Lecture – 51
Strong and Weak Relationships (Continued) and Homophily
Fatman Evolutionary Model - Implementing Closures

We next want to implement closures in our model, whereas we have seen there are 3 kinds of closures; triadic closure we both have a common friend we become friends.

(Refer Slide Time: 00:13)

Foci closure; we both are going to the same gym, so we become friends; membership closure I am your friend, you go to gym so I start going to gym. Now, how do we implement these closures in our model is now the question. So, how have we implemented these closures before? So we have used a formula if you remember.

So, what we have done is; if there are 2 people A and B and these people have let us say some K number of common friends, then assume that each of these friends; each of these common friends contributes to a probability p. So, one common friend contributes to a probability of p that these 2 people will become friends. Assume the; it is 0.4, so one person says that with 40 percent probability, these 2 people A and B will become friends and then we looked at this formula that the probability that these 2 people will become friends in the next iteration is 1 minus; 1 minus p to the power k.

So, we will be using the same formula for implementing all 3 kinds of closures. Now, how do we implement it? First we will look at every 2 possible combination of nodes. So, when we pick 2 nodes from this network; we will pick every possible pair of 2 nodes. So, when we pick these 2 nodes from the network what can be their types; so, both of these can be the person nodes; person A, person B or one can be a person node and another can be a foci node.

So person and foci node or vice versa foci node and person node, so, mainly one person node and one foci node and in the last case both of these can be the foci node. How do closures happen in all of these 3 cases? So, in the first case when there are 2 people node; so if we look at the common neighbors, so these 2 nodes. So, we know that in our model the common neighbors can be a person as well as the foci. So, we use the same formula for both of these things, so we capture both the triadic closure and the foci closure using the same formula.

So, we assume that whether it is a person which is your common friend or it is a foci where both of these are going contributes to a probability p of them becoming friends. So, as I told you that this is a very; we are trying to keep things very simple here. So, once we write the complete coding; we can actually change both of these probabilities, we can assume that or we can code it that if there is a person node; it has a higher probability of making them friends and if it is a foci node, it has a less probability of making them friends, but for now we keep both of these the same. So, whether it is a common friend or common foci; it contributes to a probability p of both of these becoming friends and then again we use the same formula. So, that formula helps us in achieving both things here triadic closure and foci closure.

(Refer Slide Time: 03:38)
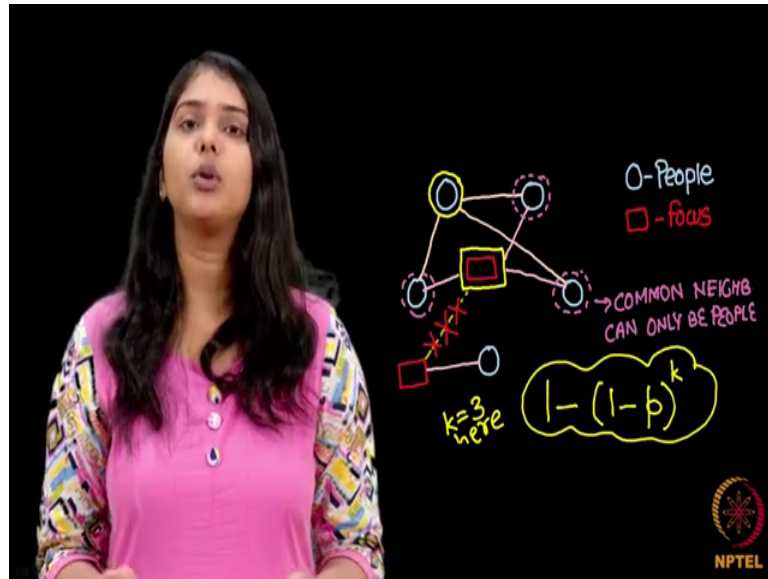
Let us now look at the second case, so here is a person and then here is a social foci; now we look at their common neighbors. Now, if you look at their common neighbors; you observe that their common neighbors can only be persons, their common neighbors cannot be foci, why, because there are no edges between the foci nodes, so 2 foci nodes are never connected to each other, it is the edges are only between person to person and person to foci. So, we have a person and we have a foci here and all their common neighbors are different-different persons. So, this means that this person here can have k of his friends going to the social foci and again we assume that each of these common friends contributes to a probability of p.

