Estimating network properties from snowball sampled data
Working Paper 11-01

Johannes Illenberger[a], Gunnar Flötteröd[b]

[a]*Berlin Institute of Technology, Group of Transport Systems Planning and Transport Telematics, Salzufer 17–19, D-10587 Berlin, Germany, +49 30 31478793, illenberger@vsp.tu-berlin.de*
[b]*KTH – Royal Institute of Technology, Department of Transport Science, Teknikringen 72, 114 28 Stockholm, Sweden, gunnar.floetteroed@abe.kth.se*

Abstract

This article addresses the estimation of topological network parameters from data obtained with a snowball sampling design. An approximate expression for the probability of a vertex to be included in the sample is derived. Based on this sampling distribution, estimators for the mean degree, the degree correlation, and the clustering coefficient are proposed. The performance of these estimators and their sensitivity with respect to the response rate are validated through Monte Carlo simulations on several test networks. Our approach has no complex computational requirements and is straightforward to apply to real-world survey data. In a snowball sample design, each respondent is typically enquired only once. Different from the widely used estimator for Respondent-Driven Sampling (RDS), which assumes sampling with replacement, the proposed approach relies on sampling without replacement and is thus also applicable for large sample fractions. From the simulation experiments, we conclude that the estimation quality decreases with increasing variance of the network degree distribution. Yet, if the degree distribution is not to broad, our approach results in good estimates for the mean degree and the clustering coefficient, which, moreover, are almost independent from the response rate. The estimates for the degree correlation are of moderated quality.

Keywords: snowball sampling, statistical inference, Monte Carlo simulation

1. Introduction

The increasing availability of large data sets has enabled great advances in the empirical research on social networks. Electronic databases, such as the internet movie database www.imdb.com or the scientific paper database arxiv.org, represent proxy-data sources from which social networks can be inferred. These networks, which can be in the order of up to 10^5 vertices, are usually embedded in an institutional setting or in a specific community, with regard to the above examples: movie actors (Amaral et al., 2000) and authors of scientific papers (Newman, 2001). Large social networks outside of such settings are rather hard to obtain since appropriate proxy-data is rare and even if existing, privacy regulations make its access nearly impossible. The researcher then needs to turn to the traditional "paper and pencil" survey to directly sample a social network.

A straightforward approach is to draw random respondents, denoted as *egos*, and ask them about their social contacts, denoted as *alters*. This so-called "ego-centric" network sampling approach (Wasserman and Faust, 1994) produces star-like networks, which provide insights into the relations between egos and alters. Higher topological networks properties (e.g. transitivity and degree correlation), however, remain unrevealed. In principle, it would be possible to draw a sufficiently large sample such that the ego-centric networks become connected, but practically, such an approach would be prohibitively expensive.

The *snowball sampling* approach, also called chain-referral or link-tracing, addresses this issue. In snowball sampling, an initial set of respondents, denoted as *seeds*, is enquired to report their alters. These alters are then invited to participate in the survey and to report their alters in turn. This procedure is repeated for a given number of iterations (also denoted as *waves* or *stages* (Goodman, 1961)) or until the desired number of vertices is sampled. Snowball sampling reveals more complex network structures than the ego-centric approach because it is not constrained to first degree relations.

A drawback of snowball sampling is that it bears several possible sources of bias (Atkinson and Flint, 2001). Since the recruiting of new respondents is done, or at least influenced, by the respondents themselves, the researcher has only limited control over which individuals are included in the sample. Furthermore, if strong homophily exists between individuals, there is a danger that the snowball is caught in a homogeneous cluster.

Another source of bias is the underlying network topology, which governs the progress of the snowball. Well-connected individuals, that is, vertices with a high degree, have a higher probability to be revealed in a snowball sample than less strongly connected individuals. Well-connected vertices are thus overrepresented in the sam-

ple and an inference of statistical network properties may need to correct for this bias.

Several methods to account for the snowball sampling bias have been proposed in the past (Goodman, 1961; Frank, 1979; Snijders, 1992; Frank and Snijders, 1994; Thompson and Frank, 2000; Heckathorn, 2002; Chow and Thompson, 2003; Thompson, 2006; Volz and Heckathorn, 2008; Handcock and Gille, 2010). Yet, since snowball sampling can be implemented in quite different variants, each specification usually requires its own inference approach. This article treats a snowball sampling design where the sampling frame is known and which is targeted at revealing topological properties of social networks. Specific to this sampling design, we propose a design-based inference method to estimate the mean degree, the degree correlation, and the clustering coefficient. We further address the case of non-responding vertices. Validation is done with Monte Carlo simulations on four test networks, including sensitivity tests towards the response rate. Our approach is computational simple and straightforward to apply to real-world survey data (see for instance Illenberger et al., 2011).

The remainder of this article is organised as follows: Section 2 defines the considered snowball sampling design and gives an overview of related work. Section 3 derives an approximation of the inclusion probability of a vertex in the sample and presents the resulting sampling corrections. Section 4 evaluates the proposed estimators in several settings and on different networks. Finally, Sec. 5 concludes the article.

2. Specification and related work

In this section, a formal specification of the considered snowball sampling design is given and related literature is discussed.

2.1. Formal specification

The snowball sampling variant considered in this article is targeted at revealing the topology of a network. It is assumed that the sampling frame is given, that is, the population from which samples are drawn is known to the researcher. This differs from other snowball sampling studies, for instance implementing Respondent-Driven Sampling, where the sampling frame in unknown and the snowball mechanism is used to access a hard-to-reach population.

The snowball is initialised by sampling a predefined number of seed-vertices uniformly and without replacement. Each seed-vertex is asked to report its alters. All reported alters that were not surveyed before are in turn asked for their alters. This process is repeated until some stopping criterion is fulfilled.

This protocol requires to keep track of which vertices were already sampled. For the further processing of the results, it is also necessary to distinguish between vertices that responded to the survey and those that did not. Causes of non-responsiveness may an individual rejecting the survey or termination of the survey after an individual was indicated as an alter but before that individual could have been inquired in turn.

The vertices having responded to the survey are in the following called *ego-vertices*, and those that were indicated as alters but did not respond for any reason are called *alter-vertices*. The differentiation between ego- and alter-vertices is crucial since some vertex properties, such as the degree, are only known for ego-vertices.

Ego-vertices of iteration i are stored in set $\mathcal{S}^{(i)}$. For convenience, we introduce the notations $\mathcal{S} := \{\mathcal{S}^{(i)}\}_i$, which denotes the joined set of ego-vertices over all iterations.

Consider an undirected and unweighted graph without self-loops. Let $\mathcal{V}$ be the set of vertices, and let N be its size. Further, it is assumed that each vertex in the graph is reachable by each other vertex in one or more steps. The considered snowball sampling algorithm proceeds as follows:

1. Set iteration counter i to 0.
2. Initialise the empty sets $\mathcal{S}^{(i)}$, $\mathcal{A}^{(i)}$, and $\mathcal{R}^{(i)}$.
3. Draw $n^{(i)}$ vertices (seeds) uniformly and without replacement from $\mathcal{V}$. Those vertices are added to $\mathcal{R}^{(i)}$ and represent the candidates for recruitment.
4. Repeat until $\mathcal{S}$ contains at least the desired number of vertices, i has reached some maximum value, or $\mathcal{R}^{(i)}$ is empty:
   (a) Try to recruit each vertex in $\mathcal{R}^{(i)}$, i.e. enquire it to report its neighbours (alters).
   (b) Move all vertices of $\mathcal{R}^{(i)}$ that did respond to the enquiry to $\mathcal{S}^{(i)}$. Move all vertices of $\mathcal{R}^{(i)}$ that did not respond to $\mathcal{A}^{(\leq i)}$.
   (c) Define $\mathcal{R}^{(i+1)}$ as the set of all reported neighbours that are neither in $\mathcal{S}^{(\leq i)}$ nor $\mathcal{A}^{(\leq i)}$.
   (d) Increase iteration counter i by one.

Steps 4a–4c are illustrated in Fig. 1. The above algorithm specifies that sampling is done without replacement, this means that an ego-vertex is never enquired twice, and thus the sampled graph does not contain double edges.

This sampling design is the same as *breadth-first search* in computer science with the difference that vertices may be non-responding.

For details on operational specifications and on a real-world study implementing the present sampling design the interested reader is referred to Kowald et al. (2010) and Illenberger et al. (2011).

2.2. Related work

Snowball sampling can be implemented in quite different variants. Especially when considering a design-based inference approach, each variant requires its own estimation method. The design-based inference directly accounts

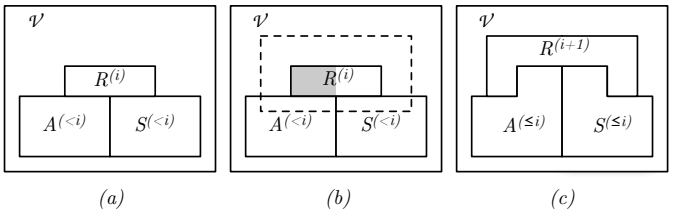


Figure 1: Recruitment process within one iteration. (a) State at the beginning of iteration i, (b) Vertices in $\mathcal{R}^{(i)}$ are enquired. Some vertices are non-responding (grey area). Responding vertices report their alters (dashed square). (c) Vertices that did respond are moved to $\mathcal{S}^{(\leq i)}$, vertices that did not respond are moved to $\mathcal{A}^{(\leq i)}$, $\mathcal{R}^{(i+1)}$ contains the remaining reported alters if they are not already in $\mathcal{S}^{(\leq i)}$ or $\mathcal{A}^{(\leq i)}$.

for the sampling design to obtain an estimator of a specific property or a specific class of properties. In a model-based inference approach, the researcher makes assumptions about a model that describes the population and then estimates the model's parameters from the sampled data.

