ResDisMapper v1.0 User Manual

Qian Tang, Tak Fung and Frank E. Rheindt

Nov 2019

Introduction

Generally, this *R* package is used to map resistance to dispersal on the basis of individual-based genetic distance. The package uses the principle of Isolation by distance (IBD), which is widely used in population genetic studies to quantify natural dispersal. Specifically, the package models IBD as the expected relationship between genetic and geographical distances for individuals in a population or set of geographically continuous populations, and deviations from this general trend (IBD residual, Keis et al., 2013) are used to map resistance to dispersal. Positive IBD residuals accumulate in areas which have a relatively high resistance to dispersal and vice versa. Resistance values calculated from IBD residuals are relative values, such that high resistance values do not necessarily indicate a barrier and low resistance values do not necessarily indicate a corridor. Please interpret the results based on the hypothesis of your study. For example, in a biological invasion scenario, areas with negative resistance values can be interpreted as areas with human-facilitated dispersal of an invasive species, whereas in a habitat fragmentation scenario, areas with negative resistance values can be interpreted as remaining "corridors" of habitat that have low resistance to dispersal compared with areas cleared of habitat.

Who may find it useful

This *R* package is designed to study the effects of landscape or other environmental features on the dispersal of individuals within a relatively small area. All studied individuals should come from a single population or a set of geographically continuous populations. Because the *R* package uses IBD residuals to indicate the resistance to dispersal, inputting multiple isolated (highly diverged) populations or species can confuse the IBD modelling as different populations may have different IBD trends. Therefore, please check PCA plots to make sure that there are no significantly diverged clusters before using the *R* package.

To guarantee the accuracy of the IBD model and subsequent analyses, we recommend that users have ensured that there is a continuous distribution of geographical distance among samples. Therefore, a sampling scheme of even distribution across the study area is highly recommended.

How to cite

Qian, T., Fung, T., Rheindt, F.E. *ResDisMapper*: An *R* package for fine-scale mapping of resistance to dispersal. *Molecular Ecology Resources*.

Contributors

Tang Qian <tangqiannus@gmail.com>

Tak Fung <tfung2000@gmail.com>

Frank E. Rheindt <dbsrfe@nus.edu.sg>

Installation

The *R* package is available on GitHub (https://github.com/takfung/ResDisMapper).

ResDisMapper can be installed from GitHub by typing the following line in *R*, which uses the *R* package devtools:

```
devtools::install_github("takfung/ResDisMapper")
```

This line should also install *R* packages that *ResDisMapper* depends upon if these are not already installed on your computer. The *R* packages that *ResDisMapper* directly depends upon are adegenet, ggplot2, poppr, raster, rgl, Rmisc and sp. If the installation of a dependent *R* package fails and is still missing after running the line above, please install the package using the install.packages function.

Input data

The *R* package is designed for multilocus data, such as SNPs, AFLPs and microsatellites. The *R* package intake is in the form of two files, a genetic data file in GENEPOP format and a geographical data file containing tab-delimited coordinates of sampling localities.

Missing data

The *R* package can handle missing genetic data. Missing data generate NAs in the genetic distance matrix and hence in the IBD residuals. In the *R* package, all NAs in the calculated IBD residuals are

removed in the rdm_residual function. For other ways of handling missing genetic data, please refer to the documentation of poppr *R* package (Kamvar et al., 2014). We caution that a high level of missing genetic data will substantially affect the accuracy of the results.

Workflow

For one dataset, the entire flow of analyses using *ResDisMapper* will typically take about one to a few hours, but the actual time depends on the amount of data and the proposed resolution of the resistance map. Below, we provide the general workflow for the application of *ResDisMapper*, together with a specific example application to a dataset for rock pigeons in Singapore (for a description of this dataset, see Tang et al., 2018). This dataset is stored at https://github.com/takfung/ResDisMapper/tree/master/Example_datasets, as the two files "Pigeon.gen" and "Pigeon.coordinates". Each file can be downloaded directly by accessing the link, clicking on the name of the file, clicking the "Raw" icon with the right mouse button, and then selecting "Save Link As". Alternatively, each file can be downloaded using the function download.file in *R* together with the url for the file. In addition to the pigeon dataset, an example dataset for Golden-crowned Sifakas in northern Madagascar is also stored in the same place, as the two files "Sifaka.gen" and "Sifaka.coordinates". In each step of the workflow, we describe the tasks performed and detail any functions from *ResDisMapper* that are used to perform the tasks. We also show a specific application of the *ResDisMapper* functions to our example pigeon dataset, with accompanying figures that show graphs drawn using output from the functions.

Step 1. Prepare your data according to the required input format. The data should be compiled as two files that contain the genotype data and geographical locations of the sampled individuals. Specifically, the genotype data needs to be in a GENEPOP file with extension *.gen*. In this GENEPOP file, alleles are specified using three digits. The geographical locations need to be in a text file with the first column showing the name of each individual and the second and third columns showing the corresponding geographical x and y coordinates. We recommend that projected coordinates be used instead of longitude/latitude coordinates, because projected coordinates are more commonly used at relatively small spatial scales and correspond to a unit of distance that is more easily interpretable (i.e., a meter or a multiple of a meter). If your sampling localities are in longitude/latitude coordinates, we recommend that you convert them to projected coordinates using online tools (for example, https://www.latlong.net/lat-long-utm.html) or the *R* function spTransform in the *R* package rdgal.

