MATH 300 Final Project

June 19, 2020
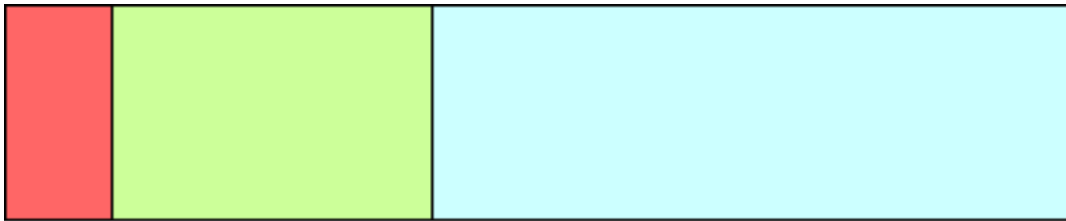
Team Members

1. Ali Hamza | 05084

2. Maham Shoaib | 04911

3. Hassan Naseem | 05102

1 Tasks

1.1 Task 1

In this section we made a function $task1(steps, start, leftprob, stopprob)$. The function takes steps using an algorithm that makes use of an algorithm that can be explained using the following diagram.

Let us assume

- The red region is the probability of not moving

- The green region is the probability of moving left

- The blue region is the probability of moving right

The algorithm simply checks if the randomly generated numbers $0 \le n \le 1$ lies within a certain region and makes movement accordingly.

Running the algorithm on multiple steps the following graphs were obtained. To find the expected average distance at a certain step the we ran multiple simulations of the code and calculated an average step at each step. This rendered the following random walk that is made up of average steps. This is then an expected random walk.

1.2 Task 2

In task 2 we took a similar approach as task 1. The only difference was that we simulated a random walk for two particles on the same graph. However, the same issues of randomness arose and it was impossible to predict the average step at which the two particles met.

Similar, to task 1 we simulated multiple random walks of the two particles and calculated the average step at which the particles were meeting for a certain number of steps and distance. The data of each intercepting step was recorded into an array and the average was calculated to find an expected intercepting step.

We also plotted the average random walks of the the two particles and the result of that was as follows: It is worth noting, however, that this graph is also subject to change based on different parameters. However, it seems to be stable on multiple executions. There is some deviation though.

1.3   Task 3

In order to determine the step size and the orientation, we assumed that the step size is a discreet random variable between 0, 0.5, 1 and orientation is a discreet random variable between 0, $\frac{\pi}{2}$, π, $\frac{3\pi}{2}$, 2π. We have used directional words such as right referring to orientation of 0 as well as 2π and up referring to $\frac{\pi}{2}$, etc.

We have used an algorithm similar to those used in the previous tasks to determine the probability of a step size and orientation being used. In the case of the particle exceeding the 100 $unit^2$ region, the particle just reverses the direction that it was moving and bounces back into the region. The trajectory is then plotted on a 3 dimensional plane. This can be seen in the following graph where the region is being exceeded and a curved outline can be seen, showcasing the polar movement of the particle.
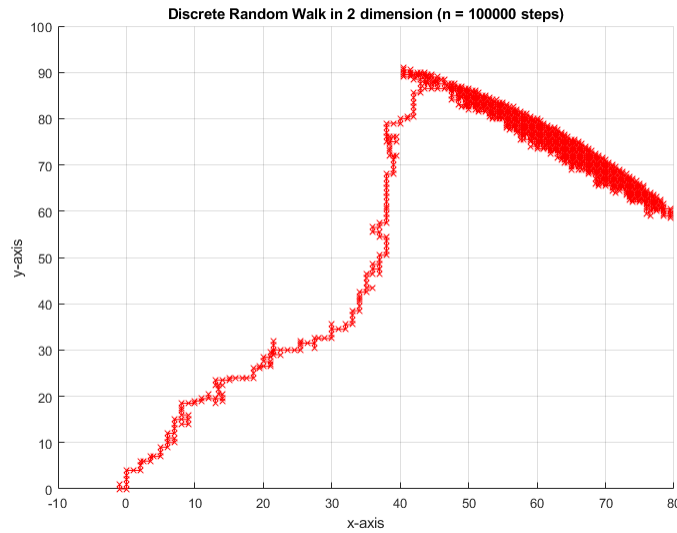
1.4   Task 4

For this section we have repeated a similar procedure to that used in Task 1 except that in this case the step size is a continuous random variable between 0-1 determined by the built-in **randi** function that chooses a a number within the range with a uniform probability distribution.

