Read Me
Project: Virus Propagation
Unity Id: ndgandh2
Option 1: Static Contact Network

Goal: Analyze the propagation of a virus in a static contact network and prevent a network-wide epidemic.

Input: Parameter values for experiments:

- Transmission probabilities $\beta 1 = 0.20$ and $\beta 2 = 0.01$.
- Healing probabilities $\delta 1 = 0.70$ and $\delta 2 = 0.60$.
- Number of available vaccines $k1 = 200$.

Graph: Static contact network (i.e., one undirected unweighted graph):

- static.network

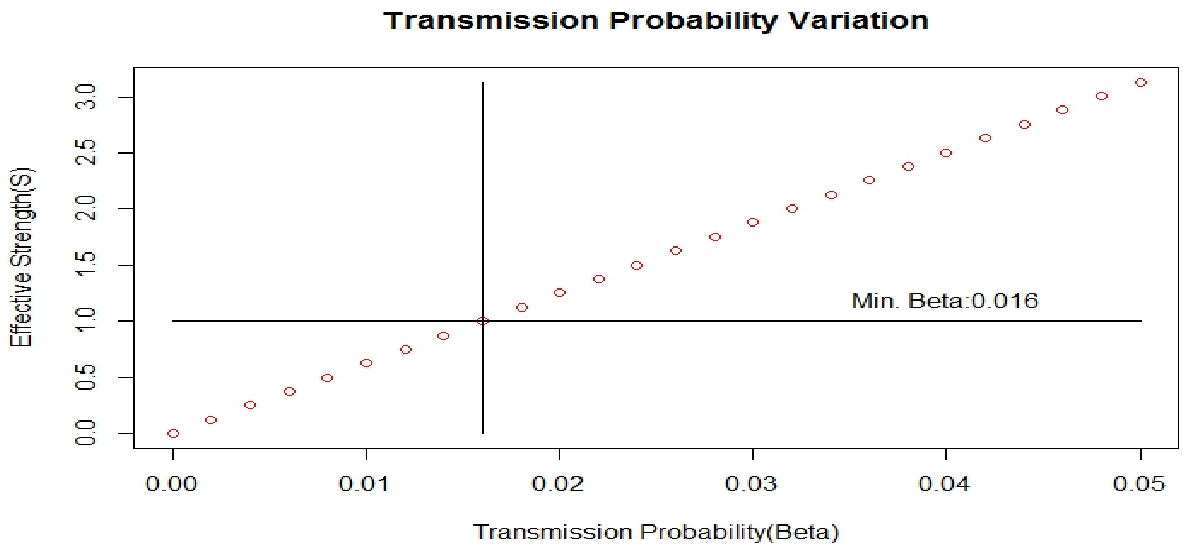
Q1. For the SIS (susceptible, infected, susceptible) Virus Propagation Model (VPM), with transmission probability $\beta = \beta 1$, and healing probability $\delta = \delta 1$, calculate the effective strength (s) of the virus on the static contact network provided (static.network). See supplementary material provided for details on the SIS VPM and on how to calculate the effective strength of a virus. Answer the following questions:

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

- Largest Eigen Value: 43.8547
- Effective Strength: 12.52991
- Since the effective strength(s) > 1, infection will grow and result in an epidemic.

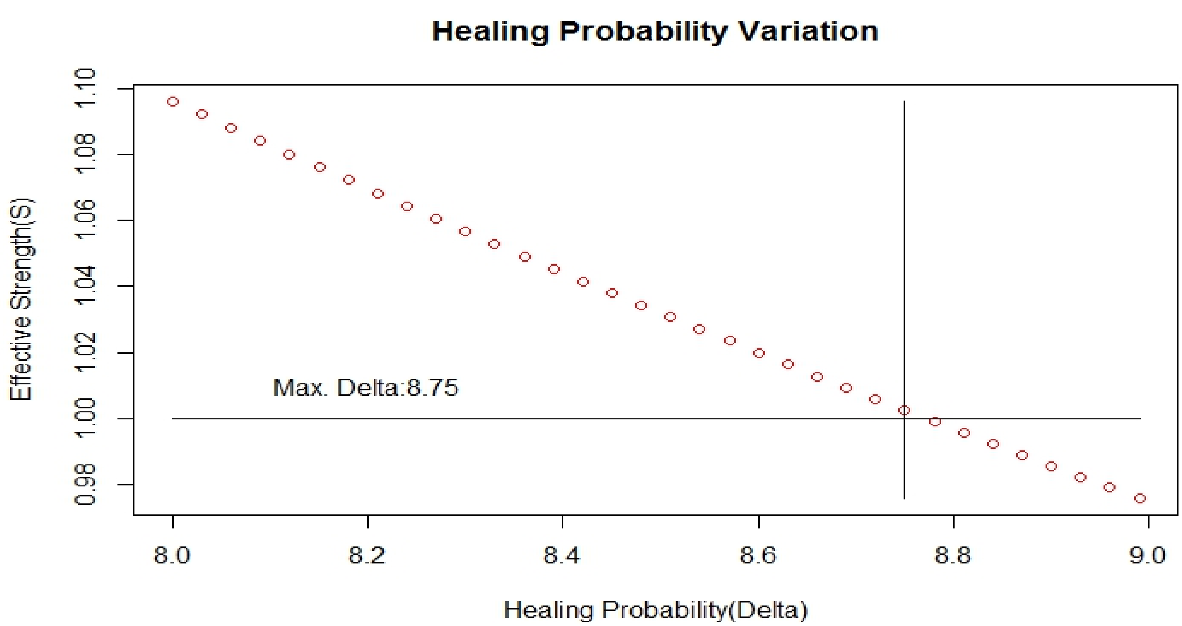
b. **Keeping δ fixed, analyze how the value of β affects the effective strength of the virus. What is the minimum transmission probability (β) that results in a network-wide epidemic?**

