

continuas-sitio-web.R

r3172621

2025-06-02

```
# 1. Configurar directorio y cargar datos
setwd("/cloud/project/proyecto")

datos <- read.csv("archivo depurado nuevo 12.csv",
                  header = TRUE, sep = ";", dec = ".")

##### LIBRERÍAS #####
install.packages("e1071")

## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.4'
## (as 'lib' is unspecified)
install.packages("ggplot2")

## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.4'
## (as 'lib' is unspecified)
install.packages("tidyr")

## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.4'
## (as 'lib' is unspecified)
library(e1071)      # Para kurtosis y skewness
library(ggplot2)    # Para gráficas modernas
library(tidyr)      # Para pivot_longer

##### FUNCIONES #####

# Conversión de DMS a grados decimales
convertir_DMS_a_decimal <- function(dms) {
  dms <- gsub(",", ".", dms)
  partes <- strsplit(dms, ":")
  decimal <- sapply(partes, function(x) {
    grados <- as.numeric(x[1])
    minutos <- as.numeric(x[2])
    segundos <- as.numeric(x[3])
    signo <- ifelse(grados < 0, -1, 1)
    grados_abs <- abs(grados)
    grados_abs + minutos / 60 + segundos / 3600 * signo
  })
  return(decimal)
}

# Tabla de frecuencias
crear_tabla_frecuencias <- function(datos_numericos) {
```

```

n <- length(datos_numericos)
k <- ceiling(1 + 3.322 * log10(n))
rango <- range(datos_numericos)
amplitud <- ceiling((rango[2] - rango[1]) / k)
breaks <- seq(from = floor(rango[1]), to = ceiling(rango[2]) + amplitud, by = amplitud)
clases <- cut(datos_numericos, breaks = breaks, right = FALSE, include.lowest = TRUE)

tabla_frec <- as.data.frame(table(clases))
colnames(tabla_frec) <- c("intervalo", "ni")

marca_clase <- sapply(strsplit(gsub("\\[|\\]|\\]", "", tabla_frec$intervalo), ","), function(x) {
  mean(as.numeric(x))
})

tabla_frec$hi <- tabla_frec$ni / sum(tabla_frec$ni)
tabla_frec$ni_asc <- cumsum(tabla_frec$ni)
tabla_frec$hi_asc <- cumsum(tabla_frec$hi)
tabla_frec$ni_dsc <- rev(cumsum(rev(tabla_frec$ni)))
tabla_frec$hi_dsc <- rev(cumsum(rev(tabla_frec$hi)))
tabla_frec$mc <- round(marca_clase, 5)

tabla_frec[, c("intervalo", "mc", "ni", "hi", "ni_asc", "hi_asc", "ni_dsc", "hi_dsc")]
}

# Estadísticos descriptivos
estadisticos_desc <- function(x) {
  data.frame(
    Indicador = c("n", "Mínimo", "Máximo", "Media", "Mediana",
      "Desviación Estándar", "Varianza", "Coef. de Variación (%)",
      "Curtosis", "Asimetría"),
    Valor = round(c(length(x), min(x), max(x), mean(x), median(x),
      sd(x), var(x), (sd(x)/mean(x))*100,
      kurtosis(x), skewness(x)), 3)
  )
}

# Ojiva base
graficar_ojiva <- function(tabla_frec, titulo, xlab) {
  Li <- as.numeric(sub("\\[|\\]|\\]", "", sapply(strsplit(as.character(tabla_frec$intervalo), ","), `\[
  Ls <- as.numeric(sub("\\[|\\]|\\]", "", sapply(strsplit(as.character(tabla_frec$intervalo), ","), `\[
  Hi_asc <- tabla_frec$hi_asc * 100
  Hi_dsc <- tabla_frec$hi_dsc * 100

  plot(Li, Hi_dsc, type = "b", main = titulo, ylab = "Frecuencia Acumulada (%)",
    xlab = xlab, col = "green", pch = 19)
  lines(Ls, Hi_asc, type = "b", col = "blue", pch = 19)
  legend("bottomright", legend = c("Descendente", "Ascendente"), col = c("green", "blue"), lty = 1, pch
}

# Ojiva con ggplot2
graficar_ojiva_gg <- function(tabla_frec, titulo, xlab) {
  df <- data.frame(
    MC = tabla_frec$mc,

```

```

Ascendente = tabla_frec$hi_asc * 100,
Descendente = tabla_frec$hi_dsc * 100
)
df_long <- pivot_longer(df, cols = c("Ascendente", "Descendente"),
                        names_to = "Tipo", values_to = "Frecuencia_Acumulada")

ggplot(df_long, aes(x = MC, y = Frecuencia_Acumulada, color = Tipo)) +
  geom_line(size = 1) + geom_point(size = 2) +
  scale_color_manual(values = c("Ascendente" = "blue", "Descendente" = "red")) +
  labs(title = titulo, x = xlab, y = "Frecuencia_Acumulada (%)") +
  theme_minimal()
}

```

```

#===== DATOS =====

```

```

# Asegúrate de que DATOS esté cargado antes
# Ejemplo: DATOS <- read.csv("archivo.csv")

```

```

latitude_base <- convertir_DMS_a_decimal(as.character(datos$LATITUDE_BASE_4C))
latitude_base <- na.omit(latitude_base)

```

```

#===== ANÁLISIS =====

```

```

# Tabla de frecuencias
tabla_latitude <- crear_tabla_frecuencias(latitude_base)
print("Tabla de frecuencias LATITUDE_BASE_4C:")

```

```

## [1] "Tabla de frecuencias LATITUDE_BASE_4C:"

```

```

print(tabla_latitude)

```

```

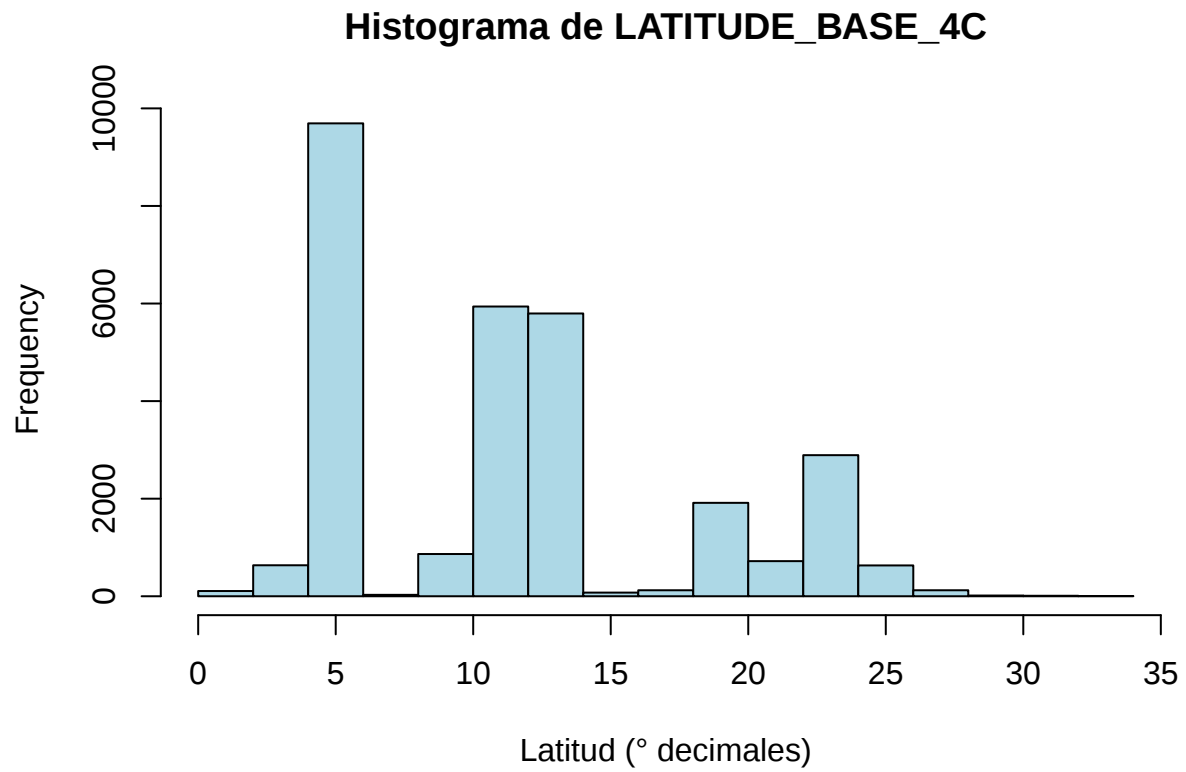
##      intervalo  mc    ni      hi ni_asc    hi_asc ni_dsc      hi_dsc
## 1      [0,3)  1.5   400 0.0135249366    400 0.01352494  29575 1.0000000000
## 2      [3,6)  4.5  10035 0.3393068470   10435 0.35283178  29175 0.9864750634
## 3      [6,9)  7.5    40 0.0013524937   10475 0.35418428  19140 0.6471682164
## 4      [9,12) 10.5  6794 0.2297210482   17269 0.58390533  19100 0.6458157227
## 5     [12,15) 13.5  5829 0.1970921386   23098 0.78099746  12306 0.4160946746
## 6     [15,18) 16.5   165 0.0055790363   23263 0.78657650   6477 0.2190025359
## 7     [18,21) 19.5  2052 0.0693829248   25315 0.85595943   6312 0.2134234996
## 8     [21,24) 22.5  3475 0.1174978867   28790 0.97345731   4260 0.1440405748
## 9     [24,27) 25.5   737 0.0249196957   29527 0.99837701    785 0.0265426881
## 10    [27,30) 28.5    33 0.0011158073   29560 0.99949281     48 0.0016229924
## 11    [30,33) 31.5    15 0.0005071851   29575 1.00000000    15 0.0005071851
## 12    [33,36] 34.5     0 0.0000000000   29575 1.00000000     0 0.0000000000

```

```

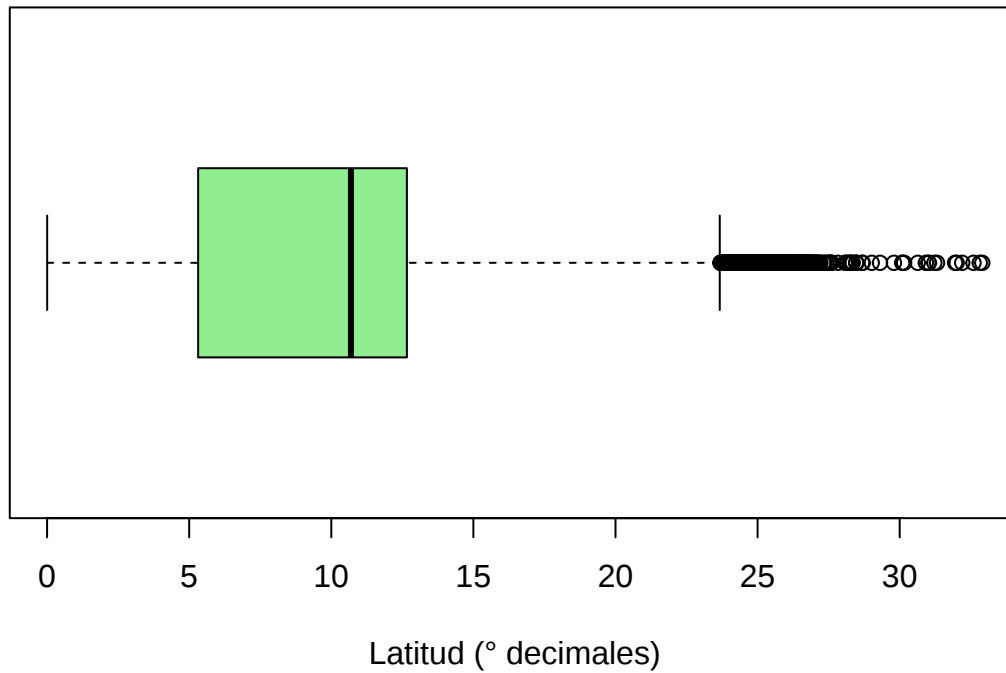
# Histograma
hist(latitude_base,
      main = "Histograma de LATITUDE_BASE_4C",
      xlab = "Latitud (° decimales)",
      col = "lightblue",
      border = "black")

