

# Evaluación curso Introducción a la Geoestadística

*Guzmán López*

*30 de noviembre de 2016*

## Contents

<b>Consigna</b>	<b>1</b>
<b>Ambiente de trabajo en R</b>	<b>1</b>
Información de la sesión . . . . .	2
Librerías utilizadas . . . . .	2
Cargar datos . . . . .	3
<b>Respuesta 1</b>	<b>3</b>
Estructura y resumen de los datos . . . . .	3
Descripción de la variación . . . . .	4
Normalidad . . . . .	10
Transformaciones . . . . .	11
Q-Q Plots . . . . .	14
Análisis de tendencias . . . . .	19
Covarianza y correlación . . . . .	21
Índices de autocorrelación . . . . .	21
Supuestos distribucionales . . . . .	21
Cambios en la media dentro del lote . . . . .	21
Valores atípicos . . . . .	23
<b>Respuesta 2</b>	<b>23</b>
Variograma empírico . . . . .	23
Estructura espacial . . . . .	23
Proceso espacial (isotropía y anisotropía) . . . . .	23
Dependencia espacial entre campañas . . . . .	23

## Consigna

Se adjunta un archivo de datos espaciales de **rinde de maíz** y **covariables ambientales** para un lote de la región Pampeana para dos campañas agrícolas. El archivo está en formato de planilla de cálculo, con 2310 registros organizados en ocho campos/variables: Identificador, coordenadas planares UTM20S, cota topográfica o elevación del terreno, profundidad de napa en floración para primera y segunda campaña (C1 y C2) y rendimiento de maíz en t/ha para la primera y segunda campaña. Los datos de rinde fueron obtenidos mediante monitores montados a cosechadoras y la profundidad de napa fue estimada a partir de la integración de datos en pozos de observación, modelos de simulación hidrodinámicos del acuífero y modelos digitales de elevación del terreno.

## Ambiente de trabajo en R

En esta sección se establecieron y describen las características del ambiente de trabajo en R que fueron necesarias para poder comenzar a responder a las preguntas de la evaluación.

## Información de la sesión

```
## R version 3.3.1 (2016-06-21)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Antergos Linux
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
## [1] RevoUtilsMath_10.0.0
##
## loaded via a namespace (and not attached):
## [1] backports_1.0.4  magrittr_1.5      RevoUtils_10.0.1 rprojroot_1.1
## [5] tools_3.3.1      htmltools_0.3.5  yaml_2.1.14      Rcpp_0.12.8
## [9] stringi_1.1.2    rmarkdown_1.2    knitr_1.15.1     stringr_1.1.0
## [13] digest_0.6.10    evaluate_0.10
```

Se utilizó el Ambiente de Desarrollo Integrado (*IDE*) de RStudio (version 0.99.903) para R.

## Librerías utilizadas

Se cargaron todas las librerías que fueron utilizadas para la realización de este trabajo.

```
# Geoestadística
library('gstat')
library('geoR')
library('geoRglm')

# Objetos espaciales
library('sp')

# Algunos estadísticos
library('moments')

# Reestructurar datos
library('reshape2')

# Transformaciones Box-Cox
library('forecast')

# Gráficos

# Versión dev de ggplot2, requiere:
# devtools::install_github("hadley/scales")
# devtools::install_github("tidyverse/ggplot2")
library('ggplot2')
library('ggplot2')
library('gridExtra')

# Escalas de colores
library('RColorBrewer')
```

## Cargar datos

Se cargaron en R los datos espaciales de **rinde de maíz** y **covariables ambientales** para un lote de la región Pampeana para dos campañas agrícolas. Para ello, el archivo original en formato *Hoja de cálculo de Microsoft Excel* (DataExamenGeo16.xlsx) fue convertido a un *Documento de texto plano* (DataExamenGeo16.csv). Para la conversión se mantuvieron las cabeceras originales, se asignó como delimitador de campos la coma (,) y el punto (.) como separador decimal y se asignó una extensión de archivo de tipo .csv (valores separados comas).

```
# Cargar datos
datosRindeMaiz <- read.csv(file = "data/DataExamenGeo16.csv", header = TRUE, sep = ",", dec = ".")

# Mostrar las primeras seis filas
head(datosRindeMaiz)
```

##	ID	Lat	Long	Elev	NapaDicC1	NapaDicC2	maizC1	maizC2
## 1	1	3913.263	16153.52	245.99	4.98	4.38	6.8833	4.1816
## 2	2	3933.332	16153.53	245.90	4.85	4.25	9.1300	6.4079
## 3	3	3953.291	16153.54	245.59	4.50	3.90	7.6300	6.2962
## 4	4	3973.250	16153.56	245.48	4.34	3.74	7.5800	6.9209
## 5	5	3993.319	16153.57	245.34	4.16	3.56	9.3425	7.3573
## 6	6	4013.278	16153.58	245.23	4.01	3.41	10.4800	7.1703

Los datos **rendimiento de maíz** (*maizC1* y *maizC2*) y **covariables ambientales** (*Elev*, *NapaDicC1* y *NapaDicC2*) fueron de tipo variables continuas. Mientras que el espacio fue de dos dimensiones dado por  $x = \{Long, Lat\}$ .

### Crear objeto espacial (SpatialPointsDataFrame)

El objeto **datosRindeMaiz** de tipo `data.frame` fue transformado a un objeto espacial de tipo `SpatialPointsDataFrame` asignando las columnas *Long* y *Lat* como las coordenadas del objeto.

```
coordinates(datosRindeMaiz) <- c("Long", "Lat")
```

### Asignar Sistema de Coordenadas de Referencia (CRS)

Dado que se utilizó un Sistema de Coordenadas de Referencia relativo, se asignó al objeto **datosRindeMaiz** un Sistema de Coordenadas de Referencia vacío.

```
proj4string(datosRindeMaiz) <- CRS(as.character(NA))
```

## Respuesta 1

- 1) Realice un análisis descriptivo exhaustivo para evaluar los supuestos distribucionales del rinde para ambas campañas, la existencia de cambios en la media dentro del lote y la presencia de valores atípicos. Justifique todas las decisiones que tome respecto de los datos originales.

### Estructura y resumen de los datos

El objeto **datosRindeMaiz** posee:

1. un objeto de clase `data.frame` llamado **data** con los atributos de los datos de rinde de maíz sin las coordenadas (2310 observaciones y 6 variables: *ID*, *Elev*, *NapaDicC1*, *NapaDicC2*, *maizC1*, *maizC2*)

2. un objeto de clase `integer` llamado `coords.nrs` con dos valores que significan el número de columna del `data.frame` original del que se tomaron los datos de las coordenadas
3. un objeto de clase `matrix` llamado `coords` de 2310 filas por 2 columnas con las coordenadas (Long y Lat)
4. un objeto de clase `matrix` llamado `bbox` de 2 filas por 2 columnas con los límites espaciales (Long min, Long max, Lat min y Lat max)
5. un objeto de clase CRS llamado `proj4string` con el Sistema de Coordenadas de Referencia

```
str(datosRindeMaiz)
```

```
## Formal class 'SpatialPointsDataFrame' [package "sp"] with 5 slots
##   ..@ data      : 'data.frame':  2310 obs. of  6 variables:
##   .. ..$ ID      : int  [1:2310]  1 2 3 4 5 6 7 8 9 10 ...
##   .. ..$ Elev    : num  [1:2310] 246 246 246 245 245 ...
##   .. ..$ NapaDicC1: num  [1:2310] 4.98 4.85 4.5 4.34 4.16 4.01 3.93 3.8 3.69 3.59 ...
##   .. ..$ NapaDicC2: num  [1:2310] 4.38 4.25 3.9 3.74 3.56 3.41 3.33 3.2 3.09 2.99 ...
##   .. ..$ maizC1   : num  [1:2310] 6.88 9.13 7.63 7.58 9.34 ...
##   .. ..$ maizC2   : num  [1:2310] 4.18 6.41 6.3 6.92 7.36 ...
##   ..@ coords.nrs : int  [1:2]  3 2
##   ..@ coords     : num  [1:2310, 1:2] 16154 16154 16154 16154 16154 ...
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. ..$ : NULL
##   .. .. ..$ : chr  [1:2] "Long" "Lat"
##   ..@ bbox       : num  [1:2, 1:2] 16154 3913 17233 4733
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. ..$ : chr  [1:2] "Long" "Lat"
##   .. .. ..$ : chr  [1:2] "min" "max"
##   ..@ proj4string: Formal class 'CRS' [package "sp"] with 1 slot
##   .. .. ..@ projargs: chr NA
```

