

# Aplicación Estadística en Shiny para Variables Cualitativas y Cuantitativas

SALOME MARGARET QUISPE HILASACA

## 1. Fundamento Teórico

Esta aplicación tiene como objetivo permitir el análisis estadístico automático de variables cualitativas y cuantitativas a partir de un archivo cargado por el usuario. Se incluyen:

- **Pruebas de Normalidad:** Verifican si una variable sigue una distribución normal (Shapiro-Wilk, Kolmogorov-Smirnov).
- **Pruebas Paramétricas:** Prueba t de Student, Welch y ANOVA, para comparar medias.
- **Pruebas No Paramétricas:** Wilcoxon, Mann-Whitney, Kruskal-Wallis, cuando no se cumple normalidad.
- **Pruebas de Proporciones:** Chi-cuadrado y Fisher para comparar frecuencias.
- **Correlaciones:** Pearson y Spearman, para estudiar la relación entre variables numéricas.

## 2. Código en R (Shiny)

```
library(shiny)
library(ggplot2)
library(DT)
library(readxl)
library(bslib)

ui <- fluidPage(
  theme = bs_theme(bootswatch = "flatly", primary = "#2C3E50"),
  titlePanel("App Completa de Pruebas Estadísticas"),
  sidebarLayout(
    sidebarPanel(
      fileInput("archivo", "Sube CSV o Excel", accept = c(".csv", ".xlsx")),
      uiOutput("var_response"),
      uiOutput("var_group"),
      uiOutput("prueba")
    ),
    mainPanel(
```

```

        DTOutput("tabla"),
        verbatimTextOutput("resultado"),
        plotOutput("grafico")
    )
)
)

server <- function(input, output, session) {
  datos <- reactive({
    req(input$archivo)
    ext <- tools::file_ext(input$archivo$name)
    if (ext == "csv") read.csv(input$archivo$datapath)
    else read_excel(input$archivo$datapath)
  })

  output$tabla <- renderDT({ datatable(datos()) })

  output$var_response <- renderUI({
    selectInput("respuesta", "Variable de análisis", choices =
      names(datos()))
  })

  output$var_group <- renderUI({
    selectInput("grupo", "Variable de grupo", choices = names(
      datos()))
  })

  output$prueba <- renderUI({
    selectInput("test", "Prueba estadística", choices = c(
      "Resumen estadístico",
      "Prueba de Normalidad (Shapiro-Wilk)",
      "Prueba de Normalidad (Kolmogorov-Smirnov)",
      "Chi-cuadrado",
      "Fisher Exacta",
      "Prueba t de Student",
      "Prueba de Welch",
      "ANOVA",
      "Kruskal-Wallis",
      "Mann-Whitney",
      "Wilcoxon",
      "Correlación de Pearson",
      "Correlación de Spearman"
    ))
  })

  output$resultado <- renderPrint({
    df <- datos()
    y <- df[[input$respuesta]]
    g <- as.factor(df[[input$grupo]])

    switch(input$test,

```

```

"Resumen estadístico" = summary(y),
"Prueba de Normalidad (Shapiro-Wilk)" = shapiro.test(y),
"Prueba de Normalidad (Kolmogorov Smirnov)" = ks.test(
  scale(y), "pnorm"),
"Chi-cuadrado" = chisq.test(table(g, y)),
"Fisher Exacta" = fisher.test(table(g, y)),
"Prueba t de Student" = t.test(y ~ g, var.equal = TRUE),
"Prueba de Welch" = t.test(y ~ g, var.equal = FALSE),
"ANOVA" = summary(aov(y ~ g)),
"Kruskal-Wallis" = kruskal.test(y ~ g),
" Mann Whitney " = wilcox.test(y ~ g),
"Wilcoxon" = wilcox.test(y, paired = TRUE),
"Correlación de Pearson" = cor.test(df[[input$respuesta]],
  df[[input$grupo]], method = "pearson"),
"Correlación de Spearman" = cor.test(df[[input$respuesta
  ]], df[[input$grupo]], method = "spearman")
)
})

output$grafico <- renderPlot({
  df <- datos()
  y <- df[[input$respuesta]]
  g <- as.factor(df[[input$grupo]])
  if (is.numeric(y)) {
    boxplot(y ~ g, col = rainbow(length(unique(g))), main = "
      Boxplot")
  } else {
    barplot(table(y), col = "steelblue", main = "Frecuencia")
  }
})
}

shinyApp(ui, server)

```

### 3. Imágenes de la Aplicación

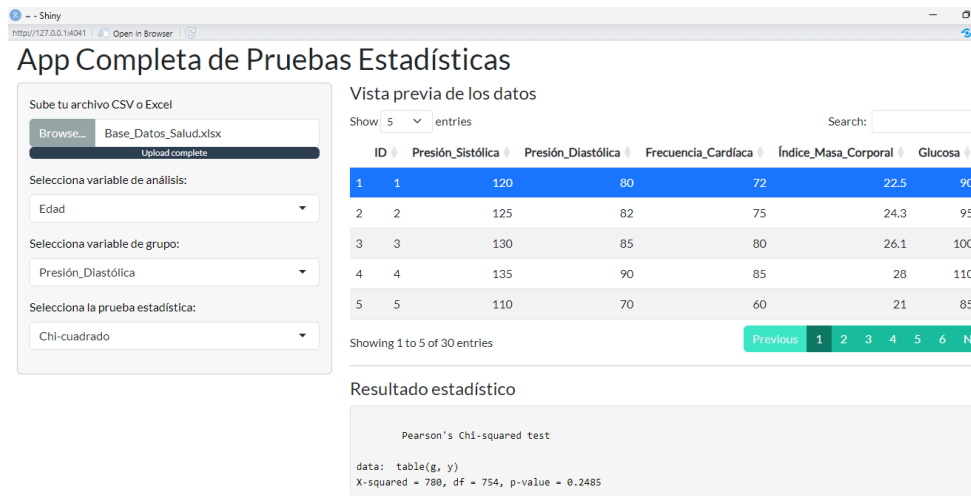


Figura 1: Interfaz principal de la app Shiny

## Gráfico

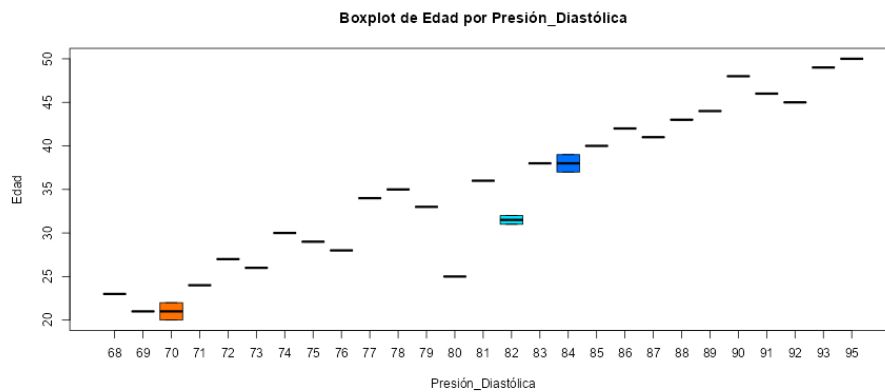


Figura 2: Ejemplo de salida gráfica (boxplot)