For this person also to go to this social foci and again we use the same formula. So, you see how we just use one formula for implementing all these 3 kinds of closures; once we call it we can actually change the things here and make it more advanced. Next, we want to implement closure property in this graph, in the meantime; we comment the homophily. So, that we can clearly see what is happening in closure and we have here closure G.

So as we have discussed there are 3 kinds of closures which we have to implement become here define closure G and what you have to do for closure is we know that in one iteration everybody is going to look at their common friends to any 2 people or any 2 people there going to look at all their common friends and they are going to decide whether in the next iteration, they are going to connect to each other or not.

So, what we do here is take an array let us say array 1 and this array will have 3 values. So, how will this array look like; this array will be composed of sub arrays and each sub array will be having 3 values. The first value is the first node, second value is the second node and third value is the probability of connection. So, when we are going in an iteration of closure G; we do not keep adding edges side by side like we did for homophily. Because, if you add an edge between you looked at 1 and 2 and you added an edge between them, it can have complications for the addition of the coming up edges.

So, the addition of every edge should be independent of what has been added before. So, to maintain that independence we have here this array; array 1 and so these are the kinds of values this array 1 will holds, so here it can be 2, 5 and then 0.7 and so on. So, how do we go about doing this, so initially we have this array 1 which is empty and then now by after looking at all the nodes and their common neighbors, we have to decide whether we have to add an edge between them or not.
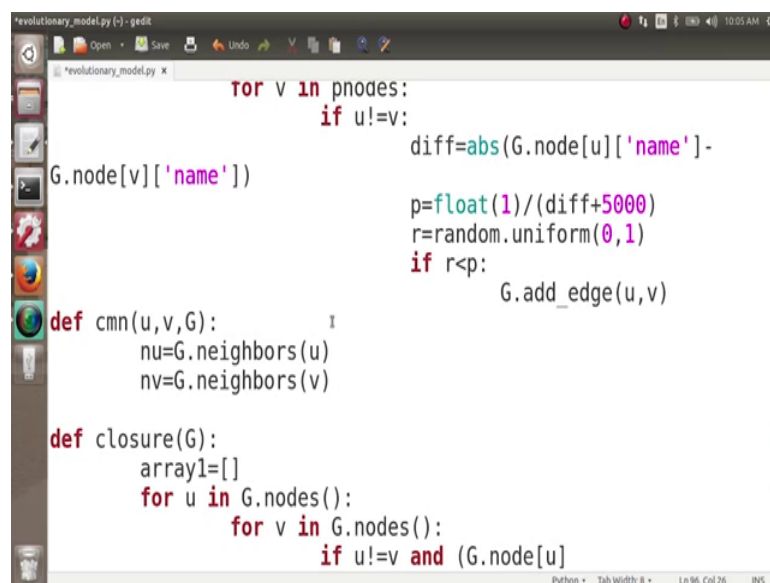
As we have seen that for any kind of closure to happen, the only condition required is whatever nodes we are picking both of though node should not be the foci nodes. Otherwise, if both of them are the people nodes; it results in a simple closure and otherwise the other 2 cases result in a membership closure and a foci closure; the only condition is both of the node should not be foci nodes.

Now, what do we do is for u in G dot nodes and then for v in G dot nodes; now both of these nodes should be the foci nodes at least one of these should be the person nodes. So, if first of all u is not equals to v and either G dot node; of u G dot node u; type equals to equals to person or G dot node; v, type equals to type equals to equals to person. So, one of they should be person; in that case we have to implement that closure. We already know; what was our formula for closure and we are using that same formula for all 3 kinds of closure.

So, here we find out the value of a probability p which is nothing, but our formula so our formula was 1 minus, 1 minus p to the power of k. So, for having 1 minus p to the power k; we have this function math dot power. So, we have to add this package math at the (Refer Time: 08:47) which will do, so 1 minus math dot power and power 1 minus p to the power of k, but now what is k here; k is the number of common neighbors . So, we need the number of common neighbors between u and v. So, we have here k equals to; find k equals to common neighbors.

