# Stat 534 Exam 1

Kenny Flagg

March 10, 2017

1. *Suppose we have an intrinsically stationary process with semivariogram*

$$\left(\frac{1}{2}\right) Var\left(Z(\mathbf{s}_i) - Z(\mathbf{s}_j)\right) = \gamma\left(\mathbf{s}_i - \mathbf{s}_j\right) = \gamma\left(\mathbf{h}_{ij}\right)$$

*On an earlier homework you showed that*

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \gamma\left(\mathbf{h}_{ij}\right) \leq 0$$

*for any sites* $\mathbf{s}_i$, $i = 1, \ldots, n$ *and for any constants* $a_i$, $i = 1, \ldots, n$ *with* $\sum_{i=1}^{n} a_i = 0$ *but you did it under an assumption of second order stationarity. We will now establish it in general.*

(a) *First show that* $-\left(\frac{1}{2}\right)\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \left(Z(\mathbf{s}_i) - Z(\mathbf{s}_j)\right)^2\right\} = \left\{\sum_{i=1}^{n} a_i Z(\mathbf{s}_j)\right\}^2$.

Proof:

$$\begin{aligned}
-\left(\frac{1}{2}\right)\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \left(Z(\mathbf{s}_i) - Z(\mathbf{s}_j)\right)^2\right\} &= -\left(\frac{1}{2}\right)\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \left(Z(\mathbf{s}_i)^2 - 2Z(\mathbf{s}_i)Z(\mathbf{s}_j) + Z(\mathbf{s}_j)^2\right)\right\} \\
&= -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j Z(\mathbf{s}_i)^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j Z(\mathbf{s}_i)Z(\mathbf{s}_j) \\
&\quad - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j Z(\mathbf{s}_j)^2 \\
&= -\frac{1}{2}\sum_{i=1}^{n} a_i Z(\mathbf{s}_i)^2 \sum_{j=1}^{n} a_j + \sum_{i=1}^{n} a_i Z(\mathbf{s}_i) \sum_{j=1}^{n} a_j Z(\mathbf{s}_j) \\
&\quad - \frac{1}{2}\sum_{j=1}^{n} a_j Z(\mathbf{s}_j)^2 \sum_{i=1}^{n} a_i \\
&= 0 + \sum_{i=1}^{n} a_i Z(\mathbf{s}_i) \sum_{j=1}^{n} a_j Z(\mathbf{s}_j) + 0 \\
&= \left\{\sum_{i=1}^{n} a_i Z(\mathbf{s}_i)\right\}^2.
\end{aligned}$$

(b) *Now take expectations of both sides to establish the result.*

$$\begin{aligned}
\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \gamma\left(\mathbf{h}_{ij}\right) &= \left\{\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \left(\frac{1}{2}\right)\left(E\left[\left(Z(\mathbf{s}_i)-Z(\mathbf{s}_j)\right)^2\right] - \left[E(Z(\mathbf{s}_i)-Z(\mathbf{s}_j))\right]^2\right)\right\} \\
&= E\left[\left(\frac{1}{2}\right)\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \left(Z(\mathbf{s}_i)-Z(\mathbf{s}_j)\right)^2\right\}\right] - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j (\mu(\mathbf{s}_i)-\mu(\mathbf{s}_j))^2 \\
&= E\left[-\left\{\sum_{i=1}^{n} a_i Z(\mathbf{s}_j)\right\}^2\right] - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \mu(\mathbf{s}_i)^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \mu(\mathbf{s}_i)\mu(\mathbf{s}_j) \\
&\quad - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \mu(\mathbf{s}_j)^2 \\
&= -E\left[\left\{\sum_{i=1}^{n} a_i Z(\mathbf{s}_j)\right\}^2\right] - \frac{1}{2}\sum_{i=1}^{n} a_i \mu(\mathbf{s}_i)^2 \sum_{j=1}^{n} a_j + \sum_{i=1}^{n} a_i \mu(\mathbf{s}_i) \sum_{j=1}^{n} a_j \mu(\mathbf{s}_j) \\
&\quad - \frac{1}{2}\sum_{i=1}^{n} a_i \sum_{j=1}^{n} a_j \mu(\mathbf{s}_j)^2 \\
&\leq 0.
\end{aligned}$$

