

UNIVERSIDAD NACIONAL DEL ORIENTE "UNO"

CARRERA DE INGENIERIA DE SISTEMAS



CALCULADORA ESTADISTICA INTERACTIVA

Autor: Cesar Gustavo Villegas Crespo

Tutor: Delcy Nogales Rosado

Materia: Probabilidad y Estadística

Semestre: Tercer Semestre

Fecha: 5/06/2024

Santa Cruz, Bolivia

1. RESUMEN

El proyecto consiste en desarrollar una calculadora estadística que facilita el análisis de datos a través de un entorno interactivo. Los usuarios pueden crear estudios, ingresar datos y generar tablas de frecuencia con los datos numéricos y su cantidad de los datos. Si el usuario desea se puede agrupar y clasificarlos por clases los datos.

El programa ofrece opciones de visualización mediante gráficos de barras, polígonos de frecuencia o gráficos de torta, permitiendo una interpretación visual y comprensible de los datos estadísticos. Además, incluye un cuadro informativo con las medidas de tendencia central, como la media, mediana y moda, proporcionando un análisis estadístico completo.

Un aspecto destacado del proyecto es la capacidad de guardar los resultados en archivos Excel, facilitando la exportación y el uso posterior de los datos analizados. La metodología empleada incluye el uso de Python y Tkinter para la interfaz gráfica, junto con bibliotecas adicionales para el manejo y visualización de datos.

El impacto potencial de este proyecto es significativo, ya que proporciona una herramienta accesible y poderosa para estudiantes, investigadores y profesionales que necesitan realizar análisis estadísticos de manera eficiente y precisa.

2. PLANTEAMIENTO DEL PROBLEMA

El proyecto aborda la necesidad de simplificar y agilizar el análisis estadístico de datos mediante una calculadora interactiva. En el contexto actual, la capacidad de manejar eficientemente datos numéricos como no numéricos y representarlos de manera visual es crucial para estudiantes, investigadores y profesionales en diversos campos. La dificultad radica en la variedad y cantidad de datos, especialmente cuando se trata de datos numéricos que requieren una agrupación adecuada para su análisis.

Las limitaciones incluyen la complejidad de realizar cálculos estadísticos precisos y la interpretación visual de datos variados. Además, la necesidad de exportar resultados de manera estructurada y legible, como en archivos Excel, también es un desafío importante que aborda el proyecto.

El impacto potencial de resolver estos problemas es significativo, ya que proporciona una herramienta accesible y poderosa para mejorar la eficiencia y precisión en el análisis estadístico, beneficiando tanto a estudiantes que aprenden conceptos estadísticos como a profesionales que dependen de análisis rigurosos en sus investigaciones y decisiones.

3. OBJETIVOS

3.1 Objetivo General

Desarrollar una calculadora estadística interactiva que permita a los usuarios realizar análisis detallados y visuales de datos numéricos, facilitando la creación de tablas de frecuencia y gráficos representativos. Además, integrar funcionalidades para calcular y mostrar medidas de tendencia central de manera clara y accesible.

3.2 Objetivos Específicos

- Analizar los conceptos básicos de la estadística descriptiva.
- Conocer y desarrollar ejercicios de estadística descriptiva.
- Implementar un sistema de ingreso de datos versátil que permita registrar tanto datos numéricos.
- Automatizar la generación de tablas de frecuencia, adaptándolas dinámicamente según la cantidad y naturaleza de los datos ingresados.
- Integrar opciones para graficar las tablas de frecuencia mediante gráficos de barras, polígonos de frecuencia o gráficos de torta, ofreciendo flexibilidad visual en la representación de datos.
- Desarrollar un cuadro informativo que presente de manera clara las medidas de tendencia central (media, mediana y moda) calculadas a partir de los datos ingresados.
- Implementar funcionalidades de exportación que permitan guardar los resultados de los análisis realizados en archivos Excel, facilitando la gestión y el uso posterior de la información estadística.

