Optimization in the Small-Data, Large-Scale Regime

Vishal Gupta

Abstract This chapter introduces the small-data, large-scale optimization regime, an asymptotic setting that arguably better describes certain data-driven optimization applications than the more traditional large-sample regime. We highlight unique phenomena that emerge in the small-data, large-scale regime, and show how these phenomena cause certain traditional data-driven optimization algorithms like sample average approximation (SAA) to fail. We then propose a new debiasing approach that has provably good performance in this regime, highlighting a new path forward for research and development into these types of applications.

1 Why Small Data?

Despite the promises of Big Data, data in modern operations research applications can be scarce. Worse, this data scarcity is typically unavoidable. For example, in some systems, such as financial markets, data are rapidly time-varying. Consequently, only the most recent data are indicative of current conditions and obtaining additional, relevant data is impossible. In other settings, data-collection can be expensive, either financially or operationally. For example, when optimizing early-childhood interventions to prevent adult obesity, it might take years to observe a single data point. Finally, in some settings such as medicine and education, data are highly regulated by privacy laws. These laws prohibit decision-makers from directly accessing protected data, leaving them instead to work with either i) coarser, aggregated "summary" data (see, e.g., (Gupta et al., 2020) for discussion) or ii) anonymized data that are deliberately contaminated to protect privacy (see, e.g., (Dwork, 2008)). In all these settings, we either cannot access as much data as we would

Vishal Gupta
USC Marshall School of Business, Los Angeles CA 90089 e-mail: guptavis@usc.edu

ideally like, or we cannot access the kind of data we would ideally like. In this sense, data are fundamentally scarce, and any estimates of uncertain parameters in the system necessarily have low precision.

In the context of operations management, specifically, data scarcity sometimes arises as a result of personalization or customization. Indeed, real-world applications often require making thousands of separate decisions simultaneously, each customized to a particular, person, product and instant of time. Such personalization exacerbates data scarcity, since there may only be a few similar people, products, and times historically from which to draw.

This "personalization induced data scarcity" is not simply a pathological possibility, but rather a commonplace occurrence. For example, Gupta and Rusmevichientong (2021) studies data from a large online retailer that sells hundreds of thousands of products per quarter. The authors show that even among the most popular product categories, half of all product types sold have fewer than 10 total sales in the last quarter. Similarly, the MovieLens25M dataset (Harper and Konstan, 2015) consists of 25 million ratings of 62,000 movies by 162,000 users. Despite this size, 60% of movies have 10 or fewer ratings. Finally, Liu and Li (2017) observe that even when using real-time GPS traffic data from millions of drivers, many arcs in urban road network are traveled relatively infrequently, leading to "stale" data that are too old to be meaningful. Similar examples, with large datasets describing a huge number of uncertain parameters but where most parameters have a fairly limited amount of relevant data, abound throughout operation research.

In the absence of strong modeling assumptions, data scarcity limits our ability to estimate uncertain quantities effectively. Hence, most uncertain parameters in these settings necessarily admit, at best, low-precision estimates. We term decision-making settings with these features –i.e., *many* uncertain parameters, each with a *low precision estimate* – the small-data, large-scale regime.

Despite the prevalence of applications in the small-data, large-scale regime, however, most data-driven optimization methods are inspired by and analyzed in the large-sample regime, i.e., the setting where the available data are increasing, and all uncertain parameters admit increasingly precise estimates. Many data-driven algorithms behave *very* differently in these two regimes, suggesting provably good theoretical performance in the large-sample regime might tell us nothing about an algorithm's practical performance in the small-data, large-scale regime.

Consequently, this chapter focuses on the small-data, large-scale regime, with particular emphasis on unique phenomena not typically seen in the large-sample regime. Our goal is twofold: i) understand how these phenomena impact the performance of certain "traditional" data-driven optimization algorithms, and, ii) exploit these new phenomena to design better algorithms tailored to applications in this regime.

Philosophically, the distinction between large-sample and small-data, large-scale regimes mirrors the distinction between the macroscopic and molecular scales in physics. We now know that certain phenomenon, like statistical and quantum mechanical effects, are essentially negligible when modeling every day objects at the macroscopic scale such as cars, people and buildings. However, at the molecular scale, these forces dominate other forces such as gravity and friction, and hence objects at this scale behave in “unintuitive” ways. Indeed, the guiding principle of nanotechnology is that one can engineer systems at the molecular scale to directly exploit these unintuitive phenomena to achieve performance not possible at the macroscopic sale.

Our goal in studying the small-data, large-scale regime is similar. We seek to describe and understand the new “unintuitive” phenomena that emerge in this regime in order to exploit them in the aforementioned applications, much in the same way nanotechnology does for the molecular scale.

1.1 Structure

The remainder of this chapter is organized as follows: We first introduce a somewhat stylized data-driven optimization model that allows us to easily contrast the small-data, large-scale and large-sample regimes. We then highlight unique phenomenon arising in this regime and show that algorithms designed with large-sample intuition can have very poor behavior in the small-data, large-scale regime. In the second part of the chapter, we develop an alternative approach based on debiasing to illustrate that there do exist – at least in our stylized model – simple algorithms which have excellent behavior in both regimes.

2 Contrasting the Large-Sample and Small-Data, Large-Scale Regimes

2.1 Model

We begin with the optimization model

$$\boldsymbol{x}^* \in \arg\min_{\boldsymbol{x} \in \mathcal{X}} \boldsymbol{\mu}^\top \boldsymbol{x}, \tag{1}$$

where $\mathcal{X} \subseteq \mathbb{R}^n$ is a known, feasible region, and $\boldsymbol{\mu} \in \mathbb{R}^n$ is an unknown, deterministic vector representing uncertain parameters. Throughout, we assume that we observe a random variable $\boldsymbol{Z} \in \mathbb{R}^n$ representing an estimate of $\boldsymbol{\mu}$ such that

$$\mathbb{E}\left[\boldsymbol{Z}\right] = \boldsymbol{\mu}, \quad \text{and} \quad \mathbb{E}\left[(Z_j - \mu_j)^2\right] = 1/\nu_j \quad j = 1, \ldots, n. \tag{2}$$

In words, Z_j is an unbiased estimator of μ_j with precision ν_j. (Recall precision is the reciprocal of variance.) We make no assumptions on the convexity or shape of $\mathcal{X}$; it may be a discrete set or involve non-linear, non-convex constraints. For convenience, we define $\nu_{\mathsf{min}} \equiv \min_j \nu_j$ and $\nu_{\mathsf{max}} \equiv \max_j \nu_j$.

Albeit stylized, Problem (1) subsumes network optimization applications with uncertain edge costs such as minimum spanning tree, shortest path, the traveling salesman, and matching on graphs (Bertsimas and Tsitsiklis, 1997). As noted in Elmachtoub and Grigas (2021), with a clever reformulation, Problem (1) also subsumes some inventory optimization applications with uncertain demands like the economic lot-sizing problem. Finally, through non-linear transformations (that may introduce non-convexities in $\mathcal{X}$), Problem (1) can also model certain multiproduct pricing problems and portfolio optimization problems (Gupta et al., 2021). In this sense, Problem (1) represents a general setting under which to study the small-data, large-scale regime.

Our use of the probabilistic model Eq. (2), however, deviates somewhat from the traditional operations literature. Equation (2) abstracts away from the data generation mechanism and instead focuses on the properties of the estimators Z_j built from that data. Importantly, this framework allows us to describe and analyze both the large-sample and small-data, large-scale regimes in a variety of data settings with a minimal amount of mathematical overhead.

Namely, instances of Problem (1) under Eq. (2) fall under the small-data, large-scale regime when n is very large relative to ν_{max} (a large number of uncertainties, but all estimates are imprecise). By contrast, such instances fall under the large-sample regime when n is small relative to ν_{min} (a fixed number of uncertainties, and all estimates are very precise). One can formalize these definitions by introducing an asymptotic sequence of instances of Problem (1) (see Gupta and Rusmevichientong (2021) for details), but the extra formalism offers little insight in what follows, and, hence, we prefer these loose descriptions.

We next provide some examples illustrating how these definitions of both regimes in terms of n, ν_{min} and ν_{max} provide a unified framework for analyzing several different data settings:

Independent, Identically Distributed (I.I.D) Data

Following Gupta and Rusmevichientong (2021), suppose that for each $j = 1, \ldots, n$, we observe $\{\xi_{j1}, \ldots, \xi_{j,N_j}\}$ i.i.d. draws of a random variable ξ_j with mean μ_j. A natural estimator for μ_j is the sample average $Z_j \equiv N_j^{-1} \sum_{k=1}^{N_j} \xi_{k,N_j}$, which is unbiased. Notice the precision of Z_j is proportional to N_j. Thus, our intuitive notion of large-sample asymptotics, i.e., $N_j \to \infty$ for all j, corresponds to $\nu_{\mathsf{min}} \to \infty$. By contrast, our intuitive notion of small-data, i.e. N_j small and fixed for all j, corresponds to ν_{max} small and fixed. Large-scale naturally corresponds to large n. In this way, both large-sample and small-data, large-scale regimes can be described entirely by the precisions and dimension n in Eq. (2) without explicitly modeling the i.i.d. sampling.

