SNA Report : Cascading Behaviour in Networks

Name : Maurya Sujeet
Roll no. : 17BCS016

Idea :

Let a given graph’s node have default behaviour “B” .There is some payoff for each behaviour.A new behaviour “A” will be adopted by nodes if they find out that the benefit in adopting “A” is greater than “B”.

Take 2 nodes as initial adopters.For given node find out number neighbours having particular behaviour i.e. either “A” or “B” .Now multiply number of neighbors having particular behaviour(for simplicity take “A”) with payoff of “A”.Now check for the given node that if

(Number of neighbour having “A” behaviour) * payoff of “A” > (Number of neighbour having “B” behaviour) * payoff of “B”

Then adopt “A” otherwise “B”.

If every node adopts “A” then it is called complete cascade else it will be called incomplete cascade.

Code explanation :

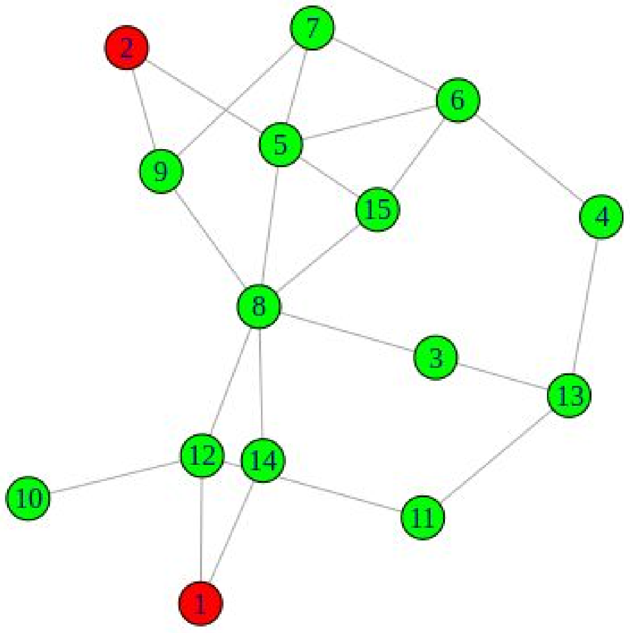
1. Take a random graph of 20 nodes and 22 edges.
2. Assign default behaviour “B” to every node

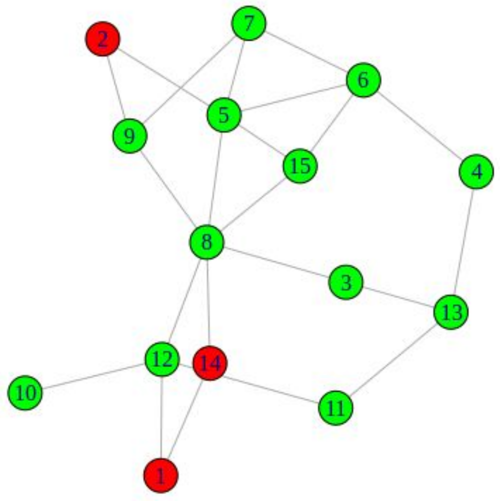
3. Take 2 initial adopters of “A”
4. Colour node having behaviour “B” and “A” with green and red respectively
5. Now calculate payoff for each node in following way:
    - Give payoff for each behaviour
    - For each node in graph
        - If node’s behaviour == “B”
            - Num_ng_a = find out how many neighbours has “A” behaviour
            - Num_ng_b = find out how many neighbours has “B” behaviour
            - T_payoff_a = Num_ng_a * payoff_a
            - T_payoff_b = Num_ng_b * payoff_b
            - Now check if T_payoff_a > T_payoff_b
            - If it's true then present node will adopt “A” or else it will keep “B”
6. Now reassign behavior of each node based on the above adoption
7. Repeat 5th and 6th step by increasing the payoff value of “A” and for every pair of nodes (initial adopters).For every pair of nodes(as initial adopters),stop recalculation when all nodes adopts “A” behaviour or stop it when even after re-calculating payoff for all nodes, no new nodes adopts behaviour “A”.

Output:

Case 1 :

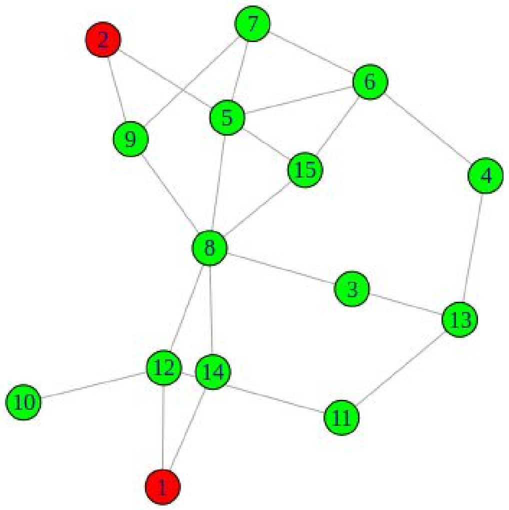
```
Payoff for b : 1
Payoff for a : 2
For nodes :  1   2
[1] "incomplete Cascade"
Cascade size :  3
```



