Near-Optimal Design of Experiments via Regret Minimization

Zeyuan Allen-Zhu [*1] **Yuanzhi Li** [*2] **Aarti Singh** [*3] **Yining Wang** [*3]

Abstract

We consider computationally tractable methods for the experimental design problem, where k out of n design points of dimension p are selected so that certain optimality criteria are approximately satisfied. Our algorithm finds a $(1+\varepsilon)$-approximate optimal design when k is a *linear* function of p; in contrast, existing results require k to be super-linear in p. Our algorithm also handles all popular optimality criteria, while existing ones only handle one or two such criteria. Numerical results on synthetic and real-world design problems verify the practical effectiveness of the proposed algorithm.

1. Introduction

Experimental design is an important problem in statistics and machine learning research (Pukelsheim, 2006). Consider a linear regression model

$$\boldsymbol{y} = \mathbf{X}\boldsymbol{\beta}_0 + \boldsymbol{w}, \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{n\times p}$ is a pool of n design points, $\boldsymbol{y}$ is the response vector, $\boldsymbol{\beta}_0$ is a p-dimensional unknown regression model and $\boldsymbol{w}$ is a vector of i.i.d. noise variables satisfying $\mathbb{E}w_i = 0$ and $\mathbb{E}w_i^2 < \infty$. The experimental design problem is to select a small subset of rows (i.e., design points) $\mathbf{X}_S$ from the design pool $\mathbf{X}$ so that the statistical power of estimating $\boldsymbol{\beta}_0$ is maximized from noisy response $\boldsymbol{y}_S$ on the selected designs $\mathbf{X}_S$.

As an example, consider a material synthesis application where p is the number of variables (e.g., temperature, pressure, duration) that are hypothesized to affect the quality of the synthesized material and n is the total number of combinations of different parameters of experimental conditions. As experiments are expensive and time-consuming, one wishes to select $k \ll n$ experimental settings from $\mathbf{X}$ that are the most statistically efficient for establishing a model that connects experimental parameters with synthesized material quality, $\boldsymbol{y}$. The experimental design problem is also related to many machine learning tasks, such as linear bandits (Deshpande & Montanari, 2012; Huang et al., 2016), diversity sampling (Kulesza & Taskar, 2012) and active learning (Ma et al., 2013; Chaudhuri et al., 2015; Hazan & Karnin, 2015; Balcan & Long, 2013; Wang & Singh, 2016).

Since statistical efficiency can be measured in various ways, there exist a number of *optimality criteria* to guide the selection of experiments. We review some optimality criteria in Sec. 2 and interested readers are referred to Sec. 6 of (Pukelsheim, 2006) for a comprehensive review.

Typically, an optimality criterion is a function $f : \mathbb{S}_p^+ \to \mathbb{R}$ that maps from the p-dimensional positive definite cone to a real number. The experimental design problem can then be formulated as a combinatorial optimization problem:

$$S^*(k) = \operatorname*{arg\,min}_{S\in\mathcal{S}(n,k)} f(\mathbf{X}_S^\top \mathbf{X}_S), \tag{2}$$

where S is either a set or a multi-set of size k, and $\mathbf{X}_S \in \mathbb{R}^{k\times p}$ is formed by stacking the rows of $\mathbf{X}$ that are in S. The constraint set $\mathcal{S}_{1/2}(n,k)$ is defined as follows:

1. **With replacement**: $\mathcal{S}_1(n,k) = \{S \text{ multi-set} : S \subseteq [n], |S| \le k\}$. Under this setting, $\mathbf{X}_S$ may contain duplicate rows of the design pool $\mathbf{X}$;
2. **Without replacement**: $\mathcal{S}_2(n,k) = \{S \text{ standard set} : S \subseteq [n], |S| \le k\}$. Under this setting, $\mathbf{X}_S$ only contains distinct rows of the design pool $\mathbf{X}$.

The "with replacement" setting is classical in statistics literature, where the multiple measurements in $\boldsymbol{y}$ with respect to the same design point lead to different values with statistically independent noise. The "without replacement" setting, on the other hand, is more relevant in machine learning applications, because labels are not likely to change if the same data point (e.g., the same image) is considered twice. Finally, it is worth pointing out that the "with replacement" setting is *easier*, because it can be reduced (in polynomial time) to the "without replacement" setting by replicating each row of $\mathbf{X}$ for k times.

For many popular choices of f, the exact optimization

[*]Author names listed in alphabetic order. [1]Microsoft Research, Redmond, USA [2]Princeton University, Princeton, USA [3]Carnegie Mellon University, Pittsburgh, USA. Correspondence to: Yining Wang <yiningwa@cs.cmu.edu>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

Table 1. Comparison with existing results on computationally efficient experimental design. An algorithm produces a subset $\hat{S}$ of size k, and the approximation ratio is defined as $f(\mathbf{X}_{\hat{S}}^\top \mathbf{X}_{\hat{S}})/\min_{S\in\mathcal{S}_b(n,r)} f(\mathbf{X}_S^\top \mathbf{X}_S)$ for $r \le k$. T_{OPT} denotes the time complexity for solving the continuous convex optimization problem in Eq. (6).

	Without replacement?	Deterministic?	Time complexity	Constraints†	Criteria	Approx. ratio
Pipage rounding	Yes	Yes	$T_{\mathrm{OPT}} + O(n^2p^2)$	$k = r \ge p$	D,T	$(1-1/e)^{-1}$
(Bouhtou et al., 2010)	Yes	No	$T_{\mathrm{OPT}} + O(n)$	$k = r \ge p$	D	$(n/k)^{1/p}$
(Avron & Boutsidis, 2013)	Yes	Yes	$O(n^2p^2)$	$k = r \ge p$	A	$\frac{n-p+1}{k-p+1}$
(Wang et al., 2016)	No	No	$T_{\mathrm{OPT}} + O(nk)$	$k = \Omega(r)$, and $r = \Omega((p\log p)/\varepsilon^2)$	A,V	$1+\varepsilon$
(Wang et al., 2016)	Yes	Yes	$T_{\mathrm{OPT}} + O(p^6)$	$k = r = \Omega(p^2/\varepsilon)$	A,V	$1+\varepsilon$
This paper — Eq. (3)	No	Yes	$T_{\mathrm{OPT}} + \tilde{O}(nkp^2)$	$k = r = \Omega(p/\varepsilon^2)$	A,D,T,E,V,G	$1+\varepsilon$
This paper — Eq. (4)	Yes	Yes	$T_{\mathrm{OPT}} + \tilde{O}(nkp^2)$	$k = r > 2p$	A,D,T,E,V,G	$O(1)$
This paper — Eq. (5)	Yes	Yes	$T_{\mathrm{OPT}} + \tilde{O}(nkp^2)$	$k = \Omega(r), r \ge p/\varepsilon^2$	A,D,T,E,V,G	$1+\varepsilon$

† In $\tilde{O}(\cdot)$ and $\tilde{\Omega}(\cdot)$ we hide logarithmic dependency over n, p and k.

problem in Eq. (2) is NP-hard (Çivril & Magdon-Ismail, 2009; Černỳ & Hladík, 2012). In this paper, we propose a computationally tractable algorithm that approximately computes Eq. (2) for a wide range of optimality criteria, and under very weak conditions on n, k and p.

Below is our main theorem:

Theorem 1.1. *Suppose $b \in \{1,2\}$, $n > k > p$ and let $f : \mathbb{S}_p^+ \to \mathbb{R}$ be a* regular *optimality criterion (cf. Definition 2.1). There exists a polynomial-time algorithm that outputs $\hat{S} \in \mathcal{S}_b(n,k)$ for any input matrix $\mathbf{X} \in \mathbb{R}^{n\times p}$ with full column rank, and $\hat{S}$ satisfies the following:*

1. *For $b = 1$ (with replacement), there exists an absolute constant $C_0 \le 32$ such that, for any $\varepsilon \in (0,1)$, if $k \ge C_0 p/\varepsilon^2$ then*

$$f(\mathbf{X}_{\hat{S}}^\top \mathbf{X}_{\hat{S}}) \le (1+\varepsilon)\cdot \min_{S\in\mathcal{S}_1(n,k)} f(\mathbf{X}_S^\top \mathbf{X}_S)\ . \quad (3)$$

2. *For $b = 2$ (without replacement) and any $\xi > 2$, there exists constant $C_1(\xi) > 0$ depending only on ξ such that, if $k \ge \xi p$ then*

$$f(\mathbf{X}_{\hat{S}}^\top \mathbf{X}_{\hat{S}}) \le C_1(\xi)\cdot \min_{S\in\mathcal{S}_2(n,k)} f(\mathbf{X}_S^\top \mathbf{X}_S)\ . \quad (4)$$

   Moreover, for $\xi \ge 4$ we have $C_1(\xi) \le 32$.