Weakly Stationary Time Series

Building on our previous example, suppose now that the sequence $(\xi_{j1}, \ldots, \xi_{j,N_j})$ are not i.i.d. for each j, but follow a weakly stationary time series. One can confirm that sample mean is still an unbiased estimate for μ_j, but its precision depends not only on N_j, but also the auto-covariance structure of the time series. In particular, for a highly-autocorrelated time series, information accumulates slowly, and N_j must be fairly large before one can learn μ_j precisely.

Fortunately, we can still discuss both regimes without explicitly specifying this covariance structure by again appealing to the precisions ν_{min} and ν_{max}. In the large sample setting, ν_{min} will be large relative to n, while in the small-data, large-scale setting ν_{max} will be small relative to n, and n will be large.

Regression Settings with Contextual Information

Finally, suppose that we observe independent observations $(\xi_j, \boldsymbol{W}_j)$ for $j = 1, \ldots, n$, where $\mathbb{E}[\xi_j] = \mu_j$ and $\boldsymbol{W}_j \in \mathbb{R}^p$ is a fixed covariate that is informative for the j^{th} uncertain parameter. For example, in logistics and routing applications, μ_j might represent the travel time on road j, and $\boldsymbol{W}_j$ might encode relevant information like the speed limit and length of road j. In such a setting, it is common to estimate μ_j by $Z_j \equiv \boldsymbol{W}_j^\top \boldsymbol{\beta}^{\mathsf{OLS}}, \quad j = 1, \ldots, n$, where

$$\boldsymbol{\beta}^{\mathsf{OLS}} \in \arg\min_{\boldsymbol{\beta}} \sum_{j=1}^{n} (\xi_j - \boldsymbol{W}_j^\top \boldsymbol{\beta})^2$$

is the ordinary least-squares fit, perhaps after transforming the covariates $\boldsymbol{W}_j$.

The behavior of these estimates depend subtly on the interplay between n, p, *and* the eigenspectrum of the matrix $\boldsymbol{W} = (\boldsymbol{W}_1^\top, \ldots, \boldsymbol{W}_n^\top)^\top \in \mathbb{R}^{n \times p}$. However, under the usual homoscedastic assumptions, the precision of Z_j is known to be proportional to

$$\nu_j \propto \left(\boldsymbol{W}_j^\top \left(\boldsymbol{W}^\top \boldsymbol{W} \right)^{-1} \boldsymbol{W}_j \right)^{-1}.$$

Hence, we can still describe the large-sample and small-data, large-scale regimes without explicitly without having to specify details about the structure of $\boldsymbol{W}$. Namely, this model is in the large-sample regime if ν_{min} is large relative to n, and is in the small-data, large-scale regime if n is large relative to ν_{max}.

Finally, we note that we have *not* assumed that $\boldsymbol{Z}$ is multivariate Gaussian, but in many of the estimation settings described above, one would expect intuitively that $\boldsymbol{Z}$ is approximately distributed as a multivariate Gaussian. Hence, we will often consider this special case to develop intuition.

We next use our above model to highlight a first important difference in these regimes.

2.2 Failure of Sample Average Approximation (SAA)

Sample average approximation (SAA), also called empirical risk minimization (ERM) in the machine learning literature, is arguably the most fundamental data-driven optimization procedure. Many other popular procedures including regularized ERM and distributionally robust optimization are, at least intuitively, motivated as refinements of SAA.

In our setting, the SAA procedure plugs in the estimator $\boldsymbol{Z}$ for the unknown $\boldsymbol{\mu}$ in Problem (1) and returns the resulting solution:

$$\boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z}) \in \arg\min_{\boldsymbol{x}\in\mathcal{X}} \boldsymbol{Z}^\top \boldsymbol{x}. \tag{3}$$

Under fairly mild assumptions, SAA has excellent performance in the large-sample regime. In our setting specifically, one can prove

Theorem 0.1 (SAA in Large-Sample Regime)

Consider an instance of Problem (1) under Eq. (2) *where* $\mathcal{X} \subseteq [0,1]^n$. *The expected sub-optimality of SAA relative to the full-information optimum satisfies*

$$0 \le \mathbb{E}\left[\boldsymbol{\mu}^\top \boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})\right] - \boldsymbol{\mu}^\top \boldsymbol{x}^* \le \frac{2n}{\sqrt{\nu_{\mathsf{min}}}}.$$

In particular, in large-sample settings when ν_{min} is large relative to n, SAA performs comparably to the full-information solution. For clarity, recall in the i.i.d. setting of our previous example, $\nu_{\mathsf{min}} \propto \min_j N_j$, and hence Theorem 0.1 shows expected performance of SAA converges to the full-information optimum at a the "usual" rate of $O(N_{\min}^{-1/2})$. The proof of Theorem 0.1 is quite standard and, hence, omitted.

Since $\nu_{\mathsf{max}} \ge \nu_{\mathsf{min}}$, the above bound is vacuous in the small-data, large-scale regime, i.e., when n is large relative to ν_{max}. This is not merely a weakness in analysis; SAA can have very poor performance in this regime, as seen in the following example:

Poor Performance of SAA in Small-Data, Large-Scale Regime

Consider an instance of Problem (1) under Eq. (2) where $Z_j \sim N(\mu_j, 1/\nu_j)$ is normally distributed,

$$(\mu_j, \nu_j) = \begin{cases} (0, 0.01) & \text{if } j \text{ is odd,} \\ (-1, 1) & \text{if } j \text{ is even,} \end{cases}$$

and

$$\mathcal{X} = \left\{ \boldsymbol{x} \in [0,1]^n : \sum_{j=1}^n x_j = .01n \right\}.$$

For convenience, assume $.01n$ is an integer. In words, the problem seeks to identify the worst 1% of the μ_j given the noisy estimates Z_j. The full information optimal value is $-.01n$ obtained by choosing any $.01n$ even components.

The SAA solution $x_j^{\mathsf{SAA}} = \mathbb{I}\{Z_j \le q_n\}$ where q_n is any solution to the equation

$$\frac{1}{n} \sum_{j=1}^n \mathbb{I}\{Z_j \le q\} = .01.$$

Write

$$\frac{1}{n} \sum_{j=1}^n \mathbb{I}\{Z_j \le q\} \;=\; \frac{1}{2} \cdot \frac{2}{n} \sum_{j:j \text{ odd}} \mathbb{I}\{Z_j \le q\} + \frac{1}{2} \cdot \frac{2}{n} \sum_{j:j \text{ even}} \mathbb{I}\{Z_j \le q\},$$

and note each sum consists of $n/2$ terms. Since the Z_j are i.i.d. for odd j, we have by the uniform law of large numbers that

$$\frac{2}{n} \sum_{j:j \text{ odd}} \mathbb{I}\{Z_j \le q\} \to_p \mathbb{P}(Z_1 \le q) = \Phi(q\sqrt{\nu_1}) = \Phi(.1q),$$

uniformly in q as $n \to \infty$, where $\Phi(\cdot)$ is the standard normal cumulative distribution function. Similarly, $\frac{2}{n} \sum_{j:j \text{ even}} \mathbb{I}\{Z_j \le q\} \to_p \mathbb{P}(Z_2 \le q) = \Phi(q+1)$ as $n \to \infty$. Hence, $q_n \to_p q^*$ where

$$\frac{1}{2}\Phi(.1q^*) + \frac{1}{2}\Phi(q^* + 1) = .01.$$

The value q^* can be determined numerically as $q^* \approx -20.54$. Then, an entirely analogous argument shows the scaled performance of SAA satisfies

$$\frac{1}{n}\boldsymbol{\mu}^\top \boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z}) \to_p \frac{1}{2} \cdot 0 \cdot \mathbb{P}\left(Z_1 \le q^*\right) + \frac{1}{2} \cdot 1 \cdot \mathbb{P}\left(Z_2 \le q^*\right).$$

Hence, the relative performance of SAA to the full-information optimum satisfies

$$\frac{\boldsymbol{\mu}^\top \boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})}{\boldsymbol{\mu}^\top \boldsymbol{x}^*} \to_p \frac{\mathbb{P}\left(Z_2 < q^*\right)}{-.02} < -10^{-83},$$

a negligibly small fraction.

Worse, had we simply chosen a feasible solution at random, our expected performance would be $-.005n$, yielding 50% relative performance to the full-information optimum. Thus, SAA performs substantively worse than random guessing in this example.

A clever reader might argue that the crux of the issue in the preceding example is that SAA does not leverage the precision information ν_j, and hence is “tricked” into selecting many of the odd components. A more clever algorithm that leveraged this information could avoid such a mistake.

