UNIVERSITY OF ZAGREB
FACULTY OF ELECTRICAL ENGINEERING AND COMPUTING

MASTER THESIS No. 1419

Influence of Neighbours on One's Own opinion

Ivan Bestvina

Zagreb, June 2017.

UNIVERSITY OF ZAGREB
FACULTY OF ELECTRICAL ENGINEERING AND COMPUTING
MASTER THESIS COMMITTEE

Zagreb, 3 March 2017

MASTER THESIS ASSIGNMENT No. 1419

Student: **Ivan Bestvina (0036475086)**
Study: Computing
Profile: Software Engineering and Information Systems

Title: **Influence of Neighbours on One's Own opinion**

Description:

Using a network representation analyse the influence of neighbours on one's own attitudes and opinions on several simulated and real datasets-collected on social networks. Determine the influence of the first and second neighbourhood of a given node. Compare the average opinion in the network with the opinions of nodes depending on opinions of their neighbours. Furthermore, analyse for each node its prediction of the average opinion depending on its own opinion and opinions of its neighbours. Finally, analyse the estimation of average opinion from a sample of nodes for cases when each node report (1) its own opinion and (2) the average of its own opinion and opinions of its neighbours. All methods and auxiliary scripts should be developed in Python. The code is to be documented using comments and should follow the appropriate style guides and hosted on a publicly available Github repository.

Issue date: 10 March 2017
Submission date: 29 June 2017

Mentor:
Associate Professor Mile Šikić, PhD

Committee Chair:
Assistant Professor Igor Mekterović, PhD

Committee Secretary:
Assistant Professor Boris Milašinović, PhD

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
ODBOR ZA DIPLOMSKI RAD PROFILA

Zagreb, 3. ožujka 2017.

DIPLOMSKI ZADATAK br. 1419

Pristupnik: **Ivan Bestvina (0036475086)**
Studij: Računarstvo
Profil: Programsko inženjerstvo i informacijski sustavi

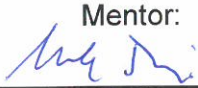
Zadatak: **Utjecaj susjeda na vlastito mišljenje**

Opis zadatka:

Koristeći mrežni prikaz analizirati utjecaj susjeda na vlastite stavove i mišljenja koristeći nekoliko simuliranih i stvarnih skupova podataka prikupljenih na društvenim mrežama. Odrediti utjecaj prvih i drugih susjedstva na određeni čvor. Usporediti prosječan stav cijele mreže sa stavom pojedinih čvorova u zavisnosti o njihovim susjedstvima. Dodatno, za svaki čvor analizirati njegovo predviđanje prosječnog stava mreže u zavisnosti o njegovom stavu i stavu njegovih susjeda. Konačno, analizirati procjenu prosječnog stava mreže na temelju uzorka čvorova za dva slučaja: (1) kada svaki čvor dojavljuje vlastiti stav i (2) kada svaki čvor dojavljuje prosjek između svog stava i stava svojih susjeda. Sve metode i popratne skripte trebaju biti napisane u programskom jeziku Python. Programski kod treba biti dokumentiran i javno dostupan preko repozitorija GitHub.

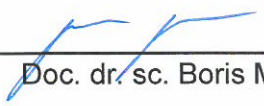
Zadatak uručen pristupniku: 10. ožujka 2017.
Rok za predaju rada: 29. lipnja 2017.

Mentor:



Izv. prof. dr. sc. Mile Šikić

Djelovođa:



Doc. dr. sc. Boris Milašinović

Predsjednik odbora za diplomski rad profila:

Doc. dr. sc. Igor Mekterović

I would like to thank my mentor, Prof. Mile Šikić, my family, Nives, and friends for their help and support.

CONTENTS

List of Figures

LIST OF TABLES

1. Introduction

"Human is by nature a social animal", wrote Aristotle in the 4th century BC. Today, almost two and a half thousand years later, humans are connected more than ever before, spanning between them vast complex networks of friendships and collaborations. These networks, much like humans themselves, grow and evolve, and are influenced in their shape by a wide range of factors, including technology, economy and politics.

As these factors become more and more dynamic, and their influence more and more direct, social networks change more rapidly. And as they change, the factors which influenced them react to that change, and the influence loops back. It is evident then that social networks and social processes are tightly coupled, and that one is incomprehensible without the other. We can never fully discern, for example, the influence of friends on one's opinions, and the influence of opinions on the choice of friends. But we can measure the correlation of these characteristics, and we can predict one from the other.

This is the main theme of this thesis. It tries to tap into these mutual influences, to better understand their mechanics and predict their outcomes. More specifically, it explores how polarization over political issues reflects on the social network topology and how, by seeing only parts of these networks, we can infer their global characteristics.

First section covers some general topics about social networks. Following sections describe how community voting preferences are distributed, how this distribution is tied to the network topology, and how we can use this knowledge to predict person's votes. Last part, after defining the Kronecker graph model as the basis of social network modelling, continues to explain how this model can be extended to model and analyse public opinion polarization. Finally, model is tested on real and simulated datasets, and results are presented.

2. Complex social networks

When we talk about social networks, we generally talk about graphs in which people are represented as vertices, or nodes, and friendships, acquaintances, collaborations, or, as is the case with the popular notion of "social networks", online contacts, are represented by edges. Besides the meaning of edges between nodes, these networks differ substantially in their size, shape, and their dynamics. They can represent only a small group of people, as is the case with the famous Zachary's karate club described in Zachary (1977) and illustrated in 2.1, or a large part of the world (online social networks, for example). They can be static, showing us just a snapshot of human interactions, or they can be dynamic, temporal, revealing how those interactions changed through time.

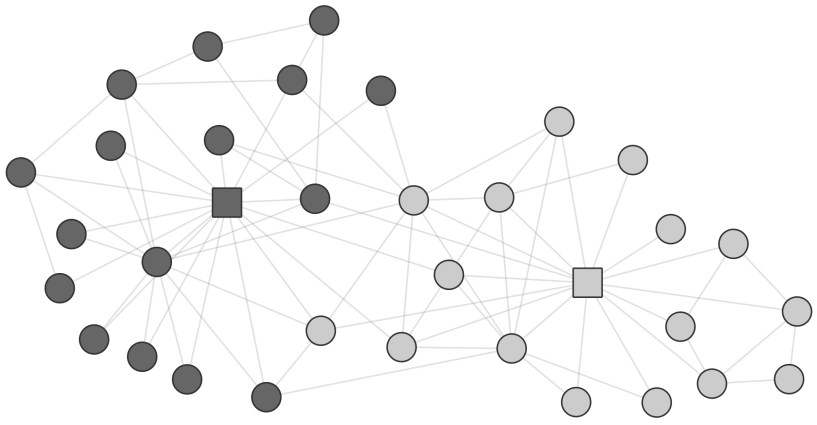


Figure 2.1: Zachary's karate club is a social network of a university karate club. It was described by Wayne W. Zachary in Zachary (1977) as an example of a split of the network in two clusters when a polarizing issue emerges. In this case, there was a conflict between the club president and an instructor over the price of karate lessons. By using the maximum flow – minimum cut Ford–Fulkerson algorithm, Zachary identified the correct side of the dispute for all but one members. Colour represents the two sides, with square nodes being the leaders.

In this chapter, we will briefly explain the main commonalities between these networks, which would later prove very useful in their analysis and modelling.

2.1. Social neighbourhood

Throughout this thesis, we will write in great detail about social neighbourhoods, but first, we need to define what they really are. As we already explained in the previous paragraph, the term *social network* covers a wide range on networks. Because of this, *social neighbourhood* is also a loose term. Here, we will focus mostly on acquaintance networks – social networks in which an edge between nodes A and B means that "Person A *knows* person B". The largest data source of these networks are online social networks, if we assume that, for example, a friendship on Facebook implies an acquaintance in real life. So, when we are mentioning someone's social neighbourhood, we are referring to people who are acquainted with that person.

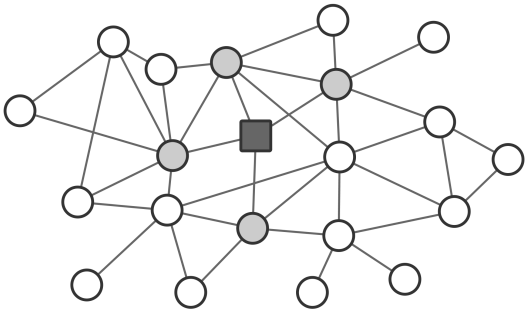
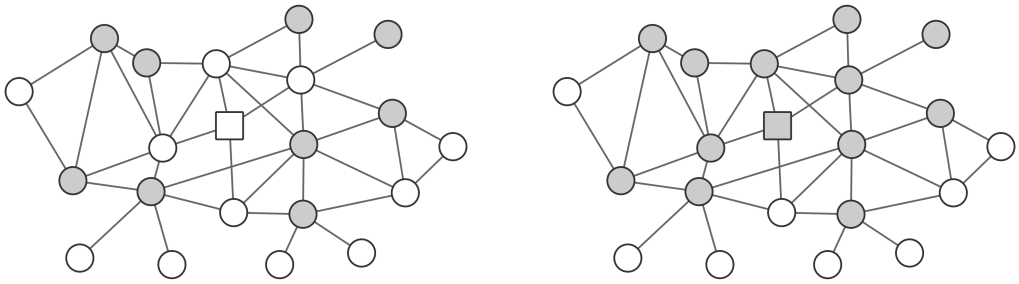


Figure 2.2: Example of node neighbourhood. Shaded nodes are neighbours of the square node.

Second neighbourhood

Second neighbourhood can be defined in multiple ways. For example, it can be defined as the set of nodes which are two steps away from the central node. Similarly, it can be a set of nodes to which there exists a path of length two from the central node. These two definitions, although they may seem equivalent, are quite different, as shown in the figure 2.3: the second one may contain first neighbours also, and it contains the central node itself.

Yet another definition might deal not with sets, but with lists of nodes, and use each second neighbour multiple times, depending on how many different two-step paths exist between it and the central node. This third definition will best reflect the model we will later describe, so it is the one we will use.



(a) **(b)**

Figure 2.3: Two definitions of the second neighbourhood. Grey nodes represent second neighbours of the rectangular node. In a) second neighbours are nodes which are two steps away from the central node. In b) second neighbours are nodes which have a path of two steps to the central node.

2.2. Social influence

Throughout this thesis we will often mention *social influence*. This term can, however, be misleading, because it implicates that there is a one way effect of our society, mainly our friends, on our opinions and actions. This is, of course, wrong, as we, at least partially, choose our friends and the society we live in based on our interests, beliefs, etc.

Here, we will not delve deeper into the differentiation of these effects. When we mention *social influence*, we will refer simply to the correlation between our and our friend's opinions, without exploring how this correlation emerges.

2.3. Networks characteristics

There are many different network characteristics which most biological, social, and technological networks share. Here, we will concentrate on those which are important for this thesis. For a more in-depth exploration of these, and others, one should consult *Networks: An Introduction*, Newman (2010).

2.3.1. Degree distribution

Degree of a node in a graph is the number of edges connected to that node, or, in other words, the size of its neighbourhood. Complex networks degree distributions are one of the most researched topics in the field of complex network theory. The whole theory, it could be argued, was initiated by this research, starting with the analysis of

the number of citations scientific papers receive in de Solla Price (1965). Later, in Barabási and Albert (1999), the idea was revisited, this time applied to the number of links pointing to the world wide web pages. In both cases, and many, many others, it was found that the degree distributions are heavy tailed. And from both findings, similar random network generator models emerged, utilizing the idea of preferential attachment (Price (1976) and Barabási and Albert (1999) respectively).

These random network models generate the so-called scale-free networks, distinct by their power-law degree distributions, expressed formally as $P(k) \sim k^{-\lambda}$, where $P(k)$ is the probability of a node having degree k, and λ is usually between 2 and 3. The term *scale-free* originates from the fact that their degree distributions remain scale-invariant while they grow.

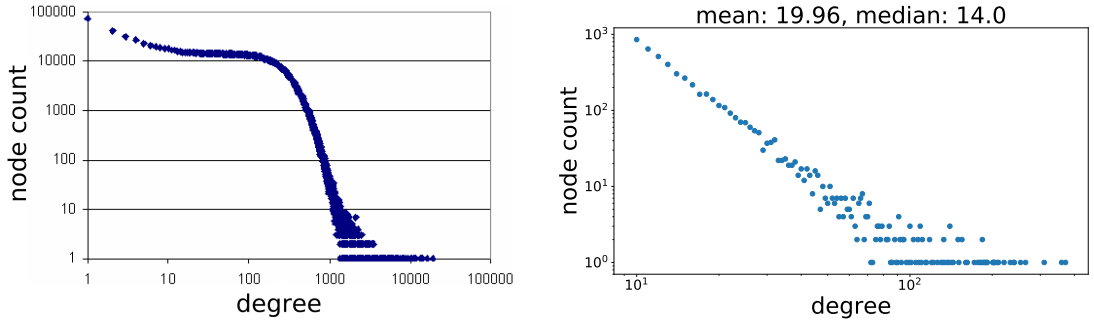
Barabási–Albert (BA) model mentioned above remains one the most widely used random network models, and is often the default approach to modelling different kinds of network processes. But, because of the many discrepancies between the BA model and real-world social networks, it is not suitable here. One of these discrepancies is the fact that, unlike other real-world networks, social network degree distributions do not follow power laws exactly. An example of such network is the Croatian constitutional referendum of 2013 network, degree distribution of which is shown in figure 3.3b.

One of the main problems with social network modelling using the BA model is the fact that it is uncommon for a social network to have the largest proportion of lowest degree nodes (an order of magnitude larger than the next lowest degree). In fact, peak in the degree distribution of social networks is often much closer to its median. Other distribution shapes are also common, such as Facebook users degree distribution from Golder et al. (2007), shown in figure 2.4a.

2.3.2. Density

Network density, denoted by ρ, is equal to the ratio of the number of edges E and the number of nodes N, which is also equal to double the average degree. Defined this way, density does not seem to convey any new understandings not already present in the degree distribution. But density becomes very important once we look at real-world evolving networks. If we measure the number of nodes $N(t)$ and the number of edges $E(t)$ through time, as network grows, we observe the *densification power law*, which can formally be expressed as:

$$E(t) \propto N(t)^a, \tag{2.1}$$



(a) **(b)**

Figure 2.4: Comparison of degree distributions of a) part of the Facebook network (Golder et al. (2007)), and b) network generated with the Barabási–Albert model. Note how BA network follows the power-law distribution, while Facebook is a combination of the power-law and bell-shaped distributions.

with a typically greater than 1 (Leskovec et al. (2010)). This means that, while the network grows, it densifies, and the average degree increases.

We can intuitively think about this if we imagine a small, remote society with only a few dozen members, and a large urban area with a few million people. It is expected then that the size of the average social neighbourhood is much larger in the latter case.

This densification power law is not obeyed by the BA model, which is another reason to look elsewhere for social network modelling.

2.3.3. Diameter

Let G be a connected graph. Then the diameter d of G is the length of the shortest path between two most distant nodes. More formally, it is the largest *geodesic distance*[1] of G.

This measure, much like density, gains its full meaning only when viewed in the context of growing networks. In the beginning, while networks are small, diameter grows with them as expected. But after a certain threshold, it surprisingly stabilizes, or even starts to reduce (Leskovec et al. (2005)). This diameter reduction is closely connected to the densification of the network (denser network – smaller diameter), and as such is also missed by the BA model.

Networks which express this behaviour are called small-world networks, a concept which was popularised by Stanley Milgram in Travers and Milgram (1967). Later, this

[1] Shortest path between two nodes in a graph is called a *graph geodesic*, so its length is called the *geodesic distance*.

became known as the *six degrees of separation* phenomenon, which states that most of the people are connected trough no more than six steps of acquaintances.

Because diameter is such a sensitive measure, and is undefined for unconnected networks, it is rarely use as is in practice. Instead, a more robust measure called *integer effective diameter* (IED), suggested by Tauro et al. (2001), is used. IED is defined as the minimum number of steps in which some fraction q (usually $q = 0.9$) of the node pairs are connected. Also interesting is the function $g(h)$ which denotes the fraction of node pairs separated by most h steps.

IED was further generalized with *effective diameter* by Leskovec et al. (2005), which is defined as the point x at which the function $g(x)$ achieves the value 0.9, where the function $g(x)$ is defined over all positive real numbers x by linearly interpolating between the points $(h, g(h))$ for integer values of h.

2.3.4. Clustering

Networks clustering is quantified by their clustering coefficient, which is formally expressed as:

$$C = \frac{3 \times \text{number of triangles}}{\text{number of connected triplets}}. \tag{2.2}$$

In other words, clustering defines the probability of one's two neighbours being connected. High clustering coefficients lead to interesting local structures, and can increase the network diameter, if the density is constant. It is also related to an interesting phenomenon called homophily, which, in the case of networks with two types of nodes, can lead to polarization.

Homophily

Homophily is a characteristic of networks in which similar nodes have a higher chance of being connected. It can occur in many different ways, based on the definition of node similarity. For instance, in social networks, similar nodes may represent similar demographic characteristics, similar interests, or similar topological characteristics of nodes (their degree and centrality). In this thesis, we concentrate on opinion similarities. This, some authors claim, leads to the emergence of information bubbles (Pariser (2012), and Bozdag and van den Hoven (2015)).

Polarization

Many recent elections and referendums, such as Brexit, United States presidential election of 2016 and Croatian constitutional referendum of 2013 (data of which we will focus on in this thesis), were all described by the media as polarizing issues. What this means in the context of network homophily, is that networks split into two large clusters, similarly to the network shown in figure 3.1. Multiple authors, such as in Del Vicario et al. (2016), have suggested that this kind of polarization leads to the emergence of echo chambers and information filters, which further negatively influence users' emotions regarding the subject.

One of the main ideas behind this thesis, which will later be explained in detail, is that we could predict the outcome of such elections and referendums if we could somehow measure or infer the sizes of these two clusters.

3. Neighbourhood vote distribution

Imagine a social network of a country during one of the polarizing political processes we mentioned before. Similarly, one can imagine any other, non-social network which has polarized nodes. Because complete large social networks are infeasible to obtain, we will, for illustration purposes, use the network of US political blogs from 2005, collected and presented in Adamic and Glance (2005), shown in figure 3.1. It is evident that the network is naturally split into two clusters - liberal and conservative. This split is the result of a higher connectivity between blogs of the same political preference, than between differently oriented ones. This difference in connectivity will be the basis of modelling presented in chapter 7.

