

## **Spatial Analysis in R: An introduction to data manipulation, spatial data analysis and statistical modeling.**

University of Georgia, Athens – October 8<sup>th</sup>-10<sup>th</sup> 2019

Tuesday, 9:00am – 5:00pm; Wednesday, 9:00am – 5:00pm; Thursday, 8:30am – 3:00pm

Instructors: Melanie A. Murphy and Jeffrey S. Evans

The statistical program R has become the de facto standard in the statistical, spatial and ecological modeling communities. Due to an extensive academic user community, developing R libraries, the software is readily extendable to encompass a very large variety of statistical and analytical approaches including: complex multi-criteria data summaries, linear/nonlinear regression, machine learning, spatial statistics, time series, ordination, imputation, clustering and wildlife modeling with well-developed spatial object classes. Additionally, customized analysis is easily implemented through scripting, user-written functions and integration of external code support. R is object oriented with a complex scoping language that provides a robust programming environment. The software also has the capacity for online access, database connection and flexible data import functionality. The R programming environment, combined with extensive “built-in” functions provides robust data analysis, manipulation and modeling to both novice and advanced users.

This workshop will focus on spatial analysis and modeling in R using experiential learning where students are expected to formulate and implement analysis. As such, participants will be required to conceptualize, develop procedural steps and then write R code. Instructors will provide brief background followed by “hands on” interactive guidance to assist in this process. The content and structure of the exercises will be high-paced with the goal to build basic knowledge in programming, complex spatial analytical methods and statistics. We will illustrate the flexibility of R for implementing specialized analysis that leverages a variety of data and novel statistical methodologies. Participants can expect to come away with skills to perform exploratory analysis, implement basic modeling approaches and perform complex data manipulation, summary and query tasks.

## DAY 1 - TUESDAY, OCTOBER 8<sup>TH</sup>

### Section 1 (9:00am – 12:00pm) – Foundations of R for spatial analysis

Introduction to coding structure, object classes, data manipulation, writing functions, for loops and reading/writing spatial data. We will introduce the basic foundations of R logic and coding syntax, object-oriented structure, object classes and looping. Building on these basics, we will then cover data manipulation of the various spatial class objects introducing indexing, query, merging and subsetting data. We will also introduce creating, reading and writing vector spatial classes. For those wishing to improve their R skills in preparation for the workshop, some suggested foundation tutorial materials will be made available.

#### Learning objectives:

1. *Understanding vector, data frame, matrix, list object types and other object classes*
2. *Understanding base functions for manipulating and query of data.*
3. *Understanding bracket “indexing” and data manipulation.*
4. *Understanding the use of apply functions*
5. *Implementing “for/while” loops for repetitive task and simulation.*
6. *How to write functions*
7. *Creating, reading and writing spatial classes.*

### Section 2 (1:00pm – 5:00pm) – Spatial data query, manipulation, overlay and distance/neighbor analysis

One limitation to full implementation of spatial analysis in R is ability to implement some basic GIS functions. In this section, we will cover various vector overlay procedures, distance and proximity analysis. We will focus on spatial data analysis, starting with data manipulation, query and overlay. We will also cover distance and proximity analysis thus, providing a foundation to topics covered later in the workshop (e.g., assessing spatial autocorrelation, network analysis). To provide a context to common spatial analytic tasks, we will replicate a few tasks available in common GIS software and then extend them into specialized analysis.

#### Learning objectives

1. *Understanding vector overlay analysis and translate complex analysis to workflows*
2. *Build skills for using available libraries/functions to replicate, customize and batch common GIS tasks.*
3. *Ability to conduct distance and proximity analysis on vector data.*

## DAY 2 - WEDNESDAY, OCTOBER 9<sup>TH</sup>

### **Section 3** (9:00am – 12:00pm) – Spatial autocorrelation and testing model assumptions.

Understanding spatial autocorrelation is fundamental for understanding spatial process, testing model assumptions and conducting spatial analysis. In this section, we will cover what is spatial autocorrelations, importance and implications of spatial autocorrelation in modeling ecological systems, how to assess spatial autocorrelation (global and local), and how to test model assumptions using spatial autocorrelation tests.

Learning objectives:

1. *Understand how and why to test for spatial autocorrelation.*
2. *Be able to interpret spatial autocorrelation results for both population and sample data.*
3. *Implement spatial autocorrelation analyses to test parametric model assumptions.*

### **Section 4** (1:00pm – 3:00pm) – Raster data

Introduction to raster analysis including moving window, overlay and integration with vector data.

Learning objectives:

1. *Understanding import/export of raster formats and raster data structures*
2. *Understanding univariate, overlay and descriptive raster analysis*
3. *Understanding raster moving window analysis*
4. *Applying functions to rasters*
5. *Implementing raster and vector integration*

### **Section 5** (3:00pm – 5:00pm) – Quantifying landscape structure

Will provide a general understanding of landscape pattern, fragmentation and gradients including derivation and implementation of landscape metrics.

