Space-filling designs

O. Roustant, INSA Toulouse

INSA Toulouse, 2023

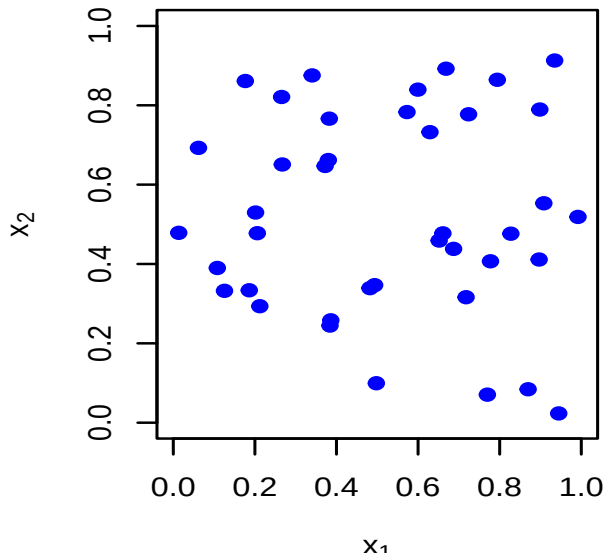
Context and objectives

We aim at studying a complex and deterministic function $f_{\text{sim}} : \mathbb{X} = [0,1]^d \to \mathbb{R}$.
We create an initial dataset, or *design of experiments* (DoE), $X = \{x_1, \ldots, x_n\}$.
Desirable properties for the DoE:

- (space-fillingness) For exploration purpose, as no information on f_{sim} is available.
- (no replication) Evaluate f_{sim} several times at the same design point gives the same result.
- (stability by projection) To avoid losing information if $f_{\text{sim}} : \mathbb{R}^d \to \mathbb{R}$ actually depends on $m < d$ variables or linear combinations of variables.

Why not using a random uniform design?

Obviously points generated at random uniformly in $\mathbb{X}$ satisfy the three constraints. However, space-fillingness is not optimal with few points, as visible below:



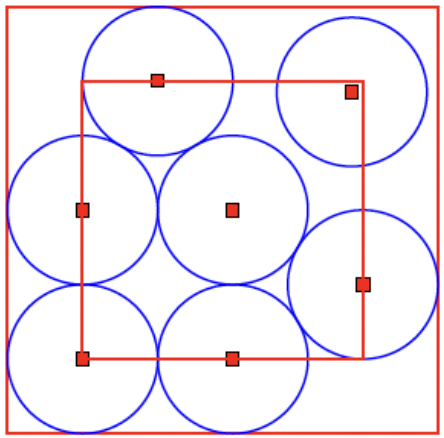
Maximin and minimax designs

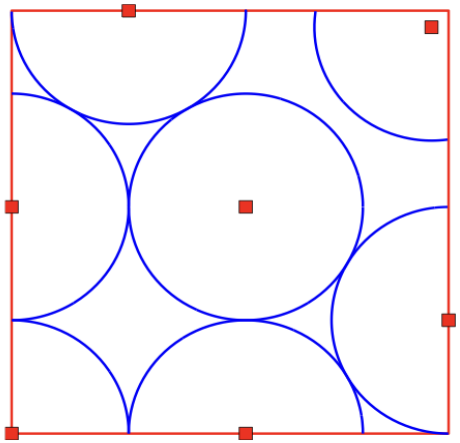
The *maximin-distance* criterion is the minimal distance between the design points:

$$\Phi_{Mm}(X) = \min_{1 \leq i < j \leq n} \|x_i - x_j\|$$

A design that maximizes this distance is called *maximin*.

When $\mathbb{X}$ is convex, finding a maximin design is equivalent to a *sphere-packing problem*, i.e. finding a set of non-overlapping spheres contained in $\mathbb{X}$ with a maximal radius (Pronzato, 2017).





