University of Pisa

COMPUTER SCIENCE DEPARTMENT

Academic Year 2017/2018

Social Network Analysis

Final Report

Mehrdad Babazadeh - 568710

Gabriele Leone - 563955

Federica Trevisan - 568019

Index

1 Introduction and data collection

This following report aims to analyze the network of a Facebook user.
The data has been downloaded from networkrepository.com, an online repository of scientific data. The network is composed of **6472 nodes** and **266378 edges**. It represents the collection of the Facebook Network of the user William77, its social friendship network extracted from Facebook consisting of people (nodes) with edges representing friendship ties.
It is and undirected and unweighted network.
For this analysis the Python libraries NetworkX, Ndlib and the visual tool Cytoscape have been used.
In the first phase of the analysis, the network has been analyzed and compared with a random generated network (Erdős–Rényi model) and with a scale free network (Barabási-Albert Model). The two networks have been generated with the same number of nodes of the initial one, in order to make the comparison easier.
The report continues with the community discovery, tie strength spreading analysis and link prediction.
The code of this project is available at the following github repository:
https://github.com/federikovi/SocialNetworkAnalysis.

2 Network analysis

The following results have been obtained using Cytoscape and NetworkX. All the analysis in this phase have been calculated on the real network, on the Erdős–Rényi model and on the Barabási-Albert model. The results have been compared in the following plots.

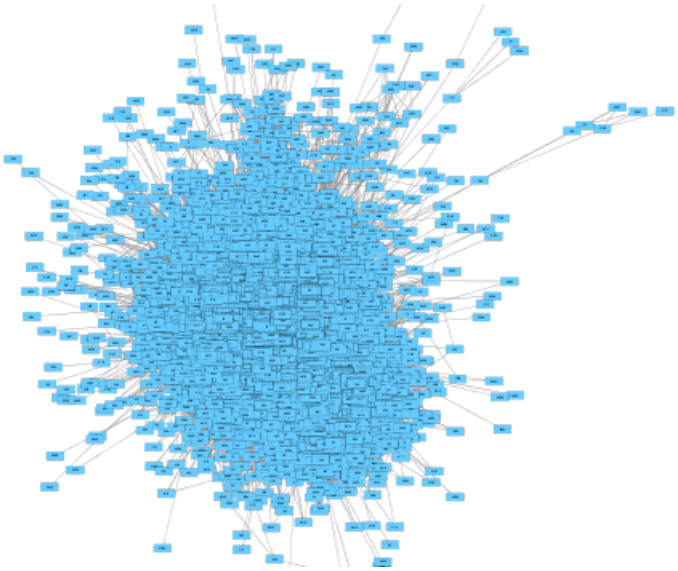

Figure 1: Facebook network visualization

The graph in the Figure 1 has been obtained using Cytoscape; as already said before it is undirected, unweighted with 6472 nodes and 266378 edges.

Clustering coefficient :	***0.254***	*Number of nodes :*	***6472***
Connected components :	***1***	*Network density :*	***0.013***
Network diameter :	***8***	*Network heterogeneity :*	***0.873***
Network radius :	***4***	*Isolated nodes :*	***0***
Network centralization :	***0.161***	*Number of self-loops :*	***0***
Shortest paths :	***41880312 (100%)***	*Multi-edge node pairs :*	***0***
Characteristic path length :	***2.605***	*Analysis time (sec) :*	***23.502***
Avg. number of neighbors :	***82.317***		

Figure 2: Real network statistics

At a first glance from the statistics it emerges that the network is composed by an unique connected component, a giant one. This implies the absence of isolated nodes and self-interactions, since no user can be friend of himself.

Degree distribution analysis

The degree distribution *p(k)* represents the probability that a randomly chosen node has degree *k*.

$$p(k) = \frac{N_{\mathrm{k}}}{N} \tag{1}$$

Where N_k is the number of nodes with degree k and N is the total number of nodes.
In a random network, the degree distribution is well approximated to a Poisson distribution, while in a scale-free network it is approximated to a power-law distribution.

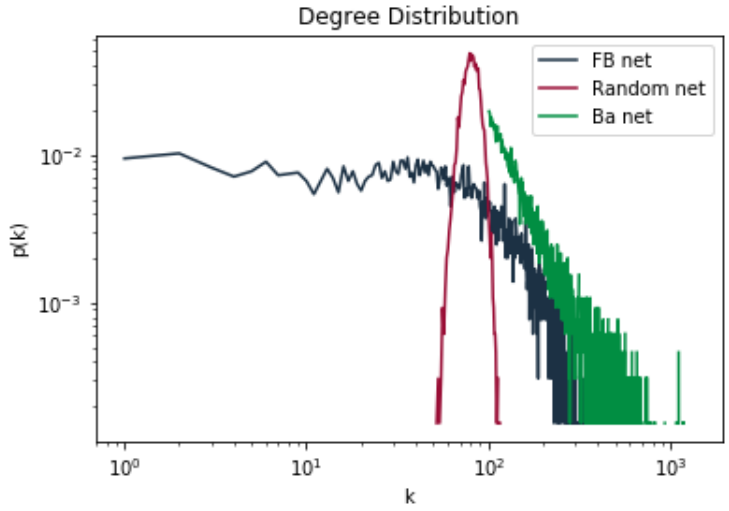

Figure 3: Degree distribution

As seen in Figure 3, both the real network and the scale-free network have a similar trend, slightly different from the random one (in red).
In addition, it can be noticed that for the real network:

- The user with highest degree is '999' with degree 106;
- There are no nodes with degree 0, because there are no isolated nodes;
- The average degree of the whole network is 82 and it has been calculated with the average degree formula for undirected graphs.

$$< k >= \frac{2L}{N} \tag{2}$$

Where L stands for number of links and N for number of nodes.

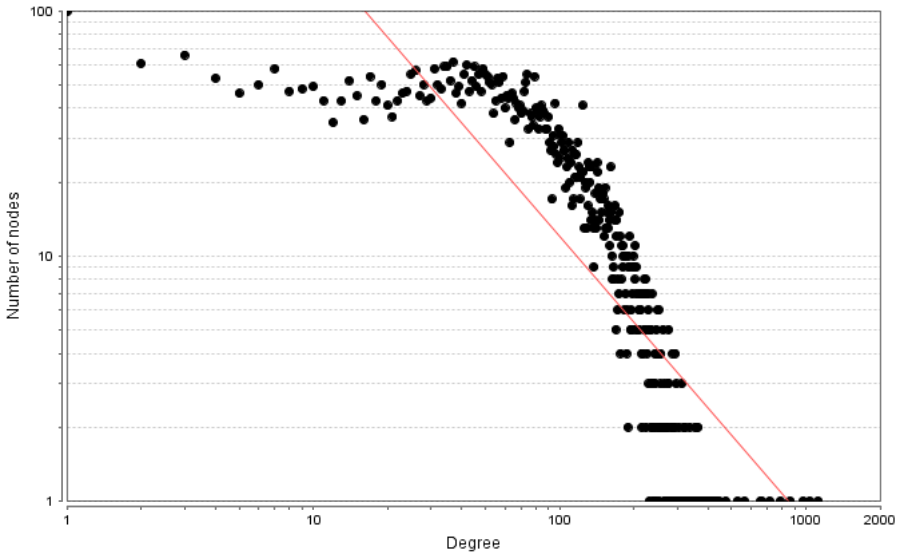

Figure 4: Degree distribution compared with a power-law

Looking at the Figure 4 it can be noticed that there's a dense amount of nodes with a small degree, and a few hubs with elevate degree.
An hub is a node with a number of links that greatly exceeds the average. The emergence of hubs is a consequence of a scale-free property of networks and is associated with power-law distribution. In this case hubs represent the active Facebook users with highest number of connections, friendships, in the social network.

Connected component analysis

The three networks are composed by an unique connected component. There are no subgraphs.

Connected components:

- Original network: 1
- ER network: 1
- BA network: 1

Path analysis

Distance in a network plays a fundamental role in determining the interactions between components of a system.
A path is a sequence of nodes with the property that each consecutive pair is connected by an edge. For each network, it has been calculated the average shortest path length.
Average shortest paths:

- Real network: 2.60508718273159
- ER network: 2.3337271699408544
- BA network: 1.9782535526478409

The following histogram represents the distribution on the shortest path lengths in the real network.

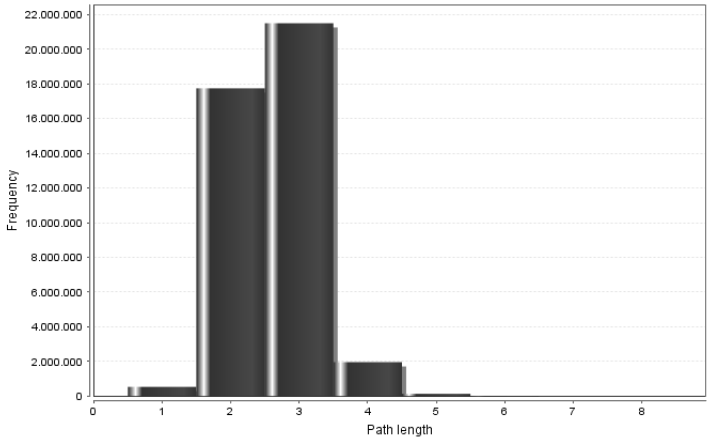

Figure 5: Shortest path length distribution for real network

The diameter of the real network, that is the maximum shortest path, is equal to 8.

Clustering coefficient, density analysis

For this analysis it has been calculated the average clustering coefficient, that represents the clustering degree of a network. The obtained values are:

- FB network average cluster coefficient: 0.2536135569835697
- ER network average cluster coefficient: 0.012708472729686632
- BA network average cluster coefficient: 0.07904584337205917

The graph density calculates the number of edges in a network with respect to the total number of possible edges.

- FB network density: 0.012720917647413897
- ER network density: 0.012720917647413897
- BA network density: 0.031028422137829345

Centrality analysis

Centrality measures help understanding how important is a node in a network. The following indicators have been analyzed:

Degree centrality

The degree centrality of a node represents the number of nodes connected to it. It measures the ability of a node to spread immediately information in the network.

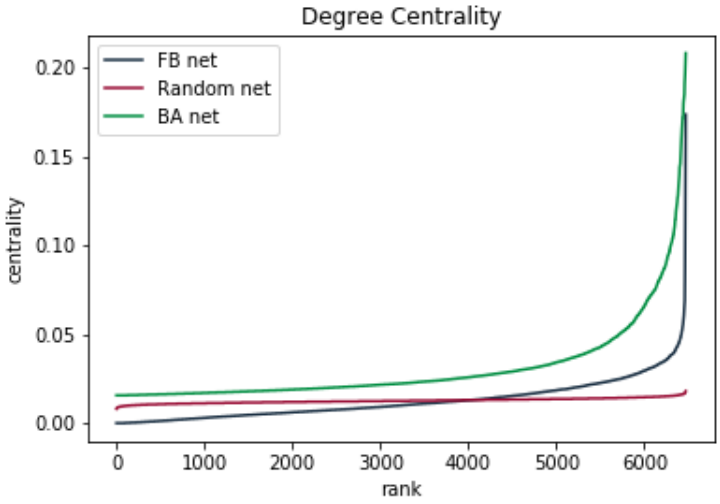

Figure 6: Degree centrality

From the Figure 6 it emerges that the random network has a low degree distribution almost uniform for all nodes. Regarding the real and scale-free networks, the majority of the nodes have a low value, that increases. This can be seen from the shape of the rising curves.
The fact that the scale-free network is the highest is an indicator of the presence of hubs.

Closeness centrality

The closeness centrality is the average distance to the neighbors.
In the Figure 7 is shown that the nodes of the scale-free network present higher values of closeness uniformly distributed. The random network also has the same trend, but with lower values of closeness. In the real network, the values are really low at the beginning, increasing in the end. The higher the closeness, the higher the centrality of a node.

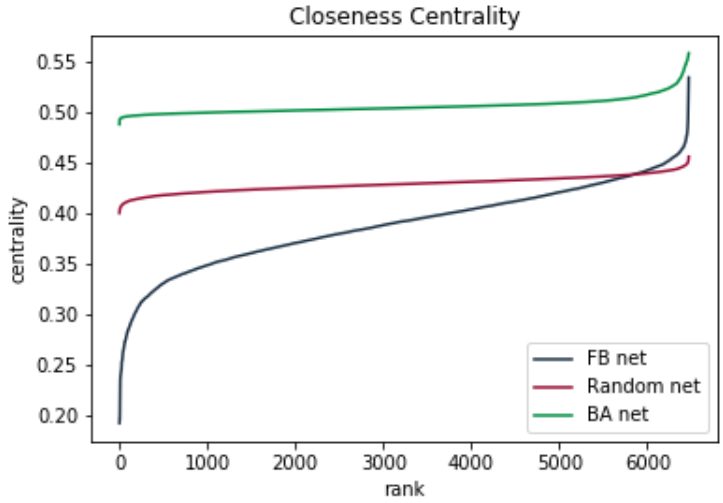

Figure 7: Closeness centrality

Betweenness centrality

The betweenness centrality calculates the number of shortest paths that go through a node. It gives information on the importance of a node: the higher the betweenness, the more important is the user profile of the network.

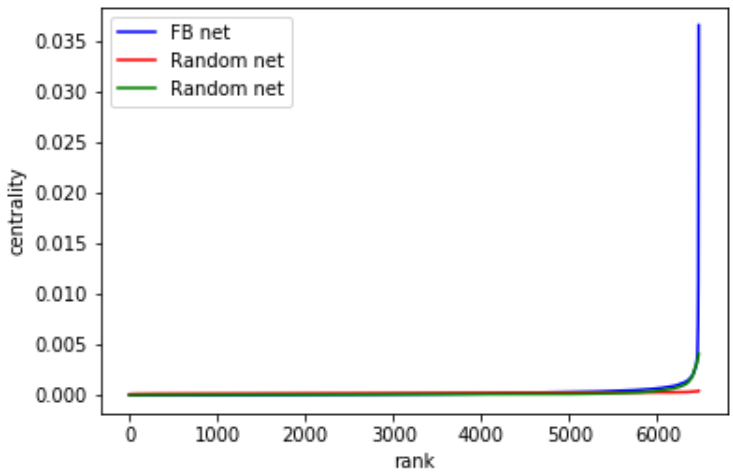

Figure 8: Betweenness centrality

In this case the networks have similar trends, with low values slightly increasing only at the end, in particular for real network.

Page rank

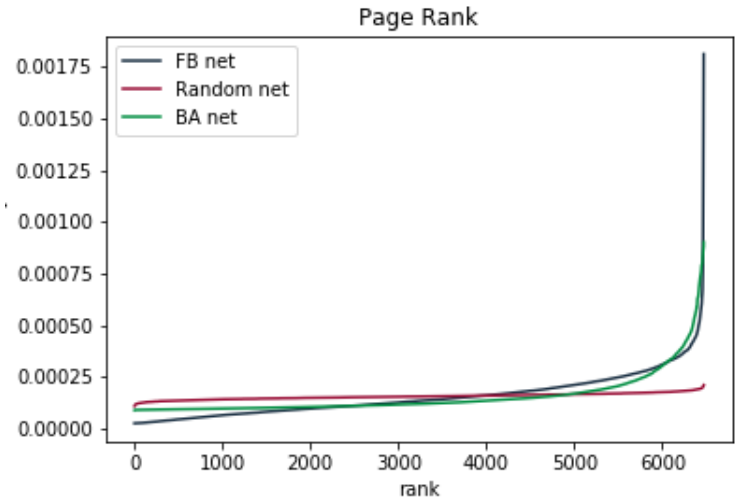

Figure 9: Page rank

The page rank is the probability that a random walker would visit a node. Also in this case, from the plot is clear that all the networks have a high number of nodes having a small value of page rank, that increases more for the real and scale-free network, while the random stays uniform.

3 Community discovery

The aim of Community Discovery algorithms is to identify communities which are hiding behind a complex network structure. Three different algorithms have been applied: DEMON, K-clique, Louvain and label propagation.

The results have then been compared with quality measured obtained using the Partition Quality algorithm.

DEMON

This algorithm exploits the concept of *percolation*, which is used to define a community as a set of nodes grouped together because of the propagation of the same property.

For every node n:
- the *ego network* of n is extracted;
- n is removed from the ego network;
- *Label Propagation* is executed;
- n is inserted in every community that has been found;
- the set of communities C is updated.

Using eps=0.25 and as minimum community size=3, 122 communities have been found.

The values related to the partition quality are the following:

Indexes	min	max	avg	std
Internal Density	0.003473	0.183333	0.034221	0.035993
Average Degree	3.666667	95.695067	42.561722	29.518625
Conductance	0.001657	0.948526	0.605837	0.273117

Modularity (no overlap): 0.157068

K-Clique

K-clique is a percolation algorithm that searches for chains of cliques of size k-1 (where k is a user defined value). A k-clique community is the union of all cliques of size k that can be reached through adjacent (sharing k-1 nodes) k-cliques.

The algorithm produces overlapping partitions, not assuring a complete node coverage.

Louvain

This algorithm is made of two phases. In the first one (modularity optimization) little communities are found by optimizing the measure of modularity. In the second one (community aggregation) the nodes which belong to the same community are aggregated and a new network is created in which the nodes are the communities found in the first phase. These steps are repeated until the maximum modularity is reached. Due to the aggregation logic, the final output will be maed of few but big communities.

The Louvain algorithm has made it possible to find 12 communities. The values related to the partition quality are the following:

Index	min	max	avg	std
Internal Density	0.005347	0.166667	0.063728	0.054321
Average Degree	2.000000	72.449791	38.511600	21.401352
Conductance	0.076923	0.652044	0.377723	0.188813

Modularity (no overlap):	0.427814

Label propagation

Community sets are determined by label propagation. The algorithm produces a complete, non overlapping, node coverage.
The values related to the partition quality are the following:

Indexes	min	max	avg	std
Internal Density	0.003276	0.250000	0.148450	0.084240
Average Degree	1.000000	83.430815	13.471197	24.996935
Conductance	0.000425	0.538462	0.259343	0.169843

Modularity (no overlap)	0.003798

Comparisons of partitions produced by different community discovery algorithms have been executed, for instance Louvain and label propagation. Among the different measures the NF1 score has been used.

Index	Value
Ground Truth Communities	6472.000000
Identified Communities	10.000000
Community Ratio	0.001545
Ground Truth Matched	0.609394
Node Coverage	647.200000
NF1	0.000000

Figure 10: NF summary Louvain, Label propagation

Comparing the quality measures regarding Internal Edge Density, the best communities are obtained with label propagation algorithm.
Taking into account the Average Internal Degree, best communities in term of quality have been obtained with the DEMON algorithm.
For Conductance and Modularity, Louvain and DEMON give the best results.

4 Tie strength

The task of this section is to study the *resilience* of the network by analyzing the impact of strong and weak ties on the connectivity of the network.
In more detail, the edges have been sorted from the strongest to the weakest. Then the 50000 first strongest ties have been removed. The same procedure has been repeated for the 50000 weakest and the results have been compared in order to study the resilience of the network in terms of remaining connected components.
The strength measure used for the edges is the overlapping coefficient:

$$Oij = \frac{|N(i) \cap N(j)|}{|N(i) \cup N(j)|} \tag{3}$$

For each edge:
$\cap$ = number of neighbor nodes to i and j
$\cup$ = number of neighbor nodes to i or j

When the strength of the tie increases, the overlapping coefficient increases as well.

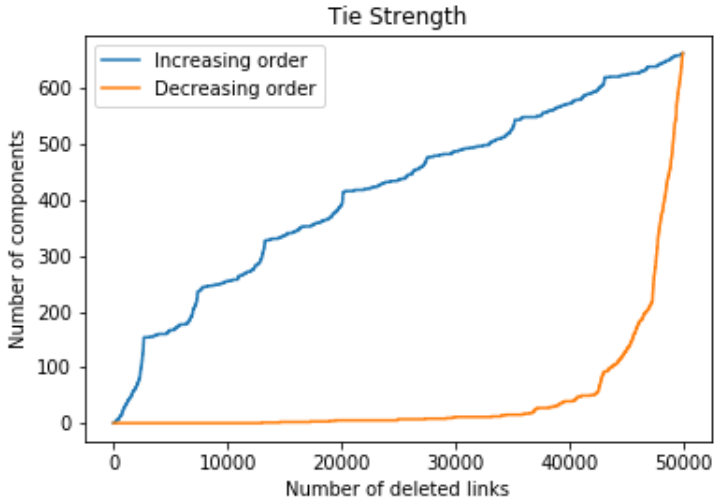

Figure 11: Tie strength

The Figure shows the impact of the strong and weak ties removal on the network.
On the x axis there are the number of removed edges, while on the y axis the number of connected components in which the network is breaking apart.
The blue line represents the edge removal in increasing order in terms of overlapping coefficient (from lowest to the highest). The orange line represents the edge removal in decreasing order, from highest overlapping coefficient to lowest.
It emerges that removing from the weaker to the stronger tie, the network breaks apart faster; after only 20000 edges removed, the network is decomposed in more than 400 components.
Removing the edges from stronger to weaker ties, the network doesn't break apart, but it keeps its condition of unique component.

5 Spreading

The objective of this section is to simulate on the 3 networks (real, random and scale-free) 4 different processes of epidemic modeling.
The chosen models are:

- SI, susceptible-infected
- SIS, susceptible-infected-susceptible
- SIR, susceptible-infected-recovered
- Threshold model

Each model is described in detail.

SI, susceptible-infected

The first model used is characterized by two states: susceptible (S), infected (I) and by and infection rate β. In this model the epidemic rate has an exponential growth due to the fact that all the nodes of the network are considered as susceptible, so all the nodes will be infected.
The following plots have been obtained testing the model with the Python library NDLIB. It is visible that the infected nodes have an exponential growth. At time 0 the whole population is composed only by susceptible individuals. With the time increasing, since every infected individual meets less susceptible peer, the infection curve grows slower. The epidemy stops when the infection has taken the whole population.
In the scale-free network the growth is faster due to the presence of hubs.
$\beta = \mathbf{0{,}001}$

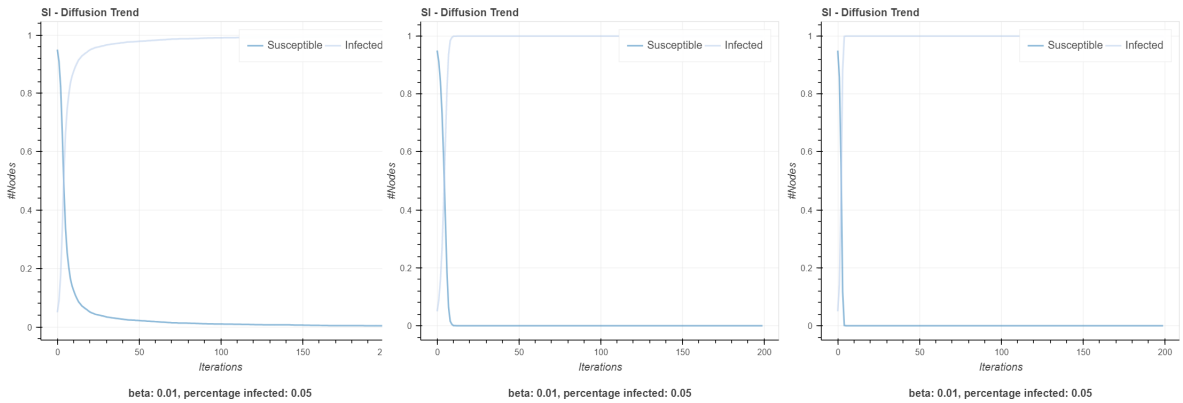

Figure 12: SI model for A-Real Facebook network, B-Random network and C-Scale-free network, β 0,01

If β decreases (from 0,01 to 0,001), then the speed of the infection decreases as well.

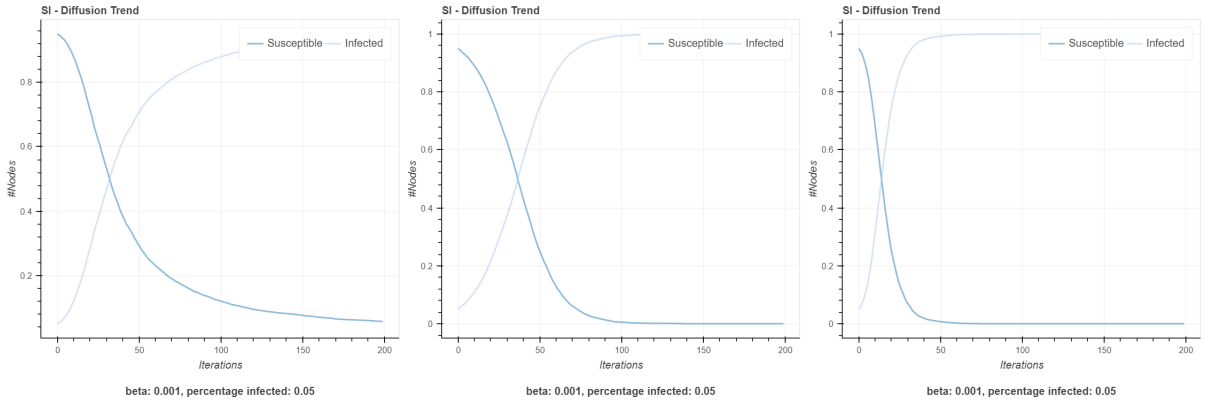

Figure 13: SI model for A-Real Facebook network, B-Random network and C-Scale-free network with β **0,001**

SIS, susceptible-infected-susceptible

The second model is described by the states: susceptible infected (SI), where an infected node can turn become susceptible (S) again.
There's an infection rate β and a recovery rate μ.
With these values the basic reproductive number λ can be calculated, the virus reproductive rate.

$$\lambda = \frac{\beta}{\mu} \tag{4}$$

The basic reproductive rate indicates the average number of nodes infected generated by a single infected node in a completely susceptible society.
In this case, the chosen values of β is 0.001, γ is 0.01 and the *infected percentage* 0.05.

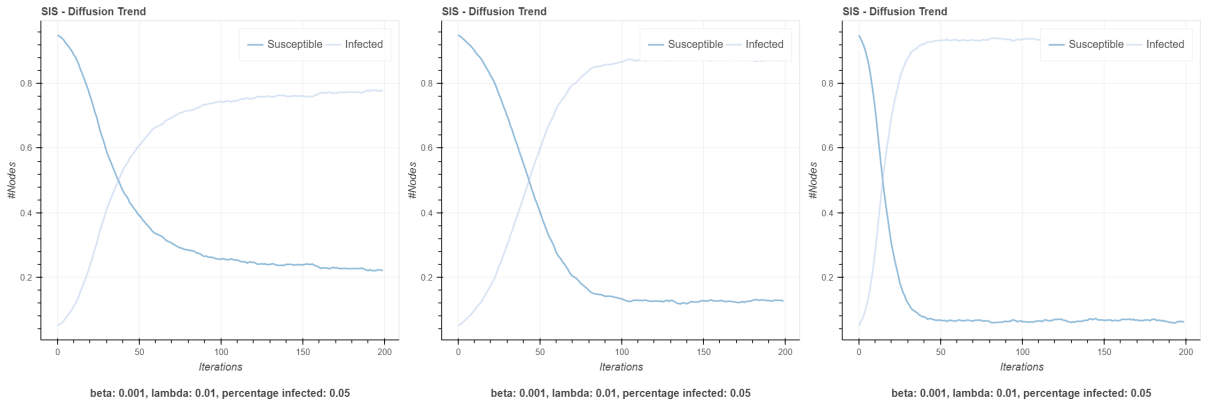

Figure 14: SIS model for A-Real Facebook network, B-Random network and C-Scale-free network

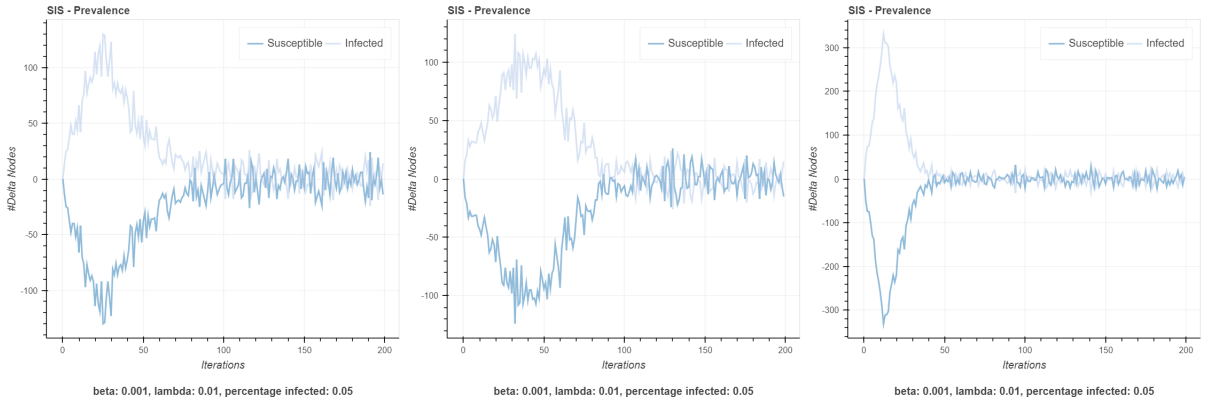

Figure 15: SIS model for A-Real Facebook network, B-Random network and C-Scale-free network

SIR, susceptible-infected-recovered

The third model has three states: susceptible and infected (SI) and recovered (R). The last state is for the individuals who have been in the (S) and (I) states and developed a sort of immunity.
Comparing the plots it can be noticed that the trend of the three curves is similar for the networks.
When the percentage of susceptible nodes decreases, then the percentage of infected nodes increases.
When the infected curve reaches the pick, it starts decreasing, letting the recovery rate curve increase.
It can be noticed that in the scale-free network the process is faster.

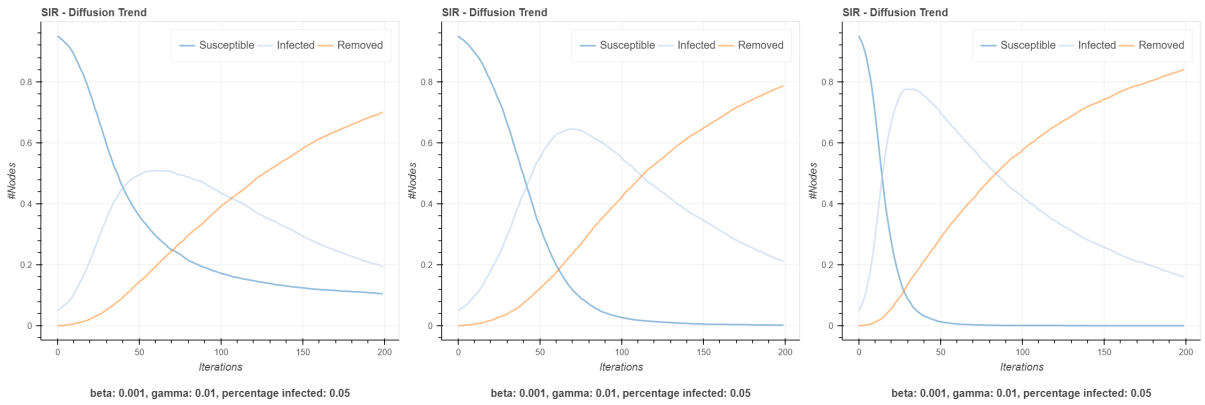

Figure 16: SIR model for A-Real Facebook network, B-Random network and C-Scale-free network

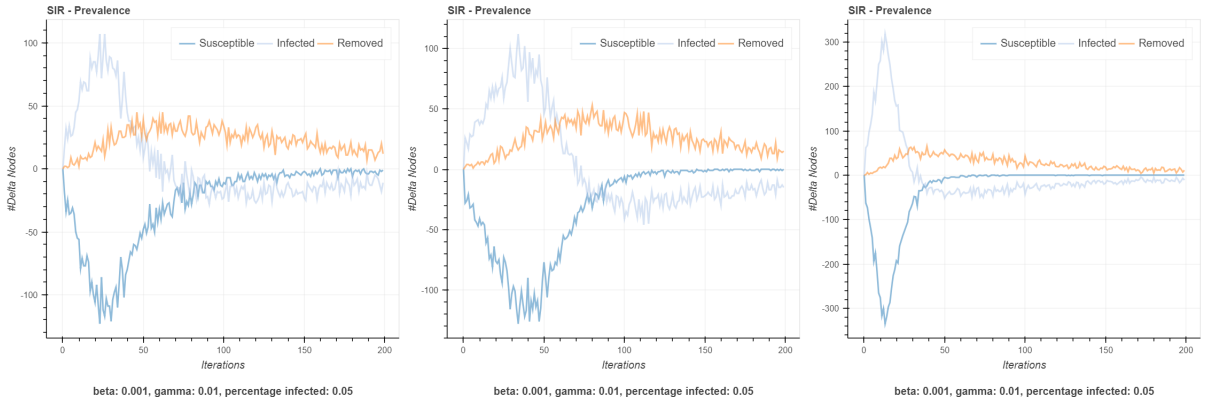

Figure 17: SIR model for A-Real Facebook network, B-Random network and C-Scale-free network