## Descripción de la variación

### Distribución de frecuencias

Para ilustrar como la distribución de los datos de cada variable se ubica con respecto a su mediana o para identificar valores extremos se realizaron Histogramas y Diagramas de cajas para cada una de las variables.

```
# Ver histogramas de las variables
ggDatosRindeMaiz <- ggplot(data = datosRindeMaiz@data)

# Maíz C1
ggHistMaizC1 <- ggDatosRindeMaiz +
  geom_histogram(aes(x = datosRindeMaiz$maizC1),
    binwidth = 0.5, fill = "orange", col = "white", alpha = 0.8) +
  labs(title = "", x = "Rendimiento maíz (t/ha)", y = "Frecuencia") +
  scale_x_continuous(limits = c(0, 18), breaks = seq(0, 18, 2)) +
  scale_y_continuous(limits = c(0, 325), breaks = seq(0, 325, 50)) +
  theme(plot.margin = unit(c(-1, 1, 1, 1), "mm"))

ggBoxPlotMaizC1 <- ggDatosRindeMaiz +
  geom_boxplot(aes(y = datosRindeMaiz$maizC1, x = rep("RM", length(datosRindeMaiz$maizC1))),
    fill = "orange", col = "black", alpha = 0.8) +
  labs(title = "Camp.1", x = "", y = "") +
  scale_y_continuous(limits = c(0, 18), breaks = seq(0, 18, 2), position = "top") +
```

```

theme(plot.margin = unit(c(1, 1, -1, 1), "mm")) +
coord_flip()

# Napa C1
ggHistNapaDicC1 <- ggDatosRindeMaiz +
  geom_histogram(aes(x = datosRindeMaiz$NapaDicC1),
    binwidth = 0.25, fill = "darkblue", col = "white", alpha = 0.7) +
  labs(title = "", x = "Profundidad napa (m)", y = "Frecuencia") +
  scale_x_continuous(limits = c(1.5, 6.75), breaks = seq(1.5, 6.75, 1)) +
  scale_y_continuous(limits = c(0, 350), breaks = seq(0, 350, 50)) +
  theme(plot.margin = unit(c(-1, 1, 1, 1), "mm"))

ggBoxPlotNapaDicC1 <- ggDatosRindeMaiz +
  geom_boxplot(aes(y = datosRindeMaiz$NapaDicC1, x = rep("PN", length(datosRindeMaiz$NapaDicC1))),
    fill = "darkblue", col = "black", alpha = 0.7) +
  labs(title = "Camp.1", x = "", y = "") +
  scale_y_continuous(limits = c(1.5, 6.75), breaks = seq(1.5, 6.75, 1), position = "top") +
  theme(plot.margin = unit(c(1, 1, -1, 1), "mm")) +
  coord_flip()

# Maíz C2
ggHistMaizC2 <- ggDatosRindeMaiz +
  geom_histogram(aes(x = datosRindeMaiz$maizC2),
    binwidth = 0.5, fill = "orange", col = "white", alpha = 0.8) +
  labs(title = "", x = "Rendimiento maíz (t/ha)", y = "Frecuencia") +
  scale_x_continuous(limits = c(0, 18), breaks = seq(0, 18, 2)) +
  scale_y_continuous(limits = c(0, 325), breaks = seq(0, 325, 50)) +
  theme(plot.margin = unit(c(-1, 1, 1, 1), "mm"))

ggBoxPlotMaizC2 <- ggDatosRindeMaiz +
  geom_boxplot(aes(y = datosRindeMaiz$maizC2, x = rep("RM", length(datosRindeMaiz$maizC2))),
    fill = "orange", col = "black", alpha = 0.8) +
  labs(title = "Camp.2", x = "", y = "") +
  scale_y_continuous(limits = c(0, 18), breaks = seq(0, 18, 2), position = "top") +
  theme(plot.margin = unit(c(1, 1, -1, 1), "mm")) +
  coord_flip()

# Napa C2
ggHistNapaDicC2 <- ggDatosRindeMaiz +
  geom_histogram(aes(x = datosRindeMaiz$NapaDicC2),
    binwidth = 0.25, fill = "darkblue", col = "white", alpha = 0.7) +
  labs(title = "", x = "Profundidad napa (m)", y = "Frecuencia") +
  scale_x_continuous(limits = c(1.5, 6.75), breaks = seq(1.5, 6.75, 1)) +
  scale_y_continuous(limits = c(0, 350), breaks = seq(0, 350, 50)) +
  theme(plot.margin = unit(c(-1, 1, 1, 1), "mm"))

ggBoxPlotNapaDicC2 <- ggDatosRindeMaiz +
  geom_boxplot(aes(y = datosRindeMaiz$NapaDicC2, x = rep("PN", length(datosRindeMaiz$NapaDicC2))),
    fill = "darkblue", col = "black", alpha = 0.7) +
  labs(title = "Camp.2", x = "", y = "") +
  scale_y_continuous(limits = c(1.5, 6.75), breaks = seq(1.5, 6.75, 1), position = "top") +
  theme(plot.margin = unit(c(1, 1, -1, 1), "mm")) +
  coord_flip()