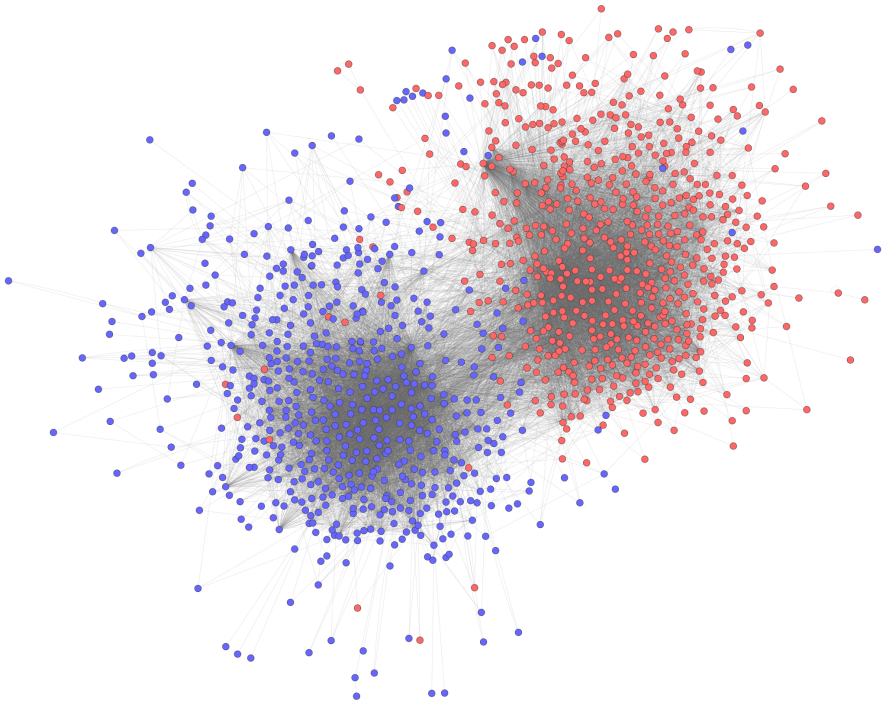


Figure 3.1: US political blogs network from 2004. Blue nodes represent liberal, and red nodes conservative blogs. Edge between two nodes represents a hyperlink from one blog to another. Data collected and presented in Adamic and Glance (2005). Note the evident polarization between two political options.

In this chapter we are interested in the distribution of neighbourhood opinions, or, in the case of political blogs, neighbourhood political preferences. Generally, we will talk about *votes*, as this thesis focuses mainly on elections and referendums.

Neighbourhood vote distribution is defined as the distribution of fractions of neighbourhoods which vote for some option. Following is the more formal definition of this distribution.

Let $G = (N, E)$ be a (social) graph with a set of nodes N and a set of edges E. Let $deg(i)$ be the degree of node i. Let $Nb_1(i) \equiv Nb(i)$ be the first neighbourhood of node i. Similarly, let $Nb_2(i)$ be the second neighbourhood. Let $v : N \to V$ be the *voting* function which maps each node of G to a vote from V. Term *vote*, as we mentioned before, is here used to represent all kinds of opinions, beliefs and interests, depending on what exactly we are modelling. This thesis is focused on binary votes, so let $V = \{a, b\}$, where a and b are two different voting options. For consistency, we will refer to the voting option more frequent in the data as option a. Finally, let $q_1(i, k) \equiv q(i, k)$ be the fraction of first neighbours of node i for which $v(j) = k$, and, equivalently, $q_2(i, k)$ the fraction of the second neighbourhood. Let $Q(k)$ be the fraction of all nodes who vote k. Formally,

$$q(i,k) = \frac{|\{j : j \in Nb(i), v(j) = k\}|}{deg(i)} \tag{3.1}$$

$$q_2(i,k) = \frac{|\{j : j \in Nb_2(i), v(j) = k\}|}{|Nb_2(i)|} \tag{3.2}$$

$$Q(k) = \frac{|\{i : i \in N, v(i) = k\}|}{|N|} \tag{3.3}$$

First and second neighbourhood vote distribution is now the distribution of $q(i, k)$ and $q_2(i, k)$ respectively, usually with $k = a$. Because of the small-world phenomenon described in section 2.3.3, third neighbourhood often contains most of the nodes in the network, and as such is not really interesting.

In the following plots, we will describe the first and second neighbourhood vote distributions of two real-world networks: political blogs (Adamic and Glance (2005)) shown in figure 3.1, and the social network of Croatian constitutional referendum of 2013, shown in figure 3.2. Latter is composed of Facebook users who participated in the online poll Piškorec et al. (2016). Note that the participant votes are highly biased – 75% of them voted *No*, while only 34% of the total population did so. Such bias is very common, especially in online polls where participants are mostly young people living in urban areas. Apart from that, this poll has been publicized mostly

through the left-leaning media, which further increased the bias. Trying to find a good election outcome predictor from such biased data was one of the main motivations behind this research. Overview of these two networks is given in table 3.1, and their degree distributions shown in figure 3.3.

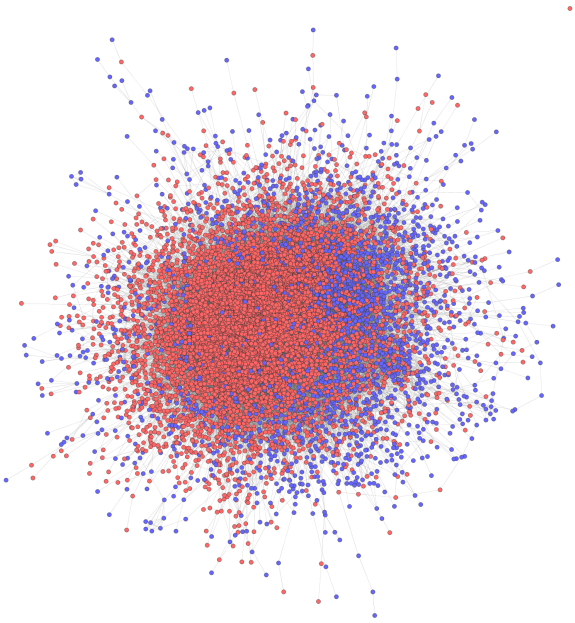
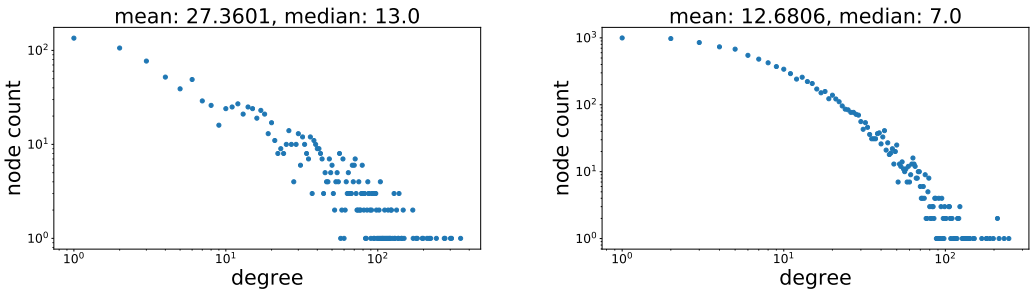


Figure 3.2: Croatian constitutional referendum of 2013 network. Blue nodes represent "Yes", and red nodes "No" voters. Edge between two nodes represents a Facebook friendship between them. Data collected and presented in Piškorec et al. (2016). Note that polarization, although less evident than in the case of the political blogs network, still exists.



(a) Political blogs network.

(b) Croatian referendum network.

Figure 3.3: Degree distributions. Note how neither (especially **b**) is following the power law exactly.

	Political blogs	Referendum
$\|N\|$	1490	10174
$\|E\|$	19025	64506
ρ	12.77	6.34
d	7	13
C	0.321	0.164
a	Conservative	"No"
b	Liberal	"Yes"
$Q(a)$	52.0%	75.4%

Table 3.1: Overview of main network characteristics of political blogs and Croatian constitutional referendum networks.

3.1. First neighbourhood

Political blogs

Figure 3.4 shows the first neighbourhood distribution of political blogs network. Plot is divided into two colours – blue bars represent liberal, and red bars conservative blogs. Horizontal axes represents the fraction of first neighbours who are conservative, and vertical a fraction of nodes with that fraction of conservative first neighbours. Mean and variance of the two distributions are denoted above the graph. It is evident that there exists a significant division between the two types of nodes, and that it is easy to distinguish them based on their neighbourhoods. We already observed this in figure 3.1. Another very interesting feature of this distribution is the amount of nodes which are surrounded exclusively with their voting peers (total of about 50% of nodes).

Croatian referendum

Figure 3.5 shows the first neighbourhood distribution of the Croatian referendum network. Here, red bars represent "No" votes, while blue bars represent "Yes" votes. Horizontal axes represents the fraction of first neighbours who vote "No".

This plot is significantly different from the previous, political blogs one. This is due to the high bias in node sampling – "No" voters were much more likely to participate in the poll. As we will show later, this bias can easily be modelled by sampling nodes with probability proportional to their neighbourhood "No" voters fraction. Because of this, "No" voters distribution is similar to the conservative political blogs one, but "Yes" voters neighbourhood distribution is completely different.

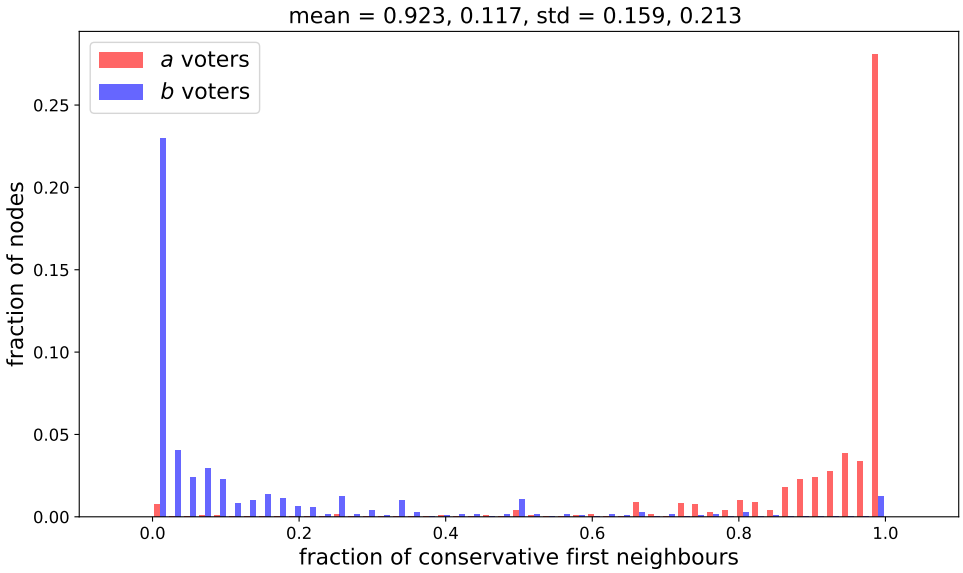


Figure 3.4: First neighbourhood vote distribution of the political blogs network. Blue bars represent liberal, and red bars conservative blogs. Horizontal axes represents the fraction of first neighbours who are conservative, and vertical a fraction of nodes with that fraction of conservative first neighbours. Mean and variance of the two distributions are denoted above the graph. Note the clear division between two types of nodes.

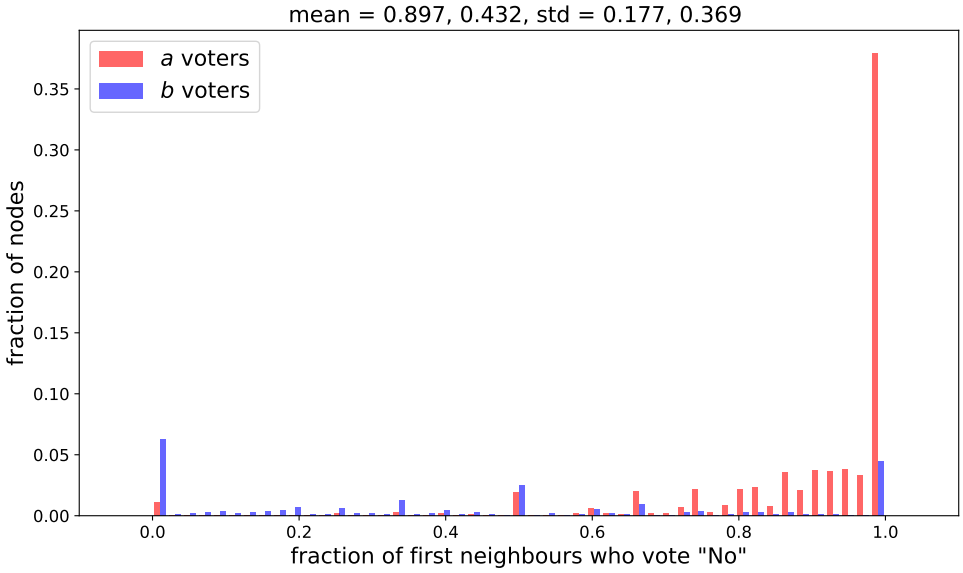


Figure 3.5: First neighbourhood vote distribution of the Croatian referendum network. Blue bars represent "Yes", and red bars "No" voters. Horizontal axes represents the fraction of first neighbours who vote "No", and vertical a fraction of nodes with that fraction of "No" voting first neighbours. Mean and variance of the two distributions are denoted above the graph. Note the similarity between the "No" voters' and political blogs neighbourhood distributions, whereas "Yes" voters' distribution is spread out.

3.2. Second neighbourhood

First neighbourhood distributions are problematic to visualize because some fractions naturally occur more often than others, as node degrees are relatively small (1, 0, 0.5 and such fractions are the most frequent ones). To avoid this problem and obtain smoother graphs, we will look at the second neighbourhoods distribution.

Political blogs

Figure 3.6 shows the second neighbourhood distribution of political blogs network. As expected, graph is similar in shape to one in figure 3.4, but is smoother and less skewed. Division between the two sides still exists, and it is still easy to predict one's vote from its neighbourhood.

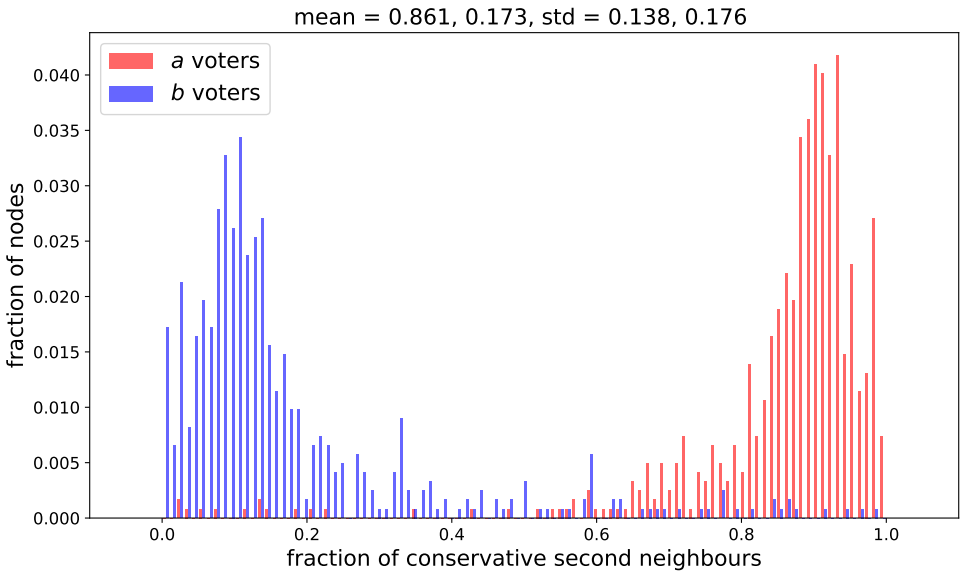


Figure 3.6: Second neighbourhood vote distribution of the political blogs network. Blue bars represent liberal, and red bars conservative blogs. Horizontal axes represents the fraction of second neighbours who are conservative, and vertical a fraction of nodes with that fraction of conservative second neighbours. Mean and variance of the two distributions are denoted above the graph. Note the similarity between this, and the previous, first neighbourhood distribution plot. Division is still clear, and graph is much smoother.

Croatian referendum

Figure 3.7 shows the second neighbourhood distribution of the Croatian referendum network. As with the political blogs, second neighbourhood distribution is similar in

shape with the first neighbourhood one, but is also smoother and less skewed. Uniform distribution of "Yes" voters is very interesting, and can be better understood if we look at the network itself in figure 3.2. Here we can see that, although clusterization exists, two sides of the network are much closer. Also, there is a significant amount of blue nodes which are deep in the red cluster.

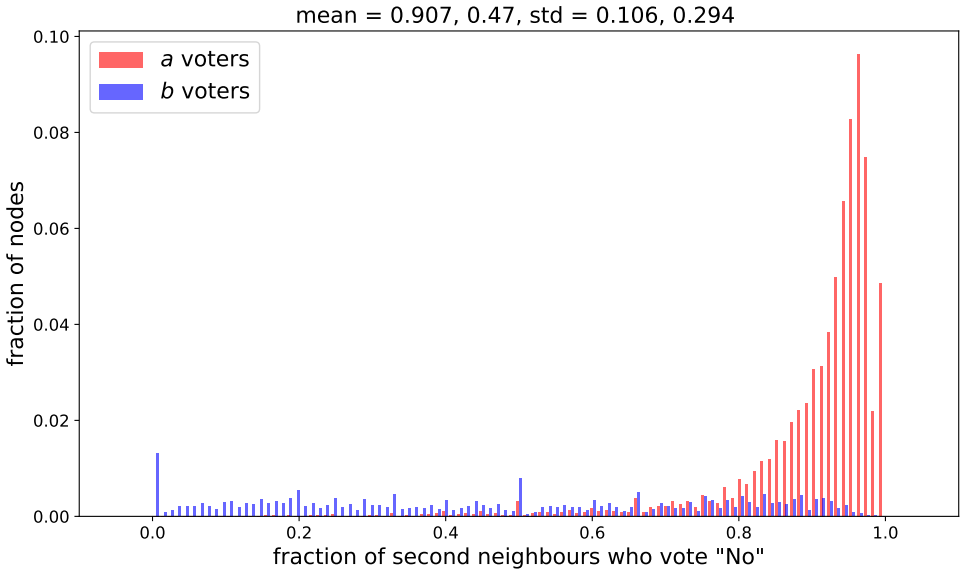


Figure 3.7: Second neighbourhood vote distribution of the Croatian referendum network. Blue bars represent "Yes", and red bars "No" voters. Horizontal axes represents the fraction of second neighbours who vote "No", and vertical a fraction of nodes with that fraction of "No" voting second neighbours. Mean and variance of the two distributions are denoted above the graph. Graph is almost identical to the first neighbourhood one, but much smoother, and the difference between "Yes" and "No" voters' distributions is much more clear.

4. Predicting one's opinion

One of the main goals of this thesis was to determine how predictable opinions and voting options are from the information about the social neighbourhood. In this chapter we will present the results of such predictive modelling, using two different approaches: neighbourhood majority vote and random forests.

4.1. Neighbourhood majority vote

In the last chapter we have seen how nodes with different voting options have significantly different first and second neighbourhoods. Motivated by this finding, we tried using values $q_k(i, a), k = 1, 2$ directly, by predicting the node's vote as a majority vote in the observed neighbourhood. In other words, all nodes left of the 0.5 mark in the last chapter's figures are classified as "blue", and all nodes right of that mark "red". Table 4.1 shows the results, and figure 4.1 shows the ROC curve.

As expected, prediction on political blogs network is much easier, as node clustering is much higher. Referendum network has a high share of falsely classified b nodes for both $k = 1$ and $k = 2$. This was also expected considering their neighbourhood distributions.

	Political blogs		Referendum	
True/Predicted	$k = 1$	$k = 2$	$k = 1$	$k = 2$
a/a	621	621	7453	7532
b/b	568	545	1943	1377
a/b	15	15	221	141
b/a	18	41	558	1124
Accuracy	0.9730	0.9542	0.9234	0.8757

Table 4.1: Neighbourhood majority based vote prediction results.

4.2. Random forest

Different approach to voting prediction is to use some of the more advance machine-learning algorithms, such as random forests. Random forest is a supervised ensemble learning method for both classification and regression. It is a combination of multiple decision trees which can be trained in parallel. To make a prediction, it uses either the majority vote of these trees (for classification), or their average (for regression). They are often used instead of decision trees to correct their overfitting.

