University of California, Los Angeles
Department of Statistics

Statistics C173/C273 **Instructor: Nicolas Christou**

Universal kriging

The Ordinary Kriging (OK) that was discussed earlier assumes a constant mean model given by

$$Z(s) = \mu + \delta(s)$$

where $\delta(s)$ has mean zero and variogram $2\gamma(h)$. Many times this is a too simple model to use. The mean can be a function of the coordinates X, Y, in some linear, quadratic, or higher form. For example the value of Z at location s can be expressed now as

$$Z(s_i) = \beta_0 + \beta_1 X_i + \beta_2 Y_i + \delta(s_i), \quad \text{linear}$$

Or

$$Z(s_i) = \beta_0 + \beta_1 X_i + \beta_2 Y_i + \beta_3 X_i^2 + \beta_4 X_i Y_i + \beta_5 Y_i^2 + \delta(s_i), \quad \text{quadratic, etc.}$$

If this is the case then we say that there is a trend of the polynomial type. We need to take this into account when we find the kriging weights. The predicted value $Z(s_0)$ at location s_0 will be again a linear combination of the observed $Z(s_i), i = 1, \cdots, n$ values:

$$\hat{Z}(s_0) = w_1 Z(s_1) + w_2 Z(s_2) + \cdots + w_n Z(s_n) = \sum_{i=1}^{n} w_i Z_i$$

where

$$w_1 + w_2 + \cdots + w_n = 1$$

Suppose now that a trend of the linear form is present. Then the value $\hat{Z}(s_0)$ can be expressed as

$$\hat{Z}(s_0) = \sum_{i=1}^{n} w_i Z_i = \sum_{i=1}^{n} w_i \beta_0 + \sum_{i=1}^{n} w_i \beta_1 X_i + \sum_{i=1}^{n} w_i \beta_2 Y_i + \sum_{i=1}^{n} w_i \delta(s_i)$$

or

$$\hat{Z}(s_0) = \sum_{i=1}^{n} w_i Z_i = \beta_0 + \beta_1 \sum_{i=1}^{n} w_i X_i + \beta_2 \sum_{i=1}^{n} w_i Y_i + \sum_{i=1}^{n} w_i \delta(s_i) \tag{1}$$

But also the value of $Z(s_0)$ can be expressed (based on the linear trend) as

$$Z(s_0) = \beta_0 + \beta_1 X_0 + \beta_2 Y_0 + \delta(s_0) \tag{2}$$

Compare (1) and (2). In order to ensure that we have an unbiased predictor we will need the following conditions:

$$\sum_{i=1}^{n} w_i X_i = X_0$$

$$\sum_{i=1}^{n} w_i Y_i = Y_0$$

and

$$\sum_{i=1}^{n} w_i = 1$$

As with ordinary kriging, to find the weights when a trend is present we need to minimize the mean square error (MSE) of prediction

$$\min \quad \left(Z(s_0) - \sum_{i=1}^{n} w_i Z(s_i) \right)^2$$

subject to the above constraints. This minimization will be unconstrained if we incorporate the 3 constraints in the objective function. The result is a system of $n+3$ equations for the linear trend.

If the trend is quadratic we will need $n+6$ equations, and $n+10$ equations for a three-dimensional quadratic trend, etc. On the next page the system of equations for the linear trend example is presented in matrix form.

Note: We should try to understand why the trend exists based on the nature of our data, use a simple form of the trend if possible, and avoid extrapolation beyond the available data. Once we decided about which trend to use, we subtract this trend from the observed data to obtain the residuals. We then use the residuals to compute the sample variogram, fit a model variogram to it, predict the values at the unsampled locations ("kriged" the residuals), and finally add the kriged residuals back to the trend.

