

Instruction manual for Graphical User Interface



Table of Contents

[Introduction 1](#_Toc83722200)

[Overview of the study 2](#_Toc83722201)

[Dataset 2](#_Toc83722202)

[Simulations 2](#_Toc83722203)

[Loading dataset 2](#_Toc83722204)

[Data preprocessing 2](#_Toc83722205)

[Graphical User interface 2](#_Toc83722206)

[Visualization interface 2](#_Toc83722207)

[Result and analysis 3](#_Toc83722208)

[Instruction to run the code 3](#_Toc83722209)

[Technology stack and software used for performing the experiments 3](#_Toc83722210)

# Introduction

With the advancement in technology number of techniques are being developed that can be used for data mining. For that purpose, this study is conducted in which a graphical user interface is designed that will allow the user to generate information that can be used for data analysis. To obtain the results, tkinter python library has been used that provides functionality and also helps in designing the interface for better visualization to discover the significant insights and hidden features by observing the patterns. The user can select a specific time period in the GUI for most of those features. Furthermore, there are various options given to select the classes of the dataset to perform the analysis. The GUI provides dynamic features in which user can select various features and display visualized patterns and also discover the important insights to gather the necessary data. Moreover, the navigation button is added in order to perform all those functions analyze the visualizations.

## Overview of the study

This study has been divided into 3 sections - experimental analysis, designing graphical user interface and testing. Initially, the idea of the system was designed and various functionalities were written using python for the GUI and along with it backend processing was performed using pandas and Numpy. To provide the detailed instructions, user manual, testing report and summary have been prepared. In the end of the study, conclusion and performance of the model is described.

