

# Diwali Sales Analysis Data

<sup>[1]</sup> Vaishnavi.S , <sup>[2]</sup> Jothi Sheeba.S

<sup>[1]</sup> Student: Department Of Mca, Er Perumal Manimekalai College Of Engineering(Autonomous) ,Hosur, Tamil Nadu, India

<sup>[2]</sup> Assistant Professor, Department Of Mca, Er Perumal Manimekalai College Of Engineering(Autonomous),Hosur, Tamil Nadu, India

---

*Abstract: The Diwali Sales Analysis Data program is an interactive data visualization and analysis tool that leverages Python's Tkinter, Pandas, Matplotlib, and Seaborn libraries. This program allows users to explore sales data from the Diwali festival through various interactive graphical representations. It is designed to provide insights into the purchasing trends during Diwali, focusing on demographic factors such as gender, age, marital status, occupation, and location (state).*

*The application begins by importing and cleaning the sales data, including handling missing values, converting data types, and renaming columns for clarity. Once the data is processed, users can access multiple visualizations, such as gender distribution, product category breakdown, age group analysis, state-wise sales, and marital status. These visualizations are created using Seaborn's count plots and bar charts, making it easy to spot patterns in the dataset.*

*The program is built with an intuitive graphical user interface (GUI) using Tkinter, enabling users to navigate through various analyses by clicking interactive buttons. Each plot is embedded within the GUI, allowing users to visually interpret trends and make data-driven decisions.*

*Overall, this program serves as a powerful yet accessible tool for analyzing Diwali sales, helping businesses and analysts gain insights into customer behaviour and sales performance during the festival period.*

---

## I. INTRODUCTION

The **Diwali Sales Analysis Data** program is a comprehensive tool designed to analyze and visualize sales data from the Diwali festival, one of the most important and widely celebrated festivals in India. The program utilizes Python's Tkinter library for creating a user-friendly graphical user interface (GUI), along with powerful data manipulation and visualization libraries such as Pandas, Matplotlib, and Seaborn.

Diwali is a peak season for retail sales, and understanding consumer behaviour during this time can provide valuable insights for businesses, marketers, and analysts. This program allows users to explore various aspects of the sales data, such as demographic breakdowns (gender, age group, marital status, and occupation), product categories, and regional sales performance (state-wise analysis).

The data used in the program is pre-processed to remove irrelevant columns, handle missing values, and format it for easy analysis. Once cleaned, the user can generate different types of visualizations, including bar charts and count plots, to uncover patterns and trends in the sales during Diwali.

The interactive GUI offers users the flexibility to choose from a range of analyses, making the program accessible to both technical and non-technical users. Whether for business intelligence, market research, or simple data exploration, this tool provides a clear and concise way to understand the dynamics of Diwali sales, aiding in more informed decision-making.

## II. SOFTWARE REQUIREMENTS

The **Diwali Sales Analysis Data** program provides an interactive platform for analyzing sales data using Python. It leverages libraries like **Tkinter** for the graphical interface, **Pandas** for data manipulation, and **Matplotlib/Seaborn** for creating visualizations. The program allows users to analyze sales based on various factors such as **Gender, Product Category, Age Group, State, Occupation, and Marital Status** through easy-to-use buttons and plots.

### KEY FEATURES:

- **Data Cleaning:** Unnecessary columns and missing data are handled using Pandas.

- **Visualizations:** Interactive plots like bar charts and count plots are displayed.
- **Modular Design:** Functions are used for each type of analysis, allowing for easy expansion.