The Kriging System with linear trend as a function of the x, y coordinates

$$
\begin{pmatrix}
\gamma(s_1-s_1) & \gamma(s_1-s_2) & \gamma(s_1-s_3) & \cdots & \cdots\ \gamma(s_1-s_n) & x_1 & y_1 & 1 \\
\gamma(s_2-s_1) & \gamma(s_2-s_2) & \gamma(s_2-s_3) & \cdots & \cdots\ \gamma(s_2-s_n) & x_2 & y_2 & 1 \\
\cdots & \cdots & \ddots & \cdots & \cdots & \vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots & \ddots & \cdots & \vdots & \vdots & \vdots \\
\gamma(s_n-s_1) & \gamma(s_n-s_2) & \gamma(s_n-s_3) & \cdots & \gamma(s_n-s_n) & x_n & y_n & 1 \\
x_1 & x_2 & \cdots & \cdots & x_n & 0 & 0 & 0 \\
y_1 & y_2 & \cdots & \cdots & y_n & 0 & 0 & 0 \\
1 & 1 & \cdots & \cdots & 1 & 0 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
w_1 \\ w_2 \\ \vdots \\ \vdots \\ w_n \\ \lambda_1 \\ \lambda_2 \\ \lambda_0
\end{pmatrix}
=
\begin{pmatrix}
\gamma(s_0-s_1) \\ \gamma(s_0-s_2) \\ \vdots \\ \vdots \\ \gamma(s_0-s_n) \\ x_0 \\ y_0 \\ 1
\end{pmatrix}
$$

Example of universal kriging:

Let's return to the simple example with the 7 observed points as shown below. We use the exponential semivariogram model with $c_0 = 0, c_1 = 10, \alpha = 3.33$, $\gamma(h) = 10(1 - e^{-\frac{h}{3.33}})$.

s_i	x	y	$z(s_i)$
s_1	61	139	477
s_2	63	140	696
s_3	64	129	227
s_4	68	128	646
s_5	71	140	606
s_6	73	141	791
s_7	75	128	783
$\mathbf{s_0}$	**65**	**137**	**???**

First we calculate the distance matrix as shown below:

$$\text{Distance matrix} = \left(\begin{array}{ccccccccc} & s_0 & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & s_7 \\ s_0 & 0.00 & 4.47 & 3.61 & 8.06 & 9.49 & 6.71 & 8.94 & 13.45 \\ s_1 & 4.47 & 0.00 & 2.24 & 10.44 & 13.04 & 10.05 & 12.17 & 17.80 \\ s_2 & 3.61 & 2.24 & 0.00 & 11.05 & 13.00 & 8.00 & 10.05 & 16.97 \\ s_3 & 8.06 & 10.44 & 11.05 & 0.00 & 4.12 & 13.04 & 15.00 & 11.05 \\ s_4 & 9.49 & 13.04 & 13.00 & 4.12 & 0.00 & 12.37 & 13.93 & 7.00 \\ s_5 & 6.71 & 10.05 & 8.00 & 13.04 & 12.37 & 0.00 & 2.24 & 12.65 \\ s_6 & 8.94 & 12.17 & 10.05 & 15.00 & 13.93 & 2.24 & 0.00 & 13.15 \\ s_7 & 13.45 & 17.80 & 16.90 & 11.05 & 7.00 & 2.65 & 13.15 & 0.00 \end{array}\right)$$

Using the universal kriging equations (assuming that there is a linear trend in our data) the weights, the predicted value, and the variance of the error of prediction are computed as follows:

Universal kriging using the variogram:

$$\boldsymbol{W} = \boldsymbol{\Gamma^{-1}}\gamma = \begin{pmatrix} 0 & 4.893 & 9.564 & 9.800 & 9.510 & 9.740 & 9.952 & 61 & 139 & 1 \\ 4.893 & 0 & 9.637 & 9.798 & 9.093 & 9.510 & 9.938 & 63 & 140 & 1 \\ 9.564 & 9.637 & 0 & 7.095 & 9.800 & 9.889 & 9.637 & 64 & 129 & 1 \\ 9.800 & 9.798 & 7.095 & 0 & 9.755 & 9.847 & 8.775 & 68 & 128 & 1 \\ 9.510 & 9.093 & 9.800 & 9.755 & 0 & 4.893 & 9.775 & 71 & 140 & 1 \\ 9.740 & 9.510 & 9.889 & 9.847 & 4.893 & 0 & 9.806 & 73 & 141 & 1 \\ 9.952 & 9.938 & 9.637 & 8.775 & 9.775 & 9.806 & 0 & 75 & 128 & 1 \\ 61 & 63 & 64 & 68 & 71 & 73 & 75 & 0 & 0 & 0 \\ 139 & 140 & 129 & 128 & 140 & 141 & 128 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}^{-1} \begin{pmatrix} 7.384 \\ 6.614 \\ 9.109 \\ 9.420 \\ 8.664 \\ 9.316 \\ 9.823 \\ 65 \\ 137 \\ 1 \end{pmatrix}.$$

