# Documentation: Planning for drunks - Show me the way to go home...

## Summary

Version 1.0.0

This documentation aims to provide an overview of the agent-based model, DrunkModel.py, that was developed for Assessment 2 of GEOG5990M Programming for Geographical Information Analysis: Core Skills.

The model simulates a town of 25 drunk people trying to “stumble” back to their homes from the pub. It outputs a density map and .txt file to show where drunks stumbled on their way back to their homes.

It was developed in Spyder (Python 3.7) and the following documentation describes how to run it from this IDE.

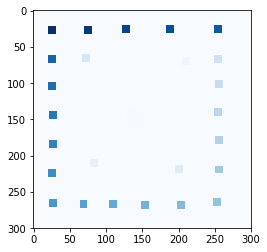
## How to Run

Files required to run the model:

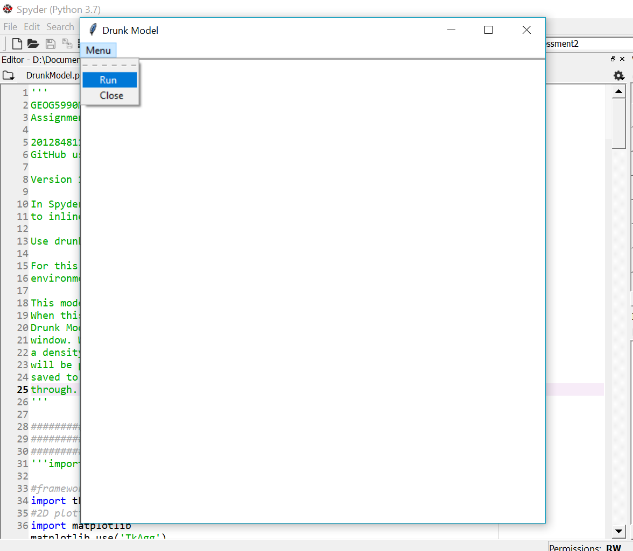
* DrunkModel.py – this is the main model from which the program is run,
* Drunkframework.py – this contains the Drunk class and outlines the characteristics of the drunks,
* town\_plan.txt – This is a 300 by 300 raster file representing the town in which the drunks stumble about in. The pub is denoted by 1s, the houses by the numbers 10-250, in increments of 10, and everywhere else, zeros.

In Spyder set Tools > Preferences > Ipython console > Graphics > Set backend

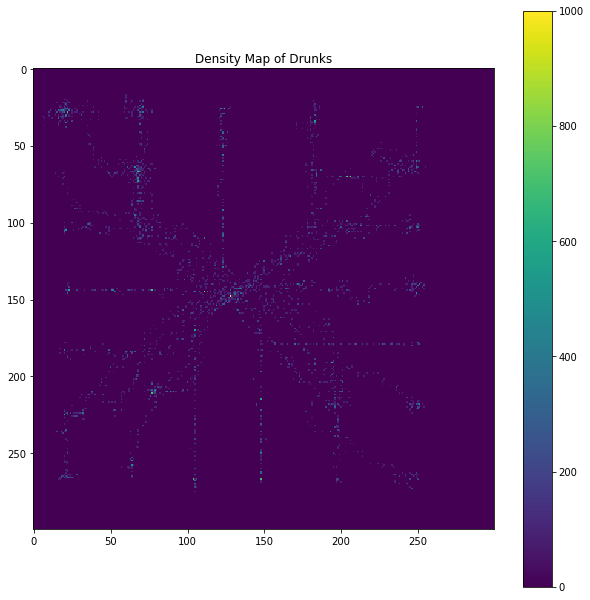
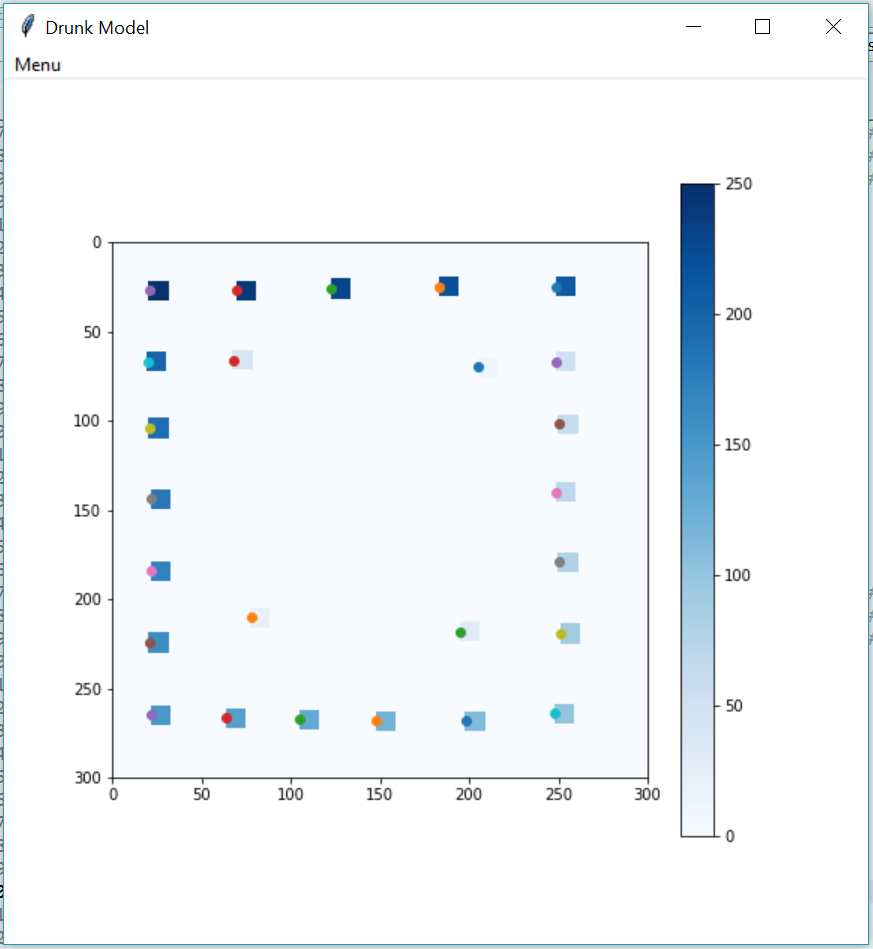
to inline.



