BOSTON UNIVERSITY

GRADUATE SCHOOL OF ARTS AND SCIENCES

Dissertation

COMPLEXITY PENALIZED METHODS FOR STRUCTURED AND UNSTRUCTURED DATA

by

ALEKSANDRINA VALERIEVA GOEVA

B.S., Sofia University, 2011
M.A., Boston University, 2013

Submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

2017

Approved by

First Reader

Eric D. Kolaczyk, PhD
Professor of Mathematics and Statistics

Second Reader

Henry Lam, PhD
Assistant Professor of Industrial and Operations Engineering

Acknowledgments

First, I would like to thank my advisors Henry Lam and Eric Kolaczyk for their mentorship. They are the pillars that supported me through my journey from being a student to becoming an aspiring researcher and I have learned a great deal from working with them.

I would also like to acknowledge the help of our collaborators Jason Qian (UMich), Bo Zhang (IBM) and Richard Lehoucq (Sandia), whose invaluable input contributed greatly to our projects, and the help of my groupmates for their useful feedback.

The six years spent at Boston University would not have been the same without the guidance of my professors and the company of my peers.

I am particularly grateful to my officemate Anthea for her friendship and compassion, to Soudeh for her wisdom, to Lilly for being a good friend, and to Karrie for being nearly inseparable through the beginning stages of the program.

Throughout the years I have had the chance to interact with a myriad of professors. Their professionalism and passion for teaching, science and life have guided me in my own development. I would like to cordially thank professor Mamikon Ginovyan for not only cultivating a strong sense of rigor, but also for genuinely caring and for always being there with helpful advice, professor Mark Kon - for his everlasting patience in explaining concepts countless times from scratch and for always offering his help, professor Lorenzo Rosasco (MIT) - for his crystal clear explanation of machine learning and exemplary devotion to rigor, professor Luis Carvalho - for sparking our interest in Bayesian inference and highlighting the importance of computation in the job of a statistician, professor Murad Taqqu - for his inspiring teaching style and lectures in probability, professor Daniel Sussman for reminding me how to be excited about research through his endless enthusiasm, professor Haviland Wright for bringing a fresh perspective, and our chair - professor Tasso Kaper - for giving all-around life

advice and for making the department feel like home with his welcoming attitude and by greeting everyone by name.

Becoming a graduate student at Boston University meant moving from my home-country, Bulgaria, to the States, hence putting a large (geographical) distance between me and my dear friends from home. Our friendships have time and time again survived the test of space and time, and I would not be complete without the presence of Nadya, Reni, Meli, Adi, Tanya and Krisi in my life.

Here in Boston I have been incredibly lucky to cross paths with some truly unique personalities and even more lucky to befriend them. The list is long, but I cannot omit thanking my roommate through the years, Yered - for enduring more than one can ask for, my fellow misfit Nahom - for always having my back, Molly - for her encouragement to overcome any fear, Sixing - for his love for cilantro, and last but not least, Benjamin for inspiring the artist in me, provoking me to get out of my comfort zone, believing in me and helping me discover that I am capable of more than I know, understanding me and being there through a fair amount of adventures.

A tremendous thanks goes to my parents, Nelly and Valeri, for their unconditional love and support, for planting the seed of curiosity in me early on, for teaching me critical thinking, showing me how to appreciate good times and how to be resilient in hard times, for encouraging me to set high goals and believing there isn't anything that is beyond my reach. I would like to thank Daniel for being my good older brother.

COMPLEXITY PENALIZED METHODS FOR STRUCTURED AND UNSTRUCTURED DATA

ALEKSANDRINA VALERIEVA GOEVA

Boston University, Graduate School of Arts and Sciences, 2017

Major Professor: Eric D. Kolaczyk, PhD
Professor of Mathematics and Statistics

ABSTRACT

A fundamental goal of statisticians is to make inferences from the sample about characteristics of the underlying population. This is an inverse problem, since we are trying to recover a feature of the input with the availability of observations on an output. Towards this end, we consider complexity penalized methods, because they balance goodness of fit and generalizability of the solution. The data from the underlying population may come in diverse formats - structured or unstructured - such as probability distributions, text tokens, or graph characteristics. Depending on the defining features of the problem we can chose the appropriate complexity penalized approach, and assess the quality of the estimate produced by it. Favorable characteristics are strong theoretical guarantees of closeness to the true value and interpretability. Our work fits within this framework and spans the areas of simulation optimization, text mining and network inference. The first problem we consider is model calibration under the assumption that given a hypothesized input model, we can use stochastic simulation to obtain its corresponding output observations. We formulate it as a stochastic program by maximizing the entropy of the input distribution subject to moment matching. We then propose an iterative scheme via simulation

to approximately solve it. We prove convergence of the proposed algorithm under appropriate conditions and demonstrate the performance via numerical studies. The second problem we consider is summarizing text documents through an inferred set of topics. We propose a frequentist reformulation of a Bayesian regularization scheme. Through our complexity-penalized perspective we lend further insight into the nature of the loss function and the regularization achieved through the priors in the Bayesian formulation. The third problem is concerned with the impact of sampling on the degree distribution of a network. Under many sampling designs, we have a linear inverse problem characterized by an ill-conditioned matrix. We investigate the theoretical properties of an approximate solution for the degree distribution found by regularizing the solution of the ill-conditioned least squares objective. Particularly, we study the rate at which the penalized solution tends to the true value as a function of network size and sampling rate.

Contents

List of Figures

List of Abbreviations

ER	.....	Erdos-Renyi
HPC	.....	Hierarchical Poisson Convolution
MD	.....	Mirror Descent
MDSA	.....	Mirror Descent Stochastic Approximation
ME	.....	Maximum Entropy
NMF	.....	Non-negative Matrix Factorization
SA	.....	Stochastic Approximation

Chapter 1

Introduction

At the heart of any statistical estimation problem is the following: there is a characteristic of interest from an underlying (unobserved) population, and through some method we obtain an estimate of that characteristic calculated from a sample. This falls within the framework of inverse problems in the sense that we are trying to recover a feature of the input with the availability of observations on an output. To this end, complexity penalized methods are a preferred approach since they are a class of methods that balance the goodness of fit to the observed data and the generalizability of the solution to the broader unobserved population. The data from the underlying population may come in a variety of formats - structured or unstructured - scoping probability distributions, natural text tokens, or graph characteristics, to name a few. Once we have chosen the appropriate flavor of complexity penalized approach, naturally we want to assess the quality of the estimate that we produce. Desired features of the estimate are strong theoretical guarantees of closeness to the true population value, empirical validation on real data sets, and clarity, rigor and interpretability of the solution. The following work fits within this mindset and spans the areas of simulation optimization, text mining and network inference.

The first problem we consider is an inverse problem belonging to the world of model calibration under the assumption that given a hypothesized input model, we can use stochastic simulation to obtain its corresponding output observations. This kind of inverse problem is observed in some service operations settings where data are

available only for system outputs but we do not know the constituent input models. Examples are service call centers and patient flows in clinics, where sometimes only the waiting time or the queue length data are collected for economic or operational reasons, and the data on the "input distributions", namely interarrival and service times, are limited or unavailable. We take a nonparametric viewpoint, and formulate this inverse problem as a stochastic program by maximizing the entropy of the input distribution subject to moment matching. We then propose an iterative scheme via simulation to approximately solve the program. We prove convergence of the proposed algorithm under appropriate conditions and demonstrate the performance via numerical studies.

The second problem we consider is in the field of topic allocation to text documents. This is a discussion based on the paper *A regularization scheme on word occurrence rates that improves estimation and interpretation of topical content*, (Airoldi and Bischof, 2015). When we have a large corpora of text documents, and we desire to efficiently describe/summarize them, we seek a small collection of words that characterize each document in the collection. An accepted approach to tackling this task is to disregard the order in which words appear in the text, i.e. the bag of words approach, and organize the previously unstructured data into a word count matrix. Then, to extract the most suitable topics, one might be tempted to choose the most frequent words within each text (this idea can be phrased as a regularization across the rows of the topic word matrix). However, such frequently used words may be common across many fields and may not be exclusive to the real topic of the document. Hence, in (Airoldi and Bischof, 2015), the authors propose a novel regularization scheme, within a complex Bayesian framework, that penalizes across the columns of the topic word matrix to leverage frequency and exclusivity of the selected topics. Our contribution is a different representation of the problem, from a frequentist point

of view, that gives clear insight into the nature of this novel regularization approach.

The third problem arises in the context of networks. Networks are widely used to model the relationships among elements in a system. Many empirical networks observed today can be viewed as samples of an underlying network, for example, large-scale online social networks. Hence, it is of fundamental interest to investigate the impact of the network sampling mechanism on the quality of characteristics estimated from the sampled network. We focus on the degree distribution as a fundamental feature. Under many popular sampling designs, this problem can be stated as a linear inverse problem characterized by an ill-conditioned matrix. This matrix relates the expectation of the sampled degree distribution to the true underlying degree distribution and depends entirely on the sampling design. The work of (Zhang et al., 2015) introduces an approximate solution for the degree distribution by regularizing the solution of the ill-conditioned least-squares problem corresponding to the naive estimator. We aim to theoretically characterize the distance between this penalized weighted least-squares estimator and the true degree counts vector. We succeed in the subcase of ego-centric design and achieve partial results for other more complicated sampling designs. We accompany the theoretical results with numerical simulations and visualizations that further illustrate the behavior of the estimator and allow us to make comparisons between the different sampling designs we have considered.

Chapter 2

Reconstructing Input Models via Simulation Optimization

2.1 Introduction

Stochastic simulation takes probability models as input and generates random outputs for subsequent performance analyses. The accuracy of the input model assumptions is critical to the analyses' credibility. Conventionally, the input models are conferred either through physical implication or expert opinion, or observable via input data. In this work, we ask a converse question: Given only *output* data from a stochastic system, can one infer about the input model?

The main motivation of asking this question is that, in many situations, a simulation modeler plainly may not have the availability of direct data or knowledge about the input. The only way to gain such knowledge could be data from other sources that are at the output level. For instance, such complication arises in the context of building a simulation model for a contract fulfillment center, where service agents work on a variety of processing tasks and, despite the abundant transaction data stored in the center's IT system, there is no record on the start, completion, or service times spent by each agent on each particular task. Similarly, in clinic operations, patients often receive service in multiple phases such as initial checkup, medical tests and doctor's consultation. Patients' check-in and check-out times could be accurately noted, but the "service" times provided by the medical staff could very well be unrecorded.

Clearly, these service time distributions are needed to build a simulation model, if an analyst wants to use the model for sensitivity or system optimization purpose.

The problem of inferring an input model from output data is sometimes reffered to as *model calibration*. In the simulation literature, this is often treated as a refinement process that occurs together with iterative comparisons between simulation reports and real-world output data (a task known as *model validation*; (Sargent, 2005; Kleijnen, 1995)). If simulation reports differ significantly from output data, the simulation model is re-calibrated (which can involve both the input distributions and system specifications), re-compared, and the process is iterated. Suggested approaches to compare simulation with real-world data are to conduct statistical tests such as two-sample mean-difference tests (Balci and Sargent, 1982) or the Schruben-Turing test (Schruben, 1980). Beyond that, inferring input from output seems to be an important problem that has not been largely discussed in the stochastic simulation literature (Nelson, 2016). Thus, on a high level, one contribution of this work is to study the first systematic framework for the input model calibration problem.

The setting we consider can be briefly described as follows. We assume an input model is missing and we make no parametric assumptions on the probability distribution. We assume, however, that certain output random variable from a well-specified system is observable with some data. Our task is to nonparametrically infer the input distribution. A key insight we use that distinguishes this problem from model calibration in other literature (e.g., computer experiments) is the intrinsic probabilistic structure of the system. Namely, the input and the output in stochastic simulation are represented as probability distributions, or in other words, the relation that links the observed and the to-be-calibrated objects is a (simulable) map between the input and output spaces of distributions. Our calibration method is designed to take such a relation into account.

More specifically, we use a moment-based approach to calibrate the input model. We match the moments of a sequence of statistics collected from the simulation and real-world data at the output level. Unless we can afford to match an infinite number of such moments, however, this scheme in general can lead to many candidate input models, therefore we face the unidentifyability issue. To tackle this problem, we propose entropy maximization as a criterion to identify the best model. This approach is motivated by the interpretation of the maximum entropy (ME) distribution as the conditional distribution given all prior information. Hence it is the most "natural" distribution without any further knowledge (Van Campenhout and Cover, 1981). Furthermore, this approach has been successfully used in the context of density estimation (Barron and Sheu, 1991). In our work, we offer justification of the consistency of our moment-matching approach, while at the same time we also reveal some finite-sample limitations. Roughly speaking, "natural" models such as unimodal or smooth distributions are amenable with our approach, but more "complicated" distributions are more difficult to infer and our scheme may not be powerful enough to calibrate such underlying model. We also provide some numerical results to substantiate these observations.

Beyond proposing and justifying our formulation, another key contribution of this work is algorithmic. In particular, we study a simulation optimization procedure that targets our ME moment-matching problem. A particularly challenging characteristic of this problem is that the moment statistics are at the output level, which is generally a nonlinear function of the input distribution. As a simple example, consider the mean waiting time of a queue as a function of the service time distribution. Thus the constraints in the optimization formulation are stochastic and could be nonconvex. To our best knowledge, there is no literature on this class of simulation optimization problems. As our key algorithmic contribution, we propose and analyze a

stochastic quadratic penalty method. The idea is to reformulate the optimization into a sequence of sum-of-squares optimizations with stochastic objectives and deterministic convex constraints, whereby each element of the sequence can be solved efficiently using mirror descent stochastic approximation (MDSA) (Nemirovski et al., 2009). MDSA is a constrained stochastic approximation (SA) algorithm that is especially suitable for our setting. We analyze the details of our reformulation. As a part of the algorithm, we develop a gradient estimation procedure and analyze the convergence of our MDSA. Note that, for a general simulation model, the ME formulation and our reformulation only guarantees a local optimum because of non-convexity. In practice, this issue can be addressed with the use of multi-start procedures. We also propose visualizations to validate our method.

The remainder of the chapter is organized as follows. Section 2.2 reviews the related literature. Section 2.3 introduces the problem setting and explains our optimization formulation. Section 2.4 presents and analyzes our algorithm. Section 2.5 reports numerical results. Section 3.4 concludes and discusses future work. Additionally, the Appendix at the end of the manuscript contains some Auxiliary Theorems A.1 and Supplementary Materials A.2 with the details of an additional variant of our method.

2.2 Related literature

We organize the literature review in two aspects, one related to the model calibration problem, and one related to our optimization approach.

2.2.1 Literature Related to Our Problem Setting

Input modeling and uncertainty in stochastic simulation focus mostly on the input level. (Barton, 2012) and (Song et al., 2014) review some major methods of quantifying the statistical errors from finite input data. Commonly used approaches therein

include the delta or two-point method (Cheng and Holland, 1998; Cheng and Holland, 2004), Bayesian methodology and model averaging (Chick, 2001; Zouaoui and Wilson, 2004) and resampling methods (Barton and Schruben, 2001; Barton et al., 2013). Our problem is more closely related to model calibration. In the simulation literature, this is often considered together with model validation (Sargent, 2005; Kleijnen, 1995). Conventional approaches compare simulated data with real-world historical output data through the use of common statistical or Turing tests (Balci and Sargent, 1982; Schruben, 1980), then conduct re-calibration, and repeat the process until the data are successfully validated (Banks et al., 2009; Kelton and Law, 2000).

Other literatures focus on the fact that the model calibration problem is an *inverse problem* (Tarantola, 2005). The general focus is the identification of parameters or functions that can only be inferred from transformed outputs. In the field of signal processing, the linear inverse problem (e.g., (Csiszár, 1991; Donoho et al., 1992)) reconstructs signals from measurements of linear transformations. Common practices in this context consist of minimization of an objectie function comprised of least-squares loss and a penalty, such as the entropy penalty, which also provides justification of our main formulation. In computer experiments (Santner et al., 2013), surrogate models based on complex physical laws require the calibration of physical parameters. Such models have wide scientific applications, such as weather prediction, oceanography, nuclear physics, and acoustics (e.g., (Wunsch, 1996; Shirangi, 2014)). Bayesian and Gaussian processes methodologies are commonly used (e.g., (Kennedy and O'Hagan, 2001; Currin et al., 1991)). We point out that Bayesian methods could be a potential alternative to the approach considered in this work, but since we consider discrete-event systems, one might need to resort to sophisticated techniques such as approximate Bayesian computation (Marjoram et al., 2003).

Also related to our work is the body of research on inference problems in the

context of queueing systems. The first stream similar to our work aims at inferring the constituent probability distributions of a queueing model based on its output data, e.g., queue length or waiting time data, collected either continuously or at discrete time points. The focus is on systems whose structures allow closed-form analyses or are amenable to analytic approximations via, for instance, the diffusion limit. The majority of the papers in this field assume that the inferred distribution(s) comes from a parametric family and use maximum likelihood estimators (Basawa et al., 1996; Pickands III and Stine, 1997; Basawa et al., 2008; Fearnhead, 2004; Wang et al., 2006; Ross et al., 2007; Heckmüller and Wolfinger, 2009; Whitt, 2012). Others work on nonparametric inference by exploiting specific queueing system structures (Bingham and Pitts, 1999; Hall and Park, 2004; Moulines et al., 2007; Feng et al., 2014). A related stream of literature studies point process approximation (see Section 4.7 of (Cooper, 1972), (Whitt, 1981; Whitt, 1982), and the references therein), based on a parametric approach and is motivated by traffic pattern modeling in communication networks. Finally, there are also a number of studies inspired by the "queue inference engine" by (Larson, 1990). But, instead of inferring the input models, many of these studies use transaction data to estimate the performance of a queueing system directly and hence do not take on the form of an inverse problem (see (Mandelbaum and Zeltyn, 1998) for a good survey of the earlier literature and (Frey and Kaplan, 2010) and its references for more recent progress). Several papers estimate both the queueing operational performance and the constituent input models (e.g., (Daley and Servi, 1998; Kim and Park, 2008; Park et al., 2011)), and can be considered to belong to both this stream and the aforementioned first stream of literature.

2.2.2 Literature Related to Our Methodology

Our formulation uses the widely used notion of maximum entropy (ME). In information theory, entropy can be viewed as the amount of intrinsic randomness (Cover and

Thomas, 1991). (Csiszár, 1991) studied axiomatic properties of ME (or more generally the I-divergence class). (Donoho et al., 1992) used ME to recover sparse signals. (Barron and Sheu, 1991) studied the use of moment-constrained ME in nonparametric density estimation and analyzed the convergence rate in terms of Kullback-Leibler (KL) divergence. (Lindley, 1956; DeGroot, 1962; Box and Hill, 1967; Bernardo, 1979; Chick and Ng, 2002) studied the use of entropy criterion to maximize information and identify important parameters in experiments. In finance, (Avellaneda et al., 2001) studied ME calibration of risk-neutral measures from derivative prices. This technique, known as the weighted Monte Carlo, has also been studied as a variance reduction technique (Glasserman and Yu, 2005).

Our optimization reformulation is inspired by the quadratic penalty method (Bertsekas, 1999), which is a deterministic nonlinear programming technique that reformulates the constraints as squared penalty and sequentially tunes the penalty parameter to approach optimality. Our algorithm to solve the sequence of optimizations in the reformulation utilizes MDSA proposed by (Nemirovski et al., 2009). (Nemirovski et al., 2009) analyzed convergence guarantees on convex programs with stochastic objectives. (Ghadimi and Lan, 2013) investigated related methods for nonconvex programs, and (Ghadimi and Lan, 2015) and (Dang and Lan, 2015) studied generalizations incorporating accelerated gradient and coordinate decomposition. The particular scheme of MDSA we consider uses entropic penalty, and is known as the entropic descent algorithm (Beck and Teboulle, 2003).

2.3 Setting and Formulation

We assume a discrete probability distribution $\mathbf{p} = (p_1, \ldots, p_n)$ for the input model, on the support set $\mathcal{S} = \{z_1, \ldots, z_n\}$, where the support size n can be potentially large. We let $\mathbf{X} = (X_1, \ldots, X_\tau)$, where $X_t \in \mathbb{S}$, be an i.i.d. sequence of input variates

each distributed under $\mathbf{p}$ over a random horizon τ. We denote the function $h(\cdot) \in \mathbb{R}$ as the system logic from the input sequence $\mathbf{X}$ to the output $h(\mathbf{X})$. We assume that h is completely specified and is computable, even though it may not be writable in closed-form, i.e. we can evaluate the output given $\mathbf{X}$. For example, $\mathbf{X}$ can denote the sequence of interarrival or service times for the customers in a queue, and $h(\mathbf{X})$ is the average queue length until the first idle time. Note that we can work in a more general framework where h depends on both $\mathbf{X}$ and other independent input sequences, say $\mathbf{Y}$, that possess known or observable distributions. In other words, we can have $h(\mathbf{X}, \mathbf{Y})$ as the output. Our subsequent discussion can be trivially extended to this case, and hence we will suppress these auxiliary input sequences throughout our exposition.

Consider the situation that only $h(\mathbf{X})$ can be observed via data. Let $y_1, \ldots, y_N$ be N observations of $h(\mathbf{X})$. Our task is to calibrate $\mathbf{p}$.

We match the moment-based statistics of the simulation output and the empirical output data. More precisely, let $\phi_j(\cdot) : \mathbb{R} \to \mathbb{R}, j = 1, \ldots, m$ be m moment functions specified by the modeler. Natural examples of ϕ_j include polynomials $\phi_j(y) = y^j$, and quantile-based functions $\phi_j(y) = I(y \leq c_j)$ for given values c_j, where I denotes the indicator function. We want to find $\mathbf{p}$ such that

$$E_{\mathbf{p}}[\phi_j(h(\mathbf{X}))] = \hat{\mu}_j \quad \text{for} \quad j = 1, \ldots, m \tag{2.1}$$

where $E_{\mathbf{p}}[\cdot]$ denotes the expectation with respect to the i.i.d. input process $\mathbf{X}$ each distributed as $\mathbf{p}$ (in other words, the product measure $\mathbf{p} \times \mathbf{p} \times \cdots$) and the random time τ. $\hat{\mu}_j$ is the empirical moment

$$\hat{\mu}_j = \frac{1}{N} \sum_{r=1}^{N} \phi_j(y_r), \ j = 1, \ldots, m \tag{2.2}$$

Note that, when the support size n is larger than m, there are typically more than

one $\mathbf{p}$ that satisfies (2.1). Moreover, depending on what the function h is, there may be a fundamental barrier in fully recovering $\mathbf{p}$ even if there is a full knowledge on the distribution of $h(\mathbf{X})$. For example, when h is identically equal to a constant, any $\mathbf{p}$ will give a perfect fit. The phenomenon of being unable to uniquely recover $\mathbf{p}$ is generally known as non-identifiability in the inverse modeling literature (e.g. (Tarantola, 2005)).

Our approach to reduce the number of distributions $\mathbf{p}$ that satisfy (2.1) is to maximize entropy, namely

$$\begin{array}{ll} \max & R(\mathbf{p}) = -\sum_{i=1}^{n} p_i \log p_i \\ \text{subject to} & E_{\mathbf{p}}[\phi_j(h(\mathbf{X}))] = \hat{\mu}_j \quad \text{for} \quad j = 1, \ldots, m \\ & \mathbf{p} \in \mathcal{P} \end{array} \tag{2.3}$$

where $\mathcal{P} = \{\mathbf{p} : \sum_{i=1}^{n} p_i = 1,\ p_i \geq 0 \text{ for } i = 1, \ldots, n\}$ is the probability simplex on $\mathcal{S} = \{z_1, \ldots, z_n\}$. $R(\mathbf{p})$ denotes the entropy of $\mathbf{p}$. The decision variable in (2.3) is the unknown input model $\mathbf{p}$.

2.3.1 Why Moment Matching?

Moment estimators are ubiquitous in parametric estimation (e.g., (Hall, 2005)). In our framework, which is closer to nonparametric, the moment-matching is based on two beliefs: 1) Given sufficiently many moments, one can recover the distribution of the output with high accuracy; 2) There is a one-to-one map from the input distribution to the output distribution. These two beliefs together would lead to the recovery of the input model to high accuracy given enough output data and moments matched. The following presents this logic more rigorously:

Lemma 2.3.1. *Denote $F_Y : \mathbb{R} \to [0, 1]$ as the distribution function of the output $Y = h(\mathbf{X})$. We write $F_Y(\mathbf{p})$ to highlight the dependence of F_Y in terms of the input distribution $\mathbf{p}$. Let $\mathbf{p}^*$ be the true input distribution, and $\mu_j = E_{\mathbf{p}^*}[\phi_j(h(\mathbf{X}))]$ for a chosen sequence of moment functions $\phi_j,\ j = 1, 2, \ldots$. Assume the following:*

1. *Let $\mathcal{N}(\mathbf{p}^*) \subset \mathcal{P}$ be a small neighborhood of $\mathbf{p}^*$. For every $\mathbf{p} \in \mathcal{N}(\mathbf{p}^*)$, $F_Y(\mathbf{p})$ is*

completely identified by a finite number of ϕ_j, i.e. $F_Y(\mathbf{p})$ is uniquely determined by

$$\int \phi_k(y) dF_Y(\mathbf{p})(y) = \mu_j, j = 1, \ldots, l$$

among the set of distributions $\mathcal{P}_Y = \{F_Y(\mathbf{q}) : \mathbf{q} \in \mathcal{P}\}$, for some $l > 0$. This defines a map $\mathcal{G} : \mathcal{N}(\mathbf{p}^*) \to \mathcal{N}(\boldsymbol{\mu})$, where $\mathcal{N}(\boldsymbol{\mu})$ denotes a small neighborhood of $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_l) \in \mathbb{R}^l$. We assume moreover that $\mathcal{G}$ is one-to-one.

2. *$\mathcal{G}^{-1}$, the inverse of $\mathcal{G}$, is continuous.*

3. *The empirical moment $\hat{\mu}_j \to \mu_j$ a.s. as the output sample size $N \to \infty$, for $j = 1, \ldots, l$.*

Then $\mathcal{G}^{-1}(\hat{\boldsymbol{\mu}}) \to \mathbf{p}^*$ a.s. as $N \to \infty$, where $\hat{\boldsymbol{\mu}} = (\hat{\mu}_1, \ldots, \hat{\mu}_l)$.

Proof of Lemma 2.3.1:

As $N \to \infty$, by Assumption 3, we have $\hat{\mu}_j \to \mu_j$. Hence by Assumptions 1 and 2, we have $\mathcal{G}^{-1}(\hat{\boldsymbol{\mu}}) \to \mathbf{p}^*$. $\square$

Lemma 2.3.1 is of course a rather trivial result, stating that with enough moments and a one-to-one map from the input to the output distribution, one would be able to consistently calibrate the input model. The conditions in the lemma are generally very difficult to verify in practice, since the map $\mathcal{G}$ is related to the simulation process that could be highly complex. The use of Lemma 2.3.1 is more about pointing out the limitation of moment-matching: we expect that, if the input support size is n, we need $l = n - 1$ moments to form a well-determined system of equations and subsequently recover the input model. Hence when n is big, the number of moments matched needs also be big. However, with finite output observation size, statistical errors prohibit matching too many moments, since in this case some of the empirical moments could be badly estimated (i.e., overfitting). Thus in practice one would only match a small number of moments. This in turn implies that one would have an under-determined system of equations that results in plenty of candidate input models. An additional criterion, namely the entropy, is therefore needed to pin down the choices.

2.3.2 Why Maximize Entropy?

The use of the ME criterion is strongly inspired from results in density estimation. Let us for a moment take the simulation model aside, and consider the estimation of the density from a finite number of i.i.d. data. It is known that by increasing the number of matched moments suitably with the sample size, the ME density subject to empirical moment matching is a consistent density estimator and, moreover, elicits explicitly obtainable convergence rate to the true density, in terms of KL divergence, according to the class of moment functions used (Barron and Sheu, 1991). The only difference between our setting and this classical setting is the layer of transformation induced by the simulation model. We note that one could potentially derive the precise conditions on the simulation model to generalize these classical results. However, such an attempt may not be meaningful because the obtained conditions are likely unverifiable for the complex simulation maps typically occurring in practice. We thus choose to only justify the ME criterion on an intuitive level.

In addition, entropy has been widely used as a proxy for the expected gain or information in experimental design (e.g., (Lindley, 1956; Box and Hill, 1967; Bernardo, 1979)). The entropy here often refers to the output distribution, though input distribution has also been considered if the goal is to optimize inference at the input level (Chick and Ng, 2002). ME is also used in financial option pricing to infer the most "natural" risk-neutral measure nonparametrically (Avellaneda et al., 2001). This work in a sense follows these widely taken viewpoints of ME, under the presumption that we do not have any additional information about the input model.

2.4 Optimization Procedure

This section focuses on solving our main formulation (2.3). Our strategy consists of two parts: a reformulation of (2.3) into a sequence of optimization programs with

deterministic convex constraints, which is inspired from the quadratic penalty method in nonlinear programming (Section 2.4.1), and a constrained SA procedure for finding local optima for each of these programs (Section 2.4.2).

2.4.1 Transforming into a Sequence of Stochastic Programs with Entropy Constraints

Note that the constraints in (2.3) are in general nonlinear because the i.i.d. input sequence means that the expectation $E_{\mathbf{p}}[\cdot]$ is a convolution of $\mathbf{p}$. In fact, $E_{\mathbf{p}}[\phi_j(h(\mathbf{X}))]$ is a high-dimensional polynomial in $\mathbf{p}$, and is in general non-convex. Moreover, this polynomial can involve a huge (or even infinite) number of terms and hence its evaluation requires simulation approximation.

To handle such non-convex stochastic constraints, consider a sequence of optimization programs parametrized by η

$$\begin{array}{ll} \min & \sum_{j=1}^{m}(E_{\mathbf{p}}[\phi_j(h(\mathbf{X}))] - \hat{\mu}_j)^2 \\ \text{subject to} & R(\mathbf{p}) \geq \eta \\ & \mathbf{p} \in \mathcal{P} \end{array} \tag{2.4}$$

This sequence satisfies the following key properties:

Theorem 2.4.1. *Suppose that $E_{\mathbf{p}}[\phi_j(h(\mathbf{X}))]$ is a continuous function in $\mathbf{p}$, for each $j = 1, \ldots, m$. Let $W^*(\eta)$ denotes the optimal value of (2.4) indexed at η. We have*

1. *$W^*(\eta)$ decreases as η decreases from $\log n$ to 0.*
2. *An optimal solution of (2.4), at any $\eta \in [0, \log n]$, exists. Denote this optimal solution as $\mathbf{p}^*(\eta)$.*
3. *If there exists an $\eta^* = \sup\{\eta \in [0, \log n] : W^*(\eta) = 0\}$, then $\mathbf{p}^*(\eta^*)$ is optimal for (2.3).*
4. *If there does not exist any $\eta \in [0, \log n]$ such that $W^*(\eta) = 0$, then (2.3) is infeasible.*

In view of Theorem 2.4.1, our strategy is to solve (2.4) at different values of η and identify η^*. This strategy is a modification of the quadratic penalty method for solving deterministic nonlinear programs, where equality constraints are relaxed into the objective function with squaring (Bertsekas, 1999). Note that while the standard quadratic penalty method solves a sequence of optimizations with objectives consisting of both the primal objective and the relaxed squared constraints, our formulation (2.4) has chosen to put the primal objective as a constraint in the optimization sequence. This arrangement is beneficial for a few reasons. First, it allows us to more easily locate a stochastic root $\eta^* \in [0, \log n]$ for which we can look for an optimal solution, whereas the standard quadratic penalty method requires the sequencing index to go to ∞ and does not offer an explicit guideline on when to stop searching. Second, our primal objective, as an entropy, has advantageous concave structure that allows running efficient MDSA (which we will discuss momentarily) when translated as constraint. Third, η has the interpretation as the entropy level of the distributions to be considered. For convenience we call our method the *stochastic quadratic penalty method*. In the Supplementary Materials, we provide some discussion on applying a variant of our approach that is closer to the conventional quadratic penalty method.

2.4.2 Constrained Stochastic Approximation for Solving the Optimization Sequence

Although formulation (2.4) is still non-convex, its constraints are convex and deterministic, which can be handled more easily using SA than in the original formulation (2.3). This section investigates the design and analysis of an MDSA algorithm for finding a local optimum of (2.4).

MDSA is the stochastization of the mirror descent (MD) method, an iterative procedure for solving deterministic convex programs (Nemirovski and Yudin, 1983). MD was first motivated for optimizations in general normed space, where the space

of the solution (i.e. the primal) is different from the space of the gradient (i.e. the dual), and hence the usual gradient descent scheme does not make sense. MD uses the insight of first mapping the solution to the dual space (using a map associated with a strongly convex function called the distance-generating function), moving the solution along the gradient direction in the dual space, and mapping it back to the primal (via the so-called prox-mapping). In the Euclidean space, it is not necessary to use MD since the primal and the dual space are the same. But, by choosing a suitable primal-dual mapping, MD can provide advantages in the convergence speed in terms of less dependence on the problem dimension. Moreover, the resulting optimization subroutine required in the iteration can be very efficient.

To describe the algorithm, MD finds the next iterate via optimizing the objective function linearized at the current solution, together with a penalty on the distance of movement of the next iterate. When the objective function is only accessible via simulation, the linearized objective function, or the gradient, at each iteration can only be estimated, in which case the procedure becomes MDSA (Nemirovski et al., 2009). More precisely, given a current iterate $\mathbf{p}^k$, MDSA solves

$$\begin{array}{ll} \min & \gamma^k \hat{\boldsymbol{\psi}}^{k\prime}(\mathbf{p}-\mathbf{p}^k) + V(\mathbf{p}^k, \mathbf{p}) \\ \text{subject to} & R(\mathbf{p}) \geq \eta \\ & \mathbf{p} \in \mathcal{P} \end{array} \tag{2.5}$$

where $\hat{\boldsymbol{\psi}}^k$ carries the gradient information at $\mathbf{p}^k$, $V(\cdot,\cdot)$ is some distance measure known as the prox-function (Nemirovski et al., 2009), and γ^k is the step size at iteration k. (2.5) thus minimizes over the feasible set of $\mathbf{p}$ with an objective function linearized at $\mathbf{p}^k$, penalized by the distance $(1/\gamma^k)V(\mathbf{p}^k,\cdot)$. To implement this scheme, we need to investigate: 1) how to obtain $\hat{\boldsymbol{\psi}}^k$, 2) the complexity of the program (2.5) with a choice of V, and 3) the convergence property of the procedure in relation to γ^k. The next three subsections present these investigations respectively.

Even though MDSA can provably converge to a local optimum, the non-convexity of (2.4) means that there is no guarantee of finding a global one. However, we can obtain some evidence of global convergence by scrutinizing the monotonic pattern of $W^*(\eta)$ as predicted by Theorem 2.4.1. We will revisit this discussion in our numerical experiments in Section 2.5.

Gradient Estimation

We denote $W(\mathbf{p})$ as the objective function in (2.4). Though $W(\mathbf{p})$ is a function on the Euclidean space and in principle can be differentiated in a standard manner, naive differentiation of $W(\mathbf{p})$ with respect to $\mathbf{p}$ in general does not lead to any simulable form. This is because an arbitrary perturbation of $\mathbf{p}$ can shoot outside the probability simplex, and the resulting gradient will be a high-dimensional polynomial in $\mathbf{p}$ that has no probabilistic interpretation. To get around this issue, we use the idea of the Gateaux derivative defined on a functional of probability distribution (Serfling, 2009). This consists of restricting the perturbations of $\mathbf{p}$ within the probability simplex as represented by the mixtures $(1-\epsilon)\mathbf{p}+\epsilon\mathbf{1}_i$, where $\mathbf{1}_i$ is a point mass at the support point z_i and $0 \leq \epsilon \leq 1$ is a mixture parameter. The idea is to differentiate $W((1-\epsilon)\mathbf{p}+\epsilon\mathbf{1}_i)$ at $\epsilon = 0$. The resulting quantity, which we call $\psi_i(\mathbf{p})$, satisfies the following:

Proposition 2.4.1. *We have:*

1. *Suppose W is differentiable in $\mathcal{P}$, then*

$$\nabla W(\mathbf{p})'(\mathbf{q}-\mathbf{p}) = \boldsymbol{\psi}(\mathbf{p})'(\mathbf{q}-\mathbf{p}) \tag{2.6}$$

for any $\mathbf{q} \in \mathcal{P}$, where $\boldsymbol{\psi}(\mathbf{p}) = (\psi_1(\mathbf{p}), \ldots, \psi_n(\mathbf{p}))'$ and

$$\psi_i(\mathbf{p}) = \frac{d}{d\epsilon} W((1-\epsilon)\mathbf{p}+\epsilon\mathbf{1}_i)\bigg|_{\epsilon=0^+}$$

2. *Assume, for all* $j = 1, \ldots, m$, $E_{\mathbf{p}}[|\phi_j(h(\mathbf{X}))|^{l+\theta}] < \infty$ *for an integer* $l \geq 1$ *and some small* $\theta > 0$, *and* τ *satisfies* $E_{\mathbf{p}}[e^{\beta\tau}] < \infty$ *for some small* $\beta > 0$. *Also assume* $\mathbf{p} = (p_1, \ldots, p_n)$ *where each* $p_i > 0$. θ, β *might depend on* $\mathbf{p}$. *Then* $\psi_i(\mathbf{p})$ *is finite for all* i *and is equal to*

$$\psi_i(\mathbf{p}) = 2\sum_{j=1}^{m}(E_{\mathbf{p}}[\phi_j(h(\mathbf{X}))] - \hat{\mu}_j)E_{\mathbf{p}}[\phi_j(h(\mathbf{X}))S_i(\mathbf{X};\mathbf{p})] \tag{2.7}$$

$$= 2E_{\mathbf{p}}\left[\sum_{j=1}^{m}(\phi_j(h(\mathbf{X})) - \hat{\mu}_j)\phi_j(h(\tilde{\mathbf{X}}))S_i(\tilde{\mathbf{X}};\mathbf{p})\right] \tag{2.8}$$

where

$$S_i(\mathbf{x};\mathbf{p}) = \sum_{t=1}^{\tau}\frac{I_i(x_t)}{p_i} - \tau$$

for $\mathbf{x} = (x_1, \ldots, x_\tau)$. *Here* $I_i(x) = 1$ *if* $x = z_i$ *and 0 otherwise.* $\mathbf{X}$ *and* $\tilde{\mathbf{X}}$ *are two independent copies of the i.i.d. input process generated under* $\mathbf{p}$. *Moreover, for all* i, j *and* $1 \leq s \leq l$, *the moments*

$$E_{\mathbf{p}}\left[(\phi_j(h(\mathbf{X})))^s\right],\ E_{\mathbf{p}}\left[(\phi_j(h(\mathbf{X}))S_i(\mathbf{X};\mathbf{p}))^s\right]$$

are continuous in $\mathbf{p}$.

Equation (2.6) guarantees that the Gateaux derivative $\boldsymbol{\psi}(\mathbf{p})$ behaves the same as a standard gradient $\nabla W(\mathbf{p})$ when applying to any directions within the probability simplex $\mathcal{P}$, which are the only directions we consider in MDSA. Importantly, $\boldsymbol{\psi}(\mathbf{p})$ is simulable by using (2.8). Through (2.7) and (2.8), each component $\psi_i(\mathbf{p})$ can now be expressed via the function $S_i(\cdot;\mathbf{p})$ that plays the role of the score function as in the conventional likelihood ratio method (also known as the score function method) (Glynn, 1990; Reiman and Weiss, 1989; Rubinstein, 1989; L'Ecuyer, 1990) in parametric sensitivity analysis. Thus Proposition 2.4.1 can be viewed as a nonparametric version of the likelihood ratio method. (Ghosh and Lam, 2015a) and (Ghosh and Lam, 2015b) have also studied such type of methods for finding the gradients of expectation-type performance measures. Proposition 2.4.1 Part 2 generalizes

their results to sums of squared expectations and random time horizons, relaxes their boundedness condition on the performance function, and identifies the conditions to guarantee finite moments of the estimator and their continuity with respect to the underlying probability measure.

In view of Proposition 2.4.1, we have the following gradient estimation scheme:

Corollary 2.4.1. *Under the assumptions in Proposition 2.4.1 Part 2, an unbiased estimator for* $\boldsymbol{\psi}(\mathbf{p})$ *is given by* $\widehat{\boldsymbol{\psi}(\mathbf{p})} = (\widehat{\psi_1(\mathbf{p})}, \ldots, \widehat{\psi_n(\mathbf{p})})'$*, where*

$$\widehat{\psi_i(\mathbf{p})} = 2\sum_{j=1}^{m} \frac{1}{M_1} \sum_{r=1}^{M_1} (\phi_j(h(\mathbf{X}^{(r)})) - \hat{\mu}_j) \frac{1}{M_2} \sum_{r=1}^{M_2} \phi_j(h(\tilde{\mathbf{X}}^{(r)})) S_i(\tilde{\mathbf{X}}^{(r)}; \mathbf{p}) \tag{2.9}$$

and $\mathbf{X}^{(r)}$*'s and* $\tilde{\mathbf{X}}^{(r)}$*'s are* M_1 *and* M_2 *independent copies of the i.i.d. input process generated under* $\mathbf{p}$ *and are used simultaneously in all* $\widehat{\psi_i(\mathbf{p})}$*.*

Solving Stepwise Subprograms in MDSA

We discuss the choice of the prox-function V and how to solve program (2.5). Following (Nemirovski et al., 2009), we take V as the Bregman divergence

$$V(\mathbf{p}, \mathbf{q}) = \omega(\mathbf{q}) - \omega(\mathbf{p}) - \nabla\omega(\mathbf{p})'(\mathbf{q} - \mathbf{p}) \tag{2.10}$$

where $\omega(\cdot)$ is called the distance-generating function and is strongly convex, i.e.

$$\omega(\mathbf{q}) - \omega(\mathbf{p}) \geq \nabla\omega(\mathbf{p})'(\mathbf{q} - \mathbf{p}) + \frac{\alpha}{2}\|\mathbf{q} - \mathbf{p}\|^2 \tag{2.11}$$

for all $\mathbf{p}, \mathbf{q}$ in the feasible region, $\|\cdot\|$ is some norm, and $\alpha > 0$. One choice of ω that is especially appropriate for our feasible space, the probability simplex, is the negative entropy $\omega(\mathbf{p}) = \sum_{i=1}^{m} p_i \log p_i$. In this case $\alpha = 1$ and $\|\cdot\|$ is taken as the L_1-norm. The function $V(\mathbf{p}, \mathbf{q})$ derived from (2.10) is the KL divergence given by

$$V(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^{n} q_i \log \frac{q_i}{p_i} \tag{2.12}$$

We shall use this choice in our procedure. The iteration (2.5) using (2.12), but without the constraint $R(\mathbf{p}) \geq \eta$ and the stochasticity, is also called the entropic descent algorithm (Beck and Teboulle, 2003).

Consider a generic formulation of (2.5) written as

$$\begin{array}{ll} \min & \boldsymbol{\xi}'(\mathbf{q}-\mathbf{p}) + V(\mathbf{p},\mathbf{q}) \\ \text{subject to} & R(\mathbf{q}) \geq \eta \\ & \mathbf{q} \in \mathcal{P} \end{array} \tag{2.13}$$

where $V(\mathbf{p},\mathbf{q})$ is defined by (2.12), for some given $\mathbf{p} = (p_1, \ldots, p_n)$ and $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_n)$. The solution of (2.13) is in the following form:

Proposition 2.4.2. *Denote*

$$\kappa(\mathbf{p}, \boldsymbol{\xi}, \alpha) = \frac{\sum_{i=1}^n p_i^{\frac{1}{1+\alpha}} e^{-\frac{\xi_i}{1+\alpha}} \left(-\frac{1}{1+\alpha}\log p_i + \frac{\xi_i}{1+\alpha}\right)}{\sum_{i=1}^n p_i^{\frac{1}{1+\alpha}} e^{-\frac{\xi_i}{1+\alpha}}} + \log \sum_{i=1}^n p_i^{\frac{1}{1+\alpha}} e^{-\frac{\xi_i}{1+\alpha}} \tag{2.14}$$

Suppose $\kappa(\mathbf{p}, \boldsymbol{\xi}, 0) \geq \eta$*, then the optimal solution of* (2.13) *is given by* $\mathbf{q}^* = (q_1^*, \ldots, q_n^*)$*, where*

$$q_i^* = \frac{p_i e^{-\xi_i}}{\sum_{l=1}^n p_l e^{-\xi_l}} \tag{2.15}$$

Otherwise, if $\kappa(\mathbf{p}, \boldsymbol{\xi}, 0) < \eta$*, then*

$$q_i^* = \frac{p_i^{\frac{1}{1+\alpha^*}} e^{-\frac{\xi_i}{1+\alpha^*}}}{\sum_{l=1}^n p_l^{\frac{1}{1+\alpha^*}} e^{-\frac{\xi_l}{1+\alpha^*}}} \tag{2.16}$$

where α^* *is a positive root (potentially* ∞*) of the equation* $\kappa(\mathbf{p}, \boldsymbol{\xi}, \alpha) = \eta$*.*

Proposition 2.4.2 entails that the subprogram in each iteration can be solved as a one-dimensional root-finding problem, which can be implemented efficiently by, e.g., a bisection search.

Convergence Analysis

The MDSA algorithm is depicted in Algorithm 1. Steps 1 and 2 there come from the discussion in Sections 2.4.2 and 2.4.2 respectively. Step 3 is a technical step in

securing theoretical convergence (as will be shown momentarily). We point out that Steps 2 and 3 combined are in effect solving (2.5) with $\mathcal{P}$ replaced by $\mathcal{P}(\epsilon)$, a restricted version of the original (2.5). Step 2 first solves (2.5) under $\mathcal{P}$. If its optimal solution lies in $\mathcal{P}(\epsilon)$, then this is immediately an optimal solution for the restricted problem. Otherwise, Step 3 is carried out to find the optimal solution for the restricted problem directly.

Algorithm 1 MDSA for solving (2.4)

Input: A small parameter $\epsilon > 0$, initial solution $\mathbf{p}^1 \in \mathcal{P}(\epsilon) = \{\mathbf{p} : \sum_{i=1}^n p_i = 1, p_i \geq \epsilon \text{ for } i = 1, \ldots, n\}$, a step size sequence γ^k, and sample sizes M_1 and M_2.

Iteration: For $k = 1, 2, \ldots$, do the following: Given $\mathbf{p}^k$,

1. Estimate $\hat{\boldsymbol{\psi}}^k = (\hat{\psi}_1^k, \ldots, \hat{\psi}_n^k)$ with

$$\hat{\psi}_i^k = 2 \sum_{j=1}^m \frac{1}{M_1} \sum_{r=1}^{M_1} (\phi_j(h(\mathbf{X}^{(r)})) - \mu_j) \frac{1}{M_2} \sum_{r=1}^{M_2} \phi_j(h(\tilde{\mathbf{X}}^{(r)})) S_i(\tilde{\mathbf{X}}^{(r)}; \mathbf{p}^k)$$

where $\mathbf{X}^{(r)}$ and $\tilde{\mathbf{X}}^{(r)}$ are M_1 and M_2 independent copies of the input process generated under i.i.d. replications of $\mathbf{p}^k$, which are used simultaneously for all components of $\hat{\boldsymbol{\psi}}^k$.

2. If $\kappa(\mathbf{p}^k, \gamma^k \hat{\boldsymbol{\psi}}^k, 0) \geq \eta$ where κ is defined in (2.14), then output $\mathbf{p}^{k+1} = (p_1^{k+1}, \ldots, p_n^{k+1})$, where

$$p_i^{k+1} = \frac{p_i^k e^{-\gamma^k \hat{\psi}_i^k}}{\sum_{l=1}^n p_l^k e^{-\gamma^k \hat{\psi}_l^k}}$$

Otherwise, if $\kappa(\mathbf{p}^k, \gamma^k \hat{\boldsymbol{\psi}}^k, 0) < \eta$, then

$$p_i^{k+1} = \frac{p_i^{k \frac{1}{1+\alpha^k}} e^{-\frac{\gamma^k \hat{\psi}_i^k}{1+\alpha^k}}}{\sum_{l=1}^n p_l^{k \frac{1}{1+\alpha^k}} e^{-\frac{\gamma^k \hat{\psi}_l^k}{1+\alpha^k}}}$$

where α^k is a positive root (potentially ∞) of the equation $\kappa(\mathbf{p}^k, \gamma^k \hat{\boldsymbol{\psi}}^k, \alpha) = \eta$.

3. If $p_i^{k+1} < \epsilon$ for some i, then solve the convex optimization (2.5) but with $\mathcal{P}$ replaced by the set $\mathcal{P}(\epsilon)$. Output its solution as $\mathbf{p}^{k+1}$.

The reason why we consider such a restricted problem is to guarantee that $\mathbf{p}^k$ does not have any components that are too small. In turn, this is because the form of the gradient estimator $\hat{\boldsymbol{\psi}}^k$ contains p_i^k at the denominator, and a small p_i^k can blow

up its variance. By restricting p_i^k to be at least ϵ, the variance of $\hat{\boldsymbol{\psi}}^k$ is bounded, and convergence of Algorithm 1 can be shown in situations where the optimal solution for (2.4) is indeed in $\mathcal{P}(\epsilon)$.

We mention that Step 3 is really a technicality for theoretical correctness and does not seem to have practical implications. All the experiments we perform (in Section 2.5) do not run into the problem of vanishing p_i^k. For this reason we do not attempt to find analytical solution for the optimization (2.5) under $\mathcal{P}(\epsilon)$ but rather just impose it as a general convex optimization problem.

Theorem 2.4.2. *Suppose the assumptions in Proposition 2.4.1 Part 2 hold with $l = 2$. Assume there exists a unique optimal solution $\mathbf{p}^* \in \mathcal{P}(\epsilon)$ for (2.4) such that $\boldsymbol{\psi}(\mathbf{p})'(\mathbf{p} - \mathbf{p}^*) = 0$ if and only if $\mathbf{p} = \mathbf{p}^*$. Choose the step size sequence $\{\gamma^k\}$ such that*

$$\sum_{k=1}^{\infty} \gamma^k = \infty, \quad \sum_{k=1}^{\infty} (\gamma^k)^2 < \infty$$

Then $\mathbf{p}^k$ generated in Algorithm 1 converges to $\mathbf{p}^*$ a.s..

The condition $\boldsymbol{\psi}(\mathbf{p})'(\mathbf{p} - \mathbf{p}^*) = 0$ is a generalization of the first order local optimality condition $\boldsymbol{\psi}(\mathbf{p}) = \mathbf{0}$ in unconstrained optimization. It is in line with the standard condition $\boldsymbol{\psi}(\mathbf{p})'(\mathbf{p} - \mathbf{p}^*) > 0$ for all $\mathbf{p} \neq \mathbf{p}^*$ used in the SA literature (e.g., (Benveniste et al., 2012; Broadie et al., 2011)).

The proof of Theorem 2.4.2 follows the framework in (Blum, 1954), which considers SA on unconstrained problems.

2.5 Numerical Results

We provide numerical illustration of our method. We focus on a stylized M/G/1 queue, where we assume known i.i.d. exponential interarrival time distribution. Our goal is to calibrate the unknown i.i.d. service time distribution. In each experiment, we generate N i.i.d. realizations of $h(\mathbf{X})$ under a "true" service time distribution, for

a specified function h. We take these N realizations as our output data, and apply our formulation (2.3) and the stochastic quadratic penalty method in Section 2.4 to infer the service time distribution.

To apply the stochastic quadratic penalty method, we need to solve (2.4) for a grid of η such that η^* can be approximately located. We set the maximum search value of η to be the largest possible entropy $\log n$, and the minimum search value such that it is clearly far below the threshold η^*. This requires empirically keeping track of the approximate value of $W^*(\eta)$, the optimal value in (2.4), as we decrement η from $\log n$.

Denote by A_t the interarrival time between the t-th and $(t+1)$-st customers, and by X_t the service time of the t-th customer. In all our examples shown below, we let the true service time be a discrete random variable over the set $\{1/n, 2/n, \ldots, (n-1)/n, 1\}$, i.e. $z_i = i/n$, and $\mathbf{p} = (p_1, \ldots, p_n)$ is obtained by

$$p_i := \int_{\frac{i-1}{n}}^{\frac{i}{n}} f(x)dx, \tag{2.17}$$

for some continuous probability density function $f(x)$. We use the moment function $\phi_j(y) = I(y \leq c_j), j = 1, 2, \ldots, m$ where c_j's are some quantiles. Rather than fixing these c_j, we choose them as the $i/(m+1)$-quantiles of the N data points. Though introducing a small bias in our procedure, this gives us a reasonable set of quantiles to match against our simulation outputs.

We will use two choices of h, vary the support size n, the number of quantile-based moments m, output data size N, and the function $f(x)$ in our experiments.

Our first choice of $h(\mathbf{X})$ is the time-averaged number of customers in the system

during a busy period. Specifically,

$$\begin{aligned} h\left(X_1, X_2, \ldots, X_\tau\right) &= \frac{\sum_{t=1}^{\tau}\left(D_t - E_{t-1}\right)}{E\tau}, \\ D_t = \sum_{k=1}^{t} X_k, E_t &= \sum_{k=1}^{t} A_k, E_0 = 0, \\ \text{where } \tau := \min &\left\{ t \geq 1 : \sum_{k=1}^{t} A_k > \sum_{k=1}^{t} X_k \right\}. \end{aligned} \tag{2.18}$$

We set A_t as unit mean exponentials. Here we show that the gradient estimator (2.9) is valid for this setting. Note that the assumption of finite moment in Proposition 2.4.1 Part 2 trivially holds since ϕ_j's are indicator functions. As for the stopping time τ, if the probability measure governing X_t puts positive weight on each $\frac{d}{n}, d = 1, \ldots, n$, we have $E[X_t] < 1 = E[A_t]$. It is obvious that $E[e^{\theta(X_t - A_t)}] < \infty$ for some $\theta > 0$. Therefore

$$\begin{aligned} P(\tau > T) &= P\left(\sum_{t=1}^{s}(X_t - A_t) \geq 0, \forall s \leq T\right) \\ &\leq P\left(\sum_{t=1}^{T}(X_t - A_t) \geq 0\right) \leq e^{-\delta T}, \text{ for all } T \geq 1, \end{aligned}$$

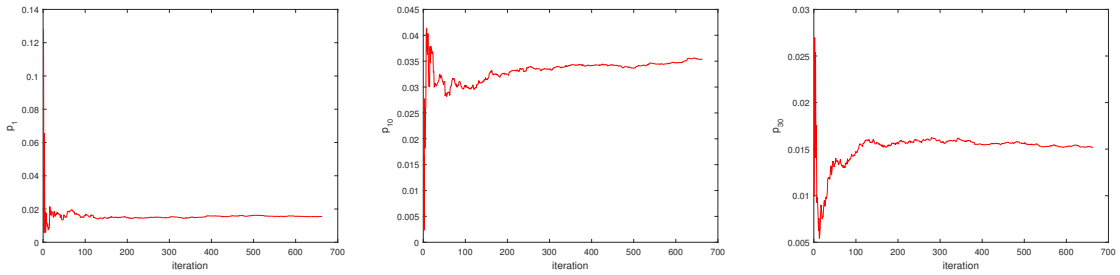
for some $\delta > 0$, where the last inequality follows from Lemma 2.6.2 in (Durrett, 2010). This verifies the assumption on τ because an exponentially decaying tail implies finiteness of moment generating function at a neighborhood of zero. We set $f(x)$ as a uni-modal density

$$f(x) = \frac{1}{\text{Beta}\left(2, 4\right)} x\left(1 - x\right)^3, \tag{2.19}$$

and $n = 50$.

We first set the number of observed data $N = 10^5$, a large size so that the empirical moments are very close to the true moments. This is to illustrate the efficacy of our method without introducing another layer of errors due to insufficient output data.

The parameters of Algorithm 1 are set to be $\gamma^k = 10/k, M_1 = 300, M_2 = 300$. The algorithm is terminated as soon as one of the following conditions is met: (1) the difference between the average of the last 50 iterates and that of the last 51 to 100 iterates is less than 1×10^{-3}, (2) the number of iteration exceeds 1×10^3. Figure 2·1 shows a set of typical trace plots with $n = 50$ for different components of $\mathbf{p}^k$ as Algorithm 1 progresses. We can see the evidence of convergence under our stopping criterion and parameter setting.

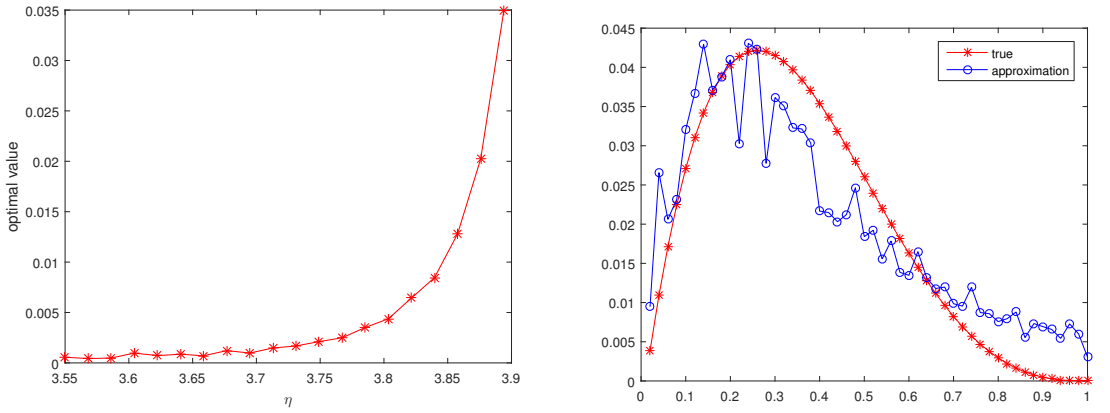


(a) p_1 **(b)** p_{10} **(c)** p_{30}

Figure 2·1: Trace plots of different components of the input probability vector $\mathbf{p}$. Support size $n = 50$, number of quantile-based moments $m = 9$. The algorithm terminates after 668 iterations.

Figure 2·2 shows our input reconstruction results by matching $m = 9$ quantile-based moments. Figure 2·2a shows the estimated values of $W^*(\eta)$ over a 20-point grid for η between 3.55 and 3.9. We can see that $W^*(\eta)$ stays at zero until around $\eta = 3.7$, and then starts to increase. Thus we identify η^* as approximately 3.7. Note that, even though our MDSA algorithm only guarantees convergence to local optima, the depicted monotone trend of $W^*(\eta)$ shows evidence that our algorithm lands at the global optima. In general, a non-monotone trend alarms that our MDSA algorithm misses global optima, whereas a monotone trend can serve as a validity check of global convergence. Figure 2·2b shows the optimal $\mathbf{p}^*$ obtained at $\eta^* = 3.7$. The shape of the reconstructed mass function follows the truth quite well, although

it is not as smooth. Our method is capable of locating the mode, and correctly gives a decreasing trend towards both sides of the mode.

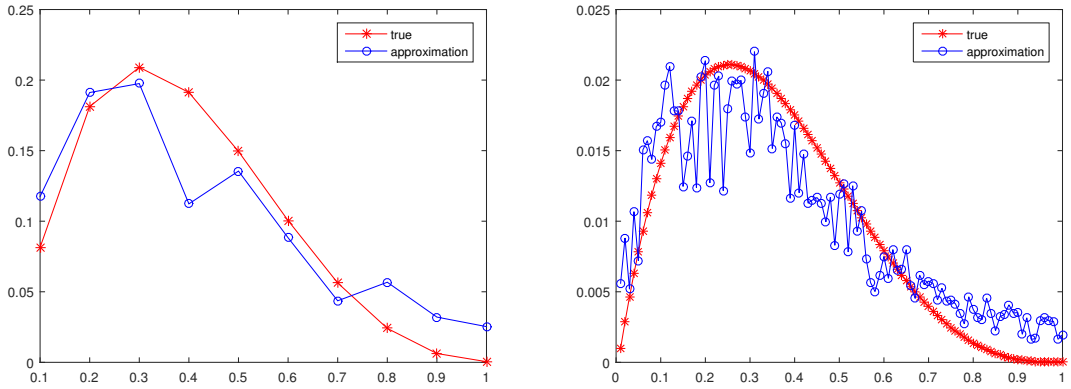


(a) Optimal value of (2.4) against η. Run time = 11 min.

