

# Package: territoria (via r-universe)

August 11, 2024

**Type** Package

**Title** Clustering Observations from Breeding Birds into Territoria

**Version** 0.0.3

**Description** Clusters individual observations based on breeding indication and distance between observations.

**License** GPL-3

**URL** <https://github.com/inbo/territoria>

**BugReports** <https://github.com/inbo/territoria/issues>

**Imports** assertthat, mvtnorm, RSQLite, spatstat.geom, spatstat.random

**Config/checklist/communities** inbo

**Config/checklist/keywords** breeding bird; cluster

**Encoding** UTF-8

**Language** en-GB

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.3

**Repository** <https://inbo.r-universe.dev>

**RemoteUrl** <https://github.com/inbo/territoria>

**RemoteRef** HEAD

**RemoteSha** 23e08ce479aefe17928d61c74e6a0bb8c6b249a5

## Contents

cluster_observation . . . . .	2
connect_db . . . . .	2
distance_matrix . . . . .	3
get_cluster . . . . .	3
import_observations . . . . .	3
simulate_observations . . . . .	4

<b>Index</b>	<b>5</b>
--------------	----------

---

cluster\_observation     *Cluster observations*

---

**Description**

Cluster observations

**Usage**

```
cluster_observation(conn, status, max_dist = 336, reset = FALSE)
```

**Arguments**

conn	a DBI connection to an SQLite database.
status	highest status to include while clustering the observations.
max_dist	maximum clustering distance in m.
reset	reset the current clustering. Defaults to FALSE

---

connect\_db                 *Connect to a database*

---

**Description**

Connect to a database

**Usage**

```
connect_db(db = ":memory:")
```

**Arguments**

db	The file name of the database
----	-------------------------------

---

distance_matrix	<i>Calculate the distance matrix</i>
-----------------	--------------------------------------

---

**Description**

Calculate the distance matrix

**Usage**

```
distance_matrix(conn, max_dist = 336)
```

**Arguments**

conn	a DBI connection to an SQLite database.
max_dist	maximum clustering distance in m.

---

get_cluster	<i>Get the information from the clusters</i>
-------------	--

---

**Description**

Get the information from the clusters

**Usage**

```
get_cluster(conn)
```

**Arguments**

conn	a DBI connection to an SQLite database.
------	---

---

import_observations	<i>Import the observations</i>
---------------------	--------------------------------

---

**Description**

The function overwrites any existing table with observations.

**Usage**

```
import_observations(observations, conn, max_dist = 336)
```

**Arguments**

observations	a data.frame with the observations.
conn	a DBI connection to an SQLite database.
max_dist	maximum clustering distance in m.

---

simulate\_observations *Simulate a dataset in a square area*

---

### Description

Simulate a dataset in a square area

### Usage

```
simulate_observations(  
  density = 1e-05,  
  area = 4e+06,  
  gamma = 0.5,  
  max_dist = 336,  
  n_survey = 4,  
  p_detection = 0.6,  
  status_distribution = c(0.2, 0.5, 0.3)  
)
```

### Arguments

density	Density as the number of territoria per m <sup>2</sup> .
area	Area in which to simulate territoria in m <sup>2</sup> .
gamma	interaction parameter of the Strauss process. See gamma in spatstat.core::rStrauss().
max_dist	maximum clustering distance in m.
n_survey	Number of surveys.
p_detection	Probability of detection within a survey.
status_distribution	a weighting vector for statuses. The order of the vector is the number of the status.

# Index

cluster\_observation, 2  
connect\_db, 2  
distance\_matrix, 3  
get\_cluster, 3  
import\_observations, 3  
simulate\_observations, 4