Day 2

Introduction to R

The R environment

A good introduction to R is in the website `https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf`.

- An effective data handling and storage facility.
- A suite of operators for calculations on arrays, in particular matrices.
- A large, coherent, integrated collection of intermediate tools for data analysis.
- Graphical facilities for data analysis and display either directly at the computer or on hard-copy.
- A well developed, simple and effective programming language (called 'S') which includes conditionals, loops, user defined recursive functions and input and output facilities.

R and Statistics

- The R environment has not been developed only for Statistics but the number of statistical packages developed is hugh.
- There are about 25 packages supplied with R (called “standard” and “recommended” packages).
- More packages are available through the CRAN family of Internet sites (via `https://CRAN.R-project.org`).
- R will give minimal output and store the results in an object for subsequent interrogation by further R functions.
- The most important command is `help()`. To get more information on any specifc named function, for example `solve`, the command is
  ```
  > help(solve)
  ```

Vector and Matrix Manipulations

- `> x <- c(10.4, 5.6, 3.1, 6.4, 21.7)`

  Creats avector with 5 elements.

- `> y <- c(x, 0, x))`

  Creates two replicates of `x` with a zero in the middle.

- $>$ v $<$- 2*x + x*x+ 1

  generates a new vector v of length 5 constructed by adding together, element by element the three terms. The length of `v` is 5.

- In addition we can use the functions `log`, `exp`, `sin`, `cos`, `tan`, `sqrt`, `mean sd`, `var`, `min`, `max`, `summary`, `quantile`, `median`, `hist`, `boxplot`, `sort`, `rank`, `plot`, `ts.plot`, etc.

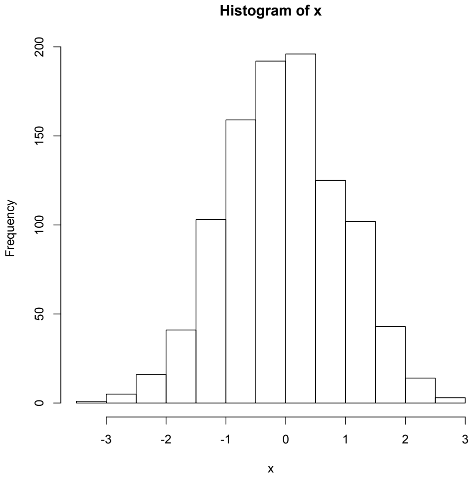
- To generate random vectors we can use the following functions: `runif`, `rnorm`, `rpois`, `rgamma`, `rt`, `rchisq`, `rcauchy`, `rweibull`, `rgeom`, etc.

Vector and Matrix Manipulations

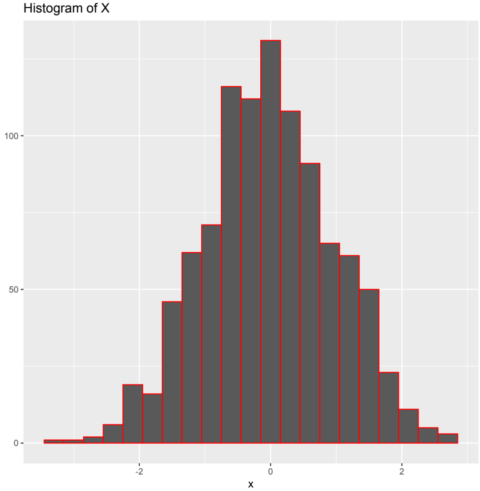
Example

```
library(ggplot2)
x=rnorm(1000,0,1)
summary(x)
sd(x)
hist(x)
theme_update(plot.title = element_text(hjust = 0.5))
qplot(x, geom="histogram", binwidth=
0.3,col=I("red"), main="Histogram of X",
xlim=c(-4,4))
boxplot(x,col="red",main="Boxplot of X")
x[43]=10
boxplot(x,col="red",main="Boxplot of X")
```

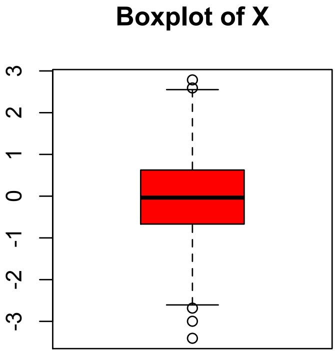
Simple Examples

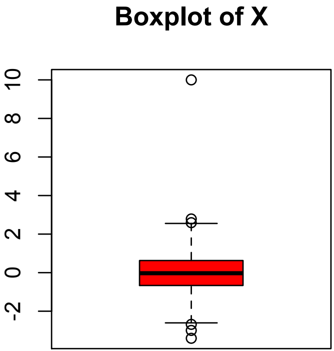


Simple Examples



Simple Examples





Reding Builtin Datsasets

- R contains many internal datasets. To access any of them just write the name. Some old packages need `data(name)`