#### TECHNOLOGIES USED:

- **Tkinter** for GUI
- **Pandas** for data handling
- **Matplotlib** and **Seaborn** for plotting

#### HARDWARE REQUIREMENTS

- Processor:** A minimum of 1.5 GHz dual-core processor (Intel or AMD recommended) for smooth operation.
- Memory (RAM):** At least 4 GB of RAM (8 GB or higher recommended for larger datasets).
- Storage:** A minimum of 100 MB of available disk space for the program and dataset, with additional space required based on the size of datasets or any generated reports.
- Display:** A screen resolution of 1366x768 pixels or higher for proper display of the graphical interface.
- Input Devices:** Keyboard and mouse for navigation through the interface.
- Internet:** An internet connection is not required for local use but may be necessary for downloading required libraries or updates.

#### TECHNOLOGY

- Python:** The core programming language used to build the application due to its simplicity and versatility in data processing and GUI development.
- Tkinter:** A standard Python library for creating graphical user interfaces (GUIs). Tkinter is used to build the main application window and provide interactive buttons for users to navigate between different visualizations.
- Pandas:** A powerful data manipulation and analysis library in Python. It is used to load, clean, and preprocess the Diwali sales dataset. Operations such as removing irrelevant columns, handling missing data, and converting data types are done using Pandas.
- Seaborn:** A data visualization library based on Matplotlib. It is used for generating attractive and informative statistical graphics such as count plots and bar charts to visualize key aspects of the sales data.
- Matplotlib:** A plotting library for Python that is used to create high-quality static, animated, and interactive visualizations. In this program, it is used to render the plots within the Tkinter GUI.
- CSV:** The sales data is stored in a CSV (Comma-Separated Values) file, which is loaded and processed using Pandas to extract relevant insights.

### III. EXISTING SYSTEM

The existing system for sales data analysis involves using separate tools like **Excel**, **Tableau**, or **Power BI** for data processing and visualization. Users manually clean and analyze data in spreadsheets or databases, then export it to visualization tools for chart creation. These tools, while powerful, are disconnected, requiring users to switch between platforms, which is time-consuming. Additionally, interaction with the data is limited, especially for non-technical users. There is no seamless integration between data processing and visualization, which hampers efficiency and limits real-time, dynamic analysis.

### PROPOSED SYSTEM

The proposed system integrates data processing, visualization, and user interaction in a single platform using Python libraries like Tkinter, Pandas, NumPy, Matplotlib, and Seaborn. Unlike traditional systems that require switching between multiple tools (e.g., Excel, Tableau), this system provides a user-friendly graphical interface where users can interact with the Diwali sales data and visualize different insights, such as gender, age group, product category, and state sales distribution. The system processes and cleans data efficiently using Pandas, while Matplotlib and Seaborn generate interactive and dynamic charts. This integrated approach improves the efficiency of data analysis, enhances user experience, and supports real-time data updates, enabling businesses to make data-driven decisions.

### IV. MODULES

#### 1. Data Import and Pre-processing

The first step in the program involves importing and cleaning the Diwali sales dataset. The data is read from a CSV file using `pandas.read_csv()`. The dataset is then inspected using methods like `df.Head()`, `df.Info()`, and `df.Describe()` to understand its structure, check for missing values, and confirm the data types of each column. Any irrelevant or unnecessary columns are removed using `df.Drop()`. The dataset is cleaned by handling missing data through `df.drop()`, ensuring no null values remain. Additionally, the Amount column is converted to an integer type using `as type('int')`, ensuring consistency across numerical fields. Lastly, the column Marital Status is renamed to Shaadi to improve clarity.

#### 2. GUI Setup Using Tkinter

The graphical user interface (GUI) is built using Tkinter. The main window is created with the desired dimensions (1000x500) and the title "...DIWALI SALES DATA ANALYSIS...". The window is then populated with various widgets such as buttons and labels. A function, `clear_widgets()`, is implemented to remove any previous widgets before displaying a new plot. This ensures that when a user navigates between different types of data analysis, the display is clean and only the relevant information is shown.

#### 3. Data Visualization Functions

□ **Gender Distribution Plot:** The `gender()` function creates a count plot to show the distribution of sales by gender. The `sns.count plot()` method is used, and the bars are labelled with the exact count using `ax.bar_label()`.