```



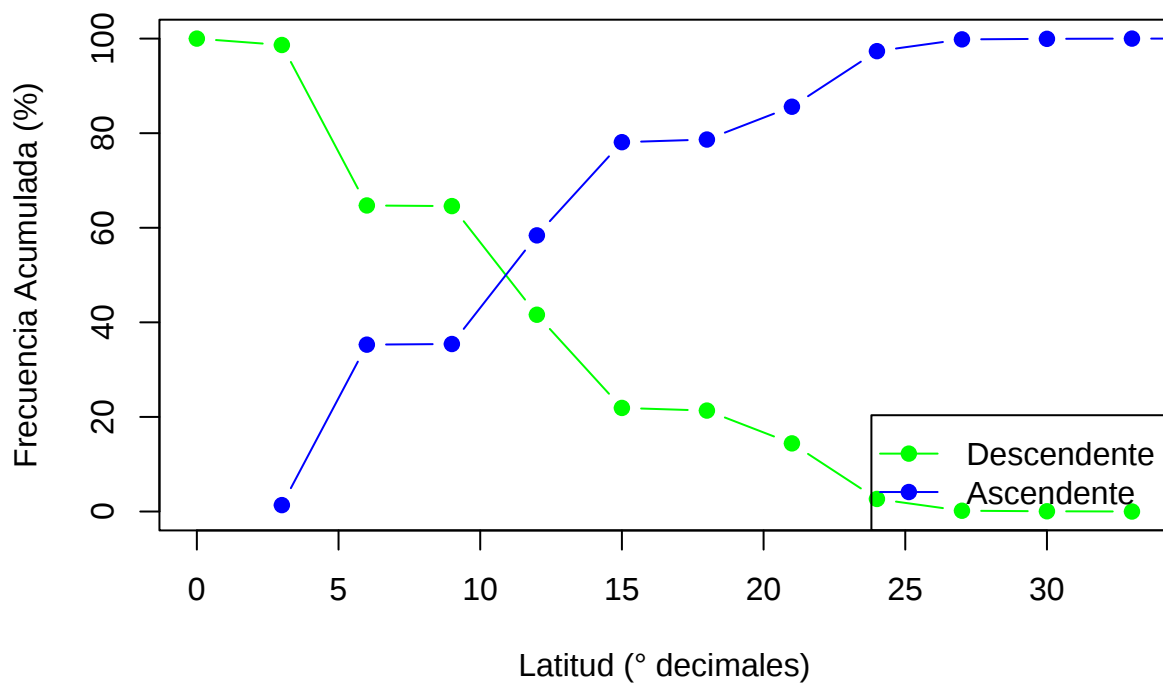
```
# Diagrama de caja
boxplot(latitude_base,
        horizontal = TRUE,
        col = "lightgreen",
        main = "Diagrama de Caja de LATITUDE_BASE_4C",
        xlab = "Latitud (° decimales)")
```

Diagrama de Caja de LATITUDE_BASE_4C



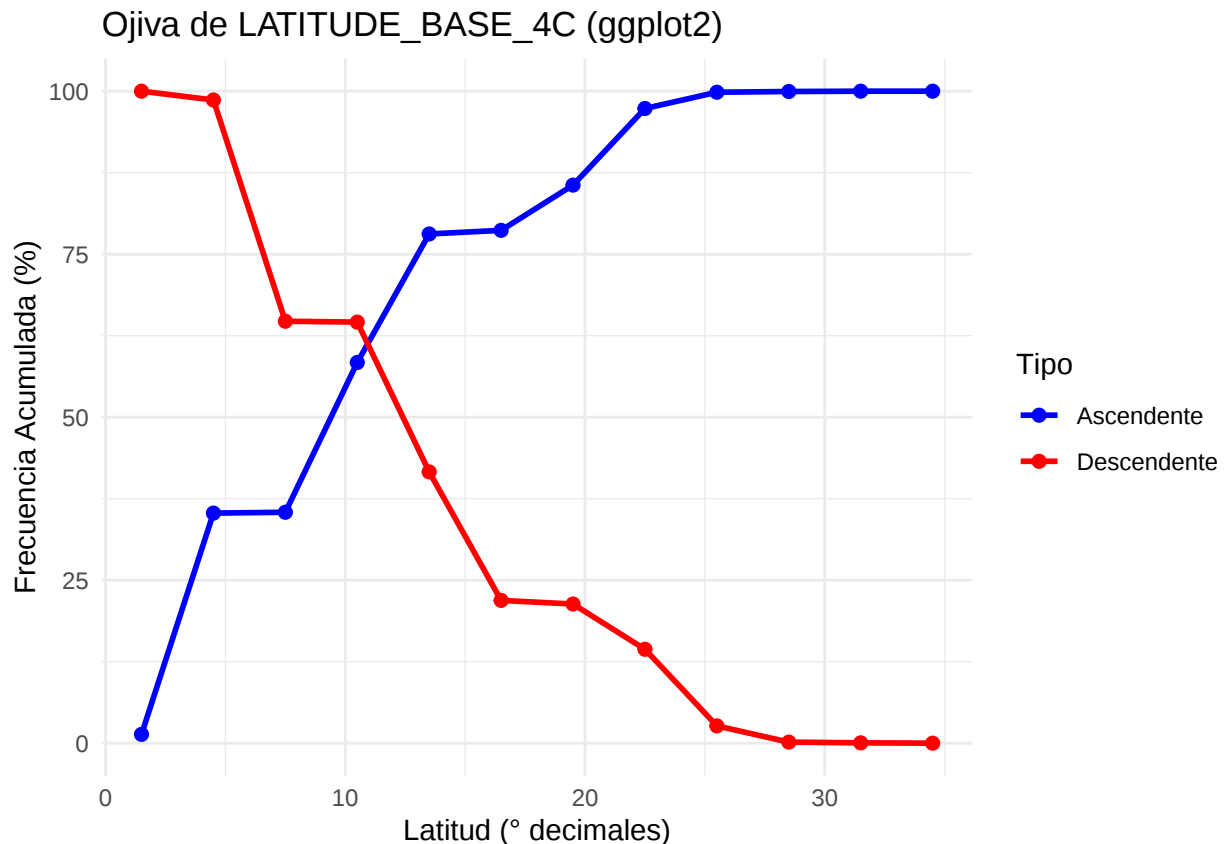
```
# Ojiva base
graficar_ojiva(tabla_latitudo,
               titulo = "Ojiva de Frecuencia Acumulada - LATITUDE_BASE_4C",
               xlab = "Latitud (° decimales)")
```

Ojiva de Frecuencia Acumulada – LATITUDE_BASE_4C



```
# Ojiva con ggplot2
print(graficar_ojiva_gg(tabla_latitude,
                        titulo = "Ojiva de LATITUDE_BASE_4C (ggplot2)",
                        xlab = "Latitud (° decimales)"))

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



```
# Estadísticos descriptivos
print("Estadísticos descriptivos LATITUDE_BASE_4C:")

## [1] "Estadísticos descriptivos LATITUDE_BASE_4C:"
print(estadisticos_desc(latitude_base))
```

##	Indicador	Valor
## 1	n	29575.000
## 2	Mínimo	0.003
## 3	Máximo	32.907
## 4	Media	11.418
## 5	Mediana	10.691
## 6	Desviación Estándar	6.246
## 7	Varianza	39.016
## 8	Coef. de Variación (%)	54.705
## 9	Curtosis	-0.545

```
## 10          Asimetría      0.687
#variable longitud base 4c

#===== DATOS - LONGITUD =====

longitude_base <- convertir_DMS_a_decimal(as.character(datos$LONGITUDE_BASE_4C))
longitude_base <- na.omit(longitude_base)

#===== ANÁLISIS =====

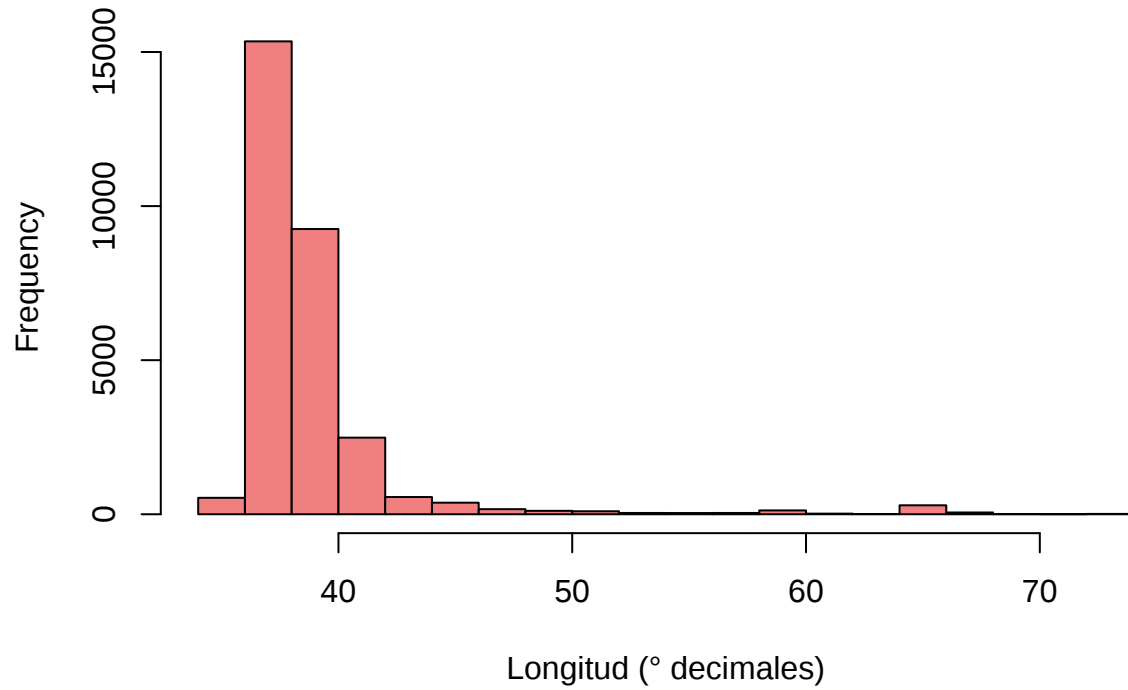
# Tabla de frecuencias
tabla_longitude <- crear_tabla_frecuencias(longitude_base)
print("Tabla de frecuencias LONGITUDE_BASE_4C:")