Although this intuition is partially true, it is not the whole story. Indeed, Gupta and Rusmevichientong (2021) establishes the following theorem which shows that *no* data-driven algorithm exists which can achieve more than a fraction of the full-information performance in the small-data, large-scale regime. This behavior sharply contrasts Theorem 0.1.

Theorem 0.2 (Full-Information Optimum is Unattainable) *Given any data-driven algorithm $\boldsymbol{x}(\cdot)$ such that $\boldsymbol{x}(\boldsymbol{Z}) \in [0,1]^n$ almost surely, there exists an instance of Problem (1) with $\mathcal{X} = [0,1]^n$, and $\nu_j = 1$, $\mu_j \in \{-1, +1\}$ and $Z_j \sim \mathcal{N}(\mu_j, 1/\nu_j)$ for all j, such that*

$$\frac{\mathbb{E}\left[\boldsymbol{\mu}^\top \boldsymbol{x}(\boldsymbol{Z})\right]}{\boldsymbol{\mu}^\top \boldsymbol{x}^*} < .842.$$

The bound is not tight, but already highlights a distinct phenomena in the small-data, large-scale regime not present in the large-sample regime. No algorithm, even one with knowledge of the precisions, can expect to achieve a large fraction of full-information performance for all instances.

2.3 Best-in-Class Performance

Since full-information performance is not generally achievable, we instead establish a different benchmark to assess data-driven procedures. To this end, we next define a notion of “best-in-class” performance for a given policy class. For simplicity of exposition, we focus our discussion on plug-in policies:

Definition 0.1 (Plug-in Policy) Given functions $f_j : \mathbb{R} \mapsto \mathbb{R}$, we define the *plug-in policy* $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$ corresponding to $\boldsymbol{f}(\cdot) = (f_1(\cdot), \ldots, f_n(\cdot))^\top$ to be

$$\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) \in \arg\min_{\boldsymbol{x} \in \mathcal{X}} \boldsymbol{f}(\boldsymbol{Z})^\top \boldsymbol{x}, \tag{4}$$

where $\boldsymbol{f}(\boldsymbol{Z}) \in \mathbb{R}^n$ is the vector with j^{th} component $f_j(Z_j)$. Given a set of functions $\mathcal{F}$, we further define the corresponding class of plug-in polices to be $\left\{ \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) : \boldsymbol{f} \in \mathcal{F} \right\}$.

We stress that the component functions $f_j(Z_j)$ in the definition may differ by j or depend on auxiliary information.

Plug-in policies are computationally attractive because computing the policy for a fixed $\boldsymbol{f}(\cdot)$ requires solving an optimization problem of the same form as Problem (1). Thus, if there exists a specialized algorithm for solving Problem (1) – as is the case with many transportation, inventory management and pricing problems – the same algorithm can be used to evaluate $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$.

We next consider some examples:

Sample Average Approximation (SAA) as a Plug-In Policy

SAA is an example of a plug-in policy where $f_j(Z_j) = Z_j$.

Plug-Ins for Linear Classes

Consider our previous regression set-up where $\boldsymbol{W}_j$ encodes (known) covariate information for μ_j. A classical predict-then-optimize approach might first find the ordinary least-squares estimate $\boldsymbol{\beta}^{\mathsf{OLS}}$, and then solve Problem (1) after replacing μ_j by $\boldsymbol{W}_j^\top \boldsymbol{\beta}^{\mathsf{OLS}}$. We can alternatively view this approach as a plug-in policy corresponding to the functions $(Z_1 \mapsto \boldsymbol{W}_1^\top \boldsymbol{\beta}^{\mathsf{OLS}}, \ldots, Z_n \mapsto \boldsymbol{W}_n^\top \boldsymbol{\beta}^{\mathsf{OLS}})$.

This policy is in turn a special case of a plug-in policy corresponding to the linear function class

$$\mathcal{F}^{\mathsf{Linear}} = \{\boldsymbol{Z} \mapsto (\boldsymbol{W}_1^\top \boldsymbol{\beta}, \ldots, \boldsymbol{W}_n^\top \boldsymbol{\beta})^\top : \boldsymbol{\beta} \in \mathbb{R}^p\}.$$

In Elmachtoub and Grigas (2021), the authors argue that there exist plug-in policies in this larger class that significantly outperform the plug-in policy corresponding $\boldsymbol{\beta}^{\mathsf{OLS}}$.

Observe that members of $\mathcal{F}^{\mathsf{Linear}}$ are constant valued (they do not depend on $\boldsymbol{Z}$), and, hence, the corresponding plug-in policies also do not depend on $\boldsymbol{Z}$. We call such classes of plug-in policies *non-data-driven.* Non-data-driven policy classes are common in machine learning, but do not cover all examples of interest in data-driven optimization. For example, the SAA policy does depend on $\boldsymbol{Z}$ and hence does not belong to the non-data-driven plug-in policy class corresponding to $\mathcal{F}^{\mathsf{Linear}}$.

We next describe a data-driven plug-in policy class that does contain SAA as a member:

Plug-Ins Based on Mixed-Effects Regression

Define

$$\mathcal{F}^{\mathsf{ME}} = \left\{ \boldsymbol{Z} \mapsto \Big(\frac{\nu_1}{\nu_1+\tau} Z_1 + \frac{\tau}{\nu_1+\tau} \boldsymbol{W}_1^\top \boldsymbol{\beta}, \ \ldots, \frac{\nu_n}{\nu_n+\tau} Z_n + \frac{\tau}{\nu_n+\tau} \boldsymbol{W}_n^\top \boldsymbol{\beta} \Big)^\top \right.$$
$$\left. : \ \tau \in \mathbb{R}_+, \boldsymbol{\beta} \in \mathbb{R}^p \right\}.$$

In words, members of $\mathcal{F}^{\mathsf{ME}}$ proxy each μ_j as an interpolation between Z_j and a linear fit based on $\boldsymbol{\beta}$, where τ controls the degree of interpolation and the precision ν_j attenuates the effect. This form of interpolation arises naturally in a mixed-effects regression model of the unknown $\boldsymbol{\mu}$ where we assume $\boldsymbol{W}_j$ corresponds to some shared (fixed) effects and there is some unknown, random effect for each j. Moreover, the plug-in policy corresponding to $\tau = 0$ is $\boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})$ (c.f. Problem (3)), and the plug-in policies corresponding to $\tau = \infty$ exactly recover the plug-in policies corresponding to $\mathcal{F}^{\mathsf{Linear}}$. Thus, $\mathcal{F}^{\mathsf{ME}}$ strictly generalizes $\mathcal{F}^{\mathsf{Linear}}$.

Given any plug-in policy class, we define its "best" member, depending on the data $\boldsymbol{Z}$:

Definition 0.2 (Oracle Policy) Given a class $\mathcal{F}$ of functions, the *oracle plug-in policy* $\boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z})$ is defined by

$$\boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z}) = \boldsymbol{x}^{\boldsymbol{f}_{\mathsf{OR}}}(\boldsymbol{Z}) \text{ where } \boldsymbol{f}_{\mathsf{OR}} \in \operatorname*{arg\,min}_{\boldsymbol{f} \in \mathcal{F}} \boldsymbol{\mu}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}). \tag{5}$$

The oracle policy minimizes the true performance, similar to the full-information solution $\boldsymbol{x}^*$ (c.f. Problem (1)). However unlike $\boldsymbol{x}^*$, the oracle policy is restricted to use a member of the given class. We stress, the oracle policy is defined with respect to a particular realization of the data $\boldsymbol{Z}$, and is, thus, random.

By construction, no plug-in policy from $\mathcal{F}$ outperforms its oracle member. In this sense, the oracle policy is a strong benchmark. On the other hand, computing $\boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z})$ *seemingly* requires knowledge of $\boldsymbol{\mu}$, so it is not clear that we can identify a member of the given class with performance comparable to $\boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z})$ using only the data at hand. (We show later that this is indeed possible in certain cases.)

Importantly, oracle policies for well-chosen policy classes often enjoy favorable properties. For example, the element of $\mathcal{F}^{\mathsf{ME}}$ corresponding to parameters $(\tau, \boldsymbol{\beta})$ can be interpreted as the posterior mean estimate of $\boldsymbol{\mu}$ assuming the data are drawn from the following Bayesian model:

$$\begin{aligned} \mu_j &\sim \mathcal{N}(\boldsymbol{W}_j^\top \boldsymbol{\beta}, 1/\tau) \quad \text{independently across } j = 1, \ldots, n, \\ Z_j \mid \boldsymbol{\mu} &\sim \mathcal{N}(\boldsymbol{\mu}_j, 1/\nu_j) \quad \text{independently across } j = 1, \ldots, n. \end{aligned}$$

Consequently, the corresponding plug-in policy is the Bayes optimal policy for this model. A standard result in Bayesian statistics is that under very mild assumptions, Bayes polices are admissible, i.e., no other data-driven policy pareto-dominates their performance across all values of $\boldsymbol{\mu}$, whether or not the prior is correctly specified. Hence, since the oracle policy must perform at least as well as each element of the class, it too inherits this favorable property and is non-dominated.

