

Reporte Técnico: Aplicación Shiny de Análisis Estadístico

Facultad de Estadística e Informática

Docente: Fred Torres Cruz
Repositorio GitHub: APP shiny
Nombre: Adelmi Cordova Apaza
Código: 230850

1. Descripción General

La aplicación Shiny desarrollada permite realizar un análisis estadístico exploratorio e inferencial a partir de archivos en formato `.csv` o `.xlsx`. Está pensada para usuarios con o sin experiencia previa en programación, facilitando la interpretación de datos de manera visual e intuitiva.

2. Características Principales

- Carga dinámica de archivos Excel o CSV.
- Selección de variables para análisis.
- Estadísticas descriptivas: media, mediana, desviación estándar, etc.
- Gráficos: histograma y diagrama de cajas.
- Análisis inferencial (prueba t, ANOVA).
- Interpretación automática del resultado del análisis.

3. Beneficios de Uso

- Interfaz sencilla y adaptable a diferentes niveles de conocimiento.
- Ideal para docentes, investigadores y estudiantes.

- Rápida generación de resultados e interpretación.
- Sin necesidad de escribir código para realizar análisis comunes.

4. Guía Rápida de Uso

1. Abre la aplicación localmente o en línea.
2. Carga un archivo `.csv` o `.xlsx`.
3. Selecciona 2 o 3 variables numéricas.
4. Haz clic en “**Analizar**”.
5. Observa las estadísticas, gráficos y sugerencias de análisis.

5. Público Objetivo

Este aplicativo está dirigido a:

- Estudiantes de estadística, economía, psicología, educación.
- Profesionales que necesiten interpretar datos sin usar software estadístico complejo.
- Docentes que buscan enseñar análisis estadístico de forma interactiva.

6. Código Fuente

A continuación se presenta el código fuente principal de la aplicación Shiny (resumen):

