

Introduction to Kriging in R

Nabil A.

October 14, 2015

Introduction

When I first started working with spatial data in R, it took me a while before I could get used to it. There were several obstacles, not the least of which being just to find which tutorials in which sequence would best help me grasp the larger picture of SP* classes. My own personal difficulties aside, I eventually acclimated myself to them, and joined the r-sig-geo group to follow up with any developments pertinent to my work.

So, partly as a means of contributing to documentation that would help my own code be more readily understood by others, and partly due to finding that professionals on the r-sig-geo group were encountering some similar questions I had when I first started, I thought I would present an overview of some of the functionality available for working with spatial data in R.

This document, though, is intended to be an introduction to working with kriging in R. A familiarity with kriging is already assumed; there are already more comprehensive resources available for anyone interested, whether broad overviews or more mathematical expositions with all the gory matrix algebra details. In particular, Allison Lassiter's blog (<http://allisonlassiter.com/author/allison-lassitergmail-com/>) was a helpful resource when I was just getting started.

Packages Used

In working with spatial data the `sp` library is essential, and what many other packages build off of. More specifically, the spatial classes (e.g., `SpatialPoints`, `SpatialPointsDataFrame`, etc) are defined in the `sp` package, while spatio-temporal classes are defined in the `spacetime` package.

For working with spatial (and spatio-temporal) data, we use the `gstat` package, which includes functionality for kriging, among other many things.

```
library(sp)
library(gstat)
```

Dataset

The data we are using is the `meuse` dataset, which comes with the `sp` package.

```
# packages for manipulation & visualization
suppressPackageStartupMessages({
  library(dplyr) # for "glimpse"
  library(ggplot2)
```