Analysis of Human Disease Network

Neetha Reddy
IMT2018050

Dataset Overview:

Online Mendelian Inheritance in Man (OMIM)[1] publishes a linked data version of Diseasome, a network of 4,300 disorders and disease genes linked by known disorder-gene associations for exploring all known phenotype and disease gene associations, indicating the common genetic origin of many diseases. The list of disorders, disease genes, and associations between them was obtained from OMIM.

Goh et al. (2017)[2] created a bipartite graph consisting of two disjoint sets of nodes. One set corresponds to all known genetic disorders, whereas the other set corresponds to all known disease genes in the human genome. A disorder and a gene are then connected by a link if mutations in that gene are implicated in that disorder. The dataset contains 516 nodes and 1188 edges where each node represents a disorder gene and the presence of an edge between 2 nodes indicates existence of a disorder-gene association.

Background:

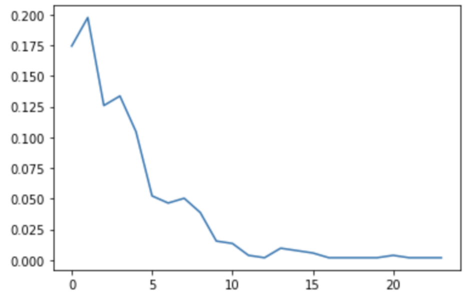
- ➔ **Centrality**
  - ◆ Any natural setting is an interconnected web of several complex networks. Often to understand the nature of these networks, centrality measures are used as a tool to understand a node's prominence in the network. The intuition is that in most networks some vertices or edges are more central than others. Based on the context, the prominence could be a measure of structural power, status, prestige, or visibility and in this case mutation.
  - ◆ There are several ways to determine the centrality of a unit in a network depending on the aspects of the problem in question. The prominent centrality methods that I would be looking in this particular article are:
    - i. Degree Centrality
    - ii. Closeness Centrality
    - iii. Betweenness Centrality
    - iv. Eigen-Vector Centrality
  - ◆ We will now take the above dataset as an example to explore the aforementioned concepts of centrality to show what each centrality index addresses in this context.
- ➔ **Tools used**
  - ◆ Python3
  - ◆ *Libraries:* Plotly, Networkx, Matplotlib, Numpy.

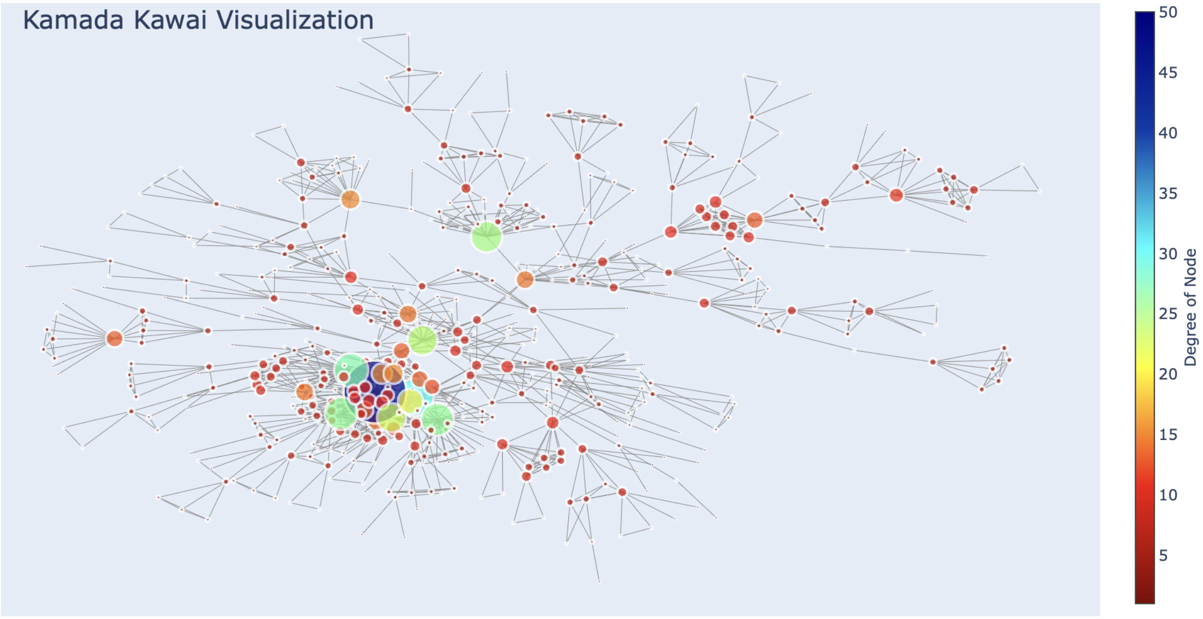
[1] "diseasome | Biological Networks | Network Data Repository." https://networkrepository.com/bio-diseasome.php. Accessed 10 Dec. 2021.
[2] "The human disease network | PNAS." 22 May. 2007, https://www.pnas.org/content/104/21/8685. Accessed 10 Dec. 2021.

