P5 - Graph Data Mining

Vivek Reddy Karri (ID: 200315262)

Virus Propagation in Time evolving graphs

November 16, 2019

Q1) For the SIS (susceptible, infected, susceptible) Virus Propagation Model (VPM), calculate the largest eigenvalue of the system-matrix and answer the folowing questions?

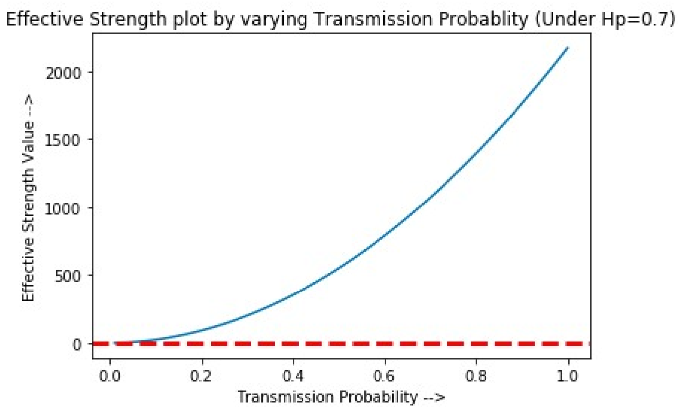
1a) Will the infection spread across the network (i.e., result on an epidemic), or will it die quickly?

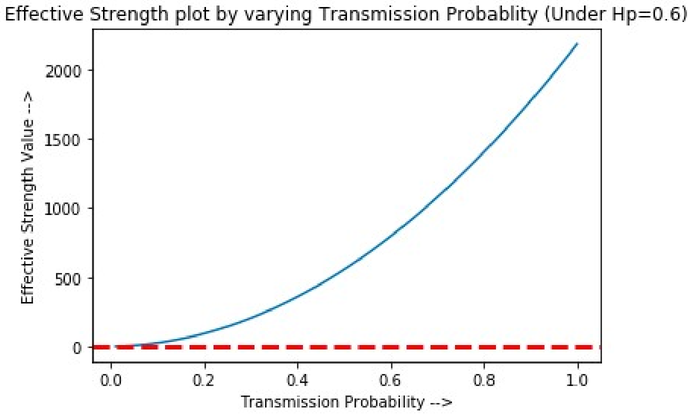
Effective Strength Results:

```
Note: Tp => Transmission Probability, Hp => Healing Probability

Using β1 : 0.20,  β2: 0.01
Largest Eigen Value: 91.46220624150669

Using δ1 : 0.70, δ2 : 0.60
Largest Eigen Value: 0.6193063043519553
```

According, to these results the virus will die out for the SIS Virus Propagation Model with δ_1: 0.01 and δ_2: 0.60. However, the virus will result in an epidemic for the β_1: 0.20 and β_2: 0.70.

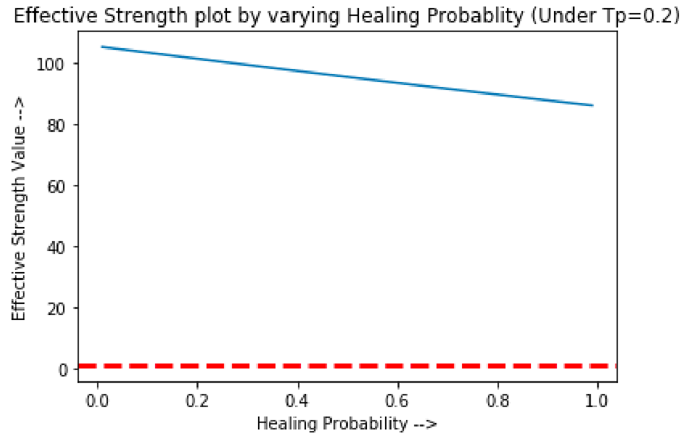
1b) Keeping δ fixed, analyze how the value of β affects the effective strength of the virus What is the minimum transmission probability (β) that results in a network wide epidemic?

