Fall 12-21-2011

Mind Economy: Dynamic Graph Analysis of Communications

Alexy Khrabrov
University of Pennsylvania, alexy.khrabrov@gmail.com

Mind Economy: Dynamic Graph Analysis of Communications

Abstract
Social networks are growing in reach and impact but little is known about their structure, dynamics, or users' behaviors. New techniques and approaches are needed to study and understand why these networks attract users' persistent attention, and how the networks evolve. This thesis investigates questions that arise when modeling human behavior in social networks, and its main contributions are:

• an infrastructure and methodology for understanding communication on graphs;

• identification and exploration of sub-communities;

• metrics for identifying effective communicators in dynamic graphs;

• a new definition of dynamic, reciprocal social capital and its iterative computation

• a methodology to study influence in social networks in detail, using

• a class hierarchy established by social capital

• simulations mixed with reality across time and capital classes

• various attachment strategies, e.g. via friends-of-friends or full utility optimization

• a framework for answering questions such as "are these influentials accidental"

• discovery of the "middle class" of social networks, which as shown with our new metrics and simulations is the real influential in many processes

Our methods have already lead to the discovery of "mind economies" within Twitter, where interactions are designed to increase ratings as well as promoting topics of interest and whole subgroups. Reciprocal social capital metrics identify the "middle class" of Twitter which does most of the "long-term" talking, carrying the bulk of the system-sustaining conversations. We show that this middle class wields the most of the actual influence we should care about — these are not "accidental influentials." Our approach is of interest to computer scientists, social scientists, economists, marketers, recruiters, and social media builders who want to find and present new ways of exploring, browsing, analyzing, and sustaining online social networks.

Degree Type
Dissertation

Degree Name
Doctor of Philosophy (PhD)

Graduate Group
Computer and Information Science

MIND ECONOMY:
DYNAMIC GRAPH ANALYSIS OF COMMUNICATIONS

Alexy Khrabrov

A DISSERTATION
in
Computer and Information Science

Presented to the Faculties of the University of Pennsylvania
in Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

2011

Supervisor of Dissertation

Lyle H. Ungar, Associate Professor
Computer and Information Science

Graduate Group Chair

Jianbo Shi, Professor
Computer and Information Science

Dissertation Committee

Jonathan M. Smith

Professor
Engineering and Applied Science
Computer and Information Science

Mark Liberman

Professor
Linguistics
Computer and Information Science

Shawndra Hill

Assistant Professor
Operations and Information Management

George Cybenko

Professor
Thayer School of Engineering
Dartmouth College

MIND ECONOMY:
DYNAMIC GRAPH ANALYSIS OF COMMUNICATIONS

Alexy Khrabrov

To my father, Professor Victor Alexandrovich Khrabrov,
the inventor of the first electric rocket flown into space.

Per aspera ad astra

Acknowledgments

This thesis is a synthesis of many years of research and industrial experience, which had begun with my arrival in the United States at the invitation of Prof. George Cybenko of Thayer School of Engineering, Dartmouth College. Now, coming a full circle, I am a researcher at Thayer again, and a citizen of the United States myself, with the new citizens — our children Edward and Eva — added already with my wife Olga (and more on the way). My family makes everything clear and worthwhile.

I want to thank George for the great opportunity he opened, his scientific guidance, and sharing his wisdom and ability to see what's really important. When working with him, you appreciate how invariably sharp, patient and understanding George is with people in any project he undertakes. As a great scientist himself with strong computational contributions and a mathematical view of the world, managing multiple large-scale projects without ever losing his cool, George is always the example to follow for me.

I want to thank my new country, the United States of America, for upholding freedom and liberty, leading the world in innovation, and providing people of all backgrounds with amazing opportunities to come together and do great things for the world and each other. Through its democratic system and appreciation of science and technology, the US enables such great institutions of learning as the University of Pennsylvania and Dartmouth College, and such singular laboratories of the Internet innovation as the Silicon Valley — where we're headed next.

I thank Professor Emeritus Noah Prywes, one of the founders of the CIS, who gave me my first job at CCCC, one of the oldest computer companies in the US, and recommended me to the CIS program at Penn. Bonnie Webber was my original Penn advisor. Her knowledge, focus on medical research applications, and engaging manner illuminated my first year at Penn, and with her I also started working with Curt Langlotz, one of the first MD / PhDs using natural language processing in medical domain — my first application of functional programming with both academic and industrial impact, a tradition continued throughout my career and in this thesis. Bonnie had left for Edinburgh and was succeeded by Lyle Ungar. When I took my first AI class with Lyle, I wished this lively guy in plaid shirt could be my advisor — which did happen, and Lyle had become one of the most engaging scientists I was lucky to work with. Lyle fuses broad knowledge of data mining and machine learning with very practical approach to problem solving — and he enables it by being one of the best communicators inside and outside of science I've ever met. I'm glad he had put up with me through all these years, with so many ideas — and I'm positive we'll find more areas of collaboration going forward.

My committee is an amazing group of experts in social networks, data mining, and computational linguistics. Mark Liberman was supportive of my research at Dartmouth and had asked the key question — are these Accidental Influentials, by any chance? — which employed our Reciprocal Social Capital metrics to the fullest. Jonathan Smith, the chair, grounds it in all-encompassing computer networking expertise, and Shawndra Hill adds an operational and business perspective which, as you can see, is the most welcome pattern here.

The Human Terrain group at Thayer School, as well as friends in Computer Science department provided a stimulating intellectual environment. Vincent Berk, John Murphy, Ian Gregorio-De Souza, David Twardowski, Nils Sandell, Gabe Stocco and Robert Savell were the best colleagues and officemates. Sergey Bratus, my classmate since the math high school #57 in Moscow, is always a great friend and idea checker. Paul Thompson, Sergey, and Anna Rumshisky collaborated on an applied ontology research for GM, which kept me in Dartmouth's gravity field even form Moscow and St Petersburg.

Mikhail Gronas, although a professor of Russian, can well fill a Valley startup incubator with ideas. Andrew Campbell does cool, and beautifully managed, pervasive computing research, while Lorenzo Torresani, a top machine learning guy whose courses I audited and enjoyed, was a neighbor dad, his great family becoming friends with ours. We inherited our Dartmouth apartment, and many great things and even ideas, form Bob Hearn, a legendary figure with interests and contributions to Apple, Bach, games and theories of the mind.

Professor Rajiv Alur, the former graduate chair, has outlined the graduation plan for me to execute at Dartmouth, which I followed. Professor Jianbo Shi, the current graduate chair, guided it through completion and took the time to listen to the whole presentation in advance, providing insightful feedback. Mark Steedman and Mitch Marcus gave important Computational Linguistics courses, with complimentary approaches, and years worth of direction.

Alexandrin Popescul and Andrew Schein are steadfast friends and colleagues since the Ph.D. program under Lyle, and now future neighbors, — with Alex also a colleague again, — in the Bay Area. Alex and Andrew provided numerous hours of discussion of the technologies and also their purposes and applications. Their successful application of data mining in industry is the best use of what Lyle and others had taught us and an example to follow, again.

Val Tannen had first introduced me to ML, which I'm using to this day (and in this thesis extensively). He and Jean Gallier provided steadfast academic support at Penn, and Norm Badler and Rajiv Alur helped my candidacy come around and track to the plan, keeping the eyes on the prize. Eduardo Glandt had always made things nicer with his charm, and oversees SEAS and CIS through amazing advances, making it an even better place to graduate from.

My father, Professor Victor Khrabrov, is a physicist and the inventor of the first electric rocket flown into space. I owe my interests in science and engineering, as well as in Cicero, literature, arts, and much more, to him. He was always a supreme optimist and was happy to see me following through with my Ph.D. I hope he'll be proud of me from wherever he is now. My mother, Anna Khrabrova (Drabkina), was always supportive and believed in me to the extent she'd become a resident of Philadelphia too. To her, my Penn odyssey had a benefit of my regular visits (by train!), and enjoying Philly together, including its great classical orchestra at the beautiful Kimmel Center.

I have to thank the city of Philadelphia here, for being a home city, a social network enabling science and creativity and inseparable from the Penn experience. Dartmouth, as my first home in the US, and then our first long-term family home, is also a special place like no other. (Snowshoeing!) While on the subject of places, I thank the Lake Baikal for bringing me and Olga together, which opened the new vistas we'd come to love—Buenos Aires, Seattle, Hanover, and San Francisco to come.

Dmitri Levonian, a founding partner at Renova Capital (now Svarog Capital Advisors), is a rare example of a deep thinker and successful financier who can implement his vision via scaling a global business. Working with him I got a chance to attend to all aspects of an R&D enterprise, go together to the Machine Learning Summit in Copenhagen, and do focused computational research of English and Russian.

My friends and colleagues at Amazon.com, notably Dean Kassmann, Chris Jones, Ryan Rawson, and Jacob Nelson, taught me that you can — and should! — apply advanced concepts in process control, supply chain management, personalization, and functional programming to reality at scale to improve productivity and do better.

Peter Yianilos has brought me to the NECI, with the startup idea of a global, eternal, indestructible Intermemory – whose time is still to come one day. Peter is awesome, and can make things clear simply by the tone of his voice; his math skills are a stellar match to his pioneering business record, and overall he is a great example to follow. The team at NECI included Lee Giles, a father figure to us all, Steve Lawrence (now at Google) and Dave Pennock (now at Yahoo Research), all great friends. Gary Flake went on to found and lead industrial labs at Yahoo and Microsoft, showing how to scale serious research in practice.

The teams at Google, Microsoft, Efficient Frontier, and Klout had all shown interest in the research presented here. I have chosen the latter, for its startup spirit and vision, but I respect all of them and see them as my future colleagues in the field of practical social network mining.

I have to mention those whom I always remember — my grandfathers who fought in the World War II and defended our lives. My paternal grandfather, Alexander Khrabrov, was lost in action in 1941, defending Moscow. My maternal grandfather, Semion Drabkin, had become a war hero, delivering the ultimatum at Stalingrad, personally capturing Nazi criminals who testified at Nuremberg, liberating cities, and working together with the Allies in the denazification of Germany. He went on to become a chief missile designer, — but it were his grandsons, not his missiles, which ended up in the US. His perseverance at Stalingrad had inspired my own through the Ph.D. process.

Last, but not least, I appreciate the world-wide coffee shops which made writing this thesis so enjoyable. *La Colombe* in Philadelphia, *Small World* in Princeton, *Joe Bar* in Seattle, *Tucker Box* in White River Junction, *Boston Dreams* in Windsor, Vermont, and *Dirt Cowboy* and *Rosie Jekes* in Hanover, New Hampshire — I love the people, the coffee, the ideas, and it all comes together there!

ABSTRACT

MIND ECONOMY:

DYNAMIC GRAPH ANALYSIS OF COMMUNICATIONS

Alexy Khrabrov

Lyle H. Ungar

Online social networks are novel mechanisms used for distributed interaction whose growth has been enabled by affordable and ubiquitous communications and computing. Facebook has over 700 million members (larger than the US population), and Twitter broadcasts over 200 million "tweets" daily. (Tweets are projected to reach 1 billion daily within a year, compared with about 600 million pieces of mail handled by USPS daily.) Social networks are growing in reach and impact but little is known about their structure, dynamics, or users' behaviors. New techniques and approaches are needed to study and understand why these networks attract users' persistent attention, and how the networks evolve.

This thesis investigates questions that arise when modeling human behavior in social networks, and its main contributions are:

- an infrastructure and methodology for understanding communication on graphs;
- identification and exploration of sub-communities;
- metrics for identifying effective communicators in dynamic graphs;
- a new definition of dynamic, reciprocal social capital and its iterative computation
- a methodology to study influence in social networks in detail, using
- a class hierarchy established by social capital
- simulations mixed with reality across time and capital classes
- various attachment strategies, e.g. via friends-of-friends or full utility optimization
- a framework for answering questions such as "are these influentials accidental," by performing multiple simulations with various starting conditions from reality, and seeing how various rules

lead to new worlds in which the new winners emerge, comparing those possible winners the real ones, and using the comparison to segment the factors contributing to success

- discovery of the "middle class" of social networks, which as shown with our new metrics and simulations is the real influential in many processes

Our methods have already lead to the discovery of "mind economies" within Twitter, where interactions are designed to increase ratings as well as promoting topics of interest and whole subgroups. Reciprocal social capital metrics identify the "middle class" of Twitter which does most of the "long-term" talking, carrying the bulk of the system-sustaining conversations. We show that this middle class wields the most of the actual influence we should care about — these are not "accidental influentials." Our approach is of interest to computer scientists, social scientists, economists, marketers, recruiters, and social media builders who want to find and present new ways of exploring, browsing, analyzing, and sustaining online social networks.

Contents

List of Tables

List of Figures

Chapter 1

Summary

The main contributions of this thesis are the ideas of reciprocity in defining social capital computationally in social networks, and the class hierarchy resulting from it, together with a series of metrics revealing difference in behavior of different classes and patterns of their interactions.

We define reciprocal social capital as a process placing a single real value on each user at each time cycle (e.g., day to day). The values are recomputed transactionally every cycle based on their previous values and the actions of the individuals in the last cycle. Each user maintains a balance of communications with any other user when a directed tweet exists between the two. If you tweeted more to somebody than he tweeted to you, he "owes" you the difference. In this model, people who are cognizant of such an imbalance and try to keep their balance even are motivated to keep the threads of conversations going. We reward those who "repay" their "debts" in each cycle. We similarly reward the users who were able to get some of the replies "owed" to them. We also reward users for getting "unsolicited" mentions. We do not reward outgoing tweets which are not replies. This reflects the Internet dynamics in which it does not cost much to send out many tweets to all kinds of people (as is done by spambots). Previous capital is decayed each cycle and the rewards from the current cycle are added, hence one must maintain a high pace of productive activity (leading to capital rewards) in order to maintain a high ranking.

Once we have the value of the metric computed for each user, we can rank them by their social capital. We observe that the rankings follow a power law distribution which leads to a hierarchy. Such hierarchy is common in any kind of financial wealth distribution in an economy. We define classes of

exponentially increasing sizes, and we investigate how various communication behaviors help users to stay in higher or lower classes.

In order to understand human communication behavior, we represent it as a combination of one or several well-defined fundamental strategies. We preserve the number of users and their order of appearance in the network, as well as their out-degrees in each cycle, and alter only the attachment points (whom the talk to). We try three kinds of attachment strategies – proportionally to in-degree, or to social capital, or random; and we follow it in three distinct target scopes – finding attachment targets globally, among the friends of friends, replicating a common friends discovery mechanism on Twitter, or through a strategy which optimizes reciprocal social capital of a user directly based on the reply reward in his control. Each simulation is a probabilistic mixture of these three kinds of attachment and three target scopes.

We allow for a given length of reality to be followed verbatim before starting a simulation. We compare sensitivity of various behaviors to the length of such a reality prefix, to see how the starting conditions affect the results. While seeding the communication balances and other parameters to bootstrap the simulation, we compute the reciprocal social capital for the prefix, so the simulation "hits the ground running."

Finally, we have both overall and selective simulations in which we change everybody's behavior, or we change it only the people in a selected set of classes, respectively. This allows us to see when a certain class matches a proposed model of communication better than others. The selective, or bucketed, simulations recompute the class hierarchy before every cycle, and simulate only the users in the classes specified while replicating the behaviors of others verbatim from reality.

For each of the simulations we obtain a class distribution for each day of the study. We then compare each class by day with the reality, and consider how well each mix of behaviors captures that class.

We find that the simulations based on our social capital capture the middle class, the one million people between the "poor" and the "elites," most effectively. Capital-based simulations are less sensitive to the starting conditions than simulations based on in-degree. Friends-of-friends and local utility optimization strategies take group dynamics into account and thus achieve a good match for many

"smarter" simulations, even the overall ones (changing everybody). Group dynamics is a hard problem, and simulations such as that of Watts and Dodds do not model it, leading to global dynamics only. We show that group dynamics leads to different behaviors at the global level.

Our middle class is producing 40% of all communications, almost as much as all of the "poor" and twice as much as all the "elite" classes combined.

We develop several novel metrics for our class structure –

- *starrank* relates a user to his audience, either mentioners or repliers
- interclass communication metrics reveal whether users in some class prefer talking to the higher, lower, or same class users as themselves
- "skew" is a non-parametric characteristic of a "politician-like" behavior, checking whether a user rewards his higher contributors (giving him more tweets) by returning proportionally more replies

Applying these metrics to our hierarchies we found distinct differences in behavior patterns. The "upper middle class," the 100,000 users above the middle class, have the highest *starrank* by mentions, commanding the highest gap between their rank and the average rank of their mentioners. Skew correlates fairly well overall with the social capital, but it correlates differently by class, confirming that the behavior of classes is in fact quite different.

The classes in the hierarchies of the smarter simulations show consistency from day to day with low churn in most classes and no significant changes when fed with more and more reality for starting conditions.

We fitted regression tree models to see which behaviors in the simulations match various classes more closely and we found significant differences in these models. For instance, the top class is only affected by the starting conditions length and then by whether local utility optimization is used, while the middle class favors social-capital based both global and then local friends-of-friends attachment.

Observing the behavior of different classes as matched by appropriate simulations, we conclude that our hierarchy reflects a real segmentation of user behaviors in which influencers aggregate in certain higher classes. Our reciprocal social capital metric defines a middle class which consistently matches the behavior of the real middle class.

We believe our models show a real hierarchy of effectiveness in conversation with distinct behaviors used to achieve and retain it. These results confirm Katz-Lazarsfeld [38] models rather than Watts-Dodds [65] models. We also suggest that the middle class is the "real influencer" in terms of the effects this class has on the overall communication dynamics, and we propose that increasing one's standing with the middle class is a way to get reliable and stable influence in social media networks such as Twitter.

Chapter 2

Introduction

On March 25th, 2011, "The Fifth Down," an NFL blog of the New York Times, opened as follows [61].

> An independent research company called Twitalizer ranked Bengals wide receiver Chad Ochocinco as the second-most influential personality on Twitter, according to David Leonhardt's article on The 6th Floor blog on Thursday.
>
> Ochocinco ranks behind the Brazilian comedian Rafinha Bastos. He ranks well ahead of President Obama, Justin Bieber, Oprah Winfrey, Sarah Palin, and yes, Charlie Sheen. Twitalizer doesn't just track followers (Lada Gaga and Bieber have the most), but "mentions" and other criteria that determine who is making real contributions to the discussion.
>
> Leonhardt's article barely mentions Johnson — he appears in the top-ten list — but the football world quickly picked up the story of his influence. Let's face it, we do not have much else to do. What makes Johnson more influential than, say, Terrell Owens or Aaron Rodgers, let alone Bieber or Palin? Take it from a longtime Chad Johnson/Ochocinco psychoanalyst: the recipe for Internet domination is simple.

Written off season, during an NFL lockout, this column indicates the reach of social networks, their mainstream appeal, and the fact that many people care about influence in such networks.

How did we end up here? What can it possibly mean to "have influence" in a "social network"? Did Chad Ochocinco get to the top by accident, or does he deserve to be there? These are the key themes of this thesis.

2.1 Overview

Modern human society is first and foremost a society of the mind — a social mind, a new collective ecosystem self-organizing and communicating via the Internet, as well as comprehending itself and organizing various collective actions. The thought becomes action after reaching a critical mass of actors. A concept which becomes action goes through a lifecycle of invention, being noticed by the influencers, spreading, taking hold, and crossing some sort of a phase transition. A telling feature of any successful idea is its transition from individual to group to global level in the network. Determining these levels and detecting how people and ideas relate to them is key to understanding network dynamics.

Spreading news, knowledge and ideas, discussing them, and forming groups around them has always been the essence of culture. Political thinkers and rules, artists and philosophers are examples of the original social networks which circulated relevant knowledge and made a discourse out of it.

In a setup where concepts are tangible as reproducible units and can be retranslated, they take on qualities of currency. For instance, the concepts underlying the backbone of the Internet itself relate to code. In recent years we have seen a dramatic rise in social coding via social code-sharing networks such as *github*, *bitbucket.com*, *_patch-tag.com*, etc. Code ideas are often expressed as working snippets (*gists*), and contain URLs to use and run right away.

When ideas' traction is quantifiable and idea exchanges lead to measurable change in the rankings of both actors and goods being exchanged, an economy inevitably evolves. We call it the *Mind Economy*. This thesis is a detailed examination of the mind economy of communications in social networks.

Our Mind Economy study focuses on the fundamental questions of communication on a network, the goals of each actor and group of actors, and understanding the nature of influence. Simply put, we want to know

- why do people twitter?
- who has influence on Twitter?
- is there a social capital formulation which captures most of people's utilities?

- can we characterize people by their influence and the types of social behaviors related to attaining it?

Since online social networks are driven by software, strongly depend on the algorithmic underpinnings, and produce large volumes of data, a computational approach is required for large-scale data mining of their mind economies.

Accidental Influentials

In their seminal 1955 book "Personal Influence," Katz and Lazarsfeld [38] proposed the concept of *influentials* in social networks, certain key people who propagate and filter media streams in their communities. Watts and Dodds, in their own 2008 paper [65], commonly referred to as "Accidental Influentials," showed that for a diffusion model with cascades, it is not necessary to have influentials in order to excite an information cascade in a typical network. A Twitter study co-authored by Watts [4] shows that influentials are not propagating URLs much farther than the rest of us. So, are these influentials indeed accidental?

In this thesis we look at a basic form of influence, the effectiveness of a communicator in the social network. We use our own reciprocal social capital as a metric to track how a person prioritizes his or her replies, paying attention to "communication balances" owed and to whom. For instance, we note when somebody answers questions from an interlocutor with a higher social capital faster and more often than requests coming from those below in the capital hierarchy. Any such metric leads to a ranking among the users, from which we build a class system, resembling a human hierarchy of wealth.

In the US the top 1% of population by wealth owns 35% of all of the nation's net worth, while the top 10% is left with only 63% of it. The second 10% of the population by wealth comprises 12% of net worth, and the bottom 80% collectively hold only 15% of total net worth. Figure 2.1 is taken from G. William Domhoff, the author of "Who Rules America" [22]. His web site, `whorulesamerica.net` [21], contains many instructive graphs segmenting the wealth hierarchy in the US and other countries. The segmentation is consistently following the same power law, with different exponents. Figure 2.2 shows that financial wealth, defined as the net worth minus owner-occupied real estate, comprises the bulk of the middle class equity.

Such division causes emotional reactions. Bob Herbert, who was a columnist for the New York Times for almost 18 years, cites these statistics in his March 26, 2011 column "Losing Our Way":

> There is plenty of economic activity in the U.S., and plenty of wealth. But like greedy

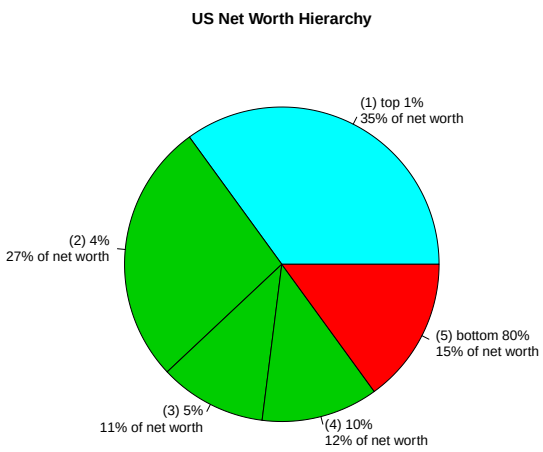

Figure 2.1: A small percentage owns the majority of US net worth

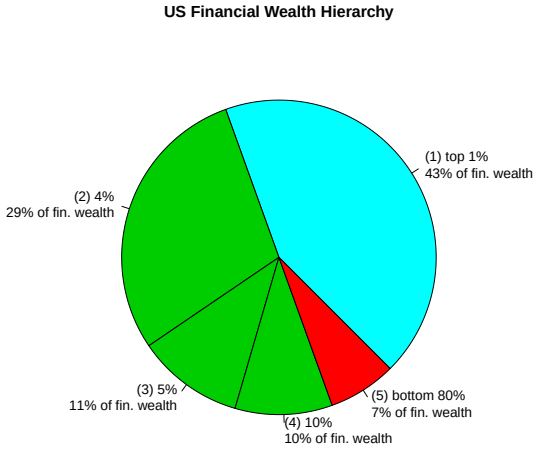

Figure 2.2: A small percentage owns the majority of US financial wealth, following the Zipf law.

> children, the folks at the top are seizing virtually all the marbles. Income and wealth inequality in the U.S. have reached stages that would make the third world blush. As the Economic Policy Institute has reported, the richest 10 percent of Americans received an unconscionable 100 percent of the average income growth in the years 2000 to 2007, the most recent extended period of economic expansion. Americans behave as if this is somehow normal or acceptable. It shouldn't be, and didn't used to be. Through much of the post-World War II era, income distribution was far more equitable, with the top 10 percent of families accounting for just a third of average income growth, and the bottom 90 percent receiving two-thirds. That seems like ancient history now. The current maldistribution of wealth is also scandalous. In 2009, the richest 5 percent claimed 63.5 percent of the nation's wealth. The overwhelming majority, the bottom 80 percent, collectively held just 12.8 percent.

(Interestingly, Herbert ends his column (as this text and as a series) to address the imbalance: "This is my last column for The New York Times after an exhilarating, nearly 18-year run. I'm off to write a book and expand my efforts on behalf of working people, the poor and others who are struggling in

our society.")

While policy opinions differ on what to do with such distributions, there are many such hierarchies in nature and societies which are caused by many factors. In Economics such distributions are commonplace, and we use a Zipfian hierarchy when modeling the distribution of social capital wealth.

In order to see whether the influence hierarchy is random or not, we simulate many parallel worlds where users talk to each other based on various global and local optimization criteria. Some of these criteria are simple, such as attaching randomly, or proportionally to the global in-degrees; some are complex, such as attaching to a friend of a friend proportionally to the number of their own mentions or their social capital; or, ultimately, optimizing your local capital gain, using the same reward function which is the foundation of the final ranking. We either simulate the new worlds from scratch, or seed them with a few weeks of the real dynamic graph. In all cases, we preserve the order of users joining the network and their out-degrees for each day. A full quantitative definition of our Reciprocal Social Capital is introduced in 6. We describe our simulation process in detail and define the the concepts of parallel worlds, local optimization, capital gain, reality seeding, and other processes used to build the simulations in Chapter 7.

Having built the dynamic hierarchy of influencers we compare the "winners" (highest-ranking communicators) in the corresponding positions of a simulation with the real ones, and also across the worlds. We look at the staying power in the same class bucket across days within the same world, and overlap of the respective buckets across different instances of the same world type, seeded with different lengths of reality to obtain week-shifted worlds, or compare winners across different worlds altogether.

We observe that the more intelligent a simulation is the higher is its staying power (the winners persist in their leadership positions longer), thus showing that the influencers – those in the higher classes – are not random in their own worlds. The overlap of the simulated influencers with the real-world ones is minimal, but it also increases with more intelligent attachment strategies.

We tested our assumptions by altering the formula of our reciprocal social capital (RSC) in several ways. First, instead of rewarding reciprocity, we reversed the sign on its rewards, and observed that

the precision falls for all classes when the capital is computed universally (for all classes at the same time, i.e. unbucketed). We call this social capital version NSC (Negative Social Capital). Note that the precision is comparing how well we capture the class hierarchy of the original Twitter graph, where the links are unaltered – yet the hierarchy itself is computed for a certain version of the social capital, here it is the original RSC.

We compute NSC in a the same bucketed way as RSC (placing people in exponentially-sized buckets after sorting them by the appropriate capital). When, after that, we model only the original middle class bucket (the 1,000,000 people), and leave the dynamics of the other classes intact, we observe an interesting effect that the precision of the four top-most ("elite") classes drops significantly, while we get a slightly better capture of the three lower classes, including the middle class. This holds only when we use the original version of RSC/NSC where we count all incoming mentions separately and sum them up to use as their own norm. The incoming-only (non-reciprocal) mentions may dominate in such a setting, since generally the reciprocal component is much lower than the non-reciprocal one. This is especially pronounced in lower classes, who appear to be less attentive in keeping the balances even.

For these cases, we considered another alteration of the RSC formula, whereby we normalize the non-reciprocal component of the return, so as to make it have the same normalized scale as the reciprocal component, RSC-IN (In-Normalized). We also compute a version in which we ignore the nonreciprocal inputs altogether (RSC-NO). We observe that a bucketed computation for NRSC-IN-B delivers a uniformly poorer prediction against the normalized NRSC-B, as opposed to the split in the unnormalized case (predicting some classes better for one vs. the other method), while RSC-NO-B fares uniformly poorer, often significantly so, than RSC. Thus negating reciprocity itself disturbs the hierarchy significantly up to and including the middle class. Interestingly, RSC-IN-B performs poorly compared to RSC-B in all classes when only the middle class is modeled, showing that capturing the general mentions is still important for the middle class. We also considered another, slighter variation of the original formula, in which we eliminate the directed multiplier term, using only the total balance. For the original RSC, we get slightly better capture when computed across all classes. When modeling the middle

class alone, we lose precision for all classes, significantly so the first five classes. Similarly, for IN-normalized general mentions equalized to return mentions, we get a better capture of all classes when modeled together, yet a general precision loss when modeling the middle class separately, especially seen in all but the poorest classes.

From these variations we conclude that both reciprocity and directionality are used with the expected effect in the original RSC. While the exact weighting and normalization of the components can be tuned to serve specific ranking goals depending on the behaviors of interest, the overall model reflects these behaviors quite reliably.

We conclude that the influentials are not random in their own worlds. The influentials correspond best to what the attachment strategies are accordingly to the common rules in these worlds, but they are also those graph nodes who were well-positioned in their social network so as to take advantage of these rules. For instance, a global uniform attachment leads to a stable celebrity bucket (the top class). This bucket remains quite stable even though the rest of the buckets are changing at a faster rate. The celebrity classes stay very stable during the lifespan of their world, but are comprised of different sets of people across various independent simulations of the uniform world. At the same time, top buckets of smarter simulations show overlaps higher than 50%, more than overlap of any other two simulations, showing that their success under the rules of those worlds is repeatable even when many edges are rearranged, and thus their success is not random.

Also, the overlap of the winners (top classes) of all non-bucketed simulations with the real winners is small, proving that the winners in the real world are lucky to have the set of rules which distinguished them. If the accepted attachment strategies were slightly different, others could easily replace the current winners in some of the top buckets, and stay there under wide ranges of new conditions and their combinations. In that sense, if we agree that our world with its own communication rules is itself random in a series of many similar ones where such rules can be slightly altered, the influencers might indeed be accidental. The actual celebrities tend to be projected into the social network from the outside world. While celebrities may not be easy to approximate via any of our strategies, we simulate the middle class surprisingly well.

We show that simply prescribing a desirable distribution of influence from the real world, as we do with *creps* simulations, leads to a stable and reproducible middle class. It shows that order leads to a stable mind economy. We also show how bucketed simulations allow tracking of how composite strategies apply to separate classes, or any combination of classes, either preserved or simulated.

Additionally, the middle and upper middle classes of social networks are less random across the more intelligent simulations than in the more random, simpler simulations. The middle class of social networks is the main discovery of this thesis. It carries 40% of all communications, while comprising (roughly) only 20-25% of population. These users are the actual majority influencers, and they got where they are due to a diligent attention to communication, its priorities, dynamics, and energy demands. In that sense, we conclude that success is earned, at least for the members of the middle class.

Finally, we conclude the middle class is the real influencer, and it is anything but accidental. For the kinds of effects which matter, these are the real non-accidental influentials we should care about. We believe that the middle class of the social networks is the most "rational" one, following the expectations of human communication most closely and so most suitable for modeling via our social capital.

Figure 2.3 shows how several groups of simulations, loosely corresponding to either Katz-Lazarsfeld or Watts-Dodds models relate to reality and each other. Our social-capital based models, taking into account many features of influencers as effective communicators, are close to each other and reality, as compared to many simpler simulations based on either uniform or mentions-based attachment, as found in the simulations of Watts-Dodds.

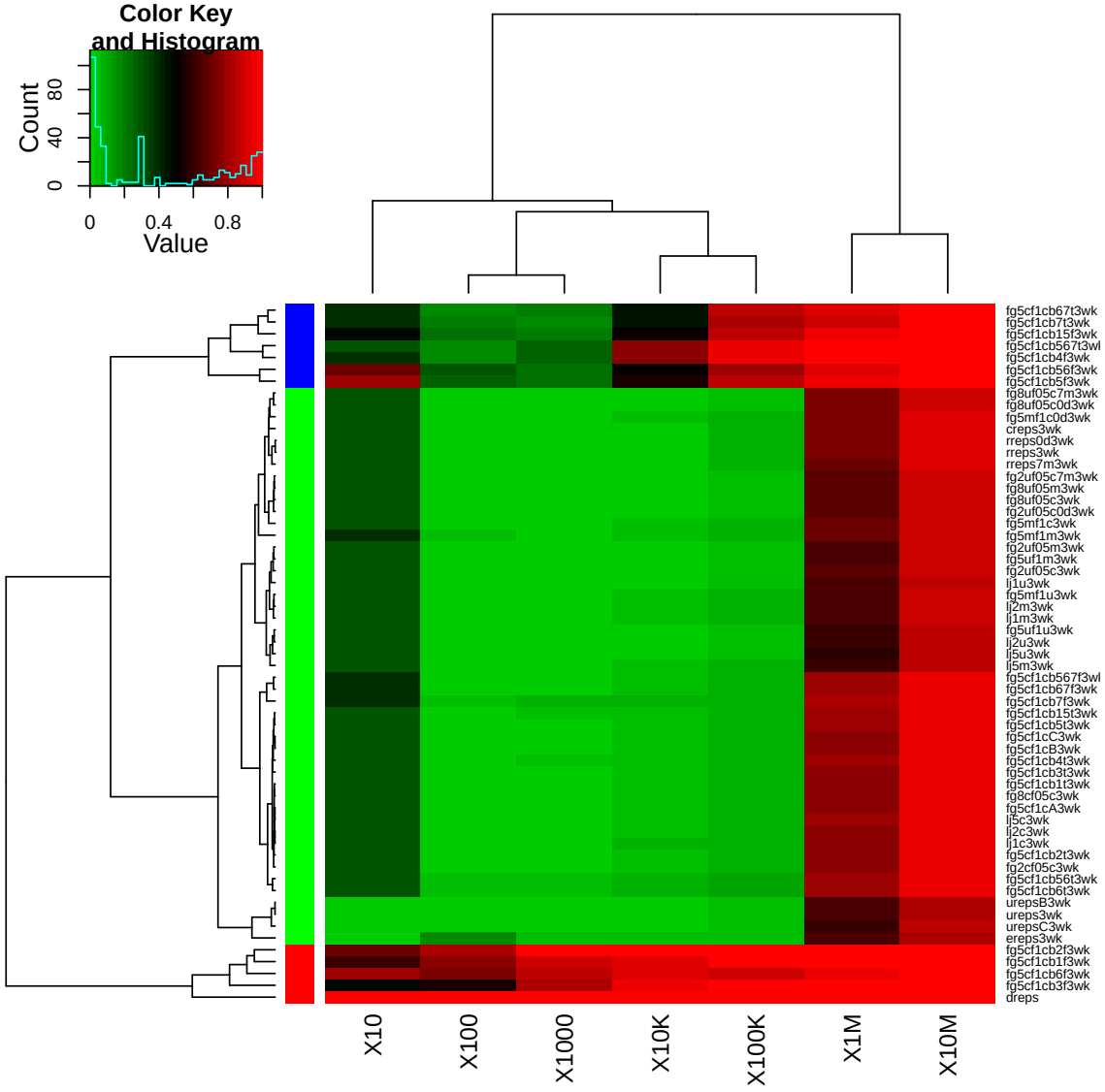

Figure 2.3: Clustering of all simulations seeded by 3 weeks of reality, by overlap with reality. *Dreps*, the reality itself, ends up closer to our most intelligent capital-based worlds (red), including the simulated middle class. Simpler networks, including those using only global attachment strategies, end up in a more distant cluster.

2.2 History

Social networks exist in some form for as long as human society itself. What made them subject to our computer treatment is their current online incarnation, whereby their operation is code-based, algorithmically guided, and their data is available in bulk for serious data mining. History of social networks is rich and eventful, just like human history itself which can be considered as an evolution of the kinds of social networks humans organize into, such as tribes, businesses, guilds, classes, and nations.

The first recorded blogger, in our analysis, is Cicero. While in Rome, engrossed in the politics of the failing Republic, Cicero used to write almost daily to his friend Atticus who was in Athens [11]. According to some accounts, a messenger from Atticus picked up a new letter from Cicero and delivered it to Athens, where Atticus had a stable of scribes making copies. The copies were delivered to many prominent Romans all over the Empire for a subscription fee. In this way Atticus enabled Cicero to have an income from his writing talents – and also spread Cicero's influence [56]. We see in Cicero's example many key features of real influencer in a social network.

Thanks to Atticus, we also have many letters which Cicero wrote in reply, and in order to maintain his vast social network. Every effective blogger from our online social networks has some, rarely all, qualities which Cicero demonstrates in his own correspondence:

1. a vast network of contacts
2. engagement with topics of the day and issues of public interest
3. high-intensity writing, frequent and consistent in time
4. engagement with the readers
5. writing for well-defined communities, the political class and the philosophers
6. careful attention to the balance of communications

While we don't have enough data volume to quantify these points for Cicero, we can quantify most of them in online social networks. Careful attention to the balance of communications (point 6) corresponds naturally to our reciprocal social capital definition (see Section 6).

The first online social network of notice, which is still going strong today, is LiveJournal[1] (LJ), started by Brad Fitzpatrick in 1999 as a way to for friends to share their dairies and notes online. It pioneered most of the features of online networks we now take for granted, such as

- an account for each user with posts along the timeline
- a mechanism to follow friends (their updates can be reviewed as a news stream). This predates RSS and is a key feature of LJ.
- explicit communities, where you can create or join a group and then post to it. Communities' membership and posts can be controlled by moderators, e.g. be by invitation only, or open for all.
- Anybody can read anybody else's journal except for the "friends-only" or private posts.

These features are present, in some form, in all modern online social networks. Communities may not be explicit, such as Twitter, but then some way to self-organize is always found, such as users adding metadata in the form of *hashtags* (like `#clojure` for the community of Clojure fans).

Facebook and Twitter follow up on the basic framework of LiveJournal. Facebook is generally closed, ensuring the network grows by expressly approved links. Photo sharing is integrated into Facebook making it the largest photo-sharing site in the world. The closed nature of Facebook ecosystem necessitates in-network applications, using the platform API. Twitter limits "posts," to 140 character "tweets." URL shortening and media sharing are supported by a plethora of external services. Other networks worth mentioning are LinkedIn, Classmates, and various national versions of the above. To take one country for example, in Russia there exists odnoklassniki.ru (classmates, similar to classmates.com[2]),

[1] http://livejournal.com/

[2] http://classmates.com

and vkontakte.ru (“in touch”, analogous to LinkedIn[3].

[3]http://linkedin.com/

2.3 Twitter

Twitter is the key social network of our time, at least as far as the scientific community is concerned. Started in 2006, Twitter currently has about a hundred million users worldwide. In 2009, when we started getting the *gardenhose* (a fraction of all tweets, see Section 8.2), we were receiving about two million tweets a day; a year later, the gardenhose was bringing more than five million daily. As opposed to Facebook, tweets are all is the open. (Those using private tweets are a small minority with specific uses, such as explicit chat.) Anybody can tweet to anybody else, given they know their Twitter nickname, such as `@john`. The tweet will appear in `@john`'s stream when he checks his Twitter account. A communication network of a Twitter user is constrained only by his or her own abilities.

Originally designed to be ready for transmission via SMS, each "tweet" is limited to no more than 140 characters in length. A significant portion of all tweets contain a URL, making Twitter the largest ever machine for URL transmission. Due to the character limit, various "URL shortening" services exist, creating and maintaining mappings of long actual URLs to shorter, unique ones. Some shortening services, such as bit.ly[4], use the mapping to track clicks and URL percolation.

Multi-stage processes are possible where in-network and external influence alternate. Twitter's always-connected nature makes it an ideal vehicle for URL percolation. Trending often spreads across the Internet subcultures, as evidenced by many a YouTube phenomena gaining their momentum on Twitter. Justin Bieber, whose fans' mind economy we study in 5.3, started as a YouTube boy-singer sensation, was discovered by a talent scouting agency and noticed by Usher, the hip-hop star, eventually sang for `@barackobama` and broke into mainstream via the People magazine. Each stage in life of Justin Bieber's ecosystem, or "beliebers", is reflected on Twitter, which has made Bieber's celebrity via the *trending topics* mechanism — which in turn was understood and collectively manipulated by the ever-growing Bieber ecosystem.

Twitter maintains a list of top ten "trending topics," computed by a proprietary algorithm, clearly counting the frequency of mentions with some heavy timing preference towards the present. There are

[4]http://bit.ly

also local trends per geographic area.

Twitter translates all kinds of trends – both in-network, such as *memes*, and projections of the real world, or outside reality, such as `#michaeljackson`. The latter often represent major news and may be distinguished by multiple, almost simultaneous, injection points. It can be an interesting project to use the injection dynamics for differentiating the externally induced trends, and perhaps also their strength and globality.

As Mikhail Gronas, a researcher of Russian folklore and global online discourse, noted: "Twitter is a global megaphone, or, as we observe, a global echo chamber, which amplifies trending echoes from the outside world, but also has power to process and project them back into the world as Twitter itself grows in influence" [31].

When studying influence, it's important to distinguish between such global mirror and global megaphone behaviors of the network. It's hard to control for all the externalities on an injected concept.

We collaborate with the Web Ecology Project[5] on online memetics studies. While a separate and rapidly evolving area with its own conference ROFLcon[6] started by Tim Hwang, online memetics, overall, describes just one of the complex processes transpiring over Twitter, overlapping with many other complex percolation processes. *Memes* are real in-network phenomena, often represented by a YouTube clip of quirky, catchy, or paradoxical nature. They are interesting because of their percolation patterns and staying power. Tracking memes requires a different approach than identifying opinion change based on evidence accumulation [60] — they are short-lived and viral, reflecting the fast pace of online life.

Twitter trends often get their names and self-organize via *hashtags*. Users spontaneously or consciously tag their tweets with a `#phraseprecededbyhashsign`, allowing searchers to identify, guess and extract thematic *ad hoc* communities. It's fascinating to watch how a hashtag spreads through the system virally. Many communities have an established hashtag or set of hashtags, such as `#clojure`

[5] http://webecologyproject.org

[6] http://ROFLcon.org

or `#scala`. Internet-rich conferences such as Clojure Conj[7] have "official" hashtags of their own, in this case `#clojureconj`. It should be kept in mind that any hashtags can be latched on by spammers to promote their tweets. Hence, content-based community identification is still preferable in addition to hashtags.

In *Twitterverse* (the Twitter universe, including Twitter and the ecosystem of startups around it) there is a service to overcome any limitation and complement any feature, for example, to tweet with more than 140 characters, there is Twit Longer[8], for self-organizing into "tribes", there are Twibes[9], etc.

[7] http://clojure-conj.org

[8] http://www.twitlonger.com/

[9] http://www.twibes.com/

2.4 Rankings

Whenever a social system self-organizes into any kind of purposeful structure, including online social networks, hierarchies take shape. Since human relationships are (literally) inherently hierarchical (from the family mechanism and up), each pairwise interaction may say something about the status of the participants. Similarly, any group solving any kind of problems develops leadership, while "mere" discussion groups may stay loosely organized and relatively leaderless [64].

It is also in human nature to identify those "at the top" for all kinds of competitive criteria. Since Twitter allows one to "follow" others and shows the counter of your followers, the number of followers has emerged as a first proxy for Twitter influence.

Initial attention to the number of followers lead to early wars to be on top by having the most of them, notably the famous battle of Ashton Kutcher vs. CNN. Playing on the quirky, meme-loving personal dynamics of the early twitterers, Ashton, a (celebrity) person, managed to accumulate more followers than CNN, a global news corporation. As of the time of this writing (October 2010), `@aplusk` has roughly 6 M (million) followers, `@cnn` has 1.35 M, and `@cnnbrk` (CNN breaking news) has 3.5 M followers.

The number of followers is certainly a measure of popularity, often representing real-world celebrity – such as `@barackobama`, who has 5.7 M followers.

However, this number is not the most useful in many cases, as it reflects the global interest in a person who may not be really active on Twitter itself. While President Obama has millions of Twitter followers, he had practically no participation in any discussions conducted through Twitter during the time span of our research. Consequently, his Twitter account is at the bottom of our conversational metrics. Our reciprocal social capital is especially clear on this, as it rewards active and persistent involvement in the day to day business of communication on Twitter as if it were politics – with careful attention to individual balances of addresses and replies.

Simply maximizing the number of followers as a criteria of influence had lead to a plethora of spam-bots who first follow each other and then you, expecting a "follow back" from those accounts set to

auto-follow. An active market of spamming software has developed to follow and unfollow people *en masse*, as well as to buy (robo-)followers outright. For these reasons, we prefer not to use on the number of followers as the most important feature of a node on Twitter.

Some of the sites in Twitter ecosystem that compute various ratings are listed in Twitter ranking services (Table 2.1).

Table 2.1: Twitter ranking services

URL	Ranking kinds
Twitturly[g]	ranking of URLs in tweets
Crowdeye[h]	Crowdeye Rank, numeric influence
Klout[i]	rankings of influencers in subject areas
Twitaholic[j]	top rankings by followers
WeFollow[k]	rankings by followers per category
Twinfluence[l]	includes Reach, Velocity, Social Capital

[a] http://twitturly.com
[b] http://crowdeye.com
[c] http://klout.com
[d] http://twitaholic.com
[e] http://wefollow.com
[f] http://twinfluence.com
[g] http://twitturly.com
[h] http://crowdeye.com
[i] http://klout.com
[j] http://twitaholic.com
[k] http://wefollow.com
[l] http://twinfluence.com

When reviewing various Twitter ranking services, one cannot help notice that for the majority of the proprietary rankings it is unclear which factors figure in which, and with what weights. In our discussions with various startups in the area we proposed Open Influence (OI) standards and benchmarks. They allow us to

- compare rankings from different services
- explain how a particular ranking was obtained
- confirm, for the users of rankings, e.g. marketers, that their budgeting is justified if based on such OI-compliant rankings
- enable competitions to improve the rankings with objective performance metrics for comparison

2.5 Communication Networks

In this thesis we focus on direct communications from one person to another or possibly to a specific target group enumerated by name. While most *Twitterverse* services focus on the followers and the number of followers a Twitterer has, we look at the actual dialogues taking place in public tweets. Conversation is a basic unit of social interaction. Conversations are fundamental and unfold via the same scenarios regardless of the medium through which they are conducted. From face to face, to paper mail, to phone, email, instant messaging, and now to online social networks, conversations follow the same essential patterns, and networks arising from them demonstrate the same inherent qualities. Twitter conversations may form regular chains or be episodic. They can be conducted via *ad hoc* tweets or systematic *replies*, with the latter ones memorizing the ID of the original tweet.

Algorithmically, our focus on conversations is realized via building and studying the *replier graph*. For any user A who has ever mentioned another user B, the replier graph contains all of A's messages to B (which may be either "for" or "about" B), segmented by days. A note on terminology: in a tweet from A mentioning `@B`, A is the mentioner of B, and B is the replier of A. Time segmentation helps us study how this graph evolves.

For every day in the study, we compute a pagerank-type score and a *D-Rank*, a dynamic function of the pagerank, for each user, together with a series of features such as the number of mentions a user gives or receives. The daily-versioned features enable exploratory data analysis of the conversational dynamics by looking at the relative decline or growth in specific features for each user every day, separately or relative to others. For instance, we find the longest periods of growth in the number of times a user A is mentioned by other users on a day d, $m = |M(A, d)|$, over a contiguous period of days, and also compute its acceleration over that period, dm/dt. Those accelerating the most, or sustaining the longest growth period, or both, are considered influential and are worth closer modeling.

2.6 Network Features

Most of the work in mining Twitter is exploratory in nature and reflects the desire to discover "what's going on," a fundamental question we set out to tackle from various angles. Twitter, of all online media, is the closest to a social happening and this is how we approach it, as a series of subsequent iterations from individual, to group, to global, and then back to individual with global reference of reciprocal social capital. A similar path is followed by other researchers.

Our friends from the Web Ecology Project look at such features as

- memes, such as Old Spice ads and their audience make up
- Events such as `#IranElections` [27], `#WorldCup`, etc.
- methods of simulating twitter audience
- detecting sentiment, language, gender.

Due to the sheer volume of tweets and Twitterers, users tend to self-organize into communities as followers by using hash tags and conversations. A variety of algorithms for community identification exists for online networks such as the Web. Most of these algorithms are using link structure only, and the majority identify only non-overlapping communities.

Flake et al. [24] employ a recursive definition, that a community is a cluster with more inside links than outside, and use network flow to segment the whole network at once. Baumes, Goldberg at al. [6, 29] allow for overlapping communities but they rely on the communication graph structure only. Cazabet et al. [13] use network dynamics to detect strongly overlapping communities using the order in which edges are added, relying purely on the graph structure.

Although these approaches are valid and produce meaningful communities, we prefer to mimic actual group-finding process humans employ when trying to find a good fit for themselves in a social network.

2.7 *EDA* for Social Networks

We approach the problem of browsing a social network, similar to data sampling, from the point of view of a typical user new to the network who wishes to find the most interesting set of people discussing the topics most interesting to the user. It turns out that "similar" people and topics tend to group together, and finding topics helps identifying the people (and vice versa). Furthermore, if people are "about" some topics, and we treat their network nodes as "buckets of ideas," we find that those buckets usually hold a fixed set of closely related ideas, as shown in public behavior. We model these assumptions in our *topical communities* browsing process.

First, we index all of the text in all of our tweets of interest as a single corpus and build the communication graph from them. To find a topical community, we come up with a specific form of a topic of interest, expressed as a string of text. Using our index, we find all of the pairs of users who exchanged tweets matching the topic, in at least one or both directions. We sort all such communication pairs in the decreasing order of their total number of exchanges, again counting either all or only the topical exchanges, obtaining a sorted list of pairs. At the top of the list there's a *pair of repliers* A, B. This pair will be the seed of our topical community.

Once the seed is obtained, we grow the community by inducting each new member C, one by one, such that he has "parents" A, B who are already in the community and exchanged the topical tweets (textually matching the topic) with both of them. We call it *a triangle rule*. Our topical communities turn out to be coherent and the process converges very quickly.

Given the community, we can analyze its combined text. We do it by looking for *Statistically Improbable Phrases* (SIPs) as representative of the communal discourse. We also introduce the notion of *Fringe* as those members who have just one friend in the community. Again, treating the Fringe's textual output as a whole, we can identify its own SIPs. Those sets of SIPs can then become starting points for the next iteration of network browsing.

2.8 Dynamic Graph Analysis

If social network exploration is similar to statistical data sampling, then our dynamic data analysis is akin to, the *Exploratory Data Analysis* (EDA) of Tukey [63]. Topical communities allow for an iterative, localized process to pivot and browse. A global index of tweet texts can be compiled for seeding topical repliers, in turn used for local community building. Together, this generative process provides a set of tools to identify interesting people focused on related topics across the whole network.

We're interested in people who are influential. We similarly may look for those whose influence is declining. Since we work with the communication graph, where `@alice` replies to, or mentions, `@bob`, we can start by looking for such `@bobs` who have the most mentioners. Ranking by pure mentions does not reflect those important nodes who are mentioned by just a few subnodes but with a lot of mentioners of their own, i.e. by a few other important nodes. The latter clearly calls for a *PageRank*-like measure.

The communication graph is dynamic, changing every day, whereas PageRank is a stationary distribution on a given static graph. We solve this problem by introducing *D-Rank*, a measure which replaces the PageRank of a node in a given daily snapshot by its rank-order position in the sorted PageRank list for that day normalized by the list length.

D-Rank then becomes a foundation of *StarRank*, which reflects an individual's importance relative to his audience. Given the star configuration of a node and its repliers we first compute D_A, the average D-Rank of the audience weighted by the number of replies each mentioner has contributed to the "star" We then take the ratio of the star's own D-Rank over D_A. The prefix "star" in *StarRank* has two meanings – the configuration used to compute it, as well as the fact that the ratio will be high for real celebrities, or "stars."

D-Rank and StarRank are designed to be comparable across days so we plot them day by day and then find interesting patterns in the resulting time series. We look specifically for

- The people who have the longest monotonically increasing or decreasing consecutive runs of D-Rank or StarRank

- —""— for nonicreasing/nondecreasing
- —""—, but looking at the acceleration of the value at hand, i.e. its rational derivative

Our sorting, filtering, and ranking techniques amount to a simple data model which is at the heart of any EDA. This is the approach Prof. Mitch Marcus of UPenn used in his very successful statistical NLP course since the 1990s, piping together Unix tools like `cat | sort | uniq | top` to perform NLP tasks from the command line. And for many social networking tasks it remains the most effective approach, especially given the volume of data we work with.

Each day's snapshot can be handled separately, and its PageRank computed. PageRank is the least parallelizable part here as it reflects the whole graph structure. After that, however, its derivatives D-Rank and StarRank can be computed in parallel for each node, which our *Clojure*-based implementation 8.4 fully exploits, with nearly linear speedup from a single letter `p`, making a parallel `pmap` out of a sequential `map`.

Using our simple ranking models we find very interesting, quite unexpected sets of influential users, on which we report in the 5.2 and 5.3 sections. We discover the whole new "Justin Bieber ecosystem" with its attending *Mind Economy*, and then see the presence of similar forms everywhere.

2.9 Mind Economy

We want to move even deeper with our global metrics advancing our dynamic graph analysis while reflecting the community structure from the exploration techniques described above. We also want to preserve the inherent temporality of our data set.

Using the *Mind Economy* insights from our analysis of influence we strive to define a quantitative *social capital* metric which would revitalize the term from the various fuzzy uses it acquired and bootstrap a meaningful virtual economy on top of the real data evolution we observe. We note that *capital* can not be just a random count, such as a "number of friends on a program committee" [49]. Capital is made through a certain wealth creation process, and it may be exchanged through transactions wherein meaningful adjustment occurs, dependent on the parties' starting amount of capital. All these requirements lead to our iteratively defined *Reciprocal Social Capital* (RSC). We literally "replay" the whole original world of Twitter by following all daily transactions in order and adjusting each node's capital according to our RSC definition from day to day. In that definition, we focus on reciprocity of communication rewarding those who reply to their mentioners, and whose repliers eventually mention them in return. We model this balance-conscious system on the village social capital of Tuscany, which has lead to especially cohesive, thriving communities in real life, enduring through history to this day [28].

The result of this analysis is a revelatory picture of a robust *middle class* of social network users, who conduct conversations that concern and reinforce their long-term interests and businesses, and shoulder the bulk of purpose-driven Twitter communication.

2.10 Roadmap

In the rest of this thesis we elaborate all of the above points.

Chapter 4 goes in detail regarding our textual network browsing approach, the EDA of social networks.

Chapter 5 reviews our dynamic graph analysis, new metrics, resulting influence models, and showcases some of the influence findings

Chapter 6 describes our re-appropriation of the notion of social capital from the soft sciences, with a robust, time sensitive, economic definition, and introduces a mind economy built on our definition of reciprocal social capital. In this chapter, we close with the discovery of the middle class of social networks.

Chapter 7 contains a detailed methodology of simulating social networks mixed with reality and segmenting various attachment strategies that contribute to one's position in the social capital hierarchy, which allow one to compare the effects of those strategies in various parallel futures. The overlap of such simulations with reality can specifically be attributed to the starting conditions (the timing of joining the network and the original connections) and participants' well-defined behaviors, and addresses the question of the effect of "accidental influentials"

Chapter 8 highlights the computational infrastructure we created in order to overcome the challenges that the sheer volume of data posed. We direct the reader to the open source projects, themselves hosted on the social coding hub, developed through the Internet Relay Chat (IRC) and Twitter collaboration with many of the same people who built many of the collaborative and communication-based social media platforms that we use. Thus research and development come the full circle.

