

## CÓDIGO EN R

```
#Lectura de la base de datos principal.
Sonido <- read.csv2("Base de Datos TF.csv", na.string = ".")

#Nuevas variables para la base de datos principal.

#Variable de Media.
Sonido$Mean_dB <- with(Sonido, (MAX_dB + MIN_dB)/2)

#Variable de Piso 1 y el piso 2.
Sonido$PISO <- with(Sonido, ifelse(LUGAR == "Lu_1" | LUGAR == "Lu_2", "PISO_2", "PISO_1"))

#Resultados y resúmenes
str(Sonido)
summary(Sonido)
attach(Sonido)

# Filtros para analizar normalidad
Piso_1 <- droplevels(subset(Sonido, PISO == "PISO_1"))

Piso_2 <- droplevels(subset(Sonido, PISO == "PISO_2"))

Martes <- droplevels(subset(Sonido, DIA == "MARTES"))

Miercoles <- droplevels(subset(Sonido, DIA == "MIERCOLES"))

Jueves <- droplevels(subset(Sonido, DIA == "JUEVES"))

Viernes <- droplevels(subset(Sonido, DIA == "VIERNES"))


Lugar_1 <- droplevels(subset(Sonido, LUGAR == "Lu_1"))

Lugar_2 <- droplevels(subset(Sonido, LUGAR == "Lu_2"))

Lugar_3 <- droplevels(subset(Sonido, LUGAR == "Lu_3"))

Lugar_4 <- droplevels(subset(Sonido, LUGAR == "Lu_4"))

Lugar_5 <- droplevels(subset(Sonido, LUGAR == "Lu_5"))

Semana_1 <- droplevels(subset(Sonido, SEMANA == "1"))

Semana_2 <- droplevels(subset(Sonido, SEMANA == "2"))

#Ajuste a la Normal, a estos se los nombramos más adelante como "estadísticos"
require(MASS)
require(car)

ajuste.Mean <- fitdistr(Sonido$Mean_dB, "normal")
```

```

ajuste.piso1 <- fitdistr(Piso_1$Mean_dB, "normal")
ajuste.piso2 <- fitdistr(Piso_2$Mean_dB, "normal")
ajuste.mart <- fitdistr(Martes$Mean_dB, "normal")
ajuste.mier <- fitdistr(Miercoles$Mean_dB, "normal")
ajuste.juev <- fitdistr(Jueves$Mean_dB, "normal")
ajuste.vier <- fitdistr(Viernes$Mean_dB, "normal")
ajuste.lugar1 <- fitdistr(Lugar_1$Mean_dB, "normal")
ajuste.lugar2 <- fitdistr(Lugar_2$Mean_dB, "normal")
ajuste.lugar3 <- fitdistr(Lugar_3$Mean_dB, "normal")
ajuste.lugar4 <- fitdistr(Lugar_4$Mean_dB, "normal")
ajuste.lugar5 <- fitdistr(Lugar_5$Mean_dB, "normal")
ajuste.semana1 <- fitdistr(Semana_1$Mean_dB, "normal")
ajuste.semana2 <- fitdistr(Semana_2$Mean_dB, "normal")
#Función para graficar densidades respecto a la teórica normal.
dibujarDensidades <- function(estadisticos, dataset, tituloX, tituloY, color){
  hist(dataset, freq = FALSE)
  curve(dnorm( x = x, mean = estadisticos$estimate[1], sd = estadisticos$estimate[2]),
    from = min(dataset), to = max(dataset),
    col = color, ylim = c(0,0.25), xlab = tituloX, ylab = tituloY, las=1, main = "", add = T)
}

#Gráficos esperados a ver que tipo de distribución es.
dibujarDensidades(ajuste.Mean, Sonido$Mean_dB, "Mean_dB", "Densidad", "blue")

#Pruebas de Normalidad.
require(nortest)

test <- function(datos){
  ##Test de Shapiro-Wilk
  sw <- shapiro.test(datos$Mean_dB)

  ##Lilliefors (Kolmogorov-Smirnov)
  ll<- lillie.test(datos$Mean_dB)

  ##Test de Anderson-Darling
  Ad<- ad.test(datos$Mean_dB)

  ##Resultados
  sw
  ll
  Ad

```

```

    }
#Resultados de los test
test(Sonido)

test(Piso_1)

test(Piso_2)

test(Martes)

test(Miercoles)

test(Jueves)

test(Viernes)

test(Lugar_1)

test(Lugar_2)

test(Lugar_3)

test(Lugar_4)

test(Lugar_5)

test(Semana_1)

test(Semana_2)

#Otros test que sirven para verificar la normalidad.
#require(goftest)
#cv <- cvm.test(MIN_dB, "pnorm", mean = ajuste.MIN$estimate[1], sd=ajuste.MIN$estimate[2])
#Ks <- ks.test(Mean_dB, "pnorm", mean = ajuste.Mean$estimate[1], sd=ajuste.Mean$estimate[2])
#cv <- cvm.test(MIN_dB, "pnorm", mean = ajuste.MIN$estimate[1], sd=ajuste.MIN$estimate[2])

#Función para generar Gráficos de cuantiles y distribuciones.
GraficCuantiles <- function(datos, ajuste, texto){
  a <- paste("Comparación de Cuantiles", as.character(texto), sep = " ")
  op <- par(no.readonly = TRUE)
  par(mfrow = c(1, 2))
  with(data = NULL, {
    plot(ecdf(datos), main = NULL)
    curve(pnorm(x, ajuste$estimate[1], ajuste$estimate[2]), col = 2, add = T, lwd=2, main = NULL)
    qqPlot(datos, "norm", mean = ajuste$estimate[1], sd = ajuste$estimate[2], pch = 19,
           ylab = "Decibeles", xlab = "Cuantil normal")
  })
  title(main = a, line = -1, cex = 1, outer = T, font.main = 2)
  par(op)
}

#Gráfico principal que muestra que todos nuestros datos distribuyen normal por separado.
plot(ecdf(Sonido$Mean_dB), main = NULL)
curve(pnorm(x, ajuste.Mean$estimate[1], ajuste.Mean$estimate[2]), col = 2, add = T, lwd=2, main = NULL)