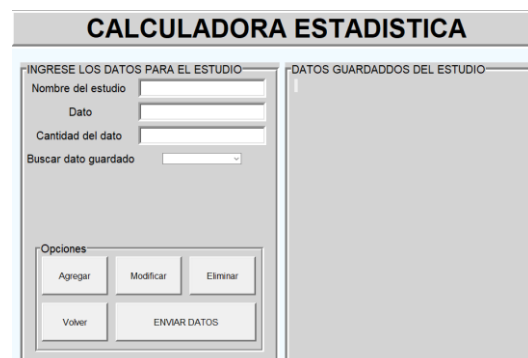
4. PROPUESTA DE SOLUCIÓN

La solución propuesta es una calculadora estadística interactiva desarrollada en Python utilizando la biblioteca Tkinter para la interfaz gráfica. La aplicación permite a los usuarios realizar análisis detallados de datos, independientemente de si son numéricos u otros tipos. Los componentes principales incluyen:

- **Interfaz de Usuario Amigable:**

Sistema de ingreso de datos versátil que acepta datos numéricos.

Opciones claras para crear estudios, ingresar datos y realizar análisis estadísticos.



- **Generación de Tablas de Frecuencia:**

Automatización de la generación de tablas de frecuencia adaptadas dinámicamente según la cantidad de datos ingresados.

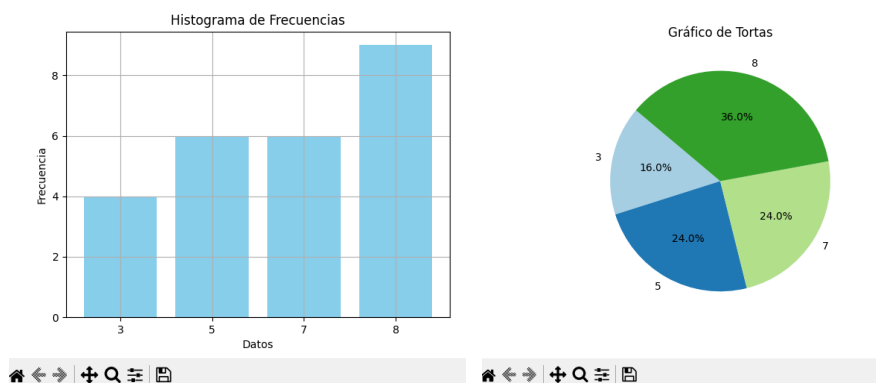
Capacidad para agrupar datos numéricos en clases, si es necesario, optimizando la visualización y el análisis.

CALCULADORA ESTADISTICA							
Opciones							
INFORMACION							
G. BARRAS							
G. TORTAS							
G. POLIGONOS							
GUARDAR							
ANTERIOR							
MENU							
SALIR							

Tabla Frecuencia							
Date	F	FA	FR	FAR			
3	4	4	16.0	16.0			
5	6	10	24.0	40.0			
7	6	16	24.0	64.0			
8	9	25	36.0	100.0			

- **Visualización Gráfica:**

Opciones para representar visualmente las tablas de frecuencia mediante gráficos de barras, polígonos de frecuencia o gráficos de torta, permitiendo una interpretación intuitiva de los datos.



- **Análisis Estadístico:**

Cuadro informativo integrado que muestra medidas de tendencia central (media, mediana y moda) calculadas a partir de los datos ingresados, facilitando la comprensión y la interpretación de los resultados.

Informacion del estudio estadistico	
NOMBRE DEL ESTUDIO:	cesar
CANTIDAD DE DATOS UNICOS:	4
CANTIDAD DE DATOS INGRESADOS:	25
MEDIA ARITMETICA:	6
MEDIANA:	7
MODA:	8

- **Exportación de Resultados:**

Funcionalidad para exportar la tabla de frecuencia en un archivo Excel. Esto facilita la gestión y el uso posterior de los datos analizados.

	A	B	C	D	E	F	G
1	Dato	F	FA	FR	FAR		
2	3	4	4	16.0	16.0		
3	5	6	10	24.0	40.0		
4	7	6	16	24.0	64.0		
5	8	9	25	36.0	100.0		
6							

5. METODOLOGÍA DE DESARROLLO

- **Herramientas empleadas en el desarrollo:**

1. **Lenguaje de Programación:**

Python: utilizado como el principal lenguaje de programación para el desarrollo de este proyecto, siendo más específico la versión 3.12.2



2. **Entorno de Desarrollo Integrado (IDE):**

Visual Studio Code: Se emplea como el entorno de desarrollo integrado para la codificación, depuración y gestión de proyectos.