1.5   Task 5

This section is done in a similar manner to that of Task 3, however in this case both the step size and orientation are continuous random variables between 0-1 and 0-2π respectively. The variables were chosen in a similar manner to those in Task 4 by employing the **randi** function.

1.6   Task 7

In this task we have used a similar algorithm to that in Task 3 with the only exception that the orientation is a continuous random variable between 0-2π which is chosen in a similar manner to that in Task 4 by employing the **randi** function.

1.7 Task 8

Task 8 had many similar attributes to the task 2. Therefore, we used a similar approach to solve this problem. Using the random walk algorithm in task 5, we generated the random walk data of two particles and stored the data in a 3 dimensional array. Where we stored the (x, y) coordinates at each step of each particle. This allowed us to run a simulation to find an average step at which the euclidean distance of the two particles was ≤ 1. Then we further ran a simulation to find multiple expected aforementioned steps. This allowed us to model a distribution of the means. This produced a trend the show cased the mean variance in the means of the step at which the two particles have distance of 1 unit or less. This can be visualized in the following graph
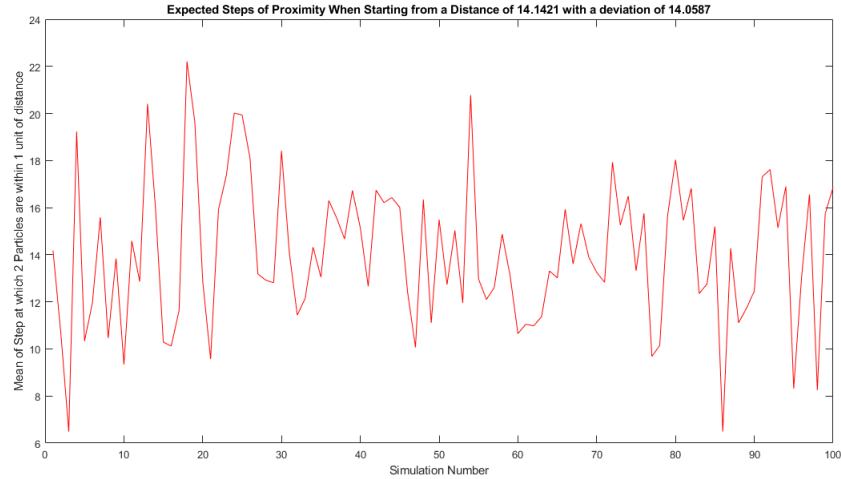
1.8 Task 9 Bonus

This task is a modification of task 8 for a specific application that is to capture the visual depiction of the Social Distancing aspects; an exercise that is being prescribed to all human beings during the Pandemic caused by the novel Covid-19 corona virus. The modifications made to Task 8 to achieve Task 9 is done by increasing the number of nodes where essentially the nodes represent people and this time the nodes are kept a distance of more than 6 units away from each other with the assumption that 1 unit maybe thought of as 1 feet, which was the condition kept to find the average number of steps. Our model can be used to calculate the probabilities of n number of nodes that are within 6 units of distance of contracting the virus. We carried out this task on 3 and 4 number of nodes and came up with the conclusion that as the number of nodes increase the probability of contracting the virus increases becuase the greater the number of nodes, the more closer they are likely to come towards each other in a confined space

2 Bibliography

1. https://www.mathworks.com/help/matlab/math/multidimensional-arrays.html

2. https://www.mit.edu/~kardar/teaching/projects/chemotaxis(AndreaSchmidt)/random.html

3. https://www.researchgate.net/publication/269284111_On_time_scale_invariance_of_random_walks_in_confined_space

3 Extras

1. Simulation video of random walk in 2D: https://youtu.be/99tQ8SE9IXw

4 Github Repository

1. https://github.com/hurryingauto3/ProbabilityProject

5 Work Distribution

Team Member	Work Done
Ali Hamza	Tasks 3, 4, 5, 8 + Report
Maham Shoaib	Tasks 1, 2, 7, 8 + Report
Hassan Naseem	Task 9 + Task 9 Report