Proceedings of the 2014 Winter Simulation Conference
A. Tolk, S. D. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, eds.

RECONSTRUCTING INPUT MODELS VIA SIMULATION OPTIMIZATION

Aleksandrina Goeva

Department of Mathematics and Statistics
Boston University
Boston, MA 02215, USA

Henry Lam

Department of Mathematics and Statistics
Boston University
Boston, MA 02215, USA

Bo Zhang

IBM T.J. Watson Research Center
Yorktown Heights, NY 10598, USA

ABSTRACT

In some service operations settings, data are available only for system outputs but not 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. In this paper, we study the problem of estimating these input distributions with only the availability of the output data, a problem usually known as the inverse problem, and we are interested in the context where stochastic simulation is required to generate the outputs. 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.

1 INTRODUCTION

In this paper, we study a model reconstruction problem: Consider a stochastic system that turns input model, represented by probability distributions, into system outputs. The system structure specification is assumed known, in the sense that the output can be simulated given the input model. Data are obtained for the outputs but not on the input model, and the task is to infer the latter only from the output data. The motivation of this problem comes from service operations. Basawa, Bhat, and Lund (1996) discussed two such instances. For example, when doing market surveys, for economic and operational reasons, cable companies may collect data on waiting times of customers instead of detailed recording of all interarrival and service times. Similarly, the waiting times of patients in clinic for consultation and surgery can be recorded, but not directly for the interarrival and service times. In both of these situations, the input model is the interarrival or the service time distribution, and the observed outputs are the waiting times of customers. From risk management and sensitivity analysis point of view, it is useful to infer these input distributions, since they can be used to generate scenarios for prediction. Inferring these distributions is the topic of study in this paper.

The above problem is referred classically as the inverse problem (Tarantola 2005). This concerns inference on parameter, function, or in our case probability distribution, when direct observation of the object of interest is unavailable. There are at least two significant branches of literature on this class of problems, which we shall survey now and also point out the new elements in our setting. The first is so-called linear inverse problem arising in the statistics literature (Csiszár 1991). This includes the reconstruction of signals from the measurement of linear transformations, and more relevantly, the reconstruction of

probability density that satisfies moment conditions. The typical approach to these problems is to set up optimization programs that minimize certain L_2-distance or statistical divergence. We shall use a similar idea in this paper, but since the input-output relation is not linear nor in closed form in our situations of interest, it forces us to use new simulation optimization strategies.

The second line of related literature lies in the emerging field of uncertainty quantification in applied mathematics. This concerns the quantification of errors made by complex models, often called surrogate models, that attempt to approximate the actual physical phenomena (Santner, Williams, and Notz 2003). These models typically contain parameters that need to be calibrated, which can be viewed as an inverse problem, and the major challenge lies on the uncertainty of how well the surrogate model approximates the truth. A predominant methodology in the literature is to use Gaussian process to approximate the model discrepancy error and to use Bayesaian updates to infer the error and also the parameter values (Kennedy and O'Hagan 2001). In this paper, we do not consider surrogate model. Instead we assume the system structure specification is completely known up to simulation. The challenge for us is the inference of the input probability distribution as a functional object. In particular, we will take a nonparametric viewpoint, i.e. not assuming any specific family for the probability distribution. This is because when the output data is abundant, the input-output relation in our setting typically allows for a nonparametric reconstruction of the input model, and this is superior to introducing parametric models that can suffer from bias issues.

Regarding past literature, we also mention that there have been several works on estimation of interarrival and service times for queues, either parametrically or nonparametrically. See, for instance, Fearnhead (2004), Basawa, Bhat, and Zhou (2008), Ross, Taimre, and Pollett (2007), and Bingham and Pitts (1999). These papers typically focus on analyzing the structure of the queues (e.g. $G/G/1$) and investigate estimation procedures that are suited to their particular structures. The observations can come from continuous or discrete time points and can be the queue lengths or the waiting times. In this paper we do not deal with observations as time series, and will focus on using output samples from a prespecified time. However, we also do not assume any particular structure of the system, but only assume it is simulable. We leave the extension to time series data to future work.

As mentioned before, we borrow ideas from linear inverse problems and attempt to set up optimization programs that incorporate output information. More concretely, we shall match the moments between simulation outputs and the observed output data. However, since we do not impose any parametric assumptions, there can be many possible solutions to satisfy these moment conditions (This is also known as non-identifiability in model calibration; see, for instance, Tuo and Wu (2013)). In order to alleviate this issue, we use entropy maximization as a criterion, and there are at least three reasons for choosing this as the objective function. The first is the natural interpretation of entropy maximizing distribution as the conditional distribution given all the prior information (Csiszár 1984, Van Campenhout and Cover 1981). Second, in the classical setting without the input-output structure, consistency results are known as the number of matched moments increases (Barron and Sheu 1991). Third, minimizing the so-called I-divergence, which contains entropy as a special case, is known to possess many desirable axiomatic properties (Csiszár 1991). Because of all these reasons, we shall set entropy as the objective in our optimization formulation, subject to moment constraints on the output level.

These optimization formulations are non-convex in nature. As such, we focus on finding local optimum and will demonstrate numerically how well it works. Our main idea of local maximization involves two elements. First, we consider a version of duality for these programs, by turning them into a sequence of optimization programs on the squared distances between the simulated and the observed moments. Second, we propose a stochastic version of the feasible direction method, or the conditional gradient method in the nonlinear programming literature, to solve these sequences of programs. This consists of deriving a representation for the stochastic gradient based on the use of Gateaux derivatives with respect to the input model, which is adapted from the work of Ghosh and Lam (2014).

