SECTION-1

Name: Sagar Narla

Worked alone.

Code submitted and also can be found at:
https://github.com/sagarnarla/GDMProject4

SECTION-2

OPTION 1

Question 1(a)
Will the infection spread across the network quickly?

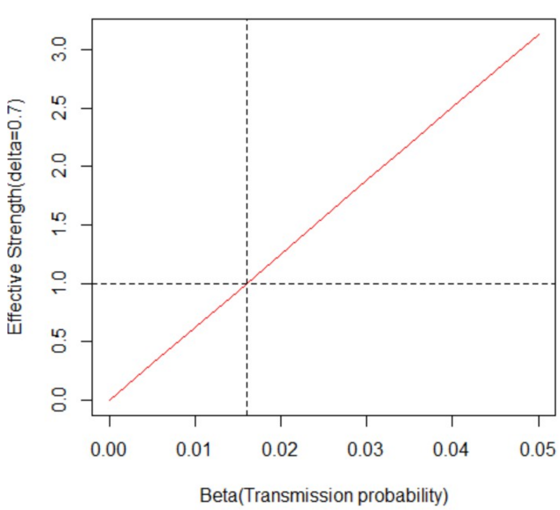
Answer
For beta1=0.2 and delta1=0.7
Effective strength=12.53
Since s > 1
Hence YES the virus will spread quickly.

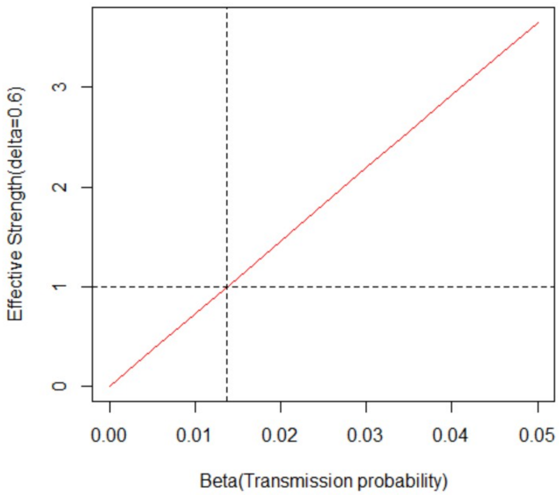
For beta2=0.01 and delta2=0.6
Effective strength=0.73
Since s < 1
Hence NO the virus will NOT spread quickly.

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

Answer
For δ=0.7 and required that s=1
β=0.015

For δ=0.6 and required that s=1
β=0.013



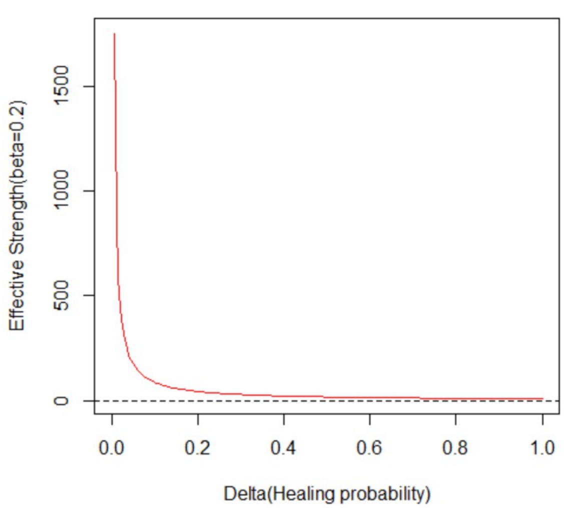


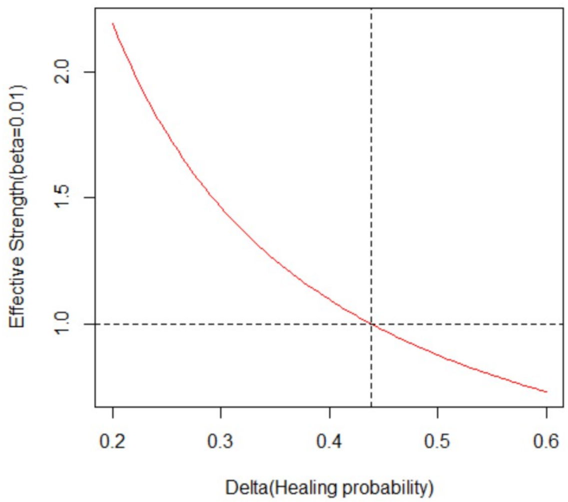
The effective strength of the virus is directly proportional to the transmission probability. Hence we can see the linear relationship in the plot. As the β increases s increases as well. This supports the rational that is a virus is highly contagious it will have a high β and consequently a higher effective strength.