- Minimum transmission probability (β) that results in a network-wide epidemic: 0.016



c. **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 network-wide epidemic?**

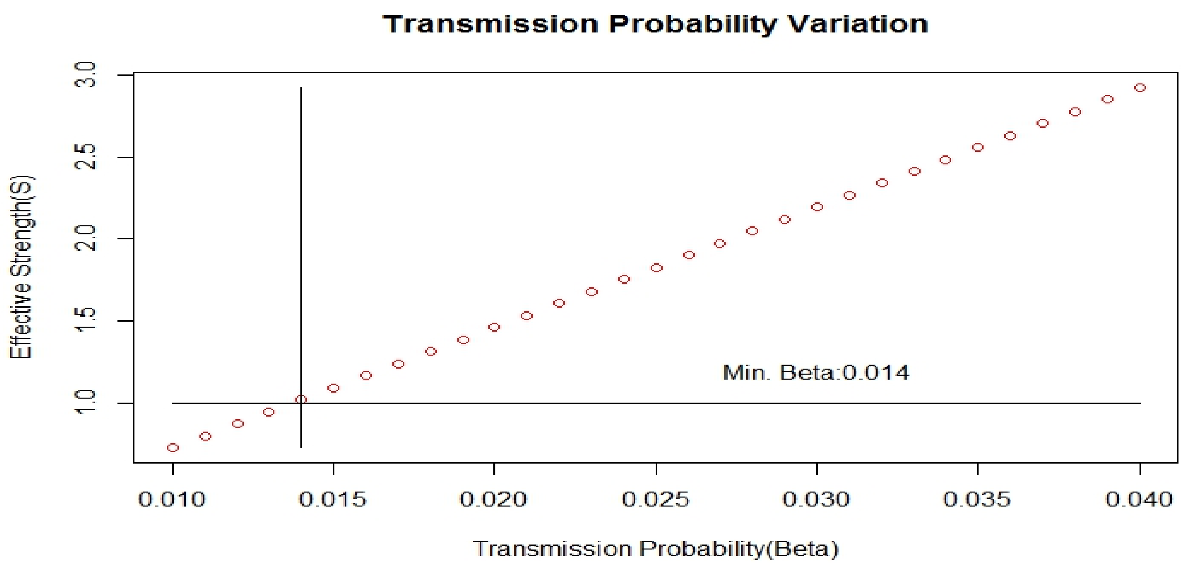
- Maximum healing probability (δ) that results in a network-wide epidemic: 8.75



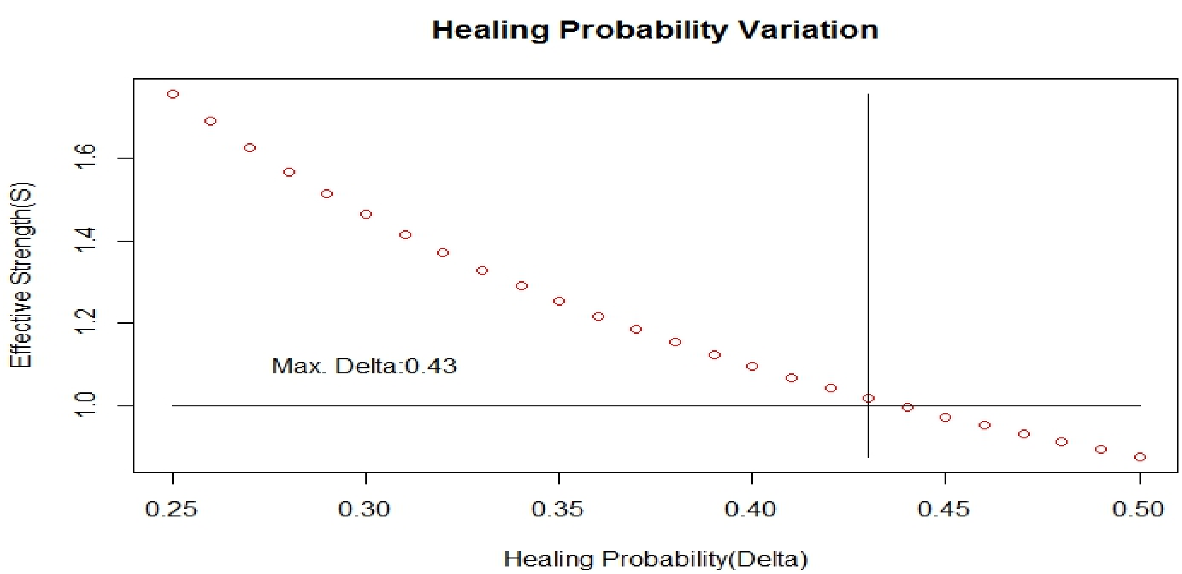
d. Repeat (1), (1a), (1b) and (1c) with $\beta = \beta_2$, and $\delta = \delta_2$.

- Largest Eigen Value: 43.8547
- Effective Strength: 0.7309116
- Since the effective strength(s) < 1, infection will die quickly and there will be no epidemic.