2. *Let $X_0 \sim Gamma(\alpha, \beta)$ with the parameterization*

$$f(x) = \frac{1}{\Gamma(\alpha)\beta^{\alpha}} x^{\alpha-1}\exp(-x/\beta); \qquad x > 0$$

*and 0 elsewhere. Let $X_i \sim Gamma(\alpha_i, \beta)$ for $i = 1, \ldots, n$. We construct a one-dimensional regularly spaced random field at locations $i = 1, \ldots, n$*

$$Z(s_i) = X_0 + X_i; \qquad i = 1, \ldots, n.$$

*You can assume that $X_0, X_1, \ldots, X_n$ are independent.*

(a) *What is the distribution of $Z(s_i)$?*

Because $X_0$ and $X_i$ are independent Gamma random variables,

$$Z(s_i) = X_0 + X_i \sim \text{Gamma}(\alpha + \alpha_i, \beta).$$

(b) *Find $E(Z(s_i))$ and $Var(Z(s_i))$. You can use known properties of the Gamma distribution to answer this question, i.e. you can just write down the answer if you know it or can find it.*

$$E(Z(s_i)) = \frac{\alpha + \alpha_i}{\beta} \qquad \text{and} \qquad Var(Z(s_i)) = \frac{\alpha + \alpha_i}{\beta^2}.$$

(c) *Find $Cov(Z(s_i), Z(s_j))$ for $i \neq j$.*

If $i \neq j$,

$$\begin{aligned} Cov(Z(s_i), Z(s_j)) &= Cov(X_0 + X_i, X_0 + X_j) \\ &= Var(X_0) + Cov(X_0, X_j) + Cov(X_i, X_0) + Cov(X_i, X_j) \\ &= \frac{\alpha}{\beta^2} + 0 + 0 + 0 \\ &= \frac{\alpha}{\beta^2}. \end{aligned}$$

(d) *Is this a second-order stationary process? Justify your answer.*

This process is not second order stationary because the mean and variance of $Z(s_i)$ vary by location.

3. *The* `lansing` *data set in the* `spatstat` *package contains spatial locations of several different species of trees. We will be looking and comparing the distributions of black oaks and maples.*

```
require(spatstat)
data(lansing)
blackoak <- split(lansing)$blackoak
maple <- split(lansing)$maple
```

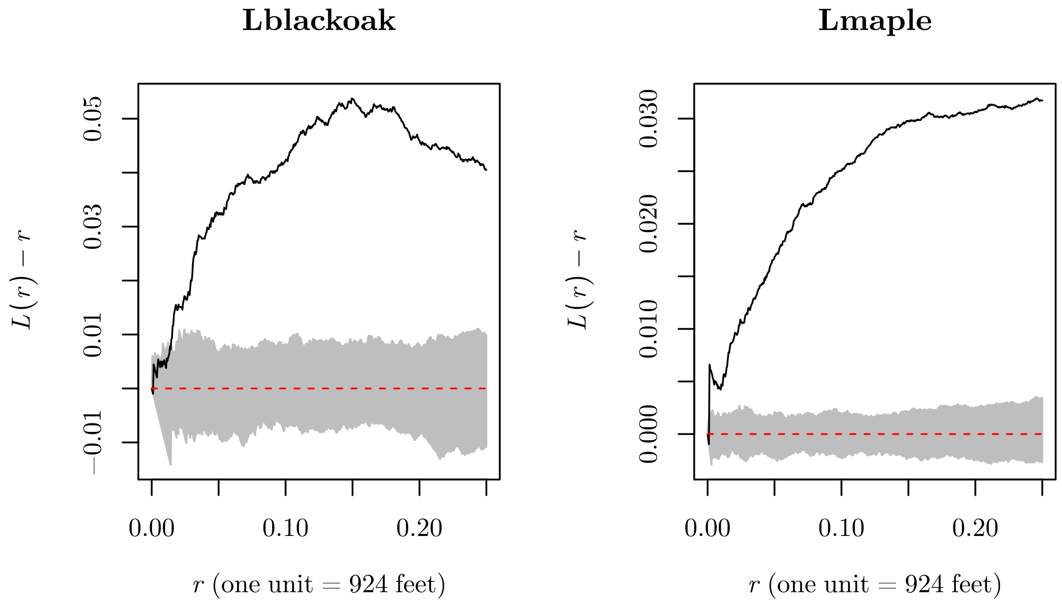
*The rectangular region is a unit square. Use the isotropic edge corrected version when applicable below. Answer the following questions. You will be computing several simulation envelopes below. Be patient and keep nsim=99, the default.*