□ **Product Category Distribution:** The `product()` function generates a similar count plot but for the `Product_Category` column, displaying how sales are distributed across different product categories.

□ **Age Group Analysis:** The `age()` function visualizes sales across different age groups, separated by gender, using a count plot with the `hue` parameter set to `Gender`. This allows the program to display the age group distribution for each gender.

□ **State-wise Sales Distribution:** The `state()` function aggregates the total orders by State, sorting the states based on the highest sales. The resulting data is displayed using a bar plot to visualize the top states by sales volume.

□ **Occupation Distribution:** The `occupation()` function shows the distribution of sales based on the Occupation field, using a count plot similar to the others.

□ **Marital Status Analysis:** The `marital_status()` function visualizes the distribution of sales based on marital status, where the data has been renamed from Marital Status to Shaadi.

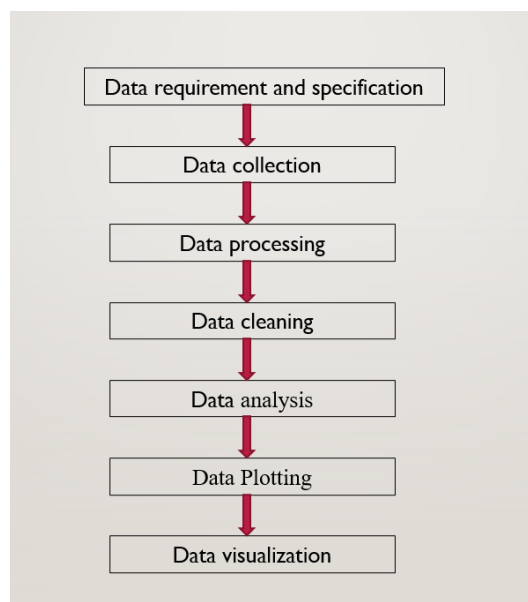
#### 4. Main Menu and Navigation

The main menu is a critical part of the program. It contains buttons that allow users to select which type of data visualization they wish to explore. The `homepages()` function sets up the main page with labelled buttons for each analysis type. Buttons are created for each plot (Gender, Product, Age, State, Occupation, and Marital Status) with corresponding commands to call the relevant visualization function when clicked. This modular approach to navigation ensures that users can easily explore different aspects of the Diwali sales data by simply clicking the relevant button.

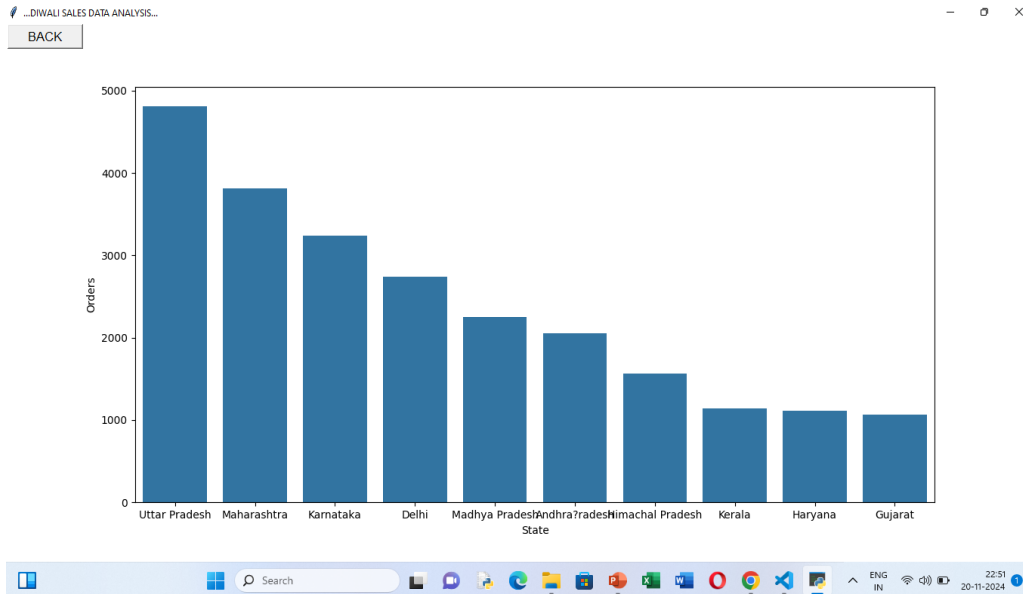
#### 5. Program Execution and Main Loop

After setting up the interface and visualizations, the program enters the Tkinter main event loop by calling `w.mainloop()`. This loop is necessary to keep the window open and responsive to user actions. It listens for events such as button clicks and updates the GUI accordingly. The main loop ensures that the program continues running until the user decides to close the window.

#### ARCHITECTURE DIAGRAM



## V. RESULT



This program is a Python-based interactive GUI application for analyzing Diwali sales data using Tkinter and data visualization libraries like Matplotlib and Seaborn. The program allows users to explore different aspects of the sales dataset by selecting from six analysis options: Gender, Product Category, Age Group, State, Occupation, and Marital Status. Each option generates a corresponding plot (e.g., bar charts, count plots) embedded in the Tkinter window. The data is cleaned and pre-processed before analysis, with null values removed and data types corrected. Users can navigate through the visualizations and return to the home page using a "BACK" button, or close the application with a "CLOSE" button. The program provides an intuitive way to explore and interpret sales trends during Diwali.

## VI. CONCLUSION


This program successfully provides an interactive and user-friendly platform for analyzing and visualizing Diwali sales data. By leveraging Python libraries such as Tkinter for the graphical interface and Matplotlib/Seaborn for data visualization, the application allows users to explore key insights across various dimensions, including gender distribution, product categories, age groups, state-wise sales, occupation, and marital status. The data pre-processing steps ensure that the analysis is based on clean and accurate information.

The intuitive interface, featuring interactive charts and the ability to navigate between different plots, makes the program accessible for users without requiring advanced technical skills. The inclusion of a back navigation feature and an exit button further enhances usability.

Overall, this tool can be valuable for businesses and analysts seeking to gain a deeper understanding of sales trends, customer demographics, and regional performance during the Diwali season. It serves as a powerful yet simple solution for data-driven decision-making and business analysis.

## REFERENCE

- Python Software Foundation. (2024). *Tkinter — Python Interface to Tcl/Tk*. Retrieved From <https://docs.python.org/3/library/tkinter.html>

- Matplotlib Development Team. (2024). *Matplotlib — Visualization with Python*. Retrieved From <https://matplotlib.org/stable/contents.html>
- Waskom, M. (2024). *Seaborn: Statistical Data Visualization*. Retrieved From <https://seaborn.pydata.org/>
- Pandas Development Team. (2024). *Pandas Documentation*. Retrieved From <https://pandas.pydata.org/pandas-docs/stable/>
- Oliphant, T. E., Et Al. (2024). *Numpy Documentation*. Retrieved From <https://numpy.org/doc/stable/>
- Matplotlib Development Team. (2024). *Embedding Matplotlib in Tkinter*. Retrieved From [https://matplotlib.org/stable/users/embedding\\_tk.html](https://matplotlib.org/stable/users/embedding_tk.html)
- John D. Hunter, et al. (2020). *Matplotlib: Python Plotting for Everyone*. Retrieved from <https://matplotlib.org/stable/contents.html>
-  Jake VanderPlas. (2016). *Python Data Science Handbook: Essential Tools for Working with Data*. O'Reilly Media.
- Sales Data Analysis, Kothari, C.R. (2004). *Research Methodology: Methods and Techniques*. New Age International.
- CSV Module Documentation, Python Software Foundation. (n.d.). *CSV File Reading and Writing*