A spatial tool to support landscape planning to increase or maintain the production of environmental benefits for human well-being.

Landscape Services Restoration tool

Incorporating landscape patterns and ecosystem services in decisions for restoration

Gabriela Teixeira Duarte, Felipe Martello and Milton Cezar Ribeiro

Landscape Services Restoration tool (LSRestoration) - incorporating landscape patterns and ecosystem services in decisions for restoration

Main authors: Gabriela Teixeira Duarte, Felipe Martello and Milton Cezar Ribeiro

Contributing authors: Kai A. M. Chan, Matthew Mitchell, Edward J. Gregr, and Adriano P. Paglia.

Citation: Please cite the article "Duarte GT, Martello F, Mitchell M, Gregr EJ, Paglia AP, Chan KAM, Ribeiro MC. A framework and rapid assessment tool for multifunctional landscapes - incorporating landscape services in restoration planning. In prep."

1. Introduction

Ecological restoration of natural areas is necessary in landscapes that suffered from natural habitat loss and, consequently, have their capacity to provide ecological benefits to humans and high-quality habitats for biodiversity reduced (Barral et al., 2015). Often, these landscapes host people in private properties and evoke different economic interests, which create trade-offs with natural habitats conservation strategies. To encompass these landscapes and, at the same time, create more opportunities for funding restoration actions, the concept of ecosystem services (ES) can be fruitful (Goldman et al., 2008). ES are the benefits that people gain from ecosystems, including the production of goods (e.g. food, wood and honey), maintenance of processes (e.g. water purification, pollination of crops and disease control) and conditions (e.g. recreation in nature, scenic beauty and spiritual benefits) (MEA, 2005).

The provision of ES often depends on the composition, configuration and interaction of landscape elements (e.g. the different anthropic and natural areas; Duarte et al., 2018), and, in this case, they can be called landscape services (Termorshuizen & Opdam, 2009). Therefore, it is necessary to plan the location of natural areas to be restored that have the potential to increase or maintain the provision of these services in the landscape. In this sense, we developed the Landscape Services Restoration tool (LSRestoration), a landscape design tool with spatial explicit functions that generate different maps with areas of interest for restoration depending on the ES of concern. As the provision of different ES can be influenced by the same landscape pattern (composition and configuration), they can form bundles of services with similar important areas for restoration. Therefore, this tool follows the framework provided by Duarte et al. (in prep) that describes criteria to bundle different services based on the location of service providing units, the beneficiaries areas relationship with these units and type of the service flow.

To increase the availability and generality of the tool, we used two open source software for the programming and data processing: R (R Core Team, 2017) and Grass GIS (GRASS Development Team, 2015), respectively. Here we briefly describe the framework, but we encourage users to check the work of Duarte et al (in prep.) for better understanding.

2. Framework – bundling ecosystem services

The corresponding bundle of a specific ES will depend on the answer regarding 3 characteristics of the ES of concern. These characteristics and answers are better exemplified in the Duarte et al. (in prep) work:

1) What is the type of Service Providing Unit (SPU)? The following categories can be used to answer this question:
    a. Anthropogenic - the production of the service occurs in anthropogenic ecosystems.
    b. Natural - the production of the service occurs in natural or semi-natural ecosystems.
    c. Co-occurrence - for service production, it is necessary that both natural and anthropogenic ecosystems are interleaved.
    d. Landscape - the landscape is the providing unit of the service
2) What is the type of interaction between the Service Providing Unit and the Service Beneficiary Area (SPU-SBA)? The following categories can be used to answer this question (Fisher et al. 2009; Burkhard *et al.* 2014):
    a. In situ - SPU and SBA are realized in the same location.
    b. Directional - SBA in a specific location due to flow direction from the SPU.
    c. Omni-directional - SPU in one location, SBAs in the surrounding landscape without directional bias.
    d. Decoupled - ecosystem service can be traded over long distances.
3) How does the Delivery of the service occur? The following categories can be used to answer this question:
    a. Local - the flow of the ES occurs in one location.
    b. Infrastructure - some infrastructure (for example roads and processing plants) are necessary for the occurrence of ES flow.
    c. Organism movement - the flow of the ES occurs through organism movement.
    d. Water Bodies - the flow of the ES occurs through water bodies.
    e. Slope - the flow of the ES occurs downslope.
    f. Atmosphere - the flow of the ES occurs through the atmosphere.

You should understand that the answer to these questions is context dependent. We encourage the inclusion of stakeholders in the assessment of each one of them, to better capture all ES of concern. In addition, we did not aim to fulfill all possible combinations of answers for the three questions, instead, we focused on the ones we judge have scientific support and that are manageable from a landscape structure perspective.

We described in Table 1 the bundles described in Duarte et al. (in prep.) article, with examples of ES that can follow in each bundle. As state in the article, the tool does not contain a function for bundles 1 and 10, as the first bundle does not benefit from natural restoration, and the latter corresponds mostly to cultural ES influenced by landscape heterogeneity that we judge too context depended to model.

Table 1 – Proposed bundles of ecosystem services that are manageable at the landscape scale, with their respective categories regarding the Service Providing Unit (SPU) type, the spatial relationship between SPUs and Services Beneficiary Areas (SPU-SBA), and how the spatial delivery of the ecosystem service occurs. We provide examples of ecosystem services that could fit into each bundle.

Bundle Number	SPU	SPU-SBA	Delivery	Ecosystem Services Examples
1	Anthropogenic	Decoupled	Infrastructure	Cultivated crops; Reared animals and their outputs; Plant-based resources
2	Natural	Decoupled	Infrastructure	Wild plants, algae, animals and their outputs; Fibres and other materials from plants, algae and animals for direct use or processing; Materials from plants, algae and animals for agricultural use; Genetic materials from all biota
3	Natural	In Situ	Local	Bio-remediation or Filtration/sequestration/storage/ accumulation by micro-organisms, algae, plants, animals, and ecosystems; Maintaining nursery populations and habitats; Wild plants, algae, animals and their outputs; Fibres and other materials from plants, algae and animals for direct use, processing or agricultural use
4	Natural	Directional	Water bodies	Flood protection; Storm protection; Surface water for drinking; Surface water for non-drinking purposes
5	Natural	Omni-direction	Organism movement	Maintaining nursery populations and habitats; Genetic materials from all biota
6	Natural	Decoupled	Atmosphere	Global climate regulation by reduction of greenhouse gas concentrations
7	Co-occurrence	Omni-direction	Atmosphere	Micro and regional climate regulation
8	Co-occurrence	Omni-direction	Organism movement	Pollination; Seed dispersal; Pest control; Disease control
9	Co-occurrence	Directional	Slope	Mass stabilisation and control of erosion rates
10	Landscape	Omni-direction	Infrastructure	Experiential use of plants, animals and land-/seascapes in different environmental settings; Physical use of land-/seascapes in different environmental settings; Aesthetic

3.1. Getting started

LSRestoration tool runs as a script packaged in R environment. To run LSRestoration, you must have installed previously:

1. R software (version 3.X or above). The software should be installed following the procedures on the website: https://www.r-project.org
2. GRASS GIS (version 7.X or above). The software should be installed following the procedures on the website: http://grass.osgeo.org. For Mac OSx users, note that GRASS GIS should be installed in Application Folder of the user that is using the R software.
3. The R packages "rgrass7", "raster", "rgeos", "akima", "rgdal" and "maptools". Please, make sure that this packages are already installed in R (using the install.packages() command), before running LSRestoration.
4. If running bundle #5, LSCorridors GRASS package is required. The software should be installed following the procedures on the website: https://github.com/LEEClab/LS_CORRIDORS.

Both softwares above (R, GRASS GIS) are open source and free. Running LSRestoration does not require further R programming skills, but it does require basic skills in R like knowing how to import and export raster files in R environment and following scripts (for some examples see Martello, 2016). We provide in this User Guide the commands for installing the R packages and LSRestoration.

To use the tool for a specific context, you must compile data described in the bundles' section below that you wish to run and format them as indicated.

3.2. Formatting your data

Before running LSRestoration, it is necessary to format your data. Although subsequent sections of this guide describe how to prepare input data for each bundle, there are several formatting guidelines common to all models:

1. Spatial data should be in raster format, preference to tiff files (.tif).
2. All input data for a given bundle should be in the same Datum and projection. This projection should be in meters.
3. All input data for a given bundle should have the same spatial resolution (cell size) and extension. Depending on the resolution and extension of your raster data, the bundle function could take too long to run.
4. The raster files must have no missing data.

4.1. Basic inputs needed for all bundles

All the functions described below need three general information to work. They should be input in the R environment before running the function using the same object name as provided here:

1 – "dir.input": a character vector, corresponding to the absolute filepath representing the working directory of the R process, where the input raster files are, and where the output will be saved.

2 – "dir.grass": a character vector, corresponding to the absolute filepath representing the folder where Grass GIS is installed.

3 – "output.name": a character vector, corresponding to the name of the output raster map.