The *i*th row in each of the two files needs to correspond to the *i*th sampled individual (i.e. the *i*th row corresponds to the same sampled individual). If this is not the case, then the rows of the geographical

location file need to be rearranged appropriately, for example using *R*. We have uploaded an *R* script showing an example of how the rows of the geographical location file can be rearranged appropriately (https://github.com/takfung/ResDisMapper/tree/master/Documentation/sort_rows.R).

Step 2. Calculate the genetic and geographical distances for each pair of sampled individuals from the data, fit an expected IBD trend to the genetic and geographical distances, visualize the genetic and geographical distances together with the expected IBD trend, and use the expected IBD trend to calculate the IBD residuals for each data point (pair of genetic and geographical distances). These tasks can be performed by loading *ResDisMapper* using

```
library(ResDisMapper)
```

and then using the function rdm_IBD. There are four arguments in this function:

Gen_raw: A string specifying the path to a file of genotype data in GENEPOP format, with alleles specified using three digits. The GENEPOP file must have the extension *.gen*.

Geo_raw: A string specifying the path to a text file showing the name (1st column) and geographical x and y coordinates (2nd and 3rd columns) of each individual (each row). The first row needs to specify the column headings, and all entries need to be tab-delimited.

Dist_method: An integer from 1 to 6 specifying the method used to calculate genetic distance among individuals. Integers 1 to 6 correspond to the methods incorporated into the following six functions from the poppr *R* package, respectively: diss.dist, nei.dist, rogers.dist, reynolds.dist, edwards.dist, and provesti.dist. The default value is 1.

IBD_method: An integer specifying the method used to calculate the IBD residual for each pair of individuals. A value of 1 means that the residuals are calculated by fitting a straight line ($y = ax + b$, where y is the genetic distance; x is the geographical distance; and a and b are the fitted parameters) to all pairs of genetic and geographical distances, and measuring the distance from each pair to the fitted line. The fitted straight line represents the combinations of genetic and geographical distances that are expected from IBD. A value of 2 means that a non-linear curve of the form $y = a + b\,(1\text{-}\exp(\text{-}\exp(c)\,x))$ is fitted instead of a straight line, where y is the genetic distance; x is the geographical distance; and a, b and c are the fitted parameters. The default value is 1. The function rdm_IBD automatically plots the data points together with the fitted line or curve.

After running rdm_IBD, an object of class dist, containing a matrix showing the IBD residuals for pairs of individuals, is output. Also, a scatter plot of the IBD model will be presented. For our example pigeon dataset, the function can be run using

```
IBD.res <- rdm_IBD(Gen_raw = "files/Pigeon.gen", Geo_raw = "files/Pigeon.coordinates", Dist_method = 4, IBD_method = 1)
```

to calculate the IBD residuals using method 4 to calculate the genetic distances and a straight line as the expected IBD trend. Here, "files" is the path to the folder containing the pigeon dataset. We recommend that users run rdm_IBD with all available methods of genetic distance calculation, as we do for our example pigeon dataset (Fig. 1).