In this sense, comparing the performance of a given data-driven algorithm to performance of an oracle policy from a suitable policy class is arguably a more natural approach than comparing to the (unachievable) full-information optimal performance. Indeed, much of the existing literature in small-data, large-scale optimization focuses on identifying policies with performance comparable to an oracle policy, i.e., near best-in-class performance, and we will do the same throughout the remainder.

2.4 Shortcomings of Cross-Validation

To summarize, we have reduced our study to the problem of identifying a policy with near best-in-class performance. A standard approach to such problems is cross-validation. In this section we show that the performance of cross-validation in the small-data, large-scale regime is complex; in general it might perform quite poorly, however, in some special cases it has provably good performance. These two distinct behaviors sharply contrast with the strong performance of cross-validation in the large-sample regime, highlighting yet another new phenomenon that emerges in the small-data, large-scale regime.

While there are many variants of cross-validation, we focus below on hold-out validation for simplicity. At a high-level, hold-out validation uses half the available data, i.e., *training data*, to train a policy, and then estimates the performance of that policy on the remaining half of the data, i.e., *hold-out data*. One typically then compares the performance of different policies on the hold-out data to select a member of a policy class. The hope is that this procedure identifies a policy with near best-in-class performance.

Since our general model Eq. (2) abstracts away from the data generation procedure, to model hold-out validation we will need some additional assumptions and notation. Our setup will mirror our previous example of "Independent, Identically Distributed (I.I.D.)" from Section 2.1.

Specifically, we assume that we observe

$$\{\xi_{j,1}, \ldots, \xi_{j,N_j}\} \text{ drawn i.i.d. such that } \mathbb{E}[\xi_{j,1}] = \mu_j, \quad j = 1, \ldots, n. \tag{6}$$

(For convenience, assume N_j is even for each j.) We then estimate μ_j by $Z_j \equiv \frac{1}{N_j} \sum_{k=1}^{N_j} \xi_{j,k}$. Our estimate of μ_j based on the training data is $Z_j^{\mathsf{train}} \equiv \frac{2}{N_j} \sum_{k \le N_j/2} \xi_{j,k}$. Similarly, our estimate of μ_j based on the hold-out set is $Z_j^{\mathsf{hold}} \equiv \frac{2}{N_j} \sum_{k > N_j/2} \xi_{j,k}$.

With this notation, given a class $\mathcal{F}$, policy selected by hold-out cross validation is

$$\boldsymbol{x}^{\mathsf{HO}}(\boldsymbol{Z}) = \boldsymbol{x}^{\boldsymbol{f}_{\mathsf{HO}}}(\boldsymbol{Z}) \quad \text{where} \quad \boldsymbol{f}_{\mathsf{HO}} \in \arg\min_{\boldsymbol{f} \in \mathcal{F}} \boldsymbol{Z}^{\mathsf{HO}^\top} \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}^{\mathsf{train}}). \tag{7}$$

Intuitively, the objective function of Problem (7) is meant to estimate $\boldsymbol{\mu}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$, i.e., the objective defining the oracle policy in Problem (5).

The next example adapted from Gupta et al. (2021) shows that in the small-data, large-scale regime, this procedure might provide a poor estimate of oracle performance for a fixed policy and, hence, might fail to identify the best-in-class policy.

Cross-Validation Can Perform Poorly

Consider an instance of Problem (1) under Eq. (2) in which $\mathcal{X} = [0,1]^n$. Suppose $N_j = 2$ for all j and

$$\xi_j \sim \begin{cases} \mathcal{N}(-1,1) & \text{if } j \le .14n \\ \mathcal{N}(1,1) & \text{otherwise.} \end{cases}$$

Thus, the precision of each Z_j is 2, and $Z_j^{\mathsf{train}} = \xi_{j1}$ while $Z_j^{\mathsf{hold}} = \xi_{j2}$. For convenience, assume $0.14n$ is an integer.

Finally, take $\mathcal{F} = \{\boldsymbol{Z} \mapsto \mathbf{1}, \boldsymbol{Z} \mapsto \boldsymbol{Z}\}$ to have only two members. The corresponding plug-in policies are i) the zero-policy which has all components equal to zero and ii) the SAA solution $\boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})$.

By inspection, the oracle performance of the zero-policy is 0. On the other hand, following an argument entirely analogous to our example in Section 2.2, one can see that as $n \to \infty$, the scaled, oracle performance of $\boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})$ converges to

$$\frac{1}{n} \boldsymbol{\mu}^\top \boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z}) \to_p -.14\Phi(\sqrt{2}) + .86\Phi(-\sqrt{2}) \approx -.0614.$$

Hence, an oracle would prefer SAA.

Next consider hold-out cross-validation. Cross-validation correctly estimates the performance of the zero-policy to be 0. On the other hand, the scaled cross-validation performance of SAA is

$$\frac{1}{n}\sum_{j=1}^{n} \xi_{j2}\mathbb{I}\left\{\xi_{j1} \le 0\right\} \to_p -.14\mathit{\Phi}(1) + .86\phi(-1) \approx .0186.$$

This is a very poor estimate of the oracle SAA performance. Moreover, hold-out cross-validation incorrectly suggests choosing the zero policy as best-in-class almost surely as $n \to \infty$.

In summary, hold-out cross-validation fails in two ways in the previous example: First, it provides a poor estimate of the SAA policy that remains poor even as $n \to \infty$. This shortcoming alone would not be enough to dismiss cross-validation as an inviable approach. Indeed, if cross-validation misestimated the performance of all policies by the same constant amount, it could still be used to identify a best-in-class policy. However, as seen above, cross-validation also fails in a second way; it misestimates differently for different policies, and hence picks a poor policy from the policy class.

As discussed in Gupta et al. (2021), the key issue behind the shortcoming of cross-validation in this setting is that the hold-out objective Problem (7) does not actually estimate the oracle objective $\boldsymbol{\mu}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$, but rather estimates the objective $\boldsymbol{\mu}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}^{\mathsf{train}})$. In the large-sample regime where precisions are high, $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$ and $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}^{\mathsf{train}})$ are reasonably close, so cross-validation is an effective strategy. However, in the small-data, large-scale regime where precisions are low, sacrificing half that precision when training the policy causes $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$ and $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}^{\mathsf{train}})$ to be quite different. Hence, cross-validation does not identify a near best-in-class policy.

That said, as mentioned, there are special cases where cross-validation does identify a best-in-class policy in the small-data, large-scale regime. Indeed, the above intuition suggests that for non-data-driven plug-in policy classes, e.g., the class induced by $\mathcal{F}^{\mathsf{Linear}}$, cross-validation might correctly identify a best-in-class policy since $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) = \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}^{\mathsf{train}})$ for all data realizations. This intuition is made formal in the following theorem:

Theorem 0.3 (Cross-Validation for Non-Data Driven Plug-in Classes)

Consider a non-data-driven plug-in policy class induced by the set of functions $\mathcal{F}$. Assume

i) $2 \le |\mathcal{F}| < \infty$,

ii) The data sets $\{\xi_{j,k} : k = 1, \ldots, N_j\}$ are independent across j, and

iii) $\xi_{j,k} - \mu_j$ is a subGaussian random variables with variance proxy at most σ^2 for all j and k.

Then, there exists an absolute constant C such that for any $0 < \epsilon < \frac{1}{2}$, and any instance of Problem (1) where $\mathcal{X} \subseteq [0,1]^n$, with probability at least $1 - \epsilon$

we have that

$$0 \leq \boldsymbol{\mu}^\top \boldsymbol{x}^{\mathsf{HO}}(\boldsymbol{Z}) - \boldsymbol{\mu}^\top \boldsymbol{x}^{OR}(\boldsymbol{Z}) \; \leq \; C\sigma\sqrt{n \log |\mathcal{F}| \log(1/\epsilon)}.$$

Proof Since the plug-in policies do not depend on $\boldsymbol{Z}$, we write $\boldsymbol{x}^{\boldsymbol{f}}$ instead of $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$. Similarly, we write $\boldsymbol{x}^{\mathsf{OR}}$ and $\boldsymbol{x}^{\mathsf{HO}}$.

The first inequality is immediate from the definition of $\boldsymbol{x}^{\mathsf{OR}}$. For the second, observe that

$$\begin{aligned}
\boldsymbol{\mu}^\top \boldsymbol{x}^{\mathsf{HO}} - \boldsymbol{\mu}^\top \boldsymbol{x}^{\mathsf{OR}} \; &= \; \left(\boldsymbol{\mu} - \boldsymbol{Z}^{\mathsf{HO}}\right)^\top \boldsymbol{x}^{\mathsf{HO}} + \boldsymbol{Z}^{\mathsf{HO}^\top} \left(\boldsymbol{x}^{\mathsf{HO}} - \boldsymbol{x}^{\mathsf{OR}}\right) + \left(\boldsymbol{Z}^{\mathsf{HO}} - \boldsymbol{\mu}\right)^\top \boldsymbol{x}^{\mathsf{OR}} \\
&\leq \; \left(\boldsymbol{\mu} - \boldsymbol{Z}^{\mathsf{HO}}\right)^\top \boldsymbol{x}^{\mathsf{HO}} + \left(\boldsymbol{Z}^{\mathsf{HO}} - \boldsymbol{\mu}\right)^\top \boldsymbol{x}^{\mathsf{OR}} \\
&\leq \; 2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| \left(\boldsymbol{Z}^{HO} - \boldsymbol{\mu}\right)^\top \boldsymbol{x}^{\boldsymbol{f}} \right|,
\end{aligned}$$

where the first inequality follows from the definition of $\boldsymbol{x}^{\mathsf{HO}}$. For a fixed $\boldsymbol{f}$, the random variable $\left(\boldsymbol{Z}^{HO} - \boldsymbol{\mu}\right)^\top \boldsymbol{x}^{\boldsymbol{f}}$ is mean zero and subGaussian. From our independence assumption, its variance proxy is at most

$$\sigma^2 \sum_{j=1}^{n} (x_j^{\boldsymbol{f}})^2 \; \leq \; \sigma^2 n,$$

since $\mathcal{X} \subseteq [0,1]^n$. Thus, our upper-bound is the supremum of at most $2\,|\mathcal{F}|$ mean-zero, subGaussian random variables. By Massart's Lemma (Wainwright, 2019, eq. 2.67), we can bound

$$\mathbb{E}\left[2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| \left(\boldsymbol{Z}^{HO} - \boldsymbol{\mu}\right)^\top \boldsymbol{x}^{\boldsymbol{f}} \right| \right] \; \leq \; 4\sigma\sqrt{n \log |\mathcal{F}|}.$$

To prove the stronger high-probability result claimed in the theorem, we need to show that the supremum concentrates at this value. To that end, (Pollard, 1990, Lemma 3.2) shows[1] that there exists an absolute constant C_1 such that

$$\mathbb{E}\left[\exp\left(\frac{2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| \left(\boldsymbol{Z}^{HO} - \boldsymbol{\mu}\right)^\top \boldsymbol{x}^{\boldsymbol{f}} \right|}{C_1 \sigma \sqrt{n \log |\mathcal{F}|}} \right)^2 \right] \; \leq 5.$$

Applying Markov's inequality and collecting constants then completes the proof. □

Theorem 0.3 asserts the sub-optimality of cross-validation scales like $O_p(\sqrt{n})$. In most settings of interest, the oracle performance $\boldsymbol{\mu}^\top \boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z})$ scales

[1] Pollard (1990) states this result in terms of the Ψ-Orlicz norm. Recall for any random variable Y, we define $\|Y\|_{\Psi} = \inf\{C > 0 : \mathbb{E}\left[\exp(Y^2/C^2)\right] \leq 5\}$. The Ψ-Orlicz norm is closely related to the subGaussian parameter of a random variable. See e.g. (Gupta and Rusmevichientong, 2021, Appendix A) or Rivasplata (2012).

like $O_p(n)$. Thus, Theorem 0.3 proves that in these settings the relative suboptimality of the policy chosen by cross-validation relative to the oracle policy is vanishing at a rate of $O_p(1/\sqrt{n})$. In this sense, cross-validation identifies a near-best-in-class policy asymptotically in the small-data, large-scale regime for non-data driven plug-in policy classes.

Most of the regularity conditions in Theorem 0.3 can be weakened. For example, by leveraging classical results for suprema of subGaussian processes, we can relax the finiteness of $\mathcal{F}$ to requiring that $\mathcal{F}$ have finite metric entropy. In this way, one can show that hold-out cross-validation does asymptotically select a best-in-class policy from $\mathcal{F}^{\mathsf{linear}}$ in the small-data, large-scale regime, provided the dimension of W_j is not too large.

Theorem 0.3 contrasts with the behavior in our previous example; cross-validation does not fail in either of the two aforementioned ways. The above proof bounds the error over *all* policies in the policy class simultaneously. Hence, with high probability, cross-validation asymptotically correctly estimates the performance of *every* policy in the policy class, and can identify a best-in-class policy asymptotically. This contrasting behavior when treating data-driven and non-data driven plug-in policies again highlights a subtlety of cross-validation in the small-data, large-scale regime not that is not present in the large-sample regime.

Finally, while somewhat beyond the scope of this chapter, we remark that Gupta and Kallus (2021) has shown additional new phenomenon for cross-validation in a slightly different setting. Loosely, they show that if we randomize the amount of data N_j for each component in a particular fashion, then cross-validation *does* allow us to identify a best-in-class policy for many data-driven policy classes in the small-data, large-scale regime with high-probability. However, even with this randomization, cross validation does *not* correctly estimate the oracle performance of any given policy in those classes; rather it uniformly misestimates by an unknown multiplicative constant. In this sense, randomizing the amount of data appears to be a “middle-ground” between our earlier counterexample and Theorem 0.3, addressing one of the failures of cross-validation but not the other.

Developing a complete theory of cross-validation in the small-data, large-scale regime remains an open question. In the next section, we pursue an entirely different avenue for policy selection in the small-data, large-scale regime.

3 Debiasing In-Sample Performance

Since the shortcomings of cross-validation stem from sacrificing part of the data when training and part of the data when evaluating the performance of a policy, one might consider instead selecting a policy by optimizing

$$\min_{\boldsymbol{f}\in\mathcal{F}} \boldsymbol{Z}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}), \tag{8}$$

so that all the data are used in both steps. Unfortunately, for most interesting policy classes, this strategy fails, due to the well-known in-sample bias or "over-fitting" problem. The next theorem illustrates the issue:

Theorem 0.4 (SAA Optimizes a Biased Objective) *Suppose there exists an $\boldsymbol{f}_{\mathsf{SAA}} \in \mathcal{F}$ such that $\boldsymbol{x}^{\boldsymbol{f}_{\mathsf{SAA}}}(\boldsymbol{Z}) = \boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})$. Then,*

$$\boldsymbol{f}_{\mathsf{SAA}} \in \arg\min_{\boldsymbol{f}\in\mathcal{F}} \boldsymbol{Z}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}).$$

Proof Write

$$\boldsymbol{Z}^\top \boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z}) \geq \min_{\boldsymbol{f}\in\mathcal{F}} \boldsymbol{Z}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) \ \geq \ \min_{\boldsymbol{x}\in\mathcal{X}} \boldsymbol{Z}^\top \boldsymbol{x} = \boldsymbol{Z}^\top \boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z}),$$

where the first inequality follows because $\boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z}) = \boldsymbol{x}^{\boldsymbol{f}_{\mathsf{SAA}}}(\boldsymbol{Z})$, the second inequality follows because $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) \in \mathcal{X}$ for all $\boldsymbol{f} \in \mathcal{F}$ by construction, and the last equality follows by definition of $\boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})$. Thus, we have equality throughout, proving the theorem. □

Consequently, for any sufficiently rich plug-in policy class, optimizing Problem (8) returns the SAA solution, which we have already seen can perform quite poorly in the small-data, large-scale regime.

Some reflection shows that at least part of the issue here is that $\boldsymbol{Z}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$ is a biased estimate of the oracle objective $\boldsymbol{\mu}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$ whenever $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$ depends on $\boldsymbol{Z}$ (i.e., for truly data-driven plug-in classes).

Hence, our approach to identifying a best-in-class policy will be to first debias this estimator.