Visualisations[3,4]:

Degree distribution of the graph

```
[<matplotlib.lines.Line2D at 0x7f592cbbcc50>]
```





- On plotting the histogram of the degree distribution of the graph, I observed that most of the nodes in the given dataset have their connectivities in the range [1,7] on an average. The minimum degree and maximum degree of the graph are 1 and 50 respectively.
- I have used the Kamada Kawai layout to visualize all the subsequent networks since it has the least number of edge crossovers and gives us a clearer picture of the entire network. Further, I used the jet color map to represent the connectivity of each node in all the subsequent network visualisations.

Degree centrality

- Formally, degree centrality of each node is the proportion of its degree to the total degree in the graph.

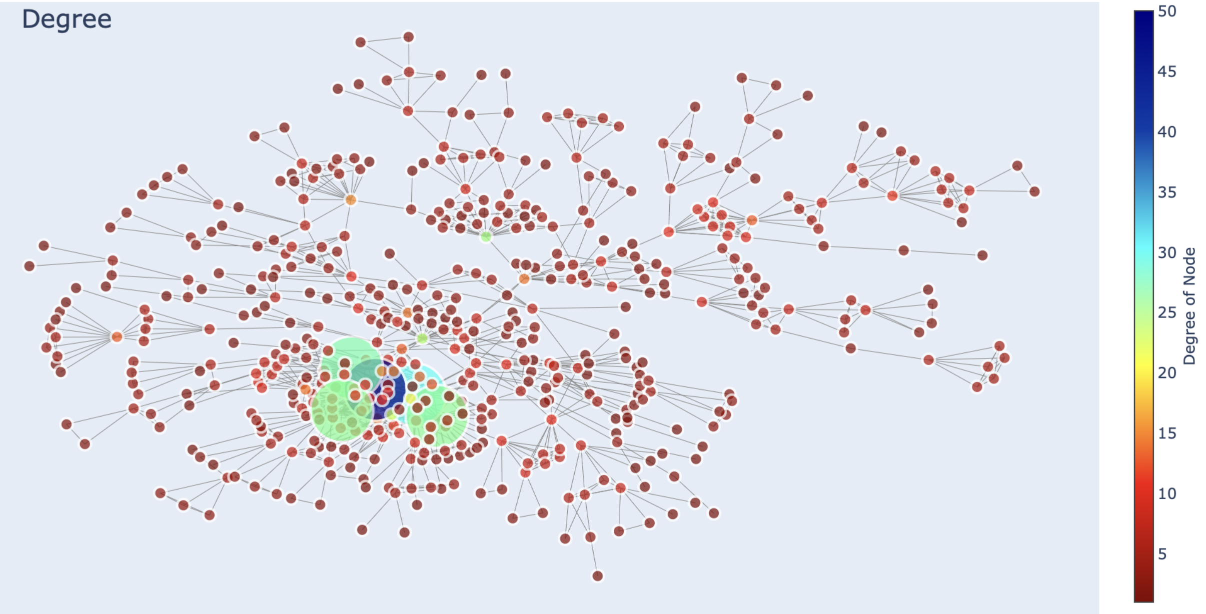
[3] "Network Graphs in Python - Plotly." https://plotly.com/python/network-graphs/. Accessed 10 Dec. 2021.