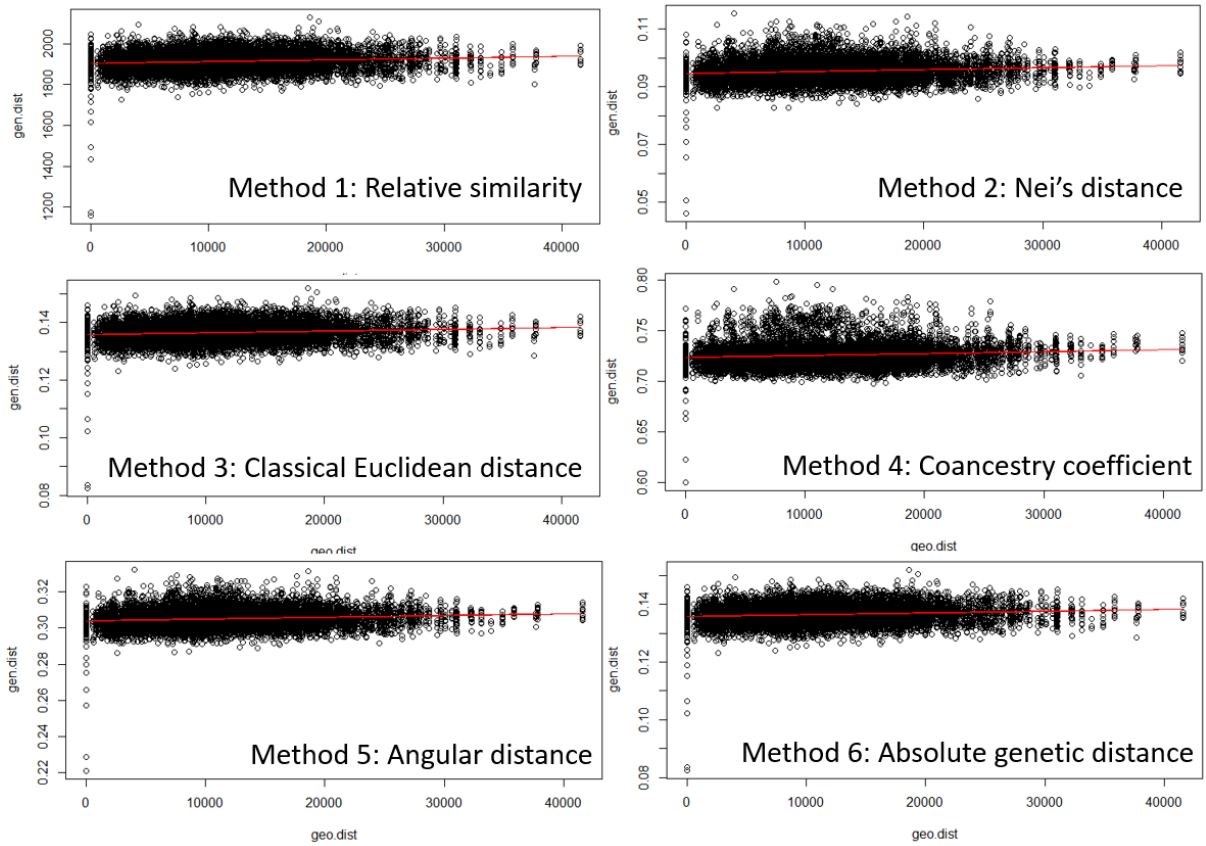

Fig. 1. Plots of genetic distance against geographic distance, produced by running rdm_IBD with the example pigeon dataset and each of the six available methods of calculating genetic distance. The black dots refer to the data points. A linear model is used for the expected IBD trend.

For each method of calculating genetic distance, we also recommend running rdm_IBD with the two methods of modelling the expected IBD trend, using a linear or non-linear model. For example, using Method 4 for calculating genetic distance, we run rdm_IBD using the pigeon dataset and the two methods of modelling the expected IBD trend (Fig. 2).

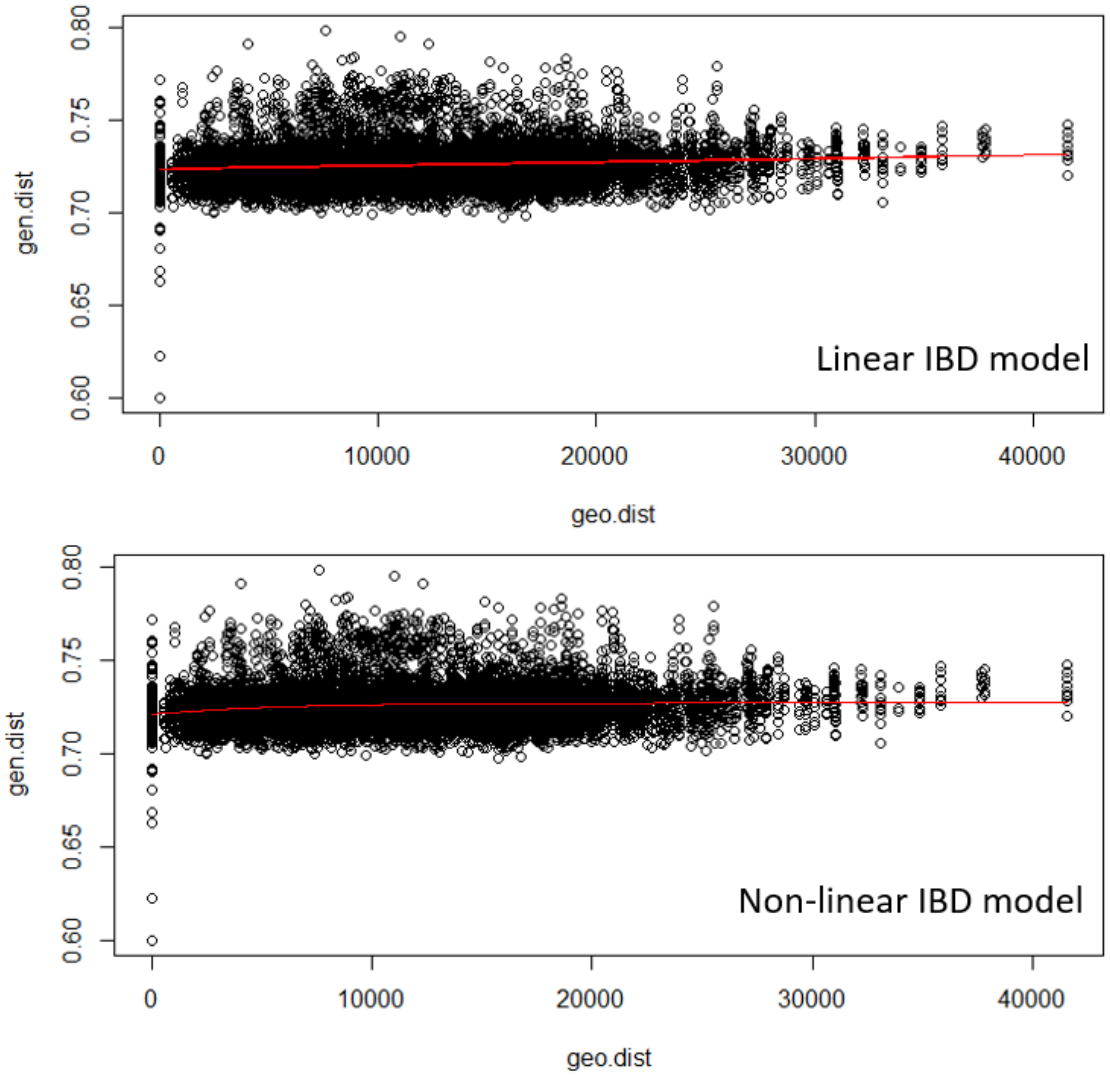

Fig. 2. Plots of genetic distance against geographic distance, produced by running rdm_IBD with the example pigeon dataset, Method 4 for calculating the genetic distance, and the two different methods of modelling the expected IBD trend (linear and non-linear model). The black dots refer to the data points, whereas the red lines refer to the fitted IBD model.