Listing 1: Fragmento de app.R

```
library(shiny)
library(readxl)
library(dplyr)
library(ggplot2)
library(DescTools)
library(shinythemes)
library(DT)
library(psych)

ui <- fluidPage(
  theme = shinytheme("flatly"),
  titlePanel(div(h2("Análisis Estadístico Profesional",
    style = "color:#2c3e50;font-weight:bold"))),

  tabsetPanel(
    tabPanel("Cargar Datos",
```

```
fluidRow(
  column(4,
    wellPanel(
      fileInput("archivo", "      Selecciona un
      archivo (.csv o .xlsx):", accept = c(".
      csv", ".xlsx")),
      uiOutput("varSeleccion"),
      checkboxInput("estandarizar", "      Estandarizar datos", value = FALSE),
      actionButton("analizar", "      Analizar",
        class = "btn btn-primary btn-block")
    )
  ),
  column(8,
    h4("      Vista previa de los datos"),
    DTOutput("vistaPrevia")
  )
),

tabPanel("      Estad sticas",
  fluidRow(
    column(6,
      h4("      Medidas Descriptivas"),
      DTOutput("estadisticas")
    ),
    column(6,
      h4("      Histograma"),
      plotOutput("grafico")
    )
  ),
  hr(),
  h4("      Diagrama de Cajas"),
  plotOutput("boxplot")
),

tabPanel("      An lisis",
  fluidRow(
    column(6,
      h4("      Prueba sugerida"),
      verbatimTextOutput("sugerencia")
    ),
    column(6,
      h4("      Resultado"),
      verbatimTextOutput("resultado")
    )
  ),
  hr(),
```

```
        h4("          Interpretación del resultado"),
        textOutput("interpretacion")
    )
)
)

server <- function(input, output) {
  datos <- reactive({
    req(input$archivo)
    ext <- tools::file_ext(input$archivo$name)
    if (ext == "csv") {
      read.csv(input$archivo$datapath)
    } else if (ext == "xlsx") {
      read_excel(input$archivo$datapath)
    } else {
      validate("Formato no soportado. Usa .csv o .xlsx")
    }
  })

  datos_filtrados <- reactive({
    req(input$vars)
    df <- datos()[, input$vars, drop = FALSE]
    if (input$estandarizar) {
      df <- as.data.frame(scale(df))
    }
    df
  })

  output$vistaPrevia <- renderDT({
    datatable(head(datos(), 10), options = list(scrollX = TRUE))
  })

  output$varSeleccion <- renderUI({
    req(datos())
    selectInput("vars", "          Selecciona variables num ricas (2 o 3):",
               choices = names(datos()), multiple = TRUE)
  })

  output$sugerencia <- renderText({
    req(input$vars)
    n <- nrow(datos())
    v <- input$vars

    if (length(v) == 2) {
      if (n <= 35) {
        return("Sugerencia: aplicar prueba t para comparar dos grupos.")
      }
    }
  })
}
```

```
    } else {
      return("Sugerencia: aplicar ANOVA o regresión lineal.")
    }
  } else if (length(v) == 3) {
    return("Sugerencia: aplicar ANOVA factorial para evaluar efectos múltiples.")
  } else {
    return("Selecciona 2 o 3 variables para realizar un análisis.")
  }
})

output$estadisticas <- renderDT({
  req(input$vars)
  dat <- datos_filtrados()
  if (!all(sapply(dat, is.numeric))) return(NULL)

  stats <- psych::describe(dat[, c("mean", "median", "sd", "min",
    "max", "range", "skew", "kurtosis")])
  datatable(round(stats, 3), options = list(pageLength = 5))
})

output$grafico <- renderPlot({
  req(input$vars)
  dat <- datos_filtrados()
  ggplot(dat, aes_string(x = input$vars[1])) +
    geom_histogram(bins = 15, fill = "#3498db", color = "white") +
    theme_minimal() +
    labs(title = paste("Histograma de", input$vars[1]),
      x = input$vars[1], y = "Frecuencia")
})

output$boxplot <- renderPlot({
  req(input$vars)
  dat <- datos_filtrados()
  ggplot(stack(dat), aes(x = ind, y = values, fill = ind)) +
    geom_boxplot() +
    theme_minimal() +
    labs(title = "Diagrama de cajas por variable", x = "Variable",
      y = "Valor") +
    theme(legend.position = "none")
})

resultado_analisis <- reactive({
  req(input$vars)
  dat <- datos()
  v <- input$vars
```

```
if (!all(sapply(dat[, v], is.numeric))) return(NULL)

if (length(v) == 2 && nrow(dat) <= 35) {
  return(t.test(dat[[v[1]]], dat[[v[2]]]))
} else if (length(v) == 3) {
  formula <- as.formula(paste(v[1], "~", v[2], "+", v[3]))
  return(summary(aov(formula, data = dat)))
} else {
  return(NULL)
}
})

output$resultado <- renderPrint({
  res <- resultado_analisis()
  if (is.null(res)) return("No se pudo realizar el análisis con las variables seleccionadas.")
  print(res)
})

output$interpretacion <- renderText({
  res <- resultado_analisis()
  if (is.null(res)) return("No se aplicó ninguna prueba estadística.")

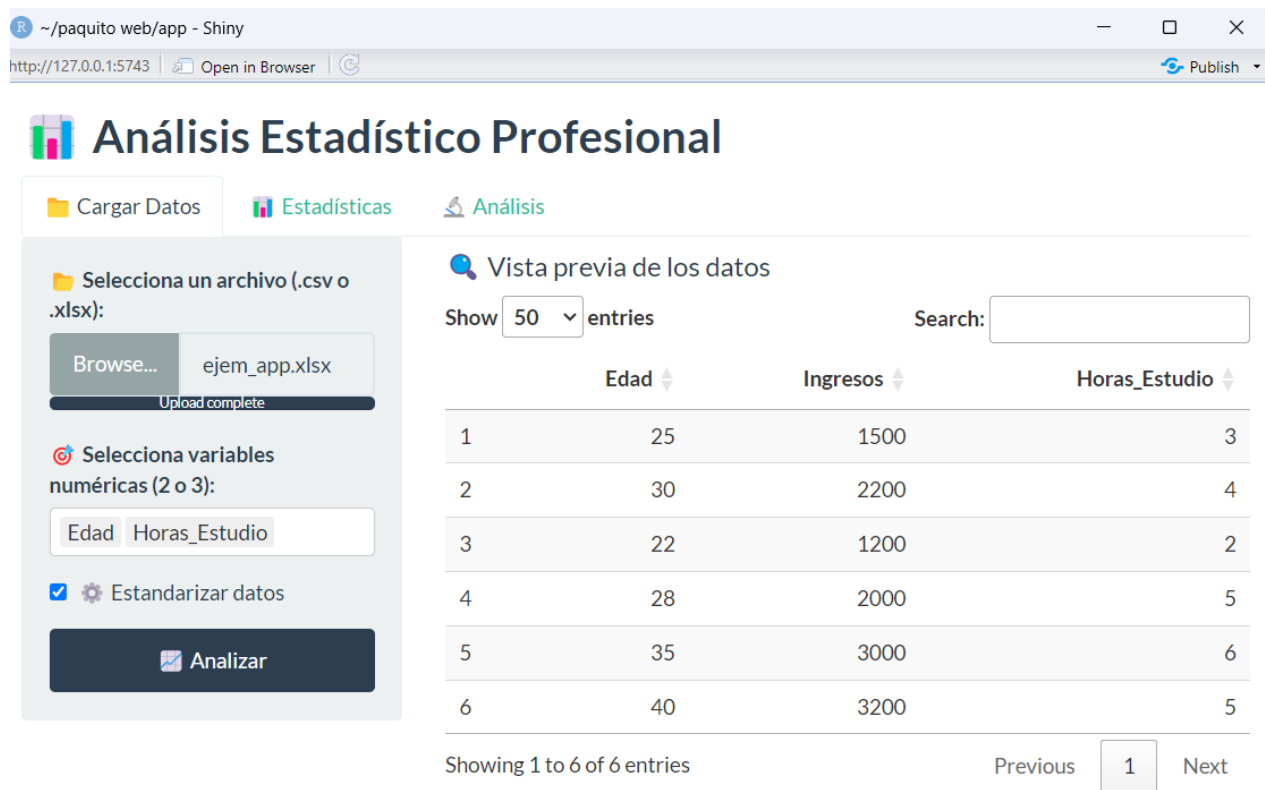
  if (inherits(res, "htest")) {
    p <- res$p.value
    if (p < 0.05) {
      return(paste("El valor p=", round(p, 4), "es menor que 0.05. Se rechaza la hipótesis nula: hay diferencias significativas."))
    } else {
      return(paste("El valor p=", round(p, 4), "es mayor o igual a 0.05. No se rechaza la hipótesis nula: no hay diferencias significativas."))
    }
  }

  if (inherits(res, "anova")) {
    p <- res[[1]][["Pr(>F)"]][1]
    if (p < 0.05) {
      return(paste("El valor p=", round(p, 4), "indica diferencias significativas entre los grupos. Se rechaza la hipótesis nula."))
    } else {
      return(paste("El valor p=", round(p, 4), "indica que no hay diferencias significativas entre los grupos."))
    }
  }
})
```

```
    return("Resultado no interpretable.")
  })
}

shinyApp(ui, server)
```