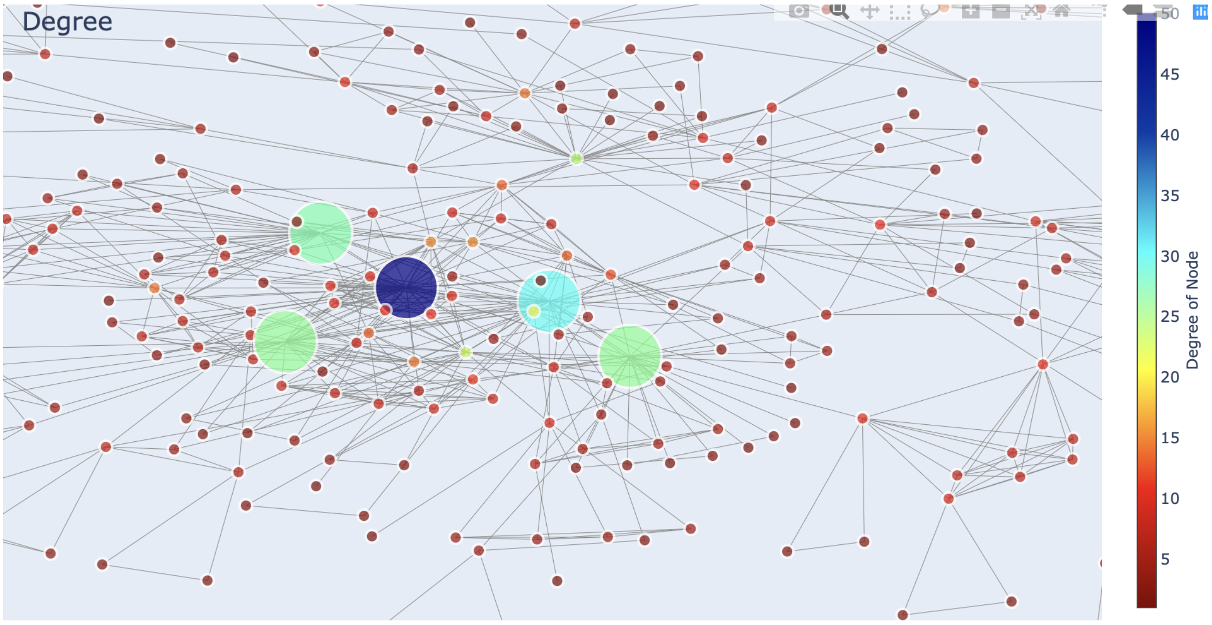
[4] "Centrality — NetworkX 2.6.2 documentation." https://networkx.org/documentation/stable/reference/algorithms/centrality.html. Accessed 10 Dec. 2021.

$$c_D(v) = \frac{d(v)}{\Sigma_{\forall v \epsilon V} d(v)}$$

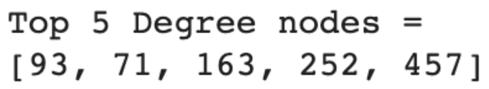
Here $d(v)$is the degree of node v.

- It is a measure of the direct influence that a node has on other nodes in the graph. Since this dataset deals with undirected graph, we are not looking at in-degree and out-degree metrics distinctively.
- While degree centrality is widely used, considering only this measure to form inferences about the network might give us unexpected results. This is especially the case when the highest degree nodes are located in the periphery of the graph.





I have amplified the sizes of five nodes with the highest degree centralities by a factor of 5 as shown in the visualisation. These five are also the nodes with the highest degrees in the graph.



```
Top 5 Degree nodes =
[93, 71, 163, 252, 457]
```