Chapter 3

Previous Work

3.1 Influence

Influence in social networks is an area approached by many disciplines, yet influence is most universally understood as an ability to affect a diffusion process, such as pushing a *meme* and/or an URL toward world fame "on teh internets" (sic).

Kempe, Kleinberg, et al. [40] consider the question of maximizing influence diffusion by selecting an initial set of k users such that the final affected audience will be the largest. They look at two established models of influence processes:

- Linear Threshold model, pioneered by Granovetter and Schelling [30, 58]
- Independent Cascade model, analyzed by Goldenberg, Libai, and Muller [34, 35]

Both models deal with a simplified setup in which there are two states, a and b, and switching states depends on the state of the neighbors talking to a node. In the Linear Threshold model, everybody has a threshold, and if the sum of incoming edges from the activated neighbors exceeds the threshold, a switch occurs. In the Independent Cascade model such a switch is governed by a table of pairwise conditional probabilities. Kempe et al. [39] show that they can achieve at least 2/3 of the ideal maximal influence set by following a greedy strategy of adding a node with maximum influence at each stage.

When tracking influence Cosley, Kleinberg, et al. [17] look at continuous sequential data versus snapshots. They consider two of the most fundamental definitions of influence, one based on a small set of "snapshot" observations of a social network, and the other based on detailed temporal dynamics. The former is particularly useful because large-scale social network data sets are often available only in snapshots or crawls. The latter, however, provides a more detailed process model of how influence spreads. Cosley, Kleinberg, et al. study the relationship between these two ways of measuring influence, in particular, they establish how to infer the more detailed temporal measure from the more readily observable snapshot measure. The analysis is validated by using the history of social interactions on Wikipedia, the result of which is the first large-scale study to exhibit a direct relationship between snapshot and temporal models of social influence.

Bakshy, Watts, et al. [4] track diffusion events that took place on the Twitter follower graph over a two month interval in 2009. They found that the largest cascades tend to be generated by users who have been influential in the past and who currently have a large number of followers. They also found that URLs that were rated more interesting and/or elicited more positive feelings by workers on Mechanical Turk were more likely to spread. In spite of these intuitive results they find that predictions of which particular user or URL will generate large cascades are relatively unreliable. They conclude, therefore, that word-of-mouth diffusion can only be harnessed reliably by targeting large numbers of potential influencers, thereby capturing average effects. Finally, they consider a family of hypothetical marketing strategies which are defined by the relative cost of identifying versus compensating potential "influencers." They find that although under some circumstances, the most influential users are also the most cost-effective, under a wide range of plausible assumptions the most cost-effective performance can be realized using several "ordinary influencers," individuals who exert average or even less-than-average influence.

3.2 Sociology

Tarde (1898) famously started to seek *statistique de conversation* as justification and method of understanding the public opinion formation. In the words of Katz,

> Both Tarde (1898) [62] and Habermas (1989) [32] may be said to have theorized a public sphere based on the sequence media-conversation-decision-action. In Tarde's scheme, the media deliver a menu of political issues to the cafés and coffee shops and salons. Discussion of these issues percolate more "considered opinions." These opinions circulate from café to café until they crystallize into Public Opinion, which feeds back to government, the media, and individual decisions. As already noted, it is obvious that the "two-step flow" — media to conversation to opinion — has a major presence in these theories.

George Kingsley Zipf, in his visionary book "Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology" [67], establish many important principles that are used in our work directly, specifically the

- principle of simple utility optimization leading to complex system behaviors
- principle that power laws result from various human distributions (now called the Zipf law)

3.3 Network Structure

Barabási's 1999 *Science* paper "Emergence of Scaling in Random Networks" [5] started a whole area of structural network studies that focus on network generation processes and the resulting behaviors which were found to be dependent on the network structure. His work covers both theoretical properties of scale-free networks and real large-scale networks such as mobility patterns of cell phone users at country scales. It was discovered that most people have a fixed mobility pattern, with a small number of regular locales and patterns of travel between them which can be matched by scaling and rotation.

An important insight of Barabási's work is that specific human behavior patterns lead to identifiable patterns at the group and social level. In the same vein we show how basic elementary strategies of attachment, selecting conversation partners, in communication networks lead to clear stratification of social capital.

Khrabrov, Ungar et al. [46] show that the underlying attachment strategy leads to different network degrading behaviors under random faults or enemy attacks. From the hindsight vantage point of human behavior modeling it is an interesting inversion that shows how different network topologies can react to different external behaviors by degrading either gracefully or abruptly in the face of faults or attacks. If we consider node removal detachment, or negative attachment, we can see how attachment patterns governed by simple strategies, such as detaching the hubs, can lead to long-term patterns in network overall performance as indicated by the diameter of the connected component growing rapidly or slowly. Such effects are true for attachment as well.

Marc Smith and others created NodeXL, a social-networking analysis plugin for Excel, implementing many centrality features in practice. Their NodeXL book [20] shows how communities of researchers and corporations employ network behavior metrics for knowledge discovery and for driving business thus fusing theory and practice.

3.4 Social Capital

Motidyang, in his thesis [53], examines the notion of the social capital in virtual communities, defining it in the following way.

> For more than a decade, researchers use the term to mean the set of trust, institutions, social norms, social networks, and organizations that shape the interactions of actors within a society and that are considered to be useful and assets for communities to prosper both economically and socially. Despite growing popularity of social capital especially, among researchers in the social sciences and the humanities, the concept remains ill-defined and its operation and benefits limited to terrestrial communities. In addition, proponents of social capital often use different approaches to analyze it and each approach has its own limitations.

His thesis examines social capital within the context of technology-mediated communities (also known as virtual communities). It presents a computational model of social capital that serves as basis for understanding, formalizing, computing and discussing social capital. The thesis employs an eclectic set of approaches and procedures to explore, analyze, understand and model social capital (SC) in two types of virtual communities: virtual learning communities (VLCs) and distributed communities of practice (DCoPs).

Key findings from the various studies in Motidyang's thesis indicated that SC is a multi-layered, multivariate, multidimensional, imprecise, and ill-defined construct that has emerged from a diverse set of terminology, but it is still useful as a way to explore and understand social networking issues that may influence our understanding of collaboration and learning in virtual communities. Further, the model predictions and sensitivity analysis suggest variables such as trust, different forms of awareness, social protocols and the type of the virtual community hosting the interactions are all important in discussion of SC in virtual communities. Each variable has different level of sensitivity to and effect on social capital.

The major contributions of the thesis are the detailed exploration of SC in virtual communities and the use of an integrated set of approaches in studying and modeling SC. Further, the Bayesian Belief Network approach that is applied in the thesis may also be applied to modeling similar complex online social systems.

The overview presented in this thesis describes a variety of approaches to, and qualifications of, "social capital." As noted by Jackson, such terminological disparity is one of the reason why the term is not used in his book [36]. We define our own Reciprocal Social Capital in a manner respecting the following economic considerations:

- pairwise transactions
- value decays with time unless more activity occurs
- transactional adjustment takes into account the capitals of both parties
- iterative definition launches a dynamic economy on all nodes through time

Mobius, Quoc-Anh, and Rosenblat [52] define SC (SC) as:

> SC helps to internalize externalities for which there is no market and where transactions costs are too high to write complete contracts. Informal credit arrangements, financial and in-kind assistance to neighbors and friends or investments in public goods are just one of the many examples of SC.

They provide a simple definition of SC within a community, and by building on the work of Andreoni and Miller [2] measure SC in a real-world social network using a series of experiments.

Their methodology distinguishes between two sources of SC: preference-based, and cooperative SC. Preference-based SC is based on the principles of simple altruism – in that agents can internalize the externalities if they take each other's utility into account. However, the strength of altruism is expected to vary systematically with the relative position of agents within the social structure which makes the empirical calibration of such a model dependent on the individual degree of altruism for each actor.

Cooperative SC arises from repeated interactions between pairs or groups of agents. Such interactions make agents appear to act like altruists even if they have perfectly selfish preferences. Due to the multiplicity of equilibria in repeated games the empirical calibration of the model provides different perspectives on the extent and relative importance of cooperative SC.

There is evidence of both cooperative and preference-based SC. While there is considerable heterogeneity in the base level of altruism amongst agents, the authors found that preference-based SC increases the weight on a friend's utility by about 15% while cooperative SC adds another 5%.

While applying to a small-scale network of students with terrestrial sociological methods of eliciting utilities, the game-theoretic underpinnings distinguish this approach as a serious attempt at a quantitative definition of SC. If this approach lacks dynamic or currency qualities, it is because of the strict definition adopted by the authors. It is a promising avenue of investigation, capable of many potential investigational intersections with our reciprocal SC.

In his monograph "Social and Economic Networks," [36] Matthew Jackson considers a wide spectrum of models regarding network formation. Especially interesting is the economic approach which puts a maintenance price and benefits on links, and then examines, for a variety of costs and rewards, the resulting network topologies. This approach shows how common-sense assumptions on elementary transactions between network members can lead to specific higher-level structures following from them. We follow a similar philosophy regarding our reciprocal social capital.

3.5 Accidental Influentials

The question of personal influence in a social network, in its modern sense, is usually traced back to the seminal 1955 book "Personal Influence" by Elihu Katz and Peter Lazarsfeld [38]. The main concept this book is credited for is the two-step influence model, defining "influentials" as those community leaders who filter and structure media streams for consumption by their local audiences. However, another important contribution is emphasized by Katz and Lazarsfeld. In Katz's own words from the preface to the 50th anniversary edition of "Personal Influence,"

> 1. The leadership of "opinion leaders" is spelled with a small "1." As everyday influentials, they are ubiquitous; it has proven of little use to single them out one by one. The secret is to locate those segments of a population that influence other segments in each particular domain.

We would like to argue that community leadership is a structural characteristic of a person, not limited to their ability to diffuse information, and role of leaders can change depending on the context. In our RSC terms, leaders of the poor sometimes are followers of the rich. Katz says,

> Where you find an opinion leader, you are bound to find a conversation – about politics, fashion, marketing, movies, education, sports, etc. Studying conversations is a better way to understand persuasion than simply nominating leaders and followers.

Duncan Watts and Peter Dodds, in their paper [65], consider the question of how quickly the influencers can cause a cascade that will span the whole network, and whether the role of influencers is as significant as presented by Katz and Lazarsfeld. Watts and Dodds create and examine a synthetic network, in which they define diffusion as propagation of change of a binary state similar to a neural network. Their influencers are simply the top 10% of an influence distribution. Their analysis concludes that the average threshold of excitability is a much more important determinant for the overall cascade power than the influence of a node which sparked it, or a set of such nodes.

The popular press has referred to Watts and Dodds' paper as "Accidental Influentials," and arrived to a conclusion that in many, if not most, settings influentials are somehow random. Harvard Business Review, in its 2007 February issue, lists Accidental Influentials as #1 in the list of 100 breakthrough ideas [66], with its key author, Duncan Watts himself clearly being an important influential proposing it! But just as Malcolm Gladwell promoted the ideas of influentials in social epidemics as important, the promotion of the idea of random influentials may be seen as an example of the media employing a two-step influentials model itself to cast doubt on the notion of influentials across the board. Watts ad Dodds's academic paper and its Harvard Business Review summary are carefully framed for the specific diffusion problem they were designed for.

At the same time, Watts and Dodds study a specific artificial model, and frame their findings as specific questions for Katz-Lazarsfeld model, as Watts and Dodds clarified in our meetings and private communications. We quote from Duncan Watts' email, with permission.

> I don't think, however, that there is any fundamental disagreement between Katz and Lazarsfeld's view of the world circa 1955, and what we wrote in 2007. What we were disputing, rather, was the way in which their ideas have been interpreted subsequently.
>
> The problem, I think, was the conflation of the idea of opinion leaders with that of diffusion– specifically that the exponential growth observed in diffusion curves is attributable to opinion leaders. This claim is demonstrably false, at least as a necessary condition, as is easily verified with just about any diffusion model. Nevertheless, the idea has tremendous appeal and caught on in the marketing world, culminating in Gladwell's coinage of "social epidemics" which he explicitly attributes to the "law of the few".
>
> Katz-Lazarsfeld never said anything about social epidemics or the law of the few, so in disputing that social change is driven by a few special people who trigger social epidemics, we aren't critiquing them at all.
>
> Another point that I think is worth clarifying (although I thought it was clear already) is that we don't claim that some people are not more influential than others. No doubt some

> people are, and if you want to call these people influentials, go ahead. That's not what we were objecting to. Rather what we were objecting to was two points: first, that people use the term "influentials" (or influencers or whatever) to refer to LOTS of different types of people, who exert different kinds of influence via very different mechanisms. So before one starts making any claims about what influentials do or do not do, one first has to be very clear and explicit about what one means by the term in the first place. No-one was doing this, and unfortunately they still aren't. The second point we were making is that just because some people are more influential than others doesn't mean that they are responsible for triggering social epidemics. It's not even clear, in fact, that social epidemics are the right way to think about social change.
>
> Unfortunately, both these points have been ignored or misunderstood. It might be useful to clarify them, but unfortunately my experience to date is that most people aren't very interested in clarification. They like stories, and influencers make for great stories. So although I think it's a worthwhile exercise to start mapping out the different kinds of influence, and the different ways in which different people become influential, it's not going to transform the debate overnight.

In their 2011 paper "Everybody is an Influencer," Bakshy, Watts, et al. [4] track the spread of shortened URLs by Twitter users, and show that the role of important individuals in diffusing such contents is not decisive. The kind of influence implied by Bakshy, Watts, et al. is very specific ability to cause a change of mind among the interlocutors, or spread certain types of information. This is a model similar to that studied by Cosley, Kleinberg et al., with regard to collaborative Wikipedia editing [17].

We argue that diffusing an opinion to change one's mind is different from forwarding memes contained in URLs, different from from an edit of a Wiki page, and vastly different leadership qualities are required to succeed at any one of these tasks. The influentials for any of them may, or may not, overlap with those good at the other tasks.

The kind of influence we focus on relates to the fundamental properties of human social networks in such as they are always used for communication. We believe that before anything of value can occur in such a network, such as an information cascade, the communication routes must already exist, or at least be determined to a large degree. We use the transport network as our analogy, before you study traffic, you need to establish where the routes are.

We look at how important and reliable individual links are by assigning value to individuals based on how well they maintain such links with other individuals. We study replies, which we argue are basic communication elements, as opposed to following RSS feeds and other activities based on network-specific artifacts. Replying follows the same dynamics as any other human conversation, with the same set of expectations of the need to reply and everything else it entails, as we describe in detail in Chapter 6.

Katz notes [ibid.] how social network analysis, including Watts', is a natural evolution of the original Katz and Lazarsfeld's Decatur study:

> By now, there's a flourishing field of research on social networks (Watts, 2003), inspired, in part, by the Columbia tradition (Kadushin, 1976; Burt, 1999). While diffusion is one of the applications of this work, even here – as in reception studies – only little effort is being made to bring the media back in. It is interesting to note, again, how the media have almost dropped out of sight: compare the evolutionary course that led from selectivity to gratifications to reception with the course that led from interpersonal relations to diffusion to social networks.

Chapter 4

Exploration

Exploratory Data Analysis, or EDA, is a standard first step in modern data statistical mining [63]. A good example of a workbench where you can perform statistical EDA is R, the open-source statistical programing language and ecosystem. We propose an EDA process for social network data. The question becomes, given a snapshot, a dynamic dataset, or a stream of OSN data, how do we find points of interest, such as people, topics, and conversations relevant to our concerns? How do we begin to define them? One way to gain insight into OSNs is to identify tweets about a *topic of interest*. Tweets can be searched by keywords, in order to select those about a specific topic. Then, people interested in the topic can be found on the end points of those identified messages. Once the dialogues are discovered, communities can be structured around them. The process works as follows. First, we select those dialogues with the highest number of exchanges. Then, for each conversing pair A, B we look for all such Cs which talk both to A and B, preferably on the given topic also. Growing by mutual connectivity only, in triangles, or by topical ties can be parameterized.

Social networks generally provide an implementation of some kind of groups or communities that users can voluntarily join. Twitter does not have this functionality, and within Twitter there is no notion of a formal group or community. We propose a method for identification of communities and an assignment of semantic meaning to the discussion topics of the resulting communities. By using this analysis method on a sample of roughly a month's worth of Tweets from the Twitter's "gardenhose"

feed, we demonstrate the discovery of meaningful user communities on Twitter.

We examine Twitter data streaming in real time and treat it as a sensor. Twitter is a social network which pioneered microblogging with the messages fitting an SMS. Now Twitter is used by individuals, businesses, media outlets and even devices all over the world for status updates via plethora of clients, browsers, smart phones and PDAs. Often an aggregate trend of statuses may represent an important development in the world, as has been demonstrated with the Iran and Moldova elections, and the anniversary of the Tiananmen protests in China.

We propose using Twitter as a sensor for tracking individuals and communities of interest, and for characterizing individual roles and dynamics of their communications. We develop a novel algorithm of community identification in social networks based on direct communication, as opposed to structuring communities via follower links. We show ways to find communities of interest and then how to browse their neighborhoods by either the similarity or diversity of individuals and via groups adjacent to the one of interest. We use frequent collocations and statistically improbable phrases to summarize the focus of the community, thus giving a quick overview of its main topics. Our methods provide insight into the largest social sensor network in the world and constitute a platform for social sensing.

4.1 Exploration Workflow

The preprocessing of social network communications creates two databases, for content and structure. It consists of the following steps:

1. obtain the stream of data from the social network
2. index the twits in a text search engine (Lucene)
3. build the communication graph (Berkeley DB)

Preprocessing is illustrated in Figure 4.1. Note that the term "preprocessing" here does not preclude online operation as the data stream can be indexed continuously, and both Lucene and Berkeley DB allow for efficient real-time growth.

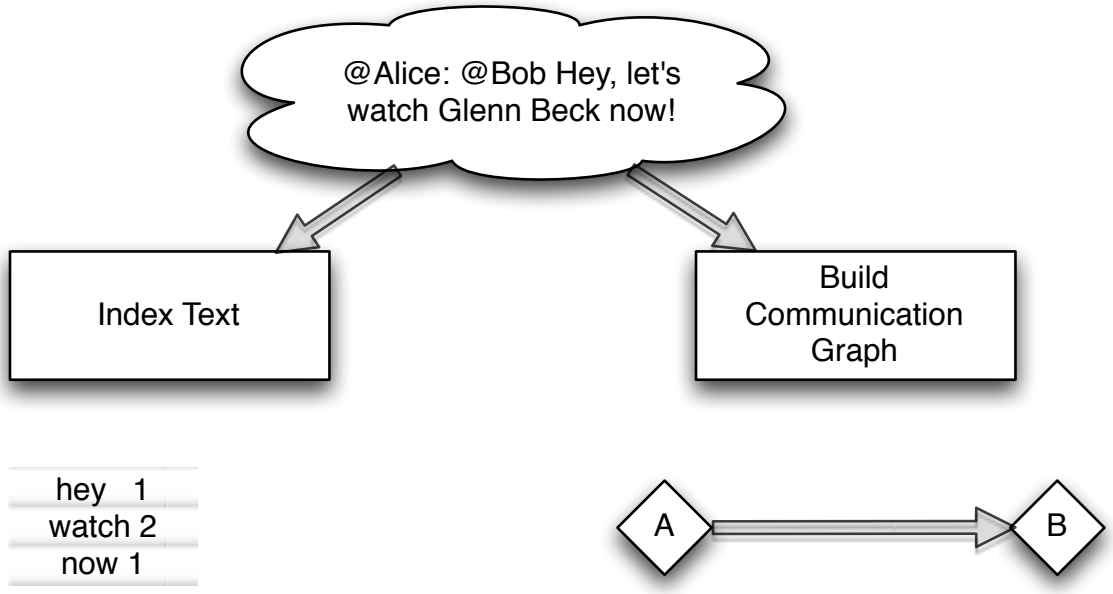

Figure 4.1: Preprocessing social network data for exploration. Message text is indexed in a full text search index, and the communication graph is stored in a fast associative map database.

Once the communication graph and the message text index are built, the community sensing workflow proceeds as follows:

1. Come up with a set of keywords to occur in the community of interest

2. Identify a seeding pair of repliers about the topic

3. Grow a community of replier triangles from the pair

4. Identify the characteristic phrases in the community discourse

5. Identify the fringe of the community, select the new topics and seeding pairs for further iterations

The exploration workflow is shown in Figure 4.2.

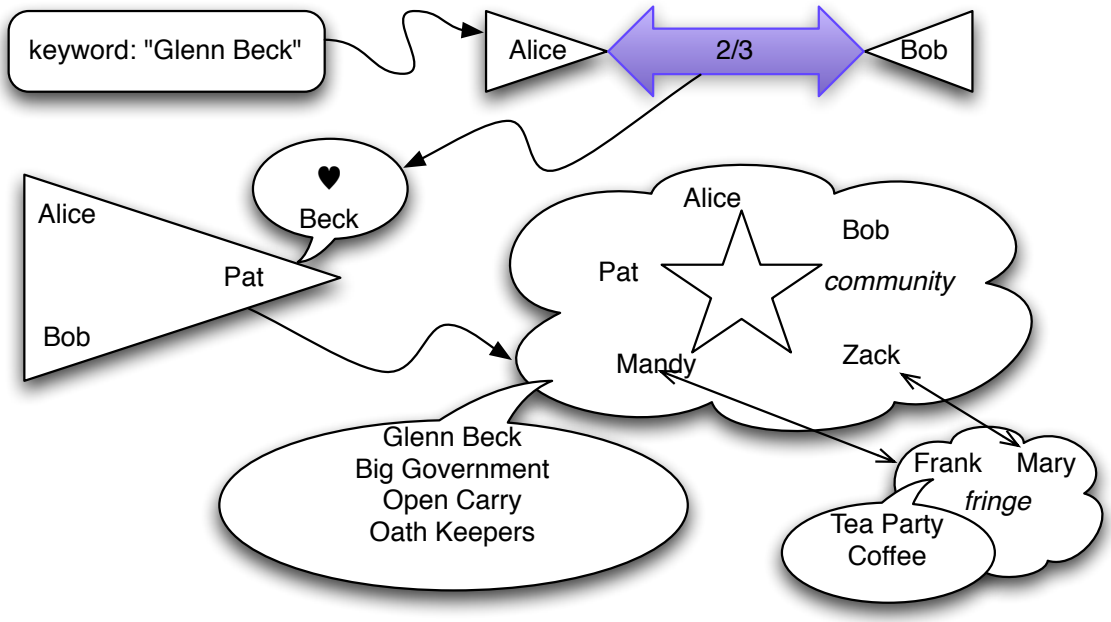

Figure 4.2: Exploration workflow. A keyword is searched and the pairs using it most are found. Then a community is grown around such a pair, and its topics are extracted. The fringe and its topics are also listed, allowing for subsequent pivoting and iteration.

We begin the analysis by searching the Lucene index for pairs of users who discuss a keyword. All of the twit contents are indexed. Various twit fields are treated separately which allows us to find any replies quickly.

We explored finding both weak and strong pairs of users. A weak user pair is a pair of users who at least once in their corpora use the same keyword in at least one of their twits and additionally have at least one directed message between them. A strong pair of users has at least one directed message between them which uses the keyword being searched for. For our analysis we used strong user pairs

as we found that they gave us results more like what we were attempting to find as a community. Additionally, before generating communities based on these pairs we pruned them based on the number of times the keyword was mentioned between the users.

We then used these user pairs as seeds for growing communities. We used a recursive community definition based on a triangle property such that a user is a member of a community if they have been sent messages by two of the users of the community. If that is the case, the "child" user is added to the community connected to the two "parent" users. The child users are added recursively, until there are no more users to add. Any users who have been sent messages by exactly one of the members of the community is considered to be on the "fringe" outside of the community, or simply the fringe.

We performed collocation and Statistically Improbable Phrase (SIP) analysis on these communities in order to find the resulting topics of conversation in the communities. These Natural Language Processing (NLP) approaches are used to identify the most characteristic phrases describing the community discourse so as to approximate "what are they talking about."

4.2 Previous Exploration Work

Community identification in graphs with no specific community marking is an active research area. Flake et al. [25] propose a definition of a community as a group with more links inside than outside. It turns out such a definition requires a simultaneous partitioning of the whole graph into communities in order to work and the communities cannot overlap. Mishra et al. [51] overcome this with a notion of (α, β) clusters. A subset of vertices forms an (α, β)-cluster if every vertex in the cluster is adjacent to at least a β-fraction of the cluster and every vertex outside the cluster is adjacent to at most an α-fraction of the cluster. Such clusters can overlap. Backstrom et al. [3] use self-declaration as criteria for membership, namely joining a LiveJournal community which requires explicit joining, or publishing in a conference for DBLP-based communities. Statistically Improbable Phrases (SIPs) are effectively used by Amazon to characterize contents of books. They give a very good idea of what a book is about. Savell and Cybenko [57] provide quantitative formalisms for social behavior in a group context. Khrabrov and Cybenko [45] provide metrics of group cohesion and influence. Sociologists have many useful metrics for community formation with quantitative measures of influence as well. Bonacich [9] and Friedkin [26] provide metrics which can be directly applicable to many Twitter-mining tasks.

4.3 Text Indexing

For topical search we index the twits and search them with Lucene [18]. Lucene is a powerful open-source search engine which uses flat-file inverted indices providing fast full-text indexing and searching. The fundamental unit in Lucene is a document which can have any number of fields, each with a number of indexing parameters.

We created a custom analyzer for Lucene thus allowing us to preserve the meaningful special characters that appear in twits. The indexer tokenizes the text based on characters which are neither letters, numbers nor the @ or # symbols, and creates a full-text search index of those tokens. We also stored metadata for each twit, including the date the twit was posted to the twitter stream, whether the twit contains a hashtag, is it directed, and, if so, who it was directed to.

We created two Lucene indices on a month's worth of twits from the Twitter *gardenhose* stream, that contain a statistically representative fraction of all twits (which can be as high as a fourth). The first index was created with each twit as a document. A second index was then created on each user's corpora as a document. Using these two indices we created a number of API functions to perform textual analysis on the Twitter data.

In order to ensure that we would receive meaningful results from the analysis of the Twitter data we created a custom stop word list generator which could generate a stop-word list based on the whole corpora of words used in twits. We created a number of functions which performed textual analysis on the dataset.

1. *topChatters* – gets the users with the most conversation messages
2. *twitUserIds* – gets all the users and twits for a term
3. *topWordsByTwit* – gets the top words in the combined corpora of the list of twits
4. *topWordsByUser* – gets the top words in the combined corpora of the list of users
5. *userPairTopics* – gets the top words for pairs of users
6. *getStopList* – creates a stop list from the most commonly used words in the corpora of all twits

In addition to the API functions which interact with the full index on a per twit basis we created API functions which operate on the per user corpora index. These functions primarily use queries with very large numbers of terms to perform cosine similarity analysis on the corpora of a single user or a group of users, comparing them against another user or group of users. Through this procedure we were able to compare a candidate to join a group to the corpora of the group as a whole in order to rank users who are considered to be on the fringe of a community for inclusion into the community.

4.4 Community Detection

Many previous and current Twitter popularity metrics focus on the followers graph. Indeed, Twitterers create a datastream, similar to RSS news, by subscribing to, or following, other Twitterers. However, as pointed out by Finin et al. [37], the number of followers doesn't necessarily predict influence or social bonds. What matters are actions and in case of Twitter, direct communications, called replies.

Anybody can address anybody else publicly, by simply inserting the addressee's nickname, preceded with @, into the twit (which is then called reply). We can extract replies from all twits and build a communication graph out of them. Such a graph can be static, being a snapshot at a certain time, or dynamic, where nodes and edges are added with timestamps, constituting a multigraph. When looking for communities we propose to do so through the communication graph thus revealing those engaged with each other, presumably over a set of topics.

Our exploration workflow as outlined in Figure 4.2, begins with a search phrase such as "Glenn Beck." We then identify pairs of repliers from the communication multigraph who exchange this phrase in both directions, and sort the pairs by the number of total exchanges they had. The most active pair is the top one, and this pair becomes our seed candidate for the community. We grow the community from that seed pairs.

Our community identification algorithms is indirectly inspired by Backstrom and Kleinberg [3], who observed that when studying the dynamics of the community growth, a new member is much likelier to join a community when he has two friends already in the community not yet connected themselves (an open triad). We simplify it by requiring that a prospective member has at least two friends already in the community, not insisting the two friends have no links.

Hence, the algorithm proceeds as follows. We start (*) with a pair, (A, B), in a queue, who are the seed members of the community. They may or may not have a link. (An option may further control whether there should be messages in each direction, signifying mutual interest, i.e. at least two, or one will suffice.) We sort friends of A in the decreasing order of the total number of messages exchanged, and then sort the same way with B's friends. From this we find the first shared friend in the list, C. C

is then added to the community, and the pairs $(A, C), (B, C)$ are added to the queue. The process (*) is then repeated until the queue is exhausted. Figure 4.2 contains an example where the seeding pair, Alice and Bob, both talk to a third user, Pat, the most, hence bringing her into the community. The process is repeated until no new triangles exist, or until a set limit is reached. Two kinds of limits can be set: by the total size of the community, or the distance, in generations, from the seeding pair. The generations are natural layers, and computed as follows. The seeding pair members both are generation 0. In a triangle, the new member gets the lowest parent generation plus 1. The process can be stopped when a given layer is reached.

There are certain asymmetries in the communities grown by the versions of this algorithm where only one direction of the edges are required. If we allow for a single directed edge as a (topical) relationship marker we get different communities when starting with the pair (A, B) rather than (B, A). Such differences may be considered as a result of the bias towards A or B, or compensated for by requiring edges in both directions. We ensure symmetry by requiring both edges.

In our studies using several seeding topics, we found that the communities saturate fairly quickly, conforming to the idea of a tightly knit cluster [51] being characteristic for a community.

The fringe of a community is a set of users connected, in the communication graph, to at least one member of the community, but who themselves are not members of the community. Once the fringe is detected its twit set can be examined for SIPs. The SIPs can provide a set of choices for further exploration, in order to restart the community building process. We outline such a process as a basis of data exploration below.

Community graphs are build from a set of directed edges – $E(from, to, weight)$ – where weight is the number of messages which traversed the edge in the dataset. Since the time sample was so long there is an not insignificant number of edges with low weights. Such low weighted edges represent transient communications between users and more closely resemble noise than structural components of the communities.

In order then to create meaningful community structures, the edge set E must be trimmed of edges with low weight which can, if further analysis is performed, obscure the tightly knit central clusters

we hope to find. For the data sets used for this paper we chose a cutoff weight based on the obvious threshold as seen in the histogram of the weights on the graph, usually a number of twits between 1 and 5.

4.5 Exploration Results

Using the aforementioned analysis procedure we found resulting communities and their related collocations and SIPs for a number of search terms. We describe one such complete workflow for a political community centered around Glenn Beck.

Glenn Beck is a controversial conservative cable television show host on the Fox News network. We performed a search for a strong pair of users who mentioned both "Glenn" and "Beck" in at least one directed twit. We then grew communities from the set of the 38 seed pairs that resulted using the recursive community growth algorithm described above. We expected that, because of Glenn Beck's controversial nature, we would be able to find distinct communities with little or no cross interaction.

The full graph of the 5 interacting communities can be found in Figure 4.3, Appendix A, "Full Glenn Beck Graph."

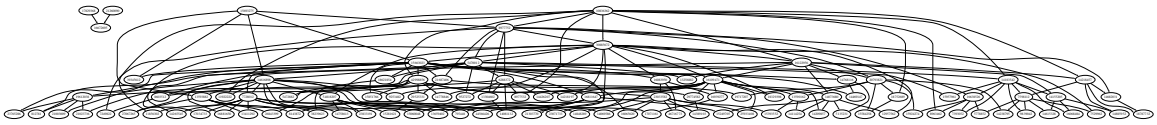

Figure 4.3: The full graph for one of the five major communities found for the the Glenn Beck search. The node labels are the users unique Twitter IDs.

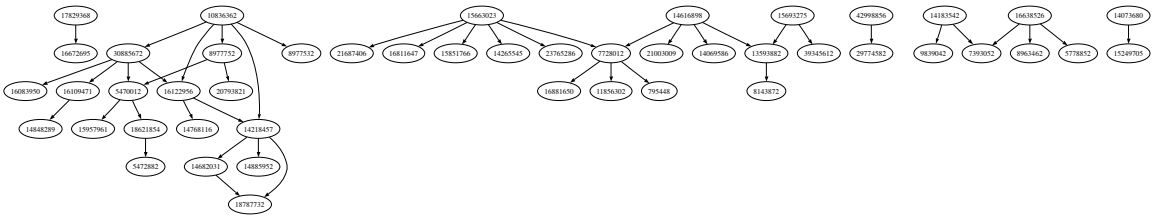

Figure 4.4: The reduced graph for one of the five major communities found for the the Glenn Beck search. Edges which did not meet a minimum weight were pruned from the graph. If a node was unconnected after pruning it does not appear in the graph. The node labels are the users unique Twitter IDs.

4.6 Discussion Topics

For the communities we identifed we looked at the typical topics discussed. One way to find the topics is to look at the top *n-grams* [14] used in the text of the messages exchanged. These will often be trivial and shared across many communities, with bigrams such as "I need…" A more interesting source for characteristic topics are the SIPs. They are widely used on Amazon.com to characterize a book. The SIPs are found by looking at n-grams and comparing their actual frequency to the model where the words may happen to be together randomly. These n-grams where such repeated co-occurrence is very unlikely for random words are "improbable" in terms of randomness, in such that they uncover some regularity.

Some of the Glenn Beck community SIPs are shown in Table 4.6.

Our method is a convenient way to browse through a social network. Given the topical community search, we have both an entry point and a pivoting procedure. After a community is seeded, grown, and its topics are revealed via SIPs, the explorer can either stop there and look at the users of interest, or detect the fringe and pick a topic from the fringe's SIPs or a pair crossing the fringe boundary. That pair, or a fringe SIP can be used to seed a new community for the next step of browsing, which we call pivoting.

When keywords contain spatially or temporally localized topics, our workflow can be used for community sensing – following the communities in a specific environment in the topical or geographical space. For instance, the 2009 crisis in the New York State Senate can be isolated using the name of a key Democratic senator, Pedro Espada Jr., whose actions helped bring the senate business to a halt

Reining Medical Suits	Freedoms First Rally
Voting Rights Act	Martin Luther King
Gitmo Detainee Lawyer	Convo Gitmo Detainee
Able Bodied One	Payer Health Care
Speaking Your Truth	Town Hall Friday
Henry Louis Gates	Gov Mark Sanford

Table 4.1: Statistically Improbable Phrases from the Glenn Beck community. The phrases are trigrams which co-occur much more than random three words would. They are the political issues and personalities of the day, circa Summer of 2009.

thus causing much debate on Twitter. Those users who mention his name are likely to follow the state politics; similar topics exist for engaging issues in every locale.

Chapter 5

Influence

The web is changing every second, and becomes more and more interactive. Traditional communication networks, such as email, still dominate, and in many cases require analysis, such as has been done on the Enron data set. When presented with such a dataset, an important question is, who's important here and who's not? Whose influence should we discern behind the dynamics of the network to understand its processes?

While forming a social circle on Twitter, people often follow those who are authorities on their professional or personal interests. Since most of the experts now publish their Twitter *@nickname* on the Web, it is easy to follow them. In many cases people converse on Twitter as they used to do on blogs. Advice is often sought in conversations, and decision are arrived at via consulting the subject or social authorities. An influential microblogger can make or break a company's reputation, speed up technology uptake, accelerate a trend, or contribute to a decrease in popularity of a company, person, or idea. It is important for those in the public arena to understand who wields influence in the social networks, how they become influential, and how this influence evolves or devolves over time.

In this paper we study conversational dynamics-based influence in communication network graphs. We base our analysis on Twitter as a publicly available communication network. We look at the communication graph formed by messages, or tweets, from one person mentioning another with replies (or mentions) mapping to an edge in the graph. We look at several graph metrics dynamically, computing

them for every day in the study, and then look at the users whose metrics dominate or accelerate faster, or grow longer than others. Our results show that such users form interesting classes, and one of our metrics, the *starrank*, identifies a point in time when a user may be considered a public person.

We believe that this work can be used in a variety of ways. Individuals can track the changes and the dynamics of their influence as well as their peers' influence, competitive intelligence analysts can focus on the dynamics of consumer opinions regarding their products and services with a deeper understanding of who has key influence in shaping attitudes and opinions, and operations researchers can study actors changing influence within an organization.

Java, Finin, et al., [37] looked at the power users of Twitter and found that they activated more replies. They emphasize the difference between the declared network of followers and the actual network of social interactions. Backstrom and Kleinberg [3] studied community growth in two social networks, LiveJournal and DBLP (computer conferences). They looked at a few temporal snapshots, discovered certain rules of community growth, and identified several graph structures which make such processes as joining a community more likely. Khrabrov and Cybenko [43] applied sequence modeling to the cell phone tracks of a group of MIT students, using n-gram models and suffix trees. Savell [57] and Chung [16] developed techniques for identifying and analyzing business process dynamics within social networks using a variety of methods including Process Query Systems [19].

5.1 Dataset and Methodology

We started monitoring Twitter using its new Streaming API from June 2009 onward. Our subscription level, the so-called "gardenhose," provided a significant fraction of all daily tweets. We started with about 2-3 million tweets a day initially, and in a year were getting up to 5 million tweets a day. The working set for the studies in this paper consists of a 100 million tweets from October 16 to November 17, 2009. From those tweets we computed our metrics on the leading 90 million tweets, for a time period of three consecutive weeks.

We built the replier graph from this working set in an incremental way. For each new day, we added that day's mentions as directed edges, and counted them as the number of repliers for the source and the number of mentions the target. We computed the pagerank [10] of each user at the end of the day, using the Jung2 toolkit [55] with $\alpha = 0.15$, over 100 iterations. Given the pagerank for each user, for each day, we defined the *drank* of every user that day by:

- Sorting the *(user,pagerank)* pairs by increasing pagerank,
- Replacing each user's pagerank by its position (starting from 0), and
- Optionally normalizing into 0-1 range by dividing by the list's length.

Table 5.1 contains a step-by-step example of the *drank* computation.

The number of users with a given pagerank increases daily, as more and more users engage in Twitter conversations. The daily number of ranked users is shown in Figure 5.1.

Normalization compensates for the ever growing number of users active on Twitter every day. However, the effects of growth are better observed on the integral *dranks*. We refer to the integral, non-

Table 5.1: *Drank* Computation

[a:0.1 b:0.02 c:0.3 d:0.2]	original user:pageranks
[b:0.02 a:0.1 d:0.2 c:0.3]	sorted by pageranks ascending
[b:0 a:1 d:2 c:3]	pageranks replaced by position, 0-based - this is the *dirank*
[b:0 a:1/4 d:2/4 c:3/4]	normalized by the list length - this is the *drrank*

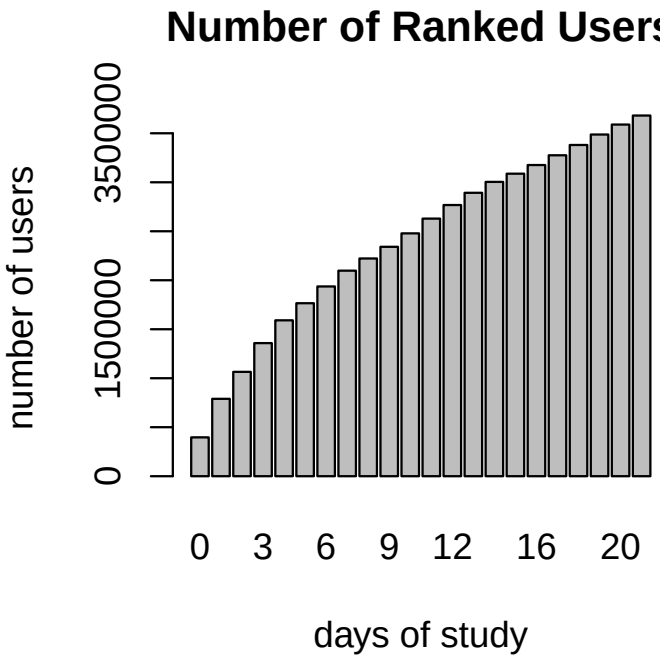

Figure 5.1: Twitter's user base continues to grow significantly

normalized *drank* as *dirank*, and the rational, normalized one as *drrank*. Simple metrics, such as the number of mentions a user gave or received each day, are represented as integers, which in the dynamic contexts accumulate as integer lists. We look at the number of repliers (different targets mentioned by the same source), mentions (different replies with the same target), and total number of tweets by a user, daily, and aggregate the data into integer lists.

The resulting histograms contain only those users whose complete corpus is nondecreasing or increasing. To look at all users, we partition their lists into longest contiguous nondecreasing or increasing subsequences, and compute the maximal length and/or acceleration of such a subsequence. The acceleration is a ratio of the last element over the first; since many first elements are 1, we also look at a "tougher" version, in which the acceleration is with respect to the second element. The minimal length of the subsequence in each case is 3, showing at least two days of growth (or at least no decrease).

Many features related to influence increase or decline along with time, and are represented as a

Table 5.2: Cont. Longest Increasing Subsequences

[1 2 3 0 4 5 8]	original sequence
[[1 2 3][0][4 5 8]]	subseqs partitioned by $<$,*maxlen* 3
$[\frac{3}{1}\ 0\ \frac{8}{4}]$	subseq accelerations, *maxxel* $3/1$, *maxxel-tough* $8/5$

Table 5.3: Grow or Fall

[1 2 3 5 0 7 6]	original sequence
5 pairs up, 2 pairs down	:grow both simple and qualified
rate of change is $6/1$	passes :twice-higher filter

real-valued time series. Examples are the number of mentions a user receives daily, his *drrank*, and *drstarrank*. Totals such as number of mentions given or received can be counted separately for each day, or cumulatively from the beginning of the study. Here we describe several primitives which we combine for our analyses.

Contiguous Longest Increasing (generally Monotonic) Subsequences – This operation (*clis*) takes a sequence, an ordering function – one of $<, \leq, \geq, >$ and partitions it into subsequences such that each subsequence is ordered according to the predicate. Acceleration for each subsequence is the ratio of the last element to the first; the *:tough* variant divides by the second instead, skipping the frequent 1 for temporal count sequences. We can drop the leading 0s also when computing accelerations, or filter for them by multiplying the ratios out against a threashold. We denote the maximum length of a *clis* subsequence as *maxlen*, and the maximum acceleration as *maxxel* (or *maxxel-tough*). Table 5.2 provides an example of the *clis* transform.

Grow or Fall – This operation (*growfall*) takes a sequence and counts how many successors are greater or less than their predecessors. A decision is made whether the sequence is mostly "grow" or "fall" by either the simple $(1:1)$ or qualified majority $(2:1)$. Optional *twice-higher* filter keeps only those sequences where the change between the first and last element is twice or more. The result can be either categorical – *:grow*, *:fall*, or *:neutral*; or quantitative – the rate of change, with 0 for neutrals, and the sequence returned ordered by the rate of change. Table 5.3 shows an example of the *growfall* transform.

The *starrank* considers a user's importance with respect to his neighborhood. Figure 5.2 shows a

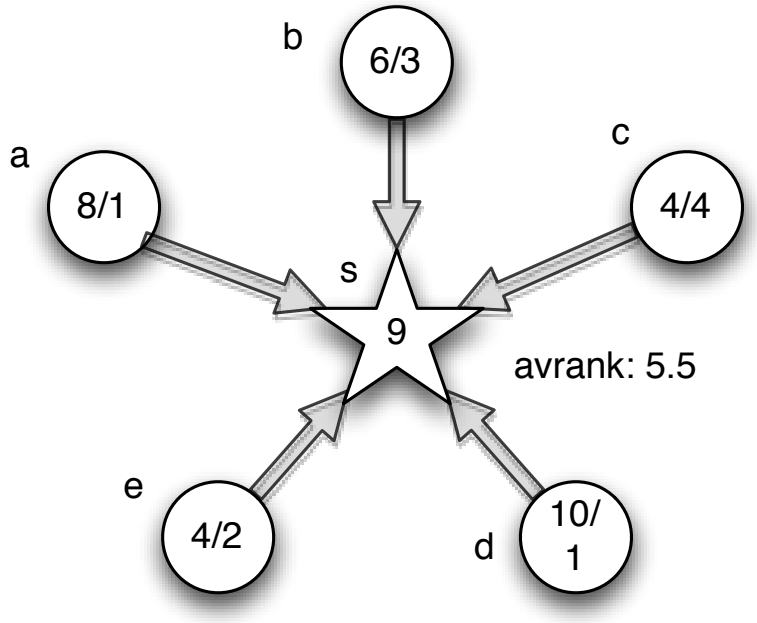

Figure 5.2: Example of *starrank* computation. The center user with *drank* of 9 is mentioned by 5 other users with the given *drank*, exchanging one or more tweets that day, e.g. $r/n = 6/3$ means 3 mentions by a user of *drank* 6. Then the *distarrank* is the average of rs weighted by the ns, here $(8*1+6*3+4*4+10*1+4*2)/(1+3+4+1+2) = 5.5$

user with his mentioners on a particular day, along with his and their *drank*s. The *starrank* is an average of the neighbors' ranks, weighted by the number of communications with each neighbor for that day. Depending on the kind of the *drank* we use, we'll get a corresponding *starrank*. For instance, a user's *distarrank* for a day will be an average of the *dirank*s of her neighbors, i.e. of their positions in that day's sorted order of pageranks.

5.2 Influence Findings

For individual ranks sequences we can look at the longest growth or decline periods. We partition each sequence into contiguous increasing or decreasing subsequences and take the longest one. For the number of mentions, we look at the total amount of users for whom it is nondecreasing, and look at their longest runs over days.

Figure 5.3 shows how many users all with non-decreasing mention runs have full sequences of at least a given number of days. Each bucket is the total number of users who achieve that many days of stability or growth.

Figure 5.4 shows the same for non-increasing *dirank*. In our study the top nondecreasing rank of 0 is invariably held by Justin Bieber (see below on the Bieber "Ecosystem").

The users with the longest period of growth in their *drrank* include:

- Brazilians. Most of our influence growth metrics for repliers bring up a lot of Brazilian journalists, stars, and power users. Apparently, Brazilian twitterers are more communicative than others, using Twitter more like a conversation medium than a diary. The top ones include *@leobarcellos*, a web designer, *@natyperdomo*, "Publicitária," and more.

- *@alfiehitchcock* – a London photographer posting his pictures via TwitPic

- *@Theresamcardle* – an "Optimist; Toilet seat marketer at Kohler; UW-Madison MBA student"

Those with the longest contiguous increase in real *starrank* of mentioners – i.e. losing influence – include such accounts as *@gm_web*, an automatic repost of a weather station, and *@AlexanderFog* – a self-referential DJ mentioning himself in every tweet.

The second-order characteristic of growth is acceleration. For each contiguous increasing or decreasing subsequence the acceleration is given by the ratio of the final state to the original value. For a quantity such as the real rank, where growth means decrease, we invert it so that the bigger acceleration, the better the growth is. Once again, the pack of *starrank* acceleration-sorted users is dominated

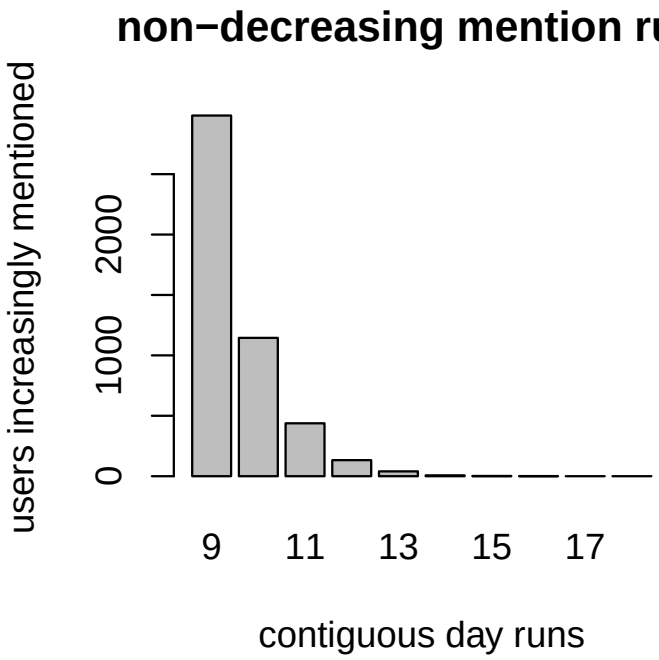

Figure 5.3: The number of users whose daily mentions are all nondecreasing, per day

by many of the same Brazilians thus showing in this instance that the longest active growth also often is the fastest one. Some of the new faces here are

- *@jc_schuster* – a country music fan
- *@drosa_shannon* – "twin,married, love DOOL,BL,American Idol and Tweeting"
- *@carlaciccarelli* – "Publicit'aria e Produtora de Comerciais,V'ideos e Eventos"
- *@Dollbabyv* – a girl writing about her dating, referencing a Blogspot blog

The *starrank* clearly demonstrates how a public figure's tweets constantly spread his or her influence through to ever increasing realms of new users, with lower *dirank*, thus increasing the star's average rank with respect to its audience.

Figure 5.5 shows the daily *distarrank* of Justin Bieber, the most influential replier. It's not normalized to show the effect of the influx of the new users.

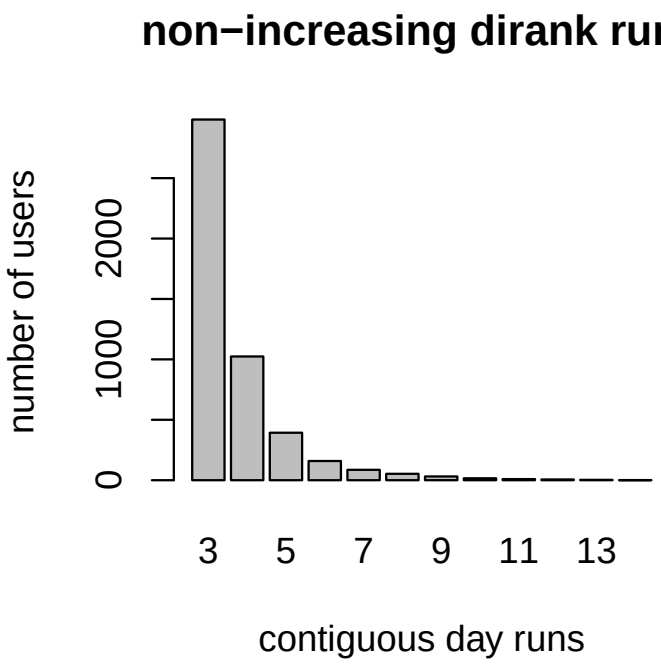

Figure 5.4: The number of users whose own daily *diranks* are all nondecreasing, per day

The normalized *starrank*, *drstarrank*, when decreasing for the longest runs over days, shows the cumulative growth in the importance of a community around the user with respect to all others. When sorted by such a longest run, we get interesting classes of influential users, such as

- *@donniewahlberg* – leader of the New Kids on the Block band and fans, who has a far-reaching and active following
- *@bowwow614* – the rapper Bow Wow
- *@faydra_deon* – "Minister; Computer Applications Trainer; Website Designer; WordPress/Joomla! Customizer; Grammar Queen; Online Bookstore Owner; AKA," posting her own "questions of the day."

When we look at the acceleration of the *starrank*, we find the users with the fastest growing communities by importance. Some of the top users with the accelerating mentioners are:

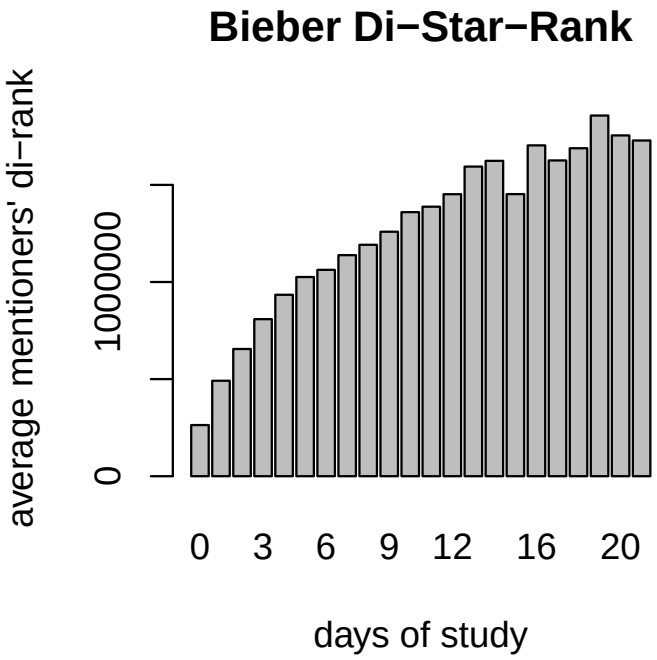

Figure 5.5: The average *dirank* of Justin Bieber's fans, decreasing daily, shows his star power spreading to the masses

- *@JoycePascowitch* – "jornalista, glamourosa….e estressada!", a Brazilian journalist
- *@Biofa* – "Jornalista cruzeirense que ama futebol, mulher e rock'n roll (meu Deus, como isso 'e bom!!!!)", a Brazilian sports journalist
- *@Dejdia* – a photographer in LA, who uses a Flickr page as the URL, and who has her own ecosystem of fans
- *@Minni_w* – "ddub Soldier (Army of NKOTB)", a fan of *@DonnieWahlberg*, one of the top-page-ranked users, the band leader for the New Kids On The Block (NKOTB). She connects with the Bieber network (see below).

The snapshots of the pagerank relationship to number of mentions reveals certain stable patterns. We cluster the pagerank versus the number of mentions graph into groups of 1000 points and plot the median of xs and ys which we call a blocked projection. Figure 5.6 shows how the resulting graphs

form a "harp" pattern which stays stable through days 10 and 20. We filter x to stay within 25 median mentions on a day, the remaining few clusters group the outliers with exceptional dynamics (or statics).

A similar pattern persists for the relationship between the pagerank of a user and the sheer number of twits by that user. A block projection in Figure 5.7 shows that despite an increase in one's number of tweets, the rank grows only up to a point, after which it starts to fall again. We show it for day 20 (day 10 is similar).