(a) *What does the K function measure?*

The $K$ function measures the average number of events within a radius $h$ of a randomly-selected event, scaled by the intensity over the whole region.

(b) *It is often easier to interpret the L function than the K function. Based on the L function do the black oaks appear to be clustered or do they appear to be regularly distributed? Do the maples appear to be clustered or do they appear to be regularly distributed? Justify your answer. Simulation envelopes will help you give a better answer to this question.*

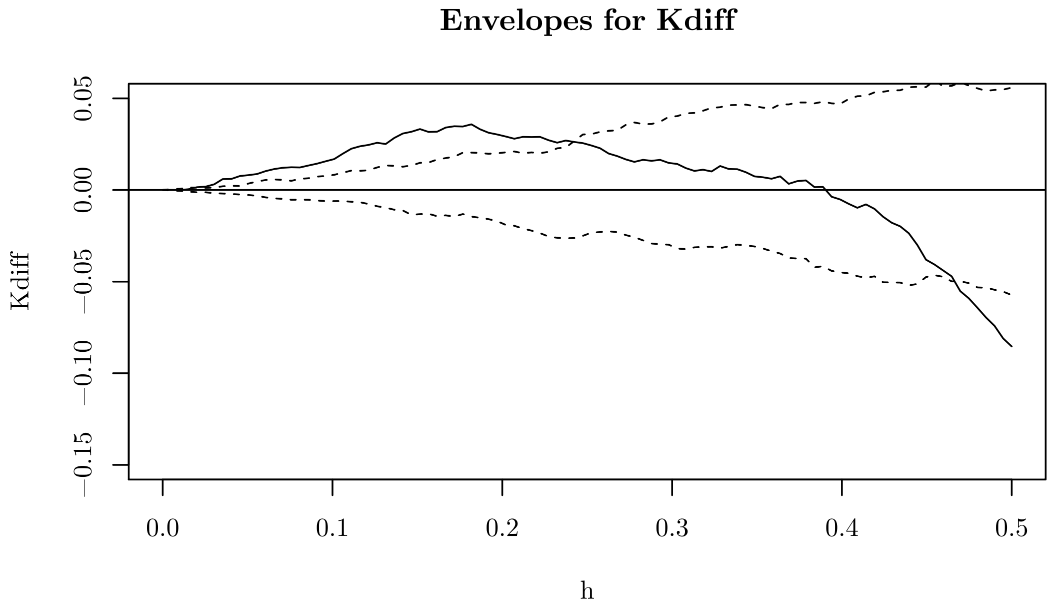
```
par(mfrow = c(1, 2))
Lblackoak <- envelope(blackoak, fun = Lest, correction = 'iso')
plot(Lblackoak,.-r ~ r, legend = FALSE)
Lmaple <- envelope(maple, fun = Lest, correction = 'iso')
plot(Lmaple,.-r ~ r, legend = FALSE)
```



The black oaks appear to be clustered because the $L$ function is larger than expected under complete spatial randomness, meaning trees tend to be closer together than expected. Likewise, the maples also appear to be clustered because their $L$ function is larger than expected under CSR.