(b) $N = 10^5, n = 50, m = 9, \eta^* = 3.713$.

Figure 2·2: Optimal values of (2.4) and reconstruction performance in the uni-modal case.

Next we vary the support size n to investigate its effect on the quality of the estimate yielded by our method. Figures 2·3 (a) and (b) show the reconstructed distribution, i.e., the estimated $\mathbf{p}^*(\eta^*)$, at $n = 10$ and 100 respectively. The reconstructed distributions both follow quite closely the shape of the truth. The one for $n = 100$ is noticeably more bumpy. This is because n represents essentially the number of parameters in our estimation. As n grows, the estimation variance increases, where the variance comes from both the output data noise and the simulation noise in our reconstruction.



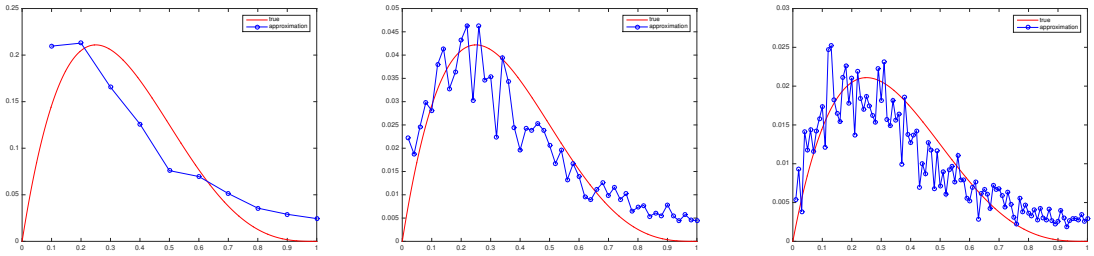
(a) $N = 10^5, n = 10, m = 9, \eta^* = 2.121$, run time = 7 min.

(b) $N = 10^5, n = 100, m = 9, \eta^* = 4.409$, run time = 13 min.

Figure 2·3: Reconstructed versus the true distribution in the unimodal case for different n.

Although our analyses and experiments above assume discrete true distributions, we show how our method can be applied in the case of *continuous* true distributions, where we can calibrate using discrete distributions as approximation. Figure 2·4 illustrates the reconstructed distributions, using different n and support points $z_k = k/n$, to recover a continuous service time distribution with its density given by (2.19) based on output data of the time-averaged number of customers in the system during a busy period. To make the comparison fair, we scale down the true density function by a factor of $1/n$ in each of the three cases to obtain the red curves in each of the three plots (this is because the mass at z_k would be approximately $1/n$ times the original density value at z_k, if the true density is discretized over the set of z_k's). The results are similar to the discrete true distribution setting. In all of the cases $n = 10$, 50, and 100, the general shape of the truth can be recovered. The case $n = 10$ has low resolution and cannot capture the sharp shrink of density at the left end. When $n = 50$ and $n = 100$ we can recover it relatively well including near the end points, but the case $n = 100$ jitters more as the estimation variance grows. In general, there

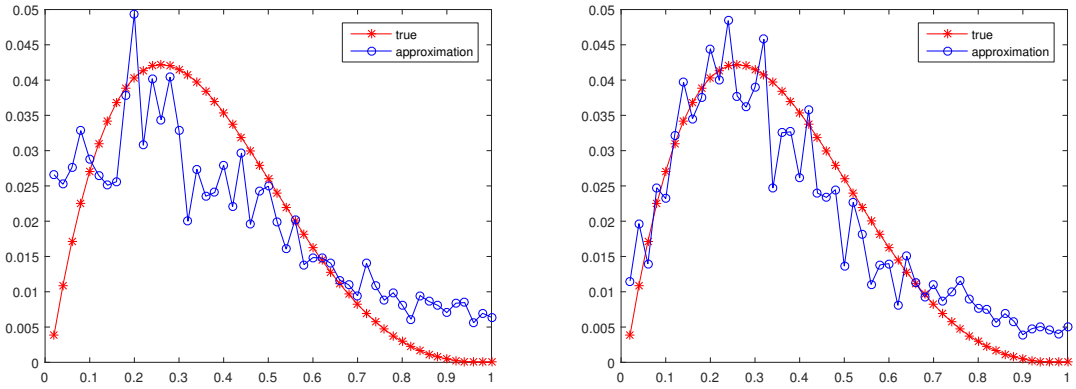
is a trade-off in choosing an optimal n: a small n leads to a small variance but large bias, whereas a large n leads to a large variance but small bias. In our example one can argue that $n = 50$ achieves a balance and is the best among the three cases in capturing the characteristic of the continuous density.



(a) $n = 10$ **(b)** $n = 50$ **(c)** $n = 100$

Figure 2·4: Reconstructed versus the true distribution for different n, for continuous true distribution.

Next we investigate the effect of the number of matched quantiles m. Figures 2·5 (a) and (b) show the results when we match $m = 4$ and 14 quantiles respectively. While $m = 4$ still appears acceptable, we can see considerable gain at $m = 14$, with the distribution matching almost perfectly for the most part. Compared with $m = 9$ (Figure 2·2 (b)), $m = 14$ shows slightly better fit. Here, the data size $N = 10^5$ seems big enough to support matching as many as 14 quantiles without overfitting.

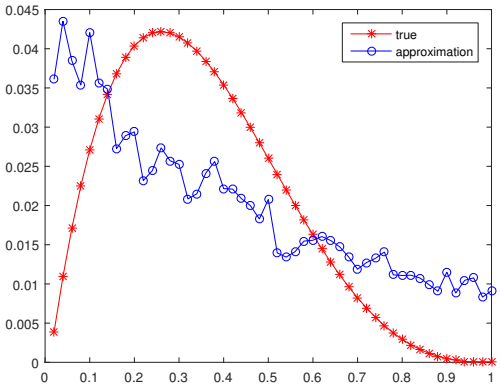


(a) $N = 10^5, n = 50, m = 4, \eta^* = 3.768$, run time = 8 min.

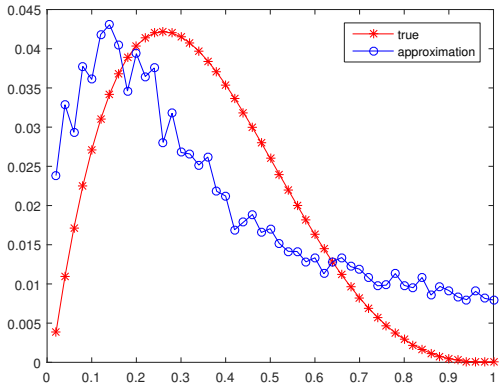
(b) $N = 10^5, n = 50, m = 14, \eta^* = 3.694$, run time = 12 min.

Figure 2·5: Reconstructed versus the true distribution in the unimodal case for different m.

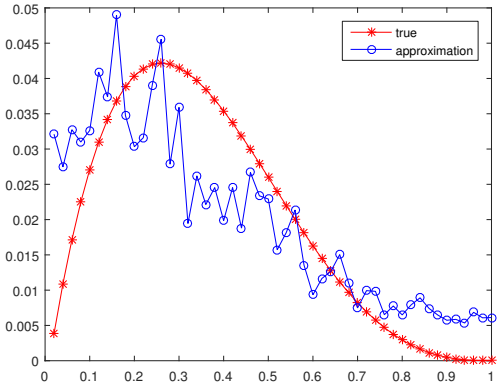
We now test our method in the case of a moderate number of output data points N, to mimic a situation that might occur often in practice. Figure 2·6 shows the reconstructed distribution compared with the truth, for the four combinations of $N = 200$ or 500, and $m = 4$ or 9. What we observe is that gathering more data (from 200 to 500) does not elicit dramatic improvements. The case of $N = 200$ and $m = 4$ cannot capture the shape of the left tail. However, the case of $N = 200$ and $m = 9$ seems to be able to recover the left tail behavior better. Of course, we are exploring one particular realization of the data, and the details of the reconstruction accuracy could be subject to statistical noise. Nonetheless, the plots show that using a relatively large number of quantiles can still give promising results with the availability of even only 200 observations.



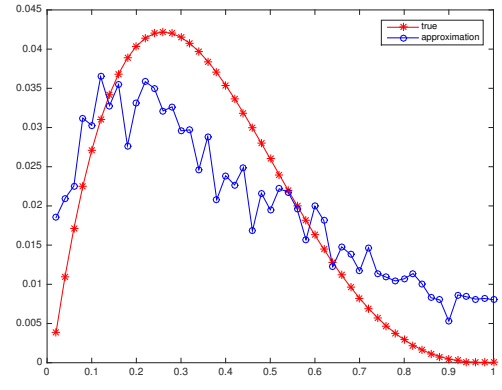
(a) $N = 200, n = 50, m = 4, \eta^* = 3.807$, run time = 7 min.



(c) $N = 200, n = 50, m = 9, \eta^* = 3.768$, run time = 11 min.



(b) $N = 500, n = 50, m = 4, \eta^* = 3.729$, run time = 8 min.



(d) $N = 500, n = 50, m = 9, \eta^* = 3.803$, run time = 12 min.

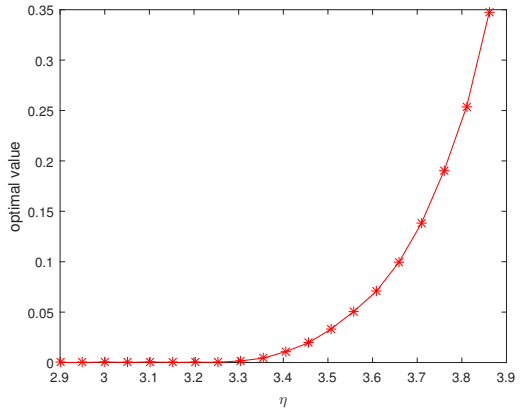
Figure 2·6: Reconstructed versus the true distribution in the uni-modal case with fewer output data sizes.

Next we test our method on a monotone and a bi-modal distribution. Figures 2·7 and 2·8 show the results where the true probability mass functions of the service time X_1 are given by (2.17) with

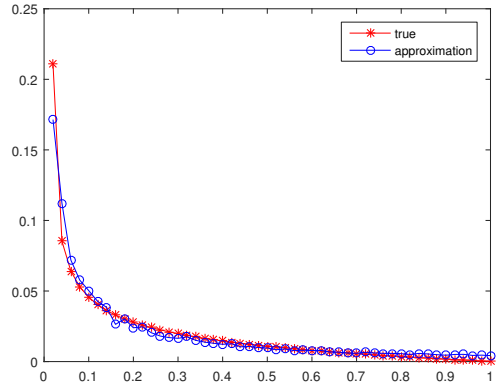
$$f(x) = \frac{1}{\text{Beta}\left(\frac{1}{2}, 2\right)} x^{-\frac{1}{2}} (1-x) \quad \text{and} \quad \frac{0.4}{\text{Beta}\,(5,2)} x^4 (1-x) + \frac{0.6}{\text{Beta}\,(3,9)} x^2 (1-x)^8,$$

respectively. We use $n = 50$ and $m = 9$ for both settings. Figure 2·7a shows the

values of $W^*(\eta)$ on a grid between $\eta = 2.9$ and 3.9. The η^* can be approximately located at 3.3. Figure 2·7b shows that our reconstructed input distribution recovers the monotonicity of the true mass function very well. On the other hand, Figures 2·8a and 2·8b show the results for a bi-modal distribution, where the reconstruction seems unable to capture the two modes. However, even in this unfavorable setup, we seem to be capable of capturing the overall trend of more masses on the left than on the right. Figures 2·9a and 2·9b also show the similar behavior of our reconstructions when the true distributions are continuous.

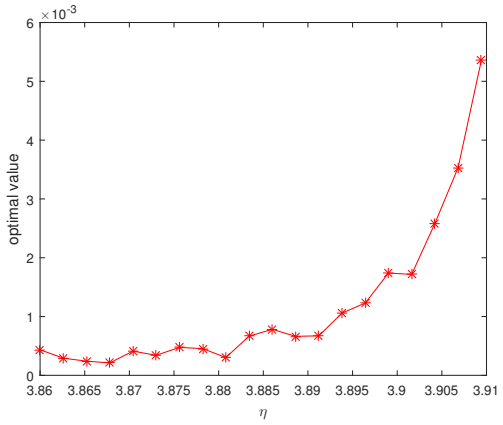


(a) Optimal value of (2.4) against η. Run time = 9 min.

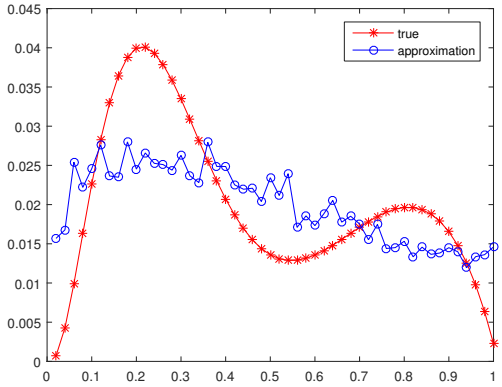


(b) $N = 10^5, n = 50, m = 9, \eta^* = 3.305$.

Figure 2·7: Optimal values of (2.4) and reconstruction performance in the monotone case.

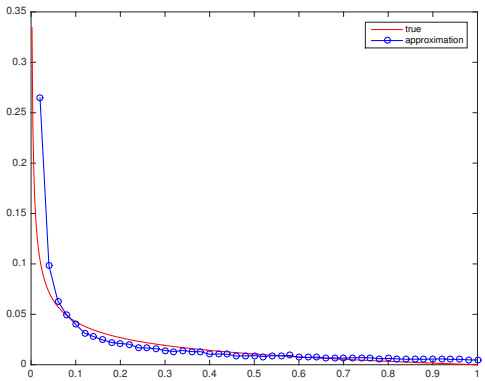


(a) Optimal value of (2.4) against η. Run time = 11 min.

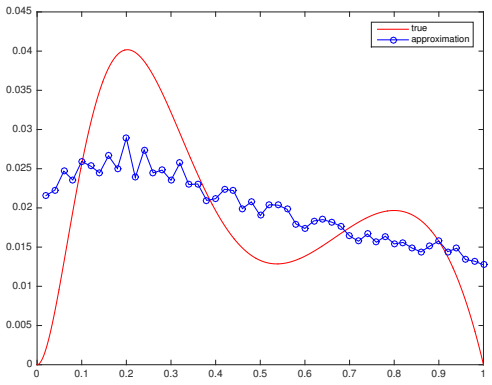


(b) $N = 10^5, n = 50, m = 9, \eta^* = 3.883$

Figure 2·8: Optimal values of (2.4) and reconstruction performance in the multi-modal case.



(a) Monotone case.



(b) Multi-modal case.

Figure 2·9: Continuous true distribution. $n = 50$.

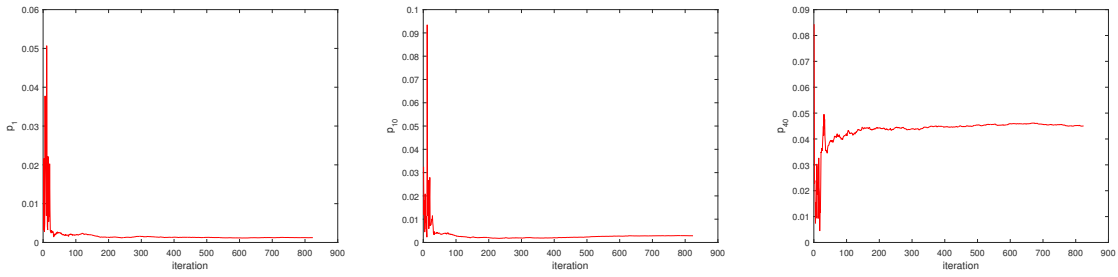
Our second choice of $h(\mathbf{X})$ is the average waiting time of the first 50 customers

after starting from an empty single-server queue. In this case,

$$h\left(X_1, X_2, \ldots, X_T\right) = \frac{1}{T+1} \sum_{t=1}^{T+1} W_t, \tag{2.20}$$

$$\text{where } T = 49,\ W_t = \max\left\{0, W_{t-1} + X_{t-1} - A_{t-1}\right\} \text{ for } t \geq 2,\ W_1 := 0.$$

Note that the validity of the gradient estimator (2.9) is obvious because all ϕ_j's are indicator functions and τ is a deterministic time. We let A_t follow a known exponential distribution with mean 1/5. The support size n and the number of quantile-based moments m are set to be 50 and 9, respectively. The same algorithmic parameter setting and stopping criterion as in the previous set of experiments are adopted. Typical trace plots are given in Figure 2·10 to demonstrate that the algorithm does converge.



(a) p_1 **(b)** p_{10} **(c)** p_{40}

Figure 2·10: Trace plots of different components of the probability vector $\mathbf{p}$. Support size $n = 50$, number of quantile-based moments $m = 9$. The algorithm terminates after 619 iterations.

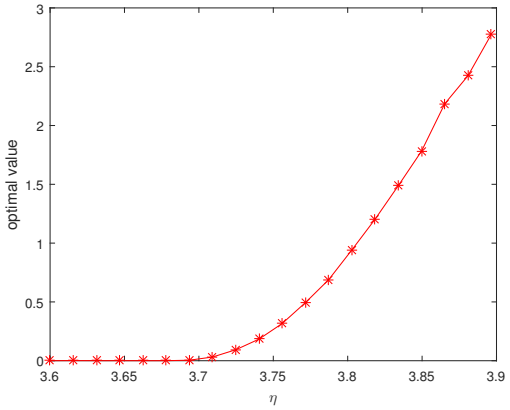
As in the previous set of experiments, our method is tested in the cases of monotone (Figure 2·11), uni-modal (Figure 2·12), and multi-modal (Figure 2·13) distributions, where the underlying probability functions of the service time X_1 are again

given by (2.17) but with $f(x)$ now replaced by

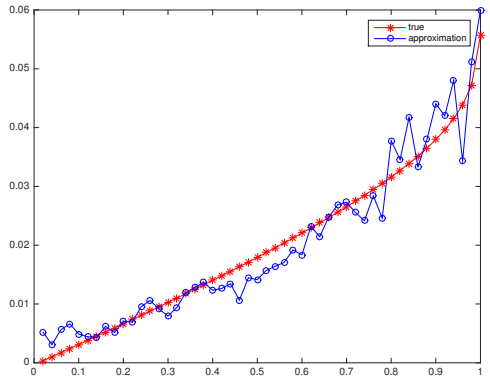
$$\frac{1}{\text{Beta}\,(2, 0.9)} x\,(1-x)^{-0.1}\,, \qquad \frac{1}{\text{Beta}\,(2, 4)} x\,(1-x)^{3}\,,$$

$$\text{and} \quad \frac{0.4}{\text{Beta}\,(2, 5)} x\,(1-x)^{4} + \frac{0.6}{\text{Beta}\,(9, 3)} x^{8}\,(1-x)^{2}\,,$$

respectively. The results are generally similar to the previous set of experiments. Our method recovers the monotone distribution very well. It cannot fully recover the peaks in the other two cases, but it can capture the overall trend of more masses on the left than the right end. Figure 2·14 demonstrates the same behavior in the case when the true distribution is continuous.

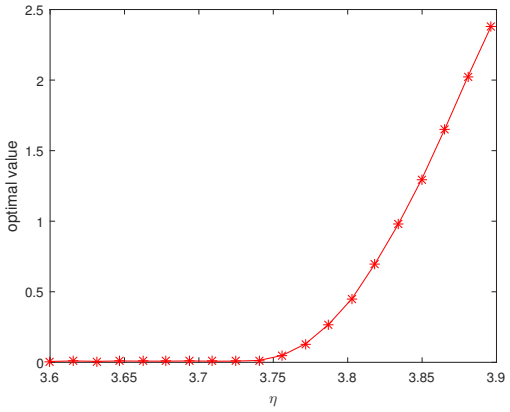


(a) Optimal value of (2.4) against η. Run time = 30 min.

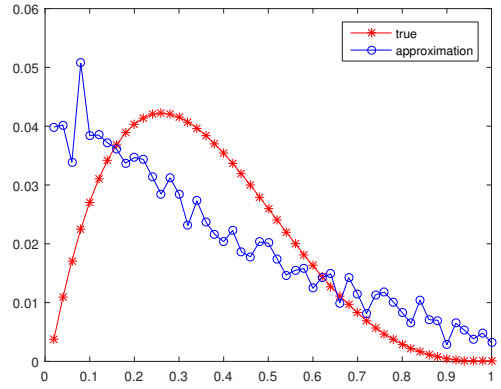


(b) $N = 10^5, n = 50, m = 9, \eta^* = 3.678$.

Figure 2·11: Optimal values of (2.4) and reconstruction performance in the monotone case.

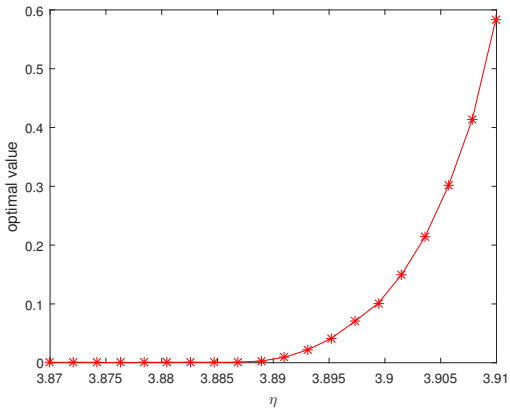


(a) Optimal value of (2.4) against η. Run time = 27 min.

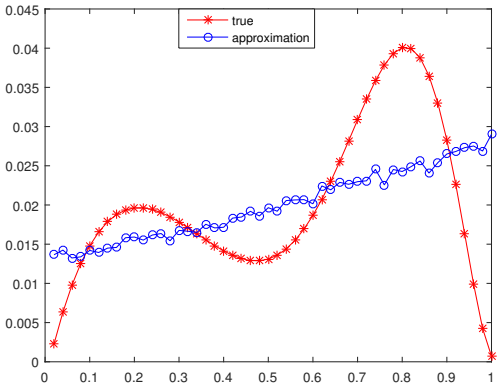


(b) $N = 10^5, n = 50, m = 9, \eta^* = 3.725$.

Figure 2·12: Optimal values of (2.4) and reconstruction performance in the uni-modal case.

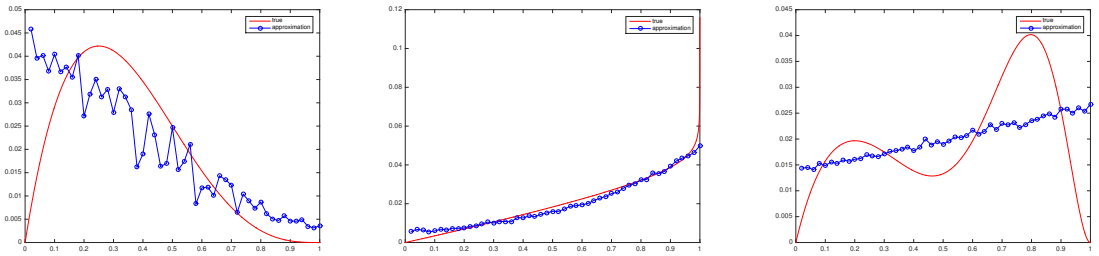


(a) Optimal value of (2.4) against η. Run time = 26 min.



(b) $N = 10^5, n = 50, m = 9, \eta^* = 3.887$.

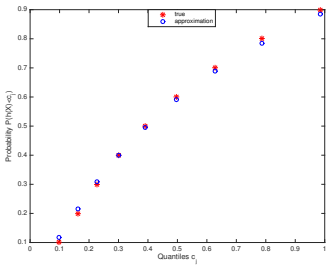
Figure 2·13: Optimal values of (2.4) and reconstruction performance in the multi-modal case.



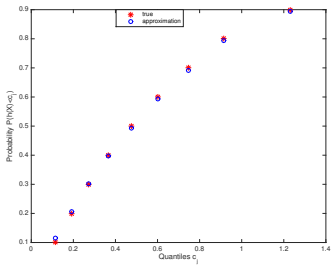
(a) Uni-modal case. **(b)** Monotone case. **(c)** Multi-modal case.

Figure 2·14: Continuous true distribution. $n = 50$. Y = average wait of first 50 customers starting from an empty system.

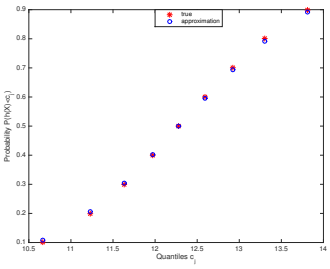
Finally, Figure 2·15 compares the output probabilities at the matched quantiles from the simulation outputs generated using our reconstructed input model, with those from the output data. We illustrate this comparison for uni-modal, monotone, and bi-modal truths, and for the two types of outputs we have considered. We see that the probabilities are matched very well in all cases. This suggests that for the multi-modal case, the mismatch between our reconstructed distribution and the truth is due to either the insensitivity of the output distribution, or the incapability of the quantiles we use in unveiling the shape of the input distribution.



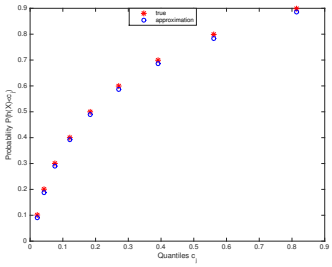
(a) Uni-modal case for time-averaged # of customers, i.e., Figure 2·2b.



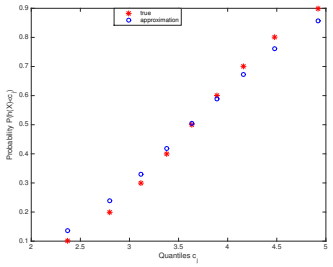
(c) Multi-modal case for time-averaged # of customers, i.e., Figure 2·8b.



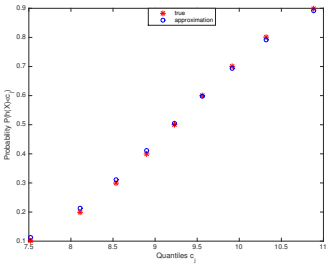
(e) Monotone case for 50-customer average wait, i.e., Figure 2·11b.



(b) Monotone case for time-averaged # of customers, i.e., Figure 2·7b.



(d) Uni-modal case for 50-customer average wait, i.e., Figure 2·12b.



(f) Multi-modal case for 50-customer average wait, i.e., Figure 2·13b.

Figure 2·15: Moment matches between the simulation output and the true output.

2.6 Summary

We have studied a framework to calibrate the input models in stochastic simulation with only the availability of output data. This inverse model calibration problem appears to be generally understudied in the simulation literature, yet it could arise in many contexts. We have proposed a moment-based approach to nonparametrically infer the input by matching moment statistics, such as quantile probabilities, at the output level. To alleviate the non-identifiability issue, we reduce our model search space by maximizing the entropy among all moment-matching models.

This formulation in general gives a simulation optimization problem that consists

of stochastic nonlinear equality constraints. We have converted this difficult problem into a sequence of simulation optimization programs with deterministic convex constraints, and subsequently proposed an MDSA algorithm to solve them efficiently. We have analyzed the convergence properties of our method. Our numerical experiments show that the reconstructed input distributions are generally capable of capturing the overall trends of the truths. They perform particularly well in the case of simple distributions, but with some degradation in the case of more complex distributions.

Chapter 3

A Frequentist Perspective on Hierarchical Poisson Convolution Models for Topic Allocation

3.1 Introduction and Motivation

This chapter is a discussion based on the paper: (Airoldi and Bischof, 2015) *A regularization scheme on word occurrence rates that improves estimation and interpretation of topical content*. At the heart of their work, as indicated in the title, is a novel regularization scheme, which is intended to address certain shortcomings of existing methods in the literature on dimensionality reduction principles and techniques for topic modeling in document analysis. Multilevel models are a popular set of methods that are considered flexible and powerful in finding latent structure in high dimensional data (McLachlan and Peel, 2004) . However, although these models are successfully reducing the dimension, there is no guarantee that the resulting low-dimensional projections they produce are particularly interpretable in terms of quantities of scientific interest. In (Airoldi and Bischof, 2015), the authors propose a new regularization scheme, that incorporates how words are used differentially across topics, and results in more interpretable summaries. The proposed novel regularization is carefully motivated, and empirical results thoroughly demonstrate its effectiveness. At the same time, it can be said that the regularization is rather complex and, as a result, interpretation arguably suffers to some extent, particularly at a first reading. Accordingly,

we have taken as our goal in this discussion to attempt to lend further insight into the nature of the regularization proposed in (Airoldi and Bischof, 2015). Towards this end, while the authors take a formally Bayesian approach to modeling and estimation, here we adopt for our purpose the perspective of complexity-penalized regularization, as an alternative lens through which to view the authors' contributions. Throughout we consider certain simplifications of the assumptions of the proposed model, where we feel doing so lends additional insight, hopefully without excessive loss of fidelity to the original.

3.2 Hierarchical Poisson Convolution Formulation

In the work of (Airoldi and Bischof, 2015), the hierarchical Poisson convolution (HPC) model, conditional on the topic hierarchy tree, can be summarized by the graphical model diagrammed in our Figure 3·1.