The answer is:

$$\boldsymbol{W} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \\ w_7 \\ \lambda_1 \\ \lambda_2 \\ \lambda_0 \end{pmatrix} = \begin{pmatrix} 0.220 \\ 0.342 \\ 0.147 \\ 0.073 \\ 0.151 \\ 0.040 \\ 0.027 \\ -0.066 \\ 0.023 \\ 2.301 \end{pmatrix}.$$

The last 3 elements of the $\boldsymbol{W}$ vector are the Lagrange multipliers, $\lambda_0 = 2.3, \lambda_1 = -0.066, \lambda_2 = 0.023$. We can verify that the sum of the elements 1 through 7 is equal to 1, as it should be.

The predicted value is equal to:

$$\hat{z}(s_0) = \sum_{i=1}^{7} w_i z(s_i) = 0.220(477) + \cdots + 0.027(783) = 567.54.$$

The variance of the error of prediction is equal to:

$$\sigma^2(s_0) = \sum_{i=1}^{n} w_i \gamma(s_0 - s_i) + \lambda_0 + \lambda_1 x_{10} + \cdots + \lambda_k x_{k0}$$

Or

$$\sigma^2(s_0) = 0.220(7.384) + \cdots + 0.027(9.823) + 2.301 - 0.066(65) + 0.023(137) = 9.044.$$

Universal kriging using the covariance:
The same example is solved using the covariance function: $C(h) = 10e^{-\frac{h}{3.33}}$. Reminder: The corresponding semivariogram function was $\gamma(h) = 10(1 - e^{-\frac{h}{3.33}})$.

$$\boldsymbol{W_1} = \boldsymbol{C_1^{-1}} \boldsymbol{c_1} = \begin{pmatrix} 10 & 5.103 & 0.435 & 0.199 & 0.489 & 0.259 & 0.048 & 61 & 139 & 1 \\ 5.103 & 10 & 0.362 & 0.202 & 0.905 & 0.489 & 0.061 & 63 & 140 & 1 \\ 0.435 & 0.362 & 10 & 2.902 & 0.199 & 0.111 & 0.362 & 64 & 129 & 1 \\ 0.199 & 0.202 & 2.902 & 10 & 0.244 & 0.152 & 1.222 & 68 & 128 & 1 \\ 0.489 & 0.905 & 0.199 & 0.244 & 10 & 5.103 & 0.224 & 71 & 140 & 1 \\ 0.259 & 0.489 & 0.111 & 0.152 & 75.103 & 10 & 0.193 & 73 & 141 & 1 \\ 0.048 & 0.061 & 0.362 & 1.222 & 0.224 & 0.193 & 10 & 75 & 128 & 1 \\ 61 & 63 & 64 & 68 & 71 & 73 & 75 & 0 & 0 & 0 \\ 139 & 140 & 129 & 128 & 140 & 141 & 128 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}^{-1} \begin{pmatrix} 2.612 \\ 3.382 \\ 0.889 \\ 0.579 \\ 1.333 \\ 0.682 \\ 0.176 \\ 65 \\ 137 \\ 1 \end{pmatrix}.$$

The answer is:

$$\boldsymbol{W_1} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \\ w_7 \\ -\lambda_1 \\ -\lambda_2 \\ -\lambda_0 \end{pmatrix} = \begin{pmatrix} 0.220 \\ 0.342 \\ 0.147 \\ 0.073 \\ 0.151 \\ 0.040 \\ 0.027 \\ 0.067 \\ -0.023 \\ -2.300 \end{pmatrix}.$$

The last 3 elements of the $\boldsymbol{W_1}$ vector are the Lagrange multipliers, $\lambda_0 = 2.3, \lambda_1 = -0.067, \lambda_2 = 0.023$. We can verify that the sum of the elements 1 through 7 is equal to 1, as it should be.