## [1] "Tabla de frecuencias LONGITUDE_BASE_4C:"
print(tabla_longitude)
```

##	intervalo	mc	ni	hi	ni_asc	hi_asc	ni_dsc	hi_dsc
## 1	[34,37)	35.5	7930	0.2681318681	7930	0.2681319	29575	1.0000000000
## 2	[37,40)	38.5	17205	0.5817413356	25135	0.8498732	21645	0.7318681319
## 3	[40,43)	41.5	2879	0.0973457312	28014	0.9472189	4440	0.1501267963
## 4	[43,46)	44.5	542	0.0183262891	28556	0.9655452	1561	0.0527810651
## 5	[46,49)	47.5	216	0.0073034658	28772	0.9728487	1019	0.0344547760
## 6	[49,52)	50.5	157	0.0053085376	28929	0.9781572	803	0.0271513102
## 7	[52,55)	53.5	59	0.0019949281	28988	0.9801522	646	0.0218427726
## 8	[55,58)	56.5	65	0.0021978022	29053	0.9823500	587	0.0198478445
## 9	[58,61)	59.5	141	0.0047675402	29194	0.9871175	522	0.0176500423
## 10	[61,64)	62.5	12	0.0004057481	29206	0.9875232	381	0.0128825021
## 11	[64,67)	65.5	330	0.0111580727	29536	0.9986813	369	0.0124767540
## 12	[67,70)	68.5	24	0.0008114962	29560	0.9994928	39	0.0013186813
## 13	[70,73)	71.5	7	0.0002366864	29567	0.9997295	15	0.0005071851
## 14	[73,76]	74.5	8	0.0002704987	29575	1.0000000	8	0.0002704987

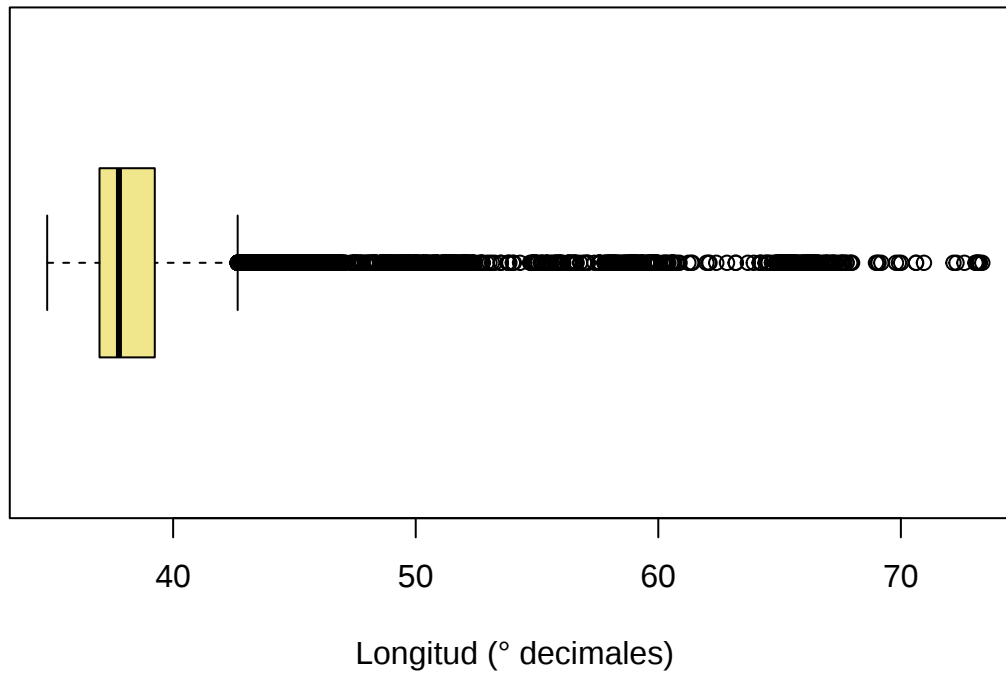
```
# Histograma
hist(longitude_base,
      main = "Histograma de LONGITUDE_BASE_4C",
      xlab = "Longitud (° decimales)",
      col = "lightcoral",
      border = "black")
```

Histograma de LONGITUDE_BASE_4C



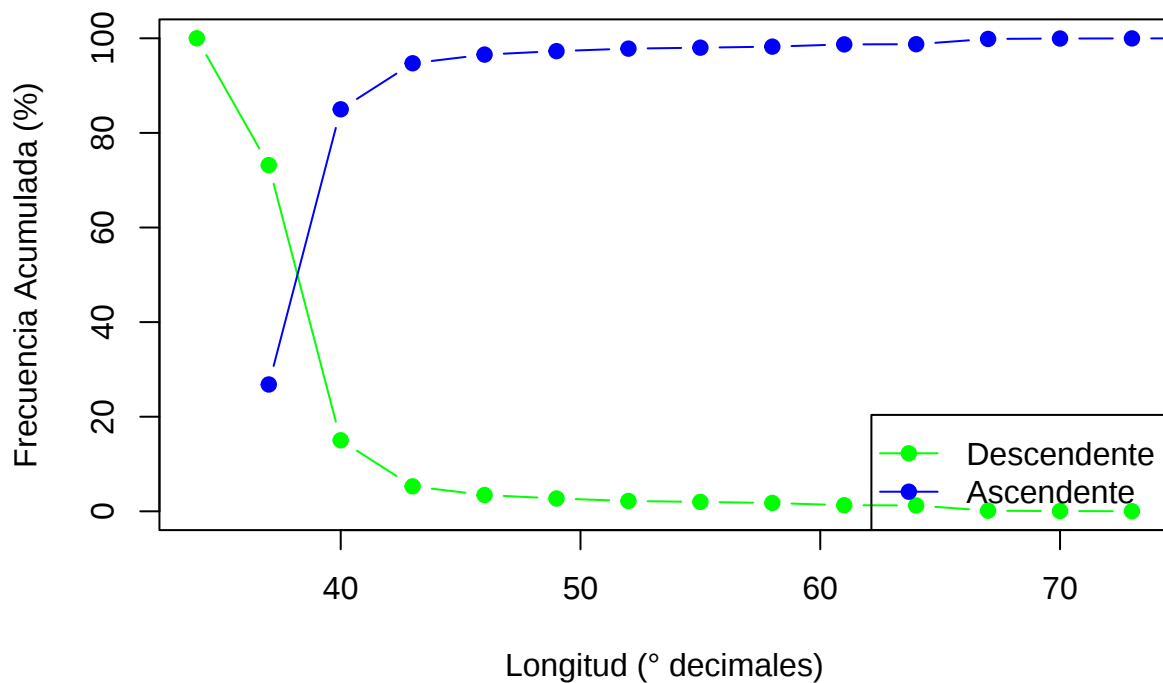
```
# Diagrama de caja
boxplot(longitude_base,
        horizontal = TRUE,
        col = "khaki",
        main = "Diagrama de Caja de LONGITUDE_BASE_4C",
        xlab = "Longitud (° decimales)")
```

Diagrama de Caja de LONGITUDE_BASE_4C

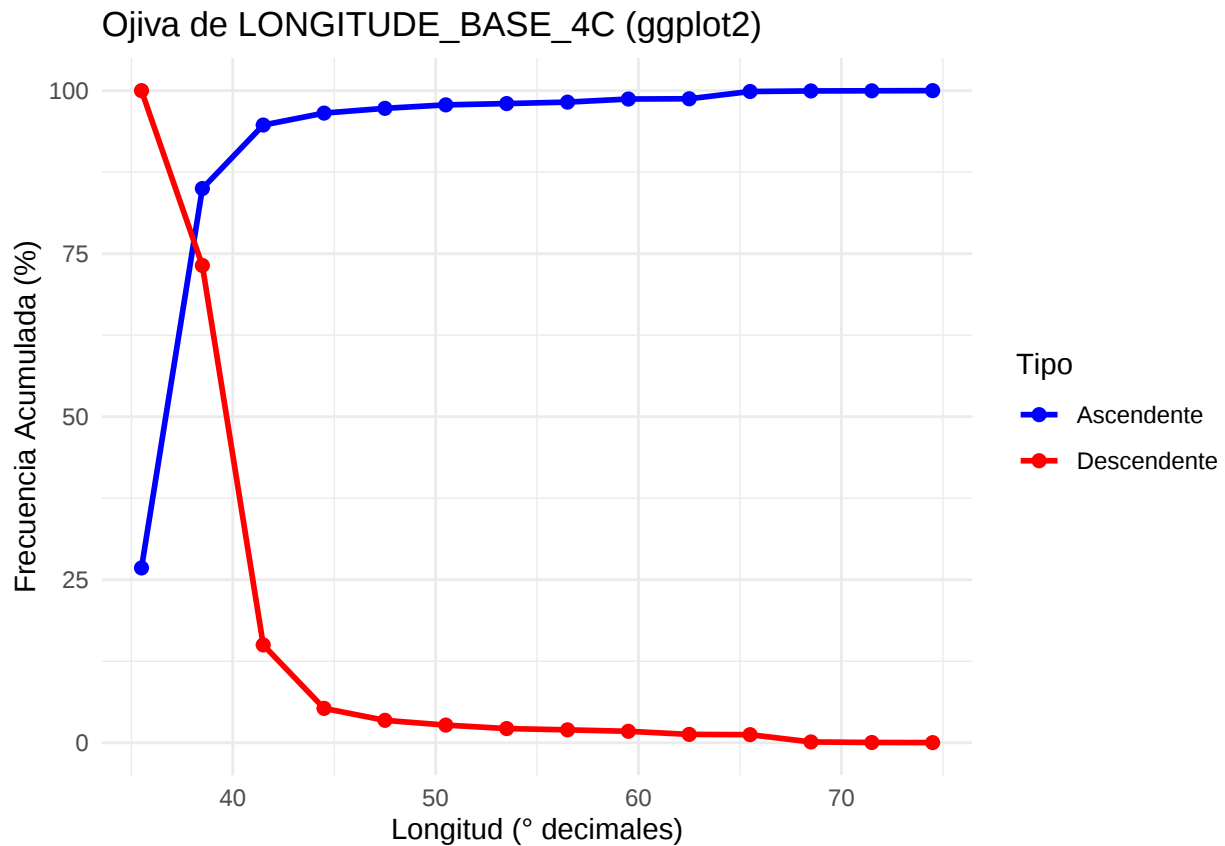


```
# Ojiva base
graficar_ojiva(tabla_longitude,
               titulo = "Ojiva de Frecuencia Acumulada - LONGITUDE_BASE_4C",
               xlab = "Longitud (° decimales)")
```

Ojiva de Frecuencia Acumulada – LONGITUDE_BASE_4C



```
# Ojiva con ggplot2
print(graficar_ojiva_gg(tabla_longitude,
  titulo = "Ojiva de LONGITUDE_BASE_4C (ggplot2)",
  xlab = "Longitud (° decimales)"))
```



```
# Estadísticos descriptivos
print("Estadísticos descriptivos LONGITUDE_BASE_4C:")

## [1] "Estadísticos descriptivos LONGITUDE_BASE_4C:"
print(estadisticos_desc(longitude_base))
```

##	Indicador	Valor
## 1	n	29575.000
## 2	Mínimo	34.807
## 3	Máximo	73.357
## 4	Media	38.783
## 5	Mediana	37.766
## 6	Desviación Estándar	4.088
## 7	Varianza	16.713
## 8	Coef. de Variación (%)	10.541
## 9	Curtosis	26.622
## 10	Asimetría	4.805

```
#variable latitud base dd

library(e1071)
library(ggplot2)
```

```

library(tidyr)

limpiar_y_convertir_num <- function(vec) {
  vec <- trimws(vec)
  vec <- gsub(",", ".", vec)
  vec_num <- as.numeric(vec)
  return(vec_num)
}

latitude_base_dd_raw <- datos$LATITUDE_BASE_DD
latitude_base_dd_num <- limpiar_y_convertir_num(latitude_base_dd_raw)

num_invalidos <- sum(is.na(latitude_base_dd_num) & !is.na(latitude_base_dd_raw))
cat("Número de valores no numéricos o inválidos en LATITUDE_BASE_DD:", num_invalidos, "\n")