7. Evidencia Visual



The screenshot shows the initial view of the 'Análisis Estadístico Profesional' Shiny application. The interface is divided into a sidebar and a main panel.

Sidebar:

- Cargar Datos:** Includes a file selection button 'Selecciona un archivo (.csv o .xlsx):' with a 'Browse...' button and a file named 'ejem_app.xlsx'. Below this is a 'Selección de variables numéricas (2 o 3):' section with input fields for 'Edad' and 'Horas_Estudio'. There is also a checkbox for 'Estandarizar datos' which is checked.
- Estadísticas:** A button to view statistical summaries.
- Análisis:** A button to perform the analysis.
- Analizar:** A large button at the bottom of the sidebar to execute the analysis.

Main Panel:

- Vista previa de los datos:** A section showing a preview of the data table. It includes a 'Show' dropdown set to '50' and a 'Search' input field.
- Data Table:** A table with 3 columns: 'Edad', 'Ingresos', and 'Horas_Estudio'. It displays 6 entries.
- Footer:** Shows 'Showing 1 to 6 of 6 entries' and pagination controls with 'Previous', '1', and 'Next' buttons.

	Edad	Ingresos	Horas_Estudio
1	25	1500	3
2	30	2200	4
3	22	1200	2
4	28	2000	5
5	35	3000	6
6	40	3200	5

Figura 1: Vista inicial de la aplicación con opciones de carga.