We have used the random forest implementation provided by the *scikit-learn* Python library.

Inputs

Training data for the random forest is in the form of a list of data-points, each of which is represented by a feature vector. For the vote prediction we have used the following features: $deg(i)$, $|Nb_2(i)|$, $q_1(i, a)$ and $q_2(i, a)$. Best results were achieved with 50 or more decision trees. Training set consisted of 70% of data.

Results

As is shown in table 4.2 and figure 4.2, results are very similar to the neighbourhood majority ones. This indicates that there is no additional cross-feature information with significant predictive power.

True/Predicted	Political blogs	Referendum
a/a	189	2223
b/b	173	628
a/b	3	78
b/a	2	124
Accuracy	$0.97(\pm 0.02)$	$0.97(\pm 0.02)$

Table 4.2: Random forest voting prediction results. Accuracy score is now based on a 5-fold cross validation.

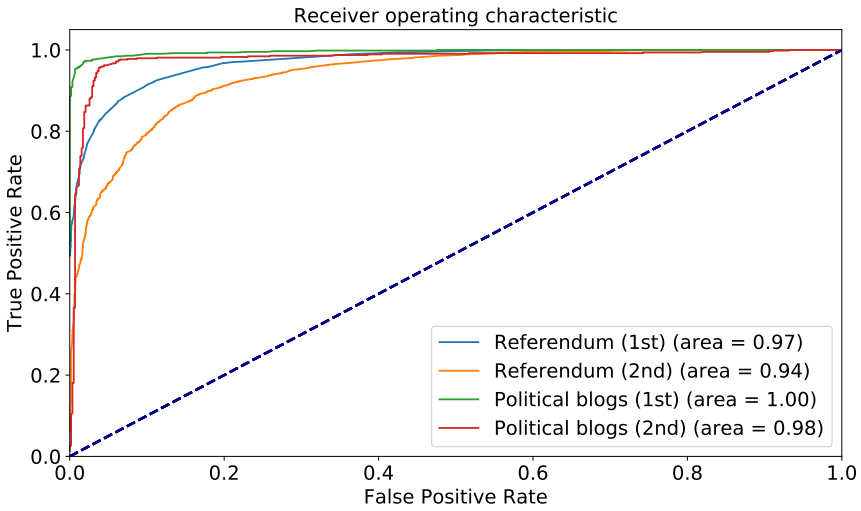


Figure 4.1: ROC curve of node vote prediction based on its first and second neighbourhood. We set the probability of a node to vote a to be the fraction of neighbours (first or second) who vote a. Note how the first neighbourhood is, in both networks, a better predictor.

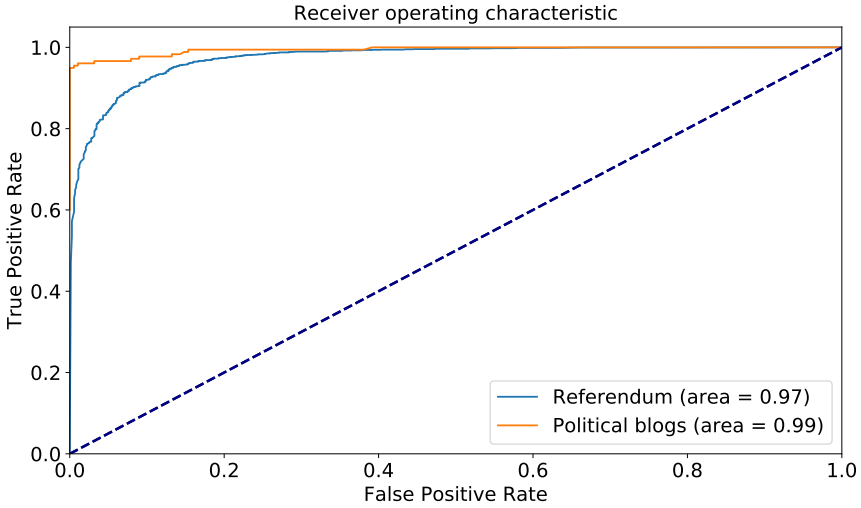


Figure 4.2: ROC curve of node vote prediction using random forest classification, with $deg(i)$, $|Nb_2(i)|$, $q_1(i,a)$ and $q_2(i,a)$ as features. Model performs well on both networks, but does not outperform the trivial approach of neighbourhood majority vote shown in figure 4.1.

5. Modelling social networks and vote distributions - naive approaches

Complete understanding of successful models is hard to achieve without first taking a look at some of the simpler but weaker ones. In this chapter we will start by explaining how Erdős–Rényi networks provided us with an interesting research direction for the problem of predicting election results, and then we will describe how we initially tried to solve it using the BA model.

5.1. Erdős–Rényi model

Erdős–Rényi (ER) model (Erdös and Rényi (1959)) is the oldest and the most well researched random network model, used primarily in physics in the field of percolation theory. It is often used as the baseline model of some more advance modelling techniques because of its simplicity.

There are two variants of the model:

- $G(n, p)$ variant creates a network with n nodes, and an edge between each pair of nodes with probability p,
- $G(n, m)$ variant creates a network with n nodes and m randomly placed edges.

Because edge placement is completely random, degree distributions of ER networks are binomial (see figure 5.2).

Apart from network modelling, we need to model how votes are distributed trough that network. As ER networks are completely different from the social networks we are interested in, there is no point in trying to develop a realistic voting model on them. For this reason, we will use a completely random voting – with some probability p_a node votes a, otherwise b, so we expect $Q(a) \approx p_a$. A small example of such network is shown in figure 5.1. Figure 5.3 shows the first and second neighbourhood distributions of a larger ER network ($G(5000, 0.01)$, $p_a = 0.7$), and figure 5.2 its degree distribution.

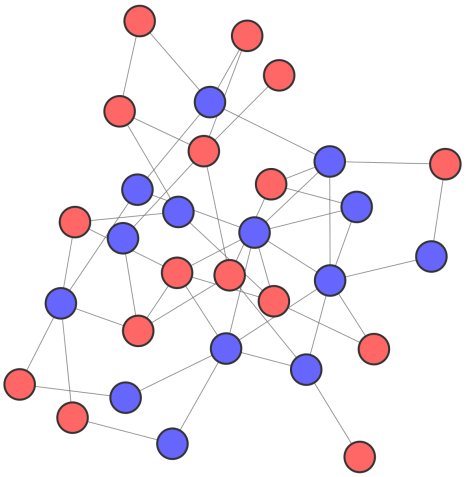


Figure 5.1: Erdős–Rényi social network modelling example. Network created with the $G(30, 50), p_a = 0.5$ model.

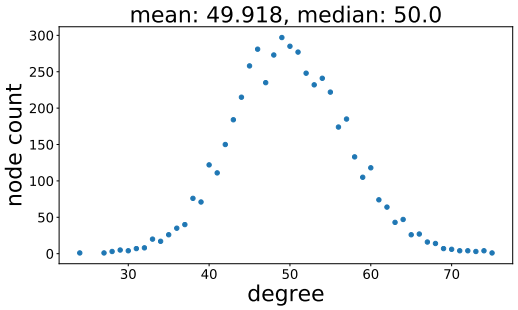


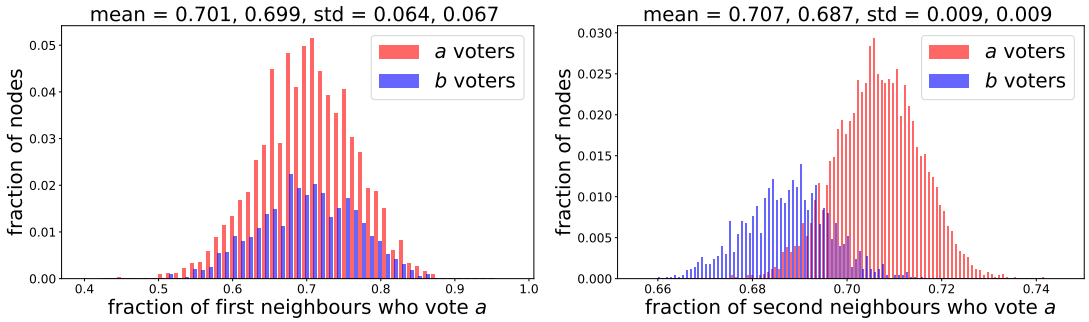
Figure 5.2: Erdős–Rényi network degree distribution. Network created with the $G(5000, 0.01)$ model.

Note how the first neighbourhood vote distribution for both types of nodes is the same, while there is a clear difference between them in the second neighbourhood distribution. This is due to our non-standard definition of second neighbourhood, which includes the central node itself, so nodes who vote a have, on average, one more a voting second neighbour (themselves) than nodes who vote b.

Further, note how both first and second neighbourhood distribution average is equal to $Q(a)$, due to a complete voting randomness. Finally, note how the standard deviation of neighbourhood distributions is an order of magnitude smaller than the standard deviation of votes themselves, which is equal to $p_a(1 - p_a)$, because voting can be viewed as a Bernoulli variable. For instance, standard deviation of voting in our example network where $p_a = 0.7$ is 0.21, while first and second neighbourhood vote distribution standard deviation is 0.066 and 0.009 respectively.

Two cluster Erdős–Rényi model

To overcome the fact that ER model generates homogeneous networks, we might do the following: instead of connecting each pair of nodes with probability p, we could group nodes into two clusters of sizes N_a and N_b, and then connect pairs of nodes in the same group with probability p_{in}, and pairs of nodes in different groups with probability p_{out}. By setting $p_{in} > p_{out}$, two clusters would emerge. Although we have successfully created the network macro structure similar to the one found in real-world networks, on a more local scale these networks still behave as ordinary ER ones – mainly, the

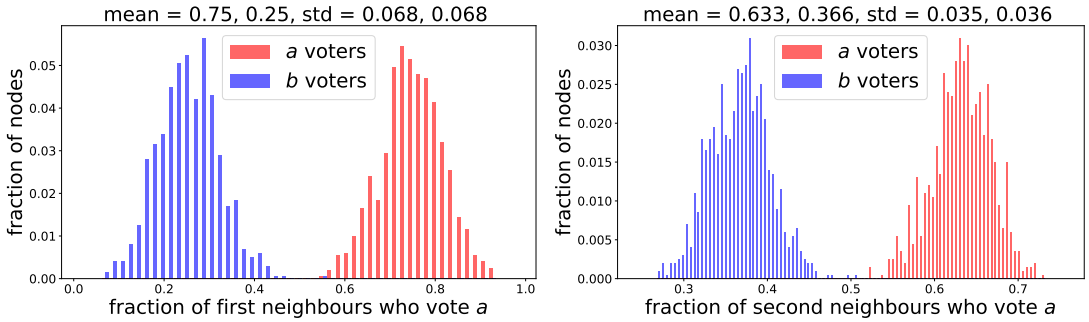


(a) First neighbourhood vote distribution. **(b)** Second neighbourhood vote distribution.

Figure 5.3: First and second neighbourhood vote distribution of ER network with randomly distributed votes. Network created with $G(5000, 0.01)$ and $p_a = 0.7$. Note how the second neighbourhood distributions differ, although voting was completely random – this is due to the second neighbourhood definition we use, which also includes the central node itself.

neighbourhood vote distribution is still bell shaped for both votes, as shown in figure 5.4.

The goal of this model was not to explain the processes we've seen in real networks, but to show how observed distributions were, in some sense, extreme in comparison with this fully random model.



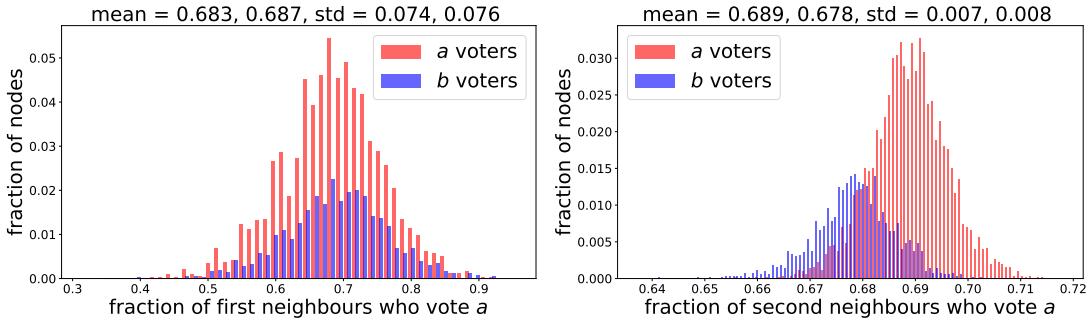
(a) First neighbourhood vote distribution. **(b)** Second neighbourhood vote distribution.

Figure 5.4: First and second neighbourhood vote distribution of a two cluster ER network – network with two sets of nodes (a and b voters), and different edge probabilities for same (p_{in}) and differently (p_{out}) voting nodes. Network created with parameters $N_a = N_b = 1000, p_{in} = 0.03, p_{out} = 0.01$. Note how, although there is a clear split between a and b neighbourhood distributions, both of them are still bell shaped.

5.2. Barabási–Albert model

Barabási–Albert model is a model controlled by two parameters: number of nodes n, and the initial degree of each node m. Model first generates a connected network of m nodes. After that, nodes are added one by one, each with m edges. Each of these edges is attached to one of the pre-existing nodes, chosen with probabilities proportional to their degree.

BA model produces scale-free networks, as we have already discussed in section 2.3.1, which have a few extremely well connected nodes (called *hubs*), and a large number of low-degree nodes. Because of this, when voting is random as it was in previous chapter, mean of the neighbourhood distribution varies more significantly around $Q(a)$, depending on the votes of those few *hubs* (when calculating the mean of the neighbourhood distribution, each node contributes proportionally to its degree). Example is given in figure 5.5 – note the difference between the mean of neighbourhood vote distribution and $Q(a) = 0.7$. When voting is not random this mean can vary even more, depending on the voting process.



(a) First neighbourhood vote distribution. **(b)** Second neighbourhood vote distribution.

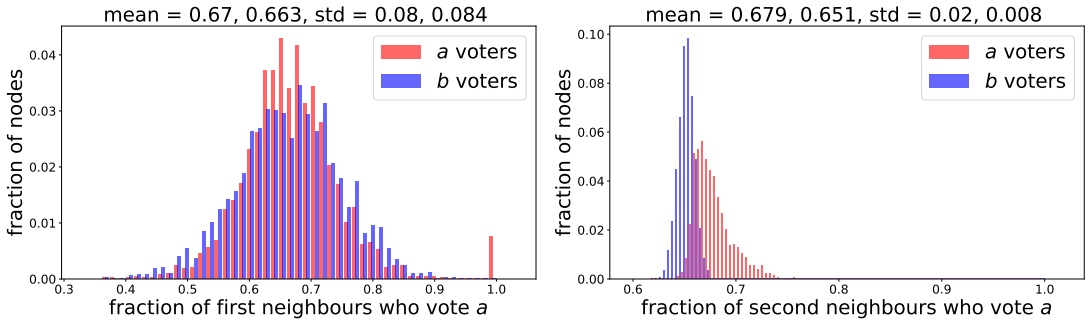
Figure 5.5: First and second neighbourhood vote distribution of BA network with random voting, with p_a of nodes voting a. Network created with parameters $n = 5000, m = 25$ and $p_a = 0.7$. Note the same effect on the second neighbourhood distribution due to our definition of the second neighbourhood – there is a difference between a and b voters although the voting is random.

Main problem with modelling real-world networks with polarizing voting using the BA model is that it cannot produce macro network topologies we require, mainly the two cluster split. To overcome this, we tried modelling the voting process by assigning vote a to nodes which are at maximum distance d of some node, and vote b to the rest. By changing parameter d, we can change the ratio between the number of a and b votes. This way, a votes would be densely clustered in one part of the network. As one

could expect, this did not produce realistic results, because votes b were to disperse. Neighbourhood distribution of an example of this process is shown in 5.6.

Next, we tried a similar, but symmetrical process: pick two *hubs* as a and b vote sources, and then assign votes to nodes based on which source they are closer to. By changing the weight of distance to each of the sources, we can control the ratio of votes. Neighbourhood distribution of an example of this process is shown in 5.7.

Apart from neighbourhood vote distributions being very different from the real-world ones, there were other major problems with these models. Most notable of these problems was the one illustrated in more detail in table 5.1 – if we group nodes based on their votes, and look at the probability of an edge existing between two nodes of the same group, and two nodes of different groups, we see that the real clusterization did not take place. In simple terms, nodes who vote differently are too well connected.

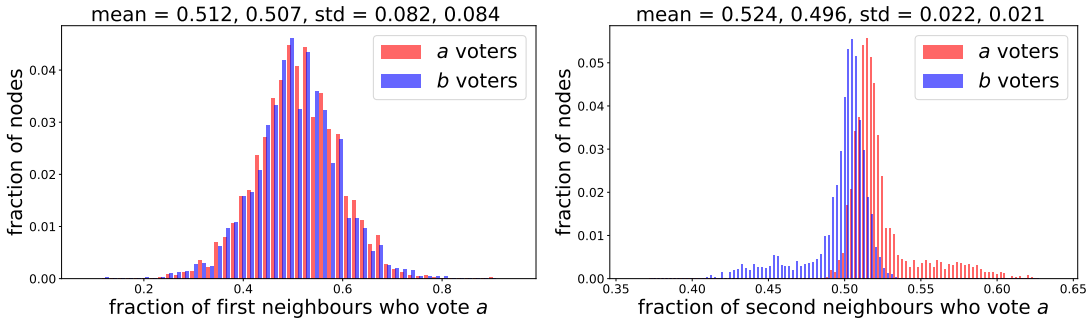


(a) First neighbourhood vote distribution. **(b)** Second neighbourhood vote distribution.

Figure 5.6: First and second neighbourhood vote distribution of BA network with one source vote assignment – a single hub is chosen randomly, and then p_a of nodes closest to it are assigned to vote a, while others vote b. Network created with parameters $n = 5000, m = 25$ and $p_a = 0.5$.

Edge probability	Real world		Barabási–Albert	
between nodes	Referendum	Political blogs	One source	Two source
p_{aa}	0.0017	0.0388	0.0190	0.0102
p_{bb}	0.0019	0.0426	0.0039	0.0098
p_{ab}	0.0004	0.0042	0.0085	0.0099

Table 5.1: Edge probability between same and different voting nodes. Edge probability is calculated as the ratio of the number of edges with given endpoints ($a - a$, $b - b$ or $a - b$), and the total possible number of such edges. Note that in real-world networks nodes who vote differently are significantly less connected, while it is not the case with BA networks.



(a) First neighbourhood vote distribution. **(b)** Second neighbourhood vote distribution.

Figure 5.7: First and second neighbourhood vote distribution of BA network with two source vote assignment – two *hubs* are chosen randomly as a and b vote sources, and nodes are assigned a vote based on which source they are closer to. Nodes equally distant from both hubs vote randomly. Note the similarity of the first neighbourhood between all three types of voting in figures 5.5, 5.6, and here. Also note the long-tail second neighbourhood distribution of both a and b, but their opposite orientation compared to the political blogs network in figure 3.6.

Two cluster Barabási–Albert model

Two cluster ER model produced a clear split in the neighbourhood vote distribution, while BA model produced, at least partially, skewed distributions closer to the real-world ones. Because of this, one could be led to expect that the combination of these two models would produce realistic networks. There are two possible ways to achieve this.