4 – "rest.map": the restoration map, in a raster file format, which should contain the delimitation (or boundaries) of the area available for restoration in the landscape or a surface of restoration feasibility/suitability. Although the restoration map can reflect the restoration feasibility/suitability in a cell on the map, its values should be relative. That is, a specific group of cells has a higher (or lower) restoration feasibility/suitability/cost compared to another group. For example, if a region in the landscape has 3 times less suitability (or 3 times higher cost) for restoration than the rest of the landscape, due to soil properties, the group of cells that correspond to that region should have a value of 1 and the rest of the regions available for restoration in the landscape a value of 3. In this example, the models will prioritize areas more likely to be restored (value 3), as they have higher

suitability for restoration (or lower costs). If a region in the landscape has, for any reason, no restoration feasibility/suitability, the cell values in the rater map should correspond to zero. For example, if there is an urban region, a road, or a river in the landscape, the values in that cells should be zero for the rest.map input. The bundle functions below, will not consider in the prioritization regions with zero values in the rest.map, and the output maps will show these regions with Not Available (NA) values.

Therefore, the R commands to begin working with the bundle functions are:

```
install.packages(c("maptools", "rgdal", "rgeos","akima","rgrass7","raster"))
```

```
install.packages("~\\lsrestoration_1.0.2.tar.gz", repos = NULL, type="source") # This is just an example of filepath. Please give the correct filepath to the latest LSRestoration .tar.gz file in work computer.
```

```
library(lsrestoration)
```

```
dir.input<-"C:\\Users\\User_Name\\LSRestoration\\BundleX"  #This is just an example of filepath. You should change to your own inputs filepath. An example for Mac OSx users, is "~/Applications/GRASS-7.4.0.app/Contents/Resources".
```

```
setwd(dir.input)
```

```
dir.grass<-"C:\\Program Files\\GRASS GIS 7.4.0" # This is just an example of filepath. Make sure that you change it for the correct filepath where GRASS GIS is installed.
```

```
output.name<-"rest.benf.bundleX " #This is just an example of file name. You should change to your own preference.
```

```
rest.map<- raster("rest_surface.tif") #use here the correct file name. This raster file should be saved in the same input folder described in the "dir.input".
```

Below we described other inputs and necessary information for each one of the bundles functions.

4.2. Bundle 2 (SPU = natural, SPU-SBA = decoupled, Delivery = infrastructure)

Bundle purpose and parameters

This function creates a restoration surface that prioritizes areas near the infrastructure needed for the service flow and near the already established SPUs in the landscape. We assumed that the areas restored within this bundle should be near infrastructure to guarantee ES delivery, and near already established SPU so that species of interest could more easily recolonize them. You have the option to input a maximum distance from the infrastructure and from the already established SPUs, in a way that areas further than this maximum distance will have no value for restoration, that is, a value equal to zero. The flowchart represented in figure 1 contains the inputs, process and outputs of this bundle function. Therefore, for this bundle, you need to input:

1. "infra.map": a raster map with the location of the infrastructures needed for the service delivery. It should be a binary map with cells corresponding to the infrastructure sites with values equal 1 and the other cells with values equal to zero.
2. "spu.map": a binary raster map with the already established SPUs in the landscape. Cells in the map corresponding to SPU areas should have a value equal to 1 and everything else with a value equal to zero.
3. "dist.infra" (optional): an integer number corresponding to the maximum distance (in meters) from the infrastructure where the restoration areas should be located.
4. "dist.spu" (optional): an integer number corresponding to the maximum distance (in meters) from the already established SPUs where the restoration areas should be located.

Bundle operation and outputs

The function will create a raster map with the Euclidian distances from the already established SPUs and another Euclidian distance map from the infrastructure sites. Both resultant distances maps will integrate a Generalized Linear Model (GLM) analysis that will predict the restoration priority for each one of the landscape cells. Therefore, these priority values are response variables in function of distance from SPU and infrastructure (with interaction between predictors). As both maximum distances are optional inputs, if you choose not to input one or both variables, the function will use the highest value in the corresponding distance map. The final step is to multiply the resultant map from the GLM analysis with the restoration map ("rest.map"). These final values are standardized from zero to one. The function gives two outputs:

1. "rest.priority2": a raster map with the restoration priorities for the ecosystem services of interest.
2. a plotted graph in R environment, that corresponds to the response variables of the GLM model (restoration priority) plotted with two maximum distances used in the model. We give an example of a graph in figure 2.

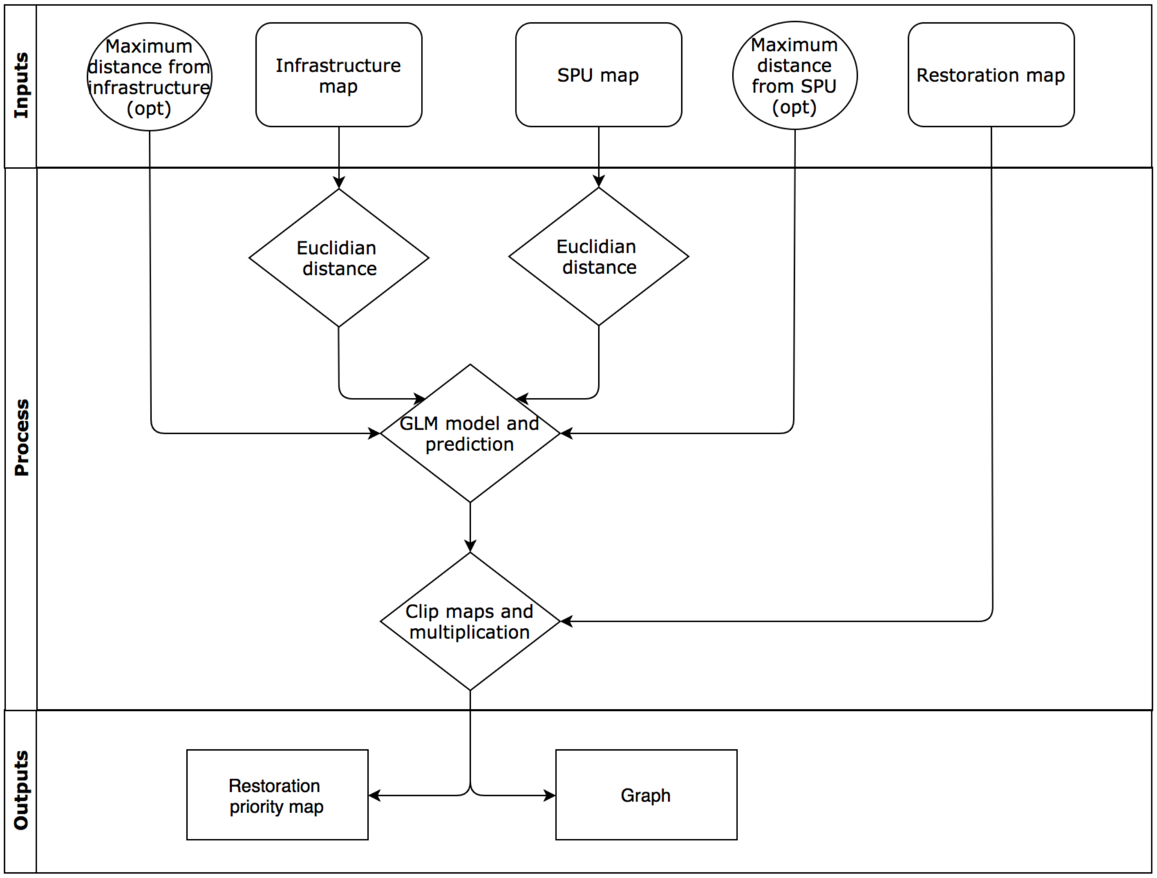

Figure 1 - Flowchart representing the inputs, process and outputs for the function from bundle number 2

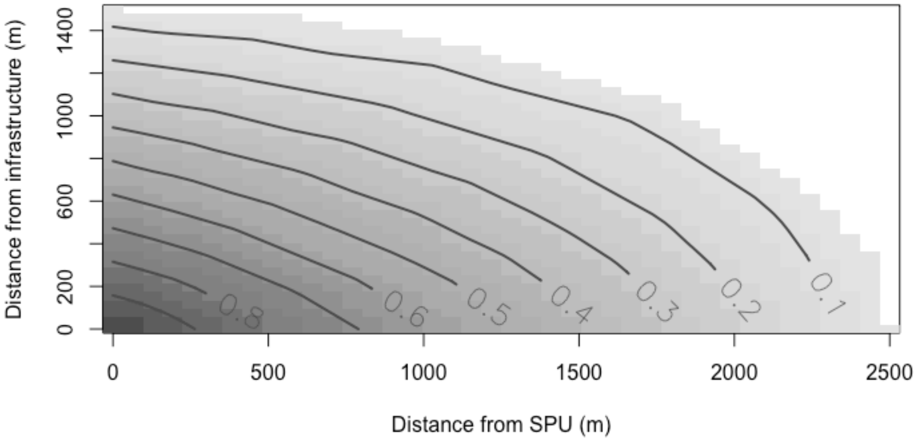

Figure 2 - Graphic representing the restoration priority values in relation to the distances from the Service Providing Units (SPU) and from the Infrastructure. Darker gray colors in the cells represent higher restoration priority values. The gray lines are isolines of the corresponded restoration priority value.