```

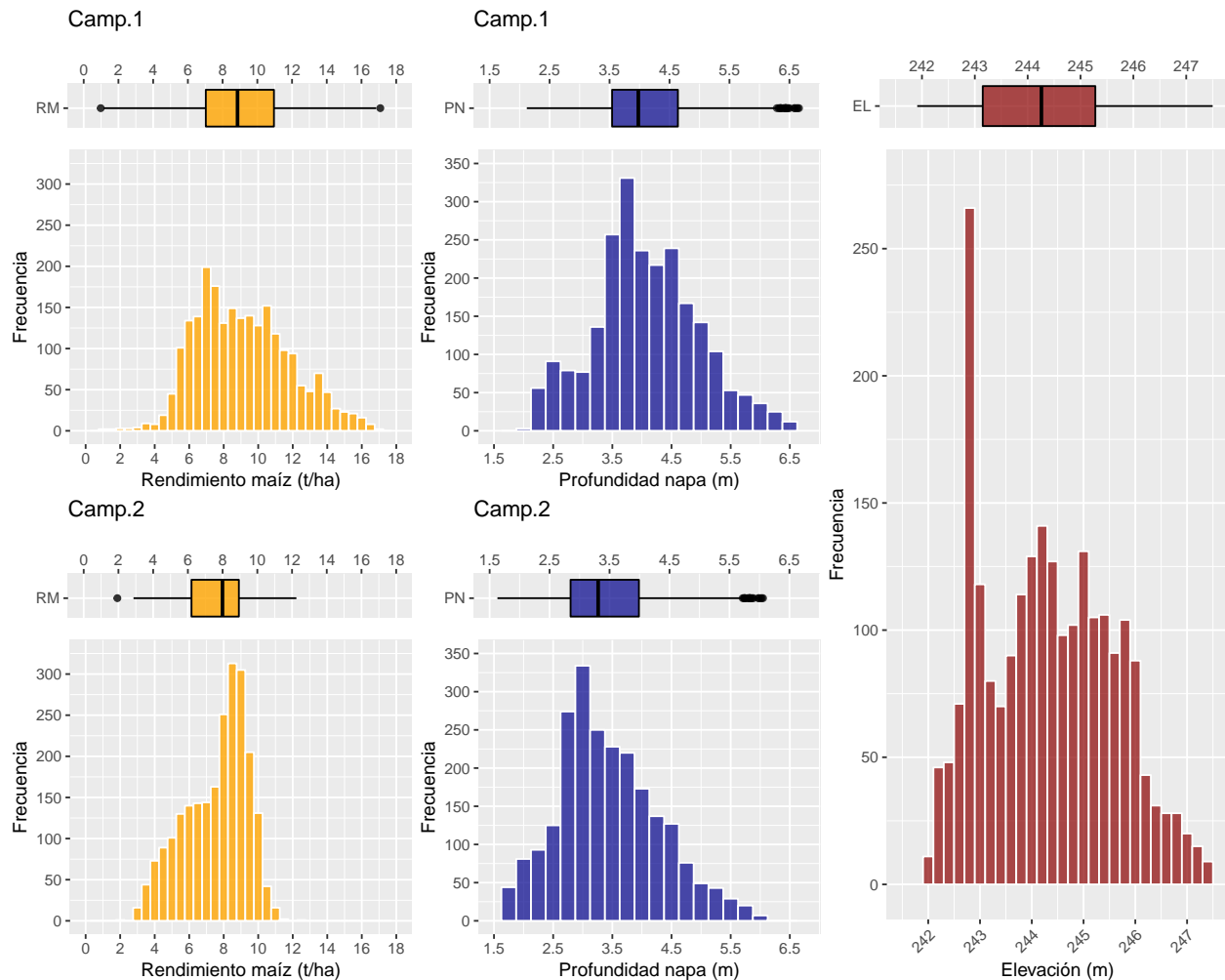
```

# Elevación
ggHistElev <- ggDatosRindeMaiz +
  geom_histogram(aes(x = datosRindeMaiz$Elev),
    binwidth = 0.2, fill = "darkred", col = "white", alpha = 0.7) +
  labs(title = "", x = "Elevación (m)", y = "Frecuencia") +
  scale_x_continuous(limits = c(241.5, 247.5), breaks = seq(240, 248, 1)) +
  scale_y_continuous(limits = c(0, 275), breaks = seq(0, 275, 50)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1), plot.margin = unit(c(-1, 1, 1, 1), "mm"))

ggBoxPlotElev <- ggDatosRindeMaiz +
  geom_boxplot(aes(y = datosRindeMaiz$Elev, x = rep("EL", length(datosRindeMaiz$Elev))),
    fill = "darkred", col = "black", alpha = 0.7) +
  labs(title = "", x = "", y = "") +
  scale_y_continuous(limits = c(241.5, 247.5), breaks = seq(240, 248, 1), position = "top") +
  theme(plot.margin = unit(c(1, 1, -1, 1), "mm")) +
  coord_flip()

# Ver plot
grid.arrange(ggBoxPlotMaizC1,
  ggHistMaizC1,
  ggBoxPlotMaizC2,
  ggHistMaizC2,
  ggBoxPlotNapaDicC1,
  ggHistNapaDicC1,
  ggBoxPlotNapaDicC2,
  ggHistNapaDicC2,
  ggBoxPlotElev,
  ggHistElev,
  ncol = 3, nrow = 4,
  layout_matrix = cbind(c(1,2,3,4), c(5,6,7,8), c(9,10,10,10)),
  widths = c(1, 1, 1), heights = c(1, 3, 1, 3))

```



Se observó para los datos de *rendimiento de maíz* que el valor de la mediana fue mayor para la *campaña 1* que para la *campaña 2* y que la dispersión de valores entorno a la mediana fue mayor para la *campaña 1* que para la *campaña 2*. Por otro lado, la frecuencia de valores de *rendimiento de maíz* por debajo de la mediana fue mayor para la *campaña 2* que para la *campaña 1* y la frecuencia de valores de *rendimiento de maíz* por encima de la mediana fue mayor para la *campaña 1* que para la *campaña 2*. Se observaron valores extremos de *rendimiento de maíz* mayores y menores a la mediana para la *campaña 1* y solo menores a la mediana para la *campaña 2*.

Para los datos de la *profundidad de napa* se observó que el valor de la mediana fue mayor para la *campaña 1* que para la *campaña 2* y que la dispersión de valores entorno a la mediana y la frecuencia de valores mayores y menores a la mediana fue semejante para ambas campañas. En ambas campañas se observaron valores extremos mayores a la mediana.

Para los datos de *elevación* se observó dispersión de valores entorno a la mediana y la frecuencia entre valores mayores y menores a la mediana fue semejante. No se observaron valores extremos para la *elevación*.

## Tendencia central y dispersión

Para analizar cuantitativamente el rango, la tendencia central y la dispersión de los datos de rendimiento de maíz y covariables ambientales se creó una función personalizada llamada `ResumenEstadisticas`. Además se incluyó que dicha función retorne el tipo de clase de la variable en R y la cantidad de celdas vacías (NA).

```

ResumenEstadisticas <- function(df) {

  # Número de columnas del data.frame
  ncol = ncol(df)

  # Estadísticas
  maximo = numeric()
  minimo = numeric()

  media = numeric()
  mediana = numeric()
  moda = vector()

  varianza = numeric()
  desvioEstandar = numeric()
  cv = numeric()
  asimetria = numeric()
  kurtosis = numeric()

  NAs = numeric()
  clases = character()

  # Función para calcular la moda
  Moda <- function(x) {
    ux = unique(x)
    ux[which.max(tabulate(match(x, ux)))]
  }

  # Iteración que recorre todas las columnas del data.frame
  for(i in 1:ncol) {

    # Condición de que la columna para las estadísticas tenga que ser de tipo numérica
    if(is.numeric(df[,i])) {

      mediana[i] = median(df[,i], na.rm = TRUE)
      media[i] = mean(df[,i], na.rm = TRUE)
      moda[i] = Moda(x = df[,i])
      varianza[i] = var(df[,i], na.rm = TRUE)
      desvioEstandar[i] = sd(df[,i], na.rm = TRUE)
      cv[i] = 100 * (sd(df[,i], na.rm = TRUE) / mean(df[,i], na.rm = TRUE))
      asimetria[i] = moments::skewness(df[,i], na.rm = TRUE)
      kurtosis[i] = moments::kurtosis(df[,i], na.rm = TRUE)
      minimo[i] = min(df[,i], na.rm = TRUE)
      maximo[i] = max(df[,i], na.rm = TRUE)
      NAs[i] = length(which(is.na(df[,i]) == TRUE))
      clases[i] = class(df[,i])

    }

    # En caso de que no sea de tipo numérica asignar NAs a las
    # estadísticas pero calcular el número de NAs y tipo de datos
    else {

```



```

    mediana[i] = NA
    media[i] = NA
    moda[i] = Moda(x = df[,i])
    varianza[i] = NA
    desvioEstandar[i] = NA
    cv[i] = 100 * NA
    asimetria[i] = NA
    kurtosis[i] = NA
    minimo[i] = NA
    maximo[i] = NA
    NAs[i] = length(which(is.na(df[,i]) == TRUE))
    clases[i] = class(df[,i])

  }
}

# Generar data.frame con todas las estadísticas
summary = data.frame("mínimo" = minimo,
                     "máximo" = maximo,
                     "media" = media,
                     "mediana" = mediana,
                     "moda" = moda,
                     "varianza" = varianza,
                     "ds" = desvioEstandar,
                     "cv" = cv,
                     "asimetría" = asimetria,
                     "kurtosis" = kurtosis,
                     "NAs" = NAs,
                     "clase" = clases)

# Asignar el nombre de las variables (columnas)
# como nombre de las filas para el objeto de salida
rownames(summary) <- colnames(df)

# Retorno de la función
return(summary)
}

```

Se llamó a la función `ResumenEstadisticas` pasando como parámetro el `data.frame` llamado `data` contenido en el objeto espacial `datosRindeMaiz` y se asignó el retorno de la función a un nuevo objeto llamado `resumenDatosRindeMaiz`. Además, se imprimió en consola el resultado de la ejecución del comando.

```

# Crear objeto con las principales estadísticas y mostrarlo en consola
(resumenDatosRindeMaiz <- ResumenEstadisticas(df = datosRindeMaiz@data))

