CSCE 5210 SECTION 002
FUNDAMENTALS OF ARTIFICIAL INTELLIGENCE
(FALL 2022 1)

PROJECT-1 MILESTONE 3

MILESTONE 3 R4:

In this Milestone, we need to make a report on how the time taken for journeys is affected by change in parameters like:

- Connectivity of the Graph
- Change of Standard Deviation of the Normal Distribution Function used for p.

We also need to explain why the changes occur.

a) CHANGING THE CONNECTIVITY PARAMETER OF GRAPH:

According to Project-1 Specification we increased the connectivity parameter of the graph from 0.03 to 0.06. It was doubled in this case. We made this change by changing the value of (probability of edge creation) parameter in gnp_random_graph() function.

When this change was made we saw a significant decrease in the time taken for the random journeys that are generated.

This can be shown by the two images below: -

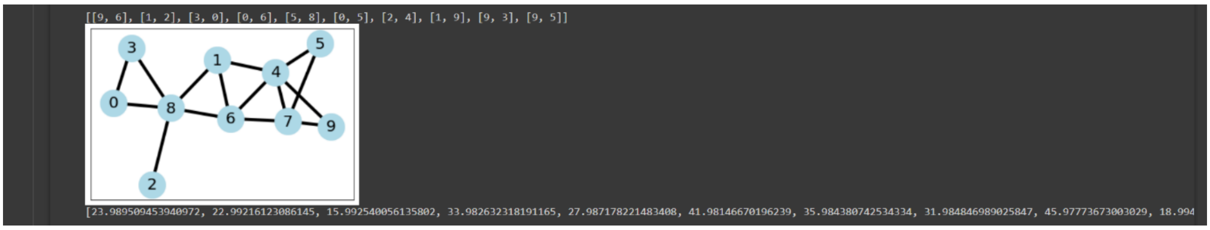


Image of Output with 0.03 connectivity

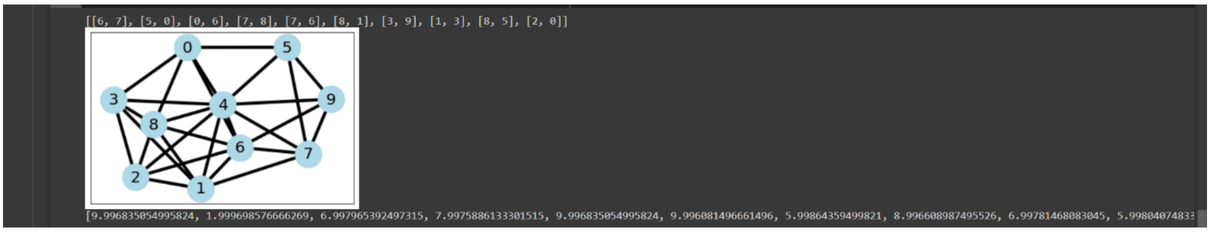


Image of output with 0.06 connectivity

NOTE: For sake of better understanding a graph with only ten nodes is used to demonstrate the change and the labels are not printed to get a clear view of the graph.

The first line in each of the image is a list of 10 random journeys generated at a random time. The image is the Plotted Graph. The next line shows the time required to complete the above journeys at some point.

Explanation:

The main reason as to why the change in parameter led to a decrease in time taken for the journeys is because of the increase in the number of paths available to take to get from Point A to Point B in the graph.

The increase in number of paths available means we have more options to choose from. We know that the weights are randomly assigned, which means that if the graph contains more lesser values, then the Dijkstra Algorithm obviously selects the lesser values over the larger ones, which is precisely what happened here.

Since weights for the edges are assigned randomly and there is an increase in the number of edges, the chances for the edges being assigned a smaller weight also increases which means that we have more lesser weights to choose from, than before the change in parameter.

To illustrate this point, assume there are two nodes with a single edge between them with a weight of 100.

Now we add another edge between the two points with a weight that is randomly selected between 1 and 100.

Before adding the second edge our only option to traverse between the two nodes is to use the edge with the weight 100. Now that we have another edge we can choose the best between the two. Even if the second edge is randomly assigned a weight of 99 it still results in a decrease even if it is one unit. If the randomly assigned weight is even lesser we can expect a further decrease in time to traverse between the two nodes.

If we keep on adding edges between the two nodes, then the random assigner might even produce an edge with a weight of 1 unit.

So to sum up the explanation the increase in number of edges which also increased the chances of getting edges with lesser weights is the reason as to why the time taken for journeys decreased significantly.

b) CHANGING THE STANDARD DEVIATION OF THE NORMAL DISTRIBUTION USED FOR “p”:

According to Project-1 Specification we decreased the standard deviation of the Normal Distribution that is used to generate p from 0.25 to 0.125. It was halved in this case. We made this change by changing the value of (standard deviation) parameter in random.normalvariate() function.

When we change the standard deviation and look how overall time is affected we cannot compare the two cases.