This paper is organized as follows. In Section 2, we lay out the notation, setting and formulation for our problem. In Section 3, we detail our procedure to tackle our formulation, and also discuss some properties of the procedure. Then in Section 4 we report some numerical results.

2 SETTING, NOTATION, FORMULATION

We denote $\mathbf{p}$ as the probability distribution of the input model. The system output is $h(\mathbf{X})$ where $\mathbf{X} = (X_1, \ldots, X_T) \in \mathbb{R}^T$ is an i.i.d. sequence each distributed under $\mathbf{p}$, and T is the time horizon. The function $h : \mathbb{R}^T \to \mathbb{R}$ transforms the input into output. For example, $\mathbf{X}$ can denote the sequence of interarrival or service times for each customer in a queue, and $h(\mathbf{X})$ is the waiting time of the T-th customer. In our present exposition we shall only consider $\mathbf{p}$ having discrete finite support on $\{z_1, z_2, \ldots, z_K\}$, so that $\mathbf{p} = (p_1, \ldots, p_K)$. Our viewpoint is that when K is large, the discrete distribution can approximate well a continuous true model, but of course, this will introduce an extra layer of discretization error, and its quantification is an important topic for future work.

As discussed in the introduction, we are interested in situations where the output $h(\mathbf{X})$ can be observed. Let $y_1, \ldots, y_M$ be M observations of $h(\mathbf{X})$. Our task is to estimate $\mathbf{p}$. Depending on what the function h is, it may or may not be possible to perform this task. Think of, for example, h is identically equal to a constant. Then any $\mathbf{p}$ will give perfect fit. This phenomenon, often called non-identifiability, is well-documented (e.g. Tuo and Wu (2013)). To get around this issue, we shall focus on finding a distribution $\mathbf{p}$ that has the maximum entropy among all candidates that fit the observed y_i's, which also enjoys some advantageous interpretations and consistency properties (at least for simpler settings) as outlined in the introduction.

To be more precise, we let

$$\mu_j = \frac{1}{M} \sum_{m=1}^{M} y_m^j$$

be the empirical j-th moment of $h(\mathbf{X})$. We consider the optimization program

$$\begin{array}{ll} \max & -\sum_{k=1}^{K} p_k \log p_k \\ \text{subject to} & E_{\mathbf{p}}[h(\mathbf{X})^j] = \mu_j, \quad j = 1, \ldots, J \\ & \mathbf{p} \cdot \mathbf{1} = 1 \\ & \mathbf{p} \geq \mathbf{0} \end{array} \tag{1}$$

The decision variable is $\mathbf{p}$. Here $E_{\mathbf{p}}[\cdot]$ denotes the expectation under $\mathbf{p}$, and the last two constraints merely state that $\mathbf{p}$ has to be a probability distribution. The integer J represents the number of moments that are matched. The quantities $\mathbf{1}$ and $\mathbf{0}$ represent vectors with 1 and 0 in each component respectively.

We mention that if some prior information on $\mathbf{p}$ is known, the objective of (1) can be modified to incorporate such information. For example, if $\mathbf{p}$ is known to be close to a distribution, say $\mathbf{p}^0$ (for instance from other data sources), then one can minimize the Kullback-Leibler divergence, or relative entropy, between $\mathbf{p}$ and this prior distribution $\mathbf{p}^0$. In this case the objective becomes $\min \sum_{k=1}^{K} p_k \log(p_k / p_k^0)$ where $\mathbf{p}^0 = (p_1^0, \ldots, p_K^0)$.

Note that the quantities $E_{\mathbf{p}}[h(\mathbf{X})^j]$ are non-linear in $\mathbf{p}$ in general, because of the i.i.d. structure of $\mathbf{X}$. When the i.i.d. structure is not present and the quantities are linear in $\mathbf{p}$, Barron and Sheu (1991) proved that $\mathbf{p}$ converges to the true model as the number of observations M and the number of matching moments J increase at suitable rates, and moreover confidence region can be formed, under some regularity conditions (including continuity) of $\mathbf{p}$. This also motivates our formulation (1), although here we shall focus on how to solve (1) and leave the full consistency analysis for our setting to future work.

3 PROPOSED PROCEDURE

Our strategy to handle (1) consists of two parts: first, we turn (1) into a sequence of minimization programs. Second, we derive a stochastic version of the feasible direction method for finding the local optima for these programs.

3.1 Transforming into a Sequence of Stochastic Programs with Entropy Constraints

While the optimization (1) is natural, the constraints are typically non-linear and need to be simulated. As far as we know, the literature on dealing with stochastic non-convex constraints is very limited. Here we propose an approach alike quadratic penalty (Bertsekas 1999). More concretely, we consider a sequence of optimization programs

$$\begin{array}{ll} \min & \sum_{j=1}^{J}(E_{\mathbf{p}}[h(\mathbf{X})^j]-\mu_j)^2 \\ \text{subject to} & \sum_{k=1}^{K} p_k \log p_k \le \eta - \log K \\ & \mathbf{p}\cdot\mathbf{1}=1 \\ & \mathbf{p}\ge\mathbf{0} \end{array} \tag{2}$$

for increasing values of $\eta \ge 0$ (note that $-\log K$ is the minimum value of $\sum_{k=1}^{K} p_k \log p_k$, by putting $\mathbf{p}$ as the uniform distribution, and so $-\log K$ is put in the right hand side of the entropy constraint for convenience). As η increases, the optimal value of (2) decreases since the feasible region enlarges. Suppose there is a point where the optimal value decreases to zero and stays thereafter. The optimal solution to this particular η value will give the optimal solution to (1), and the corresponding $-\eta+\log K$ is the optimal value for (1). This is because any smaller η (and correspondingly larger objective value in (1)) cannot be achieved by any $\mathbf{p}$ within the feasible region in (1). On the other hand, if the optimal value of (2) is larger than zero for any $\eta \ge 0$, then there is no feasible solution to (1).