```

##	mínimo	máximo	media	mediana	moda	varianza
## ID	1.0000	2310.0000	1155.500000	1155.50000	1.000	4.448675e+05
## Elev	241.9100	247.5000	244.310848	244.26000	242.770	1.560964e+00
## NapaDicC1	2.1200	6.6600	4.070797	3.98000	3.690	7.859209e-01
## NapaDicC2	1.6300	6.0600	3.442294	3.31000	3.030	7.373578e-01
## maizC1	0.9800	17.0850	9.134139	8.85935	7.000	7.433428e+00
## maizC2	1.9381	12.2521	7.552832	7.99375	9.402	3.309716e+00
##	ds	cv	asimetría	kurtosis	NAs	clase
## ID	666.9838829	57.7225342	0.0000000	1.800000	0	integer

```
## Elev      1.2493853  0.5113916  0.2129448  2.135257  0 numeric
## NapaDicC1  0.8865218 21.7776002  0.1903320  2.875718  0 numeric
## NapaDicC2  0.8586954 24.9454385  0.4559732  2.906922  0 numeric
## maizC1     2.7264314 29.8488072  0.3713567  2.693631  2 numeric
## maizC2     1.8192624 24.0871547 -0.5174745  2.397373  0 numeric
```

A través del objeto `resumenDatosRindeMaiz` fue posible obtener el/los estadístico/s necesarios. Por ejemplo:

1. Obtener el valor de la mediana del Rendimiento del maíz para la Campaña 1

```
# Ejemplo 1
(resumenDatosRindeMaiz["maizC1", "mediana"])
```

```
## [1] 8.85935
```

2. Obtener los valores de mínimo y máximo de la Elevación

```
# Ejemplo 2
(resumenDatosRindeMaiz["Elev", c("mínimo", "máximo")])
```

```
##      mínimo máximo
## Elev 241.91  247.5
```

Se observaron dos celdas vacías para los datos de *rendimiento de maíz* de la *campaña 2*. Todas las variables fueron de tipo numéricas a excepción de la variable identificadora *ID* que fue un entero.

### Tendencia central

Los valores de la media ( $\bar{z}$ ) y la mediana ( $Md$ ) del *rendimiento de maíz* para la *campaña 1* fueron mayores que para la *campaña 2* ( $\bar{z}_{Rm1} = 9.13 \text{ t.ha}^{-1}$ ,  $Md_{Rm1} = 8.86 \text{ t.ha}^{-1}$  y  $\bar{z}_{Rm2} = 7.55 \text{ t.ha}^{-1}$ ,  $Md_{Rm2} = 7.99 \text{ t.ha}^{-1}$ ) mientras que el valor de la moda ( $Mo$ ) fue de  $Mo_{Rm1} = 7.00 \text{ t.ha}^{-1}$  y  $Mo_{Rm1} = 9.40 \text{ t.ha}^{-1}$  respectivamente.

Los valores de la media y la mediana de la *profundidad de la napa* para la *campaña 1* ( $\bar{z} = 4.07 \text{ m}$ ,  $Md = 3.98 \text{ m}$ ) fueron mayores que para la *campaña 2* ( $\bar{z} = 3.44 \text{ m}$ ,  $Md = 3.31 \text{ m}$ ) mientras que el valor de la moda fue de  $3.69 \text{ m}$  y  $3.03 \text{ m}$  respectivamente.

Los valores de la media, la mediana y la moda para la *elevación* fueron de  $244.31 \text{ m}$ ,  $244.26 \text{ m}$  y  $242.77 \text{ m}$  respectivamente.

### Dispersión

El rango ( $R = x_{min}, x_{max}$ ) de la variable del *rendimiento de maíz* para la *campaña 1* fue mayor y con valores más altos y más bajos que para la *campaña 2* ( $R_{Rm1} = 0.9800 \text{ t.ha}^{-1}$ ,  $17.0850 \text{ t.ha}^{-1}$ ,  $R_{Rm2} = 1.9381 \text{ t.ha}^{-1}$ ,  $12.2521 \text{ t.ha}^{-1}$ ), para la variable *profundidad de napa* el rango fue similar para las dos campañas ( $R_{Pn1} = 2.12 \text{ m}$ ,  $6.66 \text{ m}$ ,  $R_{Pn2} = 1.63 \text{ m}$ ,  $6.06 \text{ m}$ ) y el rango de la *elevación* fue de  $R_{El} = 241.91 \text{ m}$ ,  $247.50 \text{ m}$ .

Los valores de la varianza y la mediana del *rendimiento de maíz* para la *campaña 1*

### Normalidad

Para analizar la normalidad de las variables se realizó la prueba de normalidad de Shapiro-Wilk. La hipótesis nula de la prueba de Shapiro-Wilk es que los datos son normales, por lo cual un *p-valor* significativo indicó que se rechazó la hipótesis nula, es decir que la variable analizada fuese normal.

```
# Elevación
shapiro.test(datosRindeMaiz$Elev)
```

```
##
## Shapiro-Wilk normality test
##
```

```
## data:  datosRindeMaiz$Elev
## W = 0.97116, p-value < 2.2e-16
# Profundidad de la napa (C1)
shapiro.test(datosRindeMaiz$NapaDicC1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datosRindeMaiz$NapaDicC1
## W = 0.99124, p-value = 1.297e-10
```

```
# Profundidad de la napa (C2)
shapiro.test(datosRindeMaiz$NapaDicC2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datosRindeMaiz$NapaDicC2
## W = 0.98135, p-value < 2.2e-16
```

```
# Rendimiento del maíz (C1)
shapiro.test(datosRindeMaiz$maizC1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datosRindeMaiz$maizC1
## W = 0.98251, p-value = 3.07e-16
```

```
# Rendimiento del maíz (C2)
shapiro.test(datosRindeMaiz$maizC2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datosRindeMaiz$maizC2
## W = 0.95925, p-value < 2.2e-16
```

Para el *rendimiento de maíz* y las covariables ambientales se rechazó la hipótesis nula de la prueba de normalidad de Shapiro-Wilk.