Betweenness centrality

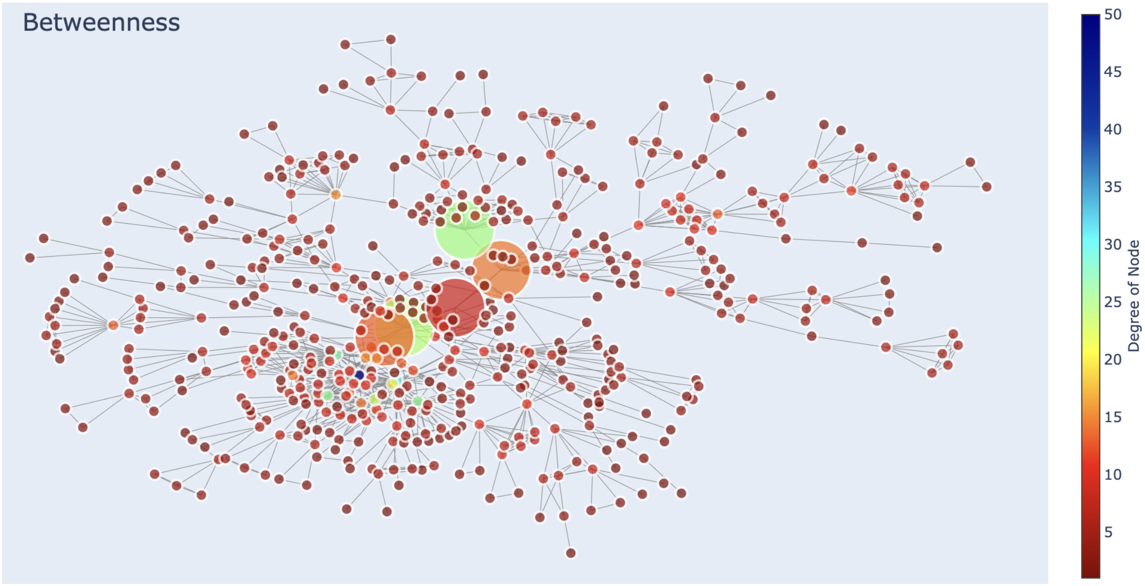
- Formally, betweenness centrality of each node is the proportion of the number of shortest paths that pass through that node to the total number of shortest paths in the graph.

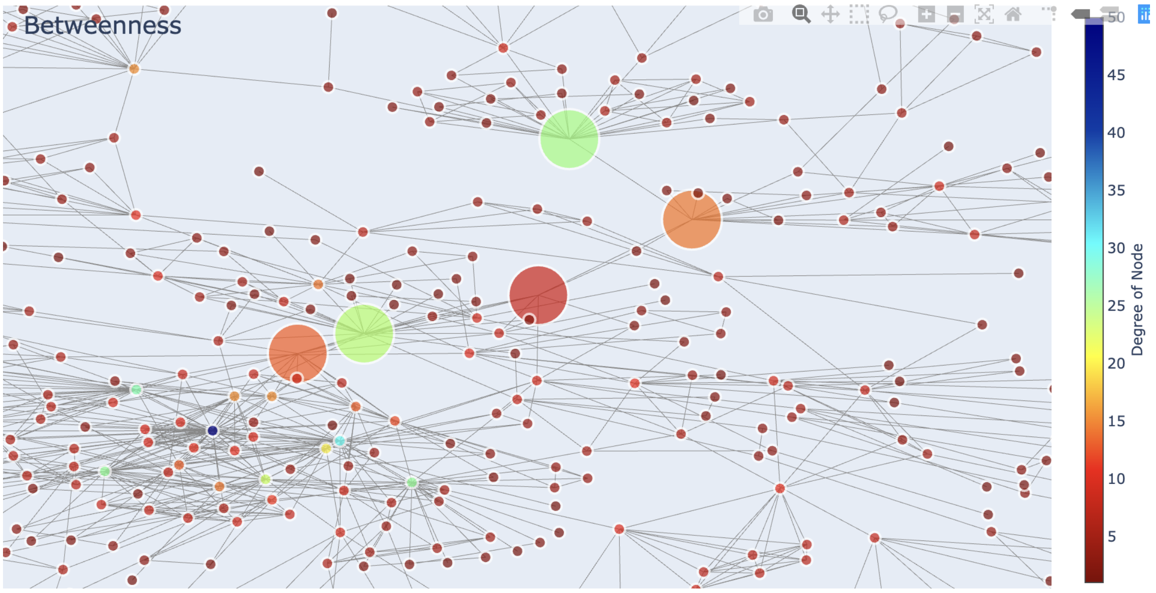
$$c_B(v) = \sum_{s\neq v}\sum_{t\neq v}\frac{\sigma_{st}(v)}{\sigma_{st}}$$

Here $\sigma_{st}(v)$ is the number of shortest paths from node s to node t which pass through node v

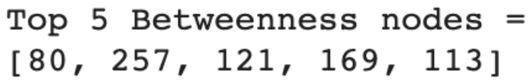
σ_{st} is the number of shortest paths from node s to node t.

- This measure calculates a node's relative ability to facilitate paths (act as "bridges") between other nodes. It can be used to identify nodes in the graph whose failure might cause congestion issues.