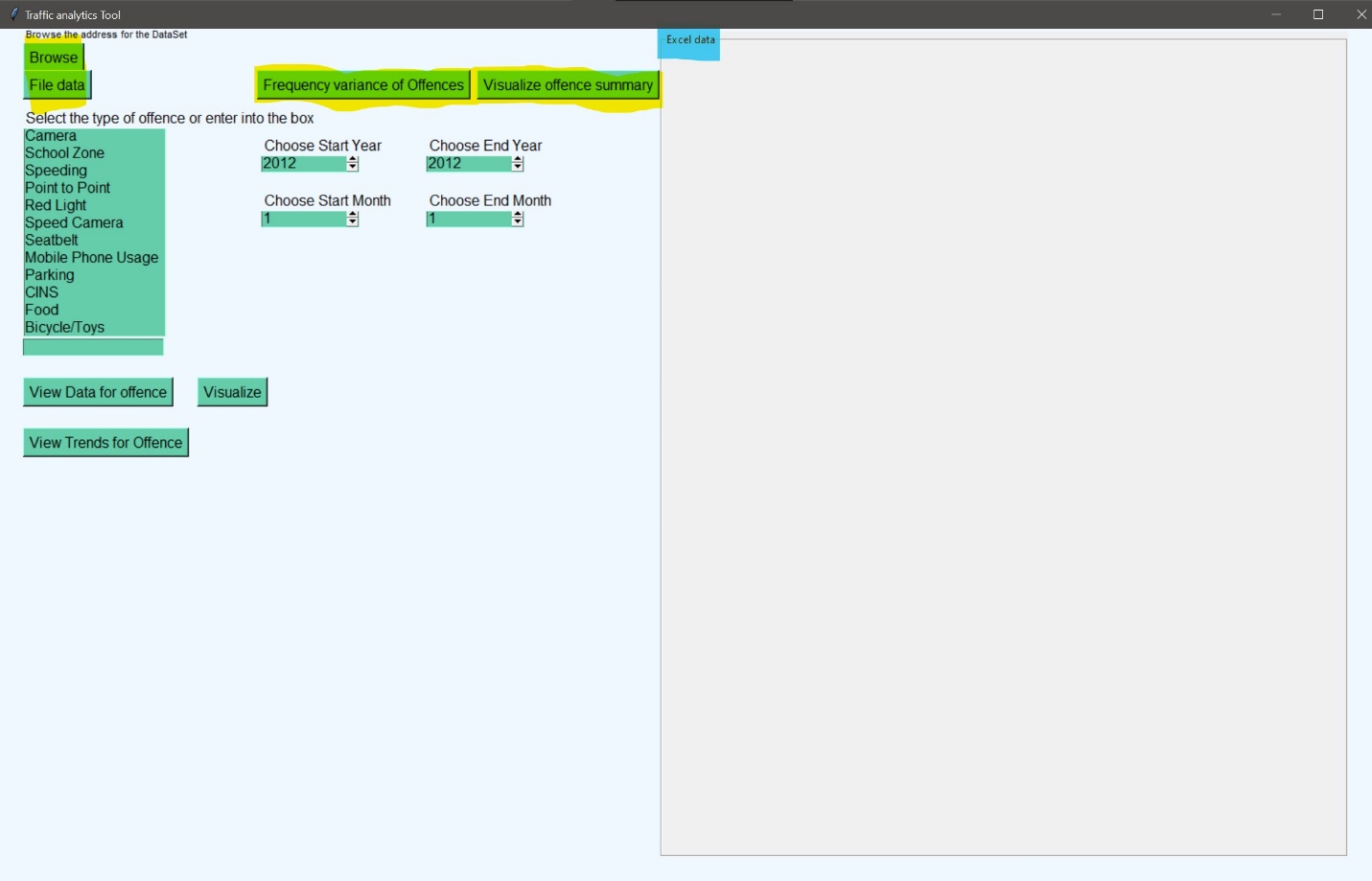
## Dataset

Secondary data has been used for this study which is downloaded from kaggle in the form of a CSV file. The link for the dataset for the analysis is- <https://www.kaggle.com/llihan/australia-nsw-traffic-penalty-data-20112017>.

# Simulations

## Loading dataset and data preprocessing

Initially the dataset is loaded using the browse button available over the GUI, which extracts the specific data from the .csv format. The dataset is then stored on the main memory and the whole attributes is visualized over the console.