After running rdm_IBD with the six different methods of calculating genetic distance and two methods of modelling the expected IBD trend, 12 sets of IBD residuals are produced. Further analyses can be performed using each of these 12 sets of IBD residuals, or on a subset of these 12 sets. For our example using the pigeon dataset, we perform further analyses using the IBD residuals generated by running rdm_IBD using Method 4 for calculating genetic distance and a linear model for IBD.

Step 3. Visualize the distribution of IBD residuals in the form of line segments over the landscape area. This can be done by running the function rdm_residual in *ResDisMapper*. There are six arguments in the function rdm_residual:

IBD.res: A distance matrix of pairwise IBD residuals output from rdm_IBD.

Geo_raw: A string specifying the path to a text file showing the name (1st column) and geographical x and y coordinates (2nd and 3rd columns) of each individual (each row). The first row needs to specify the column headings, and all entries need to be tab-delimited.

min.dist: IBD residuals are only calculated for pairs of individuals that are separated by a distance greater than min.dist. The default value is 1.

max.dist: IBD residuals are only calculated for pairs of individuals that are separated by a distance less than max.dist. The default value is Inf.

n_resolution: Specifies the number of cells that the landscape is divided into, for the purposes of creating the 3-D plot. The landscape is divided into n_resolution cells along both coordinate axes. Individuals that appear in the same cell are represented by a single point on the plot. The default value is 50.

proj: The coordinate system that is used for plotting. The default is EPSG 4326, which corresponds to WGS 84.

After running rdm_residual, an object of class SpatialLinesDataFrame, containing coordinates of the line segments joining each pair of individuals and the corresponding IBD residuals, is output. Also, a 3-D plot of the spatial lines data frame will be presented, which can be rotated. If the plot of genetic distance against geographical distance in Step 2 shows a discrete pattern, or if the dispersal range is known for the population considered, users can specify the geographical distance range (min.dist and max.dist) within which line segments are considered for calculation of resistance in Step 4. The 3-D plot of IBD residuals can also be used to decide what resolution the grid representing the landscape considered should be, which is used in calculating resistance in Step 4. Users need to make sure that most of the grid cells have enough intersecting line segments to provide a largely continuous resistance map in Step 5.

For our pigeon dataset, we run rdm_residual using

```
Res_SLDF <- rdm_residual(IBD.res = IBD.res, Geo_raw = "files/Pigeon.coordinates", min.dist = 1, max.dist = 10000, n_resolution = 50, proj = sp::CRS("+init=epsg:4326"))
```

to visualize the IBD residuals with max.dist = 10000 and n_resolution = 50 (Figs. 3 and 4). The value of max.dist corresponds to 10 km, which is the dispersal range of rock pigeons in Singapore as derived from a spatial autocorrelation analysis. We also visualized the IBD residuals using max.dist = Inf (Fig. 3) and n_resolution = 20 (Fig. 4). Using different values of n_resolution gives a rough idea of the number of grid cells that can be defined without producing many discontinuities in the resistance map (due to grid cells with no intersecting line segments).

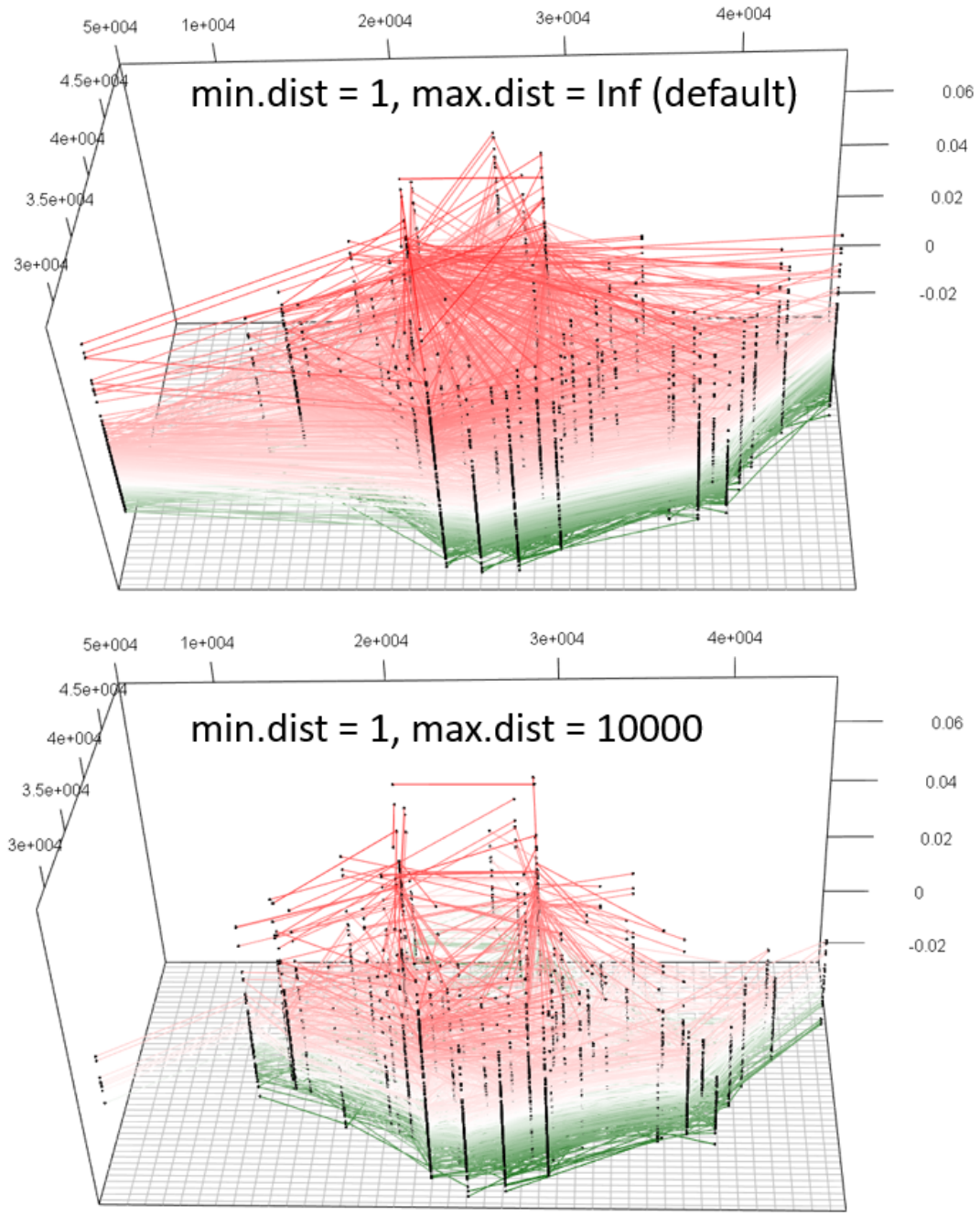

Fig. 3. 3-D plot visualization of IBD residuals as line segments, produced by running rdm_residual with the example pigeon dataset and with different geographical distance ranges. Only line segments within the geographical distance range specified are plotted. Line segments with different colors from dark green to dark red correspond to IBD residuals that are increasingly positive. The vertical axis refers to the values of the residuals.

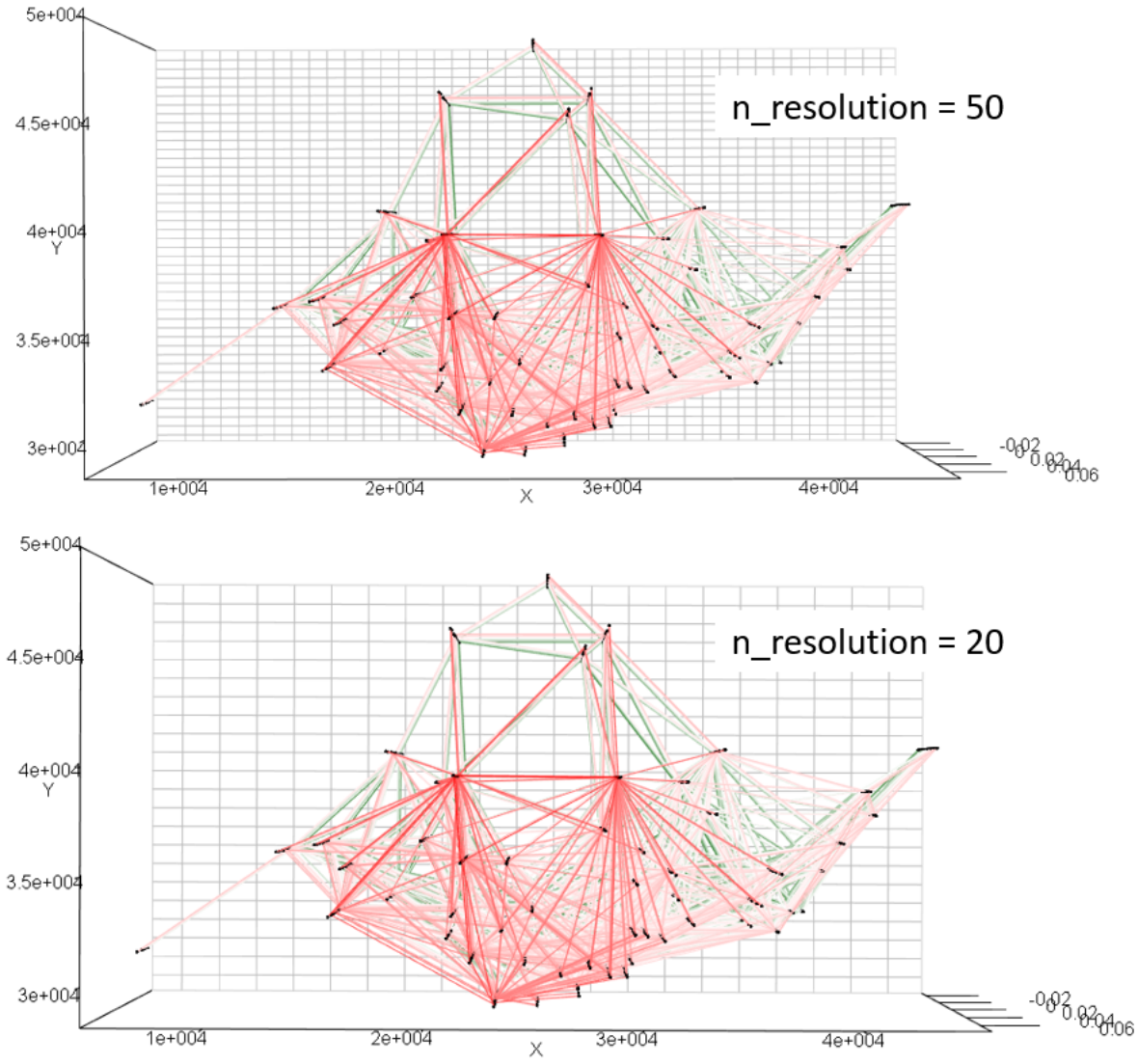

Fig. 4. 3-D plot visualization of IBD residuals as line segments, produced by running rdm_residual with the example pigeon dataset and with max.dist = 10000 and different grid size resolutions (number of cells in each row or column). Line segments with different colors from dark green to dark red correspond to IBD residuals that are increasingly positive. The *z* axis refers to the values of the residuals.