To sum up, we have

Lemma 1 Let Z_η^* be the optimal value of (2) parametrized by η. The optimal value of (1) is equal to $-\eta^*+\log K$ where $\eta^* := \min\{\eta \ge 0 : Z_\eta^* = 0\}$. Moreover, $\eta^* = \infty$ if and only if (1) is infeasible. If $\eta^* < \infty$, then the optimal solution to (2) at η^* will be optimal for (1).

The above approach resembles the use of quadratic penalty in nonlinear programming, which in a simple form will entail solving (1) by transforming it into a sequence of programs

$$\begin{array}{ll} \max & -\sum_{k=1}^{K} p_k \log p_k - c^n \sum_{j=1}^{J}(E_{\mathbf{p}}[h(\mathbf{X})^j]-\mu_j)^2 \\ \text{subject to} & \mathbf{p}\cdot\mathbf{1}=1 \\ & \mathbf{p}\ge\mathbf{0} \end{array} \tag{3}$$

where c^n is a sequence that diverges to ∞ (Bertsekas 1999). The reason we do not directly adopt (3) (or its more advanced version) is that the entropy objective function in (1), when transforming into a constraint in (2), leads to a handy analytical solution when using a feasible direction method, as will be discussed in the next section.

We mention that the squared moment distances in the objective function of (2) have also appeared in model calibration (see, for instance, Yuan, Ng, and Tsui (2013)), where the best values of a set of parameters in the system or the input model are sought after. These papers, however, often focus on parametric inference for surrogate models, which differs from our nonparametric motivation, and hence no entropy constraint is imposed in those cases, nor is there an interpretation of maximizing the entropy via η^* in the constraint as in (2).

3.2 A Feasible Direction Method

In this subsection we discuss an iterative method to find local optimum for (2). The method is based on a stochastic version of the feasible direction, also known as the conditional gradient method, in deterministic nonlinear programming. Given a current solution, say $\mathbf{p}^n$, we first approximate the gradient of $\sum_{j=1}^{J}(E_{\mathbf{p}}[h(\mathbf{X})^j]-\mu_j)^2$ at $\mathbf{p}^n$, and then consider an optimization with a linear objective as the inner product between the gradient and $\mathbf{p}$, searched over the feasible region.

Since the objective in (2) involves expectation that is not in closed form, the gradient will typically need to be estimated via simulation. In the following, we will adapt a result in Ghosh and Lam (2014) to express the gradient in expectation form. This result has the spirit of the likelihood ratio method in conventional

derivative estimation, as it involves a score-function-alike object in the expectation. From there we can run a constrained stochastic approximation scheme. We consider the following characterization:

Lemma 2 Given $\mathbf{p}^0 = (p_1^0, \ldots, p_K^0)$ where each $p_k^0 > 0$, we have

$$\nabla \left(\sum_{j=1}^{J} (E_{\mathbf{p}^0}[h(\mathbf{X})^j] - \mu_j)^2 \right) \cdot (\mathbf{p} - \mathbf{p}^0) = 2 \sum_{j=1}^{J} (E_{\mathbf{p}^0}[h(\mathbf{X})^j] - \mu_j) E_{\mathbf{p}^0}[h(\mathbf{X})^j \mathbf{S}(\mathbf{X})] \cdot (\mathbf{p} - \mathbf{p}^0)$$
$$= 2E_{\mathbf{p}^0} \left[\sum_{j=1}^{J} (h(\mathbf{X})^j - \mu_j) h(\tilde{\mathbf{X}})^j \mathbf{S}(\tilde{\mathbf{X}}) \right] \cdot (\mathbf{p} - \mathbf{p}^0) \tag{4}$$

for any probability simplex $\mathbf{p}$ on the support $\{z_1, \ldots, z_K\}$, where $\mathbf{S}(\mathbf{X}) = (S_1(\mathbf{X}), \ldots, S_K(\mathbf{X}))$ and

$$S_k(\mathbf{x}) = \sum_{t=1}^{T} \frac{I_k(x_t)}{p_k^0} - T$$

for any $\mathbf{x} = (x_1, \ldots, x_T) \in \mathbb{R}^T$. Here $I_k(x) = 1$ if $x = z_k$ and 0 otherwise. Moreover, $\mathbf{X}$ and $\tilde{\mathbf{X}}$ are two independent copies of the sequence $\mathbf{X}$ under $\mathbf{p}^0$.

The optimization objective in Ghosh and Lam (2014) does not involve the square operation, and so does not need to generate two independent copies of $\mathbf{X}$ and $\tilde{\mathbf{X}}$. As discussed in their work, the quantity $E_{\mathbf{p}^0}[h(\mathbf{X})^j \mathbf{S}(\mathbf{X})]$ is the Gateaux derivative of $E_{\mathbf{p}^0}[h(\mathbf{X})^j]$ with respect to the probability simplex $\mathbf{p}$. In other words, for each k, denoting $\mathbf{1}_k$ as the vector with 1 at the k-th component and 0 otherwise (i.e. a degenerate probability mass at z_k), a perturbation of probability distribution from $\mathbf{p}^0$ to the mixture $(1-\varepsilon)\mathbf{p}^0 + \varepsilon \mathbf{1}_k$ leads to

$$E_{\mathbf{p}^0}[h(\mathbf{X})^j S_k(\mathbf{X})] = \lim_{\varepsilon \to 0} \frac{1}{\varepsilon} (E_{(1-\varepsilon)\mathbf{p}^0 + \varepsilon \mathbf{1}_k}[h(\mathbf{X})^j] - E_{\mathbf{p}^0}[h(\mathbf{X})^j]). \tag{5}$$

We point out that having a Gateaux derivative interpretation is preferable to using the standard partial derivative $(\partial/\partial p_k) E_{\mathbf{p}}[h(\mathbf{X})^j]|_{\mathbf{p}=\mathbf{p}^0}$ directly. This is because the gradient $\nabla E_{\mathbf{p}^0}[h(\mathbf{X})^j]$ itself does not offer probability interpretation, as the coordinate-wise perturbation of $\mathbf{p}$ shoots outside the feasible region of probability simplex. This in turn makes the gradient non-simulable. On the other hand, the definition (5) and hence (4) involves a simulable gradient. In particular, the function $S_k(\mathbf{x})$ can be interpreted as a nonparametric version of the likelihood ratio in the likelihood ratio method (or the score function method) in classical derivative estimation applied to the function $E_{\mathbf{p}^0}[h(\mathbf{X})^j]$, whose resemblance can be seen in the proof of Lemma 2.

Proof of Lemma 2. We focus on $E_{\mathbf{p}^0}[h(\mathbf{X})^j]$ for any given j. To facilitate our proof, let $Z_j(\mathbf{p}^0) = E_{\mathbf{p}^0}[h(\mathbf{X})^j]$. Consider

$$\nabla \left((Z_j(\mathbf{p}^0) - \mu_j)^2 \right) \cdot (\mathbf{p} - \mathbf{p}^0) = 2(Z_j(\mathbf{p}^0) - \mu_j) \nabla Z_j(\mathbf{p}^0) \cdot (\mathbf{p} - \mathbf{p}^0).$$

We argue that $\nabla Z_j(\mathbf{p}^0) \cdot (\mathbf{p} - \mathbf{p}^0) = \mathbf{g}_j(\mathbf{p}^0) \cdot (\mathbf{p} - \mathbf{p}^0)$, where $\mathbf{g}_j(\mathbf{p}^0) = \left(\frac{d}{d\varepsilon} Z_j((1-\varepsilon)\mathbf{p}^0 + \varepsilon \mathbf{1}_k)\big|_{\varepsilon=0} \right)_{k=1,\ldots,K}$. Indeed,

$$\frac{d}{d\varepsilon} Z_j((1-\varepsilon)\mathbf{p}^0 + \varepsilon \mathbf{1}_k)\Big|_{\varepsilon=0} = \nabla Z_j(\mathbf{p}^0) \cdot (\mathbf{1}_k - \mathbf{p}^0) = \frac{\partial}{\partial p_k} Z_j(\mathbf{p}^0) - \nabla Z_j(\mathbf{p}^0) \cdot \mathbf{p}^0$$

and so

$$\mathbf{g}_j(\mathbf{p}^0) = \nabla Z_j(\mathbf{p}^0) - (\nabla Z_j(\mathbf{p}^0) \cdot \mathbf{p}^0) \mathbf{1}.$$

Therefore,

$$\mathbf{g}_j(\mathbf{p}^0) \cdot (\mathbf{p} - \mathbf{p}^0) = \nabla Z_j(\mathbf{p}^0) \cdot (\mathbf{p} - \mathbf{p}^0) - (\nabla Z_j(\mathbf{p}^0) \cdot \mathbf{p}^0)(\mathbf{1} - \mathbf{1}) = \nabla Z_j(\mathbf{p}^0) \cdot (\mathbf{p} - \mathbf{p}^0)$$

which concludes our claim.

Next, we show that the k-th component of $\mathbf{g}_j(\mathbf{p}^0)$ is $E_{\mathbf{p}^0}[h(\mathbf{X})^j S_k(\mathbf{X})]$. Indeed,

$$\begin{aligned}
& \frac{d}{d\varepsilon} Z_j((1-\varepsilon)\mathbf{p}^0 + \varepsilon \mathbf{1}_k)\Big|_{\varepsilon=0} \\
= & \frac{d}{d\varepsilon} \sum_{k_1,\ldots,k_T} h(z_{k_1}, z_{k_2}, \ldots, z_{k_T}) \prod_{t=1}^{T} ((1-\varepsilon)p_{k_t}^0 + \varepsilon I_k(z_{k_t}))\Big|_{\varepsilon=0} \\
= & \sum_{k_1,\ldots,k_T} h(z_{k_1}, z_{k_2}, \ldots, z_{k_T}) \frac{d}{d\varepsilon} \sum_{t=1}^{T} \log((1-\varepsilon)p_{k_t}^0 + \varepsilon I_k(z_{k_t}))\Big|_{\varepsilon=0} \prod_{t=1}^{T} p_{k_t}^0 \\
= & \sum_{k_1,\ldots,k_T} h(z_{k_1}, z_{k_2}, \ldots, z_{k_T}) \sum_{t=1}^{T} \frac{-p_{k_t}^0 + I_k(z_{k_t})}{p_{k_t}^0} \prod_{t=1}^{T} p_{k_t}^0 \\
= & E_{\mathbf{p}^0}[h(\mathbf{x}) S_k(\mathbf{x})].
\end{aligned}$$