So, let us have simple function in cmn common; common neighbors between u and v in the graph G. So, now we need to define this function here; define cmn common neighbors of u, v, G and how do we find its value is first of all we need the neighbors of u.

(Refer Slide Time: 09:57)
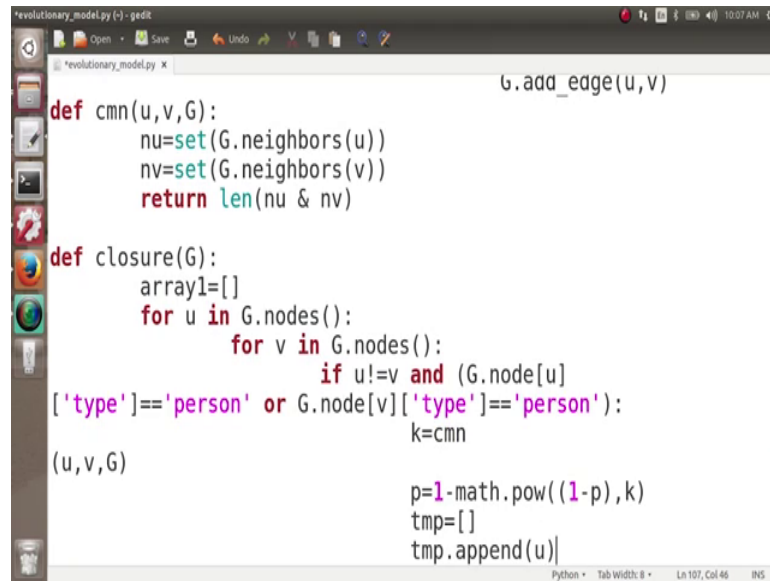
So, neighbors of u equals to G dot neighbors of u and then neighbors of v equals to G dot neighbors of v and we want the common one.
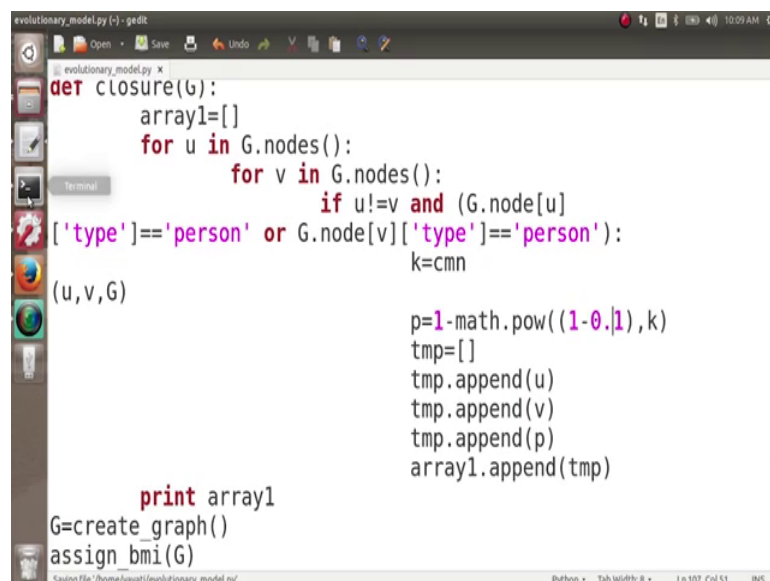
(Refer Slide Time: 10:28)

So, for having common one we need a set intersection, so let us make both of these value a sets. So, n u is the set having the neighbors of u and n, v is a set having the neighbors of v and what we have to return is the length of the set which set n, u and n, v which only looks at the common elements of u and v; to be written length of n u and n v. So, we have this probability here; math dot power 1 minus p to the power of k. Now, we have to feed these values inside our array which has to had 3 parts u, v and the probability of connection.