- Minimum transmission probability (ß) that results in a network-wide epidemic: 0.014



- Maximum healing probability (d) that results in a network-wide epidemic: 0.43

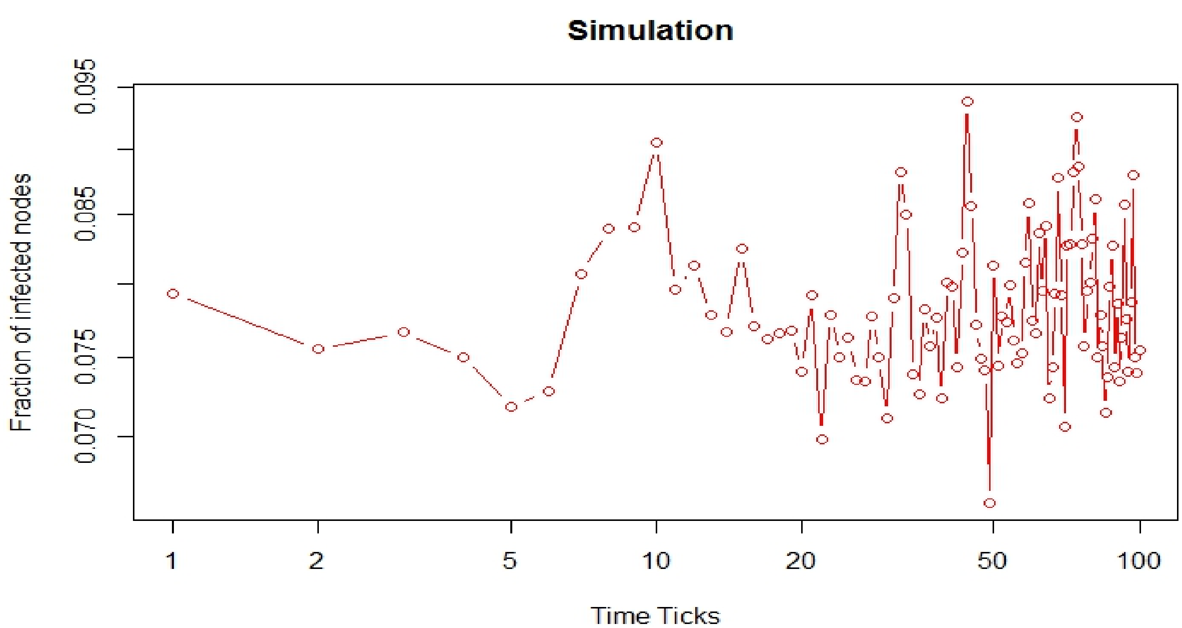


Q2. Write a program that simulates the propagation of a virus with the SIS VPM, given a static contact network, a transmission probability (β), a healing probability (δ), a number of initially infected nodes (c), and a number of time steps to run the simulation (t). The initially infected nodes should be chosen from a random uniform probability distribution. At each time step, every susceptible (i.e., non-infected) node has a β probability of being infected by neighboring infected nodes, and every infected node has a δ probability of healing and becoming susceptible again. Your program should also calculate the fraction of infected nodes at each time step.

a. **Run the simulation program 10 times for the static contact network provided (static.network), with $\beta = \beta1$, $\delta = \delta1$, $c = n/10$ (n is the number of nodes in the network), and $t = 100$.**

   Number of initially infected nodes: 571
   Virus Infection is spread across the network

b. **Plot the average (over the 10 simulations) 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)?**

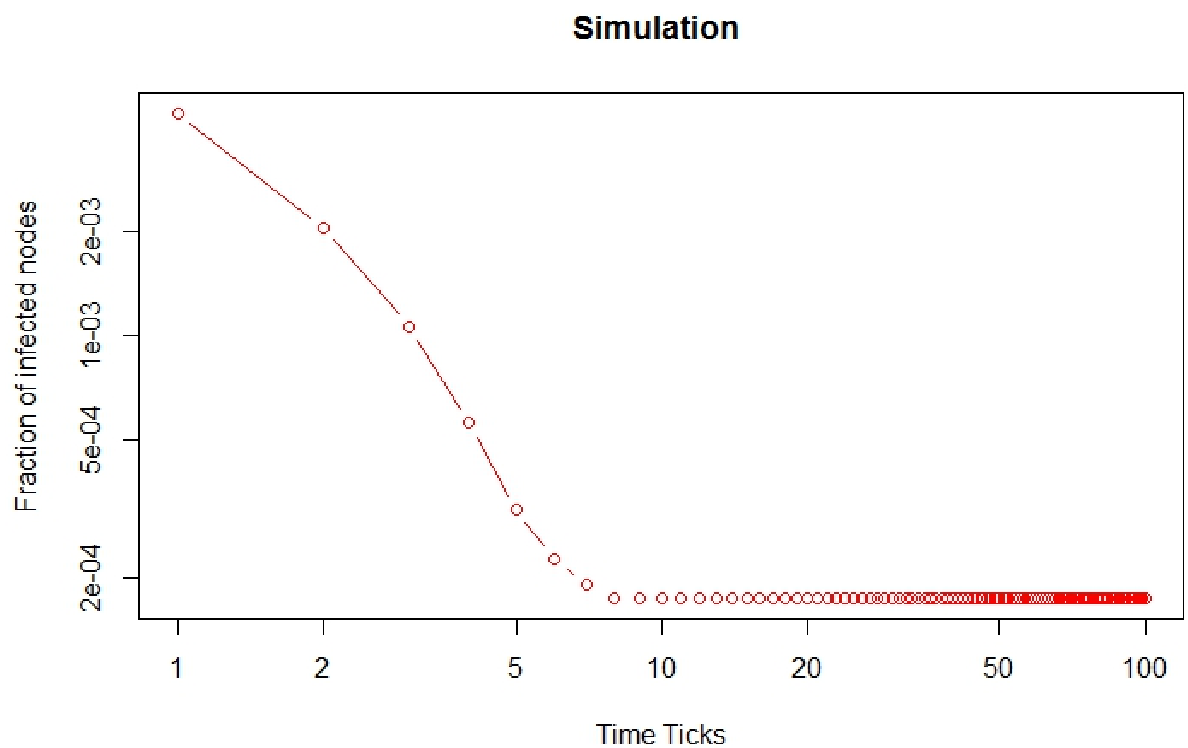


The infection in the network spread and did not die.