## Número de valores no numéricos o inválidos en LATITUDE_BASE_DD: 0

latitude_base_dd <- na.omit(latitude_base_dd_num)

if(length(latitude_base_dd) == 0) {
  stop("No hay datos numéricos válidos en LATITUDE_BASE_DD después de la limpieza.")
}

tabla_latitude_dd <- crear_tabla_frecuencias(latitude_base_dd)
cat("Tabla de frecuencias LATITUDE_BASE_DD:\n")

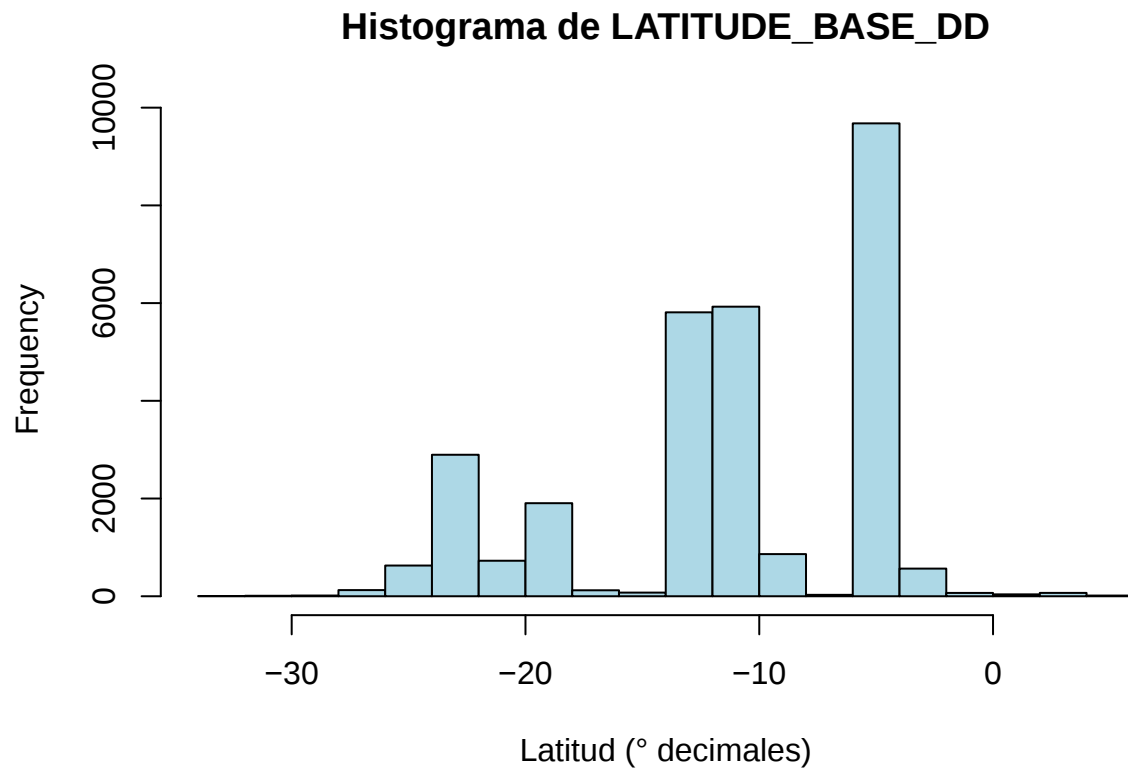
## Tabla de frecuencias LATITUDE_BASE_DD:

print(tabla_latitude_dd)

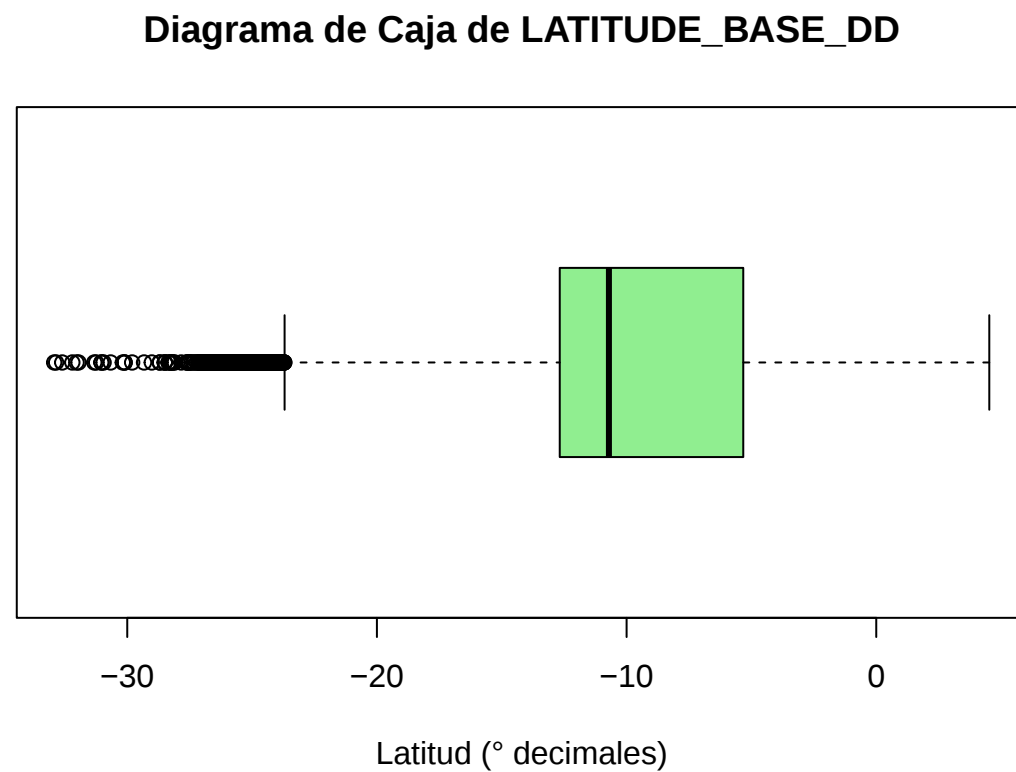
##   intervalo    mc   ni          hi ni_asc          hi_asc ni_dsc          hi_dsc
## 1  [-33,-30) -31.5   15 0.0005071851    15 0.0005071851  29575 1.0000000000
## 2  [-30,-27) -28.5   34 0.0011496196    49 0.0016568047  29560 0.999492815
## 3  [-27,-24) -25.5  736 0.0248858833   785 0.0265426881  29526 0.998343195
## 4  [-24,-21) -22.5 3476 0.1175316991  4261 0.1440743872  28790 0.973457312
## 5  [-21,-18) -19.5 2052 0.0693829248  6313 0.2134573119  25314 0.855925613
## 6  [-18,-15) -16.5  164 0.0055452240  6477 0.2190025359  23262 0.786542688
## 7  [-15,-12) -13.5 5844 0.1975993238 12321 0.4166018597  23098 0.780997464
## 8  [-12,-9)  -10.5 6779 0.2292138631 19100 0.6458157227  17254 0.583398140
## 9   [-9,-6)   -7.5   40 0.0013524937 19140 0.6471682164  10475 0.354184277
## 10  [-6,-3)   -4.5 9987 0.3376838546 29127 0.9848520710  10435 0.352831784
## 11   [-3,0)   -1.5  326 0.0110228233 29453 0.9958748943    448 0.015147929
## 12    [0,3)    1.5   64 0.0021639899 29517 0.9980388842    122 0.004125106
## 13    [3,6]    4.5   58 0.0019611158 29575 1.0000000000     58 0.001961116

hist(latitude_base_dd,
      main = "Histograma de LATITUDE_BASE_DD",
      xlab = "Latitud (° decimales)",
      col = "lightblue",
      border = "black")

```

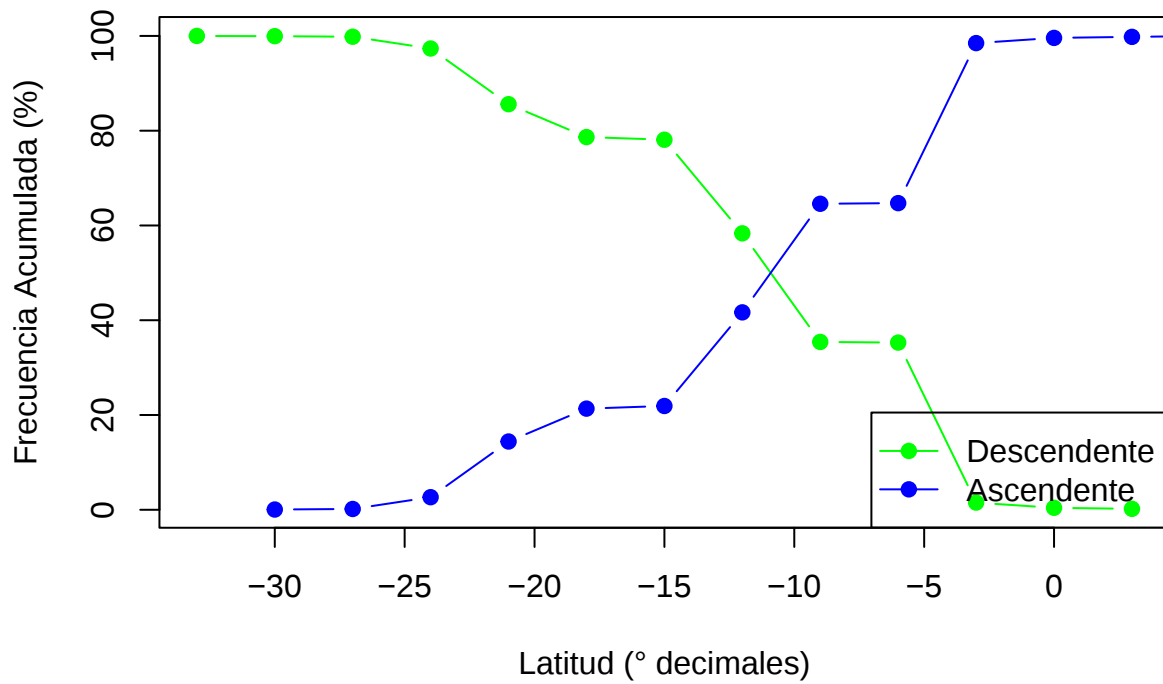