Some characteristic aspects of a snowball sampling design that can be relevant to the inference approach are the following:

- Does the snowball run on a directed or an undirected graph?
- What is the sampling distribution for the seed-vertices?
- How is the *branching rule* defined? Are all alters recruited or, for instance, is there a recruiting probability for each alter?

An overview of relevant snowball sampling designs with respect to these criteria is given in Tab. 1 and discussed in the following.

One of the first authors who use the term snowball sampling is Goodman (1961). He focuses on the estimation of the number of undirected edges in a network from a snowball sample with a fixed number of iterations. Quite differently from later studies, Goodman defines the underlying graph to be regular such that each vertex has the same predefined degree.

Frank (1979) and later also Snijders (1992) address the estimation of the vertex inclusion probability and edge inclusion probability. Knowledge about the inclusion probabilities allows for unbiased estimates of population totals and means. Both authors show that the inclusion probabilities for a snowball sampling that is run only to the first iteration can be directly calculated. Snijders (1992) also considers snowball samples with multiple iterations. If a snowball sample is run for $2i-1$ iterations, then the inclusion probability of a vertex can be calculated because the number of vertices with geodesic distance (number of edges in shortest path) $\leq i$ is known, and thus each possible recruiting path can be identified. However, this requires to perform $i-1$ additional iterations just to calculate the shortest paths and further requires that the branching rule is defined such that all vertices reported by an ego-vertex participate in the survey, i.e. each vertex is fully expanded.

The problem of estimating the vertex in-degree from a snowball sample is addressed with Monte Carlo simulations by Johnson et al. (1989). They investigate the effects of the number of seeds, the number of iterations, and the maximum number of neighbours each vertex is allowed to report on the estimated in-degrees. Johnson et al. highlight that larger in-degrees are estimated with a lower error than smaller in-degrees. They state that the number of iterations accounts for most of the estimation errors, whereas the number of seeds has only a minor effect.

A comparison of snowball sampling with the egocentric sampling approach and link-sampling, that is, a random draw of edges, is presented by Lee et al. (2006). Naturally, the latter sampling approach is only applicable if edges are observable. They conduct numerical simulations on real-world networks including a protein interaction network, the Internet at the autonomous systems level, and a co-authorship network. Their results indicate that snowball sampling underestimates several topological network properties such as the exponents of the power-law degree distribution, the betweenness distribution, and the degree correlation.

A common application of snowball sampling is to access specific populations that are difficult or even impossible to reach through direct sampling. Such applications are addressed by Frank and Snijders (1994) and Heckathorn (1997, 2002). Heckathorn's approach, known as Respondent-Driven Sampling (RDS), is probably the most common real-world application of snowball sampling. Especially in medical research, RDS is of interest as it allows to access hidden or hard-to-reach populations such as drug-users or HIV infected people.

In RDS, the selection of seeds is typically non-uniform but aims at individuals who are somehow related to the target population. In theory, RDS requires a respondent to recruit only one neighbour. Hence, the sampling process constitutes a random walk on a graph with the transition probability from vertex v to adjacent vertex w being $p_{vw} = 1/k_v$, where k_v denotes the (out-)degree of v (Volz and Heckathorn, 2008). With each additional step, i.e. with each additional sample, this process approaches a known equilibrium distribution from which the selection probability of a vertex can be derived. Thus, the error of the estimates decreases with increasing sample size. This, however, also implies that the sampling process is with replacement, which means that an individual can be recruited multiple times – an aspect in which RDS differs from the above sampling designs. Gile (2011) presents a treatment of RDS as a successive sampling process. This respects the actual without-replacement nature of the operative RDS process. A comprehensive review of the RDS methodology including a detailed discussion of the strengths and weaknesses is given by Gile and Hand-

Table 1: Comparison of different snowball sampling designs. Notation for branching rules, based on the convention that k_i indicates the degree of vertex i: "k_i" = all neighbours are reported; "$\propto k_i$" = number of reported neighbours is propotional to the vertex degree; "k^*" = number of reported neighbours is constant for all vertices.

Reference	Graph	Sampling design for seeds	Branching rule	Replacement
Goodman (1961)	undirected	uniform	k_i; all vertices have same degree	without
Frank (1979)	directed	"	k_i; only one iteration	"
Johnson et al. (1989)	"	"	$\min\{k^*, k_i\}$	"
Snijders (1992)	both	"	$\propto k_i$	"
Thompson and Frank (2000); Chow and Thompson (2003)	"	various	k_i	"
Kwanisai (2006)	"	"	$\propto k_i$	"
Thompson (2006)	directed	"	k_i with probability d, random new vertex with $1-d$	both
Lee et al. (2006)	undirected	uniform ($n = 1$)	k_i	without
Volz and Heckathorn (2008)	"	non-uniform	$k^* = 1$	with
Handcock and Gille (2010)	both	uniform	k_i	without
Gjoka et al. (2011)	undirected	uniform ($n = 1$)	$k^* = 1$ with probability dependent on neighbour's degree	with
Kurant et al. (2011)	"	"	$k^* = 1$ with probability dependent on edge weight	"
present study	undirected	uniform	k_i; alters can be non-responding	without

cock (2010). An application of RDS in the context of social network analysis, rather then accessing hidden populations, is presented by Wejnert (2010).

An obvious extension to the random walk mechanism is to define the transition probability as dependent on the neighour's degree, for instance, so that the transition probability decreases if the neighbour's degree is greater then the current vertex's degree. This compensates for the degree bias already within the sampling process (Gjoka et al., 2011). Likewise, one can define the transition probability dependent on edge weights, where weights are adjusted so that the process is more likely to walk to those vertices that are of interest to the researcher (Kurant et al., 2011). In either case, knowledge about the current vertex's neighbours is required (the neighbour's degree or the neighbour's property on which the edge weight is calculated). When sampling an online network, this information can be obtained with less effort. When sampling a real-world social network, however, this approach is rather difficult.

Thompson and Frank (2000) propose a stochastic block model with edge probabilities dependent on a dichotomous vertex variable. It can be regarded, for instance, as a model of an HIV population where the probability of a relation between two individuals depends on the HIV-status of both vertices. Thompson and Frank give an expression for the likelihood of the sample data, which is also used by Chow and Thompson (2003) together with the same graph model to obtain Bayesian estimators. Kwanisai (2006) addresses the same graph model as well but uses a link-tracing design where only a fraction of links is traced. Using the family of exponential random graph models, as it is done by Handcock and Gille (2010), adds more flexibility to the model but at the cost of higher computational complexity. In general, such model-based inference approaches enjoy more flexibility compared to design-based approaches. However, at least for larger networks, direct calculations are usually infeasible because they imply enumeration over a large number of network and parameter configurations. Even with well developed Markov chain Monte Carlo methods, to which these models lend themselves admirably, the computational effort remains high.

Thompson (2006) proposes a sampling design that is not only dependent on the topology of the network but also on vertex attributes and edge weights. He presents four estimators for the population mean, starting with a rather simple estimator based on the initial drawing of seed-vertices to composite estimators that further include the conditional selection probabilities of samples in succeeding iterations. What makes the approaches computational expensive is that they require the enumeration over all possible re-orderings of the sample selection sequence. For small networks, the number of permutations is manageable for direct computation. Yet, for large networks this approach requires usage of Markov chain Monte Carlo methods.

The present study develops a design-based approach to estimate structural network properties. The snowball is initialised with a uniform sample of seeds. Each vertex is assumed to report all of its neighbours; however, neighbours may be non-responding with a constant probability. It is assumed that the network consists of one component, which means that the snowball can be run until all vertices are sampled.

3. Estimation

Snowball sampling selects vertices with unequal inclusion probabilities. However, in contrast to other sampling strategies, such as importance sampling, the inclusion probabilities are not deliberately chosen but are, except for the initial and first iteration, unknown. While all inclusion probabilities are equal in the zero-th iteration, they scale with the vertex degree in the first iteration because each neighbour is a potentially recruiting vertex. In succeeding iterations, the inclusion probability of a vertex does not only depend on its degree but also on the degrees of its neighbours.

3.1. Inclusion probability and population mean

In the remainder of this article, the following notation is used: Quantities that are calculated based on different iterations of the snowball sampling are written with the iteration index in parentheses in the superscript. For instance, the number of ego-vertices sampled in iteration i is denoted by $n^{(i)}$, and the number of ego-vertices that have been sampled up to and including iteration i is denoted by $n^{(\leq i)}$. Symbols without an iteration index refer to the complete sample. For example, π_v is the inclusion probability of vertex v in the entire sample.