  >data(AirPassengers)
  >AirPassengers

  R-Output
  AirPassengers

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1949	112	118	132	129	121	135	148	148	136	119	104	118
1950	115	126	141	135	125	149	170	170	158	133	114	140
⋮												
1959	360	342	406	396	420	472	548	559	463	407	362	405
1960	417	391	419	461	472	535	622	606	508	461	390	432

Reading External Files

- The simplest way to upload an external file is to use one of the functions `read.table()` or `scan()`.

  `x=read.table("location of the file") x=scan("location of the file")`

- For example, the following command will open the file beer.csv which is an excel file.

  ```
  beer<-read.table("/Users/ronnyvallejos/Documents/beer.csv")
  ```

  R-Output

  ```
  > beer
      V1
  1 93.2
  2 96.0
  .
  .
  .
  475 119.0
  476 153.0
  ```


UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

A Summary for Matrices

- A matrix is a bidimensional array that can be defined in R through the functions `matrix()` or `array()`

```
> x <- array(1:20, dim=c(4,5))
> x <- matrix(1:20, nrow=4)
```

R- Output

> x

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	1	5	9	13	17
[2,]	2	6	10	14	18
[3,]	3	7	11	15	19
[4,]	4	8	12	16	20

- `x=matrix(rnorm(100,0,1),ncol=10, nrow=10)`

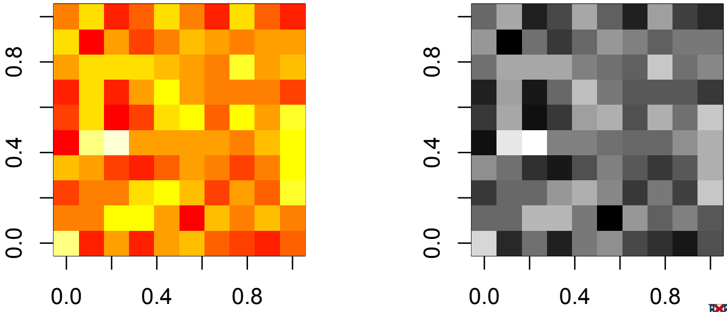
UNIVERSIDAD TECNICA FEDERICO SANTA MARIA

A Summary for Matrices

- A*B : Element-wise multiplication.
- A%*% B: Matrix multiplication.
- t(A): Transpose.
- diag(A): diagonal of a matrix.
- solve(A,B): Solve the linear equation $Ax = B$.
- x=eigen(A): x$val Eigenvalues of a matrix.
- x$vec Eigenvectors of a matrix.
- cbind(A,B): Combine matrices(vectors) horizontally. Returns a matrix.
- rbind(A,B): Combine matrices(vectors) vertically. Returns a matrix.
- R can handle images as matrices defined in a color scale.

Creating Images in R

```
x=matrix(rnorm(100,0,1),ncol=10, nrow=10)
par(mfrow=c(1,2), pty="s",oma=c(0.2,0.2,0.2,0.2))
image(x)
image(x, col = gray((0:32)/32))
```



Digital Images

- We let $\mathbb{R}_+$ denote the nonnegative real line, and we let $\mathbb{R}_+^N$ denote the set of $N-$dimensional vectors with nonnegative components.

- An image is considered to be an element $\boldsymbol{x} \in \mathbb{R}_+^N$.

- Alternatively, assume that the finite set of gray intensities $\{X(i,j) : 1 \leq i \leq n, 1 \leq j \leq m\}$ can be arranged into an $n \times m$ matrix $\boldsymbol{X}$ such that the (i,j)th element $\boldsymbol{X}(i,j) = X(i,j)$, i.e., $\boldsymbol{X} \in \mathcal{M}_{n\times m}(\mathbb{R})$.

- The vectorization of $\boldsymbol{X}$ is $\boldsymbol{x} = \mathrm{vec}(\boldsymbol{X}) = (\boldsymbol{x}_1^\top, \ldots, \boldsymbol{x}_N^\top)^\top$.

UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

Images in R

- There is a variety of available routines and packages to analyze images in R.
- The package png has a routine to load images.
- For instance, to display the Rlogo_old.png image we can use the following R code:

```
> library(png)
> library(RCurl)
> url <- "http://srb2gv.mat.utfsm.cl/files/img/Rlogoold.png"
> img <- readPNG(getURLContent(url))
> d <- dim(img)
> rows <- 1:561
> cols <- 1:724
> img <- t(img[rev(rows),cols,1])
> image(img, col = gray((0:32)/32), xaxt = "n", yaxt = "n")
> box()
```

R Output

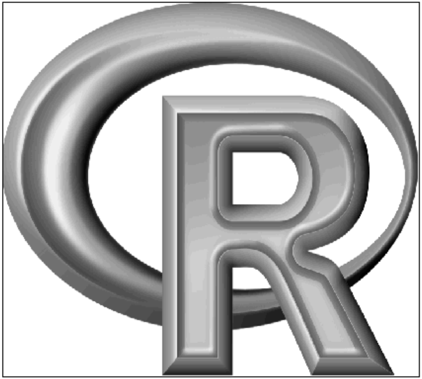


Figure: First band of the ‘old’ Rlogo image plotted in a gray scale.

Geostatistical Analysis with R

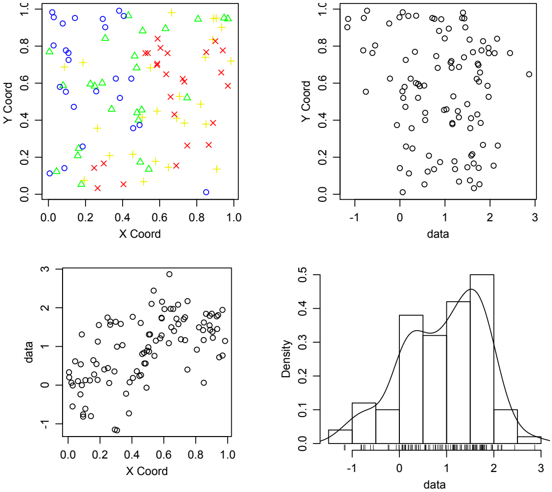
- There is a variety of programs and packages to perform geostatistical analysis.
- In this course all the analysis were developed in R, specifically the univariate geostatistical analysis was done using the R package `geoR`.
- To use a number of rotuines inside the package `geoR` the dataset needs to be a `geodata` object. To transform a matrix x to a geodata object we can use the command x<-as.geodata(x)
- The value of x will have:
  `coords` an $n \times 2$ matrix where n is the number of spatial locations. This can be access using x$coords.
  `data`: a vector of length n, for the univariate case. This can be access using x$data.

Geostatistical Analysis with R

- The dataset `s100` is an internal geostatistical set of data consisting of 100 observations, generated from a multivariate normal distribution with exponential covariance. To access this data set we just type `s100`.
- A preliminary statistical analysis can be carried out using the command `plot`. The display show four graphs with different patters that can be useful in practice.
- We just use the code

```
>s100
>plot(s100)
```

Geostatistical Analysis with R



UNIVERSIDAD TECNICA FEDERICO SANTA MARIA

Geostatistical Analysis with R

- Kriging estimates are based on normality of the spatial vector. From the previous Figure we have reasons to believe that the dataset is not normal.
- The Box-Cox transformation can be used to transform the original data. The function is

$$Y = \begin{cases} \log(X + \lambda_2), & \lambda = 0, \\ \frac{(X+\lambda_2)^\lambda}{\lambda - 1}, & \text{otherwise} \end{cases}$$

- For the s100 dataset we used the code

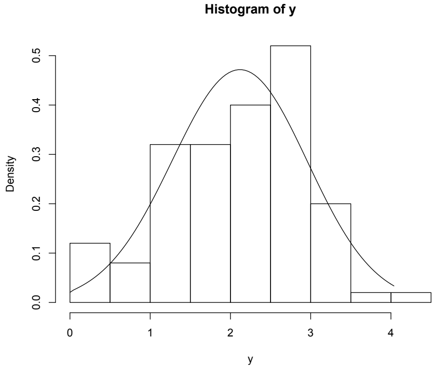
```
min(s100$data) # the minimum value is -1.167695
aux=boxcoxfit(s100$data+1.17) #this computes the lambda values
plot(x) # shows the new histogram of the transform variable
```