Question 1(c)
Keeping β fixed, analyze how the value of δ affects the effective strength of the virus (suggestion: plot your results). What is the maximum healing probability (δ) that results in a network-wide epidemic?

Answer
For β=0.2 and required that s=1
δ=1 (Even with maximum healing probability the virus' effective strength cannot be brought under 1)

For β=0.2 and required that s=1
δ=0.43

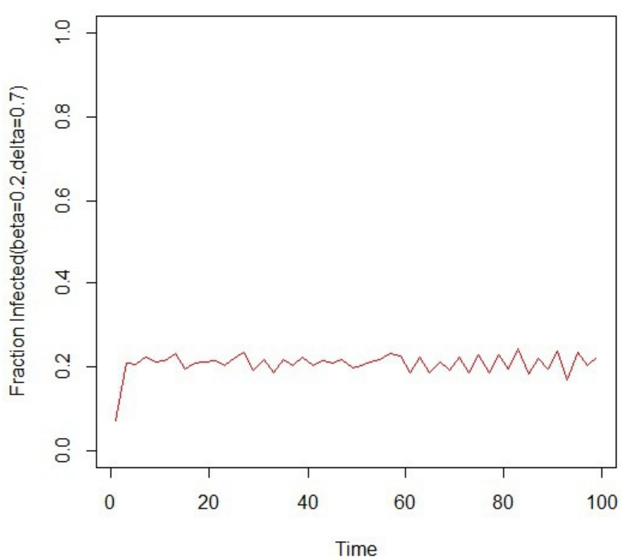




The effective strength of the virus is inversely proportional to the healing probability for the virus. This can be seen in the plot as well. Thus if it is easier to heal the virus then the effective strength drops as supported by the plot.

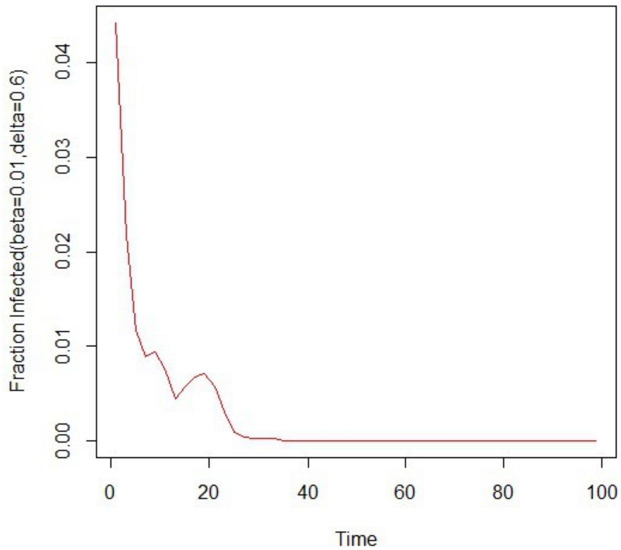
Question 2

Simulation for Fraction of Infected nodes using $\beta=0.2$, $\delta=0.7$



As expected from the Effective strength we can see that virus spreads through the network. (s>1)

Simulation for Fraction of Infected nodes using β=0.2, δ=0.7



As expected due to low effective strength the virus dies off. (s<1)

Question 3(a)
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.

Answer
Given a limited number (k) of vaccinations. The optimal immunization policy involves selecting the nodes of the contact network that maximally reduce the spectral radius of the graph. That is we need to find the subset of k nodes which when removed from the graph (along with their incident edges) will result in the largest drop in the largest eigen value of the graph.

The time complexity of finding such an optimal subset would be to try all possible subsets of nodes of size k and smaller. For each subset we would remove the nodes from the graph (update graph) and find the eigenvalue of the new contact graph. We would have to compute the reduction achieved and update the best known reduction if the current reduction is better.
Overall there would be $\binom{n}{k}$ possible subsets. For each subset the Graph Update would take $\theta(n^2)$ time. Computing eigenvalue will take $\theta(n^3)$.
Total complexity = $\binom{n}{k} * \theta(n^3)$

Here I have assumed the graph is represented as an Adjacency matrix as this would be required for eigenvalue decomposition anyways.

