CHI-RESEARCHER



*Código Fuente*

**T-RESEARCHER**

**Autores:**

1

2

3

**ÍNDICE**

**04**

**Código Fuente**

**05**

**Archivo: app.r**

**1. CÓDIGO FUENTE:**

La aplicación ChiResearcher ha sido desarrollada utilizando Python y el framework Streamlit con el objetivo de facilitar la realización de pruebas estadísticas de Chi-Cuadrado. Esta herramienta está diseñada para que los usuarios puedan llevar a cabo tres tipos principales de pruebas: independencia, bondad de ajuste y homogeneidad, todo de manera interactiva. Una característica clave de la aplicación es su capacidad para generar reportes detallados que incluyen resultados, gráficos visualmente atractivos y análisis interpretativo, con la opción de descargar estos reportes en formato de texto o PDF.

El desarrollo de ChiResearcher se realizó utilizando tecnologías modernas y ampliamente adoptadas en el análisis de datos. Python fue elegido por ser uno de los lenguajes más versátiles y populares en la ciencia de datos. Streamlit, por su parte, se utilizó para crear una interfaz web interactiva que facilita la experiencia del usuario al realizar las pruebas estadísticas. Para la carga y manipulación de datos, se implementaron las bibliotecas pandas y numpy, las cuales son fundamentales para el manejo de grandes volúmenes de información en formato tabular, permitiendo realizar operaciones matemáticas y estadística básica de manera eficiente.

Además, la biblioteca SciPy fue empleada para realizar los cálculos de las pruebas de Chi-Cuadrado, ya que ofrece todas las funciones estadísticas necesarias para ejecutar comparaciones de medias, calcular valores p, y realizar análisis de frecuencias esperadas y observadas. Para la visualización de datos, se utilizó Plotly Express, una herramienta que permite generar gráficos interactivos y personalizables, mejorando la interpretación de los resultados obtenidos. Por último, la generación de informes en formato PDF se realizó con la biblioteca FPDF, la cual fue crucial para personalizar y estructurar los reportes, incluyendo gráficos y resultados de las pruebas realizadas por el usuario.

La estructura de la aplicación permite que el flujo de trabajo comience con la carga de datos por parte del usuario, ya sea en formato CSV o Excel. Luego, el usuario selecciona el tipo de prueba de Chi-Cuadrado que desea realizar: independencia, bondad de ajuste o homogeneidad. Posteriormente, se realiza la prueba estadística correspondiente, mostrando los resultados tanto numéricos como visuales mediante gráficos. Al final del proceso, la aplicación ofrece la posibilidad de generar un informe en formato PDF con los resultados obtenidos, lo cual incluye gráficos de apoyo para facilitar la comprensión de los datos analizados. Esto permite al usuario obtener un reporte claro y comprensible de los análisis realizados.

ARCHIVO: APP.PY

import streamlit as st

import pandas as pd

import numpy as np

from scipy import stats

from scipy.stats import chi2\_contingency

import plotly.express as px

# Configuración de la página

st.set\_page\_config(page\_title="Pruebas de Chi-Cuadrado", layout="wide")

# Título y descripción de la aplicación

st.title("Aplicación de Pruebas de Chi-Cuadrado")

st.write("""

Esta aplicación permite realizar diferentes pruebas de Chi-Cuadrado, incluyendo independencia, bondad de ajuste, y homogeneidad.

Puedes cargar tus datos, seleccionar las variables y realizar la prueba de forma interactiva.

""")

# Sidebar para configuración del modo

st.sidebar.title("Configuración de la Interfaz")

theme = st.sidebar.radio("Selecciona el tema", ["Claro", "Oscuro"])

# Aplicación del tema

if theme == "Oscuro":

    st.markdown(

        """

        <style>

        body {

            background-color: #0e1117;

            color: white;

        }

        .css-1aumxhk, .css-16idsys, .css-1ukrds8 {

            background-color: #0e1117;

        }

        .css-10trblm {

            color: white;

        }

        .css-1cpxqw2 a {

            color: #ff4b4b;

        }

        .stButton>button {

            background-color: #ff4b4b;

            color: white;

        }

        .stDownloadButton>button {

            background-color: #ff4b4b;

            color: white;

        }

        </style>

        """,

        unsafe\_allow\_html=True

    )

else:

    st.markdown(

        """

        <style>

        body {

            background-color: white;

            color: black;

        }

        .css-1aumxhk, .css-16idsys, .css-1ukrds8 {

            background-color: white;

        }

        .css-10trblm {

            color: black;

        }

        .css-1cpxqw2 a {

            color: #584bff;

        }

        .stButton>button {

            background-color: #584bff;

            color: white;

        }

        .stDownloadButton>button {

            background-color: #584bff;

            color: white;

        }

        </style>

        """,

        unsafe\_allow\_html=True

    )

# Carga de datos

uploaded\_file = st.sidebar.file\_uploader("Cargar archivo CSV o Excel", type=["csv", "xlsx"])

if uploaded\_file:

    if uploaded\_file.name.endswith('.csv'):

        df = pd.read\_csv(uploaded\_file)

    else:

        df = pd.read\_excel(uploaded\_file)

    st.write("Datos cargados:")

    st.dataframe(df.head())