## Transformaciones

Se transformó los datos originales a nuevas escalas para acercarlos a una distribución normal. Se probaron las transformaciones logarítmicas, raíz cuadrada y Box-Cox.

```
# Transformar elevación

# Logaritmo
ElevLog <- log(datosRindeMaiz@data$Elev)
shapiro.test(ElevLog) # Probar normalidad de la transformación
```

```
##
##  Shapiro-Wilk normality test
##
## data:  ElevLog
```

```
## W = 0.97134, p-value < 2.2e-16
# Raíz cuadrada
ElevSqrt <- sqrt(datosRindeMaiz@data$Elev)
shapiro.test(ElevSqrt) # Probar normalidad de la transformación

##
## Shapiro-Wilk normality test
##
## data: ElevSqrt
## W = 0.97125, p-value < 2.2e-16
# Box-Cox
lambdaElev <- BoxCox.lambda(x = datosRindeMaiz@data$Elev) # Hallar Lambda
ElevBC <- BoxCox(x = datosRindeMaiz@data$Elev, lambda = lambdaElev) # Transf. vector
shapiro.test(ElevBC) # Probar normalidad de la transformación

##
## Shapiro-Wilk normality test
##
## data: ElevBC
## W = 0.9715, p-value < 2.2e-16
# Transformar profundidad napa (C1)
# Logaritmo
NapaDicC1Log <- log(datosRindeMaiz@data$NapaDicC1)
shapiro.test(NapaDicC1Log)

##
## Shapiro-Wilk normality test
##
## data: NapaDicC1Log
## W = 0.98123, p-value < 2.2e-16
# Raíz cuadrada
NapaDicC1Sqrt <- sqrt(datosRindeMaiz@data$NapaDicC1)
shapiro.test(NapaDicC1Sqrt)

##
## Shapiro-Wilk normality test
##
## data: NapaDicC1Sqrt
## W = 0.99189, p-value = 4.568e-10
# Box-Cox
lambdaNapaDicC1 <- BoxCox.lambda(x = datosRindeMaiz@data$NapaDicC1)
NapaDicC1BC <- BoxCox(x = datosRindeMaiz@data$NapaDicC1, lambda = lambdaNapaDicC1)
shapiro.test(NapaDicC1BC)

##
## Shapiro-Wilk normality test
##
## data: NapaDicC1BC
## W = 0.9875, p-value = 2.42e-13
# Transformar profundidad napa (C2)
```

```

# Logaritmo
NapaDicC2Log <- log(datosRindeMaiz@data$NapaDicC2)
shapiro.test(NapaDicC2Log)

##
##  Shapiro-Wilk normality test
##
## data:  NapaDicC2Log
## W = 0.99209, p-value = 6.856e-10

# Raíz cuadrada
NapaDicC2Sqrt <- sqrt(datosRindeMaiz@data$NapaDicC2)
shapiro.test(NapaDicC2Sqrt)

##
##  Shapiro-Wilk normality test
##
## data:  NapaDicC2Sqrt
## W = 0.99313, p-value = 6.067e-09

# Box-Cox
lambdaNapaDicC2 <- BoxCox.lambda(x = datosRindeMaiz@data$NapaDicC2)
NapaDicC2BC <- BoxCox(x = datosRindeMaiz@data$NapaDicC2, lambda = lambdaNapaDicC2)
shapiro.test(NapaDicC2BC)

##
##  Shapiro-Wilk normality test
##
## data:  NapaDicC2BC
## W = 0.99138, p-value = 1.709e-10

# Transformar rendimiento maíz (C1)

# Logaritmo
maizC1Log <- log(datosRindeMaiz@data$maizC1)
shapiro.test(maizC1Log)

##
##  Shapiro-Wilk normality test
##
## data:  maizC1Log
## W = 0.96474, p-value < 2.2e-16

# Raíz cuadrada
maizC1Sqrt <- sqrt(datosRindeMaiz@data$maizC1)
shapiro.test(maizC1Sqrt)

##
##  Shapiro-Wilk normality test
##
## data:  maizC1Sqrt
## W = 0.99035, p-value = 2.61e-11

# Box-Cox
lambdaMaizC1Log <- BoxCox.lambda(x = datosRindeMaiz@data$maizC1)
maizC1BC <- BoxCox(x = datosRindeMaiz@data$maizC1, lambda = lambdaMaizC1Log)
shapiro.test(maizC1BC)

```

```
##
## Shapiro-Wilk normality test
##
## data:  maizC1BC
## W = 0.9473, p-value < 2.2e-16

# Transformar rendimiento maíz (C2)

# Logaritmo
maizC2Log <- log(datosRindeMaiz@data$maizC2)
shapiro.test(maizC2Log)

##
## Shapiro-Wilk normality test
##
## data:  maizC2Log
## W = 0.90957, p-value < 2.2e-16

# Raíz cuadrada
maizC2Sqrt <- sqrt(datosRindeMaiz@data$maizC2)
shapiro.test(maizC2Sqrt)

##
## Shapiro-Wilk normality test
##
## data:  maizC2Sqrt
## W = 0.93893, p-value < 2.2e-16

# Box-Cox
lambdaMaizC2Log <- BoxCox.lambda(x = datosRindeMaiz@data$maizC2)
maizC2BC <- BoxCox(x = datosRindeMaiz@data$maizC2, lambda = lambdaMaizC2Log)
shapiro.test(maizC2BC)

##
## Shapiro-Wilk normality test
##
## data:  maizC2BC
## W = 0.96354, p-value < 2.2e-16
```

Ninguna de las transformaciones de los datos permitió que se pudiera fallar en rechazar la hipótesis nula de normalidad del test de Shapiro-Wilk.

## Q-Q Plots

```
# Q-Q plot rendimiento maíz (C1)
ggQQmaizC1 <- ggDatosRindeMaiz +
  geom_qq(aes(sample = maizC1)) +
  geom_abline(intercept = mean(datosRindeMaiz@data$maizC1, na.rm = TRUE),
              slope = sd(datosRindeMaiz@data$maizC1, na.rm = TRUE), col = "red") +
  labs(title = "Rend.maíz C1", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQmaizC1Log <- ggDatosRindeMaiz +
  geom_qq(aes(sample = log(maizC1))) +
  geom_abline(intercept = mean(log(datosRindeMaiz@data$maizC1), na.rm = TRUE),
              slope = sd(log(datosRindeMaiz@data$maizC1), na.rm = TRUE), col = "red") +
```

```

labs(title = "Log Rend.maíz C1", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQmaizC1Sqrt <- ggDatosRindeMaiz +
  geom_qq(aes(sample = sqrt(maizC1))) +
  geom_abline(intercept = mean(sqrt(datosRindeMaiz@data$maizC1), na.rm = TRUE),
    slope = sd(sqrt(datosRindeMaiz@data$maizC1), na.rm = TRUE), col = "red") +
  labs(title = "Raíz cuadrada Rend.maíz C1", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQmaizC1BC <- ggDatosRindeMaiz +
  geom_qq(aes(sample = maizC1BC)) +
  geom_abline(intercept = mean(maizC1BC, na.rm = TRUE),
    slope = sd(maizC1BC, na.rm = TRUE), col = "red") +
  labs(title = "Box-Cox Rend.maíz C1", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

# Q-Q plot rendimiento maíz (C2)
ggQQmaizC2 <- ggDatosRindeMaiz +
  geom_qq(aes(sample = maizC2)) +
  geom_abline(intercept = mean(datosRindeMaiz@data$maizC2, na.rm = TRUE),
    slope = sd(datosRindeMaiz@data$maizC2, na.rm = TRUE), col = "red") +
  labs(title = "Rend.maíz C2", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQmaizC2Log <- ggDatosRindeMaiz +
  geom_qq(aes(sample = log(maizC2))) +
  geom_abline(intercept = mean(log(datosRindeMaiz@data$maizC2), na.rm = TRUE),
    slope = sd(log(datosRindeMaiz@data$maizC2), na.rm = TRUE), col = "red") +
  labs(title = "Log Rend.maíz C2", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQmaizC2Sqrt <- ggDatosRindeMaiz +
  geom_qq(aes(sample = sqrt(maizC2))) +
  geom_abline(intercept = mean(sqrt(datosRindeMaiz@data$maizC2), na.rm = TRUE),
    slope = sd(sqrt(datosRindeMaiz@data$maizC2), na.rm = TRUE), col = "red") +
  labs(title = "Raíz cuadrada Rend.maíz C2", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQmaizC2BC <- ggDatosRindeMaiz +
  geom_qq(aes(sample = maizC2BC)) +
  geom_abline(intercept = mean(maizC2BC, na.rm = TRUE),
    slope = sd(maizC2BC, na.rm = TRUE), col = "red") +
  labs(title = "Box-Cox Rend.maíz C2", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

# Q-Q plot profundidad de la napa (C1)
ggQQNapaDicC1 <- ggDatosRindeMaiz +
  geom_qq(aes(sample = NapaDicC1)) +
  geom_abline(intercept = mean(datosRindeMaiz@data$NapaDicC1),
    slope = sd(datosRindeMaiz@data$NapaDicC1), col = "red") +
  labs(title = "Prof.napa C1", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQNapaDicC1Log <- ggDatosRindeMaiz +
  geom_qq(aes(sample = log(NapaDicC1))) +
  geom_abline(intercept = mean(log(datosRindeMaiz@data$NapaDicC1),
    slope = sd(log(datosRindeMaiz@data$NapaDicC1), col = "red") +
  labs(title = "Log Prof.napa C1", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQNapaDicC1Sqrt <- ggDatosRindeMaiz +

