project-2

Dadi Kumar

April 2024

1 Question 1

Choose the top leader by running a random walk on the graph with teleportation.

1.1 Answer 1

1.2 NODES

First we will read the given excel file.Now we need to find the nodes so we will extract all the elements in column 'email address'.created a list emails which stores all emails now we dont need full email we need only first 11 letters of email. for example let email be yyyybrnennu@iitrpr.ac.in (where yyyybrnennu represent entry number)but impressions have entry number and name so both nodes and elements of column have entry number as common so we need only first 11 strings of emial so we created a new list nodes to save nodes. and we are using .upper() becaus coloums have capital letters . now we create a Graph G with the elements of nodes list as nodes of this graph

1.3 EDGE LIST

So now create empty list edgelist and we create rowdata gives us the elements from column 2 to 32 in row i and this becomes edge list .But now we have have a problem some blocks are not filled but we want last 11 letters of each block with respect to row so we wrote if type(i)==str we add element to list k and we append k to edgelist. SO by this we get edge list.

1.4 CONSTRUCTION OF GRAPH

for i in range len(nodes) for neb in edgelist so we cant add edge between nodes[i] and neb .so we will add this edges to graph G.

1.5 PAGERANK

As sir discussed in week 6 videos we have an amazing algorithm to rank the nodes . I am using pagerank by random walk first i created a dictionary points with keys as nodes and values initializing to 0.First we slect a random node from graph .nodess. for i in range number of iterations points[s] increased by 1. we create a neighbours list which has all neighbours of s. r is a random decimal from 0 to 1 if r is greater than 0.15 then we enter this loop we enter this loop with probability 0.85 , because if 2 nodes are having out going edges only among themselves and not connected to any other but they have incoming edges in this case if s becomes one of 2 nodes then the points will be shared only between them so we may get wong answer to remove this we are entering the loop with 0.85 probability.

In this loop we if neighbours are there then we slect a random neighbour and let it be e , else we select a random node from nodes and let it be s. If e is in nodes s=e, else points[e]=0,append e to nodes list and append empty list to edgelist. if r is less than 0.15 then we again select s as random node from the nodes . Total visits equal to sum of points of all nodes. Now pagerank value is a dictionary with keys as nodes and values as (score/total visits) for node,score in points.items() and return pagerank values.

LEDER: Now we run for number of iteration equal to 1 lakh. Now we should sort as per pagerank values. On sorting we get the rank of nodes . Now leader means the Person with rank 1. In our case the person comes out to be the person with entry number 2023CSB1091. He is aadit mahagen so he is the leader.

2 Question2

Recommend missing links using the matrix method explained in the class. I am trying to do this as explained in class so i am taking the adjeceny matrix which have 0s and 1s . And now our question is to predict the missing edgs that is number of ones if all people met all people and collected impressions.

2.1 Code

I took the same graph which i used in question 1. Now to find adjacency matrix we can do it manually by if nodes are connected 1 else 0 but pyton has inbuilt function so i used the inbuilt function. n is equal to total number of nodes . So i done it as explained in class so our task is to select a element of matrix which is 0 and try to guess its value i guessed it using the technique by assuming all the rows are linearly dependent and finding how they are dependent . SO our task is to remove the row and column corresponding to the element whose value need to be calculated. So we will get a (n-1 x n-1) matrix . Let it be A.We also want elements of its row and column except those element .Let the row matrix be B.Let the column matrix be C. We have inbuilt function to remove rows and colums as discussed in class by raj sir . So if we have the A and B so we can find how the find how the n-1 elements of each column in A related to B. By assuming our element that we need to find will also have same relation with its n-1 Column elements we will try to get its value . let the value of element be value. So to find how A and B are related we can do $A^{-1} \times B$. As we know that $A \times X = B$. where X is the matrix which shows the linear relation . Now we we calculate the value of element using value $= X \times C$. But we always cant do this because A might not be invertable. So python has an inbuilt function by using that i calculated these . I created two empty lists newly connected nodes and newly disconnected to store newly connected and not connected respectively.Now for i in range(n) and for j in range(n) and if matrix[i][j] is not zero then i find the values of those elements i find the value of element using the above discussion . And if value is greater than 1 then i change the value of element in adjacency matrix to 1 and also append i,j in newly connected else append them in newly not connected. Finally i print newly connected . this is also missing edges asked in question And i also printed number of missing edges the list newly connected contains all the missing links corresponding to the i,j th index in the original adjacency matrix . Finally i got total number of missing edges are 5510.

3 Question3

: Sir told us to choose a creative problem . So i though to observe the number of people impressed by a particular node and stored the data now as statistically our data should follow bell curve . So by using a python code i checked how close our data to ideal behaviour . And also print the nodes which are not following bell curve .

3.1 Answer 3

:

3.2 Storing scores and respective nodes

We created a graph same as question 1 then we removed the nodes which filled by other people as impressed nodes but they did not gave any impression data. Because they are filled by very less people and because of them the nodes with less people impressed will be more and we will be heavily diveated from ideal bell graph.Now i created a list d which stores all the impression of every nodes . I created this beacause to find score(Here score means the number of nodes impressed by given node) of a node i can see how many times that node is present in d that is the number of people impressed by this node. Now i created an empty lists points . points list stores node and respective score (Here score means the number of nodes impressed by given node) . for i in range of length of b(b is a list containing nodes). Initialize score to 0 ,for j in range of length d . Now to calculate score we need to calculate how many times that node is in d. SO if b[i] equal to d[j] then i increase score by 1 . By this i get score of each node. now we create new list presons to store the persons in 0 to 5 percentage as first element and 5 to 10 as 2 nd and so on.