```
boxplot(latitude_base_dd,
  horizontal = TRUE,
  col = "lightgreen",
  main = "Diagrama de Caja de LATITUDE_BASE_DD",
  xlab = "Latitud (° decimales)")
```

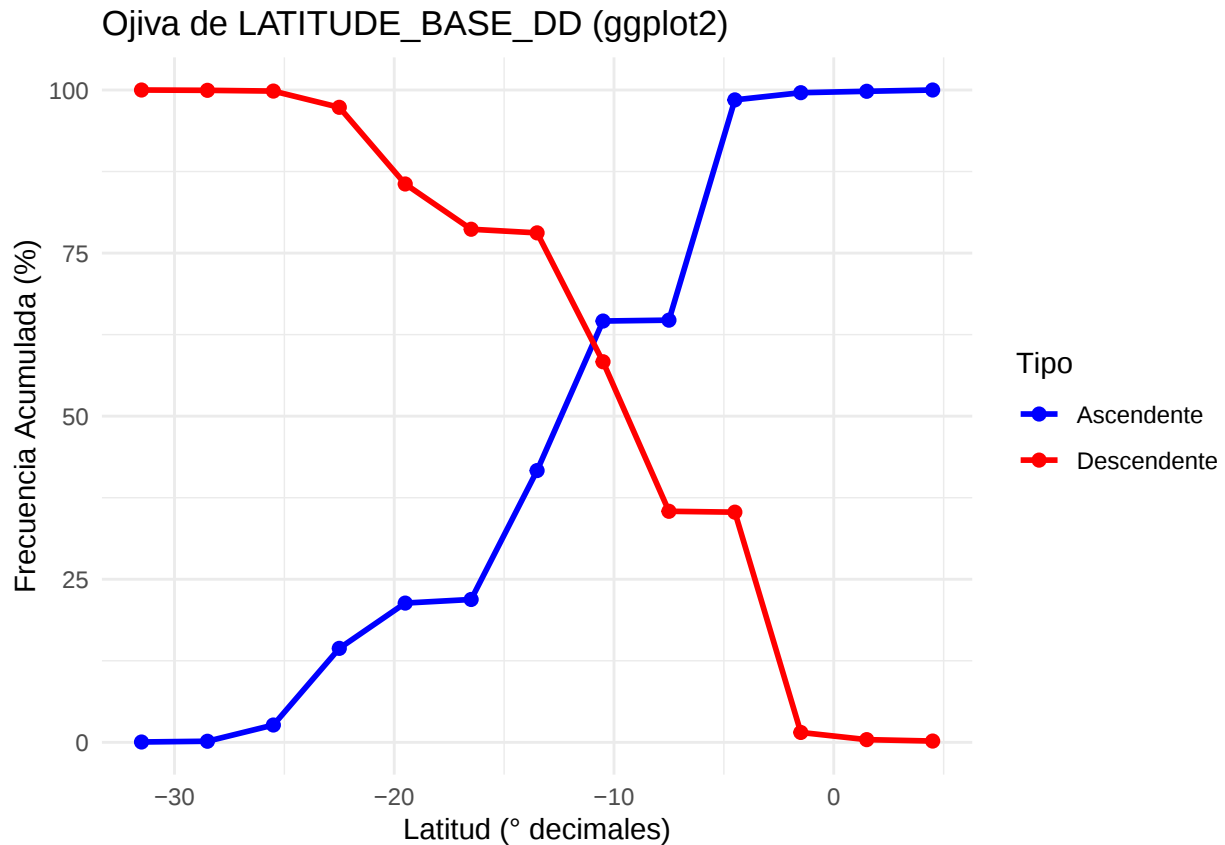


```
graficar_ojiva(tabla_latitude_dd,
  titulo = "Ojiva de Frecuencia Acumulada - LATITUDE_BASE_DD",
  xlab = "Latitud (° decimales)")
```

Ojiva de Frecuencia Acumulada – LATITUDE_BASE_DD



```
print(graficar_ojiva_gg(tabla_latitude_dd,
  titulo = "Ojiva de LATITUDE_BASE_DD (ggplot2)",
  xlab = "Latitud (° decimales)"))
```



```
cat("Estadísticos descriptivos LATITUDE_BASE_DD:\n")
```

```
## Estadísticos descriptivos LATITUDE_BASE_DD:
```

```
print(estadisticos_desc(latitude_base_dd))
```

```
##          Indicador      Valor
## 1              n 29575.000
## 2           Mínimo  -32.927
## 3           Máximo   4.528
## 4            Media  -11.414
## 5          Mediana  -10.708
## 6 Desviación Estándar   6.285
## 7          Varianza  39.499
## 8 Coef. de Variación (%) -55.064
## 9          Curtosis  -0.509
## 10         Asimetría  -0.650
```

```
#variables de longitud base dd
```

```
longitude_base_dd_raw <- datos$LONGITUDE_BASE_DD
```

```
longitude_base_dd_num <- limpiar_y_convertir_num(longitude_base_dd_raw)
```

```
num_invalidos <- sum(is.na(longitude_base_dd_num) & !is.na(longitude_base_dd_raw))
```

```
cat("Número de valores no numéricos o inválidos en LONGITUDE_BASE_DD:", num_invalidos, "\n")
```

```
## Número de valores no numéricos o inválidos en LONGITUDE_BASE_DD: 0
```

```

longitude_base_dd <- na.omit(longitude_base_dd_num)

if(length(longitude_base_dd) == 0) {
  stop("No hay datos numéricos válidos en LONGITUDE_BASE_DD después de la limpieza.")
}

tabla_longitude_dd <- crear_tabla_frecuencias(longitude_base_dd)
cat("Tabla de frecuencias LONGITUDE_BASE_DD:\n")

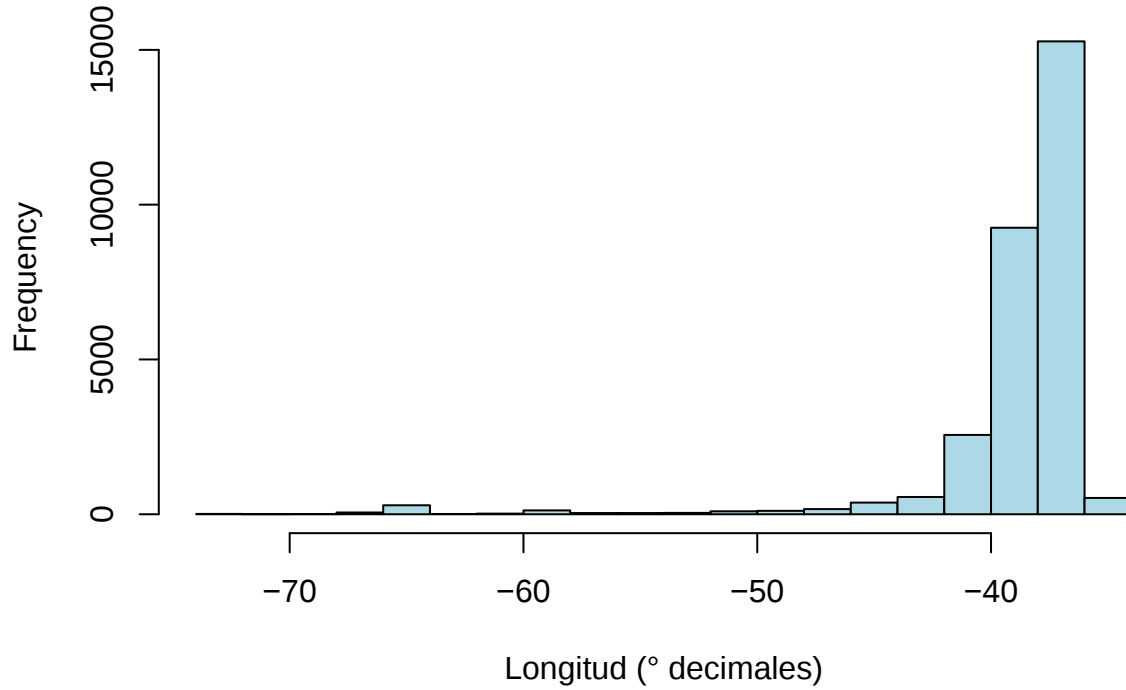
## Tabla de frecuencias LONGITUDE_BASE_DD:
print(tabla_longitude_dd)

##      intervalo      mc      ni      hi ni_asc      hi_asc ni_dsc      hi_dsc
## 1  [-74,-71) -72.5      11 0.0003719358      11 0.0003719358 29575 1.000000000
## 2  [-71,-68) -69.5      11 0.0003719358      22 0.0007438715 29564 0.999628064
## 3  [-68,-65) -66.5     332 0.0112256974     354 0.0119695689 29553 0.999256128
## 4  [-65,-62) -63.5      24 0.0008114962     378 0.0127810651 29221 0.988030431
## 5  [-62,-59) -60.5      89 0.0030092984     467 0.0157903635 29197 0.987218935
## 6  [-59,-56) -57.5      97 0.0032797971     564 0.0190701606 29108 0.984209637
## 7  [-56,-53) -54.5      48 0.0016229924     612 0.0206931530 29011 0.980929839
## 8  [-53,-50) -51.5     132 0.0044632291     744 0.0251563821 28963 0.979306847
## 9  [-50,-47) -48.5     137 0.0046322908     881 0.0297886729 28831 0.974843618
## 10 [-47,-44) -45.5     516 0.0174471682    1397 0.0472358411 28694 0.970211327
## 11 [-44,-41) -42.5     844 0.0285376162    2241 0.0757734573 28178 0.952764159
## 12 [-41,-38) -39.5   11532 0.3899239222   13773 0.4656973795 27334 0.924226543
## 13 [-38,-35) -36.5   15799 0.5342011834   29572 0.9998985630 15802 0.534302620
## 14 [-35,-32] -33.5       3 0.0001014370   29575 1.0000000000       3 0.000101437

hist(longitude_base_dd,
      main = "Histograma de LONGITUDE_BASE_DD",
      xlab = "Longitud (° decimales)",
      col = "lightblue",
      border = "black")