The first equality in (4) follows by summing up over all the $j = 1, \ldots, J$ moments. The second equality in (4) is obvious by the property of independence. □

With the above gradient characterization, we now design our stochastic approximation scheme. Say we start from an initial probability simplex $\mathbf{p}^0$. For each iteration, given $\mathbf{p}^n$, Algorithm 1 gives the procedure to update to $\mathbf{p}^{n+1}$. Under suitable conditions, Algorithm 1 should converge to a stationary probability simplex $\mathbf{p}^*$, in the sense that $\mathbf{p}^*$ solves the optimization problem

$$\begin{array}{ll}
\min & 2\sum_{j=1}^{J} (E_{\mathbf{p}^*}[h(\mathbf{X})^j] - \mu_j) E_{\mathbf{p}^0}[h(\mathbf{X})^j \mathbf{S}(\mathbf{X})] \cdot (\mathbf{p} - \mathbf{p}^*) \\
\text{subject to} & \sum_{k=1}^{K} p_k \log p_k \le \eta - \log K \\
& \sum_{k=1}^{K} p_k = 1 \\
& p_k \ge 0 \text{ for } k = 1, \ldots, K
\end{array}$$

Although here we do not give a rigorous proof of convergence, we shall list out and explain all the specifications of the algorithm below.

1. The sequence of step sizes ε^n is taken to be θ/n for some $\theta > 0$. This is the typical choice in running stochastic approximation scheme for unconstrained problems, or for constrained problems with the use of projections.
2. The sample sizes m_1^n and m_2^n at step n grows to ∞ as n increases, say at a rate cn^γ for some $\gamma > 0$. The reason this is needed is because each $\mathbf{q}^n$ carries a bias relative to the solution of (7) that has ξ_k replaced by the actual gradient $2\sum_{j=1}^{J} (E_{\mathbf{p}^*}[h(\mathbf{X})^j] - \mu_j) E_{\mathbf{p}^0}[h(\mathbf{X})^j \mathbf{S}(\mathbf{X})]$, even though ξ_k itself is unbiased for estimating $2\sum_{j=1}^{J} (E_{\mathbf{p}^*}[h(\mathbf{X})^j] - \mu_j) E_{\mathbf{p}^0}[h(\mathbf{X})^j \mathbf{S}(\mathbf{X})]$. This bias can accumulate over the iterations, and the growing sample size aims to compensate this error.
3. The optimization (7) has analytical solution, given by

$$q_k = \frac{e^{\beta \xi_k}}{\sum_{k=1}^{K} e^{\beta \xi_k}} \tag{8}$$

where $\beta < 0$ satisfies

$$\beta \frac{\sum_{k=1}^{K} \xi_k e^{\beta \xi_k}}{\sum_{k=1}^{K} e^{\beta \xi_k}} - \log \frac{1}{K} \sum_{k=1}^{K} e^{\beta \xi_k} = \eta \tag{9}$$

if this root exists. Otherwise $q_k = 1$ for $k = \operatorname{argmin} \xi_k$ and 0 for all other k's. The expression (8) follows by solving the first order condition of the Lagrangian formulation of (7) and then verifying optimality through a convexity argument. Finding β in (9) requires a one-dimensional deterministic search. See, for example, Glasserman and Xu (2014) and Lam (2013).

Algorithm 1 Each Iteration of the Stochastic Feasible Direction Method for Optimization (2)

1. Construct a probability simplex $\tilde{\mathbf{p}}^n$ such that it is absolutely continuous with respect to $\mathbf{p}^n$ and that each nonzero component of $\tilde{\mathbf{p}}^n$ is at least as large as some small positive constant δ.

2. Simulate m_1^n i.i.d. copies of $h(\mathbf{X}_T)^j \tilde{S}_k(\mathbf{X}_T)$ under $\tilde{\mathbf{p}}^n$, simultaneously for all $k = 1,\ldots,K$ and $j = 1,\ldots,J$, where

$$\tilde{S}_k(\mathbf{X}_T) = \sum_{t=1}^{T} \frac{I_k(x_t) - p_k^n}{\tilde{p}_k^n}$$

For convenience, let these m_1^n samples be $w_{kji}, i = 1,\ldots,m_1^n$.

3. Simulate another m_2^n i.i.d. copies of $h(\mathbf{X}_T)^j$ simultaneously for each $j = 1,\ldots,J$. Say they are $r_{ji}, i = 1,\ldots,m_2^n$.

4. Compute

$$\xi_k = 2\sum_{j=1}^{J} \left(\frac{1}{m_1^n} \sum_{i=1}^{m_1^n} w_{kji} \right) \left(\frac{1}{m_2^n} \sum_{i=1}^{m_2^n} r_{ji} - \mu_j \right) \tag{6}$$

for each $k = 1,\ldots,K$.

5. Solve the optimization:

$$\begin{array}{ll} \min & \sum_{k=1}^{K} p_k \xi_k \\ \text{subject to} & \sum_{k=1}^{K} p_k \log p_k \le \eta - \log K \\ & \sum_{k=1}^{K} p_k = 1 \\ & p_k \ge 0 \text{ for } k = 1,\ldots,K \end{array} \tag{7}$$

Say the optimal solution is $\mathbf{q}^n$.