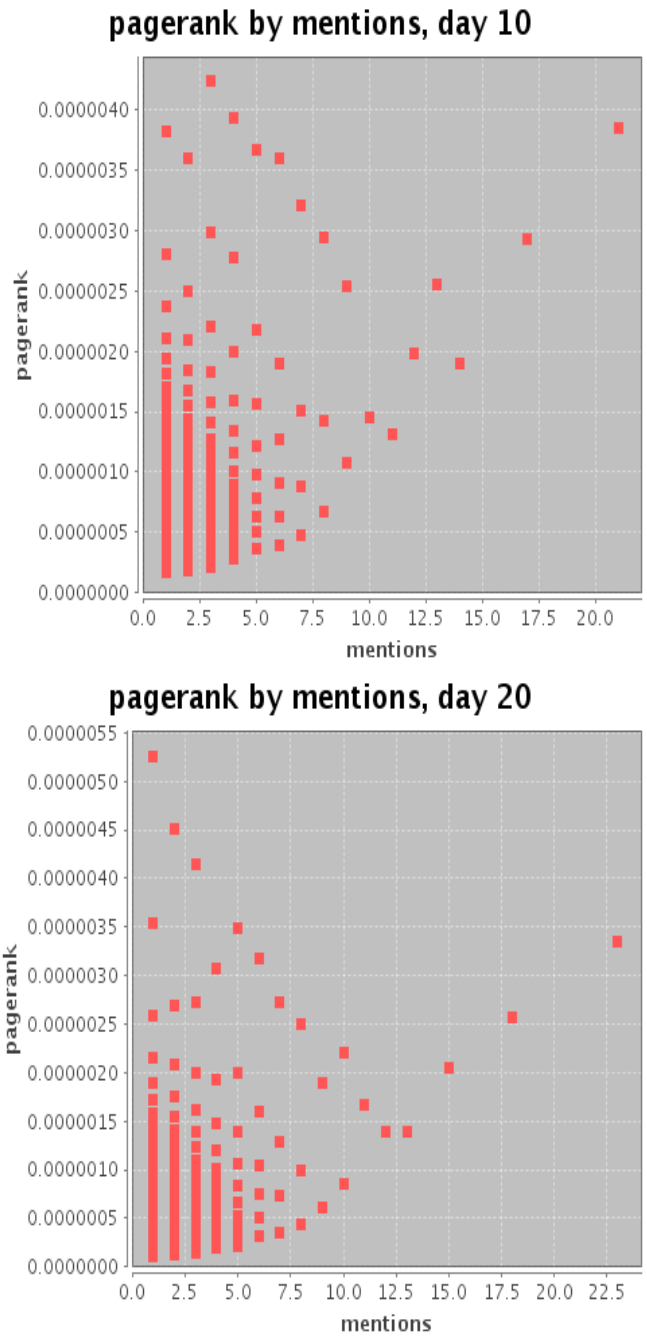

Figure 5.6: Pagerank improves with the number of mentions only so much, then ratcheting mentions is counterproductive. The harp pattern persists throughout the days

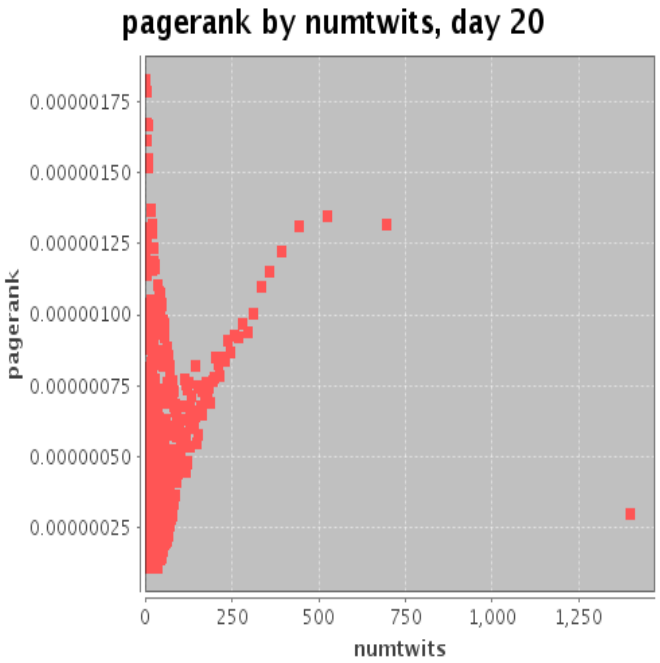

Figure 5.7: Pagerank improves with the number of twits only so far as well. X is the cumulative number of twits, Y is the resulting pagerank

5.3 The Pagerank Generation

The most surprising discovery we made in the Twitter network is the *Bieber Ecosystem*. Its growth principle can be summarized as: *You can't be Justin Bieber, but you can be Amanda D*. The person with pagerank 0, the most influential person in Twitter mentions in our dataset, each day, every day of the study, is Justin Bieber. Despite his Christmas 2009 performance at a Washington, D.C. concert for President Obama, not many people over age 15 know who Justin Bieber is at the time. An informal polling of seminar participants and friends over 18 confirmed an almost absolute lack of Bieber awareness around January, 2010. At the same time, Justin Bieber, a YouTube boy singer phenomenon, commanded a significant mindshare among the pop culture fans ranging in age from 11 to 15 years old, primarily teen girls in Canada and the United States, but also omnipresent globally from Brazil to Slovenia.

The most interesting aspect of the Bieber Ecosystem is that it is comprised of the first Twitter generation who is growing up with social networks in hand. This generation adapts to Twitter's influence seeking ways almost naturally. Their behaviors evolve to efficiently rise in the pageranks. The second

and third daily position by pagerank are most often occupied by several high-caliber fans of Justin Bieber. One of them, Amanda D., self-billed as a singer and songwriter in the South, is too young to drive and as her parents wouldn't take her to a Justin's concert in the nearest big city one of her online goals is to get him to perform in her hometown – as described on a free blog website with a link offered to Justin and others to share. What's most fascinating of the self-named "beliebers" is their active and incessant manipulation of the pagerank by constantly mentioning each other and thus growing their followers' network.

Mentioning another fan is called a "shoutout," and shoutouts are offered for trade on a quid pro quo basis ("SHOUTOUTS FOR SHOUTOUTS!"). Multiple accounts are created for special purposes, e.g. *@wewanttomeetjustin* is maintained by the two top *beliebers* for the purpose of aggregating other fans who want to meet Justin, discussing venues, options, plans, etc. User *@BieberFame* has created a separate *@JBieberCash* account noting its creation date and that Justin followed the same day, and openly referring to the original account as the creator. Very common tweets state that the number of followers is near a certain round number, such as 10,300, with only about 50 remaining, so all new followers will get a follow-back and a shoutout. Many tweets are simply multiple shoutcasts, lists of names.

The top-pageranked *beliebers* got where they are by active and clever Twitter presence and increasing their influence in a variety of ways. They trade shouts and create multiple Twitter accounts for focused subgroups, tend them regularly, and team with other top *beliebers* positioning themselves at the head of the pack. They also claim to maintain special relationship with Justin, direct-messaging with him, and offer to relay other fans' question via this exclusive channel. As far as we can see, these advantages are real. Justin mentioned both two top *beliebers*, thus propelling their pageranks to the top. The *starrank* pattern of the top fans resembles that of the stars themselves, showing a steady increase of mentions and an increase of the average rank of the mentioners propagating "into the masses." Amanda is so popular and effective that she got her own fan account, "Amanda D's Army," which in turn is run by one of the top-level Bieber fans using the same techniques which proved to work so well with Bieber's fans.

@Kekeinaction shows up high in the list of the longest contiguous growth in *distarrank* of mentioners, on almost all days of the study (21). She maintains heavy cooperation with a lot of other Bieber fans, and also was a star of the movie *"Akeelah and the Bees."* This shows how the younger segment of the audience, focused on teenage culture, is densely connected. Another example is a top fan from Slovenia, *@jbieber_fever24*, discovered at 7th position of the longest *distarrank* decreasing runs. Her bio reads, "Hey I'm Tea I'm from Slovenia and I'm 14.I LOVE Justin Bieber!I also like Tay Swift,Miley Cyrus and Sel Gomez!Justin followed me 25/10/2009!" She uses trading shoutouts, ramping up the number of followers, maintaining multiple accounts such as *@JBieber_Babes*, and inviting other fans to participate in all of those activities together as ways to increase her own pagerank.

Chapter 6

Reciprocal Social Capital

Social networks such as Twitter and Facebook increase their mindshare daily, and many online activities are determined by the interactions there. In order to understand the dynamics of social networks we need to identify the key players defined as those participants who are in some way important, influential, and possess significant social capital. The term "social capital" is used in sociology and computer analysis of static networks, however we need to come up with the new and more rigorous definitions in order to address the dynamic nature of our network under study. Our solution is a set of dynamic metrics of importance, called *drank* and *starrank*, which allow ranking over time and in comparison to one's audience (network neighborhood). Using these metrics, we uncover fascinating worlds inside Twitter, such as the Justin Bieber Ecosystem and Brazilian sport journalists and their fans. Building upon the insights from those communities we define our version of Reciprocal Social Capital as an iterative update rule, rewarding those who facilitate balanced and stable communication. Running a complete world emulation with our rules we end up with the social capital distribution which places the hard-talking "middle class" nodes at the top, leading to a new ranking and understanding of the Twitter dynamics.

6.1 Modeling Social Capital Overview

Given a large social network that changes over time, such as Twitter, the question becomes how do we find the key people who are important, influential in some sense, who have a lot of social capital. How do we formally define and measure these terms? Computer and social scientists, economists, operations researchers, and educators have all proposed quantitative and qualitative approaches for describing social capital, and as a consequence of so many approaches the term "social capital" is so widely (over)used that some researchers shy away from relying on it altogether [36]. Furthermore, the three qualities listed often overlap. We define them usefully and distinctly for our networks, applying these new definitions so as to find interesting phenomena such as fan-based economies of ratings and trends, and multi-modal collaboration of hackers thus advancing open-source projects online via social coding.

The modern economy is knowledge-based and knowledge generation closely corresponds to value and wealth creation. The processes in which knowledge is created, refined, and made actionable are increasingly shifting to social networks on the Internet and are embedded in social media sites such as Twitter and Facebook, or are found in cooperation on Wikipedia, the social coding portal GitHub, etc. Certain members and groups are key to the value-creating processes in these networks where people are united by the information they work with. These individuals contributing the most and best knowledge, or processing it in the best way, are recognized as the main contributors and as a result get to set the agenda for the whole group. In fact, a lot of communications are questions for the most respected members, seeking explanation or coordination. It is important to quantify which members gain importance in these mind economies in order to best know how they work.

The quintessential mind economy is the programmers' community. Their industry develops by ideas becoming code becoming startups becoming large companies like Amazon and Google. The resources traded by hackers are most commonly URLs of code repositories on github[1], the open-source social coding portal. Github is organized around `git`, a distributed source code management system (SCM)

[1] http://github.com/

authored by Linus Torvalds and originally used to develop the Linux kernel, now it is a *de facto* platform used to collaborate on open-source projects. While github is used to store and modify code, people working on it often converse on the Internet Relay Chat (IRC) and Twitter. Some of the most active geek communities on Twitter center on advanced programming languages such as *#scala*, *#clojure*, and *#haskell* – these are the "hash tags" used to mark tweets so that they can be found as a group. Coincidentally, these are also the names of the corresponding IRC channels.

Another community with high traffic is what we call Justin Bieber ecosystem, discovered and described in [45]. Justin Bieber is a boy pop-singer phenomenon, originating on the YouTube and spawning an intense following there and on Twitter. Many of his fans are teenage girls with a variation of "bieber" in their Twitter nick and are united in their adoration of `@justinbieber` and pushing him up into the top 10 trending topics on Twitter. In this process, the members learn to increase their own ratings by swapping shoutouts (mentions) and trading shoutouts for follows. An economy of rating-increasing behavior develops. Those who get followed by Justin Bieber increase their standing among the "beliebers" immensely, as do those who organize other fans around better schemes "how to meet Justin." High-intensity group behaviors are key in social dynamics and change, and can be studied for the first time on a coherent social organism of a Twitter community with its drivers and influencers, and the methods they use to direct it.

6.2 Approach

In this thesis we approach the problem of characterizing influence via dynamic analysis of communication graphs. We build a communication graph of mentions in which one user talks with another via public tweets. If `@alice` tweets: "`@bob`, did you see this: http://bit.ly/xyz", she *replies* to `@bob`, who is thus her *replier*, while she is his *mentioner*. While many Twitter analyses and ranking sites focus on the number of followers [37], we prefer to analyze and rank communication as a form of active behavior with its many social implications, manifesting itself similarly in social media networks, email networks, and real life.

Traditional sociology measures versions of centrality on static graphs [8]. Their networks are small and their tools are often Excel spreadsheets and add-ons [20]. By contrast, our data subset consists of a 100 million tweets by 5 million users, over a period of 35 days. We have to take temporal nature of these data into account as a key feature of the model.

First, we propose a measure of importance we call Dynamic PageRank, or *drank* [45]. For every day in the study, we treat the communication graph as a directed multigraph and compute the PageRank of all the nodes present that day. We translate such PageRanks into relative ranks, from 0 to 1, showing where the node stands in the overall sorted list of ranks.

We then define *starrank* as a ratio of a node's *drank* to the average_drank_ of its communication partners. There are also variations where we count a node's repliers and his mentioners either together or separately. Both *drank* and *starrank* can be compared across days even though the number of nodes on Twitter increases every day.

6.3 What is Social Capital

We propose a mathematically well-defined measure of social capital, which we call *Reciprocal Social Capital*. It accrues for those dialogue participants who better maintain their question-answering balance in conversations, in that they do reply to those who address them, and they get replies from those to whom they have talked before. This capital can also favor stronger ties in which you keep talking to your current interlocutors, or it can reward the exploration of new partners. The model is parameterized with weights (rewards) for different kinds of behaviors, and previous capital decays with time.

Most definitions of Social Capital are in fact just other kinds of importance measures. Getoor [49] simply uses the term to denote the number of your co-authors on conference program committees. A good review of the definitions of social capital used in computer science is provided in [53]. We model our *Reciprocal Social Capital* on the social capital of Tuscany villagers who remember how many times they had lent everyday items like salt to each other, versus how many times they got even by borrowing some anchovies, or more salt, in return. A good description of such social capital working in real life is provided in [28]. Every community member in those social-capital-rich areas has a clear mental balance of the favors given and received and this figures in every subsequent social and economic transaction. The communities are cohesive and mutually supportive which enhances the participants' social comfort and quality of life, providing calibrated expectations and thus a safety cushion. We'd like to model social capital in the online communities accordingly.

Our first computational model is "reciprocal" repayment of communications, in which instead of a carefully maintained balance of favors we have a communication network where you repay questions (mentions) by replying (mentioning in return). There is a temporal notion of capital accumulation, along with its decay in the face of inaction.

When looking for influence in social networks, several classes of problems turned out to be closely related to our definition of influence in a community. We addressed them in the additional papers on which this thesis also builds.

In our work on network resilience [46], we considered the question of network structure in as such

it enables the networks to withstand random faults or malicious attacks which are taking out some nodes one by one. We used an application of dynamic graph analysis to examine how the influential nodes can help keep the network together. It is one of the first studies of the effect of a mix of random faults and malicious attacks on a network, and our analysis compared behavior of differently structured networks, such as scale-free or random, under different destructuring scenarios.

In addition to Twitter, we studied a sensor-based social network, resulting from the MIT Reality experiments [23]. A fundamental question in dynamic systems is the agents' identity. We address it [44], and were able to identify a majority of the MIT Reality participants from just about 10 hops in their cell phone traces. Social importance of the subjects is related to their patterns of action (motion) and resulting interactions.

6.4 Reciprocal Social Capital Definition

Repliers of a node are the addressees of its tweets. *Mentioners* of a node are those who tweet to it. If a tweet from `@alice`*mentions* `@bob`*, then* `@bob` *is a replier of* `@alice` *and* `@alice` *is a mentioner of* `@bob`*. To recap, replier of a node is someone that that node repliers _to*. In other words, from a node's perspective, repliers are out-degree, mentioners are in-degree. We define the following variables.

symbol	definition
S_v^t	Social Capital of node v at time t. Superscript t generally denotes "by time t." Specifically *during* time step t is denoted as @t.
$G^t(V, E)$	graph G with nodes V and edges E
$w_{uv}^{@t}$	total weight of directed edges $u \to v$, i.e. the number of tweets from u to v during time step @t
W_{uv}	total number of undirected edges between u and v: $W_{uv} = w_{uv} + w_{vu}$
B_{uv}	Balance of back and forth tweets from u to v: $B_{uv} = w_{uv} - w_{vu}$
M_u	$\{v \mid w_{vu} > 0\}$, i.e. the mentioners of u
$R_u^{@t}$	$\{v \mid w_{uv}^{@t} > 0\}$, i.e. repliers of u specifically during the timestep @t
$O_{uv}^{@t}$	outgoing activity of a node rewarded by social capital at timestep t
$A_{uv}^{@t}$	incoming activity in this cycle rewarded just for mentions (all)
$B_{uv}^{@t}$	incoming mentions in this cycle repaying previous replies (balance)
α, β, γ	model parameters

$$O_u^{@t} = \frac{1}{\sum_{V^{t-1}} O^{@t-1}} \sum_{v \in M_u^{t-1} \cap R_u^{@t-1} | B_{uv}^{t-1} < 0} |B_{uv}^{t-1}| w_{uv}^{@t-1} W_{uv}^{t-1} S_v^{t-1} \tag{6.1}$$

$$B_u^{@t} = \frac{1}{\sum_{V^{t-1}} B^{@t}} \sum_{v \in M_u^{@t-1} | B_{uv}^{t-1} > 0} B_{uv}^{t-1} w_{vu}^{@t-1} W_{uv}^{t-1} S_v^{t-1} \tag{6.2}$$

$$A_u^{@t} = \frac{1}{\sum_{V^{t-1}} B^{@t}} \sum_{v \in M_u^{@t-1}} w_{vu}^{@t-1} W_{uv}^{t-1} S_v^{t-1} \tag{6.3}$$

$$I_u^{@t} = \gamma B_u^{t-1} + (1-\gamma) A_u^{t-1} \tag{6.4}$$

$$S_u^t = \alpha S_u^{t-1} + (1-\alpha)(\beta O_u^{t-1} + (1-\beta)(\gamma B_u^{t-1} + (1-\gamma) A_u^{t-1}) \tag{6.5}$$

Some notes on the definitions. O_u^t is the node u's output gaining social capital, thus we want to reward those who redress an imbalance of input and answer those who mentioned you more than you had replied to them. The summation is defined exactly over those with whom you have a deficit in replying: $v \in M_u^{t-1} \cap R_u^{@t-1} | B_{uv}^{t-1} < 0$. It means the target node mentioned you at some point prior, you replied to it in this cycle, and before that, you owed it a reply since it tweeted more to you than you did to it. For each such deficit node you replied to, finally, we multiply the number of replies in that cycle, $w_{uv}^{@t-1}$, by the balance you owed, $|B_{uv}^{t-1}|$, the value of the relationship, W_{uv}^{t-1}, and the importance of the replier S_v^{t-1}. We normalize the O's so that they all sum to 1, and reward each node proportionally to the value of the redress in the reply imbalance it actively contributed in this cycle.

Similarly, I_u^t is the node u's input worth of social capital. We generally consider all input as "good," since we do not distinguish between bad publicity, or good one, but we distinguish mentions redressing the mentioners' own deficit with us as worthing more than just any mention. Those repaying mentions we reward with a multiplier for the balance owed in addition to the usual cycle contribution, relationship value, and the mentioner's social capital.

Note that in the output, we do not have a general activity term for all replies even those not redressing an imbalance as we do in the second term of the input, thus we don't reward random replying, and users do not get more social capital by just "cold-calling" everybody *en masse* .

It is easy to see that such a definition of social capital allows for an iterative economy by launching the update rule defining S_u^t in terms of S_u^{t-1} as shown in the last formula above. The core questions which may allow for more quantitative treatment with our metrics:

- Why do people twitter? What is the utility and can it be captured by a form of social capital? Can a single definition suffice for all members of a network? Do "beliebers" differ from hackers and how, with respect to their forms of social capital and behaviors increasing it?

- How can you increase your social capital or influence in the fastest, and most robust ("honest," irreversible) way? How can we distinguish fake influence from the real thing?

6.5 Capital-Based Mining

Given our metrics of influence and social capital, we explore interesting individuals from our large Twitter data set and interpret what these metrics mean in real life.

The first results of our full-world Reciprocal Social Capital emulation show a new kind of nodes at the top, we call "the middle class of social networks." These are the people with a certain amount of followers from a 100 to a 1,000, thus weaving a web of intense dialogues with other such nodes. These users provide the core of the ongoing communication, promoting trends and stars in the rankings, or conversely bringing the rankings of uninteresting subjects down by not talking about them, and integrating other nodes, especially the beginners into system-wide dialogues. We believe our metrics are a more useful tool than the traditional PageRank in that they reflect the dynamic nature of the conversational networks and highlight the groups making lasting ongoing contributions.

A typical example of this is beauty industry. Two owners of Singapore beauty salons come on top in terms of continuous conversation threads kept the longest. They discuss wedding photos, haircuts, design, etc. They have a constant and purpose-driven market, thousands of followers each, and discuss news of their industry. In this way they keep tabs on their market niche, satisfy their followers with specific goals, and stay on top in the ratings.

This illustrates one of the differences in the utility of tweeting for different classes of users. Celebrities, those most followed in a typical definition, often get on top by projection from the real world, YouTube, etc. On the contrary, their top fans and other in-Twitter "phenomenal" users have to generate interest through their tweets alone to keep their high ranks.

While spammers achieve a short-term notoriety by mass following hoping for automatic follow-backs we do not register those numbers since we look for real conversations and mentions. Top celebrities like `@donniewahlberg` and `@justinbieber` understand this and cultivate their fan base, encouraging top fans once in a while. The timing feature of our metrics smooths over the spikes typical in social networks and reveal true value.

Chapter 7

Success is Earned

7.1 Accidental Influentials, or Not?

In the 1955 seminal book "Personal Influence," Katz and Lazarsfeld proposed the concept of influentials in social networks, propagating and filtering media streams in their communities. Although the focus of their study was information diffusion, Katz and Lazarsfeld for the first time fused media studies with dynamics of social groups at local level, and they identified many features of their opinion leaders, whom the called "the influentials" for being important community members by many criteria.

Watts and Dodds, in their 2007 paper "Influentials, Networks, and Public Opinion Formation" [65], popularly known as "Accidental Influentials" [66], showed that for a diffusion model with cascades it is not necessary to have influentials to excite a typical network. It is mostly the average low threshold on excitability of a majority which decides whether a full cascade will occur or not, instead of who started it, regardless if that person was an influential or not. Despite the specific setting, Watts and Dodds's idea that influentials are somehow "accidental" took a life of its own, and was reported by Harvard Business Review as a number one idea on list of the influential ideas for the year (presumably not accidental).

Since we studied influence in real social networks, for our family of metrics we asked if our influentials are accidental, or not? And what can this question possibly mean for a broad class of definitions

of influence, of which there are many?

Let us say that we pick some definition of influence which suits a particular problem at hand. The only general requirement for an influence measure we will impose is that it establishes an order or ranking among all of the people in the study, and that the "top influentials" are simply the most highly ranked people in this list. A consideration of influence then can focus on the very top, or look broadly at the classes of influence, similarly to the rich, the poor, and the middle class in a human economy.

Our question then can be made more concrete as follows. If influence is a function of behavior and connectivity can we generally say that somebody becomes influential due to their "intrinsic" qualities such as efficient behavior, or, on the contrary, is their influence mostly a product of luck, such as the propitious time and place of that individual's appearance in the network?

7.2 Taught by Randomness

One way to answer a philosophical question about whether something in this world is random or not is to create parallel worlds with slightly different rules, mix them with the real one at some point, and evaluate both according to the same metrics of interest for the processes we care about. When creating the new worlds by varying one feature at a time we can hope to tease apart the features which make a difference in the course of events. In this world, there's only one Twitter, albeit its character is constantly changing. For any given dataset, we have only one order of influentials for any one metric. However, we can separate the components of behavior which contribute to that influence, and, preserving some part of the network position and some of the behavior, alter the other parts in order to see how the influence is affected. By teasing the starting conditions and the behaviors apart, we can hope to get to the root of our main question.

We introduce a series of strategies to grow the graph from some point onwards by using a combination of behaviors in which we reward in the actual graph, as the rewards in the random ones. We control dynamic characteristics by

- adding the same new users on the same days as in the original graph

- every user having the same outdegree for each day as in the original graph

Thus, after a certain point in time, we only rearrange the sinks of some edges in the simulation. We use three classes of strategies to attach the edges:

- local utility, which is defined as optimizing exactly the same reward which is computed via social capital and then used to rank influence in the first place, but only for the action under control of a node, i.e. to whom it will reply (not from whom it will get the replies). The latter would be a global optimization, while the former is still an individual one.
- local friends-of-friends, in which this strategy attaches to a friend of a friend with attractive characteristics, likely to be seen due to a large number of his/her own friends, mentions, or their own reciprocal social capital.
- global – this strategy approximates following users notable via global phenomena, such as celebrities or trends, or efficient communicators with overall leading social capital rank.

In each simulated world we mix these strategies via jump probabilities thus creating a composite behavior and hopefully capturing some subset of the real world. We see who wins in the simulated world, and compare the winners in each position, or rather class, to the real ones for that day and class. We seed the simulated world with the real one, hence establishing communication balances, friends-of-friends networks, and social capital distributions in advance, in different proportions. If starting conditions are key, and simulation strategies are reasonable, we'd see the original winners keep their place in large numbers. By varying the degree of randomness in the simulated strategies, we can see how quickly the winners dissipate from either the real or the simulated world. If the simulated world is governed by its own rules which are self-consistent, we would see the same winners across days. When, once emerging, the winners will persist for the duration of the world, it will mean their behavior is consistent with the rules of that world.

Additionally, we can alter only a specified subset of users, such as those in a given sets, or "buckets," of classes. Using this fine-grained filter on top of a given strategy, we can see which classes fit the strategy better, and if they don't, we can see does it make a difference.

The remainder of this chapter is structured to review

- Reciprocal Social Capital, in which we introduce our definition of social capital and the iterative algorithm to compute it on a dynamic communications graph.
- The methodology of influence studies, which outlines the fundamentals of our parallel world simulations and extracting regularity through overlaps with reality
- Simulation strategies that show how the parallel worlds are constructed
- The actual simulations relying on these strategies
- Simulation tables, which form cross-linked hierarchy of three levels of key results

7.3 Methodology of Influence Studies

We use our reciprocal (conversational) social capital as the measure of influence. We define and compute it iteratively for each day. Once the capital is computed for each user in a time period, we rank all users according to their social capital for that period (daily). A value is thus associated with each user, and the top-ranked users are the most influential in this metric.

A fundamental feature of our study that is similar to Katz-Lazarsfeld and different from Watts-Dodds is the reliance on conversations as pathways of influence. In fact, this tradition in sociology originates with Tarde, who called attention to the *statistique de conversation* [62] as a means of quantitative study of public opinion.

What does it actually mean to have influence according to the reciprocal social capital metrics? One gets to the top in this metric by being attentive to one's balance of communication and by maintaining a high absolute value of dialogues with other partners who also have high social capital, that is, by being an effective communicator and maintaining a good standing in a community of other effective communicators.

The form of influence we consider directly relates to the communication pathways. Our influencers play a role in the bulk of actual conversations, which is shown in the volume metrics. Diffusion models, such as Watts-Dodds, define influence as an ability to propagate information through the social graph specifically igniting cascades which affect a bulk of the network. We submit that prior to any such diffusion beginning that the channels of communication must be established over which the diffusion will be taking place. A good analogy is a railroad. The trains of thought must run over the rails of communication links. Our influencers build the communication links and carry most of the discourse. They provide the liquidity of conversations. Bakshy, Watts et al. [4] look at the shortened URL spreading via follower graph, and show that the effect of the influentials is inconclusive, making them again seem accidental. But following others is not a typical human communication activity – it's more like RSS subscription, a passive scan with no evidence of actual reading or other behavior. On the other hand, conversations, in all its forms – face-to-face, phone, email, forum threads, comment

feature	#tweets	%tweets	%replies
all tweets	92,229,974	100%	
replies	29,490,600	32%	
all URLs	20,476,482	22%	
reply URLs	1,417,664		5%
all questions	12,021,562	13%	
reply questions	5,565,838		19%

Table 7.1: Twitter URL and question statistics overall and in replies for our 2009 dataset. URLs spread much less via replies than overall, while questions are more common in replies.

discussions, Twitter replies – share the same dynamics and expectations as any dialogue in any media found throughout human history. When we look how much URL diffusion occurs along the reply links a different picture emerges.

Conversations are rarely about diffusion. Less then 5% of all replies contain a URL, while near 19% are questions (defined simply as strings containing a question mark). These statistics are shown in Table 7.3.

Since we base our study on conversations only a node shows up in our graph when a reply edge first appears originating or ending in it, such as when a user replies to somebody or someone replies to a user. We record the order of those edges exactly. We enable playback cycle by cycle (day by day), of any social graph. For every new day we record which users appeared first in that cycle, and how many edges each new user has issued.

We replay those edges literally, thus recreating the original graph for one more day at a time, day after day, for all days in the study. We can compute any iterative function on the nodes (and potentially edges), thus we compute our reciprocal social capital as a function of the previous day's capital, balances of communications existing so far, and the fact that an edge was established in this cycle. The balances and the capitals are all adjusted together, transactionally, progressing in a discretized time from one cycle to another.

7.4 Simulation Strategies

There are three kinds of strategies we utilize:

- Global
- Friends-of-Friends, or FOF
- Local Utility

Principles of Simulation

Instead of replaying the edges we can attach them somewhat differently, thus perturbing the original growth process. We employ several different kinds of such simulations which are described in detail below. We preserve the original number of outgoing edges for each user in each cycle, but do not control for a similar distribution of receiving edges. Here we rely on the fact that replying is an active decision, while receiving a reply is outside of the receiver's control, generally speaking — even though we reward users for getting back the replies owed them, as proving capable of prudent dialogue management.

When simulating using any given strategy we can actually start later in the organic growth process. For all of our key simulation techniques we start following the playback from scratch, then do another simulation following a period of 1 week of the actual graph growth, then start simulating after 2 weeks, etc. In order to achieve a smooth transition to the simulation we compute all of the features required by a specific simulation from the end state of the original graph at the time of the hand-off. Our strategies can then be mixed within a single simulation depending on jump probability parameters and on whether the data for more complex local computations are actually available at a given day for a given user.

GlobalUniform

Given a user *fromUser* and a new edge to issue from him, the user *toUser* is simply an equiprobable choice among all users existing at the moment. If the edge ends up going to a user who only now appears in the graph we note that a user addition with initial social capital actually occurs prior to the edge addition so the original *toUser* will have a chance to receive the edge as well as those present already.

GlobalMentions

Here, we consider how many mentions all the eligible *toUser*s have already received prior to this cycle and then we generate an attachment with a probability proportional to the *toUser*'s total number of edges received so far. We call this total edge count *byMass*, as opposed to a distinct mentioner count *byUser* (see 7.4).

GlobalConstants

Our *GlobalConstants* strategy takes a list of values, such as the actual social capital numbers expected in this cycle for the original graph (*dreps*), and associates them with the same nodes as in *dreps*. We then pick a node in this graph-cycle proportional to that given probability.

GlobalRealValues

The *GlobalRealValues* strategy computes a real-valued function, such as the actual social capital obtained in this simulation, and picks a node proportional to it. The difference in regards to *GlobalConstants* is that while the latter takes a value from a predefined list here the value is actually computed, hence the growth simulation is inseparable from the iterative social capital computation via playback. These and subsequent simulations are performed together with social capital computation while the preceding global strategies are achieved by composing the graph first for all days, and then running it through

the capital computation separately (as with the real graph which already exists before we start playing it back and computing its capital distributions).

FOF (Friends of Friends)

These are local strategies, in which every *fromUser* looks at his list of friends and picks a friend of a friend to attach to. This strategy corresponds to a common scenario where a *fromUser* is talking to his/her friends and sees a friend mention his/her friend, which, in turn, causes an attachment. We perform two types of FOF attachments.

FOFUniform

Available in this approach there is an equiprobable attachment to all FOFs. Only the number of friends matter for each friend. First, a cumulative mass of such FOF numbers is assembled for each user, and a friend is picked proportionally to his/her number of friends, then, a friend of such friend is chosen equiprobably.

FOFMentions

Here we look at the overall number of mentions each friend of a friend has accumulated and pick one proportionally. For each *fromUser* there exists an array of friends and the total number of mentions each of such friend's own friends had generated, from which we pick a friend in proportion to the total number of second-level mentions . Among that friend's friends we pick one proportionally to his/her own overall number of direct mentions.

Local Utility

This strategy concerns influence-seeking behavior of a user directly. We apply exactly the same utility function here that is used to compute the reciprocal social capital once the edges are in place — or, rather, its outgoing part, which computes the possible rewards for replying to someone to whom we owe a balance of communication. This is a local optimization for the *fromUser*.

Bucketed Selection

Bucketed selection is not a class of strategies in itself but rather a meta-strategy which can be applied to any of the above composite attachment strategies. In our original simulations, when we change the attachments after a point in time we do so for all users . We assume everybody will behave in a similar fashion in principle but will end up with a different social capital due to their different energy level, starting communication balances, etc.

In bucketed replier simulations, or *breps*, we specify which buckets are going to be simulated, and the remaining ones are left intact. We refer to the buckets by their number, top to bottom, 1-based, which also coincides with the log_{10} of the bucket size, ranging from 1 to 7. A specifier such as *56f* means "buckets 5 and 6, keep them = *false*," which translates to simulate (redo) buckets 5 ad 6, while keeping buckets 1,2,3,4,7 intact, copying them from the original *dreps*. Suffix *t* would mean "keep = *yes*", so that *56t* means "preserve buckets 5 and 6 from the original *dreps*," and simulate buckets 1,2,3,4,7 according to the underlying primary strategy.

Separating individual buckets and their groups allows us to analyze the effects of modifying behavior by class as it is being generated. Social capital and its subsequent bucketed hierarchy is recomputed every cycle, hence modifying attachments per bucket, which we do prior to computing the capital in a cycle, is reflected immediately. This action leads to a somewhat different bucketing and new modifications in the affected buckets.

Extra Strategy Features

Extra strategy features can aid in generating more simulations. Those listed here were considered but did not add much diversity.

We also implemented a *byUser* version where instead of the total number of incoming edges we consider the total number of distinct users who replied to a given user (collapsing the incoming edges from the same *fromUser*), but we did not find any interesting distinctions from the default *byMass* version.

Instead of the probability of attachment linearly proportional to the total number of mentions (or

mentioning users), we could also transform to a Gaussian (and fit one first over the general population). Various kernel transforms are possible.

We could theoretically optimize over a subset of potential incoming edges, but that would involve summation over *fromUser*s and be in a new class of global strategies, incompatible with the ones above, which are limited to *fromUser*.

7.5 Actual Simulations

Using the above strategies as basic elements, we combine them with one or more jump probability parameters into actual simulations as discussed below. The original dynamic graph which records all of the replies across the *fromUser* → day → *toUser* dimensions is called *dreps*. All other simulations lead to a similarly structured dynamic reply graph which we give a short root name in order to distinguish the simulation class. A suffix is then used to show the combination of the parameters selected and the week from which the simulation continues the actual *dreps*. For every root name we list all of the suffixes computed for that root class except for the week designator.

Global Uniform Replies — ureps

Ureps are generated using the *GlobalUniform* attachment only. They comprise a stochastic graph with a given set of nodes and predefined outdegrees for each cycle. Each *ureps* graph leads to a distribution of social capital, ranking, buckets, and bucket-based comparisons, which serve as a baseline for more elaborate simulations.

Global Metions Repliers — ereps

Ereps are generated using the *GlobalMentions* strategy only. They are getting fairly accurate estimating capital, especially when started not form scratch but mixed after a week or more of *dreps*.

Global Constants — creps

Creps are generated using the *GlobalConstants* strategy only. We use the actual social capitals from *dreps* as expected values. Despite the artificial character of this setup — attachment is proportional to a prescribed, not earned, social capital — the fact is that we obtain a distribution with a similar ranking and bucketing, which leads to a remarkable result of reproducible middle and upper middle classes, as discussed in Results, 7.8.

Global Real Values — rreps

Rreps are generated with the *GlobalRealValues* strategy only, using the actual social capital as computed on the fly — but there are two parameters related to the way the capital is computed in comparisons at different time points. Our Social Capital, defined in Chapter 6, uses exponential decay by multiplying the previous value by 0.1 each day. Every day there is always a wave of newcomer users who are in a capital-advantaged position only because they are new. When comparing capitals we usually downgrade any user with less than 7 days of history to the minimum possible capital in our study, $1e-35$ (across 35 days). However, when used as an interval, determining the proportionality of attachment probability, this makes attachment to a new user highly unlikely. Thus we also experiment with a higher minimum-capital value, such as $1e-7$. Generally, the parameters for mature social capital comparisons are *minDays* and *minCap*, applied as follows:

- If the user exists for more than *minDays* we use his actual capital
- Otherwise, we use *minCap*

For *rreps*, we tried both *minCap* value of $1e-35$ and $1e-7$, with *minDays* = 7 in both cases. The corresponding graph groups are called *rreps* and *rreps7m*.

Local Utility with Global Jump — lreps

These simulations take a single jump probability parameter, *jumpProb*, and proceed as follows. For every new edge, with a probability *jumpProb*, we jump to a global attachment. This can be either *GlobalUniform*, or *GlobalMentions*, as specified by a global strategy parameter. When we don't jump we attach to the user that maximizes our reward for returning owed balance of reciprocal communications. When there's no balance to maintain, i.e. there is no incoming edges, we jump to global.

The *lreps* simulations are named with a prefix *lj*, followed by the decimal part of the *jumpProb* parameter, and the global strategy letter, with u for uniform and m for mentions.

Consider *lj2m* — here we have *jumpProb* = 0.2, i.e. do local utility attachment with probability 0.8, otherwise global by mentions with the remaining probability 0.2.

Local Utility with Local and Global Jumps — freps

These simulations use two jump probability parameters, *jumpProbUtil* and *jumpProbFOF*. They show how likely it is that we will jump away from utility attachment and whether we will perform a *FOF*-based attachment, or jump to a global one. In addition to these jump probabilities there are also two strategy parameters — *FOF* and *Global*. The *freps* simulations are named with a prefix f, followed by the global block mark g, then by a decimal part of the *jumpProbUtil* probability, then the global strategy designator, u, m, or c (for *GlobalUniform*, *GlobalMentions*, or *GlobalSocCap* attachments, respectively), then the *FOF* block mark f, the *jumpProbFOF* decimal part, and the *FOF* strategy designator, u, m, or c (for *FOFUniform*, *FOFMentions*, or FOFSocCap, respectively), potentially followed by the decimal digits of the minimum capital assumed for those users with less than *minDays* (7) of maturity and its suffix m. No m means the standard value, $1e-35$, is used; *0d* means *minDays* = 0, i.e. the actual, unadjusted capital is always used.

E.g., the simulation

fg2uf05c7m1wk

will try to do local utility attachment with probability 0.8 — unless it jumps away from utility with *jumpProbUtil* = 0.2, — then do a *FOFSocSap* attachment with probability 0.95, or, jumping from it with *jumpProbFOF* = 0.05, to perform a *GlobalUniform* attachment in the end.

7.6 Simulation Tables

The full list of all simulations and links to their result tables follows. Each simulation is first analyzed for interday bucket stability, *srates* (Figures 7.9–2.3), and then for overlaps with reality, *overx-dreps* (Figures 7.1–7.4), in its own week-shifted instances, *overx-self* (Figures 7.5–7.8), or in differently randomized runs. We look at volumes per bucket, both outgoing replies (Figure 7.12) and incoming mentions (Fig-

ures 7.13–7.16). Starranks by mentions (Figures 7.17–7.20 tell us about the relationship of a user and his mentioners, per bucket, while starrank by replies (Figures 7.21–7.24) shows how users associates themselves actively with others. Bucket to bucket communication, by replies, tracks how people in one class talk to those above, below, or on their own level (Figures 7.25–7.36), and mentions between buckets are segmented by class relationship as well (Figures 7.37–7.45). Finally, we look how our Reciprocal Social Capital correlates with our own Skew metric, or how "politician-like" a user is when returning his base's contributions proportionally, using Kendall's τ (Figures 7.47–7.50).

Each of the *srates* and *overx* comparison is done across all buckets and days, producing a 35×7 (days $\times$ buckets) table. We show the first 4 instances of each type, the first simulated from scratch, and three others seeded with one, two, and three weeks of *dreps*, respectively. Due to the reciprocal social capital break-in period of one week and edge effects of the cutoff, we drop the first week and the last two days of study, showing 27 days of each.

These full tables occupy the bulk of the appendix. Several more categories are available online. In order to compress the data displays, we compute two aggregate lines of each table as a 7-bucket-length vector, summarizing

- averages of all non-1.0 cells
- medians of all non-1.0 cells

The 1.0 cells appear as a result of comparing identical *dreps*-based buckets in week-seeded, *dreps*-mixed simulations. In such cases, we are interested in what happens next, so we look at the actual simulated cells only. These summary vectors are assembled as rows of summary tables, *averages* and *medians*, for each class of simulations. When we list all of the relevant full and summary tables below we cross-link to the full table and summary one for each simulation class. We list the lines in the summary tables containing the rows summarizing the corresponding full tables. The summary tables have row numbers for quick reference, as well as the respective simulation names as row names.

ureps list

There's only one conceptual kind of *ureps* simulation. We run it several times, initializing the random number generator differently every time. Each group contains the base, suffixed 0, and 4 with increasing number of weeks of seeding with *dreps*, from *1wk* to *4wk*:

- *ureps*
  - *ureps0*
  - *ureps1wk*
  - *ureps2wk*
  - *ureps3wk*
  - *ureps4wk*

From now on, we'll show only the roots of the simulation groups, like this:

- *ureps*
- *urepsB*
- *urepsC*

ereps list

Here we list the roots of other global-only simulation groups:

- *ereps* — global mentions-based attachment only
- *creps* — global prescription-based attachment only
- *rreps* — global capital-based attachment only, standard capital maturity
- *rreps7m* — global capital-based attachment only, 1e-7 adjusted capital before 7 days maturity
- *rreps0d* — global capital-based attachment only, no capital maturity

lreps list

- *lj1m* — local utility attachment with probability 0.9, otherwise (0.1) jump to global attachment by mentions
- *lj1u* — local utility attachment with probability 0.9, otherwise (0.1) jump to global uniform attachment
- *lj2m* — local utility attachment with probability 0.8, otherwise (0.2), jump to global attachment by mentions
- *lj2u* — local utility attachment with probability 0.8, otherwise (0.2), jump to global uniform attachment
- *lj5m* — local utility attachment with probability 0.5, otherwise (0.5) jump to global attachment by mentions
- *lj5u* — local utility attachment with probability 0.5, otherwise (0.5) jump to global uniform attachment

freps list

- *fg2cf05c* — local utility with probability 0.8, jump to FOF with 0.2, then capital-based FOF with probability 0.95 or jump to global, then capital-based global attachment with probability 0.05, standard capital maturity
- *fg2uf05c0d* — local utility with probability 0.8, jump to FOF with 0.2, then capital-based FOF with probability 0.95 or jump to global, then uniform global attachment with probability 0.05, no capital maturity
- *fg2uf05c* — local utility with probability 0.8, jump to FOF with 0.2, then capital-based FOF with probability 0.95 or jump to global, then uniform global attachment with probability 0.05, standard capital maturity
- *fg2uf05c7m* — local utility with probability 0.8, jump to FOF with 0.2, then capital-based FOF with probability 0.95 or jump to global, then uniform global attachment with probability 0.05, 1e-7 adjusted capital before 7 days maturity
- *fg2uf05m* — local utility with probability 0.8, jump to FOF with 0.2, then mentions-based FOF with probability 0.95 or jump to global, then uniform global attachment with probability 0.05, standard capital maturity
- *fg5mf1c* — local utility with probability 0.5, jump to FOF with 0.5, then mentions-based FOF with probability 0.9 or jump to global, then uniform global attachment with probability 0.1, standard capital maturity
- *fg5mf1c0d* — local utility with probability 0.5, jump to FOF with 0.5, then mentions-based FOF with probability 0.9 or jump to global, then uniform global attachment with probability 0.1, no capital maturity
- *fg5mf1m* — local utility with probability 0.5, jump to FOF with 0.5, then mentions-based FOF with probability 0.9 or jump to global, then mentions-based global attachment with probability

0.1, standard capital maturity

- *fg5mf1u* — local utility with probability 0.5, jump to FOF with 0.5, then uniform FOF with probability 0.9 or jump to global, then mentions-based global attachment with probability 0.1, standard capital maturity
- *fg5uf1m* — local utility with probability 0.5, jump to FOF with 0.5, then mentions-based FOF with probability 0.9 or jump to global, then uniform global attachment with probability 0.1, standard capital maturity
- *fg5uf1u* — local utility with probability 0.5, jump to FOF with 0.5, then uniform FOF with probability 0.9 or jump to global, then uniform global attachment with probability 0.1, standard capital maturity
- *fg8cf05c* — local utility with probability 0.2, jump to FOF with 0.8, then capital-based FOF with probability 0.95 or jump to global, then capital-based global attachment with probability 0.05, standard capital maturity
- *fg8uf05c0d* — local utility with probability 0.2, jump to FOF with 0.8, then capital-based FOF with probability 0.95 or jump to global, then uniform global attachment with probability 0.05, no capital maturity
- *fg8uf05c* — local utility with probability 0.2, jump to FOF with 0.8, then capital-based FOF with probability 0.95 or jump to global, then uniform global attachment with probability 0.05, standard capital maturity
- *fg8uf05c7m* — local utility with probability 0.2, jump to FOF with 0.8, then capital-based FOF with probability 0.95 or jump to global, then uniform global attachment with probability 0.05, 1e-7 adjusted capital before 7 days maturity
- *fg8uf05m* — local utility with probability 0.2, jump to FOF with 0.8, then mentions-based FOF with probability 0.95 or jump to global, then uniform global attachment with probability 0.05, standard capital maturity

- *fg5cf1cA* — local utility with probability 0.5, jump to FOF with 0.5, then capital-based FOF with probability 0.9 or jump to global, then capital-based global attachment with probability 0.1, standard capital maturity, run A
- *fg5cf1cB* — local utility with probability 0.5, jump to FOF with 0.5, then capital-based FOF with probability 0.9 or jump to global, then capital-based global attachment with probability 0.1, standard capital maturity, run B
- *fg5cf1cC* — local utility with probability 0.5, jump to FOF with 0.5, then capital-based FOF with probability 0.9 or jump to global, then capital-based global attachment with probability 0.1, standard capital maturity, run C

Bucketed simulations are even more expensive to compute than the most involved, three-stage composite *freps*. Hence we perform a complete set of bucketed simulations based on the same underlying *freps* simulation, namely

- *fg5cf1c* – local utility with probability 0.5, jump to FOF with 0.5, then capital-based FOF with probability 0.9 or jump to global, then capital-based global attachment with probability 0.1, standard capital maturity

The following bucketing is performed:

- *fg5cf1cb1f* – simulate bucket 1, preserve all others
- *fg5cf1cb1t* – keep bucket 1 and simulate the rest
- *fg5cf1cb2f* – simulate bucket 2, preserve all others
- *fg5cf1cb2t* – keep bucket 2 and simulate the rest
- *fg5cf1cb3f* – simulate bucket 3, preserve all others
- *fg5cf1cb3t* – keep bucket 3 and simulate the rest
- *fg5cf1cb4f* – simulate bucket 4, preserve all others

- *fg5cf1cb4t* – keep bucket 4 and simulate the rest
- *fg5cf1cb5f* – simulate bucket 5, preserve all others
- *fg5cf1cb5t* – keep bucket 5 and simulate the rest
- *fg5cf1cb6f* – simulate bucket 6, preserve all others
- *fg5cf1cb6t* – keep bucket 6 and simulate the rest
- *fg5cf1cb7f* – simulate bucket 7, preserve all others
- *fg5cf1cb7t* – keep bucket 7 and simulate the rest
- *fg5cf1cb15f* – simulate buckets 1 and 5, preserve all others
- *fg5cf1cb15t* – keep buckets 1 and 5, simulate the rest
- *fg5cf1cb56f* – simulate bucket 5 and 6, preserve all others
- *fg5cf1cb56t* – keep bucket 5 and 6, simulate the rest
- *fg5cf1cb67f* – simulate bucket 6-7, preserve buckets 1-5
- *fg5cf1cb67t* – keep bucket 6-7, simulate buckets 1-5
- *fg5cf1cb567f* – simulate buckets 5-7, preserve buckets 1-4
- *fg5cf1cb567t* – keep buckets 5-7, preserve buckets 1-4

7.7 Evaluation

Once the simulations are generated (with a possible initial mix-in of the real history) and their dynamic graphs and social capitals are computed we evaluate the hierarchy of influence for every day, every class, and compare the results within and between simulations.

Buckets

Financial capital leads to a power-law hierarchy. A small minority controls an overwhelming majority of the financial wealth [22]. As shown by George Kingsley Zipf [67], almost any man-made ranking leads to a power-law distribution of set size vs. rank, now known as the Zipf law. Based on this structure, we study our social capital distribution in terms of buckets of exponentially increasing sizes. For simplicity, we choose the bucket sizes as the powers of 10. Since we have about 5 million users total by the end of the study (35 days), our bucket sizes are shown in table 7.7.

We now discuss our analyses in detail and present the key findings. Several thousand tables are produced from real world data and simulations based on them, supporting our story, available online in a cross-linked PDF.

Ranking

A given a set of capitals for all users in a day is sorted in descending order, and grouped together by users with equal capitals. Each *arank* ranking position is occupied by such a list, in which all users in a list have the same rank and any two different lists correspond to different ranks.

Once the aranks are established, we fill the rank buckets starting from the top one, of size 10. We add users from arank lists (sorted in descending order) until a bucket is filled. If adding the next arank list will overflow the bucket, we push the list down the exponentially larger buckets until a bucket wide

10	100	1,000	10,000	100,000	1,000,000	10,000,000

Table 7.2: We use bucket sizes which are powers of 10. These buckets are populated for our 5 million users.

enough to fit the list is found. If any intermediate buckets are skipped they will remain empty and filling will continue from the last bucket.

Classes

Once the buckets are computed, they establish the classes of influence. The first three buckets contain the rich — the top 10 top users, the next 100 celebrities, and the 1,000 elite users. The largest bucket contains the poor masses. The preceding bucket of size 1M is our middle class, as will be shown by various analogous metrics, with the still earlier bucket, of size 100K, corresponding to the upper middle class. Our major finding is that the middle class is carrying the bulk of the conversation and is effectively replicated with our reciprocal social capital measures in *rreps*, *lreps*, and *freps* simulations.

Staying Power

For every kind of a dynamic graph, real, synthetic, or mixed with social capital computed daily we now have daily buckets that reflect the classes of capital rank. If our metric is continuous and people's behavior is meaningful with regard to our metrics, we should see the membership of those buckets rotate at a reasonable low rate. The staying power metric compares the set membership of each bucket and finds the intersection of today's and yesterday's sets. Given that the ith bucket for dth day is B_i^d, the staying power in that bucket from day d to day $d+1$ is

$$\frac{|B_i^{d+1} \cap B_i^d|}{|B_i^d|}$$

Overlaps

For two different simulations we can compare how similar the buckets are by computing the bucket overlap between respective buckets for respective days, using the same formula as for staying power — except here, *bucket1* and *bucket2* are not buckets in the same positions from consecutive days of the same simulation, but from same day and position in two different simulations. This is the primary way

to see how well the original $dreps$ classes are reproduced by a simulation, and also to check whether the same simulation, shifted by weeks, is consistent with itself.

Overlaps with *dreps* show how well we can reproduce the actual social capital distribution. Overlap between *dreps* and *ureps* is a baseline of what we can expect in a random case. The majority of users are in the poor bucket where the most overlap does occur.

We also compute overlap between simulations from the same class, but mixed at different weeks — e.g., for class X, we compute overlaps

$X0$ and $X1wk$

$X1wk$ and $X2wk$

...

These simulations differ only by the starting conditions, — how much of the original *dreps* was used to seed them. The week-shifted overlap will show how much the buckets results depend on the starting conditions as opposed to the simulation-specific ones.

Overlaps with different runs of the same class are computed in the same way as overlaps with different simulations – buckets in the same positions for the same days are intersected, and the ratio of cardinality of the intersection to the left bucket size is taken. This overlap shows how stable each simulation class is.

When we compute an overlap between two different simulations, we take a series of intersections of every two respective buckets. We can then establish the staying power of those members in the intersections across days. If there is a stable core persisting throughout such intersections it would point at a regularity in this process and vice versa.

Volume per bucket

A very important characteristic of classes in discourse is which share of talking is actually carried out by each class. Hence we compute volume per bucket per day in two forms:

- Absolute, which simply counts the replies issued from each bucket
- Relative, which represents the fraction of all replies had originated in each bucket

We compute volumes for replies and mentions separately. In this way we can see that our middle class carries the bulk of the communication in proportion to its own size. In the original *dreps*, the 1 million-strong middle class is responsible for about 40% of all replies, about the same as the replies of more than 3 million of the poor's combined. Figurewhere are the figures? 7.12 contains the median number of replies issued from each bucket, as a fraction of the total, for all of our simulations. We further summarize the results in the following figures.

Bucket to bucket

In addition to the sheer overall volume of communications originating (or ending) in each bucket, we also want to look at bucket-to-bucket communication — i.e, for each bucket, how much replies from it are ending in every bucket including itself, or for mentions, how many of the mentions come from each bucket. Having computed the inter-bucket counts or fractions for every pair of buckets, how do we compactly represent such a matrix? The difficulty is that for our dynamic graph we have a set of buckets across a set of days already, and a full bucket-to-bucket table would require 3D representations or colorings. In order to stay with the table format, for each bucket-to-bucket array from a given bucket A, we sum communications to all the buckets higher than A, to A itself, and to all those lower than A. We then represent each of the three types of communication — with (a) higher classes, (b) lower classes, or (c) the same ones — as a separate table, thus yielding three tables per simulation, or six tables when replies and mentions are considered.

Starrank per bucket

Starrank was defined previously for comparing a rank of a node with the average rank of its audience. Previously, we used *drank*, a relativized pagerank, for ranking users. Here, we base our starrank off of social capital directly and compute it as follows on a daily basis.

Let x be a node, and $A(x)$ its audience, defined via some neighborhood metric. Specifically, we look at repliers x talks to and mentioners talking to x. For every node a in A, let n_a be the number of links between x and a of the required nature (e.g., a reply from x to a or a mention of x by a). Then average audience rank is

$$Ar(x) = \frac{\sum_{a \in A} n_a S(a)}{\sum_{a \in A} n_a}$$

Finally, starrank of x with respect to $A(x)$ is

$$Sr(x) = \frac{S(x)}{Ar(x)}$$

In our starrank tables, we show all three components, per bucket per day, average social capital, average audience rank, and their ratio, the starrank.

Longevity and Mobility

For any given number of days d, and a bucket B, we define users of *longevity* L_B^d as the set of users who stayed in B for d or more days overall. We than define $|L\_B^d|$ as the cardinality of that set. Note, the residency days in the bucket need not be contiguous.

We compute longevity sets for several fixed numbers of days, namely 7, 10, and 15, and we present the results in longevity cardinality tables per bucket.

More generally we can ask for a given user not only how long he stayed in a bucket, but where has he been before? We look at a general trajectory of a user across buckets, specifically asking the questions

- where did he start from?
- where did he end up?

- what was his deepest fall and his highest rise?
- where had he stayed the longest?

Skew

Skew is a measure of how well a user calibrates his communications in proportion to the contributions he gets. A politician's response would be skewed towards the higher contributors, hence the name "skew." In other words, skew answers a question, how "politician-like" is the behavior of this node?

We define skew as a non-parametric measure for a user with a set of incoming contributions (mentions) and corresponding responses (replies). First, we sort the list of contributing users in the descendant order of their contributions. Then we take a midpoint of that list which can be either chosen as either

- half of the top-most users vs the low-contributing half, i.e. simply by the number of users
- half of the total contributions, which will be different if we consider multiple replies as a proportionally higher contribution.

We used the latter by partitioning the user list in two so that each half will have approximately the same number of edges issued to our star U in hand, and so that the users in one, top half, A_0, all have the same or higher number of edges given to S than the bottom half B_0.

Now we define R_{A_0} as the total number of responses which U gave back to $A+0$ in this cycle, versus the number R_{B_0} he returned to B_0.

We define the original skew,

$$\begin{cases} S_0 = \frac{R_{A_0}}{R_{B_0}} & \text{if} \quad R_{B_0} > 0 \\ \infty & \text{otherwise} \end{cases}$$

Then we take A_0 and repeat the process, splitting it in two halves, A_1 and B_1, aligning responses from R_A against those top two quarters of the original contributions, and computing $S_1 = \frac{R_{A_0}}{R_{B_0}}$ if $R_{B_0} > 0$, inf otherwise. We repeat diving the top half until either a given fixed skew vector size, for instance 4, or until it empties out.

