Practical Spatial Statistics & Econometrics with R

Session 4: Computing Experimental Variograms - Part I

Saif Ali, IIIT Delhi

How to excel at spatial stats (or anything else)?

Understanding

Clear conceptual understanding

Listening, Reading, Thinking, Writing

Skill

Apply understanding to real world problems.

Doing, Trying, Failing, Coding

~~Watching to a lot of lectures (like this one)~~

~~Reading many programming books~~

Pause and Play frequently!

What should we know/will we learn in this session?

Understanding

What we should know:

- The idea/definition of spatial stationarity
- The definition of a semivariogram
- Estimator for experimental variogram
- Variogram cloud

Skill

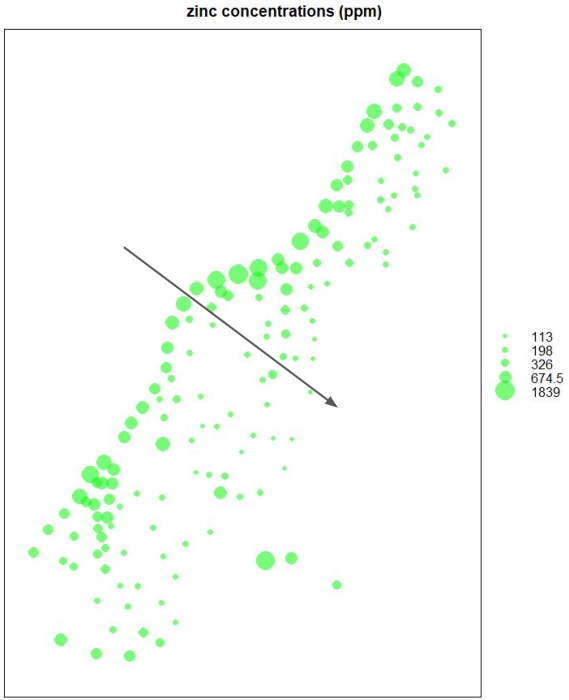
What we should have already done:

- Made a spatial points dataframe using the `meuse` data set

What we will do now:

- **Consider spatial stationarity**
- **Make a variogram cloud**
- **Estimate a variogram**
- **Remove trend from data**

Stationarity - BEFORE we start estimating a variogram



Looking at this realization, can we *decide* on a spatially stationary stochastic process to model zinc concentrations?

Cannot verify stationarity but can rule it out.

Look carefully. There is a trend from NW to SE.

Trends imply non-stationarity. We have to remove the trend.

Demo 4: Live Coding Session with R

Exercise

```
lzn.vgm.detrend = variogram(log(zinc)~x+y, sp.data.in)
```

What does the clause "`log(zinc)~x+y`" mean in the above code fragment?

Answer: It means that we want to estimate the experimental variogram after removing the trend from the zinc variable. We want to estimate the trend using `x` and `y` as regressors. We believe that trend is purely a function of location. This may or may not be true and it depends on how knowledgeable we are about heavy metal pollution.

Summary

- **Wrote our very first R function!**
- **Estimated an experimental variogram using `meuse` data set**
- **Compared variograms with and without trend removal**
- **Learned about a new library - plotly**
- **R functions, types**
  - `variogram, plot_ly`