Bundle example

For running this bundle function, you need to type the following R commands:

```
##Inputs for bundle 2. The raster names and distance values are examples. Use the correct file name when inputting a raster file and change the distance value according to your context.

spu.map<-raster("spu.tif")

infra.map<-raster("roads.tif")

dist.spu<-850

dist.infra<-1000

#Running the bundle 2 function

bundle2.results<- bundle2(dir.grass, rest.map, spu.map, infra.map, dist.spu, dist.infra, output.name)
```

4.3. Bundle 3 (SPU = natural, SPU-SBA = in situ, Delivery = local)

Bundle purpose and parameters

The aim of the function is to give restoration priority to areas around the specific sites that need the ecosystem service. As this is a local service with in-situ delivery, we assume that the main landscape pattern contributing to the maintenance and improvement of the production of the service will be the quantity of natural areas surrounding specific sites. Figure 3 shows a flowchart representing the inputs, process and output for this bundle. To create a restoration priority map, you need to input:

1. "site.map": a raster map with the location of the sites that need the service. It should be a binary map with cells corresponding to the specific sites with values equal 1 and the other cells with values equal to zero.
2. "dist.site": an integer value related to the maximum distance (in meters) from the site. This distance corresponds to the radius of the area of influence of the ecosystem service. For example, if it is necessary a bioremediation on specific sites, the distance would be the maximum length that could influence the bioremediation through restoration. Another example is for food production that has local consumption (as natural fruits, nuts and meat for subsistence). In this case, the distance would be the maximum radius necessary to create a habitat for the species of interested.

Note that the function will prioritize areas closer to the input sites. For this, we created a distance-decay function follows a sigmoid curve (figure 4), that can have different shapes according to your preference. To the sigmoid curve adapt to your study area context, it is necessary to input the following parameter:

3. "alpha": an integer value, between 1 and 5, that corresponds to the alpha parameter for the sigmoid curve. This parameter will change the inclination and shape of the curve (see figure 4 for alpha examples). The higher the alpha, the sigmoid curve will decay more similar to a linear function. With lower alpha values, the distribution of priority values will follow a sigmoid curve, and you will have a

group of cells in the restoration map closer to the inputted sites with priority values with high values, and a group of cells with low values. We recommend a sigmoid distance-decay curve with alpha near 1, especially in cases when the user wants to prioritize restoration in the site proximity and also think in a buffer area around this priority region that will be restored after, or partially.

Bundle operation and outputs

First, the function will create a raster map with the Euclidian distances from the sites. Then, it will use these values of maximum distance and the alpha value, to construct a sigmoid function that gives new values to the cells. The final step is to multiply the resultant map from the sigmoid function with the restoration map ("rest.map"). These final values are standardized from zero to one. This function has two outputs:

1. "rest.priority3": a raster map with the restoration priorities for the ecosystem services of interest.
2. two plotted graphs in R environment, that correspond to the relationship between the distance from the sites and the restoration priority. One graph corresponds to the maximum distance from sites found in the whole landscape. The other one corresponds to the maximum distance from sites selected by the user. With these graphs, the user can do simulations and compare different maximum distances, and different alpha values. We give an example of graphs in figure 5.

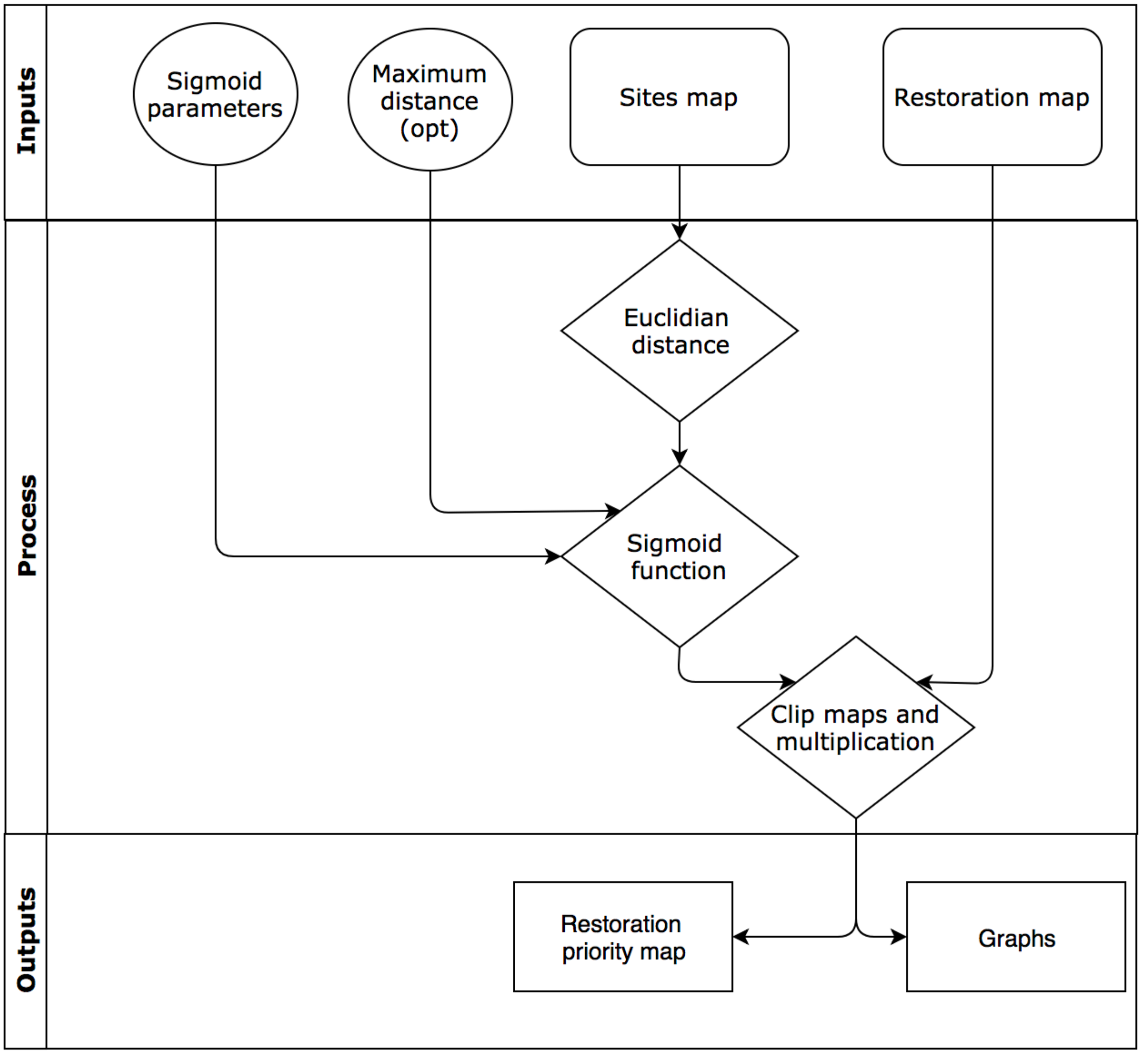

Figure 3 - Flowchart representing the inputs, process and outputs for the function from bundle number 3

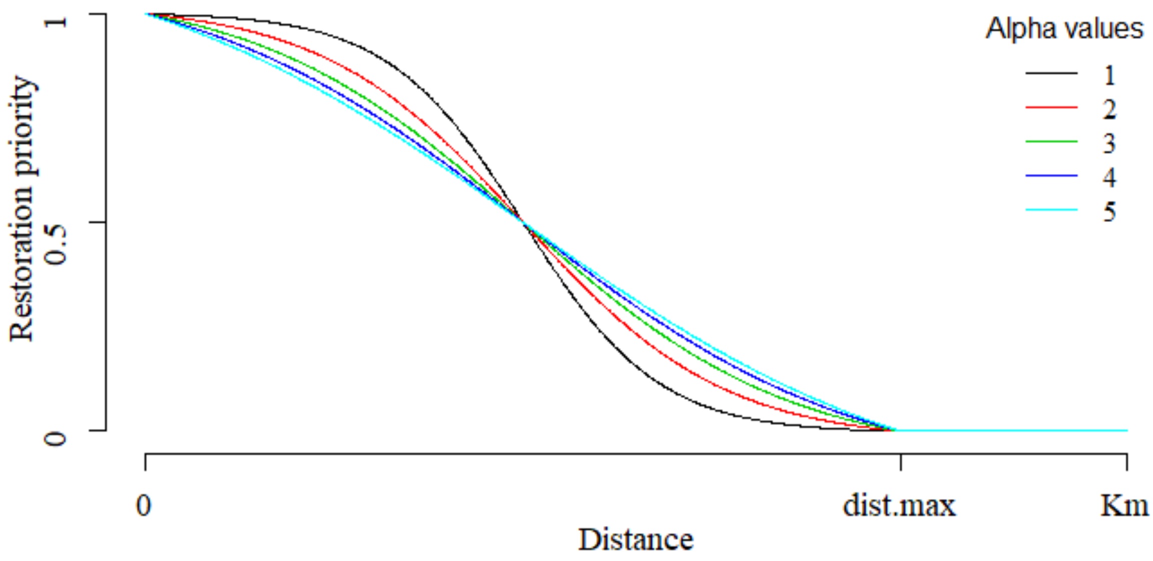

Figure 4 - Graph with exemplification of sigmoid curves for different values of the alpha parameter, with distance as the explanatory variable and the restoration priority as the response variable. Different values of the alpha parameter will change the shape of the curve. The "dist.max" is the maximum distance set by the user (named "dist.site" in bundle 3, and "dist.water" in bundle 4).