We now collect *skew* vectors for all users in a time slice, considering their daily contributions and responses, and we sort the skew list by first preferring longer vectors to shorter ones, meaning more

action at the top, even if not always preferring the top half at each step and then component-wise comparisons for vectors of equal length, with two infs being equal and ties at a position broken by (recursively) comparing the rest. Listing 7.7 shows the comparison rules verbatim as implemented in OCaml.

Listing 7.1: Comparing two skew vectors accounts for variable length, and distinguishes $\infty = -1$

```
let rec compareSkew ?(lengthFirst=true) xs ys =
    let rec aux xs ys =
        match xs,ys with
        | x::xs,y::ys when x = -1. && y = -1. ->
            compare xs ys
        | x::xs,y::ys when x = -1.              ->  1
        | x::xs,y::ys when           y = -1. -> -1
        | x::xs,y::ys                       ->
            let c = compare x y in if c = 0 then compare xs ys else c
        | x::xs,_                           ->  1
        | _,y::ys                           -> -1
        | _                                 ->  0
    in
    if lengthFirst then begin
        let c = compare (L.length xs) (L.length ys) in
        if c <> 0 then c else aux xs ys
        end
    else
        aux xs ys
```

In order to evaluate our *skew* findings, we use the skew ordering above to sort all users in a cycle by skew, and we then compare the ordering with the reciprocal social capital ordering. We use Kendall's τ

as a well-established non-parametric measure of alignment between two sorted lists of the same length.

Kendall's τ is a computationally expensive non-parametric metric with most implementations readily available running in $O(n^2)$ time in the length of the input arrays. This approach works for many hand-made examples but fails quickly when scaling up to the millions of points such as found with Twitter user ranks on a given day.

There are at least two papers reporting $O(n \log n)$ algorithms, Knight 1966 [47] and Christensen 2005 [15]. We implemented efficient bindings to a fast C version of the Knight's algorithm and solved several interesting problems on the way. Since skews are not numbers but vectors of varying length they have to be sorted using their own comparator (shown above). However, storing those vectors as is would not allow for very fast comparisons we need in order to make this computation feasible.

For our implementation,

- we first sort the array of (user,capital) tuples by capital (in ascending order)
- we sort the skews for the corresponding users and map their sorted list onto an integer range, 1-based, keeping the ties equal, creating unique 1-1 mapping, except that any missing skew is mapped to 0. Such scores induce the same ordering on users as skews
- for each capital in the first list we look up its user's skew score in the first list's order of users. The user list is now no longer necessary, having achieved the alignment of the two numeric lists
- we then compute Kendall's τ on these two numeric arrays (they are passed to C as OCaml's big arrays)

7.8 Results

Overlaps with Reality

Figures 7.1,7.2,7.3,7.4 show the evolution of reality overlap as the amount of reality seeding increases.

Global uniform strategies also coupled with mentions or capital-based *FOFs*, are closer to reality when starting from scratch then gradually yielding to global mentions and mentions-mentions, finally grouping with global capital and capital-capital simulations. Smarter simulations achieve higher overlap with reality especially when using our reciprocal social capital.

Figure 7.1 shows the reality overlap clustering of all of the basic simulations run from scratch without mixing any of the *dreps* reality itself.

There's no bucketed simulations here, as buckets kick in after a week for the regularly matured social capital. Middle and poor class users show much overlap with the originals, and the baselines for *ureps* are darker than most, meaning we're already doing a good job capturing some classes.

Figure 7.2 shows the reality overlap clustering of all of the basic simulations run after mixing in one week of the *dreps* reality itself.

The bucketed simulations join in, and while those modifying only small elite buckets, *1,2,3f*, are predictably close to reality as is *6f*, thus simulating the complete one million strong middle class with the capital-capital strategy, *fg5cf1c*.

Figure 7.3 shows the reality overlap clustering of all of the basic simulations run after mixing in two weeks of the *dreps* reality itself.

The bucketed middle class stays close to reality here as well.

We also see increasing capture of the celebrity class, the top ten "social capitalists," by both bucketed and non-bucketed simulations, specifically

- creps2wk
- fg8uf05m2wk

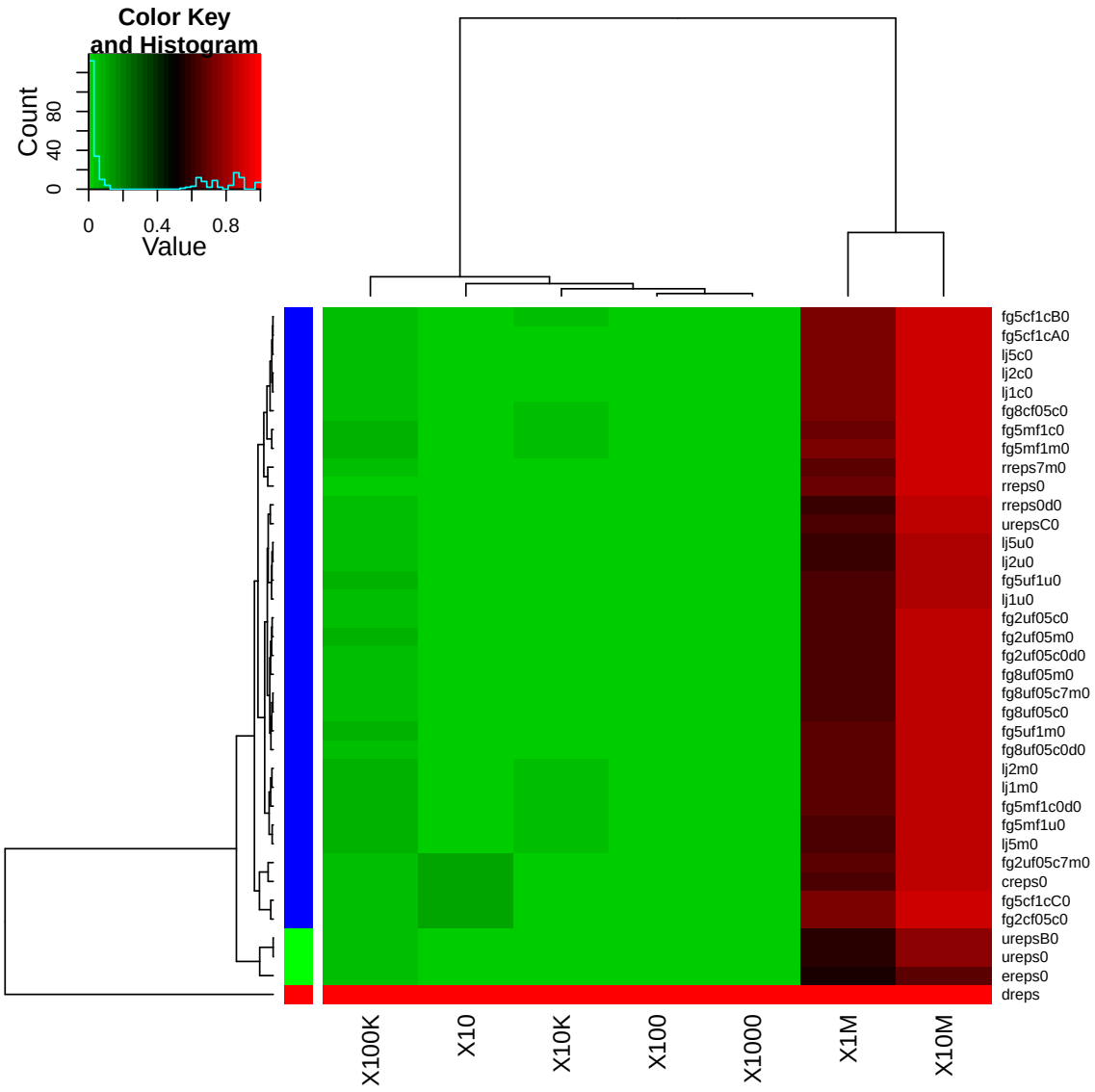

Figure 7.1: Reality overlap clustering of all simulations started from scratch.

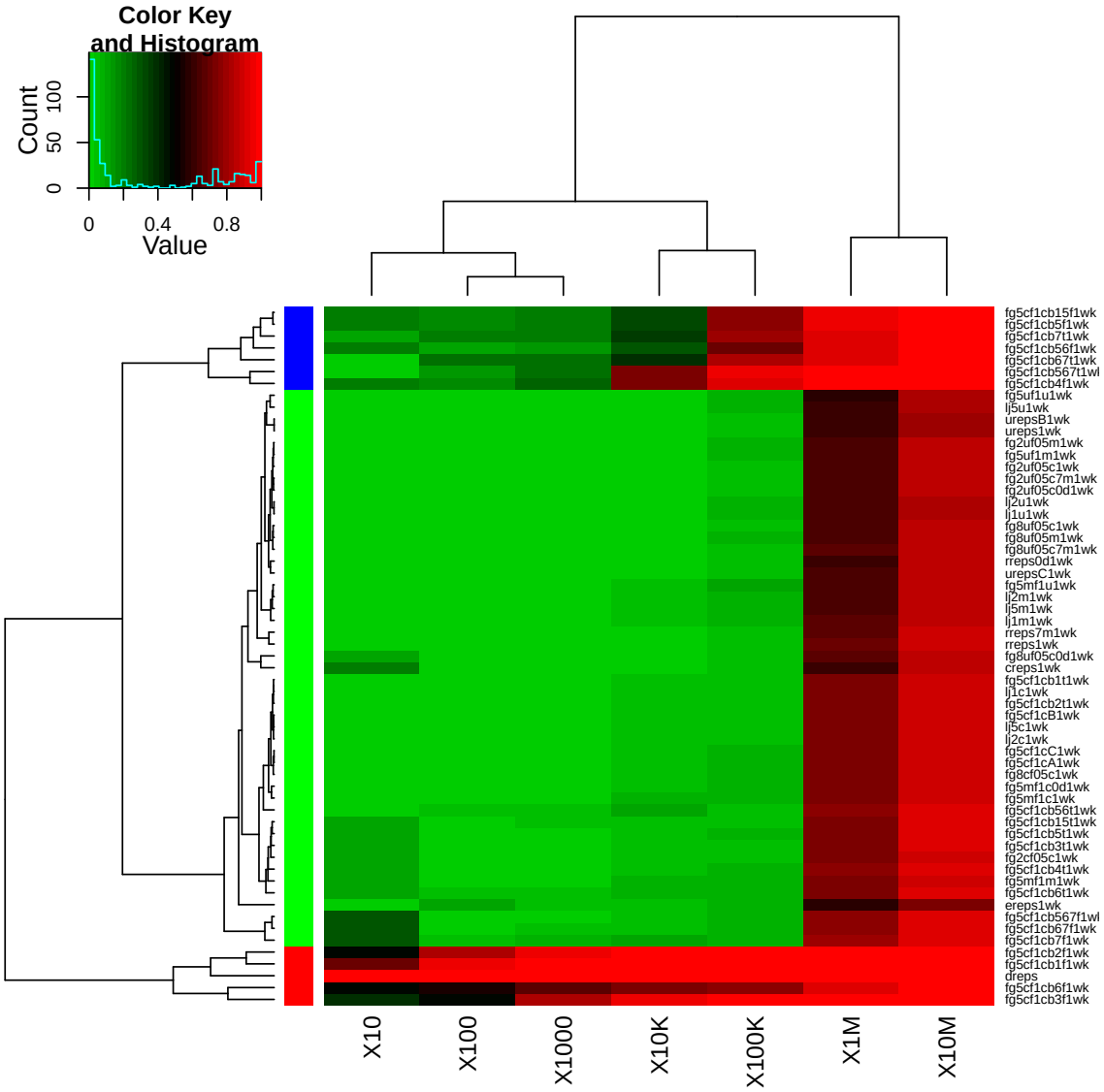

Figure 7.2: Reality overlap clustering of all simulations seeded by 1 week of reality.

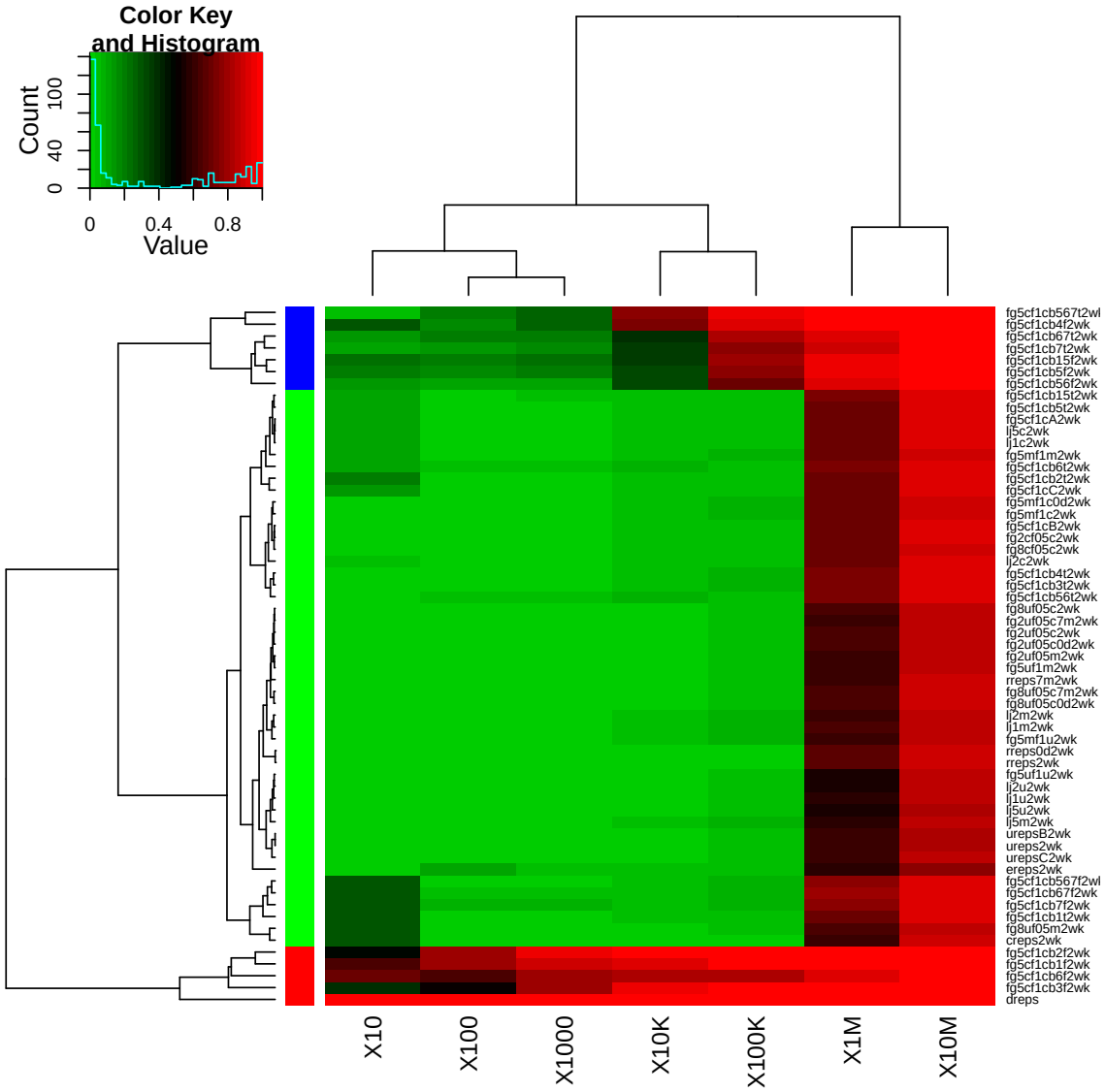

Figure 7.3: Reality overlap clustering of all simulations seeded by 2 weeks of reality.

Figure 7.4 shows the reality overlap clustering of all of the basic simulations run after mixing in three weeks of the *dreps* reality itself.

The simulated middle class remains really close to reality, and celebrities are captured better than in most of the other simulations, except the uniform *ureps* runs. Notably, the mentions-mentions strategy with 0.5 utility probability does well:

- fg5m1m

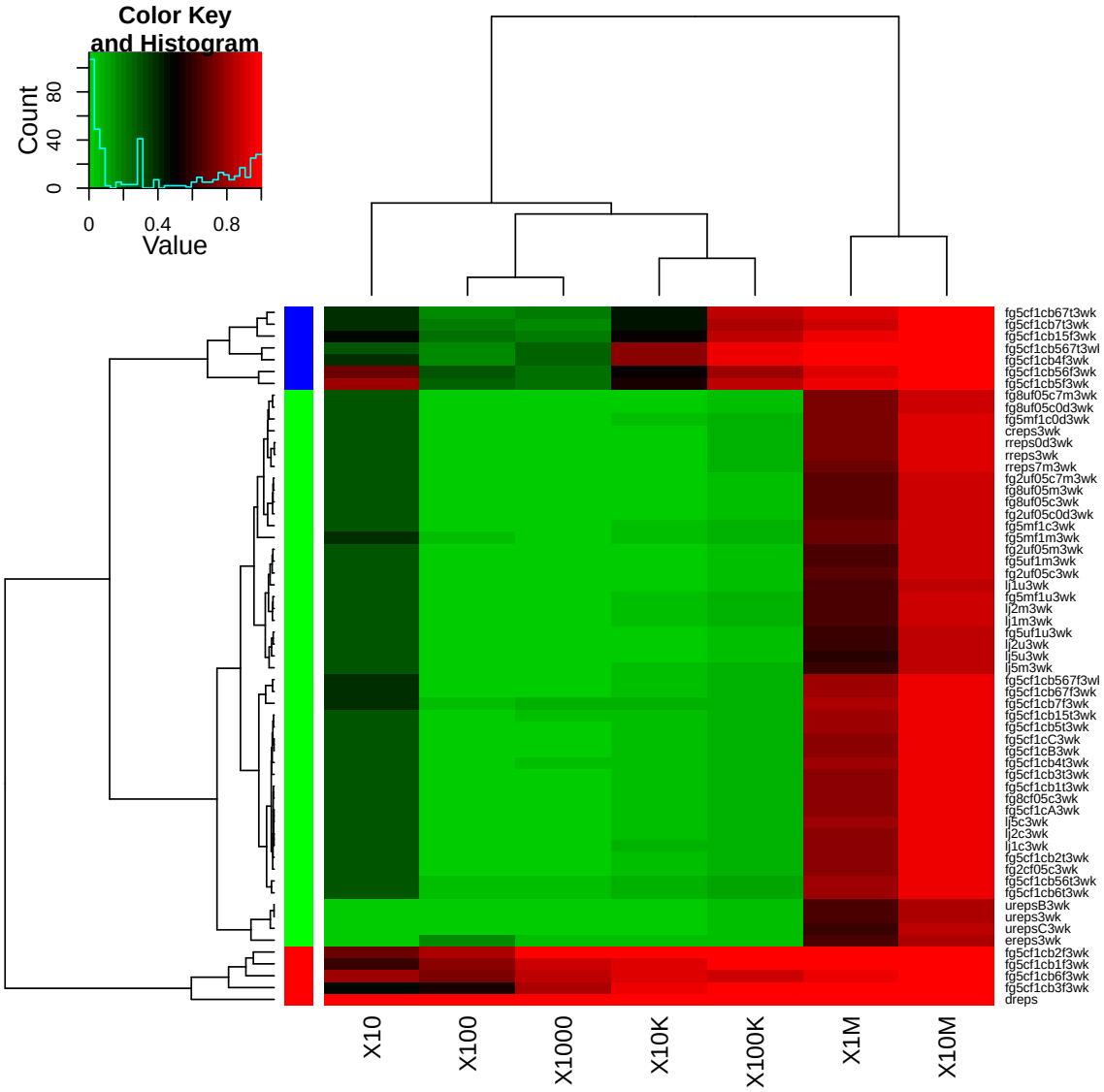

Figure 7.4: Reality overlap clustering of all simulations seeded by 3 weeks of reality.

Overlap with Self, Reality-shifted

For all simulations with several runs differing only by the amount of reality seeding, we compute the overlap of social wealth hierarchies between the simulations separated by a week of reality with as many steps as there are such pairs. This usually results in 4 pairs for non-bucketed, or 2 pairs for bucketed simulations.

Capital-capital FOF-based simulations show good stability right away, as seen in Figure 7.5, with especially strong showing in the top 1,000 elite bucket and the middle class.

Rrreps and *creps* do well in the upper-middle class while FOF-based, uniform global strategies capture the celebrity bucket.

The pattern continues into the first through second week pair, Figure 7.6, in which non-bucketed capital-based simulations, both FOF and non-FOF utility based, improve capturing rate in the middle range of classes considerably. The newly arrived bucketed simulations do especially well. This continues into the second through third week, Figure 7.7, and third through fourth week as well, Figure 7.8.

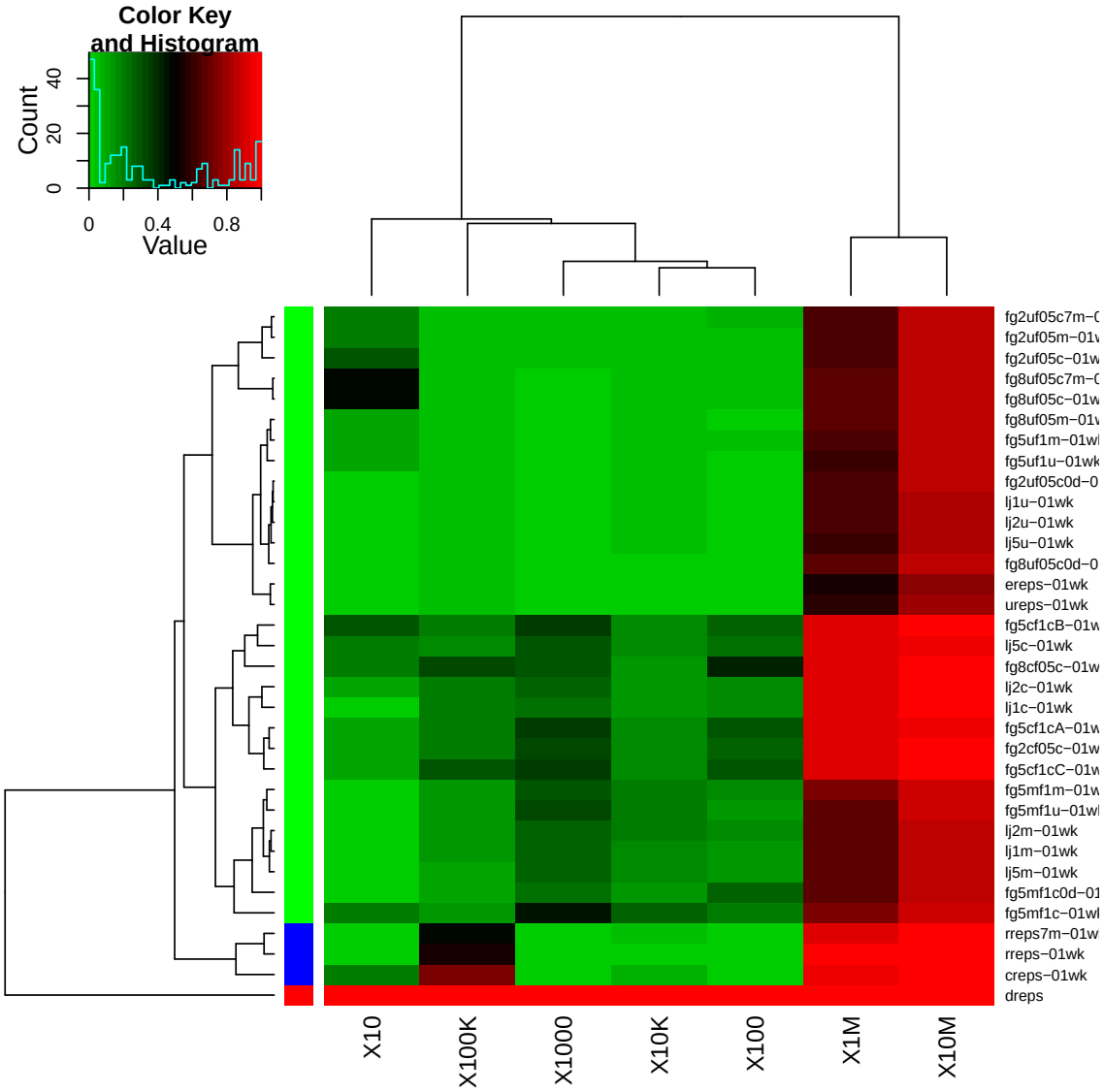

Figure 7.5: Overlap with the same strategy seeded by one more week of reality, 0 vs 1 week.

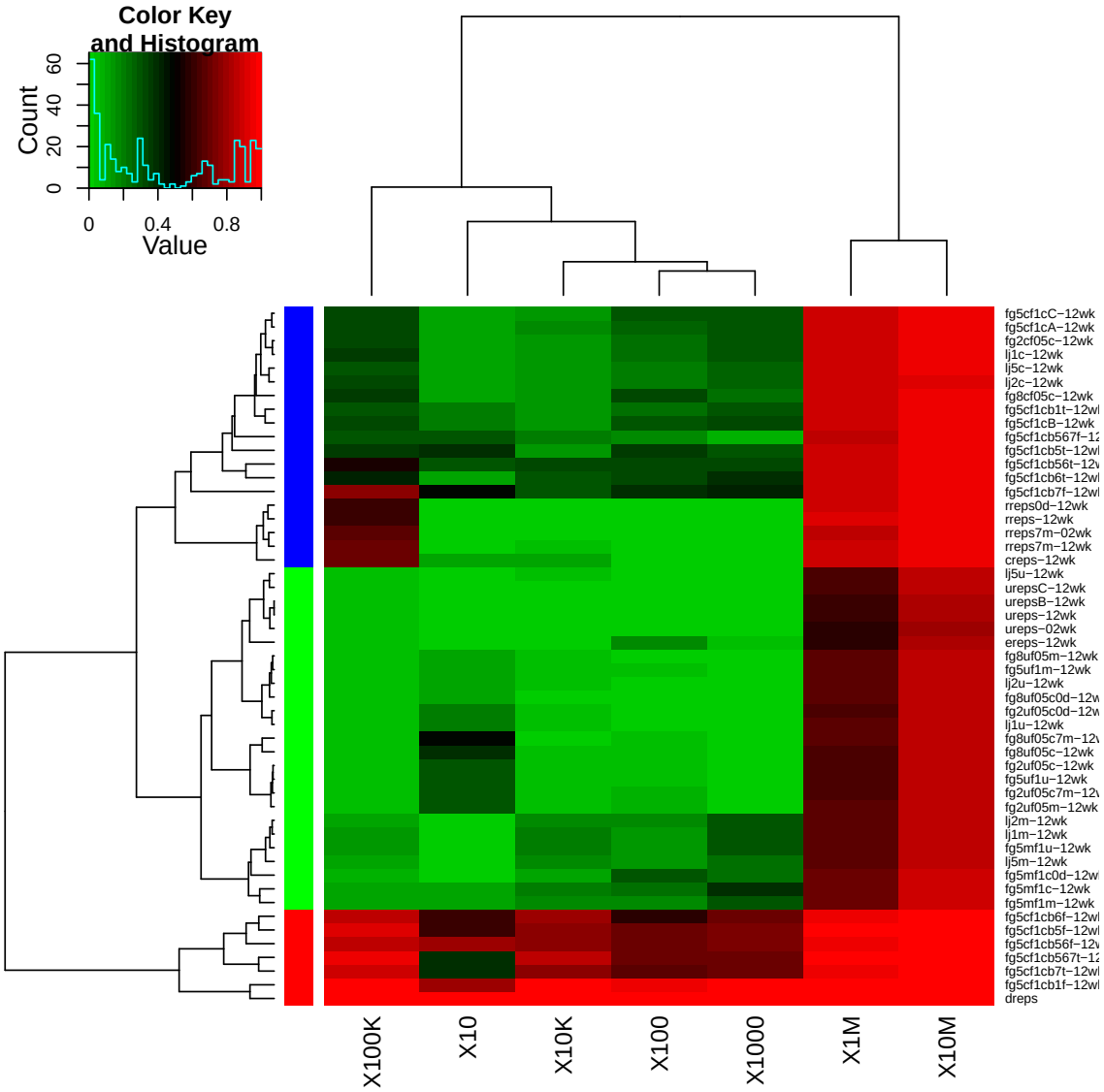

Figure 7.6: Overlap with the same strategy seeded by one more week of reality, 1 vs 2 weeks.

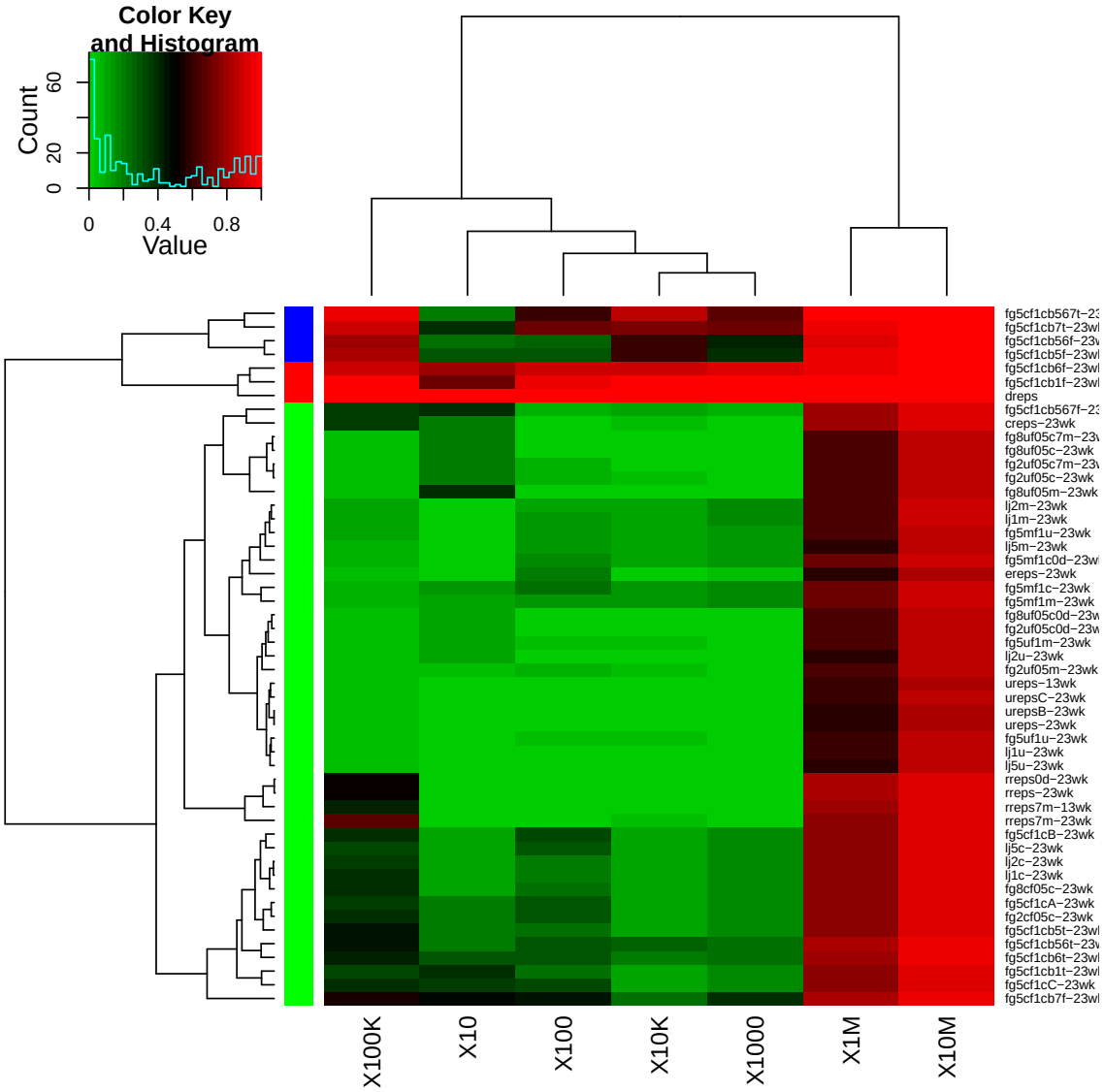

Figure 7.7: Overlap with the same strategy seeded by one more week of reality, 2 vs 3 weeks.

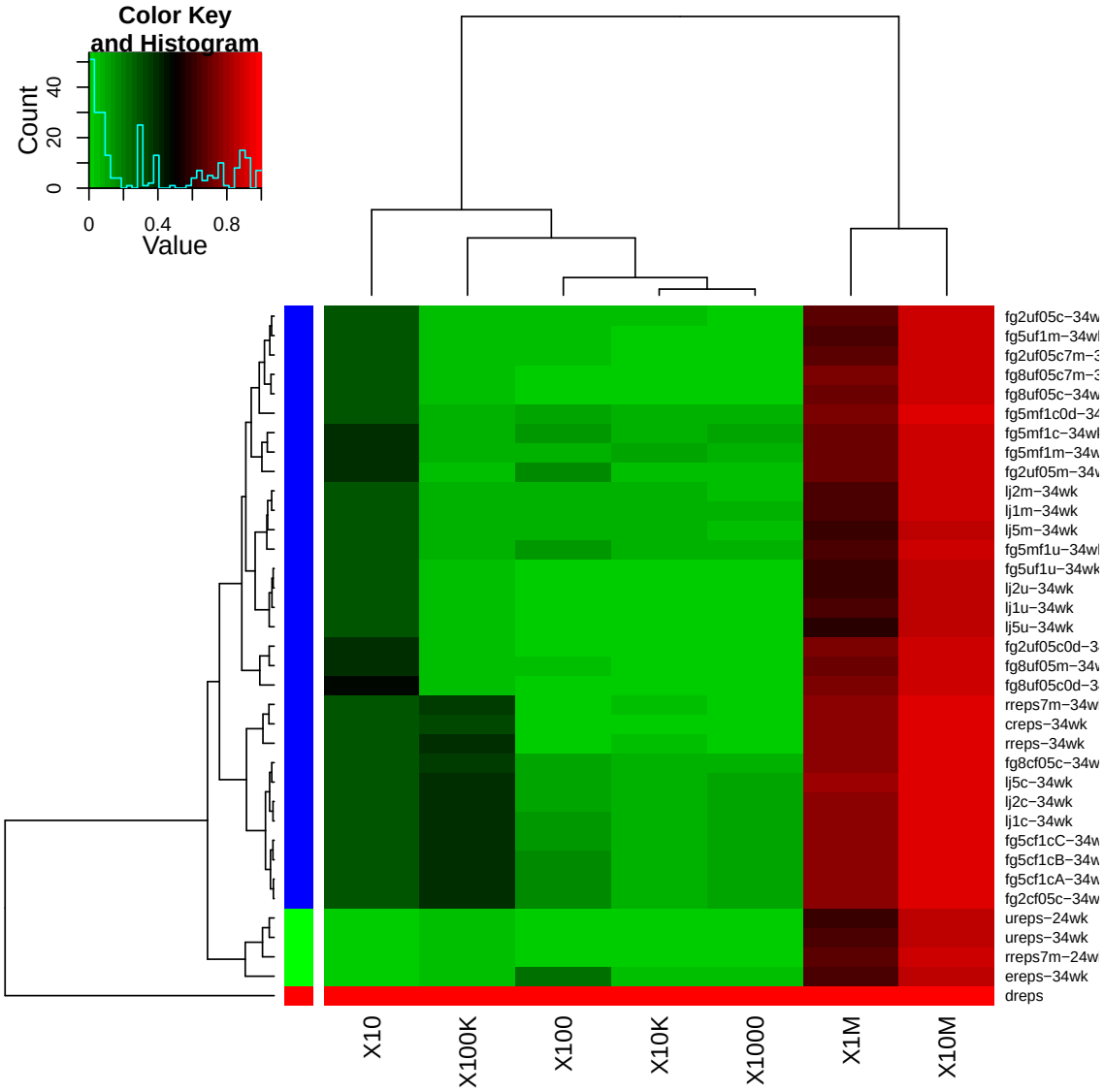

Figure 7.8: Overlap with the same strategy seeded by one more week of reality, 3 vs 4 weeks.

Staying Rates

Figures 7.9,7.10,7.11,2.3 show the creation of staying power as the amount of reality seeding increases.

Uniform global strategies coupled with capital-based *FOF* strategies are closer to *dreps* initially. Then mentions-based global ones get closer, and finally capital-capital ones dominate. It shows that effective communication gets closer to reality when seeded with more history of such communication.

Figure 7.9 shows the staying power clustering of all of the basic simulations run from scratch without mixing any of the *dreps* reality.

Several *freps* simulations are near *dreps*, notably fg8uf05m0 and fg5uf1m0. Also *ereps0* is not far. Capital-capital ones are in a separate cluster.

Figure 7.10 shows the staying power clustering of all of the basic simulations seeded with one week of the *dreps* reality.

Bucketed middle class simulation, *6f*, is close, and *ereps_* is not far again. Simulated celebrities, *1f*, are in a separate cluster. *2f* is next to *dreps*, and *fg5mf1c* and *fg5mf1m* are next.

Figure 7.11 shows the staying power clustering of all of the basic simulations seeded with two weeks of the *dreps* reality.

The bucketed ones are the closest as usual – *2f*, *15f*. Then we have

- *lj1,5,2u*
- *fg8uf05c0d*

  6f is further in the cluster, next to *1f*.

Figure 2.3 shows the staying power clustering of all of the basic simulations seeded with three weeks of the *dreps* reality.

The bucketed are now bracketing reality, with the middle class *6f* one of the closest. Among the closest nonbucketed ones are

- *fg5cf1cC*
- *rreps7m*

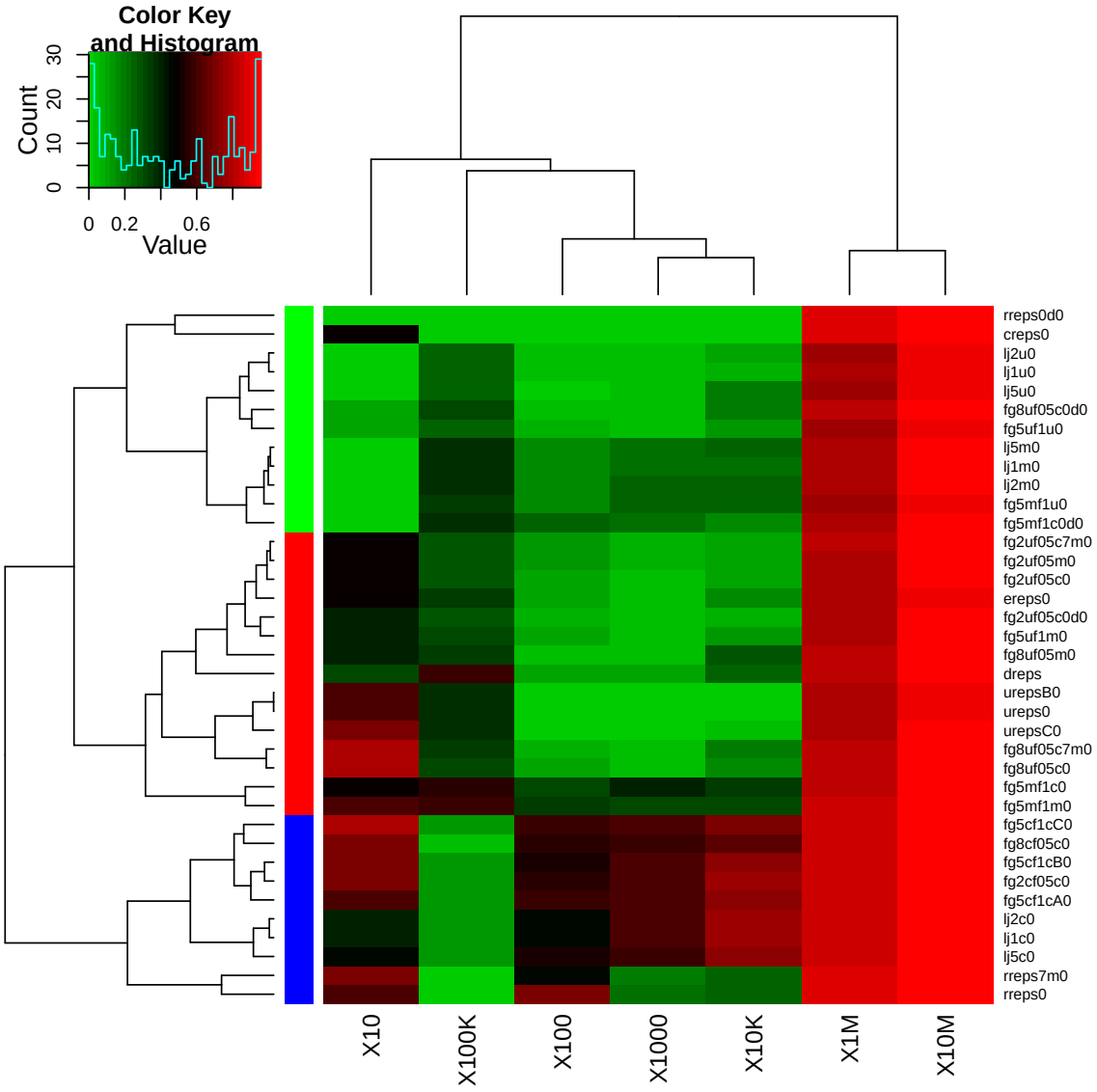

Figure 7.9: Staying power clustering of all simulations started from scratch.

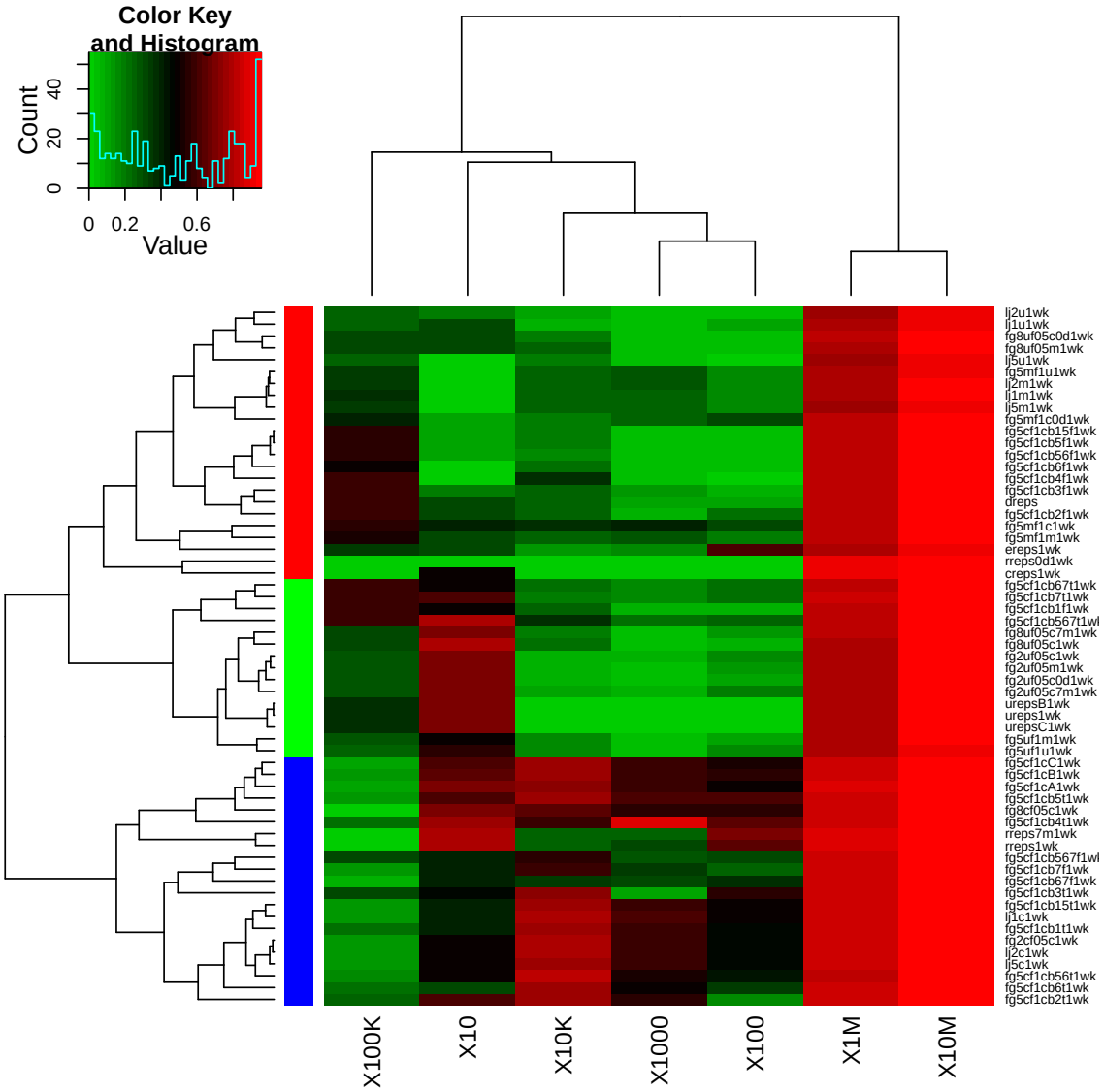

Figure 7.10: Staying power clustering of all simulations seeded by 1 week of reality.

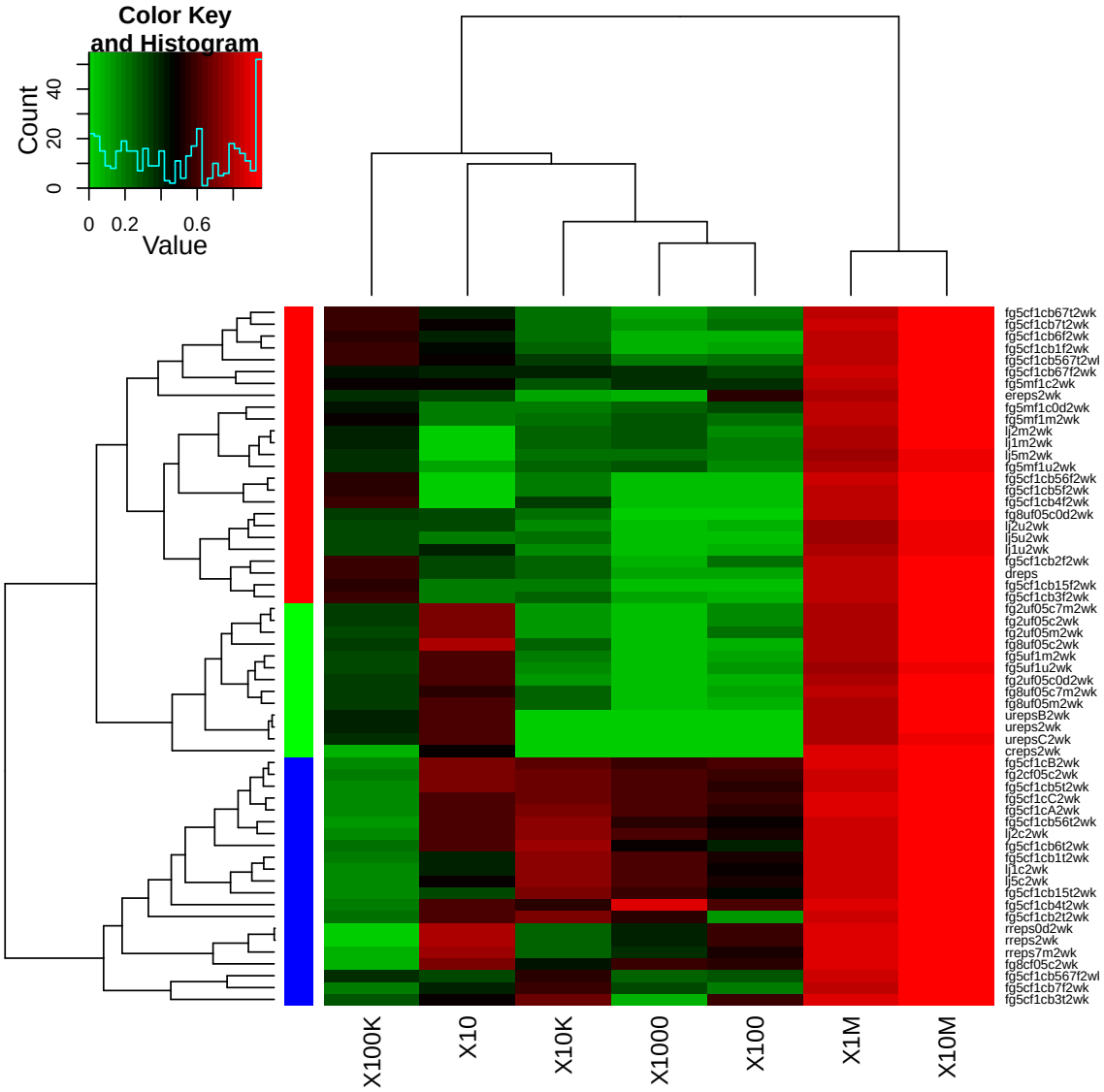

Figure 7.11: Staying power clustering of all simulations seeded by 2 weeks of reality.

- *ereps*

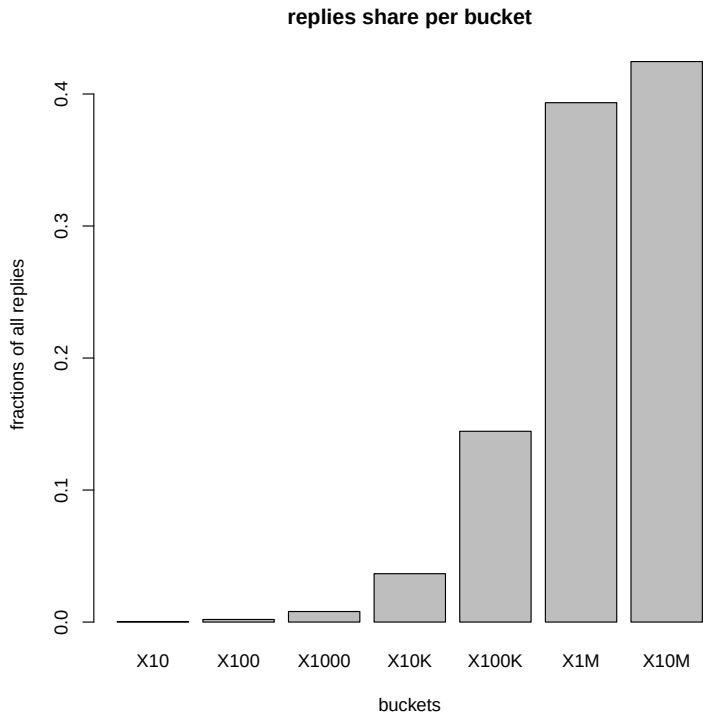

Figure 7.12: Medians of all bucket reply shares, for all simulations, per bucket.

Volumes per Bucket

Volume of Replies

The volume of replies per bucket follows mostly the same pattern, shown here as a barplot of the medians per bucket, across all simulations. The middle class contributes practically as much as all the poor, combined, about 40% of all replies.

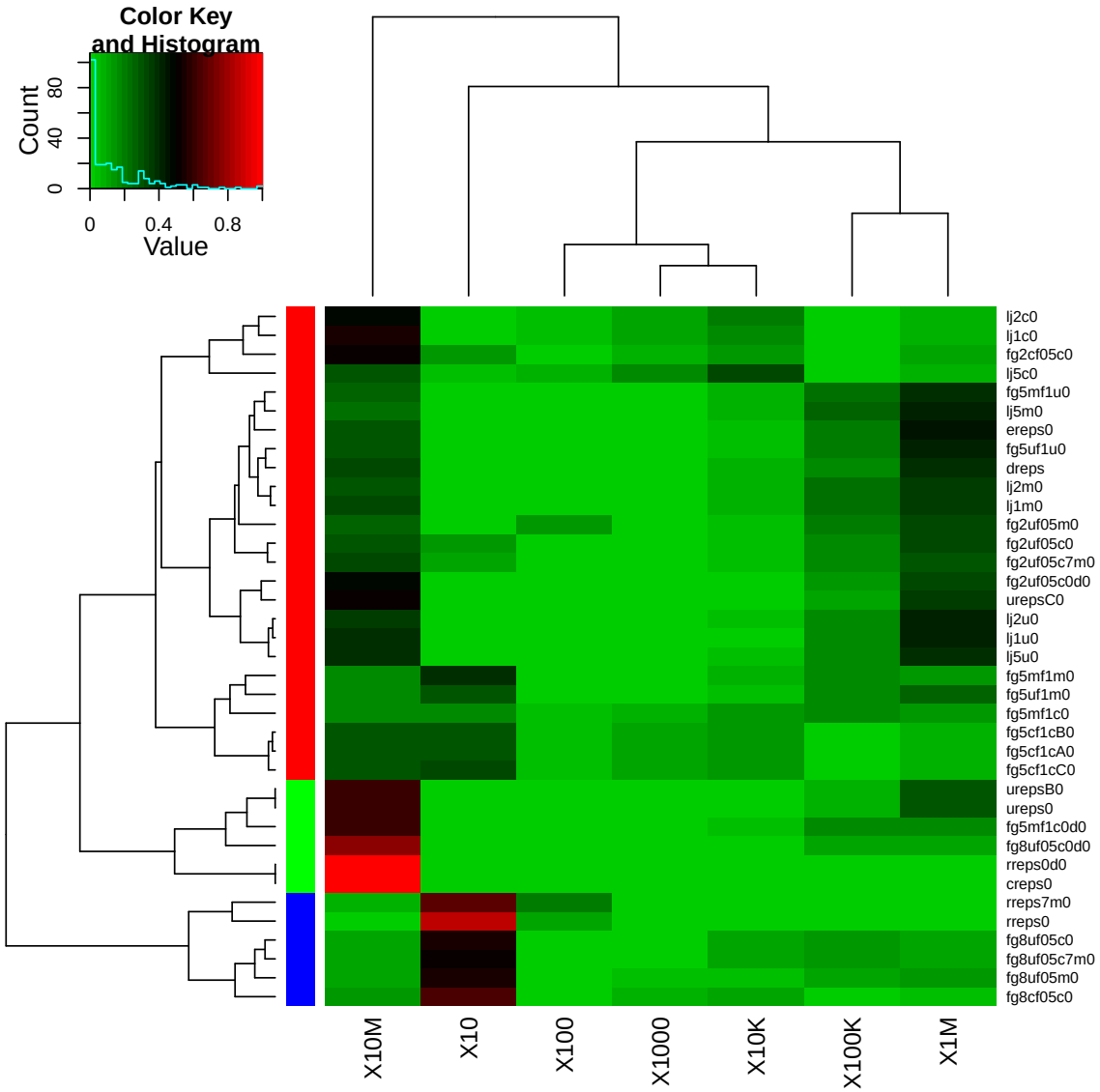

Figure 7.13: Per-bucket mentions volume for all simulations started from scratch.

Volume of Mentions

Figure 7.13 shows per-bucket volume of mentions (incoming communications) for all simulations run from scratch without mixing any of the *dreps* reality.

Initially, strategies with a smaller utility probability are closer to reality, along with global mentions, pure or combined.

Figure 7.14 shows per-bucket volume of mentions (incoming communications) for all of the simula-

tions, seeded with one week of the *dreps* reality.

Once the bucketed simulations are considered after their warm-up week, we have the usual elite-changing simulations clustering around reality, with the preserved middle class, *6t*, hovering nearby, as well as two FOF-based simulations with *jumpProbUtil* of 0.5, and global mentions with utility:

- fg5uf1u
- fg5mf1u
- lj5m

Figure 7.15 shows per-bucket volume of mentions (incoming communications) for all of the simulations, seeded with two weeks of the *dreps* reality.