For further analyses, we use the SpatialLinesDataFrame object corresponding to max.dist = 10000.

Step 4. Calculate the resistance values over the landscape considered, together with whether the resistance values are statistically different from null values and are statistically different from zero. This can be done using the function rdm_resistance in *ResDisMapper*. There are seven arguments in the function rdm_resistance:

IBD.res: A distance matrix of pairwise IBD residuals output from rdm_IBD.

Res_SLDF: An object of class SpatialLinesDataFrame, containing the IBD residuals for each pair of individuals in a population and the coordinates of line segments connecting the pairs of individuals over the defined landscape. This object is produced by the function rdm_residual.

nrows: Number of grid cells in each row of the raster map of resistance to dispersal. The default value is 30.

ncols: Number of grid cells in each column of the raster map of resistance to dispersal. The default value is 30.

conf_intervals: The coverage of the confidence interval generated for each resistance value in a grid cell, expressed as a proportion. This interval measures the uncertainty in the (observed) resistance in each cell. If a cell has no intersecting line segments or only one, then a confidence interval cannot be calculated (a warning message is produced) and the cell is no longer considered for further calculation. If the confidence interval does not overlap 0, then the resistance is statistically different from 0. The default value is 0.95 (corresponding to 95% intervals).

random_rep: Number of random resamples of the IBD residuals used to construct the null distribution of resistances in each grid cell. For a grid cell with *n* intersecting line segments and *n* corresponding IBD residuals, a random resample consists of randomly sampling *n* IBD residuals from the set of all IBD residuals over the entire landscape, without replacement. The default value is 1,000.

outputfile: A string specifying the path to and name of the .csv file that is produced, containing resistance information in the output data frame. The default is 'resistance_map.csv'.

The function rdm_resistance takes as input an object of class dist, which is the output of Step 2, and an object of class SpatialLinesDataFrame, which is the output of Step 3. We would recommend you choose the resolution of the resistance raster map according to the density and distribution of your samples. The function also requires a name for the output file (in .csv format), including all useful information of the resistance raster map. The other options in this function (coverage of confidence intervals and number of random resamples for the generation of null distributions of resistances) can generally be left as the default values unless you have a particular reason to use different values.

After running rdm_resistance, a data frame with resistance information for each grid cell in the landscape with more than one intersecting line segment is output. Information from this data frame corresponds to a raster map of resistance values. In the data frame, each row corresponds to one cell, and the eight different columns refer to the resistance, the number of intersecting line segments, the x and y coordinates specifying the location of the mid-point of the cell, the lower and upper limits of the confidence interval for the resistance, the sign of the product of the lower and upper limits of the confidence interval, and the percentile of the null distribution of resistances corresponding to the observed resistance. When calculating this data, the function rdm_resistance prints out messages

showing the current stage of calculation. There are four stages: Calculating (1) the resistances, (2) the numbers of intersecting line segments, (3) the lower limits of the confidence intervals for the resistances, and (4) the upper limits of the confidence intervals for the resistances. The information in this data frame is reproduced in the .csv output file.

For our example pigeon dataset, we run rdm_resistance using

```
F.df <- rdm_resistance(IBD.res = IBD.res, Res_SLDF = Res_SLDF, nrows = 25, ncols = 50,
conf_intervals = 0.95, random_rep = 1000, outputfile = "files/resistance_map.csv")
```

Running rdm_resistance typically takes about an hour to a few hours, so you can monitor progress in the R console (Fig. 5).

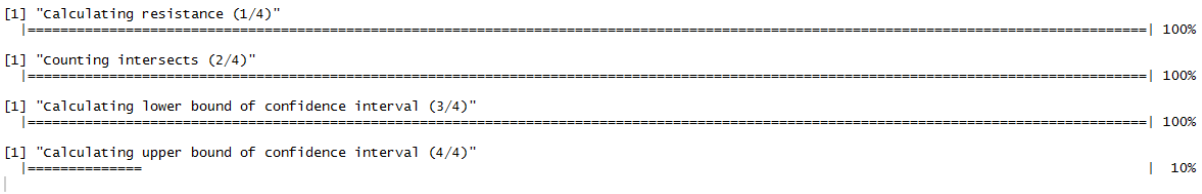

Fig. 5. Screen shot of R console while processing a call of rdm_resistance using the example pigeon dataset.

Step 5. Visualize the resistance map using the output from Step 4. This can be done using the function rdm_mapper in *ResDisMapper*. There are eight arguments in the function rdm_mapper:

F.df: A data frame containing eight columns with resistance information for each grid cell in a landscape. Each row corresponds to one cell, and the eight different columns refer to the resistance, the number of intersecting line segments, the x and y coordinates specifying the location of the mid-point of the cell, the lower and upper limits of the confidence interval for the resistance, the sign of the product of the lower and upper limits of the confidence interval, and the percentile of the null distribution of resistances corresponding to the observed resistance. This object is produced by the function rdm_resistance.