3.1 Stein Correction

We leverage a classical result for Gaussian distributions attributed to Charles Stein and frequently called Stein's Lemma:

Lemma 0.1 (Stein's Lemma) *Suppose $Y \sim \mathcal{N}(\mu, \sigma^2)$. Then, for any function $g : \mathbb{R} \mapsto \mathbb{R}$ that is almost everywhere differentiable and for which both expectations are defined, we have*

$$\mathbb{E}\left[(Y-\mu)g(Y)\right] = \sigma^2 \mathbb{E}\left[g'(Y)\right].$$

Proof We first treat the case where $\mu = 0$ and $\sigma = 1$. Then, using integration by parts,

$$\mathbb{E}\left[Yg(Y)\right] = \frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} yg(y)e^{-y^2/2}dy = \frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} g'(y)e^{-y^2/2}dy = \mathbb{E}\left[f'(Y)\right],$$

proving the special case. For general (μ, σ), define the function $\overline{g}(t) = g(\mu + \sigma t)$, so that

$$\mathbb{E}\left[Y g(Y)\right] = \mathbb{E}\left[(\mu + \sigma\xi)\overline{g}(\xi)\right] = \mathbb{E}\left[\mu\overline{g}(\xi)\right] + \sigma\mathbb{E}\left[\xi\overline{g}(\xi)\right],$$

where $\xi \sim \mathcal{N}(0,1)$. Applying the lemma to the last expectation yields

$$\mathbb{E}\left[Y g(Y)\right] = \mathbb{E}\left[\mu\overline{g}(\xi)\right] + \sigma\mathbb{E}\left[\overline{g}'(\xi)\right] = \mathbb{E}\left[\mu g(Y)\right] + \sigma^2\mathbb{E}\left[g'(Y)\right].$$

Rearranging completes the proof. □

Stein's Lemma provides a tool to estimate the bias of $\boldsymbol{Z}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$ when $\boldsymbol{Z}$ is a multivariate Gaussian. Namely,

$$\mathbb{E}\left[(\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})\right] \;=\; \sum_{j=1}^{n} \mathbb{E}\left[(Z_j - \mu_j)\mathbb{E}\left[x_j^{\boldsymbol{f}}(\boldsymbol{Z}) \mid Z_j\right]\right].$$

Define the function $g_j(t) \;\equiv\; \mathbb{E}\left[x_j^{\boldsymbol{f}}(\boldsymbol{Z}) \mid Z_j = t\right]$. Then, applying Stein's Lemma to each element of the sum shows

$$\mathbb{E}\left[(\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})\right] \;=\; \sum_{j=1}^{n} \frac{1}{\nu_j}\mathbb{E}\left[g_j'(Z_j)\right].$$

Of course, the challenge is that we do not have a simple expression for $g'(Z_j)$. Instead, we approximate this derivative by a central finite step difference, i.e., we heuristically argue that for small h,

$$g_j'(Z_j) = \frac{g_j(Z_j + h) - g_j(Z_j - h)}{2h} + O(h^2).$$

Hence, we might expect that

$$\begin{aligned}
\mathbb{E}\left[(\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})\right] \;&=\; \sum_{j=1}^{n} \frac{\mathbb{E}\left[g_j(Z_j + h) - g_j(Z_j - h)\right]}{2h\nu_j} + O(nh^2) \\
&=\; \sum_{j=1}^{n} \frac{\mathbb{E}\left[x_j^{\boldsymbol{f}}(\boldsymbol{Z} + h\boldsymbol{e}_j) - x_j^{\boldsymbol{f}}(\boldsymbol{Z} - h\boldsymbol{e}_j)\right]}{2h\nu_j} + O(nh^2),
\end{aligned}$$

where $\boldsymbol{e}_j$ is the j^{th} coordinate vector.

Gupta and Rusmevichientong (2021) makes the above heuristic argument rigorous by dealing with potential points of non-differentiability and precisely quantifying the remainder. Indeed, they prove a slightly stronger theorem which applies when $\boldsymbol{Z}$ is possibly not multivariate Gaussian, but is well-approximated by a multivariate Gaussian. For simplicity of exposition, we summarize their result in the Gaussian case only:

Theorem 0.5 (Bias of the Stein Correction for Gaussian Estimates)

Suppose that for each $j = 1, \ldots, n$, we have that $Z_j \sim \mathcal{N}(\mu_j, 1/\nu_j)$, independently across j. Finally, let

$$B^{\boldsymbol{f}}(\boldsymbol{Z}, h) \equiv \sum_{j=1}^{n} \frac{x_j^{\boldsymbol{f}}(\boldsymbol{Z} + h\boldsymbol{e}_j) - x_j^{\boldsymbol{f}}(\boldsymbol{Z} - h\boldsymbol{e}_j)}{2h\nu_j}. \tag{9}$$

Then, for any $0 < h < \frac{1}{2}$, and any plug-in policy $\boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$, we have that

$$\left| \mathbb{E}\left[\boldsymbol{\mu}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})\right] - \mathbb{E}\left[\boldsymbol{Z}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})\right] + B^{\boldsymbol{f}}(\boldsymbol{Z}, h) \right| \leq 4h^2 n.$$

Theorem 0.5 asserts that by choosing h small enough, we can estimate the performance $\boldsymbol{\mu}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z})$ of a plug-in policy in an almost unbiased fashion by the bias-corrected quantity $\boldsymbol{Z}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) - B^{\boldsymbol{f}}(\boldsymbol{Z})$. At first glance, this analysis suggests choosing h arbitrarily small. As we will see, h controls a bias-variance tradeoff for our estimator; small h does induce small bias, but comes at the cost of large variance.

Given the central role of Stein's Lemma in its derivation, we term $B^{\boldsymbol{f}}(\boldsymbol{Z})$ the *Stein Correction*. Evaluating $B^{\boldsymbol{f}}(\boldsymbol{Z})$ from the data is straightforward but computationally cumbersome, since in principle we must compute $2n$ different plug-in policies corresponding to the $\pm h$ perturbations of the n components. Gupta and Rusmevichientong (2021) and Gupta et al. (2021) each discuss possible refinements that exploit either duality or the sensitivity analysis of the underlying Problem (1) to speed up the computation.

Finally, we remark that in the non-Gaussian case, Gupta and Rusmevichientong (2021) generalize the above result so that the error term contains an additional term that does not vanish as $h \to 0$ and depends on the degree to which $\boldsymbol{Z}$ is non-Gaussian.

3.2 From Unbiasedness to Policy Selection

Theorem 0.5 suggests the following procedure for identifying a near-best-in-class policy: Choose some small $h > 0$, then select

$$\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) = \boldsymbol{x}^{\boldsymbol{f}_{\mathsf{Stein}}}(\boldsymbol{Z}) \quad \text{where} \quad \boldsymbol{f}_{\mathsf{Stein}} \in \operatorname*{arg\,min}_{\boldsymbol{f} \in \mathcal{F}} \boldsymbol{Z}^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) - B^{\boldsymbol{f}}(\boldsymbol{Z}, h). \tag{10}$$

Unfortunately, Theorem 0.5 alone is not enough to ensure this procedure identifies a near best-in-class policy, even asymptotically in the small-data, large-scale regime. Namely, since Theorem 0.5 only treats the bias of our estimator, we need also to establish that certain random quantities concentrate at their expectations.

More specifically, let $\boldsymbol{f}_{\mathsf{Stein}}, \boldsymbol{f}_{\mathsf{OR}} \in \mathcal{F}$ be the functions such that $\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) = \boldsymbol{x}^{\boldsymbol{f}_{\mathsf{Stein}}}(\boldsymbol{Z})$ and $\boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z}) = \boldsymbol{x}^{\boldsymbol{f}_{\mathsf{OR}}}(\boldsymbol{Z})$. Then, write

$$\begin{aligned}
&\boldsymbol{\mu}^\top \left(\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - \boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z})\right) \\
&\quad = \; (\boldsymbol{\mu} - \boldsymbol{Z})^\top \boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) + B^{\boldsymbol{f}_{\mathsf{Stein}}}(\boldsymbol{Z}, h) \\
&\qquad + \boldsymbol{Z}^\top \boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - B^{\boldsymbol{f}_{\mathsf{Stein}}}(\boldsymbol{Z}, h) - \boldsymbol{Z}^\top \boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z}) + B^{\boldsymbol{f}_{\mathsf{OR}}}(\boldsymbol{Z}, h) \qquad (11)\\
&\qquad + (\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z}) - B^{\boldsymbol{f}_{\mathsf{OR}}}(\boldsymbol{Z}, h) \\
&\quad \le \; (\boldsymbol{\mu} - \boldsymbol{Z})^\top \boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) + B^{\boldsymbol{f}_{\mathsf{Stein}}}(\boldsymbol{Z}, h) + (\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z}) - B^{\boldsymbol{f}_{\mathsf{OR}}}(\boldsymbol{Z}, h),
\end{aligned}$$

where the inequality follows from the definition of $\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z})$ (c.f. Problem (10)). Rearranging and upper bounding by the worst-case in the policy class shows

$$\begin{aligned}
\boldsymbol{\mu}^\top \left(\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - \boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z})\right) &\le \; 2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| (\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) + B^{\boldsymbol{f}}(\boldsymbol{Z}, h) \right|. \\
&\le \; 2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| (\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) \right| + 2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| B^{\boldsymbol{f}}(\boldsymbol{Z}, h) \right| \\
&\le \; 2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| (\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) - \mathbb{E}\left[(\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) \right] \right| \\
&\quad + 2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| B^{\boldsymbol{f}}(\boldsymbol{Z}, h) - \mathbb{E}\left[B^{\boldsymbol{f}}(\boldsymbol{Z}, h) \right] \right| \\
&\quad + 2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| \mathbb{E}\left[(\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) - B^{\boldsymbol{f}}(\boldsymbol{Z}, h) \right] \right|.
\end{aligned}$$

Theorem 0.5 bounds the last term. Thus,

$$\begin{aligned}
&\underbrace{\boldsymbol{\mu}^\top \left(\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - \boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z})\right)}_{\text{Sub-Optimality of Our Procedure}} \\
&\qquad \le \; 2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| (\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) - \mathbb{E}\left[(\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^{\boldsymbol{f}}(\boldsymbol{Z}) \right] \right| \qquad (12a)\\
&\qquad\quad + 2 \sup_{\boldsymbol{f} \in \mathcal{F}} \left| B^{\boldsymbol{f}}(\boldsymbol{Z}, h) - \mathbb{E}\left[B^{\boldsymbol{f}}(\boldsymbol{Z}, h) \right] \right| \qquad (12b)\\
&\qquad\quad + 4h^2 n
\end{aligned}$$

To prove that $\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z})$ has near best-in-class performance, we must argue that the above two suprema are vanishingly small in the small-data, large-scale regime relative to the oracle performance.