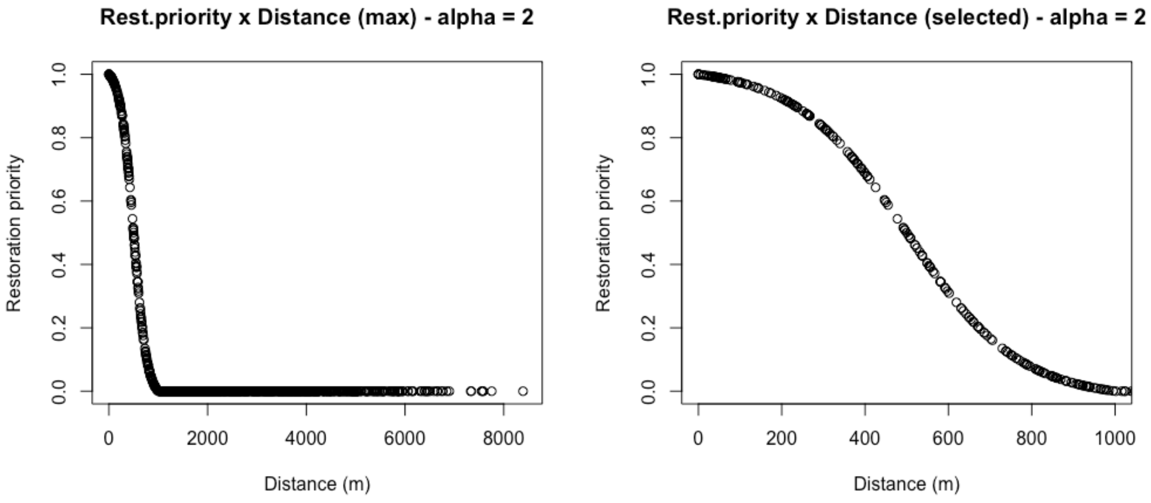

Figure 5 - Graphs with exemplification of the relationship between restoration priority values and the maximum distance from sites (or water bodies in the case of bundle 4) in the landscape (left) and maximum distance from the site (or water bodies in the case of bundle 4) selected by the user (right).

Bundle example

For running this bundle function, you need to type the following R commands:

```
##Inputs for bundle 3. The raster names, alpha and distance values are examples. Use the correct file name when inputting a raster file and change the distance and alpha values according to your context.

site.map<-raster("mines.tif")

dist.site<-500

alpha<-0.5

#Running the bundle 3 function

bundle3.results<- bundle3(dir.grass, rest.map, site.map, dist.site, alpha, output.name)
```

4.4. Bundle 4 (SPU = natural, SPU-SBA = directional, Delivery = water bodies)

Bundle purpose and parameters

The aim of the function is to give restoration priority to areas buffering water bodies. We assume that areas surrounding water springs, rivers and lakes, will serve as the main source of water delivered services. The flowchart for this function is shown in figure 6. The function for this bundle works very similar to the function for bundle 3. The main difference here is that you need to input a raster map with the water bodies of interest instead of the site's map. Therefore, the inputs for this function are:

1. “water.map”: a raster map with the location of the water bodies that need or deliver the service. This is a binary map with cells corresponding to the water bodies with values equal 1 and the other cells values equal zero.
2. “dist.water”: an integer value related to the maximum distance (in meters) from the water body. This distance corresponds to the maximum width of the buffer area that can influence the process occurring in water bodies. This is very context depend,

and to define this distance you should consider the types of water bodies, their own size and width, soil types, etc.

Like the previous bundle, this function will give priority for areas closer to water bodies and the distance-decay function follows a sigmoid curve (examples in figure 4). Therefore, it also needs:

3. "alpha": an integer value, between 1 and 5, that corresponds to the alpha parameter for the sigmoid curve. This parameter will change the inclination and shape of the curve (see figure 4 for alpha examples). The higher the alpha, the sigmoid curve will decay more similar to a linear function. The same principles for the alpha parameter in bundle 3 apply here.

Bundle operation and outputs

First, the function will create a raster map with the Euclidian distances from the water bodies. Then, it will use these values of distance and the alpha value, to construct a sigmoid function that gives new values to the cells. The final step is to multiply the resultant map from the sigmoid function with the restoration map ("rest.map"). These final values are standardized from zero to one. This function has two outputs:

1. "rest.priority4": a raster map with the restoration priorities for the ecosystem services of interest.
2. two plotted graphs in R environment, that correspond to the relationship between the distance from the water bodies and the restoration priority. One graph corresponds to the maximum distance from water bodies found in the whole landscape. The other one corresponds to the maximum distance from water bodies selected by the user. With these graphs, the user can do simulations and compare different maximum distances, and different alpha values. We give an example of graphs in figure 5.

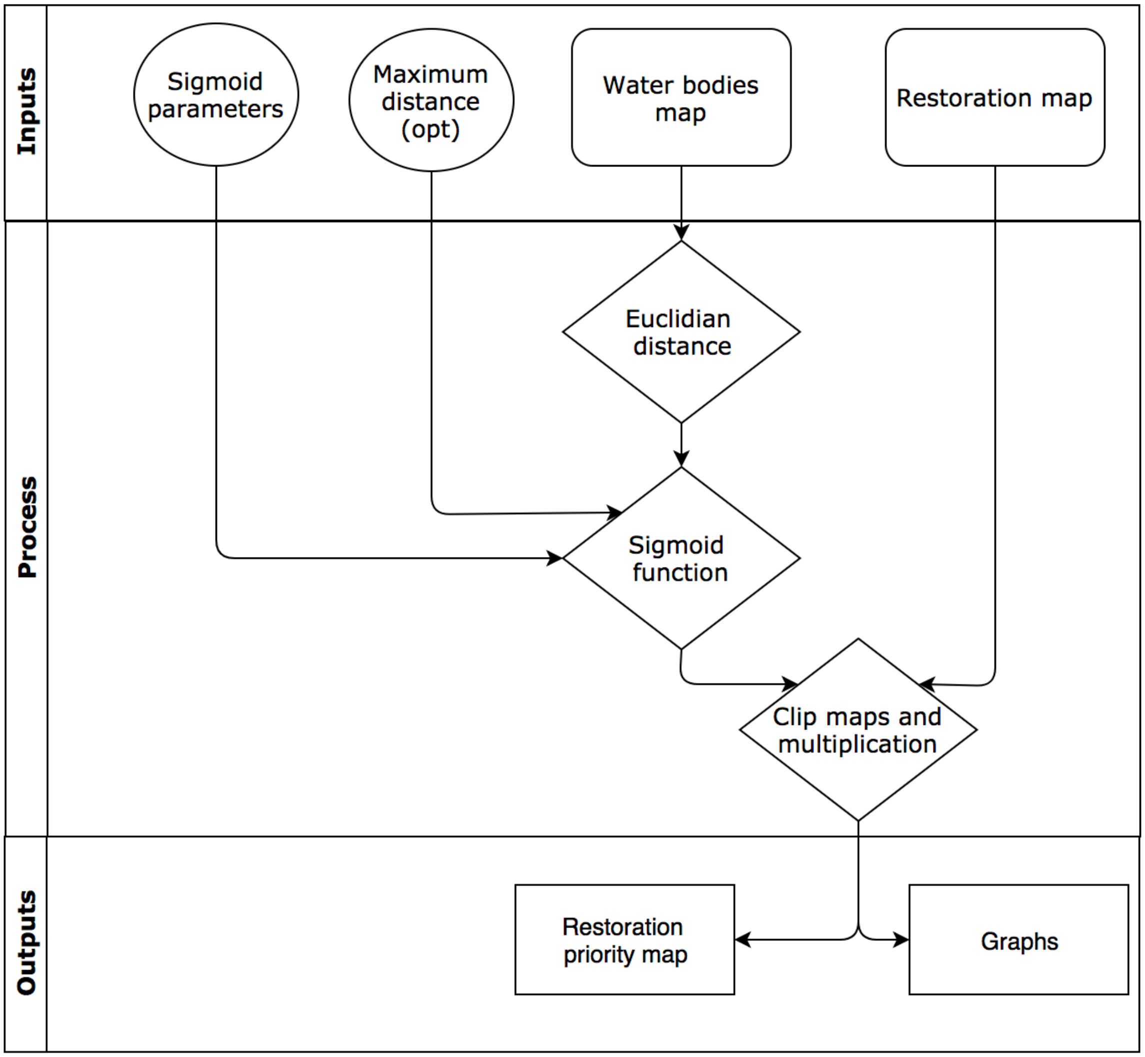

Figure 6 - Flowchart representing the inputs, process and outputs for the function from bundle number 4