Threshold model

The analysis on epidemic diffusion has been concluded with the application of the threshold model to the networks. In this model each node has a threshold that, if exceeded, makes the node infected.
There's also a percentage of initially infected nodes, that are the beginning of the epidemic diffusion.
The initial nodes infects the neighbors nodes only if the infected neighbors are more than the threshold.
Testing the model on the three networks gives these the results:

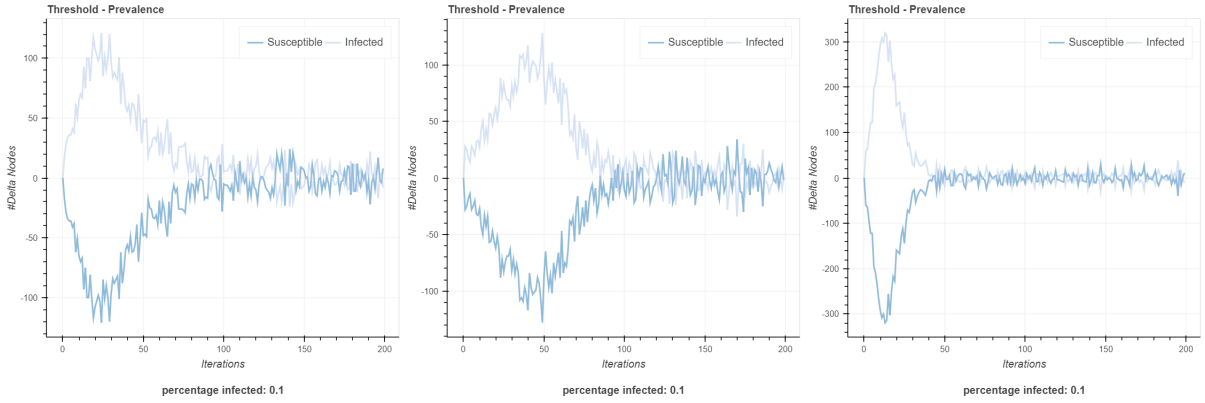

Figure 18: Threshold model for A-Real Facebook network, B-Random network and C-Scale-free network

The % of initial infected nodes is 0.25 and the threshold is 0,1. In the three cases the epidemic diffusion succeed.

6 Link prediction

The link prediction consists in finding a set of edges that, starting from a training set, have a high probability of being inserted in the test set. In particular, for each edge the predictor value, called *score*, gets calculated.
The score indicates the probability that a given edge enters the network in a future interval of time.
The predictors used in this section are Jaccard and SimRank.
Analyzing the two curves in Figure 17 and 18 it can be seen that trends are similar for Jaccard and SimRank measures.

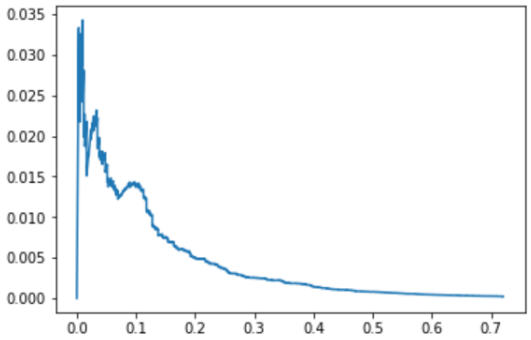

Figure 19: Link prediction, jaccard, x-axis recall, y-axis precision

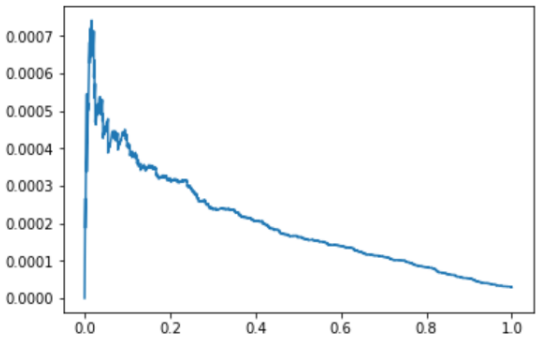

Figure 20: Link prediction, simrank, x-axis recall, y-axis precision

In conclusion, we can say that Jaccard has the best performances on the network, because in the initial phase it has a higher values of precision.