First is to generate a BA network, and then, randomly, connect some previously disconnected nodes with the same vote, and disconnect some of the previously connected nodes with different votes. This would result in an ER network *superimposed* onto the BA one, because the rewiring would not consider the initial BA edge probabilities. This results in networks which, although their clusters are sufficiently disconnected, have neighbourhood vote distributions too similar to ER ones. Realistically looking distributions in this model occurs only after p_{in} is a few orders of magnitude smaller than p_{out}, which is not the case with real-world networks.

Second possible approach is to randomly choose the vote of the new node when it is created (while generating the network), and then combine the preferential attachment process with the homophily principle to select its neighbours – preferring nodes with the same vote and a higher degree. This approach is, on the other hand, too complicated to formally analyse. Because of this, we turn to the Kronecker graph model, explained in the following chapter, which both formalizes easily, and produces realistic results.

6. Kronecker graph model

Kronecker graph model is a non-random graph model for generating various types of graphs Leskovec et al. (2008). In this chapter we will first explain the main idea and the mechanics of the Kronecker graph model. Next we will explain the stochastic version of this model, and how it can be adapted to model social networks. Finally, we will analyse the neighbourhood vote distributions of networks generated by this model.

6.1. Main idea

Real-world networks can be viewed on many different scales – from local, node oriented analysis which deals with degree distributions, neighbourhood distributions and similar information, up to macro structures like communities and clusters. And in many cases, these macro structures self-repeat on local levels, giving rise to a form of self-similarity often found in nature Song et al. (2005). We already intuitively understand this – societies are clustered based on political, ideological, socioeconomical, geographical and other factors, and these clusters are similarly interconnected on multiple scales. From continents to streets and buildings, from major social and economic systems to minor policy disputes, and from extreme income inequalities to local disparities.

Main idea behind Kronecker graphs is that, by defining either the global or the local structure of the graph, we have defined it in its entirety. In other words, to model large networks, we only need to understand either their smallest components, or their overall topologies.

6.2. Kronecker product and power

To explain how the Kronecker graphs are generated, we must first define the Kronecker product and Kronecker power. Let $\mathbf{A} = [a_{i,j}]$ and $\mathbf{B}$ be matrices of sizes $n \times m$ and $n' \times m'$ respectively. Then, Kronecker product $\mathbf{K} = \mathbf{A} \otimes \mathbf{B}$ is a $(n \cdot n') \times (m \cdot m')$

matrix given by:

$$K = A \otimes B = \begin{bmatrix} a_{1,1}\mathbf{B} & a_{1,2}\mathbf{B} & \dots & a_{1,m}\mathbf{B} \\ a_{2,1}\mathbf{B} & a_{2,2}\mathbf{B} & \dots & a_{2,m}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1}\mathbf{B} & a_{n,2}\mathbf{B} & \dots & a_{n,m}\mathbf{B} \end{bmatrix} \tag{6.1}$$

We can now define the Kronecker power as a consecutive application of Kronecker product:

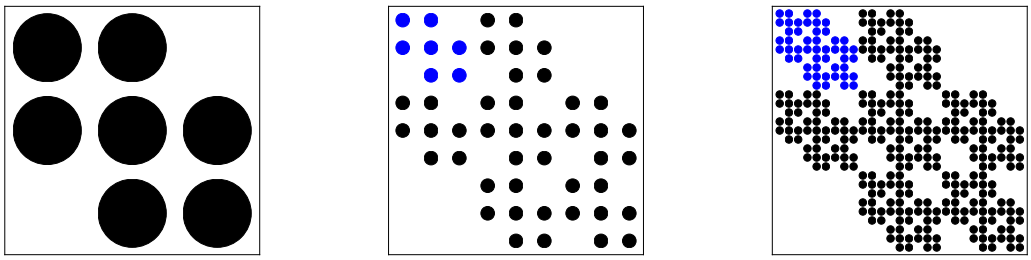
$$\mathbf{K}_1^{[k]} = \mathbf{K}_k = \underbrace{\mathbf{K}_1 \otimes \mathbf{K}_1 \otimes \dots \mathbf{K}_1}_{k \text{ times}} = \mathbf{K}_{k-1} \otimes \mathbf{K}_1 \tag{6.2}$$

Lastly, we need to define the Kronecker product of two graphs. Let G and H be graphs with adjacency matrices A_G and A_H. Then the Kronecker product of graphs $G \otimes H$ is a graph defined with the adjacency matrix $A_{G \otimes H} = A_G \otimes A_H$.

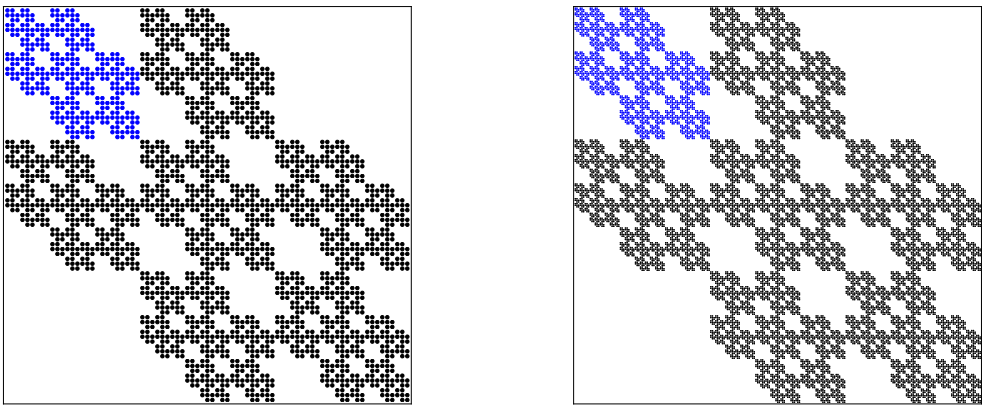
6.3. Kronecker graphs

Kronecker graphs are non-randomly generated graphs, produced by the k-th Kronecker power of $n \times n$ initiator adjacency matrix K_1. These initiator matrices are usually small, with $n < 10$. Example of this process is given in figure 6.1.

Kronecker graphs display a wide range of properties found in real-world networks and described in section 2.3 – degree distributions which are a combination of power law and bell shaped curves, densification power law, diameter reduction, etc. For a more in-depth exploration of Kronecker graphs, see Leskovec et al. (2008). There is, however, one important drawback – they are not random. To overcome this, Leskovec et al. (2008) introduce stochastic Kronecker graphs.



(a) K_1 **(b)** $K_2 = K_1 \otimes K_1$ **(c)** $K_3 = K_2 \otimes K_1$



(d) $K_4 = K_3 \otimes K_1$ **(e)** $K_5 = K_4 \otimes K_1$

Figure 6.1: Examples of adjacency matrices of Kronecker graphs $K_k = K_{k-1} \otimes K_1$ for $k = 1 \ldots 5$. Initiator graph is a triangle with self loops on all nodes.

6.4. Stochastic Kronecker graphs

Stochastic Kronecker graphs are a generalization of Kronecker graphs when matrix K_1 is a stochastic matrix – each element represents a probability of an edge existing between the two corresponding nodes. Once the matrix K_k is calculated, each pair of nodes (i, j) is connected with probability $K_k[i, j]$. [1]

Without a way to efficiently generate networks which are similar to some given real-world ones, random network model is not useful. Trivial way of finding the initiator matrix K_1 which would generate networks similar to some given G with n^k nodes

[1] Leskovec et al. (2008) present a much faster algorithm for the graph generation, but it is out of scope of this thesis, as we are interested in stochastic Kronecker graphs from the theoretical analysis standpoint.

would be as follows. Let σ be some permutation of the nodes of G, i.e. a function which assigns some unique node j to each node i, such that σ_i is the i-th node of the permutation. Let $P(G|K_1, \sigma)$ be a probability of graph G being created with the initiator probability matrix K_1, and permutation σ:

$$P(G|K_1, \sigma) = \prod_{(u,v)\in G} K_k[\sigma_u, \sigma_v] \prod_{(u,v)\notin G} (1 - K_k[\sigma_u, \sigma_v]) \tag{6.3}$$

Next, we define the *log-likelihood* of some initiator matrix generating the graph G:

$$l(K_1|G) = \log P(G|K_1) = \log \sum_{\sigma} P(G|K_1, \sigma)P(\sigma|K_1) \tag{6.4}$$

In general, $P(\sigma|K_k)$ is constant, so equation 6.4 can be simplified to:

$$l(K_1|G) = \log \sum_{\sigma} P(G|K_1, \sigma) \tag{6.5}$$

To find the best matrix K_1 for some graph G in a naive manner we could perform some sort of gradient descent using $l(K_1|G)$. This, however, is not feasible, as complexity of calculating the *log-likelihood* is $O(N!N^2)$, where N is the number of nodes in G (Leskovec et al. (2008)).

Fortunately, by sampling the permutations σ using the Metropolis sampling (Gamerman and Lopes (2006)), Leskovec et al. (2008) show how this problem can be solved in linear time.

To model real-world networks described in this thesis, we have used the implementation of this algorithm provided in Leskovec and Sosič (2016).

6.5. Modelling polarized social networks

Although stochastic Kronecker graphs produce realistic networks, it is not easy to generate a macro two cluster split. If we specified this split in matrix K_1, then each of those clusters would again be split into two subclusters, and so on, and this may not reflect our network exactly. To solve this, we propose a generalized stochastic Kronecker graph model with the additional $n_0 \times n_0$ matrix K_0. To generate the network using K_0, instead of using the K_k as our edge probability matrix, we use $K'_k = K_0 \otimes K_{k-1}$. This way, by changing K_0, we can define both the ratio of votes (cluster sizes), and the edge probabilities p_{in} and p_{out}, as follows.

Let N_a and N_b be the number of votes a and b respectively, with $N_a + N_b = N = n_0 \cdot n^{k-1}$, where n is the size of matrix K_1. Let $Q(a)$ and $Q(b)$ be fractions of nodes

who vote a and b respectively ($Q(a)/Q(b) = N_a/N_b, Q(a) + Q(b) = 1$). Let p'_{in} and p'_{out} be some initial probabilities[2] of two nodes being connected (with same and different votes respectively).

We define the matrix K_0 as a square matrix of size n_0 with rows and columns divided into two groups which will later become the two clusters. We denote the sizes of these groups with n_{0a} and n_{0b}, so that $n_{0a} + n_{0b} = n_0$ and $n_{0a}/n_{0b} = Q(a)/Q(b)$.

This way matrix is naturally divided into two square submatrices – edges inside clusters, and two rectangle submatrices – edges connecting the clusters, as shown in figure 6.2. We then set the elements of the matrix to p'_{in} and $p'out$.

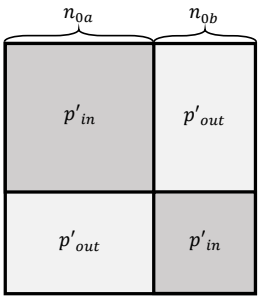


Figure 6.2: Diagram of a typical K_0 matrix of size $n_0 = n_{0a} + n_{0b}$. Elements (i, j) where $i, j < n_{0a}$ or $i, j >= n_{0a}$ are filled with the initial in-cluster edge probability p'_{in}, and the rest is filled with the initial edge probability between two clusters p'_{out}. Usually, $p'_{in} > p'_{out}$.

It is evident now that the resulting graph, generated by sampling edges from the probability matrix K'_k, will have two clusters of sizes N_a and N_b. Further, if we represent the matrix K'_k as we represented the Kronecker product matrix in equation 6.1, we see that the edge probabilities p_{in} and p_{out} would be such that $p_{in}/p_{out} = p'_{in}/p'_{out}$ holds. This, for $n_{0a} = 2$ and $n_{0b} = 1$, is shown in the following equation:

$$K'_k = K_0 \otimes K_{k-1} = \begin{bmatrix} p'_{in}K_{k-1} & p'_{in}K_{k-1} & p'_{out}K_{k-1} \\ p'_{in}K_{k-1} & p'_{in}K_{k-1} & p'_{out}K_{k-1} \\ p'_{out}K_{k-1} & p'_{out}K_{k-1} & p'_{in}K_{k-1} \end{bmatrix} \tag{6.6}$$

In other words, $p_{in} = C \cdot p'_{in}$ and $p_{out} = C \cdot p'_{out}$, where C is the value which depends on the average matrix K_{k-1}. This means that we can control parameters p_{in} and p_{out}, solving the problem of unrealistic edge probabilities encountered in BA networks in section 5.2.

[2]Note that these probabilities are not equal to p_{in} and p_{out} of the final graph.

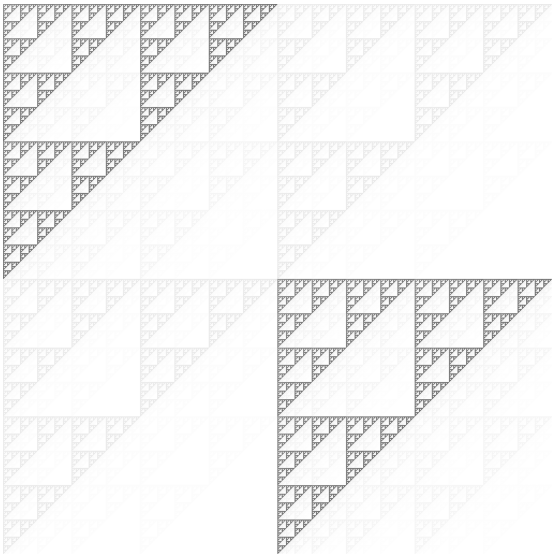


Figure 6.3: Example of the polarized stochastic Kronecker network edge probability matrix. Matrix generated from K_0 defined in equation 6.7, with $p'_{in} = 0.8$, $p'_{out} = 0.2$ and $k = 11$. Shade of grey represents the probability of an edge between two nodes. Note how the two clusters clearly emerge, and how nodes are similarly, but with different overall probabilities, connected to other nodes inside and outside of their cluster.

6.6. Neighbourhood vote distribution

We will now go through all of the model parameters, explaining how they affect the neighbourhood vote distributions. All of the following figures show the neighbourhood vote distributions of Kronecker networks generated using the same K_1 matrix, which was derived from the referendum network by using the algorithm mentioned above:

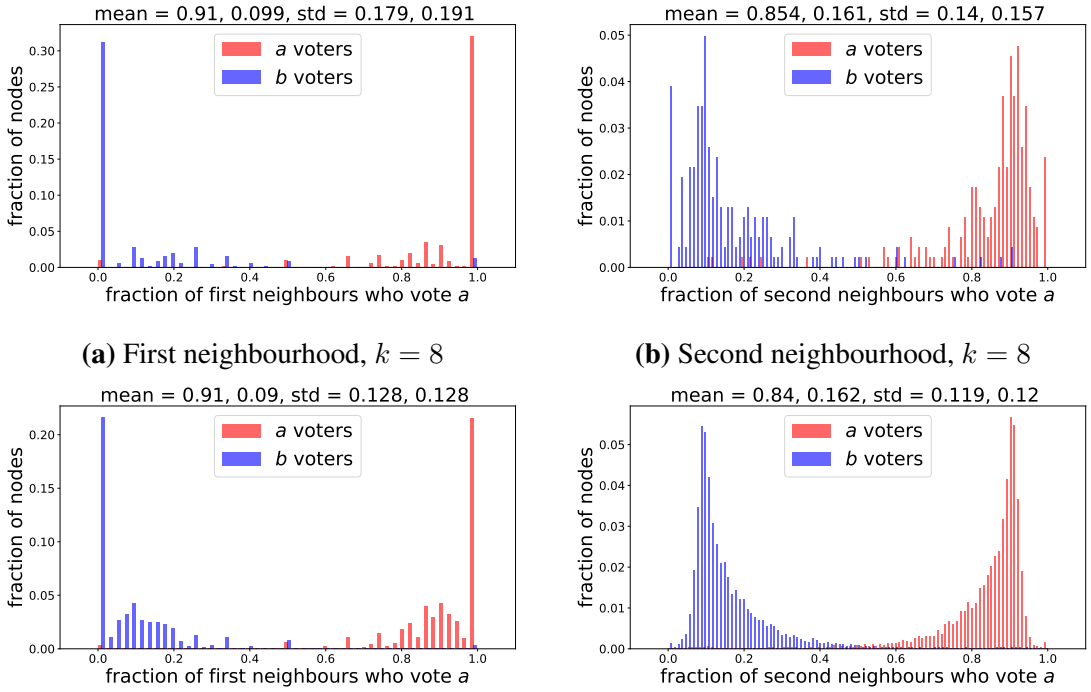
$$K_1 = \begin{bmatrix} 0.7701 & 0.7943 \\ 0.7943 & 0.0965 \end{bmatrix} \tag{6.7}$$

6.6.1. Parameter k

Parameter k is the main size-controlling parameter, besides the sizes of the initiator matrices. Apart from the size itself, because Kronecker model is designed to produce similar graphs of all scales, nothing else about the distributions changes, as we see in figure 6.4.

6.6.2. Parameters p'_{in} and p'_{out}

Unlike parameter k, parameters p'_{in} and p'_{out} have a direct impact on the neighbourhood vote distributions. Because of the quantity of graphs in the following figure, figure



(a) First neighbourhood, $k = 8$

(b) Second neighbourhood, $k = 8$

(c) First neighbourhood, $k = 13$

(d) Second neighbourhood, $k = 13$

Figure 6.4: Effect of parameter k on polarized Kronecker networks neighbourhood vote distributions. For both networks, $p'_{in} = 0.9$, $p'_{out} = 0.09$ and $Q(a) = 0.5$. It is evident, as we expected, that there is no effect of the size of the network on the neighbourhood distributions.

6.5 shows only the second neighbourhood distributions, but first neighbourhood ones behave in the same manner. For all networks $k = 12$.

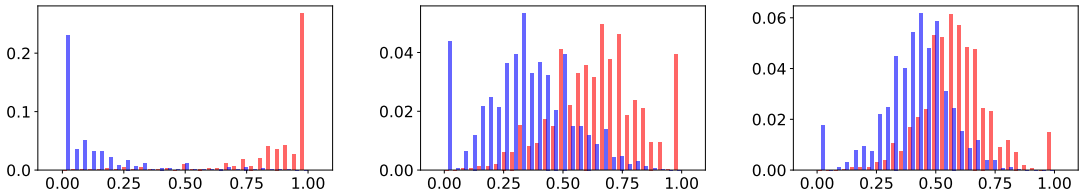
It is evident from figure 6.5 that neighbourhood vote distributions of a and b voters are closer as p'_{out} approaches p'_{in}. Note, however, that even when $p'_{in} = p'_{out}$, second neighbourhood distributions do not fully overlap. This is because our definition of second neighbourhood includes the central node itself. First neighbourhood vote distribution for such case would be the same for both groups of voters.

6.6.3. Parameters n_{0a} and n_{0b}

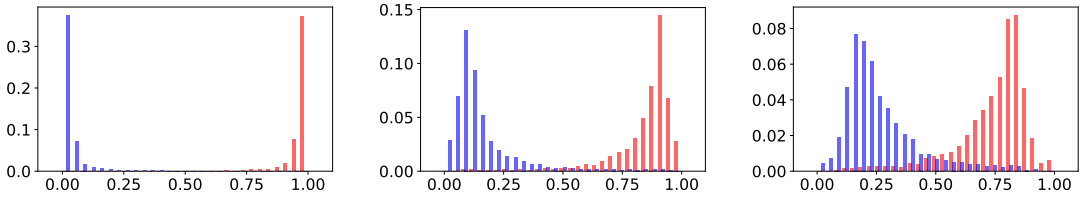
Parameters n_{0a} and n_{0b} control the ratio of a and b votes in the network. In figure 6.6 we fix the parameter $n_{0b} = 2$, and change the parameter n_{0a} from 2 to 5. This way we can observe what happens to the network as it becomes more and more unbalanced.