UNIVERSIDAD TECNICA FEDERICO SANTA MARIA

Geostatistical Analysis with R

R-Output Fitted parameters:

lambda	beta	sigmasq
1.0329436	1.1229064	0.7513006

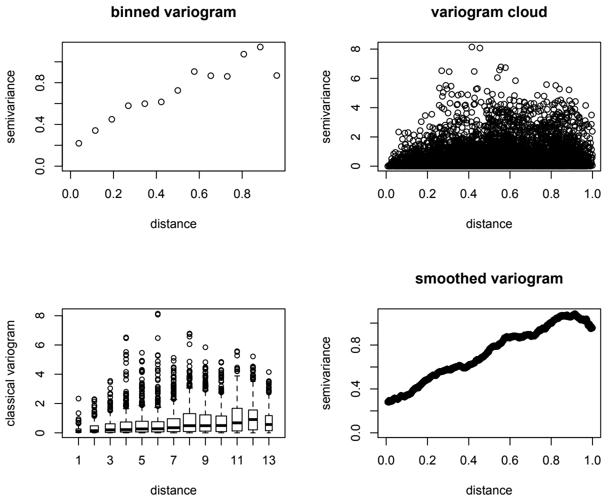


Geostatistical Analysis with R

Computing the Empirical Variogram

```
# binned variogram
vario.b <- variog(s100, max.dist=1)
# variogram cloud
vario.c <- variog(s100, max.dist=1, op="cloud")
#binned variogram and stores the cloud
vario.bc <- variog(s100, max.dist=1, bin.cloud=TRUE)
# smoothed variogram
vario.s <- variog(s100, max.dist=1, op="sm", band=0.2)
```

Geostatistical Analysis with R

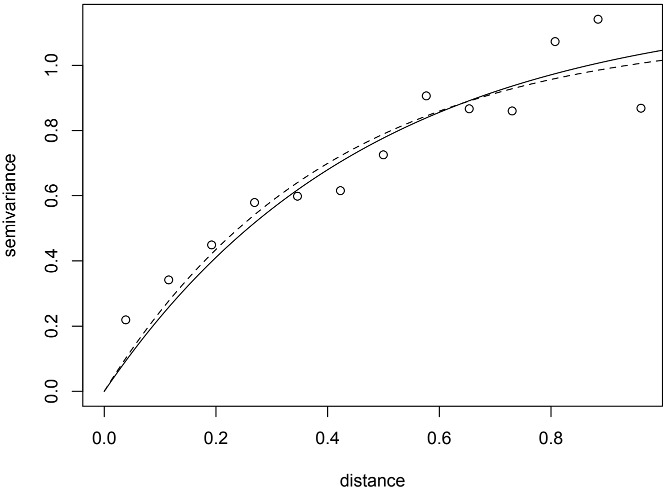


Geostatistical Analysis with R

Fitting a Variogram Model

```
vario100 <- variog(s100, max.dist=1)
ini.vals <- expand.grid(seq(0,1,l=5), seq(0,1,l=5))
ols <- variofit(vario100, ini=ini.vals, fix.nug=TRUE, wei="equal")
summary(ols)
wls <- variofit(vario100, ini=ini.vals, fix.nug=TRUE)
summary(wls)
plot(vario100)
lines(wls)
lines(ols, lty=2)
```

Geostatistical Analysis with R



Geostatistical Analysis with R

R-Output

variofit (OLS):

```
            sigmasq phi tausq kappa
initial.value "1" "0.25" "0" "0.5"
status        "est" "est" "fix" "fix"
loss value: 0.167887026209413
```

variofit(WLS):

```
      sigmasq phi tausq kappa
initial.value "1" "0.25" "0" "0.5"
status         "est" "est" "fix" "fix"
loss value: 68.37535548371
```

The loss functiobn is

$$\mathsf{LOSS}(\boldsymbol{\theta}) = \sum_k (\widehat{\gamma}_k - \gamma_k(\boldsymbol{\theta}))^2$$

Spatial Prediction (Kriging)

- Given the sample $(X(\boldsymbol{s}_1), \dots, X(\boldsymbol{s}_n))^\top$, our goal is to prefict the variable of interest in a new location called $\boldsymbol{s}_0$ in which the process has not been observed. That is, to provide a value for $X(\boldsymbol{s}_0)$.

- The spatial prediction has the advantage that the uncertainty of the estimate can be computed. This measure provides information about the variability of the estimate.

- The method is called Kriging in honor to Daniel Krige (1929-2013), an enginner from South Africa who studied the problem of predicting a spatial process on the plane.

- The mathematical tools were developed later by Matheron (1930-2000), a french mathematician.

Spatial Prediction (Kriging)

Theorem (Best Predictor)

Let $\boldsymbol{U}$ be a random vector and Y a random variable with finite second moment. Then one of the two conditions hold:

1. For any function g, $\mathbb{E}[(Y - g(\boldsymbol{U}))^2] = \infty$.
2. $\mathbb{E}[(Y - \mathbb{E}[Y|\boldsymbol{U}])^2] \leq \mathbb{E}[Y - g(\boldsymbol{U})^2]$, for all g. The equality is granted if $g(\boldsymbol{U}) = \mathbb{E}[Y|\boldsymbol{U}]$.

