



Universidad Nacional de Colombia

FACULTAD DE CIENCIAS

PARCIAL 1

Estadística Espacial

Integrantes:

John Anderson Guarín López
German Camilo Vasquez Herrera

10 de abril de 2024

Punto 1

Enunciado:

Sea:

$$\mathbf{Y}(\mathbf{s}) = \begin{pmatrix} Y(s_1) \\ \vdots \\ Y(Y_{100}) \end{pmatrix} \sim \text{NMV}(\mu(s), \Sigma), \text{ con}$$
$$\mu(s_i) = \beta_0 + \beta_1 x_i + \beta_2 y_i, s_i = (x_i, y_i)$$
$$\Sigma = C(h) = \sigma^2 \exp\left(\frac{-h}{\phi}\right)$$

NMV: Normal Multivariado. Asigne valores a β_0 , β_1 , β_2 , σ^2 y ϕ .

- Plantee el predictor kriging universal.
- ¿Cuál es (en términos matriciales) el sistema de ecuaciones correspondiente (usando las funciones de semivariograma y de covarianza)? ¿Cuál es la expresión correspondiente a la varianza de predicción?
- Ejercicio de computo en R: Genere un enmallado regular de 100 nodos y usando la función `dist` de R encuentre la matriz de distancias y calcule $C(h)$, siendo h la distancia Euclidiana entre cada par de nodos. Simule una muestra de $\mathbf{Y}(\mathbf{s})$ empleando el método de la descomposición de Cholesky, estime un semivariograma, ajuste un modelo (estime los parámetros) y obtenga el mapa de predicciones y de las varianzas de predicción.

Respuesta:

- Plantee el predictor kriging universal.

De acuerdo al modelo planteado en el enunciado, el predictor Kriging Universal, se define de la siguiente forma:

$$\hat{Y}(s_0) = \sum_{i=1}^n \lambda_i Y(s_i)$$

Donde λ_i corresponde a los pesos en el kriging universal, de acuerdo a la variable $Y(s_i)$.

- ¿Cuál es (en términos matriciales) el sistema de ecuaciones correspondiente (usando las funciones de semivariograma y de covarianza)? ¿Cuál es la expresión correspondiente a la varianza de predicción?

Para determinar el sistema de ecuaciones correspondientes, basta con minimizar la varianza del error de predicción del kriging universal ($V(Y^*(s_0) - Y(s_0))$). Por lo tanto, tenemos lo siguiente:

$$\begin{aligned}
V(Y^*(s_0) - Y(s_0)) &= E(Y^*(s_0) - Y(s_0))^2 \\
&= E \left(\left(\sum_{i=1}^n \lambda_i (\mu(s_i) - \epsilon(s_i)) \right) - (\mu(s_0) - \epsilon(s_0)) \right)^2 \\
&= E \left[\left(\sum_{i=1}^n \lambda_i (\mu(s_i) - \mu(s_0)) \right) + \left(\sum_{i=1}^n \lambda_i (\epsilon(s_i) - \epsilon(s_0)) \right) \right]^2 \\
&= E \left[\left(\sum_{i=1}^n \lambda_i (\epsilon(s_i) - \epsilon(s_0)) \right)^2 \right] \\
&= \sum_{i=1}^n \sum_{j=1}^n E(\epsilon(s_i) \epsilon(s_j)) - 2 \sum_{j=1}^n E(\epsilon(s_i) \epsilon(s_0)) + E(\epsilon(s_0))^2
\end{aligned}$$

Ahora bien, si usamos $C_{ij} = COV(\epsilon(s_i), \epsilon(s_j))$ y $\sigma^2 = E(\epsilon(s_0))^2$, se tiene lo siguiente:

$$V(Y^*(s_0) - Y(s_0)) = \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j C_{ij} - 2 \sum_{i=1}^n \lambda_i C_{i0} + \sigma^2$$

Luego, si se incluye la restricción dada por la condición de insesgamiento, se debe minimizar lo siguiente:

$$\sigma_{ku}^2 = \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j C_{ij} - 2 \sum_{i=1}^n \lambda_i C_{i0} + \sigma^2 \sum_{l=1}^P \mu_l \left[\sum_{i=1}^n \lambda_i f_l(s_i) - f(s_0) \right]$$

En términos de la función de semivarianza, tenemos lo siguiente:

$$\sigma_{ku}^2 = \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j \gamma_{ij} - 2 \sum_{i=1}^n \lambda_i \gamma_{i0} + \sigma^2 \sum_{l=1}^P \mu_l \left[\sum_{i=1}^n \lambda_i f_l(s_i) - f(s_0) \right]$$