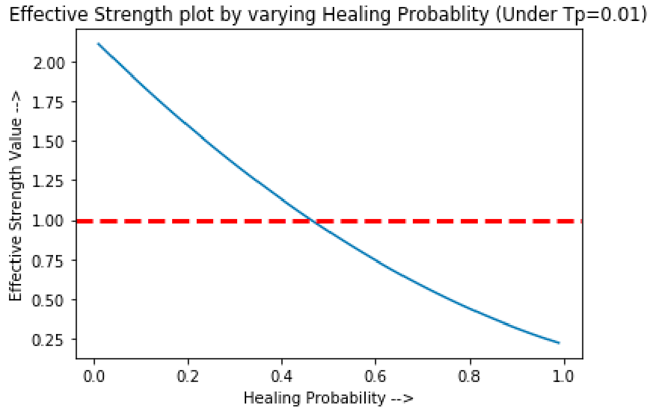




Minimum Transmission Probability found out for an Epidemic in both the cases are 0.01.

1c) Keeping β fixed, analyze how the value of δ affects the effective strength of the virus What is the maximum healing probability (δ) that results in a networkwide epidemic?



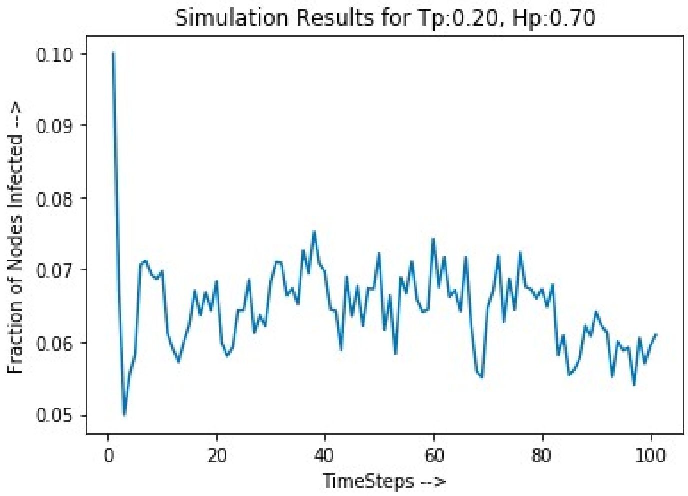


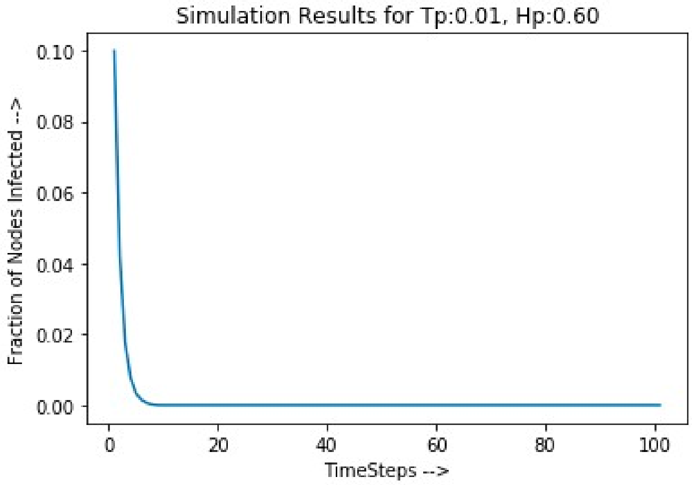
Maximum Healing Probability which would result in a network wide epidemic is found out to be 0.48 when the Transmission Probability is 0.01, In the other case, no healing probability can stop an outbreak.

Q1d) Repeat (1), (1a), (1b) and (1c) with β = β2 , and δ = δ2 .

The results for β2 and δ2 are given in the previous question's itself.