    # Selección del tipo de prueba de Chi-Cuadrado

    st.write("### Selección del Tipo de Prueba de Chi-Cuadrado")

    test\_type = st.selectbox("Selecciona el tipo de prueba", ["Independencia", "Bondad de Ajuste", "Homogeneidad"])

    columns = df.select\_dtypes(include=['object', 'category']).columns.tolist()

    if test\_type == "Independencia":

        st.write("### Prueba de Independencia")

        var1 = st.selectbox("Selecciona la primera variable categórica", columns)

        var2 = st.selectbox("Selecciona la segunda variable categórica", [col for col in columns if col != var1])

        # Tabla de contingencia

        st.write("### Tabla de Contingencia")

        contingency\_table = pd.crosstab(df[var1], df[var2])

        st.write(contingency\_table)

        # Cálculo de la prueba Chi-Cuadrado

        st.write("### Resultados de la Prueba Chi-Cuadrado")

        chi2, p\_value, dof, expected = chi2\_contingency(contingency\_table)

        st.write(f"Chi-cuadrado: {chi2:.4f}")

        st.write(f"p-valor: {p\_value:.4f}")

        st.write(f"Grados de libertad: {dof}")

        st.write("Frecuencias esperadas:")

        st.write(expected)

        # Interpretación de resultados

        st.write("### Interpretación de Resultados")

        alpha = st.number\_input("Nivel de significancia (alpha)", value=0.05)

        interpretacion = ("Rechazamos la hipótesis nula. Existe una relación significativa entre las variables."

                          if p\_value < alpha else

                          "No podemos rechazar la hipótesis nula. No existe evidencia suficiente para afirmar una relación significativa entre las variables.")

        st.write(interpretacion)

        # Visualización gráfica

        st.write("### Visualización Gráfica")

        # Gráfico circular para la primera variable

        fig1 = px.pie(df, names=var1, title=f"Distribución de {var1}")

        st.plotly\_chart(fig1, use\_container\_width=True)

        # Gráfico circular para la segunda variable

        fig2 = px.pie(df, names=var2, title=f"Distribución de {var2}")

        st.plotly\_chart(fig2, use\_container\_width=True)

    elif test\_type == "Bondad de Ajuste":

        st.write("### Prueba de Bondad de Ajuste")

        variable = st.selectbox("Selecciona la variable categórica", columns)

        # Frecuencias observadas

        observed = df[variable].value\_counts().sort\_index()

        st.write("Frecuencias observadas:")

        st.write(observed)

        # Frecuencias esperadas

        expected\_input = st.text\_area("Ingresa las frecuencias esperadas (separadas por comas)")

        expected = np.array([float(x) for x in expected\_input.split(",")])

        # Prueba de bondad de ajuste

        chi2, p\_value = stats.chisquare(f\_obs=observed, f\_exp=expected)

        st.write(f"Chi-cuadrado: {chi2:.4f}")

        st.write(f"p-valor: {p\_value:.4f}")

        # Interpretación de resultados

        st.write("### Interpretación de Resultados")

        alpha = st.number\_input("Nivel de significancia (alpha)", value=0.05)

        interpretacion = ("Rechazamos la hipótesis nula. La distribución observada difiere significativamente de la esperada."

                          if p\_value < alpha else

                          "No podemos rechazar la hipótesis nula. No existe evidencia suficiente para afirmar que la distribución observada difiere de la esperada.")

        st.write(interpretacion)

    elif test\_type == "Homogeneidad":

        st.write("### Prueba de Homogeneidad")

        variable = st.selectbox("Selecciona la variable categórica", columns)

        group = st.selectbox("Selecciona la variable de grupo", columns)

        contingency\_table = pd.crosstab(df[variable], df[group])

        st.write("Tabla de Contingencia:")

        st.write(contingency\_table)

        chi2, p\_value, dof, expected = chi2\_contingency(contingency\_table)

        st.write(f"Chi-cuadrado: {chi2:.4f}")

        st.write(f"p-valor: {p\_value:.4f}")

        st.write(f"Grados de libertad: {dof}")

        st.write("Frecuencias esperadas:")

        st.write(expected)

        # Interpretación de resultados

        st.write("### Interpretación de Resultados")

        alpha = st.number\_input("Nivel de significancia (alpha)", value=0.05)

        interpretacion = ("Rechazamos la hipótesis nula. Las distribuciones en los grupos no son homogéneas."

                          if p\_value < alpha else

                          "No podemos rechazar la hipótesis nula. No existe evidencia suficiente para afirmar que las distribuciones en los grupos son diferentes.")

        st.write(interpretacion)

    # Generar reporte

    reporte = f"""

    Resultados de la Prueba Chi-Cuadrado ({test\_type})\n

    Variables analizadas: {var1 if test\_type == 'Independencia' else variable} y {var2 if test\_type == 'Independencia' else 'N/A'}\n

    Chi-cuadrado: {chi2:.4f}\n

    p-valor: {p\_value:.4f}\n

    Grados de libertad: {dof if test\_type != 'Bondad de Ajuste' else 'N/A'}\n

    Nivel de significancia (alpha): {alpha}\n

    Interpretación: {interpretacion}

    """

    # Descargar reporte en texto

    st.download\_button(

        label="Descargar Reporte (Texto)",

        data=reporte,

        file\_name='reporte\_chi\_cuadrado.txt',

        mime='text/plain'

    )

else:

    st.write("Por favor, carga un archivo de datos para comenzar.")