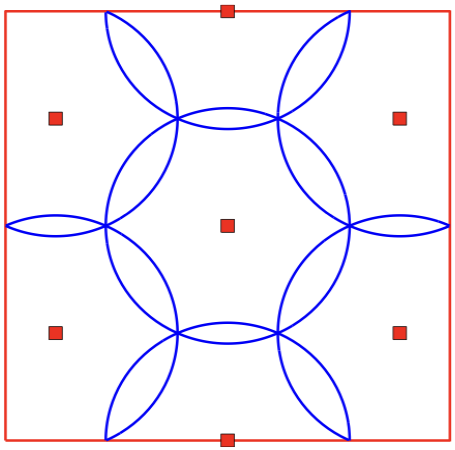
Maximin and minimax designs

The *minimax-distance* criterion is the largest distance between the points in $\mathbb{X}$ and the design points:

$$\Phi_{mM}(X) = \max_{x \in \mathbb{X}} \min_{i=1,\dots,n} \|x - x_i\|$$

The DoEs that minimize this criterion are called *minimax*.

Finding a minimax design is a *sphere covering problem*, i.e. finding a set of spheres that cover $\mathbb{X}$ with a minimal radius.



Minimax DoEs and GP prediction

For many isotropic kernels, i.e. such that $k(x, x')$ depends on $\|x - x'\|$, the kriging variance at x obtained with X, denoted $k_c(x, x; X)$, verifies:

$$\sup_{x \in \mathbb{X}} k_c(x, x; X) \leq S(\Phi_{mM}(X))$$

where S is an increasing function (Schaback, 1995).

Pros and cons

- Minimax DoEs will tend to reduce the global uncertainty of the GP prediction.
- Maximin DoEs are easier to compute

Common practice justified by inequalities between the maximin and minimax criteria: compute only maximin DoEs.

Low discrepancy sequences

Let P be the Lebesgue measure on $\mathbb{X}$, and $P_X = \frac{1}{n}\sum_{i=1}^{n} \delta_{x_i}$ the discrete measure associated to X.

The discrepancy of X with respect to a family of sets $\mathcal{R}$ of $\mathbb{X}$ is defined by

$$\begin{aligned} D(X,\mathcal{R}) &= \sup_{R\in\mathcal{R}} \left| \frac{\text{Card}(i \in \{1,\dots,n\} \text{ s.t. } x_i \in R)}{n} - P(R) \right| \\ &= \sup_{R\in\mathcal{R}} \left| \int_R dP_X(x) - \int_R dP(x) \right| \end{aligned}$$

Examples where $D(X,\mathcal{R})$ can be computed with a closed-form expression

- standard discrepancy $D(X)$: $R = \prod_{i=1}^{n}[a_i, b_i]$ with $0 \leq a_i < b_i \leq 1$
- *star discrepancy* $D^\star(X)$: $R = \prod_{i=1}^{n}[0, b_i]$

Low discrepancy sequences

The star discrepancy gives a control of the quadrature error:

(Koksma-Hlakwa theorem)

For a large class of functions f, we have the inequality

$$\left| \frac{1}{n} \sum_{i=1}^{n} f(x_i) - \int_{\mathbb{X}} f(x) dx \right| \leq V(f)\, D^{\star}(X)$$

where $V(f)$ is the Hardy-Krause total variation of f.

Notice that $V(f)$ does not depend on X, thus the upper bound will be small, uniformly on f, if $D^{\star}(X)$ is small.

Low discrepancy sequences

The class of *low discrepancy sequences* (LDS), used in Quasi Monte Carlo integration, gathers DoE for which $\exists c, s > 0$ such that for all $n \geq 1$,

$$D(X) \leq c \frac{(\log n)^s}{n}$$

To be compared with the rate of convergence of $\frac{1}{\sqrt{n}}$ with Monte Carlo integration.

