Data Analysis and Visualization Software

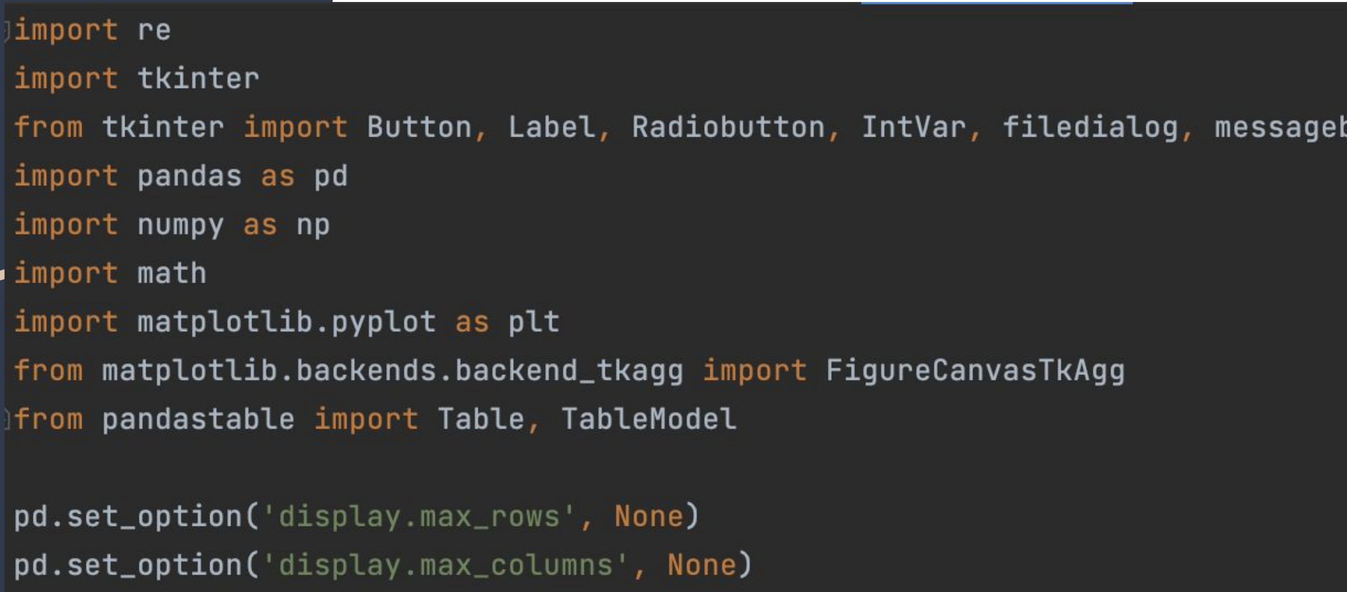
Sean Phelan

Long Term Goals and Future Applications

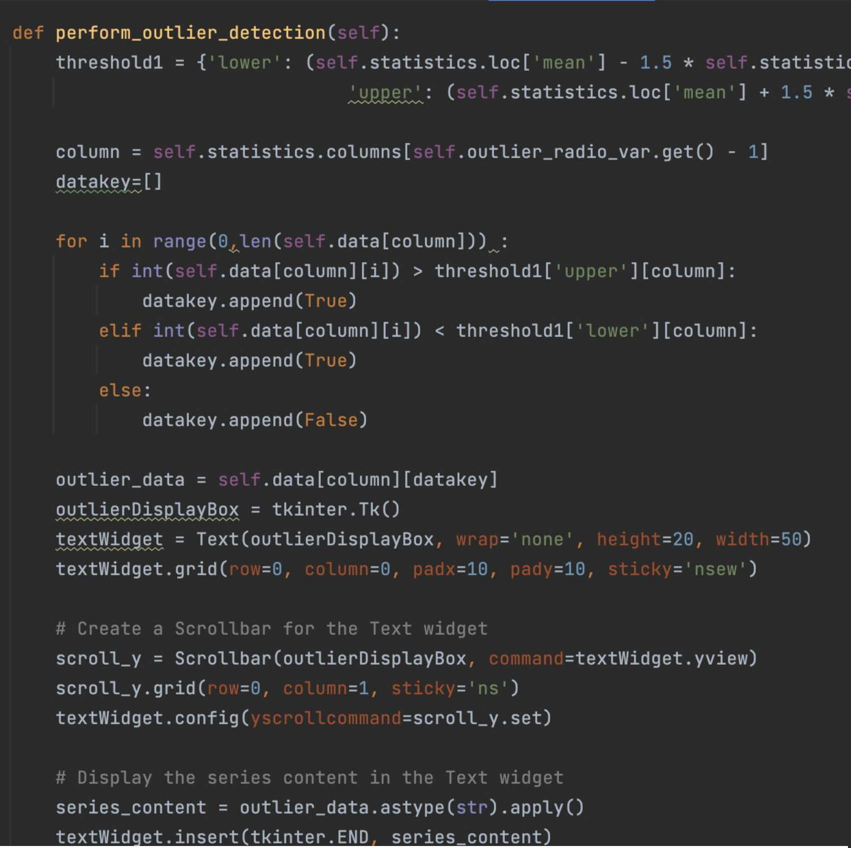
- I am a statistics major, and will be doing research next semester; both will require the analysis of many datasets.
- For data exploration and transformation, this software will speed up my workflow and allow me to quickly analyze trends within many datasets.
- Specifically, I will be able to quickly perform data cleaning and perform exploratory data analysis using this software.