When can we expect these suprema to be vanishingly small? To develop some intuition, we first study a special case in which Problem (1) decouples into n separate optimization problems.

Theorem 0.6 (Near Best-In-Class Performance for Decoupled Feasible Regions)

Consider an instance of Problem (1) under Eq. (2) *where the feasible region admits a factorization of the form $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$ for some sets $X_j \subseteq [0,1]$ for $j = 1, \ldots n$. Suppose further that $\boldsymbol{Z}$ is a multivariate Gaussian with independent components. Finally, consider a plug-in policy class induced by the function class $\mathcal{F}$ where $2 < |\mathcal{F}| < \infty$. Then, there exists a constant C not depending on h, n or $\mathcal{F}$ such that for any $0 < \epsilon < \frac{1}{2}$*

$$0 \leq \underbrace{\boldsymbol{\mu}^\top \left(\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - \boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z})\right)}_{\textit{Sub-Optimality of Our Procedure}} \leq C \log(1/\epsilon) \sqrt{\log |\mathcal{F}|} \cdot \frac{\sqrt{n}}{h} + Cnh^2.$$

In particular, if we let $h = O(n^{-1/6})$, then the sub-optimality of our procedure is $O_p(n^{2/3})$.

Recall that in most applications, we expect that $\boldsymbol{\mu}^\top \boldsymbol{x}^{\mathsf{OR}}(\boldsymbol{Z})$ itself will scale like $O_p(n)$. Hence, in these application, the lemma proves that the relative sub-optimality of our procedure is vanishing in the small-data, large-scale limit.

Proof Our approach will be to bound the two suprema in Eq. (12). We first write them explicitly

$$\textit{Eq.}\ (12\text{a}) = \sup_{\boldsymbol{f} \in \mathcal{F}} \left| \sum_{j=1}^n (Z_j - \mu_j) x_j^{\boldsymbol{f}}(\boldsymbol{Z}) - \mathbb{E}\left[(Z_j - \mu_j) x_j^{\boldsymbol{f}}(\boldsymbol{Z})\right] \right|,$$

$$\textit{Eq.}\ (12\text{b}) = \sup_{\boldsymbol{f} \in \mathcal{F}} \left| \sum_{j=1}^n \frac{x_j(\boldsymbol{Z} + h\boldsymbol{e}_j) - x_j(\boldsymbol{Z} - h\boldsymbol{e}_j) - \mathbb{E}\left[x_j(\boldsymbol{Z} + h\boldsymbol{e}_j) - x_j(\boldsymbol{Z} - h\boldsymbol{e}_j)\right]}{2h\nu_j} \right|.$$

The argument of each suprema is the sum of mean-zero random variables. Under our assumption on $\mathcal{X}$, the j^{th} component of the solution $x_j(\boldsymbol{Z})$ only depends on Z_j, but does not depend on Z_k for $k \neq j$. Thus, the terms of these sums are independent. This observation is crucial. Said differently, both suprema can be interpreted as suprema of an empirical process and hence analyzed with standard techniques (see, e.g., Pollard (1990) for a canonical reference).

To that end, we first bound the supremum in Eq. (12b). For a fixed $\boldsymbol{f}$, each term in the sum has magnitude at most $\frac{1}{h\nu_{\mathsf{min}}}$. Hence, each term is sub-Gaussian with variance proxy at most $\frac{1}{h\nu_{\mathsf{min}}}$. Since the terms are independent, the entire sum (for a fixed $\boldsymbol{f}$) is subGaussian with variance proxy at most $\frac{n}{h\nu_{\mathsf{min}}}$. Finally, since the suprema is over a finite set, we expect the supremum cannot grow too large. Indeed, by Massart's Lemma (Wainwright, 2019, Eq. (2.67)), we know that

$$\mathbb{E}\left[\sup_{\boldsymbol{f} \in \mathcal{F}} \left| B^{\boldsymbol{f}}(\boldsymbol{Z}, h) - \mathbb{E}\left[B^{\boldsymbol{f}}(\boldsymbol{Z}, h)\right] \right| \right] \leq 2\sqrt{\frac{n \log |\mathcal{F}|}{h\nu_{\mathsf{min}}}}.$$

To prove a stronger, high-probability bound, we invoke the discussion leading up to (Pollard, 1990, Eq. (7.4)). This discussion shows that there exists a constant C_1 such that with probability at least $1-\epsilon/2$, this supremum is at most $C_1 \log(1/\epsilon)\sqrt{\frac{n}{h}} \cdot \sqrt{\log |\mathcal{F}|}$. (See also Theorem A.1 of Gupta et al. (2021) for clarification.)

We now treat the supremum in Eq. (12a). Intuitively, the analysis is similar but it is more tedious to establish that each term of the sum is subGaussian. Instead we invoke a generic result from empirical process theory that encapsulates the relevant argument. Specifically, note that $\left|(Z_j-\mu_j)x_j^{\boldsymbol{f}}(\boldsymbol{Z})\right| \leq |Z_j-\mu_j|$. Hence the vector $|\boldsymbol{Z}-\boldsymbol{\mu}|$ with j^{th} component $|Z_j-\mu_j|$ is an envelope for the process. Moreover, by Lemma A.1, Part (iv) of Gupta and Rusmevichientong (2021), the Orlicz norm[2] $\|\| \, |\boldsymbol{Z}-\boldsymbol{\mu}| \, \|_2\|_\Psi$ is at most $\sqrt{\frac{2n}{\nu_{\mathsf{min}}}}$. Hence, by Theorem A.1 of Gupta et al. (2021), there exists a constant C_2 such that with probability at least $1-\epsilon/2$, the Eq. (12a) is at most $C_2 \log(1/\epsilon)\sqrt{n \log |\mathcal{F}|}$.

Combining both bounds and collecting constants proves the theorem. □

Theorem 0.6 already highlights the aforementioned tradeoff with h. As we let $h \to 0$, the error due to misestimating the bias vanishes, but the stochastic error stemming from Eq. (12b) blows up.

Using fairly standard machinery from empirical process theory, it is straightforward to generalize Theorem 0.6 to the setting where $|\mathcal{F}|$ is infinite, but $\mathcal{F}$ has finite metric entropy. We refer the interested reader to Pollard (1990). Similarly, our analysis of the two suprema above only required that the components Z_j were subGaussian and independent. Hence, by leveraging the more general form of Theorem 0.5 in Gupta and Rusmevichientong (2021), one can also easily generalize Theorem 0.6 to the case where $\boldsymbol{Z}$ is only approximately Gaussian.

Unfortunately, for more interesting optimization problems where $\mathcal{X}$ does not factorize, the proof of Theorem 0.6 breaks down. The issue is that even for a fixed $\boldsymbol{f}$, the terms of the sums composing the suprema are *not* independent because $x_j^{\boldsymbol{f}}(\boldsymbol{Z})$ potentially depends on the entire vector $\boldsymbol{Z}$. The nature of this dependence hinges on the structure of $\mathcal{X}$ in Problem (1) in a potentially complex way.

Nonetheless, Theorem 0.6 provides a blueprint for how one might analyze these cases. Namely,

i) Use the structure of $\mathcal{X}$ to argue that the terms $x_j^{\boldsymbol{f}}(\boldsymbol{Z}_j)$ are only "weakly-dependent" across j. More precisely, we must argue that the sums inside the suprema of Eq. (12) each concentrate at a rate $o_p(n)$ for a fixed $\boldsymbol{f} \in \mathcal{F}$.
ii) Use empirical process theory to bound each of the suprema with these weakly dependent sums in terms of the "size" of $\mathcal{F}$, i.e., either its cardinality $|\mathcal{F}|$ or its metric entropy.

[2] See footnote 1 for details on the Orlicz-norm.