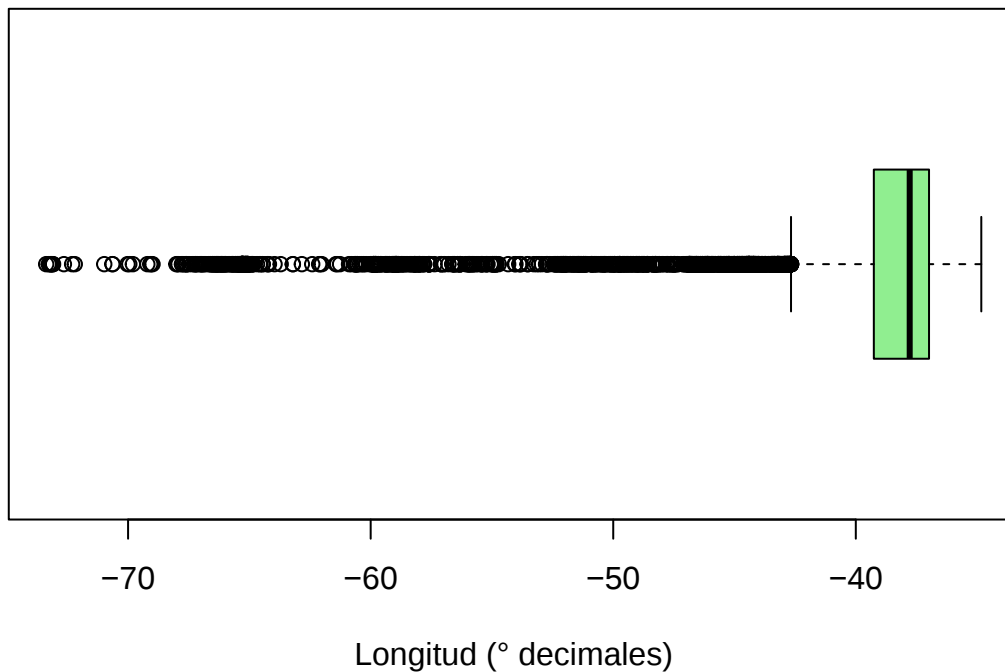
```

Histograma de LONGITUDE_BASE_DD



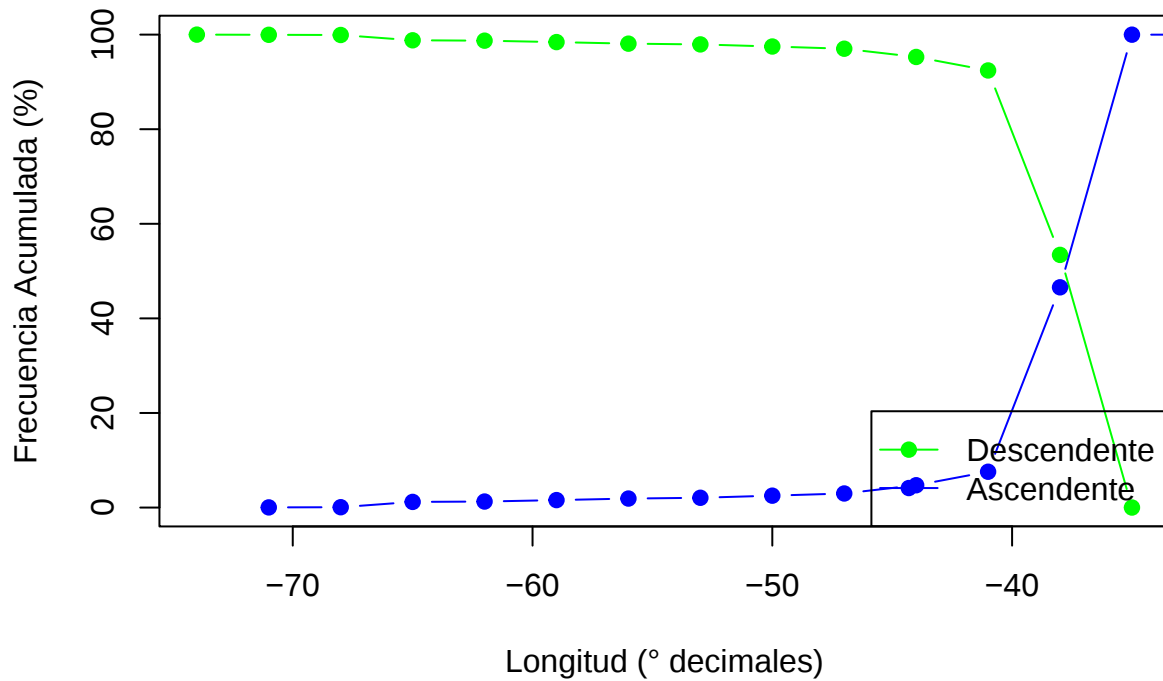
```
boxplot(longitude_base_dd,  
        horizontal = TRUE,  
        col = "lightgreen",  
        main = "Diagrama de Caja de LONGITUDE_BASE_DD",  
        xlab = "Longitud (° decimales)")
```

Diagrama de Caja de LONGITUDE_BASE_DD

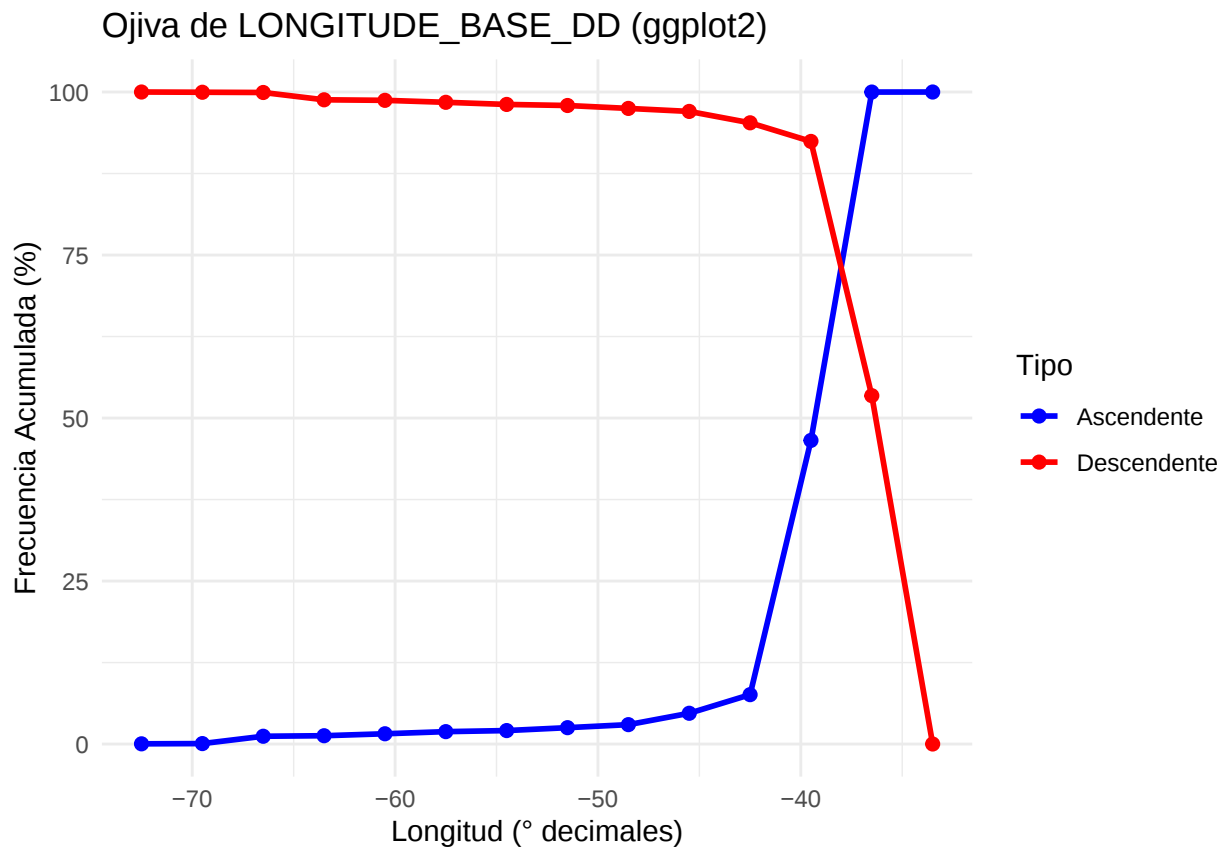


```
graficar_ojiva(tabla_longitude_dd,
               titulo = "Ojiva de Frecuencia Acumulada - LONGITUDE_BASE_DD",
               xlab = "Longitud (° decimales)")
```

Ojiva de Frecuencia Acumulada - LONGITUDE_BASE_DD



```
print(graficar_ojiva_gg(tabla_longitude_dd,
                        titulo = "Ojiva de LONGITUDE_BASE_DD (ggplot2)",
                        xlab = "Longitud (° decimales)"))
```



```
cat("Estadísticos descriptivos LONGITUDE_BASE_DD:\n")
```

```
## Estadísticos descriptivos LONGITUDE_BASE_DD:
```

```
print(estadisticos_desc(longitude_base_dd))
```

```
##          Indicador      Valor
## 1              n 29575.000
## 2           Mínimo   -73.377
## 3           Máximo   -34.826
## 4            Media   -38.800
## 5          Mediana   -37.778
## 6 Desviación Estándar    4.088
## 7          Varianza   16.713
## 8 Coef. de Variación (%) -10.536
## 9          Curtosis   26.621
## 10         Asimetría    -4.805
```

```
#variable sondador
```

```
profundidade_sondado_rm <- as.numeric(datos$PROFUNDIDADE_SONDADOR_M)
```

```
## Warning: NAs introduced by coercion
```

```
profundidade_sondado_rm <- na.omit(profundidade_sondado_rm)
```

```
if(length(profundidade_sondado_rm) == 0){
  stop("La variable 'profundidade_sondado_rm' no tiene datos numéricos válidos o está vacía.")
}
```

```

}

n_intervals <- 12
breaks <- pretty(range(profundidade_sondado_rm), n = n_intervals)

intervalos <- cut(profundidade_sondado_rm,
                  breaks = breaks,
                  include.lowest = TRUE,
                  right = FALSE)

tabla_frecuencias <- table(intervalos)

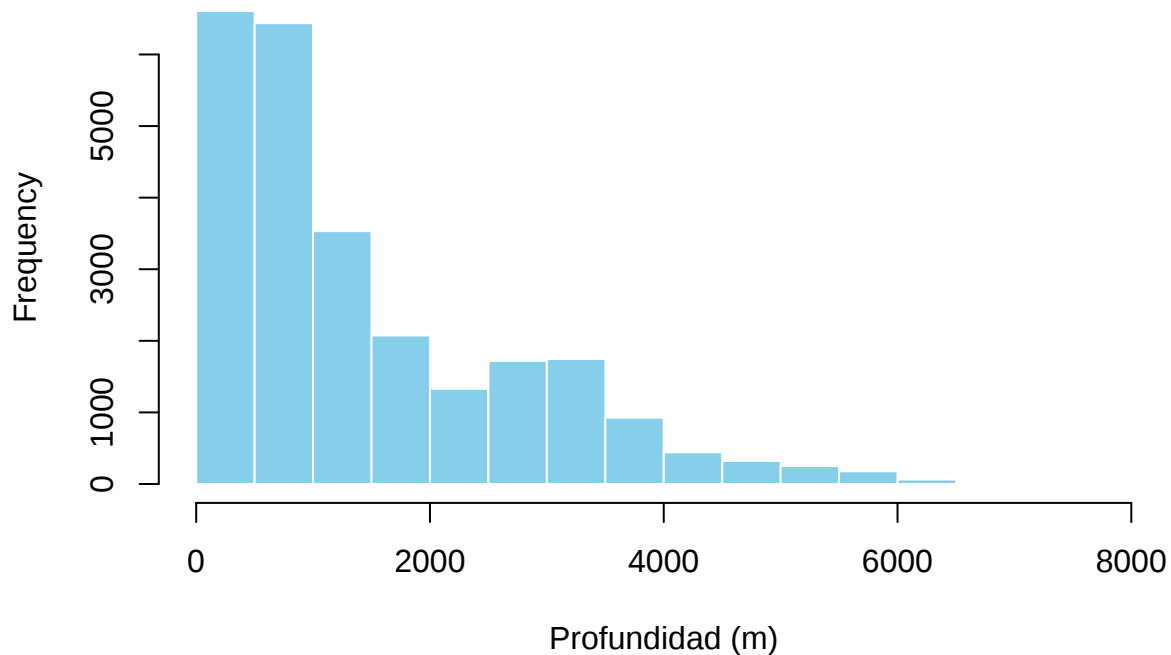
print(tabla_frecuencias)

## intervalos
##          [0,500)      [500,1e+03) [1e+03,1.5e+03) [1.5e+03,2e+03) [2e+03,2.5e+03)
##             6565             6465             3513             2090             1329
## [2.5e+03,3e+03) [3e+03,3.5e+03) [3.5e+03,4e+03) [4e+03,4.5e+03) [4.5e+03,5e+03)
##             1707             1763             929             446             324
## [5e+03,5.5e+03) [5.5e+03,6e+03) [6e+03,6.5e+03) [6.5e+03,7e+03) [7e+03,7.5e+03)
##             253             181             64             21             8
## [7.5e+03,8e+03]
##              2

hist(profundidade_sondado_rm,
     breaks = breaks,
     main = "Histograma de profundidade_sondado_rm",
     xlab = "Profundidad (m)",
     col = "skyblue",
     border = "white")