Si derivamos la expresión anterior, respecto a $\lambda_1, \lambda_2, \dots, \lambda_{100}, \mu_1, \mu_2, \dots, \mu_{100}$ e igualando a 0, se igualan las expresiones a γ_{i0} y a f_{l0} , las cuales se pueden expresar de forma matricial, de la siguiente forma:

$$\begin{pmatrix}
\gamma_{11} & \dots & \gamma_{1100} & f_{11} & \dots & f_{p1} \\
\gamma_{21} & \dots & \gamma_{2100} & f_{12} & \dots & f_{p2} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\gamma_{1001} & \dots & \gamma_{100100} & f_{100} & \dots & f_{p100} \\
f_{11} & \dots & f_{1100} & 0 & \dots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
f_{p1} & \dots & f_{p100} & 0 & \dots & 0
\end{pmatrix}
\begin{pmatrix}
\lambda_1 \\
\lambda_2 \\
\vdots \\
\lambda_{100} \\
\mu_1 \\
\vdots \\
\mu_{100}
\end{pmatrix}
=
\begin{pmatrix}
\gamma_{10} \\
\gamma_{20} \\
\vdots \\
\gamma_{1000} \\
f_{10} \\
\vdots \\
f_{p0}
\end{pmatrix}$$

Finalmente, la expresión correspondiente a la varianza de la predicción, queda de la siguiente forma:

$$\sigma_{ku}^2 = \sum_{i=1}^n \lambda_i \gamma_{i0} + \sum_{l=1}^P \mu_l f_l(s_0)$$

- Ejercicio de computo en R: Genere un enmallado regular de 100 nodos y usando la función `dist` de R encuentre la matriz de distancias y calcule $C(h)$, siendo h la distancia Euclidiana entre cada par de nodos. Simule una muestra de $\mathbf{Y}(\mathbf{s})$ empleando el método de la descomposición de Cholesky, estime un semivariograma, ajuste un modelo (estime los parámetros) y obtenga el mapa de predicciones y de las varianzas de predicción.

A continuación se deja expresado el código en R, separándolo por partes, de acuerdo a lo solicitado en el enunciado. Primero se deja el código con las librerías utilizadas para la solución del punto.

```
# Librerías Utilizadas ####

rm(list=ls())
if (!require(geoR)){install.packages("geoR");library(geoR)}
if (!require(gstat)){install.packages("gstat");library(gstat)}
if (!require(MASS)){install.packages("MASS");library(MASS)}
if (!require(scatterplot3d)){install.packages("scatterplot3d");
library(scatterplot3d)}
```

Ahora bien, este es el código que muestra la función de la tendencia y el gráfico respectivo de la misma.

```
# Punto 1 ####

## Grafico de la tendencia de acuerdo al modelo ####

beta_0 <- 4
beta_1 <- 5
beta_2 <- 2

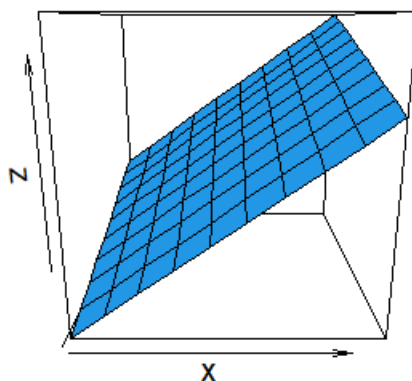
tendencia <- function(x, y){
  4 + 5*x + 2*y
}

## Preparar variables ####
x <- seq(0, 9, length = 10)
y <- seq(0, 9, length = 10)
z <- outer(x, y, tendencia)
persp(x, y, z, col=4, main = "Grafico de la Funcion de
  Tendencia")
```

El gráfico resultante es el siguiente:

Gráfico 1: Función de tendencia del modelo

Gráfico de la Función de Tendencia



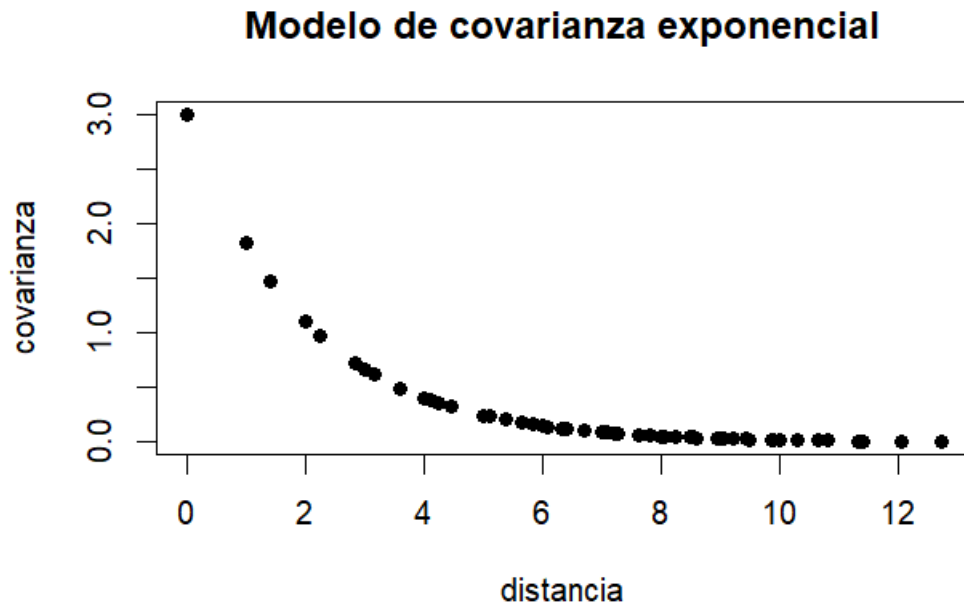
Fuente: Elaboración propia

Siguiendo con el código, a continuación se muestra el gráfico del modelo de covarianza exponencial:

```
## Definicion del modelo de covarianza ####  
  
grid<-expand.grid(seq(0,9),seq(0,9))  
plot(grid)  
distancia<-dist(grid,diag=T,upper=T)  
distancia<-as.matrix(distancia)  
sig2 <- 3  
phi <- 2  
covarianza<-sig2*exp((-distancia)/phi) # Modelo de covarianza  
exponencial  
covarianza<-as.matrix(covarianza)  
plot(distancia,covarianza,pch=16, main="Modelo de covarianza  
exponencial")
```

El gráfico resultante es el siguiente:

Gráfico 2: Modelo de Covarianza Exponencial



Fuente: Elaboración propia

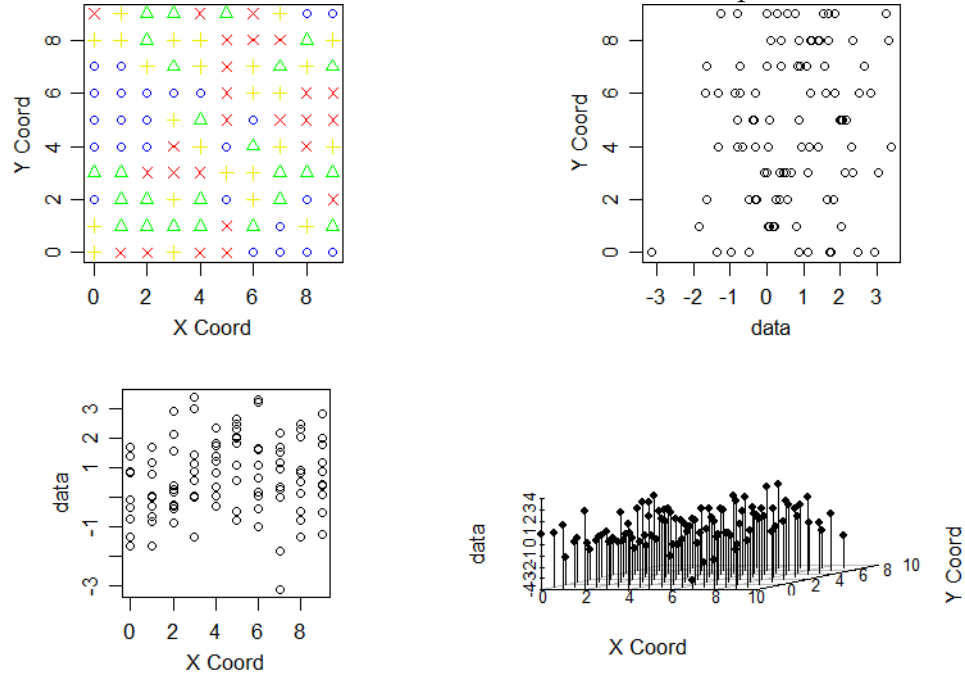
Luego, una vez obtenido el modelo de covarianza exponencial, se realiza la respectiva simulación, usando la descomposición de Cholesky. Por lo que tenemos lo siguiente:

```
## Simulacion del modelo exponencial, usando descomposicion de
  Cholesky #####

set.seed(123)
medias.ncte <- beta_0+beta_1*grid[,1]+beta_2*grid[,2]
covar_chol <- chol(covarianza)
normal_sample <- matrix(rnorm(100), nrow = 100)
normal.ncte<-covar_chol%*%normal_sample
dim(covarianza)
length (normal.ncte)
var(normal.ncte-medias.ncte)
datosncte<-cbind(grid[,1],grid[,2], normal.ncte)
datosncte<-as.geodata(datosncte, coords=1:2, var=3) # 100 datos
  (uno para cada coordenada)
plot(datosncte, scatter3d=TRUE)
```

El gráfico obtenido es el siguiente:

Gráfico 3: Simulación del Modelo de Covarianza Exponencial



Fuente: Elaboración propia

Una vez realizada la simulación del modelo de covarianza exponencial, se realiza la estimación del variograma de datos no estacionarios. Por lo que se procede a realizar el cálculo y la estimación del modelo de variograma con los residuales, obteniendo lo siguiente:

```
## Estimacion del variograma de datos no estacionarios ####

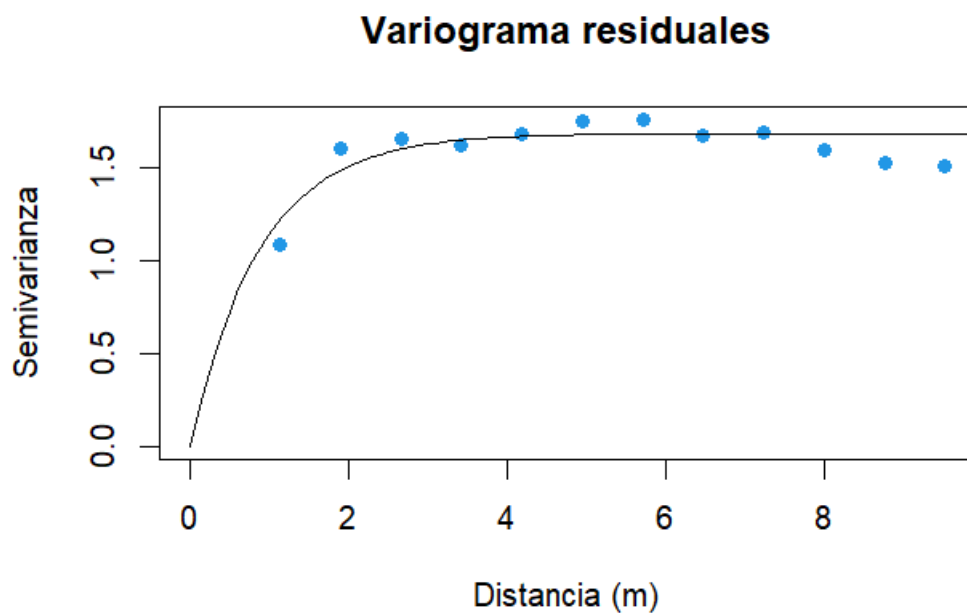
### Calculo de los residuales ###
modelo<-lm(normal.ncte~grid[,1]+grid[,2])
modelo
beta0_est<-modelo$coefficients[[1]]
beta1_est<-modelo$coefficients[[2]]
beta2_est<-modelo$coefficients[[3]]
residuales<-normal.ncte-(beta0_est+beta1_est*grid[,1]+
beta2_est*grid[,2])
residuales_coord<-as.matrix(cbind(x_coord=grid[,1],
y_coord=grid[,2], residuales))
residuales<-as.geodata(residuales_coord, coords=1:2, var=3)

## Estimacion del modelo de variograma (con los residuales) ####
variog_residuales <- variog(residuales, option = "bin",
max.dist = 10)
plot(variog_residuales)
ini.vals <- expand.grid(seq(2,5,l=5), seq(5,10,l=5))
modelo_residuales <- variofit(variog_residuales, ini=ini.vals,
```

```
fix.nug=FALSE, wei="npair", min="optim")
plot(variog_residuales, xlab="Distancia
(m)", ylab="Semivarianza", main = "Variograma residuales",
pch=16, col=4)
lines(modelo_residuales)
```

El gráfico resultante es el siguiente:

Gráfico 4: Variograma de los residuales



Fuente: Elaboración propia

Después se realiza la predicción por kriging universal, donde primero definimos el enmallado de predicción, el cual queda de la siguiente manera:

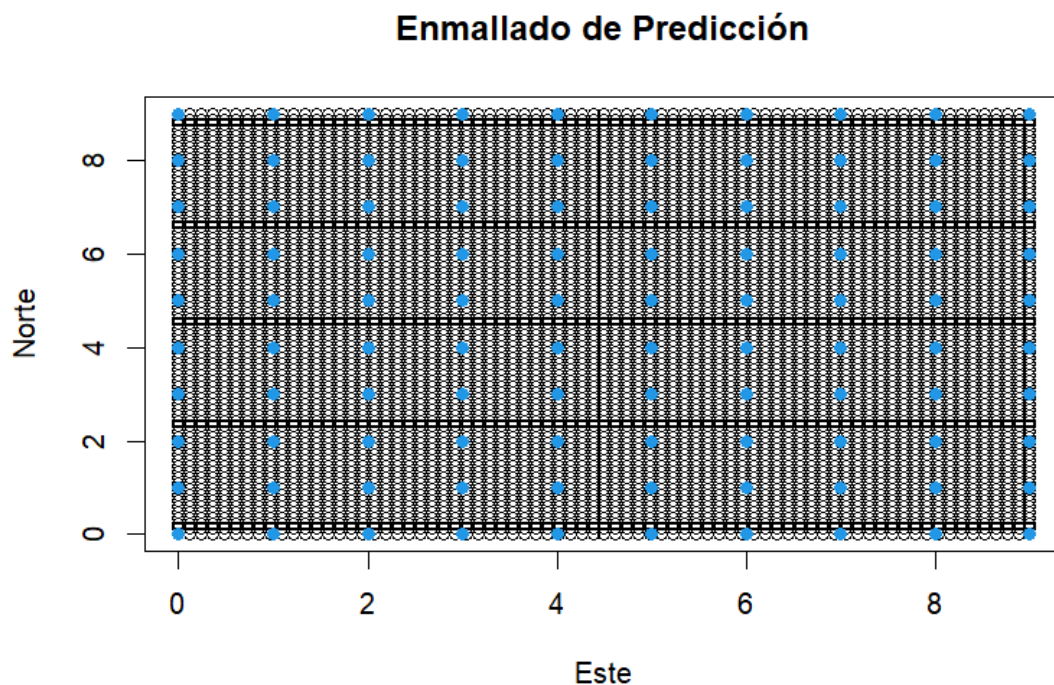
```
## Hacer prediccion por kriging universal ####

### Definir el enmallado de prediccion ####

x<-c(0,9,9,0,0)
y<-c(0,0,9,9,0)
datoscte.borde<-cbind(x,y)
plot(datoscte.borde)
datosncte.grid<-expand.grid(Este=seq(0,9,1=75),
  Norte=seq(0,9,1=75))
plot(datosncte.grid, main = "Enmallado de Prediccion")
points(grid, col=4, lwd=4, pch=16)
```


El enmallado de predicción queda de la siguiente forma:

Gráfico 5: Enmallado de Predicción Kriging Universal



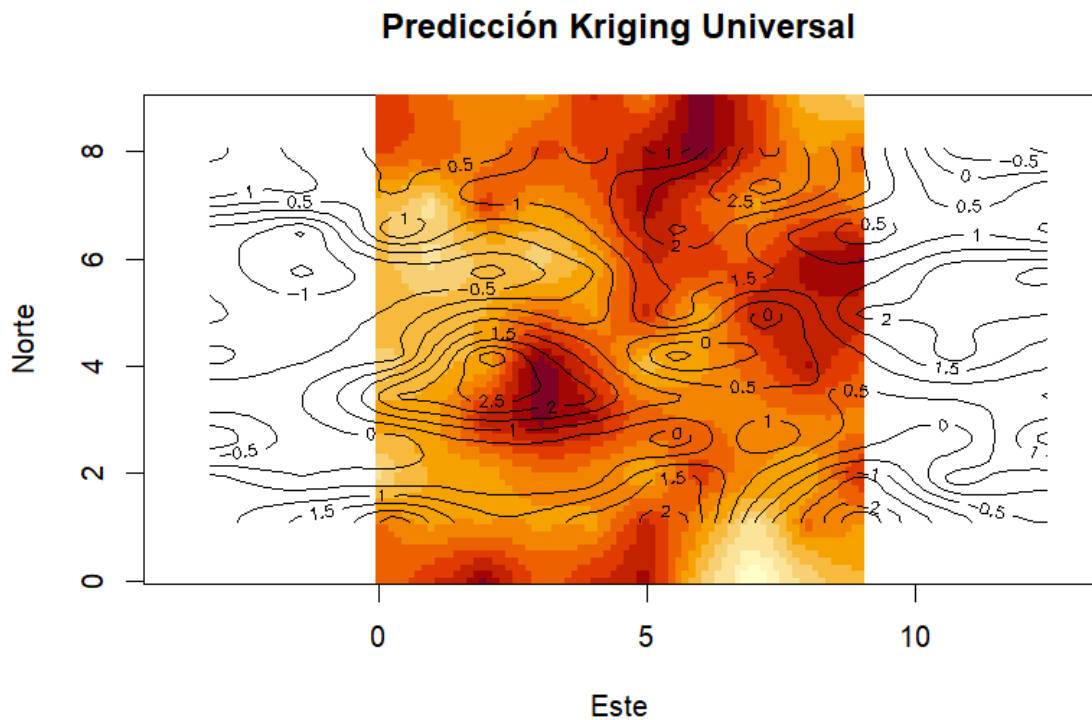
Fuente: Elaboración propia

Luego se realiza el gráfico de Kriging Universal, mostrando el mapa de calor con las predicciones obtenidas, por lo que tenemos lo siguiente:

```
### Kriging Universal ###  
  
names(modelo_residuales)  
Sigsq_est<-modelo_residuales$cov.pars[1]  
phi_est<-modelo_residuales$cov.pars[2]  
modelo_residuales  
datosncte.kc<-krige.conv(datosncte, loc=datosncte.grid, krige=  
  krige.control(nugget=0,trend.d="1st", trend.l="1st",  
               cov.pars=c(Sigsq=Sigsq_est, phi_est=  
  phi_est)))  
image(datosncte.kc, main="Prediccion Kriging Universal",  
      xlab="Este", ylab="Norte")  
contour(datosncte.kc,main="", add=TRUE, drawlabels=TRUE)
```

El gráfico obtenido es el siguiente:

Gráfico 6: Predicción Kriging Universal



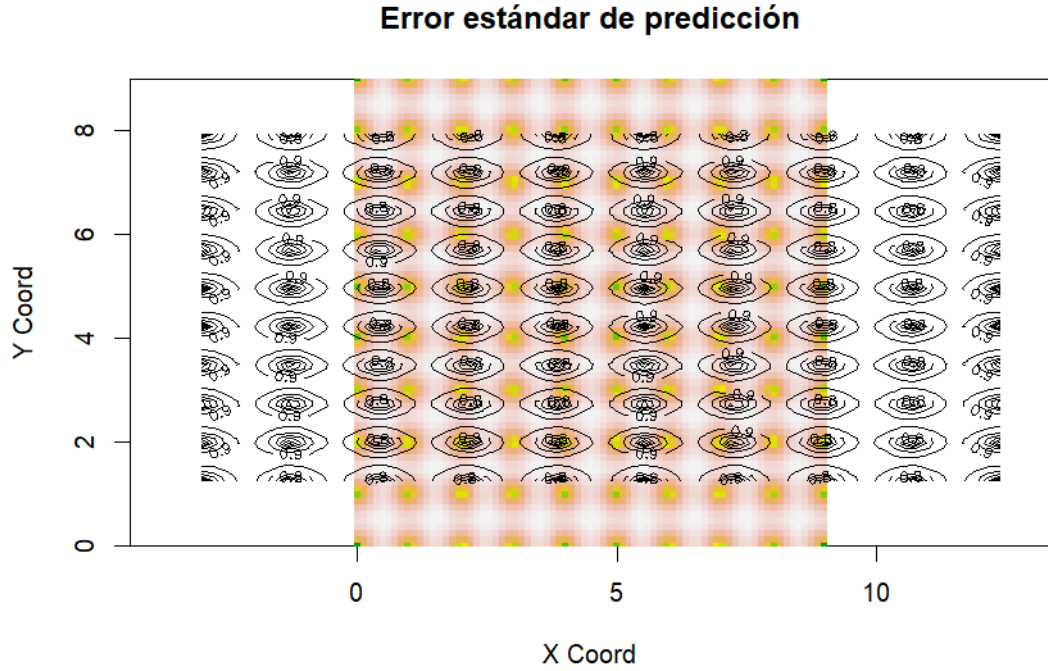
Fuente: Elaboración propia

Finalmente, una vez obtenido el gráfico de la predicción del kriging universal, se elabora el gráfico del error estándar de predicción, el cual queda de la siguiente forma:

```
### Error estandar de prediccion ####  
  
image(datosncte.kc, xlim=c(0,9),  
      ylim=c(0,9),val=sqrt(datosncte.kc$krige.var), main="Error  
      estandar de prediccion", col=terrain.colors(100))  
contour(datosncte.kc, val=sqrt(datosncte.kc$krige.var),  
        main="", add=TRUE, drawlabels=TRUE)  
summary(sqrt(datosncte.kc$krige.var))
```

El código muestra el siguiente gráfico:

Gráfico 7: Error estándar de predicción



Fuente: Elaboración propia

Punto 2

Enunciado:

Demuestre que el predictor kriging es óptimo bajo normalidad multivariada (asumiendo pérdida cuadrática).

Respuesta:

Para esta demostración es necesario el uso del siguiente teorema:

Sea $\mathbf{Y}=(Y_1, \dots, Y_n)$ un vector compuesto de n variables aleatorias. Sea Y_0 una variable aleatoria y $\hat{Y}_0 = \omega(Y)$ una función de Y tal que $\phi : \mathbb{R} \longrightarrow \mathbb{R}$. Entonces, $E(Y_0 - \hat{Y}_0)$ es mínimo cuando $\hat{Y}_0 = E(Y_0|Y)$.

Luego, sea $\{Y(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^2\}$ una colección de variables aleatorias, asumiendo normalidad tenemos

$$\mathbf{Y} = \begin{pmatrix} Y(x_0) \\ Y(x_1) \\ \vdots \\ Y(x_n) \end{pmatrix} \sim N \left(\begin{pmatrix} \mu \\ \mu \\ \vdots \\ \mu \end{pmatrix}, \begin{pmatrix} \sigma^2 & \mathbf{c}^T \\ \mathbf{c} & \Sigma \end{pmatrix} \right)$$

con $\mathbf{c} = \text{Cov}(Y, Y(x_0)) = (\sigma_{10}, \dots, \sigma_{n0})^T$ y

$$\Sigma = \begin{pmatrix} \sigma^2 & \sigma_{12} & \dots & \sigma_{1n} \\ \sigma_{21} & \sigma^2 & \dots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \dots & \sigma^2 \end{pmatrix}$$

Entonces por propiedades de la distribución normal multivariada tenemos que:

$$\mathbf{Y} = \begin{pmatrix} \mathbf{Y}^{(2)} \\ \mathbf{Y}^{(2)} \end{pmatrix} \sim N \left(\begin{pmatrix} \mu^{(1)} \\ \mu^{(2)} \end{pmatrix}, \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} \right)$$

por lo tanto $\mathbf{Y}^{(1)} | \mathbf{Y}^{(2)} \sim N(\mu^{(1)} + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{Y}^{(2)} - \mu^{(2)}), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})$.

Así, teniendo en cuenta lo anterior y la distribución de \mathbf{Y} tenemos que:

$$Y(x_0) | \mathbf{Y} \sim N(\mu + \mathbf{c}^T \Sigma^{-1}(\mathbf{Y} - \mu), \sigma^2 - \mathbf{c}^T \Sigma^{-1} \mathbf{c})$$

entonces, el mejor predictor para el modelo normal teniendo en cuenta la función de pérdida cuadrática es

$$\begin{aligned} \hat{Y}(x_0) &= E(Y(x_0) | \mathbf{Y}) = \mu + \mathbf{c}^T \Sigma^{-1}(\mathbf{Y} - \mu) \\ &= \mu + \lambda^T (\mathbf{Y} - \mu) \\ &= \mu + \sum_{i=1}^n \lambda_i (Y(x_i) - \mu) \end{aligned}$$

El cual corresponde al predictor del Kriging.

Punto 3

Enunciado:

Demuestre que la varianza del predictor kriging es igual a la varianza condicional bajo normalidad.

Respuesta:

Tomando el resultado del **Punto 2**:

$$Y(x_0) | \mathbf{Y} \sim N(\mu + \mathbf{c}^T \Sigma^{-1}(\mathbf{Y} - \mu), \sigma^2 - \mathbf{c}^T \Sigma^{-1} \mathbf{c})$$

en otras palabras, la varianza del mejor predictor para el modelo normal con la función de pérdida cuadrática es simplemente la misma varianza condicional bajo normalidad, la cual es:

$$V(Y(x_0) | \mathbf{Y}) = \sigma^2 - \mathbf{c}^T \Sigma^{-1} \mathbf{c}$$

Y como observamos esta varianza coincide con la del predictor kriging porque con $\lambda^T = \mathbf{c}^T \Sigma^{-1}$, tenemos que:

$$\begin{aligned}\sigma_k^2 &= \lambda^T \Sigma \lambda - 2\mathbf{c}^T \lambda + \sigma^2 \\ &= \mathbf{c}^T \Sigma^{-1} \Sigma \Sigma^{-1} \mathbf{c} - 2\mathbf{c}^T \Sigma^{-1} \mathbf{c} + \sigma^2 \\ &= \sigma^2 - \mathbf{c}^T \Sigma^{-1} \mathbf{c}\end{aligned}$$

Punto 4

Enunciado:

Considere Y_1, \dots, Y_n , variables aleatorias Gaussianas con media μ , varianza σ^2 , y $Cov[Y_i, Y_j] = \sigma^2 \rho$, $\forall i \neq j$. ¿Es $S^2 = (n-1)^{-1} \sum_{i=1}^n (Y_i - \bar{Y})^2$ un estimador insesgado de σ^2 ? Haga la comprobación matemáticamente y mediante un ejercicio de simulación en R.

Respuesta:

Para determinar si S^2 es insesgado, basta con probar que $E[S^2] - \sigma^2 = 0$. Por lo tanto, tenemos lo siguiente:

$$\begin{aligned}
E[S^2] &= E \left[\frac{\sum_{i=1}^n (Y_i - \bar{Y})^2}{n-1} \right] \\
&= \frac{E \left[\sum_{i=1}^n (Y_i - \bar{Y})^2 \right]}{n-1} \\
&= \frac{E \left[\sum_{i=1}^n (Y_i^2 - 2Y_i\bar{Y} + \bar{Y}^2) \right]}{n-1} \\
&= \frac{E \left[\sum_{i=1}^n Y_i^2 - \sum_{i=1}^n 2Y_i\bar{Y} + \sum_{i=1}^n \bar{Y}^2 \right]}{n-1} \\
&= \frac{E \left[\sum_{i=1}^n Y_i^2 - 2\bar{Y} \sum_{i=1}^n Y_i + n\bar{Y}^2 \right]}{n-1} \\
&= \frac{E \left[\sum_{i=1}^n Y_i^2 - 2\bar{Y}n\bar{Y} + n\bar{Y}^2 \right]}{n-1} \\
&= \frac{E \left[\sum_{i=1}^n Y_i^2 - 2n\bar{Y}^2 + n\bar{Y}^2 \right]}{n-1} \\
&= \frac{E \left[\sum_{i=1}^n Y_i^2 - n\bar{Y}^2 \right]}{n-1} \\
&= \frac{\sum_{i=1}^n E[Y_i^2] - E[n\bar{Y}^2]}{n-1} \\
&= \frac{\sum_{i=1}^n E[Y_i^2] - nE[\bar{Y}^2]}{n-1} \\
&= \frac{\sum_{i=1}^n (Var(Y_i) + E[Y_i]^2) - n(Var(\bar{Y}) + E[\bar{Y}]^2)}{n-1} \\
&= \frac{\sum_{i=1}^n Var(Y_i) + \sum_{i=1}^n E[Y_i]^2 - nVar(\bar{Y}) - nE[\bar{Y}]^2}{n-1} \\
&= \frac{n\sigma^2 + n\mu^2 - nVar(\bar{Y}) - n\mu^2}{n-1} \\
&= \left(\frac{n}{n-1} \right) (\sigma^2 - Var(\bar{Y})) \\
&= \left(\frac{n}{n-1} \right) \left(\sigma^2 - \frac{\sigma^2}{n} \{1 + (n-1)\rho\} \right) \\
&= \left(\frac{n}{n-1} \right) \left(\sigma^2 \left(1 - \frac{1 + (n-1)\rho}{n} \right) \right) \\
&= \left(\frac{n}{n-1} \right) \left(\sigma^2 \left(\frac{n - (1 + (n-1)\rho)}{n} \right) \right) \\
&= \frac{\sigma^2(n-1 - (n-1)\rho)}{n-1} \\
&= \frac{\sigma^2(n-1)(1-\rho)}{n-1} \\
&= \sigma^2(1-\rho)
\end{aligned}$$

Como $E[S^2] = \sigma^2(1-\rho)$, entonces tenemos que $E[S^2] - \sigma^2 \neq 0$. Por lo tanto, S^2 no es un estimador insesgado de σ^2 . Este estimador puede ser insesgado, si y solo si, $\rho = 0$.

Ahora bien, un ejemplo práctico del resultado obtenido, lo podemos observar mediante la siguiente simulación, donde se le asignan valores a n, μ, σ^2 y ρ . Para corroborar el sesgo de S^2 , el siguiente código muestra los siguientes casos: uno cuando $\rho = 0$ y el otro cuando $\rho \neq 0$. Por lo tanto, tenemos los siguientes resultados:

```
## Caso 1: rho=0 (Caso insesgado) ####

### Asignacion valores ###
n<-100
mu<-0
sig2<-1
rho<-0
m<-100

### Vector de medias simulacion ###
media<-rep(mu,n)

### Matriz de varianzas y covarianzas simulacion ###
Sig<-matrix(sig2*rho, nrow=n, ncol=n)
diag(Sig)<-sig2
var_mues<-c()
set.seed(17112000)
for(i in 1:m){
  datos<-mvrnorm(1,media,Sig)
  var_mues[i]<-var(datos)
}
mean(var_mues)
[1] 0.9923835
```

Tal y como se puede observar, con $\rho = 0$ obtenemos un S^2 muy cercano al valor asignado de σ^2 , en cual, para este ejercicio se le asignó el valor de 1.

Ahora bien, observemos lo que sucede en el caso 2, cuando $\rho \neq 0$.

```
## Caso 2: rho!=0 (Caso sesgado) ####

### Asignacion valores ###
n<-100
mu<-0
sig2<-1
rho<-0.5
m<-100

### Vector de medias simulacion ###
media<-rep(mu,n)
```

```

### Matriz de varianzas y covarianzas simulacion ###
Sig<-matrix(sig2*rho, nrow=n, ncol=n)
diag(Sig)<-sig2
var_mues<-c()
set.seed(17112000)
for(i in 1:m){
  datos<-mvrnorm(1,media,Sig)
  var_mues[i]<-var(datos)
}
mean(var_mues)
[1] 0.497272

```

Como se puede observar, con un valor de $\rho \neq 0$ se obtiene un valor de S^2 alejado del valor asignado de σ^2 . Por lo que se concluye que S^2 solo es insesgado cuando $\rho = 0$.