Thoughts on Spatiotemporal Data Standards in R

Blair Christian
blair.christian@gmail.com

Introduction

Spatiotemporal data is any set of observations which have space and time components. There are rich tool kits in R for working on either spatial or temporal data, but few which support spatiotemporal data. Currently, it is necessary to manually slice and dice your spatiotemporal data in order to apply these existing R tools. This lack of infrastructure in the S4 object sense has greatly hindered the use and development of spatiotemporal modeling. This is a proposal to add a small amount of structure in the form of S4 objects in order to leverage the existing spatial and temporal packages in R in a seamless manner, in addition to creating an environment conducive to code reuse. I am thinking about this with a possible future endpoint in the spirit of bioconductor or Rmetrics. I'll put a copy of this up on an informal website, http://www.isds.duke.edu/ jbc30/SpatioTemporal/ and move it to R-forge as we develop this.

Audience

End User

As an end user, some use cases include loading the data one time, then calling one or more models from one or more packages. For example, being able to run both, say, spBayes models and INLA models, in order to compare various computational and model properties. (eg performance in practice, comparison of predictive errors or other quantities of interest)

Model Maker

As a model maker, I don't want to reinvent the wheel. If other packages provide adequate functions for some needed algorithm, a data standard should help especially in pre and possibly in post processing.

Data Provider

As a data provider, one would like to provide data in an easy to obtain, easy to work with and be easy to cite. With Amy Braverman's comments in mind from the recent SAMSI conference, if NASA could provide data access similar to the NIH's entrez system. (something in process now, they have almost 4 Pb of imaging data at NASA, most of which isn't currently available. Metadata allowing the data provider room for descriptions of how the data were collected, who to contact with questions, a website for more information and a citation for using the data also

To summarize the benefits of a spatiotemporal classes and methods:

- Reproducible data analysis (see the bioconductor policy of R's bioconductor project)
- increase code reuse
- increase productivity
- possibility for better context/information for datasets (a metadata requirement)

Spatiotemporal Data Standards

Some general needs for a data representation include:

- data import from flat files or relational database
- structure for metadata (see bioconductor)
  - creator, maintainer, location on web, contact info, algorithm/satellite used, dates, original purpose/funding, citation of data
- change of support tools, foe example applied to upscaling/downscaling
  - (methods for sp, which could then be extended to spatiotemporal)
- registration?
- data export and visualization
  - export in kml format for portability
  - google maps, for example see swine flu tracker
  - other types of visualization relevant for data, eg trip library for following the trips of tagged animals
  - generic movie showing how grid data- kriging results in this example- change over time, via the animation package [Xie, 2009]

- Methods for accessing data
  - time series by location
  - spatial data at a time
  - function of all spatial or all temporal (eg mean, max, etc) at a given location/time (again, something that could be implemented for Spatial class and inherited by a spatiotemporal class)

There are a few assumptions I have been making, including:

- R is the right place
- can handle big datasets (with out of core/db storage)
- easier/more convenient to call "real" code in C/fortran, etc from R than rolling ones own

There are some risks

- too much structure
- too hard to use
- audience not ready to adopt (not seen as a problem yet)

Types of Spatiotemporal Data

The combinatorics of spatiotemporal data are large, and may increase with change of support methods. I really don't have any idea what the best way to proceed here is, but I have a few stabs that aren't so elegant. Thoughts? What do you want supported asap on the spatial side? temporal side?

I have one set of thoughts about the high level design (see section below)

- store data in 3 parts, link with unique identifiers; spatial, temporal, data observations
- spatial extends/implements Spatial in sp package [Pebesma and Bivand, 2005]
- temporal part extends/implements a time series or fda or splines or . . . package

There is a bit of existing code that should serve as some representation of the need for spatiotemporal data in different areas:

- spBayes [Finley et al., 2009], inla (example of models sharing input format that could be compared)
- geoR [Jr and Diggle, 2001], fields [Furrer et al., 2009], etc (retrieve those spatial objects at a given time)

- tseries, fda [Ramsay et al., 2009], its [Portfolio et al., 2009], zoo [Zeileis and Grothendieck, 2005], xts [Ryan and Ulrich, 2009] etc (retrieve those temporal objects at a given location)
- leverage code (eg bioBase [Gentleman et al., 2004])- if there's a package for spatial epi that is useful (eg trip [Sumner, 2009]), then we should embrace code reuse instead of reinventing the wheel

Wishlists

Here are some thoughts that are random and not comprehensive. Email me to add some.

Short Term Wishlist

- focus on data storage
  - easy to store
  - easy to retrieve
  - some basic EDA for spatiotemporal
    - movies
    - standard epi plot over time? (epi spatial package?)
    - seamless spatial views at time
    - seamless temporal views at location
- S4 classes and methods
- aid dissemination (google maps display)
- export to standard format (kmz? for google maps, GRASS, arcGIS, ...)
- examples?
- vignette?
- start small? (regular point/grid?)

Long Term Wishlist

- database support for massive (out of core) datasets
- advice to data providers (eg NASA, EPA, etc) for online data access (see NIH stuff, entrez)
- people who's advice I would like from an OO + R perspective: Bioconductor team (Seth Falcon? bioBase authors?); Dirk Eddelbuettel; Dough Bates; whoever reads this far ...