6. Update $\mathbf{p}^{n+1} = (1-\varepsilon^n)\mathbf{p}^n + \varepsilon^n \mathbf{q}^n$ for some step size ε^n.

4. In general, m_1^n should also scale with T linearly, i.e. $m_1^n = \Theta(T)$. This is because the variance of each sample of $S_k(\mathbf{X})$ is typically of order T, and offsetting this order T variance by picking an order T sample size tends to give more stable convergence.
5. The measure $\tilde{\mathbf{p}}^n$ introduced in each step is a change-of-measure from $\mathbf{p}^n$ to ensure that the variance of the gradient estimate does not blow up. Correspondingly, the quantity $\tilde{\mathbf{S}}$ is modified from $\mathbf{S}$ that takes into account the likelihood ratio between $\mathbf{p}^n$ and $\tilde{\mathbf{p}}^n$. The reason why the variance of the direct samples of $h(\mathbf{X})S_k(\mathbf{X})$ may blow up is because, from the form of $S_k(\mathbf{X})$, it can be seen that $E_{\mathbf{p}}[h(\mathbf{X})S_k(\mathbf{X})] = \sum_{t=1}^{T} E_{\mathbf{p}}[h(\mathbf{X})|X_t = z_k] - TE_{\mathbf{p}}[h(\mathbf{X})]$. This probabilistic interpretation implies that the variance of the direct estimator for $E_{\mathbf{p}}[h(\mathbf{X})|X_t = z_k]$ can have huge variance as the probability of the event $X_t = z_k$ gets tiny. The use of $\tilde{\mathbf{p}}^n$ is designed to avoid such issue. One specific choice of $\tilde{\mathbf{p}}^n$ is the following: if any nonzero component of $\mathbf{p}^n$ is less than δ, move enough weight from the largest component to that component to make up the shortfall. Repeat until all nonzero components have weights at least δ.

Algorithm 1 aims to obtain local optimum for (2), given a specific value of η. Going back to the original formulation (1), we need to apply this algorithm across different values of η. By Lemma 1 and its preceding discussion, one would expect the optimal objective value to decrease as η decreases. The first η where the optimal objective value is exactly zero will correspond to the optimal value for (1). In general, this cannot be evaluated exactly. In the numerical examples in the next section, we will merely run Algorithm 1 across η and scrutinize the plot of the optimal objective values to pick the η. In future work, we will investigate the use of bisection method or other search method to find this thresholding η, which we believe can be done given the monotone relation of η with the optimal values.

4 NUMERICAL RESULTS

In this section we provide some numerics on using Algorithm 1 to find local optima for a sequence of optimizations (2). As a proof of concept, we consider the simple setting of single-server queues. We assume that the service times are known to be i.i.d. exponential with rate 1.2, but the interarrival times are i.i.d. with unknown distribution. To fit into our setup, for the first experiment, we assume the true interarrival time distribution is discrete over five points spanning across 0.5 to 2.5, with mean approximately 1. For the second experiment, we assume the distribution is discrete over 100 points across 0 to 5, also with mean approximately 1, which serves as a more realistic scenario.

For each of these experiments, we form a synthetic data set of $100,000$ samples from the output of the queue, which is set to be the waiting time of the 15-th customer. The large size of the synthetic data essentially sets the moments to be very close to the true moments, and this is intentional as we want to focus on assessing whether we can reconstruct the input distribution without introducing another layer of errors from the outputs. We set up our optimization (1) to match up to the fourth moment, i.e. $J = 4$. Converting (1) into (2), we then find the local optima for (2) using Algorithm 1. We do this across different values of η, and we set $\theta = 2/3$, $\gamma = 1$, and $m_1^n = m_2^n = 14n$ in Algorithm 1.

For the first experiment, i.e. the interarrival time distribution has a support of 5 grid points, we vary η from 0.1 to 0.4, at a scale of 0.01. For each η we run 300 iterations for Algorithm 1. Figure 1 shows the values of each of the five probability weights against number of iterations, for $\eta = 0.21$, which is a value roughly corresponding to the entropy of the true distribution, i.e. η^* in Lemma 1.

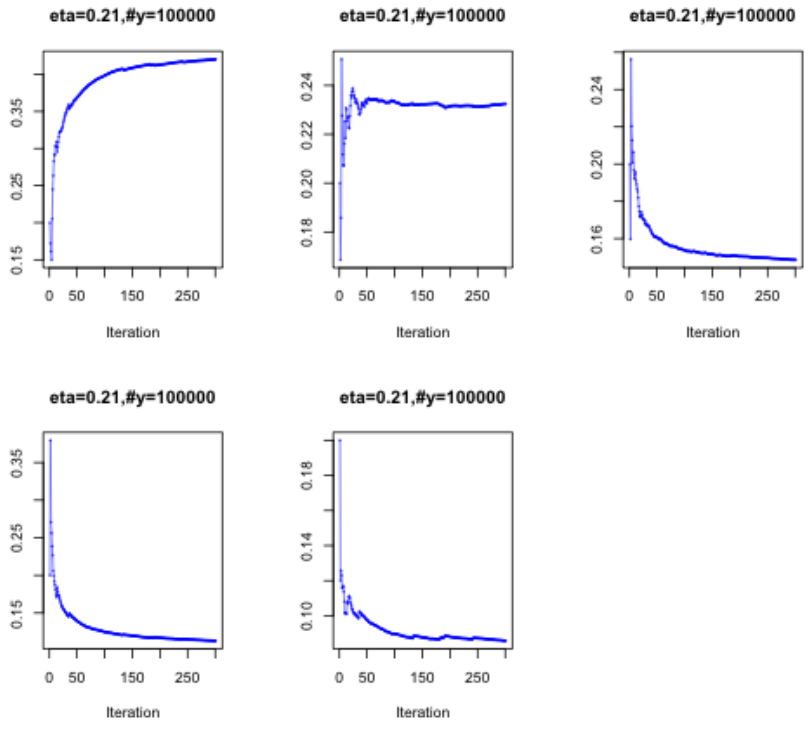

Figure 1: Trace plot for the 5 estimated probability weights on the grid against the number of iterations