First thing to note when examining figure 6.6 is the speed at which the unbalance of the network effects the neighbourhood distribution – even at the ratio of 60%-40% distributions are completely different from the balanced network ones.

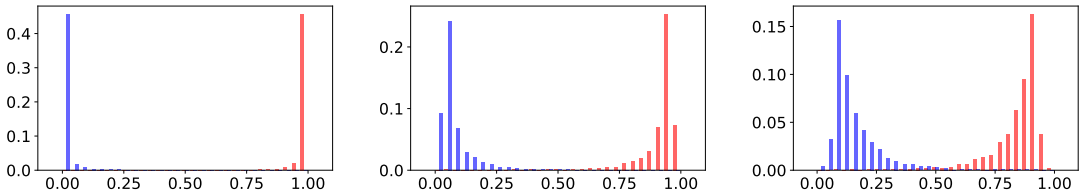
Second, note how the majority vote distribution holds its shape, while the minority spreads – something we have already observed in the referendum network.



(a) $p'_{in} = 0.1, p'_{out} = 0.01$ **(b)** $p'_{in} = 0.1, p'_{out} = 0.05$ **(c)** $p'_{in} = 0.1, p'_{out} = 0.1$

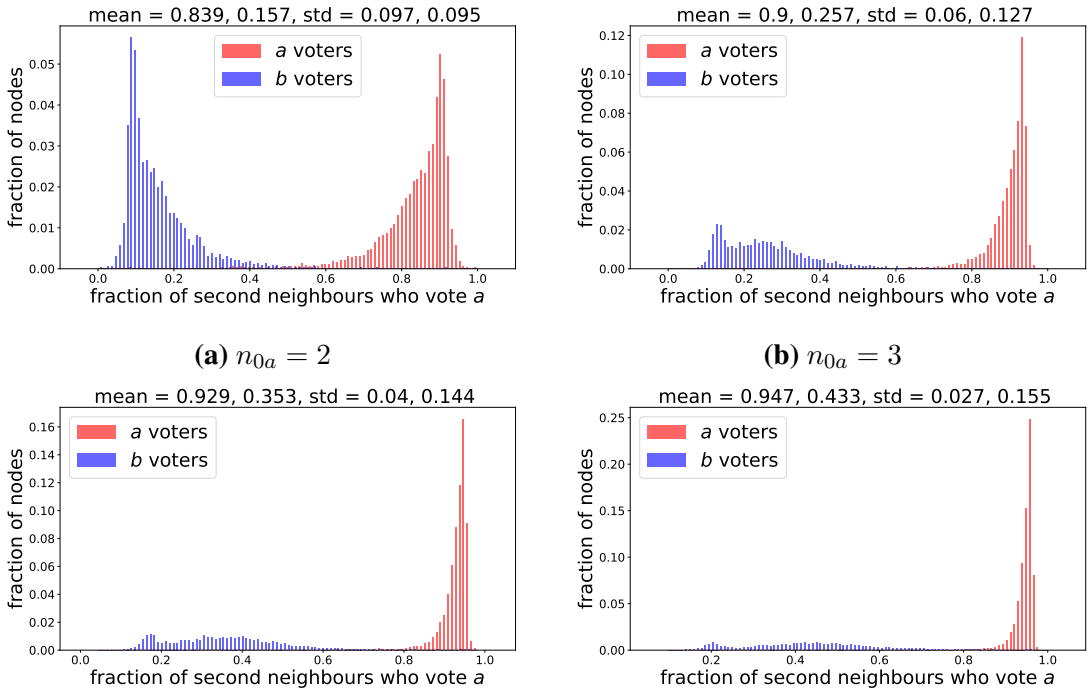


(d) $p'_{in} = 0.5, p'_{out} = 0.01$ **(e)** $p'_{in} = 0.5, p'_{out} = 0.05$ **(f)** $p'_{in} = 0.5, p'_{out} = 0.1$



(g) $p'_{in} = 1, p'_{out} = 0.01$ **(h)** $p'_{in} = 1, p'_{out} = 0.05$ **(i)** $p'_{in} = 1, p'_{out} = 0.1$

Figure 6.5: Effect of parameters p'_{in} and p'_{out} on polarized Kronecker networks second neighbourhood vote distributions. For all networks $k = 12$, $Q(a) = 0.5$. Horizontal axes represents the fraction of second neighbours who vote a, and vertical a fraction of nodes with that fraction of a voting second neighbours. Red bars represent a voters, while $blue$ bars represent b voters. Note how the network is more divided as p'_{in} increases and p'_{out} decreases, as is expected from the model.



(a) $n_{0a} = 2$ **(b)** $n_{0a} = 3$

(c) $n_{0a} = 4$ **(d)** $n_{0a} = 5$

Figure 6.6: Effect of parameters n_{0a} and n_{0b} on polarized Kronecker networks neighbourhood vote distributions. Because it is only the ratio of these parameters that matters, we fix $n_{0b} = 2$. For all networks, $k = 11$, $p'_{in} = 0.9$ and $p'_{out} = 0.09$. It is evident that the vote share ratio $Q(a)$ has a strong effect on the both neighbourhood distributions, most notably on the minority one.

7. Election results prediction on polarized Kronecker networks

Modelling polarized social networks may be useful in many applications, mainly to better understand them and predict their behaviour. But another important role of any modelling is the inference of the whole object from only partially available data – in this case, predicting the vote shares on social networks from a small subset of observed nodes. Everything we covered so far indicates that there is a strong relationship between the ratio of network's votes and neighbourhood distributions, and that these distributions might be much easier to sample than the votes themselves – their variance might be smaller, and they might be much more resistant to sampling bias, which is the main problem of any social prediction.

In this chapter, we will explore how the knowledge of neighbourhood distributions may be used in election predictions and how it performs on network simulations. We start by explaining the first estimator of parameter $Q(a)$, which we later expand and improve.

7.1. First neighbourhood based vote share estimator

Throughout this thesis we have viewed the neighbourhood vote distribution from the perspective of network itself. Now we will focus on it in the context of adjacency matrices.

Each element of the edge probability matrix is a Bernoulli random variable, denoting the probability of an edge existing between corresponding nodes. Once the edges are sampled from this matrix, sum of each row represents the degree of that node. Because this is the sum of n Bernoulli variables, it is, by definition, a Poisson binomial distribution (PBD), and its mean and variance are simply the sum of the means and variances of n Bernoulli variables:

$$\mu_{PBD} = \sum_i p_i \tag{7.1}$$

$$\sigma_{PBD} = \sum_i (1 - p_i)p_i \tag{7.2}$$

Further, because each of these probabilities is small, with $p_i << 0.1$, and because n is large, PBD may be approximated with the Poisson distribution, as stated in the Le Cam's theorem (Le Cam et al. (1960)), with parameter $\lambda = \mu_{PBD}$. This means that not only can we model the entire row sum with the Poisson distribution, but also each of its parts, if we know their mean (and if they are sufficiently large). This idea will prove to be very useful.

Let us focus now on a single row i of the edge probability matrix K'_k, which corresponds to some a voter. As is evident from equation 6.6, such row consists of n_{0a} repetitions of row vector $p'_{in}K_{k-1}[j]$ and n_{0b} repetitions of row vector $p'_{out}K_{k-1}[j]$, where j is the corresponding row of matrix K_{k-1}, which is equal to $i \mod n_{k-1}$[1], where n_{k-1} is the size of matrix K_{k-1}. Fraction of neighbours who vote a is the ratio of the number of edges connecting the node to a voters, and its degree. In other words, it is the ratio of the Poisson binomial variables X and Y, where X is, as its probability vector, using only the first N_a entries, and Y is using the whole row. Further, as we stated above, X and Y can be approximated with Poisson variables, with parameters λ_X and λ_Y.

Expectation of the ratio of two Poisson random variables, as stated in the Ogliore et al. (2011), is approximately:

$$\mathbb{E}[X/Y] \approx \frac{\lambda_X}{\lambda_Y}(1 + \frac{1}{\lambda_Y} + \frac{2}{\lambda_Y^2}) \tag{7.3}$$

for independent variables X and Y, where λ_X and λ_Y are the expected values of those variables.

Because our λ_Y, representing the node's degree, is sufficiently large (especially in real social networks, when degree represents the number of acquaintances), this can be simplified to:

$$\mathbb{E}[X/Y] \approx \frac{\lambda_X}{\lambda_Y} \tag{7.4}$$

As we are analysing the neighbourhood distributions separately for a and b voters, we will introduce the following four random variables:

[1] Presuming we use zero-indexing.

– X_{aa}, the fraction of a voting neighbours of a node who also votes a

– X_{bb}, the fraction of b voting neighbours of a node who also votes b

– X_{ab}, the fraction of b voting neighbours of a node who votes a

– X_{ba}, the fraction of a voting neighbours of a node who votes b

Combining the equation 7.1 and our insight into the shape of the single row of matrix K'_k from earlier, we can write the expected values of these variables as:

$$\begin{aligned} \mathbb{E}[X_{aa}] &= C \cdot Q(a) \cdot p'_{in} \\ \mathbb{E}[X_{bb}] &= C \cdot Q(b) \cdot p'_{in} \\ \mathbb{E}[X_{ab}] &= C \cdot Q(b) \cdot p'_{out} \\ \mathbb{E}[X_{ba}] &= C \cdot Q(a) \cdot p'_{out} \end{aligned} \tag{7.5}$$

where C is the value which depends on the average of row vectors $K_{k-1}[i]$, and the size of the matrix K_0. Now, using the equation 7.4 we can write the expected values of neighbourhood distributions as:

$$\begin{aligned} \mathbb{E}[q(i,a)|v(i)=a] = \mu_a = \mathbb{E}\left[\frac{X_{aa}}{X_{aa}+X_{ab}}\right] \approx \\ \approx \frac{\mathbb{E}[X_{aa}]}{\mathbb{E}[X_{aa}]+\mathbb{E}[X_{ab}]} = \frac{Q(a)\cdot p'_{in}}{Q(a)\cdot p'_{in}+Q(b)\cdot p'_{out}} \end{aligned} \tag{7.6}$$

$$\begin{aligned} \mathbb{E}[q(i,a)|v(i)=b] = \mu_b = \mathbb{E}\left[\frac{X_{ba}}{X_{ba}+X_{bb}}\right] \approx \\ \approx \frac{\mathbb{E}[X_{ba}]}{\mathbb{E}[X_{ba}]+\mathbb{E}[X_{bb}]} = \frac{Q(a)\cdot p'_{out}}{Q(a)\cdot p'_{out}+Q(b)\cdot p'_{in}} \end{aligned} \tag{7.7}$$

Values of μ_a and μ_a are collected empirically from online polls, and it is our goal to estimate the value of $Q(a)$, predicting the election results.

Using what we have defined so far, this is infeasible. But there is another constraint on these variables which we can introduce. We have stated earlier that p'_{in} and p'_{out} control the density of the network, but density is also controlled by the values of matrix K_0. For example, if instead of p'_{in} and p'_{out} we used only half of those values, we could compensate it by doubling the values of K_{k-1}[2]. This means that we can add a constraint $p'_{in} + p'_{out} = 1$ without the loss of generality. Note that we already have $Q(a) + Q(b) = 1$. Now, the above problem becomes feasible, and its solution is our first estimator of $Q(a)$:

[2]We control the scale of matrix K_{k-1} by controlling K_1.

$$Q(a) \approx \frac{\mu_a \mu_b \sqrt{\mu_a \mu_b - \mu_a^2 \mu_b - \mu_a \mu_b^2 + \mu_a^2 \mu_b^2}}{\mu_a + \mu_b - 1} \tag{7.8}$$

It is interesting to note that, when $\mu_a + \mu_b \approx 1$, this equation can be reduced to $Q(a) = (\mu_a + \mu_b)/2$, and in that case $Q(a) \approx 0.5$.

7.2. Second neighbourhood based vote share estimator

When examining second neighbourhoods in graphs, the usual approach is to look at the squared adjacency matrices A^2, where each element represents the number of two-hop paths between corresponding nodes. We, however, do not have access to adjacency matrices, as we are modelling random networks, but are instead working with edge probability matrices, where each element represent the expected number of edges between the two nodes (this is due to the expected value of the Bernoulli variable being equal to the probability of that variable being 1). When we square these matrices, each element is equal to:

$$K^2[i,j] = \sum_k K[i,k] \cdot K[k,j] \tag{7.9}$$

or, in terms of edge probabilities, it is the sum of the expected number of two-hop paths between nodes i and j, which travels through some node k, over all k. Because each two-hop path is independent of others (as each travels trough a distinct middle node) this is equal to the expected number of such two-hop paths between nodes i and j. It is for this reason that we have defined the second neighbourhood the way we did in section 2.1.

Let us now focus on these elements in the context of our polarized Kronecker networks. First, it is important to note that edge probability matrices of these networks are always symmetrical – everything that we have stated about the structure of their rows at the beginning of this chapter applies to the columns as well. Consider now some element $K^2[i,j]$ of the squared edge probability matrix, such that $i, j \leq N_a$ (so that this element corresponds to some edge between two nodes who vote a). Its value is the result of the dot product of the i-th and the j-th row (as rows are equal to columns) of matrix K. Both these vectors can be written as:

$$v = [p'_{in}x_1, p'_{in}x_2, \ldots p'_{in}x_{N_a}, p'_{out}x_{N_a+1}, \ldots p'_{out}x_{N_a+N_b}] \tag{7.10}$$

where vector x is equal to the corresponding vector $K_{k-1}[i]$ repeated $n_{0a} + n_{0b} = n_0$ times, as can be observed in equation 6.6. Their dot product is now equal to:

$$\begin{aligned} v \cdot v' = p'^2_{in} x_1 x'_1 + p'^2_{in} x_2 x'_2 + \ldots + p'^2_{in} x_{N_a} x'_{N_a} + \\ + p'^2_{out} x_{N_a+1} x'_{N_a+1} + \ldots + p'^2_{out} x_{N_a+N_b} x'_{N_a+N_b} \end{aligned} \tag{7.11}$$

This can be simplified to

$$v \cdot v' = p'^2_{in}(x_{1\ldots N_a} \cdot x'_{1\ldots N_a}) + p'^2_{out}(x_{N_a+1\ldots N_b} \cdot x'_{N_a+1\ldots N_b}) \tag{7.12}$$

Because vectors x and x' both consist of n_0 repeated subvectors of matrix K_{k-1}, and because $N_a/N_b = n_{0a}/n_{0b}$, we can simplify this term further:

$$v \cdot v' = p'^2_{in} n_{0a}(K_{k-1}[i] \cdot K_{k-1}[j]) + p'^2_{out} n_{0b}(K_{k-1}[i] \cdot K_{k-1}[j]) = \tag{7.13}$$

which is finally equal to:

$$v \cdot v' = (K_{k-1}[i] \cdot K_{k-1}[j])(p'^2_{in} n_{0a} + p'^2_{out} n_{0b}) \tag{7.14}$$

Same holds true for the other two types of element of the matrix K: edges between nodes who both vote b ($i, j > N_a$), and edges between differently voting nodes. The only thing that changes is the factor with which $(K_{k-1}[i] \cdot K_{k-1}[j])$ is multiplied. For edges between two b voters it is equal to $p'^2_{out} n_{0a} + p'^2_{in} n_{0b}$, and for edges between nodes who vote differently it is $p'_{in} p'_{out} n_0$. In all these factors, values of n_{0a} and n_{0b} can be replaced with $n_0 Q(a)$ and $n_0 Q(b)$ respectively. These three factors can now be inserted into equation 7.5 in place of p'_{in} and p'_{out}, and the analysis can be done in similar a manner. Finally, the same formula for $Q(a)$ emerges:

$$Q(a) \approx \frac{\mu_a \mu_b \sqrt{\mu_a \mu_b - \mu_a^2 \mu_b - \mu_a \mu_b^2 + \mu_a^2 \mu_b^2}}{\mu_a + \mu_b - 1} \tag{7.15}$$

where μ_a and μ_b are now means of second neighbourhood vote distributions. This result is very important since the distinction between first and second neighbours in a society is not that clear.

7.3. Results when simulating biased sampling

One of the main problems with polling data is the high bias of the population sample. In this section we will show how the relationship between sample size and prediction performance varies over different networks and different sampling biases, to confirm that our approach is less affected by this bias.

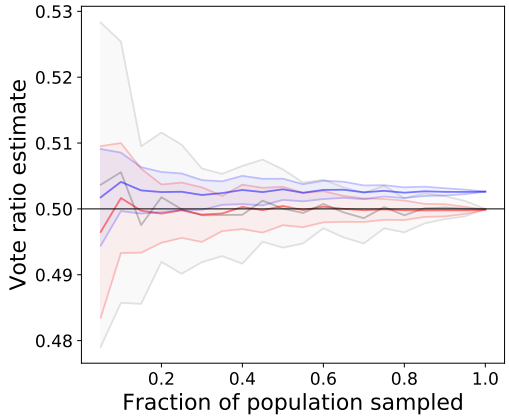
To simulate the sampling bias for some sample size s, we repeatedly, without replacement, select node i from the population with probability $p(i)$, using a form of *roulette-wheel* selection, until we sampled s nodes. As we aim to replicate the biased sampling where nodes who vote a have a higher probability of being sampled, we set $p(i) = \beta \cdot B(i) + (1 - \beta)$, where β is the bias strength controlling parameter such that $0 < \beta < 1$, and $B(i)$ is the sampling *weight* of a node, with $B(i) = 1$ if node votes a, otherwise $B(i) = 0$. In other words, we are introducing sampling bias solely based on the voting, and the ratio of probabilities of sampling nodes who vote a and nodes who vote b is $1/(1 - \beta)$. Note that if $\beta = 1$, nodes who vote b would never be sampled. For this reason, we avoid $\beta > 0.99$. There are also other possible features on which $B(i)$ can be based, such as the first or second neighbourhood vote shares, various topological features, etc.

Each of the following plots shows the prediction's mean and variance over 25 different sample sizes – one for each 4% of the total node count. For each sample size, we have run 50 simulations, to have a good approximation of the mean and variance. To better understand how our measure performs, we have plotted it against the results of a trivial $Q(a)$ estimator usually used in polls – the average of all the observed votes.

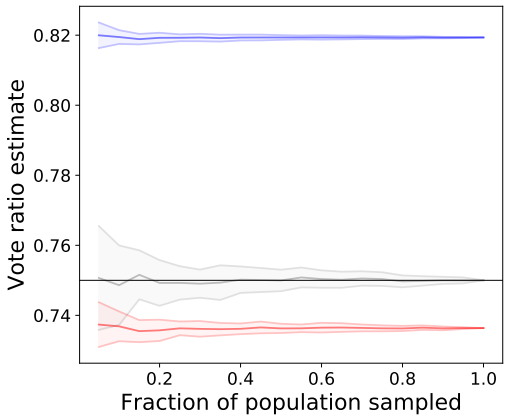
First interesting observation from the results is that our estimators are not at all affected by the sampling bias. This means that the only difference the sample size makes is the variance reduction, and even this is not that significant.