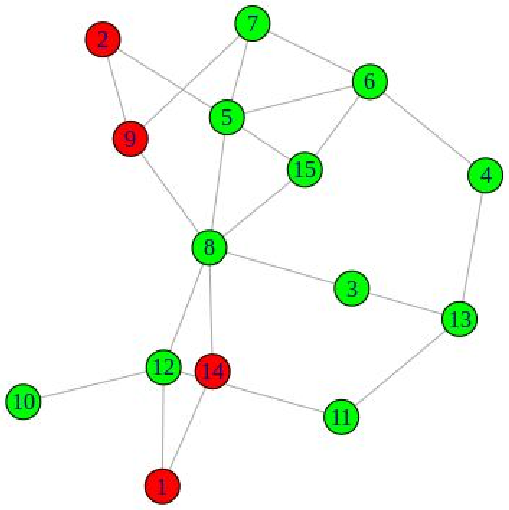


Case 2 :

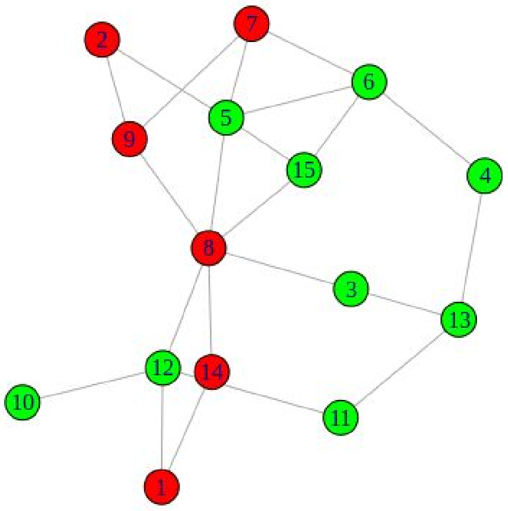
```
Payoff for b : 1
Payoff for a : 3
For nodes :  1   2
[1] "Complete Cascade"
Cascade size : 15
```



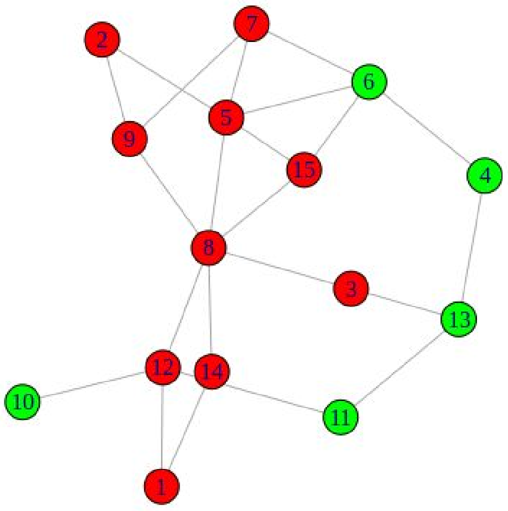
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15



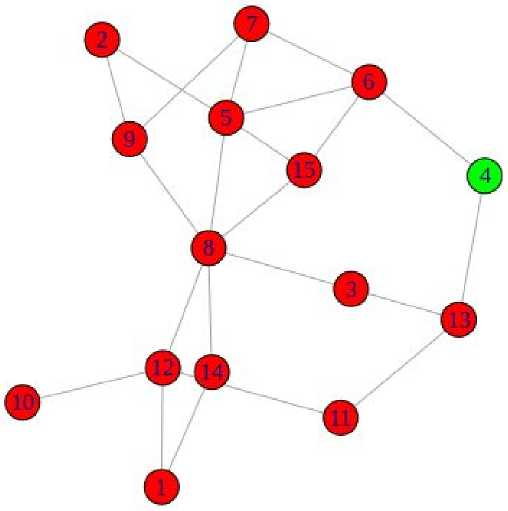
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