To obtain estimators for the population total and mean of a quantity of interest, one requires the π-expanded values y_v/π_v, where y_v is the quantity of interest for a sampled vertex v. The inclusion probabilities π_v are unknown a priori, but they can be estimated from the data.

Denote by $\pi_v^{(\leq i)}$ the probability that vertex v is included in a snowball sample that has been run up to and including iteration i. Given a 100 percent response rate, this equals the probability that one of v's neighbours has been sampled in or before the previous iteration $i-1$. Observing that the probability that a vertex v is *not* sampled in or before iteration i is the joint probability that none of its neighbours w has been sampled in or before the previous iteration $i-1$, and assuming as a first simplification that the events of being not sampled are independent, one obtains the following approximation:

$$\pi_v^{(\leq i)} \approx 1 - \prod_{w \sim v} \left(1 - \pi_w^{(<i)}\right) , \qquad (1)$$

where $w \sim v$ reads as "w is a neighbour of v". Here and in the following, the subscript v denotes the "vertex under consideration", and the subscript w refers to its neighbour(s). The probability $\pi_w^{(<i)}$ is, however, just as unknown as $\pi_v^{(\leq i)}$. A second simplification, which will turn

out later to yield quite satisfactory results, is to assume that all neighbours of v are included in the sample up to iteration $i-1$ independently and with equal probabilities. This assumptions implies that a vertex v reveals no information about the sampling probabilities of its neighbours. Since these probabilities actually depend on the degrees of the neighbours, an implicit assumption is that there is no degree correlation in the network.

Based on this assumption, the *ex post* inclusion probability of a neighbour w is approximated by

$$\pi_w^{(<i)} \approx \frac{n^{(<i)}}{N}. \tag{2}$$

That is, neighbours are treated as if they had been obtained from uniform random sampling without replacement, and the resulting estimator of $\pi_v^{(\leq i)}$ in Eq. 1 becomes

$$\hat{\pi}_v^{(\leq i)} := 1 - \prod_{w \sim v} \left(1 - \frac{n^{(<i)}}{N}\right). \tag{3}$$

Since the factors in Eq. 3 are equal for all neighbours, one obtains

$$\hat{\pi}_v^{(\leq i)} := \hat{\pi}^{(\leq i)}(k_v) := 1 - \left(1 - \frac{n^{(<i)}}{N}\right)^{k_v}, \tag{4}$$

where k_v is the degree of vertex v and $\hat{\pi}^{(\leq i)}(k)$ is an estimator of the inclusion probability that only depends on the degree of a considered vertex and the sample size of the previous iteration. This estimator is applicable for $i > 0$; in the 0th iteration, samples are drawn uniformly such that $\hat{\pi}^{(0)}(k) = \pi^{(0)}(k) = n^{(0)}/N$.

Given the estimated inclusion probabilities $\hat{\pi}_v$, one obtains

$$\hat{t}_y := \sum_{v \in \mathcal{S}} \frac{y_v}{\hat{\pi}_v} \tag{5}$$

as an estimator for the population total of the quantity y, where y_v denotes the quantity of interest for vertex v and $\mathcal{S}$ denotes the set of sampled vertices. Based on this, the *weighted sample mean* (Särndal et al., 1992) constitutes an estimator for the population mean:

$$\hat{\bar{y}}_{(\text{wsm})} := \left(\sum_{v \in \mathcal{S}} \frac{1}{\hat{\pi}_v}\right)^{-1} \sum_{v \in \mathcal{S}} \frac{y_v}{\hat{\pi}_v}. \tag{6}$$

3.2. Comparison with the RDS-estimator

The estimator widely used in studies applying Respondent-Driven Sampling considers the sampling process to be analogous to a random walk on a network. That is, the process is treated as if a respondent recruits only one random contact, whereas the recruit may have already been sampled. In practice, Respondent-Driven Sampling allows also the recruitment of multiple neighbours (see Goel and Salganik (2009) for the effects of multiple recruitment). The random walk constitutes a Markov process, which in equilibrium occupies a vertex v with probability proportional to degree (Salganik and Heckathorn, 2004):

$$\frac{k_v}{Nn} \cdot \sum_{u \in \mathcal{S}} \frac{1}{k_u}. \tag{7}$$

This expression is derived from a with replacement sampling strategy. For a relatively small sample size n, one may assume that the probability of visiting v more than once approaches zero, such that the estimated inclusion probability of v becomes n-times this expression:

$$\hat{\pi}_{v,(\text{rds})} := \frac{k_v}{N} \cdot \sum_{u \in \mathcal{S}} \frac{1}{k_u}. \tag{8}$$

Inserting this into the weighted sample mean (Eq. 6) yields the estimator

$$\hat{\bar{y}}_{(\text{rds})} = \left(\sum_{v \in \mathcal{S}} \frac{1}{k_v}\right)^{-1} \sum_{v \in \mathcal{S}} \frac{y_v}{k_v}, \tag{9}$$

which in the following is referred to as the *RDS-estimator* (Volz and Heckathorn, 2008).

The major difference between $\hat{\bar{y}}_{(\text{rds})}$ and $\hat{\bar{y}}_{(\text{wsm})}$ is that the RDS-estimator assumes sampling with replacement, whereas $\hat{\bar{y}}_{(\text{wsm})}$ assumes sampling without replacement. For small sample sizes, the inclusion probabilities of sampling with replacement approximate those of sampling without replacement and thus $\hat{\bar{y}}_{(\text{rds})}$ approximates $\hat{\bar{y}}_{(\text{wsm})}$. Yet, $\hat{\bar{y}}_{(\text{wsm})}$ is consistent in that it predicts an inclusion probability of one if all vertices are sampled, whereas the RDS-estimator invariably predicts a dependency of the inclusion probability on the degree.

4. Simulation

4.1. Simulation setup

To validate the performance of the proposed estimator, a series of simulation experiments is conducted. All experiments implement the snowball sampling design according to the specification of Sec. 2.1. For comparison, a naive estimator that does no bias-correction and the RDS-estimator are applied. However, the comparison with the RDS-estimator is limited because it is designed for a selection of seed-vertices proportional to degree. Here, seed-vertices are randomly selected. Simulation experiments are conducted on four test networks (Tab. 2):

- *random*: An Erdős-Rényi random network (Erdős and Rényi, 1959) as a reference network.
- *hepth*: A collaboration network of authors in high energy physics theory from the arxiv.org database (Newman, 2001). This network exhibits high transitivity and degree correlation.

– *condmat*: A network similar to the *hepth* network but from collaborations on condensed matters (Newman, 2001). It differs from the *hepth* network in that it is larger and its degree distribution has a greater variance.

– *slashdot*: A social network generated from user interactions of the website slashdot.org (Leskovec et al., 2009). The *slashdot* network is compared to the other networks quite large and exhibits a considerably greater variance of the degree distribution.

The *hepth* network and *condmat* network include multiple disconnected components. From both networks the giant component is extract so that the snowball simulation is able to sample the entire network.

In order to account for the stochasticity in the simulations, each experiment is repeated 1000 times. Snapshots of the sampled networks are stored and analysed after the collection of predefined numbers of ego-vertices. Since this can occur within the expansion of an iteration, the order in which the vertices of that iteration are processed is randomised at the beginning of each iteration.

4.2. Simulation with full response rate

We first consider a configuration where each vertex is responding. Simulations are initialised with ten randomly drawn seed-vertices and are run until the entire network is sampled. The relative error is used as an indicator for the estimators' performances.

4.2.1. Mean degree

A naive estimator for the mean degree that does not account for the bias, i.e. just averages the observed degree, is

$$\hat{\bar{k}}_{(\text{obs})} = \frac{1}{n} \sum_{v \in \mathcal{S}} k_v \ . \tag{10}$$

From Eq. 6, the estimator

$$\hat{\bar{k}}_{(\text{wsm})} = \left(\sum_{v \in \mathcal{S}} \frac{1}{\hat{\pi}_v} \right)^{-1} \sum_{v \in \mathcal{S}} \frac{k_v}{\hat{\pi}_v} \tag{11}$$

is obtained. The corresponding RDS-estimator is

$$\hat{\bar{k}}_{(\text{rds})} = \left(\sum_{v \in \mathcal{S}} \frac{1}{k_v} \right)^{-1} \cdot n \ . \tag{12}$$

The above estimators consider only ego-vertices in $\mathcal{S}$ because the true degree of newly detected alter-vertices (vertices in $\mathcal{R}$) is unknown.

The performance of the various mean degree estimators is visualised in Figure 2. This figure shows the mean, 0.1 quantile, and 0.9 quantile of the distribution of relative errors in the simulation ensemble.

The estimates of $\hat{\bar{k}}_{(\text{obs})}$ clearly reveal the bias of the snowball mechanism in all four networks. The estimates are permanently above the real mean degree, except, of course, for the last snapshot where the entire network is sampled. The bias is the strongest for small sample sizes where high degrees are heavily overrepresented. Regarding the four test networks, the bias is moderate for the *random* network, more pronounced for the *hepth* network and *condmat* network, and the strongest for the *slashdot* network. This observation indicates that the bias scales with the variance of the degree distribution. This is natural because the bias is bound to the value range of the underlying degree distribution.