Bundle example

For running this bundle function, you need to type the following R commands:

##Inputs for bundle 4. The raster names, alpha and distance values are examples. Use the correct file name when inputting a raster file and change the distance and alpha values according to your context.

water.map<-raster("water.tif")

dist.water<-350

alpha<-1

#Running the bundle 4 function

bundle4.results<- bundle4(dir.grass, rest.map, water.map, dist.water, alpha, output.name)

4.5.Bundle 5 (SPU = natural, SPU-SBA = omnidirectional, Delivery = organism movement)

Bundle purpose and parameters

The aim of this function is to give restoration priority to regions that can connect the already established SPUs, as movement of organisms between natural areas need to be increased or maintained for the service flow. For this function, we recommend that you install and use the recently released LSCorridors software (Ribeiro et al., 2017). For LSCorridors to work, you need to input. It is also possible that you use other corridors simulation software. Here, we will describe the LSCorridors output and how your bundle function uses it, but few free to use any software you want.

Bundle operation and outputs

After running LSCorridors, the software will give, as one of its outputs, a raster map showing how many of the corridor simulations passed through each pixel of the map. The authors call these values the Route Selection Frequency Index (RSFI), and state that "high RSFI values indicate areas (pixel) that are more likely to be used as corridors according to species requirements included in the resistance surface and should, therefore, receive special attention of the decisions makers". For comparisons purposes with the other bundles, the bundle 5 function takes the RSFI map resultant from a simulation in the LSCorridors software, multiplies it with the restoration map and rescales the product from zero to 1. Therefore, higher the value in the RSFI map, the higher is the restoration priority of that pixel in the final map. Figure 7 has the flowchart for this bundle function.

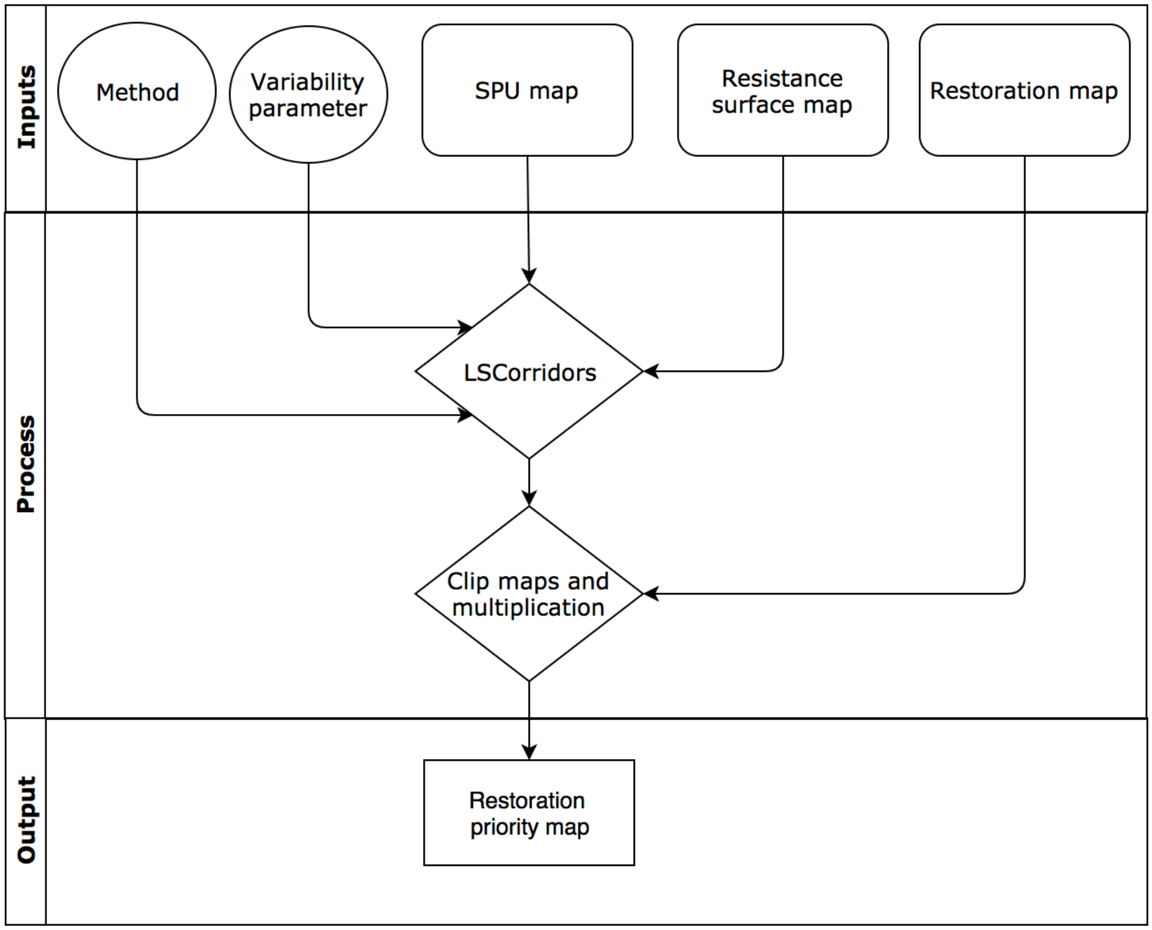

Figure 7 - Flowchart representing the inputs, process and outputs for the function from bundle number 5

Bundle example

For running this bundle function, you need to type the following R commands:

##Inputs for bundle 5. The raster name is an example. Use the correct file name when inputting a raster file.

corridors.map<-raster("RSFI_lscorridors.tif")

#Running the bundle 5 function

bundle5.results<- bundle5(rest.map, corridors.map, output.name)

4.6. Bundle 6 (SPU = natural, SPU-SBA = decoupled, Delivery = atmosphere)

Bundle purpose and parameters

The aim of this function is to give restoration priority to areas around already established SPUs, considering the shape of the fragments to reduce edge effects, the matrix extension and the contrast between the SPU and the adjacent matrix. We assume that fragments with fewer edge effects, lower adjacent matrix extension and contrast will retain more carbon in the above and belowground biomass, contributing to mitigate the greenhouse effects. For this function you should input:

1. "spu.map": a binary raster map with the already established SPUs in the landscape. Cells in the map corresponding to SPU areas should have a value equal to 1 and everything else with a value equal to zero.
2. "dist.edge": an integer number corresponding to the distance (in meters) that the expected edge effects could percolate inside the SPUs and decrease carbon stock and sequestration.

Bundle operation and outputs

The function will create a raster map with the Euclidian distances from the SPUs in the "spu.map". After it, the function also takes spu.map for a move-window analysis, where the central pixel will have the sum value of its neighbors. The "dist.edge" value corresponds to half of one of the move-window sides. In the last step, the function multiplies the new map created after the move-window analysis, the Euclidian distance map and the restoration map. These final values are standardized from zero to one (Figure 8).