Q2) For the modified virus propagation simulation program, and the set of 2 alternating contact networks provided answer these questions? Plot the average fraction of infected nodes at each time step. Did the infection spread across the network, or did it die quickly? Do the results of the simulation agree with your conclusions in (1a)?





For this choice of probabilities, Tp:0.20, Hp: 0.70:

The virus did not die as predicted in the previous question.

For the choice of probabilities, Tp:0.01, Hp:0.60

The virus died very quickly .

The results of the simulation does comply with the conclusion made in the previous question.

Q3) Write a program that implements an immunization policy to prevent the virus from spreading across the network. Given a number of available vaccines (k) and a contact network, your program should select k nodes to immunize. The immunized nodes (and their incident edges) are then removed from the contact network. Answer the following Questions on the below given Heuristics?

Policy A: Select k random nodes for immunization.
Policy B: Select the k nodes with highest degree for immunization.
Policy C: Select the node with the highest degree for immunization. Remove this node (and its incident edges) from the contact network. Repeat until all vaccines are administered.
Policy D: Find the eigenvector corresponding to the largest eigenvalue of the contact network's adjacency matrix. Find the k largest (absolute) values in the eigenvector. Select the k nodes at the corresponding positions in the eigenvector.

Q3a) What do you think would be the optimal immunization policy? What would be its time complexity? Would it be reasonable to implement this policy? Justify.

Based on the observations from the questions Q3b, Q3c, Q3d, Q3e, I would think that Policy C is an optimal immunization policy over other 3. Although the complexity O(k*E), looks higher than the other approaches, It can be reduced by storing the values computed and not computing degrees every time. By this implementation this would be reduced to $O(E+K*\Sigma_{i=1:k} deg(i))$, which is again equal to O(E). with a bit of sophistication, it would be reasonable to implement this policy.

Q3b) What do you think is the intuition behind this heuristic?

Policy A: This assumes a constant probability over the chances of virus proagation over every node. Which isn't true always.
Poicy B: This assigns higher probability for the nodes which has more connections in the graph.
Policy C: This is a slight improvement over B, because the node with highest degree might change after removing the previously highest degree node and in a way bit more intuitve than Policy B.
Policy D: This Approach ranks selection of nodes in a different way than selecting highest degree nodes which is by looking at the largest eigen vector and by removing k-nodes corresponding to the k-largest values in the vector.

Q3c) Write a pseudocode for this heuristic immunization policy. What is its time complexity?

Policy A:

Pseudo Code:
Step 0) There a n Infected nodes in the graph.
Step 1) Randomly Select k nodes. Since the selection is random, it doesnt matter on which alternating point of graph, you are randomly sampling from.
Step 2) Remove Edges corresponding to those nodes and remove nodes (if any) which are also present in n-infected nodes.

Time Complexity:
1) Selecting k-nodes O(k).
2) Removing edges $O(K*\Sigma_{i=1:k} deg(i))$
3) So Overall Time complexity is $O(K*\Sigma_{i=1:k} deg(i))$.

Policy B:

Pseudo Code:
Step 0) There a n Infected nodes in the graph.
Step 1) Remove n-highest degree nodes from the graph and immuniza them
Step 2) Remove their corresponding edges and continue their simulation

Time Complexity:
1) Find k-highest degree nodes O(E+V).
2) Removing edges $O(K*\Sigma_{i=1:k} \deg(i))$

3) So Overall Time complexity is O(E) assuming k<<V.

Policy C:

Pseudo Code:
Step 0) There a n Infected nodes in the graph.
Step 1) Select the highest degree node for immunization.
Step 2) Remove this node from the contact network (Remove the edges)
Step 3) Repeat the above steps for k step.

Time Complexity:
1) Selecting highest Degree node (O(E+V)) and repeating this for k steps gives O(kE+kV)
2) Removing edges $O(K*\Sigma_{i=1:k} \deg(i))$
3) So Overall Time complexity is O(kE).