```

```

qqPlot(Sonido$Mean_dB, "norm", mean = ajuste.Mean$estimate[1], sd = ajuste.Mean$estimate[2], pch = 19,
       ylab = "Decibeleles", xlab = "Cuantil normal")

#Generando los gráficos utiliznado la función antes descrita utilizando los filtros y
concluir normalidad.

GraficCuantiles(Sonido$Mean_dB, ajuste.Mean, "Total")

GraficCuantiles(Piso_1$Mean_dB, ajuste.piso1, "Piso 1")

GraficCuantiles(Piso_2$Mean_dB, ajuste.piso2, "Piso 2")

GraficCuantiles(Martes$Mean_dB, ajuste.mart, "Martes")

GraficCuantiles(Miercoles$Mean_dB, ajuste.mier, "Miércoles")

GraficCuantiles(Jueves$Mean_dB, ajuste.juev, "Jueves")

GraficCuantiles(Viernes$Mean_dB, ajuste.vier, "Viernes")

GraficCuantiles(Lugar_1$Mean_dB, ajuste.lugar1, "Lugar 1")

GraficCuantiles(Lugar_2$Mean_dB, ajuste.lugar2, "Lugar 2")

GraficCuantiles(Lugar_3$Mean_dB, ajuste.lugar3, "Lugar 3")

GraficCuantiles(Lugar_4$Mean_dB, ajuste.lugar4, "Lugar 4")

GraficCuantiles(Lugar_5$Mean_dB, ajuste.lugar5, "Lugar 5")

GraficCuantiles(Semana_1$Mean_dB, ajuste.semana1, "Semana 1")

GraficCuantiles(Semana_2$Mean_dB, ajuste.semana2, "Semana 2")

# Recodificación
require(RcmdrMisc)
require(sandwich)

## Recodificamos la variable PISO para que quede como factor.
Sonido <- within(Sonido, {
  PISO <- Recode(PISO, '1 = "PISO_1"; 2 = "PISO_2"; ; ;', as.factor.result = TRUE)
})

## Recodificamos la variable SEMANA para que quede como factor.
Sonido <- within(Sonido, {
  SEMANA <- Recode(SEMANA, '1 = "1"; 2 = "2"; ; ;', as.factor.result = TRUE)
})

## Organizamos los niveles de los días para que se muestren en orden.
Sonido$DIA <- with(Sonido, factor(DIA, levels = c('MARTES', 'MIERCOLES', 'JUEVES', 'VIERNES'),
  ordered = TRUE))

#Función de gráficos de diferencia de medias.
GenerarGraficosMedias <- function(Parametro,Categorica, titulox, tituloy){
  with(Sonido, plotMeans(Parametro, Categorica, error.bars = "conf.int", level = 0.95,

```

```

    connect = TRUE, main = "Comparación de medias", xlab = titulox, ylab = tituly))
}

#Gráfico de diferencia de medias respecto al PISO
GenerarGraficosMedias(Mean_dB, Sonido$PISO, "PISO", "Decibeles")

#Gráfico de diferencia de medias respecto al LUGAR
GenerarGraficosMedias(Mean_dB, Sonido$LUGAR, "LUGAR", "Decibeles")

#Gráfico de diferencia de medias respecto a la SEMANA
GenerarGraficosMedias(Mean_dB, Sonido$SEMANA, "SEMANA", "Decibeles")

#Gráfico de diferencia de medias respecto al DIA
GenerarGraficosMedias(Mean_dB, Sonido$DIA, "DIA", "Decibeles")

require(ggplot2)

#Gráfico de densidades por PISO
g1 <- ggplot(Sonido, aes(x = Sonido$Mean_dB, group = Sonido$PISO, col = PISO))
g1 + geom_line(stat = "density") + ggtitle("Comparacion por Piso") +
theme(plot.title = element_text(hjust = 0.5)) +
  ylab("Proporciones") + xlab("dB")

#Gráfico de densidades por LUGAR
g1 <- ggplot(Sonido, aes(x = Sonido$LUGAR, group = Sonido$LUGAR, col = LUGAR))
g1 + geom_line(stat = "density") + ggtitle("Comparacion por Lugar") +
theme(plot.title = element_text(hjust = 0.5)) +
  ylab("Proporciones") + xlab("dB")

#Gráfico de densidades por SEMANA
g1 <- ggplot(Sonido, aes(x = Sonido$SEMANA, group = Sonido$SEMANA, col = SEMANA))
g1 + geom_line(stat = "density") + ggtitle("Comparacion por SEMANA") +
theme(plot.title = element_text(hjust = 0.5)) +
  ylab("Proporciones") + xlab("dB")

#Gráfico de densidades por DIA
g1 <- ggplot(Sonido, aes(x = Sonido$Mean_dB, group = Sonido$DIA, col = DIA))
g1 + geom_line(stat = "density") + ggtitle("Comparacion por Días") +
theme(plot.title = element_text(hjust = 0.5)) +
  ylab("Proporciones") + xlab("dB")

```