Libraries Used

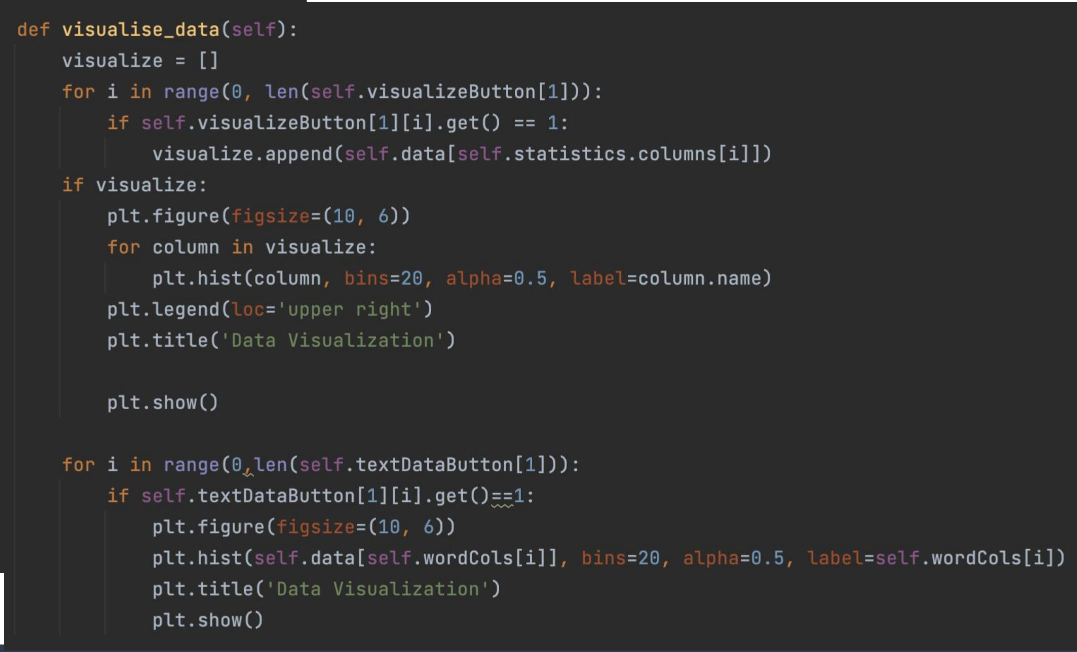
- Tkinter for gui
- Pandas for data management and analysis
- Numpy for calculations
- Matplotlib.pyplot for data visualization
- Re for text parsing



```
import re
import tkinter
from tkinter import Button, Label, Radiobutton, IntVar, filedialog, messageb
import pandas as pd
import numpy as np
import math
import matplotlib.pyplot as plt
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
from pandastable import Table, TableModel

pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)
```