Yes, the results of the simulation agree with the answer of 1(a) - infection resulted in an epidemic!

c. **Repeat (2a) and (2b) with $\beta = \beta 2$, and $\delta = \delta 2$.**

Nodes initially infected: 571
Virus epidemic has been prevented



The infection in the network quickly died out.

Yes, the results of the simulation agree with the answer of 1(d) - infection did not result in an epidemic and quickly died out.

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.

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.**

- The optimal immunization policy would be the one in which we select k such nodes in the network, removing which causes a large marginal decrease in the largest eigenvalue $\lambda 1$.

- Time complexity for this policy would be very high because we need to calculate eigen value for each of the subgraphs of k nodes.
- From the immunization policies A, B, C and D, policy B would be the most optimal because it caused the largest decrease in $\lambda 1$, and caused the epidemic to die quickly.
- Time complexity of this policy is reasonable because this policy doesn't require any high computational calculations.

For your program, use the following heuristic immunization policies:
- 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.

For each heuristic immunization policy (A, B, C, and D) and for the static contact network provided (static.network), answer the following questions:

b. **What do you think is the intuition behind this heuristic?**
c. **Write a pseudocode for this heuristic immunization policy. What is its time complexity?**
d. **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?**
e. **Keeping β and δ fixed, analyze how the value of k affects the effective strength of the virus on the immunized contact network. Estimate the minimum number of vaccines necessary to prevent a network-wide epidemic.**
f. **Given k = k1, β = β1, δ = δ1, c = n/10, and t = 100, run the simulation from problem (2) for the immunized contact network 10 times. Plot the average fraction of infected nodes at each time step. Do the results of the simulation agree with your conclusions in (3d)?**

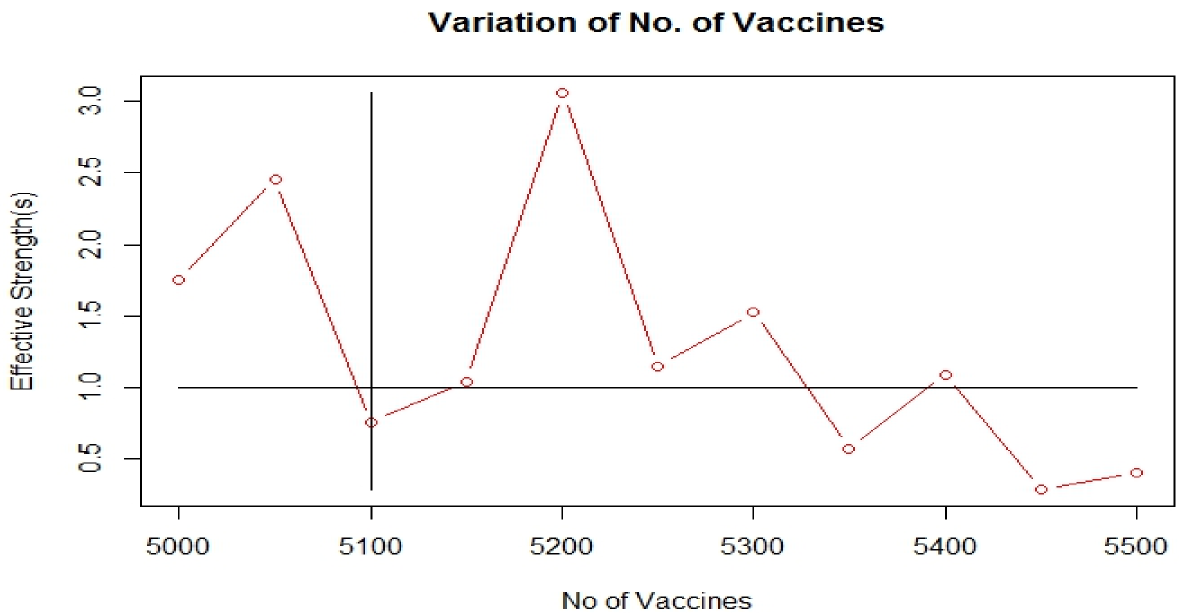
POLICY A:
Intuition:
Select k random nodes for immunization. Intuition behind this policy is that if the same policy is applied for a number of times, sometimes epidemic would be prevented.