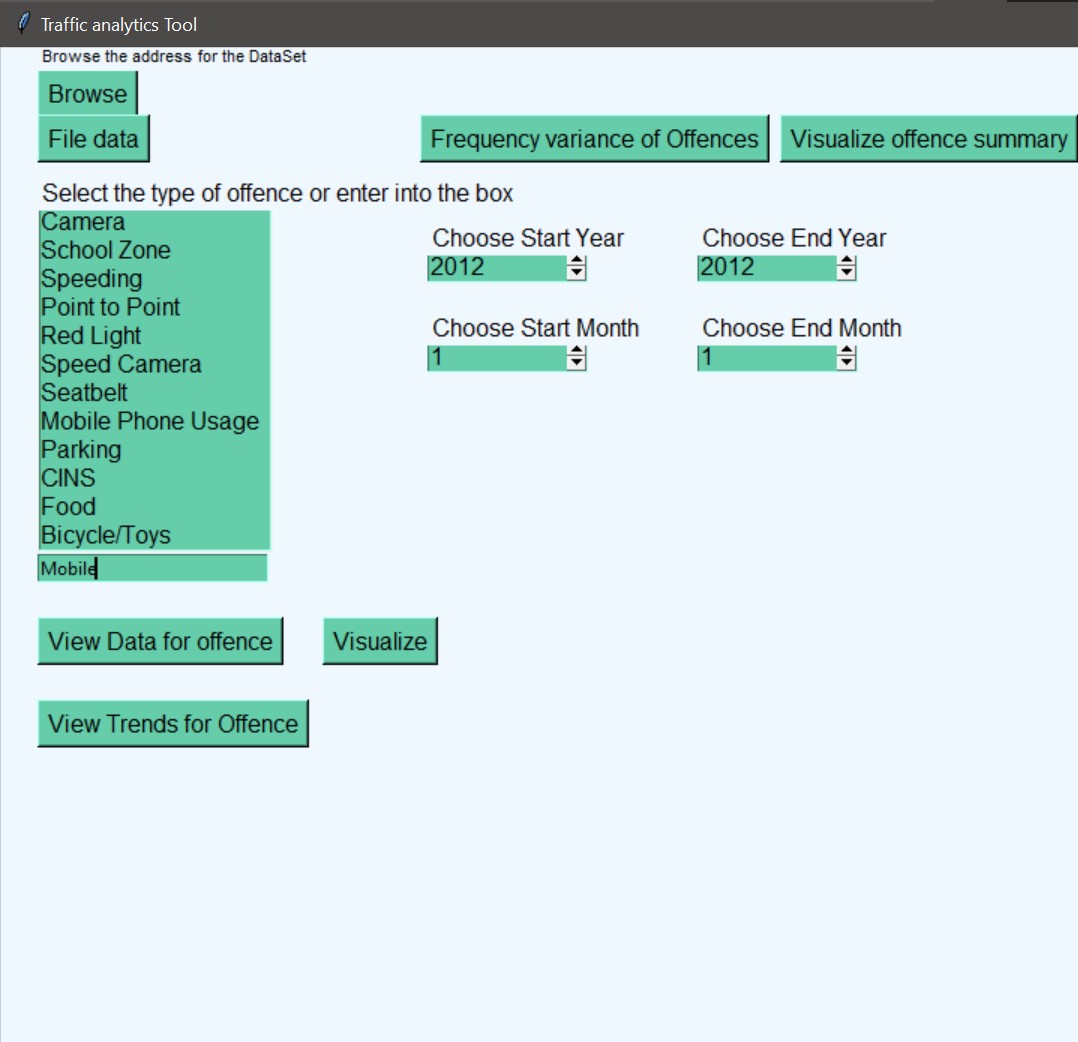


After taking input the data is cleaned using the data wrangling function in the code which all the nan, empty values are dropped and removed. Moreover, this helps in the processing each value of the attributes present in the data frame.

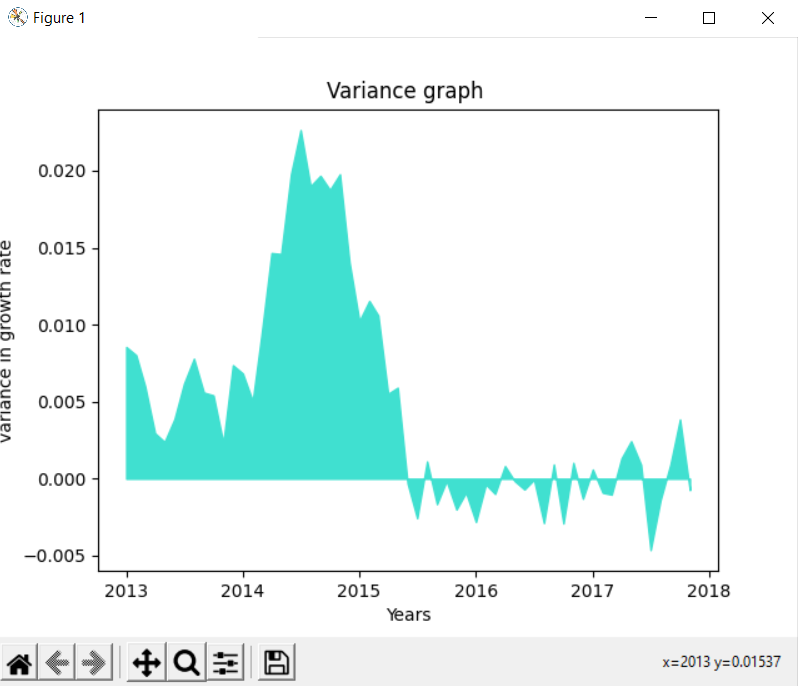
## Graphical User interface

GUI has various options and functionalities that can be seen on the interface which can be used for generating dynamic plots for various attributes. After the csv has been uploaded to through the browse button, backend code processing has been done including the data cleaning then these buttons can be used to display/analyze information.

**File Data-** This button is used to display the number or rows and columns in the excel sheet before the analysis.



**Frequency Variance of offences-** This will display the following graph with the variance in growth rate to the total years plotted as a histogram. The graph has been shown below.



**Visualize offence summary-** This button is used to display all trends plotted with different colored lines on the same graph with each colored line depicting a different tag against the years.

**Select type of offence or type the keyword in the box-** A specific offence code can be selected to show the trend after selecting the time period from the time period frame.

**Displaying the data on the excel data frame-** The date can be viewed for a particular tag in the excel data frame by clicking the *View data for the offence button.*

**Visualizing the data in a graph-** The data can be visualized in a graph after selecting a tag with the x-axis depicting the months and the y-axis showing the month and year in MM/YY format.

## Instruction to run the code

There are mainly four steps need to be followed to run the code and perform the visualization and analysis using the graphical user interface. The steps include:

1. Running the python script/file using the Spyder IDE, visual studio or windows command prompt.
2. After opening the GUI browse the csv file and load the data and the data wrangling is performed to remove the null or empty values.
3. Over the GUI, there is visualize button use it to perform the visualization.
4. For performing the analysis, shape can be observed of the dataset and moreover the various other attributes can be chosen from the offences table and can change the month and year to get the visualization for specific classes.
5. After performing the analysis, the cross option in the top right side of the GUI can be clicked to exit the interface.

# Technology stack and software used for performing the experiments

|  |  |
| --- | --- |
| **Technology stack used for building the GUI** | **Version** |
| Spyder IDE | 5.0 |
| Python | 3.8.6 |
| Matplotlib | 3.4.2 |
| Numpy | 1.21.1 |
| Pandas | 1.1.5 |
| Tkinter | 8.6.08 |