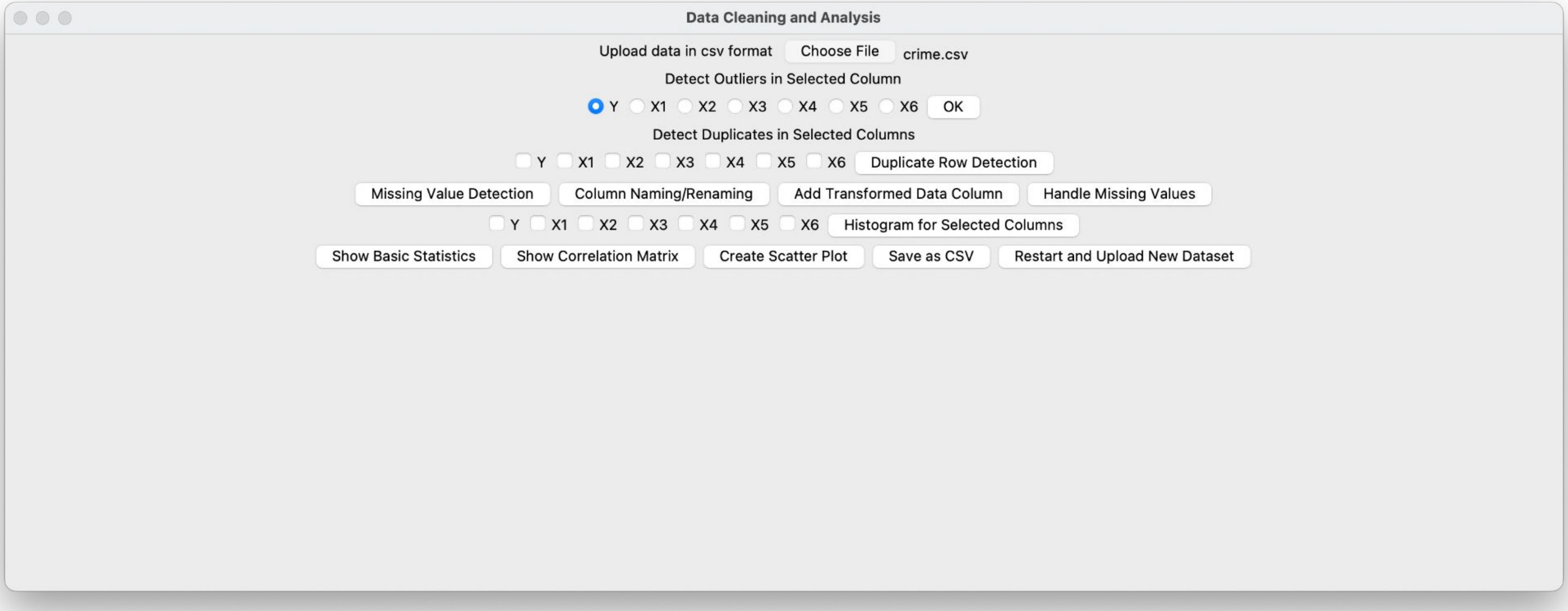


```
def perform_outlier_detection(self):
    threshold1 = {'lower': (self.statistics.loc['mean'] - 1.5 * self.statistics.loc['iqr']),
                              'upper': (self.statistics.loc['mean'] + 1.5 * self.statistics.loc['iqr'])}

    column = self.statistics.columns[self.outlier_radio_var.get() - 1]
    datakey=[]

    for i in range(0,len(self.data[column])) :
        if int(self.data[column][i]) > threshold1['upper'][column]:
            datakey.append(True)
        elif int(self.data[column][i]) < threshold1['lower'][column]:
            datakey.append(True)
        else:
            datakey.append(False)

    outlier_data = self.data[column][datakey]
    outlierDisplayBox = tkinter.Tk()
    textWidget = Text(outlierDisplayBox, wrap='none', height=20, width=50)
    textWidget.grid(row=0, column=0, padx=10, pady=10, sticky='nsew')

    # Create a Scrollbar for the Text widget
    scroll_y = Scrollbar(outlierDisplayBox, command=textWidget.yview)
    scroll_y.grid(row=0, column=1, sticky='ns')
    textWidget.config(yscrollcommand=scroll_y.set)

    # Display the series content in the Text widget
    series_content = outlier_data.astype(str).apply()
    textWidget.insert(tkinter.END, series_content)
```



```
def visualise_data(self):
    visualize = []
    for i in range(0, len(self.visualizeButton[1])):
        if self.visualizeButton[1][i].get() == 1:
            visualize.append(self.data[self.statistics.columns[i]])
    if visualize:
        plt.figure(figsize=(10, 6))
        for column in visualize:
            plt.hist(column, bins=20, alpha=0.5, label=column.name)
        plt.legend(loc='upper right')
        plt.title('Data Visualization')

        plt.show()

    for i in range(0,len(self.textDataButton[1])):
        if self.textDataButton[1][i].get()==1:
            plt.figure(figsize=(10, 6))
            plt.hist(self.data[self.wordCols[i]], bins=20, alpha=0.5, label=self.wordCols[i])
            plt.title('Data Visualization')
            plt.show()
```

We have many modules in this program, at least one per button


Data Cleaning and Analysis
Upload data in csv format
Choose File
crime.csv
Detect Outliers in Selected Column
● Y ○ X1 ○ X2 ○ X3 ○ X4 ○ X5 ○ X6
OK
Detect Duplicates in Selected Columns
□ Y □ X1 □ X2 □ X3 □ X4 □ X5 □ X6
Duplicate Row Detection
Missing Value Detection
Column Naming/Renaming
Add Transformed Data Column
Handle Missing Values
□ Y □ X1 □ X2 □ X3 □ X4 □ X5 □ X6
Histogram for Selected Columns
Show Basic Statistics
Show Correlation Matrix
Create Scatter Plot
Save as CSV
Restart and Upload New Dataset

Here is the main UI with a .csv file already loaded. Here are some of the button outputs.