```

```

geom_qq(aes(sample = sqrt(NapaDicC1))) +
geom_abline(intercept = mean(sqrt(datosRindeMaiz@data$NapaDicC1)),
            slope = sd(sqrt(datosRindeMaiz@data$NapaDicC1)), col = "red") +
labs(title = "Raíz cuadrada Prof.napa C1", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQNapaDicC1BC <- ggDatosRindeMaiz +
  geom_qq(aes(sample = NapaDicC1BC)) +
  geom_abline(intercept = mean(NapaDicC1BC, na.rm = TRUE),
              slope = sd(NapaDicC1BC, na.rm = TRUE), col = "red") +
  labs(title = "Box-Cox Prof.napa C1", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

# Q-Q plot profundidad de la napa (C2)
ggQQNapaDicC2 <- ggDatosRindeMaiz +
  geom_qq(aes(sample = NapaDicC2)) +
  geom_abline(intercept = mean(datosRindeMaiz@data$NapaDicC2),
              slope = sd(datosRindeMaiz@data$NapaDicC2), col = "red") +
  labs(title = "Prof.napa C2", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQNapaDicC2Log <- ggDatosRindeMaiz +
  geom_qq(aes(sample = log(NapaDicC2))) +
  geom_abline(intercept = mean(log(datosRindeMaiz@data$NapaDicC2)),
              slope = sd(log(datosRindeMaiz@data$NapaDicC2)), col = "red") +
  labs(title = "Log Prof.napa C2", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQNapaDicC2Sqrt <- ggDatosRindeMaiz +
  geom_qq(aes(sample = sqrt(NapaDicC2))) +
  geom_abline(intercept = mean(sqrt(datosRindeMaiz@data$NapaDicC2)),
              slope = sd(sqrt(datosRindeMaiz@data$NapaDicC2)), col = "red") +
  labs(title = "Raíz cuadrada Prof.napa C2", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQNapaDicC2BC <- ggDatosRindeMaiz +
  geom_qq(aes(sample = NapaDicC2BC)) +
  geom_abline(intercept = mean(NapaDicC2BC, na.rm = TRUE),
              slope = sd(NapaDicC2BC, na.rm = TRUE), col = "red") +
  labs(title = "Box-Cox Prof.napa C2", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

# Q-Q plot elevación
ggQQElev <- ggDatosRindeMaiz +
  geom_qq(aes(sample = Elev)) +
  geom_abline(intercept = mean(datosRindeMaiz@data$Elev),
              slope = sd(datosRindeMaiz@data$Elev), col = "red") +
  labs(title = "Elevación", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQElevLog <- ggDatosRindeMaiz +
  geom_qq(aes(sample = log(Elev))) +
  geom_abline(intercept = mean(log(datosRindeMaiz@data$Elev)),
              slope = sd(log(datosRindeMaiz@data$Elev)), col = "red") +
  labs(title = "Log Elevación", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQElevSqrt <- ggDatosRindeMaiz +
  geom_qq(aes(sample = sqrt(Elev))) +
  geom_abline(intercept = mean(sqrt(datosRindeMaiz@data$Elev)),
              slope = sd(sqrt(datosRindeMaiz@data$Elev)), col = "red") +

```



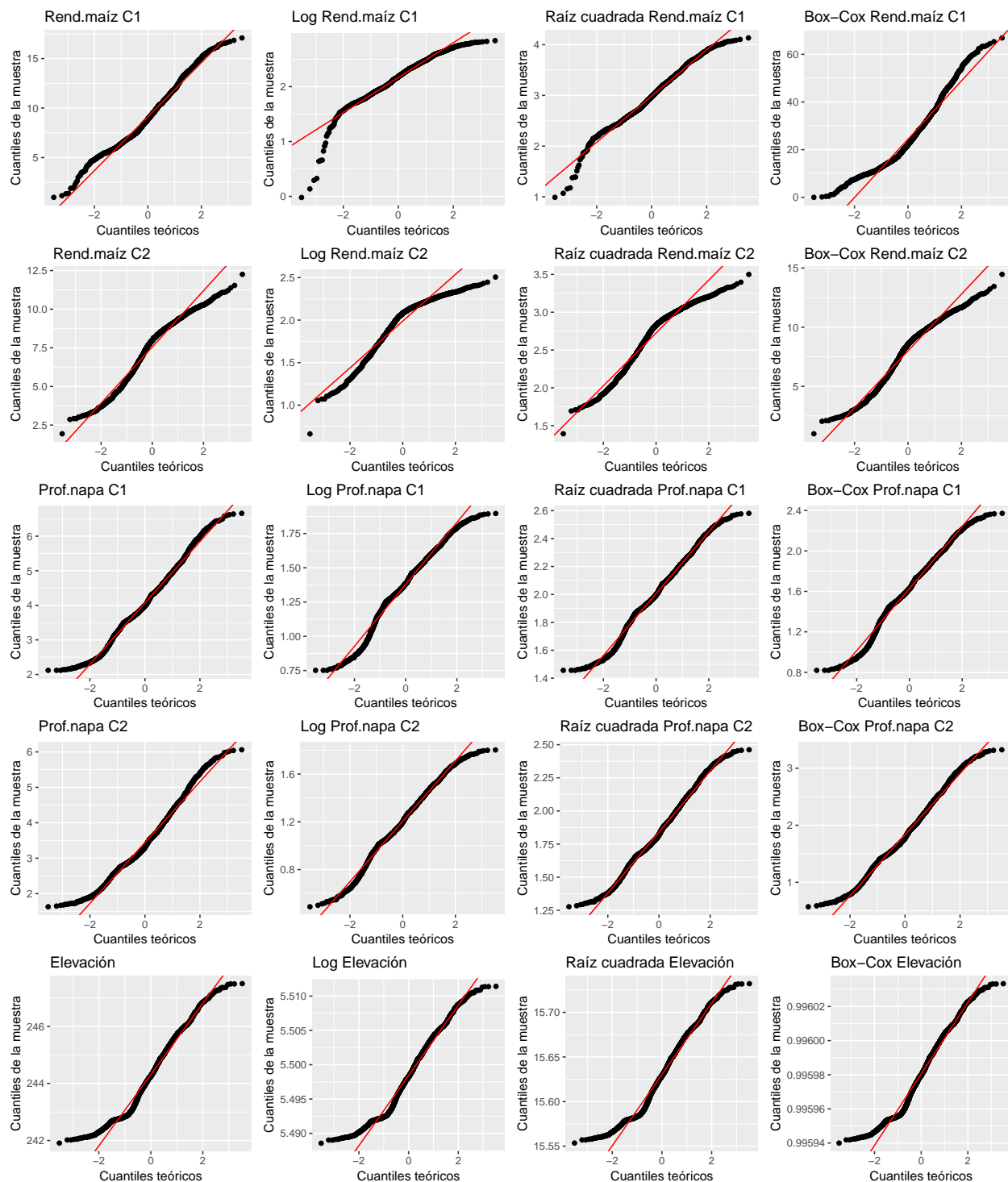
```

  labs(title = "Raíz cuadrada Elevación", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

ggQQElevBC <- ggDatosRindeMaiz +
  geom_qq(aes(sample = ElevBC)) +
  geom_abline(intercept = mean(ElevBC), slope = sd(ElevBC), col = "red") +
  labs(title = "Box-Cox Elevación", x = "Cuantiles teóricos", y = "Cuantiles de la muestra")

# Ver plots
grid.arrange(ggQQmaizC1, ggQQmaizC2, ggQQNapaDicC1, ggQQNapaDicC2, ggQQElev,
  ggQQmaizC1Log, ggQQmaizC2Log, ggQQNapaDicC1Log, ggQQNapaDicC2Log, ggQQElevLog,
  ggQQmaizC1Sqrt, ggQQmaizC2Sqrt, ggQQNapaDicC1Sqrt, ggQQNapaDicC2Sqrt, ggQQElevSqrt,
  ggQQmaizC1BC, ggQQmaizC2BC, ggQQNapaDicC1BC, ggQQNapaDicC2BC, ggQQElevBC,
  ncol = 4, nrow = 5,
  layout_matrix = cbind(c(1,2,3,4,5), c(6,7,8,9,10), c(11,12,13,14,15), c(16,17,18,19,20))
)

```



A partir del análisis de los Q-Q plots (cuantiles - cuantiles) se observó que todas las transformaciones probadas no acercaron sustantivamente los datos originales a distribuciones normales. Incluso, los Q-Q plots para los datos originales confirman el desvío de los datos de una distribución normal.