(c) *Compare the two L functions and discuss whether or not the 2 processes appear to be the same. You can use the results from (a) but you should also look at the difference more formally using the following also provided in an attached script file.*

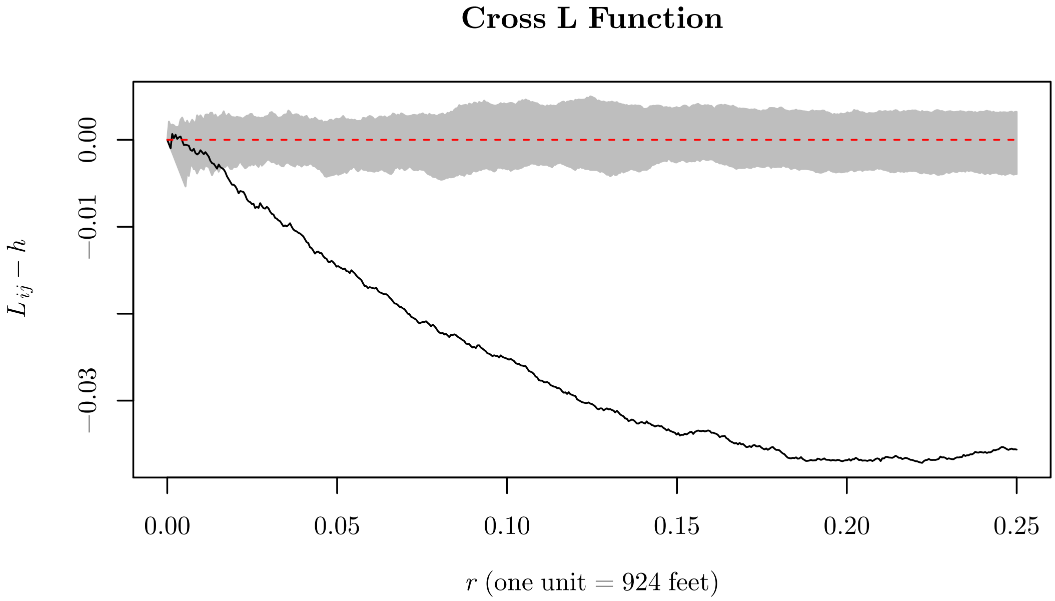
```
require(splancs)
# specify radii
h <- seq(0, 0.5, l = 100)
# get coordinates
tree.poly <- list(x = c(blackoak$x, maple$x), y = c(blackoak$y, maple$y))
# recompute the K functions
kblackoak <- khat(as.points(blackoak), bboxx(bbox(as.points(tree.poly))), h)
kmaple <- khat(as.points(maple), bboxx(bbox(as.points(tree.poly))), h)
# get the differences
k.diff <- kblackoak - kmaple
# generate the envelope
env <- Kenv.label(as.points(blackoak), as.points(maple),
                  bboxx(bbox(as.points(tree.poly))), nsim = 99, s = h)
# plot the results
plot(h, seq(-0.15, 0.05, l = length(h)), type = 'n', ylab = 'Kdiff',
     main = 'Envelopes for Kdiff')
lines(h, k.diff)
lines(h, env$low, lty = 2)
lines(h, env$up, lty = 2)
abline(h = 0)
```



At a glance, the plots in part (b) indicate that the processes are similar. For both, $L(h) - h$ increases quickly until about $h = 0.15$ units and then levels off. However, the $L$ function for black oaks is a bit larger than the $L$ function for maples. We formally test equality of the $L$ or $K$ functions with a simulation envelope for $K_{oak}(h) - K_{maple}(h)$. The observed difference in $K$ functions is not contained entirely in the envelope, so we have evidence that the locations of the two tree species result from two separate processes.

(d) *Plot $L_{ij} - h$ versus $h$ for black oaks and maples.*

```
Kplot <- envelope(lansing, Kcross, i = 'blackoak', j = 'maple')
plot(Kplot, sqrt(./pi)-r ~ r, ylab = expression(L[ij] - h),
     main = 'Cross L Function', legend = FALSE)
```



*What type of relationship between the point patterns of the two species of trees is indicated by this plot? Justify your answer.*

The cross $L$ function is smaller than expected under independence, indicating that fewer trees of one species than expected are found within a given distance of a tree of the other species. This suggests an inhibitory process where trees of the same species tend to cluster together rather than growing near trees of the other species.

