ebird/about/ebird-taxonomy">http://ebird.org/content/ebird/about/ebird-taxonomy</a>
lat	Decimal latitude. value between -90.00 and 90.00, up to two decimal places of precision. Defaults to latitude based on IP.
lng	Decimal longitude. value between -180.00 and 180.00, up to two decimal places of precision. Defaults to longitude based on IP.
back	Number of days back to look for observations (between 1 and 30, defaults to 14).
max	Maximum number of result rows to return in this request (between 1 and 10000, defaults to all).
locale	Language/locale of response (when translations are available). See <a href="http://java.sun.com/javase/6/docs/api/j">http://java.sun.com/javase/6/docs/api/j</a> (defaults to en_US).
provisional	Should flagged records that have not been reviewed be included? (defaults to FALSE).
hotspot	Should results be limited to sightings at birding hotspots? (defaults to FALSE).
sleep	Time (in seconds) before function sends API call (defaults to zero. Set to higher number if you are using this function in a loop with many API calls).
...	Curl options passed on to <a href="#">GET</a>

**Value**

A data.frame containing the collected information:

"comName": species common name

"howMany": number of individuals observed, NA if only presence was noted

"lat": latitude of the location.

"lng": longitude of the location.

"locID": unique identifier for the location

"locName": location name

"locationPrivate": TRUE if location is not a birding hotspot

"obsDt": observation date formatted according to ISO 8601 (e.g. 'YYYY-MM-DD', or 'YYYY-MM-DD hh:mm'). Hours and minutes are excluded if the observer did not report an observation time.

"obsReviewed": TRUE if observation has been reviewed, FALSE otherwise

"obsValid": TRUE if observation has been deemed valid by either the automatic filters or a regional viewer, FALSE otherwise

"sciName" species' scientific name

**Author(s)**

Rafael Maia <rm72@zips.uakron.edu>

## References

<http://ebird.org/>

## Examples

```
## Not run:  
nearestobs('spizella arborea', 42, -76)  
nearestobs('spizella arborea', 42,-76, max=10, provisional=TRUE, hotspot=TRUE)  
  
## End(Not run)
```

# Index

## \*Topic **package**

rebird-package, [2](#)

ebirdfreq, [2](#)

ebirdgeo, [4](#)

ebirdhotspot, [5](#)

ebirdloc, [7](#)

ebirdnotable, [9](#)

ebirdregion, [11](#)

ebirdregioncheck, [13](#)

ebirdtaxonomy, [14](#)

GET, [3](#), [4](#), [6](#), [7](#), [10](#), [12–14](#), [16](#)

getlatlng, [15](#)

nearestobs, [15](#)

rebird (rebird-package), [2](#)

rebird-package, [2](#)