## Análisis de tendencias

Para analizar si existen tendencias entre el rendimiento de maíz y las covariables ambientales se realizaron gráficos de dispersión de puntos junto con líneas basadas en suavizados de la medias locales de los valores.

```
# Rendimiento maíz vs Longitud
ggTend_MaizC1VsLong <- ggDatosRindeMaiz +
  geom_point(aes(y = maizC1, x = datosRindeMaiz@coords[, "Long"])) +
  geom_smooth(aes(y = maizC1, x = datosRindeMaiz@coords[, "Long"])) +
  labs(title = "", x = "Longitud", y = "Rendimiento maíz C1 (t/ha)")

ggTend_MaizC2VsLong <- ggDatosRindeMaiz +
  geom_point(aes(y = maizC2, x = datosRindeMaiz@coords[, "Long"])) +
  geom_smooth(aes(y = maizC2, x = datosRindeMaiz@coords[, "Long"])) +
  labs(title = "", x = "Longitud", y = "Rendimiento maíz C2 (t/ha)")

# Rendimiento maíz vs Latitud
ggTend_MaizC1VsLat <- ggDatosRindeMaiz +
  geom_point(aes(y = maizC1, x = datosRindeMaiz@coords[, "Lat"])) +
  geom_smooth(aes(y = maizC1, x = datosRindeMaiz@coords[, "Lat"])) +
  labs(title = "", x = "Latitud", y = "Rendimiento maíz C1 (t/ha)")

ggTend_MaizC2VsLat <- ggDatosRindeMaiz +
  geom_point(aes(y = maizC2, x = datosRindeMaiz@coords[, "Lat"])) +
  geom_smooth(aes(y = maizC2, x = datosRindeMaiz@coords[, "Lat"])) +
  labs(title = "", x = "Latitud", y = "Rendimiento maíz C2 (t/ha)")

# Rendimiento maíz vs Elevación
ggTend_MaizC1VsElev <- ggDatosRindeMaiz +
  geom_point(aes(y = maizC1, x = Elev)) +
  geom_smooth(aes(y = maizC1, x = Elev)) +
  labs(title = "", y = "Rendimiento maíz C1 (t/ha)", x = "Elevación (m)")

ggTend_MaizC2VsElev <- ggDatosRindeMaiz +
  geom_point(aes(y = maizC2, x = Elev)) +
  geom_smooth(aes(y = maizC2, x = Elev)) +
  labs(title = "", y = "Rendimiento maíz C2 (t/ha)", x = "Elevación (m)")

# Rendimiento maíz vs Profundidad de la napa
ggTend_MaizC1VsNapaDicC1 <- ggDatosRindeMaiz +
  geom_point(aes(y = maizC1, x = NapaDicC1)) +
  geom_smooth(aes(y = maizC1, x = NapaDicC1)) +
  labs(title = "", y = "Rendimiento maíz C1 (t/ha)", x = "Profundidad napa (m)")

ggTend_MaizC2VsNapaDicC2 <- ggDatosRindeMaiz +
  geom_point(aes(y = maizC2, x = NapaDicC2)) +
  geom_smooth(aes(y = maizC2, x = NapaDicC2)) +
  labs(title = "", y = "Rendimiento maíz C2 (t/ha)", x = "Profundidad napa (m)")

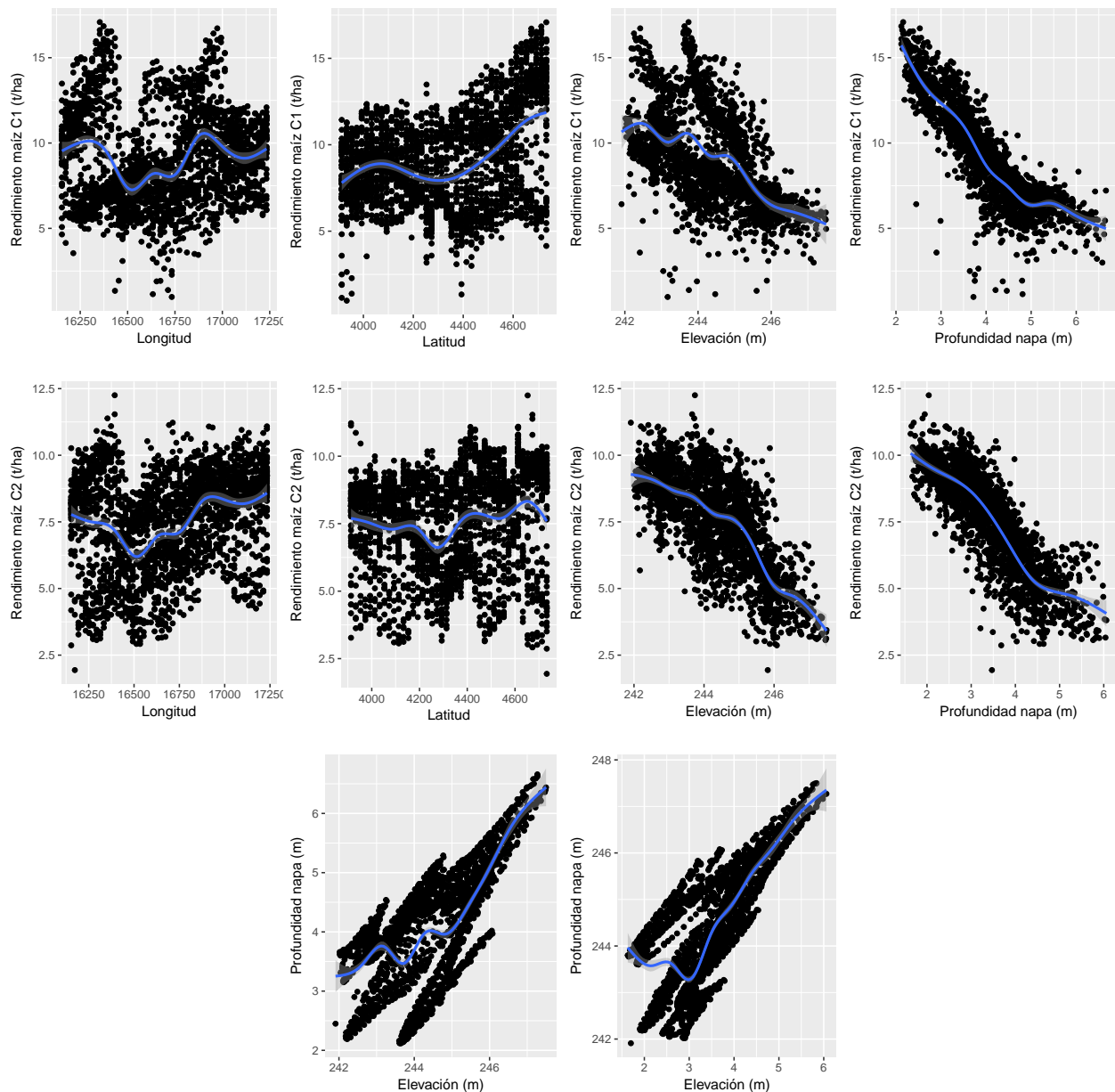
# Profundidad de la napa vs Elevación
ggTend_NapaDicC1VsElev <- ggDatosRindeMaiz +
  geom_point(aes(x = Elev, y = NapaDicC1)) +
  geom_smooth(aes(x = Elev, y = NapaDicC1)) +
  labs(title = "", x = "Elevación (m)", y = "Profundidad napa (m)")
```

```

ggTend_NapaDicC2VsElev <- ggDatosRindeMaiz +
  geom_point(aes(y = Elev, x = NapaDicC2)) +
  geom_smooth(aes(y = Elev, x = NapaDicC2)) +
  labs(title = "", x = "Elevación (m)", y = "Profundidad napa (m)")

# Ver plots
X11()
grid.arrange(ggTend_MaizC1VsLong, ggTend_MaizC1VsLat, ggTend_MaizC1VsElev, ggTend_MaizC1VsNapaDicC1,
  ggTend_MaizC2VsLong, ggTend_MaizC2VsLat, ggTend_MaizC2VsElev, ggTend_MaizC2VsNapaDicC2,
  ggTend_NapaDicC1VsElev, ggTend_NapaDicC2VsElev,
  ncol = 4, nrow = 3,
  layout_matrix = rbind(c(1,2,3,4), c(5,6,7,8), c(NA,9,10,NA))
)

```



A partir de los gráficos de dispersión y suavizado de medias, no se observó una tendencia entre el rendimiento de maíz y la latitud o longitud. Sin embargo, se observaron tendencias negativas entre el rendimiento de maíz y la elevación y también con la profundidad de la napa para las dos campañas. Se observó que el rendimiento de maíz decrece con la elevación del terreno y la profundidad de la napa. Asimismo, se observó una tendencia positiva entre la profundidad de la napa y la elevación.