(e) *Based on the above, comment on the null hypotheses of independence and random labeling.*

In part (c) we found evidence that the processes differ between the species, so we reject the hypothesis of random labeling. In part (d) we found evidence that trees of one species tend not to grow near trees of the other species, so we reject the hypothesis of independence.

4. *You were sent the wheat data set on a previous homework assignment. You want to predict the value of $Z$ (yield) at an arbitrary location. Assume a pure nugget effect model.*

   (a) *What are the kriging weights and what is the predicted value?*

```
wheat <- read.table('wheat.txt', header = TRUE)
nrow(wheat)

[1] 224

mean(wheat$z)

[1] 25.52701
```

   There are $n = 224$ observations so the kriging weights are $\lambda_i = \frac{1}{224}$ for all $i$. The predicted value is simply $\bar{Z} = 25.53$.

   (b) *What is the estimate of the sill?*

```
var(wheat$z)

[1] 55.50518
```

   The estimate of the sill is the sample variance, $\widehat{\sigma}^2 = 55.51$.

   (c) *What is the kriging standard error (note that this is a prediction error)?*

   The kriging standard error is $\sqrt{\widehat{Var}(Z) + \widehat{Var}(\bar{Z})} = \sqrt{\widehat{\sigma}^2 + \dfrac{\widehat{\sigma}^2}{n}} = 55.75.$