Bucketed align around *dreps* in the usual formation, with the previously noted non-bucketed ones staying around:

- fg5mf1u
- lj5m

The simulated middle class is near the opposite combination, global uniform after FOF-mentions, in another subcluster of *dreps*:

- fg5cf1cb6f
- fg5uf1m

Figure 7.16 shows per-bucket volume of mentions (incoming communications) for all of the simulations, seeded with three weeks of the *dreps* reality.

Finally, fractions look quite similar. Clustering keeps our early non-bucketed close:

- fg5mf1u
- lj5m

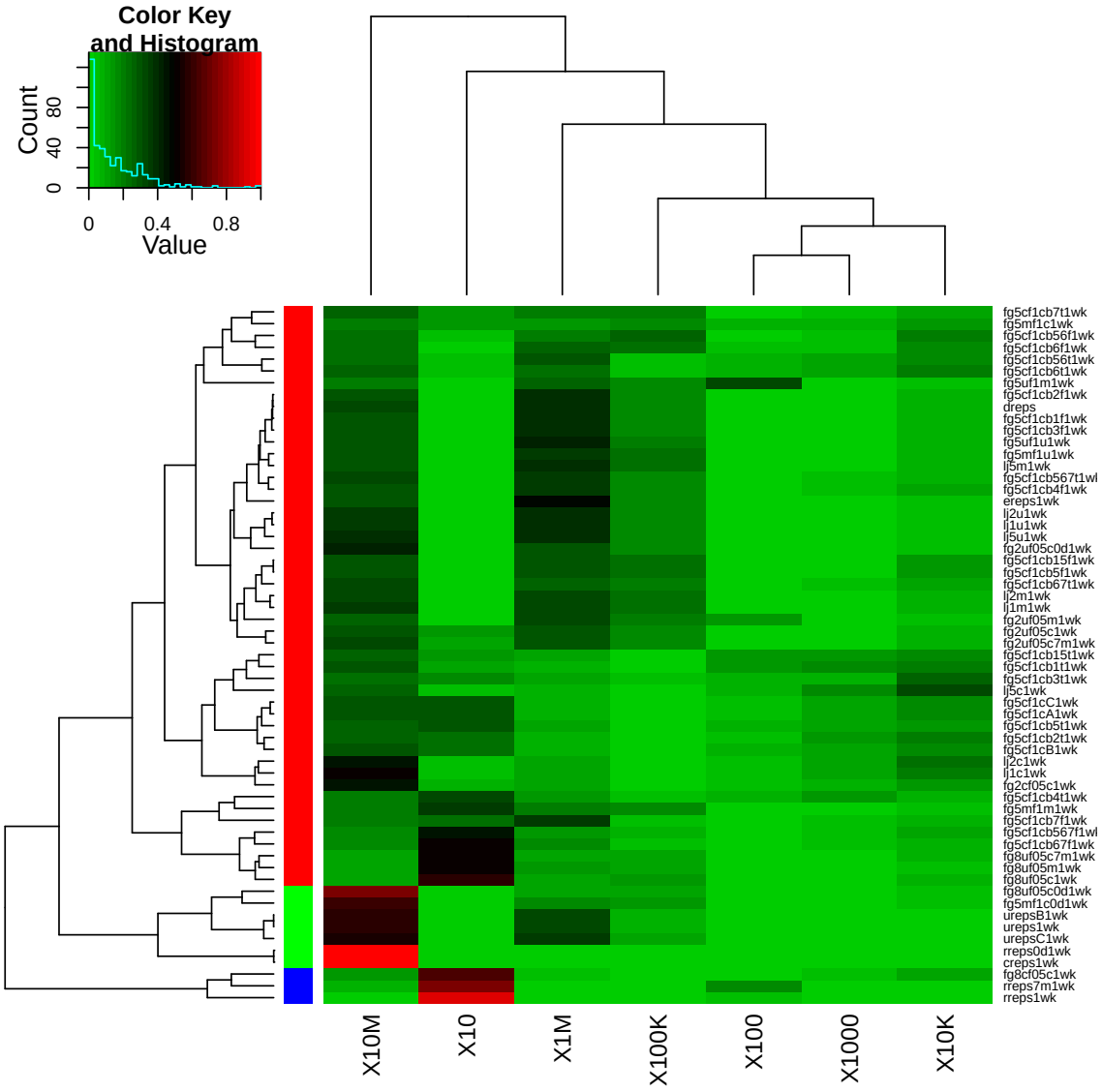

Figure 7.14: Per-bucket mentions volume for all simulations seeded by 1 week of reality.

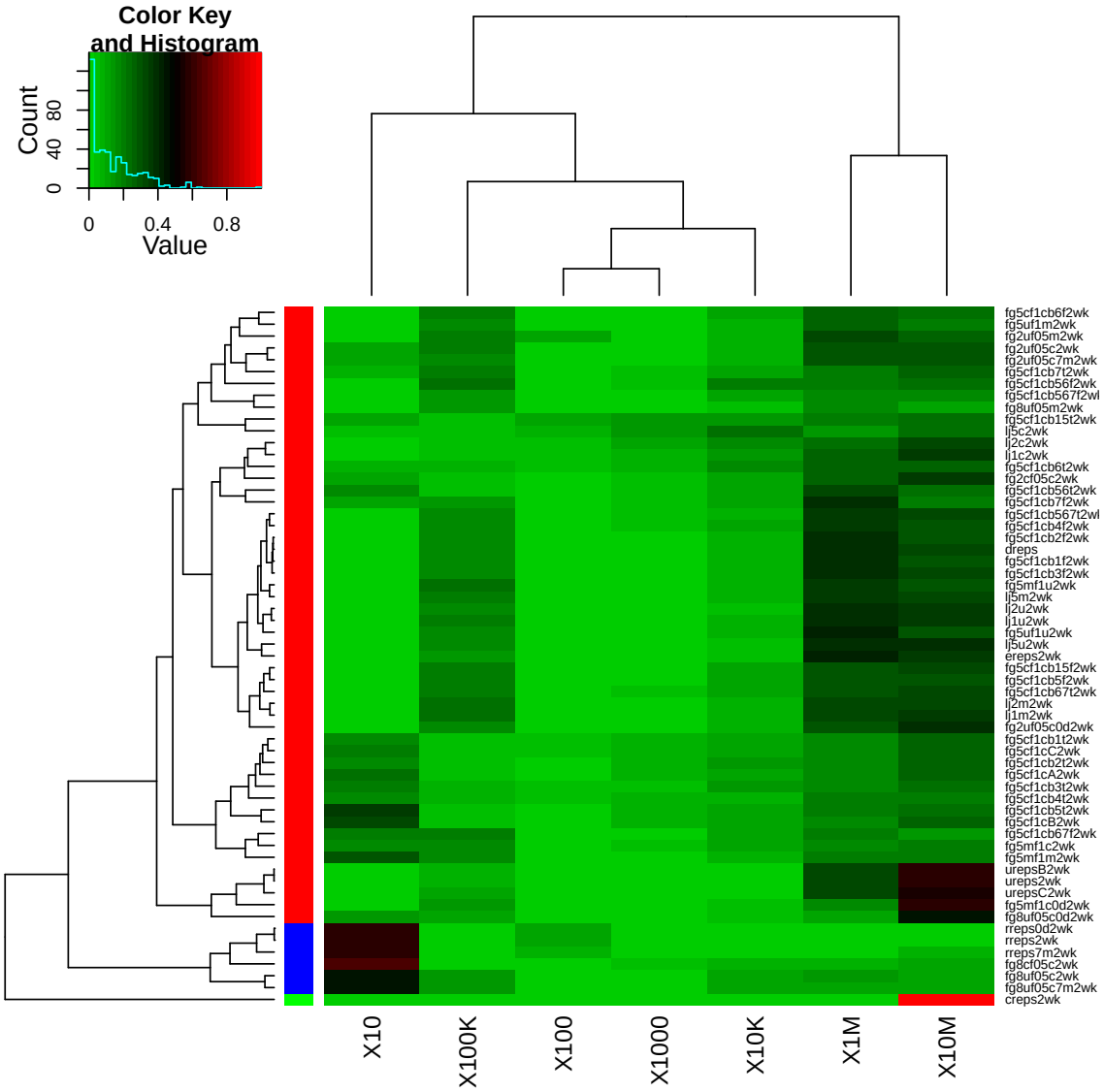

Figure 7.15: Per-bucket mentions volume for all simulations seeded by 2 weeks of reality.

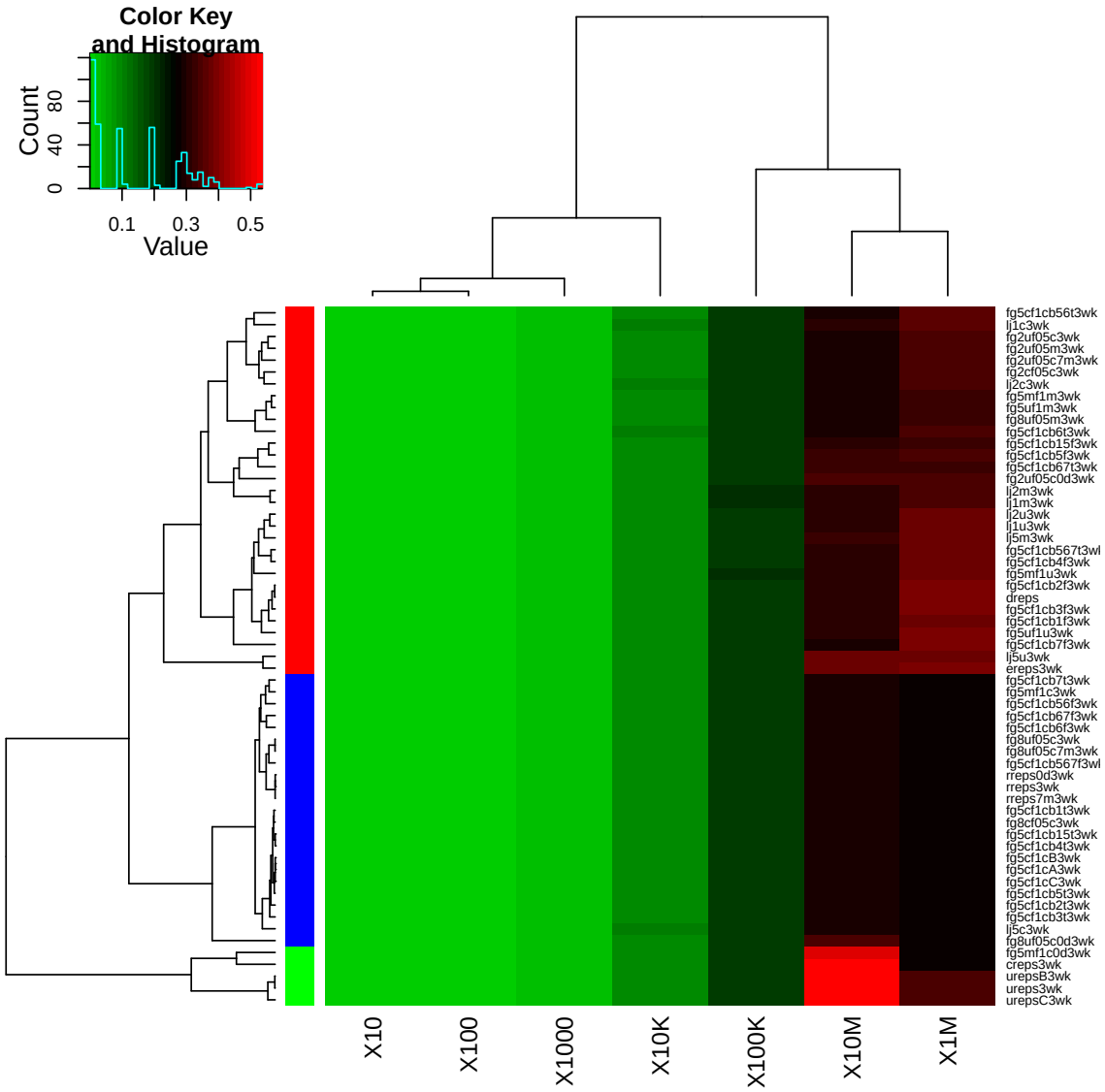

Figure 7.16: Per-bucket mentions volume for all simulations seeded by 3 weeks of reality.

The simulated middle class is in a separate cluster since it underestimates the poor, while the preserved original does better.

Starranks

Starranks are ratios, and as such fall into a wider range than overlaps or volumes, which are simple fractions from 0 to 1. In order to compare the differences in starranks per buckets, ranging widely, we operate with $\log 10$ of starranks. This also clearly shows how lower classes generally talk to a higher-ranked audience, while the top classes talk to the lower-ranked classes.

Starranks for Mentions

We cluster median starranks per bucket aggregated as overall medians for all days of each simulation grouped by distance from each other and *dreps*, the reality. We segment the simulations into four overall classes, 0, 1wk, 2wk, and 3wk, as defined by the amount of reality used to seed them.

Figure 7.17 shows the median starranks based on the ratios of one's social capital over the weighted average of one's audience's social capital per-bucket for mentions (incoming communications), for all simulations run from scratch without mixing any of the *dreps* reality.

We find that capital-capital simulations are the majority in the *dreps* cluster.

Figure 7.18 shows median starranks based on the ratios of one's social capital over the weighted average of one's audience's social capital per-bucket for mentions for all of the simulations, seeded with one week of the *dreps* reality.

The bucketed simulations changing only the smaller elite buckets are naturally closer here, but *ereps1wk* is a notable closest non-bucketed simulation.

Interestingly the simulations keeping or redoing the middle class, *6t*/*6f*, end up close together, showing that the middle class is well approximated by the underlying capital-capital FOF-based strategy.

We discover a block of high starrank in upper middle class for many simulations containing a global uniform strategy (and a couple of mentions-based one).

Figure 7.19 shows median starranks based on the ratios of one's social capital over the weighted average of one's audience's social capital per-bucket for mentions for all of the simulations, seeded with two weeks of the *dreps* reality.

The combination redoing the top 4 buckets and keeping the 3 lower ones, *567t*, is close to *dreps*, followed by *67t* while the poor by themselves, _7t, are father away thus showing that the middle class plays a significant role.

We have a similar block of high starrank in upper middle class for many simulations containing a global uniform strategy, only two of them FOF-capital based, with rather large jump probability away from local utility (0.5 and 0.8).

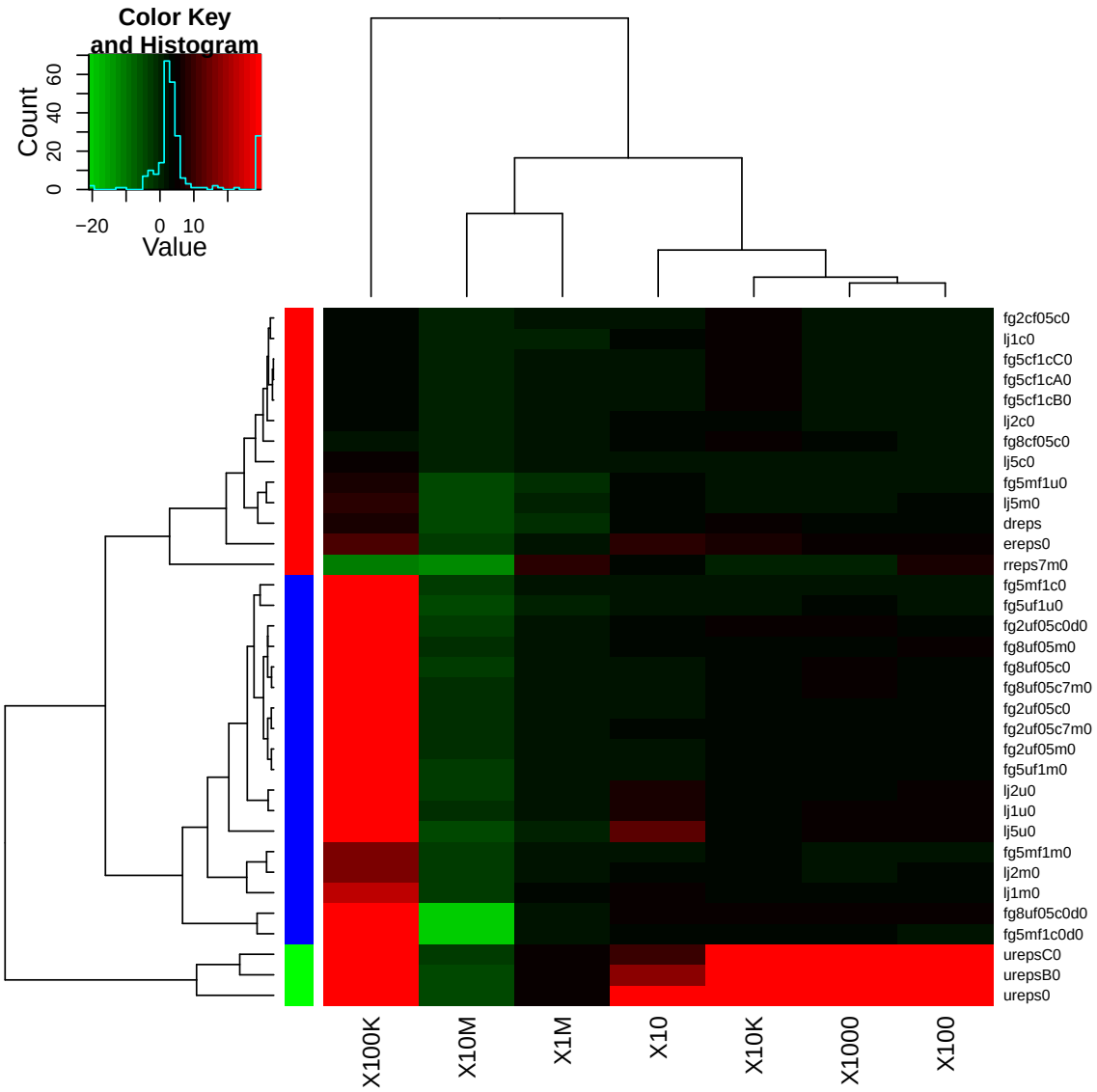

Figure 7.17: Starranks by mentions for all simulations started from scratch.

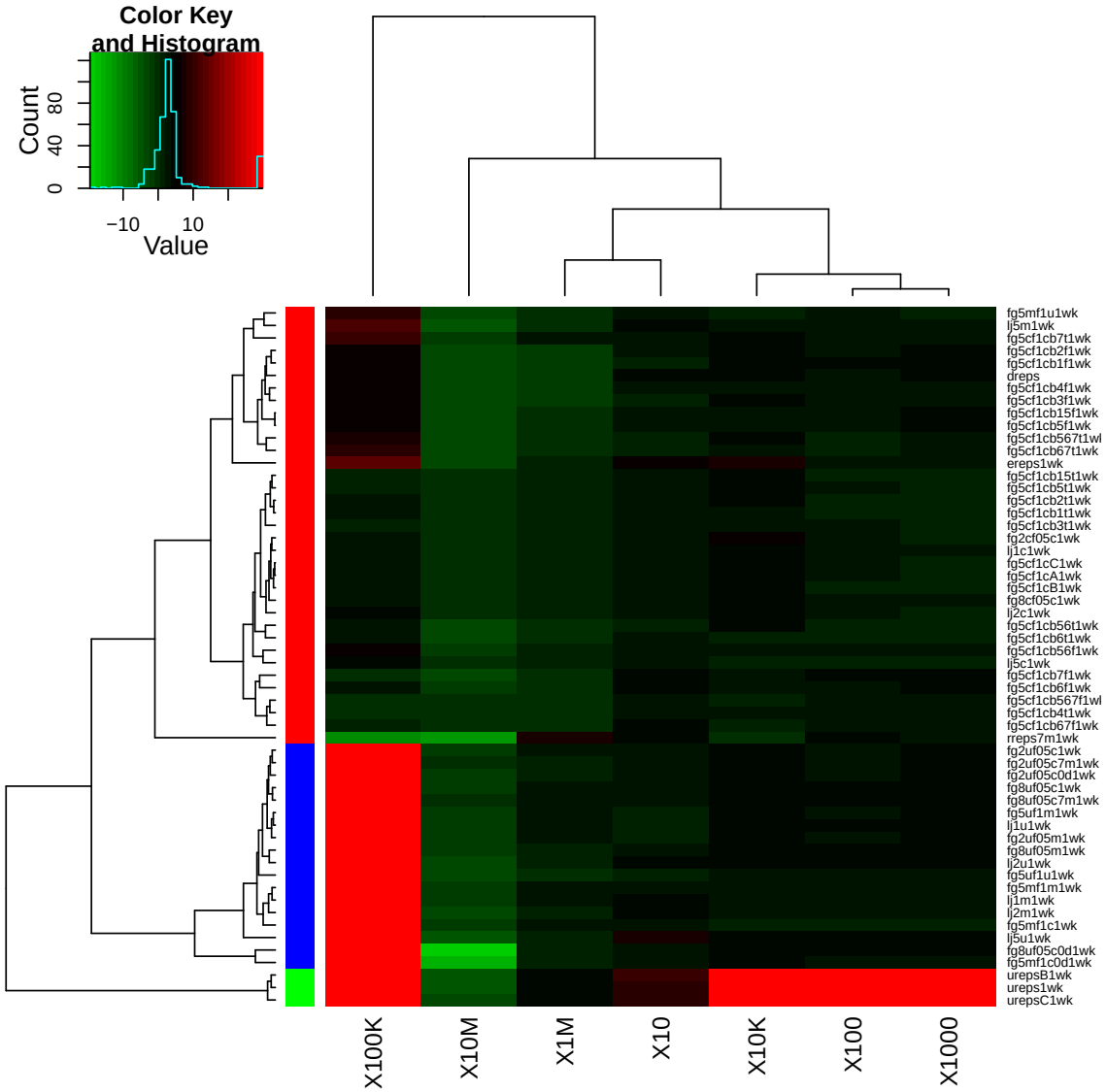

Figure 7.18: Starranks by mentions for all simulations seeded by 1 week of reality.

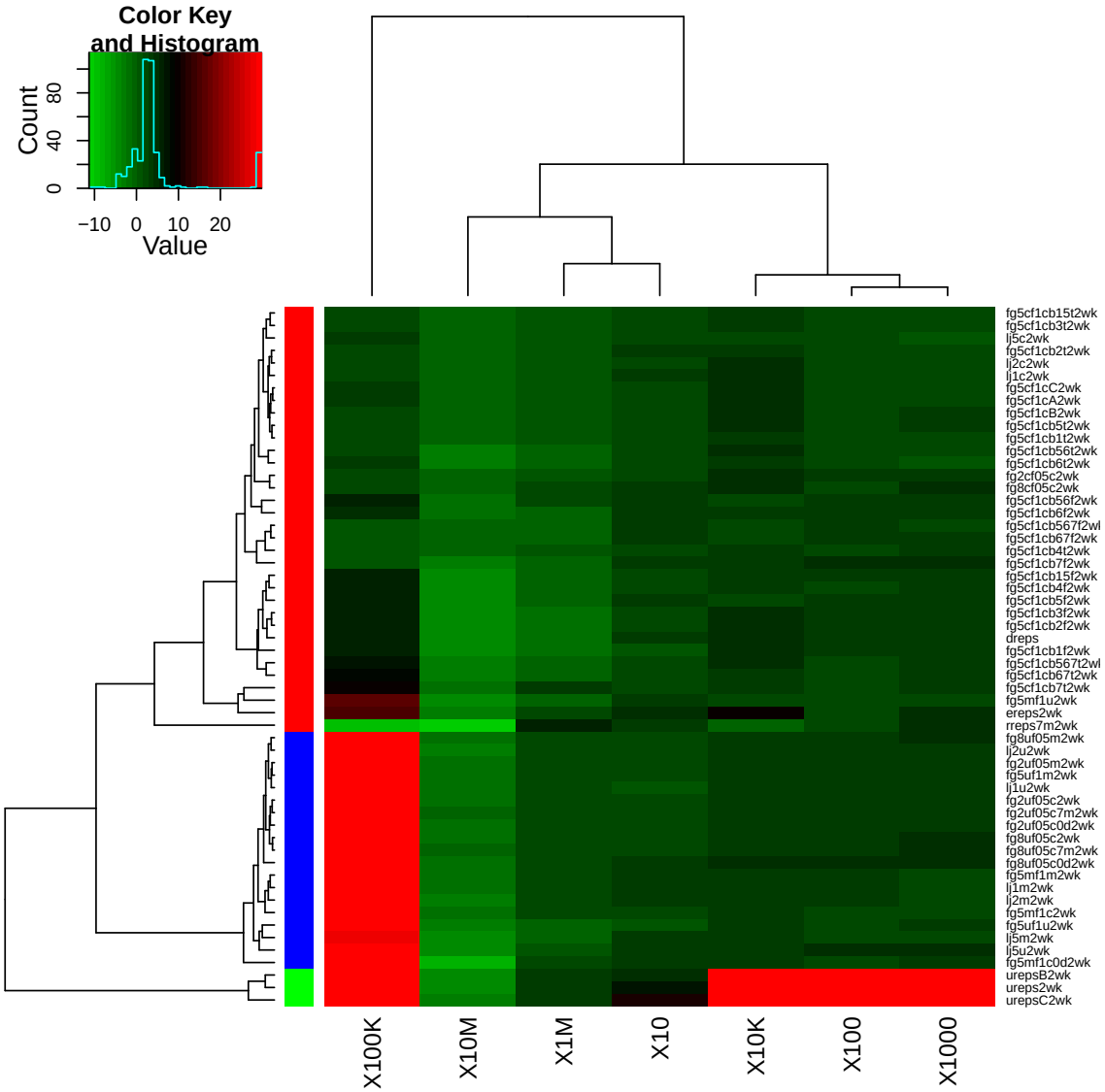

Figure 7.19: Starranks by mentions for all simulations seeded by 2 weeks of reality.

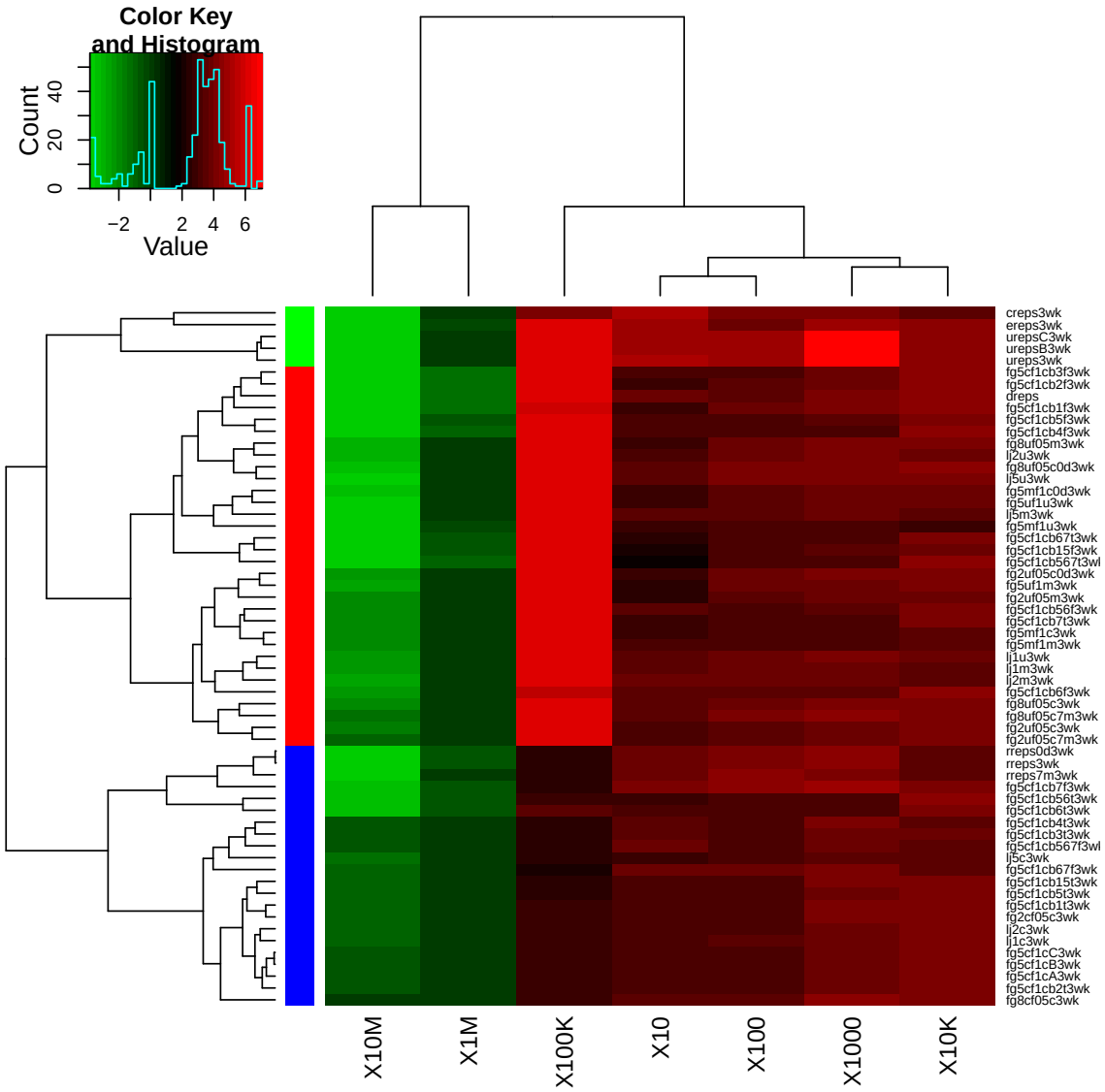

Figure 7.20: Starranks by mentions for all simulations seeded by 3 weeks of reality.

Figure 7.20 shows median starranks, based on the ratios of one's social capital over the weighted average of one's audience's, per-bucket for mentions (incoming communications) for all of the simulations, seeded with three weeks of the *dreps* reality.

The range is shrinking here, from -2 to 6, while it exceeded the [-10,10] range for 1-2 weeks.

We have an even bigger block of simulations with extremely high starrank in the upper-middle class, now containing *dreps*.

Starranks for Replies

We cluster median starranks per bucket aggregated as overall medians for all days of each simulation by distance from each other and *dreps* (the reality). We segment the simulations into four overall classes, 0, 1wk, 2wk, and 3wk, as defined by the amount of reality used to seed them.

The initial generation places two non-bucketed simulations around *dreps*:

- fg5uf1u
- fg5mf1u

In the first weeks, the simulated middle class *6f* is already near reality. Notable non-bucketed in the reality cluster are

- fg5uf1u
- lj2m
- fg5m1u

The same picture, and the same wide range, of starranks by mentions persists after week 2 of reality. The middle class *6f* is right there, after *1,2f*, and our non-bucketed stay near:

- fg5uf1u
- ereps
- lj5m
- fg5mf1u
- fg8uf05c0d

Finally, in the third week, the exponent range of the starrank by mentions shrinks from [-30,30] to [-6,6], with the elite being all positive, dominating their audiences.

Zero-maturity capital-uniform non-bucketed simulations surround *dreps* now, with a FOF-uniform / global uniform one from before, as well as utility, no-FOF, global uniforms:

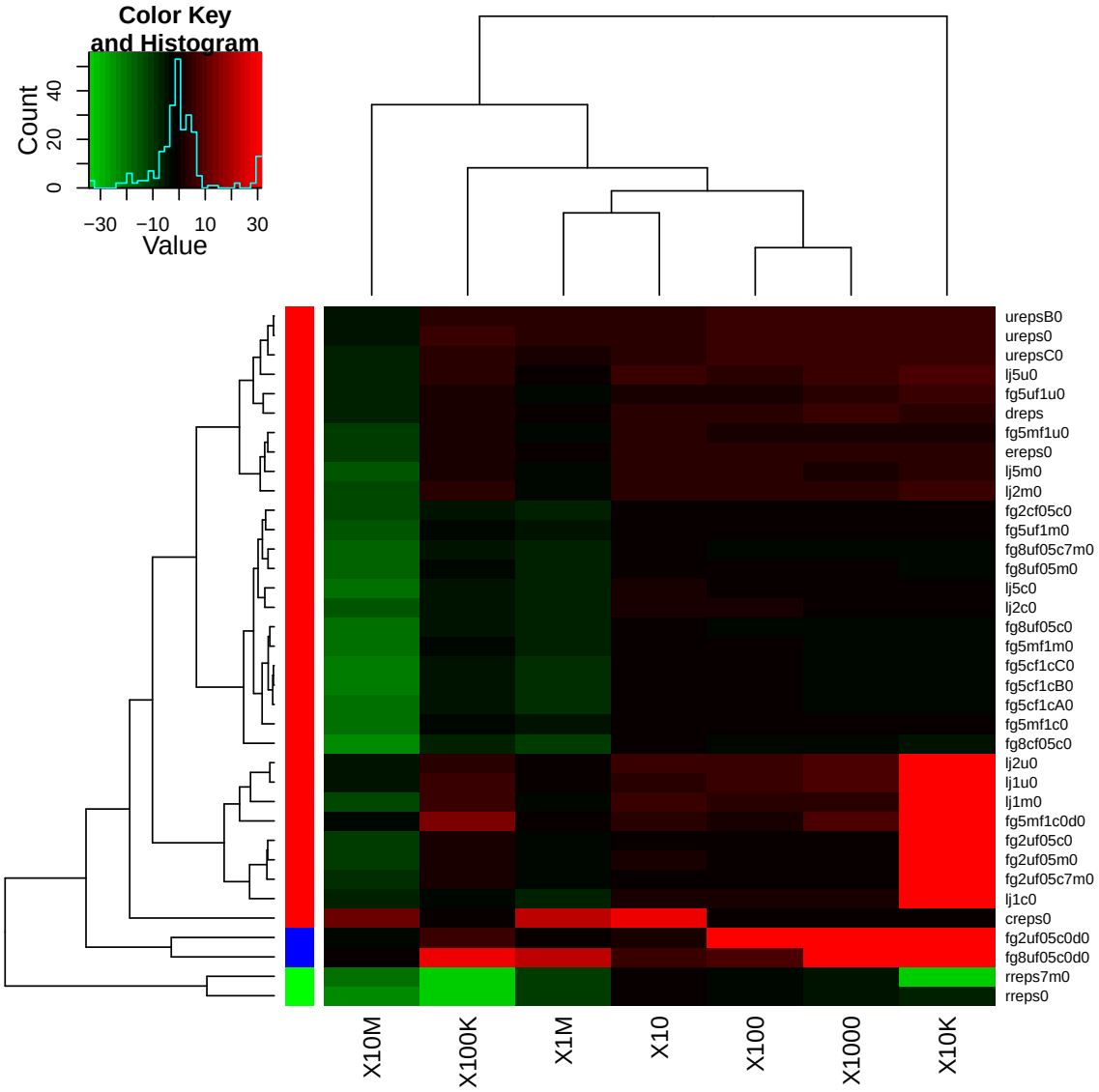

Figure 7.21: Starranks by replies for all simulations started from scratch.

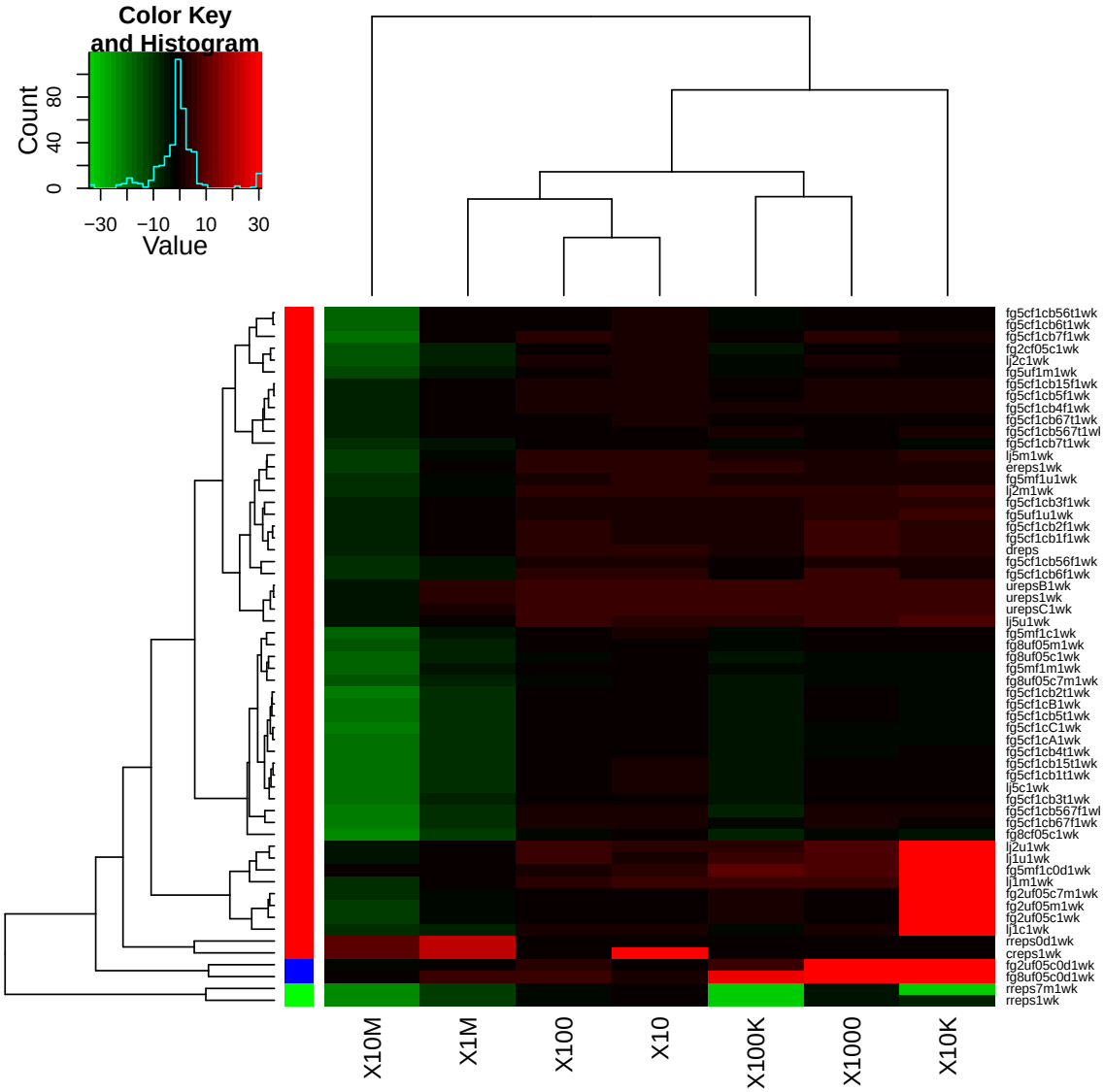

Figure 7.22: Starranks by replies for all simulations seeded by 1 week of reality.

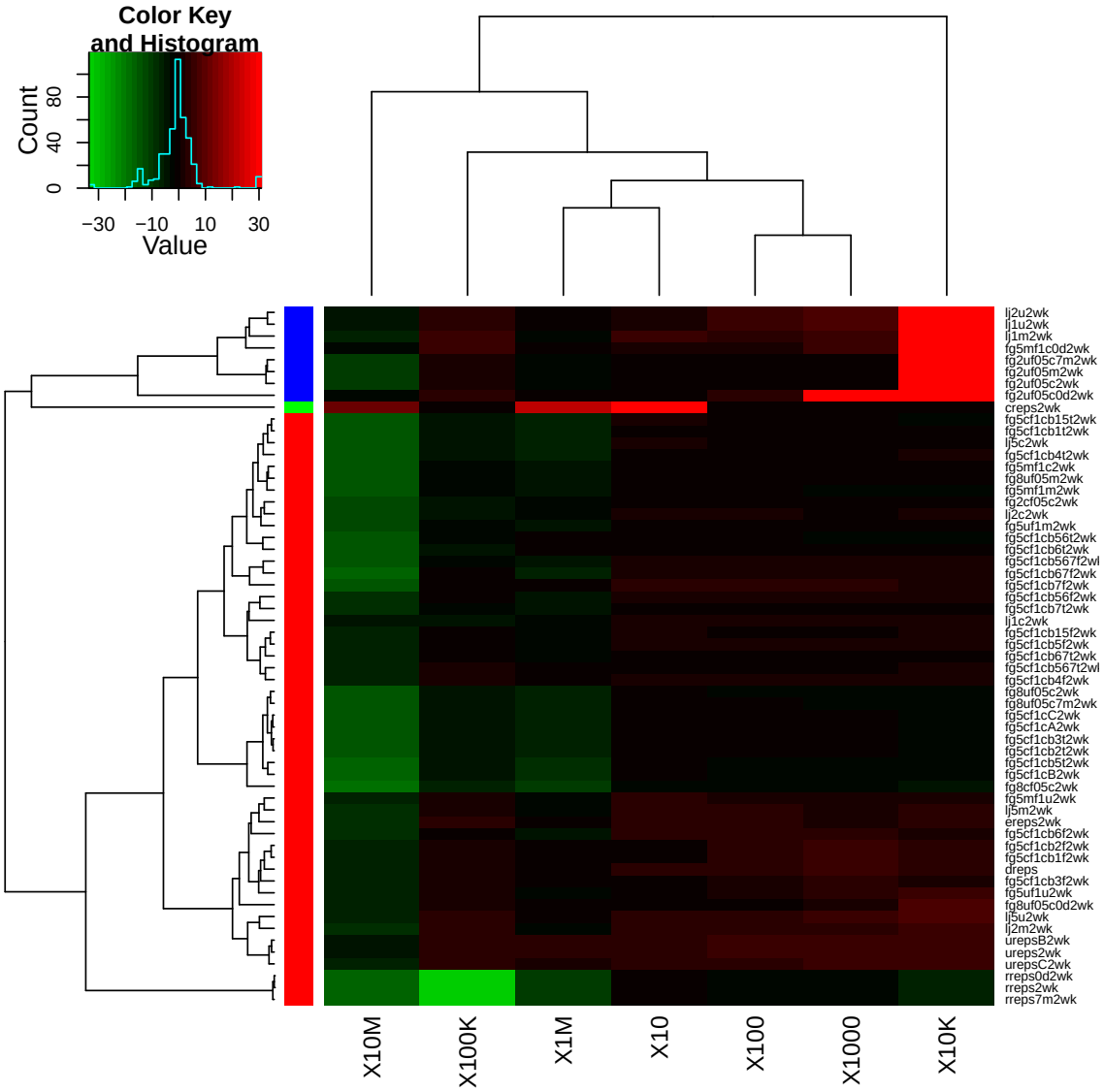

Figure 7.23: Starranks by replies for all simulations seeded by 2 weeks of reality.

- fg8uf05c0d
- fg2uf05c0d
- fg5mf1c0d
- fg5uf1u
- lj1,2,5m

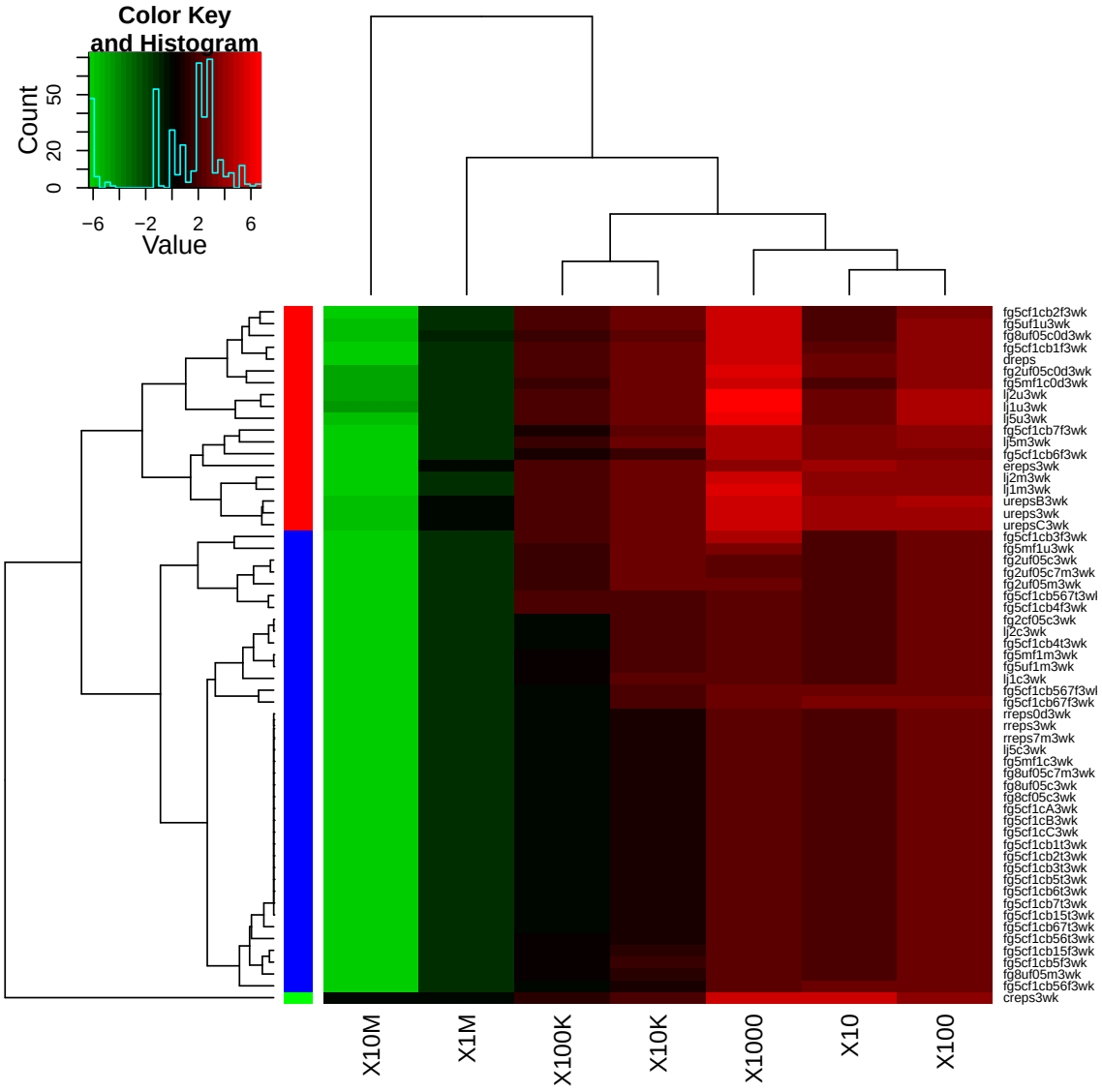

Figure 7.24: Starranks by replies for all simulations seeded by 3 weeks of reality.

Bucket to Bucket Communication

Bucket to Bucket Replies

Communication to the higher-class buckets proceeds as follows.

When we generate from scratch — Figure 7.25, — global-uniform strategies fall closer to reality at first, such as

- fg5uf1u
- lj1,2,5u
- ereps
- fg5mf1u

The first week, Figure 7.26, bucketed join with small elite changes *1,2,3f* but also the simulated poor, *7f*, cluster around it. Among the non-bucketed, neighbors include FOF mentions-uniform with global mentions-uniform strategies:

- fg5uf1u
- lj1,2,5u
- ereps
- fg5mf1u
- lj1,2,5m

The second week, Figure 7.27, besides the simulated elites, we get the upper-middle class, separately and with the celebrities, still the same non-bucketed favorites:

- fg5cf1cb15f
- fg5cf1cb5f

- fg5uf1u
- lj1,2,5u
- ereps
- fg5mf1u
- lj1,2,5m
- fg8uf05c0d
- fg2uf05c0d

Finally, the third week, Figure 7.28, shows homogeneity across simulations with the small simulated elites surrounding reality, and also

- lj1,2,5u
- fg5cf1cb15f
- fg5cf1cb5f
- fg5uf1u
- ereps

Communication with the lower-class buckets unfolds so we can immediately see, in Figure 7.29, that utility-based strategies, with global uniform or mentions, and also 0.5 probability of FOF uniform mix, show up near the *dreps* reality:

- fg5uf1u
- fg5mf1u
- lj1,2,5m
- ereps

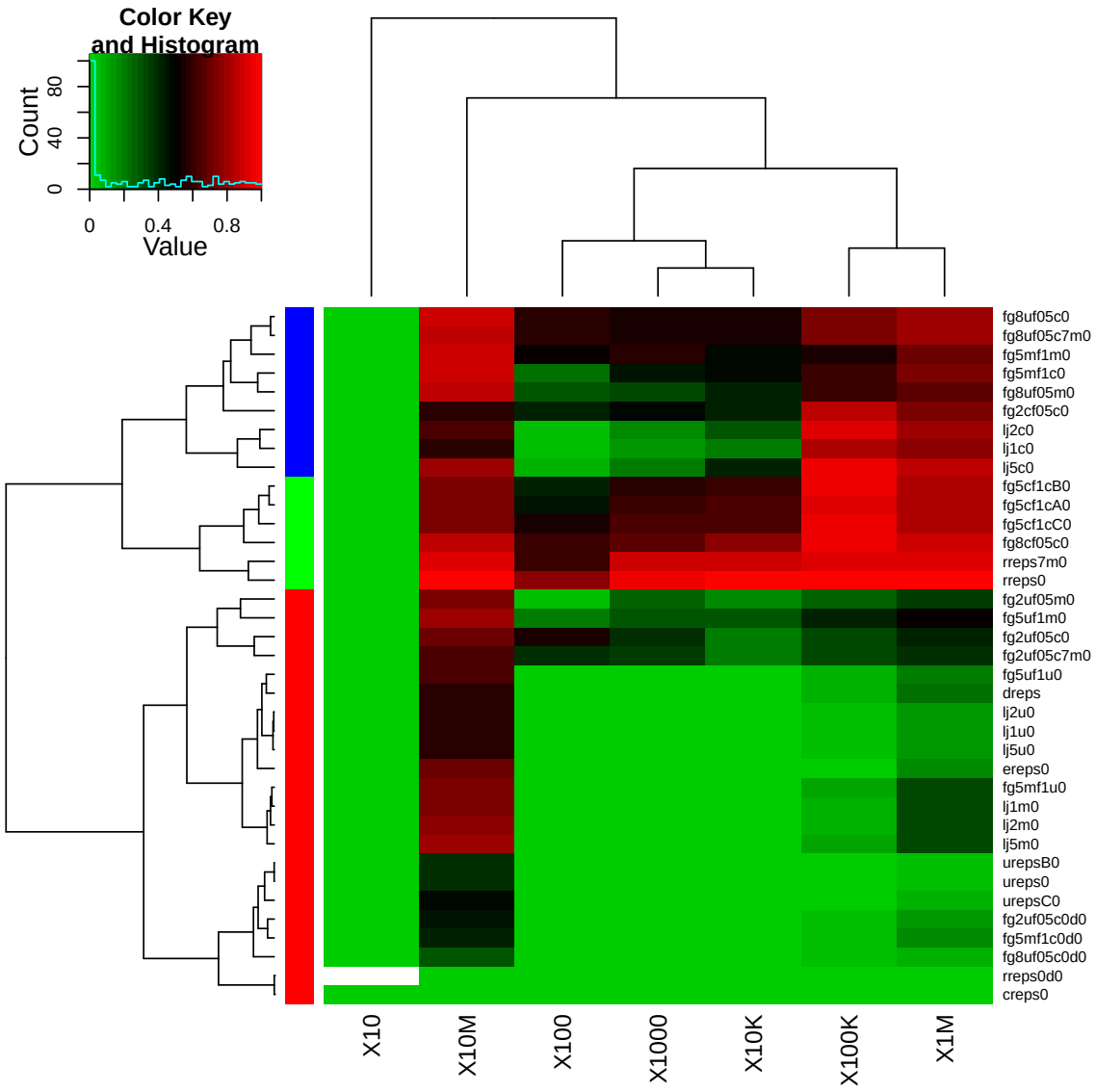

Figure 7.25: The fraction of all replies to higher buckets for all simulations started from scratch.

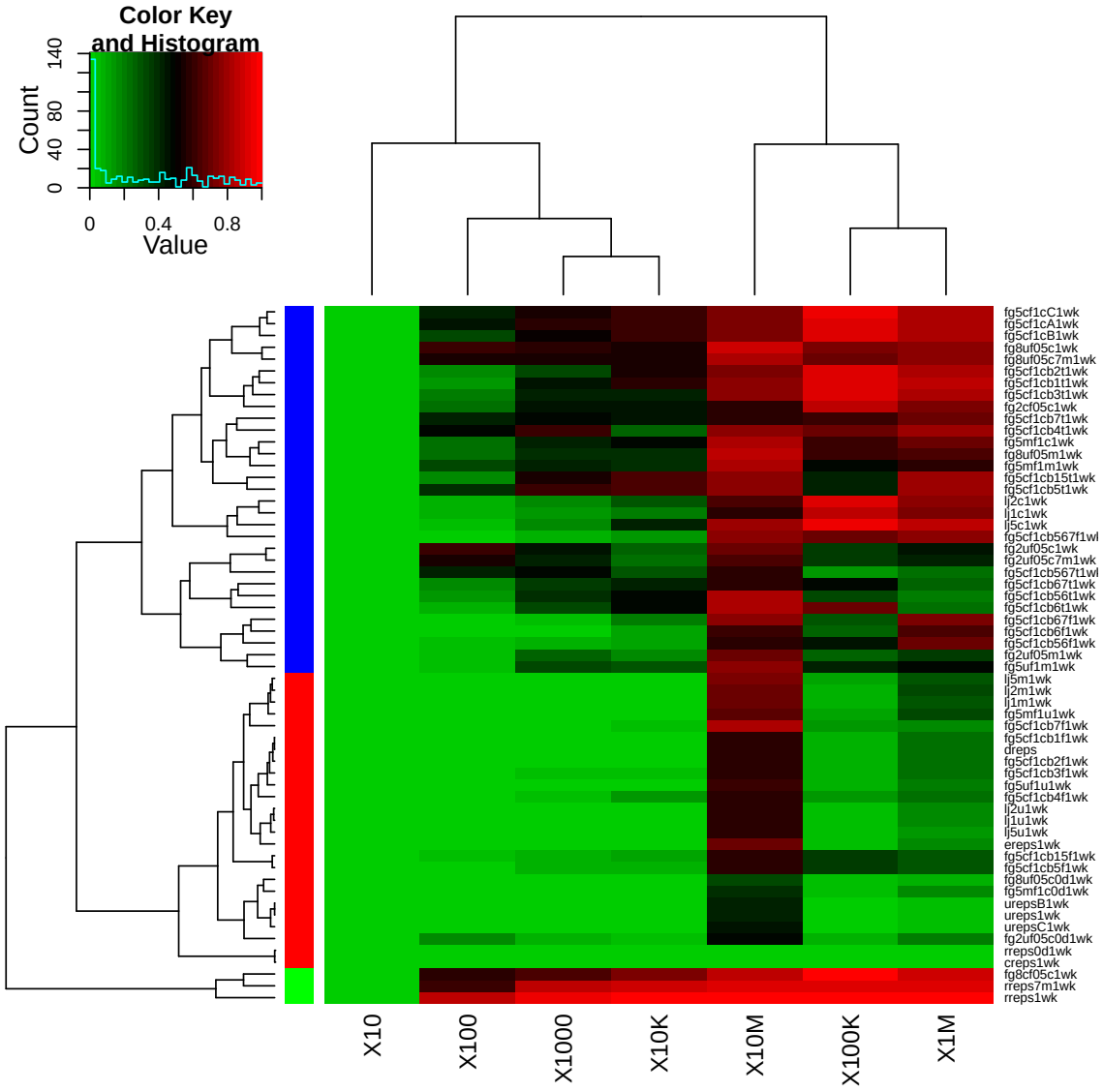

Figure 7.26: The fraction of all replies to higher buckets for all simulations seeded by 1 week of reality.