Second is the fact that, when vote share is balanced ($Q(a) \approx 0.5$) and there is a any bias in the sampling, our approach produces significantly better results than the trivial method. This becomes even more evident when bias is strong. On the other hand, when vote share is not balanced, both the first and second neighbourhood based estimators become biased, and, compared to the trivial method, do not perform well. In the next section we will explore how this bias can be corrected.

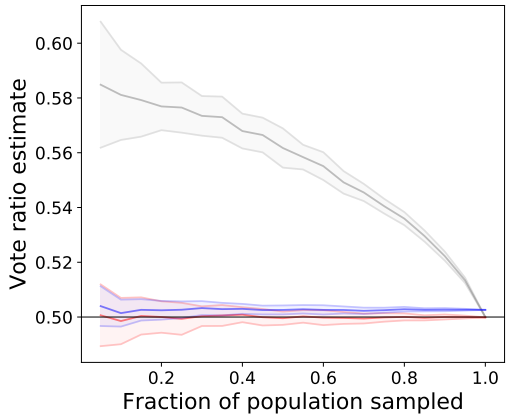
Lastly, it is important to note the difference between the estimators using first and second neighbourhood vote distributions. It seems from these results that these two estimators always bound the real result from both sides, with first neighbourhood based estimator always being biased towards more unbalanced, and second neighbourhood based one towards more balanced networks.



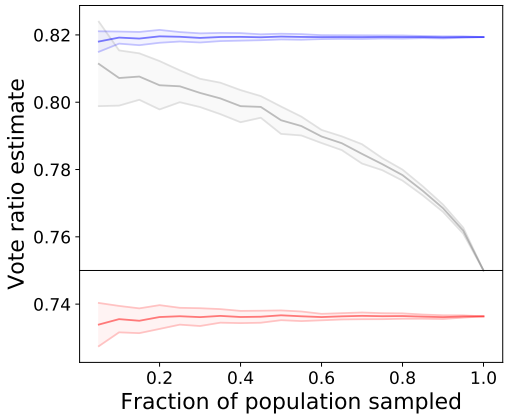
(a) $Q(a) = 50\%, \beta = 0$



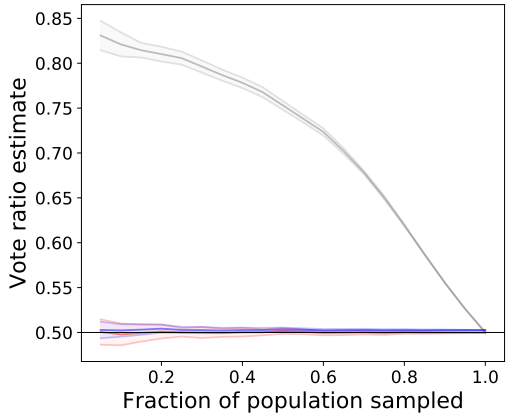
(b) $Q(a) = 75\%, \beta = 0$



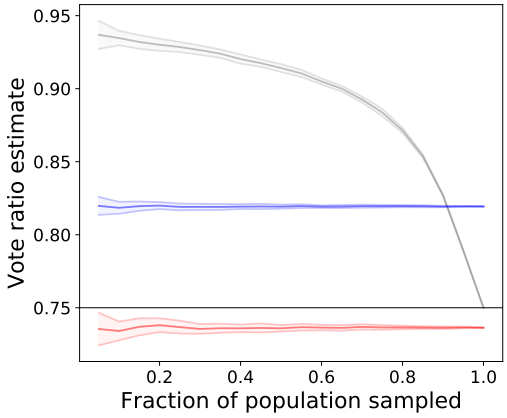
(c) $Q(a) = 50\%, \beta = 0.3$



(d) $Q(a) = 75\%, \beta = 0.3$

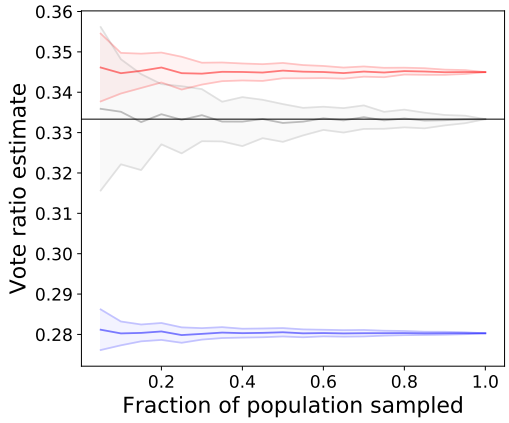


(e) $Q(a) = 50\%, \beta = 0.8$

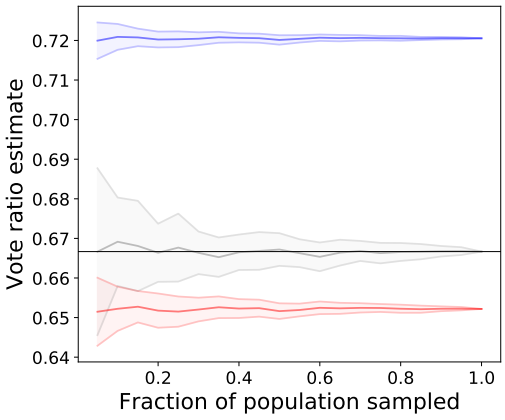


(f) $Q(a) = 75\%, \beta = 0.8$

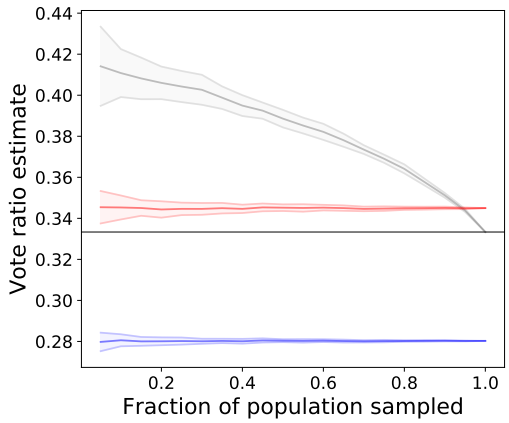
Figure 7.1: Results of vote share estimation using neighbourhood estimator, with biased sampling for $Q(a) = 50\%, 75\%$, $\beta = 0.0, 0.3, 0.8$, $p'_{in} = 0.8$, $p'_{out} = 0.2$, $k = 12$. Red graphs show the vote share estimation based on the first, and blue graphs based on the second neighbourhood vote distributions. Grey graphs show the trivial vote share estimation which, as its estimate, uses the vote share of the sample. Note the estimation bias of both the first and the second neighbourhood estimators for $Q(a) = 75\%$.



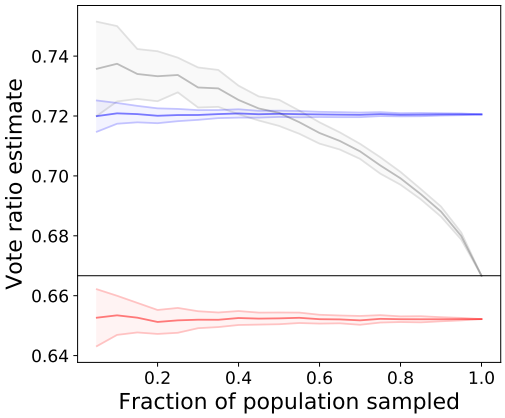
(a) $Q(a) = 33\%, \beta = 0$



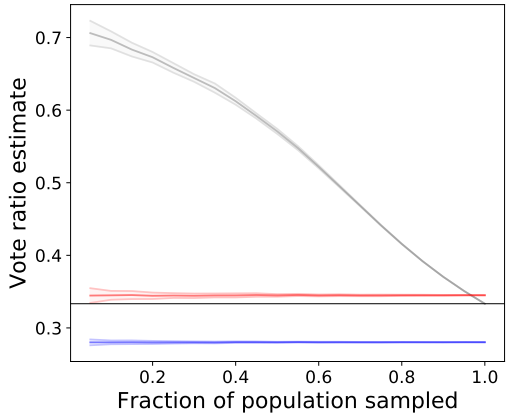
(b) $Q(a) = 67\%, \beta = 0$



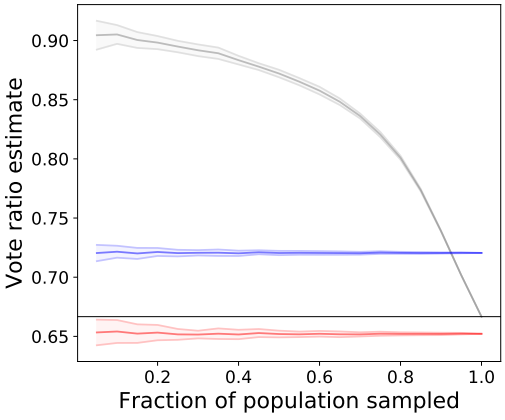
(c) $Q(a) = 33\%, \beta = 0.3$



(d) $Q(a) = 67\%, \beta = 0.3$



(e) $Q(a) = 33\%, \beta = 0.8$



(f) $Q(a) = 67\%, \beta = 0.8$

Figure 7.2: Results of vote share estimation using neighbourhood estimator, with biased sampling for $Q(a) = 33\%, 67\%$, $\beta = 0.0, 0.3, 0.8$, $p'_{in} = 0.8$, $p'_{out} = 0.2$, $k = 12$. Red graphs show the vote share estimation based on the first, and blue graphs based on the second neighbourhood vote distributions. Grey graphs show the trivial vote share estimation which, as its estimate, uses the vote share of the sample. Note the estimation bias of both the first and the second neighbourhood estimators.

7.4. Correcting the prediction error

As we have observed in the previous section, although our method has a smaller variance and converges faster than the trivial method, it is biased – even when whole network is sampled, estimated vote share is not equal to the real one. Scale of this error depends on many factors, and trying to predict it so we can adjust for it proved to be very hard. There is, however, another approach to solving this problem, and it relies on understanding where this error stems from.

Previously, describing equation 7.3, we have stated that it stands true only for independent variables X and Y. But in our case, when X represents the number of a voting neighbours, and Y the node's degree, these variables are not at all independent. And it is because of that dependence that the prediction error arises. Figure 7.3 shows how the node's degree and its number of b voting neighbours is correlated to its number of a voting neighbours.

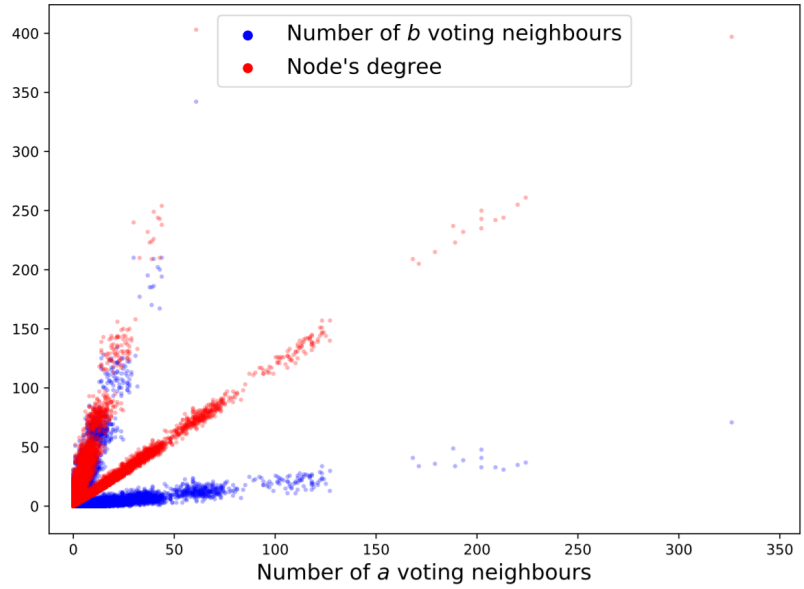


Figure 7.3: Correlation between the amount of a and b voting neighbours (blue) and the amount of a voting neighbours and node's degree (red). Correlation coefficient of the former is 0.2440, and of the latter is 0.9430. It is interesting to note how the two clusters are easy to identify in both plots, as two separate linearly correlated sets of points.

It is evident that this correlation does not stem from the correlation between a and b voting neighbours, but from the fact that node's degree is always larger or equal than the number of a voting neighbours. This leads us to the idea to, instead of using the fraction of neighbours who vote a in our calculations, try to use the ratio of a and b

voting neighbours $q(i,a)/q(i,b) = q(i,a)/(1-q(i,a))$. Although the information we need about each node stays the same, the approximation of the ratio of two Poisson variables becomes much more precise.

This new measure leads to a similar theoretical result as the previous one, as we will show later, and its bias is highly reduced. But, compared to our earlier approach, its variance is much larger, as some nodes, mainly those who themselves vote a, have this a/b ratio very high (recall the amount of nodes for which $q(i,a) = 1$). For this reason, we will use two different measures for two different voting options – for nodes who vote a we will use the ratio b/a, and for nodes who vote b the ratio a/b. We call this new measure *Them vs. us*, and we formally define it as:

$$q'(i) = \begin{cases} \frac{q(i,b)}{q(i,a)} = \frac{1-q(i,a)}{q(i,a)}, & \text{if } v(i) = a \\ \frac{q(i,a)}{q(i,b)} = \frac{q(i,a)}{1-q(i,a)}, & \text{otherwise} \end{cases} \tag{7.16}$$

Similarly to equations 7.6 and 7.7, we define:

$$\mathbb{E}[q'(i)|v(i) = a] = \mu'_a = \mathbb{E}\left[\frac{X_{ab}}{X_{aa}}\right] \approx \frac{\mathbb{E}[X_{ab}]}{\mathbb{E}[X_{aa}]} = \frac{Q(b) \cdot p'_{out}}{Q(a) \cdot p'_{in}} \tag{7.17}$$

$$\mathbb{E}[q'(i)|v(i) = b] = \mu'_b = \mathbb{E}\left[\frac{X_{ba}}{X_{bb}}\right] \approx \frac{\mathbb{E}[X_{ba}]}{\mathbb{E}[X_{bb}]} = \frac{Q(a) \cdot p'_{out}}{Q(b) \cdot p'_{in}} \tag{7.18}$$

There are now two solutions of $Q(a)$ in terms of μ'_a and μ'_b:

$$Q(a) \approx \begin{cases} \frac{-\mu'_b + \sqrt{\mu'_a \mu'_b}}{\mu'_a - \mu'_b} \\ -\frac{\mu'_b + \sqrt{\mu'_a \mu'_b}}{\mu'_a - \mu'_b} \end{cases} \tag{7.19}$$

but the second one is always either negative or larger than 1, for $\mu'_a, \mu'_b \geq 0$, so the resulting estimator of $Q(a)$ is:

$$Q(a) \approx \frac{-\mu'_b + \sqrt{\mu'_a \mu'_b}}{\mu'_a - \mu'_b} \tag{7.20}$$

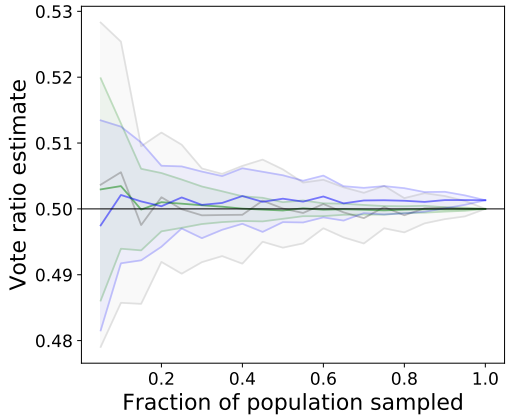
In the following subsection we will present the results of vote share prediction using the *Them vs. us* estimator on biased sampling simulations.

Simulation results of *Them vs. us* method

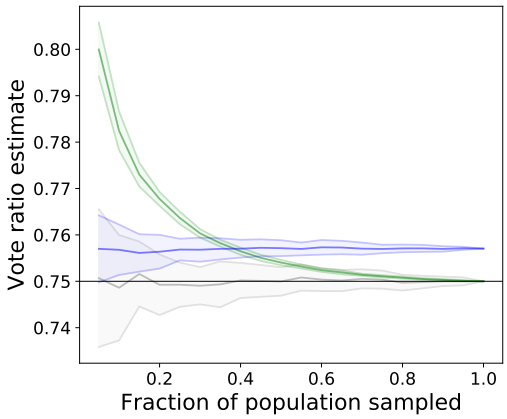
Similarly to our previous simulation results, we are here showing the results of this new estimator. Additionally, plots now also show the results of a trivial estimator which uses not only the votes of observed nodes, but of their neighbours also, which we call the *neighbourhood trivial* estimator. Although the information needed for it is

a superset of the information needed for our approach – as we only look at each neighbourhood cumulatively, while here we need to consider each neighbour individually, to avoid counting them multiple times in different neighbourhoods which would result in a biased estimator – it generally does not perform better, especially when the sampling process is highly biased. Note also its bad performance when there is no bias, but network is unbalanced, as neighbourhoods, on average, have a higher share of majority voters than the network itself, because majority voters' average degree is higher. This difference in average degree is due to the fact that nodes are better connected to other nodes who vote the same, and there are more majority voters in the network.

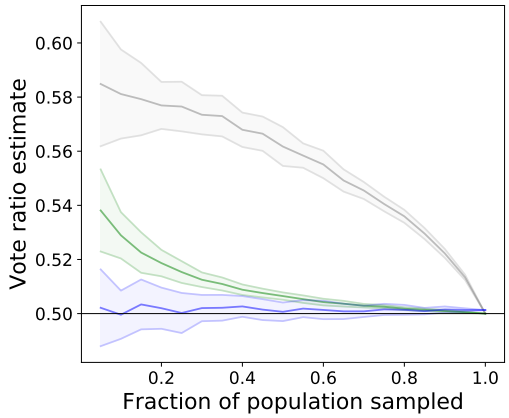
It is important to note that in real-world applications samples are rarely larger than 5%. Because of that, we can say with much confidence that our approach is better not only compared to the trivial, but also to the neighbourhood trivial approach, which uses a superset of information, as we discussed earlier. This neighbourhood trivial approach is comparable to ours only in the case of an unbiased sampling of a balanced network, which is rare.