Notation:

- $\boldsymbol{X}(\boldsymbol{s}) = (X(\boldsymbol{s}_1), \ldots, X(\boldsymbol{s}_n))^\top$ available data.
- $X(\boldsymbol{s}_0)$ unknown value of X at $\boldsymbol{s}_0$.
- $p(\boldsymbol{X}, \boldsymbol{s}_0)$ predictor of X at s_0.
- $L(X(\boldsymbol{s}_0), p(\boldsymbol{X}, \boldsymbol{s}_0)) = (X(\boldsymbol{s}_0) - p(\boldsymbol{X}, \boldsymbol{s}_0))^2$ is called the loss function.

Spatial Prediction (Kriging)

- $\mathbb{E}[L] = \mathbb{E}[(X(\boldsymbol{s}_0) - p(\boldsymbol{X}, \boldsymbol{s}_0))^2]$ is called risk function.
- We cosnider the predictor

$$p(\boldsymbol{X}, \boldsymbol{s}_0) = \mathbb{E}[X(\boldsymbol{s}_0)|\boldsymbol{X}(\boldsymbol{s})].$$

- We have $\mathbb{E}[\mathbb{E}[X(\boldsymbol{s}_0)|\boldsymbol{X}(\boldsymbol{s})]] = \mathbb{E}[X(\boldsymbol{s}_0)]$.
- For a multivariate normal distribution, let $\boldsymbol{W} = (\boldsymbol{U} \ \boldsymbol{V})^\top$ be a partitioned vector . Then

$$\mathbb{E}[\boldsymbol{W}] = \mathbb{E}\begin{bmatrix} \boldsymbol{U} \\ \boldsymbol{V} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mu}_u \\ \boldsymbol{\mu}_v \end{bmatrix},$$

$$\text{var}[\boldsymbol{W}] = \begin{bmatrix} \boldsymbol{\Sigma}_u & \boldsymbol{\Sigma}_{uv} \\ \boldsymbol{\Sigma}_{vu} & \boldsymbol{\Sigma}_v \end{bmatrix},$$

$$\mathbb{E}[\boldsymbol{U}|\boldsymbol{V}] = \boldsymbol{\mu}_u + \boldsymbol{\Sigma}_{uv}\boldsymbol{\Sigma}_v^{-1}(\boldsymbol{V} - \boldsymbol{\mu}_v),$$

$$\text{var}[\boldsymbol{U}|\boldsymbol{V}] = \boldsymbol{\Sigma}_u - \boldsymbol{\Sigma}_{uv}\boldsymbol{\Sigma}_v^{-1}\boldsymbol{\Sigma}_{uv}^\top$$

Spatial Prediction (Kriging)

We define

- $\boldsymbol{U} = X(\boldsymbol{s}_0)$,
- $\boldsymbol{V} = \boldsymbol{X}(\boldsymbol{s})$,
- $\mathsf{var}[\boldsymbol{X}(\boldsymbol{s})] = \boldsymbol{\Sigma}$,
- $\mathbb{E}[X(\boldsymbol{s}_0)] = \mu(\boldsymbol{s}_0)$,
- $\mathsf{cov}[\boldsymbol{X}(\boldsymbol{s}), X(\boldsymbol{s}_0)] = \boldsymbol{\sigma}$,
- $\mathbb{E}[\boldsymbol{X}(s)] = \boldsymbol{\mu}(\boldsymbol{s})$.

Then the best predictor is:

$$\mathbb{E}[X(\boldsymbol{s}_0)|\boldsymbol{X}(\boldsymbol{s})] = \mu(\boldsymbol{s}_0) + \boldsymbol{\sigma}^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{X}(\boldsymbol{s}) - \boldsymbol{\mu}(\boldsymbol{s})).$$

In the Gaussian case

$$\mathbb{E}[(X(\boldsymbol{s}_0) - p(\boldsymbol{X}, \boldsymbol{s}_0))^2] = \mathsf{var}[X(\boldsymbol{s}_0)] - \boldsymbol{\sigma}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\sigma}$$

Simple Kriging

- Let X be a spatial process such that we observe $X(\boldsymbol{s}_1), \ldots, X(\boldsymbol{s}_n)$. We propose a predictor of the form

$$p(\boldsymbol{X}, \boldsymbol{s}_0) = \sum_{i=1}^{n} \lambda_i X(\boldsymbol{s}_i) + \lambda_0, \tag{1}$$