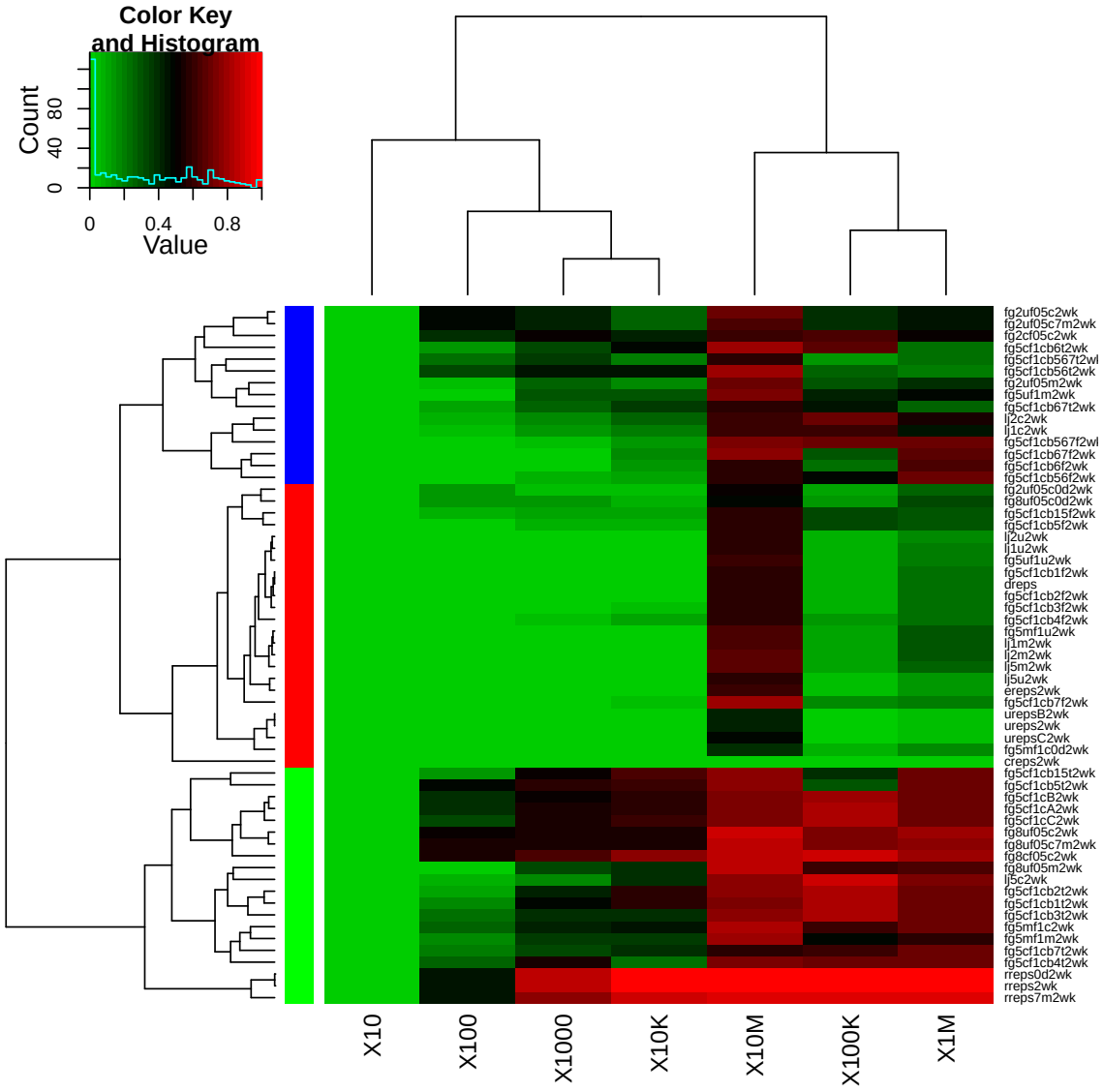

Figure 7.27: The fraction of all replies to higher buckets for all simulations seeded by 2 weeks of reality.

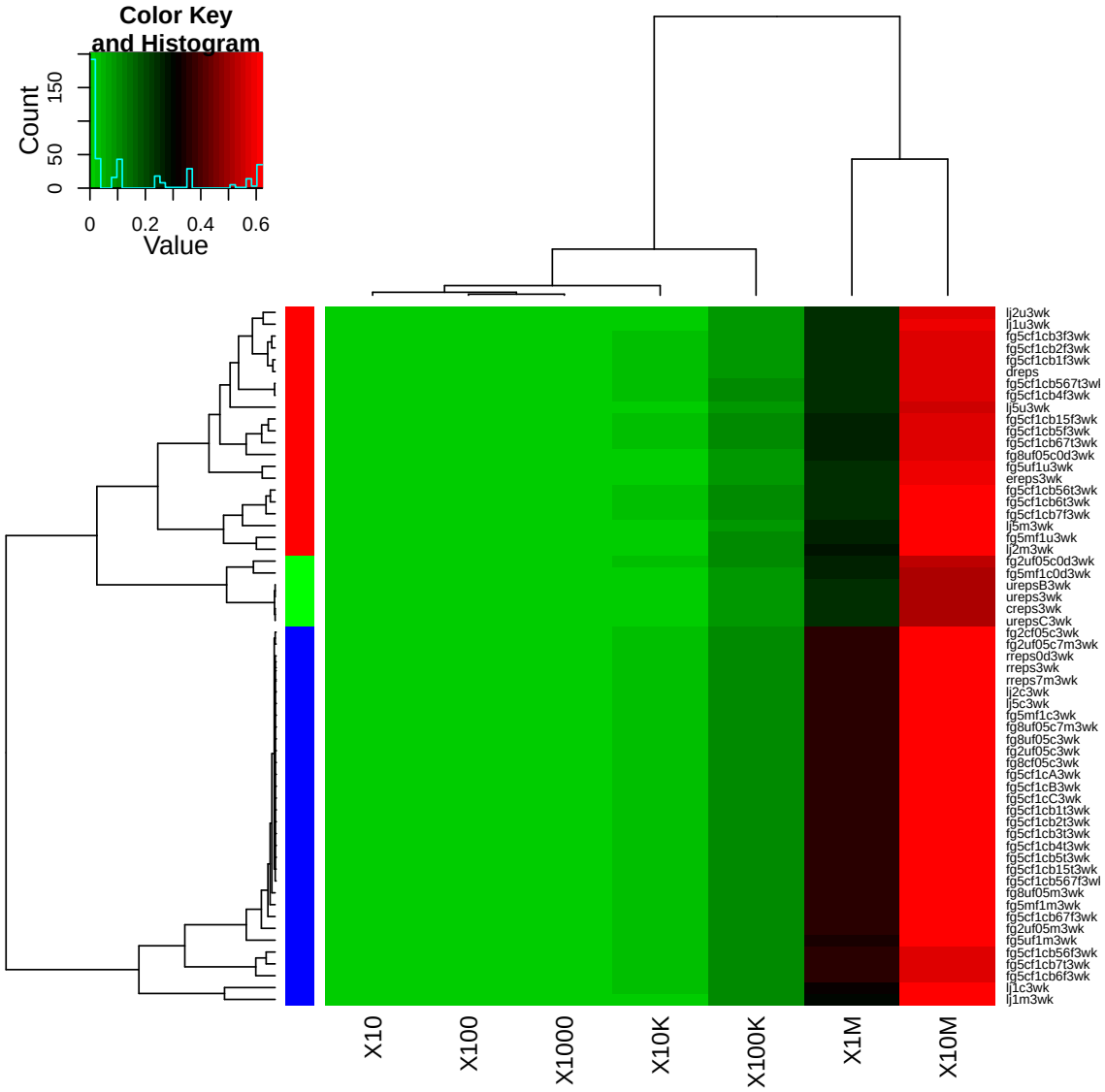

Figure 7.28: The fraction of all replies to higher buckets for all simulations seeded by 3 weeks of reality.

With first week of reality seeded as the starting conditions, as seen in Figure 7.30, we see bucketed simulations, and also higher ratios of talking down from the elite buckets. Small elites and the same winners (those with the highest starrank) as above are the closest to reality. Capital-based simulations cluster together, both bucketed and not, as we bucket the same base, *fg5cf1c*.

The second week, Figure 7.31, we see the same pattern, with a few more global uniform combinations of FOF and pure utility:

- lj1,2,5u
- fg8uf05c0d
- fg5mf1c0d

Finally, as we can see in Figure 7.32, the third week is fairly homogeneous and continues the evolving patterns, with the winners being small simulated elites and

- fg2uf05c0d
- fg5uf1u
- lj1u
- fg5mf1u
- lj1,2m

Communications which stay within the same bucket, for each bucket, develop as follows, with the ratios of all replies allocated to each bucket counted.

When simulating from scratch — Figure 7.33, — we get our by now usual group of winners:

- fg5uf1u
- fg5mf1u
- lj1,2,5m

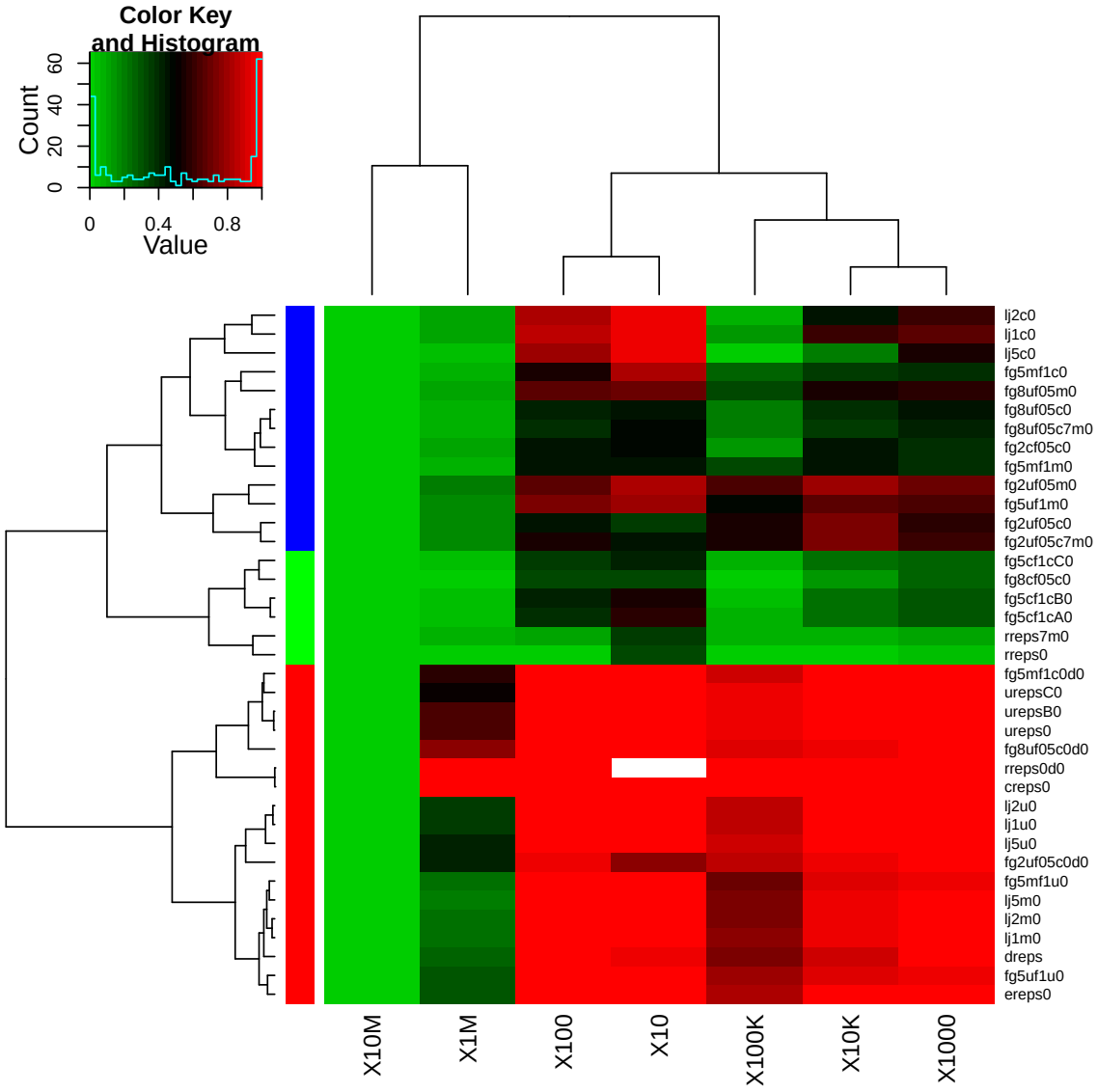

Figure 7.29: The fraction of all replies to lower buckets for all simulations started from scratch.

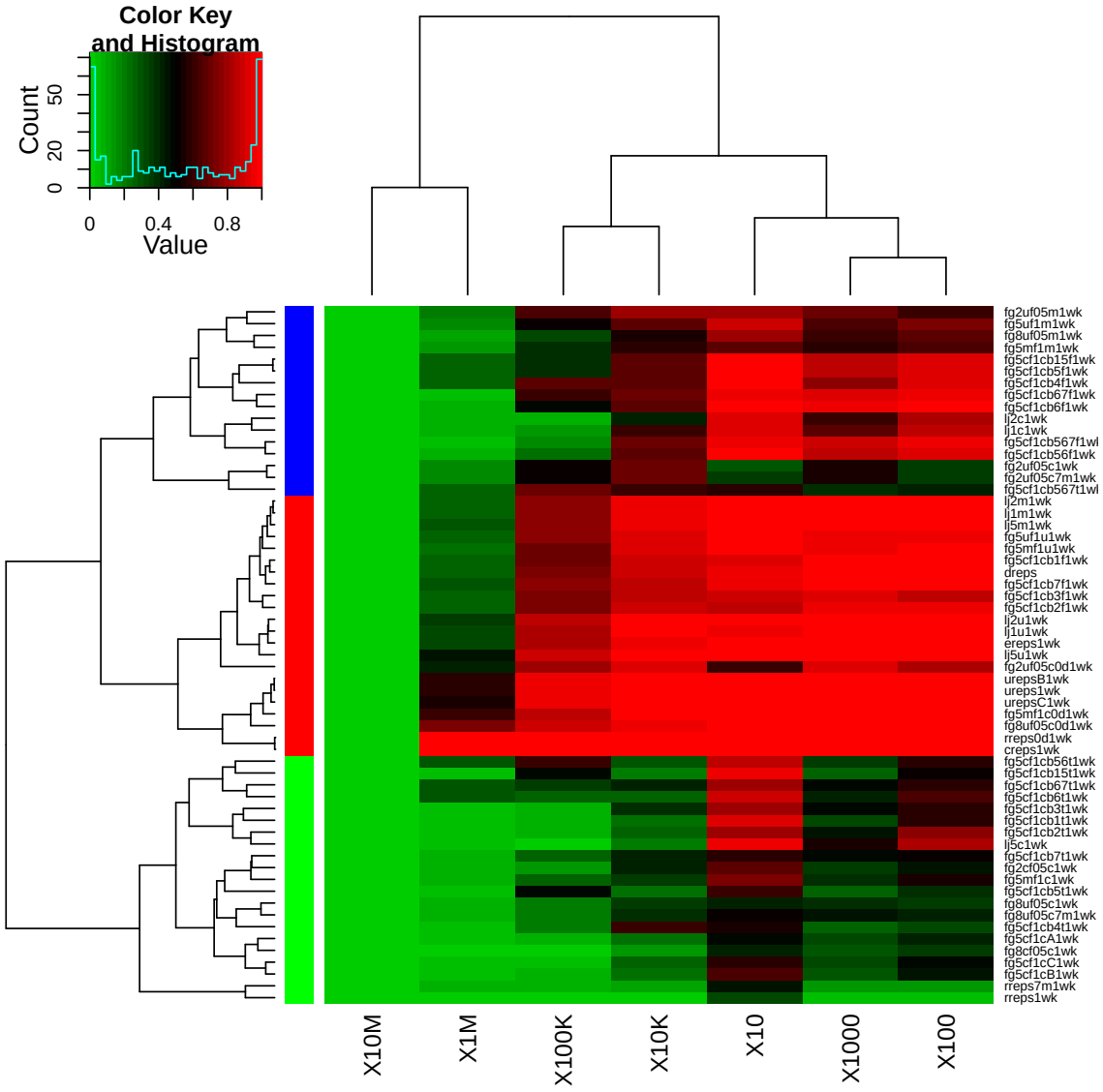

Figure 7.30: The fraction of all replies to lower buckets for all simulations seeded by 1 week of reality.

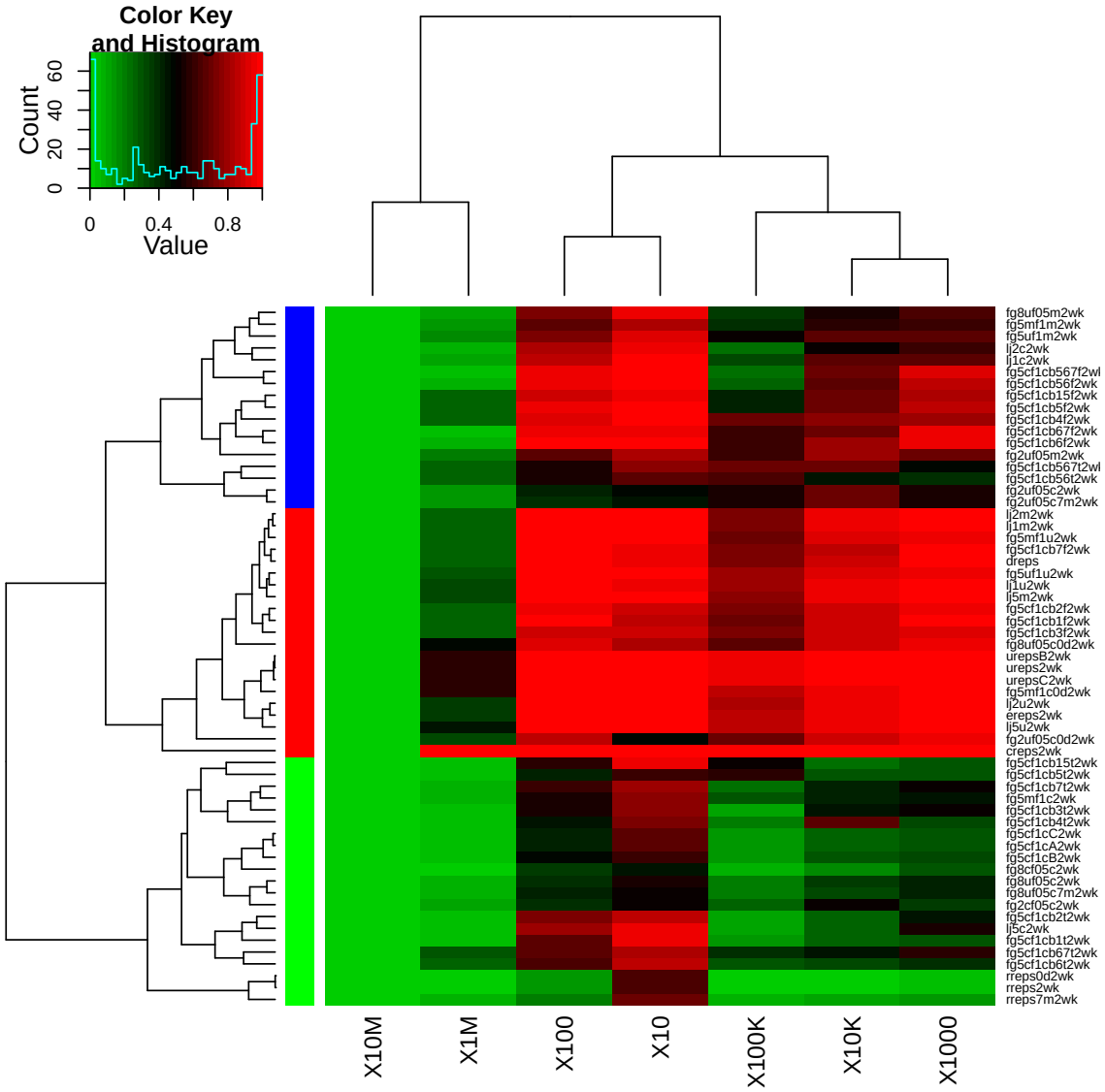

Figure 7.31: The fraction of all replies to lower buckets for all simulations seeded by 2 weeks of reality.

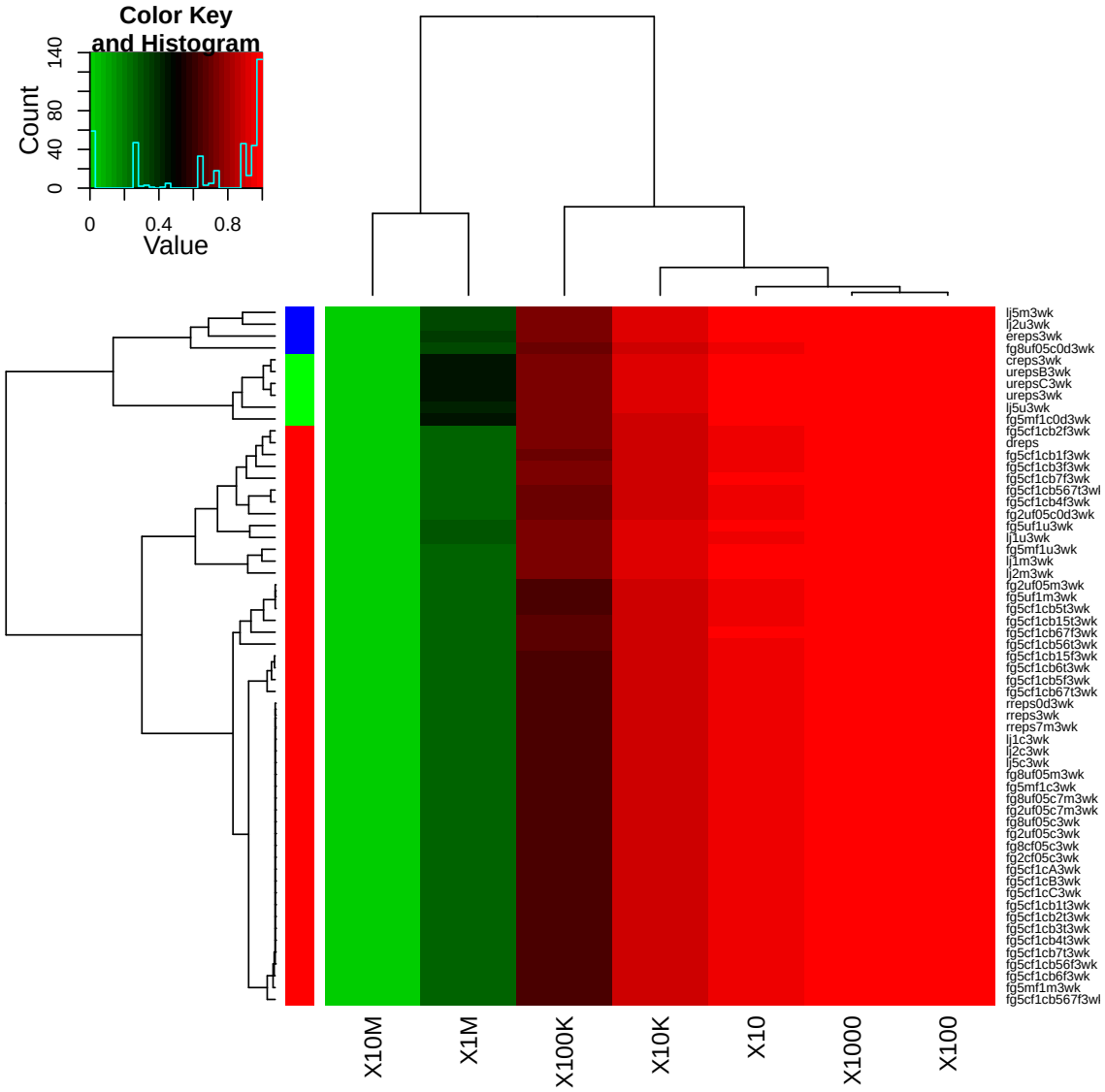

Figure 7.32: The fraction of all replies to lower buckets for all simulations seeded by 3 weeks of reality.

- ereps
- fg8uf05m
- lj1,2,5c

The first week — Figure 7.34 — adds the bucketed simulations, but they don't surround the reality just yet! Our winners remain as above, with the small elites joined nearby by the simulated poor, and the simulated celebrity and upper-middle class combo, followed by just the latter:

- fg5cf1cb7f
- fg5cf1cb15f
- fg5cf1cb5f

The same winners persist through the second week — Figure 7.35 — to the third — Figure 7.36, — when they are joined by the ebullient simulated middle class and immature capital-uniform FOF:

- fg5cf1cb6f
- fg8uf05c0d

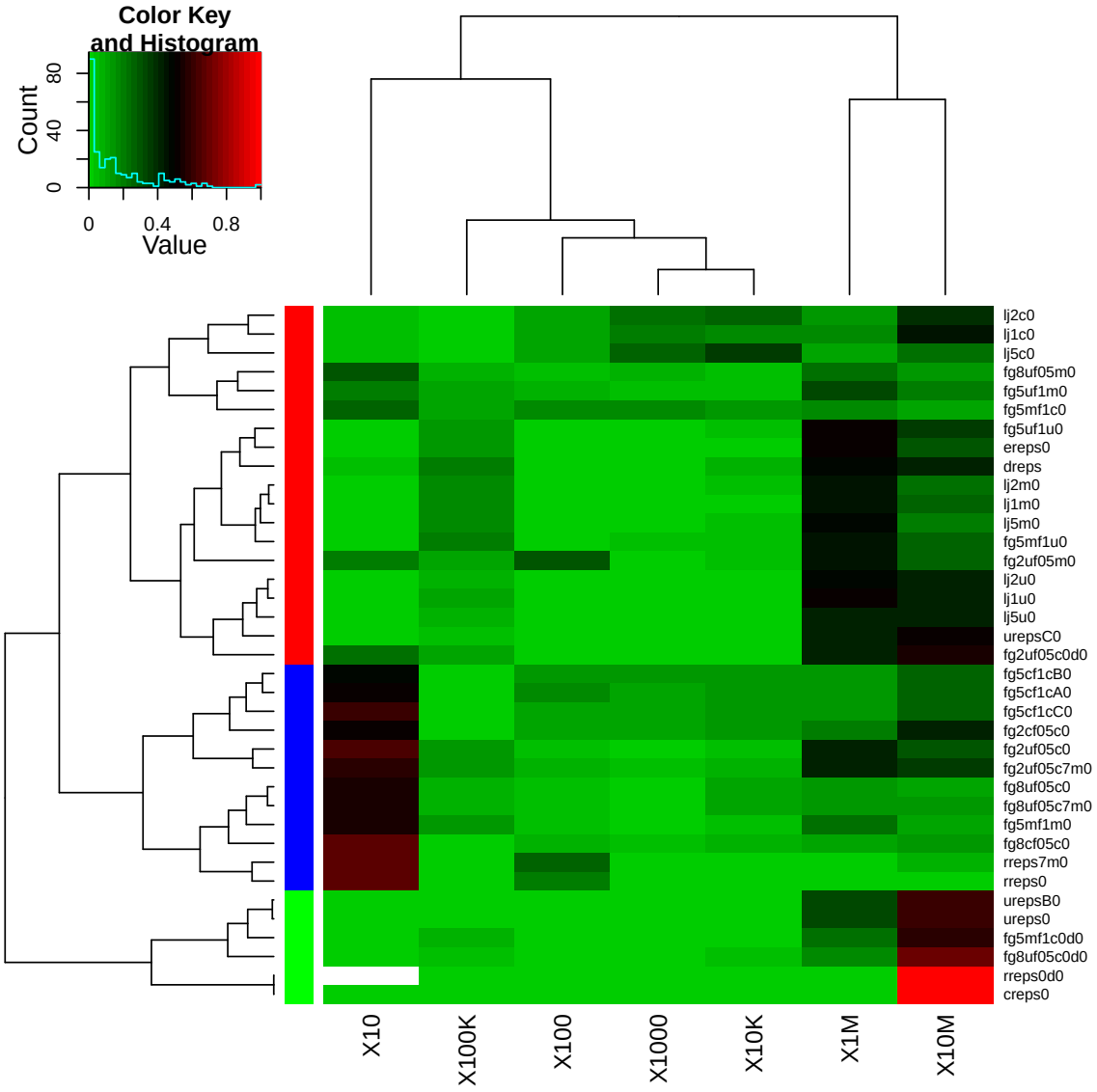

Figure 7.33: The fraction of all replies to the same bucket for all simulations started from scratch.

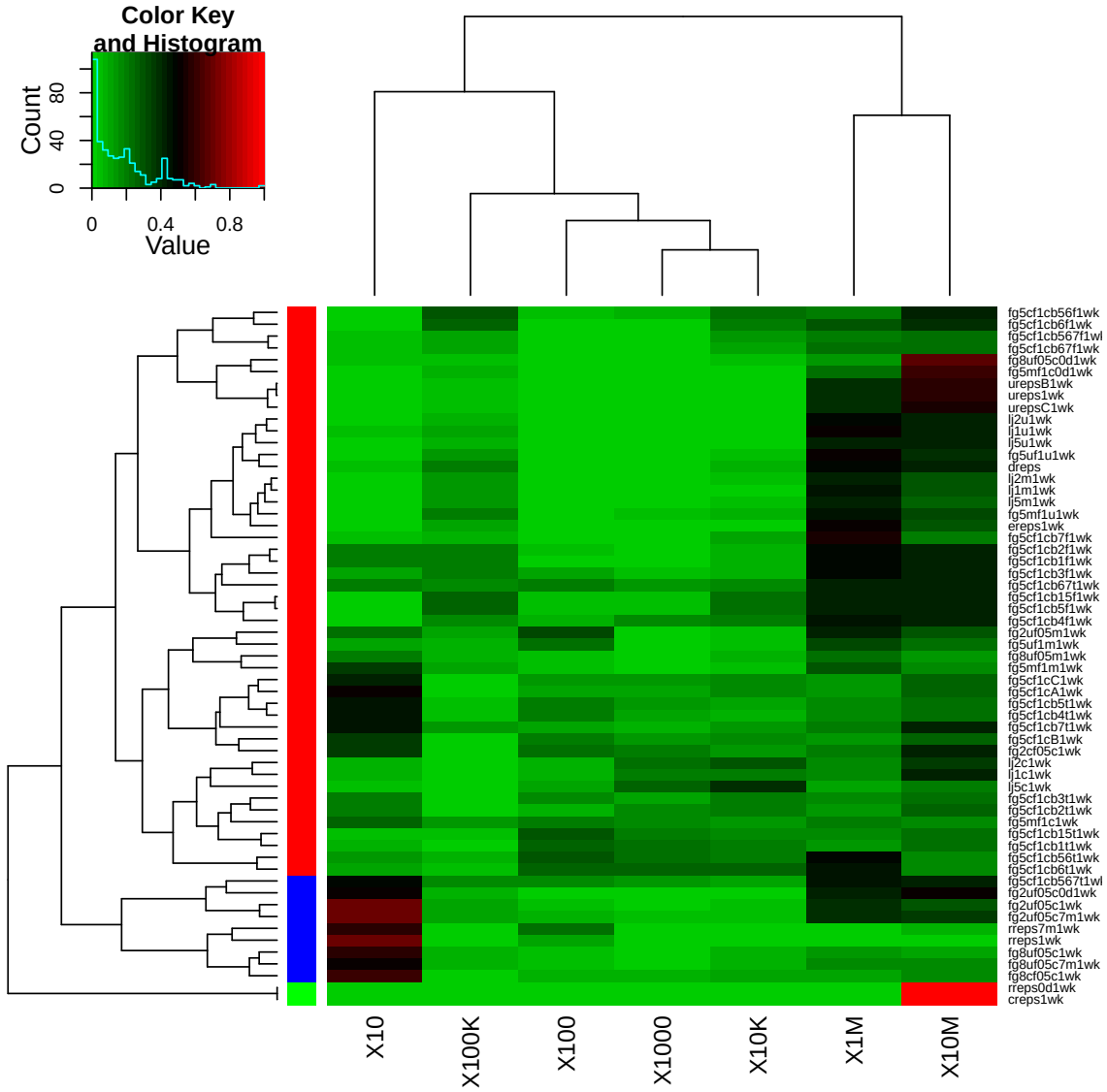

Figure 7.34: The fraction of all replies to the same bucket for all simulations seeded by 1 week of reality.

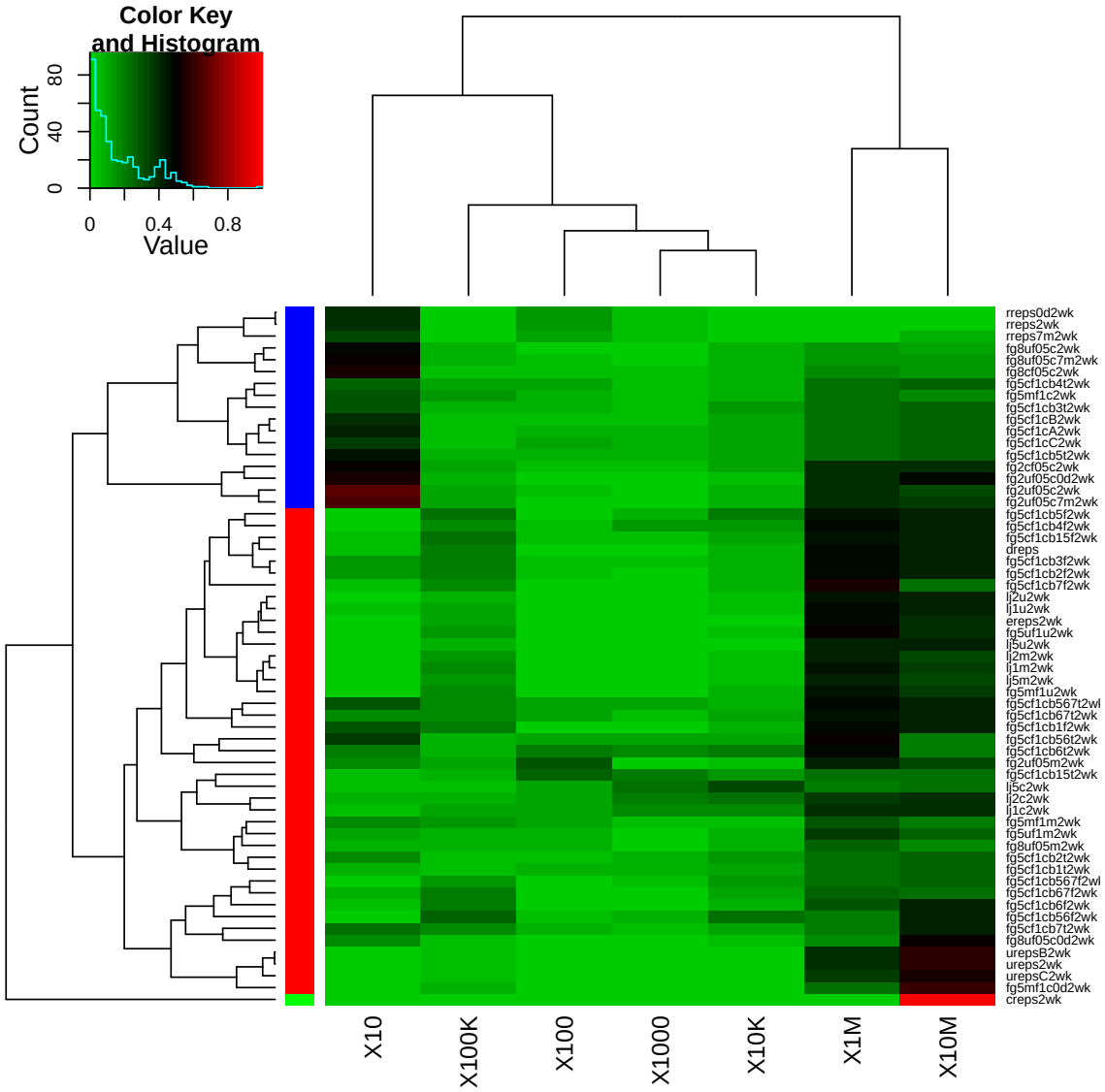

Figure 7.35: The fraction of all replies to the same bucket for all simulations seeded by 2 weeks of reality.

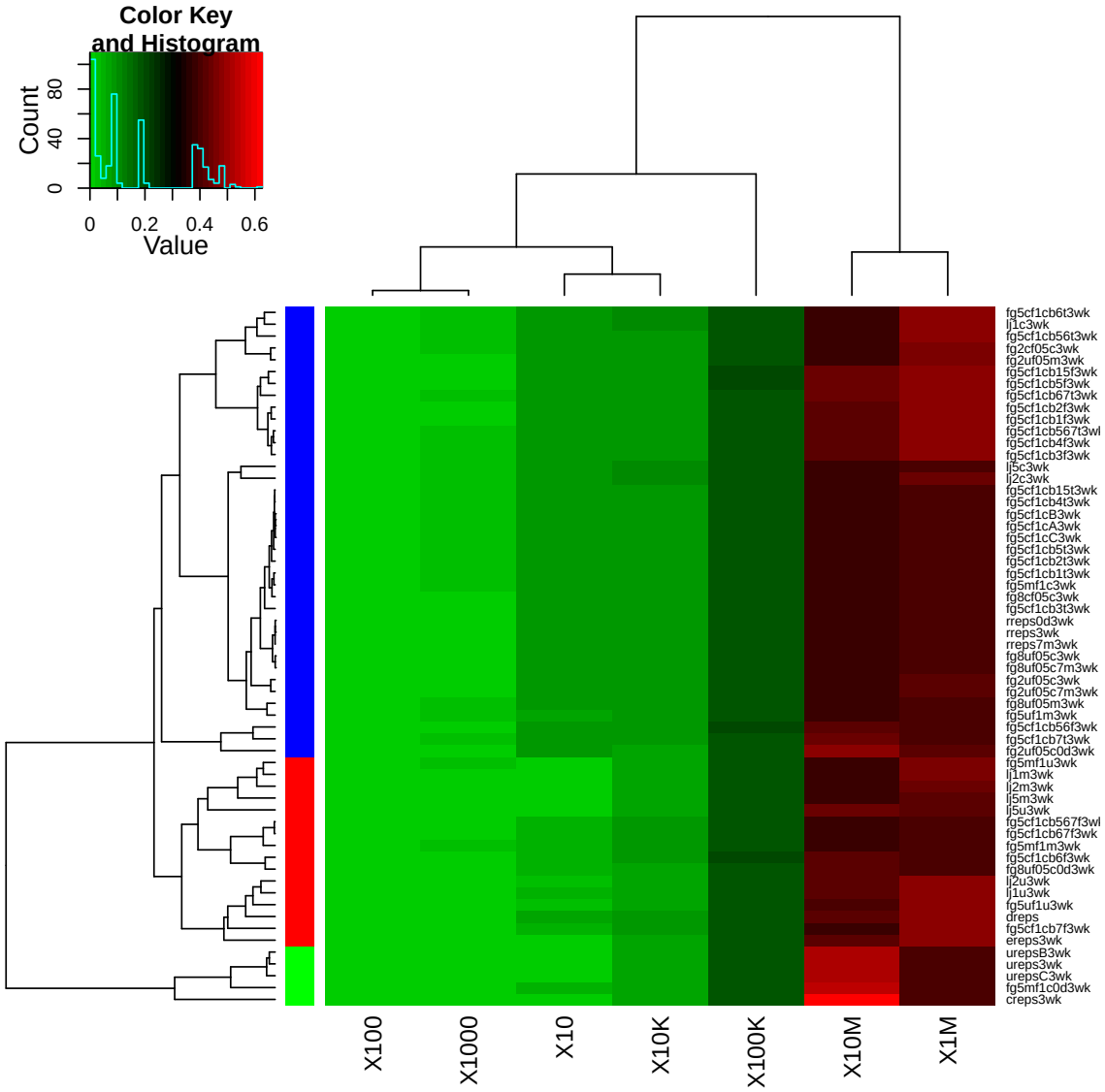

Figure 7.36: The fraction of all replies to the same bucket for all simulations seeded by 3 weeks of reality.

Bucket to Bucket Mentions

Since bucket to bucket mentions from the higher classes are few and decrease proportionally to the target bucket size we do a log transform on the ratios.

When generating from scratch, Figure 7.37, we get a somewhat different set of nearest neighbors than for replies:

- fg5mf1u
- fg5mf1c
- fg5mf1m
- fg8uf05m
- fg5mf1c0d

The first week, Figure 7.38, brings the simulated middle class right next to reality. The capital-based block is nearby. The second week, Figure 7.39, keeps the middle class next to *dreps*, as well as:

- fg8uf05m
- fg5uf1m
- fg5cf1cb67f
- fg5cf1cb7f
- fg5cf1cb567f

Finally, by the third week, Figure 7.40, the middle class is still the closest winner, among the following:

- fg5cf1cb6f
- fg2uf05c

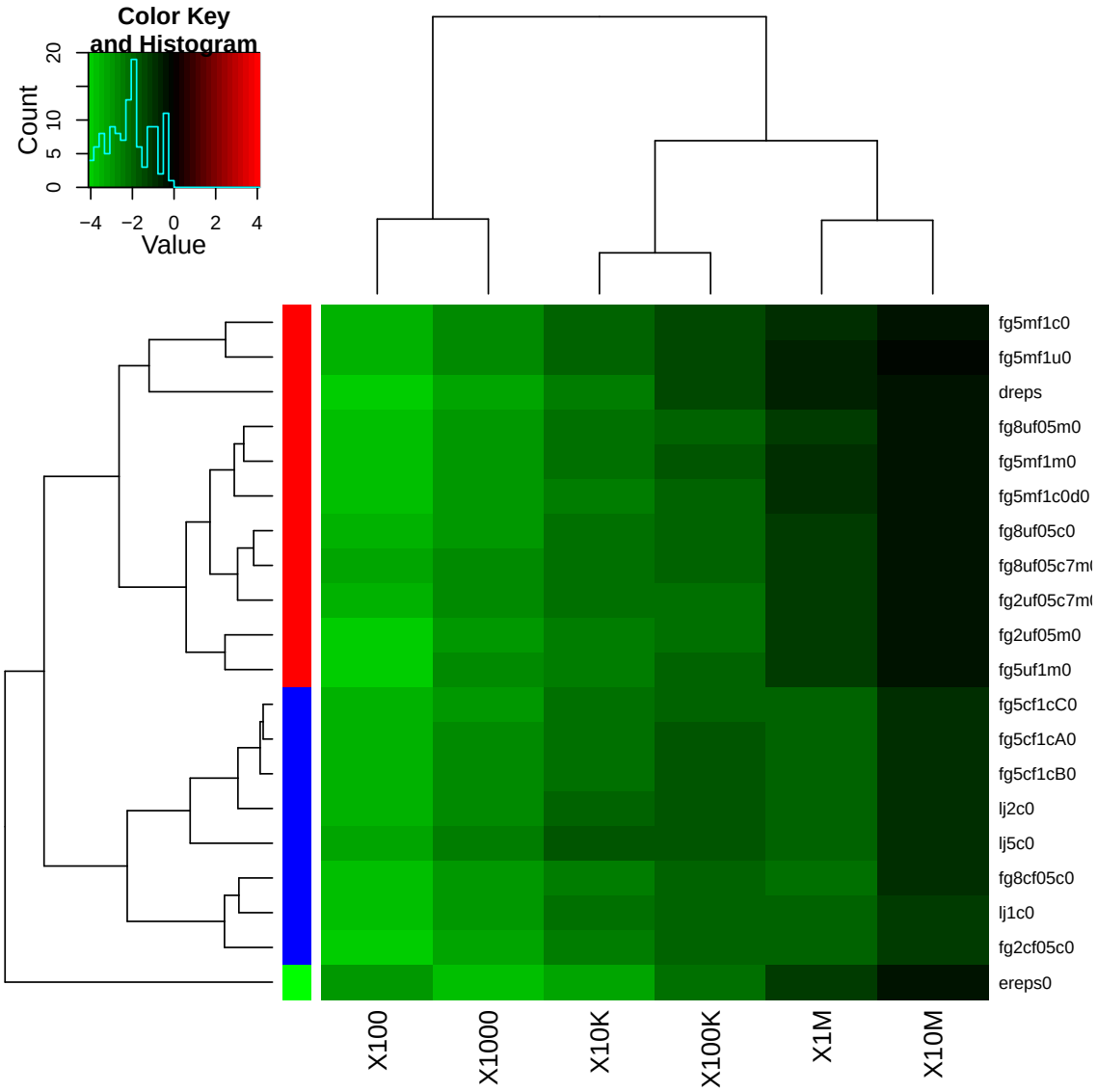

Figure 7.37: The fraction of all mentions from higher buckets for all simulations started from scratch.

- fg8cf05c
- fg5cf1cb56f
- rreps
- rreps0d

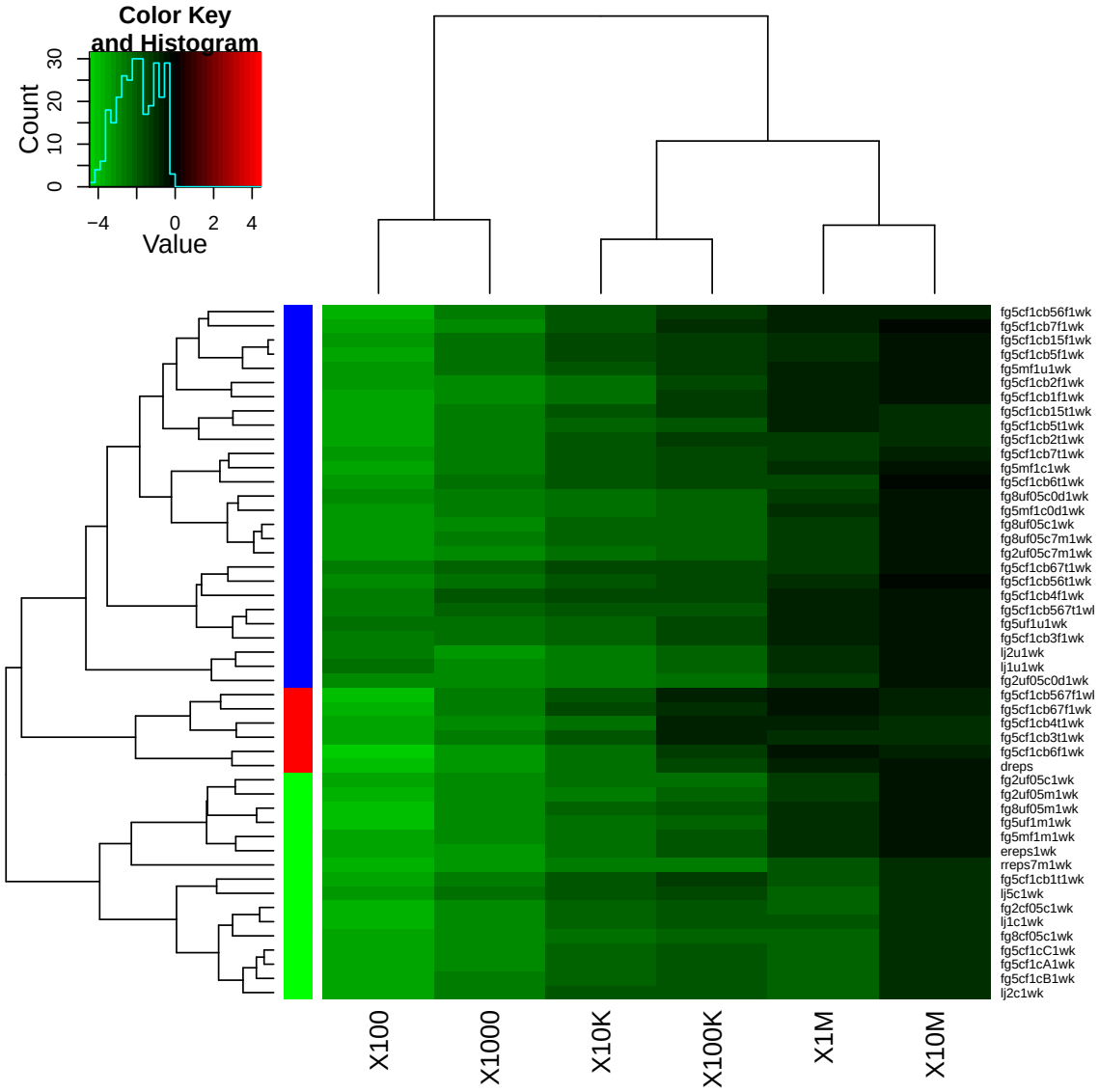

Figure 7.38: The fraction of all mentions from higher buckets for all simulations seeded by 1 week of reality.

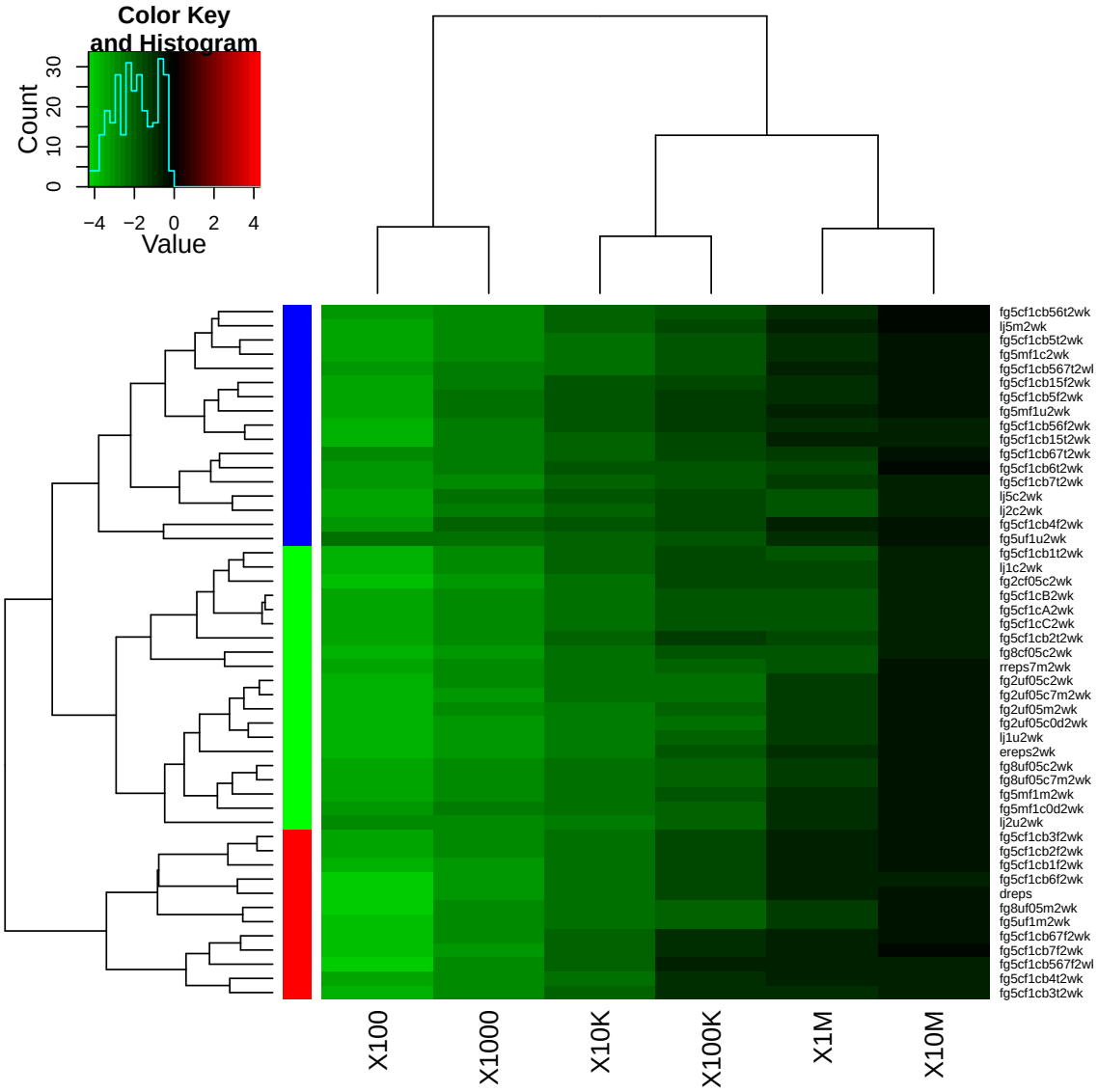

Figure 7.39: The fraction of all mentions from higher buckets for all simulations seeded by 2 weeks of reality.

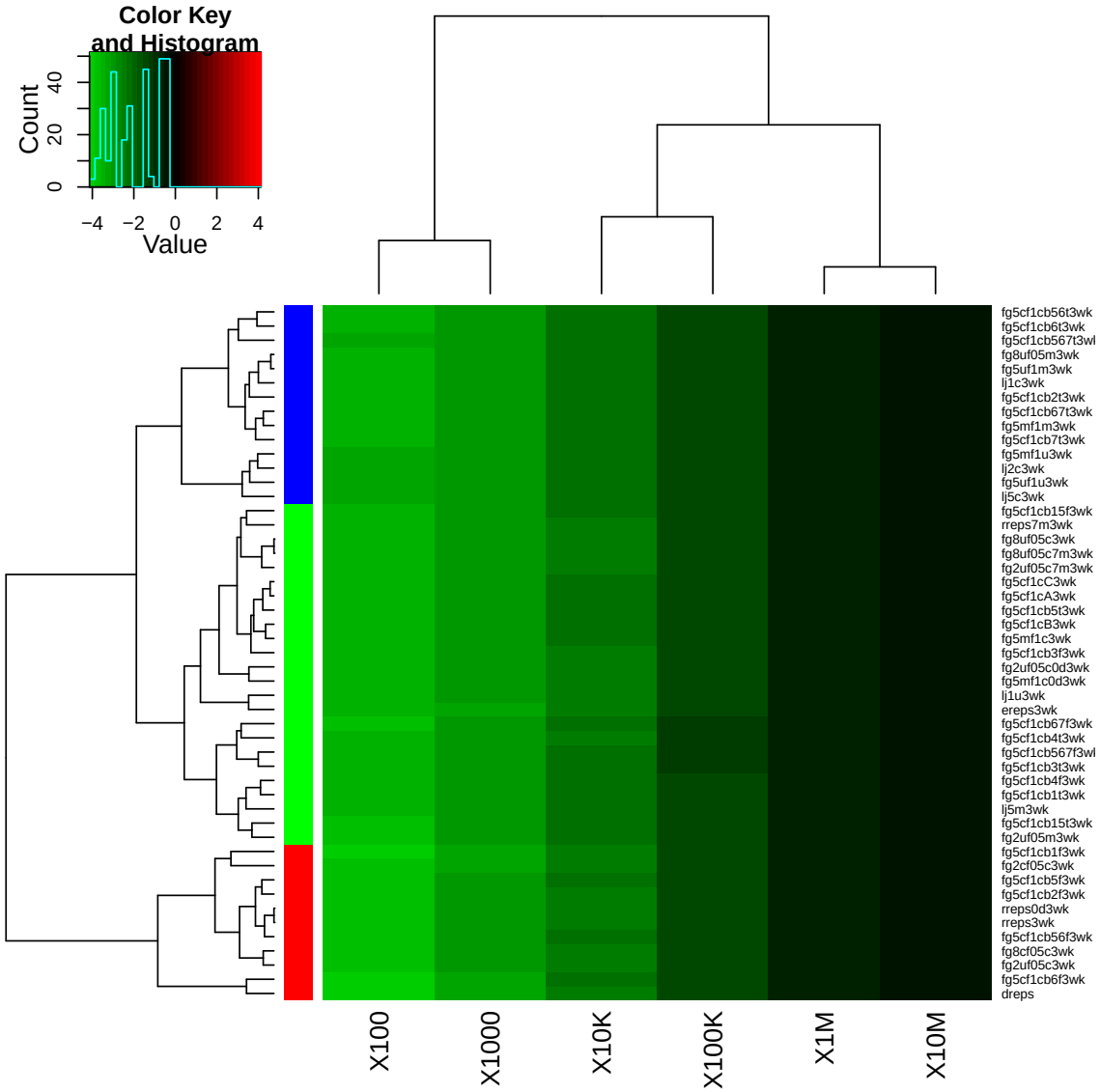

Figure 7.40: The fraction of all mentions from higher buckets for all simulations seeded by 3 weeks of reality.