The predicted value is equal to:

$$\hat{z}(s_0) = \sum_{i=1}^{7} w_i z(s_i) = 0.220(477) + \cdots + 0.027(783) = 567.54.$$

The variance of the error of prediction is equal to:

$$\sigma^2(s_0) = C(0) - \sum_{i=1}^{n} w_i C(s_0 - s_i) + \lambda_0 + \lambda_1 x_{10} + \cdots + \lambda_k x_{k0}$$

Or

$$\sigma^2(s_0) = 10 - [0.220(2.612) + \cdots + 0.027(0.176)] + 2.301 - 0.066(65) + 0.023(137) = 9.044.$$

Universal kriging using geoR

```
> a <- read.table("http://www.stat.ucla.edu/~nchristo/statistics_c173_c273/
       kriging_11.txt", header=TRUE)
> b <- as.geodata(a)
```

Using the `ksline` function:

```
>  univ_kr <- ksline(b, cov.model="exp", cov.pars=c(10,3.33), nugget=0,
   locations=c(65,137), m0="kt", trend=1)
ksline: kriging location:  1 out of 1
Kriging performed using global neighbourhood
Warning message:
assuming that there is only 1 prediction point in: .check.locations(locations)
```

And here is the estimate and its variance:

```
> univ_kr$predict
[1] 567.6581
> univ_kr$krige.var
[1] 9.04282
```

Note: The argument `trend=1` is required if the argument `m0="kt"` is used.

Using the `krige.conv` function: The `locations` argument in the `krige.conv` function must be a data frame. You can create it as `dd <- data.frame(65,137)`. Or simply use `data.frame(65,137)` as shown below:

```
> kc <- krige.conv(b, locations=dd,krige=krige.control(type.krige="ok",
        cov.model="exp", cov.pars=c(10, 3.33), nugget=0,
        trend.l="1st", trend.d="1st"))

kc <- krige.conv(b, locations=data.frame(65,137),krige=krige.control(type.krige="ok",
       cov.model="exp", cov.pars=c(10, 3.33), nugget=0,
       trend.l="1st", trend.d="1st"))
```

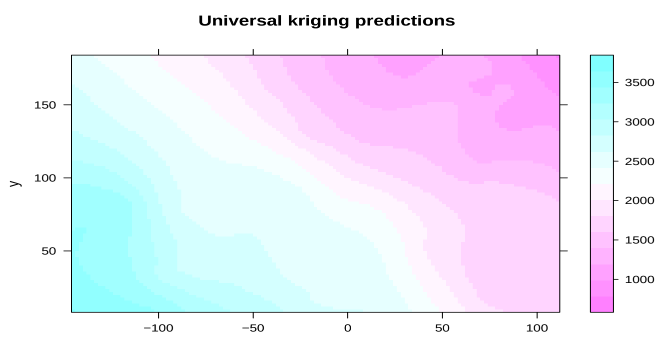
Note: The arguments `trend.l, trend.d` specify the trend of the locations to be predicted and the trend of the data. They must be the same.