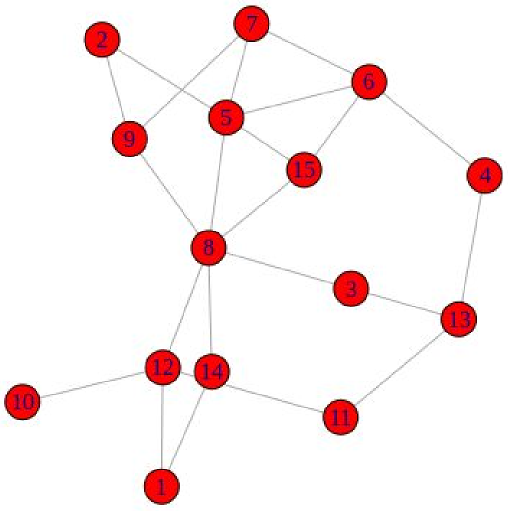


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15



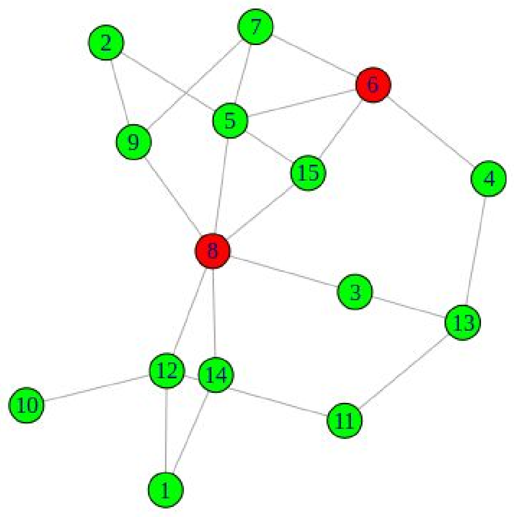
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

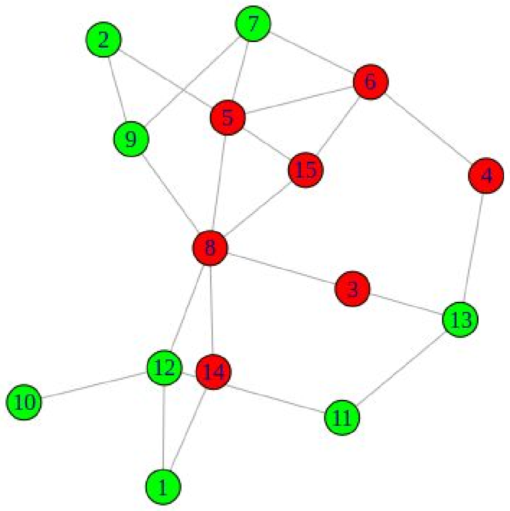

7
2
6
5
9
15
4
8
3
13
12
14
10
11
1


7
2
6
5
9
15
4
8
3
13
12
14
10
11
1

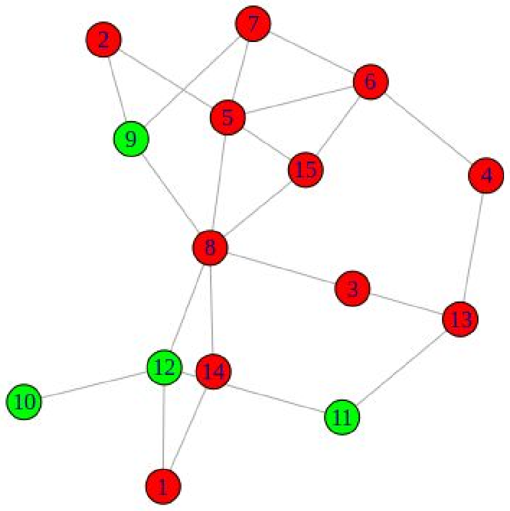
Case 3 :

```
Payoff for b : 1
Payoff for a : 2
For nodes :  8   6
[1] "Complete Cascade"
Cascade size : 15
```

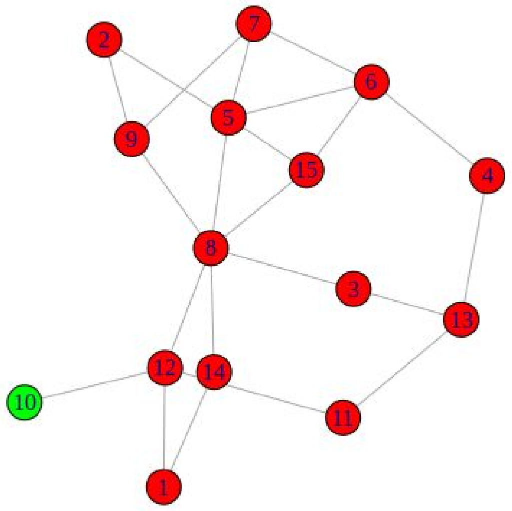


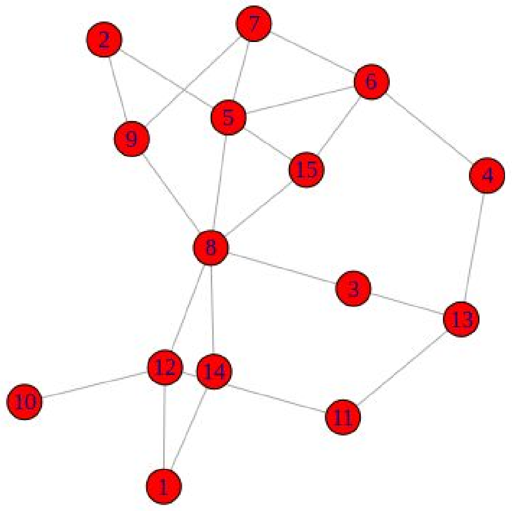


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15


2
7
6
5
9
15
4
8
3
13
12
14
10
11
1


2
7
6
5
9
15
4
8
3
13
12
14
10
11
1