UNIVERSITY OF ZAGREB
FACULTY OF ELECTRICAL ENGINEERING AND COMPUTING

MASTER THESIS No. 1286

Detectability of Patient Zero Depending on its Position in the Network

Iva Miholić

Zagreb, June 2016.

Umjesto ove stranice umetnite izvornik Vašeg rada.

CONTENTS

LIST OF ALGORITHMS

LIST OF TABLES

1. Introduction

A *network* is a set of items with connections between them. The Internet, the World Wide Web, social networks like genealogical trees, networks of friends or co-workers, biological networks like epidemiological networks, networks of citations between papers, distribution systems like postal delivery routes: they all take a form of networks. Most social, biological and technological networks have specific structural properties. Such networks are referred to as *complex networks*. An example of a complex network of scientific collaborations is presented in Figure 1.1.

A network structure or a topology can be mathematically modelled as a graph with set of vertices (or nodes) representing the items of the network. The network structure can then be analysed using graph theory. An edge between two nodes represents a connection between the two corresponding items. Edges can be directed or undirected, depending on the nature of the connection.

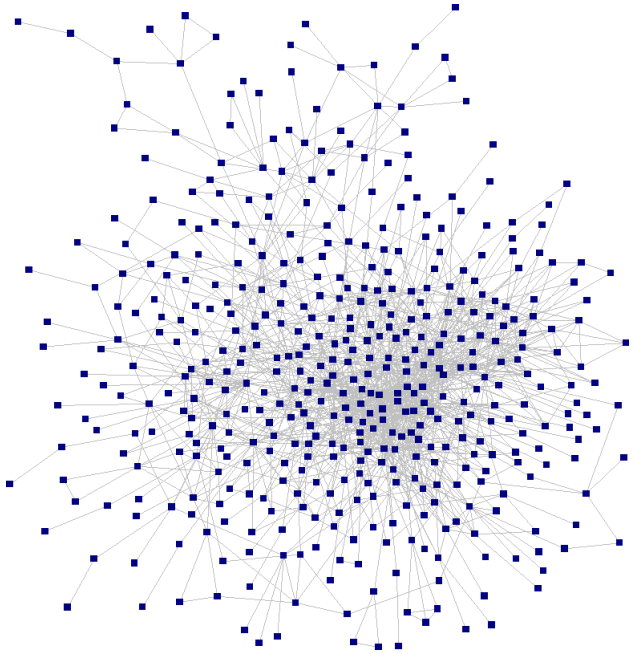

Figure 1.1: A network graph of Paul Erdős and his collaborators, courtesy of Krebs [1]. The nodes represent mathematicians and the edges represent the relationship "wrote a paper with".

To better mimic the real-world (complex) network structure, it is common to add attributes to nodes and/or edges or to have both directed and undirected edges on the same graph.

For large-scaled complex networks that have millions or billions of vertices, the study in the form of traditional graph theory is not sufficient or sometimes possible. When this is the case, the statistical methods for quantifying large complex networks are used.

The ultimate goal of the study of complex network structure is to understand and explain the workings of systems built upon the network such as spreading of disease or information propagation.

After statistical properties analysis, the model of the system or a process is created. The model can help us understand the meaning of statistical properties - how they came to be as they are and how they relate to the behaviour of a networked system. Based on statistical properties and using the right model, the behaviour of networked systems can be determined and predicted.

The basis of the complex network theory, the structure analysis and the process modelling can be found in Newman [2].

1.1. Epidemic processes in complex networks

The models for stochastic processes such as disease spreading are categorized as homogeneous or heterogeneous mixing frameworks. The former assume that all individuals in a population have an equal probability of contact and different equations can be applied to understand epidemic dynamics. Since such models fail to describe the realistic scenario of disease spreading, heterogeneity is introduced by using a network structure.

There is an extremely close relationship between epidemiology and network theory since the connections between individuals (or group of individuals) allowing an infectious disease to propagate naturally define a contact network. Simplest epidemic dynamics consider a system with fixed total population consisting of N individuals modelled with undirected contacting network. We define the contact network as an undirected and non-weighted graph $G(N, L)$ with fixed set of nodes N and fixed set of links L. A link (u, v) between two nodes exists if the two corresponding members were in contact during the epidemic time.

The structure of the network has profound impact on the contagnion dynamics but in order to understand the evolution of the epidemic over time we have to define the

basic individual-level processes that govern the epidemic spreading. Complementary to the network, epidemic modelling describes the dynamical evolution of the contagion process within a population. The state of the art results on epidemic modelling in complex networks can be found in Pastor-Satorras et al. [3].

Classic epidemic models generally assume the network is static during epidemic process while the population can be divided into different classes or compartments depending on the stage of the disease, such as susceptible (those who can contract the infection), infectious (those who contracted the infection and are contagious), recovered, removed or immune. The model defines the basic processes that govern the transition of individuals from one compartment to another. Each member of population can be a part of exactly one compartment at once.

Understanding the structure of the transmission network along with choosing the right epidemic model allows us to predict the distribution of infection and to simulate the full dynamics in order to control disease or plan immunization. In this thesis we will focus on SIR model for epidemic spreading and its modification, the ISS model for modelling rumour diffusion.

1.2. Finding patient zero

The inverse problem of estimating the initial epidemic conditions like localizing the source of an epidemic commonly known as the patient zero problem has only recently been formulated.

In the patient zero problem the source(s) of an epidemic or information diffusion propagation are determined based on limited knowledge of network structure or partial history of the propagation. The survey of methods for identifying the propagation source in networks can be found in Jiang et al. [4].

In the case of the SIR model there are three different approaches. Zhu and Ying [5] proposed a simple path counting approach and prove that the source node minimizes the maximum distance (Jordan centrality) to the infected nodes on infinite trees. Lokhov et al. [6] used a dynamic message-passing algorithm and estimate the probability that a given node produces the observed snapshot using a mean-field approch and an assumption of a tree-like contact network.

Antulov-Fantulin et al. [7] introduce analytical combinatoric, as well as Monte-Carlo based methods for epidemic source detection problem. These methods produce exact and approximate source probability distribution for any network topology based on a snapshot of the epidemic at known discrete time T. The provided benchmark

results show Monte-Carlo based MAP estimators outperform previous results on a lattice network for the SIR model.

Additionally, these methods are applicable to many heterogeneous mixing models (SIR, IS, ISS) and are able to introduce uncertainty in the epidemic starting time, as well as uncertainty of temporal ordering of interactions. Even though the introduced Monte Carlo methods assume the epidemic started from a single source, one can also discriminate such hypothesis using Kolmogorov-Smirnov test [7].

1.3. Effects of network topology on epidemic spreading and detectability of patient zero

Complex networks show various levels of correlation in their topology which can have an impact on dynamical processes running on top of them.

Real-world networks of relevance for epidemic spreading are different from regular lattices. Networks are hierarchically organized with a few nodes that may act as hubs and where the vast majority of nodes have few direct connections.

Although randomness in the connection process of nodes is always present, organizing principles and correlations in the connectivity patterns define network structures that are deeply affecting the evolution and behavior of epidemic and contagion process. These complex features often find their signature in statistical distributions which are generally heavy tailed and skewed.

Antulov-Fantulin et al. [7] have introduced a metric for source detectability based on the entropy of estimated source probability distribution. The detectability of source node differs based on models parameters concerning the rate of disease spreading. Since topological properties of the network have profound impact on epidemic dynamics, the detectability of source node in relation to its topological properties is an interesting analytical problem.

2. Complex network structure

Most of real networks in social and biological systems are characterized by similar topological properties: small average path length, high clustering coefficients, fat tailed scale-free degree distributions and local network structure observable in the presence of communities.

2.1. Measures and metrics

Since larger networks can be difficult to envision and describe only by the graph G, we observe more detailed insights of the structure of these networks with various metrics.

Degree distribution

Degree distribution $P(k)$ defines the probability that a vertex in the network interacts with exactly k other vertices. That is, $P(k)$ is the fraction of nodes in the network with degree equal to k.

Scale-free power-law degree distribution of the form $P(k) = Ak^{-\gamma}$ where $2 < \gamma < 3$ appears in wide variety of complex networks. The networks with such property are referred to as *scale-free networks*. This feature is a consequence of two generic mechanisms: networks expand continuously by the addition of new vertices and new vertices attach preferentially to sites that are already well connected [8]. It is often said the scale-free distributions have "fat tails" since there tends to be many more nodes with higher degree compared to a Poisson degree distribution in a network with links formed completely independently.

Geodesic path

A path in a network is defined as an arbitrary sequence of vertices in which each pair of adjacent vertices is directly connected in the graph.

A geodesic path is the shortest path between two vertices. The small world network property observable in complex networks is considered to be present when average shortest path length is comparable to the logarithm of the network size.

Centrality

Centrality measures compare nodes and say something about how a given node relates to the overall network.

Degree centrality describes how connected a node is in terms of direct connections. For a vertex v in a network with n vertices it is defined as $\frac{deg(v)}{n-1}$. Since the degree centrality captures only centrality in terms of direct connections, it doesn't measure node's marginal contribution to the network when the node has relatively few links but lies in a critical location in the network which can be the case.

Closeness centrality describes how close a given vertex is to any other vertex. Let d_{ij} denote the length of geodesic path from vertex i to vertex j. For vertex v closeness centrality C_v is defined as harmonic mean between the distances of geodesic paths from vertex v to all others:

$$C_v = \frac{1}{n-1} \sum_{j \neq v} \frac{1}{d_{vj}}. \tag{2.1}$$

Betweenness centrality describes how well situated a vertex is in terms of the paths it lies on. Let σ_{st} be the number of geodesic paths between pairs of vertices v_s and v_t and let $\sigma_{st}(v_i)$ be the number of the geodesic paths σ_{st} which pass via vertex v_i. The betweenness centrality is than defined as

$$C(v_i) = \sum_{s,t} \frac{\sigma_{st}(v_i)}{\sigma_{st}}. \tag{2.2}$$

Neighbours characteristics like eigenvector centrality measure how important, central or influential nodes neighbours are and capture a concept the vertex is more important if it has more important neighbours.

Let's define the adjacency matrix A of network G with N nodes as a matrix of size $N \times N$ that contains non-zero element A_{ij} if there exist an edge between vertices i and j. For an unweighed network all non-zero elements of A are equal to one. Note the adjacency matrix is symmetric for undirected graphs and generally asymmetric for directed graphs.

For the given vertex v, **eigenvector centrality** C_v [9] is proportional to the sum of centralities of its neighbours:

$$\lambda C_v = \sum_k A_{vk} C_k. \tag{2.3}$$

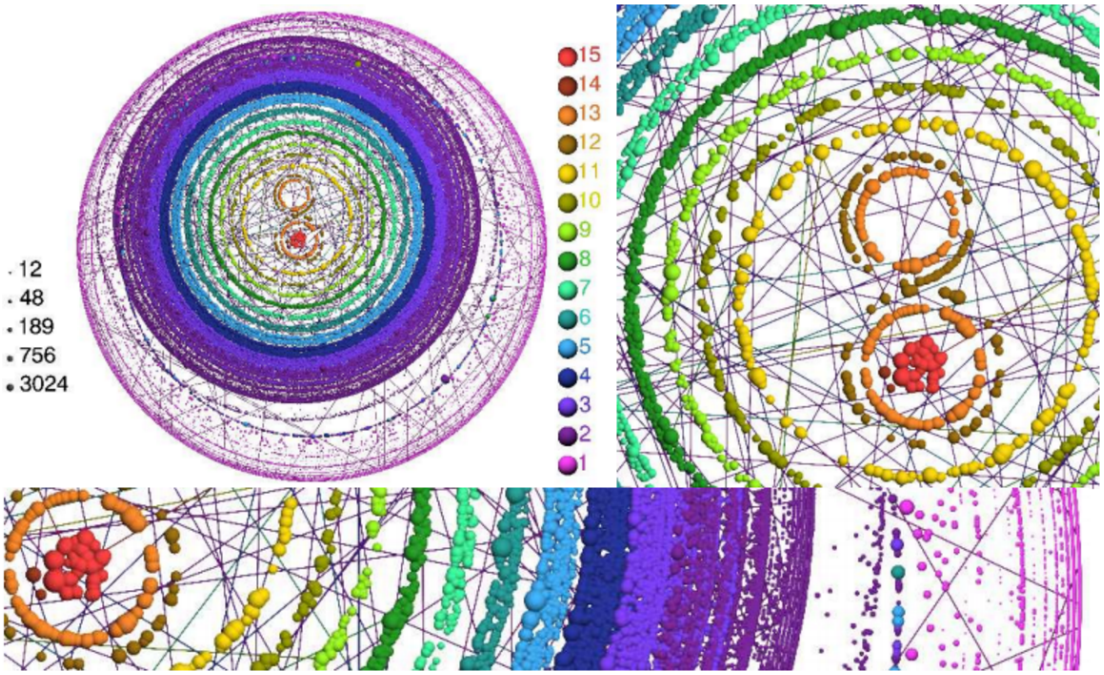

Figure 2.1: Graphical representation of a fraction of the .fr domain of Web, courtesy of Alvarez-Hamelin et al. [10]. The vertices of the same coreness are represented with the same color.

Consequently, C_v is eigenvector of adjacency matrix A corresponding to eigenvalue λ. The standard convention is to use the eigenvector associated with the largest eigenvalue for the eigenvector centrality.

K-core

A k-core of undirected graph G is a maximal connected subgraph of G in which all vertices have degree at least k. The k-core is a measure of how sparse the graph is. Additionally, a vertex u has coreness c if it belongs to a c-core but not to $(c+1)$-core.

The k-core can be obtained in $O(|L|)$ time by iteratively removing all vertices of degree less than k from the graph.

The k-core decomposition refers to a process of determining the coreness of each node and grouping the nodes according to their coreness. The concept of k-core (decomposition) was introduced to study the clustering structure of social networks and to describe the evolution of random graphs. K-core decomposition of complex networks reveals rich k-core architectures as presented in Figure 2.1.

2.2. Modelling global network structure

2.2.1. Erdős Rényi graph model

Traditionally, networks of complex topology have been described with the random graph theory of Erdős and Rényi [11], but in the absence of data on large networks, the predictions of the ER theory were rarely tested in the real world.

This random graph model assumes we start with N vertices and connect each pair of vertices with probability p. The formation is independent across links so the probability of generating a network with exactly m links is equal to $p^m(1-p)^{\frac{N(N-1)}{2}-m}$ and the expected number of links is $\langle d \rangle = pN(N-1)/2$.

The degree distribution of the generated random network is

$$P(d) = \binom{N-1}{d} p^d (1-p)^{N-1-d}. \tag{2.4}$$

For large n, the degree distribution follows a Poisson distribution $P(d) = e^{-\lambda}\lambda^d/d!$, where

$$\lambda = N\binom{N-1}{d} p^d (1-p)^{N-1-d}.$$

Erdős and Rényi [11] have described the behaviour of degree distribution for various values of p. The important result talks about the emergence of one large graph component for higher values of p. In more detail,

- if $Np < 1$ a generated graph will almost surely have no connected components of size larger than $O(\log(N))$,
- if $Np = 1$ a graph will almost surely have a largest component whose size is of order $N^{2/3}$,
- if $Np > 1$ a graph will almost surely have a unique giant component, i.e. no other component will contain more than $O(\log(N))$ vertices.

In Erdős-Rényi model, maximum coreness is related to the average degree $\langle d \rangle$. Since the topology is very homogeneus, it is also expected most vertices will belong to the same k-core that is also the highest.

While random network can observe features like diameters small relative to the network size, they lack certain features that are prevalent among complex networks, such as high clustering and presence of communities.

2.2.2. Barabási-Albert graph model

Barabási-Albert model is the model of evolving a scale-free network which uses a preferential attachment property thus creating a heterogeneous topology. The preferential attachment mechanism is one of two generating mechanisms of scale-free networks [8] and refers to building the network gradually where each new vertex tends to connect to old vertices that are already well connected within the old network.

The Barabási-Albert graph is generated starting from m_0 isolated vertices. At each time step new vertices with m edges are added to the network $m < m_0$. The new vertex will create an edge to the existing node v_i with probability proportional to its degree k_i.

The Barabási-Albert graph model produces a power law degree distribution $P(k) \approx k^{-3}$ in the limit of growth time, i.e. number of vertices. The average geodesic path increases logarithmically with the size of the network.

By repetitively connecting each new node to the previous graph with exactly m edges, we obtain a graph where any subgraph has a vertex of degree at most m and the k-core of the graph is m.

3. Epidemic process modelling

In the focus of this thesis are heterogeneous epidemic models on the contact network formed by connections between single contacting individuals with transitions of individuals between compartments happening in discrete time steps.

3.1. SIR model

Wide range of diseases that provide immunity to the host can be successfully modelled on a network whose members take one of three possible roles at a time: susceptible (S), infected (I) or recovered (R) [12].

The diffusion of disease takes place between infected nodes and their susceptible neighbours. An infectious node may also recover from the disease. The recovery grants permanent immunity effectively erasing the member from the contacting network. The possible events can be represented as

$$S + I \xrightarrow{p} 2I, \quad I \xrightarrow{q} R. \tag{3.1}$$

In the SIR model infection and recovery process completely determine the epidemic evolution. The transitions (3.1) occur spontaneously and independently in each time step. In discrete-time formulation an infected individual when meeting susceptible will infect the neighbouring susceptible with probability p at each time step. The recovery probability q is the probability the infected individual will recover for each time step.

The transition probabilities p and q are often assumed constant and equal for all nodes in the same epidemic process.

3.1.1. Simulating the discrete SIR epidemic

For the contacting network represented by graph $G(N, L)$ and SIR parameters p and q, we are able to simulate one time step of discrete SIR process. Let s_t, i_t and r_t denote sets of nodes that are respectively susceptible, infected and recovered after time step t.

At time step t all previously infected nodes i_{t-1} will try to infect their susceptible neighbours independently of each other and at the same time. Afterwards the passive recovery process will try to turn them to recovering nodes, each with probability q.

This process can be simulated with NaiveSIR algorithm [13] by putting all the initially infected nodes in the queue. While traversing the nodes, we try to infect each neighbouring node. When the new node gets infected, it gets pushed to the queue.

SIR simulation of one time step t is described in Algorithm 1.

Algorithm 1: One time step of NaiveSIR simulation on graph $\mathbf{G}$.

Data: $\mathbf{G}$ - network, (p, q) -parameters of the SIR model, Iq - queue of infected nodes, I - bitset of infected nodes, S - bitset of susceptible nodes, R - bitset of recovered nodes

```
infected_size = size(Iq)
for k = 1 to infected_size do
    if Iq is empty then
        break
    dequeue(u, Iq)
    foreach v ∈ nei(u) do
        if v ∈ S then
            let transmission u → v occur with probability p
            if u → v occured then
                update I(v) and S(v)
    let transmission u → v occur with probability q
    if u → v occured then
        update I(u) and R(u)
    else
        enqueue(u, Iq)
return {S, I, R}
```

Time and space complexity of NaiveSIR algorithm

For algorithm complexity analysis standard big-O notation is used (asymptotic upper bound within a constant factor) [14]. In a single SIR step, simulation tries to infect all neighbours of infected nodes that are susceptible, i.e. $O(\langle d \rangle)$ nodes where $\langle d \rangle$ denotes the average node degree. Since after each SIR step each infected node is recovered with probability q, the average number of time steps the node spends in infected state is a

sample from geometric distribution $P(\Delta T = \Delta t) = (1-q)^{\Delta t - 1} q$ with expectation $\frac{1}{q}$. Total time complexity for one infected node is thus $O(\frac{\langle d \rangle}{q})$. Finally, the average case running time of the NaiveSIR algorithm is equal to $O(\frac{E[X]\langle d \rangle}{q})$ where $E[X]$ denotes total expected number of infected nodes [13].

The space complexity of NaiveSIR algorithm with respect to the number of links L of the graph G is equal to $O(L)$ since the memory holds a contact network G in a form of adjacency list ($O(L)$), queue of infected nodes ($O(N)$) and indicators of each compartment that are best implemented as a bitset data structure ($O(1)$).

3.1.2. Probability of compartment transitions in one time step of SIR simulation

Probability of compartment transitions in one time step of SIR simulation can be easily evaluated. Let $nei(v)$ indicate a set of all neighbours of node v, $nei(V)$ a set of all neighbours of all nodes in set V and $nei_V(v) = nei(v) \cap V$ a set of all neighbours of v that are also in V. After k-th time step of the SIR process the resulting i_k and r_k were given. At time step k, only initially active nodes i_{k-1} and their neighbours $nei(i_{k-1})$ actively participate in the epidemic process. For each node v in $i_{k-1} \cup nei(i_{k-1})$, one of four independent events may happen during time step k and they are easily detectable based on i_{k-1}, r_{k-1}, i_k and r_k:

- E_1 : **if** $v \notin i_{k-1}$ and $v \notin r_{k-1}$ and $v \in i_k$
  node v was infected with probability $1 - (1-p)^{nei_{i_{k-1}}(v)}$
- E_2 : **if** $v \notin i_{k-1}$ and $v \notin r_{k-1}$ and $v \notin i_k$
  node v was not infected with probability $(1-p)^{nei_{i_{k-1}}(v)}$
- E_3 : **if** $v \in i_{k-1}$ and $v \in r_k$
  node v was recovered with probability q
- E_4 : **if** $v \in i_{k-1}$ and $v \notin r_k$
  node v was not recovered with probability $1 - q$

Since all events $E_1 - E_4$ are independent and sets of nodes corresponding to each event are disjoint while completely covering the set of active nodes $i_{k-1} \cup nei(i_{k-1})$, the conditional probability of one time step SIR transition $P(i_k, r_k | i_{k-1}, r_{k-1})$ can be calculated as

$$\begin{aligned} P(i_k, r_k | i_{k-1}, r_{k-1}) = &\left[\Pi_{v \in E_1}(1 - (1-p)^{nei_{i_{k-1}}(v)})\right] \left[\Pi_{v \in E_2}(1-p)^{nei_{i_{k-1}}(v)}\right] \\ &\cdot \left[\Pi_{v \in E_3} q\right] \left[\Pi_{v \in E_4}(1-q)\right]. \end{aligned} \tag{3.2}$$

Set $nei_{i_{k-1}}(v)$ denotes the set of all neighbours of v that were infected at the beginning of time step k, i.e. the set $nei(v) \cap i_{k-1}$.

3.2. Epidemic models as social contagion processes

Even though infectious diseases represent the central focus of epidemic modelling, the model where an individual is strongly influenced by the interaction with its peers is present in several other domains, especially in social context in the diffusion of information, the propagation of rumour and adoption of innovation or behaviours. Since the social contacts can in these domains generate epidemic-like outbreaks, simple models for information diffusion are epidemic models modified to specific features of social contagion. The crucial difference to pathogen spreading is that transmission of information involves intentional acts by both the sender and the receiver and it is often beneficial for both participants.

3.2.1. Rumour spreading with ISS model

The need to study rumour spreading presents itself in a number of important technological and commercial applications where it is desirable to spread the "epidemic" as fast and as efficient as possible. In examples such as rumour based protocols for resource discovery and marketing campaigns that use rumour like strategies (viral marketing) the problem translates to design of an epidemic algorithm in such a way that the given information reaches as much nodes as possible, similarly to a rumour.

Models for rumour spreading are variants of the SIR model in which the recovery process does not occur spontaneously, but rather is a consequence of interactions. The modification mimics the idea it is worth spreading the rumour as long as it is novel for the recipient.

This process can be formalized as a model where each of N members of the contacting network can be a part of one of three compartments: ignorant (S), spreader (I) and stifler (R). Ignorants have not heard the rumour and are susceptible to being informed. Spreaders are actively spreading the rumour, while stiflers know about the rumour but they're not spreading it.

The spreading process evolves by direct contacts of spreaders with others in the population. When a spreader meets an ignorant, the latter turns into a new spreader with probability a. When a spreader meets another spreader or a stifler, the former spreader turns into stifler with probability b and the latter remains unchanged. This

model is known as ISS (Ignorant-Spreader-Stifler) model [15]. The possible events can be represented as

$$S + I \xrightarrow{a} 2I, \quad R + I \xrightarrow{b} 2R, \quad 2I \xrightarrow{b} R + I. \tag{3.3}$$

Since we are examining the spreading process in discrete time, at each time step the current spreaders try to interact with their neighbours. A modification of the NaiveSIR algorithm for rumour spreading simulation of one time step t is described by Algorithm 2.

Algorithm 2: One time step of ISS simulation with modified NaiveSIR algorithm on graph $\mathbf{G}$.

Data: $\mathbf{G}$ - network, (a, b) - parameters of the ISS model, Iq - priority queue of spreader nodes, I - bitset of spreader nodes, S - bitset of ignorant nodes, R - bitset of stifler nodes

```
stifler_size = size(Iq)
for k = 1 to stifler_size do
    if Iq is empty then
        break
    dequeue(u, Iq)
    foreach v ∈ nei(u) do
        if v ∈ S then
            let transmission u → v occur with probability a
            if u → v occured then
                update I(v) and S(v)
        else
            let transmission v → u occur with probability b
            if v → u occured then
                update I(u) and R(u)
    if u ∈ I then
        enqueue(u, Iq)
return {S, I, R}
```

4. Patient zero – single source epidemic detection

In accordance with Antulov-Fantulin et al. [7], we will focus on a patient zero problem given snapshot of population at time T and complete knowledge of underlying contacting network modelled by graph $G(N, L)$ with assumption the epidemic has started from a single source node and that it is governed by the SIR process with known p and q.

The estimators proposed by Antulov-Fantulin et al. [7] will be presented in this chapter, while the newly proposed estimators based on importance sampling technique will be presented in the next chapter.

4.1. Problem definition

Let random vector $\vec{S} = (S(1), \ldots, S(N))$ indicate the nodes that got infected up to a predefined temporal threshold T with SIR(p, q) epidemic process on network G with N nodes. $S(i)$ is a Bernoulli random variable with the value 1 if the node i got infected before time T from the start of the epidemic process.

We observe one realization $\vec{s}_*$ of $\vec{S}$ – the epidemic snapshot at time T and want to infer which nodes from the set of infected or recovered nodes $\Theta = \{\theta_1, \theta_2, \ldots, \theta_m\}$ are most likely to be the source of observed epidemic process. The finite set of possible source nodes Θ is determined by realization $\vec{s}_*$.

A maximum aposteriori probability estimate (MAP) is the node with the highest probability for being the source of the epidemic spread for given target realization $\vec{s}_*$:

$$\hat{\theta}_{MAP} = \text{arg max}_{\theta_i \in \Theta} P(\Theta = \theta_i | \vec{S} = \vec{s}_*) \tag{4.1}$$

By applying the Bayes theorem with equal apriori probabilities $P(\Theta = \theta_i)$, probability

in (4.1) can be expressed as

$$P(\Theta = \theta_i|\vec{S} = \vec{s}_*) = \frac{P(\vec{S} = \vec{s}_*|\Theta = \theta_i)P(\Theta = \theta_i)}{\sum_{\theta_k \in \Theta} P(\vec{S} = \vec{s}_*|\Theta = \theta_k)P(\Theta = \theta_k)} = \frac{P(\vec{S} = \vec{s}_*|\Theta = \theta_i)}{\sum_{\theta_k \in \Theta} P(\vec{S} = \vec{s}_*|\Theta = \theta_k)}. \tag{4.2}$$

4.2. Direct Monte Carlo epidemic source detector

The integration problem

$$\mathbf{E_f}[h(X)] = \int_X h(x)f(x)dx \tag{4.3}$$

of computing the expectation of the function $h(X) : X \to \mathbb{R}$ of random variable X with density $f(X)$ can be estimated using Monte Carlo technique with n samples $X_1, \ldots, X_n$ generated from density f as the empirical average

$$h_n = \frac{1}{n}\sum_{j=1}^{n} h(X_j). \tag{4.4}$$

The convergence of h_n towards $\mathbf{E_f}[h(X)]$ is assured by the Strong Law of Large Numbers.

Inferring the probability $P(\vec{S} = \vec{s}_*|\Theta = \theta_i)$ up to multiplicative constant is an integration problem equivalent to expectation of Kronecker delta function $\delta(\vec{S}) = 1\{\vec{S} = \vec{s}_*\}$ where $\vec{S}$ is a random variable governed by probability distribution $P(\vec{S}|\Theta = \theta_i)$.

Let m_i denote number of realizations out of n that completely correspond to $\vec{s}_*$ for a fixed source θ_i estimated using Monte Carlo technique:

$$m_i = \sum_{j=1}^{n} 1\{\vec{S}_i = s_*\} \tag{4.5}$$

where $\vec{S}_i$ are drawn from $P(\vec{S}|\Theta = \theta_i)$.

The estimate m_i is obtained using Direct Monte Carlo technique by simulating epidemic process up to time T starting from a single infected node θ_i and checking whether the generated realization of $\vec{S}_i$ coincides with $\vec{s}_*$. Since m_i is estimation of $P(\vec{S} = \vec{s}_*|\Theta = \theta_i)$ up to multiplicative constant $1/n$ for all $\theta_i \in \Theta$, we derive Direct Monte Carlo MAP detector based on the estimation of probability $P(\Theta = \theta_i|\vec{S} = \vec{s}_*)$ by combining (4.5) with Bayes rule (4.2):

$$\hat{P}_i^{\,n} = \hat{P}(\Theta = \theta_i|\vec{S} = \vec{s}_*) = \frac{m_i}{m} \tag{4.6}$$

where $m = \sum_{j=1}^{n} m_j$.

The accuracy of Direct Monte Carlo estimation is controlled by convergence conditions. Upon estimating two source PDF's $\hat{P}_i^n$ and $\hat{P}_i^{2n}$ with n and $2n$ independent simulations respectively, the estimated distribution is said to converge when the following conditions are satisfied:

$$|\hat{P}_{MAP}^{2n} - \hat{P}_{MAP}^{n}|/\hat{P}_{MAP}^{2n} \leq c \text{ and } |\hat{P}_i^{2n} - \hat{P}_i^{n}| \leq c \quad \forall \theta_i \in \Theta. \tag{4.7}$$

The term $\hat{P}_{MAP}$ corresponds to MAP probability of estimated distribution $\hat{P}$.

If the size of realization $\vec{s}_*$ is big, the number of simulations required to obtain reliable estimations can be large. This makes it is crucial to optimise the simulation procedure.

Since the estimations for different source node candidates are independent, the computations can be parallelised.

Additionally, a prunning mechanism can be incorporated. If a simulation infects a node that was not infected during the target epidemic represented by realization $\vec{s}_*$, it is safe to stop the simulation prior to ending time T and call the partial sample unequal to target realization $\vec{s}_*$.

Algorithm 3: Direct Monte Carlo estimation of number of realizations out of n simulations completely corresponding to $\vec{s}_*$ after T time steps for a fixed source node θ_i.

Data: $\mathbf{G}$ - network, (p, q) - parameters of the SIR process, $\vec{s}_*$ - target realization, T - temporal threshold, θ_i - proposed source node, n - number of simulations

$m_i = 0$

for $d = 1$ *to* n **do**

  for $t = 1$ *to* T **do**

    Continue SIR simulation (d, p, q, θ_i) for time step t and obtain $\vec{S}_t^{(d)}$

    if $\exists j \in N : (S_t^{(d)}(j) == 1$ ***and*** $s_*(j) == 0)$ **then**

      break

  if $\vec{S}_T^{(d)}$ ***equals*** $\vec{s}_*$ **then**

    $m_i = m_i + 1$

return m_i

4.3. Soft Margin epidemic source detector

Let $\vec{S}_\theta^{(j)}$ denote j-th sample (outcome) obtained by Monte Carlo simulation of contagion process with source node θ and duration of T time steps. $\vec{S}_\theta^{(j)}$ is one realization of random binary vector $\vec{S}_\theta$ that describes the outcome of an epidemic process. A similarity measure $\varphi : (\vec{S}_\theta \times \vec{S}_\theta) \to [0, 1]$ can be defined between any two realizations of $\vec{S}_\theta$. For example, φ can be defined as the Jaccard similarity function:

$$\varphi(\vec{S}_1, \vec{S}_2) = \frac{\vec{S}_1 \cap \vec{S}_2}{\vec{S}_1 \cup \vec{S}_2} = \frac{\sum_{j=1}^{N}(S_1(j) = 1 \text{ and } S_2(j) = 1)}{\sum_{j=1}^{N}(S_1(j) = 1 \text{ or } S_2(j) = 1)}. \tag{4.8}$$

Moreover, we can define a discrete random variable $\varphi(\vec{s}_*, \vec{S}_\theta)$ that measures the similarity between fixed realization $\vec{s}_*$ and random realization from $\vec{S}_\theta$. Let PDF of that random variable be $f_\theta(x)$ where $x = \varphi(\vec{s}_*, \vec{S}_\theta)$. Since $\varphi(\vec{s}_*, \vec{S})$ takes discrete values, the probability density function is an integral of a range of Dirac delta functions, each positioned at one value of $\varphi(\vec{s}_*, \vec{S})$ and weighted by corresponding probability.

By using Monte Carlo method we can take PDF definition as an integration problem (4.3) and sample from this discrete distribution $\{p_1, \ldots, p_d\}$ of $\varphi(\vec{s}_*, \vec{S})$ to obtain the PDF estimate:

$$f_\theta(x) = \int_0^1 p_k \delta(x - x_k)dx \approx \frac{1}{n}\sum_{i=1}^{n} \delta(x - \varphi(\vec{s}_*, \vec{S}_\theta^{(i)})) \tag{4.9}$$

where $\delta(x)$ denotes the Dirac delta function.

4.3.1. Soft Margin estimator

The Soft Margin estimator is defined as

$$\hat{P}_a(\vec{S} = \vec{s}_*|\Theta = \theta) = \int_0^1 w_a(x)\hat{f}_\theta(x)dx \tag{4.10}$$

where $w_a(x)$ is a weighting function and $\hat{f}_\theta(x)$ is the estimated PDF of the random variable $\varphi(\vec{s}_*, \vec{S}_\theta)$. For $w_a(x)$ Antulov-Fantulin et al. [7] proposed a Gaussian weighting form $w_a(x) = e^{-(x-1)^2/a^2}$.

With Soft Margin approximation the problem definition is altered to estimating the number of realizations with similarity to $\vec{s}_*$ in the interval around $\varphi = 1$ defined by Gaussian weighting function $w_a(x)$, as opposed to estimating the number of realizations with similarity strictly equal to $\varphi = 1$ like with Direct Monte Carlo method.

Appropriate values of parameter a can be deduced from contour plot in Figure 4.1. For a close to 1, Soft Margin approximation includes more asimilar samples. In

the limit where $a \to 0$ Soft Margin approximation converges to Direct Monte Carlo estimate.

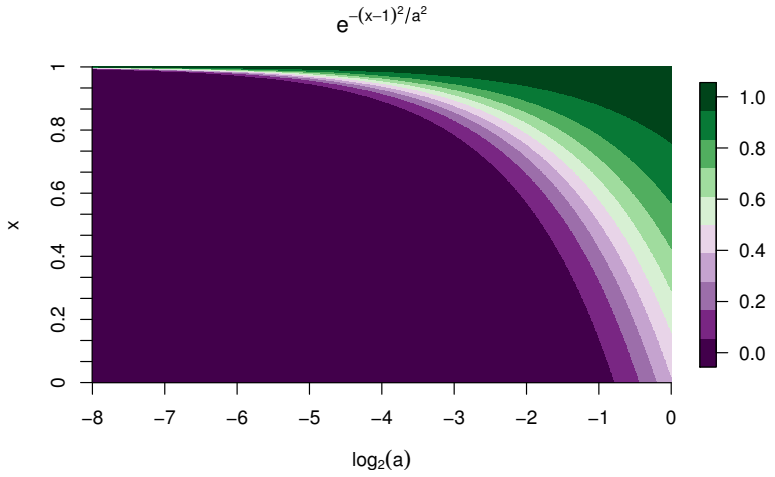

Figure 4.1: Contour plot of Gaussian weighting function $w_a(x) = e^{-(x-1)^2/a^2}$.

The Soft Margin formula 4.10 can be further simplified by combining with 4.9:

$$\begin{aligned} \hat{P}_a(\vec{S} = \vec{s}_*|\Theta = \theta) &= \int_0^1 w_a(x)\hat{f}_\theta(x)dx \\ &= \int_0^1 w_a(x)\frac{1}{n}\sum_{i=1}^{n}\delta(x - \varphi(\vec{s}_*, \vec{S}_\theta^{(i)}))dx, \end{aligned} \tag{4.11}$$

and further by using the property of Dirac delta function $\int_{-\infty}^{\infty} f(x)\delta(x-b)dx = f(b)$:

$$\begin{aligned} \hat{P}_a(\vec{S} = \vec{s}_*|\Theta = \theta) &= \frac{1}{n}\sum_{i=1}^{n}\int_0^1 w_a(x)\delta(x - \varphi(\vec{s}_*, \vec{S}_\theta^{(i)}))dx \\ &= \frac{1}{n}\sum_{i=1}^{n} w_a(\varphi(\vec{s}_*, \vec{S}_\theta^{(i)})) \\ &= \frac{1}{n}\sum_{i=1}^{n} e^{\frac{(\varphi_i - 1)^2}{a^2}}. \end{aligned} \tag{4.12}$$

Since our final goal is estimation of probability distribution $P(\Theta = \theta_i|\vec{S} = \vec{s}_*)$, for numerical reasons it is wise to use likelihood $n\hat{P}(\Theta = \theta_i\vec{S} = \vec{s}_*)$ in the calculation of 4.2 instead of the estimated probability $\hat{P}(\Theta = \theta_i|\vec{S} = \vec{s}_*)$ when the number of simulations n used to estimate $\hat{P}$ is the same for all potential source nodes.

Note that it's not needed to determine constant a in advance. The parameter a can be chosen as the infinum of the set of parameters for which the source probability distribution estimate $\hat{P}_a(\Theta = \theta_i|\vec{S} = \vec{s}_*)$ has converged under the convergence property 4.7.

Additionally, for a fixed number of simulations n, PDF's based on different parameters a can be estimated with one set of samples by evaluating 4.12 for different values of parameter a.

Algorithm 4: Soft Margin approximation of $P(\vec{S} = \vec{s}_*|\Theta = \theta_i)$ for a fixed source node θ_i.

Data: $\mathbf{G}$ - network, (p, q) - parameters of the SIR process, $\vec{s}_*$ - target realization, T - temporal threshold, θ_i - proposed source node, n - number of simulations, a - Soft Margin parameter

for $i = 1$ *to* n **do**
    Run SIR simulation (p, q, θ_i) for T time steps and obtain $\vec{S}_T^{(i)}$
    Calculate and save $\varphi_i = \varphi(\vec{s}_*, \vec{S}_T^{(i)})$
Calculate $\hat{P}(\vec{S} = \vec{s}_*|\Theta = \theta_i) = \frac{1}{n}\sum_{i=1}^{n} e^{\frac{-(\varphi_i - 1)^2}{a^2}}$
return $\hat{P}(\vec{S} = \vec{s}_*|\Theta = \theta_i)$

4.4. Time complexity of Direct Monte Carlo and Soft Margin epidemic source detectors

The average run time complexity of Monte Carlo epidemic source detectors Direct Monte Carlo and Soft Margin is $mn\overline{RT}_M$, where m denotes the number of potential sources in the observed realization, n the number of samples of the random variable $\vec{S}_\theta$ and $\overline{RT}_M$ denotes the average run-time complexity of sampling one realization from contagion process M [7].

Note that in the worst-case scenario the number of potential sources is proportional to the network size, but in reality we are mostly interested in epidemic source detection problems in which the number of potential sources is much smaller than the network size.

Additionally, different Monte Carlo estimators have different convergence properties with respect to number of samples n. Under convergence conditions 4.7 and $c = 0.05$ the Soft Margin estimator converges for $n \in [10^4, 10^6]$ on the benchmark lattice dataset on which Direct Monte Carlo requires $n \in [10^6, 10^8]$ simulations for each potential source node, as presented in Figure 6.7.

5. Importance sampling based epidemic single source detection

5.1. Importance sampling

Importance sampling is a technique for estimating properties of a particular distribution with samples generated from a different distribution than the one of interest. The technique is used with Monte Carlo method as an estimator variance reduction technique since the new sampling distribution of choice is usually biased towards realizations that have more impact on the parameters being estimated.

Suppose we want to estimate area under $f(x)$ plotted in Figure 5.1. With Monte Carlo technique we sample uniformly at random from x and add each sampled value $f(x)$ to the estimate. The importance of each sample in the estimate depends on the value of the function in that point. To gain better estimate with fewer number of samples one might want to sample x from a density similar to $f(x)$. By sampling from $g(x)$ to estimate area under $f(x)$, values of x that are more included in the estimation will be sampled more frequently.

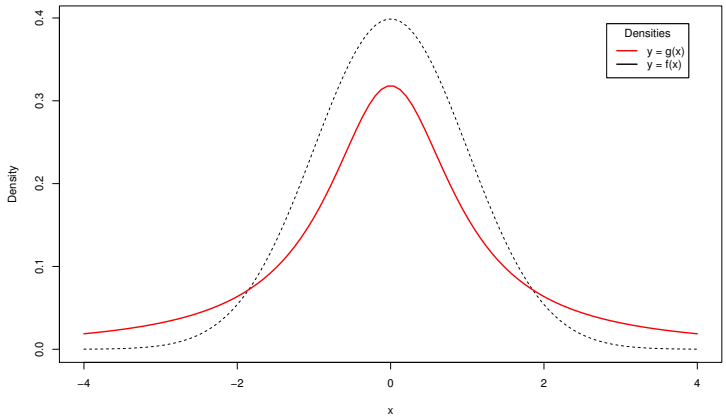

Figure 5.1: Target density $y = f(x)$ and biased importance density $y = g(x)$.

The method of importance sampling is estimation of the integration problem (4.3) based on generating a sample $X^{(1)}, \ldots, X^{(n)}$ from a given biased distribution g when in fact the samples $X^{(i)}$ come from the target distribution f:

$$\mathbf{E_f}[h(X)] = \int_X h(x)f(x)dx = \int_X h(x)\frac{f(x)}{g(x)}g(x)dx \approx \frac{1}{n}\sum_{j=1}^{n}\frac{f(X^{(j)})}{g(X^{(j)})}h(X^{(j)}). \quad (5.1)$$

By choosing to sample from the biased distribution g, we are left with the extra weight $w^{(j)} = \frac{f(X^{(j)})}{g(X^{(j)})}$ from the integral. The weight $w^{(j)}$ corrects the bias of the sampling procedure.

The new estimator converges whatever the choice of distribution g, as long as $supp(g) \supset supp(f)$[1], i.e. for each generated sample the weight $w^{(j)}$ has to be finite [16].

Note the estimation can be done with unbiased estimate

$$\frac{1}{n}\sum_{i=1}^{n} w^{(i)}h(X^{(i)}), \quad (5.2)$$

or with a weighted estimate

$$\frac{\sum_{i=1}^{n} w^{(i)}h(X^{(i)})}{\sum_{i=1}^{n} w^{(i)}}. \quad (5.3)$$

When using the weighted estimate, we only need to know the ratio $f(x)/g(x)$ up to a multiplicative constant. Although inducing a small bias, the weighted estimate often has a smaller mean squared error than the unbiased one [16].

5.1.1. Measuring the quality of importance distribution

By properly choosing $g(\cdot)$ one can reduce the estimator variance substantially. In order to make the estimation error small, one wants to choose $g(x)$ as close in shape to $f(x)h(x)$ as possible. The efficiency of importance distribution is difficult to measure.

Effective sample size (ESS) is commonly used to measure how different the importance distribution is from the target distribution. ESS will give the size of an iid sample set with the same variance as the current sample set. Suppose we have n independent samples generated from $g(\mathbf{x})$. The ESS is then defined as

$$\text{ESS}(n) = \frac{n}{1 + var_g[w(x)]}. \quad (5.4)$$

The variance here is estimated as a square of the coefficient of variation of the weights:

$$cv^2 = \frac{\sum_{j=1}^{n}(w^{(j)} - \bar{w})^2}{(n-1)\bar{w}^2} \quad (5.5)$$

[1] $supp(g) = \{x|g(x) \neq 0\}$

where $\bar{w}$ is sample average of the weights $w^{(j)}$. Effective sample as a measure of efficiency can be partially justified by the delta method [17].

5.2. Sequential importance sampling

Since it is not trivial to design a good importance sampling density, especially for high dimensional problems, one may build up the importance density sequentially [17]. Suppose we can decompose x as $\mathbf{x} = (x_1, \ldots, x_d)$ where each of the x_j may be multi-dimensional. That is especially helpful when the state space of x_{t+1} is an augmentation of state x_t. The importance distribution can then be constructed as

$$g(\mathbf{x}) = g_1(x_1)g_2(x_2|x_1)g_3(x_3|x_1, x_2)\ldots g_d(x_d|x_1, \ldots, x_{d-1})$$

With recursive form we hope to obtain some guidance from the target density while building up the importance density. We can then rewrite the target density as

$$f(\mathbf{x}) = f_1(x_1)f_2(x_2|x_1)f_3(x_3|x_1, x_2)\ldots f_d(x_d|x_1, \ldots, x_{d-1})$$

and the weights as

$$w(\mathbf{x}) = \frac{f_1(x_1)f_2(x_2|x_1)f_3(x_3|x_1, x_2)\ldots f_d(x_d|x_1, \ldots, x_{d-1})}{g_1(x_1)g_2(x_2|x_1)g_3(x_3|x_1, x_2)\ldots g_d(x_d|x_1, \ldots, x_{d-1})} \tag{5.6}$$

which suggests a recursive monitoring and computing of importance weight

$$w_t(\mathbf{x}_t) = w_{t-1}(\mathbf{x}_{t-1})\frac{f(x_t|\mathbf{x}_{t-1})}{g(x_t|\mathbf{x}_{t-1})}. \tag{5.7}$$

In other words, we build samples and importance weights sequentially. Each partial sample $\mathbf{x}_{t-1}$ is extended using generator based on transitional importance density $g(x_t|\mathbf{x}_{t-1})$. For the generated sample $x_1, \ldots x_t$ the target transitional density $f(x_t|\mathbf{x}_{t-1})$ is calculated. Final importance weight $w_d(\mathbf{x}_t)$ can be calculated using the series of transitional target and importance densities for the particular sample. At the end, $w_d(\mathbf{x}_d)$ is equal to $w(\mathbf{x})$.

By using the recursive process we can stop generating further components of $\mathbf{x}$ if the partial weight derived from the sequentially generated partial samples is too small in relation to other weights and we can take advantage of current set of samples x_{t-1} in design of $g_t(x_t|\mathbf{x}_{t-1})$.

Finally, the sequential importance sampling method can be defined as d Sequential importance sampling (SIS) steps as presented in Algorithm 5.

Algorithm 5: Sequential importance sampling (SIS) procedure

Data: n - number of samples

for $i = 1$ *to* n **do**
  Initialize $X_0^{(i)}$.
for $t = 1$ *to* d **do**
  for $i = 1$ *to* n **do**
    Draw $X_t^{(i)}$ from $g_t(x_t|\mathbf{X_{t-1}^{(i)}})$, and let $\mathbf{X}_t^{(i)} = (\mathbf{X}_{t-1}^{(i)}, X_t^{(i)})$.
    Compute

$$w_t(\mathbf{X}_t^{(i)}) = w_{t-1}(\mathbf{X}_{t-1}^{(i)}) \frac{f(X_t^{(i)}|\mathbf{X}_{t-1}^{(i)})}{g(X_t^{(i)}|\mathbf{X}_{t-1}^{(i)})}.$$

5.2.1. Improving the SIS procedure with resampling

When the system grows, the variance of the importance weights w_t increases since the process becomes martingale [18]. After a certain number of steps, many of the weights become very small and a few very large. In that situation one may want to use a resampling strategy.

The role of resampling is to prune away "bad" samples and to split the good ones by rearranging the samples in existing sample set and modifying their weights accordingly. The new set of samples is also properly weighted by new weights with respect to g.

The resampling step is done on the existing partial sample set before expanding it with the SIS step before inner loop in Algorithm 5.

Two classic resampling techniques - simple random sampling and residual sampling - are presented as Algorithms 6 and 7. Residual sampling dominates the simple random sampling in having smaller estimator variance [17].

Algorithm 6: Simple random sampling for the SIS procedure

Data: $S_t = \{\mathbf{X}_t^{(j)}, j = 1, \ldots, n\}$ - collection of n partial samples of length t, which are properly weighted by the collection of weights $W_t = \{w_t^{(j)}, j = 1, \ldots, n\}$ with respect to the density g

Sample a new set of partial samples, S_t' from S_t according to weights $w_t^{(j)}$.
Assign equal weights W_t/n, to samples in S_t' where $W_t = \sum_{i=1}^{n} w_t^{(i)}$.

Algorithm 7: Residual resampling for the SIS procedure

Data: $S_t = \{\mathbf{X}_t^{(j)}, j = 1, \ldots, n\}$ - collection of n partial samples of length t, which are properly weighted by the collection of weights $W_t = \{w_t^{(j)}, j = 1, \ldots, n\}$ with respect to the density g

1 **for** $j = 1$ *to* n **do**

2 Retain $k_j = \lfloor n w_t^{(*j)} \rfloor$ copies of $\mathbf{X}_t^{(j)}$ where $w_t^{(*j)} = w_t^{(j)}/W_t$.

3 Let $n_r = n - \sum_{j=1}^{n} k_j$.

4 Obtain n_r draws from S_t with probabilities proportional to $n w_t^{(*j)} - k_j, \forall j = 1, \ldots n$.

5 Assign equal weights W_t/n, to samples in S_t' where $W_t = \sum_{i=1}^{n} w_t^{(i)}$.

5.2.2. Resampling schedule

The resampling step tends to result in a better group of ancestors so as to produce better descendants. The success of resampling, however, relies heavily on the Markovian structure among the state variables $x_1, x_2, \ldots x_d$. If resampling from set $\{\mathbf{x}_{t-1}^{(j)}, j = 1, \ldots n\}$ is not equivalent to resampling from $\{x_{t-1}^{(j)}, j = 1, \ldots, n\}$ – the set of the "current state", frequent resampling will rapidly impoverish diversity of the partial samples produced earlier. When no simple Markovian structure is present, frequent resampling generally gives bad results.

For this reason, it is desirable to prescribe scheduling for resampling to take place. The resampling schedule can be either deterministic or dynamic. When the schedule is dynamic, some small bias may be introduced.

With a deterministic schedule, we conduct resampling at time $t_0, 2t_0, \ldots,$ where t_0 is given in advance. In a dynamic schedule, a sequence of thresholds $c_1, c_2, \ldots, c_d$ is given in advance. We monitor the coefficient of variation of the weights cv_t^2 and invoke the resampling step when event $cv_t^2 > c_t$ occurs. A typical sequence of c_t can be $c_t = a + bt^{\alpha}$.

Increasing c_t after each SIS step makes sense since, as the system evolves, cv_t^2 increases stochastically while the variance of importance weights increases.

5.3. Sequential importance sampling epidemic source detector

Given snapshot $\vec{s}_*$ that holds all infected nodes up to time T we want to determine the probability $P(\theta_i|\vec{S} = \vec{s}_*)$ of an epidemic starting in node θ_i. Since all apriori probabilities $P(\theta_i)$ are the same, we can approximate aposteriori probabilities $P(\vec{S} = \vec{s}_*|\theta_i)$ and use them to determine $P(\theta_i|\vec{S} = \vec{s}_*)$ as we did in 4.2. These aposteriori probabilities were estimated with Direct Monte Carlo and Soft Margin method up to a multiplicative constant which can also be done using Sequential importance sampling technique.

First note the SIS step as defined in Algorithm 5 is based on densities of a complete history of the process or, at time t, all the process steps up to time t. The target density is thus the joint probability of all steps taken in the process. Since we are only interested in the final realization, it makes sense to use target and importance probability distributions of the form

$$\begin{aligned} f(s_T) &= f_t(i_1, r_1) f_t(i_2, r_2|i_1, r_1) f_t(i_3, r_3|i_2, r_2) \dots f_t(i_T, r_T|i_{T-1}, r_{T-1}) \\ g(s_T) &= g_1(i_1, r_1) g_2(i_2, r_2|i_1, r_1) g_3(i_3, r_3|i_2, r_2) \dots g_t(i_T, r_T|i_{T-1}, r_{T-1}) \end{aligned} \tag{5.8}$$

where i_k denotes a vector of infected nodes after SIS step k, and r_k denotes a vector of recovered nodes after step k. Note that $i_k \cup r_k = s_k$ and $i_k \cap r_k = \emptyset$. Each SIS step corresponds to one time step of discrete SIR simulation and realization s_T belongs to discrete random variable $\vec{S}_\theta$.

Each adjacent element of the sequence $(i_1, r_1), (i_2, r_2), (i_3, r_3), \dots, (i_T, r_T)$ is connected with one SIR step. These SIR steps build up the target distribution $f(s_T)$ and since each discrete SIR step is described by the same process 3.1, the target transitional distributions $f_t(i_k, r_k|i_{k-1}, r_{k-1})$ for each SIS step are the same. On the other hand, the SIS procedure let us change the transitional importance distributions $g_k(i_k, r_k|i_{k-1}, r_{k-1}), k \in \{1..T\}$ based on the current set of partial samples as long as we are able to calculate target transitional probability $f_t(i_k, r_k|i_{k-1}, r_{k-1})$ for each sample.

5.3.1. Modelling the target distribution

We can evaluate the partial target distribution $f_t(i_k, r_k|i_{k-1}, r_{k-1})$ in closed form. This is exactly the probability of one time step SIR transition given with Formula 3.2:

$$P(i_k, r_k|i_{k-1}, r_{k-1}) = \left[\Pi_{v\in E_1}(1-(1-p)^{nei_{i_{k-1}}(v)})\right]\left[\Pi_{v\in E_2}(1-p)^{nei_{i_{k-1}}(v)}\right] \cdot \left[\Pi_{v\in E_3} q\right]\left[\Pi_{v\in E_4}(1-q)\right],$$

with events $E_1 - E_4$ as defined in 3.1.2.

5.3.2. Modelling the importance distribution

With our sequential sampling procedure we will try to estimate the ratio of realizations at time T that are equal to $\vec{s}_*$ for some fixed starting node θ_i up to a multiplicative constant. The ideal importance distribution is biased towards that goal. Since we are building the final source distribution sequentially from partial samples, our biased sampling must sample reasonably well at each step – it must not be to "slow" or too "fast", especially since it is not clear what samples at mid steps are valuable to us.

However, it is certain we do not want to infect the nodes that were never infected or recovered in the snapshot $\vec{s}_*$. By excluding those nodes from events $E_1 - E_4$ while retaining fixed p we implicitly make the probability of infection per time step for the nodes in the target realization higher.

Modified transitions and their corresponding transitional probabilities at each time step can be determined from i_{k-1}, r_{k-1} and $\vec{s}_*$. For the SIR model, the transitions are similar to 3.1:

- T_1' : **if** $v \in s_*$ and $v \notin i_{k-1}$ and $v \notin r_{k-1}$
  - infect v with probability $1-(1-p)^{nei_{i_{k-1}}(v)}$
- T_2' : **if** $v \notin s_*$ and $v \notin i_{k-1}$ and $v \notin r_{k-1}$
  - infect v with probability 0
- T_3' : **if** $v \in i_{k-1}$
  - recover node v with probability q.

This biased transitional distribution $g_k(i_k, r_k|i_{k-1}, r_{k-1})$ defined by one-time-step long transitions $T_1' - T_3'$ can be sampled without conducting any SIR simulations. The sample is generated by traversing the nodes in the target realization $\vec{s}_*$. If the node is not infected yet, it may be infected if its neighbours are infected with probability $1-(1-p)^{nei_{i_{k-1}}(v)}$, as in transition T_1'. Otherwise, the recovery process is simulated with probability q for recovery. Since all transitions are independent, the biased transitional

probability can be computed easily. The generator of partial samples governed by biased transitional distribution $g_k(i_k, r_k|i_{k-1}, r_{k-1})$ is presented in Algorithm 8.

Algorithm 8: Partial sample generator based on importance distribution $g_k(i_k, r_k|i_{k-1}, r_{k-1})$

Data: (p, q) - parameters of the SIR process, i_{k-1} - infected nodes at the beginning of time step k, r_{k-1} recovered nodes at the beginning of time step k, s_* - target realization

Result: i_k - infected nodes after time step k, r_k - recovered nodes after time step k, $g = g(i_k, r_k|i_{k-1}, r_{k-1})$ - probability of generating realizations i_k, r_k based on i_{k-1}, r_{k-1} from the importance generator

$i_k = i_{k-1}, r_k = r_{k-1}, g = 1$
foreach $v \in nei(i_{k-1}) \cap s_*$ **do**
  if $v \notin i_{k-1}$ ***and*** $v \notin r_{k-1}$ **then**
    $D = |nei_{i_{k-1}}(v)|$
    let *infection* occur with probability $p_v = 1 - (1-p)^D$
    if *infection* ***occured*** **then**
      update $i_k(v)$
      $g = g \cdot p_v$
    else
      $g = g \cdot (1 - p_v)$
foreach $v \in i_{k-1} \cap s_*$ **do**
  let *recovery* occur with probability q
  if *recovery* ***occured*** **then**
    update $i_k(v)$ and $r_k(v)$
    $g = g \cdot q$
  else
    $g = g \cdot (1 - q)$
return (i_k, r_k, g)

In order to calculate importance weights $w_k(i_k, r_k)$ recursively like in 5.7, one needs to be able to compute target transitional probability $f_t(i_k, r_k|i_{k-1}, r_{k-1})$ based on known partial sample i_k, r_k and its base i_{k-1}, r_{k-1}. This computation can be done by traversing potential active nodes in the SIR process: infected nodes at the beginning of time step k defined by set i_{k-1} for recovery, and their susceptible neighbours for infection. Computation of target transitional probability $f_t(i_k, r_k|i_{k-1}, r_{k-1})$ is described

by Algorithm 9.

Note the biased generator in Algorithm 8 can be run with arbitrary parameters p and q. The samples generated with arbitrary p and q can be included in the set of samples as long as the transitional target distribution can be calculated.

Algorithm 9: Computation of $f(i_k, r_k|i_{k-1}, r_{k-1})$ - target transitional probability of generated partial sample

Data: (p, q) - parameters of the SIR process, i_{k-1} - infected nodes at the beginning of step k, r_{k-1} recovered nodes at the beginning of step k, i_k - infected nodes after step k, r_k - recovered nodes after step k, s_* - target realization

$f = 1$
foreach $v \in nei(i_{k-1})$ **do**
  if $v \notin i_{k-1}$ ***and*** $v \notin r_{k-1}$ **then**
    $D = |nei_{i_{k-1}}(v)|$
    if $v \in i_k$ **then**
      $f = f \cdot [1 - (1 - p)^D]$
    else
      $f = f \cdot (1 - p)^D$
foreach $v \in i_{k-1}$ **do**
  if $v \in r_k$ **then**
    $f = f \cdot q$
  if $v \notin r_k$ **then**
    $f = f \cdot (1 - q)$
return f

5.3.3. Building the epidemic source detector

To estimate ratio of realizations that are equal to the target realization $\vec{s_*}$ up to multiplicative constant at time T for a fixed source θ, SIS step (Algorithm 5) will be conducted T times. In each step, the samples are generated using importance distribution $g(i_k, r_k|i_{k-1}, r_{k-1})$ and the transitional target probability $f(i_k, r_k|i_{k-1}, r_{k-1})$ as well as transitional importance probability $g(i_k, r_k|i_{k-1}, r_{k-1})$ for the generated sample are calculated. Based on the data one can obtain the transitional partial weight w_k for said partial sample.

After expanding the partial samples T times, one obtains the estimate on number of

realizations equal to the target realization $\vec{s}_*$ by taking weighted average of all weights corresponding to the realization hits in the final sample set,

$$m_i = \frac{1}{n} \sum_{j=1}^{n} w_T^{(j)} 1\{s_T^{(j)} = \vec{s}_*\}, \tag{5.9}$$

where $s_T^{(j)} = i_T^{(j)} \cup r_T^{(j)}$ corresponds to one realization of discrete random variable $\vec{S}_\theta$. Finally, the pseudocode for the source detector based on sequential importance sampling is presented in Algorithm 10.

Algorithm 10: Sequential importance sampling estimation of expected number of realizations completely corresponding to $\vec{s}_*$ after T time steps for a fixed source node θ_i

Data: $\mathbf{G}$ - network, (p, q) - parameters of the SIR process, θ_i - proposed source node, $\vec{s}_*$ - target realization, T - temporal threshold, n - number of samples

for $j = 1$ *to* n **do**
  Initialize $i_0^{(j)} = \{\theta_i\}, r_0^{(j)} = \emptyset, w_0^{(j)} = 1$
for $t = 1$ *to* T **do**
  resample$(\{(i_{t-1}^{(j)}, r_{t-1}^{(j)}) \ \forall j \in [1..n]\})$
  for $j = 1$ *to* n **do**
    Draw $i_t^{(j)}, r_t^{(j)}$ from $g_t(i_t, r_t | i_{t-1}^{(j)}, r_{t-1}^{(j)})$ with Algorithm 8.
    Compute $f_t(i_t^{(j)}, r_t^{(j)} | i_{t-1}^{(j)}, r_{t-1}^{(j)})$ with Algorithm 9.
    Compute

$$w_t(i_t^{(j)}, r_t^{(j)}) = w_{t-1}(i_{t-1}^{(j)}, r_{t-1}^{(j)}) \frac{f_t(i_t^{(j)}, r_t^{(j)} | i_{t-1}^{(j)}, r_{t-1}^{(j)})}{g_t(i_t^{(j)}, r_t^{(j)} | i_{t-1}^{(j)}, r_{t-1}^{(j)})}.$$

$m_i = 0$
for $j = 1$ *to* n **do**
  if $i_T^{(j)} \cup r_T^{(j)}$ ***equals*** $\vec{s}_*$ **then**
    $m_i = m_i + w_T(i_T^{(j)}, r_T^{(j)})$
return (m_i)

It is obvious the accuracy of the estimate 5.9 depends on number of realizations in the sample set that are equal to $\vec{s}_*$ after final SIS step. To increase this number, one can alter the generation of samples in the final SIS step to maximize the number of realization hits. This can safely be done by altering the importance generator and

making all eligible nodes infected with probability $p = 1$ at the final step of the SIS procedure.

It may also be reasonable to increase p at each step of the SIS procedure but it is not clear when or in what volume this should be done. By using too high or too low p at earlier SIS steps, obtained samples will in general have really small final weight w_T since its target transitional probability will be close to 0.

Additionally, one might want to use a resampling technique for simulations with many SIS steps or to fix said small weights. This has to be done carefully too since our target event is rare and weights w are naturally small. By using the resampling schedule based on ESS or vc^2, the decision on when to resample is not governed by absolute value of the weights, but rather on their coefficient of variation. For example, resampling can be invoked in SIS step t when $vc^2(\mathbf{w}_{t-1}) > 2^t$.

5.3.4. Soft Margin SIS source detector

Incorporating Soft Margin ideas to sequential importance sampling based epidemic source detector allows us to use all generated samples in the estimation of m_i, as opposed to using only the ones completely corresponding to the target realization $\vec{s}_*$ as presented in Algorithm 10.

Using the set of generated biased samples $\{(i_T^{(j)}, r_T^{(j)}) \ \forall j \in [1..n]\}$, we approximate m_i with

$$m_i \approx \sum_{j=1}^{n} w_T(i_T^{(j)}, r_T^{(j)}) e^{\frac{(\varphi_j - 1)^2}{a^2}} \tag{5.10}$$

where $e^{\frac{(\varphi_j - 1)^2}{a^2}}$ corresponds to Gaussian weighting function with $\varphi_i = \varphi(s_T^{(j)}, \vec{s}_*)$ defined as Jaccard similarity 4.8.

Compared to Soft Margin detector described in Algorithm 4, Soft Margin SIS detector uses the set of biased samples and will have smaller estimator variance for the same parameter a and the same number of samples. Additionally, since all biased samples are realizations equal to some subset of $\vec{s}_*$, the realizations containing nodes not in $\vec{s}_*$ are excluded from approximation 5.10, as opposed to Soft Margin approximation where these realizations are also being used in the approximation.

5.3.5. Time and space complexity of SIS source detector

To generate estimation of source probability distribution $P(\Theta = \theta_i | \vec{S} = \vec{s}_*)$ one needs to run Sequential importance sampling procedure as described in Algorithm 10 m

times where m denotes the number of potential sources in the observed realization, i.e. number of realizations of random variable Θ.

Within each Sequential importance sampling procedure, partial sampling step is conducted T times, with each time n partial samples being extended with the biased generator Algorithm 8.

Time complexity of generating one full sample (running Algorithm 8 T times) is $O(m\text{min}(T, \frac{1}{q}))$ since one tries to infect or recover at most once no more than each node in the target realization $\vec{s}_*$ which is at most m nodes and the expected number of time steps an infected node remains active is $\text{min}(T, \frac{1}{q})$. Infecting or recovering process is simulated by regular uneven dice throw.

Each extension of a partial sample is followed by calculation of target transitional probability based on generated partial sample presented in Algorithm 9. The time complexity of running this algorithm for each extension of a partial sample is $O(m\langle d\rangle\text{min}(T, \frac{1}{q}))$ where $\langle d\rangle$ denotes average node degree. The term $m\langle d\rangle$ corresponds to average upper bound on the size of set $nei_{i_{k-1}} \cup i_{k-1}$ of infected nodes and their neighbours. This is exactly the set being traversed in the calculation of target transitional probability. Each infected node is expected to be in that set for $\text{min}(T, \frac{1}{q})$ time steps.

Finally, generating estimation $P(\Theta = \theta_i|\vec{S} = \vec{s}_*)$ for all nodes in Θ has time complexity $O(nm^2\langle d\rangle\text{min}(T, \frac{1}{q}))$ since Algorithm 8 and Algorithm 9 are being run Tn times where n is the number of samples being used.

While the average time complexity of Direct Monte Carlo and Soft Margin Monte Carlo is $O(nmE[X]\langle d\rangle\text{min}(T, \frac{1}{q}))$ for the SIR model, comparing time complexity of the SIS procedure to Direct Monte Carlo and Soft Margin Monte Carlo is reduced down to where the size of the realization m lays in the distribution of epidemic size for the particular epidemic environment whose expectation is $E[X]$.

Introducing resampling to SIS detector does not influence the upper bound time complexity. Introducing Soft Margin estimation to SIS detector does not influence the upper bound time complexity either.

When talking about time complexity, one has to mention the range of number of samples for which the estimators typically converge. Breakdown on the converging number of simulations for epidemic dynamics on a lattice network is presented in Figure 6.7 and analysed in the next chapter.

The SIS detector can also be parallelised by assigning each worker a set of samples to expand. To enable resampling one needs to aggregate the partial weights for all partial samples corresponding to the same potential source node at each SIS step. In

that situation a worker may be assigned a set of potential source nodes for which it must estimate the expected number of target realizations m_i.

Regarding space complexity, sequential importance sampling requires a stored graph in a form of adjacency list and n stored partial samples represented by 3 bit-sets. The overall space complexity is $O(\max(L, n))$, where L is number of links in the network and n is number of samples.

6. Analysis of epidemic source detectors on the benchmark dataset

Before analysing performance of MAP-based epidemic source detectors it is important to note that the inverse problem of finding the epidemic source is ill-posed [19] while there might not exist a unique solution and the solution may change drastically with small change in initial conditions. For this reason a high accuracy in terms of determining the true source node cannot be expected.

The main goal in designing new epidemic source detectors is obtaining the results in terms of accuracy as similar to Direct Monte Carlo as possible with better convergence properties and shorter time of execution.

6.1. Benchmark dataset

Antulov-Fantulin et al. [7] provided a dataset of SIR realizations along with their estimations obtained with Direct Monte Carlo for 4 classes of detection problems based on SIR parameters (p, q, T): $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5)$ and $D = (p = 0.7, q = 0.7, T = 5)$. The benchmark dataset contains 160 such realizations on the lattice of size 30×30.

Different classes of SIR parameters are used in the dataset since each class yields epidemics of different expected size. Moreover, the source nodes of epidemics on the grid network for classes of SIR parameters A and B present lower detectability than source nodes of epidemics on the grid network for classes of SIR parameters C and D since they belong to low to medium and medium to high detectability zone, respectively, as presented in Figure 7.1. Source nodes of epidemics governed by SIR parameters in classes A and B thus produce lower accuracy on average.

The benchmark estimations of Antulov-Fantulin et al. [7] were obtained with Direct Monte Carlo under convergence conditions 4.7 with $c = 0.05$. These estimations provided along with the benchmark dataset can be used as a benchmark detector to

measure similarity of other MAP based epidemic source detectors with Direct Monte Carlo.

The similarity and performance of source detectors will be compared using accuracy, accuracy w.r.t. MAP estimation of Direct Monte Carlo benchmark detector and distribution of required number of samples for which the estimations converge.

Direct Monte Carlo, Soft Margin and Sequential importance sampling based epidemic source detectors were implemented in C++ and parallelized using OpenMPI [20] library. The complete source code is hosted on GitHub [1].

6.2. Correctness of Direct Monte Carlo implementation

Direct Monte Carlo implementation was ran on the benchmark dataset with convergence conditions 4.7, $c = 0.05$. The correctness of Direct Monte Carlo implementation is confirmed with coinciding accuracies of Direct Monte Carlo implementation and the benchmark detector as presented in Figure 6.1.

The number of simulations required to fulfil convergence conditions was typically in the interval $n \in [10^6, 10^9]$ as presented in Figure 6.2 and reported by Antulov-Fantulin et al. [7] for their Direct Monte Carlo implementation referred in the plot as the benchmark detector.

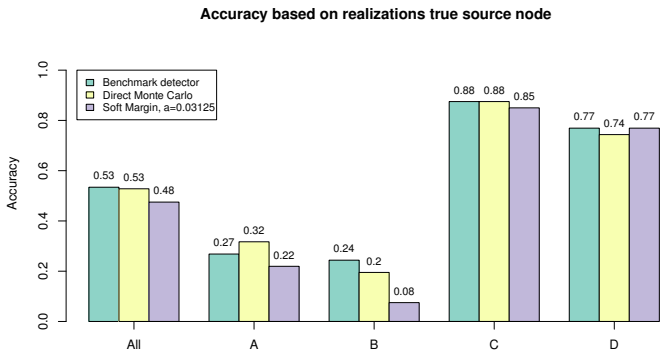

Figure 6.1: Accuracy of Direct Monte Carlo and Soft Margin implementations on the benchmark dataset with convergence conditions 4.7, $c = 0.05$. For Soft Margin fixed parameter $a = 1/2^5$ was used. Classes A, B, C, D correspond to classes of SIR parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$.

[1] https://github.com/imih/cmplx

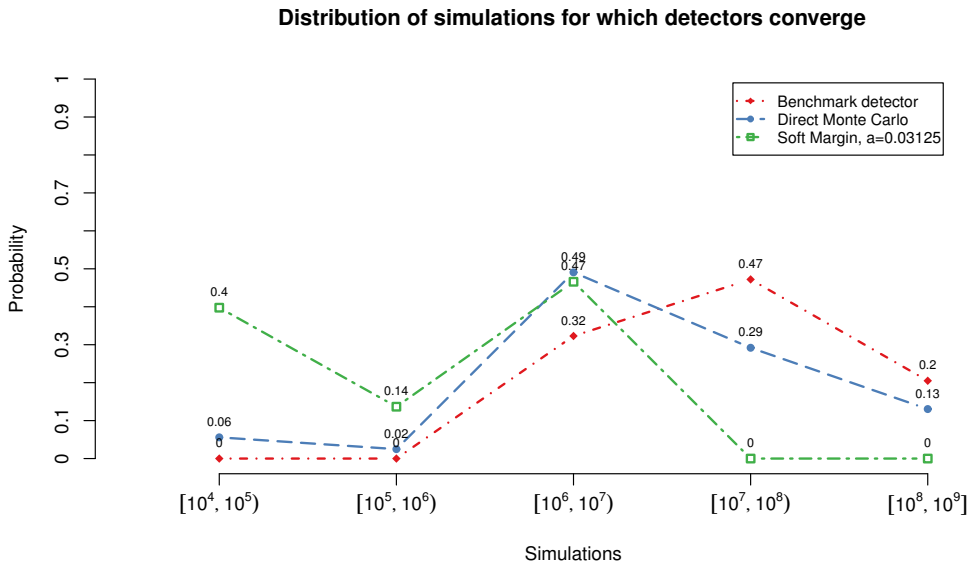

Figure 6.2: Distribution of number of samples needed for detectors to converge on the benchmark dataset under convergence conditions 4.7, $c = 0.05$ for Direct Monte Carlo and Soft Margin implementations and Direct Monte Carlo benchmark detector.

6.3. Correctness of Soft Margin implementation

To show correctness of the Soft Margin implementation the detector was ran on the benchmark dataset. To make the error in approximation the same for all benchmark instances fixed parameter $a = \frac{1}{2^5}$ was used.

Since Soft Margin provides an approximation of Direct Monte Carlo method it is expected that Soft Margin will yield results with accuracy (governed by parameter a) not higher than the accuracy of Direct Monte Carlo, as presented in Figure 6.1.

The Soft Margin estimations follow the estimations of Direct Monte Carlo fairly well, especially for classes of SIR parameters C and D on which the source node is highly detectable, as presented in Figure 6.3. This is in accordance to results obtained by Soft Margin implementation of Antulov-Fantulin et al. [7] referred in the plot as the Soft Margin benchmark detector. The Soft Margin benchmark detector was reported to be ran under the same convergence conditions as the implemented Soft Margin detector.

Soft Margin requires fewer number of simulations for estimations to converge. Converging number of simulations n was selected based on convergence conditions 4.7, $c = 0.05$ and they were tipically in the range $n \in [10^4, 10^6]$ as presented in Figure

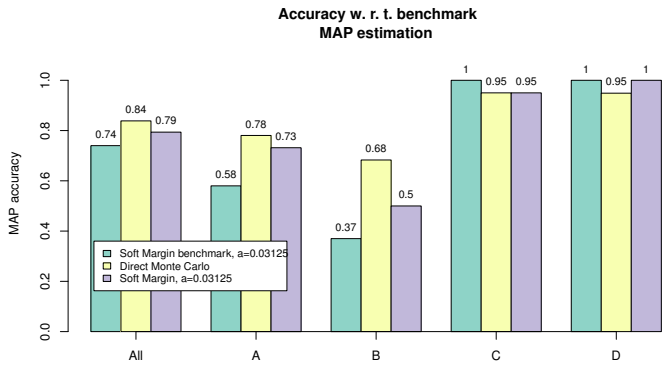

Figure 6.3: MAP accuracy of Soft Margin detector of Antulov-Fantulin et al. [7], Direct Monte Carlo and Soft Margin implementations. MAP accuracy is accuracy relative to MAP estimation of the benchmark detector. The simulations were held with convergence conditions 4.7, $c = 0.05$ and fixed parameter $a = 1/2^5$ for Soft Margin. Classes A, B, C, D correspond to classes of SIR parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$.

6.2 and reported by Antulov-Fantulin et al. [7] for their Soft Margin implementation.

6.4. Sequential importance sampling epidemic source detector

Sequential importance sampling was run under importance sampling distribution defined by biased generator presented in Algorithm 8 with the following properties:

- parameter p is fixed in steps $t < T$ and $p = 1$ in the final step $t = T$,
- parameter q is fixed,
- at each step, only nodes that are in the given final realization may be infected,
- nodes that are infected may be recovered with probability q.
- the simulations are held under Direct Monte Carlo convergence conditions 4.7 with $c = 0.05$, starting from $n = 10^4$ samples.

To show correctness of Sequential importance sampling detector accuracy of estimations was compared with estimations of the benchmark detector. Accuracies for benchmark detector and Sequential importance sampling detector follow similar pattern overall and for each of the four classes of SIR parameters, as presented in Figure 6.4. For classes A and B they are low, and for classes C and D they are high.

Compared to accuracy of the Soft Margin detector, Sequential importance sampling outperforms the Soft Margin in terms of accuracy which is especially highlighted for epidemics on low detectability SIR parameters of classes A and B. For classes C and D that belong to medium to high detectability zone of parameters the source can also be successfully estimated by the Soft Margin detector.

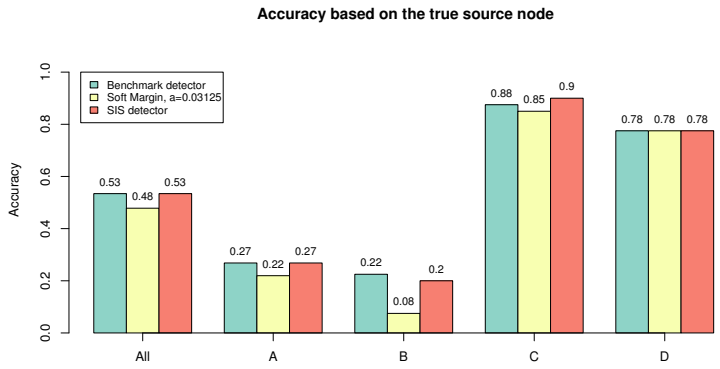

Figure 6.4: Accuracy of Sequential importance sampling detector on the benchmark dataset with convergence conditions 4.7, $c = 0.05$ for classes of SIR parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$.

To present the overall similarity of estimations of Sequential importance sampling detector to the ones of the benchmark detector, MAP accuracy is used. This accuracy refers to the portion of MAP estimations that are equal to corresponding MAP estimations of the benchmark detector provided in the benchmark dataset.

Antulov-Fantulin et al. [7] provided MAP accuracies for range of epidemic source detectors out of which their Soft Margin implementation yields highest MAP accuracy on the benchmark dataset.

Comparison of MAP accuracy for Soft Margin detector and SIS detector presented in Figure 6.5 shows how Sequential importance sampling detector outperforms Soft Margin in having higher MAP accuracy.

In other words, the estimations of Sequential importance sampling detector are more similar to Direct Monte Carlo estimations than Soft Margin approximations are. This makes sense since Soft Margin method is the approximation of Direct Monte Carlo. Sequential importance sampling detector differs from Direct Monte Carlo only by smaller estimator variance while still being a valid Monte Carlo method.

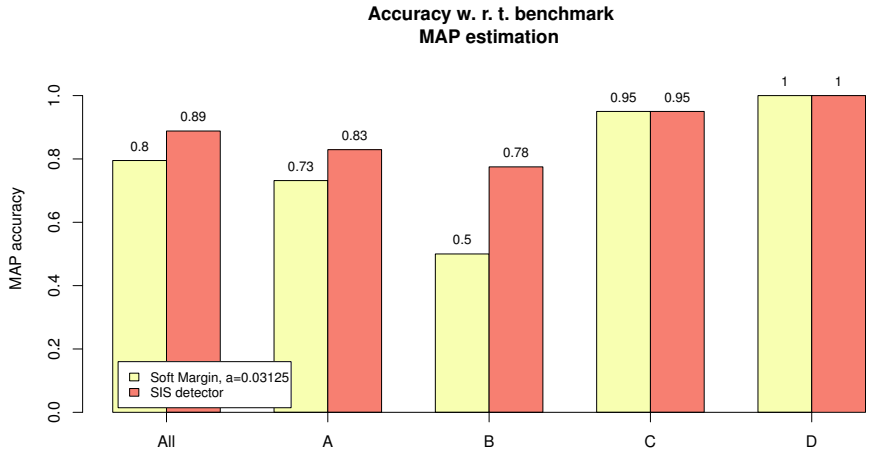

Figure 6.5: MAP accuracy of Sequential importance sampling detector on the benchmark dataset with convergence conditions 4.7, $c = 0.05$ for classes of SIR parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$.

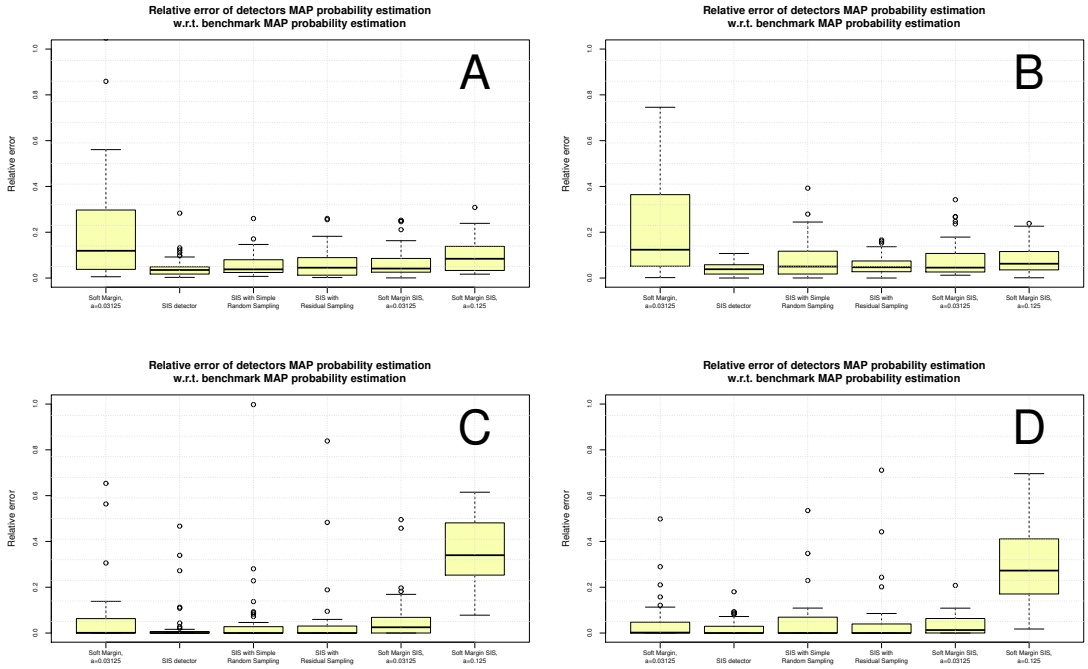

Figure 6.6: Relative error of estimated MAP source probability w.r.t the estimated MAP probability of the benchmark detector for a range of Sequential importance sampling detectors grouped by classes of SIR parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$.

The similarity between estimations obtained with Sequential importance sampling and Direct Monte Carlo detector also presents itself as a low relative error of MAP probability estimation w. r. t. benchmark Direct Monte Carlo estimated MAP probability across all classes of SIR parameters as presented in Figure 6.6. The relative error of Sequential importance sampling is substantially smaller than the same error for Soft Margin detector across all classes of SIR parameters.

One crucial metric of time complexity for Monte Carlo based techniques is the distribution of number of samples required for detector to converge under convergence conditions 4.7, $c = 0.05$. The distribution of converging number of samples for estimations on the benchmark dataset reveals Sequential importance sampling detector requires between 10^4 and 10^6 samples to converge on average for benchmark source detection problems on the lattice network as presented in Figure 6.7, while Direct Monte Carlo detector requires at least 10^6 simulations for estimations to converge on the same dataset.

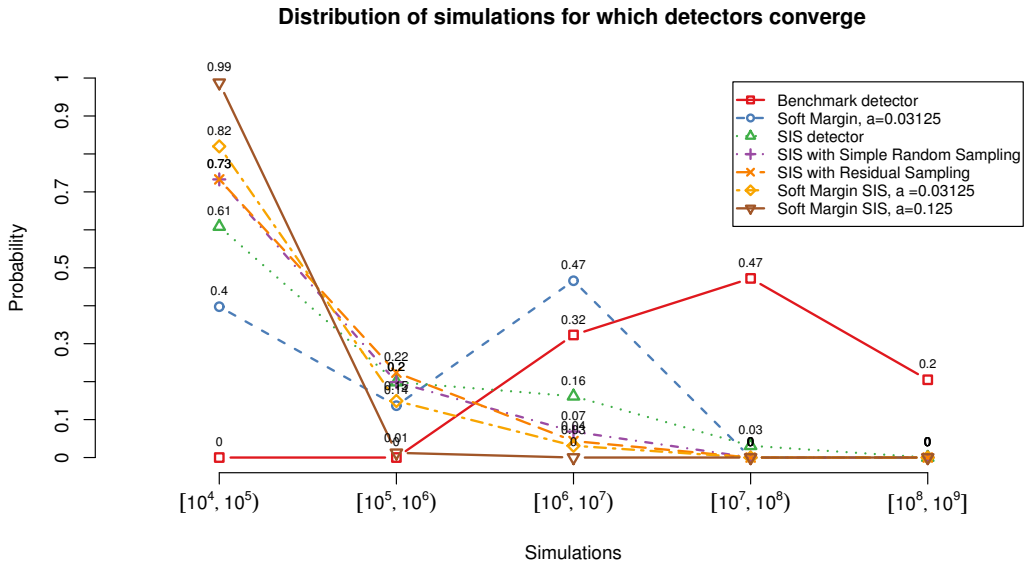

Figure 6.7: Distribution of number of samples required for detectors to converge on the benchmark dataset under convergence conditions 4.7, $c = 0.05$ for a range of Sequential importance sampling detectors on the benchmark dataset.

Let's examine the form of distribution of required converging number of samples for Direct Monte Carlo benchmark detector, Soft Margin and Sequential importance sampling detectors in more detail. The distributions of required converging number of samples for Direct Monte Carlo and Soft Margin take similar form. Introduction of the Soft Margin approximation shifted left the distribution of converging number of

samples for one order of size. Around 50% percent detection problem estimations on the benchmark dataset still require 10^6 samples for estimation to converge.

On the other hand, using sequential importance sampling significantly alters the distribution of required converging number of samples that now takes exponentially decreasing form in which between 60% and 80% percent of the detection problems required less than 10^5 samples to converge.

6.5. Sequential importance sampling detector with resampling

To examine the impact of resampling techniques on Sequential importance sampling detector, simple random sampling and residual sampling were incorporated. The resampling schedule was based on coefficient of variation of the weights cv^2 defined as a measure of effectiveness of importance distribution in 5.5.

Resampling is invoked before the generation of new partial samples at t-th SIS step as marked in Algorithm 10 if it holds $vc^2(\mathbf{w}_{t-1}) \geq 2^t$ where $\mathbf{w}_{t-1}$ is set of weights corresponding to partial samples generated at the previous SIS step. The variation of weights cv^2 increases drastically after each SIS step on the benchmark dataset and resampling limits the increase of variation, as presented in Figure 6.8.

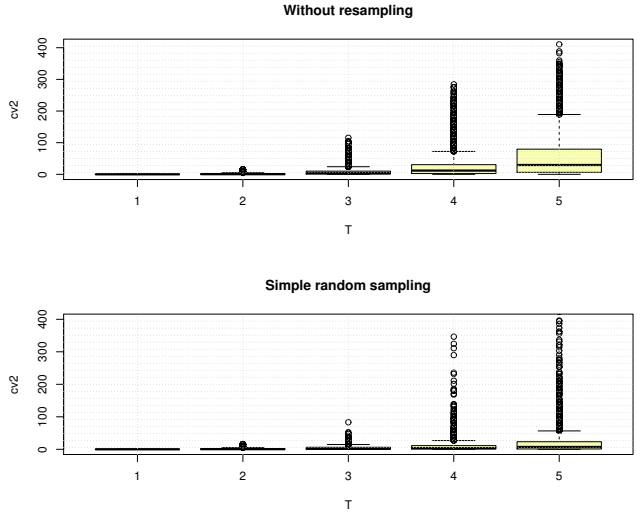

Figure 6.8: Distribution of squared variation of weights cv^2 across discrete time steps T for detection problems in the benchmark dataset estimated with Sequential importance sampling detector without resampling and with simple random sampling using $n = 10^5$ samples.

Incorporating resampling step to Sequential importance sampling detector does not effect accuracy nor MAP accuracy on average for the benchmark dataset as presented in Figure 6.9 and 6.10, respectively.

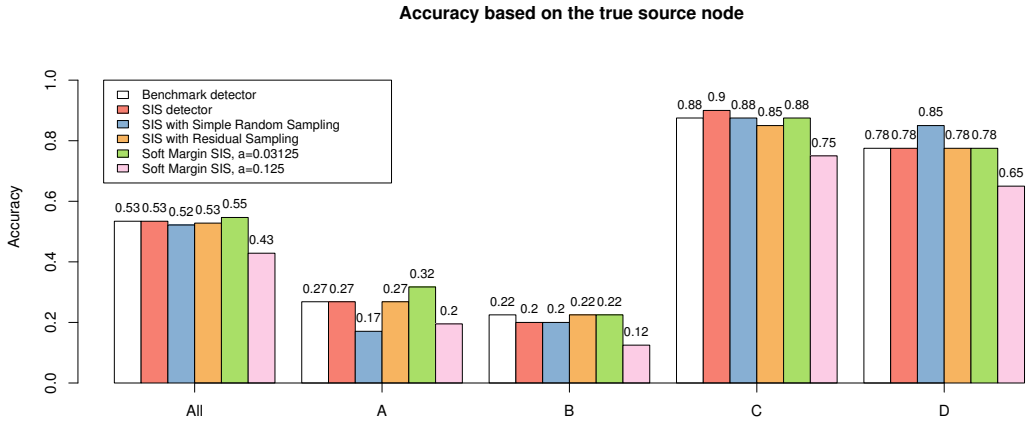

Figure 6.9: Accuracy of Sequential importance sampling detectors on the benchmark dataset with convergence conditions 4.7, $c = 0.05$ for classes of SIR parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$.

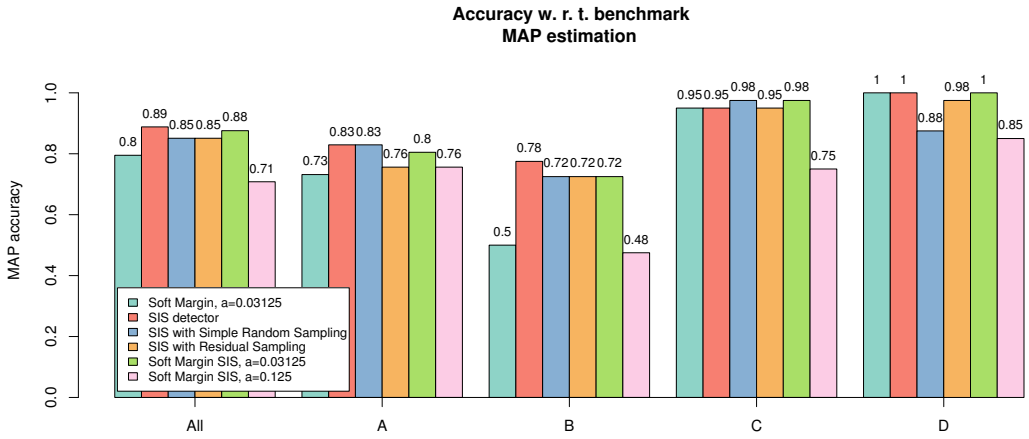

Figure 6.10: MAP accuracy of Sequential importance sampling detectors on the benchmark dataset with convergence conditions 4.7, $c = 0.05$ for classes of SIR parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$.

However, incorporating resampling technique introduces a small bias in the estimator visible as a change in distribution of relative error of MAP probability as presented

in Figure 6.6. The relative MAP probability error also reveals smaller estimator variance of residual sampling compared to simple random sampling on estimations for classes of SIR parameters B and D with high recovery rate.

The impact of resampling on the source detector is also visible in higher ratio of benchmark instances requiring smaller number of drawn samples to converge, as presented in Figure 6.7.

It is interesting to observe accuracy in the group of detection problems converging with the same number of samples. This group accuracy presented in Figure 6.11 shows the SIS detector with incorporated resampling methods observes higher accuracy for problems requiring less than 10^5 samples to converge compared to Soft Margin detector on corresponding group of detection problems even though this group of detection problems for SIS detectors is bigger than the corresponding group of detection problems requiring the same number of simulations to converge with Soft Margin detector as presented in Figure 6.7.

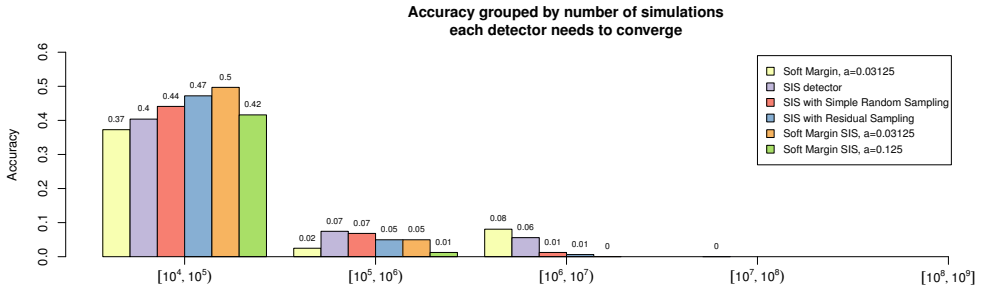

Figure 6.11: Accuracy of range of Sequential importance sampling detectors on the benchmark dataset calculated seperately for each group of benchmark detection problems corresponding to the same number of samples their estimation required to converge under convergence conditions 4.7, $c = 0.05$.

Additionally, with group accuracy minor difference in variance of estimators incorporating simple random sampling and residual sampling is confirmed.

6.6. Sequential importance sampling and Soft Margin

Adding Soft Margin approximation to Sequential importance sampling detector means evaluating final samples with Gaussian weighting as presented in 5.10. We will observe two detectors, both with fixed parameters $a = \frac{1}{2^5}$ and $a = \frac{1}{2^3}$.

Even though Soft Margin SIS detector with $a = \frac{1}{2^3}$ converges faster as presented

in Figure 6.7, its overall accuracy is lower than the accuracy of other SIS detectors, as presented in Figures 6.9, 6.10, 6.6, 6.11.

On the other hand, the Soft Margin SIS detector with $a = \frac{1}{2^5}$ outperforms benchmark detector and other SIS detectors with higher accuracy as presented in Figure 6.9. This detector also converges more quickly on the benchmark dataset as presented in Figure 6.7.

Moreover, the group accuracy of detection problems requiring less than 10^5 samples to converge for Soft Margin SIS detector is higher than the same accuracy for other SIS detectors and Soft Margin as presented in Figure 6.11.

6.7. Experimental execution time

For Soft Margin, Sequential importance sampling and Soft Margin SIS detector time execution was measured and compared. Execution time was measured on 12 cpu cores on Intel(R) Xeon CPU E5645 processor, 2.40GHz each. Detection was executed on the benchmark dataset with $n = 10^4$ samples using 16 MPI processes. The results are presented in Table 6.1

	Soft Margin	Sequential importance sampling	Soft Margin SIS
Min	0.1170 s	2.460 s	2.457 s
Median	0.9556 s	4.154 s	4.247 s
Mean	91.1847 s	4.708 s	4.703 s
Max	666.6846 s	9.225 s	9.239 s

Table 6.1: Execution times in seconds for Soft Margin and Soft Margin SIS detectors on the benchmark dataset estimated with $n = 10^4$ samples per potential source node.

Execution time of Soft Margin variates depending on expected epidemic size, while Sequential importance sampling detectors sample epidemics only from the subset of infected nodes in the given snapshot $\vec{s}_*$ making variation in execution time smaller.

The execution times of Sequential importance sampling and Soft Margin SIS detectors do not differ substantially.

7. Detectability of patient zero

The source detectability $D(\vec{s_*})$ can be defined via Shannon entropy $H(\vec{s_*})$ of the estimated source probability distribution $P(\Theta = \theta_i|\vec{S} = \vec{s_*})$ normalized by entropy of the uniform distribution as $D(\vec{s_*}) = 1 - H(\vec{s_*})$ [7].

When entropy $H(\vec{s_*})$ of $P(\Theta = \theta_i|\vec{S} = \vec{s_*})$ is close to 1, the detector has estimated that all potential nodes have the same probability of being the source node of observed epidemic, while low entropy H corresponds to the case where detector filtered out a single node or a few nodes as potential epidemic sources.

Apart from entropical detectability, when talking about ability the node can be detected by the source detector as an epidemic source, accuracy of the MAP estimator and number of samples that are on average required for estimation to converge play an important role.

Accuracy of estimation is crucial since it measures the overall successfulness of the estimation for the particular epidemic environment – network topology, source node, epidemic model, its parameters and, consequently, average epidemic size. At the same time, the number of samples a detector requires for its estimations to converge gives a picture of the detection complexity in terms of state space size for the particular epidemic environment.

7.1. Detectability based on parameters of the SIR model

To show how detectability presents itself in the parametric space of SIR parameters p and q, several simulations were ran using Soft Margin SIS detector with fixed $a = \frac{1}{2^5}$. The Soft Margin SIS detector was chosen since it produced the highest accuracy on the benchmark dataset as presented in Figure 6.9.

For each set of parameters (p, q) and various 4-connected lattice networks of different size, 50 epidemic simulations were conducted for $T = 5$ time steps starting from the central node in the lattice network. All epidemic simulations of size 1 were excluded. Soft Margin SIS detector was ran on each epidemic simulation with con-

verging conditions 4.7, $c = 0.05$ and number of samples in range $n \in [10^4, 10^6]$.

Influence of parameters p and q on entropical detectability for the SIR model is presented in Figure 7.1. For simulations on lattice of size 30×30 we observe the existence of different detectability regimes (or entropy regions), as reported by Antulov-Fantulin et al. [7]. Three entropy regions are observed: low detectability-high entropy region $(p < 0.2)$, intermediate detectability - intermediate entropy region $(0.2 < p < 0.7)$ and high detectability-low entropy region $(p > 0.7)$. The entropy regions are similarly distributed for different values of $q \in \{0, 0.5, 1\}$.

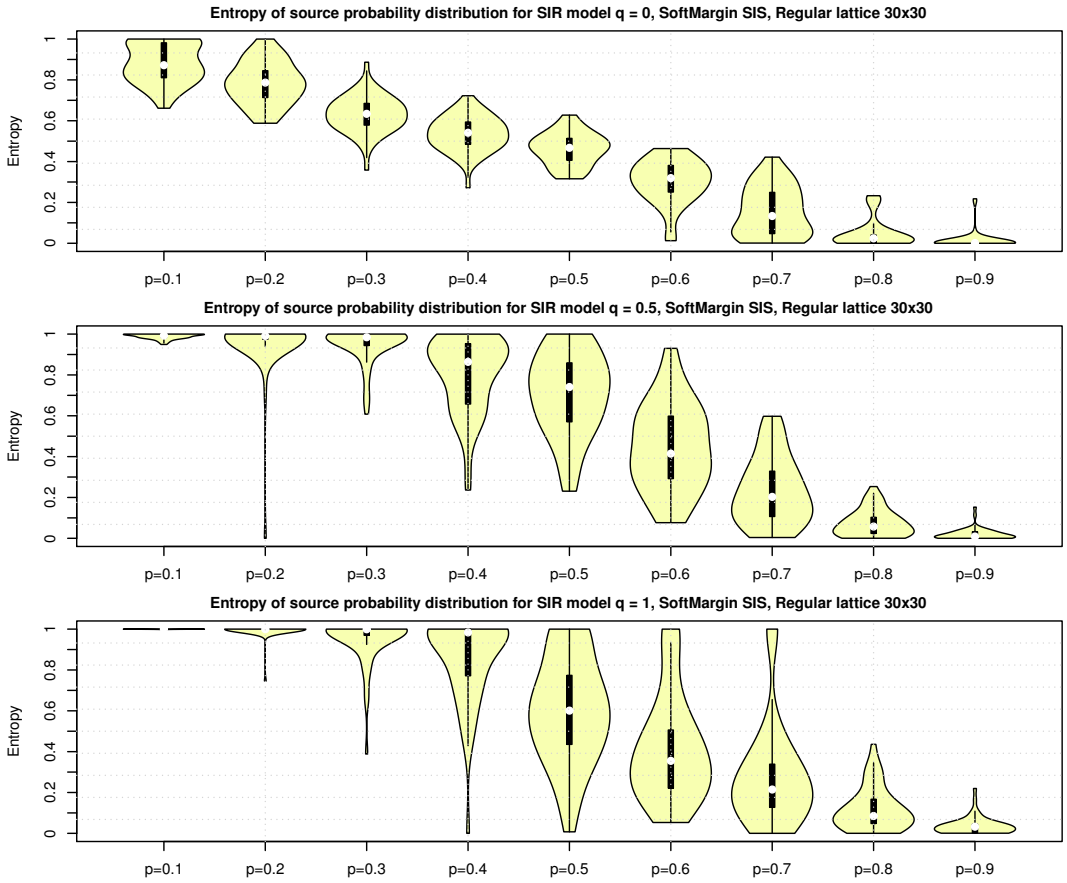

Figure 7.1: Violin plots of estimated entropy distribution for source probability distributions on 4-connected lattice 30×30 estimated with Soft Margin SIS method with $10^4 - 10^6$ samples and fixed $a = \frac{1}{2^5}$ under SIR model with different parameters p in range $0.1 - 0.9$, $q = \{0, 0.5, 1\}$ and $T = 5$.

Accuracy of source detections, as presented in Figure 7.2, follows the behaviour of entropical detectability – the accuracy grows while entropy gets lower. Additionally, accuracy does not differ significantly for the same value of parameter p across different values of parameter $q \in \{0, 0.5, 1\}$.

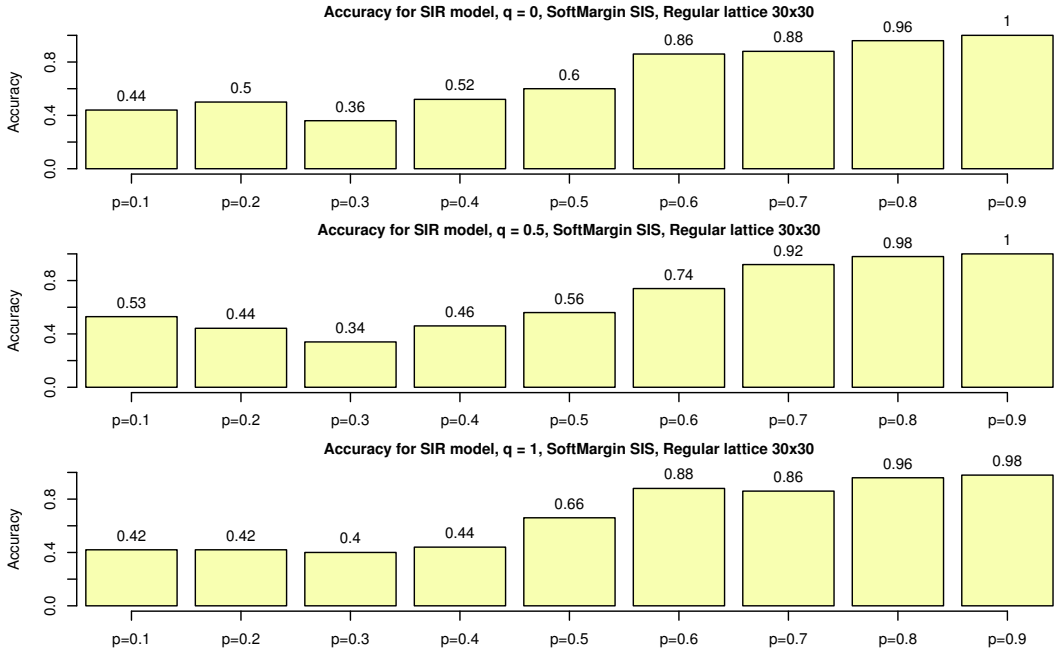

Figure 7.2: Accuracy of source MAP estimation on 4-connected lattice 30×30 estimated with Soft Margin SIS method with $10^4 - 10^6$ samples and fixed $a = \frac{1}{2^5}$ under SIR model with different parameters p in range $0.1 - 0.9$, $q = \{0, 0.5, 1\}$ and $T = 5$.

While observing the distribution of samples for which estimations converge under convergence conditions 4.7, $c = 0.05$ one can also detect the same three detectability regions, as presented in Figure 7.3. The region requiring the most samples is the intermediate entropy region while the high entropy and low entropy regions require minimal number of samples for estimations to converge. Across values of recovery parameter q, the estimations on intermediate entropy region and median value of q require the most samples to converge.

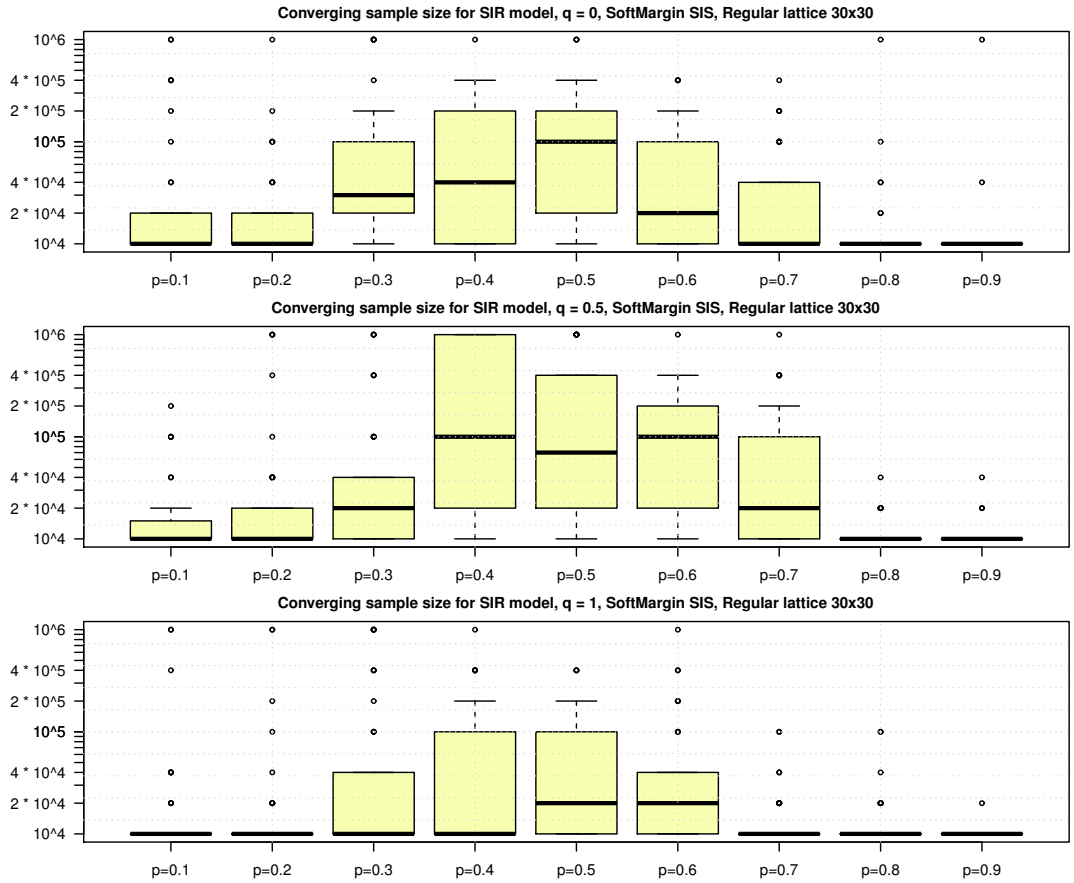

Figure 7.3: Box plots of converging samples distribution on 4-connected lattice 30×30 estimated with Soft Margin SIS method with $10^4 - 10^6$ samples and fixed $a = \frac{1}{2^5}$ under SIR model with different parameters p in range $0.1 - 0.9$, $q = \{0, 0.5, 1\}$ and $T = 5$.

It is interesting to see how detectability gets restricted in a regime where network topology restricts the epidemic spreading. By simulating epidemic on smaller lattices, the epidemic spreading gets restricted by the network size for smaller lattices and for higher value of infection parameter p, as presented in Figure 7.4.

For simulations in which the network size restricts epidemic spreading, the entropy is high as the realizations from different sources are almost identical. As the lattice grows and epidemic is less restricted by the network size, entropy distribution takes expected mean-decreasing form, as presented in Figure 7.5.

On the other hand, the accuracy is lower for epidemic-restricting network sizes and parameters, as presented in Figure 7.6. The biggest impact on accuracy reveals itself in the intermediate entropy region. The lower accuracy for the lower values of p – the ones for which the epidemic is still not restricted by the lattice size – can be explained by increasing symmetry of epidemic snapshots for smaller lattices, i.e. there tends to be serveral potential source nodes with the same MAP probability.

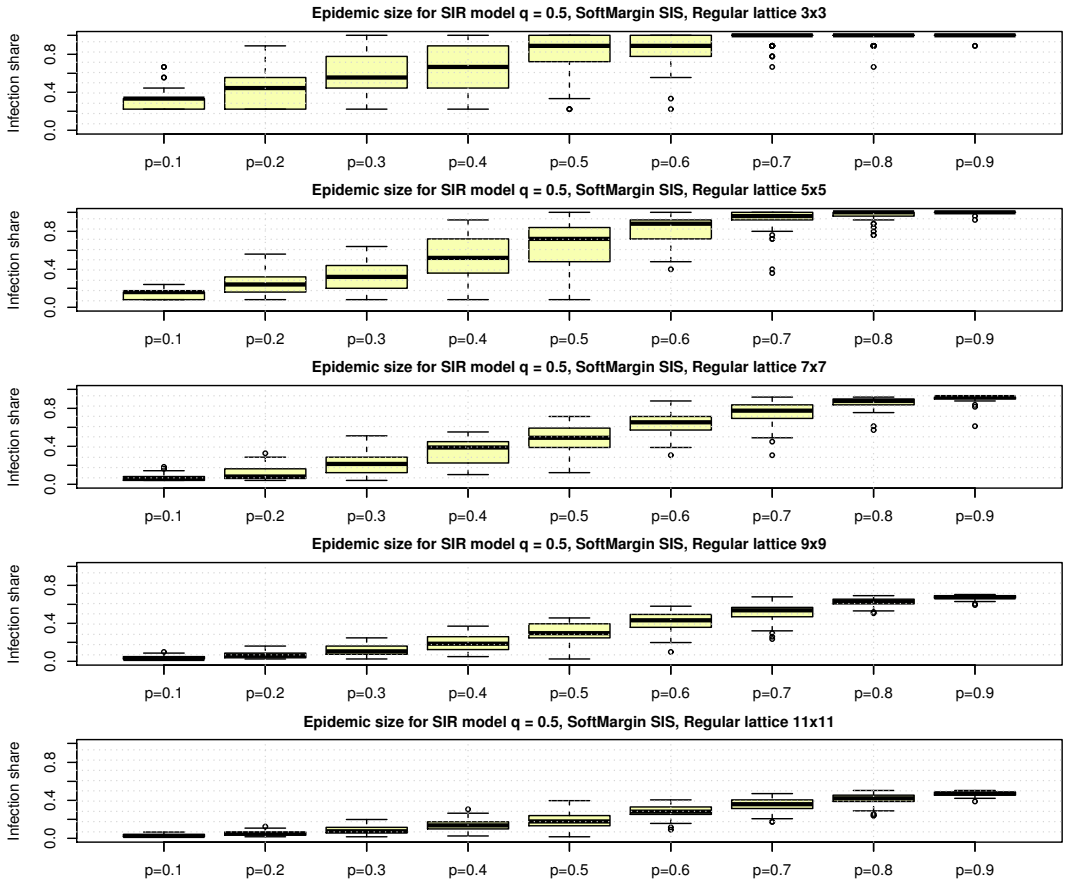

Figure 7.4: Box plots of size of epidemics simulated on 4-connected lattices of different sizes under SIR model with different parameters p in range $0.1 - 0.9$, fixed $q = 0.5$ and $T = 5$.

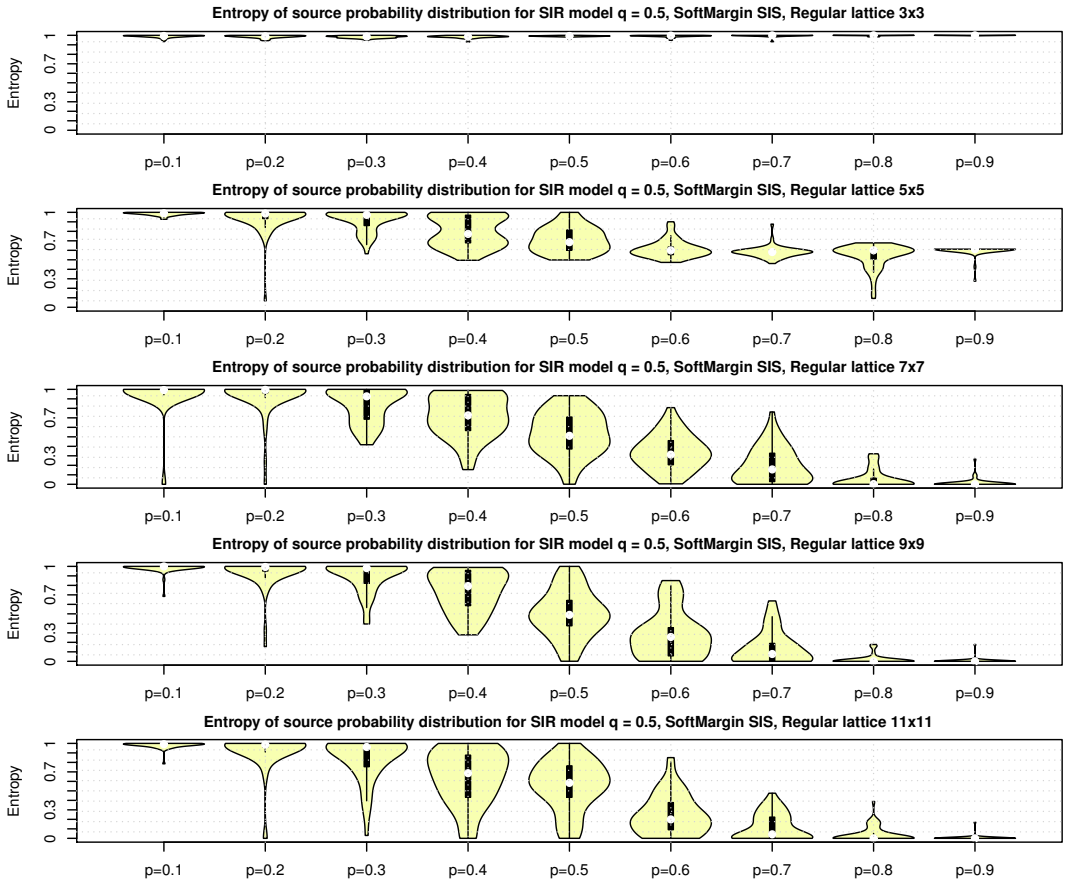

Figure 7.5: Violin plots of estimated entropy distribution for source probability distributions on 4-connected lattices of different sizes estimated with Soft Margin SIS method with $10^4 - 10^6$ samples and fixed $a = \frac{1}{2^5}$ under SIR model with different parameters p in range $0.1 - 0.9$, fixed $q = 0.5$ and $T = 5$.

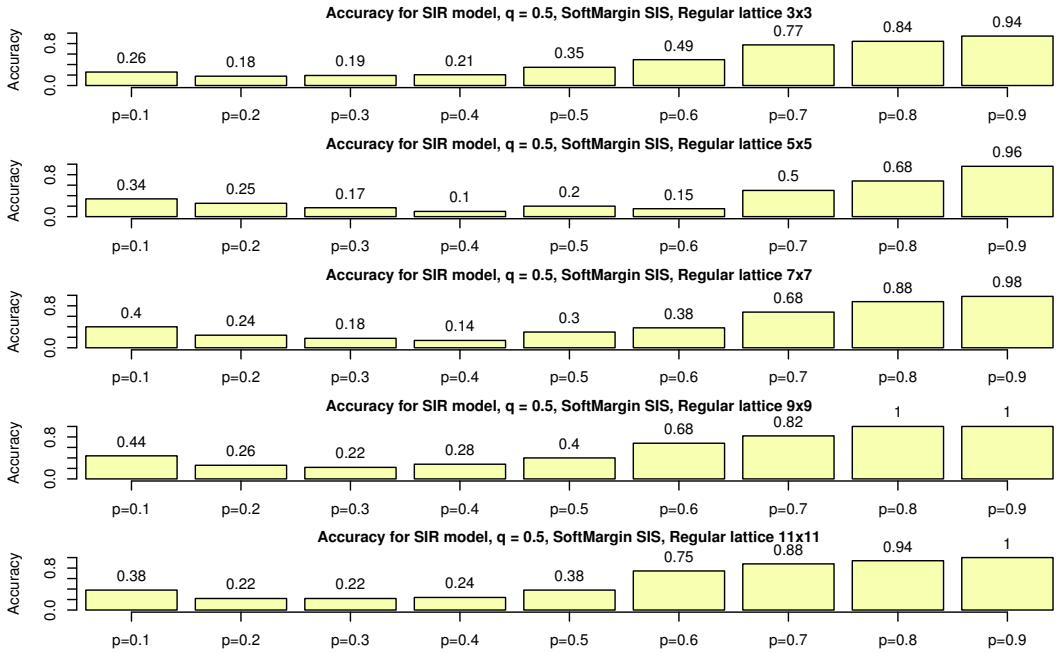

Figure 7.6: Accuracy of source MAP estimation on 4-connected lattice of different sizes estimated with Soft Margin SIS method with $10^4 - 10^6$ samples and fixed $a = \frac{1}{2^5}$ under SIR model with different parameters p in range $0.1 - 0.9$, fixed $q = 0.5$ and $T = 5$.

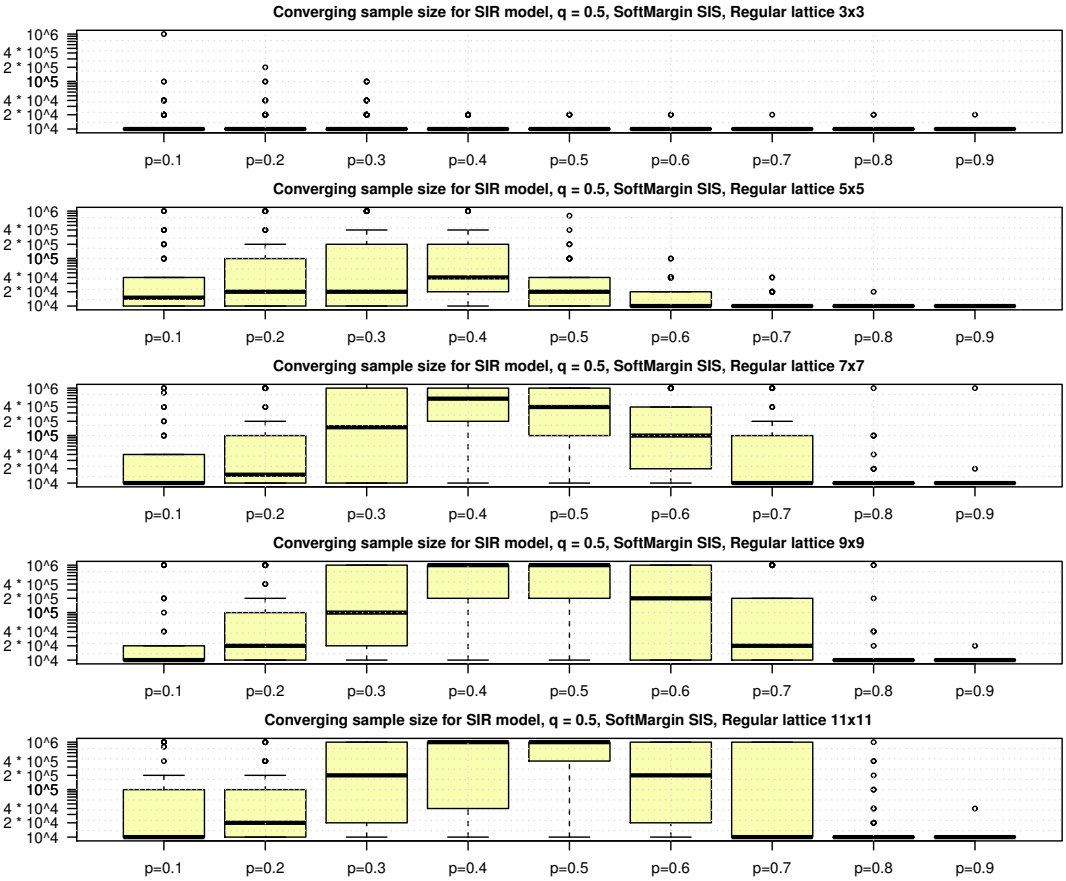

Figure 7.7: Box plots of converging samples distribution on 4-connected lattice of different sizes estimated with Soft Margin SIS method with $10^4 - 10^6$ samples and fixed $a = \frac{1}{2^5}$ under SIR model with different parameters p in range $0.1 - 0.9$, fixed $q = 0.5$ and $T = 5$.

7.2. Detectability based on parameters of the ISS model

To show how detectability presents itself in the parametric space of ISS parameters a and b several simulations were ran using Soft Margin detector with adaptive Soft Margin parameter a chosen from $\{\frac{1}{2^3}, \frac{1}{2^4}, \ldots, \frac{1}{2^9}\}$.

For each set of ISS parameter pairs (a, b) 50 simulations of ISS rumour spreading were conducted for $T = 5$ time steps starting from the central node in $4-$connected lattice network of 30×30 nodes. All simulations of size 1 were excluded from analysis. Soft Margin detector was ran on each rumour spreading simulation with converging conditions 4.7, $c = 0.05$. Estimations were conducted based on number of ISS simulations in range $n \in [10^4, 10^6]$.

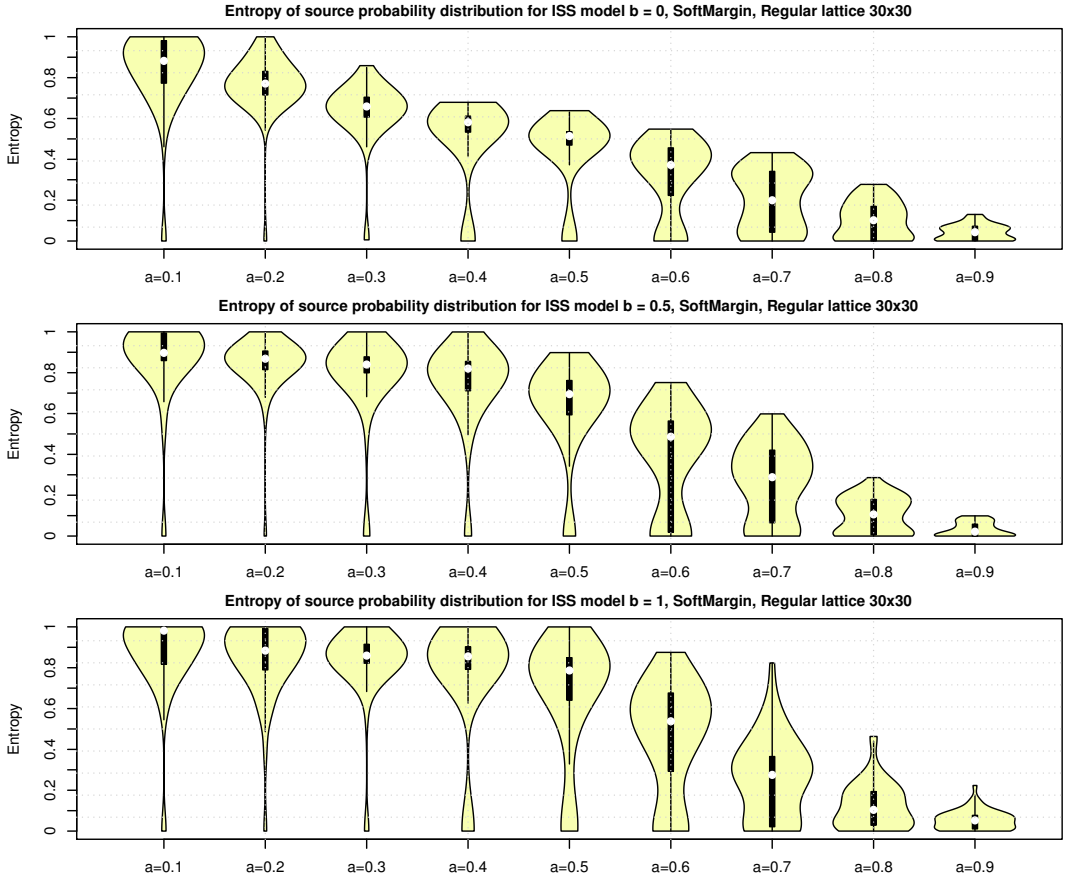

Figure 7.8: Violin plots of estimated entropy distribution for source probability distributions on 4-connected lattice 30×30 estimated with Soft Margin method with $10^4 - 10^6$ simulations and adaptive a chosen from $\{1/2^3, 1/2^4, \ldots, 1/2^9\}$ under ISS model with different parameters a in range $0.1 - 0.9, b = \{0, 0, 5, 1\}$ and $T = 5$.

Entropical detectability for the rumour spreading ISS model presented in Figure 7.8 shows behaviour similar to entropical detectability under epidemic SIR model pre-

sented in Figure 7.1. In the model with no rumour decay where $b = 0$, ISS model is equivalent to SIR model, as are their entropy distributions. The 3 entropy-detectability regions distinguishable in the entropy distribution over parameter p for the SIR model can be distinguished in the entropy distribution over parameter a for the ISS model.

However, as the value of parameter b in the ISS model compared to the same value of parameter q in the SIR model represents more aggressive form of recovery, the low detectability - high entropy region grows more rapidly with higher value of b, taking over the range of parameters $a < 0.4$ and $a < 0.5$ for $b = 0.5$ and $b = 1.0$, respectively.

On the other hand, accuracy of source detection for the ISS model remains stable for the same value of parameter a and different values of parameter b as presented in Figure 7.9, similarly to the same accuracy for source detection under SIR model presented in Figure 7.2. For the same value of parameter b, accuracy grows with higher value of parameter a.

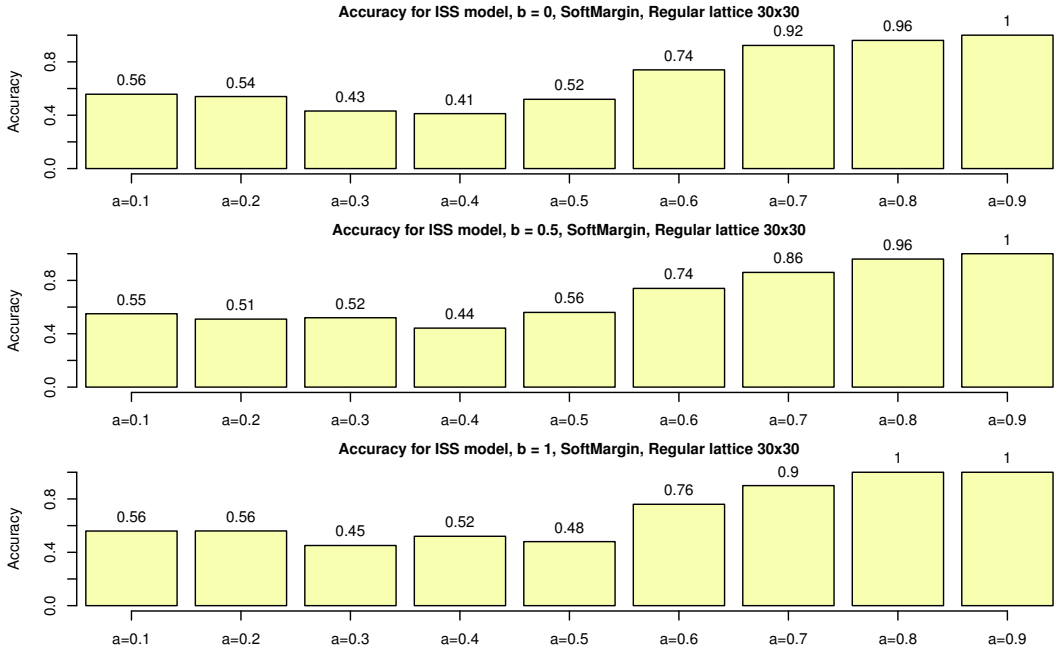

Figure 7.9: Accuracy of source MAP estimation on 4-connected lattice 30×30 estimated with Soft Margin method with $10^4 - 10^6$ simulations and adaptive a chosen from $\{1/2^3, 1/2^4, \ldots, 1/2^9\}$ under ISS model with different parameters a in range $0.1 - 0.9, b = \{0, 0.5, 1\}$ and $T = 5$.

7.3. Detectability of patient zero based on its position in the network

To analyse how detectability changes across node features of the source node we will examine patient zero problem on two different network topologies: Erdős-Rényi and Barabási-Albert graphs. These network topologies differ in distribution of centrality measures of its nodes so epidemic spreads differently across each network.

Two graph datasets were generated for this purpose. The Erdős-Rényi dataset consists of 50 graphs with $N = 100$ nodes generated with $p = 0.01$. This probability is the threshold of emergence of the giant component for a graph with $N = 100$ nodes and the dataset contains only connected graphs. The second dataset consists of 50 graphs with $N = 100$ nodes generated as Barabási-Albert graphs with $m = 2$ attaching edges.

On the generated network impact of centrality measures of the source node on source detectability will be analysed. Centrality measures include degree, closeness, betweenness, eigenvector centrality and coreness.

7.3.1. Erdős-Rényi graph

Before analysing how detectability changes for source nodes with different topological properties, let's examine the distribution of these properties on generated Erdős-Rényi $N = 100$, $p = 0.01$ graphs.

In Table 7.1 summary of statistics for each centrality measure is presented. The distribution of frequencies for each measure is presented in Figure 7.10. Degree distribution takes the form of binomial distribution as expected. The centralities are positively correlated.

In generated Erdős-Rényi graphs, expected degree and betweenness of a node are relatively small compared to the network size.

	Degree	**Closeness**	**Betweenness**	**Eigenvector centrality**	**Coreness**
Min	1	0.1713	0.00	0.001575	1
Median	5	0.3246	83.33	0.334041	3
Mean	4.657	0.3222	106.14	0.367168	2.831
Max	14	0.4304	698.57	1.000000	4

Table 7.1: Summary of cumulative statistics of distributions for degree, closeness, betweenness, eigenvector centrality and coreness of the nodes in 50 generated Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes.

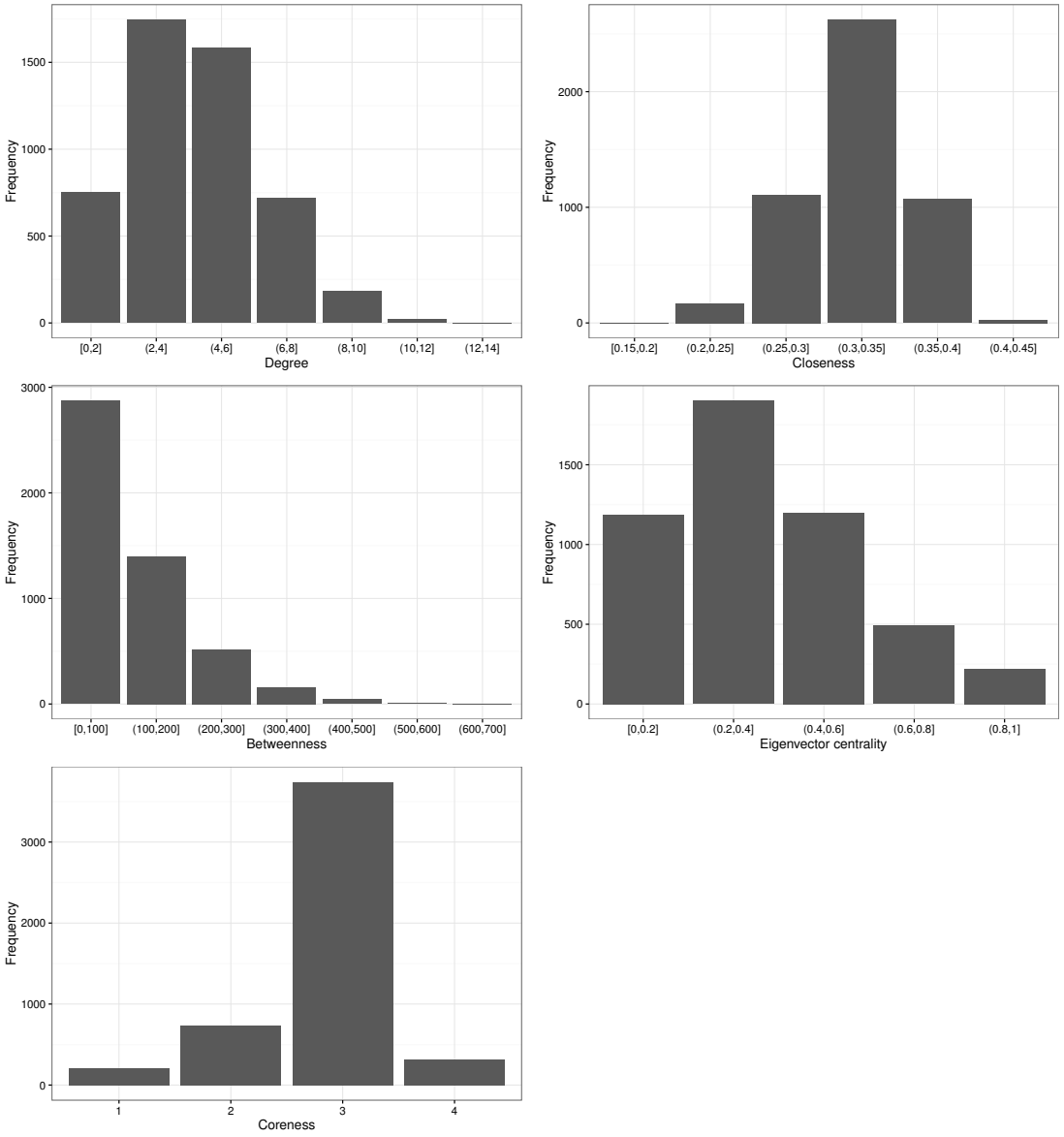

Figure 7.10: Frequencies of degree, closeness, betweenness, eigenvector and coreness centralities on 50 generated Erdős-Rényi graphs with $N = 100$ and $p = 0.01$.

Degree centrality

The epidemics simulated on Erdős-Rényi graphs infect more nodes when started from a node with higher degree as presented in Figure 7.11. Epidemics simulated with high infection rate $p = 0.7$ in classes of SIR parameters C and D were highly restricted by the network size and we can already predict that will have impact on detectability. As expected, high recovery rate $q = 0.7$ in class B limits the size of epidemic compared to epidemics with SIR parameters in class A and low recovery rate $q = 0.3$.

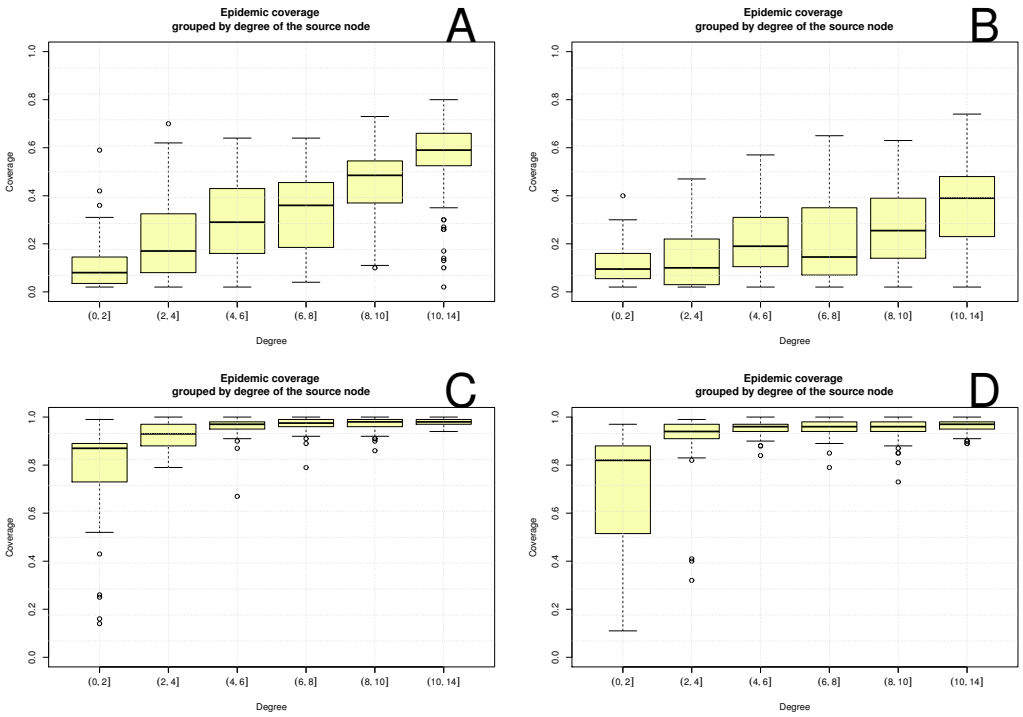

Figure 7.11: Box plots of size of epidemics simulated on on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by degree of the source node.

The impact of restricting network size for parameter classes C and D is presented in entropy distributions with high expectation as presented in Figure 7.12. For classes of parameters A and B, the entropy is also high and difference in degree of the source node does not significantly alter its expected value. Additionally, the epidemics simulated with higher recovery rate (classes B and D) exibit higher expected entropy compared to their low recovery counterparts (classes A and C, respectively).

The source detection accuracy is lower for source nodes that have higher degree, as presented Figure 7.13. The degree of the source node is more restrictive for accuracy on epidemics simulated from classes of parameters C and D.

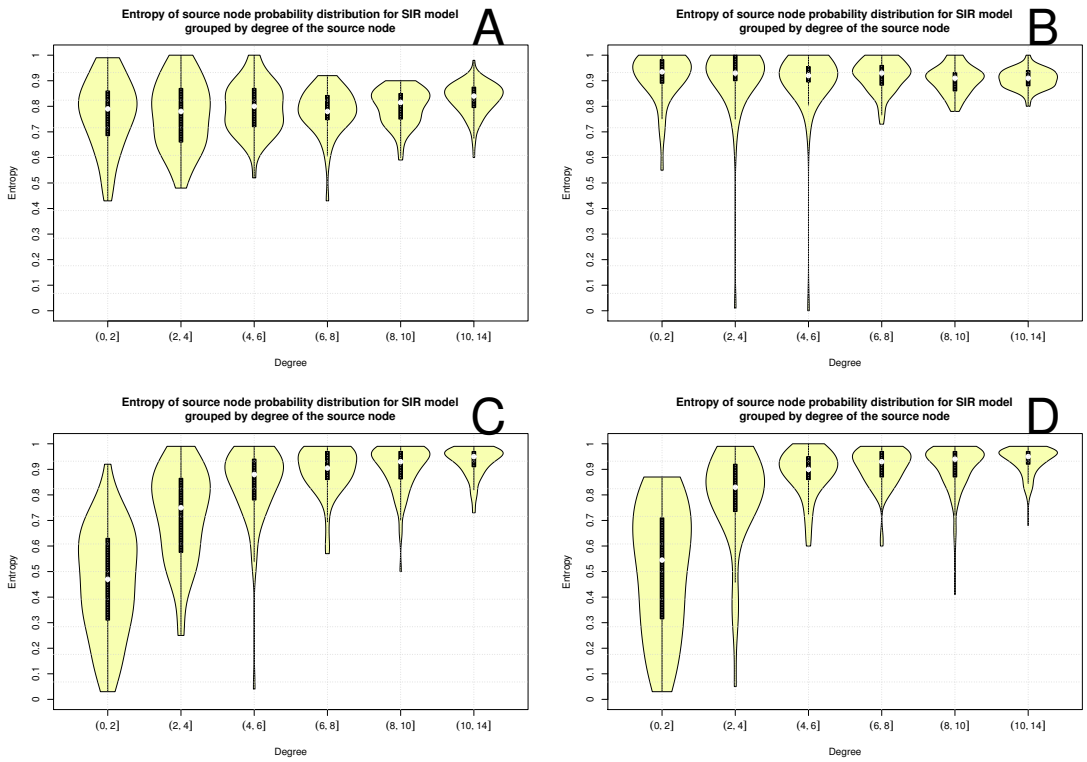

Figure 7.12: Violin plots of estimated entropy distribution for estimated source probability distribution on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by degree of the source node.

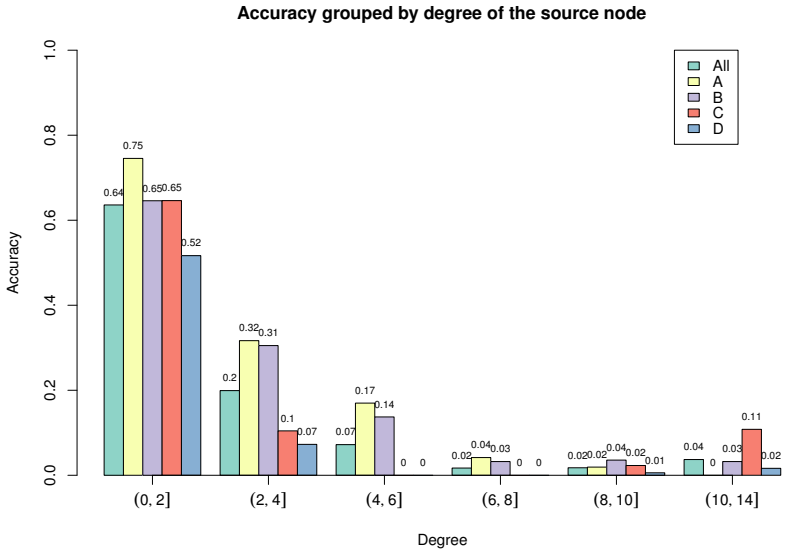

Figure 7.13: Source detection accuracy on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by degree of the source node.

The converging number of samples required for estimations to converge on Erdős-Rényi graphs is on average in range $n \in [4 \cdot 10^4, 4 \cdot 10^5]$. For classes of SIR parameters A and B the expected number of simulations does not differ significantly for different degree of the source node, as presented in Figure 7.14. For classes of SIR parameters C and D, required number of samples is in range $n \in [10^4, 10^5]$. For largest values of degree of the source node, the required number of converging samples for classes of parameters C and D is minimal.

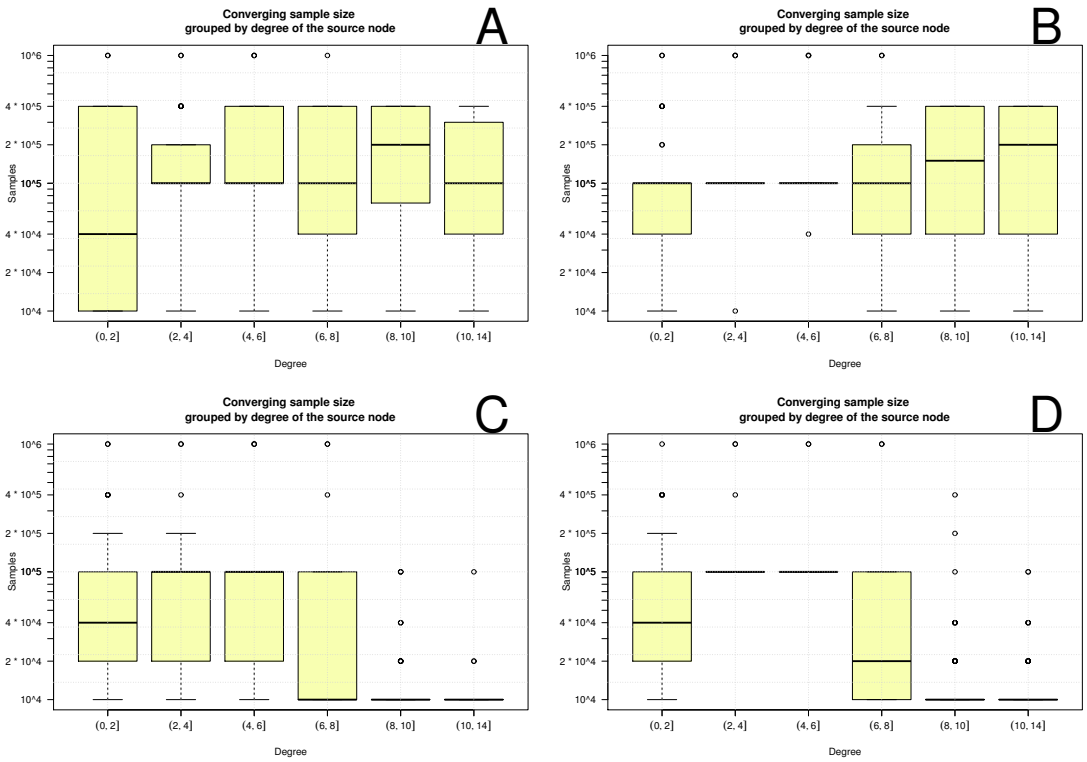

Figure 7.14: Box plots of converging samples distribution for source MAP estimations on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by degree of the source node.

Closeness centrality

Closeness centrality corresponds to how close a given node is to any other node, as defined in 2.1. For Erdős-Rényi graph, the nodes with higher closeness usually have a higher degree compared to the nodes of lower closeness so the results for degree and closeness will be similar.

Size of simulated epidemics grows with higher closeness, as presented in Figure

7.15.

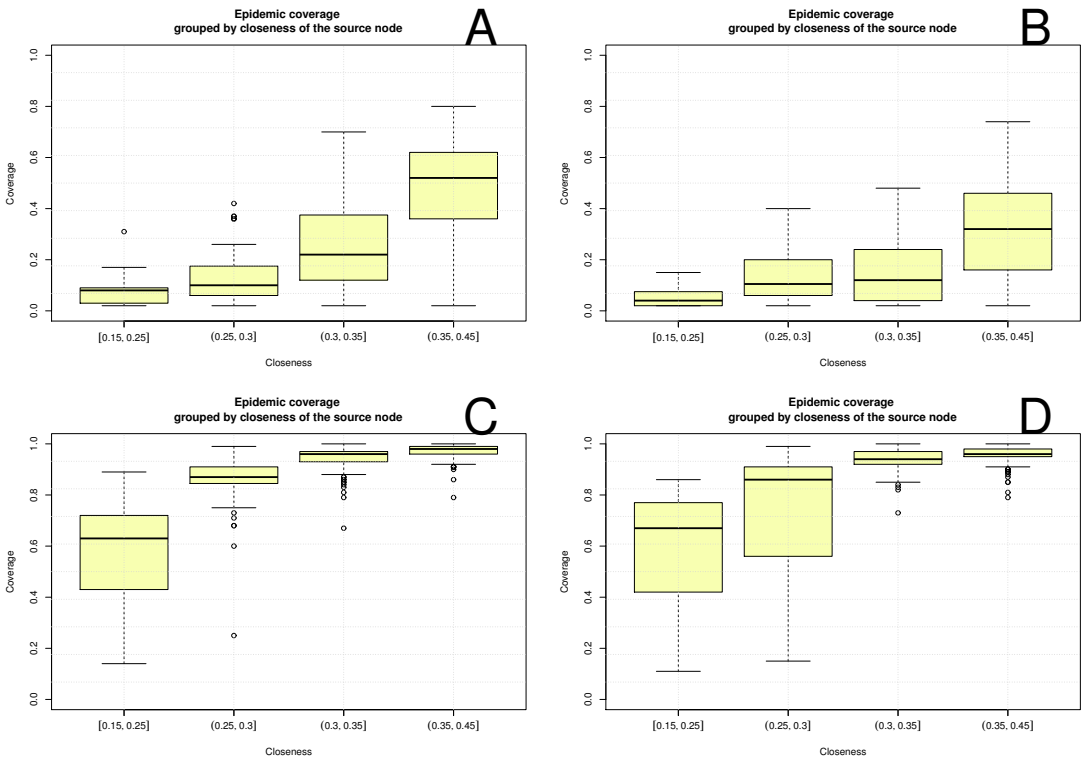

Figure 7.15: Box plots of size of epidemics simulated on on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by closeness of the source node.

Entropy distribution over different values of closeness of the source node stays constant and high in expectation for classes of SIR parameters A and B. For classes of SIR parameters C i D that correspond to high detectability - low entropy region for detection on the grid, entropy is low for low values of closeness and grows higher when closeness is high, as presented in Figure 7.16.

Accuracy is also high for lower values of closeness, as presented in Figure 7.17. Note the closeness in range $[0.15, 0.3)$ mostly corresponds to nodes with degree equal to 1 and 2 (Figure 7.10) on which we've already seen high accuracy of source detection (Figure 7.13). Accuracy is more restricted by closeness for classes of SIR parameters C and D.

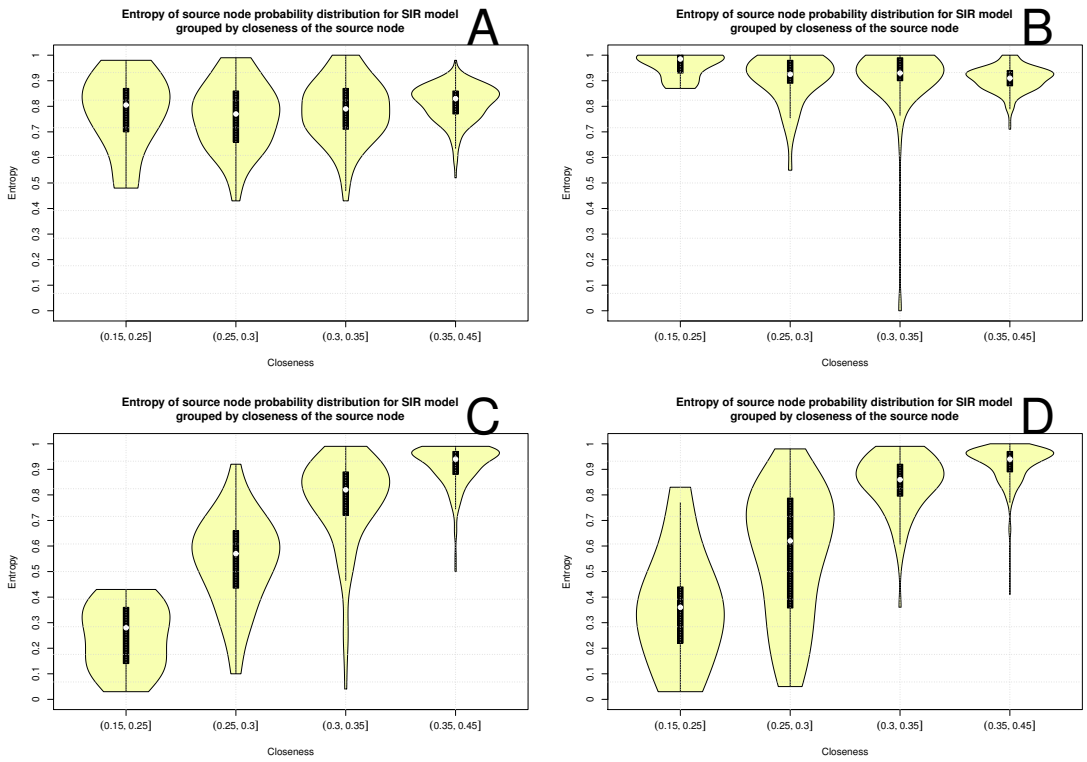

Figure 7.16: Violin plots of estimated entropy distribution for estimated source probability distribution on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by closeness of the source node.

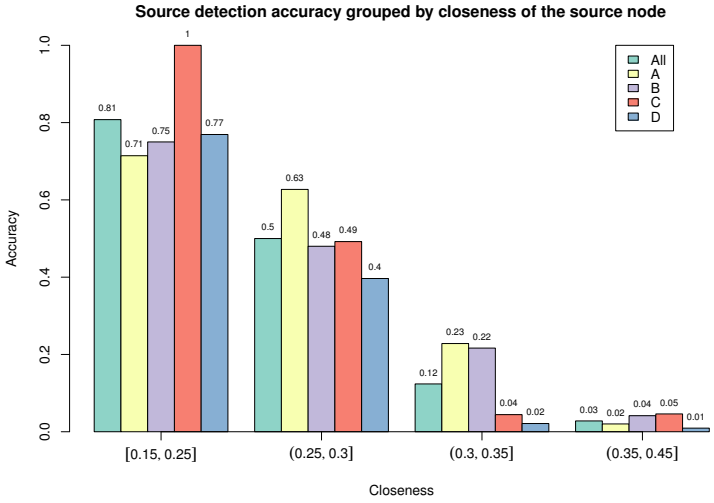

Figure 7.17: Source detection accuracy on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by closeness of the source node.

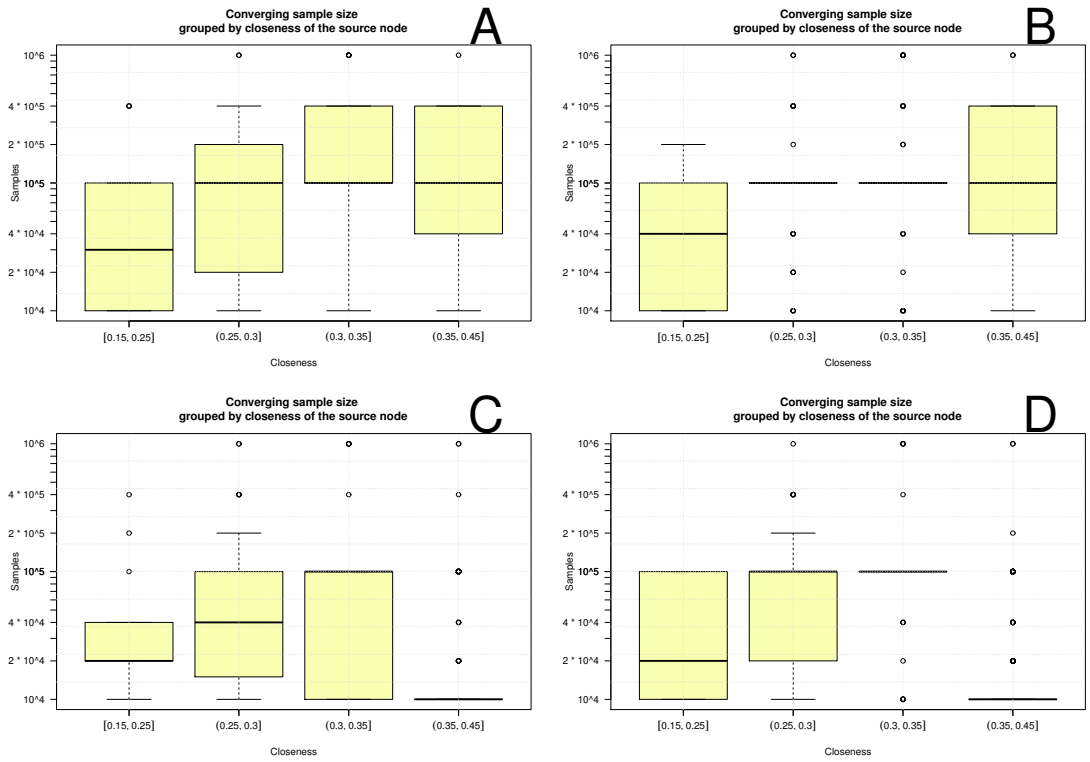

Figure 7.18: Box plots of converging samples distribution for source MAP estimations on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by closeness of the source node.

Betweenness centrality

Betweenness centrality describes how well situated a vertex is in terms of the paths it lies on, as defined in 2.2. Betweenness positively correlates with degree and closeness for Erdős-Rényi graph, so detectability results grouped by betweenness of the source node are similar to the ones for degree and closeness. Entropy distribution, accuracy, and distribution of converging number of samples grouped by betweenness of the source node are presented in Figure 7.19, 7.20 and 7.21, respectively.

Detection accuracy is lower for source nodes with higher betweenness and it gets more restrictive for epidemics based on SIR parameters in classes C and D.

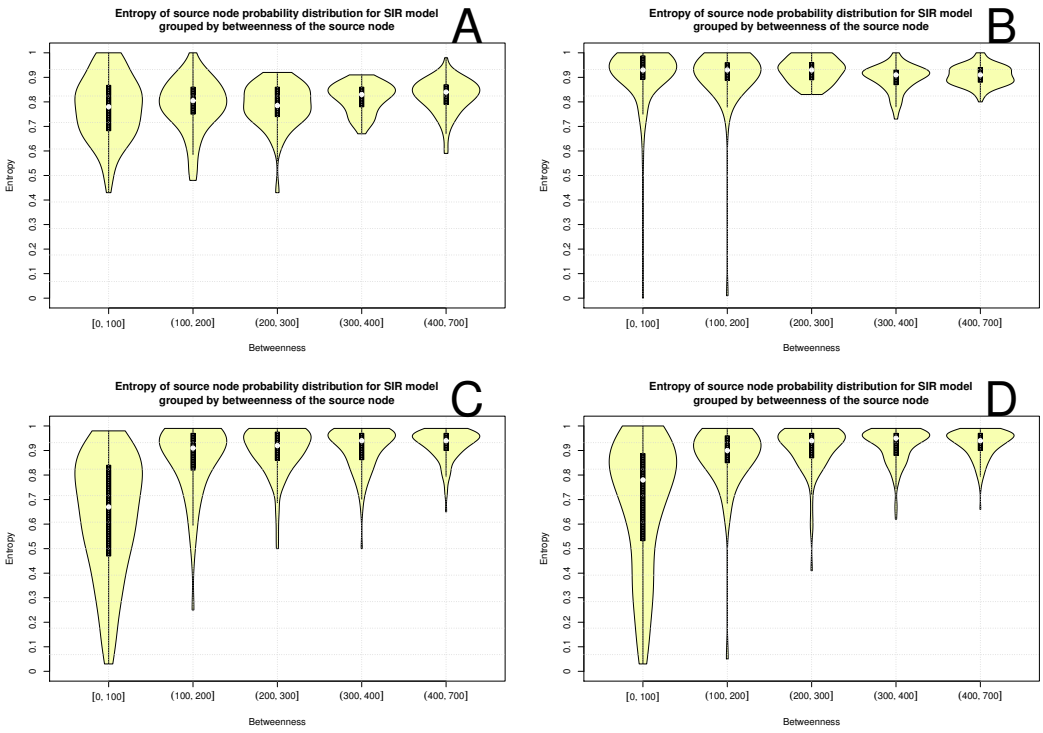

Figure 7.19: Violin plots of estimated entropy distribution for estimated source probability distribution on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by betwenness of the source node.

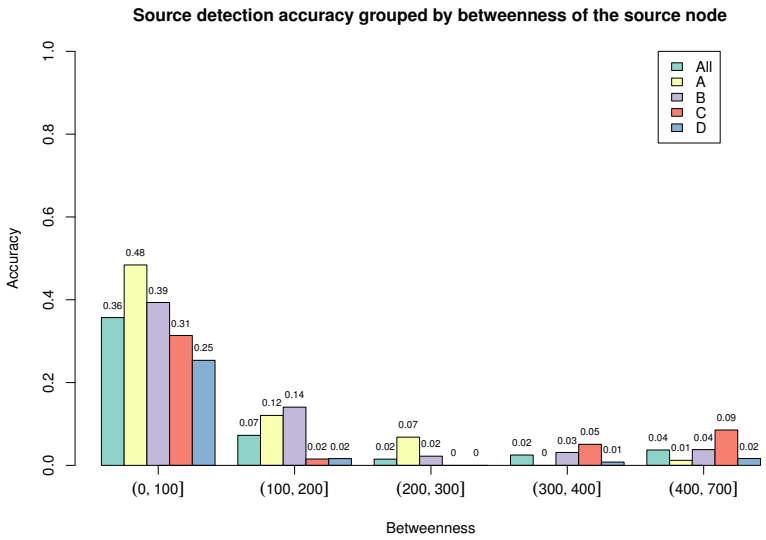

Figure 7.20: Source detection accuracy on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by betweenness of the source node.

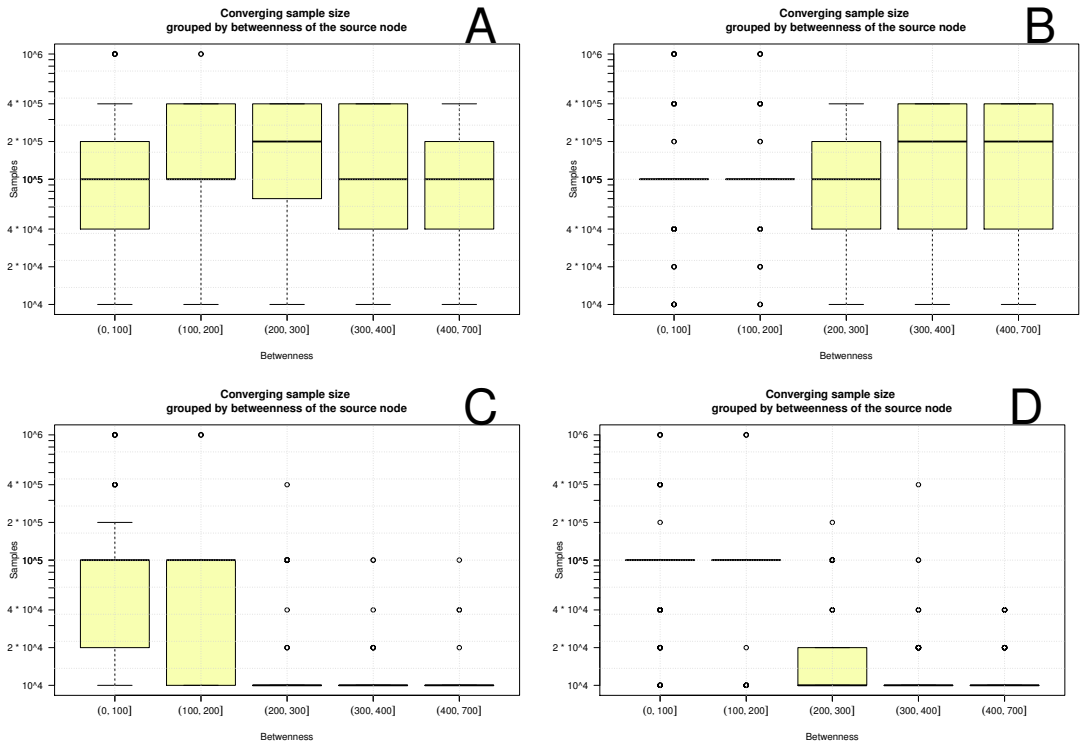

Figure 7.21: Box plots of converging samples distribution for source MAP estimations on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by betweenness of the source node.

Eigenvector centrality

Eigenvector centrality describes centrality of the source node in terms of node centrality of its neighbours, as defined in 2.3. Eigenvector centrality positively correlates with degree, closeness and betweenness for Erdős-Rényi graph as presented in Figure 7.10, so detectability results grouped by eigenvector centrality of the source node are similar to the ones for degree, closeness and betweenness. Entropy distribution, accuracy, and distribution of converging number of samples grouped by eigenvector centrality of the source node are presented in Figure 7.22, 7.23 and 7.24, respectively.

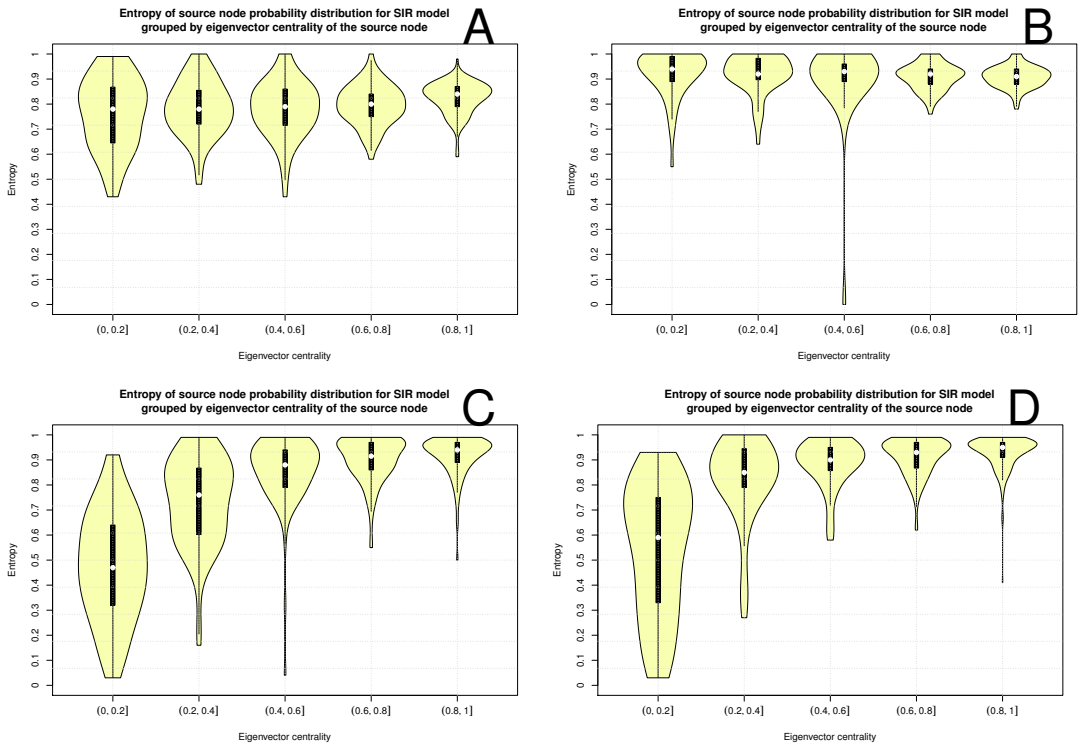

Figure 7.22: Violin plots of estimated entropy distribution for estimated source probability distribution on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by eigenvector centrality of the source node.

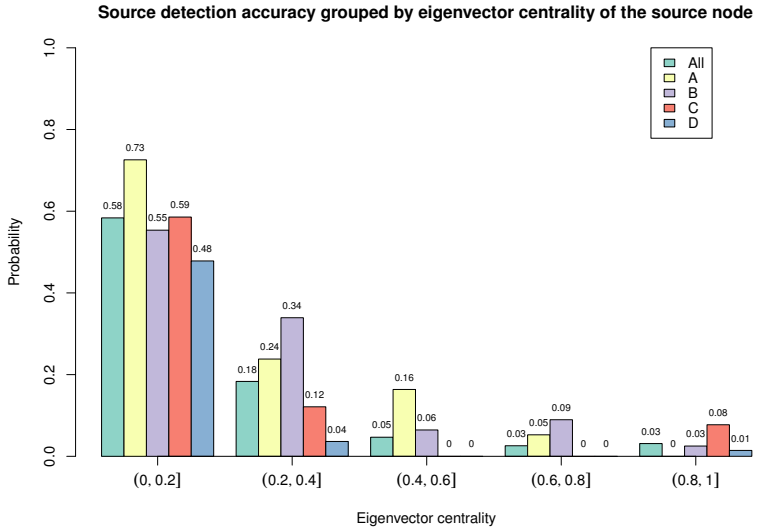

Figure 7.23: Source detection accuracy on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by eigenvector centrality of the source node.

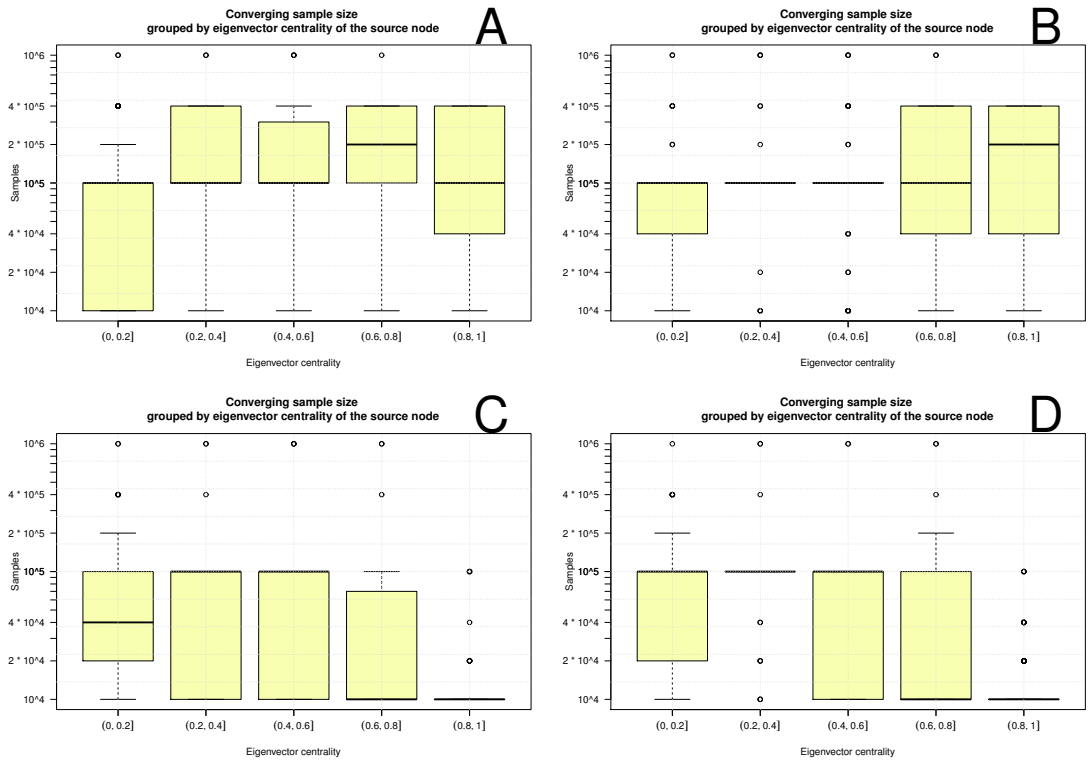

Figure 7.24: Box plots of converging samples distribution for source MAP estimations on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by eigenvector centrality of the source node.

Coreness

Coreness of the source node also positively correlates with degree, closenes, betweenness and eigenvector centrality for Erdős-Rényi graph as presented in Figure 7.10, so detectability results grouped by coreness of the source node are similar to the ones for other measures. Entropy distribution, accuracy, and distribution of converging number of samples grouped by eigenvector centrality of the source node are presented in Figure 7.25, 7.26 and 7.27, respectively.

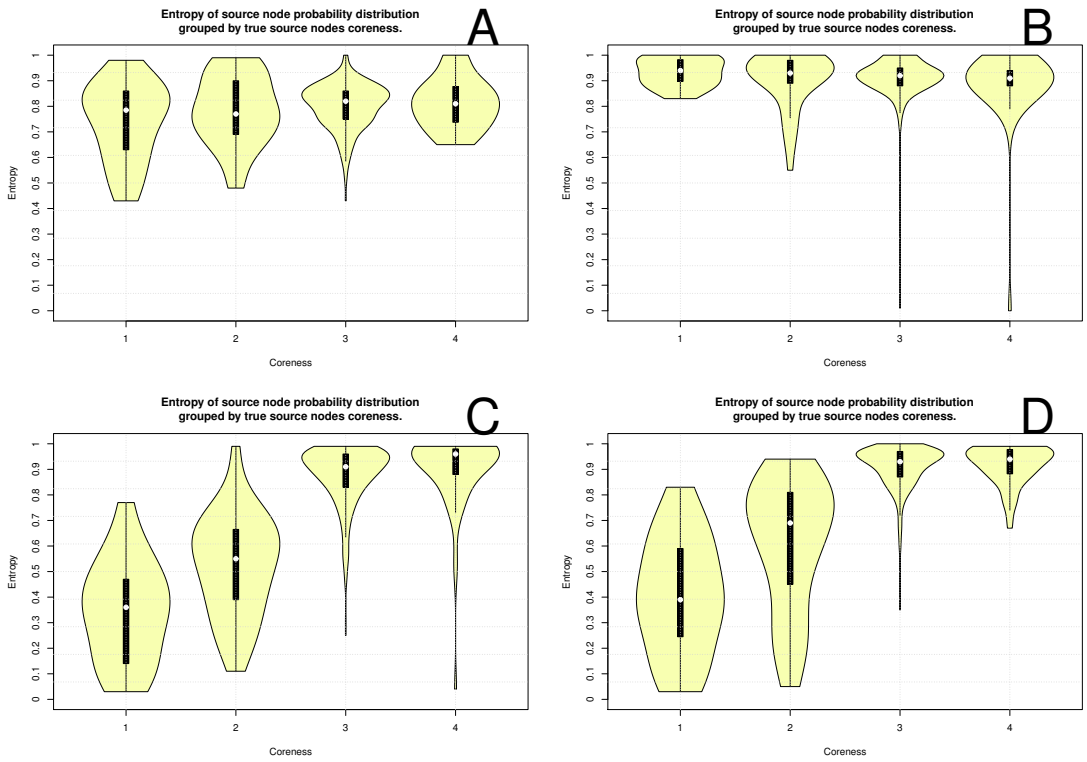

Figure 7.25: Violin plots of estimated entropy distribution for estimated source probability distribution on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by coreness of the source node.

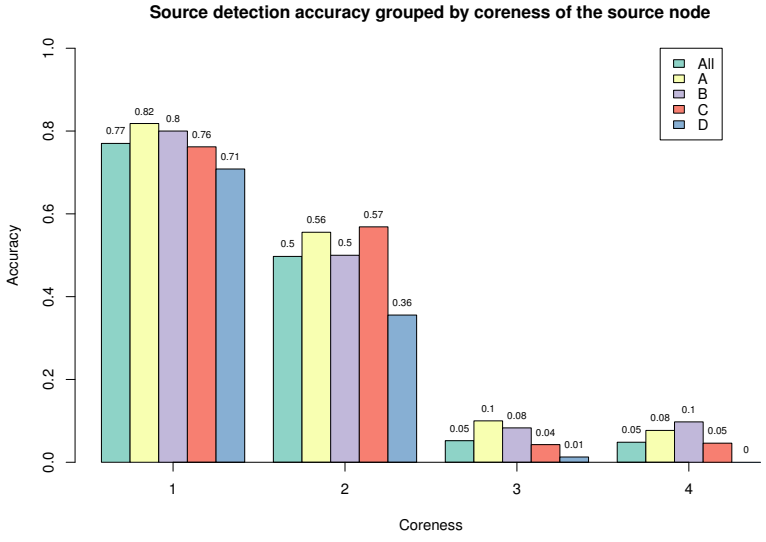

Figure 7.26: Source detection accuracy on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by coreness of the source node.

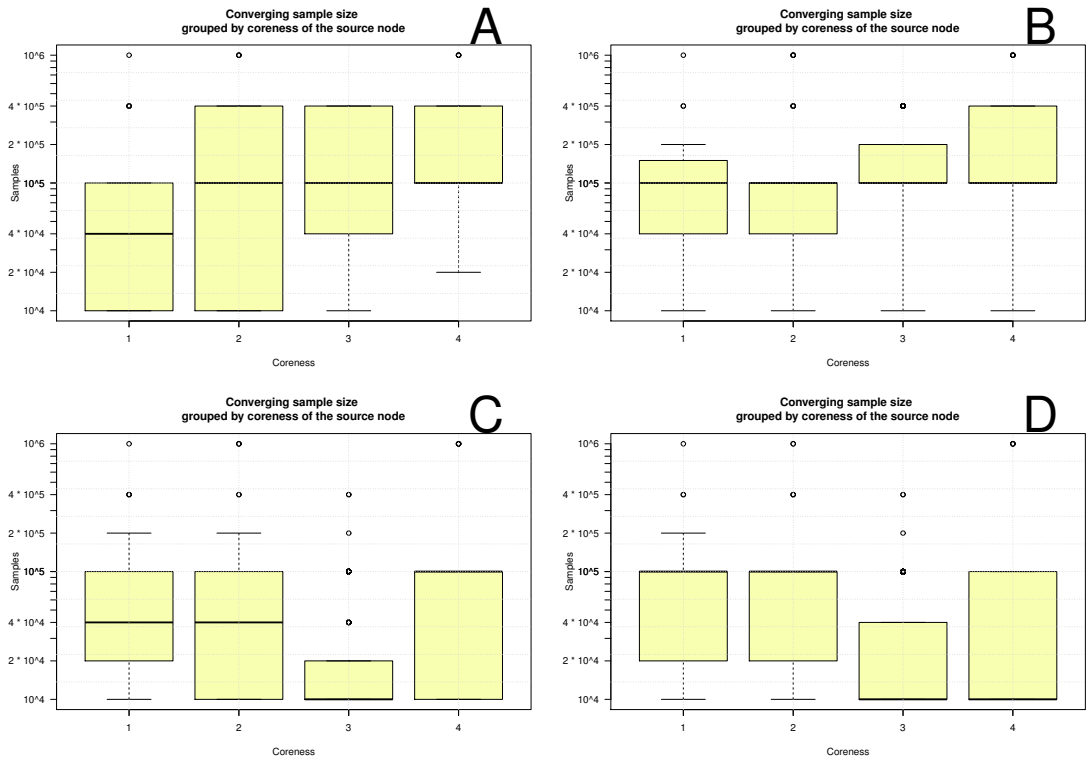

Figure 7.27: Box plots of converging samples distribution for source MAP estimations on Erdős-Rényi connected graphs with $p = 0.01$ and $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by coreness of the source node.

7.3.2. Barabási-Albert graph

Barabási-Albert graph is generated using preferential attachment property. We will use a set of Barabási-Albert graphs wtih $n = 100$ nodes that were evolved with $m = 2$ attaching edges for each added node. For these graphs coreness is constant and equal to m so it will be omitted from analysis.

The summary of statistics for centrality measures on generated $2-$ Barabási-Albert graphs is presented in Table 7.2. Apart from constant coreness and compared to distribution of the same measures on Erdős-Rényi graphs of the same size, degree, closeness and betweenness can take higher values.

	Degree	Closeness	Betweenness	Eigenvector centrality	Coreness
Min	2	0.2421	0	0.003098	2
Median	2	0.3808	5.924	0.102812	2
Mean	3.96	0.3823	82.952	0.116188	2
Max	108	0.8250	4517.507	1.000000	2

Table 7.2: Summary of cummulative statistics of distributions for degree, closeness, betweenness, eigenvector centrality and coreness of the nodes in 50 generated Barabási-Albert graphs with $m = 2$ and $N = 100$ nodes.

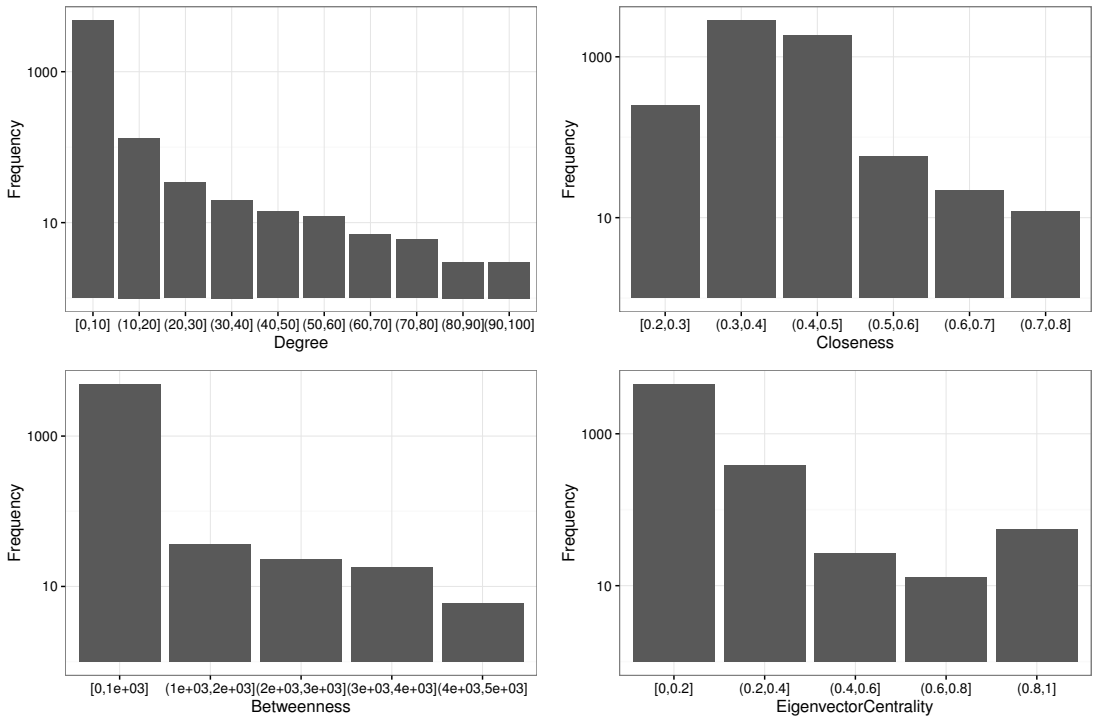

Figure 7.28: Distribution of degree, closeness, betweenness and eigenvector centrality in 2-Barabási-Albert dataset.

Additionally, degree distribution is scale-free, as expected, while distributions of other centrality measures take similar form as presented in Figure 7.28. Ranges of values for each centrality measure upon which the detectability is analysed has been chosen according to these distributions. Between each pair of centrality measures the correlation is mostly positive.

Degree centrality

Epidemic simulations on 2- Barabási-Albert graphs are restricted by the network size and are expected to infect all nodes in the network for classes of SIR parameters A, C, D as presented in Figure 7.29. The structure of 2- Barabási-Albert graphs, precisely their short average path length, help in epidemic spreading.

For SIR parameters $B = (p = 0.3, q = 0.3)$ expected size of epidemic positively correlates with degree of the source node. The epidemics simulated with SIR parameters with lower recovery rate (classes A and C) have higher expected epidemic size compared to the corresponding classes with higher recovery rate (classes B and D, respectively).

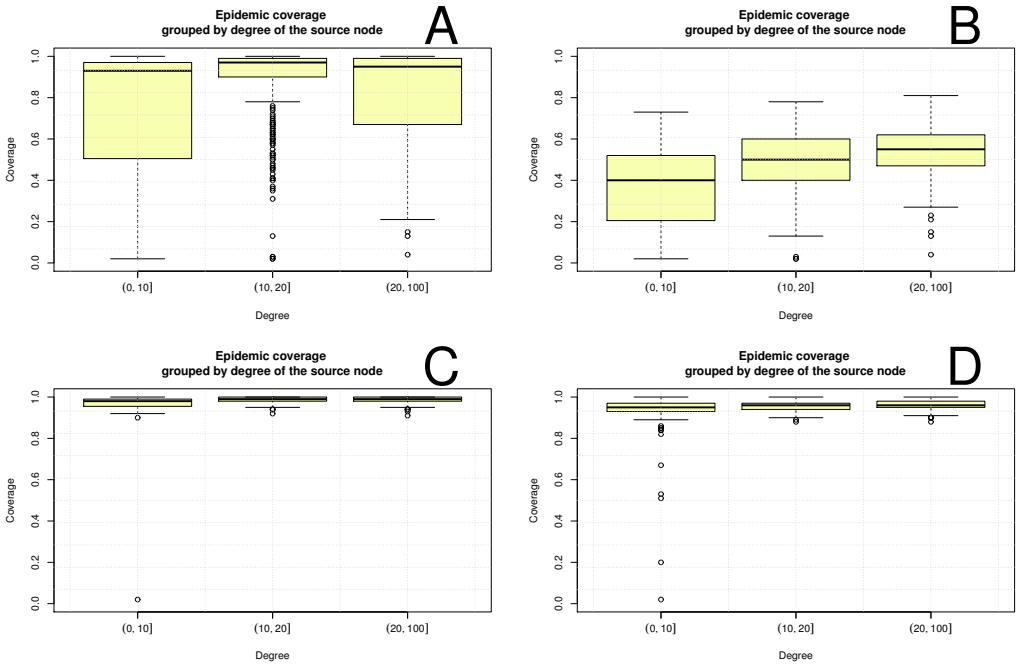

Figure 7.29: Box plots of size of epidemics simulated on 2-Barabási-Albert graphs with $N = 100$ nodes under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by degree of the source node.

The entropy distributions, on 2-Barabási-Albert graphs take high values for all SIR parameter classes, no matter the degree of the source node, as presented in Figure 7.31.

Moreover, for 2-Barabási-Albert graphs detection accuracy is generally low. Except for network size restricting epidemics with parameter class C, the accuracy is highest for detection for network non-restricting epidemics with SIR parameter class B and low degree source nodes, as presented in Figure 7.30.

The emergence of detectability for high degrees of the source node for source detection problems with SIR parameters in class C gives a hint the local network structure plays a role in detectability since usually the high degree nodes are the ones at the core of these locally connected groups. This property isn't noticeable in Erdős-Rényi graphs since they do note have with such local structure.

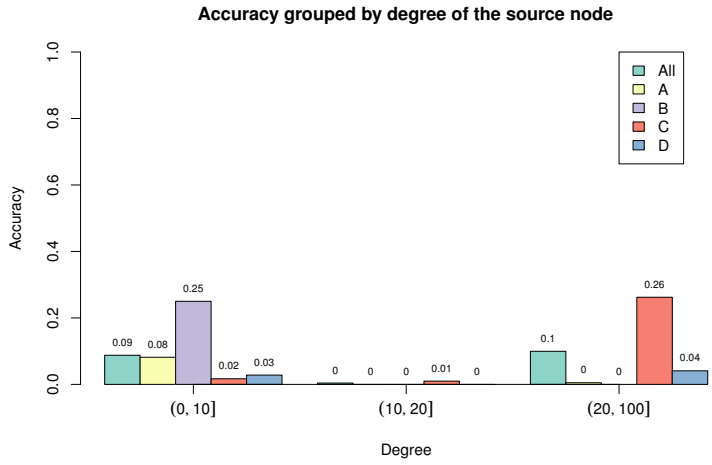

Figure 7.30: Source detection accuracy on 2-Barabási-Albert graphs with $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by degree of the source node.

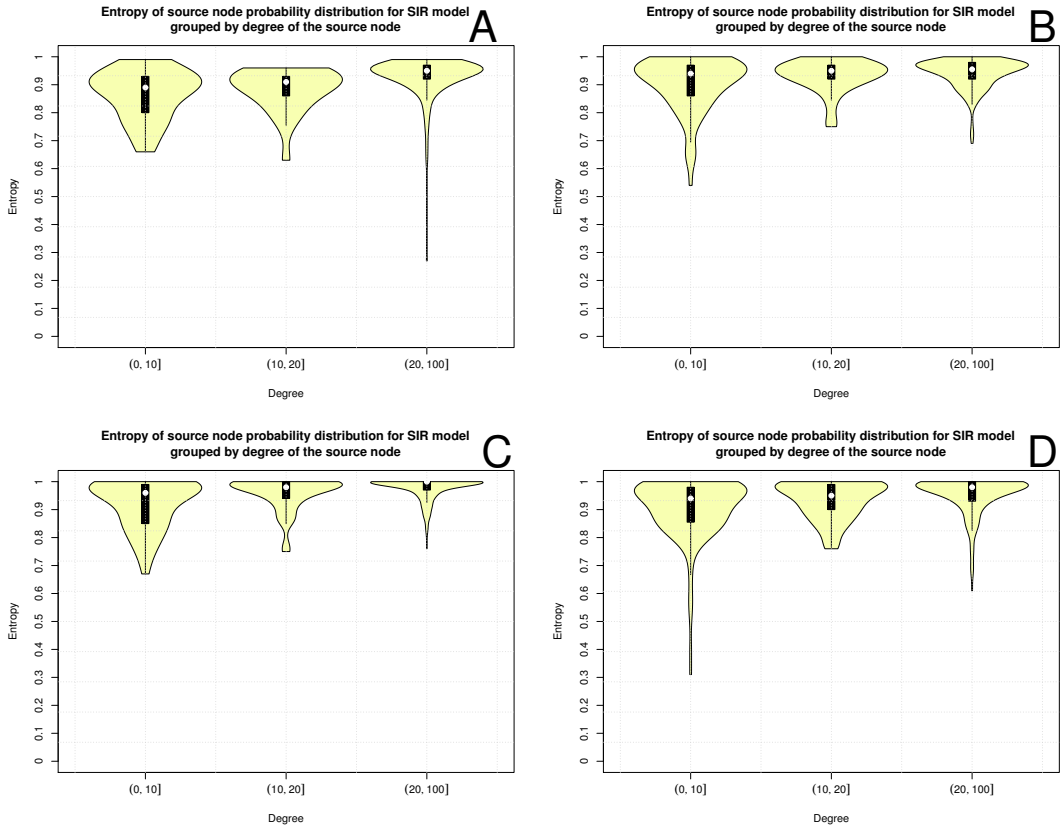

Figure 7.31: Violin plots of estimated entropy distribution for estimated source probability distribution on 2-Barabási-Albert graphs with $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by degree of the source node.

For 2-Barabási-Albert graphs converging number of samples negatively correlates with degree of the source node, as presented for classes of parameters A and D in Figure 7.32. Additionally, epidemics with higher recovery rate require more samples to converge compared to corresponding low recovery rate class pairs.

The expected size of required samples is in range $[4 \cdot 10^4, 10^6]$ for classes of SIR parameters A and B. For SIR parameter classes C and D, the converging number of samples is usually lower, as presented in Figure 7.32. It is worth mentioning the number of samples was upper bounded by 10^6 during detection process and it is expected even more samples are actually required to have higher detection accuracy for 2-Barabási-Albert graphs with classes of SIR parameters A and B.

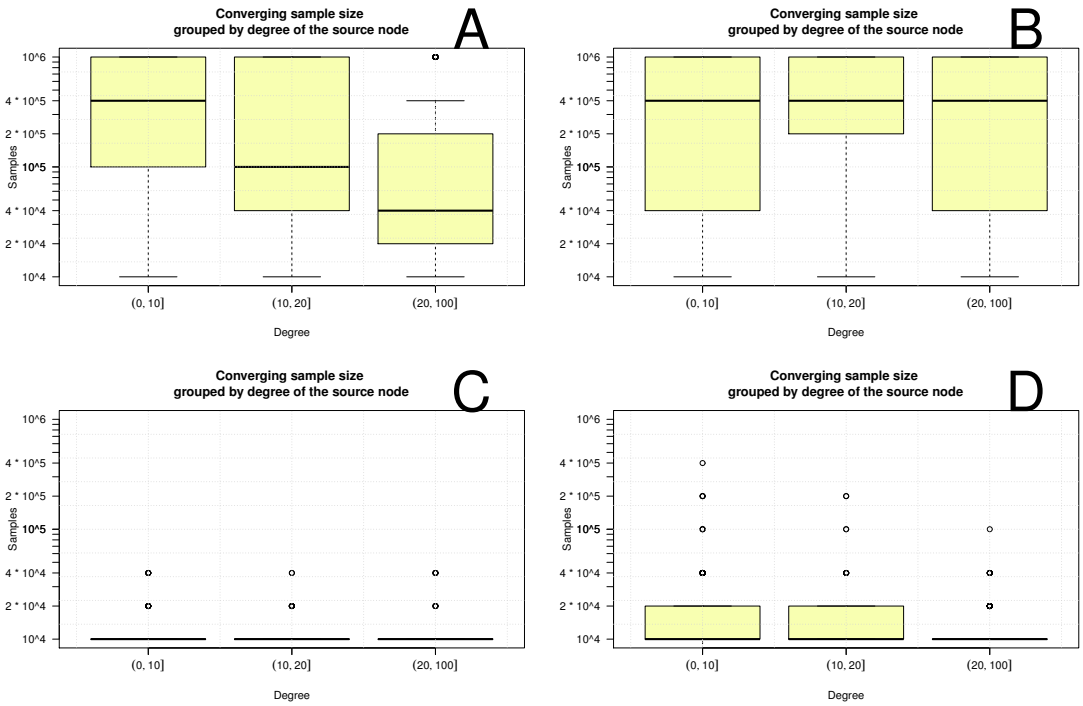

Figure 7.32: Box plots of converging samples distribution for source MAP estimations on 2-Barabási-Albert graphs with $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by degree of the source node.

Closeness centrality

Closeness corresponds to how close a given node is to any other node in the network, as defined in 2.1.

For classes of SIR parameters A and B, expected epidemic size positively correlates with closeness and the higher recovery rate limits the epidemic coverage, as presented in Figure 7.33.

Except accuracy on network size restricted simulations with SIR parameter class C, accuracy is highest for SIR parameter class A and B $(p = 0.3)$ when closeness of source node is low, as presented in Figure 7.34. The high accuracy for high values of closeness and SIR parameters in class C can be explained by local network structure.

Finally, it is interesting to see how the required converging number of samples for source estimation on SIR epidemic negatively correlates with higher closeness of the source nodes, as presented in Figure 7.35.

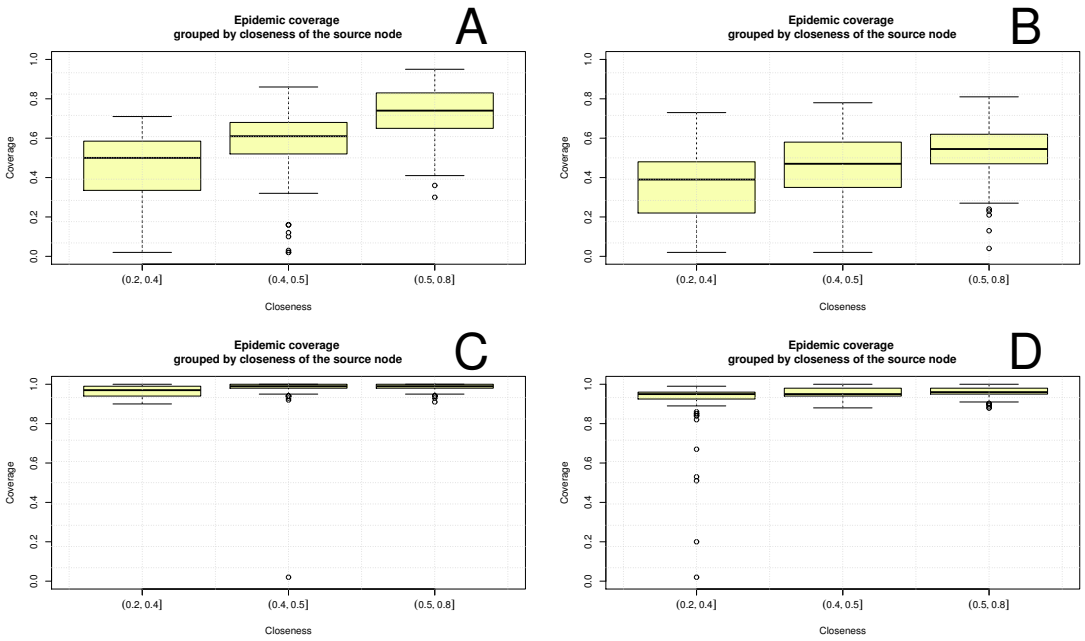

Figure 7.33: Box plots of size of epidemics simulated on 2-Barabási-Albert graphs with $N = 100$ nodes under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by closeness of the source node.

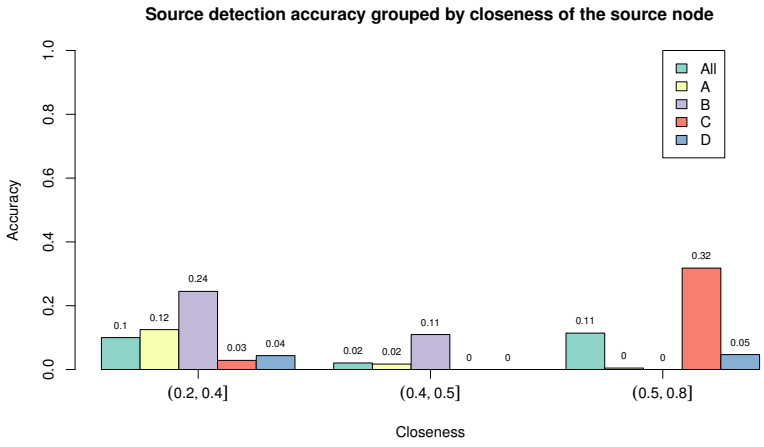

Figure 7.34: Source detection accuracy on 2-Barabási-Albert graphs with $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by closeness of the source node.

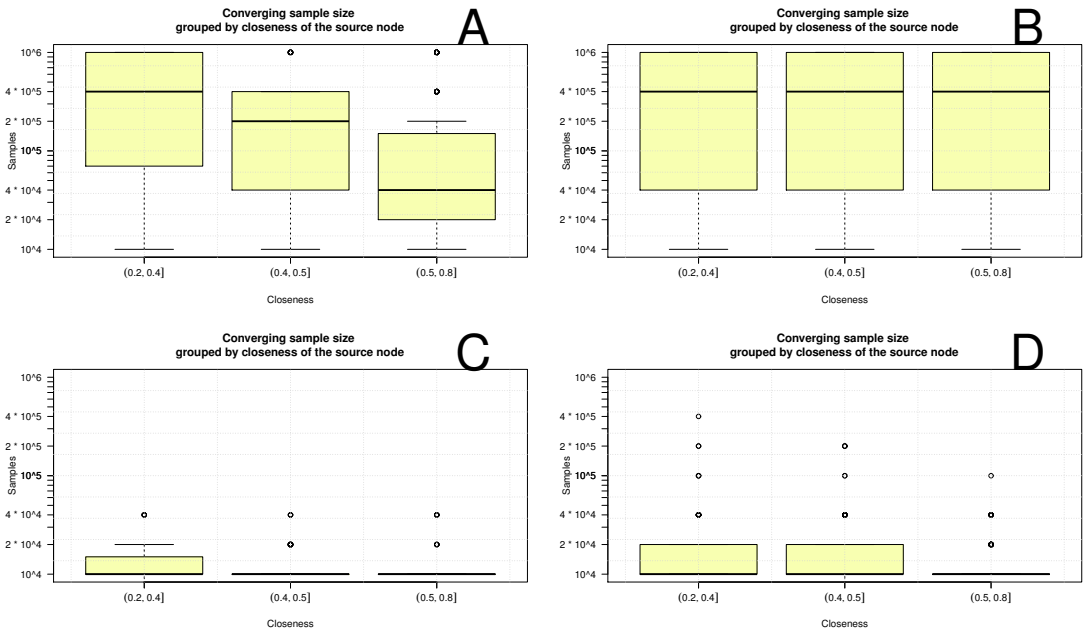

Figure 7.35: Box plots of converging samples distribution for source MAP estimations on 2-Barabási-Albert graphs with $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by closeness of the source node.

Betweenness centrality

Betweenness centrality describes how well situated a vertex is in terms of the paths it lies on as defined in 2.2.

Similarly to closeness, epidemic size grouped by betweenness of the source node shows the epidemics of smaller size are expected to start from the source node with lower betweenness, at least for SIR parameter classes A and B where epidemic is still localized within network size, as presented in Figure 7.36.

For accuracy, on the other hand, betweenness shows to be the centrality measure that separates the detections so the ones more probable to produce correct results have source node of high betweenness, at least for SIR parameters in class C, as presented in Figure 7.37. This can be explain by local network structure since the centres of local communities usually have high centrality.

Similarly to closeness, the converging number of samples for source detection is negatively correlated with betweenness, as presented in Figure 7.38. Source detection for parameter classes with high recovery rate require more samples to converge than their low recovery counterparts.

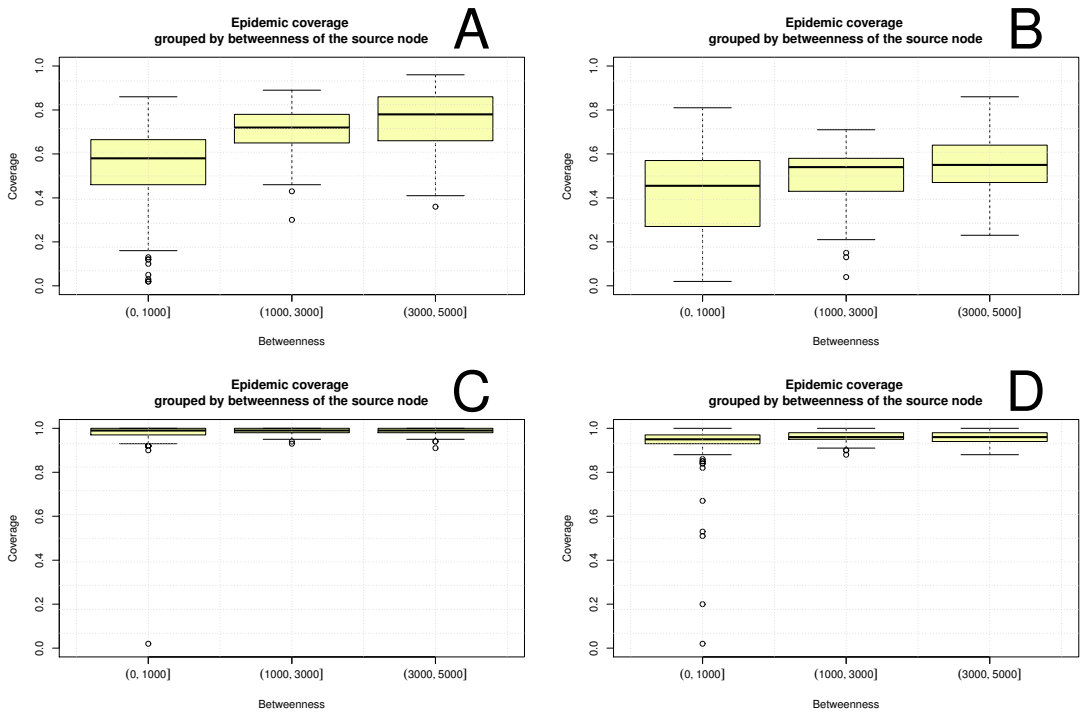

Figure 7.36: Box plots of size of epidemics simulated on 2-Barabási-Albert graphs with $N = 100$ nodes under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by betweenness of the source node.

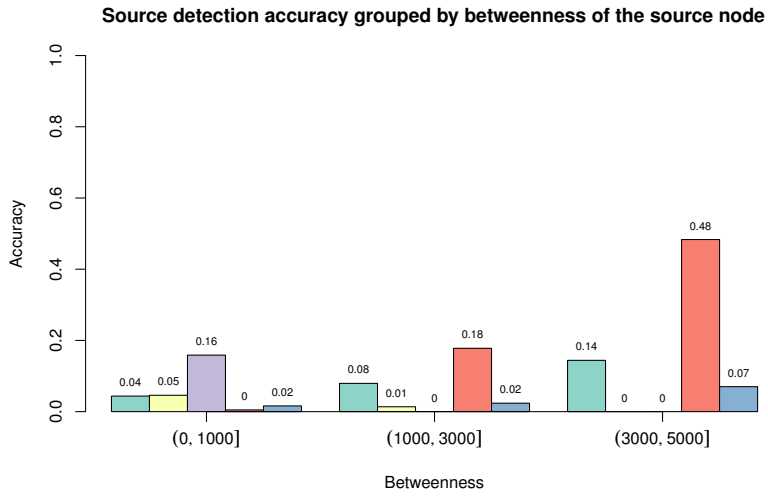

Figure 7.37: Source detection accuracy on 2-Barabási-Albert graphs with $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by betweenness of the source node.

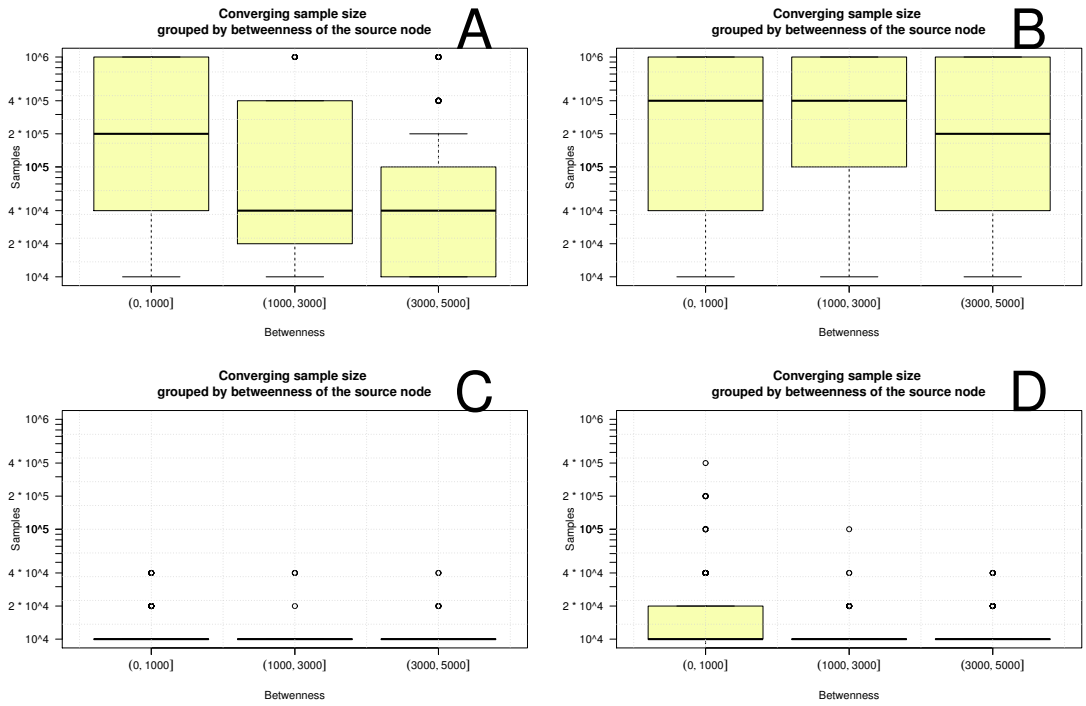

Figure 7.38: Box plots of converging samples distribution for source MAP estimations on 2-Barabási-Albert graphs with $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by betweenness of the source node.

Eigenvector centrality

Eigenvector centrality describes centrality of the source node in terms of node centrality of its neighbours, as defined in 2.3. Epidemic source detectability grouped by eigenvector centrality shows similar results to closeness and betweenness for epidemic coverage, accuracy and converging number of samples, as presented in Figure 7.39, 7.40 and 7.41, respectively since the measures correlate positively.

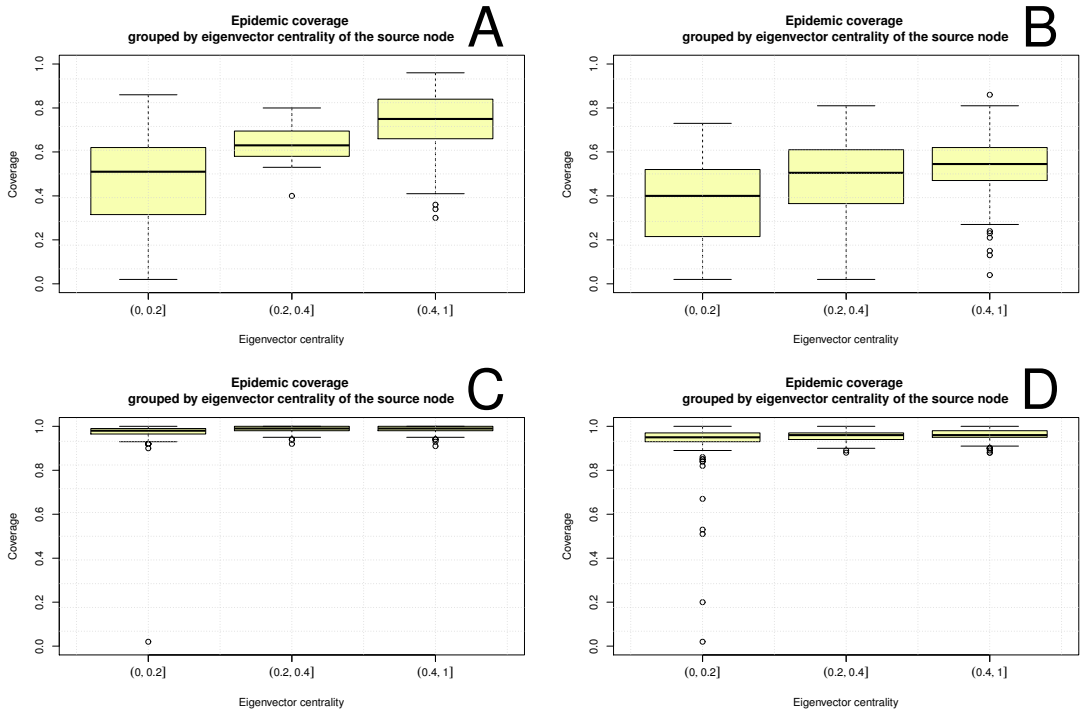

Figure 7.39: Box plots of size of epidemics simulated on 2-Barabási-Albert graphs with $N = 100$ nodes under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by eigenvector centrality of the source node.

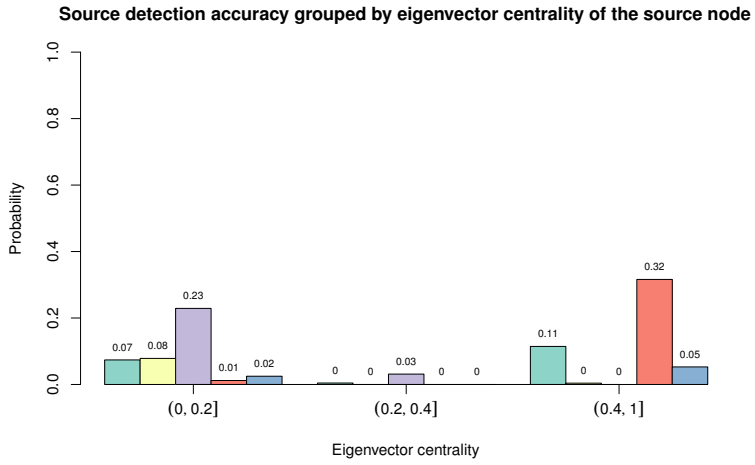

Figure 7.40: Source detection accuracy on 2-Barabási-Albert graphs with $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by eigenvector centrality of the source node.

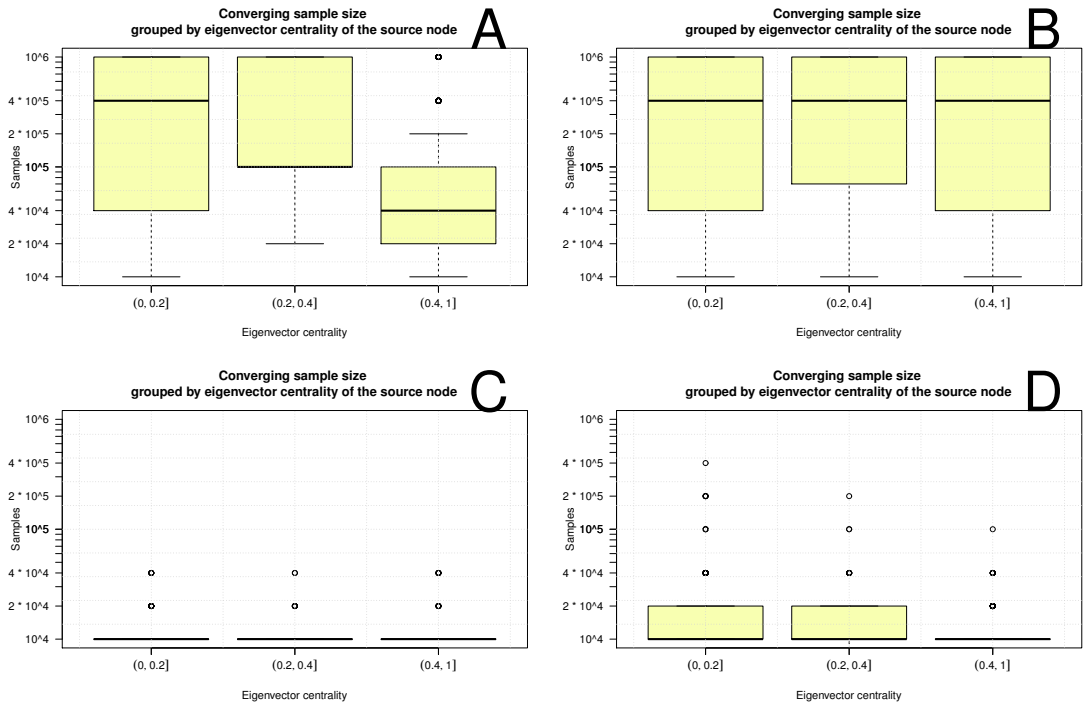

Figure 7.41: Box plots of converging samples distribution for source MAP estimations on 2-Barabási-Albert graphs with $N = 100$ nodes estimated with Soft Margin SIS detector with $10^4 - 10^6$ simulations and fixed $a = \frac{1}{2^5}$ under SIR model with parameters $A = (p = 0.3, q = 0.3, T = 5), B = (p = 0.3, q = 0.7, T = 5), C = (p = 0.7, q = 0.3, T = 5), D = (p = 0.7, q = 0.7, T = 5)$ grouped by eigenvector centrality of the source node.

8. Conclusion

Detectability of patient zero is source detection problem based on partial history of epidemic spreading and underlying network structure. Assuming the epidemic started from a single source and based on the known infected nodes up to fixed time threshold simulated with discrete SIR epidemic spreading model, it is possible to construct MAP classifier based on estimation of epidemic source distribution. These Direct Monte Carlo and Soft Margin methods of Antulov-Fantulin et al. [7] were implemented and results on provided benchmark dataset were reproduced for SIR and ISS models.

Importance sampling method and partial generation of epidemic spreading samples with Sequential importance sampling algorithm provide optimization of Monte Carlo methods in terms of higher accuracy and faster convergence rate while omitting the necessity of direct simulation of epidemic spreading.

Source detection detectability in terms of entropical detectability and accuracy is presented to be higher for higher values of SIR parameter p and ISS parameter a for rumour spreading model on the grid network. This directly correlates with epidemic size and unique position of each node in the epidemic which pushes towards a unique solution of the ill-posed problem of source detection and, consequently, lower entropy and better accuracy of estimated source probability distribution.

When analysing detectability on non-grid random Erdős-Rényi graph with $N = 100$ nodes and $p = 0.01$, the role of SIR parameters p and q in source detectability intensifies. The expected entropy of source distribution for detection on epidemic with higher recovery rate is higher. With high recovery rate there is more ways to obtain the snapshot upon which we base the detection in terms of what nodes got infected and what nodes got recovered. Additionally, the probability of obtaining our snapshot is smaller than probability of obtaining the same snapshot without recovery and the state space stays at most of the size of state space with corresponding epidemic parameters but without recovery. This means more nodes will have similar source probability distribution and expected entropy will be higher.

Detectability is higher for lower values of centrality in Erdős-Rényi graph since the

source nodes with lower values of centrality produce smaller epidemics. Accuracy of source detector on Erdős-Rényi graph is negatively correlated with centrality metrics accordingly.

Additionally, SIR parameter pairs that on the grid network obtain low detectability (the ones with low infection rate), produce on Erdős-Rényi graph accuracy comparable to accuracy of SIR parameter classes (the ones with high infection rate) that have high detectability on the grid network too. This can be explained by the random structure of Erdős-Rényi graph where each node is by default structurally more unique than the node in the grid network and therefore more distinguishable, even within small epidemics.

The source node detectability is more restricted by centrality metrics of the source node for the epidemics with high infection rate.

Detectability based on centrality measures on 2-Barabási-Albert graphs with $N = 100$ nodes is presented to be higher for the source node having lower centrality in terms of degree, closeness, betweenness and eigenvector centrality for both graph types. The detectability is similarly positively correlated with centrality measures of the source node. The presence of local network structure in Barabási-Albert graphs and importance of nodes of high centrality in local communities plays important role in detectability of source nodes with high centrality. Namely, the detectability of these nodes as the source nodes of epidemic is high. The detectability of nodes with low centrality remains high too, similarly to Erdős-Rényi graph.

BIBLIOGRAPHY

[1] Valdis Krebs. The social graph of a facmous mathematician. http://www.orgnet.com/Erdos.html, 2014.

[2] M. E. J. Newman. The structure and function of complex networks. *SIAM REVIEW*, 45:167–256, 2003.

[3] Romualdo Pastor-Satorras, Claudio Castellano, Piet Van Mieghem, and Alessandro Vespignani. Epidemic processes in complex networks. *Rev. Mod. Phys.*, 87:925–979, Aug 2015. doi: 10.1103/RevModPhys.87.925. URL http://link.aps.org/doi/10.1103/RevModPhys.87.925.

[4] Jiaojiao Jiang, Sheng Wen, Shui Yu, Yang Xiang, Wanlei Zhou, and Ekram Hossain. Identifying propagation sources in networks: State-of-the-art and comparative studies. *IEEE Communications Surveys and Tutorials, accepted*, 17(9), 2014.

[5] Kai Zhu and Lei Ying. Information source detection in the SIR model: A sample path based approach. *CoRR*, abs/1206.5421, 2012. URL http://arxiv.org/abs/1206.5421.

[6] Andrey Y. Lokhov, Marc Mézard, Hiroki Ohta, and Lenka Zdeborová. Inferring the origin of an epidemic with a dynamic message-passing algorithm. *Phys. Rev. E*, 90:012801, Jul 2014. doi: 10.1103/PhysRevE.90.012801. URL http://link.aps.org/doi/10.1103/PhysRevE.90.012801.

[7] Nino Antulov-Fantulin, Alen Lančić, Tomislav Šmuc, Hrvoje Štefančić, and Mile Šikić. Identification of patient zero in static and temporal networks: Robustness and limitations. *Phys. Rev. Lett.*, 114:248701, Jun 2015. doi: 10.1103/PhysRevLett.114.248701. URL http://link.aps.org/doi/10.1103/PhysRevLett.114.248701.

[8] Albert lászló Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 1999.

[9] Phillip Bonacich. Power and centrality: A family of measures. *American Journal of Sociology*, 92(5):1170–1182, 1987.

[10] Ignacio Alvarez-Hamelin, Alain Barrat, and Ro Vespignani. k-core decomposition: a tool for the visualization of large scale networks. Technical report.

[11] P. Erdős and A. Rényi. On random graphs. I. *Publ. Math. Debrecen*, 6:290–297, 1959.

[12] W. O. Kermack and A. G. McKendrick. Contributions to the mathematical theory of epidemics—i. *Bulletin of Mathematical Biology*, 53(1):33–55, 1991. ISSN 1522-9602. doi: 10.1007/BF02464423. URL http://dx.doi.org/10.1007/BF02464423.

[13] Nino Antulov-Fantulin, Alen Lancic, Hrvoje Stefancic, and Mile Sikic. Fastsir algorithm: A fast algorithm for simulation of epidemic spread in large networks by using SIR compartment model. *CoRR*, abs/1202.1639, 2012. URL http://arxiv.org/abs/1202.1639.

[14] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 1994. ISBN 0201558025.

[15] Yamir Moreno, Maziar Nekovee, and Amalio F. Pacheco. Dynamics of rumor spreading in complex networks. *Phys. Rev. E*, 69:066130, Jun 2004. doi: 10.1103/PhysRevE.69.066130. URL http://link.aps.org/doi/10.1103/PhysRevE.69.066130.

[16] Christian P. Robert and George Casella. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005. ISBN 0387212396.

[17] Jun S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Publishing Company, Incorporated, 2008. ISBN 0387763694, 9780387763699.

[18] Wing Hung Wong Augustine Kong, Jun S. Liu. Sequential imputations and bayesian missing data problems. *Journal of the American Statistical Association*, 89(425):278–288, 1994. ISSN 01621459. URL http://www.jstor.org/stable/2291224.

[19] Sybil P. Parker, editor. *McGraw-Hill Dictionary of Scientific and Technical Terms (5th Ed.)*. McGraw-Hill, Inc., New York, NY, USA, 1994. ISBN 0-07-042333-4.

[20] Edgar Gabriel, Graham E. Fagg, George Bosilca, Thara Angskun, Jack J. Dongarra, Jeffrey M. Squyres, Vishal Sahay, Prabhanjan Kambadur, Brian Barrett, Andrew Lumsdaine, Ralph H. Castain, David J. Daniel, Richard L. Graham, and Timothy S. Woodall. Open MPI: Goals, concept, and design of a next generation MPI implementation. In *Proceedings, 11th European PVM/MPI Users' Group Meeting*, pages 97–104, Budapest, Hungary, September 2004.

Odredivost nultog pacijenta ovisno o njegovoj poziciji u mreži

Sažetak

Odredivost nultog pacijenta je problem traženja izvora zaraze na temelju djelomične povijesti epidemiološke dinamike i mrežne strukture. Uz pretpostavku da je epidemija krenula iz jednog čvora i na temelju cjelokupnog znanja o mreži te poznavanja zaraženih čvorova do fiksnog vremenskog trenutka u diskretnom SIR modelu širenja zaraze, moguće je konstruktirati MAP klasifikator na temelju estimacije distribucije vjerojatnosti izvora zaraze pomoću Monte Carlo metoda, a koje se dodatno optimiziraju uzorkovanjem po važnosti. Izvedeni novi Soft Margin sekvencijalni Monte Carlo detektor usprediv je s detektorom iz *benchmarka* uz bolju konvergenciju i bolju vremensku složenost.

Pokazano je da detektabilnost kao entropijska detekatiblnost i točnost detekcije ovise o vrijednostima parametara SIR modela za širenje epidemije, a slično je pokazano i za detekciju izvora glasine prema ISS modelu širenja glasine. Za SIR model pokazano je kako je lakše otkriti izvor zaraze ako početni čvor ima manju centralnost.

Ključne riječi: kompleksne mreže, odredivost nultog pacijenta, širenje epidemije, širenje glasina, Monte Carlo metode, sekvencijalni Monte Carlo

Detectability of Patient Zero Depending on its Position in the Network

Abstract

Detectability of patient zero is source detection problem based on partial history of epidemic spreading and underlying network structure. Assuming the epidemic started from a single source and based on the known infected nodes up to fixed time threshold simulated with discrete SIR epidemic spreading model, it is possible to construct MAP classifier based on estimation of epidemic source distribution using Monte Carlo methods optimised with importance sampling. The new Soft Margin Sequential Monte Carlo detector is comparable with the benchmark detector with better convergence and better time complexity.

It is presented how detectability in terms of entropical detectability and accuracy depends on parameters of the SIR model for epidemic source detection, as well as how parameters of ISS model for rumour spreading govern the rumour source detection. For the SIR model it is presented the source node with lower centrality is expected to be more detectable.

Keywords: complex networks, detectability of patient zero, epidemic spreading, rumour spreading, Monte Carlo methods, Sequential Monte Carlo