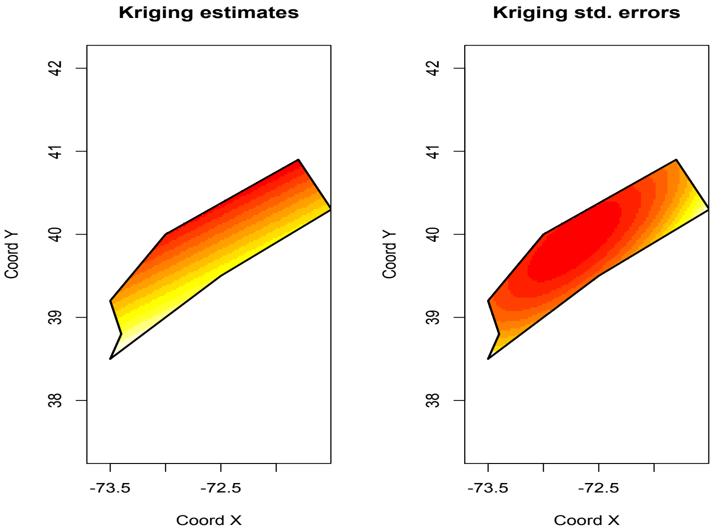
  where $\lambda_0, \ldots, \lambda_n$ are the unknown parameters.
- Suppose a linear model as

$$\boldsymbol{X}(\boldsymbol{s}) = \boldsymbol{\mu}(\boldsymbol{s}) + \boldsymbol{\epsilon}(\boldsymbol{s}),$$

  where $\boldsymbol{\epsilon}(\boldsymbol{s}) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$.
- Simple Kriging assume that $\boldsymbol{\mu}(\boldsymbol{s})$ and $\boldsymbol{\Sigma}$ are known.
- The goal is to find a predictor like (1) such that $\mathbb{E}[(X(\boldsymbol{s}_0) - p(\boldsymbol{X}, \boldsymbol{s}_0))^2]$ is minumum.

Simple Kriging

- We rewrite the predictor as
$$p(\boldsymbol{X}, \boldsymbol{s}_0) = \lambda_0 + \boldsymbol{\lambda}^\top \boldsymbol{X}(\boldsymbol{s}),$$
where $\lambda_0 \in \mathbb{R}$ and $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n)^\top$.
- $\mathbb{E}[(p(\boldsymbol{X}, \boldsymbol{s}_0) - X(\boldsymbol{s}_0))^2] = \sigma^2 + \boldsymbol{\lambda}^\top \boldsymbol{\Sigma} \boldsymbol{\lambda} - 2\boldsymbol{\sigma}^\top \boldsymbol{\lambda}$.
- $\boldsymbol{\lambda} = \text{argmin}_{\boldsymbol{\lambda}}(\sigma^2 + \boldsymbol{\lambda}^\top \boldsymbol{\Sigma} \boldsymbol{\lambda} - 2\boldsymbol{\sigma}^\top \boldsymbol{\lambda})$.
- $\boldsymbol{\lambda} = \boldsymbol{\Sigma}^{-1} \boldsymbol{\sigma}$.
- Kriging Predictor $p(\boldsymbol{X}, \boldsymbol{s}_0) = \mu(\boldsymbol{s}_0) + \boldsymbol{\sigma}^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{X}(\boldsymbol{s}) - \boldsymbol{\mu}(\boldsymbol{s}))$.
- Kriging Variance
$\sigma^2_{\mathsf{SK}} = \mathbb{E}[(p(\boldsymbol{X}, \boldsymbol{s}_0) - X(\boldsymbol{s}_0))^2] = \sigma^2 - \boldsymbol{\sigma}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\sigma}$.
- The kriging predictor and its variance match the best predictor and its variance.
- The variance of the Kriging predictor can also be plotted as a map.


Kriging estimates
Kriging std. errors
Coord Y
Coord X
42
41
40
39
38
-73.5
-72.5

R code: GeoR

```
summ=summary(scallops.geo)
linf=seq(summ$coor[1,1],summ$coor[2,1],l=101)
lsup=seq(summ$coor[1,2],summ$coor[2,2],l=101)
pred.grid<-expand.grid(linf,lsup)
vertices=rbind(c(-73.5,38.5),c(-73,39),c(-72.5,39.5),c(-71.5,40.3),
c(-71.8,40.9),c(-73,40),c(-73.5,39.2),c(-73.4,38.8))
polygon(vertices,angle=30,density=20)
krig=krige.control(trend.d = "1st", trend.l = "1st",
cov.pars=c(3.2,16.4),nugget=1.7, cov.model="wave")
kcver<-krige.conv(scallops.geo, loc=pred.grid, krige=krig,borders=vertices)
image(kcver, loc = pred.grid, xlab="Coord X", ylab="Coord Y",
main="Kriging estimates")
image(kcver, loc = pred.grid, xlab="Coord X", ylab="Coord Y",
val=sqrt(kcver$krige.var), main="Kriging std. errors")
```