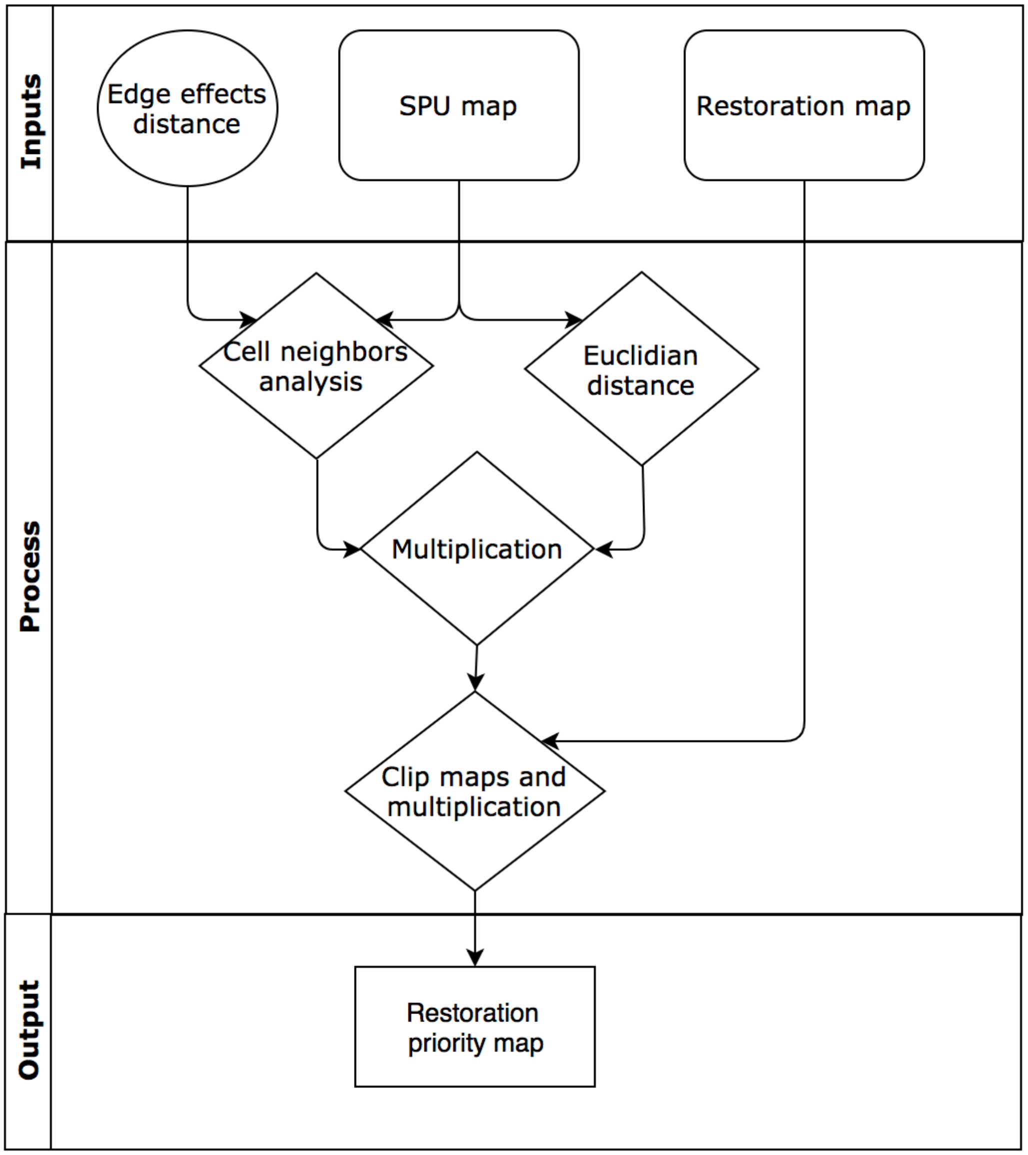

Figure 8 - Flowchart representing the inputs, process and outputs for the function from bundle number 6

Bundle example

For running this bundle function, you need to type the following R commands:

##Inputs for bundle 6. The raster name and distance values are examples. Use the correct file name when inputting a raster file and change the distance values according to your context.

spu.map<-raster("spu.tif")

dist.edge<-150

#Running the bundle 6 function

bundle6.results<- bundle6(dir.grass, rest.map, spu.map, dist.edge, output.name)

4.7. Bundle 7 (SPU = co-occurrence, SPU-SBA = in-situ, Delivery = atmosphere)

Bundle purpose and parameters

The aim of the function is to give restoration priority to areas near and within SBA, spaced apart, and sparse from the already established SPUs. We assume that the benefit of restoration is smaller near the already established SPUs, as these areas are already under the effects of the ESs that the SPUs provide. The function of this bundle creates a solution with new restoration patches that are interleaved within and near the SBA areas. Each one of these patches will benefit the surrounding area. The distance between the restoration areas should be thought to minimize the overlap between benefited areas of different patches (Brosi et al., 2008). Therefore, the inputs necessary for this function are:

1. "spu.map": a binary raster map with the already established SPUs in the landscape. Cells in the map corresponding to SPU areas should have a value equal to 1 and everything else with a value equal to zero.
2. "sba.map": a raster map with SBA location in the landscape. Cells in the map corresponding to SBA areas should have a value different than zero and everything else with a value equal to zero. The values of SBA cells should be relative and represent the demand for an ecosystem service. For example, if a region in the landscape needs twice more of that service than another region, the cells in the first region should have values equal to 2 and the other region values equal to 1.
3. "patch.max": the maximum size of a patch that will be restored, in square meters. This size is context dependent and should be defined as the sufficient area to provide the benefit for that distance of influence. Therefore, larger areas than the one inputted in this parameter will not have higher benefits.
4. "patch.min": the minimum size of a patch that will be restored, in square meters. This size is context dependent and should be defined as the minimum area to be

restored. That is the minimum area that can provide the service. Therefore, areas smaller than the one inputted in this parameter does not provide sufficient benefits.

5. "dist.inf": the distance of influence, in meters, that the benefit of the service can percolate in the landscape from the already established SPU and/or from new restored patches. The area surrounding the SPU and patches within this distance will be considered as the area benefiting from the services of this bundle. Therefore, the distance between two adjacent new patches corresponds to twice the dist.inf parameter.

Bundle operation and outputs

Figure 9 contains the flowchart with the inputs, process and outputs of this bundle function. In the first step, the function uses the "dist.infl" parameter to create a buffer around the "spu.map" and excludes the SBA areas that are already benefiting from the established SPUs. Then, the function creates a hexagonal grid, with the distance between the centers of this hexagons equals the sum of the radius of the "patch.max" area and the "dist.infl" parameter. After, the function creates two regions for each hexagon: a) the core region, where the restored patches are proposed to fall in – thus including the "rest.map" cell values to formulate this core region; b) the benefited region, that represents the region benefited from the corresponded core region – thus including the "dist.infl" parameter and the "sba.map" cell values to formulate this benefited region. The function excludes the core regions that have areas smaller than the "patch.min" parameter, and that the corresponded benefited region do not overlap with the "sba.map". Then, it multiplies the sum of values in the core region with the sum of values in the respective benefited region. The results are standardized from zero to one. This creates a restoration priority index, and it is attributed to all cells in the same core region. Therefore, this function has two outputs:

1. "rest.benf7": raster map with the remaining core regions, with its values corresponding to the restoration priority index.

2. "hex.grid": a hexagonal grid corresponding to the locations of the restoration patches, with a shapefile format. It shows in the attribute table the total area of each core region, the total area of its respective benefited region, and the restoration priority index.

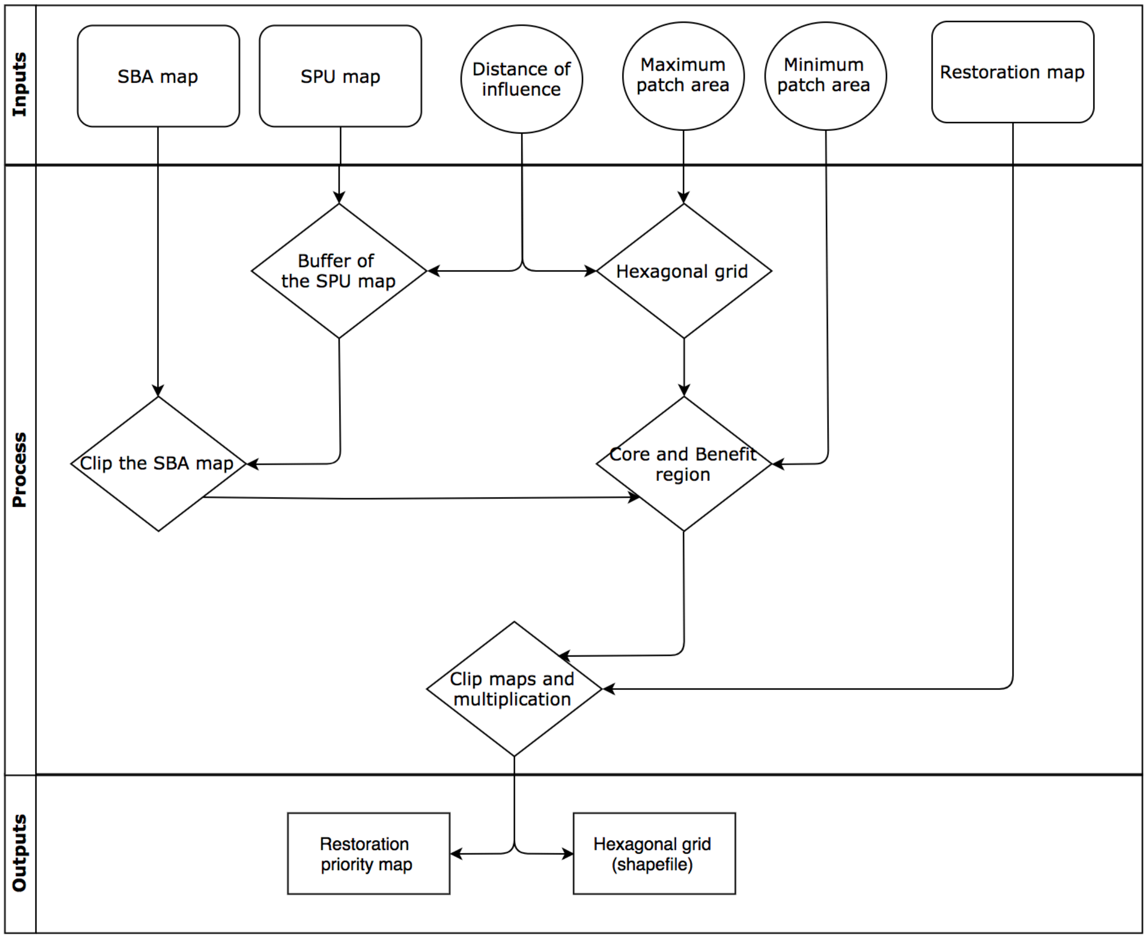

Figure 9 - Flowchart representing the inputs, process and outputs for the function from bundle number 7