PseudoCode:
```
function randomNodeImmunization(graph, k)
        NV  = number of vertices in the graph
        ImmunizableNodes = sample(NV, k)
        graph.deleteVertices(ImmunizableNodes)
```

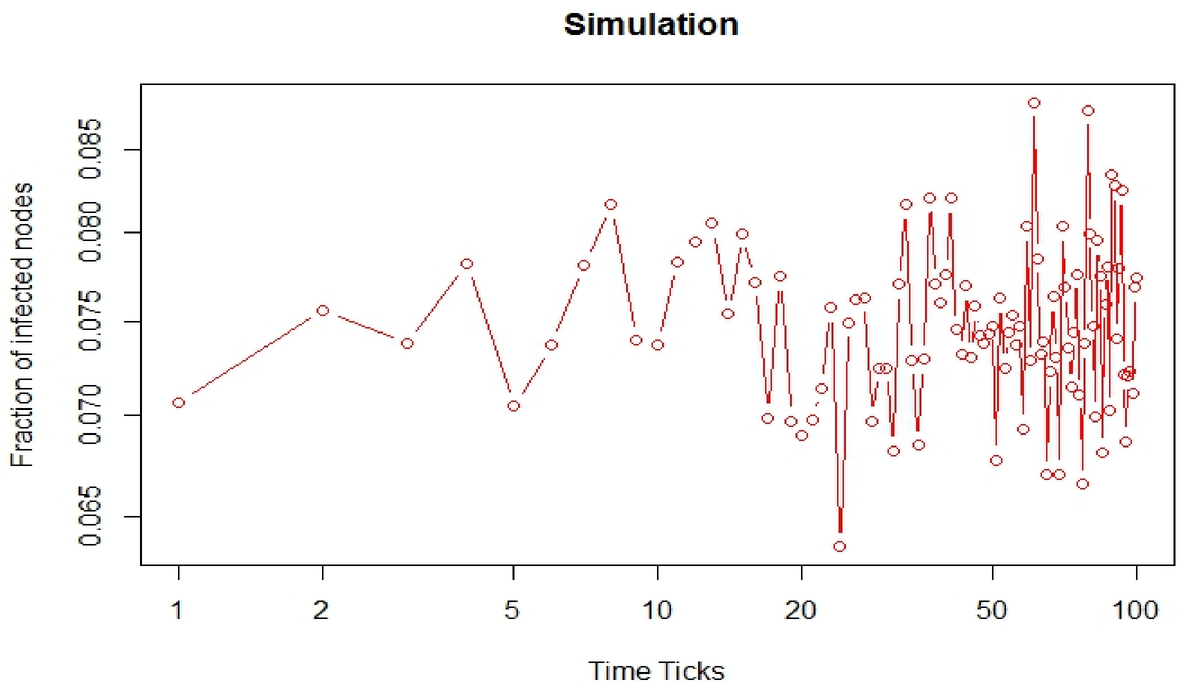
return graph

Immunized Contact Network Analysis:
Largest Eigen Value: 42.55883
Effective Strength: 12.15966

After applying the immunization policy also, the effective strength of the virus is still > 1 and the epidemic will not be prevented.



Above variation of k shows that a very large number of vaccines (around 5100 - 5400) will be required to prevent the epidemic.



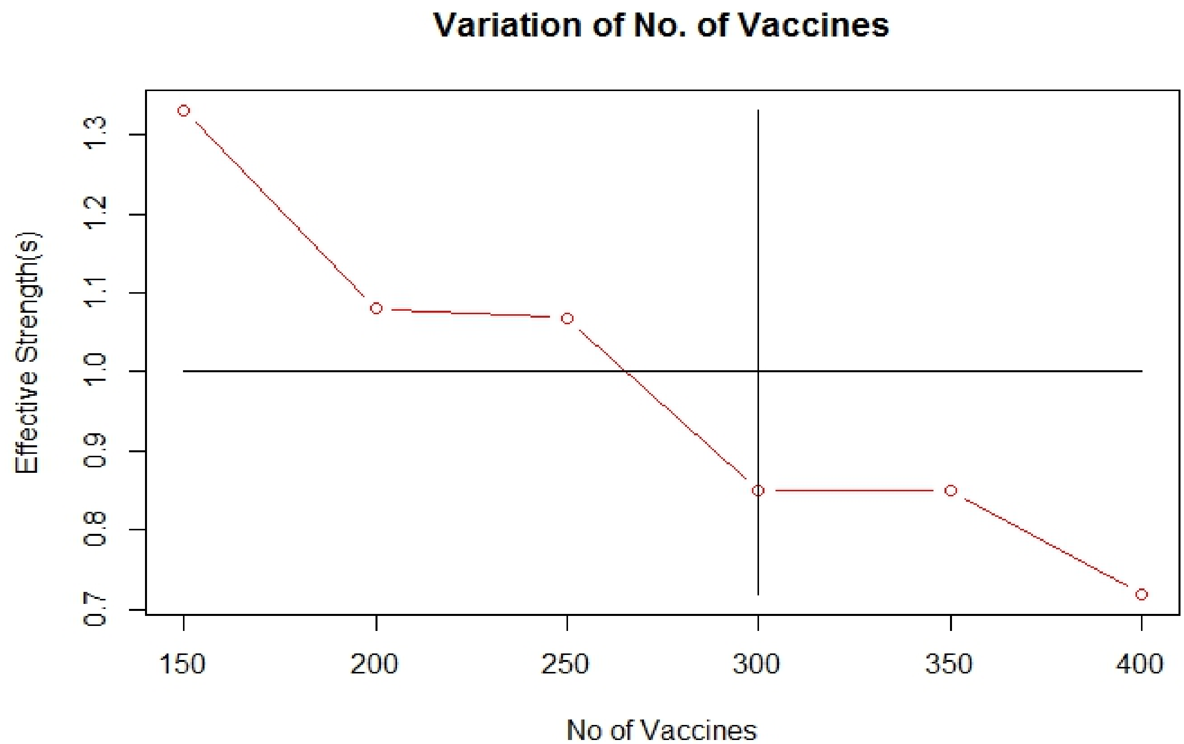
Yes, the simulation results agree with the conclusion in 3(d).