I have amplified the sizes of five nodes with the highest betweenness centralities by a factor of 5 as shown in the visualisation. This visualization shows that the nodes with highest betweenness centralities are nodes which connect large clusters by acting as "bridges" and these nodes can even have minimal degrees as shown by the red node in the above graph.



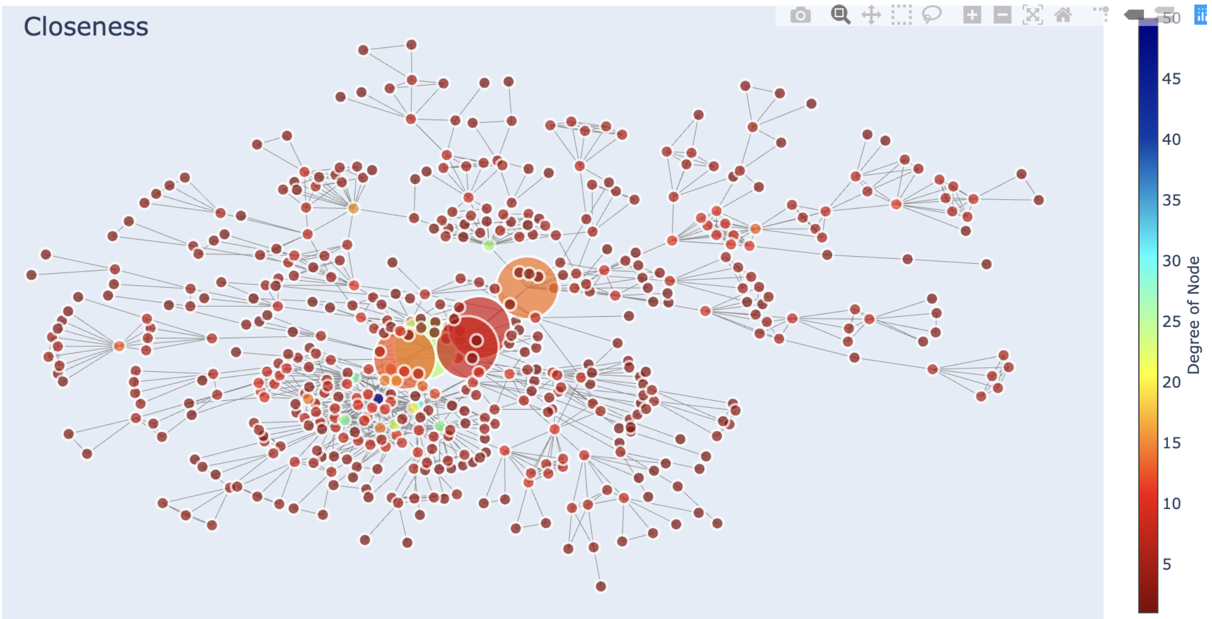
```
Top 5 Betweenness nodes =
[80, 257, 121, 169, 113]
```