As we can see that the overall complexity of the algorithm is very high (exponential) it would become infeasible for graphs of reasonable size. The reasonable numbers here would be at least in the factor of 10^5, because we would be modeling it over actual population sizes.

Question 3

Answer

Policy A:

<u>Intuition</u>

This policy has no real strategy involved. This approach might be appropriate if there is no knowledge about the contact network. However such a strategy might warrant its use even when we know the contact network. Example: when we cannot select people we want to immunize or have no control over immunization administration (in cases where immunization is voluntary).

<u>Pseudo code</u>

```
PolicyA(G,k)
        random_immunization_sample = random_sample(G.V,k)
        G.delete( random_immunization_sample)
```

The complexity of the code is dominated by the deletion operation. Hence it is O(|V|+|E|).

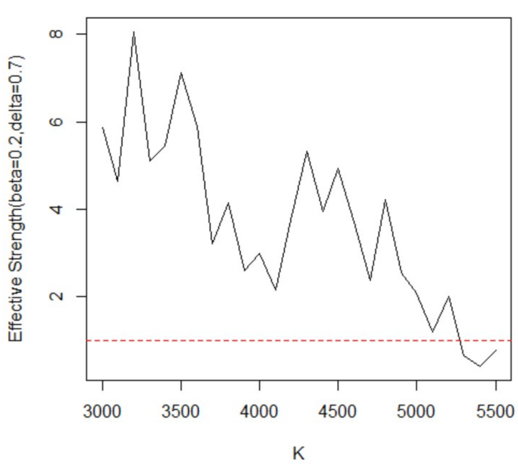
<u>Effective strength</u>

This immunization policy did reduce the effective strength but however not significantly.

s=11.34

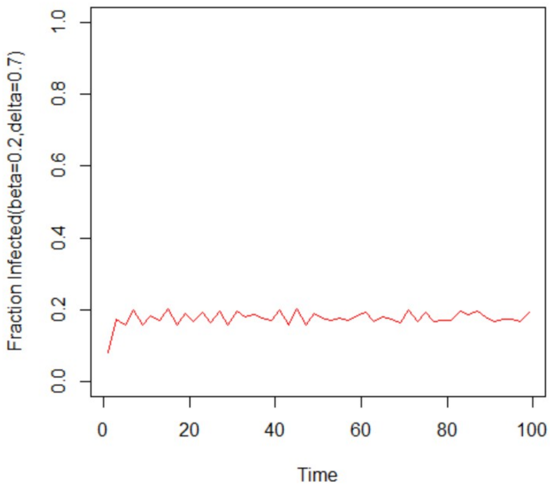
Since s>1 it will still result in epidemic.

<u>Vaccines Needed</u>



The number of immunization needed to stop the epidemic is close to 5200. Which is very ineffective as it close to the size of the graph itself.

Immunized simulation



As expected due to high Effective Strength of the virus despite random immunizations we can see that the virus persists in the network and becomes an epidemic.

Policy B:

Intuition

This policy works on the intuition that the most connected nodes would be result in the maximum propagation of the virus. If such highly connected nodes get infected they in turn can infect many other nodes because they would be in contact with the largest number of other nodes. Hence immunizing such connected nodes can severly hamper the rampant spread of the virus.

Pseudo code

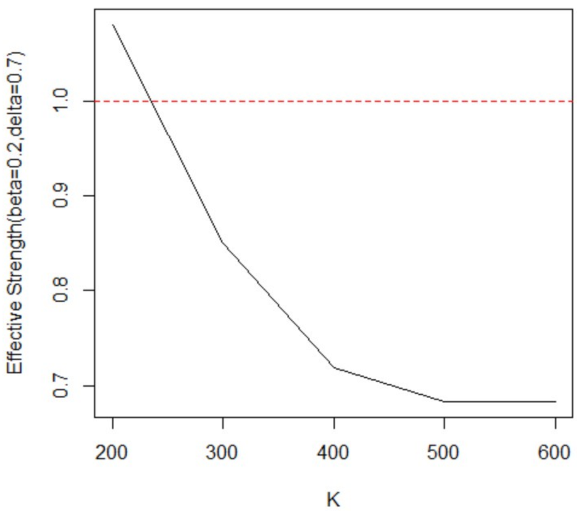
```
PolicyB(G,k)
        nodes_degree = G.degree()
        sort(nodes_degree)
        immunization_sample=select(1,k,nodes_degree) // Select top K nodes
        G.delete(immunization_sample)
```