5. *Carbon-Nitrogen data example: We looked at estimating the semivariogram of the residuals from a simple linear regression model of total carbon on total nitrogen in class. We used* `gls` *to do this (as part of incorporating a spatial covariance structure into the regression) specifying an exponential covariance model and estimating the parameters using both maximum likelihood and REML. Let's check to see what the* `likfit` *function in the* `geoR` *package would return as parameter estimates (nugget, practical range, and partial sill) and see if the results are comparable. Some of the relevant R code is included in the attached script file. Compare the estimates on page 11 of the Spatial Regression notes and the estimates you get out of* `likfit`*. Use the same starting values.*

```
# Get the CN data.
CN.dat <- read.table('CN.dat', header = TRUE)
names(CN.dat) <- c('x', 'y', 'tn', 'tc', 'cn')
CN.lm <- lm(tc ~ tn, data = CN.dat)
resids <- residuals(CN.lm)

# Convert to a geodata object.
require(geoR)
resids.dat <- cbind(CN.dat$x, CN.dat$y, resids)
resids.dat <- data.frame(resids.dat)
names(resids.dat) <- c('x', 'y', 'resids')
resids.geodat <- as.geodata(resids.dat, coords.col = 1:2, data.col = 3)
```

```
# Use likfit() on the residuals.
CNlikexp.ml <- likfit(resids.geodat, cov.model = 'exponential',
                      ini.cov.pars = c(0.00159, 15),
                      fix.nugget = TRUE, nugget = 0,
                      lik.method = 'ML')
CNlikexp.reml <- likfit(resids.geodat, cov.model = 'exponential',
                        ini.cov.pars = c(0.00159, 15),
                        fix.nugget = TRUE, nugget = 0,
                        lik.method = 'REML')
CNlikexp.ml.nugget <- likfit(resids.geodat, cov.model = 'exponential',
                             ini.cov.pars = c(0.00159, 15),
                             fix.nugget = FALSE, nugget = 0.4 * 0.00159,
                             lik.method = 'ML')
CNlikexp.reml.nugget <- likfit(resids.geodat, cov.model = 'exponential',
                               ini.cov.pars = c(0.00159, 15),
                               fix.nugget = FALSE, nugget = 0.4 * 0.00159,
                               lik.method = 'REML')

# Now fit gls the models.
CNglsexp.ml <- gls(tc ~ tn, data = CN.dat, method = 'ML',
                   correlation = corExp(15, form = ~x+y))
CNglsexp.reml <- gls(tc ~ tn, data = CN.dat, method = 'REML',
                     correlation = corExp(15, form = ~x+y))
CNglsexp.ml.nugget <- gls(tc ~ tn, data = CN.dat, method = 'ML',
                          correlation = corExp(c(15, 0.4), form = ~x+y,
                                                nugget = TRUE))
CNglsexp.reml.nugget <- gls(tc ~ tn, data = CN.dat, method = 'REML',
                            correlation = corExp(c(15, 0.4), form = ~x+y,
                                                 nugget = TRUE))


# Put all this in a nice table.
compare <- data.frame(
  Model = paste('Exponential', rep(c('REML', 'ML'), 2)),
  'Nugget (gls)' = c(NA, NA,
                     coef(CNglsexp.reml.nugget$modelStruct$corStruct, unconstrained = FALSE)['nugget'],
                     coef(CNglsexp.ml.nugget$modelStruct$corStruct, unconstrained = FALSE)['nugget']),
  'Nugget (likfit)' = c(NA, NA,
                        CNlikexp.reml.nugget$nugget /
                          (CNlikexp.reml.nugget$sigmasq + CNlikexp.reml.nugget$nugget),
                        CNlikexp.ml.nugget$nugget /
                          (CNlikexp.ml.nugget$sigmasq + CNlikexp.ml.nugget$nugget)),
   'Sill (gls)' = c(CNglsexp.reml$sigma^2,
                    CNglsexp.ml$sigma^2,
                    CNglsexp.reml.nugget$sigma^2,
                    CNglsexp.ml.nugget$sigma^2),
   'Sill (likfit)' = c(CNlikexp.reml$sigmasq,
                       CNlikexp.ml$sigmasq,
                       CNlikexp.reml.nugget$sigmasq + CNlikexp.reml.nugget$nugget,
                       CNlikexp.ml.nugget$sigmasq + CNlikexp.ml.nugget$nugget),
   'Range (gls)' = 3 * c(coef(CNglsexp.reml$modelStruct$corStruct, unconstrained = FALSE)['range'],
                         coef(CNglsexp.ml$modelStruct$corStruct, unconstrained = FALSE)['range'],
                         coef(CNglsexp.reml.nugget$modelStruct$corStruct, unconstrained = FALSE)['range'],
                         coef(CNglsexp.ml.nugget$modelStruct$corStruct, unconstrained = FALSE)['range']),
   'Range (likfit)' = c(CNlikexp.reml$practicalRange,
                        CNlikexp.ml$practicalRange,
                        CNlikexp.reml.nugget$practicalRange,
                        CNlikexp.ml.nugget$practicalRange),
   check.names = FALSE
)
print(xtable(compare, digits = 5), include.rownames = FALSE, size = 'footnotesize')
```

| Model | Nugget (gls) | Nugget (likfit) | Sill (gls) | Sill (likfit) | Range (gls) | Range (likfit) |
|---|---|---|---|---|---|---|
| Exponential REML | | | 0.00171 | 0.00170 | 42.38463 | 41.60476 |
| Exponential ML | | | 0.00167 | 0.00167 | 40.98451 | 40.42383 |
| Exponential REML | 0.35623 | 0.35379 | 0.00177 | 0.00175 | 175.08927 | 168.48791 |
| Exponential ML | 0.36308 | 0.35955 | 0.00167 | 0.00166 | 144.92937 | 141.47099 |

Getting the output to be comparable took some work because `gls`'s "nugget" is the proportion of the total sill and its "sill" is the total sill, while `likfit`'s "nugget" is the actual nugget and its "sill" is the partial sill. To create the table above I adjusted the `likfit` output, adding the nugget to the partial sill so the "sill" column shows the total sill, and dividing the nugget by the total sill so the nugget column is the proportion.

The practical range estimates from `likfit` are a little smaller than those from `gls` but the sill and nugget estimates are essentially the same.