Geo_raw: A string specifying the path to a text file showing the name (1st column) and geographical x and y coordinates (2nd and 3rd columns) of each individual (each row). The first row needs to specify the column headings, and all entries need to be tab-delimited.

r_size: Specifies the size of each grid cell in the plotted raster map of resistances, which can be adjusted to match the size of the plotted landscape. Default value of 5.

p_signf: Specifies the percentiles of the null distribution of resistances used to define statistically significant high or low resistance values. An area of statistically significant high resistance to dispersal is a group of grid cells with a resistance that is above the 100×(1-p_signf) % percentile,

whereas an area of statistically significant low resistance to dispersal is a group of grid cells with a resistance that is below the 100×(p_signf)% percentile. Default value of 0.05.

p_size: Specifies the size of the sampling points in the plotted raster map of resistances. Default value of 2.

p_col: Specifies the color of the sampling points in the plotted raster map of resistances. Default color is yellow.

disp_all_cells: Specifies whether to display all cells or just cells with resistance values that are statistically different from 0, corresponding to a value of 1 or 0, respectively. Default value is 0.

disp_contours: Specifies whether to display contour lines delineating cells with statistically significant resistance values, corresponding to a value of 1 or 0, respectively. Default value is 1.

The input data frame can be directly set as the output of the function rdm_resistance in Step 4, or imported from the .csv file produced by the function. In the map, you will see cells of different colors ranging from red to green. The colors indicate different levels of resistance, with red referring to high resistance and green referring to low resistance. Red and green contour lines can be drawn on the map, delineating areas of high and low resistance to dispersal that are statistically significant, respectively. These correspond to areas with resistance values that are higher or lower than those from a null distribution with high probability. If a contour line encounters a grid cell with no data (no intersecting line segments) for calculating a resistance value, then the line terminates without forming a loop. In addition, there is an option to display only cells with resistances that are statistically different from 0.

For our example pigeon dataset, we visualize the resistance values over Singapore using

```
rdm_mapper(F.df = F.df, Geo_raw = "files/Pigeon.coordinates", r_size = 5, p_signf = 0.05, p_size = 2, p_col = "black", disp_all_cells = 0, disp_contours = 1)
```

Fig. 6 shows an annotated interpretation of the resulting resistance map.

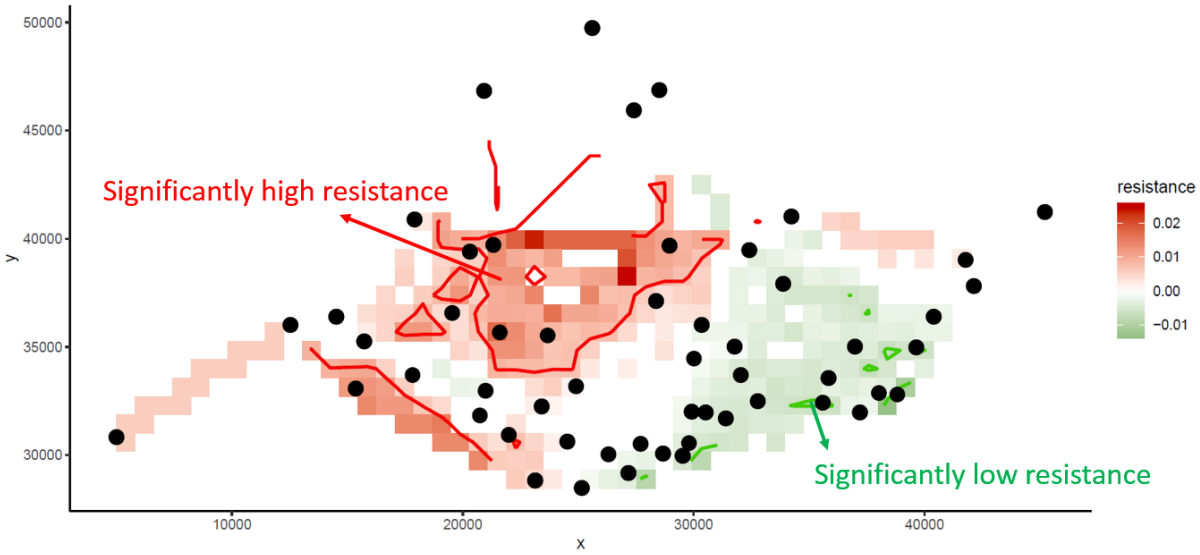

Fig. 6. Resistance map produced by the function rdm_mapper with the example pigeon dataset and max.dist = 10000. The annotations describe the meaning of the different colors and contours. Areas that lie within the red/green contours refer to areas with resistance values that are higher/lower than those from a null distribution with high probability ("statistical significance"). Only cells with resistance values that are statistically different from 0 ("statistical certainty") are displayed. The black circles indicate sampling points.

We note that for the pigeon dataset, using different resolutions changes the pattern of resistance quantitatively but not qualitatively (Fig. 7).