We found that these trace plots are quite typical, i.e. for different η the pattern is largely the same. The algorithm seems to have converged by 300 iterations.

Next, Figure 2 below shows the estimated probability weights compared to the truth. Each of the graphs shows the comparison after the 300-th iteration. The red dots are the truth, and the blue asterisks are the

estimated weights. It can be seen that the differences between the weights shrink as η increases, although there seems to be still some discrepancy left at η greater than or equal to 0.21 (the value corresponding to the entropy of the true distribution). We expect this discrepancy to reduce further if we add more matching moments in our formulation.

One observation is that even though the estimated weights do not seem too far off even for small values of η, e.g. $\eta = 0.1$, and the improvement in approximating the truth does not seem too dramatical on these four graphs as η increases, we will show in Figure 3 that indeed the objective value in (2) drops significantly as η increases.

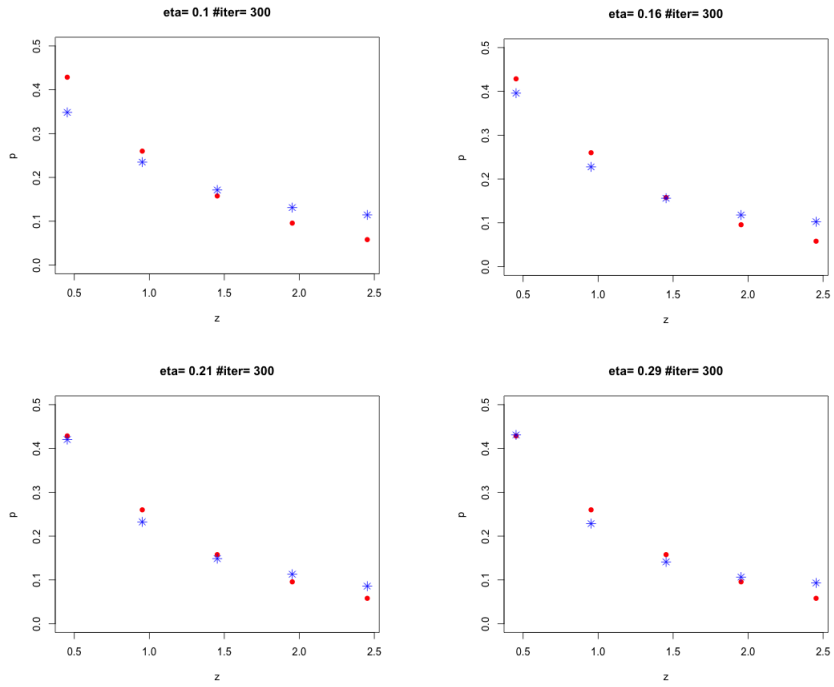

Figure 2: Plot of estimated vs true probability weights for the 5-grid distribution for different η's

Figure 3 shows the (local) optimal values of (2) against 31 values of η spanning from 0.1 to 0.4. As η^* is around 0.21, we expect the optimal value to be 0 when $\eta \approx 0.21$ or above. This is indeed the case in Figure 3. The optimal value is decreasing for small η up to about $\eta = 0.20$, and remains close to 0 thereafter. We note that these objective values are obtained by running simulation on the moments of outputs and then evaluating using the objective function in (2), and hence are subject to sampling error. The magnitude of the objective value can be large, as the graph suggests, because of the contribution from the square of the fourth moment, which amplifies the objective value greatly. Note that even though we are able to locate the threshold roughly to be 0.20 from the graph, we have not provided a principled approach to find this numerically in this work. One future direction is to design decision rules to locate the threshold and to analyze the associated error bounds.

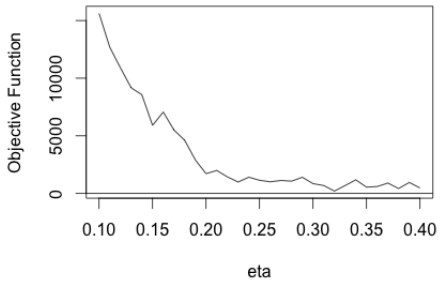

Figure 3: Optimal values of optimization (2) against η

We have repeated the experiment for a 100-grid interarrival time distribution, again with mean approximately 1. Figure 4 shows a typical trace plot of the first 5 weights (out of 100) against the number of iterations, which runs up to $1,000$ in this case.