(Refer Slide Time: 11:33)
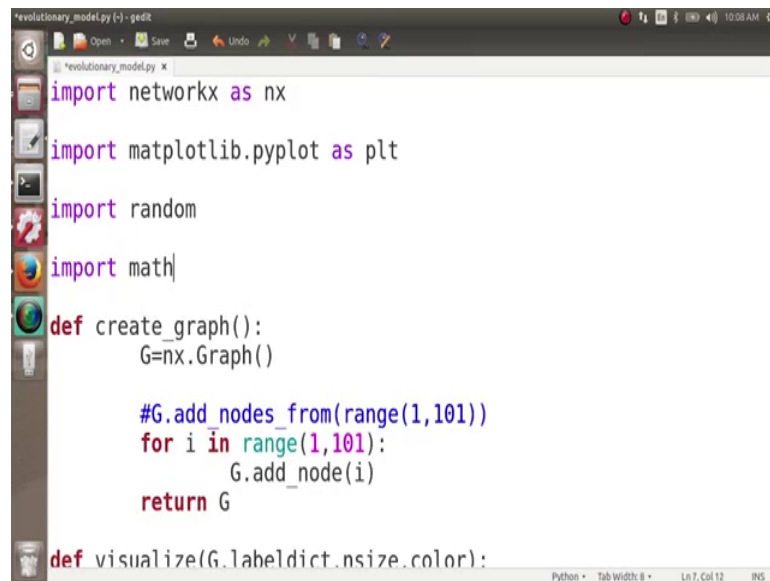
So, we need a sub array here let us name is strength, so the first value which we are going to append to temp phase u and the second value which we are going to append to temp is v and then the third value which we append to temp is p.

So, temp has these 3 values and at the end we append temp in array 1; array 1 dot append temp. So, we have made here this array 1; for all possible pairs of nodes, let us quickly look at how this array 1 looks like.

(Refer Slide Time: 12:14)
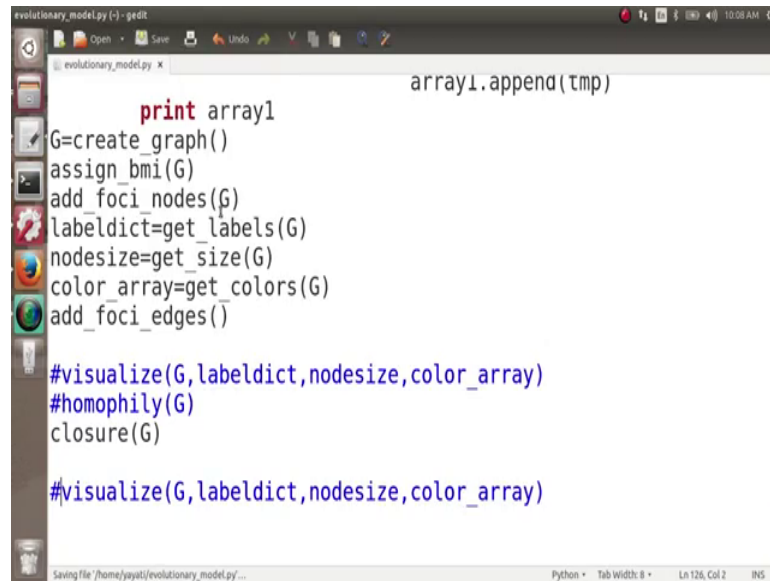
So, let us print array 1 and see how it looks like; now we forgot to import the package math, we go up and here we import math.
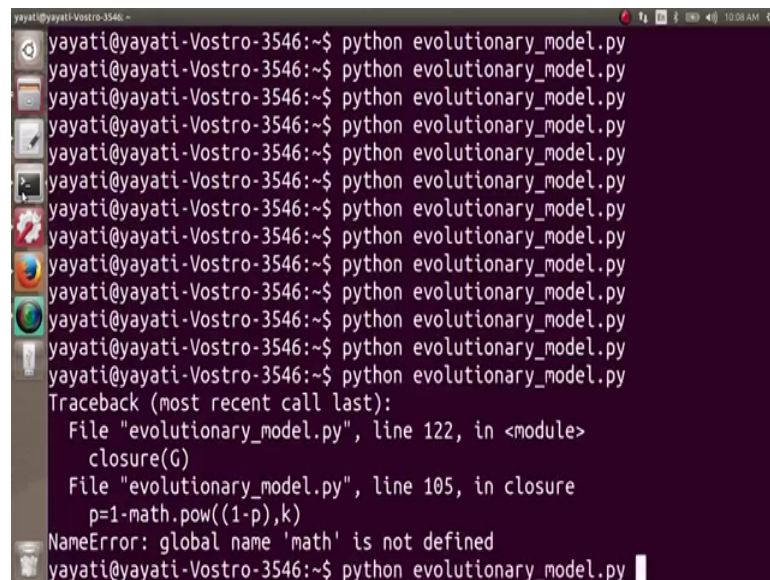
(Refer Slide Time: 12:35)