3. *For $b = 2$ (without replacement) and any $\varepsilon \in (0, 1/2)$, if k, r satisfy $k \ge 4(1+7\varepsilon)r$ and $r \ge p/\varepsilon^2$, then*

$$f(\mathbf{X}_{\hat{S}}^\top \mathbf{X}_{\hat{S}}) \le (1+\varepsilon)\cdot \min_{S\in\mathcal{S}_2(n,r)} f(\mathbf{X}_S^\top \mathbf{X}_S). \quad (5)$$

We interpret the significance of Theorem 1.1 as follows.

- Under a very mild condition of $k > 2p$, our polynomial-time algorithm finds a set $\hat{S} \subset [n]$ of size k, with objective value $f(\mathbf{X}_{\hat{S}}^\top \mathbf{X}_{\hat{S}})$ being at most $O(1)$ a constant times the optimum. See Eq. (4).
- If replacement ($b = 1$) or over-sampling ($k > r$) is allowed, the approximation ratio can be tightened to $1+\varepsilon$ for arbitrarily small $\varepsilon > 0$. See Eq. (3) and (5).
- In all of the three cases, we only require k to grow linearly in p. Recall that $k \ge p$ is necessary to ensure the singularity of $\mathbf{X}_{\hat{S}}^\top \mathbf{X}_{\hat{S}}$. In contrast, no polynomial-time algorithm has achieved $O(1)$ approximation in the regime $k = O(p)$ for non-submodular optimality criteria (e.g., A- and V-optimality) under the without replacement setting.
- Our algorithm works for any regular optimality criterion. To the best of our knowledge, no known polynomial-time algorithm can achieve a $(1+\varepsilon)$ approximation for the D- and T-optimality criteria, or even an $O(1)$ approximation for the E- and G-optimality criteria. See Table 1 for a comparison.

The key idea behind our proof of Theorem 1.1 is a regret minimization characterization of the least eigenvalue of positive semidefinite (PSD) matrices. Similar ideas were developed in (Allen-Zhu et al., 2015; Silva et al., 2016) to construct efficient algorithms for linear-sized graph sparsifiers. In this paper we adopt the regret minimization framework and present novel potential function analysis for the specific application of experimental design.

1.1. Notations

$\mathbb{S}_p^+$ is the positive definite cone of $p \times p$ matrices: a $p \times p$ symmetric matrix $\mathbf{A}$ belongs to $\mathbb{S}_p^+$ if and only if $\boldsymbol{v}^\top \mathbf{A}\boldsymbol{v} > 0$ for all $\boldsymbol{v} \in \mathbb{R}^p \backslash \{\mathbf{0}\}$. For symmetric matrices $\mathbf{A}$ and $\mathbf{B}$, we write $\mathbf{A} \preceq \mathbf{B}$ if $\boldsymbol{v}^\top(\mathbf{A}-\mathbf{B})\boldsymbol{v} \ge 0$ for all $\boldsymbol{v} \in \mathbb{R}^p$. The inner product $\langle \mathbf{A}, \mathbf{B}\rangle$ is defined as $\langle \mathbf{A}, \mathbf{B}\rangle = \mathrm{tr}(\mathbf{B}^\top \mathbf{A}) = \sum_{i,j=1}^p \mathbf{A}_{ij}\mathbf{B}_{ij}$. We use $\|\mathbf{A}\|_2 = \sup_{\boldsymbol{v}\in\mathbb{R}^p\backslash\{\mathbf{0}\}} \|\mathbf{A}\boldsymbol{v}\|_2/\|\boldsymbol{v}\|_2$ to denote the spectral norm, and $\|\mathbf{A}\|_F = \sqrt{\sum_{i,j=1}^p \mathbf{A}_{ij}^2} = \sqrt{\langle \mathbf{A}, \mathbf{A}\rangle}$ to denote

the Frobenius norm of $\mathbf{A}$. For $\mathbf{A} \succeq \mathbf{0}$, we write $\mathbf{B} = \mathbf{A}^{1/2}$ as the unique $\mathbf{B} \succeq \mathbf{0}$ that satisfies $\mathbf{B}^2 = \mathbf{A}$. For a design pool $\mathbf{X} \in \mathbb{R}^{n\times p}$, we use $\boldsymbol{x}_i \in \mathbb{R}^p$ to denote the i-th row of $\mathbf{X}$. We use $\sigma_{\min}(\mathbf{A})$ for the least (smallest) singular value of a PSD matrix $\mathbf{A}$.

1.2. Related work

Experimental design is an old topic in statistics research (Pukelsheim, 2006; Fedorov, 1972). Computationally efficient experimental design algorithms (with provable guarantees) are, however, a less studied field. In the case of *submodular* optimality criteria (e.g., D- and T-optimality), the classical *pipage rounding* method (Ageev & Sviridenko, 2004; Horel et al., 2014; Ravi et al., 2016) combined with semi-definite programming results in computationally efficient algorithms that enjoy a constant approximation ratio. Bouhtou et al. (2010) improves the approximation ratio when k is very close to n. Deshpande & Rademacher (2010); Li et al. (2017) considered polynomial-time algorithms for sampling from a D-optimality criterion. These algorithms are not applicable to non-submodular criteria, such as A-, V-, E- or G-optimality.

For the particular A-optimality criterion, (Avron & Boutsidis, 2013) proposed a greedy algorithm with an approximation ratio of $O(n/k)$ with respect to $f(\mathbf{X}^\top\mathbf{X})$. It was shown that in the worst case $\min_{|S|\le k} f(\mathbf{X}_S^\top\mathbf{X}_S) \approx O(n/k)\cdot f(\mathbf{X}^\top\mathbf{X})$ and hence the bound is tight. However, for general design pool $\min_{|S|\le k} f(\mathbf{X}_S^\top\mathbf{X}_S)$ could be far smaller than $O(n/k)\cdot f(\mathbf{X}^\top\mathbf{X})$, making the theoretical results powerless in such scenarios. Wang et al. (2016) considered a variant of the greedy method and showed an approximation ratio quadratic in design dimension p and independent of pool size n.

Wang et al. (2016) derived algorithms based on effective resistance sampling (Spielman & Srivastava, 2011) that attain $(1+\varepsilon)$ approximation ratio if $k = \Omega(p\log p/\varepsilon^2)$ and repetitions of design points are allowed. The algorithm fundamentally relies on the capability of "re-weighting" (repeating) design points and cannot be adapted to the more general "without replacement" setting. Naive sampling based methods were considered in (Wang et al., 2016; Chaudhuri et al., 2015; Dhillon et al., 2013), which also achieve $(1+\varepsilon)$ approximation but requires the subset size k to be much larger than the condition number of $\mathbf{X}$.

A related however different topic is low-rank matrix column subset selection and CUR approximation, which seeks column subset $\mathbf{C}$ and row subset $\mathbf{R}$ such that $\|\mathbf{X} - \mathbf{C}\mathbf{C}^\dagger\mathbf{X}\|_F$ and/or $\|\mathbf{X} - \mathbf{C}\mathbf{U}\mathbf{R}\|_F$ are minimized (Drineas et al., 2008; Boutsidis & Woodruff, 2014; Wang & Singh, 2015b; Drineas & Mahoney, 2005; Wang & Zhang, 2013; Wang & Singh, 2015a). These problems are unsupervised in nature and do not in general correspond to statistical properties under supervised regression settings. Pilanci & Wainwright (2016); Raskutti & Mahoney (2014); Woodruff (2014) considered fast methods for solving ordinary least squares (OLS) problems. They are computationally oriented and typically require knowledge of the full response vector $\boldsymbol{y}$, which is different from the experimental design problem.

2. Regular criteria and continuous relaxation

We start with the definition of *regular* optimality criteria:

Definition 2.1 (Regular criteria)**.** *An optimality criterion $f : \mathbb{S}_p^+ \to \mathbb{R}$ is* regular *if it satisfies the followig properties:*

1. Convexity*:* [1] *$f(\lambda\mathbf{A} + (1-\lambda)\mathbf{B}) \le \lambda f(\mathbf{A}) + (1-\lambda) f(\mathbf{B})$ for all $\lambda \in [0,1]$ and $\mathbf{A}, \mathbf{B} \in \mathbb{S}_p^+$;*
2. Monotonicity*: If $\mathbf{A} \preceq \mathbf{B}$ then $f(\mathbf{A}) \ge f(\mathbf{B})$;*
3. Reciprocal multiplicity*: $f(t\mathbf{A}) = t^{-1} f(\mathbf{A})$ for all $t > 0$ and $\mathbf{A} \in \mathbb{S}_p^+$.*

Almost all optimality criteria used in the experimental design literature are regular. Below we list a few popular examples; their statistical implications can be found in (Pukelsheim, 2006):

- A-optimality (**Average**): $f_A(\mathbf{\Sigma}) = \frac{1}{p}\mathrm{tr}(\mathbf{\Sigma}^{-1})$;
- D-optimality (**Determinant**): $f_D(\mathbf{\Sigma}) = (\det|\mathbf{\Sigma}|)^{-\frac{1}{p}}$;
- T-optimality (**Trace**): $f_T(\mathbf{\Sigma}) = p/\mathrm{tr}(\mathbf{\Sigma})$;
- E-optimality (**Eigenvalue**): $f_E(\mathbf{\Sigma}) = \|\mathbf{\Sigma}^{-1}\|_2$;
- V-optimality (**Variance**): $f_V(\mathbf{\Sigma}) = \frac{1}{n}\mathrm{tr}(\mathbf{X}\mathbf{\Sigma}^{-1}\mathbf{X}^\top)$;
- G-optimality: $f_G(\mathbf{\Sigma}) = \max \mathrm{diag}(\mathbf{X}\mathbf{\Sigma}^{-1}\mathbf{X}^\top)$.

The (A-, D-, T-, E-) criteria concern estimates of regression coefficients and the (V-, G-) criteria are about in-sample predictions. All criteria listed above are regular. Note that for D-optimality the proxy function $g_D(\mathbf{\Sigma}) = -\log\det(\mathbf{\Sigma})$ is considered to satisfy the convexity property. In addition, by the standard arithmetic inequality we have that $f_T \le f_D \le f_A \le f_E$ and that $f_V \le f_G$.

Although exact optimization of the combinatorial problem Eq. (2) is intractable, it is nevertheless easy to solve a *continuous relaxation* of Eq. (2) given the convexity property in Definition 2.1. We consider the following continuous optimization problem:

$$\boldsymbol{\pi}^*(b) = \operatorname*{arg\,min}_{\boldsymbol{\pi}=(\pi_1,\cdots,\pi_n)} \quad f\left(\sum_{i=1}^n \pi_i \boldsymbol{x}_i\boldsymbol{x}_i^\top\right), \qquad (6)$$
$$s.t. \quad \boldsymbol{\pi} \ge \mathbf{0}, \quad \|\boldsymbol{\pi}\|_1 \le r, \quad \mathbb{I}[b=2]\cdot\|\boldsymbol{\pi}\|_\infty \le 1.$$

[1] This property could be relaxed to allow a proxy function $g : \mathbb{S}_p^+ \to \mathbb{R}$ being convex, where $g(\mathbf{A}) \le g(\mathbf{B}) \Leftrightarrow f(\mathbf{A}) \le f(\mathbf{B})$.

The $\|\boldsymbol{\pi}\|_1 \le r$ constraint makes sure only r rows of $\mathbf{X}$ are "selected", where $r \le k$ is a parameter that controls the degree of oversampling. The $0 \le \pi_i \le 1$ constraint enforces that each row of $\mathbf{X}$ is "selected" at most once and is only applicable to the without replacement setting ($b = 2$). Eq. (6) is a *relaxation* of the original combinatorial problem Eq. (2), which we formalize below:

Fact 2.1. *For $b \in \{1, 2\}$ we have $f(\sum_{i=1}^n \pi_i^*(b)\boldsymbol{x}_i\boldsymbol{x}_i^\top) \le \min_{S \in \mathcal{S}_b(n,r)} f(\mathbf{X}_S^\top \mathbf{X}_S)$*

In addition, because of the monotonicity property of f the sum constraint must bind:

Fact 2.2. *For $b \in \{1, 2\}$ it holds that $\sum_{i=1}^n \pi_i^*(b) = r$.*

Proofs of Facts 2.1 and 2.2 are straightforward and are placed in the supplementary material.

Both the objective function and the constraint set in Eq. (6) are convex, and hence it can be efficiently solved to global optimality by conventional convex optimization algorithms. In particular, for differentiable f we suggest the following projected gradient descent (PGD) procedure:

$$\boldsymbol{\pi}^{(t+1)} = \mathcal{P}_C\left(\boldsymbol{\pi}^{(t)} - \gamma_t \nabla f(\boldsymbol{\pi}^{(t)})\right), \tag{7}$$

where $\mathcal{P}_C(\boldsymbol{x}) = \arg\min_{\boldsymbol{y} \in C} \|\boldsymbol{x} - \boldsymbol{y}\|_2$ is the projection operator onto the feasible set $C = \{\boldsymbol{\pi} \in \mathbb{R}^p : \boldsymbol{\pi} \ge \mathbf{0}, \|\boldsymbol{\pi}\|_1 \le r, \mathbb{I}[b = 2] \cdot \|\boldsymbol{\pi}\|_\infty \le 1\}$ and $\{\gamma_t\}_{t \ge 1} > 0$ is a sequence of step sizes typically chosen by backtracking line search. When f is not differentiable everywhere, projected subgradient descent could be used with either constant or diminishing step sizes. We defer detailed gradient computations to the supplementary material. It was shown in (Wang et al., 2016; Su et al., 2012) that the projection operator $\mathcal{P}_C(\boldsymbol{x})$ could be efficiently computed up to precision δ in $O(n \log(\|\boldsymbol{x}\|_\infty/\delta))$ operations.

3. Sparsification via regret minimization

The optimal solution $\boldsymbol{\pi}^*$ of Eq. (6) does *not* naturally lead to a valid approximation of the combinatorial problem in Eq. (2), because the number of non-zero components in $\boldsymbol{\pi}^*$ may far exceed k. The primary focus of this section is to design efficient algorithms that *sparsify* the optimal solution $\boldsymbol{\pi}^*$ into $\boldsymbol{s} \in [k]^n$ (with replacement) or $\boldsymbol{s} \in \{0, 1\}^n$ (without replacement), while at the same time bounding the increase in the objective.

Due to the monotonicity and reciprocal multiplicity properties of f, it suffices to find a sparsifier $\boldsymbol{s}$ that satisfies

$$\left(\sum_{i=1}^n s_i \boldsymbol{x}_i \boldsymbol{x}_i^\top\right) \succeq \tau \cdot \left(\sum_{i=1}^n \pi_i^* \boldsymbol{x}_i \boldsymbol{x}_i^\top\right) \tag{8}$$

for some constant $\tau \in (0, 1)$. By Definition 2.1, Eq. (8) immediately implies $f(\sum_{i=1}^n s_i \boldsymbol{x}_i \boldsymbol{x}_i^\top) \le \tau^{-1} f(\sum_{i=1}^n \pi_i^* \boldsymbol{x}_i \boldsymbol{x}_i^\top)$. The key idea behind our algorithm is a regret-minimization interpretation of the least eigenvalue of a positive definite matrix, which arises from recent progress in the spectral graph sparsification literature (Silva et al., 2016; Allen-Zhu et al., 2015).

In the rest of this section, we adopt the notation that $\boldsymbol{\Pi} = \mathrm{diag}(\boldsymbol{\pi}^*)$ and $\mathbf{S} = \mathrm{diag}(\boldsymbol{s})$, both being $n \times n$ non-negative diagonal matrices. We also use $\mathbf{I}$ to denote the identity matrix, whose dimension should be clear from the context.

3.1. The whitening trick

Consider the linear transform $\boldsymbol{x}_i \mapsto (\mathbf{X}\boldsymbol{\Pi}\mathbf{X}^\top)^{-1/2}\boldsymbol{x}_i =: \tilde{\boldsymbol{x}}_i$. It is easy to verify that $\sum_{i=1}^n \pi_i^* \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_i^\top = \mathbf{I}$. Such a transform is usually referred to as *whitening*, because the sample covariance of the transformed data is the identity matrix. Define $\mathbf{W} = \sum_{i=1}^n s_i \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_i^\top$. We then have the following:

Proposition 3.1. *For $\tau > 0$, $\mathbf{W} \succeq \tau\mathbf{I}$ if and only if $(\sum_{i=1}^n s_i \boldsymbol{x}_i \boldsymbol{x}_i^\top) \succeq \tau(\sum_{i=1}^n \pi_i^* \boldsymbol{x}_i \boldsymbol{x}_i^\top)$.*

Proof. The proposition holds because $\mathbf{W} \succeq \tau\mathbf{I}$ if and only if $(\mathbf{X}\boldsymbol{\Pi}\mathbf{X}^\top)^{1/2}\mathbf{W}(\mathbf{X}\boldsymbol{\Pi}\mathbf{X}^\top)^{1/2} \succeq \tau\mathbf{X}\boldsymbol{\Pi}\mathbf{X}^\top$, and that $(\mathbf{X}\boldsymbol{\Pi}\mathbf{X}^\top)^{1/2}\mathbf{W}(\mathbf{X}\boldsymbol{\Pi}\mathbf{X}^\top)^{1/2} = \mathbf{X}\mathbf{S}\mathbf{X}^\top$. □

Proposition 3.1 shows that, without loss of generality, we may assume $\sum_{i=1}^n \pi_i^* \boldsymbol{x}_i \boldsymbol{x}_i^\top = \mathbf{X}\boldsymbol{\Pi}\mathbf{X}^\top = \mathbf{I}$. The question of proving $\mathbf{W} = \mathbf{X}\mathbf{S}\mathbf{X}^\top \succeq \tau\mathbf{I}$ is then reduced to lower bounding the smallest eigenvalue of $\mathbf{W}$.

Recall that $\mathbf{W}$ can be written as a sum of rank-1 PSD matrices $\mathbf{W} = \sum_{t=1}^k \mathbf{F}_t$, where $\mathbf{F}_t = \boldsymbol{x}_i \boldsymbol{x}_i^\top$ for some $i \in [n]$. In the next section we give a novel characterization of the least eigenvalue of $\mathbf{W}$ from a regret minimization perspective. The problem of lower bounding the least eigenvalue of $\mathbf{W}$ can then be reduced to bounding the regret of a particular Follow-The-Regularized-Leader (FTRL) algorithm, which is a much easier task as FTRL admits closed-form solutions.

3.2. Smallest eigenvalue as regret minimization

We first review the concept of regret minimization in a classical linear bandit setting. Let $\Delta_p = \{\mathbf{A} \in \mathbb{R}^{p \times p} : \mathbf{A} \succeq \mathbf{0}, \mathrm{tr}(\mathbf{A}) = 1\}$ be an *action space* that consists of positive semi-definite matrices of dimension p and unit trace norm. Consider the linear bandit problem, which operates in k iterations. At iteration t, the player chooses an action $\mathbf{A}_t \in \Delta_p$; afterwards, a "reference" action $\mathbf{F}_t \succeq \mathbf{0}$ is observed and the loss $\langle \mathbf{F}_t, \mathbf{A}_t \rangle$ is incurred. The objective of the player is to minimize his/her *regret*:

$$R(\{\mathbf{A}_t\}_{t=1}^k) := \sum_{t=1}^k \langle \mathbf{F}_t, \mathbf{A}_t \rangle - \inf_{\mathbf{U} \in \Delta_p} \sum_{t=1}^k \langle \mathbf{F}_t, \mathbf{U} \rangle,$$

which is the "excess loss" of $\{\mathbf{A}_t\}_{t=1}^k$ compared to the single optimal action $\mathbf{U} \in \Delta_p$ in hindsight, knowing all the reference actions $\{\mathbf{F}_t\}_{t=1}^k$. A popular algorithm for regret minimization is *Follow-The-Regularized-Leader (FTRL)*, also known to be equivalent to *Mirror Descent (MD)* (McMahan, 2011), which solves for

$$\mathbf{A}_t = \arg\min_{\mathbf{A}\in\Delta_p} \left\{ w(\mathbf{A}) + \alpha \cdot \sum_{\ell=1}^{t-1} \langle \mathbf{F}_\ell, \mathbf{A}\rangle \right\}. \tag{9}$$

Here $w(\mathbf{A})$ is a regularization term and $\alpha > 0$ is a parameter that balances model fitting and regularization. For the proof of our purpose we adopt the $\ell_{1/2}$-regularizer $w(\mathbf{A}) = -2\mathrm{tr}(\mathbf{A}^{1/2})$ introduced in (Allen-Zhu et al., 2015), which leads to the closed-form solution

$$\mathbf{A}_t = \left(c_t \mathbf{I} + \alpha \sum_{\ell=1}^{t-1} \mathbf{F}_\ell \right)^{-2}, \tag{10}$$

where $c_t \in \mathbb{R}$ is the unique constant that ensures $\mathbf{A}_t \in \Delta_p$. The following lemma from (Allen-Zhu et al., 2015) bounds the regret of FTRL using the particular $\ell_{1/2}$-regularizer:

Lemma 3.1 (Theorem 3.2 of (Allen-Zhu et al., 2015), specialized to $\ell_{1/2}$-regularization)**.** *Suppose $\alpha > 0$, $\mathrm{rank}(\mathbf{F}_t) = 1$ and let $\{\mathbf{A}_t\}_{t=1}^k$ be FTRL solutions defined in Eq. (10). If $\alpha\langle \mathbf{F}_t, \mathbf{A}_t^{1/2}\rangle > -1$ for all t, then*

$$\begin{aligned} R(\{\mathbf{A}_t\}_{t=1}^k) &:= \sum_{t=1}^k \langle \mathbf{F}_t, \mathbf{A}_t\rangle - \inf_{\mathbf{U}\in\Delta_p} \sum_{t=1}^k \langle \mathbf{F}_t, \mathbf{U}\rangle \\ &\le \alpha \sum_{t=1}^k \frac{\langle \mathbf{F}_t, \mathbf{A}_t\rangle \langle \mathbf{F}_t, \mathbf{A}_t^{1/2}\rangle}{1+\alpha\langle \mathbf{F}_t, \mathbf{A}_t^{1/2}\rangle} + \frac{2\sqrt{p}}{\alpha}. \end{aligned}$$

Now consider each $\mathbf{F}_t = \boldsymbol{x}_{i_t}\boldsymbol{x}_{i_t}^\top$ to be the outer product of a design point selected from the design pool $\mathbf{X}$. One remarkable consequence of Lemma 3.1 is that, in order to lower bound the smallest eigenvalue of $\sum_{t=1}^k \mathbf{F}_t$, which by definition is $\inf_{\mathbf{U}\in\Delta_p}\langle \sum_{t=1}^k \mathbf{F}_t, \mathbf{U}\rangle$, it suffices to lower bound $\sum_{t=1}^k \langle \mathbf{F}_t, \mathbf{A}_t\rangle$. Because $\mathbf{A}_t$ admits closed-form expression in Eq. (10), choosing a sequence of $\{\mathbf{F}_t\}_{t=1}^k$ with large $\sum_{t=1}^k \langle \mathbf{F}_t, \mathbf{A}_t\rangle$ becomes a much more manageable analytical task, which we shall formalize in the next section.

3.3. Proof of Theorem 1.1

Re-organizing terms in Lemma 3.1 we obtain

$$\inf_{\mathbf{U}\in\Delta_p} \sum_{t=1}^k \langle \mathbf{F}_t, \mathbf{U}\rangle \ge \sum_{t=1}^k \frac{\langle \mathbf{F}_t, \mathbf{A}_t\rangle}{1+\alpha\langle \mathbf{F}_t, \mathbf{A}_t^{1/2}\rangle} - \frac{2\sqrt{p}}{\alpha}. \tag{11}$$

The k near-optimal design points are selected in a sequential manner. Let $\Lambda_t \in \mathcal{S}_b(n,t)$ be the set of selected design points at or prior to iteration t ($\Lambda_0 = \emptyset$), and define $\mathbf{F}_t = \boldsymbol{x}_{i_t}\boldsymbol{x}_{i_t}^\top$, where i_t is the design point selected at iteration t. Define also $\mathbf{\Lambda}_t = \sum_{\ell=1}^t \mathbf{F}_\ell = \sum_{i\in\Lambda_t} \boldsymbol{x}_i\boldsymbol{x}_i^\top$.

We first consider the with replacement setting $b = 1$.

Lemma 3.2. *Suppose $\sum_{i=1}^n \pi_i^* \boldsymbol{x}_i\boldsymbol{x}_i^\top = \mathbf{I}$ where $\pi_i^* \ge 0$ and $\sum_{i=1}^n \pi_i^* = r$. Then for $1 \le t \le k$ we have that $\max_{i\in[n]} \frac{\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle}{1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle} \ge \frac{1}{r+\alpha\sqrt{p}}$.*

Proof. Recall that $\mathrm{tr}(\mathbf{A}_t) = 1$ and $\sum_{i=1}^n \pi_i^* \boldsymbol{x}_i\boldsymbol{x}_i^\top = \mathbf{I}$. Subsequently, $\sum_{i=1}^n \pi_i^* \langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle = 1$. On the other hand, we have that $\sum_{i=1}^n \pi_i^*(1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle) = \sum_{i=1}^n \pi_i^* + \alpha\cdot\mathrm{tr}(\mathbf{A}_t^{1/2}) \overset{(a)}{\le} r + \alpha\cdot\mathrm{tr}(\mathbf{A}_t^{1/2}) \overset{(b)}{\le} r + \alpha\sqrt{p}$. Here (a) is due to the optimization constraint that $\|\boldsymbol{\pi}^*\|_1 \le r$, and (b) is because $\mathrm{tr}(\mathbf{A}_t^{1/2}) = \|\boldsymbol{\sigma}(\mathbf{A}_t^{1/2})\|_1 \le \sqrt{p}\|\boldsymbol{\sigma}(\mathbf{A}_t^{1/2})\|_2 = \sqrt{p}\sqrt{\|\boldsymbol{\sigma}(\mathbf{A}_t)\|_1} = \sqrt{p}\sqrt{\mathrm{tr}(\mathbf{A}_t)} = \sqrt{p}$, where $\boldsymbol{\sigma}(\cdot)$ is the vector of all eigenvalues of a PSD matrix. Combining both inequalities we have that $\max_{i\in[n]} \frac{\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle}{1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle} \ge \frac{\sum_{i=1}^n \pi_i^*\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle}{\sum_{i=1}^n \pi_i^*(1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle)}$, where the right-hand side is lower bounded by $1/(r+\alpha\sqrt{p})$. □

Let $i_t = \arg\max_{i\in[n]} \frac{\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle}{1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle}$ be the design point selected at iteration t. Combining Eq. (11) and Lemma 3.2,

$$\mathbf{\Lambda}_k = \sum_{i\in\Lambda_k} \boldsymbol{x}_i\boldsymbol{x}_i^\top \succeq \left(\frac{k}{r+\alpha\sqrt{p}} - \frac{2\sqrt{p}}{\alpha} \right) \mathbf{I}. \tag{12}$$

To prove Eq. (3), set $\alpha = 8\sqrt{p}/\varepsilon$. Because $k = r \ge C_0 p/\varepsilon^2$, we have that $\frac{k}{r+\alpha\sqrt{p}} - \frac{2\sqrt{p}}{\alpha} \ge \frac{1}{1+8\varepsilon/C_0} - \frac{\varepsilon}{4}$. With $C_0 = 32$ the right-hand side is lower bounded by $1-\varepsilon/2$. Eq. (3) is thus proved because $(1-\varepsilon/2)^{-1} \le 1+\varepsilon$.

We next consider the without replacement setting $b = 2$.

Lemma 3.3. *Fix arbitrary $\beta \in (0,1]$ and suppose $\sum_{i=1}^n \pi_i^* \boldsymbol{x}_i\boldsymbol{x}_i^\top = \mathbf{I}$ where $\pi_i^* \in [0,\beta]$ and $\sum_{i=1}^n \pi_i^* = r$. Then for all $1 \le t \le k$,*

$$\max_{i\notin\Lambda_{t-1}} \frac{\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle}{1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle} \ge \frac{1-\beta\sigma_{\min}(\mathbf{\Lambda}_{t-1}) - \sqrt{p}/\alpha}{r+\alpha\sqrt{p}}.$$

Proof. On one hand, we have $\sum_{i\notin\Lambda_{t-1}} \pi_i^* \langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle \overset{(a)}{\ge} \langle \mathbf{A}_t, \mathbf{I} - \beta\mathbf{\Lambda}_{t-1}\rangle \overset{(b)}{=} 1 - \mathrm{tr}\left[(\alpha\mathbf{\Lambda}_{t-1} + c_t\mathbf{I})^{-2} \beta\mathbf{\Lambda}_{t-1} \right] = 1 + \frac{\beta c_t}{\alpha} - \frac{\beta}{\alpha}\mathrm{tr}\left[(\alpha\mathbf{\Lambda}_{t-1} + c_t\mathbf{I})^{-1} \right] = 1 + \frac{\beta c_t}{\alpha} - \frac{\mathrm{tr}(\mathbf{A}_t^{1/2})}{\alpha} \overset{(c)}{\ge} 1 + \frac{\beta c_t}{\alpha} - \frac{\sqrt{p}}{\alpha}$. Here (a) is due to $\sum_{i=1}^n \pi_i^* \boldsymbol{x}_i\boldsymbol{x}_i^\top = \mathbf{I}$ and $\pi_i^* \in [0,\beta]$; (b) is due to $\langle \mathbf{A}_t, \mathbf{I}\rangle = \mathrm{tr}(\mathbf{A}_t) = 1$ and (c) is proved in the proof of Lemma 3.2. Because $\alpha\mathbf{\Lambda}_{t-1} + c_t\mathbf{I} \succ \mathbf{0}$, we conclude that $c_t \ge -\alpha\sigma_{\min}(\mathbf{\Lambda}_{t-1})$ and therefore $\sum_{i\notin\Lambda_{t-1}} \pi_i^* \langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle \ge 1 - \beta\sigma_{\min}(\mathbf{\Lambda}_{t-1}) - \sqrt{p}/\alpha$. On the other hand, $\sum_{i\notin\Lambda_{t-1}} \pi_i^*(1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle) \le$

$r + \alpha\sqrt{p}$ by the same argument as in the proof of Lemma 3.2. Subsequently, $\max_{i\notin\Lambda_{t-1}} \frac{\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle}{1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle} \geq \frac{\sum_{i\notin\Lambda_{t-1}} \pi_i^* \langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle}{\sum_{i\notin\Lambda_{t-1}} \pi_i^*(1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle)} \geq \frac{1-\beta\sigma_{\min}(\mathbf{\Lambda}_{t-1})-\sqrt{p}/\alpha}{r+\alpha\sqrt{p}}$. □

Let $i_t = \arg\max_{i\notin\Lambda_{t-1}} \frac{\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle}{1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle}$. Combining Eq. (11) and Lemma 3.3 with $\beta = 1$, we have that

$$\mathbf{\Lambda}_k \succeq \left(\sum_{t=1}^{k} \frac{1-\kappa_t-\sqrt{p}/\alpha}{r+\alpha\sqrt{p}} - \frac{2\sqrt{p}}{\alpha}\right)\mathbf{I}, \qquad (13)$$

where $\kappa_t := \sigma_{\min}(\mathbf{\Lambda}_t)$. We are now ready to prove Eqs. (4,5) in Theorem 1.1.

Proof of Eq. (4). Note that

$$\mathbf{\Lambda}_k \succeq \sup_{u>0} \min\left\{u, \frac{1-u-\sqrt{p}/\alpha}{r+\alpha\sqrt{p}}\cdot k - \frac{2\sqrt{p}}{\alpha}\right\}\mathbf{I}. \qquad (14)$$

Eq. (14) can be proved by a case analysis: if $u \leq \kappa_t$ for some $1 \leq t \leq k$ then $\sigma_{\min}(\mathbf{\Lambda}_k) \geq \sigma_{\min}(\mathbf{\Lambda}_{t-1}) \geq u$; otherwise $1-\kappa_t-\sqrt{p}/\alpha \geq 1-u-\sqrt{p}/\alpha$ for all $1 \leq t \leq k$. Suppose $k = r \geq \xi p$ for some $\xi > 2$. and let $\alpha = \nu\sqrt{p}$, $u = \frac{(1-2/\xi)\nu-3}{\nu(2+\nu/\xi)}$, where $\nu > 1$ is some parameter to be specified later. Eq. (14) then yields $\mathbf{\Lambda}_k \succeq \frac{(1-2/\xi)\nu-3}{\nu(2+\nu/\xi)}\mathbf{I}$. Because $\xi > 2$, it is possible to select $\nu > 0$ such that $C_1(\xi)^{-1} = \frac{(1-2/\xi)\nu-3}{\nu(2+\nu/\xi)} > 0$. Finally, for $\xi \geq 4$ and $\nu = 8$ we have $C_1(\xi)^{-1} \geq 1/32$. Eq. (4) is thus proved. □

Proof of Eq. (5). Let $\beta \in (0,1)$ be a parameter to be specified later, and define $\mathbf{\Sigma}_\beta^* := \sum_{\pi_i^*\geq\beta} \pi_i^* \boldsymbol{x}_i\boldsymbol{x}_i^\top$ and $\bar{\mathbf{\Sigma}}_\beta^* := \mathbf{I} - \mathbf{\Sigma}_\beta^* = \sum_{\pi_i^*<\beta} \pi_i^* \boldsymbol{x}_i\boldsymbol{x}_i^\top$. Let $\hat{S}$ be constructed such that it includes *all* points in $S_\beta^* := \{i : \pi_i^* \geq \beta\}$, plus the resulting set by running Algorithm 1 on the remaining weights smaller than β, with subset size $k - k' = k - |S_\beta^*|$. Define $\alpha = 2\sqrt{p}/\varepsilon$, $r' := \sum_{\pi_i^*\geq\beta} \pi_i^*$, $\tilde{k} := k - k'$ and $\tilde{r} := r - r' + \alpha\sqrt{p} = r - r' + 2p/\varepsilon$. Let $\mathbf{\Lambda} = \sum_{i\in\hat{S}} \boldsymbol{x}_i\boldsymbol{x}_i^\top$ be the sample covariance of the selected subset. By the definition of $\hat{S}$ and Lemma 3.3, together with the whitening trick (Sec. 3.1) on $\bar{\mathbf{\Sigma}}_\beta^*$, we have

$$\begin{aligned}\mathbf{\Lambda} &\succeq \mathbf{\Sigma}_\beta^* + \sup_{u>0} \min\left\{u, (1-\beta u-\varepsilon/2)\tilde{k}/\tilde{r} - \varepsilon\right\}\bar{\mathbf{\Sigma}}_\beta^* \\ &\succeq \sup_{u>0} \min\left\{u, (1-\beta u-\varepsilon/2)\tilde{k}/\tilde{r} - \varepsilon\right\}\mathbf{I},\end{aligned}$$

where the second line holds because $\mathbf{\Sigma}_\beta^* + \bar{\mathbf{\Sigma}}_\beta^* = \mathbf{I}$ and $u \leq 1$. Now set $\beta = 0.5$ and note that $k' \leq r'/\beta \leq 2r'$ by definition of S_β^*. Subsequently, $r \geq p/\varepsilon^2$ and $k \geq 4(1+7\varepsilon)r$ for $\varepsilon \in (0, 1/2)$ implies that $\frac{\tilde{k}}{\tilde{r}} \geq \frac{1+2\varepsilon}{(1-\varepsilon/2)(1-\beta)}$, which yields $u \geq 1-\varepsilon/2$ and hence $f(\mathbf{X}_{\hat{S}}^\top\mathbf{X}_{\hat{S}}) \leq (1+\varepsilon)f(\mathbf{X}_{S^*}^\top\mathbf{X}_{S^*})$. Eq. (5) is thus proved. □

Algorithm 1 Near-optimal experimental design

Input: design pool $\mathbf{X} \in \mathbb{R}^{n\times p}$, budget parameters $k \geq r \geq p$, algorithmic parameter $\alpha > 0$.
Solve the convex optimization problem Eq. (6) with parameter s; Let $\boldsymbol{\pi}^*$ be the optimal solution;
Whitening: $\mathbf{X} \leftarrow \mathbf{X}(\mathbf{X}^\top\mathrm{diag}(\boldsymbol{\pi}^*)\mathbf{X})^{-1/2}$;
Initialization: $\Lambda_0 = \emptyset$;
for $t = 1$ to k **do**
  $c_t \leftarrow$ FINDCONSTANT$(\sum_{i\in\Lambda_{t-1}} \boldsymbol{x}_i\boldsymbol{x}_i^\top, \alpha)$;
  $\mathbf{A}_t \leftarrow (c_t\mathbf{I} + \sum_{i\in\Lambda_{t-1}} \boldsymbol{x}_i\boldsymbol{x}_i^\top)^{-2}$;
  If $b = 1$ then $\Gamma_t = [n]$; else $\Gamma_t = [n]\backslash\Lambda_{t-1}$;
  $i_t \leftarrow \arg\max_{i\in\Gamma_t} \frac{\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t\rangle}{1+\alpha\langle \boldsymbol{x}_i\boldsymbol{x}_i^\top, \mathbf{A}_t^{1/2}\rangle}$;
  $\Lambda_t = \Lambda_{t-1} \cup \{i_t\}$;
end for
Output: $\hat{S} = \Lambda_k$.

Algorithm 2 FINDCONSTANT$(\mathbf{Z}, \alpha)$

Initialization: $c_\ell = -\sigma_{\min}(\mathbf{Z})$, $c_u = \sqrt{p}$; $\epsilon = 10^{-9}$;
while $|c_\ell - c_u| > \epsilon$ **do**
  $\bar{c} \leftarrow (c_\ell + c_u)/2$;
  If $\mathrm{tr}[(\bar{c}\mathbf{I} + \mathbf{Z})^{-2}] > 1$ then $c_\ell \leftarrow \bar{c}$; else $c_u \leftarrow \bar{c}$;
end while
Output: $c = (c_\ell + c_u)/2$.

Our proof of Theorem 1.1 is constructive and yields a computationally efficient iterative algorithm which finds subset $\hat{S} \in \mathcal{S}_b(n,k)$ that satisfies the approximation results in Theorem 1.1. In Algorithm 1 we give a pseducode description of the algorithm, which makes use of a binary search routine (Algorithm 2) that finds the unique constant c_t for which $\mathrm{tr}(\mathbf{A}_t) = \mathrm{tr}[(c_t\mathbf{I} + \sum_{i\in\Lambda_{t-1}} \boldsymbol{x}_i\boldsymbol{x}_i^\top)^{-2}] = 1$. Note that for Eq. (5) to be valid, it is necessary to run Algorithm 2 on the remaining set of $\boldsymbol{\pi}^*$ after including all points $\boldsymbol{x}_i$ with $\boldsymbol{\pi}_i^* \geq 1/2$ in $\hat{S}$.

4. Extension to generalized linear models

The experimental algorithm presented in this paper could be easily extended beyond the linear regression model. For this purpose we consider the *Generalized Linear Model (GLM)*, which assumes that

$$y|\boldsymbol{x} \overset{i.i.d.}{\sim} p(y|\boldsymbol{x}^\top\boldsymbol{\beta}_0),$$

where $p(\cdot|\cdot)$ is a *known* distribution and $\boldsymbol{\beta}_0$ is an unknown p-dimensional regression model. Examples include the logistic regression model $p(y=1|\boldsymbol{x}) = \frac{\exp(\boldsymbol{x}^\top\boldsymbol{\beta}_0)}{1+\exp(\boldsymbol{x}^\top\boldsymbol{\beta}_0)}$, the Possion count model $p(y_i = y|\boldsymbol{x}) = \frac{\exp(y\boldsymbol{x}^\top\boldsymbol{\beta}_0 - e^{-\boldsymbol{x}^\top\boldsymbol{\beta}_0})}{y!}$, and many others.

Let $S \in \mathcal{S}_b(n,k)$ be the set of selected design points from $\mathbf{X}$. Under the classical statistics regime,

Table 2. Simulation results on synthetic data of size $n = 1000$ and $k = 50$. Uniform sampling and weighted sampling are run for 50 independent trials and the median objective is reported. "Inf" means the sample covariance $\mathbf{X}_S^\top \mathbf{X}_S$ does not belong to $\mathbb{S}_p^+$.

	$k = 2p = 100$						$k = 3p = 150$					
	f_A	f_D	f_T	f_E	f_V	f_G	f_A	f_D	f_T	f_E	f_V	f_G
UNIFORM SAMPLING	34.29	7.25	2.05	349.4	101.4	381.4	24.61	6.40	2.03	196.2	73.7	219.1
WEIGHTED SAMPLING	23.42	4.57	Inf	Inf	60.22	202.6	11.18	4.26	0.96	Inf	46.20	119.5
FEDOROV'S EXCHANGE	23.17	5.52	1.15	172.9	44.43	117.7	12.26	4.65	1.22	173.7	73.97	101.8
(running time /secs)	*4.6*	*26*	*2442*	*28*	*488*	*8893*	*296*	*282*	*311*	*360*	*< 1*	*11478*
ALGORITHM 1	12.55	4.72	1.19	53.52	50.47	90.77	11.90	4.60	1.27	41.53	45.97	80.94
(running time /secs)	*< 1*	*< 1*	*< 1*	*< 1*	*< 1*	*< 1*	*< 2*	*< 2*	*< 2*	*< 2*	*< 2*	*< 2*
	$k = 5p = 250$						$k = 10p = 500$					
UNIFORM SAMPLING	20.02	5.82	2.00	137.1	60.2	155.2	17.57	5.51	2.02	103.9	52.93	123.5
WEIGHTED SAMPLING	10.36	4.23	1.14	Inf	41.91	90.61	11.22	4.53	1.44	52.75	43.04	80.74
FEDOROV'S EXCHANGE	11.70	5.84	1.38	116.1	53.14	133.67	12.13	5.52	1.65	108.4	45.05	99.07
(running time /secs)	*441*	*< 1*	*352*	*2552*	*196*	*1152*	*100*	*< 1*	*575*	*< 1*	*1515*	*26804*
ALGORITHM 1	11.14	4.67	1.38	36.67	45.6	76.20	11.60	4.77	1.56	49.27	45.14	81.78
(running time /secs)	*< 2*	*< 2*	*< 2*	*< 2*	*< 2*	*< 2*	*< 5*	*< 5*	*< 5*	*< 5*	*< 5*	*< 5*

the maximum likelihood (ML) estimator $\hat{\boldsymbol{\beta}}^{\mathrm{ML}} = \arg\min_{\boldsymbol{\beta}} \sum_{i \in S} \log p(y_i | \boldsymbol{x}_i^\top \beta)$ is asymptotically efficient, and its asymptotic variance equals the *Fisher's information*

$$I(\mathbf{X}_S; \boldsymbol{\beta}_0) := \sum_{i \in S} \mathbb{E}_{y | \boldsymbol{x}_i^\top \boldsymbol{\beta}_0} \left[-\frac{\partial^2 \log p(y | \boldsymbol{x}_i; \boldsymbol{\beta}_0)}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^\top} \right] \overset{\eta_i = \boldsymbol{x}_i^\top \boldsymbol{\beta}_0}{=\!=} \sum_{i \in S} \mathbb{E}_{y | \eta_i} \left[-\frac{\partial^2 \log p(y | \eta_i)}{\partial \eta_i^2} \right] \cdot \boldsymbol{x}_i \boldsymbol{x}_i^\top.$$

Here the second equality is due to the sufficiency of $\boldsymbol{x}_i^\top \boldsymbol{\beta}_0$ in a GLM. Note that for the linear regression model $\boldsymbol{y} = \mathbf{X}\boldsymbol{\beta}_0 + \boldsymbol{w}$, the ML estimator is the ordinary least squares (OLS) $\hat{\boldsymbol{\beta}} = (\mathbf{X}_S^\top \mathbf{X}_S)^{-1} \mathbf{X}_S \boldsymbol{y}_S$ and its Fisher's information equals the sample covariance $\mathbf{X}_S^\top \mathbf{X}_S$. The experimental design problem can then be formalized as follows: [2]

$$\min_{S \in \mathcal{S}_b(n,k)} f(I(\mathbf{X}_S; \boldsymbol{\beta}_0)) = \min_{S \in \mathcal{S}_b(n,k)} f\left(\sum_{i \in S} \boldsymbol{z}_i \boldsymbol{z}_i^\top \right); \quad \boldsymbol{z}_i = \sqrt{-\mathbb{E}_{y | \eta_i} \left[-\frac{\partial^2 \log p(y_i | \eta_i)}{\partial \eta_i^2} \right]}, \eta_i = \boldsymbol{x}_i^\top \boldsymbol{\beta}_0. \tag{15}$$

Suppose $\check{\boldsymbol{\beta}}$ is a "pilot" estimate of $\boldsymbol{\beta}_0$, obtained from a uniformly sampled design subset S_1. A near-optimal design set S_2 can then be constructed by minimizing Eq. (15) using $\check{\eta}_i = \boldsymbol{x}_i^\top \check{\boldsymbol{\beta}}$. Such an approach was adopted in sequential design and active learning for ML estimators (Chaudhuri et al., 2015; Khuri et al., 2006); however, with our algorithm the quality of S_2 is greatly improved.

[2] Under very mild conditions $\mathbb{E}[-\frac{\partial^2 \log p}{\partial \eta^2}] = \mathbb{E}[(\frac{\partial \log p}{\partial \eta})^2]$ is non-negative (Van der Vaart, 2000).

5. Numerical results

We compare the proposed method with several baseline methods on both synthetic and real-world data sets. We only consider the harder "without replacement" setting, where each row of $\mathbf{X}$ can be selected at most once.

5.1. Methods and their implementation

We compare our algorithm with three simple heuristic methods that apply to all optimality criteria:

1. **Uniform sampling**: $\hat{S}$ is sampled uniformly at random without replacement from the design pool $\mathbf{X}$;
2. **Weighted sampling**: first the optimal solution $\boldsymbol{\pi}^*$ of Eq. (6) is computed with $r = k$; afterwards, $\hat{S}$ is sampled without replacement according to the distribution specified by $\boldsymbol{\pi}^*/k$. Recall that (Wang et al., 2016) proved that weighted sampling works when k is sufficiently large compared to p (cf. Table 1). [3]
3. **Fedorov's exchange (Miller & Nguyen, 1994)**: the algorithm starts with a random subset $S_0 \in \mathcal{S}_b(n,k)$ and iteratively exchanges two coordinates $i \in S_0$, $j \notin S_0$ such that the objective is minimized after the exchange. The algorithm terminates if no such exchange can reduce the objective, or T iterations are reached.

All algorithms are implemented in MATLAB, except for the Fedorov's exchange algorithm, which is implemented in C due to efficiency concerns. We also apply the Sherman-Morrison formula $(\mathbf{A} + \lambda \boldsymbol{u}\boldsymbol{u}^\top)^{-1} = \mathbf{A}^{-1} + \frac{\lambda \mathbf{A}^{-1} \boldsymbol{u}\boldsymbol{u}^\top \mathbf{A}^{-1}}{1 + \lambda \boldsymbol{u}^\top \mathbf{A}^{-1} \boldsymbol{u}}$ and the matrix determinant lemma $\det(\mathbf{A} + \lambda \boldsymbol{u}\boldsymbol{u}^\top) =$

[3] Fact 2.2 ensures that $\boldsymbol{\pi}^*/k$ is a valid probability distribution.

Table 3. Results on the Minnesota wind speed dataset ($n = 2642, p = 15, k = 30$). MSE is defined as $\sqrt{\frac{1}{n}\|\boldsymbol{y} - \mathbf{V}\hat{\boldsymbol{\beta}}\|_2^2}$.

	f_V	MSE	f_G	MSE
UNIFORM SAMPLING	94.1	1.10	3093	1.34
WEIGHTED SAMPLING	21.4	0.89	2451	1.13
FEDOROV'S EXCHANGE	10.0	0.86	29.2	0.78
(running time /secs)	*15*	-	*1857*	-
ALGORITHM 1	10.8	0.72	29.2	0.76
(running time /secs)	*< 1*	-	*< 1*	-
FULL-SAMPLE OLS	-	0.55	-	0.55

$(1 + \lambda \boldsymbol{u}^\top \mathbf{A}^{-1} \boldsymbol{u}^\top) \det(\mathbf{A})$ to accelerate computations of rank-1 updates of matrix inverse and determinant. For uniform sampling and weighted sampling, we report the median objective of 50 indpendent trials. We only report the objective for one trial of Fedorov's exchange method due to time constraints. The maximum number of iterations T for Fedorov's exchange is set at $T = 100$. We always set $k = r$ in the optimization problem Eq. (6), and details of solving Eq. (6) are placed in the appendix. In Algorithm 1 we set $\alpha = 10$; our similuations suggest that the algorithm is not sensitive to α.

5.2. Synthetic data

We synthesize a 1000×50 design pool $\mathbf{X}$ as follows:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_A & \mathbf{0}_{500\times 25} \\ \mathbf{0}_{500\times 25} & \mathbf{X}_B \end{bmatrix}.$$

$\mathbf{X}_A$ is a 500×25 random Gaussian matrix, re-scaled so that the eigenvalues of $\mathbf{X}_A^\top \mathbf{X}_A$ satisfy a quadratic decay: $\sigma_j(\mathbf{X}_A^\top \mathbf{X}_A) \propto j^{-2}$; $\mathbf{X}_B$ is a 500×25 Gaussian matrix with i.i.d. standard Normal variables. Both $\mathbf{X}_A$ and $\mathbf{X}_B$ have comparable Frobenius norm.

In Table 2 we report results on all 6 optimality criteria $(f_A, f_D, f_T, f_E, f_V, f_G)$ for $k \in \{2p, 3p, 5p, 10p\}$. We also report the running time (measured in seconds) of Algorithm 1 and the Fedorov's exchange algorithm. The other two sampling based algorithms are very efficient and always terminate within one second. We observe that our algorithm has the best performance for f_E and f_G, while still achieving comparable results for the other optimality criteria. It is also robust when k is small compared to p, while sampling based methods occasionally produce designs that are not even full rank. Finally, Algorithm 1 is computationally efficient and terminates within seconds for all settings.

5.3. The Minnesota wind speed dataset

The Minnesota wind dataset collects wind speed information across $n = 2642$ locations in Minnesota, USA for a period of 24 months (for the purpose of this experiment, we only use wind speed data for one month). The 2642 locations are connected with 3304 bi-directional roads, which form an $n \times n$ sparse unweighted undirected graph G. Let $\mathbf{L} = \mathrm{diag}(\boldsymbol{d})?\mathbf{G}$ be the $n \times n$ Laplacian of $\mathbf{G}$, where $\boldsymbol{d}$ is a vector of node degrees, and let $\mathbf{V} \in \mathbb{R}^{n\times p}$ be an orthonormal eigenbasis corresponding to the smallest p eigenvalues of $\mathbf{L}$. (Chen et al., 2015) shows that the relatively smooth wind speed signal $\boldsymbol{y} \in \mathbb{R}^n$ can be well approximated by using only $p = 15$ graph Laplacian basis.

