

# Probability Project Observations

Omema Ahmed oa04320

Aiman Junaid aj05161

Fatima Zehra fz04316

June 19, 2020

## Task 1

To find the expected distance from the starting point (i.e. origin by default or the given coordinates), we ran simulations for 1000 walks and calculated their average. Our function increments the distance of a random walk at each step, which is 1 in our case, and returns its sum. Negative coordinates denote movement on left of origin, and positive denote movement on right. To move left, we subtract 1 from x, and to move right we add 1. The shape of our resulting graph was a Gaussian curve, which is because binomial converges to Normal distribution. We tested our model for a range of probabilities, and each gives a different expected distance based on their probabilities and starting position.

## Task 2

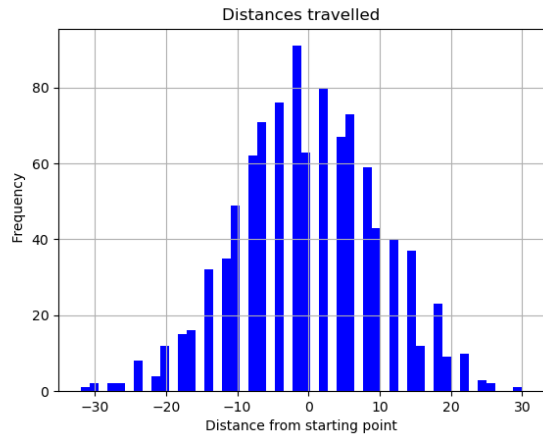
To calculate the expected time for two random walks to meet, we calculated the total steps it took both of them to reach the same x-coordinate, and ran simulation for 1000 walks to take average. In our function, we started one walk from  $x=0$ , and randomly assigned the starting position of second walk. We tested our model for unequal probabilities, for one position being fixed (not moving), and for different starting position of both walks. The resulting graph displays that the time it took for most of them to meet, was within 5000 steps.

## Task 3

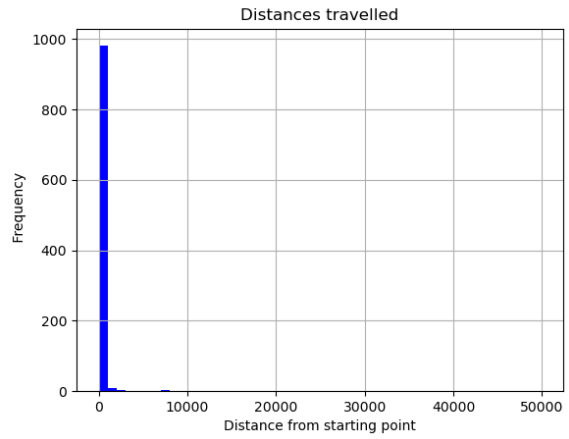
To simulate a 2D random walk, we used matplotlib's animation. We assumed that after reaching the boundary of the circle, if the next position of the walk is beyond the boundary, it will reflect back with the distance between the boundary and the new point.

## Task 4

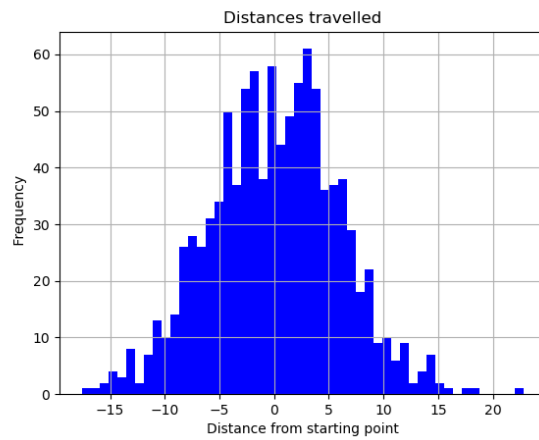
We used a uniform random variable, and calculated the step size that ranged between 0-1, using numpy's `random.uniform()` function. Our resulting graph was a Gaussian curve, but in comparison with **Task 1**, it is smoother as we used continuous values for step. Again, we calculated the expected distance using the same method as explained earlier for **Task 1**.



(a) Task 1



(b) Task 2



(c) Task 4

Task 5

Task 6

Task 7

Task 8

Graphs

Bibliography