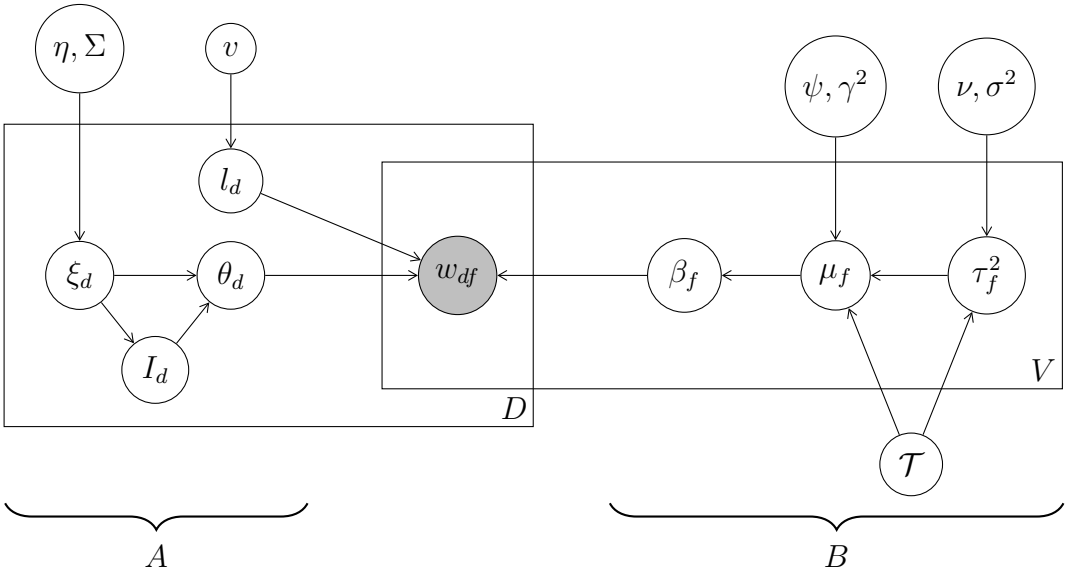


Figure 3·1: Graphical model diagram of the HPC model. Plates indicate replication, outside circles are hyper-parameters for priors, and shading means a quantity is observed. (Note: I_d is not necessarily assumed observed here.)

As indicated in the figure, structure on the word frequency matrix W is provided by imposing structure on documents (left) and words (right). Let $\boldsymbol{\beta}_f$ be a $K \times 1$ vector of occurrence rates for word $f \in \{1, \ldots, V\}$, across all K topics in the topic hierarchy. Define $\boldsymbol{\alpha}_d = l_d \boldsymbol{\theta}_d$, where l_d is a scalar and $\boldsymbol{\theta}_d$ is a $K \times 1$ vector containing the proportion with which document $d \in \{1, \ldots, D\}$ belongs to each one of the K topics. Below we describe the HPC generative process:

- **Topic membership parameters**

  Let ξ_d be a $(K \times 1)$ topic affinity vector.

  For document $d \in \{1, \ldots, D\}$

  $$\xi_d \sim N(\eta, \Sigma)$$

  For topic $k \in \{1, \ldots, K\}$

  $$I_{dk} \sim Bern\left(\frac{1}{1 + e^{-\xi_{dk}}}\right)$$

  the role of I_{dk} is to indicate whether topic k is active in document d.

  Define

  $$\theta_{dk}(I_d, \xi_d) = \frac{e^{\xi_{dk}} I_{dk}}{\sum_{j=1}^{K} e^{\xi_{dj}} I_{dj}}$$

- **Tree parameters**

  In (Airoldi and Bischof, 2015), the known hierarchy of topics is utilized by assuming that words are used similarly in topics that are neighbors on the tree $\mathcal{T}$. The tree structure is shown in Figure 3·2.

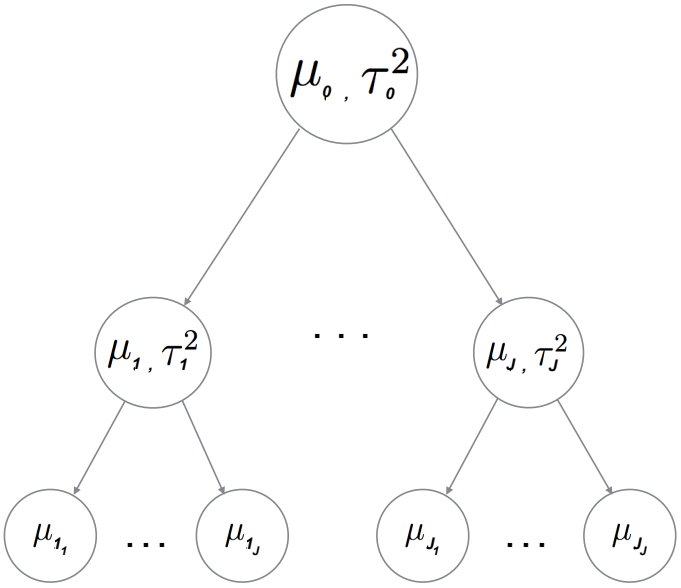


Figure 3·2: Details of the tree plate from Figure 3·1.

For word $f \in \{1, \ldots, V\}$

$$
\begin{aligned}
\mu_{f0} \sim & N(\psi, \gamma^2) \\
\tau_{f0}^2 \sim & \text{Inv-}\chi^2(\nu, \sigma^2) \\
& \forall j \in \{1, \ldots, J\} : \\
& \quad \mu_{fj} \sim N(\mu_{f0}, \tau_{f0}^2) \\
& \quad \tau_{fj}^2 \sim \text{Inv-}\chi^2(\nu, \sigma^2) \\
& \quad \forall j \in \{1, \ldots, J\} : \\
& \quad\quad \mu_{fj_1}, \ldots, \mu_{fj_J} \sim N(\mu_{fj}, \tau_{fj}^2)
\end{aligned}
$$

- **Word counts**

$$w_{df} \sim Poisson(\boldsymbol{\alpha}_d^T \boldsymbol{\beta}_f)$$

Therefore, $\mathbb{E}[W] = AB$, where the d-th row of A is $\boldsymbol{\alpha}_d$ and the f-th column of B is $\boldsymbol{\beta}_f$. Hence, ignoring the (important) scaling inherent in the parameters l_d, the proposed model can be viewed usefully as constraining a certain non-negative matrix factorization (NMF), i.e., $\Rightarrow W \approx AB$. This factorization is reflected at the bottom of Figure 3·1 here, and shown explicitly in Figure 3·3.

3.3 From Bayesian to Frequentist Perspective

It can be argued that the regularization described by the generative model above is rather complex and hence, difficult to interpret. To address this, we attempt to lend further insight into the nature of the regularization proposed. To achieve this, instead of the formally Bayesian approach to modeling and estimation, here we adopt for our purpose the perspective of complexity-penalized regularization, as an alternative lens through which to view the novel regularization scheme.

3.3.1 Representing the HPC Model as NMF

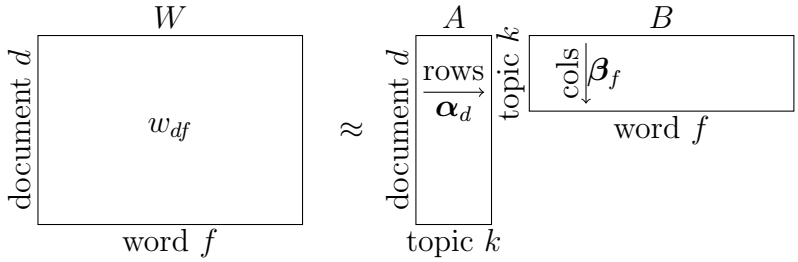


Figure 3·3: Representation of HPC model likelihood parameterization as a non-negative matrix factorization (NMF).

Now consider the structure lent to the matrices A and B in this NMF, through the priors adopted by (Airoldi and Bischof, 2015) in their HPC model. We connect our NMF approach to the original parameterization through a re-derivation of the log-posterior distribution of A and B given the observed word count matrix W, with the

goal of producing a complexity-penalized formulation of the optimization problem underlying the proposed HPC-based estimation of these two matrices.

3.3.2 Derivation of the Log-posterior Likelihood

Writing the log-posterior as

$$\log \mathbb{P}(A, B|W) \approx \log \mathbb{P}(W|A, B) + \log \mathbb{P}(A) + \log \mathbb{P}(B)$$

we begin with the likelihood. Formally, the likelihood is Poisson, given by the equation below:

$$\begin{aligned}\log \mathbb{P}(W|A, B) &= \log \prod_{d=1}^{D} \prod_{f=1}^{V} \frac{(\boldsymbol{\alpha}_d^T \boldsymbol{\beta}_f)^{w_{fd}} e^{-\boldsymbol{\alpha}_d^T \boldsymbol{\beta}_f}}{w_{fd}!} \\ &= \sum_{d=1}^{D} \sum_{f=1}^{V} \left(w_{fd} \log(\boldsymbol{\alpha}_d^T \boldsymbol{\beta}_f) - \boldsymbol{\alpha}_d^T \boldsymbol{\beta}_f - \log(w_{fd}!) \right)\end{aligned}$$

However, in the literature on NMF, various error functions have been proposed, with the most widely used arguably being squared-error loss, see section 14.6 of (Friedman et al., 2001). This suggests approximating the log-likelihood $\log \mathbb{P}(W|A, B)$ by the quantity $||W - AB||_{\mathcal{F}}^2$, where $|| \cdot ||_{\mathcal{F}}$ denotes the Frobenius norm.

Next consider the priors on A and B. Beginning with $\mathbb{P}(A)$, and treating the document lengths l_d as fixed and known, we write

$$\log \mathbb{P}(A) = \log \mathbb{P}\left(\{\boldsymbol{\alpha}_d\}_{d=1}^{D} \right) = \sum_{d=1}^{D} \log \mathbb{P}(l_d \boldsymbol{\theta_d}) = \sum_{d=1}^{D} \log \mathbb{P}(\boldsymbol{\theta}_d) + c \ ,$$

where here and elsewhere c denotes an arbitrary constant (not necessarily the same). Now

$$\mathbb{P}(\boldsymbol{\theta}_d) = \sum_{\boldsymbol{I}_d} \int_{\boldsymbol{\xi}_d} \mathbb{P}(\boldsymbol{\theta}_d|\boldsymbol{I}_d, \boldsymbol{\xi}_d) \mathbb{P}(\boldsymbol{I}_d|\boldsymbol{\xi}_d) \mathbb{P}(\boldsymbol{\xi}_d) m(\boldsymbol{I}_d, \boldsymbol{\xi}_d) \ . \tag{3.1}$$

But note that

$$\mathbb{P}(\boldsymbol{\theta}_d|\boldsymbol{I}_d,\boldsymbol{\xi}_d) = \begin{cases} 1, \text{ iff } supp(\boldsymbol{\theta}_d) = supp(\boldsymbol{I}_d) \text{ and } \boldsymbol{\xi}_d \in \mathcal{A} \\ 0, \text{ otherwise} \end{cases}$$

where

$$\mathcal{A} = \left\{ \boldsymbol{\xi}_d : \text{ for } k \in supp(\boldsymbol{\theta}_d), \theta_{dk} = \frac{e^{\xi_{dk}}}{\sum_k e^{\xi_{dk}}} := f(\xi_{d|k}) \right\} \ .$$

Furthermore, for any $\boldsymbol{\xi}_d$ there is only one $\boldsymbol{I}_d$ that satisfies $supp(\boldsymbol{\theta}_d) = supp(\boldsymbol{I}_d)$. Finally, for $k \notin supp(\boldsymbol{\theta}_d)$, $\boldsymbol{\xi}_d$ can take on any value. Combining these observations and simplifying the resulting expressions, we obtain that

$$\log \mathbb{P}(A) = \sum_{d=1}^{D} -\frac{1}{2\lambda^2} \sum_{k \in supp(\boldsymbol{\theta}_d)} \left(f^{-1}(\boldsymbol{\theta}_d)[k] - \boldsymbol{\eta}[k] \right)^2 + c \ , \tag{3.2}$$

where $[k]$ indicates the k-th entry of a vector and λ is the scale parameter for the (conditional) normal prior on $\boldsymbol{\theta}$.

For $\mathbb{P}(B)$, we can write

$$\log \mathbb{P}(B) = \sum_{f=1}^{V} \log \mathbb{P}\left(\boldsymbol{\beta}_f\right) = \sum_{f=1}^{V} \left(-\mathbf{1}^T \boldsymbol{\mu}_f + \log \mathbb{P}\left(\boldsymbol{\mu}_f\right) \right) \ ,$$

where $\boldsymbol{\mu}_f = \log(\boldsymbol{\beta}_f)$ is the collection of all log-rates in the tree for word f. Now suppose the dispersion parameters $\tau_{f,k}^2$ are treated as fixed and known. In the HPC model the elements $\mu_{f,k}$ of $\boldsymbol{\mu}_f$ are then conditionally independent normal in a Markov fashion down the topic hierarchy tree, from root to leaves. So, ignoring the contribution of the corpus-level term in the prior, $\log \mathbb{P}(B)$ can be expressed as

$$\sum_{f=1}^{V} \left(-\mathbf{1}^T \boldsymbol{\mu}_f - \sum_{j \in int(\mathcal{T})} \frac{1}{2\tau_{f,j}^2} ||\boldsymbol{\mu}_{f,ch(j)} - \mu_{f,j}\mathbf{1}||_2^2 \right) \ , \tag{3.3}$$

where $int(\mathcal{T})$ is the set of interior nodes (i.e., non-leaves) of the topic tree $\mathcal{T}$ and $ch(j)$ denotes the children of node j in $\mathcal{T}$.

Combining the above arguments, we arrive at the following complexity-penalized NMF problem as an approximation of the posterior maximization posed in the paper:

$$\min_{A,B} \left\{ ||W - AB||_{\mathcal{F}}^2 + \underbrace{\lambda_1 \sum_{d=1}^{D} \|\boldsymbol{\xi}_d(A) - \boldsymbol{\eta}\|^2}_{\substack{\text{over active topics} \\ \text{regularization} \\ \text{on rows of } A}} + \underbrace{\sum_{f=1}^{V} \left(\mathbf{1}^T \boldsymbol{\mu}_f(B) + \sum_{j \in int(\mathcal{T})} \frac{1}{2\tau_{f,j}^2} ||\boldsymbol{\mu}_{f,ch(j)}(B) - \mu_{f,j}(B)\mathbf{1}||_2^2 \right)}_{\text{regularization on cols of } B} \right\} \tag{3.4}$$

3.4 Discussion of the Advantages and Disadvantages of the Complexity Penalized Likelihood Formulation

The representation in (3.4) allows us finally to make several observations.

1. The posterior-based estimation strategy associated with the HPC model can be viewed, to a reasonable extent, as being in the family of NMF solutions with ℓ_2-based penalties. Previously, for example, (Pauca et al., 2004) have applied a penalty proportional to $||B||_{\mathcal{F}}^2$, while (Pauca et al., 2006) have incorporated both $||A||_{\mathcal{F}}^2$ and $||B||_{\mathcal{F}}^2$. However, in the current paper there are at least three key differences: (a) the nonlinear and atomized fashion (i.e., over active topics only) in which A enters the penalty; (b) the hierarchical nature of the ℓ_2 penalty for B; and (c) the addition of the linear term $\mathbf{1}^T \boldsymbol{\mu}_f(B)$. Furthermore, we note that B is penalized on a logarithmic scale (i.e., since $\boldsymbol{\mu}_f = \log(\boldsymbol{\beta}_f)$) and that the penalty on the log-rates of words in columns of B differs markedly from $||B||_{\mathcal{F}}^2$. The regularization on the columns of B that we arrive at combines principles of ℓ_2 penalties with the use of hierarchies that is popular in topic modeling (e.g., (Blei et al., 2010)). The manner in which children log-rates are shrunk towards their parents can be interpreted as a variant of the ridge fusion penalty, discussed in (Price et al., 2015), along paths from root to leaves. Note too that, where the $\boldsymbol{\mu}_f$ are positive, we have $\mathbf{1}^T \boldsymbol{\mu}_f(B) = ||\boldsymbol{\mu}_f||_1$, in which case

it is perhaps tempting to think of the penalty on B in the spirit of a convex combination of ℓ_1 and ℓ_2 norms.

2. From a computational perspective, the optimization in (3.4) is somewhat non-standard. Suppose the elements ξ are unconstrained. The last two terms of the objective function (i.e., deriving from $P(A)$ and $P(B)$) are convex in the ξ and μ parameterizations. And the elements of the product AB in the first term are sums of products of exponential functions applied to the ξ and μ, albeit with a renormalization in the ξ variable and an unbounded domain for both variables. So it seems possible that convex optimization procedures could be used to solve this problem, with appropriate care. However, the atomization implicit in the role the set $\mathcal{A}$ plays in the penalty on the ξ (and hence A) arising through the use of multinomial sampling of word-topic associations in the prior on A, requires thought. It might be possible to relax the problem to a more tractable variant. Alternatively, one might focus on the supervised version of the unsupervised posterior optimization we consider here, as (Airoldi and Bischof, 2015) do in their applications, replacing $\mathbb{P}(A, B|W)$ by $\mathbb{P}(A, B|W, I)$ throughout, which simplifies away this challenge. In any event, from the computational perspective, a strength of the probabilistic approach adopted by (Airoldi and Bischof, 2015) in formulating their regularization is readily apparent – the resulting optimization problem becomes primarily a problem of designing an appropriate Monte Carlo sampler.

3. There are several parameters in the HPC model that we have assumed here to be fixed and known. Our treatment of the document lengths l_d (important to the formulation of the problem in (Airoldi and Bischof, 2015), and a key way in which their work differs from much of that in the literature on topic models) is equivalent to conditioning on $\boldsymbol{l} \equiv \{l_d\}$, as it is done in (Airoldi and

Bischof, 2015) as well. On the other hand, our treatment of the variances $\tau^2_{f,j}$ is analogous to needing to set the regularization parameter(s) in a ridge regression. The probabilistic perspective adopted here facilitates an inferential approach to setting these parameters.

4. The manner in which the regularization introduced by (Airoldi and Bischof, 2015) is re-expressed in (3.4) is useful in helping to further highlight a central feature of their approach: the regularization is across topics over words (i.e., within columns of B, over rows) rather than the converse. It is this feature which appears to facilitate gains in interpretability.

3.5 Summary

In this work we provide a frequentist formulation complementing a Bayesian approach to topic allocation of text documents. In particular, we present a complexity penalized likelihood perspective and formulate the problem as a minimazion of an NMF-based least squares loss function plus penalty terms on the complexity of the matrices that make up the factorization. Our formulation highlights the novel approach of regularizing over words across topics which fosters interpretability of the inferred topics.

Chapter 4

Estimating Network Degree Distributions Under Sampling

4.1 Introduction, Setup and Notation

Networks, also referred to as graphs, are commonly used across many disciplines to represent the connections between elements in a system. When it comes to observing networks empirically, it is often the case that we do not have access to the whole population of vertices and edges. Consider large-scale online social media networks, which typically contain millions of vertices, and are stored on multiple memory locations. It might be too costly, too time consuming, or we may not be granted access due to privacy constraints to the entirety of the network, but only a fraction of it might be visible to us. Hence, instead of containing full information, the empirically observed network reveals only a partial view of the underlying phenomenon. Such an empirically observed network can be viewed as a sample from the underlying network. Since our fundamental goal as statisticians is to make inferences from the sample about characteristics of the underlying population, it is of interest to investigate how closely the characteristics estimated from the sampled network resemble the ones of the true network. This is not a trivial task, and there is no theory similar to the one for estimators based on independent and identically distributed samples. In the development of this line of estimation theory, we aim to characterize the impact of the dependence inherent to the elements of a network.

Let $G = (V, E)$ be a graph, with a set of vertices denoted by V of dimension n_V, and a set of edges, E, and let $G^* = (V^*, E^*)$ be the corresponding sampled graph, where V^* is a subset of V and E^* is a subset of E.

Network characteristics that are of interest include the degree distribution, density, network diameter, distribution of the clustering coefficient, distribution of the singular values of the adjacency matrix, etc. In the work below we focus on the network degree counts vector, which, up to re-scaling is equivalent to the network degree distribution, arguably one of the most fundamental features of a graph.

Figure 4·1 below illustrates the effect of sampling on the degree distribution. As we can see, the sampled degree distributions do not appear to be fair representatives of the true degree distribution. Furthermore, the observed degree distribution seems to vary greatly from sample to sample.

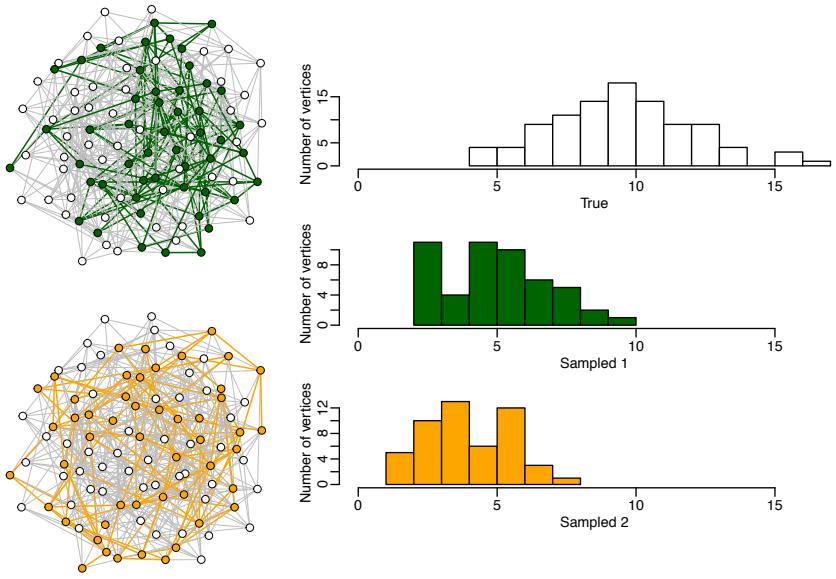


Figure 4·1: Left: Two induced subgraph samples (colored in green and yellow) generated from the same true graph (ER with 100 vertices and 500 edges) with the same sampling rate p = 50%. Right: Degree counts of the true graph and the two sampled graphs.

Recall that the degree of a vertex is defined as the number of edges coming out of it. Denote by $N = (N_0, \ldots, N_{m-1})$ the degree counts vector of the true network and let $N^* = (N_0^*, \ldots, N_{m-1}^*)$ be the degree counts vector of the observed (sampled) network, where $m-1$ is the maximum vertex degree in the true network.

As shown by (Frank, 1980), (Frank, 1981), under certain network sampling designs, the expectation of the observed degree counts vector is a linear combination of the true degree counts vector:

$$\begin{aligned} N^* &= PN + \varepsilon \\ \mathbb{E}\varepsilon &= 0 \\ \mathbb{E}\varepsilon\varepsilon^T &= C = Cov(N^*) \end{aligned} \tag{4.1}$$

where N is the true degree counts vector, N^* is the observed degree counts vector, P is a (possibly ill-conditioned) linear operator, ε is a noise vector. Introduced by the sampling, the noise vector ε, and hence N^*, are random variables. We are adopting a designed-based approach (as opposed to model-based), i.e. ε and P depend only on the sampling mechanism, and not on the structure of the network itself.

Our goal is to estimate N, the true degree counts vector. Since $\mathbb{E}N^* = PN$, then a natural choice of an unbiased estimator is $\widehat{N}_{\text{naive}} = P^{-1}N^*$. However, this naive estimator may not be computable, since the matrix P may be non-invertible. Even when it is invertible, we are not guaranteed non-negativity of the solutions (see Figure 4·2 below).

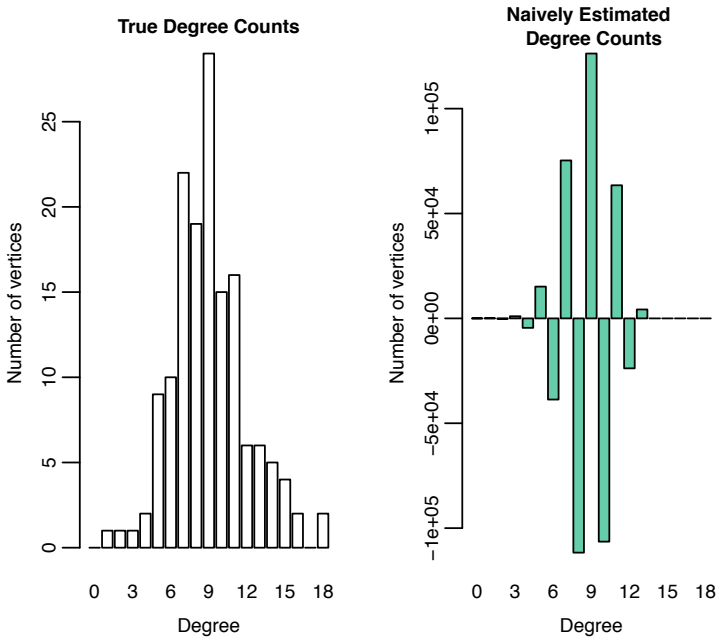


Figure 4·2: Left: Degree counts from an ER graph with 150 vertices and 680 edges. Right: Naive estimate of degree counts. Data drawn according to induced subgraph sampling with sampling rate p = 60%.

Since we desire non-negative degree counts, we consider solutions of the form

$$\widetilde{N} \in \mathcal{C} := \{\widetilde{N} : \widetilde{N} \geq 0 \text{ and } \mathbf{1}^T \widetilde{N} = n_V\}$$

The second part of the constraint comes from the fact that the sum of the number of nodes of various degrees is equal to the total number of nodes in the network.

4.2 Constrained Penalized Weighted Least-Squares Solution

(Zhang et al., 2015) propose the following estimator for the true degree counts vector:

$$\min_{\widetilde{N}} \ (P\widetilde{N} - N^*)^T C^{-1} (P\widetilde{N} - N^*) + t||D\widetilde{N}||_2^2 \tag{4.2}$$
$$\text{subject to } \ \widetilde{N} \in \mathcal{C}$$

where t is a regularization parameter, and D is a second-order differencing operator. The penalty term $||D\widetilde{N}||_2^2$ is introduced because P is often ill-conditioned.

Hence, the solution to the above problem will always be non-negative (by constraint), and existence and uniqueness of the the solution are guaranteed by the inclusion of the penalty term since it makes the objective function convex.

4.3 Theoretical Properties of the Unconstrained Estimator

Extensive numerical studies in (Zhang et al., 2015) indicate that this estimator performs well. We aim to develop theoretical guarantees of the quality of the constrained penalized weighted least-squares estimator. We succeed in the subcase of ego-centric design and achieve partial results and accompanying numerical results for other more complicated sampling designs. The theory we derive is limited to describing the behavior of the unconstrained estimator and we leave the details of the constrained case for future work.

4.3.1 Target Quantity

Our goal is to quantify how far off the solution $\widehat{N}$ is from the true degree counts vector N. Formally, the appropriate notion of distance between $\widehat{N}$ and N in our problem is $||P\widehat{N} - PN||_{C^{-1}}^2$, which takes into account the correlation between the entries of N^*. Note that $\widehat{N}$ is random, since it depends on the observed degree counts vector N^*, and hence, the target distance is also a random quantity. Therefore, we aim to bound $||P\widehat{N} - PN||_{C^{-1}}^2$ with high probability.

4.3.2 Complexity Functional

Following the formulation of the objective function (4.2) and focusing on the unconstrained solution, we find useful for our further derivations to define the following

complexity functional, inspired by (Donoho et al., 1997)

$$K(\widetilde{N}, \cdot) = (P\widetilde{N} - \cdot)^T C^{-1}(P\widetilde{N} - \cdot) + t||D\widetilde{N}||_2^2 \tag{4.3}$$

We aim to show that minimizing the complexity functional introduced above favors solutions $\widetilde{N}$ close to the true vector N in a weighted (by C^{-1}) l_2 norm squared spirit, appropriately regularized to make the problem solvable and encouraging $\widetilde{N}$ to be smooth.

Now consider the following key estimators:

- **Unconstrained minimum empirical complexity estimate:**

$$\widehat{N} = \text{argmin}\, K(\widetilde{N}, N^*) \tag{4.4}$$

  This unconstrained minimizer of the emipiral complexity is the solution we can find in practice.

  If we had an oracle who knows the true degree counts vector N, then we could use our complexity minimization rationale and produce the

- **Unconstrained minimizer of theoretical complexity:**

$$N^0 = \text{argmin}\, K(\widetilde{N}, PN) \tag{4.5}$$

4.3.3 Main Inequality

Since our goal is to quantify how far off the solution $\hat{N}$ is from the true degree counts vector N, we are interested in bounding $||P\widehat{N} - PN||_{C^{-1}}^2$ with high probability. To this end, we have derived the following inequality:

$$||P\widehat{N} - PN||_{C^{-1}}^2 \leq K(\hat{N}, PN) \leq K(N^0, PN) + 2 < \varepsilon, C^{-1}(P\hat{N} - PN^0) > \tag{4.6}$$

Proof follows as in (Donoho et al., 1997):

We begin by stating that for any $t > 0$ it holds:

$$||P\widehat{N} - PN||^2_{C^{-1}} \leq ||P\widehat{N} - PN||^2_{C^{-1}} + t||D\widehat{N}|| = K(\hat{N}, PN) \tag{4.7}$$

Now recall that $N^* = PN + \varepsilon$, hence we can replace PN with $N^* - \varepsilon$:

$$\begin{aligned} K(\hat{N}, PN) &= ||P\widehat{N} - PN||^2_{C^{-1}} + t||D\widehat{N}||^2_2 \\ &= (P\widehat{N} - (N^* - \varepsilon))^T C^{-1}(P\widehat{N} - (N^* - \varepsilon)) + t||D\widehat{N}||^2_2 \\ &= (P\widehat{N} - N^*)^T C^{-1}(P\widehat{N} - N^*) + t||D\widehat{N}||^2_2 \\ &+ 2 < \varepsilon, C^{-1}(P\widehat{N} - N^*) > + \varepsilon^T C^{-1}\varepsilon \end{aligned} \tag{4.8}$$

Note that we can combine the first two terms of (4.8) to get the empirical complexity functional of $\widehat{N}$. Therefore, the above equation becomes

$$= K(\widehat{N}, N^*) + 2 < \varepsilon, C^{-1}(P\widehat{N} - N^*) > + \varepsilon^T C^{-1}\varepsilon \tag{4.9}$$

In order to get an upper bound, we will use the fact that $K(\widehat{N}, N^*) \leq K(N^0, N^*)$, which is true because of equation (4.4), i.e $K(\widehat{N}, N^*) \leq K(\cdot, N^*)$. Hence, we can upper bound (4.9) by

$$K(N^0, N^*) + 2 < \varepsilon, C^{-1}(P\widehat{N} - N^*) > + \varepsilon^T C^{-1}\varepsilon \tag{4.10}$$

Now we will add and subtract PN^0 inside the second term of the inner product of (4.10), to complete a square. Thus, (4.10) now becomes

$$\begin{aligned} =&(PN^0 - N^*)^T C^{-1}(PN^0 - N^*) + t||DN^0||^2_2 \\ &+ 2 < \varepsilon, C^{-1}((P\widehat{N} - PN^0) + (PN^0 - N^*)) > + \varepsilon^T C^{-1}\varepsilon \end{aligned}$$

$$=(PN^0 - N^*)^T C^{-1}(PN^0 - N^*) + 2 < \varepsilon, C^{-1}(PN^0 - N^*) > + \varepsilon^T C^{-1}\varepsilon$$
$$+ t||DN^0||_2^2 + 2 < \varepsilon, C^{-1}(P\widehat{N} - PN^0) > \quad (4.11)$$

We notice that the first three terms of (4.11) are the expanded square $||PN^0 - N^* + \varepsilon||^2_{C^{-1}}$, which is equal to $||PN^0 - PN||^2_{C^{-1}}$. Hence, (4.11) is

$$= (PN^0 - PN)^T C^{-1}(PN^0 - PN) + t||DN^0||_2^2 + 2 < \varepsilon, C^{-1}(P\widehat{N} - PN^0) >$$
$$= K(N^0, PN) + 2 < \varepsilon, C^{-1}(P\widehat{N} - PN^0) >$$

This concludes the proof of the main inequality (4.6).

We can rewrite the second term in the right-hand side of (4.6) in order to emphasize its structure as a function of the noise vector ε. To do that, we use the closed form for the unconstrained solutions $\widehat{N}$ and N^0:

$$\widehat{N} = (P^T C^{-1} P + tD^T D)^{-1} P^T C^{-1} N^*$$
$$N^0 = (P^T C^{-1} P + tD^T D)^{-1} P^T C^{-1} PN$$

Plugging them in $P\widehat{N} - PN^0$ yields:

$$P\widehat{N} - PN^0 = P(P^T C^{-1} P + tD^T D)^{-1} P^T C^{-1} (N^* - PN)$$
$$= P(P^T C^{-1} P + tD^T D)^{-1} P^T C^{-1} \varepsilon$$

Hence, the inequality now becomes:

$$||P\widehat{N} - PN||^2_{C^{-1}} \leq K(N^0, PN) + 2 < \varepsilon, C^{-1} P(P^T C^{-1} P + tD^T D)^{-1} P^T C^{-1} \varepsilon >$$

Now, denoting

$$A := C^{-1} P(P^T C^{-1} P + tD^T D)^{-1} P^T C^{-1} \quad (4.12)$$

the main inequality (4.6) takes the form:

$$||P\widehat{N} - PN||^2_{C^{-1}} \leq K(N^0, PN) + 2\varepsilon^T A\varepsilon \tag{4.13}$$

We denote the first term, $K(N^0, PN) =: K^0$, and identify it as the **approximation error**. This quantity is the **ideal** value of the complexity functional, in the sense that it can be obtained only with an oracle who has full knowledge of the true vector of degree counts N and selects the best estimator by minimizing the theoretical complexity functional $K(\cdot, PN)$.

The second term in (4.13) is a **random** error component since it depends on the noise vector ε introduced by the sampling.

To explain why the main inequality (4.13) is useful in assessing the closeness of $\widehat{N}$ to N, let us revisit its more expanded form:

$$||P\widehat{N} - PN||^2_{C^{-1}} \leq K(\widehat{N}, PN) \leq K^0 + 2\varepsilon^T A\varepsilon$$

We hope that the estimator $\widehat{N}$ leads to a theoretical complexity, $K(\widehat{N}, PN)$, that is almost as good as the ideal theoretical complexity, K^0. Thus, we hope that the distribution of our target quantity, $||P\widehat{N} - PN||^2_{C^{-1}}$, takes on values close to the ideal theoretical quantity K^0. In other words, we would like to be able to show that $\mathbb{P}\left(||P\widehat{N} - PN||^2_{C^{-1}} - K^0 > \lambda\right)$ is small, where λ is some appropriate constant. Let's elaborate on this.

From the main inequality (4.13), we can conclude that

$$\mathbb{P}\left(||P\widehat{N} - PN||^2_{C^{-1}} - K^0 > \lambda\right) < \mathbb{P}\left(2\varepsilon^T A\varepsilon > \lambda\right)$$

Therefore, our aim is to bound $\mathbb{P}\left(\varepsilon^T A\varepsilon > \lambda\right)$.

4.3.4 Concentration Inequality

In this section we investigate the probability that the random error term $\varepsilon^T A\varepsilon$ takes on large values. A traditional approach towards this goal is to study how likely $\varepsilon^T A\varepsilon$ is to be larger than its mean value. Therefore, we focus on

$$\mathbb{P}\left(\varepsilon^T A\varepsilon - \mathbb{E}[\varepsilon^T A\varepsilon] > const\right)$$

One of the main difficulties of studying the behavior of $\varepsilon^T A\varepsilon$ comes from the fact that the entries of the ε vector are correlated. The hammer that helps us overcome this difficulty is a variant of Azuma's inequality, Lemma 4.1 on page 19 from (Borgs et al., 2008):

Lemma: Let k be a positive integer and let $c > 0$. Let $Z = (Z_1, \ldots, Z_k)$, where $Z_1, \ldots, Z_k$ are independent random variables. Let f be a measurable function. Suppose that $|f(x) - f(y)| \leq c$ whenever $x = (x_1, \ldots, x_k)$ and $y = (y_1, \ldots, y_k)$ differ only in one coordinate. Then

$$\mathbb{P}\left(f(Z) > \mathbb{E}[f(Z)] + Tc\right) < e^{-T^2/2k} \tag{4.14}$$

Indeed, we are able to use this lemma in our case, because we can consider $Z = (Z_1, \ldots, Z_k)$ to be the binary vector of independent coin flips at each vertex of the true network that determines the sampled graph. In all of the sampling designs considered in this work, there is always the notion of these underlying independent coin flips. Now let us translate the condition in the lemma that the vectors x and y only differ in one coordinate. Since our vectors represent sequences of coin flips, what we need to do is to consider one sequence in which, say the l-th coin flip resulted in a success (we will denote this sequence with 1_l), and a second sequence which is the same as the first one, except that it resulted in a failure in position l (denoted as 0_l). Naturally, we define the function $f(Z) := \varepsilon^T A\varepsilon =< \varepsilon, A\varepsilon >$. Then all that

we need to do is to show that $|f(1_l) - f(0_l)| \leq c$, $\forall l \in V$. In other words, we want to determine the value of a Lipschitz constant c that tells us how "smooth" $\varepsilon^T A\varepsilon$ is when we change the value of any one coin flip. The mechanism through which switching one coin flip influences $\varepsilon^T A\varepsilon$ under distinct sampling designs is different. Hence, the derivation of the concentration inequality will require a different approach under each sampling design. In 4.3.5 we carry out the details of its derivation and numerical studies under the three main sampling designs we consider.

4.3.5 Behavior of the Estimator Under Different Sampling Designs

We are able to explicitly derive an upper bound for the probability in equation (4.14) in the case of ego-centric sampling. Other sampling designs that obey our framework are, for example, induced and one-wave snowball sampling. The derivation of an exact analytical expression for these sampling designs is challenging, due to combinatorics, and we did not find approximations with our current approach. Thus, we make it our goal, to pursue the detailed derivation of the ego-centric case, additionally illustrate its behavior numerically, and use the insight gained to carefully design numerical studies to lend insight into possible approximate analytical expressions for the other sampling designs.

Ego-centric Sampling

Ego-centric sampling is a nonadaptive sampling design in which first a set of vertices is selected through a sequence of n_V i.i.d. Bernoulli(p) trials (i.e., one independent coin flip at each vertex) and then all edges incident to the selected vertices are observed. Therefore, a vertex in the sampled graph is observed to have degree k if and only if this vertex is selected and has degree k in the true graph. Hence,

$$N_k^* = \sum_{\{u:d_u=k\}} \mathbb{1}\{u \in V^*\}$$

where N_k^* denotes the number of vertices of degree k in the sampled graph, $k \in \{0, \ldots, m-1\}$. Recall that the max degree of the true network is equal to $m-1$.

$$N_k^* \sim \text{Bin}(N_k, p)$$

and the N_k^* are independent.

For ego-centric sampling, the entries of the matrix P are given by

$$P_{\text{ego}}(i,j) = \begin{cases} p & , i = j \\ 0 & , i \neq j \end{cases}$$

Thus, the $m \times m$ matrix P_{ego} is

$$P_{\text{ego}} = \begin{pmatrix} p & & & 0 \\ & p & & \\ & & \ddots & \\ 0 & & & p \end{pmatrix}$$

The $m \times m$ covariance matrix of N^* is

$$\begin{aligned} \text{Cov}(N^*) = C &= p(1-p)\text{diag}(N) \\ &= p(1-p) \begin{pmatrix} N_0 & & & & 0 \\ & N_1 & & & \\ & & N_2 & & \\ & & & \ddots & \\ 0 & & & & N_{m-1} \end{pmatrix} \end{aligned}$$

Hence, the inverse of the covariance C is given by

$$C^{-1} = \frac{1}{p(1-p)} \begin{pmatrix} \frac{1}{N_0} & & & & 0 \\ & \frac{1}{N_1} & & & \\ & & \frac{1}{N_2} & & \\ & & & \ddots & \\ 0 & & & & \frac{1}{N_{m-1}} \end{pmatrix}$$

The following assumptions are not necessary but are made to simplify the exposition:

- The maximum observed degree is equal to the true maximum degree.
- $N_k > 0, \forall k = 0, \ldots, m-1$ (otherwise consider the general inverse of the covariance matrix).

Now we are ready to tackle the details of the concentration inequality. Recall that, to apply the lemma, we need to show smoothness of $\varepsilon^T A \varepsilon$ and we need to determine the Lipschitz constant c. Let us assume that switching off the l-th coin flip causes a vertex of degree k to be turned off. We can express this as follows:

$$\begin{aligned}
\varepsilon(1_l) &= N^*(1_l) - PN \\
\varepsilon(0_l) &= N^*(0_l) - PN \\
&= N^*(1_l) - \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \leftarrow k^{\text{th}} \quad - PN \\
\Rightarrow \varepsilon(0_l) &= \varepsilon(1_l) - \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \leftarrow k^{\text{th}}
\end{aligned}$$

Now we can plug this in to get:

$$
\begin{aligned}
&|f(1_l) - f(0_l)| \\
&= |\varepsilon(1_l)^T A \varepsilon(1_l) - \varepsilon(0_l)^T A \varepsilon(0_l)| \\
&= \left| \varepsilon(1_l)^T A \varepsilon(1_l) - \left(\varepsilon(1_l) - \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \right)^T A \left(\varepsilon(1_l) - \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \right) \right| \\
&= \left| 2\varepsilon(1_l)^T A \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix}^T A \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \right| \\
&= |2\varepsilon^T A_{\bullet k} - A_{kk}|
\end{aligned}
$$

where $A_{\bullet k}$ denotes the k-th column of the matrix A. Note that the above holds since the matrix A defined by (4.12) is symmetric in the case of ego-centric sampling. However, bounding $|2\varepsilon^T A_{\bullet k} - A_{kk}|$, $\forall k$ is not an easy task, to a large degree due to the dependence between the entries of the noise vector ε. To make the derivation tractable, we propose a transformation that would "diagonalize" the problem (i.e. decouple it to a system of m linear equations). The proposed transformation is the following:

Theorem:

(i) There are nonsingular matrices M and L and nonnegative diagonal matrices F

and E such that

$$L^T L = C^{-1}, \tag{4.15}$$
$$LP = FM,$$
$$D^T D = M^T E^T E M$$

(ii) Whenever the system (4.15) holds, the transformed variables

$$x = MN$$
$$y^* = LN^*$$

satisfy

$$(PN - N^*)^T C^{-1} (PN - N^*) = ||Fx - y^*||_2^2$$
$$||DN||_2^2 = ||\mathbb{E}x||_2^2$$

We provide a fully constructive proof, closely following the proof of Theorem 11.1 in (Neumaier, 1998), that directly translates into an algorithm for computing M, L, F, and E.

Proof: We begin by factorizing $C = SS^T$ (i.e., the Cholesky decomposition of the covariance matrix) and consider, for some $\rho \neq 0$, the QR-factorization

$$\begin{bmatrix} S^{-1}P \\ \rho D \end{bmatrix} = QR \tag{4.16}$$

with $Q_{(2M \times M)} = \begin{bmatrix} Q_{1(M \times M)} \\ Q_{2(M \times M)} \end{bmatrix}$, $Q^T Q = I$ and upper triangular R.

Thus,

$$S^{-1}P = Q_1 R$$
$$\rho D = Q_2 R$$
$$Q_1^T Q_1 + Q_2^T Q_2 = I$$

Using the SVD

$$Q_1 = UFW^T \tag{4.17}$$

with orthogonal U,W and a non-negative diagonal matrix F, we define

$$M := W^T R$$
$$L := U^T S^{-1}$$

From (4.17), we find

$$\begin{aligned} F^T F &= (U^T Q_1 W)^T (U^T Q_1 W) \\ &= W^T Q_1^T Q_1 W \end{aligned}$$

so that the diagonal matrix

$$\begin{aligned} I - F^T F &= W^T W - W^T Q_1^T Q_1 W \\ &= W^T Q_1^T Q_2 W \\ &= (Q_2 W)^T (Q_2 W) \end{aligned}$$

is positive definite. Therefore its entries are non-negative, and we can form the non-negative diagonal matrix

$$E := \rho^{-1}(I - F^T F)^{1/2} \tag{4.18}$$

with component-wise square roots.

Now it holds that

- $L^T L = S^T U U^T S^{-1} = S^T S^{-1} = (SS^T)^{-1} = C^{-1}$
- $LP = U^T S^{-1} P = U^T Q_1 R = U^T U F W^T R = F W^T R = FM$
- $\rho D = Q_2 R = Q_2 W W^T R = Q_2 W M$
- $\rho^2 D^T D = (Q_2 W M)^T (Q_2 W M) = M^T (Q_2 W)^T (Q_2 W) M$
  $= M^T (I - F^T F) M = \rho^2 M^T E^T E M$ since E is a square diagonal matrix.

This concludes the proof of the theorem.

Notes (as provided by (Neumaier, 1998)):

1. When $C = I$, the factorization (4.15) is generally referred to as generalized singular value decomposition (GSVD).
2. An implementation may proceed according to (4.16), (4.17), and (4.18); in (4.16) one should choose $\rho = \dfrac{||S^{-1}P||_\infty}{||D||_\infty}$ or a similar expression to ensure that $S^{-1}P$ and D have similar magnitude.

In the new coordinate system we have

$$x := MN$$

$$y^* := LN^*$$

$$z := L\varepsilon$$

After the transformation, the penalty and the loss terms from the objective function

in (4.2) take the form

$$\begin{aligned} ||DN||_2^2 &= N^T D^T DN = N^T (M^T E^T EM) N \\ &= N^T (EM)^T (EM) N = ||EMN||_2^2 = ||Ex||_2^2 \\ (PN - N^*)^T C^{-1} (PN - N^*) &= (PN - N^*)^T L^T L (PN - N^*) \\ &= ||L(PN - N^*)||_2^2 = ||FMN - y^*||_2^2 = ||Fx - y^*||_2^2 \end{aligned}$$

The inverse problem can now be written as

$$y^* = Fx + z$$

The constrained penalized weighted least-squares estimator can be found from:

$$\begin{aligned} \min_{\widetilde{x}} & ||F\widetilde{x} - y^*||_2^2 + t||\mathbb{E}\widetilde{x}||_2^2 \\ \text{s.t.} \quad & M^{-1}\widetilde{x} \geq 0 \\ & \mathbf{1}^T M^{-1}\widetilde{x} = n_V \end{aligned}$$

It is straight-forward to verify that the new objective function and the new set of constraints $\mathcal{C}' := \{\widetilde{x} : M^{-1}\widetilde{x} \geq 0 \text{ and } \mathbf{1}^T M^{-1}\widetilde{x} = n_V\}$ are both convex.
Just like before, we narrow our attention to the unconstrained solution, for which we have the following closed-form expression:

$$\widehat{x} = (F^T F + tE^T E)^{-1} F^T y^*$$

We denote the complexity functional for the transformed problem as

$$K'(\tilde{x}, \cdot) = ||F\tilde{x} - \cdot||_2^2 + t||E\tilde{x}||_2^2$$

Hence, the main inequality (4.6) can be re-expressed as

$$\begin{aligned} ||P\widehat{N} - PN||^2_{C^{-1}} &= ||F\widehat{x} - Fx||^2_2 \\ &\leq K'(\widehat{x}, Fx) \\ &\leq K'(x^0, Fx) + 2 < z, F\widehat{x} - Fx^0 > \end{aligned}$$

Again, we aim to re-write the second term from the right-hand side of the inequality in order to emphasize its structure in terms of the "whitened" noise vector z. It is important to note that the components of z are uncorrelated, but are still dependent.

$$\begin{aligned} &\widehat{x} = (F^TF + tE^TE)^{-1}F^Ty^* \\ &x^0 = (F^TF + tE^TE)^{-1}F^TFx \\ \Rightarrow &F\widehat{x} - Fx^0 = F(F^TF + tE^TE)^{-1}F^Tz \\ \Rightarrow &< z, F\widehat{x} - Fx^0 >= z^TF(F^TF + tE^TE)^{-1}F^Tz \end{aligned}$$

Denote $W = F(F^TF + tE^TE)^{-1}F^T$ and note that W is a non-negative diagonal matrix. Therefore,

$$\begin{aligned} &< z, F\widehat{x} - Fx^0 >= z^TWz \\ &= \sum_{i=0}^{m-1} z_i^2 W_{ii} := \sum_{i=0}^{m-1} z_i^2 w_i \end{aligned}$$

where w is a vector containing the diagonal entries of the non-negative diagonal matrix W.

Recall that the main purpose of introducing the transformation was to make the derivation of the Lipschitz constant c tractable. The function f is given by z^TWz,

and we want to show that $|f(1_l) - f(0_l)| \leq c$.

$$\begin{aligned} |f(1_l) - f(0_l)| &= \left| \sum_{i=0}^{m-1} \left(z_i^2(1_l) - z_i^2(0_l) \right) w_i \right| \\ &= \left| < w, z^2(1_l) - z^2(0_l) > \right| \\ &\leq ||w||_2 ||z^2(1_l) - z^2(0_l)||_2 \end{aligned}$$

Since $||w||_2$ is a constant, we move to bounding $||z^2(1_l) - z^2(0_l)||_2$ by a constant.

$$\begin{aligned} z &= L\varepsilon = L(N^* - PN) \\ \Rightarrow z(1_l) &= L\left(N^*(1_l) - PN\right) \\ z(0_l) = L\left(N^*(0_l) - PN\right) &= L\left(N^*(1_l) - \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \leftarrow k^{\text{th}} \ - PN \right) \end{aligned}$$

because, according to our notation, switching off the l-th indicator turns off a vertex of degree k. Thus, we get

$$z(0_l) = z(1_l) - L \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \leftarrow k^{\text{th}} \ = z(1_l) - L_{\bullet k}$$

Let us look into the L matrix in more detail. Recall that $L^T L = C^{-1}$ where L is a

$m \times m$ matrix. Then,

$$L = \frac{1}{\sqrt{p(1-p)}} \begin{pmatrix} \frac{1}{\sqrt{N_0}} & & & & 0 \\ & \frac{1}{\sqrt{N_1}} & & & \\ & & \frac{1}{\sqrt{N_2}} & & \\ & & & \ddots & \\ 0 & & & & \frac{1}{\sqrt{N_{m-1}}} \end{pmatrix}$$

We are interested in the k-th column of L, i.e. $L_{\bullet k}$

$$L_{\bullet k} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \frac{1}{p(1-p)} \cdot \frac{1}{N_k} \\ 0 \\ \vdots \\ 0 \end{pmatrix} \leftarrow k^{\text{th}} \text{ entry}$$

Note that $L_{ik} = 0, \forall i \neq k$. Hence we get

$$z_i(0_l) = z_i(1_l) - L_{ik}$$

Plugging in this result yields:

$$\begin{aligned} &||z^2(1_l) - z^2(0_l)||_2 \\ &= \sqrt{\sum_{i=0}^{m-1} \left(z_i^2(1_l) - z_i^2(0_l)\right)^2} = \sqrt{\sum_{i=0}^{m-1} \left(z_i^2(1_l) - (z_i(1_l) - L_{ik})^2\right)^2} \\ &= \sqrt{\sum_{i=0}^{m-1} \left(z_i^2(1_l) - z_i^2(1_l) + 2z_i(1_l)L_{ik} - L_{ik}^2\right)^2} \\ &= \sqrt{\sum_{i=0}^{m-1} \left((2z_i(1_l) - L_{ik})L_{ik}\right)^2} \end{aligned}$$

Recall again that $L_{ik} = 0, \forall i \neq k$. Thus, only the k-th term in the summation remains:

$$
\begin{aligned}
&= \sqrt{\left((2z_k(1_l)L_{kk} - L_{kk}^2\right)^2} \\
&= \left|2z_k(1_l)L_{kk} - L_{kk}^2\right| \\
&= \left|2\frac{1}{\sqrt{p(1-p)N_k}}(N_k^* - pN_k)\frac{1}{\sqrt{p(1-p)N_k}} - \left(\frac{1}{\sqrt{p(1-p)N_k}}\right)^2\right| \\
&= \frac{2}{p(1-p)N_k}\left|N_k^* - pN_k - \frac{1}{2}\right|
\end{aligned}
$$

Since N_k^*, pN_k, and $\frac{1}{2}$ are all greater than 0, we get

$$\leq \frac{2N_k^* + 2pN_k + 1}{p(1-p)N_k}$$

Now, since the number of observed vertices of degree $k, \forall k \in \{0, \ldots, m-1\}$ is less or equal to the number of vertices of degree k in the true graph, i.e. $N_k^* \leq N_k$, we have

$$\leq \frac{2N_k + 2pN_k + 1}{p(1-p)N_k} = \frac{2(1+p)}{p(1-p)} + \frac{1}{p(1-p)N_k}$$

Therefore, for a fixed k, we have

$$|f(1_l) - f(0_l)| \leq ||w||_2 \left(\frac{2(1+p)}{p(1-p)} + \frac{1}{p(1-p)N_k}\right)$$

Putting everything together, we have an expression for the Lipschitz constant c:

$$\Rightarrow c = \max_k ||w||_2 \left(\frac{2(1+p)}{p(1-p)} + \frac{1}{p(1-p)N_k}\right) = \boxed{||w||_2 \left(\frac{3+2p}{p(1-p)}\right)} \tag{4.19}$$

Recall the main inequality (4.13):

$$||P\widehat{N} - PN||^2_{C^{-1}} \leq K^0 + 2\varepsilon^T A\varepsilon \tag{4.20}$$
$$= K^0 + 2z^T Wz$$

We have derived, in the ego-centric case, with c given by (4.19), the following bound

$$\mathbb{P}\left(z^T Wz > \mathbb{E}[z^T Wz] + Tc\right) < e^{-T^2/2n_V} \tag{4.21}$$

Hence, $\mathbb{P}\left(2z^T Wz > 2\mathbb{E}[z^T Wz] + T2c\right) < e^{-T^2/2n_V}$. Therefore, combining the main inequality (4.13) and the concentration inequality (4.21), we can conclude that

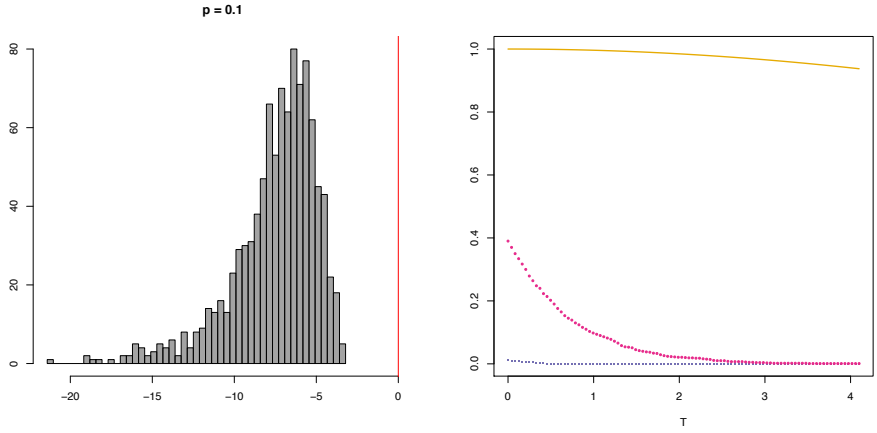
$$\mathbb{P}\left(||P\widehat{N} - PN||^2_{C^{-1}} - K^0 > \mathbb{E}[2\varepsilon^T A\varepsilon] + 2cT\right) \tag{4.22}$$
$$< \mathbb{P}\left(2\varepsilon^T A\varepsilon > \mathbb{E}[2\varepsilon^T A\varepsilon] + 2cT\right) < e^{-T^2/2n_V}$$

Our final step in the analysis of the ego-centric case is to visualize the main inequality (4.20) and the concentration result (4.22) for several sampling rates.

The setup of the simulations is the following - the true graph is Erdos-Renyi with 1000 vertices and 50000 edges. We explore the following sampling rates: 0.1, 0.2, 0.3, 0.5, 0.7, all under ego-centric sampling.

We construct the plots below by generating 1000 samples from the true network, and for each one we find the value of the target quantity and the random term. For each value of the sampling rate, we produce two plots. The first one demonstrates that the main inequality (4.20) holds. To produce this plot, we take the difference between the LHS and the RHS of (4.20) for each one of the 1000 samples, (i.e. $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$), and we plot its empirical distribution. The second plot verifies the concentration result. The color of the box around the expressions in

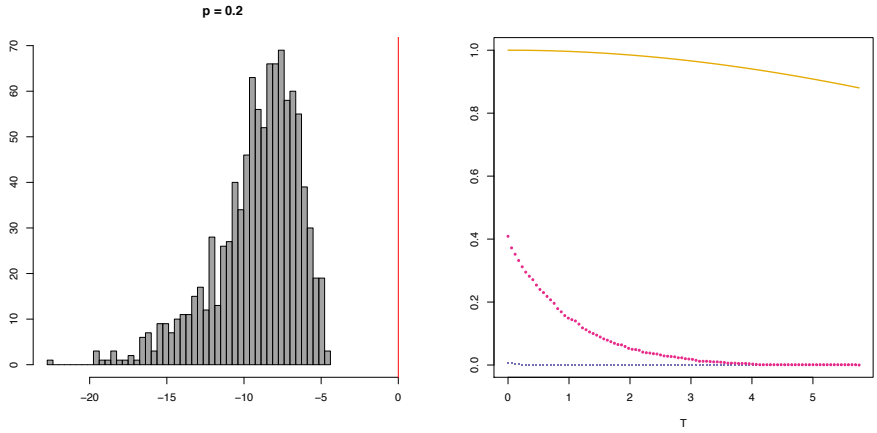
equation (4.22) is reflected in the plots below. Finally, in Figure 4·8, we address the question of the relative position of the ideal theoretical complexity with respect to the sampling distribution of the target quantity by visualizing $\mathbb{P}\left(||P\widehat{N} - PN||^2_{C^{-1}} > K^0\right)$ for the different values of the sampling rate.



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

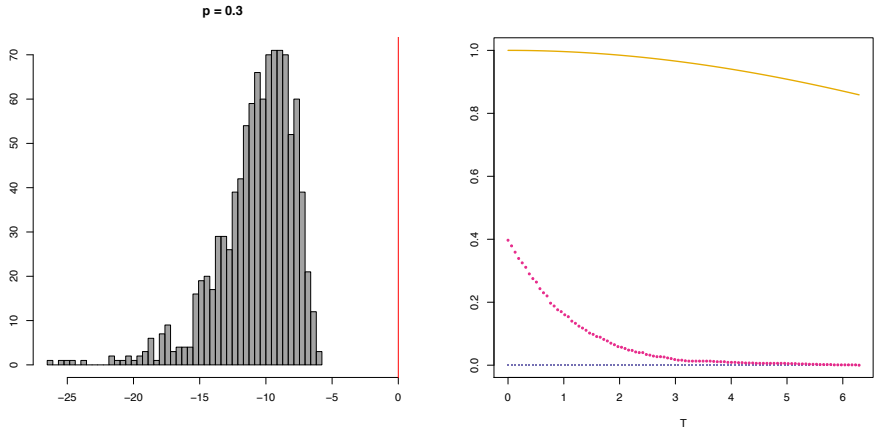
Figure 4·3: $p = 0.1$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

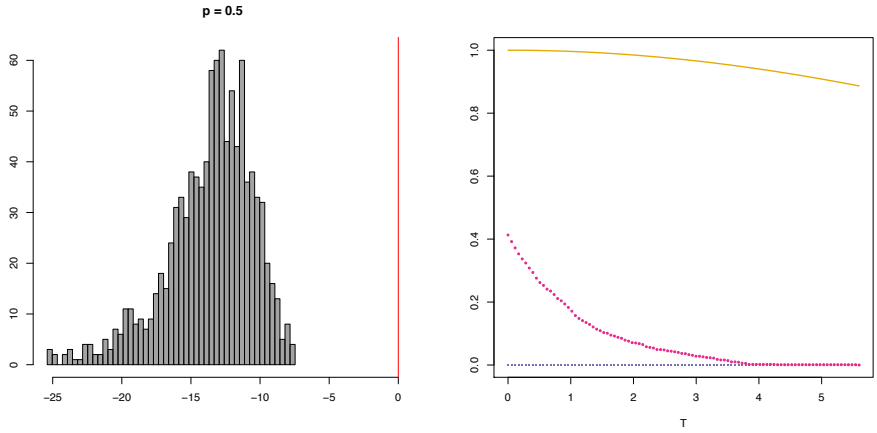
Figure 4·4: $p = 0.2$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 + 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

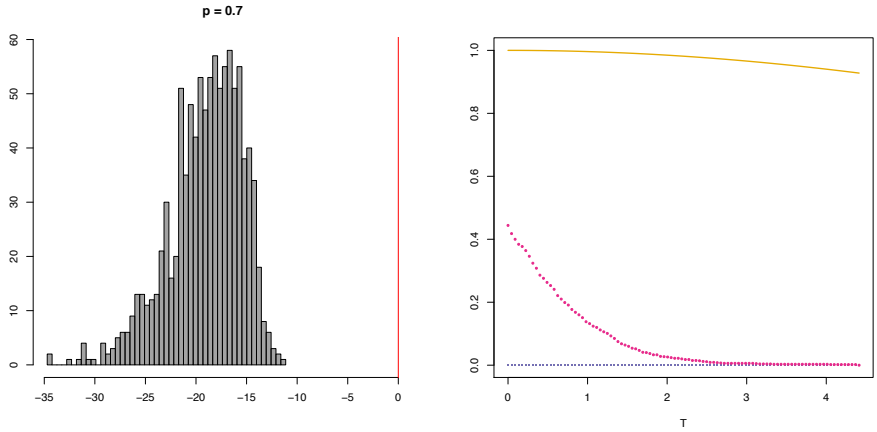
Figure 4·5: $p = 0.3$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

Figure 4·6: $p = 0.5$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

Figure 4·7: $p = 0.7$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)

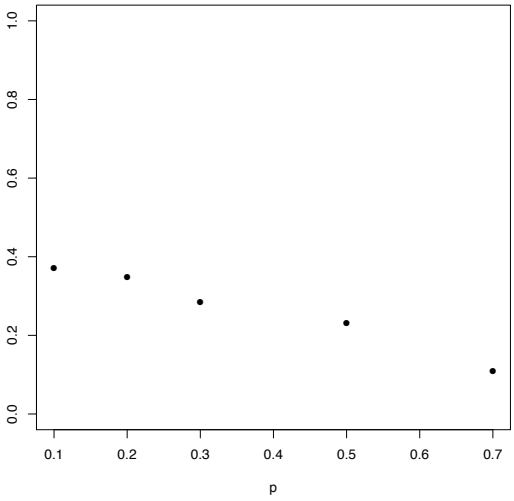


Figure 4·8: Probability that the target quantity is greater than the ideal ($||P\widehat{N} - PN||^2_{C^{-1}} > K^0$) for different values of the sampling rate

What we observe in all of the above plots is that both the main inequality and the concentration result hold. On all of the left-hand side plots we observe that the empirical distribution of the difference is to the left of 0, and the gap between $||P\widehat{N} - PN||^2_{C^{-1}}$ and $K^0 + 2\varepsilon^T A\varepsilon$ is increasing as the sampling rate p increases. On the right-hand side plots we observe that the upper bound (in yellow) of (4.22) is not tight, which is not surprising, since concentration inequalities generally do not produce tight bounds. As we vary the sampling rate from very low sampling (0.1) to high sampling (0.7) we observe that the probability that the target quantity, $||P\widehat{N} - PN||^2_{C^{-1}}$, is larger than the ideal value, K^0, decreases. This is an anticipated behavior, since higher sampling rate means we get to build an estimator based on a fuller view of the true network.

After describing the effect of varying the sampling rate, we are also interested in the performance of the estimator as we vary the size of the true network. We consider an Erdos-Renyi with 1000 vertices and 150000 edges. In Appendix A.3.1 we include the corresponding plots for this larger network and we observe the same trends.

To recap, we considered ego-centric sampling as the tractable case for closed form derivation of the concentration inequality. We proceed by investigating induced and one-wave snowball sampling. However, once we leave the world of i.i.d. degree sampling (ego-centric case), we were not able to carry out the derivation all the way through. We formulate the problem under the new design, and go through with the analogous derivations until we hit the combinatorial burden of the more complex sampling designs. To close the loop and draw final conclusions about the behavior of the estimator, we perform the same numerical study and provide visualizations.

Induced Subgraph

Induced subgraph sampling is a non-adaptive sampling scheme in which a set of vertices is selected through independent Bernoulli(p) trials at each vertex of the true graph and then all edges connecting selected vertices are observed. In other words, what we observe is the subgraph induced by the subset of sampled vertices. This sampling design is considered parsimoneous and has been used in the analysis of technological and biological networks (Stumpf and Wiuf, 2005).

The relation (4.1) holds for this sampling scheme. The form of the matrix P is given by

$$P_{\text{ind}}(i,j) = \begin{cases} \binom{j}{i} p^{i+1}(1-p)^{j-i} & , 0 \leq i \leq j \leq m-1 \\ 0 & , 0 \leq j < i \leq m-1 \end{cases}$$

For induced sampling, the matrix P is not diagonal. Figure 4·9 below illustrates the shape of the P matrix.

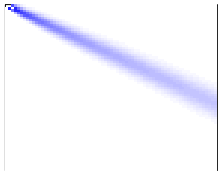


Figure 4·9: Heatmap of the values of the matrix P for induced subgraph sampling. The darker the color, the higher the value.
Sample generated from true graph (ER with 800 vertices and 20000 edges) with p = 60%.

A vertex in the sampled graph is observed to have degree k if and only if this vertex is selected, has degree k or higher in the true graph, and is connected to exactly k other sampled vertices. Hence,

$$N_k^* = \sum_{r=k}^{m-1} \sum_{u=1}^{n_V} \mathbb{1}\{u \in V^*, d_u^* = k, d_u = r\}$$

where $k \in \{0, \ldots, m-1\}$.

The N_k^*s are not independent for this design, hence the covariance matrix is not diagonal. Their mean is still given by $\mathbb{E}[N^*] = PN$. Expressions for their variance and covariance can be found in (Zhang et al., 2015).

Next, we turn our attention to the concentration inequality. Recall that, to apply the lemma, we need to show smoothness of $\varepsilon^T A\varepsilon$ and determine the Lipschitz constant c. Let us assume the l-th vertex is of degree k in the sampled network. Figure 4·10 below depicts the mechanism of induced subgraph sampling, and the effect of including (in the middle) and not including (on the right) the l-th vertex. The nodes highlighted in the network on the left represent the set of sampled vertices (including the l-th vertex). In the middle display, we color the induced subgraph (when the l-th vertex is switched on). In the right display, we show how the induced subgraph

changes when we switch off the l-th vertex.

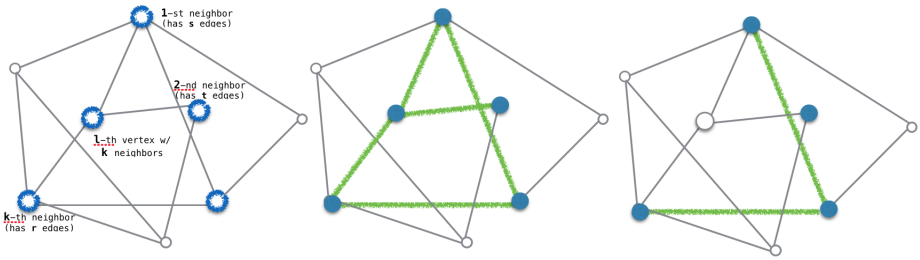


Figure 4·10: (Left) True Graph, (Middle) l-th vertex in the sample, (Right) l-th vertex not in the sample.

We express the effect of switching off the l-th vertex on ε below:

$$\varepsilon(1_l) = N^*(1_l) - PN$$

$$\varepsilon(0_l) = N^*(0_l) - PN$$

$$= N^*(1_l) + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \leftarrow k^{\text{th}} + \underbrace{\begin{pmatrix} 0 \\ \vdots \\ 1 \\ -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \\ \\ \leftarrow (s-1)^{\text{th}} \\ \leftarrow s^{\text{th}} \\ \\ \\ \\ \end{matrix} + \ldots + \begin{pmatrix} 0 \\ \vdots \\ 1 \\ -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \\ \\ \leftarrow (r-1)^{\text{th}} \\ \leftarrow r^{\text{th}} \\ \\ \\ \\ \end{matrix}}_{k \text{ such vectors}} - PN$$

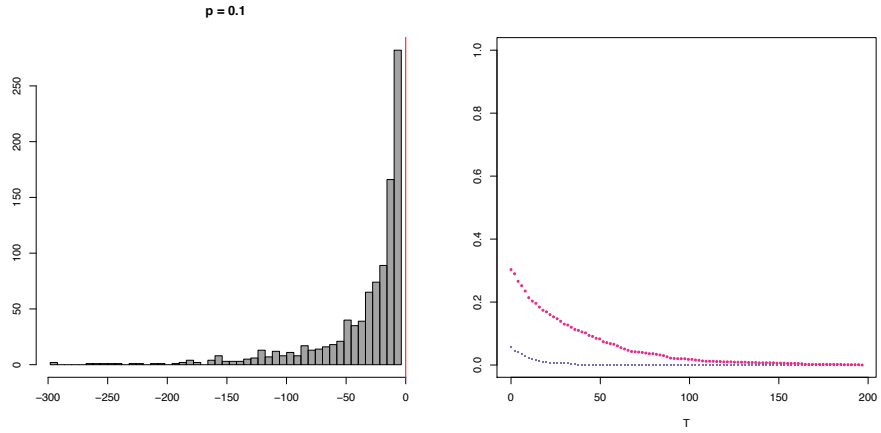
$$\Rightarrow \varepsilon(0_l) = \varepsilon(1_l) + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \leftarrow k^{\text{th}} + \begin{pmatrix} 0 \\ \vdots \\ 1 \\ -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \\ \\ \leftarrow (s-1)^{\text{th}} \\ \leftarrow s^{\text{th}} \\ \\ \\ \\ \end{matrix} + \ldots + \begin{pmatrix} 0 \\ \vdots \\ 1 \\ -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \\ \\ \leftarrow (r-1)^{\text{th}} \\ \leftarrow r^{\text{th}} \\ \\ \\ \\ \end{matrix}$$

The vector with a -1 at position k indicates that since we have switched vertex l, which is of degree k, then we must decrease the count of vertices of degree k by 1. The next k vectors are for each of the neighbors in the sampled graph of the l-th vertex. Let the first neighbor have s neighbors in the sampled graph. The perturbation causes

this vertex to have its degree now become $s-1$, since it is no longer connected to the vertex we have switched off, but it is still connected to all its other neighbors in the induced subgraph. Therefore, to adjust for this change in the degree counts vector, we need to decrease the number of vertices of degree s by 1, and increase the number of vertices of degree $s-1$ by 1. Analogously, we do the same for each of the other neighbors of the vertex we are perturbing.

Since we do not know the degree of the k neighbors of the l-th vertex, coming up with a closed form expression for the Lipschitz constant c would involve performing the computation over the set of all possible degrees that the k neighbors could have. The vast combinatorial burden of such a computation leads us to use numerical studies, like the ones that accompany the theory in the ego-centric case, in order to demonstrate the theoretical properties of the estimator in the case of induced subgraph sampling.

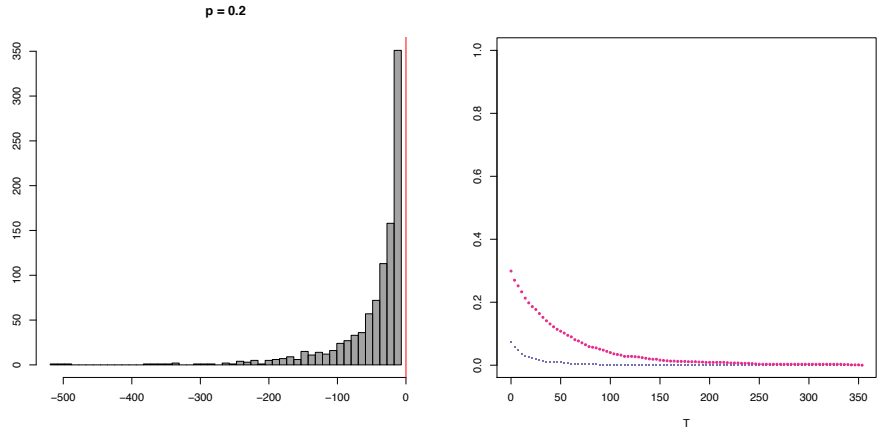
The setup of the simulations is the same as in the ego-centric case - the true graph is Erdos-Renyi with 1000 vertices and 50000 edges. We produce the same kind of plots as the ones in the ego-centric case. The only difference is that, since we were not able to derive a closed form for the Lipschitz constant c, we only plot the first and middle term of (4.22).



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

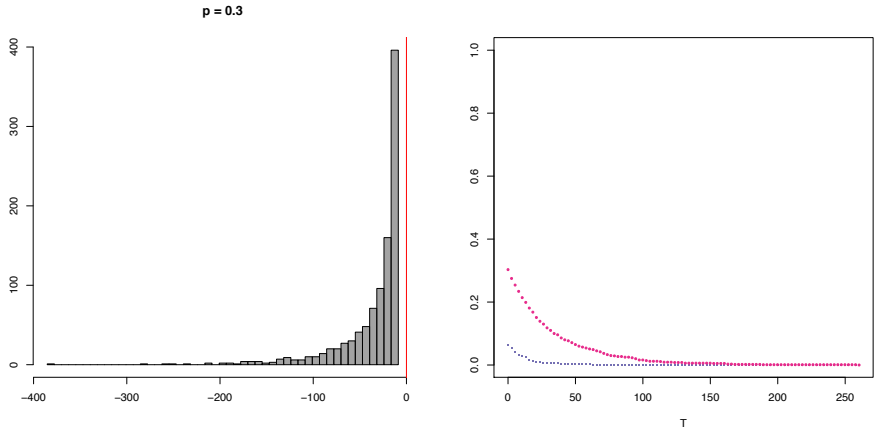
Figure 4·11: $p = 0.1$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

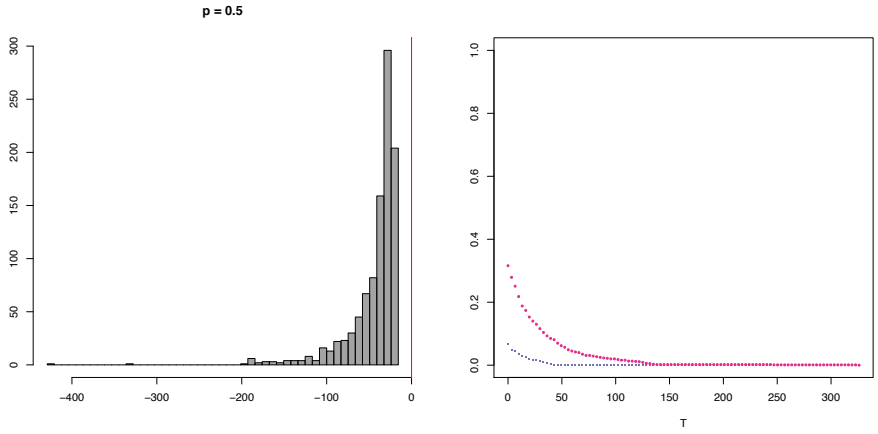
Figure 4·12: $p = 0.2$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 + 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

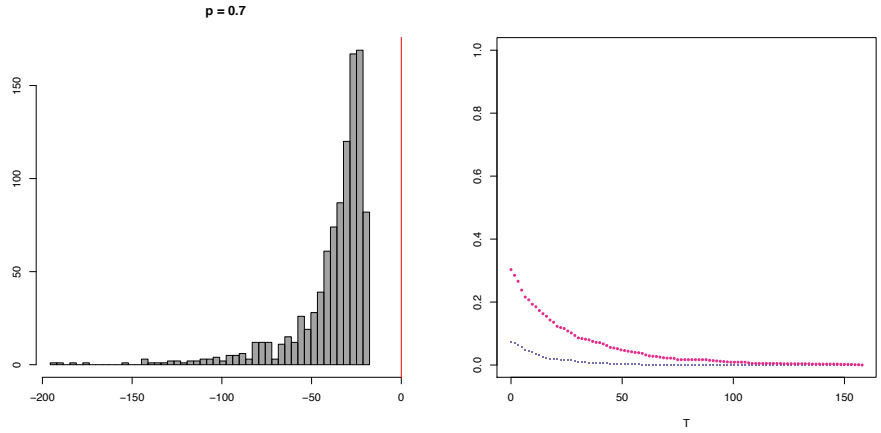
Figure 4·13: $p = 0.3$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

Figure 4·14: $p = 0.5$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

Figure 4·15: $p = 0.7$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)

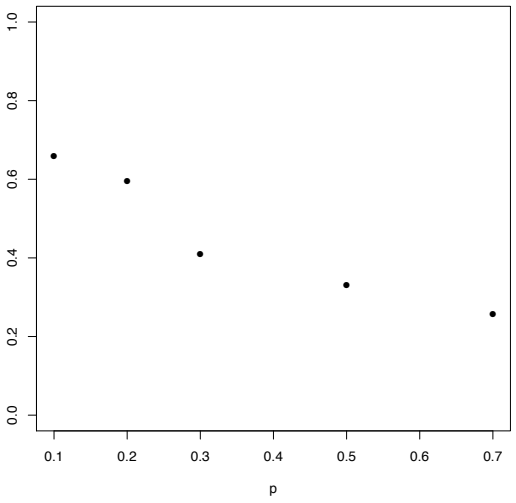


Figure 4·16: Probability that the target quantity is greater than the ideal ($||P\widehat{N} - PN||^2_{C^{-1}} > K^0$) for different values of the sampling rate

The estimator behaves similarly in the induced subgraph and the ego-centric cases, in the sense that the main inequality (4.20) and concentration (4.22) hold,

and $\mathbb{P}\left(||P\widehat{N} - PN||^2_{C^{-1}} > K^0\right)$ decreases as we increase p. However, there is at least one difference we should point out. The shape of the sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$ is now heterogeneous (skewed) as opposed to being homogeneous in the ego-centric case.

The behavior of the estimator remains similar as we increase the network size. See Appendix A.3.1 for the corresponding plots of the larger network. We notice that the probabilities that the target quantity is larger than the ideal theoretical complexity are bigger than the corresponding probabilities for the smaller network. This suggests that increasing the size of the network makes the problem of estimating the degree counts vector more difficult under induced subgraph sampling.

One-wave Snowball Sampling

In one-wave snowball sampling, there are two stages. In the initial stage, a set of vertices is drawn according to independent Bernoulli(p) trials. All edges coming out of the sampled vertices are observed. Then, in the second stage, we observe every vertex from the neighborhood of the vertices sampled at the first stage, and all edges incident to them. Therefore, one-wave snowball sampling is an adaptive sampling design. This sampling scheme has been used in social network studies (Rolls et al., 2012), and is similar to ego-centric sampling. The matrix P is diagonal, with entries given by:

$$P_{\text{snow}}(i,j) = \begin{cases} 1-(1-p)^{i+1} & , i = j \\ 0 & , i \neq j \end{cases}$$

A vertex in the sampled graph is observed to have degree k if and only if this vertex has degree k in the true graph, and is selected or is a neighbor of a selected vertex.

Hence,

$$N_k^* = \sum_{\{u:d_u=k\}} \mathbb{1}\{u \in V^*\}$$

where $k \in \{0, \ldots, m-1\}$ and the N_k^*s are not independent.

Again, we turn our attention to the concentration inequality, meaning that, to apply the lemma, we need to show smoothness of $\varepsilon^T A\varepsilon$ and determine the Lipschitz constant c. Let us again assume the l-th vertex is of degree k in the sampled network. Figures 4·17, 4·18, 4·19 focus on a piece of the network and depict the mechanism of one-wave snowball sampling, along with the effect of including (in the middle) and not including (on the right) the l-th vertex. The nodes highlighted in the network on the left in all three figures represent the set of sampled vertices (including the l-th vertex) within the piece of the network that we are considering. In the middle display, we color the resulting subgraph (when the l-th vertex is switched on) under the one-wave snowball sampling design. In the right display, we show how the sampled subgraph changes when we switch off the l-th vertex. We lay out three base cases where we consider which, if any, of the near-by vertices to node l are also included in the sample. Generally, in addition to the base cases, a combination of case 2 and case 3 also may occur, along with variations on the number of neighboring vertices that are also in the sample.

Case 1: None of the neighbors or neighbors of neighbors of the l-th vertex are also in the sample

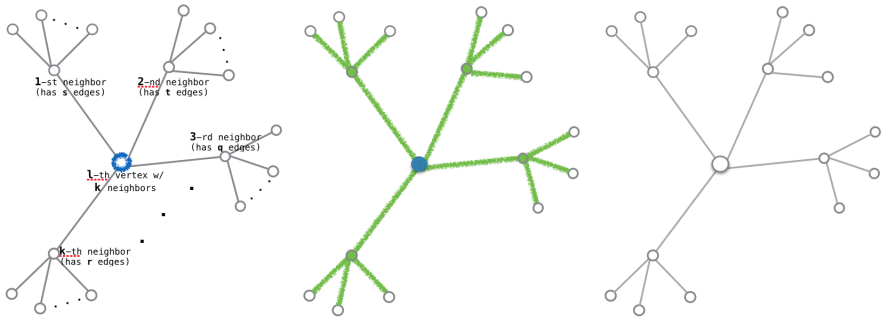


Figure 4·17: Case 1. (Left) Piece of the True Graph, (Middle) l-th vertex included in the sample, (Right) l-th vertex not in the sample.

Case 2: One of the neighbors of the l-th vertex is also in the sample

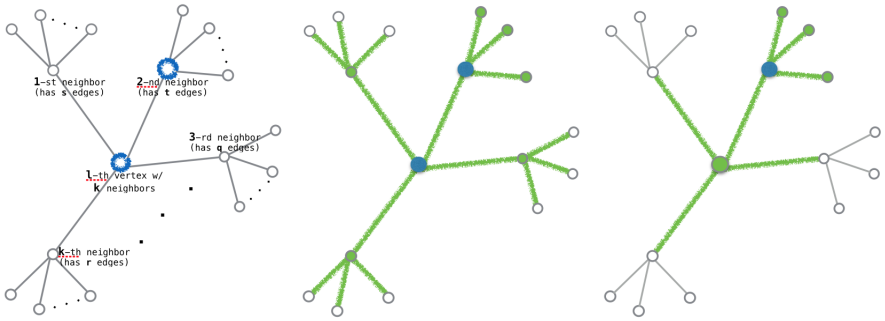


Figure 4·18: Case 2. (Left) Piece of the True Graph, (Middle) l-th vertex included in the sample, (Right) l-th vertex not in the sample.

Case 3: One of neighbors of neighbors of the l-th vertex is also in the sample

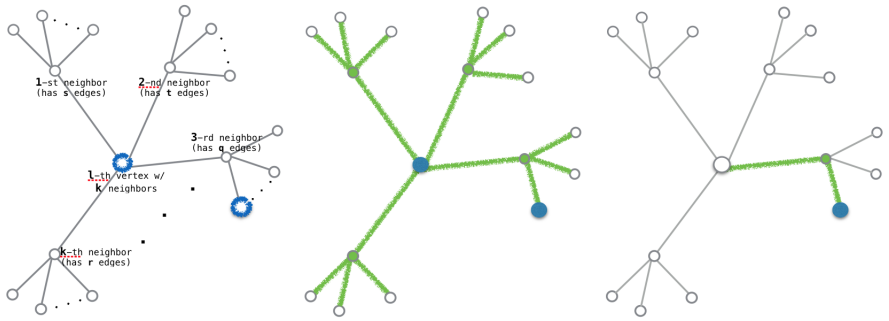


Figure 4·19: Case 3. (Left) Piece of the True Graph, (Middle) l-th vertex included in the sample, (Right) l-th vertex not in the sample.

We express the effect of switching off the l-th vertex on ε below:

$$\begin{aligned} \varepsilon(1_l) &= N^*(1_l) - PN \\ \varepsilon(0_l) &= N^*(0_l) - PN \\ &= N^*(1_l) - u_l - PN \\ \Rightarrow \varepsilon(0_l) &= \varepsilon(1_l) - u_l \end{aligned}$$

where u_l is a vector, such that

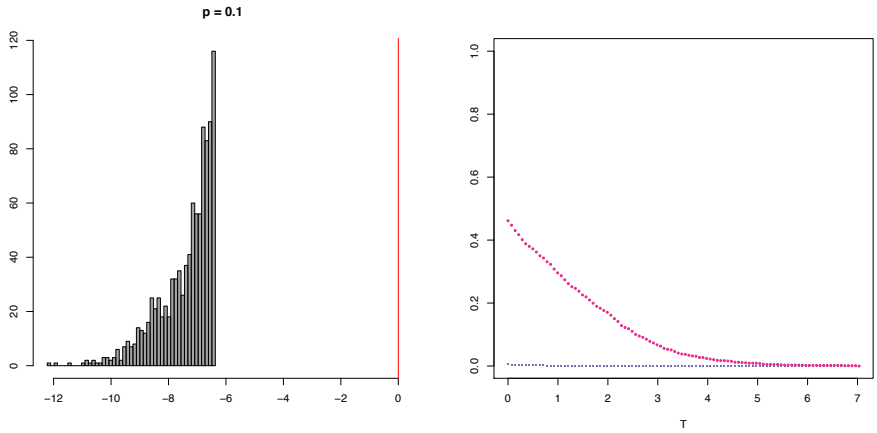
$$u_l : \left[\begin{array}{l} u_l^T \mathbf{1} \leq k+1 \\ u_l \in \mathbb{N}^m \end{array} \right. \tag{4.23}$$

The positive integer valued m-dimensional vector u_l has the form (4.23) because, as we can see from the base cases in Figures 4·17, 4·18, 4·19, switching off the l-th vertex (of degree k) could end up causing up to $k+1$ vertices to be excluded from the sampled subgraph. Therefore, we may have to decrease by one up to $k+1$ (possibly overlapping) entries of the degree vector, corresponding to the degrees of the excluded vertices.

As in the induced subgraph case, here again we do not know the degree of all k neighbors of the l-th vertex, or how many neighbors or neighbors of neighbors are simultaneously included in the sample. Hence, making it unfeasible to come up with a closed form for the Lipschitz constant c. Therefore, we resort to numerical studies to demonstrate the theoretical properties of the estimator in the case of one-wave snowball sampling.

We use a modification of the previous sampling set up. The true graph is again Erdos-Renyi with 1000 vertices, but now with 500 edges. If we were to have a graph as dense as before, we would end up observing all edges, even with low rates of sampling, due to the propagating nature of one-wave snowball sampling. Additionally, the sampling rates of 0.1, 0.2, 0.3, 05, and 0.7 now refer to the percentage of the total number of vertices sampled as a result of the two stages of this sampling design.

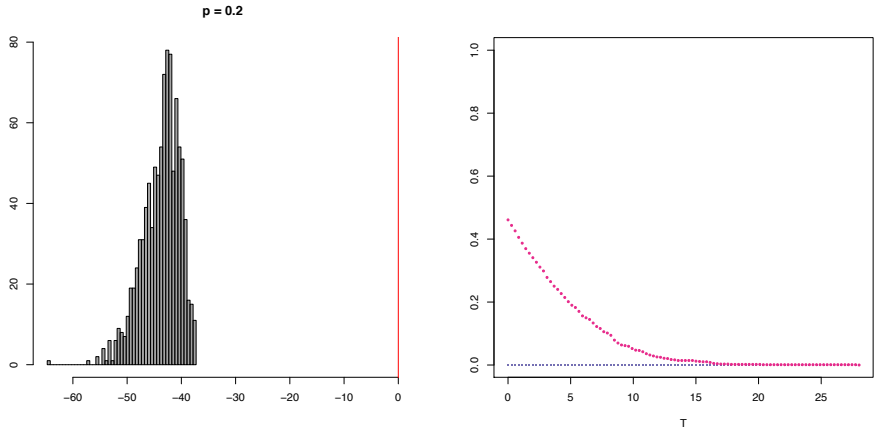
The plots below are constructed in the same way as the plots in the ego-centric and induced cases.