And at the end is this is become a (Refer Time: 12:37) code, so at the end for the time being we comment and visualize functions as well and now let us look at the array.

(Refer Slide Time: 12:42)

There is some problem, so this p here; p in the formula, this p here was supposed to be a constant. What was that constant value of probability? That is the probability each of your common neighbor contributes to u forming a connection. Let us come back, when we come here to closure p equals to this probability is supposed to be some value here.

So, let us say each of my neighbour contributes 0.01 probability of connection; let us rather say 0.1 probability of connection; then the more common neighbors we have, morer probability of connection becomes; let us go back execute it.

(Refer Slide Time: 13:52)

So, now you can see here your array and as we can see that as of now most of the values are here 0 and there is a reason for it to be 0 because there are very less common neighbors here.
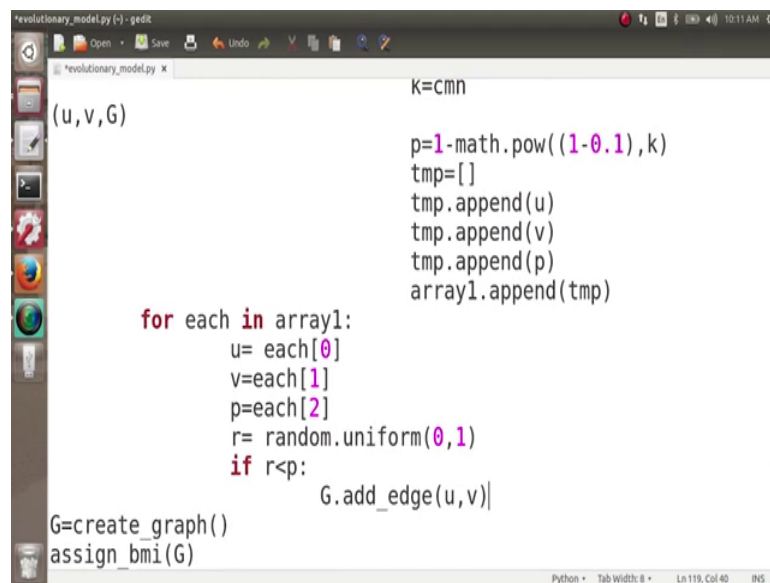
(Refer Slide Time: 14:10)

So, you can see here in some cases we have here this value here 0.09998, so since it is the very beginning of our codes. So, initially we had seen that the common neighbors only exist between, so there is no homophily as of now; we have commented the function homophily. The common neighbors will exist only when 2 people are the part of same social foci.

So, you can see here if you look at the nodes here 100, 87 and then 100, 74; 100, 56. So, these are mostly the people node, which are connected to the same social foci and this is the probability of these 2 nodes getting connected with each other. So, we apply a closure here, it is mostly since there is no edge between 2 people; it is mostly going to be happening a foci closure only. So, here was this array 1; now we have to make edges in accordance with this array 1.

(Refer Slide Time: 15:21)

So, that is very simple; so, we go for each in array 1; so, for every entry in array 1; what do we do; our first value, first node is each 0 and then our second node is each 1 and then our probability of connection is each 2. As we have done before, we choose our random real number from 0 to 1; r equals to random dot uniform 0 to 1 and if r is less then p, we add an edge G dot add edge; edge between u and v.

So, this is one iteration of our function closure let us execute and see it.