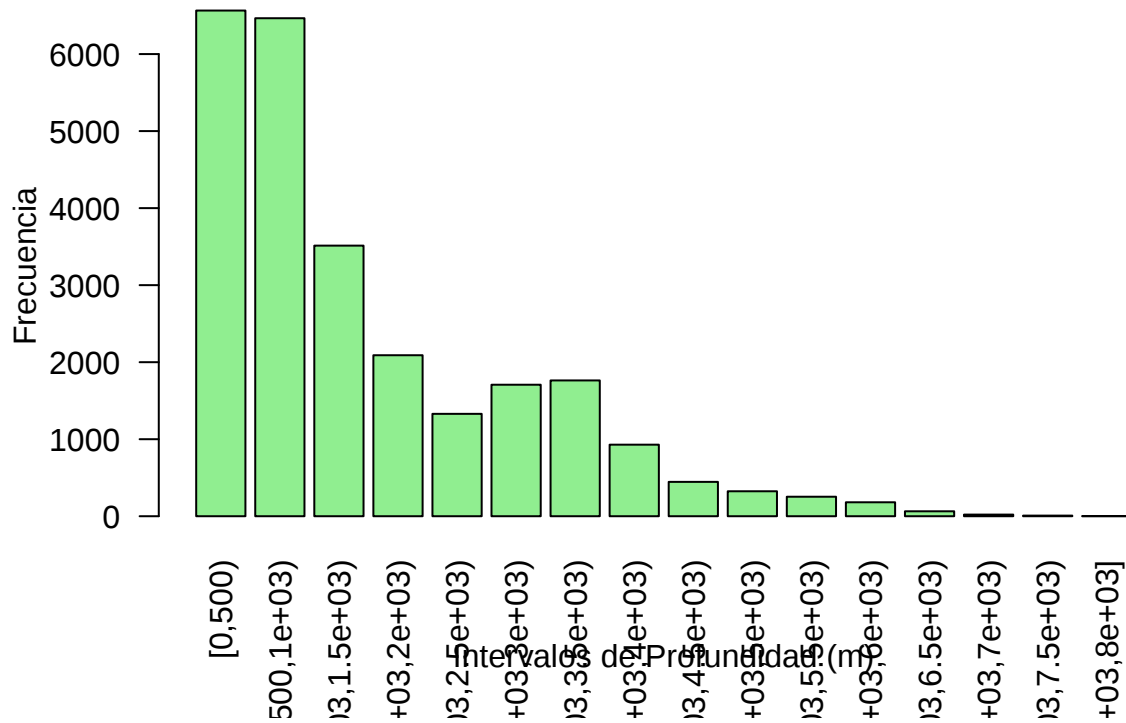
```

Histograma de profundidade_sondado_rm



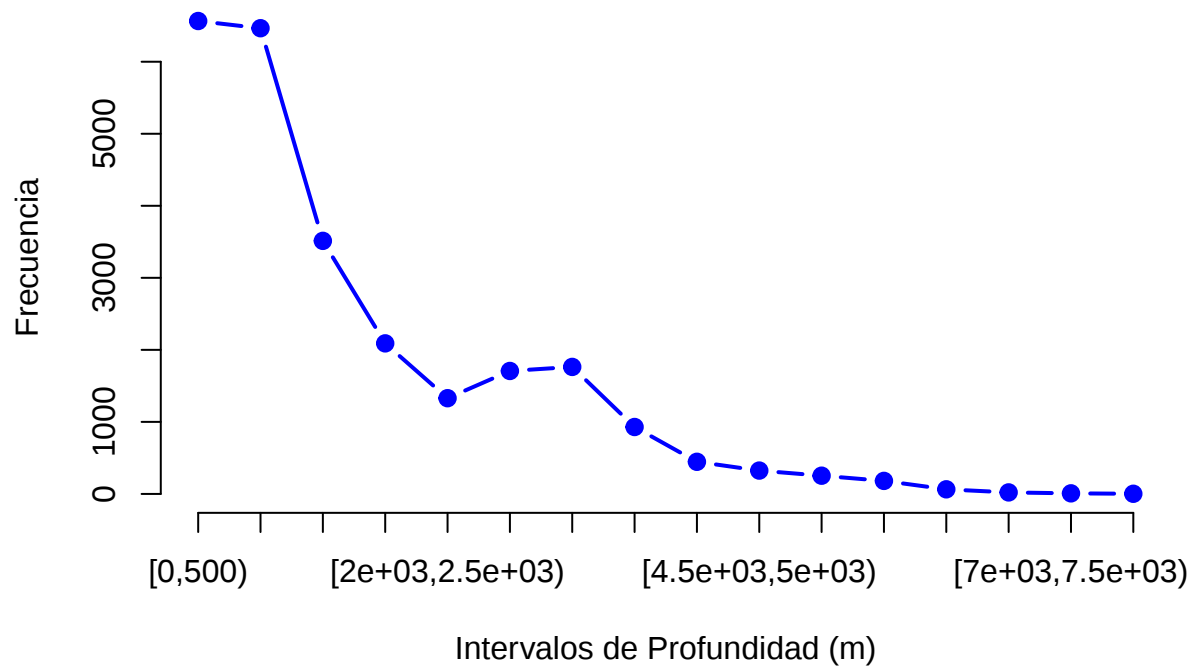
```
barplot(tabla_frecuencias,
        main = "Diagrama de Barras - profundidad_sondado_rm (Agrupada)",
        xlab = "Intervalos de Profundidad (m)",
        ylab = "Frecuencia",
        col = "lightgreen",
        border = "black",
        las = 2)
```

Diagrama de Barras – profundidad_sondado_rm (Agrupada)



```
plot(tabla_frecuencias,
      type = "b",
      main = "Gráfico de Frecuencia - profundidad_sondado_rm (Agrupada)",
      xlab = "Intervalos de Profundidad (m)",
      ylab = "Frecuencia",
      col = "blue",
      pch = 19)
```

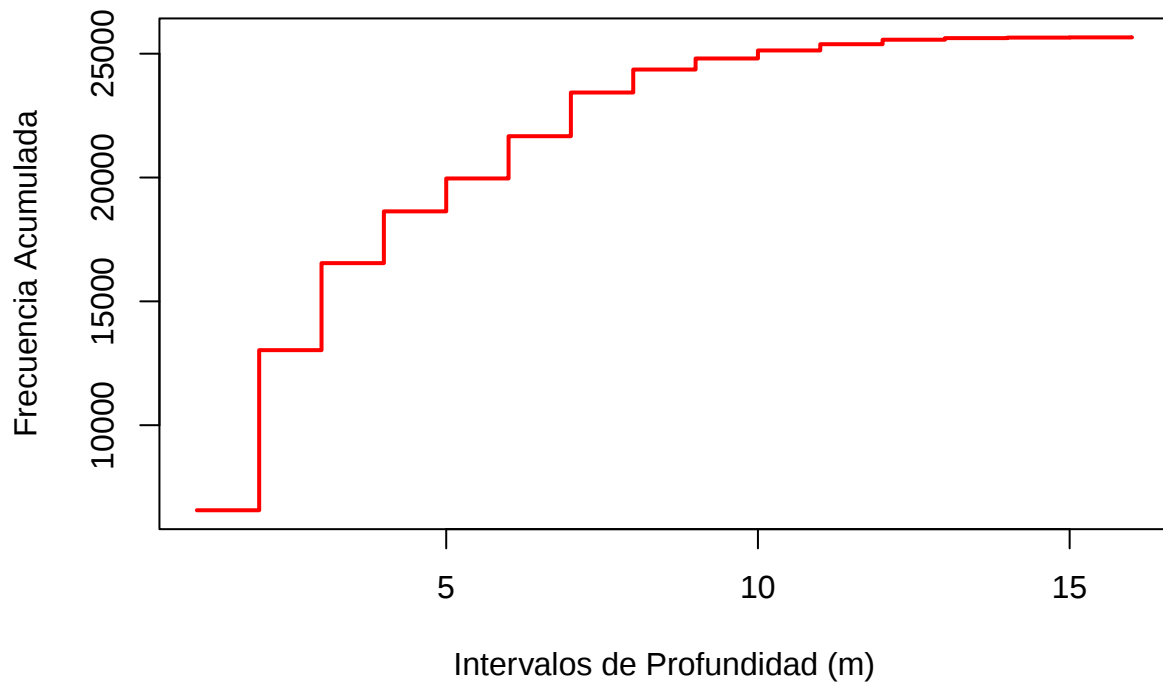
Gráfico de Frecuencia – profundidad_sondado_rm (Agrupada)



```
ojiva <- cumsum(tabla_frecuencias)

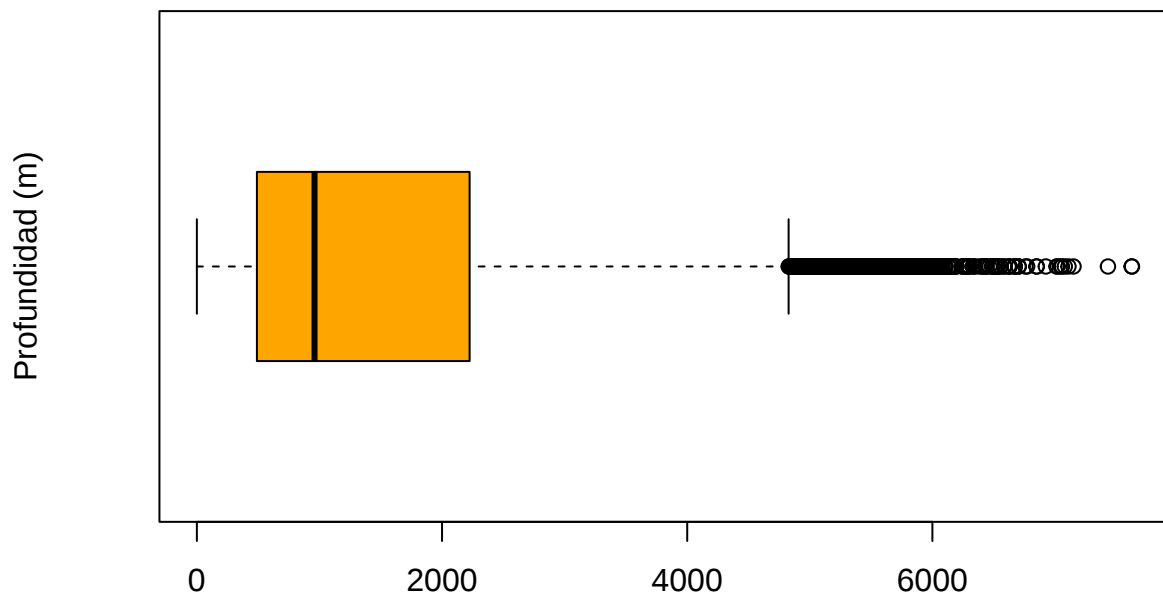
plot(ojiva,
     type = "s",
     main = "Ojiva de profundidad_sondado_rm",
     xlab = "Intervalos de Profundidad (m)",
     ylab = "Frecuencia Acumulada",
     col = "red",
     lwd = 2)
```

Ojiva de profundidad_sondado_rm



```
boxplot(profundidade_sondado_rm,  
  main = "Diagrama de Caja - profundidade_sondado_rm",  
  ylab = "Profundidad (m)",  
  col = "orange",  
  horizontal = TRUE)
```

Diagrama de Caja – profundidade_sondado_rm



```
#profundidade media en m
```

```

profundidade_medida_m <- as.numeric(datos$PROFUNDIDADE_MEDIDA_M)

## Warning: NAs introduced by coercion
profundidade_medida_m <- na.omit(profundidade_medida_m)

if(length(profundidade_medida_m) == 0){
  stop("La variable 'profundidade_medida_m' no tiene datos numéricos válidos o está vacía.")
}

n_intervals <- 12
breaks <- pretty(range(profundidade_medida_m), n = n_intervals)

intervalos <- cut(profundidade_medida_m,
                  breaks = breaks,
                  include.lowest = TRUE,
                  right = FALSE)

tabla_frecuencias <- table(intervalos)