The estimates of $\hat{\bar{k}}_{(\text{wsm})}$ show a similar behaviour as the naive estimator: the estimation error increases with the width of the degree distribution. While the weighted sample mean estimator performs well for the *random* network and the *hepth* network, it performs worse for the *condmat* network and scatters quite heavily with the *slashdot* network, which has by far the greatest variance of $\text{Var}(k) = 1686.79$. If the entire network is sampled, the estimator predicts the true mean degree for all networks. This indicates the estimator's consistency.

A comparison of $\hat{\bar{k}}_{(\text{wsm})}$ and the RDS-estimator $\hat{\bar{k}}_{(\text{rds})}$ shows that both estimators perform nearly equivalent for small sample sizes. Yet, if the number of ego-vertices increases, the with replacement condition is violated. The RDS-estimator then fails to predict the true values, which becomes visible in that $\hat{\bar{k}}_{(\text{rds})}$ considerably underestimates the true mean degree.

4.2.2. Degree correlation

The degree correlation is quantified by the Pearson correlation coefficient of the degrees of the vertices on either side of all edges in the network (Newman, 2002):

$$r = \frac{\frac{1}{M} \sum\limits_{e \in \mathcal{E}} k_{v,e} k_{w,e} - \left(\frac{1}{M} \sum\limits_{e \in \mathcal{E}} \frac{1}{2} \left(k_{v,e} + k_{w,e} \right) \right)^2}{\frac{1}{M} \sum\limits_{e \in \mathcal{E}} \frac{1}{2} \left(k_{v,e}^2 + k_{w,e}^2 \right) - \left(\frac{1}{M} \sum\limits_{e \in \mathcal{E}} \frac{1}{2} \left(k_{v,e} + k_{w,e} \right) \right)^2} \ , \tag{13}$$

where $k_{v,e}$ and $k_{w,e}$ denote the degrees of the two adjacent vertices v and w of an edge e (in arbitrary yet unique order) and $\mathcal{E}$ denotes the set of all edges in the network with $M = |\mathcal{E}|$ being its size.

To obtain a naive estimator, we evaluate Eq. 13 only for the set of sampled edges $\mathcal{T}$ with size $m = |\mathcal{T}|$, where an edge is denoted as sampled if both adjacent vertices are in $\mathcal{S}$:

$$\hat{r}_{(\text{obs})} = \frac{\frac{1}{m} \sum\limits_{e \in \mathcal{T}} k_{v,e} k_{w,e} - \left(\frac{1}{m} \sum\limits_{e \in \mathcal{T}} \frac{1}{2} \left(k_{v,e} + k_{w,e} \right) \right)^2}{\frac{1}{m} \sum\limits_{e \in \mathcal{T}} \frac{1}{2} \left(k_{v,e}^2 + k_{w,e}^2 \right) - \left(\frac{1}{m} \sum\limits_{e \in \mathcal{T}} \frac{1}{2} \left(k_{v,e} + k_{w,e} \right) \right)^2} \ . \tag{14}$$

In contrast to one-point properties, such as the degree, the sample of interest is now an edge. The inclusion probability $\pi_{(vw)}$ of an edge $e = (vw)$ follows from the observation

Table 2: Descriptive statistics of test networks.

Network	N	$\bar{k}$	Var(k)	min$\{k\}$	max$\{k\}$	C	r
random	36456	9	8.95	1	26	≈ 0	0.002
hepth	8638	5.74	41.62	1	65	0.48	0.24
condmat	36458	9.42	173.91	1	278	0.66	0.18
slashdot	82168	12.27	1686.79	1	2552	0.06	-0.07

that the probability that an edge is sampled before or in iteration i equals the probability that at least one of its adjacent vertices v or w is sampled before or in iteration $i-1$. Hence,

$$\hat{\pi}_{(vw)}^{(\leq i)} = \left(\hat{\pi}_v^{(<i)} + \hat{\pi}_w^{(<i)}\right) - \left(\hat{\pi}_v^{(<i)}\hat{\pi}_w^{(<i)}\right) \ , \tag{15}$$

where again independence of the sampling events is assumed. An estimator $\hat{r}_{(\text{wsm})}$ of the degree correlation can now be obtained by (i) estimating M, the total number of edges, by $\hat{M} = \sum_{e\in\mathcal{T}} 1/\hat{\pi}_{(vw)}$ and (ii) estimating the edge inclusion probabilities according to Eq. 15 from the approximate vertex inclusion probabilities:

$$\hat{r}_{(\text{wsm})} = \frac{\frac{1}{\hat{M}}\sum\limits_{e\in\mathcal{T}} \frac{k_{v,e}k_{w,e}}{\hat{\pi}_{(vw)}} - \left(\frac{1}{\hat{M}}\sum\limits_{e\in\mathcal{T}} \frac{(k_{v,e}+k_{w,e})}{2\hat{\pi}_{(vw)}}\right)^2}{\frac{1}{\hat{M}}\sum\limits_{e\in\mathcal{T}} \frac{\left(k_{v,e}^2+k_{w,e}^2\right)}{2\hat{\pi}_{(vw)}} - \left(\frac{1}{\hat{M}}\sum\limits_{e\in\mathcal{T}} \frac{(k_{v,e}+k_{w,e})}{2\hat{\pi}_{(vw)}}\right)^2} \ . \tag{16}$$

The resulting estimator does not require knowledge of M, which is typically unknown (or even the quantity of interest) in real-world applications.

The corresponding RDS-estimator $\hat{r}_{(\text{rds})}$ is obtained by replacing $\hat{\pi}_v^{(<i)}$ and $\hat{\pi}_w^{(<i)}$ in Eq. 15 with the expression for the vertex inclusion probability in Eq. 8.

The performance of the various degree correlation estimators is visualised in Figure 3. Since the degree correlation of the *random* network and *slashdot* network is close to zero, the absolute error (Fig. 3(a) and 3(d)) instead of the relative error (as in Fig. 3(b) and 3(c)) is shown. In all four networks the predictions of all three estimators heavily scatter, yet, expect for $\hat{r}_{(\text{rds})}$ in Fig. 3(b)–3(d), the variance decreases with increasing sample fraction. Considering the simulation ensemble mean, $\hat{r}_{(\text{wsm})}$ predicts reasonable values in case of the *random* and *hepth* network (Fig. 3(a)–3(b)). To the contrary, in case of the *condmat* and *slashdot* network the naive estimator even yields better results than $\hat{r}_{(\text{wsm})}$ (Fig. 3(c)–3(d)). Only with large sample sizes $\hat{r}_{(\text{wsm})}$ outperforms the naive estimator. Again, the broader degree distribution of the *condmat* and *slashdot* network reduces the performance. The RDS-estimator $\hat{r}_{(\text{rds})}$ performs similar as $\hat{r}_{(\text{wsm})}$ for smaller sample fractions, however, expect for the *random* network, with increasing sample size the predictions of the RDS-estimator heavily scatter.

When comparing $\hat{r}_{(\text{wsm})}$ and $\hat{r}_{(\text{rds})}$ with the naive estimator, it needs to be acknowledged that the two more complex estimators rely on an additional approximation when estimating $\hat{\pi}_{(vw)}$ from $\hat{\pi}_v$ and $\hat{\pi}_w$, which is likely to negatively affect their performance.

4.2.3. Transitivity

Network transitivity is quantified with the definition of the clustering coefficient by Watts and Strogatz (1998):

$$C = \frac{1}{N}\sum_{v\in\mathcal{V}} \frac{2m_v}{k_v\left(k_v-1\right)} \ , \tag{17}$$

where m_v denotes the number of edges that connect neighbours of v.

When estimating transitivity from a snowball sample, only ego-vertices all neighbours of which are also ego-vertices are accounted for because the neighbourhood of alter-vertices is unknown (in particular, edges between alter-vertices are missing). This means that only those ego-vertices that have been sampled strictly before the last iteration are considered. Since C represents an average over a vertex property, obtaining the estimators is straightforward. The naive estimator is

$$\hat{C}_{(\text{obs})}^{(\leq i)} = \frac{1}{n^{(<i)}}\sum_{v\in\mathcal{S}^{(<i)}} \frac{2m_v}{k_v\left(k_v-1\right)} \ ; \tag{18}$$