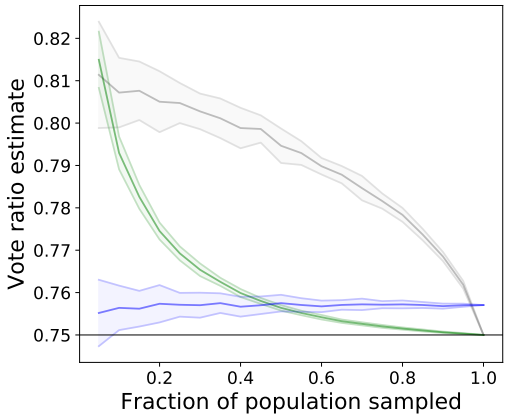
(a) $Q(a) = 50\%, \beta = 0$



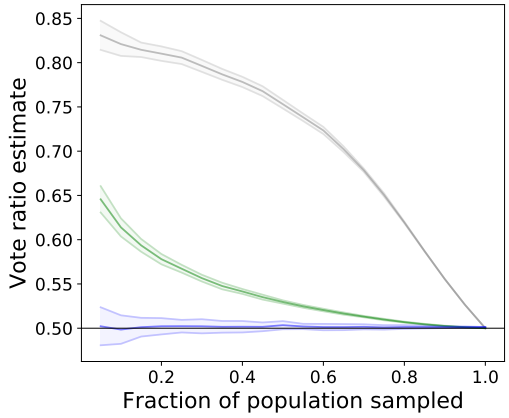
(b) $Q(a) = 75\%, \beta = 0$



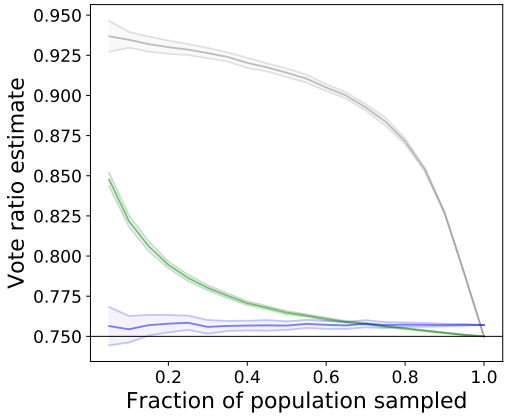
(c) $Q(a) = 50\%, \beta = 0.3$



(d) $Q(a) = 75\%, \beta = 0.3$

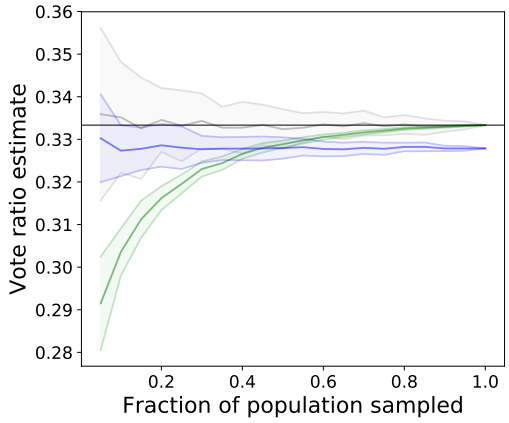


(e) $Q(a) = 50\%, \beta = 0.8$

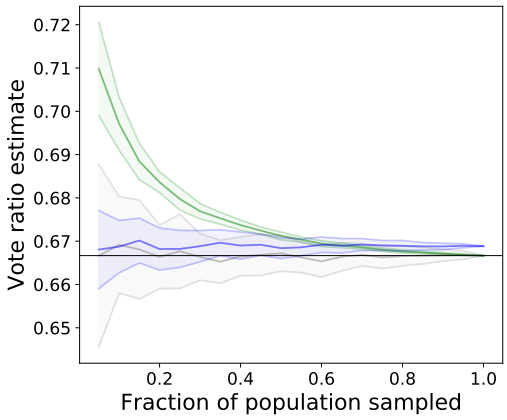


(f) $Q(a) = 75\%, \beta = 0.8$

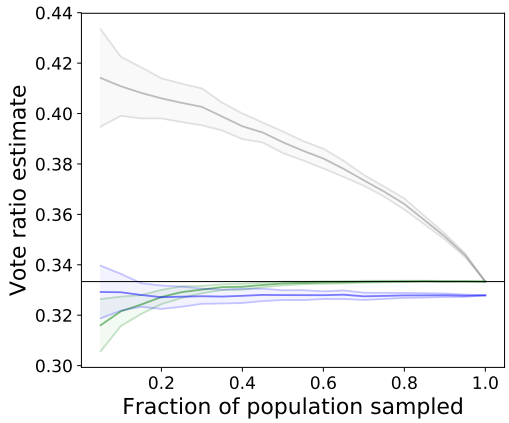
Figure 7.4: Results of vote share estimation using *Them vs. us* estimator, with biased sampling for $Q(a) = 50\%, 75\%$, $\beta = 0.0, 0.3, 0.8$, $p'_{in} = 0.8$, $p'_{out} = 0.2$, $k = 12$. Blue graphs show the results of *Them vs. us* estimator and grey graphs of the trivial estimator. Green graphs show the results of the trivial estimator which, as its sample, uses the neighbours of the sampled nodes too.



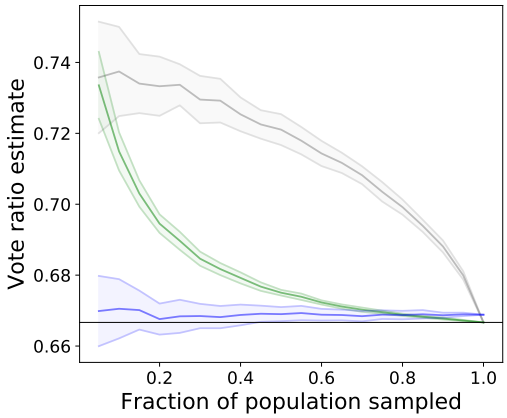
(a) $Q(a) = 33\%, \beta = 0$



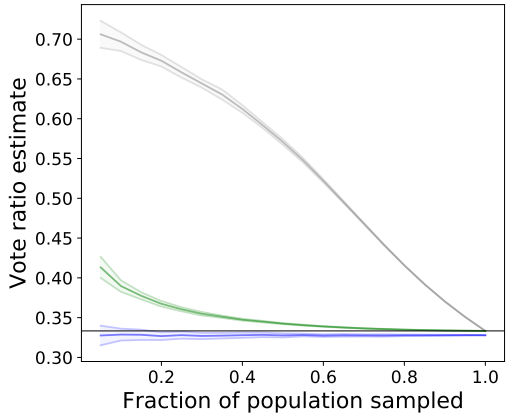
(b) $Q(a) = 67\%, \beta = 0$



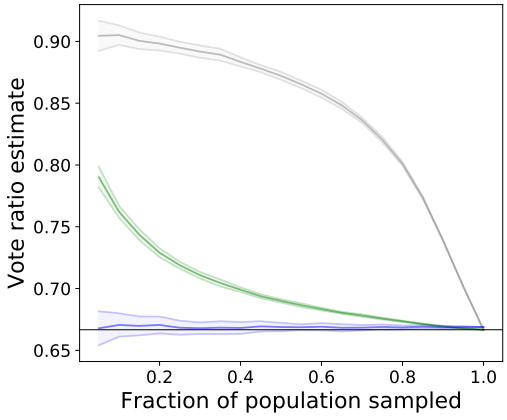
(c) $Q(a) = 33\%, \beta = 0.3$



(d) $Q(a) = 67\%, \beta = 0.3$



(e) $Q(a) = 33\%, \beta = 0.8$



(f) $Q(a) = 67\%, \beta = 0.8$

Figure 7.5: Results of vote share estimation using *Them vs. us* estimator, with biased sampling for $Q(a) = 33\%, 67\%$, $\beta = 0.0, 0.3, 0.8$, $p'_{in} = 0.8$, $p'_{out} = 0.2$, $k = 12$. Blue graphs show the results of *Them vs. us* estimator and grey graphs of the trivial estimator. Green graphs show the results of the trivial estimator which, as its sample, uses the neighbours of the sampled nodes too.

7.5. Unequally dense clusters

So far we have only looked at networks in which both clusters are equally dense – initial parameter p'_{in} was the same for both voters a and b. This was also our only presumption about the social graphs, apart from the idea that they can be successfully modelled with Kronecker graphs, which was already confirmed in Leskovec et al. (2010).

In this chapter we will briefly explore how our approach needs to change if we were to model networks with unequally dense clusters.

7.5.1. Vote ratio estimator

Let $p'_{in,a}$ and $p'_{in,b}$ be the initial probabilities of an edge existing between two nodes who vote the same, a and b respectively. Similarly to equations 7.17 and 7.18, we define:

$$\mathbb{E}[q'(i)|v(i)=a] = \mu''_a = \mathbb{E}\left[\frac{X_{ab}}{X_{aa}}\right] \approx \frac{\mathbb{E}[X_{ab}]}{\mathbb{E}[X_{aa}]} = \frac{Q(b)\cdot p'_{out}}{Q(a)\cdot p'_{in,a}} \tag{7.21}$$

$$\mathbb{E}[q'(i)|v(i)=b] = \mu''_b = \mathbb{E}\left[\frac{X_{ba}}{X_{bb}}\right] \approx \frac{\mathbb{E}[X_{ba}]}{\mathbb{E}[X_{bb}]} = \frac{Q(a)\cdot p'_{out}}{Q(b)\cdot p'_{in,b}} \tag{7.22}$$

In this form, as before when we were solving equations 7.6 and 7.6, and later 7.17 and 7.18, it is infeasible to solve for $Q(a)$ in terms of μ''_a and μ''_b, without the additional constraints. But now, adding either the constraint $p'_{in,a} + p'_{out} = 1$ or $p'_{in,b} + p'_{out} = 1$ is not sufficient, and we cannot add both without the loss of generality. Because of this, we conclude that the final estimator must rely on prior knowledge of parameters $p'_{in,a}$ and $p'_{in,b}$.

In the following section we will explore how these parameters can be estimated. For now, let us presume that we have successfully done so. Solving for $Q(a)$ in terms of μ''_a and μ''_b now becomes feasible when either of the above constraints is introduced. We will use the first one: $p'_{in,a} + p'_{out} = 1$. Both of the above equations now result in their own estimator:

$$Q(a) = \frac{p'_{in,a} - 1}{-\mu''_a \cdot p'_{in,a} + p'_{in,a} - 1} \tag{7.23}$$

$$Q(a) = \frac{\mu''_b \cdot p'_{in,b}}{\mu''_b \cdot p'_{in,b} - p'_{in,b} + 1} \tag{7.24}$$

To increase our precision, as our prediction we will use their average, but it is important to note that prediction, although a less successful one[3], can be made with either one. This then means that, without a single direct information about one of the two voting groups, we can still make predictions much better than the trivial approach.

7.5.2. Results

Figures 7.6 and 7.7 show the results of this estimator when $p'_{in,a} = 0.8$, $p'_{in,b} = 0.6$ and $p'_{out} = 0.2$, and when $p'_{in,a}$ and $p'_{in,b}$ are known in advance. For comparison, we have included the results of the previous, *Them vs. us* estimator which is unaware of the different p'_{in} parameters. Although we can clearly see that if we presume that $p'_{in,a} = p'_{in,b}$ we do get an estimation bias, it is still true that this estimation is, for small samples, better than the neighbourhood trivial one. When, on the other hand, $p'_{in,a}$ and $p'_{in,b}$ are known in advance, estimation is even better, and estimation bias is negligible.

This motivates us to try to estimate parameters $p'_{in,a}$ and $p'_{in,b}$, if we presume that we know which node pairs from our sample are connected. We examine this further in the next section.

7.5.3. Estimating cluster densities

In general, we do not have access to the exact values of $p'_{in,a}$ and $p'_{in,b}$. In that case, we need to estimate them first, before using the above method. To be more precise, we need to estimate the parameters $R_a = p'_{out}/p'_{in,a}$ and $R_b = p'_{out}/p'_{in,b}$. This can be done in a straightforward manner, estimating these values directly from the observed subgraph – for example, parameter p'_{out} can be estimated as the ratio between the number of observed edges between two differently voting nodes, and the maximum possible number of such edges, which is a product of the number of nodes voting a, and the number of nodes voting b in the observed sample. We can now write

$$\begin{aligned} \mu''_a &= R_a \cdot \frac{Q(a)}{1 - Q(a)} \\ \mu''_b &= R_b \cdot \frac{1 - Q(a)}{Q(a)} \end{aligned} \tag{7.25}$$

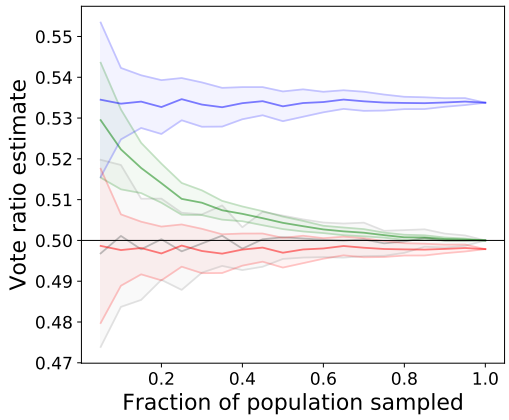
Our estimators now become

$$Q(a) = \frac{R_a}{\mu''_a + R_a} \tag{7.26}$$

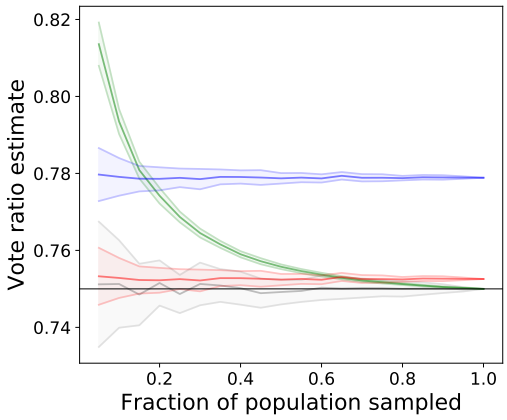
[3]Both of these estimators are slightly biased and have a larger variance then their average. As they are biased in the opposite directions, averaging them cancels this bias out.

$$Q(a) = \frac{\mu_b''}{\mu_b'' + R_b} \tag{7.27}$$

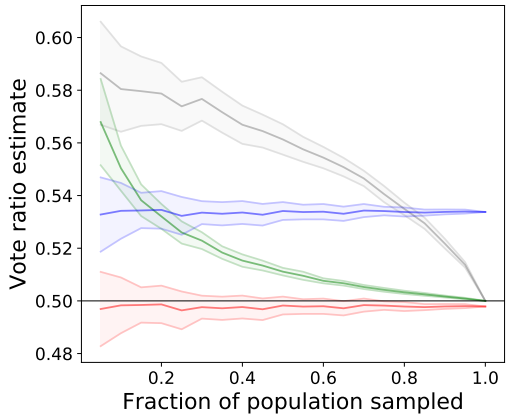
As before, we will be using the average of these two estimators. Figures 7.8 and 7.9 show the results of vote share estimation using this estimator on the same networks used in the previous section. As we can see, although variance is significantly higher when R_a and R_b are not known in advance, estimator performs better than when we presumed that $p'_{in,a} = p'_{in,b}$, and is in all cases better than the neighbourhood trivial estimator.



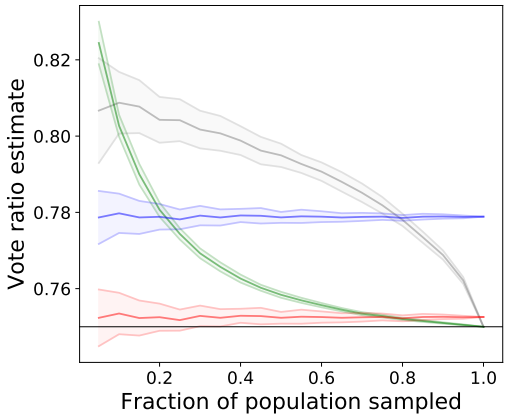
(a) $Q(a) = 50\%, \beta = 0$



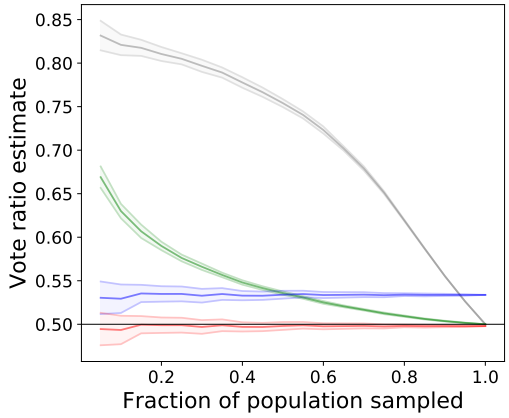
(b) $Q(a) = 75\%, \beta = 0$



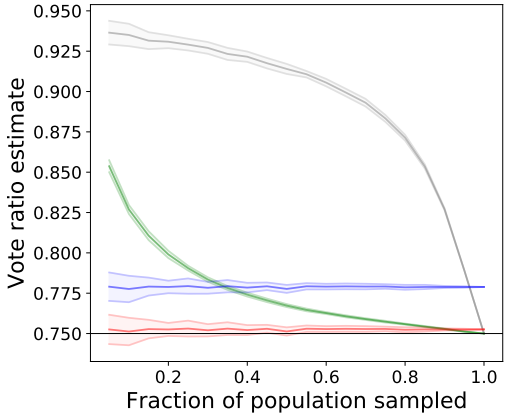
(c) $Q(a) = 50\%, \beta = 0.3$



(d) $Q(a) = 75\%, \beta = 0.3$

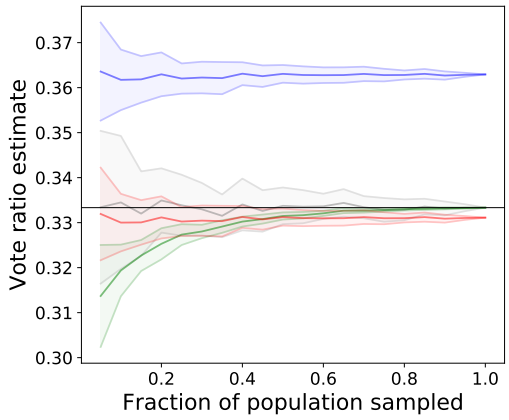


(e) $Q(a) = 50\%, \beta = 0.8$

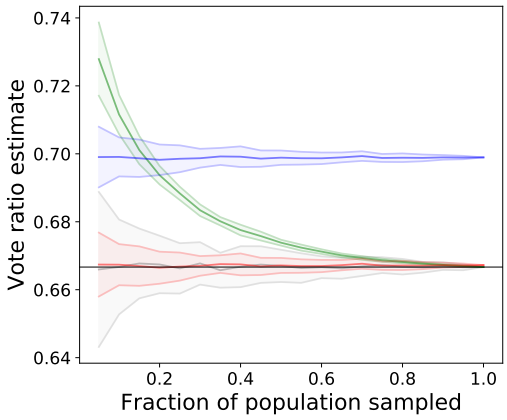


(f) $Q(a) = 75\%, \beta = 0.8$

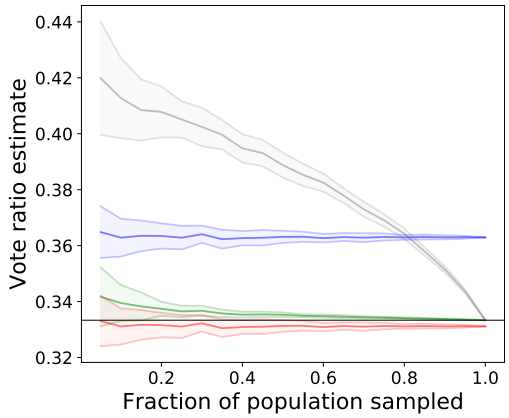
Figure 7.6: Results of vote share estimation using *Them vs. us* estimator when p'_{in} and p'_{out} are different but known, with biased sampling for $Q(a) = 50\%, 75\%$, $\beta = 0.0, 0.5, 0.9$, $p'_{in,a} = 0.8$, $p'_{in,b} = 0.6$, $p'_{out} = 0.2$, $k = 12$. Red graphs show the results of *Them vs. us* estimator when p'_{in} and p'_{out} are known. Blue graphs show the results of a normal *Them vs. us* estimator, grey graphs of the trivial estimator, and green graphs of the trivial neighbourhood estimator. Note how there is no estimation bias when p'_{in} and p'_{out} are known.