For this model, town\_plan.txt has been used for the environment. The code allows for an alternative csv environment to be imported.

The simulation is run from tkinter GUI. When the code is run, a window will appear on the computer screen called Drunk Model. To run the model, click "Run" from the "Menu" in this window.

When the model has met the "stopping condition", where all the drunks have arrived home, close the window and a density map showing the points where the drunks have passed through, will be printed to the console and saved as density.png to the current directory, along with density.txt file containing the density points. 100 is added to every point the drunks pass through.



## Program Description

This program does the following:

1. Pulls in a csv/txt file with values for an environment and finds out the pub point and the home points.
2. Displays the pub and homes on the screen.
3. Animates the drunks leaving the pub and reaching their homes.
4. Stores how many drunks pass through each point on the map to a density.txt file.
5. Displays the density of drunks that have passed through each point on the map and saves it to density.png.

There are 25 drunks, each of which have homes assigned to them according to their number. Houses are labelled 10-250 in increments of 10. The drunks randomly “stumble” left, right, up and down, back to their homes from the pub based on conditions met in a for loop. An element of bias is introduced to give the drunks some direction to their homes.

#Can change number of places moved by drunk. Currently a random number

#between 1 and 5.

unitsMoveBy = random.randint(1,5)

#x coordinate of drunk plus a number.

moveR = self.\_x + unitsMoveBy

#x coordinate of drunk minus a number.

moveL = self.\_x - unitsMoveBy

#y coordinate of drunk plus a number.

moveUp = self.\_y + unitsMoveBy

#y coordinate of drunk minus a number.

moveDown = self.\_y - unitsMoveBy

#Distance between drunk's current position and drunk's home.

currDist = self.home\_distance(self.houseCoords, self.\_x, self.\_y)

#Potential distance to drunk's home after move taken below.

moveRDist = self.home\_distance(self.houseCoords, moveR, self.\_y)

moveLDist = self.home\_distance(self.houseCoords, moveL, self.\_y)

moveUpDist = self.home\_distance(self.houseCoords, self.\_x, moveUp)

moveDownDist = self.home\_distance(self.houseCoords, self.\_x, moveDown)

#move along x axis i.e. right or left.

#if random number produced (between 0 and 1) is less than 0.5

if random.random() < 0.5:

#and if value of the environment at the location of the drunk

#plus a number on the x axis is equal to 0 or house location

if moveRDist <= currDist or random.random() < self.randomness:

#then drunk moves right

self.\_x = moveR

elif moveLDist <= currDist or random.random() < self.randomness:

#then drunk moves left

self.\_x = moveL

if random.random() < 0.5:

if moveUpDist <= currDist or random.random() < self.randomness:

#then drunk moves up

self.\_y = moveUp

elif moveDownDist <= currDist or random.random() < self.randomness:

#then drunk moves down

self.\_y = moveDown

#print(self.houseno, currDist) #test

#Adds 100 to environment at every step

#n.b. y, x not x, y to be inline with reversed environment axes

self.route\_environ[self.\_y][self.\_x] += 100

The basic algorithm is, for each drunk (who will have numbers between 10 and 250 assigned before leaving the pub), move the drunk randomly left/right/up/down in a loop that picks randomly the way it will go. When it hits the correctly numbered house, stop the process and start with the next drunk. At each step for each drunk, add one to the density for that point on the map.

Additional marks are awarded for the following.

Stopping the drunks from retracing their steps, or other ways of helping them find their way home.

There are 25 drunks (house numbers 250, 240, 230...etc...10). Each drunk should be given a number before leaving the pub, and they will need to find the home with the same number in the town plan.

300 x 300 pixel raster file of the town plan: [drunk.plan](http://www.geog.leeds.ac.uk/courses/computing/study/core-python/assessment2/drunk.plan) ([GIF version for comparison](http://www.geog.leeds.ac.uk/courses/computing/study/core-python/assessment2/drunk.gif) - this should not be used in the project). Each line in the file is a line in the raster image, starting at the top left corner. The backgroung is denoted by the number zero, the pub by one's, and the houses by the other numbers.

## Development and Issues

This program was developed using code from my Assessment 1 agent-based model for this module. That model can be seen on my GitHub repository at: <https://github.com/hannahwh05/GEOG5990M_Programming>.

## Information about the Developer

I am a current University of Leeds MSc GIS Student (2018-19) learning core skills in Python. I graduated from University of Southampton in 2016 with a 2:1 in BSc Geography.

Link to my website: <https://hannahwh05.github.io/>