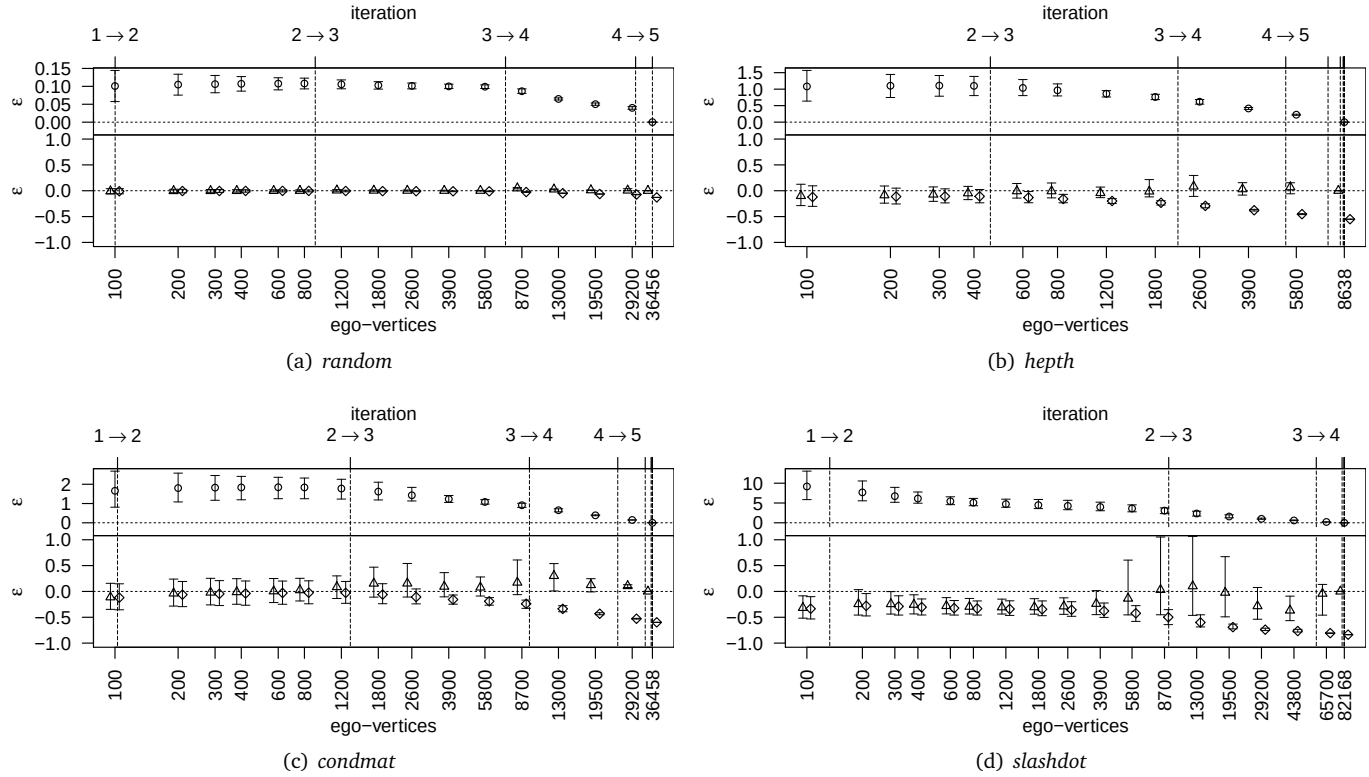
using the weighted sample mean yields

$$\hat{C}_{(\text{wsm})}^{(\leq i)} = \left(\sum_{v\in\mathcal{S}^{(<i)}} \frac{1}{\hat{\pi}_v^{(<i)}}\right)^{-1} \sum_{v\in\mathcal{S}^{(<i)}} \frac{2m_v}{k_v\left(k_v-1\right)}\cdot\frac{1}{\hat{\pi}_v^{(<i)}} \ , \tag{19}$$

and the corresponding RDS-estimator results in

$$\hat{C}_{(\text{rds})}^{(\leq i)} = \left(\sum_{v\in\mathcal{S}^{(<i)}} \frac{1}{k_v}\right)^{-1} \sum_{v\in\mathcal{S}^{(<i)}} \frac{2m_v}{k_v\left(k_v-1\right)}\cdot\frac{1}{k_v} \ . \tag{20}$$

The performance of the various transitivity estimators is visualised in Figure 4. Since the clustering coefficient of the *random* network and *slashdot* network is close to zero, the absolute error (Fig. 4(a) and 4(d)) instead of the relative error (as in Fig. 4(b) and 4(c)) is shown. Both collaboration networks, *hepth* and *condmat*, exhibit a negative correlation between the vertex degree and the number of triangles connected to the vertex. For that reason, transitivity is underestimated with the naive estimator, in which high degree vertices, i.e. those that contribute less triangles, are overrepresented. The estimator $\hat{C}_{(\text{wsm})}$ performs



(a) *random* (b) *hepth*

(c) *condmat* (d) *slashdot*

Figure 2: Relative error ϵ of the mean degree estimated with the naive estimator $\hat{\bar{k}}_{(\text{obs})}$ (◯), the weighted sample mean estimator $\hat{\bar{k}}_{(\text{wsm})}$ (△), and the RDS-estimator $\hat{\bar{k}}_{(\text{rds})}$ (◇). The symbol indicates the ensemble mean, the lower whisker indicates the 0.1 quantile, and the upper whisker indicates the 0.9 quantile. The data is extracted after sampling the number of ego-vertices indicated at the x-axis ticks, but the symbols are shifted to the right and left, respectively, for better visibility. Snowball iteration transitions are indicated by vertical dashed lines (averaged over the simulation ensemble). The x-axis is log-scaled.

well for both networks. The estimates, however, are more precise in the *condmat* network than in the *hepth* network, although the first network is more prone to the bias. Presumably, the higher clustering coefficient of the *contmat* network allows for a better estimation even with smaller sample sizes. At average the estimators correctly predict a clustering coefficient of zero for the *random* network. Considering the *slashdot* network the naive estimator underestimates the true clustering coefficient. The estimator $\hat{C}_{(\text{wsm})}$ predicts values with an absolute error ranging from -0.05 and 0.1, yet at average approximately zero.

Generally, the RDS-estimator $\hat{C}_{(\text{rds})}$ shows the same behaviour as with the mean degree and degree correlation. In this case, however, $\hat{C}_{(\text{rds})}$ significantly diverges from $\hat{C}_{(\text{wsm})}$ just in the last or last two snapshots, respectively.

4.3. Sensitivity towards variations in the response rate

The response rate usually depends on the characteristics of the population. Of course, arrangements such as a monetary incentives can increase the response rate, but for real-world applications, it is crucial to investigate the sensitivity of the estimator towards a reduced response rate.

In the following, the proposed estimation approach is extended in order to account for variations in the response rate. It is assumed that a fraction $1-\alpha$ of vertices are non-responding. In the snowball simulation, these vertices are selected from a uniform distribution before the sampling starts and are not expanded during the snowball. This approach implies the assumption that the response probability is equally distributed over all vertices and does not change throughout the sampling process. It turns out that a snowball sample with a low response rate behaves like a snowball sample with a high response rate on a thinned-out network. Simulation experiments are conducted with response rates from 0.1 to 1.0 in 0.1 steps. Due to the high computational effort, the simulation ensemble size is reduced to 200 simulations per parameter configuration.

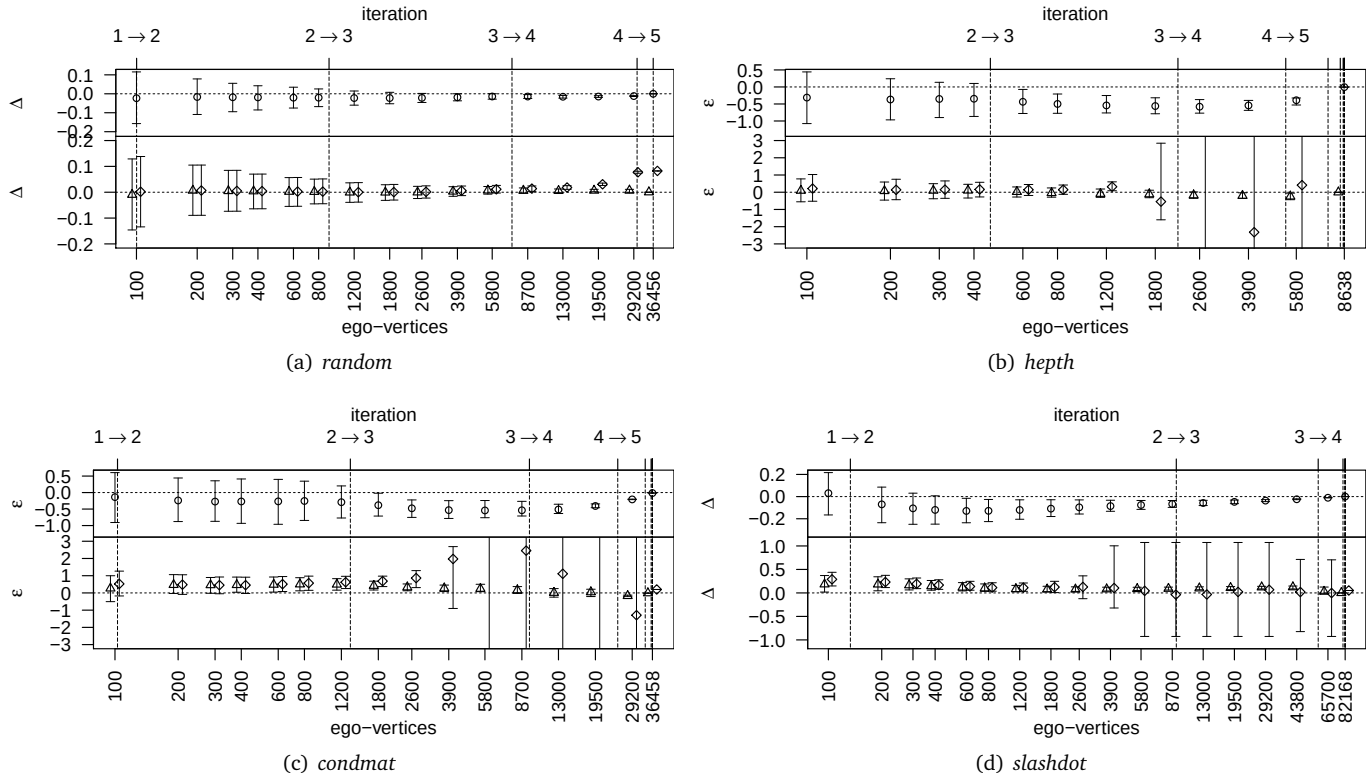
4.3.1. Extensions to the estimator

The estimated inclusion probabilities are straightforward to extend:

$$\hat{\pi}_{v,\alpha} := \alpha\hat{\pi}_v. \tag{21}$$

The response rate α is estimated from the survey data through

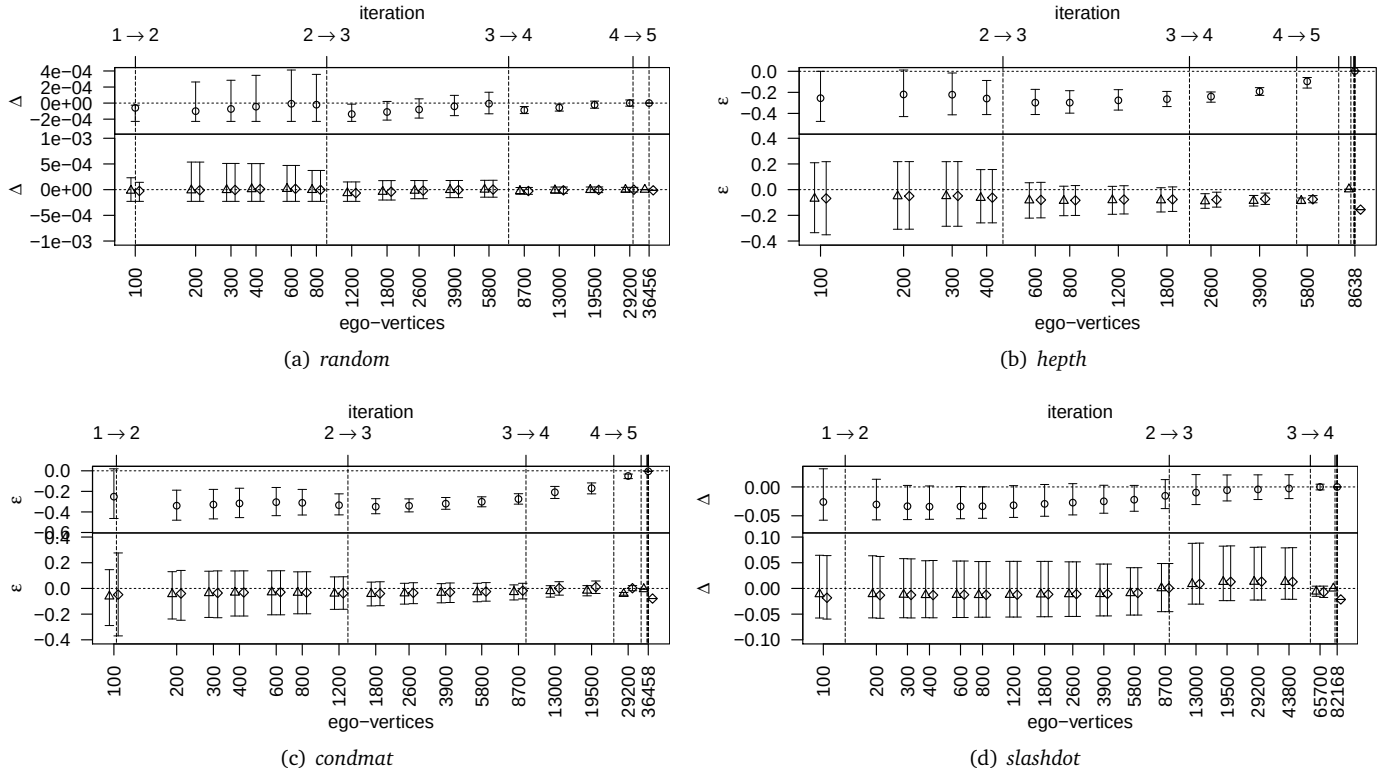
$$\hat{\alpha}^{(\leq i)} := \frac{n^{(\leq i)}}{n^{(<i)} + a^{(<i)}} \tag{22}$$



(a) *random* (b) *hepth*

(c) *condmat* (d) *slashdot*

Figure 3: **Relative error** ϵ ((b) and (c)) and **absolute error** Δ ((a) and (d)) of the degree correlation estimated with the naive estimator $\hat{r}_{(\mathrm{obs})}$ (◯), the weighted sample mean estimator $\hat{r}_{(\mathrm{wsm})}$ (△), and the RDS-estimator $\hat{r}_{(\mathrm{rds})}$ (◊). The symbol indicates the ensemble mean, the lower whisker indicates the 0.1 quantile, and the upper whisker indicates the 0.9 quantile. The data is extracted after sampling the number of ego-vertices indicated at the x-axis ticks, but the symbols are shifted to the right and left, respectively, for better visibility. Snowball iteration transitions are indicated by vertical dashed lines (averaged over the simulation ensemble). The x-axis is log-scaled.



(a) *random* (b) *hepth*

(c) *condmat* (d) *slashdot*

Figure 4: **Relative error** ϵ ((b) and (c)) and **absolute error** Δ ((a) and (d)) of the clustering coefficient estimated with the naive estimator $\hat{C}_{(\text{obs})}$ (◯), the weighted sample mean estimator $\hat{C}_{(\text{wsm})}$ (△), and the RDS-estimator $\hat{C}_{(\text{rds})}$ (◇). Thy symbol indicates the ensemble mean, the lower whisker indicates the 0.1 quantile, and the upper whisker indicates the 0.9 quantile. The data is extracted after sampling the number of ego-vertices indicated by the x-axis ticks, but the symbols are shifted to the right and left, respectively, for better visibility. Snowball iteration transitions are indicated by vertical dashed lines (averaged over the simulation ensemble). The x-axis is log-scaled.

where $n^{(\leq i)}$ denotes the number of ego-vertices sampled up to and including iteration i, $n^{(<i)}$ denotes the number of ego-vertices sampled strictly before iteration i, and $a^{(<i)}$ denotes the number of alter-vertices sampled strictly before iteration i. In words, the numerator corresponds to the number of vertices that have responded to an inquiry before or in iteration i, and the denominator corresponds to the number of all vertices that have been inquired strictly before iteration i and hence could have replied before or in iteration i.

The previously developed estimators are extended for the response rate by replacing $\hat{\pi}_v$ with $\hat{\pi}_{v,\alpha}$. It turns out that α cancels out in estimators based on the weighted sample mean (Eq. 11 and Eq. 19). An exception is the estimator for the degree correlation (Eq. 16) because the inclusion probability of an edge does not scale linearly with the inclusion probability of the adjacent vertices.

Particular care is required for the estimation of transitivity. Section 4.2.3 already mentions the issue of missing edges m_v between alter-vertices. With low response rates, this issue becomes even more distinct. If the neighbours of ego-vertex v are non-responding, it is likely that the sample data misses existing edges between the neighbours. This means that the value of m_v in Eq. 17 is likely to be underestimated. The number of edges m_v needs therefore to be estimated.

Let n_v denote the number of neighbours of ego-vertex v that are in $\mathcal{S}$. In consequence, $k_v - n_v$ represents the number of neighbours of v that are not in $\mathcal{S}$, i.e. neighbours of v that are either non-responding or have not been enquired. Further, denote by p the average probability of an edge connecting two neighbours of v. The estimated number of missing edges is p times the number of possible edges between unsampled neighbours $(k_v - n_v)(k_v - n_v - 1)/2$. The estimated number of edges $\hat{m}_v$ is then the sum of actually observed edges $\tilde{m}_v$ and the estimated number of missing edges:

$$\hat{m}_v = \tilde{m}_v + p\frac{1}{2}(k_v - n_v)(k_v - n_v - 1) \ . \tag{23}$$

Probability p can be obtained from the hitherto sampled data as the fraction of the sum of observed edges $\sum_{v\in\mathcal{S}} \tilde{m}_v$ over the sum of observable opportunities to connect to neighbours of v. Note that the observable opportunities are not $k_v(k_v-1)/2$ since edges between non-responding neighbours cannot be observed. Instead, it is the number of possible edges that can occur between responding neighbours plus the number of possible edges that can occur between responding and non-responding neighbours.

The alter-vertices of the last iteration, i.e. those that

are not enquired yet, can be treated as non-responding vertices. This allows to estimate m_v also for the ego-vertices of the last iteration. Thus, there is no need to exclude the ego-vertices on the last iteration, as it is done in Sec. 4.2.3.

4.3.2. Simulation results

The sensitivity of the weighted sample mean estimator with respect to the response rate α is visualised in Fig. 5–7. These figures show the relative error of the estimated value in dependency of the number of ego-vertices and response rate.

Varying α reveals the sensitivity of $\hat{\bar{k}}_{(\text{wsm})}$ towards the response rate. The effect becomes visible in that the estimation errors of $\hat{\bar{k}}_{(\text{wsm})}$ correlate with the iteration transitions. The "relief" in Fig. 5 stretches along the iteration lines, meaning technically that the level curves of the error landscape are aligned with the iteration lines. A snowball sample with a low response rate can be considered as a snowball sample on a thinned-out network. In either case, the global progression of the snowball through the network is similar. The results suggest that this coverage is decisive for the evolution of the estimation error, which evolves according to the number of iterations but not according to the number of sampled vertices. Again, Fig. 5 shows that with a greater variance of the degree distribution the estimator's performance becomes worse. Regarding the *random* network and *hepth* network, the relative error seldom exceeds 10 %, even with low response rates. However, for the *condmat* network and *slashdot* network, which both exhibit a high variance of the degree distribution, the relative error is up to 40 %.

A strong correlation between $\hat{r}_{(\text{wsm})}$ and α is observed in Fig. 6. Analogous to the mean degree estimates, the "relief" stretches along the iteration transitions. However, no rule can be identified if a high or low response rate improves the estimator's performance. While for the *hepth* network a high response rate is beneficial, it is rather the contrary for the *condmat* network. Just along the transition of the third to the fourth iteration the estimator approximately predicts the true degree correlation. The correlation is also visible in the *slashdot* network where the absolute error ranges from 0 to 0.2. To the contrary, considering the *random* network the correlation can not be identified. Yet, also in the latter two networks no predictable trend can be identified.

The estimator $\hat{C}_{(\text{wsm})}$ is quite robust towards variations in α regarding the *condmat* network (Fig. 7(b)). Just with sample sizes below 200 ego-vertices and response rates below 0.4 it shows some divergence. Similar, yet less pronounced, is the behaviour regarding the *hepth* network. For sample sizes above 300 ego-vertices the estimation quality is nearly independent from α. Apart from the regions with small sample sizes and low response rate, the estimation error in both networks does not exceed 10 %. In the *slashdot* network the absolute error varies between -0.04 and 0.04 but without any identifiable pattern. In the *random* network it is in all areas approximately zero.

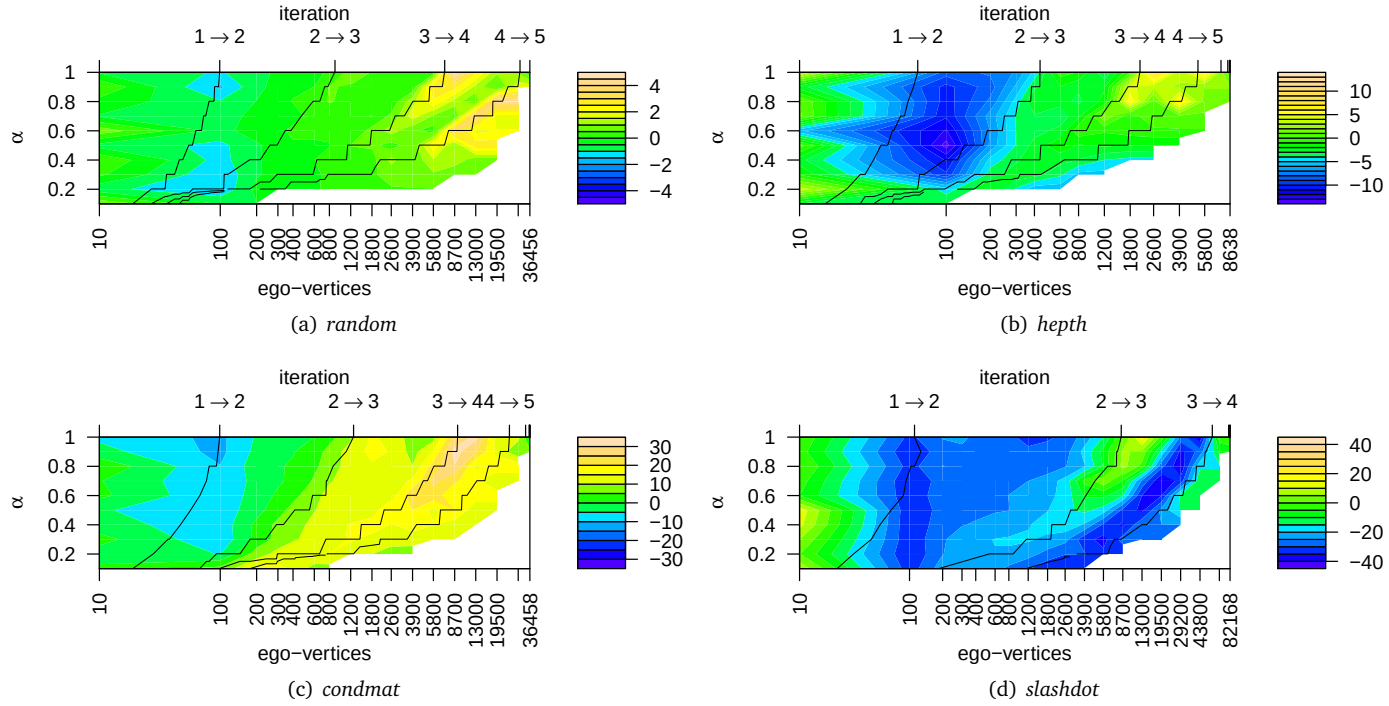
5. Conclusion

This article proposes a design-based inference approach to estimate topological network properties from data obtained with a snowball sampling design. The approach estimates the inclusion probability of a vertex based on a simple approximation that only involves its degree and the total number of vertices sampled so far. Using the estimated inclusion probabilities, estimators for the mean degree, the degree correlation, and the clustering coefficient are obtained.

Different from the widely used RDS-estimator, which assumes sampling with replacement, our approach relies on sampling without replacement. As a consequence, the proposed estimator is consistent in that it predicts the true values if the entire network is sampled. For small sample sizes, where sampling with replacement approximates sampling without replacement, our estimator is at least as good as the RDS-estimator. Yet, one has to admit that the RDS-estimator is in fact designed for a non-random selection of seed-vertices. Compared to more complex inference methods, such as model-based approaches, the present method has no complex computational requirements and is straightforward to apply to real-world survey data.

Various simulation experiments are conducted to investigate the estimators' behaviour under different conditions. These include four test networks and variations in the response rate. From the simulation studies, five major conclusions can be drawn:

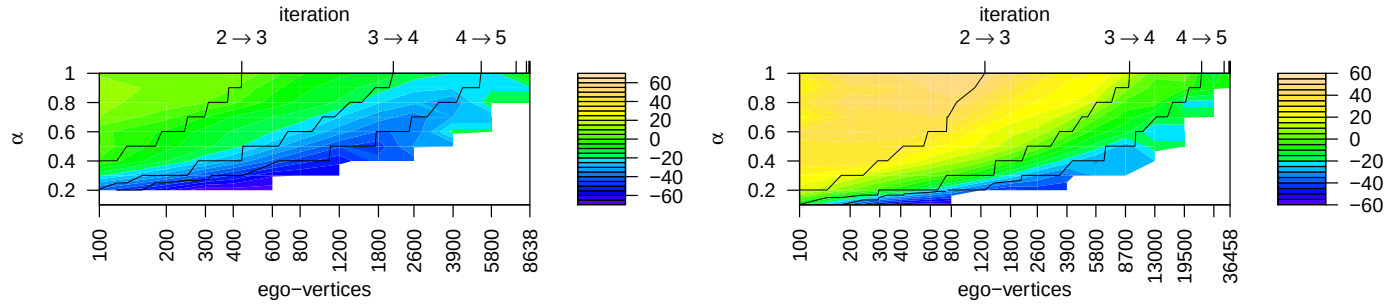
- The oversampling of vertices with high degree scales with the variance of the network degree distribution so that the estimator performs worse for networks with high variance. This indicates that the proposed estimator counteracts the bias but does not completely remove it.
- The estimator is quite robust towards variations in the response rate, which can be of practical importance. Yet, the overall dynamics of the error over the iterations do not follow a predictable trend.
- Considering networks that do not exhibit a too broad degree distribution, say, with a maximum degree of $k < 100$, the mean degree estimation is quite satisfactory even with low response rates.
- With just 300 sampled vertices, the predictions of the clustering coefficient are quite precise and are almost independent from the response rate.
- The estimation error of the degree correlation peaks up to 60 % rendering the proposed estimator not reliable for this particular purpose.