Bundle example

For running this bundle function, you need to type the following R commands:

##Inputs for bundle 7. The raster names, patch area and distance values are examples. Use the correct file name when inputting a raster file and change the distance and area values according to your context.

spu.map<-raster("spu.tif")

sba.map<-raster("sba.tif")

patch.max<- 2000

patch.min<- 1500

dist.inf<- 350

#Running the bundle 7 function

bundle7.results<- bundle7(dir.grass, rest.map, spu.map, sba.map, patch.max, patch.min, dist.inf, output.name)

4.8.Bundle 8 (SPU = co-occurrence, SPU-SBA = in-situ, Delivery = organism movement)

Bundle purpose and parameters

The bundle 8 function is similar to the bundle 7. They both aim to give restoration priority to areas near and within SBA, spaced apart, and sparse from the already established SPUs. However, in this bundle, we assume that the benefit of restoration is higher in areas near the already established SPUs, as these SPU areas will serve as the source of the organism that can potentially colonize the new restored areas. What the tool does for this bundle is to create a solution with new restoration patches that are interleaved within and near the SBA areas. Each one of these patches will benefit the area surrounding it. The distance between the restoration areas should be thought to minimize the overlap between benefited areas of different patches (Brosi et al., 2008). Therefore, the inputs necessary for this function are:

1. "spu.map": a binary raster map with the already established SPUs in the landscape. Cells in the map corresponding to SPU areas should have a value equal to 1 and everything else with a value equal to zero.
2. "sba.map": a raster map with SBA location in the landscape. Cells in the map corresponding to SBA areas should have a value different than zero and everything else with a value equal to zero. The values of SBA cells should be relative and represent the demand for an ecosystem service. For example, if a region in the landscape needs twice more of that service than another region, the cells in the first region should have values equal to 2 and the other region values equal to 1.
3. "patch.max": the maximum size of a patch that will be restored, in square meters. This size is context dependent and should be defined as the sufficient area to

provide the benefit for that distance of influence. Therefore, larger areas than the one inputted in this parameter will not have higher benefits.

4. "patch.min": the minimum size of a patch that will be restored, in square meters. This size is context dependent and should be defined as the minimum area to be restored. That is the minimum area that can provide the service. Therefore, areas smaller than the one inputted in this parameter does not provide sufficient benefits.
5. "dist.inf": the distance of influence, in meters, that the benefit of the service can percolate in the landscape from the already established SPU and/or from new restored patches. The area surrounding the SPU and patches within this distance will be considered as the area benefiting from the services of this bundle. Therefore, the distance between two adjacent new patches corresponds to twice the dist.inf parameter.

Bundle operation and outputs

Figure 10 contains the flowchart with the inputs, process and outputs of this bundle function. In the first step, the function uses the "dist.infl" parameter to create a buffer around the "spu.map" and excludes the SBA areas that are already benefiting from the established SPUs. Then, the function creates a hexagonal grid, with the distance between the centers of this hexagons equals the sum of the radius of the "patch.max" area and the "dist.infl" parameter. After, the function creates two regions for each hexagon: a) the core region, where the restored patches are proposed to fall in – thus including the "rest.map" cell values to formulate this core region; b) the benefited region, that represents the region benefited from the corresponded core region – thus including the "dist.infl" parameter and the "sba.map" cell values to formulate this benefited region. The function excludes the core regions that have areas smaller than the "patch.min" parameter, and that the corresponded benefited region do not overlap with the "sba.map". Then, it multiplies the sum of values in the core region with the sum of values in the respective benefited region. The results are standardized from zero to one. This creates a restoration priority index, and it is attributed to all cells in the same core region. A different step from bundle 7, is that this function multiplies the remaining core regions cells with Euclidian distance values from the already established SPUs. The results are standardized from zero to one.

Therefore, this function has two outputs:

1. "rest.benf7": raster map with the remaining core regions, with its values corresponding to the product between the restoration priority index and the Euclidian distance from the already established SPUs.

2. "hex.grid": a hexagonal grid corresponding to the locations of the restoration patches, with a shapefile format. It shows in the attribute table the total area of each core region, the total area of its respective benefited region, and the restoration priority index.

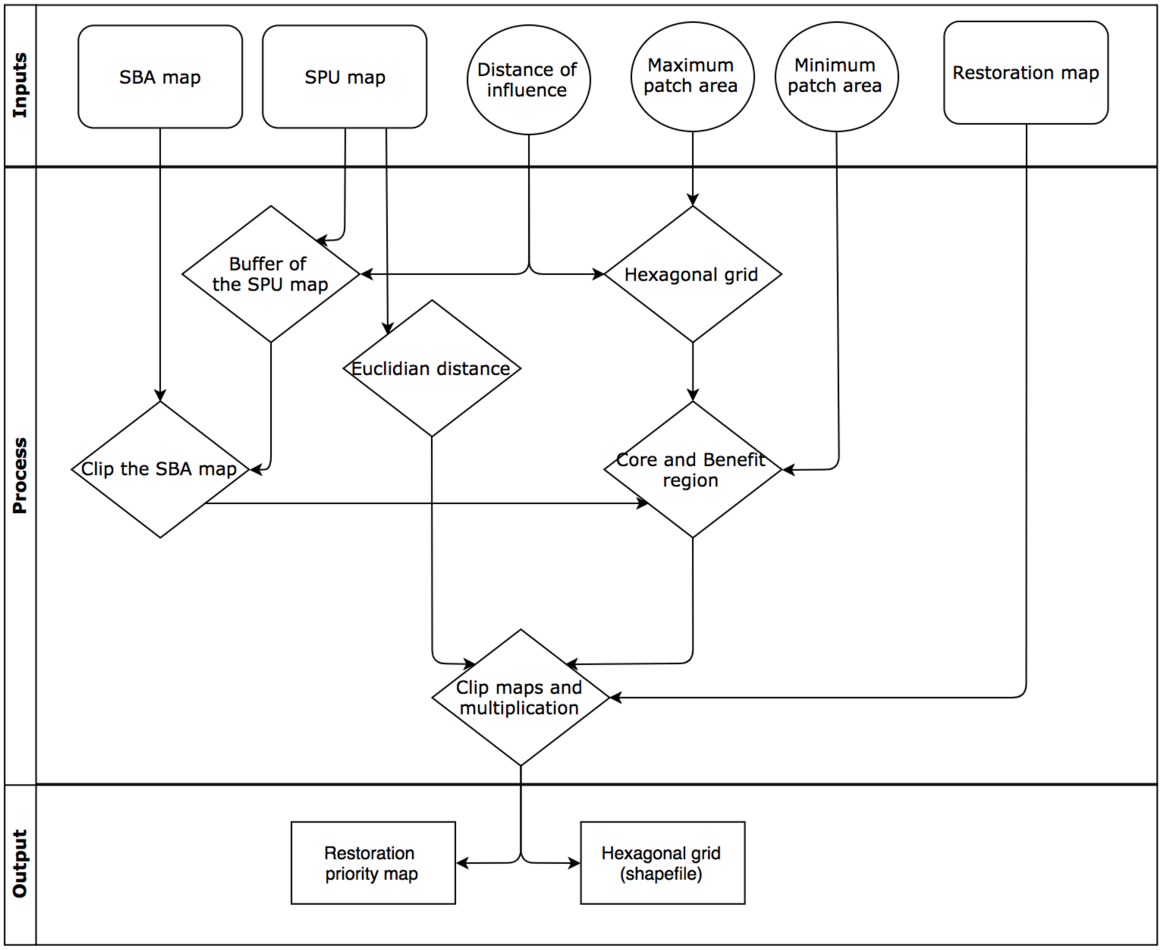

Figure 10 - Flowchart representing the inputs, process and outputs for the function from bundle number 8

Bundle example

For running this bundle function, you need to type the following R commands:

```
##Inputs for bundle 8. The raster names, patch area and distance values are examples. Use the correct file name when inputting a raster file and change the distance and area values according to your context.

spu.map<-raster("spu.tif")

sba.map<-raster("sba.tif")

patch.max<- 120

patch.min<- 90

dist.inf<- 300

#Running the bundle 8 function

bundle8.results<- bundle8(dir.grass, rest.map, spu.map, sba.map, patch.max, patch.min, dist.inf, output.name)
```

4.9.Bundle 9 (SPU = co-occurrence, SPU-SBA = directional, Delivery = slope)

Bundle purpose and parameters

The aim of this function is to give restoration priority to areas with steeper and longer slopes. We assume that the benefit from restoration will increase with the increase of erosion potential, which is related to the slope length-gradient factor in the Universal Soil Loss Equation (Wischmeier & Smith, 1978). The function uses an already established tool in the GRASS GIS software to calculate the slope length-gradient factor (LS factor) for each cell in the landscape. With everything else equal, the higher the LS factor, the higher the long-term average annual soil loss and, therefore, the increased need to restore natural areas on that site to maintain the retention of soil, nutrients and other services. Figure 11 contains the flowchart with the inputs, process and output of this bundle function. For this tool to work, it is necessary to input:

1. "dem.map": a raster file with the digital elevation model for the landscape of interest.