The complexity of the program would be sum of the operation to find the degree of all the nodes: O(|V|), operation to sort all the nodes based on their degree O(|V|log|V|), selecting the top k nodes with the highest degree O(1) and finally deleting these nodes O(|V|+|E|).
Hence total O(|V|+|E|+|V|log|V|)

Effective strength

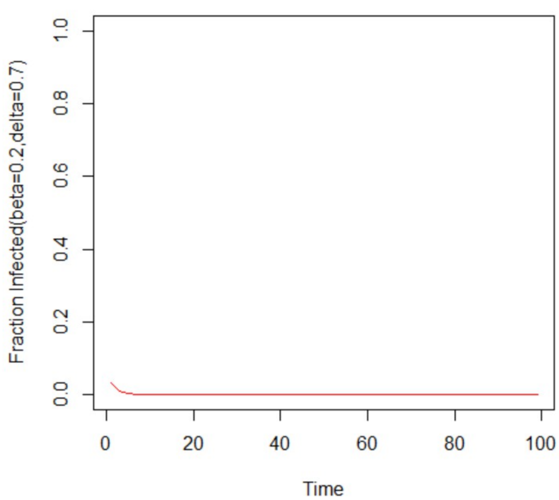
This immunization policy did reduce the effective strength but however s remained over 1.
s=1.080
Since s>1 it will still result in epidemic.

Immunizations needed



As we can see from the graph the effective strength falls around 250 vaccinations. This is an effective number considering the size of the graph.

Simulation



Contrary to the effective strength computed we see that the virus is brought under control just by administering 200 vaccines.

We must note that due random nature of initial infections we might end up with different results for this simulation since the value of s very close to 1. Hence the virus might end up dying off if the right nodes do not get infected.

Policy C:

Intuition

This builds on the policy B and argues that since we already immunize a highly connected node it no longer will act as a host and will not be involved in the virus propagation. Thus once a node is immunized we no longer should consider it in our virus propagation network. Every node that is connected to an immunized node will lower its chances of infection. To take this into account we need to recalculate the new nodes that might be most susceptible given that a node in the network is already immunized and harmless. This policy might depend on the fact that immunization is permanent or the probability of immunization loss is very low.

Pseudo code

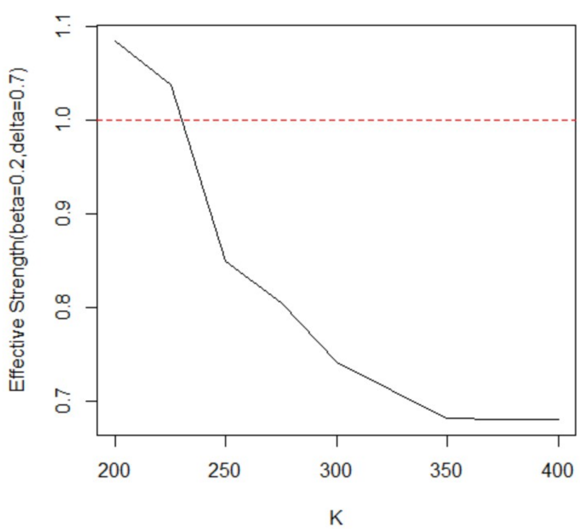
```
PolicyC(G,k)
        for i from 1 to k
        {
                node_degree=G.degree()
                immunization_node=max(node_degree)
                G.delete(immunization_node)
        }
```

The complexity of the code in each iteration is O(|V|) to compute degree of all nodes plus O(|V|) to find the node with max degree plus the cost of deleting that node from the graph O(|V|+|E|). Hence to total over all iterations would be: O(k|V|+k|E|)

Effective Strength

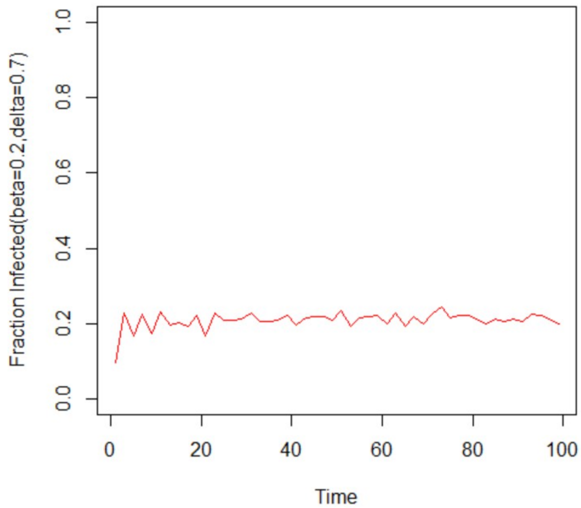
This immunization policy did reduce the effective strength.
S=1.085
Since s>1 it will still result in epidemic.