Extensions

- Ordinary Kriging $\boldsymbol{\mu}(\boldsymbol{s})$ is unknown but constant, $\boldsymbol{\Sigma}$ is unknown.
- Universal Kriging $\boldsymbol{\mu}(\boldsymbol{s})$ is a specific trend not constant, $\boldsymbol{\Sigma}$ is unknown.
- Block Kriging Change of support. The goal is to predict on a continuous domain.
- Transgaussian Kriging If the process is not Gaussian sometimes it is possible to apply a transformation to the Gaussian predictor.
- Indicator Kriging Prediction for binary variables on the space.

The Cross-Variogram

- Consider two intrinsic stationary processes $\{X(\boldsymbol{s}) : \boldsymbol{s} \in D\}$ and $\{Y(\boldsymbol{s}) : \boldsymbol{s} \in D\}$ with semivariograms $\gamma_X(\cdot)$ and $\gamma_Y(\cdot)$.
- The cross-variogram is defined as

$$\gamma_{XY}(\boldsymbol{h}) = \frac{1}{2}\mathbb{E}[(X(\boldsymbol{s}+\boldsymbol{h}) - X(\boldsymbol{s}))(Y(\boldsymbol{s}+\boldsymbol{h}) - Y(\boldsymbol{s}))],$$

  where $\boldsymbol{s}, \boldsymbol{s}+\boldsymbol{h} \in D$.
- The cross-covariogram is defined as

$$C_{XY}(\boldsymbol{h}) = \mathbb{E}[(X(\boldsymbol{s}) - \mathbb{E}[X(\boldsymbol{s})])(Y(\boldsymbol{s}+\boldsymbol{h}) - \mathbb{E}[Y(\boldsymbol{s}+\boldsymbol{h}])].$$

- The codispersion coefficient is

$$\rho_{XY}(\boldsymbol{h}) = \frac{\gamma_{XY}(\boldsymbol{h})}{\sqrt{\mathbb{E}[X(\boldsymbol{s}+\boldsymbol{h}) - X(\boldsymbol{s})]^2\mathbb{E}[Y(\boldsymbol{s}+\boldsymbol{h}) - Y(\boldsymbol{s})]^2}}.$$

- The cross-correlation coefficient is

$$\varrho_{XY}(\boldsymbol{h}) = \frac{C_{XY}(\boldsymbol{h})}{\sqrt{\mathbb{E}[X(\boldsymbol{s}) - \mathbb{E}[X(\boldsymbol{s})]]^2\mathbb{E}[Y(\boldsymbol{s}+\boldsymbol{h}) - \mathbb{E}[Y(\boldsymbol{s}+\boldsymbol{h})]]^2}}.$$

Homework Problems

1. Let $\{X(\boldsymbol{s}) : \boldsymbol{s} \in \mathbb{R}^d\}$ denote a fractional Brownian motion process with mean zero and covariance cov$[X(\boldsymbol{s}), X(\boldsymbol{u})] = \sigma^2\{||\boldsymbol{s}||^{2H} + ||\boldsymbol{u}||^{2H} - ||\boldsymbol{s} - \boldsymbol{u}||^{2H}\}; 0 < H < 1.$
   a. Show that $X(\cdot)$ is intrinsically stationary and evaluate its variogram $2\gamma(\cdot)$.
   b. Let $2\widehat{\gamma}(\boldsymbol{h})$ be an estimate of $2\gamma(h)$. Explain how σ^2 and H could be estimated from a plot of $\log(2\widehat{\gamma}(\boldsymbol{h}))$ versus $\log(||\boldsymbol{h}||)$.
2. Let $\boldsymbol{Y}(\boldsymbol{s}) = (Y(\boldsymbol{s}_1), \dots, Y(\boldsymbol{s}_n))^\top$ be a spatial random vector such that
$$\boldsymbol{Y}(\boldsymbol{s}) = \boldsymbol{X}(\boldsymbol{s})\boldsymbol{\beta} + \epsilon(\boldsymbol{s}),$$
   where $\boldsymbol{X}(\boldsymbol{s})$ is an $n \times p$ design matrix, $\boldsymbol{\epsilon}(\boldsymbol{s}) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}(\boldsymbol{\theta}))$ with $\boldsymbol{R}(\boldsymbol{\theta})$ being a non-singular correlation matrix and $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^k$.
   a. Compute the generalized least squares (GLS) estimate of $\boldsymbol{\beta}$ and prove that it is unbiased.