Policy D:

Pseudo Code:
Step 0) There a n Infected nodes in the graph.
Step 1) Find the eigen vector corresponding to the largest eigen value of the System-Matrix.
Step 2) Remove k largest values in the eigen vector and the corresponding nodes from the graph

Time Complexity:
1) Selecting k-nodes O(k).
2) Time Complexity to find largest eigen values is $O(V^2)$
3) Time complexity to remove k-largest values by looking at the eigen vector is O(N), assuming k<<N.
4) Removing edges from the graph would result in $O(K*\Sigma_{i=1:k} \deg(i))$
5) So overall complexity would be $O(V^2)$, assuming k<<V.

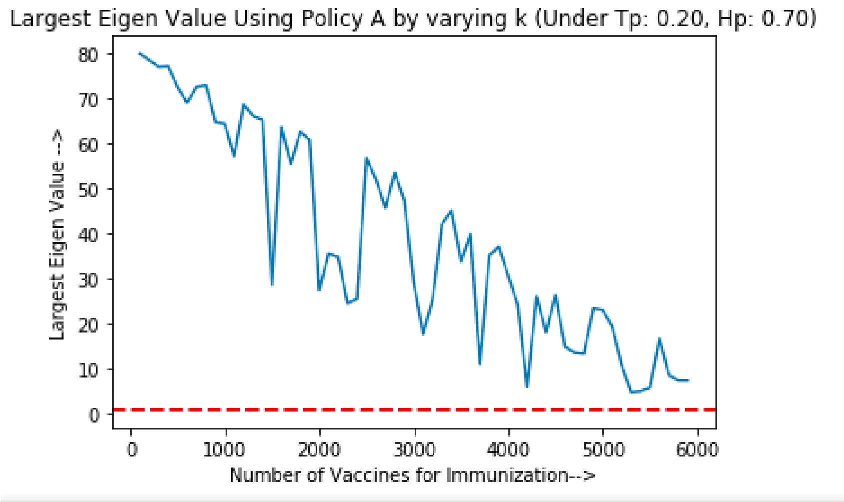
Q3d) Given k = k1, β = β1 , and δ = δ1, calculate the effective strength (s) of the virus on the immunized contact network (i.e., contact network without immunized nodes). Did the immunization policy prevented a network wide epidemic?

```
Num Infected Nodes after simulation was found to be: 370
..............................Policy A...............................
Before Immunization: 370
After Immunization: 356
Largest Eigen Value was found out to be : 77.678495
Note: As expected, this did not prevent the epidemic.
..............................Policy B...............................
```

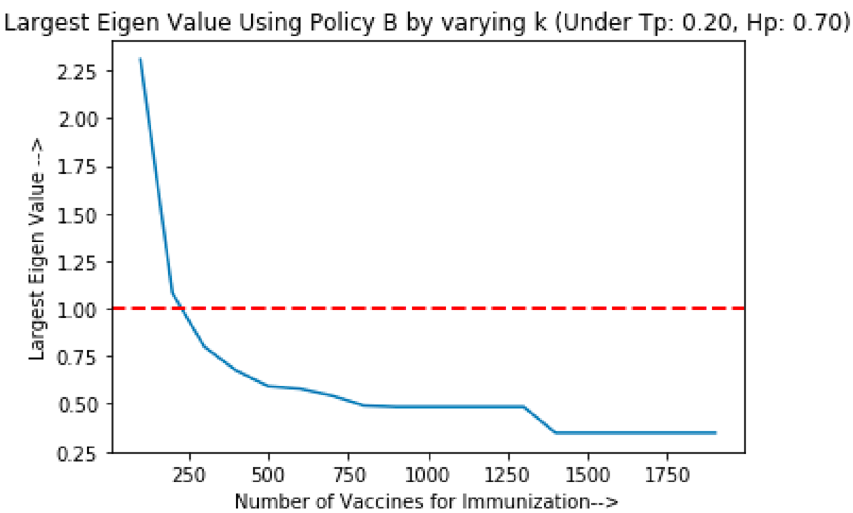
```
Before Immunization: 370
After Immunization: 330
Largest Eigen-Value was found out to be : 1.08098485
Note: As the values is close to one, it can be safely assumed that this policy can stop the
outbreak.
..............................Policy C..............................
Before Immunization: 370
After Immunization: 334
Largest Eigen-Value was found out to be : 1.04681282
Note: As the values is close to one, it can be safely assumed that this policy can stop the
outbreak. Also this works better than Policy B.
..............................Policy D..............................
Before Immunization: 370
After Immunization: 353self.zombies = self.zombies.difference(set(immunized_nodes))
Largest Eigen Value was found out to be : 26.8368125
Note: Although, this may not be as bad as the Policy A, this still is not a good fit as an
immunization policy.

........................................... END .................................................
```