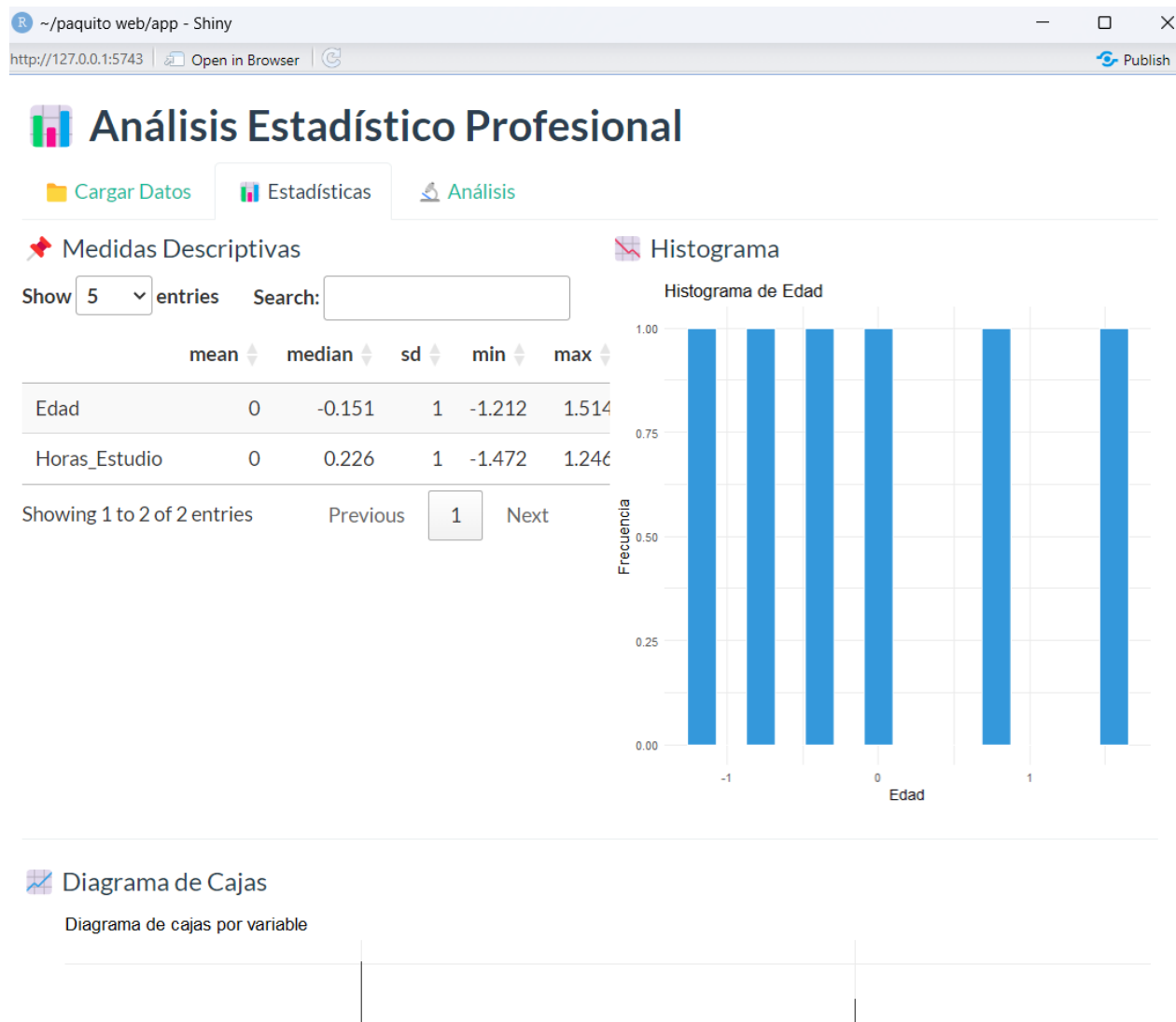


Figura 2: Resultados del análisis estadístico con gráficos.

~/paquito web/app - Shiny
http://127.0.0.1:5743 | Open in Browser | Publish

Análisis Estadístico Profesional

Cargar Datos Estadísticas **Análisis**

Prueba sugerida

Sugerencia: aplicar prueba t para comparar dos grupos

Resultado

```
Welch Two Sample t-test

data:  dat[[v[1]]] and dat[[v[2]]]
t = 9.3537, df = 5.4957, p-value = 0.0001401
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 18.92215 32.74452
sample estimates:
mean of x mean of y
30.000000  4.166667
```

Interpretación del resultado

El valor $p = 1e-04$ es menor que 0.05. Se rechaza la hipótesis nula: hay diferencias significativas.

Figura 3: Vista inicial de la aplicación con analisis

8. Repositorio en GitHub

Puedes acceder al código completo, datos de prueba y documentación en el siguiente enlace:

<https://github.com/Adelmi195/APP-shiny/blob/main/app.R>