Homework Problems

b. If $\boldsymbol{X}(\boldsymbol{s})\boldsymbol{\beta} = \mathbf{1}\mu$, $\mu \in \mathbb{R}$, prove that the Fisher information quantity satisfies

$$I(\mu) = \mathbb{E}\left[-\frac{\partial^2 \log L(\boldsymbol{Y}, \mu, \boldsymbol{R}(\boldsymbol{\theta}))}{\partial \mu^2}\right] \geq 1.$$

Comment briefly.

c. If $\boldsymbol{X}(\boldsymbol{s})\boldsymbol{\beta} = \mathbf{1}\mu$, $\mu \in \mathbb{R}$, and our goal is to predict the process $X(\cdot)$ in the unobserved location $\boldsymbol{s}_0$, consider the predictor of the form

$$\widehat{X}(\boldsymbol{s}_0) = \widehat{\mu} + \boldsymbol{c}^\top \boldsymbol{R}(\boldsymbol{\theta})^{-1}\left(\boldsymbol{X}(\boldsymbol{s}) - \mathbf{1}\widehat{\mu}\right),$$

where $\widehat{\mu}$ is the GLS of μ. Prove that

$$\widehat{X}(\boldsymbol{s}_0) = \left[\boldsymbol{c}^\top + \frac{\mathbf{1}^\top(1 - \boldsymbol{c}^\top \boldsymbol{R}(\boldsymbol{\theta})^{-1}\mathbf{1})}{\mathbf{1}^\top R(\boldsymbol{\theta})^{-1}\mathbf{1}}\right]\boldsymbol{R}(\boldsymbol{\theta})^{-1} Y(\boldsymbol{s}).$$

Homework Problems

3. To explore the capabilities of R for managing images, load the imagematrix.R function from the website https://cran.r-project.org/src/contrib/Archive/rimage/. This function plots RGB images.

a. In the R command window, run the following code to create an RGB image. Check that the result is a complete yellow image of size 512×512.

```
> n <- 512
> m <- 512
> R1 <- matrix(rep(1, n * m), n, m)
> G1 <- matrix(rep(1, n * m), n, m)
> B1 <- matrix(rep(0, n * m), n, m)
> W <- array(dim = c(n,m,3), c(R1,G1,B1))
> plot(imagematrix(W))
```

b. Repeat the previous example with the following code. What color is the resulting image?

```
> R2 <- matrix(rep(0, n * m), n, m)
> G2 <- matrix(rep(0.5, n * m), n, m)
> B2 <- matrix(rep(1, n * m), n, m)
> M <- array(dim = c(n,m,3), c(R2,G2,B2))
> plot(imagematrix(M))
```

3. The Modified t Test

- Assume that the two processes $X(\boldsymbol{s})$ and $Y(\boldsymbol{s})$ have been measured on $A = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n\} \subset D$.

- Also assume that $\boldsymbol{X} = (X(\boldsymbol{s}_1), \ldots, X(\boldsymbol{s}_n))^\top$ and $\boldsymbol{Y} = (Y(\boldsymbol{s}_1), \ldots, Y(\boldsymbol{s}_n))^\top$ follow a multivariate normal distribution with covariance matrices $\boldsymbol{\Sigma}_X$ y $\boldsymbol{\Sigma}_Y$, respectively.

- A parametric hypothesis testing procedure can be used to dilucidate the hypotheses of presence or absence of correlation between $X(\boldsymbol{s})$ and $Y(\boldsymbol{s})$ considering:

$$\mathsf{H}_0 : \rho_{XY} = 0 \qquad \text{against} \qquad \mathsf{H}_1 : \rho_{XY} \neq 0.$$

- This problem was studied by Cliford et al. (1989) and Dutilleul (1993).

The Modified t Test

- the fundamental idea comes from noticing that the statistic

$$t = \frac{r_{XY}\sqrt{M-2}}{\sqrt{1-r_{XY}^2}} \underset{approx.}{\sim} t_{M-2}.$$

- M is called the *effective sample size* which is defined through

$$M = 1 + (\mathrm{var}[r_{XY}])^{-1},$$

where $r_{XY} = s_{XY}/\sqrt{s_X^2 s_Y^2}$, $\overline{X} = \frac{1}{n}\sum_i X(\boldsymbol{s}_i)$, $s_X^2 = \frac{1}{n}\sum_i (X(\boldsymbol{s}_i) - \overline{X})^2$, $s_{XY} = \frac{1}{n}\sum_i (X(\boldsymbol{s}_i) - \overline{X})(Y(\boldsymbol{s}_i) - \overline{Y})$, and similarly for $\overline{Y}$ and s_Y^2.

UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

Day 3