HARVARD UNIVERSITY

Graduate School of Arts and Sciences

DISSERTATION ACCEPTANCE CERTIFICATE

The undersigned, appointed by the

Harvard John A. Paulson School of Engineering and Applied Sciences
have examined a dissertation entitled:

“AI for Population Health: Melding Data and Algorithms on Networks”

presented by: Bryan Wilder

Signature Milind Tambe
Typed name: Professor M. Tambe

Signature ________________________________
Typed name: Professor A. Procaccia

Signature Finale Doshi-Velez
Typed name: Professor F. Doshi-Velez

Signature ________________________________
Typed name: Dr. M. Majumder

June 1, 2021

AI for Population Health: Melding Data and Algorithms on Networks

A dissertation presented

by

Bryan Wilder

to

The School of Engineering and Applied Sciences

in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy
in the subject of
Computer Science

Harvard University
Cambridge, Massachusetts
June 2021

Dissertation Advisor: Professor Milind Tambe Bryan Wilder

AI for Population Health:
Melding Data and Algorithms on Networks

Abstract

As exemplified by the COVID-19 pandemic, our health and wellbeing depend on a difficult-to-measure web of societal factors and individual behaviors. My research aims to build computational methods which can impact such social challenges. This effort requires new algorithmic and data-driven paradigms which span the full process of gathering costly data, learning models to understand and predict interactions, and optimizing the use of limited resources in interventions. In response to such needs, this thesis presents methodological developments at the intersection of machine learning, optimization, and social networks which are motivated by on-the-ground collaborations on HIV prevention, tuberculosis treatment, and the COVID-19 response. These projects have produced deployed applications and policy impact. One example is the development of an AI-augmented intervention for HIV prevention among homeless youth. This system was evaluated in a field test enrolling over 700 youth and found to significantly reduce key risk behaviors for HIV.

Contents

List of Tables

List of Figures

Acknowledgments

I feel incredibly lucky to have had the chance to work with my advisor, Milind Tambe. Thank you, Milind, for giving me the perfect balance of support, advice, freedom, and encouragement. Your passion for work which changes the world is truly infectious. I've learned so much from you and will always be grateful.

I would also like to thank the members of my committee: Ariel Procaccia, Finale Doshi-Velez, and Maia Majumder. All of you have given me valuable advice and support throughout my time at Harvard. To Maia, thank you for going above and beyond in helping me navigate the world of public health.

I am grateful to many other mentors, official and unofficial. Eric Rice, for showing me firsthand what meaningful work looks like and for inviting me to do it with you. Ece Kamar, for giving me the push and the support to think deeply about problems. Eric Horvitz, for your irrepressible enthusiasm, many ideas, and valuable guidance. Bistra Dilkina, for modeling intellectual excellence and commitment. Michael Mina, for willingness to be bold.

The work in this thesis reflects the many coauthors I have been fortunate to collaborate with: Matthew Brown, James Burke, Marie Charpignon, April Chen, Han Ching Ou, Vinod Choudhary, Jaih Craddock, Angel Desai, Graham Diguiseppi, Bistra Dilkina, Priya Donti, Eric Ewing, Aaron Ferber, Anthony Fulginiti, Shahrzad Gholami, James Hay, Kayla de la Haye, Chyna Hill, Eric Horvitz, Lily Hu, Juliana Hudson, Nicole Immorlica, Shahin Jabbari, Mohammad Azizi Javad, Stefanie Jegelka, Ece Kamar, Harshavardhan Kamarthi, Jackson Killian, Zico Kolter, Kavya Kopparapu, Daniel Larremore, Evan Lester, Jose Luna, Maiamuna Majumder, Aditya Mate, Michael Mina, Laura Onasch-Vera, Roy Parker, Andrew Perrault, Robin Petering, Robin Petering, Aida Rahmattalabi, Balaraman Ravindran, Amit Sharma, Soraya Shehata, Nicole Sintov, Matthew Staib, Sze-Chuan Suen, Dana Thomas, Alan Tsang, Phebe Vayanos, Priyesh Vijayan, Eric Rice, Yevgeniy Vorobeychik, Po-Wei Wang, Kai Wang, Nicole Wilson, Darlene Woo, Amulya Yadav, Amanda Yoshioka-Maxwell, Yair Zick. I am grateful for the chance to learn from every one of you.

This journey would have had little of its joy without my friends and labmates, at both

Harvard and USC: Fei Fang, Thanh Nguyen, Leandro Marcolino, Chao Zhang, Yundi Qian, Ben Ford, Debarun Kar, Arunesh Sinha, Yasaman Abbasi, Shahrzad Gholami, Elizabeth Orrico, Omkar Thakoor, Becca Funke, Aaron Schlenker, Sara Mc Carthy, Haifeng Xu, Elizabeth Bondi, Lily Xu, Andrew Perrault, Shahin Jabbari, Kai Wang, Eric Ewing, Aaron Ferber, Caleb Robinson, Arpita Gupta, Han-Ching Ou, Aditya Mate, Jackson Killian, Sanket Shah, Christoph Siebenbrunner, Haipeng Chen, Arpita Biswas, Rediet Abebe. There is far too many that I could say about all of you. I will single out just two groups of people. First: both Aarons, Amulya, Sara, Eric, Andrew, and Lily for exploring the outdoors with me. I'm grateful for the time to talk about any and everything with you (indoors or out). Second: Aaron and Jack for being consistently wonderful friends, gracious roommates, and, relatedly, for tolerating my love of fire.

Finally, thank you to my family. Chuqing, for making every part of life better. I can't wait for our next adventure. Rachel, for being my oldest friend and constant supporter. And most of all, my parents. Thank you, Mom and Dad, for your years of love, encouragement, and guidance. None of this would have been possible without you.

Introduction

Societies around the world face an array of complex obstacles to human welfare: health, homelessness, poverty, and an array of other interlocking challenges disproportionately burden the most vulnerable. Fields such as public health and social work study central questions in the understanding of these phenomena and the design of effective interventions to improve well-being and access to opportunity. These efforts often raise difficult computational questions and it seems clear that data-driven strategies could enable more targeted actions, responsive to the needs of specific communities. However, engagement from the AI community with population or community-level questions has thus far been somewhat limited, especially when compared to the huge amount of work which focuses on individual-level health (i.e., clinical settings). Important advances have been made on specific questions such as forecasting epidemics [YSK15, BFH$^+$18, AXRP19, WCM19, ZY20], planning vaccination campaigns [PAI$^+$13, SAPV15, ZAS$^+$16], or allocating services related to homelessness or poverty [KWVR19, KDF19, AKW20]. However, relative to the enormous impact that advances on these problems would have, there is thus far only limited systematic understanding of the principles behind designing AI approaches to such domains. Accordingly, the question studied in this thesis is:

How can we build AI which impacts population health?

Designing algorithmic or machine learning approaches for these real-world settings poses a number of challenges. To name a few:

- Such problems are situated in a complex social context involving a *range of stakeholders*.

At the very least, this includes both members of the impacted community, who are often in some way vulnerable or marginalized, and government or nonprofit actors charged with providing services to the population. Computational approaches must be designed with awareness of these dynamics to produce interventions which center the needs of impacted communities and are practically useable by actors who have the capacity to implement an intervention.

- Beyond the need to consider people as stakeholders, the systems which we seek to intervene in are fundamentally social, with outcomes driven by the collective decisions, behaviors, and interactions of people in a give community. This creates a need for algorithms to explicitly reason about *networks* and other forms of social structure.
- Interventions are inevitably subject to *limited resources*. No public health agency, NGO, or housing authority has the ability to offer all of the services they would like to everyone they would like, forcing fraught decisions about resource allocation.
- The targeting of these resources relies on *limited data*. Important information, e.g., the social structures underlying a domain, or the true burden of disease, is often not well known. Gathering data to answer these questions is itself an expensive endeavor which trades off against other important goals.

As a field, artificial intelligence is in the early stages of its engagement with such domains. A great deal of work has been done on topics such as health or sustainability, and related areas such as fairness and accountability. However, comparatively little work specifically engages with population or community-level interventions, or has succeeded in transferring proposed AI techniques from the lab to the community. This work contained in this thesis encapsulates the complete process of working from community engagement, to data, to decisions, to implementation. Along the way, we lay algorithmic foundations for optimization and resource allocation under uncertainty, and for the design of machine learning models which enable effective decision making. Figure 1 presents an overview of this process.

It is worth addressing at the outset that the use of computing in socially consequential settings has been subject to substantial criticism. One concern is the propensity for computational approaches to entrench existing biases or inequalities [BG18, ALMK16, OPVM19, BCZ$^+$16]. Another is that computational researchers typically have operated within existing power structures instead of being pursuing broader societal reforms [Eub18, Gre20, DK20, WCD$^+$18, Hof19]. The viewpoint taken in this thesis is that technology never provides the solution by itself to a social problem. However, I contend that technology *can* serve to augment human capabilities and, in doing so, substantially increase the effectiveness of larger efforts to address social challenges. Accordingly, the responsibility of computational researchers is to work closely with stakeholders in a given domain to understand how computation can support their goals [Hay11, FHB17, BXANK21]. It is incumbent on researchers to account for the impact of their work on all of those who are effected, including those who have less control over, or are not well served by, existing organizational structures [IK21]. This often requires spending time to conduct fieldwork and learn from people in a variety of roles (e.g., members of partner organizations, researchers in other disciplines who have relevant expertise, frontline workers, individuals receiving services, etc.). The work presented in this thesis on HIV prevention and tuberculosis treatment was informed by exactly this process. A more extended discussion of related considerations can be found in Chapter 4 (which includes a reflection on lessons learned about the process of community-oriented AI research), Chapter 11 (which considers the problem of training machine learning models to complement human capabilities), and Chapter 14 (which outlines an agenda for the training of computational researchers competent to work in community settings).

Overview of contributions

This thesis is divided into four parts. Part I (Networks and health) covers the algorithmic foundations, real-world deployment, and evaluation of an AI-augmented intervention for HIV prevention amongst youth experiencing homelessness. Part II (Uncertainty and optimization) presents algorithmic approaches to solving optimization problems motivated by

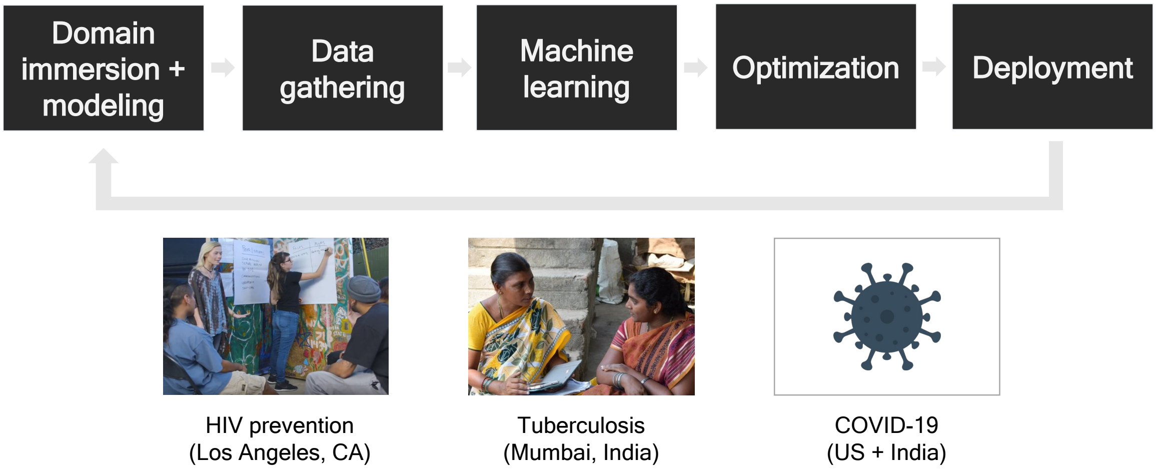

Domain immersion + modeling
Data gathering
Machine learning
Optimization
Deployment
HIV prevention (Los Angeles, CA)
Tuberculosis (Mumbai, India)
COVID-19 (US + India)

Figure 1: *The process of developing and deploying AI-augmented interventions, alongside the main application domains for this thesis.*

resource allocation in public health settings, with a particular focus on allocating resources under uncertainty. Part III (Learning and decisions) introduces a set of techniques for integrating downstream decision or optimization problems into the training of machine learning models, an approach we refer to as decision-focused learning. It also provides a case study of these ideas in the context of tuberculosis treatment. Part IV (Inference and epidemics) presents work undertaken in response to the COVID-19 pandemic, focusing on the development of modeling and inference techniques to uncover unobserved disease dynamics. This thesis draws on material published in [WIRT18, Wil18a, Wil18b, WST18, WOVH$^+$18, WDT19, KWS$^+$19, TWR$^+$19, WEDT19, WHK20, WOVD$^+$21, WCK$^+$20, WMT21]. We now present an overview of the contributions made in each chapter.

Part I: Networks and health

Part 1 discusses the development and deployment of an AI-augmented intervention for HIV prevention among youth experiencing homelessness (YEH). HIV is a key public health challenge for YEH, with reported prevalence in the range of 2-11% [YR11]. One promising intervention relies on *peer leaders* recruited from the population of YEH to advocate for the adoption of protective behaviors. The most common way of selecting peer leaders is to

Figure 2: *Left: Administering surveys at a drop-in center for youth experiencing homelessness. Right: community partners for HIV prevention intervention.*

identify the most popular individuals in the social network of the youth [KMS+97] (formally, the highest degree nodes). However, the peer leader model has suffered notable failures in other HIV prevention contexts [G+10], potentially attributable to how peer leaders are selected [SZL15]. The algorithmic question becomes: *how can we identify the most influential set of peer leaders for a behavioral intervention?* This question has relevance far beyond HIV prevention; analogous social network interventions are used widely across development, medicine, education, etc. [KHS+15, PSA16, BCDJ13, VP07].

The problem of maximizing information diffusion on social networks has been extensively studied in theoretical computer science [KKT03, CWY09, CWW10, GLL11a, BBCL14, TXS14]. The canonical problem formulation, introduced by Kempe, Kleinberg, and Tardos [KKT03], asks the algorithm to select a limited budget of up to k seed nodes from a graph G. The objective is to maximize the expected number of nodes reached with information under a model of information diffusion such as the independent cascade model, where every edge independently propagates influence with some probability. Efficient algorithms, rooted in submodular optimization, are now available for the influence maximization problem as a result of the long history of work. However, there has been little interaction between the computation literature on influence maximization and network interventions in fields like public health or social work. Computational work has mainly focused on scaling algorithms for the standard problem formulation to increasingly large graphs (often motivated by advertising), while interventionists in health domains have not used explicitly algorithmic approaches to optimize the selection of peer leaders. As a result of this gap, existing algorithmic tools are difficult to apply in the field: they assume access to data (such as the structure of the social network) which is difficult and expensive to acquire in a real-world setting.

Part I of the thesis begins by developing and analyzing algorithms to address some of the key technical bottlenecks to the real-world use of influence maximization. **Chapter 1** introduces the problem of influence maximization with an unknown network, where the structure of the graph G is not known ahead of time. Instead, the algorithm has the ability to query the graph structure by surveying a chosen node, revealing that node's set of neighbors. Reflecting the fact that the real-world instantiation of this process involves face-to-face surveys with homeless youth, these queries are expensive and we would like to use as few as possible. The goal is thus to (1) select a set of seed nodes who are guaranteed to be approximately as influential as the optimal set given full knowledge of the network structure and (2) bound the number of queries to the graph structure required to obtain this approximate solution. In the worst case, we find that these goals are incompatible: there are graphs where any algorithm which obtains a constant-factor approximation to the optimal influence spread must query all but a $o(1)$ fraction of the network. Fortunately though, we show that realistic forms of network structure can be leveraged for an exponentially better outcome. In particular, graphs often exhibit community structure, where nodes lie in distinct groups, with much denser connections within communities than between them. Our theoretical analysis shows that when graphs have this kind of structure, formalized in the canonical stochastic block model, we can obtain a guaranteed constant-factor approximation to the omniscient optimum while querying only $O(\log n)$ nodes. The algorithm simulates a series of short random walks on the graph, and uses the information gathered on these walks to reconstruct a limited amount of the community structure. Despite the fact that most of the community structure cannot be identified from such little data, we prove that it is possible to identify a set of nodes which seed the largest communities in the graph with constant probability.

Chapter 2 tackles another challenge related to the limited data often available in targeting real-world network interventions. In particular, the parameters describing how information will propagate across the graph G are typically not known, and are very difficult to elicit at all. For example, in the independent cascade model, we might have a parameter θ_{ij} for every pair of nodes (i, j) which gives the probability that i will influence j. Since the parameters θ will typically be unknown, an attractive solution concept is given by the robust optimization problem:

$$\max_{|S| \leq k} \min_{\theta \in \Theta} f(S, \theta)$$

where we seek a set of nodes S, up to a budget constraint k, which maximizes the worst-case influence spread over an uncertainty set of possible parameters Θ. Here $f(S, \theta)$ denotes the expected number of nodes influenced by seed nodes S under parameters θ. We can view this as an instance of the more general *robust submodular optimization* problem where f is an arbitrary submodular function with parameterization θ. While algorithms have been developed for robust submodular optimization [KMGG08, AHN$^+$17, CLSS17], their runtime scales with the size of the uncertainty set Θ because it is necessary to essentially enumerate the entire uncertainty set in every iteration of the algorithm. This poses a serious computational burden in combinatorial domains like influence maximization where Θ becomes large – perhaps exponentially large in the size of the graph [HK16], or a continuous set [SWJ19]. We introduce the first approximation algorithm for robust submodular optimization whose runtime is *independent* of $|\Theta|$. Instead, it requires only that we can efficiently solve the inner minimization problem $\min_{\theta \in \Theta} \mathbb{E}_{S \sim P}[f(S, \theta)]$ for a fixed distribution over sets P. To demonstrate the broad reach of this algorithmic framework, we show applications to a range of other domains (budget allocation [SKIK14] and network security games [TYK$^+$10]).

Chapter 3 begins the process of applying these algorithmic foundations to the HIV prevention problem. We propose a system called CHANGE which integrates together ideas from robust optimization, network sampling techniques inspired by Chapter 1, and additional operational modifications to produce a field-ready system for influence maximization. We find that the last mile before deployment introduces significant challenges. These challenges were uncovered by conducting real-world pilot tests of potential algorithmic systems, in collaboration with partners in social work and local community organizations. These pilot tests allowed us to refine and validate our algorithmic ideas to propose the final CHANGE system, setting the stage for the larger-scale trial discussed in Chapter 4.

One example of such a last-mile problem concerns the sampling algorithm from Chapter 1. Recall that the algorithm queries the network by simulating a series of random walks. While theoretically appealing, this approach can be difficult to practically operationalize because it requires finding a specific sequence of youth (who may or may not be present that day at the center offering services). We propose a simplification of the algorithm which uses less adaptivity (i.e., fewer queries which depend on the results of previous queries) and which succeeds by leveraging an additional property of social networks referred to as the friendship paradox. Another challenge relates to attendance at the intervention itself. Prospective peer leaders selected by the algorithm attend a day-long training on HIV awareness and outreach skills. However, there are any number of barriers which could prevent a given YEH from being able to attend on the day-of. In practice, a series of classes will be conducted, each of which trains a set of 3-4 peer leaders. Further, the youth selected as peer leaders in any given class can be chosen with knowledge of which youth actually attended previous classes. We accommodate this complication by building a stochastic extension of the objective function reflecting imperfect attendance and proving that that multi-stage optimization problem enjoys the favorable property of adaptive submodularity, removing the need to look far into the future when planning.

Chapter 4 presents the results of a large-scale field trial of CHANGE. Together with social work colleagues and three drop-in centers which offer services to YEH in the Los Angeles area, we ran a trial which enrolled 718 youth over the course of several years. This trial compared three arms: first, where peer leaders were selected using CHANGE;

second, where the highest degree nodes were selected (the method mostly commonly used in practice); and third, an observation-only control group. *Our results show that youth in the CHANGE arm of the trial experienced significant improvements in their rates of condomless sex, a key risk behavior for HIV (approximately a 30-40% reduction in relative risk).* By comparison, there was at most limited statistical evidence for any improvement by youth in the degree centrality arm. This study provides, to our knowledge, the first rigorous empirical evaluation of algorithmic techniques for social network interventions in health. We draw two broad conclusions from this process. First, AI has a great deal of potential to impact the design of interventions in socially important contexts such as public health and social work – algorithmic approaches which systematically optimize the entire set of peer leaders seem to offer a large improvement over the status quo. Second, AI decidedly does not work "out of the box". The development of a successful AI-augmented intervention resulted from a long process of engagement with domain experts and community members, weeks spent by us (the AI researchers) at the drop-in centers doing field work, and then, *during and after* that process, the design and analysis of algorithms to address the new technical challenges we discovered. The chapter concludes with a reflection on the lessons learned from this experience.

Part II: Uncertainty and optimization

Part 2 of the thesis explores a range of optimization problems inspired by public health domains, developing the algorithmic theory to support effective decision-making under uncertainty.

Chapter 5 studies an optimization problem motivated by preventing the spread of an infectious disease. The particular motivation deals with tuberculosis (TB), a devastating epidemic which impacts millions of people each year. For TB, as with many infectious diseases, interventions must be planned under considerable uncertainty about important parameters such as the underlying distribution of disease in the population or the rate of spread of the disease between different population groups. We introduce an age-stratified model of disease spread aimed at capturing a range of endemic diseases like TB. We then consider the problem of targeting a budget-constrained intervention which increases the uptake of treatment in chosen population groups (e.g., case-finding drives where frontline health workers attempt to enroll undiagnosed TB patients in treatment). We show that, while this problem is nonconvex, it exhibits a continuous submodularity property which allows us to provide an efficient approximation algorithm for stochastic optimization over the unknown parameters. Experimental results on a problem setting motivated by TB in India, as well as a scenario motivated by gonorrhea in the US, show that our algorithm produces more effective strategies than classical heuristics.

Chapter 6 studies the problem of risk-averse submodular optimization. Decision makers in many domains face uncertainty. In response to this uncertainty, a common objective is to find a decision which maximizes expected utility. However, expected utility maximization may be inappropriate in many settings (including socially consequential ones) where it is disproportionately important to avoid negative outcomes. Risk measures such as the conditional value at risk (CVaR) formalize this objective. The problem of optimizing the CVaR of a decision is well-understood algorithmically when the decision can be modeled as a convex optimization problem. However, the picture is much less clear when the decision-maker's utility is nonconvex. In this chapter, we study the problem of maximizing the CVaR of a continuous submodular function (including, as a special case, the disease prevention objective introduced in Chapter 5). We introduce the first approximation algorithm for this problem, which obtains an efficient $\left(1 - \frac{1}{e}\right)$-approximation to the optimal solution. We also show how this algorithm can be extended to handle optimization of set functions in the portfolio setting (where a distribution over sets is chosen). We provide experimental results for a contagion detection problem, where the goal is to detect a spreading process on a graph (such as an infectious disease or environmental contamination) by placing sensors. These results show that CVaR optimization provides an effective and principled way to diversify the set of selections and hedge against the risk of catastrophic outcomes.

Chapter 7 considers the challenge of ensuring fairness in network interventions, such as the HIV prevention intervention introduced in Part 1. Especially in socially consequential settings, it is important to guarantee that interventions will have an equitable impact across groups in the population; for network interventions, we might particularly worry that already-marginalized groups will be less well connected in the network and so will not be prioritized by algorithms which maximize total influence spread. We propose optimization formulations which modify the influence maximization problem to explicitly consider group-level fairness. We find that these problems fall under the umbrella of the more general multi-objective submodular optimization problem, which provides a flexible abstraction for algorithmic instantiations of group fairness in this domain. For this more general problem, we introduce a new approximation algorithm which improves simultaneously on the approximation ratio and runtime offered by the previous state of the art.

Part III: Learning and decisions

Part 3 of this thesis studies settings where decision-making under uncertainty can be explicitly informed by machine learning from historical data. It develops a set of techniques to integrate a decision problem of interest into the training of a machine learning model so that the model can be trained directly to induce the best decisions possible (instead of minimizing a surrogate loss function measuring predictive accuracy). The goal is to enable more actionable *decision-focused* machine learning, where machine learning models can easily be trained for their intended use case – an especially important concern when machine learning will be just one step in an intervention for a larger social challenge.

Chapter 8 introduces the problem of training a machine learning model which will be used downstream in a discrete optimization problem. Specifically, suppose that we would like to solve an optimization problem $\max_{x \in X} f(x, \theta)$ where x is a (discrete) decision variable and θ is an unknown parameter. Previous chapters considered approaches for handling θ like robust or stochastic optimization, which are appropriate when only a set or distribution of possible scenarios can be formulated. However, suppose that we have a set of features y with which θ can be predicted. Now, we can train a machine learning model $m(y)$ which outputs a predicted $\hat{\theta}$ and solve $\max_{x \in X} f(x, \hat{\theta})$ (optimizing under the prediction made by the model). Define $x^*(\theta) = \arg\max_{x \in X} f(x, \theta)$ to be the optimal decision given parameters θ. The question is how we should train the model m to predict θ, knowing that we ultimately care about $x^*(\theta)$. Suppose that our training data consists of a set of historical examples $\{y_i, \theta_i\}$ iid from an unknown joint distribution P. This is the standard supervised learning setting. Our true objective is to find a model m which optimizes

$$\mathop{\mathbb{E}}_{y,\theta \sim P}[f(x^*(m(y)), \theta)], \tag{1}$$

which gives the expected utility of the decision induced by m. However, a standard *two-stage* approach would first train m to minimize a surrogate loss function ℓ, and then plug the resulting $\hat{\theta}$ into an optimization algorithm to produce x^*. Misalignment between the loss function and the optimization problem means that the two-stage process can be far from optimal for decision-making, especially for difficult learning problems where any model is far from perfect.

We instead propose approaches to directly train m to optimize the decision-focused objective in Equation 1. The key idea is to differentiate through the function $x^*(\theta)$, which allows m to be trained end-to-end via gradient descent. This requires computing derivatives through the solution of the optimization problem, expressing how x^* varies as a function of the predictions θ. Previous work on differentiable optimization [RU00, AK17, DAK17] focuses on strongly convex optimization problems. However, many socially important settings involve the allocation of indivisible resources, and discrete optimization is by construction not differentiable.

This chapter proposes to use a continuous relaxation of the discrete problem as a differentiable surrogate during training. We study two common classes of discrete optimization: linear programs and submodular maximization. In the case of linear programs, we prove that adding a quadratic regularizer to the objective function suffices to guarantee differentiability. For submodular maximization, we show that the multilinear relaxation provides an effective and differentiable continuous surrogate, and provide a means to efficiently

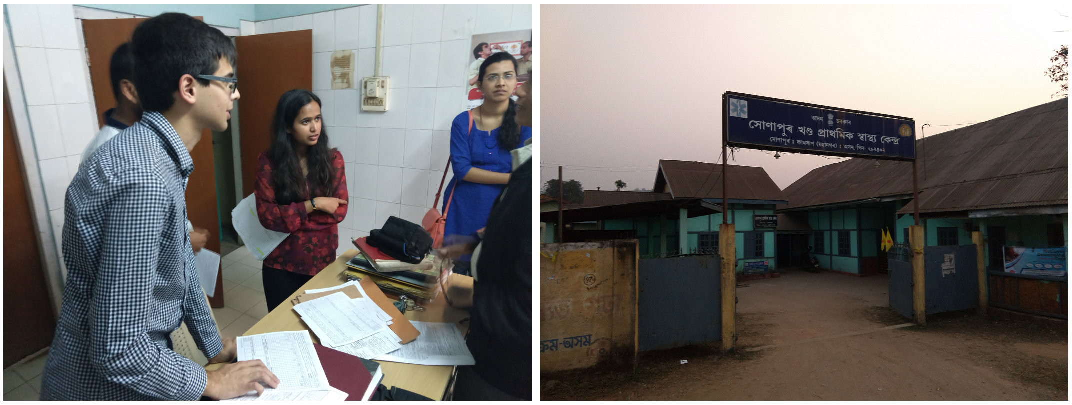

অসম চৰকাৰ
সোণাপুৰ খণ্ড প্ৰাথমিক স্বাস্থ্য কেন্দ্ৰ
সোণাপুৰ ঃ কামৰূপ (মহানগৰ) ঃ অসম, পিন- ৭৮২৪০২

Figure 3: *Left: tuberculosis clinic, Sonapur, India. Right: the primary health center which houses the TB clinic.*

compute dual variables needed for the backward pass. Experimental results on real and synthetic datasets show that our proposed methods are able to improve the quality of downstream decisions compared to standard two-stage methods.

Chapter 9 applies the techniques developed in Chapter 8 to the problem of improving adherence to tuberculosis medication. First-line treatment for TB requires a six-month course of antibiotics. Non-adherence to antibiotic treatment can lead to a range of complications such as reinfection with active TB or the development of antibiotic resistance. In many countries, such as India, frontline health workers are responsible for supporting patients in staying on treatment. This work was conducted in collaboration with the Government of Maharashtra, focusing on TB care in the city of Mumbai. TB patients in Mumbai use a digital adherence technology called 99DOTS, where patients leave a missed call at a specially chosen phone number each day to indicate that they took their medication. The challenge is that, given high case loads, health workers have limited resources to follow up with patients who are at risk of nonadherence. For example, a frontline health worker might make home visits to particularly at-risk patients, but these visits can be made to at most a small number of patients each day.

We used historical adherence data from the 99DOTS platform to develop a decision-focused approach which first predicts the risk of nonadherence for each patient and then suggests an optimal assignment of health workers to visit patients. We draw on the techniques formulated in Chapter 8 to formulate a differentiable relaxation of a linear program which models the matching of health workers to locations. We compare this decision-focused approach to a two-stage approach which trains the machine learning model using a standard loss function. *Our results on the 99DOTS dataset, which encompasses over 17,000 patients and 2.1 million interactions, show that the decision-focused approach results in approximately 15% more successful interventions than the two-stage approach.* This is despite the fact that the two-stage baseline has better predictive accuracy, as measured by the AUC. The broader takeaway is that when machine learning models will be used in the context of a larger intervention, standard measures of predictive accuracy are not a sufficient measure of their performance. We need approaches which are expressive enough to allow machine learning models to be trained with the complete pipeline of the intervention in mind.

Chapter 10 explores more of the technical foundations of decision-focused learning, proposing an alternate strategy for differentiable optimization. While differentiable relaxations can provide appealing training surrogates for some problems, it can often be difficult to find relaxations which induce tractable training landscapes. Moreoever, approaches which derive from convex optimization typically require $O(n^3)$ time for each backward pass, where n is the number of decision variables. In this paper, we propose to instead learn representations which enable easier optimization. Specifically, we use a trainable model (e.g., a graph convolutional network) to map a discrete input to a continuous representation space. Then, we solve a simple surrogate problem in the continuous space, and interpret the output of the surrogate as a solution to the original discrete problem of interest. This idea is instantiated in the chapter using k-means clustering as the surrogate problem. For example, if we wished to choose k vertices of a graph as locations for facilities, we could produce a continuous embedding of the nodes of the graph, use soft k-means clustering to group the nodes into clusters in the embedding space, and then locate facilities at the nodes which are closest to the cluster centers. The model is trained end-to-end, with the clustering algorithm as part of the training loop. Essentially, this allows the model to learn a continuous representation which maps the original discrete problem to an instance of clustering. We show that this strategy enjoys a range of favorable properties, both theoretically and empirically: we can obtain a provably good approximation to the backward pass in time $O(n)$ instead of $O(n^3)$, and the resulting model empirically offers strong generalization to unseen graphs. More broadly, we argue that this system is an instantiation of a paradigm for combining learning and optimization, where we co-train both the learning and optimization components of the system by combining representation learning with a simpler differentiable algorithm that provides a surrogate for optimization.

Chapter 11 explores ideas related to decision-focused learning in a different setting: one where a machine learning system will function as part of a team with human experts. For example, imagine a machine learning model which will be used to diagnose the presence of cancerous cells in an image. In reality, this model is unlikely to be used in isolation. Rather, some cases will be escalated to a human pathologist if the outcome is unclear, and this expert might have information or forms of reasoning which are unavailable to the algorithm. We propose that machine learning models which will be used as part of such teams should be trained specifically for *complementarity* with the human expert. That is, the model should be able to recognize the characteristic strengths and weaknesses of humans and machines in the domain. This would allow its training to focus on instances which are difficult for humans, with the knowledge that other instances might be difficult for the model but can safely be left to the human. We propose training strategies which directly optimize for the combined performance of the team as a whole, instead of optimizing the model's performance in isolation. One technical contribution is to introduce a differentiable surrogate for the value of information computation, where the model estimates whether the expected benefit to querying the human outweighs the cost of doing so. We conduct experiments on two real world domains: a citizen science task, and breast cancer diagnosis. In both domains, our results show that optimizing ML models specifically for complementarity results in better team performance.

Part IV: Inference and epidemics

The final part of this thesis contains work directly related to the COVID-19 pandemic. The aim is to develop computational tools, rooted in modeling and Bayesian inference, which can make help make sense of noisy outbreak data, uncover the dynamics of the pandemic across different populations, and inform the design of policy interventions.

Chapter 12 develops an agent-based model for COVID-19 dynamics which incorporates a rich array of demographic and household structure information. We develop a Bayesian inference method to infer posterior distributions over unknown model parameters, allowing us to uncover unknowns in the dynamics of early outbreaks in Hubei, Lombardy, and New York City. We find evidence for consequential differences in dynamics across these populations, e.g., differing levels of transmissibility and case ascertainment in the first wave. We then simulate the potential impacts of policy interventions which reduce the contact levels of particular age groups in each population. We similarly find that, depending on variations in first-wave dynamics, demography, and social structure, there is no "one size fits all" solution. Instead, the most effective set of contact reductions varies across all three populations.

Chapter 13 focuses on inferring the growth rate of a partially observed epidemic. This problem is motivated by the challenge of tracking the spread of COVID-19 (or another infectious disease) at a fine-grained level, where noise in the detection of infections via testing begins to drown out the signal. We introduce a Bayesian model which places a Gaussian process prior over the time-varying reproduction number R_t. The model explicitly includes the process by which infections are observed, allowing for a great deal of flexibility in specifying the observation distribution. For example, it can encompass an arbitrary distribution of delays between infection and detection, varying sensitivity of different kinds of tests over the course of infection, or particular sampling designs (e.g., cross-sectional or longitudinal sampling of people to be tested). To accommodate this flexibility, we develop a scalable stochastic variational inference strategy which combines a modified variational bound, the ability to differentiate through portions of the model of disease transmission, and the ability partially marginalize out some sources of randomness. We find that the ability to capture partial observability is crucial in small-sample settings, where standard methods become unreliable, delivering inaccurate and poorly calibrated predictions. By contrast, our method retains better accuracy and strong calibration properties even when observations are noisy.

Part I

Networks and health

Chapter 1

Exploratory Influence Maximization on Unknown Networks

In contexts ranging from health [VP07, RTC$^+$12] to international development [BCDJ13], practitioners have used the social network of their target population to spread information and change behavior. While previous work has delivered computationally efficient algorithms for this *influence maximization* problem [KKT03, CWW10, BBCL14, TXS14], this literature has had little interaction with the use of social network interventions in socially impactful domains. Part 1 of this thesis focuses on the design of algorithms for targeting network interventions which respond specifically to the challenges of community health settings. This work was motivated specifically by an application to HIV prevention for youth experiencing homelessness, and later chapters present a pilot test and larger-scale trial of an algorithm in this domain. The first several chapters of the thesis lay the algorithmic groundwork for this effort, focusing on theoretical analysis of the key technical challenges introduced by real-world domains.

One common challenge is that the network is not initially known and must be gathered via laborious field observations. For example, collecting network data from vulnerable populations such as homeless youth, while crucial for health interventions, requires significant time spent gathering field observations [RTC$^+$12]. Social media data is often unavailable when access to technology is limited, for instance in developing countries or with vulnerable populations. Even when such data is available, it often includes many weak links which are not effective at spreading influence [BFJ$^+$12]. For instance, a person may have hundreds of Facebook friends whom they barely know. In principle, the entire network could be reconstructed via surveys, and then existing influence maximization algorithms applied. However, exhaustive surveys are very labor-intensive and often considered impractical [VP07]. For influence maximization to be relevant to many real-world problems, it must contend with limited *information* about the network, not just limited *computation*.

The major informational restriction is the number of nodes which may be surveyed to explore the network. Thus, a key question is: *how can we find influential nodes with a small number of queries?* We formalize this problem as *exploratory influence maximization* and seek a principled algorithmic solution, i.e., an algorithm which makes a small number of queries and returns a set of seed nodes which are approximately as influential as the globally optimal seed set. Existing field work uses heuristics, such as sampling some percentage of the nodes and asking them to nominate influencers [VP07]. To our knowledge, no previous work directly addresses this question from an algorithmic perspective (see related work).

We show that for general graphs, any algorithm for exploratory influence maximization may perform arbitrarily badly unless it examines almost the entire network. However, real world networks often have strong *community* structure, where nodes form tightly connected subgroups which are only weakly connected to the rest of the network [LLDM09]. Consequently, influence mostly propagates locally. Community structure has been used to develop computationally efficient influence maximization algorithms [WCSX10, CZP$^+$14]. Here, we use it to design a highly information-efficient algorithm. We make four contributions. *First*, we introduce exploratory influence maximization and show that it is intractable for general graphs. *Second*, we present the ARISEN algorithm, which exploits community structure to find influential nodes. *Third*, we show that ARISEN has strong empirical performance on an array of real world social networks. *Fourth*, we formally analyze ARISEN on graphs drawn from the Stochastic Block Model (SBM) [FW81], a widely studied model of community structure. We prove that it approximates the optimal influence if the entire network were known by querying only a *logarithmic* number of nodes in the network size.

1.1 Exploratory influence maximization

As a motivating example, consider a homeless youth shelter which wishes to spread HIV prevention information [RTC$^+$12]. The shelter would try to select the most influential peer leaders to spread information, but the youths' social network is not initially known. Constructing the network requires a laborious survey [RTC$^+$12]. Our motivation is to mitigate this effort by querying only a few youth. Such queries require much less time than the day-long training peer leaders receive. We now formalize this problem.

Influence maximization: The influence maximization problem [KKT03], starts with a graph $G = (V, E)$, where $|V| = n$ and $|E| = m$. We assume that G is undirected; social links are typically reciprocal [SPRG12]. An influencer selects K seed nodes, aiming to maximize the expected size of the resulting influence cascade. We assume that influence propagates according to the independent cascade model (ICM), the most prevalent model in the literature. Initially, all nodes are inactive except for the seeds. When a node activates, it independently activates each of its neighbors with probability q. q is often assumed to be the same for all edges [CWW10, KKT03, YCXJ$^+$16]. Let $f(S)$ denote the expected number of activated nodes with seed set $S \subseteq V$. The objective is to compute $\arg\max_{|S| \leq K} f(S)$.

Local information: The edge set E is not initially known. Instead, the algorithm explores portions of the graph using local operations. We use the popular "Jump-Crawl" model [BK10], where the algorithm may either jump to a uniformly random node, or crawl along an edge from an already surveyed node to one of its neighbors. When visited, a node reveals all of its edges. We say that the *query cost* of an algorithm is the total number of nodes visited using either operation. Our goal is to find influential nodes with a query cost that is much less than n, the total number of nodes.

Stochastic Block Model: In our theoretical analysis, we assume that the graph is drawn from the stochastic block model (SBM), which provides a formal setting in which to analyze graphs with community structure. The SBM originated in sociology [FW81] and lately has been intensively studied in computer science and statistics (see e.g. [AS15, KMM$^+$13, MNS15]). In the SBM, the network is partitioned into disjoint communities $C_1....C_L$. Each within-community edge is present independently with probability p_w and each between-community edge is present independently with probability p_b. Recall that the Erdős-Rényi random graph $\mathcal{G}(n, p)$ is the graph on n nodes where every edge is independently present with probability p. In the SBM, community C_i is internally drawn as $\mathcal{G}(|C_i|, p_w)$ with additional random edges to other communities. While the SBM is a simplified model, our experimental results show that ARISEN also performs well on real-world graphs. ARISEN takes as input the parameters n, p_w, and p_b, but is not given any prior information about the realized draw of the network. It is reasonable to assume that the model parameters are known since they can be estimated using existing network data from a similar population (in our experiments, we show that this approach works well). For instance in HIV prevention, homeless youth social networks have been shown to exhibit community structure and several studies have gathered networks from which to infer p_w and p_b [YCXJ$^+$16, RTC$^+$12].

Our theoretical analysis will use a particular range of values for p_w and p_b. As formally defined, the SBM encompasses a wide range of possible topologies, depending on how the parameters p_w and p_b are set. Figure 1.1 gives a few examples, ranging from the a bipartite graph to an Erdős-Rényi graph. The community-structure graph that we intend to model is Figure 1.1(a). We later define a parameter range which produces such networks.

Objective: We compare to the globally optimal solution, i.e, the best performance if the entire network were known. Let $f_E(S)$ give the expected number of nodes influenced by seed set S when the set of realized edges are E. Let $\mathcal{A}(E)$ be the (possibly random) seed set containing our algorithm's selections given edge set E. Let OPT be the expected value of the globally optimal solution which seeds K nodes. We aim to prove that $\mathbb{E}[f_E(\mathcal{A}(E))] \geq \alpha OPT$ for some approximation ratio α, where the expectation is over the randomness in the graph, the algorithm's choices, and the ICM.

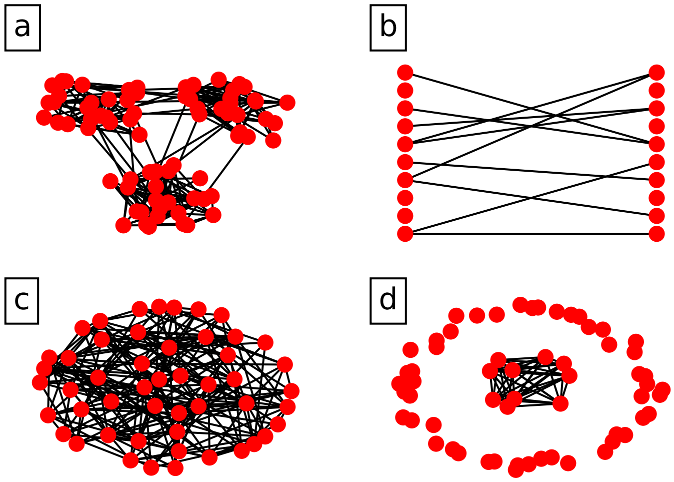
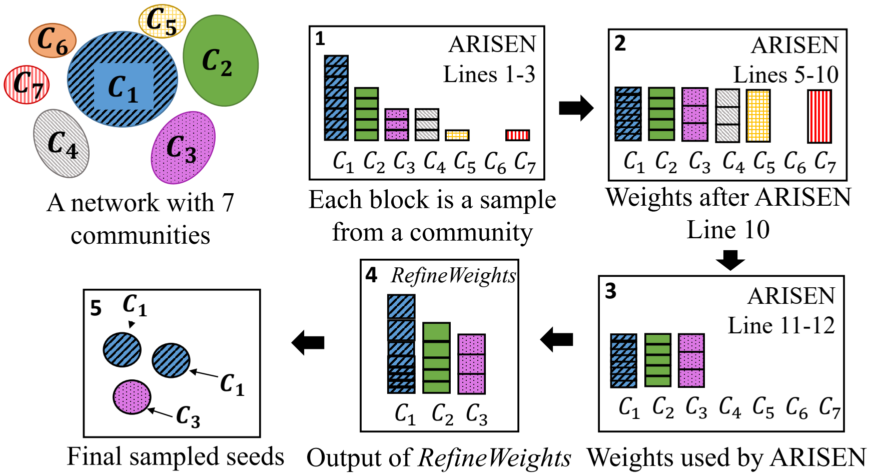


Figure 1.1: *Example SBM networks. (a) A community structured network (*$p_w = 0.1$*,* $p_b = 0.005$*). (b) A bipartite graph (2 communities,* $p_w = 0$*,* $p_b = 0.1$*). (c) An Erdős-Rényi graph (1 community,* $p_w = 0.2$*). (d) One small community with* $p_w = 1$*; the rest in a community with* $p_w = 0, p_b = 0$*.*

1.2 Related work

First, Yadav et al. [YCXJ$^+$16] and Wilder et al. [WYI$^+$17a] studied dynamic influence maximization over a series of rounds. Some edges are "uncertain" and are only present with some probability; the algorithm can gain information about these edges in each round. However, most edges are known in advance. By contrast, our work does not require *any* known edges. Mihara et al. [MTO15] also consider influence maximization over a series of rounds, but in their work the network is initially unknown. In each round, the algorithm makes some queries, selects some seed nodes, and observes all of the nodes which are activated by its chosen seeds. The ability to observe activated nodes makes our problem incomparable with theirs because activations can reveal a great deal about the network and give the algorithm information that even their benchmark does not have. Further, activations are unobservable in many domains (e.g. medical ones) for privacy and legal reasons. Carpentier and Valko [CV16] study a bandit setting where the algorithm does not know the network but observes the number of activations at each round. However, in applications of interest (e.g., HIV prevention) it is not feasible to conduct many low-reward



C_1
C_2
C_3
C_4
C_5
C_6
C_7
A network with 7 communities
1
ARISEN Lines 1-3
C_1 C_2 C_3 C_4 C_5 C_6 C_7
Each block is a sample from a community
2
ARISEN Lines 5-10
C_1 C_2 C_3 C_4 C_5 C_6 C_7
Weights after ARISEN Line 10
3
ARISEN Line 11-12
C_1 C_2 C_3 C_4 C_5 C_6 C_7
Weights used by ARISEN
4 *RefineWeights*
C_1 C_2 C_3
Output of *RefineWeights*
5
C_1
C_1
C_3
Final sampled seeds

Figure 1.2: *Example run of ARISEN with* $K = 3$ *(explained further in text). Each block is one sample, with current weight proportional to its height (e.g., in Frame 2,* C_5 *has one sample with very high weight).*

trial campaigns.

Another line of work concerns local graph algorithms, where a local algorithm only uses the neighborhoods around individual nodes. Borgs et al. [BBC+12] study local algorithms for finding the root node in a preferential attachment graph and for constructing a minimum dominating set. Other work aims to find nodes with high PageRank using local queries [BPP13, BBCT14]. These algorithms are not suitable for our problem since a great deal of previous work has observed that seeding high PageRank nodes can prove highly suboptimal for influence maximization [KSNM09, CWW10, JHC12]. Essentially, PageRank identifies a set of nodes that are *individually* central, while influence maximization aims to find a set of nodes which are *collectively* best at diffusing information. We also emphasize that our technical approach is entirely distinct from work on PageRank. Lastly, Alon et al. [AFLT15]. attempt to infer a ground truth from the opinions of agents with an unknown social network, a different task from ours with correspondingly distinct techniques.

1.3 Hardness result

We seek algorithms whose query cost grows slowly with n. The following shows that no algorithm with strictly sublinear query cost obtains a constant factor approximation for general graphs. The notation $o(1)$ refers to a term which goes to 0 as $n \to \infty$.

Theorem 1. *There exists a family of graphs on which any algorithm with query cost $O(n^{1-\epsilon})$ for some $\epsilon > 0$ has approximation ratio no better than $o(1)$.*

Proof. Consider a family of graphs which consist of a clique on $\log n$ nodes along with $n - \log n$ isolated nodes. Let $q = 1$ and $K = 1$. The algorithm gets influence $\log n$ if it selects a node in the clique, and influence 1 otherwise. The probability it ever samples the clique is at most $1 - (1 - \frac{\log n}{n})^{O(n^{1-\epsilon})} \leq 1 - e^{-\frac{\log n}{O(n^{\epsilon})}}(1 - \frac{\log^2 n}{n})^{O(n^{\epsilon})} = o(1)$. Hence, its expected influence is $o(1)\log n + 1$, while OPT is $\log n$, giving approximation ratio $\frac{o(1)\log n + 1}{\log n} = o(1)$. □

The general impossibility of sample-efficient algorithms motivates our focus on graphs with community structure, as formalized in the stochastic block model.

1.4 The ARISEN algorithm

We now introduce our main contribution, the ARISEN algorithm (*Approximating with Random walks to Influence a Socially Explored Network*). At a high level, ARISEN aims to leverage community structure in a network by choosing a set of K seed nodes which cover the largest K communities in the network. Figure 1 shows an example, explained in detail later. The idea behind ARISEN (Algorithm 1) is to sample a set of T random nodes $\{v_1 ... v_T\}$ from G and explore a small subgraph H_i around each v_i by taking R steps of a random walk (Lines 1-3). R and T are inputs; Section 1.5 gives settings which obtain theoretical guarantees. Intuitively, T should be greater than K (the number of seeds) so we can be sure of sampling each of the largest K communities. R is the number of steps taken on the random walk, chosen to ensure that enough samples are taken to estimate the average degree of a community accurately. The subgraphs H_i are used to construct a weight vector $\boldsymbol{w}$ where w_i gives the weight associated with v_i (Lines 10-12). The algorithm then independently samples each seed from $\{v_1 ... v_T\}$ with probability proportional to $\boldsymbol{w}$ (Line 13).

We first formalize the objective that ARISEN optimizes, which is a lower bound on its true influence. Let $f(X, C_i)$ denote the influence of seed set X on the subgraph C_i and

Algorithm 1 ARISEN(R, T)

```
 1: for $i = 1...T$ do
 2:     Sample $v_i$ uniformly random from $G$.
 3:     $H_i = R$ nodes on a random walk from $v_i$, after discarding the first $B$ nodes.
 4: end for
 5: for $i = 1...T$ do
 6:     Add $geometric\left(\frac{d(u)}{\Delta - d(u)}\right)$ copies of each node $u \in H_i$ to $H_i$
 7:     $\hat{d} = \frac{1}{R}\sum_{u \in H_i} d(u)$
 8:     $\hat{S}_i = \frac{\hat{d} - p_b n}{p_w - p_b}$
 9: end for
10: $w_j = \frac{n}{\hat{S}_j T}$.
11: $\tau = \max\{\hat{S}_j | \sum_{\{i | \hat{S}_i \geq \hat{S}_j\}} w_i \geq K\}$
12: For any $j$ with $\hat{S}_j < \tau$, set $w_j = 0$.
13: Sample $u_1...u_K \overset{iid}{\sim} \boldsymbol{w'}$
14: return $u_1...u_K$
```

$g(X) = \sum_{i=1}^{L} f(X, C_i)$, i.e., the influence spread within each community without considering between-community edges. ARISEN aims to optimize $\mathbb{E}[g(X)]$. Note that $f(X, G) \geq g(X)$ always holds. When p_b is low and little influence spreads between communities (which is the case that we study), g is a good proxy for the true influence. We now explain ARISEN in detail, and how it optimizes the surrogate objective g. Our focus on g is justified in Section 1.5, where we bound the gap between $\mathbb{E}\left[g(X)\right]$ and OPT.

In the SBM, each community C_i has expected average degree $d_i = |C_i|p_w + (n - |C_i|)p_b$. Solving for $|C_i|$, we can estimate the size of the community from its average degree. Since we do not have direct access to d_i, ARISEN estimates d_i (and hence $|C_i|$) using the nodes sampled in the random walk (Lines 7-8); we discard the first B nodes in this sampling to avoid biasing the estimate. Since a random walk is biased towards high degree nodes, we simulate the addition of a large number of self-loops at each node (Line 6). In particular, we simulate the addition of $\Delta - d(u)$ self-loops at every node, where Δ is a bound on the largest degree in the graph (for the stochastic block model, approximately $O(\log n)$). This means that the final set of samples are drawn according to a random walk on a *regular* graph with the same vertices, which ensures that in the stationary distribution each node has equal probability of being visited. Effectively, this counteracts the bias of a random walk towards high-degree nodes by adding extra copies of low-degree nodes to the set of samples.

In order to choose seed nodes using the estimated sizes, a natural idea would be to choose the K samples with the largest estimated size. However, this fails because large communities are sampled more often and will be seeded many times, which is redundant. E.g., in the example in Figure 13.2, placing all of the seeds in C_1 would be suboptimal compared to also seeding C_2. The difficulty is that using local information, we will not know which samples belong to the same community. One solution is to weight each sample *inversely* to its size (Line 6), and then sample seeds with probability proportional to the weights. This evens out the sampling bias towards large communities. Using weighted sampling gives us a principled way to prioritize samples and facilitates later steps which tune the weights to improve performance. In Figure 13.2, all communities have total weight of 1 after inverse weighting (Frame 2).

Next, the weights are truncated so that only the largest K communities receive nonzero weight (Line 7). After this step, the largest K communities have weight 1 and all smaller communities have weight 0 (at least approximately, due to sampling errors). For example, Frame 3 of Figure 13.2 shows that only C_1, C_2 and C_3 have nonzero weight. Suppose that we draw K seeds using the resulting weights. In each draw, each of the top K communities is seeded with probability approximately $\frac{1}{K}$. Thus, the cumulative probability that each is seeded is nearly $1 - \left(1 - \frac{1}{K}\right)^K \geq 1 - 1/e$. This reasoning is formalized in our theoretical guarantees.

1.5 Theoretical analysis

This section provides the main theoretical guarantees for ARISEN on graphs drawn from the stochastic block model. We start with assumptions and preliminaries which formalize the graph structure we study. Since the generic block model can capture a wide range of behaviors, we must place some restrictions on p_w and p_b to model real-world networks. While it is often possible to prove approximation guarantees for ARISEN in other settings, we focus on a particular parameter range which produces networks with community structure.

First, we assume that each of the top K communities occupies a constant fraction of the network:

Assumption 1. For each $i = 1...K$, $|C_i| > cn$ for some constant $c > 0$ independent of n.

Next, we assume that each community is internally connected:

Assumption 2. For all communities C_i, it holds that $p_w > \frac{\log |C_i|}{|C_i|}$ and $p_w = O\left(\frac{\log |C_i|}{|C_i|}\right)$.

Here, $\frac{\log |C_i|}{|C_i|}$ is the threshold above which an Erdős-Rényi graph is connected with high probability [JLR11]. Below the threshold, a constant fraction of the nodes lie outside the largest component. Next, we require that between-community edges are limited:

Assumption 3. $p_b < \frac{1}{n}$.

This assumption ensures that the between-community edges do not themselves form a large (constant-size) connected component in G. This assumption is necessary for us to be able to use average degrees to estimate the size of a community; otherwise, between-community edges begin to interfere with the estimates. Finally, we require that it is possible to start a large influence cascade within each community:

Assumption 4. For all $i = 1...K$, $p_w q |C_i| > 1$.

This implies that it is possible for an influence cascade to reach a linear portion of the community. Otherwise, if $p_w q |C_i| < 1$, at most $O(\log |C_i|)$ nodes can be influenced by any constant number of seeds (via Lemma 16). We focus on when it is possible for influence maximization to have large results, not when only a vanishingly small fraction of nodes can possibly be reached.

We now state some helpful preliminaries before providing an overview of the analysis. We often use the following connection between the joint behavior of the SBM/ICM on the one hand, and the connected components of an Erdős-Rényi random graph on the other. The ICM can be seen as removing each edge independently with probability $1 - q$. A node is influenced if afterwards it lies in the same connected component as a seed node [KKT03]. Since each community is itself an Erdős-Rényi graph, the connected components induced by the ICM in each community are distributed exactly as those in an Erdős-Rényi graph with connection probability $p_w q$. A well-known result characterizes the component sizes:

Lemma 1 ([JLR11]). *Consider the Erdős-Rényi graph $\mathcal{G}(n, p)$. If $np < 1$, then with probability $1 - o(1)$, its largest connected component has size at most $\frac{3}{(1-np)^2} \log n$. If $np > 1$, then with probability $1 - o(1)$, its largest component has size $(1 + o(1))\beta n$. β is the solution to $\beta + e^{-\beta np} - 1 = 0$.*

We denote by $\beta(x)$ the fraction of nodes contained in the largest connected component of $\mathcal{G}(x, p_w q)$ (assuming that $x p_w q > 1$ and the event in Lemma 16 occurs). $\beta(|C_i|)$ gives the fraction of C_i that can be reached by a cascade.

We now introduce two quantities which appear in the approximation ratio that we show. First, define

$$\bar{\beta} = \frac{1}{K} \sum_{i=1}^{K} \beta(|C_i|)$$

to be the average fraction of the largest K communities which is reached by an influence cascade. Second, let G_{comm} be the graph G with all between-community edges removed and define

$$\gamma = \frac{OPT(G_{comm})}{OPT(G)}$$

to be the fraction of the optimal influence spread on G which is attainable on G_{comm}. γ measures the strength of the community structure which ARISEN leverages; $\gamma = 1$ when the communities are entirely disconnected while γ decreases as they become less distinct. Given these quantities, the following is our main theoretical result:

Theorem 2. *For any $\epsilon < \frac{1}{K}$, ARISEN can be implemented using $O\left(\frac{1}{\epsilon^4} \log(n) \log^2\left(\frac{1}{\epsilon}\right) \log\log\left(\frac{1}{\epsilon}\right)\right)$ samples with approximation ratio*

$$\left(1 - \frac{1}{e} - \epsilon - o(1)\right) \cdot \bar{\beta} \cdot \gamma.$$

This is obtained by setting $T = O\left(\frac{1}{\epsilon^2} \log \frac{1}{\epsilon}\right)$ and $O\left(\frac{1}{\epsilon^2} \log(n) \log\left(\frac{1}{\epsilon}\right) \log\log\left(\frac{1}{\epsilon}\right)\right)$.

For the number of samples, T (the number of outer-loop samples to take) is set so that each community is sampled a number of times which closely approximates its true proportion of the network size. R is set so that each random walk has sufficient time to mix and return an accurate estimate of the average degree (and hence the size of the community). For the approximation ratio, the first term is nearly $1 - 1/e$, up to error terms which decrease as n becomes large and ϵ small. We show that each of the top K communities is seeded with probability close to $1 - 1/e$. The proof tracks the intuition outlined when ARISEN was described: each community receives total weight close to 1, giving it probability close to $\frac{1}{K}$ of being hit by each of K seeds.

The second term, $\bar{\beta}$, is the average fraction over the top K communities which can be influenced by a seed node (via Assumption 2(a)). These nodes form a giant connected component under the ICM. Consider a given seed node u_i, which is a uniformly random node in some C_i. With probability $\beta(|C_i|)$, u_i lies in this component; hence it influences at least $\beta(|C_i|)^2|C_i|$ nodes in expectation. The best that OPT can do is to influence the entire connected component with certainty, giving an influence spread of $\beta(|C_i|)|C_i|$. The ratio between these terms is $\frac{\beta(|C_i|)^2|C_i|}{\beta(|C_i|)|C_i|} = \beta(|C_i|)$, which we show can be approximated across the top K communities by $\bar{\beta}$. Essentially, $\bar{\beta}$ expresses the difficulty of finding influential nodes within each community and increases as the product $\mu p_w q$ becomes larger.

Finally, γ captures the extent to which seeding the largest K communities is a successful strategy; i.e., whether most influence diffuses via within-community edges. This is because OPT may leverage between-community influence propagation but in the worst case ARISEN may succeed in using only within-community edges. We next show that γ is bounded by a constant for stochastic block model networks with limited between-community influence spread:

Theorem 3. *Let $c_{max} = \max_i p_b q \cdot (n - |C_i|)|C_i|$ be the maximum number of expected influence transmission events into any community and $\mu = \frac{1}{K}\sum_{i=1}^{K} \beta(|C_i|)|C_i|$ be the average size of the giant component induced by the ICM in the top K communities. Suppose that $c_{max} < 1$. Then,*

$$\gamma \geq \frac{1 - c_{max}}{12 \log \frac{n}{\mu}} = \Theta(1).$$

Combined with Theorem 2, this ensures a constant-factor approximation guarantee. Intuitively, the condition $c_{max} < 1$ ensures that it is not possible to start a cascade reaching most of the communities in the network by seeding a single community. While it is clearly possible to give guarantees for this case (even for choosing seeds completely at random), our analysis focus on the more challenging case when influence spreads mostly within communities.

1.6 Practical improvements

The seeding strategy used by ARISEN suffices to obtain the approximation guarantee proved below and is the best possible for some networks. However, equal division of the seed nodes over the largest K communities is overly pessimistic in other cases, such as when some communities are much larger than others. In such cases, it would be better to focus more seed nodes on large communities. RefineWeights tunes the weights $\boldsymbol{w}$ to account for such scenarios. In essence, RefineWeights tries to exploit easier cases where some communities are much larger than others by producing new weights $\boldsymbol{w}'$. The final set of seed nodes can then be sampled iid from $\boldsymbol{w}'$.

RefineWeights (Algorithm 2) starts in Line 2 by defining v_i to be the most influential node in the sampled subgraph H_i (instead of the random starting node). Lines 5-11 successively modify each element of $\boldsymbol{w}$. Starting with the weights corresponding to the highest-value communities, RefineWeights asks whether g would be increased by doubling the w_i under consideration (Line 7). If yes, we set $w_i = 2w_i$ and ask if it can be doubled again. If no, RefineWeights performs a binary search between w_i and $2w_i$ to find the best setting (Line 10). Then, it moves on to the weight corresponding to the next community. In the example in Figure 13.2, Frame 4 shows that the weights of samples from C_1 and C_2 have been increased. Each change is made only if it improves g, so we have:

Proposition 1. Let $\boldsymbol{w}$ the weight vector used in ARISEN and $\boldsymbol{w}'$ be the output of RefineWeights. Then, $\mathbb{E}_{X\sim\boldsymbol{w}'}\left[g(X)\right] \geq \mathbb{E}_{X\sim\boldsymbol{w}}\left[g(X)\right]$.

The key difficulty is determining if each modification increases g. In the EstVal procedure, we provide a way to estimate g using only local knowledge:

Algorithm 2 RefineWeights

```
1: for $i = 1...T$ do
2:     $v_i = \arg\max_{v \in H_i} f(v, H_i)$
3: end for
4: $\boldsymbol{w}' = \boldsymbol{w}$
5: sort $\boldsymbol{w}'$ in increasing order by $f(v_i, H_i)$
6: for $i = 1...T$ do
7:     while $\textsc{EstVal}(2w'_i, w_{-i}) > \textsc{EstVal}(\boldsymbol{w}')$ do
8:         $w'_i = 2w'_i$
9:     end while
10:    $w'_i = \text{BinarySearch}(w'_i, 2w'_i)$
11: end for
```

Proposition 2. $\textsc{EstVal}(\boldsymbol{w}) = \mathbb{E}_{X \sim \boldsymbol{w}}\left[g(X)\right]$

We give the main idea here; see the appendix for a proof and pseudocode for EstVal. Take any seed set X. Note that the influence within each C_i depends only on nodes in $X \cap C_i$, which we write as X_{C_i}. So, g can be rewritten as $g(X) = \sum_{i=1}^{L} \mathbb{E}[f(X_{C_i}, C_i)]$. If we knew X_{C_i}, then we could calculate $\mathbb{E}\left[f(X_{C_i}, C_i)\right]$ by simulating draws from the SBM for the unobserved portions of C_i. Concretely, let H_i be the subgraph observed in community C_i, with estimated size $\hat{S}_i$. We simulate the rest of C_i by adding $\hat{S}_i - |H_i|$ new nodes, with edges between them and H_i randomly generated from the SBM. This is sufficient to choose the best seed within H_i, as in Line 2. For Line 7, we need to estimate g. The obstacle is not knowing which of the $v_1 ... v_T$ lie in the same community (since a node will contribute less influence if there is another seed from the same community). However, we do know (approximately) how many other times each community is sampled, and the (approximate) weight that those samples will receive, so g can be estimated by averaging some careful simulations. Via a standard Hoeffding bound (Kempe et al. 2015), $O(\frac{n^2}{\epsilon^2} \log \frac{1}{\epsilon})$ simulations per EstVal call guarantee error ϵ with high probability.

1.7 Experiments

We now present experiments assessing ARISEN's empirical performance. We make two modifications to ARISEN to improve its practical performance. First, we set $B = 0$ and $\Delta = 0$;

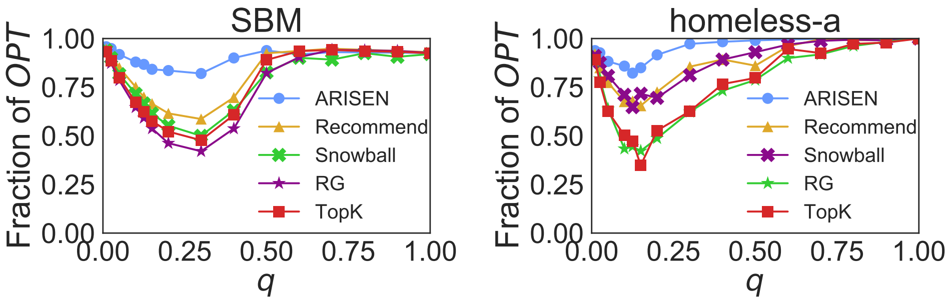


Figure 1.3: *Influence compared to OPT as q varies.*

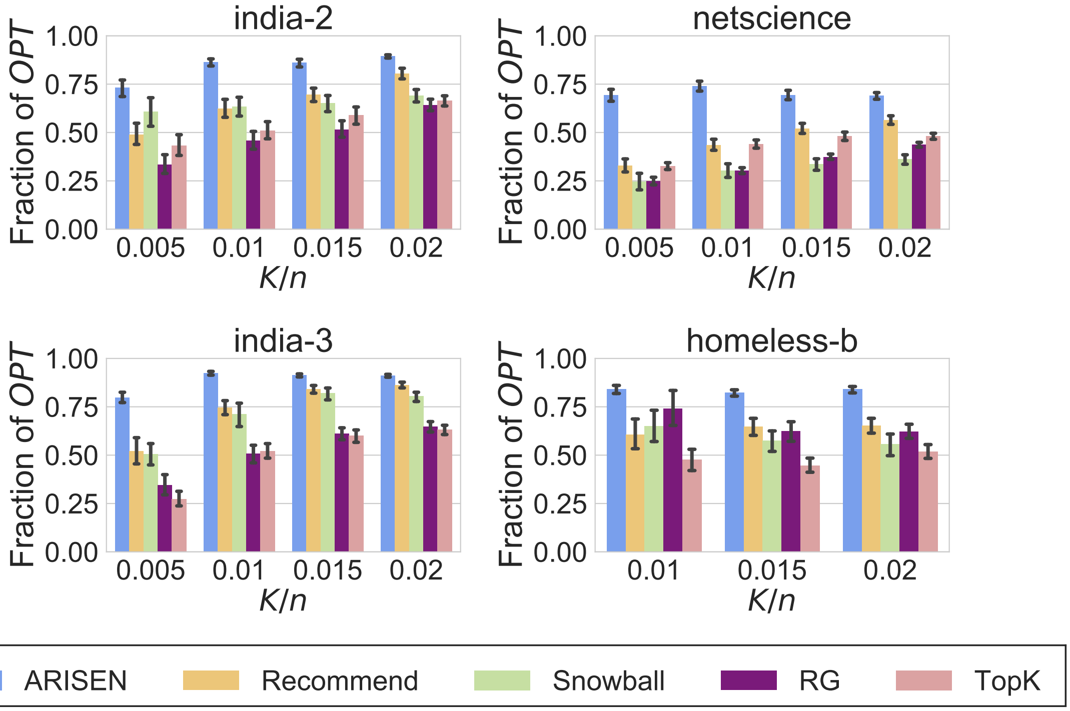


Figure 1.4: *Influence spread compared to OPT as K varies with* $q = 0.15$.

i.e., skipping the burn-in and degree correction steps. While these are necessary to obtain general theoretical guarantees, we find that in practice it is preferable to accept some bias in the estimates in return for reduced variance and fewer queries. Second, we warm-start RefineWeights by proposing $\boldsymbol{w'} = \exp(\boldsymbol{w})$ as the initialization since we observed that large weights are often doubled several times.

We compare ARISEN with an array of baselines on several networks. We focus on networks with about 100-1000 nodes because this is the size of real-world social groups of interest to us. The first network is *homeless*: Two networks (a and b) gathered from homeless youth in Los Angeles and used to study HIV prevention with 150-200 nodes each. Second, *india*: Three networks of the household-level social contacts of villages in rural India. Gathered by Banerjee et al. [BCDJ14] to study diffusion of information about microfinance programs, with 250-350 nodes each. Third, *netscience*[1]: a collaboration network of network science researchers with 1461 nodes. Fourth, *SBM*: a synthetic SBM graphs with 1000 nodes. There are 10 communities with size from 350 to 30 nodes ($p_w = 6 \cdot 10^{-3}$, $p_b = 2 \cdot 10^{-5}$). We approximate the optimal value by running TIM [TXS14], a state of the art influence maximization algorithm, on each full network. For each real network, p_w and p_b are estimated from a different network in the same category (for netscience, we use another collaboration network, astro-ph[1]). For SBM, we use another network from the same distribution. We present a cross-section of results across the datasets but the general trends hold for all networks. Exhaustive results are in the appendix.

We consider four benchmarks. First, *random greedy* (RG). RG uses the same query budget as ARISEN, but queries nodes uniformly randomly. It then runs TIM on the graph composed of the edges these queries reveal. Hence, RG uses a sophisticated seed selection technique, but not ARISEN's sampling procedure. Second, *TopK*. TopK uses ARISEN's random walk sampling (lines 1-3), but seeds the K samples with highest estimated community size. RG and TopK jointly test the importance of ARISEN's sophisticated methods for sampling the network and selecting seed nodes, respectively. Third, *recommend*, which for each of the

[1] http://www-personal.umich.edu/ mejn/netdata/

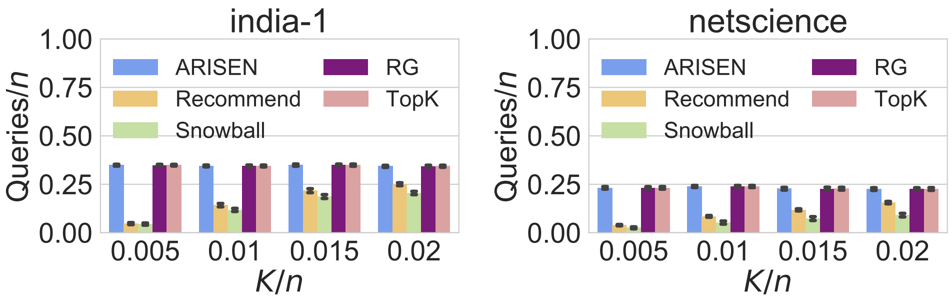


Figure 1.5: *Query complexity as K varies.*

Table 1.1: *ARISEN's % influence gain with 25% fewer seeds.*

Network/baseline	Rec.	Snowball	RG	TopK
homeless-a	24.2	9.9	20.7	91.1
india-1	0.03	6.6	25.7	29.1
netscience	4.8	63.9	35.4	43.4

K nodes, first queries a random node and then seeds their highest degree friend. Fourth, *snowball*, which starts from a random node and seeds that node's highest degree neighbor. It then seeds the highest degree neighbor of the first seed, and so on. Recommend and snowball are the most common strategies in the field [VP07].

Figure 3 shows that ARISEN obtains substantially higher influence spread than the baselines, often exceeding the best baseline by 20-50%. The x axis varies q. Each point gives the fraction of OPT achieved for that q, averaged over 50 runs. E.g., the point at $q = 0.2$ for SBM indicates that ARISEN's value was $0.8 \cdot OPT$. We take $K = 0.01n$, focusing on when few seeds are available (as in previous work [CWW10]). All differences ($q \in [0.01, 0.7]$) are statistically significant (t-test, $p < 10^{-7}$). The gap between ARISEN and the baselines is particularly high in the difficult case of small but nonzero q. When q is close to 0, all algorithms perform close to OPT since little is possible. When q is very high, influence maximization is easy and nearly any algorithm performs well [CWW10]. Thus, Figure 1.4 presents results where K is varied with $q = 0.15$ fixed (since this is the hard case). We see that ARISEN uniformly outperforms the baselines, particularly when K is small. As K becomes larger, the baselines improve (again because the problem becomes easier). However, they are still outperformed by ARISEN.

In particular, we conclude from RG's poor performance that ARISEN's random walk based query scheme substantially improves on uniformly sampling an equivalent number of nodes. The comparison with TopK confirms that ARISEN's weighted seed selection is also necessary since simply seeding the largest communities does poorly. In combination, this demonstrates that ARISEN's major elements are both needed to ensure good empirical performance.

Figure 1.5 examines each algorithm's query cost (each selects the same number of seeds). The appendix lists R and T values; here we just focus on the total queries. ARISEN uses more queries than recommend and snowball, and an equal number to RG and TopK. However, recommend and snowball use more queries as K increases, with query cost close to ARISEN for $K = 0.02 \cdot n$. ARISEN's query cost is uniformly in the range $0.20 \cdot n - 0.35 \cdot n$, a relatively small portion of the network in absolute terms. This query budget is justified by ARISEN's larger influence spread, which makes more efficient use of seed nodes. Intervening to seed a node is often much more costly than querying its edges, as in the HIV domain where an intervention is a day-long class. Table 1.1 shows the percent by which ARISEN's influence spread exceeds each baseline when the baseline uses $K = 0.02 \cdot n$ but ARISEN uses 25% fewer seeds. ARISEN outperforms all of the baselines, often by over 20%. Hence, ARISEN delivers higher influence with fewer costly seeds.

1.8 Conclusion

Optimizing the design of network interventions when information about the network structure is costly to acquire is a challenging algorithmic problem with many applications. This chapter investigated one example of this problem, focused on influence maximization for community-structured networks. In this setting, both strong theoretical guarantees and promising empirical performance can be obtained by an algorithm which uses only a small amount of local information. Many challenges remain for future work, e.g., the development of algorithms which can leverage alternate forms of structure, or sampling schemes tailored to other optimization problems.

Chapter 2

Robust submodular optimization

Submodular functions are ubiquitous due to wide-spread applications ranging from machine learning, to viral marketing, to mechanism design. Intuitively, submodularity captures diminishing returns (formalized later). One application of submodularity particularly relevant to this thesis is the *influence maximization* problem of selecting the most influential set of k nodes from a graph. In this chapter, we develop the underlying algorithmic techniques to optimize the worst case over a set of submodular functions, due to either uncertainty about the correct objective function or the presence of an explicitly adversarial actor. In order to demonstrate the generality of this framework, applications are shown in other areas besides influence maximization, while later chapters will apply these techniques to the problem of influence maximization under uncertainty in the context of HIV prevention.

As a running example for this chapter, consider the network security game introduced by Tsai et al. [TYK$^+$10]. A defender can place checkpoints on k edges of a graph. An attacker aims to travel from a source node to any one of several targets without being intercepted. Each player has an exponential number of strategies since the defender may choose any set of k edges and the attacker may choose any path. Hence, previous approaches to computing the optimal defender strategy were either heuristics with no approximation guarantee, or else provided guarantees but ran in worst-case exponential time [JKV$^+$11, IOAI16].

However, this game has useful structure. The defender's best response to any attacker mixed strategy is to select the edges which are most likely to intersect the attacker's chosen path. Computing this set is a submodular optimization problem [JCT13]. We give a general algorithm for computing approximate minimax equilibria in zero-sum games where the maximizing player's best response problem is a monotone submodular function. Our algorithm obtains a $(1-\frac{1}{e})^2-$approximation (modulo an additive loss of ϵ) to the maximizing player's minimax strategy. This algorithm runs in pseudopolynomial time *even when both action spaces are exponentially large* given access to a weakened form of a best response oracle for the adversary. Pseudopolynomial means that the runtime bound depends polynomially on largest value of any single item (which we expect to be a constant for most cases of interest). Our algorithm approximately solves a non-convex, non-smooth continuous extension of the problem and then rounds the solution back to a pure strategy in a randomized fashion. To our knowledge, no subexponential algorithm was previously known for this problem with exponentially large strategy spaces. Our framework has a wide range of applications, corresponding to the ubiquitous presence of submodular functions in artificial intelligence and algorithm design (see Krause and Golovin [KG14] for a survey).

One prominent class of applications is *robust submodular optimization*. A decision maker is faced with a set of submodular objectives $f_1 ... f_m$. They do not know which objective is the true one, and so would like to find a decision maximizing $\min_i f_i$. Robust submodular optimization has many applications because uncertainty is so often present in decision-making. We start by studying the randomized version of this problem, where the decision maker may select a distribution over actions such that the worst case *expected* performance is maximized [KRG11, CLSS17, WYI$^+$17b]. This is equivalent to computing the minimax equilibrium for a game where one player has a submodular best response. Our techniques for solving such games also yield an algorithm for the deterministic robust optimization problem, where the decision maker must commit to a single action. Specifically, we obtain bicriteria approximation guarantees analogous to previous work [KMGG08] under significantly more general conditions.

We make three contributions. First, we define the class of *submodular best response*

(SBR) games, which includes the above examples. Second, we introduce the EQUATOR algorithm to compute approximate equilibrium strategies for the maximizing player. Third, we give example applications of our framework to problems with no previously known approximation algorithms. We start out by showing that network security games [TYK$^+$10] can be approximately solved using EQUATOR. We then introduce and solve the robust version of a classical submodular optimization problem: robust maximization of a coverage function (which includes well-known applications such as budget allocation and sensor placement). Finally, we experimentally validate our approach for network security games and robust budget allocation. We find that EQUATOR produces near-optimal solutions and easily scales to instances that are too large for previous algorithms to handle.

2.1 Problem description

Formulation: Let X be a set of items with $|X| = n$. A function $f : 2^X \to R$ is submodular if for any $A \subseteq B$ and $i \in X \setminus B$, $f(A \cup \{i\}) - f(A) \geq f(B \cup \{i\}) - f(B)$. We restrict our attention to functions that are *monotone*, i.e., $f(A \cup \{i\}) - f(A) \geq 0$ for all $i \in X, A \subset X$. Without loss of generality, we assume that $f(\emptyset) = 0$ and hence $f(S) \geq 0 \, \forall S$. Let $\mathcal{F} = \{f_1 ... f_m\}$ be a finite set of submodular functions on the ground set X. m may be exponentially large. Let $\Delta(S)$ denote the set of probability distributions over the elements of any set S. Oftentimes, we will work with independent distributions over X, which can be fully specified by a vector $\boldsymbol{x} \in R^n_+$. x_i gives the marginal probability that item i is chosen. Denote by $p^I_{\boldsymbol{x}}$ the independent distribution with marginals $\boldsymbol{x}$. Let $\mathcal{I}$ be a collection of subsets of X. For instance, we could have $\mathcal{I} = \{S \subseteq X : |S| \leq k\}$. We would like to find a minimax equilibrium of the game where the maximizing player's pure strategies are the subsets in $\mathcal{I}$, and the minimizing player's pure strategies are the functions in $\mathcal{F}$. The payoff to the strategies $S \in \mathcal{I}$ and $f_i \in \mathcal{F}$ is $f_i(S)$. We call a game in this form a *submodular best response* (SBR) game. For the maximizing player, computing the minimax equilibrium is equivalent to solving

$$\max_{p \in \Delta(\mathcal{I})} \min_{f \in \mathcal{F}} \mathop{\mathbb{E}}_{S \sim p} [f(S)] \tag{2.1}$$

where $S \sim p$ denotes that S is distributed according to p.

Example: network security games. To make the setting more concrete, we now introduce one of our example domains, the network security game of Tsai et al. [TYK$^+$10]. There is a graph $G = (V, E)$. There is a source vertex s (which may be a supersource connected to multiple real sources) and a set of targets T. An attacker wishes to traverse the network starting from the source and attack a target. Each target t_j has a value τ_j. The attacker picks a $s - t_j$ path for some $t_j \in T$. The defender attempts to catch the attacker by protecting edges of the network. The defender may select any k edges, and the attacker is caught if any of these edges lies on the chosen path. We use the normalized utilities defined by Jain et al. [JCT13], which give the defender utility $\tau_j > 0$ if an attack on t_j is intercepted and 0 if the attack succeeds. Thus, each path P from s to t_j for the attacker induces an objective function f_P for the defender: for any set of edges S, $f_P(S) = \tau_j$ if $S \cap P \neq \emptyset$, otherwise $f_P(S) = 0$. f_P is easily seen to be submodular [JCT13]. Hence, we have a SBR game with $\mathcal{I} = \{S \subseteq E : |S| \leq k\}$ and $\mathcal{F} = \{f_P : P \text{ is a path from } S \text{ to } T\}$.

Allowable pure strategy sets: Our running example is when the pure strategies $\mathcal{I}$ of the maximizing player are all size k subsets: $\mathcal{I} = \{S \subseteq X : |S| \leq k\}$. In general, our algorithm works when $\mathcal{I}$ is any matroid; this example is called the *uniform* matroid. We refer to [KVKV12] for more details on matroids. Here, we just note that matroids are a class of well-behaved constraint structures which are of great interest in combinatorial optimization. A useful fact is that any linear objective can be exactly optimized over a matroid by the greedy algorithm. For instance, consider the above uniform matroid. If each element j has a weight w_j, the highest weighted set of size k is obtained simply by taking the k items with highest individual weights. Let $k = \max_{S \in \mathcal{I}} |S|$ be the size of the largest pure strategy. E.g., in network security games k is the number of defender resources. In general, k is the rank of the matroid.

We now introduce some notation for the continuous extension of the problem. Let $\mathbf{1}_S$ be the indicator vector of the set S (i.e., an n-dimensional vector with 1 in the entries of elements that are in S and 0 elsewhere). Let $\mathcal{P}(\mathcal{M})$ be the convex hull of $\{\mathbf{1}_S : S \in \mathcal{I}\}$. Note that $\mathcal{P}(\mathcal{M})$ is a polytope.

Best response oracles: A best response oracle for one player is a subroutine which computes the pure strategy with highest expected utility against a mixed strategy for the other player. We assume that an oracle is available for the minimzing player. However, we require only a weaker oracle, which we call an *best response to independent distributions oracle* (BRI). A BRI oracle is only required to compute a best response to mixed strategies which are independent distributions, represented as the marginal probability that each item in X appears. Given a vector $\boldsymbol{x} \in R^n_+$, where x_i is the probability that element $i \in X$ is chosen, a BRI oracle computes $\arg\min_{f_i \in \mathcal{F}} \mathbb{E}_{S \sim p^I_x}[f_i(S)]$. We use $S \sim \boldsymbol{x}$ to denote that S is drawn from the independent distribution with marginals $\boldsymbol{x}$. As we will see later, sometimes a BRI oracle is readily available even when the full best response is NP-hard.

Robust optimization setting: One prominent application of SBR games is robust submodular optimization. Robust optimization models decision making under uncertainty by specifying that the objective is not known exactly. Instead, it lies within an uncertainty set $\mathcal{U}$ which represents the possibilities that are consistent with our prior information. Our aim is to perform well in the worst case over all objectives in $\mathcal{U}$. We can view this as a zero sum game, where the decision maker chooses a distribution over actions and nature adversarially chooses the true objective from $\mathcal{U}$. A great deal of recent work has been devoted to the setting of randomized actions, both because randomization can improve worst-case expected utility [DKW16], and because the randomized version often has much better computational properties [KRG11, OSU16]. Randomized decisions also naturally fit a problem setting where the decision maker will take several actions and wants to maximize their total reward. Any single action might perform badly in the worst case; drawing the actions from a distribution allows the decision maker to hedge their bets and perform better overall.

2.2 Previous work

We discuss related work in two areas. First, solving zero-sum games with exponentially large strategy sets. Efficient algorithms are known only for limited special cases. One approach is to represent the strategies in a lower dimensional space (the space of marginals). We elaborate more below since our algorithm uses this approach. For now, we just note that previous work [ADH$^+$16, Xu16, CJLBM16] requires that the payoffs be *linear* in the lower dimensional space. Linearity is a very restrictive assumption; ours is the first algorithm which extends the marginal-based approach to general submodular functions. This requires entirely different techniques.

In practice, large zero sum games are often solved via the *double oracle* algorithm [MGB03, BKLP14, BJTK15, HCP09]. Double oracle starts with each player restricted to only a small number of arbitrarily chosen pure strategies and repeatedly adds a new strategy for each player until an equilibrium is reached. The new strategies are chosen to be each player's best response to the other's current mixed strategy. This technique is appealing when equilibria have sparse support, and so only a few iterations are needed. However, it is easy to give examples where *every* pure strategy lies in the support of the equilibrium, so double oracle will require exponential runtime. Our algorithm runs in guaranteed polynomial time.

Second, we give more background on robust submodular optimization. Krause et al. [KMGG08] introduced the problem of maximizing the minimum of submodular functions, which corresponds to Problem 2.1 with the maximizing player restricted to pure strategies. They show that the problem is inapproximable unless P = NP. They then relax the problem by allowing the algorithm to exceed the budget constraint (a bicriteria guarantee). Our primary focus is on the *randomized* setting, where the algorithm respects the budget constraint but chooses a distribution over actions instead of a pure strategy. This randomized variant was studied by Wilder et al. [WYI$^+$17b] for the special case of influence maximization. Krause et al. [KRG11] and Chen et al. [CLSS17] studied general submodular functions using very similar techniques: both iterate dynamics where the adversary plays a no-regret learning algorithm and the decision maker plays a greedy best response. This algorithm maintains a variable for every function in $\mathcal{F}$ and so is only computationally tractable when $\mathcal{F}$ is small. By contrast, we deal with the setting where $\mathcal{F}$ is exponentially large. However, we lose an extra factor of $(1 - 1/e)$ in the approximation ratio.

We also extend our algorithm to obtain bicriteria guarantees for the deterministic robust submodular optimization problem (where we select a single feasible set). Our guarantees apply under significantly more general conditions than those of Krause et al. [KMGG08] but have weaker approximation guarantee; details can be found in the discussion after Theorem 7.

2.3 Preliminaries

We now introduce techniques our algorithm builds on.

Multilinear extension: We can view a set function f as being defined on the vertices of the hypercube $\{0,1\}^n$. Each vertex is the indicator vector of a set. A useful paradigm for submodular optimization is to extend f to a continuous function over $[0,1]^n$ which agrees with f at the vertices. The *multilinear extension* F is defined as

$$F(\boldsymbol{x}) = \sum_{S \subseteq X} f(S) \prod_{j \in S} x_j \prod_{j \notin S} 1 - x_j.$$

Equivalently, $F(\boldsymbol{x}) = \mathbb{E}_{S \sim \boldsymbol{x}}[f(S)]$. That is, $F(\boldsymbol{x})$ is the expected value of f on sets drawn from the independent distribution with marginals $\boldsymbol{x}$. F can be evaluated using random sampling [CCPV11] or in closed form for special cases [IJB14]. Note that for any set S and its indicator vector $\mathbf{1}_S$, $F(\mathbf{1}_S) = f(S)$. One crucial property of F is *up-concavity* [CCPV11]. That is, F is concave along any direction $\boldsymbol{u} \succeq \mathbf{0}$ (where $\succeq$ denotes element-wise comparison). Formally, a function F is *up-concave* if for any $\boldsymbol{x}$ and any $\boldsymbol{u} \succeq 0$, $F(\boldsymbol{x} + \xi\boldsymbol{u})$ is concave as a function of ξ.

Correlation gap: A useful property of submodular functions is that little is lost by optimizing only over independent distributions. Agrawal et al. [ADSY10] introduced the concept of the correlation gap, which is the maximum ratio between the expectation of a function over an independent distribution and its expectation over a (potentially correlated) distribution with the same marginals. Let $D(\boldsymbol{x})$ be the set of distributions with marginals $\boldsymbol{x}$. The correlation gap $\kappa(f)$ of a function f is defined as

$$\kappa(f) = \max_{x \in [0,1]^n} \max_{p \in D(\boldsymbol{x})} \frac{\mathbb{E}_{S \sim p}[f(S)]}{\mathbb{E}_{S \sim p_x^I}[f(S)]}.$$

For any submodular function $\kappa \leq \frac{e}{e-1}$. This says that, up to a loss of a factor $1 - 1/e$, we can restrict ourselves to independent distributions when solving Problem 2.1.

Swap rounding: Swap rounding is an algorithm developed by Chekuri et al. [CVZ10] to round a fractional point in a matroid polytope to an integral point. We will use swap rounding to convert the fractional point obtained from the continuous optimization problem to a distribution over pure strategies. Swap rounding takes as input a representation of a point $\boldsymbol{x} \in \mathcal{P}(\mathcal{M})$ as a convex combination of pure strategies. It then merges these sets together in a randomized fashion until only one remains. For *any* submodular function f and its multilinear extension F, the random set R satisfies $\mathbb{E}[f(R)] \geq F(\boldsymbol{x})$. I.e., swap rounding only increases the value of any submodular function in expectation.

2.4 Algorithm for SBR games

In this section, we introduce the EQUATOR (*EQUilibrium via stochAsTic frank-wOlfe and Rounding*) algorithm for computing approximate equilibrium strategies for the maximizing player in SBR games. Since the pure strategy sets can be exponentially large, it is unclear what it even means to compute an equilibrium: representing a mixed strategy may require exponential space. Our solution to this dilemma is to show how to efficiently *sample* pure strategies from an approximate equilibrium mixed strategy. This suffices for the maximizing player to implement their strategy. Alternatively, we can build an approximate mixed strategy with sparse support by drawing a polynomial number of samples and outputing the uniform distribution over the samples. In order to generate these samples, EQUATOR first solves a continuous optimization problem, which we now describe.

The marginal space: A common meta-strategy for solving games with exponentially large strategy sets is to work in the lower-dimensional space of *marginals*. I.e., we keep track of only the marginal probability that each element in the ground set is chosen. To illustrate this, let p be a distribution over the pure strategies $\mathcal{I}$, and $\boldsymbol{x} \in \mathcal{P}(\mathcal{M})$ denote a vector giving the marginal probability of selecting each element of X in a set drawn according to p. Note that $\boldsymbol{x}$ is n-dimensional while p could have dimension up to 2^n. Previous work has used marginals for linear objectives. A linear function with weights w satisfies $\mathbb{E}_{S\sim p}\left[\sum_{j\in S} w_j\right] = \sum_{j=1}^n w_j \Pr[j \in S] = \sum_{j=1}^n w_j x_j$, so keeping track of only the marginal probabilities $\boldsymbol{x}$ is sufficient for exact optimization. However, submodular functions do not in general satisfy this property: the utilities will depend on the full distribution p, not just the marginals $\boldsymbol{x}$. We will treat a given marginal vector $\boldsymbol{x}$ as representing an independent distribution where each j is present with probability x_j (i.e., $\boldsymbol{x}$ compactly represents the full distribution $p_{\boldsymbol{x}}^I$). The expected value of $\boldsymbol{x}$ under any submodular function is exactly given by its multilinear extension, which is a continuous function.

Continuous extension: Let $G = \min_i F_i$ be the pointwise minimum of the multilinear extensions of the functions in $\mathcal{F}$. Note that for any marginal $\boldsymbol{x}$, $G(\boldsymbol{x})$ is exactly the objective value of $p_{\boldsymbol{x}}^I$ for Problem 2.1. Hence, optimizing G over all $\boldsymbol{x} \in \mathcal{P}$ is equivalent to solving Problem 2.1 restricted to independent distributions. Via the correlation gap, this restriction only loses a factor $(1 - 1/e)$: if the optimal full distribution is p_{OPT}, then the independent distribution with the same marginals as p_{OPT} has at least $(1 - 1/e)$ of of p_{OPT}'s value under any submodular function. Previous algorithms [CCPV11, BMBK17] for optimizing up-concave functions like G do not apply because G is nonsmooth (see below). We introduce a novel Stochastic Frank-Wolfe algorithm which smooths the objective with random noise. Its runtime does not depend directly on $|\mathcal{F}|$ at all; it only uses BRI calls.

Rounding: Once we have solved the continuous problem, we need a way of mapping the resulting marginal vector $\boldsymbol{x}$ to a distribution over the pure strategies $\mathcal{I}$. Notice that if we simply sample items independently according to $\boldsymbol{x}$, we might end up with an invalid set. For instance, in the uniform matroid which requires $|S| \leq k$, an independent draw could result in more than k items even if $\sum_i x_i \leq k$. Hence, we sample pure strategies by running the swap rounding algorithm on $\boldsymbol{x}$. In order to implement the maximizing player's equilibrium strategy, it suffices to simply draw a sample whenever a decision is required. If a full description of the mixed strategy is desired, we show that it is sufficient to draw $\Theta\left(\frac{1}{\epsilon^2}(\log|\mathcal{F}| + \log\frac{1}{\delta})\right)$ independent samples via swap rounding and return the uniform distribution over the sampled pure strategies.

To sum up, our strategy is as follows. First, solve the continuous optimization problem to obtain marginal vector $\boldsymbol{x}$. Second, draw sampled pure strategies by running randomized swap rounding on $\boldsymbol{x}$.

2.4.1 Solving the continuous problem

The linchpin of our algorithmic strategy is solving the optimization problem $\max_{\boldsymbol{x}\in\mathcal{P}(\mathcal{M})} G(\boldsymbol{x})$. In this section, we provide the ingredients to do so.

Properties of G: We set the stage with four important properties of G (proofs are given in the supplement). First, while G is not in general concave, it is *up-concave*:

Lemma 2. *If $F_1...F_m$ are up-concave functions, then $G = \min_i F_i$ is up-concave as well.*

The proof is similar to the proof that the minimum of concave functions is concave. Up-concavity of G is the crucial property that enables efficient optimization.

Second, G is Lipschitz. Specifically, let $M = \max_{i,j} f_i(\{j\})$ be the maximum value of any single item. It can be shown that $||\nabla F_i||_\infty \leq M \ \forall i$ since (intuitively), the gradient of F_i is related to the marginal gain of items under f_i. From this we derive

Lemma 3. *G is M-Lipschitz in the ℓ_1 norm.*

Third, G is not smooth. For instance, it is not even differentiable at points where the minimizing function is not unique. This complicates the problem of optimizing G and renders earlier algorithms inapplicable.

Algorithm 3 EQUATOR(BRI, FO, LO, u, c, K, r)

```
$\boldsymbol{x}^0 \leftarrow u\mathbf{1}$
//Stochastic Frank-Wolfe algorithm
for $\ell = 1...K$ do
    for $t = 1...c$ do
        Draw $\boldsymbol{z} \sim \mu(u)$
        $F_t \leftarrow$ BRI($\boldsymbol{x}^{\ell-1} + \boldsymbol{z}$)
        $\tilde{\nabla}_t^\ell \leftarrow FO(F_t, \boldsymbol{x}^{\ell-1} + \boldsymbol{z})$
    end for
    $\tilde{\nabla}^\ell \leftarrow \frac{1}{c}\sum_{t=1}^{c} \tilde{\nabla}_t^\ell$
    $\boldsymbol{v}^\ell \leftarrow LO(\tilde{\nabla}^\ell)$
    $\boldsymbol{x}^\ell \leftarrow \boldsymbol{x}^{\ell-1} + \frac{1}{K}\boldsymbol{v}^\ell$
end for
$\boldsymbol{x}_{final} \leftarrow \boldsymbol{x}^K - u\mathbf{1}$
//Sample from equilibrium mixed strategy
Return $r$ samples of SwapRound($\boldsymbol{x}_{final}$)
```

Fourth, at any point $\boldsymbol{x}$ where the minimizing function F_i is unique, $\nabla G(\boldsymbol{x}) = \nabla F_i(\boldsymbol{x})$. Hence, we can compute $\nabla G(\boldsymbol{x})$ by calling the BRI to find F_i, and then computing $\nabla F_i(\boldsymbol{x})$. In general, $\nabla F_i(\boldsymbol{x})$ can be computed by random sampling [CCPV11], and closed forms are known for particular cases [IJB14].

Randomized smoothing: We will solve the continuous problem $\max_{\boldsymbol{x} \in \mathcal{P}(\mathcal{M})} G(\boldsymbol{x})$. Known strategies for optimizing up-concave functions [BMBK17] rely crucially on G being smooth. Specifically, ∇G must be Lipschitz continuous. Unfortunately, G is not even differentiable everywhere. Even between two points $\boldsymbol{x}$ and $\boldsymbol{y}$ where G is differentiable, $\nabla G(\boldsymbol{x})$ and $\nabla G(\boldsymbol{y})$ can be arbitrarily far apart if $\arg\min_i F_i(\boldsymbol{x}) \neq \arg\min_i F_i(\boldsymbol{y})$. No previous work addresses nonsmooth optimization of an up-concave function.

To resolve this issue, we use a carefully calibrated amount of random noise to smooth the objective. Let $\mu(u)$ be the uniform distribution over the ℓ_∞ ball of radius u. We define the smoothed objective $G_\mu(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{z} \sim \mu(u)}\left[G(\boldsymbol{x} + \boldsymbol{z})\right]$ which averages over the region around $\boldsymbol{x}$. This (and similar) techniques have been studied in the context of convex optimization [DBW12]. We show that G_μ is a good smooth approximator of G.

Lemma 4. *G_μ has the following properties:*

- *G_μ is up-concave.*

- $|G_\mu(\boldsymbol{x}) - G(\boldsymbol{x})| \leq \frac{Mnu}{2} \quad \forall \boldsymbol{x}$.

- *G_μ is differentiable, with $\nabla G_\mu(x) = \mathbb{E}[\nabla G(\boldsymbol{x} + \boldsymbol{z})]$.*

- *∇G_μ is $\frac{M}{\mu}$−Lipschitz continuous in the ℓ_1 norm.*

Hence, we can use G_μ as a better-behaved proxy for G since it is both smooth and close to G everywhere in the domain. The main challenge is that G_μ and its gradients are not available in closed form. Accordingly, we randomly sample values of the perturbation $\boldsymbol{z}$ and average over the value of G (or its gradient) at these sampled points.

2.4.2 Stochastic Frank-Wolfe algorithm (SFW)

We propose the SFW algorithm (Algorithm 3) to optimize G_μ. SFW generates a series of feasible points $\boldsymbol{x}^0 ... \boldsymbol{x}^K$, where K is the number of iterations. Each point is generated from the last via two steps. First, SFW estimates the gradient of G_μ. Second, it takes a step towards the point in $\mathcal{P}$ which is furthest in the direction of the gradient. To carry out these steps, SFW requires three oracles. First, a linear optimization oracle LO which, given an objective $\boldsymbol{w}$, returns $\arg\max_{v \in \mathcal{P}(\mathcal{M})} \boldsymbol{w}^\top \boldsymbol{v}$. In the context of our problem, LO outputs the indicator vector of the set $S \in \mathcal{I}$ which maximizes the linear objective $\boldsymbol{w}$. S can be efficiently found via the greedy algorithm. The other two oracles concern gradient evaluation. One is the BRI oracle discussed earlier. The other is a stochastic first-order oracle FO which, for any function F_i and point $\boldsymbol{x}$, returns an unbiased estimate of $\nabla F_i(\boldsymbol{x})$.

The algorithm starts at $\boldsymbol{x}^0 = \boldsymbol{0}$. At each iteration ℓ, it averages over c calls to FO to compute a stochastic approximation $\tilde{\nabla}^\ell$ to $\nabla G_\mu(\boldsymbol{x}^{\ell-1})$ (Lines 4-9). For each call, it draws a random perturbation $\boldsymbol{z} \sim \mu(u)$ and uses the BRI to find the minimizing F at $\boldsymbol{x}^{\ell-1} + \boldsymbol{z}$. It then queries FO for an estimate of $\nabla F(\boldsymbol{x}^{\ell-1} + \boldsymbol{z})$. Lastly, it takes a step in the direction of $\boldsymbol{v}^\ell = LO(\tilde{\nabla}^\ell)$ by setting $\boldsymbol{x}^\ell = \boldsymbol{x}^{\ell-1} + \frac{1}{K}\boldsymbol{v}^\ell$ (Lines 10-11). Since $\boldsymbol{x}^\ell$ at each iteration is a combination of vertices of $\mathcal{P}(\mathcal{M})$, the output is guaranteed to be feasible. The intuition for why the algorithm succeeds is that it only moves along nonnegative directions (since

$\boldsymbol{v}^\ell$ is always nonnegative). This is in contrast to gradient-based algorithms for *concave* optimization, which move in the (possibly negative) direction $\boldsymbol{v}^\ell - \boldsymbol{x}^\ell$. As an up-concave function, G_μ is concave along all nonegative directions. By moving only in such directions we inherit enough of the nice properties of concave optimization to obtain a $(1 - 1/e)-$ approximation.

A small technical detail is that adding random noise $\boldsymbol{z}$ could result in negative values, for which the multilinear extension is not defined. To circumvent this, we start the algorithm at $\boldsymbol{x}^0 = u\mathbf{1}$ (i.e., with small positive values in every entry) and then return $\boldsymbol{x}_{final} = \boldsymbol{x}^K - u\mathbf{1}$ (Line 13).

2.4.3 Theoretical bounds

Let T_1 be the runtime of the linear optimization oracle and T_2 be the runtime of the first-order oracle. We prove the following guarantee for SFW:

Theorem 4. *For any $\epsilon, \delta > 0$, there are parameter settings such that SFW finds a solution $\boldsymbol{x}^K$ satisfying $G(\boldsymbol{x}^K) \geq (1 - \frac{1}{e})OPT - \epsilon$ with probability at least $1 - \delta$. The runtime of the algorithm is $\tilde{\mathcal{O}}\left(T_1 \frac{M^2k^2n}{\epsilon^2} + T_2 \frac{k^4M^4n}{\epsilon^4} \log \frac{1}{\delta}\right)$*[1]*.*

We remark that T_1 is small since linear optimization over $\mathcal{P}(\mathcal{M})$ can be carried out by a greedy algorithm. For instance, the runtime is $T_1 = \mathcal{O}(n \log n)$ for the uniform matroid, which covers many applications. T_2 is typically dominated by the runtime of the BRI since it is known how to efficiently compute the gradient of a submodular function [CCPV11, IJB14].

Based on this result, we show the following guarantee on a single randomly sampled set that EQUATOR returns after applying swap rounding to the marginal vector $\boldsymbol{x}_{final}$.

Theorem 5. *With $r = 1$, EQUATOR outputs a set $S \in \mathcal{I}$ such that $\min_i \mathbb{E}[f_i(S)] \geq (1 - \frac{1}{e})^2 OPT - \epsilon$ with probability at least $1 - \delta$. Its time complexity is the same as SFW.*

Proof. Suppose that p_{OPT} is the distribution achieving the optimal value for Problem 2.1. Let $\boldsymbol{x}^*$ be the optimizer for the problem $\max_{\boldsymbol{x} \in \mathcal{P}(\mathcal{M})} G(\boldsymbol{x})$. That is, $\boldsymbol{x}^*$ can be interpreted as

[1] The $\tilde{\mathcal{O}}$ notation hides logarithmic terms

the marginals of the independent distribution which maximizes $\min_i \mathbb{E}_{S\sim p^I_{x^*}}[f_i(S)]$. With slight abuse of notation, let p^I_{OPT} be the independent distribution with the same marginals as p_{OPT}. By applying the correlation gap to each $f_i \in \mathcal{F}$ and taking the min, we have

$$\min_{f_i\in\mathcal{F}} \mathop{\mathbb{E}}_{S\sim p_{OPT}}[f_i(S)] \leq \frac{e}{e-1} \min_{f_i\in\mathcal{F}} \mathop{\mathbb{E}}_{S\sim p^I_{OPT}}[f_i(S)].$$

By definition of $\boldsymbol{x}^*$, $G(\boldsymbol{x}^*) \geq \min_{f_i\in\mathcal{F}} \mathbb{E}_{S\sim p^I_{OPT}}[f_i(S)]$. Hence, we have that $G(\boldsymbol{x}^*) \geq (1-1/e)\min_i \mathbb{E}_{S\sim p^I_{x^*}}[f_i(S)] = (1-1/e)OPT$. Via Theorem 4, the marginal vector $\boldsymbol{x}$ that our algorithm finds satisfies $G(\boldsymbol{x}) \geq (1-\frac{1}{e})G(\boldsymbol{x}^*) - \epsilon \geq (1-\frac{1}{e})^2 OPT - \epsilon$. Lastly, Chekuri et al. [CVZ10] show that swap rounding outputs an independent set S satisfying $\mathbb{E}[f_i(S)] \geq F_i(S)$ for any $f_i \in \mathcal{F}$, which completes the proof. □

This guarantee is sufficient if we just want to implement the maximizing player's strategy by sampling an action. We also prove that if a full description of the maximizing player's mixed strategy is desired, drawing a small number of independent samples via swap rounding suffices:

Algorithm 4 Efficient bicriteria approximation

```
Run EQUATOR to obtain $\boldsymbol{x}_{final}$.
for $j = 1...e\log\frac{1}{\delta}$ do
    run SwapRound($\boldsymbol{x}_{final}$) $\frac{8\log|\mathcal{F}|}{\epsilon^3} + 1$ times, yielding $S^j_1...S^j_r$.
    $S_j \leftarrow S^j_1 \cup S^j_2 \cup .... \cup S^j_r$
end for
return $\arg\max_{S_j} \min_{f_i\in\mathcal{F}} f_i(S_j)$
```

Theorem 6. *Draw $r = \mathcal{O}\left(\frac{1}{\epsilon^3}\left(\log|\mathcal{F}| + \log\frac{1}{\delta}\right)\right)$ samples using independent runs of randomized swap rounding. The uniform distribution on these samples is a $(1-\frac{1}{e})^2 - \epsilon$ approximate equilibrium strategy for the maximizing player with probability at least $1-\delta$. The runtime is $\mathcal{O}\left(\frac{rk^2M^2n}{\epsilon}\right)$.*

This also gives a simple way of obtaining a single feasible set (pure strategy) which has a bicriteria guarantee for the robust optimization problem. As pointed out by Chen et al. [CLSS17], since the f_i are all monotone, taking the union of the sets output by swap rounding gives a single set with at least as much value. Algorithm 4 implements this procedure. It first solves the fractional problem by running EQUATOR. Then, it carries out a series of independent iterations. Each iteration j draws $\frac{8\log|\mathcal{F}|}{\epsilon^3}$ sets via swap rounding and stores their union S_j. It then returns the best of the S_j. Via our concentration bound for the distribution produced in each iteration (Theorem 6), each iteration succeeds in producing a "good" set with probability at least $\frac{1}{e}$. Algorithm 4 runs $e \log \frac{1}{\delta}$ iterations so that at least one succeeds with probability at least $1 - \delta$.

Theorem 7. *Algorithm 4 returns a single set S which is the union of at most $\frac{8\log|\mathcal{F}|}{\epsilon^3} + 1$ elements of $\mathcal{I}$ and satisfies $\min_{f_i \in \mathcal{F}} f_i(S) \geq \left(1 - \frac{1}{e}\right)^2 \max_{S^* \in \mathcal{I}} \min_{f_i \in \mathcal{F}} f_i(S^*) - \epsilon$ with probability at least $1 - \delta$.*

The strongest existing bicriteria guarantee is for the SATURATE algorithm of Krause et al. [KMGG08], which outputs a set which matches the optimal value attainable using a set of size k using $\left(\log\left(\max_{v \in X} \sum_{f_i \in \mathcal{F}} f_i(\{v\})\right) + 1\right) k$ items. Our S maintains logarithmic dependence on $|\mathcal{F}|$, but also contains dependence on ϵ. Moreoever, it is only a $(1 - \frac{1}{e})^2$-approximation to the optimal solution quality. However, our result is much more general than that of Krause et al. and handles situations that SATURATE cannot. First, our result applies when $\mathcal{F}$ is accessible only through an oracle, where SATURATE relies on explicitly enumerating the functions. Second, our result applies when $\mathcal{I}$ is *any* matroid, where SATURATE applies only to cardinality-constrained problems. To our knowledge, this is the first computationally efficient bicriteria algorithm under either condition.

2.5 Improving the approximation ratio

In this section, we examine the conditions under which it is possible to improve EQUATOR's $\left(1 - \frac{1}{e}\right)^2$-approximation to $\left(1 - \frac{1}{e}\right)$. The earlier analysis lost a factor $\left(1 - \frac{1}{e}\right)$ in two places: the use of the correlation gap to bound the loss introduced by only tracking marginals, and the use of SFW to solve the continuous relaxation. While the second factor is difficult to improve, we can eliminate the loss from the correlation gap when a stronger best response oracle for the adversary is available. Specifically, we define a *best response to mixture of independent distributions* (BRMI) oracle to be an algorithm which, given a list of marginal vectors $\boldsymbol{x}^1...\boldsymbol{x}^\rho$, outputs

$$\arg\min_{f_i\in\mathcal{F}} \frac{1}{\rho}\sum_{j=1}^{\rho} F_i(\boldsymbol{x}^j).$$

We will be interested in BRMI oracles which take time polynomial in ρ. As the name implies, a BRMI oracle can compute adversary best responses to any distribution which is explicitly represented as a mixture of independent distributions with given marginals. By contrast, a BRI is restricted to a single independent distribution. A BRMI is a considerably more powerful oracle because, with sufficiently large ρ, any distribution can be arbitrarily well-approximated by a mixture of independent distributions (a statement which is formalized below). Hence, the algorithm we propose maintains ρ copies of the decision variables $\boldsymbol{x}^1...\boldsymbol{x}^\rho$ for a value of ρ which will be set later. We aim to maximize

Algorithm 5 EQUATOR with improved approximation guarantee

```
Set $\rho = O\left(\frac{W^2 \log|\mathcal{F}|}{\epsilon^2}\right)$
Use SFW to solve the problem $\max_{\boldsymbol{x}^1..\boldsymbol{x}^\rho \in \times_{j=1}^{\rho}\mathcal{P}} \min_{f_i\in\mathcal{F}} \frac{1}{\rho}\sum_{j=1}^{\rho} F_i(\boldsymbol{x}^j)$, obtaining $\boldsymbol{x}^1..\boldsymbol{x}^\rho$
Set $r = \mathcal{O}\left(\frac{1}{\epsilon^3}\log\left(\frac{|\mathcal{F}|\rho}{\delta}\right)\right)$
for $i = 1...\rho$ do
    Draw sets $S^i_1...S^i_r$ independently as SwapRound($\boldsymbol{x}^i$).
end for
Return the uniform distribution on $\{S^i_j : i = 1...\rho, j = 1...r\}$.
```

$$\max_{\boldsymbol{x}^1..\boldsymbol{x}^\rho \in \times_{j=1}^{\rho}\mathcal{P}} \min_{f_i\in\mathcal{F}} \frac{1}{\rho}\sum_{j=1}^{\rho} \mathop{\mathbb{E}}_{S\sim\boldsymbol{x}^j}[f_i(S)]$$

which we recognize as being equivalent to the problem

$$\max_{\boldsymbol{x}^1..\boldsymbol{x}^\rho \in \times_{j=1}^{\rho}\mathcal{P}} \min_{f_i\in\mathcal{F}} \frac{1}{\rho}\sum_{j=1}^{\rho} F_i(\boldsymbol{x}^j) \tag{2.2}$$

It is easy to check that $\frac{1}{\rho}\sum_{j=1}^{\rho} F_i(\boldsymbol{x}^j)$ is an up-concave function which inherits all of the smoothness properties of the F_i. Hence, we can use SFW to obtain a $\left(1-\frac{1}{e}\right)$-approximate solution to Problem 2.2 provided that we have a BRMI oracle with which to compute gradients. After solving Problem 2.2, we can use swap rounding to produce feasible sets with guaranteed approximation ratio. For a single set, we first select a $j \in \{1...\rho\}$ uniformly at random and then run swap rounding on $\boldsymbol{x}^j$. To output a full distribution, as in Theorem 6, we draw $r = \mathcal{O}\left(\frac{1}{\epsilon^3}\log\left(\frac{|\mathcal{F}|\rho}{\delta}\right)\right)$ samples from each of the $\boldsymbol{x}^j$ and then output the uniform distribution over the combined set of samples. The extra logarithmic dependence on ρ ensures that we can take a final union bound over the ρ batches of swap rounding. The entire procedure is summarized in Algorithm 5. We let W be an upper bound on the value of f for any feasible set: $W \geq \max_{f_i\in\mathcal{F}, S\in\mathcal{I}} f_i(S)$. Note that $W \leq nM$ always holds via submodularity, but tighter bounds might apply for particular functions.

We have the following approximation guarantee for Algorithm 5. We note that the idea of optimizing over a mixture of independent distributions has been used in [DX17], but we prove Lemma 5 (establishing that a good mixture exists) for completeness.

Theorem 8. *Given access to a BRMI oracle for any SBR game instance, Algorithm 5 returns a distribution p which satisfies $\min_{f_i\in\mathcal{F}} \mathbb{E}_{S\sim p}\left[f_i(S)\right] \geq \left(1-\frac{1}{\epsilon}\right) OPT - \epsilon$ with probability at least $1-\delta$.*

Proof. We first establish that there exists a near-optimal distribution over elements of $\mathcal{I}$ with support size at most $O\left(\frac{W^2\log|\mathcal{F}|}{\epsilon^2}\right)$:

Lemma 5. *Take any collection of functions $\mathcal{F}$ with $\max_{f_i\in\mathcal{F}, S\in\mathcal{I}} f_i(S) \leq W$ and a distribution $p \in \Delta(\mathcal{I})$. There exists a distribution q supported on at most $\rho = O\left(\frac{W^2\log|\mathcal{F}|}{\epsilon^2}\right)$ elements of $\mathcal{I}$ which satisfies $\mathbb{E}_{S\sim q}[f_i(S)] \geq \mathbb{E}_{S\sim p}[f_i(S)] - \epsilon$ for all $f_i \in \mathcal{F}$.*

Proof. We will use the probabilistic method. Suppose that we draw $\rho = \frac{W^2\log|\mathcal{F}|}{\epsilon^2}$ samples $S_1...S_\rho$ independently from p and let q be the uniform distribution on the samples. Fix an arbitrary function f_i. Via Hoeffding's inequality, we have that

$$\Pr\left[\mathop{\mathbb{E}}_{S\sim p}[f_i(S)] - \frac{1}{r}\sum_{i=1}^{r} f_i(S) \geq \epsilon\right] \leq e^{-\frac{2r\epsilon^2}{W^2}} \leq \frac{1}{|\mathcal{F}|^2}$$

and this holds simultaneously for all scenarios y with probability at least $1 - \frac{1}{|\mathcal{F}|} > 0$ via union bound. That is, we have a random sampling procedure which outputs a distribution q satisfying $\mathbb{E}_{S\sim q}[f_i(S)] \geq \mathbb{E}_{S\sim p}[f_i(S)] - \epsilon$ for all $f_i \in \mathcal{F}$ with positive probability. Via the probabilistic method we are guaranteed that such a distribution (i.e., one which is a uniform distribution on at most ρ elements of $\mathcal{I}$) exists. □

Now, note that Algorithm 5 maximizes over the set $\times_{j=1}^{\rho}\mathcal{P}$, which includes the distribution q. Via the guarantee for SFW (Theorem 4), SFW returns $\boldsymbol{x}^1 ... \boldsymbol{x}^\rho$ satisfying

$$\min_{f_i\in\mathcal{F}} \frac{1}{\rho}\sum_{j=1}^{\rho} F_i(\boldsymbol{x}^j) \geq \left(1 - \frac{1}{e}\right) OPT - \epsilon.$$

We ignore for convenience the issue of adjusting all of the ϵ values by a constant factor. Now we just need to establish that the rounding procedure succeeds. A simple variation on the proof of Theorem 6 suffices: we claim that $\frac{1}{r}\sum_{a=1}^{r} f_i(S_j^a) \geq \mathbb{E}_{S\sim\boldsymbol{x}^j}[f_i(S)] - \epsilon$ holds for each i, j with probability at least $1 - \frac{\delta}{\rho|\mathcal{F}|}$ via our choice of r. Taking union bound over all $i = 1...|\mathcal{F}|$ and $j = 1...r$ completes the proof. □

2.6 Applications

We now give several examples of domains that our algorithm can be applied to. In each of these cases, we obtain the first guaranteed polynomial time constant-factor approximation algorithm for the problem. The key part of both applications is developing a BRI (the first order oracle is easily obtained in closed form via straightforward calculus).

Network security games: Earlier, we formulated network security games in the SBR framework. All we need to solve it using EQUATOR is a BRI oracle. The full attacker best response problem is known to be NP-hard [JKV$^+$11]. However, it turns out the best response to an *independent* distribution is easily computed. Index the set of paths and let P_i

be the ith path, ending at a target with value τ_i. Let $P(t_j)$ be the set of all paths from the (super)source s to t_j. Let f_i be the corresponding submodular objective. Given a defender mixed strategy $\boldsymbol{x}$, the attacker best response problem is to find $\min_i \mathbb{E}_{S\sim\boldsymbol{x}}[f_i(S)]$. We can rewrite this as

$$\begin{aligned}\min_i \mathop{\mathbb{E}}_{S\sim\boldsymbol{x}}[f_i(S)] &= \min_i \mathop{\mathbb{E}}_{S\sim\boldsymbol{x}}[\tau_i \mathbf{1}[S\cap P_i \neq \emptyset]]\\ &= \min_{t_j\in T} \tau_j \min_{P\in P(t_j)} \mathop{\mathbb{E}}_{S\sim\boldsymbol{x}}[\mathbf{1}[S\cap P\neq\emptyset]]\\ &= \min_{t_j\in T} \tau_j \min_{P\in P(t_j)} 1-\prod_{e\in P}[1-x_e]\end{aligned}$$

We can now solve a separate problem for each target t_j and then take the one with lowest value. For each t_j, we solve a shortest path problem. We aim to find a $s-t_j$ path which maximizes the product of the the weights $1-x_e$ on each edge. Taking logarithms, this is equivalent to finding the path which minimizes $-\sum_{e\in P}\log(1-x_e) = \sum_{e\in P}\log\frac{1}{1-x_e}$. This is a shortest path problem in which each edge has nonnegative weight $\log\frac{1}{1-x_e}$, and so can be solved via Dijkstra's algorithm. With the attacker BRI in hand, applying EQUATOR yields the first subexponential-time algorithm for network security games.

Robust coverage and budget allocation: Many widespread applications of submodular functions concern coverage functions. A coverage function takes the following form. There a set of items U, and each $j\in U$ has a weight w_j. The algorithm can choose from a ground set $X=\{a_1...a_n\}$ of actions. Each action a_i covers a set $A_i\subseteq U$. The value of any set of actions is the total value of the items that those actions cover: $f(S)=\sum_{j\in\bigcup_{i\in S}A_i} w_j$. We can also consider probabilistic extensions where action a_i covers each $j\in A_i$ independently with probability p_{ij}. This framework includes budget allocation, sensor placement, facility location, and many other common submodular optimization problems. Here we consider a *robust coverage* problem where the weights $\boldsymbol{w}$ are unknown. For concreteness, we focus on the budget allocation problem, but all of our logic applies to general coverage functions.

Budget allocation models an advertiser's choice of how to divide a finite budget B between a set of advertising channels. Each channel is a vertex on the left hand side L of a bipartite graph. The right hand R consists of customers. Each customer $v \in R$ has a value w_v which is the advertiser's expected profit from reaching v. The advertiser allocates their budget in integer amounts among L. Let $y(s)$ denote the amount of budget allocated to channel $s \in L$. The advertiser solves the problem

$$\max_{\boldsymbol{y}:||\boldsymbol{y}||_1 \leq B} f_{\boldsymbol{w}}(\boldsymbol{y}) = \sum_{v \in R} w_v \left[1 - \prod_{s \in L} (1 - p_{sv})^{y(s)} \right]$$

where p_{sv} is the probability that one unit of advertising on channel s will reach customer v. This a probabilistic coverage problem where the action set X contains B copies[2] of each $s \in L$ and the feasible decisions $\mathcal{I}$ are all size B subsets of X. Choosing b copies of node s corresponds to setting $y(s) = b$. Budget allocation has been the subject of a great deal of recent research [AGT12, SKIK14, MIFK15].

In the robust optimization problem, the profits $\boldsymbol{w}$ are not exactly known. Instead, they belong to a polyhedral uncertainty set $\mathcal{U}$. This is very realistic: while an advertiser may be able to estimate the profit for each customer from past data, they are unlikely to know the true value for any particular campaign. We remark that Staib and Jegelka [SJ17] also considered a robust budget allocation problem, but their problem has uncertainty on the probabilities p_{st}, not the profits $\boldsymbol{w}$. Further, they consider a continuous problem without the complication of rounding to discrete solutions.

As an example uncertainty set, consider the D-norm uncertain set, which is common in robust optimization [BPS04, SJ17]. The uncertainty set is defined around a point estimate $\hat{\boldsymbol{w}}$ as

$$\mathcal{U}_{\gamma}^{\hat{\boldsymbol{w}}} = \{\boldsymbol{w} : \exists \boldsymbol{c} \in [0,1]^{|R|}, w_i = (1 - c_i)\hat{w}_i,\ ||\boldsymbol{c}||_1 \leq \gamma\}.$$

This can be thought of as allowing an adversary to scale down each entry of $\hat{\boldsymbol{w}}$ with a total budget of γ. In our case, $\hat{\boldsymbol{w}}$ is the advertiser's best estimate from past data, and

[2] We use this formulation for simplicity, but it is possible to use only $\log B$ copies of each node [EN16].

they would like to perform well for all scenarios within $\mathcal{U}_\gamma^{\hat{w}}$. γ defines the advertiser's tolerance for risk. The problem we want to solve is $\max_{p\in\Delta(\mathcal{I})} \min_{\boldsymbol{w}\in\mathcal{U}_\gamma^{\hat{w}}} \mathbb{E}_{\boldsymbol{y}\sim p}[f_{\boldsymbol{w}}(\boldsymbol{y})]$, which we recognize as an instance of Problem 2.1. For any fixed distribution p, we have by linearity of expectation

$$\mathop{\mathbb{E}}_{\boldsymbol{y}\sim p}[f_{\boldsymbol{w}}(\boldsymbol{y})] = \sum_{v\in R} w_v \mathop{\mathbb{E}}_{\boldsymbol{y}\sim p}\left[1 - \prod_{s\in L}(1-p_{sv})^{y(s)}\right].$$

Note that the inner expectation (which is the total probability that each $v \in R$ is reached) is constant with respect to $\boldsymbol{w}$. Hence, the adversary's best response problem of computing $\min_{\boldsymbol{w}\in\mathcal{U}} \mathbb{E}_{\boldsymbol{y}\sim p}[f_{\boldsymbol{w}}(\boldsymbol{y})]$ is a linear program and can be easily solved. The coefficients of this LP (the inner expectation in the above sum) can easily be computed exactly for any independent distribution. Further, since any LP has an optimal solution among the vertices of $\mathcal{U}_\gamma^{\hat{w}}$, we can without loss of generality restrict the adversary's pure strategies to a finite (though exponentially large) number.

Lastly, we remark that it also possible to obtain a BRMI for this problem. For any distribution p, we can find a best response via linear programming provided that the coefficients $\mathbb{E}_{\boldsymbol{y}\sim p}\left[1 - \prod_{s\in L}(1-p_{sv})^{y(s)}\right]$ can be computed for each $v \in R$. This is easy when p is given explicitly as a mixture of independent distributions $\boldsymbol{x}^1 ... \boldsymbol{x}^\rho$ since we just average over the corresponding term for each individual $\boldsymbol{x}^i$. Hence, we can use Algorithm 5 to obtain a $\left(1-\frac{1}{e}\right)$-approximation. Nevertheless, we use the original EQUATOR algorithm in our experiments and find that it performs near-optimally despite its theoretically weaker approximation ratio.

2.7 Experiments

We now show experimental results from applying EQUATOR to these two domains.

Network security games: We first study the network security game defined above. We compare EQUATOR to the SNARES algorithm [JCT13] which is the current state of the art algorithm with guaranteed solution quality. SNARES uses a double oracle approach to find a *globally* optimal solution. However, it incorporates several domain-specific heuristics which

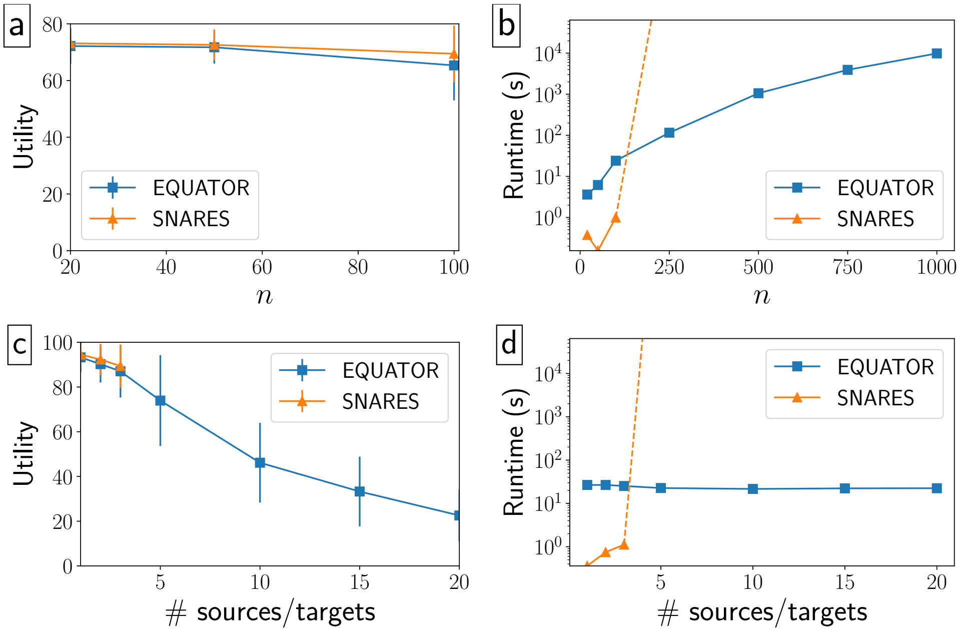


Figure 2.1: *Experimental results for network security games.*

substantially improve its runtime over a standard implementation of double oracle. We note that Iwashita et al. [IOAI16] proposed a newer double-oracle style algorithm which first preprocesses the graph to remove unnecessary edges. We do not compare to this approach because the preprocessing step can be applied equally well to either EQUATOR or double oracle. We use random geometric graphs, which are commonly used to assess algorithms for this domain due to their similarity to real world road networks [JCT13, IOAI16]. As in Jain et al. [JCT13], we use density $d = 0.1$ with the value of each target drawn uniformly at random in $[0, 100]$. We set k to be one percent of the number of edges. Each data point averages over 30 random instances. EQUATOR was run with $K = 100, c = 60, u = 0.1$.

Figure 2.1 shows the results. Figures 2.1(a) and 2.1(b) vary the network size n with three randomly chosen source and target nodes. Figure 2.1(a) plots utility (i.e., how much loss is averted by the defender's allocation) as a function of n. Error bars show one standard deviation. We see that EQUATOR obtains utility within 6% of SNARES, which computes a global optimum. Figure 2.1(b) shows runtime (on a logarithmic scale) as a function of n. SNARES was terminated after 10 hours for graphs with 250 nodes, while EQUATOR easily scales to 1000 nodes. Next, Figures 2.1(c) and 2.1(d) show results as the number of sources and targets grows. As expected, utility decreases with more sources/targets since the number of resources is constant and it becomes harder to defend the network. EQUATOR obtains utility within 4% of SNARES. However, SNARES was terminated after 10 hours for just 5 source/targets, while EQUATOR runs in under 25 seconds with 20 source/targets.

Robust budget allocation: We compare three algorithms for robust budget allocation. First, EQUATOR. Second, double oracle. We use the greedy algorithm for the defender's best response (which is a $(1 - 1/e)$-approximation) since the exact best response is intractable. For the adversary's best response, we use the linear program discussed in the section on robust coverage. Third, we compare to "greedy", which greedily optimizes the advertiser's return under the point estimate $\hat{w}$. Greedy was implemented with lazy evaluation [Min78] which greatly improves its runtime at no cost to solution value. We generated random bipartite graphs with $|L| = |R| = n$ where each potential edge is present with probability 0.2 and for each edge (u, v), $p_{u,v}$ is draw uniformly in $[0, 0.2]$. $\hat{w}$ was randomly generated with each coordinate uniform in $[0.5, 1.5]$. Our uncertainty set is the D-norm set around $\hat{w}$ with $\gamma = \frac{1}{2}n$, representing a substantial degree of uncertainty. The budget was $B = 5 + 0.01 \cdot n$ since the problem is hardest when B is small relative to n. EQUATOR was run with $K = 20, c = 10, u = 0.1$.

Figure 2.2 shows the results. Each point averages over 30 random problem instances (error bars would be hidden under the markers). Figure 2.2(a) plots the profit obtained by each algorithm when the true w is chosen as the worst case in $\mathcal{U}_{\gamma}^{\hat{w}}$, with n increasing on the x axis. Figure 2.2(b) plots the average runtime for each n. We see that double oracle produces highly robust solutions. However, for even $n = 500$, its execution was halted after 10 hours. Greedy is highly scalable, but produces solutions that are approximately 40% less robust than double oracle. EQUATOR produces solution quality within 7% of double oracle and runs in less than 30 seconds with $n = 1000$.

Next, we show results on a real world dataset from Yahoo webscope [Yah07]. The dataset

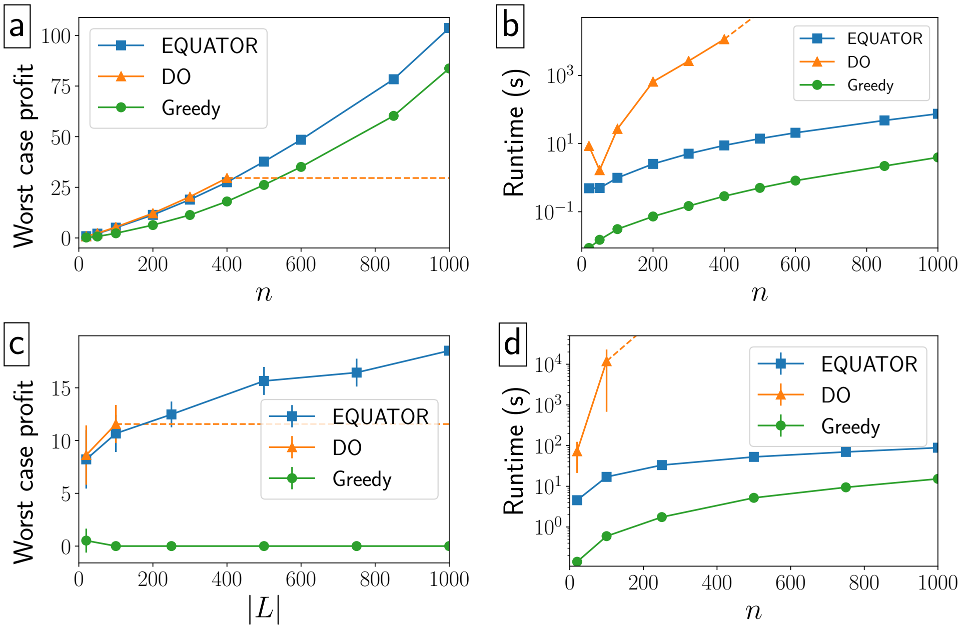


Figure 2.2: *Experimental results for budget allocation.*

logs bids placed by advertisers on a set of phrases. We create a budget allocation problem where the phrases are advertising channels and the accounts are targets; the resulting problem has $|L| = 1000$ and $|R| = 10,394$. Other parameters are the same as before. We obtain instances of varying size by randomly sampling a subset of L. Figures 2.2(c-d) show results (averaging over 30 random instances). In Figure 2.2(c), we see that both double oracle and EQUATOR find highly robust solutions, with EQUATOR's solution value within 8% of that of double oracle. By contrast, greedy obtains *no* profit in the worst case for $|L| > 20$, validating the importance of robust solutions on real problems. In Figure 2.2(d), we observe that double oracle was terminated after 10 hours for $n = 500$ while EQUATOR scales to $n = 1000$ in under 40 seconds. Hence, EQUATOR is empirically successful at finding highly robust solutions in an efficient manner, complementing its theoretical guarantees.

Discussion and conclusion

This chapter introduces the class of submodular best response games, capturing the zero sum interaction between two players when one has a submodular best response problem. Examples include network security games and robust submodular optimization problems. We study the case where the set of possible objective functions is very large (exponential in the problem size), arising from an underlying combinatorial structure. Our main result is a pseudopolynomial time algorithm to compute an approximate minimax equilibrium strategy for the maximizing player when the set of submodular objectives admits a certain form of best response oracle. We instantiate this framework for two example domains, and show experimentally that our algorithm scales to much larger instances than previous approaches.

One interesting direction for future work is to extend this framework to new application domains. Submodular structure is present in many problems, e.g., sensor placement in water networks [KMGG08] or cyber-security monitoring [HLP$^+$15]. Both seem natural domains for future work, but designing appropriate best response oracles may be algorithmically challenging. Another open direction is to extend our framework to cases where only *approximate* best responses are available for the adversary. This would enable applications even in settings where an exact BRI is computationally intractable.

Chapter 3

CHANGE: piloting a field-ready approach to influence maximization

This chapter presents and field-tests a novel, practical agent for influence maximization, the challenge of selecting a small set of seed nodes in a social network who will diffuse information to many others. Such techniques have important applications ranging from preventative health [VP07, APMV01] to international development [BCDJ13]. It is inherently a multiagent problem because nodes (agents) make decisions in response to those around them [ZPV15, MS12].

We are particularly motivated by the challenge of preventing HIV spread among homeless youth [RTC$^+$12, YR11, RMRB07] (although our contributions would also assist other public health interventions). Here, influence maximization is used to select homeless youth who will serve as *peer leaders* and spread messages about HIV prevention through their social network. Pilot studies in this domain have shown that algorithmic approaches have great promise, substantially outperforming status-quo heuristics [YWR$^+$17]. However, current algorithms have a high barrier to entry: they require a great deal of time to gather the complete social network, expertise to select appropriate parameters, and computational power to run the algorithms. None of these are likely available to the resource-strained service providers ultimately responsible for deployment.

Gathering network data is particularly onerous because it requires individually surveying over a hundred youth. Network collection is more time intensive than simple survey methods, requiring days of time for a dedicated team of social work researchers. It is infeasible for service providers with many other responsibilities.

The other barriers are also serious impediments to wide-scale adoption of influence maximization. Service providers will not have access to the high-performance computing resources required by previous algorithms, where high computational cost is often incurred to find solutions robust to unknown parameters. For instance, DOSIM, a state of the art algorithm for robust influence maximization [WYI$^+$17b], requires hours on a high-performance computing system. A deployed system would need to run in minutes on a laptop.

This chapter presents CHANGE (CompreHensive Adaptive Network samplinG for social influencE), a novel, end-to-end agent for influence maximization which addresses the above barriers via a set of algorithmic contributions. CHANGE is easy to deploy, but this simplicity is crucially enabled by a series of insights into the social structure of homeless youth (which may be useful for other vulnerable populations). We conducted a pilot test of CHANGE's performance in a real deployment by a drop-in center serving homeless youth in a major U.S. city. CHANGE was used to plan a series of interventions designed to spread HIV awareness among the youth. *CHANGE obtained comparable influence spread to state of the art algorithms while surveying only 18% of nodes for network data*, a finding which is backed by additional simulation results.

Overall, CHANGE offers a practical, field-tested vehicle for deployed influence maximization which drastically lowers the barrier to entry. *To our knowledge, this is the first real-world pilot study of a network sampling algorithm for influence maximization and only the second ever field test of* any *influence maximization algorithm.*

Overview of algorithmic contributions: We now summarize how CHANGE handles the challenges above. We discuss related work in Section 3.1; however, none addresses these challenges.

First, to address the *data gathering* challenge, we present an easily deployable sampling protocol which randomly selects a small set of youth to interview. For each of these youth, a randomly chosen neighbor is also interviewed. We show that this procedure gathers enough of the network to enable influence maximization even though it surveys only a small number of nodes directly.

Second, to address *computational power* challenge (which in turn stems from unknown parameters), we present a heuristic for selecting influence maximization solutions which are robust to uncertainty in the probability p that influence will spread. We show that this heuristic finds solutions which obtain approximately 90% of the maximum possible influence spread under *any* value for p. Importantly, this heuristic runs in minutes on a laptop, while DOSIM (the previously proposed algorithm for this problem) requires hours to days of time on a high performance cluster.

Third, we integrate these components with an *adaptive greedy* algorithm for planning interventions and prove the first theoretical guarantee for influence maximization under execution errors. The challenge is that some youth selected as peer leaders may not attend the intervention [WYI^{+}17b, YWR^{+}17]. Our algorithm selects its action with such uncertainties in mind, observes which youth do attend, and then plans the next round using this observation. We prove that it obtains a constant-factor approximation to the *optimal* adaptive policy.

Overview of field deployment contributions: We conducted two pilot studies of CHANGE, each addressing distinct questions.

First, we conducted a *feasibility study* with two objectives. (i) We confirm that CHANGE's mechanism for sampling the network to gather edge data is implementable with a homeless youth population. This is nontrivial because homeless youth are often difficult to locate, making finding particular youth to query for network ties difficult. (ii) We validate that the data gathered is sufficiently accurate to enable influence maximization. Self-reported ties are subject to bias and forgetting [Bre00], making it important to investigate whether they are accurate enough to find influential nodes. This point is of broader interest, since previous influence maximization work has largely used self-reported ties [YWR$^+$17, WYI$^+$17b], but *no previous field study has validated their accuracy for influence maximization*. To address these questions we collected network data from 72 youth at a drop-in center via a range of methods: CHANGE's sampling mechanism, self-reports from the entire network, field observations by research staff, and interviews with staff members. Our results show that CHANGE's sampling mechanism is feasible, and that self-reported data is sufficient for high-quality influence maximization.

Second, we conduct an *intervention study* of the entire CHANGE agent with an additional set of 64 homeless youth. This includes network data collection, peer leader selection, and HIV awareness trainings for the selected peer leaders. We then conducted a follow-up survey to assess how many youth received information about HIV. While CHANGE only collected data from 18% of youth in the network, the peer leaders that it selects successfully reached 80% of the youth. This is comparable to previously tested algorithms HEALER and DOSIM which gather the *entire* network. This result provides evidence that CHANGE can obtain influence spread comparable to the highly sophisticated algorithms proposed by previous work, while eliminating crucial barriers to real world deployment.

Third, we give an analysis of the real network data to explain why CHANGE can succeed while gathering such a small portion of the network. Our explanation draws on *friendship paradox*, a phenomenon observed in social networks where a typical node's neighbors have more ties than the node itself. We demonstrate this phenomenon occurs across both of the networks that we gathered and show how CHANGE exploits it to produce sampled networks which are substantially more informative for influence maximization than a comparable number of uniformly random samples.

3.1 Related Work

Influence maximization was introduced by Kempe et al. [KKT03], and has been extensively studied since then [CWW10, TXS14, CDPW14, GLL11a, JHC12, GLL11b, LBNZ17, MN11]. Most work has focused on algorithms which are scalable to extremely large networks, primarily in the context of online viral marketing. Recently, HIV prevention (and preventative health more broadly) has emerged as a new application area for influence maximization which brings its own set of research challenges. Yadav et al. [YCXJ$^+$16] proposed HEALER, a POMDP-based algorithm for selecting influential peer leaders. Subsequently, Wilder et al. [WYI$^+$17b] introduced the DOSIM algorithm which uses robust optimization to account for uncertainty about the true probability of influence propagation. Our approach to parameter robustness is similar to techniques in robust MDP planning [KVM$^+$12], though the domains are entirely different.

Yadav et al. [YWR$^+$17] conducted a real-world pilot study of HEALER and DOSIM, and found that both algorithms significantly outperformed the status-quo heuristic used by agencies (selecting high-degree nodes). However, neither algorithm addresses any of the challenges described above. Both assume that the entire social network is provided as input, which is unrealistic in practice due to the enormous effort required. Further, only DOSIM handles uncertainty about the probability of influence spread, and its method for doing so is extremely computationally intensive (see Section 3.3.3). Separate work by Wilder et al. [WIRT18] considered network data collection. They proposed the ARISEN algorithm which samples a portion of youth in the network to collect data from. While ARISEN can be theoretically analyzed for certain network structures, it is not practically suitable to deployment because it relies on querying a sequence of specific youth who may be difficult to locate (see Section 3.3.2). Moreover, ARISEN does not consider either parameter uncertainty or execution errors (the possibility that some peer leaders will not attend), both of which we incorporate into CHANGE.

3.2 Problem description

Motivating domain: Our work is designed to overcome the challenges in deploying influence maximization techniques to support community-driven interventions. We are specifically motivated by the challenge of raising awareness about HIV among homeless youth. Typically, an HIV awareness intervention will be provided by a drop in center or other organization which serves homeless youth. Each intervention is a day-long class followed by weekly hour-long meetings. Hence (as is typical in many intervention domains), the service provider will almost never have enough resources to deliver the intervention to all of the youth that frequent the center; instead, the intervention is usually delivered to 15-20% of the population[1]. Further, limitations on space and personnel mean that the intervention can typically be delivered to only 4-6 youth at a given time, so the training is broken up over a series of small sessions. These youth are trained as *peer leaders* who communicate with other youth about HIV prevention. This amplifies the reach of the intervention through the social network of the homeless youth. The question is which youth will make the most effective peer leaders, able to reach the greatest number of their peers. This is an *influence maximization* problem, which we now formalize.

Influence: The youth have a social network represented as a graph $G = (V, E)$. Each youth is initially inactive, meaning that they have not received information about HIV prevention. Once nodes are activated by the intervention, they have a chance to influence their peers. We model this process through a variant on the classical independent cascade model (ICM) which has been used by previous work on HIV prevention and better reflects realistic time dynamics [YCXJ$^+$16, WYI$^+$17b, YWR$^+$17]. The process unfolds over discrete time steps $t = 1...T$, where T is a time horizon. There is a propagation probability p. When a node becomes active, it attempts to activate each of its neighbors. Each attempt succeeds independently with probability p. Activation attempts are made at each time step until either the neighbor is influenced or the time horizon is reached.

Note that the assumption that p is uniform across edges is without much loss. As noted by He and Kempe [HK16], a uniform p is equivalent to each edge drawing an individual propagation probability i.i.d. from a distribution with mean p. This is because the following processes are analytically equivalent: (1) propagate influence with probability p and (2) draw a propagation probability q from a distribution with $\mathbb{E}[q] = p$ and then propagate

[1] Note that while CHANGE directly surveys ∼18% of youth, they name others as friends, resulting in a larger sampled graph.

influence with probability q. Hence, our model subsumes *any* stochastic model where the probabilities are drawn from a common prior.

Interventions: At each time step $t = 1...T$, the algorithm selects a seed set A_t containing up to K nodes. However, each seed node may or may not actually attend the intervention. This problem is particularly acute with homeless youth since a number of factors could prevent a given youth from attending (e.g., being arrested, running out of money for a bus ticket, etc.). Hence, we assume that each node v has a hidden state $x_v \in \{present, absent\}$. Each node's state is drawn independently from some prior distribution D. For simplicity, we will take D to set each node to be present with probability q. However, all of our analysis applies to arbitrary distributions. For each $v \in A_t$, if $x_v = present$, then v is activated. Nothing occurs if $x_v = absent$. Note that an absent node can still become activated by others, since they may still be in contact with others in the social network. After the set A_t is chosen, the intervention occurs and the hidden state of each $v \in A_t$ is observed. We denote the set of all observations received at time t as O_t.

The algorithm may use this information to plan the next intervention. In other words, the problem is *adaptive*. To model adaptivity, we introduce the notion of a *policy*. A policy maps from past actions and observations to the action that should be taken next. Let $\mathcal{A} = \{S \subseteq V : |S| \leq K\}$ be the set of all possible actions. A *history* is the current sequence of actions chosen and observations received, denoted by $\psi_t = ((A_1, O_1), (A_2, O_2), ...(A_t, O_t))$. Let Ψ be the set of all possible histories. A policy is a mapping $\pi : \Psi \to \mathcal{A}$. Let $A(\psi_t) = (A_1...A_t)$ be the sequence of actions taken and $O(\psi) = (O_1...O_t)$ be the corresponding observations (whether each peer leader was present or absent). Recall that youth are trained in groups of 4-6; the policy selects a group of youth to invite given who was trained previously. We denote the objective as $f(A(\psi)|O(\psi))$. f is the expected number of nodes influenced by the seed nodes in $A(\psi)$ conditioned on the observations in $O(\psi)$. We overload notation and let $f(\pi) = \mathbb{E}_{\psi \sim \pi}[f(A(\psi)|O(\psi))]$ be the expected reward from running policy π, where the expectation ranges over the hidden state $\boldsymbol{x}$ (which determines π's actions) as well as the influence process. We seek a policy maximizing $f(\pi)$.

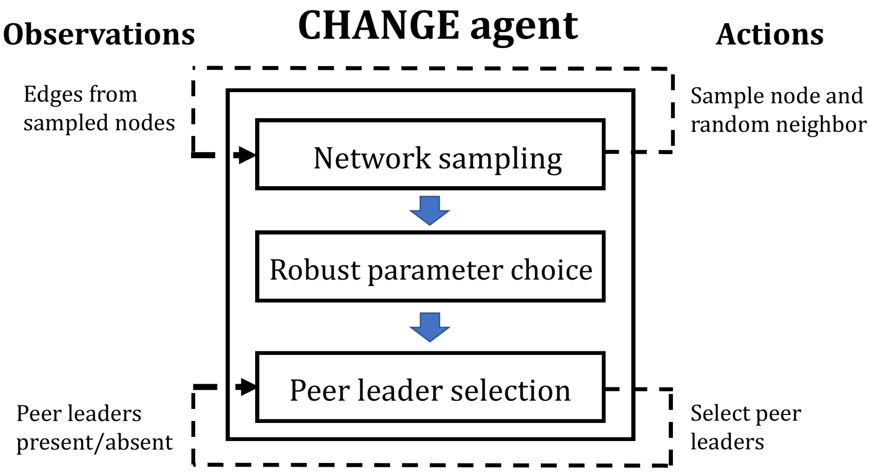


Figure 3.1: *Illustration of the CHANGE agent.*

Uncertainty about network structure and parameters: We consider extensions to the core adaptive influence maximization problem which account for the lack of information endemic in field deployments. First, we consider the case where the structure of the network (the edges E) are unknown. To address this challenge, we give our agent a budget of M queries to run before conducting the intervention. Each query may target either a uniformly random node, or the neighbor of a node already queried. When a node is queried, it reveals all of its edges. The goal is to use the M queries to uncover a set of edges which suffice to identify influential nodes.

We then consider an unknown propagation probability. Here, we take a robust optimization approach and look for a policy which performs well across a range of possible values for p. More detail on this part of the problem can be found in Section 3.3.3.

3.3 CHANGE: a new agent for influence maximization in the field

We now introduce the CHANGE agent for end-to-end influence maximization. Figure 3.1 illustrates the three components of the agent. We start with the last component, peer leader selection, since the other components exist to provide the data that the peer leader selection algorithm requires. Peer leader selection is performed by an adaptive greedy algorithm (Algorithm 6), which handles the chance that some peer leaders may not attend the intervention and plans solutions using the observations obtained so far. Algorithm 6 requires as input a (sample of) social network and a propagation probability p. Algorithms 7 and 8 provide these inputs.

3.3.1 Adaptive greedy planning

Algorithm 6 Adaptive greedy

```
for $t = 1...T$ do
    $A_t = \emptyset$
    for $k = 1...K$ do //greedily select seeds for action $t$
        $v = \arg\max_{v \in V} \Delta(A_t \cup \{v\} | \psi_{t-1}) - \Delta(A_t | \psi_{t-1})$
        $A_t = A_t \cup \{v\}$
    end for
    execute $A_t$ and observe $O_t$
    $\psi_t = \psi_{t-1} + (A_t, O_t)$ //add action/observation to history
end for
```

Given as input the graph G and propagation probability p, finding the optimal policy is a difficult planning problem. There are 2^n possible hidden states and $\binom{n}{K}$ possible actions. While it is possible to formulate the problem as a POMDP, these exponentially large state and action spaces place even small instances beyond the reach of off-the-self solvers. Hence, we exploit the structure of the problem to formulate a scalable greedy algorithm which obtains (provably) near-optimal solutions.

Pseudocode for adaptive greedy, our online planning algorithm, can be found in Algorithm 6. Algorithm 6 selects the action at each step which maximizes the expected gain in influence spread, conditioned on the observations received so far. Then, it waits until this action has been executed, observes which peer leaders attended the intervention, and greedily plans the next step. Formally, let $\Delta(A_t|\psi_{t-1}) = f(A(\psi_{t-1}) \cup A_t|O(\psi_{t-1})) - f(A(\psi_{t-1})|O(\psi_{t-1}))$ denote the expected marginal gain to selecting A_t at time t. The greedy policy is to select $A_t = \arg\max_{|A| \leq K} \Delta(A|\psi_{t-1})$ (the outer loop of Algorithm 6). However, computing the maximizing action is itself computationally intractable (as there are $\binom{n}{K}$ possible choices). Hence, Algorithm 6 uses an additional greedy inner loop which greedily selects the elements of A_t one at a time (lines 3-5). Note that Δ can be computed by averaging over random simulations over both the hidden state (which nodes are present/absent) as well as how influence spreads via the ICM.

We prove the following theorem, which shows that greedy planning is sufficient to obtain a guaranteed approximation ratio:

Theorem 9. *Let π_G be Algorithm 6's greedy policy and π_* be an optimal policy. It holds that $f(\pi_G) \geq \left(\frac{e-1}{2e-1}\right) f(\pi_*)$.*

A proof may be found in the supplemental material. We use the *adaptive submodularity* framework of Golovin and Krause, which generalizes the classical notion of a submodular set function to adaptive policies. Their framework does not directly apply to our problem since our algorithm selects a *sequence* of actions, not a set. The order in which actions are selected matters since peer leaders who are selected earlier will have more time to influence others. We show that our problem can be reformulated as maximizing an adaptive submodular set function subject to a more complex set of constraints (a partition matroid). *This is the first approximation guarantee for adaptive influence maximization under execution errors, which is a well-known challenge in domains such as ours [WYI+17b, YWR+17].*

3.3.2 Network collection

Algorithm 7 Network sampling

```
input:  vertex set $V$, budget $M$
$E = \emptyset$ //set of edges observed
$S = \emptyset$ //set of nodes surveyed
for $i = 1...\frac{M}{2}$ do
    Sample $v$ uniformly at random from $V \setminus S$
    $S = S \cup \{v\}$
    $E = E \cup \{(v, u) : u \in N(v)\}$
    Sample $u$ uniformly at random from $N(v) \setminus S$
    $E = E \cup \{(u, w) : w \in N(u)\}$
    $S = S \cup \{u\}$
end for
return $E$
```

The adaptive greedy algorithm assumes that the graph G is fully specified. However, in order for an intervention to deployed in practice, the social network needs to be laboriously gathered by interviewing the entire population of homeless youth (potentially hundreds of youth in total). This is not practical for a service provider to carry out on their own. We present an approach (Algorithm 7) which randomly samples a small number of youth to survey. Our procedure is easy for a service provider to implement in the field without much computational assistance. This simplicity is enabled by underlying insights about the structure of homeless youth social networks, which may assist with intervention design in other vulnerable populations.

We assume that the service provider has the ability to survey up to M youth. Each youth, when surveyed, reveals all of their edges. Algorithm 7 chooses $\frac{M}{2}$ nodes uniformly at random from the population to survey (line 5). For each surveyed node, it choses a uniformly random neighbor to survey as well (line 8). Lastly, it returns the graph consisting of the reported edges. The intuition for why this procedure succeeds is that it leverages the *friendship paradox*: a phenomena where a random node's neighbor has more friends, on average, than the node itself. Essentially, high-degree nodes are overrepresented when we sample a random neighbor instead of a uniformly random node. Thus, Algorithm 7 is disproportionately likely to find central nodes in the network who will reveal many edges and may be good potential seeds. We elaborate using empirical data from our pilot studies in Section 3.5.4.

We contrast here our sampling procedure with the previously proposed algorithm for influence maximization with an unknown network, ARISEN [WIRT18]. ARISEN simulates a random walk by starting at a random node, moving to a random neighbor of the first node, then to a random neighbor of the second and so on. Its motivation is very different. It exploits community structure, where nodes form densely connected subgraphs which are only loosely connected to the rest of the network. ARISEN uses each walk to estimate the size of the community that it lies in and attempts to seed large communities. By contrast, Algorithm 7 leverages a distinct structural property (the friendship paradox). This shift is motivated by practicality. In the feasibility study, only 53% of contacts listed by youth could be located at the center. Hence, it is relatively easy to find at least one contact, as prescribed by Algorithm 7, but much harder to reach a chain of 5-10 youth as in ARISEN.

3.3.3 Parameter robustness

Algorithm 8 Robust parameter selection

```
input: parameter values $p_1 ... p_L$
for $i = 1 ... L$ do
    for $j = 1 ... L$ do
        $g(p_i, p_j)$ = value obtained by Algorithm 6 using $p_i$ evaluated under $p_j$
    end for
end for
return $\arg\max_{i=1...L} \min_{j=1...L} \frac{g(p_i, p_j)}{g(p_j, p_j)}$
```

A further complication is that the adaptive greedy algorithm assumes that the propagation probability p is known, in order to calculate the marginal gain Δ. However, p is never known precisely in practice; each intervention takes months to deploy so we are unlikely to observe the many repeated cascades needed to for learning-based approaches. Previous work has attempted to resolve this dilemma via *robust influence maximization* [HK16, CWW10, LVK16, WYI$^+$17b] which finds a seed set which performs well in the worst case over an uncertainty set of possible parameters. However, the only previous work which addresses robust influence maximization in an adaptive domain is the DOSIM algorithm. DOSIM requires hours or even days of runtime on a high-performance computing cluster because it needs to brute force over a grid of possible parameter settings. Such computational expense is far beyond the capabilities of the average service provider, motivating the development of lightweight but effective heuristics for robust influence maximization.

Algorithm 8 gives the heuristic used by CHANGE. It searches for a good nominal value of the parameter p, which (when given to Algorithm 6) will result in high performance no matter what the true value of p actually is. We first discretize the interval $[0, 1]$ into L points $p_1 ... p_L$. Let $g(p_i, p_j)$ denote the expected influence obtained when we run adaptive greedy planning based on propagation probability p_i, but the true parameter is p_j. We then find $p^* = \arg\max_{i=1...L} \min j = 1...L \frac{g(p_i,p_j)}{g(p_j,p_j)}$. Here, $\frac{g(p_i,p_j)}{g(p_j,p_j)}$, is the ratio of the value based on planning with parameter p_i to the value that could have been obtained if we new the true parameter p_j. p^* is the parameter which maximizes the worst-case value of this ratio. Notably, this requires only L^2 runs of adaptive greedy; we take $L = 10$ in practice. By contrast, DOSIM requires $O\left(\frac{n}{\epsilon}\right)^3$ runs of a greedy algorithm to achieve approximation error ϵ. This quickly reaches thousands (or tens of thousands) of runs even for moderately sized networks and requires high-performance computing resources.

We investigate the performance of this heuristic on two real homeless youth social networks, Network A and Network B [YCXJ+16, WYI+17b]. Both were gathered from youth at a different drop-in center and contain approximately 150 nodes. Table 3.1 shows $\frac{g(p_i,p_j)}{g(p_j,p_j)}$, the percentage of optimality, for several combinations of p_i and p_j. For instance, the entry for Network A in the row corresponding to 0.2 and the column corresponding to 0.01 indicates that when adaptive greedy plans on $p = 0.2$, but the true parameter is actually $p = 0.01$, it obtains 88.7% of the optimal value possible. In both networks, Algorithm 8 selects $p^* = 0.2$ as the optimal choice: it has value at least 88.7% of the optimum under all parameter combinations in Network A and value at least 92.9% of the optimum on Network B. While this still improves on a naive choice which ignores robustness, we observe that all of the values in the table are relatively high. This indicates that influence maximization in this domain may not be highly sensitive to the exact choice of parameter.

To explain this phenomenon, Figure 3.2 shows the seed set chosen for Network A under different values of p. We observe a clear trend: with low p, the seeds are clustered more tightly together in the core of the network, and as p grows an increasing fraction of the seeds move to the periphery of the network. Intuitively, when p is high, a few seed nodes suffice to influence the core of the network. Thus, the greedy algorithm extracts higher marginal return by using seed nodes to cover outlying regions which are less likely to have been reached from the core. $p = 0.2$ represents a "goldilocks" solution where the core of the network is heavily covered without being oversaturated, and hence performs well across many values of p. However, other parameter choices can still do well because the majority

Table 3.1: *Percentage of optimum obtained by planning based on parameter on row, when true parameter is given by column.*

	Network A				Network B			
p	0.01	0.2	0.5	0.8	0.01	0.2	0.5	0.8
0.01	100	81.0	83.4	88.1	100	86.8	88.3	89.7
0.2	88.7	100	97.0	96.8	93.2	100	95.6	92.9
0.5	85.5	95.7	100	98.8	88.6	96.9	100	97.1
0.8	84.9	93.1	97.8	100	89.1	92.0	99.3	100

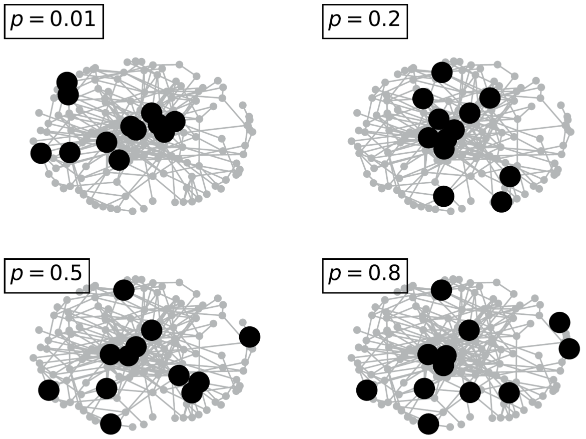


Figure 3.2: *Seeds chosen under different values of p.*

of the possible value is located in the core of the network, which all seed sets devote several seeds to.

3.3.4 Simulation experiments

We now examine the performance of the CHANGE agent in a series of experiments using real-world data collected from homeless youth populations at different drop-in centers. We use networks collected from our own and previous pilot studies. The first network is the one we collected from the youth enrolled for CHANGE's intervention study. The other two networks were gathered by Yadav et al., also from real homeless youth, for their pilot studies of the HEALER and DOSIM algorithm. The main question is whether CHANGE is able to find influential seed nodes while only surveying a small fraction of the network. We ran CHANGE in simulation on each of the real-world networks, querying $M = 12$ nodes

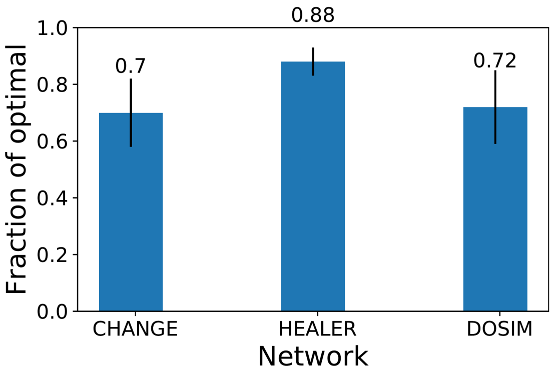


Figure 3.3: *Simulated influence of CHANGE compared to adaptive greedy run on the full network. The x axis denotes which pilot study the network is taken from.*

to obtain a sampled graph. This is 15-20% of the number of nodes in each network. Then, CHANGE selected $K = 4$ seed nodes in each of $T = 3$ rounds (reflecting the setup used in the intervention study). We conducted 30 independent trials for each network.

Figure 3.3 compares the number of non-peer leaders reached by CHANGE compared to the number reached by adaptive greedy (Algorithm 6) when it was given the entire network in advance. We also tried comparing to the DOSIM agent [WYI$^+$17b] and obtained near-identical results. We see that CHANGE obtains 70-88% of the influence spread which is achievable if we knew the entire network in advance (comparable to previous work on network sampling [WIRT18]). However, CHANGE surveyed only 15-20% of the nodes in the network. This simulation, conducted on networks gathered from real homeless youth populations, provides evidence that CHANGE can find influential peer leaders using only a small amount of data.

3.4 Pilot study procedure

The major contribution of this work is carrying out a pilot study which tests the CHANGE agent in a field deployment at a real drop in center serving homeless youth in a major U.S. city. Here, we outline the procedure followed for the pilot study. There were two studies, the feasibility study and the intervention study. In the feasibility study, we just tested the

Table 3.2: *Number of youth recruited, trained, and retained for follow-up in each study. CHANGE refers to the study conducted in this work to test the CHANGE agent. The other columns are taken from Yadav et al. [YWR+17], who conducted pilot tests of HEALER and DOSIM.*

	CHANGE	HEALER	DOSIM
Youth recruited	64	62	56
Queried for links	18.75%	100%	100%
PL trained	15.6%	17.7%	17.85%
Retained	54.7%	73%	73%

first component of CHANGE (network data collection) to validate that it works in practice to gather high-quality data. In the intervention study, we carried out actual interventions with homeless youth at the center. This step used all three steps of the CHANGE agent: we gathered the network, found a robust set of parameters, and then carried out interventions.

For each of the studies, we enrolled (respectively) 72 and 64 youth. Each youth was paid $20 to enroll in the study (all monetary incentives were the same as prior studies [YWR+17]). We ran CHANGE's data collection mechanism, randomly sampling a subset of youth to query for ties. Each youth who enrolled was also asked to complete a baseline survey. As part of this survey, we also gathered the *full* network consisting of ties from all of the youth. *We emphasize that this data was collected just for analysis. We did not use the full network to plan interventions, and we would not expect an agency to conduct this step in a regular deployment.* In the feasibility study, we also gathered edges via field observations and interviews with agency staff in order to validate our data collection via comparison to alternate mechanisms (see Section 3.5.1).

In the intervention study, social workers delivered the *Have You Heard* intervention, previously published in the public health literature [RTC+12]. The social workers conducted a day-long class with the selected youth, covering HIV awareness and prevention, and training the youth as peer leaders to communicate with others at the agency. Peer leaders were paid $60. Three sets of peer leaders were selected by CHANGE, with approximately 4 peer leaders in each set. This matches the number used in previous influence maximization pilot studies [YWR+17]. Table 3.2 reports specific values on the number of youth enrolled, queried for edges, and trained as peer leaders for our pilot test as well as pilot tests of previous algorithms. One month after the start of the study, we conducted a follow up survey with all of the youth who initially enrolled. Some youth were lost to follow up (see Table 3.2). We asked the youth whether they had received information about HIV prevention from a peer who was part of the study. Youth were paid $20 to respond to the follow up survey. We emphasize that all aspects of the intervention study (the training materials for peer leaders, survey instruments, etc.) are identical to Yadav et al. [YWR$^+$17], so our results are directly comparable.

3.5 Pilot study results

3.5.1 Feasibility study

We address two questions in the feasibility study. First, can Algorithm 7 (CHANGE's network sampling) be implemented with homeless youth? Second, is the resulting self-reported data accurate enough for influence maximization?

The challenge in the first question is that homeless youth can be difficult to locate. However, we were able to locate at least one neighbor for at least 80% of youth queried who were not isolates (i.e., named at least one neighbor). We conclude that Algorithm 7 is feasible for homeless youth populations. When no neighbor could be located, we drew a new random youth.

We now turn to the second question, which is of broader interest. Previous work on influence maximization in the field uses primarily self-reported network data [YCXJ$^+$16, WYI$^+$17b, YWR$^+$17]. Note that gathering ties from social media has proven unreliable for homeless youth populations both due to limited access to social media websites and mismatch between social media ties and true relationships. More broadly, self-reported network data is the best available to researchers in many field settings [Bre00]. However, self reported ties are subject to their own limitations (forgetfulness, reticence, etc. [Bre00]). *To our knowledge, no previous work has validated whether self-reported ties suffice for influence*

Table 3.3: *Number of edges gathered by each method and the percentage overlap with edges gathered via self-report.*

	Self-report	Observed	Staff	All
Number edges	51	23	46	112
Overlap with self-reports	100%	8.7%	13%	40%

maximization. Our results show that self-reported data has important limitations (many edges discovered by other means were not self-reported), consistent with a large literature on network data collection methods [Bre00]. However, self-reported data sufficed to find near-optimal seed sets despite these limitations.

We gathered data via several methods: traditional self-reporting, field observations by the research staff, and interviews with staff members at the agency. This yielded three distinct sets of edges. Figure 3.4 shows the three networks, along with the composite graph obtained by combining edges from all three data sources. We see that self-reports give a fairly accurate global picture of the network. However, the other two data sources fill in many specific edges omitted in the self-reported data. Table 3.3 gives the number of edges gathered by each method, and the percentage of those edges which were contained in the self-reported data. We see that a high level of disagreement between the data collection methods on the status of individual edges: only 8.7% of ties from field observations and 13% of ties reported by staff members were reported by the youth themselves. In total, field observations and staff reports uncovered 69 edges, compared to 51 reported by the youth (with little overlap between the two). This is consistent with prior knowledge: a review of research on network data collection shows that anywhere from 10-80% of edges may be forgotten in self-reported data [Bre00]. Another study comparing self-reported ties to observed interactions found that the two data sources were moderately correlated (median $r = 0.51$), but far from identical [GFCX03].

While many ties may be absent in self-reported data, our ultimate objective is to find influential nodes (not reconstruct the network for its own sake). Hence, we now assess the robustness of influence maximization to missing edges. Given the propensity for forgetting

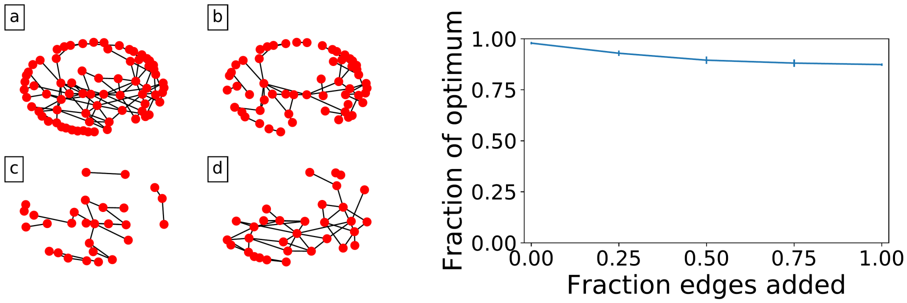


Figure 3.4: *Left: Networks gathered using different methods. (a) All methods combined. (b) Self reported ties. (c) Field observations. (d) Staff observations. Right: Fraction of optimal value obtained using self-reported data as additional edges are added. Error bars show one standard deviation.*

in self-reported data, we conclude that all edges which *are* self-reported do exist [Bre00], but many existing edges are not self-reported. Nevertheless, it is unlikely that all of the edges observed by field researchers or staff truly exist since reports by outside observers are typically less reliable than self-reports [GFCX03]. Thus, we conduct a simulation experiment in which a randomly selected portion of the non self-reported edges are added to the graph.

Figure 3.4 shows the performance of the greedy algorithm as the number of edges added increases. Each point on the x axis represents a fraction of edges which were observed by either field researchers or staff, but not reported by the youth themselves, to add to the graph. E.g., the point 0.25 indicates that a random set comprising 25% of edges which were not self reported are added to the self reported edges to obtain the final graph. Each point averages over 30 draws for this random set. The y axis shows the fraction of optimality obtained by running the adaptive greedy algorithm on just the self reported network. We approximate the optimum by running adaptive greedy on the full network, representing the best possible under full information. The values are consistently high, with very low standard deviation. Even when all of the unreported edges are added, so adaptive greedy does not know about the majority of edges in the graph, it still obtains at least 87% of the optimal value. In reality, not all of the unreported edges are real links, so we would expect even better performance in practice. We conclude that even though self-reported data may miss some edges, it still suffices to identify the influential nodes.

3.5.2 Intervention study

We now turn to our second pilot study, which tested the entirety of the CHANGE agent. In this study, we recruited a separate population of 64 homeless youth from a drop-in center. Table 3.2 gives the total number of youth recruited for different activities, as well as the corresponding figures for previous pilot tests of the HEALER and DOSIM algorithms by Yadav et al. [YWR$^+$17]. We gathered the full social network from all 64 youth, and in parallel ran Algorithm 7 with a budget of $M = 12$ youth to collect a sampled network (querying 18.75% of youth in total for links). *Only the sampled network was used to plan interventions; the full network was gathered only for analysis.* We then ran the CHANGE policy for three steps, training 10 total peer leaders (15.6% of the network). This percentage is comparable to previous studies (HEALER and DOSIM trained approximately 17% of the network each). However, HEALER and DOSIM used the entire network to plan their intervention, compared to the 18.75% of sampled youth used by CHANGE. At one month, we conducted a follow-up survey to assess whether youth received information about HIV prevention from the peer leaders. 54.7% of youth were retained in the follow-up survey, which is a somewhat lower percentage than in previous studies. Nevertheless, we obtain a population of 34 youth who provided follow-up data.

3.5.3 Influence spread results

We now present our core result: the number of youth who received a message about HIV prevention. We examine the percentage of youth in the follow-up group who were not peer leaders (and hence eligible to become influenced) who reported receiving information. Figure 3.5 shows this percentage for our pilot study of CHANGE as well as the percentages reported by Yadav et al. [YWR$^+$17] in their pilot studies of the state of the art algorithms HEALER and DOSIM. CHANGE reached 80% of non-peer leaders compared to approximately 70% for each of HEALER and DOSIM. *Thus, CHANGE was able to reach just as many youth while gathering data from only 18.75% of the network.* The 10% difference between CHANGE and HEALER/DOSIM could be attributable to random variation; we do

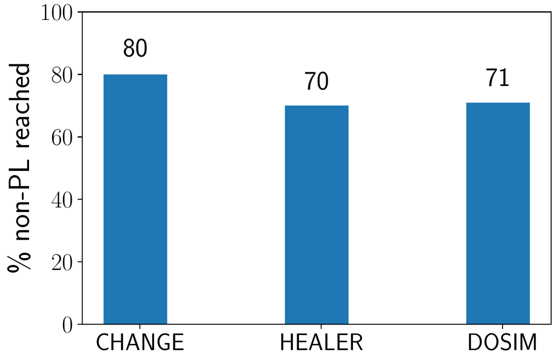


Figure 3.5: *Percentage of youth who were not peer leaders reached by each algorithm in its respective real-world pilot test.*

not claim that CHANGE is actually more effective than algorithms which gather the entire network. Nevertheless, this result provides empirical evidence that CHANGE can perform comparably to existing state of the art influence maximization agents while drastically reducing the amount of data required.

We now take steps to ensure that our results are not an artifact of a difference between the structures of the different networks from each pilot test or of random variation. First, we recall our simulation results in Figure 3.3, which indicate that CHANGE performs competitively with algorithms which are given the entire graph on three different real-world networks. Second, Table 3.4 shows a range of statistics for each network. CHANGE's networks is fairly similar to that of HEALER and DOSIM. However, it is somewhat sparser: its density (the fraction of possible edges which are present) is 0.043 compared to 0.079 for HEALER and 0.059 for DOSIM. This translates into somewhat longer average path lengths and larger diameter. However, sparser structure should only work *against* CHANGE since there are fewer edges along which influence can propagate. Hence, it is unlikely that CHANGE's strong performance is attributable to anomalous network structure.

3.5.4 Explaining CHANGE's success

In this section we attempt to explain why CHANGE can find seed sets which have near-optimal influence spread by surveying only a small fraction of youth. The intuitive explana-

Table 3.4: *Aggregate network statistics for the complete network in each algorithm's pilot study. "Diameter" is the diameter of the largest connected component.*

	CHANGE	HEALER	DOSIM
Diameter	12	8	8
Density	0.043	0.079	0.059
Avg. path length	4.88	3.38	3.15
Avg. clustering coeff.	0.221	0.397	0.195
Modularity	0.654	0.568	0.568

tion for this is a property that many social networks are known to possess: the friendship paradox [Fel91, UKBM11, HKL13]. Specifically, a randomly chosen neighbor of a given node is likely to have higher degree than the node itself. Our algorithm leverages the friendship paradox by surveying both a random node and a randomly chosen friend of that node.

Figure 3.6 plots two quantities for the networks collected in the feasibility and intervention studies. First, the degree distribution. Second, the distribution of the degree of a randomly chosen neighbor of a randomly chosen node. This is the degree distribution of the nodes that Algorithm 7 samples in its second step. We see that the neighbor degree distribution is skewed towards higher degrees. In the feasibility network, the mean degree is 3.11 while the mean friend's degree is 4.56. In the intervention network, the mean degree is 2.98 while the mean friend's degree is 4.04. This suggests that by querying a random neighbor of each node, our algorithm is able to preferentially locate nodes who are useful in two ways. First, high degree nodes provide more information about the network. Second, they are more likely to be influential peer leaders and may serve as a useful set of candidates which adaptive greedy can refine.

3.6 Discussion and conclusion

This chapter presents the CHANGE agent for influence maximization, a multiagent problem with many applications in preventative health and other domains. CHANGE addresses major barriers to the deployment of influence maximization by service providers through a

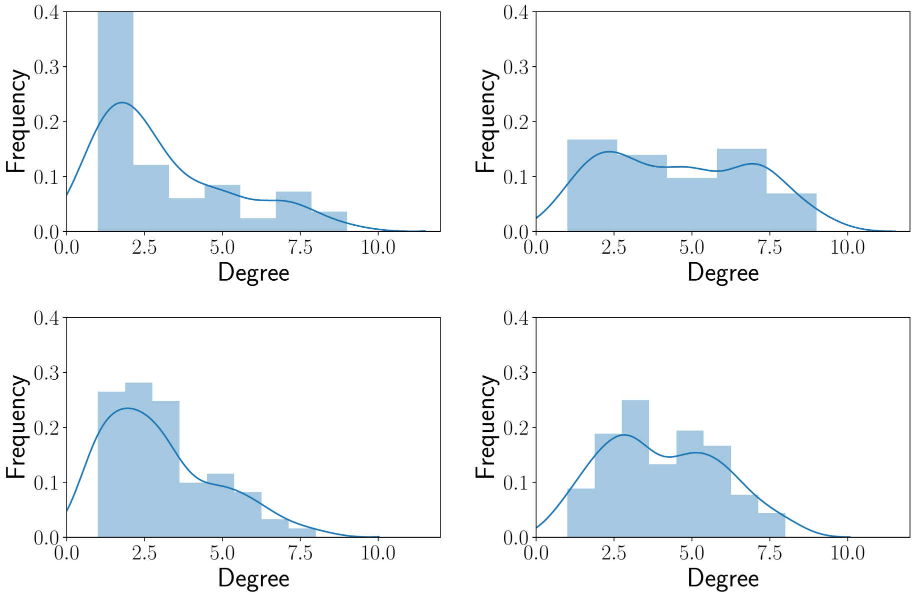


Figure 3.6: *Degree distributions. Top row: feasibility study. Bottom: intervention study. Left: standard degree distribution. Right: degree of a random neighbor.*

series of algorithmic contributions, backed by simulation results on real-world networks. We then conducted a real-world pilot study of CHANGE with a drop-in center serving homeless youth, the first such pilot study of sampling-based influence maximization and only the second study testing *any* influence maximization agent in the real world. CHANGE obtained comparable influence spread to previously field tested algorithms, but surveyed only 18% of youth to obtain network data. CHANGE has empirical promise in delivering high-quality influence maximization solutions in a manner which can be feasibly implemented by a service provider.

While the algorithms underlying CHANGE are easy to implement, they draw on a series of insights into the social behavior of homeless youth. One lesson learned is that, to be successful in the field, algorithms must be designed with their target population and setting in mind. CHANGE both navigates challenges specific to homeless youth (e.g., the difficulty of locating youth to query for edges or serve as peer leaders) and leverages properties of their social network (the friendship paradox). Our experience shows that accounting for both challenges and opportunities in the target population is crucial to produce a practically deployable algorithm.

Chapter 4

Field trial of an AI-augmented intervention for HIV prevention among youth experiencing homelessness

Each year, approximately 4.2 million youth in the United States experience some form of homelessness [MDM$^+$18]. One of the key health challenges for this population is high HIV prevalence, with reported prevalence in the range of 2-11% [YR11], up to 10 times that for youth with stable housing [Nat12].

One proposed mechanism for fostering behavior change in high-risk populations is the *peer change agent* model. The main idea is to recruit peer leaders from the population of youth experiencing homelessness (YEH) to serve as advocates for HIV awareness and prevention. Use of peer leaders has been suggested in the public health and social science literature due to the central role that peers play in risk behaviors for YEH, including related to HIV spread [GJdlHTG13, RMBAY10, RBAMM12]. Indeed, peer change agent models have succeeded in past HIV prevention interventions in other contexts [MKOS09]. However, there have also been notable failures [G$^+$10], and it has been argued such failures may be attributable to how peer leaders are selected [SZL15]. The long-standing and most widely adopted method in the public health literature for selecting peer leaders is to identify the most popular individuals in the social network of the youth [KMS+97] (formally, the highest degree nodes). This poses the question: are high-degree youth the best peer leaders to disseminate messages about HIV prevention? This question has relevance far beyond HIV prevention; analogous social network interventions are used widely across development, medicine, education, etc. [KHS+15, PSA16, BCDJ13, VP07].

Information dissemination on social networks is the focus of a long line of research in computer science. In particular, the *influence maximization* problem, formalized by [KKT03], asks how a limited number of seed nodes can be selected from a social network to maximize information diffusion. Influence maximization has been the subject of extensive work by the theoretical computer science and artificial intelligence communities [CWY09, CWW10, GLL11a, BBCL14, TXS14]. However, to our knowledge, no work prior to this project had connected the computational literature on influence maximization to the use of network-driven interventions in public health and related fields. Computational work has mainly focused on developing highly efficient algorithms for use on large-scale social media networks (often motivated by advertising), while interventionists in health domains have not used explicitly algorithmic approaches to optimize the selection of peer leaders. Previous computational work assumed access to data (e.g., the full network structure and a model of information spread) which are simply not available in a public health context.

This chapter reports the results of a project which bridges the gap between computation and health interventions. As a research team composed of computer scientists and social workers, we developed, implemented, and evaluated an intervention for HIV prevention in YEH where the peer leaders are algorithmically selected. This intervention was developed over the course of several years, alternating between algorithm design and smaller-scale pilot tests to evaluate feasibility. The final system, which we refer to as CHANGE (CompreHensive Adaptive Network samplinG for social influencE), was evaluated in a large-scale field trial enrolling 713 youth across two years and three sites. The trial compared interventions planned with CHANGE to those using the standard public health methodology of selecting the youth with highest degree centrality (DC), as well as an observation-only control group (OBS). *Results from this field trial demonstrate that CHANGE was substantially more effective than the standard DC method at increasing adoption of behaviors protective against HIV spread.* To our knowledge, this is the first empirically validated success of using AI methods to improve social network interventions for health. It is critically important for “AI for Social Good" work to result in deployed and rigorously evaluated interventions, and this chapter provides one such example.

The remainder of the chapter is organized as follows. First, we survey related work from both a computational and application perspective. Second, we introduce a formalization of the problem of selecting peer leaders from a computational perspective. Third, we briefly review the design of the CHANGE system to address this problem (deferring most details to earlier technical publications [WOVH$^+$18, Wil18a, WIRT18]). Fourth, we present the design of the field trial. Fifth, we present and analyze results from the trial. Sixth, we discuss lessons learned over the course of the project which may help inform future attempts to design and implement AI-augmented public health interventions.

4.1 Related Work

A great deal of research in computer science has been devoted to the influence maximization problem. The majority of this has focused on computationally efficient algorithms for large networks [CWY09, CWW10, GLL11a, BBCL14, TXS14] and assumes that the underlying social network and model of information diffusion are perfectly known. There is also more recent literature on algorithms to learn or explore these properties. Predominantly though, such work requires many repeated interactions with the system. For example, algorithms to estimate the parameters of an unknown model of information diffusion [DLBS14, PAH15, NPS15, HXKL16, KSSW18] typically require the observation of hundreds of cascades on the same network. Collecting this amount of data is intractable for public health interventions, where a single round of the intervention takes months. Other work concerns the bandit setting, where the algorithm can repeatedly select sets of nodes and observe the resulting cascade [WKVV17, CWY13, WC17]. Similarly, these algorithms accept poor performance in early rounds as the price for improvement over the long run, but waiting tens or hundreds of rounds for improved performance is not an option in our domain. Such techniques are a much better fit for problems concerning online social networks (for example, in advertising domains) where repeated experiments and large datasets are possible.

The most closely related related computational work to ours concerns a robust version of the influence maximization problem [HK18, CLT$^+$16, LVK16], building on the earlier work of [KMGG08] on general robust submodular maximization problems. Our algorithm for robust submodular optimization, for which an overview is provided below, differs from these approaches mainly in that it solves a fractional relaxation of the problem instead of repeatedly calling a greedy algorithm for discrete submodular optimization, which helps improve computational performance.

There is a large literature on social network interventions in public health [VP07, KHS$^+$15], clinical medicine [YHWH03], international development [CDJS15, BCDJ13], education [PSA16], etc. Common strategies involve selecting high degree nodes (as compared to in our trial), selecting nodes at random, or asking members of the population to nominate others as influencers. The empirical evidence for the relative effectiveness of different strategies is mixed; [KHS$^+$15] reports no or marginal improvement for nominations vs random selections (depending on the outcome measure), while [BCDJ19] report statistically significant improvements for a nomination-based selection mechanism. [CEU18] introduce improved statistical methods to compare the effectiveness of seeding strategies and conclude that nomination-based strategies do not measurably improve performance. Indeed, [AMS18] show that in some theoretical network models it may be preferable to recruit a slightly larger number of influencers at random rather than carefully map the network. We contribute to this literature by developing and empirically evaluating an algorithmic framework which combines both features reminiscent of the nomination-based strategies proposed by others (for gathering information about network structure) as well as robust optimization techniques for jointly optimizing the entire set of influencers who are selected (not part of previous empirically evaluated strategies).

4.2 Problem Description

The population of youth are the nodes of a graph $G = (V, E)$. We seek to recruit a set of youth S to be peer leaders, where $|S| \leq k$. In domain terms, this budget constraint reflects the fact that peer leaders are given a resource-intensive training and support process. The objective is to maximize the total expected number of youth who receive information about HIV prevention, given by the function $f(S)$. Here, f encapsulates the dynamics of a probabilistic model of information diffusion across the network (discussed below). The optimization problem $\max_{|S| \leq k} f(S)$ is the subject of the well-known influence maximization problem. When the objective function f is instantiated using common models for information diffusion, the resulting optimization problem is submodular (i.e., there are diminishing returns to selecting additional peer leaders). While finding an optimal solution is NP-hard, a simple greedy algorithm obtains a $(1 - 1/e)$-approximation [KKT03].

The most common choice for the model of information diffusion is the independent cascade model. In this model, each node who receives information transmits it to each of their neighbors with probability p. All such events are independent. The process proceeds in discrete time steps where each newly informed node attempts to inform each of their neighbors, and concludes when there are no new activations. $f(S)$ calculates the number of nodes who receive information when the nodes S are informed at the start of the process, in expectation over the random propagation.

The standard influence maximization problem concludes here. However, while developing an algorithmic framework applicable to public health contexts, we came across challenges which must be solved before, during, and after the setting imagined in standard influence maximization. These challenges opened up new algorithmic questions, addressed in a series of publications in the AI literature [WOVH$^+$18, Wil18a, WIRT18]. Here, we detail three steps for deploying an influence maximization intervention in the field.

First, information about the network structure G must be gathered. Previous work on influence maximization assumed that the network structure is known in advance. While this assumption may be reasonable for online social networks, we aim to disseminate information through the network consisting of real-world interactions between youth at a given center. Moreover, pilot studies revealed that information from an online social network (Facebook) was a poor proxy for actual connections at the center – not all youth used Facebook, and of those who did, many were not friends with their actual contacts at the drop-in center. Instead, network information must be gathered through in-person interviews where social workers ask youth to list those who they regularly interact with. Collecting data in this manner is time-consuming and expensive, often requiring a week or more of effort on the part of the social work team. Accordingly, the first stage of our algorithmic problem is to decide which nodes to query for network information. The algorithm is allowed to make M queries, where each query reveals the edges associated with the selected node. The queries can be adaptive, i.e., the choice of the ith node to be queried can depend on the answers given by nodes $1...i-1$.

Second, this network information is used to select an initial set of peer leaders. This stage more closely resembles the standard influence maximization problem. However, there is an additional complication that the propagation probability p is not known. Indeed, there is no data source from which it could be inferred (as opposed to online platforms with abundant data; see related work). Instead, we formulate an uncertainty set $\mathcal{U}$ containing a set of possible values for p which are consistent with prior knowledge (in CHANGE, we took $\mathcal{U}$ to be a discretization of the interval [0,1], reflecting limited prior knowledge). The aim is to find a set S which performs near-optimally for every scenario contained in $\mathcal{U}$. Formally, this corresponds to the robust optimization problem

$$\max_{|S|\leq k} \min_{p\in\mathcal{U}} \frac{f(S,p)}{OPT(p)}$$

where $OPT(p)$ denotes $\max_{|S|\leq k} f(S,p)$, i.e., the best achievable objective value if the propagation probability p were known. Normalizing by $OPT(p)$ encourages the algorithm to find a set S which simultaneously well-approximates the optimal value for each $p \in \mathcal{U}$ and avoids the trivial solution where solution to the inner min problem is always the smallest possible value of p. Note that since $OPT(p)$ is constant with respect to S, $\frac{f(S,p)}{OPT(p)}$ remains submodular with respect to S. Robust optimization of submodular functions is substantially more difficult than optimization of a single submodular function; in fact, it is provably inapproximable in general [KMGG08] and the aim is instead to approximate a tractable relaxation of the problem.

Third, after an initial set of peer leaders S is identified, recruitment proceeds in an adaptive manner. Not all youth invited to become peer leaders will actually attend the training session. A number of potential barriers exist, e.g., a given youth could have been arrested or not have had enough money for a bus ticket. Formally, we model that each youth who is invited will actually attend with probability q (based on experience in pilot studies, we took $q = 0.5$), where the attendance of each youth is independent of the others. For a given value of p, the resulting objective function is $f(S, p, q)$, which takes an expectation over both the randomness in which nodes are successfully influenced at the start of the process and in the subsequent diffusion. It is easy to show [WOVH$^+$18] that f remains submodular with this additional randomness. Because of this variation in attendance, as well as capacity limits for the initial training, peer leaders are recruited over multiple rounds, where the peer leaders selected in round t can depend on those who were successfully recruited in rounds $1...t-1$. In each round t, we select a set of peer leaders S_t with $|S_t| \leq k_t$ and observe which nodes are successfully recruited as peer leaders. The process continues for T rounds in total.

4.3 System Design

Our final proposed system for intervention planning is called CHANGE. CHANGE was originally introduced in [WOVH$^+$18]. The final version of CHANGE summarized here is nearly the same as the original, with the exception of the algorithm used for robust optimization, which was separately developed and published in [Wil18a]. We now provide an overview of CHANGE, mirroring the steps of the earlier problem formulation.

Network sampling

CHANGE uses a simple but well-motivated heuristic to select a subset of nodes to be queried for network information (in the discussion section, we briefly review our earlier work on a more theoretically sophisticated solution, and the rationale for choosing a simpler method). The chosen method splits the query budget M into two halves. Each query in the first half is made to a node selected uniformly at random from the network. Each query in the second half follows a query in the first half, and selects a uniformly random neighbor of the first node. This design is motivated by the friendship paradox, the observation that high-degree nodes are overrepresented when we sample random neighbors [Fel91]. Hence, the two stages of the query process balance between competing objectives: the first step encourages diversity, since random sampling ensures that we cover many different parts of the network, while the second step tends towards high-degree nodes who can reveal a great deal of network information.

Robust optimization

We now provide an overview of how CHANGE handles parameter uncertainty within a single stage of the planning process, before considering the multi-stage problem (with uncertain attendance) below. As mentioned above, max-min submodular optimization is NP-hard to approximate (within any nonzero factor) [KMGG08]. Accordingly, we need to somehow relax the problem to obtain meaningful guarantees. Let $\mathcal{I}$ denote the set of all feasible solutions (sets S where $|S| \leq k$) and $\Delta(\mathcal{I})$ be the set of all distributions over $\mathcal{I}$ (i.e., the $|\mathcal{I}|$-dimensional simplex). We developed an algorithm for the problem

$$\max_{D \in \Delta(\mathcal{I})} \min_{p \in \mathcal{U}} \mathop{\mathbb{E}}_{S \sim D} \left[\frac{f(S, p)}{OPT(p)} \right] \tag{4.1}$$

which allows the algorithm to select a distribution over feasible sets and evaluates the worst case only in expectation over this distribution. In game theoretic terms, this allows the algorithm to select a mixed strategy instead of a pure strategy. At run-time, we sample from D; the resulting set has guaranteed performance in expectation over the sampling, but strong guarantees cannot be obtained ex-post for the sampled set (as a result of the computational hardness of the original max-min problem). However, in practice we find that sampling several random sets and selecting the best one gives excellent empirical performance (i.e., closely matching or exceeding the expected value of the distribution).

Our algorithm for this problem, detailed in [Wil18a], uses a compact representation of the space of distributions (keeping track of only the marginal probability that each node is selected instead of each of the exponentially many potential subsets). It solves a fractional relaxation of the discrete max-min problem using this compact representation via a stochastic first-order method which is adapted to the particular properties of the objective. Then, we can use known rounding algorithms for submodular maximization to sample random sets from the distribution encoded by the solution to the fractional relaxation. This procedure guarantees a $(1 - 1/e)^2$-approximation for Problem 4.1, which can be improved to $(1 - 1/e)$ with some additional steps (which we did not find empirically necessary).

Multi-stage intervention with attendance uncertainty

We handle the multi-stage nature of the intervention by running the robust optimization problem at each stage, calculating the objective function in expectation over which peer leaders will attend and conditioning on the selection of those who have attended previous interventions. Formally, this means that at stage $t > 1$, we solve

$$\max_{D \in \Delta(\mathcal{I})} \min_{p \in \mathcal{U}} \mathop{\mathbb{E}}_{S_t \sim D} \left[\frac{f(S_t \cup S_1 \cup ... \cup S_{t-1}, p, q)}{\max_{|S^*| \leq k} f(S^* \cup S_1 \cup ... \cup S_{t-1}, p, q)} \right]$$

where $S_1 ... S_{t-1}$ denote the sets of peer leaders who were succsesfully recruited in each previous stage. It is easy to show that the inner objective f remains submodular in S_t (see [WOVH$^+$18]), and so we retain the earlier guarantees on the quality of the solution obtained at each individual step. Moreoever, in [WOVH$^+$18] we show that the multi-stage problem as

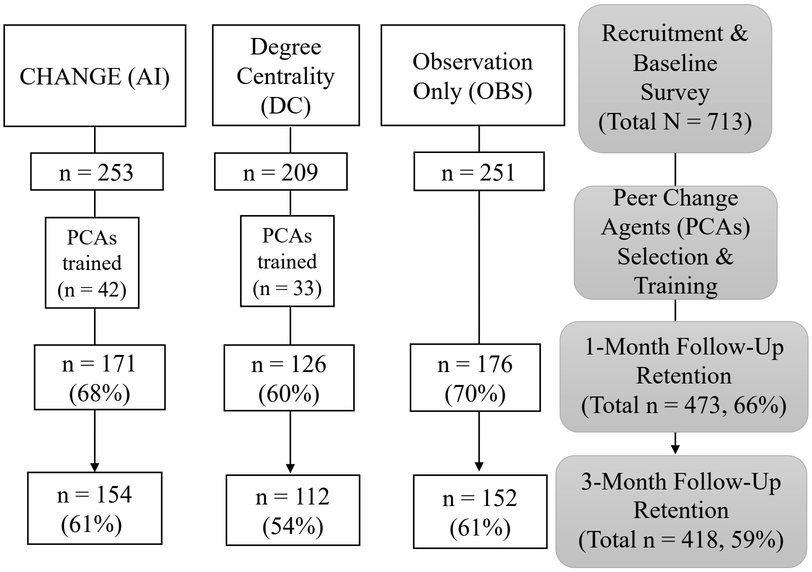


Figure 4.1: *Number of participants recruited and retained in each arm of the study.*

a whole enjoys the property of *adaptive submodularity*, meaning that for any fixed parameter value p, solving

$$\max_{D \in \Delta(\mathcal{I})} \mathop{\mathbb{E}}_{S_t \sim D} \left[\frac{f(S_t \cup S_1 \cup ... \cup S_{t-1}, p, q)}{\max_{|S^*| \leq k} f(S^* \cup S_1 \cup ... \cup S_{t-1}, p, q)} \right]$$

at each step t and selecting the resulting set S_t enjoys an approximation guarantee relative to the optimal adaptive policy for selecting a sequence of sets $S_1 ... S_t$ (again, with respect to a fixed p). More detailed discussion of the theoretical properties can be found in [WOVH$^+$18].

4.4 Study Design

We now move to the empirical portion of the project and provide an overview of the design of the field trial. All study procedures were approved by our institution's Institutional Review Board. The study was designed to compare the efficacy of two different means of selecting peer leaders: the CHANGE system described above and the standard DC approach in public health (selecting the highest-degree youth). We additionally included an observation-only control group (OBS), for three arms in total. The study was conducted at three drop-in centers for YEH in a large US city. Drop-in centers provide basic services to

YEH (e.g., food, clothing, case management, mobile HIV testing). Due to high transience in the YEH population, most clients at a given center leave within approximately six months. Accordingly, we tested each of the three methods at each of the the three drop-in centers (giving nine deployments in total, each with a unique set of youth), ensuring that successive deployments at a given drop-in center were separated by six months. Youth were only allowed to enroll in the study once, so even the small number of youth who were present at the center across multiple deployments were included only on the first time they attempted to enroll. Testing each method at each drop-in center helps account for differences in the demographic and other characteristics of youth who tend to access services at each center.

Each of the nine deployments used the following procedure. Figure 4.1 shows the number of youth recruited and retained for each phase of the study in each arm.

First, youth were recruited at the drop-in center over the course of a week to participate in the study. All participants gave informed consent. Each participant completed a baseline survey which assessed demographic characteristics, sexual behaviors, and HIV knowledge. Demographic characteristics included age, birth sex, gender identity, race/ethnicity, and sexual orientation. Youth were also surveyed about their living situation and relationship status.

Second, peer leaders were selected and trained (for the CHANGE and DC arms of the study). Each individual training consisted of approximately 4 youth and there were 3-4 trainings per deployment (depending on exact attendance). In total, approximately 15% of survey participants in each deployment were trained as peer leaders. In the CHANGE arm of the study, network information was queried from approximately 20% of the participants (sampled according to the mechanism described above). In the DC arm, we used a full survey of the network to find high-degree nodes, in order to give the strongest possible implementation to compare to.

Third, peer leaders had three months to disseminate HIV prevention messages. Peer leaders were supported via 7 weeks of 30-minute check-in sessions with study researchers, which focused on positive reinforcement of their successes as well as problem-solving strategies and goals for the future. All peer leaders attended at least one check-in session, with modal attendance at five sessions. Peer leaders received $60 in compensation for attending the initial training and $20 for each check-in session.

Fourth, follow-up surveys were administered to the original study participants from the first step. Follow-up surveys assessed the same characteristics as the baseline survey. Differences in reported sexual behavior between baseline and follow-up were used as the primary metrics to evaluate the interventions. All such metrics were self-reported; we followed best practices in social science research to minimize bias in self-reported data (surveys were self-administered on a tablet and participants were guaranteed anonymity, each of which aim to reduce social desirability bias in reporting sensitive information). Additionally, any bias would be expected to influence each arm of the study equally, including the observation-only control group.

The training component of the peer change agent intervention was delivered by two or three facilitators from the social work research team. The training lasted approximately 4 hours (one half-day). Training was interactive and broken into six 45-minute modules on the mission of peer leaders (sexual health, HIV prevention, communication skills, leadership skills, and self-care). Peer leaders were asked to promote regular HIV testing and condom use through communication with their social ties at the drop-in center.

4.5 Study Results

We now present the results of the field trial, starting with an overview of the outcome variables and methodology for statistical analysis, and then giving the main results.

4.5.1 Outcome Variables

We compare two outcome variables across arms of the study. First, condomless anal sex (CAS), assessed via a survey question asking whether youth had anal sex without a condom at least once in the previous month. Second, condomless vaginal sex (CVS), assessed via a survey question asking whether youth had vaginal sex without a condom at least once in the previous month. CAS and CVS are both important behavioral risk factors for HIV transmission and so provide a direct assessment of the success of the intervention at producing a material health impact.

4.5.2 Statistical Methodology

We provide both the average value of each outcome variable at each time point for the three arms of the study as well as a statistical analysis. The statistical analysis used a mixed effects model [GH06, WWG14] (also referred to as a multi-level model). Mixed effects models are an extension of generalized linear models often used to analyze clustered or longitudinal data. We specified a linear model for each outcome variable which included terms for both the improvement caused by participating in a given arm of the study (our estimand of interest) as well as terms for a range of control variables which account for differences in demographics and the baseline rate of risk behaviors in each arm of the study. The demographic control variables were age, birth sex, transgender identity, LGBQ identity, the combination of male sex and LGBQ identity, race, committed relationship, housing status, and drop-in center. We also included a “time" variable to account for changes in the entire population over time regardless of participation in a particular arm of the study. This combination of control variables helps separate the impact of the intervention from pre-existing differences between arms of the study and intervention-independent trends. Since the outcome of interest is binary, we used a logistic link function in the model.

We also incorporate random effects in the model to account for potential correlations between data points. Specifically, we expect there to be correlations between responses from the same participant and between responses from participants in the same intervention group. Failing to correct for these dependencies would result in erroneously precise estimates, effectively overestimating the true sample size. Mixed effects models extend the standard generalized linear model to incorporate a random error term which is shared across a subset of observations, creating a correlation between the responses [GH06, WWG14]. We incorporate one such random effect for each participant and one for each intervention group, modeling the nested dependence structure of the data. Let x_i denote the covariates for individual i, $a_{\text{CHANGE},i}$ be an indicator variable for the assignment of individual i to the CHANGE arm, $\cdot a_{\text{DC},i}$ be an indicator variable for the assignment of individual i to the DC arm, $c(i)$ denote the intervention group of individual i, p_i be an indicator variable for the selection of individual i as a peer leader, and β be the model parameters. The end model is of the following form:

$$\text{logit}(y_{it}) = \beta_{\text{cov}} x_i + \beta_{\text{CHANGE}} \cdot t \cdot a_{\text{CHANGE},i} + \beta_{\text{DC}} \cdot t \cdot a_{\text{DC},i} + \beta_{\text{PL}} \cdot t \cdot p_i + \epsilon_i + \epsilon_{c(i)} + \epsilon_{it}.$$

In this model, log-odds of the outcome y_{it} are influenced by several sets of contributions. First, there is a contribution from the individual-level covariates, estimated as β_{cov}. Second, there is a contribution from assignment to a treatment arm, estimated as β_{CHANGE} and β_{DC} respectively. These terms are incorporated in the model as interactions between the treatment assignment and time, modeling improvement over time in individuals assigned to a given arm. Third, we control for improvement in individuals selected as peer leaders (regardless of which arm of the intervention they are in) via the inclusion of a time-peer leader interaction with coefficient β_{PL}. This helps us isolate improvement in the directly treated peer leaders from spillover into the group as a whole. Finally, we include the random effects ϵ_i and $\epsilon_{c(i)}$, which respectively capture correlations in multiple responses from the same individual and in responses from individuals in the same group. ϵ_{it} is an independent random term for each response. We also tested a Mundlak-Chamberlain style specification which guards against violations of the assumptions of random effects models by allowing the $\epsilon_{c(i)}$ to be correlated with the group means of the covariates [Mun78, Cha82]. This model resulted in essentially identical estimates.

We employ a Bayesian analysis with inference performed using rstan [Sta21, Sta20], in which we obtain a posterior distribution over the model parameters of interest (using the default weakly informative priors provided by rstan). This analysis allows us to better capture uncertainty in the estimates. One particular concern with frequentist approaches for our data is that calculation of standard errors and p-values is difficult when the number of clusters is small (9 in total), making asymptotic approximations unreliable. The Bayesian approach allows us to sample from the true posterior (using Markov chain Monte Carlo) without using such approximations.

We present the results in terms of 90% and 95% credible intervals over the estimated model parameters. For the treatment effects associated with CHANGE and DC, we also show the posterior over the relative risk, which averages out over the distribution of other covariates in the study population to obtain the average risk of a given outcome in participants in the treatment arm in question compared to the control group. E.g., a relative risk of 80% indicates that, on average, a participant in the given treatment arm has 80% of the probability of a given behavior compared to a participant in the control group, all other features held equal.

Results are known only for youth who completed the follow-up surveys, leading to missing data due to participant attrition (as is expected for a study enrolling YEH). Of the 713 participants who completed the baseline survey, 245 (34%) missed the 1-month follow-up, 300 (42%) missed the 3-month follow-up, and 180 (25%) missed both follow-ups. However, missingness had no statistically significant association with CAS or CVS, indicating that youth were not significantly over or under represented in the follow-up data based on their baseline level of risk behavior.

4.5.3 Results

We now present the main results of the statistical analysis, shown in Figure 4.2. As shown in the figures, for both outcomes there is little evidence of a baseline difference between the arms or of an intervention-independent improvement over time (i.e., the credible intervals for all such estimated parameters include 0).

We find that CAS reduced in the CHANGE group over time, with a posterior median relative risk of 0.58 (95% CrI: 0.35-0.93). The relative risk of 0.58 indicates that, in the model estimates, a youth who is enrolled in the CHANGE arm of the study has 42% lower probability to engage in CAS than if they were enrolled in the observation-only group. That

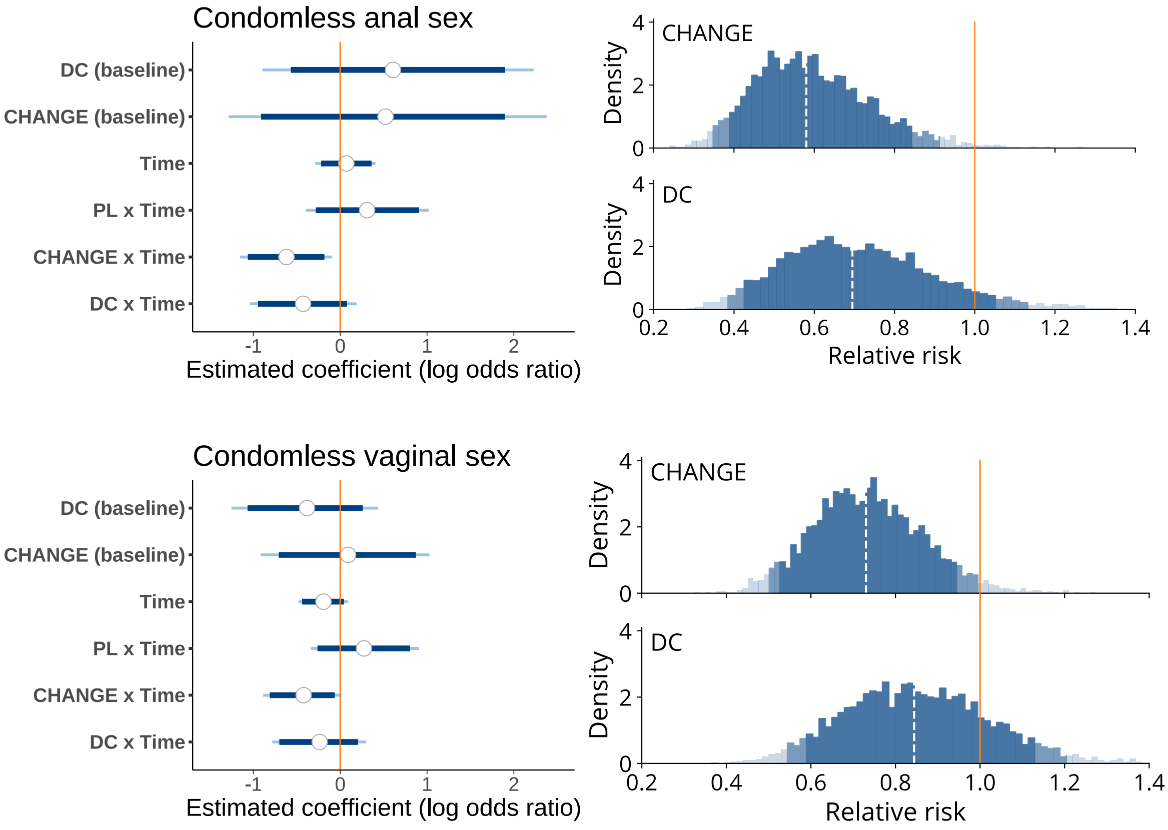


Figure 4.2: *Posterior estimates for the CAS (top) and CVS (bottom) outcomes. Left: credible intervals for estimated coefficients in the linear mixed effects model. Thick lines denote 90% credible intervals, thin lines denote 95% credible intervals, and circles denote the posterior median. Right: posterior distribution of the relative risk post-intervention for each treatment arm. Shaded regions denote the 90% and 95% credible intervals, with the dashed line giving the posterior median. In the left-hand figure, the "baseline" category measures pre-existing differences between the groups (relative to the observation-only group) on enrollment in the study. The "Group x Time" category measures the estimated per-unit-time impact of participating in each arm of the intervention (relative to the observation-only group, and after controlling for both demographics and baseline behaviors). "PL x Time" refers to improvement over time in youth selected to be peer leaders across either arm of the intervention. "Time" gives the estimated contribution of a trend over time independent of which arm of the study a participant was enrolled in.*

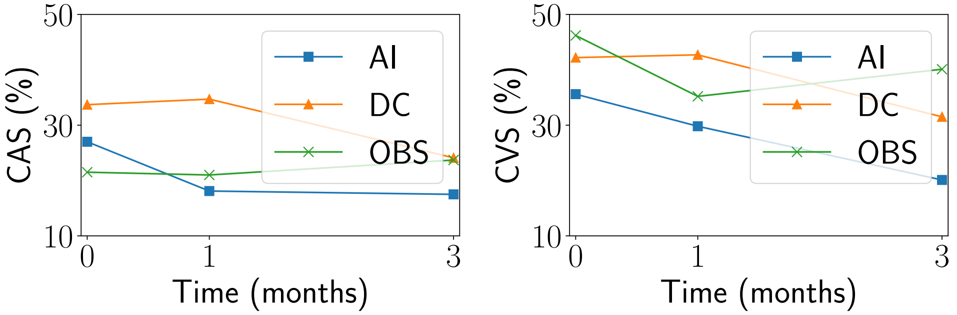


Figure 4.3: *Average value of each outcome variable at each point in time for the three arms. These plots show the results without any statistical processing, while the analysis above attempts to control for pre-existing differences between participants in each arm.*

is, on average a youth who is enrolled in CHANGE has 42% lower probability to engage in CAS post-intervention than a youth with identical starting characteristics (including baseline rate of CAS) who did not receive the intervention. For the DC group, the posterior median of 0.70 relative risk indicated a tendency towards improvement, though the 95% credible interval did not exclude 1 (95% CrI: 0.39, 1.15) and so an improvement cannot be demonstrated with high probability.

Moving to the second outcome, the model estimated that the median relative risk for CVS in the CHANGE group was 0.73, with credible interval narrowly inclusive of 1 (95% CrI: 0.50, 1.01). The 90% credible interval excluded 1, as shown in Figure 4.2. For the DC group, the median relative risk was 0.84 (95% CrI: 0.54, 1.21).

We conclude from the analysis that CHANGE produced an improvement in HIV risk behaviors with high probability (at least 97.3% probability of improvement in the posterior for both outcomes). DC showed a tendency towards improvement in these behaviors but a neutral or negative effect could not be excluded at the 90% credible interval level. Since the analysis controlled for the selection of youth as peer leaders, estimated improvements can be attributed to changes across the entire group, including youth who were not selected as peer leaders. This provides evidence that the intervention created improvements via spillovers from the directly treated peer leaders to the youth who were not selected, as hoped.

Direct examination of the average values of the outcome variables for each arm at each point in time (Figure 4.3) shows another interesting trend. Improvements in the CHANGE group happen faster than the DC group: most of the improvement for CHANGE occurs by the one-month survey, while improvements in the DC group are not fully realized until month three. Fast results are important for two reasons. First, rapid adoption of protective behaviors helps to immediately curtail transmission in a high-risk population. Second, high transience among YEH means that a non-negligible portion of youth will have left the center by the time a three-month intervention is completed. We conclude that the AI-augmented intervention implemented with CHANGE has an additional potential advantage over an intervention where peer leaders are selected with the standard DC method.

4.6 Discussion

This project provides evidence that AI methods can be used to improve the effectiveness of social network interventions in public health. Our field trial of the AI-augmented intervention resulted in estimated reductions of approximately 30-40% in the relative risk of engaging in key risk behaviors post-intervention. By comparison, an intervention planned with the status quo method of selecting high degree nodes resulted in a median estimated improvement of 16-30% in relative risk. More broadly, we hope that our experiences over the course of the project can provide generalizable lessons about how AI research can be successfully employed for social good. There have been recent attempts by others to synthesize principles for AI for Social Good research [FCKT20, TCH$^+$20]. We offer a complementary perspective shaped by the process of deploying a specific community-level intervention. In particular, existing discussions of best practice often focus in large part on ethics, data privacy, and building trust with stakeholders. While such considerations are indispensable, it is also important for the research community to investigate the on-the-ground components of developing and deploying an impactful intervention. We highlight five points.

First, the starting point was to listen to domain experts and understand where in the problem domain AI could be most impactful. We did not approach this project with a preexisting intention to apply influence maximization to the choice of peer leaders. Rather, this emerged organically from discussions between the AI and social work sides of the research team as a topic where an AI-augmented intervention was both technically feasible and likely to improve outcomes. *Success is less likely when AI researchers start with a favored technique and search for an application.*

Second, data was overwhelmingly the bottleneck to the AI component of the intervention. Computational work on influence maximization to date had largely assumed a great deal of information would be known – the structure of the graph, the model for information diffusion, etc. None of this information was in fact available for YEH (or would likely be available in other public health settings). Moreoever, gathering this data is itself time-consuming and costly, requiring unsustainable effort on the part of an agency wishing to deploy the intervention on their own. Much of the technical focus of the research consisted of finding ways to reduce the amount of data which needed to be gathered for the intervention to succeed. *Finding ways to reduce or eliminate data needs through improved algorithm design is an important part of producing deployable AI interventions in a community health context.*

Third, simplicity is valuable. As an example, prior to developing CHANGE, we designed a much more theoretically sophisticated algorithm for collecting network data which enjoyed provable guarantees for certain families of graphs [WIRT18]. However, it quickly became apparent that this algorithm would be difficult to deploy in practice because it required a large number of sequential queries (the node which is queried on step 1 determines the node who is to be queried on step 2, and so on). This was impractical in the context of a program working with YEH where any given youth may be difficult to find, interrupting the entire process. More generally, if the algorithm requires tight coupling with the outside world (many steps where information is input, the algorithm recommends a very specific action, more information is input, and so on) then there are more things that can go wrong which are not captured in the computational formalization of the problem. This poses a contrast to the way that simplicity is often operationalized in AI for social good work as either *explainability* [FCKT20] or as *methodological* simplicity [TCH$^+$20] (e.g., using well-developed techniques instead of a new algorithm). Both explainability and methodological simplicity are of course valuable in many settings but in our experience neither was first-order requirement: the algorithm can solve a complex optimization problem internally so long as the way that it interacts with the outside world is simple and robust. *We believe that this operational simplicity is an under-emphasized design criterion for AI for Social Good.*

Fourth, smaller pilot tests were a valuable part of the project prior to embarking on a larger field trial. We conducted several such tests, each of which consisted of a deployment at a single drop-in center, in order to test earlier versions of our system [WOVH^{+}18, YCXJ^{+}16, YWR^{+}17]. This helped reveal key issues which needed to be addressed. For example, we quickly discovered that a plan to collect network information via Facebook was not viable with this population and that manual collection of network data entailed a great deal of effort. We also quickly observed that peer leaders often did not attend the training, requiring on-the-fly adjustments over the course of the program. Addressing such issues was necessary to the success of the overall project (and turned out to provide much of the technical challenge involved). *It would have been very difficult to identify these challenges without piloting algorithms in the actual environment where they will be used.* It was also helpful for computer scientists on the research team to be regularly present onsite during the pilot deployments to learn more about the environment and help coordinate the initial attempts at using the algorithm.

Fifth, community engagement and trust was essential to the success of the project. Beyond the research team, a number of stakeholders needed to be involved in the process. For example, we needed buy-in from each of the drop-in centers to conduct the study at the center, enroll their clients, and use their facilities. We regularly convened a community advisory board with representatives from each of the drop-in centers along with members of the research team to provide information about the study progress, explain the methods being used, and share information which could be helpful to other center activities. Just as critical as the center leadership though, were the youth themselves. We asked youth to disclose sensitive information, including their HIV risk behaviors and social contacts.

Especially for the YEH population, which is less inclined than most to engage with authority figures, building trust is essential. We found two factors to be especially important in establishing this trust. First, the social work portion of the research team had deep roots in the community, having regularly offered services at these drop-in centers for the past ten years. Second, transparency about why information was being collected was critical. We observed substantially increased willingness to disclose information related to social contacts when researchers explained how this information would be used in the study (i.e., that a computer program would be used to select some people as peer leaders based on their contacts) than when such an explanation was not proactively given. *A critical part of the peer change agent model is empowering youth to make a difference in their community, and this philosophy extends to the way that AI should be used in a community setting.*

Our hope is that this project provides one example towards a broader research agenda aiming at AI techniques which can be successfully used to improve health and equity within our communities. A great deal of work remains. Just within the context of social network intervention, future work should explore other intervention designs (e.g., interventions which attempt to modify network structure by fostering supportive relationships), methods for further reducing data requirements (e.g., by using administrative data to infer social connections), and more deeply investigate the relationship between information diffusion and behavioral change. However, the results from this trial provide evidence that AI can substantially improve the quality of services offered to the most vulnerable among us.

Part II

Uncertainty and optimization

Chapter 5

Targeting interventions against infectious diseases under uncertainty

Treatable infectious diseases cause hundreds of thousands of cases of disability and death worldwide. Often, this burden is caused by long-term diseases which are continuously present in the population, as opposed to short-term epidemics like influenza. For instance, tuberculosis (TB) deaths in India numbered over 480,000 in 2014 [WHO15b], and even developed nations like the U.S. have observed over 395,000 cases of gonorrhea in 2015 [CDC15]. In both cases, many individuals remain undiagnosed although treatment is available. Outreach efforts to increase screening can lower disease burden; e.g., the Indian government conducts advertising campaigns for TB awareness. Limited resources require these campaigns to be carefully targeted at the most effective groups for reducing disease. Targeting is complicated by changing population dynamics, as individuals age and migrate over time, as well as by uncertainty around disease transmission rates. Officials currently make such decisions by hand as no algorithmic assistance is available.

To remedy this situation, we design an algorithm to divide a limited outreach budget between demographic groups in order to minimize long term disease prevalence under uncertain population dynamics. Our approach contrasts with existing algorithms for disease control, which often consider disease spread between nodes on a static graph

[SAPV15, BCGS10]. This is a sensible model of short term disease spread but is less suitable for long-term planning in diseases such as TB or gonorrhea, where people are born, die, age, and move [LS12]. Accounting for changes in the underlying agents is particularly salient for a policymaker who must divide resources between demographic groups over many years to maximize societal long-term health. For instance, India produces 5 year plans to combat TB [RNT16]. Our approach also contrasts with previous work on agent-based disease models [JR17, LBK$^+$10]. Such models may include realistic behaviors, but their complexity usually precludes algorithmic approaches to find the optimal policy in an entire feasible set.

An additional challenge, largely unexplored in previous algorithmic work, is that of uncertainty. Data is always limited; policymakers are never sure of exactly how many people are infected in each group, or of the contact patterns between them. In order to impact real world policy, algorithms for resource allocation must account for such uncertainties.

We introduce a model which both captures underlying agent dynamics and can be solved using an algorithmic approach in a stochastic setting. We make four main contributions. *First*, we present the MCF-SIS model (Multiagent Continuous Flow-SIS) where disease spreads in a multiagent system with birth, death, and movement. The system evolves according to SIS (susceptible-infected-susceptible) dynamics and is stratified across age groups. This introduces a new problem in multiagent systems: computing the optimal resource allocation under MFS-SIS, as in the case where an outreach campaign must decide how to divide limited advertising dollars (or rupees) between the groups.

MCF-SIS introduces a continuous, nonconvex, highly nonlinear optimization problem which cannot be solved by existing methods. Many factors must be accounted for. E.g., between-group disease transmission makes focusing on the groups with the most infected agents suboptimal. Moreover, agents in a targeted group are not cured instantaneously, so, e.g., to reduce prevalence in age group 30, we may need to start targeting resources at age 27. Lastly, we consider a stochastic setting where parts of the model (contact patterns between agents, the number of infected agents in each group, etc.) are not known exactly but are drawn from a distribution.

Our *second* contribution shows that optimal allocation in MCF-SIS is a *continuous submodular* problem. This opens up a novel set of optimization techniques which have not previously been used in disease prevention. Continuous submodularity generalizes submodular set functions to continuous domains. Intuitively, infections averted by spending one unit of treatment resources can no longer be averted by additional spending, creating diminishing returns. *Continuous submodularity is deliberately enabled by our modeling choices, in particular our shift from the discrete, graph-based setting common in previous work [SAPV15, BCGS10] to a continuous, population-based model.*

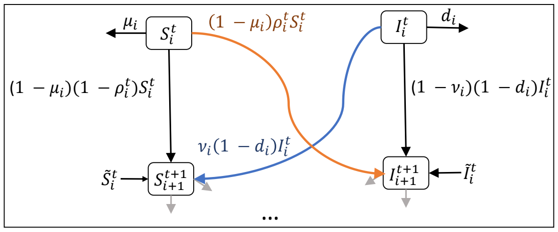
Our *third* contribution is a new algorithm called DOMO (Disease Outreach via Multiagent Optimization), which obtains an efficient $(1-1/e)$-approximation to the optimal allocation. Our algorithm builds on a recent theoretical framework for submodular optimization [BMBK17]. DOMO's generalization of this framework to the stochastic setting may be of independent interest.

Our *fourth* contribution is to instantiate MCF-SIS in two domains using empirical data which takes into account behavioral, demographic, and epidemic trends: first, TB spread in India, and second, gonorrhea in the United States. DOMO averts 13,000 annual person-years of TB and 20,000 person-years of gonorrhea compared to current policy.

5.1 MCF-SIS: a new modeling approach

The MCF-SIS model has two goals: to enable both realistic population dynamics and efficient optimization. In MCF-SIS, a finite population evolves in discrete time. Each agent has two possible states. In the susceptible (S) state, an agent has not contracted the disease. In the infected (I) state, an agent can transmit the disease to others. They can also be cured and return to the susceptible state.

The population is segmented into n groups. Our running example is where each group is an age range because transmission patterns for infection vary over age. Figure 5.1 shows this instantiation of the model. However, our techniques generalize to any way of segmenting the population (e.g., geographic location or occupation). We denote the number



$$\beta = \begin{bmatrix} 0.4 & 0.1 \\ 0.1 & 0.3 \end{bmatrix} \quad \nu = \begin{bmatrix} 0.3 \\ 0.4 \end{bmatrix} \quad d = \begin{bmatrix} 0.2 \\ 0.2 \end{bmatrix} \quad \mu = \begin{bmatrix} 0.05 \\ 1 \end{bmatrix}$$

$$I^0 = \begin{bmatrix} 10 \\ 10 \end{bmatrix} \quad S^0 = \begin{bmatrix} 10{,}000 \\ 10{,}000 \end{bmatrix} \quad N^0 = \begin{bmatrix} 10{,}010 \\ 10{,}010 \end{bmatrix} \quad M = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad \tilde{I}^0 = \begin{bmatrix} 5 \\ 2 \end{bmatrix}$$

$$\text{(1) } M^{\mathrm{T}}\left(diag(S^0)\, diag(1-\mu)\, \beta\, diag\left(\frac{1}{N^0}\right) + diag(1-\nu) diag(1-d) \right) = \begin{bmatrix} 0 & 0 \\ 0.94 & 0.09 \end{bmatrix}$$

$$\text{(2) } \mathrm{I}^1 = \begin{bmatrix} 0 & 0 \\ 0.94 & 0.09 \end{bmatrix} I^0 + \tilde{I}^0 = \begin{bmatrix} 5 \\ 12.3 \end{bmatrix}$$

Figure 5.1: *Top: Illustration of the MCF-SIS model. Bottom: a single step in the model with 2 age groups.*

of susceptible agents in each group at time t as the vector S^t where S_i^t is the number of susceptible agents of group i. Likewise, I^t gives the number of infected agents. The total population is $N^t = S^t + I^t$. At each time step, agents move between groups according to a movement matrix M, where M_{ij} is the fraction of agents in group i who move to group j. For instance, when the groups represent age, agents advance from age i to $i+1$. So, we have $M_{i,i+1} = 1$, $i = 1...n-1$, and all other entries of M are zero. Agents die of natural causes at rate μ_i. New agents enter the population through birth or migration, given by the vector $\tilde{S}^t$. We also allow for an exogenous inflow of infected agents $\tilde{I}^t$.

Disease spreads through contact with infected agents, described by the matrix β: agents in group i interact with group j with frequency β_{ij}. A fraction $\rho_i^t = \sum_{j=1}^n \beta_{ij} \frac{I_j^t}{N_j^t}$ of group i encounters an infected agent and becomes infected themselves. At each time step, a fraction d_i of infected agents in group i die. Of those who do not die, a fraction ν_i are cured and become susceptible again. ν_i is referred to as the clearance rate, and captures the total rate at which infected agents are diagnosed, enter treatment, and are successfully cured.

While compartmental models like ours do not simulate the micro-level details of individual agents, they can be realistic enough to capture long term trends. Similar models are commonly used in health policy analyses [WWC+05, CMF11, DGCS12].

Interventions: We consider optimal resource allocation modeled through the clearance vector ν. Suppose a policymaker can conduct outreach to selected groups. Since some percentage of people who see an advertisement will enter treatment, we can model the policymaker's decisions as increasing ν_i for the targeted groups. We suppose the algorithm has a budget K for new advertising to split among the groups. ν starts at a lower bound L, reflecting pre-campaign treatment rates. The algorithm may select any post-campaign ν with $||\nu - L||_1 \leq K$ and $L_i \leq \nu_i \leq U_i \; \forall i$, where $U < 1$ is an upper bound. Note that U is strictly less than 1 because we can never realistically treat 100% of any given group. Denote the set of ν satisfying these constraints as the feasible polytope $\mathcal{P}$. While we focus on the above $\mathcal{P}$ for concreteness, our approach works for any downward-closed polytope. The goal is to select a $\nu \in \mathcal{P}$ which minimizes the total infected agents over a time horizon T.

Optimization formulation: We assume that the number of infected agents (and hence deaths) is small compared to the total population. For instance, less than 1% of the total population is infected with TB in India or gonorrhea in the U.S. [WHO15b, CDC15]. Therefore, we consider the total population size (the vector N^t) as fixed independently of ν. Thus, the state of the system is captured just by the infected vector I^t. A single group i evolves as

$$I_i^{t+1} = \sum_{j=1}^{n} M_{ji} \Big(S_j^t (1 - \mu_j) \sum_{k=1}^{n} \beta_{jk} \frac{I_k^t}{N_k^t} \\ + (1 - \nu_j)(1 - d_j) I_j^t \Big) + \tilde{I}_i^t.$$

The expression in parentheses is the number of infected agents in group j. The first term is the number who are newly infected and the second is the number from previous steps who are not cured and do not die. The outer summation accounts for the number of these infected agents who transition from group j to group i. Lastly, we add the new arrivals $\tilde{I}_i^t$.

We can iterate the equation for each group forward from $t = 1...T$ in order to obtain the total number of infected agents at time T. Instead though, will work with an equivalent matrix formulation of the system for ease of notation. For convenience, we will use the augmented state vector $x^t = [I^t \;\; 1]$. That is, x^t is the number of infected people appended with a single one. The one is just for mathematical convenience. We formulate a time-varying linear operator $B^t(\nu)$ such that $x^{t+1} = B^t(\nu)x^t$ via the block form

$$B^t(\nu) = \begin{bmatrix} A^t(\nu) & \tilde{I}^t \\ \vec{0} & 1 \end{bmatrix}$$

where the block $A^t(\nu)$ is defined as

$$A^t(\nu) = M^\top \Big(diag(S^t) diag(1-\mu)\ \beta\ \ diag\Big(\frac{1}{N^t}\Big) \\ + diag(1-\nu) diag(1-d) \Big).$$

where $diag(v)$ is the matrix with the entries of v on the diagonal. $A^t(\nu)I^t$ gives the number of infected agents given only the internal dynamics of the population, resulting in a total of $A^t(\nu)I^t + \tilde{I}^t$ infected agents. Figure 5.1 shows an example of a simple case of the model with two age groups. Example parameter values are given, along with the initial prevalence I^0. The first equation computes the matrix $A^0(\nu)$. The second applies $A^0(\nu)$ to the initial prevalence I^0 and then adds the exogenous inflow $\tilde{I}^0$. The number of infected agents in group 1 increases because more agents are infected than cured. These agents then transition to group 2. Because there are only two groups in our modeled population for this example, all agents in group 2 exit the modeled population.

We aim to minimize the total infected agents over T steps:

$$\begin{aligned} &\min \sum_{t=1}^{T} c^\top \left[\prod_{j=t}^{1} B^j(\nu) \right] x^0 \\ &1^\top \nu \leq 1^\top L + K \qquad (5.1) \\ &L_i \leq \nu_i \leq U_i \quad \forall i = 1...n \end{aligned}$$

c can be any nonnegative cost vector, e.g., $c = [\vec{1}\ \ 0]$ (n ones and a zero) sums over the number of infected agents in each group. x^0 is the initial state (number of infected agents).

We use the notation $\prod_{j=t}^{1} B^t(\nu)$ as shorthand for $B^t(\nu)B^{t-1}(\nu)...B^1(\nu)$.

5.2 Algorithmic approach

We now turn to computing a (near) optimal solution to Problem 5.1. This is a continuous optimization problem since each ν_i may take any value in $[L_i, U_i]$. Unfortunately, the objective function is nonconvex, which rules out standard methods for efficiently obtaining good solutions. It is also highly nonlinear since the decision variables ν are raised to the power T, which may be large (e.g., a time horizon of 10 or 20 years). This suggests that many local optima could be present and renders optimization more complicated.

However, MCF-SIS's definition contains useful structure. Intuitively, resources have diminishing returns: infections averted by increasing one ν_i can no longer be treated by increasing some other ν_j. Diminishing returns suggests submodularity. However, since our optimization problem is not discrete, standard submodularity and the greedy algorithm do not apply. Instead, we show that our objective is *continuous submodular*, a generalization of submodularity to continuous domains [Bac15, BMBK17]. Continuous submodularity enables efficient optimization and allows us to handle the stochastic case in a natural manner. This framework is crucially enabled by the modeling choice to shift from the discrete, graph-based setting common in previous work [SAPV15, BCGS10, CHT09] to a continuous, population-based model. Not only does our model account for population dynamics, but it is also more amenable to optimization.

We now define continuous submodularity[1]. Let $\wedge$ and $\vee$ denote coordinatewise minimum and maximum respectively. A function $F : R^n \to R$ is continuous submodular if $F(x) + F(y) \geq F(x \vee y) + F(x \wedge y)$ for all feasible x, y. This is reminiscent of submodular set functions, but extended to the continuous domain. F is called continuous supermodular if the inequality is reversed. If F is continuous submodular, $-F$ is continuous supermodular. Note that continuous submodularity is not convexity or concavity; it is a distinct class of

[1] Technically, we use the stronger condition of *DR-submodularity*. Details related to showing our objective is DR-submodular can be found in the supplement.

functions with distinct optimization techniques.

We will draw on these techniques to Problem 5.1. Bian et al. [BMBK17] define a theoretical framework for optimizing continuous submodular functions. In order to make use of this framework, we need to first show that our problem falls into it. Then, we need to fill in the algorithmic components required to instantiate the approach that the framework suggests (two oracles explained below). Lastly, we need to prove that our objective is sufficiently smooth for the resulting algorithm to converge in a reasonable number of iterations. None of these pieces are covered by previous work; they are algorithmic contributions specific to our domain.

We start out by showing that the continuous submodularity framework applies. Denote the objective of Problem 5.1 as $F(\nu)$. We will show that F is continuous *supermodular* which in turn implies that $-F$ is continuous *submodular*. Since minimizing F is equivalent to maximizing $-F$, this will allow us to design an efficient algorithm based on continuous submodularity. Our proof that F is supermodular has two steps. First, we show that F is a *posynomial* in the variables $1 - \nu_i$. A posynomial is a polynomial with entirely nonnegative coefficients[2]. Then, we will show that any function which is a posynomial in $1 - \nu$ is continuous supermodular in ν. We start by showing the following:

Lemma 6. *F is a posynomial in the variables $1 - \nu_i$.*

Proof. First, note that F depends on ν only through the term $diag(1 - \nu)$. Note also that every term in the expression for the block $A^t(\nu)$ is nonnegative. Since matrix multiplication is just a series of multiplications and additions, it follows that $c^\top \prod_{j=t}^{1} \left[B(\nu)^j\right] x_0$ (and hence the sum over time $t = 1...T$) is a polynomial in $1 - \nu$ and all of the coefficients of this polynomial are nonnegative. This can also be seen through the expression for the evolution of a single group, which contains only terms of the form $(1 - \nu_i)$ multiplied by nonnegative coefficients. □

Note that this step hinged on MCF-SIS's continuous, population-based nature. Since F is

[2] Under some circumstances, posynomials can be optimized via geometric programming. Unfortunately, this does not work for our problem since the feasible set is not convex in $1 - \nu$.

a posynomial in $1 - \nu_i$, it can be written in the form $F(\nu) = \sum_{j=1}^{\ell} a_j \prod_{i=1}^{n} (1 - \nu_i)^{p_{ij}}$ where a_j is a nonnegative coefficient for term j and p_{ij} is a nonnegative integer. This representation does not have to be computed; its existence is just useful for the proofs.

We now turn to showing that any function that is a posynomial in $1 - \nu$ is continuous supermodular in ν. Our result builds on the following lemma:

Lemma 7 (Staib and Jegelka (2017)). *Let $f_1 ... f_n : R \to R_+$ be nonnegative, differentiable functions which are either all nonincreasing or all nondecreasing. Then, $F(x) = \prod_{i=1}^{n} f_i(x)$ is continuous supermodular.*

Using this, we show the following:

Lemma 8. *Whenever F is a posynomial in $1 - \nu$, it is also continuous supermodular in ν.*

Proof. First, note that continuous supermodularity is preserved under nonnegative linear combinations. Hence, we focus on an individual term $\prod_{i=1}^{n} (1 - \nu_i)^{p_{ij}}$ in the posynomial representation of F. For each $i = 1 ... n$, define $f_i(\nu) = (1 - \nu_i)^{p_{ij}}$. Note that each f_i is nonincreasing in ν_i since $0 \leq \nu_i < 1$. Further, $f_i(\nu_i) \geq 0$ always holds. The conclusion now follows from Lemma 7. □

To sum up: we want to minimize F, which via Lemma 6 is a posynomial in $1 - \nu$. Via Lemma 8, this implies that F is continuous supermodular in ν. Hence, maximizing $-F$ is a continuous submodular optimization problem. We will actually maximize the objective $G(\nu) = -F(\nu) + M$, where M is any constant large enough to ensure that G is nonnegative. Clearly, this is also equivalent to minimizing F.

5.2.1 The DOMO algorithm

We now introduce the DOMO algorithm (Disease Outreach via Multiagent Optimization, Algorithm 9) to exploit continuous submodularity. We start out with the deterministic setting where model parameters are fully known. Here, DOMO builds on the Frank-Wolfe approach [BMBK17] (though new techniques are needed in the stochastic setting). DOMO generates a series of feasible solutions $\nu^0 ... \nu^R$, where R is the total number of iterations. More iterations imply greater accuracy (Theorem 1 bounds the number needed). The algorithm starts at $\nu^0 = L$, the lower bound. Each iteration alternates between two steps (lines 4-5). First, it computes the gradient of the objective at the current point. Second, it takes a step in the direction of the point which optimizes the gradient over the feasible set $\mathcal{P}$. Essentially, at each iteration the algorithm spends a fraction $\frac{1}{R}$ of the budget according to the current gradient. Higher R allows finer control over the solution.

It is known that this strategy gives a $(1 - 1/e)$-approximation for continuous submodular functions [BMBK17]. *However, it is not an out-of-the-box approach (even in the deterministic setting).* DOMO requires two oracles (specific to our problem) to instantiate the algorithm – a gradient oracle which supplies the gradient of G at any given point, and a linear optimization oracle which maximizes a given linear function over the feasible set $\mathcal{P}$. Additionally, the number of iterations (and hence runtime) required is potentially unbounded. We prove (in Theorem 1) that our objective is sufficiently smooth for the algorithm to converge efficiently. We first supply appropriate oracles.

Gradient oracle: Instead of tediously computing the posynomial representation, we directly compute the gradient using the block matrix representation of MCF-SIS. Denote $Y^t(\nu) = \prod_{j=1}^t B(\nu)^j$. We can concisely express the gradient via matrix calculus [PP08]:

$$\frac{\partial G(\nu)}{\nu_i} = -\sum_{t=1}^{T} \mathrm{Tr}\left[\left(\frac{\partial c^\top Y^t(\nu) x_0}{Y^t(\nu)}\right)^\top \frac{\partial Y^t(\nu)}{\nu_i}\right] =$$
$$-\sum_{t=1}^{T} \mathrm{Tr}\left[c x_0^\top \sum_{\ell=1}^{t} \left[\prod_{i=t}^{\ell+1} B^i(\nu)\right] \frac{\partial B^\ell(\nu)}{\partial \nu_i} \left[\prod_{i=\ell-1}^{1} B^i(\nu)\right]\right]$$

where the first step is the chain rule and the second follows from the product rule and induction on n. Tr denotes the matrix trace. This reduces gradient evaluation to computing $\frac{\partial}{\partial \nu_i} B^j(\nu)$ for each i and j. B^j depends on ν_i only through the block $A^j(\nu)$, so we have $\frac{\partial}{\partial \nu_i} B^j(\nu) = (1 - d_i) M^\top J_{i,i}$, where $J_{i,i}$ is the matrix with a one in entry (i,i) and zeros elsewhere. By appropriately ordering multiplications, the entire procedure uses T matrix multiplications.

Linear optimization oracle: Since $\mathcal{P}$ is a polytope, linear optimization could be performed by solving a linear program. However, exploiting the special structure of $\mathcal{P}$ lets us perform linear optimization in time $O(n \log n)$ via a simple greedy algorithm (function GreedyLinear in Algorithm 9). This algorithm simply orders the group $i = 1...n$ according to $\nabla_i G(\nu)$ (Line 10). It then proceeds through the groups in this order, spending as much of the budget as possible (Line 15) before moving on to the next.

Algorithm 9 DOMO

```
 1: function DOMO($R, K$)
 2:     $\nu^0 \leftarrow L$
 3:     for $k = 1...R$ do
 4:         $\nabla^k \leftarrow$ GradientOracle($\nu^{k-1}$)
 5:         $y^k \leftarrow$ LinearOracle($\nabla^k, K$)
 6:         $\nu^k \leftarrow \nu^{k-1} + \frac{1}{R} y^k$
 7:     end for
 8:     return $\nu^R$
 9: end function
10: function GreedyLinear($\nabla$, $K$)
11:     $y \leftarrow L$
12:     $\pi \leftarrow$ ordering of $1...n$ such that $\nabla_{\pi(i)} \geq \nabla_{\pi(i+1)}$ $\forall i$
13:     $i \leftarrow 0$
14:     while $||y - L||_1 < K$ do
15:         $y_{\pi(i)}$ += $\min(U_{\pi(i)} - L_{\pi(i)}, K - ||y - L||_1)$
16:         $i$ += 1
17:     end while
18:     return $y$
19: end function
```

Lemma 9. GreedyLinear *finds an optimal solution to the linear optimization problem over* $\mathcal{P}$.

Proof. We recognize the linear optimization problem as a fractional knapsack problem where each item takes up the same amount of space. The value of each item is the corresponding entry of the gradient. Hence, greedily taking as much as possible of the highest-value items is optimal. □

Convergence analysis: The runtime of Algorithm 9 depends on the number of iterations R, which Bian et al. [BMBK17] show must be proportional to the Lipschitz constant of the gradient of G. Essentially, functions with Lipschitz continuous gradients are smooth in a sense that allows the algorithm to quickly converge. Let $U_{max} = \max_i U_i$. We bound the Lipschitz constant for our model and show that

Theorem 10. *For any $\epsilon > 0$, using $R = \frac{K}{\epsilon}\left(\frac{T}{1-U_{max}}\right)^2$ iterations, the ν output by Algorithm 9 satisfies $G(\nu) \geq \left(1 - \frac{1}{e} - \epsilon\right) G(\nu^*)$, where ν^* is an optimal solution.*

The proof is given in the supplement due to space constraints. We remark that U_{max} is always bounded away from 1 since we can never realistically treat 100% of any single group. Thus, the number of iterations is $O\left(\frac{KT^2}{\epsilon}\right)$. Each iteration requires one linear optimization over $\mathcal{P}$ (which takes time $O(n \log n)$ using GreedyLinear) and one gradient evaluation (which takes time $O(Tn^\omega)$, where ω is the matrix multiplication constant). The final runtime is $O\left(\frac{KT^3 n^\omega}{\epsilon}\right)$.

5.3 Stochastic optimization

In reality, some parameters of the multiagent system will not be known exactly. For instance, the contact matrix β is almost never precisely known in practice. Additionally, for many diseases, there is considerable uncertainty about the initial prevalence I^0 (Suen et al. 2015). We now extend DOMO to the stochastic case where model parameters are drawn from a distribution instead of known exactly. Hence, we can infer an appropriate prior distribution from whatever data is available and optimize the expected value over this distribution. Our formulation is very general, and will allow any of the parameters (M, β, I^0, $\tilde{I}$, etc.) to be unknowns. Suppose that we have an uncertainty set Ξ for the joint values of the unknowns and Ξ is equipped with a distribution D. Let $G(\cdot, \xi), \xi \in \Xi$ denote the objective for any fixed set of parameters. We wish to solve the stochastic problem

$$\max_{\nu \in \mathcal{P}} \mathop{\mathbb{E}}_{\xi \sim D} \left[G(\nu, \xi)\right] \tag{5.2}$$

Such stochastic problems are typically difficult computationally. For instance, Zhang et al. [ZP14b] study vaccination on a graph when the initially infected nodes (I^0 in our model) are uncertain. To design a scalable algorithm, they must assume that D is an independent distribution. However, I^0 for different groups will clearly be correlated because of the underlying multiagent dynamics. A common approach to accounting for uncertainty without such strong assumptions is robust optimization, which solves the worst case problem $\max_{\nu \in \mathcal{P}} \min_{\xi \in \Xi} G(\nu, \xi)$. Han et al. [HPNP15] take this approach for a vaccination problem when the graph β is unknown. However, robust optimization introduces a computationally challenging bilevel optimization problem which requires specialized techniques. This makes it difficult to incorporate uncertainty in multiple parts of the model.

We resolve these difficulties through an alternate approach which efficiently handles uncertainty over *any* of the model parameters, expressed through an *arbitrary* distribution D. Moreover, we obtain provable guarantees just as in the deterministic case. We start out by noting that the objective of Problem 5.2 is continuous submodular since it is a nonnegative linear combination of continuous submodular functions. Also note that Algorithm 9 accesses the objective only through GradientOracle. While we can no longer access the gradient in closed form, the key idea is to instead use a stochastic approximation. At each iteration, we draw r samples $\xi_1 ... \xi_r$ i.i.d. from D. Our estimate of the gradient is $\widehat{\nabla} = \frac{1}{r} \sum_{i=1}^{r} \nabla G(\nu, \xi_i)$. We then modify Line 5 of Algorithm 9 to call LinearOracle$(\widehat{\nabla})$.

To our knowledge, no previous work has analyzed stochastic continuous submodular optimization. We give a new analysis which draws on tools for analyzing stochastic concave problems [HL16]. We extend these techniques to (nonconcave) continuous submodular functions and prove the following guarantee:

Theorem 11. *Using $r = \left(\frac{4KT}{1-U_{\max}}\right)^2$ samples, DOMO outputs a ν satisfying $\mathbb{E}\left[G(\nu, \xi)\right] \geq \left(1 - \frac{1}{e} - \epsilon\right) \mathbb{E}\left[G(\nu^*, \xi)\right]$ where ν^* is an optimal solution to Problem 5.2. The number of iterations is the same as Theorem 1.*

Note that the guarantee for $\mathbb{E}[G(\nu, \xi)]$ exactly matches that for $G(\nu)$ in the deterministic case. Further, our analysis generalizes to *any* smooth continuous submodular function and may be of interest in other domains.

Table 5.1: *Infected people per 100,000 according to the National Family Health Survey. Reported in (Suen et al. 2015).*

Year/Age	30	35	40	...
1993	555.499	680.426	1059.359	...
1998	781.136	940.218	827.718	...
2003	329.045	453.052	522.364	...
2005	539.154	625.982	722.140	...

Table 5.2: *Example parameters. E.g., $d = 0.544$ indicates that 54.4% of people with active TB die each year. Ranges indicate variance over age and/or year.*

Parameter	Value	Source
Starting pop.	355,692,752	[UN15]
Total infected	2,949,057	(Suen et al. 2015)
Status quo ν	0.07 - 0.13	[RNT16]
μ	0.003 - 0.02	[WHO15a]
d	0.544	(Tiemersma 2011)

5.4 Experiments

We now present experimental results on two real-world problem instances: TB prevention in India and gonorrhea prevention in the United States. In both, we produce a highly realistic evaluation by instantiating MCF-SIS using demographic and epidemiological data drawn from a variety of governmental and NGO sources. Since prevalence numbers are highly uncertain, and the contact matrix β not explicitly known, we estimate a distribution over both using this data and apply DOMO as described above. MCF-SIS is stratified by age. We account for migration and births by comparing the number of individuals in each age group at each year to the next, after accounting for non-disease deaths.

Tuberculosis: True TB prevalence in India is subject to great uncertainty, as many patients do not report to approved treatment facilities [RNT10]. We estimate prevalence (the initial infected vector I^0 and new arrivals $\tilde{I}$) using age-stratified data provided by the Indian government for the years 1993-2005 [IIP14], see Table 5.1. These figures are reported with 95% confidence intervals; we sample values of $(I^0, \tilde{I})$ within these assuming a Gaussian distribution. Table G.1 shows example parameter values. For each sample, we find the $\hat{\beta}$ that minimizes the mean squared error between the observed I and that predicted by MCF-SIS.

Figure 5.3 shows an example $\hat{\beta}$; darker cells represent more interaction. The matrix is sparse, with most entries along the diagonal (representing within-group interaction) and a few groups who interact more with others. Population statistics and disease parameters (e.g., d) are taken from World Health Organization lifetables, the Indian government Revised National Tuberculosis Control Program reports, and United Nations Statistics Division demographic reports (see supplement). Our model includes 30 age groups representing ages 30-60.

Gonorrhea: We infer the initial prevalence I^0 and new arrivals $\tilde{I}$ from reported disease cases. However, up to 80% of cases are asymptomatic and may remain undetected [CDC15]. We assume a uniform distribution for the true prevalence at every age (I) with an upper limit of 4 times the reported values and a lower bound equal to the value reported by the U.S. Centers for Disease Control. We generate a set of sampled $(I^0, \tilde{I})$ from this uniform distribution. For each sample, we infer the β matrix which best matches the age-stratified prevalence rate in the same manner as for TB. Data on population demographics is taken from the WHO and the U.S. Census (details in the supplement). Our model includes 46 age groups, representing ages 15-60.

Baselines: No previous work directly addresses our setting, so we define several baselines. First, *degree*, which greedily spends the budget on the groups with highest weighted degree with respect the contact matrix β. This captures the intuition that groups which are in contact with many others are important targets for treatment Second, *eigen*, which greedily spends the budget on groups according to their eigenvector centrality in β. Degree and eigen test whether it is necessary to consider population dynamics, or if just the contact matrix is sufficient. Third, *myopic*, which selects the ν that will result in the largest reduction in infections after a single timestep in the MCF-SIS. This can solved exactly via linear programming. Myopic tests if DOMO's long-term reasoning is needed. Fourth, *prevalence*, which allocates resources greedily to the groups with the largest number of infected individuals. This is common practice in epidemiology. Fifth, *equal*, which splits the budget equally over all groups. Sixth, *SQ* which allocates the budget proportional to the

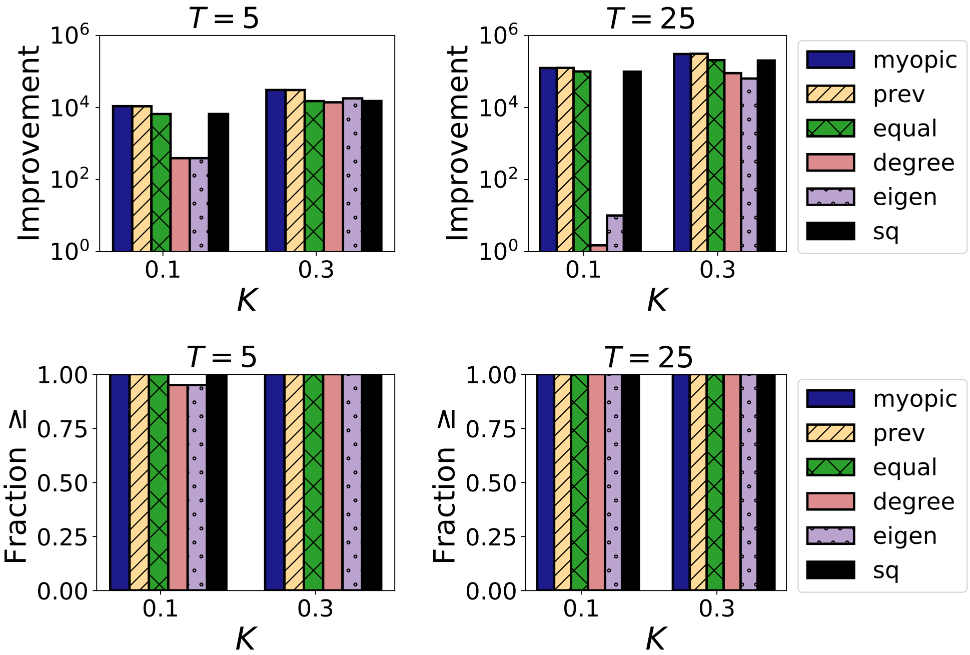


Figure 5.2: *Top: Improvement in mean case-years averted by DOMO over each other algorithm. Bottom: Fraction of 100 sampled instances in which DOMO's averted case-years is at least as high as each baseline.*

status-quo ν produced by current government policies. This models spending the budget according to the same strategy as is currently used.

Results: For each domain, we obtain the status quo treatment rate ν_{SQ} from existing data (which is the lower bound L). Then, we assume that a policymaker may distribute an additional budget K via an outreach campaign. We set $U = 1.05 \cdot \nu_{SQ}$. We do not plot runtime because all algorithms, implemented in Python, run in under 10 minutes on all datasets and parameter combinations. DOMO is run with $R = 100$ iterations and $r = 100$ samples.

We start with TB. The top row of Figure 5.2 shows the improvement in objective value of DOMO over each other algorithm. Improvement is in terms of disease burden: the total person-years of disease summed over time 1...T (the objective function). Each plot shows two values of K on the x axis corresponding to small and large budgets. The y axis plots the difference between the disease burden under each baseline versus DOMO (note the log scale). Disease burden is calculated by averaging over 100 samples for the unknown

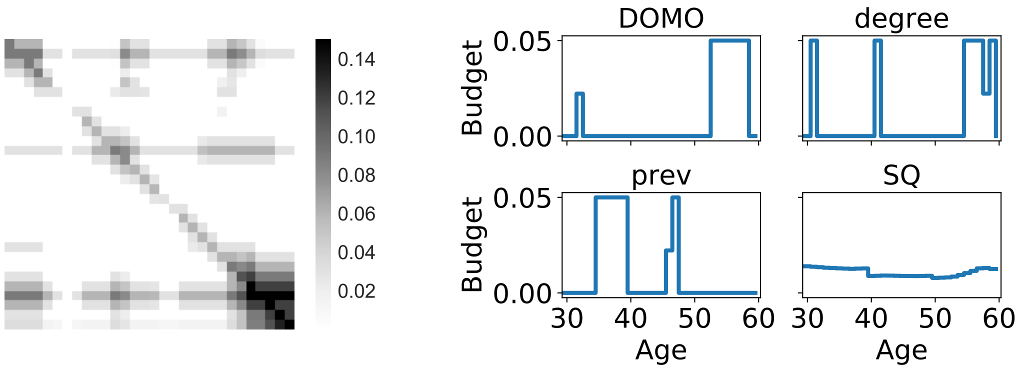


Figure 5.3: *Left: a sampled β matrix. Right: illustrated allocation*

parameters. One plot shows the short horizon $T = 5$, and one shows the long horizon $T = 25$.

DOMO outperforms all baselines (has positive improvement) under each configuration. The difference is larger for $K = 0.3$ than $K = 0.1$, indicating that DOMO makes more strategic use of the additional resources. Most differences also grow as T increases; the longer time horizon presents a more challenging planning problem. The two closest competitors are degree and eigen, which obtain relatively close values for $K = 0.1$. However, their gap with DOMO increases substantially for $K = 0.3$. The performance gap is very significant in policy terms: for $K = 0.3, T = 25$, DOMO averts (approximately) between 64,000 to 300,000 person-years of disease more than each baseline. All differences are statistically significant (t-test, $p < 0.001$). Given the annual death rate $d = 0.544$, DOMO averts over 6,500 TB deaths per year compared to the status quo governmental policy.

Further, DOMO performs optimally out of all considered algorithms in at least 90% of specific realizations of the parameters. In Figure 5.2, the y axis shows the fraction out of 100 randomly sampled parameter combinations in which DOMO performed at least as well as every other algorithm. We again plot results for $T = 5, 25$, and $K = 0.1, 0.3$. The values are all fairly high, ranging from 0.9 to 1. We conclude that our stochastic optimization approach successfully captures uncertainty in this domain because it has high performance almost all circumstances, not just in expectation.

Figure 5.3 contrasts the allocation made by DOMO and other policies. We focus on $K = 0.3, T = 10$. Each line shows the amount of budget the corresponding algorithm allocates to each group (shown on the x axis). To avoid crowding the plot, we show DOMO, degree, prevalence, and SQ. Myopic's allocation was very close to prevalence while eigen's was similar to degree. We see that SQ allocates the budget relatively uniformly, while DOMO concentrates heavily on particular groups. Moreover, DOMO does not simply allocate to high-prevalence groups. This indicates that DOMO exploits long term dynamics beyond which agents are immediately infected. DOMO also does not simply allocate to high degree groups. Its allocation overlaps with the high degree elderly groups, but places little budget on the high degree groups near ages 30 and 40. We conclude that DOMO leverages non-obvious patterns in the multiagent system's dynamics to outperform simpler heuristics.

We find that DOMO also performs better than the baselines in our gonorrhea example (Figure 5.4). Generally, results are similar to TB, so due to space limitations, we show results for $T = 25$. The left hand figure shows the improvement in disease burden that DOMO makes over each baseline; DOMO substantially outperforms all of the baselines for both values of K. For $K = 0.3, T = 25$, DOMO results in at least 500,000 fewer person-years of disease than any other algorithm. The right hand figure plots the fraction of sampled instances in which DOMO performs at least as well as each algorithm. DOMO outperforms equal, degree, eigen, and SQ in 100% of instances. It outperforms myopic in approximately 75% of instances, and prevalence in 60-70%. DOMO's expected performance is much higher than prevalence because in those sampled instances where DOMO outperforms prevalence, it does so by a large margin. When DOMO outperforms prevalence, it does so by 3.9 million person-years on average. Conversely, when prevalence outperforms DOMO, it does so by approximately 200,000 person-years on average.

5.5 Conclusion and additional related work

We develop an algorithmic approach to targeting disease outreach campaigns which synthesizes agent-based modeling and algorithmic disease control. A large body of work in health

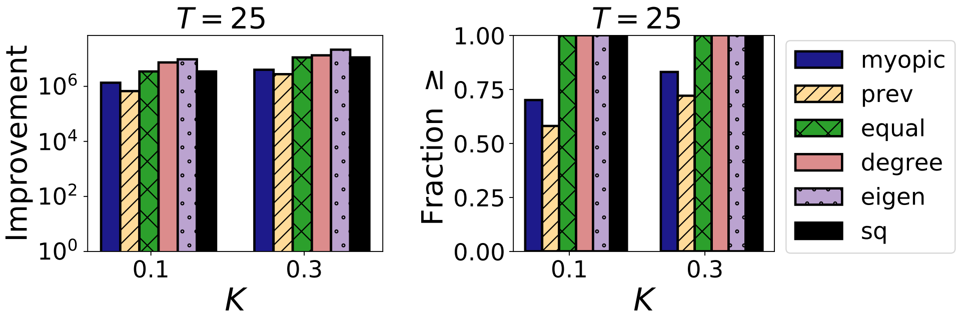


Figure 5.4: *Results for gonorrhea instance. Left: improvement in case-years averted by DOMO over each other algorithm. Right: fraction of instances in which DOMO performs at least as well as each baseline.*

policy and agent based modeling develops realistic disease models [JR17, PD09, SEM14, BS14, PHSE17]. None use an algorithmic approach for disease control, instead examining a limited set of policies that can be exhaustively searched. In contrast, we consider the challenge of algorithmically optimizing over the entire feasible set.

Much algorithmic work focuses on immunizing the nodes of a graph to limit disease spread [CTP+16, SAPV15, SHL15, BCGS10, DOT14]. None of this work considers the challenges of population dynamics. While others may examine subgroup dynamics [ZAS+16] or time trends [PTV+10], our work presents a novel approach to optimizing resource allocation for infectious diseases in a stochastic setting.

Chapter 6

Risk-averse submodular optimization

This chapter explores risk-averse optimization, a related algorithmic problem to robust optimization. Although not used in the HIV prevention application, it provides another example of how techniques spanning the boundary between continuous and discrete optimization can be used to tackle a wide range of problems.

Decision-making under uncertainty is an ubiquitous problem. Suppose we want to maximize a function $F(\boldsymbol{x}, y)$, where $\boldsymbol{x}$ is a vector of decision variables and y a random variable drawn from a distribution D. A natural approach is to maximize $\mathbb{E}_y \left[F(\boldsymbol{x}, y) \right]$, i.e., to maximize the expected value of the chosen decision. However, decision makers are often *risk-averse*: they would rather minimize the chance of having a very low reward than focus purely on the average. This is a rational behavior when failure can have large consequences. For instance, if a corporation suffers a disastrous loss, they may simply go out of business. Or in many cases, low performance entails safety issues. For instance, if a sensor network for water contamination detects problems instantly in 80% cases, but fails entirely in 20%, the population will inevitably be exposed to an unacceptable health risk. It is much better to have a sensor network which always detects contaminants, even if it requires somewhat more time on average.

Hence, it is natural to move beyond average-case analysis and optimize a risk-aware objective function. One widespread choice is the *conditional value at risk* (CVaR). CVaR takes a tunable parameter α. Roughly, it measures the performance of a decision in the worst α fraction of scenarios. It is known that when the objective F is a concave function, then CVaR can be optimized via a concave program as well. However, many natural objective functions are *not* concave, and no general algorithms are known for nonconcave functions. We focus on *submodular* functions. Submodularity captures diminishing returns and appears in application domains ranging from viral marketing [KKT03], to machine learning [KT12], to auction theory [Von08]. We analyze submodular functions in two settings:

Continuous: Continuous submodularity, which has lately received increasing attention [Bac15, BMBK17, SJ17] generalizes the notion of a submodular set function to continuous domains. Many well-known discrete problems (e.g., sensor placement, influence maximization, or facility location) admit natural extensions where resources are divided in a continuous manner. Continuous submodular functions have also been extensively studied in economics as a model of diminishing returns or strategic substitutes [KOS00, Sam16]. Our main result is a $(1 - \frac{1}{e})$-approximation algorithm for maximizing the CVaR of any monotone, continuous submodular function. No algorithm was previously known for this problem.

Portfolio of discrete sets: Our results for continuous submodular functions also transfer to set functions. We study a setting where the algorithm can select a distribution over feasible sets, which is of interest when the aim is to select a portfolio of sets to hedge against risk [OY17]. Similar settings have also been studied in robust submodular optimization [KRG11, CLSS17, Wil18a]. We give a black-box reduction from the discrete portfolio problem to CVaR optimization of continuous submodular functions, allowing us to apply our algorithm for the continuous problem. The state of the art for the discrete portfolio setting is an algorithm by Ohsaka and Yoshida [OY17] for CVaR influence maximization. Our results are stronger in two ways: (i) they apply to *any* submodular function and (ii) give stronger approximation guarantee. Allowing the algorithm to select a convex combination of sets is provably necessary: Maehara [Mae15] proved that restricted to single sets, it is NP-hard to compute any multiplicative approximation to the CVaR of a submodular set function.

We experimentally evaluate our algorithm for sensor resource allocation, focusing on two domains: detecting contagion or rumors in social networks, and detecting contamination in water networks. In both cases, our algorithm substantially outperforms baselines.

6.1 Problem description

In this section, we formally define continuous submodularity and the conditional value at risk. We first study the continuous setting and then extend our results to discrete portfolio optimization.

Continuous submodularity: Let $\mathcal{X} = \prod_{i=1}^{n} \mathcal{X}_i$ be a subset of R^n, where each $\mathcal{X}_i$ is a compact subset of R. A twice-differentiable function $F : \mathcal{X} \rightarrow R$ is *diminishing returns submodular* (DR-submodular) if for all $\boldsymbol{x} \in \mathcal{X}$ and all $i, j = 1...n$, $\frac{\partial^2 F(\boldsymbol{x})}{\partial x_i \partial x_j} \leq 0$ [BMBK17]. Intuitively, the gradient of F only shrinks as $\boldsymbol{x}$ grows, just as the marginal gains of a submodular set function only decrease as items are added. Continuous submodular functions need not be convex or concave (concavity requires that the Hessian is negative semi-definite, not that the individual entries are nonpositive). We consider *monotone* functions, where $F(\boldsymbol{x}) \leq F(\boldsymbol{y})$ $\forall \boldsymbol{x} \preceq \boldsymbol{y}$ ($\preceq$ denotes element-wise inequality). We assume that F lies in $[0, M]$ for some constant M. Without loss of generality, we assume $F(0) = 0$ (normalization).

In our setting F is a function of both the decision variables $\boldsymbol{x}$ and a random parameter y. Specifically, we consider functions $F(\boldsymbol{x}, y)$ where $F(\cdot, y)$ is continuous submodular in $\boldsymbol{x}$ for each fixed y. We allow any DR-submodular F which satisfies some standard smoothness conditions. First, we assume that F is L_1-Lipschitz for some constant L_1 (for concreteness, with respect to the ℓ_2 norm[1]). Second, we assume that F is twice differentiable with L_2-Lipschitz gradient. Third, we assume that F has bounded gradients, $||\nabla F||_2 \leq G$. Only the last condition is strictly necessary; our approach can be extended to any F with bounded gradients via known techniques [DBW12].

Conditional value at risk: Intuitively, the CVaR measures performance in the α worst

[1] We use the ℓ_2 norm for concreteness. However, our arguments easily generalize to any ℓ_p norm.

fraction of cases. First, we define the *value at risk* at level $\alpha \in [0, 1]$:

$$\mathrm{VaR}_\alpha(\boldsymbol{x}) = \inf\{\tau \in R : \Pr_y \left[F(\boldsymbol{x}, y) \leq \tau\right] \geq \alpha\}.$$

That is, $\mathrm{VaR}_\alpha(\boldsymbol{x})$ is the α-quantile of the random variable $F(\boldsymbol{x}, y)$. CVaR is the expectation of $F(\boldsymbol{x}, y)$, conditioned on it falling into this set of α-worst cases:

$$\mathrm{CVaR}_\alpha(\boldsymbol{x}) = \mathop{\mathbb{E}}_{y}\left[F(\boldsymbol{x}, y) | F(\boldsymbol{x}, y) \leq \mathrm{VaR}_\alpha(\boldsymbol{x})\right].$$

CVaR is a more popular risk measure than VaR both because it counts the impact of the entire α-tail of the distribution and because it has better mathematical properties [RU00].

Optimization problem: We consider the problem of maximizing $\mathrm{CVaR}_\alpha(\boldsymbol{x})$ over $\boldsymbol{x}$ belonging to some feasible set $\mathcal{P}$. We allow $\mathcal{P}$ to be any downward closed polytope. A polytope is downward closed if there is a lower bound $\boldsymbol{\ell}$ such that $\boldsymbol{x} \succeq \boldsymbol{\ell}$ $\forall \boldsymbol{x} \in \mathcal{P}$ and for any $\boldsymbol{y} \in \mathcal{P}$, $\boldsymbol{\ell} \preceq \boldsymbol{x} \preceq \boldsymbol{y}$ implies that $\boldsymbol{x} \in \mathcal{P}$. Without loss of generality, we assume that $\mathcal{P}$ is entirely nonnegative with $\boldsymbol{\ell} = 0$. Otherwise, we can define the translated set $\mathcal{P}' = \{\boldsymbol{x} - \boldsymbol{\ell} : \boldsymbol{x} \in \mathcal{P}\}$ and corresponding function $F'(\boldsymbol{x}, y) = F(\boldsymbol{x} - \boldsymbol{\ell}, y)$. Let $d = \max_{\boldsymbol{x}, \boldsymbol{y} \in \mathcal{P}} ||\boldsymbol{x} - \boldsymbol{y}||_2$ be the diameter of $\mathcal{P}$.

We want to solve the problem $\max_{\boldsymbol{x} \in \mathcal{P}} \mathrm{CVaR}_\alpha(\boldsymbol{x})$. It is important to note that $\mathrm{CVaR}_\alpha(\boldsymbol{x})$ need *not* be a smooth DR-submodular function in $\boldsymbol{x}$. However, we would like to leverage the nice properties of the underling F. Towards this end, we note that the above problem can be rewritten in a more useful form [RU00]. Let $[t]^+ = \max(t, 0)$. Maximizing $\mathrm{CVaR}_\alpha(\boldsymbol{x})$ is equivalent to solving

$$\max_{\boldsymbol{x} \in \mathcal{P}, \tau \in [0, M]} \tau - \frac{1}{\alpha} \mathbb{E}\left[\left[\tau - F(\boldsymbol{x}, y)\right]^+\right] \tag{6.1}$$

where τ is an auxiliary parameter. For any fixed $\boldsymbol{x}$, the optimal value of τ is $\mathrm{VaR}_\alpha(\boldsymbol{x})$ [RU00]. It is known that when $F(\cdot, y)$ is *concave* in $\boldsymbol{x}$, this is a concave optimization problem.

However, little is known when F may be nonconcave.

6.2 Related work

CVaR enjoys widespread popularity as a risk measure in many domains, ranging from finance [MOS07] to electricity production [YKRW11]. More broadly, there is a burgeoning interest in methods which move beyond expected performance [ECGS11, YJTO11, YN13, HN15]. Oftentimes, this concern is motivated by safety-critical domains where an algorithm designer must be able to minimize the risk of disastrous events, not just guarantee good results on average. Here, we survey the closest related work, dealing with CVaR optimization.

Rockafellar and Uryasev [RU00] introduced CVaR and proposed a linear program for optimizing it. This linear program only applies when utility is linear in the decision variables. Iyengar and Ma [IM13] and Hong and Liu [HL09] present faster gradient-based algorithms for the linear case. Here, we deal with nonlinear functions. The LP approach can be extended via solving a general concave program when the utilities are concave. Our main contribution is extending the range of optimizable functions to include nonconcave continuous submodular objectives. Another body of work focuses on CVaR in reinforcement learning and MDPs [PG13, TCGM15, CTMP15]. Lastly, [OY17] study CVaR for discrete influence maximization; we contrast our results with theirs when we discuss the discrete portfolio setting.

6.3 Preliminaries

We now review techniques for optimizing smooth continuous submodular functions. These do not directly apply to CVaR, but our solution builds on them. An important property is that continuous submodular functions are concave along nonnegative directions. Formally,

Definition 1. A function $F(\boldsymbol{x})$ is *up-concave* if for any $\xi \in [0, 1]$ and $\boldsymbol{y} \in \mathcal{P}$, $F(\boldsymbol{x} + \xi \boldsymbol{y})$ is concave in ξ.

All continuous submodular functions are up-concave [BMBK17]. Monotone up-concave algorithms are optimized via a modified Frank-Wolfe algorithm [BMBK17, CCPV11]. Frank-Wolfe is a gradient-based algorithm originally introduced to maximize concave functions. Consider an objective F. Frank-Wolfe algorithms start at an initial point $\boldsymbol{x}^0 \in \mathcal{P}$ and then generate a series of feasible solutions $\boldsymbol{x}^1 ... \boldsymbol{x}^K$ for some number of iterations K. At each step k, the algorithm calculates the gradient at the current point, $\nabla F(\boldsymbol{x}^{k-1})$. It then takes a step towards the point $\boldsymbol{v}^k \in \mathcal{P}$ which lies furthest in the direction of the gradient. That is, $\boldsymbol{v}^k$ is the solution to the linear optimization problem $\arg\max_{v \in \mathcal{P}} \langle \boldsymbol{v}, \nabla F(\boldsymbol{x}^{k-1}) \rangle$. In the standard Frank-Wolfe algorithm for concave functions, the algorithm then updates to a convex combination of $\boldsymbol{x}^{k-1}$ and $\boldsymbol{v}^k$ by setting $\boldsymbol{x}^k = \boldsymbol{x}^{k-1} + \gamma_k \left(\boldsymbol{v}^k - \boldsymbol{x}^{k-1} \right)$ for some step size γ_k. Note that some entries of $\boldsymbol{x}^k$ may be smaller than the corresponding entries of $\boldsymbol{x}^{k-1}$. This is necessary for optimality: the algorithm may need to backtrack if it has made some entry too large.

This update rule does not work for up-concave functions because the objective is not concave along negative directions. Hence, the update for the modified Frank-Wolfe algorithm is $\boldsymbol{x}^k = \boldsymbol{x}^{k-1} + \gamma_k \boldsymbol{v}^k$, which only increases each coordinate. Because the algorithm is unable to backtrack, it achieves a $(1 - 1/e)$-approximation instead of the global optimum which is achievable for fully concave functions. The process is analogous to the greedy algorithm for submodular set functions, which successively includes elements based on their current marginal gain. The continuous Frank-Wolfe algorithm instead successively increases entries in the solution vector based on the current gradient.

6.4 Algorithmic approach

We now introduce the RASCAL (Risk Averse Submodular optimization via Conditional vALue at risk) algorithm for continuous submodular CVaR optimization. RASCAL solves Problem 6.1, which is a function of both the decision variables $\boldsymbol{x}$ and the auxiliary parameter τ. Roughly, τ should be understood as a threshold maintained by the algorithm for what constitutes a “bad" scenario: at each iteration, RASCAL tries to increase $F(\boldsymbol{x}, y)$ for those

Algorithm 10 RASCAL

```
Require: $K, u, s, LO$
$\mathcal{Y} \leftarrow s$ samples i.i.d. from $D$
$x^0 \leftarrow 0$, $\tau \leftarrow 0$
for $k = 1...K$ do
    $\tilde{\nabla} \leftarrow$ SmoothGrad$(x^{k-1}, \tau, u)$
    $v \leftarrow LO(\tilde{\nabla})$
    $x^k \leftarrow x^{k-1} + \frac{1}{K}v$
    $\tau \leftarrow$ SmoothTau$(x^{k-1}, u)$
end for
return $x^K$

function SmoothGrad($x$, $\tau$, u)
    $I_y(\tau) \leftarrow \max(\min(\frac{F(x,y)-\tau}{u}, 1), 0)\ \forall y \in \mathcal{Y}$
    return $\sum_{y \in \mathcal{Y}} I_y(\tau) \nabla_x F(x, y)$
end function

function SmoothTau($x$, u)
    $\mathcal{B} = \{F(x,y) | y \in \mathcal{Y}\} \cup \{F(x,y) + u | y \in \mathcal{Y}\}$
    Sort $\mathcal{B}$ in ascending order, obtaining $\mathcal{B} = \{b_1 ... b_{|\mathcal{B}|}\}$.
    $i^* = \min\{i = 1...|\mathcal{B}| : g(b_i) > \alpha s\}$
    $A \leftarrow \{y \in \mathcal{Y} : b_{i^*-1} < F(x,y) < b_{i^*}\}$
    $C \leftarrow \{y \in \mathcal{Y} : F(x,y) \leq b_{i^*-1}\}$
    Return the $\tau$ which solves the linear equation
        $$\sum_{y \in A} \frac{F(x,y) - \tau}{u} + |C| = \alpha s$$
end function
```

scenarios y such that $F(x, y) \leq \tau$.

Before describing the optimization algorithm more formally, we deal with the challenge that the expectation in Problem 6.1 cannot generally be evaluated in closed form. We replace the expectation with the average of a set of sampled scenarios. Suppose that we draw a set of samples $y_1 ... y_s$ i.i.d from D. Call the set of samples $\mathcal{Y}$. Then we can estimate $\mathbb{E}\left[[\tau - F(x,y)]^+\right] \approx \frac{1}{s}\sum_{y \in \mathcal{Y}} [\tau - F(x,y)]^+$. With sufficiently many samples, this approximation will be accurate to any desired level of accuracy:

Lemma 10. *Take $s = O\left(\frac{nM^2}{\epsilon^2} \log \frac{1}{\delta} \log \frac{L_1}{\epsilon}\right)$ samples and let $\widehat{CVaR_\alpha}$ be the empirical CVaR on the samples. Then, $|CVaR_\alpha(x) - \widehat{CVaR_\alpha}(x)| \leq \frac{\epsilon}{3}$ holds for all $x \in \mathcal{P}$ with probability at least $1 - \delta$.*

The proof is in the supplement. As a minor technicality, we assume that $F(\boldsymbol{x}, y_i)$ takes a distinct value for each $\boldsymbol{x}$ and $y_i \in \mathcal{Y}$ so that an exact α-quantile exists. This is without loss of generality since we can always add an arbitrarily small "tie breaker" value r_i, using $F(\boldsymbol{x}, y_i) + r_i$ instead.

We can now formally introduce RASCAL (Algorithm 10). RASCAL maximizes the objective $H(\boldsymbol{x}, \tau) = \tau - \frac{1}{\alpha s} \sum_{y \in \mathcal{Y}} [\tau - F(\boldsymbol{x}, y)]^+$. Maximizing H is equivalent to maximizing the sampled CVaR. RASCAL is a coordinate ascend style algorithm. Each iteration first makes a Frank-Wolfe style update to $\boldsymbol{x}$ (lines 4-6). This step assumes access to a linear optimization oracle LO which maximizes a given linear function over $\mathcal{P}$. RASCAL then sets τ to its optimal value given the current $\boldsymbol{x}$ (line 7). This approach is motivated by the unique properties of H. It can be shown that H is jointly up-concave in the variable $(\boldsymbol{x}, \tau)$. However, H is not monotone in τ. Indeed, H is decreasing in τ for $\tau > \text{VaR}_\alpha(\boldsymbol{x})$. The Frank-Wolfe algorithm relies crucially on monotonicity; nonmonotonicity is much more difficult to handle.

Instead, we exploit a unique form of structure in H. Specifically, H is monotone in $\boldsymbol{x}$, but only up-concave (not fully concave). Conversely, while H is nonmonotone in τ, we can easily solve the one-dimensional problem $\max_{\tau \in [0,M]} H(\boldsymbol{x}, \tau)$ for any fixed $\boldsymbol{x}$ (we explain how later). Our approach makes use of both properties: the Frank-Wolfe update leverages monotone up-concavity in $\boldsymbol{x}$, while the update to τ leverages easy solvability of the one-dimensional subproblem.

In order to make this approach work, two ingredients are necessary. First, we need access to the gradient of H in order to implement the Frank-Wolfe update for $\boldsymbol{x}$. Unfortunately, H is not even differentiable everywhere. We instead present a smoothed estimator SmoothGrad which restores differentiability at the cost of introducing a controlled amount of bias. Second, we need to solve the one-dimensional problem of finding the optimal value of τ. We in fact introduce a subroutine SmoothTau which solves a smoothed version of the optimal τ problem.

Smoothed gradient: We now calculate the gradient of the objective with respect to $\boldsymbol{x}$,

$\nabla_x H(x, \tau)$. Essentially, H counts the value of all scenarios y for which $F(x, y) \leq \tau$. If $F(x, y) \neq \tau \, \forall y \in \mathcal{Y}$ then

$$\nabla_x H(x, \tau) = \frac{1}{\alpha s} \sum_{y \in \mathcal{Y}: F(x,y) \leq \tau} \nabla_x F(x, y).$$

Unfortunately, if there is a $y \in \mathcal{Y}$ such that $F(x, y) = \tau$, then H may not be differentiable at x. To see this, consider the directional derivatives from two different directions. From a nonpositive direction, $F(x, y)$ is always below τ and hence will count towards the gradient. From a positive direction, $F(x, y)$ may lie above τ in which case its contribution will be zero. Frank-Wolfe algorithms require differentiability (in fact, they require a Lipschitz gradient). This is not a minor technical point: if the gradient can radically change over small regions, then gradient-based updates may prove fruitless. Thus, RASCAL uses a smoothed gradient estimate over the region from τ to $\tau + u$ for some small $u > 0$:

$$\textsc{SmoothGrad}(x, \tau) = \frac{1}{u} \int_{z=0}^{u} \nabla_x H(x, \tau + z) dz$$

The intuition is that we average over a small window of τ values so that the contribution of a given scenario to the gradient does not suddenly drop to 0 if x increases slightly. Note that as we have sampled a finite set of s scenarios, the set of points at which H is not differentiable has measure 0. Hence, the integral exists. We now show how to exactly evaluate the integral (see Algorithm 1, lines 11-14 for pseudocode). We have

$$\begin{aligned}
&\frac{1}{u} \int_{z=0}^{u} \nabla_x H(x, \tau + z) dz \\
&= \frac{1}{u} \int_{z=0}^{u} \sum_{y \in \mathcal{Y}} 1 \left[F(x, y) \leq \tau + z \right] \nabla_x F(x, y) dz \\
&= \sum_{y \in \mathcal{Y}} \nabla_x F(x, y) \int_{z=0}^{u} \frac{1}{u} 1 \left[F(x, y) \leq \tau + z \right] dz
\end{aligned}$$

where $1[\cdot]$ is the indicator function. Now value of the inner integral is equivalent to $\max(\min(\frac{F(x,y)-\tau}{u}, 1), 0)$. Call this value $I_y(\tau)$. By the above, $\textsc{SmoothGrad}(\boldsymbol{x}, \tau) = \sum_{y \in \mathcal{Y}} I_y(\tau) \nabla_{\boldsymbol{x}} F(\boldsymbol{x}, y)$. This can be computed in time $O(s(T_1 + T_2))$, where T_1 is the time to evaluate F and T_2 is the time to differentiate it.

Finding the optimal $\boldsymbol{\tau}$: The update SmoothTau sets τ to its optimal value over a smoothed window of size u (in order to match SmoothGrad). Specifically, we find the τ minimizing $\frac{1}{u}\int_{z=0}^{u} H(\boldsymbol{x}, \tau) dz$. Recall that for the unsmoothed H, the optimal setting for τ is $\text{VaR}_\alpha(\boldsymbol{x})$, i.e., the value such that F takes value at most τ in an α-fraction of scenarios. An analogous property holds for the smoothed version:

Lemma 11. *Define $g(\tau) = \sum_{y \in \mathcal{Y}} I_y(\tau)$. (a) τ maximizes $\frac{1}{u}\int_{z=0}^{u} H(\boldsymbol{x}, \tau) dz$ if $g(\tau) = \alpha s$. (b) g is piecewise linear and monotone decreasing.*

In Lemma 11(a), the condition $g(\tau) = \alpha s$ expresses that an α-fraction of the scenarios weighted by $I_y(\tau)$ should have $F(\boldsymbol{x}, y) \leq \tau$. The key property for efficiently finding the τ which satisfies this condition is given in Lemma 11(b): g is piecewise linear and monotone decreasing in τ. This follows since it is the sum of functions which share these properties (the I_y). SmoothTau (Algorithm 1, lines 16-23) uses these properties as follows. The breakpoints of g are $F(\boldsymbol{x}, y)$ and $F(\boldsymbol{x}, y) + u$ for each $y \in \mathcal{Y}$ (line 17). Let these breakpoints $\mathcal{B} = \{b_1 ... b_{2s}\}$ be sorted in ascending order. We can find the τ such that $g(\tau) = \alpha s$ by first finding the interval such that $g(b_i) \leq \alpha s \leq g(b_{i+1})$ (line 19). Within this interval, g is linear and hence we can solve exactly for the desired point (lines 20-22). This process takes time $O(sT_1)$.

6.5 Theoretical analysis

We now prove that by taking appropriate choices for the smoothing parameter u and the number of steps K, RASCAL efficiently obtains a provably approximate solution. Our main theoretical result is as follows:

Theorem 12. *For any $\epsilon > 0$, by taking $u = \frac{\epsilon}{3(1+\frac{1}{\alpha})}$, RASCAL outputs a solution $\boldsymbol{x} \in \mathcal{P}$ satisfying $CVaR_\alpha(\boldsymbol{x}) \geq (1 - 1/e)OPT - \epsilon$ with probability at least $1 - \delta$. There are $K = O\left(\frac{L_2 d^2}{\alpha \epsilon} + \frac{L_1 G d^2}{\alpha^2 \epsilon^2}\right)$*

iterations, requiring $O(sK)$ total evaluations of F, $O(sK)$ evaluations of ∇F, and K calls to LO.

The rest of this section is devoted to proving Theorem 28. We start out by introducing a surrogate objective that we consider for the sake of analysis. Let

$$\tilde{H}(\boldsymbol{x}, \tau) = \frac{1}{u} \int_{z=0}^{u} H(\boldsymbol{x}, \tau + z) dz.$$

This is the smoothed version of the objective, which SmoothGrad computes the gradient for. Let $\tau(x) = \max_\tau \tilde{H}(\boldsymbol{x}, \tau)$ be the optimal setting for τ under $\boldsymbol{x}$. Note that this is with respect to the smoothed objective $\tilde{H}$, so $\tau(\boldsymbol{x})$ is not necessarily $\text{VaR}_\alpha(\boldsymbol{x})$. We first show that H and $\tilde{H}$ are close:

Lemma 12. $|\tilde{H}(\boldsymbol{x}, \tau) - H(\boldsymbol{x}, \tau)| \leq \frac{u\left(1+\frac{1}{\alpha}\right)}{2} \;\; \forall \boldsymbol{x}, \tau$

The main idea is to show that H is Lipschitz with respect to τ, so we do not change the value of the function too much by changing τ slightly. This lemma essentially bounds the bias introduced by SmoothGrad.

Now we turn to the main step: showing that the coordinate ascent strategy makes an appropriate amount of progress towards the optimum at each iteration. Note that at the end of each iteration k, RASCAL sets $\tau^k \leftarrow \tau(\boldsymbol{x}^k)$. This is because SmoothTau exactly computes the optimal setting for τ with respect to the smoothed objective $\tilde{H}$. Let $\tilde{\boldsymbol{x}}^* = \max_{\boldsymbol{x} \in \mathcal{P}} \tilde{H}(\boldsymbol{x}, \tau(\boldsymbol{x}))$ be the point achieving the optimal value of $\tilde{H}$. Since RASCAL always sets τ to its optimal value in SmoothTau, the gap from optimality at the end of iteration k is exactly

$$\Delta^k := \tilde{H}(\tilde{\boldsymbol{x}}^*, \tau(\tilde{\boldsymbol{x}}^*)) - \tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k))$$

Our aim is to show that the gap Δ^k decreases by a factor of $(1 - \gamma_k)$ at each iteration (up to a small amount of additive loss). We start out by providing an upper bound on Δ^k in terms of the current gradient.

Lemma 13. *At each iteration $k = 1...K$,*

$$\tilde{H}(\tilde{\boldsymbol{x}}^*, \tau(\tilde{\boldsymbol{x}}^*)) - \tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)) \\ \leq \max_{\boldsymbol{v} \in \mathcal{P}} \langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)), \boldsymbol{v} \rangle.$$

The proof uses the underlying up-concavity of F combined with the concavity-preserving properties of CVaR. The intuition is that any concave function is upper bounded by its linearization at a given point (though the bound is weaker than for concave functions [LJJ15] because F is only up-concave). Lemma 37 gives us a benchmark to track progress: it suffices to show that the improvement in iteration k is at least $\gamma_k \max_{\boldsymbol{v} \in \mathcal{P}} \langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)), \boldsymbol{v}^k \rangle$ since this implies that we make up at least a γ_k fraction of the current gap from optimality.

We now express the actual improvement that is made. At iteration k, the Frank-Wolfe update moves from $\boldsymbol{x}^{k-1}$ to $\boldsymbol{x}^{k-1} + \gamma_k \boldsymbol{v}^k$. Integrating over the transition between these two points gives

$$\tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)) - \tilde{H}(\boldsymbol{x}^{k-1}, \tau(\boldsymbol{x}^{k-1})) = \\ \int_{\xi=0}^{1} \langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^{k-1} + \xi \gamma_k \boldsymbol{v}, \tau(\boldsymbol{x}^{k-1} + \xi \gamma_k \boldsymbol{v})), \gamma_k \boldsymbol{v} \rangle d\xi. \tag{6.2}$$

What we would like is for the gradient to stay relatively constant as we move from $\boldsymbol{x}^{k-1}$ to $\boldsymbol{x}^{k-1} + \gamma_k \boldsymbol{v}^k$. This is because we chose $\boldsymbol{v}^k$ to lie in the direction of $\nabla_{\boldsymbol{x}} \tilde{H}$ at the starting point $\boldsymbol{x}^{k-1}$. If the gradient changes very sharply along the way, then we may not not actually improve the objective value very much.

There are two obstacles to showing that the gradient is smooth enough. The first is that the value of τ in Equation 6.2 may change with ξ. We can deal with this as follows. Note that that since $\boldsymbol{v}^k$ is nonnegative, $\boldsymbol{x}^{k-1} + \xi \gamma_k \boldsymbol{v}^k \succeq \boldsymbol{x}^{k-1}$ holds for all $\xi \in [0, 1]$. It is easy to see that $\tau(\boldsymbol{x})$ is monotone increasing in $\boldsymbol{x}$. Thus, $\tau(\boldsymbol{x}^{k-1} + \xi \gamma_k \boldsymbol{v}) \geq \tau(\boldsymbol{x}^k)$. By looking at the expression for $\nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}, \tau)$, we can see if that if we increase the value of τ, then the gradient can only increase because more scenarios can contribute. Formally,

Lemma 14. *If $\boldsymbol{x}_2 \succeq \boldsymbol{x}_1$, $\nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}_2, \tau(\boldsymbol{x}_2)) \succeq \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}_2, \tau(\boldsymbol{x}_1))$.*

Applying Lemma 38 to Equation 6.2 gives

$$\begin{aligned}&\tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)) - \tilde{H}(\boldsymbol{x}^{k-1}, \tau(\boldsymbol{x}^{k-1}))\\&\qquad \geq \int_{\xi=0}^{1} \langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^{k-1} + \xi \gamma_k \boldsymbol{v}, \tau(\boldsymbol{x}^{k-1})), \gamma_k \boldsymbol{v} \rangle d\xi\end{aligned}$$

The second obstacle is that $\nabla_{\boldsymbol{x}} \tilde{H}$ might change sharply as we vary $\boldsymbol{x}$ from $\boldsymbol{x}^{k-1}$ to $\boldsymbol{x}^{k-1} + \gamma_k \boldsymbol{v}^k$. However, this is exactly what SmoothGrad is designed to avoid. Formally, the gradient of $\tilde{H}$ is Lipschitz:

Lemma 15. *If $\forall y \in \mathcal{Y}$, $F(\cdot, y)$ is L_1-Lipschitz and $\nabla_{\boldsymbol{x}} F(\cdot, y)$ is L_2 Lipschitz with $||\nabla_{\boldsymbol{x}} F||_2 \leq G$, then $\nabla_{\boldsymbol{x}} \tilde{H}$ is $\frac{1}{\alpha}\left(L_2 + \frac{L_1 G}{u}\right)$ $-$Lipschitz.*

This gives us the tools to finish the proof. Let $C = \frac{1}{\alpha}\left(L_2 + \frac{L_1 G}{u}\right)$ be the Lipschitz constant of $\nabla_{\boldsymbol{x}} \tilde{H}$. The Cauchy-Shwartz inequality and Lemma 39 yield

$$\begin{aligned}&\langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^{k-1} + \xi \gamma_k \boldsymbol{v}, \tau(\boldsymbol{x}^{k-1})), \boldsymbol{v} \rangle\\&\geq \langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^{k-1}, \tau(\boldsymbol{x}^{k-1})), \boldsymbol{v} \rangle - \xi \gamma_k C ||\boldsymbol{v}||_2^2\end{aligned}$$

and hence

$$\begin{aligned}&\tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)) - \tilde{H}(\boldsymbol{x}^{k-1}, \tau(\boldsymbol{x}^{k-1}))\\&\geq \gamma_k \int_0^1 \langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^{k-1}, \tau(\boldsymbol{x}^{k-1})), \boldsymbol{v} \rangle - \xi \gamma_k C ||\boldsymbol{v}||_2^2 d\xi\\&= \gamma_k \langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^{k-1}, \tau(\boldsymbol{x}^{k-1})), \boldsymbol{v} \rangle - \frac{\gamma_k^2 C ||\boldsymbol{v}||_2^2}{2}\\&\geq \gamma_k \Delta^{k-1} - \frac{\gamma_k^2 C d^2}{2}\end{aligned}$$

and by rearranging we obtain

$$\Delta^k \leq (1 - \gamma_k) \Delta^{k-1} - \frac{\gamma_k^2 C d^2}{2}.$$

This is exactly what we wanted to show: the gap shrinks by a factor $(1-\gamma_k)$ each iteration, up to a small amount of additive loss. From here, the proof proceeds by fairly standard arguments which may be found in the supplement.

6.6 Discrete portfolio optimization

We may also want to optimize the CVaR of a submodular *set* function, as opposed to the continuous functions that we have dealt with so far. We study the portfolio optimization problem [OY17] where the decision maker may select any *distribution* over feasible sets. Equivalently, they select a decision which is a convex combination of feasible decisions but which is not guaranteed to lie in the original feasible set itself [CLSS17]. This is a natural setting for CVaR optimization because the decision maker essentially hedges their bets between multiple options.

Formally, we are given a collection of submodular set functions $f(\cdot, y)$ on a ground set X, where y is a random variable. There is a collection of feasible sets $\mathcal{I}$. For instance, $\mathcal{I}$ could be all size-k subsets. In general, our algorithm works when $\mathcal{I}$ is any matroid. The algorithm selects a distribution q over the sets in $\mathcal{I}$. The objective is to maximize $\text{CVaR}_\alpha\left(\sum_{S\in\mathcal{I}} q_S f(S, y)\right)$.

We provide a black-box reduction from this problem to the continuous submodular CVaR optimization problem considered earlier. Since RASCAL solves the continuous problem, we immediately obtain efficient algorithms for a range of portfolio problems. Formally,

Theorem 13. *Given access to an α-approximation algorithm for the continuous CVaR problem, there is an algorithm which obtains value at least $\alpha OPT - \epsilon$ for the discrete portfolio CVaR problem.*

A proof is deferred to the supplement. The main idea is to translate from the discrete to continuous settings via the multilinear extension [CCPV11]. The multilinear extension F of a submodular set function f is a continuous function defined on the hypercube $[0,1]^{|X|}$ which agrees with f at the vertices. We apply the promised continuous CVaR algorithm to the multilinear extensions $F(\cdot, y)$ and then use known rounding techniques [CVZ10] to

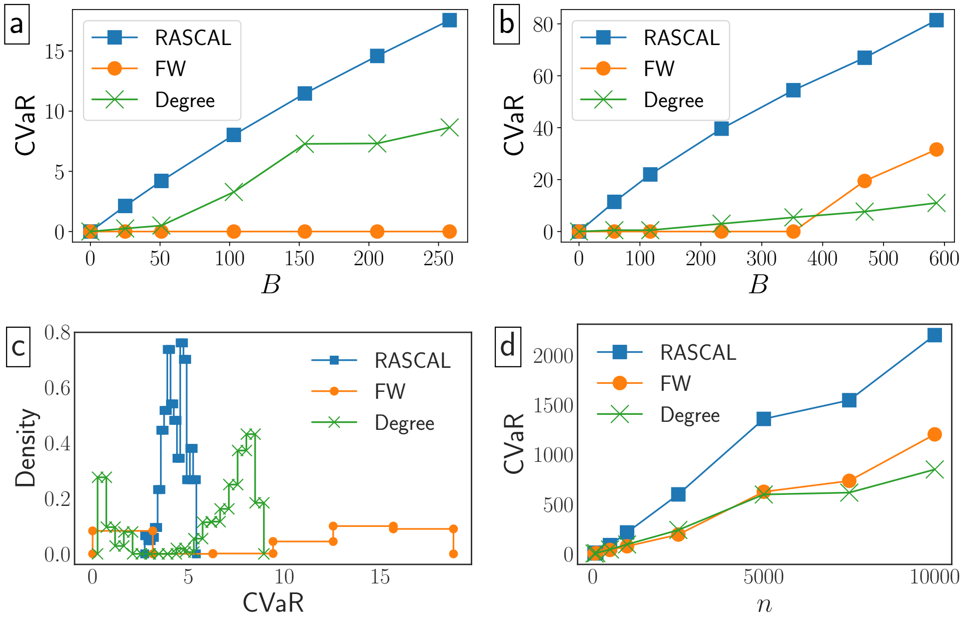


Figure 6.1: *Results for the continuous time independent cascade model. (a) netscience as B varies (b) euroroad as B varies (d) histogram of values for netscience with $B = 0.1n$. (d) Watts-Strogatz networks as n varies*

convert the fractional solution to a distribution over integral points which preserves the fractional solution's CVaR value. However, some additional technical steps are needed to make this strategy work (e.g., we need to maintain multiple copies of the decision variables to get the optimal approximation ratio).

We note that this result strengthens that of Ohsaka and Yoshida [OY17] in two respects. First, their result applies only to influence maximization, while ours applies to any submodular function. Second, they obtain the additive approximation $OPT - \frac{1}{e}$ when the objective values are rescaled by n (the total number of nodes in the graph for influence maximization) to the interval $[0, 1]$. Hence, their bound does not apply when $OPT \leq \frac{1}{e}n$, which is very possible since CVaR counts worst-case outcomes. We have only an arbitrarily small ϵ of additive loss, which allows for stronger guarantees when OPT is small.

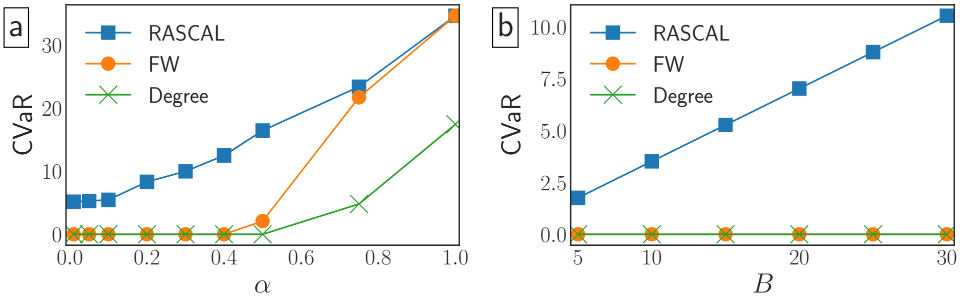


Figure 6.2: *Results for BWSN*

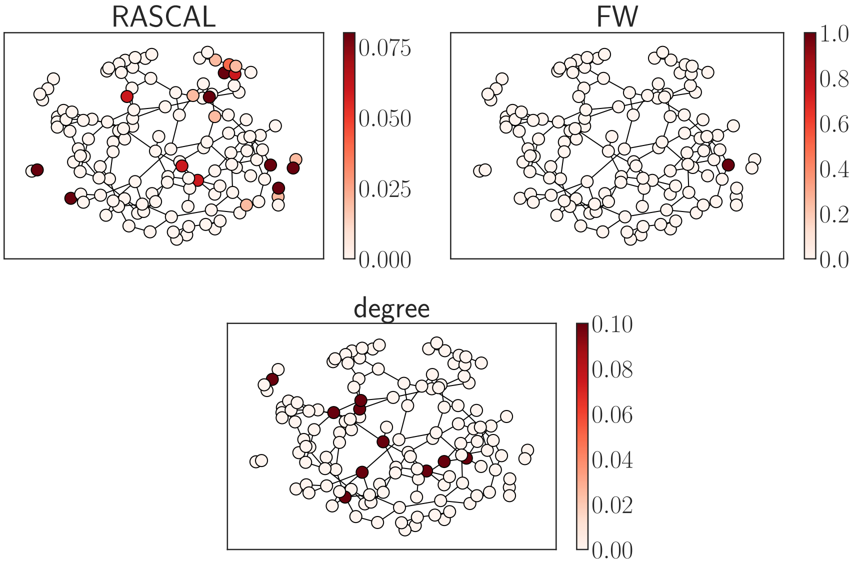


Figure 6.3: *Example allocations for BWSN*

6.7 Experiments

We show experimental results for the sensor resource allocation problem, where the goal is to use a limited sensing budget to quickly detect a contagion spreading through a network [LKG$^+$07, SY15, BMBK17]. We are given a graph $G = (V, E)$ with $|V| = n$. A contagion starts at a random node y and spreads over time according to a given stochastic process. Let t_v be the time at which the contagion reaches each node v. t_v is a random variable which depends on both the source node y and the stochastic contagion process. The vector $\boldsymbol{t}$ collects t_v for all $v \in V$. We assume that $t_v < \infty$ $\forall v \in V$ (every node is eventually reached). If this does not hold, we can cut the process off after some large time horizon. Let t_∞ be the maximum possible value of t_v.

The decision maker has a budget B (e.g., energy) to spend on sensing resources. x_v represents the amount of energy allocated to the sensor at node v. When contagion reaches v at time t_v, the sensor detects with probability $1 - (1 - p)^{x_v}$ and otherwise fails. Essentially, investing an extra unit of energy in sensor v buys an extra chance to detect the contagion with probability p. Fix a vector of times $\boldsymbol{t}$, and order the nodes $v_1 ... v_n$ so that $t_{v_1} \leq t_{v_2} \leq ... \leq t_{v_n}$. The objective F for source y is expected amount of detection time that is saved by the sensor placements:

$$F(\boldsymbol{x}, \boldsymbol{t}) = t_\infty - \sum_{i=1}^{n} t_{v_i} \left(1 - (1 - p)^{x_i}\right) \prod_{j<i} (1 - p)^{x_j}$$

where the summation counts the probability that sensor i succeeds but all $j < i$ fail. Is is known that F is DR-submodular [BMBK17]. Previous work maximizes $\mathbb{E}_{\boldsymbol{t}} \left[F(\boldsymbol{x}, \boldsymbol{t})\right]$, the expected utility over the random source node and diffusion process. Here, we consider instead $\text{CVaR}_\alpha(\boldsymbol{x})$, where the scenarios are all possible time vectors $\boldsymbol{t}$. Essentially, we want to perform well when the contagion starts in hard to detect portions of the network or spreads in an unlikely way. We take the CVaR with respect to $\boldsymbol{t}$ but not the success or failure of the sensors because no algorithm can successfully detect contagion when almost all sensors fail *and* the source and diffusion pattern are worst-case.

Domains: We consider two sensing domains. In both, the source node is uniformly random. First, contagion spreading according to the continuous time independent cascade model (CTIC). This models applications like detecting news or a disease in a social network. The CTIC is variant of the independent cascade model proposed by Gomez-Rodriguez et al. [GRLK12] which better reflects the temporal dynamics of real-world social processes. Each edge (u, v) has propagation time $\rho_{u,v}$ drawn from an exponential distribution with mean λ. The contagion starts at y ($t_y = 0$). Letting $\delta(v)$ be v's neighbors, $t_v = \min_{u \in \delta(v)} t_u + \rho_{u,v}$. That is, t_v is the first time contagion spreads from a neighbor to v.

We show experiments on several networks. First, *netscience*[2]: a collaboration network of network science researchers with 1461 nodes. Second, *euroroad*: a network of European cities and roads between them, with 1,174 nodes. Third, synthetic Watts-Strogatz networks (parameters k = 2, p = 0.1). These allow us to test our algorithm on a similar graphs as n grows. For all networks, $\lambda = 5$, $p = 0.01$, and we simulate 1000 scenarios (random source nodes and propagation times).

Second, we consider detecting contamination in a water network via the Battle of Water Sensor Networks (BWSN). BWSM [OUS$^+$08] simulates the spread of contamination through a 126-node water network consisting of junctions, tanks, pumps, and the links between them. The network is a real water distribution network from an anonymous location, and the t values are provided by EPANET, a highly realistic water distribution simulator designed by the U.S. Environmental Protection Agency. We use $p = 0.001$ and simulate 1000 random scenarios (source node and t values).

Baselines: No previous work directly addresses our setting. We consider two competitive baselines. First, *FW*, which uses the Frank-Wolfe algorithm of Bian et al. [BMBK17] to maximize the expected reward. Maximizing expected value is default approach to decision making under uncertainty. Second, *degree*, a heuristic for producing risk-averse solutions. Specifically, degree allocates one unit of budget to each of the B nodes with highest degree. This disperses the budget throughout the network, hedging against unlikely outcomes.

[2] http://www-personal.umich.edu/ mejn/netdata/

Results for CTIC: Figure 6.1 shows results under the CTIC. Figures 6.1(a) and 6.1(b) show the CVaR of each algorithm on the netscience and euroroad networks as the budget B varies on the x axis. RASCAL substantially outperforms both FW and Degree. This indicates that maximizing expected value is not a sufficient proxy for risk-aversion under uncertainty. In fact, FW obtains *zero* value for many values of B, indicating that its sensor selection is useless in the 10% worst cases. Degree often performs better than FW, indicating some benefit to heuristically hedging against possible contagion sources. However, RASCAL's principled optimization still results in much higher performance. Figure 6.1(c) shows a histogram of each algorithm's value across the different scenarios on netscience. RASCAL's reward distribution is tightly concentrated, which is desirable from the perspective of risk aversion. By contrast, FW and degree have more bimodal distributions, with the potential for both very low and high reward. Lastly, Figure 6.1(d) shows the CVaR obtained by each algorithm for Watts-Strogatz networks as the network size n grows on the x axis. RASCAL again obtains much higher value across the board. RASCAL scales easily to 10,000 nodes, running in under 1 minute.

Results for BWSN: We now examine our second domain, water network sensor management. Figure 6.2 shows the CVaR obtained by each algorithm. Figure 6.2(a) shows α on the x axis, varying the decision maker's degree of risk aversion. Throughout, $B = 10$. RASCAL substantially outperforms FW and degree until $\alpha = 0.6$, at which point FW becomes competitive. However, for $\alpha \leq 0.4$, both FW and degree obtain zero value. This indicates that even when the decision maker is not severely risk averse (e.g., preferring to focus on the worst 50% of scenarios), they can substantially benefit from using our principled approach to optimizing CVaR. It is natural to ask whether the baselines are competitive when there are more resources available, allowing them to cover a larger portion of the network. Figure 6.2(b) shows the results as the budget B is varied on the x axis with $\alpha = 0.1$. FW and degree still obtain a CVaR of zero even when the budget is tripled to $B = 30$. By contrast, RASCAL's value steadily grows as it makes productive use of the additional resources.

Lastly, Figure 6.3 shows an example of the allocation produced by each algorithm for

$B = 10$, $\alpha = 0.1$. RASCAL disperses its resources throughout the network. It places some resources on central nodes, but also spends a portion of the budget on outlying parts of the network where contagions will not be detected by centrally placed sensors. On the other hand, FW concentrates is *entire* budget on one central node. Degree, by design, disperses its budget more widely. However, it spends the budget largely on central nodes, instead of balancing between central and outlying nodes like RASCAL. We conclude that RASCAL successfully balances different scenarios to find risk-averse solutions.

Chapter 7

Fairness in influence maximization

Influence maximization in social networks is a well-studied problem with applications in a broad range of domains. Consider, for example, a group of at-risk youth; outreach programs try to provide as many people as possible with useful information (e.g., HIV safety, or available health services). Since resources (e.g., social workers) are limited, it is not possible to personally reach every at-risk individual. It is thus important to target *key community figures* who are likely to spread vital information to others. Formally, individuals are nodes V in a social network, and we would like to influence or *activate* as many of them as possible. This can be done by initially *seeding* k nodes (where $k \ll |V|$). The seed nodes activate their neighbors with some probability, who activate their neighbors and so forth. Our goal is to identify k seeds such that the maximal number of nodes is activated. This is the classic *influence maximization problem* [KKT03], that has received much attention in the literature.

In recent years, the influence maximization framework has seen application to many social problems, such as HIV prevention for homeless youth [YWR$^+$18, WOVH$^+$18], public health awareness [VP07], financial inclusion [BCDJ13], and more. Frequently, small and marginalized groups within a larger community are those who benefit the most from attention and assistance. It is important, then, to ensure that the allocation of resources reflects and respects the diverse composition of our communities, and that each group receives a fair allocation of the community's resources. For instance, in the HIV prevention domain we may wish to ensure that members of racial minorities or of LGBTQ identity are not disproportionately excluded; this is where our work comes in.

Our Contributions

This chapter introduces the problem of fair resource allocation in influence maximization. Our *first contribution* is to propose fairness concepts for influence maximization. We start with a *maximin* concept inspired by the legal notion of disparate impact; formally it requires us to maximize the minimum fraction of nodes within each group that are influenced. While intuitive and well-motivated, this definition suffers from shortcomings that lead us to introduce a second concept, *diversity constraints*. Roughly, diversity constraints guarantee that every group receives influence commensurate with its "demand", i.e., what it could have generated on its own, based on a number of seeds proportional to its size. Here, to compute a group's demand, we allow it a number of seeds proportional to its size, but require that it spreads influence using only nodes in the group. Hence, a small but well connected group may have a better claim for influence than a large but sparsely connected group.

Our *second contribution* is an algorithmic framework for finding solutions that satisfy either fairness concept. While the classical influence maximization problem is submodular (and hence easily solved with a greedy algorithm), fairness considerations produce strongly non-submodular objectives. This renders standard techniques inapplicable. We show that both fairness concepts can be reduced to *multi-objective* submodular optimization problems, which are substantially more complex. Our key algorithmic contribution is a new method for general multi-objective submodular optimization which has substantially better approximation guarantee than the current best algorithm [Udw18], and often better runtime as well. This result may be of independent interest.

Our *third contribution* is an analytical exploration of the *price of group fairness* in influence maximization, i.e., the reduction in social welfare with respect to the unconstrained influence maximization problem due to imposing a fairness concept. We show that the price of diversity can be high in general for both concepts and under a range of settings.

Our *fourth contribution* is an empirical study on real-world social networks that have been used for a socially critical application: HIV prevention for homeless youth. Our results show that standard influence maximization techniques often cause substantial fairness violations by neglecting small groups. Our proposed algorithm substantially reduces such violations at relatively small cost to overall utility.

Related Work

[KKT03] introduced influence maximization and proved that since the objective is submodular, greedily selecting nodes gives a $\left(1 - \frac{1}{e}\right)$-optimal solution. There has since been substantial interest among the AI community both in developing more scalable algorithms (see [LFWT18] for a recent survey) , as well as in addressing the challenges of deployment in public health settings [YCXJ$^+$16, WIRT18]. Recently, such algorithms have been used in real-world pilot tests for HIV prevention amongst homeless youth [YWR$^+$18, WOVH$^+$18], driving home the need to consider fairness as influence maximization is applied in socially sensitive domains. To our knowledge, no previous work considers fairness specifically for influence maximization. Some literature exists on targeted influence maximization problems [PNR15, WPS18, CLFC19] where the objective is to reach a specific set of nodes and not others; by contrast, our goal is to ensure that every group receives a fair amount of influence spread. The techniques we introduce to optimize fairness metrics are related to research on multi-objective submodular maximization (outside the context of fairness), and we improve existing theoretical guarantees for this general problem [CVZ10, Udw18].

Outside of influence maximization, the general idea of diversity as an optimization constraint has received considerable attention in recent years; it has been studied in multi-winner elections (see [BFI$^+$18, FSST17] for an overview), resource allocation [BCH$^+$18], and matching problems [ADF17, HHK$^+$17]. We note that some of the above works (e.g. [ADF17] and [SCFD17]) use a submodular objective function as a means of achieving diversity; interestingly, while the classic influence maximization target function is submodular, it is no longer so under diversity constraints. Group fairness has been studied extensively in the voting theory literature, where the objective is to identify a committee of k candidates that will satisfy subsets of voters (see a comprehensive overview in [FSST17]). There have also been several works on group fairness in fair division, defining notions of group envy-freeness [CFSV19, FMS18, SHS18, TLH$^+$11], and a group maximin share guarantee [BBKN19, Suk18]. One line of work in operations research uses mixed-integer programming to enforce that different groups receive the equal utility (or at least that each group's utility satisfies some lower bound) [BFT13, AVW$^+$18]. This manner of defining fairness is relatively close to our own; e.g., our diversity constraints give one way of instantiating what this lower bound should be. Computationally, we introduce efficient algorithms specifically for the submodular optimization instead of using mixed-integer programming.

7.1 Model

Agents are embedded in a social network $G = (V, E)$. An edge $(i, j) \in E$ represents the ability for agent v_i to influence or *activate* v_j. G may be undirected or directed.

Diversity

Each agent in our network may identify with one or more groups within the larger population. These represent different ethnicities, genders, sexual orientations, or other groups for which fair treatment is important. Our goal is to maximize influence in a way such that each group receives at least a "fair" share of influence (more on this below). Let us designate these groups as $\mathcal{C} = \{C_1, \ldots C_m\}$. Each **group** C_i represents a non-empty subset of V, $\emptyset \neq C_i \subseteq V$. Each agent must belong to at least one group, but may belong to multiple groups; i.e. $C_1 \cup C_2 \cup \ldots C_m = V$. In particular, this allows for the expression of intersectionality, where an individual may be part of several minority groups.

Influence Maximization

We model influence using the *independent cascade model* [KKT03], the most common model in the literature. All nodes begin in the inactive state. The decision maker then selects k *seed nodes* to activate. Each node that is activated makes one attempt to activate each of its inactive neighbors; each attempt succeeds independently with probability p (all of our results also hold for nonuniform probabilities). Newly activated nodes attempt to activate their neighbors and so on, with the process terminating once there are no new activations.

We define the *influence* of nodes $A \subseteq V$, denoted $\mathcal{I}_G(A)$, as the expected number of nodes activated by seeding A. Of these, let $\mathcal{I}_{G,C_i}(A)$ be the expected number of activated vertices from C_i. Traditional influence maximization seeks a set A, $|A| \leq k$, maximizing $\mathcal{I}_G(A)$. Using a slight abuse of notation, let $\mathcal{I}_G(k)$ be the maximum influence that can be achieved by selected k seed nodes. That is, $\mathcal{I}_G(k) = \max_{|A|=k} \mathcal{I}_G(A)$. Analogously, we define $\mathcal{I}_{G,C_i}(k)$ as the maximum expected number of vertices from C_i that can be activated by k seeds. We now propose two means of capturing group fairness in influence maximization.

Maximin Fairness

Maximin Fairness captures the straightforward goal of improving the conditions for the *least well-off* groups. That is, we want to maximize the minimum influence received by any of the groups, as proportional to their population. This leads to the following utility function based on seed nodes A:

$$U^{\text{Maximin}}(A) = \min_i \frac{\mathcal{I}_{G,C_i}(A)}{|C_i|}$$

Subject to this maximin constraint, we seek to maximize overall influence. Thus, we define $\mathcal{I}_G^{\text{Maximin}} = \mathcal{I}_G(B)$ with $B = \arg\max_{A \subseteq V, |A|=k} U^{\text{Maximin}}(A)$. That is, $\mathcal{I}_G^{\text{Maximin}}$ is the expected number of nodes activated by a seed configuration that maximizes the minimum proportional influence received by any group. This corresponds to the legal concept of *disparate impact*, which roughly states that a group has been unfairly treated if their "success rate" under a policy is substantially worse than other groups (see [BS16] for an overview).

Therefore, maximin fairness may be significant to governmental or community organizations which are constrained to avoid this form of disparity. However, optimizing for equality of outcomes may be undesirable when some groups are simply much better suited than others to a network intervention. For instance, if one group is very poorly connected, maximin fairness would require that large number of nodes be spent trying to reach this group, even though additional seeds have relatively small impact.

Diversity Constraints

We now propose an alternate fairness concept by extending the notion of individual rationality to *Group Rationality*. The key idea is that no group should be better off by leaving the (influence maximization) game with their proportional allocation of resources and allocating them internally. For each group C_i, let $k_i = \lceil k|C_i|/|V|\rceil$ be the number of seeds that would be fairly allocated to the group C_i based on the group's size within the larger population, rounded up to remove any doubt that this group receives a fair share. k_i is the *fair allocation* of seeds to the group.

Let $G[C_i]$ be the subgraph induced from G by the nodes C_i. This represents the network formed by group C_i if they were to separate from the original network. Now, we define the *group rational influence* that each group C_i can expect to receive as the number of nodes they expect to activate if they left the network, with their fair allocation of k_i seeds. We denote this group rational influence for C_i as $\mathcal{I}_{G[C_i]}(k_i)$. Then, we devise a set of *diversity constraints* that any group rational seeding configuration A with k seeds must satisfy: $\mathcal{I}_{G,C_i}(A) \geq \mathcal{I}_{G[C_i]}(k_i), \forall i$. That is, the influence received by each group is at least equal to what each group may accomplish on its own when given its fair share of k_i seed nodes.

The diversity constraint objective function is to maximize the expected number of nodes activated, subject to the above diversity constraint. The utility for selecting seed nodes A is:

$$U^{\text{Rational}}(A) = \begin{cases} \mathcal{I}_G(A), & \text{if } \mathcal{I}_{G,C_i}(A) \geq \mathcal{I}_{G[C_i]}(k_i), \forall i. \\ 0, & \text{otherwise.} \end{cases}$$

The maximum expected influence obtained via a group rational seeding configuration A is called the *rational influence* $\mathcal{I}_G^{\text{Rational}} = \mathcal{I}_G(B)$, where $B = \arg\max_{A \subseteq V, |A|=k} U^{\text{Rational}}(A)$. Note that since even the standard influence maximization problem is already NP-hard and must be approximated, our computational guarantees will relax the above constraint, requiring that each group receive influence within some factor α of $\mathcal{I}_{G[C_i]}(k_i)$.

Price of Fairness

To measure the cost of ensuring a fair outcome for the diverse population, we will measure the Price of Fairness, the ratio of optimal influence to the best achievable influence under our two fairness criteria. Here *optimal influence* $\mathcal{I}_G^{\text{OPT}} = \mathcal{I}_G(k)$, which is the maximum amount of expected influence that can be obtained using any choice of k seed nodes. We omit the subscript where the context is clear.

$$PoF^{\text{Rational}} = \frac{\mathcal{I}^{\text{OPT}}}{\mathcal{I}^{\text{Rational}}} \quad PoF^{\text{Maximin}} = \frac{\mathcal{I}^{\text{OPT}}}{\mathcal{I}^{\text{Maximin}}}$$

7.2 Optimization

The standard approach to influence maximization is based on *submodularity*. Formally, a set function f on ground set V is submodular if for every $A \subseteq B \subseteq V$ and $x \in V \setminus B$, $f(A \cup \{x\}) - f(A) \geq f(B \cup \{x\}) - f(B)$. This captures the intuition that additional seeds provide diminishing returns. However, both of our fairness concepts are easily shown to violate this property (proofs are deferred to the appendix):

Theorem 14. *U^{Maximin} and U^{Rational} are not submodular.*

We remark that each individual function $\mathcal{I}_{G,C_i}$, i.e., the number of nodes in group i who are reached, is submodular. However, this property does not hold for the combined objectives U^{Maximin} and U^{Rational}. Hence, we cannot apply the greedy heuristic to group-fair influence maximization. Instead, we now show that optimizing either utility function reduces to *multiobjective submodular maximization*, a more general problem defined as follows. The input to the problem is a set of monotone submodular functions $f_1 ... f_m$ and corresponding target values $W_1 ... W_m$. We assume that the f_i are normalized ($f_i(\emptyset) \geq 0$). The multiobjective submodular maximization problem is to find a set S satisfying $|S| \leq k$ with $f_i(S) \geq W_i$ for all i, assuming that such an S exists.

7.2.1 Reduction to Multiobjective Submodular Maximization

We now show that each of the fairness-aware influence maximization objectives can be reduced to solving a small number of instances of multiobjective submodular maximization with appropriately chosen functions f_i and targets W_i. Our reductions leverage the property that the underlying influence functions $\mathcal{I}_{G,C_i}$ are submodular even though the group-fair objectives are not. We start with U^{Maximin}. Here, we define $f_i = \frac{\mathcal{I}_{G,C_i}}{|C_i|}$ to be group i's influence spread normalized by the size of the group. All of the target values W_i will be equal, i.e., $W_1 = W_2 = ... W_m = W$. Assume that we have a subroutine for multiobjective submodular maximization. If the multiobjective problem is feasible for a given value of W, then the subroutine outputs a set S satisfying $U^{\text{Maximin}}(S) \geq W$. Hence, we simply binary search for the highest value of W for which the multiobjective problem remains feasible.

For U^{Rational}, we let $f_i = \mathcal{I}_{G,C_i}$ and set the target $W_i = \mathcal{I}_{G[C_i]}(k_i)$. This represent the constraint that group i must receive at least their group-rational share of utility. We then add another objective function $f_{\text{total}} = \mathcal{I}_G$ representing the combined utility and binary search for the highest value W_{total} such that the targets $W_1 ... W_m, W_{\text{total}}$ are feasible. This represents the largest achievable total utility, subject to diversity constraints. Having reduced both fairness concepts to multiobjective submodular maximization, we turn to algorithms for this core problem. We present an algorithm with substantially improved theoretical guarantees for the general multiobjective problem, and then show how our algorithm can be applied to fair influence maximization.

7.2.2 Previous Techniques

The multiobjective submodular problem was introduced by Chekuri et al. [CVZ10], who gave an algorithm which guarantees $f_i \geq (1 - \frac{1}{e})W_i$ for all i provided that the number of objectives m is smaller than the budget k (when $m = \Omega(k)$, the problem is provably inapproximable [KMGG08]). Unfortunately, this algorithm is of mostly theoretical interest since it runs in time $O(n^8)$. Udwani [Udw18] recently introduced a practically efficient algorithm; however it obtains an asymptotic $(1 - \frac{1}{e})^2$-approximation instead of the optimal $\left(1 - \frac{1}{e}\right)$. We remedy this gap by providing a practical algorithm obtaining an asymptotic $\left(1 - \frac{1}{e}\right)$-approximation (Algorithm 11). Its runtime is comparable to, and under many conditions faster than, the algorithm of [Udw18].

Previous algorithms [CVZ10, Udw18] start from a common template in submodular optimization, which we also build on. The main idea is to relax the discrete problem to a continuous space. For a given submodular function f, its *multilinear extension* F is defined on n-dimensional vectors x where $0 \leq x_j \leq 1$ for all $j \in V$. x_j represents the probability that item j is included in the set. Formally, let $S \sim x$ denote a set which includes each j independently with probability x_j. Then, we define $F(x) = \mathbb{E}_{S \sim x}[f(S)]$, which can be evaluated using random samples.

7.2.3 Algorithm Overview

The main challenge is to solve the continuous optimization problem, which is where our technical contribution lies. Algorithm 11 describes the high-level procedure, which runs our continuous optimization subroutine (line 2) and then rounds the output to a discrete set (line 3). Line 1, which ensures that all items with value above a threshold τ are included in the solution, is a technical detail needed to ensure the rounding succeeds. The rounding process captured in lines 1 and 3 is fairly standard and used by both previous algorithms [CVZ10, Udw18]. Our main novelty lies in an improved algorithm for the continuous problem, MultiFW.

Algorithm 11 Multiobjective Optimization$(\gamma, \tau, T, T', \eta)$

```
$S_1 = \{j : f_i(\{j\}) \geq \tau$ for some $i\}$
$x =$MultiFW$(k - |S_1|, \{\gamma\,(W_i - f_i(S))\}_{i=1}^m)$
$S_2 =$SwapRound$(x_{\text{int}})$ //see [CVZ10]
return $S_1 \cup S_2$
```

Algorithm 12 Multiobjective Frank-Wolfe$(k, \{W_i\})$

```
1: $x^0 = 0$
2: for $t = 1...T$ do
3:     $v^t =$ S-SP-MD$(x, \{i : W_i - F_i(x^{t-1}) \geq \epsilon\})$
4:     $x^t = x^{t-1} + \frac{1}{T} v^t$
5: end for
6: return ApproxDecomposition$(x^T)$ //see [MLVW17]
7: function S-SP-MD$(x, \mathcal{I})$
8:     Initialize $v$ s.t. $||v||_1 = k$ and $y \in \Delta(\mathcal{I})$ arbitrarily
9:     for $\ell = 1...T'$ do
10:        Sample $i \sim y$; set $\hat{\nabla}_v = \frac{1}{W_i - F_i(x)} \mathcal{A}^i_{\text{grad}}(x)$
11:        Sample $j \sim v$; $\hat{\nabla}_y = k \cdot \text{diag}\left(\frac{1}{\vec{W} - \vec{F}(x)}\right) \mathcal{A}^j_{\text{item}}(x)$
12:        $y = \frac{y e^{-\eta \hat{\nabla}_y}}{||y e^{-\eta \hat{\nabla}_y}||_1}$
13:        $v = k \frac{\min\{v e_v^{\eta \hat{\nabla}}, 1\}}{||\min\{v e_v^{\eta \hat{\nabla}}, 1\}||_1}$
14:    end for
15: end function
```

MultiFW implements a Frank-Wolfe style algorithm to simultaneously optimize the multilinear extensions $F_1 ... F_m$ of the discrete objectives. The algorithm proceeds over T iterations. Each iteration first identifies v^t, a good feasible point in continuous space (Algorithm 12, line 3). Then, the current solution x^t is updated to add $\frac{1}{T} v^t$ (line 4). Since each point v^t is feasible, x^t is a convex combination of feasible points and hence always remains feasible. The key to the algorithm is a good choice of the direction v^t at each iteration. Roughly, we would like to choose v^t in a way that ensures we make progress towards meeting the target W_i for each F_i. If our current solution quality $F_i(x^{t-1})$ is very far from W_i then v^t should focus heavily on improving the value of F_i. By contrast, if $F_i(x^{t-1})$ is already close to W_i, then v^t should focus on improving other objectives instead. This process is formally accomplished via a subroutine S-SP-MD which we introduce in the subsection below.

The output of Algorithm 12 is the final point x^T produced after T iterations. There is one technical detail to take care of, reflected in line 5. Common rounding algorithms for submodular maximization require not just the fractional point x^T, but also a representation of x^T as a convex combination of integral points, i.e., as a combination of binary vectors representing feasible sets. The rounding algorithm will then merge these binary vectors together to produce the final output set. Producing this convex combination is the well-known *Caratheodory* problem of decomposing a point in a polytope into a combination of vertices. While the problem can be solved exactly via convex optimization, doing so may incur unnecessarily high runtime. To reduce the time complexity of the algorithm, we find the decomposition via an approximate method recently introduced by [MLVW17]. The details of this method are unimportant (we use it just as a black-box; any method for solving the Caratheodory problem would suffice). In our theoretical analysis, we show that the loss in solution quality due to using an approximate decomposition is negligible (formally, an arbitrarily small ϵ).

7.2.4 Choosing the Direction

The key challenge is to efficiently find a v^t that makes sufficient progress towards *every* objective simultaneously. We accomplish this by introducing the subroutine S-SP-MD (lines 6-12), which runs a carefully constructed version of stochastic saddle-point mirror descent [NJLS09]. We first motivate and formalize the problem that S-SP-MD attempts to solve. Then, we give some background on mirror descent and explain the steps of the algorithm.

As explained earlier, v^t must be chosen so that it makes progress towards those objectives for which $W_i - F_i(x^{t-1})$ is large (i.e., we are far from the target). As a first step, we will ignore all i for which $W_i - F_i(x^{t-1}) < \epsilon$, since for these objectives the current solution is already sufficiently good. Let $\mathcal{I}$ denote the set of remaining objectives where $W_i - F_i(x^{t-1}) \geq \epsilon$. For each i, let $\nabla F_i(x^{t-1})$ denote the gradient of F_i. We will use the gradients of the functions in

$\mathcal{I}$ to choose v^t. Specifically, our goal is to find a feasible v such that

$$\nabla F_i(x^{t-1}) \cdot v \geq W_i - F_i(x^{t-1}) \; \forall i \in \mathcal{I}. \tag{7.1}$$

It can be shown that such a v always exists whenever the overall multiobjective problem is feasible. If we can find this v, the progress we make at each iteration is proportional to our current gap from the targets, resulting in the desired $(1 - 1/e)$-approximation after sufficiently many iterations. Note that the LHS of Problem 7.1 is linear in the decision variable v, while the RHS is constant with respect to v. This implies that we could (in principle) find a feasible v via linear programming. Naively however, this approach would entail $O(n^3)$ runtime per iteration.

Our first step towards an efficient solution is to convert Problem 7.1 into a single maxmin problem. Specifically, we can solve the problem

$$\max_{||v||_1 \leq k} \min_{i \in \mathcal{I}} \frac{\nabla F_i(x^{t-1}) \cdot v}{W_i - F_i(x^{t-1})} \tag{7.2}$$

and it is easy to see that if a solution v has objective value at least 1 for the maxmin Problem 7.2, then it is also feasible for Problem 7.1. We now make a final reformulation to obtain a problem amenable to optimization. Specifically, let $\Delta(\mathcal{I})$ denote the set of all distributions over $\mathcal{I}$. We will consider the saddle-point problem

$$\max_{||v||_1 \leq k} \min_{y \in \Delta(\mathcal{I})} \sum_{i \in \mathcal{I}} y_i \frac{\nabla F_i(x^{t-1}) \cdot v}{W_i - F_i(x^{t-1})} \tag{7.3}$$

where the min now ranges over all *distributions* over $\Delta(\mathcal{I})$ instead of single elements. It is easy to see that the solutions of the two problems are equivalent (since the minimizing distribution will always put probability one on a single element). However, replacing the discrete min with one over a continuous set allows us to draw on continuous optimization techniques to obtain an efficient solution, as explained in the next section.

7.2.5 Stochastic Saddle-Point Method

We employ a method based on stochastic saddle-point mirror descent (S-SP-MD), introduced by [NJLS09]. Essentially, this algorithm views Problem 7.3 as a game between a max player and a min player. Both players update their decision variables (v and y respectively) by using gradient updates based on the objective in Problem 7.3. Intuitively, the min player will put large weights where the max player is doing badly, forcing the max player to improve v. The algorithm uses two key ideas to make this process efficient. The first is that, instead of using standard gradient descent, mirror descent modifies the updates to better exploit the structure of the feasible set. For our case, this results in the exponentiated gradient updates given in lines 11-12 of Algorithm 12. Essentially, each player multiplies their current solution by $e^{-\eta \nabla}$, where ∇ is that player's current gradient and η is a learning rate. Then, they rescale to maintain feasibility.

However, this process assumes that the gradients ∇ are easily available, an assumption that does not hold in common submodular problems. For instance, for influence maximization the gradients depend on the random influence process and cannot be calculated exactly. Hence, the second key idea uses stochastic methods to efficiently estimate the gradients (e.g., using simulations of influence spread). Formally, instead of assuming that ∇F_i can be computed exactly, we will instead make the much weaker assumption that we can obtain an unbiased estimate of it, i.e., a random vector $\hat{\nabla}$ satisfying $\mathbb{E}[\hat{\nabla}] = \nabla F_i$. Efficient estimates of this form are known for many submodular problems (e.g., coverage functions or facility location [KLHK17]), and we show below how to create one for influence maximization. However, even using stochastic gradients may entail unnecessarily high runtime since we still have to compute the estimated gradients for every objective i with respect to every item (node) j. Accordingly, our proposed updates use more restricted oracles that return stochastic estimates of only a *subset* of the full gradients. This is a key element of obtaining near-linear runtime.

Specifically, we assume access to two gradient oracles. First, a stochastic gradient oracle $\mathcal{A}^i_{\text{grad}}$ for each multilinear extension F_i. Given a point x, $\mathcal{A}^i_{\text{grad}}(x)$ satisfies $\mathbb{E}[\mathcal{A}^i_{\text{grad}}] =$

$\nabla_x F_i(x)$. Second, a stochastic gradient oracle $\mathcal{A}^j_{\text{item}}$ corresponding to each item $j \in [n]$ (in influence maximization, the items are the potential seed nodes). $\mathcal{A}^j_{\text{item}}(x)$ satisfies $\mathbb{E}[\mathcal{A}^j_{\text{item}}(x)] = \left[\nabla_{x_j} F_1(x) ... \nabla_{x_j} F_m(x)\right]$. We assume that $||\mathcal{A}^i_{\text{grad}}(x)||_\infty, ||\mathcal{A}^j_{\text{item}}(x)||_\infty \leq c$ for some constant c.

Our algorithm calls $\mathcal{A}^i_{\text{grad}}$ and $\mathcal{A}^j_{\text{item}}$ for only a *single* i and j each iteration (instead of enumerating over all i, j as would be naively required). The results are then scaled so that they remain unbiased estimates of the true gradients. The process is formally shown in lines 8-9 of Algorithm 12. Line 8 computes gradients with respect to v for the maximizing player, a process which works as follows. Differentiating Equation 7.2 with respect to v, we obtain

$$\begin{aligned}\nabla_v &= \sum_{i \in \mathcal{I}} y_i \frac{\nabla F_i(x^{t-1})}{W_i - F_i(x^{t-1})} \\ &= \mathop{\mathbb{E}}_{i \sim y}\left[\frac{\nabla F_i(x^{t-1})}{W_i - F_i(x^{t-1})}\right] \end{aligned} \tag{7.4}$$

where $i \sim y$ denotes drawing i at random according to the probability distribution y. From this expression, we see that an unbiased estimate of Equation 7.4 can be obtained by first sampling a single $i \sim y$, and then calling $\mathcal{A}^i_{\text{grad}}$ to obtain an unbiased estimate of $\nabla F_i(x^{t-1})$. We then return $\frac{1}{W_i - F_i(x^{t-1})}\mathcal{A}^i_{\text{grad}}$, which has expectation equal to Equation 7.4. The reasoning behind the gradients for the min player, calculated in line 9, is analogous: we sample a single j and return an appropriately scaled call to $\mathcal{A}^j_{\text{item}}$.

7.2.6 Approximation Guarantee

With these techniques in hand, our theoretical analysis shows that S-SP-MD ensures rapid convergence to an ϵ-optimal solution for Problem 7.2. This convergence property for the inner subroutine then, in turn, allows us to show that the overall strategy employed in Algorithm 11 attains the desired approximation guarantee. Formally, our main theoretical result is given by the following theorem. Here, $b = \max_{i,j} f_i(\{j\})$ is the maximum value of a single item.

Theorem 15. *Given a feasible set of target values $W_1 ... W_n$, Algorithm 11 outputs a set S such that $f_i(S) \geq (1-\epsilon)\left(1 - \frac{m}{k(1+\epsilon')\epsilon^3}\right)\left(1-\frac{1}{e}\right) W_i - \epsilon$ with probability at least $1-\delta$. Asymptotically as $k \to \infty$, the approximation ratio can be set to approach $1 - 1/e$ so long as $m = o(k \log^3 k)$. The algorithm requires $O(nm)$ ϵ'-accurate value oracle calls, $O(m\frac{bk^2}{\epsilon}\log\frac{1}{\delta})$ ϵ-accurate value oracle calls, $O\left(\frac{bk^4c^2}{\epsilon^5}\log\left(n + \frac{bk}{\delta\epsilon}\right)\right)$ calls to $\mathcal{A}_{grad}$ and $\mathcal{A}_{item}$, and $O\left(\frac{nk^2b^2}{\epsilon^2} + \frac{mk^2b}{\epsilon} + \frac{k^3b^2}{\epsilon^2}\right)$ additional work.*

This says that Algorithm 11 asymptotically converges to a $\left(1-\frac{1}{e}\right)$-approximation when the budget k is larger than the number of objectives m (i.e., the conditions under which the problem is approximable). All terms in the approximation ratio are identical to Udwani [Udw18], except that we improve their factor $\left(1-\frac{1}{e}\right)^2$ to $\left(1-\frac{1}{e}\right)$. The runtime is also identical apart from the time to solve the continuous problem (MultiFW vs their corresponding subroutine). This is difficult to compare since our respective algorithms use different oracles to access the functions. However, both kinds of oracles can typically be (approximately) implemented in time $O(n)$. Udwani's algorithm uses $O(n)$ oracle calls, while our's requires $O(bk^4c^2 \log n)$. For large-scale problems, n typically grows much faster than k, b, and c (all of which are often constants, or near-so). Hence, trading $O(n^2)$ runtime for $O(n \log n)$ can represent a substantial improvement. We present a more detailed discussion in the appendix.

7.2.7 Instantiation for Influence Maximization

To instantiate Algorithm 11 for influence maximization, we just need to supply appropriate stochastic gradient oracles. To our knowledge, no such oracles were previously known for influence maximization, which is substantially more complicated than other submodular problems because of additional randomness in the objective; naive extensions of previous methods require $O(n^2)$ time. We provide efficient $O(kn \log n)$ time stochastic gradient oracles by introducing a randomized method to simultaneously estimate many entries of the gradient at once. Details may be found in the appendix. The main idea is to use simulations of the influence process to estimate the marginal contribution that seeding each node would make towards the objective. Even to produce a noisy estimate, a naive method would require two simulations per node: one where the node is chosen as a seed and one where it is not. Since each simulation takes $O(n)$ runtime this requires $O(n^2)$ time overall. Our proposed method uses only $O(k \log n)$ simulations, but shares information across them in order to simultaneously estimate the marginal contribution made by all n nodes. .

7.3 Price of Fairness

In this section, we show that both definitions for the Price of Fairness can be unbounded; moreover, allowing nodes to join multiple groups can, counter-intuitively, worsen the PoF. The proofs in this section show undirected examples demonstrating the worst case. The results naturally serve as examples in a directed setting.

Theorem 16. *As $n \to \infty$ and $p \to 0$, there exists a family of graphs such that $PoF^{\text{Rational}} \to \infty$.*

Proof. We construct a graph G with two parts. In Part L, we have $s-1$ vertices all disjoint except for two vertices; label one of these $x3$. In Part S, we have a star with $s+1$ nodes. Label a leaf node x_1 and the central node x_2. We define two groups: C_1 is comprised of the s degree-1 vertices of S, and C_2 for the remaining s vertices, which includes the vertices of L and the central vertex x_2 of the star. There are $k = 2$ seeds, and since $|C_1| = |C_2|$, they each have a fair allocation of $k_1 = k_2 = 1$ seeds. Since the subgraph induced by C_1 is comprised of isolated vertices, they have a rational allocation of $\mathcal{I}_{G[C_1]}(1) = 1$. The subgraph induced by C_2 is a collection of isolated vertices and a K_2, its rational allocation is $\mathcal{I}_{G[C_2]}(1) = 1 + p$.

We are interested in two seeding configurations: $A = \{x_1, x_3\}$ and $B = \{x_2, x_3\}$. We can verify that configuration A is fair. The A activates $1 + p$ nodes in Part L, and $1 + p + (s-1)p^2$ in Part S, for a total of $\mathcal{I}_G(A) = 2 + 2p + (s-1)p^2$.

Now consider configuration B. C_1 receives ps influence, and since $p < \frac{2}{n} = \frac{1}{s}$, C_1 does not receive its group rational share of influence. However, we can verify that this seeding is optimal. Part L receives $(1 + p)$ influence, and Part S receives $1 + ps$. Therefore, $\mathcal{I}_G(B) = 2 + p + ps$.

We may then calculate our Price of Fairness:

$$PoF^{\text{Rational}} = \frac{\mathcal{I}_G^{\text{OPT}}}{\mathcal{I}_G^{\text{Rational}}} = \frac{2+p+ps}{2+2p+(s-1)p^2}$$

And if we take the limit as $n \to \infty$, $s \to \infty$, $PoF \to 1/p$. Finally, as as $p \to 0$, $PoF \to \infty$. □

The appendix details a similar result for Maximin Fairness:

Theorem 17. *As $n \to \infty$ and $p \to 0$, there exists a family of graphs such that $PoF^{\text{Maximin}} \to \infty$.*

Frequently, an individual may identify with multiple groups. Intuitively, we might expect such multi-group membership to improve the influence received by different groups and make group-fairness easier to achieve (see the appendix for an example). However, in the following, we show that this is not always true — giving even a single node membership in a second group can cause the Price of Fairness to worsen by an arbitrarily large amount.

Theorem 18. *Let G be a graph with groups C_1 and C_2, and G' with groups C_1' and C_2', where $G' = G$, $C_1' = C_1$ and C_2' is obtained from C_2 by the addition of one vertex x_1 ($x_1 \in C_1$, $x_1 \notin C_2$). There exists a family of such graphs such that $\lim_{n\to\infty} \frac{PoF_{G'}^{\text{Rational}}}{PoF_G^{\text{Rational}}} = \infty$.*

Proof. Consider a graph G with two components: one component K contains 2 vertices joint by an edge, the other component S is a star with $s+1$ vertices ($s \geq 1/p$). There are two groups: C_1 contains all degree-1 vertices from S and one vertex from K; C_2 contains the other vertex x_1 from K and the central vertex x_2 from S. There is one seed ($k = 1$), and the fair allocation of seeds to each group is $k_1 = k_2 = 1$.

Since the induced subgraphs for both groups comprise only of isolated nodes, the group rational influence for each group is $\mathcal{I}_{G[C_1]} = \mathcal{I}_{G[C_2]} = 1$. Therefore, the seed set $\{x_2\}$ is both fair and optimal, giving an expected influence of $\mathcal{I}_G(\{x_2\}) = 1 + ps$.

Now, let us modify G by letting x_1 belong to *both* communities to obtain G', and communities C_1' and C_2'. The group rational influence for C_2' remains the same (its members have not changed) but $\mathcal{I}_{G'[C_1']}$ has increased to $1 + p$ (by seeding x_1). In fact, this forces the fair allocation to seed x_1 instead of x_2, for a fair influence of $\mathcal{I}_{G'}(\{x_1\}) = 1 + p$.

As $n \to \infty$, $\lim_{n\to\infty} \frac{PoF_{G'}^{\text{Rational}}}{PoF_G^{\text{Rational}}} = \lim_{s\to\infty} \frac{1+ps}{1+p} = \infty$. □

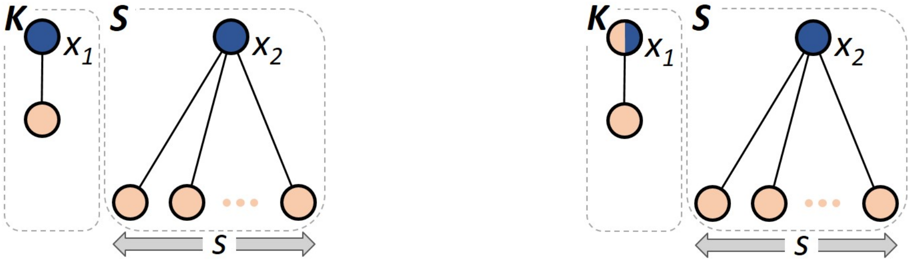


Figure 7.1: *Left: G with Disjoint Groups. Right: G′ with Overlapping Groups.*

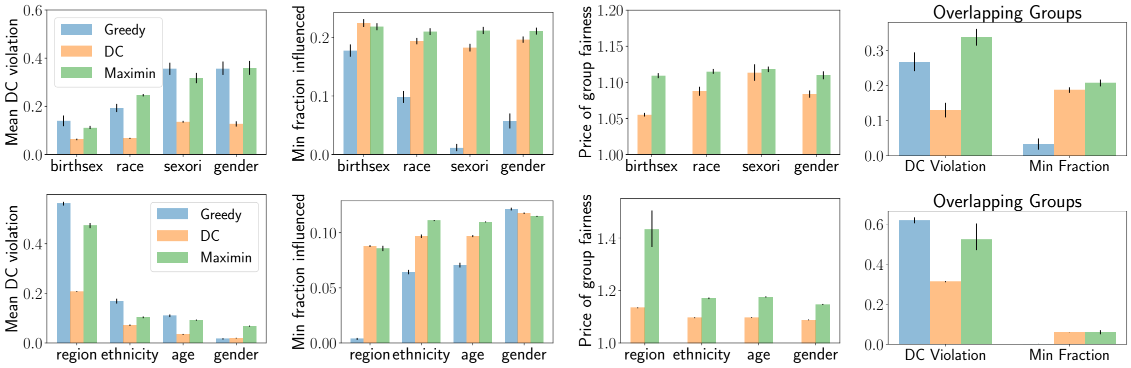


Figure 7.2: *Average performance on homeless youth social networks (top) and simulated Antelope Valley networks (bottom).*

A slightly weaker result can be obtained for Maximin Fairness where the construction of the graphs depend on p. The proof is provided in the appendix.

Theorem 19. *Let G be a graph with groups C_1 and C_2, and G' with groups C_1' and C_2', where $G' = G$, $C_1' = C_1$ and C_2' is obtained from C_2 by the addition of one vertex x_1 ($x_1 \in C_1$, $x_1 \notin C_2$). Given propagation probability p, we may construct a family of such graphs such that $\lim_{n\to\infty} \frac{PoF^{\text{Maximin}}_{G'}}{PoF^{\text{Maximin}}_{G}} \to \infty$.*

7.4 Experimental Results

We now investigate the empirical impact of considering fairness in influence maximization. We start with experiments on a set of four real-world social networks which have been previously used for a socially critical application: HIV prevention for homeless youth. Each network has 60-70 nodes, and represents the real-world social connections between

Table 7.1: *Network characteristics.*

Characteristic	Net. 1	Net. 2	Net. 3	Net. 4
Density	0.012	0.032	0.022	0.034
Modularity	0.803	0.713	0.604	0.537
Median group size	13.0	9.5	16.0	9.5

a set of homeless youth surveyed in a major US city. Each node in the network is associated with demographic information: their birth sex, gender identity, race, and sexual orientation. The networks can be made available upon request; all code is available at `https://github.com/bwilder0/fair_influmax_code_release`. Table 7.1 gives some aggregate statistics for each network. Each demographic attribute gives a partition of the network into anywhere from 2 to 6 different groups. For each partition, we compare three algorithms: the standard greedy algorithm for influence maximization, which maximizes the total expected influence (Greedy), Algorithm 11 used to enforce diversity constraints (DC), and Algorithm 11 used to find a maximin fair solution (Maximin). We set the propagation probability to be $p = 0.1$ and fixed $k = 15$ seeds (varying these parameters had little impact). We average over 30 runs of the algorithms on each network (since all of the algorithms use random simulations of influence propagation), with error bars giving bootstrapped 95% confidence intervals.

Figure 7.2 (top) shows that the choice of solution concept has a substantial impact on the results. For the diversity constraints case, we summarize the performance of each algorithm by the mean percentage violation of the constraints over all groups. For the maximin case, we directly report the minimum fraction influenced over all groups. We see that greedy generates substantial unfairness according to either metric: it generates the highest violations of diversity constraints, and has the smallest minimum fraction influenced. Greedy actually obtains near-zero maximin value with respect to sexual orientation. This results from it assigning one seed to a minority group in a single run and zero in others.

DC performs well across the board: it reduces constraint violations by approximately 55-65% while also performing competitively with respect to the maximin metric (even without explicitly optimizing for it). As expected, the Maximin algorithm generally obtains the best maximin value. DC actually attains slightly better maximin value for one attribute (birthsex); however, the difference is within the confidence intervals and reflects slight fluctuations in the approximation quality of the algorithms. However, Maximin performs surprisingly poorly with respect to diversity constraint violations. This indicates that optimizing exclusively for equal influence spread may force the algorithm to focus on poorly connected groups which exhibit severe diminishing returns. DC is able to attain almost as much influence in such groups but is then permitted to focus its remaining budget for higher impact. Interestingly, the price of fairness is relatively small for both solution concepts, in the range 1.05-1.15 (though it is higher for maximin than for DC). This indicates that while standard influence maximization techniques can introduce substantial fairness violations, mitigating such violations may be substantially less costly in real world networks than the theoretical worst case would suggest.

Finally, the rightmost plot in the top row of Figure 7.2 explores an example with overlapping groups. Specifically, we consider the race and birthsex attributes so that each node belongs to two groups. Constraint violations are somewhat higher than for either attribute individually, but the price of fairness remains small (1.07 for DC and 1.13 for Maximin).

In Figure 7.2 (bottom), we examine 20 synthetic networks used by [WOdlHT18] to model an obesity prevention intervention in the Antelope Valley region of California. Each node in the network has a geographic region, ethnicity, age, and gender, and nodes are more likely to connect to those with similar attributes. Each network has 500 nodes and we set $k = 25$. Overall the results are similar to the homeless youth networks. One exception is the high price of fairness that maximin suffers with respect to the “region" attribute (over 1.4), but the other PoF values are relatively low (below 1.2). We also observe that greedy obtains the (slightly) best maximin performance for gender, likely because the network is sufficiently well-mixed across genders that fairness is not a significant concern (as confirmed by the extremely low DC violations). Absent true fairness concerns, greedy may perform slightly better since it solves a simpler optimization problem. However, in the last figure, we examine overlapping groups given by region and ethnicity and observe that greedy actually obtains zero maximin value, indicating that there is one group that it never reached across any run.

7.5 Conclusions

In this chapter, we examine the problem of selecting key figures in a population to ensure the fair spread of vital information across all groups. This problem modifies the classic influence maximization problem with additional fairness provisions based on legal and game theoretic concepts. We examine two methods for determining these provisions, and show that the "Price of Fairness" for these provisions can be unbounded. We propose an improved algorithm for multiobjective maximization to examine this problem on real world data sets. We show that standard influence maximization techniques often neglect smaller groups, and a diversity constraint based algorithm can ensure these groups receive a fair allocation of resources at relatively little cost. As automated techniques become increasingly prevalent in society and governance, our technique will help ensure that small and marginalized groups are fairly treated.

Part III

Learning and decisions

Chapter 8

Melding the data-decisions pipeline for discrete optimization

The goal in many real-world applications of artificial intelligence is to create a pipeline from data, to predictive models, to decisions. Together, these steps enable a form of evidence-based decision making which has transformative potential across domains such as healthcare, scientific discovery, transportation, and more [HM10, Hor10]. This pipeline requires two technical components: machine learning models and optimization algorithms. Machine learning models use the data to predict unknown quantities; optimization algorithms use these predictions to arrive at a decision which maximizes some objective. Our concern here is combinatorial optimization, which is ubiquitous in real-world applications of artificial intelligence, ranging from matching applicants to public housing to selecting a subset of movies to recommend. We focus on common classes of combinatorial problems which have well-structured continuous relaxations, e.g., linear programs and submodular maximization. A vast literature has been devoted to combinatorial optimization [KVKV12]. Importantly though, optimization is often insufficient without the broader pipeline because the objective function is unknown and must predicted via machine learning.

While machine learning has witnessed incredible growth in recent years, the two pieces of the pipeline are treated entirely separately by typical training approaches. That is, a system designer will first train a predictive model using some standard measure of accuracy, e.g., mean squared error for a regression problem. Then, the model's predictions are given as input to the optimization algorithm to produce a decision. Such *two-stage* approaches are extremely common across many domains [WXQ^{+}06, FNP^{+}16, MVDB17, XDF^{+}16]. This process is justified when the predictive model is perfect, or near-so, since completely accurate predictions also produce the best decisions. However, in complex learning tasks, all models will make errors and the training process implicitly trades off where these errors will occur. When prediction and optimization are separate, this tradeoff is divorced from the goal of the broader pipeline: to make the best decision possible.

We propose a *decision-focused learning* framework which melds the data-decisions pipeline by integrating prediction and optimization into a single end-to-end system. That is, the predictive model is trained using the quality of the decisions which it induces via the optimization algorithm. Similar ideas have recently been explored in the context of convex optimization [DAK17], but to our knowledge ours is the first attempt to train machine learning systems for performance on *combinatorial* decision-making problems. Combinatorial settings raise new technical challenges because the optimization problem is discrete. However, machine learning systems (e.g., deep neural networks) are often trained via gradient descent.

Our first contribution is a general framework for training machine learning models via their performance on combinatorial problems. The starting point is to relax the combinatorial problem to a continuous one. Then, we analytically differentiate the optimal solution to the continuous problem as a function of the model's predictions. This allows us to train using a continuous proxy for the discrete problem. At test time, we round the continuous solution to a discrete point.

Our second contribution is to instantiate this framework for two broad classes of combinatorial problems: linear programs and submodular maximization problems. Linear programming encapsulates a number of classical problems such as shortest path, maximum flow, and bipartite matching. Submodular maximization, which reflects the intuitive phenomena of diminishing returns, is also ubiquitous; applications range from social networks [KKT03] to recommendation systems [VB10]. In each case, we resolve a set of technical challenges to produce well-structured relaxations which can be efficiently differentiated through.

Finally, we give an extensive empirical investigation, comparing decision-focused and traditional methods on a series of domains. Decision-focused methods often improve performance for the pipeline as a whole (i.e., decision quality) despite worse predictive accuracy according to standard measures. Intuitively, the predictive models trained via our approach focus specifically on qualities which are important for making good decisions. By contrast, more generic methods produce predictions where error is distributed in ways which are not aligned with the underlying task.

8.1 Problem description

We consider combinatorial optimization problems of the form $\max_{x \in \mathcal{X}} f(x, \theta)$, where $\mathcal{X}$ is a discrete set enumerating the feasible decisions. Without loss of generality, $\mathcal{X} \subseteq \{0,1\}^n$ and the decision variable x is a binary vector. The objective f depends on a parameter $\theta \in \Theta$. If θ were known exactly, a wide range of existing techniques could be used to solve the problem. In this chapter, we consider the challenging (but prevalent) case where θ is unknown and must be inferred from data. For instance, in bipartite matching, x represents whether each pair of nodes were matched and θ contains the reward for matching each pair. In many applications, these affinities are learned from historical data.

Specifically, the decision maker observes a feature vector $y \in \mathcal{Y}$ which is correlated with θ. This introduces a learning problem which must be solved prior to optimization. As in classical supervised learning, we formally model y and θ as drawn from a joint distribution P. Our algorithm will observe training instances $(y_1, \theta_1)...(y_N, \theta_N)$ drawn iid from P. At test time, we are give a feature vector y corresponding to an *unobserved* θ. Our algorithm will use y to predict a parameter value $\hat{\theta}$. Then, we will solve the optimization problem $\max_x f(x, \hat{\theta})$ to obtain a decision x^*. Our utility is the objective value that x^* obtains with respect to the *true but unknown* parameter θ, $f(x^*, \theta)$.

Let $m : \mathcal{Y} \to \Theta$ denote a model mapping observed features to parameters. Our goal is to (using the training data) find a model m which maximizes expected performance on the underlying optimization task. Define $x^*(\theta) = \arg\max_{x \in \mathcal{X}} f(x, \theta)$ to be the optimal x for a given θ. The end goal of the data-decisions pipeline is to maximize

$$\mathop{\mathbb{E}}_{y,\theta \sim P} \left[f(x^*(m(y)), \theta) \right] \tag{8.1}$$

The classical approach to this problem is a *two-stage* method which first learns a model using a task-agnostic loss function (e.g., mean squared error) and then uses the learned model to solve the optimization problem. The model class will have its own parameterization, which we denote by $m(y, \omega)$. For instance, the model class could consist of deep neural networks where ω denotes the weights. The two-stage approach first solves the problem $\min_\omega \mathbb{E}_{y,\theta \sim P} \left[\mathcal{L}(\theta, m(y, \omega)) \right]$, where $\mathcal{L}$ is a loss function. Such a loss function measures the overall “accuracy" of the model's predictions but does not specifically consider how m will fare when used for decision making. The question we address is whether it is possible to do better by specifically training the model to perform well on the decision problem.

8.2 Previous work

There is a growing body of research at the interface of machine learning and discrete optimization [VFJ15, BD17, KDN$^+$17, KDZ$^+$17]. However, previous work largely focuses on either using discrete optimization to find an accuracy-maximizing predictive model or using machine learning to speed up optimization algorithms. Here, we pursue a deeper synthesis; to our knowledge, this work is the first to train predictive models using combinatorial optimization performance with the goal of improving decision making.

The closest work to ours in motivation is [DAK17], who study task-based convex optimization. Their aim is to optimize a convex function which depends on a learned parameter. As in their work, we use the idea of differentiating through the KKT conditions.

However, their focus is entirely on continuous problems. Our discrete setting raises new technical challenges, highlighted below. Elmachtoub and Grigas [EG17] also propose a means of integrating prediction and optimization; however, their method applies strictly to linear optimization and focuses on linear predictive models while our framework applies to nonlinear problems with more general models (e.g., neural networks). Finally, some work has noted that two-stage methods lead to poor optimization performance in specific domains [BL09, FNT$^+$15].

Our work is also related to recent research in structured prediction [BYM17, TG18, NMBC18, DK17]. which aims to make a prediction lying in a discrete set. This is fundamentally different than our setting since their goal is to *predict* an external quantity, not to *optimize* and find the best decision possible. However, structured prediction sometimes integrates a discrete optimization problem as a module within a larger neural network. The closest such work technically to ours is [TSK18], who design a differentiable algorithm for submodular maximization in order to predict choices made by users. Their approach is to introduce noise into the standard greedy algorithm, making the probability of outputting a given set differentiable. There are two key differences between our approaches. First, their approach does not apply to the decision-focused setting because it maximizes the likelihood of a *fixed* set but cannot optimize for finding the best set. Second, exactly computing gradients for their algorithm requires marginalizing over the $k!$ possible permutations of the items, forcing a heuristic approximation to the gradient. Our approach allows closed-form differentiation.

Some deep learning architectures differentiate through gradient descent steps, related to our approach in the submodular setting. Typically, previous approaches explicitly unroll T iterations of gradient descent in the computational graph [Dom12]. However, this approach is usually employed for *unconstrained* problems where each iteration is a simple gradient step. By contrast, our combinatorial problems are constrained, requiring a projection step to enforce feasibility. Unrolling the projection step may be difficult, and would incur a large computational cost. We instead exploit the fact that gradient ascent converges to a local optimum and analytically differentiate via the KKT conditions.

8.3 General framework

Our goal is to integrate combinatorial optimization into the loop of gradient-based training. That is, we aim to directly train the predictive model m by running gradient steps on the objective in Equation 8.1, which integrates both prediction and optimization. The immediate difficulty is the dependence on $x^*(m(y, \omega))$. This term is problematic for two reasons. First, it is a discrete quantity since x^* is a decision from a binary set. This immediately renders the output nondifferentiable with respect to the model parameters ω. Second, even if x^* were continuous, it is still defined as the solution to an optimization problem, so calculating a gradient requires us to differentiate through the argmax operation.

We resolve both difficulties by considering a continuous relaxation of the combinatorial decision problem. We show that for a broad class of combinatorial problems, there are appropriate continuous relaxations such that we can analytically obtain derivatives of the continuous optimizer with respect to the model parameters. This allows us to train any differentiable predictive model via gradient descent on a continuous surrogate to Equation 8.1. At test time, we solve the true discrete problem by rounding the continuous point.

More specifically, we relax the discrete constraint $x \in \mathcal{X}$ to the continuous one $x \in conv(\mathcal{X})$ where $conv$ denotes the convex hull. Let $x(\theta) = \arg\max_{x \in conv(\mathcal{X})} f(x, \theta)$ denote the optimal solution to the continuous problem. To train our predictive model, we would like to compute gradients of the whole-pipeline objective given by Equation 8.1, replacing the discrete quantity x^* with the continuous x. We can obtain a stochastic gradient estimate by sampling a single (y, θ) from the training data. On this sample, the chain rule gives

$$\frac{df(x(\hat{\theta}), \theta)}{d\omega} = \frac{df(x(\hat{\theta}), \theta)}{dx(\hat{\theta})} \frac{dx(\hat{\theta})}{d\hat{\theta}} \frac{d\hat{\theta}}{d\omega}$$

The first term is just the gradient of the objective with respect to the decision variable x, and the last term is the gradient of the model's predictions with respect to its own internal parameterization.

The key is computing the middle term, which measures how the optimal decision changes with respect to the prediction $\hat{\theta}$. For continuous problems, the optimal continuous decision x must satisfy the KKT conditions (which are sufficient for convex problems). The KKT conditions define a system of linear equations based on the gradients of the objective and constraints around the optimal point. Is is known that by applying the implicit function theorem, we can differentiate the solution to this linear system [GFC$^+$16, DAK17]. In more detail, recall that our continuous problem is over $conv(\mathcal{X})$, the convex hull of the discrete feasible solutions. This set is a polytope, which can be represented via linear equalities as the set $\{x : Ax \leq b\}$ for some matrix A and vector b. Let (x, λ) be pair of primal and dual variables which satisfy the KKT conditions. Then differentiating the conditions yields that

$$\begin{bmatrix} \nabla_x^2 f(x,\theta) & A^T \\ diag(\lambda)A & diag(Ax-b) \end{bmatrix} \begin{bmatrix} \frac{dx}{d\theta} \\ \frac{d\lambda}{d\theta} \end{bmatrix} = \begin{bmatrix} \frac{d\nabla_x f(x,\theta)}{d\theta} \\ 0 \end{bmatrix} \tag{8.2}$$

By solving this system of linear equations, we can obtain the desired term $\frac{dx}{d\theta}$. However, the above approach is a general framework; our main technical contribution is to instantiate it for specific classes of combinatorial problems. Specifically, we need (1) an appropriate continuous relaxation, along with a means of solving the continuous optimization problem and (2) efficient access to the terms in Equation 8.2 which are needed for the backward pass (i.e., gradient computation). We provide both ingredients for two broad classes of problems: linear programming and submodular maximization. In each setting, the high-level challenge is to ensure that the continuous relaxation is differentiable, a feature not satisfied by naive alternatives. We also show how to efficiently compute terms needed for the backward pass, especially for the more intricate submodular case.

8.3.1 Linear programming

The first setting that we consider is combinatorial problems which can be expressed as a linear program with equality and inequality constraints in the form

$$\max \theta^T x \;\text{ s.t. }\; Ax = b,\; Gx \leq h \tag{8.3}$$

Example problems include shortest path, maximum flow, bipartite matching, and a range of other domains. For instance, in a shortest path problem θ contains the cost for traversing each edge, and we are interested in problems where the true costs are unknown and must be predicted. Since the LP can be regarded as a continuous problem (it just happens that the optimal solutions in these example domains are integral), we could attempt to apply Equation 8.2 and differentiate the solution. This approach runs into an immediate difficulty: the optimal solution to an LP may not be differentiable (or even continuous) with respect to θ. This is because the optimal solution may “jump" to a different vertex. Formally, the left-hand side matrix in Equation 8.2 becomes singular since $\nabla_x^2 f(x, \theta)$ is always zero. We resolve this challenge by instead solving the regularized problem

$$\max \theta^T x - \gamma ||x||_2^2 \;\text{ s.t. }\; Ax = b,\; Gx \leq h \tag{8.4}$$

which introduces a penalty proportional to the squared norm of the decision vector. This transforms the LP into a strongly concave quadratic program (QP). The Hessian is given by $\nabla_x^2 f(x, \theta) = -2\gamma I$ (where I is the identity matrix), which renders the solution differentiable under mild conditions:

Theorem 20. *Let $x(\theta)$ denote the optimal solution of Problem 8.4. Provided that the problem is feasible and all rows of A are linearly independent, $x(\theta)$ is differentiable with respect to θ almost everywhere. If A has linearly dependent rows, removing these rows yields an equivalent problem which is differentiable almost everywhere. Wherever $x(\theta)$ is differentiable, it satisfies the conditions in Equation 8.2.*

Proof. We start with the case where all rows of A are linearly independent. Here, the result follows easily from Theorem 1 of [AK17] since the Hessian matrix is γI and hence guaranteed to be positive definite.

When A has linearly dependent rows, we argue that these rows can be removed without changing the feasible region. Consider two rows a_i and a_j such that for all x, $a_i^\top x = ca_j^\top x$ for some scalar c. We are guaranteed that the problem is feasible, meaning that there exists an x which satisfies both constraints simultaneously. For this x, we have $a_i^\top x = b_i$ and $a_j^\top x = b_j$. But since $a_i^\top x = ca_j^\top x$, we must have $b_i = cb_j$. Accordingly, constraint i is satisfied if and only if constraint j is satisfied, and so removing one of the constraints leaves the feasible set unchanged. Applying this argument inductively yields the theorem. □

Moreover, we can control the loss that regularization can cause on the original, linear problem:

Theorem 21. *Define $D = \max_{x,y \in conv(\mathcal{X})} ||x - y||^2$ as the squared diameter of the feasible set and OPT to be the optimal value for Problem 8.3. We have $\theta^\top x(\theta) \geq OPT - \gamma D$.*

Proof. Let $x_{max} = \arg\max_{y \in conv(\mathcal{X})} ||y||^2$. We have that

$$\begin{aligned}
\theta^\top x(\theta) &= \max_y \left[\theta^\top y - \gamma ||y||^2\right] + ||x(\theta)||^2 \\
&\geq \max_y \left[\theta^\top y\right] - \gamma ||x_{max}||^2 + \gamma ||x(\theta)||^2 \\
&= \max_y \left[\theta^\top y\right] + \gamma \left(||x(\theta)||^2 - ||x_{max}||^2\right) \\
&\geq OPT - \gamma ||x(\theta) - x_{max}||^2 \\
&\geq OPT - \gamma D
\end{aligned}$$

where the second inequality uses the reverse triangle inequality. □

Together, these results give us a differentiable surrogate which still enjoys an approximation guarantee relative to the integral problem. Computing the backward pass via Equation 8.2 is now straightforward since all the relevant terms are easily available. Since $\nabla_x \theta^\top x = \theta$, we have $\frac{d\nabla_x f(x,\theta)}{d\theta} = I$. All other terms are easily computed from the optimal primal-dual pair (x, λ) which is output by standard QP solvers. We can also leverage a recent QP solver [AK17] which maintains a factorization of the KKT matrix for a faster backward pass. At test time, we simply set $\gamma = 0$ to produce an integral decision.

8.3.2 Submodular maximization

We consider problems where the underlying objective to maximize a set function $f : 2^V \to R$, where V is a ground set of items. A set function is *submodular* if for any $A \subseteq B$ and any $v \in V \setminus B$, $f(A \cup \{v\}) - f(A) \geq f(B \cup \{v\}) - f(B)$. We will restrict our consideration to submodular functions which are *monotone* ($f(A \cup \{v\}) - f(A) \geq 0 \; \forall A, v$) and *normalized* $f(\emptyset) = 0$. This class of functions contains many combinatorial problems which have been considered in machine learning and artificial intelligence (e.g., influence maximization, facility location, diverse subset selection, etc.). We focus on the cardinality-constrained optimization problem $\max_{|S| \leq k} f(S)$, though our framework easily accommodates more general matroid constraints.

Continuous relaxation: We employ the canonical continuous relaxation for submodular set functions, which associates each set function f with its *multilinear extension* F [CCPV11]. We can view a set function as defined on the domain $\{0,1\}^{|V|}$, where each element is an indicator vector which the items contained in the set. The extension F is a continuous function defined on the hypercube $[0,1]^{|V|}$. We interpret a given fraction vector $x \in [0,1]^{|V|}$ as giving the marginal probability that each item is included in the set. $F(x)$ is the expected value of $f(S)$ when each item i is included in S independently with probability x_i. In other words, $F(x) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} 1 - x_i$. While this definition sums over exponentially many terms, arbitrarily close approximations can be obtained via random sampling. Further, closed forms are available for many cases of interest [IJB14]. Importantly, well-known rounding algorithms [CCPV11] can convert a fractional point x to a set S satisfying $\mathbb{E}[f(S)] \geq F(x)$; i.e., the rounding is lossless.

As a proxy for the discrete problem $\max_{|S| \leq k} f(S)$, we can instead solve $\max_{x \in conv(\mathcal{X})} F(x)$, where $\mathcal{X} = \{x \in \{0,1\}^{|V|} : \sum_i x_i \leq k\}$. Unfortunately, F is not in general concave. Nevertheless, many first-order algorithms still obtain a constant factor approximation. For instance, a variant of the Frank-Wolfe algorithm solves the continuous maximization problem with the optimal approximation ratio of $(1 - 1/e)$ [CCPV11, BMBK17].

However, non-concavity complicates the problem of differentiating through the continuous optimization problem. Any polynomial-time algorithm can only be guaranteed to output a *local* optimum, which need not be unique (compared to strongly convex problems, where there is a single global optimum). Consequently, the algorithm used to select $x(\theta)$ might return a *different* local optimum under an infinitesimal change to θ. For instance, the Frank-Wolfe algorithm (the most common algorithm for continuous submodular maximization) solves a linear optimization problem at each step. Since (as noted above), the solution to a linear problem may be discontinuous in θ, this could render the output of the optimization problem nondifferentiable.

We resolve this difficulty through a careful choice of optimization algorithm for the forward pass. Specifically, we use apply projected stochastic gradient ascent (SGA), which has recently been shown to obtain a $\frac{1}{2}$-approximation for continuous submodular maximization [HSK17]. Although SGA is only guaranteed to find a local optimum, each iteration applies purely differentiable computations (a gradient step and projection onto the set $conv(\mathcal{X})$), and so the final output after T iterations will be differentiable as well. Provided that T is sufficiently large, this output will converge to a local optimum, which must satisfy the KKT conditions. Hence, we can apply our general approach to the local optimum returned by SGA. The following theorem shows that the local optima of the multilinear extension are differentiable:

Theorem 22. *Suppose that x^* is a local maximum of the multilinear extension, i.e.,., $\nabla_x F(x^*, \theta) = 0$ and $\nabla_x^2 F(x^*, \theta) \succ 0$. Then, there exists a neighborhood $\mathcal{I}$ around x^* such that the maximizer of $F(\cdot, \theta)$ within $\mathcal{I} \cap conv(\mathcal{X})$ is differentiable almost everywhere as a function of θ, with $\frac{dx(\theta)}{d\theta}$ satisfying the conditions in Equation 8.2.*

Proof. Since $\mathcal{X} = \{x \in \{0,1\}^{|V|} : \sum_i x_i \leq k\}$, $conv(\mathcal{X})$ is described by the two inequality constraints $-Ix \leq 0$ and $1^\top x \leq k$. It is easy to see that the corresponding constraint matrix A has full row rank. Even though F is not concave, any stationary point (x, λ) must satisfy the KKT conditions. By applying the implicit function theorem to differentiate these equations, we get the form

$$\begin{bmatrix} \nabla_x^2 F(x,\theta) & A^T \\ diag(\lambda)A & diag(Ax-b) \end{bmatrix} \begin{bmatrix} \frac{dx}{d\theta} \\ \frac{d\lambda}{d\theta} \end{bmatrix} = \begin{bmatrix} \frac{d\nabla_x f(x,\theta)}{d\theta} \\ 0 \end{bmatrix}$$

So long as the right hand side matrix is invertible almost everywhere, the implicit function theorem guarantees that $\frac{dx}{d\theta}$ exists in a neighborhood of x and satisfies the above conditions. Note that at a local maximum, we have $\nabla_x^2 F(x,\theta) \succ 0$, implying that the Hessian matrix must be invertible. Accordingly, it is easy to show that the RHS matrix is nonsingular by applying the same logic as [AK17] (Theorem 1). □

We remark that Theorem 22 requires a local maximum, while gradient ascent may in theory find saddle points. However, recent work shows that random perturbations ensure that gradient ascent quickly escapes saddle points and finds an approximate local optimum [JGN$^+$17].

Efficient backward pass: We now show how the terms needed to compute gradients via Equation 8.2 can be efficiently obtained. In particular, we need access to the optimal dual variable λ as well as the term $\frac{d\nabla_x F(x,\theta)}{d\theta}$. These were easy to obtain in the LP setting but the submodular setting requires some additional analysis. Nevertheless, we show that both can be obtained efficiently.

Optimal dual variables: SGA only produces the optimal primal variable x, not the corresponding dual variable λ which is required to solve Equation 8.2 in the backward pass. We show that for cardinality-constrained problems, we can obtain the optimal dual variables analytically given a primal solution x. Let λ_i^L be the dual variable associated with the constraint $x_i \geq 0$, λ_i^U with $x_i \leq 1$ and λ^S with $\sum_i x_i \leq k$. By differentiating the Lagrangian, any optimum satisfies

$$\nabla_{x_i} f(x) - \lambda_i^L + \lambda_i^U + \lambda_i^S = 0 \quad \forall i$$

where complementary slackness requires that $\lambda_i^L = 0$ if $x_i > 0$ and $\lambda_i^U = 0$ if $x_i < 1$.

Further, it is easy to see that for all i with $0 < x_i < 1$, $\nabla_{x_i} f(x)$ must be equal. Otherwise, x could not be (locally) optimal since we could increase the objective by finding a pair i, j with $\nabla_{x_i} f(x) > \nabla_{x_j} f(x)$, increasing x_i, and decreasing x_j. Let ∇_* denote the shared gradient value for fractional entries. We can solve the above equation and express the optimal dual variables as

$$\lambda^S = -\nabla_*, \;\; \lambda_i^L = \lambda^S - \nabla_{x_i} f, \;\; \lambda_i^U = \nabla_{x_i} f - \lambda^S$$

where the expressions for λ_i^L and λ_i^U apply only when $x_i = 0$ and $x_i = 1$ respectively (otherwise, complementary slackness requires these variables be set to 0).

Computing $\frac{\mathbf{d}}{\mathbf{d}\text{`}}\nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x},\text{`})$: We show that this term can be obtained in closed form for the case of probabilistic coverage functions, which includes many cases of practical interest (e.g. budget allocation, sensor placement, facility location, etc.). However, our framework can be applied to arbitrary submodular functions; we focus here on coverage functions just because they are particularly common in applications. A coverage function takes the following form. There a set of items U, and each $j \in U$ has a weight w_j. The algorithm can choose from a ground set V of actions. Each action a_i covers each item j independently with probability θ_{ij}. We consider the case where the probabilities θ are be unknown and must be predicted from data. For such problems, the multilinear extension has a closed form

$$F(x, \theta) = \sum_{j \in U} w_j \left(1 - \prod_{i \in V} 1 - x_{ij}\theta_{ij}\right)$$

and we can obtain the expression

$$\frac{d}{d\theta_{kj}} \nabla_{x_i} F(x, \theta) = \begin{cases} -\theta_{ij} x_k \prod_{\ell \neq i,k} 1 - x_\ell \theta_{\ell j} & \text{if } k \neq i \\ \prod_{k \neq i} 1 - x_k \theta_{kj} & \text{otherwise.} \end{cases}$$

Table 8.1: *Solution quality of each method for the full data-decisions pipeline.*

	Budget allocation			Matching	Diverse recommendation		
$k =$	5	10	20	–	5	10	20
NN1-Decision	**49.18 ± 0.24**	**72.62 ± 0.33**	**98.95 ± 0.46**	2.50 ± 0.56	**15.81 ± 0.50**	**29.81 ± 0.85**	**52.43 ± 1.23**
NN2-Decision	44.35 ± 0.56	67.64 ± 0.62	93.59 ± 0.77	**6.15 ± 0.38**	13.34 ± 0.77	26.32 ± 1.38	47.79 ± 1.96
NN1-2Stage	32.13 ± 2.47	45.63 ± 3.76	61.88 ± 4.10	2.99 ± 0.76	4.08 ± 0.16	8.42 ± 0.29	19.16 ± 0.57
NN2-2Stage	9.69 ± 0.05	18.93 ± 0.10	36.16 ± 0.18	3.49 ± 0.32	11.63 ± 0.43	22.79 ± 0.66	42.37 ± 1.02
RF-2Stage	**48.81 ± 0.32**	**72.40 ± 0.43**	**98.82 ± 0.63**	3.66 ± 0.26	7.71 ± 0.18	15.73 ± 0.34	31.25 ± 0.64
Random	9.69 ± 0.04	18.92 ± 0.09	36.13 ± 0.14	2.45 ± 0.64	8.19 ± 0.19	16.15 ± 0.35	31.68 ± 0.71

8.4 Experiments

We conduct experiments across a variety of domains in order to compare our decision-focused learning approach with traditional two stage methods. We start out by describing the experimental setup for each domain. Then, we present results for the complete data-decisions pipeline in each domain (i.e., the final solution quality each method produces on the optimization problem). We find that decision-focused learning almost always outperforms two stage approaches. To investigate this phenomenon, we show more detailed results about what each model learns. Two stage approaches typically learn predictive models which are more accurate according to standard measures of machine learning accuracy. However, decision-focused methods learn qualities which are important for optimization performance even if this leads to lower accuracy in an overall sense.

Budget allocation: We start with a synthetic domain which allows us to illustrate how our methods differ from traditional approaches and explore when improved decision making is achievable. This example concerns budget allocation, a submodular maximization problem which models an advertiser's choice of how to divide a finite budget k between a set of channels. There is a set of customers R and the objective is $f(S) = \sum_{v \in R} 1 - \prod_{u \in S}(1 - \theta_{uv})$, where θ_{uv} is the probability that advertising on channel u will reach customer v. This is the expected number of customers reached. Variants on this problem have been the subject of a great deal of research [AGT12, SKIK14, MIFK15].

In our problem, the matrix θ is not known in advance and must be learned from data. The ground truth matrices were generated using the Yahoo webscope [Yah07] dataset which logs bids placed by advertisers on a set of phrases. In our problem, the phrases are channels and the accounts are customers. Each instance samples a random subset of 100 channels and 500 customers. For each edge (u, v) present in the dataset, we sample θ_{uv} uniformly at random in [0,0.2]. For each channel u, we generate a feature vector from that channel's row of the matrix, θ_u via complex nonlinear function. Specifically, θ_u is passed through a 5-layer neural network with random weight matrices and ReLU activations to obtain a feature vector y_u. The learning task is to reconstruct θ_u from y_u. The optimization task is to select k channels in order to maximize the number of customers reached.

Bipartite matching: This problem occurs in many domains; e.g., bipartite matching has been used to model the problem of a public housing programs matching housing resources to applicants [BCH+18] or platforms matching advertisers with users [BK07]. In each of these cases, the reward to matching any two nodes is not initially known, but is instead predicted from the features available for both parties. Bipartite matching can be formulated as a linear program, allowing us to apply our decision-focused approach. The learning problem is to use node features to predict whether each edge is present or absent (a classification problem). The optimization problem is to find a maximum matching in the predicted graph.

Our experiments use the cora dataset [SNB+08]. The nodes are scientific papers and edges represent citation. Each node's feature vector indicating whether each word in a vocabulary appeared in the paper (there are 1433 such features). The overall graph has 2708 nodes. In order to construct instances for the decision problem, we partitioned the complete graph into 27 instances, each with 100 nodes, using metis [KK98]. We divided the nodes in each instance into the sides of a bipartite graph (of 50 nodes each) such that the number of edges crossing sides was maximized. The learning problem is much more challenging than before: unlike in budget allocation, the features do not contain enough information to reconstruct the citation network. However, a decision maker may still benefit from leveraging whatever signal is available.

Diverse recommendation: One application of submodular optimization is to select diverse sets of item, e.g. for recommendation systems or document summarization. Suppose

Table 8.2: *Accuracy of each method according to standard measures.*

	Budget allocation	Matching		Diverse recommendation	
	MSE	CE	AUC	CE	AUC
NN1-Decision	0.8673e-02 ± 1.83e-04	0.994 ± 0.002	0.501 ± 0.011	1.053 ± 0.005	0.593 ± 0.003
NN2-Decision	1.7118e-02 ± 2.65e-04	0.689 ± 0.004	**0.560 ± 0.006**	1.004 ± 0.022	0.577 ± 0.008
NN1-2Stage	0.0501e-02 ± 2.67e-06	0.696 ± 0.001	0.499 ± 0.013	0.703 ± 0.001	0.389 ± 0.003
NN2-2Stage	0.0530e-02 ± 2.27e-06	**0.223 ± 0.005**	0.498 ± 0.007	0.690 ± 0.000	**0.674 ± 0.004**
RF-2Stage	**0.0354e-02 ± 4.17e-06**	0.693 ± 0.000	0.500 ± 0.000	**0.689 ± 0.000**	0.500 ± 0.000

that each item i is associated with a set of topics $t(i)$. Then, we aim to select a set of k items which collectively cover as many topics as possible: $f(S) = |\bigcup_{i \in S} t(i)|$. Such formulations have been used across recommendation systems [AKBW15], text summarization [TO09], web search [AGHI09] and image segmentation [PJB14].

In many applications, the item-topic associations $t(i)$ are not known in advance. Hence, the learning task is to predict a binary matrix θ where θ_{ij} is 1 if item i covers topic j and 0 otherwise. The optimization task is to find a set of k items maximizing the number of topics covered according to θ. We consider a recommendation systems problem based on the Movielens dataset [Gro11] in which 2113 users rate 10197 movies (though not every user rated every movie). The items are the movies, while the topics are the top 500 actors. In our problem, the movie-actor assignments are unknown, and must be predicted only from user ratings. This is a *multilabel classification problem* where we attempt to predict which actors are associated with each movie. We randomly divided the movies into 101 problem instances, each with 100 movies. The feature matrix y contains the ratings given by each of the 2113 users for the 100 movies in the instance (with zeros where no rating is present).

Algorithms and experimental setup: In each domain, we randomly divided the instances into 80% training and 20% test. All results are averaged over 30 random splits. Our decision-focused framework was instantiated using feed-forward, fully connected neural networks as the underlying predictive model. All networks used ReLU activations. We experimented with networks with 1 layer, representing a restricted class of models, and 2-layer networks, where the hidden layer (of size 200) gives additional expressive power. We compared two training methods. First, the decision-focused approach proposed above.

Second, a two stage approach that uses a machine learning loss function (mean squared error for regression tasks and cross-entropy loss for classification). *This allows us to isolate the impact of the training method since both use the same underlying architecture.* We experimented with additional layers but observed little benefit for either method. All networks were trained using Adam with learning rate 10^{-3}. We refer to the 1-layer decision focused network as *NN1-Decision* and the 1-layer two stage network as *NN1-2Stage* (with analogous names for the 2-layer networks). We also compared to a random forest ensemble of 100 decisions trees (*RF-2Stage*). Gradient-based training cannot be applied to random forests, so benchmark represents a strong predictive model which can be used by two stage approaches but not by our framework. Lastly, we show performance for a random decision.

Solution quality: Table 8.1 shows the solution quality that each approaches obtains on the full pipeline; i.e., the objective value of its decision evaluated using the true parameters. Each value is the mean (over the 30 iterations) and a bootstrapped 95% confidence interval. For the budget allocation and diverse recommendation tasks, we varied the budget k. The decision-focused methods obtain the highest-performance across the board, tied with random forests on the synthetic budget allocation task.

We now consider each individual domain, starting with budget allocation. Both decision-focused methods substantially outperform the two-stage neural networks, obtaining at least 37% greater objective value. This demonstrates that with fixed predictive architecture, decision-focused learning can greatly improve solution quality. NN1-Decision performs somewhat better than NN2-Decision, suggesting that the simpler class of models is easier to train. However, NN1-2Stage performs significantly worse than NN1-Decision, indicating that alignment between training and the decision problem is highly important for simple models to succeed. RF-2Stage performs essentially equivalently to NN1-Decision. This is potentially surprising since random forest are a much more expressive model class. As we will see later, much of the random forest's success is due to the fact that the features in this synthetic domain are very high-signal; indeed, they suffice for near-perfect reconstruction. The next two domains, both based on real data, explore low-signal settings where highly

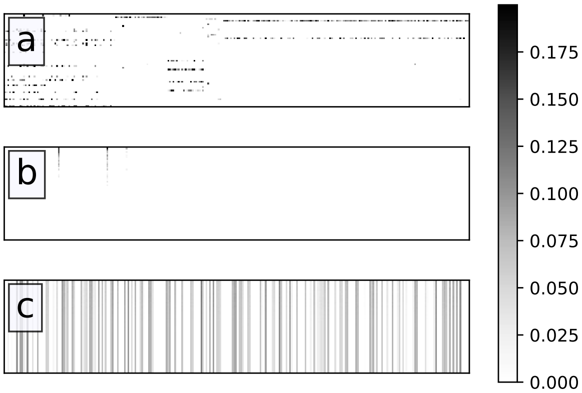


Figure 8.1: *Visualization of predictions made by each model. (a) ground truth (b) NN1-2Stage (c) NN1-Decision*

accurate recovery is impossible.

In bipartite matching, NN2-Decision obtains the highest overall performance, making nearly *over 70% more matches* than the next best method (RF-2Stage, followed closely by NN2-2Stage). Both 1-layer models perform extremely poorly, indicating that the more complex learning problem requires a more expressive model class. However, the highly expressive RF-2Stage does only marginally better than NN2-2Stage, demonstrating the critical role of aligning training and decision making.

In the diverse recommendation domain, NN1-Decision has the best performance, followed closely by NN2-Decision. NN2-2Stage trails by 23%, and NN1-2Stage performs extremely poorly. This highlights the importance of the training method within the same class of models: NN1-Decision obtains approximately 2.7 times greater objective value than NN1-2Stage. RF-2Stage also performs poorly in this domain, and is seemingly unable to extract any signal which boosts decision quality above that of random.

Exploration of learned models: We start out by showing the accuracy of each method according to standard measures, summarized in Table 8.2. For classification domains (diverse recommendation, matching), we show cross-entropy loss (which is directly optimized by the two stage networks) and AUC. For regression (the budget allocation domain), we show mean squared error (MSE). For budget allocation and diverse recommendation, we fixed $k = 10$.

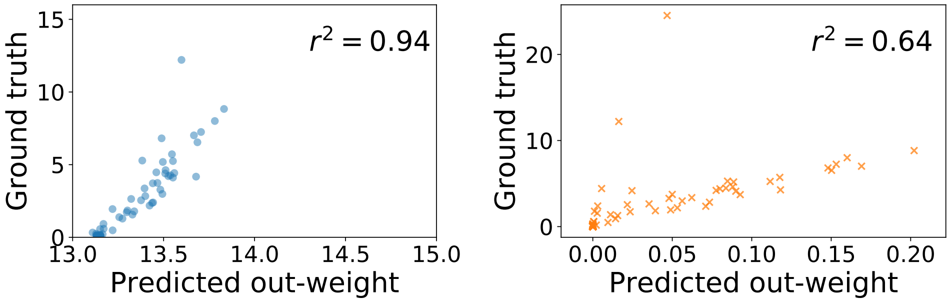


Figure 8.2: *Left: our method's predicted total out-weight for each item. Right: predictions from two stage method.*

The two-stage methods are, in almost all cases, significantly more accurate than the decision-focused networks despite their worse solution quality. Moreoever, no accuracy measure is well-correlated with solution quality. On budget allocation, the two decision-focused networks have the worst MSE but the best solution quality. On bipartite matching, NN2-2Stage has better cross-entropy loss but much worse solution quality than NN2-Decision. On diverse recommendation, NN2-2Stage has the best AUC but worse solution quality than either decision-focused network.

This incongruity raises the question of what differentiates the predictive models learned via decision-focused training. We now show more a more detailed exploration of each model's predictions. We focus first on the simpler case of the synthetic budget allocation task, comparing NN1-Decision and NN1-2Stage. However, the higher-level insights generalize across domains, detailed after.

Figure 8.1 shows each model's predictions on an example instance. Each heat map shows a predicted matrix θ, where dark entries correspond to a high prediction and light entries to low. The first matrix is the ground truth. The second matrix is the prediction made by NN1-2Stage, which matches the overall sparsity of the true θ but fails to recover almost all of the true connections. The last matrix corresponds to NN1-Decision and appears completely dissimilar to the ground truth. Nevertheless, these seemingly nonsensical predictions lead to the best quality decisions.

To investigate the connection between predictions and decision, Figure 8.2 aggregates

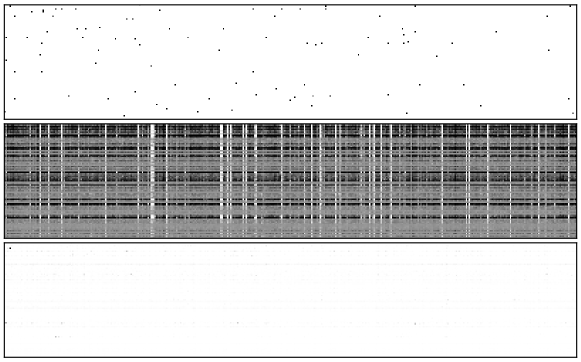


Figure 8.3: *Diverse recommendation predictions. Top to bottom: ground truth, our method's prediction (by NN2-Decision), two stage prediction (by NN2-2Stage)*

each model's predictions at the channel level. Formally, we examine the predicted out-weight for each channel u, i.e., the sum of the row θ_u. This is a coarse measure of u's importance for the optimization problem; channels with connections to many customers are more likely to be good candidates for the optimal set. Surprisingly, NN1-Decision's predicted out-weights are extremely well correlated with the ground truth out-weights ($r^2 = 0.94$). However, the absolute magnitude of its predictions are skewed: the bulk of channels have low outweight (less than 1), but NN1-Decision's predictions are all at least 13. By contrast NN1-2Stage has poorer correlation, making it less useful for identifying the outliers which comprise the optimal set. However, it better matches the values of low out-weight channels and hence attains better MSE. This illustrates how aligning the model's training with the optimization problem leads it to focus on qualities which are specifically important for decision making, even if this compromises accuracy elsewhere.

We now show more detailed analysis of the predictions made by each model in the other two domains: diverse recommendation and bipartite matching. The general trends are similar to those observed for budget allocation (although the results are somewhat messier for the real-data domains). We see that the decision-focused neural network makes apparently nonsensical predications. However, the out-weight that it predicts for each item is better correlated with the ground truth than for the two stage method.

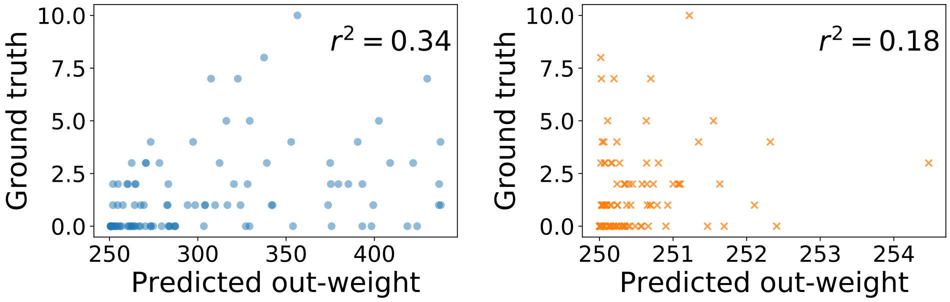


Figure 8.4: *Diverse recommendation predicted outweight according to NN2-Decision (right) and NN2-2Stage (left).*

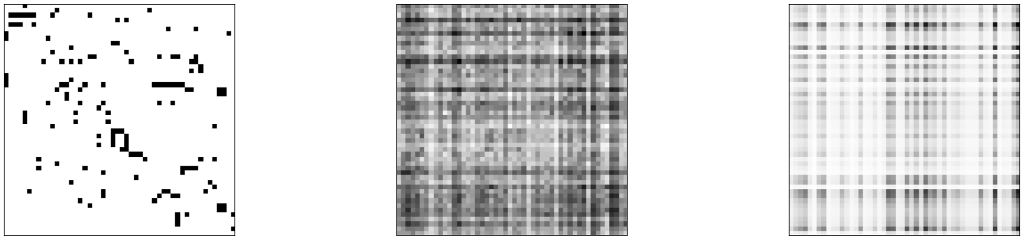


Figure 8.5: *Bipartite matching predictions. Left to right: ground truth adjacency matrix, our method's prediction (NN2-Decision), two stage prediction (NN2-2Stage).*

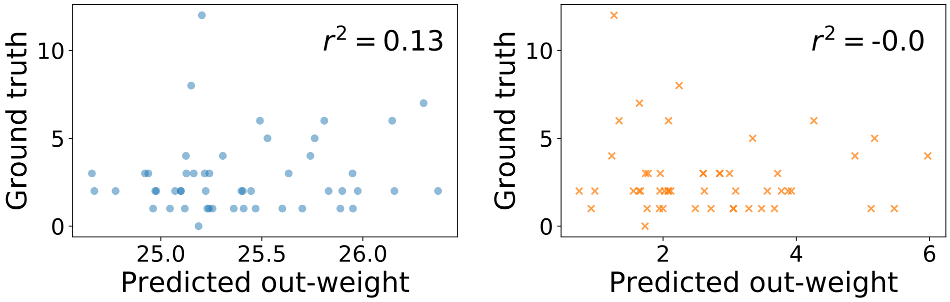


Figure 8.6: *Bipartite matching predicted outweight according to NN2-Decision (right) and NN2-2Stage (left).*

8.5 Conclusion

We propose a means of integrating a broad family of combinatorial optimization problems into the training of machine learning models by differentiating through solutions to a continuous relaxation of the discrete problem. This process aligns predictions with the end goals of a decision maker. Experimental results show that decision-focused learning can substantially improve solution quality (measured in terms of final optimization performance) across a variety of domains. By contrast, standard machine learning loss functions often fail to prioritize the qualities required for successful decision making. These results demonstrate that true end-to-end training is an important component of building a data-decisions pipeline.

Chapter 9

Decision-focused learning for tuberculosis medication adherence

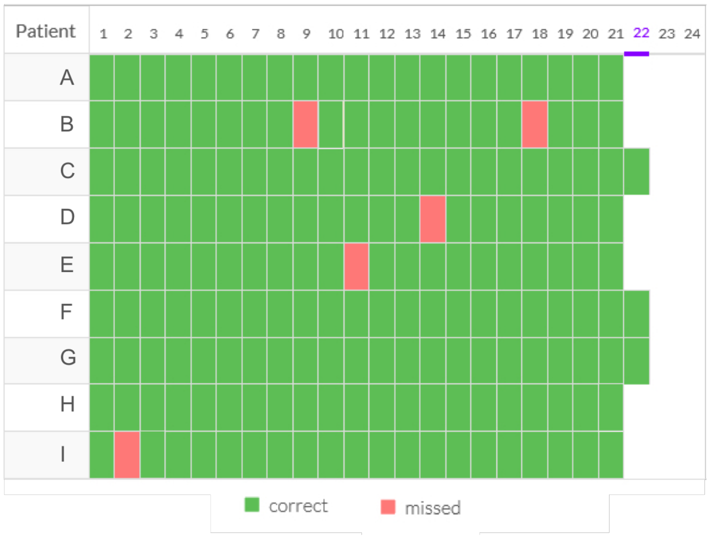
Tuberculosis (TB) is one of the largest challenges in public health, ranking in the top ten causes of death worldwide [Org18]. Treatment for TB requires a long (typically at least six month) course of daily antibiotics. Low adherence to this treatment results in a range of problems for patients and the community, including greater risk of reinfection or death and the development of drug-resistant strains [TGS$^+$05]. The gold-standard protocol for TB treatment, recommended by the WHO, is directly observed treatment (DOTS) where a health worker watches a patient take their medication each day. However, DOTS is impractical in much of the world for a variety of reasons and creates substantial barriers to care, especially if patients are required to travel to a treatment center every day to receive their medication. Digital adherence technologies (DATs) offer a more flexible alternative for frontline health workers to monitor adherence and offer support to nonadherent patients [SdMM$^+$18]. One example of a DAT, specifically relevant to this chapter, is the 99DOTS system [CGL$^+$19]. In 99DOTS, patients place a toll-free call to a specially-generated phone number each day to verify that they took their medication. Globally, a number of DATs have been proposed in different settings, with evidence to suggest that their adoption can improve adherence rates for a range of diseases [HMT$^+$17, CKB$^+$16, SDG$^+$15].

One potential use of DATs is to prioritize the highest-risk patients for potentially costly interventions by frontline health workers. Oftentimes, health workers face high case loads and cannot carry out time-intensive interventions (e.g., home visits) with the entire population of patients who might benefit. Accordingly, DATs offer an opportunity for health workers to more easily observe which patients are at greater risk of non-adherence and targeted their limited resources accordingly. Currently, such targeted happens mostly in a reactive fashion, where health workers intervene with patients after that patient has missed several doses in the recent past.

This chapter explores the potential for machine learning to support a more proactive stance by predicting the risk of future non-adherence for each patient and suggesting an optimal set of interventions to visit patients before they miss doses. This project is based on a collaboration with the Government of Maharashtra which focused on TB care in the city of Mumbai. In Mumbai, TB patients are enrolled in the 99DOTS system [CGL+19], developed by the healthcare technology company Everwell [**?**]. We start by formulating an optimization problem which models the challenge of assigning health workers to carry out in-person interventions with patients. Then, using historical adherence data from 99DOTS on TB patients in Mumbai, we develop a machine learning model to predict future patient adherence, which appears as an unknown parameter in the objective function of the optimization problem. We compare the machine learning model's performance when trained using the decision-focused methodology introduced in Chapter 8 to when it is trained using a standard two-stage approach. Our results show that decision-focused training improves the number of successful interventions suggested by the system by approximately 15%.

9.1 Optimization formulation

We focus on a specific optimization problem that models the allocation of health workers to intervene with patients who are at risk in the near future. This provides a case study to evaluate the potential benefits of decision-focused learning in this domain. However, we emphasize that our system can be easily modified to capture other intervention problems.



Patient	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩			
B	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟩	🟩			
C	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩		
D	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟩	🟩	🟩	🟩	🟩	🟩			
E	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩			
F	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩		
G	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩		
H	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩			
I	🟩	🟥	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩			

🟩 correct 🟥 missed

Figure 9.1: *99DOTS electronic adherence dashboard seen by health workers for a given month. Missed doses are marked in red while consumed doses are marked in green.*

Such flexibility is one benefit to our technical approach, which allows the ML model to *automatically* adapt to the problem specified by a domain expert.

Our optimization problem models a health worker who plans a series of interventions over the course of a week. The health worker is responsible for a population of patients across different locations, and may visit one location each day. We use location identifiers at the level of the TB Unit since this is the most granular identifier which is shared by the majority of patients in our dataset. Visiting a location allows the health worker to intervene with any of the patients at that location. The optimization problem is to select a set of locations to visit which maximizes the number of patients who receive an intervention *on or before the first day they would have missed a dose*. We refer to this quantity as the number of *successful interventions*, which we choose as our objective for two reasons. First, it measures the extent to which the health worker can proactively engage with patients before adherence suffers. Second, this objective resolves an important challenge in evaluating counterfactual outcomes for this domain. The key problem is that we do not observe the exact set of interventions carried out by health workers in the dataset. However, we do know that existing policies call for health workers to intervene with patients after they miss several consecutive doses, at which point the system marks the patient as "high" priority instead of "medium". Our objective counts patients who start the week at "medium" priority but who will in the future start to miss doses. This allows our objective to measure the extent to which health workers successfully intervene proactively with patients who would not otherwise have been targets for intervention but who are actually at high risk.

We now show how this optimization problem can be formalized as a linear program. We have a set of locations $i = 1...L$ and patients $j = 1...N$ where patient j has location ℓ_j. Over days of the week $t = 1...7$, the objective coefficient c_{jt} is 1 if an intervention on day t with patient j is successful and 0 otherwise. Our decision variable is x_{it}, and takes the value 1 if the health worker visit location i on day t and 0 otherwise. With this notation, the final LP is as follows:

$$\begin{aligned} \max_x \sum_{t=1}^{7} \sum_{i=1}^{L} x_{it} &\left(\sum_{j:\ell_j = i} c_{jt} \right) \\ \text{s.t. } \sum_{i=1}^{L} x_{it} &\leq 1, t = 1...7 \\ \sum_{t=1}^{7} x_{it} &\leq 1, i = 1...L \\ 0 \leq x_{it} &\leq 1 \quad \forall i, t \end{aligned}$$

where the second constraint prevents the objective from double-counting multiple visit to a location. We remark that the feasible region of the LP can be shown to be equivalent to a bipartite matching polytope, implying that the optimal solution is always integral.

9.2 Integrating machine learning and optimization

The machine learning task is to predict the values of the c_{jt}, which are unknown at the start of the week. To train machine learning models for this task, we use data provided by the Government of Maharashtra on TB patients in Mumbai. We have two main sets of information about each patients. First, some basic demographic information (weight-band, age-band, gender and treatment center ID). Second, the patient's history of adherence to

Table 9.1: *Data Summary. *Doses per patient was calculated only on patients enrolled at least 6 months before Sept 2018.*

Metric	Count
Total doses recorded	2,169,976
—By patient call	1,459,908
—Manual (entered by health worker)	710,068
Registered phones	38,000
Patients	16,975
Health centers	252
Doses recorded per patient*	
—Quartiles	57/149/188
—Min/Mean/Max	1/136/1409
Active patients per center per month	
—Quartiles	7/18/35
—Min/Mean/Max	1/25/226

date as recorded by 99DOTS. Patients in 99DOTS receive each sleeve of pills wrapped in a cover. The cover contains a hidden phone number associated with each pill; when patients retrieve the pill for each day, they also reveal the associated phone number. Patients place a toll-free call to the number in order to indicate that they took their medication. The dataset records whether this call was received each day, along with some metadata such as the time of day at which the call was placed. In total, the data contains over 2.1 million dose records for about 17,000 patients, served by 252 health centers across Mumbai from Feb 2017 to Sept 2018. **Table 9.1** provides an overview of the dataset. Using this data, we compare the performance of three predictive models. Each of these models makes a prediction about the value of the c_{jt} variables representing future adherence for a patient. We compare models based both on their predictive accuracy for this task as well as the solution quality induced by using their predictions to solve the above LP (i.e., using the predictions to produce a proposed set of interventions by health workers).

First, we implement a baseline referred to as lw-Misses. This baseline approximates the current policies used by frontline health workers, where patients are prioritized for intervention after missing some number of doses. To implement this heuristic on the context of the task of predicting c_{jt}, we threshold the number of doses patient j missed in the last week, setting $c_{jt} = 0$ for all t if this value falls below the threshold τ and $c_{jt} = 1$ otherwise. We used $\tau = 1$ since it performed best.

Second, we trained a neural network to predict the true c_{jt} as a classification prediction task using cross-entropy loss. This model, referred to as LEAP, uses a fully-connected layer to combine the output of both a LSTM (which uses the time series of a patient's past adherence) and that of another fully-connected layer (which uses the patient's demographic features). More details on LEAP can be found in [KWS+19]. This model represents a well-engineered two stage approach.

Third, we trained the same LEAP architecture to predict c_{jt} using performance on the above optimization problem as the loss function. This is accomplished by using the quadratically regularized LP formulation introduced in Chapter 8. We refer to this model as LEAP-Decision.

We created instances of the decision problem by randomly partitioning patients into groups of 100, modeling a health worker under severe resource constraints (as they would benefit most from such a system). We included all patients, including those with no missed doses in the last week, since the overall resource allocation problem over locations must still account for them.

Figure 9.2 shows results for this task. In the top row, we see that LEAP and LEAP-Decision both outperform lw-Misses, as expected. LEAP-Decision improves the number of successful interventions by approximately 15% compared to LEAP, demonstrating the value of tailoring the learned model to a given planning problem. LEAP-Decision actually has worse AUC than either LEAP or lw-Misses, indicating that typical measures of machine learning accuracy are not a perfect proxy for utility in decision making. To investigate what specifically distinguishes the predictions made by LEAP-Decision, the bottom row of **Figure 9.2** shows scatter plots of the predicted utility at each location according to LEAP and LEAP-Decision versus the true values. Visually, LEAP-Decision appears better

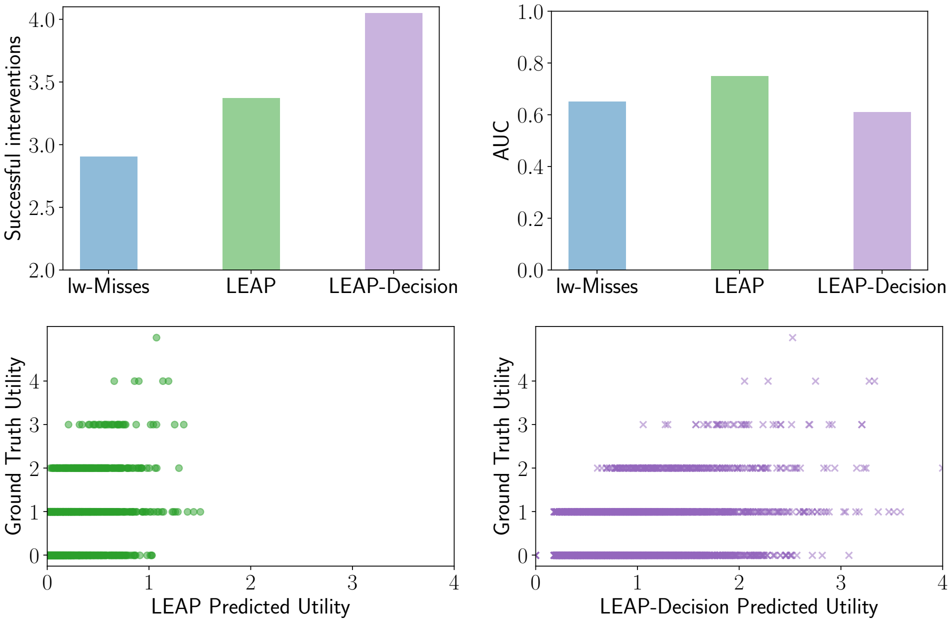


Figure 9.2: *Results for decision focused learning problem. Top row: successful interventions and AUC for each method. Bottom row: visualizations of model predictions.*

able to distinguish the high-utility outliers which are most important to making good decisions. Quantitatively, LEAP-Decision's predictions have worse correlation with the ground truth overall (0.463, versus 0.519 for LEAP), but better correlation on locations where the true utility is strictly more than 1 (0.504 versus 0.409). Hence, decision-focused training incentivizes the model to focus on making accurate predictions specifically for locations that are likely to be good candidates for an intervention. This demonstrates the benefit of our flexible machine learning modeling approach, which can use custom-defined loss functions to automatically adapt to particular decision problems.

Chapter 10

Learning to optimize on graphs

While deep learning has proven enormously successful at a range of tasks, an expanding area of interest concerns systems that can flexibly combine learning with optimization. Examples include recent attempts to solve combinatorial optimization problems using neural architectures [VFJ15, KDZ$^+$17, BPL$^+$16, KvHW19], as well as work which incorporates explicit optimization algorithms into larger differentiable systems [AK17, DAK17, WDT19]. The ability to combine learning and optimization promises improved performance for real-world problems which require decisions to be made on the basis of machine learning predictions by enabling end-to-end training which focuses the learned model on the decision problem at hand.

We focus on graph optimization problems, an expansive subclass of combinatorial optimization. While graph optimization is ubiquitous across domains, complete applications must also solve machine learning challenges. For instance, the input graph is usually incomplete; some edges may be unobserved or nodes may have attributes that are only partially known. Recent work has introduced sophisticated methods for tasks such as link prediction and semi-supervised classification [PARS14, KW17, SKB$^+$18, HYL17, ZC18], but these methods are developed in isolation of downstream optimization tasks. Most current solutions use a two-stage approach which first trains a model using a standard loss and then plugs the model's predictions into an optimization algorithm ([YG12, BAC16, BSBS18, BCP$^+$16, TWL$^+$16]). However, predictions which minimize a standard loss function (e.g., cross-entropy) may be suboptimal for specific optimization tasks, especially in difficult settings where even the best model is imperfect.

A preferable approach is to incorporate the downstream optimization problem into the training of the machine learning model. A great deal of recent work takes a pure end-to-end approach where a neural network is trained to predict a solution to the optimization problem using supervised or reinforcement learning [VFJ15, KDZ$^+$17, BPL$^+$16, KvHW19]. However, this often requires a large amount of data and results in suboptimal performance because the network needs to discover algorithmic structure entirely from scratch. Between the extremes of an entirely two stage approach and pure end-to-end architectures, *decision-focused learning* [DAK17, WDT19] embeds a solver for the optimization problem as a differentiable layer within a learned system. This allows the model to train using the downstream performance that it induces as the loss, while leveraging prior algorithmic knowledge for optimization. The downside is that this approach requires manual effort to develop a differentiable solver for each particular problem and often results in cumbersome systems that must, e.g, call a quadratic programming solver every forward pass.

We propose a new approach that gets the best of both worlds: incorporate a solver for a simpler optimization problem as a differentiable layer, and then learn a representation that maps the (harder) problem of interest onto an instance of the simpler problem. Compared to earlier approaches to decision-focused learning, this places more emphasis on the representation learning component of the system and simplifies the optimization component. However, compared to pure end-to-end approaches, we only need to learn the reduction to the simpler problem instead of the entire algorithm.

In this work, we instantiate the simpler problem as a differentiable version of k-means clustering. Clustering is motivated by the observation that graph neural networks embed nodes into a continuous space, allowing us to approximate optimization over the discrete graph with optimization in continuous embedding space. We then interpret the cluster assignments as a solution to the discrete problem. We instantiate this approach for two classes of optimization problems: those that require *partitioning* the graph (e.g., community detection or maxcut), and those that require *selecting a subset of K nodes* (facility location, influence maximization, immunization, etc). We don't claim that clustering is the right algorithmic structure for all tasks, but it is sufficient for many problems as shown in this chapter.

In short, we make three contributions. First, we introduce a general framework for integrating graph learning and optimization, with a simpler optimization problem in continuous space as a proxy for the more complex discrete problem. Second, we show how to differentiate through the clustering layer, allowing it to be used in deep learning systems. Third, we show experimental improvements over both two-stage baselines as well as alternate end-to-end approaches on a range of example domains.

10.1 Related work

We build on a recent work on decision-focused learning [DAK17, WDT19, DSB$^+$19], which includes a solver for an optimization problem into training in order to improve performance on a downstream decision problem. A related line of work develops and analyzes effective surrogate loss functions for predict-then-optimize problems [EG17, BEGT19]. Some work in structured prediction also integrates differentiable solvers for discrete problems (e.g., image segmentation [DK17] or time series alignment [MB18]). Our work differs in two ways. First, we tackle more difficult optimization problems. Previous work mostly focuses on convex problems [DAK17] or discrete problems with near-lossless convex relations [WDT19, DK17]. We focus on highly combinatorial problems where the methods of choice are hand-designed discrete algorithms. Second, in response to this difficulty, we differ methodologically in that we do not attempt to include a solver for the exact optimization problem at hand (or a close relaxation of it). Instead, we include a more generic algorithmic skeleton that is automatically finetuned to the optimization problem at hand.

There is also recent interest in training neural networks to solve combinatorial optimization problems [VFJ15, KDZ$^+$17, BPL$^+$16, KvHW19]. While we focus mostly on combining graph learning with optimization, our model can also be trained just to solve an optimization problem given complete information about the input. The main methodological difference is that we include more structure via a differentiable k-means layer instead of using more generic tools (e.g., feed-forward or attention layers). Another difference is that prior work mostly trains via reinforcement learning. By contrast, we use a differentiable approximation to the objective which removes the need for a policy gradient estimator. This is a benefit of our architecture, in which the final decision is fully differentiable in terms of the model parameters instead of requiring non-differentiable selection steps (as in [KDZ$^+$17, BPL$^+$16, KvHW19]). We give our end-to-end baseline (“GCN-e2e") the same advantage by training it with the same differentiable decision loss as our own model instead of forcing it to use noisier policy gradient estimates.

Finally, some work uses deep architectures as a part of a clustering algorithm [TGC$^+$14, LUZ17, GGLY17, SSL$^+$18, NHG$^+$19], or includes a clustering step as a component of a deep network [GRB$^+$16, GvSS17, YYM$^+$18]. While some techniques are similar, the overall task we address and framework we propose are entirely distinct. Our aim is not to cluster a Euclidean dataset (as in [TGC$^+$14, LUZ17, GGLY17, SSL$^+$18]), or to solve perceptual grouping problems (as in [GRB$^+$16, GvSS17]). Rather, we propose an approach for graph optimization problems. Perhaps the closest of this work is Neural EM [GvSS17], which uses an unrolled EM algorithm to learn representations of visual objects. Rather than using EM to infer representations for objects, we use k-means in graph embedding space to solve an optimization problem. There is also some work which uses deep networks for graph clustering [XGF16, YCH$^+$16]. However, none of this work includes an explicit clustering algorithm in the network, and none consider our goal of integrating graph learning and optimization.

10.2 Setting

We consider settings that combine learning and optimization. The input is a graph $G = (V, E)$, which is in some way partially observed. We will formalize our problem in terms of

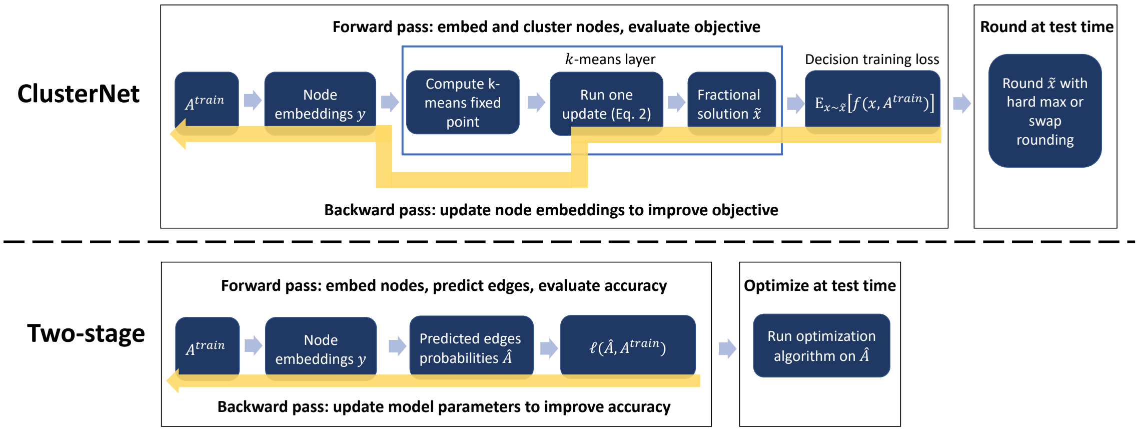

ClusterNet
Forward pass: embed and cluster nodes, evaluate objective
A^{train}
Node embeddings y
k-means layer
Compute k-means fixed point
Run one update (Eq. 2)
Fractional solution $\tilde{x}$
Decision training loss
$\mathrm{E}_{x\sim\tilde{x}}[f(x, A^{train})]$
Backward pass: update node embeddings to improve objective
Round at test time
Round $\tilde{x}$ with hard max or swap rounding
Two-stage
Forward pass: embed nodes, predict edges, evaluate accuracy
A^{train}
Node embeddings y
Predicted edges probabilities $\hat{A}$
$\ell(\hat{A}, A^{train})$
Backward pass: update model parameters to improve accuracy
Optimize at test time
Run optimization algorithm on $\hat{A}$

Figure 10.1: *Top:* ClusterNet*, our proposed system. Bottom: a typical two-stage approach.*

link prediction as an example, but our framework applies to other common graph learning problems (e.g., semi-supervised classification). In link prediction, the graph is not entirely known; instead, we observe only training edges $E^{train} \subset E$. Let A denote the adjacency matrix of the graph and A^{train} denote the adjacency matrix with only the training edges. The learning task is to predict A from A^{train}. In domains we consider, the motivation for performing link prediction, is to solve a decision problem for which the objective depends on the full graph. Specifically, we have a decision variable x, objective function $f(x, A)$, and a feasible set $\mathcal{X}$. We aim to solve the optimization problem

$$\max_{x \in \mathcal{X}} f(x, A). \tag{10.1}$$

However, A is unobserved. We can also consider an inductive setting in which we observe graphs $A_1, ..., A_m$ as training examples and then seek to predict edges for a partially observed graph from the same distribution. The most common approach to either setting is to train a model to reconstruct A from A^{train} using a standard loss function (e.g., cross-entropy), producing an estimate $\hat{A}$. The *two-stage* approach plugs $\hat{A}$ into an optimization algorithm for Problem 10.1, maximizing $f(x, \hat{A})$.

We propose end-to-end models which map from A^{train} directly to a feasible decision x. The model will be trained to maximize $f(x, A^{train})$, i.e., the quality of its decision evaluated on the training data (instead of a loss $\ell(\hat{A}, A^{train})$ that measures purely predictive accuracy). One approach is to “learn away" the problem by training a standard model (e.g., a GCN) to map directly from A^{train} to x. However, this forces the model to entirely rediscover algorithmic concepts, while two-stage methods are able to exploit highly sophisticated optimization methods. We propose an alternative that embeds algorithmic structure into the learned model, getting the best of both worlds.

10.3 Approach: ClusterNet

Our proposed ClusterNet system (Figure 1) merges two differentiable components into a system that is trained end-to-end. First, a *graph embedding* layer which uses A^{train} and any node features to embed the nodes of the graph into $\mathbb{R}^p$. In our experiments, we use GCNs [KW17]. Second, a layer that performs *differentiable optimization*. This layer takes the continuous-space embeddings as input and uses them to produce a solution x to the graph optimization problem. Specifically, we propose to use a layer that implements a differentiable version of K-means clustering. This layer produces a soft assignment of the nodes to clusters, along with the cluster centers in embedding space.

The intuition is that cluster assignments can be interpreted as the solution to many common graph optimization problems. For instance, in community detection we can interpret the cluster assignments as assigning the nodes to communities. Or, in maxcut, we can use two clusters to assign nodes to either side of the cut. Another example is maximum coverage and related problems, where we attempt to select a set of K nodes which cover (are neighbors to) as many other nodes as possible. This problem can be approximated by clustering the nodes into K components and choosing nodes whose embedding is close to the center of each cluster. We do not claim that any of these problems is exactly reducible to K-means. Rather, the idea is that including K-means as a layer in the network provides a useful inductive bias. This algorithmic structure can be fine-tuned to specific problems by training the first component, which produces the embeddings, so that the learned representations induce clusterings with high objective value for the underlying downstream optimization task. We now explain the optimization layer of our system in greater detail. We start by detailing the forward and the backward pass for the clustering procedure, and then explain how the cluster assignments can be interpreted as solutions to the graph optimization problem.

10.3.1 Forward pass

Let x_j denote the embedding of node j and μ_k denote the center of cluster k. r_{jk} denotes the degree to which node j is assigned to cluster k. In traditional K-means, this is a binary quantity, but we will relax it to a fractional value such that $\sum_k r_{jk} = 1$ for all j. Specifically, we take $r_{jk} = \frac{\exp(-\beta||x_j - \mu_k||)}{\sum_\ell \exp(-\beta||x_j - \mu_\ell||)}$, which is a soft-min assignment of each point to the cluster centers based on distance. While our architecture can be used with any norm $|| \cdot ||$, we use the negative cosine similarity due to its strong empirical performance. β is an inverse-temperature hyperparameter; taking $\beta \to \infty$ recovers the standard k-means assignment. We can optimize the cluster centers via an iterative process analogous to the typical k-means updates by alternately setting

$$\mu_k = \frac{\sum_j r_{jk} x_j}{\sum_j r_{jk}} \ \forall k = 1...K \quad r_{jk} = \frac{\exp(-\beta||x_j - \mu_k||)}{\sum_\ell \exp(-\beta||x_j - \mu_\ell||)} \ \forall k = 1...K, j = 1...n. \tag{10.2}$$

These iterates converge to a fixed point where μ remains the same between successive updates [Mac03]. The output of the forward pass is the final pair (μ, r).

10.3.2 Backward pass

We will use the implicit function theorem to analytically differentiate through the fixed point that the forward pass k-means iterates converge to, obtaining expressions for $\frac{\partial \mu}{\partial x}$ and $\frac{\partial r}{\partial x}$. Previous work [DAK17, WDT19] has used the implicit function theorem to differentiate through the KKT conditions of optimization problems; here we take a more direct approach that characterizes the update process itself. Doing so allows us to backpropagate gradients from the decision loss to the component that produced the embeddings x. Define a function

$f : \mathbb{R}^{Kp} \to \mathbb{R}$ as

$$f_{i,\ell}(\mu, x) = \mu_i^\ell - \frac{\sum_j r_{jk} x_j^\ell}{\sum_j r_{jk}} \tag{10.3}$$

Now, (μ, x) are a fixed point of the iterates if $f(\mu, x) = \mathbf{0}$. Applying the implicit function theorem yields that $\frac{\partial \mu}{\partial x} = -\left[\frac{\partial f(\mu,x)}{\partial \mu}\right]^{-1} \frac{\partial f(\mu,x)}{\partial x}$, from which $\frac{\partial r}{\partial x}$ can be easily obtained via the chain rule.

Exact backward pass: We now examine the process of calculating $\frac{\partial \mu}{\partial x}$. Both $\frac{\partial f(\mu,x)}{\partial x}$ and $\frac{\partial f(\mu,x)}{\partial \mu}$ can be easily calculated in closed form (see appendix). Computing the former requires time $O(nKp^2)$. Computing the latter requires $O(npK^2)$ time, after which it must be inverted (or else iterative methods must be used to compute the product with its inverse). This requires time $O(K^3p^3)$ since it is a matrix of size $(Kp) \times (Kp)$. While the exact backward pass may be feasible for some problems, it quickly becomes burdensome for large instances. We now propose a fast approximation.

Approximate backward pass: We start from the observation that $\frac{\partial f}{\partial \mu}$ will often be dominated by its diagonal terms (the identity matrix). The off-diagonal entries capture the extent to which updates to one entry of μ indirectly impact other entries via changes to the cluster assignments r. However, when the cluster assignments are relatively firm, r will not be highly sensitive to small changes to the cluster centers. We find to be typical empirically, especially since the optimal choice of the parameter β (which controls the hardness of the cluster assignments) is typically fairly high. Under these conditions, we can approximate $\frac{\partial f}{\partial \mu}$ by its diagonal, $\frac{\partial f}{\partial \mu} \approx I$. This in turn gives $\frac{\partial \mu}{\partial x} \approx -\frac{\partial f}{\partial x}$.

We can formally justify this approximation when the clusters are relatively balanced and well-separated. More precisely, define $c(j) = \arg\max_i r_{ji}$ to be the closest cluster to point j. Proposition 1 (proved in the appendix) shows that the quality of the diagonal approximation improves exponentially quickly in the product of two terms: β, the hardness of the cluster assignments, and δ, which measures how well separated the clusters are. α (defined below) measures the balance of the cluster sizes. We assume for convenience that the input is scaled so $||x_j||_1 \leq 1 \, \forall j$.

Theorem 23. *Suppose that for all points j, $||x_j - \mu_i|| - ||x_j - \mu_{c(j)}|| \geq \delta$ for all $i \neq c(j)$ and that for all clusters i, $\sum_{j=1}^{n} r_{ji} \geq \alpha n$. Moreover, suppose that $\beta\delta > \log \frac{2\beta K^2}{\alpha}$. Then, $\left|\left|\frac{\partial f}{\partial \mu} - I\right|\right|_1 \leq \exp(-\delta\beta)\left(\frac{K^2\beta}{\frac{1}{2}\alpha - K^2\beta\exp(-\delta\beta)}\right)$ where $||\cdot||_1$ is the operator 1-norm.*

We now show that the approximate gradient obtained by taking $\frac{\partial f}{\partial \mu} = I$ can be calculated by unrolling a single iteration of the forward-pass updates from Equation 10.2 at convergence. Examining Equation 10.3, we see that the first term (μ_i^ℓ) is constant with respect to x, since here μ is a fixed value. Hence,

$$-\frac{\partial f_k}{\partial x} = \frac{\partial}{\partial x}\frac{\sum_j r_{jk}x_j}{\sum_j r_{jk}}$$

which is just the update equation for μ_k. Since the forward-pass updates are written entirely in terms of differentiable functions, we can automatically compute the approximate backward pass with respect to x (i.e., compute products with our approximations to $\frac{\partial \mu}{\partial x}$ and $\frac{\partial r}{\partial x}$) by applying standard autodifferentiation tools to the final update of the forward pass. Compared to computing the exact analytical gradients, this avoids the need to explicitly reason about or invert $\frac{\partial f}{\partial \mu}$. The final iteration (the one which is differentiated through) requires time $O(npK)$, *linear* in the size of the data.

Compared to differentiating by unrolling the entire sequence of updates in the computational graph (as has been suggested for other problems [Dom12, ADG$^+$16, ZJRP$^+$15]), our approach has two key advantages. First, it avoids storing the entire history of updates and backpropagating through all of them. The runtime for our approximation is independent of the number of updates needed to reach convergence. Second, *we can in fact use entirely non-differentiable operations to arrive at the fixed point*, e.g., heuristics for the K-means problem, stochastic methods which only examine subsets of the data, etc. This allows the forward pass to scale to larger datasets since we can use the best algorithmic tools available, not just those that can be explicitly encoded in the autodifferentiation tool's computational graph.

10.3.3 Obtaining solutions to the optimization problem

Having obtained the cluster assignments r, along with the centers μ, in a differentiable manner, we need a way to **(1)** differentiably interpret the clustering as a soft solution to the optimization problem, **(2)** differentiate a relaxation of the objective value of the graph optimization problem in terms of that solution, and then **(3)** round to a discrete solution at test time. We give a generic means of accomplishing these three steps for two broad classes of problems: those that involve *partitioning the graph into K disjoint components*, and those that that involve *selecting a subset of K nodes*.

Partitioning: **(1)** We can naturally interpret the cluster assignments r as a soft partitioning of the graph. **(2)** One generic continuous objective function (defined on soft partitions) follows from the random process of assigning each node j to a partition with probabilities given by r_j, repeating this process independently across all nodes. This gives the expected training decision loss $\ell = \mathbb{E}_{r^{hard} \sim r}[f(r^{hard}, A^{train})]$, where $r^{hard} \sim r$ denotes this random assignment. ℓ is now differentiable in terms of r, and can be computed in closed form via standard autodifferentiation tools for many problems of interest (see Section 10.4). We remark that when the expectation is not available in closed form, our approach could still be applied by repeatedly sampling $r^{hard} \sim r$ and using a policy gradient estimator to compute the gradient of the resulting objective. **(3)** At test time, we simply apply a hard maximum to r to obtain each node's assignment.

Subset selection: **(1)** Here, it is less obvious how to obtain a subset of K nodes from the cluster assignments. Our continuous solution will be a vector x, $0 \leq x \leq 1$, where $||x||_1 = K$. Intuitively, x_j is the probability of including x_j in the solution. Our approach obtains x_j by placing greater probability mass on nodes that are near the cluster centers. Specifically, each center μ_i is endowed with one unit of probability mass, which it allocates to the points x as $a_{ij} = \text{softmin}(\eta||x - \mu_i||)_j$. The total probability allocated to node j is $b_j = \sum_{i=1}^{K} a_{ij}$. Since we may have $b_j > 1$, we pass b through a sigmoid function to cap the entries at 1; specifically, we take $x = 2 * \sigma(\gamma b) - 0.5$ where γ is a tunable parameter. If the resulting x exceeds the budget constraint ($||x||_1 > K$), we instead output $\frac{Kx}{||x||_1}$ to ensure a feasible solution.

Table 10.1: *Performance on the community detection task*

	Learning + optimization					Optimization				
	cora	cite.	prot.	adol	fb	cora	cite.	prot.	adol	fb
ClusterNet	**0.54**	**0.55**	**0.29**	**0.49**	**0.30**	**0.72**	**0.73**	**0.52**	**0.58**	0.76
GCN-e2e	0.16	0.02	0.13	0.12	0.13	0.19	0.03	0.16	0.20	0.23
Train-CNM	0.20	0.42	0.09	0.01	0.14	0.08	0.34	0.05	0.57	**0.77**
Train-Newman	0.09	0.15	0.15	0.15	0.08	0.20	0.23	0.29	0.30	0.55
Train-SC	0.03	0.02	0.03	0.23	0.19	0.09	0.05	0.06	0.49	0.61
GCN-2stage-CNM	0.17	0.21	0.18	0.28	0.13	-	-	-	-	-
GCN-2stage-Newman	0.00	0.00	0.00	0.14	0.02	-	-	-	-	-
GCN-2stage-SC	0.14	0.16	0.04	0.31	0.25	-	-	-	-	-

Table 10.2: *Performance on the facility location task.*

	Learning + optimization					Optimization				
	cora	cite.	prot.	adol	fb	cora	cite.	prot.	adol	fb
ClusterNet	**10**	**14**	**6**	**6**	**4**	**9**	**14**	**6**	**5**	**3**
GCN-e2e	12	15	8	**6**	5	11	**14**	7	6	5
Train-greedy	14	16	8	8	6	**9**	**14**	7	6	5
Train-gonzalez	12	17	8	**6**	6	10	15	7	7	**3**
GCN-2Stage-greedy	14	17	8	7	6	-	-	-	-	-
GCN-2Stage-gonzalez	13	17	8	**6**	6	-	-	-	-	-

(2) We interpret this solution in terms of the objective similarly as above. Specifically, we consider the result of drawing a discrete solution $x^{hard} \sim x$ where every node j is included (i.e., set to 1) independently with probability x_j from the end of step **(1)**. The training objective is then $\mathbb{E}_{x^{hard} \sim x}[f(x^{hard}, A^{train})]$. For many problems, this can again be computed and differentiated through in closed form (see Section 10.4).

(3) At test time, we need a feasible discrete vector x; note that independently rounding the individual entries may produce a vector with more than K ones. Here, we apply a fairly generic approach based on pipage rounding [AS04], a randomized rounding scheme which has been applied to many problems (particularly those with submodular objectives). Pipage rounding can be implemented to produce a random feasible solution in time $O(n)$ [KLHK17]; in practice we round several times and take the solution with the best decision loss on the observed edges. While pipage rounding has theoretical guarantees only for specific classes of functions, we find it to work well even in other domains (e.g., facility location). However, more domain-specific rounding methods can be applied if available.

10.4 Experimental results

We now show experiments on domains that combine link prediction with optimization.

Learning problem: In link prediction, we observe a partial graph and aim to infer which unobserved edges are present. In each of the experiments, we hold out 60% of the edges in the graph, with 40% observed during training. We used a graph dataset which is not included in our results to set our method's hyperparameters, which were kept constant across datasets (see appendix for details). The learning task is to use the training edges to predict whether the remaining edges are present, after which we will solve an optimization problem on the predicted graph. The objective is to find a solution with high objective value measured on the *entire* graph, not just the training edges.

Optimization problems: We consider two optimization tasks, one from each of the broad classes introduced above. First, *community detection* aims to partition the nodes of the graph into K distinct subgroups which are dense internally, but with few edges across groups. Formally, the objective is to find a partition maximizing the modularity [New06b], defined as

$$Q(r) = \frac{1}{2m} \sum_{u,v \in V} \sum_{k=1}^{K} \left[A_{uv} - \frac{d_u d_v}{2m} \right] r_{uk} r_{vk}.$$

Here, d_v is the degree of node v, and r_{vk} is 1 if node v is assigned to community k and zero otherwise. This measures the number of edges within communities compared to the expected number if edges were placed randomly. Our clustering module has one cluster for each of the K communities. Defining B to be the modularity matrix with entries $B_{uv} = A_{uv} - \frac{d_u d_v}{2m}$, our training objective (the expected value of a partition sampled according to r) is $\frac{1}{2m}\text{Tr}\left[r^\top B^{train} r\right]$.

Second, minmax *facility location*, where the problem is to select a subset of K nodes from the graph, minimizing the maximum distance from any node to a facility (selected node). Letting $d(v, S)$ be the shortest path length from a vertex v to a set of vertices S, the objective is $f(S) = \min_{|S| \leq k} \max_{v \in V} d(v, S)$. To obtain the training loss, we take two steps. First, we replace $d(v, S)$ by $\mathbb{E}_{S\sim x}[d(v, S)]$, where $S \sim x$ denotes drawing a set from the product distribution with marginals x. This can easily be calculated in closed form [KLHK17]. Second, we replace the min with a softmin.

Baseline learning methods: We instantiate ClusterNet using a 2-layer GCN for node embeddings, followed by a clustering layer. We compare to three families of baselines. *First*, GCN-2stage, the two-stage approach which first trains a model for link prediction, and then inputs the predicted graph into an optimization algorithm. For link prediction, we use the GCN-based system of [SKB$^+$18] (we also adopt their training procedure, including negative sampling and edge dropout). For the optimization algorithms, we use standard approaches for each domain, outlined below. *Second*, “train", which runs each optimization algorithm only on the observed training subgraph (without attempting any link prediction). *Third*, GCN-e2e, an end-to-end approach which does not include explicit algorithm structure. We train a GCN-based network to directly predict the final decision variable (r or x) using the same training objectives as our own model. Empirically, we observed best performance with a 2-layer GCN. This baseline allows us to isolate the benefits of including algorithmic structure.

Baseline optimization approaches: In each domain, we compare to expert-designed optimization algorithms found in the literature. In community detection, we compare to “CNM" [CNM04], an agglomerative approach, “Newman", an approach that recursively partitions the graph [New06a], and “SC", which performs spectral clustering [VL07] on the modularity matrix. In facility location, we compare to “greedy", the common heuristic of iteratively selecting the point with greatest marginal improvement in objective value, and “gonzalez" [Gon85], an algorithm which iteratively selects the node furthest from the current set. “gonzalez" attains the optimal 2-approximation for this problem (note that the minmax facility location objective is non-submodular, ruling out the usual $(1 - 1/e)$-approximation).

Datasets: We use several standard graph datasets: cora [SNB$^+$08] (a citation network with 2,708 nodes), citeseer [SNB$^+$08] (a citation network with 3,327 nodes), protein [Col17c] (a protein interaction network with 3,133 nodes), adol [Col17a] (an adolescent social network with 2,539 vertices), and fb [Col17b, LM12] (an online social network with 2,888 nodes). For facility location, we use the largest connected component of the graph (since otherwise distances may be infinite). Cora and citeseer have node features (based on a bag-of-words representation of the document), which were given to all GCN-based methods. For the other datasets, we generated unsupervised node2vec features [GL16] using the training edges.

10.4.1 Results on single graphs

We start out with results for the combined link prediction and optimization problem. Table 10.1 shows the objective value obtained by each approach on the full graph for community detection, with Table 10.2 showing facility location. We focus first on the "Learning + Optimization" column which shows the combined link prediction/optimization task. We use $K = 5$ clusters; $K = 10$ is very similar and may be found in the appendix. ClusterNet outperforms the baselines in nearly all cases, often substantially. GCN-e2e learns to produce nontrivial solutions, often rivaling the other baseline methods. However, the explicit structure used by our approach ClusterNet results in much higher performance.

Interestingly, the two stage approach sometimes performs worse than the train-only baseline which optimizes just based on the training edges (without attempting to learn). This indicates that approaches which attempt to accurately reconstruct the graph can *sometimes* miss qualities which are important for optimization, and in the worst case may simply add noise that overwhelms the signal in the training edges. In order to confirm that the two-stage method learned to make meaningful predictions, in the appendix we give AUC values for each dataset. The average AUC value is 0.7584, indicating that the two-stage model does learn to make nontrivial predictions. However, the small amount of training data (only 40% of edges are observed) prevents it from perfectly reconstructing the true graph. This drives home the point that decision-focused learning methods such as ClusterNet can offer substantial benefits when highly accurate predictions are out of reach even for sophisticated learning methods.

We next examine an optimization-only task where the entire graph is available as

Table 10.3: *Inductive results. "%" is the fraction of test instances for which a method attains top performance (including ties). "Finetune" methods are excluded from this in the "No finetune" section.*

	Community detection					Facility location			
	synthetic		pubmed			synthetic		pubmed	
No finetune	Avg.	%	Avg.	%	No finetune	Avg.	%	Avg.	%
ClusterNet	**0.57**	**26/30**	**0.30**	**7/8**	ClusterNet	**7.90**	**25/30**	7.88	3/8
GCN-e2e	0.26	0/30	0.01	0/8	GCN-e2e	8.63	11/30	8.62	1/8
Train-CNM	0.14	0/30	0.16	1/8	Train-greedy	14.00	0/30	9.50	1/8
Train-Newman	0.24	0/30	0.17	0/8	Train-gonzalez	10.30	2/30	9.38	1/8
Train-SC	0.16	0/30	0.04	0/8	2Stage-greedy	9.60	3/30	10.00	0/8
2Stage-CNM	0.51	0/30	0.24	0/8	2Stage-gonz.	10.00	2/30	**6.88**	**5/8**
2Stage-Newman	0.01	0/30	0.01	0/8	ClstrNet-1train	7.93	12/30	7.88	2/8
2Stage-SC	0.52	4/30	0.15	0/8					
ClstrNet-1train	0.55	0/30	0.25	0/8					
Finetune					Finetune				
ClstrNet-ft	0.60	20/30	0.40	2/8	ClstrNet-ft	8.08	12/30	8.01	3/8
ClstrNet-ft-only	0.60	10/30	0.42	6/8	ClstrNet-ft-only	7.84	16/30	7.76	4/8

input (the "Optimization" column of Tables 10.1 and Table 10.2). This tests ClusterNet's ability to learn to solve combinatorial optimization problems compared to expert-designed algorithms, even when there is no partial information or learning problem in play. We find that ClusterNet is highly competitive, meeting and frequently exceeding the baselines. It is particularly effective for community detection, where we observe large (> 3x) improvements compared to the best baseline on some datasets. At facility location, our method always at least ties the baselines, and frequently improves on them. These experiments provide evidence that our approach, which is automatically specialized during training to optimize on a given graph, can rival and exceed hand-designed algorithms from the literature. The alternate learning approach, GCN-e2e, which is an end-to-end approach that tries to learn to predicts optimization solutions directly from the node features, at best ties the baselines and typically underperforms. This underscores the benefit of including algorithmic structure as a part of the end-to-end architecture.

10.4.2 Generalizing across graphs

Next, we investigate whether our method can learn generalizable strategies for optimization: can we train the model on one set of graphs drawn from some distribution and then apply it to unseen graphs? We consider two graph distributions. First, a synthetic generator introduced by [WOdlHT18], which is based on the spatial preferential attachment model [Bar11] (details in the appendix). We use 20 training graphs, 10 validation, and 30 test. Second, a dataset obtained by splitting the pubmed graph into 20 components using metis [KK98]. We fix 10 training graphs, 2 validation, and 8 test. At test time, only 40% of the edges in each graph are revealed, matching the "Learning + optimization" setup above.

Table 10.3 shows the results. To start out, we do not conduct any fine-tuning to the test graphs, evaluating entirely the generalizability of the learned representations. ClusterNet outperforms all baseline methods on all tasks, except for facility location on pubmed where it places second. We conclude that the learned model successfully generalizes to completely unseen graphs. We next investigate (in the "finetune" section of Table 10.3) whether ClusterNet's performance can be further improved by fine-tuning to the 40% of observed edges for each test graph (treating each test graph as an instance of the link prediction problem from Section 10.4.1, but initializing with the parameters of the model learned over the training graphs). We see that ClusterNet's performance typically improves, indicating that fine-tuning can allow us to extract additional gains if extra training time is available.

Interestingly, *only* fine-tuning (not using the training graphs at all) yields similar performance (the row "ClstrNet-ft-only"). While our earlier results show that ClusterNet can learn generalizable strategies, doing so may not be necessary when there is the opportunity to fine-tune. This allows a trade-off between quality and runtime: without fine-tuning, applying our method at test time requires just a single forward pass, which is extremely efficient. If additional computational cost at test time is acceptable, fine-tuning can be used to improve performance. Complete runtimes for all methods are shown in the appendix. ClusterNet's forward pass (i.e., no fine-tuning) is extremely efficient, requiring at most 0.23 seconds on the largest network, and is *always faster than the baselines* (on identical hardware). Fine-tuning requires longer, on par with the slowest baseline.

We lastly investigate the reason why pretraining provides little to no improvement over only fine-tuning. Essentially, we find that ClusterNet is extremely sample-efficient: using only a single training graph results in nearly as good performance as the full training set (and still better than all of the baselines), as seen in the “ClstrNet-1train" row of Table 10.3. *That is,* ClusterNet *is capable of learning optimization strategies that generalize with strong performance to completely unseen graphs after observing only a single training example.* This underscores the benefits of including algorithmic structure as a part of the architecture, which guides the model towards learning meaningful strategies.

10.5 Conclusion

When machine learning is used to inform decision-making, it is often necessary to incorporate the downstream optimization problem into training. Here, we proposed a new approach to this decision-focused learning problem: include a differentiable solver for a simple proxy to the true, difficult optimization problem and learn a representation that maps the difficult problem to the simpler one. This representation is trained in an entirely automatic way, using the solution quality for the true downstream problem as the loss function. We find that this “middle path" for including algorithmic structure in learning improves over both two-stage approaches, which separate learning and optimization entirely, and purely end-to-end approaches, which use learning to directly predict the optimal solution. Here, we instantiated this framework for a class of graph optimization problems. We hope that future work will explore such ideas for other families of problems, paving the way for flexible and efficient optimization-based structure in deep learning.

Chapter 11

Learning to complement humans

Systems developed via machine learning (ML) are increasingly competent at performing tasks that have traditionally required human expertise, with emerging applications in medicine, law, transportation, scientific discovery, and other disciplines (e.g., [EKN$^+$17, CEW$^+$18, MP19]). To date, engineers have constructed models by optimizing model performance in isolation rather than seeking richer optimizations that consider human-machine teamwork.

Optimizing ML performance in isolation overlooks the common situation where human expertise can contribute complementary perspectives, despite humans having their own limitations, including systematic biases [TK74]. We introduce methods for optimizing team performance, where machines take on some parts of the task and humans others. In an ideal world, the machine would be able to handle all instances itself. For complex domains though, this rarely holds in practice, whether due to limited data or model capacity, outliers, superior perceptual or reasoning abilities of people on a given task, or evidence or context available only to humans. When perfect accuracy is unattainable, the machine should focus its limited capacity on regions of the space where it offers the most benefit (e.g., on cases that are challenging for humans), while pursuing human expertise to handle others. We develop methods aimed at training the ML model to complement the strengths of the human, accounting for the cost of querying an expert. While human-machine teamwork can

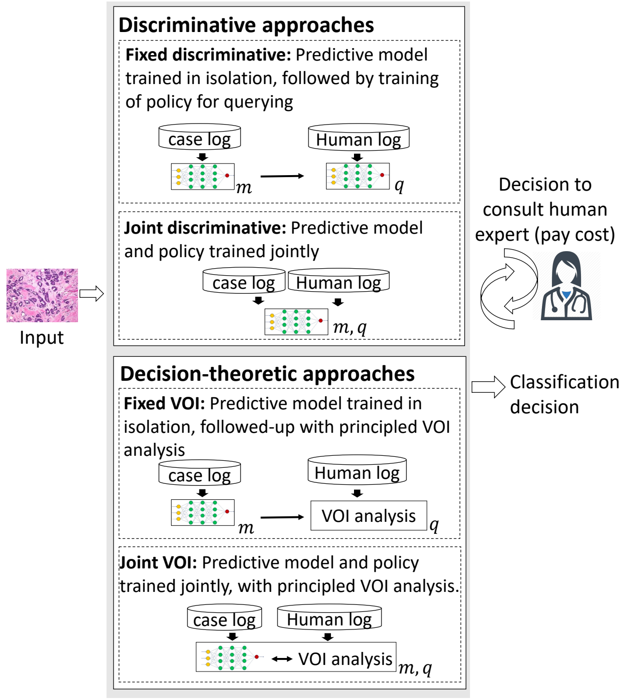

Discriminative approaches
Fixed discriminative: Predictive model trained in isolation, followed by training of policy for querying
case log
Human log
m
q
Joint discriminative: Predictive model and policy trained jointly
case log
Human log
m, q
Input
Decision to consult human expert (pay cost)
Decision-theoretic approaches
Classification decision
Fixed VOI: Predictive model trained in isolation, followed-up with principled VOI analysis
case log
Human log
m
VOI analysis
q
Joint VOI: Predictive model and policy trained jointly, with principled VOI analysis.
case log
Human log
VOI analysis
m, q

Figure 11.1: *Illustration of task and proposed approaches.*

take many forms, we focus here on settings where a machine takes on the tasks of deciding which instances require human input and then fusing machine and human judgments.

Prior work includes systems that determine when to consult humans [HP07, KHH12, RBC^{+}19]. However, the predictive models are still trained to maximize their own, solitary performance, rather than to leverage the distinctive strengths of machines and humans. The latter requires a shift in the learning objective so as to optimize team performance via instance-sensitive decisions about when to seek human input. To our knowledge, the methods we present are the first to optimize human-AI teams by jointly training ML systems together with policies for allocating tasks to human experts versus machines. We make four contributions:

First, we propose a family of approaches to training an ML system for human-machine complementarity as schematized in Figure 11.1. The run-time system combines machine predictions with human input, which may come at additional cost. During training, we use logged human responses to the task to simulate queries to a human. We study both *discriminative* and *decision-theoretic* approaches to optimizing model performance, taking the complementarity of humans and machines into consideration. A baseline approach in either family would first construct an ML model to predict the answer to a given task and then build a policy for deciding when to query the human, taking the predictive model as fixed. We introduce the first generic procedures that operate end-to-end, focused on team performance. With these approaches, we jointly optimize the predictive model and the query policy for team performance, accounting for human-machine complementarities. In the discriminative setting, we introduce a combined loss function that uses a soft relaxation of the query policy for training, along with a technique for making discrete query decisions at run time. In the decision-theoretic setting, we introduce a differentiable surrogate for value of information (VOI) calculations, which allows joint training of the predictive model and the VOI-based query policy through backpropagation. In both cases, joint training focuses the predictive model on instances where the human will not be queried, amplifying complementarity.

Second, we demonstrate the benefits of optimizing for team performance in human-machine teams for two real-world domains of societal importance: scientific discovery (a galaxy classification task) and medical diagnosis (detection of breast cancer metastasis). Via comparative studies, we highlight the importance of guiding learning to optimize the performance of human-machine teams.

Third, we pursue experimental insights about when and how complementarity-focused training provides benefits. We find evidence for two conclusions: First, training for complementarity is most important when the ML model has limited capacity, forcing it to pick parts of the task to focus on. This suggests that an emphasis on team performance is particularly necessary for difficult tasks that machines cannot perfectly master on their own. Second, training for complementarity has larger benefits when there is an asymmetric cost to errors (e.g., false negatives are more costly than false positives). The need to prioritize among potential errors increases the returns of optimizing for team utility.

Fourth, we analyze how our methods distribute instances to the human and machine and how these allocations reflect differences in relative capabilities. We find that humans and machines may make qualitatively different kinds of errors. Moreover, the errors made by the ML model change under joint training as the model places more emphasis on instances that are difficult for humans. Via joint training, human and machine errors become different in structured ways that can be leveraged by the methods to improve team performance.

11.1 Related Work

Previous work shows that human-machine teams can be more effective than either individually [HP07, KHH12], including for medical domains [WKG$^+$16, RBC$^+$19]. However, in some others [TAIK18, ZLB20], potential complementarity has been difficult to leverage.

Sharing our motivation for developing techniques that harness human-machine complementarity, the work by [RBC$^+$19] and [DKGGR20] study when a model should outsource a given instance to a human. [RBC$^+$19] is most closely related to our fixed decision-theoretic algorithm; their approach considers predictive variance for the human and machine at each point to allocate human effort. However, the ML model is always fixed, instead of being trained for complementarity. [DKGGR20] propose a method to select the parameters of a ridge regression model jointly with a set of training instances to allocate to the human. Our work differs in three important ways: (i) they do not train a query policy to allocate new instances at run time, (ii) our methods apply to arbitrary differentiable models (not just ridge regression), (iii) we provide a characterization of why some methods are more or less effective at leveraging complementarity.

Other related work addresses the complementary question of designing ML models as an aid for a human who is charged with making decisions [GHEG19, GC19, HRB$^+$19, LRG$^+$18]. Some of this work emphasizes the need for ML models to account for human reasoning, in particular for humans to learn when to trust the ML model [BNK$^+$19a, BNK$^+$19b], but does not optimize the model for complementarity. We focus on cases were the ML system decides which instances require human input.

11.2 Problem Formulation

We formalize the problem of optimizing human-AI teamwork for predictive tasks. We start with the standard supervised learning setting, predicting labels $y \in \mathcal{Y}$ from features $x \in \mathcal{X}$. We focus on multiclass classification, where $\mathcal{Y}$ is a discrete set, but our methods apply to regression with minor modifications. As is typical, we train a model m with parameters θ, which produces a prediction $\hat{y} = m_\theta(x)$. The difference is that each instance may also be labeled by a human. Our training data contains instances $\{(x, y, h)\}_1^N \sim P$ where $h \in \mathcal{Y}$ is a human's prediction and P is an (unknown) joint distribution. The machine must decide, for each instance, whether to predict on its own or first consult a human expert.

Specifically, the machine learning model first sees x and then decides whether to pay a cost c to observe h. $q_\theta(x)$ denotes the query policy, which outputs 1 when the human is queried and 0 otherwise. The model makes a prediction $\hat{y}$, which may depend on h if $q_\theta(x) = 1$. The team's utility is $u(y, \hat{y})$ if the human is not queried, and $u(y, \hat{y}) - c$ if they are. One choice for the utility is $u(y, \hat{y}) = 1[y = \hat{y}]$ (predictive accuracy), but our framework extends easily to asymmetric weightings of different errors. We aim to maximize out-of-sample utility,

$$\begin{aligned} \mathop{\mathbb{E}}_{(x,y,h)\sim P} \Big[& q_\theta(x)\left(u(y, m_\theta(x, h)) - c\right) \\ & +(1 - q_\theta(x))\left(u(y, m_\theta(x))\right)\Big]. \end{aligned} \tag{11.1}$$

The first term gives the team utility when the human is queried, and the second when they are not. Conventional supervised learning targets only the second term; our formulation includes the query decision, and the impact of the additional information provided by the human, on the team's overall accuracy.

11.3 Approach

A standard approach to optimizing for human-machine teamwork would first train the model in isolation m to predict the labels y given x. Then, m is taken as fixed when constructing the query policy q (as, e.g., in [RBC+19]). We propose an alternate approach: joint training that considers explicitly the relative strengths of the human and machine. We introduce methods for both discriminative and decision-theoretic approaches, and now introduce each family in more detail.

11.3.1 Discriminative Approaches

Discriminative approaches learn functions for m and q which directly map from features to decisions, without building intermediate probabilistic models for the different components of the system. We first introduce a baseline “fixed" method for training a discriminative system and then propose a means to jointly train the model and query policy together for complementarity with people.

Fixed Discriminative Approach

Traditional *fixed discriminative* approaches train a model m in isolation to perform the task, making the assumption that there is no ability to query the human. That is, we train m to optimize $\mathbb{E}_{(x,y)\sim P}[u(y, m_\theta(x))]$ using any number of well-established methods. Then, taking m as fixed, we construct a query policy q by optimizing Equation 11.1.

Joint Discriminative Approach

In distinction to the fixed approach, we present a *joint discriminative* method that trains the ML model m_θ end-to-end with the query policy q_θ so that m_θ can prioritize instances allocated by q_θ to the machine. The goal is to optimize a training surrogate for the team utility in Equation 11.1. In the notation, $m_\theta(x)$ denotes the distribution over classes output by the model, and h gives the one-hot encoding of the human responses.

We propose a differentiable surrogate for Equation 11.1, which can be optimized via stochastic gradient descent whenever the models are themselves differentiable (e.g., neural networks). During training, we will allow $q_\theta(x)$ to take continuous values. This soft relaxation both ensures differentiability and speeds learning by propagating gradient information for both cases (querying and not querying). The most direct relaxation for Equation 11.1 is

$$q_\theta(x)\ell(y, m_\theta(x, h)) + (1 - q_\theta(x))\ell(y, m_\theta(x)) + cq_\theta(x)$$

where ℓ is any standard loss, which may be weighted to capture asymmetries in the utility u. This replaces the potentially discontinuous u with a differentiable loss ℓ defined on soft predictions (probability distributions), along with a penalty scaling c by the query probability $q_\theta(x)$. In experiments, this direct relaxation often produced unstable training; intuitively, the predictions and query policy may be spiky in some regions, giving a rapidly changing training signal. The loss we use is

$$\ell(y, q_\theta(x)m_\theta(x, h) + (1 - q_\theta(x))m_\theta(x)) + cq_\theta(x)$$

which measures the loss of a fractional prediction that combines the human and machine outputs. The combination tends to behave more smoothly, enabling better training. A key feature of this loss is that it allows the predictions $m_\theta(x)$ to focus on instances that rely heavily on the machine. If $q_\theta(x)$ for some x is close to 1, then the loss for x depends only weakly on $m_\theta(x)$, incentivizing m to focus on instances where q is lower instead.

When the human is queried, the general formulation allows $m_\theta(x, h)$ to output a prediction different than the human response h. However, we observe stronger empirical performance using the simplification $m_\theta(x, h) = h$ (though training a separate model for $m_\theta(x, h)$ results in similar qualitative conclusions). Intuitively, often the correct decision after querying is to output h, and including a separate model only adds unnecessary parameters.

For this simplified formalization, we introduce the following run-time query policy: we need a way of converting the fractional q to a 0 or 1 decision (whether to actually query the human). In an idealized setting where the human label was free, the run-time prediction would be $\arg\max\left(q_\theta(x)h + (1 - q_\theta(x))m_\theta(x)\right)$ (i.e., the highest-probability label in the combined prediction). A naive thresholding scheme would query the human if $q_\theta(x) > 0.5$ (or another fixed value). However, we can approximate the idealized prediction more closely by incorporating a measure of the ML model's confidence, $\max\left(m_\theta(x)\right)$. Specifically, we query the human if

$$(1 - q_\theta(x))\max\left(m_\theta(x)\right) < q_\theta(x)$$

which results in a query if $q_\theta(x)$ is sufficiently high, *or* the model is sufficiently uncertain. More formally, when this condition holds, the idealized prediction must align with h since $\max\left(q_\theta(x)h\right) > \max\left((1 - q_\theta(x))m_\theta(x)\right)$.

11.3.2 Decision-Theoretic Approaches

A decision-theoretic approach to human-machine teams, as described in [KHH12], is to construct probabilistic models for both the ML task and the human response. This allows a follow-up step that calculates the expected value of information for querying the human.

Fixed Value of Information Approach

The *fixed value of information (VOI)* method trains three probabilistic models. $p_\alpha(y|x)$ models the distribution of the label given the features, $p_\beta(h|x)$, the human response given the features, and $p_\gamma(y|h, x)$, the label given both the features and the human response. α, β, γ are model parameters. Each model is individually trained to fit its intended target. In our implementation, we use neural networks trained via gradient descent, followed by a sigmoid calibrator trained using the Platt method [Pla99, NMC05]. Calibration is necessary for the predicted probabilities to give meaningful expected utilities.

At execution time, we use these models to estimate the value of querying the human. The estimated expected utility of the ML model without querying the human is

$$u_{\mathrm{nq}} = \max_{\hat{y} \in \mathcal{Y}} \left(\sum_{y \in \mathcal{Y}} p_\alpha(y|x) u(\hat{y}, y) \right)$$

i.e., the value of the prediction with highest expected utility according to $p_\alpha(y|x)$. Before querying the human, we cannot know the value of h and hence the post-query distribution $p_\gamma(y|x,h)$ is also unknown. However, we can estimate the expected utility by averaging over $p_\beta(h|x)$,

$$u_{\mathrm{q}} = \mathop{\mathbb{E}}_{h \sim p_\beta(h|x)} \left[\max_{\hat{y} \in \mathcal{Y}} \left(\sum_{y \in \mathcal{Y}} p_\gamma(y|x,h) u(\hat{y}, y) \right) \right] - c$$

and then query the human whenever $u_{\mathrm{q}} > u_{\mathrm{nq}}$.

Joint Value of Information Approach

We propose a new decision-theoretic method, which we refer to as a *joint VOI* approach, that optimizes the utility of the combined system end-to-end, instead of training the best probabilistic model for each individual component. Retaining the structure of the fixed VOI system can be viewed as an inductive bias which allows the model to start from well-founded probabilistic reasoning and then to be fine-tuned for complementarity. To benefit from this inductive bias, we instantiate each of the probabilistic models p_α, p_β, and p_γ with a neural network followed by a Platt calibration layer, just like the fixed VOI approach. However, with joint VOI all of the neural network parameters are trained together via an end-to-end loss, which is grounded in the VOI calculation. We update the calibration layer every t steps to maintain well-calibrated probabilities.

Algorithm 13 outlines joint VOI training. We optimize a surrogate for team utility via stochastic gradient descent, so each iteration first samples a minibatch of data points. For each point, we simulate a differentiable VOI calculation which draws on soft versions of the team's utility if the human were queried (u_{q}) and if the human were not queried (u_{nq}), along with the cost to query. Specifically, line 4 computes $u_{\mathrm{nq}}(\hat{y})$, the expected utility of predicting $\hat{y}$ (according to p_α) when the human is not queried. Line 5 takes a softmax over all potential

Algorithm 13 Joint VOI training

```
1: for $T$ iterations do
2:     Sample a minibatch $B \subseteq [n]$
3:     for $i \in B$ do
4:         for $\hat{y} \in \mathcal{Y}$ do
5:             $u_{\text{nq}}(\hat{y}) = \sum_{y \in \mathcal{Y}} p_\alpha(y|x_i) u(\hat{y}, y)$
6:         end for
7:         $u_{\text{nq}} = \sum_{\hat{y} \in \mathcal{Y}} \frac{u_{\text{nq}}(\hat{y}) \exp(u_{\text{nq}}(\hat{y}))}{\sum_{y' \in \mathcal{Y}} \exp(u_{\text{nq}}(y'))}$
8:         for $\hat{y} \in \mathcal{Y}$ do
9:             $u_{\text{q}}(\hat{y}, h) = \sum_{y \in \mathcal{Y}} p_\gamma(y|x_i, h) u(\hat{y}, y)$
10:        end for
11:        $u_{\text{q}} = \sum_h p_\beta(h|x) \sum_{\hat{y}} \frac{u_{\text{q}}(\hat{y},h) \exp(u_{\text{q}}(\hat{y},h))}{\sum_{y' \in \mathcal{Y}} \exp(u_{\text{q}}(y',h))}$
12:        $q = \frac{\exp(u_{\text{q}})}{\exp(u_{\text{q}}) + \exp(u_{\text{nq}})}$
13:        $\ell^i_{\text{combined}} = \ell(q\, p_\gamma(\cdot|x_i, h_i)$
14:            $+(1-q) p_\alpha(\cdot|x_i)) + qc$
15:    end for
16:    Backpropagate $\frac{1}{|B|} \sum_{i \in B} \ell^i_{\text{combined}}$
17:    Every $t$ iterations: update calibrators
18: end for
```

$\hat{y}$ in order to achieve a differentiable approximation to the best achievable expected utility without a query. Similarly, line 6 computes the expected utility $u_{\text{q}}(\hat{y}, h)$ of predicting $\hat{y}$ supposing that the human was queried and responded with h. Line 7 takes a softmax over $\hat{y}$ for each fixed h (the inner sum), and then an expectation over $h \sim p_\beta$ (the outer sum). This approximates the expected utility of observing h and then predicting the best $\hat{y}$ given the observation. Line 8 makes a soft query decision via a softmax over u_{nq} and u_{q}.

Using the output (query decision and prediction) of the differentiable VOI calculation, we compute a team loss ℓ_{combined}, which uses the same form as in the joint discriminative model. We average this loss over the minibatch and backpropagate it to update the predictive models. During this process, we freeze the parameters of the calibration layers of the models. The calibration layers are updated using the Platt procedure every t steps in order to ensure that the model remains well-calibrated even under end-to-end training.

Compared to the fixed model, the joint model uses well-calibrated models to calculate the expected utility of a query. However, it encourages these models to fit most carefully to parts of the space that the are best handled by the machine, and obtains human expertise for others.

11.4 Experiments

We conducted experiments in two real-world domains to explore opportunities for human-machine complementarity and methods to best leverage the complementarity.

11.4.1 Domains

We first explore a scientific discovery task from the Galaxy Zoo project. Here, citizen scientists label images of galaxies as one of five classes to help understand the distribution of galaxies and their evolution. We use 10,000 instances for training and 4,000 for testing. Each instance contains visual features which previous work extracted from the dataset [LSS+08, KHH12] for x. The human response h is the label assigned by a single volunteer (who may make mistakes), while the ground truth y is the consensus over many (> 30) volunteers.

We next study the medical diagnosis task of detecting breast cancer metastasis in lymph node tissue sections from women with a history of breast cancer. We use data from the CAMELYON16 challenge [BVVD+17]. Each instance contains a whole-slide image of a lymph node section. Each image was labeled by an expert pathologist with unlimited time, providing the ground truth y. It was also labeled by a panel of pathologists under realistic time pressure whose diagnoses contain errors; we sample h from the panel responses.

The dataset consists of 127 images. There are also 270 images without panel responses, with which we pretrain the ML models. To develop our models, we follow common practice from high-scoring competition entries (our implementation is based on [Vek16]). We first train a convolutional network (Inception-v3 [SVI+16]) to predict whether cancer is present in 256×256 pixel patches sampled from the larger whole-slide images. Then, we use Inception-v3 to predict the probability of cancer in each patch, giving a probability heatmap for each slide. We extract visual features from the heatmap (e.g., size of the largest cancer region, eccentricity of the enclosing ellipse, etc). These features are the input x into the human-AI

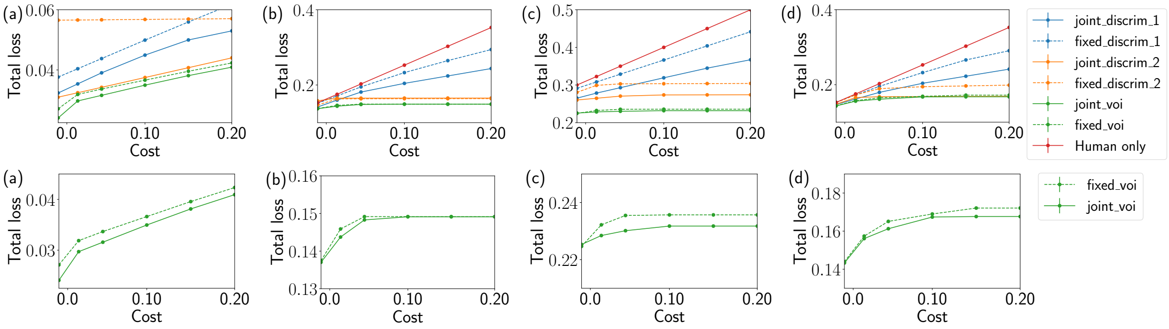


Figure 11.2: *Total loss (classification error + cost of queries to human) as a function of the cost of a human query. Top row: All approaches. Bottom row: Zooming in on decision-theoretic approaches. (a) Galaxy Zoo (b) CAMELYON16 (c) CAMELYON16, doubling the cost of false negatives. (d) CAMELYON 16, reducing hidden layers to 20 neurons (from 50). We omit the "human only" baseline for Galaxy Zoo since it has over twice the loss of any other method. All differences between fixed and joint models are statistically significant for Galaxy Zoo, and on the CAMELYON16 task for the discriminative models (Student t-test, $p < 10^{-3}$). Due to the small size of the CAMELYON16 dataset (127 samples), not all VOI comparisons are statistically significant, but the larger differences approach significance (e.g, $p < 0.15$ for the point with largest difference in each of Figures 11.2(c-d)).*

task. This workflow produced the highest-scoring competition entries, ensuring we compare using a state-of-the-art ML method.

11.4.2 Models

We compare each of the four approaches introduced earlier: fixed versus joint discriminative and VOI models. All use neural networks with ReLU activations and dropout ($p = 0.2$). Our experiments vary the number of layers and hidden units to examine the impact of model capacity. We also show a "Human only" baseline that always queries the human and outputs their response h.

11.4.3 Results

We first examine the performance of these methods for the two tasks. Fig 11.2 shows each method's total loss (combining classification error and the cost of human queries). For each model, the dashed line shows the fixed version and the solid line denotes joint. For the joint models, we train the model under a range of weightings of classification loss vs query cost, and each x-axis point selects the version with lowest total loss for that cost. We show

Table 11.1: *Comparison of joint and fixed VOI models across a range of settings. "Layers" gives the number of layers used in the predictive models, "Hidden," the number of hidden units, and "% diff.," the percentage improvement of the joint over fixed model (given as the min, average, and max improvement in loss over costs from 0 to 0.2).*

Task	Layers	Hidden	% diff. (min / avg / max)
GZ	1	-	21.8 / 38.9 / 73.3
GZ	2	50	2.13 / 9.02 / 14.0
GZ	2	100	-1.05 / 8.89 / 13.5
CAM.	1	-	-3.10 / 4.51 /10.4
CAM. (asym.)	1	-	-1.26 / 5.13 / 15.2
CAM.	2	20	0.30 / 1.82 / 2.65
CAM. (asym.)	2	20	-0.80 / 1.91 / 4.85
CAM.	2	50	0.00 / 0.03 / 2.31
CAM. (asym.)	2	50	-0.67 / 1.70 / 2.28

discriminative models with one- and two-layer networks. Because the one- and two-layer VOI models have fairly different losses (which compresses the plots), we only show two layers. Table 11.1 gives results for all VOI configurations.

The joint models, which optimize for complementarity, uniformly outperform or tie their fixed counterparts. For Galaxy Zoo, joint training leads to 21-73% reduction in loss for the one-layer VOI models and 10-15% reduction in loss for two-layer VOI. The reductions are 10-15% and 29% for the one and two layer discriminative models respectively. For CAMELYON16, joint training improves the one-layer discriminative model by up to 20% and the one-layer VOI model by up to 10%. For deeper models, joint training ties the fixed approach or makes modest improvements (around 2% reduction in loss). Next, we vary the

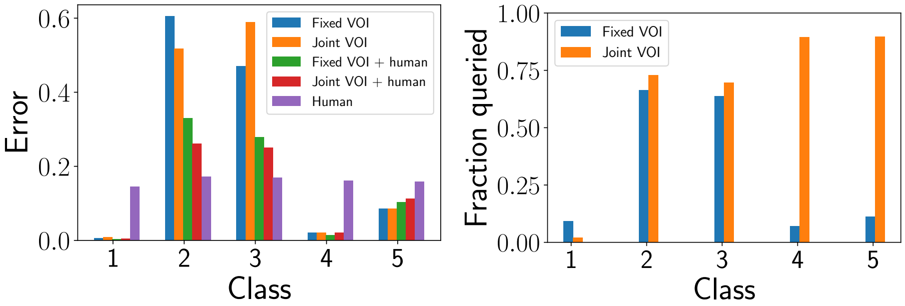


Figure 11.3: *Detailed analysis on Galaxy Zoo task. Left: Error rate of machine versus human models for each class. Right: Fraction of instances in each class queried by the machine.*

problem setting to explore the factors that influence the benefits of joint training. First, we vary the capacity of the models, as measured by the number of hidden units. Figures 11.2b and 11.2d compare the total loss of different approaches when hidden unit sizes is reduced from 50 to 20. Table 11.1 examines the effect of model capacity on the VOI-based approaches. **Overall, joint training provides larger benefits with limited model capacity**. For example, for CAMELYON16, the reduction of loss from joint training for discriminative approaches is up to 15% when hidden units are reduced to 20, whereas for the 50 neuron condition the two discriminative approaches are tied (two-layer models). This dovetails with earlier results that showed larger gains for shallower models. Essentially, a lower-capacity model has more potential bias (since it represents less complex hypotheses which cannot fit the ground truth as closely). This makes aligning the training process with team performance more important because some errors are inevitable; joint training helps the model focus its limited predictive ability on the most important regions. In theory, sufficiently large datasets would let us train arbitrarily complicated models that perfectly recover the ground truth, rendering simple models unnecessary. In practice, limited data requires us to prevent overfitting by restricting model capacity; maximizing the performance of simple models is valuable in many tasks.

The second experimental modification introduces an asymmetric loss for CAMELYON16: motivated by high cost of missing diagnoses in many areas of medicine (such as failing to recognize the recurrence of illness in patients with a history of cancer), we weight false negatives twice as heavily as false positives. **The gaps between the fixed and joint models grow under asymmetric costs**. For example, in Figure 11.2(b) (equal costs), the two-layer model performance of discriminative or VOI approaches were previously tied. In Figure 11.2(c) (asymmetric costs), the joint approaches now outperform their fixed counterparts by up to 10% (discriminative family) and 4.8% (VOI). Optimizing combined team performance is especially helpful when it is necessary to prioritize between potential errors.

Finally, we examine how joint training influences the capabilities of the ML system in relation to those of humans. We start with the Galaxy Zoo task (two-layer models, 50 hidden units, cost = 0.1). Figure 11.3 shows the error rates of the fixed and joint VOI models for

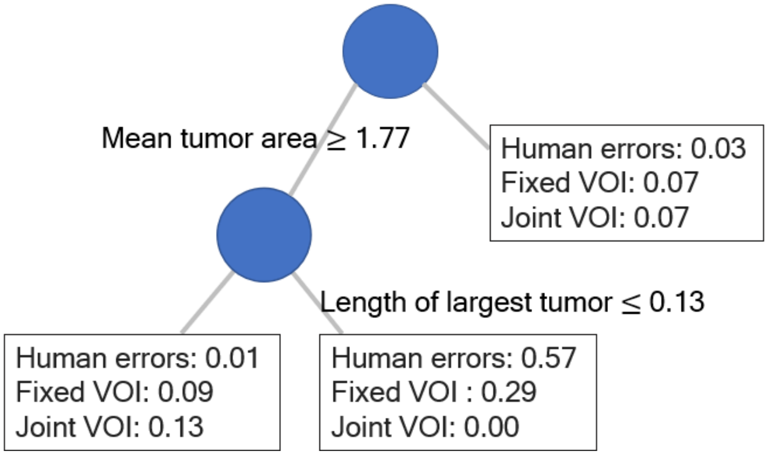


Figure 11.4: *Error rates of humans and decision-theoretic approaches for prominent feature regions of CAMELYON16.*

each of the five classes when acting alone and when paired with people. Both the error rates of the two approaches on classes 2 and 3, and the way they query humans show differences, indicating that joint optimization changes how the ML system learns and makes decisions. The joint approach makes more queries to humans for classes that are hard for the machine and less for class 1, which is easy for the machine (note that class 1 accounts for over 70% of instances). This behavior improves team performance on classes 2 and 3 without diminishing performance on class 1. For class 3, the error rate of the joint VOI model is higher than its counterpart when acting alone, but lower when combined with the human, a reduction in loss that cannot be simply explained by the marginal increase in human queries. This shows that the joint model can harness human input more effectively by discovering input spaces within individual classes where the benefits of complementarity can be realized, and also that joint training encourages the model to manage tradeoffs in accuracy to leverage the ability to query the human.

We observe similar behavior for CAMELYON16. Here, we find clear structure in the human errors, uncovered by fitting the decision tree shown in Figure 11.4 (for the uniform-cost task with two-layer models and 50 hidden units). Over 68% of human errors are concentrated in a region containing just 10% of instances, identified using two features. For each leaf, we show the error rate of the human, the fixed VOI model, and the joint

VOI model. The joint model prioritizes the region that contains most of the human errors, improving from the 0.29 error rate of the fixed model to perfect accuracy. This comes at the cost of increased errors in the far-left leaf; however, in this region the human is almost perfectly accurate. Overall, this tradeoff made by the joint optimization leads to a 2% overall reduction in loss. In other words, the distribution of errors incurred by the joint model shifts to complement the strengths and weaknesses of the human.

11.5 Conclusion and Future Work

We studied how ML systems can be optimized to complement humans via the use of discriminative and decision-theoretic modeling methodologies. We evaluated the proposed approaches by performing experiments with two real-world tasks and analyzed the problem characteristics that lead to higher benefits from training focused on leveraging human-machine complementarity. The methods presented are aimed at optimizing the expected value of human-machine teamwork by responding to the shortcomings of ML systems, as well as the capabilities and blind spots of humans. With this framing, we explored the relationship between model capacity, asymmetric costs and ML-human complementarity. We see opportunities for studying additional aspects of human-machine complementarity across different settings. Directions include optimization of team performance when interactions between humans and machines extend beyond querying people for answers, such as settings with more complex, interleaved interactions and with different levels of human initiative and machine autonomy. We hope that the methods and results presented will stimulate further pursuit of opportunities for leveraging the complementarity of people and machines.

Part IV

Inference and epidemics

Chapter 12

Modeling and inference for population-specific COVID dynamics

Since December 2019, the COVID-19 pandemic – propagated by the novel coronavirus, SARS-CoV2 – has resulted in significant morbidity and mortality [BQNS$^+$20], and key factors such as existing comorbidities and age play a role in an increased risk of mortality [ZYD$^+$20]. Epidemiological studies have provided significant insights into the disease and its transmission dynamics to date [XdGM$^+$20, RA20, LPC$^+$20, KRD$^+$20]. However, national and regional governments have implemented broad-reaching policies in response to rising case counts and stressed healthcare systems and tailoring these polices based on an understanding of how population-specific demography impacts outbreak dynamics will be vital. Previous modeling studies have not incorporated the rich set of household demographic features needed to address such questions.

This study develops a stochastic agent-based model for SARS-CoV2 transmission which accounts for distributions of age, household types, comorbidities, and contact between different age groups in a given population (Fig. 12.1). Our model accounts for both within-household contact (simulated via household distributions taken from census data) and out-of-household contact using age-stratified, country-specific estimated contact matrices [PCJ17]. We instantiate the model for Hubei, China; Lombardy, Italy; and New York City,

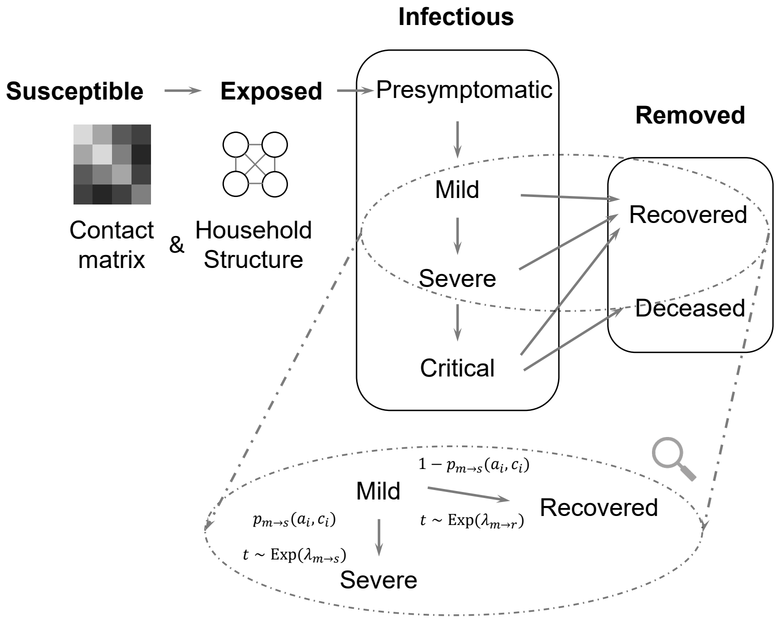


Figure 12.1: *We use a modified SEIR model, where the infectious states are subdivided into levels of disease severity. The transitions are probabilistic and there is a time lag for transitioning between states. For example, the magnified section shows the details of transitions between mild, recovered, and severe states. Each arrow consists of the probability of transition (e.g., $p_{m \to s}(a_i, c_i)$ denotes the probability of progressing from mild to severe) as well as the associated time lag (e.g., the time t for progression from mild to severe is drawn from an exponential distribution with mean $\lambda_{m \to s}$). a_i and c_i denote the age and set of comorbidities for the infected individual i.*

United States, developing a Bayesian inference strategy for estimating the distribution of unknown parameters using data on reported deaths during the first wave of the epidemic in each location. This enables us to uncover differences in the initial dynamics of the epidemic in each location. We also examine how transmission by particular age groups contributes to infections and deaths in each location, allowing us to compare the efficacy of efforts to reduce transmission across said groups. There is large between-population variation in the role played by any individual age group. However, across populations, both infections and deaths are substantially reduced by a combination of population-wide physical distancing and "salutary sheltering" – a term we coin here to describe individuals who shelter in place irrespective of their exposure or infectious state – by half the individuals in a specific age group, without the need for potentially untenable policies such as indefinite sheltering of all older adults.

12.1 Methods

This section provides an overview of our modeling and inference strategy. Additional details may be found in the appendix.

12.1.1 Model

We develop an agent-based model for COVID-19 spread which accounts for the distributions of age, household types, comorbidities, and contact between different age groups in a given population. The model follows a *susceptible-exposed-infectious-removed (SEIR)* template [VdDLM99, BKO15]. Specifically, we simulate a population of n agents (or individuals), each with an age a_i, a set of comorbidities c_i, and a household (a set of other agents). We stratify age into ten-year intervals and incorporate hypertension and diabetes as comorbidities due to their worldwide prevalence [RAA+18] and association with higher risk of in-hospital death for COVID-19 patients [ZYD+20]. However, our model can be expanded to include other comorbidities of interest in the future. The specific procedure we use to sample agents from the joint distribution of age, household structures, and comorbidities may be found in the appendix. We focus on modeling household contacts in particular detail because of the documented frequency of within-household transmission [KRD+20] and the previous suggestion that patterns of contact within the household may play a large role in shaping the epidemic [EPBV20]. It is important to acknowledge that available data sources only suffice to model the joint distribution of age and household structure, whereas sampled comorbidities are conditioned only on the age of each agent (ignoring potential correlations between the comorbidity statuses of household members). However, this procedure still captures the marginal distribution of comorbidities over age in the population and hence the aggregate impact of COVID-19 on said population.

The disease is transmitted over a contact structure, which is divided into in-household and out-of-household groups. Each agent has a household consisting of a set of other agents (see the appendix for details on how households are generated using country-specific census information). Individuals infect members of their households at a higher rate than out-of-household agents. We model out-of-household transmission using country-specific estimated contact matrices [PCJ17]. These matrices state the mean number of daily contacts an individual of a particular age stratum has with individuals from each of the other age strata. We assume demographics and contact patterns in each location are well-approximated by country-level data.

The model iterates over a series of discrete time steps, each representing a single day, from a starting time t_0 to an end time T. There are two main components to each time step: disease progression and new infections. The progression component is modeled by drawing two random variables for each individual each time they change severity levels (e.g., on entering the mild state). The first random variable is Bernoulli and indicates whether the individual will recover or progress to the next severity level. The second variable represents the amount of time until progression to the next severity level. We use exponential distributions for almost all time-to-event distributions, a common choice in the absence of specific distributional information [All10, Col15]. The exception is the incubation time between presymptomatic and mild states, where more specific information is available; here, we use a log-normal distribution based on estimates by [LGB+20]. The appendix (Table S1) summarizes all distributions and their parameters and describes how we estimate age- and comorbidity-dependent severity progression. The “mild" state in our model encompasses the entire gradient of individuals who may have specific symptoms of COVID-19 but do not warrant hospitalization, those with paucisymptomatic or subclinical infections, and those with no detectable symptoms at all. Our model does not currently distinguish between the transmissibility of individuals in any of these states but could be extended with a more detailed characterization.

In the new infections component, infected individuals infect each of their household members with probability p_h at each time step. p_h is calibrated so that the total probability of infecting a household member before either isolation or recovery matches the estimated secondary attack rate for household members of COVID-19 patients (i.e., the average fraction of household members infected) [LEK20]. Infected individuals draw outside-of-household contacts from the general population using the country-specific contact matrix. For an infected individual of age group i, we sample $w^s_{ij} \sim \text{Poisson}(M^s_{ij})$ contacts for each age group j and setting s where M^s is the country-specific contact matrix for setting s. We include contacts in work, school, and community settings. Poisson distributions are a standard choice for modeling contact distributions [PCJ17]. Then, we sample w^s_{ij} contacts of age j uniformly with replacement, and each contact is infected with the probability p_{inf}, the probability of infection given contact. There is evidence to suggest that the probability of infection is higher for an older individual than younger given the same exposure [ZLL+20], consistent with decline in immune function with age. We adjust for this by letting the probability of infection be βp_{inf} when the exposed individual is over the age of 60, for $\beta > 1$. β is calibrated to match the fraction of deaths in China attributed to individuals over the age of 60, resulting in a value of 1.25. This is consistent with the relationship between age and attack rate amongst close contacts of a confirmed case reported by [ZLL+20], where the increase in risk of infection for a contact over 65 years old was estimated in the range 1.12–1.92.

12.1.2 Inference of posterior distributions

We infer unknown model parameters and states in a Bayesian framework. This entails placing a prior distribution over the unknown parameters, and then specifying a likelihood function for the observable data, the time series of deaths reported in a location. We posit the following generative model for the observed deaths:

$$p_{\text{inf}}, d_{\text{mult}}, t_0 \sim \mathcal{U}$$

$$d_1....d_T \sim ABM(p_{\text{inf}}, d_{\text{mult}}, t_0)$$

$$o_t \sim \text{NegativeBinomial}(d_t, \sigma^2_{obs}) \; t = 1...T$$

where $\mathcal{U}$ denotes a joint uniform prior, ABM denotes a draw from the stochastic agent-based dynamics, $d_1...d_T$ are the time series output by the simulation, and $o_1...o_T$ are the number of deaths observed on the corresponding dates. We model the observations as drawn from a negative binomial distribution (appropriate for overdispersed count data) with dispersion parameter σ^2_{obs}. We separately estimated σ^2_{obs} by fitting an autoregressive negative binomial regression to the observed counts using the R package tscount [LFF15]. The negative binomial observation model was strongly preferred to a Poisson model (see Table S2 with AIC values). Together, the likelihood function is given by

$$\mathcal{L}(p_{\text{inf}}, d_{\text{mult}}, t_0, d_1...d_T) = \prod_{t=1}^{T} \Pr\left[o_t | d_t, \sigma^2_{obs}\right].$$

To obtain the posterior distribution, we use Latin hypercube sampling to draw many (10-80 thousand per location, depending on the size of the prior ranges) samples from the joint uniform prior over $p_{\text{inf}}, d_{\text{mult}}$ and t_0, and then sample the latent variables $d_1...d_T$ at each combination of parameters. We compute the likelihood for the full sample (including the latent variables). This allows us to use importance sampling to resample values of $(p_{\text{inf}}, d_{\text{mult}}, t_0, d_1...d_T)$ according to the posterior distribution. Finally, we marginalize out $d_1...d_T$ to obtain the posterior over the parameters $p_{\text{inf}}, d_{\text{mult}}, t_0$, along with unobservable state variables of the simulation such as the number of infected individuals at each step.

12.2 Results

12.2.1 Inferring differences in dynamics between populations

Using our model, we estimate posterior distributions over unobserved quantities which characterize the dynamics of the epidemic in a particular location. This section presents estimates for two quantities: first, the basic reproduction number r_0, and second, the rate at which infections are documented. Neither quantity is directly observable in the data due to substantial underdocumentation of infections; however, these estimates are needed to characterize the scope of the outbreak in a particular location, the degree to which existing testing strategies capture new infections, and the rate at which infections are expected to increase in the absence of any intervention. These findings are critical to formulate policy interventions that are tailored to the outbreak as it evolves in a given population. We start by providing a brief overview of our inference strategy and model validation and then present the main estimates.

There are four model parameters for which values are not precisely estimated in the literature. Each such parameter is instead drawn from a prior distribution. First is p_{inf}, the probability of infection given contact with an infected individual. This determines the level of transmissibility of the disease. Second is t_0, the start time of the infection, which is not precisely characterized in most locations and has an impact due to rapid doubling times. Third is a parameter d_{mult}, which accounts for differences in mortality rates between locations that are *not* captured by demographic factors in the model (e.g., the impact of variation in health system capacities). d_{mult} is a multiplier to the baseline mortality rate from [VOD+20] and is applied uniformly across age groups. We also include an age-specific multiplier to the mortality rate for individuals over 60 in Lombardy, which is calibrated independently of the other parameters to match the fraction of deaths attributed to the 60+ age group (which is significantly higher in Lombardy than the other two locations [VOD+20, oHH20, ORB20]). Further discussion of the age-specific distribution of deaths can be found in the appendix. Fourth is δ_c, the reduction in person-to-person contact after mobility restrictions were imposed in each location. Following mobility restrictions, the expected number of contacts between agents in any two age groups outside the household is reduced to δ_c times its starting value. For Hubei, we fix this parameter using a post-lockdown contact survey [ZLL+20]. For Lombardy and New York City, post-lockdown surveys are not available and so we estimate δ_c within the Bayesian framework. Details of the prior distributions and the modeled scenario in each location can be found in the appendix.

By conditioning on the observed time series of deaths, we obtain a joint posterior distribution over both the unobserved model states, such as the number of people infected at each time step, as well as the three unknown parameters. We use reported deaths because they are believed to be better documented than infections and perform a sensitivity analysis to account for possible underdocumentation of deaths [KLSK20, MBF$^+$20]. Fig. 12.2 shows that the model closely reproduces the observed time series of deaths in each location. In the appendix, we also perform out-of-sample validation by fitting the model using a portion of the time series and assessing the accuracy of the predictive posterior distribution on data that was not used to fit the model.

The left column of Fig. 12.3 shows the posterior distribution over r_0 in each location. Substantial differences are evident between the three locations. The posterior median is 2.23 in Hubei (90% credible interval: 2.10–2.37), 2.95 in Lombardy (2.80–3.19), and 3.20 in New York City (2.71–3.93). The estimates for Hubei fall within the range of a number of existing estimates [MM20], while the interval for Lombardy is similar to the interval 2.9–3.2 estimated by previous work [GPA$^+$20]. The estimated r_0 for New York City is larger than either Hubei or Lombardy. To our knowledge, this estimate constitutes the first r_0 assessment in the literature specifically for New York City. The relative ranking of r_0 for the three populations is not impacted by a sensitivity analysis for underreporting of deaths, shown in Fig. 12.3. Death totals from Hubei have been substantially revised upwards to correct for underreporting in the early stages of the epidemic [Bri20a], but such corrections are either unavailable or rapidly evolving for Lombardy and New York City. Our sensitivity analysis assumes that deaths in Lombardy and New York City are twice what was reported, consistent with preliminary investigations of excess mortality data [KLSK20, MBF$^+$20]. In this scenario, the posterior median value of r_0 rises slightly to 3.12 in Lombardy and remains constant (at 3.20) in New York City. However, the estimated value of δ_c for each location rises sharply, indicating that the model explains increased deaths in this scenario via the possibility of less severe contact reductions during lockdown.

The right column of Fig. 12.3 shows the posterior distribution over the fraction of infections that were documented in each location (obtained by dividing the number of confirmed cases in each location by the number of infections in the simulation under each sample from the posterior). Documentation rates are uniformly low, indicating undocumented infections in all locations; however, we estimate lower documentation in

Lombardy (90% credible interval: 5.1–6.0%) than in either New York City (5.4–12.7%) or Hubei (6.4–12.1%). Documentation rates are substantially lower when assuming twice the reported deaths in Lombardy and New York City (Fig. 12.3, bottom row).

Although we estimate a substantial number of undocumented infections, all locations remain potentially vulnerable to second-wave outbreaks, with the median percentage of the population infected at 1.3% in Hubei, 13.8% in Lombardy, and 22.0% in New York City. Note that in Hubei, our estimate is for the entire province of Hubei, with a population of 58.5 million people, including – but not limited to – the city of Wuhan. Recent serological surveys have estimated 25% of the population previously infected in New York City [Gov20], consistent with our distribution. When assuming that deaths are underreported by a factor of two in Lombardy and New York City, the median percentage infected is 28.2% in Lombardy and 38.7% in New York City[1]. Overall, our estimates for r_0 and the remaining population of susceptible individuals indicate that Hubei, Lombardy, and New York City could experience new outbreaks in the absence of continued interventions to reduce transmission. Despite this, between-population differences remain substantial; Hubei, Lombardy, and New York City have each had distinct experiences with COVID-19 that must be considered with respect to future policy responses.

12.2.2 Containment Policies: Salutary Sheltering and Physical Distancing

Various interventions – from complete lockdown to physical distancing recommendations – have been implemented worldwide in response to COVID-19. Within these are a range of alternatives. For example, a government could encourage some percentage of a given age group to remain sheltered in place, while the rest of the population could continue in-person work and social activities. Age-specific policies are particularly relevant because

[1] Of note, even in a scenario with substantially more deaths than documented, it is possible for the fraction infected to be lower than these estimates. Our model's contact patterns capture the general population, but there is the potential for excess deaths to occur disproportionately in high-risk settings with anomalous contact patterns (e.g., reports have linked a large number of deaths to elder care facilities [YLIS20]). In such circumstances, higher total deaths would not necessarily indicate a substantial increase in the fraction of the entire population infected.

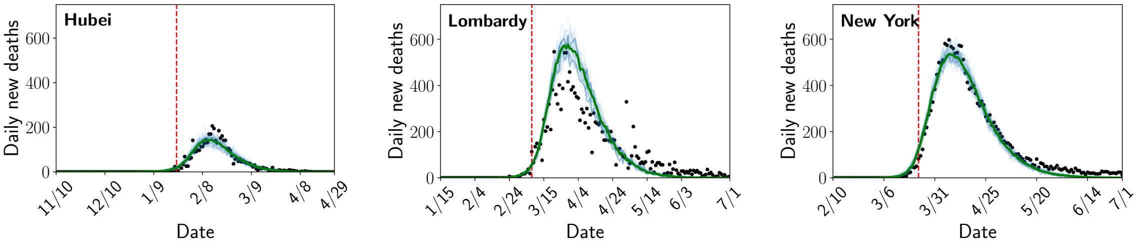


Figure 12.2: *Posterior distribution over the number of deaths each day compared to the number of reported deaths. Light blue lines are individual samples from the posterior, green is the median, and the black dots are the number of reported deaths. The red dashed line represents the start of modeled contact reductions in each location.*

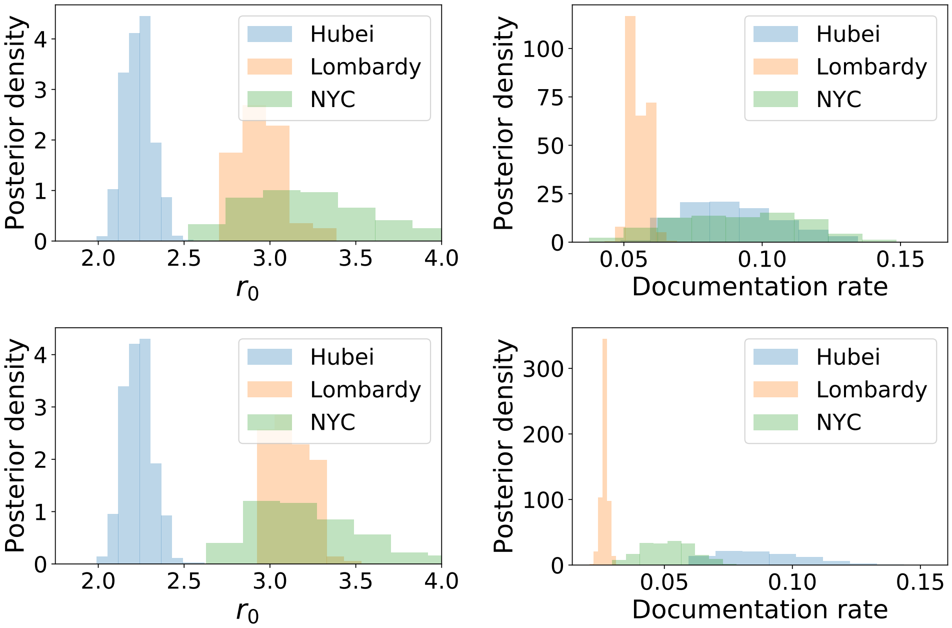


Figure 12.3: *Posterior distribution over r_0 and the fraction of infections documented in each location. Top row: conditioning on reported deaths. Bottom row: conditioning on deaths in New York City and Lombardy being twice what was reported.*

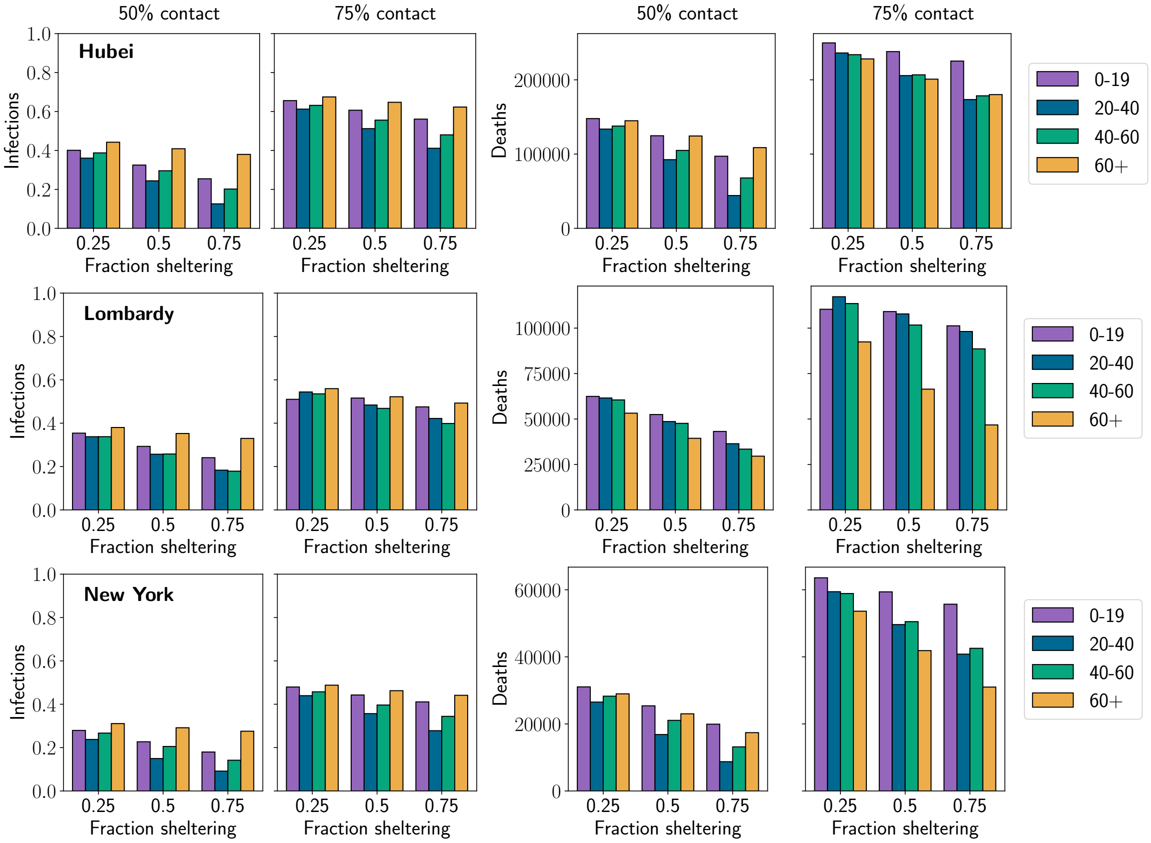


Figure 12.4: *Number of new infections and new deaths in second-wave outbreak scenarios for each location. Each column shows a different level of physical distancing by the population as a whole, where contacts between all age groups are reduced to the given percentage of their starting value. The x-axis within each plot shows the result when the given fraction of a single age group shelters at home (in addition to physical distancing by the rest of the population). The result of this combination of sheltering and distancing is represented by a bar, where the color of the bar indicates the age group which engaged in sheltering (see legend). The height of the bar gives the total number of infections or deaths in the population in that scenario. Each row gives the results for a single location, where the first two plots show the fraction of the population which is newly infected in the second wave, and the next two plots show the number of new deaths which occur.*

they have already been employed in some countries (e.g., US CDC recommendations that people above 65 years old shelter in place [fDCP20a]) and because older age groups are more likely to be able to telecommute, at least in the US [MRL12, oLS19].

Here, we investigate to what extent a second-wave outbreak in each of our three locations of interest can be mitigated by encouraging a single age group to engage in salutary sheltering or whether the entire population must also be asked to adopt physical distancing. We compare scenarios that combine varying levels of two different interventions: (1) salutary sheltering by a given fraction of a single age group modeled by eliminating all outside-of-household contact for agents who engage in sheltering; and (2) physical distancing by the population as a whole, modeled by reducing the expected number of outside-of-household contacts between all agents (who are not engaging in salutary sheltering) to a given percentage of their original value. While this case study applies to Hubei, Lombardy, and New York City, it could be extended to other locations using population-specific demographic data as well. The appendix includes details of all experiments described along with sensitivity analyses where the impact of physical distancing is further varied and where the population begins in a completely susceptible state.

Fig. 12.4 shows the number of new infections or deaths in each location during the second wave as we vary three quantities: (1) the reduction in contacts due to physical distancing by the entire population, (2) the age group which engages in salutary sheltering, and (3) the fraction of that age group which shelters in place. All results are averages over population-level parameters from the posterior distributions estimated in the previous section. We highlight several main results. The appendix provides a further breakdown of results from each scenario in terms of infections and deaths in those above and below 60 years of age.

First, the marginal impact of salutary sheltering by different age groups in limiting infections in the second-wave outbreak depends on the level of physical distancing adopted by the rest of the population. When physical distancing is high (25% of the original level of contact, shown in the appendix), the second-wave outbreak never infects a significant number of people because the effective reproduction number remains below 1. When physical distancing is not widely adopted (75% of the original level of contact), the outbreak reaches a significant fraction of the population no matter which group engages in sheltering (at least 30% of the population and often more becomes infected). However, in the middle scenario (50% of the original level of contact), the population is in a state where sheltering by members of a group with a large number of average contacts can significantly reduce the extent of total infections. Typically, members of the 20-40 and 40-60 age groups have more contacts than those in older or younger groups [PCJ17], so sheltering by both these groups can sharply reduce the fraction of the population infected in the second wave.

Second, the importance of sheltering by each age group in preventing deaths varies according to the level of physical distancing adopted by the rest of the population. When returning to a near-normal level of contact makes infection of a significant fraction of the population unavoidable (75% of normal contact), deaths are most appreciably reduced by sheltering the 60+ age group, since older individuals are at much higher risk of death after infection than those in younger age groups. However, in the intermediate scenario of 50% contact reduction, it may be more effective for members of younger age groups (20-40 or 40-60) to engage in salutary sheltering. While these individuals are typically at lower risk of death than those in the 60+ group, they also have a significantly larger number of average daily contacts [PCJ17]. By sheltering, they help shield older groups from infection more effectively than if an equivalent fraction of the older group engaged in sheltering themselves.

Third, the impact of sheltering by these groups across different scenarios is impacted by between-population differences. Each population has differences in contact patterns, the estimated probability of infection on contact (p_{inf}), the fraction who were infected in the initial outbreak (assuming short-term immunity against reinfection during the second outbreak), and the vulnerability of older individuals. For example, sheltering by the 60+ age group reduces deaths much more substantially in Lombardy than in either Hubei or New York City because Italian fatalities are concentrated more heavily in older groups, with 95% of reported deaths in the 60+ age group compared to 80% in Hubei and 74% in New York

City [VOD$^+$20, oHH20, ORB20]. As a result, it is still slightly preferable in terms of averted deaths to shelter the 60+ group in Lombardy even in scenarios where there would be an advantage to sheltering by younger groups in other locations (50% contact levels). Another example is in Hubei, where the fraction of the population that is newly infected in the second wave is larger than in either Lombardy or New York City (despite a lower estimated r_0 in Hubei). This is because we estimate that a non-negligible portion of Lombardy and New York City were both previously infected, while the population of Hubei province is still almost entirely susceptible (see previous section). The interplay of demographics, social structures, and the impact of the first outbreak create a range of between-population differences across scenarios.

Building on this analysis of Hubei, Lombardy, and New York City, our model suggests that hybrid policies that combine targeted salutary sheltering by one sub-population and physical distancing by the rest can substantially mitigate infections and deaths due to a second-wave outbreak. However, the relative importance of sheltering by different age groups is strongly impacted by the extent to which physical distancing is adopted by the rest of the population and by a range of factors which can differ between populations. This suggests that demography and behavior in a particular place must be carefully considered while developing population-level interventions. Our analysis can be readily extended to other locations by parameterizing our model for a new population using existing demographic data and age-stratified contact patterns, allowing analysis of population-specific interventions.

12.3 Discussion and Future Work

In this study, we developed a model of SARS-CoV2 transmission that incorporates household structure, age distributions, comorbidities, and age-stratified contact patterns in Hubei, Lombardy, and New York City and created simulations using available demographic information from these three locations. Our findings suggest that in some locations, substantial reductions in SARS-CoV2 spread can be achieved by less drastic options short of population-wide sheltering in place. Instead, targeted salutary sheltering of specific age groups combined with adherence to physical distancing by the rest of the population may be sufficient to thwart a substantial fraction of infections and deaths. Physical distancing could be achieved by engaging in activities such as staggered work schedules, increasing spacing in restaurants, and prescribing times to use the gym or grocery store. Specific mechanisms and considerations for implementing physical distancing are documented in the supplementary text. It is important to note that between-population differences in the impact of sheltering different age groups can be substantial. Contact patterns, household structures, and variation in fatality rates (whether due to demographics or factors such as health system capacity) all influence the number of infections or deaths averted by sheltering a particular group. Thus, the implementation of physical distancing and sheltering policies should be tailored to the dynamics of COVID-19 in a particular population.

From a pragmatic perspective, targeted salutary sheltering may not be realistic for all populations. Its feasibility relies on access to safe shelter, which does not reflect reality for all individuals. In addition, sociopolitical realities may render this recommendation more feasible in some populations than in others. Concerns for personal liberty, discrimination against sub-segments of the population, and societal acceptability may prevent the adoption of targeted salutary sheltering in some regions of the world. Allowing salutary sheltering to operate on a voluntary basis using a shift system (rather than for indefinite time periods) may address some of these issues. Future work should formulate targeted recommendations about salutary sheltering and physical distancing by age group or other stratification adapted to a specific country's workforce.

One strength of this study is our ability to assess targeted interventions such as salutary sheltering in a population-specific manner. Existing modeling work of COVID-19 has largely focused on simpler compartmental or branching process models which do not allow for such assessments. While these models have played an important role in estimating key parameters such as r_0 [KRD$^+$20, RA20] and the rate at which infections are documented [DSNT$^+$20], as well as in the evaluation of prospective non-pharmaceutical interventions

[KTLG20, HAG+20], they do not characterize how differences in demography impact the course of an epidemic in a particular location. Our focus on population-specific demography allows for further refinement of current mortality estimates and is a strength of this study. r_0 estimates in this study are generally comparable to other estimates in the literature [MM20], although our model yields higher estimates for New York City and Lombardy than Hubei – possibly due to differential mask-wearing practices [FSX+20] or adoption of behavioral interventions such as hand hygiene [DGAA+08]. Reporting rates estimated in this study were generally lower than those in prior studies [RHA+20], although the trend across locations is consistent. One potential explanation is that Russell et al. estimate documentation from death data using a case fatality rate (CFR) from the literature while our model uses an *infection* fatality rate (IFR). The IFR is lower because it includes all infections, not only those that become confirmed cases. A lower fatality rate in turn implies that each additional infection is less likely to result in death, and so a greater number of total infections are required to account for the observed number of deaths.

One key advantage of our framework is its flexibility. Our model is modifiable to test different policies or simulate additional features with greater fidelity across a variety of populations. Examples of future work that can be accommodated include analysis of contact tracing and testing policies, health system capacity, and multiple waves of infection after lifting physical distancing restrictions. Our model includes the necessary features to simulate these scenarios while remaining otherwise parsimonious, a desirable feature given uncertainties in data reporting.

This study is not without limitations however. While several comorbidities associated with mortality in COVID-19 were accounted for, the availability of existing data limited the incorporation of all relevant comorbidities. Most notably, chronic pulmonary disease was not included although it has been associated with mortality in COVID-19 [fDCP20b], nor was smoking, despite its prevalence in both China and Italy [PX19, LZP+17]. Gender-mediated differences were also excluded, which may be important for both behavioral reasons (e.g., adoption of hand-washing [GMGS97, JSG+03]) and biological reasons (e.g, the potential protective role of estrogen in SARS-CoV infections [CFM$^+$17]). Nevertheless, these factors can all be incorporated into the model as additional data becomes available.

Additionally, our second-wave scenarios assumed that individuals who were infected previously are immune to reinfection during the second wave. The duration of acquired immunity to SARS-CoV2 has not been precisely defined, though antibody kinetics have been studied in recent work [LTS$^+$20, SGM$^+$20, IJN$^+$20]. If reinfection during a second wave is common, more individuals may be infected than predicted by our simulations (though mortality may be lower if previous infection is protective against adverse effects).

Finally, it is worth noting that we have not yet attempted to model super-spreader events in our existing framework. Such events may have been consequential in South Korea [Bri20b], and future work could attempt to model the epidemic there by incorporating a dispersion parameter into the contact distribution, a method which has been employed in other models [RA20].

Despite these limitations, this study demonstrates the importance of considering population and household demographics when attempting to better define outbreak dynamics for COVID-19. Furthermore, this model highlights potential policy implications for non-pharmaceutical interventions that account for population-specific demographic features and may provide alternative strategies for national and regional governments moving forward.

Chapter 13

Bayesian inference for partially observed epidemics

A key goal for public health is effective surveillance and tracking of infectious disease outbreaks. This work is motivated in particular by the COVID-19 pandemic but the methods we describe are applicable to other diseases as well. A central question is to estimate the empirical rate of transmission over time, often formalized via the reproduction number $R_t, t = 1...T$. R_t describes the expected number of secondary infections caused by someone infected at time t. Accurate estimates of R_t are critical to detect emerging outbreaks, forecast future cases, and measure the impact of interventions imposed to limit spread.

R_t is typically estimated using daily case counts, i.e., the number of new infections detected via testing each day. Standard methods, including prominent dashboards developed for COVID-19, provide accurate estimates under idealized conditions for the observation of cases and have been successfully used at a national or state level where many observations are available and sampling variation averages out [AHT$^+$20, FMG$^+$20, SVK20]. However, successful reopening will require programs which track spread at the level of particular colleges, workplaces, or towns, where *partial observability* poses several challenges. *First*, only a small number of infected people may be tested. It is estimated that only about 10% of SARS-COV-2 infections in the US result in a confirmed test [HRL$^+$20] and we could expect even fewer in populations with a high prevalence of asymptomatic or mild infections (e.g., college students). *Second*, the biological properties of the test play an important role. For example, a PCR test which detects viral RNA will show positive results at different times than an antibody or antigen test. Further, there can be substantial heterogeneity across individuals. *Third*, testing programs may collect samples in a particular way which impacts the observations. For example, one suggestion for schools and workplaces to reopen is to institute regular surveillance testing of a fraction of the population in order to detect outbreaks and catch asymptomatic carriers [LWL$^+$21]. The observations will depend on the fraction of the population enrolled in testing (potentially small due to budget constraints) along with the sampling design (e.g., cross-sectional vs longitudinal).

This chapter presents *GPRt*, a novel Bayesian approach to estimating R_t which accounts for partial observability in a flexible and principled manner (illustrated in Figure 13.1). This method yields well-calibrated probabilistic estimates (the posterior distribution). Our model places a Gaussian process (GP) prior over R_t, allowing it to be an arbitrary smooth function. Then, we explicitly model the sampling process which generates the observations from the true trajectory of infections. While this substantially improves accuracy (as we show experimentally) it creates a much more difficult inference problem than has been previously considered. Specifically, our model contains tens or hundreds of thousands of discrete latent variables, preventing the application of out-of-the-box methods. Moreover, the values of many variables are tightly correlated in the posterior distribution, further complicating inference. To make inference computationally tractable, we propose a novel stochastic variational inference method, enabled by a custom stochastic gradient estimator for the variational objective. Extensive experiments show that our method recovers an accurate and well-calibrated posterior distribution in challenging situations where previous methods fail.

13.1 Related Work

There is a substantial body of work which attempts to infer unknowns in a disease outbreak. A frequent target for inference is the *basic* reproduction number R_0 [MM20, RA20, WCK$^+$20];

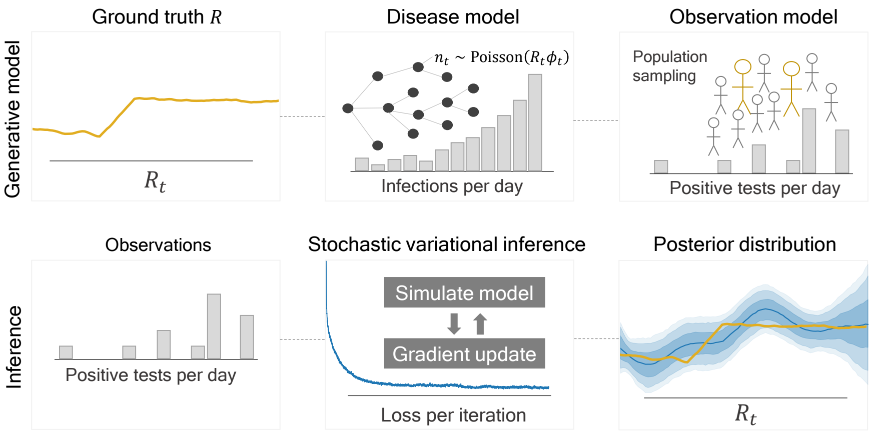


Figure 13.1: *Illustration of our GPRt method. Top row: the generative model GPRt posits for the observed data. Bottom row: the inference process to recover a posterior over R_t.*

by contrast, we attempt the more challenging task of estimating a reproduction number which can vary arbitrarily over time. There is a literature of both classic methods for estimating R_t [WT04, CFFC13, CCFJ19] and newer methods developed for the COVID-19 pandemic and implemented in popular dashboards [AHT+20, SVK20]. None of these methods incorporate partial observability and we empirically demonstrate that GPRt improves substantially over methods in each category. Another strand of literature develops maximum likelihood or particle filter estimates of the parameters of an epidemiological model [KIPB08, DKB13, CCD18]. However, their work focuses on accommodating a complex model of the underlying disease dynamics; by contrast, we develop methods for probabilistically well-grounded inferences under a complex observation model. There is also a great deal of computational work more broadly concerned with disease control. Examples include optimization problems related to vaccination or quarantine decisions [SAPV15, ZP14a, ZAVP15], machine learning methods for forecasting (without recovering a probabilistic view of R_t) [CKL+14, RGM+15], and agent-based simulations of disease dynamics [BEM08]. Our work complements this literature by allowing inference of a distribution over R_t from noisy data, which can serve as the input that parameterizes an optimization problem or simulation.

13.2 Model

We now introduce a model for a disease process with a time-varying reproduction number. Subsequently, we introduce example models for how the observations are generated from the disease process which our framework can support.

13.2.1 Disease Model

We use a standard stochastic model of disease transmission similar to other R_t estimation methods [WT04, CFFC13, CCFJ19]; our contribution is a more powerful inference methodology which can accommodate complex observation models alongside a GP prior. Let $\boldsymbol{R} = [R_1...R_T]$ be the vector with R_t for each time. From $\boldsymbol{R}$, the disease model defines a distribution over a vector $\boldsymbol{n} = [n_1...n_T]$ with the number of people newly infected each day. The main idea is that, over the course of a given individual's infection, they cause a Poisson-distributed number of new infections with mean determined by $\boldsymbol{R}$. Specifically, if on day t individual i has been infected for h^i days the expected number of new infections caused by i that day is

$$\lambda_t^i = w_{h^i} R_t.$$

Here w_{h^i} gives the level of infectiousness of an individual h^i days post-infection. w is normalized so that $\sum_h w_h = 1$. For later convenience, we will define $\phi_t = \sum_{i=1}^N w_{h^i}$ to be the total infectiousness in the population before scaling by R_t (N is the total population size).

Each day, each infected individual i draws $n_t^i \sim \text{Poisson}(\lambda_t^i)$ other individuals to infect. We also incorporate infections from outside the population, with a mean of γ such infections per day. We assume the rate of external infection is constant with respect to our time but our model could be extended to a time-varying γ. We treat γ mostly as a nuisance parameter: our true objective is to infer $\boldsymbol{R}$, but doing so requires accounting for the potential that some detected cases are due to infections from outside the population. We define $n_t = \sum_{i=1}^N n_t^i + \text{Poisson}(\gamma)$ to be the total number of new infections. Since $\boldsymbol{\phi}$ is fixed given the time series $\boldsymbol{n}$, we denote it as a function $\boldsymbol{\phi}(\boldsymbol{n})$. We denote the probabilistic disease model induced by a specific choice of $\boldsymbol{R}$ and γ as $M(\boldsymbol{R}, \gamma)$ and the draw of a time series of infections from the model as $\boldsymbol{n} \sim M(\boldsymbol{R}, \gamma)$.

13.2.2 Observation Model

We now depart from the standard disease model used in previous work and describe a wide-ranging set of examples for how our framework can accommodate models of the process which generates the observed data from the latent (unknown) true infections. Previous work assumes either perfect observability or else the simplest of the three observation models we describe below (uniform undersampling). Our focus is where observations are generated by a medical test which confirms the presence of the pathogen of interest. Individuals have some probability of being tested at different times (depending on the testing policy adopted) and then test positive with a probability which depends on the biological characteristics of the disease and the test in question.

Modeling tests

The kind of test employed determines when an individual is likely to test positive during the course of infection. For example, for COVID-19, PCR tests are commonly used to detect SARS-COV-2 RNA. They are highly sensitive and can detect early infections. Most infected individuals become PCR-negative within the week or two following infection as viral RNA is cleared [KLL$^+$20]. By contrast, serological tests detect the antibodies produced after infection. An individual is not likely to test serologically positive until a week or more post-infection, but may then continue to test serologically positive for months afterwards [IJN$^+$20]. The observable data generated by a serological testing program is likely quite different than a PCR testing program since the time-frame and variance of when individuals test positive differs strongly. A range of other examples are possible (e.g., antigen tests) and can be easily incorporated into our framework.

Our model adopts a generic representation of a particular test as a distribution D over t_{convert}, the number of days post-infection when an individual begins to test positive and

t_{revert}, the number of days post-infection when they cease to test positive. For an infected individual i, we write $t^i_{\text{convert}}, t^i_{\text{revert}} \sim D$. Our method only assumes the ability to sample from D, meaning that we can directly plug in the results of lab studies assessing the properties of a test. $t^i_{\text{convert}}, t^i_{\text{revert}}$ are unobserved: we only get to see if an individual tests positive at a given point in time, not the full range of times that they *would* have tested positive.

Next, we describe a series of example models for how and when individuals are tested, which reveal observations depending on the status $(t^i_{\text{convert}}, t^i_{\text{revert}})$ for each person who is tested. For convenience, we let $t^i_{\text{convert}} = \infty$ for an individual who is never infected. Note however that we can model false negatives by having D sometimes set $t^i_{\text{convert}} = \infty$ for an infected individual, or false positives by returning finite t^i_{convert} for an uninfected individual. We denote the number of observed positive tests on day t as x_t. An observation model is a distribution over $\boldsymbol{x}$ given $\boldsymbol{n}$, denoted $\boldsymbol{x} \sim Obs(\boldsymbol{n})$. Each setting below describes one such distribution.

Uniform undersampling

In this setting, each individual who is infected is tested independently with some probability p_{test} (e.g., if they individually decide whether to seek a test). To model this process, we introduce two new sets of latent random variables. First, a binary variable $z^i \sim \text{Bernoulli}(p_{\text{test}})$, indicating whether individual i is tested. Second, a delay c^i, giving the number of days between t^i_{convert} and when individual i is tested. We can integrate out the z^i and obtain the following conditional distribution for the observed number of positive tests x_t:

$$x_t | c, t_{\text{convert}} \sim \text{Binomial}\left(\sum_{i=1}^{N} 1[t = t^i_{\text{convert}} + c^i], p_{\text{test}}\right)$$

where $1[\cdot]$ denotes the indicator function of an event. However, we cannot analytically integrate out t_{convert} and c.

Cross-sectional testing

Here, a uniformly random sample of s_t individuals are tested on each day t. This models a random screening program (e.g., testing random employees each day as they enter a workplace). In this case, we have

$$x_t | t_{\text{convert}}, t_{\text{revert}} \sim \text{Binomial}\left(s_t, \frac{1}{N}\sum_{i=1}^{N} 1[t^i_{\text{convert}} \leq t < t^i_{\text{revert}}]\right)$$

This expression provides the likelihood of $\boldsymbol{x}$ after conditioning on the latent variables $t^i_{\text{convert}}, t^i_{\text{revert}}$, though there is no closed-form expression conditioning only on $\boldsymbol{n}$.

Longitudinal testing

In this setting, a single sample from the population is chosen up front and every individual in the sample is tested every d days. We again denote the total number of individual tested on day t as s_t, but note that now the group of individuals who are tested repeats every d days. Longitudinal testing offers different (and potentially more revealing) information than cross-sectional testing since when an individual first tests positive, we know that they did *not* test positive d days ago. However, it complicates inference by introducing correlations between the test results at different time steps. Let A_t denote the set of individuals who are tested at time t. We assume that the complete sample $\bigcup_{t=1}^{d} A_t$ is chosen uniformly at random from the population, with the chosen individuals then randomly partitioned between the d days. We have

$$x_t = \sum_{i \in A_t} 1[t^i_{\text{convert}} > t - d \text{ and } t^i_{\text{convert}} \leq t \text{ and } t^i_{\text{revert}} > t]$$

where $t^i_{\text{convert}} > t - d$ captures that i was not positive on their previous test. This introduces correlations between x_t and x_{t-d}, so there is not a simple closed-form expression for the distribution of the time series $\boldsymbol{x}$ even after conditioning on t^i_{convert} and t^i_{revert}. (as there is in the cross-sectional case). We will instead build a flexible framework for inference which can just as well use a kind of sample of the log-likelihood.

Inference Problem

We will place a Gaussian process (GP) prior over $\boldsymbol{R}$, resulting in the following generative model:

$$\boldsymbol{R} \sim \mathcal{GP}(1, \mathcal{K}), \gamma \sim \text{Exp}(\bar{\gamma})$$

$$\boldsymbol{n} \sim M(\boldsymbol{R}, \gamma)$$

$$\boldsymbol{x} \sim Obs(\boldsymbol{n}).$$

where $\mathcal{GP}(1, \mathcal{K})$ denotes a Gaussian process with constant mean 1 and kernel $\mathcal{K}$ and $\text{Exp}(\bar{\gamma})$ is an exponential prior on γ with mean $\bar{\gamma}$. Given the observation $\boldsymbol{x}$, our goal is to compute the resulting posterior distribution over $\boldsymbol{R}$ and γ. However, is complicated by the fact that $\boldsymbol{x}$ is determined by a large number of discrete latent variables, primarily $\boldsymbol{n}$ (the time series of infections) and $\{t^i_{\text{convert}}, t^i_{\text{revert}}\}_{i=1}^N$, the times when each individual tests positive. A typical strategy for inference in complex Bayesian models is Markov Chain Monte Carlo (MCMC). However, MCMC is difficult to apply because of tight correlations between the values of variables over time: due to the GP prior, we expect values of $\boldsymbol{R}$ to be closely correlated between timesteps, and successive values of $\boldsymbol{n}$ are highly correlated via the model M. Formulating good proposal distributions for high-dimensional, tightly correlated random variables is notoriously difficult and has presented problems for GP inference via MCMC in other domains [TLR08].

The other main approach to Bayesian inference is *variational inference*, where we attempt to find the best approximation to the posterior distribution within some restricted family. Modern variational inference methods, typically intended for deep models such as variational autoencoders [KW13], use a combination of autodifferentiation frameworks and the reparameterization trick to differentiate through the variational objective [KTR$^+$17]. This process is highly effective for models with only continuous latent variables. However, our model has many thousands of discrete latent variables which cannot be reparameterized in a differentiable manner. Typical solutions to this problem would be to either integrate out the discrete variables or to replace them with a continuous relaxation [JGP17, VMB+18]. Neither solution is attractive in our case – the structure of the model does not allow us to integrate out the discrete variables analytically, while a continuous relaxation is infeasible because our latent variables have a strict integer interpretation (every infection requires in a particular individual becoming test-positive at particular points in time).

The last resort to differentiate through discrete probabilistic models is the score function estimator [PBJ12], which is often difficult to apply due to high variance. GPRt uses a combination of techniques which exploit the structure of infectious disease models to develop an estimator with controlled variance. *First*, we develop a more tractable variational lower bound which is amenable to stochastic optimization. *Second*, we hybridize the reparameterization and score function estimators across different parts of the generative model to take advantage of the properties of each component. *Third*, we develop techniques to sample low-variance estimates of the log-likelihood for each of the observation models introduced earlier. These techniques are introduced in the next section.

GPRt: Variational Inference Algorithm

We now derive *GPRt*, a novel variational inference method for R_t estimation. GPRt approximates the true (uncomputable) posterior over $(\boldsymbol{R}, \gamma)$ via a multivariate normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix Σ. μ_t is the posterior mean for R_t while $\Sigma_{t,t'}$ gives the posterior covariance between R_t and $R_{t'}$. μ_γ is the mean for γ and $\Sigma_{\gamma,\cdot}$ gives its covariance with R. The diagonal $\Sigma_{t,t}$ gives the variance of the posterior over R at each time, capturing the overall level of uncertainty. The aim is to find a $\boldsymbol{\mu}$ and Σ which closely approximate the true posterior. Let $q(\boldsymbol{R}, \gamma|\boldsymbol{\mu}, \Sigma)$ denote the variational distribution. Let p be the true generative distribution, where $p(\boldsymbol{R}, \gamma, \boldsymbol{x})$ is the joint distribution over $\boldsymbol{x}$ and $(\boldsymbol{R}, \gamma)$, $p(\boldsymbol{R}, \gamma)$ is the prior over $(\boldsymbol{R}, \gamma)$, and $p(\boldsymbol{R}, \gamma|\boldsymbol{x})$ is the posterior over $(\boldsymbol{R}, \gamma)$ after conditioning on $\boldsymbol{x}$.

The aim of variational inference is to maximize a lower bound on the total log-probability of the evidence $\boldsymbol{x}$:

$$\log p(\boldsymbol{x}) \geq \mathop{\mathbb{E}}_{\boldsymbol{R},\gamma\sim q}[\log p(\boldsymbol{R},\gamma)] + \mathop{\mathbb{E}}_{\boldsymbol{R},\gamma\sim q}[\log p(\boldsymbol{x}|\boldsymbol{R},\gamma)] \\ - \mathop{\mathbb{E}}_{\boldsymbol{R},\gamma\sim q}[\log q(\boldsymbol{R},\gamma|\boldsymbol{\mu},\Sigma)]$$

where the right-hand side is referred to as the *Evidence Lower Bound (ELBO)*. Our goal is to maximize the ELBO via gradient ascent on the parameters $\boldsymbol{\mu}$ and Σ. This requires us to develop an estimator for the gradient of each term in the ELBO. The first term is the negative cross-entropy between q and the prior $p(\boldsymbol{R},\gamma)$. Because both q and p have simple parametric forms, this can be easily computed and differentiated. The last term is the entropy, which is similarly tractable. The middle term is the expected log-likelihood. Developing an estimator for the gradient of this term is substantially more complicated and will be our focus. In fact, for computational tractability we will actually develop an estimator for a lower bound on the expected log-likelihood; substituting this lower bound into the ELBO still gives a valid lower bound on $\log p(\boldsymbol{x})$ and so is a sensible objective.

13.2.3 Gradient Estimator

The essential problem is that computing the log-likelihood of $\boldsymbol{R}$ requires integrating out the discrete latent variable $\boldsymbol{n}$ induced by the disease spread model, which is computationally intractable. The aim of this section is to develop the following stochastic estimator:

Theorem 24. *Let L be the Cholesky factor of Σ, $\xi \sim N(0, I)$, and $\begin{bmatrix} \boldsymbol{R} \\ \gamma \end{bmatrix} = \boldsymbol{\mu} + L\xi$. Let $\boldsymbol{n} \sim M(\boldsymbol{R},\gamma)$. Finally, define*

$$\hat{\nabla} = \nabla_{\boldsymbol{\mu},L} \log M(\boldsymbol{n}|\boldsymbol{R},\gamma) \log p(\boldsymbol{x}|\boldsymbol{n}).$$

There exists a function $g(\boldsymbol{\mu},\Sigma)$ with $\mathbb{E}_{\boldsymbol{R},\gamma\sim q}[\log p(\boldsymbol{x}|\boldsymbol{R},\gamma)] \geq g(\boldsymbol{\mu},\Sigma)$ $\forall \boldsymbol{\mu},\Sigma$ and $\mathbb{E}[\hat{\nabla}] = \nabla g(\boldsymbol{\mu},\Sigma)$.

Essentially, Theorem 24 states that $\hat{\nabla}$ is an unbiased estimator for a lower bound on the expected log-likelihood, exactly what we need to optimize a lower bound on $\log p(\boldsymbol{x})$ by stochastic gradient methods. Moreoever, as we will highlight below, we can efficiently compute the terms of $\hat{\nabla}$ via a combination of leveraging the structure of the disease model to apply autodifferentiation tools and novel sampling methods for the observation model. We now derive $\hat{\nabla}$.

Proof. Expanding the dependence of $\boldsymbol{x}$ on $\boldsymbol{n}$ we can rewrite the log-likelihood as

$$\mathop{\mathbb{E}}_{\boldsymbol{R},\gamma\sim q}[\log p(\boldsymbol{x}|\boldsymbol{R},\gamma)] = \mathop{\mathbb{E}}_{\boldsymbol{R},\gamma\sim q(\boldsymbol{\mu},\Sigma)}\left[\log\left(\mathop{\mathbb{E}}_{\boldsymbol{n}\sim M(\boldsymbol{R},\gamma)}[p(\boldsymbol{x}|\boldsymbol{n})]\right)\right]$$

It is not clear how to develop a well-behaved gradient estimator for this expression because we wish to differentiate with respect to the parameters governing two nested expectations, one within the log. However, via Jensen's inequality, we can derive the lower bound

$$\mathop{\mathbb{E}}_{\boldsymbol{R},\gamma\sim q}[\log p(\boldsymbol{x}|\boldsymbol{R},\gamma)] \geq \mathop{\mathbb{E}}_{\boldsymbol{R},\gamma\sim q(\boldsymbol{\mu},\Sigma)}\left[\mathop{\mathbb{E}}_{\boldsymbol{n}\sim M(\boldsymbol{R},\gamma)}[\log p(\boldsymbol{x}|\boldsymbol{n})]\right],$$

pushing the log inside the expectation. We will substitute this bound into the ELBO, obtaining a valid lower bound to maximize. The key advantage is that our new lower bound admits an efficient stochastic gradient estimator. We start with the inner expectation and attempt to compute a gradient with respect to $\boldsymbol{R}$ (which controls the distribution of the simulation results $\boldsymbol{n}$). Using *score function estimator* gives

$$\begin{aligned}&\nabla_{\boldsymbol{R},\gamma}\mathop{\mathbb{E}}_{\boldsymbol{n}\sim M(\boldsymbol{R},\gamma)}[\log p(\boldsymbol{x}|\boldsymbol{n})]\\&=\mathop{\mathbb{E}}_{\boldsymbol{n}\sim M(\boldsymbol{R},\gamma)}[\nabla_{\boldsymbol{R},\gamma}\log M(\boldsymbol{n}|\boldsymbol{R},\gamma)\log p(\boldsymbol{x}|\boldsymbol{n})]\end{aligned}$$

which expresses the gradient with respect to $(\boldsymbol{R},\gamma)$ in terms of the gradient of the probability density of the disease model M with respect to $(\boldsymbol{R},\gamma)$. It turns out that $\log M(\boldsymbol{n},\boldsymbol{\phi}|\boldsymbol{R},\gamma)$ can be easily computed. Recall that $n_t = \sum_{i=1}^{N} n_t^i + \text{Poisson}(\gamma)$, where $n_t^i \sim \text{Poisson}(R_t\phi_t^i)$. Using the Poisson superposition theorem, we have that $n_t \sim \text{Poisson}(\sum_{i=1}^{N} R_t\phi_t^i + \gamma)$ (while ϕ_t is a deterministic function of $n_1...n_{t-1}$). Accordingly, we have that

$$\begin{aligned}\log M(\boldsymbol{n}|\boldsymbol{R},\gamma) &= \sum_{t=1}^{T}\log\Pr[n_t|n_1...n_T]\\&=\sum_{t=1}^{T} n_t\log\left(R_t\phi_t(\boldsymbol{n})+\gamma\right) - e\left(R_t\phi_t(n)+\gamma\right) - \log n_t!\end{aligned}$$

where the second line substitutes the Poisson log-likelihood. This expression can be easily differentiated with respect to $\boldsymbol{R}$ and γ in closed form. Accordingly, we obtain an unbiased estimate of the gradient of our lower bound by sampling $\boldsymbol{n} \sim M(\boldsymbol{R}, \gamma)$ and computing

$$\nabla_{\boldsymbol{R},\gamma} \log M(\boldsymbol{n}|\boldsymbol{R}, \gamma) \log p(\boldsymbol{x}|\boldsymbol{n}).$$

This suffices to estimate the gradient with respect to $(\boldsymbol{R}, \gamma)$. However, our goal is to differentiate with respect to $\boldsymbol{\mu}$ and Σ, which control the distribution over $(\boldsymbol{R}, \gamma)$. Fortunately, $\boldsymbol{R}$ and γ are continuous. So, we can exploit the reparameterization trick by writing $(\boldsymbol{R}, \gamma)$ as a function of a random variable whose distribution is fixed. Specifically, since Σ is positive semi-definite, it has a Cholesky decomposition $\Sigma = LL^T$ (for convenience, we actually optimize over L instead of Σ). Sampling a standard normal variable $\xi \sim N(0, I)$ and letting $\begin{bmatrix} \boldsymbol{R} \\ \gamma \end{bmatrix} = \boldsymbol{\mu} + L\xi$ is equivalent to sampling $\boldsymbol{R}, \gamma \sim N(\boldsymbol{\mu}, \Sigma)$. We rewrite the lower bound as

$$g(\boldsymbol{\mu}, \Sigma) = E_{\xi \sim N(0,I)} \left[\mathop{\mathbb{E}}_{\boldsymbol{n} \sim M(\boldsymbol{R}(\xi), \gamma(\xi))} [\log p(\boldsymbol{x}|\boldsymbol{n})] \right]$$

where $\boldsymbol{\mu}$ and L appear only as parameters of the deterministic function expressing $\boldsymbol{R}$ and γ in terms of ξ, instead of in the distribution of a random variable. Taking a sample from each of the expectations and substituting the score function estimator now gives the desired expression for $\hat{\nabla}$. □

Using Theorem 24, our final gradient estimator will sample b values for ξ, run the model M once for each of the resulting values of $\boldsymbol{R}$ to sample $\boldsymbol{n}$, and then compute

$$\frac{1}{b} \sum_{k=1}^{b} \nabla_{\boldsymbol{\mu},L} \log M(\boldsymbol{n}(k)|\boldsymbol{R}(\xi(k)), \gamma(\xi(k))) \log p(\boldsymbol{x}|\boldsymbol{n}(k)),$$

easily accomplished with standard autograd tools given the closed-form expressions for $\boldsymbol{R}(\xi)$, $\gamma(\xi)$, and $\log M(\boldsymbol{n}|\boldsymbol{R})$. In practice, we also use the mean $\log p(\boldsymbol{x}|\boldsymbol{n}(k))$ as a simple control variate to reduce variance [SB98].

Longitudinal	0.5%	1%	2%	5%
WT	0.481 ± 0.147	0.405 ± 0.11	0.395 ± 0.0947	0.401 ± 0.113
Cori	1.74 ± 0.774	1.18 ± 0.573	0.806 ± 0.373	0.546 ± 0.212
EpiNow	0.329 ± 0.211	0.265 ± 0.145	0.25 ± 0.135	0.267 ± 0.16
GPRt	**0.228 ± 0.0713**	**0.2 ± 0.055**	**0.183 ± 0.0579**	**0.186 ± 0.0692**
Cross-sectional	0.05%	0.1%	0.2%	0.5%
WT	0.474 ± 0.149	0.396 ± 0.132	0.369 ± 0.102	0.358 ± 0.101
Cori	1.3 ± 0.676	0.859 ± 0.442	0.554 ± 0.184	0.502 ± 0.197
EpiNow	0.306 ± 0.199	0.277 ± 0.174	0.294 ± 0.184	0.302 ± 0.205
GPRt	**0.215 ± 0.063**	**0.178 ± 0.0509**	**0.177 ± 0.049**	**0.172 ± 0.0471**
Uniform underreporting	1%	2%	5%	10%
WT	0.395 ± 0.105	0.389 ± 0.106	0.377 ± 0.111	0.382 ± 0.104
Cori	0.892 ± 0.552	0.614 ± 0.355	0.412 ± 0.162	0.38 ± 0.108
EpiNow	0.311 ± 0.193	0.31 ± 0.186	0.359 ± 0.231	0.394 ± 0.245
GPRt	**0.204 ± 0.0806**	**0.22 ± 0.0878**	**0.181 ± 0.0677**	**0.181 ± 0.0467**

Table 13.1: *Mean absolute error of each method averaged over instances and time points for each setting, along with standard deviation of the absolute error. Individual column headings give the percentage of the population enrolled in testing. Results shown for PCR testing in the outbreak testing; see appendix for a complete version of the table with serological testing and the random trend setting.*

13.2.4 Computing the Likelihood

We now turn to the task of computing the log-likelihood function $\log p(\boldsymbol{x}|\boldsymbol{n})$, which measures the log-likelihood of observing the sequence of positive test results $\boldsymbol{x}$ given $\boldsymbol{n}$ new infections per day. Unfortunately, the log-likelihood is not available in closed form for any of the settings that we consider because it depends on additional latent variables (e.g., $t_{\text{convert}}, t_{\text{revert}}, c$, or A). We will show that it suffices to develop an estimator which lower-bounds the log-likelihood and that such estimators can be efficiently implemented for each of the observation models we consider. Specifically, denote the collection of latent variables used in a particular observation model as α. Then, we have

$$\log p(\boldsymbol{x}|\boldsymbol{n}) = \log \left(\mathop{\mathbb{E}}_{\alpha}[p(\boldsymbol{x}|\boldsymbol{n}, \alpha)|\boldsymbol{n}] \right),$$

which presents a similar difficulty as in developing our earlier lower bound: sampling α to approximate the inner expectation does not result in an unbiased estimator due to the outer log. Using Jensen's inequality in the same way gives

$$\log p(\boldsymbol{x}|\boldsymbol{n}) \geq \mathop{\mathbb{E}}_{\alpha}\left[\log p(\boldsymbol{x}|\boldsymbol{n}, \alpha)|\boldsymbol{n}\right],$$

and so substituting the right-hand side into our variational objective preserves validity of the lower bound. The RHS has the crucial advantage that we can now develop an unbiased estimator by drawing a single sample of α, which can then be substituted into the stochastic gradient estimator of Theorem 24. That is, for each of the simulation results $n(1)...n(b)$ we sample a corresponding value for the latent variables, $\alpha(1)...\alpha(b)$ and use the gradient estimator

$$\frac{1}{b}\sum_{k=1}^{b} \nabla_{\mu,L} \log M(\boldsymbol{n}(k)|\boldsymbol{R}(\xi(k)), \gamma(\xi(k))) \log p(\boldsymbol{x}|\boldsymbol{n}(k), \alpha(k))$$

This works without issue for the uniform undersampling and cross-sectional models where we can obtain a closed form for the log likelihood after conditioning on the appropriate latent variables. However, the longitudinal testing model presents additional complications. In particular, after sampling the latent variables t_{convert}, t_{revert}, and A_t, the number of positive tests becomes deterministic quantity. Denote this simulated trajectory of positive tests $\tilde{x}$. If $\tilde{x} = x$, then $p(\boldsymbol{x}|\ t_{\text{convert}}, t_{\text{revert}}, A) = 1$ and otherwise $p(\boldsymbol{x}|\ t_{\text{convert}}, t_{\text{revert}}, A) = 0$. This renders the above gradient estimator useless because $\log p(\boldsymbol{x}|\boldsymbol{n}(k), \alpha(k)) = -\infty$ unless the simulated trajectory *exactly* matches the observed data (a very low-probability event). While $-\infty$ is technically a valid lower bound for the variational objective, it is not very useful for optimization. Essentially, we need to develop a lower-variance estimator where the lower bound is more useful.

We now present one such improved estimator. The intuition is that we can marginalize out a great deal of the randomness in the naive estimator by only revealing the results of random draws determining A_t a single individual at a time. We start by sampling t_{convert} and t_{revert}. Note that we can expand $\log p(\boldsymbol{x}|\boldsymbol{n}, t_{\text{convert}}, t_{\text{revert}}) = \sum_{t=1}^{T} \log p(x_t|x_1...x_{t-1}, \boldsymbol{n}, t_{\text{convert}}, t_{\text{revert}})$ and consider the likelihood at each day t after conditioning on the results observed on previous days. To compute an estimate for this sum, we introduce a new object, the

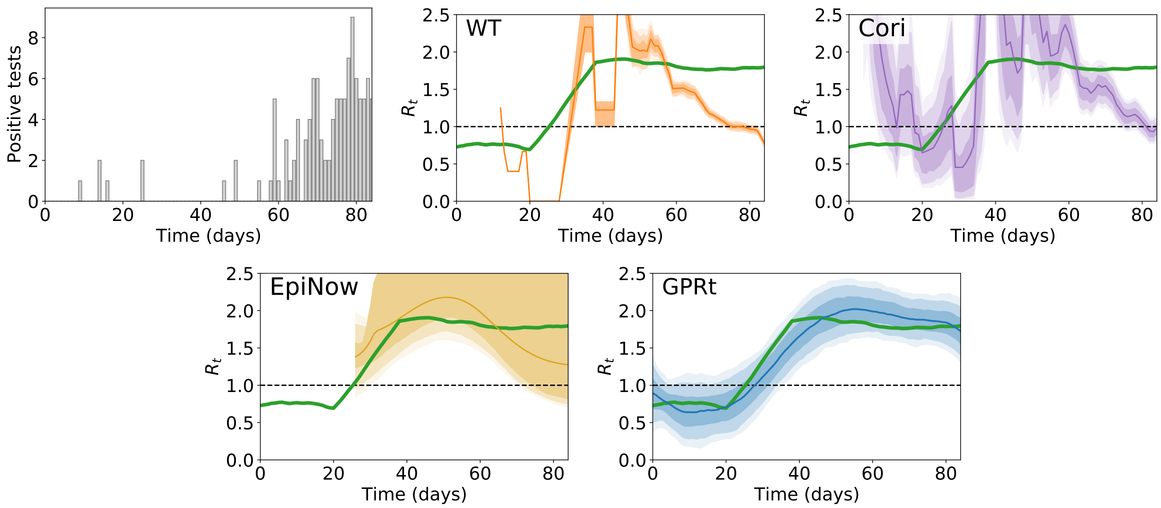


Figure 13.2: *Observed x_t and the distribution over R_t returned by each method on an example in the outbreak setting with longitudinal sampling, $d = 14$, and a 1% sample. The green line gives the ground truth R_t.*

series of matrices C^t. At each time t, $C^t[t_1, t_2]$ denotes the number of individuals who have $t_{\text{convert}} = t_1$, $t_{\text{revert}} = t_2$, and have not yet actually tested positive by time t. Since A_t is selected uniformly at random from the population, independent of the infection process, the x_t individuals who test positive on day t are drawn uniformly at random from the set of all individuals who converted between days $t - d$ and t, and who have not yet reverted. Let n_{draws} denote the number of individuals in A_t who have not yet tested positive by time t and $n_{\text{conv}} = \sum_{t_1=t-d}^{t} \sum_{t_2=t+1}^{T} C^t[t_1, t_2]$ denote the number of individuals who are “eligible" to test positive at time t. Now $x_t | x_1 ... x_{t-1}, \boldsymbol{n}, C^t$ follows a binomial distribution with n_{draws} draws and success probability $\frac{n_{\text{conv}}}{N - \sum_{i=1}^{t-1} x_i}$. Accordingly, the log-likelihood $\log p(x_t | x_1 ... x_{t-1}, \boldsymbol{n}, C^t)$ can be computed in closed form. After this, we can sample $C^t | C^{t+1}$ by selecting a uniformly random individual to remove from C^{t+1}. We can view this as iteratively revealing the test-positive members of A_t after conditioning on the sequence of previous test results, instead of sampling the entire set up front as in the naive method.

13.3 Experimental Results

We test the performance GPRt vs standard baselines on a wide variety of settings. We choose three baselines which have been recommended by leading epidemiologists as methods of choice for COVID-19 [GMB$^+$20]. First is the *Wallinga-Teunis (WT)* method [WT04], which uses the distribution of the time between an infected person and their secondary infections to simulate possible who-infected-who scenarios, each of which induces a particular R_t. WT assumes that cases are observed exactly and that there is no delay in observation. Second is the method of *Cori et al. (Cori)* [TSvG$^+$19, CFFC13] which computes a Bayesian posterior distribution in a similar Poisson branching process infection model. The difference is that their method does not place a GP prior over R_t (instead the posterior factorizes over time) and does not model the sampling method or delays. For both WT and Cori, we apply a common heuristic to correct for time-to-reporting delays, which is shift the method's predictions by the mean delay. Third is *EpiNow* [AHT$^+$20], a MCMC method recently developed for COVID-19 which places a GP prior over $\boldsymbol{R}$ and accounts for the delay distribution, but does not model partial observability.

We test each method in an array of settings, with different distributions for both the true value of R_t and the observations. We include two different settings for the ground truth R_t. First, the *outbreak* setting where $\boldsymbol{R}$ starts below 1 and rises above 1 at a random time. Second, the *random trend* setting where $\boldsymbol{R}$ follows a linear trend which changes randomly at multiple points in time. Details of the settings and other experimental parameters are in the appendix.

We also include different observation models characterized by the test used, the sampling method, and the sample size. We include both *PCR* and *serological* tests, using previously estimated distributions for D [KLL$^+$20, IJN$^+$20]. We also include three sampling models introduced earlier: *uniform underreporting*, *cross-sectional*, and *longitudinal*. Finally, for each of the four combinations of tests and sampling method, we include four different sample sizes. Many sizes model a challenging setting with sparse observations, representing highly limited testing capacity. Note that the sample sizes evaluated are different for each

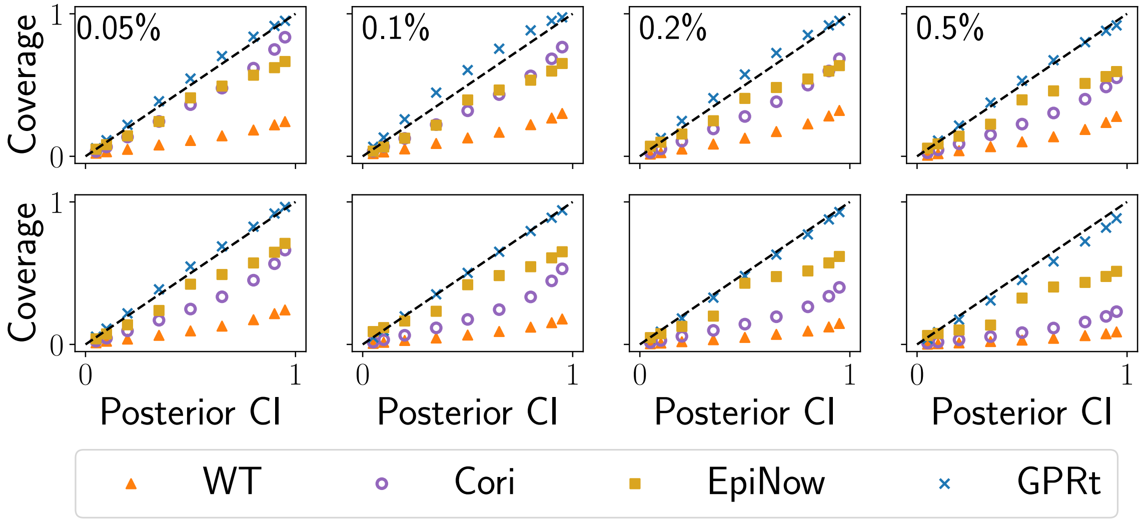


Figure 13.3: *Calibration of each method for cross-sectional testing. Top row: PCR. Bottom row: serological. The number in the upper-left hand corner of each column gives % tested per day. Each individual plot shows the calibration of each method for that setting. Each (x, y) point gives the fraction of the ground truth data points (y) which are covered by the method's posterior interval at level* $\mathbf{x}$*. So, e.g., a point placed at (0.6, 0.7) would indicate that 60% of the ground truth data points fell into the method's 70% credible interval. The dashed diagonal line shows perfect calibration and points lying closer to this line indicate better calibration.*

method because they have different interpretations, e.g. 1% in the cross-sectional case means sampling 1% of the population each day while in the longitudinal case it would mean 1% every d days. For each setting, Table H.2 shows the mean absolute error between the posterior mean $\boldsymbol{R}$ produced by each method and the ground truth. Each entry averages over 100 instances. For longitudinal testing we use $d = 14$; results for other values are very similar (see appendix).

Across almost all settings, our method has lower MAE than any baseline, often by a substantial margin (reducing error by a factor of 2-10x). Notably, GPRt performs well even with extremely limited data (e.g., when testing 0.05% of the population per day or when 1% of infections are observed). Performance improves with more data, but the gains limited (e.g., 0.02-0.04 MAE), indicating that our method is able to make effective use of even very sparse data.

Figure 13.2 shows a representative example. The observed data is quite sparse, with 0-8 positive tests observed per day. Our method recovers a posterior which closely tracks the ground truth. WT produces an estimate which is correlated with the ground truth but has many fluctuations and overly tight confident intervals. Cori is not appropriate for data this sparse and produces a widely fluctuating posterior. EpiNow does not return an estimate for much of the time series, only estimating the part with denser observations (we gave the baselines an advantage by only evaluating their MAE where they returned an estimate). Moreoever, even in the higher-observation portion, it is less accurate than GPRt.

Finally, Figure 13.3 shows *calibration*, a metric which evaluates the entire posterior (not just the mean). Intuitively, calibration reflects that, e.g., 90% of the data should fall into the 90%-credible interval of the posterior. Calibration is critical for the posterior distribution to be interpretable as a valid probabilistic inference, and for it to be useful in downstream decision making. Figure 13.3 shows the fraction of the data which is covered by the credible intervals of each method. This figure shows cross-sectional testing in the outbreak setting, but results for other settings are very similar (see appendix). *GPRt is close to perfectly calibrated (the dotted diagonal line) while the baseline methods are not well calibrated*. The baselines suffer from two problems. First, as to be expected from their higher MAE, they are biased and so their credible intervals often exclude the truth. Second, they are over-confident: paradoxically, their calibration worsens with increased data since the larger sample size makes them more confident in their erroneous prediction. We conclude that GPRt offers better calibrated inferences than the baselines, giving reason to think that Bayesian methods which explicitly incorporate partial observability could be uniquely helpful in downstream tasks.

Chapter 14

Conclusion

This thesis presents a set of contributions in both the technical foundations and practical deployment of AI for public health. On a technical level, this work focuses on the development of optimization algorithms and machine learning models which are able to effectively integrate the process of acting under uncertainty, particularly with limited data and in the context of networked interactions. Specific contributions include the development of algorithms for influence maximization on unknown networks (Chapter 1), submodular optimization under uncertainty (Chapters 2 and 6), methods for incorporating discrete optimization into the training of machine learning models (Chapters 8 and 10), and Bayesian inference methods for partially observed epidemic processes (Chapter 13). Together, these methods help bridge the entirety of the data-decisions pipeline, which spans the process of gathering potentially costly data, using that data to train machine learning models, and using those models to drive downstream decision-making via optimization.

We have also applied these techniques to concrete problems of social importance; indeed, much of the work in this thesis was undertaken from the start in collaboration with nonprofit organizations or experts in public health and social work. Chapters 3 and 4 presented the development and deployment of an AI-augmented intervention for HIV prevention amongst youth experiencing homelessness. We evaluated this system in a field trial enrolling over 700 youth and demonstrated that it reduced key risk behaviors for HIV. To our knowledge, this comprised the first successful evaluation of AI methods for social network interventions in health. Chapter 9 showed how the decision-focused learning techniques that were introduced to integrate discrete optimization and machine learning can be applied to the problem of targeting interventions for tuberculosis treatment. Finally, Chapter 12 used agent-based modeling and Bayesian inference to answer key scientific questions about the dynamics of COVID-19 outbreaks in early hotspots.

A great deal of work remains to be done. Fundamentally, I believe that our understanding of how AI can be used to support community-driven interventions is still in its infancy. Across a range of interconnected domains – public health, social work, education, development, sustainability, and more – questions recur about how we can use data to better understand hard-to-measure social dynamics and formulate interventions which improve human welfare and access to opportunity. These domains present unique challenges for AI researchers. We often come to them as outsiders, needing to build relationships and learn a new context before we can understand how AI might fit into the picture at all. Data might be limited, or expensive to collect when it needs to be gathered by people. Trust in AI methods might be hard to build. Operational challenges may arise throughout the process of an attempted deployment. Methods which are robust, actionable, parsimonious with data and resources, and which humans can decide how and when to trust will have the greatest chance of making an impact.

Specific challenges for future work can be found in all of the areas covered by this thesis. For example, in the setting of networks and influence maximization, how can we develop a more empirically-grounded view of the dynamics of social influence? Can influence reasonably be modeled as a submodular function, or do more complex dynamics need to be incorporated into our algorithms? Increasing returns might easily occur when network ties help to consolidate benefits or buffer against shocks (e.g., the role of social connections in maintaining newly adopted healthy behaviors or providing assistance after financial setbacks). Targeting interventions subject to these self-reinforcing social dynamics requires new computational foundations since the existing influence maximization literature is built almost entirely on assumptions of decreasing returns (formally, submodularity).

In the setting of integrating machine learning and optimization, one challenge for future work is to couple discrete optimization and reinforcement learning for sequential decision-making. So far, my work has focused on integrating solvers for single-shot discrete optimization problems into deep learning. However, in both health and a wide range of other fields, no single decision is made in isolation. Rather, a sequence of choices is made, with new information learned at each step. Standard RL approaches are aimed at settings with either a small number of discrete actions, or else a continuous space. However, real-world resource allocation problems often pose a *combinatorial* set of possible actions at each stage. How can we develop methods which embed algorithmic structure from the discrete optimization literature into the architecture of reinforcement learning agents, giving them the tools to reason about combinatorial spaces?

In the context of computational epidemiology, how do we develop more expressive models which can integrate together disparate sources of data? Outbreak data features a range of modalities (case counts, hospitalizations, serological surveys, crowdsourced symptom reports, smartphone mobility data, etc.) and geographic scales (from nations and states, to cities, to individual schools or workplaces). Observations at all scales may be noisy and biased, with especially sparse observations at fine scales. It is difficult to synthesize these data sources into a principled probabilistic view of the outbreak across scales. Could we develop methods, perhaps drawing on deep generative models and graph representation learning, which can make sense of new kinds of social and epidemiological data? Additional challenges arise in attempting to parse out other forms of heterogeneity; for example, considering variations over time in crucial factors like transmissibility, fatality rates, behaviors, and policies. Better methods are needed to support scalable inference in these more complex models, and to distinguish when specific parameters are even identifiable from the available data and when they are not.

On a broader level, the academic community has a great deal of work ahead in order to train researchers who can take on the challenge of interdisciplinary, community-oriented work involving AI. These projects require many skills which are absent from standard curricula, some of which are difficult to develop in a typical classroom setting. Work on social challenges typically requires the ability to collaborate closely with domain experts outside of computer science (e.g., public health, medicine, or social work). Such collaborations are much richer if both parties develop a common language and have some familiarity with the basics of each other's disciplines. Students should have opportunities to augment computational coursework with introductory training in application domains of interest to develop this fluency. Computer science students might also benefit from supplementary methodological training in areas such as biostatistics or econometrics, to acquire the tools to rigorously evaluate the results of implementing an intervention. However, coursework in any of these areas can only set the foundations for productive engagement with other disciplines. There are many additional skills which are gained only through experience. For example, the process of identifying how potential targets for intervention relate to computational capabilities, investigating data sources, developing appropriate models or formulations, and iterating in response to feedback or pilot tests are all critical components of successful AI-augmented interventions. Also necessary are various "soft skills" in interdisciplinary collaboration; e.g., developing a real understanding of the capabilities, motives, and incentives of domain experts or other stakeholders. Computer science students need opportunities to work through these issues themselves, with support from faculty. Such opportunities may come through joint courses with students in other disciplines, interdisciplinary research projects, programs which facilitate engagement with outside organizations, or a variety of other mechanisms.

No single technical advance will suffice to solve a complex social problem. However, I am confident that AI researchers have a role to play, if we are able to work in concert with domain experts, nonprofit organizations, governments, and community members. It is these broader coalitions that will be needed to drive real change. By simultaneously building the foundations of a more actionable AI and taking as inspiration the needs of those around us, I hope that we will rise to the challenge of creating inclusively beneficial AI.

References

[ADF17] F. Ahmed, J. P. Dickerson, and M. Fuge. Diverse weighted bipartite *b*-matching. In *International Joint Conference on Artificial Intelligence*, pages 35–41, 2017.

[ADG$^+$16] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems*, pages 3981–3989, 2016.

[ADH$^+$16] AmirMahdi Ahmadinejad, Sina Dehghani, MohammadTaghi Hajiaghayi, Brendan Lucier, Hamid Mahini, and Saeed Seddighin. From duels to battlefields: Computing equilibria of blotto and other games. In *AAAI Conference on Artificial Intelligence*, 2016.

[ADSY10] Shipra Agrawal, Yichuan Ding, Amin Saberi, and Yinyu Ye. Correlation robust stochastic optimization. In *ACM-SIAM Symposium on Discrete Algorithms*, 2010.

[AFLT15] Noga Alon, Michal Feldman, Omer Lev, and Moshe Tennenholtz. How robust is the wisdom of the crowds? In *International Joint Conference on Artificial Intelligence*, pages 2055–2061, 2015.

[AGHI09] Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. Diversifying search results. In *WSDM*, pages 5–14. ACM, 2009.

[AGT12] Noga Alon, Iftah Gamzu, and Moshe Tennenholtz. Optimizing budget allocation among channels and influencers. In *Proceedings of the 21st International Conference on World Wide Web*, 2012.

[AHH$^+$20] Sam Abbott, Joel Hellewell, Joe Hickson, James Munday, Katelyn Gostic, Peter Ellis, Katharine Sherratt, Hamish Gibbs, Robin Thompson, Sophie Meakin, Nikos Bosse, Paul Mee, and Sebastian Funk. Epinow2: Estimate real-time case counts and time-varying epidemiological parameters.

https://github.com/epiforecasts/EpiNow2, 2020.

[AHN$^+$17] Nima Anari, Nika Haghtalab, Naor, Joseph (Seffi), Sebastian Pokutta, Mohit Singh, and Alfredo Torrico. Robust submodular maximization: Offline and online algorithms. *arXiv preprint arXiv:1710.04740*, 2017.

[AHT$^+$20] Sam Abbott, Joel Hellewell, Robin N Thompson, Katharine Sherratt, Hamish P Gibbs, Nikos I Bosse, James D Munday, Sophie Meakin, Emma L Doughty, June Young Chun, et al. Estimating the time-varying reproduction number of SARS-CoV-2 using national and subnational case counts. *Wellcome Open Research*, 5(112):112, 2020.

[AK17] Brandon Amos and J Zico Kolter. Optnet: Differentiable optimization as a layer in neural networks. In *International Conference on Machine Learning*, 2017.

[AKBW15] Azin Ashkan, Branislav Kveton, Shlomo Berkovsky, and Zheng Wen. Optimal greedy diversity for recommendation. In *International Joint Conference on Artificial Intelligence*, pages 1742–1748, 2015.

[AKW20] Rediet Abebe, Jon Kleinberg, and S Matthew Weinberg. Subsidy allocations in the presence of income shocks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7032–7039, 2020.

[All10] Paul Allison. *Survival analysis using SAS: a practical guide*. SAS Institute, 2010.

[ALMK16] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica*, 2016.

[AMS18] Mohammad Akbarpour, Suraj Malladi, and Amin Saberi. Diffusion, seeding, and the value of network information. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 641–641, 2018.

[APMV01] Cheryl Alexander, Marina Piazza, Debra Mekos, and Thomas Valente. Peers, schools, and adolescent cigarette smoking. *Journal of Adolescent Health*, 29(1):22–30, 2001.

[ARL$^+$18] Nesreen K Ahmed, Ryan Rossi, John Boaz Lee, Theodore L Willke, Rong Zhou, Xiangnan Kong, and Hoda Eldardiry. Learning role-based graph embeddings. *arXiv preprint arXiv:1802.02896*, 2018.

[AS04] Alexander A Ageev and Maxim I Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization*, 8(3):307–328, 2004.

[AS15] Emmanuel Abbe and Colin Sandon. Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery. In *IEEE Symposium on Foundations of Computer Science*, pages 670–688. IEEE, 2015.

[AVW$^+$18] M. Azizi, P. Vayanos, B. Wilder, E. Rice, and M. Tambe. Designing fair, efficient, and interpretable policies for prioritizing homeless youth for housing resources. In *International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research*, 2018.

[AXRP19] Bijaya Adhikari, Xinfeng Xu, Naren Ramakrishnan, and B Aditya Prakash. Epideep: Exploiting embeddings for epidemic forecasting. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 577–586, 2019.

[Bac15] Francis Bach. Submodular functions: from discrete to continous domains. *arXiv preprint arXiv:1511.00394*, 2015.

[BAC16] Matthew Burgess, Eytan Adar, and Michael Cafarella. Link-prediction enhanced consensus clustering for complex networks. *PLOS One*, 11(5):e0153384, 2016.

[Bar11] Marc Barthélemy. Spatial networks. *Physics Reports*, 499(1-3):1–101, 2011.

[BBC$^+$12] Christian Borgs, Michael Brautbar, Jennifer Chayes, Sanjeev Khanna, and Brendan Lucier. The power of local information in social networks. In *Conference on Web and Internet Economics*, pages 406–419. Springer, 2012.

[BBCL14] Christian Borgs, Michael Brautbar, Jennifer Chayes, and Brendan Lucier. Maximizing social influence in nearly optimal time. In *Proceedings of the twenty-fifth annual ACM-SIAM Symposium on Discrete Algorithms*, pages 946–957. SIAM, 2014.

[BBCT14] Christian Borgs, Michael Brautbar, Jennifer Chayes, and Shang-Hua Teng. Multiscale matrix sampling and sublinear-time pagerank computation. *Internet Mathematics*, 10(1-2):20–48, 2014.

[BBKN19] S. Barman, A. Biswas, S. K. Krishnamurthy, and Y. Narahari. Groupwise maximin fair allocation of indivisible goods. In *AAAI Conference on Artificial Intelligence*, 2019.

[BCDJ13] Abhijit Banerjee, Arun G Chandrasekhar, Esther Duflo, and Matthew O Jackson. The diffusion of microfinance. *Science*, 341(6144), 2013.

[BCDJ14] Abhijit Banerjee, Arun G Chandrasekhar, Esther Duflo, and Matthew O Jackson. Gossip: Identifying central individuals in a social network. Technical report, National Bureau of Economic Research, 2014.

[BCDJ19] Abhijit Banerjee, Arun G Chandrasekhar, Esther Duflo, and Matthew O Jackson. Using gossips to spread information: Theory and evidence from two randomized controlled trials. *The Review of Economic Studies*, 86(6):2453–2490, 2019.

[BCGS10] Christian Borgs, Jennifer Chayes, Ayalvadi Ganesh, and Amin Saberi. How to distribute antidote to control epidemics. *Random Structures & Algorithms*, 37(2):204–222, 2010.

[BCH$^+$18] Nawal Benabbou, Mithun Chakraborty, Xuan-Vinh Ho, Jakub Sliwinski, and Yair Zick. Diversity constraints in public housing allocation. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 973–981, 2018.

[BCP$^+$16] Giulia Berlusconi, Francesco Calderoni, Nicola Parolini, Marco Verani, and Carlo Piccardi. Link prediction in criminal networks: A tool for criminal intelligence analysis. *PlOS One*, 11(4):e0154244, 2016.

[BCZ$^+$16] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in Neural Information Processing Systems*, 2016.

[BD17] Dimitris Bertsimas and Jack Dunn. Optimal classification trees. *Machine Learning*, 106(7):1039–1082, 2017.

[BEGT19] Othman El Balghiti, Adam N Elmachtoub, Paul Grigas, and Ambuj Tewari. Generalization bounds in the predict-then-optimize framework. *arXiv preprint arXiv:1905.11488*, 2019.

[BEM08] Christopher L Barrett, Stephen G Eubank, and Madhav V Marathe. An interaction-based approach to computational epidemiology. In *AAAI Conference on Artificial Intelligence*, pages 1590–1593, 2008.

[BFH$^+$18] Logan C Brooks, David C Farrow, Sangwon Hyun, Ryan J Tibshirani, and Roni Rosenfeld. Nonmechanistic forecasts of seasonal influenza with iterative one-week-ahead distributions. *PLoS Computational Biology*, 14(6):e1006134, 2018.

[BFI$^+$18] R. Bredereck, P. Faliszewski, A. Igarashi, M. Lackner, and P. Skowron. Multiwinner elections with diversity constraints. In *AAAI Conference on Artificial Intelligence*, pages 933–940, 2018.

[BFJ$^+$12] Robert M Bond, Christopher J Fariss, Jason J Jones, Adam DI Kramer, Cameron Marlow, Jaime E Settle, and James H Fowler. A 61-million-person experiment in social influence and political mobilization. *Nature*, 489(7415):295–298, 2012.

[BFT13] D. Bertsimas, V. Farias, and N. Trichakis. Fairness, efficiency, and flexibility in organ allocation for kidney transplantation. *Operations Research*, 61(1):73–87, 2013.

[BG18] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on Fairness, Accountability and Transparency*, pages 77–91. PMLR, 2018.

[BJTK15] Branislav Bosanskỳ, Albert Xin Jiang, Milind Tambe, and Christopher Kiekintveld. Combining compact representation and incremental generation in large games with sequential strategies. In *AAAI Conference on Artificial Intelligence*, 2015.

[BK07] Umesh Bellur and Roshan Kulkarni. Improved matchmaking algorithm for semantic web services based on bipartite graph matching. In *IEEE International Conference on Web Services*, pages 86–93. IEEE, 2007.

[BK10] Michael Brautbar and Michael J Kearns. Local algorithms for finding interesting individuals in large networks. In *Innovations in Theoretical Computer Science*, pages 188–199, 2010.

[BKLP14] Branislav Bosansky, Christopher Kiekintveld, Viliam Lisy, and Michal Pechoucek. An exact double-oracle algorithm for zero-sum extensive-form games with imperfect information. *Journal of Artificial Intelligence Research*, 51:829–866, 2014.

[BKO15] Frank Ball, Edward Knock, and Philip O'Neill. Stochastic epidemic models featuring contact tracing with delays. *Mathematical Biosciences*, 266:23–35, 2015.

[BL09] Alina Beygelzimer and John Langford. The offset tree for learning with partial labels. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009.

[BMBK17] Andrew An Bian, Baharan Mirzasoleiman, Joachim M. Buhmann, and Andreas Krause. Guaranteed non-convex optimization: Submodular maximization over continuous domains. In *International Conference on Artificial Intelligence and Statistics*, 2017.

[BNK$^+$19a] Gagan Bansal, Besmira Nushi, Ece Kamar, Walter S Lasecki, Daniel S Weld, and Eric Horvitz. Beyond accuracy: The role of mental models in human-AI team performance. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 7, pages 2–11, 2019.

[BNK$^+$19b] Gagan Bansal, Besmira Nushi, Ece Kamar, Daniel S Weld, Walter S Lasecki, and Eric Horvitz. Updates in human-AI teams: Understanding and addressing the performance/compatibility tradeoff. In *AAAI Conference on Artificial Intelligence*, volume 33, pages 2429–2437, 2019.

[BPL$^+$16] Irwan Bello, Hieu Pham, Quoc V Le, Mohammad Norouzi, and Samy Bengio. Neural combinatorial optimization with reinforcement learning. *arXiv preprint arXiv:1611.09940*, 2016.

[BPP13] Marco Bressan, Enoch Peserico, and Luca Pretto. The power of local information in pagerank. In *The Web Conference (WWW)*, pages 179–180. ACM, 2013.

[BPS04] Dimitris Bertsimas, Dessislava Pachamanova, and Melvyn Sim. Robust linear optimization under general norms. *Operations Research Letters*, 32(6):510–516, 2004.

[BQNS$^+$20] David Baud, Xiaolong Qi, Karin Nielsen-Saines, Didier Musso, Léo Pomar, and Guillaume Favre. Real estimates of mortality following COVID-19 infection. *The Lancet*, 2020.

[Bre00] Devon D Brewer. Forgetting in the recall-based elicitation of personal and social networks. *Social Networks*, 22(1):29–43, 2000.

[Bri20a] British Broadcasting Corporation. Coronavirus: China outbreak city Wuhan raises death toll by 50%, 2020. `https://www.bbc.com/news/world-asia-china-52321529`, Last Accessed: 2020-05-17.

[Bri20b] British Broadcasting Corporation. Coronavirus: South Korea emergency measures as infections increase, 2020. `https://www.bbc.com/news/world-asia-51582186`.

[BS14] Rahmatollah Beheshti and Gita Sukthankar. A normative agent-based model for predicting smoking cessation trends. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multiagent Systems*, pages 557–564, 2014.

[BS16] S. Barocas and A. Selbst. Big data's disparate impact. *California Law Review*, 104:671, 2016.

[BSBS18] Ashwin Bahulkar, Boleslaw K Szymanski, N Orkun Baycik, and Thomas C Sharkey. Community detection with edge augmentation in criminal networks. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2018.

[BV14] A. Badanidiyuru and J. Vondrák. Fast algorithms for maximizing submodular functions. In *ACM-SIAM Symposium on Discrete Algorithms*, pages 1497–1514, 2014.

[BVVD$^+$17] Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes Van Diest, Bram Van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen AWM Van Der Laak, Meyke Hermsen, Quirine F Manson, Maschenka Balkenhol, et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *Journal of the American Medical Association*, 318(22):2199–2210, 2017.

[BXANK21] Elizabeth Bondi, Lily Xu, Diana Acosta-Navas, and Jackson A Killian. Envisioning communities: A participatory approach towards ai for social good. In *AAAI/ACM Conference on AI, Ethics, and Society*, 2021.

[BYM17] David Belanger, Bishan Yang, and Andrew McCallum. End-to-end learning for structured prediction energy networks. In *International Conference on Machine Learning*, 2017.

[BYW$^+$20] Yan Bai, Lingsheng Yao, Tao Wei, Fei Tian, Dong-Yan Jin, Lijuan Chen, and Meiyun Wang. Presumed asymptomatic carrier transmission of COVID-19. *The Journal of the American Medical Association*, 2020.

[Car20] Fabrizio Carinci. COVID-19: Preparedness, decentralisation, and the hunt for patient zero. *British Medical Journal*, 2020.

[CCD18] Bernard Cazelles, Clara Champagne, and Joseph Dureau. Accounting for non-stationarity in epidemiology by embedding time-varying parameters in stochastic models. *PLoS Computational Biology*, 14(8):e1006211, 2018.

[CCFJ19] Finlay Campbell, Anne Cori, Neil Ferguson, and Thibaut Jombart. Bayesian inference of transmission chains using timing of symptoms, pathogen genomes and contact data. *PLoS Computational Biology*, 15(3):e1006930, 2019.

[CCPV11] Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011.

[CDC15] CDC. Reported STDs in the United States., 2015.

[CDJS15] Jing Cai, Alain De Janvry, and Elisabeth Sadoulet. Social networks and the decision to insure. *American Economic Journal: Applied Economics*, 7(2):81–108, 2015.

[CDPW14] Edith Cohen, Daniel Delling, Thomas Pajor, and Renato F Werneck. Sketch-based influence maximization and computation: Scaling up with guarantees. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 629–638. ACM, 2014.

[Cen19] Minnesota Population Center. Integrated public use microdata series, international: Version 7.2 [dataset], 2019. `https://doi.org/10.18128/D020.V7.2`.

[CEU18] Alex Chin, Dean Eckles, and Johan Ugander. Evaluating stochastic seeding strategies in networks. *arXiv preprint arXiv:1809.09561*, 2018.

[CEW$^+$18] Hongming Chen, Ola Engkvist, Yinhai Wang, Marcus Olivecrona, and Thomas Blaschke. The rise of deep learning in drug discovery. *Drug Discovery Today*, 23(6):1241–1250, 2018.

[CFFC13] Anne Cori, Neil M Ferguson, Christophe Fraser, and Simon Cauchemez. A new framework and software to estimate time-varying reproduction numbers during epidemics. *American Journal of Epidemiology*, 178(9):1505–1512, 2013.

[CFM$^+$17] Rudragouda Channappanavar, Craig Fett, Matthias Mack, Patrick Ten Eyck, David Meyerholz, and Stanley Perlman. Sex-based differences in susceptibility to severe acute respiratory syndrome coronavirus infection. *The Journal of Immunology*, 198(10):4046–4053, 2017.

[CFSV19] V. Conitzer, R. Freeman, N. Shah, and J. Wortman Vaughan. Group fairness for the allocation of indivisible goods. In *AAAI Conference on Artificial Intelligence*, 2019.

[CG20] Benedict Carey and James Glanz. Hidden outbreaks spread through U.S. cities far earlier than americans knew, estimates say. *The New York Times*, 2020. `https://www.nytimes.com/2020/04/23/us/coronavirus-early-outbreaks-cities.html`.

[CGL$^+$19] Andrew Cross, Nakull Gupta, Brandon Liu, Vineet Nair, Abhishek Kumar, Reena Kuttan, Priyanka Ivatury, Amy Chen, Kshama Lakshman, Rashmi Rodrigues, et al. 99dots: a low-cost approach to monitoring and improving medication adherence. In *Proceedings of the Tenth International Conference on Information and Communication Technologies and Development*, page 15. ACM, 2019.

[Cha82] Gary Chamberlain. Multivariate regression models for panel data. *Journal of Econometrics*, 18(1):5–46, 1982.

[Chi20] World Health Organization China. Report of the WHO-China Joint Mission on Coronavirus Disease 2019 (COVID-19), 2020. `https://www.who.int/docs/default-source/coronaviruse/who-china-joint-mission-on-covid-19-final-report.pdf`.

[CHT09] Fan Chung, Paul Horn, and Alexander Tsiatas. Distributing antidote using pagerank vectors. *Internet Mathematics*, 6(2):237–254, 2009.

[CJLBM16] Hau Chan, Albert Xin Jiang, Kevin Leyton-Brown, and Ruta Mehta. Multilinear games. In *Conference on Web and Internet Economics*, 2016.

[CKB$^+$16] Marya E Corden, Ellen M Koucky, Christopher Brenner, Hannah L Palac, Adisa Soren, Mark Begale, Bernice Ruo, Susan M Kaiser, Jenna Duffecy, and David C Mohr. Medlink: A mobile intervention to improve medication adherence and processes of care for treatment of depression in general medicine. *Digital Health*, 2:2055207616663069, 2016.

[CKL$^+$14] Prithwish Chakraborty, Pejman Khadivi, Bryan Lewis, Aravindan Mahendiran, Jiangzhuo Chen, Patrick Butler, Elaine O Nsoesie, Sumiko R Mekaru, John S Brownstein, Madhav V Marathe, et al. Forecasting a moving target: Ensemble models for ili case count predictions. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 262–270. SIAM, 2014.

[CLFC19] H. Chen, G. Loukides, J. Fan, and H. Chan. Limiting the influence to vulnerable users in social networks: A ratio perspective. In *International Conference on Advanced Information Networking and Applications*, 2019.

[CLSS17] Robert Chen, Brendan Lucier, Yaron Singer, and Vasilis Syrgkanis. Robust optimization for non-convex objectives. In *Advances in Neural Information Processing Systems*, 2017.

[CLT14] Elisabetta Carrà, Margherita Lanz, and Semira Tagliabue. Transition to adulthood in Italy: An intergenerational perspective. *Journal of Comparative Family Studies*, 45(2):235–248, 2014.

[CLT$^+$16] Wei Chen, Tian Lin, Zihan Tan, Mingfei Zhao, and Xuren Zhou. Robust influence maximization. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[CMF11] Christina H Chan, Caitlin J McCabe, and David N Fisman. Core groups, antimicrobial resistance and rebound in gonorrhoea in north america. *Sexually Transmitted Infections*, 2011.

[CNM04] Aaron Clauset, Mark EJ Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 70(6):066111, 2004.

[Coh97] E. Cohen. Size-estimation framework with applications to transitive closure and reachability. *Jour. of Comp. and Sys. Sciences*, 55(3):441–453, 1997.

[Col15] David Collett. *Modelling survival data in medical research.* CRC Press, 2015.

[Col17a] Koblenz Network Collection. Adolescent health. `http://konect.uni-koblenz.de/networks/moreno_health`, 2017.

[Col17b] Koblenz Network Collection. Facebook (nips). `http://konect.uni-koblenz.de/networks/ego-facebook`, 2017.

[Col17c] Koblenz Network Collection. Human protein (vidal). `http://konect.uni-koblenz.de/networks/maayan-vidal`, 2017.

[CTMP15] Yinlam Chow, Aviv Tamar, Shie Mannor, and Marco Pavone. Risk-sensitive and robust decision-making: a CVaR optimization approach. In *Advances in Neural Information Processing Systems*, pages 1522–1530, 2015.

[CTP$^+$16] Chen Chen, Hanghang Tong, B Aditya Prakash, Charalampos E Tsourakakis, Tina Eliassi-Rad, Christos Faloutsos, and Duen Horng Chau. Node immunization on large graphs: Theory and algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 28(1):113–126, 2016.

[CV16] Alexandra Carpentier and Michal Valko. Revealing graph bandits for maximizing local influence. In *International Conference on Artificial Intelligence and Statistics*, pages 10–18, 2016.

[CVZ10] C. Chekuri, J. Vondrak, and R. Zenklusen. Dependent randomized rounding via exchange properties of combinatorial structures. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 575–584, Oct 2010.

[CWW10] Wei Chen, Chi Wang, and Yajun Wang. Scalable Influence Maximization for Prevalent Viral Marketing in Large-scale Social Networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, pages 1029–1038, New York, NY, USA, 2010. ACM.

[CWY09] Wei Chen, Yajun Wang, and Siyu Yang. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 199–208, 2009.

[CWY13] Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: General framework and applications. In *International Conference on Machine Learning*, pages 151–159, 2013.

[CZP$^+$14] Yi-Cheng Chen, Wen-Yuan Zhu, Wen-Chih Peng, Wang-Chien Lee, and Suh-Yin Lee. Cim: community-based influence maximization in social networks. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(2):25, 2014.

[DAK17] Priya Donti, Brandon Amos, and J Zico Kolter. Task-based end-to-end model learning in stochastic optimization. In *Advances in Neural Information Processing Systems*, pages 5484–5494, 2017.

[Das11] Gautam Dasarathy. A simple probability trick for bounding the expected maximum of n random variables. 2011., 2011.

[DBW12] John C Duchi, Peter L Bartlett, and Martin J Wainwright. Randomized smoothing for stochastic optimization. *SIAM Journal on Optimization*, 22(2):674–701, 2012.

[DGAA$^+$08] Gabriella Di Giuseppe, Rossella Abbate, Luciana Albano, Paolo Marinelli, and Italo Angelillo. A survey of knowledge, attitudes and practices towards avian influenza in an adult population of Italy. *BMC Infectious Diseases*, 8(1):36, 2008.

[DGCS12] David W Dowdy, Jonathan E Golub, Richard E Chaisson, and Valeria Saraceni. Heterogeneity in tuberculosis transmission and the role of geographic hotspots in propagating epidemics. *Proceedings of the National Academy of Sciences*, 109(24):9557–9562, 2012.

[DK17] Josip Djolonga and Andreas Krause. Differentiable learning of submodular models. In *Advances in Neural Information Processing Systems*, pages 1013–1023, 2017.

[DK20] Catherine D'Ignazio and Lauren F Klein. *Data feminism*. Mit Press, 2020.

[DKB13] Joseph Dureau, Konstantinos Kalogeropoulos, and Marc Baguelin. Capturing the time-varying drivers of an epidemic using stochastic dynamical systems. *Biostatistics*, 14(3):541–555, 2013.

[DKGGR20] Abir De, Paramita Koley, Niloy Ganguly, and Manuel Gomez-Rodriguez. Regression under human assistance. In *AAAI Conference on Artificial Intelligence*, 2020.

[DKS14] Anirban Dasgupta, Ravi Kumar, and Tamas Sarlos. On estimating the average degree. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 795–806, 2014.

[DKW16] Erick Delage, Daniel Kuhn, and Wolfram Wiesemann. “dice”-sion making under uncertainty: When can a random decision reduce risk? Technical Report EPFL-ARTICLE-220662, 2016.

[DLBS14] Nan Du, Yingyu Liang, Maria Balcan, and Le Song. Influence function learning in information diffusion networks. In *International Conference on Machine Learning*, pages 2016–2024, 2014.

[DM10] Moez Draief and Laurent Massouli. *Epidemics and rumours in complex networks*. Cambridge University Press, 2010.

[Dom12] Justin Domke. Generic methods for optimization-based modeling. In *Artificial Intelligence and Statistics*, pages 318–326, 2012.

[DOT14] Kimon Drakopoulos, Asuman Ozdaglar, and John N Tsitsiklis. An efficient curing policy for epidemics on graphs. *IEEE Transactions on Network Science and Engineering*, 1(2):67–75, 2014.

[DSB$^+$19] Emir Demirovic, Peter J Stuckey, James Bailey, Jeffrey Chan, Chris Leckie, Kotagiri Ramamohanarao, and Tias Guns. Prediction + optimisation for the knapsack problem. In *International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research*, 2019.

[DSNT$^+$20] Pablo De Salazar, Rene Niehus, Aimee Taylor, Caroline Buckee, and Marc Lipsitch. Using predicted imports of 2019-nCoV cases to determine locations that may not be identifying all imported cases. *medRxiv*, 2020.

[DX17] Shaddin Dughmi and Haifeng Xu. Algorithmic persuasion with no externalities. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, pages 351–368. ACM, 2017.

[DXW$^+$20] Zhanwei Du, Xiaoke Xu, Ye Wu, Lin Wang, Benjamin Cowling, and Lauren Ancel Meyers. Serial interval of COVID-19 among publicly reported confirmed cases. *Emerging Infectious Diseases*, 2020.

[ECGS11] Stefano Ermon, Jon Conrad, Carla P Gomes, and Bart Selman. Risk-sensitive policies for sustainable renewable resource allocation. In *International Joint Conference on Artificial Intelligence*, pages 1942–1948, 2011.

[EG17] Adam N Elmachtoub and Paul Grigas. Smart “predict, then optimize". *arXiv preprint arXiv:1710.08005*, 2017.

[EKN$^+$17] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115, 2017.

[EN16] Alina Ene and Huy L Nguyen. A reduction for optimizing lattice submodular functions with diminishing returns. *arXiv preprint arXiv:1606.08362*, 2016.

[EPBV20] Albert Esteve, Inaki Permanyer, Diederik Boertien, and James W Vaupel. National age and co-residence patterns shape COVID-19 vulnerability. *medRxiv*, 2020.

[Eub18] Virginia Eubanks. *Automating inequality: How high-tech tools profile, police, and punish the poor*. St. Martin’s Press, 2018.

[Fac20] CIA World Factbook. Field listing: median age, 2020. `https://www.cia.gov/library/publications/the-world-factbook/fields/343.html`, Last Accessed: 2020-03-28.

[FCKT20] Luciano Floridi, Josh Cowls, Thomas C King, and Mariarosaria Taddeo. How to design AI for social good: Seven essential factors. *Sci Eng Ethics*, 2020.

[fDCP17] Centers for Disease Control and Prevention. New York City diabetes ABC profile 2011–2012, 2017.

[fDCP20a] Centers for Disease Control and Prevention. People who are at higher risk for severe illness, 2020. `https://www.cdc.gov/coronavirus/2019-ncov/need-extra-precautions/people-at-higher-risk.html`.

[fDCP20b] Chinese Center for Disease Control and Prevention. The epidemiological characteristics of an outbreak of 2019 novel coronavirus diseases (COVID-19). *China CDC Weekly*, 2(8):113–122, 2020. `http://weekly.chinacdc.cn//article/id/e53946e2-c6c4-41e9-9a9b-fea8db1a8f51`.

[Fel91] Scott L Feld. Why your friends have more friends than you do. *American Journal of Sociology*, 96(6):1464–1477, 1991.

[FHB17] Batya Friedman, David G Hendry, and Alan Borning. A survey of value sensitive design methods. *Foundations and Trends in Human-Computer Interaction*, 11(2):63–125, 2017.

[FMG$^+$20] Seth Flaxman, Swapnil Mishra, Axel Gandy, H Juliette T Unwin, Thomas A Mellan, Helen Coupland, Charles Whittaker, Harrison Zhu, Tresnia Berah, Jeffrey W Eaton, et al. Estimating the effects of non-pharmaceutical interventions on covid-19 in europe. *Nature*, 584(7820):257–261, 2020.

[FMS18] B. Fain, K. Munagala, and N. Shah. Fair allocation of indivisible public goods. In *ACM Conference on Economics and Computation*, pages 575–592, 2018.

[FNP$^+$16] Fei Fang, Thanh Hong Nguyen, Rob Pickles, Wai Y Lam, Gopalasamy R Clements, Bo An, Amandeep Singh, Milind Tambe, Andrew Lemieux, et al. Deploying PAWS: Field optimization of the protection assistant for wildlife security. In *Conference on Innovative Applications of Artificial Intelligence*, pages 3966–3973, 2016.

[FNT$^+$15] Benjamin Ford, Thanh Nguyen, Milind Tambe, Nicole Sintov, and Francesco Delle Fave. Beware the soothsayer: From attack prediction accuracy to predictive reliability in security games. In *International Conference on Decision and Game Theory for Security*, 2015.

[FSST17] P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon. Multiwinner voting: A new challenge for social choice theory. In Ulle Endriss, editor, *Trends in Computational Social Choice*, chapter 2. AI Access, 2017.

[FSX$^+$20] Shuo Feng, Chen Shen, Nan Xia, Wei Song, Mengzhen Fan, and Benjamin Cowling. Rational use of face masks in the COVID-19 pandemic. *The Lancet Respiratory Medicine*, 2020.

[FW81] Stephen E Fienberg and Stanley S Wasserman. Categorical data analysis of single sociometric relations. *Sociological Methodology*, 12:156–192, 1981.

[G$^+$10] NIMH Research Group et al. Results of the NIMH collaborative HIV/sexually transmitted disease prevention trial of a community popular opinion leader intervention. *Journal of Acquired Immune Deficiency Syndromes*, 54(2):204–214, 2010.

[GC19] Ben Green and Yiling Chen. The principles and limits of algorithm-in-the-loop decision making. *ACM Conference on Computer-Supported Cooperative Work and Social Computing*, 2019.

[GFC$^+$16] Stephen Gould, Basura Fernando, Anoop Cherian, Peter Anderson, Rodrigo Santa Cruz, and Edison Guo. On differentiating parameterized argmin and argmax problems with application to bi-level optimization. *arXiv preprint arXiv:1607.05447*, 2016.

[GFCX03] Scott D Gest, Thomas W Farmer, Beverley D Cairns, and Hongling Xie. Identifying children's peer social networks in school classrooms: Links between peer reports and observed interactions. *Social Development*, 12(4):513–529, 2003.

[GGLY17] Xifeng Guo, Long Gao, Xinwang Liu, and Jianping Yin. Improved deep embedded clustering with local structure preservation. In *International Joint Conference on Artificial Intelligence*, 2017.

[GH06] Andrew Gelman and Jennifer Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press, 2006.

[GHEG19] Nina Grgić-Hlača, Christoph Engel, and Krishna P Gummadi. Human decision making with machine assistance: An experiment on bailing and jailing. *ACM Conference on Computer-Supported Cooperative Work and Social Computing*, 2019.

[GJdlHTG13] Harold D Green Jr, Kayla de la Haye, Joan S Tucker, and Daniela Golinelli. Shared risk: who engages in substance use with American homeless youth? *Addiction*, 108(9):1618–1624, 2013.

[GL16] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855–864. ACM, 2016.

[GLL11a] Amit Goyal, Wei Lu, and Laks VS Lakshmanan. Celf++: optimizing the greedy algorithm for influence maximization in social networks. In *Proceedings of the 20th International Conference Companion on World Wide Web*, pages 47–48. ACM, 2011.

[GLL11b] Amit Goyal, Wei Lu, and Laks VS Lakshmanan. Simpath: An efficient algorithm for influence maximization under the linear threshold model. In *2011 IEEE 11th International Conference on Data Mining (ICDM)*, pages 211–220. IEEE, 2011.

[GMB+20] Katelyn M Gostic, Lauren McGough, Edward Baskerville, Sam Abbott, Keya Joshi, Christine Tedijanto, Rebecca Kahn, Rene Niehus, James A Hay, Pablo M De Salazar, et al. Practical considerations for measuring the effective reproductive number, Rt. *medRxiv*, 2020.

[GMGS97] Maryellen Guinan, Maryanne McGuckin-Guinan, and Alice Sevareid. Who washes hands after using the bathroom? *American Journal of Infection Control*, 25(5):424–425, 1997.

[Gon85] Teofilo F Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985.

[Goo20] Google. COVID-19 community mobility reports, 2020. `https://www.google.com/covid19/mobility/`.

[Gov20] Governor's Press Office. Governor Cuomo Announces Phase II Results of Antibody Testing Study Show 14.9% of Population Has COVID-19 Antibodies, 2020. `https://youtu.be/vGGkrjDlh8g?t=220`, Last Accessed: 2020-08-01.

[GPA+20] Giorgio Guzzetta, Piero Poletti, Marco Ajelli, Filippo Trentini, Valentina Marziano, Danilo Cereda, Marcello Tirani, Giulio Diurno, Annalisa Bodina, Antonio Barone, Lucia Crottogini, Maria Gramegna, Alessia Melegaro, and Stefano Merler. Potential short-term outcome of an uncontrolled COVID-19 epidemic in Lombardy, Italy, February to March 2020. *Eurosurveillance*, 25(12), 2020.

[GRB+16] Klaus Greff, Antti Rasmus, Mathias Berglund, Tele Hao, Harri Valpola, and Jürgen Schmidhuber. Tagger: Deep unsupervised perceptual grouping. In *Advances in Neural Information Processing Systems*, 2016.

[Gre20] Ben Green. Data science as political action: Grounding data science in a politics of justice. *Available at SSRN 3658431*, 2020.

[GRHS$^+$20] Ana S Gonzalez-Reiche, Matthew M Hernandez, Mitchell Sullivan, Brianne Ciferri, Hala Alshammary, Ajay Obla, Shelcie Fabre, Giulio Kleiner, Jose Polanco, Zenab Khan, et al. Introductions and early spread of SARS-CoV-2 in the New York City area. *medRxiv*, 2020.

[GRLK12] Manuel Gomez-Rodriguez, Jure Leskovec, and Andreas Krause. Inferring networks of diffusion and influence. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(4):21, 2012.

[Gro11] GroupLens. Movielens dataset, 2011.

[GvSS17] Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. Neural expectation maximization. In *Advances in Neural Information Processing Systems*, 2017.

[HAG$^+$20] Joel Hellewell, Sam Abbott, Amy Gimma, Nikos Bosse, Christopher Jarvis, Timothy Russell, James Munday, Adam Kucharski, John Edmunds, Sebastian Funk, and Rosalind Eggo. Feasibility of controlling COVID-19 outbreaks by isolation of cases and contacts. *The Lancet Global Health*, 2020.

[Hay11] Gillian R Hayes. The relationship of action research to human-computer interaction. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 18(3):1–20, 2011.

[HCP09] Erik Halvorson, Vincent Conitzer, and Ronald Parr. Multi-step multi-sensor hider-seeker games. In *International Joint Conference on Artificial Intelligence*, 2009.

[HHK$^+$17] N. Hamada, C. Hsu, R. Kurata, T. Suzuki, S. Ueda, and M. Yokoo. Strategy-proof school choice mechanisms with minimum quotas and initial endowments. *Artificial Intelligence*, 249:47–71, 2017.

[HK16] Xinran He and David Kempe. Robust influence maximization. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 885–894. ACM, 2016.

[HK18] Xinran He and David Kempe. Stability and robustness in influence maximization. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 12(6):1–34, 2018.

[HKL13] Nathan Oken Hodas, Farshad Kooti, and Kristina Lerman. Friendship paradox redux: Your friends are more interesting than you. *International AAAI Conference on Web and Social Media*, 13:8–10, 2013.

[HKP12] Christopher Hoffman, Matthew Kahle, and Elliot Paquette. Spectral gaps of random graphs and applications to random topology. *arXiv preprint arXiv:1201.0425*, 2012.

[HL09] L Jeff Hong and Guangwu Liu. Simulating sensitivities of conditional value at risk. *Management Science*, 55(2):281–293, 2009.

[HL16] Elad Hazan and Haipeng Luo. Variance-reduced and projection-free stochastic optimization. In *International Conference on Machine Learning*, pages 1263–1271, 2016.

[HLJA07] Andreas Handel, Ira M Longini Jr, and Rustom Antia. What is the best control strategy for multiple infectious disease outbreaks? *Proceedings of the Royal Society B: Biological Sciences*, 274(1611):833–837, 2007.

[HLP$^+$15] Nika Haghtalab, Aron Laszka, Ariel D Procaccia, Yevgeniy Vorobeychik, and Xenofon Koutsoukos. Monitoring stealthy diffusion. In *Proceedings of the 2015 IEEE International Conference on Data Mining (ICDM)*, pages 151–160. IEEE Computer Society, 2015.

[HM10] Eric Horvitz and Tom Mitchell. From data to knowledge to action: A global enabler for the 21st century. *Computing Community Consortium*, 1, 2010.

[HMT$^+$17] Jessica E Haberer, Nicholas Musinguzi, Alexander C Tsai, BM Bwana, C Muzoora, PW Hunt, JN Martin, DR Bangsberg, et al. Real-time electronic adherence monitoring plus follow-up improves adherence compared with standard electronic adherence monitoring. *AIDS (London, England)*, 31(1):169–171, 2017.

[HN15] Darrell Hoy and Evdokia Nikolova. Approximately optimal risk-averse routing policies via adaptive discretization. In *AAAI Conference on Artificial Intelligence*, 2015.

[Hof19] Anna Lauren Hoffmann. Where fairness fails: data, algorithms, and the limits of antidiscrimination discourse. *Information, Communication, & Society*, 22(7):900–915, 2019.

[Hor10] Eric Horvitz. From data to predictions and decisions: Enabling evidence-based healthcare. *Computing Community Consortium*, 6, 2010.

[HP07] Eric Horvitz and Tim Paek. Complementary computing: policies for transferring callers from dialog systems to human receptionists. *User Modeling and User-Adapted Interaction*, 17(1-2):159–182, 2007.

[HP15] Zhan Hu and Xizhe Peng. Household changes in contemporary China: An analysis based on the four recent censuses. *The Journal of Chinese Sociology*, 2(1):9, 2015.

[HPNP15] S. Han, V. M. Preciado, C. Nowzari, and G. J. Pappas. Data-driven network resource allocation for controlling spreading processes. *IEEE Transactions on Network Science and Engineering*, 2(4):127–138, Oct 2015.

[HRB$^+$19] Sophie Hilgard, Nir Rosenfeld, Mahzarin R Banaji, Jack Cao, and David C Parkes. Learning representations by humans, for humans. *arXiv preprint arXiv:1905.12686*, 2019.

[HRL$^+$20] Fiona P Havers, Carrie Reed, Travis Lim, Joel M Montgomery, John D Klena, Aron J Hall, Alicia M Fry, Deborah L Cannon, Cheng-Feng Chiang, Aridth Gibbons, et al. Seroprevalence of antibodies to SARS-CoV-2 in 10 sites in the united states, march 23-may 12, 2020. *JAMA Internal Medicine*, 2020.

[HSK17] Hamed Hassani, Mahdi Soltanolkotabi, and Amin Karbasi. Gradient methods for submodular maximization. In *Advances in Neural Information Processing Systems*, pages 5843–5853, 2017.

[HXKL16] Xinran He, Ke Xu, David Kempe, and Yan Liu. Learning influence functions from incomplete observations. In *Advances in Neural Information Processing Systems*, pages 2073–2081, 2016.

[HYL17] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, 2017.

[HZWJ19] Dan He, Xuying Zhang, Zhili Wang, and Yu Jiang. China fertility report, 2006–2016. *China Population and Development Studies*, 2(4):430–439, 2019.

[IIP14] IIPS. International Institute for Population Sciences. District level household and facility survey. New Delhi, India. `http://www.rchiips.org/obtainingdata.html`, 2014.

[IJB14] Rishabh K. Iyer, Stefanie Jegelka, and Jeff A. Bilmes. Monotone closure of relaxed constraints in submodular optimization: Connections between minimization and maximization. In *Conference on Uncertainty in Artificial Intelligence*, 2014.

[IJN^{+}20] Anita S Iyer, Forrest K Jones, Ariana Nodoushania, Meagan Kelly, Margaret Becker, Damien Slater, Rachel Mills, Erica Teng, Mohammad Kamruzzaman, Wilfredo F Garcia-Beltran, et al. Dynamics and significance of the antibody response to SARS-CoV-2 infection. *medRxiv*, 2020.

[IK21] Azra Ismail and Neha Kumar. Ai in global health: The view from the front lines. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–21, 2021.

[IM13] Garud Iyengar and Alfred Ka Chun Ma. Fast gradient descent method for mean-CVaR optimization. *Annals of Operations Research*, 205(1):203–212, 2013.

[IOAI16] Hiroaki Iwashita, Kotaro Ohori, Hirokazu Anai, and Atsushi Iwasaki. Simplifying urban network security games with cut-based graph contraction. In *International Conference on Autonomous Agents and Multiagent Systems*, 2016.

[Ita19] Italian National Institute of Statistics. Median age in Lombardy, 2019. `https://www4.istat.it/it/lombardia/dati?qt=gettable&dataset=DCIS_INDDEMOG1&dim=21,0,0`, Last Accessed: 2020-03-28.

[JCT13] Manish Jain, Vincent Conitzer, and Milind Tambe. Security scheduling for real-world networks. In *International Conference on Autonomous Agents and Multiagent Systems*, 2013.

[JGN^{+}17] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *International Conference on Machine Learning*, 2017.

[JGP17] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*, 2017.

[JHC12] Kyomin Jung, Wooram Heo, and Wei Chen. Irie: Scalable and robust influence maximization in social networks. In *IEEE International Conference on Data Mining (ICDM)*, pages 918–923. IEEE, 2012.

[JKV$^+$11] Manish Jain, Dmytro Korzhyk, Ondřej Vaněk, Vincent Conitzer, Michal Pěchouček, and Milind Tambe. A double oracle algorithm for zero-sum security games on graphs. In *International Conference on Autonomous Agents and Multiagent Systems*, 2011.

[JLR11] Svante Janson, Tomasz Luczak, and Andrzej Rucinski. *Random graphs*, volume 45. John Wiley & Sons, 2011.

[JR17] Akshay Jindal and Shrisha Rao. Agent-based modeling and simulation of mosquito-borne disease transmission. In *Proceedings of the 16th Conference on Autonomous Agents and Multiagent Systems*, pages 426–435, 2017.

[JSG$^+$03] Durell Johnson, Danielle Sholcosky, Karen Gabello, Robert Ragni, and Nicole Ogonosky. Sex differences in public restroom handwashing behavior associated with visual behavior prompts. *Perceptual and Motor Skills*, 97(3):805–810, 2003.

[KDF19] Amanda Kube, Sanmay Das, and Patrick J Fowler. Allocating interventions based on predicted outcomes: A case study on homelessness services. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 622–629, 2019.

[KDN$^+$17] Elias B. Khalil, Bistra Dilkina, George L. Nemhauser, Shabbir Ahmed, and Yufen Shao. Learning to run heuristics in tree search. In *International Joint Conference on Artificial Intelligence*, 2017.

[KDZ$^+$17] Elias B. Khalil, Hanjun Dai, Yuyu Zhang, Bistra Dilkina, and Le Song. Learning combinatorial optimization algorithms over graphs. In *Advances in Neural Information Processing Systems*, 2017.

[KG14] Andreas Krause and Daniel Golovin. Submodular function maximization., 2014.

[KHH12] Ece Kamar, Severin Hacker, and Eric Horvitz. Combining human and machine intelligence in large-scale crowdsourcing. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 467–474, 2012.

[KHS$^+$15] David A Kim, Alison R Hwong, Derek Stafford, D Alex Hughes, A James O'Malley, James H Fowler, and Nicholas A Christakis. Social network targeting to maximise population behaviour change: a cluster randomised controlled trial. *The Lancet*, 386(9989):145–153, 2015.

[KIPB08] Aaron A King, Edward L Ionides, Mercedes Pascual, and Menno J Bouma. Inapparent infections and cholera dynamics. *Nature*, 454(7206):877–880, 2008.

[KK98] George Karypis and Vipin Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing*, 20(1):359–392, 1998.

[KKT03] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 137–146. ACM, 2003.

[KLHK17] Mohammad Karimi, Mario Lucic, Hamed Hassani, and Andreas Krause. Stochastic submodular maximization: The case of coverage functions. In *Advances in Neural Information Processing Systems*, 2017.

[KLL$^+$20] Lauren M Kucirka, Stephen A Lauer, Oliver Laeyendecker, Denali Boon, and Justin Lessler. Variation in false-negative rate of reverse transcriptase polymerase chain reaction–based SARS-CoV-2 tests by time since exposure. *Annals of Internal Medicine*, 2020.

[KLSK20] Josh Katz, Denise Lu, and Margot Sanger-Katz. What is the real coronavirus death toll in each state? *The New York Times*, 2020. `https://www.nytimes.com/interactive/2020/05/05/us/coronavirus-death-toll-us.html`.

[KMGG08] Andreas Krause, H Brendan McMahan, Carlos Guestrin, and Anupam Gupta. Robust submodular observation selection. *Journal of Machine Learning Research*, 9(Dec):2761–2801, 2008.

[KMM$^+$13] Florent Krzakala, Cristopher Moore, Elchanan Mossel, Joe Neeman, Allan Sly, Lenka Zdeborová, and Pan Zhang. Spectral redemption in clustering sparse networks. *Proceedings of the National Academy of Sciences*, 110(52):20935–20940, 2013.

[KMS$^+$97] Jeffrey A Kelly, Debra A Murphy, Kathleen J Sikkema, Timothy L McAuliffe, Roger A Roffman, Laura J Solomon, Richard A Winett, Seth C Kalichman, and The Community HIV Prevention Research Collaborative. Randomised, controlled, community-level HIV-prevention intervention for sexual-risk behaviour among homosexual men in US cities. *The Lancet*, 350(9090):1500–1505, 1997.

[KOS00] Levent Koçkesen, Efe A Ok, and Rajiv Sethi. The strategic advantage of negatively interdependent preferences. *Journal of Economic Theory*, 92(2):274–299, 2000.

[KRD$^+$20] Adam Kucharski, Timothy Russell, Charlie Diamond, Yang Liu, John Edmunds, Sebastian Funk, and Rosalind Eggo. Early dynamics of transmission and control of COVID-19: a mathematical modelling study. *The Lancet Infectious Diseases*, 2020.

[KRG11] Andreas Krause, Alex Roper, and Daniel Golovin. Randomized sensing in adversarial environments. In *International Joint Conference on Artificial Intelligence*, 2011.

[KSNM09] Masahiro Kimura, Kazumi Saito, Ryohei Nakano, and Hiroshi Motoda. Finding influential nodes in a social network from information diffusion data. In *Social Computing and Behavioral Modeling*, pages 1–8. Springer, 2009.

[KSSW18] Dimitris Kalimeris, Yaron Singer, Karthik Subbian, and Udi Weinsberg. Learning diffusion using hyperparameters. In *International Conference on Machine Learning*, pages 2420–2428, 2018.

[KT12] Alex Kulesza and Ben Taskar. Determinantal point processes for machine learning. *Foundations and Trends in Machine Learning*, 5(2–3):123–286, 2012.

[KTLG20] Stephen Kissler, Christine Tedijanto, Marc Lipsitch, and Yonatan Grad. Social distancing strategies for curbing the COVID-19 epidemic. *medRxiv*, 2020.

[KTR$^+$17] Alp Kucukelbir, Dustin Tran, Rajesh Ranganath, Andrew Gelman, and David M Blei. Automatic differentiation variational inference. *The Journal of*

Machine Learning Research, 18(1):430–474, 2017.

[KvHW19] Wouter Kool, Herke van Hoof, and Max Welling. Attention, learn to solve routing problems! In *International Conference on Learning Representations*, 2019.

[KVKV12] Bernhard Korte, Jens Vygen, B Korte, and J Vygen. *Combinatorial optimization*, volume 2. Springer, 2012.

[KVM$^+$12] Jun-young Kwak, Pradeep Varakantham, Rajiv Maheswaran, Milind Tambe, Timothy Hayes, Wendy Wood, and Burcin Becerik-Gerber. Towards robust multi-objective optimization under model uncertainty for energy conservation. In *AAMAS workshop on Agent Technologies for Energy Systems (ATES)*, 2012.

[KW13] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2013.

[KW17] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

[KWS$^+$19] Jackson A Killian, Bryan Wilder, Amit Sharma, Vinod Choudhary, Bistra Dilkina, and Milind Tambe. Learning to prescribe interventions for tuberculosis patients using digital adherence data. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2430–2438, 2019.

[KWVR19] Naveena Karusala, Jennifer Wilson, Phebe Vayanos, and Eric Rice. Street-level realities of data practices in homeless services provision. *ACM Conference on Computer-Supported Cooperative Work and Social Computing*, 3:1–23, 2019.

[LBK$^+$10] Bruce Y Lee, Shawn T Brown, George W Korch, Philip C Cooley, Richard K Zimmerman, William D Wheaton, Shanta M Zimmer, John J Grefenstette, Rachel R Bailey, Tina-Marie Assi, et al. A computer simulation of vaccine prioritization, allocation, and rationing during the 2009 h1n1 influenza pandemic. *Vaccine*, 28(31):4875–4879, 2010.

[LBNZ17] Weihua Li, Quan Bai, Tung Doan Nguyen, and Minjie Zhang. Agent-based influence maintenance in social networks. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pages 1592–1594, 2017.

[LEK20] Yang Liu, Rosalind Eggo, and Adam Kucharski. Secondary attack rate and superspreading events for SARS-CoV-2. *The Lancet*, 2020.

[LFF15] Tobias Liboschik, Konstantinos Fokianos, and Roland Fried. tscount: An r package for analysis of count time series following generalized linear models. *Journal of Statistical Software*, 2015.

[LFWT18] Y. Li, J. Fan, Y. Wang, and K. L. Tan. Influence maximization on social graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 30(10):1852–1872, 2018.

[LGB$^+$20] Stephen Lauer, Kyra Grantz, Qifang Bi, Forrest Jones, Qulu Zheng, Hannah Meredith, Andrew Azman, Nicholas Reich, and Justin Lessler. The incubation period of coronavirus disease 2019 (COVID-19) from publicly reported confirmed cases: Estimation and application. *Annals of Internal Medicine*, 2020.

[LGW$^+$20] Qun Li, Xuhua Guan, Peng Wu, Xiaoye Wang, Lei Zhou, Yeqing Tong, Ruiqi Ren, Kathy Leung, Eric Lau, Jessica Wong, et al. Early transmission dynamics in Wuhan, China, of novel coronavirus–infected pneumonia. *New England Journal of Medicine*, 2020.

[LJJ15] Simon Lacoste-Julien and Martin Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. In *Advances in Neural Information Processing Systems*, pages 496–504, 2015.

[LKG$^+$07] Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, and Natalie Glance. Cost-effective outbreak detection in networks. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 420–429, 2007.

[LLDM09] Jure Leskovec, Kevin J Lang, Anirban Dasgupta, and Michael W Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123, 2009.

[LM12] Jure Leskovec and Julian J Mcauley. Learning to discover social circles in ego networks. In *Advances in neural information processing systems*, pages 539–547, 2012.

[LP17] David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.

[LPC+20] Ruiyun Li, Sen Pei, Bin Chen, Yimeng Song, Tao Zhang, Wan Yang, and Jeffrey Shaman. Substantial undocumented infection facilitates the rapid dissemination of novel coronavirus (SARS-CoV2). *Science*, 2020.

[LRG+18] Isaac Lage, Andrew Ross, Samuel J Gershman, Been Kim, and Finale Doshi-Velez. Human-in-the-loop interpretability prior. In *Advances in Neural Information Processing Systems*, 2018.

[LS12] Douglas A Luke and Katherine A Stamatakis. Systems science methods in public health: dynamics, networks, and agents. *Annual Review of Public Health*, 33:357–376, 2012.

[LSS+08] Chris J Lintott, Kevin Schawinski, Anže Slosar, Kate Land, Steven Bamford, Daniel Thomas, M Jordan Raddick, Robert C Nichol, Alex Szalay, Dan Andreescu, et al. Galaxy zoo: morphologies derived from visual inspection of galaxies from the sloan digital sky survey. *Monthly Notices of the Royal Astronomical Society*, 389(3):1179–1189, 2008.

[LTS+20] Quan-Xin Long, Xiao-Jun Tang, Qiu-Lin Shi, Qin Li, Hai-Jun Deng, Jun Yuan, Jie-Li Hu, Wei Xu, Yong Zhang, Fa-Jin Lv, et al. Clinical and immunological assessment of asymptomatic SARS-CoV-2 infections. *Nature Medicine*, pages 1–5, 2020.

[LUZ17] Marc T Law, Raquel Urtasun, and Richard S Zemel. Deep spectral clustering learning. In *International Conference on Machine Learning*, 2017.

[LVK16] Meghna Lowalekar, Pradeep Varakantham, and Akshat Kumar. Robust influence maximization. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 1395–1396, 2016.

[LWL+21] Daniel B Larremore, Bryan Wilder, Evan Lester, Soraya Shehata, James M Burke, James A Hay, Milind Tambe, Michael J Mina, and Roy Parker. Test sensitivity is secondary to frequency and turnaround time for COVID-19 surveillance. *Science Advances*, 2021.

[LZP+17] Alessandra Lugo, Piergiorgio Zuccaro, Roberta Pacifici, Giuseppe Gorini, Paolo Colombo, Carlo La Vecchia, and Silvano Gallus. Smoking in italy in 2015-2016: prevalence, trends, roll-your-own cigarettes, and attitudes towards incoming regulations. *Tumori Journal*, 103(4):353–359, 2017.

[Mac03] David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[Mae15] Takanori Maehara. Risk averse submodular utility maximization. *Operations Research Letters*, 43(5):526–529, September 2015.

[MB18] Arthur Mensch and Mathieu Blondel. Differentiable dynamic programming for structured prediction and attention. In *International Conference on Machine Learning*, 2018.

[MBF$^+$20] Chirag Modi, Vanessa Boehm, Simone Ferraro, George Stein, and Uros Seljak. Total covid-19 mortality in italy: Excess mortality and age dependence through time-series analysis. *medRxiv*, 2020.

[MCM$^+$17] Pietro Modesti, Maria Calabrese, Ilaria Marzotti, Hushao Bing, Danilo Malandrino, Maria Boddi, Sergio Castellani, and Dong Zhao. Prevalence, awareness, treatment, and control of hypertension among Chinese first-generation migrants and Italians in Prato, Italy: The CHIP study. *International Journal of Hypertension*, 2017.

[MDM$^+$18] Matthew H Morton, Amy Dworsky, Jennifer L Matjasko, Susanna R Curry, David Schlueter, Raúl Chávez, and Anne F Farrell. Prevalence and correlates of youth homelessness in the United States. *Journal of Adolescent Health*, 62(1):14–21, 2018.

[MGB03] H. Brendan McMahan, Geoffrey J. Gordon, and Avrim Blum. Planning in the presence of cost functions controlled by an adversary. In *International Conference on Machine Learning*, 2003.

[MIFK15] Atsushi Miyauchi, Yuni Iwamasa, Takuro Fukunaga, and Naonori Kakimura. Threshold influence model for allocating advertising budgets. In *International Conference on Machine Learning*, pages 1395–1404, 2015.

[Min78] Michel Minoux. Accelerated greedy algorithms for maximizing submodular set functions. *Optimization Techniques*, pages 234–243, 1978.

[MKOS09] Amy Medley, Caitlin Kennedy, Kevin O'Reilly, and Michael Sweat. Effectiveness of peer education interventions for HIV prevention in developing countries: a systematic review and meta-analysis. *AIDS Education and Prevention*, 21(3):181–206, 2009.

[MLVW17] V. Mirrokni, R. Paes Leme, A. Vladu, and S. C. Wong. Tight bounds for approximate Carathéodory and beyond. In *International Conference on Machine Learning*, pages 2440–2448, 2017.

[MM20] Maimuna S Majumder and Kenneth D Mandl. Early in the epidemic: impact of preprints on global discourse about covid-19 transmissibility. *The Lancet Global Health*, 8(5):e627–e630, 2020.

[MN11] Mayur Mohite and Y Narahari. Incentive compatible influence maximization in social networks and application to viral marketing. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 3*, pages 1081–1082, 2011.

[MNS15] Elchanan Mossel, Joe Neeman, and Allan Sly. Reconstruction and estimation in the planted partition model. *Probability Theory and Related Fields*, 162(3-4):431–461, 2015.

[MOS07] Renata Mansini, Włodzimierz Ogryczak, and M Grazia Speranza. Conditional value at risk and related linear programming models for portfolio optimization. *Annals of operations research*, 152(1):227–256, 2007.

[Mou20] Yascha Mounk. The extraordinary decisions facing Italian doctors, 2020. `https://www.theatlantic.com/ideas/archive/2020/03/who-gets-hospital-bed/607807/`.

[MP19] John O McGinnis and Russell G Pearce. The great disruption: How machine intelligence will transform the role of lawyers in the delivery of legal services. *Actual Probs. Econ. & L.*, page 1230, 2019.

[MR10] Rajeev Motwani and Prabhakar Raghavan. *Randomized algorithms*. Chapman & Hall/CRC, 2010.

[MRL12] Peter Mateyka, Melanie Rapino, and Liana Christin Landivar. Home-based workers in the United States, 2012. `https://www.census.gov/prod/2012pubs/p70-132.pdf`.

[MS12] Mahsa Maghami and Gita Sukthankar. Identifying influential agents for advertising in multi-agent markets. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 687–694, 2012.

[MTO15] Shodai Mihara, Sho Tsugawa, and Hiroyuki Ohsaki. Influence maximization problem for unknown social networks. In *International Conference on Advances in Social Network Analysis and Mining*, pages 1539–1546. ACM, 2015.

[Mun78] Yair Mundlak. On the pooling of time series and cross section data. *Econometrica*, pages 69–85, 1978.

[MVDB17] Ayan Mukhopadhyay, Yevgeniy Vorobeychik, Abhishek Dubey, and Gautam Biswas. Prioritized allocation of emergency responders based on a continuous-time incident prediction model. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 168–177, 2017.

[Nat12] National HCH Council. HIV/AIDS among persons experiencing homelessness. In Focus Quarterly Research Review, 2012.

[New06a] Mark EJ Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74(3):036104, 2006.

[New06b] Mark EJ Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006.

[NHG$^+$19] Azade Nazi, Will Hang, Anna Goldie, Sujith Ravi, and Azalia Mirhoseini. Gap: Generalizable approximate graph partitioning framework. *arXiv preprint arXiv:1903.00614*, 2019.

[NJLS09] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

[NMBC18] Vlad Niculae, André FT Martins, Mathieu Blondel, and Claire Cardie. Sparsemap: Differentiable sparse structured inference. In *International Conference on Machine Learning*, 2018.

[NMC05] Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *International Conference on Machine Learning*, pages 625–632, 2005.

[NPS15] Harikrishna Narasimhan, David C Parkes, and Yaron Singer. Learnability of influence in networks. In *Advances in Neural Information Processing Systems*, pages 3186–3194, 2015.

[oHH16] New York City Department of Health and Mental Hygiene. Hypertension in New York City: Disparities in prevalence, 2016.

[oHH20] NYC Department of Health and Mental Hygiene. Coronavirus disease 2019 (COVID-19) daily data summary, 2020. `https://www1.nyc.gov/assets/doh/downloads/pdf/imm/covid-19-daily-data-summary-deaths-05172020-1.pdf`.

[oLS19] U.S. Bureau of Labor Statistics. Labor force statistics from the current population survey, 2019. `https://www.bls.gov/cps/cpsaat08.htm`.

[OPVM19] Ziad Obermeyer, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453, 2019.

[ORB20] Graziano Onder, Giovanni Rezza, and Silvio Brusaferro. Case-fatality rate and characteristics of patients dying in relation to COVID-19 in Italy. *The Journal of the American Medical Association*, 2020.

[Org18] World Health Organization. *Global tuberculosis report 2018*. World Health Organization, 2018. Licence: CC BY-NC-SA 3.0 IGO.

[OSU16] James B. Orlin, Andreas Schulz, and Rajan Udwani. Robust monotone submodular function maximization. In *Conference on Integer Programming and Combinatorial Optimization (IPCO)*, 2016.

[OUS$^+$08] Avi Ostfeld, James G Uber, Elad Salomons, Jonathan W Berry, William E Hart, Cindy A Phillips, Jean-Paul Watson, et al. The battle of the water sensor networks (BWSN). *J. Water Resour. Plan. Manag.*, 134(6):556–568, 2008.

[OY17] Naoto Ohsaka and Yuichi Yoshida. Portfolio optimization for influence spread. In *WWW*, pages 977–985, 2017.

[P$^+$15] Daniel Paulin et al. Concentration inequalities for markov chains by marton couplings and spectral methods. *Electronic Journal of Probability*, 20, 2015.

[PAH15] Jean Pouget-Abadie and Thibaut Horel. Inferring graphs from cascades: A sparse recovery framework. *International Conference on Machine Learning*, 2015.

[PAI$^+$13] B Aditya Prakash, Lada Adamic, Theodore Iwashyna, Hanghang Tong, and Christos Faloutsos. Fractional immunization in networks. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 659–667. SIAM, 2013.

[PARS14] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 701–710. ACM, 2014.

[PBJ12] John Paisley, David Blei, and Michael Jordan. Variational bayesian inference with stochastic search. *arXiv preprint arXiv:1206.6430*, 2012.

[PCJ17] Kiesha Prem, Alex Cook, and Mark Jit. Projecting social contact matrices in 152 countries using contact surveys and demographic data. *PLoS Computational Biology*, 13(9):e1005697, 2017.

[PD09] Liliana Perez and Suzana Dragicevic. An agent-based approach for modeling dynamics of contagious disease spread. *International Journal of Health Geographics*, 8(1):50, 2009.

[Pen11] Xizhe Peng. China's demographic history and future challenges. *Science*, 333(6042):581–587, 2011.

[PG13] LA Prashanth and Mohammad Ghavamzadeh. Actor-critic algorithms for risk-sensitive MDPs. In *Advances in Neural Information Processing Systems*, pages 252–260, 2013.

[PHSE17] Sébastien Picault, Yu-Lin Huang, Vianney Sicard, and Pauline Ezanno. Enhancing sustainability of complex epidemiological models through a generic multilevel agent-based approach. In *International Joint Conference on Artificial Intelligence (IJCAI'2017)*, pages 374–380, 2017.

[PJB14] Adarsh Prasad, Stefanie Jegelka, and Dhruv Batra. Submodular meets structured: Finding diverse subsets in exponentially-large structured item sets. In *Advances in Neural Information Processing Systems*, 2014.

[Pla99] John C Platt. Using analytic QP and sparseness to speed training of support vector machines. In *Advances in Neural Information Processing Systems*, pages 557–563, 1999.

[PNR15] R. Pasumarthi, R. Narayanam, and B. Ravindran. Near optimal strategies for targeted marketing in social networks. In *International Conference on Autonomous Agents and Multiagent Systems*, 2015.

[Pol20] Politico. Italian doctors on coronavirus frontline face tough calls on whom to save, 2020. `https://www.politico.eu/article/coronavirus-italy-doctors-tough-calls-survival/`.

[PP08] Kaare Brandt Petersen and Michael Syskind Pedersen. The matrix cookbook. *Technical University of Denmark*, 7:15, 2008.

[PSA16] Elizabeth Levy Paluck, Hana Shepherd, and Peter M Aronow. Changing climates of conflict: A social network experiment in 56 schools. *Proceedings of the National Academy of Sciences*, 113(3):566–571, 2016.

[PTV$^+$10] B Aditya Prakash, Hanghang Tong, Nicholas Valler, Michalis Faloutsos, and Christos Faloutsos. Virus propagation on time-varying networks: Theory and immunization algorithms. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 99–114. Springer, 2010.

[PX19] Mark Parascandola and Lin Xiao. Tobacco and the lung cancer epidemic in china. *Translational Lung Cancer Research*, 8:S21–S30, 2019.

[RA20] Julien Riou and Christian Althaus. Pattern of early human-to-human transmission of Wuhan 2019 novel coronavirus (2019-nCoV), December 2019 to January 2020. *Eurosurveillance*, 25(4), 2020.

[RAA$^+$18] Gregory Roth, Degu Abate, Kalkidan Hassen Abate, Solomon Abay, Cristiana Abbafati, Nooshin Abbasi, Hedayat Abbastabar, Foad Abd-Allah, Jemal Abdela, Ahmed Abdelalim, et al. Global, regional, and national age-sex-specific mortality for 282 causes of death in 195 countries and territories, 1980–2017: A systematic analysis for the Global Burden of Disease Study 2017. *The Lancet*, 392(10159):1736–1788, 2018.

[RBAMM12] Eric Rice, Anamika Barman-Adhikari, Norweeta G Milburn, and William Monro. Position-specific HIV risk in a large network of homeless youths. *American Journal of Public Health*, 102(1):141–147, 2012.

[RBC+19] Maithra Raghu, Katy Blumer, Greg Corrado, Jon Kleinberg, Ziad Obermeyer, and Sendhil Mullainathan. The algorithmic automation problem: Prediction, triage, and human effort. *arXiv preprint arXiv:1903.12220*, 2019.

[RGM+15] Theodoros Rekatsinas, Saurav Ghosh, Sumiko R Mekaru, Elaine O Nsoesie, John S Brownstein, Lise Getoor, and Naren Ramakrishnan. Sourceseer: Forecasting rare disease outbreaks using multiple data sources. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 379–387. SIAM, 2015.

[RHA+20] Timothy Russell, Joel Hellewell, Sam Abbott, Christopher Jarvis, Kevin van Zandvoort, Stefan Flasche, Rosalind Eggo, John Edmunds, and Adam Kucharski. Using a delay-adjusted case fatality ratio to estimate under-reporting, 2020. `https://cmmid.github.io/topics/covid19/severity/global_cfr_estimates.html`. Accessed 03-26-20.

[RMBAY10] Eric Rice, William Monro, Anamika Barman-Adhikari, and Sean D Young. Internet use, social networking, and HIV/AIDS risk for homeless adolescents. *Journal of Adolescent Health*, 47(6):610–613, 2010.

[RMRB07] Eric Rice, Norweeta G Milburn, and Mary Jane Rotheram-Borus. Pro-social and problematic social network influences on HIV/AIDS risk behaviours among newly homeless youth in los angeles. *AIDS Care*, 19(5):697–704, 2007.

[RNT10] RNTCP. Training module for medical practitioners. New Delhi, India: Revised National Tuberculosis Control Programme, Ministry of Health and Family Welfare, 2010.

[RNT16] RNTCP. Revised national tuberculosis control programme annual status report. New Delhi, India: Ministry of Health and Family Welfare. `http://tbcindia.nic.in/showfile.php?lid=3180`, 2016.

[RSS+20] Camilla Rothe, Mirjam Schunk, Peter Sothmann, Gisela Bretzel, Guenter Froeschl, Claudia Wallrauch, Thorbjörn Zimmer, Verena Thiel, Christian Janke, Wolfgang Guggemos, et al. Transmission of 2019-nCoV infection from an asymptomatic contact in Germany. *New England Journal of Medicine*, 2020.

[RTC+12] Eric Rice, Eve Tulbert, Julie Cederbaum, Anamika Barman Adhikari, and Norweeta G Milburn. Mobilizing homeless youth for HIV prevention. *Health Education Research*, 27(2):226–236, 2012.

[RU00] R Tyrrell Rockafellar and Stanislav Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2:21–42, 2000.

[Sam16] Thomas Sampson. Assignment reversals: Trade, skill allocation and wage inequality. *J. Econ. Theory*, 163:365–409, 2016.

[SAPV15] Sudip Saha, Abhijin Adiga, B Aditya Prakash, and Anil Kumar S Vullikanti. Approximation algorithms for reducing the spectral radius to control epidemic spread. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 568–576. SIAM, 2015.

[SB98] Richard S Sutton and Andrew G Barto. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

[SBGF15] SC Suen, E Bendavid, and JD Goldhaber-Fiebert. Cost-effectiveness of improvements in diagnosis and treatment accessibility for tuberculosis control in india. *The International Journal of Tuberculosis and Lung Disease*, 19(9):1115–1124, 2015.

[SCFD17] C. Schumann, S. N. Counts, J. S. Foster, and J. P. Dickerson. The diverse cohort selection problem: Multi-armed bandits with varied pulls. *CoRR*, abs/1709.03441, 2017.

[SDG$^+$15] Lora L Sabin, Mary Bachman DeSilva, Christopher J Gill, Zhong Li, Taryn Vian, Xie Wubin, Cheng Feng, Xu Keyi, Lan Guanghua, Jessica E Haberer, et al. Improving adherence to antiretroviral therapy with triggered real time text message reminders: the china through technology study (cats). *Journal of Acquired Immune Deficiency Syndromes*, 69(5):551, 2015.

[SdMM$^+$18] Ramnath Subbaraman, Laura de Mondesert, Angella Musiimenta, Madhukar Pai, Kenneth H Mayer, Beena E Thomas, and Jessica Haberer. Digital adherence technologies for the management of tuberculosis therapy: mapping the landscape and research priorities. *BMJ global health*, 3(5):e001018, 2018.

[SEM14] Samarth Swarup, Stephen G Eubank, and Madhav V Marathe. Computational epidemiology as a challenge domain for multiagent systems. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 1173–1176, 2014.

[SGM$^+$20] Jeffrey Seow, Carl Graham, Blair Merrick, Sam Acors, Kathryn JA Steel, Oliver

Hemmings, Aoife O'Bryne, Neophytos Kouphou, Suzanne Pickering, Rui Galao, et al. Longitudinal evaluation and decline of antibody responses in SARS-CoV-2 infection. *medRxiv*, 2020.

[SHL15] Chonggang Song, Wynne Hsu, and Mong Li Lee. Node immunization over infectious period. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 831–840. ACM, 2015.

[SHS18] E. Segal-Halevi and W. Suksompong. Democratic fair allocation of indivisible goods. In *International Joint Conference on Artificial Intelligence*, pages 482–488, 2018.

[SJ17] Matthew Staib and Stefanie Jegelka. Robust budget allocation via continuous submodular functions. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 3230–3240, 2017.

[SKB$^+$18] M. Schlichtkrull, T. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, 2018.

[SKIK14] Tasuku Soma, Naonori Kakimura, Kazuhiro Inaba, and Ken-ichi Kawarabayashi. Optimal budget allocation: Theoretical guarantee and efficient algorithm. In *International Conference on Machine Learning*, pages 351–359, 2014.

[SNB$^+$08] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93, 2008.

[Sou20] South China Morning Post. Coronavirus: China's first confirmed COVID-19 case traced back to November 17, 2020. `https://www.scmp.com/news/china/society/article/3074991/coronavirus-chinas-first-confirmed-covid-19-case-traced-back`.

[SPRG12] Tiziano Squartini, Francesco Picciolo, Franco Ruzzenenti, and Diego Garlaschelli. Reciprocity of weighted networks. *Scientific Reports*, 2012.

[SSL$^+$18] Uri Shaham, Kelly Stanton, Henry Li, Boaz Nadler, Ronen Basri, and Yuval Kluger. Spectralnet: Spectral clustering using deep neural networks. In

International Conference on Learning Representations, 2018.

[ST13] Daniel A Spielman and Shang-Hua Teng. A local clustering algorithm for massive graphs and its application to nearly linear time graph partitioning. *SIAM Journal on Computing*, 42(1):1–26, 2013.

[Sta18a] Statista. Biennial average number of household members in Italy from 2012 to 2018, 2018. `https://www.statista.com/statistics/671945/biennial-average-number-of-families-with-children-italy/`, Last Accessed: 2020-03-28.

[Sta18b] Statista. Household structures in Italy in 2018, 2018. `https://www.statista.com/statistics/730604/family-structures-italy/`, Last Accessed: 2020-03-28.

[Sta18c] Statista. Number of couples with children in Italy from 2012 to 2018, by number of children, 2018. `https://www.statista.com/statistics/570106/number-of-couples-with-children-italy/`, Last Accessed: 2020-03-28.

[Sta18d] Statista. Number of single parents in Italy from 2011 to 2018, by number of children, 2018.

[Sta18e] Statista. Number of single-person households in Italy from 2012 to 2018, 2018. `https://www.statista.com/statistics/728061/number-of-single-person-households-italy/`, Last Accessed: 2020-03-28.

[Sta20] Stan Development Team. RStan: the R interface to Stan, 2020. R package version 2.21.2.

[Sta21] Stan Development Team. Stan modeling language users guide and reference manual, 2.27, 2021.

[Suk18] W. Suksompong. Approximate maximin shares for groups of agents. *Mathematical Social Sciences*, 92:40–47, 2018.

[SVI$^+$16] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In

Conference on Computer Vision and Pattern Recognition, pages 2818–2826, 2016.

[SVK20] Kevin Systrom, Thomas Vladek, and Mike Krieger. rt.live (september 5, 2020). `https://github.com/rtcovidlive/covid-model`, 2020.

[SWJ19] Matthew Staib, Bryan Wilder, and Stefanie Jegelka. Distributionally robust submodular maximization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 506–516. PMLR, 2019.

[SY15] Tasuku Soma and Yuichi Yoshida. A generalization of submodular cover via the diminishing return property on the integer lattice. In *Advances in Neural Information Processing Systems*, pages 847–855, 2015.

[SZL15] John A Schneider, A Ning Zhou, and Edward O Laumann. A new HIV prevention network approach: sociometric peer change agent selection. *Social Science & Medicine*, 125:192–202, 2015.

[TAIK18] Sarah Tan, Julius Adebayo, Kori Inkpen, and Ece Kamar. Investigating human+ machine complementarity for recidivism predictions. *arXiv preprint arXiv:1808.09123*, 2018.

[TCGM15] Aviv Tamar, Yinlam Chow, Mohammad Ghavamzadeh, and Shie Mannor. Policy gradient for coherent risk measures. In *Advances in Neural Information Processing Systems*, 2015.

[TCH$^+$20] Nenad Tomašev, Julien Cornebise, Frank Hutter, Shakir Mohamed, Angela Picciariello, Bec Connelly, Danielle CM Belgrave, Daphne Ezer, Fanny Cachat van der Haert, Frank Mugisha, et al. AI for social good: unlocking the opportunity for positive impact. *Nature Communications*, 11(1):1–6, 2020.

[Tea20] CDC COVID-19 Response Team. Severe outcomes among patients with coronavirus disease 2019 (COVID-19)—United States, February 12–March 16, 2020, 2020. `https://www.cdc.gov/mmwr/volumes/69/wr/mm6912e2.htm`.

[TG18] Lifu Tu and Kevin Gimpel. Learning approximate inference networks for structured prediction. In *International Conference on Learning Representations*, 2018.

[TGC$^+$14] Fei Tian, Bin Gao, Qing Cui, Enhong Chen, and Tie-Yan Liu. Learning deep representations for graph clustering. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[TGS$^+$05] Aleyamma Thomas, PG Gopi, T Santha, V Chandrasekaran, R Subramani, N Selvakumar, SI Eusuff, K Sadacharam, and PR Narayanan. Predictors of relapse among pulmonary tuberculosis patients treated in a dots programme in south india. *The International Journal of Tuberculosis and Lung Disease*, 9(5):556–561, 2005.

[Tit16] Michalis Titsias. One-vs-each approximation to softmax for scalable estimation of probabilities. In *Advances in Neural Information Processing Systems*, 2016.

[TK74] Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131, 1974.

[TLH$^+$11] T. Todo, R. Li, X. Hu, T. Mouri, A. Iwasaki, and M. Yokoo. Generalizing envy-freeness toward groups of agents. In *International Joint Conference on Artificial Intelligence*, pages 386–392, 2011.

[TLR08] Michalis K Titsias, Neil Lawrence, and Magnus Rattray. Markov chain monte carlo algorithms for gaussian processes. *Inference and Estimation in Probabilistic Time-Series Models*, 9, 2008.

[TO09] Hiroya Takamura and Manabu Okumura. Text summarization model based on maximum coverage problem and its variant. In *Conference of the European Chapter of the Association for Computational Linguistics*, 2009.

[TO17] Yukako Tatsumi and Takayoshi Ohkubo. Hypertension with diabetes mellitus: significance from an epidemiological perspective for Japanese. *Hypertension Research*, 40(9):795–806, 2017.

[TSK18] Sebastian Tschiatschek, Aytunc Sahin, and Andreas Krause. Differentiable submodular maximization. In *International Joint Conference on Artificial Intelligence*, 2018.

[TSvG$^+$19] RN Thompson, JE Stockwin, Rolina D van Gaalen, JA Polonsky, ZN Kamvar, PA Demarsh, Elisabeth Dahlqwist, Siyang Li, Eve Miguel, Thibaut Jombart, et al. Improved inference of time-varying reproduction numbers during infectious disease outbreaks. *Epidemics*, 29:100356, 2019.

[TWL+16] Suo-Yi Tan, Jun Wu, Linyuan Lü, Meng-Jun Li, and Xin Lu. Efficient network disintegration under incomplete information: the comic effect of link prediction. *Scientific Reports*, 6:22916, 2016.

[TWR+19] Alan Tsang, Bryan Wilder, Eric Rice, Milind Tambe, and Yair Zick. Group-fairness in influence maximization. In *International Joint Conference on Artificial Intelligence*, 2019.

[TXS14] Youze Tang, Xiaokui Xiao, and Yanchen Shi. Influence maximization: Near-optimal time complexity meets practical efficiency. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, pages 75–86. ACM, 2014.

[TYK+10] Jason Tsai, Zhengyu Yin, Jun-young Kwak, David Kempe, Christopher Kiekintveld, and Milind Tambe. Urban security: Game-theoretic resource allocation in networked physical domains. In *National Conference on Artificial Intelligence (AAAI)*, 2010.

[Udw18] R. Udwani. Multi-objective maximization of monotone submodular functions with cardinality constraint. In *Advances in Neural Information Processing Systems*, pages 9513–9524, 2018.

[UKBM11] Johan Ugander, Brian Karrer, Lars Backstrom, and Cameron Marlow. The anatomy of the facebook social graph. *arXiv preprint arXiv:1111.4503*, 2011.

[UN15] UN. United Nations Statistics Division Demographic Statistics. `http://data.un.org/Data.aspx?d=POP&f=tableCode%3A22`, 2015.

[Una20] Unacast. Social distancing scoreboard, 2020. `https://www.unacast.com/covid19/social-distancing-scoreboard`.

[Uni19] United Nations. World population prospects 2019, 2019.

[VB10] Paolo Viappiani and Craig Boutilier. Optimal bayesian recommendation sets and myopically optimal choice query sets. In *Advances in Neural Information Processing Systems*, 2010.

[VdDLM99] Pauline Van den Driessche, Michael Li, and James Muldowney. Global stability of SEIRS models in epidemiology. *Canadian Applied Mathematics*

Quarterly, 7:409–425, 1999.

[Vek16] Arjun Vekariya. Implementation of camelyon'16 grand challenge. `https://github.com/arjunvekariyagithub/camelyon16-grand-challenge`, 2016.

[VFJ15] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. In *Advances in Neural Information Processing Systems*, 2015.

[VL07] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[VMB$^+$18] Arash Vahdat, William Macready, Zhengbing Bian, Amir Khoshaman, and Evgeny Andriyash. Dvae++: Discrete variational autoencoders with overlapping transformations. In *International Conference on Machine Learning*, pages 5035–5044, 2018.

[VOD$^+$20] Robert Verity, Lucy Okell, Ilaria Dorigatti, Peter Winskill, Charles Whittaker, Natsuko Imai, Gina Cuomo-Dannenburg, Hayley Thompson, Patrick Walker, Han Fu, et al. Estimates of the severity of COVID-19 disease. *medRxiv*, 2020.

[Von08] Jan Vondrák. Optimal approximation for the submodular welfare problem in the value oracle model. In *ACM Symposium on Theory of Computing*, pages 67–74, 2008.

[VP07] Thomas W Valente and Patchareeya Pumpuang. Identifying opinion leaders to promote behavior change. *Health Education & Behavior*, 34(6):881–896, 2007.

[WC17] Qinshi Wang and Wei Chen. Improving regret bounds for combinatorial semi-bandits with probabilistically triggered arms and its applications. In *Advances in Neural Information Processing Systems*, pages 1161–1171, 2017.

[WCD$^+$18] Meredith Whittaker, Kate Crawford, Roel Dobbe, Genevieve Fried, Elizabeth Kaziunas, Varoon Mathur, Sarah Mysers West, Rashida Richardson, Jason Schultz, and Oscar Schwartz. *AI Now Report 2018*. AI Now Institute at New York University New York, 2018.

[WCK$^+$20] Bryan Wilder, Marie Charpignon, Jackson A Killian, Han-Ching Ou, Aditya Mate, Shahin Jabbari, Andrew Perrault, Angel N Desai, Milind Tambe, and

Maimuna S Majumder. Modeling between-population variation in covid-19 dynamics in hubei, lombardy, and new york city. *Proceedings of the National Academy of Sciences*, 117(41):25904–25910, 2020.

[WCM19] Lijing Wang, Jiangzhuo Chen, and Madhav Marathe. Defsi: Deep learning based epidemic forecasting with synthetic information. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9607–9612, 2019.

[WCSX10] Yu Wang, Gao Cong, Guojie Song, and Kunqing Xie. Community-based greedy algorithm for mining top-k influential nodes in mobile social networks. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1039–1048. ACM, 2010.

[WCZ$^+$18] Zengwu Wang, Zuo Chen, Linfeng Zhang, Xin Wang, Guang Hao, Zugui Zhang, Lan Shao, Ye Tian, Ying Dong, Congyi Zheng, Jiali Wang, Manlu Zhu, William Weintraub, and Runlin Gao. Status of hypertension in China. *Circulation*, 137(22):2344–2356, 2018.

[WDT19] Bryan Wilder, Bistra Dilkina, and Milind Tambe. Melding the data-decisions pipeline: Decision-focused learning for combinatorial optimization. In *AAAI Conference on Artificial Intelligence*, 2019.

[WEDT19] Bryan Wilder, Eric Ewing, Bistra Dilkina, and Milind Tambe. End to end learning and optimization on graphs. In *Advances in Neural and Information Processing Systems*, 2019.

[WHK20] Bryan Wilder, Eric Horvitz, and Ece Kamar. Learning to complement humans. In *International Joint Conference on Artificial Intelligence*, 2020.

[WHO15a] WHO. World Health Organization. Life Tables. `http://www.who.int/gho/mortality_burden_disease/life_tables/life_tables/en/`, 2015.

[WHO15b] WHO. World Health Organization. Tuberculosis country profiles. `http://www.who.int/tb/country/data/profiles/en/`, 2015.

[Wil18a] Bryan Wilder. Equilibrium computation and robust optimization in zero sum games with submodular structure. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, 2018.

[Wil18b] Bryan Wilder. Risk-sensitive submodular optimization. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, 2018.

[WIRT18] Bryan Wilder, Nicole Immorlica, Eric Rice, and Milind Tambe. Maximizing influence in an unknown social network. In *AAAI Conference on Artificial Intelligence*, 2018.

[WKG$^+$16] Dayong Wang, Aditya Khosla, Rishab Gargeya, Humayun Irshad, and Andrew H Beck. Deep learning for identifying metastatic breast cancer. *arXiv preprint arXiv:1606.05718*, 2016.

[WKVV17] Zheng Wen, Branislav Kveton, Michal Valko, and Sharan Vaswani. Online influence maximization under independent cascade model with semi-bandit feedback. In *Advances in neural information processing systems*, pages 3022–3032, 2017.

[WMT21] Bryan Wilder, Michael J Mina, and Milind Tambe. Tracking disease outbreaks from sparse data with Bayesian inference. In *AAAI Conference on Artificial Intelligence*, 2021.

[WOdlHT18] B. Wilder, H. Ou, K. de la Haye, and M. Tambe. Optimizing network structure for preventative health. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 841–849, 2018.

[WOVD$^+$21] Bryan Wilder, Laura Onasch-Vera, Graham Diguiseppi, Robin Petering, Chyna Hill, Amulya Yadav, Eric Rice, and Milind Tambe. Clinical trial of an ai-augmented intervention for hiv prevention in youth experiencing homelessness. In *AAAI Conference on Artificial Intelligence*, 2021.

[WOVH$^+$18] Bryan Wilder, Laura Onasch-Vera, Juliana Hudson, Jose Luna, Nicole Wilson, Robin Petering, Darlene Woo, Milind Tambe, and Eric Rice. End-to-end influence maximization in the field. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems*, pages 1414–1422, 2018.

[WPS18] Y. Wen, W. Peng, and H. Shuai. Maximizing social influence on target users. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2018.

[WRC$^+$20] W Joost Wiersinga, Andrew Rhodes, Allen C Cheng, Sharon J Peacock, and Hallie C Prescott. Pathophysiology, transmission, diagnosis, and treatment of coronavirus disease 2019 (covid-19): a review. *Journal of the American Medical Association*, 2020.

[WST18] Bryan Wilder, Sze-Chuan Suen, and Milind Tambe. Preventing infectious disease in dynamic populations under uncertainty. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[WT04] Jacco Wallinga and Peter Teunis. Different epidemic curves for severe acute respiratory syndrome reveal similar impacts of control measures. *American Journal of Epidemiology*, 160(6):509–516, 2004.

[WWC$^+$05] Peter J White, Helen Ward, Jackie A Cassell, Catherine H Mercer, and Geoff P Garnett. Vicious and virtuous circles in the dynamics of infectious disease and the provision of health care: gonorrhea in britain as an example. *The Journal of Infectious Diseases*, 192(5):824–836, 2005.

[WWG14] Brady T West, Kathleen B Welch, and Andrzej T Galecki. *Linear mixed models: a practical guide using statistical software*. CRC Press, 2014.

[WXQ$^+$06] Hao Wang, Haiyong Xie, Lili Qiu, Yang Richard Yang, Yin Zhang, and Albert Greenberg. Cope: traffic engineering in dynamic networks. In *Sigcomm*, volume 6, page 194, 2006.

[WYI$^+$17a] Bryan Wilder, Amulya Yadav, Nicole Immorlica, Eric Rice, and Milind Tambe. Uncharted but not uninfluenced: Influence maximization with an uncertain network. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 740–748, 2017.

[WYI$^+$17b] Bryan Wilder, Amulya Yadav, Nicole Immorlica, Eric Rice, and Milind Tambe. Uncharted but not uninfluenced: Influence maximization with an uncertain network. In *Proceedings of the 16th Conference on Autonomous Agents and Multiagent Systems*, pages 1305–1313, 2017.

[XDF$^+$16] Yexiang Xue, Ian Davies, Daniel Fink, Christopher Wood, and Carla P Gomes. Avicaching: A two stage game for bias reduction in citizen science. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 776–785, 2016.

[XdGM$^+$20] Bo Xu, Alomía de Gutiérrez, Sumiko Mekaru, Kara Sewalk, Lauren Goodwin, Alyssa Loskill, Emily Cohn, Yulin Hswen, Sarah Hill, María Mercedes Cobo, Alexander Zarebski, et al. Epidemiological data from the COVID-19 outbreak, real-time case information. *Scientific Data*, 7, 2020.

[XGF16] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International Conference on Machine Learning*, pages 478–487, 2016.

[Xu16] Haifeng Xu. The mysteries of security games: Equilibrium computation becomes combinatorial algorithm design. In *ACM Conference on Economics and Computation*, 2016.

[XWH+13] Yu Xu, Limin Wang, Jiang He, Yufang Bi, Mian Li, Tiange Wang, Linhong Wang, Yong Jiang, Meng Dai, Jieli Lu, Min Xu, Yichong Li, Nan Hu, Jianhong Li, Shengquan Mi, Chung-Shiuan Chen, et al. Prevalence and Control of Diabetes in Chinese Adults. *The Journal of the American Medical Association*, 310(9):948–959, 09 2013.

[Yah07] Yahoo. Yahoo! webscope dataset ydata-ysm-advertiser-bids-v1 0. `http://research.yahoo.com/Academic_Relations`, 2007.

[YCH+16] Liang Yang, Xiaochun Cao, Dongxiao He, Chuan Wang, Xiao Wang, and Weixiong Zhang. Modularity based community detection with deep learning. In *International Joint Conference on Artificial Intelligence*, volume 16, pages 2252–2258, 2016.

[YCXJ+16] Amulya Yadav, Hau Chan, Albert Xin Jiang, Haifeng Xu, Eric Rice, and Milind Tambe. Using social networks to aid homeless shelters: Dynamic influence maximization under uncertainty. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 740–748, 2016.

[YG12] Bowen Yan and Steve Gregory. Detecting community structure in networks using edge prediction methods. *Journal of Statistical Mechanics: Theory and Experiment*, 2012(09):P09008, 2012.

[YHWH03] Jane M Young, Michael J Hollands, Jeanette Ward, and CD'Arcy J Holman. Role for opinion leaders in promoting evidence-based surgery. *Archives of Surgery*, 138(7):785–791, 2003.

[YJTO11] Zhengyu Yin, Manish Jain, Milind Tambe, and Fernando Ordónez. Risk-averse strategies for security games with execution and observational uncertainty.

In *AAAI Conference on Artificial Intelligence*, 2011.

[YKRW11] Sheena Yau, Roy H Kwon, J Scott Rogers, and Desheng Wu. Financial and operational decisions in the electricity sector. *Int J Prod Econ*, 134(1):67–77, 2011.

[YLIS20] Karen Yourish, K.K. Rebecca Lai, Danielle Ivory, and Mitch Smith. One-third of all U.S. coronavirus deaths are nursing home residents or workers. *The New York Times*, 2020. `https://www.nytimes.com/interactive/2020/05/09/us/coronavirus-cases-nursing-homes-us.html`.

[YN13] Jia Yuan Yu and Evdokia Nikolova. Sample complexity of risk-averse bandit-arm selection. In *International Joint Conference on Artificial Intelligence*, pages 2576–2582, 2013.

[YR11] Sean D Young and Eric Rice. Online social networking technologies, HIV knowledge, and sexual risk and testing behaviors among homeless youth. *AIDS and Behavior*, 15(2):253–260, 2011.

[YSK15] Shihao Yang, Mauricio Santillana, and Samuel C Kou. Accurate estimation of influenza epidemics using google search data via argo. *Proceedings of the National Academy of Sciences*, 112(47):14473–14478, 2015.

[YWR$^+$17] Amulya Yadav, Bryan Wilder, Eric Rice, Robin Petering, Jaih Craddock, Amanda Yoshioka-Maxwell, Mary Hemler, Laura Onasch-Vera, Milind Tambe, and Darlene Woo. Influence maximization in the field: The arduous journey from emerging to deployed application. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pages 150–158, 2017.

[YWR$^+$18] A. Yadav, B. Wilder, E. Rice, R. Petering, J. Craddock, A. Yoshioka-Maxwell, M. Hemler, L. Onasch-Vera, M. Tambe, and D. Woo. Bridging the gap between theory and practice in influence maximization: Raising awareness about HIV among homeless youth. In *International Joint Conference on Artificial Intelligence*, pages 5399–5403, 2018.

[YYM$^+$18] Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. In *Advances in Neural Information Processing Systems*, pages 4800–4810, 2018.

[ZAS+16] Yao Zhang, Abhijin Adiga, Sudip Saha, Anil Vullikanti, and B Aditya Prakash. Near-optimal algorithms for controlling propagation at group scale on networks. *IEEE Transactions on Knowledge and Data Engineering*, 28(12):3339–3352, 2016.

[ZAVP15] Yao Zhang, Abhijin Adiga, Anil Vullikanti, and B Aditya Prakash. Controlling propagation at group scale on networks. In *2015 IEEE International Conference on Data Mining*, pages 619–628. IEEE, 2015.

[ZC18] Muhan Zhang and Yixin Chen. Link prediction based on graph neural networks. In *Advances in Neural Information Processing Systems*, 2018.

[ZJRP+15] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international Conference on Computer Vision*, pages 1529–1537, 2015.

[ZLB20] Yunfeng Zhang, Q Vera Liao, and Rachel KE Bellamy. Effect of confidence and explanation on accuracy and trust calibration in AI-assisted decision making. In *ACM Conference on Fairness, Accountability, and Transparency*, 2020.

[ZLL+20] Juanjuan Zhang, Maria Litvinova, Yuxia Liang, Yan Wang, Wei Wang, Shanlu Zhao, Qianhui Wu, Stefano Merler, Cecile Viboud, Alessandro Vespignani, et al. Changes in contact patterns shape the dynamics of the COVID-19 outbreak in China. *Science*, 2020.

[ZP14a] Yao Zhang and B Aditya Prakash. Dava: Distributing vaccines over networks under prior information. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 46–54. SIAM, 2014.

[ZP14b] Yao Zhang and B Aditya Prakash. Scalable vaccine distribution in large graphs given uncertain data. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 1719–1728. ACM, 2014.

[ZPV15] Haifeng Zhang, Ariel D Procaccia, and Yevgeniy Vorobeychik. Dynamic influence maximization under increasing returns to scale. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 949–957, 2015.

[ZY20] Christoph Zimmer and Reza Yaesoubi. Influenza forecasting framework based on Gaussian processes. In *International Conference on Machine Learning*, pages 11671–11679. PMLR, 2020.

[ZYD$^+$20] Fei Zhou, Ting Yu, Ronghui Du, Guohui Fan, Ying Liu, Zhibo Liu, Jie Xiang, Yeming Wang, Bin Song, Xiaoying Gu, et al. Clinical course and risk factors for mortality of adult inpatients with COVID-19 in Wuhan, China: a retrospective cohort study. *The Lancet*, 2020.

Appendix A

Appendix to Chapter 1

A.1 Theoretical analysis of ARISEN

In this section, we present proofs of our guarantees for the performance of ARISEN.

A.1.1 Preliminaries

We study influence maximization using local information on a graph drawn from the stochastic block model (SBM). There is a fixed vertex set V, where $|V| = n$ is known to the algorithm. The vertices are partitioned into communities $C_1 ... C_L$ where each $C_i \subseteq V$. We assume that the communities are ordered as $|C_1| \geq |C_2| \geq .... \geq |C_L|$ The set of edges is sampled according to the following process:

1. Each edge (u, v) where u and v belong to the same community is independently present with probability p_w.

2. Each edge (u, v) where u and v belong to different communities is independently present with probability p_b.

Influence propagates according to the independent cascade model (ICM) where each edge has equal propagation probability q. This process can be viewed as follows. Each edge in the graph is independently kept with probability q and discarded with probability $1 - q$.

Then, a node is influenced by a given seed set if it lies in the same connected component as a seed. The intuition for this view (which originated with Kempe et al. [KKT03]) is that flipping all of the process's random coins in advance is equivalent to flipping them one at a time, as each node is influenced. The SBM and ICM can thus been seen as jointly inducing a graph where each within-community edge is present with probability $p_w q$ and each between-community edge is present with probability $p_b q$.

The algorithm has a budget of K nodes which it may select as seeds. We assume without loss of generality that $L \geq K$. If $L < K$, then all claims follow by analyzing the expected utility on $C_1 ... C_L$ instead of $C_1 ... C_K$. Let $f_E(S)$ give the expected number of nodes influenced in the independent cascade model by the set of nodes S when the set of realized edges are E. Let $OPT(E)$ give the greatest influence spread using any subset of K nodes when the realized edges are E. Our algorithm is denoted by $\mathcal{A}$; the set containing its selections given edge set E is denoted by $\mathcal{A}(E)$. Note that since $\mathcal{A}$ is randomized, $\mathcal{A}(E)$ is itself a random variable. We aim to prove that

$$\mathbb{E}[f_E(\mathcal{A}(E))] \geq \alpha \, \mathbb{E}[OPT(E)]$$

for some approximation ratio α. The expectations range over the randomness in the realization of E and the decisions of $\mathcal{A}$. Let $OPT = \mathbb{E}[OPT(E)]$.

We now state some facts about Erdős-Rényi random graphs which will be useful for our analysis. The following lemma can be found in any reference on random graphs (see, e.g. Janson, Luczak, and Rucinski [JLR11]):

Lemma 16. *Let $\mathcal{G}(n, p)$ be the Erdős-Rényi graph on n vertices with connection probability p.*

- *If $np > \log n$, then with probability $1 - o(1)$, the graph is connected.*
- *If $1 < np < \log n$, then with probability $1 - o(1)$ the largest connected component has size $(1 + o(1))\beta n$, where β is the unique solution to the equation $\beta - 1 + \exp(-\beta np) = 0$. All other components have size $O(\log n)$.*

- *If $np < 1$, then with probability $1 - o(1)$ the largest connected component has size $O(\log n)$.*

In our case, each community is internally an Erdős-Rényi random graph with size $|C_i|$ and connection probability p_w. The portion of each community with is internally connected under both the SBM and the ICM is the giant connected component of an Erdős-Rényi random graph with size $|C_i|$ and connection probability $p_w q$. With a slight abuse of notation, we use the function $\beta(x)$ to refer to the size of the giant connected component induced by the SBM/ICM in a community with size x. We impose the following:

Assumption 1. *$p_w = O\left(\frac{\log n}{n}\right)$, and for all for all communities $|C_i|$, $p_w > \frac{\log |C_i|}{|C_i|}$. Intuitively, the subgraph formed by each community should be connected, but the graph is still relatively sparse. Our analysis can be extended to the dense case (e.g., $p_w = \Theta(1)$), but this is not the situation of interest for real world networks.*

For the influence process, we require

Assumption 2. *For all communities $|C_i|$, $p_w q |C_i| > 1$. This requires that the ICM and SBM jointly induce a giant connected component in each community, i.e., that an influence cascade can reach a linear portion of the community.*

We also require that all communities occupy a constant fraction of the graph:

Assumption 3. *For all communities C_i, there is some constant $c > 0$, independent of n, for which $|C_i| > cn$*

We focus on the case where p_b is sufficiently small that the communities in the graph do not themselves form a giant connected component under the ICM. While it is clearly possible to prove guarantees for the case where p_b is above this threshold (since a linear portion of the network will be connected and could be hit just by random sampling), this is not the case we are interested in modeling from an applications perspective. To formalize the threshold for p_b, we require that every community has (in expectation) less than one live edge to other communities.

Assumption 4. $p_b q \cdot (n - |C_i|)|C_i| < 1 \; \forall C_i$.

Lastly, we assume

Assumption 5. $p_b < \frac{1}{n}$.

This implies that the between-community edges by themselves do not create a giant connected component in G (which is clearly what we expect in practice).

A.1.2 Summary

ARISEN and its motivation

The idea behind ARISEN is to improve on naive sampling by estimating the size of the community that each random sample lies in, and then choose the largest communities for seeding. Each community C_i is an Erdős-Rényi graph which has average degree $d_i = |C_i|p_w + (n - |C_i|)p_b$. An estimate of d_i, combined with knowledge of p_w and p_b, yields an estimate of $|C_i|$. d_i can be estimated by simulating a series of random walk through the community to obtain a sampled set of degrees.

Having estimated the size of the community that each sampled node lies in, a natural approach would be to choose the K samples with the largest estimated size as seed nodes. However, this idea fails because there is no way to tell (using only local information) whether two sampled nodes lie in the same community: they might lie in different communities which have very similar average degree. Hence, simply choosing the samples with the largest estimated size might just seed the same community K times, which gives an approximation ratio no better than $\frac{1}{K}$ in the worst case.

The idea that we use to overcome this issue is to independently choose each sample as a seed with probability *inversely* proportional to its size. Since large communities are sampled more often, this inverse weighting “evens out" the sampling bias towards large communities and ensures that, in expectation, each of the top K communities is seeded exactly once.

Proof overview

We start by showing that the number of random samples taken in Step 1 is sufficiently large that every community will be sampled $(1 \pm \epsilon)T\frac{|C_i|}{n}$ times, and that each time it is sampled, its estimated size $\hat{S}$ will be within a multiplicative ϵ of the true value $|C_i|$. This ensures that the top K communities will be assigned total weight close to 1 in Steps (2)-(3). Each one is hit with probability approximately $1 - (1 - \frac{1}{K})^K \geq 1 - 1/e$, and a random sample within the community lands in the giant connected component induced by the ICM with probability $\beta(|C_i|)$. The major challenge is to control the effects of sampling error. While standard concentration bounds suffice to show that the estimates taken in Step 1 are accurate to within relative error ϵ, the weights are truncated in Step 3. This has the potential to amplify small errors in sampling, so the bulk of the analysis is spent in ensuring that the total utility remains close to a $\bar{\beta}(1 - 1/e)$ fraction of the top K communities after Step 3.

We then prove a bound on γ when $p_b > 0$. The intuition is that OPT can be bounded by the combined size of the largest K connected components in a subcritical Erdős-Rényi graph in which each community forms a node. However, formalizing this intuition requires a more intricate analysis.

A.1.3 Proof of main approximation result

Theorem 25. *For any $\epsilon < \frac{1}{K}$, ARISEN can be implemented using $O\left(\frac{1}{\epsilon^4}\log(n)\log^2\left(\frac{1}{\epsilon}\right)\log\log\left(\frac{1}{\epsilon}\right)\right)$ samples with approximation ratio*

$$\left(1 - \frac{1}{e} - \epsilon - o(1)\right) \cdot \bar{\beta} \cdot \gamma.$$

This is obtained by setting $T = O\left(\frac{1}{\epsilon^2}\log\frac{1}{\epsilon}\right)$ and $O\left(\frac{1}{\epsilon^2}\log(n)\log\left(\frac{1}{\epsilon}\right)\log\log\left(\frac{1}{\epsilon}\right)\right)$.

Proof.

We start out by stating a simple bound on OPT:

Lemma 17. *With probability $1 - o(1)$, $OPT \leq \frac{1}{\gamma}\sum_{i=1}^{K}\beta(|C_i|)|C_i|$.*

Proof. If all between-community edges are removed G, the set of nodes influenced by OPT

is upper bounded by the total size of the K largest connected components when each within-community edge is sampled with probability qp_w. Via Lemma 16, with probability $1 - o(1)$, each community C_i has a giant connected component of size $\beta(|C_i|)|C_i|$, with all other components having size $O(\log |C_i|)$. Via Assumption 3, all communities have size scaling as $\Theta(n)$, so for any C_i, C_j, the giant connected component in C_i is larger than the second largest connected component in C_j. Hence, the K largest connected components correspond to the giant connected components of the K largest communities. Using the definition of γ completes the proof. □

Analysis of Step 1

Step 1 nests two levels of sampling: T nodes are sampled uniformly at random from the entire graph, and then R samples are taken from the community that each of these nodes lie in. Define $\rho = \frac{|C_K|}{n}$ to be the fraction of the graph occupied by the K-th largest community. Note that $\rho = \Theta(1)$. At this point, we set $T = 96\left(\frac{1}{\epsilon^2\rho}\right)\log\frac{1}{\epsilon\rho}$. We will first show that, at the outer level, the number of times that each community is sampled is concentrated well. Then, we will show that the inner loop accurately estimates the size of each sampled community. The first claim follows from a straightforward application of the Chernoff bound. The second claim requires a more involved analysis of our random walk sampler. The following two lemmas formalize these guarantees on the output of Step 1. Their proofs are given in Section A.1.4.

Lemma 18. *Let X_j^i be the indicator variable for the event that sample j lands in community C_i. With probability at least $1 - \epsilon$,*

$$\left(1 - \frac{\epsilon}{2}\right) T\frac{|C_i|}{n} \le \sum_{j=1}^{T} X_i^j \le \left(1 + \frac{\epsilon}{2}\right) T\frac{|C_i|}{n}$$

holds for all i with $|C_i| \geq (1 - \epsilon)|C_K|$.

Lemma 19. *There are settings $R = O\left(\frac{1}{\epsilon^2}\log n \log T\right)$ and $B = O(\log n \log \frac{1}{\epsilon})$ such that, across all T iterations, given R random walk samples each sampled community C, the estimated size $\hat{d}$ satisfies $\left(1 - \frac{\epsilon}{4}\right) p_w|C| \le \hat{d} \le \left(1 + \frac{\epsilon}{4}\right) p_w|C|$ with probability $1 - o(1)$*

Corollary 1. *With probability at least* $1 - o(1)$, $\left(1 - \frac{\epsilon}{2}\right)|C_i| \leq \hat{S}_i \leq \left(1 + \frac{\epsilon}{2}\right)|C_i|$ *holds for every* $i = 1...T$.

Proof. We estimate the size as $\hat{S} = \frac{\hat{d} - np_b}{p_w - p_b}$. We now show the upper bound on $\hat{S}$; the argument for the lower bound is exactly the same.

$$\begin{aligned}
\hat{S} &\leq \frac{(1 + \frac{\epsilon}{4})(p_w|C_i| + (n - |C_i|)p_b) - np_b}{p_w - p_b} \\
&\leq \frac{(1 + \frac{\epsilon}{4}\epsilon)(p_w|C_i| + (n - |C_i|)p_b - np_b) + \frac{1}{4}\epsilon np_b}{p_w - p_b} \\
&= \left(1 + \frac{\epsilon}{4}\epsilon\right)|C_i| + \frac{\frac{1}{4}\epsilon np_b}{p_w - p_b}.
\end{aligned}$$

So, we just need to bound the size of $\frac{\frac{1}{4}\epsilon np_b}{p_w - p_b}$ relative to $|C|$. We know by Assumption 5 that $\frac{\epsilon}{4}\epsilon np_b < \frac{\epsilon}{4}$. Further, using Assumption 1,

$$\begin{aligned}
p_w - p_b &> \frac{\log|C_i|}{|C_i|} - \frac{1}{n} \\
&\geq \frac{\log|C_i|}{|C_i|} - \frac{1}{|C_i|} \\
&\geq \frac{1}{|C_i|}
\end{aligned}$$

from which we conclude that $\frac{\frac{1}{4}\epsilon np_b}{p_w - p_b} \leq \frac{1}{4}\epsilon|C_i|$. Thus, $\hat{S} \leq (1 + \frac{\epsilon}{2})|C_i|$.

□

We emphasize that Lemma 19 applies to *all* communities that are sampled, not just those which have size at least $(1 - \epsilon)|C_K|$. However, Lemma 18 only applies to communities with size at least $(1 - \epsilon)|C_K|$. That is, each sampled community's size estimate is accurate, but small communities may not be reliably sampled. Note that the total query cost is $R \cdot T$, which implies the bound in the theorem statement after noting that ρ is constant with respect to ϵ and n.

Analysis of Step 2

We now analyze the probability that each community in the top K is seeded based on the above estimates. Consider any community $C_i \in \{C_1 ... C_K\}$. C_i is seeded if any of the sampled nodes from it are chosen in Step 4, and the probability of this event is determined by the total amount of weight which is allocated to nodes in C_i. In this step, we show that the total weight assigned to each of the top K communities is close to 1. Formally,

Lemma 20. *For any community C_i, let $w(C_i)$ be the total weight assigned to C_i. Suppose that C_i satisfies*

- $\left(1 - \frac{\epsilon}{2}\right) T \frac{|C_i|}{n} \leq \sum_{j=1}^{T} X_i^j \leq \left(1 + \frac{\epsilon}{2}\right) T \frac{|C_i|}{n}$
- $\left(1 - \frac{\epsilon}{2}\right) |C_i| \leq \hat{S}_i \leq \left(1 + \frac{\epsilon}{2}\right) |C_i|$ *each time C_i is sampled.*

Then, $1 - \epsilon \leq w(C_i) \leq 1 + \epsilon$.

Proof. We have

$$\begin{aligned}
w(C_i) &= \sum_{j=1}^{T} \mathbf{1}\{j \in C_i\} w_j \\
&= \sum_{j=1}^{T} \mathbf{1}\{j \in C_i\} \frac{n}{\hat{S}_j T} \\
&\geq \left(1 - \frac{\epsilon}{2}\right) T \frac{|C_i|}{n} \frac{n}{\left(1 + \frac{\epsilon}{2}\right) |C_i| T} \\
&= \frac{1 - \frac{\epsilon}{2}}{1 + \frac{\epsilon}{2}} \\
&\geq 1 - \epsilon
\end{aligned}$$

A similar argument shows that $w(C_i) \leq (1 + \epsilon)$ also holds. □

Corollary 2. *With probability at least $1 - \epsilon - o(1)$, $1 - \epsilon \leq w(C_i) \leq 1 + \epsilon$ holds for every community C_i sampled during Step 1 with $\hat{S}_i > 0$.*

Proof. Via Lemma 18 and Corollary 1 (and union bound), we can apply Lemma 20 to each community sampled in Step 1 with total probability at least $1 - \epsilon$. □

Analysis of Steps 3 and 4

Now, we need to analyze the impact of the truncation in Step 3. If the size of every community were perfectly estimated, then this step would set the weight of each community with size less than C_K to zero, leaving only $C_1 ... C_K$ to be seeded in Step 4. The following analysis controls the loss that can be suffered due to sampling errors.

For instance, it is possible that C_K could have "borderline" size arbitrarily close to $|C_{K+1}|$, in which case much of this weight may be truncated in favor of samples from C_{K+1}. For this to occur, a sample from C_{K+1} must have estimated size higher than a sample from C_K. But since the size of each sampled community is well-estimated, this implies that $|C_{K+1}|$ is actually very close to $|C_K|$, so not much is lost.

We now formalize this intuition. Recall that Step 3 calculates a threshold τ: the algorithm keeps all samples j where $\hat{S}_j \geq \tau$ and discards those with $\hat{S}_j < \tau$ by setting $w_j = 0$. Let $w(C_i)$ denote the total weight of community C_i before truncation and $w_T(C_i)$ denote its total weight after truncation. We define four sets of communities

- *Small* $= \{C_i | \ |C_i| < \frac{1-\frac{\epsilon}{2}}{1+\frac{\epsilon}{2}}|C_K|\}$. These are communities we would like to show never displace samples from communities in the other three sets.
- $A = \{C_i | \ \frac{1-\frac{\epsilon}{2}}{1+\frac{\epsilon}{2}}|C_K| \leq |C_i| < |C_K|\}$. These are communities with size less than $|C_K|$, but which we might not be able to detect and truncate due to sampling errors.
- $B = \{C_i | \ |C_K| \leq |C_i| \leq \frac{1+\frac{\epsilon}{2}}{1-\frac{\epsilon}{2}}|C_K|\}$. These are communities with size at least $|C_K|$, but which are small enough that they might be confused with communities in A.
- *Large* $= \{C_i | \ |C_i| > \frac{1+\frac{\epsilon}{2}}{1-\frac{\epsilon}{2}}|C_K|\}$. These are communities we would like to show are never truncated.

First, we show that communities in *Small* and *Large* behave well under truncation, in the sense that no samples from *Large* are truncated and no samples from *Small* displace samples from $B \cup$ *Large*. In what follows we condition on the events in Corollaries 1 and 2.

Lemma 21. *w_T satisfies the following conditions:*

- $w_T(Small) \leq K - w_T(B) - w_T(Large)$

- $w_T(Large) = w(Large) \leq |Large| + \epsilon|Large|$

Proof. If a community C_i is in *Small*, then the size estimated for each of its samples satisfies

$$\hat{S} \leq \left(1 + \frac{\epsilon}{2}\right)|C_i| < \left(1 + \frac{\epsilon}{2}\right)\left(\frac{1 - \frac{\epsilon}{2}}{1 + \frac{\epsilon}{2}}\right)|C_K| = \left(1 - \frac{\epsilon}{2}\right)|C_K|$$

Hence, samples from communities in *Small* will always have estimated size less than every sample from $B \cup Large$, from which the first claim follows. The same logic shows that every sample from communities in *Large* has estimated size higher than every sample from C_K. This implies (via $\sum_{i=1}^{K-1} w(C_i) \leq (K-1) + \epsilon(K-1)$) that each sample's estimated size lies above τ, proving the second claim. □

We recall here that choosing a random sample from a community C_i has probability $\beta(|C_i|)$ of hitting the giant connected component induced by the ICM, in which case it influences a fraction $\beta(|C_i|)$ of the nodes in the community. Hence, the total expected utility is

$$\sum_{C_i} \beta(|C_i|)^2 \Pr[C_i \text{ is seeded}]|C_i| \ \geq \sum_{C_i \in A \cup B \cup Large} \beta(|C_i|)^2 \Pr[C_i \text{ is seeded}]|C_i|$$

We refer to the value of the above summation restricted to a particular set of communities as the total utility obtained from that set. We now proceed to bound the total utility obtained from $A \cup B$, and then the total utility obtained from *Large*.

Lemma 22. *The total utility obtained from $A \cup B$ is at least*

$$(|B| - 4\epsilon K)\left(1 - \frac{1}{e}\right) \cdot \beta((1-\epsilon)|C_K|)^2(1-\epsilon)|C_K|.$$

Proof. Via Lemma 21, at least $K - |Large| - \epsilon|Large| = |B| - \epsilon|Large|$ weight must be allocated to communities in A and B. Hence, the total expected utility obtained from these communities is

$$\begin{aligned}\sum_{C_i \in A \cup B} \beta(|C_i|)^2 \Pr[C_i \text{ is seeded}]|C_i| &= \sum_{C_i \in A \cup B} \beta(|C_i|)^2 \left(1 - \left(1 - \frac{w(C_i)}{K}\right)^K\right)|C_i| \\ &\geq \sum_{C_i \in A \cup B} \beta(|C_i|)^2 \left(1 - e^{-w(C_i)}\right)|C_i| \\ &\geq \beta\left((1-\epsilon)|C_K|\right)^2 (1-\epsilon)\,|C_K| \sum_{C_i \in A \cup B} 1 - e^{-w(C_i)}. \quad \text{(A.1)}\end{aligned}$$

Given the above constraints on the total amount of weight allocated to each community, the value of Equation (A.1) is at least the value of the following optimization problem:

$$\min \beta\left((1-\epsilon)|C_K|\right)^2 (1-\epsilon)\,|C_K| \sum_{C_i \in A \cup B} 1 - e^{-w(C_i)}$$

$$w(C_i) \leq 1 + \epsilon \quad \forall C_i \in A \cup B$$

$$\sum_{C_i \in A \cup B} w(C_i) \geq |B| - \epsilon|Large|$$

Here the first constraint is due to Corollary 2, and the second is due to the argument at the start of this lemma. Let Q be the optimal value of the above optimization problem. The objective is the sum of identical concave functions in each variable $w(C_i)$. Hence, the minimum is achieved when as many of the $w(C_i)$ as possible are set to $1+\epsilon$, with one community receiving the leftover weight. Specifically, $\left\lfloor \frac{|B|-\epsilon|Large|}{1+\epsilon} \right\rfloor$ communities receive weight $1+\epsilon$. Since $\epsilon < \frac{1}{K}$, $\left\lfloor \frac{|B|-\epsilon|Large|}{1+\epsilon} \right\rfloor = |B| - 1$. Hence, the remaining community receives weight

$$|B| - \epsilon|Large| - (1+\epsilon)(|B|-1) \geq 1 - \epsilon|Large| - \epsilon|B| = 1 - \epsilon K$$

Hence, we can lower bound Q as

$$\begin{aligned} Q &\geq \left((|B|-1)\left(1 - e^{-(1+\epsilon)}\right) + \left(1 - e^{-(1-\epsilon K)}\right)\right) \cdot \beta((1-\epsilon)|C_K|)^2(1-\epsilon)|C_K| \\ &\geq \left(|B|\left(1 - e^{-(1+\epsilon)}\right) - \left(e^{\epsilon K} - e^{\epsilon}\right)\right) \cdot \beta((1-\epsilon)|C_K|)^2(1-\epsilon)|C_K| \\ &\geq \left(|B|\left(1 - e^{-(1+\epsilon)}\right) - \left(\frac{1}{1-\epsilon K} - e^{\epsilon}\right)\right) \cdot \beta((1-\epsilon)|C_K|)^2(1-\epsilon)|C_K| \end{aligned}$$

$$\begin{aligned}
&\geq \left(|B|\left(1-e^{-(1+\epsilon)}\right)-(1+2\epsilon K-e^{\epsilon})\right)\cdot\beta((1-\epsilon)|C_K|)^2(1-\epsilon)|C_K| \\
&\geq \left(|B|\left(1-e^{-(1+\epsilon)}\right)-2\epsilon K\right)\cdot\beta((1-\epsilon)|C_K|)^2(1-\epsilon)|C_K| \\
&\geq (|B|-4\epsilon K)\left(1-e^{-(1+\epsilon)}\right)\cdot\beta((1-\epsilon)|C_K|)^2(1-\epsilon)|C_K| \\
&\geq (|B|-4\epsilon K)\left(1-\frac{1}{e}\right)\cdot\beta((1-\epsilon)|C_K|)^2(1-\epsilon)|C_K|
\end{aligned}$$

□

Lemma 23. *The total utility obtained from Large is at least* $\sum_{C_i\in Large}\beta(|C_i|)^2|C_i|\left(1-\frac{1}{e}-\epsilon\right)$.

Proof. Follows directly from Lemma 21, which implies that every $C_i \in Large$ satisfies $w(C_i) \geq 1-\epsilon$. As a result, $\Pr[C_i \text{ is seeded}] \geq 1-e^{-(1-\epsilon)}$. Then, we have

$$\begin{aligned}
1-\frac{1}{e^{1-\epsilon}} &\geq 1-\frac{1}{1-\epsilon}\frac{1}{e} \\
&\geq 1-(1+2\epsilon)\frac{1}{e} \\
&\geq 1-\frac{1}{e}-\epsilon.
\end{aligned}$$

□

After some more algebra, this leads to our final bound on the total utility:

$$\begin{aligned}
&\sum_{C_i\in A\cup B\cup Large}\beta(|C_i|)^2\Pr[C_i \text{ is seeded}]|C_i| \\
&\geq |B|\left(1-\frac{1}{e}\right)\cdot\beta((1-\epsilon)|C_K|)^2(1-\epsilon)|C_K|+\sum_{C_i\in Large}\beta(|C_i|)^2|C_i|\left(1-\frac{1}{e}-\epsilon\right)- \\
&\qquad\qquad 4\epsilon K\beta((1-\epsilon)|C_K|)^2|C_K|\left(1-\frac{1}{e}\right) \\
&\geq \sum_{i=1}^{K}\left(1-\frac{1}{e}-\epsilon\right)(1-4\epsilon)\beta((1-3\epsilon)|C_i|)^2(1-3\epsilon)|C_i|
\end{aligned}$$

Now, we consider the term $\beta((1-3\epsilon)|C_i|)$. We abuse notation now and write β as a function of the average degree $\bar{d}$ in the influence graph It is known that

$$\beta(\bar{d}) = 1+\frac{1}{\bar{d}}W(-\bar{d}e^{-\bar{d}})+o(1)$$

where W is the Lambert W function. Taking derivatives yields

$$\frac{d}{d\bar{d}}\beta(\bar{d}) = -\frac{1}{\bar{d}^2}W(-\bar{d}e^{-\bar{d}}) + \frac{1}{x} \cdot \frac{1}{e^{W(-\bar{d}e^{-\bar{d}})} - \bar{d}e^{-\bar{d}}}$$

which over $\bar{d} > 1$ (as guaranteed by Assumption 2) satisfies $|\frac{d}{d\bar{d}}\beta(\bar{d})| \leq 1$. Accordingly, we have that

$$\beta((1-3\epsilon)|C_i|) \geq \beta(|C_i|) - 3\epsilon - o(1) = \left(1 - \frac{3\epsilon}{\beta(|C_i|) - o(1)}\right)\beta(|C_i|).$$

Continuing to bound the total influence spread, we have

$$\begin{aligned}
&\sum_{i=1}^{K}\left(1 - \frac{1}{e} - \epsilon\right)(1-4\epsilon)\beta((1-3\epsilon)|C_i|)^2(1-3\epsilon)|C_i| \\
&\geq \left(1 - \frac{1}{e} - \epsilon\right)(1-4\epsilon)\left(1 - \frac{3\epsilon}{\min_{i=1...K}\beta(|C_i|)} - o(1)\right)^2 \sum_{i=1}^{K}\beta(|C_i|)^2|C_i| \\
&\geq \left(1 - \frac{1}{e} - \frac{11\epsilon}{\min_{i=1...K}\beta(|C_i|)} - o(1)\right)\sum_{i=1}^{K}\beta(|C_i|)^2|C_i|
\end{aligned}$$

By Lemma 17, $OPT \leq \frac{1}{\gamma}\sum_{i=1}^{K}\beta(C_i)|C_i|$. Accordingly, we would like to bound the term

$$\frac{\sum_{i=1}^{K}\beta(|C_i|)^2|C_i|}{\sum_{i=1}^{K}\beta(|C_i|)|C_i|}.$$

Define $\bar{\beta} = \frac{1}{K}\sum_{i=1}^{K}\beta(|C_i|)$ to be the average size of the giant component induced by the influence process in the largest K communities. Since $|C_1| \geq |C_2| \geq .... \geq |C_K|$, we have that

$$\sum_{i=1}^{K}\beta(|C_i|)^2|C_i| \geq \bar{\beta}\sum_{i=1}^{K}\beta(|C_i|)|C_i|.$$

To conclude, we recall that in earlier steps we conditioned on events in the sampling procedure and the randomness of the graph which had combined probability $1 - \epsilon - o(1)$. Adding this up gives

$$\begin{aligned}
\mathbb{E}[f_E(\mathcal{A}(E))] &\geq (1 - \epsilon - o(1))\left(1 - \frac{1}{e} - \frac{11\epsilon}{\min_{i=1...K}\beta(|C_i|)} - o(1)\right)\bar{\beta}\cdot\gamma\cdot OPT \\
&\geq \left(1 - \frac{1}{e} - \frac{12\epsilon}{\min_{i=1...K}\beta(|C_i|)} - o(1)\right)\bar{\beta}\cdot\gamma\cdot OPT
\end{aligned}$$

and so now running the algorithm with $\epsilon' = \frac{1}{12\min_{i=1...K}\beta(|C_i|)}\epsilon = \Theta(\epsilon)$ suffices to obtain the desired approximation guarantee.

A.1.4 Concentration lemmas

We now prove that the various estimates that the algorithm takes in Step 1 are sufficiently accurate. We make frequent use of the Chernoff bound:

Lemma 24 ([MR10])**.** *Let $X_1...X_N$ be independent binary random variables. Let $X = \sum_{i=1}^{N} X_i$ and $\mu = \mathbb{E}[X]$.*

- *For* $0 < \delta < 1$, $\Pr[|X - \mu| \geq \delta\mu] \leq 2e^{-\frac{\delta^2\mu}{3}}$.
- *For* $\delta > 1$, $\Pr[X \geq (1+\delta)\mu] \leq e^{-\frac{\delta\mu}{3}}$

We now proceed to prove Lemmas 18 and 19.

Proof of Lemma 18. Note that $\mathbb{E}[\sum_{j=1}^{T} X_j^i] = T\frac{|C_i|}{n}$ Via the Chernoff bound, we have that

$$\begin{aligned}
\Pr\left[\left|\sum_{j=1}^{T} X_j^i - T\frac{|C_i|}{n}\right| > \frac{\epsilon}{2}T\frac{|C_i|}{n}\right] &\leq 2\exp\left(-\frac{1}{12}\epsilon^2 T\frac{|C_i|}{n}\right) \\
&\leq 2\exp\left(-\frac{1}{24}\epsilon^2 T(1-\epsilon)\rho\right) \qquad (|C_i| \geq (1-\epsilon)\rho n) \\
&\leq 2\exp\left(-2\log\frac{1}{\epsilon\rho}\right) \\
&\leq 2(\rho\epsilon)^2.
\end{aligned}$$

There are at most $\frac{1}{(1-\epsilon)\rho}$ communities of size at least $(1-\epsilon)\rho n$. By union bound, concentration holds for each of them with probability at least $1 - 2\epsilon\rho \geq 1 - \epsilon$. □

Proof of Lemma 19. We recall the lemma statement: There are settings $R = O\left(\frac{1}{\epsilon^2}\log n \log T\right)$ and $B = O(\log n \log \frac{1}{\epsilon})$ such that, given R random walk samples from each community C, the estimated size $\hat{S}$ satisfies $(1-\epsilon)|C| \leq \hat{S} \leq (1+\epsilon)|C|$ with probability at least $1 - o(1)$ across all T sampled communities.

Recall that our algorithm runs a random walk on the graph, simulating the presence of self-loops by adding a random number of extra copies of each node visited. The number of self-loops is chosen so that every node has an equal degree Δ (a parameter to be set later). By ensuring that every node as an equal degree, the stationary distribution of the random walk becomes uniform over the vertices. We bound the number of steps in the random walk that are needed including the steps which follow self-loops; clearly, the number of actual queries to the graph can only be smaller.

We start by analyzing the degrees of nodes in the graph. As is well-known for Erdős-Rényi graphs, degrees become tightly concentrated in the regime where the expected degree is $\Theta(\log n)$. Here we show that, with high probability, every node in community C_i has degree $\Theta(\log |C_i|)$. We condition on this event holding in the remainder of the proof.

Lemma 25. *With probability $1 - o(1)$, every node in community C_i has degree $\Theta(\log |C_i|)$.*

Proof. We first prove that for each node $v \in C_i$, $d_v = O(\log |C_i|)$. Note that d_v is the sum of Bernoulli independent random variables, with expectation $\mu = \Theta(\log |C_i|)$. Using the Chernoff bound,

$$Pr\left[d_v > (1+\delta)\mu\right] \leq e^{-\frac{1}{4}\mu\delta^2}$$

and so by taking $\delta = \Theta(1)$, we obtain that $d_v = O(\log |C_i|)$ with failure probability at most $\frac{1}{|C_i|^2}$. Taking union bound over all $|C_i|$ nodes gives combined failure probability at most $\frac{1}{|C_i|} = o(1)$.

For the other direction, we use a sharper version of the Chernoff bound:

$$Pr\left[d_v < (1-\delta)\mu\right] \leq \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\mu}.$$

We can rewrite the failure probability as

$$\exp\left(-\mu\left(\delta + (1-\delta)\log(1-\delta)\right)\right).$$

Now, note that by Assumption 1, $\mu > \log |C_i|$, i.e., there is some constant $c > 1$ independent of n for which $\mu \geq c \log |C_i|$. Now, since as $\delta \to 1$, $\delta + (1-\delta)\log(1-\delta) \to 1$, there is some value of δ, which depends only on c, such that $\delta + (1-\delta)\log(1-\delta) > \frac{1}{c}$, where we require that the inequality hold strictly. Note that since c does not depend on n, we still have $\delta = \Theta(1)$ at this point, ensuring that $(1-\delta)\mu = \Omega(\log |C_i|)$. Now the failure probability is at most $\frac{1}{|C_i|^{1+\Theta(1)}}$ and so after taking union bound over $|C_i|$ nodes, the combined failure probability is $o(1)$. □

We now address two topics. First, we show that each random walk stays within its starting community with high probability. Second, we show that the average degree in each community is estimated accurately.

Random walks do not leave their starting community: Our analysis uses the connection between the conductance of a graph and the properties of random walks on it, so we start by introducing a few definitions. The *volume* of a set of nodes S, denoted by $\mu(S)$ is the sum of the degrees of the nodes in S:

$$\mu(S) = \sum_{i \in S} d_i$$

Let $E(S, S-V)$ denote the set of edges between nodes in S and those in $V-S$. They key ratio for our analysis is

$$\Phi(S) = \frac{|E(S, V-S)|}{\mu(S)}.$$

This is nearly the same as the normal definition of conductance, which has $\min(\mu(S), \mu(V-S))$ in the denominator. However, our analysis depends only on $\mu(S)$. The key lemma that we use relates $\Phi(S)$ to the properties of a random walk in S:

Lemma 26 (Spielman and Teng [ST13], Prop. 2.5)**.** *The probability that a random walk, started from a random node of S, stays entirely within S for t steps is at least $1 - \frac{1}{2}t\Phi(S)$.*

We remark that Spielman and Teng stated the lemma for the normal conductance (not our Φ), but their analysis trivially applies to Φ as we have defined it.

Lemma 26 will be used to control the probability that any of the nodes we sample in Step 1c lie outside of the starting community. Fix any C_i. We apply the Chernoff bound to the numerator and denominator of $\Phi(C_i)$ to show that it is close to $\frac{\log n}{n}$ with high probability over the draw of G from the SBM.

First, we show that with high probability, $|E(C_i, V - C_i)| \leq \frac{7 \log n}{q}$. Let $Z = |E(C_i, V - C_i)|$ and note that Z is the sum of $(n - |C_i|)|C_i|$ indicator variables giving whether each possible between-community edge is present. From Assumption 4, we know that $\mathbb{E}[Z] < \frac{1}{q}$. Thus via the Chernoff bound we have

$$\begin{aligned}\Pr[Z > (1 + 6\log n)\, \mathbb{E}[Z]] &\leq \exp\left(-\frac{1}{3}\left(\frac{6\log n}{q}\right)\right) \\ &\leq \frac{1}{n^2}\end{aligned}$$

There are at most n total communities, so taking union bound over all of them gives total failure probability at most $\frac{1}{n}$. Conditioned on concentration holding, we have $Z \leq (6\log n + 1)\, \mathbb{E}[Z] \leq \frac{7\log n}{q}$.

Next, we examine $\mu(C_i)$. We have $\mathbb{E}[\mu(C_i)] = p_w|C_i|^2$. By assumption, $p_w|C_i|^2 \geq |C_i|$. Via Chernoff bound,

$$\begin{aligned}\Pr\left[\mu(C_i) \leq \left(1 - \sqrt{\frac{6\log n}{|C_i|}}\right) \mathbb{E}[\mu(C_i)]\right] &\leq \exp\left(-\frac{1}{3}\left(\frac{6\log n}{|C_i|}\right)|C_i|\right) \\ &\leq \frac{1}{n^2}.\end{aligned}$$

Again via union bound, the total failure probability over all communities is at most $\frac{1}{n}$.

Conditioning on the bounds on both the numerator and denominator holding, we have

$$\Phi(C_i) \leq \frac{7\log n}{\left(1 - \sqrt{\frac{6\log n}{|C_i|}}\right) q p_w |C_i|^2}$$

Since $qp_w|C_i| \geq 1$ by Assumption 2, this implies

$$\begin{aligned}\Phi(C_i) &\leq \frac{7\log n}{\left(1-\sqrt{\frac{6\log n}{|C_i|}}\right)|C_i|} \\ &= \frac{7\log n}{|C_i| - \sqrt{6|C_i|\log n}}\end{aligned}$$

Now we can apply Lemma 26 to bound the probability that the random walk leaves C_i. In any single iteration, we take R random walk steps, which leave C_i with probability at most $\frac{1}{2}R\Phi(C_i)$. There are T iterations in total, so via union bound the total probability that any random walk leaves its starting community is at most $\frac{1}{2}\Phi(C_{min})RT$ where C_{min} is the smallest community. This yields

$$\begin{aligned}\frac{1}{2}\Phi(C_{min})RT &= O\left(\left(\frac{1}{\epsilon^4\rho}\right)\log^2\frac{1}{\epsilon\rho}\log\log\frac{1}{\epsilon\rho}\log n\right)\frac{\frac{7}{2}\log n}{|C_{min}| - \sqrt{6|C_{min}|\log n}} \\ &= o(1) \qquad (\text{since } |C_{min}| = \Theta(n))\end{aligned}$$

We conclude that the total probability of any random walk leaving its starting community is at most $o(1)$.

Estimating the average degree: We now introduce some notation dealing with Markov chains. Suppose a Markov chain has transition matrix $\boldsymbol{P}$. All Markov chains we consider will have a unique stationary distribution. Let this distribution be π. The total variational distance between probability distributions p and q is

$$d_{TV}(p,q) = \sup_x |p(x) - q(x)|.$$

The mixing time of a chain is the maximum time needed for the chain to converge to its stationary distribution:

$$t_{mix}(\epsilon) = \min\{t \mid \sup_x d_{TV}(1_x \boldsymbol{P}^t, \pi) \leq \epsilon\}$$
$$t_{mix} \coloneqq t_{mix}\left(\frac{1}{4}\right)$$

The spectral gap of the chain is $1 - \lambda_2$, where λ_2 is the second eigenvalue of $\boldsymbol{P}$. Our analysis will use the well-known connection between the mixing time of a random walk, the spectral gap, and the conductance. Now, by [HKP12] (Theorem 1.1), we have that since C_i is an Erdős-Rényi graph of sufficiently large average degree, $1 - \lambda_2 = \Omega\left(1 - \frac{1}{\sqrt{\log |C_i|}}\right)$ with probability $1 - o(1)$. Let Φ denote the conductance of C_i. Via the Cheeger inequality, we have that $\Phi \geq \frac{1}{2}(1 - \lambda_2) = \Omega\left(1 - \frac{1}{\sqrt{\log |C_i|}}\right)$. These facts ensure that random walks mix quickly on C_i.

However, we need to analyze the behavior of the modified random walk where we add self loops to ensure that every node has Δ edges. We will choose $\Delta = \Theta(\log n)$ to be an upper bound on the largest degree on C_i per Lemma 25. Call the conductance of this modified random walk $\tilde{\Phi}$ and its spectral gap $1 - \tilde{\lambda}_2$. We use the relationship between Φ and $\tilde{\Phi}$ outlined by [DKS14]. Specifically, let $e(S, \bar{S})$ denote the number of edges between a set of nodes $S \subseteq C_i$ and $C_i \setminus S$. Let $d(S) = \sum_{v \in S} d_v$. For any such subset S, the conductance on that subset is

$$\Phi(S) = \frac{e(S, \bar{S})}{d(S)}$$

while for the modified graph we have

$$\tilde{\Phi}(S) \geq \frac{e(S, \bar{S})}{d(S) + \Delta |S|} \geq \frac{e(S, \bar{S})}{d(S)\left(1 + \frac{\Delta}{\min_{v \in C_i} d_v}\right)}$$

since we add at most Δ self-loops to each node. But now, since $\Delta = \Theta(\log n)$, and by Lemma 25 combined with Assumption 3, $\min_{v \in C_i} d_v = \Theta(\log |C_i|) = \Theta(\log n)$, we have that $\tilde{\Phi}(S) = \Theta(\Phi(S))$ and so taking the minimum over all subsets S,

$$\tilde{\Phi} = \Theta(\Phi) = \Omega\left(1 - \frac{1}{\sqrt{\log |C_i|}}\right).$$

Now, using the Cheeger inequality, we have that

$$1 - \tilde{\lambda}_2 \geq \frac{1}{2}\tilde{\Phi}^2 = \Omega\left(1 - \frac{1}{\sqrt{\log |C_i|}}\right).$$

Using the well-known connection between the spectral gap and mixing time (see Theorem 12.3 of [LP17]), the mixing time of the modified walk is $O\left(\frac{\log n}{1-\frac{1}{\sqrt{\log n}}}\right)$ and after a burn-in time of $O(\log n \log \frac{1}{\epsilon})$ steps, the distribution of the walk is within total variational distance ϵ of the stationary distribution (which is uniform since all nodes have degree Δ).

We will apply concentration bounds to the the new random walk. We use the following Bernstein-style concentration bound for the sum of a function of a Markov chain.

Lemma 27 (Paulin [P$^+$15]) Theorem 3.3)**.** *Let $X_1 ... X_r$ be a stationary reversible Markov chain over state space Ω with spectral gap γ and stationary distribution π. Let $g \in L^2(\pi)$, with $|g(x) - \mathbb{E}_\pi[g]| \leq C$ for every $x \in \Omega$. For every $\delta > 0$ we have*

$$\Pr_\pi\left[\left|\sum_{i=1}^{r} g(X_i) - \mathop{\mathbb{E}}_{\pi}\left[\sum_{i=1}^{r} g(X_i)\right]\right| \geq \delta\right] \leq 2\exp\left(\frac{-\delta^2(1-\lambda_2)}{4rVar_\pi(g) + 10\delta C}\right)$$

To account for the fact that the chain does not start at stationarity, we can use a burn in time of t_0 steps, which gives the following bound:

Lemma 28 (Paulin [P$^+$15]) Proposition 3.10)**.** *Suppose that the chain starts from distribution q and we discard the first t_0 samples. Let $\boldsymbol{P}$ be the transition matrix. Then*

$$\begin{aligned}\Pr_q&\left[\left|\sum_{i=t_0+1}^{r} g(X_i) - \mathop{\mathbb{E}}_{\pi}\left[\sum_{i=t_0+1}^{r} g(X_i)\right]\right| \geq \epsilon\right] \\ &\leq \Pr_\pi\left[\left|\sum_{i=t_0+1}^{r} g(X_i) - \mathop{\mathbb{E}}_{\pi}\left[\sum_{i=t_0+1}^{r} g(X_i)\right]\right| \geq \epsilon\right] + d_{TV}(q\boldsymbol{P}^{t_0}, \pi).\end{aligned}$$

We will apply this lemma to the random walk Markov chain where the states are the nodes of the graph and $g(v) = d_v$. Since the stationary distribution of the modified random walk is uniform, $\mathbb{E}_\pi[g] = d_{avg}$. We will use $\delta = \frac{1}{2} r\epsilon d_{avg}$ because this suffices to ensure that

$$(1-\epsilon)d_{avg} \leq \frac{1}{r}\sum_{i=t_0+1}^{r} d_i \leq (1+\epsilon)d_{avg}.$$

In order to apply the lemma to bound the failure probability for the estimate, we just have to bound the variance and maximum deviation of g. Note that after conditioning on the event in Lemma 25, $O(\log|C_i|)$ is a bound on the largest degree in the graph, and hence on the variance of the degrees d_v as well. Since the mean d_{avg} is also $\Theta(\log|C_i|)$, by taking $R = \Theta\left(\frac{1}{\epsilon^2}\log n \log T\right)$, we can ensure that the failure probability is at most $\frac{1}{nT}$. After taking union bound over all T sampling iterations, the combined failure probability is at most $\frac{1}{n} = o(1)$. Since the burn-in time is $O(\log n \log \frac{1}{\epsilon})$, the total number of steps is still $\Theta\left(\frac{1}{\epsilon^2}\log n \log T\right)$. This concludes the proof of Lemma 19. □

This completes the proof of Theorem 1.

A.1.5 Bounding between-community influence

We prove the following guarantee on the relative sizes of $\sum_{i=1}^K |C_i|$ and OPT in the $p_b > 0$ setting:

Lemma 29. *Let $\mu = \frac{1}{K}\sum_{i=1}^K \beta(|C_i|)|C_i|$ denote the average size of the giant components of the top K communities. Then we have*

$$\sum_{i=1}^{K} \beta(|C_i|)|C_i| \;\geq\; \left(\frac{1-c_{max}}{12\log\frac{n}{\mu}}\right) OPT$$

Proof. Let $X_1 ... X_K$ be the sizes of the K largest connected components induced by the SBM and ICM. We have $OPT \leq \mathbb{E}\left[\sum_{i=1}^K X_i\right]$. Each X_i contains the giant connected component in one or more communities. Let C_i^* be the (random) community which is the largest community whose giant component is contained in X_i. Let $\boldsymbol{C}^*$ be a vector which collects $|C_1^*| ... |C_K^*|$. Clearly, we have $\sum_{i=1}^K |C_i| \geq \mathbb{E}\left[\sum_{i=1}^K |C_i^*|\right]$. We will now bound the amount by which $\mathbb{E}\left[\sum_{i=1}^K X_i\right]$ can exceed $\mathbb{E}\left[\sum_{i=1}^K |C_i^*|\right]$, which in turn lets us bound $\sum_{i=1}^K |C_i|$ in terms of OPT.

The crucial step is to bound a single X_i relative to $|C_i^*|$. We show

Lemma 30. $\mathbb{E}[X_i|\boldsymbol{C^*}] \leq \left(\frac{12}{1-c_{max}}\right) \log\left(\frac{n}{|C_i^*|}\right) \beta(|C_i^*|)|C_i^*|$

Proof. We analyze a branching process, similar to that used to analyze the subcritical Erdős-Rényi graph. This process starts at a single node, and then reveals the status of all potential edges to the remaining nodes. Each edge that exists creates a new child and the process then explores the edges of each child. The size of the connected component is the total number of nodes explored by the branching process.

Our analysis will collapse the giant connected component of each community into a single node in a higher-level branching process. This allows us to bound the total number of nodes of G that can be absorbed into a connected component. The major challenge for us to analyze the branching process is that the communities need not have equal sizes, so we cannot apply the analysis of the Erdős-Rényi graph exactly. We prove that the number of nodes reached in the true branching process is stochastically dominated by one in which every community in the graph has size $|C_i^*|$.

Conditioning on $\boldsymbol{C^*}$ (as in the lemma statement) complicates the branching process because if a given community is reached, then it has a chance to reach a community with size above $|C_i^*|$, or to reach a community in one of the other components. Hence, conditioning on $\boldsymbol{C^*}$ reduces the probability that the branching process will reach any of the other communities in the graph. However, the true process is stochastically dominated by a branching process on the subgraph induced by the communities with size at most $|C_i^*|$; call this graph G_A. Essentially, in this process we ignore that conditioning on $|C_i^*|$ can indirectly limit the number of nodes reached, and that the other components could "compete" with X_i for nodes. To formalize this reasoning, we define two branching processes:

BP-cond: This is the "true" branching process. Pick a node in C_i^* to start from. From the starting node, reveal the status (live or not) of all edges from this node's community to other communities. These revelations follow a distribution which conditions on (1) not reaching a community with size greater than C_i^* or (2) reaching a community which belongs to the other $K-1$ components. Note that the BP-cond's corresponding to each of the K largest components could have a complicated joint distribution but we do not need to fully describe it (as will be seen below).

BP-A: Pick a node in the largest community of G_A. Follow the branching process from that node using only edges between nodes in G_A (but ignoring the two conditions for BP-cond)

Let Z_{cond} (resp Z_A) be a random variable giving the total number of nodes in communities reached by BP-cond (resp. BP-A). We have

Claim 1. *Z_{cond} is stochastically dominated by Z_A*

Proof. Let Y_e $\forall e \in V \times V$ be an indicator variable for the event that edge e is live (i.e., it is present in both the SBM and ICM). $\boldsymbol{Y}$ is a vector which collects all of the Y_e. Let $h(\boldsymbol{Y})$ denote the total number of nodes reached by the branching process when the status of the edges are specified by $\boldsymbol{Y}$. Note that h is monotone nondecreasing in $\boldsymbol{Y}$. The distribution of Z_{cond} or Z_A can be simulated by drawing $\boldsymbol{Y}$ from the distribution induced by the corresponding branching process and then returning $h(\boldsymbol{Y})$. Consider any subset $E' \subseteq \{e \in V \times V\}$. We couple Z_{cond} and Z_A by having them share Y_e for all $e \notin E'$. We have

$$\Pr_{\boldsymbol{Y} \sim \text{BP-cond}|\{Y_e|e \notin E'\}} \left[Y_e = 1 \; \forall e \in E'\right] \leq \Pr_{\boldsymbol{Y} \sim \text{BP-A}|\{Y_e|e \notin E'\}} \left[Y_e = 1 \; \forall e \in E'\right]. \tag{A.2}$$

To see this, note that under BP-cond, the probability that $Y_e = 1$ $\forall e \in E'$ is either

- The probability of this event under the Y_e's original marginal distribution (drawn from the SBM and ICM) if setting them equal to 1 would not violate either condition for BP-cond.

- 0 if setting them to 1 would create a violation

However, BP-A always follows the first case, which assigns at least as high a probability to the event $Y_e = 1$ $\forall e \in E'$. The claim then follows from Equation A.2 combined with the monotonicity of h. □

This claim allows us to analyze BP-A in place of BP-cond. However, BP-A is still difficult to deal with because the sizes of the communities may be different. So, we introduce a process which simulates a graph where all communities have size $|C_i^*|$.

BP-B: Let G_B be a graph divided into communities of size $|C_i^*|$, with $\frac{n}{|C_i^*|}$ communities in total. Follow the same process as in in BP-A (starting from the same node), except on G_B instead of G_A. Z_B gives the total number of nodes reached.

Claim 2. *Z_A is stochastically dominated by Z_B.*

Proof. The idea is to interpolate between BP-A and BP-B by considering a series of local moves in which we split one of the communities in G_B into two smaller communities. Consider a series of graphs $G_B = G_1 .... G_W = G_A$ with the following property: G_{i+1} is equal to G_i except that a single community C_i of G_i is split into two communities C_i^1 and C_i^2. With each G_i, we can associate a branching process BP-i and corresponding Z_i. We will show that for any i, Z_i stochastically dominates Z_{i+1}. Since for any G_A there exists a sequence of local moves that can produce it from G_B, this will show that Z_B stochastically dominates Z_A.

To prove that Z_i stochastically dominates Z_{i+1}, we couple BP-i and BP-$(i+1)$ by sharing the status (live or not) of every edge in the graph between them. If BP-$(i+1)$ reaches either C_i^1 or C_i^2, then BP-i reaches $C_i = \ C_i^1 \cup C_i^2$. Hence, every community that is visited by BP-$(i+1)$ is also visited by BP-i. This establishes that Z_i stochastically dominates Z_{i+1}, as desired. □

BP-B is a nonuniform branching process in which the distribution of the number of children at each step depends on the total number of communities which remain to be explored. Note that G_B has $\frac{n}{|C_i^*|}$ communities in total. Suppose that BP-B has explored k communities so far. Define q_{eff} to be the “effective" probability of a live edge between two communities:

$$q_{\text{eff}} = 1 - (1 - p_b q)^{|C_i^*|^2}$$

By definition, we have $q_{\text{eff}} \frac{n}{|C_i^*|} \leq c_{max} < 1$. The number of children spawned by the kth community is distributed as $Bin(\frac{n}{|C_i^*|} - k, q_{\text{eff}})$. Since this nonuniform process is difficult to analyze, we note that it is stochastically dominated by a final branching process:

BP-uniform: A Galtson-Watson branching process with offspring distribution $Bin(\frac{n}{|C_i^*|}, q_{\text{eff}})$.

X_i represents the ith largest connected component in G, which we have established is stochastically dominated by the corresponding component generated by BP-B. For simplicity, we upper bound the ith largest component by the single largest component. In G_B there are at most $\frac{n}{|C_i^*|}$ components. The maximum of $\frac{n}{|C_i^*|}$ draws from BP-B is stochastically dominated by the maximum of $\frac{n}{|C_i^*|}$ draws of $Z_{uniform}$.

Claim 3. *Draw $Z_1 ... Z_N$ iid as $Z_{uniform}$. Then*

$$\mathbb{E}\left[\max Z_i\right] \leq 12 \left(\frac{1}{1 - \frac{n}{|C_i^*|} q_{\mathit{eff}}} \right) \log N$$

Proof. For any j, let ξ_j be iid from $Bin(\frac{n}{|C_i^*|}, q_{\text{eff}})$. Draief and Massoulie [DM10] (Lemma 1.9) give the following tail bound for Z_i:

$$\Pr[Z_i \geq K] \leq \Pr\left[\sum_{j=1}^{K} \xi_j \geq K\right]$$

$\sum_{j=1}^{K} \xi_j$ is distributed as $Bin(K \frac{n}{|C_i^*|}, q_{\text{eff}})$, so via Chernoff bound we have

$$\Pr\left[\sum_{j=1}^{K} \xi_j \geq K\right] \leq \exp\left(-\frac{1}{3} K \left(1 - \frac{n}{|C_i^*|} q_{\text{eff}}\right)\right)$$

So, we see that Z_i is stochastically dominated by an exponential random variable with mean $\lambda = \frac{1}{3}\left(1 - \frac{n}{|C_i^*|} q_{\text{eff}}\right)$. Dasarathy [Das11] (Eq. 7) show that the expected maximum of N exponential variables is upper bounded by $\frac{2 \log N}{\lambda\left(1 - \frac{1}{N}\right)}$. Noting that $1 - \frac{1}{N} \geq \frac{1}{2}$ and $\lambda \geq \frac{1}{3}(1 - c_{max})$, the claim follows. □

By substituting $N = \frac{n}{|C_i^*|}$ into Claim 3 and multiplying by $\beta(|C_i^*|)|C_i^*|$ (the size of the giant connected component of each community in G_A), we conclude the proof of the lemma.

We remark here that the reason we have a factor $1 - c_{max}$ and not $(1 - c_{max})^2$ is that we have bounded the *expectation* of the maximum of the N variables, not given a bound that holds with high probability.

□

With the key lemma in hand, we are now ready to proceed to the proof of our bound on OPT. Let $OPT(\boldsymbol{C}^*)$ be a random variable which gives the expected optimal value conditioned on $\boldsymbol{C}^*$.

$$\begin{aligned}
OPT &= \mathop{\mathbb{E}}_{\boldsymbol{C}^*}\left[\mathbb{E}\left[OPT(\boldsymbol{C}^*)\middle|\boldsymbol{C}^*\right]\right] \\
&\leq \mathop{\mathbb{E}}_{\boldsymbol{C}^*}\left[\sum_{i=1}^{K} X_i \middle| \boldsymbol{C}^*\right] \\
&\leq \mathop{\mathbb{E}}_{\boldsymbol{C}^*}\left[\sum_{i=1}^{K} \frac{12}{1 - c_{max}} \log\left(\frac{n}{|C_i^*|}\right) \beta(|C_i^*|)|C_i^*|\right] \quad \text{(Lemma 30)} \\
&\leq \sum_{i=1}^{K} \frac{12}{1 - c_{max}} \log\left(\frac{n}{|C_i|}\right) \beta(|C_i|)|C_i| \qquad\qquad (|C_i| \ \geq |C_i^*|) \\
&\leq \sum_{i=1}^{K} \frac{12}{1 - c_{max}} \log\left(\frac{n}{\beta(|C_i|)|C_i|}\right) \beta(|C_i|)|C_i|.
\end{aligned}$$

Given the guarantee that $\sum_{i=1}^{K} \frac{12}{1-c_{max}} \log\left(\frac{n}{\beta(|C_i|)|C_i|}\right) \beta(|C_i|)|C_i| \ \geq OPT$, we now analyze how small $\sum_{i=1}^{K} \beta(|C_i|)|C_i|$ can be. Define $y_i = \beta(|C_i|)|C_i|$. We are interested in the value of the optimization problem

$$\begin{aligned}
&\min_{y_1 \dots y_K} \sum_{i=1}^{K} y_i \\
&\text{s.t.} \ \sum_{i=1}^{K} \frac{12}{1 - c_{max}} \log\left(\frac{n}{y_i}\right) y_i \ \geq OPT
\end{aligned}$$

This can be reformulated as the convex program

$$\min_{y_1 \dots y_K} \sum_{i=1}^{K} y_i$$

$$\text{s.t. } -\sum_{i=1}^{K} \frac{12}{1-c_{max}} \log\left(\frac{n}{y_i}\right) y_i \; \leq -OPT$$

We structurally characterize the optimal solution as follows. Let v^* denote the optimal value of the above convex program. Note that Slater's condition holds, and so we have strong duality. Consider the Lagrange dual function

$$\mathcal{L}(\lambda) = \inf_{y_1...y_K} \sum_{i=1}^{K} y_i + \lambda \left(OPT - \sum_{i=1}^{K} \frac{12}{1-c_{max}} \log\left(\frac{n}{y_i}\right) y_i \right)$$

where the dual problem is

$$\max_{\lambda \geq 0} \mathcal{L}(\lambda).$$

Let λ^* be the optimal value of the Lagrange multiplier. We write

$$\begin{aligned} v^* &= \mathcal{L}(\lambda^*) \\ &= \inf_{y_1...y_K} \sum_{i=1}^{K} y_i + \lambda^* \left(OPT - \sum_{i=1}^{K} \frac{12}{1-c_{max}} \log\left(\frac{n}{y_i}\right) y_i \right) \end{aligned} \tag{A.3}$$

Examining Equation A.3, let $y_i^*...y_K^*$ be values of $y_1...y_K$ which achieve v^*. We must have that $y_i^*...y_K^*$ maximize $\sum_{i=1}^{K} \log\left(\frac{n}{y_i^*}\right) a_i$ subject to $\sum_{i=1}^{K} y_i^* = v^*$ (otherwise a smaller value could have been achieved). Since $\frac{12}{1-c_{max}} \log\left(\frac{n}{y_i^*}\right) a_i$ is concave, Jensen's inequality gives

$$\sum_{i=1}^{K} \log\left(\frac{n}{y_i^*}\right) y_i^* \leq K \log\left(\frac{n}{\frac{1}{K}\sum_{i=1}^{K} y_i^*}\right) \left(\frac{1}{K}\sum_{i=1}^{K} y_i^*\right).$$

That is, $\sum_{i=1}^{K} \log\left(\frac{n}{y_i^*}\right) y_i^*$ is maximized when $y_1 = y_2 = ... = y_K = \frac{1}{K}\sum_{i=1}^{K} y_i$. Thus, the optimal value v^* can be obtained when we restrict the space of feasible $y_1...y_K$ to points where all are equal. Let $\mu = \frac{1}{K}\sum_{i=1}^{K} \beta(|C_i|)|C_i|$ denote the average size of the giant components of the top K communities. We rephrase the original optimization problem as

$$\min_{\mu} \mu K$$
$$\text{s.t. } \mu K \left(\frac{12}{1 - c_{max}} \log \left(\frac{n}{\mu} \right) \right) \geq OPT$$

The constraint in this problem gives a lower bound on the possible size of μK. Thus we have

$$\sum_{i=1}^{K} |C_i| \ = \mu K \geq \left(\frac{1 - c_{max}}{12 \log \frac{n}{\mu}} \right) OPT$$

which concludes the proof of the lemma. □

A.2 Estimating the surrogate objective g

In this section, we explain more detail our procedure for estimating the surrogate objective (g). Recall that we defined $g(X) = \sum_{i=1}^{L} f(X, C_i)$, i.e, the influence spread of X considering only within-community edges. We would like a way of estimated $\mathbb{E}[g(X)]$ using only local information. Note that the influence spread within each C_i depends only on the nodes in $X \cap C_i$, which we write as X_{C_i} for short. So, $\mathbb{E}[g(X)]$ can be rewritten as $\mathbb{E}\left[\sum_{i=1}^{L} f(X_{C_i}, C_i)\right]$. If we knew X_{C_i}, then we could calculate $\mathbb{E}\left[f(X_{C_i}, C_i)\right]$ by simulating draws from the SBM for the unobserved portions of C_i conditioned on the presence of the subgraphs that the algorithm visited. Thus, the main challenge is that we do not know what community each node belongs to.

We start out by rewriting the influence bound in terms of the marginal contribution made by each v_i. Let $\chi(v)$ give the community of vertex v. We can write the bound as

$$g(X) = \sum_{i=1}^{T} \mathop{\mathbb{E}}_{X \sim \boldsymbol{w}} \left[f(X_{\chi(v_i)} \cap \{v_1 ... v_i\}, \chi(v_i) - f(X_{\chi(v_i)} \cap \{v_1 ... v_{i-1}\}, \chi(v_i))\right]$$

where $X \sim \boldsymbol{w}$ denotes a seed set X with each element independently sampled with probability proportional to $\boldsymbol{w}$. Taken at face value, this does not seem like an improvement because we still do not know $X_{\chi(v_i)}$ for each term. However, since we have an estimate for the size of $\chi(v_i)$, we know (approximately) how many other times $\chi(v_i)$ will have been sampled as well (approximately) the weight that each of these samples will have received. For each node, we can simulate a set $sim(v_i)$ which contains v_i plus a sample from the distribution of the other nodes that ARISEN sampled from $\chi(v_i)$ in its random walks. The only issue is that we do not know where each node of $sim(v_i)$ lies in the order $\{v_1 ... v_T\}$, i.e., whether it takes “precedence" over v_i when we compute the marginal contributions. The final ingredient we need to overcome this obstacle is to realize that there is nothing special about the ordering $\{v_1 ... v_T\}$; we can equivalently rearrange the nodes in any order. In fact, we take the expectation over a uniformly random permutation π of the ordering: we first draw π and then sum in the order $v_{\pi(1)} .... v_{\pi(T)}$. Via linearity of expectation, we can take a different permutation for each term $i = 1....T$, where the permutation in term i need only a establish an ordering among the elements of $sim(v_i)$. For any set X, let $[X]^i_\pi$ represent the first i elements of X in the permutation π. Then we can write the influence bound as

$$g(X) = \sum_{i=1}^{T} \mathop{\mathbb{E}}_{\pi, sim(v_i), X} \left[f([X \cap sim(v_i)]^i_\pi, \chi(v_i)) - f([X \cap sim(v_i)]^{i-1}_\pi, \chi(v_i)) \right].$$

In this final form, we can calculate each term by averaging over simulations of $sim(v_i)$, an ordering π on $sim(v_i)$, and set of seed nodes from $sim(v_i)$ that are chosen (given the simulated weights). As discussed earlier, we can the compute f by averaging over simulations of the draw of C_i, and simulating the ICM on each simulated community. Complete pseudocode for EstVal is given in Algorithm 14. The proof that EstVal accurately estimates g follows immediately from the construction given above.

Algorithm 14 EstVal

```
for $i = 1...len(\boldsymbol{w})$ do
    for $j = 1...M$ do
        Simulate $G_i^j$ from $\mathcal{G}(p_w, \hat{S}_i)$ conditioned on $H_i$ appearing.
        for $k = 1...P$ do
            $\pi$ = a uniformly random permutation on $V(G_i^j)$
            $N \sim$ Binom $(T, \frac{\hat{S}_i}{n})$
            Draw $u_1...u_N$ uniformly random from $V(G_i^j) \setminus V(H_i)$
            for $\ell = 1...N$ do
                $w_\ell^{samp}$ = weight $\boldsymbol{w}$ assigns to a node with value $f(u_\ell, G_i^j)$
            end for
            $X$ = a random subset of $u_1...u_N$ when $K - 1$ nodes are chosen from all samples, the total weight is $||\boldsymbol{w}||_1$, and $u_1...u_N$ have corresponding weights from $w^{samp}$
            for $u \in X$ do
                if $\pi(s_i) > \pi(u)$, remove $u$ from $X$
            end for
            $val+ = \frac{1}{MP}\left(1 - \left(1 - \frac{w_i}{||\boldsymbol{w}||_1}\right)^K\right)\left[f(\{s_i\} \cup X, G_i^j) - f(X, G_i^j)\right]$
        end for
    end for
end for
return $val$
```

A.3 Additional experimental results

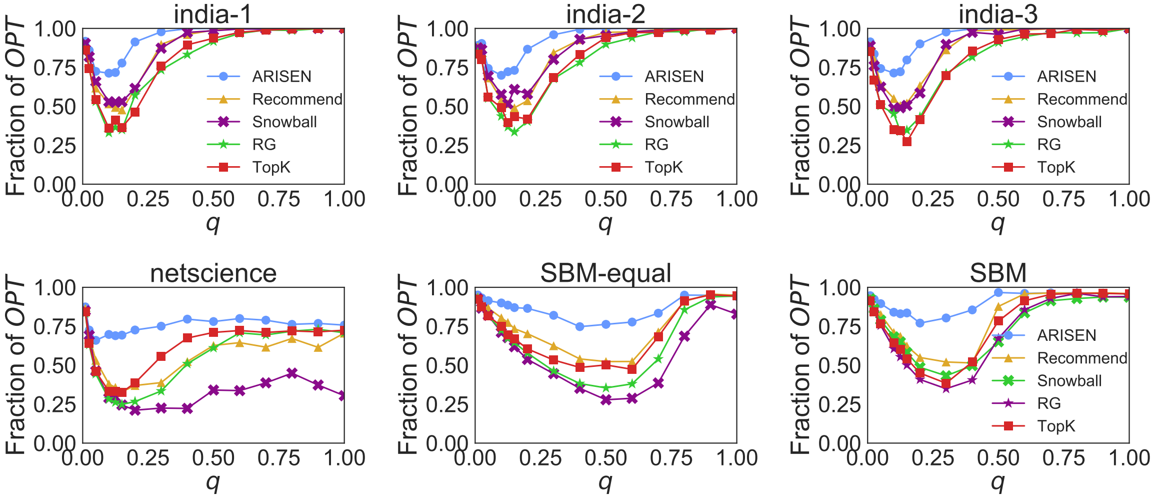
A.3.1 Parameter settings

In all runs we set $B = 0$ (no burn-in). The values for R and T can be found in the table below.

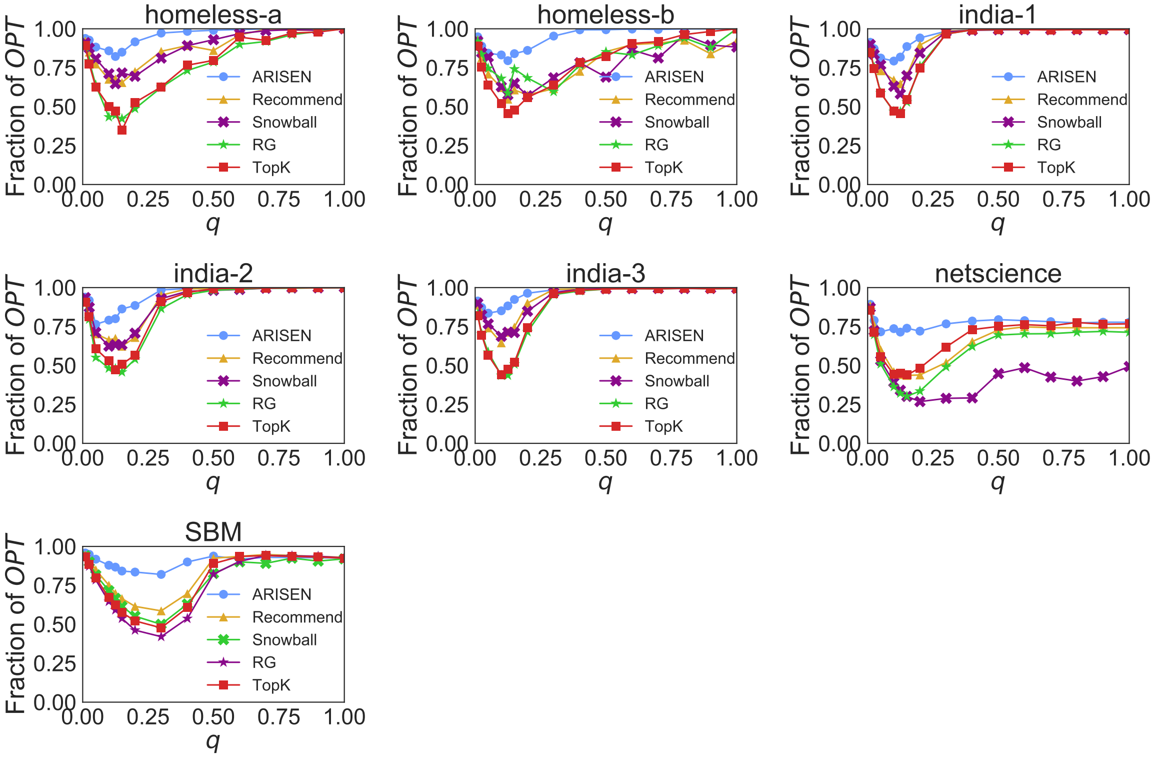
Network	K	T	R
homeless-a	$0.01 \cdot n$	5	10
homeless-a	$0.015 \cdot n$	5	10
homeless-a	$0.02 \cdot n$	5	10
homeless-b	$0.01 \cdot n$	7	12
homeless-b	$0.015 \cdot n$	7	12
homeless-b	$0.02 \cdot n$	7	12
india-1	$0.005 \cdot n$	10	15
india-1	$0.01 \cdot n$	10	15
india-1	$0.015 \cdot n$	10	15
india-1	$0.02 \cdot n$	10	15
india-2	$0.005 \cdot n$	7	12
india-2	$0.01 \cdot n$	10	12
india-2	$0.015 \cdot n$	10	12
india-2	$0.02 \cdot n$	10	12
india-2	$0.005 \cdot n$	6	25
india-2	$0.01 \cdot n$	12	25
india-2	$0.015 \cdot n$	18	25
india-2	$0.02 \cdot n$	25	25
netscience	$0.005 \cdot n$	40	25
netscience	$0.01 \cdot n$	40	25
netscience	$0.015 \cdot n$	40	25
netscience	$0.02 \cdot n$	40	25
SBM	$0.005 \cdot n$	6	25
SBM	$0.01 \cdot n$	12	25
SBM	$0.015 \cdot n$	18	25
SBM	$0.02 \cdot n$	25	25

A.3.2 Influence spread

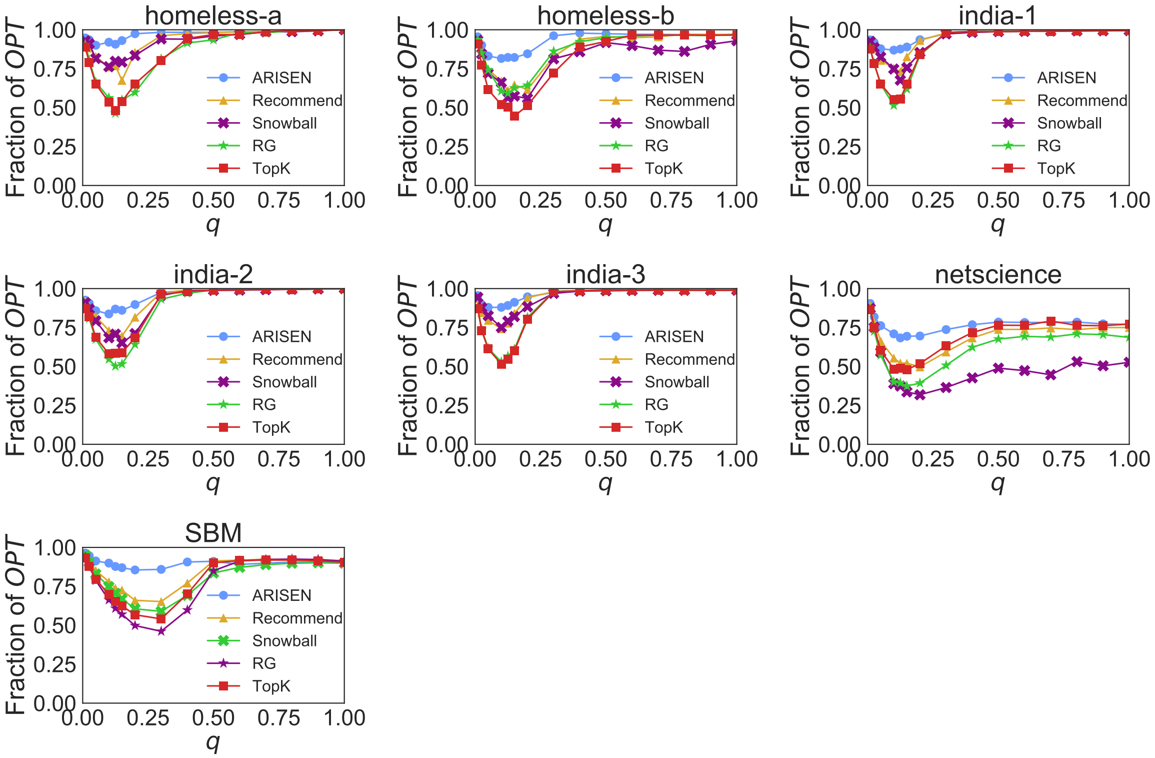
$K = 0.005 \cdot n$



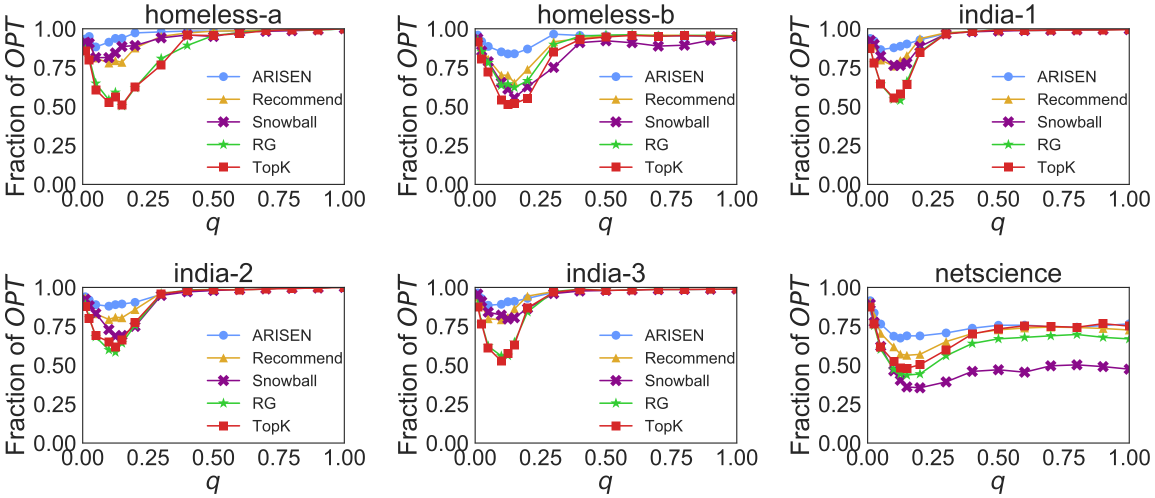
$K = 0.01 \cdot n$

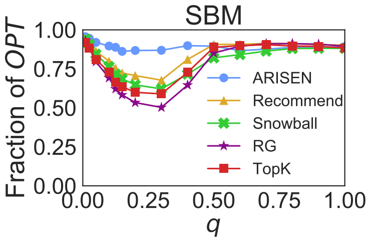


$K = 0.015 \cdot n$

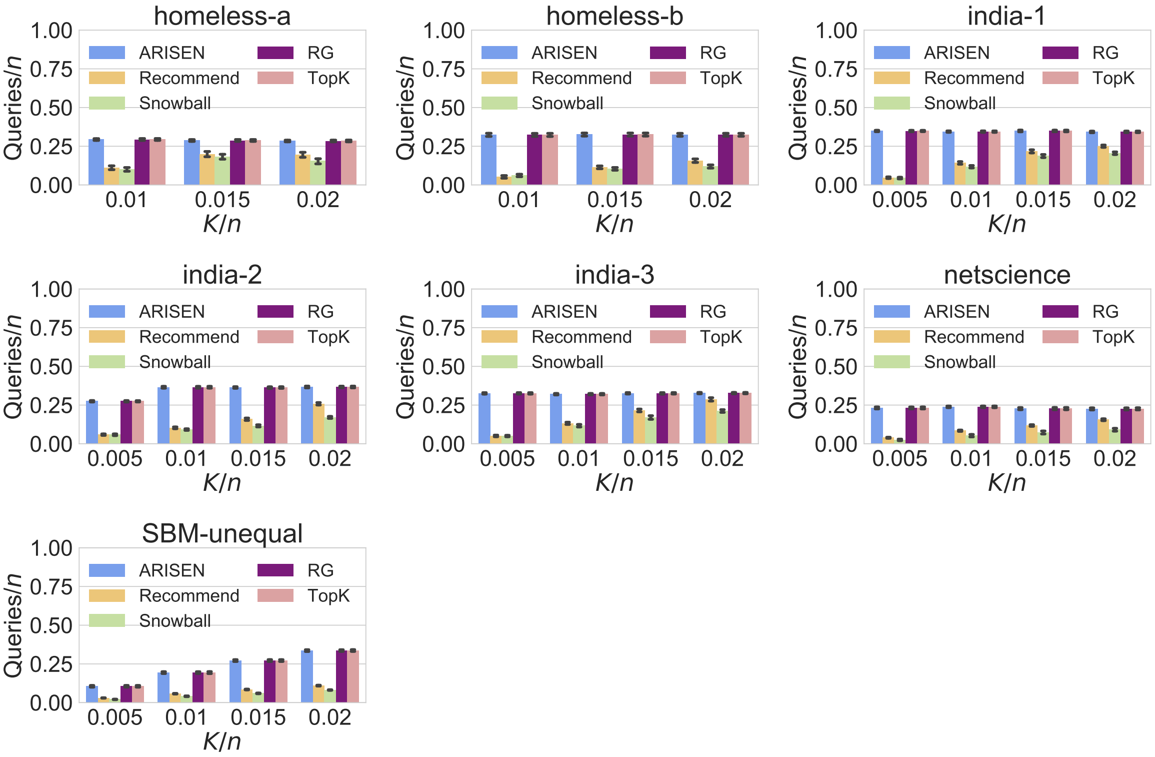


$K = 0.02 \cdot n$





A.3.3 Query cost



Appendix B

Appendix to Chapter 2

B.1 Missing proofs

We start out by proving some lemmas from the main text.

Proof of Lemma 2. Let $\boldsymbol{u} \succeq 0$. We would like to show that for any $\boldsymbol{x}$ and any $\xi \geq 0$, $G(\boldsymbol{x} + \xi \boldsymbol{u})$ is concave as a function of ξ. Fix any $\xi_1, \xi_2 \geq 0$ and any $\lambda \in [0, 1]$. We have

$$\begin{aligned} \min_i F_i(\boldsymbol{x} + (\lambda\xi_1 + (1-\lambda)\xi_2)\boldsymbol{u}) &\geq \min_i \left[\lambda F_i(\boldsymbol{x} + \xi_1 \boldsymbol{u}) + (1-\lambda) F_i(\boldsymbol{x} + \xi_2 \boldsymbol{u})\right] \\ &\geq \lambda \min_i F_i(\boldsymbol{x} + \xi_1 \boldsymbol{u}) + (1-\lambda) \min_i F_i(\boldsymbol{x} + \xi_2 \boldsymbol{u}) \end{aligned}$$

where the first inequality follows because each F_i is individually up-concave. □

Proof of Lemma 46. G is differentiable at a point $\boldsymbol{x}$ precisely when there is a unique F_i such that $F_i(\boldsymbol{x}) = \min_j F_j(\boldsymbol{x})$. Here, we have $\nabla G(\boldsymbol{x}) = \nabla F_i(\boldsymbol{x})$. Note that $\frac{\partial F_i}{\partial x_j}\Big|_{\boldsymbol{x}} = \mathbb{E}[f_i(R(\boldsymbol{x})|j \in R(\boldsymbol{x}))] - \mathbb{E}[f_i(R(\boldsymbol{x})|j \notin R(\boldsymbol{x}))] = \mathbb{E}[f_i(j|R(\boldsymbol{x} - x_j))]$. By submodularity, we conclude that $\frac{\partial F_i}{\partial x_j}\Big|_{\boldsymbol{x}} \leq f_i(\{j\}) \leq M$. Further, $\frac{\partial F_i}{\partial x_j}\Big|_{\boldsymbol{x}} \geq 0$ always holds by monotonicity. Thus, $||\nabla G(\boldsymbol{x})||_\infty \leq M$. □

Let μ be the uniform probability distribution over the ℓ_∞ ball of radius u. Define the smoothed function $G_\mu(\boldsymbol{x}) = \mathbb{E}_{z \sim \mu}[G(\boldsymbol{x} + \boldsymbol{z})]$. We will show the following properties of G_μ:

Proof of Lemma 4. For the first property, we start out by fixing the draw of $\boldsymbol{z}$ from μ. Following the logic of Lemma 2, we have that

$$\begin{aligned}\min_i F_i(\boldsymbol{x}+\boldsymbol{z}+(\lambda\xi_1+(1-\lambda)\xi_2)\boldsymbol{u}) &\geq \min_i \lambda F_i(\boldsymbol{x}+\boldsymbol{z}\xi_1\boldsymbol{u})+(1-\lambda)F_i(\boldsymbol{x}+\boldsymbol{z}+\xi_2\boldsymbol{u})\\ &\geq \lambda\min_i F_i(\boldsymbol{x}+\boldsymbol{z}+\xi_1\boldsymbol{u})+(1-\lambda)\min_i F_i(\boldsymbol{x}+\boldsymbol{z}+\xi_2\boldsymbol{u}).\end{aligned}$$

Since these inequalities hold for any fixed $\boldsymbol{z}$, they also hold in expectation over a random $\boldsymbol{z}$, so we conclude that G_μ is up-concave.

For the second property: since $||\nabla G||_\infty \leq M$, G is M-Lipschitz with respect to the ℓ_1 norm. Thus, we have

$$\mathbb{E}[G(\boldsymbol{x}+\boldsymbol{z})] \leq G(\boldsymbol{x}) + M\,\mathbb{E}[||\boldsymbol{z}||_1] \leq G(\boldsymbol{x}) + \frac{Mnu}{2}$$

and analogously, $\mathbb{E}[G(\boldsymbol{x}+\boldsymbol{z})] \geq G(\boldsymbol{x}) - \frac{Mnu}{2}$.

The third property follows from the fact that G is differentiable almost everywhere. To see this, note that G is differentiable wherever there is a unique minimizing F_i, in which case $\nabla G = \nabla F_i$. Suppose that there is not a unique minimizer at some point $\boldsymbol{x}$. There are two cases. First, if there is an open ball around $\boldsymbol{x}$ such that the minimizing functions at $\boldsymbol{x}$ coincide at every point in the ball, then their gradients also coincide in the ball. Thus, G is still differentiable at $\boldsymbol{x}$. Second, if no such open ball exists, then the set of points at which G is not differentiable has measure zero. Hence, taking a random perturbation of the input avoids such points with probability 1.

For the proof of the fourth property, we follow the argument of Duchi et al. (2012). We first claim that

$$||\nabla G_\mu(\boldsymbol{x}) - \nabla G_\mu(\boldsymbol{y})||_\infty = ||\,\mathbb{E}\left[\nabla G(\boldsymbol{x}+\boldsymbol{z})\right] - \mathbb{E}\left[\nabla G(\boldsymbol{y}+\boldsymbol{z})\right]||_\infty \leq M\int |\mu(\boldsymbol{z}-\boldsymbol{x}) - \mu(\boldsymbol{z}-\boldsymbol{y})|d\boldsymbol{z}. \tag{B.1}$$

We prove this claim as follows. Without loss of generality, we take $\boldsymbol{x} = 0$ for this step of the proof (via a linear change of variables). Let $g(\boldsymbol{x})$ be a function that is defined as $\nabla G(\boldsymbol{x})$ where G is differentiable. At the (measure 0) set of points where G is not differentiable, we define g to be equal to $\nabla F_i(\boldsymbol{x})$ for an arbitrary $i \in \arg\min_j F_j(\boldsymbol{x})$. With probability 1, $\mathbb{E}[g(\boldsymbol{x}+\boldsymbol{z})] = \mathbb{E}[\nabla G(\boldsymbol{x}+\boldsymbol{z})]$ We have

$$\begin{aligned}
\mathbb{E}[g(\boldsymbol{z}) - g(\boldsymbol{y}+\boldsymbol{z})] &= \int g(\boldsymbol{z})\mu(\boldsymbol{z})dz - \int g(\boldsymbol{y}+\boldsymbol{z})\mu(\boldsymbol{z})dz \\
&= \int g(\boldsymbol{z})\mu(\boldsymbol{z})dz - \int g(\boldsymbol{z})\mu(\boldsymbol{z}-\boldsymbol{y})dz \\
&= \int_{I_>} g(\boldsymbol{z})\left[\mu(\boldsymbol{z}) - \mu(\boldsymbol{z}-\boldsymbol{y})\right] + \int_{I_<} g(\boldsymbol{z})\left[\mu(\boldsymbol{z}-\boldsymbol{y}) - \mu(\boldsymbol{z})\right]
\end{aligned}$$

where $I_> = \{\boldsymbol{z}|\mu(\boldsymbol{z}) > \mu(\boldsymbol{z}-\boldsymbol{y})\}$ and $I_< = \{\boldsymbol{z}|\mu(\boldsymbol{z}) < \mu(\boldsymbol{z}-\boldsymbol{y})\}$. Taking norms, we have

$$\begin{aligned}
||\,\mathbb{E}[g(\boldsymbol{z}) - g(\boldsymbol{y}+\boldsymbol{z})]||_\infty &\leq \sup_{z \in I_> \cup I_<} ||g(\boldsymbol{z})||_\infty \left| \int_{I_>} \left[\mu(\boldsymbol{z}) - \mu(\boldsymbol{z}-\boldsymbol{y})\right] + \int_{I_<} \left[\mu(\boldsymbol{z}-\boldsymbol{y}) - \mu(\boldsymbol{z})\right] \right| \\
&\leq M \left| \int_{I_>} \left[\mu(\boldsymbol{z}) - \mu(\boldsymbol{z}-\boldsymbol{y})\right] + \int_{I_<} \left[\mu(\boldsymbol{z}-\boldsymbol{y}) - \mu(\boldsymbol{z})\right] \right| \\
&= M \int |\mu(\boldsymbol{z}) - \mu(\boldsymbol{z}-\boldsymbol{y})|dz
\end{aligned}$$

Having proved that Equation B.1 holds, we now just need to show $\int |\mu(\boldsymbol{z}-\boldsymbol{x}) - \mu(\boldsymbol{z}-\boldsymbol{y})|dz \leq \frac{||\boldsymbol{x}-\boldsymbol{y}||_1}{u}$. This follows from Duchi et al. (2012), Lemma 12.

□

We now prove a technical smoothness lemma. The argument is standard, but we include it for completeness.

Lemma 31. *For any $\boldsymbol{x}, \boldsymbol{y}$, $G_\mu(\boldsymbol{x}+\gamma\boldsymbol{y}) - G_\mu(\boldsymbol{x}) \geq \gamma \nabla G_\mu(\boldsymbol{x})^T \boldsymbol{y} - \frac{Mk^2\gamma^2}{2u}$.*

Proof. For any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{P}$, we consider the one dimensional auxiliary function $g_{\boldsymbol{x},\boldsymbol{y}}(\xi) = G_\mu(\boldsymbol{x}+\xi\boldsymbol{y})$. We have

$$
\begin{aligned}
&G_\mu(\boldsymbol{x}+\gamma\boldsymbol{y}) - G_\mu(\boldsymbol{x}) = \int_{\xi=0}^{1} \frac{dg_{\boldsymbol{x},\gamma\boldsymbol{y}}(\xi)}{d\xi} d\xi \\
&= \int_{\xi=0}^{1} \nabla G_\mu(\boldsymbol{x}+\xi\gamma\boldsymbol{y})^\top(\gamma\boldsymbol{y})d\xi \\
&= \gamma\int_{\xi=0}^{1} \nabla G_\mu(\boldsymbol{x})^\top\boldsymbol{y} + \left[\nabla G_\mu(\boldsymbol{x}+\xi\gamma\boldsymbol{y})^\top - \nabla G_\mu(\boldsymbol{x})^\top\right]\boldsymbol{y}d\xi \\
&\geq \gamma\int_{\xi=0}^{1} \nabla G_\mu(\boldsymbol{x})^\top\boldsymbol{y} - ||\nabla G_\mu(\boldsymbol{x}+\xi\gamma\boldsymbol{y})^\top - \nabla G_\mu(\boldsymbol{x})^\top||_\infty||\boldsymbol{y}||_1 d\xi \text{ (by Hölder's inequality)} \\
&\geq \gamma\int_{\xi=0}^{1} \nabla G_\mu(\boldsymbol{x})^\top\boldsymbol{y} - \frac{M}{\mu}||\xi\gamma\boldsymbol{y}||_1||\boldsymbol{y}||_1 d\xi \text{ (}\nabla G_\mu \text{ is } \frac{M}{\mu}\text{)-Lipschitz} \\
&\geq \gamma\nabla G_\mu(\boldsymbol{x})^\top - \gamma^2\int_{\xi=0}^{1}\frac{Mk^2}{u}\xi d\xi \text{ (bound on } \ell_1 \text{ diameter of } \mathcal{P}\text{)} \\
&= \gamma\nabla G_\mu(\boldsymbol{x})^\top - \frac{\gamma^2 Mk^2}{2u}
\end{aligned}
$$

which proves the lemma. □

We also use the following lemma, the proof of which can be found in Bian et al. (2017):

Lemma 32. *For any DR-submodular function G and its optimizer $\boldsymbol{x}^*$, $G(\boldsymbol{x}^*+\boldsymbol{x}) - G(\boldsymbol{x}) \leq \nabla G(\boldsymbol{x})^\top\boldsymbol{x}^*$.*

We can now proceed to prove our guarantee on the performance of the SFW algorithm for optimizing the objective G.

Proof of Theorem 4. We analyze the gain made in a single step of SFW as follows:

$$
\begin{aligned}
&G_\mu(\boldsymbol{x}^\ell) - G_\mu(\boldsymbol{x}^{\ell-1}) \geq \gamma_\ell\nabla G_\mu(\boldsymbol{x}^{\ell-1})^\top\boldsymbol{v}^\ell - \frac{Mk^2}{2u}\gamma_\ell^2 \text{ (Lemma 43)} \\
&= \gamma_\ell\tilde{\nabla}_\ell^\top\boldsymbol{v}^\ell - \gamma_\ell\left(\tilde{\nabla}_\ell - \nabla G_\mu(\boldsymbol{x}^{\ell-1})\right)^\top\boldsymbol{v}^\ell - \frac{Mk^2}{2u}\gamma_\ell^2 \\
&\geq \gamma_\ell\tilde{\nabla}_\ell^\top\boldsymbol{v}^\ell - \gamma_\ell k||\tilde{\nabla}_\ell - \nabla G_\mu(\boldsymbol{x}^{\ell-1})||_\infty - \frac{Mk^2}{2u}\gamma_\ell^2 \text{ (Hölder's inequality and } rank(\mathcal{M}) = k\text{)} \\
&\geq \gamma_\ell\tilde{\nabla}_\ell^\top\boldsymbol{x}^* - \gamma_\ell k||\tilde{\nabla}_\ell - \nabla G_\mu(\boldsymbol{x}^{\ell-1})||_\infty - \frac{Mk^2}{2u}\gamma_\ell^2 \text{ (by definition of } \boldsymbol{x}^*\text{)} \\
&= \gamma_\ell\nabla G_\mu(\boldsymbol{x}^{\ell-1})^\top\boldsymbol{x}^* - \gamma_\ell\left(G_\mu(\boldsymbol{x}^{\ell-1}) - \tilde{\nabla}_\ell\right)^\top\boldsymbol{x}^* - \gamma_\ell k||\tilde{\nabla}_\ell - \nabla G_\mu(\boldsymbol{x}^{\ell-1})||_\infty - \frac{Mk^2}{2u}\gamma_\ell^2 \\
&\geq \gamma_\ell\nabla G_\mu(\boldsymbol{x}^{\ell-1})^\top\boldsymbol{x}^* - 2\gamma_\ell k||\tilde{\nabla}_\ell - \nabla G_\mu(\boldsymbol{x}^{\ell-1})||_\infty - \frac{Mk^2}{2u}\gamma_\ell^2
\end{aligned}
$$

$$\geq \gamma_\ell \left(G_\mu(\boldsymbol{x}^* + \boldsymbol{x}^{\ell-1}) - G_\mu(\boldsymbol{x}^{\ell-1}) \right) - 2\gamma_\ell k ||\tilde{\nabla}_\ell - \nabla G_\mu(\boldsymbol{x}^{\ell-1})||_\infty - \frac{Mk^2}{2u}\gamma_\ell^2$$

(Lemma 44 and $\boldsymbol{x}^* \succeq 0$)

$$\geq \gamma_\ell \left(G_\mu(\boldsymbol{x}^*) - G_\mu(\boldsymbol{x}^{\ell-1}) \right) - 2\gamma_\ell k ||\tilde{\nabla}_\ell - \nabla G_\mu(\boldsymbol{x}^{\ell-1})||_\infty - \frac{Mk^2}{2u}\gamma_\ell^2 \text{ (monotonicity)}$$

Now we give a high probability bound on $||\tilde{\nabla}_\ell - \nabla G_\mu(\boldsymbol{x}^{\ell-1})||_\infty$. Denote by $\tilde{\nabla}_\ell^i$ the ith randomly sampled gradient and $\left[\tilde{\nabla}_\ell^i\right]_j$ its jth entry (the derivative with respect to item j). We will give a high probability bound on each individual entry of the estimated gradient and them combine them using union bound to control $||\tilde{\nabla}_\ell - \nabla G_\mu(\boldsymbol{x}^{\ell-1})||_\infty$.

Fix any individual entry j. We have $\left[\tilde{\nabla}_\ell\right]_j = \frac{1}{c}\sum_{i=1}^{c} \left[\tilde{\nabla}_\ell^i\right]_j$. Because the first-order oracle returns an unbiased estimate, we know that $\mathbb{E}\left[\left[\tilde{\nabla}_\ell\right]_j - \nabla_j G_\mu(\boldsymbol{x}^{\ell-1})\right] = 0$. Further, $\left|\left[\tilde{\nabla}_\ell\right]_j\right| \leq M$ and $\left|\nabla_j G_\mu(\boldsymbol{x}^{\ell-1})\right| \leq M$, so $\left|\left[\tilde{\nabla}_\ell\right]_j - \nabla_j G_\mu(\boldsymbol{x}^{\ell-1})\right| \leq 2M$ holds via triangle inequality. Now via Hoeffding's inequality, we have that

$$\Pr\left[\left|\sum_{i=1}^{c} \left[\tilde{\nabla}_\ell^i\right]_j - c\nabla_j G_\mu(\boldsymbol{x}^{\ell-1})\right| \geq m\frac{\epsilon}{8k}\right] \leq 2e^{-\frac{\epsilon^2 c}{128k^2M^2}}$$

and so taking $c = \frac{128M^2k^2}{\epsilon^2}\log\frac{4Kn}{\delta}$ ensures that

$$\Pr\left[\left|\left[\tilde{\nabla}_\ell\right]_j - \nabla_j G_\mu(\boldsymbol{x}^{\ell-1})\right| \geq \frac{\epsilon}{8k}\right] \leq \frac{\delta}{2Kn}.$$

By union bound, the total probability of this event holding for all n items at each of the K timesteps is at least $1 - \frac{\delta}{2}$. In all of what follows, we condition on this happening. Rearranging gives

$$\begin{aligned} G_\mu(\boldsymbol{x}^*) - G_\mu(\boldsymbol{x}^\ell) &\leq (1-\gamma_\ell)\left[G_\mu(\boldsymbol{x}^*) - G_\mu(\boldsymbol{x}^{\ell-1})\right] - 2\gamma_\ell k||\tilde{\nabla}_\ell - \nabla G_\mu(\boldsymbol{x}^{\ell-1})||_\infty - \frac{Mk^2}{2u}\gamma_\ell^2 \\ &\leq (1-\gamma_\ell)\left[G_\mu(\boldsymbol{x}^*) - G_\mu(\boldsymbol{x}^{\ell-1})\right] - \frac{\gamma_\ell \epsilon}{4} - \frac{Mk^2}{2u}\gamma_\ell^2 \end{aligned}$$

and so after K iterations we obtain

$$
\begin{aligned}
G_\mu(\boldsymbol{x}^*) - G_\mu(\boldsymbol{x}^K) &\leq \prod_{\ell=0}^{K-1}(1-\gamma_\ell)\left[G_\mu(\boldsymbol{x}^*) - G_\mu(\boldsymbol{x}^0)\right] - \sum_{\ell=0}^{K-1}\frac{\gamma_\ell \epsilon}{4} - \sum_{\ell=0}^{K-1}\frac{Mk^2}{2u}\gamma_\ell^2 \\
&\leq e^{-\sum_{\ell=0}^{K-1}\gamma_\ell} G_\mu(\boldsymbol{x}^*) - \sum_{\ell=0}^{K-1}\frac{\gamma_\ell \epsilon}{4} - \sum_{\ell=0}^{K-1}\frac{Mk^2}{2u}\gamma_\ell^2
\end{aligned}
$$

with constant stepsize $\gamma = \frac{1}{K}$, we have $\sum_{\ell=0}^{K-1}\gamma_\ell = 1$ and $\sum_{\ell=0}^{K-1}\gamma_\ell^2 = \frac{1}{K}$, this yields

$$
G_\mu(\boldsymbol{x}^*) - G_\mu(\boldsymbol{x}^K) \leq \frac{1}{e}G_\mu(\boldsymbol{x}^*) - \frac{\epsilon}{4} - \frac{Mk^2}{2uK}
$$

and hence

$$
G(\boldsymbol{x}^*) - G(\boldsymbol{x}^K) \leq \frac{1}{e}G_\mu(\boldsymbol{x}^*) - \frac{\epsilon}{4} - \frac{Mk^2}{2uK} - Mnu.
$$

and so taking $u = \frac{\epsilon}{4Mn}$ and $K = \frac{8M^2k^2n}{\epsilon^2}$ ensures that $G(\boldsymbol{x}^*) - G(\boldsymbol{x}^K) \leq \frac{1}{e}G_\mu(\boldsymbol{x}^*) - \frac{3\epsilon}{4}$

Now we just need a small trick to deal with the issue that G is only defined for $\boldsymbol{x} \in [0,1]^n$, and random perturbation by $\boldsymbol{z}$ may take us out of this range. To avoid negative values, we start the algorithm at the point $u\mathbf{1}$. Since each coordinate only increases during the course of the algorithm, we are guaranteed to query G only at nonnegative points. To deal with values greater than 1, we can instead analyze the function $H(\boldsymbol{x}) = G(\boldsymbol{x} \wedge \mathbf{1})$, where $\wedge$ denotes coordinate-wise maximum. H is also up-concave, and agrees with G at every point in $\mathcal{P}(\mathcal{M})$. After running SFW applied to H for K iterations, we obtain via Theorem 4 a solution $\boldsymbol{x}^K$ such that $H(\boldsymbol{x}^K) \geq (1-\frac{1}{e})\max_{\boldsymbol{x}\in\mathcal{P}(\mathcal{M})} H(\boldsymbol{x}) - \epsilon = (1-\frac{1}{e})\max_{\boldsymbol{x}\in\mathcal{P}(\mathcal{M})} G(\boldsymbol{x}) - \epsilon$. The issue is that we may have $\boldsymbol{x}^K \notin \mathcal{P}(\mathcal{M})$. We output the point $\boldsymbol{x}^K - u\mathbf{1}$, which is guaranteed to lie in $\mathcal{P}(\mathcal{M})$. To analyze the loss incurred, we use the following lemma

Lemma 33. *Let f be a monotone submodular function with $\max_i f(\{i\}) \leq M$. Let $R(\boldsymbol{x})$ be a random set in which every element appears independently with probability $x_i \geq u$. Then $\mathbb{E}[f(R(\boldsymbol{x} - u\mathbf{1}))] \geq \mathbb{E}[f(R(\boldsymbol{x}))] - uMn$.*

Proof. We decompose the expected value of $R(\boldsymbol{x} - u\mathbf{1})$ into the expected marginal contribution from each item:

$$\begin{aligned}
\mathbb{E}[f(R(\boldsymbol{x} - u\mathbf{1}))] &= \sum_{i=1}^{n} \Pr[i \in f(R(\boldsymbol{x} - u\mathbf{1}))]\, \mathbb{E}[f(i|(R(\boldsymbol{x} - u\mathbf{1}))] && \text{(linearity of expectation)} \\
&\geq \sum_{i=1}^{n} \Pr[i \in f(R(\boldsymbol{x} - u\mathbf{1}))]\, \mathbb{E}[f(i|(R(\boldsymbol{x}))] && \text{(submodularity)} \\
&= \sum_{i=1}^{n} (x_i - u)\, \mathbb{E}[f(i|(R(\boldsymbol{x}))] \\
&= \sum_{i=1}^{n} x_i\, \mathbb{E}[f(i|(R(\boldsymbol{x}))] - u \sum_{i=1}^{n} \mathbb{E}[f(i|(R(\boldsymbol{x}))] \\
&\geq \sum_{i=1}^{n} x_i\, \mathbb{E}[f(i|(R(\boldsymbol{x}))] - u \sum_{i=1}^{n} \mathbb{E}[f(\{i\})] && \text{(submodularity)} \\
&\geq \sum_{i=1}^{n} x_i\, \mathbb{E}[f(i|(R(\boldsymbol{x}))] - unM && (f(\{i\}) \leq M) \\
&= \mathbb{E}[f(R(\boldsymbol{x}))] - uMn.
\end{aligned}$$

□

Applying Lemma 33 to every $f_i \in \mathcal{F}$, we conclude that

$$H(\boldsymbol{x}^K - u\mathbf{1}) \geq H(\boldsymbol{x}^K) - uMn = G(\boldsymbol{x}^K) - \frac{\epsilon}{4} \geq \left(1 - \frac{1}{e}\right) G(\boldsymbol{x}^*) - \epsilon$$

which completes the proof. □

Lastly, we prove our concentration guarantee for the output of the swap rounding algorithm.

Proof of Theorem 6. For now, fix a specific function f_i. We will show that with high probability, the expected value of f_i on the empirical distribution is close to its expected value on the full distribution induced by randomized swap rounding. At the end we will take a union bound over all $f_i \in \mathcal{F}$. Let R_ℓ be the random set drawn in the ℓth iteration of randomized swap rounding. Let $\mu_0 = F_i(\boldsymbol{x}^K)$. Note that for all ℓ, $\mathbb{E}[f_i(R_\ell)] \geq \mu_0$ via the guarantee for randomized swap rounding. Let $Y = \sum_{\ell=1}^{r} f_i(R_\ell)$ and note that $\mathbb{E}[Y] \geq r\mu_0$.

Our high-level approach is to apply Markov's inequality to the random variable $e^{r\mu_0 - Y}$.

Let λ be an arbitrary parameter in $[0, 1]$ (later, we will set λ to get the best bound). For any single iteration of randomized swap rounding, Chekuri et al. bound the exponential moment of the random variable $\lambda(\mu_0 - f_i(R_\ell))$ as

$$\mathbb{E}[e^{\lambda(\mu_0 - f(R_\ell))}] \leq e^{2\lambda^2 \mu_0}.$$

Since $r\mu_0 - Y = \sum_{\ell=1}^{r} \mu_0 - f(R_\ell)$, we have

$$\begin{aligned}
\mathbb{E}\left[e^{\lambda(r\mu_0 - Y)}\right] &= \mathbb{E}\left[e^{\sum_{\ell=1}^{r} \lambda(\mu_0 - f(R_\ell))}\right] \\
&= \mathbb{E}\left[\prod_{\ell=1}^{r} e^{\lambda(\mu_0 - f(R_\ell))}\right] \\
&= \prod_{\ell=1}^{r} \mathbb{E}\left[e^{\lambda(\mu_0 - f(R_\ell))}\right] \qquad \text{(independence)} \\
&\leq e^{2r\lambda^2 \mu_0}.
\end{aligned}$$

Applying Markov's inequality yields

$$\begin{aligned}
\Pr[r\mu_0 - Y \geq \epsilon r \mu_0] &= \Pr\left[e^{\lambda(r\mu_0 - Y)} \geq e^{\epsilon r \lambda \mu_0}\right] \\
&\leq \frac{\mathbb{E}\left[e^{\lambda(r\mu_0 - Y)}\right]}{e^{\epsilon r \lambda \mu_0}} \\
&\leq e^{2r\lambda^2 \mu_0 - \epsilon r \lambda \mu_0}
\end{aligned}$$

Taking $\lambda = \frac{\epsilon}{4}$, we obtain

$$\Pr\left[\frac{1}{r}\sum_{\ell=1}^{r} f_i(R_\ell) \leq (1-\epsilon)\mu_0\right] = \Pr[r\mu_0 - Y \geq \epsilon r \mu_0] \leq e^{\frac{-r\mu_0 \epsilon^2}{8}}.$$

We now distinguish two cases. First, $\mu_0 < \epsilon$. Since $f_i(R_\ell) \geq 0 \ \forall i, \ell$, $\frac{1}{r}\sum_{\ell=1}^{r} f_i(R_\ell) \geq \mu_0 - \epsilon$ holds with probability 1. Second, $\mu_0 \geq \epsilon$. Here, we see that setting $r = \Theta\left(\frac{1}{\epsilon^3}\left(\log|\mathcal{F}| + \log\frac{1}{\delta}\right)\right)$ ensures that $\frac{1}{r}\sum_{\ell=1}^{r} f_i(R_\ell) \geq (1-\epsilon)\mu_0$ holds with probability at least $1 - \frac{\delta}{|\mathcal{F}|}$. Taking union bound over all $f_i \in \mathcal{F}$ completes the proof.

$\square$

Appendix C

Appendix to Chapter 6

C.1 Proofs for continuous submodular setting

Lemma 34. *Take* $s = \frac{2nM^2}{\epsilon^2} \log \frac{1}{\delta} \log \frac{L_1}{\epsilon}$ *samples and let* $\widehat{CVaR_\alpha}$ *be the empirical CVaR on the samples. Then,* $|CVaR_\alpha(\boldsymbol{x}) - \widehat{CVaR_\alpha}(\boldsymbol{x})| \leq \epsilon$ *holds for all* $\boldsymbol{x} \in \mathcal{P}$ *with probability at least* $1 - \delta$*.*

Proof. We can establish the result for fixed $\boldsymbol{x}$ using the proof of Ohsaka and Yoshida. We have via taking $c = L$ in their Lemma 4.4 that for any fixed $\boldsymbol{x}$, $|\mathrm{CVaR}_\alpha(\boldsymbol{x}) - \widehat{\mathrm{CVaR}_\alpha(\boldsymbol{x})}| \leq \epsilon$ with probability at least $1 - \delta$ by taking $s = \Theta\left(\frac{M^2}{\epsilon^2} \log \frac{1}{\delta}\right)$ samples. Note that we cannot directly take union bound because the set of $x \in \mathcal{P}$ is not finite. Instead, we take a uniform grid of $\left(\frac{L_1 d}{\epsilon}\right)^n$ points containing $\mathcal{P}$. Via union bound, concentration holds for all points in the grid using $s = \Theta\left(\frac{M^2}{\epsilon^2} \log\left(\left(\frac{L_1 d}{\epsilon}\right)^n \frac{1}{\delta}\right)\right) = \Theta\left(\frac{M^2 n}{\epsilon^2} \log \frac{L_1 d}{\epsilon\delta}\right)$. Now we argue that every point in $\mathcal{P}$ is close in CVaR value to a point in the grid. The grid has enough points to guarantee that for any $\boldsymbol{x}_1 \in \mathcal{P}$, there is a point $\boldsymbol{x}_2 \succeq \boldsymbol{x}_1$ within ℓ_2 distance $\frac{\epsilon}{L_2}$ of $\boldsymbol{x}_1$. Note that $\mathrm{CVaR}_\alpha(\boldsymbol{x}_2) \geq \mathrm{CVaR}_\alpha(\boldsymbol{x}_1)$ by monotonicity of F combined with monotonicity of CVaR. Additionally by monotonicity of CVaR, $\mathrm{CVaR}_\alpha(\boldsymbol{x}_2) - \mathrm{CVaR}_\alpha(\boldsymbol{x}_1)$ is maximized when for all y, $F(\boldsymbol{x}_2, y) = F(\boldsymbol{x}_1, y) + L||\boldsymbol{x}_1 - \boldsymbol{x}_2||_2$. In this case, $\{y|F(\boldsymbol{x}_1) \leq \mathrm{VaR}_\alpha(\boldsymbol{x}_1)\} = \{y|F(\boldsymbol{x}_2) \leq \mathrm{VaR}_\alpha(\boldsymbol{x}_2)\}$. Let $\mathcal{Z}$ denote this set of scenarios. We have $\mathrm{CVaR}_\alpha(\boldsymbol{x}_1) = \mathbb{E}\left[F(\boldsymbol{x}_1, y)|y \in \mathcal{Z}\right]$ and $\mathrm{CVaR}_\alpha(\boldsymbol{x}_2) = \mathbb{E}\left[F(\boldsymbol{x}_2, y)|y \in \mathcal{Z}\right]$. But since we take the expectation over a fixed set of scenarios and each $F(\cdot, y)$ is L_1-Lipschitz, the expectation must be L_1-Lipschitz as well.

Hence, $\text{CVaR}_\alpha(\boldsymbol{x}_2) - \text{CVaR}_\alpha(\boldsymbol{x}_1) \leq L_1 ||\boldsymbol{x}_1 - \boldsymbol{x}_2||_2 \leq \epsilon$. □

Lemma 35. *Define $g(\tau) = \sum_{y \in \mathcal{Y}} I_y(\tau)$. (a) τ maximizes $\frac{1}{u}\int_{z=0}^{u} H(\boldsymbol{x}, \tau) dz$ if $g(\tau) = \alpha s$. (b) g is piecewise linear and monotone decreasing.*

Proof. We start with the claim in (a). We have

$$\frac{1}{u}\int_{z=0}^{u} H(\boldsymbol{x}, \tau) dz = \frac{1}{u}\int_{z=0}^{u} \tau - \frac{1}{\alpha s} \sum_{F(x,y) \leq \tau + z} \tau - F(\boldsymbol{x}, y) dz.$$

Note that the function inside the integral is known to be concave in τ (Rockafellar and Urseyev 2000), which yields concavity of the entire function. Thus, to find a maximum it suffices to find a point where the derivative with respect to τ is 0. To this end, note that the set of z such that $F(\boldsymbol{x}, y) = \tau + z$ for some z has measure 0 and hence do not impact the value of the integral. For the remaining values of z, we have

$$\frac{d}{d\tau} = 1 - \frac{|\{y : F(\boldsymbol{x}, y) \leq \tau + z\}|}{\alpha s}$$

since the set in the numerator is constant over some interval around τ. This yields

$$\begin{aligned}
\frac{d}{dt}\frac{1}{u}\int_{z=0}^{u} H(\boldsymbol{x}, \tau) dz &= \frac{1}{u}\int_{z=0}^{u} 1 - \frac{|\{y : F(\boldsymbol{x}, y) \leq \tau + z\}|}{\alpha s} dz \\
&= \frac{1}{u}\left[1 - \frac{1}{\alpha s}\sum_{y \in \mathcal{Y}} \int_{z=0}^{1} \mathbb{1}\left[F(\boldsymbol{x}, y) \leq \tau\right] dz\right] \\
&= \frac{1}{u}\left[1 - \frac{1}{\alpha s}\sum_{y \in \mathcal{Y}} I_y(\tau)\right].
\end{aligned}$$

By inspection, the derivative is 0 when $\sum_{y \in \mathcal{Y}} I_y(\tau) = \alpha s$, which proves part (a) of the lemma.

For part (b), we simply note that each $I_y(\tau)$ is monotone decreasing and piecewise linear in τ, and g is the sum of such functions.

□

Lemma 36. *For any $\boldsymbol{x}$ and τ, $\left|\tilde{H}(\boldsymbol{x}, \tau) - H(\boldsymbol{x}, \tau)\right| \leq \frac{u\left(1+\frac{1}{\alpha}\right)}{2}$*

Proof. We start out by showing that H is $\left(1+\frac{1}{\alpha}\right)-$Lipschitz in τ. Consider any $\boldsymbol{x}$, τ, and τ' and without loss of generality let $\tau' > \tau$.

$$H(\boldsymbol{x},\tau) - H(\boldsymbol{x},\tau') = \left[\tau - \tau'\right] - \frac{1}{\alpha|\mathcal{Y}|}\sum_y \max(\tau - F(\boldsymbol{x},y),0) - \max(\tau' - F(\boldsymbol{x},y),0).$$

We consider three cases for the term inside the summation. First, $F(\boldsymbol{x},y) < \tau$. Here, $\max(\tau - F(\boldsymbol{x},y),0) - \max(\tau' - F(\boldsymbol{x},y),0) = \tau - \tau'$. Second, $\tau \leq F(\boldsymbol{x},y) < \tau'$. Here,

$$\max(\tau - F(\boldsymbol{x},y),0) - \max(\tau' - F(\boldsymbol{x},y),0) = F(\boldsymbol{x},y) - \tau'$$

and hence

$$\left|\max(\tau - F(\boldsymbol{x},y),0) - \max(\tau' - F(\boldsymbol{x},y),0)\right| \leq |\tau - \tau'|.$$

Third, $F(\boldsymbol{x},y) \geq \tau'$. Here, the term in the summation is zero.

Via the triangle inequality, we conclude that

$$\begin{aligned}\left|H(\boldsymbol{x},\tau) - H(\boldsymbol{x},\tau')\right| &\leq \left|\tau - \tau'\right| + \frac{1}{\alpha|\mathcal{Y}|}\sum_y |\tau - \tau'| \\ &\leq \left(1 + \frac{1}{\alpha}\right)|\tau - \tau'|.\end{aligned}$$

Now, since $z \in [0,u]$ holds with probability 1, we can apply the above reasoning to conclude that

$$\begin{aligned}\left|\tilde{H}(\boldsymbol{x},\tau) - H(\boldsymbol{x},\tau)\right| &= \frac{1}{u}\left|\int_{z=0}^{u} H(\boldsymbol{x},\tau+z) - H(\boldsymbol{x},\tau)dz\right| \\ &\leq \frac{1}{u}\int_{z=0}^{u} \left|H(\boldsymbol{x},\tau+z) - H(\boldsymbol{x},\tau)\right| dz \\ &\leq \frac{1}{u}\int_{z=0}^{u} \left(1+\frac{1}{\alpha}\right) z dz \\ &= \frac{u\left(1+\frac{1}{\alpha}\right)}{2}\end{aligned}$$

□

Lemma 37. *At each iteration $k = 1...K$,*

$$\tilde{H}(\tilde{\boldsymbol{x}}^*, \tau(\tilde{\boldsymbol{x}}^*)) - \tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)) \leq \max_{\boldsymbol{v} \in \mathcal{P}} \langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)), \boldsymbol{v} \rangle.$$

Proof. We will show that $\max_\tau \tilde{H}(\cdot, \tau)$ is an up-concave function. Fix any two points $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{P}$. We start out by defining the function $h : [0,1] \to R$ as $h(\xi) = H(\boldsymbol{x}_1 + \xi \boldsymbol{x}_2)$. We will show that $h(\xi, \tau)$ is jointly concave in (ξ, τ). To show joint concavity in (ξ, τ), we write

$$h(\xi, \tau) = \tau - \frac{1}{\alpha |\mathcal{Y}|} \sum_{y \in \mathcal{Y}} [\tau - F(\boldsymbol{x}_1 + \xi \boldsymbol{x}_2, y)]^+$$

The first term is linear in (ξ, τ), and so is concave. We will show that the expectation is jointly convex in (ξ, τ), from which concavity of h follows. To show this, is suffices to show that the term inside the expectation is convex for any fixed y. Note that this term is the composition of the function $(\xi, \tau) \mapsto \tau - F(\boldsymbol{x}_1 + \xi \boldsymbol{x}_2, y)$ with the function $t \mapsto [t]^+$. Since the latter is a nondecreasing convex function, the composition is convex whenever the inner function is convex. For the inner function, τ is convex because it is linear in ξ, and $-F(\boldsymbol{x}_1 + \xi \boldsymbol{x}_2, y)$ is convex because $F(\cdot, y)$ is up-concave. Thus, the claim follows.

Now define $\tilde{h}(\xi, \tau) = \tilde{H}(\boldsymbol{x}_1 + \xi \boldsymbol{x}_2)$. $\tilde{h}$ is jointly concave in (ξ, τ) because it is a non-negative linear combination of concave functions. This shows that $\max_\tau \tilde{h}(\xi, \tau)$ is concave in ξ because maximizing a jointly concave function with respect to one of its parameters yields a concave function in the remaining parameters. We conclude that $\max_\tau \tilde{H}(\cdot, \tau)$ is an up-concave function.

Fix the points $\boldsymbol{x}^k$ and $\tilde{\boldsymbol{x}}^*$ as $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ in the above definition of $\tilde{h}$. Now the conclusion follows by arguing (as in Bian et al. (2017)),

$$\begin{aligned}\tilde{H}(\tilde{\boldsymbol{x}}^*, \tau(\tilde{\boldsymbol{x}}^*)) - \tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)) &\leq \tilde{H}(\tilde{\boldsymbol{x}}^* + \boldsymbol{x}^k, \tau(\tilde{\boldsymbol{x}}^* + \boldsymbol{x}^k)) - \tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)) \\ &= \max_\tau \tilde{h}(1, \tau) - \max_\tau \tilde{h}(0, \tau)\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{d \max_\tau \tilde{h}(0,\tau)}{d\xi} \cdot 1 \text{ (since } \max_\tau \tilde{h}(\cdot,\tau)) \text{ is concave} \\
&= \langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)), \tilde{\boldsymbol{x}}^* \rangle \\
&\leq \max_{\boldsymbol{v} \in \mathcal{P}} \langle \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}^k, \tau(\boldsymbol{x}^k)), \boldsymbol{v} \rangle
\end{aligned}
$$

□

In the lemmas that follow, we use an alternate interpretation of $\tilde{H}$. Namely, we can view the smoothing process as drawing a random variable z from a uniform distribution over the interval $[0, u]$. Then, $\tilde{H}(\boldsymbol{x}, \tau) = \mathbb{E}_z [H(\boldsymbol{x}, \tau + z)]$. This is completely equivalent to the definition given in the text, but simplifies notation and concepts at a few places in the proofs below.

Lemma 38. *If $\boldsymbol{x}_2 \succeq \boldsymbol{x}_1$, $\nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}_2, \tau(\boldsymbol{x}_2)) \succeq \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}_2, \tau(\boldsymbol{x}_1))$.*

Proof. Recall that $\nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}_2, \tau) = \mathbb{E}_z [\nabla_{\boldsymbol{x}} H(\boldsymbol{x}_2, \tau + z)]$. We couple the random variables $\nabla_{\boldsymbol{x}} H(\boldsymbol{x}_2, \tau(\boldsymbol{x}_1) + z)$ and $\nabla_{\boldsymbol{x}} H(\boldsymbol{x}_2, \tau(\boldsymbol{x}_2) + z)$ by fixing z to any value where both derivatives exist (which excludes only a measure 0 set).

Note that since F is monotone in $\boldsymbol{x}$, $\nabla_{\boldsymbol{x}} F(\boldsymbol{x}, y) \succeq 0$ holds for all $\boldsymbol{x} \in \mathcal{P}$ and $y \in \mathcal{Y}$. Moreover, we can write

$$
\nabla_{\boldsymbol{x}} H(\boldsymbol{x}, \tau) = \frac{1}{\alpha |\mathcal{Y}|} \sum_{y \in \mathcal{Y} : F(\boldsymbol{x}, y) \leq \tau} \nabla_{\boldsymbol{x}} F(\boldsymbol{x}, y).
$$

It is easy to see that the function $\tau(\boldsymbol{x})$ is monotone nondecreasing and hence $\tau(\boldsymbol{x}_2) + z \geq \tau(\boldsymbol{x}_1) + z$. Thus, for all $y \in \mathcal{Y}$, $F(\boldsymbol{x}_2) \leq \tau(\boldsymbol{x}_2) + z$ only if $F(\boldsymbol{x}_2) \leq \tau(\boldsymbol{x}_1) + z$. Since each term in the above summation is nonnegative, $\nabla_{\boldsymbol{x}} H(\boldsymbol{x}_2, \tau(\boldsymbol{x}_1) + z) \succeq \nabla_{\boldsymbol{x}} H(\boldsymbol{x}_2, \tau(\boldsymbol{x}_2) + z)$. The lemma now follows by taking the expectation with respect to z. □

Lemma 39. *If $\forall y \in \mathcal{Y}$, $F(\cdot, y)$ is L_1-Lipschitz and $\nabla_{\boldsymbol{x}} F(\cdot, y)$ is L_2 Lipschitz with $||\nabla_{\boldsymbol{x}} F||_2 \leq G$, then $\nabla_{\boldsymbol{x}} \tilde{H}$ is $\frac{1}{\alpha}\left(L_2 + \frac{L_1 G}{u}\right)$ −Lipschitz.*

Proof. For any z, let $\mathcal{Y}_1(z) = \{y : F(\boldsymbol{x}_1, y) \leq F(\boldsymbol{x}_2, y) \leq \tau + z\}$. Let $\mathcal{Y}_2(z) = \{y : F(\boldsymbol{x}_1, y) \leq \tau + z < F(\boldsymbol{x}_2, y)\}$. We have

$$
\begin{aligned}
&||\nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}_1, \tau) - \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}_2, \tau)|| = \left|\left| \mathop{\mathbb{E}}_{z} \left[\nabla_{\boldsymbol{x}} H(\boldsymbol{x}_1, \tau + z)\right] - \mathop{\mathbb{E}}_{z} \left[\nabla_{\boldsymbol{x}} H(\boldsymbol{x}_2, \tau + z)\right] \right|\right| \\
&= \frac{1}{\alpha} \left|\left| \mathop{\mathbb{E}}_{z} \left[\frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}_1(z)} \nabla_{\boldsymbol{x}} F(\boldsymbol{x}_1, y) - \nabla_{\boldsymbol{x}} F(\boldsymbol{x}_2, y) - \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}_2(z)} \nabla_{\boldsymbol{x}} F(\boldsymbol{x}_1, y) \right] \right|\right| \\
&\leq \frac{1}{\alpha} \mathop{\mathbb{E}}_{z} \left[\frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}_1(z)} ||\nabla_{\boldsymbol{x}} F(\boldsymbol{x}_1, y) - \nabla_{\boldsymbol{x}} F(\boldsymbol{x}_2, y)|| + \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}_2(z)} ||\nabla_{\boldsymbol{x}} F(\boldsymbol{x}_1, y)|| \right] \\
&\leq \frac{1}{\alpha} \mathop{\mathbb{E}}_{z} \left[\frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}_1(z)} L_2 \, ||\boldsymbol{x}_1 - \boldsymbol{x}_2|| + \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}_2(z)} ||\nabla_{\boldsymbol{x}} F(\boldsymbol{x}_1, y)|| \right] \\
&\leq \frac{1}{\alpha} \mathop{\mathbb{E}}_{z} \left[\frac{|\mathcal{Y}_1(z)|}{|\mathcal{Y}|} \right] L_2 ||\boldsymbol{x}_2 - \boldsymbol{x}_1|| + \frac{1}{\alpha} \mathop{\mathbb{E}}_{z} \left[\frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}_2(z)} ||\nabla_{\boldsymbol{x}} F(\boldsymbol{x}_1, y)|| \right] \\
&\leq \frac{1}{\alpha} \mathop{\mathbb{E}}_{z} \left[\frac{|\mathcal{Y}_1(z)|}{|\mathcal{Y}|} \right] L_2 ||\boldsymbol{x}_2 - \boldsymbol{x}_1|| + \frac{1}{\alpha} \mathop{\mathbb{E}}_{z} \left[\frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}_2(z)} G \right] \\
&= \frac{1}{\alpha} \mathop{\mathbb{E}}_{z} \left[\frac{|\mathcal{Y}_1(z)|}{|\mathcal{Y}|} \right] L_2 ||\boldsymbol{x}_2 - \boldsymbol{x}_1|| + \frac{1}{\alpha} \mathop{\mathbb{E}}_{z} \left[\frac{G}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} 1 \left[y \in \mathcal{Y}_2(z)\right] \right] \\
&= \frac{1}{\alpha} \mathop{\mathbb{E}}_{z} \left[\frac{|\mathcal{Y}_1(z)|}{|\mathcal{Y}|} \right] L_2 ||\boldsymbol{x}_2 - \boldsymbol{x}_1|| + \frac{G}{\alpha |\mathcal{Y}|} \sum_{y \in \mathcal{Y}} \Pr_z \left[y \in \mathcal{Y}_2(z)\right]
\end{aligned}
$$

Now, all that remains is to bound the term $\Pr_z \left[y \in \mathcal{Y}_2(z)\right]$. Note that for all $y \in \mathcal{Y}$, $F(\boldsymbol{x}_2, y) \leq F(\boldsymbol{x}_1, y) + L_1 ||\boldsymbol{x}_1 - \boldsymbol{x}_2||$. Since z follows a uniform distribution over an interval of size u, the probability that it falls into an interval of size $L_1 ||\boldsymbol{x}_1 - \boldsymbol{x}_2||$ is at most $\frac{L_1 ||\boldsymbol{x}_1 - \boldsymbol{x}_2||}{u}$ and we conclude that

$$
||\nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}_1, \tau) - \nabla_{\boldsymbol{x}} \tilde{H}(\boldsymbol{x}_2, \tau)|| \leq \frac{1}{\alpha} \left(L_2 + \frac{L_1 G}{u} \right) ||\boldsymbol{x}_2 - \boldsymbol{x}_1||.
$$

□

C.2 Proofs for discrete portfolio optimization

We recall the problem setting for discrete submodular functions, which we refer to as the *discrete portfolio optimization* problem. We are given a collection of submodular set functions $f(\cdot, y)$ on a ground set X, where y is a random variable. There is a collection of feasible sets $\mathcal{I}$. For instance, $\mathcal{I}$ could be all size-k subsets. In general, we will focus on the setting where $\mathcal{I}$ forms a matroid. Analogously to the continuous setting, we assume that f is bounded: $\max_{y, S\in\mathcal{I}} f(S, y) \leq M$ for some $M > 0$. The algorithm selects a distribution q over the sets in $\mathcal{I}$. Let $\Delta(\mathcal{I})$ be the set of all such distributions (the $|\mathcal{I}|$-dimensional simplex). We aim to solve the problem

$$\max_{q\in\Delta(\mathcal{I})} \text{CVaR}_\alpha \left(\sum_{S\in\mathcal{I}} q_S f(S, y)\right)$$

In this section, we provide a block-box reduction from the above problem to the continuous submodular CVaR problem studied so far. We start by introducing a few useful concepts from submodular optimization in order to formulate our proposed algorithm. We then state the algorithm and prove its approximation guarantee. We will assume throughout that the number of scenarios y is at most the value s given in Lemma 1 since this suffices to obtain ϵ-accurate solutions to the true CVaR problem.

Multilinear extension: For a given a submodular function f, its multilinear relaxation F is a function defined over the continuous space $[0, 1]^{|X|}$. For any $\boldsymbol{x} \in [0, 1]^{|X|}$, let $p_{\boldsymbol{x}}$ denote the product distribution with marginals given by $\boldsymbol{x}$. We have $F(\boldsymbol{x}) = \mathbb{E}_{S\sim p_{\boldsymbol{x}}}[f(S)]$. Note that F agrees with f at each vertex of the hypercube, the points $\{0, 1\}^{|X|}$ (where we interpret each binary vector as the indicator vector of a set). The value of F, as well as its gradients, can be efficiently computed via random sampling [CCPV11], with closed forms known for common special cases [IJB14]. Here, we ignore such issues and assume that F and ∇F are available exactly (since evaluation up to arbitrary precision ϵ can be done via sampling). For any submodular function f, F is a continuous submodular function. Moreoever, F is smooth (in the sense of having Lipschitz gradients of bounded norm in terms of M; see [Wil18a]).

Swap rounding: Let $\mathcal{P}$ be the convex hull of the indicator vectors of sets in $\mathcal{I}$. Each point in $\boldsymbol{x} \in \mathcal{P}$ specifies a product distribution, and via the multilinear extension we can optimize over such distributions. However, we need to convert this product distribution back to a distribution over sets $\mathcal{I}$. Note that just sampling from $p_{\boldsymbol{x}}$ is not guaranteed to give us sets that are feasible (lie in $\mathcal{I}$). For instance, if $\mathcal{I}$ is the k-uniform matroid, sampling from a a product distribution $\boldsymbol{x} \in \mathcal{P}$ can easily produce sets with more than k elements even though $\sum_{i=1}^n x_i \leq k$. Whenever $\mathcal{I}$ is a matroid, the swap rounding algorithm of Chekuri et al. gives a means for efficiently rounding a point $\boldsymbol{x} \in \mathcal{P}$ to a single feasible set $\boldsymbol{x} \in \mathcal{P}$ in a randomized fashion. The set S satisfies $\mathbb{E}[f(S)] \geq F(\boldsymbol{x})$ for any submodular function f, and also satisfies a lower tail bound which controls the probability that $f(S)$ is significantly less than $F(\boldsymbol{x})$. Wilder [Wil18a] leverage this result to show the following:

Lemma 40. *Suppose we draw $O\left(\frac{\log \frac{N}{\delta}}{\epsilon^3}\right)$ independent samples $S_1 ... S_\ell$ via swap rounding. Then $\frac{1}{\ell}\sum_{i=1}^{\ell} f(S_i) \geq F(\boldsymbol{x}) - \epsilon$ holds for any N submodular functions and their multilinear extensions with probability at least $1-\delta$.*

Combining these two ingredients yields the following algorithm for discrete portfolio optimization:

Algorithm 15 PortfolioCVaR

```
Input: an algorithm $\mathcal{A}$ for continuous submodular CVaR maximization.
Set $r = O\left(\frac{M^2 \log s}{\epsilon^2}\right)$
Use $\mathcal{A}$ to solve the problem
    $$\max_{\boldsymbol{x}^1 ... \boldsymbol{x}^r \in \times_{i=1}^r \mathcal{P}} \text{CVaR}_\alpha \left(\frac{1}{r}\sum_{i=1}^r \mathop{\mathbb{E}}_{S \sim p_{\boldsymbol{x}^i}} [f(S, y)]\right)$$
    obtaining a solution $\boldsymbol{x}^1 ... \boldsymbol{x}^r$.
Set $\ell = O\left(\frac{\log \frac{sr}{\delta}}{\epsilon^3}\right)$
for $i = 1...r$ do
    Draw sets $S_1^i ... S_\ell^i$ independently as SwapRound($\boldsymbol{x}^i$).
end for
Return the uniform distribution on $\{S_j^i : i = 1...r, j = 1...\ell\}$.
```

This algorithm solves a continuous optimization problem using the multilinear extensions $F(\cdot, y)$. We maintain r copies of the decision variables $\boldsymbol{x}^1 ... \boldsymbol{x}^r$, a choice which is justified later (we remark that similar ideas have been used by [DX17, Wil18a] in other domains). We then draw a series of samples via swap rounding for each $\boldsymbol{x}^i$ and output the uniform distribution over the combined set of samples.

Theorem 26. *Suppose that we have access to an algorithm $\mathcal{A}$ for the continuous submodular CVaR maximization problem, which returns a solution with value at least $\alpha OPT - \epsilon$ for some $\alpha > 0$ and any $\epsilon > 0$ in time poly$(M, \frac{1}{\epsilon}, n)$. Then, for any discrete portfolio optimization problem over a matroid,* PortfolioCVaR *returns a solution with value at least $\alpha OPT - \epsilon$ with probability at least $1 - \delta$ in time poly$(M, \frac{1}{\epsilon}, n, \log \frac{1}{\delta})$.*

Remark 1. *We intentionally present the runtime at a high level (polynomial time) because our reduction is not aimed at getting the optimal approximation ratio, not the tightest possible runtime bound. The major bottleneck is that we call $\mathcal{A}$ with $\tilde{O}\left(\frac{M^2}{\epsilon^2}\right)$ copies of the decision variables. Below, we also show how a simple modification of the algorithm (operating on only n decision variables) obtains approximation ratio $\alpha(1 - \frac{1}{e})$.*

Remark 2. *The RASCAL algorithm that we introduce for the continuous setting provides an algorithm $\mathcal{A}$ which may be used in this reduction with $\alpha = \left(1 - \frac{1}{e}\right)$. Hence, we immediately obtain a $\left(1 - \frac{1}{e}\right)$-approximate algorithm for the discrete portfolio optimization problem.*

Proof. Define $OPT = \max_{p \in \Delta(\mathcal{I})} \text{CVaR}_\alpha \left(\mathbb{E}_{S \sim p}\left[f(S, y)\right]\right)$ and p be the maximizing distribution. We first claim

Lemma 41. *There is a distribution q with support on at most r sets which satisfies*

$$CVaR_\alpha \left(\mathop{\mathbb{E}}_{S \sim q}[f(S, y)]\right) \geq OPT - \epsilon$$

Proof. A stronger result is established in Wilder et al. [Wil18a]. Their Lemma 4 shows that there is a uniform distribution q on r sets which satisfies $\mathbb{E}_{S \sim q}[f(S, y)] \geq \mathbb{E}_{S \sim p}[f(S, y)] - \epsilon$

for all y. It is easy to check that since q has value ϵ-close to p for each y individually, it is also ϵ-close in aggregate CVaR value. □

In what follows, we will let q be such a distribution. Since optimizing over such bounded-support mixture distributions is computationally difficult, we now expand our search to a convex set which contains the promised good distribution q. Specifically, we optimize over the set of all distributions which are mixtures of r product distributions. We will maintain r decision variables $\boldsymbol{x}^1 ... \boldsymbol{x}^r \in \times_{i=1}^r \mathcal{P}$. We will consider the problem of maximizing $\text{CVaR}_\alpha \left(\frac{1}{r}\sum_{i=1}^r \mathbb{E}_{S\sim p_{\boldsymbol{x}^i}}\left[f(S,y)\right]\right)$. This is the CVaR of the mixture distribution on the r product distributions $\boldsymbol{x}^1 ... \boldsymbol{x}^r$. Note that the distribution q is an instance of such a distribution where $\boldsymbol{x}^i$ is the indicator vector for the set S_i (i.e., a product distribution which deterministically returns S_i) . Hence, by the above guarantee for q we have:

$$\max_{\boldsymbol{x}^1 ... \boldsymbol{x}^r \in \times_{i=1}^r \mathcal{P}} \text{CVaR}_\alpha \left(\frac{1}{r}\sum_{i=1}^r \mathop{\mathbb{E}}_{S\sim p_{\boldsymbol{x}^i}}\left[f(S,y)\right]\right) \geq \text{CVaR}_\alpha \left(\mathop{\mathbb{E}}_{S\sim q}\left[f(S,y)\right]\right) \geq OPT - \epsilon.$$

Thus, we can solve the first maximization problem to obtain a solution with value ϵ-close to the optimum. Here we use the multilinear extension. Note that in the optimization problem above, $\mathbb{E}_{S\sim p_{\boldsymbol{x}^i}}\left[f(S,y)\right]$ is exactly $F(\boldsymbol{x}^i, y)$, and $\frac{1}{r}\sum_{i=1}^r F(\boldsymbol{x}^i, y)$ is DR-submodular since it is a convex combination of DR-submodular functions. Hence, we can obtain a solution to the optimization problem with value at least $\alpha OPT - \epsilon$ by applying $\mathcal{A}$, the promised algorithm for continuous submodular CVaR optimization (line 3 of PortfolioCVaR). The only remaining issue is to convert the resulting mixture of product distributions into a distribution over elements of $\mathcal{I}$ with equivalent solution quality. Here, we use the swap rounding procedure. Via Lemma 40, it suffices to draw $O\left(\frac{\log \frac{sr}{\delta}}{\epsilon^3}\right)$ samples for each of the $\boldsymbol{x}^i$. Let the uniform distribution on the combined set of samples be q'. Via union bound over the r different product distributions, we have that $\mathbb{E}_{S\sim q'}[f(S,y)] \geq \frac{1}{r}\sum_{i=1}^r F(\boldsymbol{x}^i) - \epsilon$ holds for each $y \in \mathcal{Y}$ with probability at least $1-\delta$. Conditioning on this event, we have that

$$
\begin{aligned}
\mathrm{CVaR}_\alpha\left(\mathop{\mathbb{E}}_{S\sim q'}[f(S,y)]\right) &\geq \mathrm{CVaR}_\alpha\left(\frac{1}{r}\sum_{i=1}^{r} F(\boldsymbol{x}^i, y)\right) - \epsilon \\
&\geq \alpha OPT - 2\epsilon
\end{aligned}
$$

and now it suffices to apply the above argument with $\epsilon' = \frac{\epsilon}{2}$ to obtain the theorem. □

We close by noting that it is possible to obtain a more computationally efficient algorithm by maintaining only a single copy of the decision variables ($r = 1$), at the cost of a factor $(1 - \frac{1}{e})$ in the approximation ratio:

Theorem 27. *Setting $r = 1$,* PortfolioCVaR *returns a solution with value at least*

$$\alpha\left(1 - \frac{1}{e}\right) OPT - \epsilon$$

for any discrete portfolio optimization problem.

Proof. The reason that we needed to introduce many copies is to guarantee that our feasible set includes a distribution which is ϵ-close in CVaR value to the optimal distribution p. However, we can use a known result for submodular functions to guarantee that the feasible set includes a $(1 - \frac{1}{e})$-approximate solution even if we take $r = 1$. Specifically, we use the *correlation gap* result of Agrawal et al. [ADSY10]. Let $D(\boldsymbol{x})$ be the set of all distributions with marginals $\boldsymbol{x}$. For any submodular function f, Agrawal et al. [ADSY10] prove that

$$\max_{\boldsymbol{x}\in\mathcal{P}, q\in D(\boldsymbol{x})} \frac{\mathbb{E}_{S\sim q}[f(S)]}{\mathbb{E}_{S\sim p_{\boldsymbol{x}}}[f(S)]} \leq \frac{e}{e-1}$$

Hence, we can guarantee that

$$\max_{\boldsymbol{x}\in\mathcal{P}} \mathrm{CVaR}_\alpha\left(\mathop{\mathbb{E}}_{S\sim p_{\boldsymbol{x}}}[f(S,y)]\right) \geq \left(1 - \frac{1}{e}\right) OPT$$

and from here, the same argument as before shows that applying PortfolioCVaR with $r = 1$ results in a distribution with value at least $\alpha\left(1 - \frac{1}{e}\right) OPT - \epsilon$. □

Appendix D

Appendix to Chapter 5

D.1 Proofs

D.1.1 DR-submodularity

In order to use the Frank-Wolfe algorithm, we must verify that the objective function is *diminishing returns* (DR) submodular. While every submodular set function is also diminishing returns, this is an additional property which must be separately verified for continuous function. Since we have already showed that G is submodular, showing that it is DR-submodular amounts to verifying that $\frac{\partial^2 G}{\partial \nu_i^2} \leq 0$ always holds (Bian et al. 2017). We again focus on a single term in the posynomial expression, $\prod_{i=1}^{n}(1-\nu_i)^{p_{ij}}$. If $p_{ij} \leq 1$, we are done. Assume that $p_{ij} > 1$. Taking derivatives yields

$$\frac{\partial^2}{\partial \nu_i^2}\left[\prod_{i=1}^{n}(1-\nu_i)^{p_{ij}}\right] = p_{ij}(p_{ij}-1)(1-\nu_i)^{p_{ij}-2}\prod_{k\neq i}(1-\nu_k)^{p_{kj}}.$$

Each term in this expression is nonnegative. We conclude that $\frac{\partial^2 F}{\partial \nu_i^2} \geq 0$ and hence $\frac{\partial^2 G}{\partial \nu_i^2} \leq 0$.

D.1.2 Deterministic case

We now prove the approximation guarantee for the Frank-Wolfe algorithm applied to Problem 1 (the deterministic problem).

Proof of Theorem 1. The major step is to show that ∇G is Lipschitz. Since the ℓ_∞ and ℓ_i norms are dual, it suffices to bound $||\nabla^2 H||_\infty$ to show that ∇G is Lipschitz with respect to the ℓ_1 norm. A naive bound would depend on the sizes of the coefficients in the objective, e.g., the population size and so on. We can get around this by considering (purely for the sake of analysis) the rescaled function $\frac{G(\nu)}{|G(\nu^*)|}$. Since $\frac{1}{|G(\nu^*)|}$ is a nonnegative constant with respect to ν, maximizing the two functions is equivalent. Note that if we have a guarantee of the form $\frac{G(\nu)}{|G(\nu^*)|} \geq \left(1 - \frac{1}{e}\right) \frac{G(\nu^*)}{|G(\nu^*)|} - \epsilon$ then this implies that $G(\nu) \geq \left(1 - \frac{1}{e} - \epsilon\right) G(\nu^*)$. That is, we get a multiplicative guarantee with respect to the unscaled function. We now consider a single element of $\nabla^2 G(\nu)$ using the posynomial representation of G. We emphasize that this representation is used purely for analysis; it does not need to be known or computed by the algorithm.

$$\begin{aligned}
\left|\frac{\partial^2 G(\nu)}{\partial \nu_i^2}\right| &= \sum_j a_j p_{ij}(p_{ij} - 1)(1 - \nu_i)^{-2} \prod_k (1 - \nu_k)^{p_{kj}} \\
&\leq \left(\frac{T}{1 - U_i}\right)^2 \sum_j a_j \prod_k (1 - \nu_k)^{p_{kj}} \\
&\leq \left(\frac{T}{1 - U_i}\right)^2 |G(\nu^*)|
\end{aligned}$$

From which we conclude that $\left|\left|\nabla^2 \frac{G(\nu)}{|G(\nu^*)|}\right|\right|_\infty \leq \left(\frac{T}{1 - U_{max}}\right)^2$, where $U_{max} = \max_i U_i$. In order to apply the result of Bian et al. we actually need a bound on the Lipschitz constant of the single-dimensional auxiliary function $g_{\nu,y}(\delta) = G(\nu + \delta y)$ for any feasible ν, feasible y, and $\delta \geq 0$. Note that given any δ_1, δ_2, we have $||(\nu + \delta_1 y) - (\nu + \delta_2 y)||_1 = |\delta_1 - \delta_2| \cdot ||y||_1$. From this and the Lipschitz bound on G, we obtain that

$$\begin{aligned}
|g_{\nu,y}(\delta_1) - g_{\nu,y}(\delta_2)| &\leq |\delta_1 - \delta_2| \left(\frac{T}{1 - U_{max}}\right)^2 ||y||_1 \\
&\leq |\delta_1 - \delta_2| \left(\frac{T}{1 - U_{max}}\right)^2 K
\end{aligned}$$

So we have that for any ν and y, $g_{\nu,y}$ is $K\left(\frac{T}{1 - U_{max}}\right)^2$-Lipschitz. Corollary 1 of Bian et al. now implies that by taking $\frac{K}{2\epsilon}\left(\frac{T}{1 - U_{max}}\right)^2$ iterations in the Frank-Wolfe algorithm the guarantee in the theorem follows. □

D.1.3 Stochastic case

We now prove our approximation guarantee for the stochastic setting. We start out by establishing a useful smoothness property for G, which states that G is close to its linearization over small step sizes. To do so, we include for completeness the following technical relation between an ℓ_∞ bound on a function's norm and Lipschitz smoothness in the ℓ_1 norm.

Lemma 42. *Consider a differentiable function $f : R^n \to R^n$. If $||\nabla f(x)||_\infty \leq L \, \forall x \in \mathcal{P}$, $||f(y) - f(x)||_1 \leq L||y - x||_1 \, \forall x, y \in \mathcal{P}$.*

Proof. Fix any $x, y \in \mathcal{P}$. Define the auxiliary function $g(\delta) = f(x + \delta(y - x))$ We have $f(y) - f(x) = g(1) - g(0)$ and hence

$$\begin{aligned}
||f(y) - f(x)||_1 &= ||g(1) - g(0)||_1 \\
&= \left|\left| \int_0^1 \frac{dg(\delta)}{d\delta} \, d\delta \right|\right|_1 \\
&\leq \int_0^1 \left|\left| \frac{dg(\delta)}{d\delta} \right|\right|_1 d\delta \\
&= \int_0^1 \left|\left| \nabla f(x + \delta(y - x))^\top (y - x) \right|\right|_1 d\delta \\
&\leq \int_0^1 ||\nabla f(x + \delta(y - x))^\top||_\infty ||y - x||_1 \text{ (Hölder's inequality)} \\
&\leq L||y - x||_1
\end{aligned}$$

□

Lemma 43. *Suppose that G has an L-Lipschitz gradient in the ℓ_1 norm. Let $d = \max_{x \in \mathcal{P}} ||x||_\infty ||x||_1$. For any x, y, $G(x + \gamma y) - G(x) \geq \gamma \nabla G(x)^T y - \frac{Ld\gamma^2}{2}$.*

Proof. For any $x, y \in \mathcal{P}$, we consider the one dimensional auxiliary function $g_{x,y}(\delta) = G(x + \delta y)$. We can show that g has a gradient which is Ld-Lipschitz:

$$
\begin{aligned}
\frac{dg(\delta_1)}{d\delta} - \frac{dg(\delta_2)}{d\delta} &= \nabla G(x+\delta_1 y)^\top y - \nabla G(x+\delta_2 y)^\top y \\
&= \left(\nabla G(x+\delta_1 y) - \nabla G(x+\delta_2 y)\right)^\top y \\
&\leq ||\nabla G(x+\delta_1 y) - \nabla G(x+\delta_2 y)||_1 ||y||_\infty \text{ (by Hölder's inequality)} \\
&\leq L||(x+\delta_1 y) - (x+\delta_2)y||_1 ||y||_\infty \\
&\leq L|\delta_1 - \delta_2| \cdot ||y||_1 ||y||_\infty \\
&= Ld|\delta_1 - \delta_2|.
\end{aligned}
$$

Now, we use smoothness of g to establish that G is close to its linearization over short distances:

$$
\begin{aligned}
G(x+\gamma y) - G(x) - \nabla G(x)^T y &= g(\gamma) - g(0) - \frac{dg(0)}{d\delta} \cdot 1 \\
&= \int_{\delta=0}^{\gamma} \left[\frac{dg(\delta)}{d\delta} - \frac{dg(0)}{d\delta}\right] d\delta \\
&\leq \int_{\delta=0}^{\gamma} Ld\delta \, d\delta \\
&= \frac{Ld\gamma^2}{2}.
\end{aligned}
$$

which proves the lemma. □

We also use the following lemma, the proof of which can be found in Bian et al. (2017):

Lemma 44. *For any DR-submodular function G and its optimizer ν^*, $G(\nu^* + \nu) - G(\nu) \leq \nabla G(\nu)^\top \nu^*$.*

Using these lemmas, we prove the following general result for any smooth DR-submodular function:

Theorem 28. *Let G be a DR-submodular function which is C-Lipschitz in the ℓ_1 norm, with L-Lipschitz gradient (also in ℓ_1 norm) and $G(0) = 0$. Let $d = \max_{x \in \mathcal{P}} ||x||_\infty ||x||_1$ and $b = \max_{x \in \mathcal{P}} ||x||_1$. Then, running DOMO for $R = \frac{Ld}{\epsilon}$ iterations with $m = \left(\frac{4bC}{\epsilon}\right)^2$ samples per iteration yields a feasible $\nu \in \mathcal{P}$ which satisfies $\mathbb{E}[G(\nu)] \geq (1 - \frac{1}{e})G(\nu^*) - \epsilon$.*

Proof of Theorem 2. We consider a DR-submodular function with L-Lipschitz gradient. Further, we assume that G itself is C-Lipschitz. We analyze the gain made in a single step as follows:

$$
\begin{aligned}
&G(\nu^k) - G(\nu^{k-1}) \geq \frac{1}{R}\nabla G(\nu^{k-1})^\top y^k - \frac{Ld}{2R^2} \text{ (Lemma 43)}\\
&= \frac{1}{R}\widehat{\nabla}_k^\top y^k - \frac{1}{R}\left(\widehat{\nabla}_k - \nabla G(\nu^{k-1})\right)^\top y^k - \frac{Ld}{2R^2}\\
&\geq \frac{1}{R}\widehat{\nabla}_k^\top y^k - \frac{b}{R}||\widehat{\nabla}_k - \nabla G(\nu^{k-1})||_\infty - \frac{Ld}{2R^2} \text{ (by Hölder's inequality and } ||y||_1 \leq b)\\
&\geq \frac{1}{R}\widehat{\nabla}_k^\top \nu^* - \frac{b}{R}||\widehat{\nabla}_k - \nabla G(\nu^{k-1})||_\infty - \frac{Ld}{2R^2} \text{ (by definition of } y^k)\\
&= \frac{1}{R}\nabla G(\nu^{k-1})^\top \nu^* - \frac{1}{R}\left(G(\nu^{k-1}) - \widehat{\nabla}_k\right)^\top \nu^* - \frac{b}{R}||\widehat{\nabla}_k - \nabla G(\nu^{k-1})||_\infty - \frac{Ld}{2R^2}\\
&\geq \frac{1}{R}\nabla G(\nu^{k-1})^\top \nu^* - \frac{2b}{R}||\widehat{\nabla}_k - \nabla G(\nu^{k-1})||_\infty - \frac{Ld}{2R^2}\\
&\geq \frac{1}{R}\left(G(\nu^* + \nu^{k-1}) - G(\nu^{k-1})\right) - \frac{2b}{R}||\widehat{\nabla}_k - \nabla G(\nu^{k-1})||_\infty - \frac{Ld}{2R^2} \text{ (Lemma 44)}\\
&\geq \frac{1}{R}\left(G(\nu^*) - G(\nu^{k-1})\right) - \frac{2b}{R}||\widehat{\nabla}_k - \nabla G(\nu^{k-1})||_\infty - \frac{Ld}{2R^2} \text{ (monotonicity)}
\end{aligned}
$$

By assumption that G is C-Lipschitz, $||\nabla G||_\infty \leq C$. Via Jensen's inequality, we have

$$
\begin{aligned}
\mathbb{E}\left[||\widehat{\nabla}_k - \nabla G(\nu^{k-1})||_\infty\right] &\leq \sqrt{\mathbb{E}\left[||\widehat{\nabla}_k - \nabla G(\nu^{k-1})||_\infty^2\right]}\\
&\leq \frac{C}{\sqrt{m}}
\end{aligned}
$$

where the last step uses that averaging over m independent samples reduces the variance of $\widehat{\nabla}_k$ by a factor of m. Hence we have

$$
\mathbb{E}\left[G(\nu^*) - G(\nu^k)\right] \leq \left(1 - \frac{1}{R}\right)\mathbb{E}\left[G(\nu^*) - G(\nu^{k-1})\right] - \frac{2bC}{R\sqrt{m}} - \frac{Ld}{2R^2}.
$$

and so

$$\begin{aligned}\mathbb{E}\left[G(\nu^*) - G(\nu^R)\right] &\leq \left(1 - \frac{1}{R}\right)^R \mathbb{E}\left[G(\nu^*) - G(\nu^0)\right] - \sum_{k=0}^{R-1} \frac{2bC}{R\sqrt{m}} - \sum_{k=0}^{R-1} \frac{Ld}{2R^2} \\ &\leq \left(1 - \frac{1}{R}\right)^R \mathbb{E}\left[G(\nu^*) - G(\nu^0)\right] - \frac{2bC}{\sqrt{m}} - \frac{Ld}{2R}\end{aligned}$$

and hence

$$\mathbb{E}\left[G(\nu^*) - G(\nu^R)\right] \leq \frac{1}{e}\left[G(\nu^*) - G(0)\right] - \frac{2bC}{\sqrt{m}} - \frac{Ld}{2R}.$$

Choosing $m = \left(\frac{4bC}{\epsilon}\right)^2$ and $R = \frac{Ld}{\epsilon}$ completes the proof. □

Now, the result for our problem follows by noting the appropriate values for the Lipschitz constant and size of the feasible set. Note that the stochastic objective $H := \mathbb{E}[G(\cdot, \xi)]$ inherits the smoothness properties of the deterministic objective since it is a convex combination of such functions. Using the same reasoning as the deterministic case, the rescaled objective $\frac{H}{H(\nu^*)}$ satisfies $\left|\left|\nabla \frac{H}{H(\nu^*)}\right|\right|_\infty \leq \frac{T}{(1-U_{max})}$ and $\left|\left|\nabla^2 \frac{H}{H(\nu^*)}\right|\right|_\infty \leq \left(\frac{T}{1-U_{max}}\right)^2$. Via Lemma 42, this yields immediate bounds for the Lipschitz constants C and L. Moreover, $d = b = K$. Plugging these values into Theorem 28 yields the result.

D.2 Experiments

We now provide additional detail about the data sources used in the experiments.

D.2.1 TB

- Annual death probabilities μ: Calculated from WHO life tables (available at `http://apps.who.int/gho/data/?theme=main&vid=61780`)
- Total population N: reported in United Nations Statistics Division Demographic Statistics (available at `http://data.un.org/Data.aspx?d=POP&f=tableCode%`

3A22)

- Clearance rate ν: Taken from RNTCP Annual Performance Reports (available at http://tbcindia.gov.in/index4.php?lang=1&level=0&linkid=380&lid=2746).

- Most individuals with untreated TB will die within three years[1]. We set the annual probability of death for TB patients such that 90% will die after 3 years of infection.

D.2.2 Gonorrhea

- Estimates of prevalence I: U.S. Centers for Disease Control. https://www.cdc.gov/std/stats15/tables/21.htm Table 21. Gonorrhea — Reported Cases and Rates of Reported Cases by Age Group and Sex, United States, 2011-2015.

- Annual probability of death μ: Calculated from WHO life tables (available at http://apps.who.int/gho/data/?theme=main&vid=61780)

- Total population N: Taken from US Census Data (Annual estimates of the resident population by single year of age and sex for the United States: April 1, 2010 to July 1, 2016 (NC-EST2016-AGESEX-RES), available online at https://www.census.gov/data/datasets/2016/demo/popest/nation-detail.html)

- Once detected, most cases of gonorrhea can be effectively treated within two or three months, so the national clearance rate for this annual model depends on the detection rate of the disease. The clearance rate therefore varies over the samples used to calculate the transmission matrix.

- We assume no one dies from gonorrhea. Death rates are similar to the non-disease death rates.

[1] Tiemersma EW, van der Werf MJ, Borgdorff MW, Williams BG, Nagelkerke NJD. Natural History of Tuberculosis: Duration and Fatality of Untreated Pulmonary Tuberculosis in HIV Negative Patients: A Systematic Review. Pai M, ed. PLoS ONE. 2011;6(4):e17601. doi:10.1371/journal.pone.0017601.

Appendix E

Appendix to Chapter 7

E.1 Price of fairness

Theorem 17. *PoF^M is unbounded.*

Proof. Consider a graph G with two components: K which consists of 2 connected vertices, and S which is a star with $s+1$ nodes. Let the first group C_1 have only one node in K. All remaining nodes belong to the second group C_2, including one node x_1 in K and the central node of the star x_2. We have $k=1$ seed.

It is clear that the optimal seeding configuration is to seed x_2, which gives $\mathcal{I}^{\text{OPT}} = 1 + ps$. However, this is not a maximin fair seeding, as C_1 receives 0 influence. Instead, seeding x_1 is maximin fair, giving C_1 p influence and C_2 1 influence, giving a maximin utility $U^{\text{Maximin}}(\{x_1\}) = \min(p, \frac{1}{s+2})$. In this case, $\mathcal{I}^{\text{Maximin}} = 1 + p$.

As $s \to \infty$, $PoF_{\text{Maximin}} = \frac{1+ps}{1+p}$ becomes unboundedly large. □

Theorem 14. *U^{Maximin} and U^{Rational} are not submodular.*

We divide the proof of this theorem into two parts:

Proposition 3. Maximin utility U^{Maximin} is not submodular.

Proof. Let us consider a graph with 4 nodes $\{x, a, b, c\}$ where $\{x, a\}$ form community C_1 and $\{b, c\}$ form community C_2. Let $A = \{a, b\}$ and $B = \{a, b, c\}$ be two possible seeding

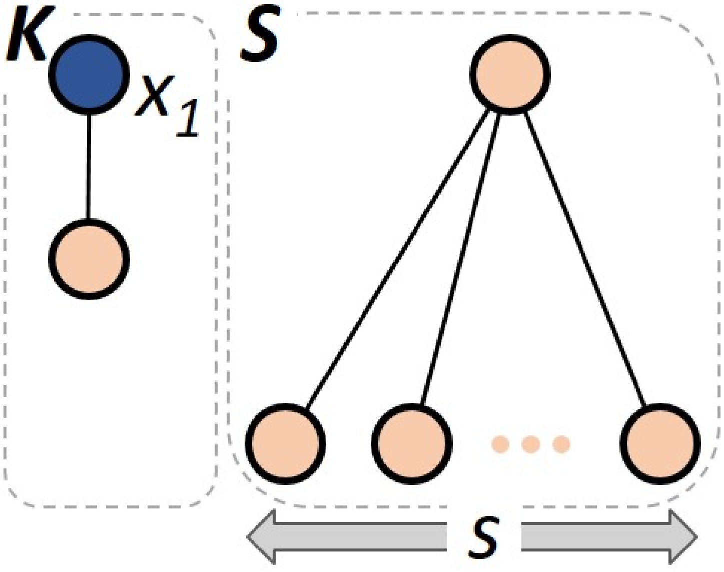


Figure E.1: *Example undirected network with unbounded PoF under maximin fairness*

configurations.

Notice that C_1 receives 1 influence in both configurations, which is weakly less than the influence received by C_2, and so, $U^{\text{Maximin}}(A) = U^{\text{Maximin}}(B) = 1/2$.

Now, consider adding x to the A and B. $U^{\text{Maximin}}(A \cup \{x\}) = 1/2$ since C_2 remains incompletely seeded. But $U^{\text{Maximin}}(B \cup \{x\}) = 1$ since both groups are fully seeded. □

Proposition 4. Group rational utility U^{Rational} is not submodular.

Proof. Recall the definition of group rational utility:

$$U^{\text{Rational}}(A) = \begin{cases} \mathcal{I}_G(A), & if\ diversity\ constraints\ satisfied \\ 0, & otherwise. \end{cases}$$

Let us consider the same graph as in Conjecture 3 with 4 nodes $\{x, a, b, c\}$ where $\{x, a\}$ form community C_1, and $\{b, c\}$ forms community C_2. $k = 4$ seeds are available, and so therefore the group rational constraints are only satisfied by seeding all vertices.

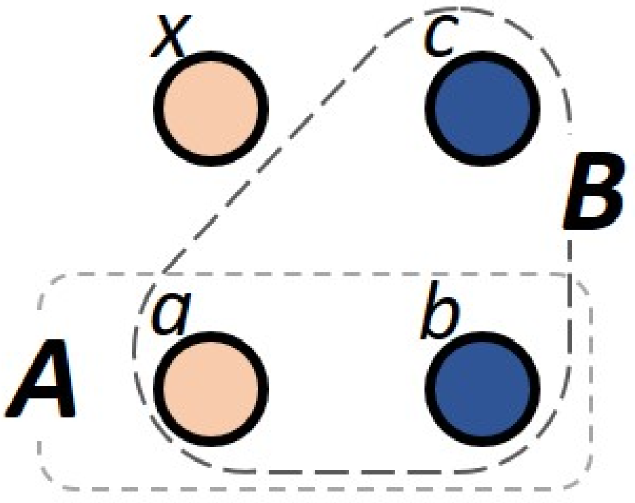


Figure E.2: *Example showing non-submodularity of maximin fairness*

Let $A = \{a, b\}$ and $B = \{a, b, c\}$. It is easy to verify that $U^{\text{Rational}}(A) = U^{\text{Rational}}(B) = U^{\text{Rational}}(A \cup \{x\}) = 0$ since none of these satisfy all group rational constraints. However, $U^{\text{Rational}}(B \cup \{x\}) > 0$, and so therefore $f(A \cup \{x\}) - f(A) < f(B \cup \{x\}) - f(B)$ for $A \subseteq B$, which contradicts the definition of submodularity.

□

Theorem 19. *Given propagation probability $p > 0$, it is possible to construct two families of graphs G with groups C_1 and C_2, and G' with groups C'_1 and C'_2, where $G' = G$ $C'_1 = C_1$ and C'_2 is obtained from C_2 by the addition of one vertex x_1 ($x_1 \in C_1$, $x_1 \notin C_2$. It is possible for $\lim_{n\to\infty} \frac{PoF^{\text{Maximin}}_{G'}}{PoF^{\text{Maximin}}_{G}} \to \infty$.*

Proof. Consider a graph G with two star components — S_1 with $s + 1$ vertices with a central node x_1, and S_2 with $t + 2$ vertices with central node x_2 ($s > t$) — and r isolated nodes. There are two groups: C_1 contains 2 vertices, x_1, a non-central node from S_2 and the $r - 2$ isolated nodes; C_2 contains $s + t + 1$ remaining vertices, including x_2 ($s > t$ and $r > 2$.) There is one seed ($k = 1$), and a total of $n = s + t + 3$ nodes.

It is easy to see that the Maximin configuration is to seed x_1, which gives C_1 1 influence,

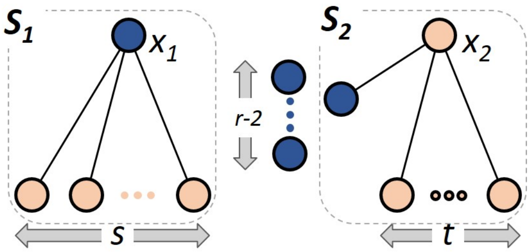


Figure E.3: *G with disjoint groups.*

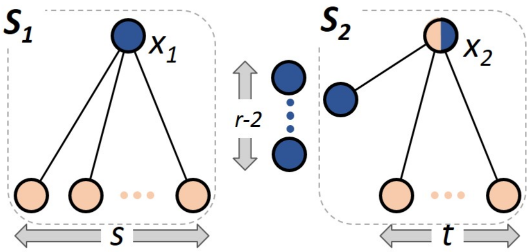


Figure E.4: *G' with overlapping groups.*

and C_2 ps influence. This gives a Maximin influence $\mathcal{I}_G^{\text{Maximin}} = 1 + ps$.[1]

Now, consider a modified graph $G' = G$, but with our groups modified by allowing x_2 to belong to both communities. That is, $C_1' = C_1$ and $C_2' = C_2 \cup \{x_2\}$. The Maximin configuration has two possibilities: either $\{x_1\}$ remains the Maximin configuration, or $\{x_2\}$ becomes the new Maximin configuration. In order for the latter case to be true, seeding $\{x_2\}$ must provide higher proportional influence to the least well-off group than seeding $\{x_1\}$.

Seeding $\{x_2\}$ generates $\frac{1+p}{r+1}$ influence for C_1, and $\frac{1+pt}{s+t+1}$ influence for C_2. Seeding $\{x_1\}$ generates $\frac{1}{r+1}$ influence for C_1, and $\frac{ps}{s+t+1}$. In order for $\{x_2\}$ to be the preferred seeding configuration, it must be true that $\min(\frac{1+p}{r+1}, \frac{1+pt}{s+t+1}) > \min(\frac{1}{r+1}, \frac{ps}{s+t+1})$. It is sufficient to require that $\frac{1+p}{r+1} > \frac{1}{r+1}$ and $\frac{1+pt}{s+t+1} > \frac{1}{r+1}$. The first condition is true for all $p > 0$. The second condition is true for $t > \frac{s}{p(r+1)-1} - \frac{r}{p(R+1)-1}$. Thus, let us set $r = s + \frac{1}{p} - 1$ (which turns the denominators to p) and so $t > \frac{1}{p}$ will satisfy the second condition.

Then,

$$\begin{aligned}\lim_{n\to\infty} \frac{PoF^{\text{Maximin}}(G')}{PoF^{\text{Maximin}}(G)} &= \lim_{n\to\infty} \frac{\mathcal{I}^{\text{Maximin}}(G)}{\mathcal{I}^{\text{Maximin}}(G')} \\ &= \lim_{n\to\infty} \frac{1+sp}{1+p(t+1)} \\ &= \lim_{n\to\infty} \frac{1+sp}{2+p}\end{aligned}$$

[1] We do not need to calculate U^{Maximin} explicitly at any point in this proof as it is not required for the proof to work.

$$= \infty$$

□

E.2 Analysis of multiobjective submodular maximization problem

Consider a collection of monotone submodular functions $f_1 ... f_m$ with corresponding multilinear extensions $F_1 ... F_m$. We will assume that the maximum singleton value of any item in the ground set V is bounded as $f_i\{v\} \leq b$ for all $i \in [m], v \in V$. Suppose that we are given a target value W_i for each f_i and would like to find a set S with $|S| \leq k$ which guarantees $f_i(S) \geq W_i$ for all i. We are promised that such an S exists. We will give an approximation algorithm for this problem which improves in terms of both runtime and approximation ratio on the best current algorithms, given by Udwani [Udw18], who in turn build on the work of Chekuri et al. [CVZ10].

Our algorithm follows the overall template of [Udw18], which carries out three steps (given a precision level ϵ).

1. Make a pass over the ground set, maintaining a set S_1. Add to S_1 every item which has value at least $\epsilon^3 W_i$ for some f_i.
2. Define $\mathcal{P}(\mathcal{M})$ to be the uniform matroid polytope for budget $k - |S_1|$. Use a subroutine to find a point $x \in \mathcal{P}(\mathcal{M})$ satisfying $F_i(x|x_{S_1}) \geq \alpha\left(W_i - f_i(S_1)\right) - \epsilon$ for all i and some approximation ratio α. This is the key step where we improve the runtime and approximation ratio.
3. Round x to a set S_2 using the swap rounding algorithm of [CVZ10] Output $S_1 \cup S_2$.

Our primary technical contribution is an algorithm for the second step which guarantees $\alpha = \left(1 - \frac{1}{e}\right)$. It uses access to three kinds of stochastic oracles for the functions and their multilinear extensions:

1. A stochastic value oracle for singletons $\mathcal{A}^i_{\text{val}}$ corresponding to each f_i. Given an item v, this oracle returns a value $\mathcal{A}^i_{\text{val}}(v)$ with $\mathbb{E}\left[\mathcal{A}^i_{\text{val}}(S)\right] = f_i(\{v\})$ and Var $\left[\mathcal{A}^i_{\text{val}}(S)\right] \leq c_{\text{val}}$.

2. A stochastic gradient oracle $\mathcal{A}^i_{\text{grad}}$ for each multilinear extension F_i. Given a point $x \in \mathcal{P}(\mathcal{M})$, $\mathcal{A}^i_{\text{grad}}(x)$ satisfies $\mathbb{E}\left[\mathcal{A}^i_{\text{grad}}\right] = \nabla_x F_i(x)$ and $\left\|\mathcal{A}^i_{\text{grad}}(x)\right\|_\infty \leq c_{\text{grad}}$

3. A stochastic gradient oracle $\mathcal{A}^j_{\text{item}}$ corresponding to each item $j \in [n]$. Given a point $x \in \mathcal{P}(\mathcal{M})$, $\mathcal{A}^j_{\text{item}}(x)$ satisfies $\mathbb{E}\left[\mathcal{A}^j_{\text{item}}(x)\right] = \left[\nabla_{x_j} F_1(x) ... \nabla_{x_j} F_m(x)\right]$ and $\left\|\mathcal{A}^j_{\text{item}}(x)\right\|_\infty \leq c_{\text{item}}$. Note that this can be simulated from the above oracle, but may sometimes admit more efficient implementations.

We now analyze this algorithm. We start by recalling a technical lemma on the smoothness of the multilinear extension:

Lemma 45 (Hassani et al. [HSK17], Lemma C.1)**.** *For any monotone submodular set function f and its multilinear extension F, $||\nabla F(x) - \nabla F(y)||_\infty \leq b||x - y||_1$ where $b = \max_{v \in V} f(\{v\})$. That is, F is b-smooth with respect to the ℓ_1 norm.*

Lemma 46. *F is b-Lipschitz in the ℓ_1 norm.*

Proof. Recall that $\nabla_{x_j} F(x) = \mathbb{E}_{S \sim x}[f(S \cup \{j\}) - f(S \setminus \{j\})]$ CITE, where $S \sim x$ denotes including each j in S independently with probability x_j. By submodularity, $\mathbb{E}_{S \sim x}[f(S \cup \{j\}) - f(S \setminus \{j\})] \leq f(\{j\}) \leq b$. Hence, $||\nabla_{x_j} F(x)||_\infty \leq b$ which proves the lemma. □

Next, we show a guarantee for the output of mirror descent in step 2(a).

Lemma 47. *For some $x \in \mathcal{P}(\mathcal{M})$, suppose that there exists a $v^* \in \mathcal{P}(\mathcal{M})$ such that $v^* \cdot \nabla F_i(x) \geq W_i - F_i(x)$ for all $i = 1...m$. Then, S-SP-MD returns a v satisfying $v \cdot \nabla F_i(x) \geq (1 - \epsilon)(W_i - F_i(x)) - \epsilon$ for all i with probability $1 - \delta$. There are $O\left(\frac{\left(c_{grad}\sqrt{k \log n} + k c_{item}\sqrt{\log n}\right)^2}{\epsilon^4} \log \frac{1}{\delta}\right)$ iterations, each requiring one call to oracles $\mathcal{A}^i_{grad}$ and $\mathcal{A}^j_{item}$ for some i and j, and $O(n + m)$ additional work.*

Proof. Our objective is to find a v satisfying $v \cdot F_i(x) \geq (1 - \epsilon)(W_i - F_i(x)) - \epsilon$, under the guarantee that such a v exists. Note that we call S-SP-MD only on the set of indices $\mathcal{I}$ where

$W_i - F_i(x) \geq \epsilon$. For all other indices, where the current solution is already within ϵ of the target, monotonicity of the F_i guarantees that $v \cdot F_i(x) \geq 0 \geq W_i - F_i(x) - \epsilon$.

The feasibility problem on the groups in $\mathcal{I}$ is equivalent to solving maxmin problem

$$\max_{v \in \mathcal{P}} \min_{i \in \mathcal{I}} \frac{v \cdot \nabla F_i(x)}{W_i - F_i(x)}$$

To see this, let OPT denote the optimal value for the maxmin problem; we are guaranteed $OPT \geq 1$. If we have v with maxmin value at least $OPT - \epsilon$, then v satisfies

$$v \cdot \nabla F_i(x) \geq (1 - \epsilon)(W_i - F_i(x)) \quad \forall i \in \mathcal{I}$$

We now prove that S-SP-MD produces a v with maxmin value at least $OPT - \epsilon$. Let A be a matrix where column i is $\frac{\nabla F_i(x)}{W_i - F_i(x)}$ for each $i \in \mathcal{I}$, and define $g(v, y) = v^\top A y$. Let $\Delta(\mathcal{I})$ be the $|\mathcal{I}|$-dimensional probability simplex. We would like to solve the problem

$$\max_{v \in \mathcal{P}(\mathcal{M})} \min_{y \in \Delta(\mathcal{I})} g(v, y)$$

which is easily seen to be equivalent to the original maxmin problem.

We will solve the above saddle point problem by running stochastic saddle point mirror descent with the negative entropy mirror map on the function g. We obtain stochastic estimates of $\nabla_v g(v, y)$ and $\nabla_y g(v, y)$ via calls to input the oracles. First, note that

$$\nabla_v g(v, y) = Ay = \sum_{i \in \mathcal{I}} y_i A_{\cdot, i} = \mathop{\mathbb{E}}_{i \sim y} [A_{\cdot, i}]$$

where $i \sim y$ denotes drawing index i with probability y_i (recall that $y \in \Delta(\mathcal{I})$ is a probability distribution). Hence, we can obtain an estimate $\hat{\nabla}_v$ of $\nabla_v g(v, y)$ by sampling $i \sim y$ and returning $\frac{1}{W_i - F_i(x)} \mathcal{A}^i_{\text{grad}}$. We are guaranteed $\left\| \hat{\nabla}_v \right\|_\infty \leq \frac{c_{\text{grad}}}{W_i - F_i(x)} \leq \frac{c_{\text{grad}}}{\epsilon}$. We take a similar strategy for $\nabla_y g(v, y)$: $v^\top A = k \left(\frac{1}{k} v\right) \hat{A} = k \, \mathbb{E}_{j \sim \frac{1}{k} v} [v_j A_j]$ (since $\frac{1}{k} v_j$ is a probability distribution). Hence, we can sample $j \sim \frac{1}{k}v$ and return $\hat{\nabla}_y = k \cdot \text{diag}\left(\frac{1}{\vec{W}-\vec{F}(x)}\right) \mathcal{A}^j_{\text{item}}(x)$. This satisfies $\left\|\hat{\nabla}_y\right\|_\infty \leq \frac{k}{\epsilon} c_{\text{item}}$.

Note that we can bound the diameter of $\mathcal{P}(\mathcal{M})$ with respect to the mirror map by $\sqrt{k \log n}$ (see [HSK17]) and the diameter of Δ^m by $\sqrt{\log m}$ (see [NJLS09]). We will run mirror descent for T' iterations. Let $\bar{x} = \frac{1}{T'}\sum_{t=1}^{T'} x^t$ and $\bar{y} = \frac{1}{T'}\sum_{t=1}^{T'} y^t$. Now applying Proposition 3.2 of Nemirovski et al. [NJLS09] implies that after T' iterations we have

$$\Pr\left[\max_{v \in \mathcal{P}(\mathcal{M})} g(v, \bar{y}) - \min_{y \in \Delta(\mathcal{I})} g(\bar{v}, y) \geq \frac{(8+2\Omega)\sqrt{5}\left(c_{\text{grad}}\sqrt{k \log n} + k c_{\text{item}}\sqrt{\log n}\right)}{\epsilon\sqrt{T}}\right] \leq 2e^{-\Omega}$$

and so taking $T' = O\left(\frac{\left(c_{\text{grad}}\sqrt{k \log n} + k c_{\text{item}}\sqrt{\log n}\right)^2 \log\frac{1}{\delta}}{\epsilon^4}\right)$ ensures that

$$\min_{y \in \Delta(\mathcal{I})} g(\bar{v}, y) \geq \max_{v \in \mathcal{P}(\mathcal{M})} \min_{y \in \Delta(\mathcal{I})} g(v, y) - \epsilon.$$

holds with probability at least $1 - \delta$. □

Theorem 29. *Suppose that there exists some $x \in \mathcal{P}(\mathcal{M})$ satisfying $F_i(x) \geq W_i$ for all $i = 1...m$. Then, after $T = \frac{bk^2}{\epsilon}$ iterations, the algorithm returns a point x^T satisfying $F_i(x^T) \geq (1-\epsilon)\left(1 - \frac{1}{e}\right) W_i - \epsilon$ for all i. Each iteration requires one call to mirror descent at success probability $\delta' = \frac{\delta\epsilon}{bk^2}$ and precision level $\epsilon' = \frac{\epsilon}{2}$, $O(m)$ ϵ-accurate value oracle calls, and $O(n)$ additional work.*

Proof. We analyze the progress that the algorithm makes with respect to each F_i over a single step t. Using the guarantee for the subroutine mirror descent (run with a precision level ϵ_1 to be set below), and assuming that the values $\{W_i\}$ are feasible, we have with probability at least $1 - \delta$

$$\begin{aligned}
F_i(x^t) - F_i(x^{t-1}) &\geq \frac{1}{T}\left[\nabla F_i(x^{t-1}) \cdot v^t\right] - \frac{b}{2}\left\|x^t - x^{t-1}\right\|_1^2 \text{(Lemma 45)} \\
&\geq \frac{1}{T}\left[\nabla F_i(x^{t-1}) \cdot v^t\right] - \frac{bk^2}{2T^2}(\ell_1 \text{ diameter of } \mathcal{P}(\mathcal{M})) \\
&\geq \frac{1}{T}\left((1-\epsilon_1)(W_i - F_i(x^{t-1})) - \epsilon_1\right) - \frac{bk^2}{2T^2}\text{(Lemma 47)}
\end{aligned}$$

which implies

$$W_i - F_i(x^t) \leq \left(1 - \frac{1-\epsilon_1}{T}\right)\left[W_i - F_i(x^{t-1})\right] + \frac{\epsilon_1}{T} + \frac{bk^2}{2T^2}$$

and so after T steps

$$\begin{aligned} W_i - F_i(x^T) &\leq \left(1 - \frac{1-\epsilon_1}{T}\right)^T \left[W_i - F_i(x^0)\right] + \epsilon_1 + \frac{bk^2}{2T} \\ &\leq \frac{1}{e^{1-\epsilon_1}} W_i + \epsilon_1 + \frac{bk^2}{2T} \end{aligned}$$

holds with probability at least $1 - T\delta$ via union bound. Taking $\epsilon_1 = \frac{\epsilon}{2}$, $T = \frac{bk^2}{\epsilon}$, and running mirror descent with success probability $\frac{\delta}{T}$ at each iteration ensures that

$$\begin{aligned} F_i(x^T) &\geq \left(1 - \frac{1}{e^{1-\epsilon}}\right) W_i - \epsilon \\ &\geq (1-\epsilon)\left(1 - \frac{1}{e}\right) W_i - \epsilon \end{aligned}$$

holds for all i with probability at least $1 - \delta$, which completes the guarantee for the solution quality. To obtain the bound on additional work done by the algorithm, we note that the only operation performed besides calling mirror descent is adding v^t to the current iterate, which takes time $O(n)$. □

Theorem 30. *Given a feasible set of target values $W_1 ... W_n$, Algorithm 11 outputs a set S such that $f_i(S) \geq (1-\epsilon)\left(1 - \frac{m}{k(1+\epsilon')\epsilon^3}\right)\left(1 - \frac{1}{e}\right) W_i - \epsilon$ with probability at least $1 - \delta$. Asymptotically as $k \to \infty$, the approximation ratio can be set to approach $1 - 1/e$ so long as $m = o(k \log^3 k)$. The algorithm requires $O(nm)$ ϵ'-accurate value oracle calls, $O(m\frac{bk^2}{\epsilon}\log\frac{1}{\delta})$ ϵ-accurate value oracle calls, $O\left(\frac{bk^4c^2}{\epsilon^5}\log\left(n + \frac{bk}{\delta\epsilon}\right)\right)$ calls to $\mathcal{A}_{grad}$ and $\mathcal{A}_{item}$, and $O\left(\frac{nk^2b^2}{\epsilon^2} + \frac{mk^2b}{\epsilon} + \frac{k^3b^2}{\epsilon^2}\right)$ additional work.*

Proof. ThresholdInclude produces a set S_1 for which each item $j \in S_1$ satisfies $f_i(\{j\}) \geq W_i(1+\epsilon')\epsilon^3$ for some i, and any $j \notin S_1$ satisfies $f_i(\{j\}) \leq W_i\epsilon^3$ for all i. Note that there can be at most $\frac{1}{(1+\epsilon')\epsilon^3}$ items with $f_i(\{j\}) \geq W_i(1+\epsilon')\epsilon^3$ for any given i (combining submodularity with our WLOG assumption that f_i is upper bounded by W_i). Hence, $|S_1| \leq \frac{m}{(1+\epsilon')\epsilon^3}$. Define

$k_1 = k - |S_1|$.

Now we lower bound the marginal gain of the fractional vector x returned by MultiobjectiveFW. So long as the target values $\{\frac{k_1}{k}\left(W_i - f_i(S_1)\right)\}$ are feasible, we are guaranteed that $F_i(x|S_1) \geq \frac{k_1}{k}\left(1 - \frac{1}{e}\right)\left(W_i - f_i(S_1)\right) - \epsilon$. for all i. To see feasibility, let S^* be the promised set satisfying the overall feasibility problem (i.e., $f_i(S^*) \geq W_i$ for all i). Let x_S denote the indicator vector of the set S. We have that $|S^* \setminus S_1| \leq k$, and $F_i(x_{S^* \setminus S_1}|x_{S_1}) = f_i(S^*|S_1) \geq W_i - f_i(S_1)$. Using Corollary 3 of [Udw18], the point $x' = \frac{k_1}{k}x_{S^* \setminus S_1}$ satisfies $F_i(x'|x_{S_1}) \geq \frac{k_1}{k}(W_i - f_i(S_1))$. x' is also feasible for the continuous problem since $||x'||_1 \leq k_1$. Now applying Theorem 29 guarantees that $F_i(x|S_1) \geq \frac{k_1}{k}\left(1 - \frac{1}{e}\right)\left(W_i - f_i(S_1)\right) - \epsilon$ with probability at least $1 - \delta$.