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

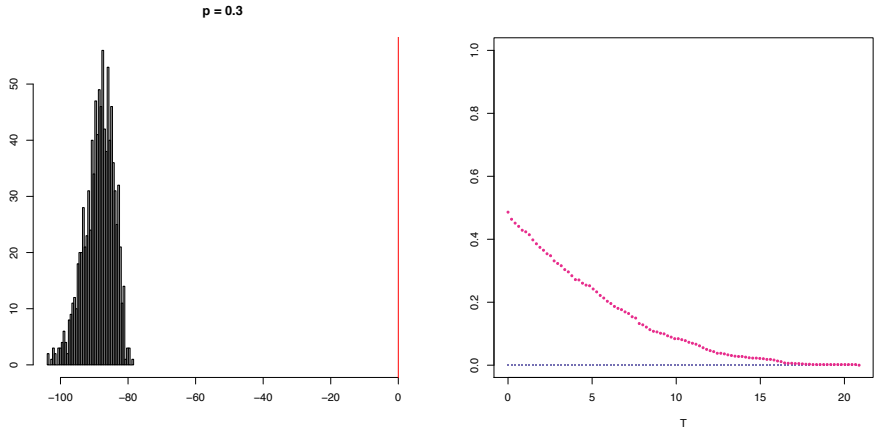
Figure 4·20: $p = 0.1$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

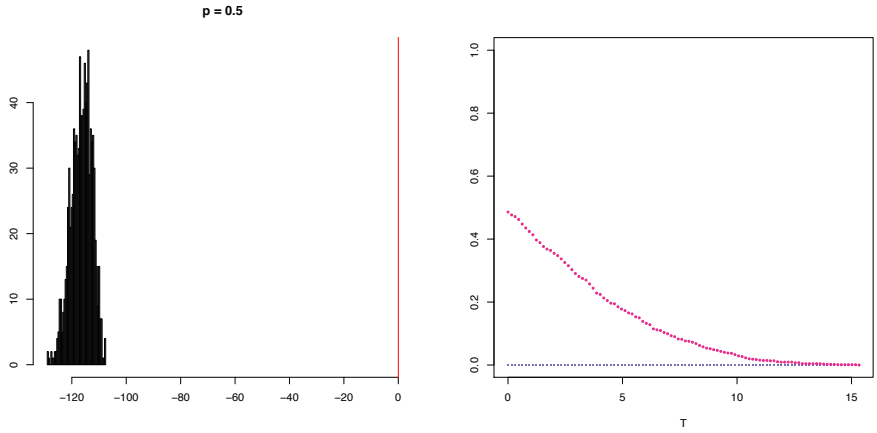
Figure 4·21: $p = 0.2$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 + 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

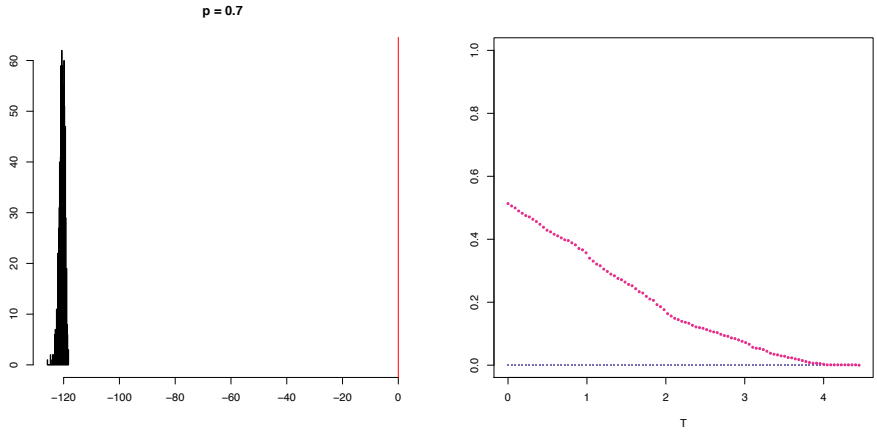
Figure 4·22: $p = 0.3$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

Figure 4·23: $p = 0.5$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

Figure 4·24: $p = 0.7$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)

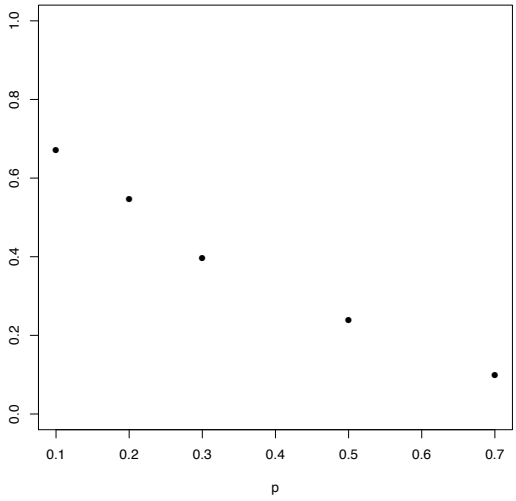


Figure 4·25: Probability that the target quantity is greater than the ideal ($||P\widehat{N} - PN||^2_{C^{-1}} > K^0$) for different values of the sampling rate

The performance of the estimator in the one-wave snowball case is overall similar to the ego-centric and induced cases. The shape of the sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$ is rather homogeneous (more similar to the ego-centric case). As we increase p the spread of the sampling distribution visibly decreases.

4.4 Summary

In this work we consider the problem of estimating the degree counts vector of a network in the context of only having the availability of a sample from the true network. We study the theoretical properties of the constrained penalized weighted least-squares estimator proposed by (Zhang et al., 2015). We identify a suitable metric that quantifies the discrepancy between the true solution and the estimator, and refer to it as our target quantity. We bound the probability that the target quantity exceeds the ideal theoretical complexity of the estimator in the ego-centric case. We bring the concentration inequality technique that we lay out for the ego-centric case

to bare in the cases of induced subgraph and one-wave snowball sampling designs. The derivation of the concentration result in these considerably more complex sampling schemes is hindered by their combinatorial complexity. Therefore, we utilize numerical studies and visualization aids to demonstrate that the estimator behaves similarly across the different sampling designs.

Chapter 5

Conclusions

This work is concerned with designing and studying the theoretical behavior of estimators based on complexity penalized methods for both structured and unstructured data. It opens any number of possibilities for future work. Some concrete routes are outlined below for each of the three projects.

In the inverse model calibration problem that we consider first, there are several directions one can pursue in order to enhance the applicability of our methodology. The first is to improve the geometric properties of the reconstructed input model, for instance the smoothness. The second is methodological development in the case of multiple input and multiple output variables. So far we have only focused on the one-to-one situation, but in practice the simulation models are likely more complex. The third is further investigation on the statistical guarantees and robustness of this line of methods, including also situations where not only the input model is unknown but the system logic in the simulation model could also be subject to errors.

In the topic allocation of text documents problem we take a Bayesian framework and provide a corresponding frequentist formulation. We derive an objective function that consists of a NMF-based least-squares objective and two penalty terms. One direction for further investigation of our formulation is to characterize the nature of the objective function and propose a feasible algorithm for its optimization. A common heuristic for optimizing the NMF-based loss function is alternating minimization. As far as the penalty terms are concerned, it is easy to check that the penalty term in

A is convex. It would require nontrivial efforts to check if the penalty term in B is also convex. If this penalty term is not convex, then a common approach is convex relaxation, i.e. replacing it by a suitably derived convex term. A consequent step after proposing an optimization algorithm would be to study the theoretical properties of the resulting estimator.

In the third project we aim to theoretically characterize the distance between the true degree counts vector of a network and the penalized weighted least-squares estimator proposed by (Zhang et al., 2015). In the subcase of ego-centric sampling we derive a concentration inequality that bounds the tail of the probability that the distance between the true and the estimated degree counts (referred to as the target distance) is larger than the ideal theoretical complexity functional. We visualize the theory via numerical studies that illustrate the relative position of the ideal to the sampling distribution of distance between the truth and the estimator. We confirm that the target distance concentrates around the ideal. For the two other more complicated sampling designs we achieve partial theoretical results. We perform analogous numerical studies and conclude that, although there are some differences across designs, especially in the shape of the empirical distribution of the target quantity, it is always concentrating around the ideal at a similar rate. Based on this observation we are hopeful that our work could be extended to show that the difference between the target distance and the ideal under induced and one-wave snowball sampling can be approximated reasonably well by their difference in the ego-centric case, the theory for which we have derived fully. Another component of the analysis of the quality of the estimator would be to study the ideal theoretical complexity and verify that its magnitude is small in an appropriate sense. Finally, recall that we carry out the theoretical derivations considering the unconstrained estimator. A logical continuation of our work would be to understand how adding the constraints influences the theory

derived so far. In the current work we examine three sampling designs - one simple case (ego-centric), and two more complex designs - one non-adaptive (induced) and one adaptive (one-wave snowball). It could be of interest to consider more sampling designs, for example a random walk on the graph.

Appendix A

Additional Proofs for Chapter 2

A.1 Auxiliary Theorems

Theorem A.1.1 (Corollary in Section 3 in (Blum, 1954)). *Let Y_k be a sequence of integrable random variables that satisfy*

$$\sum_{k=1}^{\infty} E[E[Y_{k+1} - X_k | X_1, \ldots, X_k]^+] < \infty$$

where $x^+ = x$ if $x > 0$ and 0 otherwise, and are bounded below uniformly in k. Then Y_k converges a.s. to a random variable.

Lemma A.1.1 (Lemma 2.1 in (Nemirovski et al., 2009)). *Let ω be defined in* (2.11) *and V in* (2.10). *Denote $\mathcal{X}$ as the feasible region of* (2.5). *For every $\mathbf{q} \in \mathcal{X}$, $\mathbf{p} \in \mathcal{X}^\circ$, and $\boldsymbol{\xi} \in \mathbb{R}^n$, one has*

$$V(\tilde{\mathbf{p}}, \mathbf{q}) \leq V(\mathbf{p}, \mathbf{q}) + \boldsymbol{\xi}'(\mathbf{q} - \mathbf{p}) + \frac{\|\boldsymbol{\xi}\|_*^2}{2\alpha}$$

where $\tilde{\mathbf{p}} = \min_{\mathbf{u} \in \mathcal{X}} \boldsymbol{\xi}'(\mathbf{u} - \mathbf{p}) + V(\mathbf{p}, \mathbf{u})$ is the prox-mapping acting on $\mathbf{p}$, and $\|\cdot\|_*$ is the dual norm of $\|\cdot\|$, α is the strong convexity parameter, both defined in (2.11).

A.2 Supplementary Materials

A.2.1 Quadratic Penalty Method

An application of the conventional quadratic penalty method (Bertsekas, 1999) yields the following:

Lemma A.2.1. *Suppose that* (2.3) *is feasible. Consider the sequence of optimization*

programs

$$\begin{array}{ll} \min & \sum_{j=1}^{m}(E_{\mathbf{p}}[\phi_j(h(\mathbf{X}))] - \mu_j)^2 - \lambda R(\mathbf{p}) \\ \textit{subject to} & \mathbf{p} \in \mathcal{P} \end{array} \tag{A.1}$$

for $\lambda > 0$. Let $\mathbf{p}^*(\lambda)$ be an optimal solution for (A.1) *indexed at λ. As λ decreases to 0, every limit point of the sequence $\{\mathbf{p}^*(\lambda)\}$ is an optimal solution for* (2.3)*.*

Proof. Proof of Lemma A.2.1 Consider relaxing the constraints in (2.3) to get

$$\begin{array}{ll} \min & -R(\mathbf{p}) + c\sum_{j=1}^{m}(E_{\mathbf{p}}[\phi_j(h(\mathbf{X}))] - \mu_j)^2 \\ \text{subject to} & \mathbf{p} \in \mathcal{P} \end{array} \tag{A.2}$$

for $c > 0$, which is equivalent to (A.1) with $\lambda = 1/c$. Proposition 4.2.1 in (Bertsekas, 1999) entails that as $c \to \infty$, every limit point of the sequence of optimal solutions for (A.2) converges to the optimal solution of (2.3), given that (2.3) is feasible. This concludes the lemma. □ □

A Variant of MDSA

In parallel to Section 2.4.2, we shall design an iterative procedure for solving (A.1). Note that the objective function in (A.1) consists of a non-convex, stochastic component $\sum_{j=1}^{m}(E_{\mathbf{p}}[\phi_j(h(\mathbf{X}))] - \mu_j)^2$ and a convex component $-\lambda R(\mathbf{p})$. We shall use the idea of proximal gradient (Sra et al., 2012) used for solving composite objective functions in convex problems, which iteratively linearizes the first component while keeping the second component intact at every iteration. The variant of MDSA scheme under this operation amounts to solving, given a current solution $\mathbf{p}^k$,

$$\begin{array}{ll} \min & \gamma^k \hat{\boldsymbol{\psi}}^{k'}(\mathbf{p} - \mathbf{p}^k) - \gamma^k \lambda R(\mathbf{p}) + V(\mathbf{p}^k, \mathbf{p}) \\ \text{subject to} & \mathbf{p} \in \mathcal{P} \end{array} \tag{A.3}$$

The gradient estimate $\hat{\boldsymbol{\psi}}^k$ is the same as discussed in Section 2.4.2, γ^k is the step size, and V is the KL divergence discussed in Section 2.4.2. Consider the generic formulation of (A.3) written as

$$\begin{array}{ll} \min & \boldsymbol{\xi}'(\mathbf{q} - \mathbf{p}) - \beta R(\mathbf{q}) + V(\mathbf{p}, \mathbf{q}) \\ \text{subject to} & \mathbf{q} \in \mathcal{P} \end{array} \tag{A.4}$$

Lemma A.2.2. *An optimal solution for* (A.4) *is given by $\mathbf{q}^* = (q_1^*, \ldots, q_n^*)$ where*

$$q_i^* = \frac{p_i^{\frac{1}{1+\beta}} e^{-\frac{\xi_i}{1+\beta}}}{\sum_{l=1}^{n} p_l^{\frac{1}{1+\beta}} e^{-\frac{\xi_l}{1+\beta}}} \tag{A.5}$$

Proof. Proof of Lemma A.2.2 Consider the Lagrangian for (A.4)

$$\begin{aligned} & \boldsymbol{\xi}'(\mathbf{q}-\mathbf{p}) - \beta R(\mathbf{q}) + V(\mathbf{p},\mathbf{q}) + \alpha\left(\sum_{i=1}^{n} q_i - 1\right) \\ = \;& \boldsymbol{\xi}'(\mathbf{q}-\mathbf{p}) + \beta\sum_{i=1}^{n} q_i \log q_i + \sum_{i=1}^{n} q_i \log\frac{q_i}{p_i} + \alpha\left(\sum_{i=1}^{n} q_i - 1\right) \end{aligned} \tag{A.6}$$

by relaxing the constraint $\sum_{i=1}^{n} q_i = 1$. Differentiating (A.6) with respect to $\mathbf{q}$ gives

$$\xi_i + \beta\log q_i + \beta + \log\frac{q_i}{p_i} + 1 + \alpha$$

Setting to zero gives

$$q_i \propto p_i^{\frac{1}{1+\beta}} e^{-\frac{\xi_i}{1+\beta}}$$

Using the constraint $\sum_{i=1}^{n} q_i = 1$, we get (A.5), which can be verified to satisfy the KKT condition straightforwardly. □ □

One advantage of using the representation (A.1) and the stepwise subprogram (A.3), as compared to (2.4) and (2.5) introduced in Section 2.4, is that it does not involve any root-finding in the MDSA iterations. Thus the resulting procedure is faster than that in Section 2.4. However, examining when to stop the algorithm becomes less clear as the procedure now relies on the convergence over the sequence of λ to 0, rather than a cutoff at η^* as in Section 2.4. In the experimental settings in Section 2.5 we found that it is difficult to determine when to stop in using (A.1). Since this issue outweighs the marginal advantage in removing the need of solving for a one-dimensional root, we have chosen to adopt (2.4). Nonetheless, the next section describes that the analysis of the MDSA procedure under this alternate approach in parallel to that under (2.4).

Convergence Analysis of the Variant of MDSA

The variant of MDSA for solving (A.1) is depicted in Algorithm 2. Similar to Algorithm 1, Steps 2 and 3 in Algorithm 1 combine to solve (A.3) with $\mathcal{P}$ replaced by $\mathcal{P}(\epsilon)$. The rationale for such operations lies in a technicality in guaranteeing boundedness of the gradient estimator, and subsequently algorithmic convergence, as in Algorithm 1.

Algorithm 2 MDSA for solving (2.4)

Input: A small parameter $\epsilon > 0$, initial solution $\mathbf{p}^1 \in \mathcal{P}(\epsilon) = \{\mathbf{p} : \sum_{i=1}^n p_i = 1, p_i \geq \epsilon \text{ for } i = 1, \ldots, n\}$, a step size sequence γ^k, and sample sizes M_1 and M_2.

Iteration: For $k = 1, 2, \ldots$, do the following: Given $\mathbf{p}^k$,

1. Estimate $\hat{\boldsymbol{\psi}}^k = (\hat{\psi}_1^k, \ldots, \hat{\psi}_n^k)$ with

$$\hat{\psi}_i^k = 2 \sum_{j=1}^m \frac{1}{M_1} \sum_{r=1}^{M_1} (\phi_j(h(\mathbf{X}^{(r)})) - \mu_j) \frac{1}{M_2} \sum_{r=1}^{M_2} \phi_j(h(\tilde{\mathbf{X}}^{(r)})) S_i(\tilde{\mathbf{X}}^{(r)}; \mathbf{p}^k)$$

where $\mathbf{X}^{(r)}$ and $\tilde{\mathbf{X}}^{(r)}$ are M_1 and M_2 independent copies of the input process generated under i.i.d. replications of $\mathbf{p}^k$, which are used simultaneously for all components of $\hat{\boldsymbol{\psi}}^k$.

2. Output

$$p_i^{k+1} = \frac{p_i^k \frac{1}{1+\gamma^k \lambda} e^{-\frac{\gamma^k \hat{\psi}_i^k}{1+\gamma^k \lambda}}}{\sum_{l=1}^n p_l^k \frac{1}{1+\gamma^k \lambda} e^{-\frac{\gamma^k \hat{\psi}_l^k}{1+\gamma^k \lambda}}}$$

3. If $p_i^{k+1} < \epsilon$ for some i, then solve the convex optimization (A.3) but with $\mathcal{P}$ replaced by the set $\mathcal{P}(\epsilon)$. Output its solution as $\mathbf{p}^{k+1}$.

To prove almost sure convergence of Algorithm 2, we need the following generalization of Lemma 2.1 in (Nemirovski et al., 2009):

Lemma A.2.3. *Let $\mathcal{X}$ be a convex set in $\mathbb{R}^n$ and $\|\cdot\|$ be a norm with dual $\|\cdot\|_*$. Let $\omega : \mathcal{X} \to \mathbb{R}$ be a strongly convex function that satisfies*

$$\omega(z) \geq \omega(x) + \nabla\omega(x)'(z - x) + \frac{\alpha}{2}\|z - x\|^2$$

for any $x, z \in \mathcal{X}$. Let $f : \mathcal{X} \to \mathbb{R}$ be a convex differentiable function. Define, for any $x, z \in \mathcal{X}$,

$$V_1(x, z) = \omega(z) - \omega(x) - \nabla\omega(x)'(z - x)$$

$$V_2(x, z) = f(z) - f(x) - \nabla f(x)'(z - x)$$

and $\tilde{V}(x, z) = V_1(x, z) + V_2(x, z)$. Then, given any $x, y \in \mathbb{R}^n$, we have

$$\tilde{V}(v, u) - \tilde{V}(x, u) \leq f(x) - f(v) + (y + \nabla f(x))'(u - x) + \frac{\|y\|_*^2}{2\alpha}$$

for

$$v = argmin_{z \in X}\{y'(z - x) + f(z) + V_1(x, z)\} \tag{A.7}$$

and any $u \in \mathcal{X}$.

Proof. Proof of Lemma A.2.3 Given $x, y \in \mathbb{R}^n$, define v as in (A.7), and consider

$$\begin{aligned}
&V_1(v,u) - V_1(x,u) \\
&= \omega(u) - \omega(v) - \nabla\omega(v)'(u-v) - (\omega(u) - \omega(x) - \nabla\omega(x)'(u-x)) \\
&= \omega(x) - \omega(v) - \nabla\omega(v)'(u-v) + \nabla\omega(x)'(u-x) \\
&= (\nabla\omega(x) - \nabla\omega(v) - \nabla f(v) - y)'(u-v) + \nabla\omega(x)'(v-x) \\
&+ y'(u-v) + \nabla f(v)'(u-v) + \omega(x) - \omega(v) \\
&\leq y'(u-v) + \nabla f(v)'(u-v) - V_1(x,v) \\
&= (y + \nabla f(v))'(u-v) - V(x,v) \qquad \text{(A.8)}
\end{aligned}$$

where the inequality follows from $(y + \nabla f(v) + \nabla\omega(v) - \nabla\omega(x))'(u-v) \geq 0$, by the optimality of v on the convex function $y'(z-x) + f(z) + V(x,z)$ in z. On the other hand,

$$V_2(v,u) - V_2(x,u) = f(x) - f(v) - \nabla f(v)'(u-v) + \nabla f(x)'(u-x) \qquad \text{(A.9)}$$

Hence, from (A.8) and (A.9), we have

$$\begin{aligned}
& \tilde{V}(v,u) - \tilde{V}(x,u) \\
\leq\ & y'(u-v) + f(x) - f(v) + \nabla f(x)'(u-x) - V_1(x,v) \\
=\ & y'(x-v) + f(x) - f(v) + (y + \nabla f(x))'(u-x) - V_1(x,v) \qquad \text{(A.10)}
\end{aligned}$$

Using $y'(x-v) \leq \frac{\|y\|_*^2}{2\alpha} + \frac{\alpha}{2}\|x-v\|^2$, via Young's inequality (Nemirovski et al., 2009), and $V_1(x,v) \geq \frac{\alpha}{2}\|x-v\|^2$ from the definition of V_1, we have (A.10) less than or equal to

$$f(x) - f(v) + (y + \nabla f(x))'(u-x) + \frac{\|y\|_*^2}{2\alpha}$$

□ □

The following is the analog of Theorem 2.4.2 for Algorithm 2:

Theorem A.2.1. *Suppose there exists a unique optimal solution* $\mathbf{p}^* \in \mathcal{P}(\epsilon)$ *for* (2.4) *such that* $(\boldsymbol{\psi}(\mathbf{p}) - \lambda\nabla R(\mathbf{p}))'(\mathbf{p} - \mathbf{p}^*) = 0$ *if and only if* $\mathbf{p} = \mathbf{p}^*$. *Choose a non-*

increasing step size sequence $\{\gamma^k\}$ *such that*

$$\sum_{k=1}^{\infty}\gamma^k=\infty,\ \ \sum_{k=1}^{\infty}{\gamma^k}^2<\infty$$

Then $\mathbf{p}^k$ *generated in Algorithm 2 converges to* $\mathbf{p}^*$.

Note that $\boldsymbol{\psi}(\mathbf{p})-\lambda\nabla R(\mathbf{p})$ is now the gradient (with the first term obtained from a perturbation within the probability simplex) of the corresponding objective function in (A.1), and $(\boldsymbol{\psi}(\mathbf{p})-\lambda\nabla R(\mathbf{p}))'(\mathbf{p}-\mathbf{p}^*)\geq 0$ by the optimality of $\mathbf{p}^*$. We also mention that the condition of non-increasing $\{\gamma^k\}$ can be replaced readily by eventually non-increasing $\{\gamma^k\}$.

Proof. Proof of Theorem A.2.1 By defining $\mathcal{X}=\mathcal{P}$, $\omega(\mathbf{p}^k)=\sum_{i=1}^n p_i^k\log p_i^k$, $f(\mathbf{p})=-\gamma^k\lambda R(\mathbf{p}^k)=\gamma^k\lambda\sum_{i=1}^n p_i^k\log p_i^k$, $x=\mathbf{p}^k$, $y=\gamma^k\hat{\boldsymbol{\psi}}^k$, $u=\mathbf{p}^*$, $\|\cdot\|$ as L_1-norm and $\|\cdot\|_*$ as the supremum norm in Lemma A.2.3, we have $\tilde{V}(\mathbf{p}^k,\mathbf{p}^*)=(1+\gamma^k\lambda)\sum_{i_1}^n p_i^*\log\frac{p_i^*}{p_i^k}$, $\alpha=1$, and

$$\begin{aligned}&\tilde{V}(\mathbf{p}^{k+1},\mathbf{p}^*)-\tilde{V}(\mathbf{p}^k,\mathbf{p}^*)\\&\leq\gamma^k\lambda(R(\mathbf{p}^{k+1})-R(\mathbf{p}^k))+\gamma^k(\hat{\boldsymbol{\psi}}^k-\lambda\nabla R(\mathbf{p}^k))'(\mathbf{p}^*-\mathbf{p}^k)+\frac{{\gamma^k}^2\|\hat{\boldsymbol{\psi}}^k\|_\infty^2}{2}\end{aligned}\tag{A.11}$$

Let $V(\mathbf{p},\mathbf{q})=\sum_{i_1}^n q_i^k\log\frac{q_i^k}{p_i^k}$ be the KL divergence, so that $\tilde{V}(\mathbf{p}^k,\mathbf{p}^*)=(1+\gamma^k\lambda)V(\mathbf{p}^k,\mathbf{p}^*)$. Let $\mathcal{F}^k$ be the filtration generated by $\{\mathbf{p}^1,\ldots,\mathbf{p}^k\}$. Taking conditional expectation on (A.11), we have

$$\begin{aligned}&E[V(\mathbf{p}^{k+1},\mathbf{p}^*)-V(\mathbf{p}^k,\mathbf{p}^*)|\mathcal{F}^k]\\\leq\ &\lambda\gamma^k(V(\mathbf{p}^k,\mathbf{p}^*)-E[V(\mathbf{p}^{k+1},\mathbf{p}^*)|\mathcal{F}^k]-\lambda\gamma^k(R(\mathbf{p}^k)-E[R(\mathbf{p}^{k+1})|\mathcal{F}^k])\\&+\gamma^k(\boldsymbol{\psi}(\mathbf{p}^k)-\lambda\nabla R(\mathbf{p}^k))'(\mathbf{p}^*-\mathbf{p}^k)+\frac{{\gamma^k}^2E[\|\hat{\boldsymbol{\psi}}^k\|_\infty^2|\mathcal{F}^k]}{2}\end{aligned}\tag{A.12}$$

Taking expectation and summing up on both sides of (A.12), we have

$$\begin{aligned}
&\sum_{k=1}^{K} E[E[V(\mathbf{p}^{k+1},\mathbf{p}^*) - V(\mathbf{p}^k,\mathbf{p}^*)|\mathcal{F}^k]^+] \\
\leq\ & \lambda\sum_{k=1}^{K}\gamma^k(E[V(\mathbf{p}^{k+1},\mathbf{p}^*)] - E[V(\mathbf{p}^k,\mathbf{p}^*)]) - \lambda\sum_{k=1}^{K}\gamma^k(E[R(\mathbf{p}^k)] - E[R(\mathbf{p}^{k+1})]) \\
&+ \sum_{k=1}^{K}\gamma^k E[(\boldsymbol{\psi}(\mathbf{p}^k) - \lambda\nabla R(\mathbf{p}^k))'(\mathbf{p}^* - \mathbf{p}^k)] + \sum_{k=1}^{K}\frac{{\gamma^k}^2 E\|\hat{\boldsymbol{\psi}}^k\|_\infty^2}{2} \\
=\ & \lambda\left(\gamma^1 V(\mathbf{p}^1,\mathbf{p}^*) + \sum_{k=2}^{K}(\gamma^k - \gamma^{k-1})E[V(\mathbf{p}^k,\mathbf{p}^*)] - \gamma^K E[V(\mathbf{p}^{K+1},\mathbf{p}^*)]\right) \\
&- \lambda\left(\gamma^1 R(\mathbf{p}^1) + \sum_{k=2}^{K}(\gamma^k - \gamma^{k-1})E[R(\mathbf{p}^k,\mathbf{p}^*)] - \gamma^K E[R(\mathbf{p}^{K+1},\mathbf{p}^*)]\right) \\
&+ \sum_{k=1}^{K}\gamma^k E[(\boldsymbol{\psi}(\mathbf{p}^k) - \lambda\nabla R(\mathbf{p}^k))'(\mathbf{p}^* - \mathbf{p}^k)] + \sum_{k=1}^{K}\frac{{\gamma^k}^2 E\|\hat{\boldsymbol{\psi}}^k\|_\infty^2}{2} \qquad \text{(A.13)}
\end{aligned}$$

by telescoping. Note that $V(\mathbf{p},\mathbf{p}^*) \geq 0$, $R(\mathbf{p}) \geq 0$, and $R(\mathbf{p}) \leq \log n$ for any $\mathbf{p} \in \mathcal{P}$. Moreover, $(\boldsymbol{\psi}(\mathbf{p}) - \lambda\nabla R(\mathbf{p}^k))'(\mathbf{p} - \mathbf{p}^*) \geq 0$ for any $\mathbf{p} \in \mathcal{P}$. Also, by the same argument as in the proof of Theorem 2.4.2, we have $E[\|\hat{\boldsymbol{\psi}}^k\|_\infty] \leq C$ uniformly for some $C > 0$. Therefore, (A.13) is less than or equal to

$$\begin{aligned}
&\lambda\gamma^1 V(\mathbf{p}^1,\mathbf{p}^*) + \lambda\sum_{k=2}^{K}(\gamma^{k-1} - \gamma^k)\log n + \lambda\gamma^K \log n + \frac{C^2}{2}\sum_{k=1}^{K}{\gamma^k}^2 \\
&= \lambda\gamma^1 V(\mathbf{p}^1,\mathbf{p}^*) + \lambda\gamma^1 \log n + \frac{C^2}{2}\sum_{k=1}^{K}{\gamma^k}^2
\end{aligned}$$

by telescoping. Letting $K \to \infty$, we have

$$\sum_{k=1}^{\infty} E[E[V(\mathbf{p}^{k+1},\mathbf{p}^*) - V(\mathbf{p}^k,\mathbf{p}^*)|\mathcal{F}^k]^+] \leq \lambda\gamma^1 V(\mathbf{p}^1,\mathbf{p}^*) + \lambda\gamma^1\log n + \frac{C^2}{2}\sum_{k=1}^{\infty}{\gamma^k}^2 < \infty$$

By martingale convergence theorem (Theorem A.1.1 in the Appendix), we have $V(\mathbf{p}^k,\mathbf{p}^*)$ converges a.s. to some integrable random variable V_∞.