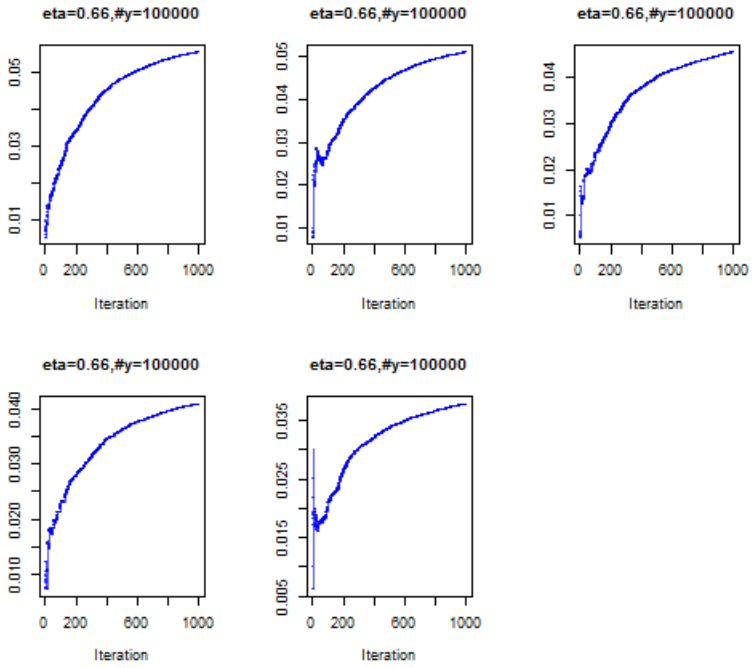

Figure 4: Trace plot for the first 5 estimated probability weights on the 100-grid against the number of iterations

Figure 5 shows the estimated probability weights over the 100 grid points after the $1,000$-th iteration, compared to the truth. The red dots represent the truth, and the blue asterisks represent the estimates. In this plot the value of $\eta = 0.66$ roughly corresponds to the entropy of the true distribution. We can see that the estimation approximates the distribution fairly well, albeit with some minor discrepancy. From the trace plots we notice that if we let the experiment run for more than $1,000$ iterations, the estimates might approximate the true probability weights even better. As in the first experiment, we expect that adding more matching moments here will improve our approximation.

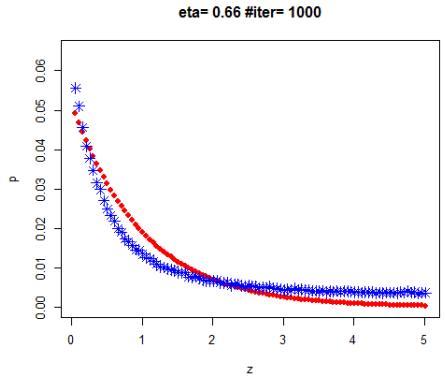

Figure 5: Plot of estimated vs true probability weights for the 100-grid distribution for an η that approximately corresponds to the entropy of the true distribution

5 DISCUSSION OF FUTURE WORK

The present paper presents an initial attempt to solve inverse problems in contexts where simulation is required for generating outputs, and when input objects are models represented by probability distributions, using a nonparametric approach. There are several directions that we are investigating ahead. The first is consistency results on our method as the number of matched moments increases relative to the output sample size. This will give a theoretical counterpart to the classical inverse problems in density estimation under moment conditions, and can shed light on how many moments, or even basis functions other than powers, that can give optimal results. Second, we plan to give rigorous convergence rate analysis for our stochastic approximation scheme, as well as to design efficient search method to pick the optimal η. Multivariate and high dimensional settings, with several input models, are also worth investigating as these are the typical scenarios in practice. Lastly, it will be important to test the applicability of our method in capturing more sophisticated, such as multimodal, input distributions in future work.

REFERENCES

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

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

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

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

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

Csiszár, I. 1984. "Sanov property, generalized I-projection and a conditional limit theorem". *The Annals of Probability*:768–793.

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.

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

Ghosh, S., and H. Lam. 2014. "A stochastic optimization approach to assessing model uncertainty". *Working paper*.

Glasserman, P., and X. Xu. 2014. "Robust risk measurement and model risk". *Quantitative Finance* 14 (1): 29–58.

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

Lam, H. 2013. "Robust Sensitivity Analysis for Stochastic Systems". *arXiv preprint arXiv:1303.0326*.

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

Santner, T. J., B. J. Williams, and W. I. Notz. 2003. *The design and analysis of computer experiments*. Springer.

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

Tuo, R., and J. Wu. 2013. "A theoretical framework for calibration in computer models: parameterization, estimation and convergence properties". *Preprint*.

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

Yuan, J., S. H. Ng, and K. L. Tsui. 2013. "Calibration of stochastic computer models using stochastic approximation methods". *IEEE Transactions on Automation Science and Engineering* 10 (1): 171–186.

AUTHOR BIOGRAPHIES

ALEKSANDRINA GOEVA is a PhD student in the Department of Mathematics and Statistics at Boston University. She received her B.A. degree in Applied Mathematics from Sofia University in 2011. Her research interests include Bayesian Statistics, Monte Carlo Markov Chain methods and Machine Learning. Her email is agoeva@bu.edu.

HENRY LAM is an Assistant Professor in the Department of Mathematics and Statistics at Boston University. He graduated from Harvard University with a Ph.D. degree in statistics in 2011. His research focuses on large-scale stochastic simulation, rare-event analysis, and simulation optimization, with application interests in service systems and risk management. His email is khlam@bu.edu.

BO ZHANG is a Research Staff Member at the Business Solutions and Mathematical Sciences Department of the IBM T.J. Watson Research Center in New York, USA. His research interests are stochastic models, stochastic optimization and control, Monte Carlo simulation, and their various applications. His email address is bozhang@gatech.edu and his webpage is at http://www2.isye.gatech.edu/~bzhang34/.