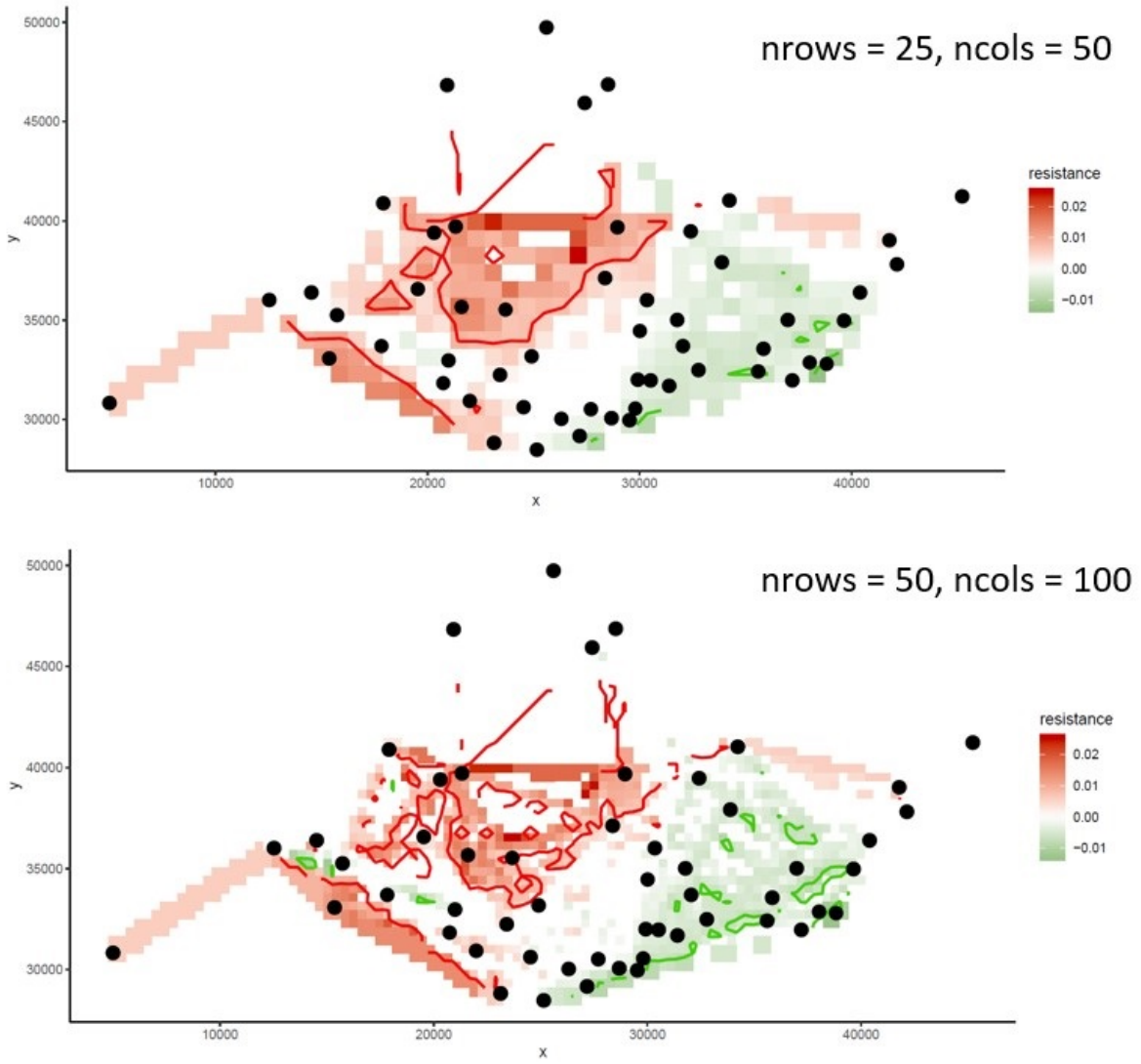

Fig. 7. Resistance maps produced by the function rdm_mapper with the example pigeon dataset, with max.dist = 10,000 and different resolutions. The black circles indicate sampling points. The bottom panel was drawn using a data frame created by running the *R* code in Step 4 with nrows = 50 and ncols = 100.

If you wish to further customize the resistance map, for example to change the colors of the resistance values, we recommend you write your own function with use of ggplot2. In addition, rdm_mapper does not support visualization of the resistance map together with geographic maps that show landscape features. However, the resistance map produced by rdm_mapper can be integrated with geographic maps using other programs. We recommend two ways of performing this integration: (1) The resistance and geographic maps can be exported as vector graphics (e.g. pdf files) and overlaid using a graphics editor (e.g. Adobe Illustrator); or (2) The .csv file containing the underlying resistance values and associated information (produced in Step 4) can be imported into *R* together with the geographic maps, and then overlaid using *R* functions such as those in ggplot2.

Citations

Kamvar ZN, Tabima JF, Grünwald NJ. (2014) Poppr: an R package for genetic analysis of populations with clonal, partially clonal, and/or sexual reproduction. *PeerJ* 2:e281.

Keis, M., Remm, J., Ho, S. Y., Davison, J., Tammeleht, E., Tumanov, I. L., ... & Margus, T. (2013). Complete mitochondrial genomes and a novel spatial genetic method reveal cryptic phylogeographical structure and migration patterns among brown bears in north‐western Eurasia. *Journal of Biogeography*, 40(5), 915-927.

Tang, Q., Low, G. W., Lim, J. Y., Gwee, C. Y., Rheindt, F. E. (2018). Human activities and landscape features interact to closely define the distribution and dispersal of an urban commensal. *Evolutionary Applications*, doi: 10.1111/eva.12650.