POLICY B:
Intuition:
Intuition of this policy is to remove the nodes with the highest degree. Removing highest degree nodes, makes the graph more disconnected and virus propagation becomes difficult.

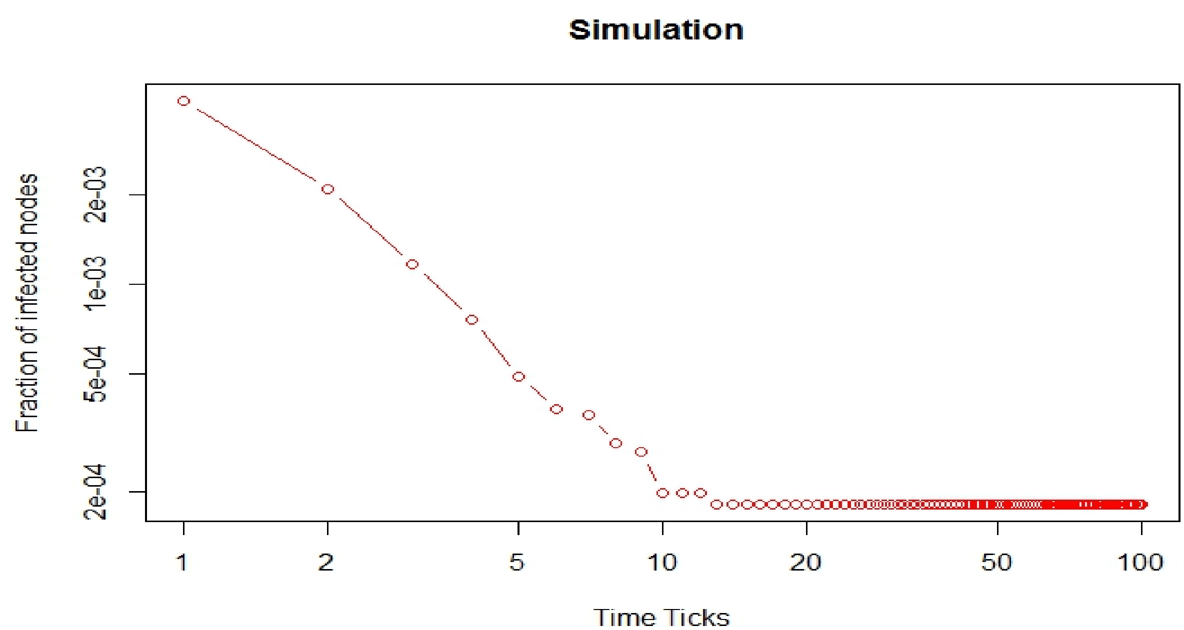
PseudoCode:
```
function highestDegreePolicy(graph, k)
        nodeDegree = degree(graph)
        ImmunizableNodes = nodeDegree[order[-nodeDegree[,1],]
        graph.deleteVertices(ImmunizableNodes)
        return graph
```

Immunized Contact Network Analysis:
Largest Eigen Value: 3.780964
Effective Strength: 1.080275

After applying the immunization policy, the effective strength of the virus has reduced by a marginal difference and the epidemic will be prevented.



Above variation of number of vaccines shows that the number of vaccines required to prevent the epidemic is around 250 - 300.



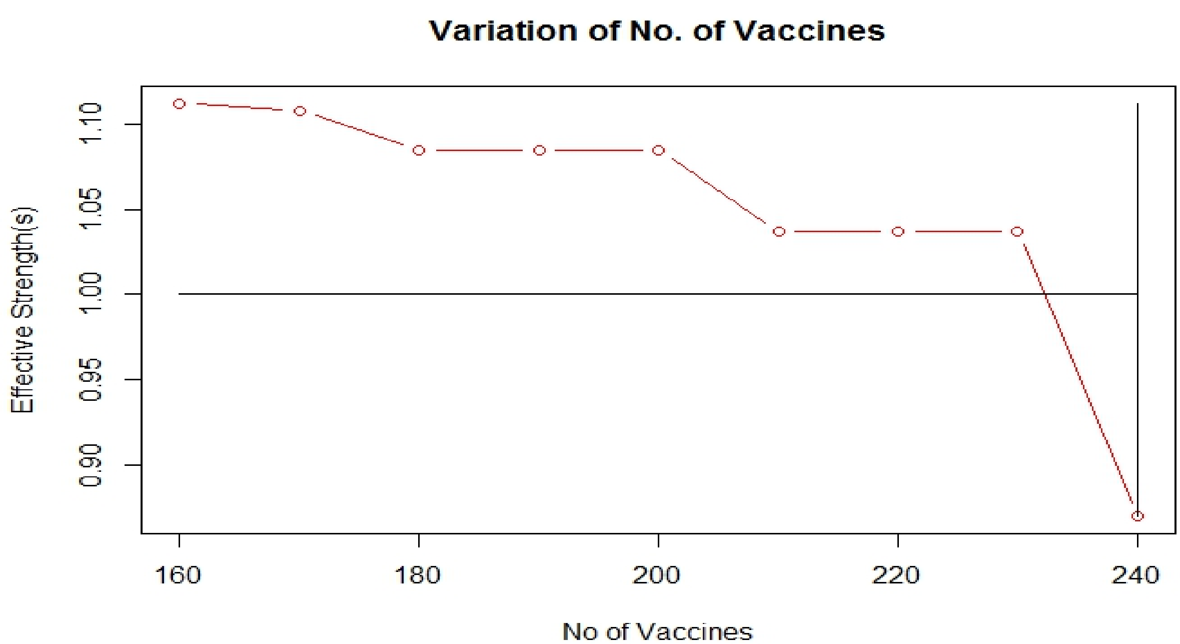
Yes, the simulation results agree with the conclusion in 3(d).