Although not trivial, this blueprint underlies the more advanced results in Gupta and Rusmevichientong (2021). Indeed, therein the authors consider the special case where $\mathcal{X}$ is polyhedral of the special form $\{\boldsymbol{x} \in [0,1]^n : \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}n\}$, where $\boldsymbol{A} \in \mathbb{R}^{m \times n}$. When $m \ll n$, the authors use a duality argument to show that the relevant terms of the sum are not too dependent, and hence the above program goes through as described. For a different debiasing procedure, Gupta et al. (2021) also follows a similar blueprint for problems that suitably decouple after fixing a small number of decision variables or removing a small number of constraints. Summarizing these results is beyond the scope of this chapter.

3.3 Stein Correction in the Large-Sample Regime

Interestingly, although we motivated $\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z})$ by the need for debiasing in the small-data, large-scale regime, this policy has excellent performance in the large-sample regime, as well:

Theorem 0.7 (Stein Correction Achieves Full-Information in Large Sample Regime)

Consider an instance of Problem (1) under Eq. (2) *such that* $\mathcal{X} \subseteq [0,1]^n$*. Suppose there exists* $\boldsymbol{f}_{\mathsf{SAA}} \in \mathcal{F}$ *such that* $\boldsymbol{x}^{\boldsymbol{f}_{\mathsf{SAA}}}(\boldsymbol{Z}) = \boldsymbol{x}^{SAA}(\boldsymbol{Z})$*. Then,*

$$0 \;\leq\; \underbrace{\mathbb{E}\left[\boldsymbol{\mu}^\top(\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - \boldsymbol{x}^*)\right]}_{\textit{Expected Sub-Optimality to Full-Info.}} \;\leq\; \frac{1}{h\nu_{\mathsf{min}}} + \frac{2n}{\sqrt{\nu_{\mathsf{min}}}}.$$

The result should be compared to Theorem 0.1. Indeed, the Stein Correction adds at most $\frac{1}{h\nu_{\mathsf{min}}}$ to the expected error compared to SAA. Moreover, in the large-sample limit, $\nu_{\mathsf{min}} \to 0$, so this term is neglibly small compared to the SAA error. In other words, the Stein Correction enjoys performance comparable to the SAA performance in the large-sample regime.

Proof The first inequality follows from the definition of $\boldsymbol{x}^*$ in Problem (1). Let $\boldsymbol{f}_{\mathsf{Stein}} \in \mathcal{F}$ be the optimizer of Problem (10).

Then write

$$\begin{aligned}\boldsymbol{\mu}^\top(\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - \boldsymbol{x}^*) \;&=\; (\boldsymbol{\mu} - \boldsymbol{Z})^\top \boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) + \boldsymbol{Z}^\top(\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - \boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})) \\ &\quad + \boldsymbol{Z}^\top(\boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z}) - \boldsymbol{x}^*) + (\boldsymbol{Z} - \boldsymbol{\mu})^\top \boldsymbol{x}^*.\end{aligned}$$

By optimality of $\boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})$ in Problem (3), the third term above is non-positive. We can use the Cauchy-Schwarz inequality to upper bound the first and last term by $\|\boldsymbol{Z} - \boldsymbol{\mu}\|_1$ since $\boldsymbol{x}^*, \boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) \in \mathcal{X} \subseteq [0,1]^n$. Thus,

$$\begin{aligned}\boldsymbol{\mu}^\top(\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - \boldsymbol{x}^*) \;&\le\; 2\|\boldsymbol{Z} - \boldsymbol{\mu}\|_1 + \boldsymbol{Z}^\top(\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - \boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})).\\ &=\; 2\|\boldsymbol{Z} - \boldsymbol{\mu}\|_1 + B^{\boldsymbol{f}_{\mathsf{Stein}}}(\boldsymbol{Z}, h) - B^{\boldsymbol{f}_{\mathsf{SAA}}}(\boldsymbol{Z}, h)\\ &\quad +\; \boldsymbol{Z}^\top\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - B^{\boldsymbol{f}_{\mathsf{Stein}}}(\boldsymbol{Z}, h) - \boldsymbol{Z}^\top\boldsymbol{x}^{\mathsf{SAA}}(\boldsymbol{Z})) + B^{\boldsymbol{f}_{\mathsf{SAA}}}(\boldsymbol{Z}, h).\end{aligned}$$

By the optimality of $\boldsymbol{f}_{\mathsf{Stein}}$ in Problem (10), the last line of the last inequality is non-positive. Moreover, $\sup_{\boldsymbol{f} \in \mathcal{F}} \left| B^{\boldsymbol{f}}(\boldsymbol{Z}, h) \right| \le \frac{1}{2h\nu_{\mathsf{min}}}$ by construction. Combining shows

$$\boldsymbol{\mu}^\top(\boldsymbol{x}^{\mathsf{Stein}}(\boldsymbol{Z}) - \boldsymbol{x}^*) \;\le\; 2\|\boldsymbol{Z} - \boldsymbol{\mu}\|_1 + \frac{1}{h\nu_{\mathsf{min}}}.$$

To complete the proof, take expectations of both sides and observe that by Jensen's inequality,

$$\mathbb{E}\left[\|\boldsymbol{Z} - \boldsymbol{\mu}\|_1\right] = \sum_{j=1}^n \mathbb{E}\left[|Z_j - \mu_j|\right] \;\le \sum_{j=1}^n \sqrt{\mathbb{E}\left[(Z_j - \mu_j)^2\right]} = \sum_{j=1}^n \frac{1}{\sqrt{\nu_j}} \;\le \frac{n}{\sqrt{\nu_{\mathsf{min}}}}.$$

Substituting above completes the proof. □

Theorem 0.7 is a heartening result! It shows that it is possible to design algorithms with provably good performance in both large-sample and small-data, large-scale regimes.

3.4 Open Questions

The debiasing approach to optimization in the small-data, large-scale regime still nascent. At time of writing there are a number of exciting open questions. For what kinds of optimization problems might we expect that the components of the solution $x_j^{\boldsymbol{f}}(\boldsymbol{Z})$ are only weakly-dependent? Is this weak-dependence strictly necessary in order to construct provably good procedures, or is it an artifact of our analysis?

From a computational perspective, how should we efficiently solve Problem (10)? In general, this problem is discontinuous and non-convex. If the space of functions $\mathcal{F}$ is fairly complex, simple enumeration may not be feasible. How then should we identify good policies?

More generally, are there better debiasing schemes than the Stein Correction? Gupta et al. (2021) considers the special case of affine plug-in policies and provides an alternate debiasing scheme that explicitly leverages optimization structure via Danskin's theorem. What are the benefits and drawbacks of these various schemes? Might we design even better schemes for particular, specialized optimization problems in inventory or revenue management? What other approaches beyond debiasing exist to attack problems in this new setting?

4 Conclusion

As the degree of personalization and customization increases in operations management and operations research applications, the ubiquity of the small-data, large-scale regime will only increase. Our goal in this chapter was to highlight some new phenomena that emerge in this regime, and to argue that these new phenomena can dramatically affect our intuition about and the performance of data-driven optimization algorithms for these applications. While developing a comprehensive theory for this regime remains outstanding, we hope that our initial steps will further motivate researchers to develop customized algorithms for these new, exciting applications that explicitly leverage these phenomena.

References

Bertsimas D, Tsitsiklis JN (1997) Introduction to Linear Optimization, vol 6. Athena Scientific Belmont, MA

Dwork C (2008) Differential privacy: A survey of results. In: International Conference on Theory and Applications of Models of Computation, Springer, pp 1–19

Elmachtoub AN, Grigas P (2021) Smart "predict, then optimize". Management Science

Gupta V, Kallus N (2021) Data pooling in stochastic optimization. Management Science

Gupta V, Rusmevichientong P (2021) Small-data, large-scale linear optimization with uncertain objectives. Management Science 67(1):220–241

Gupta V, Han BR, Kim SH, Paek H (2020) Maximizing intervention effectiveness. Management Science 66(12):5576–5598

Gupta V, Huang M, Rusmevichientong P (2021) Debiasing in-sample policy performance for small-data, large-scale optimization. arXiv URL https://arxiv.org/abs/2107.12438

Harper FM, Konstan JA (2015) The movielens datasets: History and context. ACM Transactions on Interactive Intelligent Systems (TIIS) 5(4):1–19

Liu Y, Li Z (2017) A novel algorithm of low sampling rate gps trajectories on map-matching. EURASIP Journal on Wireless Communications and Networking 2017(1):1–5

Pollard D (1990) Empirical processes: Theory and applications. In: NSF-CBMS Regional Conference Series in Probability and Statistics, JSTOR, pp i–86

Rivasplata O (2012) Subgaussian random variables: An expository note. DOI 10.13140/RG.2.2.36288.23040, URL http://www.stat.cmu.edu/ arinaldo/36788/subgaussians.pdf

Wainwright MJ (2019) High-Dimensional Statistics: A Non-Asymptotic Viewpoint, vol 48. Cambridge University Press