If $x_1, \ldots, x_n$ are sampled uniformly on $\mathbb{X}$, and $\sigma_f = \mathbb{V}\mathrm{ar}(f(x_i))^{1/2}$, then with probability $\approx 95\%$:

$$\left| \frac{1}{n} \sum_{i=1}^{n} f(x_i) - \int_{\mathbb{X}} f(x) dx \right| \leq \frac{2\sigma_f}{\sqrt{n}}$$

Low discrepancy sequences

The construction of LDS relies on number theory.

The 1D Van der Corput sequence (VdC) uses a 'mirror' decomposition of integers in base 2. If $i = \sum_{j \geq 1} b_j 2^{j-1}$ with $b_j \in [0, 1]$, the i^{th}th point is $x_i = \sum_{j \geq 1} b_j 2^{-j}$.

The Halton sequence extends this procedure in d dimensions by using the first d prime numbers as a number basis for each coordinate, i.e. base 2 for $x_{i,1}$, base 3 for $x_{i,2}$, base 5 for $x_{i,3}$ etc.

i	in base 2	$x_{1,1}$	$x_{2,1}$	in base 3	i
1	1	$2^{-1} = 1/2$	$3^{-1} = 1/3$	1	1
2	10	$0.2^{-1} + 1.2^{-2} = 1/4$	$2.3^{-1} = 2/3$	2	2
3	11	$1.2^{-1} + 1.2^{-2} = 3/4$	$0.3^{-1} + 1.3^{-2} = 1/9$	10	3
...					

Other famous LDS are Faure sequences and Sobol sequences.

