import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

pd.set\_option('mode.chained\_assignment',None) # Turn Off Pandas Error Warnings due to chained assignments. Data Sanity Checks.

dataframe = pd.read\_csv("E:\Coding\Learning\BristolUniTerm1\HotAirBalloon\VelocityData.csv") # Creating a Pandas Dataframe from the csv file input

img = plt .imread("E:\Coding\Learning\BristolUniTerm1\HotAirBalloon\BristolMap.png") # Loading Image of Bristol for use in final plot as background

timestep = 1 # frequency of readings in minutes ( in this case every minute)

originx = 1 # This is the original Origin Value, This gets updated as the function below generates the new value of X. It does this iteratively and adds previous to next.

originy = 4.9 # This is the original Origin Value, This gets updated as the function below generates the new value of Y. It does this iteratively and adds previous to next.

dataframe["Vx (mph)"] = dataframe["Vx (mph)"]/(60/timestep) # Converting Column into Miles per Minute

dataframe["Vy (mph)"] = dataframe["Vy (mph)"]/(60/timestep) # Converting Column into Miles per Minute

for index, val in enumerate(dataframe["Vx (mph)"]): # Iterate through each value in the column Vx (mph). For each value add it to the previous value and replace the value with the new value

    dataframe["Vx (mph)"][index] = originx + val # Replace current value with new calculated value

    originx = originx + val # storing the current value of x, which will be used in the next iteration as the previous value of x

for index, val in enumerate(dataframe["Vy (mph)"]): # Iterate through each value in the column Vy (mph). For each value add it to the previous value and replace the value with the new value

    dataframe["Vy (mph)"][index] = originy + val # Replace current value with new calculated value

    originy = originy + val # storing the current value of y, which will be used in the next iteration as the previous value of y

x\_values = np.array(dataframe["Vx (mph)"]) # List of all the Vx Mph Values

y\_values = np.array(dataframe["Vy (mph)"]) # List of all the Vy Mph Values

final\_position = (x\_values[-1],y\_values[-1]) # Take Last Element in the array which is the final position of the balloon

print(f"Final Position of the Balloon is {final\_position}") # Print Variable Above

plt.rcParams.update(  # Updating Parameters of the Graph to look better

        {

            "font.size": 10,

            "xtick.major.size": 4,

            "xtick.major.width": 1,

            "ytick.major.size": 4,

            "ytick.major.width": 1,

        }

    )

plt.plot(x\_values, y\_values) # Plot a Graph of Value X against Value Y

plt.imshow(img , extent=[0, 12.5 , 0, 14.41]) # Create the Plot with the Background of the imported Image of Bristol Map, 12.5 and 14.41 are the dimensions of the map along the x and y axes in miles respectively.

plt.xlabel("X (miles)") # Creating Label for X axis

plt.ylabel("Y (miles)") # Creating Label for Y axis

plt.title("")

plt.grid(axis="both") # Creates grid in x + y direction - Can change to axis="y" or axis="x"

plt.show() # Render Plot to Screen

Map

Description automatically generated

Final Position of the Balloon is (5.698758333333334, 11.569535666666669)