Example Class

An example starting place

I'm working with monitoring data at the EPA right now for 1-3 gasses. Some sites only monitor 1 gas at one set of intervals (say, hourly). Another, overlapping set of sites measures other gasses at other intervals (say, every three days). From this data, I use a variety of kriging and related methods to interpolate and predict gas levels at locations in between the sites, and at future points in time. The approaches I'm using are pretty much out of the Banerjee et al. [2004], so a big assumption is separability in temporal and spatial covariances. In my case, space is treated as continuous and I'll be treating time as regular for now. My classes will be focused only on this in the short term. I'll put up some info on the classes I use next week or the week after for those of you who are interested.

Here are some first thoughts at a very general (eg ”sp” library type) approach to classes for spatiotemporal data. Please let me know if I have made any gross mistakes here- it's been a long while since I played with S4 classes. My first thoughts are to have a main class, SpatialTemporalDataFrame, which has a unique set of spatial objects, a unique set of temporal objects, and a dataframe with the data that maps to a unique spatial location and time. My first go is something like this:

```
## unique set of spatial locations
##
## basically a container for one or more points or polygons
##
## eg something from the sp package,
##    such as SpatialPoints, SpatialPixels, SpatialPolygons
library(sp)
setClass("stSpace",
         representation(s.id="integer",
                        location="Spatial"))

validstSpaceObject <- function(object) {
  if(length(object@s.id) == length(object@location)) TRUE
  else paste("Unequal s.id, location lengths: ", length(object@s.id), ", ",
             length(object@location), sep="")
}
## assign the function as the validity method for the class
setValidity("stSpace", validstSpaceObject)


## I guess you could make it something like a
## SpatialPointsDataFrame, and put the unique ID in the data frame?
## but for generality, it would be nice to have a matching structure below
## for the time part


## unique set of time stamps
## basically a container for a time and date stamp.
library(timeDate)
setClass("stTime",
         representation(t.id="integer",
                        timedate="timeDate"))

validstTimeObject <- function(object) {
  if(length(object@t.id) == length(object@timedate)) TRUE
  else paste("Unequal t.id, timedate lengths: ", length(object@t.id), ", ",
             length(object@timedate), sep="")
}
## assign the function as the validity method for the class
setValidity("stTime", validstTimeObject)


## a dataframe subsettable by (unique) spatial and/or temporal ids
setClass("stDataFrame",
         representation(s.id="integer",
                        t.id="integer",
                        df="data.frame"))


## Seems like these should all be virtual classes?
##
## basically a mini relational database
## possibly stored that way in the future for large datasets
## see: RSQLite, RSQLiteMap packages
setClass("SpatialTemporalDataFrame",
         representation(spatial="stSpace",
                        temporal="stTime",
                        data="stDataFrame",
metadata="list"))
```

Conclusion

Thoughts?

References

S. Banerjee, B.P. Carlin, and A.E. Gelfand. *Hierarchical modeling and analysis for spatial data*. Chapman & Hall/CRC, 2004.

Andrew O. Finley, Sudipto Banerjee, and Bradley P. Carlin. *spBayes: Univariate and Multivariate Spatial Modeling*, 2009. URL `http://CRAN.R-project.org/package=spBayes`. R package version 0.1-3.

Reinhard Furrer, Douglas Nychka, and Stephen Sain. *fields: Tools for spatial data*, 2009. URL `http://www.image.ucar.edu/Software/Fields`. R package version 5.02.

Robert C Gentleman, Vincent J. Carey, Douglas M. Bates, et al. Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, 5: R80, 2004. URL `http://genomebiology.com/2004/5/10/R80`.

Paulo J. Ribeiro Jr and Peter J. Diggle. geoR: a package for geostatistical analysis. *R-NEWS*, 1(2):14–18, June 2001. URL `http://CRAN.R-project.org/doc/Rnews/`. ISSN 1609-3631.

Edzer J. Pebesma and Roger S. Bivand. Classes and methods for spatial data in R. *R News*, 5(2):9–13, November 2005. URL `http://CRAN.R-project.org/doc/Rnews/`.

Portfolio, Risk Advisory Group, and Commerzbank Securities. *its: Irregular Time Series*, 2009. URL `http://CRAN.R-project.org/package=its`. R package version 1.1.8.

J. O. Ramsay, Hadley Wickham, Spencer Graves, and Giles Hooker. *fda: Functional Data Analysis*, 2009. URL `http://CRAN.R-project.org/package=fda`. R package version 2.1.2.

Jeffrey A. Ryan and Josh M. Ulrich. *xts: Extensible Time Series*, 2009. URL `http://CRAN.R-project.org/package=xts`. R package version 0.6-7.

Michael D. Sumner. *trip: Spatial analysis of animal track data*, 2009. R package version 1.1-2.

Yihui Xie. *animation: Demonstrate Animations in Statistics*, 2009. URL `http://animation.yihui.name`. R package version 1.0-4.

Achim Zeileis and Gabor Grothendieck. zoo: S3 infrastructure for regular and irregular time series. *Journal of Statistical Software*, 14(6):1–27, 2005. URL `http://www.jstatsoft.org/v14/i06/`.