Now, taking expectation and summing up on both sides of (A.11), and by a similar

argument as above, we have

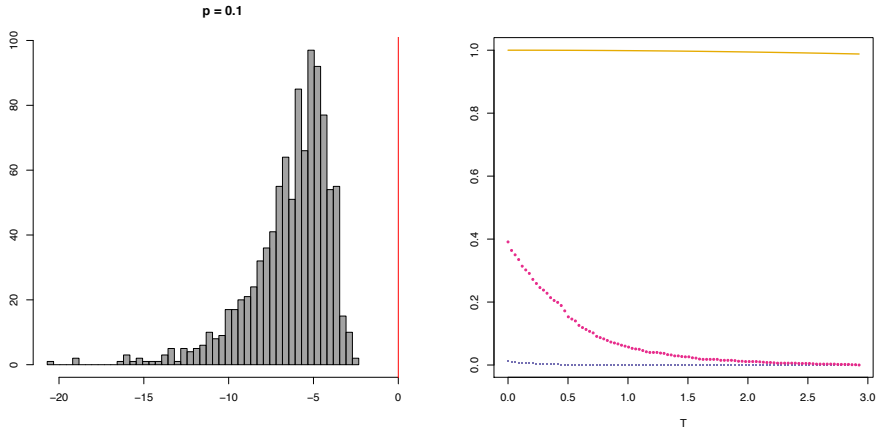
$$\sum_{k=1}^{\infty} \gamma^k E[\boldsymbol{\psi}(\mathbf{p}^k)'(\mathbf{p}^k - \mathbf{p}^*)]$$
$$\leq V(\mathbf{p}^1, \mathbf{p}^*) - EV_\infty + \lambda\gamma^1 V(\mathbf{p}^1, \mathbf{p}^*) + \lambda\gamma^1 \log n + \frac{C^2}{2}\sum_{k=1}^{\infty} \gamma^{k^2} < \infty$$

As in the proof of Theorem 2.4.2, since $\sum_{k=1}^{\infty} \gamma^k = \infty$ and $E[(\boldsymbol{\psi}(\mathbf{p}^k) - \lambda\nabla R(\mathbf{p}^k))'(\mathbf{p}^k - \mathbf{p}^*)] \geq 0$, for all k, there must exist a subsequence k_i such that $E[(\boldsymbol{\psi}(\mathbf{p}^{k_i}) - \lambda\nabla R(\mathbf{p}^k))'(\mathbf{p}^{k_i} - \mathbf{p}^*)] \to 0$ a.s.. This implies that $(\boldsymbol{\psi}(\mathbf{p}^{k_i}) - \lambda\nabla R(\mathbf{p}^k))'(\mathbf{p}^{k_i} - \mathbf{p}^*) \xrightarrow{p} 0$, which in turn implies the existence of a further subsequence l_i such that $(\boldsymbol{\psi}(\mathbf{p}^{l_i}) - \lambda\nabla R(\mathbf{p}^k))'(\mathbf{p}^{l_i} - \mathbf{p}^*) \to 0$ a.s.. From Proposition 2.4.1 Part 2, we have $\psi(\mathbf{p})$ continuous in $\mathbf{p}$. By the assumption that $(\boldsymbol{\psi}(\mathbf{p}) - \lambda\nabla R(\mathbf{p}^k))'(\mathbf{p} - \mathbf{p}^*) = 0$ only if $\mathbf{p} = \mathbf{p}^*$, and that $(\boldsymbol{\psi}(\mathbf{p}) - \lambda\nabla R(\mathbf{p}^k))'(\mathbf{p} - \mathbf{p}^*)$ is continuous in $\mathbf{p}$, we have $\mathbf{p}^{l_i} \to \mathbf{p}^*$ a.s.. Hence $V(\mathbf{p}^{l_i}, \mathbf{p}^*) \to 0$ a.s.. Since we have proved above that $V(\mathbf{p}^k, \mathbf{p}^*)$ converges a.s., this limit must be 0. Therefore, by Pinsker's inequality, we have $\mathbf{p}^k \to \mathbf{p}^*$ in total variation a.s.. This concludes the theorem. □ □

A.3 Additional Visualizations Characterizing the Quality of the Network Degree Estimator

A.3.1 Ego-centric Sampling

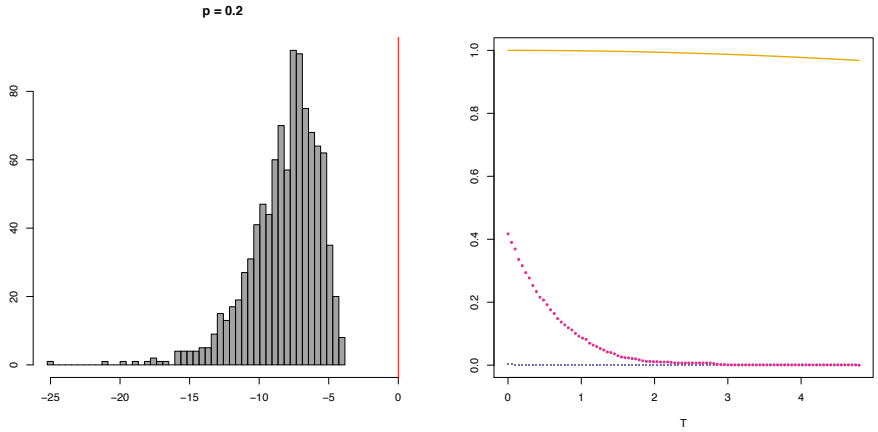
The setup of the simulation is the following - the true graph is Erdos-Renyi with 1000 vertices and 150000 edges. The plots on the left below show the sampling distribution of the target quantity and its relative location with respect to K^0 and the right hand side of the main inequality (4.20). The plots on the right visualize the concentration inequality (4.22).



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

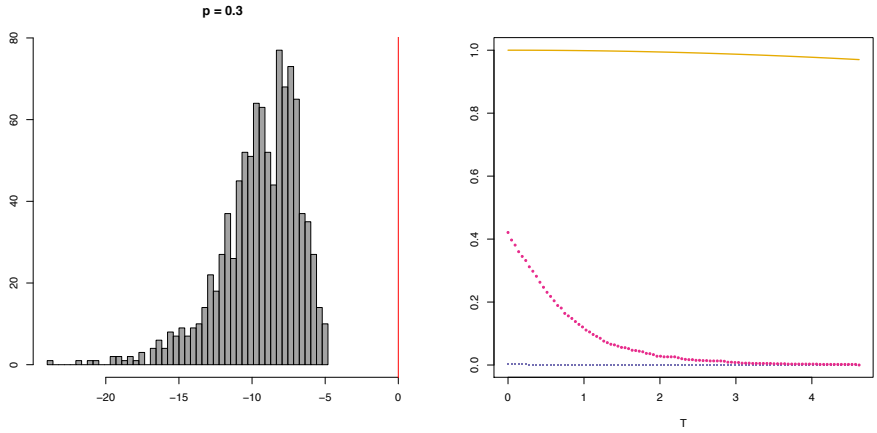
Figure A·1: $p = 0.1$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

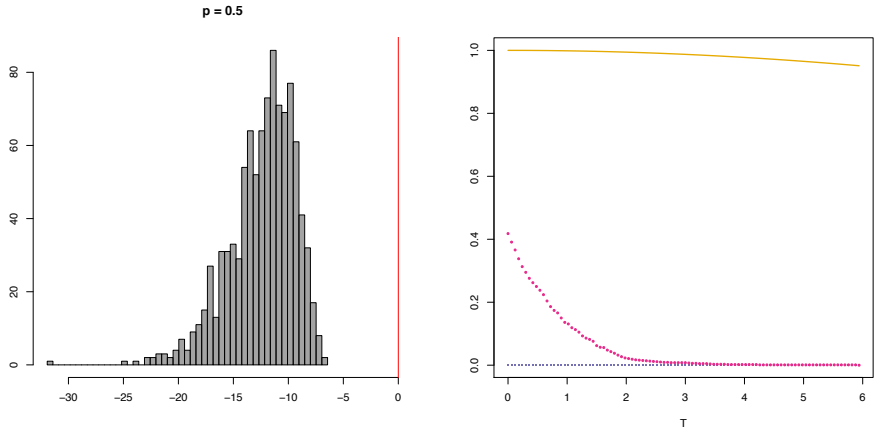
Figure A·2: $p = 0.2$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 + 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

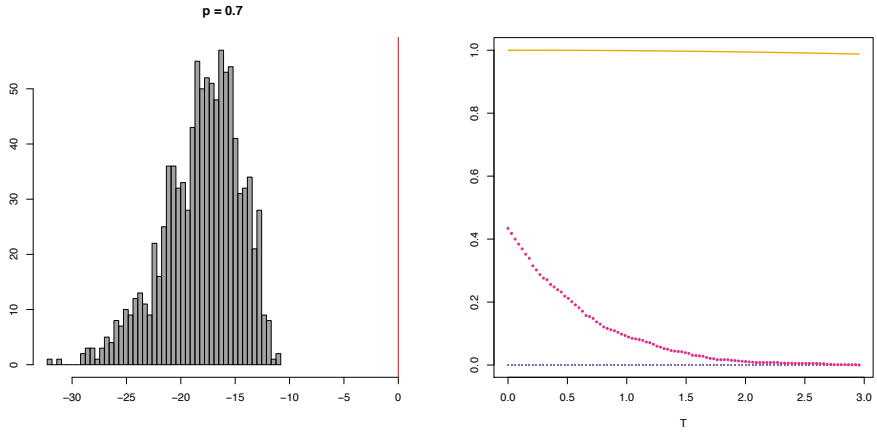
Figure A·3: $p = 0.3$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

Figure A·4: $p = 0.5$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

Figure A·5: $p = 0.7$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)

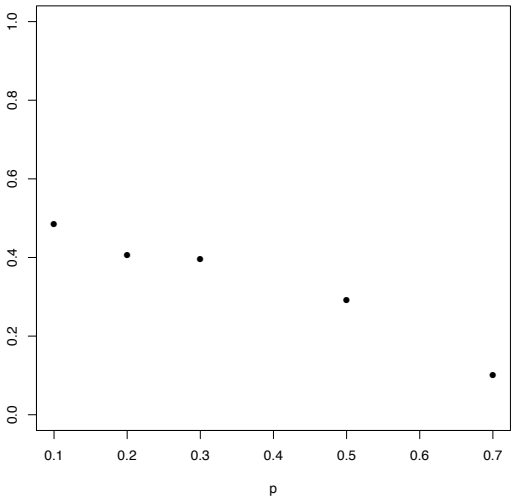
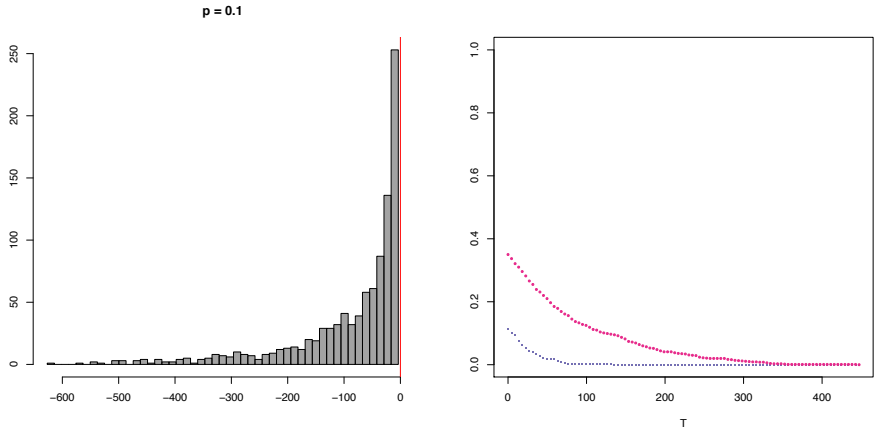


Figure A·6: Probability that the target quantity is greater than the ideal ($||P\widehat{N} - PN||^2_{C^{-1}} > K^0$) for different values of the sampling rate

A.3.2 Induced Subgraph Sampling

The true graph is Erdos-Renyi with 1000 vertices and 150000 edges. The plots in Figure A.3.2 show the sampling distribution of the target quantity and its relative

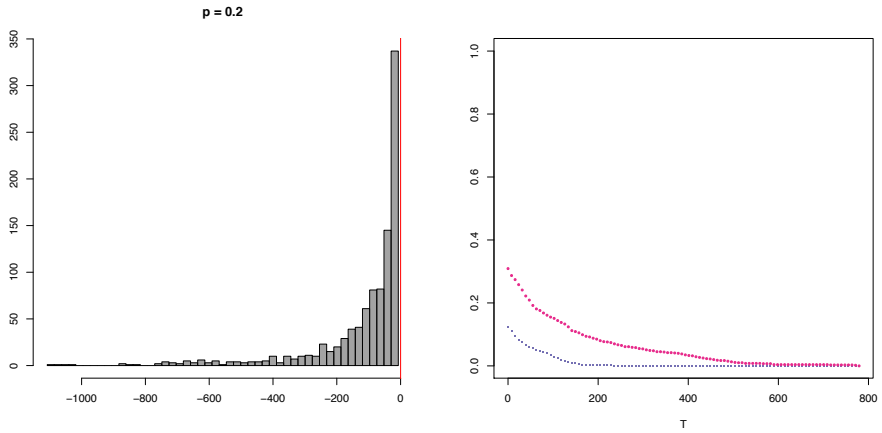
location with respect to K^0 and the right hand side of the main inequality (4.20).



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

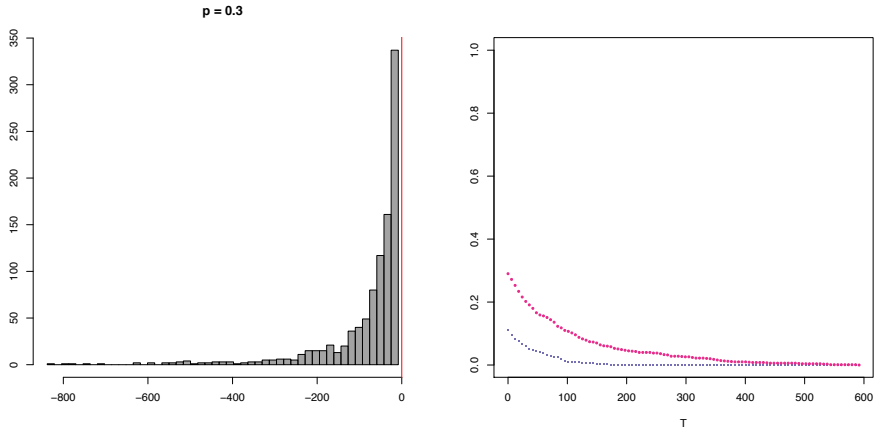
Figure A·7: $p = 0.1$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

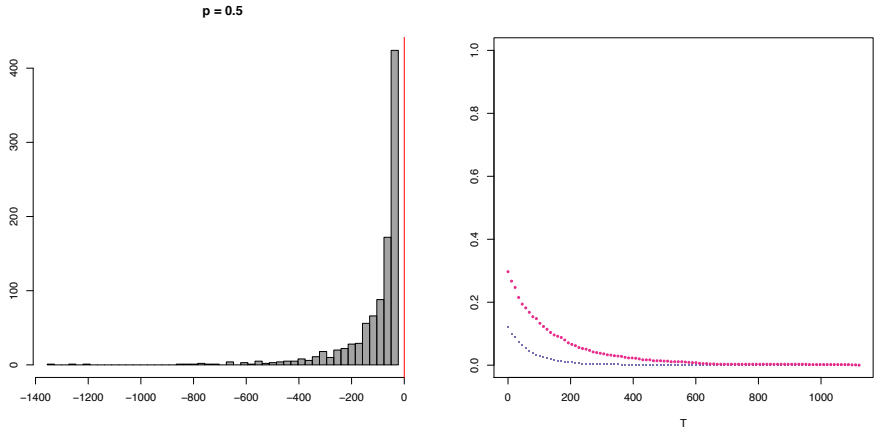
Figure A·8: $p = 0.2$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 + 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

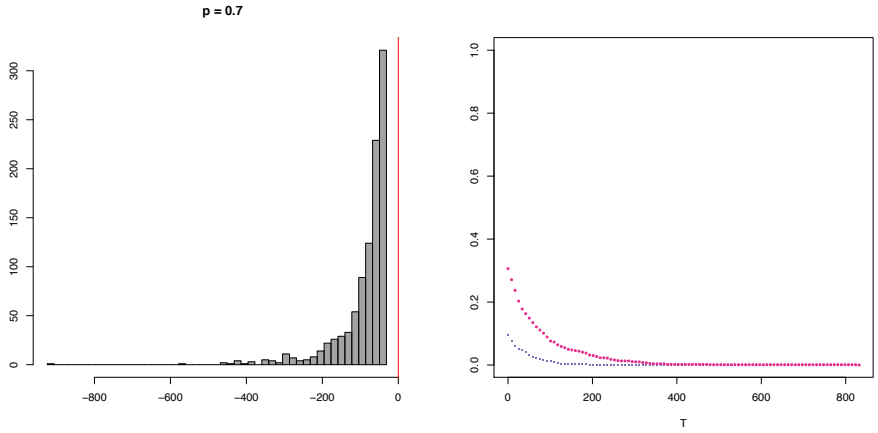
Figure A·9: $p = 0.3$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

Figure A·10: $p = 0.5$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)



(a) Sampling distribution of $||P\widehat{N} - PN||^2_{C^{-1}} - K^0 - 2\varepsilon^T A\varepsilon$

(b) Probabilities colored according to (4.22)

Figure A·11: $p = 0.7$ Visualizing Main Inequality (4.20) (Left), and Concentration Result (4.22) (Right)

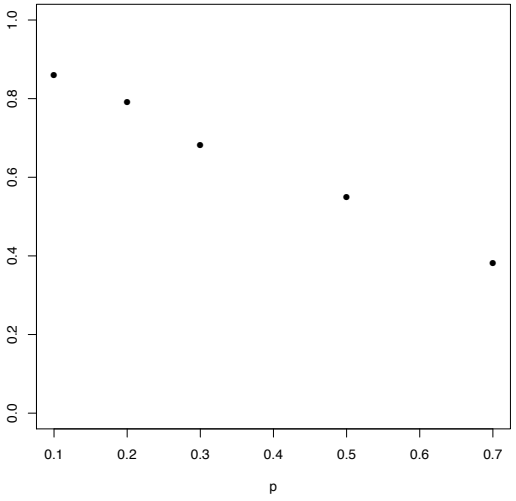


Figure A·12: Probability that the target quantity is greater than the ideal ($||P\widehat{N} - PN||^2_{C^{-1}} > K^0$) for different values of the sampling rate

References

Airoldi, E. M. and Bischof, J. M. (2015). A regularization scheme on word occurrence rates that improves estimation and interpretation of topical content. *Journal of the American Statistical Association*, (just-accepted):00–00.

Avellaneda, M., Buff, R., Friedman, C., Grandechamp, N., Kruk, L., and Newman, J. (2001). Weighted Monte Carlo: a new technique for calibrating asset-pricing models. *International Journal of Theoretical and Applied Finance*, 4(01):91–119.

Balci, O. and Sargent, R. G. (1982). Some examples of simulation model validation using hypothesis testing. In *Proceedings of the 14th Winter Simulation conference*, volume 2, pages 621–629. Winter Simulation Conference.

Banks, J., Carson, J., Nelson, B., and Nicol, D. (2009). *Discrete-Event System Simulation*. Prentice Hall Englewood Cliffs, NJ, USA, 5th edition edition.

Barron, A. R. and Sheu, C.-H. (1991). Approximation of density functions by sequences of exponential families. *The Annals of Statistics*, 19(3):1347–1369.

Barton, R. R. (2012). Tutorial: Input uncertainty in outout analysis. In *Proceedings of the 2012 Winter Simulation Conference (WSC)*, pages 1–12. IEEE.

Barton, R. R., Nelson, B. L., and Xie, W. (2013). Quantifying input uncertainty via simulation confidence intervals. *INFORMS Journal on Computing*, 26(1):74–87.

Barton, R. R. and Schruben, L. W. (2001). Resampling methods for input modeling. In *Proceedings of the 2001 Winter Simulation Conference*, volume 1, pages 372–378. IEEE.

Basawa, I., Bhat, U., and Zhou, J. (2008). Parameter estimation using partial information with applications to queueing and related models. *Statistics & Probability Letters*, 78(12):1375–1383.

Basawa, I. V., Bhat, U. N., and Lund, R. (1996). Maximum likelihood estimation for single server queues from waiting time data. *Queueing systems*, 24(1-4):155–167.

Beck, A. and Teboulle, M. (2003). Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175.

Benveniste, A., Métivier, M., and Priouret, P. (2012). *Adaptive Algorithms and Stochastic Approximations*, volume 22. Springer Science & Business Media.

Bernardo, J. M. (1979). Expected information as expected utility. *The Annals of Statistics*, pages 686–690.

Bertsekas, D. P. (1999). *Nonlinear programming*. Athena Scientific.

Bingham, N. and Pitts, S. M. (1999). Non-parametric estimation for the m/g/∞ queue. *Annals of the Institute of Statistical Mathematics*, 51(1):71–97.

Blei, D. M., Griffiths, T. L., and Jordan, M. I. (2010). The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM (JACM)*, 57(2):7.

Blum, J. R. (1954). Multidimensional stochastic approximation methods. *The Annals of Mathematical Statistics*, pages 737–744.

Borgs, C., Chayes, J. T., Lovász, L., Sós, V. T., and Vesztergombi, K. (2008). Convergent sequences of dense graphs i: Subgraph frequencies, metric properties and testing. *Advances in Mathematics*, 219(6):1801–1851.

Box, G. E. and Hill, W. J. (1967). Discrimination among mechanistic models. *Technometrics*, 9(1):57–71.

Broadie, M., Cicek, D., and Zeevi, A. (2011). General bounds and finite-time improvement for the Kiefer-Wolfowitz stochastic approximation algorithm. *Operations Research*, 59(5):1211–1224.

Cheng, R. C. and Holland, W. (1998). Two-point methods for assessing variability in simulation output. *Journal of Statistical Computation Simulation*, 60(3):183–205.

Cheng, R. C. and Holland, W. (2004). Calculation of confidence intervals for simulation output. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 14(4):344–362.

Chick, S. E. (2001). Input distribution selection for simulation experiments: accounting for input uncertainty. *Operations Research*, 49(5):744–758.

Chick, S. E. and Ng, S. H. (2002). Simulation input analysis: joint criterion for factor identification and parameter estimation. In *Proceedings of the 34th Winter Simulation Conference*, pages 400–406. Winter Simulation Conference.

Cooper, R. B. (1972). Introduction to queueing theory.

Cover, T. M. and Thomas, J. A. (1991). Information theory and statistics. *Elements of Information Theory*, pages 279–335.

Csiszár, I. (1991). Why least squares and maximum entropy? an axiomatic approach to inference for linear inverse problems. *The Annals of Statistics*, 19(4):2032–2066.

Currin, C., Mitchell, T., Morris, M., and Ylvisaker, D. (1991). Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments. *Journal of the American Statistical Association*, 86(416):953–963.

Daley, D. and Servi, L. (1998). Moment estimation of customer loss rates from transactional data. *International Journal of Stochastic Analysis*, 11(3):301–310.

Dang, C. D. and Lan, G. (2015). Stochastic block mirror descent methods for non-smooth and stochastic optimization. *SIAM Journal on Optimization*, 25(2):856–881.

DeGroot, M. H. (1962). Uncertainty, information, and sequential experiments. *The Annals of Mathematical Statistics*, pages 404–419.

Donoho, D. L. et al. (1997). Cart and best-ortho-basis: a connection. *The Annals of Statistics*, 25(5):1870–1911.

Donoho, D. L., Johnstone, I. M., Hoch, J. C., and Stern, A. S. (1992). Maximum entropy and the nearly black object. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 41–81.

Durrett, R. (2010). *Probability: Theory and Examples*. Cambridge university press.

Fearnhead, P. (2004). Filtering recursions for calculating likelihoods for queues based on inter-departure time data. *Statistics and Computing*, 14(3):261–266.

Feng, H., Dube, P., and Zhang, L. (2014). Estimating life-time distribution by observing population continuously. *Performance Evaluation*, 79:182–197.

Frank, O. (1980). Estimation of the number of vertices of different degrees in a graph. *Journal of Statistical Planning and Inference*, 4(1):45–50.

Frank, O. (1981). A survey of statistical methods for graph analysis. *Sociological methodology*, 12:110–155.

Frey, J. C. and Kaplan, E. H. (2010). Queue inference from periodic reporting data. *Operations Research Letters*, 38(5):420–426.

Friedman, J., Hastie, T., and Tibshirani, R. (2001). *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin.

Ghadimi, S. and Lan, G. (2013). Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368.

Ghadimi, S. and Lan, G. (2015). Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Mathematical Programming*, pages 1–41.

Ghosh, S. and Lam, H. (2015a). Computing worst-case input models in stochastic simulation. *Available at http://arxiv.org/pdf/1507.05609v1.pdf*.

Ghosh, S. and Lam, H. (2015b). Mirror descent stochastic approximation for computing worst-case stochastic input models. In *Proceedings of the 2015 Winter Simulation Conference*, pages 425–436. IEEE Press.

Glasserman, P. and Yu, B. (2005). Large sample properties of weighted Monte Carlo estimators. *Operations Research*, 53(2):298–312.

Glynn, P. W. (1990). Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM*, 33(10):75–84.

Hall, A. R. (2005). *Generalized method of moments*. Oxford University Press.

Hall, P. and Park, J. (2004). Nonparametric inference about service time distribution from indirect measurements. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(4):861–875.

Heckmüller, S. and Wolfinger, B. E. (2009). Reconstructing arrival processes to G/D/1 queueing systems and tandem networks. In *International Symposium on Performance Evaluation of Computer & Telecommunication Systems, 2009. SPECTS 2009.*, volume 41, pages 361–368. IEEE.

Kelton, W. D. and Law, A. M. (2000). *Simulation Modeling and Analysis*. McGraw Hill Boston.

Kennedy, M. C. and O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464.

Kim, Y. B. and Park, J. (2008). New approaches for inference of unobservable queues. In *Proceedings of the 40th Conference on Winter Simulation*, pages 2820–2825. Winter Simulation Conference.

Kleijnen, J. P. (1995). Verification and validation of simulation models. *European Journal of Operational Research*, 82(1):145–162.

Larson, R. C. (1990). The queue inference engine: Deducing queue statistics from transactional data. *Management Science*, 36(5):586–601.

L'Ecuyer, P. (1990). A unified view of the IPA, SF, and LR gradient estimation techniques. *Management Science*, 36(11):1364–1383.

Lindley, D. V. (1956). On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, pages 986–1005.

Mandelbaum, A. and Zeltyn, S. (1998). Estimating characteristics of queueing networks using transactional data. *Queueing systems*, 29(1):75–127.

Marjoram, P., Molitor, J., Plagnol, V., and Tavaré, S. (2003). Markov chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 100(26):15324–15328.

McLachlan, G. and Peel, D. (2004). *Finite mixture models*. John Wiley & Sons.

Moulines, E., Roueff, F., Souloumiac, A., and Trigano, T. (2007). Nonparametric inference of photon energy distribution from indirect measurement. *Bernoulli*, 13(2):365–388.

Nelson, B. (2016). 'Some tactical problems in digital simulation' for the next 10 years. *Journal of Simulation*, 10(1):2–11.

Nemirovski, A., Juditsky, A., Lan, G., and Shapiro, A. (2009). Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609.

Nemirovski, A. and Yudin, D. (1983). *Problem Complexity and Method Efficiency in Optimization*. Wiley, New York.

Neumaier, A. (1998). Solving ill-conditioned and singular linear systems: A tutorial on regularization. *SIAM review*, 40(3):636–666.

Park, J., Kim, Y. B., and Willemain, T. R. (2011). Analysis of an unobservable queue using arrival and departure times. *Computers & Industrial Engineering*, 61(3):842–847.

Pauca, V. P., Piper, J., and Plemmons, R. J. (2006). Nonnegative matrix factorization for spectral data analysis. *Linear algebra and its applications*, 416(1):29–47.

Pauca, V. P., Shahnaz, F., Berry, M. W., and Plemmons, R. J. (2004). Text mining using non-negative matrix factorizations. In *SDM*, volume 4, pages 452–456. SIAM.

Pickands III, J. and Stine, R. A. (1997). Estimation for an M/G/∞ queue with incomplete information. *Biometrika*, 84(2):295–308.

Price, B. S., Geyer, C. J., and Rothman, A. J. (2015). Ridge fusion in statistical learning. *Journal of Computational and Graphical Statistics*, 24(2):439–454.

Reiman, M. I. and Weiss, A. (1989). Sensitivity analysis for simulations via likelihood ratios. *Operations Research*, 37(5):830–844.

Rolls, D., Daraganova, G., Sacks-Davis, R., Hellard, M., Jenkinson, R., McBryde, E., Pattison, P., and Robins, G. (2012). Modelling hepatitis c transmission over a social network of injecting drug users. *Journal of theoretical biology*, 297:73–87.

Ross, J. V., Taimre, T., and Pollett, P. K. (2007). Estimation for queues from queue length data. *Queueing Systems*, 55(2):131–138.

Rubinstein, R. Y. (1989). Sensitivity analysis and performance extrapolation for computer simulation models. *Operations Research*, 37(1):72–81.

Santner, T. J., Williams, B. J., and Notz, W. I. (2013). *The Design and Analysis of Computer Experiments*. Springer Science & Business Media.

Sargent, R. G. (2005). Verification and validation of simulation models. In *Proceedings of the 37th Winter Simulation Conference*, pages 130–143. Winter Simulation Conference.

Schruben, L. W. (1980). Establishing the credibility of simulations. *Simulation*, 34(3):101–105.

Serfling, R. J. (2009). *Approximation Theorems of Mathematical Statistics*, volume 162. John Wiley & Sons.

Shirangi, M. G. (2014). History matching production data and uncertainty assessment with an efficient TSVD parameterization algorithm. *Journal of Petroleum Science and Engineering*, 113:54–71.

Song, E., Nelson, B. L., and Pegden, C. D. (2014). Advanced tutorial: Input uncertainty quantification. In *Proceedings of the 2014 Winter Simulation Conference*, pages 162–176. IEEE Press.

Sra, S., Nowozin, S., and Wright, S. J. (2012). *Optimization for Machine Learning*. MIT Press.

Stumpf, M. P. and Wiuf, C. (2005). Sampling properties of random graphs: the degree distribution. *Physical Review E*, 72(3):036118.

Tarantola, A. (2005). *Inverse problem theory and methods for model parameter estimation*. SIAM.

Van Campenhout, J. M. and Cover, T. M. (1981). Maximum entropy and conditional probability. *IEEE Transactions on Information Theory*, 27(4):483–489.

Wang, T.-Y., Ke, J.-C., Wang, K.-H., and Ho, S.-C. (2006). Maximum likelihood estimates and confidence intervals of an M/M/R queue with heterogeneous servers. *Mathematical Methods of Operations Research*, 63(2):371–384.

Whitt, W. (1981). Approximating a point process by a renewal process: The view through a queue, an indirect approach. *Management Science*, 27(6):619–636.

Whitt, W. (1982). Approximating a point process by a renewal process, I: Two basic methods. *Operations Research*, 30(1):125–147.

Whitt, W. (2012). Fitting birth-and-death queueing models to data. *Statistics & Probability Letters*, 82(5):998–1004.

Wunsch, C. (1996). *The Ocean Circulation Inverse Problem*. Cambridge University Press.

Zhang, Y., Kolaczyk, E. D., Spencer, B. D., et al. (2015). Estimating network degree distributions under sampling: An inverse problem, with applications to monitoring social media networks. *The Annals of Applied Statistics*, 9(1):166–199.

Zouaoui, F. and Wilson, J. R. (2004). Accounting for input-model and input-parameter uncertainties in simulation. *IIE Transactions*, 36(11):1135–1151.

Curriculum Vitae

CURRICULUM VITAE