3. **Bibliotecas y Módulos Principales de Python**

tkinter: Utilizado para la creación de interfaces gráficas de usuario (GUI)

PIL (Python Imaging Library) / Pillow: Empleado para la manipulación de imágenes dentro de la aplicación.

matplotlib.pyplot: Utilizado para la visualización de datos mediante gráficos.

collections.Counter: Empleado para contar elementos y calcular frecuencias.

4. Otros Módulos Utilizados

tkinter.messagebox: Para mostrar mensajes de alerta y diálogos.

tkinter.ttk: Proporciona widgets temáticamente consistentes en tkinter.

tkinter.font: Utilizado para manejar fuentes en la interfaz gráfica.

tkinter.filedialog: Empleado para manejar cuadros de diálogo de selección de archivos.

- **Metodología aplicada :**

Se tomaron en cuenta 3 pasos importantes en el desarrollo de dicho programa

1. .Análisis de requisitos

➤ **Requisitos funcionales:**

Capacidad de ingresar datos numéricos y mediante unos requisitos categorizarlos.

Cálculo de medidas de tendencia central: media, mediana, moda.

Generación de tablas de frecuencia para los datos ingresados

Representación gráfica en formatos de barras, polígonos y tortas.

➤ **Requisitos no funcionales:**

Interfaz de usuario intuitiva y fácil de usar utilizando Tkinter.

Escalabilidad para la posible expansión futura del programa.

2. Diseño del Programa

➤ **Arquitectura:**

Utilización del patrón MVC (Modelo-Vista-Controlador) para separar la lógica de manejo de datos y la interfaz de usuario de manera clara y organizada.

➤ **Interfaz de Usuario:**

Entrada de Datos: Área para ingresar datos numéricos y opciones para categorizar si es necesario.

Selección de Gráficos: Opciones para seleccionar el tipo de gráfico a generar (barras, polígonos, tortas).

Visualización de Resultados: Área para mostrar las medidas de tendencia central y visualización gráfica.

➤ **Modelo de Datos:**

Definición de estructuras de datos para manejar tanto datos individuales como agrupados, asegurando compatibilidad con los cálculos estadísticos y la generación de tablas.

3. Implementación

➤ **Desarrollo backend:**

Implementación de funciones para calcular media, mediana, moda y generación de tablas de frecuencia.

Integración de métodos para manejar diferentes tipos de datos numéricos y prepararlos para su visualización gráfica.

➤ **Integración de Gráficos:**

Utilización de Matplotlib para generar gráficos dinámicos basados en los datos ingresados por el usuario, permitiendo opciones de personalización.

➤ **Exportación a Excel**

Exportar la tabla de frecuencia a Excel mediante la biblioteca openpyxl.

6. IMPLEMENTACIÓN Y PRUEBAS

El proyecto se ha implementado utilizando Python y se estructura en tres archivos principales: uno para manejar los datos y guardarlos en Excel, otro para la interfaz de usuario con Tkinter, otro quien controlará la vista , y un principal archivo donde se ejecutará todo el programa.

➤ **Modelos de Datos:**

- El archivo ModeloDatos.py contiene la clase ModeloDatos, encargada de manejar y guardar los datos en un archivo Excel.
- Se utiliza la biblioteca openpyxl para la creación y manipulación de archivos Excel.

```

1 import openpyxl
2
3 class ModeloDatos:
4     def __init__(self):
5         pass
6
7     def guardar_datos_excel(self, datos, nombre_archivo, con_clases):
8         workbook = openpyxl.Workbook()
9         sheet = workbook.active
10        sheet.title = "Datos"
11
12        if con_clases:
13            # Encabezados para datos con clases
14            encabezados = ["Dato", "M", "F", "FA", "FR", "FAR"]
15        else:
16            # Encabezados para datos sin clases
17            encabezados = ["Dato", "F", "FA", "FR", "FAR"]
18
19        for col_num, encabezado in enumerate(encabezados, 1):
20            sheet.cell(row=1, column=col_num, value=encabezado)
21
22        # Datos
23        for row_num, dato in enumerate(datos, 2):
24            for col_num, valor in enumerate(dato, 1):
25                sheet.cell(row=row_num, column=col_num, value=valor)
26
27        workbook.save(nombre_archivo)
28

```

➤ Aplicación:

Este es el archivo vista.py en el cual es encargado de toda la parte visual del programa y quien maneja su funcionalidad es la clase Aplicación.