POLICY C:
Intuition:
In this policy, after removing the node with the highest degree, the degrees of remaining nodes are calculated again to get a new node with the highest degree. This process is repeated k times. Intuition is that using this method will make the graph more disconnected every time which might prevent the epidemic from spreading across the network. Computationally complexity is very high for this method, but it guarantees a marginal eigen drop after removing k nodes.

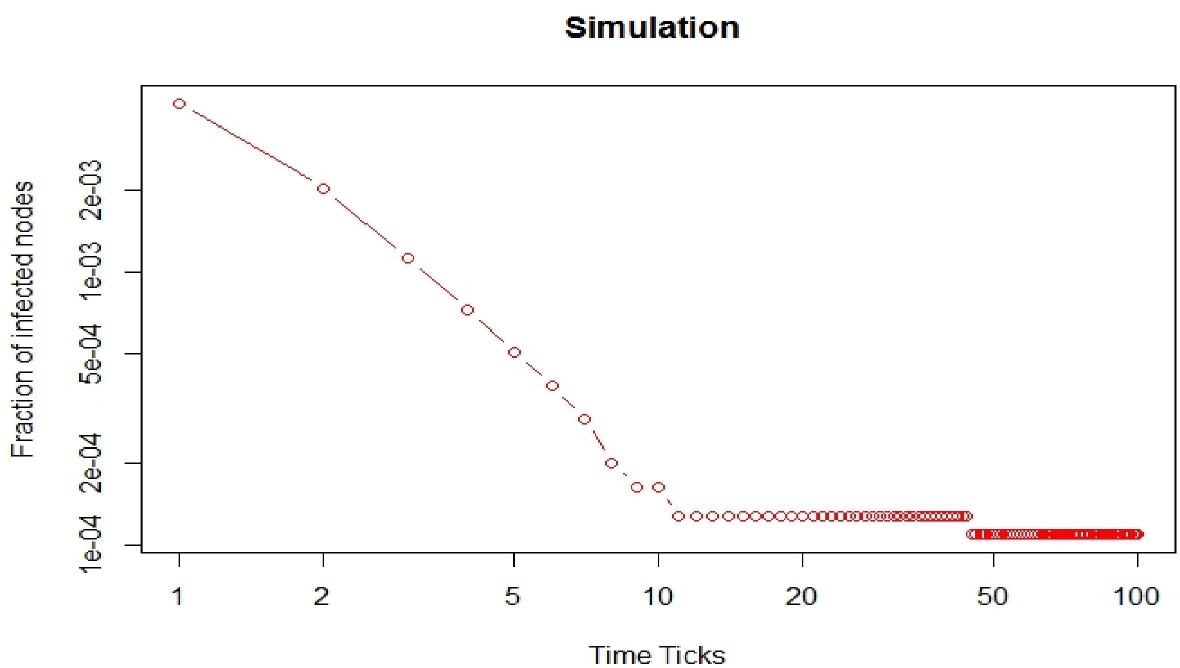
PseudoCode:
```
function highestDegreeLoopPolicy(graph, k):
    for(i in 1:k){
        nodeDeg <- which.max(degree(graph))
        graph.deleteVertices(v = nodeDeg)
    }
    return graph
```

Immunized Contact Network Analysis:
Largest Eigen Value: 3.795822
Effective Strength: 1.084521

After applying the immunization policy, the effective strength of the virus has reduced by a marginal difference and the epidemic will be prevented.



Above variation of number of vaccines shows that the number of vaccines required to prevent the epidemic is around 230 - 240.



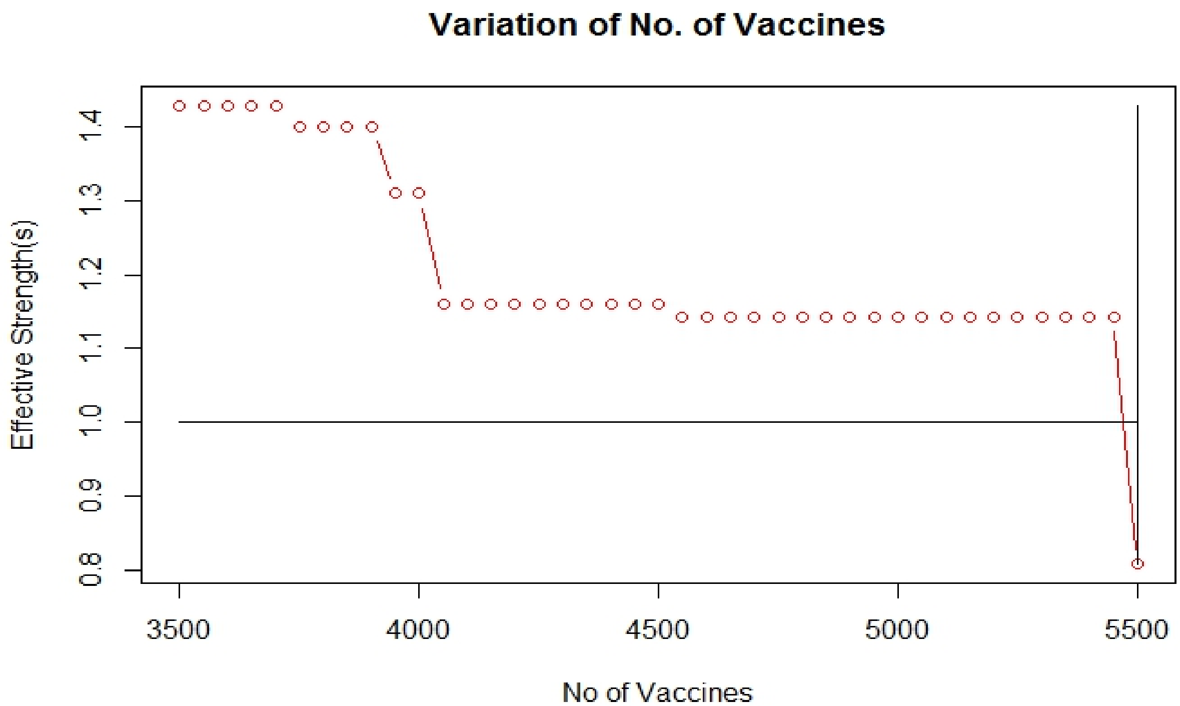
Yes, the simulation results agree with the conclusion in 3(d).