## Covarianza y correlación

## Índices de autocorrelación

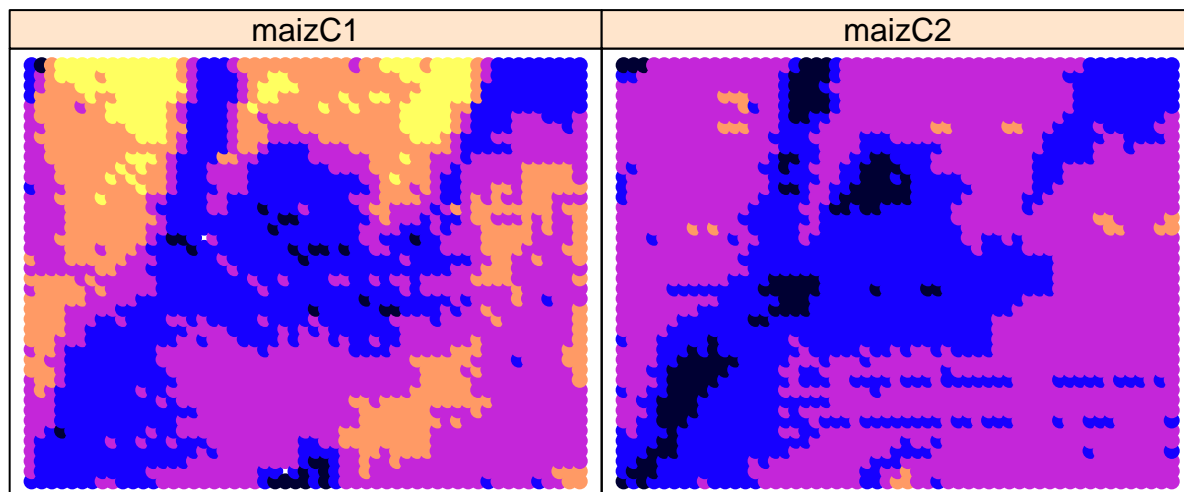
## Supuestos distribucionales

## Cambios en la media dentro del lote

Para observar los cambios en la media dentro del lote se realizaron mapas de puntos (“posting”) del rendimiento de maíz y las covariables ambientales.

```
# Mapa de puntos de rendimiento de maíz para el lote
spplot(obj = datosRindeMaiz, zcol = c("maizC1", "maizC2"), main = "Rendimiento de maíz (t/ha)")
```

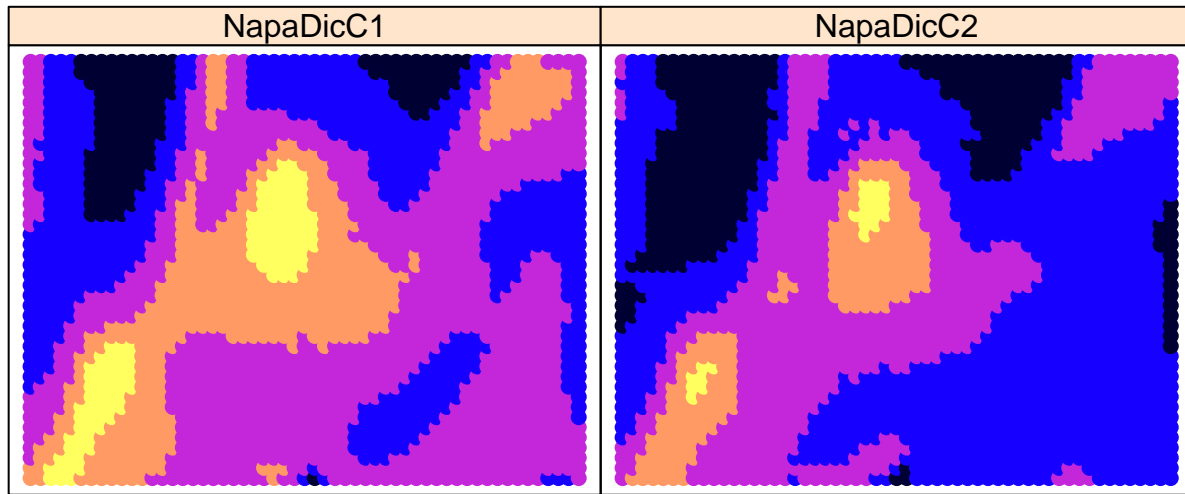
### Rendimiento de maíz (t/ha)



- [0.98,4.201]
- (4.201,7.422]
- (7.422,10.64]
- (10.64,13.86]
- (13.86,17.09]

```
# Mapa de puntos de la profundidad de la napa para el lote
spplot(obj = datosRindeMaiz, zcol = c("NapaDicC1", "NapaDicC2"), main = "Profundidad napa (m)")
```

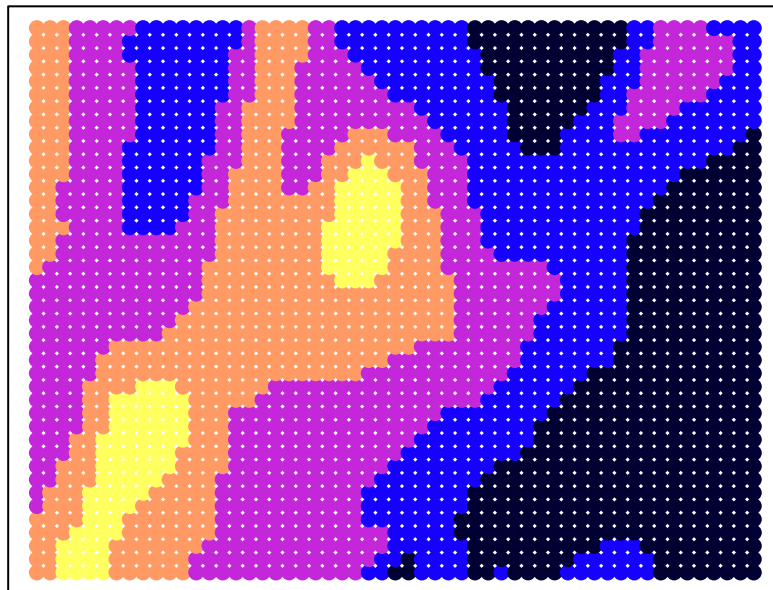
## Profundidad napa (m)



- [1.63,2.636]
- (2.636,3.642]
- (3.642,4.648]
- (4.648,5.654]
- (5.654,6.66]

```
# Mapa de puntos de la elevación del lote
spplot(obj = datosRindeMaiz, zcol = "Elev", main = "Elevación (m)")
```

## Elevación (m)



- [241.9,243]
- (243,244.1]
- (244.1,245.3]
- (245.3,246.4]
- (246.4,247.5]

**Valores atípicos**

## **Respuesta 2**

- 2) Elabore un variograma empírico apropiado para los datos de rinde en ambas campañas por separado. ¿Considera que existe estructura espacial en cada campaña? ¿El proceso espacial observado en cada caso es isotrópico o anisotrópico? ¿La dependencia espacial observada difiere entre campañas?

**Variograma empírico**

**Estructura espacial**

**Proceso espacial (isotropía y anisotropía)**

**Dependencia espacial entre campañas**