Explanation:

The change in parameter affected how the weights are updated for every 15 minutes in the clock as the percent of change is calculated by:

b*p*w

where:

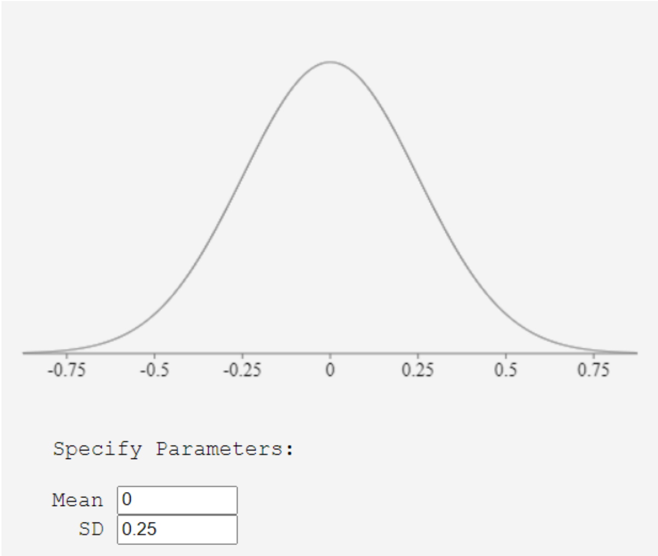
- b is base value for a time slot
- p is a random value returned from Normal Distribution
- w is the weight of edge

In this we know that b is same over an hour and w is always constant for an edge. What changes every 15 Minutes is p. So the majority of the changes that occur in weights is due to the change in p.

The change in Standard Deviation for the same mean results in change of range of the values that the Normal Distribution can take. In this case the range of values that the normal distribution takes is reduced.

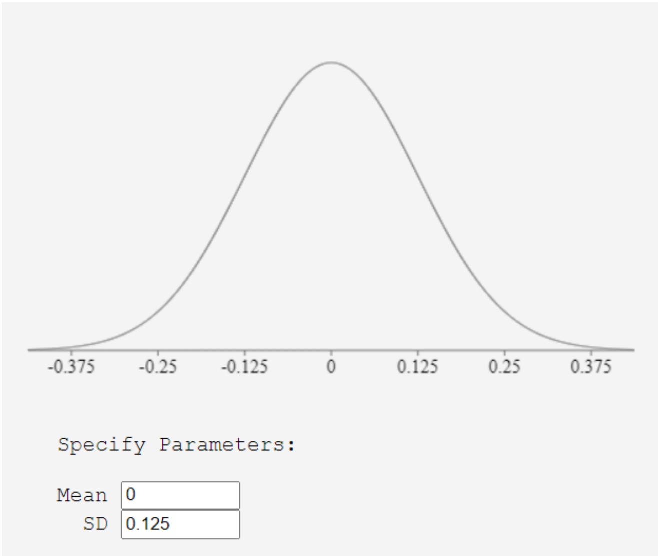
With reduced range we can say that p will take lesser values more often and therefore the percent of change in weights for every 15 minutes will decrease and hence the chances for requiring a new path will also decrease.

Take a look into the following images below:



Normal Distribution Curve with

Mean=0, SD=0.25



Normal Distribution Curve with

Mean=0, SD=0.125

By taking a look at the two curves although the shape remained same we observe that the range of values that the curve takes decreases with decrease in Standard Deviation.

What this means is that the random.normalvariate() function has higher probability of returning a smaller value (closer to 0) when standard deviation is 0.125 than compared to when the standard deviation is 0.25.

This means that percent of change (b*p*w) has a higher probability of being smaller. If that is the case, then changes that occur in the weights will also be less. This means that whether the time increases or decreases the change will be small.

Therefore, new values of total time taken either if it increases or decreases will not be much different from the current optimal value and since we change paths only when we cross a threshold of 5% change, the chances of this happening reduces and hence the journey paths will not change that frequently.

Of course since p is generated randomly there might be rare cases where p is very large relatively and hence can bring pretty big differences between new and old times whether it increases or decreases. In such cases, the journey paths might change.

This is what happens when we are comparing the times for every 15 minutes in the same graph i.e, for the graphs with same edges and weights, but when we compare two completely different graphs we cannot compare the two.

The reason why we cannot compare the two graphs based on time taken for journeys to complete for a path is because, the time taken mainly depends on weights of the edges and since weights are assigned randomly we cannot make comparisons between the two.

So to summarize we need to consider two cases when we want to see how time is affected by change in standard in deviation:

1) First, if the journeys that we are comparing are made on the same graph that is the weights and edges do not change. Even in this case what is affected is only how the weights are updated every 15 minutes and how frequently the paths are changed. The reason is that overall time depends mainly on the weights of the edges. By changing p, we only change by how much the weights will change for every 15 minutes.

2) Secondly if the journeys are made on different graphs that is, weights and edges are different. In this case, we cannot make any type comparisons as changes in weights are affected by both p and weights and overall time taken depends on weights and weights are assigned randomly.