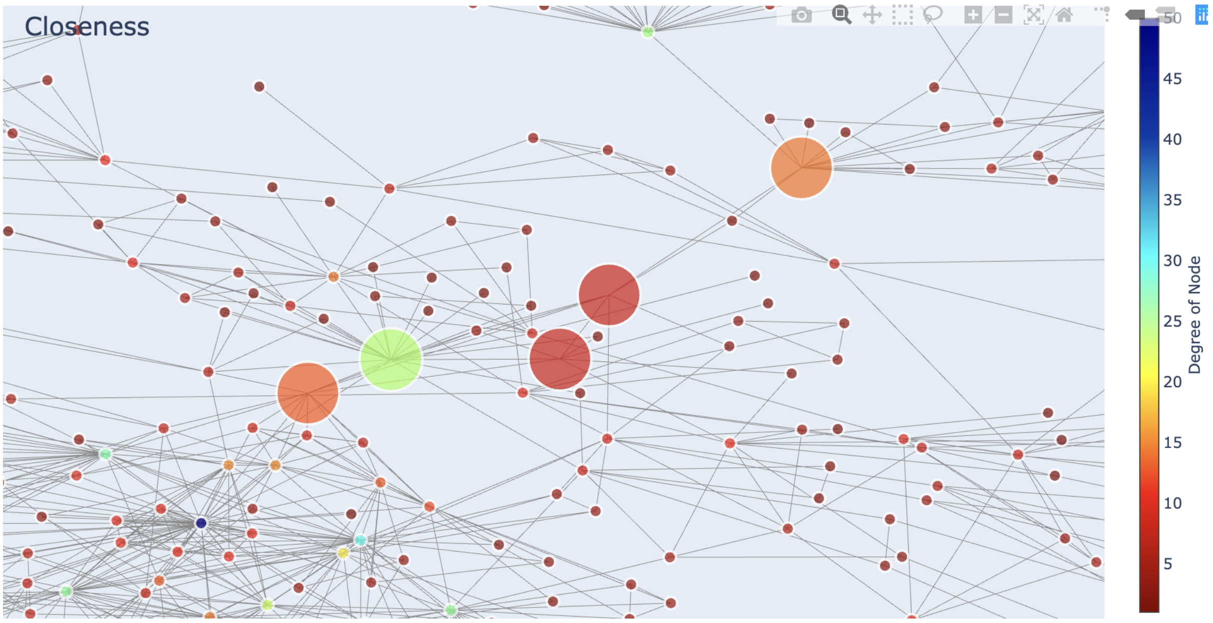
Closeness centrality

- Closeness of a node is defined as the summation of shortest path distance from this node to every other node in the graph.

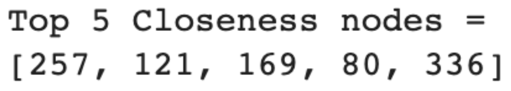
$$c_C(v) = \Sigma_{u \epsilon V} d(u, v)$$

  Here, $d(u, v)$ denotes the distance between node u and node v.
- This measure determines a node's relative ability to rapidly influence other nodes. It can be used to identify nodes in the graph which can facilitate fast transfer of information.





I have amplified the sizes of five nodes with the highest closeness centralities by a factor of 5 as shown in the visualisation. This visualization shows that these nodes can even have minimal degrees as shown by the red nodes in the above graph.