2. "threshold": a threshold value, which specifies the minimum size of an exterior watershed basin in cells number. As defined in the GRASS GIS manual, "the minimum size of drainage basins, defined by the threshold parameter, is only relevant for those watersheds with a single stream having at least the threshold of cells flowing into it. These watersheds are called exterior basins" (https://grass.osgeo.org/grass75/manuals/r.watershed.html).

Bundle operation and outputs

We suggest that the restoration map includes, if available, weights for the different soil erodibility rates (which indicates the susceptibility of soil particles to be detached and carried by the rain). The higher the erodibility rate, the higher the benefit to restore in that cell. The others USLE factors (Rainfall erosivity index, Support practice factor, Cover-management factor) can also be included in the restoration map. Keep in mind that the USLE equation corresponds to the multiplication of these factors, and their product can integrate the restoration map. Last, we advise that having a well-prepared digital elevation model is critical and it must have no missing data.

The function will first calculate the slope length gradient factor and, then, multiply the resultant map with the restoration map ("rest.map").

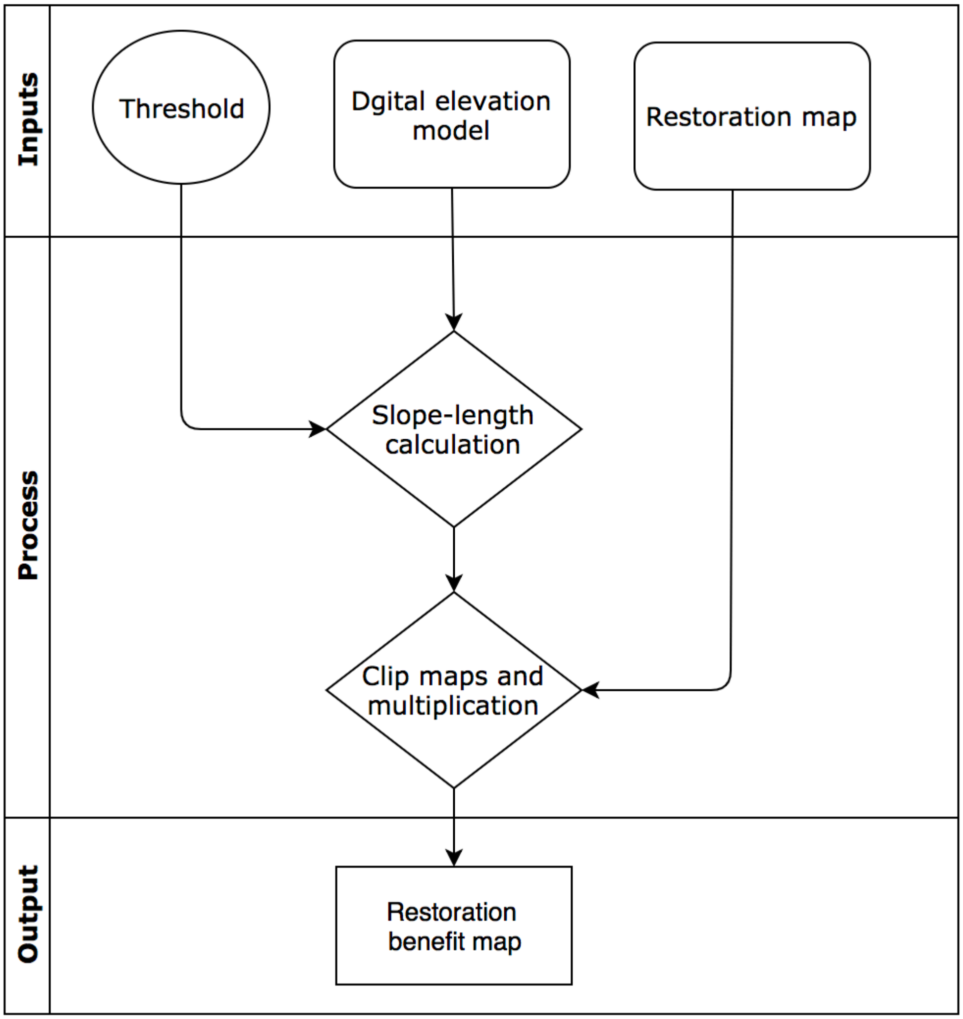

Figure 11 - Flowchart representing the inputs, process and outputs for the function from bundle number 9

Bundle example

For running this bundle function, you need to type the following R commands:

##Inputs for bundle 9. The raster names and the threshold value are examples. Use the correct file name when inputting a raster file and change the threshold value according to your context.

dem.map<-raster("dem.tif")

threshold<- 100

#Running the bundle 9 function

bundle9.results<- bundle9(dir.grass, rest.map, dem.map, threshold, output.name)

5. Reporting errors

If you encounter any issues, please post to the user's support forum at https://github.com/LEEClab/LSRestoration with the following information:

1. LSRestoration bundle you're having difficulty with
2. Explicit error message or behavior
3. If possible, a screenshot of the state of your R environment when you get the error.

6. Acknowledgments

We would like to thank all LEC - UFMG, LEEC – UNESP and CHANS Lab - UBC members for their various forms of contribution, especially Rafaela Silva and Julia Assis for their support during this work. We thank Coordination for the Improvement of Higher Education Personnel (CAPES) and National Council for Scientific and Technological Development (CNPq) for the scholarships of G.T. Duarte. M.C. Ribeiro is funded by CNPq (grant 312045/2013-1 and 312292/2016-3), PROCAD/CAPES (project # 88881.068425/2014-01) and The São Paulo Research Foundation (FAPESP - grant 2013/50421-2).

7. References

Barral, M. P., Benayas, J. M. R., Meli, P., & Maceira, N. O. (2015). Quantifying the impacts of ecological restoration on biodiversity and ecosystem services in agroecosystems: a global meta-analysis. *Agriculture, Ecosystems and Environment*, *202*, 223–231. doi:10.1016/j.agee.2015.01.009

Brosi, B. J., Armsworth, P. R., & Daily, G. C. (2008). Optimal design of agricultural landscapes for pollination services. *Conservation Letters*, *1*(1), 27–36. doi:10.1111/j.1755-263X.2008.00004.x

Duarte, G. T., Santos, P. M., Cornelissen, T. G., Ribeiro, M. C., Paglia, A. P. (2018). The effects of landscape patterns on ecosystem services: meta-analyses of landscape services. *Landscape Ecology*. https://doi.org/10.1007/s10980-018-0673-5

Goldman, R. L., Tallis, H., Kareiva, P., & Daily, G. C. (2008). Field evidence that ecosystem service projects support biodiversity and diversify options. *Proceedings of* *the National Academy of Sciences of the United States of America*, *105*(27), 9445–8. doi:10.1073/pnas.0800208105

GRASS Development Team (2017). Geographic Resources Analysis Support System (GRASS) Software, Version 7.2. Open Source Geospatial Foundation. https://grass.osgeo.org

Martello, F (2016). Landscape Ecology in R. Managing spatial data, calculating landscape metrics and simulating corridors. Available online: http://www.leec.eco.br/pdfs/Support_Landscape_ecology_R_2016_06_15.pdf.

R Core Team (2017). R: A Language and Environment for Statistical Computing. https://www.R-project.org/

Ribeiro, J. W., Silveira dos Santos, J., Dodonov, P., Martello, F., Niebuhr, B. B., & Ribeiro, M. C. (2017). LandScape Corridors (lscorridors): a new software package for modelling ecological corridors based on landscape patterns and species requirements. *Methods in Ecology and Evolution*. doi:10.1111/2041-210X.12750

Termorshuizen, J. W., & Opdam, P. (2009). Landscape services as a bridge between landscape ecology and sustainable development. *Landscape Ecology*, *24*(8), 1037–1052. doi:10.1007/s10980-008-9314-8

Wischmeier, W. H., & Smith, D. D. (1978). *Predicting rainfall erosion losses - A guide to conservation planning*. Washington, DC: USDA.