Patterns in Knowledge Production

by

Miles Kirkland Manning

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved July 2017 by the
Graduate Supervisory Committee:

Carlos Castillo-Chavez, Co-Chair
Marco Janssen, Co-Chair
Marty Anderies
Yun Kang

ARIZONA STATE UNIVERSITY

August 2017

ABSTRACT

This dissertation will look at large scale collaboration through the lens of online communities to answer questions about what makes a collaboration persist. Results address how collaborations attract contributions, behaviors that could give rise to patterns seen in the data, and the properties of collaborations that drive those behaviors.

It is understood that collaborations, online and otherwise, must retain users to remain productive. However, before users can be retained they must be recruited. In the first project, a few necessary properties of the "attraction" function are identified by constraining the dynamics of an ODE (Ordinary Differential Equation) model. Additionally, more than 100 communities of the Stack Exchange networks are parameterized and their distributions reported.

Collaborations do not exist in a vacuum, they compete with and share users with other collaborations. To address this, the second project focuses on an agent-based model (ABM) of a community of online collaborations using a mechanistic approach. The ABM is compared to data obtained from the Stack Exchange network and produces similar distributional patterns.

The third project is a thorough sensitivity analysis of the model created in the second project. A variance based sensitivity analysis is performed to evaluate the relative importance of 21 parameters of the model. Results indicate that population parameters impact many outcome metrics, though even those parameters that tend towards a low impact can be crucial for some outcomes.

DEDICATION

To Mom,

For teaching me to do a little bit extra,

and to maintain forward motion.

ACKNOWLEDGMENTS

I would like first to acknowledge the years of funding provided through the Simon A. Levin Mathematical, Computational and Modeling Sciences Center; thank you for allowing me the means to pursue my passion.

I also need to thank my committee members, who guided me through the last, hardest part of my journey. Innumerable thanks are owed to Marco Janssen, for keeping me on task and moving forward; to Carlos Castillo-Chavez, for keeping me around and supporting me throughout; to Marty Anderies, for jumping through hoops so I could progress; and to Yun Kang, for the monumental, last-minute effort needed to make my defense possible.

I would like to thank Anuj Mubayi for attending my dissertation in place of a committee member and Lingfei Wu for getting me started on the data analysis.

I also owe a debt of gratitude to the staff of SAL MCMSC. Margaret: thank you for making defense happen in spite of what at times seemed impossible hurdles. Sherry: thank you for your much-needed support and kindness.

Thank you to all of my fellow students. I appreciated both the kinship and commiseration over just how troublesome research can be.

Finally, I'd like to thank my family and friends. I suffer no delusions that I could have accomplished this on my own. Mom: thank you for driving me to school, and, really, driving me through school. Dad: Thank you for making it all possible. Thanks to Marie, for proofreading and editing even when you didn't understand what exactly you were reading. To Clare, McKenna, Mason and Max: thank you for taking so many math classes for so many years. Your questions and expectation that I could help really kept me on my toes, and kept more math in my head than anything else would have. To my whole family: thank you for the hours waiting, the hours helping, the years supporting.

Thank you Amanda, for being in the same dissertation-writing boat and willing to whine about it with me.

Thank you Megan, for always being there.

TABLE OF CONTENTS

CHAPTER Page

APPENDIX

List of Tables

List of Figures

Figure Page

Figure Page

Chapter 1

INTRODUCTION

1.1 Background

Throughout history, large scale collaboration has allowed for the foundation of cities, the establishment of trade networks, scientific progress, medical breakthroughs and the advancement of human rights. As failed collaborations do not leave much in the way of artifacts or historical records, the most visible of these collaborations are those that succeeded. Thus, exploration of what determines success in collaboration is largely limited to observing and mimicking success, rather than learning from failure.

Not all collaborations are beneficial to the public, those that are are called a public good. A public good is a good that is both non-excludable and non-rivalrous (Bade and Parkin, 2012) and this can be thought of as something that can be used by anyone and that isn't used-up by use. In practice, perfect examples of public goods are difficult to come by so we call those goods that are close, public goods. Examples of public goods include national defense, knowledge, and clean air. Public goods are susceptible to what is called the "free-rider problem". A free-rider is someone who benefits from a good without contributing to it. In public goods the possibility of free-riding reduces the perceived value of contribution as the benefit is shared by contributors and free-riders alike. One solution to the "free-rider problem" is government intervention: this can be seen for example in the maintainance of national defence with tax payer dollars, a system which minimizes the possibility of free-riders.

1.1.1 Online Public Goods

Many online collaborations can be thought of as creating a public good. Online collaborations can specifically be categorized as sources of information goods. Information goods are non-rivalrous because they are not used-up by use and many online communities are non-exclusive even if exclusion is possible. Wikipedia (`https://www.wikipedia.org/`), a household name, is a textbook example of a non-rivalrous, non-exclusive online collaboration. Although a good example of such a collaboration, it is hardly the only one: Stack Exchange (`https://stackexchange.com/`), Github (`https://github.com/`), and the Linux operating system (`https://www.linux.com/`) are all additional examples very large scale online collaborations producing a public good.

Studying collaboration through the lens of online communities carries a significant advantage over observing other collaborations or experimentation for a variety of reasons related to the specific properties of the internet. Online communities have readily available data for both successful collaborations and for failed collaborations; they also involve far more participants than could be reasonably be enrolled in a typical social science experiment. From data sourced from online collaboration we can learn what separates successful collaborations from unsuccessful collaborations rather than what successful collaborations have in-common.

1.1.2 Experimental Public Goods

The evolutionary stability of cooperation is a constant subject of interest. There are several possible explanations for how altruistic behavior might evolve, or at least apparently altruistic behavior, these include: direct reciprocity, indirect reciprocity, and kin-selection (Nowak and Highfield, 2011). The key similarity between all of these components is that they in some way share information. Direct reciprocity can drive cooperation when players interact repeatedly, it is beneficial to foster cooperation because, over the long term, it is mutually beneficial. Indirect reciprocity requires information be shared in some other way, usually a reputation system, as this forces players to carry a history of there choices with them providing similar information to that which is obtained through direct reciprocity. Kin-selection only works among kin and is based on the priniple that shared genetic information encourages altruistic behavior among related individuals for the ultimate benefit of the group.

Another cooperation reinforcing mechanism is punishment. A direct comparison between punishment and communication suggests that communication serves better than punishment to improve outcomes Janssen *et al.* (2010). However, the effects of punishment on strengthening cooperation have been observed in human subject research (Ostrom *et al.*, 1992) as well as in evolutionary simulations (Boyd *et al.*, 2003). The greatest drawback of this system is the cost of maintaining the punishment system, and the best cost is not clear. If punishment is too expensive to dole out it cannot sufficiently restrain defectors, however if it is too inexpensive it can be easily overused. Since punishment systems must be maintained, significantly more cooperation is needed to achieve the same outcomes when punishment is funded.

Cooperation often comes down to trust, and it occurs when the cooperators believe it is going to occur. In small groups cooperation is relatively easy maintain, and in reputation systems the trust is placed in the reputation rather than uniformed expectation, in punishment systems players trust that others won't risk punishment. In large groups cooperation is more difficult to maintain (Boyd *et al.*, 2003; Janssen *et al.*, 2014). Different constraints produce different results but the general trend is that cooperation decreases with group size. However, large scale cooperation does occur, sometimes driven through punishment or reputation systems, often both.

1.2 Prior Research

For the purpose of reviewing prior literature we look at three categories of related work: First, we will discuss patterns found in online communities; Second, we will summarize some of the previous models used for analyzing online communities. These patterns are summarized in Table 1.1.

1.2.1 Patterns in Online Communities

In the context of online communities, "success" is not well defined. Rather than try to categorize communities into those that are successful and those that are not, most look at the distributions of various outcomes. The patterns observed depend in part on how the data is obtained.

Reference	**Subject**	**Users per Community**	**Tasks per User**	**Findings**
Wilkinson (2008)	Peer-Production	$2 - 4$	$10 - 30$	scale free distribution of contributions per user slope of distribution task dependent scale free distribution of contributions per community
Fu *et al.* (2008)	Blogging	1	NA	power law in-degree power law out-degree

Wu *et al.* (2009)	Digg/YouTube	NA	NA	attention motivates production power law user contribution in Digg long tail user contribution in YouTube
Huberman *et al.* (1998)	Browsing	NA	NA	power law browsing behavior
Huberman and Adamic (1999)	Website	NA	NA	power law size of website
Albert and Barabási (2002)	Internet	NA	NA	power law degree distribution
Barabási and Albert (1999)	WWW	NA	NA	power law degree
Broder *et al.* (2000)	WWW	NA	NA	power law weakly connected components power law strongly connected components
Radtke (2011)	FLOSS	2	$20 - 30$	power law project developers
Ozmen *et al.* (2012)	Participatory Science	2	1	power law user contribution
Kittur and Kraut (2010)	Wiki	10	0.2	Zipf law wiki users Zipf law wiki edits

Yasseri and Kertész (2013)	Wikipedia	$0.25 - 1.25$	100	power law article editors log normal inter edit time power law session edits

Table 1.1: Characteristics of Online Communities

Web crawlers can provide information on the structure of the web, and provide the first look data for understanding the environment in which online communities live. If web pages are considered as nodes and hyperlinks as edges, both the in-degree and out-degree of blogs obey a power-law distribution (Fu *et al.*, 2008). Further, both the web (Barabási and Albert, 1999) and the internet (Albert and Barabási, 2002) have a degree distribution that is power-law. Lastly, the size of both strongly and weakly, connected components, are also power-law distributed (Broder *et al.*, 2000).

A key source of information for modeling collaborative communities is data collected from some exemplar communities. For example, Wikis are a popular platform for online collaboration and excellent source of data. The number of editors a wiki has appears to be distributed according to a Zipf law, as does the number of edits until falling below a critical mass Kittur and Kraut (2010). Furthermore, the distribution of Wikipedia editors per article appears to follow a power-law distribution for most of the range Yasseri and Kertész (2013).

The next pattern of interest is the behavior of users. First, the number of clicks made while browsing the internet is distributed according to a power law Huberman *et al.* (1998). More relevantly, user contribution seems to be driven by attention (Wu *et al.*, 2009). While this does not always result in a power-law, the distribution seems to consistently have a high probability of few contributions, and a power law on a portion of the domain (Wilkinson, 2008; Wu *et al.*, 2009; Ozmen *et al.*, 2012). The slope of this power law is dependent on the community (Wilkinson, 2008), as is the domain (Wu *et al.*, 2009).

1.2.2 Models of Online Communities

A survey of earlier papers on modeling online communities reveals mechanisms of user action thought to be present. In order to narrow the scope of research surveyed, the review is limited to works that take a mechanistic approach to modeling online communities. To further limit the works examined, models of online social networks are excluded from the survey. For the purpose of this work, communities are considered productive communities if they create or distribute knowledge, or practice peer production. Examples include open source software, Wikipedia, and question and answer forums such as Stack Exchange. With these restrictions, a few different mechanisms are found that, in various forms, are used to model online communities. These three mechanisms, known as preferential attachment, foraging, and infection, apply to different aspects of online communities.

Power law distributions are a hallmark of online communities (Wilkinson, 2008; Fu *et al.*, 2008; Barabási and Albert, 1999; Broder *et al.*, 2000; Huberman *et al.*, 2009). Preferential attachment is a mechanism for the formation of power law distributions, which has been verified to occur in online communities (Pastor-Satorras *et al.*, 2001). The idea that “the rich get richer” is captured in the positive feedback of preferential attachment. If one were to apply this idea to online communities, one could say that a new user is more likely to join the larger communities than the smaller. This concept could also be applied to tasks or connections between communities. That is, communities with a large number of tasks will draw more new tasks, or highly connected communities are more likely to form new connections.

One way of explaining the time people spend contributing to online communities is to assume they derive some benefit from this activity. If contributing satisfies some need for users, they can be considered to forage in communities for tasks. One successful foraging heuristic is win-stay/lose-shift, which is seen when a forager categorizes their outcomes as either sufficient or not, and moves from their foraging site only when their outcomes are insufficient (Nowak and Highfield, 2011; Ozmen *et al.*, 2012). In the language of online communities, one would predict that a user remains in a community only as long as the user is able to contribute to that community.

Another modeling framework with some applicability to online communities is that providided by epidemiology. It has been observed that popularity drives production in online communities, and considering contributors as infectious is one way of explaining or conceptualizing this phenomenon. Users could be seen to follow an SEIR (Susceptible, Exposed, Infectious, Recovered) progression, where unattached susceptible users (S) are exposed to a community or task by contributors (I), after which they are considered exposed (E). After some time, exposed users progress in turn to active contributors (I), eventually contributors (I) leave the community and begin a refractory period (R) during which they will not re-join the community (Ozmen *et al.*, 2012). One of the common and basic properties of infectious disease models is that the number of new infections is proportional to the number of infectious people. This corresponds to the number of active contributors in a community, thus determining its attractiveness. The above presented parallel concept highlights the importance of popularity in online communities and their self-reinforcing behavior.

From the review a few key mechanisms used in the modeling of online communities can be identified: users must follow other users; positive feedback of popularity is included in both preferential attachment and epidemiological models; contributions must be a commodity sought after by users and communities. This can be explained by the observation that attention drives contribution (Huberman *et al.*, 2009; Wu *et al.*, 2009). Since users follow tasks, and popularity predicts user movement, tasks must occur with higher frequency in larger communities. The next section describes the Stack Exchange network which will be the primary data source for this dissertation.

1.3 Stack Exchange

The main source of data for this dissertation will be the Stack Exchange network. Stack Exchange is a network of Question & Answer forums that boasts five million registered users, 3.7 million questions, and 4.6 million answers (`https://stackexchange.com/about` for current numbers). With more than 100 million unique visitors per month, Stack Exchange is a very well used information good.

As previously mentioned, the Stack Exchange network is a number of connected Question & Answer forums. Because of the connected but distinct forums, Stack Exchange is able to provide expert answers on a wide range of topics. With more than 150 (no clear consensus on if a few sites should be counted) member sites ranging from 'math.stackexchange.com' and 'serverfault.com' (created before the aggregated domain and one of the sites that may or may not be counted) to 'martialarts.stackexchange.com' and 'italian.stackexchange.com' the breadth of topics is undeniable.

Not only does the Stack Exchange network include a great many sites, it is always growing. The process for adding a site to the Stack Exchange network is handled at 'area51.stackexchange.com' here, ideas for new sites are formed, refined, and eventually tested. To create a new site a user must propose the topic to the community and provide a few example questions; the purpose of the site is refined through discussion and users are asked to commit to using the site. If a proposal generates enough interest at area51 it will go into a public beta and, if it maintains sufficient activity, become a member of the Stack Exchange network.

Though each member site of the Stack Exchange network has a unique topic, each is modeled after 'stackoverflow.com' the original Stack Exchange site. A contributor to stackoverflow's success was its gamification of the collaborative process. New users can ask and answer questions and by doing so, increase their reputation through accumulated points. As a user increases their reputation they unlock the ability to comment, vote on answers, and eventually moderation tools. Reputation is largely a measure of how familiar a user is with a given site and as such does not transfer between Stack Exchange sites. Further details about how the stack exchange sites work can be found at 'meta.stackexchange.com' the Stack Exchange site about Stack Exchange.

The Stack Exchange data used in this dissertation can be found at `https://archive.org/details/stackexchange` and was downloaded in December of 2016. The recorded data includes details and time stamps on all posts as well as users, votes, comments, et cetera. Because each site in the Stack Exchange network is distinct but also a part of the network, we see the results of many natural experiments with their results all recorded in the same way. The structure of the Stack Exchange network and the quality of recorded data allows us to look at the distributional characteristics of the outcomes of online collaborations. This data set includes data for 168 stack exchange sites and their meta sites (a Q&A forum about the site itself). Example code used to read the data is available via OpenABM at `https://www.openabm.org/model/5727/version/1/view`.

1.4 Questions

One of the questions in online collaboration is the conditions in which a community will persist. We will model both an isolated community (Chapter 2) and and a community of communities (Chapter 3 and Chapter 4) to explore these conditions. In Chapter 2 we hypothesize that the number of questions entering a Q&A forum can be predicted from the active questions. In Chapter 3 we consider the hypothesis that users move through communities like like foragers through the environment.

In this dissertation I will use simulation and mathematical modeling in conjunction with Stack Exchange data to answer questions about online collaborations and to inform on large scale collaborations in general. In the next chapter I will use differential equations and patterns in user behavior to identify plausible functional forms for how communities attract new users. In the subsequent chapter I will take an agent-based modeling approach and use known behavioral mechanisms to reproduce the distributional patterns seen in the Stack Exchange data. Lastly, I will perform a thorough sensitivity analysis on the agent-based model to identify how each mechanism influences each outcome. The manuscript will end with a conclusion discussing interpretation of results and how these projects inter-relate.

Chapter 2

POSSIBLE MECHANISMS FOR ATTRACTING QUESTIONS TO ONLINE Q&A

2.1 Introduction

The importance of user retention is well known in collaborations both online and otherwise. For a collaboration, or any group to last, you need more members to join than to leave. Improving member retention reduces the number of members leaving a collaboration, however perfect retention is not sufficient, a collaboration needs recruitment. A population is stable when recruitment and loss are balanced, this is as true for online collaborations as any other population. User retention can be measured for online communities, and because of its importance, often is. Recruitment is much harder to measure, online communities cannot directly track potential members and observe what brings them into the community. Since recruitment is as important as retention but cannot be directly observed, we turn to modeling to explore what might determine how attractive a collaboration is to new users. In this chapter we will explore how the population dynamics of Question & Answer forums constrain the possibilities for how new questions are attracted.

2.2 Model

In order to look at how online collaborations might attract contributions (and contributors) we construct a population ecology model of the questions in a question and answer forum. Question and answer forums are significant in that there are two opposing forms of contribution; questions and answers. We consider three "life stages" of questions, there are new unanswered questions, questions that have had answers attempted but have not been answered, and questions that have been answered. Not every question will go through every stage, it is possible that the first attempt at an answer results in an accepted answer. Previous work has demonstrated that there are at least two strategies for answering questions in the Stack Exchange network (Wu *et al.*, 2016). If a user adopts Type A strategy, it will favor new questions, but the answers might not have a high likelihood to be accepted. Users identified by a type B strategy contribute to questions that already had at least one attempt at an answer and are more likely to provide an answer that might be accepted. We can provide the following caricature for the two strategies. Type A users are focused on getting reputation points by responding quickly to new questions, while Type B users are more knowledgeable and experienced and only bother with questions that have not been successfully been answered before. The dynamics of this system can be seen in Figure 2.1 and in the System of Equations (2.1). Rather than model the question and user populations of a forum separately, we assume that the populations scale together. A possible explanation for this assumption is that either open questions attract users, or the same things that attract questions attract users.

$$\dot{Q}_1 = f(Q_1, Q_2) - \beta A Q_1 \tag{2.1}$$

$$\dot{Q}_2 = (1 - \alpha_A)\beta A Q_1 - \omega B Q_2 + (1 - \alpha_B)\omega B Q_2 \tag{2.2}$$

$$\dot{Q}_3 = \alpha_A \beta A Q_1 + \alpha_B \omega B Q_2 \tag{2.3}$$

The model follows the possible "life histories" of questions. A new question (Q_1) enters the system according to some attraction function $f(Q_1, Q_2)$, this attraction depends on the current activity of the system. New questions are then answered at a rate β proportional to the number of new questions and the proportion of answerers employing a type A answering strategy (A). Questions answered by type A answer-

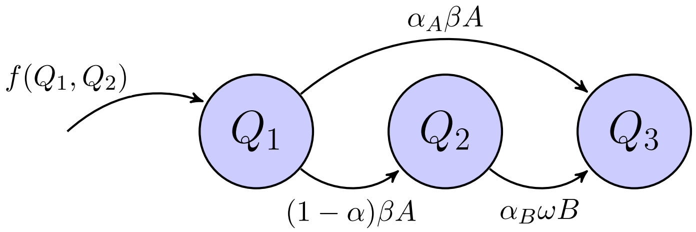


Figure 2.1: Visualization of Model of Question "Life Stages"

ers have an acceptable answer with probability α_A and are split between questions without an acceptable answer (Q_2) and questions with an acceptable answer (Q_3). Type B users answer questions that have already been attempted (Q_2) at a rate ω proportional to the number of attempted questions (Q_2) and the proportion of users employing strategy B (B). Questions answered by type B users have acceptable answers with probability α_B, however unlike with type A, if the answer is not acceptable the question does not change compartments. It is important to note that:

$$\dot{Q}_1 + \dot{Q}_2 + \dot{Q}_3 = f(Q_1, Q_2)$$

. This tells us that the only way the total number of questions increases is through the function $f(Q_1, Q_2)$, because of this the properties of $f(Q_1, Q_2)$ are important to both the model and to the real system.

By combining parameters and removing Q_3 we can re-write the system of Equations (2.1) as the system of Equations (2.4). The re-written system is equivalent to the original system as well as using fewer state variables and parameters, making it more amendable to analysis. In the system of Equations (2.4) we have three parameters: δ is the rate at which unanswered questions receive an answer, α_A is the probability that an answer provided by a type A user is acceptable, and γ is the rate at which previously answered questions receive an acceptable answer.

$$\dot{Q}_1 = f(Q_1, Q_2) - \delta Q_1 \tag{2.4}$$

$$\dot{Q}_2 = (1 - \alpha_A)\delta Q_1 - \gamma Q_2 \tag{2.5}$$

2.3 Analysis

Having formulated a model for the population ecology of questions on a Q&A forums we now turn to analysis to impose constraints on the system. Information about the real behavior of Q&A forums provides constraints on the dynamical properties the system should have. By constraining the dynamics of the model we can identify properties of the function $f(Q_1, Q_2)$, this is significant because $f(Q_1, Q_2)$ represents the flow of new questions into a Q&A forum.

Among the most obvious and consistent properties of online collaborations is their non-negativity. No collaboration can have a negative number of members or new questions and so the System of Equations (2.4) should be positively invariant for $Q_1 \geq 0$ and $Q_2 \geq 0$. To constrain the model to the positive quadrant we need $\dot{Q}_1(Q_1 = 0) \geq 0$ and $\dot{Q}_2(Q_2 = 0) \geq 0$. The resulting condition is that $f(0, Q_2) \geq 0$, considering the meaning of $f(Q_1, Q_2)$ this is an unsurprising condition that should always hold as there exist no negative questions to attract. Given the meaning of $f(Q_1, Q_2)$ the above result is generalized to $f(Q_1, Q_2) \geq 0$.

To further constrain $f(Q_1, Q_2)$ the System of Equations (2.4) is set equal to 0 and equilibria are identified. This gives a relation between Q_1 and Q_2 as well as one between Q_1 and $f(Q_1, Q_2)$, these can be found in Equation (2.8). Further, these can be combined into a condition between $Q_1^\star$ and $f(Q_1^\star, Q_2^\star)$ as seen in Equation (2.10)

$$0 = f(Q_1^\star, Q_2^\star) - \delta Q_1^\star \tag{2.6}$$

$$0 = (1-\alpha_A)\delta Q_1^\star - \gamma Q_2^\star \tag{2.7}$$

$$Q_2^\star = \frac{(1-\alpha_A)\delta}{\gamma} Q_1^\star \tag{2.8}$$

$$\delta Q_1^\star = f(Q_1^\star, Q_2^\star) \tag{2.9}$$

$$\delta Q_1^\star = f\left(Q_1^\star, \frac{(1-\alpha_A)\delta}{\gamma} Q_1^\star\right) \tag{2.10}$$

An online collaboration with no activity is unlikely to gain activity. In terms of the model "no activity" is represented as $(Q_1, Q_2) = (0, 0)$ so we expect $(0, 0)$ to be an equilibrium of the model. Using Equation (2.10) and the assumption that $(0, 0)$ is an equilibrium we have that $f(0, 0) = 0$, that is a community with no activity attracts no new questions. Additionally, given that collaborations do not grow without bound, we suspect there is a second point that satisfies the Equation (2.10), that is $f(Q_1^\star, Q_2^\star) = \delta Q_1^\star$. Thus we have $f(Q_1, Q_2) \geq 0$, $f(0, 0) = 0$, and at least one pair $Q_1^\star$, and $Q_2^\star$ such that $f(Q_1^\star, Q_2^\star) = \delta Q_1^\star$.

In addition to having at least two equilibria we can constrain the system with the condition that there not be infinite equilibria. For this to hold, the nullclines Equations (2.8) must intersect a finite number of times. Since the nullcline equation $Q_2^\star = \frac{(1-\alpha_A)\delta}{\gamma} Q_1^\star$ is a line, the other nullcline cannot be a line because the two required intersections would result in the two lines being equivalent. Thus , $\forall c_1, c_2 \in \mathbb{R}$, $f(Q_1, Q_2) \neq c_1 Q_1 + c_2 Q_2$.

Continuing with analysis of stability we can further constrain the attraction function. For a two dimensional system, stability can be determined from the trace and

f_{Q_1}	$f_\star$	stability
f_{Q_1}	$f_\star > \frac{\delta}{\gamma}$	saddle
$f_{Q_1} > \delta + \gamma$	$f_\star < \frac{\delta}{\gamma}$	unstable
$f_{Q_1} < \delta + \gamma$	$f_\star < \frac{\delta}{\gamma}$	stable

Table 2.1: Summary of the Constraints on f at an Equilibrium

determinant of the Jacobian matrix. From the trace and determinant we identify thresholds on the partial derivatives of the attraction function (f_{Q_1}, f_{Q_2}) for the stability of the system. It is at this point useful to define $f_\star$, the partial derivative of f along the linear nullcline defined in Equations (2.8) or equivalently $\frac{df}{dQ_1}$ evaluated at the interior equilibrium. The results of the stability analysis are summarized in table 2.1.

$$J = \begin{bmatrix} f_{Q_1} - \delta & f_{Q_2} \\ (1 - \alpha_A)\delta & -\gamma \end{bmatrix}$$

$$Trace(J) = f_{Q_1} - \delta - \gamma \tag{2.11}$$

$$Det(J) = \delta\gamma - \gamma f_{Q_1} - (1 - \alpha_A)\delta f_{Q_2} \tag{2.12}$$

$$f_\star = f_{Q_1} + \frac{(1 - \alpha_A)\delta}{\gamma} f_{Q_2} \tag{2.13}$$

Considering the behavior of real online collaborations we can infer constraints on f_{Q_1} and f_{Q_2}. If a collaboration has very little activity, it is expected to fail, this suggests that the no activity equilibrium is at least locally stable. Thus we have that $f_{Q_1}(0,0) < \delta + \gamma$ and that $f_\star(0,0) < \frac{\delta}{\gamma}$. Similarly there are long lived collaborations that do not grow without bound, this is indicative of a second stable equilibrium.

That is, $\exists(Q_1^\star, Q_2^\star)$ such that:

$$\dot{Q}_1(Q_1^\star, Q_2^\star) = 0 \tag{2.14}$$

$$\dot{Q}_2(Q_1^\star, Q_2^\star) = 0 \tag{2.15}$$

$$f_{Q_1}(Q_1^\star, Q_2^\star) < \delta + \gamma \tag{2.16}$$

$$f_\star(Q_1^\star, Q_2^\star) < \frac{\delta}{\gamma} \tag{2.17}$$

2.4 Parameter Estimation

As discussed in Chapter 1, Stack Exchange is a network of Question & Answer forums and we use the collected data to estimate model parameters. Because Stack Exchange is a single network with the same underlying structure and rules for all sites, we can think of the sites as natural experiments. All data was collected in the same way because it was collected by the same tool, this makes the distribution of parameter estimates we find more meaningful as it reduces the confounding variables. We use data for 168 Stack Exchange sites, the data was downloaded from the repository located at `https://archive.org/details/stackexchange`.

The parameters of the model can be estimated directly from Stack Exchange data. The process for parameter estimation begins with separating the answering population into type A and type B users. In the model, types A and B are archetypes that are completely disjoint, in reality people employ both strategies to varying degrees. To estimate identify type A users we first calculate the mean number of prior attempts an attempted question has for each user. Taking the mean of the average number of prior attempts allows us to divide the population into those that answer newer questions (type A, lower number of prior attempts), and those that answer older questions (type B, higher number of prior attempts). After each user has been categorized, calculating the fractions of users that are type A and B is trivial.

Table 2.2: Summary of the Distribution of Four Parameters Measured From Data. A and B are Dimensionless, δ and γ Have Units of $Days^{-1}$

Parameter	Min	1^{st} quartile	median	3^{rd} quartile	Max	Variance
A	0.481	0.585	0.606	0.631	0.861	$1.76 \cdot 10^{-3}$
B	0.139	0.369	0.394	0.415	0.519	$1.76 \cdot 10^{-3}$
δ	$5.85 \cdot 10^{-4}$	$1.62 \cdot 10^{-3}$	$3.53 \cdot 10^{-3}$	$1.16 \cdot 10^{-2}$	0.109	$3.56 \cdot 10^{-4}$
γ	$3.06 \cdot 10^{-4}$	$1.15 \cdot 10^{-3}$	$2.98 \cdot 10^{-3}$	$+.77 \cdot 10^{-3}$	1.36	$1.37 \cdot 10^{-2}$

Having separated the user population into types A and B users, α_A is simply the probability that a type A user has their answer accepted, this can be done similarly for type B but γ can be more directly estimated. The answer rate of type A users (β)can be estimated from the average age(ϵ_A) at which a question answered by a type A user is completed. We now have $\delta = \beta A$ and we can find δ using the average age of questions answered by type B users (ϵ_B). The relations used for estimating δ and γ can be found in Equations (2.18), this can be performed for each of over 100 Stack Exchange sites, the distributions of these estimates are summarized in table 2.2 and figure 2.2. We are also able to calculate the expected ratio between new and attempted questions, the distribution is shown in Figure 2.3.

$$\beta = (\alpha_A A \epsilon_A)^{-1} \tag{2.18}$$

$$\delta = \beta A \tag{2.19}$$

$$\gamma = \left(\epsilon_B - \frac{1}{(1-\alpha_A)\delta} \right)^{-1} \tag{2.20}$$

In addition to allowing for more meaningful simulation of the model, these parameter estimates are sufficient to tell us the relative abundances of Q_1 and Q_2 at equillibria. From the nullclines we know the ratio of new questions to attempted questions at the equillibria, this ratio is calculated for each community and shown in

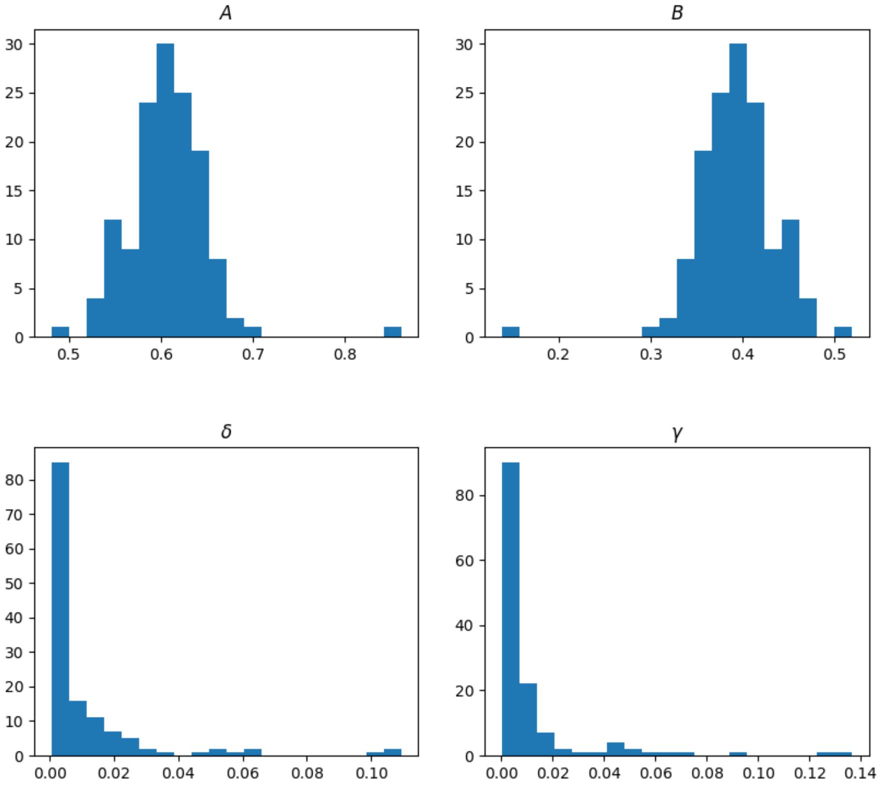


Figure 2.2: Distribution of Parameter Estimates for 136 Stack Exchange Sites

figure 2.3.

2.5 Case Results

One class of function that satisfies some of the conditions on attraction is Holling functional response. The generalized case of Holling type functional response is given in Equation (2.21). Typically, functional response is used for predation rate ($f(x)$) as a function of prey density (x). In the common interpretation, a represents the rate at which a predators encounter each prey, $\frac{1}{h}$ is the maximum predation rate, and n is used to capture the phenomenon of prey switching in which predators prey on rare

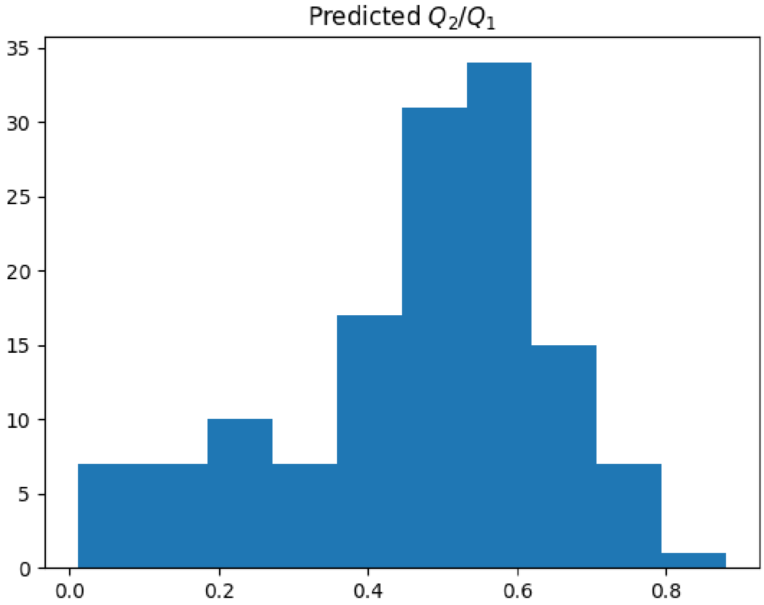


Figure 2.3: Distribution Ratio of Questions.

prey disproportionately less.

$$f\left(x\left(Q_1, Q_2\right)\right) = \frac{ax^n}{1 + ahx^n} \tag{2.21}$$

This response can be reinterpreted as the attractiveness $\left(f\left(x\left(Q_1, Q_2\right)\right)\right)$ of some attractive property $\left(x\left(Q_1, Q_2\right)\right)$. In this interpretation, a captures how quickly the attractiveness saturates, $\frac{1}{h}$ represents the maximal attractiveness, and n captures the capacity of users to go elsewhere to ask their questions. Equation (2.21) shows a Holling type III functional response, we use this format as it is the general case. Holling type I is the special case $f(x) = ax$ where $n = 1$ and $h = 0$. Holling type II is the case where $n = 1$ giving us $f(x) = \frac{ax^n}{1+ahx^n}$. For numerical analysis we use Holling type III as it is the most general case.

One of the properties of Holling type III functions is that for a domain of $[0, \infty)$ the range is $\left[0, \frac{1}{h}\right)$, this satisfies the nonnegative property of the attraction function. Another property of the attraction function is that there exists a zero equilibrium,

$x(Q_1, Q_2)$	meaning
Q_1	number of unanswered questions
Q_2	number of active answered questions
$Q_1 + Q_2$	number of active questions
$\alpha\delta Q_1 + \gamma Q_2$	answer acceptance rate

Table 2.3: Possible "Attractive" Traits of Online Communities.

this gives us the condition:

$$0 = f(x)\Big|_{Q_1=0, Q_2=0} \tag{2.22}$$

thus requiring $0 = x\big|_{Q_1=0,Q_2=0}$ when $a \neq 0$. The stability of this equilibrium is given by Table 2.1, to evaluate the stability we must differentiate, we obtain:

$$\frac{\partial f(x)}{\partial x} = \frac{anx^{n-1}}{(1+ahx^n)^2} \tag{2.23}$$

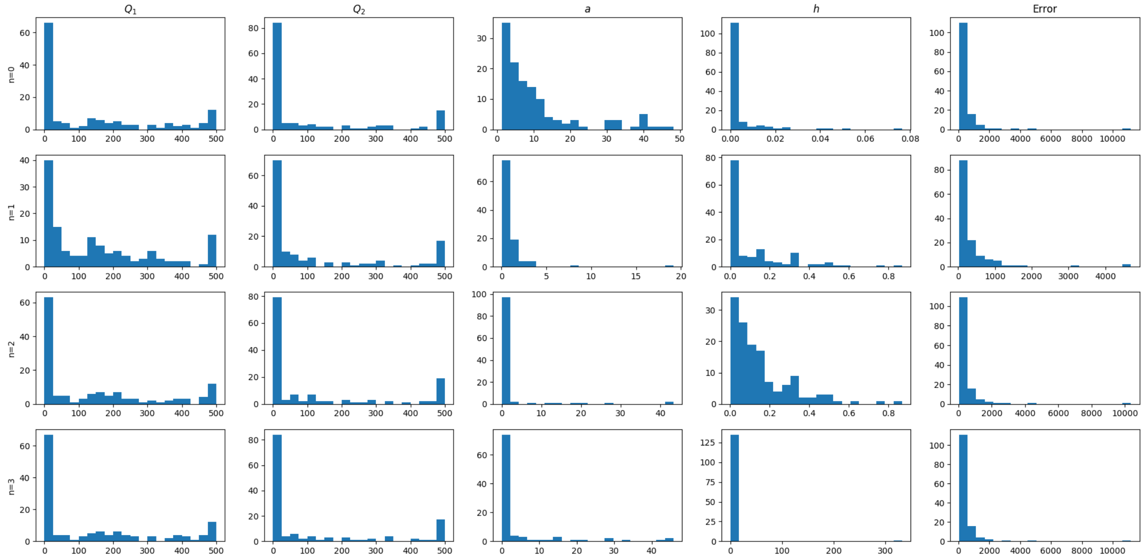
$$f_{Q_1} = \frac{anx^{n-1}}{(1+ahx^n)^2}\frac{\partial x}{\partial Q_1} \tag{2.24}$$

$$f_\star = \frac{anx^{n-1}}{(1+ahx^n)^2}\left(\frac{\partial x}{\partial Q_1} + \frac{(1-\alpha_A)\delta}{\gamma}\frac{\partial x}{\partial Q_2}\right) \tag{2.25}$$

Evaluating at $x = 0$ gives us $f_{Q_1} = 0$ and $f_\star = 0$ which indicates a stable equilibrium according to Table 2.1.We are unable to identify interior equilibria analytically and so cannot determine their stability.

At this point is becomes necessary to explore numerical results. As far as the author is aware, the points of intersection between a line and a Holling type III remains an open problem. We consider several possible "attractive traits" as input for the functional response, these are given in Table 2.5. For each trait, we parameterize the model to the Stack Exchange data, since we estimated δ and γ directly, we need fit for only three parameters: a, h, and n. This is accomplished through a least-squared gradient descent method fitted to answer acceptance data from Stack Exchange.

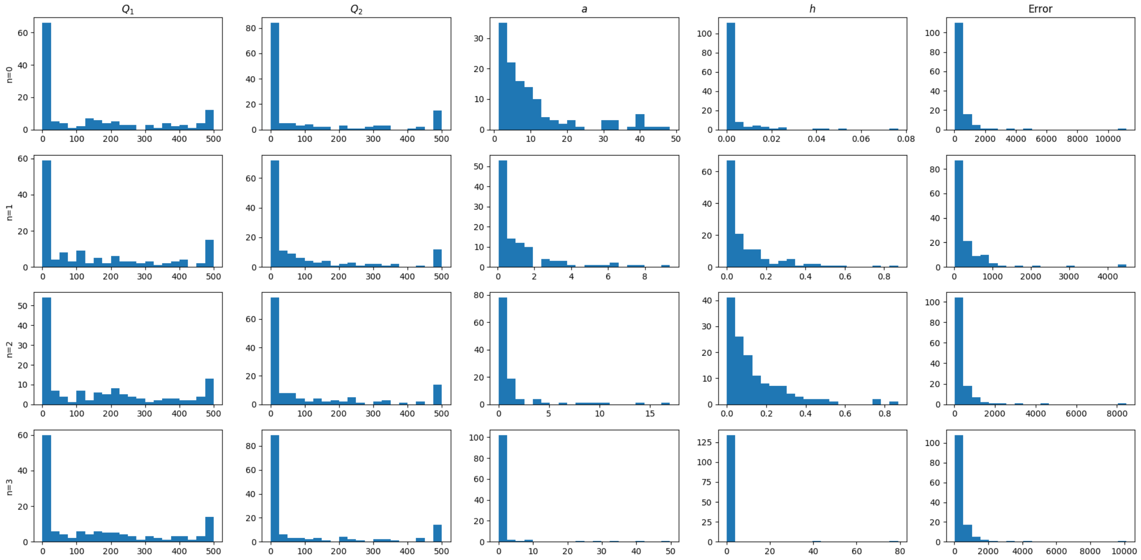
Figure 2.4: Parameterization of 136 Stack Exchange Communities Where the Attractiveness Function (2.21) is Evaluated With $x = Q_1$.



For the parameterization of the model we take α, δ, and γ from direct estimates as covered earlier. We now need only parameterize the attraction function $f(x)$. To identify the best fit there are some cases we need to go through, rather than fit n directly we consider four cases ($n = 0, 1, 2, 3$) allowing for a wide range of behaviors but constraining n to those values more commonly seen. Another set of cases is those on the input to the attraction function, these must be defined for the state space of the model thus precluding dividing by Q_1 or Q_2, the forms of x considered are listed in Table 2.5. For each case the model is fit to daily answering data for 136 communities. The results of these parameterizations can be seen in Figures 2.4 2.5 2.6 2.7.

Figure 2.4 provides distributions that come from fitting the model to the data when $x = Q_1$ that is the attractiveness of the community depends on the number on un-attempted questions. The figure should be read in rows, each row corresponds to a particular value of n which is a parameter of the general Holling type function (Equation (2.21)). The first four columns show distributions of the fitted parameters (see Figure 2.2 for distributions of estimated parameters). The first two columns are

Figure 2.5: Parameterization of 136 Stack Exchange Communities Where the Attractiveness Function (2.21) is Evaluated With $x = Q_2$.



the distributions of the fitted initial conditions and the third and fourth columns are the distributions of the Holling type function parameters. The final column shows the distribution of least-square residuals.

Figures 2.5 2.6 2.7 can be read in the same way as Figure 2.4. The difference between figures is in the trait taken to be attractive. The traits considered are un-attempted questions (Figure 2.4), attempted questions (Figure 2.5), total active questions (Figure 2.6), and the question answer rate (Figure 2.7). Each figure includes the attractiveness trait in its caption.

The parameterization results seem to suggest that $n = 1$ this can be observed from the distribution of residuals, regardless of what x is taken as the smallest maximal error is associated with $n = 1$. Looking at the mean and median of residuals, $x = \alpha\delta Q_1 + \gamma Q_2$ seems to perform best, though the choice of x does not seem crucial. Taking these choices of n and x we can complete analysis. There is not more than one interior equilibrium, this is derived starting with Equation (2.26). In Table 2.4 we have the parameter constraints on stability.

Figure 2.6: Parameterization of 136 Stack Exchange Communities Where the Attractiveness Function (2.21) is Evaluated With $x = Q_1 + Q_2$.

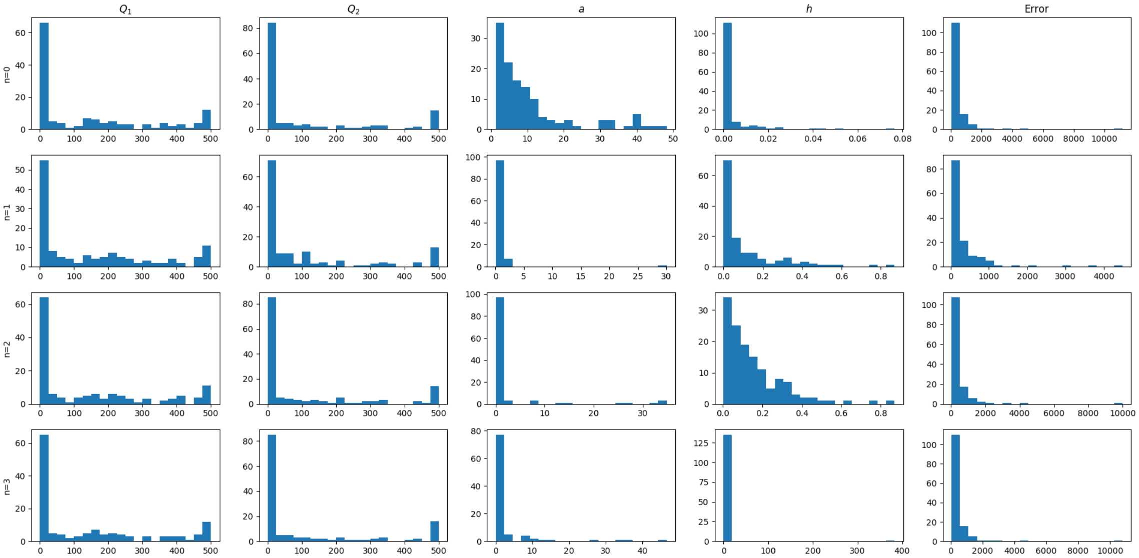
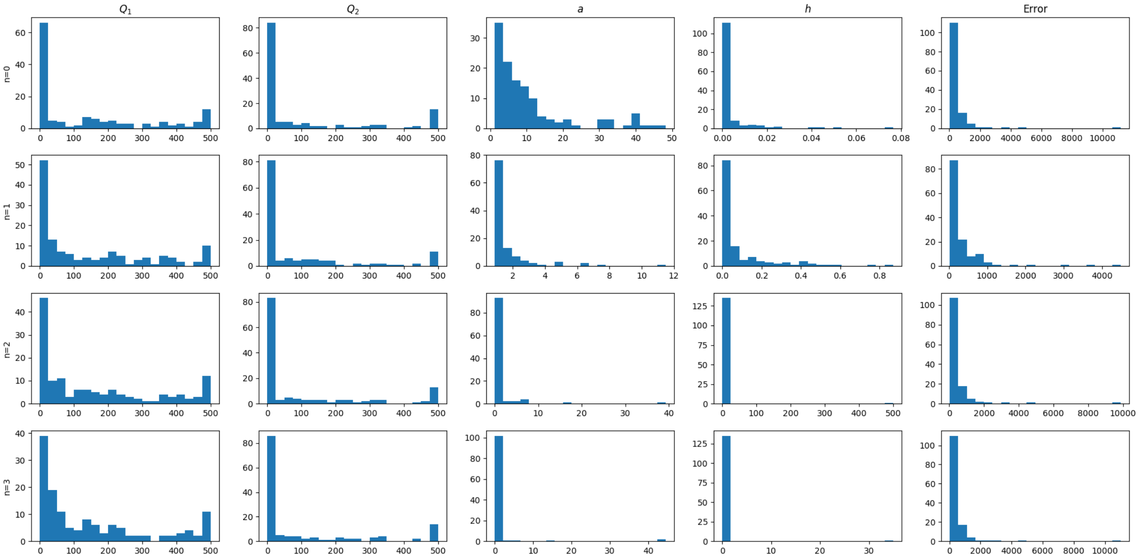


Figure 2.7: Parameterization of 136 Stack Exchange Communities Where the Attractiveness Function (2.21) is Evaluated With $x = \alpha\delta Q_1 + \gamma Q_2$.



Equilibrium	Existence	Stability
$(0,0)$	Always	$a < min\left(\frac{\delta+\gamma}{\alpha\delta}, \frac{1}{\gamma}\right)$
$(Q_1^\star, Q_2^\star)$	$a > 1$	$a > \gamma$

Table 2.4: Stability Conditions for Model Resulting from Holling Type III Attraction Function With $n = 1$.

$$0 = \frac{a\left(\alpha\delta Q_1^\star + \gamma Q_2^\star\right)}{1 + ah\left(\alpha\delta Q_1^\star + \gamma Q_2^\star\right)} - \delta Q_1^\star \tag{2.26}$$

$$0 = (1-\alpha)\delta Q_1^\star - \gamma Q_2^\star \tag{2.27}$$

$$Q_2^\star = \frac{(1-\alpha)\,\delta}{\gamma} Q_1^\star \tag{2.28}$$

$$a\left(\alpha\delta Q_1^\star + \gamma Q_2^\star\right) = \delta Q_1^\star\left(1 + ah\left(\alpha\delta Q_1^\star + \gamma Q_2^\star\right)\right) \tag{2.29}$$

$$a\left(\alpha\delta Q_1^\star + (1-\alpha)\,\delta Q_1^\star\right) = \delta Q_1^\star\left(1 + ah\left(\alpha\delta Q_1^\star + (1-\alpha)\,\delta Q_1^\star\right)\right) \tag{2.30}$$

$$a = 1 + ah\delta Q_1^\star \tag{2.31}$$

$$Q_1^\star = \frac{a-1}{ah\delta} \tag{2.32}$$

$$Q_2^\star = \frac{(a-1)\,(1-\alpha)}{ah\gamma} \tag{2.33}$$

From our analysis stability of the interior equilibrium comes from a, the rate at which users with questions encounter the forum. We now perform a sensitivity analysis to ascertain how communities might induce growth by shifting the stable equilibrium. For the purpose of this analysis we use parameters that are measurable in the data which requires decomposing δ and γ, we get $\delta = \beta A$ and $\gamma = \alpha_B \omega B$ where $A + B = 1$ and parameter meanings are as described in the model derivation. The results of the sensitivity analysis are given in Table 2.5. In general there is a balance between $Q_1^\star$ and $Q_2^\star$ that is determined by the components of δ and γ, increasing the accuracy of either user type (α_A and α_B) reduces the number of incorrectly answered questions ($Q_2^\star$). The number of questions on the site increases uniformly

$\frac{\partial Q_1^\star}{\partial A} = -\frac{1}{A^2\beta}\frac{a-1}{ah} < 0$	$\frac{\partial Q_2^\star}{\partial A} = \frac{1}{A^2\beta}\frac{a-1}{ah} > 0$
$\frac{\partial Q_1^\star}{\partial B} = \frac{1-\alpha_A}{B^2\omega\alpha_B}\frac{a-1}{ah} > 0$	$\frac{\partial Q_2^\star}{\partial B} = -\frac{1-\alpha_A}{B^2\omega\alpha_B}\frac{a-1}{ah} < 0$
$\frac{\partial Q_1^\star}{\partial \beta} = -\frac{1}{A\beta^2}\frac{a-1}{ah} < 0$	$\frac{\partial Q_2^\star}{\partial \beta} = 0$
$\frac{\partial Q_1^\star}{\partial \omega} = 0$	$\frac{\partial Q_2^\star}{\partial \omega} = -\frac{1-\alpha_A}{B\omega^2\alpha_B}\frac{a-1}{ah} < 0$
$\frac{\partial Q_1^\star}{\partial \alpha_A} = 0$	$\frac{\partial Q_2^\star}{\partial \alpha_A} = \frac{1}{B\omega\alpha_B}\frac{a-1}{ah} > 0$
$\frac{\partial Q_1^\star}{\partial \alpha_B} = 0$	$\frac{\partial Q_2^\star}{\partial \alpha_B} = -\frac{1-\alpha_A}{B\omega\alpha_B^2}\frac{a-1}{ah} < 0$
$\frac{\partial Q_1^\star}{\partial a} = \frac{1}{A\beta}\frac{1}{ah^2} > 0$	$\frac{\partial Q_2^\star}{\partial a} = \frac{1-\alpha_A}{B\omega\alpha_B}\frac{1}{ah^2} > 0$
$\frac{\partial Q_1^\star}{\partial h} = -\frac{1}{A\beta}\frac{a-1}{ah^2} < 0$	$\frac{\partial Q_2^\star}{\partial h} = -\frac{1-\alpha_A}{B\omega\alpha_B}\frac{a-1}{ah^2} < 0$

Table 2.5: Partial Derivatives of Interior Equilibrium with Respect to Each Parameter. Inequalities Hold Whenever the Equilibrium Exists.

when visibility increases (a) or when the subject interest increases ($\frac{1}{h}$).

Lastly, will we use numerical simulation to explore how the functional response parameter n and the inclusion of a residence time for questions will impact results. The parameter n represents the ability of predators to switch between prey species in the more common interpretation of Holling type functional response, the analog in our interpretation is that n captures that questions are not restricted to any subgroup. We will further include a parameter mu as a residence time of questions, this represents that questions can exit the active question population even if they are not complete, $\frac{1}{mu}$ is the average length of time before an unanswered question becomes inactive. For the purpose of simulation we will use Equations (2.34)(2.35) with parameter values given in Table 2.6 as a base case.

$$\dot{Q}_1 = \frac{a(\alpha\delta Q_1 + \gamma Q_2)^n}{1 + ah(\alpha\delta Q_1 + \gamma Q_2)^n} - (\delta + \mu)\, Q_1 \tag{2.34}$$

$$\dot{Q}_2 = (1-\alpha)\delta Q_1 - (\gamma + \mu) Q_2 \tag{2.35}$$

Figure 2.8 shows a series of phase-plane digarams for three values each of n and μ. The numerical results suggest that the number of active questions at the interior

Parameter	a	h	δ	γ	α	n	μ
Value	1.25	$2.75 \cdot 10^{-4}$	0.96	0.0888	0.628	1	0.01

Table 2.6: Parameter Values for Phase Portraits Used to Examine Effect of *mu* and *n*.

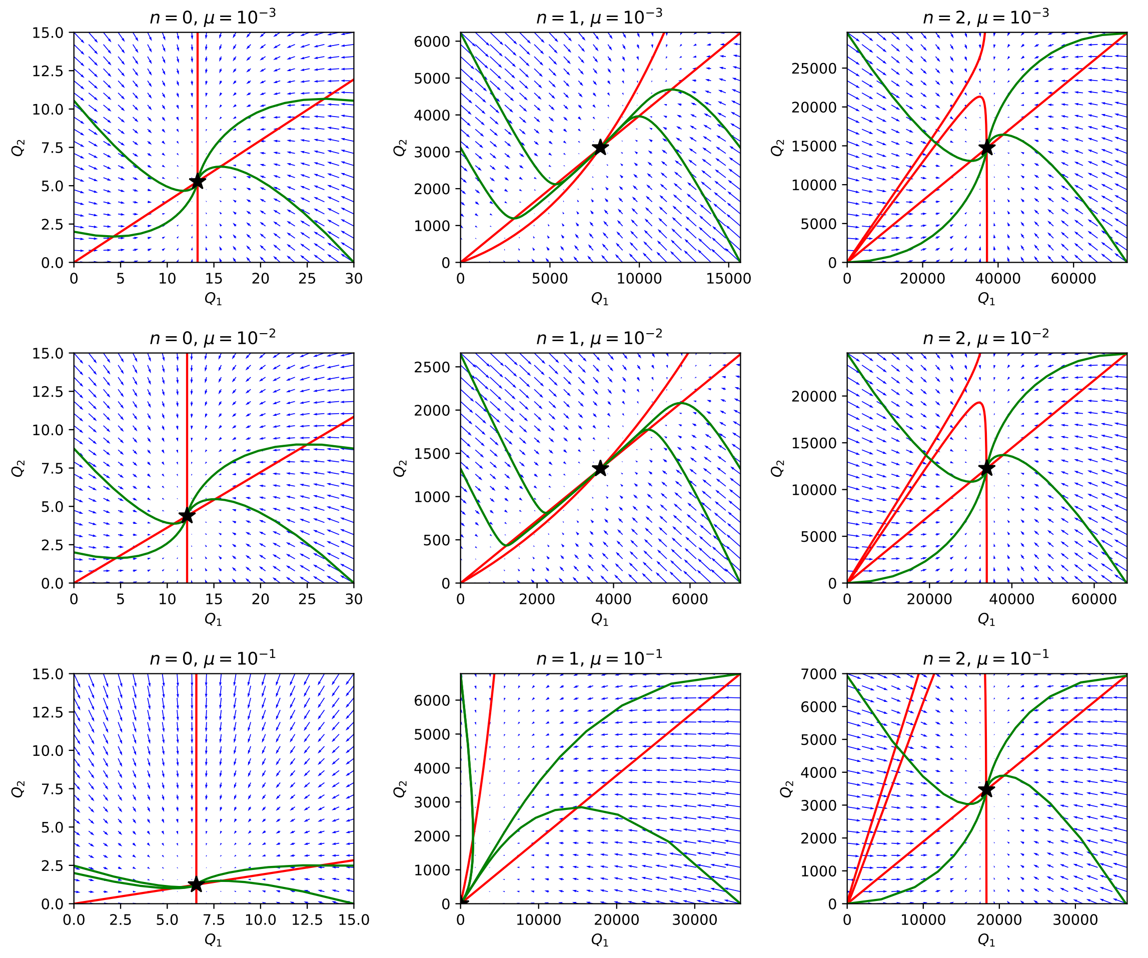


Figure 2.8: Phase Planes Showing Various Combinations of μ and n Parameters. Blue Arrows Show Parital Derivatives, Red Lines Show Nullclines, Green Lines Show Simulated Results, Black Stars Show Stable Equilibrium.

equiibrium increases with n and decreases with increases in μ. As is expected, decreasing residence time (increasing μ) results in the loss of the interior equilibrium, the threshold μ for loss of the interior equilibrium is dependent on n but is non-monotone.

2.6 Conclusions

In this chapter, we propose a model for the questions in a question and answer forum. From analysis of the model we identify some of the properties we expect from a function detailing how Q&A forums attract new questions. The constraints discovered allow us to identify a few plausible functional forms for modeling question attraction. Additionally, we were able to fit a standard functional response to data from the Stack Exchange network. The resultant model suggests that those parameters internal to the community control the balance between unanswered and attempted questions but that to change both simultaneously requires modification of the parameters of the attraction function.

Constraints on the dynamics of a simple Q&A population ecology model allow us to deduce some of the characteristics of the function representing inflow of new questions or the "attraction" of questions. One of the more limiting constraints identified is that the "attraction" function cannot be a linear combination of the state space, this precludes possibilities such as answer rate. However that does not mean that the answer rate isn't what determines a communities attractiveness. Though typically used as a predation term, Holling type functional response satisfies the constraints of the "attraction" function and can handle having answer rate as the attractiveness term. Analysis of this special case provides existence and stability conditions for an interior equilibrium i.e. a sustained community. Further, from a sensitivity analysis of the interior equilibrium we learn that those parameters describing the internal behavior of the model cannot simultaneously increase the equilibrium level of both compartments. This does not mean attractiveness cannot be optimized but it does mean that the details of the attractiveness trait must be known to select parameters to obtain an ideal balance.

In the more general context of online collaborations are results suggest that the attractiveness of a collaboration or community can be reasonably well approximated with knowledge of the communities activity. While different styles of collaboration are likely to have different forms of attraction, results do suggest that contributions beget contributions. Significantly, attractivness seems to depend on the current state of the system (as evidenced by excluding Q_3 from Equation (2.4) on). Thus, interventions should be possible relatively straight forward. However, the two parameters external to the community (a & h) can preclude a successful collaboration and it is not clear how an intervention could effectively modify these parameters.

Chapter 3

ONLINE COLLABORATION, COMPETING FOR ATTENTION

3.1 Introduction

In order to use online communities as a data source for understanding large scale collaboration, the nature of these communities must first be understood. Online communities are considered here to be social networks of exchange that are founded in and operate in online spaces. They can be seen as an environment shaped by users within these spaces, and can take on many varied forms. The tasks users complete further the goals of the community and are the resource of the environment. These tasks are provided by communities and consumed by users. There is not enough attention available to make all online communities vibrant and productive; this means that in order to better understand these spaces, it is prudent to model the user population rather than individual project. Communities harvest attention from the user population and use it to create knowledge products. Successful collaborations are those that are able to attract the attention of a population sufficient to allow that community to remain productive.

In this chapter, we describe an agent-based model of online communities. After constructing the model, we verify that the patterns that exist in real online communities exist in the simulated results. The types of patterns found in communities is summarized in the Chapter 1 Table 1.1.

3.2 Model Description

3.2.1 Introduction

In order to study the population dynamics of online communities, a simulation model is developed. In this model, users can move between communities while completing tasks that they encounter. The model includes: task generation, task allocation, user contribution, task completion, and user movement.

Three types of agents are considered in this model: tasks, users, and communities. Tasks are described by a task number, a list of past contributions, and a community to which that task belongs. Users are described by a skill number, their attention level, and the community they are participating in. Communities have a topic as well as user and task populations. The model is run in discrete time and the state variables are modified at each time step. Table 3.1 provides a complete list of variables and parameters.

The model used is a foraging model, tailored to suit online communities. Users are the foragers and they forage for tasks. However, users are heterogeneous, and a task that one user can complete might be outside the skill set of another user. Thus, tasks can be thought of as the resource users are foraging for, and users contributing to a task is what consumes that resource. Communities serve to divide the simulation environment into locations for foragers to move between.

3.2.2 Tasks

The task considered in this model is the identification of prime numbers. This task was chosen because it allows for heterogeneity in task "topic, as well as heterogeneity in agent skill. A user's skill is thought of as an integer. This integer is the number the user can divide by. A task is complete when a user successfully divides the task

Table 3.1: Summary of the Variables and Parameters of the Model. Parameter Values Are Chosen Ad Hoc and Are Provided Only for Completeness.

Variable	**Description**	**Value**
$NumComm$	Number of communities	500
$NumUser$	Number of users	500
$NumTask$	Number of tasks	500
$MinP$	Minimum task number	4
$MaxP$	Maximum task number	100
i	Community index	$i = 1, 2, ..., NumComm$
j	User index	$j = 1, 2, ..., NumUser$
k	Task index	$k = 1, 2, ..., NumTask$
s	User skill	$2 \leq s \leq \sqrt{MaxP}$
P_i	User population of community i	$0 \leq P_i \leq NumUser$
T_i	Topic of community i	$MinP \leq T_i \leq MaxP$
A_j	Attention level of user j	$A_j = 0, 1$
$S(j)$	Skill of user j	$2 \leq S(j) \leq \sqrt{MaxP}$
UC_j	Community that user j belongs to	$UC_j \in i$
TC_k	Community that task k belongs to	$UT_k \in i$
TN_k	Task number of task k	$MinP \leq TN_k \leq MaxP$
$CH_{k,s}$	Record of skill s being applied to task k	$CH_{k,s} = 0, 1$
rr	Community replacement rate	0.05
λ	Topic adjustment rate	0.1

(nonprime) or when every applicable skill has been applied to the task (prime). Tasks are the resource that users forage for. During each time step of the model tasks must be generated, allocated to communities, receive user contributions, and checked for completion.

In this model task generation occurs separately from task allocation. It is assumed that tasks are generated independently of the communities modeled. This could be justified by the observation that in online communities, those that ask questions, and those that answer them, are largely disjointed. Newly generated tasks then occur according to the parameters of the model rather than the dynamic variables. When a new task is generated it is chosen randomly to be any of the possible tasks considered (integers between $MinP$ and $MaxP$).

$$P\left(TN_{\{k|TC_k(t)=0\}}(t+1)=x\right)=\left\{\begin{array}{l|l} \frac{1}{1+MaxP-MinP} & x\in[MinP,MaxP] \\ 0 & x\notin[MinP,MaxP] \end{array}\right\}$$

Created tasks must be assigned to a community. Tasks are assigned based on two key factors: the size of the community (P_i) and the topic of the community (T_j). Community size is included to mimic the effect that popular communities receive more tasks then less popular communities. This is a form of preferential attachment, which is often included in models of online communities (Ozmen *et al.*, 2012; Kumar *et al.*, 2010). Communities are assumed to focus on specific topics, and tasks are allocated to communities with similar topics. The topic of a community is not static, but reflects the tasks that a community has recently had success with.

$$P\left(TC_{\{k|TC_k(t)=0\}}(t+1)=x\right)=\left\{\begin{array}{l|l} \frac{1}{\sum_i f(i,k)} & f(x,k)=1 \\ 0 & f(x,k)\neq 1 \end{array}\right\}$$

$$f(i,k) = \left\{ \begin{array}{c|l} 1 & TN_k(t+1) \in \left[T_i(t) - \frac{100P_i}{NumUser}, T_i(t) + \frac{100P_i}{NumUser}\right] \\ 0 & TN_k(t+1) \notin \left[T_i(t) - \frac{100P_i}{NumUser}, T_i(t) + \frac{100P_i}{NumUser}\right] \end{array} \right\}$$

Although the tasks that enter the simulation environment are independent of the communities, which community receives the task does depend on both the size and previous work of the community. Communities are considered to have a topic, bounded in the same domain as tasks, that reflects the specialty of the community. Additionally, larger communities are considered more able to meet a variety of challenges, expanding the range of tasks they can accept. The growth in range is assumed to be linear and such that if all users are in a single community, that community can accept all tasks. The new task generation and allocation occurs once per task completed in the previous round. This way, a population of $NumTask$ tasks is maintained in the simulation environment.

Once a task is assigned to a community, its users are able to contribute to it. In a given community, the users that contribute, and the tasks that get contributions, depend on the number of users with a particular skill and the number of tasks that require that skill. For each community and skill the number of users with that skill, and the number of tasks to which that skill is applicable, are counted. This can be represented as in:

$$NU(i,s) = \sum_{j|UC_j(t)=i,S(j)=s} 1$$

$$NT(i,s) = \sum_{k|UT_k(t)=i,TN_k(t+1)\leq s^2,CH_{k,s}(t)=0} 1.$$

Only unique contributions are counted, and users that are able to make a unique contribution are assumed to do so. These assumptions mean that for a given community and skill, either every user will contribute or every task will receive a contribution.

Of the larger population, a number of members equal to that of the smaller population is chosen to provide or receive the contributions. These assumptions provide probabilities for a contribution occurring for both tasks and users where the probability of a task receiving a contribution is determined by the relative abundance of users and vice versa.

$$P\left(CH_{\left\{k,s|CH_{k,s}(t)=0,TN+k\leq s^2\right\}}(t+1)=1\right)=$$
$$\left\{\begin{array}{l|l} min\left(1,\frac{NU(TC_k(t+1),s)}{NT(TC_k(t+1),s)}\right) & NT(TC_k(t+1),s)>0 \\ 0 & NT(TC_k(t+1),s)\leq 0 \end{array}\right\}$$

$$P\left(A_j(t+1)=1\right)=min\left(1,\frac{NT(TC_k(t+1),s)}{NU(TC_k(t+1),s)}\right)$$

A trivial consequence of these equations is that for each time-step, in each community, either every user with a particular skill will contribute or every task that requires that skill will have it applied.

Completed tasks are removed from the simulation at the end of every time-step, rather than as they are completed. This can be thought of as time used to verify that the task is complete or to accept a solution. The list of skills applied to a task is used to check if either a user has been able to identify the task as nonprime, or the community has identified a number as prime:

$$Nonprime(k)=\left\{\begin{array}{c|l} 1 & \{s|TN_k(t+1),CH_{k,s}(t+1)=1\}\neq\emptyset \\ 0 & \{s|TN_k(t+1),CH_{k,s}(t+1)=1\}=\emptyset \end{array}\right\}$$

$$Prime(k)=\left\{\begin{array}{c|l} 1 & 1+\sum_s CH_{k,s}>\sqrt{TN_k}(t+1) \\ 0 & 1+\sum_s CH_{k,s}\leq\sqrt{TN_k}(t+1) \end{array}\right\}$$

$$Complete(k)=max(NonPrime(k),Prime(k))$$

Nonprime numbers are detected by checking if any of the contributions to a task divides the task number, if at least one does the number cannot be prime. The identification of prime numbers requires that every skill that is applicable (less than or equal to the square root of the task) has been applied. It is possible for the function identifying primes to give false positives but, as no distinction is made, it does not affect results. Tasks that are completed need to be removed for the model to allow for new tasks. This is accomplished by the following equations:

$$CH_{\{k,s|Complete(k)=1\}}(t+1) = 0$$

$$TC_{\{k|Complete(k)=1\}}(t+1) = 0$$

The other effect of a completed task is the shift of topic in the communities, which is accomplished with a simple learning algorithm of the form:

$$T_i = (1-\lambda)\,T_i + \lambda TN_k$$

where λ is a topic adjustment rate, T_i is the topic of community i, and N_k is the task number of task k.

3.2.3 Users

The users of online communities are a heterogeneous population searching for something to hold their attention. Users differ in the skill they have for completing tasks, as well as in location. The skill of users is a fixed initial condition so users must move between communities to find tasks to contribute to. Users can leave a community either to join another community or to establish a new community.

Movement of users is driven by an algorithm based on win-stay, lose-shift, as well as preferential attachment (Nowak and Highfield, 2011; Ozmen *et al.*, 2012). Users are assumed to change communiteis every round in which they fail to contribute, and where they move to is based on the population of the other communities:

$$P\left(UC_{\{j|A_j=0\}}(t+1)=x\right)=\left\{\begin{array}{l|l} \frac{P_x}{\sum_{i\neq x}P_i} & x\neq UC_j(t) \\ 0 & x=UC_j(t) \end{array}\right\}$$

The other condition for user movement is the foundation of a new community, and it is assumed to increase linearly with the number of abandoned communities:

$$P\left(UC_j(t+1)=x\right)=\left\{\begin{array}{l|l} \frac{rr}{NumUser} & P_x=0 \\ 0 & P_x\neq 0 \end{array}\right\}$$

3.2.4 Initialization

At initialization users are assigned a skill between two and the square root of $MaxP$, which is the set of possible divisors of task numbers. Communities are assigned a topic from the same distribution as task numbers, and users are assigned to communities. Each of these assignments are assumed to be random.

3.2.5 Implementation

The model was implemented in both MATLAB and NetLogo. Code available via OpenABM at `https://www.openabm.org/model/5727/version/1/view`.

3.3 Results

The first characteristic verified is that the population of communities follows a power-law, which can be seen in Figure 3.1(a). The population of each community is taken as the number of users that contributed to it in the last time step. The method of measurement is meant to capture the active population of communities, which is the only feasible way of measuring population in real data. The distribution is stable in time (not pictured).

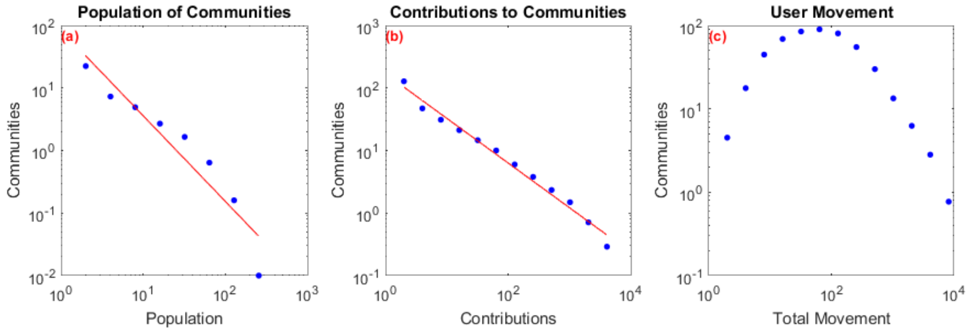


Figure 3.1: Key Distributions of Simulated Online Communities. Plot (a) Shows the Final Distribution of Population in Communities, as Well as the Power-law Approximation. Plot (B) Shows the Distribution of Contributions Throughout the Course of the Model, as Well as the Power-law of Best Fit. Plot (C) Shows the Distribution of User Movement to and from Communities over the Course of the Simulation. Results Are Averaged for 100 Runs with Parameters Equal to Those in Table 3.1.

Knowing that users are appropriately distributed throughout communities, next checked is that the contributions they make are similarly distributed. Figure 3.1(b) shows that contributions are distributed according to a power-law, or nearly a power-law. To generate this data, each community tracked the number of contributions it received. Significantly, communities with no users were considered abandoned and had their contribution count reset. This represents the creation of a new community.

Finally, user movement is measured by tallying the total movement in and out of each community. Movement is counted every time a user visits a community regardless of whether or not the user interacts with the community. As can be seen in Figure 3.1(c), this is clearly not power-law behavior. A possible explanation for this deficit is the difference in considered populations. Results that indicate browsing follows a power-law consider standard browsing behavior. The model was deliberately tailored to model the subset of users that contribute to productive communities. That this sub-population would have abnormal browsing behavior seems plausible, though it has not been confirmed.

Turning now to Stack Exchange, the model will be sampled in the same way as the data, in order to verify that it is capable of producing the same qualitative patterns. Figure 3.3 shows the result of the model analysis and should be compared to Figure 3.2, which shows the Stack Exchange analysis. Details of these analyses follow.

Analysis of the Stack Exchange data yielded four patterns that the model will similarly produce. These patterns include: community size, waiting time for a question to be answered, user movement, and neighbor connectivity. Figure 3.2 shows these patterns. How the data was sampled to generate those patterns, and what they mean, will be covered in the following paragraphs.

Figure 3.2 gives a comprehensive description on the population dynamics of the Stack Exchange system. Firstly, it can be observed that the distribution of community size, measured as the population of active users during the period of observation, is highly skewed. The largest community, stackoverflow.com (SO), has more than 2 million users whereas the smallest community has only hundreds of users. In fact, the distribution approximates Zipf's law in two orders of magnitude. Secondly, larger communities are more efficient in solving problems. Figure 3.2 shows that the average waiting time for accepted answers decreases with community size, indicating that larger communities are more efficient in solving problems. The mobility of users between communities satisfy a "gravity law", which predicts that the number of users moving between two communities is proportional to the product of the "mass" (population) of these two communities. The distribution of community size has a long-tail, i.e., there are only a few very large communities. These communities dominate the user mobility in the entire system. This was confirmed by data in the lower-right panel of Figure 3.2, displaying the assortativity of the community interaction network. To construct this network, communities are taken as nodes, and the movement of users between these community as edges. For each pair of nodes (communities), the number of moving users, in both directions, is aggregated to obtain weighted, undirected edges. After the network is constructed, edges are removed such that each node maintains its three strongest links, leaving the skeleton from a fully connected network. The constructed community interaction network is disassortative; i.e., high-degree nodes (which are large communities) tend to connect to low-degree nodes (which are small communities). This finding supports the assumption that the mobility of users in the system is dominated by a few large communities.

In short, Stack Exchange is dominated by a few large communities that 1) attract a majority of users in the system; 2) dominate the mobility of users between communities and 3) resolve most of the problems in a very short time. In other words, Stack Exchange activity mostly occurs rapidly in a few large communities, and many small communities rely on the giant communities to provide contributors, most of whom are very likely to return back to the largest communities after they have completed their task.

Since only active users are considered in the data, the model must be similarly restricted. When collecting the data, user activity is determined by posting behavior, that is, the population of a community is the number of users that have recently posted. This is replicated in the model by setting the population of a community, at a given time step, to the number of users that made a contribution to that community, during that time step. The population of communities appears to change proportionally with rank for more successful communities, however less successful communities seem to have proportionally larger variation. The model seems to predict more variation in less successful communities than the data would indicate. A possible explanation is Stack Exchange's policy on new communities. To add a community to Stack Exchange it must first prove its viability during a trial period, and no such mechanism exists in the model.

A question posted on Stack Exchange is considered answered only when the user who asked the question has accepted an answer. This can only occur after a minimum waiting period of 15 minutes. Both of these, as well as the discrete time of the model are some of the possible reasons for differences between the model and the data for this metric. The key pattern of waiting time decreasing with community size does hold, which seems to indicate that in both our model and Stack Exchange, the capacity of communities to complete tasks grows more rapidly than the number of tasks they attract.

User movement is tallied over the entire course of the simulation. This is then compared to the population of communities at the final time. This is taken, rather than a continual population count, for simplicity and because community populations stabilize quickly in the model. The result from this analysis is unsurprising and suggests that most user traffic is between large communities with very little between small communities.

Finally, the communities of both Stack Exchange and the model are dis-assortative; that is that the average degree of a community's neighbors is inversely proportional to the community's own degree. The construction of the network is crucial in obtaining this result and identical for the Stack Exchange and model data. Using the data from user movement, each community is linked to the three communities they have the most user traffic with, duplicate links are not counted, and ties are broken randomly. This results in a network with minimum degree 3, and a maximum degree of one less than the number of communities. On this network, the largest communities are interconnected, as they have high traffic in both directions; however, small communities are linked to large communities rather than being linked to each other. This results in a dis-assortative network where those most highly connected communities have the least connected neighbors.

3.4 Conclusions

In this chapter, a selection of the initial results of a population ecology oriented model of online communities are presented. In the following chapter we will perform a systematic sensitivity analysis to understand how assumptions on which the the underlying mechanisms are based affect the outcomes.

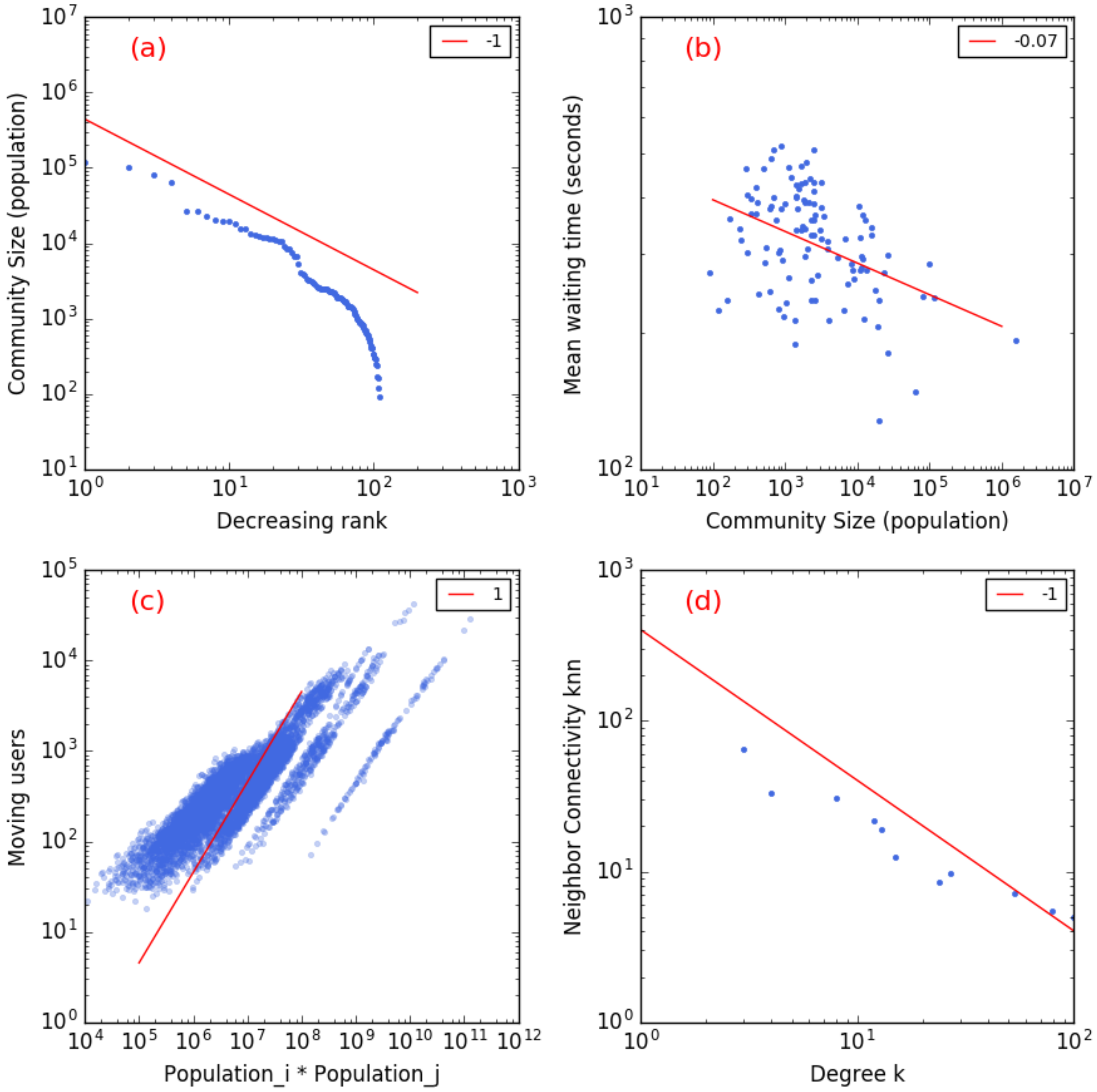


Figure 3.2: The Properties of the Communities of Stack Exchange. Starting From the Top Left Plot (a) Shows the Size of the Population of Communities Against Their Rank, This Is Compared to a Zipf Law in Red Plot (B) Shows the Mean Length of Time After a Question Is Asked Before the Community Provides an Answer Plot (C) Shows the the Undirected Movement of Users Against the Product of the Community Sizes. Plot (D) Shows the Mean Degree of Neighbors Against a Community's Own Degree, in the Network Skeleton That Preserves the Strongest Links.

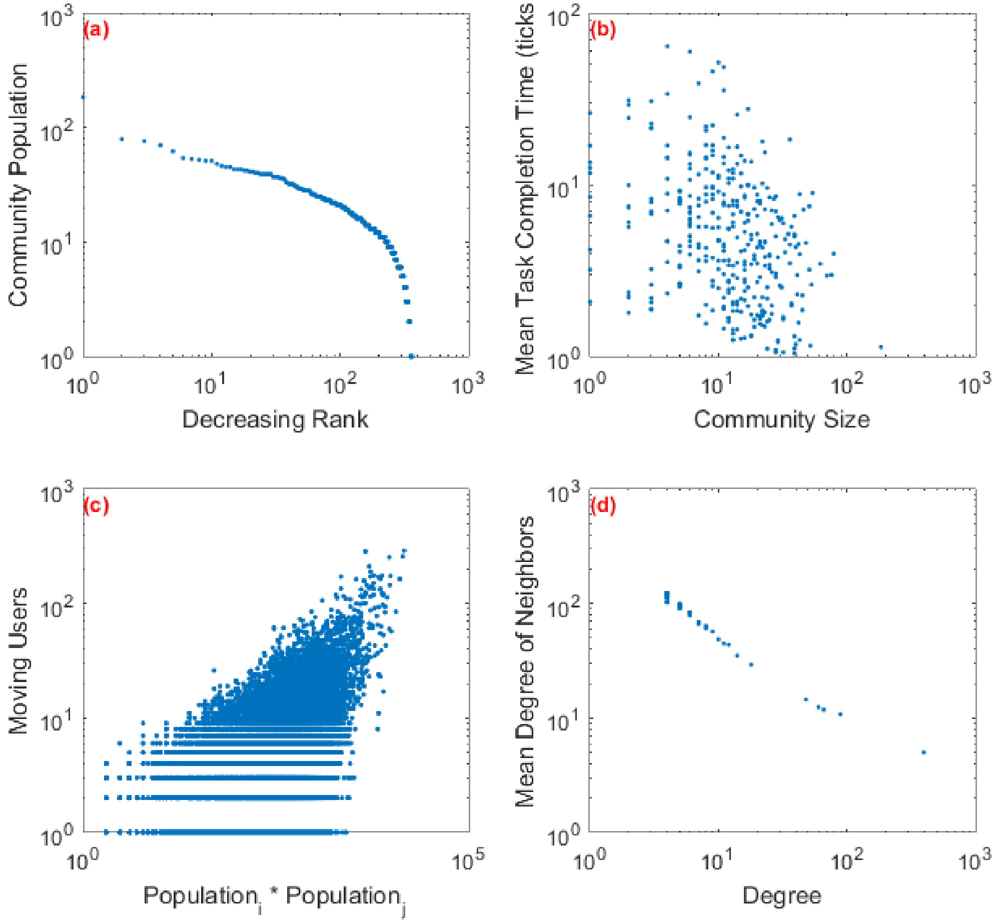


Figure 3.3: A Model of Online Communities. Starting from the Top Left Plot (a) Shows the Population of Communities Against Their Rank. Plot (B) Shows the Mean Length of Time after a Question Is Asked Before the Community Provides an Answer. Plot (C) Shows the Undirected Movement of Users Against the Product of the Community Sizes. Plot (D) Shows the Mean Degree of Neighbors Against a Community's Own Degree.

Chapter 4

EFFECTS OF BEHAVIOR ON ONLINE COLLABORATION

4.1 Introduction

In Chapter 3 we proposed an agent-based model of a multi-community online collaboration. This model was compared to data from the Stack Exchange Network and showed similar behaviors to the data. The mechanistic construction of the model in Chapter 3 as well as it's production of output similar to real world data serve to validate the model as a possible explanation for the behaviors of multi-community online collaborations. In this chapter we evaluate the sensitivity of various outcome metrics to the parameters in the model. This analysis provides hints at how we might impact the outcomes of real world collaborations as well as how those outcomes might be related.

4.2 Methods

4.2.1 Variance-Based Sensitivity Analysis

We perform a Variance-Based Sensitivity Analysis to assess the global sensitivity of the model to each parameter. The parameter space considered is explained in detail in Section 4.2.2 and the outcome metrics in Section 4.2.3.

Variance-Based Sensitivity Analysis serves to identify the impact of each combination of parameters on the outcome. Theoretically, this is accomplished by decomposing the system into a sum of functions of each combination of parameters as seen below.

$$\boldsymbol{Y} = f(\boldsymbol{X}) = f_0 + \sum_{i=1}^{n} f_i(\boldsymbol{X}_i) + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} f_{i,j}(\boldsymbol{X}_i, \boldsymbol{X}_j) + \cdots + f_{1,2,\ldots,n}(\boldsymbol{X}_1, \boldsymbol{X}_2, \cdots, \boldsymbol{X}_n)$$

Where $\boldsymbol{Y}$ is the output metric, $\boldsymbol{X}_i$ is parameter i, n is the number of parameters, f_0 is a constant, and $f_{\ldots}$ denotes a function with input parameters denoted by subscripts. The parameter space is assumed to be the unit hypercube with n dimensions, that is $0 \leq \boldsymbol{X}_i \leq 1, i = 1, 2, \ldots, n$. This parameter space can be achieved without loss of generality by rescaling the global parameter space to the unit hypercube. The decomposition is constrained to orthogonal functions so that the variance in output can be attributable to each combination of parameters. The orthogonality condition is written as:

$$\int_0^1 f_d(\boldsymbol{X}_k)\mathrm{d}\boldsymbol{X}_k = 0, \forall k \in d, d \subset \{1, 2, \ldots, n\}$$

From here it can be clearly seen that f_0 is the mean value of $\boldsymbol{Y}$ over the parameter space, this is demonstrated starting with Equation (4.1). Introducing the notation $\boldsymbol{X}_{\sim i} = \boldsymbol{X} \setminus \boldsymbol{X}_i$, we can also calculate the expectation on $\boldsymbol{Y}$ conditioned on a single parameter (Equation (4.2)) or on multiple parameters (Equation (4.3)).

$$\begin{aligned}\mathbf{E}(\boldsymbol{Y}) &= \int_0^1 \boldsymbol{Y}\mathrm{d}\boldsymbol{X} \qquad (4.1)\\ &= f_0 + \sum_{i=1}^{n} 0 + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} 0 + \cdots + 0\\ &= f_0\end{aligned}$$

$$\begin{aligned}\mathbf{E}(\boldsymbol{Y}|\boldsymbol{X}_i) &= \int_0^1 \boldsymbol{Y} \mathrm{d}\boldsymbol{X}_{\sim i} \\ &= f_0 + f_i(\boldsymbol{X}_i) + \sum_{k \neq i}^{n} 0 + \sum_{k=1}^{n-1} \sum_{j=k+1}^{n} 0 + \cdots + 0 \\ &= f_0 + f_i(\boldsymbol{X}_i)\end{aligned} \tag{4.2}$$

$$\begin{aligned}\mathbf{E}(\boldsymbol{Y}|\boldsymbol{X}_i, \boldsymbol{X}_j) &= \int_0^1 \boldsymbol{Y} \mathrm{d}\boldsymbol{X}_{\sim i,j} \\ &= f_0 + f_i(\boldsymbol{X}_i) + f_j(\boldsymbol{X}_j) + f_{i,j}(\boldsymbol{X}_i, \boldsymbol{X}_j) + \sum_{k \neq i}^{n} 0 + \sum_{k \neq i}^{n-1} \sum_{\neq j}^{n} 0 + \cdots + 0 \\ &= f_0 + f_i(\boldsymbol{X}_i) + f_j(\boldsymbol{X}_j) + f_{i,j}(\boldsymbol{X}_i, \boldsymbol{X}_j)\end{aligned} \tag{4.3}$$

As the name would indicate, Variance-Based Sensitivity Analysis is a sensitivity analysis based on variance. If we assume that $\boldsymbol{Y} = f(\boldsymbol{X})$ is square-integrable then the variance of $\boldsymbol{Y}$ can be written as in Equation (4.4). Notice that the orthogonality of function decomposition guarantees that the summation in Equation (4.4) is a summation of variances. We denote these variances as $V_{\dots}$ where the subscript matches that of the orthogonal function it is derived from, taken with Equation (4.4) we get a decomposition of the variance of $\boldsymbol{Y}$ into variances attributable to each combination of parameters as seen in Equation (4.5)

$$\begin{aligned}\mathrm{Var}(\boldsymbol{Y}) &= \int_0^1 f(\boldsymbol{X})^2 - f_0^2 \mathrm{d}\boldsymbol{X} \\ &= \sum_{k \subset \{1,2,\dots,n\}} \int_0^1 f_k(\boldsymbol{X}_k) \mathrm{d}\boldsymbol{X}_k\end{aligned} \tag{4.4}$$

$$\mathrm{Var}(\boldsymbol{Y}) = \sum_{i}^{n} V_i + \sum_{i<j}^{n} V_{i,j} + \cdots + V_{1,2,\dots,n} \tag{4.5}$$

Having decomposed the variance of $\boldsymbol{Y}$ we define the first order sensitivity index $S_i = \frac{V_i}{\mathrm{Var}(\boldsymbol{Y})}$. Notice that $1 \geq \sum_{i=1}^{n} S_i$, that is no more than 100% of the variance is attributable to first order sensitivity. The total-effect index measures the total variance attributable to a given parameter, we denote this $S_{\sim i} = \frac{\mathbf{E}(\mathrm{Var}(\boldsymbol{Y}|\boldsymbol{X}_{\sim i}))}{\mathrm{Var}(\boldsymbol{Y})} = 1 - \frac{\mathrm{Var}(\mathbf{E}(\mathrm{Y}|\mathrm{X}_{\sim \mathrm{i}}))}{\mathrm{Var}(\boldsymbol{Y})}$. Unlike with S_i, $1 \leq \sum_{i=1}^{n} S_{\sim i}$ this is due to the fact that all interactions are counted multiple times in the total-effect indexes. It is convenient to also define $V_{\sim i} = \mathbf{E}(\mathrm{Var}(\boldsymbol{Y}|\boldsymbol{X}_{\sim i}))$.

In practice we are not able to identify the orthogonal functions that contribute to $f(\boldsymbol{X})$, so we use a Monte Carlo approximation. We use the methods identified in (Saltelli *et al.*, 2010) to approximate V_i and $V_{\sim i}$. To perform this approximation we generate two independent sets of input parameters $\boldsymbol{A}$ and $\boldsymbol{B}$ where both are $N \times n$ matrices. We now define n matrices $\boldsymbol{A}_{\boldsymbol{B}}^{i}$ as matrix $\boldsymbol{A}$ with the ith column replaced by that of matrix $\boldsymbol{B}$. There are now $(n+2)N$ parameter combinations each of which is evaluated for each output metric. The Monte Carlo approximation of V_i and $V_{\sim i}$ for a given outcome metric $\boldsymbol{Y}$ are given in Equations (4.6) (4.7). From here the sensitivity indexes can be approximated as $S_i \approx \frac{V_i}{V_{\boldsymbol{Y}}}$ and $S_{\sim i} \approx \frac{V_{\sim i}}{V_{\boldsymbol{Y}}}$, where $V_{\boldsymbol{Y}}$ is the sample variance from all parameter sets.

$$V_i \approx \frac{1}{N} \sum_{j=1}^{N} f(\boldsymbol{B}_j) \left(f\left((\boldsymbol{A}_{\boldsymbol{B}}^{i})_j \right) - f(\boldsymbol{A}_j) \right) \tag{4.6}$$

$$V_{\sim i} \approx \frac{1}{2N} \sum_{j=1}^{N} \left(f\left((\boldsymbol{A}_{\boldsymbol{B}}^{i})_j \right) - f(\boldsymbol{A}_j) \right)^2 \tag{4.7}$$

4.2.2 Parameter Space

As stated in Section 4.2.1 the parameter space sampled for the sensitivity analysis must be transformed into a unit hypercube. To allow for a more thorough analysis the model is generalized from that of Chapter 3 and additional parameters were introduced. In this chapter we will only describe the changes made to the model, for a thorough description of the base model see Section 3.2. The parameters we consider can be broken into four categories: Populations, User Activity, User Movement, and Task Allocation. Each parameter is allowed to vary over a certain range given in Table 4.1 this range is then transformed to $[0, 1]$ for the application of variance based sensitivity analysis. All parameters are assumed to vary uniformly over their specified ranges.

There are three parameters governing the populations of the model. $NumUser$ is the number of users in the simulation, the user population is held constant over the course of each simulation. $NumComm$ is the number of communities considered in a given simulation, this serves as both the initial and maximum number of active communities for a given simulation. $NumTask$ is the maximum number of in-progress tasks for the simulation, at the beginning of each time-step new tasks are generated to bring the total to $NumTask$, tasks are removed if there is no community that can accept them or they are completed. Each of these three parameters vary from 100 to 2100 the lower bound was chosen to allow for meaningful outcome metrics and the upper bound was chosen to limit computation time.

Unlike in the model from Chapter 3, for the purpose of this sensitivity analysis users are no longer assumed to be always active. We introduce three parameters governing user activity: $UserAct$, $UserDeactC$, and $UserDeactN$. $UserAct$ is the probability that an inactive user will become active at the beginning of each time-step. $UserDeactC$ is the probability that an active user will become inactive following a successful contribution. $UserDeactN$ is the probability that an active user will become inactive following a failure to contribute. Users have the chance to become active at the beginning of each time-step and the chance to become inactive at the end of each time-step.

User movement can be broken into two steps, leaving a community and joining a community. There are two parameters governing when a user leaves there current community and three parameters governing their choice of community to join. In the base model all users employ a lose-shift strategy, losing in this case is failing to contribute, in the generalized case we introduce a parameter $Lose\text{–}shift$ as the probability that a user that fails to contribute will abandon their current community. We also introduce $Win\text{–}shift$ as the probability that a user that successfully contributed will leave their current community. Where a moving user goes is determined by the number of other users, available tasks, and productivity of each community. A communities attractiveness to moving users increases by $UAttract$ for each user in the community, $TAttract$ for each task in the community, and $ProdAttract$ for each task completed by the community. Each community's attractiveness is normalized by the total attractiveness and the result is the probability that a moving user will join that community.

The most parameterized step of the model is task allocation, these parameters can be further divided into three categories that govern the topic of communities, the maximum difference between a task number and a community topic, and how tasks are distributed to communities. The topic of a community is updated in two ways, each time a task is completed $T_{new} = T_{old}(1 - TopicComp) + N_{comp}TopicComp$ where T_{old} is the initial topic, T_{new} is the updated topic, N_{comp} is the task number and $TopicComp$ is a parameter. At the end of each time-step communities adjust their topic towards the mean topic of tasks that remain in the community $T_{new} = T_{old}(1 - TopicIncomp) + N_{incomp}TopicIncomp$ where N_{incomp} is the mean of task numbers in the community and $TopicIncomp$ is a parameter.

In identifying which communities could accept a given task we use four parameters. Each community has topic as well as a maximum difference between new tasks' number's and the community topic. The baseline maximum acceptable difference between a topic and a task number is $TaskDiff$. The acceptable difference increases with the community's productivity as well as user and task populations. The increase in maximum acceptable difference is $UTaskDiff$ per user, $TTaskDiff$ per task, and $ProdTaskDiff$ per completed task.

Finally, there are four parameters governing how a task is assigned to one of the communities that could accept it. Each task is assigned randomly to one of the communities that can accept it, the probability that a community receives the task is weighted according to parameters and the properties of the community. Communities with a topic more similar to the task number are more likely to receive the task, the weight on this is $TopicSim$. Having a larger user population makes a community more attractive to new tasks, the corresponding weight is $WeightUser$. A community's existing task population can attract additional tasks, the relative importance of this is $WeightTask$. Lastly, tasks might go to those communities that have completed the most tasks in the past, the degree to which prior success is counted is captured in $WeightProd$.

Parameter	Description	Range
NumUser	Number of users	100 to 2100
NumComm	Number of communities	100 to 2100
Num Task	Number of tasks	100 to 2100
UserAct	Probability that a user becomes active	0 to 1
UserDeactC	Probability that a user becomes inactive following a contribution	0 to 1
UserDeactN	Probability that a user becomes inactive following a non-contribution	0 to 1

Win-shift	Probability that a user moves after a contribution	0 to 1
Lose-shift	Probability that a user moves after a non-contribution	0 to 1
UAttract	Relative weight of user population when a user chooses a community to join	0 to 100
ProdAttract	Relative weight of productivity when a user chooses a community to join	0 to 100
TAttract	Relative weight of available tasks when a user chooses a community to join	0 to 100
TopicComp	Adjustment rate of community topic each time a task is completed	0 to 0.5
TopicIncomp	Adjustment rate to topic to incomplete tasks each timestep	0 to 1
TaskDiff	The baseline acceptable difference between a task and topic for task allocation	0 to 50
UTaskDiff	The increase to acceptable difference between task and topic per user	0 to $\frac{100}{NumUser}$
TTaskDiff	The increase to acceptable difference between task and topic per task	0 to $\frac{100}{NumTask}$
ProdTaskDiff	The increase to acceptable difference between task and topic per contribution	0 to $\frac{100}{NumTask}$
TopicSim	The relative weight of similarity between task and topic when allocating a task	0 to 1

WeightUser	The relative weight of number of users when allocating a task	0 to 1
WeightTask	The relative weight of number of tasks when allocating a task	0 to 1
WeightProd	The relative weight of total number of contributions when allocating a task	0 to 1

Table 4.1: Lists Parameters to Be Varied and the Range over Which They Vary.

4.2.3 Outcome Metrics

When sweeping the parameter space two types of data were collected, the productivity of each community, and the location of each user in each time-step. For the sensitivity analysis we focus on three areas of outcome metrics, the distribution of users across communities, the distribution of productivity across communities and how it relates to users, and the movement of users and how it relates to user population.

Since we track the location of each user over the course of the simulation we know the user population of each community at each time-step. The distribution of users stabilizes and we can measure a few outcomes from that distribution. We take the mean, median, mode of the final population distribution for each simulation. Additionally, in Chapter 4 we found that the population distribution can be approximated as a power-law so we find the power-function of best fit.

User population is not the only metric of interest for online collaborations. We track the number of contributions each community receives over the course of the model from this we can extract a few more outcome metrics. Again we can take the mean, median, and mode, now of community output. Further, it is expected that community output is related to community population so we identify the line of best fit relating population to output.

The biggest advantage to sampling from a model comes in the available movement data. The same basic metrics as used in population and output are interesting for user movement including the relation between movement and population, though this time the relation is a power function. Movement data also allows us to look at user retention in detail, to this end we find the line of best fit relating the number of visits a community receives to the number of repeated visits, this gives a retention rate for the system. We can also look at the distribution of retention rates across communities, how it relates to user population, and even how long it takes a user that leaves to return.

4.3 Results

The parameter space can be broken into four categories: populations, user activity, user movement, and task allocation. Each of these categories contains multiple parameters, the details can be found in Table 4.3. In this section we include the sensitivity analysis for some of the more interesting outcomes, the sensitivity analysis of all outcome metrics can be found in Appendix A.

The total output of the system or community of communities is an obvious metric of interest. Increases in total output mean that more knowledge is shared or produced. We find that the parameters that have a direct effect on total output fall into the categories of user activity and user movement, however less than half of the variation in outcomes is explained by first order effects. When considering the total-effect of all parameters, including interactions, all four categories have some effect. Both user

Category	Parameters
Population	NumUser, NumComm, NumTask
User Activity	UserAct, UserDeactC, UserDeactN
User Movement	Win-shift, Lose-shift, UAttract, ProdAttract, TAttract
Task Allocation	TopicComp, TopicIncomp, TaskDiff, UTaskDiff, TTaskDiff, ProdTaskDiff, TopicSim, WeightUser, WeightTask, WeightProd

Table 4.2: Categorization of Parameters by Which Mechanism They Influence.

activity and user movement parameters have interaction effects. The populations involved in the system have the largest total effect this is due to the fact that users and tasks are needed for tasks to be completed, the lack of first order effects is because increasing the number of available tasks does not result in more production unless there are sufficient users. The sensitivity of the total output is visualized in Figure 4.3.

The total amount of movement in the system is another metric of interest. Different tasks require different skills so moving users is required to prevent collaborations from stalling. It is no surprise that the parameters governing user movement have both the largest direct effect and total effect on the total user movement in the system. As with output, the population parameters had no direct effect but a substantial effect in their interactions. The sensitivity of the total movement is visualized in Figure 4.3.

For a collaboration to be productive there must by tasks that need doing and users to complete those tasks. We measure the number of tasks complete as well as tracking the movement of users. This allows us to look at the average number of tasks completed per visit to a community. We find that the distribution on this proportion is narrow as seen in Figure 4.3. Further we find through sensitivity analysis that first order effects of population, user activity, and user movement explain roughly half the

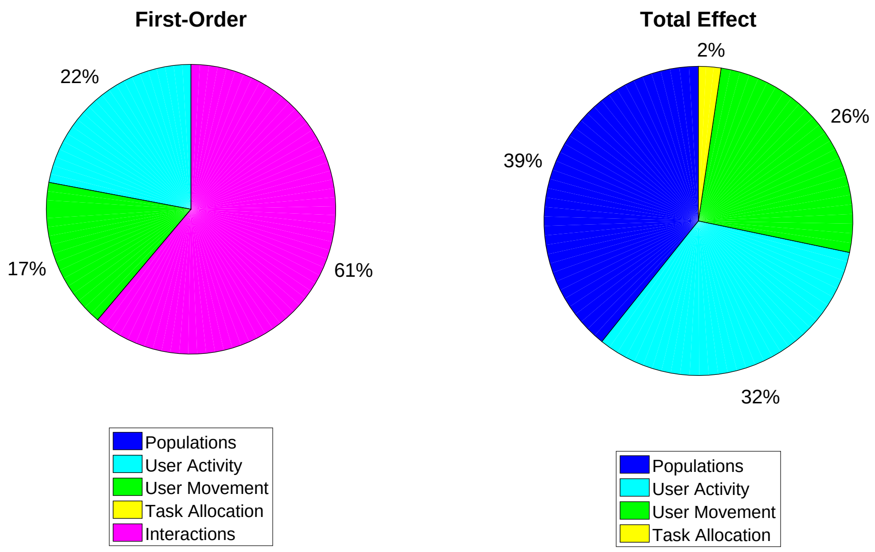


Figure 4.1: Pie Chart Showing the Effect of Various Categories of Parameters on the Total Output of All Communities in the System.

variation and that their total effects account for about a third of the variation each. The sensitivity of the output per visit is visualized in Figure 4.3

In addition to the total outcomes of the system we look at the outcomes of individual communities. This is sumarized in the median outcomes of the communities but a more thorough analysis can be found in Appendix **??**.

Variance in the median population of communities is largely explained by the population parameters. User movement has a small direct effect on the median population that increases substantially when interactions are taken into consideration. Task allocation has no direct effect on the distribution of users however the parameters do explain a portion of the variance in their interactions. The sensitivity of the median user population of communities is visualized in Figure 4.3

Variation in the median output of communities is dominated by variation in the population parameters. Less than half of the distribution of output is accounted for by

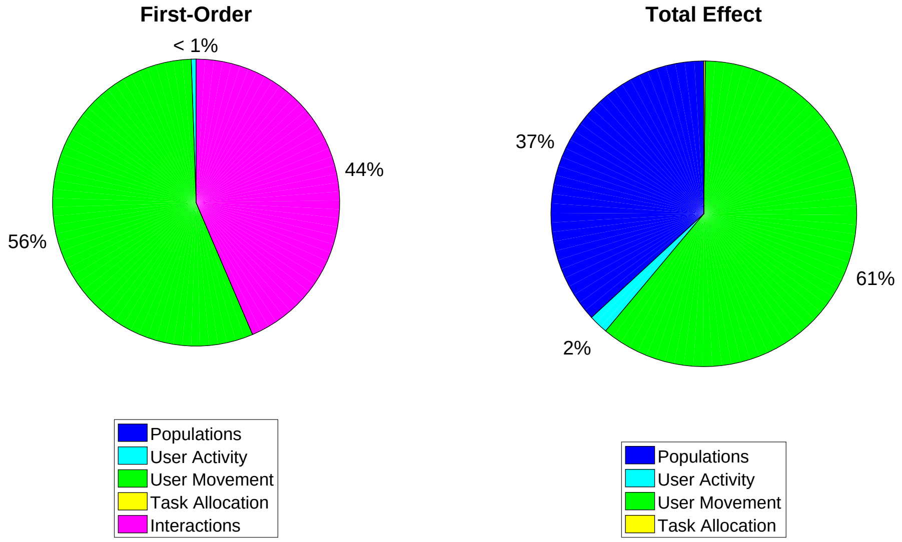


Figure 4.2: Pie Chart Showing the Effect of Various Categories of Parameters on the Total User Movement Through All Communities in the System.

first order effects. However more than three-quarters of the total-effect of parameters results from population parameters. The category that accounts for the least variance is that of user movement. The sensitivity of the median output of communities is visualized in Figure 4.3.

As we might expect from the low variance of the distribution in Figure 4.3, the sensitivity of the median movement through communities is similar to that of the median output. The most substantial differences are that more of the variance of movement is explained by first order effects, and that user movement parameters have a more significant effect when interactions are considered. The sensitivity of median movement through communities is visualized in Figure 4.3.

Since movement and output seem to vary together we look at an outcome that measures user retention in a way more in line with moving users. We define a community's return time by looking at the distribution of the time until a user that left

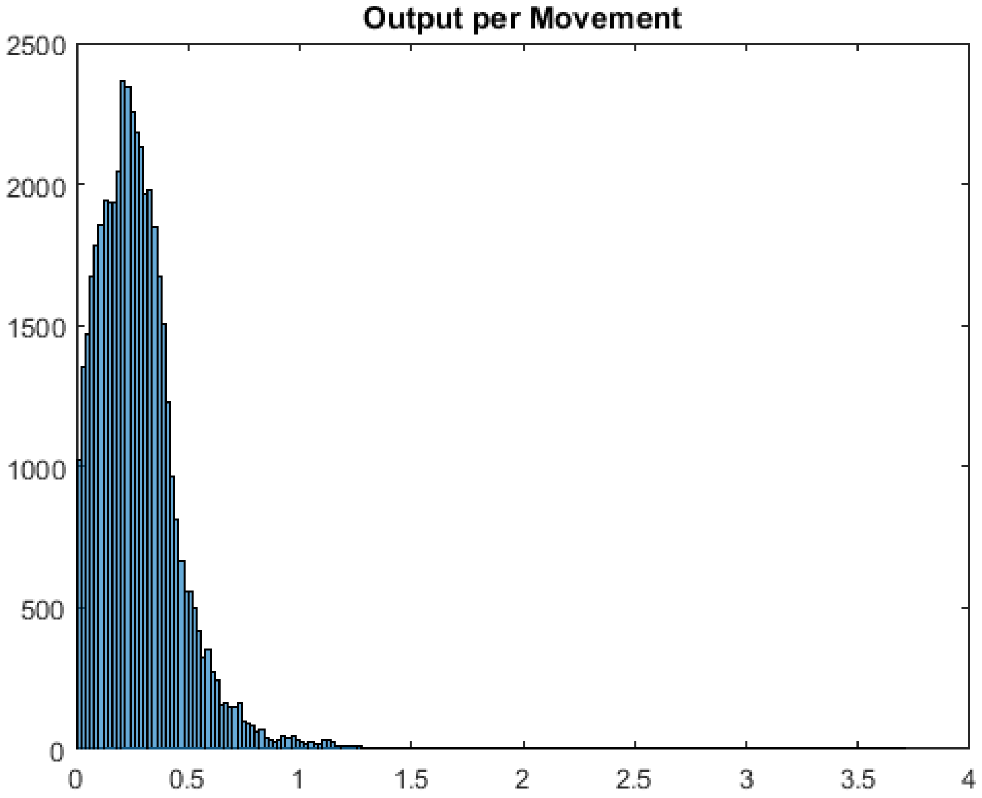


Figure 4.3: Histogram of the Ratio of User Movement to Community Output for 40,000 Simulations.

returns, and taking the median. A community's return time then is the time after leaving a community that half of users will have returned. The median return time tells how well communities are able to draw on previous users. The variance of return time is almost entirely explained by first order effects of the parameters. The population parameters have the largest first order effect, however both task allocation and user movement explain more variation when the interactions between parameters are considered. The sensitivity of median return time is visualized in Figure 4.3.

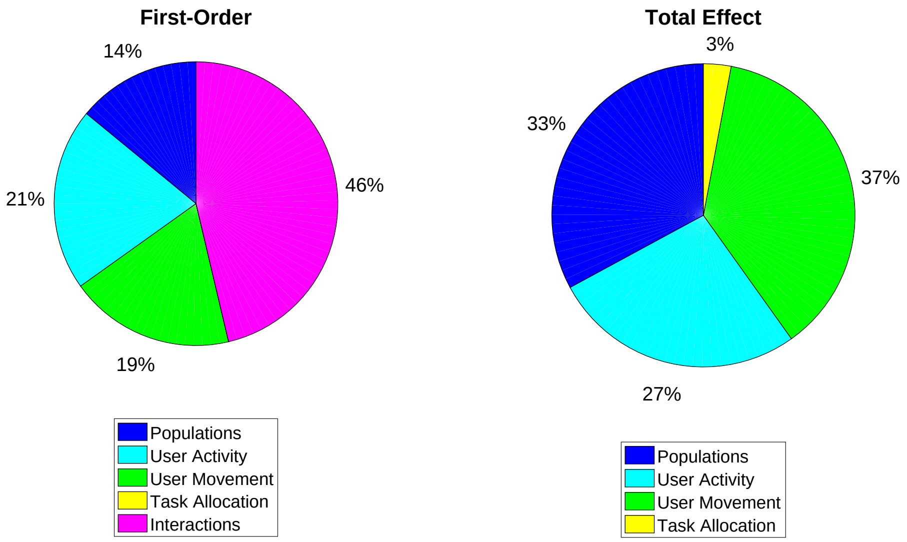


Figure 4.4: Pie Chart Showing the Effect of Various Categories of Parameters on the Ratio of Total Output to Total Movement.

4.4 Discussion

From our sensitivity analysis we are able to identify which categories of parameters are influential for various outcome metrics. Each category of parameter had a significant effect on at least one of our outcome metrics. However, not every outcome metric significantly depends on each category of parameter.

The most impactful category of the given parameters seems to be those describing populations. Parameters that descirbe the population of users, communities, and tasks have a first order effect on the majority the outcomes considered in this analysis. When including interactions, population parameters account for about one-third of the variation in outcomes totaled across communities, and well over half of the variation in median outcomes of communities for all but one case. User retention is where population parameters seem to be the least important. The is seen in observation of the median return time of users, which is heavily influenced by the first order effect

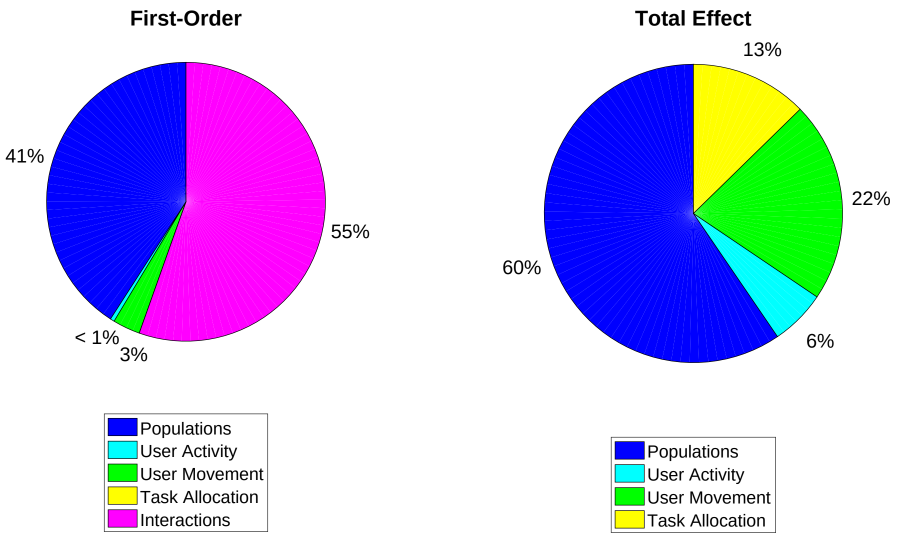


Figure 4.5: Pie Chart Showing the Effect of Various Categories of Parameters on the Median User Population of All Communities.

of population parameters but only 22% of the total effect is accounted for. For all of the metrics explored, population parameters have a significant effect and cannot be reasonably excluded.

The parameters governing user activity often have a negligible effect and can be safely excluded when interested in a selection of the outcomes considered. Though we always detect some first order effect of user activity, it never accounts for more than one-quarter of the total variance. Parameters governing user activity can be excluded if interested in the movement of users but not if interested in community output. User activity is largely irrelevant to the median population of communities, but plays a significant role in user retention.

How and when users move between communities effects all of the outcomes we considered. Unsurprisingly, the total movement in the system can be explained almost entirely by the parameters governing user movement and those controlling population

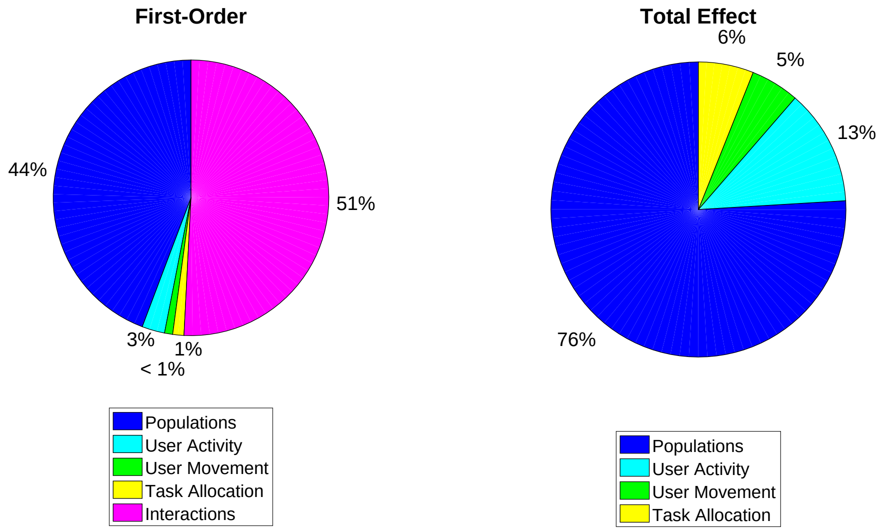


Figure 4.6: Pie Chart Showing the Effect of Various Categories of Parameters on the Median Output of All Communities.

size. Less intuitively, user movement parameters have significant effects (both first order and total) on both total output and the median output of communities. The parameters governing user movement effect all of the outcomes we considered, however that effect is least on the median output of communities.

The parameters governing task allocation are most numerous and least impactful. These parameters can be safely excluded from all of the totaled outcomes we considered, though they do have some impact on the outcomes of the median community. Despite being largely low impact, task allocation plays a significant role in explaining variation in user retention.

For the sake of legibility, parameters have largely been discussed in terms of categories. However, within those categories there are parameters that are more influential than others. The full results for each parameter can be found in Appendix A. Aside from task allocation, each category has a parameter that is most often most impactful

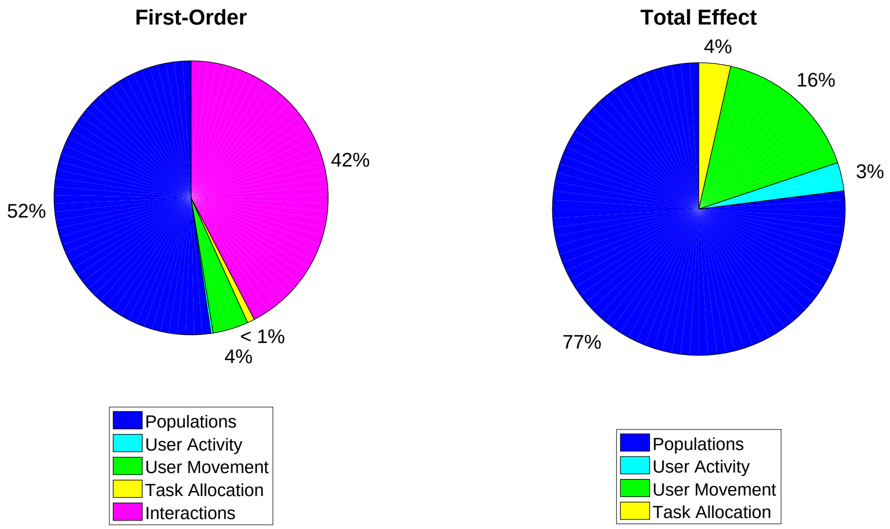


Figure 4.7: Pie Chart Showing the Effect of Various Categories of Parameters on the Median User Movement Through All Communities.

for the outcomes we've considered.

Of the population parameters we included, the number of communities seems to have the largest impact. Indeed this pattern holds for each outcome save the median return time, which is most influenced by the number of users. The dependence of distributional outcomes on the number of communities is unsurprising as it represents the sample size for most of our outcomes. It is interesting that the median return time of users to communities is more strongly influenced by the number of users than the number of communities as users. If users were to move uniformly at random the median return time could be calculated from the number of communities and the number of users would be irrelevant.

In those parameters governing user activity, the probability of an inactive user becoming active has the most significant effect on every outcome we tested. This could be because it effects every inactive user every time-step, whereas there are multiple

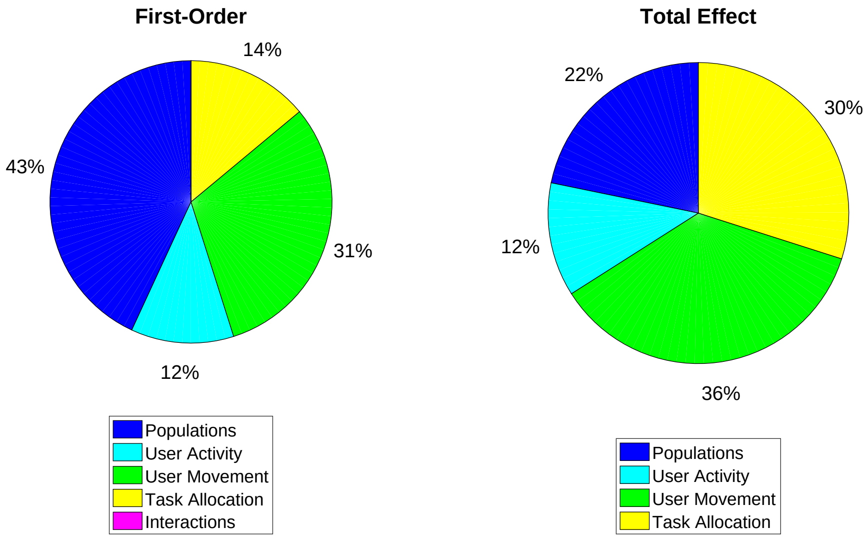


Figure 4.8: Pie Chart Showing the Effect of Various Categories of Parameters on the Median Return Time of Users to Communities They Have Previously Visited.

parameters governing the probability that an active user becomes inactive. Another possible cause is that user activation occurs immediately following user deactivation in the model flow such that if users become active with probability 1 then the other user activity parameters have no effect.

Perhaps the most interesting comparison between parameters is of two of those governing user movement. Win-stay, lose-shift is a successful foraging strategy that we adapted by assigning probabilities to win-stay and to lose-shift. Interestingly, most outcomes were most sensitive to the lose-shift parameter of those parameters governing user movement. Win-stay on the other hand rarely had any impact to speak of. This seems to suggest that it is more important to control what users do when they are bored than what they do when they are successful.

Chapter 5

CONCLUSIONS

In Chapter 2 we explore how an individual Q&A forum can influence its own success. Rather than define a model and use analysis to learn the dynamics of the system, in this chapter the analysis constrained the model. The flow of questions through a Q&A forum can be observed in the data, though how questions arrive at the forum is not so easily observed. In our model we include an “attraction” function that captures the arrival of new questions to the forum. From the dynamics of the real system we are able to constrain a few properties of this function. For instance, “attraction” must be zero when the forum is empty as the foundation of a community is a perturbation rather than a part of regular dynamics. Holling type functional response follows the criteria we identified for the “attraction” function and results in a reasonable fit for many Stack Exchange Q&A forums. Finally, a sensitivity analysis of the special case where new questions enter the forum according to a Holling type II function suggests that parameters internal to the community (ratio of answering strategies and their respective rates) control the balance between quick answers and accurate answers. Changes to external parameters (those governing the “attraction” function can uniformly increase or decrease the number of active questions on the forum.

In Chapter 3 we look at a community of communities rather than a single community. Using established mechanisms where possible, we construct an agent-based model of the users in a network of collaborations. Behaviors like preferential attachment, and win-stay, lose-shift, give rise to the same distributions of outcomes as can be observed in data from Stack Exchange. While this does not guarantee anything about the mechanisms truly at work, it is evidence that online collaborators may behave like foragers looking for something to occupy their attention and are susceptible to the availability heuristic. This information can be used to increase user retention and thereby grow collaborations.

Building on Chapter 3, Chapter 4 focuses on identifying which behavioral mechanisms drive which outcomes. Rather than find any one mechanism that can be safely excluded, we find that every mechanism plays an important role in at least one of the outcomes considered. This is primarily informative for the sake of future work, as results suggest that if you are interested in understanding how users move between communities, you need not concern yourself with how tasks are allocated. However, if you are interested in how long it takes departed users to return to a given community task allocation is more important than how often users are active.

Public goods are an important part of human society and as the world becomes more and more global, collaboration needs to scale up as well. Collaborating online is much less expensive than collaborating in person and as such is where future large scale collaborations are most likely to occur. The work in this dissertation explores what drives the outcomes of online collaborations at a couple of levels. Moving forward, facilitating constructive online interaction will only become increasingly valuable, and this is a first step towards understanding how to tailor online collaborative environments to produce the desired outcomes. Online communities can produce public goods, for example the Stack Exchange network produces answers to a variety of questions and access to those answers is non-excludable and non-rivalrous. However, a public good is not necessarily produced by altruistic contributors, this is also demonstrated in Stack Exchange. Within Stack Exchange is a point system setup to reward contributors, this allows contributors to be rewarded and thus fosters further contribution, but it does not tie a cost to accessing the good. The success of modeling contributors as foragers suggest that they are foraging for something. It is not necessary that contributors be driven by altruism, as the Stack Exchange network demonstrates it is possible to create public goods from the positive spillover of selfish interactions. Thus, when fostering online collaborations, there must be a reward for contribution to the public good that is not itself the public good.

These projects look at the how online collaborations can be encouraged on different scales. In Chapter 2 we look at a single isolated community and use known dynamics to infer possible mechanisms which in turn allow us to extrapolate the consequences of changes at the community level in the form of defined parameters. In Chapters 3 & 4 we take a wider perspective to look at how online collaboration can be encouraged more generally. Rather than look at the outcomes of a single community we look at those of a community of communities. The results of these chapters are not as useful for educing a given community towards are particular outcome, but they do tell us something about how to foster a productive online environment.

This work builds on prior work on by looking at massive data sets in the form of Stack Exchange sites. Unlike prior work modeling online communities we place an emphasis on where the problems come from, this is represented as community attractiveness in Chapter 2, and as task allocation in Chapters 3 and 4. Future research includes further refinement on what determines the attractiveness of a collaboration. Another avenue of further research is to identify the differences between those collaborations that fail to last and those communities that experience growing contributions.

We have learned that the conditions that lead to a communities persistence are, at least in part, external to the community. We also found that the conditions internal to the community served to balance activity between the categories that we considered. This may indicate a possible “community energy” that is controlled by external parameters. Quantifying the energy in communities would contribute a possible measure of success, but it may also help determine the viability of collaborations before establishment. This relates to the community of communities model explored in Chapters 3 and 4 in that communities have malleable topics that allow them to shift there focus to where they are more successful. We have also demonstrated that treating users as foragers and communities as the foraging environment produces is a viable perspective for modeling online communities.

There are several possible continuations of this research in large scale collaboration. First, though there are large scale, online collaborations, we have not verified that the results found also exist in offline collaborations. Second, a quantification of the "community energy" or available attention for communities which could be used to determine the viability of collaborations. Third, explore a dynamic model of user strategies so that the ratio between type A and type B users changes over time. Other possible extensions include making predictions with the model, testing interventions with the models, and generalizing the models to collaborations rather than online Q&A forums.

REFERENCES

Albert, R. and A.-L. Barabási, "Statistical mechanics of complex networks", Reviews of modern physics **74**, 1, 47 (2002).

Bade, R. and M. Parkin, *Foundations of Microeconomics* (Pearson Higher Ed, 2012).

Barabási, A.-L. and R. Albert, "Emergence of scaling in random networks", science **286**, 5439, 509–512 (1999).

Boyd, R., H. Gintis, S. Bowles and P. J. Richerson, "The evolution of altruistic punishment", Proceedings of the National Academy of Sciences **100**, 6, 3531–3535 (2003).

Broder, A., R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins and J. Wiener, "Graph structure in the web", Computer networks **33**, 1, 309–320 (2000).

Fu, F., L. Liu and L. Wang, "Empirical analysis of online social networks in the age of web 2.0", Physica A: Statistical Mechanics and its Applications **387**, 2, 675–684 (2008).

Huberman, B. A. and L. A. Adamic, "Internet: Growth dynamics of the world-wide web", Nature **401**, 6749, 131–131 (1999).

Huberman, B. A., P. L. Pirolli, J. E. Pitkow and R. M. Lukose, "Strong regularities in world wide web surfing", Science **280**, 5360, 95–97 (1998).

Huberman, B. A., D. M. Romero and F. Wu, "Crowdsourcing, attention and productivity", Journal of Information Science (2009).

Janssen, M. A., R. Holahan, A. Lee and E. Ostrom, "Lab experiments for the study of social-ecological systems", Science **328**, 5978, 613–617 (2010).

Janssen, M. A., M. Manning and O. Udiani, "The effect of social preferences on the evolution of cooperation in public good games", Advances in Complex Systems **17**, 03n04, 1450015 (2014).

Kittur, A. and R. E. Kraut, "Beyond wikipedia: Coordination and conflict in online production groups", in "Proceedings of the 2010 ACM conference on Computer supported cooperative work", pp. 215–224 (ACM, 2010).

Kumar, R., J. Novak and A. Tomkins, "Structure and evolution of online social networks", in "Link mining: models, algorithms, and applications", pp. 337–357 (Springer, 2010).

Nowak, M. and R. Highfield, *SuperCooperators: Altruism, Evolution, and Why We Need Each Other to Succeed* (Free Press, New York, 2011).

Ostrom, E., J. Walker and R. Gardner, "Covenants with and without a sword: Self-governance is possible.", American political science Review **86**, 02, 404–417 (1992).

Ozmen, O., J. Smith, L. Yilmaz and A. E. Smith, "A complex adaptive model of information foraging and preferential attachment dynamics in global participatory science", in "Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), 2012 IEEE International Multi-Disciplinary Conference on", pp. 65–72 (IEEE, 2012).

Pastor-Satorras, R., A. Vázquez and A. Vespignani, "Dynamical and correlation properties of the internet", Physical review letters **87**, 25, 258701 (2001).

Radtke, N. P., *FLOSSSim: Understanding the Free/Libre Open Source Software (FLOSS) Development Process through Agent-Based Modeling*, Ph.D. thesis, Citeseer (2011).

Saltelli, A., P. Annoni, I. Azzini, F. Campolongo, M. Ratto and S. Tarantola, "Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index", Computer Physics Communications **181**, 2, 259–270 (2010).

Wilkinson, D. M., "Strong regularities in online peer production", in "Proceedings of the 9th ACM conference on Electronic commerce", pp. 302–309 (ACM, 2008).

Wu, F., D. M. Wilkinson and B. A. Huberman, "Feedback loops of attention in peer production", in "International Conference on Computational Science and Engineering, 2009. CSE'09.", vol. 4, pp. 409–415 (IEEE, 2009).

Wu, L., J. A. Baggio and M. A. Janssen, "The role of diverse strategies in sustainable knowledge production", PloS one **11**, 3, e0149151 (2016).

Yasseri, T. and J. Kertész, "Value production in a collaborative environment", Journal of Statistical Physics **151**, 3-4, 414–439 (2013).

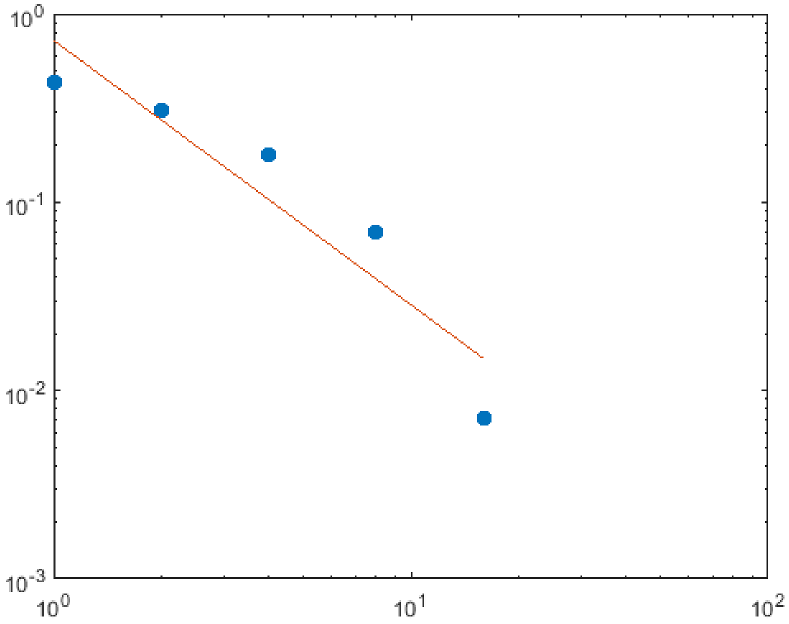
APPENDIX A

SENSITIVITY ANALYSIS

We will now go through our outcome metrics for a single simulation then look at the results of a Variance-Based Sensitivity Analysis of those metrics.

We start by looking at the distribution of users through communities. This can be roughly approximated as a power-law so we identify the power-law of best fit, this results in two metrics. Additionally we measure the mean, median, and mode population of communities. An example power-law fit of community population is shown in FigureA.1. We now have 5 metrics describing the distribution of users in communities.

Figure A.1: Community Population Distribution

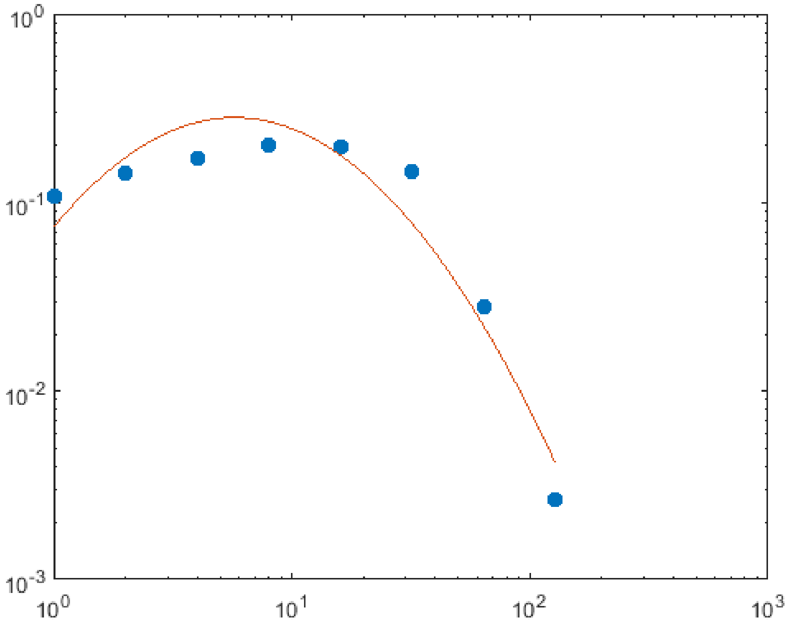


The next distribution of interest is the distribution of contributions, the contributions each community receives is totaled over the course of the simulation giving a distribution of productivity. Again we take the mean, median, and mode of this distribution giving us three metrics. Looking at the distribution of output in loglog coordinates we get a concave distribution and so we identify the function of the form $f(x) = 2^{ax^2+bx+c}$ that best fits the data, this gives us three more metrics bringing the total to 11. An example of this fit can be seen in FigureA.2.

Looking now at how users move between communities we total the number of entries and exits for each community. Like with the community output, the distribution of movement is concave in loglog coordinates so we fit it in the same way and generate three metrics, an example of this fit can be seen in FigureA.3. Again, we take the mean, median, and mode of the distribution.

When a user leaves a community, their departure is not permanent. We construct a distribution of the time until return for each community and identify the median. This measure tells us how long a community has to wait for half of the users that left to have returned. The distribution of these median return times is a metric of interest

Figure A.2: Community Output Distribution



and so we take the mean, median, and mode. We also fit this distribution with a function of the form $f(x) = 2^{ax^2+bx+c}$ which gives us an additional three outcome metrics, an example is shown in FigureA.4.

Without looking at how the outcomes of communities are inter-related we identify 23 outcome metrics. Next, we look at how metrics relate to each-other. To do this look at how community output, user movement, and median return time changes with community population, community output, and user movement. This results in six comparisons as reciprocal comparisons are not considered. For each comparison we use logarithmic binning of the independent variable and the logarithmic mean of the dependent variable inside that bin. From these binned data points we construct a linear fit for each comparison, this results in an additional 12 outcome metrics. Examples of each of these comparisons are shown in FiguresA.5A.6A.7A.8A.9A.10.

We now have 21 input parameters and 35 output metrics. We run VBSA for a total of nearly half a million simulations. The results are summarized in parts: FigureA.11 shows the distributional characteristics of the user population, FigureA.12 shows the distributional characteristics of community output, FigureA.13 shows the distributional characteristics of user movement, FigureA.14 shows the distributional characteristics of median return time, FigureA.15 shows measures of community output per visit by user, FigureA.16 shows how user population relates to other outcomes, FigureA.17 shows how community output relates to other outcomes and, FigureA.18 shows how user movement relates to other outcomes.

Figure A.3: Community User Flow Distribution

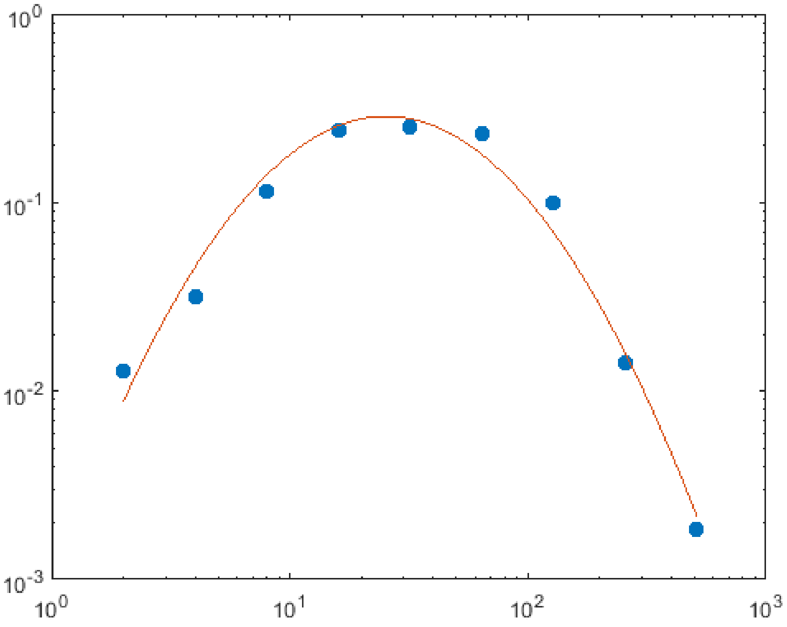


Figure A.4: Median Return Time Distribution

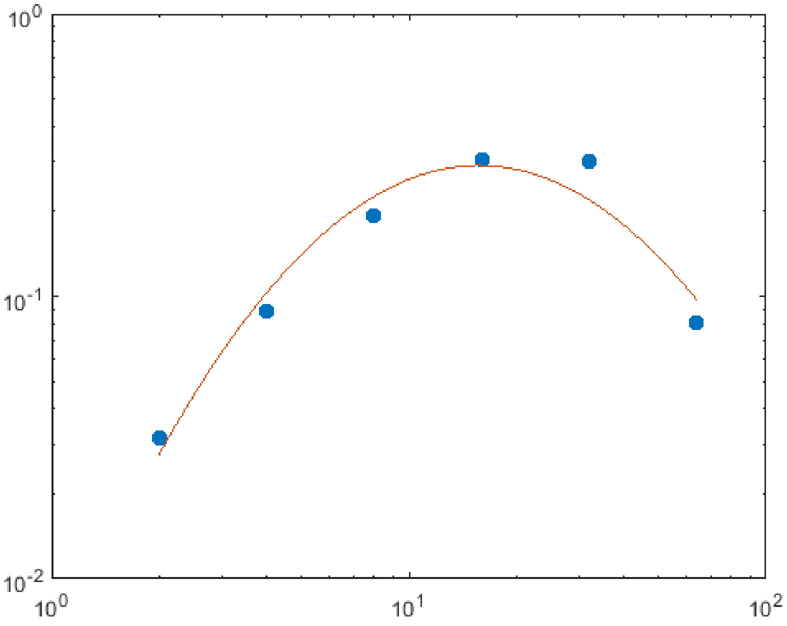


Figure A.5: Community Output vs Population

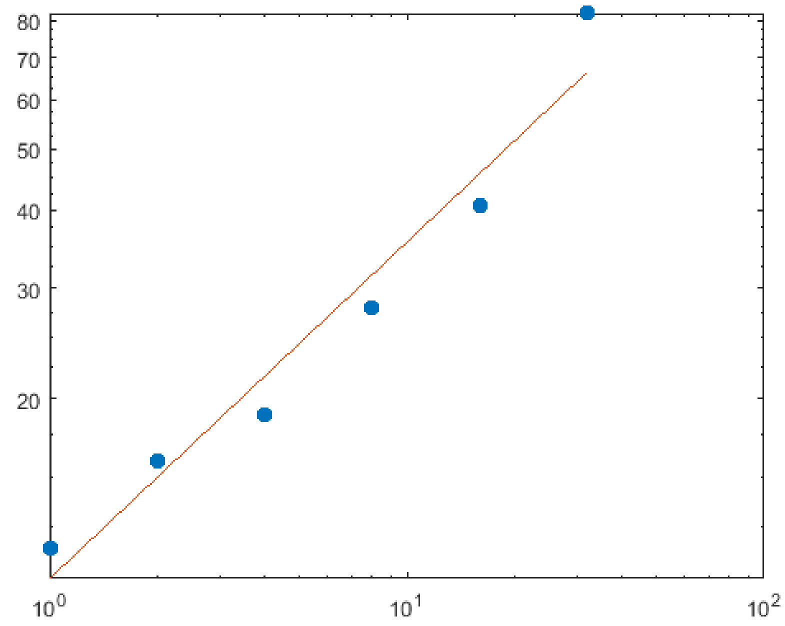


Figure A.6: Community User Movement vs Population

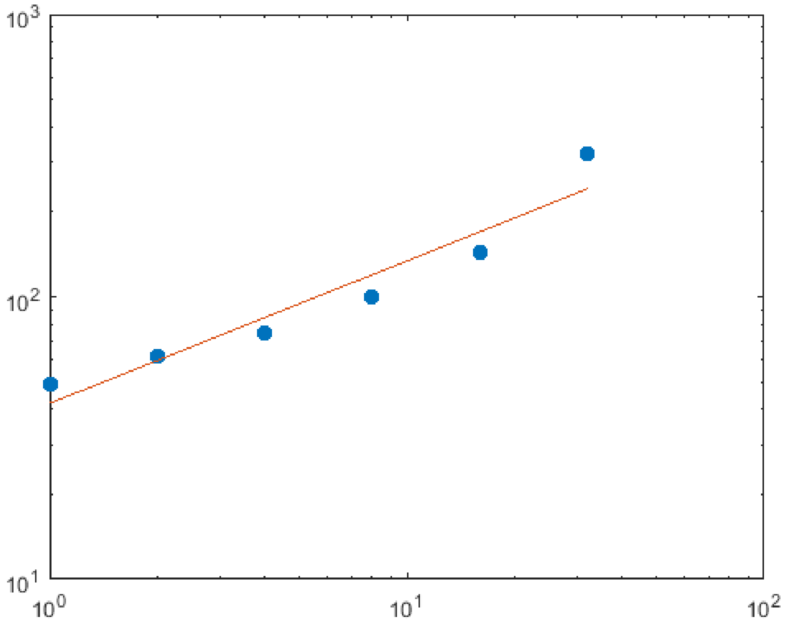


Figure A.7: Median Return Time vs Population

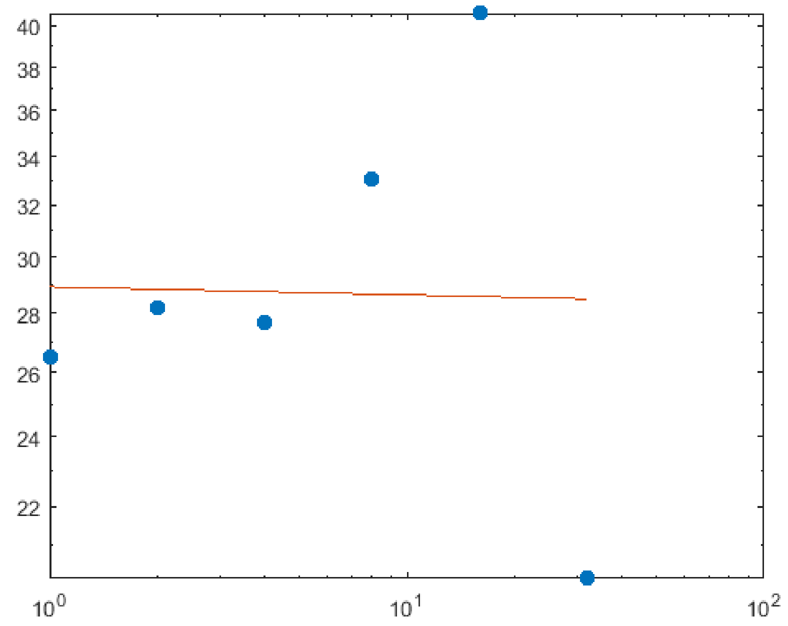


Figure A.8: Community User Movement vs Output

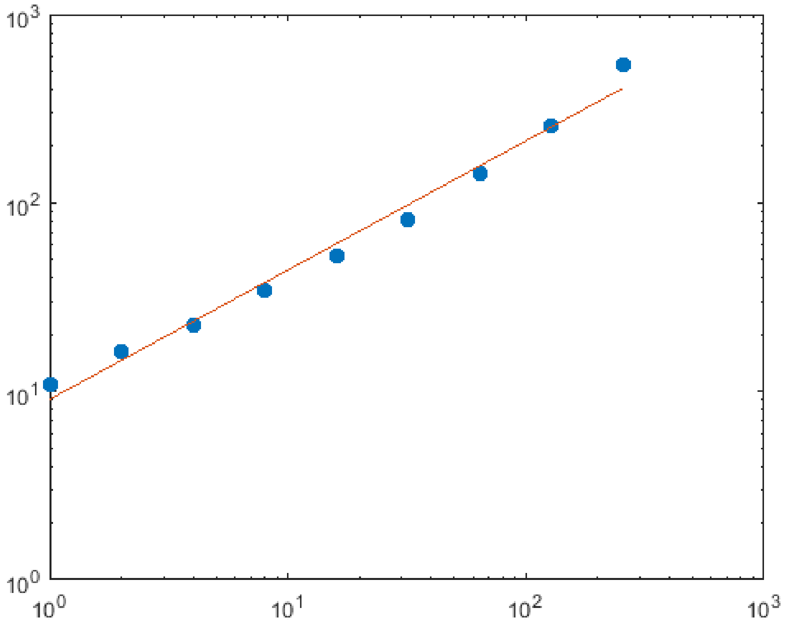


Figure A.9: Median Return Time vs Output

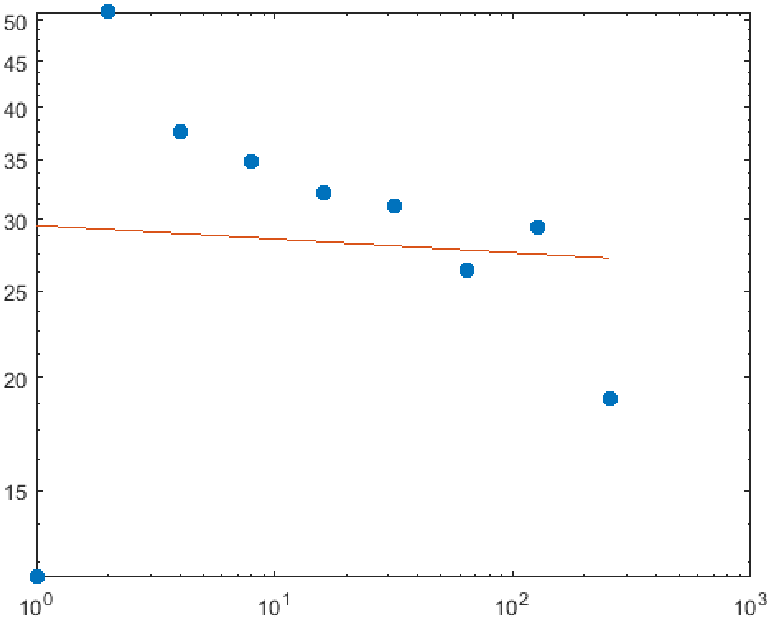


Figure A.10: Median Return Time vs User Movement

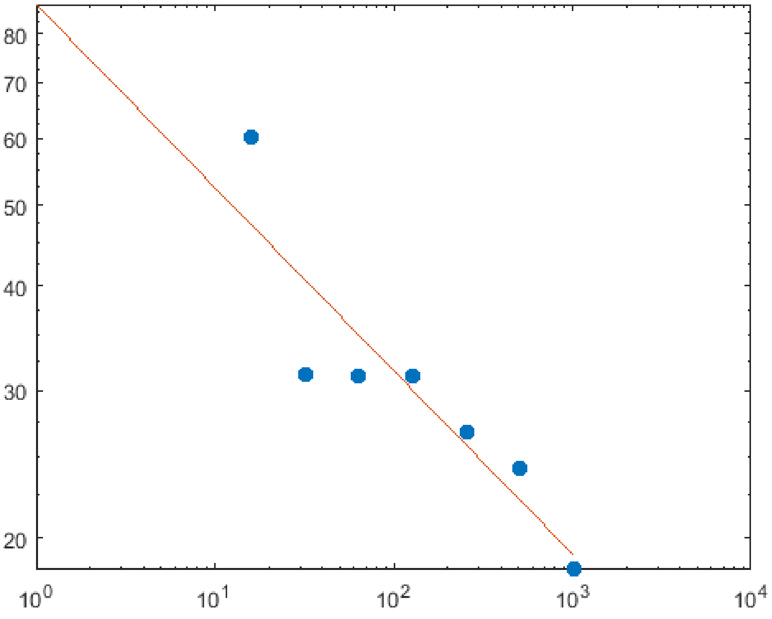


Figure A.11: User Population Outcomes

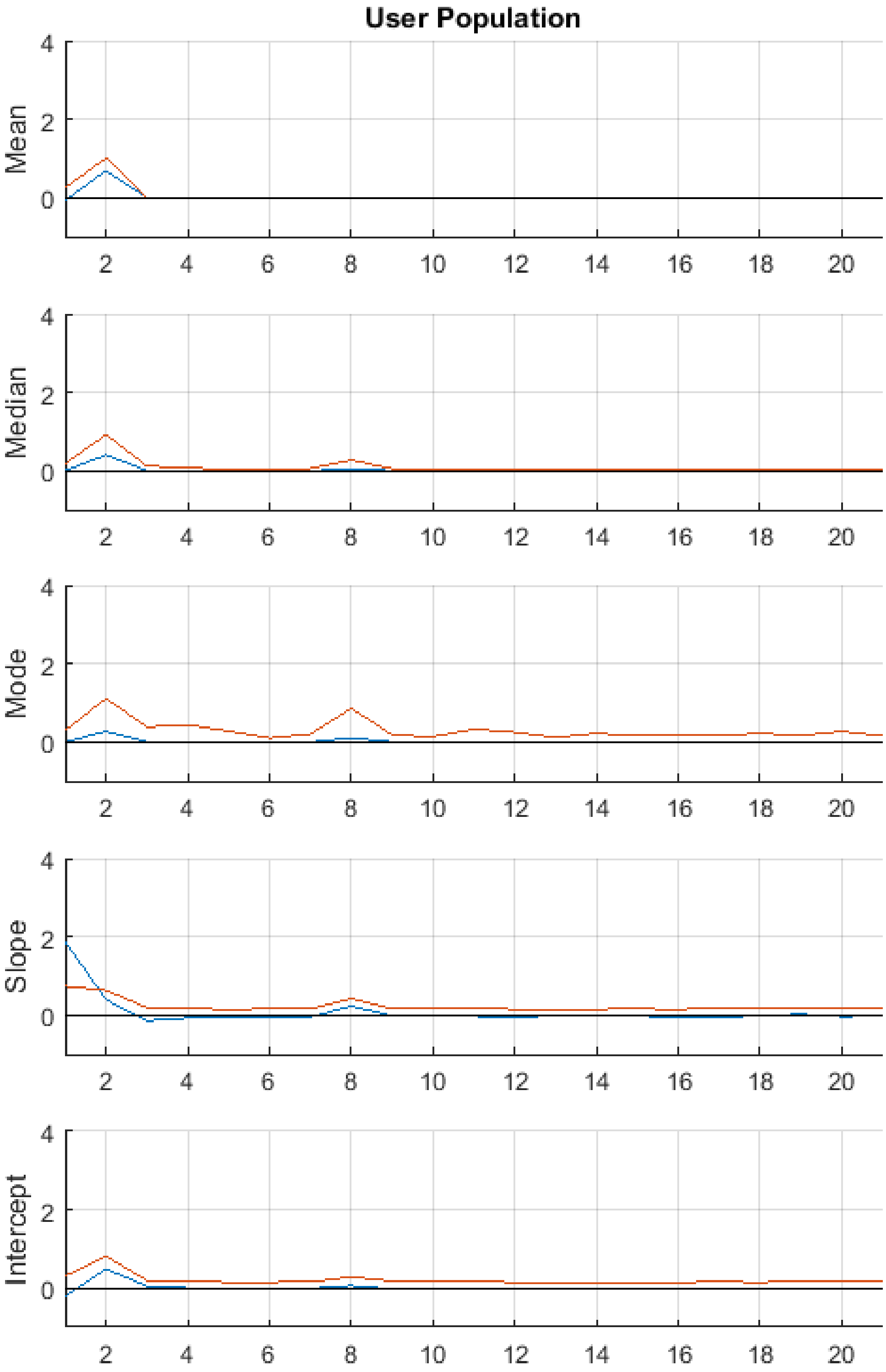


Figure A.12: Community Output Outcomes

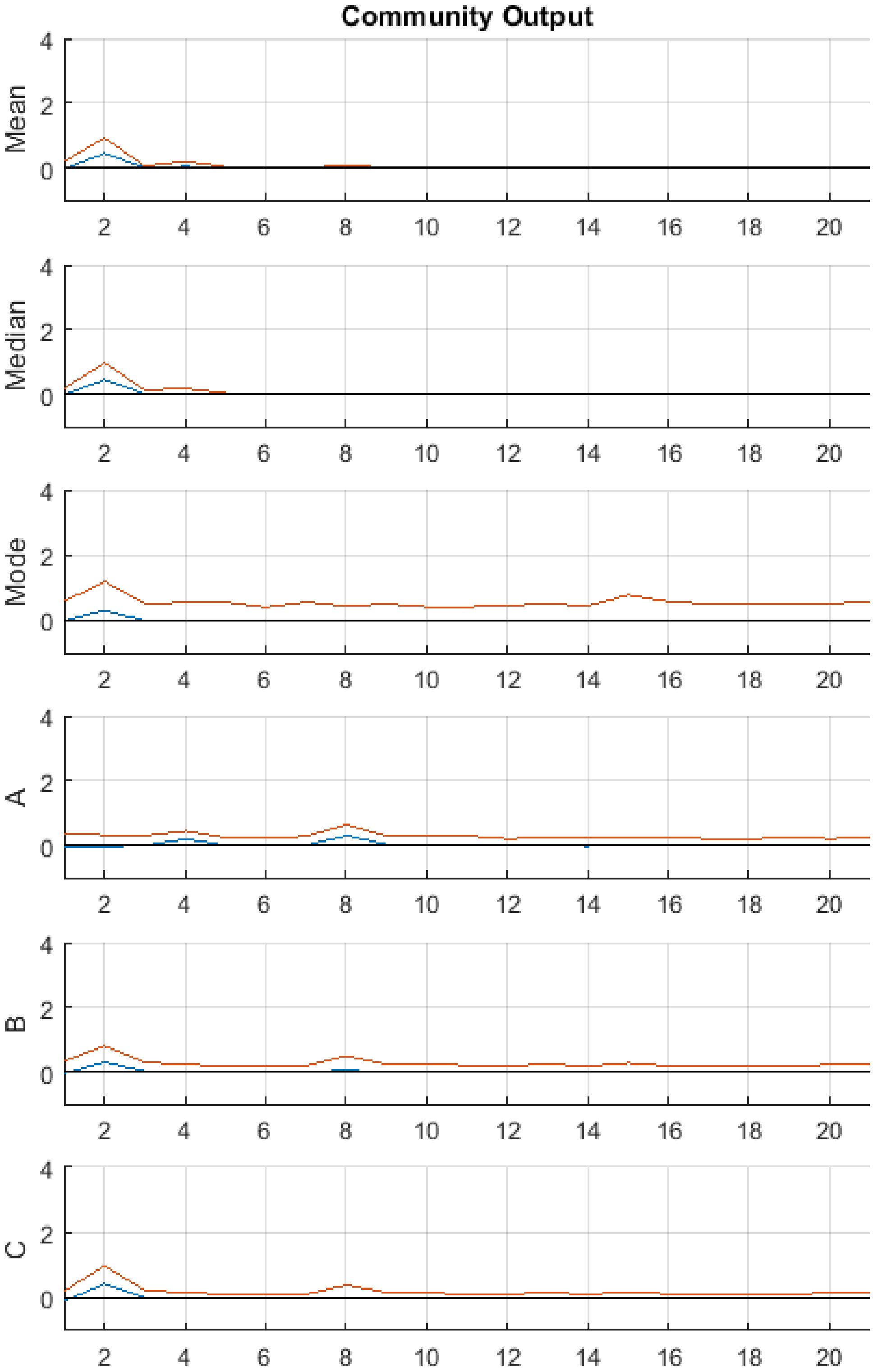


Figure A.13: User Movement Outcomes

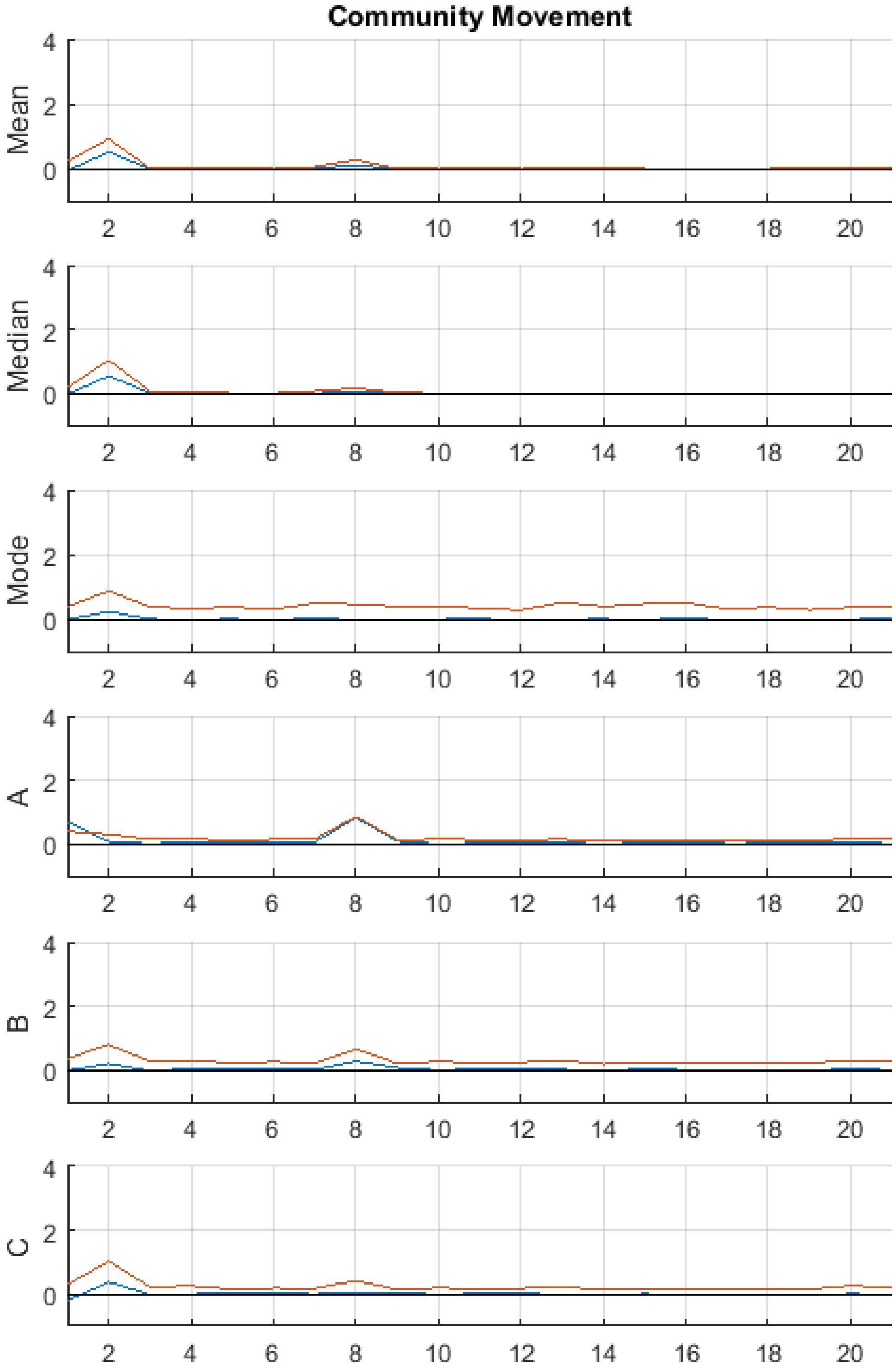


Figure A.14: Return Time Outcomes

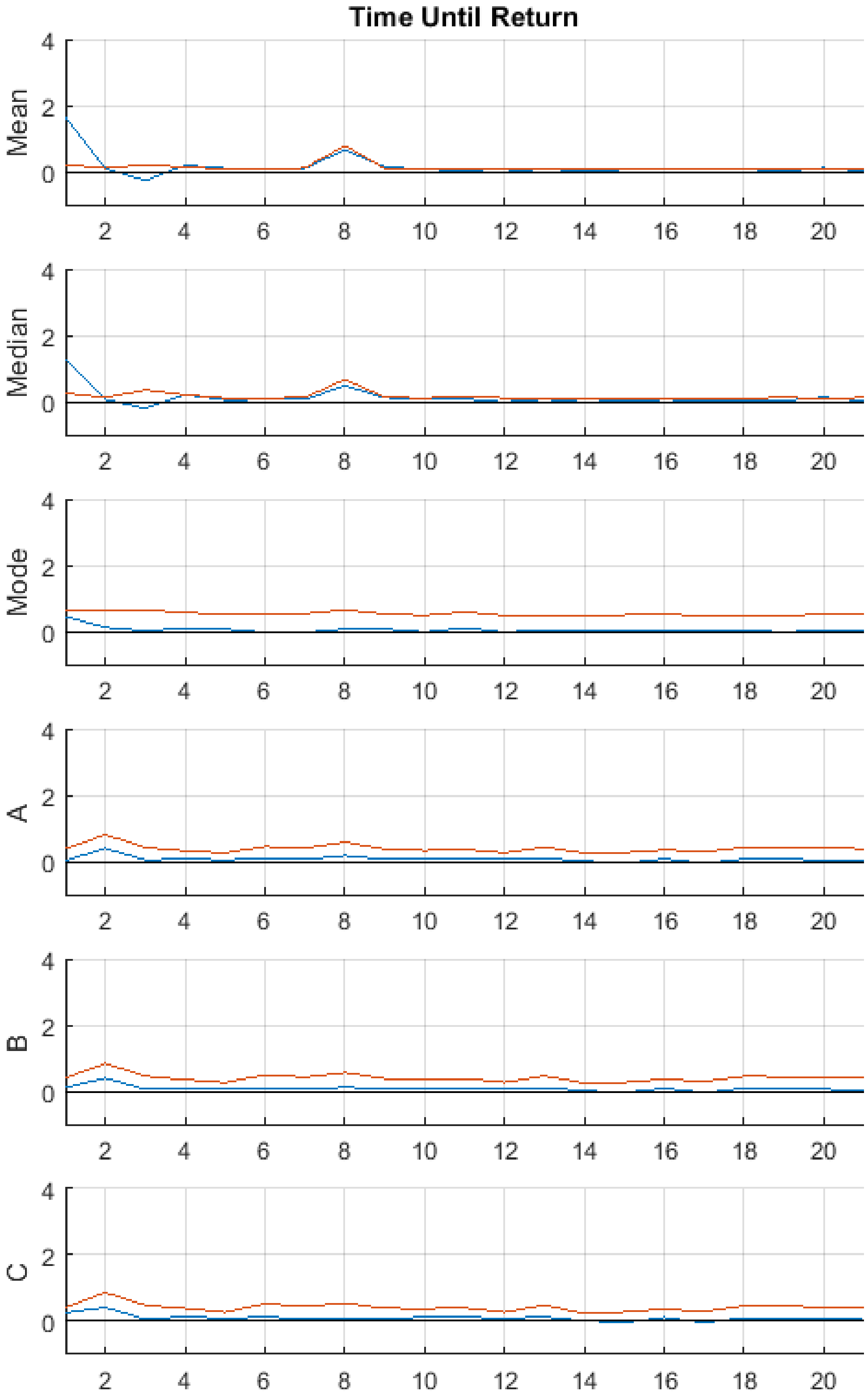


Figure A.15: Likelihood of Contribution Outcomes

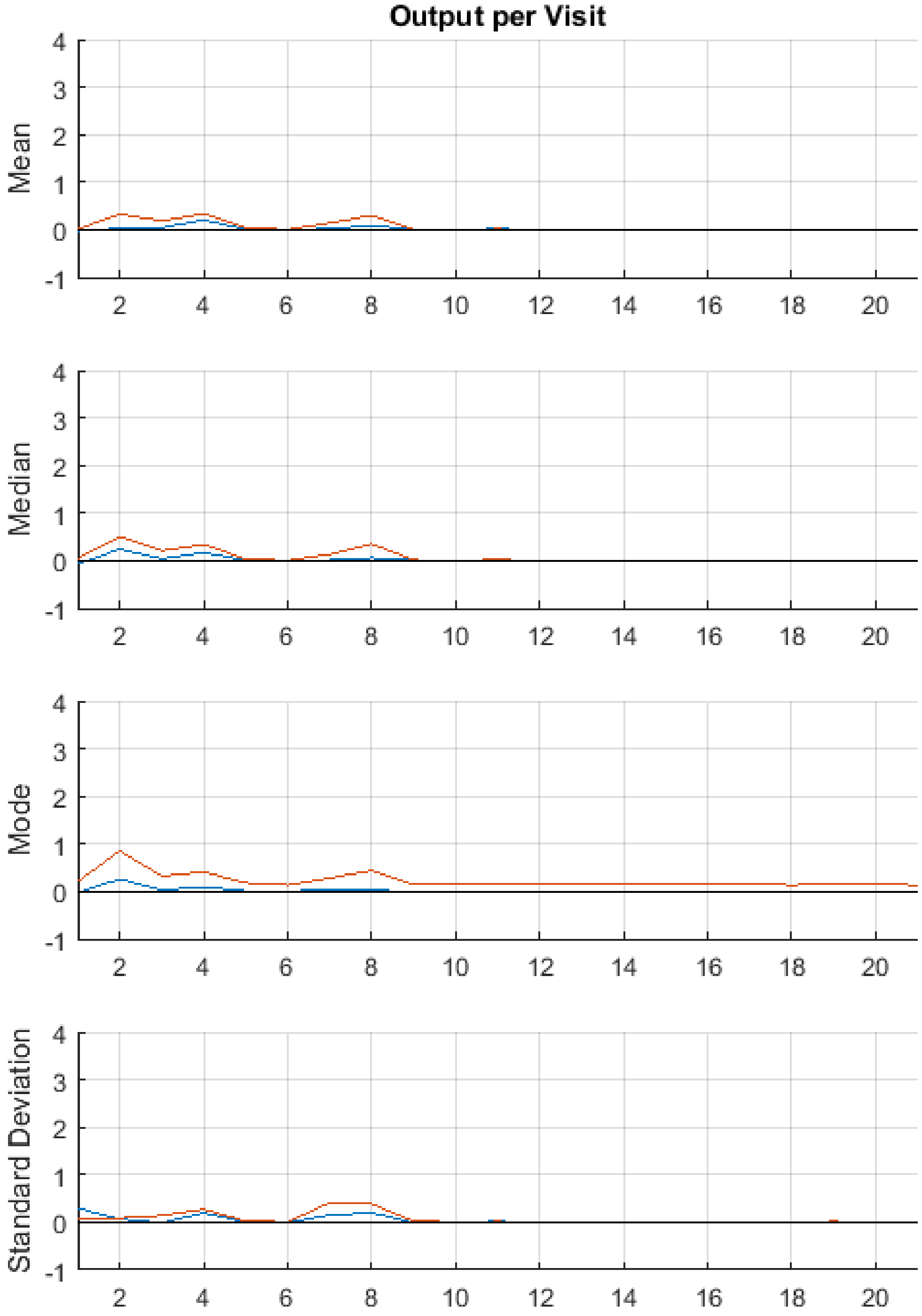


Figure A.16: Relation Between Population and Other Outcomes

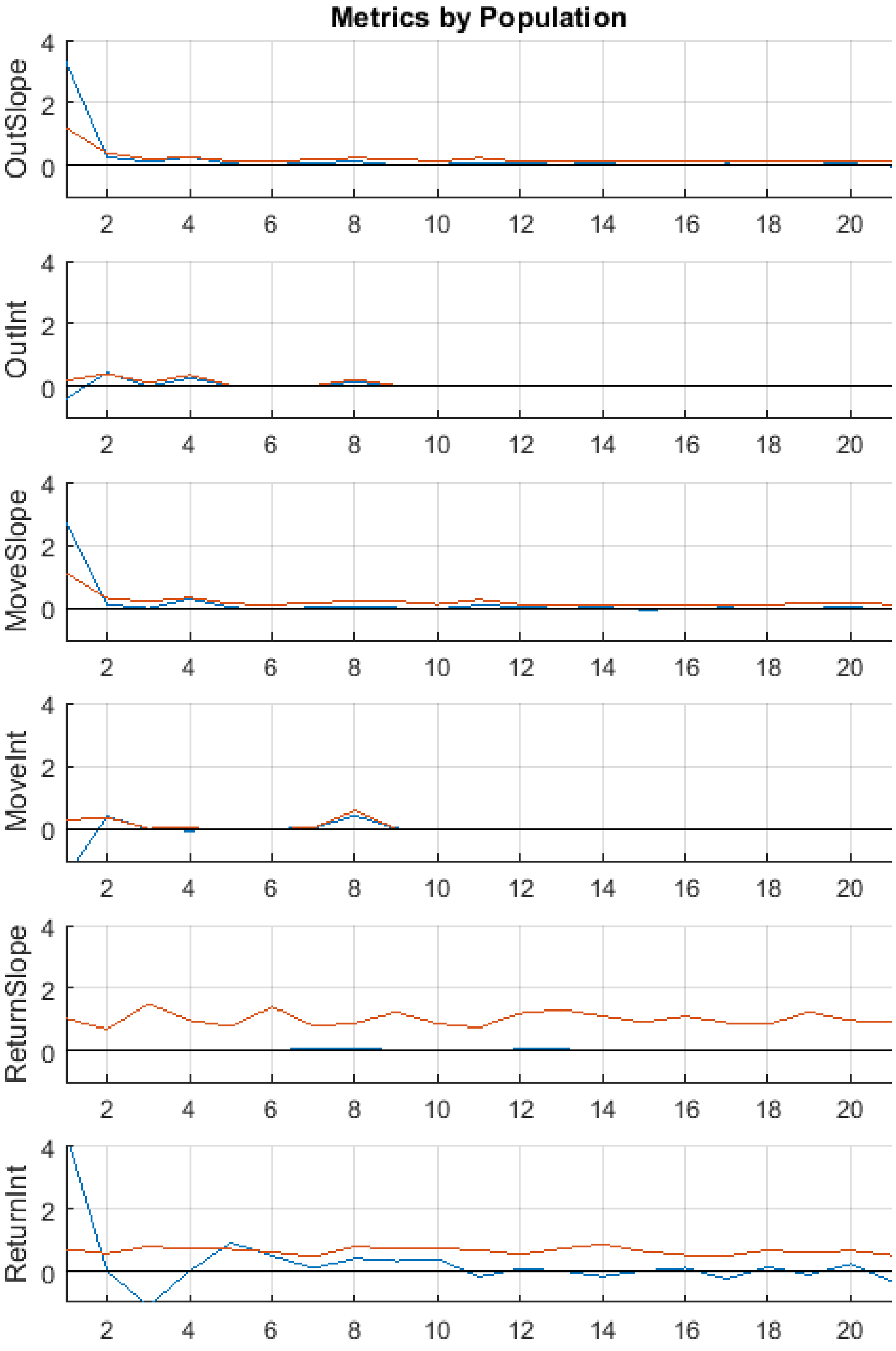


Figure A.17: Relation Between Output and Other Outcomes

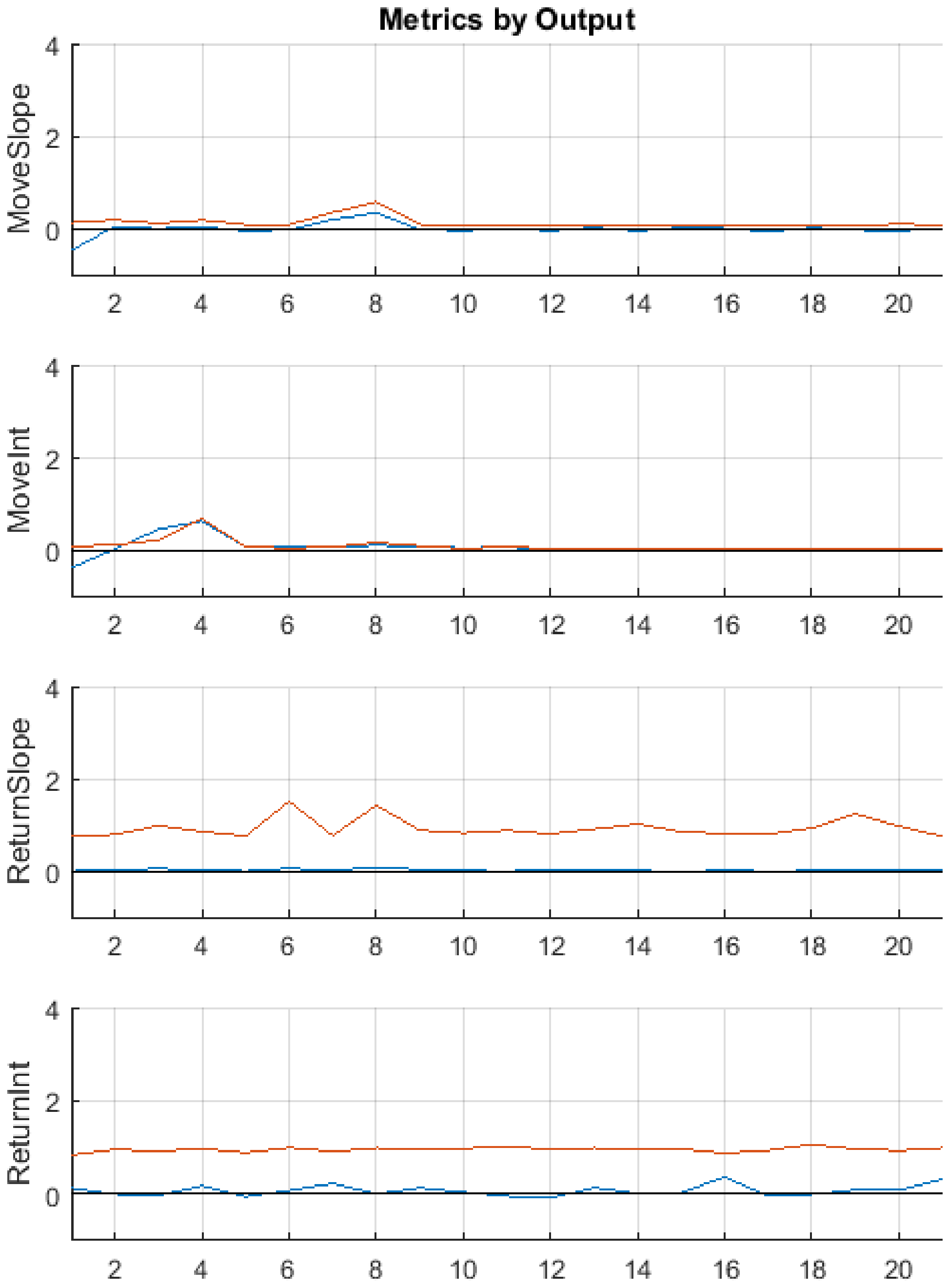
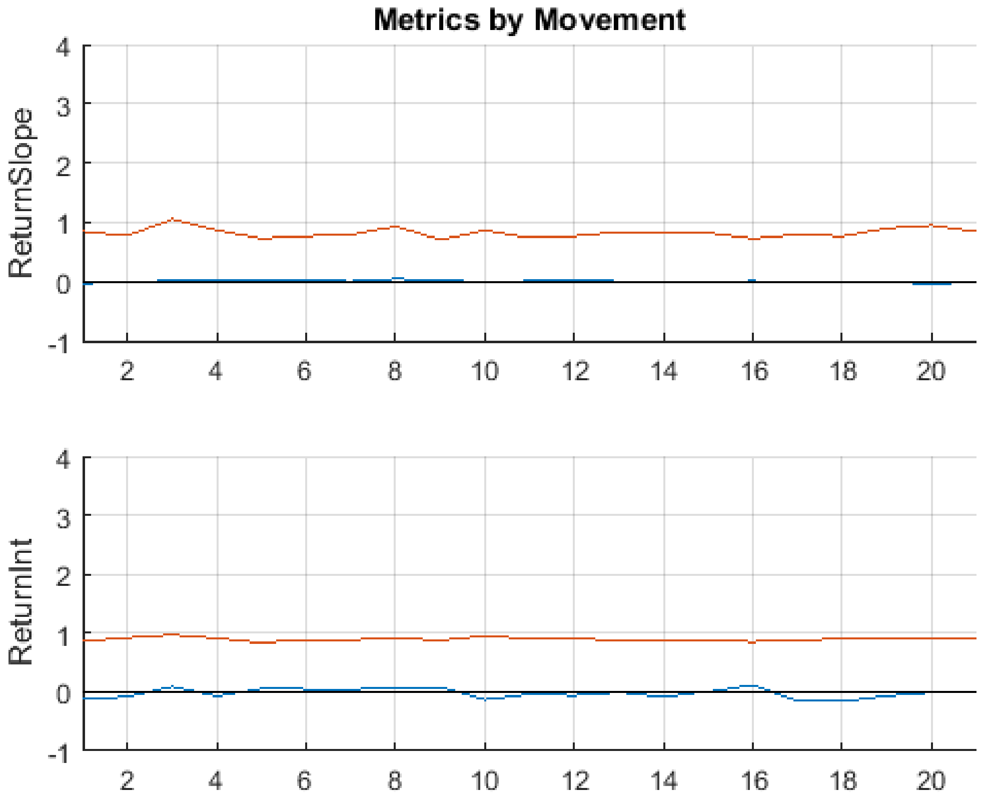


Figure A.18: Relation Between Movement and Other Outcomes


Metrics by Movement
ReturnSlope
ReturnInt