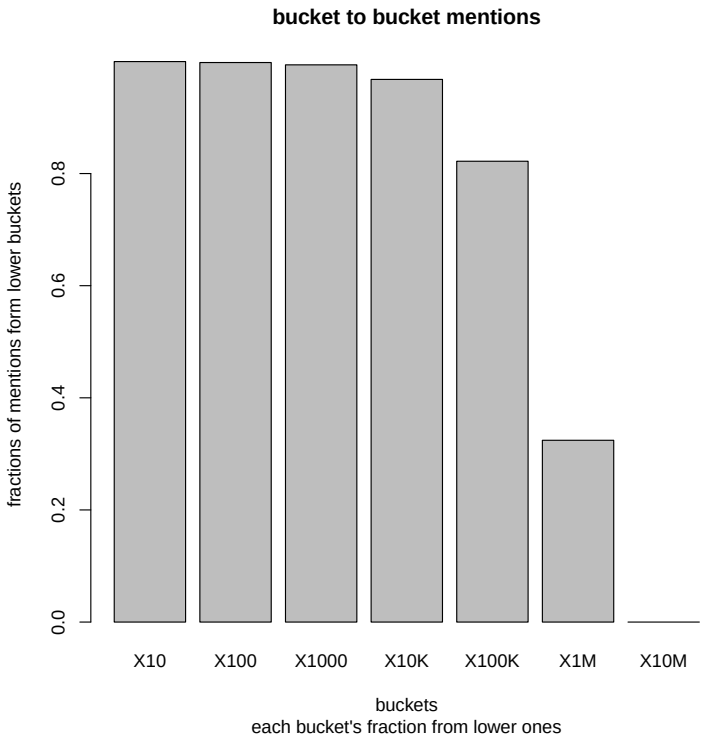

Figure 7.41: Medians of all bucket to bucket mentions, from lower ones than each one, for all simulations seeded by 3 weeks of reality, per bucket.

The pattern is very uniform here in that every bucket is mentioned by its predecessors, so that we are only showing the typical values. The bar plot in Figure 7.41 shows the values per bucket of the fraction of all mentions this particular bucket received from lower classes, aggregated as medians for bucket classes across all simulations.

The fraction of mentions, staying in the same bucket, progresses as follows.

In simulations starting from scratch, Figure 7.42, the cluster neighbor of reality is

- fg2uf05m

The first week, Figure 7.43, the winners are

- lj5c1
- fg5cf1cb7t

- fg5uf1m
- fg5cf1cb7f
- fg8uf05m
- fg5cf1cb567f
- fg5cf1cb67f

The second week, Figure 7.44, the near neighbors of *dreps* are

- fg2cf05c
- lj1c2
- fg2uf05c0d
- fg8uf05m

The third week, Figure 7.45, these simulations are in proximity to *dreps*:

- fg2uf05m
- fg5cf1cb2f
- fg5cf1cb15f
- fg5cf1cb67t
- dreps
- urepsB
- ureps3
- fg2cf05c

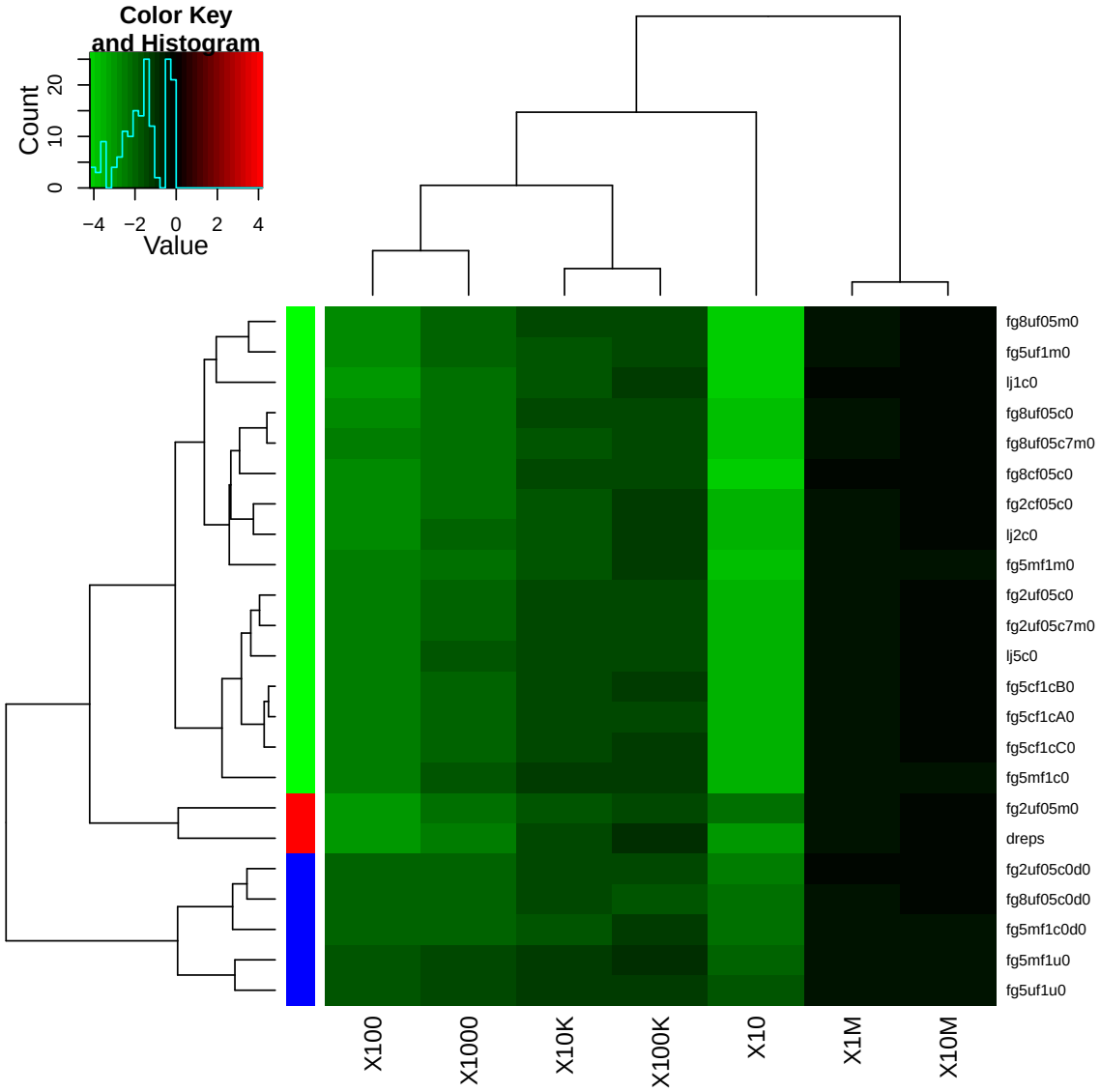

Figure 7.42: The fraction of all mentions from the same bucket for all simulations started from scratch.

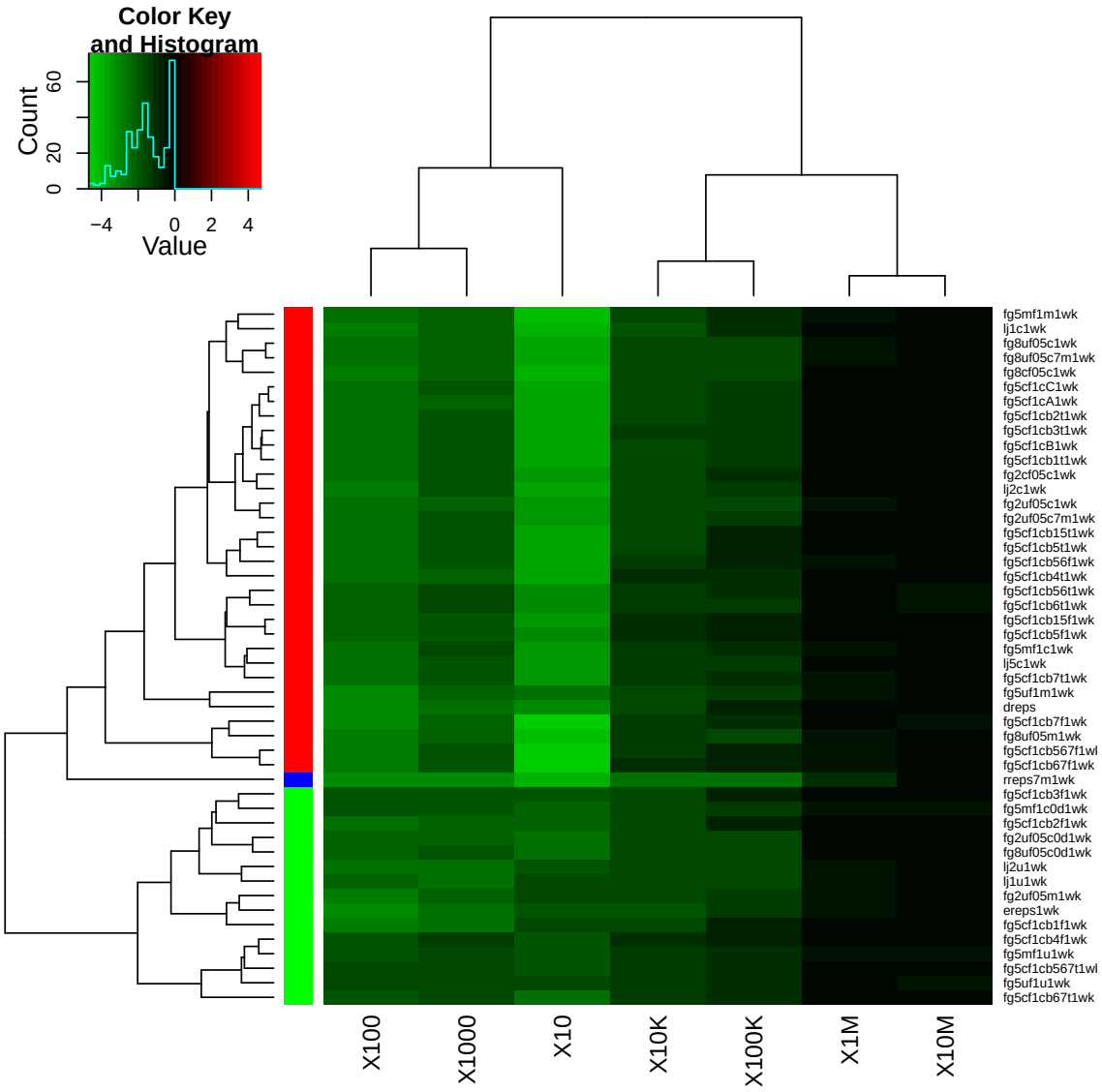

Figure 7.43: The fraction of all mentions from the same bucket for all simulations seeded by 1 week of reality.

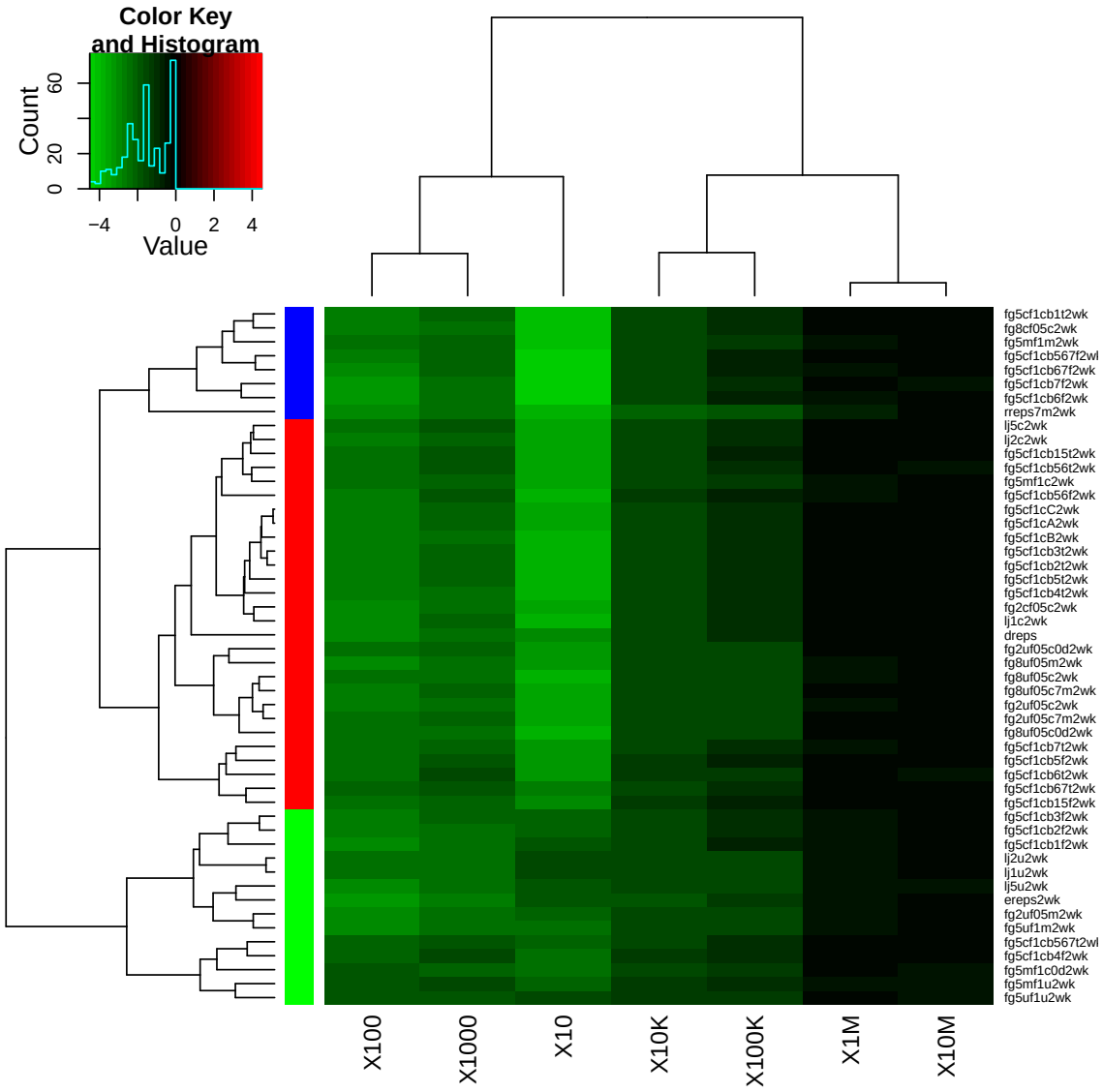

Figure 7.44: The fraction of all mentions from the same bucket for all simulations seeded by 2 weeks of reality.

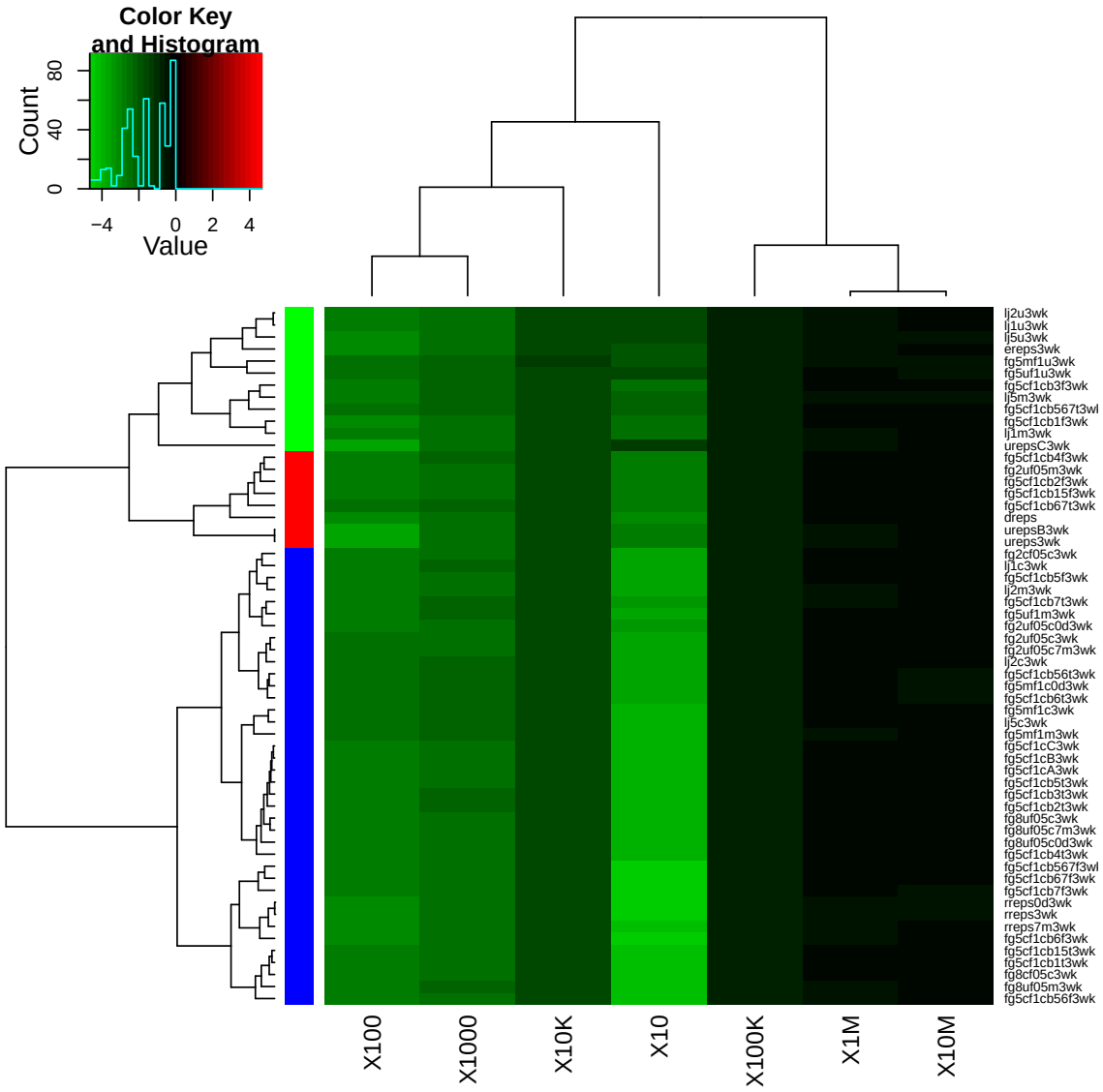

Figure 7.45: The fraction of all mentions from the same bucket for all simulations seeded by 3 weeks of reality.

Skew Correlation

Global Skew Correlation

We look at τ alignment for the full lists of all users and their capitals and skews for each day, and also perform τ alignments per bucket. For most of our strategies, we observe a Kendall's τ correlation between social capial and skew of about 0.2. It reaches 0.3 for *fg5m1u*, and is lower for *fg5c1cA* (and B,C) seeded with two or less weeks. The fact that the global τ generally is around 0.2, as it is for *dreps*, shows that politician-like behavior, with more communication activity toward the top contributors, corresponds to the social capital hierarchy.

Simulating from scratch, Figure 7.47, we observe a significant amount of low correlations among the global-only simulations and a progressive drop for capital-capital simulations. FOF-based simulations, with 0.05 jumpFOF probability, and global uniform strategy stay nearest the reality.

One week of reality seeding, Figure 7.48, brings about the bucketed simulations which cluster around *dreps* as expected, starting with small elite buckets. Interesting early neighbors are

- lj1u
- fg5mf1c

– followed by a block of the original capital-uniform simulations with large FOF probability (i.e. low 0.05 jumpFOF probability of going global away from FOF).

Capital-capital simulations, together with *rreps* and *creps*, are in a more distant cluster, falling rapidly with time.

The simulated middle class, *fg5cf1cb6f*, results in a separate cluster with

- lj2,5u
- lj1,2,5m
- fg5uf1m,u
- fg5mf1c0d,u

Their decline in time is slower than *dreps*, as opposed to capital-capital ones.

The picture consolidates after two weeks of reality, Figure 7.49, with a large cluster around *dreps*. Now we see simulated middle classes and elites, *fg5cf1cb7t*, as well as the simulated poor, *fg5cf1cb7f*, next to each other and near *dreps*. This indicates that the middle class is approximated quite well by the capital-capital strategy in relation to the poor.

The middle class, simulated by itself, *fg5cf1cb6f*, is between the previous favorites:

- fg5mf1c0d,c
- lj1,2u
- fg2uf05m
- fg5uf1m
- fg5cf1cb6f

Finally, after three weeks of reality, Figure 7.50, all but one non-bucketed capital-capital simuations are not in the distant cluster now left to globals only — *ureps*, *ereps*, *creps*, *rreps*, and *fg8cf05c*.

The preserved middle class by itself, *fg5cf1cb6t*, is closer to reality than simulated-only, *fg5cf1cb6f*, which is near the edge of the adjacent cluster next to *fg5mf1u*. FOF-based strategies with high probability of FOF attachment, 0.95, and high probability of jumping away from the initial utility, 0.8, are the closest to reality after the bucketed elites, namely

- fg8uf05m,c7m,c0d,c

Among the bucketed simulations, nearest non-elite ones are

- fg5cf1cb15f
- fg5cf1cb7t
- fg5cf1cb56f

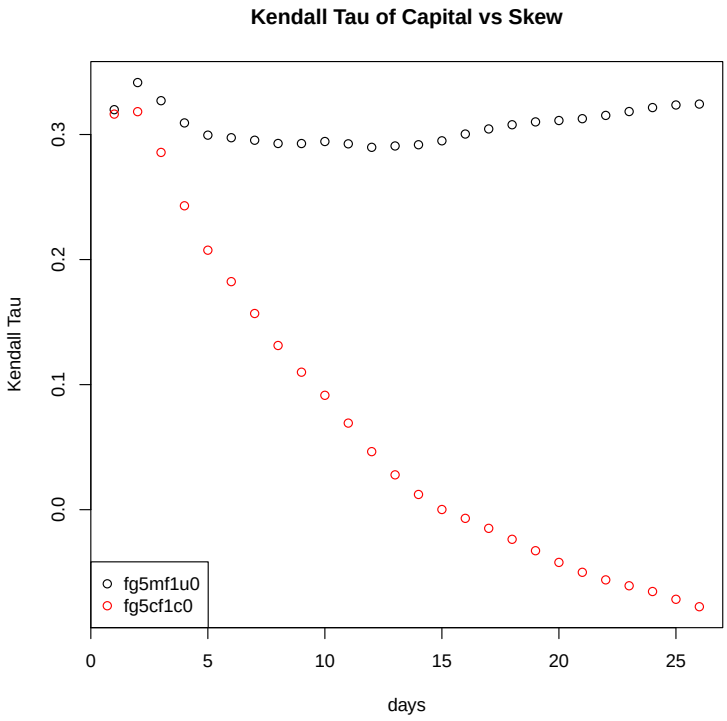

Figure 7.46: Global correlation of social capital and skew by Kendall τ. While generally staying around 0.2-0.3, it can also decrease for some strategies, like the all-capital one.

Simulating the middle and upper middle class, just two or with all the elites, with our capital-capital strategies turns out to be a good approximation of politician-like behavior, by skew.

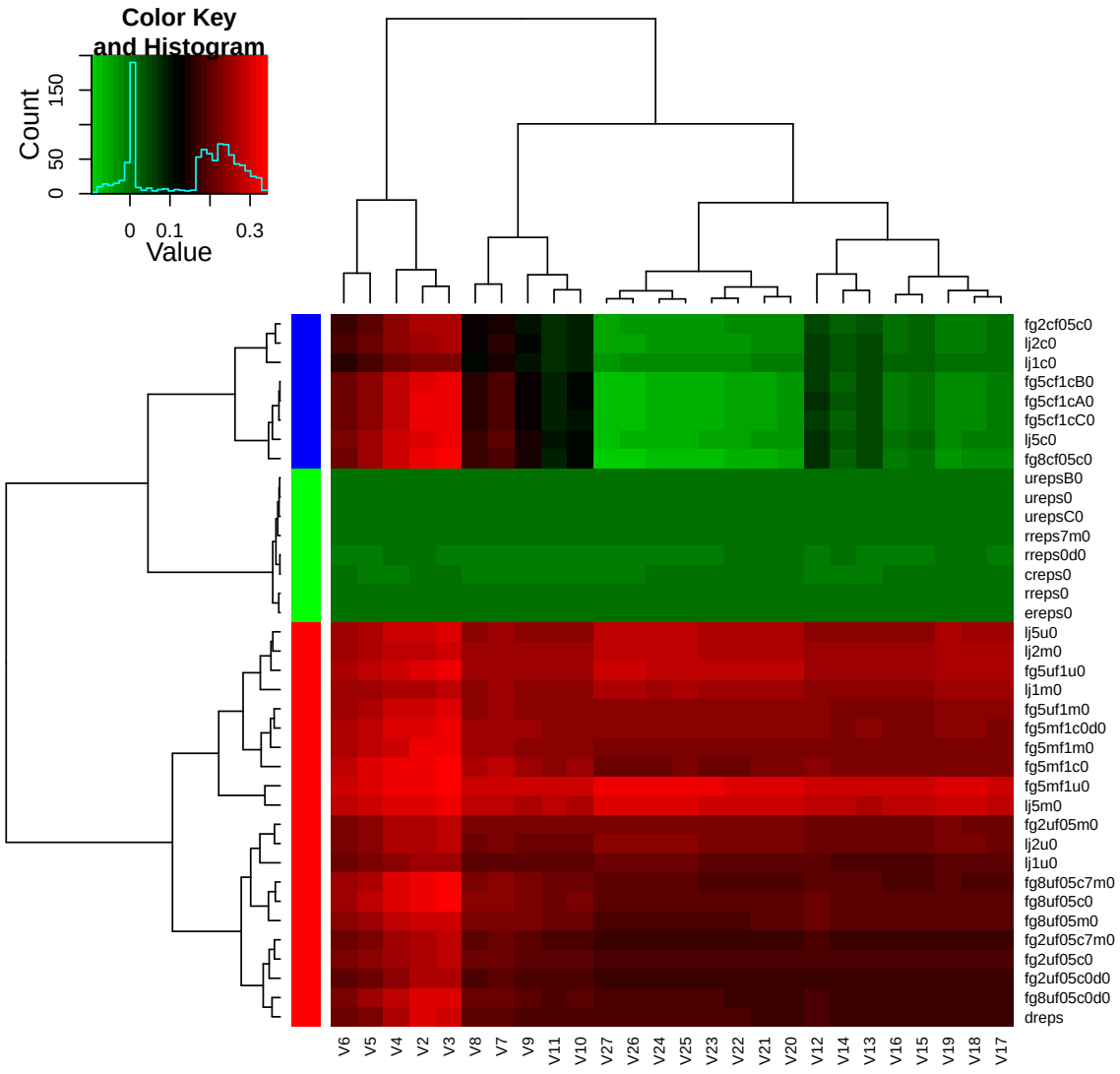

Figure 7.47: Kendall's τ correlation between social capital and skew, all simulations started from scratch.

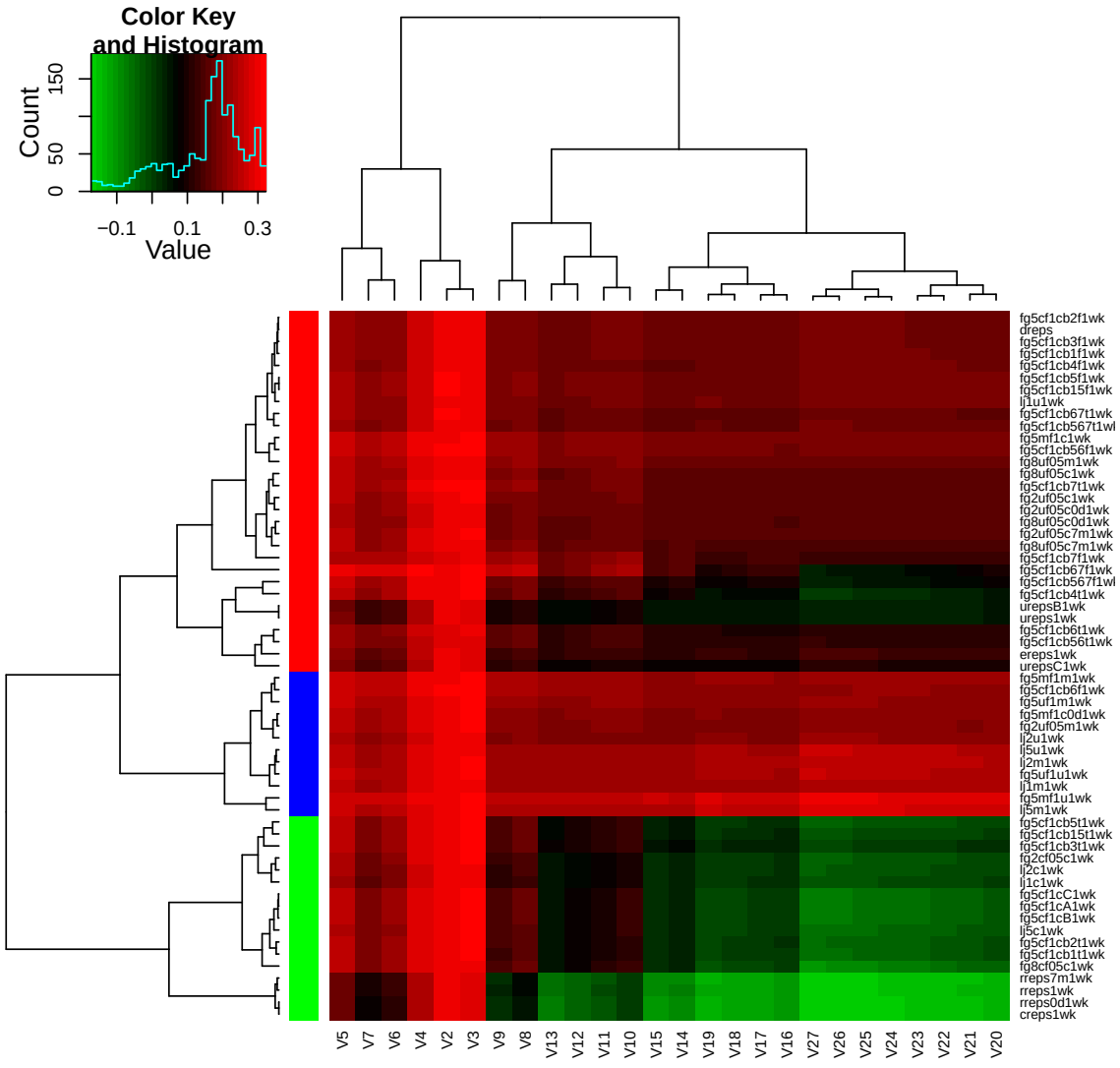

Figure 7.48: Kendall's τ correlation between social capital and skew, all simulations seeded with one week of reality.

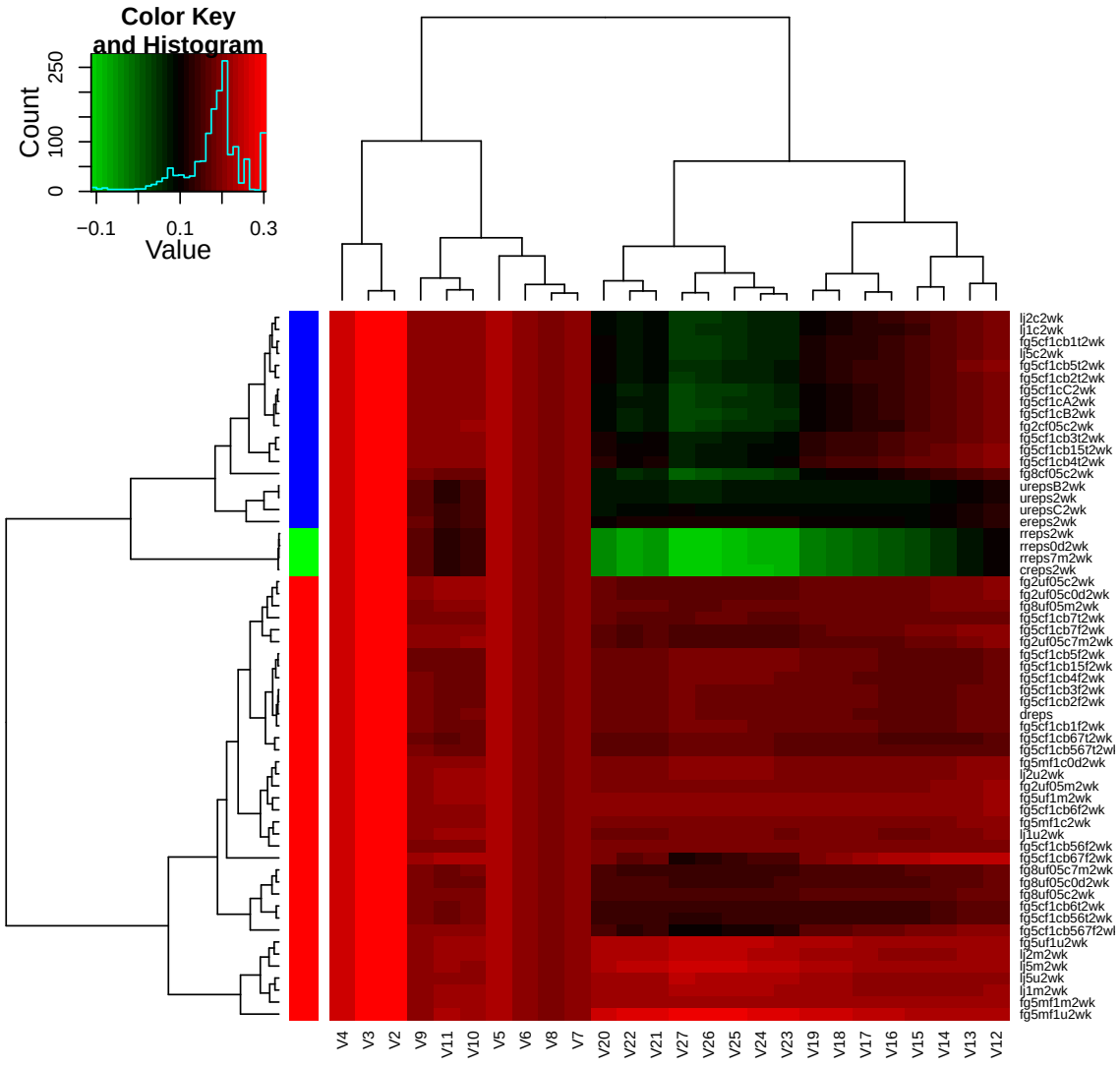

Figure 7.49: Kendall's τ correlation between social capital and skew, all simulations seeded with two weeks of reality.

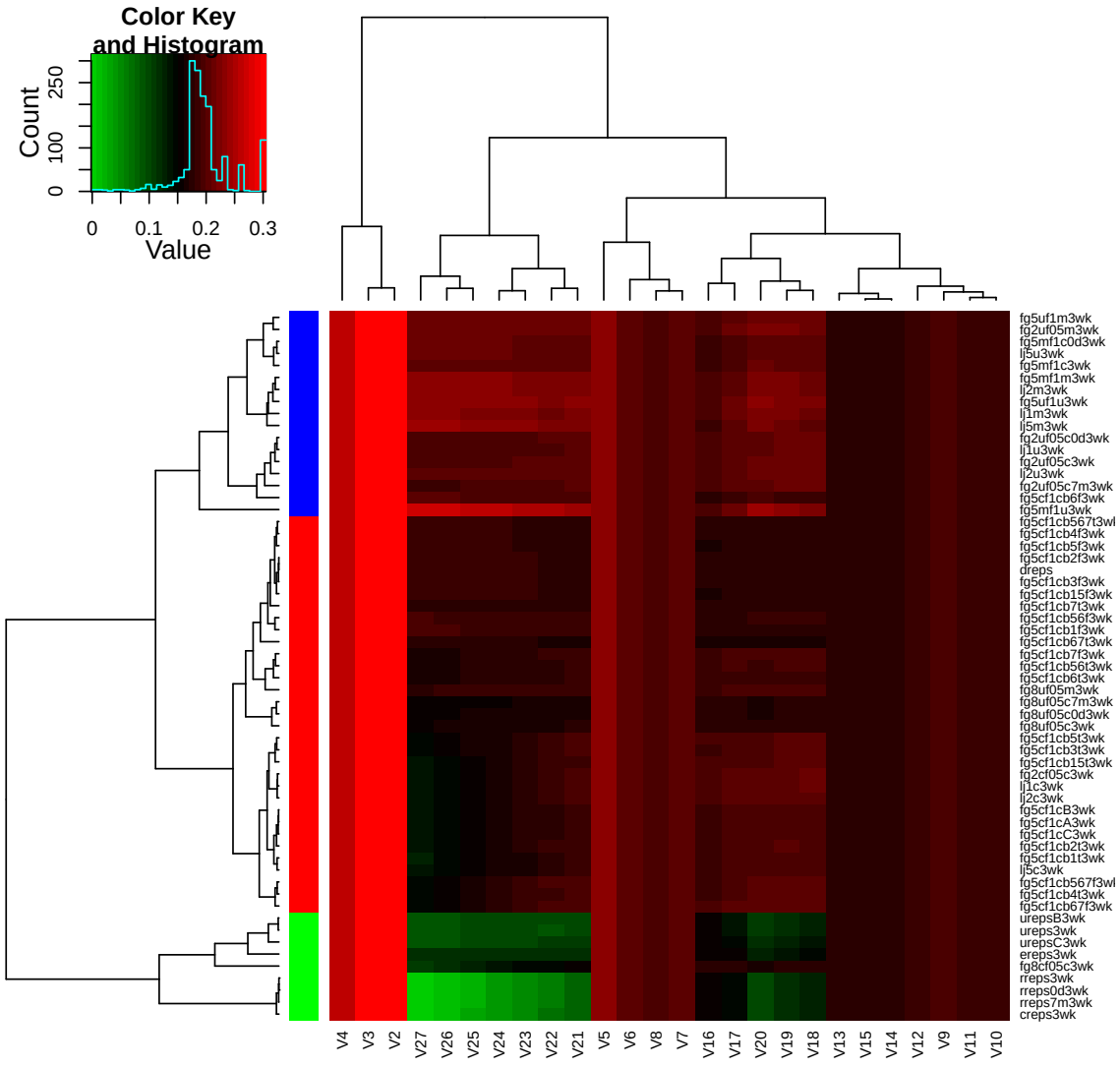

Figure 7.50: Kendall's τ correlation between social capital and skew, all simulations seeded with three weeks of reality.

Bucketed Skew Correlation

The skew-capital correlation is interesting as it can be computed on global, non-bucketed, and also bucketed scale. For the bucketed version we recompute the buckets after each simulation step and then correlate the social capital in each bucket with the resulting skew for every user in the bucket. The resulting measure is similar to the other rates except that it takes values in the range [-1,1], not [0,1] as do the rates which are bucket fractions.

The dynamics of the bucketed skew-capital correlation is presented in Figures 7.51—7.54. As in the global case, we don't see very high Kendall τ values as the typical strong correlations hover around 0.2.

When starting from scratch we get a wide range of τs, with *rreps0d* and *creps* showing strong anti-correlation in the top 10 and 1K buckets, while *fg8uf05c,7m* and *fg5mf1m,c* are the nearest neighbors. The 10K bucket, or the upper-upper middle class, is showing the best correlation for capital-based simulations. This pattern continues as we add more seeding weeks of reality.

The extremes shrink on the first week of reality seeding, when the bucketed simulations arrive, but they don't crowd *dreps* en masse right away, as the rates did. We see, as neighbors, besides the elite buckets,

- rreps0d,7m,
- fg8uf05c7m
- fg5cf1cb6f

– i.e. the capital-simulated middle class has a strong presence. The 10K and 100K buckets correlate highly for the capital-based simulations.

The second week of reality groups more bucketed simulations together with *dreps*, and our middle class *6f* is again near by. The 10K bucket still correlates well, while the 100K one correlates less so for the capital-based simulations.

Finally, with the third week we see many non-bucketed neighbors again and the upper middle class, such as

- lj5,2m
- fg5mf1c
- fg5cf1cb5f

The range shrinks more to roughly [-0.2, 0.2], and the many other bucketed simulations group separately. The poor show surprisingly homogeneous and high positive correlation while the top is mostly negative.

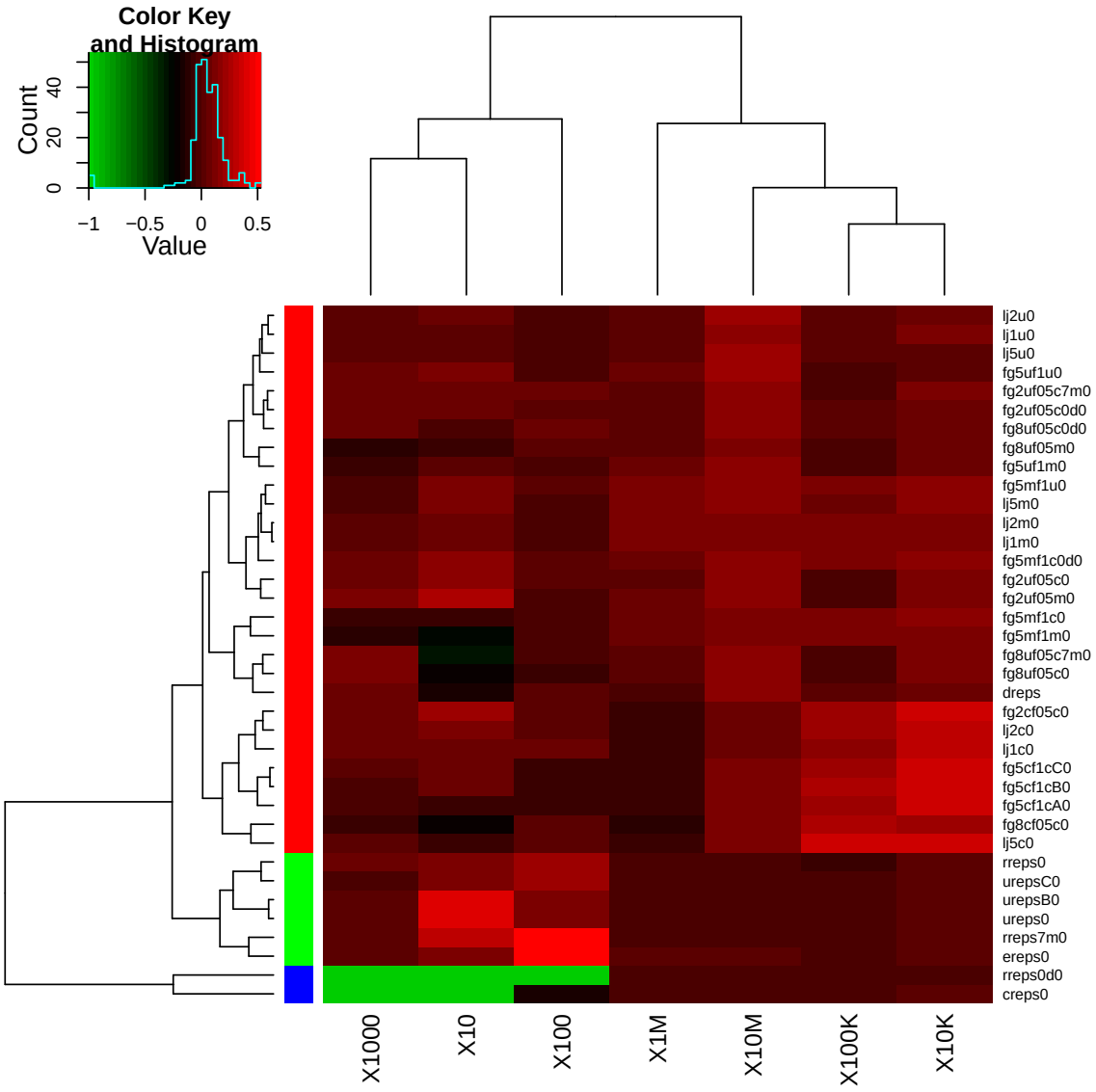

Figure 7.51: Kendall's τ correlation between social capital and skew, medians per bucket, all simulations started from scratch.

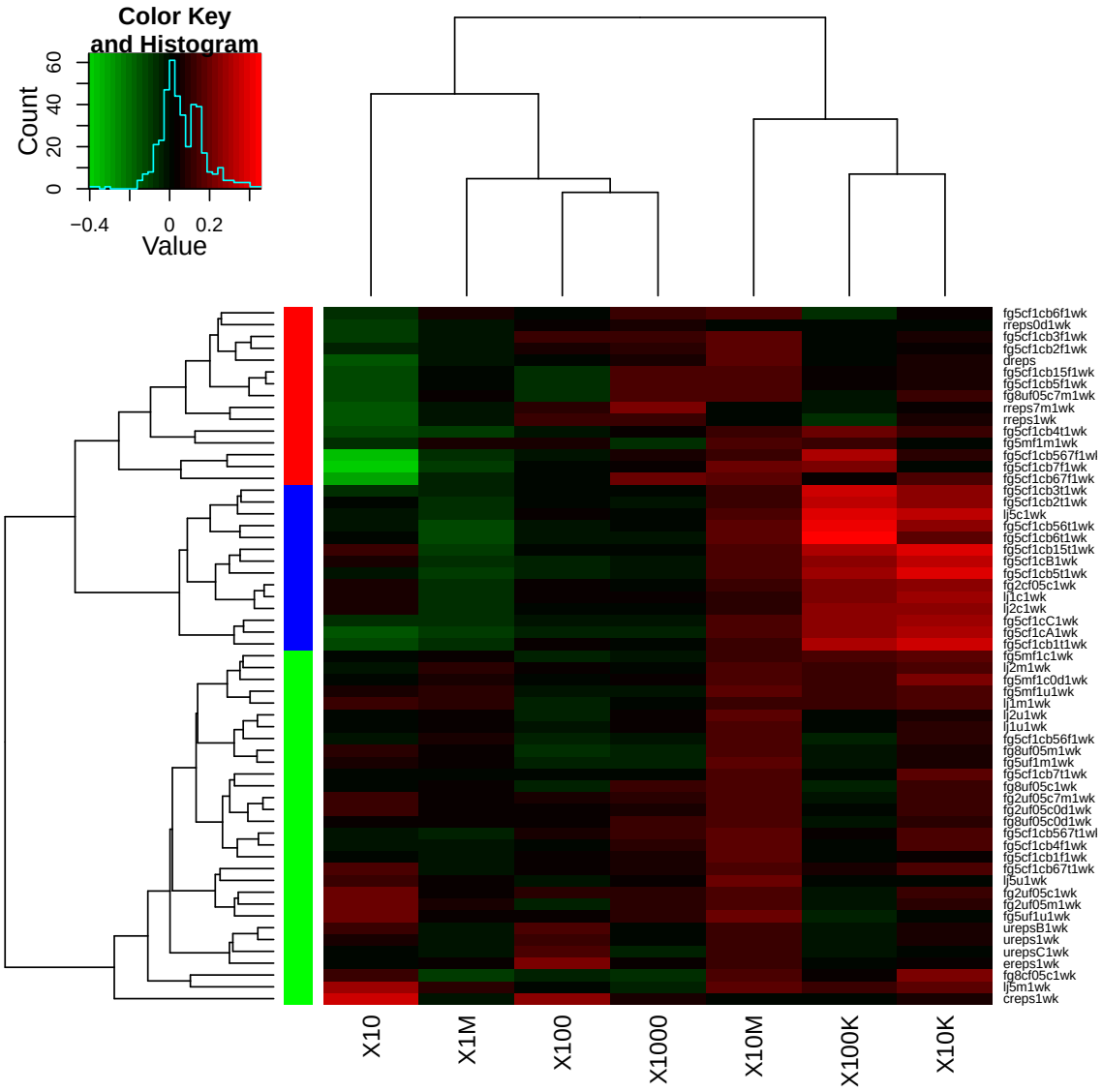

Figure 7.52: Kendall's τ correlation between social capital and skew, medians per bucket, all simulations seeded with one week of reality.

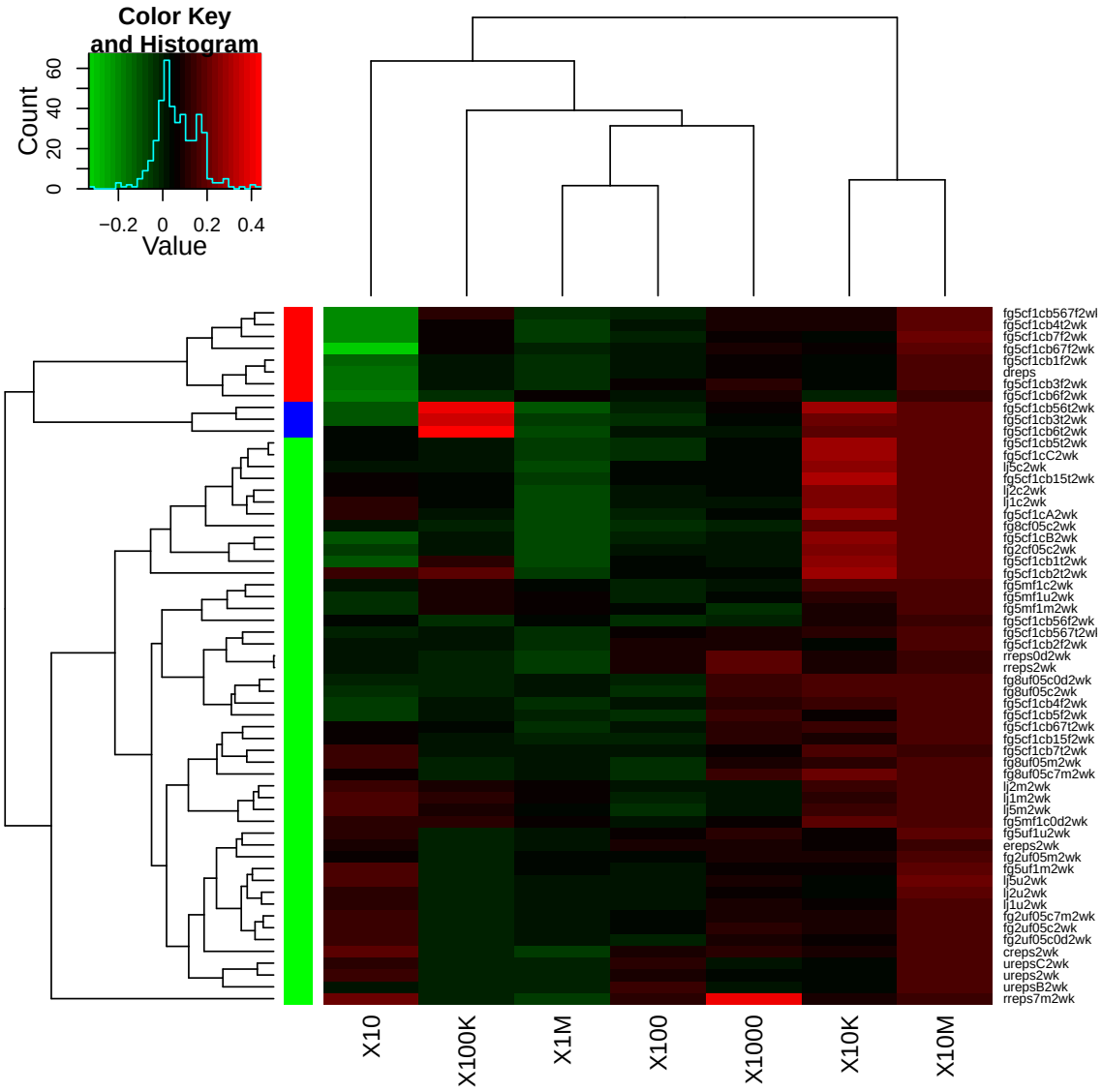

Figure 7.53: Kendall's τ correlation between social capital and skew, medians per bucket, all simulations seeded with two weeks of reality.

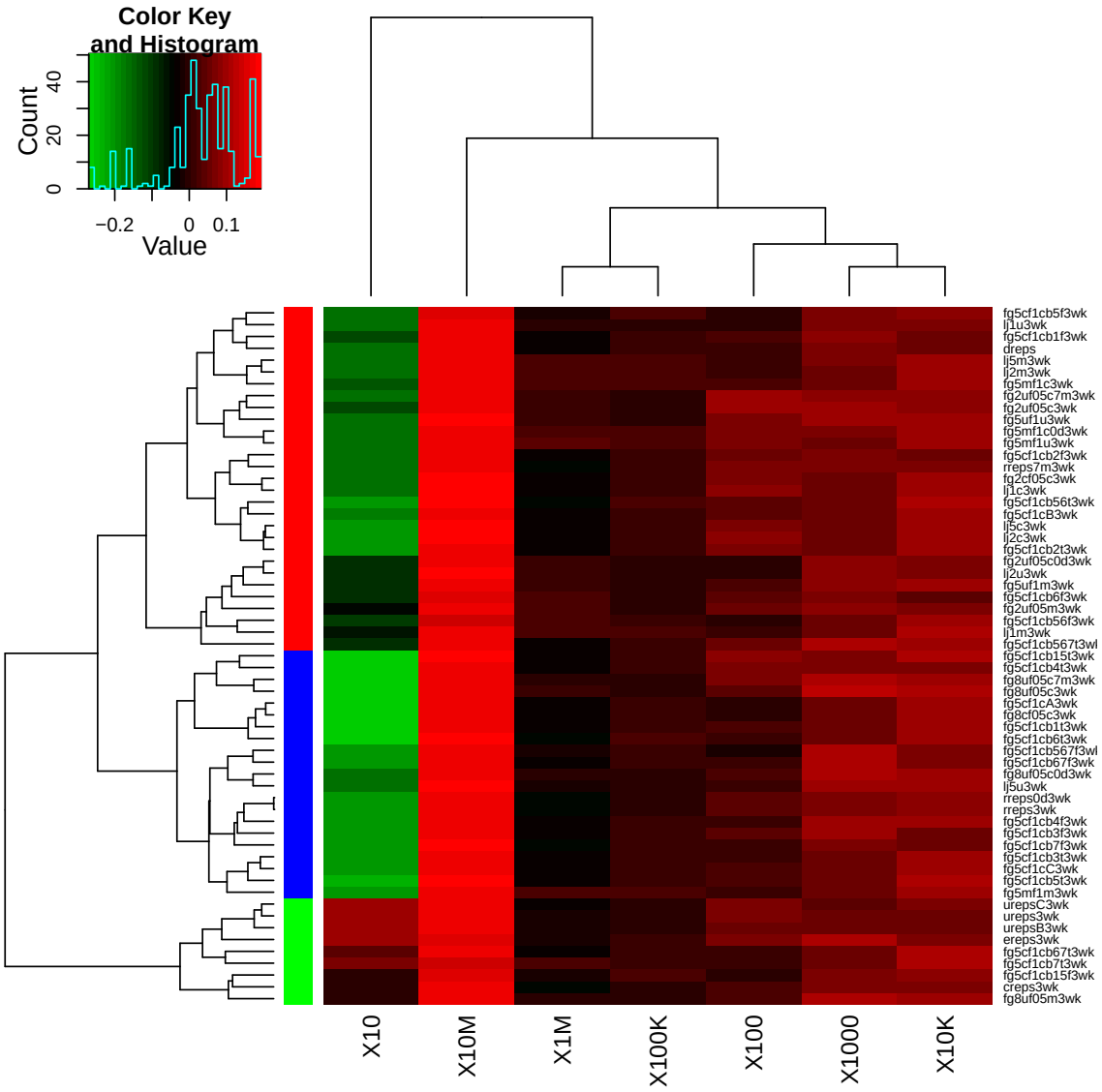

Figure 7.54: Kendall's τ correlation between social capital and skew, medians per bucket, all simulations seeded with three weeks of reality.