3.3 bell curve

A bell curve is a common type of distribution for a variable, also known as the normal distribution. The term ”bell curve” originates from the fact that the graph used to depict a normal distribution consists of a symmetrical bell-shaped curve.

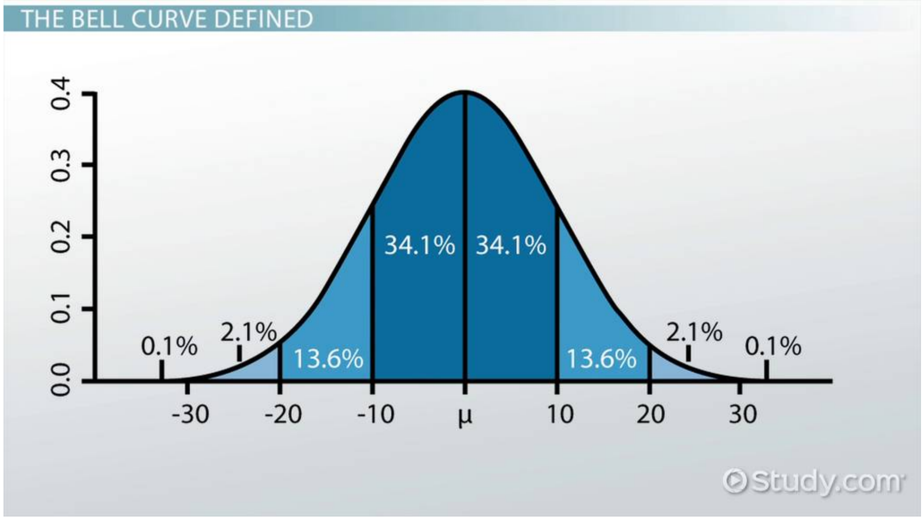

Figure 1: Theorotical bell curve

3.4 Comparing our data to ideal bell curve

Now check whether our data is close to ideal or not we will divide our poits into 20 partitions (5 percent each). That is i want to store how many nodes in 0 to 5 percent of maximum , 5 to 10 percent maximum ,,, so on , 95 to 100 percent of maximum .Now we will sort our list e and list points as per score . And note the maximum value of score . To find frequency of nodes in 0 to 5 percent of maximum , 5 to 10 percent maximum ,,, so on , 95 to 100 percent of maximum . Let p=0.05 and i created a new list final which all the lists of nodes lying in 0 to 5 percent of maximum , 5 to 10 percent maximum ,,, so on , 95 to 100 percent of maximum in another list. for i in range 1 to 21 because we need 20 partitions we create a empty list f for j in list e if the node lies in respective percentage range we append it to list f . And we append the list f to list final . Now we have the nodes in thier respective percentage distribution but we just need number of nodes . so i created a empty list freq to store length of each elements of list final . By using matplotlib plt.bar(x,freq). and label corresponding percentages in x axis and label number of people having the score lying in this percent in y axis . At last plt.show() shows us the bar graph.

3.5 Result

By observing our data bar graph it has some deviations from statistically ideal bar graph. We can observe deviation at starting , eding and also in the region of precentage corresponding to 50 to 55 percentage You can observe the figure below . so we can directly get this node by calling 11 th element of persons that is persons[10].'2022CSB1106', '2023MCB1296', '2023CSB1130', '2023CSB1110',

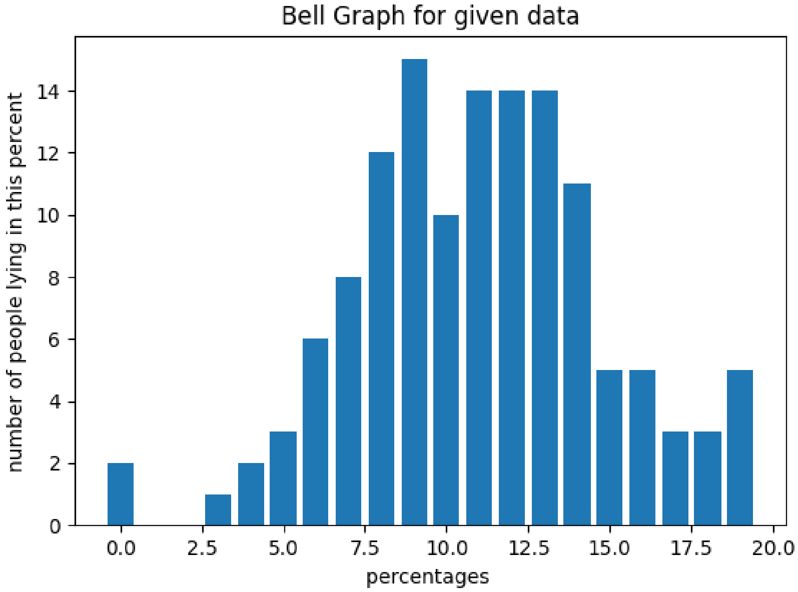

Figure 2: distribution for our data

’2023CSB1136’, ’2023CSB1106’, ’2023CSB1114’, ’2023MCB1304’, ’2020MCB1225’, ’2023CSB1141’ these nodes distribution is deviated from ideal bell curve .