Q3e) Keeping β and δ fixed, analyze how the value of k affects the effective strength of the virus on the immunized contact network (suggestion: plot your results). Estimate the minimum number of vaccines necessary to prevent a network-wide epidemic.

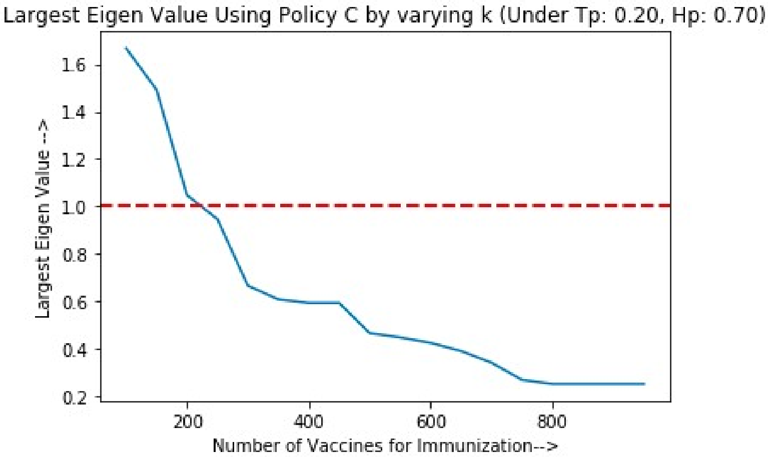
Policy A



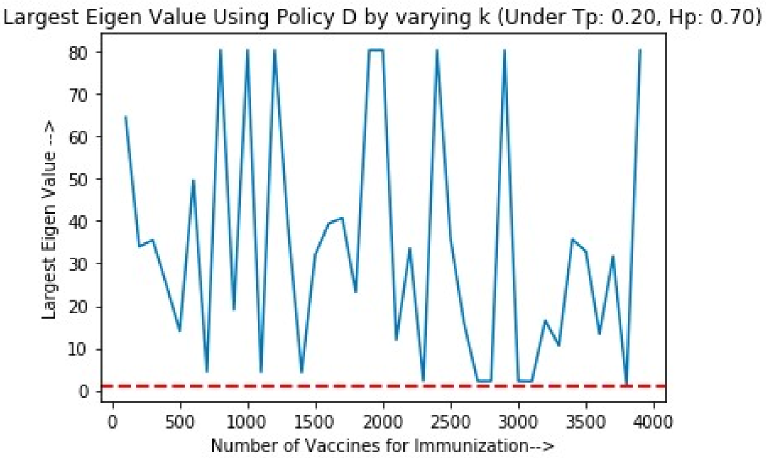
Policy B



Policy C



Policy D



Minimum Vaccines to stop an outbreak: (For Tp: 0.20, Hp: 0.70)
Policy A: K ~ V
Policy B: K ~ 200
Policy C: K ~ 200 and also less than Policy B
Policy D: This is unstable as suggested by the plot and a safe guess for the number of vaccines to stop an outbreak is K ~ V.