POLICY D:
Intuition:
The intuition is to remove the nodes corresponding to the largest (absolute) values in the eigenvector of the largest eigenvalue. By doing this, there will be a marginal drop in the largest eigenvalue and thus a marginal drop in the effective strength of the virus.

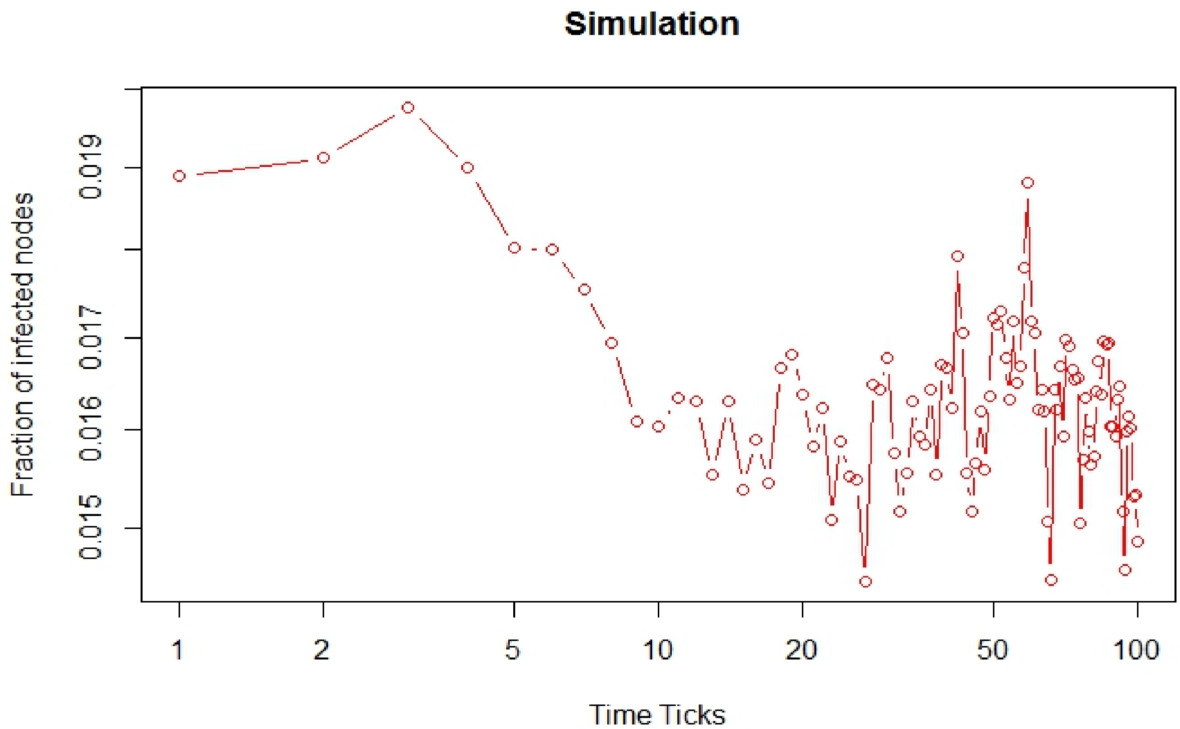
PseudoCode:
```
function largestEigenPolicy(graph, k)
        eigVector <- abs(graph.eigen(graph = G)$vectors)
        eigVector <- eigVector[order(-eigVector[,1]),]
        ImmunizableNodes <- eigDetails[1:k,2]
        graph.deleteVertices(ImmunizableNodes)
        return graph
```

Immunized Contact Network Analysis:
Largest Eigen Value: 10.74685
Effective Strength: 3.070528

After applying the immunization policy, the effective strength of the virus has reduce by a marginal difference and the epidemic will be prevented.



Above variation of k shows that a very large number of vaccines (around 5500 - 5500) will be required to prevent the epidemic.



No, the simulation results do not agree with the conclusion in 3(d).

REFERENCES:

1. Chakrabarti, Deepayan, and Christos Faloutsos. *Graph Mining: Laws, Tools, and Case Studies*. San Rafael, CA: Morgan & Claypool, 2012. Print.