La clase Aplicacion es la ventana principal de la calculadora estadística, que inicializa y muestra los diferentes marcos.

```

1 class Aplicacion:
2     def __init__(self):
3         self.ventana = Tk()
4
5         self.ventana.title("CALCULADORA ESTADISTICA")
6         self.ventana.geometry("970x675")
7         self.ventana.resizable(False,False)
8
9         # Configurar la ventana principal para que se expanda
10        self.ventana.rowconfigure(0, weight=1)
11        self.ventana.columnconfigure(0, weight=1)
12
13        self.marco_menu = MarcoMenu(self)
14        self.marco2 = MarcoDatos(self)
15        self.Marco_tabla = MarcoTabla(self)
16
17        self.marco_menu.frame.grid(row=0, column=0, sticky="nsew")
18        self.marco2.frame.grid(row=0, column=0, sticky="nsew")
19        self.Marco_tabla.frame.grid(row=0, column=0, sticky="nsew")
20
21
22
23        self.mostrar_marco(self.marco_menu.frame)
24

```

La clase aplicación es la encargada de inicializar los marcos cuyo contenido están divididos en 3 el cual viene ser desde el menú, luego el ingreso de los datos hasta la creación de la tabla de frecuencia

➤ Controlador

El archivo Controlador se encarga de inicializar la clase aplicación luego inicializa para posterior ser llamado por un archivo main el cual llama la función de dicho archivo

```
1 from Codigo.vista.vista import Aplicacion
2
3
4 class ControladorEstudio:
5
6     def __init__(self) :
7         self.__vista = Aplicacion()
8
9
10    def controlador_iniciar(self):
11        self.__vista.iniciar()
```

➤ Main

```
1 from Codigo.controlador.controlador import ControladorEstudio
2
3 if __name__ == "__main__":
4
5     controlador = ControladorEstudio()
6     controlador.controlador_iniciar()
7
```

Es es el archivo main.py por el cual es el archivo principal y llama a la función de otro archivo para inicializarlo y dar inicio al programa.

7. RESULTADOS Y DISCUSIÓN

A continuación, se presentarán los resultados más relevantes del programa.

- **Ingreso de datos:**

INGRESE LOS DATOS PARA EL ESTUDIO	
Nombre del estudio	<input type="text" value="cesar"/>
Dato	<input type="text"/>
Cantidad del dato	<input type="text"/>

En este apartado no solo deja ingresar números, sino que también letras por si sus datos no son números, como requisito el nombre no debe de estar vacío y no puede agregar letras ni números positivos ni dejarlo vacío el apartado cantidad de datos, dependiendo del problema le saldrá una ventana emergente explicándole su error.

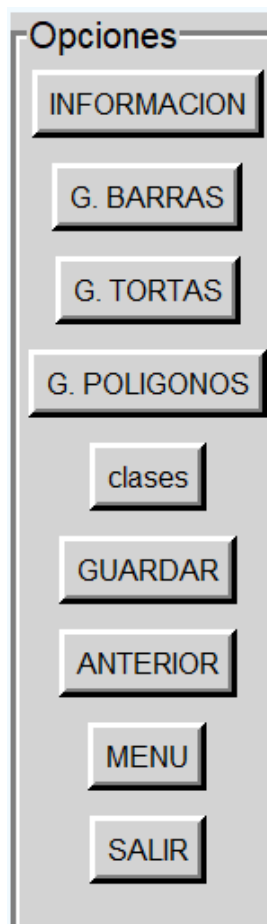
Así mismo estos datos se pueden modificar, eliminar después de ser agregados.