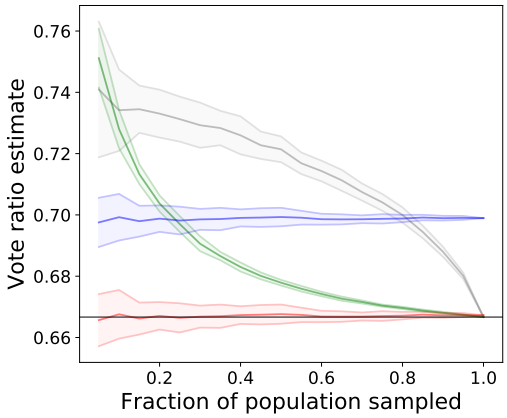
(a) $Q(a) = 33\%, \beta = 0$



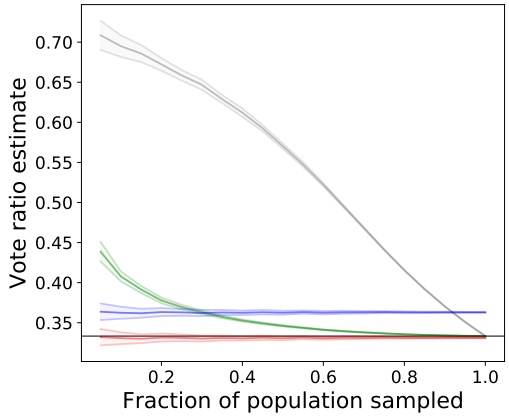
(b) $Q(a) = 67\%, \beta = 0$



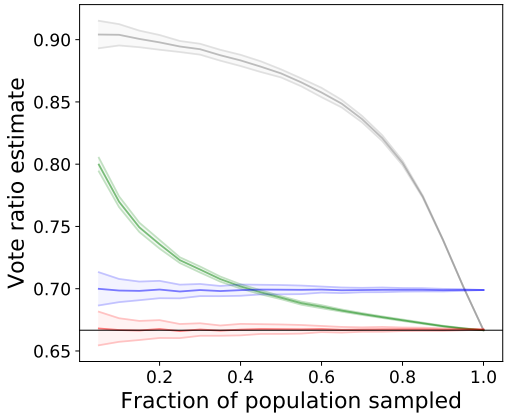
(c) $Q(a) = 33\%, \beta = 0.3$



(d) $Q(a) = 67\%, \beta = 0.3$

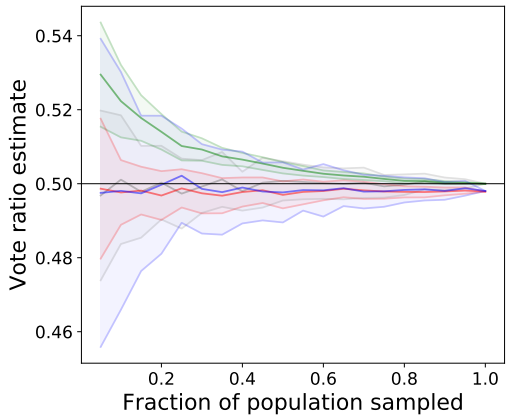


(e) $Q(a) = 33\%, \beta = 0.8$

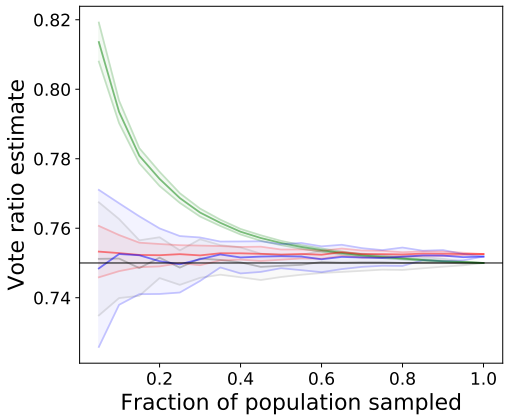


(f) $Q(a) = 67\%, \beta = 0.8$

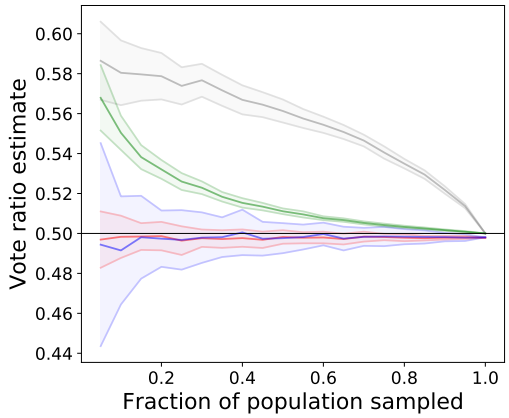
Figure 7.7: Results of vote share estimation using *Them vs. us* estimator when p'_{in} and p'_{out} are different but known, with biased sampling for $Q(a) = 33\%, 67\%$, $\beta = 0.0, 0.3, 0.8$, $p'_{in} = 0.8$, $p'_{in,b} = 0.6$, $p'_{out} = 0.2$, $k = 12$. Red graphs show the results of *Them vs. us* estimator when p'_{in} and p'_{out} are known. Blue graphs show the results of a normal *Them vs. us* estimator, grey graphs of the trivial estimator, and green graphs of the trivial neighbourhood estimator. Note how there is no estimation bias when p'_{in} and p'_{out} are known.



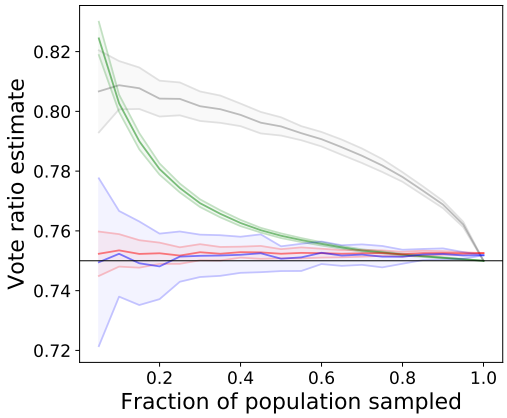
(a) $Q(a) = 50\%, \beta = 0$



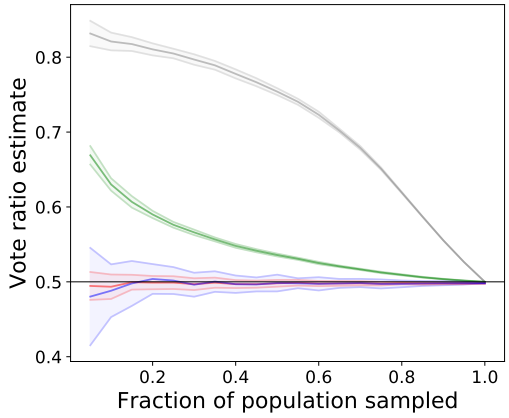
(b) $Q(a) = 75\%, \beta = 0$



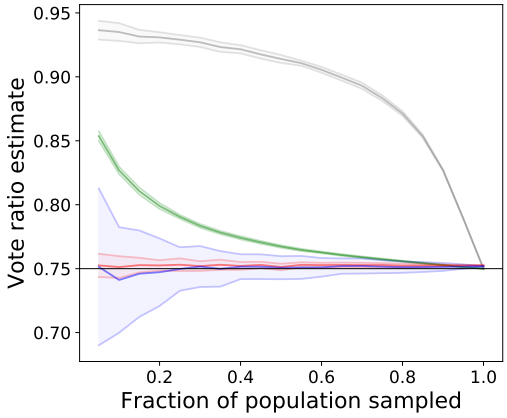
(c) $Q(a) = 50\%, \beta = 0.3$



(d) $Q(a) = 75\%, \beta = 0.3$

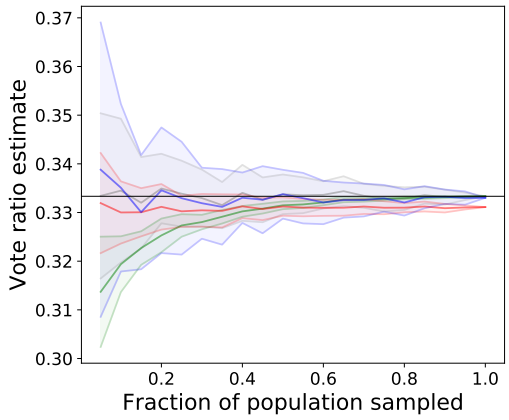


(e) $Q(a) = 50\%, \beta = 0.8$

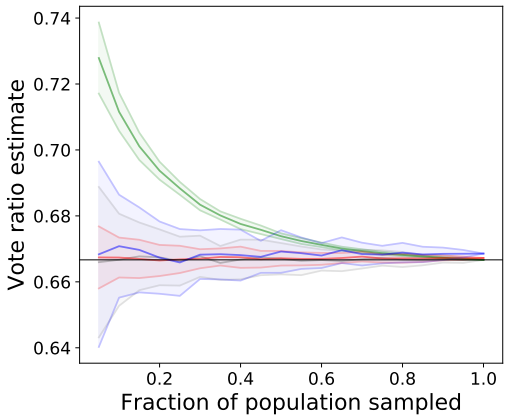


(f) $Q(a) = 75\%, \beta = 0.8$

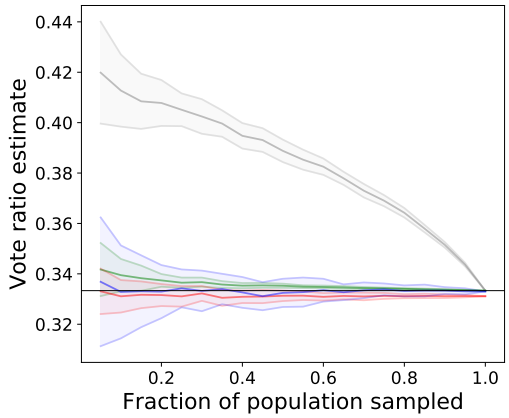
Figure 7.8: Results of vote share estimation using *Them vs. us* estimator when p'_{in} and p'_{out} are different and unknown, with biased sampling for $Q(a) = 50\%, 75\%$, $\beta = 0.0, 0.5, 0.9$, $p'_{in,a} = 0.8$, $p'_{in,b} = 0.6$, $p'_{out} = 0.2$, $k = 12$. Red graphs show the results of *Them vs. us* estimator when p'_{in} and p'_{out} are known. Blue graphs show the results of the same estimator, but with p'_{in} and p'_{out} estimated from the sample. Grey graphs show the results of the trivial estimator, and green graphs of the trivial neighbourhood estimator.



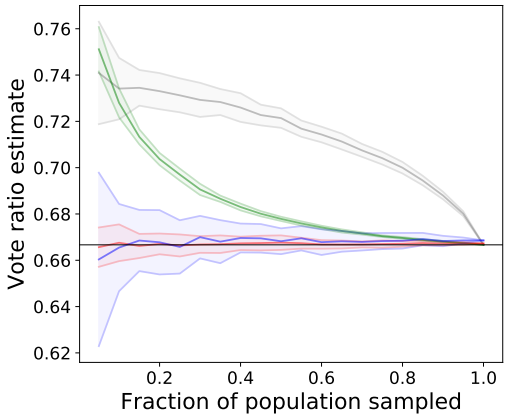
(a) $Q(a) = 33\%, \beta = 0$



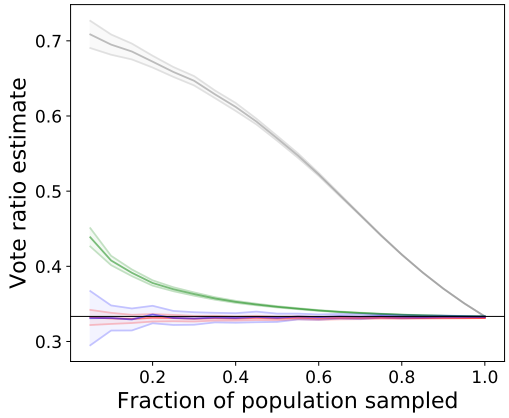
(b) $Q(a) = 67\%, \beta = 0$



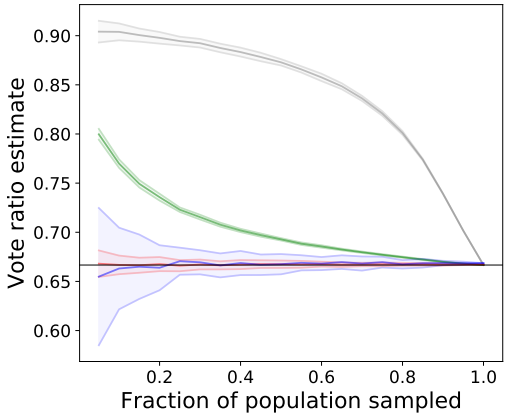
(c) $Q(a) = 33\%, \beta = 0.3$



(d) $Q(a) = 67\%, \beta = 0.3$



(e) $Q(a) = 33\%, \beta = 0.8$



(f) $Q(a) = 67\%, \beta = 0.8$

Figure 7.9: Results of vote share estimation using *Them vs. us* estimator when p'_{in} and p'_{out} are different but known, with biased sampling for $Q(a) = 33\%, 67\%$, $\beta = 0.0, 0.3, 0.8$, $p'_{in} = 0.8$, $p'_{in,b} = 0.6$, $p'_{out} = 0.2$, $k = 12$. Red graphs show the results of *Them vs. us* estimator when p'_{in} and p'_{out} are known. Blue graphs show the results of the same estimator, but with p'_{in} and p'_{out} estimated from the sample. Grey graphs show the results of the trivial estimator, and green graphs of the trivial neighbourhood estimator.

8. Conclusion

Election results prediction using the social network neighbourhood information is an interesting problem which led us to multiple novel insights into the topologies and dynamics of these networks. By using the advanced method for social network modelling – the stochastic Kronecker graph model – and expanding it so it supports polarized networks, i.e. networks which are split into two large clusters, we have identified the neighbourhood information needed to infer the sizes of these clusters, predicting the election outcome.

Three different network vote share estimators were presented – first neighbourhood estimator, second neighbourhood estimator, and *Them vs. us* estimator, which as its basis uses the ratio of neighbours who vote differently, and those who vote the same as the central node.

Although in our research we concentrated on elections, we believe that these methods can be used for other predictive modelling in networks where polarization occurs, since the basis of out approach is the stochastic Kronecker graph model which is highly adaptable.

BIBLIOGRAPHY

Lada A Adamic and Natalie Glance. The political blogosphere and the 2004 us election: divided they blog. U *Proceedings of the 3rd international workshop on Link discovery*, stranice 36–43. ACM, 2005.

Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

Engin Bozdag and Jeroen van den Hoven. Breaking the filter bubble: democracy and design. *Ethics and Information Technology*, 17(4):249–265, 2015.

Derek J. de Solla Price. Networks of scientific papers. *Science*, 149(3683):510–515, 1965. ISSN 0036-8075. doi: 10.1126/science.149.3683.510.

Michela Del Vicario, Gianna Vivaldo, Alessandro Bessi, Fabiana Zollo, Antonio Scala, Guido Caldarelli, and Walter Quattrociocchi. Echo chambers: Emotional contagion and group polarization on facebook. *Scientific reports*, 6, 2016.

Paul Erdös and Alfréd Rényi. On random graphs, i. *Publicationes Mathematicae (Debrecen)*, 6:290–297, 1959.

Dani Gamerman and Hedibert F Lopes. *Markov chain Monte Carlo: stochastic simulation for Bayesian inference*. CRC Press, 2006.

Scott A Golder, Dennis M Wilkinson, and Bernardo A Huberman. Rhythms of social interaction: Messaging within a massive online network. *Communities and technologies 2007*, stranice 41–66, 2007.

Lucien Le Cam et al. An approximation theorem for the poisson binomial distribution. *Pacific J. Math*, 10(4):1181–1197, 1960.

Jure Leskovec and Rok Sosič. Snap: A general-purpose network analysis and graph-mining library. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8 (1):1, 2016.

Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. U *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, stranice 177–187. ACM, 2005.

Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos, and Zoubin Ghahramani. Kronecker graphs: An approach to modeling networks. *arXiv preprint arXiv:0812.4905*, 2008.

Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos, and Zoubin Ghahramani. Kronecker graphs: An approach to modeling networks. *Journal of Machine Learning Research*, 11(Feb):985–1042, 2010.

Mark Newman. *Networks: An Introduction*. Oxford University Press, 2010. ISBN 0199206651.

RC Ogliore, GR Huss, and K Nagashima. Ratio estimation in sims analysis. *Nuclear instruments and methods in physics research section B: beam interactions with materials and atoms*, 269(17):1910–1918, 2011.

Eli Pariser. *The Filter Bubble: How the New Personalized Web Is Changing What We Read and How We Think*. Penguin Books, 2012. ISBN 0143121235.

Matija Piškorec, Nino Antulov-Fantulin, Iva Miholić, Tomislav Šmuc, and Mile Šikić. Modeling peer and external influence in online social networks. *arXiv preprint arXiv:1610.08262*, 2016.

Derek De Solla Price. A general theory of bibliometric and other cumulative advantage processes. *Journal of the American Society for Information Science*, 27(5):292–306, 1976. ISSN 1097-4571. doi: 10.1002/asi.4630270505.

Chaoming Song, Shlomo Havlin, and Hernan A Makse. Self-similarity of complex networks. *Nature*, 433(7024):392–395, 2005.

Sudhir Leslie Tauro, Christopher Palmer, Georgos Siganos, and Michalis Faloutsos. A simple conceptual model for the internet topology. U *Global Telecommunications Conference, 2001. GLOBECOM'01. IEEE*, svezak 3, stranice 1667–1671. IEEE, 2001.

Jeffrey Travers and Stanley Milgram. The small world problem. *Phychology Today*, 1: 61–67, 1967.

Wayne W Zachary. An information flow model for conflict and fission in small groups. *Journal of anthropological research*, 33(4):452–473, 1977.

Influence of Neighbours on One's Own opinion

Abstract

Election results prediction using the social network neighbourhood information is a complex network theory problem which includes advanced complex network modelling and complex network analysis techniques. Presuming that social networks display a high degree of homophily, which in binary elections leads to polarization and a split into two giant clusters, we translated this problem into a domain of cluster size inference, for these two clusters. First, we have analysed how the social neighbourhood is related to a person's voting preference. Next, we have shown that the prediction of this preference can be done from the neighbourhood information in a straightforward manner. Finally, we have shown how, by modelling social networks with the expanded stochastic Kronecker graph model and using our novel *Them vs. us* estimator, network's vote share can successfully be estimated, even from a small and highly biased node sample.

Keywords: Election results prediction, opinion inference, social neighbourhood, Kronecker graph model, social network polarization.

Ujecaj susjeda na vlastito mišljenje

Sažetak

Predviđanje rezultata izbora koristeći informacije o susjedstvu društvene mreže je problem iz teorije kompleksnih mreža koji uključuje napredne tehnike modeliranja i analize kompleksnih mreža. Pod pretpostavkom da društvena mreža pokazuje visok stupanj homofilije, što kod izbora s dvije izborne opcije dovodi do polarizacije i podjele mreže u dva velika klastera, preveli smo ovaj problem u domenu određivanja veličine klastera. Prvi dio rada analizira vezu između susjedstva društvene mreže i opcije za koju centralni čvor glasuje. Sljedeći dio pokazuje kako se ova opcija može izravno i uspješno predvidjeti iz informacija o društvenom susjedstvu. Naposljetku, pokazali smo kako se, modelirajući društvene mreže s proširenim stohastičkim Kroneckerovim grafovima te koristeći *Them vs. us* estimator, udio glasova u mreži može uspješno odrediti, čak i kada je uzorak iznimno nereprezentativan.

Ključne riječi: Predviđanje rezultata izbora, predviđanje mišljenja, susjedstvo društvene mreže, Kroneckerov model grafova, polarizacija društvene mreže.