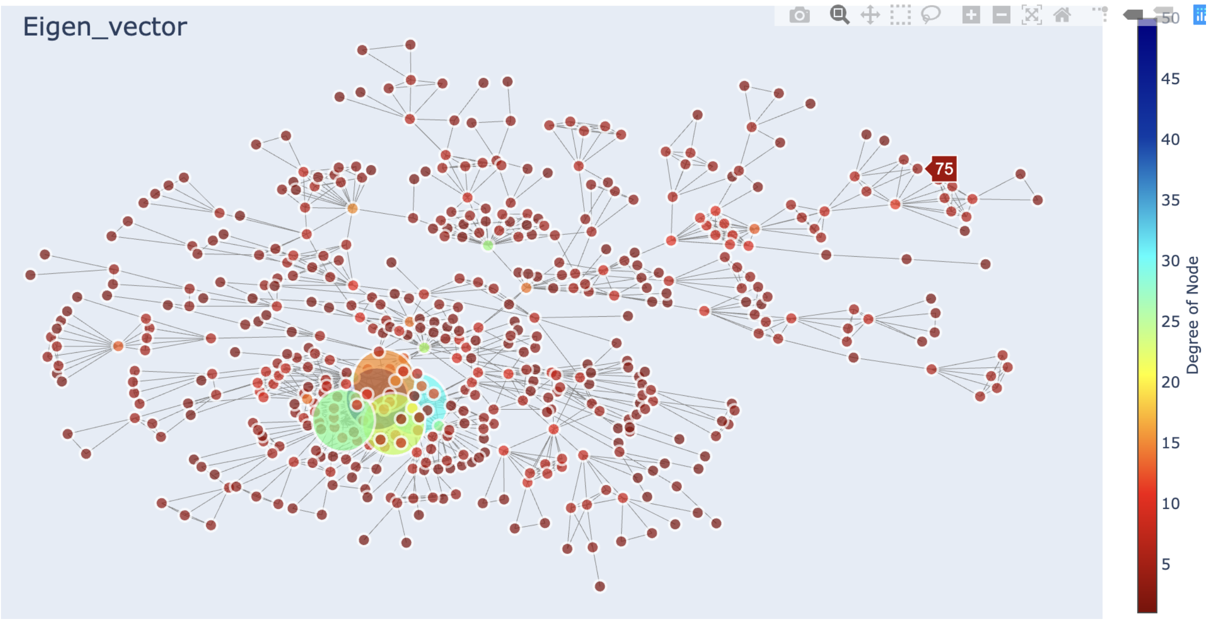


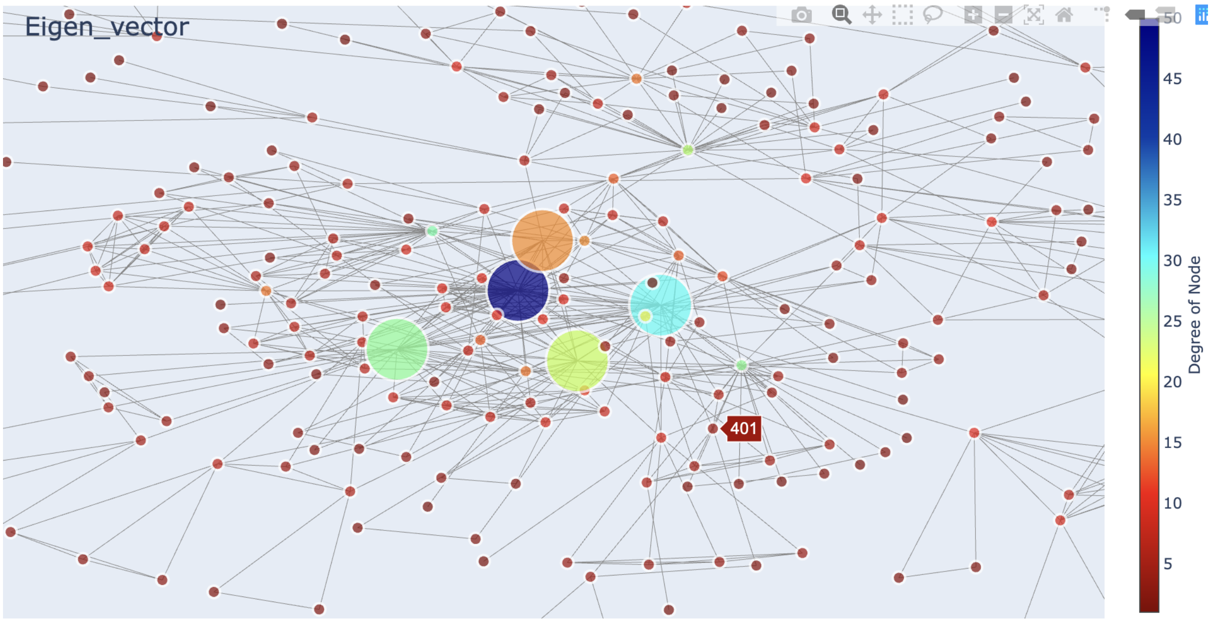
```
Top 5 Closeness nodes =
[257, 121, 169, 80, 336]
```

Eigen centrality

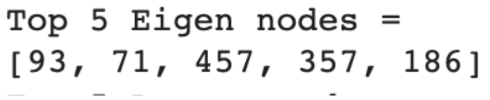
- Eigen centrality determines a node's importance keeping in mind the connectivity of its neighbouring nodes. It is computed using matrix calculation to determine the principal eigenvector using the adjacency matrix.
- Apart from being used to determine influential nodes in a network, a variant of this is used in Google's PageRank algorithm, which is used to rank web pages[5].

[5] "Analyzing the Social Web || Network Structure and Measures ...." https://ur.booksc.eu/book/41073072/21090a. Accessed 10 Dec. 2021.





I have amplified the sizes of five nodes with the highest eigen centralities by a factor of 5 as shown in the visualisation. This visualization shows that the most influential nodes need not be the most locally connected nodes as shown in the degree centrality measure.



```
Top 5 Eigen nodes =
[93, 71, 457, 357, 186]
```

Conclusion:

I can conclude that eigen centrality and closeness centrality measures can be used alternatingly for this dataset. This is because the dataset works with disease data attempting to trace the common ancestor gene of different disorders and thereby helps us in identifying causes and symptoms of the disease. These inferences may be vital while developing cures by designing medicines such that it reaches the affected part without causing damage to other organs. Other centrality measures can also be used based on the usecases.