- **Tabla de frecuencia:**

Dato	F	FA	FR	FAR
4	6	6	33.33	33.33
5	9	15	50.0	83.33
6	3	18	16.67	100.0

En esta parte se encuentras los datos que se ingresaron, pero ya generados en una tabla de frecuencia, se puede crear con datos decimales, enteros, si se supera una cantidad de datos se generara la tabla, pero por clases si no es el caso se mantendrá esa forma también se admiten datos no numéricos

- **Múltiples opciones de representación**



Información: dará la información desde el nombre del estudio, la cantidad de datos tanto únicos como el total de ellos así mismo las medidas de tendencia central.

Gráficos: depende de los datos ingresados se grafico mediante la frecuencia de los datos y el dato o en caso de ser con clases se grafica mediante la marca de clase y su frecuencia

Clases: opción para agrupar los datos según la cantidad de clases que tenga el usuario permitido hacer.

Guardar: La opción guardar guarda la información que contiene la tabla de frecuencia generada y la exporta a un archivo Excel.

Anterior y Menu: Su función es que mientras el botón volver te devuelve al apartado anterior, el botón menú como su nombre dice te devolverá al principio, tomando en cuenta que si vas al menú se borrará todos los datos

Salir: Cierra el programa por completo

8. CONCLUSIONES

- **Principales hallazgos y conclusiones del proyecto:**

Importancia de la Planificación: La planificación detallada del proyecto, incluyendo la definición de requisitos, diseño de la arquitectura y la implementación paso a paso, ha sido crucial para el éxito del desarrollo.

Modularidad y Mantenimiento del Código: Diseñar el programa de manera modular ha facilitado el desarrollo y mantenimiento del código, permitiendo futuras ampliaciones y modificaciones de manera más sencilla.

Pruebas Exhaustivas: La realización de pruebas exhaustivas ha sido fundamental para asegurar la precisión y la fiabilidad de los resultados. Las pruebas unitarias y de integración han ayudado a identificar y corregir errores a tiempo.

Interactividad y Usabilidad: Desarrollar una interfaz de usuario intuitiva y amigable ha sido clave para asegurar que la aplicación sea accesible y útil para un amplio rango de usuarios, desde estudiantes hasta profesionales.

- **Futuras actualizaciones:**

Ampliación de Funcionalidades: Incluir funcionalidades adicionales como el análisis de regresión, pruebas de hipótesis, y otros métodos estadísticos avanzados.

Soporte Multilingüe: Desarrollar versiones de la aplicación en diferentes idiomas para aumentar su accesibilidad a nivel global.

Integración con Bases de Datos: Permitir la importación y exportación de datos desde y hacia bases de datos, facilitando el manejo de grandes volúmenes de datos.

Interfaz Web: Desarrollar una versión web de la aplicación, permitiendo su uso desde cualquier dispositivo con acceso a Internet.

Optimización de Rendimiento: Continuar optimizando el rendimiento del programa para manejar conjuntos de datos aún más grandes de manera eficiente.

Funcionalidad Completa: La aplicación cumple con todos los requisitos funcionales definidos inicialmente. Permite ingresar datos, calcular medidas de tendencia central, generar tablas de frecuencia, crear gráficos (barras, polígonos, tortas) y exportar los resultados a archivos Excel.

Precisión en los Cálculos: Los cálculos estadísticos realizados por la aplicación han demostrado ser precisos, verificados a través de pruebas unitarias e integradas.

Interfaz de Usuario Intuitiva: La interfaz, desarrollada con Tkinter, es fácil de usar y facilita la interacción del usuario con el programa, permitiendo una experiencia de usuario positiva.

Versatilidad y Aplicabilidad: La aplicación es versátil y puede ser utilizada en diversos campos, incluyendo educación, investigación y negocios, demostrando ser una herramienta útil para el análisis de datos.

En conclusión, el proyecto de la calculadora estadística no solo ha cumplido con sus objetivos iniciales, sino que también ha identificado varias áreas potenciales para futuras mejoras y desarrollos. La experiencia obtenida durante el desarrollo ha sido valiosa y ha sentado una base sólida para exploraciones y ampliaciones futuras en el campo del análisis estadístico.

10. REFERENCIAS

- Chungara Castro, Victor, 2007, estadística y probabilidades, 1ra Edición, Leonardo.
- Moya Rufino, 2007, Estadística descriptiva: conceptos y aplicaciones, 1ra Edición,
- <https://docs.python.org/es/3/>
- <https://docs.python.org/es/3/library/tk.html>