(Refer Slide Time: 16:10)

So, here we will visualizing our graph as well, so this is our initial graph looks like; let us see what is going on here. So, actually there are no intersections here it has just, ok.
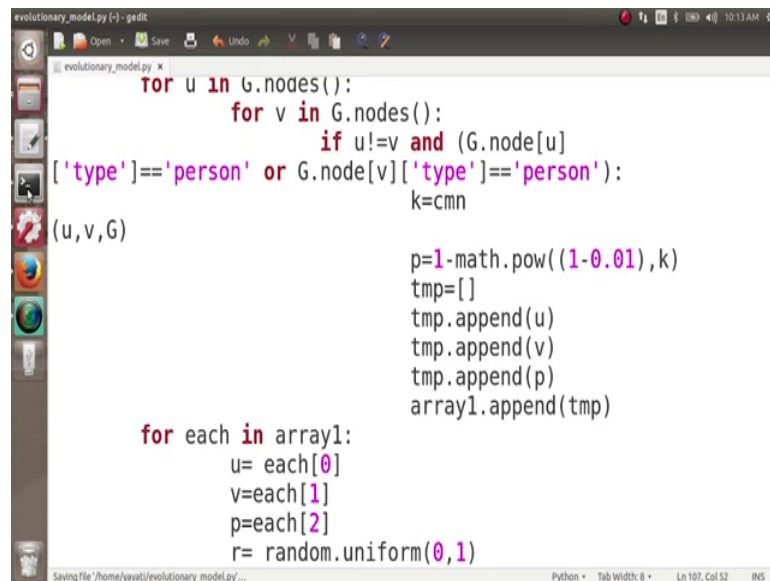
(Refer Slide Time: 16:13)
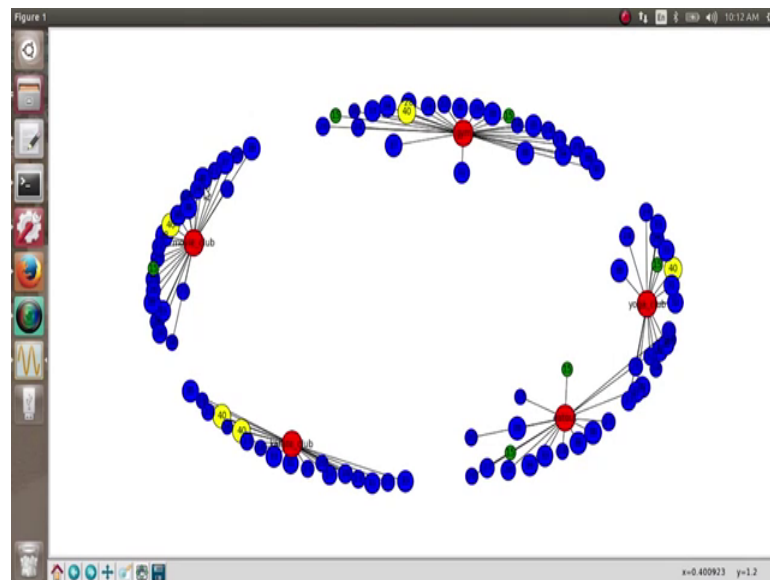
So, you can see here every node is actually the part of exactly one social foci here, we will zoom this graph and see and now this is what happens after closure. So, let us see what is happened after closure; let us look at this. So, you can see that there are more and more edges which have been added between people. So, 18 has now become connected

to 28; 28 has become connected to 24 because they were all the part of the same social foci gym.