Lastly, we need to handle the rounding process. We first take the point x and approximately decompose it into a convex combination of integral points of $\mathcal{P}$. This is done using the algorithm of Mirrokni et al. [MLVW17], which produces a point x_{int} satisfying $||x_{\text{int}} - x||_1 \leq \epsilon$ along with a decomposition of x_{int} into $O(\frac{k^2}{\epsilon^2})$ integral points of $\mathcal{P}$ ([MLVW17], Proposition 5.1). If we run this algorithm with precision level $\frac{\epsilon}{b}$, Lemma 46 guarantees that $|F_i(x_{\text{int}}) - F_i(x)| \leq \epsilon$ for all i and hence $F_i(x_{\text{int}}|S_1) \geq \frac{k_1}{k}\left(1 - \frac{1}{e}\right)\left(W_i - f_i(S_1)\right) - 2\epsilon$. Applying Lemma 2 of [Udw18] (who summarize the guarantee for swap rounding proved by [CVZ10]), carrying out $O\left(\log \frac{1}{\delta}\right)$ iterations of swap rounding and taking the best outcome produces a set S_2 which satisfies $f(S_2|S_1) \geq (1-\epsilon)\frac{k_1}{k}\left(1 - \frac{1}{e}\right)\left(W_i - f_i(S_1)\right) - 3\epsilon$ with probability at least $1 - \delta$, provided that the best outcome is determined by calling a value oracle with precision level ϵ. Adding up the final guarantee, we have

$$\begin{aligned}
f(S) &= f(S_1 \cup S_2) \\
&= f(S_1) + f(S_2|S_1) \\
&\geq (1-\epsilon)\frac{k_1}{k}\left(1 - \frac{1}{e}\right)W_1 - 3\epsilon \\
&\geq (1-\epsilon)\left(1 - \frac{m}{k\epsilon^3(1+\epsilon')}\right)\left(1 - \frac{1}{e}\right)W_1 - 2\epsilon
\end{aligned}$$

and now rescaling ϵ by a factor $\frac{1}{3}$ gives the final approximation guarantee. The asymptotic $1 - 1/e$ approximation follows by setting ϵ as in [Udw18].

We now add up the final runtime. The first thresholding step requires n value oracle calls to each of the m objectives at precision level ϵ'. MultiobjectiveFW requires $\frac{bk^2}{\epsilon}$ iterations, each of which calls mirror descent once. Each invocation of mirror descent requires a total number of oracle calls which is bounded as $O\left(\frac{1}{\epsilon^4}\left(c_{\text{grad}}\sqrt{k\log n} + c_{\text{item}}k\sqrt{\log n}\right)^2 \log\frac{bk}{\delta\epsilon}\right)$. Recalling that $c = \max\{c_{\text{item}}, c_{\text{grad}}\}$, this is upper bounded by $O\left(\frac{c^2k^2}{\epsilon^4}\log\left(n + \frac{bk}{\delta\epsilon}\right)\right)$. Each iteration of MultiobjectiveFW also uses m value oracle calls at precision level ϵ. Finally, each iteration uses additional $O(n+m)$ overhead, for a total of $O\left(\frac{(n+m)k^2b}{\epsilon}\right)$. In the rounding procedure, we first need to involve ApproximateCaratheodory with precision level $\frac{\epsilon}{b}$, which per Proposition 5.1 of [MLVW17] requires $\frac{k^2b^2}{\epsilon}$ iterations, and one linear maximization over $\mathcal{P}$ per iteration. Since $\mathcal{P}$ is the uniform matroid polytope, each linear maximization takes time $O(n)$, and so this stage contributes time $O\left(\frac{nk^2b^2}{\epsilon}\right)$. Lastly, we have the $O\left(\log\frac{1}{\delta}\right)$ iterations of swap rounding. Since x_{int} was decomposed into $\frac{k^2b^2}{\epsilon^2}$ integral points, swap rounding takes time $\frac{k^3b^2}{\epsilon^2}$ for each iteration [CVZ10]. We also need one ϵ-accurate value oracle call to each of the objective functions per iteration so that we can select the (approximately) best set. Combining these bounds results in the final stated runtime. □

E.3 Efficient stochastic gradient estimates

We now give efficient implementations for the oracles $\mathcal{A}_{\text{grad}}$ and $\mathcal{A}_{\text{item}}$. They run in combined time $O\left(k\left(|V| + |E|\right)\log^2\frac{|V|}{\delta}\right)$ time, where the operation succeeds with probability $1 - \delta$. Our implementations guarantee $c \leq 2b$ whenever they succeed.

The starting point is to recall that the gradients of the multilinear extension F_i satisfy

$$\begin{aligned}\nabla_{x_j} F_i &= \mathop{\mathbb{E}}_{S\sim x}[f(S\cup\{j\}) - f(S\setminus\{j\})]\\ &= \mathop{\mathbb{E}}_{S\sim x,\xi\sim P}[f(S\cup\{j\},\xi) - f(S\setminus\{j\},\xi)] \qquad \text{(E.1)}\end{aligned}$$

Note that for any fixed i and x_j, we can obtain a stochastic estimate of this quantity in time $O(|V| + |E|)$ by first drawing a set $S \sim x$, simulating the cascade process, and counting the number of of nodes reached with and without item j. By submodularity, the resulting estimate satisfies $f(S \cup \{j\}, \xi) - f(S \setminus \{j\}, \xi) \leq b$ for any S and ξ. Naively repeating this process over all i, j would hence require time $O(|V|(|V| + |E|)m)$. We now show how to implement the required oracles by drawing a number of samples that scales only with $k \log |V|$ instead of $|V|$.

Implementing $\mathcal{A}_{\text{item}}$ is simpler because we only need to estimate $\left[\nabla_{x_j} F_1(x) ... \nabla_{x_j} F_m(x)\right]$ for a single fixed x_j. Hence, we can draw a single S, ξ, count the number of nodes reachable in each group under ξ with set $S \setminus \{j\}$, and then count the number of nodes reachable with set $S \cup \{j\}$. This takes time $O\left(|V| + |E|\right)$.

Efficiently implementing $\mathcal{A}_{\text{grad}}$ is more difficult since we need to simultaneously estimate ∇F_i with respect to every x_j; hence, naive enumeration would take $O(|V|^2)$ time. We now detail our strategy. We start by considering a given sample (S, ξ) and show how to estimate the marginal contribution $f_i(S \cup \{j\}, \xi) - f_i(S, \xi)$ for a given i and and *all* $j \notin S$ in total runtime $O\left((|V| + |E|) \log \frac{|V|}{\delta}\right)$. We first remove all nodes from G that are reachable from S under ξ, which takes time $O\left(|V| + |E|\right)$. Any node removed in this stage has marginal contribution 0. Next, we remove all nodes that are isolated in the remaining subgraph and assign them marginal contribution 1 if they are part of group i. This stage takes time $O(|V|)$.

Now we deal with the remaining nodes. Here, determining their marginal contribution of node v to group i amounts to estimating the number of nodes of group i which are reachable from v in ξ. We use the size estimation framework of Cohen [Coh97], which allows us to simultaneously produce an unbiased estimate of every remaining node's contribution to group i in time $O\left(|E|\right)$. We apply the weighted version of the estimator, where every node in group i has weight 1 and all other nodes have weight 0. We take $O(\left(\log \frac{|V|}{\delta}\right)$ independent repetitions of the estimation process, resulting in $O\left(|E| \log \frac{|V|}{\delta}\right)$ runtime. For a given group i, and using ℓ repetitions, Cohen's estimator produces an estimate $\Delta(v)$ for each node which satisfies

1. $\mathbb{E}[\Delta(V)] = f_i(\{v\}|S)$

2. $\Pr\left[|\Delta(v) - f_i(\{v\}|S)| \geq \epsilon f_i(\{v\}|S)\right] \leq e^{-\Omega\left(\epsilon^2 \ell\right)}$ for any $0 \leq \epsilon \leq 1$

We fix $\epsilon = 1$ as an arbitrary constant and use $\ell = O\left(\log \frac{|V|}{\delta}\right)$. This allows us to use union bound combined with the second property of the estimator to argue that over all nodes combined

$$\Pr\left[\Delta(v) \geq 2b\right] \leq \Pr\left[\Delta(v) \geq 2f_i(\{v\}|S)\right] \leq \delta$$

and so the resulting gradients will satisfy our stated bounds on c_{item} and c_{grad} with high probability.

Our overall strategy is to generate enough samples that every node is missing from S in at least one of them. Then, we can use a node's marginal contribution in the sample from which it missing as its gradient estimate. Note that a node j is absent from any given sample with probability $1 - x_j$. Given budget k, at most $\frac{k}{1-\frac{1}{k+1}} = k+1$ nodes can have $x_j \geq 1 - \frac{1}{k+1}$. For any such node, we can explicitly estimate a sample of Equation E.1 using $O\left(|V| + |E|\right)$ time per node, for $O\left(k\left(|V| + |E|\right)\right)$ total. For the remaining nodes, a simple argument shows that taking $(k+1)\log\frac{|V|}{\delta}$ samples is sufficient to ensure that each node is missing from at least one sample with combined probability $1 - \delta$. Summing up, the total runtime to generate $\hat{A}$ is $O\left(k\left(|V| + |E|\right)\log^2\frac{|V|}{\delta}\right)$.

E.4 Runtime comparison with previous work

The best previous algorithm for multiobjective submodular maximization [Udw18] uses the same overall framework as us, but uses a MWU algorithm for the second stage (the continuous maximization problem). The MWU algorithm runs $O\left(\frac{m}{\epsilon^2}\right)$ iterations, where each iteration requires a call to a greedy algorithm that maximizes a weighted combination of the f_i. Using the best implementation of the greedy algorithm [BV14][2] requires $O\left(\frac{n}{\epsilon}\log\frac{n}{\epsilon}\right)$ value

[2] While there are efficient special-purpose techniques for influence maximization on a given graph, it is not obvious how to adapt them to deal with the weighted combination of group objectives.

oracle calls, for $O\left(\frac{n}{\epsilon^3}\log m \log \frac{n}{\epsilon}\right)$ such calls in total. By comparison, our algorithm accesses the function through calls to the gradient oracles $\mathcal{A}_{\text{item}}$ and $\mathcal{A}_{\text{grad}}$. It makes a number of calls to these oracles which is only logarithmic in n, scaling as $O\left(\frac{bc^2k^4}{\epsilon^3}\log\left(n + \frac{bk}{\delta\epsilon}\right)\right)$. Since gradient oracle calls can typically be implemented in similar asymptotic runtime to value oracle calls for common classes of functions (as we have demonstrated for influence maximization), our algorithm effectively saves a factor $O(n)$ runtime in exchange for worse dependence on k and b. Since we expect n to grow much faster than k or b (in many typical applications, b is a small constant [HSK17]), this is often an improvement in asymptotic runtime. For influence maximization in particular, it is easy to see that a value oracle call for a given group cannot be implemented in less than $O(|V| + |E|)$ time, which matches (up to log factors) our stochastic gradient oracle's dependence on the graph size.

Appendix F

Appendix to Chapter 10

F.1 Proofs

F.1.1 Exact expression for gradients

Define $R_i = \sum_{j=1}^n r_{ji}$ and $C_i = \sum_{j=1}^n r_{ji} x_j$. We will work with $C_i \in \mathbb{R}^{p \times 1}$ as a column vector. For a fixed i, j, we have

$$\frac{\partial f_{i,\cdot}}{\partial x_j} = -\frac{R_i x_j \left[\frac{\partial r_{ji}}{\partial x_j}\right]^\top - C_i \left[\frac{\partial r_{ji}}{\partial x_j}\right]^\top}{R_i^2} - \frac{r_{ji}}{R_i} I$$

where I denotes the p-dimensional identity matrix. Similarly, fixing i, k gives

$$\frac{\partial f_{i,\cdot}}{\partial \mu_k} = \delta_{ik} I - \frac{R_i \sum_{j=1}^n x_j \left[\frac{\partial r_{ji}}{\partial \mu_k}\right]^\top - C_i \left[\sum_{j=1}^n \frac{\partial r_{ji}}{\partial \mu_k}\right]^\top}{R_i^2}$$

F.1.2 Guarantee for approximate gradients

Theorem 31. *Suppose that for all points j, $||x_j - \mu_i|| - ||x_j - \mu_{c(j)}|| \geq \delta$ for all $i \neq c(j)$ and that for all clusters i, $\sum_{j=1}^n r_{ji} \geq \alpha n$. Moreover, suppose that $\beta\delta > \log \frac{2\beta K^2}{\alpha}$. Then, $\left|\left|\frac{\partial f}{\partial \mu} - I\right|\right|_1 \leq \exp(-\delta\beta)\left(\frac{K^2\beta}{\frac{1}{2}\alpha - K^2\beta\exp(-\delta\beta)}\right)$ where $||\cdot||_1$ is the operator 1-norm.*

We focus on the off-diagonal component of $\frac{\partial f_{im}}{\partial \mu_{k\ell}}$, given by

$$A_{(i,m),(k,\ell)} = -\frac{R_i \sum_{j=1}^{n} x_j^m \left[\frac{\partial r_{ji}}{\partial \mu_k^\ell}\right] - C_i^m \left[\sum_{j=1}^{n} \frac{\partial r_{ji}}{\partial \mu_k^\ell}\right]}{R_i^2}$$

The key term here is $\frac{\partial r_{ji}}{\partial \mu_k^\ell}$. Let $s_{ji} = -\beta||x_j - \mu_i||$ Since r is defined via the softmax function, we have

$$\frac{\partial r_{ji}}{\partial \mu_k^\ell} = \frac{\partial r_{ji}}{\partial s_{jk}} \frac{\partial s_{jk}}{\partial \mu_k^\ell}$$

where

$$\frac{\partial r_{ji}}{\partial s_{jk}} = \begin{cases} r_{ji}(1 - r_{ji}) & \text{if i = k} \\ -r_{ji} r_{jk} & \text{otherwise.} \end{cases}$$

Note now via Lemma 48, in both cases we have that

$$\left|\frac{\partial r_{ji}}{\partial s_{jk}}\right| \le K \exp(-\beta\delta)$$

Define $\epsilon = K \exp(-\beta\delta)$ and note that we have that $\left|\frac{\partial s_{jk}}{\partial \mu_k^\ell}\right| \le \beta$, since we defined s in terms of cosine similarity and have assumed that the input is normalized. Putting this together, we have

$$\begin{aligned} \left|A_{(i,m),(k,\ell)}\right| &\le \frac{\sum_{j=1}^{n} x_j^m \epsilon\beta}{R_i} + \frac{C_i^m n\epsilon\beta}{R_i^2} \\ &\le \frac{\epsilon\beta \sum_{j=1}^{n} x_j^m}{\alpha n} + \frac{\mu_i^m n\epsilon\beta}{R_i} \\ &\le \frac{\epsilon\beta \sum_{j=1}^{n} x_j^m}{\alpha n} + \frac{\mu_i^m \epsilon\beta}{\alpha} \end{aligned}$$

and so

$$
\begin{aligned}
||A||_1 &= \max_{(k,\ell)} \sum_{(i,m)} A_{(i,m),(k,\ell)} \\
&\le \max_{(k,\ell)} \sum_{(i,m)} \frac{\epsilon\beta \sum_{j=1}^n x_j^m}{\alpha n} + \frac{\mu_i^m \epsilon\beta}{\alpha} \\
&\le \max_{(k,\ell)} \sum_i \frac{\epsilon\beta n}{\alpha n} + \frac{\epsilon\beta}{\alpha} \quad \text{(since } ||x_j||_1, ||\mu_i||_1 \le 1) \\
&\le \frac{2K\epsilon\beta}{\alpha} \\
&= \frac{2K^2\beta\exp(-\beta\delta)}{\alpha}.
\end{aligned}
$$

Since by assumption $\beta\delta > \log\frac{2\beta K^2}{\alpha}$, we know that $||A||_1 < 1$ and applying Lemma 49 competes the proof.

Lemma 48. *Consider a point j and let $i = \arg\max_k r_{jk}$. Then, $r_{ji} \ge \frac{1}{1+K\exp(-\beta\delta)}$, and correspondingly, $\sum_{k\neq i} r_{jk} \le \frac{K\exp(-\beta\delta)}{K\exp(-\beta\delta)+1} \le K\exp(-\beta\delta)$.*

Proof. Equation 4 of [Tit16] gives that

$$
r_{ij} \ge \prod_{k\neq i} \frac{1}{1+\exp(-(s_i - s_k))}.
$$

Since by assumption we have $-||x_j - \mu_i|| \ge \delta||x_j - \mu_k||$, we obtain

$$
\begin{aligned}
r_{ij} &\ge \prod_{k\neq i} \frac{1}{1+\exp(-\delta\beta)} \\
&\ge \frac{1}{1+K\exp(-\delta\beta)} \quad \text{(using that } \exp(-\delta\beta) \le 1).
\end{aligned}
$$

which proves the lemma. □

Lemma 49. *Suppose that for a matrix A, $||A - I|| \le \delta$ for some $\delta < 1$ and an operator norm $||\cdot||$. Then, $||A^{-1} - I|| \le \frac{\delta}{1-\delta}$.*

Proof. Let $B = I - A$. We have

$$\begin{aligned}
A^{-1} &= (I - B)^{-1} \\
&= \sum_{i=0}^{\infty} B^i \quad \text{(using the Neumann series representation)} \\
&= I + \sum_{i=1}^{\infty} B^i
\end{aligned}$$

and so $||A^{-1} - I||_\infty = \left|\left|\sum_{i=1}^{\infty} B^i\right|\right|_\infty$. We have

$$\begin{aligned}
\left|\left|\sum_{i=1}^{\infty} B^i\right|\right|_\infty &\leq \sum_{i=1}^{\infty} ||B^i||_\infty \\
&\leq \sum_{i=1}^{\infty} ||B||_\infty^i \quad \text{(since operator norms are submultiplicative)} \\
&= \frac{\delta}{1-\delta} \quad \text{(geometric series).}
\end{aligned}$$

□

F.2 Experimental setup details

F.2.1 Hyperparameters

All methods were trained with the Adam optimizer. For the single-graph experiments, we tested the following settings on the pubmed graph (which was not used in our single-graph experiments):

- β = 1, 10, 30, 50
- learning rate = 0.01, 0.001
- training iterations = 100, 200, ..., 1000
- Number of forward pass k-means updates: 1, 3
- Whether to increase the number of k-means updates to 5 after 500 training iterations.
- GCN hidden layer size: 20, 50, 100

- Embedding dimension: 20, 50, 100

For all single-graph experiments we used $\beta = 30$ for the facility location objective and $\beta = 50$ for community detection, $\gamma = 100$, GCN hidden layer = embedding dimension = 50, 1 k-means update in the forward pass, learning rate = 0.01, and 1000 training iterations, with the number of k-means updates increasing to 5 after 500 iterations.

We tested the following set of hyperparameters on the validation set for each graph distribution

- β = 30, 50, 70, 100
- learning rate = 0.01, 0.001
- dropout = 0.5, 0.2
- training iterations = 10, 20...300
- Number of forward pass k-means updates: 1, 5, 10, 15
- Hidden layer size: 20, 50, 100
- Embedding dimension: 20, 50, 100

We selected $\beta = 70$, learning rate = 0.001, dropout = 0.2, and hidden layer = embedding dimension = 50 for all experiments. On the synthetic graphs we used 70 training iterations and 10 forward-pass k-means updates. For pubmed, we used 220 and 1, respectively.

F.2.2 Synthetic graph generation

Each node has a set of attributes y_i (in this case, demographic features simulated from real population data); node i forms a connection to node j with probability proportional to $e^{-\frac{1}{\rho}||y_i - y_j||}d(j)$ where $d(j)$ is the degree of node j. This models both the homophily and heavy-tailed degree distribution seen in real world networks. We took $\rho = 0.025$ to obtain a high degree of homophily, so that there is meaningful community structure. In order to make the problem more difficult, our method does not observe the features y; instead, we generate unsupervised features from the graph structure alone using role2vec [ARL+18] (which generates inductive representations based on motif counts that are meaningful across graphs). Each graph has 500 nodes.

F.2.3 Code

See `https://github.com/bwilder0/clusternet` for code and data used to run the experiments.

F.2.4 Hardware

All methods were run on a machine with 14 i9 3.1 GHz cores and 128 GB of RAM. For fair runtime comparisons with the baselines, all methods were run on CPU.

F.3 Results for $K = 10$

Table F.1: *Results for community detection. "-" for GCN-2Stage-Newman in the Learning + optimization section denotes that the method could not be run due to numerical issues.*

	Learning + optimization					Optimization				
	cora	cite.	prot.	adol	fb	cora	cite.	prot.	adol	fb
ClusterNet	**0.56**	**0.53**	**0.28**	**0.47**	**0.28**	**0.71**	**0.76**	**0.52**	0.55	**0.80**
GCN-e2e	0.01	0.01	0.06	0.08	0.00	0.07	0.08	0.14	0.15	0.15
Train-CNM	0.20	0.44	0.09	0.01	0.17	0.08	0.34	0.05	**0.60**	**0.80**
Train-Newman	0.08	0.15	0.15	0.14	0.07	0.20	0.22	0.29	0.30	0.47
Train-SC	0.06	0.04	0.05	0.22	0.21	0.15	0.08	0.07	0.46	0.79
GCN-2stage-CNM	0.20	0.23	0.18	0.32	0.08	-	-	-	-	-
GCN-2stage-Newman	0.01	0.00	0.00	-	0.00	-	-	-	-	-
GCN-2stage-SC	0.13	0.18	0.10	0.29	0.18	-	-	-	-	-

Table F.2: *Results for facility location*

	Learning + optimization					Optimization				
	cora	cite.	prot.	adol	fb	cora	cite.	prot.	adol	fb
ClusterNet	**9**	**14**	**7**	**5**	**2**	**8**	**13**	**6**	**5**	**2**
GCN-e2e	12	15	8	6	4	10	14	7	**5**	4
Train-greedy	14	16	8	8	6	9	14	7	6	5
Train-gonzalez	11	15	8	7	6	9	**13**	7	6	**2**
GCN-2Stage-greedy	14	16	8	7	6	-	-	-	-	-
GCN-2Stage-gonzalez	12	16	8	6	5	-	-	-	-	-

F.4 Timing Results

We run experiments on Intel i9 7940X @ 3.1 GHz with 128 GB of RAM. We report runtime in seconds. For algorithms with learned models, we report both the training time and the time to complete a single forward pass.

Table F.3: *Timing results for the community detection task (s)*

	cora	cite.	prot.	adol	fb
ClusterNet - Training Time	59.48	149.73	129.63	56.68	54.33
ClusterNet - Forward Pass	0.04	0.12	0.11	0.04	0.05
GCN-e2e - Training Time	36.83	54.99	34.60	29.04	28.17
GCN-e2e - Forward Pass	0.002	0.005	0.002	0.003	0.001
Train-CNM	1.31	1.28	1.02	1.03	2.94
Train-Newman	9.99	15.89	15.19	11.45	7.25
Train-SC	0.41	0.62	0.55	0.38	0.48
GCN-2Stage - Training Time	68.79	72.20	75.69	103.56	57.62
GCN-2Stage-CNM	119.34	178.39	159.64	101.64	142.02
GCN-2Stage-New.	37.96	58.26	51.70	33.14	43.88
GCN-2Stage-SC	0.40	0.61	0.50	0.33	0.36

Table F.4: *Timing results for the kcenter task (s)*

	cora	cite.	prot.	adol	fb
ClusterNet - Training Time	264.14	555.84	488.37	244.74	246.57
ClusterNet - Forward Pass	0.10	0.23	0.20	0.09	0.11
GCN-e2e - Training Time	237.68	511.23	446.76	229.49	221.28
GCN-e2e - Forward Pass	0.003	0.006	.005	0.004	.003
Train-Greedy	1029.18	2387	1966	619.06	1244.09
Train-Gonzalez	0.082	0.14	0.12	0.07	.066
GCN-2Stage - Training Time	73.82	70.21	103.98	75.48	104.66
GCN-2Stage-Greedy	1189.15	2367	2017	621.59	1237.871
GCN-2Stage-Gonzalez	0.18	0.28	0.25	0.13	0.13

Table F.5: *Timing results in the inductive setting for community detection task (s)*

	synthetic	pubmed
ClusterNet - Training time	6.57	13.74
ClusterNet - Forward Pass	0.003	0.008
GCN-e2e - Training time	11.40	15.86
GCN-e2e - Forward Pass	0.04	0.03
Train-CNM	0.08	0.17
Train-Newman	0.65	1.83
Train-SC	0.03	0.04
2Stage - Train	10.98	15.86
2Stage-CNM	3.23	13.73
2Stage-New.	1.12	4.29
2Stage-SC	0.04	0.10

Table F.6: *Timing results in the inductive setting for the kcenter task (s)*

	synthetic	pubmed
ClusterNet - Training Time	14.36	43.06
ClusterNet - Forward Pass	0.005	0.02
GCN-e2e - Training Time	9.49	33.73
GCN-e2e - Forward Pass	0.01	0.02
Train-Gonzalez	0.07	0.49
Train-Greedy	4.99	32.7
2Stage - Train	11.00	15.78
2Stage-Gonzalez	0.07	0.07
2Stage-Greedy	5.31	16.16

Appendix G

Appendix to Chapter 12

This section contains details of the methods (model used, parameter settings), strategy for inference, experimental details of the modeled scenarios, discussion of mechanisms for physical distancing, and additional results on model validation.

G.1 Methods

G.1.1 Model description

We develop an agent-based model for COVID-19 spread which accounts for the distributions of age, household types, comorbidities, and contact between different age groups in a given population. The model follows a *susceptible-exposed-infectious-removed (SEIR)* template [VdDLM99, BKO15].

Specifically, we simulate a population of n agents (or individuals), each with an age a_i, a set of comorbidities c_i, and a household (a set of other agents). We stratify age into ten-year intervals and incorporate hypertension and diabetes as comorbidities. These comorbidities are common worldwide [RAA+18] and have been associated with a higher risk of in-hospital death for COVID-19 patients [ZYD+20]. However, our model can be expanded to include other comorbidities of interest in the future. The specific procedure we use to sample agents from the joint distribution of age, household structures, and comorbidities is described below.

The simulation tracks two states for each individual: the *infection state* and the *isolation state*. The infection state is divided into {*susceptible, exposed, infectious, removed*}. *Susceptible* individuals are those who have never been contacted by an infectious individual. *Exposed* individuals are those who have had contact with an infectious individual, though not all exposed individuals become infectious. If an exposed individual contracts the disease, they proceed to the infectious state.[1] *Infectious* is further subdivided into severity levels {*presymptomatic, mild, severe, critical*}. We interpret mild severity as symptomatic (but not requiring hospitalization), severe as requiring hospitalization, and critical as eligible for intensive care unit (ICU) care. The *removed* state is further subdivided into {*recovered*, *deceased*}. Individuals in all severity levels can transmit the disease, but those in the *presymptomatic* state do so at a rate $\alpha < 1$ times that of symptomatic cases. The decision to incorporate reduced transmission for presymptomatic individuals is based on the fact that, though infection by presymptomatic individuals has been observed in case clusters and in examinations of serial intervals [BYW$^+$20, RSS$^+$20, DXW$^+$20], available evidence suggests that individuals with no or limited symptoms are less infectious than those with severe symptoms [LPC$^+$20]. Currently, our simulation incorporates two levels of infectiousness (before and after the onset of symptoms), but it can be adjusted as better information on how viral shedding increases with severity of illness becomes available. We acknowledge that our assumptions surrounding transmissibility and disease severity – as derived from existing literature – may serve as a limitation of our model, as many of these factors are evolving over time.

Each individual has a separate isolation state {*isolated*, *not isolated*}. If isolated, the individual is unable to infect others. We assume that (1) presymptomatic individuals are never isolated, (2) mild individuals become isolated over a mean time of λ_{isolate} days (see

[1] Currently, our simulation implementation does not separately track individuals who are exposed but do not become infected, and instead groups them with the susceptible population. This is because we assume that, if exposed again, they will become infected with the same probability as an individual who has never been exposed. However, the implementation can be modified to support either differing probabilities of contracting the disease after first exposure or policies that treat exposed and susceptible individuals differently.

Table 1) after the onset of symptoms, and (3) all severe and critical individuals are isolated. However, our simulation framework can easily accommodate different sets of assumptions about isolation (for example, preemptively isolating exposed individuals if they are known to have had contact with an infectious agent).

The disease is transmitted over a contact structure, which is divided into in-household and out-of-household groups. Each agent has a household consisting of a set of other agents. Individuals infect members of their households at a higher rate than out-of-household agents. We model out-of-household transmission using country-specific estimated contact matrices [PCJ17]. These matrices state the mean number of daily contacts an individual of a particular age strata has with individuals from each of the other age strata. We assume demographics (including age and household distribution) in Hubei and Lombardy are well-approximated by country-level data.

The model iterates over a series of discrete time steps, each representing a single day, from a starting time t_0 to an end time T. There are two main components to each time step: disease progression and new infections. The progression component is modeled by drawing two random variables for each individual each time they change severity levels (e.g., on entering the mild state). The first random variable is Bernoulli and indicates whether the individual will recover or progress to the next severity level. The second variable represents the amount of time until progression to the next severity level. We use exponential distributions for almost all time-to-event distributions, a common choice in the absence of specific distributional information [All10, Col15]. The exception is the incubation time between presymptomatic and mild states, where more specific information is available; here, we use a log-normal distribution (see $\mu_{e \to m}$ and $\sigma^2_{e \to m}$ in Table 1) based on estimates by [LGB$^+$20]. Table 1 summarizes all distributions and their parameters.

In the new infections component, individuals in the susceptible state may enter the exposed state. Infected individuals infect each of their household members with probability p_h at each time step. p_h is calibrated so that the total probability of infecting a household member before either isolation or recovery matches the estimated secondary attack rate for household members of COVID-19 patients (i.e., the average fraction of household members infected) [LEK20]. Infected individuals draw outside-of-household contacts from the general population using the country-specific contact matrix. For an infected individual of age group i, we sample $w_{ij}^s \sim \text{Poisson}(M_{ij}^s)$ contacts for each age group j and setting s where M^s is the country-specific contact matrix for setting s. We include contacts in work, school, and community settings. Poisson distributions are a standard choice for modeling contact distributions [PCJ17]. Then, we sample w_{ij}^s contacts of age j uniformly with replacement, and each contact is infected with the probability p_{inf}, the probability of infection given contact. There is evidence to suggest that the probability of infection is higher for an older individual than younger given the same exposure [ZLL$^+$20], consistent with decline in immune function with age. We adjust for this by letting the probability of infection be βp_{inf} when the exposed individual is over the age of 60, for $\beta > 1$. β is calibrated to match the fraction of deaths in China attributed to individuals over the age of 60, resulting in a value of 1.25. This is consistent with the relationship between age and attack rate amongst close contacts of a confirmed case reported by [ZLL$^+$20], where the increase in risk of infection for a contact over 65 years old was estimated in the range 1.12–1.92.

G.1.2 Sampling agents

Our process for sampling agents follows three steps that successively sample households, individual agents within households, and comorbidities for each agent. Because the full joint distributions over all of these quantities are not known, we implement a sampling procedure that respects the marginal distributions of household structure and age, as well as the marginal distribution for the occurrence of comorbidities within each age group.

First, we use information on the distribution of household structures to draw a type of household (e.g., single person, couple, nuclear family, or multigenerational family). Second, we sample the ages of the individual agents according to their role in the household (e.g., parent, child, or grandparent) combined with information about the age distribution of the population and the intergenerational interval. For China, we use household distributions from the 2010 Chinese census [HP15], intergenerational intervals from [HZWJ19], and the age distribution provided by UN population statistics [Uni19]. For Italy, we use demographic statistics from Statista online portal about the following: household structure distribution [Sta18b], single-person households [Sta18e], couples with children [Sta18c] and corresponding family size [Sta18a], and single parents with children [Sta18d]. Furthermore, we assume that children could stay within the family until the age of 30 and that couples without children were aged 30+, to account for societal patterns reported in familial studies which may have affected household distribution metrics [CLT14]. In New York City, we circumvent these steps by instead sampling individual households directly from census microdata. We use the public use microdata from the 2015 American Community Survey [Cen19]. We draw from household-level responses located in New York City, repeatedly sampling a household of individuals with their reported ages until the desired population size (8.4 million) is reached.

Third, we sample comorbidities from the corresponding country- and age-specific distributions. For China, we use estimates on age-specific prevalence of diabetes [XWH$^+$13] and hypertension [WCZ$^+$18]. For Italy, we use estimates from the Global Burden of Disease study on diabetes [RAA$^+$18] and a recent study of age-stratified hypertension prevalence [MCM$^+$17]. For New York City, we use city-level estimates of age-specific prevalence for both comorbidities [fDCP17, oHH16]. We ensure that diabetes and hypertension are appropriately correlated using a single global estimate for the probability of hypertension in individuals with diabetes [TO17]. An important limitation of this study is that using different data sources for comorbidity prevalence in each location (while necessary) may introduce bias; our analyses could be refined if more comprehensive data sources became available.

G.1.3 Estimating disease progression from age and comorbidities

Many of the parameters for this model are assigned values based on estimates in the literature, shown in Table 1. However, we currently lack a detailed understanding of the joint impact of age and comorbidities on disease progression and mortality. Currently, estimated infection fatality rates (IFRs) are available by age but not for each specific combination of age and comorbidities. To obtain these specific estimates, we model the IFR with a logistic regression fit to IFRs estimated by Verity et al. [VOD+20] on data from mainland China. The logistic model is discussed in the next section. This model yields $p_{m\to d}(a_i, c_i)$, the country-independent probability that an individual i of age a_i and comborbidity status c_i will die if infected with SARS-CoV-2. Corrections for country-specific differences in mortality are handled via the parameter d_{mult}.

The simulation also requires specific values for the probabilities of transitioning between the disease states mild, severe, critical, and death. However, there is currently insufficient information available to infer the probabilities of these individual transitions for each combination of age and comorbidity. We assume that while the absolute values of these probabilities may vary based on age and comorbidity, the *ratios* between them do not exhibit such strong dependency. In particular, we assume that there are coefficients $\gamma_{s\to c}(a_i)$ and $\gamma_{c\to d}$ such that $p_{s\to c}(a_i, c_i) = \gamma_{s\to c}(a_i) p_{m\to s}(a_i, c_i)$ and $p_{c\to d}(a_i) = \gamma_{c\to d} p_{m\to s}(a_i, c_i)$. We allow $\gamma_{s\to c}(a_i)$ to be age-specific while assuming that $\gamma_{c\to d}$ is age-homogeneous because of the information currently available to estimate them; namely, we set $\gamma_{s\to c}(a_i)$ based on the estimated probabilities of hospitalization from [VOD+20] and ICU admission by age group in the US from [Tea20] and $\gamma_{c\to d}$ based on the probability of death for all critical patients in China [fDCP20b]. Note that we assume both coefficients to be independent of the comorbodities c_i. Then, we can solve for $p_{m\to s}(a_i, c_i)$ such that

$$p_{m\to s}(a_i, c_i) \cdot \gamma_{s\to c}(a_i) p_{m\to s}(a_i, c_i) \cdot \gamma_{c\to d} p_{m\to s}(a_i, c_i) = p_{m\to d}(a_i, c_i),$$

and set $p_{s\to c}(a_i, c_i)$ and $p_{c\to d}(a_i, c_i)$ accordingly. Future work can relax the assumptions in this process as more information becomes available about how age and comorbidity impact the progression between disease states.

G.1.4 Estimating mortality from age and comorbidities

We require a model of $p_{m\to d}(a_i, c_i)$, however existing data sources only specify $p_{m\to d}(a_i)$ and $p_{m\to d}(c_i)$. To infer the joint distribution, we assume a linear (logistic) interaction between age bracket, diabetes status, and hypertension status. Specifically, we assume

$$p_{m\to d}(a_i, c_i) = \sigma\Big(\beta_{\text{age}}(a_i) + \beta_{\text{diabetes}}\mathbb{1}\left[\text{diabetes} \in c_i\right] + \beta_{\text{hypertension}}\mathbb{1}\left[\text{hypertension} \in c_i\right]\Big),$$

where $\beta_{\text{age}}(a_i)$ has a value for each age bracket (e.g., 20-30, 30-40, etc., 7 in total) and β_{diabetes} and $\beta_{\text{hypertension}}$ are scalars.

The marginal distribution $p_{m\to d}(a_i)$ is taken from [VOD$^+$20], which corrects for underreporting of infections in China. To obtain a comparable marginal distribution $p_{m\to d}(c_i)$, we scaled the reported CFR for each comorbidity group [fDCP20b] by an age-adjusted correction for reporting obtained based on [VOD$^+$20] (making the assumption that the probability of documentation is independent of comorbidity status after conditioning on age). We obtained data from the literature on the prevalence of diabetes and hypertension [WCZ$^+$18] in China by age [XWH$^+$13], as well as a single global estimate of $p(\text{hypertension}|\text{diabetes})$ [TO17]. We assume that these distributions are the same in COVID-19 patients as in the general population. However, we also conducted a sensitivity analysis to acknowledge the potential for increased comorbidity prevalence in COVID-19 patients, a scenario where comorbidities are also correlated to risk factors for transmission. The results (shown in Fig. S4) do not significantly alter our estimates. Given this information, we use gradient descent to find a set of parameters β which minimize the mean squared error in the following marginal consistency constraints, where we use x to denote the random variable of diabetes status and y to denote the random variable of hypertension status:

$$\begin{aligned}
p_{m\to d}(a_i) &= \sum_{x,y} p(x, y|a_i) p_{m\to d}(a_i, x, y), \quad \forall a_i, \\
p_{m\to d}(x) &= \sum_{a_i} p(a_i|x) \sum_{y} p(y|a_i, x) p_{m\to d}(a_i, x, y),
\end{aligned}$$

$$p_{m\to d}(y) = \sum_{a_i} p(a_i|y) \sum_{x} p(x|a_i, y) p_{m\to d}(a_i, x, y).$$

The set of estimated parameters are

$$\begin{aligned}
&\beta_{\text{age}}(18-30) = -8.49,\\
&\beta_{\text{age}}(30-40) = -7.68,\\
&\beta_{\text{age}}(40-50) = -7.41,\\
&\beta_{\text{age}}(50-60) = -6.39,\\
&\beta_{\text{age}}(60-70) = -5.41,\\
&\beta_{\text{age}}(70-80) = -4.54,\\
&\beta_{\text{age}}(80-100) = -4.05,\\
&\beta_{\text{diabetes}} = 1.22,\\
&\beta_{\text{hypertension}} = 1.58.
\end{aligned}$$

The coefficients should be interpreted relative to the baseline -8.49 value for the 18-30 group. For example, the value -7.68 for the 30-40 group indicates that the log-probability of mortality increases by 0.80 when age is increased from 18-30 to 30-40, holding comordibity status equal. Over 10 random restarts, the marginal values were always fit to within numerical tolerance by the same set of parameters (less than 0.1% maximum difference in the value of a parameter between runs). This suggests that the model parameters are fully identifiable in this setting.

G.2 Experimental settings

G.2.1 Experimental settings for Hubei

We draw a population of individuals from the age, household, and comorbidity distributions for China since more specific information is not available for Hubei (though the fraction of individuals over 65 is within the typical range for many Chinese provinces [Pen11]). We simulate a population of 58.5 million individuals, matching the population of the Hubei province. After the lockdown, all contact frequencies are reduced by $\delta_c = 0.993$, set to obtain the number of outside-of-household contacts reported in post-lockdown surveys [ZLL$^+$20]. Note that [ZLL$^+$20] reported a decline in 86.3% for total contacts, but this figure included within-household contact (which accounted for 94.1% of post-lockdown contact). We also modeled closure of schools on the lockdown date.

We set the range of the uniform prior distributions as follows. The prior over p_{inf} was set to contain all values with significant likelihood, with the final range being $[0.020, 0.035]$. The prior over t_0 was set to contain up to 7 days before the first reported case on November 17 [Sou20], and 3 days afterwards (for a set of 10 days total). It is possible that substantial new backdating of the start of the epidemic could alter our results. Finally, the parameter d_{mult} captures variation in IFR, which is not precisely known in any location. We start from age-stratified IFR estimates by Verity et al. [VOD$^+$20]. In our model, these values result in an overall IFR of approximately 0.4% (lower than the 0.66% estimated by [VOD$^+$20] because attack rates in our model are higher in younger groups, due to the larger numbers of daily contacts in younger groups vs older [PCJ17]). We placed a uniform prior over d_{mult} in the range [1,3], and then conditioned in the posterior on the IFR lying in the range 0.4–0.8%. Together, this procedure is designed to allow variation by approximately 50% around the IFR estimated by [VOD$^+$20].

G.2.2 Experimental settings for Lombardy

We simulate a population of 10 million individuals (representing the population of Lombardy) drawn from the Italian distribution of age, household structure, and comorbidity status. The full demographic information needed to parameterize the simulation was not available for Lombardy specifically, but available information suggests broadly similar characteristics (e.g., the median age in Lombardy is 45 [Ita19], comparable to Italy in general at 46.5 [Fac20]). After the lockdown on March 8, the number of contacts for all age groups is reduced to δ_c times its previous value. The prior for δ_c was uniform over the interval $[0, 0.1]$, reflecting a 90-100% reduction in outside of household contact (this interval was set to contain all values with significant likelihood). We also model closure of all schools on the lockdown date.

As in Hubei, we set the prior range for p_{inf} to include all values with significant likelihood, resulting in an interval [0.025, 0.04]. Also as in Hubei, the prior for t_0 was set to be uniform over a range of dates including up to 7 days before the infected travelers reportedly landed in Milan on January 23 [Car20], and up to 3 days afterwards. We adjusted the way the parameter d_{mult} was applied to account for the substantially different age composition of deaths in Italy than in either Hubei or New York City. Specifically, approximately 95% of reported deaths in Italy were among individuals 60 years or older, compared to approximately 80% in China [fDCP20b] or 73.6% in New York City [oHH20]. One potential factor which could contribute to disparities in death rates are reports that older individuals in severe condition may have been less likely to receive care under the triage strategies adopted in response to overburdened health systems in Italy [Pol20, Mou20]. Accordingly, instead of scaling fatality rates uniformly across age groups, we calibrated a multiplier for the fatality rate in the 60+ age group to match the fraction of deaths attributed to that group.

G.2.3 Experimental settings for New York City

We simulate a population of 8.4 million individuals (representing the population of New York City), sampled in household units from census microdata for New York City.We model a two-step reduction in contact, consistent with mobility data [Goo20, Una20]. The official lockdown was instituted on March 23, and we model a reduction in contact by δ_c on that date, with δ_c sampled uniformly from the interval [0, 0.1]. However, mobility data shows that significant reductions in mobility began the week before the official lockdown, suggesting preemptive distancing measures by individuals in anticipation of an official policy. Accordingly, we model contact in all non-household settings as reduced to 67% of its previous value starting on March 16. This factor was chosen based on the reduction in close physical encounters between cell phones represented in Unacast location data during the week of March 16 [Una20]; we opted to fix this value instead of estimating a separate parameter in order to avoid increasing the number of free parameters in the model. Our estimated values for δ_c suggest that encounter rate data from mobile phones may be a reasonable proxy for the reduction in physical contacts; our posterior mean estimate for δ_c in New York City was 0.97, while Unacast encounter data showed a peak reduction in the encounter rate of 90-99% across the various boroughs of New York City.

We set the prior range for p_{inf} to be [0.03, 0.07], again set to include all values with non-negligible likelihood. d_{mult} was given a uniform prior over the range [1, 4], allowing for but not mandating a higher IFR than Hubei. We handled the starting conditions of the epidemic differently in New York City than in Lombardy due to reports of multiple distinct importation events over the course of February [GRHS$^+$20], with modeling studies suggesting the potential for thousands of cases present by the start of March [CG20]. Instead of attempting to explicitly model multiple importations, we fixed t_0 at February 10 and placed a uniform prior over the number of infected individuals present on that date, in the range [5, 500].

G.2.4 Experimental settings for containment policies

We simulate two sets of scenarios modeling the impact of different containment policies. The first set of scenarios, shown in the main text, simulates a second-wave scenario for each location. We initialize the simulation to draws from the posterior distribution for said location. The posterior is over both the population-level model parameters, as well as the latent individual-level variables (whether each individual has been infected, etc.). Therefore, the fraction of individuals with (assumed) acquired immunity is also distributed according to the posterior. When the second-wave scenario starts for Lombardy and New York City, there is a low but non-zero level of circulation of the virus (approximately 100 actively infectious individuals). We initialize Hubei to a similar state, with 100 individuals newly infected at random when the scenario starts. In all locations, the modeled second-wave contact reductions are imposed at the start of the scenario when the number of infected individuals is low; this allows us to demonstrate the impact of the contact reductions as distinct from the cost of waiting to impose them. In order to simulate physical distancing by the entire population, we reduce the expected number of contacts M^s_{ij} between each pair of age groups i and j in setting s to $\delta_{\text{second}} M^s_{ij}$. Contacts within the household are unchanged. Our experiments examine $\delta_{\text{second}} \in [0.05, 0.25, 0.5, 0.75, 1]$. Physical distancing is also complemented by salutary sheltering by a single age group. For each age group, we simulate the impact of 25%, 50%, or 75% of the members of that age group sheltering (in addition to physical distancing at each level δ_{second} by the rest of the population). We run the simulation until the end of 2021 to ensure that the epidemic has had time to run its course in all scenarios and report the final median number of new infections and deaths for each scenario.

We also simulate a corresponding set of scenarios where the population begins in a completely susceptible state. For these simulations, population-level parameters are sampled from the posterior distribution as in the second-wave scenarios, but the population is initialized to be completely susceptible (apart from the randomly-sampled initially infectious individuals, as in the start of our experiments analyzing the first wave for each location). Contact reductions are imposed immediately, again to disentangle the impact of the reductions themselves on the number of expected infections from the cost of waiting to impose the intervention. We simulate the same set of combinations of physical distancing and salutary sheltering as above.

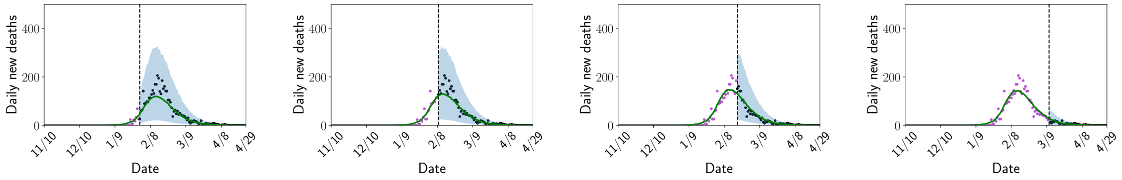


Figure G.1: *Predictive posterior for Hubei as a function of when the training period ends. Black dashed line: end of training period. Green line: posterior median. Blue shaded region: 90% credible interval. Pink dots: training data. Black dots: held-out data. The 90% credible interval of the predictive posterior contains the held-out data at all points, including when the model is fit using only data from the earliest portion of the epidemic.*

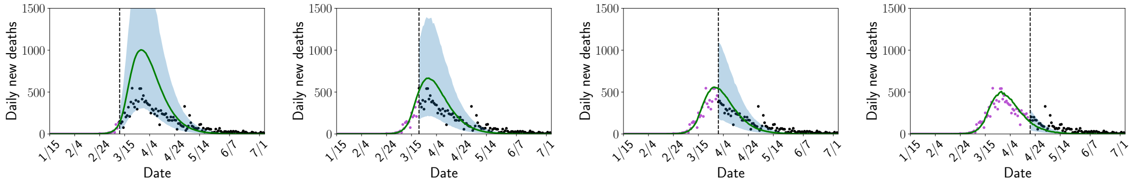


Figure G.2: *Predictive posterior for Lombardy as a function of when the training period ends. Black dashed line: end of training period. Green line: posterior median. Blue shaded region: 90% credible interval. Pink dots: training data. Black dots: held-out data. The 90% credible interval of the predictive posterior includes contains the held-out data at almost all points, including when the model is fit using only data from the earliest portion of the epidemic. The model over-predicts deaths early in the epidemic, though the timing of the peak is correctly captured early on. Much of the over-prediction is corrected with additional training data even before the peak is observed.*

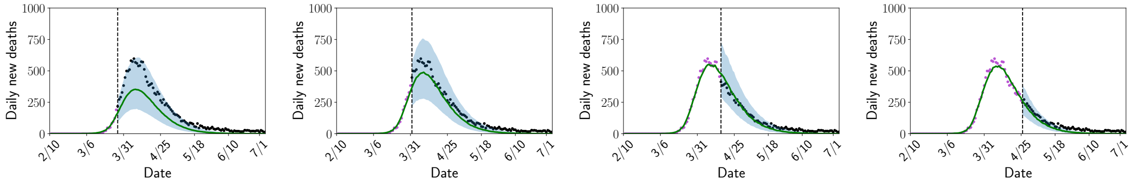


Figure G.3: *Predictive posterior for New York City as a function of when the training period ends. Black dashed line: end of training period. Green line: posterior median. Blue shaded region: 90% credible interval. Pink dots: training data. Black dots: held-out data. The 90% credible interval of the predictive posterior includes contains the held-out data at all points, including when the model is fit using only data from the earliest portion of the epidemic. Using data from only the earliest stage, the model slightly misidentifies the timing and magnitude of the peak, but these aspects of the prediction substantially improve even without observing the peak in the training data (c.f. the first vs second figure from the left).*

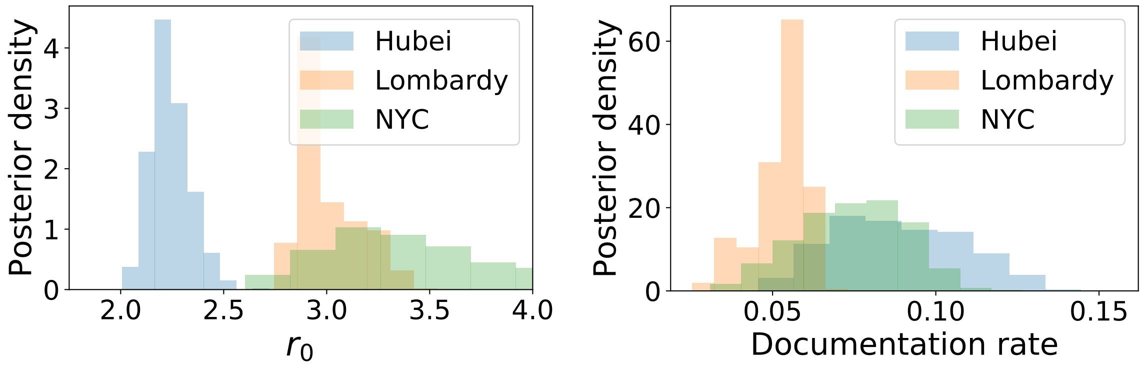


Figure G.4: *Sensitivity analysis to higher prevalence of comorbidities in the population of COVID-19 patients use to infer the age and comorbidity-specific infection fatality rate. Each plot shows the posterior distribution over a given quantity (left: the basic reproductive number r_0; right: the documentation rate for infections) for each location. In this scenario, the regression coefficients $\beta_{age}, \beta_{diabetes}, \beta_{hypertension}$ which produce $p_{m\to d}(a_i, c_i)$ (the probability of death given age a_i and comorbidities c_i) are estimated assuming that the prevalence of both diabetes and hypertension are twice as high in the COVID-19 patients in China for whom case fatality rates are available as in the general population for China. Our main analysis assumed equal prevalence in COVID-19 patients as in the general population. Our major conclusions are unaltered.*

Table G.1: *Model parameters*

Parameter	Description	Value and/or source
$p_{m\to s}(a_i, c_i)$	Prob. of progressing from mild to severe given age a_i and comorbidities c_i	Estimated from [VOD$^+$20] (see above)
$p_{s\to c}(a_i, c_i)$	Prob. of progressing from severe to critical given age a_i and comorbidities c_i	As above
$p_{c\to d}(a_i, c_i)$	Prob. of progressing from critical to death given age a_i and comorbidities c_i	As above
p_h	Prob. of infecting each household member each day	Calibrated to match [LEK20]
p_{inf}	Prob. of infecting an outside household contact	Free parameter
$\mu_{e\to m}$	Log-mean time to progress from exposed to mild (mean incubation period)	1.621 [LGB$^+$20]
$\sigma^2_{e\to m}$	Log-standard deviation time to progress from exposed to mild	0.418 [LGB$^+$20]
$\lambda_{m\to s}$	Mean time to progress from mild to severe	7 days [Chi20]
$\lambda_{s\to c}$	Mean time to progress from severe to critical	7.5 days (using 14.5 days from onset to mechanical ventilation in [ZYD$^+$20])
$\lambda_{c\to d}$	Mean time to progress from critical to death	4.5 days (subtracting $\lambda_{m\to s}$ and $\lambda_{s\to c}$ from onset-to-death in [ZYD$^+$20])
λ_{isolate}	Mean time for an individual in the mild state to isolate	4.6 days (time to first medical care [LGW$^+$20])
$\lambda_{m\to r}$	Mean time to recovery for an individual in the mild state	14 days [Chi20]
$\lambda_{s\to r}$	Mean time to recovery for an individual in the severe state	$28 - \lambda_{m\to s}$ (midpoint of onset-to-recovery for severe [Chi20])
$\lambda_{c\to r}$	Mean time to recovery for an individual in the critical state	$35 - \lambda_{m\to s} - \lambda_{s\to c}$ (midpoint of [Chi20] onset-to-recovery for critical)
α	Reduction in infectiousness before symptoms	0.55 [LPC$^+$20] [2]
M	Contact matrix (for each country)	[PCJ17]
t_0	First date with at least 5 infected individuals	Free parameter

[2] This setting for α is likely pessimistic in that Li et al.'s estimate for reduction in transmissibility is for undocumented cases, including asymptomatic cases, presymptomatic cases, and those with limited symptoms [LPC$^+$20]. Future work should examine the impact of a potentially lower α as better information on transmissibility in the asymptomatic or presymptomatic state becomes available.

Table G.2: *Comparison of Poisson and negative binomial observation models in each location, along with estimated dispersion parameter σ^2_{obs} for the negative binomial. The negative binomial model is strongly preferred by AIC in each location. G.2*

Location	Poisson AIC	Negative binomial AIC	σ^2_{obs}
Hubei	891.30	670.26	0.337
Lombardy	4741.82	877.97	0.278
New York City	657.49	533.26	0.0641

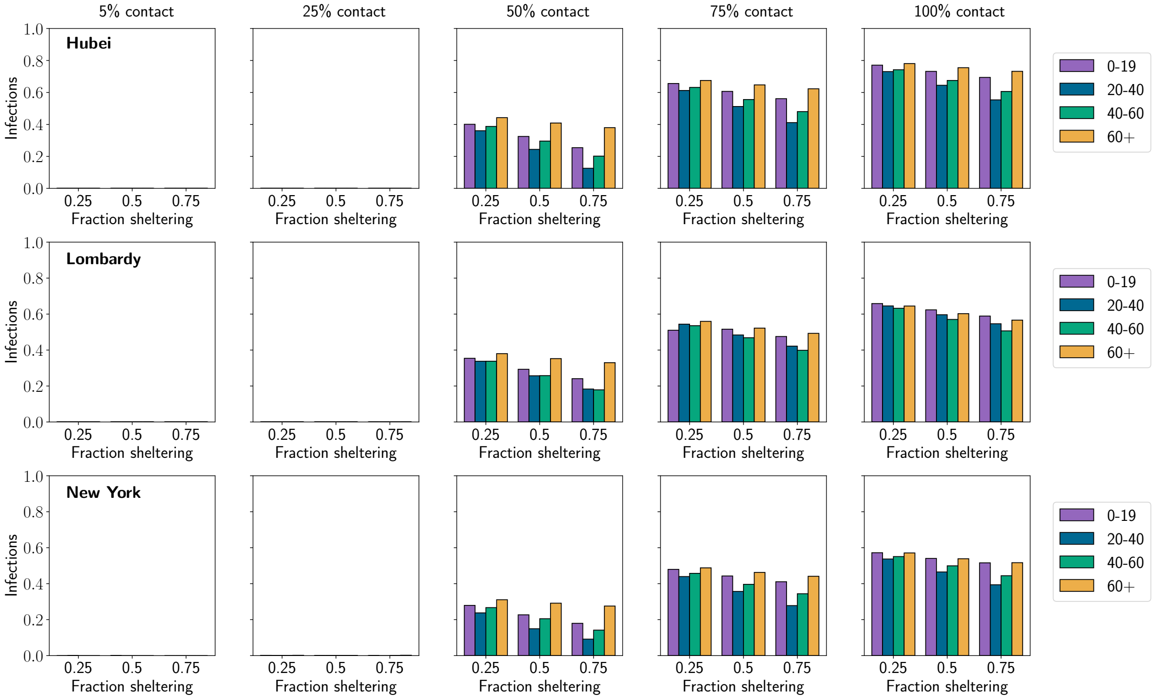


Figure G.5: *Fraction of the population newly infected in each location in each hypothetical second-wave scenario. Each row shows the results for the specified location, while each column shows a given level of physical distancing by the entire population (specified as the percentage of normal contact levels). The x-axis within each figure gives the fraction of a single age group which adopts salutary sheltering. Each bar represents a scenario where the given fraction of a single age group adopts salutary sheltering, with the color of the bar representing the identity of the group (see legend). We find that for all populations, 25% or less contact is sufficient to suppress the epidemic. At 50% contact, a significant portion of each population becomes infected (approximately 10-40% depending on the population, which group shelters, and what fraction of that group shelters). Across populations, sheltering by the 20-40 and 40-60 age groups reduces infections by the largest amount; sheltering by the 60+ group has only a minor impact.*

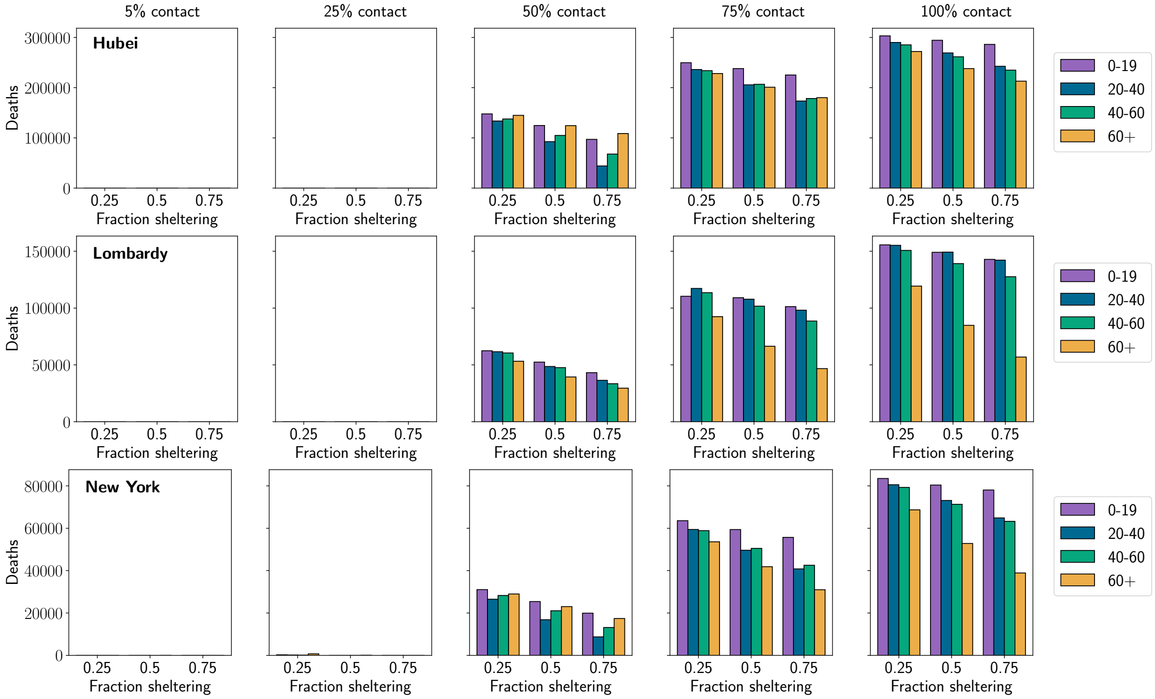


Figure G.6: *Number of deaths in each population in each hypothetical second-wave scenario. Each row shows the results for the specified location, while each column shows a given level of physical distancing by the entire population (specified as the percentage of normal contact levels). The x-axis within each figure gives the fraction of a single age group which adopts salutary sheltering. Each bar represents a scenario where the given fraction of a single age group adopts salutary sheltering, with the color of the bar representing the identity of the group (see legend). In scenarios with 25% or less contact, the outbreak is effectively surpressed (see Figure G.5) resulting in correspondingly few deaths. At 50% contact, the larger number of infections results in a larger number of deaths. For Hubei and New York City at 50% contact, deaths are reduced more effectively via sheltering by the 20-40 or 40-60 groups than by the 60+ group. In Lombardy, sheltering by the 60+ group is always the most effective at reducing deaths but the margin between the number of deaths under sheltering by the 60+ group compared to other groups is smaller under 50% contact than under higher contact levels. At 75% or higher contact, this pattern is replicated in Hubei and New York City, where sheltering by the 60+ group has the greatest marginal impact on deaths and the gap between the 60+ and other groups is larger at 100% contact than at 75%.*

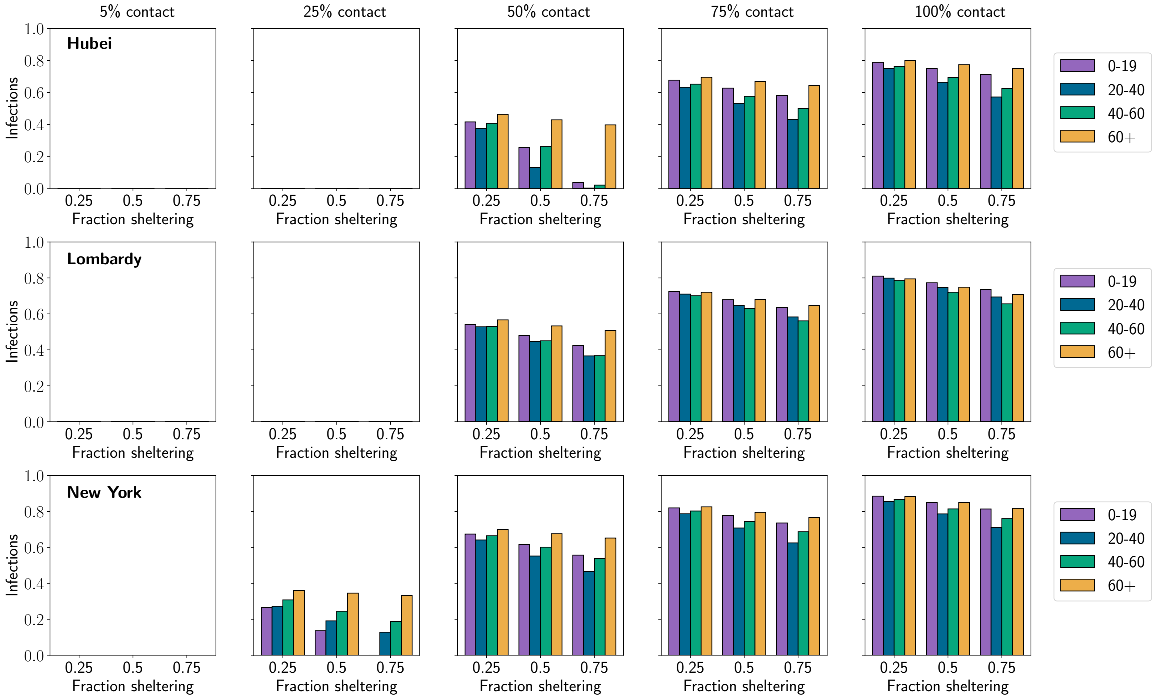


Figure G.7: *Fraction of the population infected in each population in each hypothetical scenario with a completely susceptible population. Each row shows the results for the specified location, while each column shows a given level of physical distancing by the entire population (specified as the percentage of normal contact levels). The x-axis within each figure gives the fraction of a single age group which adopts salutary sheltering. Each bar represents a scenario where the given fraction of a single age group adopts salutary sheltering, with the color of the bar representing the identity of the group (see legend). In contrast to the second-wave scenarios shown in Figure G.5, 25% contact is not always sufficient to suppress a widespread outbreak. This reflects two factors. First, the importance of acquired immunity accumulated during the first outbreak in reducing the effective reproduction number. Second, the potential for the timing of when interventions are applied to influence the total extent of infections. E.g., it is possible for the total number of eventual infections to be lower when more people are infected in the first wave than when more stringent control measures are imposed from the start [HLJA07]. At 50% contact and above, the dynamics become more similar to the second-wave scenarios, with substantial fractions of each population infected. As in Figure G.5, sheltering by the 20-40 and 40-60 age groups reduces infections by the largest amount; sheltering by the 60+ group has only a minor impact.*

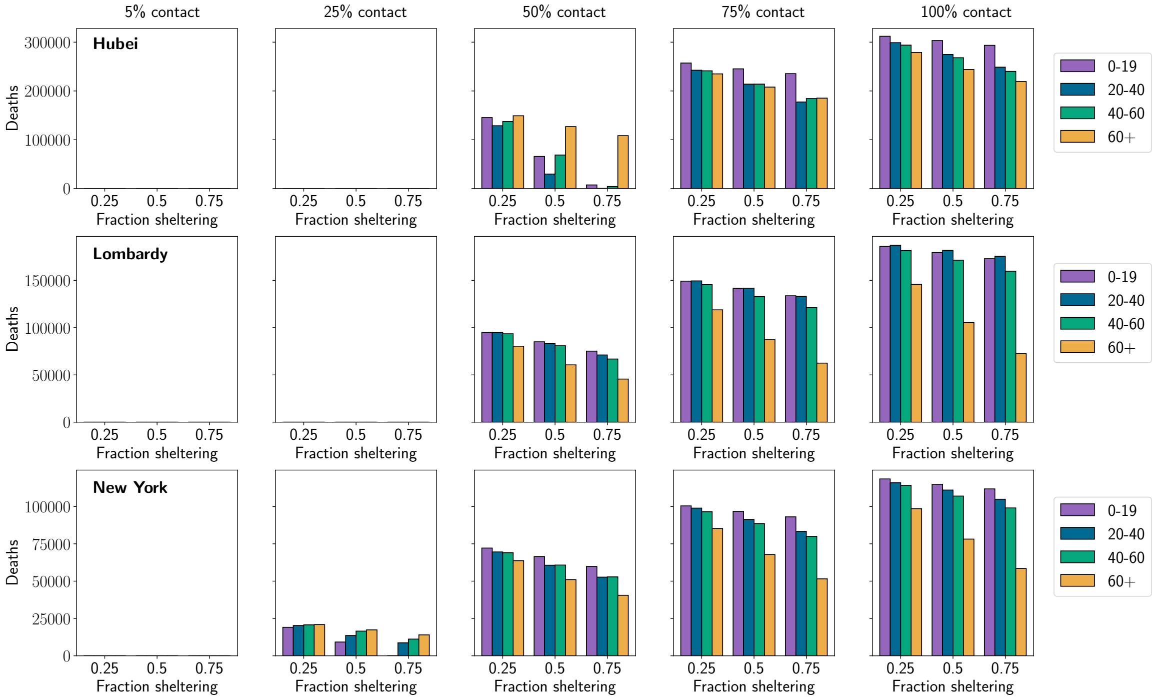


Figure G.8: *Number of deaths in each population in each hypothetical scenario with a completely susceptible population. Each row shows the results for the specified location, while each column shows a given level of physical distancing by the entire population (specified as the percentage of normal contact levels). The x-axis within each figure gives the fraction of a single age group which adopts salutary sheltering. Each bar represents a scenario where the given fraction of a single age group adopts salutary sheltering, with the color of the bar representing the identity of the group (see legend). Deaths are limited by contact levels at 25% or lower. As in the second-wave scenarios for Hubei and New York City, at low levels of contact, sheltering by the 20-40 and 40-60 age groups is more effective at reducing deaths than sheltering by the 60+ group. However, due to the larger number of infections at a given contact level in the completely-susceptible scenario as compared to the second-wave scenario, a lower level of overall contact is sometimes needed to realize this effect (25% contact in New York City instead of 50%). Once contact levels rise to 50%, only Hubei shows greater effectiveness for sheltering by the 20-40 and 40-60 age groups, and at 75% contact it is more effective for the 60+ age group to shelter for all populations.*

Table G.3: *Infections (in thousands) for a second-wave scenario in Hubei. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total infections (in thousands) in each scenario, while "0-59" and "60+" give the median number of total infections in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.2	0.2	0.0	1.0	1.0	0.1	23445.4	20439.3	3006.0	38372.9	33291.2	5081.7	45077.8	38866.5	6211.4
50%	0.2	0.2	0.0	0.6	0.5	0.1	19004.6	16514.2	2490.4	35487.6	30666.7	4820.9	42811.7	36778.4	6033.2
75%	0.2	0.2	0.0	0.6	0.5	0.1	14888.0	12894.7	1993.4	32813.8	28264.0	4549.8	40609.1	34761.2	5847.8
20-40															
25%	0.2	0.2	0.0	0.6	0.5	0.1	21084.8	18314.8	2770.0	35816.7	30952.8	4864.0	42720.3	36677.2	6043.2
50%	0.2	0.2	0.0	0.5	0.4	0.1	14269.7	12289.0	1980.7	29955.1	25636.9	4318.2	37719.7	32089.5	5630.2
75%	0.2	0.2	0.0	0.5	0.4	0.1	7348.9	6332.3	1016.7	24057.7	20378.1	3679.6	32381.1	27224.4	5156.7
40-60															
25%	0.2	0.2	0.0	0.6	0.5	0.1	22662.4	19752.0	2910.3	36943.6	32022.0	4921.6	43409.4	37362.4	6046.9
50%	0.2	0.2	0.0	0.6	0.6	0.1	17298.6	15009.9	2288.7	32505.8	28035.0	4470.8	39484.8	33801.4	5683.5
75%	0.2	0.2	0.0	0.5	0.4	0.1	11798.9	10240.3	1558.6	28086.9	24099.4	3987.5	35458.4	30183.9	5274.5
60+															
25%	0.2	0.2	0.0	0.5	0.4	0.1	25872.8	22997.9	2874.9	39481.7	35000.5	4481.2	45658.2	40315.6	5342.6
50%	0.2	0.2	0.0	0.7	0.7	0.1	23916.2	21538.6	2377.6	37869.2	34094.9	3774.3	44153.0	39680.5	4472.4
75%	0.2	0.2	0.0	0.5	0.4	0.1	22211.0	20213.5	1997.6	36441.5	33227.7	3213.8	42824.5	39048.6	3776.0

Table G.4: *Infections (in thousands) for a second-wave scenario in Lombardy. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total infections (in thousands) in each scenario, while "0-59" and "60+" give the median number of total infections in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.0	0.0	0.7	0.7	0.0	3540.9	2822.6	718.2	5101.1	3899.6	1201.4	6584.6	4945.4	1639.2
50%	0.0	0.0	0.0	0.3	0.3	0.0	2930.2	2319.9	610.3	5159.4	3955.0	1204.4	6234.3	4650.6	1583.6
75%	0.0	0.0	0.0	0.1	0.1	0.0	2409.4	1894.3	515.2	4753.2	3621.3	1132.0	5890.6	4363.8	1526.8
20-40															
25%	0.0	0.0	0.0	0.2	0.2	0.0	3376.0	2685.4	690.6	5440.0	4178.6	1261.4	6453.4	4823.1	1630.2
50%	0.0	0.0	0.0	0.2	0.2	0.0	2567.9	2014.8	553.2	4838.4	3671.2	1167.2	5962.5	4398.2	1564.3
75%	0.0	0.0	0.0	0.2	0.1	0.0	1832.9	1417.9	415.0	4218.4	3157.0	1061.4	5460.6	3963.9	1496.6
40-60															
25%	0.1	0.1	0.0	0.5	0.5	0.0	3377.9	2688.8	689.1	5353.9	4117.8	1236.2	6320.6	4724.2	1596.3
50%	0.0	0.0	0.0	0.3	0.3	0.0	2578.5	2028.5	550.0	4683.3	3565.5	1117.8	5702.0	4210.4	1491.6
75%	0.0	0.0	0.0	0.1	0.1	0.0	1786.5	1388.5	398.0	3985.8	2997.1	988.7	5065.0	3691.0	1374.0
60+															
25%	0.1	0.1	0.0	0.3	0.3	0.0	3800.9	3192.0	608.9	5593.7	4580.7	1013.0	6448.3	5170.1	1278.2
50%	0.1	0.1	0.0	0.4	0.3	0.0	3524.4	3069.0	455.5	5219.7	4484.1	735.6	6022.9	5104.1	918.8
75%	0.0	0.0	0.0	0.3	0.3	0.0	3297.4	2965.2	332.2	4927.6	4406.4	521.2	5664.5	5036.6	627.8

Table G.5: *Infections (in thousands) for a second-wave scenario in New York City. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total infections (in thousands) in each scenario, while "0-59" and "60+" give the median number of total infections in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.0	0.0	12.4	0.1	0.0	2345.5	2020.6	336.1	4028.3	3376.4	678.1	4805.6	3948.3	895.2
50%	0.0	0.0	0.0	0.1	0.1	0.0	1906.2	1587.8	283.6	3718.9	3103.3	639.2	4539.5	3735.0	861.1
75%	0.0	0.0	0.0	0.1	0.0	0.0	1509.2	1194.1	236.9	3451.1	2875.9	607.4	4333.1	3529.0	827.9
20-40															
25%	0.0	0.0	0.0	0.1	0.1	0.0	1995.9	1752.2	293.6	3692.0	3072.5	640.8	4511.5	3678.4	869.5
50%	0.0	0.0	0.0	0.2	0.0	0.0	1256.4	1101.8	192.7	2996.8	2477.3	550.3	3906.4	3153.7	796.9
75%	0.0	0.0	0.0	0.0	0.0	0.0	770.0	484.4	90.0	2335.8	1907.2	449.4	3309.8	2605.5	716.1
40-60															
25%	0.0	0.0	0.0	3.5	0.1	0.0	2242.3	1966.4	317.5	3841.8	3218.4	649.5	4625.7	3788.1	868.9
50%	0.0	0.0	0.0	8.1	0.1	0.0	1723.8	1510.8	242.8	3329.8	2821.9	573.5	4194.0	3418.0	800.5
75%	0.0	0.0	0.0	0.2	0.0	0.0	1190.7	1068.7	172.2	2890.6	2420.0	494.5	3732.4	3033.8	727.8
60+															
25%	0.0	0.0	0.0	15.0	0.2	0.0	2610.8	2350.1	296.3	4098.8	3587.1	547.6	4794.9	4118.3	709.2
50%	0.0	0.0	0.0	7.8	0.2	0.0	2449.3	2277.8	216.4	3885.0	3525.2	395.4	4525.1	4085.0	509.3
75%	0.0	0.0	0.0	17.5	0.2	0.0	2315.0	2222.5	146.3	3706.2	3489.7	261.3	4340.5	4054.0	328.7

Table G.6: *Infections (in thousands) for a fully susceptible population in Hubei. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total infections (in thousands) in each scenario, while "0-59" and "60+" give the median number of total infections in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.0	0.0	0.0	0.0	0.0	24296.0	21144.8	3151.2	39589.1	34295.6	5290.0	46137.4	39735.7	6400.5
50%	0.0	0.0	0.0	0.0	0.0	0.0	14864.2	12914.7	1925.4	36673.5	31651.5	5021.6	43865.4	37640.4	6226.8
75%	0.0	0.0	0.0	0.0	0.0	0.0	2139.0	1864.0	275.0	33972.3	29221.6	4750.7	41671.1	35624.3	6045.5
20-40															
25%	0.0	0.0	0.0	0.0	0.0	0.0	21874.8	18960.6	2911.8	36995.8	31921.2	5074.6	43873.1	37635.5	6237.6
50%	0.0	0.0	0.0	0.0	0.0	0.0	7641.0	6596.3	1042.1	31112.7	26597.1	4515.6	38817.0	32984.3	5833.9
75%	0.0	0.0	0.0	0.0	0.0	0.0	76.2	65.3	10.9	25148.4	21264.5	3884.0	33425.8	28073.7	5352.1
40-60															
25%	0.0	0.0	0.0	0.0	0.0	0.0	23800.0	20699.0	3101.1	38133.0	33002.5	5130.0	44565.3	38323.4	6242.3
50%	0.0	0.0	0.0	0.0	0.0	0.0	15228.0	13203.5	2024.6	33698.1	29020.9	4677.9	40584.2	34705.1	5879.5
75%	0.0	0.0	0.0	0.0	0.0	0.0	1167.4	1012.5	154.8	29187.1	25007.1	4182.1	36522.3	31045.5	5474.0
60+															
25%	0.0	0.0	0.0	0.0	0.0	0.0	27094.9	24043.3	3055.5	40666.9	35999.9	4668.2	46742.0	41214.4	5525.9
50%	0.0	0.0	0.0	0.0	0.0	0.0	25051.3	22519.2	2532.0	39071.9	35121.5	3948.6	45234.7	40594.2	4640.5
75%	0.0	0.0	0.0	0.0	0.0	0.0	23206.5	21080.0	2125.5	37662.2	34283.1	3379.0	43916.8	39987.2	3928.6

Table G.7: *Infections (in thousands) for a fully susceptible population in Lombardy. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total infections (in thousands) in each scenario, while "0-59" and "60+" give the median number of total infections in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.0	0.0	0.4	0.3	0.1	5404.4	4367.2	1037.1	7237.4	5651.9	1585.5	8101.9	6172.4	1929.4
50%	0.0	0.0	0.0	0.2	0.2	0.0	4797.6	3860.1	937.5	6789.2	5274.0	1515.2	7730.4	5854.6	1875.8
75%	0.0	0.0	0.0	0.2	0.1	0.0	4232.5	3394.8	837.7	6351.7	4909.7	1442.0	7362.5	5542.4	1820.1
20-40															
25%	0.0	0.0	0.0	0.7	0.6	0.1	5281.9	4257.8	1024.1	7102.3	5525.1	1577.2	7991.1	6061.2	1929.9
50%	0.0	0.0	0.0	0.3	0.2	0.1	4459.5	3569.8	889.7	6477.8	4988.6	1489.2	7478.7	5608.5	1870.2
75%	0.0	0.0	0.0	0.1	0.1	0.0	3661.7	2906.6	755.1	5830.4	4439.6	1390.8	6941.8	5135.0	1806.9
40-60															
25%	0.0	0.0	0.0	0.4	0.3	0.1	5291.7	4270.3	1021.3	7009.6	5460.7	1548.8	7846.8	5953.5	1893.3
50%	0.0	0.0	0.0	0.2	0.2	0.0	4502.3	3610.8	891.5	6307.4	4878.0	1429.5	7207.5	5413.4	1794.1
75%	0.0	0.0	0.0	0.2	0.2	0.0	3672.9	2927.1	745.8	5611.5	4303.2	1308.4	6558.6	4870.6	1688.0
60+															
25%	0.0	0.0	0.0	1.4	1.2	0.2	5666.3	4782.8	883.5	7206.9	5933.0	1273.9	7947.3	6413.2	1534.1
50%	0.0	0.0	0.0	1.2	1.0	0.1	5331.1	4657.6	673.5	6804.6	5851.6	953.1	7486.4	6352.8	1133.6
75%	0.0	0.0	0.0	1.1	1.0	0.1	5069.1	4559.4	509.7	6469.1	5778.6	690.5	7088.0	6293.4	794.6

Table G.8: *Infections (in thousands) for a fully susceptible population in New York City. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total infections (in thousands) in each scenario, while "0-59" and "60+" give the median number of total infections in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.1	0.0	2227.5	1988.4	239.2	5661.5	4875.9	785.8	6883.3	5796.8	1086.7	7433.6	6172.3	1261.7
50%	0.0	0.1	0.0	1142.8	1016.2	126.7	5179.5	4450.8	728.9	6532.9	5487.2	1045.9	7140.4	5907.8	1233.0
75%	0.0	0.1	0.0	1.6	1.5	0.2	4677.4	4010.6	667.0	6178.6	5174.0	1004.8	6833.1	5633.0	1200.5
20-40															
25%	0.0	0.1	0.0	2287.6	2045.7	242.0	5385.0	4626.6	758.6	6608.1	5540.0	1068.3	7186.9	5936.3	1251.0
50%	0.0	0.1	0.0	1603.4	1434.4	169.1	4635.2	3958.7	676.6	5944.9	4942.4	1002.8	6603.3	5399.4	1204.2
75%	0.0	0.1	0.0	1075.4	962.8	112.7	3908.1	3323.9	584.3	5247.3	4321.6	925.9	5962.6	4815.2	1147.7
40-60															
25%	0.0	0.1	0.0	2583.7	2314.0	269.8	5581.2	4814.0	767.4	6738.8	5672.3	1066.8	7282.6	6036.5	1246.5
50%	0.0	0.1	0.0	2055.8	1845.9	210.0	5049.1	4353.7	695.6	6256.0	5251.8	1004.4	6836.2	5641.0	1195.5
75%	0.0	0.1	0.0	1566.9	1411.0	155.5	4525.0	3904.7	620.4	5769.3	4834.9	934.6	6378.8	5240.0	1139.2
60+															
25%	0.0	0.1	0.0	3024.7	2759.0	265.9	5876.7	5218.4	658.5	6932.3	6047.8	884.7	7414.8	6393.5	1021.9
50%	0.0	0.1	0.0	2901.6	2695.9	205.7	5675.6	5177.9	497.9	6679.6	6018.9	661.1	7133.6	6373.8	760.3
75%	0.0	0.1	0.0	2786.6	2632.8	153.9	5475.6	5124.3	351.4	6439.2	5987.3	452.2	6865.7	6353.9	512.3

Table G.9: *Deaths (in thousands) for a second-wave scenario in Hubei. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total deaths (in thousands) in each scenario, while "0-59" and "60+" give the median number of total deaths in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.0	0.0	0.0	0.0	0.0	147.8	29.4	118.1	249.7	50.4	198.2	303.2	61.2	239.6
50%	0.0	0.0	0.0	0.0	0.0	0.0	124.7	24.7	97.7	238.0	48.6	186.7	294.4	60.0	232.2
75%	0.0	0.0	0.0	0.0	0.0	0.0	97.1	19.8	76.9	225.2	45.9	175.9	286.2	58.7	225.4
20-40															
25%	0.0	0.0	0.0	0.0	0.0	0.0	133.6	24.9	108.0	236.2	45.4	188.2	290.0	56.3	230.7
50%	0.0	0.0	0.0	0.0	0.0	0.0	92.5	16.0	75.3	205.7	37.7	164.8	269.2	50.5	216.6
75%	0.0	0.0	0.0	0.0	0.0	0.0	44.1	6.8	35.5	173.4	29.9	142.2	242.6	42.9	197.2
40-60															
25%	0.0	0.0	0.0	0.0	0.0	0.0	137.7	24.3	113.3	233.9	41.9	191.6	285.3	50.8	231.5
50%	0.0	0.0	0.0	0.0	0.0	0.0	104.9	16.3	87.6	206.7	32.4	170.5	261.5	40.9	217.6
75%	0.0	0.0	0.0	0.0	0.0	0.0	67.8	9.1	57.1	178.5	24.2	151.4	234.9	31.5	202.8
60+															
25%	0.0	0.0	0.0	0.0	0.0	0.0	144.9	30.8	113.3	228.2	50.2	176.3	272.0	60.8	209.9
50%	0.0	0.0	0.0	0.0	0.0	0.0	124.3	28.5	95.6	200.9	48.6	151.5	237.9	58.9	177.6
75%	0.0	0.0	0.0	0.0	0.0	0.0	108.7	26.3	81.7	180.1	46.6	131.7	212.9	57.3	154.4

Table G.10: *Deaths (in thousands) for a second-wave scenario in Lombardy. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total deaths (in thousands) in each scenario, while "0-59" and "60+" give the median number of total deaths in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.0	0.0	0.0	0.0	0.0	62.4	3.1	62.6	110.4	4.6	107.8	155.6	5.9	152.2
50%	0.0	0.0	0.0	0.0	0.0	0.0	52.5	2.6	52.3	109.1	4.8	107.6	149.1	5.7	146.3
75%	0.0	0.0	0.0	0.0	0.0	0.0	43.2	2.2	43.8	101.3	4.4	100.7	142.8	5.5	139.4
20-40															
25%	0.0	0.0	0.0	0.0	0.0	0.0	61.6	2.8	61.3	117.3	4.7	114.5	155.2	5.6	153.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	48.6	2.2	49.3	107.8	4.1	107.0	149.2	5.2	147.1
75%	0.0	0.0	0.0	0.0	0.0	0.0	36.4	1.6	37.3	98.1	3.7	98.3	142.2	4.8	141.6
40-60															
25%	0.0	0.0	0.0	0.0	0.0	0.0	60.5	2.6	60.1	113.5	4.1	112.3	150.7	4.9	148.8
50%	0.0	0.0	0.0	0.0	0.0	0.0	47.6	1.8	48.0	101.7	3.2	100.8	139.1	4.1	138.4
75%	0.0	0.0	0.0	0.0	0.0	0.0	33.4	1.2	34.3	88.6	2.4	88.9	127.6	3.0	127.7
60+															
25%	0.0	0.0	0.0	0.0	0.0	0.0	53.2	3.3	52.3	92.4	4.9	89.6	119.3	5.8	115.9
50%	0.0	0.0	0.0	0.0	0.0	0.0	39.4	3.1	37.9	66.4	4.8	63.0	84.8	5.7	79.8
75%	0.0	0.0	0.0	0.0	0.0	0.0	29.6	3.0	27.0	46.8	4.6	42.5	56.9	5.6	51.9

Table G.11: *Deaths (in thousands) for a second-wave scenario in New York City. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total deaths (in thousands) in each scenario, while "0-59" and "60+" give the median number of total deaths in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.0	0.0	0.3	0.0	0.0	31.0	6.0	24.0	63.6	11.0	51.4	83.5	13.5	70.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	25.4	4.9	19.5	59.4	10.6	47.9	80.4	13.3	67.0
75%	0.0	0.0	0.0	0.0	0.0	0.0	19.9	4.1	17.3	55.7	10.2	44.6	78.1	13.0	63.7
20-40															
25%	0.0	0.0	0.0	0.1	0.0	0.0	26.5	4.9	19.6	59.4	9.7	48.6	80.5	12.7	67.4
50%	0.0	0.0	0.0	0.0	0.0	0.0	16.8	2.9	14.9	49.6	8.4	41.3	73.1	11.2	61.4
75%	0.0	0.0	0.0	0.0	0.0	0.0	8.7	1.5	7.2	40.8	6.5	32.8	64.9	9.4	54.5
40-60															
25%	0.0	0.0	0.0	0.1	0.0	0.0	28.3	4.9	22.1	58.9	9.0	49.2	79.3	11.1	67.3
50%	0.0	0.0	0.0	0.0	0.0	0.0	21.1	3.1	16.8	50.5	6.7	42.6	71.3	8.6	62.0
75%	0.0	0.0	0.0	0.0	0.0	0.0	13.2	1.8	13.0	42.5	5.1	36.3	63.3	6.3	56.0
60+															
25%	0.0	0.0	0.0	0.7	0.0	0.0	29.0	6.8	20.8	53.6	11.2	40.6	68.7	13.6	54.2
50%	0.0	0.0	0.0	0.1	0.0	0.0	23.0	6.5	14.7	41.9	11.0	29.0	52.8	13.5	37.6
75%	0.0	0.0	0.0	0.0	0.0	0.0	17.4	6.2	9.9	31.0	10.6	18.9	38.9	13.6	23.2

Table G.12: *Deaths (in thousands) for a fully susceptible population in Hubei. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total deaths (in thousands) in each scenario, while "0-59" and "60+" give the median number of total deaths in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.0	0.0	0.0	0.0	0.0	145.4	28.7	116.6	257.3	52.1	205.3	312.0	63.1	249.4
50%	0.0	0.0	0.0	0.0	0.0	0.0	65.8	13.5	51.8	245.3	50.3	195.7	303.3	61.8	241.7
75%	0.0	0.0	0.0	0.0	0.0	0.0	7.4	1.5	5.9	235.5	48.6	186.4	293.5	60.5	233.0
20-40															
25%	0.0	0.0	0.0	0.0	0.0	0.0	128.7	24.2	104.9	242.5	47.1	195.5	299.0	58.7	240.7
50%	0.0	0.0	0.0	0.0	0.0	0.0	29.8	5.2	24.6	214.0	39.7	174.3	274.8	51.6	223.2
75%	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.0	0.2	177.5	31.1	146.6	248.8	44.9	204.2
40-60															
25%	0.0	0.0	0.0	0.0	0.0	0.0	137.3	24.3	113.1	241.3	43.8	197.6	293.9	53.0	241.4
50%	0.0	0.0	0.0	0.0	0.0	0.0	68.8	10.7	58.1	214.0	34.0	180.1	268.1	42.4	225.6
75%	0.0	0.0	0.0	0.0	0.0	0.0	4.2	0.6	3.6	184.4	25.3	159.0	240.1	32.3	207.9
60+															
25%	0.0	0.0	0.0	0.0	0.0	0.0	149.2	31.7	117.4	235.0	52.0	183.1	278.9	62.2	216.6
50%	0.0	0.0	0.0	0.0	0.0	0.0	126.9	29.1	97.7	208.0	50.1	157.7	244.0	60.5	183.6
75%	0.0	0.0	0.0	0.0	0.0	0.0	108.4	26.3	82.1	185.3	48.2	137.5	219.3	59.1	159.4

Table G.13: *Deaths (in thousands) for a fully susceptible population in Lombardy. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total deaths (in thousands) in each scenario, while "0-59" and "60+" give the median number of total deaths in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.0	0.0	0.0	0.0	0.0	95.1	4.6	90.6	149.2	6.2	143.1	186.0	7.0	178.8
50%	0.0	0.0	0.0	0.0	0.0	0.0	85.0	4.1	81.0	141.7	5.9	135.6	179.5	6.9	172.6
75%	0.0	0.0	0.0	0.0	0.0	0.0	75.2	3.7	71.5	133.8	5.8	127.9	173.0	6.8	166.2
20-40															
25%	0.0	0.0	0.0	0.0	0.0	0.0	94.9	4.3	90.6	149.5	5.9	143.5	187.1	6.8	180.3
50%	0.0	0.0	0.0	0.0	0.0	0.0	83.3	3.6	79.7	141.8	5.4	136.4	181.8	6.4	175.3
75%	0.0	0.0	0.0	0.0	0.0	0.0	71.1	2.9	68.1	133.1	4.9	128.3	175.5	6.0	169.7
40-60															
25%	0.0	0.0	0.0	0.0	0.0	0.0	93.5	3.9	89.6	145.5	5.3	140.2	181.6	6.1	175.5
50%	0.0	0.0	0.0	0.0	0.0	0.0	80.9	3.0	77.8	132.9	4.3	128.7	171.4	5.0	166.4
75%	0.0	0.0	0.0	0.0	0.0	0.0	66.8	2.2	64.6	121.2	3.4	117.7	159.8	3.9	155.9
60+															
25%	0.0	0.0	0.0	0.0	0.0	0.0	80.4	4.7	75.6	118.9	6.2	112.6	145.8	7.0	138.7
50%	0.0	0.0	0.0	0.0	0.0	0.0	60.6	4.5	56.2	87.2	6.0	81.2	105.4	6.9	98.4
75%	0.0	0.0	0.0	0.0	0.0	0.0	45.6	4.4	41.1	62.5	5.9	56.6	72.4	6.7	65.6

Table G.14: *Deaths (in thousands) for a fully susceptible population in New York City. Each major row heading denotes the age group which adopts salutary sheltering, and the sub-headings denote the fraction of the group which shelters. The major column headings give the level of contact amongst individuals who do not shelter. The entry "Total" gives the median number of total deaths (in thousands) in each scenario, while "0-59" and "60+" give the median number of total deaths in each segment of the population (under or over 60 years of age).*

	5% contact			25% contact			50% contact			75% contact			100% contact		
	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+	Total	0-59	60+
0-19															
25%	0.0	0.0	0.0	19.1	4.5	14.4	72.2	14.7	57.0	100.4	18.7	81.7	118.5	20.6	97.9
50%	0.0	0.0	0.0	9.2	2.2	7.0	66.5	14.1	52.6	96.7	18.3	78.5	114.9	20.3	94.6
75%	0.0	0.0	0.0	0.0	0.0	0.0	59.9	13.0	47.5	93.1	18.1	75.0	111.8	20.1	91.7
20-40															
25%	0.0	0.0	0.0	20.3	4.4	15.9	69.6	13.7	55.7	98.9	17.8	81.1	115.9	19.5	96.4
50%	0.0	0.0	0.0	13.6	3.1	10.6	60.7	12.0	48.7	91.3	16.0	75.3	111.0	18.4	92.6
75%	0.0	0.0	0.0	8.7	1.9	6.7	52.7	9.9	42.6	83.4	14.3	69.1	104.9	17.0	87.8
40-60															
25%	0.0	0.0	0.0	20.8	4.2	16.6	69.1	13.0	56.2	96.5	16.0	80.5	114.2	17.8	96.4
50%	0.0	0.0	0.0	16.5	3.1	13.0	60.8	10.3	50.5	88.6	13.1	75.5	107.0	14.6	92.4
75%	0.0	0.0	0.0	11.2	2.2	9.1	52.9	8.0	44.7	80.0	10.4	69.6	99.0	11.6	87.4
60+															
25%	0.0	0.0	0.0	20.9	5.6	15.7	63.7	15.3	48.1	85.3	18.7	66.6	98.5	20.4	78.2
50%	0.0	0.0	0.0	17.4	5.4	12.2	51.1	15.0	36.0	67.8	18.7	49.2	78.2	20.5	57.7
75%	0.0	0.0	0.0	14.0	5.1	9.0	40.5	15.0	25.5	51.5	18.3	33.3	58.5	20.4	38.1

Appendix H

Appendix to Chapter 13

H.1 Details of experiments

H.1.1 Disease and observation models

To parameterize a setting approximately based on COVID-19, we set the parameter w (which determines the distribution of the time when an infected individual causes secondary infections) to be uniform over days 3-10 post-infection. The exact time dynamics of infectiousness and viral load after SARS-COV-2 infection have not yet been precisely defined, so a uniform distribution represents a parsimonious choice for a scenario where individuals become infectious approximately two days before symptom onset and remain infectious for a week, which is consistent with available evidence [WRC+20].

To model the distribution D, we drew on existing sources. Specifically, we used the probability of PCR positivity by time since infection reported in [KLL+20] to construct the PCR distribution. For the serological case, we used the distribution of the time to IgG seroconversion from symptom onset reported in [IJN+20]. To model the time to symptom onset, we added a random draw from the log-normal distribution which [LGB+20] report provides a close fit to the time to symptom onset after SARS-COV-2 infection. For the uniform underreporting model, we set the distribution of delay after conversion to be uniform over [0, 5] days.

H.1.2 Generating the ground truth R_t

The outbreak setting models a population where the spread of the disease is initially controlled, but where a new outbreak (i.e., $R > 1$) is possible. In each of these instances, R_0 starts at a uniformly random value in [0.2, 0.8], with γ uniformly random in [5, 15]. Then, at time t, uniformly distributed over days [20, 60], R starts to rise to a value uniformly distributed in [1.2, 2]. This transition happens linearly over a number of days which is uniformly distributed in [1,20], simulating either sharper or more gradual transitions. Finally, we perturb the entire time series with random noise distributed as $0.05 \cdot \mathcal{GP}(0, \mathcal{K})$ to add small day-to-day fluctuations.

Our second setting, the random trend setting, which models a case where R switches multiple times between different trends. Specifically, $R_0 = 0.8$. Then, for $t \in \{15, 40, 60, 85\}$, we draw R_t uniformly at random in [0, 2], and set $R_{100} = R_{85}$. The full time series is piecewise linear between these points. Finally, we add random GP perturbations and a random γ as in the outbreak model.

H.1.3 Parameter settings for GPRt

For the kernel $\mathcal{K}$, we used a Matern kernel with $\nu = \frac{3}{2}$ and a length scale of 20 days. These parameters were not tuned. We used a batch size of $b = 800$, and optimized μ and Σ using the Adam optimizer with learning rate 10^{-3} for 7000 iterations. We also tested batch sizes $b = \{100, 400\}$, a learning rate of $\{10^{-2}, 5 \cdot 10^{-2}\}$, and $\{1000, 2000, 10000\}$ iterations. The final settings were selected based on the final ELBO value on one randomly generated instance for the longitudinal setting with $d = 14$ and 1% of the population tested. This random instance was not included in the set of 100 instances used to generate the results in the main paper. On instances for the uniform underreporting model with more than 5% of the population tested, we observed that the ELBO had not converged after 7000 iterations and ran 14,000 iterations instead.

H.1.4 Computational setup for experiments

We ran all experiments on nodes with two Intel 8268 "Cascade Lake" processors and 192 GB of memory per node. Our program was constrained to use 10 cores and 1 GB of memory. GPRt finished running within 3 hours. The longest-running baseline was EpiNow, which finished within 1 hour (drawing 1000 samples using 4 parallel MCMC chains). GPRt was implemented using PyTorch 1.2 for autodifferentiation. The disease model was simulated using Numba version 0.48. The random number generator was seeded at the start of each of the 100 runs using the index of that run (e.g., the 5th run was seeded with “5"). For the EpiNow baseline, we used the development version of the EpiNow2 package available on Github as of 9/6/20 [AHH$^+$20]. For the WT and Cori baselines, we used the EpiEstim R package, version 2.2-3.

H.2 Additional experimental results

H.2.1 Table 1 with serological testing

H.2.2 Varying d for longitudinal testing

H.2.3 Calibration results

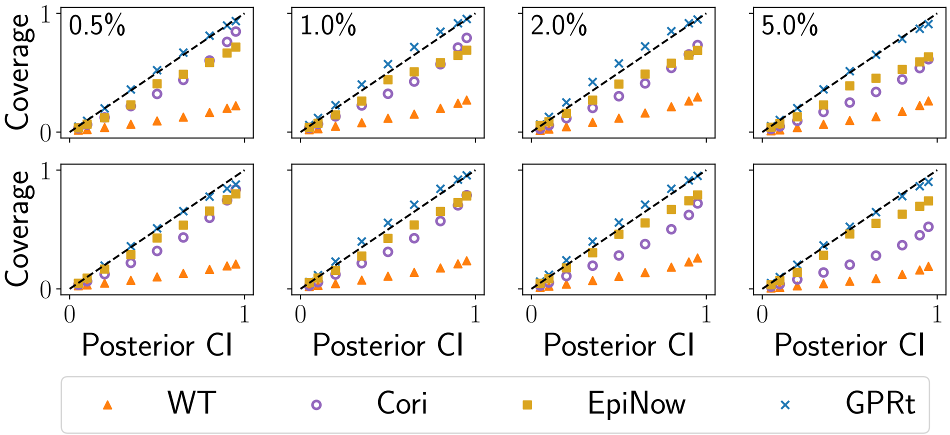


Figure H.1: *Calibration for the outbreak setting, longitudinal sampling, $d = 14$.*

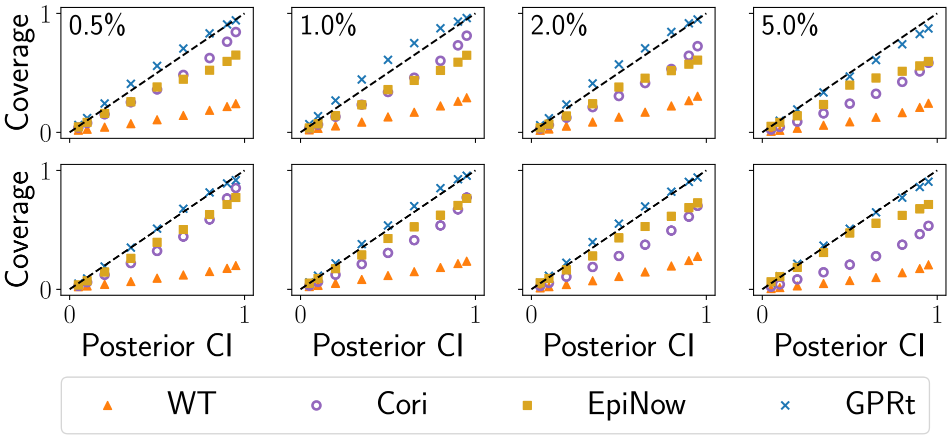


Figure H.2: *Calibration for the outbreak setting, longitudinal sampling, $d = 7$.*

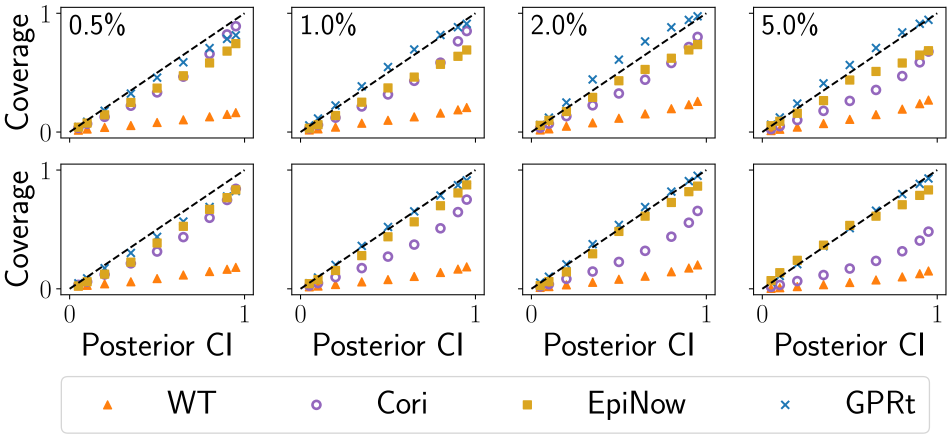


Figure H.3: *Calibration for the outbreak setting, longitudinal sampling, $d = 31$.*

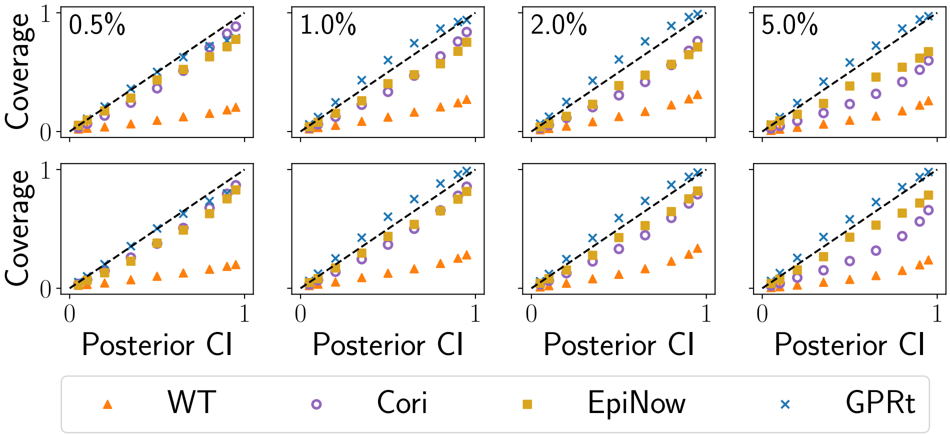


Figure H.4: *Calibration for the random trend setting, longitudinal sampling, $d = 14$.*

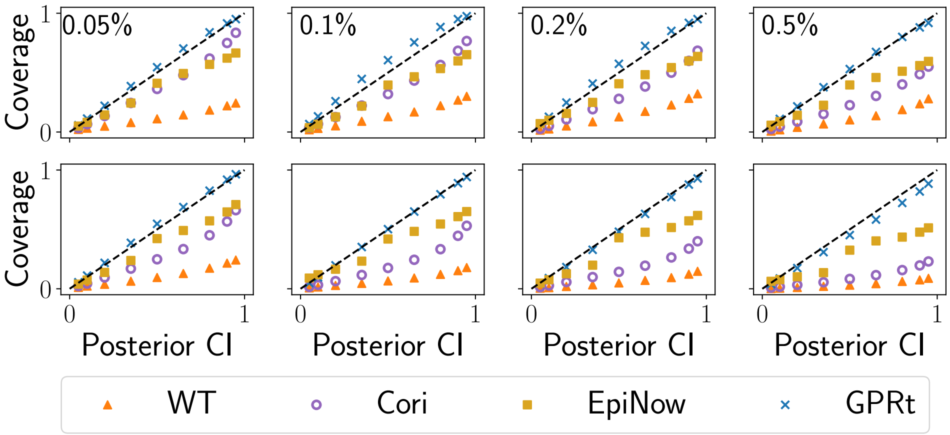


Figure H.5: *Calibration for the outbreak setting, cross-sectional sampling.*

Outbreak setting								
	PCR				Serological			
Longitudinal	0.5%	1%	2%	5%	0.5%	1%	2%	5%
WT	0.481 ± 0.147	0.405 ± 0.11	0.395 ± 0.0947	0.401 ± 0.113	0.445 ± 0.153	0.415 ± 0.117	0.423 ± 0.103	0.438 ± 0.147
Cori	1.74 ± 0.774	1.18 ± 0.573	0.806 ± 0.373	0.546 ± 0.212	1.55 ± 0.757	0.97 ± 0.577	0.621 ± 0.362	0.455 ± 0.191
EpiNow	0.329 ± 0.211	0.265 ± 0.145	0.25 ± 0.135	0.267 ± 0.168	0.308 ± 0.226	0.25 ± 0.171	0.232 ± 0.136	0.225 ± 0.134
GPRt	**0.228 ± 0.0713**	**0.2 ± 0.055**	**0.183 ± 0.0579**	**0.186 ± 0.0692**	**0.237 ± 0.0805**	**0.232 ± 0.0667**	**0.218 ± 0.0669**	**0.216 ± 0.0712**
Cross-sectional	0.05%	0.1%	0.2%	0.5%	0.05%	0.1%	0.2%	0.5%
WT	0.474 ± 0.149	0.396 ± 0.132	0.369 ± 0.102	0.358 ± 0.101	0.472 ± 0.136	0.484 ± 0.135	0.501 ± 0.132	0.509 ± 0.123
Cori	1.3 ± 0.676	0.859 ± 0.442	0.554 ± 0.184	0.502 ± 0.197	1.12 ± 0.51	0.825 ± 0.363	0.664 ± 0.246	0.584 ± 0.167
EpiNow	0.306 ± 0.199	0.277 ± 0.174	0.294 ± 0.184	0.302 ± 0.205	**0.25 ± 0.146**	0.29 ± 0.181	0.269 ± 0.153	0.295 ± 0.174
GPRt	**0.215 ± 0.063**	**0.178 ± 0.0509**	**0.177 ± 0.049**	**0.172 ± 0.0471**	0.262 ± 0.09	**0.265 ± 0.076**	**0.249 ± 0.0791**	**0.238 ± 0.0732**
Uniform underreporting	1%	2%	5%	10%	1%	2%	5%	10%
WT	0.395 ± 0.105	0.389 ± 0.106	0.377 ± 0.111	0.382 ± 0.104	0.407 ± 0.0937	0.425 ± 0.114	0.429 ± 0.133	0.408 ± 0.139
Cori	0.892 ± 0.552	0.614 ± 0.355	0.412 ± 0.162	0.38 ± 0.108	0.98 ± 0.553	0.587 ± 0.245	0.431 ± 0.149	0.38 ± 0.139
EpiNow	0.311 ± 0.193	0.31 ± 0.186	0.359 ± 0.231	0.394 ± 0.245	**0.254 ± 0.136**	**0.212 ± 0.128**	0.251 ± 0.139	0.267 ± 0.16
GPRt	**0.204 ± 0.0806**	**0.22 ± 0.0878**	**0.181 ± 0.0677**	**0.181 ± 0.0467**	0.26 ± 0.0948	0.259 ± 0.0839	**0.222 ± 0.0865**	**0.233 ± 0.0807**
Random trend setting								
	PCR				Serological			
Longitudinal	0.5%	1%	2%	5%	0.5%	1%	2%	5%
WT	0.427 ± 0.149	0.345 ± 0.101	0.321 ± 0.101	0.292 ± 0.104	0.398 ± 0.118	0.322 ± 0.0851	0.288 ± 0.0937	0.26 ± 0.0646
Cori	1.28 ± 0.678	0.872 ± 0.512	0.622 ± 0.326	0.392 ± 0.159	1.04 ± 0.666	0.57 ± 0.325	0.358 ± 0.151	0.28 ± 0.0768
EpiNow	0.332 ± 0.233	0.321 ± 0.195	0.337 ± 0.232	0.349 ± 0.244	0.364 ± 0.225	0.291 ± 0.207	0.304 ± 0.185	0.296 ± 0.209
GPRt	**0.199 ± 0.0745**	**0.187 ± 0.0652**	**0.181 ± 0.0551**	**0.157 ± 0.0476**	**0.232 ± 0.0733**	**0.216 ± 0.0754**	**0.213 ± 0.0699**	**0.194 ± 0.0694**
Cross-sectional	0.05%	0.1%	0.2%	0.5%	0.05%	0.1%	0.2%	0.5%
WT	0.392 ± 0.123	0.335 ± 0.107	0.319 ± 0.101	0.284 ± 0.096	0.381 ± 0.128	0.404 ± 0.118	0.406 ± 0.13	0.396 ± 0.115
Cori	0.94 ± 0.535	0.581 ± 0.217	0.478 ± 0.159	0.411 ± 0.117	0.68 ± 0.242	0.61 ± 0.219	0.513 ± 0.137	0.485 ± 0.117
EpiNow	0.359 ± 0.202	0.356 ± 0.191	0.421 ± 0.225	0.383 ± 0.215	0.362 ± 0.242	0.438 ± 0.216	0.436 ± 0.265	0.456 ± 0.264
GPRt	**0.192 ± 0.068**	**0.182 ± 0.0641**	**0.168 ± 0.0512**	**0.149 ± 0.0467**	**0.242 ± 0.0889**	**0.246 ± 0.0877**	**0.233 ± 0.0925**	**0.221 ± 0.0788**
Uniform underreporting	1%	2%	5%	10%	1%	2%	5%	10%
WT	0.285 ± 0.095	0.275 ± 0.0884	0.267 ± 0.107	0.259 ± 0.109	0.316 ± 0.0943	0.3 ± 0.0917	0.287 ± 0.0892	0.282 ± 0.0908
Cori	0.558 ± 0.368	0.396 ± 0.187	0.326 ± 0.128	0.281 ± 0.109	0.527 ± 0.298	0.382 ± 0.205	0.303 ± 0.108	0.276 ± 0.105
EpiNow	0.348 ± 0.244	0.315 ± 0.179	0.383 ± 0.238	0.336 ± 0.201	0.308 ± 0.173	0.356 ± 0.236	0.318 ± 0.22	0.349 ± 0.237
GPRt	**0.172 ± 0.0694**	**0.17 ± 0.0632**	**0.163 ± 0.071**	**0.181 ± 0.0682**	**0.2 ± 0.0793**	**0.213 ± 0.0831**	**0.206 ± 0.0848**	**0.211 ± 0.0742**

Table H.1: *Mean absolute error of each method averaged over instances and time points for each setting, along with standard deviation of the absolute error. "PCR" and "Serological" denote settings where the observations are generated by the respective testing method. Individual column headings give the percentage of the population enrolled in testing.*

	PCR				Serological			
Longitudinal	0.5%	1%	2%	5%	0.5%	1%	2%	5%
$d = 7$								
WT	0.47 ± 0.143	0.379 ± 0.0888	0.399 ± 0.117	0.379 ± 0.113	0.458 ± 0.129	0.427 ± 0.114	0.41 ± 0.115	0.425 ± 0.128
Cori	1.51 ± 0.774	0.975 ± 0.62	0.655 ± 0.303	0.436 ± 0.149	1.64 ± 0.841	1.02 ± 0.58	0.574 ± 0.255	0.459 ± 0.191
EpiNow	0.296 ± 0.19	0.273 ± 0.147	0.312 ± 0.191	0.288 ± 0.199	0.262 ± 0.155	0.257 ± 0.152	0.244 ± 0.133	0.232 ± 0.137
GPRt	**0.201** ± **0.057**	**0.173** ± **0.0529**	**0.176** ± **0.0512**	**0.174** ± **0.0608**	**0.233** ± **0.073**	**0.229** ± **0.0726**	**0.211** ± **0.0669**	**0.202** ± **0.0712**
$d = 31$								
WT	0.6 ± 0.241	0.462 ± 0.149	0.41 ± 0.102	0.426 ± 0.111	0.458 ± 0.152	0.484 ± 0.122	0.481 ± 0.136	0.491 ± 0.145
Cori	2.1 ± 0.795	1.81 ± 0.785	1.19 ± 0.561	0.752 ± 0.384	1.4 ± 0.757	1.13 ± 0.621	0.693 ± 0.3	0.493 ± 0.177
EpiNow	0.326 ± 0.2	0.314 ± 0.202	0.229 ± 0.116	0.263 ± 0.178	0.434 ± 0.309	0.311 ± 0.219	0.253 ± 0.172	**0.224** ± **0.183**
GPRt	**0.22** ± **0.0777**	**0.213** ± **0.0681**	**0.192** ± **0.0555**	**0.189** ± **0.0496**	**0.274** ± **0.103**	**0.266** ± **0.0977**	**0.25** ± **0.0706**	0.238 ± 0.0771

Table H.2: *Mean absolute error of each method averaged over instances and time points for each setting, along with standard deviation of the absolute error. "PCR" and "Serological" denote settings where the observations are generated by the respective testing method. Individual column headings give the percentage of the population enrolled in testing. These results show the longitudinal testing, with R generated according to the "outbreak" scenario. Two alternate values of d are shown (7 and 31). Calibration results for these settings are also shown in the following section.*

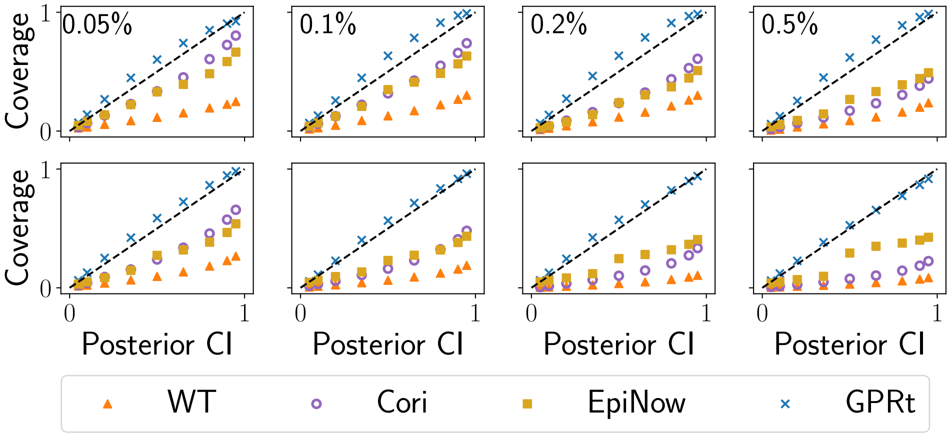


Figure H.6: *Calibration for the random trend setting, cross-sectional sampling.*

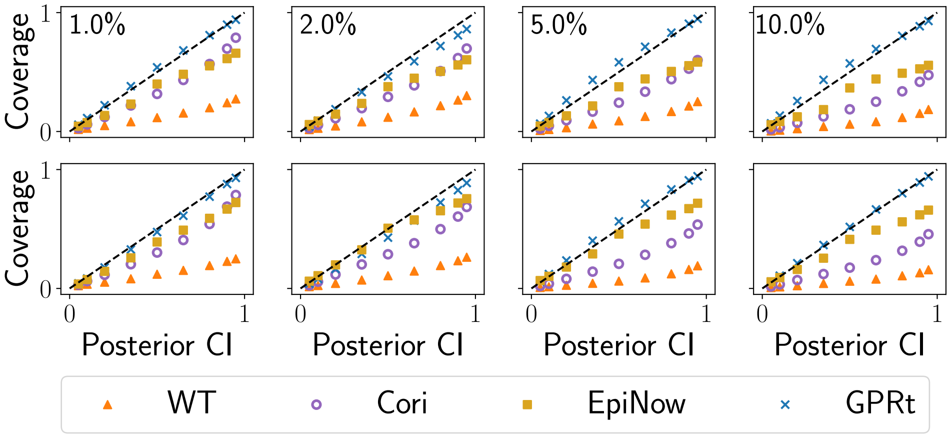


Figure H.7: *Calibration for the outbreak setting, uniform underreporting.*

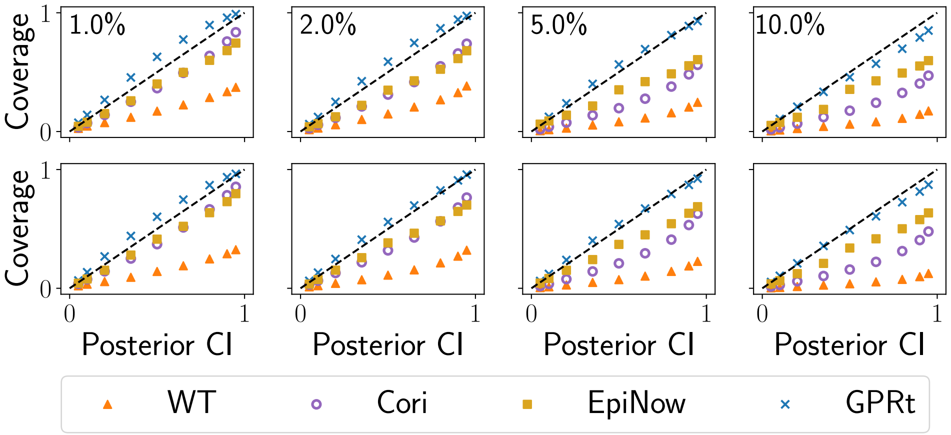


Figure H.8: *Calibration for the random trend setting, uniform underreporting.*