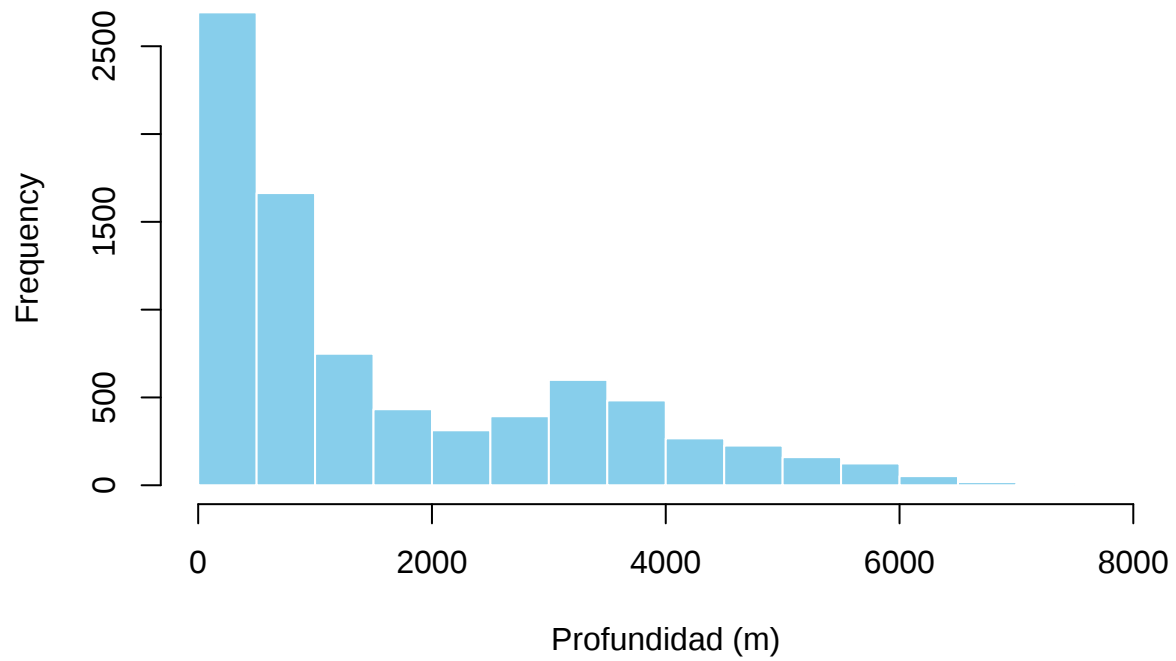
print(tabla_frecuencias)

## intervalos
##      [0,500)      [500,1e+03) [1e+03,1.5e+03) [1.5e+03,2e+03) [2e+03,2.5e+03)
##           2681           1672           744           440           310
## [2.5e+03,3e+03) [3e+03,3.5e+03) [3.5e+03,4e+03) [4e+03,4.5e+03) [4.5e+03,5e+03)
##           391           603           484           264           231
## [5e+03,5.5e+03) [5.5e+03,6e+03) [6e+03,6.5e+03) [6.5e+03,7e+03) [7e+03,7.5e+03)
##           161           124           52           18           6
## [7.5e+03,8e+03]
##           1

hist(profundidade_medida_m,
     breaks = breaks,
     main = "Histograma de profundidade_medida_m",
     xlab = "Profundidad (m)",
     col = "skyblue",
     border = "white")

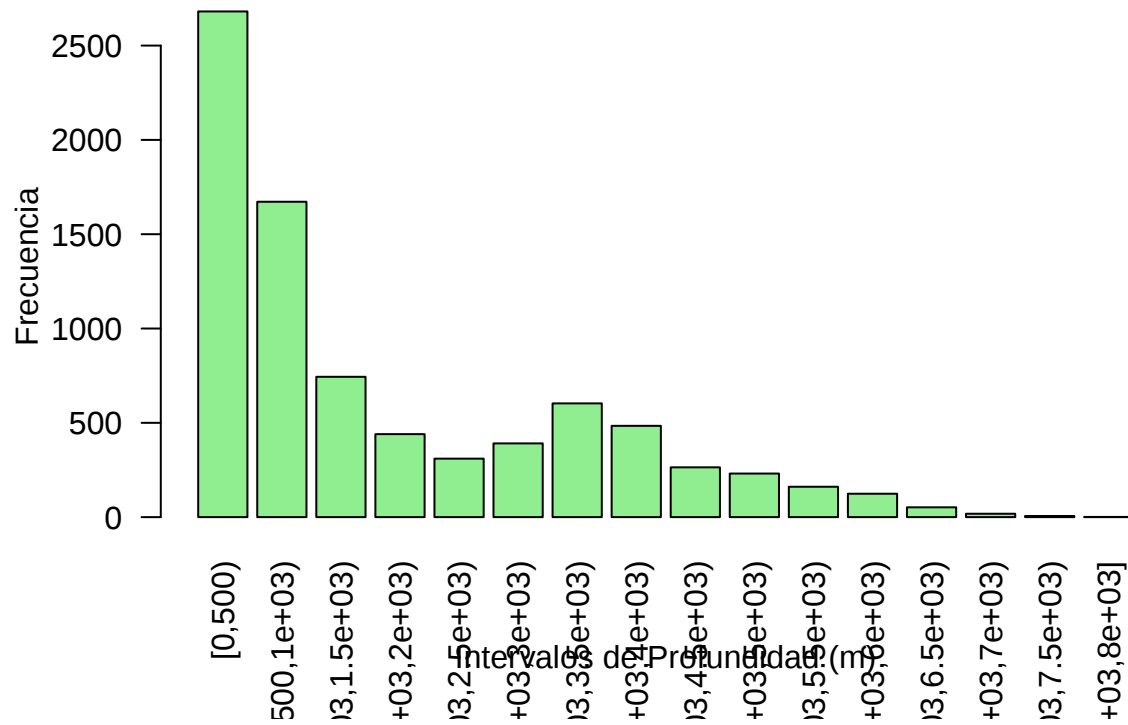
```

Histograma de profundidad_medida_m



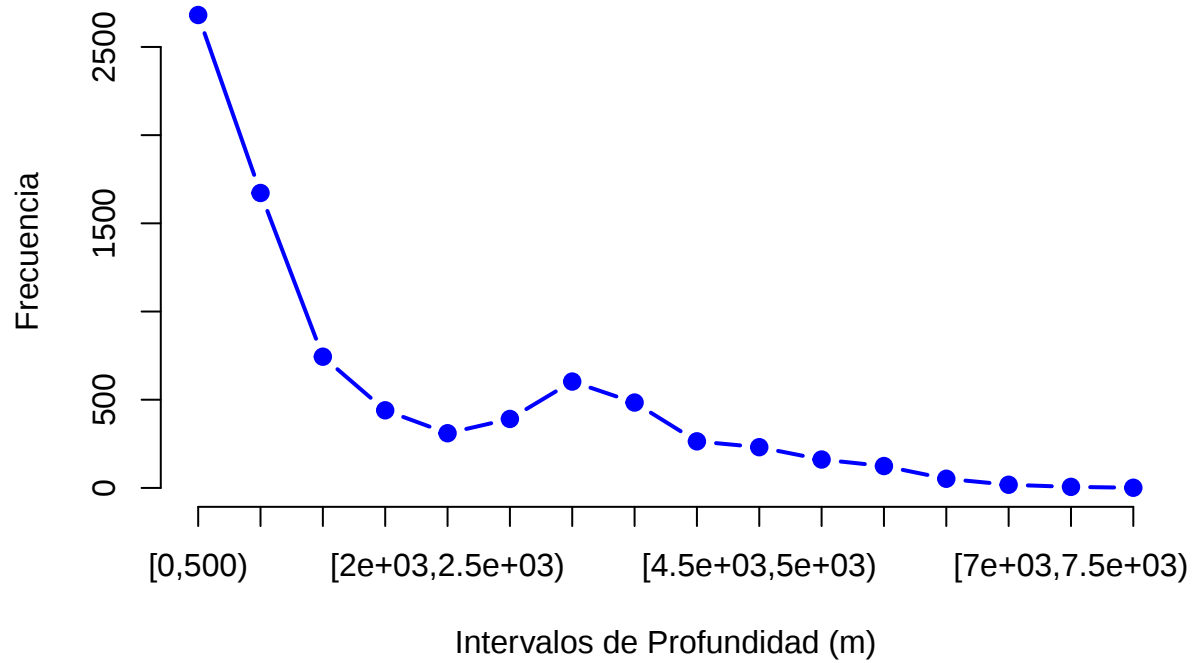
```
barplot(tabla_frecuencias,  
  main = "Diagrama de Barras - profundidad_medida_m (Agrupada)",  
  xlab = "Intervalos de Profundidad (m)",  
  ylab = "Frecuencia",  
  col = "lightgreen",  
  border = "black",  
  las = 2)
```

Diagrama de Barras – profundidad_medida_m (Agrupada)



```
plot(tabla_frecuencias,
     type = "b",
     main = "Gráfico de Frecuencia - profundidad_medida_m (Agrupada)",
     xlab = "Intervalos de Profundidad (m)",
     ylab = "Frecuencia",
     col = "blue",
     pch = 19)
```

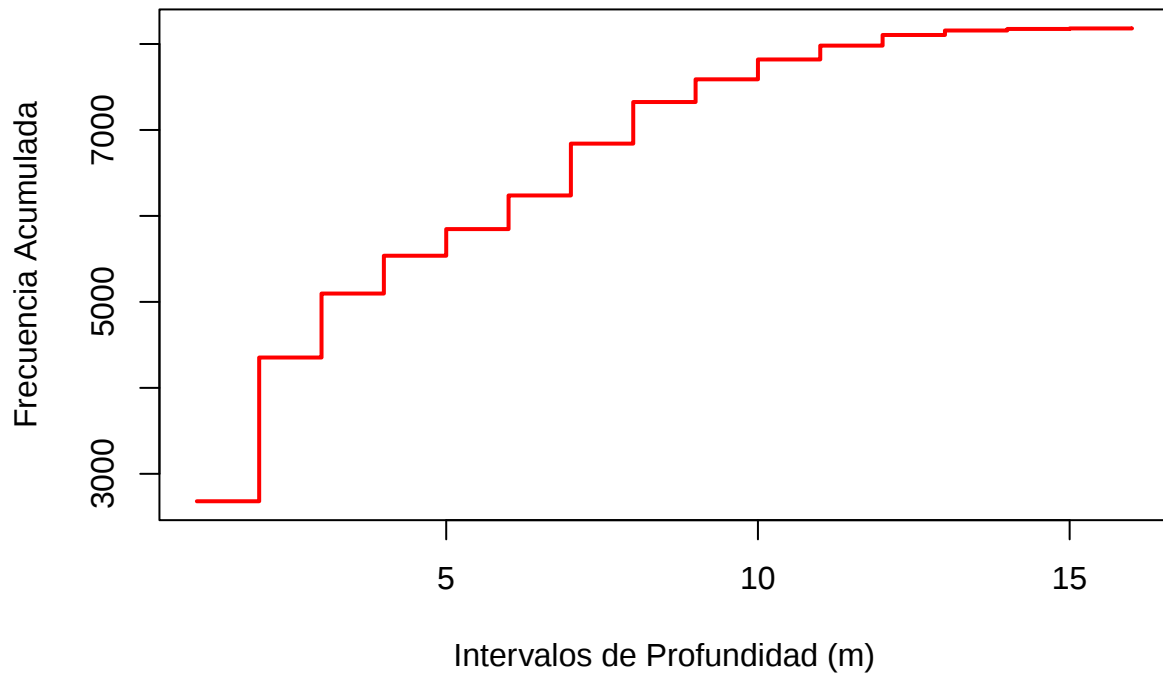
Gráfico de Frecuencia – profundidad_medida_m (Agrupada)



```
ojiva <- cumsum(tabla_frecuencias)

plot(ojiva,
     type = "s",
     main = "Ojiva de profundidad_medida_m",
     xlab = "Intervalos de Profundidad (m)",
     ylab = "Frecuencia Acumulada",
     col = "red",
     lwd = 2)
```

Ojiva de profundidad_medida_m



```
boxplot(profundidade_medida_m,  
  main = "Diagrama de Caja - profundidade_medida_m",  
  ylab = "Profundidad (m)",  
  col = "orange",  
  horizontal = TRUE)
```

Diagrama de Caja – profundidade_medida_m

