M 944, SPATIAL STATS, FALL 2013 **HW 2** *DUE: Monday, Sept 23*

This assignment should help to familiarize yourself with some of the standard models of spatial covariance / correlation.

General Rule for this course: Don't just provide code and results. You need to interpret every result that you are submitting. You may be concise – a sentence or two is usually sufficient.

P1

(a) Show that the isotropic spherical correlation function $\rho(t)$ equals 0 when $t = \theta$, i.e. the correlation length = θ where θ denotes the range parameter. In other words, for the spherical model, the range parameter is the effective range (or correlation length).

(b) Determine the correlation lengths (= effective ranges) of the following isotropic correlation functions parameterized using the range parameter θ : exponential, powered exponential, Gaussian, rational quadratic. Note: for all these the correlation length = the distance at which the correlation reaches 0.05.

P2 Assume the following 6 locations in 2-d space:

```
sx = c(2,3,4,3.5,2,2.5)
sy = c(1,1.5,2,3,3.5,4)
```

a) First: plot the locations. Then, use R to define a function that creates a pairwise distance matrix as follows

> The S (or R) function "**outer**" calculates generalized outer products of arrays. Applying this function to two vectors **a** ,**b**, does the following: It creates a matrix with # rows = length of **a**, and # columns = length of **b**, and each element (i,j) of this matrix contains the combination of a_i , b_j plus any operation applied to these two values. For example stating: FUN = "+" will add these two values, FUN = "*" will multiply, etc.

```
twod.dist = function(sx,sy) {sqrt(outer(sx,sx,"-")^2 + outer(sy,sy,"-")^2)}
```

Execute the function, and then apply it to the two vectors above. This creates the pairwise distance matrix for the above data. Print this matrix with a two digit precision (use the `round (...,2)` command) as follows:

```
distmat = twod.dist(sx,sy);   round(distmat,2)
```

b) At the above locations we observe a response which is a realization from an isotropic spatial random field with a constant mean, and constant variance that is equals to 1, and some correlation function with correlation length 2.5. In this exercise we use the matrix of pairwise distances above and calculate the variance – covariance matrix according to a given covariance function model. Write a script that calculates the following (use only elementary operations, do not use geoR specific functions)

(1) the exponential covariance matrix with the above parameters,

(2) the Whittle's covariance matrix with the above parameters. Note for the Whittle correlation function we have the relation: correlation length = 4.744 * θ , where θ is the range parameter.
For statistics Ph.D. students only: Prove this relation (hint: you may need to solve this numerically)

(3) the Gaussian covariance matrix with the above parameters.

Hint for (1),(2), and (3): You need to first find the range parameter for each model (use results from P1 b and d)) and then apply the covariance function to the entries in the pairwise distance matrix of a). Print the results with two digits. (use `round` again). This may look like this (eg: for the exponential cov model)

```
r.expo = 2.5 / 3  # calculates the range parameter
cov.expo = exp(-distmat/r.expo); round(cov.expo,2)
```

c) Obtain an overlay line plot of the three models in b over a grid of pairwise distances from 0 to 4)

```
eg:  dists = seq(.05,4,by=.05);
    plot(dists,exp(-dists/r.expo),type='l',ylim=c(0,1))  # expo model
    lines(dists,…  )   # Whittle model
    lines(dists,…  )   # Gaussian model
```

d) Use the geoR function `cov.spatial` to obtain the same plots as in c). Note that the Whittle model equals the Matern model with smoothness parameter $\nu = 1.5$ (called kappa in geoR). Read the help file for cov.spatial for all the specifics. This may look like this:

```
plot(dists,cov.spatial(dists,"expo",cov.pars=c(1,r.expo)),type='l',ylim=c(0,1))
```

etc. for the lines command.

P3: Consider the Calcium Data in geoR. They consist of soil calcium measurements in Brazil in three regions. Please read the helpfile on “ca20”.

a) Read the data and subset by only using region 3. This is the experimental region with 116 data points. Plot the data.

Hint: use:

```
data(ca20);
ca.reg3 = subset(ca20,area==3)
plot(ca.reg3)
```

Interpret the resulting graph.

b) Use the “variog” function to calculate the semivariogram. First obtain the omnidirectional (isotropic) semivariogram and compare the classical estimator with the robust estimator. Use:

```
plot(variog(ca.reg3,trend="cte", max.dist = 500,estim="classical") )
```

Use: `estim=″modulus″` for the robust estimator.
Is there a big difference between the two estimators?

c) Try a different trend: How about trend= ~ altitude (This fits a linear regression to altitude as predictor and estimates the semivariogram from the residuals)

Use:

```
plot(ca.reg3, trend= ~ altitude) # this plots the residuals
plot(variog(ca.reg3,trend=~altitude, max.dist = 500,estim="classical") )
```

Next try the following:

`trend = ="1st"` (this fits a linear regression in the coordinates), or
`trend = ="2nd"` (this fits a quadratic regression in the coordinates). Interpret the graphs.