Immunizations needed



As we can see from the plot that the effective strength drops to below 1 for vaccinations around 250. Thus the virus can be brought in control with almost 250 vaccinations which is reasonable in the size of the input.

Simulation



In accordance to the Effective Strength calculated we can see that the virus survives and propagates.

We must note that due random nature of initial infections we might end up with different results for this simulation since the value of s very close to 1. Hence the virus might end up like an epidemic if the right nodes get infected.

Policy D:

Intuition

This policy tries to approximate the optimal strategy of immunization outlined in 3(a). Since the optimal policy revolves around reduction of the connectedness of the graph which is estimated by the largest eigenvalue, this policy directly tries to reduce the largest eigenvalue by immunizing the nodes that contribute towards this large eigenvalue. The maximum contributing nodes are present in the eigenvector with the largest values. Hence the policy is to simply select the k largest nodes for immunization and hope to come close to the optimal immunization.

Pseudo Code

```
PolicyD(G,k)
        largest_eigenvector=eigenvectors(G)
        sort( largest_eigenvector)
        vector_nodes=select(1,k,largest_eigenvector)
        immunization_nodes=map(vector_nodes)

        G.delete(immunization_nodes)
```

The complexity of the code is dominated but the eigen vector computation which can be optimized to take $O(|V|^2)$ time. Sorting takes $O(|V|\log|V|)$ time and the deletion operation takes $O(|V|+|E|)$ time. Hence total: $O(|V|^2+|E|)$

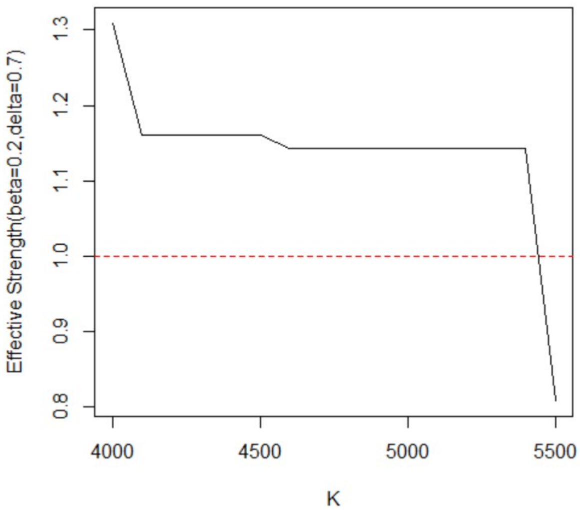
Effective strength

This immunization policy did reduce the effective strength.

S=3.53

Since s>1 it will still result in epidemic.

Immunizations Needed



As we can see from the plot in order to bring the virus under control we will have to administer as many vaccines as the size of the graph (close 5000). Thus showing that this strategy is ineffective.

BONUS SECTION

VIRAL MARKETING

If we model the marketing campaign as the virus we want to spread through a social network, we can see great similarities of VPM in marketing. These have been tabulated below:

Elements	Virus Propagation	Viral Marketing
Nodes	People	Social Network Identity
Edges	People who come in contact in a way that can propagate the virus	Social Network entities that are related (friends / followers etc.) and can possibly assert influence
β	Probability of virus transmission	Probability of spreading the message that leads to a purchase or similar favorable outcome
δ	Probability of Healing	Probability of message getting lost or ignored
Model	SIS	Similar to SIS. Each entity is either affected by the message and creates a favorable outcome. When under the marketing influence continues to purchase a produce etc. (infected). When the marketing campaign becomes ineffective the entity loses interest and goes back to being open to new influences (susceptible).

Concluding from the learning from VPM for static networks we see that advertising campaigns that are aimed at the most connected people in a social network would yield the best results in making the campaign most effective and spread out.

Hence a policy similar to B which targets the most connected people will result in better spreading the advertised message. Policy would also be highly effective but there is not equivalent to “removing” the person already targeted from the social network.
Policy B and C would be the best to implement marketing strategies.