Significance

One question worth exploring is a seemingly circular evaluation of the class hierarchy capturing precision for the different models above. First we define the capital with our own assumptions as we see fit, rank all, and segment into classes. We then alter some links at each time step using the same formulas, compute the new social capital values, and compare the overlap between each pair of the positionally corresponding buckets. We change the topology but preserve the formula. The overall setup warrants the question of how sensitive it is to a change in the formula.

Two of the major properties of our Reciprocal Social Capital are *reciprocity* and weightedness of connections, the latter taking into account directionality (also figuring in reciprocity). We change both of these properties in several ways and run the resulting new formulæ, comparing new capital hierarchies with the original ones. A noticeable deviation would mean that our model is sensitive to the behaviors which it strives to capture.

First we invert the reciprocity rewards thus penalizing instead of rewarding those who return the tweets addressed to them. In order to highlight reciprocity or its negation effect, we may additionally reduce or eliminate the term rewarding unsolicited (non-reciprocal) mentions.

Equations 7.1—7.22 define these capital variations. A simple fully negative version S' (Eq. 7.11) subtracts all of the awards which were added originally. Version $S'\dagger$ (Eq. 7.14) drops the non-reciprocal mentions altogether, while version 7.21 reduces them so that their norm is the same as that of the reciprocal rewards, and also subtracts the reduced ones. The reason to increase the non-reciprocal norm to the magnitude of the reciprocal one is to equalize the impact of both types of the rewards for the incoming traffic, since unequalized non-reciprocal rewards dominate the reciprocal ones by orders of magnitude. While dropping nonreciprocals is one option, computed as $S'\dagger$ above, equalization of norms preserves them and their bootstrapping contribution to our iterative computation. Based on the pre-existing capital of the parties, when some amount of individual capital is seeded from non-reciprocals, it starts propagating faster via the reciprocal rewards. Table 7.3 contains the overlaps of the new social capital hierarchies with the original hierarchy. The first row shows the original overlap,

which is the baseline. Odd blocks of 5 rows (rows 1—5 and 11—15) simulate all classes together (we call them "all classes" or "non-bucketed" simulations), while even ones simulate only the middle class (bucket 10^6, or *b6*), and those are referred to as "bucketed" ones. The first half (rows 1—10) uses non-normalized unsolicited mentions while the second half (rows 11-20) normalizes them by equalizing their norm to the reciprocal mentions norm. The second column refers to the equation defining the corresponding social capital version which is also referred to in the third column by the same name as in its definition in the equation. The fourth column shows the generative simulation name abbreviating the simulation process used to generate each appropriate data set.

We observe that for negative variations, those whose simulation name starts with *n*, elites in the first four classes always fare worse than in the original version of the reciprocal social capital. A negative version dropping general mentions altogether (row 3) does just a bit better for classes 5 and 6 in the original definition, but when we drop general mentions in the original definition as well, as seen in row 2, we outperform the negative ones across the board. It shows that general mentions may obscure reciprocity while reflecting general popularity.

When we equalize the mentions and compare our original, now equalized, definition (row 12) with the equalized negative ones (rows 13—14), the negatives uniformly lose in the non-bucketed case. In the bucketed case, every equalized version, both positive and negative, does worse than the original one hence reflecting the importance of the general mentions for the middle class. In fact the middle class derives much of its capital from the unsolicited mentions of the poor as do the elites. When comparing the original equalized version in row 17 to the negatives in rows 18—19, we notice an interesting inverse effect. The elites fare better with the negatives while the middle class and the poor are still captured better by the positive version. The reason for this again is the negation and diminishing of the reward for general (unsolicited) mentions, which is key to the elites. The middle class is affected a little, but not as much as the elites, thus showing that most of its capital is still coming from the positive reciprocal components.

Our positive variations include two classes of simulations. The first, described above, drops the general mentions by zeroing out their reward in Equation 7.8. We notice that this can lead to better

capture of either elites or the poor, depending on the equalization of the general mentions. Additionally we change the formula of the weights, originally taking into account the number of daily tweets in a given direction. Thus the original composite weight from Equation 7.2 becomes an undirected one in Equation 7.23. The change in dimensionality is evened out by the corresponding change in the norms (Equations 7.3, 7.7, 7.5 and their variations). Normalization allows us to use any formulæ reflecting proportionality, where component-wise normalization means summing up relative contributions across the population.

In Table 7.3, the names of our weight-altering simulations start with letter *w*. When we compare this variation in the original formula alone we notice a slight improvement in capturing most of the classes (except the elite and the poor). This shows that the fact of a reply is more important than the frequency (weight) of that very reply edge across all classes (in a global view). When we apply this variation to the original bucketed version for the middle class we lower the social capital of the first four elite classes hence showing that when comparing to the middle class, the top classes derive a lot of their relative capital from those admirers who tweet back at them several times a day (confirmed by inspections of selected ecosystems of the top twitterers such as Justin Bieber).

The weight variations combined with mention equalizations do similarly, just slightly better than the equalized original for classes 5 and 6 when non-bucketed, and capturing classes 1-3 a bit worse when bucketed. This is the same effect as in case of equalization only just in this case reducing multiple daily tweets to single ones already performs the normalization by its own hence already dampening the scale of the effect.

In conclusion of this section we see that varying components of our reciprocal social capital definition yields confirms the original assumptions, rewarding reciprocity, directionality, and effort. Disturbing reciprocity, when adjusting the non-reciprocal mentions, does capturing precision for all classes, in both bucketed and non-bucketed simulations. Comparing non-equalized versus equalized general mentions, as well as dropping the general mentions altogether, and making weights undirected by reducing multiple daily tweets in each direction to one, leads to expected loss of capturing precision of the original hierarchy. These changes show that our terms reflect the intended behaviors appropri-

ately and allow us to further fine-tune their parameters, normalization, and reduction as needed in a specific study. Overall, we see that our original formula is appropriate for the original general study of reciprocity and hierarchy.

$$\|O_u^{@t-1}\| = \sum_{V^{t-1}} O^{@t-1} \tag{7.1}$$

$$\omega_{uv}^{@t-1} = w_{uv}^{@t-1} W_{uv}^{t-1} \tag{7.2}$$

$$O_u^{@t} = \frac{1}{\|O_u^{@t-1}\|} \sum_{v \in M_u^{t-1} \cap R_u^{@t-1} | B_{uv}^{t-1} < 0} |B_{uv}^{t-1}| \omega_{uv}^{@t-1} S_v^{t-1} \tag{7.3}$$

$$\|B_u^{@t-1}\| = \sum_{V^{t-1}} B^{@t-1} \tag{7.4}$$

$$B_u^{@t} = \frac{1}{\|B_u^{@t-1}\|} \sum_{v \in M_u^{@t-1} | B_{uv}^{t-1} > 0} B_{uv}^{t-1} \omega_{uv}^{@t-1} S_v^{t-1} \tag{7.5}$$

$$\|A_u^{@t-1}\| = \sum_{V^{t-1}} A^{@t} \tag{7.6}$$

$$A_u^{@t} = \frac{1}{\|A_u^{@t-1}\|} \sum_{v \in M_u^{@t-1}} \omega_{uv}^{@t-1} S_v^{t-1} \tag{7.7}$$

$$I_u^{@t} = \gamma B_u^{@t} + (1-\gamma) A_u^{@t} \tag{7.8}$$

$$T_u^{@t} = \beta O_u^{@t} + (1-\beta) I_u^{@t} \tag{7.9}$$

$$S_u^t = \alpha S_u^{t-1} + (1-\alpha) T_u^{@t} \tag{7.10}$$

$${S'}_u^t = \alpha {S'}_u^{t-1} - (1-\alpha) T_u^{@t} \tag{7.11}$$

$$I\dagger_u^{@t} = B_u^{@t} \tag{7.12}$$

$$T\dagger_u^{@t} = \beta O_u^{@t} + (1-\beta) I\,\dagger_u^{@t} \tag{7.13}$$

$$S\dagger_u^t = \alpha S\,\dagger_u^{t-1} + (1-\alpha) T\,\dagger_u^{@t} \tag{7.14}$$

$$S'\dagger_u^t = \alpha S'\,\dagger_u^{t-1} - (1-\alpha) T\,\dagger_u^{@t} \tag{7.15}$$

$$\|\bar{A}_u^{@t-1}\| = \|A_u^{@t-1}\| \frac{\|A_u^{@t-1}\|}{\|B_u^{@t-1}\|} \tag{7.16}$$

$$\bar{A}_u^{@t} = \frac{1}{\|\bar{A}_u^{@t-1}\|} \sum_{v \in M_u^{@t-1}} \omega_{uv}^{@t-1} S_v^{t-1} \tag{7.17}$$

$$\bar{I}_u^{@t} = \gamma B_u^{@t} + (1-\gamma) \bar{A}_u^{@t} \tag{7.18}$$

$$\bar{T}_u^{@t} = \beta O_u^{@t} + (1-\beta) \bar{I}_u^{@t} \tag{7.19}$$

$$\bar{S}_u^t = \alpha \bar{S}_u^{t-1} + (1-\alpha) \bar{T}_u^{@t} \tag{7.20}$$

$${\bar{S}'}_u^t = \alpha {\bar{S}'}_u^{t-1} - (1-\alpha) {\bar{T}'}_u^{@t} \tag{7.21}$$

$$S'\dagger_u^t = \alpha S'\,\dagger_u^{t-1} - (1-\alpha) T'\,\dagger_u^{@t} \tag{7.22}$$

$$\omega^{*^{@t-1}}_{uv} = W^{t-1}_{uv} \tag{7.23}$$

$$S^{*^{t}}_{u} = \alpha S^{*^{t-1}}_{u} - (1-\alpha) T^{*^{@t}}_{u} \tag{7.24}$$

$$\bar{S^{*}}^{t}_{u} = \alpha \bar{S^{*}}^{t-1}_{u} - (1-\alpha) T^{*^{@t}}_{u} \tag{7.25}$$

As we show in the definition of $T\dagger$ (7.13), it has an awesome dagger. On the other hand, 7.1 shows a strong norm.

Table 7.3: Social Capital Variations. Negative versions underperform, while normalized mentions do a but better. Sensitivity to these terms confirms that the original formula reflects the behaviors in an expected way.

#	Eq.	Ver.	Sim. Name	10	100	1,000	10K	100K	1M	10M
1	7.10	S	fg5cf1cA2wk	0.10	0.01	0.03	0.06	0.06	0.71	0.91
2	7.14	$S\dagger$	fig5cf1c2wk	0.10	0.03	0.03	0.07	0.09	0.78	0.92
3	7.22	$S'\dagger$	n2fg5cf1c2wk	0.00	0.00	0.01	0.02	0.07	0.74	0.91
4	7.11	S'	n3fg5cf1c2wk	0.00	0.00	0.00	0.00	0.05	0.67	0.90
5	7.24	S^*	w1fg5cf1c2wk	0.00	0.02	0.03	0.06	0.08	0.78	0.91
6	7.10	S	fg5cf1cb6f2wk	0.70	0.64	0.81	0.84	0.83	0.93	0.98
7	7.14	$S\dagger$	fig5cf1cb6f2wk	0.40	0.15	0.20	0.25	0.13	0.78	0.94
8	7.22	$S'\dagger$	n2fg5cf1cb6f2wk	0.50	0.39	0.48	0.19	0.06	0.77	0.92
9	7.11	S'	n3fg5cf1cb6f2wk	0.20	0.49	0.64	0.79	0.93	0.96	0.99
10	7.24	S^*	w1fg5cf1cb6f2wk	0.30	0.38	0.65	0.70	0.75	0.93	0.97
11	7.10	S	fg5cf1cA2wk	0.10	0.01	0.03	0.06	0.06	0.71	0.91
12	7.20	$\bar{S}$	adNfg5cf1c2wk	0.00	0.03	0.03	0.07	0.08	0.75	0.91
13	7.21	$\bar{S}'$	n1Nfg5cf1c2wk	0.00	0.00	0.01	0.02	0.08	0.71	0.90
14	7.22	$S'\dagger$	n2fg5cf1c2wk	0.00	0.00	0.01	0.02	0.07	0.74	0.91
15	7.25	$\bar{S}^*$	w0Nfg5cf1c2wk	0.00	0.03	0.03	0.06	0.08	0.75	0.91
16	7.10	S	fg5cf1cb6f2wk	0.70	0.64	0.81	0.84	0.83	0.93	0.98
17	7.20	$\bar{S}$	adNfg5cf1cb6f2wk	0.30	0.17	0.21	0.19	0.23	0.85	0.96
18	7.21	$\bar{S}'$	n1Nfg5cf1cb6f2wk	0.50	0.41	0.39	0.47	0.28	0.84	0.96
19	7.22	$S'\dagger$	n2fg5cf1cb6f2wk	0.50	0.39	0.48	0.19	0.06	0.77	0.92
20	7.25	$\bar{S}^*$	w0Nfg5cf1cb6f2wk	0.00	0.09	0.13	0.19	0.20	0.83	0.96

Discussion

A question to consider is, are the influentials accidental or not after all?

The staying rate within the same simulation shows how stable the winners are within their own classes. If the churn in buckets is not too great, especially among the higher classes, we have our influentials who keep their positions for long periods of time, as is evidenced in the real world.

We see with the staying rates in simulations is that the more elaborate a simulation is the higher the staying rates are. This demonstrates that the winners in a given world are not random and they maintain their ranking from day to day. Global uniform attachment doesn't achieve any staying power for classes below celebrity level, and any combination with uniform attachment is inferior in terms of capturing precision to those with mentions- or capital-based ones. If one compares the staying power of *fg2uf05m*, *fg2uf05c*, and *fg2cf05c* or *fg5cf1c* in Figure 7.11, the capital-based global attachment significantly increases stability of the top buckets.

The uniform simulations, *ureps*, illustrate the nature of this dilemma. These simuations have almost no staying rate in any higher-class buckets except for the very top class, where the celebrities are surprisingly persistent, with a staying rate of 60-70%, see Figures 7.10–7.11. However, a comparison between different runs of *ureps*, – *ureps*, *urepsA*, and *urepsB* groups, – shows that there is almost no overlap among the celebrities across the groups. In the uniform case almost everybody is in the poor class, with meaningless connections that do not reward them with any social capital. However, even in such a setting a capital-poor user will get lucky merely by chance and return a reply thus achieving a reward of higher capital. They will be likely to stay there especially if their activity level was high.

The capital-based attachments achieve a staying rate which is in some cases even higher than the real ones. The uniform-based FOF and global simulations rank lower than the mention-based ones, and indeed, mentions are the key component of social capital. The increase of the staying rates with increase in the intelligence of the behaviors used to attain them confirms that "doing the right thing" keeps the winners on top thus proving they are not accidental under given conditions.

The overlap among the simulated winners and the real ones, while minimal, is not random. It

gets closer to 10% per bucket of the upper middle classes for some of the elaborate, utility/FOF-capitalmention global-capital/mention simulations, such as *fg5mf1m,c,c0d* in Figure 7.4. Among the non-bucketed simulations, *lj1c* does quite well, capturing some of the 10K class winners. This means that in a world with slightly different rules, other people will achieve higher social capital. If everybody attaches according to slightly different rules, those who were in a good starting position (seeded with *dreps*, show up on the same day), and are active enough (outdegree preserved), will fit better.

Among the same type of simulation generated with different initializations of the random number generator, we get much higher overlap than with any other simulation. This means that the classes we define are mostly held for the same type of strategy, such that the combination of the reciprocal social capital metric and class segmentation depends mostly on the behaviors themselves.

Thus, our conclusion is that the influentials, defined with all the assumptions and ramifications of the underlying metric and class bucketing, are not accidental in their own worlds. The more intelligently they are simulated, the better they stay on top, and grab a better share of the real-world winners in overlaps. However, the top celebrities can be persistent even in a version of a stochastic world, and a slight change of rules will lead to a new and distinct persistent hierarchy which stays around under the same conditions even though all of the edges are rearranged. Seeding with the starting conditions from the real world doesn't lead to the real world hierarchy to continue except for a few days immediately after the hand-off. However the new hierarchy is stable enough in smart simulations, yielding upper classes with relatively low churn.

The fact that simulating elites does not change the hierarchy as much as altering the upper middle class and on down tells us the elites do not define the communication structure as much as the upper middle, the middle, and the poor classes.

Simulating the middle class often does not lead to an upset in the hierarchy because ours is in fact a fairly faithful simulation. When fitting a linear model against all the buckets, determining the euclidian distance from reality, 0 for reality itself, we get buckets 4, 5 and 7. If we keep these buckets intact, we stay closer to reality. The top 3 elite buckets do not figure; the poor 7 dominates by its sheer bulk, while the upper middle class and the bucket preceding it probably use strategies quite different from

the capital-capital strategy we used.

We plotted distance of a function of preserving or simulating several bottom classes in Figures 7.8, 7.8, 7.8, 7.8. All but the middle class (bucket 6) figure as significant in a linear model on all features. While preserving most classes brings the distance lower, the middle class is simulated well enough to make its preservation not significant distance-wise.

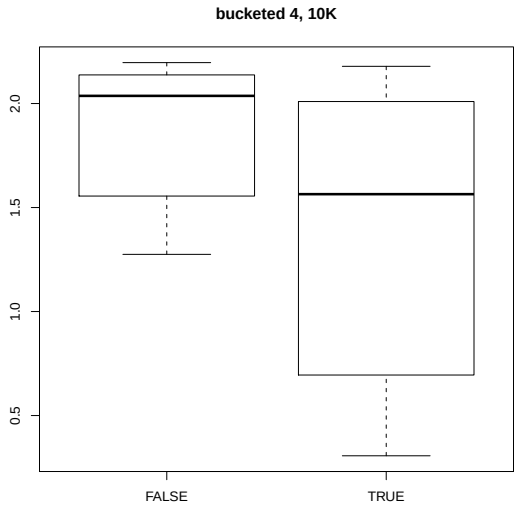

Figure 7.55: distance by the top 10K

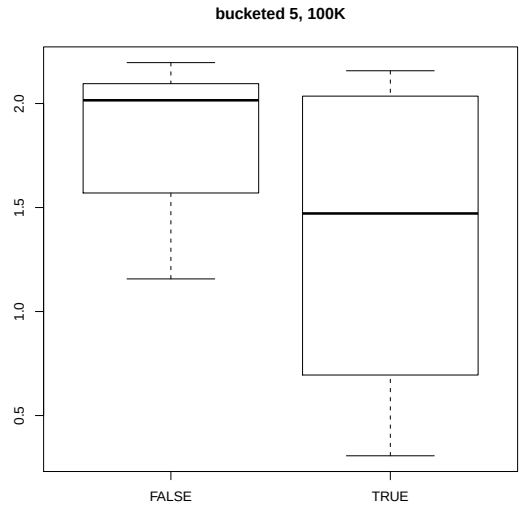

Figure 7.56: distance by the upper middle class

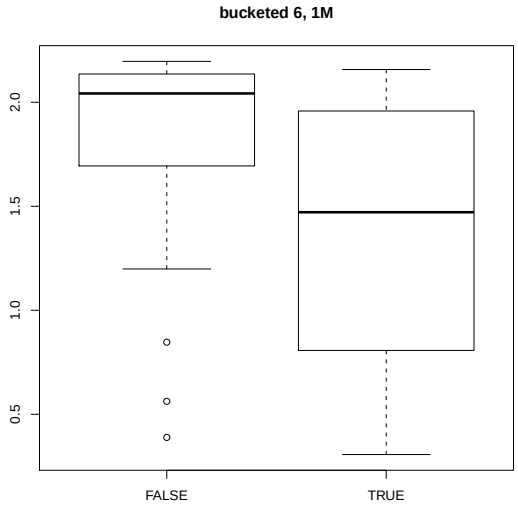

Figure 7.57: distance by middle class

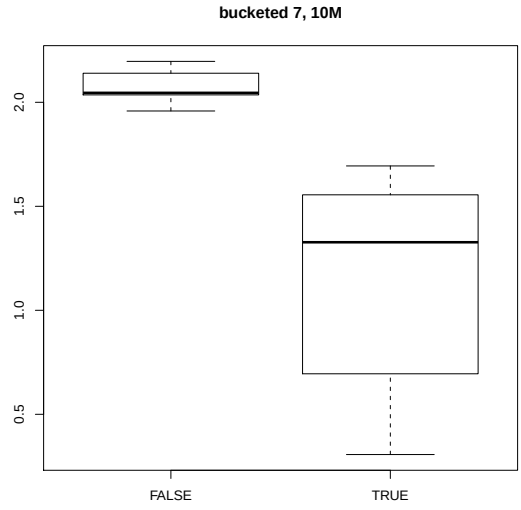

Figure 7.58: distance by the poor

Figure 7.59 shows a decision tree, fitting the distance for the overlap with reality from the ideal of 0. We use Euclidian distance for a 7-component vector corresponding to the class buckets. We intentionally treat each bucket equally in this distance throughout our metrics since preserving the actual hierarchy is important for the role of the elites while any weighting by the bucket size would overwhelm them.

We see that the poor dominate by their volume, but then capital strategies work better simpler strategies, as does utility-based attachment.

Bucket 4, the top 10K users by social capital after the preceding 1110, show the highest Kendall τ correlation with the skew measure of politician-like behavior we devised. This means the members of class are the most attentive to the difference in input between their highest and lowest contributors, and most consistent in repaying those inputs proportionally. The 10K class (bucket 4), together with the upper middle class (bucket 5), also figure as significant in linear models and decision trees regressing the overlap with reality on preserving or simulating individual buckets.

The most important finding in our work is a persistent middle class which is both defined and captured quite well by our reciprocal social capital. As defined, the 1 M bucket above the poor, in the hierarchy segmented by that capital, the middle class is responsible for about 40% of all outgoing replies overall (Figure 7.12). It clusters near reality in most of the metrics, and notably overlaps with it, when replacing "just" the whole middle class with its simulation via *fg5cf1c* simulation – attaching to FOFs and globally proportionally to social capital in the previous cycle, and using local utility optimization against the same capital.

The fact that the middle class behaves consistently near reality in both true and capital-simulated form in most of our evaluations indicates that

- the middle class is working hard to keep the conversation going
- it is trying to operate in a balanced, equitable manner
- it has a high energy to produce a big share of all discourse

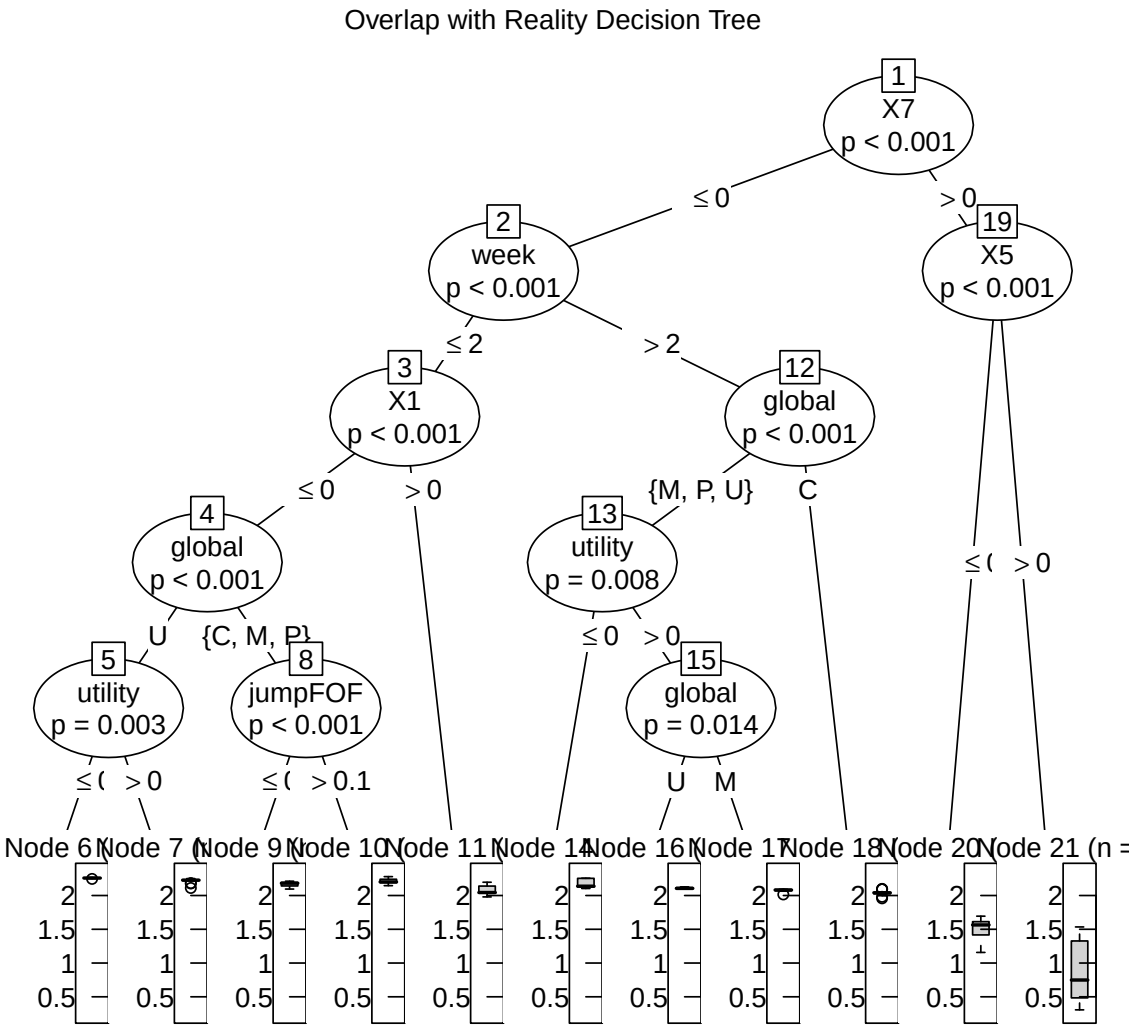

Figure 7.59: Decision tree on all features, fitting distance from reality.

- it is in no way random regardless of the elites
- it is responsible for many of the effects we care about in our measures
- and hence, is the real influencers we should care about, not the elites where randomness may depend on a point of view

Chapter 8

Data Mining Infrastructure

8.1 Importance of Social Programs

Any massive data mining of social networks has to overcome scaling problems and provide engineering solutions to the challenges which such large amounts of data pose. We do not consider algorithmic part of the effort separate from programming. On the contrary, we firmly believe that the way in which the ideas are formulated in computer science is inseparable from their effective implementation, a process which includes the lifecycle of a program as a materialized idea. Efficiency includes not just the performance of a system but also the ease of its coding, maintenance, understanding, and thus spreading and evolution.

Moreover, the modern social media community is driven by a dynamic class of cutting-edge hyperconnected coders who co-evolve the networking platforms and software environments enabling their programming, coding, scaling, and presentation. This co-evolution started with PHP, skyrocketed with Ruby on Rails, and continues with modern *functional* languages, exceedingly via open-sourced, social coding. In fact, the rise of the first social coding platform, github[1], accelerated this process. Many day-to-day discussions among those building the social networks of today and tomorrow happen on Twitter. Following modern techniques, languages, and tools is key to advancing this revolution. Good

[1] http://github.com

examples of the process are the Scala and Clojure programming languages, pushing the envelope on abstraction, software engineering philosophy, new web paradigms, and social coding.

In the spirit of the community, we make all of the code developed throughout this research open source and available on github under the Creative Commons license.

8.2 Streaming API

Gathering and storing Twitter data efficiently is significant to the success of the job of analyzing it. We investigated several setups before arriving at our current one. Previously, people used to crawl Twitter, calling the query API provided to get individual users' tweets. The API has a rate limit for queries but researchers can have their machines "whitelisted," a process which actually just raises the limit for queries quite high. Twitter admins don't like anybody getting a hold of any significant fraction of all tweets, so any attempt to crawl aggressively often results in a permanent ban. Since many people have a use for tweet streams, Twitter set up a separate Streaming API. Instead of pulling information with SOAP queries from the web service, Streaming API pushes a constant stream of tweets over an open HTML channel which stays open. There are several levels of streaming.

- The *firehose* is the full set of all tweets, pushed out in near real time. *Firehose* is provided on a commercial basis to big players such as Google and Microsoft.
- The next level is *gardenhose*, which is, in Twitter's words, "statistically representative" of the firehose. When we started gathering the gardenhose in mid-2009, we collected about 2 million tweets a day; a year later, we were getting more than 5 million tweets daily through it.
- Another stream is called *shadow*, and allows data gatherers to follow a limited set of users fully. Our limit of shadowing is the maximum that research access allows, up to 50,000 people.

The *gardenhose* seems to be a round-robin sample of the latest tweets, and it does not provide a continuous coverage of any particular individual. Nevertheless, we found that those who tweet a lot are represented proportionally in the gardenhose. When we started our studies we computed the top pairs of repliers chosen as follows. For the replier graph, we looked at all pairs of users engaged in dialogues and having the most exchanges. We then sorted them by the descending strength of the dialogues (computed as a function of the daily and total volume). We reviewed the top 50,000 of this ordered list. (It was interesting to see who was on top – several couples, separated by distance, and also users engaged in explicit talk with several partners simultaneously. Apparently they used Twitter as a sort of SMS

service without realizing it's publicly available; most of them had closed their tweets from the public a few weeks into the study.)

8.3 Storage for Analysis

Storing billions of tweets poses many significant challenges. The data comprises both a text corpus and a social graph and both aspects need to be accessible and cross-linked. We tested several types of storage and ended up with several versions of the data which was suitable for better access and specific purposes. For the text part, we employed Lucene[2], the state of the art information retrieval engine. The original Streaming API[3] is pushed as either XML or JSON with the latter being the most compact and efficient representation of heterogeneous data. MongoDB[4] allows for direct storage and indexing of JSON[5], and goes well with dynamic languages such as Clojure. Clojure's Mongo interface, congomongo[6] can open Mongo collections as lazy sequences to be consumed on a when-needed basis. We also use Berkeley DB JE[7] for parallel serialization and Tokyo Cabinet[8] with Google Protocol Buffers[9] as values for very fast adjacency list representation of the graph. The dynamic aspect of our graph has to be taken into account every step of the way. We store exact time stamps of all tweets in the comprehensive representation and discretize to the daily snapshots in our evolutionary modeling with reciprocal social capital.

[2]http://lucene.apache.org/

[3]http://dev.twitter.com/pages/streaming_api

[4]http://mongodb.org/

[5]http://json.org/

[6]http://github.com/somnium/congomongo

[7]http://www.oracle.com/technetwork/database/berkeleydb/overview/index.html

[8]http://www.1978th.net/tokyocabinet/

[9]http://code.google.com/apis/protocolbuffers/

8.4 Platform

Our social network exploration platform is developed on the Java Virtual Machine (JVM) technology. For storing the vast amounts of data coming daily, we tap into fast and robust databases:

- the Berkeley DB Java Edition (BDB JE) [59], an in-process persistent hash,
- MongoDB [1], a JSON-based client/server with in-memory cache and excellent Java driver.

We used two modern JVM languages, *Scala* [54] and *Clojure* [33], for efficient data mining. Scala is a modern object-functional language allowing for compact and expressive code while fully interoperable with the Java platform. We previously have successfully developed data mining software with the Functional Programming (FP) paradigm, and Scala FP blends with superior object orientation and Java interoperability. Scala has with Erlang-style agents for concurrency, convenient for both local and distributed parallelization. Clojure, a next-generation Lisp on the JVM, is fully interoperable with Java and provides state of the art concurrency mechanisms for shared memory architectures, thus letting us exploit our 64 GB RAM, 8 core server with significant speedup. The dynamic nature of Clojure makes it especially easy to use in data mining applications when data types are task-specific. Clojure syntax has maps, e.g. {`:a 1 :b 2`}, and vectors, `[1 2 3]`, as first-class citizens, making it easy to explore data in the REPL which is essential in our EDA-based data mining.

Several excellent projects under active development in Scala, Clojure, and Java communities help with machine learning and data mining, while others from Java can be fully utilized for NLP jobs:

- Incanter[10] – an R-like data exploratory statistical environment in Clojure
- Infer[11] – machine learning algorithms from the founders of Flightcaster and Aria Haghighi
- Clojuratica[12] – a two-way Clojure interface with Mathematica

[10] http://incanter.org

[11] http://github.com/bradford/infer

[12] http://clojuratica.weebly.com/

- OpenNLP[13] – a mature Java NLP library from UPenn's own Tom Morton and others
- Lingpipe[14] – a mature Java NLP library (also related to UPenn)
- ScalaNLP[15] – a collection of libraries for Natural Language Processing, Machine Learning, and Statistics in Scala, based on an efficient new Linear Algebra library Scala.la

Our communication graph is represented as a series of adjacency lists storing repliers for each source node. An alternative to this approach would be to store triplets `<source,target,weight>`, where weight is any object decorating the edge. When using SQL databases the triplets are the only obvious option, but since we are using either Berkeley DB Java Edition, or Tokyo Cabinet with Protobuffers, which are capable of storing a variety of JVM objects, we store the adjacency lists directly as hash maps. This approach makes it significantly faster to get all the repliers for a particular user. We use Java/Scala/Clojure interoperability to store the graph as either Java hash maps or Google Protobuffers, thus allowing them to be retrieved back from any JVM language. This experience led to optimizing Berkeley DB JE for Twitter storage by the first author with the Oracle team directly (as reported on the Oracle Java blog [48]), and the ongoing work with clojure-protobuf[16], jiraph[17] and cake[18] projects.

For the n-gram models and *SIP*s, we used *LingPipe* [12], an NLP library written in Java. For text search, we use Lucene, the premier full text search engine originally and currently developed in Java. The n-gram models are well-studied in NLP and their implementation must be compact, fast, and serializable quickly. LingPipe stores both a binary form for efficient perplexity computation and Google n-gram format for full count access.

We relied on the JVM platform and one of its functional programming languages, *Clojure*, a dynamic Lisp [33], and took advantage of the scalability and parallelism it offers. We received Twitter data via its

[13] http://opennlp.sourceforge.net/

[14] http://alias-i.com/lingpipe/

[15] http://www.scalanlp.org/

[16] http://http://github.com/ninjudd/clojure-protobuf

[17] http://github.com/ninjudd/jiraph

[18] http://github.com/ninjudd/cake

Streaming API as JSON and stored it in *MongoDB* [1], a modern *NoSQL* document database. The data mining was performed in Clojure, interfacing with MongoDB via *congomongo* [7]. Any intermediate results were stored right back as nested maps of maps or vectors in congomongo, thus allowing for transparent serialization, persistence, and indexing via MongoDB.

A graph with decorated edges is represented as a Clojure map, which, along with a vector, are first-class constructs – e.g. a graph of mentioners from the Figure 5.2 looks like

```
{:s {:a [8 1] :b [6 3] :c [4 4] :d [10 1] :e [4 2]}}
```

Such maps are subject to Clojure destructuring and functional map/reduce/filter transforms, and are expressive and concise. The code for our project is open source and available on github [42]. Visualization and model fitting is smoothly handled by Incanter, an R-like statistical environment in Clojure [50].

8.5 Accidental Influentials

The simulations required to test the "Accidental Influentials" hypothesis required modeling a large network in a very efficient way. Local utility optimization and friend of friends attachment demanded time and memory efficient representations. We decided to use the OCaml implementation as it is the fastest to serialize and deserialize vast graph data, and on par with Haskell in computation times, while using a bit less memory – 40GB vs 50GB RAM on basic simulations. With the composite strategies we hover on the edge of the 64GB RAM we have and the execution time extends from 30 minutes to 3 hours, still making simulating full bucket sets and a reasonable number of bucket combinations feasible.

Another challenge of applying dozens of metrics to hundreds of simulations is the set of results. Our table output can reach up to 40,000 files going into a finished TEX product, data inputs, and graphs. Many of the analyses have to be performed in sequence and the intermediate steps may need to be compressed which requires gigabytes of storage. We implemented a GNU Make-based pipeline, defined as 2,000 lines of modular makefiles, that govern the whole process in a functional way. Based on a target the pipeline assembles and produces every necessary step, depositing results where required. The pipeline factors out most of the reusable steps making it easy to add new metrics.

Our methodology requires dynamic analysis of correspondence between buckets across days and simulations. We output thousands of tables, 4 per page, in a format convenient to review. Medians and averages for each bucket class are output as a single row for each table and assembled into summary tables. These tables can, in turn, be either viewed as readable TEX, or output as text for R consumption. A part of the pipeline automatically formats the high-level results as an R dataframe for further statistical analysis. The OCaml system and Makefile code, as well as R scripts used to produce the graphs, are open-sourced at

```
http://github.com/alexy/katz
```

8.6 Open Source Contributions

The code for this paper is open source and available on github as a series of projects.

- LifeLang[19] – Language of Life identification *(OCaml)*
- Tfitter[20] [41] – data storage in BDB JE/PostgreSQL, topic indexing, community detection *(Scala)*
- Mongol[21] – JSON[22] processing with MongoDB[23], dynamic graph analysis *(Clojure)*
- Badjer[24] – Reciprocal Social Capital in Clojure
- Clams[25] – Reciprocal Social Capital in OCaml
- Husky[26] – Reciprocal Social Capital in Haskell
- Katz[27] – Non-accidental Influentials in OCaml

Computing Reciprocal Social Capital resulted in a very computationally demanding process. We iterated through all of the original Twitter interactions and applied the update rules to the capital values in order to advance them day by day for all talking users. For the working set this meant replaying 100 million tweets over 35 days, one full cycle per day. Much effort went into proper storage and retrieval of tweets as serializing and deserializing the graphs may take hours. We wrote OCaml and Haskell implementations of the Reciprocal Social Capital, originally developed in Clojure, to verify the complex iterative computation while enabling fast loading and storing of the data and results, and to bring the computation and result lookup up to natively compiled speeds. The "Accidental Influentials" problem was solved in OCaml, at an even larger scale. With Twitter's permission we prepared

[19] http://github.com/alexy/life-language

[20] http://github.com/alexy/tfitter

[21] http://github.com/alexy/mongol

[22] http://www.json.org/

[23] http://www.mongodb.org/

[24] http://github.com/alexy/badjer

[25] http://github.com/alexy/clams

[26] http://github.com/alexy/husky

[27] http://github.com/alexy/clams

our communication graph in a portable format and documented our Reciprocal Social Capital computation thoroughly to create a benchmark FunData1[28]. We hoped that by creating an open-source social networking shootout for functional programming languages will advance progress of the languages and the algorithms, including better data structures.

Already *FunData1* benchmark has led to the biggest RAM-consuming implementation in Haskell ever which allowed Microsoft Research, Cambridge, UK to discover an integer overflow bug in Haskell runtime's garbage collector. We thank Simon Marlow of Haskell core team for finding the bug, with the aid of the largest Amazon Elastic Cloud image available at the moment (with 68 GB of RAM, the GHC group in MSR Cambridge themselves lacking a big enough box then).

FunData1 was presented at the first Clojure Conj[29] conference and has caused significant interest already including thanks from Google researchers who want to use Haskell for similar tasks. We hope it will lead to improving the infrastructure for processing large volumes of social networking data in all of the languages involved, as it did in Haskell.

[28] http://fundata1.functional.tv

[29] http://clojure-conj.org/

Chapter 9

Conclusions

In this thesis we considered various components which animate modern online social networks so that the users keep talking through the media. Inevitably, some users talk more, and/or use the social networks more effectively than others. We look at effective communication as a foundation of influence and treat the links established by effective communicators as the backbone of the communication network.

We devise overall metrics of importance and influence which use network-wise dynamic pageranking measures, *drank* and *starrank*. These metrics allow us to compare individuals' rise and fall in influence across days and further compare the moments by acceleration and length of monotonic runs. These metrics, taken as aggregates of sorting, filtering, and by taking the top results, amount to simple, but computationally efficient, models of influence. Armed with these models we uncovered multiple unexpected layers of socio-economic phenomena as expressed by Twitter signals of recognition that convey higher rankings, and we illuminated exchanges of signals in clear patterns we dubbed *the Mind Economy of Social Networks* (hence the thesis title).

We noticed that any momentum and trend-like measure leads to ratings which are prone to celebrity bias, and we furthered our metric design to attain features of real capital – inherent temporality with sensitivity to mutual status of interacting agents, and computability via iterative update rules. This approach lets us bootstrap a parallel economy which can be played along with the actual edge formation

in the original dynamic communication graph, with capital values adjusted on each node precisely according to our utility functions. in this way, utility of various groups can be parameterized and compared.

An important question about influence is whether, in social networks, we really have "Accidental Influentials" ? Although this is a simple question, it is a deep philosophical problem that leads to a definition of randomness, success, starting conditions, and various effective behaviors. Our computer science-based way to approach it is with the world-scale real data. We designed a methodology, based on simulating parallel worlds that share pieces of reality but with controlled attachment strategies and starting conditions for some or all classes of influentials as defined by our social capital ranking, (or any other real-valued function), for any amount of starting history based in the real world.

Using our simulations we found that inside each reasonably intelligent world by using smart attachment strategies the hierarchy of influence is stable and thus, the influentials are not accidental. At the same time the overlap of most of the simulated worlds with the real one is small thus showing that most of the influentials are accidental in terms of being a good fit to their own world with a small change in the rules leading to a different, also stable, hierarchy of those who could be more effective if everybody else would behave just a bit differently.

We presented methods to explore a social network communication graph by topic and community. Our procedure started with a topic and built up a community seeded by a pair of the most active communicators on the topic and their mutual friends (triangles), added recursively. A fringe of such a community represents people related to those organized around the topic, but also pointing to other topics of interest. We used n-gram analysis and statistically improbable phrases (SIPs) to characterize the topics of interest in a community and its fringe. The SIPs allow us to select communities of interest or further topics for browsing and community building, thus enabling community sensing by pivoting over SIPs. Our workflow provides an effective way to explore a vast social network in an iterative way. When using topics focused temporally or geographically fine-grained community sensing becomes possible.

In our study we applied dynamic graph analysis to the communication graph of Twitter repliers.

We developed metrics for individual users and their communities. The metrics express the growth of influence over days by finding the longest runs of growth in mentions, ranks, *starrank*, and their accelerations. We found that the *starrank* is a good indication of influence, and discovered a whole segment of the Twitter population who are naturally engaged in rank promotion by increasing the density of the dynamic communication graph, creating multiple accounts, raising the number of followers and repliers, etc. We also found actively growing national communities, one of them in Brazil, and we identified types of online personæ, such as journalists and musicians, who quantifiably generate the most engagement from the audience in the national social network. All those metrics are based purely on the graph and temporal structure of the communications. In our future work, we will use the actual text context of the messages exchanged and n-gram modeling to further quantify the dynamics of influence in communication over social networks.

We considered the importance metrics of communication networks and the fundamental questions they should answer. We introduced Reciprocal Social Capital, a currency-like measure which rewards socially beneficial behavior. Using this metric we discovered the "middle class" of Twitter, those whose conversations carry most of the steady discourse.

The key features of our capital is its reciprocity and hierarchy. Reciprocity rewards actions which balance the social network and reflect a fundamental dynamics in human relations. We found that ordering based on rewarding reciprocity and effort in maintaining one's social network contributions, segmenting various behaviors into different classes, results in a natural hierarchy.

When varying the components and parameters of the reciprocal social capital the hierarchy changes in an expected way, demonstrating that the elites are sensitive to reciprocity and general mentions. Reducing the weights of frequent talkers captures less of the top classes, while negating the rewards disturbs the hierarchy. The assumptions we make are thus quite reasonable, and positive variations lead to the same qualitative conclusions about reciprocity and hierarchy.

Our conclusions regarding the "Accidental Influentials" question are conditional. While celebrities may be accidental to the same degree as established social norms that govern replying are themselves random, they are not accidental when the norms are fixed and the behaviors are moderately intelligent.

In the same families of simulations seeded by different random sequences the same winners emerge in the social capital hierarchy. When simulations are based on vastly different principles, the highest ranking users are different, but use of similar principles leads to more overlap. The same winners persist when the simulations are seeded with varying lengths of reality, thus showing that our capital captures a significant part of their behavior.

Most importantly, while randomness of elites is still subject to interpretation the middle class is not accidental and defines the majority of the effects we care about.

Our iterative Mind Economy provides a powerful platform for discovering individual and group utilities and quantitative modeling of values of individuals, groups, and actions according to any such utility functions which can be further parameterized as desired. Our accidental influentials methodology is the first large-scale dissection of the combination of reality and simulation taking apart communicative behaviors. The foundation built here will lead to promising research and industrial applications.

Several well-known Internet companies and startups expressed interest in implementing our methodology. Research and development progress of our Mind Economy framework can be tracked at

`http://mindeconomy.com/`

The code is open source and available at

`http://github.com/alexy`

Our Twitter graph is also linked from there, with permission. Everybody is welcome to collaborate!

Bibliography

[1] Inc. 10gen. Mongodb – a fast caching nosql document database. `http://mongodb.org/`. 203, 205

[2] James Andreoni and John Miller. Giving according to garp: An experiment on the consistency of preferences for altruism. *Econometrica*, 70(2):737–753, March 2002. 36

[3] Lars Backstrom, Daniel P. Huttenlocher, Jon M. Kleinberg, and Xiangyang Lan. Group formation in large social networks: membership, growth, and evolution. In Tina Eliassi-Rad, Lyle H. Ungar, Mark Craven, and Dimitrios Gunopulos, editors, *KDD*, pages 44–54. ACM, 2006. 47, 49, 56

[4] E. Bakshy, J.M. Hofman, W.A. Mason, and D.J. Watts. Everyone's an influencer: quantifying influence on twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 65–74. ACM, 2011. 9, 32, 40, 84

[5] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999. 34

[6] Jeffrey Baumes, Mark Goldberg, and Malik Magdon-Ismail. Efficient identification of overlapping communities. 25

[7] Andrew Boekhoff. Congomongo – a clojure wrapper for the mongodb java api. `http://github.com/somnium/congomongo/`. 205

[8] P. Bonacich. Power and centrality: A family of measures. *The American Journal of Sociology*, 92(5):1170–1182, 1987. 73

[9] Phillip Bonasich. Patterns of coalitions in exchange networks: an experimental study. *Rationality and Society*, 12(353), 2000. 47

[10] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 30(1-7):107–117, 1998. 57

[11] Shane Butler. *The hand of Cicero*. Rutledge, 2002. 16

[12] Bob Carpenter. Lingpipe – a suite of java libraries for the linguistic analysis of human language. `http://alias-i.com/lingpipe/`. 204

[13] R Cazabet, F Amblard, and C Hanachi. Detection of overlapping communities in dynamical social networks. *pubftp.computer.org*. 25

[14] Stanley Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. Technical Report TR-10-98, Harvard University, August 1998. 53

[15] D. Christensen. Fast algorithms for the calculation of kendall□s τ. *Computational Statistics*, 20(1):51–62, 2005. 110

[16] Wayne Chung, Robert Savell, Jan-Peter Sch utt, and George Cybenko. Identifying and tracking dynamic processes in social networks. *Proceedings of SPIE, Vol. 6201, Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense V, Edward M. Carapezza, Editors, 620105*, 10 May 2006. 56

[17] D. Cosley, D. Huttenlocher, J. Kleinberg, X. Lan, and S. Suri. Sequential influence models in social networks. In *4th International AAAI Conference on Weblogs and Social Media*, Washington, DC, 2010. AAAI. 32, 40

[18] Doug Cutting, Otis Gospodnetic, and Grant Ingersoll. Lucene – a high-performance, full-featured text search engine library written entirely in java. `http://lucene.apache.org/`. 47

[19] George Cybenko and Vincent H. Berk. Process query systems. *Computer*, 40(1):62–70, 2007. 56

[20] Marc Smith Derek Hansen, Ben Shneiderman. *Analyzing Social Media Networks with NodeXL: Insights from a Connected World*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 2011. 34, 73

[21] G.W. Domhoff. Who rules america now? 9

[22] G.W. Domhoff. *Who rules America?: power and politics, and social change*. McGraw-Hill Humanities/Social Sciences/Languages, 2006. 9, 102

[23] Nathan Eagle and Alex (Sandy) Pentland. Reality mining: sensing complex social systems. *Pers Ubiquit Comput*, 10(4):255–268, May 2006. 75

[24] G Flake, S Lawrence, C Giles, and F Coetzee. Self-organization of the web and identification of communities. *Communities*, Jan 2002. 25

[25] Gary William Flake, Steve Lawrence, and C. Lee Giles. Efficient identification of web communities. In *KDD*, pages 150–160, 2000. 47

[26] Noah Friedkin. *A Structural Theory of Social Influence*. Cambridge University Press, 2006. 47

[27] Devin Gaffney. #iranelection: Quantifying online activism. *Proceedings of the WebSci10: Extending the Frontiers of Society On-Line, April 26-27th, 2010, Raleigh, NC*, 2010. 25

[28] Dario Gaggio. *In Gold We Trust: Social Capital and Economic Change in the Italian Jewelry Towns*. Princeton University Press, 2007. 29, 74

[29] Mark K Goldberg, Mykola Hayvanovych, and Malik Magdon-Ismail. Measuring similarity between sets of overlapping clusters. pages 1–6, Jul 2010. 25

[30] Mark Granovetter. Threshold models of collective behavior. *The American Journal of Sociology*, 83(6):1420–1443, May 1978. 31

[31] Mikhail Gronas. Private communication, 2010. 20

[32] J. Habermas and T. Burger. *The structural transformation of the public sphere*. MIT Press, 1989. 33

[33] Rich Hickey. Clojure – the lisp that makes the jvm dynamic. `http://clojure.org/`. 203, 204

[34] E. Muller J. Goldenberg, B. Libai. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing Letters*, 12(3):211–223, 2001. 31

[35] E. Muller. J. Goldenberg, B. Libai. Using complex systems analysis to advance marketing theory development. *Academy of Marketing Science Review*, 2001. 31

[36] M.O. Jackson. *Social and economic networks*. Princeton University Press, 2008. 36, 37, 71

[37] Akshay Java, Xiaodan Song, Tim Finin, and Belle L. Tseng. Why we twitter: An analysis of a microblogging community. In *WebKDD/SNA-KDD*, pages 118–138, 2007. 49, 56, 72

[38] E. Katz, P.F. Lazarsfeld, and Columbia University. Bureau of Applied Social Research. *Personal influence: The part played by people in the flow of mass communications*. Free Press New York, 1955. 4, 9, 38

[39] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM, 2003. 31

[40] D Kempe, J Kleinberg, and É Tardos. Influential nodes in a diffusion model for social networks. *Automata*, Jan 2005. 31

[41] Alexy Khrabrov. Conversation graph mining toolkit. `http://github.com/alexy/tfitter/`. 207

[42] Alexy Khrabrov. Conversation graph mining toolkit, 2010. 205

[43] Alexy Khrabrov and George Cybenko. A language of life: Characterizing people using cell phone tracks. In *CSE (4)*, pages 495–501. IEEE Computer Society, 2009. 56

[44] Alexy Khrabrov and George Cybenko. A language of life: Characterizing people using cell phone tracks. In *Proceedings IEEE CSE'09, 12th IEEE International Conference on Computational Science and Engineering, Vancouver, BC, Canada*, pages 495–501, 2009. August 29-31. 75

[45] Alexy Khrabrov and George Cybenko. Discovering influence in communication networks using dynamic graph analysis. *Proceedings IEEE CSE'10, 13th IEEE International Conference on Computational Science and Engineering, Minneapolis, MN (to appear)*, 2010. 47, 72, 73

[46] Alexy Khrabrov, David M Pennock, C. Lee Giles, and Lyle H Ungar. Static and dynamic analysis of the internet's susceptibility to faults and attacks. *INFOCOM 2003*, Sep 2003. 34, 74

[47] W.R. Knight. A computer method for calculating kendall's tau with ungrouped data. *Journal of the American Statistical Association*, 61(314):436–439, 1966. 110

[48] Charles Lamb. Loading twitter data into berkeley db java edition. `http://www.javaoracleblog.com/java/Loading_Twitter_Data_into_Berkeley_DB_Java_Edition.jsf`. 204

[49] Louis Licamele, Mustafa Bilgic, Lise Getoor, and Nick Roussopoulos. Capital and benefit in social networks. *LinkKDD '05: Proceedings of the 3rd international workshop on Link discovery*, Aug 2005. 29, 74

[50] David Edgar Liebke. Incanter – a clojure-based, r-like platform for statistical computing and graphics. `http://incanter.org/`. 205

[51] Nina Mishra, Robert Schreiber, Isabelle Stanton, and Robert Endre Tarjan. Clustering social networks. In Anthony Bonato and Fan R. K. Chung, editors, *WAW*, volume 4863 of *Lecture Notes in Computer Science*, pages 56–67. Springer, 2007. 47, 50

[52] M Mobius, D Quoc-Anh, and T Rosenblat. Social capital in social networks. *Retrieved December*, Jan 2004. 36

[53] BKD Motidyang. *A Bayesian belief network computational model of social capital in virtual communities. Ph.D. Thesis in Computer Science*. University of Saskatchewan, 2007. 35, 74

[54] Martin Odersky. Scala – an object-oriented functional programming language. `http://scala-lang.org/`. 203

[55] Joshua O'Madadhain, Danyel Fisher, and Tom Nelson. Jung – a free and open-source java software library for manipulating, analyzing, and visualizing network data. `http://jung.sourceforge.net/`. 57

[56] J.J. Phillips. Atticus and the publication of cicero's works. *The Classical World*, pages 227–237, 1986. 16

[57] Robert Savell and George Cybenko. Mining for social processes in intelligence data streams. Social Computing, Behavioral Modeling, and Prediction, Edited by Huan Liu, John J. Salerno and Michael J. Young, 2008. 47, 56

[58] T. Schelling. *Micromotives and Macrobehavior*. Norton, 1978. 31

[59] Margo Seltzer. Berkeley db java edition – a pure-java, high performance, transactional, non-relational, persistence solution for java objects. `http://www.oracle.com/technology/products/berkeley-db/`. 203

[60] F Sudweeks and S Simoff. Culturally commercial: A cultural e-commerce framework. *Proc. OZCHI 2001*. 20

[61] Mike Tanier. Ochocinco is a master at twitter, March 25 2011. 5

[62] G. Tarde and T.N. Clark. *On communication and social influence: Selected papers*. Univ. Press, 1969. 33, 84

[63] John W. Tukey. *Exploratory Data Analysis*. Addison Wesley, 1977. 27, 42

[64] Victor H. Vroom and Philip W. Yetton. *Leadership and Decision-Making*. University of Pittsburgh Press, 1973. 22

[65] D.J. Watts and P.S. Dodds. Influentials, networks, and public opinion formation. *Journal of Consumer Research*, 34(4):441–458, 2007. 4, 9, 38, 80

[66] Duncan J. Watts, Yoshito Hori, Frédéric Dalsace, Coralie Damay, David Dubois, Michael Schrage, Harry Hutson, Barbara Perry, Eric von Hippel, Linda Stone, Michael C. Mankins, Ap Dijksterhuis,

Robert G. Eccles, Liv Watson, Mike Willis, Geoffrey B. West, Karen Fraser, Phillip Longman, Rashi Glazer, and Yoko Ishikura. the hbr list. *Harvard Business Review*, 85(2):20 – 54, 2007. 39, 80

[67] G.K. Zipf. *Human behavior and the principle of least effort: An introduction to human ecology*. Hafner New York, 1949. 33, 102