Learning objectives:

1. *Implementing “landscape metrics” using raster and vector data*
2. *Gradient metrics and surface characterization*
3. *Incorporating pattern and gradient metrics into statistical models*

### DAY 3 - THURSDAY, OCTOBER 10<sup>TH</sup>

#### **Section 6** (8:30am – 12:00pm) – Graph theoretical and gravity models

We will provide an overview to graph theoretical approaches with emphasis on gravity models. There are many ways graphs can be implemented to understand population structure and relate that structure to landscape characteristics (see Dyer and Nason 2004). In this exercise, we will focus on one specialized case. Gravity models are a type of inferential model that exploit graph characteristics. Gravity models include both at site (nodes) and between site (edges) landscape data. In this section, we will use the gravity model framework to build an empirical model of connectivity for a Columbia spotted frog dataset in central Idaho (Murphy et al. 2010).

Learning objectives:

1. *Understanding graph structures*
2. *Implementing the gravity model form*
3. *How to structure and specify a gravity model in R, AIC and model selection*
4. *Implement concepts from sections 1-5 in a real analytical framework (spatial objects, manipulating spatial data, spatial weight matrices, extract function, etc.).*

#### **Section 7** (1:00pm – 3:00pm) – Predictive models with Random Forests

We will introduce the conceptual and mathematical foundations of recursive partitioning, ensemble and Random Forests methods along with implementation of a binomial predictive model in R and methods for model selection, significance testing and validation of fit and performance. Time allowing, we will also cover overfit, the class imbalance problem and multi-model approaches for sample convergence.

Learning objectives:

1. *Concepts and caveats of recursive partitioning, ensemble methods and Random Forests*
2. *Background on “presence-only” versus “pseudo-absence” modeling approaches*
3. *How to structure data for a spatial Random Forests model*
4. *Implementation of a Random Forest model, including class imbalance, model selection and multi-model approaches for sample convergence.*
5. *Model validation*

## Class preparation

**Please do not download class material until Monday, Oct 7<sup>th</sup> so we can ensure that all materials are finalized.** After download please unzip the archive so that the directory structure stays intact. Place the directory “spatialR” immediately under the root directory (eg., C:/) and not the desktop or downloads. This will make paths manageable.

**spatialR workshop materials download:** <https://s3.amazonaws.com/RTools/spatialR/spatialR.zip>

Installation of R (>3.6.0) from [CRAN](https://cran.r-project.org/), require packages and a code editor should be completed before the first class. You will need to provide your own laptop, running a 64bit OS with at least 4GB of RAM. We are using Windows as our teaching platform. Other platforms (eg., Mac OS X, Linux) will not be supported and are used at your own risk. For Mac users; there have been numerous installation issues with rgdal and rgeos, which are two required spatial packages, so you may want to consider using a Windows partition such as Wine. Mostly, there are not issues for Linux users, but you will need to get the packages up and running before the first class. Links to data and exercises are included at the end of this document so, if you have issues downloading them please contact: [jeffrey.evans@tnc.org](mailto:jeffrey.evans@tnc.org) or [melanie.murphy@uwyo.edu](mailto:melanie.murphy@uwyo.edu)

We will be using a code editor and **NOT** RStudio as, it does not play well with some of the S4 spatial classes. I would highly recommend [Notepad++](#) but any IDE code editor with syntax highlighting for R will suffice, TINN-R, UltraEdit and, for you advanced users, ESS/EMACS are good alternatives. For the required R libraries you can copy-and-paste the below script. Please note that, in Windows, you will need to right click in the R icon and select “**Run as Administrator**” before installing libraries. If you are not administrator libraries will be installed in a personal library located in “My Documents”, which is highly undesirable.

##### START CODE #####

```
# Set site library to C:/Program Files/R/R-version/library
.Library.site <- file.path(chartr("\\", "/", R.home()), "library")
.libPaths(file.path(chartr("\\", "/", R.home()), "library"))

# set a CRAN mirror
local({r <- getOption("repos")
  r["CRAN"] <- "https://cran.cnr.berkeley.edu"
  options(repos=r)})

# Install libraries
install.packages( c("devtools", "sp", "sf", "spdep", "raster", "rgeos", "spatstat",
  "rgdal", "maptools", "spatialEco", "RANN", "RColorBrewer",
  "spatial", "Matrix", "nlme", "randomForest", "SDMTools",
  "rfUtilities", "vegan", "spgwr", "yalp", "classInt",
  "cluster", "e1071", "rgl", "rms", "velox",
  "landscapemetrics", "igraph") )

remotes::install_github("jeffreyevans/GeNetIt")
##### END CODE #####
```

I would also, highly recommend installing [Rtools](#). This is a small piece of software that allows compiling R packages from source. If you are installing development versions from places like GitHub, this is a must.

Only used if we are having versioning/development issues with our packages on CRAN:

```
remotes::install_github("jeffreyevans/spatialEco")
remotes::install_github("jeffreyevans/rfUtilities")
```

Note; if you are using a Mac to install rgdal and rgeos please follow the instructions here: [http://tlocoh.r-forge.r-project.org/mac\\_rgeos\\_rgdal.html](http://tlocoh.r-forge.r-project.org/mac_rgeos_rgdal.html)

Alternately, you can try:

```
install.packages("rgdal",repos="http://www.stats.ox.ac.uk/pub/RWin")
install.packages("rgeos",repos="http://www.stats.ox.ac.uk/pub/RWin")
```

The attached “Rprofile.site” file is an ASCII file with configuration options for R. This is not necessary but, very helpful in automatically defining your library path and default repository along with some other helpful settings and functions. Just drop the file in “C:\Program Files\R\R-3.6.1\etc” and replace the one already there.