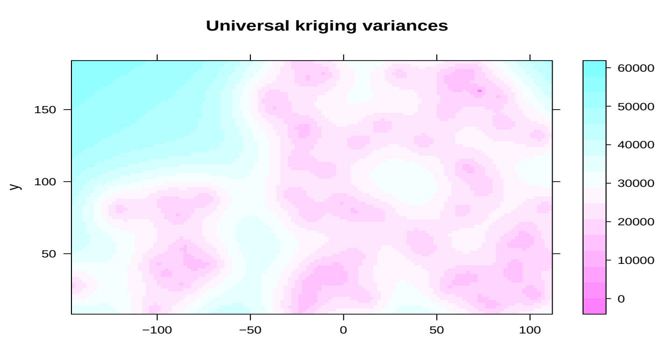
A complete example on how to use universal kriging in `geoR` and `gstat` is presented on the next page using the `Wolfcamp` data set.

```
#Read the data:
site="http://www.stat.ucla.edu/~nchristo/statistics_c173_c273/wolfcamp.txt"
a <- read.table(file=site, header=TRUE)

#Load the geoR and gstat packages:
library(geoR)
library(gstat)

#Convert the data into a geodata object (for geoR):
b <- as.geodata(a)

#Create the grid for spatial predictions:
x.range <- as.integer(range(a[,1]))
y.range <- as.integer(range(a[,2]))
grd <- expand.grid(x=seq(from=x.range[1], to=x.range[2], by=2),
y=seq(from=y.range[1], to=y.range[2], by=2))

#Perform universal kriging (geoR using the function ksline):
krig_trend <- ksline(b, cov.model="sph", cov.pars=c(30000, 60),
              nugget=10000, m0="kt", trend=1, locations=grd)

#Perform universal kriging (geoR using the function krige.conv):
kc <- krige.conv(b, locations=grd,krige=krige.control(type.krige="ok",
      cov.model="sph", cov.pars=c(30000, 60), nugget=10000,
      trend.l="1st", trend.d="1st"))

#Perform universal kriging (gstat - krige):
pr_univk <- krige(id="level", level~x+y, locations=~x+y,
model=vgm(30000, "Sph", 60, 10000),
data=a, newdata=grd)


#Compare the three methods:
pred <- cbind(pr_univk$level.pred, krig_trend$pred, kc$predict)
pred.var <- cbind(pr_univk$level.var, krig_trend$krige.var, kc$krige.var)

#Display the first 5 rows:
> pred[1:5,]
         [,1]     [,2]     [,3]
[1,] 3645.489 3645.489 3645.489
[2,] 3636.049 3636.049 3636.049
[3,] 3626.797 3626.797 3626.797
[4,] 3617.744 3617.744 3617.744
[5,] 3608.875 3608.875 3608.875
```

```
> pred.var[1:5,]
         [,1]     [,2]     [,3]
[1,] 36964.07 36964.07 36964.07
[2,] 36693.89 36693.89 36693.89
[3,] 36515.17 36515.17 36515.17
[4,] 36405.39 36405.39 36405.39
[5,] 36335.47 36335.47 36335.47


#Load the lattice package:
library(lattice)

#Construct a raster map using the kriged values:
levelplot(pr_univk$level.pred~x+y, pr_univk, aspect ="iso",
main="Universal kriging predictions")

#Construct a raster map using the variances of the kriged values:
levelplot(pr_univk$level.var~x+y, pr_univk, aspect ="iso",
main="Universal kriging variances")
```

Here are the plots: ure





Using the `image` function:

```
#Collapse the predicted values into a matrix:
qqq <- matrix(pr_univk$level.pred, length(seq(from=x.range[1], to=x.range[2], by=2)),
length(seq(from=y.range[1], to=y.range[2], by=2)))

#Construct the raster map of the predicted values:
image(seq(from=x.range[1], to=x.range[2], by=2),
seq(from=y.range[1],to=y.range[2], by=2), qqq,
xlab="West to East",ylab="South to North", main="Raster map of the predicted values")

#Collapse the variances into a matrix:
qqq1 <- matrix(pr_univk$level.var, length(seq(from=x.range[1], to=x.range[2], by=2)),
length(seq(from=y.range[1], to=y.range[2], by=2)))

#Construct the raster map of the variances:
image(seq(from=x.range[1], to=x.range[2], by=2),
seq(from=y.range[1],to=y.range[2], by=2), qqq1,
xlab="West to East",ylab="South to North", main="Raster map of the variances")
```

Here are the plots: