On proportional volume sampling for experimental design in general spaces

Arnaud Poinas · Rémi Bardenet

Received: 22 February 2021 / Accepted: 5 June 2022 / Published online: 31 December 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract
Optimal design for linear regression is a fundamental task in statistics. For finite design spaces, recent progress has shown that random designs drawn using proportional volume sampling (PVS for short) lead to polynomial-time algorithms with approximation guarantees that outperform i.i.d. sampling. PVS strikes the balance between design nodes that jointly fill the design space, while marginally staying in regions of high mass under the solution of a relaxed convex version of the original problem. In this paper, we examine some of the statistical implications of a new variant of PVS for (possibly Bayesian) optimal design. Using point process machinery, we treat the case of a generic Polish design space. We show that not only are known A-optimality approximation guarantees preserved, but we obtain similar guarantees for D-optimal design that tighten recent results. Moreover, we show that our PVS variant can be sampled in polynomial time. Unfortunately, in spite of its elegance and tractability, we demonstrate on a simple example that the practical implications of general PVS are likely limited. In the second part of the paper, we focus on applications and investigate the use of PVS as a subroutine for stochastic search heuristics. We demonstrate that PVS is a robust addition to the practitioner’s toolbox, especially when the regression functions are nonstandard and the design space, while low-dimensional, has a complicated shape (e.g., nonlinear boundaries, several connected components).

Keywords Bayesian optimal design · Volume sampling · Determinantal point processes · Search heuristics

1 Introduction

In the problem of experimental design for linear models, the main goal is to select points in a set \( \Omega \), called the design space; noisy evaluations of the regressed function at these design points are then collected to form a training set. A widespread criterion to select the design points is to minimize the variance of the resulting Gauss–Markov estimator. Given a prior, Bayesians rather minimize the posterior variance on the parameters of the linear model, which turns out to formally generalize the former classical criterion.

The literature and the available techniques are usually partitioned according to properties of the design space \( \Omega \). For instance, taking \( \Omega \) to be a finite set leads to factorial experiments (Atkinson et al. 2007, chapter 7) and row subset selection (Dereziński et al. 2018). Very often, \( \Omega \) is a rectangular subset of \( \mathbb{R}^d \), but more complicated shapes are also not uncommon in applications. For example, \( \Omega \) becomes a simplex when considering mixture designs (Atkinson et al. 2007, Chapter 16), and some of the variables can satisfy nonlinear constraints for physical reasons (Piepel et al. 2019; Atkinson et al. 2007, Example 16.3). Finally, taking \( \Omega \) to be a product of both a finite set and a compact subset of \( \mathbb{R}^d \) allows including both qualitative and quantitative variables (Atkinson et al. 2007, Example 1.2).

Whatever the choice of design space \( \Omega \), the problem of minimizing the variance matrix of the Gauss–Markov estimator (or its Bayesian counterpart) is difficult to solve, both theoretically and numerically. For starters, the Loewner order on positive definite symmetric matrices is not total. The common workaround is to minimize instead a real-valued function of these matrices, which somehow measures the size of the variance of the Gauss–Markov estimator. Even using such proxies, the resulting optimization problem remains difficult to tackle.
For finite design spaces, for instance, one faces a combinatorial optimization problem, typically framed as an optimization over integer-weighted measures on $\Omega$. One seminal idea in this discrete setting has been to rather solve a continuous relaxation of the original minimization problem, yielding a weighted measure on $\Omega$ with real weights, called an approximate optimal design. Several postprocessing procedures, like rounding, have been investigated that recover an integer-weighted measure from a real-weighted approximate optimal design (Pukelsheim 2006, Chapter 12). Recently, (Nikolov et al. 2019) introduced a novel such postprocessing procedure. They use the approximate design to define proportional volume sampling, a distribution over subsets of the (large) finite set $\Omega$ that (i) typically charges designs close to optimality, and (ii) can be sampled in polynomial time. The same distribution was also introduced in Dereziński et al. (2018) for a similar purpose, and generalized to a continuous design space in Dereziński et al. (2019). The key idea is to sample designs that strike a balance between, on the one side, each design point having a large mass under the approximate design, and on the other side, the points together corresponding to a small confidence ellipsoid for the Gauss–Markov estimator. The latter joint property is what distinguishes proportional volume sampling from typical rounding procedures. Finally, (Dereziński et al. 2020) modified proportional volume sampling distribution so that it applies to Bayesian optimal design on a finite design space.

This intense activity around proportional volume sampling in theoretical computer science prompted us to investigate the applicability of the same rationale to Bayesian or classical design in general design spaces, including continuous and mixed experimental design. To the best of our knowledge, the only previous stab at infinite design spaces is (Dereziński et al. 2019), who investigate theoretical properties of proportional volume sampling for classical design on a continuous space.

Our contributions are as follows. After a short survey of optimal design and related work in Sect. 2, we introduce in Sect. 3 a generalization of proportional volume sampling to any bounded closed Polish design space, where a Polish space is an umbrella mathematical object that includes all known design spaces, finite, mixed, or continuous. Our extension uses the formalism of point processes, which allows us to derive simple generic proofs of the known properties of PVS, i.e., that it yields unbiased estimates of the inverse information matrix and its determinant. We also prove new approximation guarantees for A- and D-optimal designs generated from PVS. On top of applying to any setting (Bayesian or not, finite or not), our bounds also slightly refine the previous results of Dereziński et al. (2019) in the classical continuous setting and of Dereziński et al. (2020) in the Bayesian finite setting. Because we focus on the standard statistical formulation of experimental design (Atkinson et al. 2007), our guarantees use PVS conditioned on the design having a fixed, used-defined cardinality. Furthermore, we give a sampling algorithm for random designs generated from this general conditioned PVS. This algorithm further highlights the connection between PVS and determinantal point processes established in the finite frequentist setting by Nikolov et al. (2019). To close Sect. 3, we illustrate with numerical simulations that general PVS generates better Bayesian experimental designs than i.i.d. designs, and that both approaches benefit from solving a common convex relaxation of the optimal design problem. This concludes the first part of the paper on the theoretical properties of PVS on general spaces.

The second part of the paper starts from the observation that, in spite of PVS’s theoretical backing, in practice it is often outperformed by standard heuristics for optimal design. This is why we examine a second way of using finite PVS, in combination with stochastic search heuristics. We describe this alternative extension of Nikolov et al. (2019) and Dereziński et al. (2020) in Sect. 4, namely a global search heuristic that uses finite PVS as a subroutine. While standard algorithms, including exact methods when applicable (De Castro et al. 2019), remain the preferred solution in the typical setting of a rectangular design space and low-degree polynomial regression functions, we show on several examples in Sect. 4.3 that our PVS-based heuristic outperforms them when the design space has a more complicated shape and the regression functions are generic. Moreover, in general, PVS is at least a useful initialization procedure for local search heuristics.

2 Background and related work

In this section, we recall the usual mathematical setting for optimal design in linear regression, and survey a few key results in the rich statistical literature on the subject.

2.1 Optimal design for linear models

Let $y_1, \ldots, y_k \in \mathbb{R}$ denote the responses of a fixed number $k \geq p$ of independent experiments with input variables $(x_1, \ldots, x_k) \in \Omega^k$. We consider a linear regression model over $p$ linearly independent regression functions $\phi_1, \ldots, \phi_p \in L^2(\Omega)$,

$$Y = \phi(X)\beta + \epsilon,$$  

(2.1)

where $X := (x_1, \ldots, x_k)^T$, $Y := (y_1, \ldots, y_k)^T$, $\epsilon \in \mathbb{R}^k$ is a random variable such that $E[\epsilon] = 0$ and $\text{Var}(\epsilon) = \sigma^2 I_k$, and $\phi(X) = (\phi_j(x_i)) \in \mathbb{R}^{k \times p}$ is called the design matrix. The feature functions $\phi_j$ are often polynomials, but experimental design for different regression functions has also been...
considered, such as trigonometric polynomials (Dette et al. 2002), Haar wavelets (Maronge et al. 2017) or B-splines (Grove et al. 2004). Assuming that the information matrix $\phi(X)^T \phi(X) + \Lambda$ is invertible, one popular estimator for $\beta$ in (2.1) is

$$\hat{\beta} := (\phi(X)^T \phi(X) + \Lambda)^{-1}(\phi(X)^T Y),$$

(2.2)

where $\Lambda$ is a positive semi-definite matrix. When $\Lambda = 0$, (2.2) corresponds to the classical Gauss–Markov estimator, while the case $\Lambda \neq 0$ corresponds to the posterior mean of $\beta$ for any prior distribution with mean 0 and variance $\sigma^2 \Lambda^{-1}$. In the rest of the paper, we often abusively refer to $\Lambda \neq 0$ as the Bayesian setting, although frequentists may also prefer to use a nonzero $\Lambda$, as in ridge regression. The posterior covariance matrix, when $\Lambda \neq 0$ and for a given design $X$, is given by

$$\sigma^2(\phi(X)^T \phi(X) + \Lambda)^{-1}.$$  

(2.3)

Upon noting that, when $\Lambda = 0$, $\hat{\beta}$ in (2.2) is Gaussianly distributed around $\beta$ with covariance matrix (2.3), both frequentists and Bayesians commonly aim at minimizing the inverse of the so-called information matrix

$$\phi(X)^T \phi(X) + \Lambda,$$

in the sense of the Loewner order. But, since the Loewner order is only a partial order on the space $S_p(\mathbb{R})^+$ of positive $p \times p$ symmetric matrices, (2.3) does not necessarily admit a global optimum. As a proxy, it is thus common to mini-

mize a decreasing convex function $h: S_p(\mathbb{R})^+ \rightarrow \mathbb{R}$ of the information matrix.

In this paper, we focus on two of the most common proxies, $h_A(M) := \text{Tr}(M^{-1})$ and $h_D(M) := \det(M^{-1})$, which are, respectively, called the A and D-optimality criteria (Pukelsheim 2006). An optimal design is thus defined as an element of $\arg\min_{X \subset \Omega^k} h(\phi(X)^T \phi(X) + \Lambda).$  

(2.4)

Even though $h$ is a convex function, this is not the case for the set of information matrices $\{\phi(X)^T \phi(X) + \Lambda, X \subset \Omega^k\}$, making the optimization problem difficult. Even when $\Omega$ is a finite space with $n$ elements, (2.4) writes as a finite optimization problem over a space with $\binom{n}{k}$ elements, which is usually too large for exhaustive search. A common technique is to solve instead a convex relaxation of (2.4).

$$\arg\min_{\nu \in M(\Omega)} h(G_{\nu}(\phi) + \Lambda) \text{ s.t. } \nu(\Omega) = k,$$

(2.5)

where $M(\Omega)$ is the space of Borel measures on $\Omega$, and $G_{\nu}(\phi) := \left(\int_{\Omega} \phi_1(\chi) \phi_j(\chi) d\nu(\chi)\right) \in \mathbb{R}^{p \times p}$ is the Gramian matrix. A solution of the convex optimization problem (2.5) is called an approximate optimal design.

### 2.2 The case of discrete design spaces

As an illustration of the difficulty of finding an optimal design when $\Omega$ is a finite set, (Summa et al. 2014) show that the problem of getting a $(1+\epsilon)$-approximation of the D-optimal design is NP-hard for small enough $\epsilon$ when $\Lambda = 0$ (i.e., in the non-Bayesian setting). A similar result for A-optimality, when $k = p$, appears in Nikolov et al. (2019). In contrast, the relaxation (2.4) becomes a convex optimization problem over a finite-dimensional space, for which efficient algorithms exist (Boyd and Vandenberghe 2004, Chapter 7.5.2). A natural question is thus how to extract a near-optimal design from an approximate design in (2.4). Several rounding algorithms were introduced for that purpose (Pukelsheim 2006, Chapter 12), such as the popular method proposed by Pukelsheim and Rieder (1992), which looks for the best design across the ones obtained by rounding up or down the coefficients of an approximate design.

Still with $\Lambda = 0$ in (2.2), Nikolov et al. (2019) and Dereziński et al. (2018) introduced an alternative to rounding, called “proportional” or “rescaled” volume sampling. The principle is to sample a random design $X$ with a probability density proportional to

$$\det(\phi(X)^T \phi(X)) d\nu^k(X),$$

where $\nu$ is an approximate optimal design. Nikolov et al. (2019) showed that for the A-optimality criterion, the generated designs are on average at least a $k/(k - p + 1)$-approximation of the A-optimal design. Dereziński et al. (2020) further introduced the regularized determinantal point process distribution as a way to generalize these results to the Bayesian setting, yielding $(1 + O(\sqrt{\ln(1/k)})$-approximations of the optimal design for several optimality criteria, including A and D, provided $k$ is large enough.

### 2.3 The case of continuous design spaces

Theory is well developed in the non-Bayesian univariate case, when $\Omega$ is a compact subset of $\mathbb{R}$ and $\Lambda = 0$. The approximate D-optimal design has been characterized as zeros of certain families of polynomials for both polynomial (Dette and Studden 1997, Theorem 5.5.3) and trigonometric regression (Dette et al. 2002, Theorem 3.1). Unfortunately, these results do not extend well to multivariate regression. When $\Omega$ is a compact subset of $\mathbb{R}^d$, some results do exist, but only for very specific design spaces and regression functions (Farrell et al. 1967; Liski et al. 2002). Theoretical knowledge in the Bayesian case is even more limited but a few results on
D-optimality in some specific cases are known (Dette 1993; Liu et al. 2020).

Regarding the relaxation (2.5) of the optimal design problem, common practice is to optimize it only over finitely supported measures (Dette and Studden 1997; De Castro et al. 2019), and to look for an optimal design within the support of a solution to that relaxation. This is partially motivated by the result that the relaxed problem (2.5) always has a finitely supported solution; see, e.g., Pukelsheim 2006, Theorem 8.2. We refer to Pronzato and Pázman (2013) for a survey of optimization methods for the relaxation (2.5). As a recent example of this line of research, (De Castro et al. 2020) gives an algorithm solving (2.5) when \( \Lambda = 0, h \) is either the \( A \) or D-optimality criterion, and the regression functions are multivariate polynomials. The algorithm is based on an elegant construction of a nested sequence of convex optimization problems, where the search space is formed by the moments of the target measure. Unfortunately, because of the quickly exploding size of these optimization problems, along with some numerical instability issues, the practical impact of this algorithm has so far remained limited to small \( p \) and \( d \). In particular, experiments in De Castro et al. (2019) consider polynomial regressions with degree up to 3 in dimension \( d \leq 3 \). The extent of this domain of applicability is confirmed by our own numerical experiments; see Sect. 4.3.

In parallel, a lot of effort has been put into designing efficient optimization heuristics for optimal design, although not many algorithms are agnostic to the properties of \( \Omega \). We describe two such families in Sect. 4: local search algorithms (Fang et al. 2006, Chapter 4) and the exchange method (Pronzato and Pázman 2013, Chapter 9.2.1).

3 Proportional volume sampling in general spaces

The distribution known as regularized determinantal point processes was introduced by Dereziński et al. (2020) to generalize the proportional volume sampling distribution of Nikolov et al. (2019) to the finite Bayesian setting. In this section, we further extend the definition and results of Dereziński et al. (2020) to any Polish design space \( \Omega \), i.e. a separable, completely metrizable topological space. To avoid later confusion with determinantal point processes in our paper, we stick to the original name of proportional volume sampling, often shortened as PVS, for our distribution.

Other than treating general design spaces, notable differences of this section with Dereziński et al. (2020) include a more general and slightly tighter bound on the average A- and D-optimality criteria of designs sampled from PVS, conditioned on having a fixed size. We also give an algorithm to sample from (possibly size-constrained) PVS that does not require rejection sampling, unlike in Dereziński et al. (2020, Lemma 13).

3.1 Definition

We define proportional volume sampling (PVS) as a point process, i.e., a distribution over the space \( \cup_{k \geq 0} \Omega^k \) of finite point configurations in \( \Omega \). For notions related to point processes, such as Janossy densities and correlation functions, we refer to Daley and Vere-Jones (2003), the standard reference. PVS is parameterized by a probability measure \( \nu \), and \( k \) independent functions \( \phi_1, \ldots, \phi_k \) in \( L^2(\Omega, \nu) \).

**Definition 3.1** Let \( \Lambda \) be a \( p \times p \) nonnegative symmetric matrix; let \( \nu \) be a finite measure on \( (\Omega, B(\Omega)) \) such that \( G_\nu(\phi) \) is non-singular, where \( B(\Omega) \) is the Borel algebra and \( G_\nu(\phi) \in \mathbb{R}^{p \times p} \) is the matrix with entries \( \int_{\Omega} \phi_i(x) \phi_j(x) d\nu(x) \). We define the proportional volume sampling distribution \( \mathbb{P}_{\text{VS}}(\phi, \Lambda) \) with reference measure \( \nu \) as the point process on \( (\Omega, \nu) \) with Janossy densities

\[
j_n(x_1, \ldots, x_n) = \frac{\det(\phi(x)^T \phi(x) + \Lambda)}{\det(G_\nu(\phi) + \Lambda) \exp(\nu(\Omega))},
\]

for all \( n \in \mathbb{N} \) and \( x \in \Omega^n \).

Some comments are in order. First, we recall that \( j_n(x_1, \ldots, x_n) d\nu(x_1) \cdots d\nu(x_n) \) can be interpreted as the probability of the point process consisting of \( n \) points, one in the neighborhood of each \( x_i \) (Daley and Vere-Jones 2003, Section 5.3). It follows that PVS strikes a balance between favoring designs that lie in regions of large mass under \( \nu \), and designs with feature vectors \( (\phi_i(x_j)) \in \mathbb{R}^n, i = 1, \ldots, k \) that span a large volume, as measured by their determinant. When \( k = p \) and \( \Lambda = 0 \), making this determinant large is equivalent to having a small-volume confidence ellipsoid for the Gauss–Markov estimator (2.2). Second, Definition 3.1 bears resemblance to that of determinantal point processes (Macchi 1975; Hough et al. 2006), an observation that we make precise in Sect. 3.3. Third, the fact that (3.1) is well defined can be seen by showing that

\[
\sum_{n \geq 0} \frac{1}{n!} \int_{\Omega^n} j_n(x) d\nu^n(x) = 1;
\]

see (Daley and Vere-Jones 2003 Proposition 5.3.II.(ii)). This can be done using both the classical Cauchy-Binet formula and a generalization of it (Johansson 2006), combined with the well-known matrix identity (Collings 1983) stating that for any \( n \times n \) matrix \( M \) and diagonal matrix \( D \) with diagonal entries \( \lambda_1, \ldots, \lambda_n \),

\[
det(M + D) = \sum_{S \subseteq \{1, \ldots, n\}} \det(M_S) \prod_{i \notin S} \lambda_i
\]
where $M_S$ is the submatrix of $M$ of the rows and columns indexed by $S$. The detailed proof is given in Sect. A.

As stated in the introduction, our goal is to investigate a generalization of a family of random designs appearing in the computer science literature. To be precise, our definition of PVS is related to previous work as follows. When $\Omega$ is a finite set, if we condition PVS to only generate designs with distinct points, then we get the distribution called regularized determinantal point process in Dereziński et al. (2020). When $\Omega$ is a general Polish set and $\Lambda = 0$ (the non-Bayesian case), if we further condition PVS to only generate designs of fixed cardinality, we get the distribution called volume-rescaled sampling in Dereziński et al. (2019). When $\Omega$ is a finite set and $\Lambda = 0$, if we again condition PVS to only generate designs with fixed size, then we get the distribution called proportional volume sampling with hardcore distribution in Nikolov et al. (2019). The latter is identical to the distribution called rescaled volume sampling in Dereziński et al. (2018).

3.2 PVS and optimal designs

Dereziński et al. (2020) show that their regularized DPP distribution gives natural unbiased estimators of both $G_v(\phi) + \Lambda^{-1}$ and $\det(G_v(\phi) + \Lambda)^{-1}$ in the discrete case. We show that this property extends to PVS defined on a general Polish spaces $\Omega$.

**Proposition 3.2** Let $X \sim \mathbb{P}^v_{\text{VS}}(\phi, \Lambda)$, then

$$\mathbb{E}[\det(\phi(X)^T \phi(X) + \Lambda)^{-1}] = \det(G_v(\phi) + \Lambda)^{-1}$$

(3.2)

and

$$\mathbb{E}[\det(\phi(X)^T \phi(X) + \Lambda)^{-1}] = \det(G_v(\phi) + \Lambda)^{-1}.$$  

(3.3)

**Proof** The proof is given in Sect. B. It relies on the fact that for any function $f : \cup_{n \geq 0} \Omega^n \rightarrow \mathbb{R}$,

$$\mathbb{E}[f(X)] = \sum_{n \geq 0} \frac{1}{n!} \int_{\Omega^n} f(x) j_n(x) d\nu^n(x);$$

(3.4)

see Daley and Vere-Jones (2003, Exercice 5.3.8). \qed

Proposition 3.2 shows that by taking $\nu$ to be a solution to the relaxation (2.5) of the D-optimal design problem, the designs generated by PVS will perform, on average, at least as well as an $A$ or $D$-optimal design. The obvious issue here is that PVS generates designs with a random number of points. When $\nu(\Omega) = k$ we later show in Corollary 3.10 that the average number of points generated by $\mathbb{P}^v_{\text{VS}}(\phi, \Lambda)$ is

$$k + \text{Tr}(G_v(\phi)\Lambda)^{-1}) \in [k, k + p].$$

When $k$ is large, this shows that $\mathbb{P}^v_{\text{VS}}(\phi, \Lambda)$ generates designs with $\approx k$ points. Dereziński et al. (2020) make such a statement precise for their variant of PVS using concentration inequalities. For $k$ large enough, they actually bound the average optimality criteria of designs generated by PVS, conditionally to their size being either equal to $k$, or lower than $k$ and completed with random i.i.d. points to reach size $k$. Instead, we focus on $A$ and D-optimality and we directly exploit the form of the Janossy densities of our PVS variant to get a tighter bound, which can be used for any value of $k \geq p$.

**Proposition 3.3** Let $X \sim \mathbb{P}^v_{\text{VS}}(\phi, \Lambda)$ such that $\nu(\Omega) = k$.

Then

$$\mathbb{E}[\det(\phi(X)^T \phi(X) + \Lambda)^{-1}||X| = k]\leq \frac{k^n(k-p)!}{k!} \frac{\det(G_v(\phi) + \Lambda)^{-1}}{1 + \frac{p-1}{k-p+1}}$$

(3.5)

with equality when $\Lambda = 0$.

**Proof** The proof is given in Sect. C. It relies on the fact that for any function $f : \Omega^k \rightarrow \mathbb{R}$,

$$\mathbb{E}[f(X)||X| = k] = \mathbb{E}[f(X)1_{|X|=k}] = \frac{1}{k!} \int_{\Omega^k} f(x) j_k(x) d\nu^k(x) = \frac{1}{k!} \int_{\Omega^k} j_k(x) d\nu^k(x).$$

This can be seen as a direct consequence of the definition of Janossy functions or (3.4). \qed

Proposition 3.3 shows that if we can find a solution $\nu_*$ to the relaxation (2.5) of the D-optimal design problem, a design sampled from $\mathbb{P}^v_{\text{VS}}(\phi, \Lambda)$, conditionally on having $k$ points, is in expectation at least a $\left(\frac{k-p}{k}\right)^k$-approximation of the (intractable) D-optimal design. In that sense, PVS can be seen as a rounding method in the sense of Pukelsheim (2006, Chapter 12.4): it takes as input a solution $\nu_*$ of the relaxed design problem, and outputs a design in the desired form of a set of $k$ points with an optimality certificate, here in expectation. The second denominator in the RHS of (3.5) results from a second-order approximation in the proof. We kept it to show the influence of the prior covariance matrix $\Lambda^{-1}$. At its lowest value, this term is equal to 1 in the non-Bayesian case ($\Lambda = 0$), and it grows to $(1 + \frac{p-1}{k-p+1})^{-1}$ as $\Lambda^{-1}$ goes towards 0. Our bound thus improves as the prior becomes more peaked.

To interpret the factor in Proposition 3.3, it is actually more convenient to measure the performance of an experimental design by its $D$-efficiency (Atkinson et al. 2007, Chapter 11.1).
\[
D_{\text{eff}}(X) = \left( \frac{\det(\phi(X)^T \phi(X) + \Lambda)}{\det(\phi(X_s)^T \phi(X_s) + \Lambda)} \right)^{1/p} \in [0, 1]. \tag{3.6}
\]

where \( X_\star \) is a D-optimal design. Since \( x \mapsto x^{-1/p} \) is a convex function, it comes

\[
\mathbb{E}_{\nu_{\text{VS}}(\phi, \Lambda)}[D_{\text{eff}}(X)|X| = k] \geq \left( \frac{k!}{(k-p)!k^p} \right)^{1/p} \times \left( 1 + \frac{p-1}{k-p+1} \left( 1 - \det(G_\nu(\phi)(G_\nu(\phi) + \Lambda)^{-1}) \right) \right)^{1/p} \times \left( \frac{\det(G_\nu(\phi) + \Lambda)}{\det(\phi(X_s)^T \phi(X_s) + \Lambda)} \right)^{1/p}.
\]

Since \( \det(G_\nu(\phi) + \Lambda) \geq \det(\phi(X_s)^T \phi(X_s) + \Lambda) \), designs sampled from \( \nu_{\text{VS}}(\phi, \Lambda) \) have, on average, a D-efficiency greater than

\[
\left( \frac{k!}{(k-p)!k^p} \right)^{1/p} \geq 1 - (p-1)/k.
\]

This improves upon the results of Dereziński et al. (2020) who found a D-efficiency of \( 1 - O(\sqrt{\ln(k)/k}) \).

Proposition 3.4

Let \( \nu \) be any finite measure on \( \Omega \), \( X \sim \nu_{\text{VS}}(\phi, \Lambda) \) and \( Y = (Y_1, \ldots, Y_k) \) where the \( Y_i \) are i.i.d. random variables with distribution \( \nu/\nu(\Omega) \). Then

\[
\mathbb{E}[\det(\phi(X)^T \phi(X) + \Lambda)^{-1} | |X| = k] \leq \mathbb{E}[\det(\phi(Y)^T \phi(Y) + \Lambda)^{-1}]\]

Now we give a bound for the A-optimality criterion of designs generated from the \( \nu_{\text{VS}}(\phi, \Lambda) \) distribution.

Proposition 3.5

Let \( X \sim \nu_{\text{VS}}(\phi, \Lambda) \) such that \( \nu(\Omega) = k \). We define

\[
m_0 := \max(\text{dim}(\text{Ker}(\Lambda)), 1).
\]

Then

\[
\mathbb{E} \left[ \text{Tr}((\phi(X)^T \phi(X) + \Lambda)^{-1}) | |X| = k \right] \leq \frac{k^{p+1-m_0} (k-p)!}{(k+1-m_0)!} \text{Tr}((G_\nu(\phi) + \Lambda)^{-1}). \tag{3.7}
\]

Proof

The proof is similar to Proposition 3.3 and can be found in Sect. E.

Proposition 3.5 shows that if we can find a solution \( \nu_\star \) of the relaxation (2.5) of the A-optimal design problem, a design sampled from \( \nu_{\text{VS}}(\phi, \Lambda) \), conditionally to having \( k \) points, is in expectation a \( \nu_{\text{VS}}(\phi, \Lambda) \)-approximation of the (intractable) D-optimal design. Defining A-efficiency as

\[
A_{\text{eff}}(X) := \frac{\text{Tr}((\phi(X_\star)^T \phi(X_\star) + \Lambda)^{-1})}{\text{Tr}((\phi(X)^T \phi(X) + \Lambda)^{-1})} \in [0, 1]. \tag{3.8}
\]

where \( X_\star \) is an A-optimal design, we get that designs sampled from \( \nu_{\text{VS}}(\phi, \Lambda) \) have, on average, an A-efficiency greater than \( \frac{k^{p+1-m_0} (k-p)!}{(k+1-m_0)!} \). In the special case where \( \Lambda = 0 \), the A-efficiency becomes \( k^{p+1}/k \), as was already shown in Dereziński et al. (2019, Theorem 2.9). This bound is actually sharp since it corresponds to the worst possible gap between the A-optimality criterion of the approximate optimal design and that of the true optimal design (Nikolov et al. 2019Theorem C.3). Unfortunately, in Bayesian linear regression with \( \Lambda \) invertible, the A-efficiency is \( \frac{k!}{(k-p)!} \), which is much larger. Still, as \( k \to \infty \), the A-efficiency is equivalent to \( 1 - p(p-1)/2k \). This improves upon the \( O(\sqrt{\ln(k)/k}) \) rate in Dereziński et al. (2020), although for small values of \( k \) the bound in Dereziński et al. (2020, Lemma 13) is tighter.

3.3 Efficient simulation of proportional volume sampling

When \( \Omega \) is finite, an algorithm has been proposed in Dereziński et al. (2020) with a sample time of \( O(np^2) \), where \( n \) is the cardinality of \( \Omega \), after a preprocessing cost of \( O(nd^2) \). This was done by showing that sampling from PVS boils down to sampling a determinantal point process (DPP) and a few additional i.i.d. points from \( \nu \). A similar result was shown for general design spaces \( \Omega \) when \( \Lambda = 0 \) in Dereziński et al. (2019). These results further highlight the links between these two families of point processes. DPPs are a large class of point processes formalized by Macchi (1975) as a fermionic analogue to photon detection in quantum optics. Since then, DPPs have been extensively studied in the literature, from random matrix theory to spatial statistics, machine learning and randomized numerical linear algebra; see, e.g., Hough et al. 2009; Lavancier et al. 2015; Kulesza and Taskar 2012; Dereziński and Mahoney 2021.
In this section, we extend the results of Dereziński et al. (2020) to a general Polish space. Additionally, for any $k \geq p$, we show that there is a natural rejection-free sampler for our PVS conditionally to the cardinality being $k$. First, we recall the definition of a determinantal point process on a general Polish set $\Omega$.

**Definition 3.6** Let $\Omega$ be a Polish set and $\nu$ be a measure on $(\Omega, B_\Omega(\Omega))$. Let $\psi_1, \ldots, \psi_k \in L^2(\Omega, \nu)$ be $k$ orthonormal functions, and let $\lambda_1, \ldots, \lambda_k \in [0, 1]$. We further define the function $K(x, y) := \sum_{i=1}^k \lambda_i \psi_i(x) \psi_i(y)$. Then the point process on $(\Omega, \nu)$ with correlation functions

$$
\rho_n(x_1, \ldots, x_n) = \det \left( \begin{array}{cccc} K(x_1, x_1) & \cdots & K(x_1, x_n) \\ \vdots & \ddots & \vdots \\ K(x_n, x_1) & \cdots & K(x_n, x_n) \end{array} \right)
$$

for all $n \geq 1$, is well defined. We call it the DPP with reference measure $\nu$ and kernel $K$, and denote it by $\text{DPP}(K, \nu)$.

Note that DPPs can be sampled by an algorithm due to Hough et al. (2009, Algorithm 18); see also Lavancier et al. (2015, Algorithm 1). Additionally, DPPs are defined through their correlation functions. In order to compare them to PVS, we first get an explicit expression of the correlation functions of proportional volume sampling.

**Proposition 3.7** The $n$th-order correlation function $\rho_n$ of the $\text{PVS}_\nu(\phi, \Lambda)$ point process is well defined for all $n \in \mathbb{N}$, and reads

$$
\rho_n(x_1, \ldots, x_n) = \frac{\det(G(\phi) + \Lambda + \phi(x)^T \phi(x))}{\det(G(\phi) + \Lambda)}. 
$$

**Proof** The proof is shown in Sect. F. It is mainly based around the usual identity linking correlation functions and Janossy densities; see Daley and Vere-Jones (2003, Lemma 5.4.III)

$$
\rho_n(x_1, \ldots, x_n) = \sum_{m \geq 0} \frac{1}{m!} \int_{\Omega^n} \prod_{j=1}^n j \int_{\Omega} j \nu(x, y) d\nu^m(y).
$$

As a consequence, we get that PVS can be expressed as the superposition of a DPP and a Poisson point process. This is to be put in parallel to Dereziński et al. (2019, Theorem 2.4) which shows a similar result in the non-Bayesian case.

**Proposition 3.8** Let $\nu$ be a finite measure on $\Omega$. Let $X$ be a DPP with kernel

$$
K(x, y) := \phi(x)(G_\nu(\phi) + \Lambda)^{-1} \phi(y)^T
$$

and reference measure $\nu$, and let $Y$ be an independent Poisson point process with intensity $\nu$. Then the distribution of the superposition $X \cup Y$ is $\text{PVS}_\nu(\phi, \Lambda)$.

**Proof** The proof is given in Sect. G. We rewrite the correlation functions of PVS as

$$
\det(I_n + (K(x_i, x_j))_{1 \leq i, j \leq n}),
$$

which corresponds to the correlation functions of the superposition of a DPP and a Poisson point process. We remark that the DPP with kernel $K$ is well defined since, as it is shown in Proposition 3.9, $K$ can be written as $\sum_{i=1}^k \lambda_i \psi_i(x) \psi_i(y)$ for some orthogonal functions $\psi_i$, while the $\lambda_i$ are the eigenvalues of the matrix

$$
G_\nu(\phi)^{1/2}(G_\nu(\phi) + \Lambda)^{-1}G_\nu(\phi)^{1/2},
$$

which are all in $[0, 1]$.

Note that when $\nu$ is not a diffuse measure, the Poisson point process with intensity $\nu$ can generate the same point multiple times. In this case, $X$ and $Y$ are understood as multisets and their superposition is understood as a multiset union. Now, in order to use this result to simulate PVS, we need a way to simulate the DPP on $L^2(\Omega, \nu)$ with kernel

$$
K(x, y) := \phi(x)(G_\nu(\phi) + \Lambda)^{-1} \phi(y)^T.
$$

This can be done if the spectral decomposition of the associated integral operator on $L^2(\Omega, \nu)$ is available (Hough et al. 2009). When $\Omega$ is finite, this is equivalent to finding the spectral decomposition of a matrix. In the general case, we can still diagonalize the relevant operator as long as $G_\nu(\phi)$ is invertible.

**Proposition 3.9** Let $\nu$ be a finite measure on $\Omega$ such that $G_\nu(\phi)$ is invertible. We write the spectral decomposition of the matrix

$$
G_\nu(\phi)^{1/2}(G_\nu(\phi) + \Lambda)^{-1}G_\nu(\phi)^{1/2}
$$

as $P^T D P$, where $D = \text{diag}(\lambda_1, \ldots, \lambda_p)$, and we define the functions

$$
(\psi_1(x), \ldots, \psi_p(x)) := (\phi_1(x), \ldots, \phi_p(x))G_\nu(\phi)^{-1/2} P.
$$

Then, the functions $\psi_i$ are orthonormal in $L^2(\Omega, \nu)$ and

$$
K(x, y) = \sum_{i=1}^p \lambda_i \psi_i(x) \psi_i(y).
$$

**Proof** We remark that, by definition,

$$
G_\nu(\psi) = P^T G_\nu(\phi)^{-1/2} G_\nu(\phi) G_\nu(\phi)^{-1/2} P = I_p.
$$
so that \((\psi_1, \ldots, \psi_p)\) is an orthonormal family of \(L^2(\Omega, \nu)\). Moreover, we can write
\[
\sum_{i=1}^{p} \lambda_i \psi_i(x) \psi_i(y) = \psi(x) D \psi(y)^T \\
= \phi(x) G_v(\phi)^{-1/2} D P D \phi(x) G_v(\phi)^{-1/2} \phi(y)^T \\
= \phi(x)(G_v(\phi) + \Lambda)^{-1} \phi(y)^T = K(x, y).
\]

A direct consequence of Proposition 3.8, proved in Sect. H, is an explicit expression of the average number of points generated by PVS.

**Corollary 3.10** Let \(X \sim \mathbb{P}_{\nu, k}(\phi, \Lambda)\), then
\[
\mathbb{E}[|X|] = \nu(\Omega) + \text{Tr}((G_v(\phi) + \Lambda)^{-1} G_v(\phi)).
\]

Proposition 3.8 also yields a natural sampling algorithm for proportional volume sampling as a superposition of an independent DPP and a Poisson point process. For ease of reference, the pseudocode of the algorithm in Proposition 3.8 is given in Fig. 1.

In practice, it is customary to fix the cardinality \(k\) of an experimental design in advance. We thus would like to condition the two point processes in the superposition of Proposition 3.8 on their union having total size \(k\). This is possible using a well-known decomposition of the DPP \(X\) with kernel \(K(x, y) := \sum_{i=1}^{p} \lambda_i \psi_i(x) \psi_i(y)\) as a mixture of projection DPPs (Hough et al. 2009). More precisely, to sample \(X\), one first samples \(p\) independent Bernoulli random variables \(I_i \sim \text{Ber}(\lambda_i)\). Then, conditionally on these Bernoullis, one uses the chain rule to sample from the projection DPP with kernel \(\sum_{i=1}^{p} I_i \psi_i(x) \psi_i(y)\), which yields \(\sum_{i=1}^{p} I_i\) points almost surely. As a consequence, conditioning PVS to having size \(k\) can be performed by first sampling independent \(I_i \sim \text{Ber}(\lambda_i)\) and an independent Poisson random variable \(N\) with parameter \(\nu(\Omega)\), conditioned on \(N + \sum_{i=1}^{p} I_i = k\). Then, the union of a DPP with kernel \(\sum_{i=1}^{p} I_i \psi_i(x) \psi_i(y)\) and \(N\) i.i.d. points with distribution \(\nu/\nu(\Omega)\) on \(\Omega\) has the same distribution as a sample from PVS conditioned to be of size \(k\). The pseudocode of this algorithm is given in Fig. 2.

The costly steps in Fig. 2 are the one-time computation of the Gramian and its square root, and the DPP sampler. For the Gramian, when exact computation is not possible, one has to resort to numerical integration, possibly even Monte Carlo methods (Robert and Casella 2004), depending on how complicated the set \(\Omega\) is, the dimension of the design space, and the regularity of the functions \(\phi_i\) and the reference measure \(\nu\). Getting an approximate Gramian makes the overall algorithm heuristic. For the DPP sampler, the number of operations is at least cubic in the number \(p\) of points in the DPP sample. The at least corresponds to the number of rejections in the \(p\) rejection samplers involved in the chain rule for projection DPPs; see Gautier et al. (2019a) for empirical investigations on these rejection numbers. The rest of the algorithm requires \(O(p^3 + dk)\) steps when PVS is conditioned on having size \(k\), and \(O(p^3 + d \nu(\Omega))\) on average otherwise. Overall, while the cost of the algorithm in Figs. 1 and 2 is at least cubic in \(p\), this is not a obstacle in practice. Indeed, experimental design is used in situations where obtaining the regression labels takes time or money, so that spending a few minutes finding a good design is considered negligible.

### 3.4 A numerical illustration of PVS for experimental design

We illustrate the theoretical results from Sect. 3.2 on a simple example. We consider the design space \(\Omega = [0, 1]^2\) and \(k = p = 10\), with the regression functions \(\phi_i\) being the \(p\) bivariate polynomials of degree \(\leq 3\), renormalized so that \(\Vert \phi_i \Vert_{L_2(\Omega)} = 1\) for all \(i\). For our example, we considered the case where the inverse prior covariance matrix \(\Lambda\) is either

---

**Algorithm 1**: A sampler for proportional volume sampling; see Proposition 3.8.

1. Compute the Gramian matrix \(G_v(\phi)\).
2. Compute the spectral decomposition \(G_v(\phi)^{1/2}(G_v(\phi) + \Lambda)^{-1} G_v(\phi)^{1/2} = P^T D P\) where \(D = \text{diag}(\lambda_1, \ldots, \lambda_p)\).
3. Compute the orthonormalized functions \((\psi_1, \ldots, \psi_p) := (\phi_1, \ldots, \phi_p) G_v(\phi)^{-1/2} / \| \phi_i \|_{L_2(\Omega)}\).
4. Sample \(X\) from a DPP\((k, \phi)\) distribution, where \(K(x, y) = \sum_{i=1}^{p} \lambda_i \psi_i(x) \psi_i(y)\).
5. Sample \(Y\) from a Poisson point process with intensity \(\nu\).
6. Return \(X \cup Y\).

**Algorithm 2**: A sampler for proportional volume sampling conditioned on having a given cardinality \(k\); see Proposition 3.8.

1. Compute the Gramian matrix \(G_v(\phi)\).
2. Compute the spectral decomposition \(G_v(\phi)^{1/2}(G_v(\phi) + \Lambda)^{-1} G_v(\phi)^{1/2} = P^T D P\) where \(D = \text{diag}(\lambda_1, \ldots, \lambda_p)\).
3. Compute the orthonormalized functions \((\psi_1, \ldots, \psi_p) := (\phi_1, \ldots, \phi_p) G_v(\phi)^{-1/2} / \| \phi_i \|_{L_2(\Omega)}\).
4. Sample \(p\) independent Bernoullis \(I_i \sim \text{Ber}(\lambda_i)\) and an independent Poisson random variable \(N\) with parameter \(\nu(\Omega)\), conditioned on \(N + \sum_{i=1}^{p} I_i = k\).
5. Sample \(X\) from a DPP\((k, \phi)\) distribution, where \(K(x, y) = \sum_{i=1}^{p} I_i \psi_i(x) \psi_i(y)\).
6. Sample \(N\) points \(Y = \{Y_1, \ldots, Y_N\} \subset \Omega\) i.i.d. with distribution \(\nu/\nu(\Omega)\).
7. Return \(X \cup Y\).
In short, when looking at Fig. 3 from left to right, we move from an informative prior to a frequentist setting. Simultaneously, we move from loose bounds and parity between PVS and iid designs to tighter bound for D-efficiency and big differences in performance between PVS and iid.

Finally, we show in Fig. 4 a few sample designs from the various distributions considered in Fig. 3c. PVS naturally forces points apart from each other, showing a so-called repulsive behavior compared to i.i.d. designs. This well-spreading is also characteristic of the support of optimal designs. We also observe that the density \( \nu \) for D-optimality, shown in Fig. 7c, takes a large value on the vertices of the square design space, a medium value on the edges and a low value in the middle. Because of the inherent repulsiveness of the PVS distribution, this means that PVS designs will usually have a point close to each vertex, a few well-spread points close to the edges, and a few points in the center of the design space, far apart from the rest. This behavior is very similar to the one of the D-optimal design shown in Fig. 4d.

We also quickly explored the D-efficiency of designs generated by PVS for a cubic regression on \([0, 1]^2\) with an added qualitative factor; see Appendix 1. We obtained essentially the same results as in Fig. 3, which confirms that PVS is not affected by having to deal with both quantitative and qualitative variables.

Overall, PVS is tractable, mathematically elegant, and comes with guarantees in the form of bounds on the expected A- and D-criteria. It also has the advantage of working for nearly any design space, any family of regression functions and in both the Bayesian and non-Bayesian setting. A simple empirical investigation confirms that it generates designs with typically lower A- and D-optimality criteria than i.i.d. sampling, and optimizing the reference measure further helps. On the negative side, the theoretical bounds can be loose. Furthermore, as seen in Fig. 3, even taking the best of thousands of PVS samples does not yield a design arbitrarily close to an optimal design. From a practical point of view, this is slightly disappointing. This is why we now investigate a different way of using PVS, namely embedding it into an iterative search algorithm for optimal designs.

4 Turning discrete PVS into a global search heuristic

In this section, we first review two standard algorithmic templates to find approximate A and D-optimal designs in the most general setting. Then, we introduce a new heuristic, named DOGS, that relies on the discrete proportional volume sampling of Nikolov et al. (2019) and Dereziński et al. (2018) to make global moves across a generic design space \( \Omega \). Finally, we investigate the numerical performance of DOGS against standard methods.
Statistics and Computing (2023) 33:29

Fig. 3 Violin and boxplots of the Bayesian A-efficiency and D-efficiency of 2000 random designs either sampled i.i.d. from \( \nu \) or from \( \mathbb{P}^{\nu}_{\text{VS}}(\phi, \Lambda) \). The density \( \nu \) is either uniform on \( \Omega \) or the optimized density shown in Fig. 7. Dash-dotted lines show the bounds of Proposition 3.3 (top) and the non-Bayesian bound of Proposition 3.5 (bottom); see Sect. 3.4 for details.

4.1 Standard heuristics for optimal designs

First, the local search algorithm (LSA) starts with a random design drawn from some initial distribution as its current best \( X_{\text{best}} \). Then, at each iteration and until a stopping criterion is met, LSA generates another random design \( Y \) from a proposal distribution supported in a neighborhood of \( X_{\text{best}} \), and sets \( X_{\text{best}} \) to \( X \) if \( X \) has a smaller optimality criterion. An iteration of LSA is computationally cheap, but this is at the cost of a slow empirical convergence and a tendency to get stuck in local minima. We refer to Fang et al. (2006, Chapter 4.2) for a description of LSA and variants, including simulated annealing. These variants add a few parameters that need to be tuned, and in our later experiments did not qualitatively change the behavior of LSA. We thus focus on the vanilla LSA algorithm.

Another standard algorithm is the exchange method (ExM; Fedorov 1972). Starting with a random initial design, ExM iteratively minimizes the optimality criterion over each point of the design, keeping the rest of the design fixed, until a stopping criterion is met. The basic operation is thus

\[
\arg \min_{x \in \Omega} h(\phi(X \cup \{x\})^T \phi(X \cup \{x\}) + \Lambda)
\]

for any design \( X \subset \mathbb{R}^d \) with cardinality \( k - 1 \). When \( \Omega \) is a (small) finite set, this can be done by exhaustive enumeration or a grid search. When \( \Omega \) is continuous, however, approximating (4.1) can be mathematically difficult or computationally heavy, depending on the dimension and shape.
of $\Omega$, as well as the regularity of the regression functions $\phi_i$. ExM is also prone to converging to local minima, and it is usually recommended to take the best result out of several restarts. Variants of ExM are presented in Pronzato and Pázman (2013) Chapter 9.2.1 but, as with LSA, we focus here on the vanilla ExM as a representative of its class.

Finally, in many applications of experimental design in continuous $\Omega$, there is some knowledge or intuition of points that are likely to be in the support of an optimal design. This is particularly common when working with design spaces with a simple shape such as spheres, simplices or cubes, and for low-dimensional polynomials as regression functions; see, e.g., Atkinson et al. (2007). In these cases, one usually forces search heuristics like ExM to focus on a finite set $C$ of well-chosen candidate points. One approach is to run a discrete optimization algorithm such as ExM on the finite set $C$, ensuring that the support of the solution is in $C$. When no useful $C$ is known and $\Omega \subset \mathbb{R}^d$, it is also common practice to run ExM on the “agnostic” finite set formed by the intersection of $\Omega$ with a regular grid.

4.2 Discrete optimization within global search (DOGS)

In spite of its near-optimality guarantees, proportional volume sampling as introduced in Sect. 3.2 is likely not to meet the needs of practitioners. Indeed, as shown in Fig. 3, even taking the best design out of thousands of PVS realizations does not get us close to an optimal design, while simple heuristics like ExM can output designs with smaller optimality criterion at a comparable cost. In this section, we propose a search heuristic based on discrete volume sampling but different from PVS. The rationale is to use volume sampling to propose global moves across $\Omega$, thus avoiding getting stuck in local minima.

The pseudocode of our DOGS heuristic is given in Fig. 5. We start from an initial random design $X_{\text{best}}$ of $k$ points drawn from any initial distribution at hand. Then, at each iteration, we find the optimal design of $k$ points among the union of the current $X_{\text{best}}$ and a random subset $X_{\text{new}} \subset \Omega^{k'}$ drawn from a proposal distribution, with a user-defined cardinality $k' \in \mathbb{N}$. This step corresponds to finding a finite optimal design of $k$ points among $k + k'$, and can be solved either exactly, by enumeration, or approximately, by a search heuristic like ExM. We choose to solve it approximately, but rather than running a costly ExM at each iteration of DOGS, we found it empirically more efficient to run a cheaper subroutine. In line with Sect. 3.2, we actually propose to sample from the PVS distribution (3.1), over the set $X_{\text{best}} \cup X_{\text{new}}$, and with reference measure $\nu$ minimizing the continuous relaxation (1.2).

The choice of the proposal distribution from which $X_{\text{new}}$ is drawn in Step 2 of Fig. 5 is where expert knowledge can be built in the algorithm. In particular, the proposal can include candidate points $C$, as explained in Sect. 4.1. The remaining points can be drawn from any distribution over $\Omega$. We typically draw i.i.d. samples from the uniform distribution over $\Omega$, since for the design spaces we have found in the literature, the uniform distribution is amenable to rejection sampling. More space-filling proposals are also possible, like a randomly perturbed grid or a randomly shifted Sobol or Halton sequence (Dick and Pilichshammer 2010), but we have found that the improvement over the uniform proposal is minor, and the experiments in Sect. 4.3 thus use i.i.d. uniform draws. Finally, a large value of the number $k'$ of points in $X_{\text{new}}$ makes the algorithm converge in fewer iterations, but each iteration is more costly due to the increase in complexity of the optimization problem (4.2). In short, like for the $\sigma$ parameter of LSA, we recommend tuning $k'$ manually using a few initial short runs. As for the complexity of the PVS sampling step, we refer to Sect. 3.3.

The computational bottleneck of DOGS is the underlying convex optimization subroutine (4.2). The complexity of the latter heavily depends on the optimality criterion and the specific convex optimization algorithm that is used, but it is hard to make a definite scaling argument; we thus show CPU times in Table 1. Additionally, unlike PVS in Sect. 3, it is difficult to produce any mathematical result on the output of DOGS. We thus focus on evaluating its empirical performance.

---

**Algorithm 3:** Discrete optimization within global search

1. Start with a random initial design $X_0 \in \Omega^k$ and set $X_{\text{best}} = X_0$. 
2. Choose a random design $X_{\text{new}}$ in $\Omega^{k'}$ for some $k' \in \mathbb{N}$. 
3. Let $\{x_1, \cdots, x_{\ell}\}$, $\ell \leq k + k'$, be the set of unique points in $X_{\text{new}}$ and $X_{\text{best}}$. Compute a solution $(p_i)_{1 \leq i \leq \ell}$ of the convex optimization problem

$$\begin{align*}
\arg\min h \left( \sum_{i=1}^{\ell} p_i \phi(x_i)^T \phi(x_i) + \Lambda \right) \\
\text{s.t. } \sum_{i=1}^{\ell} p_i = k \text{ and } p_i \geq 0 \forall i \in \{1, \cdots, k\}. 
\end{align*}$$

4. Sample $Y$ from the $P_{\nu,\lambda}^{\phi, \Lambda}(\phi, \Lambda)$ distribution over the set $\{x_1, \cdots, x_{\ell}\}$ where

$$\nu = \sum_{i=1}^{\ell} p_i \delta_{x_i}.$$ 

5. If $h(\phi(Y)^T \phi(Y) + \Lambda) < h(\phi(X_{\text{best}})^T \phi(X_{\text{best}}) + \Lambda)$ then set $X_{\text{best}} = Y$. 
6. If some stopping criterion is reached, return $X_{\text{best}}$. Otherwise, go back to step 2.
Finally, a potential alternative for step 4 in Fig. 5 is to replace PVS by the efficient rounding method of Pukelsheim and Rieder (1992), which “rounds” the solution of (4.2) into a design of size $k$. In terms of computation, the bottleneck remains the convex relaxation of the optimal design problem, and the computational time for one iteration of DOGS does not significantly change whether using PVS or efficient rounding. In practice, we observed that efficient rounding is preferable to PVS when looking for an optimal design whose size is large compared to the number of regression functions $p$. On the contrary, in the common case where $k$ is close to $p$, the repulsiveness among design points that comes with PVS allows for lower A- and D-criteria for the same amount of computation as efficient rounding. We shall see examples of the comparative behavior of these two DOGS variants in Sect. 4.3.

### 4.3 Numerical results

In this section, we compare the performances of LSA, ExM and DOGS in three applications, of increasing difficulty. The first example, coming from Atkinson et al. (2007, Example 16.3), corresponds to a three-component mixture design with quadratic constraints for polynomial regression. The second example is drawn from Grove et al. (2004); Woods et al. (2003) and illustrates regressing on a functional basis that adds B-Splines to polynomials. The third example is an engineering application from Piepel et al. (2019), and corresponds to an eight-component mixture design with a combination of linear and nonlinear constraints for polynomial regression.

We showcase several more examples, mostly taken from the textbook Atkinson et al. (2007), in a companion Jupyter Notebook provided as supplementary material.¹

We focus here on the non-Bayesian setting ($\Lambda = 0$), since it is the most common setting in the applied literature. The interested reader can find some toy examples in the Bayesian setting in the supplementary Jupyter Notebook. In our experience, the relative performances of the algorithms compared here remain unchanged in the Bayesian setting. Also, as we illustrated in Fig. 3, finding good designs is easier in the Bayesian setting, especially when the prior covariance matrix takes small values.

For all algorithms, initial designs are chosen using i.i.d. uniform draws over $\Omega$. In LSA, the proposal is generated by adding an i.i.d. Gaussian perturbation to each point in the current design, with a common, manually tuned standard deviation $\sigma$. If the perturbation of a point makes it leave the design space, we leave the original point as is in the proposed $X_{\text{new}}$. For ExM, the internal optimization subroutine is done using L-BFGS-B in *scipy* (Virtanen et al. 2020). In DOGS, the random design $X_{\text{new}}$ is sampled uniformly in $\Omega^{k'}$, unless specified otherwise. The convex optimization in Step 3 of Fig. 5 is carried out using a generic solver from the library *cvxopt* (Andersen et al. 2012) for D-optimality, and with the solver of semi-definite programs (SDPs) of the same library for A-optimality. Indeed, we follow Boyd and Vandenberghe (2004, Chapter 7.5.2) and cast the A-optimality problem (4.2) as an SDP. Finally, when step 4 of Fig. 1 is done using the PVS distribution, it is sampled using Algorithm 1, with the underlying DPP drawn using the *DPPy* library (Gautier et al. 2019b).

#### 4.3.1 Three-component mixture design with quadratic constraints

In (Atkinson et al. 2007, Example 16.3), the authors searched for a D-optimal design in

$$\Omega = \{(x_1, x_2) \in \mathbb{R}^2 \mid 0 \leq x_1 + x_2 \leq 1, \\
-4.062x_1^2 + 2.962x_1 + x_2 \geq 0.6075 \\
\text{and} \quad -1.174x_1^2 + 1.057x_1 + x_2 \leq 0.5019\}.$$ 

The regression functions $\phi_i$ are the 6 multivariate polynomials of $\mathbb{R}^2$ with degree $\leq 2$. The authors used a finite optimization method over a grid covering the design space, and obtained a six-point approximate design with equal weights. We thus chose $k = 30$, to make their approximate design a true design, i.e. with integer weights, without requiring a rounding step. We ran a few iterations of DOGS with various values of $k'$, and set for $k' = 50$. Similarly, we set the standard deviation of the Gaussian perturbations in LSA to $\sigma = 0.01$.

We present in Fig. 6a the evolution of the log D-optimality criterion of the current best design $X_{\text{best}}$ across iterations, for 200 runs of each algorithm. In LSA, an *iteration* corresponds to the generation of a random design and the comparison with the current best design. For ExM, it is one minimization of (4.1); for DOGS, an iteration is one execution of Steps 2–6 in Fig. 5. The solid line represents the median log-D-optimality criterion of each algorithm, while the boundary of the shaded area corresponds to the 5th and 95th percentile. The log D-optimality criterion of the design in Atkinson et al. (2007) and the one obtained by the exact algorithm of De Castro et al. (2019) are shown as, respectively, a dashed and a dashdotted line. We also show in Table 1 the runtime of each algorithm on a recent laptop.

We observe that DOGS outperforms both LSA and ExM in this case: DOGS only needs a few hundred iterations, totaling about 10 seconds, to find a better D-optimal design than the one in Atkinson et al. (2007). In fact, DOGS settles in the same area of $\Omega$ as the design of Atkinson et al. (2007), but is allowed to fine tune its result by not being limited to a

¹ Python code available at https://github.com/APoinas/Optimal-design-in-continuous-space
grid. However, this example should now be considered as easy, in the sense that the Lasserre hierarchy of De Castro et al. (2019) terminates in about the same time as DOGS and outputs an actual optimal design, with criterion value shown in Fig. 6a.

We can easily make the problem harder, though, by considering polynomials of higher degree. On this example, we observed that the Lasserre hierarchy failed to terminate for degrees larger than 2. This is the regime where search heuristics become useful. For instance, taking the regression functions $\phi_i$ to be the 15 multivariate polynomials of $\mathbb{R}^2$ with degree $\leq 4$, we show the results of LSA, ExM, and DOGS in Fig. 6b. We also compare these results with the average D-optimality criterion of designs obtained by a discrete exchange method on the set $\Omega \cap 0.01\mathbb{Z}^2$ of 736 candidate points, to mimic the approach of Atkinson et al. (2007). The results are shown in Fig. 6b. The rankings are similar, with DOGS finding the best solution in a small number of iterations. However, PVS cannot be credited for the performance gain in this example, where $k = 30 \gg p = 6$. Indeed, replacing PVS by efficient rounding in DOGS yields a further gain in performance.

4.3.2 Multifactor B-Spline Mixed Models

This example is inspired by Grove et al. (2004). The problem is to find a relationship between several features of a car engine, like maximum brake-torque timing (MBT), and three factors: the engine speed ($x_1$), its air-fuel ratio ($x_2$) and its load ($x_3$). The authors chose to model the relationship between MBT and $x_2$ and $x_3$ as a cubic polynomial, while the dependency of MBT in the variable $x_1$ is modeled as a maximally smooth cubic B-spline basis with three knots. We denote this basis by $\{B_1, \ldots, B_7\}$ for the remainder of this paper. Using all possible products of B-spline and polynomials would mean considering 112 basis functions of the form

$$\phi : (x_1, x_2, x_3) \mapsto B_i(x_1)x_2^\alpha x_3^\beta,$$

where $i \in \{1, \ldots, 7\}$, $\alpha \in \{0, \ldots, 3\}$ and $\beta \in \{0, \ldots, 3\}$. The authors chose to reduce this regression basis to a smaller one of $p = 31$ functions detailed in Grove et al. (2004, Equation (6)). The size of the optimal design searched is $k = 55$. The design space is unfortunately not specified, but it is alluded that its shape is complicated due to various combinations of $x_1$, $x_2$ and $x_3$ being either unphysical or potentially damaging to the engine. To illustrate the properties of the different algorithms, we use both a simple and a more complex design space.

We first consider $\Omega = [0, 1]^3$, with B-Spline knots are located at 0.25, 0.5 and 0.75. The location of the knots did not have any significant impact on the results. A set of 28 candidate points is suggested in Woods et al. (2003), namely

$$C := \{(x_1, x_2, x_3) \in \Omega, \quad x_1 \in \{\arg\max B_i, 1 \leq i \leq 7\}, \quad x_2, x_3 \in \{0, 1\}\}.$$

To investigate the impact of candidate points on DOGS, we compare a version with a uniform proposal over $\Omega^5$, to a version where the proposal is made of the union of $C$ and a uniform draw over $\Omega^2$. We compare to LSA with a manually tuned $\sigma = 0.01$, ExM, and a discrete ExM on the 700 candidate points

$$C' := \{(x_1, x_2, x_3) \in \Omega, \quad x_1 \in \{\arg\max B_i, 1 \leq i \leq 7\}, \quad x_2, x_3 \in \{0, 1/9, 2/9, \cdots, 1\}\}$$

as suggested in Woods et al. (2003). We present in Fig. 6c the evolution of the log D-optimality criterion of $X_{\text{best}}$ with respect to the iteration number, for 200 runs of each algorithm, as well as the average log D-optimality criterion for discrete ExM. We also present in Table 1 the runtime of each algorithm on a recent laptop. LSA performs worst, while both versions of ExM tie in converging to what we believe is close to the optimal criterion. DOGS again yields a quickly decreasing criterion, but its curve plateaus higher than ExM. Adding candidate points to DOGS lowers the plateau, but still not to the level of ExM. Additionally, replacing PVS in DOGS by efficient rounding does not improve results in this case. Both yield very similar results on average, but using PVS yields a significant reduction of variance of the D-optimality criterion.

A tentative argument to explain the success of discrete ExM in this example is the small-dimensional cubic design space, which is particularly amenable to a discretization by a finite grid. Similarly, for continuous ExM, the shape of $\Omega$ is natively handled by the L-BFGS-B optimization. An interesting note is that discrete ExM on an agnostic grid of 1000 candidate points $[0, 1] \cap \frac{1}{7}\mathbb{Z}^3$ does not come close to its performance when including candidate points, which confirms the candidate point suggestions of Woods et al. (2003). In a second variant of the same experiment, we now make the design space more complicated, as the real design space in Grove et al. (2004) is suggested to be. We consider

$$\Omega' = B((x_c, x_c, x_c, x_c)) \cup B((1-x_c, 1-x_c, 1-x_c, x_c),$$

where $x_c := \frac{3 - \sqrt{3}}{4}$, (4.2)

and where $B(P, R)$ denotes the ball centered at $P$ with radius $R$. This design space corresponds to two tangent Euclidean balls inside the cube $[0, 1]^3$, meeting at the point $(1/2, 1/2, 1/2)$. We did not include any candidate point in this case, and the proposal in DOGS is uniform over $\Omega^{50}$. 

© Springer
Fig. 6  $\log h_D(\phi(X_{\text{best}})^T \phi(X_{\text{best}}))$ (y-axis) vs. iteration number (x-axis) for 200 runs of each algorithm. The average log-D-optimality criterion of designs obtained by 50 runs of a discrete exchange method is shown by a dashed line.

| Table 1 | Average computation times on a laptop for one realization of discrete ExM and 1000 iterations of DOGS, LSA and ExM |
|--------|-------------------------------------------------|
|        | DOGS    | LSA    | ExM    | Disc. ExM |
| Example 4.3.1: degree $\leq 2$ | 47.9s  | 0.951s | 30.2s  |
| Example 4.3.1: degree $\leq 4$ | 51.5s  | 0.940s | 12.6s  | 17.8s  |
| Example 4.3.2: $\Omega = [0, 1]^3$ | 121s  | 4.14s  | 32.6s  | 260s   |
| Example 4.3.2: $\Omega'$ defined in (4.2) | 96.4s  | 2.67s  | 59.6s  | 268s   |
| Example 4.3.3 | 178s  | 4.63s  | 130s  | 184s    |
The discrete version of ExM uses the 719 points of \( \Omega \cap \mathbb{Z}^3 \). The results are shown in Fig. 6d.

As expected, the performance of both the continuous and discrete version of ExM suffers from the more complex design space, while DOGS seems more robust to the change of design space, outperforming the best overall result of any other approach in about 100 iterations. Using PVS in DOGS, for this setting, gave significantly better designs than efficient rounding for the first 500 iterations, while efficient rounding takes the lead beyond that number. In particular, we can see that continuous ExM shows a huge variance in its results and, even if they don’t appear in the quantiles of Fig. 6c, there were a few runs that got stuck in local optima with a criterion larger than 150, likely due to a bad initial design. This confirms the recommendation in Pronzato and Pázman (2013) to use a restart strategy. DOGS did not suffer from this kind of issue.

So far, DOGS has proved to be a robust algorithm, that fares particularly well when \( \Omega \) has a complex shape or is hard to discretize. The typical behavior of DOGS it to quickly lower its criterion before plateauing. This suggests a hybrid method, where DOGS is used until its improvement in criterion is deemed to be too small, at which point one switches to a local search like LSA. We leave the investigation of such variants to future work.

### 4.3.3 Nepheline crystallization in high-level nuclear waste glass

For our last example, we increase the dimensionality to an eight-component mixture design. (Piepel et al. 2019) study the propensity of nepheline crystals (NaAlSiO₄) to appear during the fabrication of glass made to contain nuclear waste, in relation to the proportion of the 8 components of the glass. This leads to a 7-dimensional problem. Various linear and nonlinear constraints make the design space \( \Omega \) quite complicated. No regression is mentioned in the paper; the authors rather focus on producing a space-filling design, meaning a general purpose design with points well spread over \( \Omega \). For our experiments, we take the regression functions \( \phi_i \) to be the 36 multivariate polynomials of \( \mathbb{R}^7 \) with degree \( \leq 2 \) and we search for an optimal design with 36 points, the same as the number of regressing functions.

We identified 24 points at the intersection of the linear and nonlinear constraints and use them as candidate points \( C \). We thus compare DOGS with a uniform proposal over \( \mathcal{L}_{100}^{74} \) to DOGS with a proposal taken as the union between \( C \) and a uniform draw over \( \mathcal{L}_{76}^{74} \). We still compare to LSA, with \( \sigma = 0.001 \), ExM, and its discrete version on the set \( C \cup (\Omega \cap \frac{1}{14} \mathbb{Z}^7) \) of 309 points.

Again, we show in Fig. 6c the evolution of the log D-optimality criterion of \( X_{\text{best}} \) for 200 runs of each algorithm with respect to iteration number and the average log D-optimality criterion of discrete ExM. We also show in Table 1 the runtime of each algorithm on a recent laptop. Again, using candidate points significantly improves the performance of DOGS. Additionally, using efficient rounding in DOGS makes the algorithm almost useless in this example. This rather extreme behavior confirms our claim that, when the design cardinality is close to the number of regressing functions (here \( k = p = 36 \)), the repulsiveness in PVS is to be preferred to efficient rounding. We also note that, despite clearly outperforming ExM and LSA, DOGS struggles to get any close to optimality past its initial sharp decrease. Our reference line in red is a discrete exchange method on a relatively coarse grid, and it finds a significantly better design. In our experience, this is a recurrent issue with DOGS in large dimensions. Using more space-filling proposals, like continuous PVS in (3.1) or a Sobol sequence (Dick and Pillichshammer 2010) helps a little, but not enough to outperform the discrete ExM. We conjecture that a well-tuned combination of DOGS and local moves could take the best of both worlds, but leave this to future work.

### 5 Conclusion

Our goal was to investigate some of the statistical implications of recent advances on volume sampling for discrete Bayesian optimal design.

We first turned the Bayesian version of finite proportional volume sampling (Nikolov et al. 2019; Dereziński et al. 2018) introduced by Dereziński et al. (2020) into a general distribution over any Polish space. Using point process arguments, we showed that this generalization preserves the property of giving unbiased estimates of the inverse information matrix and its determinant. Additionally, we proved approximation guarantees for the A-efficiency and D-efficiency of designs sampled from general proportional volume sampling, conditionally to having a fixed size. Through a connection with determinantal point processes, we highlighted that our general PVS can be sampled in polynomial time. We also showed that the same algorithm can easily be modified to sample random designs from PVS conditioned on having a fixed size, without using rejection sampling. This makes PVS a natural tool to extract experimental designs from the solution of the classical convex relaxation (2.5) of the optimal design problem. However, in spite of its mathematical and methodological support, we found that on simple continuous problems, PVS can be outperformed in practice by simple search heuristics.

Our investigation of the impact of PVS then took a more practical turn as we introduced DOGS, a search heuristic that combines random sampling and either discrete PVS or the so-called efficient rounding method of Pukelsheim and Rieder (1992), to make global moves across a generic design space.
Although it is costlier than popular alternatives, DOGS shines when $\Omega$ has a complicated shape and the dimension remains small, and the performance of DOGS with PVS is moreover robust to changes in $\Omega$ or the basis functions. We believe that this robustness makes DOGS an interesting addition to the practitioner’s toolbox. Interestingly, the repulsiveness in PVS only helps DOGS when the size of the design is not too large compared to the number of basis functions. This suggests theoretically analyzing point processes built using efficient rounding.

One negative point of DOGS is that when the ambient dimension $d$ is large ($\geq 5$), DOGS fails to find designs as good as a simple discrete exchange method over a reasonable set of candidate points. Overall, the general behavior of DOGS is to quickly converge towards a design with a good optimality criterion, and then to start plateauing. This suggests alternating DOGS with a local search algorithm as soon as the plateauing behavior appears. Another possible idea is to combine search heuristics using, e.g., multi-armed bandits.

Acknowledgements We thank Adrien Hardy for useful discussions throughout the project. We thank Michał Dereziński for his insightful comments and suggestions on an early draft. We acknowledge support from ERC grant Blackjack (ERC-2019-STG-851866) and ANR AI chair Baccarat (ANR-20-CHIA-0002).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

A Proof of the well-definedness of Definition 3.1

It is obvious that the Janossy densities are positive. Therefore, in order to prove that proportional volume sampling is well defined; see Daley and Vere-Jones (2003, Proposition 5.3.II.(ii)), we only need to show that

$$\sum_{n \geq 0} \frac{1}{n!} \int_{\Omega^n} j_n(x) \, dv^n(x) = 1.$$  \hfill (A.1)

We write the eigenvalues of $A$ as $\lambda_1 \leq \cdots \leq \lambda_p$ and the spectral decomposition of $A$ as $A = P^T D_\lambda P$, where $D_\lambda$ is the $p \times p$ diagonal matrix with the $\lambda_i$ as its diagonal entries. Then, we define the functions $\psi_i$, $1 \leq i \leq p$, by the linear transform of the function $\phi_i$ defined by $(\phi_1(x), \ldots, \phi_p(x)) := (\phi_1(x), \cdots, \phi_p(x)) P^T$. Finally, we have the decomposition

$$\det(\phi(x)^T \phi(x) + A) = \det(P \phi(x)^T \phi(x) P^T + D_\lambda)$$
$$= \det(\psi(x)^T \psi(x) + D_\lambda)$$
$$= \sum_{S \subset \{p\}} \lambda^S \det(\psi_S(x)^T \psi_S(x))$$

where $\psi_S := (\psi_1, \cdots, \psi_S)$ and $\lambda^S := \prod_{i \notin S} \lambda_i$, with the usual convention $\lambda^\emptyset = 1$; see Collings (1983). Now, by the discrete Cauchy–Binet formula,

$$\det(\psi_S(x)^T \psi_S(x)) = \sum_{T \subset [k]} \det(\psi_S(x_T))^2$$

where $x_T := (x_{T_1}, \cdots, x_{T_{|T|}})$. And, by using the more general Cauchy–Binet formula (Johansson 2006), we get

$$\int_{\Omega^n} \det(\psi_S(x_T))^2 \, dv^n(x) = |T|! \det(G_v(\psi_S)) \nu(\Omega)^{n-|T|}.$$  

Therefore

$$\sum_{n \geq 0} \frac{1}{n!} \int_{\Omega^n} \det(\phi(x)^T \phi(x) + A) \, dv^n(x)$$
$$= \sum_{n \geq 0} \frac{1}{n!} \sum_{S \subset \{p\}} \lambda^S \int_{\Omega^n} \det(\psi_S(x)^T \psi_S(x)) \, dv^n(x)$$
$$= \sum_{n \geq 0} \frac{1}{n!} \sum_{S \subset \{p\}} \lambda^S \sum_{T \subset [k]} |T|! \det(G_v(\psi_S)) \nu(\Omega)^{n-|T|}$$
$$= \sum_{n \geq 0} \frac{1}{n!} \sum_{S \subset \{p\}} \lambda^S \left(\sum_{|S| \geq 1} \nu(\Omega)^{n-|S|} \frac{1}{(n-|S|)!} \det(G_v(\psi_S))\right)$$
$$= \sum_{n \geq 0} \sum_{S \subset \{p\}} \lambda^S \det(G_v(\psi_S)) \frac{\nu(\Omega)^{n-|S|}}{(n-|S|)!}$$
$$= \det(G_v(\psi) + D_\lambda) \exp(\nu(\Omega))$$
$$= \det(G_v(\psi) + A) \exp(\nu(\Omega))$$

where, in the last two identities, we used the facts that (i) $G_v(\psi_S)$ is equal to $G_v(\psi)$, the submatrix of $G_v(\psi)$ whose rows and columns are indexed by $S$, and (ii) $G_v(\psi) = G_v(\phi P^T) = P G_v(\psi) P^T$. This proves (A.1).
B Proof of Proposition 3.2

First, we write

\[
\mathbb{E}\left[ (\phi(X)^T \phi(X) + A)^{-1} \right] = \sum_{n \geq 0} \frac{1}{n!} \int_{\Omega^n} (\phi(x)^T \phi(x) + A)^{-1} j_n(x) d\nu^n(x).
\]

Since \( (\phi(x)^T \phi(x) + A)^{-1} \) is the adjugate matrix of \((\phi(x)^T \phi(x) + A)\), its \((i, j)\) entry is

\[
(-1)^{i+j} \det(\phi_{-j} \phi_{-i}(x) + A_{-j,-i}),
\]

where we define \( A_{-j,-i} \) as the matrix \( A \) with its \( j \)th row and \( i \)th column removed, and \( \phi_{-j} \) as the vector \( \phi(x) \) with its \( i \)th entry removed. Therefore, the \((i, j)\) entry of the matrix \( \mathbb{E}\left[ (\phi(X)^T \phi(X) + A)^{-1} \right] \) is

\[
\sum_{n \geq 0} \frac{1}{n!} \int_{\Omega^n} \frac{(-1)^{i+j} \det(\phi_{-j} \phi_{-i}(x) + A_{-j,-i})}{\det(G_v(\phi) + A) \exp(v(\Omega))} d\nu^n(x).
\]

Using the same reasoning as in the proof of normalization in Sect. A, we get that

\[
\sum_{n \geq 0} \frac{1}{n!} \int_{\Omega^n} \det(\phi_{-j} \phi_{-i}(x) + A_{-j,-i}) d\nu^n(x)
= \det \left( \left( (\phi_a, \phi_b) \right)_{a \neq j, b \neq i} + A_{-j,-i} \right) \exp(v(\Omega)). \tag{B.1}
\]

Note that the proof in Sect. A does not rely on any symmetricity argument, so that identity (B.1) can be proved in the same way. As a consequence, we get that

\[
\sum_{n \geq 0} \frac{1}{n!} \int_{\Omega^n} \frac{(-1)^{i+j} \det(\phi_{-j} \phi_{-i}(x) + A_{-j,-i})}{\det(G_v(\phi) + A) \exp(v(\Omega))} d\nu^n(x)
= (-1)^{i+j} \Delta_{j,i}(G_v(\phi) + A) \exp(v(\Omega))
\]

which is the \((i, j)\) entry of the inverse matrix of \( G_v(\phi) + A \). This proves identity (3.2).

Finally, the proof of identity (3.3) is straightforward:

\[
\mathbb{E}\left[ \det(\phi(X)^T \phi(X) + A)^{-1} \right] = \sum_{n \geq 0} \frac{1}{n!} \int_{\Omega^n} \frac{\exp(v(\Omega))}{\det(G_v(\phi) + A) \exp(v(\Omega))} d\nu^n(x)
= \det(G_v(\phi) + A)^{-1}.
\]

C Proof of Proposition 3.3

By definition of the Janossy densities, we have

\[
\mathbb{E}\left[ \det(\phi(X)^T \phi(X) + A)^{-1} \right] = \frac{\frac{1}{k!} \int_{\Omega^k} j_k(x) \det(\phi(x)^T \phi(x) + A)^{-1} d\nu^k(x)}{\frac{1}{k!} \int_{\Omega^k} j_k(x) d\nu^k(x)}. \tag{C.1}
\]

The integral in the numerator simplifies to

\[
\int_{\Omega^k} j_k(x) \det(\phi(x)^T \phi(x) + A)^{-1} d\nu^k(x)
= \int_{\Omega^k} \det(G_v(\phi) + A) \exp(v(\Omega)) d\nu^k(x)
= v(\Omega)^k \det(G_v(\phi) + A)^{-1}. \tag{C.2}
\]

As for the denominator of (C.1), following the lines of Sect. A leads to

\[
\frac{1}{k!} \int_{\Omega^k} j_k(x) d\nu^k(x) = \sum_{S \subset [p]} \lambda^S \det(G_v(\psi_S)) \frac{v(\Omega)^{k-|S|}}{(k-|S|)!} \frac{\nu(\Omega)^{k-|S|}}{(k-|S|)!} \frac{\nu(\Omega)^{k-p}}{(k-p)!}, \tag{C.3}
\]

where the \( \psi \) functions are defined the same way as in Sect. A. Recalling that

\[
\sum_{S \subset [p]} \lambda^S \det(G_v(\psi_S)) = \det(G_v(\phi) + A),
\]

we can rewrite the sum in (C.3) as

\[
\sum_{S \subset [p]} \lambda^S \det(G_v(\psi_S)) \frac{v(\Omega)^{k-|S|}}{(k-|S|)!} \frac{\nu(\Omega)^{k-|S|}}{(k-|S|)!} \frac{\nu(\Omega)^{k-p}}{(k-p)!} \frac{\nu(\Omega)^{k-p}}{(k-p)!}
= \frac{v(\Omega)^{k-p}}{(k-p)!} \det(G_v(\phi) + A) + \sum_{S \subset [p]} \lambda^S \det(G_v(\psi_S)) \left( \frac{v(\Omega)^{k-|S|}}{(k-|S|)!} - \frac{\nu(\Omega)^{k-p}}{(k-p)!} \right).
\]

Now, since \( v(\Omega) = k \), the sequence \( i \mapsto \nu(\Omega)^i / i! \) is increasing when \( i \leq k \). Hence, for all \( S \subset [p] \) such that \( S \neq [p] \),

\[
\frac{v(\Omega)^{k-|S|}}{(k-|S|)!} \frac{\nu(\Omega)^{k-p}}{(k-p)!} \geq \frac{v(\Omega)^{k-p+1}}{(k-p+1)!} - \frac{v(\Omega)^{k-p}}{(k-p)!} = \frac{k-k-p}{(k-p+1)!} - \frac{p-1}{k-p+1}.
\]
We thus obtain
\[
\sum_{S \subseteq \{p\}} \lambda^S \det(G_v(\psi_S)) \frac{\nu(\Omega)^{k-|S|}}{(k-|S|)!} N \geq \frac{k^{k-p}}{(k-p)!} \left( \det(G_v(\phi) + \Lambda) + \frac{p-1}{k-p+1} \left( \det(G_v(\phi) + \Lambda) - \det(G_v(\phi)) \right) \right). \tag{C.4}
\]

Finally, combining (C.1), (C.2), (C.4) and the fact that \(\nu(\Omega) = k\), we get
\[
\sum_{S \subseteq \{p\}} \lambda^S \det(G_v(\psi_S)) \frac{\nu(\Omega)^{k-|S|}}{(k-|S|)!} N \geq \frac{k^{k-p}}{(k-p)!} \left( \det(G_v(\phi) + \Lambda) + \frac{p-1}{k-p+1} \left( \det(G_v(\phi) + \Lambda) - \det(G_v(\phi)) \right) \right). \tag{C.4}
\]

concluding the proof.

\[D\] Proof of Proposition 3.4

Using the convexity of \(x \mapsto 1/x\) on \(\mathbb{R}_+^*\), it comes
\[
\mathbb{E}[\det(\phi(Y)\phi(Y) + \Lambda)^{-1}] \geq \mathbb{E}[\det(\phi(Y)^T \phi(Y) + \Lambda)^{-1}]
= \left( \nu(\Omega)^{-k} \int_{\mathbb{R}^k} \det(\phi(y)^T \phi(y) + \Lambda) d\nu^k(y) \right)^{-1}.
\]

Now, in Sect. C we showed that
\[
\mathbb{E}[\det(\phi(X)^T \phi(X) + \Lambda)^{-1} | X = k]
= \frac{1}{k!} \int_{\mathbb{R}^k} j_k(x) \det(\phi(x)^T \phi(x) + \Lambda)^{-1} d\nu^k(x)
= \frac{\nu(\Omega)^k}{\exp(\nu(\Omega))} \det(G_v(\phi) + \Lambda)^{-1}
\int_{\mathbb{R}^k} \frac{\det(\phi(x)^T \phi(x) + \Lambda)}{\det(G_v(\phi) + \Lambda) \exp(\nu(\Omega))} d\nu^k(x)
= \nu(\Omega)^k \left( \int_{\mathbb{R}^k} \det(\phi(x)^T \phi(x) + \Lambda) d\nu^k(x) \right)^{-1}
\]
which concludes the proof.

\[E\] Proof of Proposition 3.5

By definition of the Janossy densities, we have
\[
\mathbb{E}\left[ \text{Tr}(\phi(X)^T \phi(X) + \Lambda)^{-1} | |X| = k \right]
= \frac{1}{k!} \int_{\Omega^k} j_k(x) \text{Tr}((\phi(x)^T \phi(x) + \Lambda)^{-1}) d\nu^k(x)
= \frac{1}{k!} \int_{\Omega^k} j_k(x) d\nu^k(x) \tag{E.1}
\]

Using the same notation as in Sect. A, we expand the numerator into
\[
\frac{1}{k!} \int_{\Omega^k} j_k(x) \text{Tr}((\phi(x)^T \phi(x) + \Lambda)^{-1}) d\nu^k(x)
= \frac{1}{k!} \int_{\Omega^k} j_k(x) \text{Tr}((\psi(x)^T \psi(x) + D_\Lambda)^{-1}) d\nu^k(x)
= \sum_{i=1}^{p} \int_{\Omega^n} \Delta_{i,i}((\psi(x)^T \psi(x) + D_\Lambda)^{-1}) d\nu^k(x). \tag{E.2}
\]

Now,
\[
\int_{\Omega^n} \Delta_{i,i}((\psi(x)^T \psi(x) + D_\Lambda)^{-1}) d\nu^k(x)
= \sum_{S \subseteq \{p\} \setminus \{i\}} \lambda^S \det(G_v(\psi_S)) \frac{\nu(\Omega)^{k-|S|}}{(k-|S|)!}, \tag{E.3}
\]
where in this case \(S^c\) denotes the complement of \(S\) relative to \([p]\setminus\{i\}\). Note that there are exactly dim(Ker(\Lambda)) eigenvalues of \(\Lambda\) equal to 0, so that the elements in the sum in (E.3) are equal to 0 when \(|S| \leq m_0 - 1\). Since \(\nu(\Omega) = k\), the sequence \(i \mapsto \nu(\Omega)^{i}/i!\) is increasing when \(i \leq k\), so that
\[
\int_{\Omega^n} \Delta_{i,i}((\psi(x)^T \psi(x) + D_\Lambda)^{-1}) d\nu^k(x)
\leq \sum_{S \subseteq \{p\} \setminus \{i\}} \lambda^S \det(G_v(\psi_S)) \frac{\nu(\Omega)^{k+1-m_0}}{(k+1-m_0)!}
= \frac{\nu(\Omega)^{k+1-m_0}}{(k+1-m_0)!} \Delta_{i,i}(G_v(\psi) + D_\Lambda)
\]
which, combined with (E.2), gives
\[
\frac{1}{k!} \int_{\Omega^k} j_k(x) \text{Tr}((\phi(x)^T \phi(x) + \Lambda)^{-1}) d\nu^k(x)
\leq \frac{\nu(\Omega)^{k+1-m_0} \text{Tr}((G_v(\phi) + \Lambda)^{-1})}{(k+1-m_0)! \exp(\nu(\Omega))}. \tag{E.4}
\]
Finally, combining (E.1), (E.4) and (C.3) gives

\[
\begin{align*}
\mathbb{E}\left[ \det(\phi(X)^T \phi(X) + A)^{-1} | X = k \right] & \leq \frac{v(\Omega)^{k+1-m_0} \text{Tr}((G_v(\phi) + A)^{-1})}{(k+1-m_0)! \exp(v(\Omega))} \\
& = \frac{v(\Omega)^{k+1-m_0} (k-p)!}{(k+1-m_0)!} \text{Tr}((G_v(\phi) + A)^{-1})
\end{align*}
\]

and since \( v(\Omega) = k \), this concludes the proof.

**F Proof of Proposition 3.7**

The Janossy densities and correlation functions of a point process are linked by the following identity; see Daley and Vere-Jones (2003, Lemma 5.4.III):

\[
\rho_n(x_1, \ldots, x_n) = \sum_{m \geq 0} \frac{1}{m!} \int_{\Omega^m} f_{n+m}(x, y) d\nu^m(y).
\]

Applying this identity to the Janossy densities of \( \mathbb{P}_V^\phi(\phi, \Lambda) \), we get

\[
\sum_{m \geq 0} \frac{1}{m!} \int_{\Omega^m} \frac{\det(\phi(x)^T \phi(x) + \phi(y)^T \phi(y) + A)}{\det(G_v(\phi) + A) \exp(v(\Omega))} d\nu^m(y).
\]

Now, for all \( x_1, \ldots, x_n \in \Omega \), using the same reasoning as in the proof of normalization in Sect. A but replacing the matrix \( \Lambda \) with the matrix \( \phi(x)^T \phi(x) + \Lambda \), we get

\[
\sum_{m \geq 0} \frac{1}{m!} \int_{\Omega^m} \det(\phi(x)^T \phi(x) + \phi(y)^T \phi(y) + A) d\nu^m(y)
\]

\[
= \det(\phi(x)^T \phi(x) + A) \exp(v(\Omega)).
\]

We then conclude that

\[
\rho_n(x_1, \ldots, x_n) = \frac{\det(G_v(\phi) + \Lambda + \phi(x)^T \phi(x))}{\det(G_v(\phi) + A)}.
\]

**G Proof of Proposition 3.8**

For any \( n \in \mathbb{N} \) and \( x \in \Omega^n \), we write \( K[x] \) for the \( n \times n \) matrix with entries \( K(x_i, x_j) \). Since \( G_v(\phi) + \Lambda \) is invertible,

\[
\rho_n(x) = \det\left(I_n + (G_v(\phi) + \Lambda)^{-1} \phi(x)^T \phi(x)\right)
\]

\[
= \det\left(I_n + \phi(x)(G_v(\phi) + \Lambda)^{-1} \phi(x)^T\right).
\]

\[
= \det(I_n + K[x]).
\]

Now, it remains to show that the superposition of \( X \) and \( Y \) has the same correlation functions to conclude that its distribution is \( \mathbb{P}_V^\phi(\phi, \Lambda) \).

Let \( n \in \mathbb{N} \), we recall that the \( n \)-th order correlation function \( \rho_n' \) of \( X \cup Y \) satisfy

\[
\mathbb{E}\left[ \sum_{x_1, \ldots, x_n \in X \cup Y} f(x_1, \ldots, x_n) \right]
\]

\[
= \int_{\Omega^n} f(x_1, \ldots, x_n) \rho_n'(x_1, \ldots, x_n) d\nu(x_1) \ldots d\nu(x_n)
\]

for all integrable functions \( f \), where the \( \neq \) symbol means that the sum is taken on distinct elements of \( X \cup Y \). Since each element of \( X \cup Y \) is either in \( X \) or \( Y \) then (G.2) can be rewritten as

\[
\mathbb{E}\left[ \sum_{x_1, \ldots, x_n \in X \cup Y} f(x_1, \ldots, x_n) \right]
\]

\[
= \sum_{S \subset [n]} \mathbb{E}\left[ \sum_{x_1, \ldots, x_n \in X \cup Y} f(x_1, \ldots, x_n) \right]
\]

\[
= \sum_{S \subset [n]} \mathbb{E}\left[ \sum_{x_1, \ldots, x_n \in S} \sum_{x_j \in Y \cup S} f(x_1, \ldots, x_n) \right]
\]

\[
= \sum_{S \subset [n]} \mathbb{E}\left[ \sum_{x_1, \ldots, x_n \in S} \int_{\Omega^n} f(x_1, \ldots, x_n) \prod_{j \in S^c} d\nu(x_j) \right]
\]

\[
= \sum_{S \subset [n]} \int_{\Omega^n} f(x_1, \ldots, x_n) \det((K(x_i, x_j))_{i,j \in S}) d\nu^n(x)
\]

This proves that the correlation functions of \( X \cup Y \) also satisfy

\[
\rho_n'(x) = \det(I_n + K[x]).
\]

Therefore, \( X \cup Y \) is distributed as \( \mathbb{P}_V^\phi(\phi, \Lambda) \).

**H Proof of Corollary 3.10**

For any \( n \in \mathbb{N} \) and \( x \in \Omega^n \), we write \( K[x] \) for the \( n \times n \) matrix with entries \( K(x_i, x_j) \). Since \( G_v(\phi) + \Lambda \) is invertible,
\[ \rho(x)dx = \phi(x)(G_v(\phi) + \Lambda)^{-1}\phi(x)^T; \text{ see identity (G.1).} \]

Therefore,

\[ \mathbb{E}[|X|] = \mathbb{E}[|Y|] + \mathbb{E}[|Z|] \]

with \( \mathbb{E}[|Y|] = v(\Omega) \) and

\[ \mathbb{E}[|Z|] = \int_{\Omega} \phi(x)(G_v(\phi) + \Lambda)^{-1}\phi(x)^T d\nu(x). \]

Since we can rewrite \( \phi(x)(G_v(\phi) + \Lambda)^{-1}\phi(x)^T \) as \( \text{Tr}((G_v(\phi) + \Lambda)^{-1}G_v(\phi)) \), we get

\[ \mathbb{E}[|Z|] = \text{Tr}((G_v(\phi) + \Lambda)^{-1}G_v(\phi)), \]

concluding the proof.

### I A parametrized reference measure for Sect. 3.4.

To parametrize \( \nu \), we write its density \( f \) as a linear combination of positive functions with nonnegative weights, that is,

\[ f(x) = \sum_{i=1}^{n} \omega_i g_i(x). \tag{1.1} \]

This way, minimizing \( h(G_v(\phi)) \) over \( \nu = f dx \) of the form (1.1) and such that \( \nu(\Omega) = k \) is equivalent to finding \((\omega_1, \cdots, \omega_n)\) minimizing

\[ h\left(\sum_{i=1}^{n} \omega_i G_{g_i}(\phi) + \Lambda \right) \text{ s.t. } \omega \succeq 0 \]

and

\[ \sum_{i=1}^{n} \omega_i \int_{\Omega} g_i(x) dx = k. \tag{1.2} \]

This is now a convex optimization problem that can be solved numerically. For our illustration, we consider that \( h \in \{h_D, h_A\} \) and the \( g_i \) to be the 231 polynomial functions of two variables with degree \( \leq 10 \) as well as their composition with \((x, y) \mapsto (1 - x, 1 - y)\), which are all nonnegative functions on \( \Omega = [0, 1]^2 \). We show in Fig. 7 the density of the measures minimizing (1.2) for both optimality criteria and for \( \Lambda \in \{I_{10}, 0.01I_{10}, 0.0001I_{10}\} \).

### J Performance of PVS when dealing with both qualitative and quantitative variables

Following a similar idea as in Sect. 3.4 and Atkinson et al. (2007, Section 14), we consider the design space \( \Omega = [0, 1]^2 \times \{0, 1\} \) and \( k = p = 11 \), with the regression functions \( \phi_i \), for \( i \leq 10 \), being the 10 bivariate polynomi-

![Fig. 7 3D plots of the densities of the measures minimizing (1.2) for the D and A-optimality criterion when the \( g_i \) functions are the binomial polynomial of degree \( \leq 10 \) as well as their composition with \((x, y) \mapsto (1 - x, 1 - y)\)](image_url)
function of degree \( \leq \) in the setting of Sect. J. The designs are either sampled i.i.d. from \( v \) or from \( P_{\nu}^*(\phi, \Lambda) \), where the density \( v \) is either uniform on \( \Omega \) or \( \nu^* \). Dashdotted lines show the bounds of Proposition 3.3.

Fig. 8 Violin and boxplots of the D-efficiency of 2000 random designs in the setting of Sect. J. The designs are either sampled i.i.d. from \( v \) or from \( P_{\nu}^*(\phi, \Lambda) \), where the density \( v \) is either uniform on \( \Omega \) or \( \nu^* \). The results are very similar result to those in Fig. 3c, where we did not have qualitative factors. This shows that the addition of qualitative factors does not deter the performance of PVS.

We show in Fig. 9 an example of design generated by PVS with or without an optimized measure, compared to a uniformly drawn design and an optimal one. We also show in Fig. 8 the performance of PVS and i.i.d. designs with reference measure being either uniform or \( \nu^* \). The results are very similar result to those in Fig. 3c, where we did not have qualitative factors. This shows that the addition of qualitative factors does not deter the performance of PVS.

References

Andersen, M., Dahl, J., Liu, Z., Vandenberghe, L.: Interior-point methods for large-scale cone programming. In: Sra, S., Nowozin, S., Wright, S. (eds.) Optimization for Machine Learning, MIT Press, chap 1, pp. 55–83 (2012)

Atkinson, A., Donev, A., Tobias, R.: Optimum Experimental Designs, with SAS. Oxford University Press, Oxford Statistical Science Series, Oxford (2007)

Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, USA (2004)

Collings, B.J.: Characteristic polynomials by diagonal expansion. Am. Stat. 37(3), 233–235 (1983)

Daley, D.J., Vere-Jones, D.: An Introduction to the Theory of Point Processes, vol. I, 2nd edn. Springer, New York (2003)

De Castro, Y., Gamboa, F., Henrion, D., Hess, R., Lasserre, J.: Approximate optimal designs for multivariate polynomial regression. Ann. Stat. 47(1), 127–155 (2019)

Dereziński, M., Warmuth, M., Hsu, D.: Leveraged volume sampling for linear regression. In: Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, pp. 2510–2519 (2018)

Dereziński, M., Liang, F., Mahoney, M.: Bayesian experimental design using regularized determinantal point processes. In: Chiappa, S., Calandra, R. (Eds.) Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, PMLR, Online Proceedings of Machine Learning Research, Vol. 108, pp. 3197–3207 (2020)

Dereziński, M., Mahoney, M.: Determinantal point processes in randomized numerical linear algebra. Not. Am. Math. Soc. 68, 1 (2021)

Dette, H.: Bayesian d-optimal and model robust designs in linear regression models. Stat.: J. Theor. Appl. Stat. 25(1), 27–46 (1993)

Dette, H., Studden, W.J.: The Theory of Canonical Moments with Applications in Statistics, Probability, and Analysis. Wiley Series in Probability and Statistics, Wiley, Hoboken (1997)

Dette, H., Melas, V., Pepeljšek, A.: D-optimal designs for trigonometric regression models on a partial circle. Ann. Inst. Stat. Math. 54, 945–959 (2002)

Dick, J., Pillichshammer, F.: Digital Nets and Sequences. Discrepancy Theory and Quasi-Monte Carlo Integration. Cambridge University Press, Cambridge (2010)

Fang, K., Li, R., Sudjianto, A.: Design and Modeling for Computer Experiments. Computer science and data analysis series, 1st edn. Chapman and Hall/CRC, Boca Raton (2006)

Farrell, R.H., Kiefer, J., Walbran, A.: Optimum multivariate designs. In: Proceedings of the Fifth Berkeley Symposium on Mathematical
Statistics and Probability, 1: Statistics, University of California Press, pp. 113–138 (1967)
Fedorov, V.: Theory of Optimal Experiments Designs. Academic Press, New York (1972)
Gautier, G., Bardenet, R., Valko, M.: On two ways to use determinantal point processes for Monte Carlo integration. Tech. rep., ICML workshop on Negative dependence in machine learning (2019a)
Gautier, G., Polito, G., Bardenet, R., Valko, M.: DPPy: DPP Sampling with Python. Journal of Machine Learning Research - Machine Learning Open Source Software (JMLR-MLOSS) (2019b)
Grove, D., Woods, D., Lewis, S.: Multifactor b-spline mixed models in designed experiments for the engine mapping problem. J. Qual. Technol. 36(4), 380–391 (2004)
Hough, J., Krishnapur, M., Peres, Y., Virag, B.: Zeros of Gaussian Analytic Functions and Determinantal Point Processes. American Mathematical Society, Providence (2009)
Hough, JB., Krishnapur, M., Peres, Y., Virág, B.: Determinantal processes and independence. Probability surveys (2006)
Johansson, K.: Random matrices and determinantal processes. Les Houches Summer School Proc. 83(C), 1–56 (2006)
Kulesza, A., Taskar, B.: Determinantal point processes for machine learning. Foundations and Trends in Machine Learning (2012)
Lavancier, F., Møller, J., Rubak, E.: Determinantal point process models and statistical inference. J. R. Stat. Soc.: Ser. B (Stat. Methodol.) 77, 853–877 (2015)
Liski, E., Mandal, N., Shah, K., Sinha, Ba.: Topics in Optimal Design, 1st edn. Lecture Notes in Statistics 163, Springer, New York (2002)
Liu, X., Yue, R.X., Chatterjee, K.: Geometric characterization of d-optimal designs for random coefficient regression models. Statist. Probab. Lett. 159, 108696 (2020)
Macchi, O.: The coincidence approach to stochastic point processes. Adv. Appl. Probab. 7, 83–122 (1975)
Maronge, J., Zhai, Y., Wiens, D., Fang, Z.: Optimal designs for spline wavelet regression models. J. Stat. Plann. Inference 184, 94–104 (2017)
Nikolov, A., Singh, M., Tantipongpipat, UT.: Proportional volume sampling and approximation algorithms for a-optimal design. In: Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, SODA ’19, 1369–1386 (2019)
Piepel, G., Stanfill, B., Cooley, S., Jones, B., Kroll, J., Vienna, J.: Developing a space-filling mixture experiment design when the components are subject to linear and nonlinear constraints. Qual. Eng. 31(3), 463–472 (2019). https://doi.org/10.1080/08982112.2018.1517887
Pronzato, L., Pázman, A.: Design of Experiments in Nonlinear Models: Asymptotic Normality, Optimality Criteria and Small-Sample Properties. Lecture Notes in Statistics, vol. 212. Springer-Verlag, New York (2013)
Pukelsheim, F.: Optimal Design of Experiments. Classics in applied mathematics 50, Society for Industrial and Applied Mathematics (2006)
Pukelsheim, F., Rieder, S.: Efficient rounding of approximate designs. Biometrika 79(4), 763–770 (1992)
Robert, C.P., Casella, G.: Monte Carlo Statistical Methods. Springer, New York (2004)
Summa, M., Eisenbrand, F., Faenza, Y., Moldenhauer, C.: On largest volume simplices and sub-determinants. In: Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms 2015 (2014)
Virtanen, P., Gommers, R., Oliphant, T., et al.: SciPy 1.0: fundamental algorithms for scientific computing in python. Nat. Methods 17, 261–272 (2020)
Woods, D., Lewis, S., Dewynne, J.: Designing experiments for multivariable b-spline models. Sankhya 65, 660–670 (2003)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.