So, you see how here the edges between people have been added, so as we can see here that probability of connection seems to be very high. We do not want this high probability of connection, so for that what we can actually do is; we go back and we deduce this probability here.
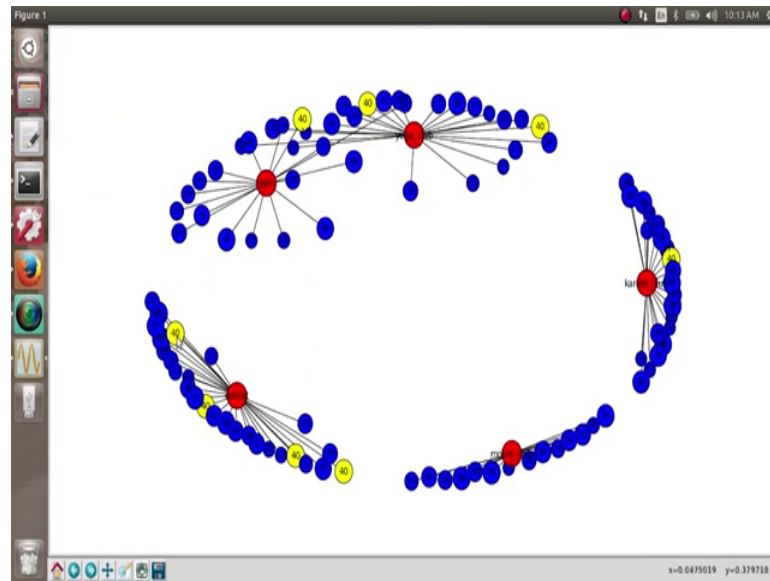
(Refer Slide Time: 17:22)

So, let us say one common neighbor contributes 0.01 probability of connection. Let us now look at, so this is our initial graph and let us see what happens here now. Let us again pick 1 here; let us say we pick this karate club and then you can, now here see that the probability of connection has reduced. So, this karate club leads to the formation of some number of edges between 2 people.

(Refer Slide Time: 17:31)

We might want to see other kinds of closures as well, so till now here only 2 people were getting connected to each other because of having a common social foci.

(Refer Slide Time: 18:05)

So, for that we would like to do homophily first so that we have some edges between the people as well and then we visualize and then we apply closer. So, for looking at, so let us for the meantime reduce this value and actually differentiating between the triadic closure and foci closure is a little bit difficult because there we edge edit between 2 people only. We would like to look at the membership closure, where one person

becomes a part of social foci because his friend is a part of social foci. So, let us try to look at that; we put this coding in a loop. So, while one closure happens and it keeps visualizing it again and again; now this is homophily, so, this is the graph. Now you can see here, this node 35; it is going to the karate club as well as it is going to the gym.
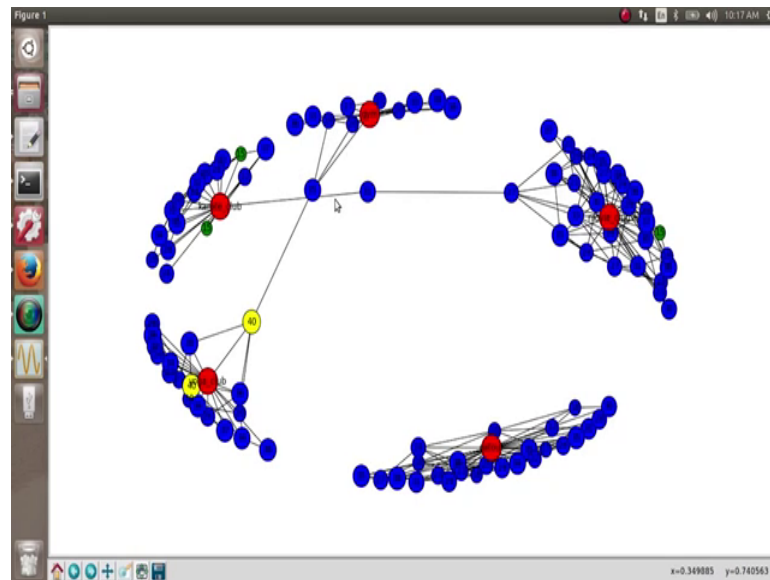
So, this node 35; it has 2 social foci is here in which it is participating, but according to our code initially node was assigned to exactly one social foci. So, most probably 35th node was initially a part of karate club only; as we see then it has identified that one of its friends 1920 or somebody was going to gym and hence 35 also decides to go and be a member of gym. So, we can actually see a membership closure also happening here; you can write this code, play around with it and look and verify it further and it will keep happening till I do not stop it.

So, let us just stop it here control z; I will stop this code here. So, till now what we have done is; we have made a graph having 100 nodes with different BMI's where every node is part of exactly one social foci. Then we implemented homophily here and then we implemented closures here and we implemented all 3 kinds of closures; triadic closure, foci closure and membership closure and we have also seen them happening on this graph.