In Table 3 we compare the mean-square error (MSE) for prediction on the full design pool $\mathbf{V}$: $\mathrm{MSE} = \sqrt{\frac{1}{n}\|\boldsymbol{y} - \mathbf{V}\hat{\boldsymbol{\beta}}\|_2^2}$. Because the objective is prediction based, we only consider the two prediction related criteria: $f_V(\boldsymbol{\Sigma}) = \mathrm{tr}(\mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{V}^\top)$ and $f_G(\boldsymbol{\Sigma}) = \max \mathrm{diag}(\mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{V}^\top)$. The subset size k is set as $k = 2p = 30$, which is much smaller than $n = 2642$. We observe that Algorithm 1 consistently outperforms the other heuristic methods, and is so efficient that its running time is negligible. It is also interesting that by using $k = 30$ samples Algorithm 1 already achieves an MSE that is comparable to the OLS on the entire $n = 2642$ design pool.

6. Concluding remarks and open questions

We proposed a computationally efficient algorithm that approximately computes optimal solutions for the experimental design problem, with near-optimal requirement on k (i.e., the number of experiments to choose). In particular, we obtained a *constant* approximation under the very weak condition $k > 2p$, and a $(1 + \varepsilon)$ approximation if replacement or over-sampling is allowed. Our algorithm works for all regular optimality criteria.

An important open question is to achieve $(1 + \varepsilon)$ relative approximation ratio under the "proper sampling" regime $k = r$, or the "slight over-sampling" regime $k = (1 + \delta)r$, for the without replacement model. It was shown in (Wang et al., 2016) that a simple greedy method achieves $(1 + \varepsilon)$ approximation ratio for A- and V-optimality provided that $k = \Omega(p^2/\varepsilon)$. Whether such analysis can be extended to other optimality criteria and whether the p^2 term can be further reduced to a near linear function of p remain open.

Another practical question is to develop fast-converging optimization methods for the continuous problem in Eq. (6), especially for criteria that are not differentiable such as the E- and G-optimality, where subgradient methods have very slow convergence rate.

Acknowledgement This work is supported by NSF grants CAREER IIS-1252412 and CCF-1563918. We thank Adams Wei Yu for providing an efficient implementation of the projection step, and other useful discussions.

References

Ageev, Alexander A and Sviridenko, Maxim I. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization*, 8(3):307–328, 2004.

Allen-Zhu, Zeyuan, Liao, Zhenyu, and Orecchia, Lorenzo. Spectral sparsification and regret minimization beyond matrix multiplicative updates. In *Proceedings of Annual Symposium on the Theory of Computing (STOC)*, 2015.

Avron, Haim and Boutsidis, Christos. Faster subset selection for matrices and applications. *SIAM Journal on Matrix Analysis and Applications*, 34(4):1464–1499, 2013.

Balcan, Maria-Florina and Long, Philip M. Active and passive learning of linear separators under log-concave distributions. In *Proceedings of Annual Conference on Learning Theory (COLT)*, 2013.

Bouhtou, Mustapha, Gaubert, Stephane, and Sagnol, Guillaume. Submodularity and randomized rounding techniques for optimal experimental design. *Electronic Notes in Discrete Mathematics*, 36:679–686, 2010.

Boutsidis, Christos and Woodruff, David P. Optimal CUR matrix decompositions. In *Proceedings of Annual Symposium on the Theory of Computing (STOC)*, 2014.

Černỳ, Michal and Hladík, Milan. Two complexity results on C-optimality in experimental design. *Computational Optimization and Applications*, 51(3):1397–1408, 2012.

Chaudhuri, Kamalika, Kakade, Sham, Netrapalli, Praneeth, and Sanghavi, Sujay. Convergence rates of active learning for maximum likelihood estimation. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2015.

Chen, Siheng, Varma, Rohan, Singh, Aarti, and Kovačević, Jelena. Signal representations on graphs: Tools and applications. *arXiv preprint arXiv:1512.05406*, 2015.

Çivril, Ali and Magdon-Ismail, Malik. On selecting a maximum volume sub-matrix of a matrix and related problems. *Theoretical Computer Science*, 410(47-49):4801–4811, 2009.

Deshpande, Amit and Rademacher, Luis. Efficient volume sampling for row/column subset selection. In *Proceedings of Annual Conference on Foundations of Computer Science (FOCS)*, 2010.

Deshpande, Yash and Montanari, Andrea. Linear bandits in high dimension and recommendation systems. In *Proceedings of Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2012.

Dhillon, Paramveer, Lu, Yichao, Foster, Dean P, and Ungar, Lyle. New subsampling algorithms for fast least squares regression. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2013.

Drineas, Petros and Mahoney, Michael W. On the Nyström method for approximating a gram matrix for improved kernel-based learning. *Journal of Machine Learning Research*, 6(12):2153–2175, 2005.

Drineas, Petros, Mahoney, Michael W, and Muthukrishnan, S. Relative-error CUR matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30(2): 844–881, 2008.

Fedorov, Valerii Vadimovich. *Theory of optimal experiments*. Elsevier, 1972.

Hazan, Elad and Karnin, Zohar. Hard-margin active linear regression. In *Proceedings of International Conference on Machine Learning (ICML)*, 2015.

Horel, Thibaut, Ioannidis, Stratis, and Muthukrishnan, S. Budget feasible mechanisms for experimental design. In *Proceedings of Latin American Symposium on Theoretical Informatics (LATIN)*, 2014.

Huang, Ruitong, Lattimore, Tor, György, András, and Szepesvári, Csaba. Following the leader and fast rates in linear prediction: Curved constraint sets and other regularities. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2016.

Khuri, Andre, Mukherjee, Bhramar, Sinha, Bikas, and Ghosh, Malay. Design issues for generalized linear models: a review. *Statistical Science*, 21(3):376–399, 2006.

Kulesza, Alex and Taskar, Ben. Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2–3):123–286, 2012.

Li, Chengtao, Jegelka, Stefanie, and Sra, Suvrit. Polynomial time algorithms for dual volume sampling. *arXiv preprint arXiv:1703.02674*, 2017.

Ma, Yifei, Garnett, Roman, and Schneider, Jeff. σ-optimality for active learning on Gaussian random fields. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2013.

McMahan, H Brendan. Follow-the-regularized-leader and mirror descent: Equivalence theorems and L1 regularization. In *Procedings of International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

Miller, Alan and Nguyen, Nam-Ky. A Fedorov exchange algorithm for d-optimal design. *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, 43(4):669–677, 1994.

Pilanci, Mert and Wainwright, Martin J. Iterative Hessian sketch: Fast and accurate solution approximation for constrained least-squares. *Journal of Machine Learning Research*, 17(53):1–38, 2016.

Pukelsheim, Friedrich. *Optimal design of experiments*. SIAM, 2006.

Raskutti, Garvesh and Mahoney, Michael. A statistical perspective on randomized sketching for ordinary least-squares. *arXiv preprint arXiv:1406.5986*, 2014.

Ravi, Sathya N, Ithapu, Vamsi K, Johnson, Sterling C, and Singh, Vikas. Experimental design on a budget for sparse linear models and applications. In *Proceedings of International Conference on Machine Learning (ICML)*, 2016.

Silva, Marcel K, Harvey, Nicholas JA, and Sato, Cristiane M. Sparse sums of positive semidefinite matrices. *ACM Transactions on Algorithms*, 12(1):9, 2016.

Spielman, Daniel A and Srivastava, Nikhil. Graph sparsification by effective resistances. *SIAM Journal on Computing*, 40(6):1913–1926, 2011.

Su, Hao, Yu, Adams Wei, and Li, Fei-Fei. Efficient euclidean projections onto the intersection of norm balls. In *Proceedings of International Conference on Machine Learning (ICML)*, 2012.

Van der Vaart, Aad W. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

Wang, Shusen and Zhang, Zhihua. Improving CUR matrix decomposition and the Nyström approximation via adaptive sampling. *Journal of Machine Learning Research*, 14(1):2729–2769, 2013.

Wang, Yining and Singh, Aarti. An empirical comparison of sampling techniques for matrix column subset selection. In *Proceedings of Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2015a.

Wang, Yining and Singh, Aarti. Provably correct active sampling algorithms for matrix column subset selection with missing data. *arXiv preprint arXiv:1505.04343*, 2015b.

Wang, Yining and Singh, Aarti. Noise-adaptive margin-based active learning and lower bounds under tsybakov noise condition. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2016.

Wang, Yining, Yu, Wei Adams, and Singh, Aarti. On computationally tractable selection of experiments in regression models. *arXiv preprints: arXiv:1601.02068*, 2016.

Woodruff, David P. Sketching as a tool for numerical linear algebra. *Foundations and Trends® in Theoretical Computer Science*, 10(1–2):1–157, 2014.