(a) *random* (b) *hepth*

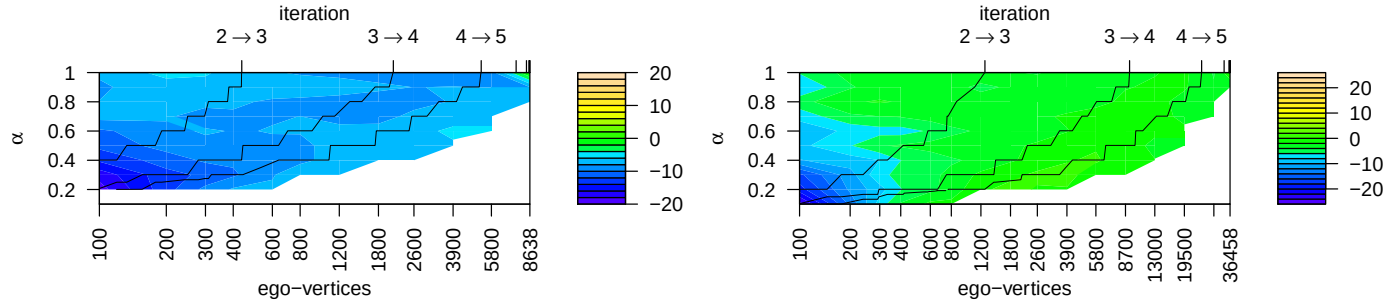
(c) *condmat* (d) *slashdot*

Figure 5: (Colour online) Relative error [%] of the mean degree (averaged over the simulation ensemble) estimated with $\hat{\bar{k}}_{(\mathrm{wsm})}$ depending on the response rate and the number of ego-vertices. Simulation with ten seed-vertices. Colours indicate the relative error and are adjusted so that green colour indicates no error.



(a) *hepth* (b) *condmat*

Figure 6: (Colour online) Relative error [%] of the degree correlation (averaged over the simulation ensemble) estimated with $\hat{r}_{(\mathrm{wsm})}$ depending on the response rate and the number of ego-vertices. Simulation with ten seed-vertices. Colours indicate the relative error and are adjusted so that green colour indicates no error.



(a) *hepth* (b) *condmat*

Figure 7: (Colour online) Relative error [%] of the clustering coefficient (averaged over the simulation ensemble) estimated with $\hat{C}_{(\mathrm{wsm})}$ depending on the response rate and number of ego-vertices. Simulation with ten seed-vertices. Colours indicate the relative error and are adjusted so that green colour indicates no error.

What has not been addressed in this article is the estimation of the variance. Information about confidence intervals would be of high practical use, yet remains open for deeper research. As it is shown by Goel and Salganik (2009) in the context of Respondent-Driven Sampling, having "thinner" networks with longer recruitment chains reduces the variance of estimates. While longer recruitment chains can be obtained by artificially holding the response rate low (by expanding only to a subset of neighbours), in the simulation experiment no clear evidence is found that the findings of Goel and Salganik hold also for the present sampling design.

Further research should also focus on the effects on variations of the number of seeds. Preliminary simulation results on this indicate a similar behaviour as with variations in the response rate. That is, initialising the snowball with more seeds yields in more sampled ego-vertices per iteration. The global progression through the network, however, remains unaffected.

Two assumptions that have been made to simplify the simulation studies are arguable: First, the total number of vertices N is usually unknown but is required for the estimation of the inclusion probability (Eq. 4). Considering large networks and relative small sample sizes, an educated guess of N is sufficient because the derivative of the estimator (Eq. 4) with respect to N approaches zero at rate N^2. Second, the response rate is assumed to be equally distributed and constant throughout the entire sampling process. One can expect to observe more heterogeneity with respect to the response behaviour in a real-world application (Kowald et al., 2010). For instance, people with large personal networks may be to busy to participate in the survey. While an adaptation of the inference framework appears possible, this is likely to require a model of the underlying population.

An aspect that is still open for further research is the estimation of global network parameters, such as the network diameter, closeness, or betweenness. An estimation of such parameters will be quite challenging: Even the estimation of a two-point property, in this case the degree correlation, turned out to be by no means trivial. Some work in this direction has been done (Lee et al., 2006; Ebbes et al., 2008; Ye et al., 2010), but more insights would in particular provide a sound basis for the modelling of the spreading of diseases or rumours. Finally, snowball sampling is the designated tool for such studies as it provides an effective method to obtain connect egocentric networks (see for instance Illenberger et al., 2011).

6. Acknowledgement

We thank Kai Nagel for helpful suggestions and support. This work was funded by the VolkswagenStiftung within the project "Travel impacts of social networks and networking tools".

References

Amaral, L. A. N., Scala, A., Barthélémy, M., Stanley, H. E., 2000. Classes of small-world networks. Proceedings of the National Academy of Sciences of the United States of America 97 (21), 11149–11152.

Atkinson, R., Flint, J., 2001. Accessing hidden and hard-to-reach populations: Snowball research strategies. Social Research Update 33.

Chow, M., Thompson, S. K., 2003. Estimation with link-tracing sampling designs – a bayesian approach. Survey Methodology 29 (2), 197–205.

Ebbes, P., Huang, Z., Rangaswamy, A., Thadakamalla, H. P., 2008. Sampling large-scale social networks: Insights from simulated networks. In: 18th Annual Workshop on Information Technologies and Systems. Paris, France.

Erdős, P., Rényi, A., 1959. On random graphs. Publicationes Mathematicae Debrecen 6, 290–297.

Frank, O., 1979. Estimation of population totals by use of snowball samples. Academic Press, New York, pp. 319–346.

Frank, O., Snijders, T., 1994. Estimating the size of hidden population using snowball sampling. Journal of Official Statistics 10 (1), 53–67.

Gile, K. J., 2011. Improved inference for respondent-driven sampling data with application to HIV prevalence estimation. Journal of the American Statistical Association 106 (493), 135–146.

Gile, K. J., Handcock, M. S., 2010. Respondent-driven sampling: An assessment of current methodology. Sociological Methodology 40 (1), 285–327.

Gjoka, M., Kurant, M., Butts, C. T., Markopoulou, A., 2011. Practical recommendations on crawling online social networks. IEEE Journal on Selected Areas in Communications 29 (9), 1872–1892.

Goel, S., Salganik, M. J., 2009. Respondent-driven sampling as markov chain monte carlo. Statistics in Medicine 28, 2202–2229.

Goodman, L. A., 1961. Snowball sampling. The Annals of Mathematical Statistics 32 (1), 148–170.

Handcock, M. S., Gille, K. J., 2010. Modeling social networks from sampled data. The Annals of Applied Statistics 4 (1), 5–25.

Heckathorn, D. D., 1997. Respondent-driven sampling: A new approach to the study of hidden populations. Social Problems 44 (2), 174–199.

Heckathorn, D. D., 2002. Respondent-driven sampling ii: Deriving vaild population estimates from chain-referral samples of hidden populations. Social Problems 49 (1), 11–34.

Illenberger, J., Kowald, M., Axhausen, K. W., Nagel, K., 2011. Insights into a spatially embedded social network from a large-scale snowball-sample. European Physical Journal B 84 (4), 549–561.

Johnson, J., Boster, J., Holbert, D., 1989. Estimating relational attributes from snowball samples through simulation. Social Networks 11, 135–158.

Kowald, M., Frei, A., Hackney, J., Illenberger, J., Axhausen, K., 2010. Collecting data on leisure travel: The link between leisure acquaintances and social interactions. Procedia — Social and Behavioral Sciences 4 (1), 38–48.

Kurant, M., Gjoka, M., Butts, C. T., Markopoulou, A., 2011. Walking on a graph with a magnifying glass: Stratified sampling via weighted random walks. In: SIGMETRICS. San Jose, California.

Kwanisai, M., 2006. Estimation in network populations. In: Proceedings of the Joint Statistical Meetings. pp. 3285–3291.

Lee, S. H., Kim, P.-J., Jeong, H., 2006. Statistical properties of sampled networks. Physical Review E 73 (016102), 1–7.

Leskovec, J., Lang, K. J., Dasgupta, A., Mahoney, M. W., 2009. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. Internet Mathematics 6 (1), 29–123.

Newman, M. E. J., 2001. Scientific collaboration networks. I. network construction and fundamental results. Physical Review E 64 (016131).

Newman, M. E. J., 2002. Assortative mixing in networks. Physical Review Letters 89 (20), 1–4.

Salganik, M. J., Heckathorn, D. D., 2004. Sampling and estimation in hidden populations using respondent-driven sampling. Sociological Methodology 34, 193–239.

Särndal, C.-E., Swensson, B., Wretman, J., 1992. Model Assisted Survey Sampling. Springer-Verlag.

Snijders, T. A. B., 1992. Estimation on the basis of snowball samples: How to weight. Bulletin de Méthodologie Sociologique 36, 59–70.

Thompson, S. K., 2006. Adaptive web sampling. Biometrics 62 (4), 1224–1234.

Thompson, S. K., Frank, O., 2000. Model-based estimation with link-tracing sampling designs. Survey Methodology 26 (1), 87–98.

Volz, E., Heckathorn, D. D., 2008. Probability based estimation theory for Respondent Driven Sampling. Journal of Official Statistics 24 (1), 79–97.

Wasserman, S., Faust, K., 1994. Social Network Analysis: Methods and Applications. Cambridge University Press.

Watts, D. J., Strogatz, S. H., 1998. Collective dynamics of 'small-world' networks. Nature 393 (440-442).

Wejnert, C., 2010. Social network analysis with respondent-driven sampling data: A study of racial integration on campus. Social Networks 32, 112–124.

Ye, Q., Wu, B., Wang, B., 2010. Distance distribution and average shortest path length estimation in real-world networks. In: Cao, L., Feng, Y., Zhong, J. (Eds.), Advanced Data Mining and Applications. Vol. 6440 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 322–333.