Low discrepancy sequences

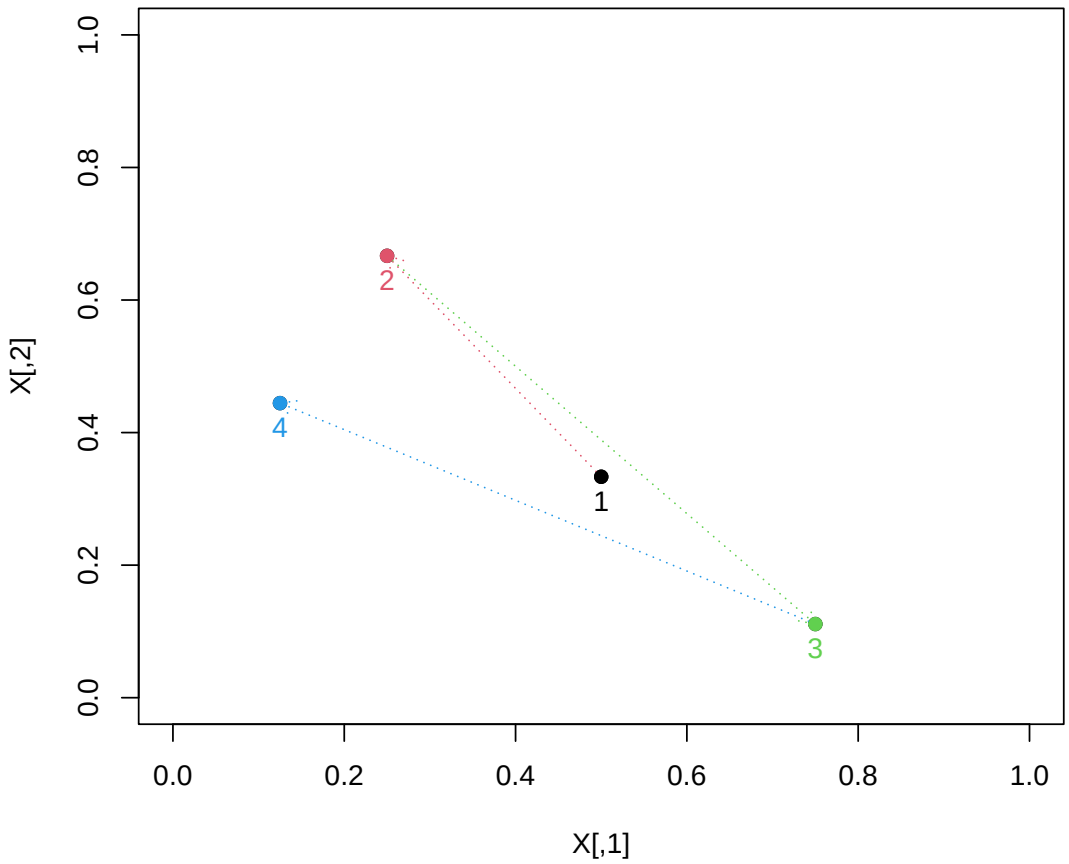


Figure: Examples of Halton sequences

Low discrepancy sequences

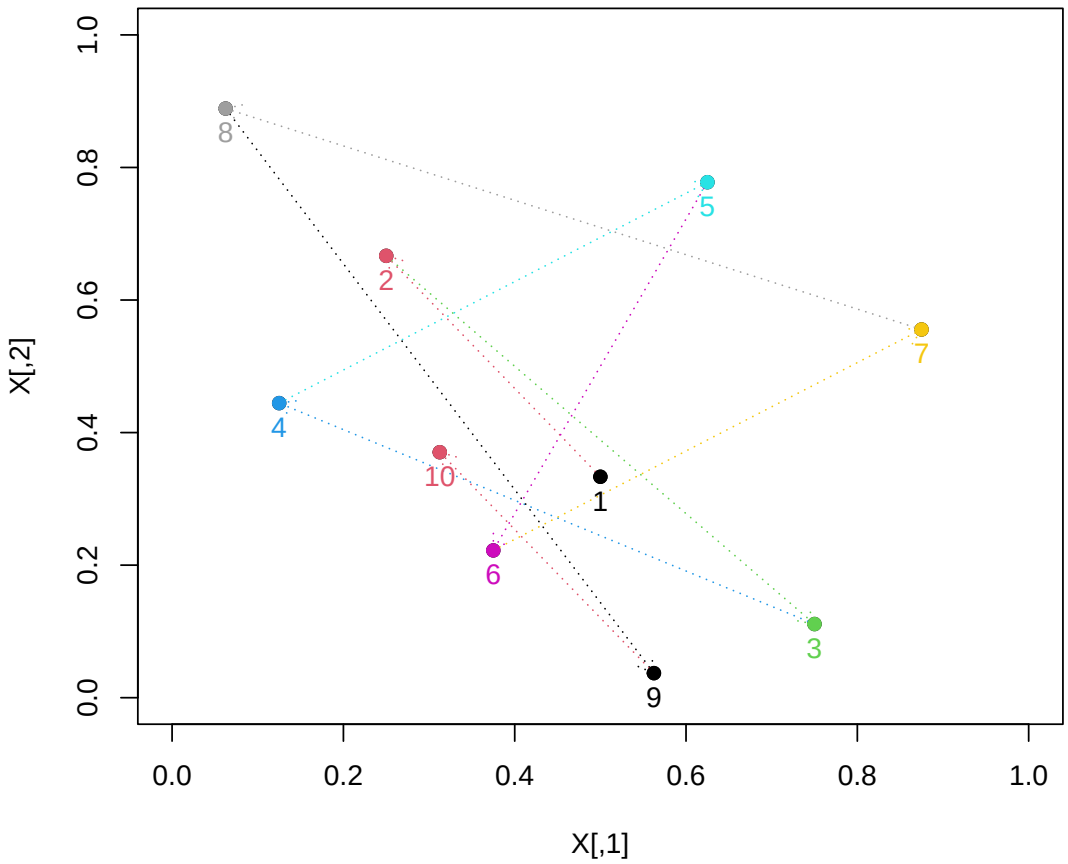


Figure: Examples of Halton sequences

Low discrepancy sequences

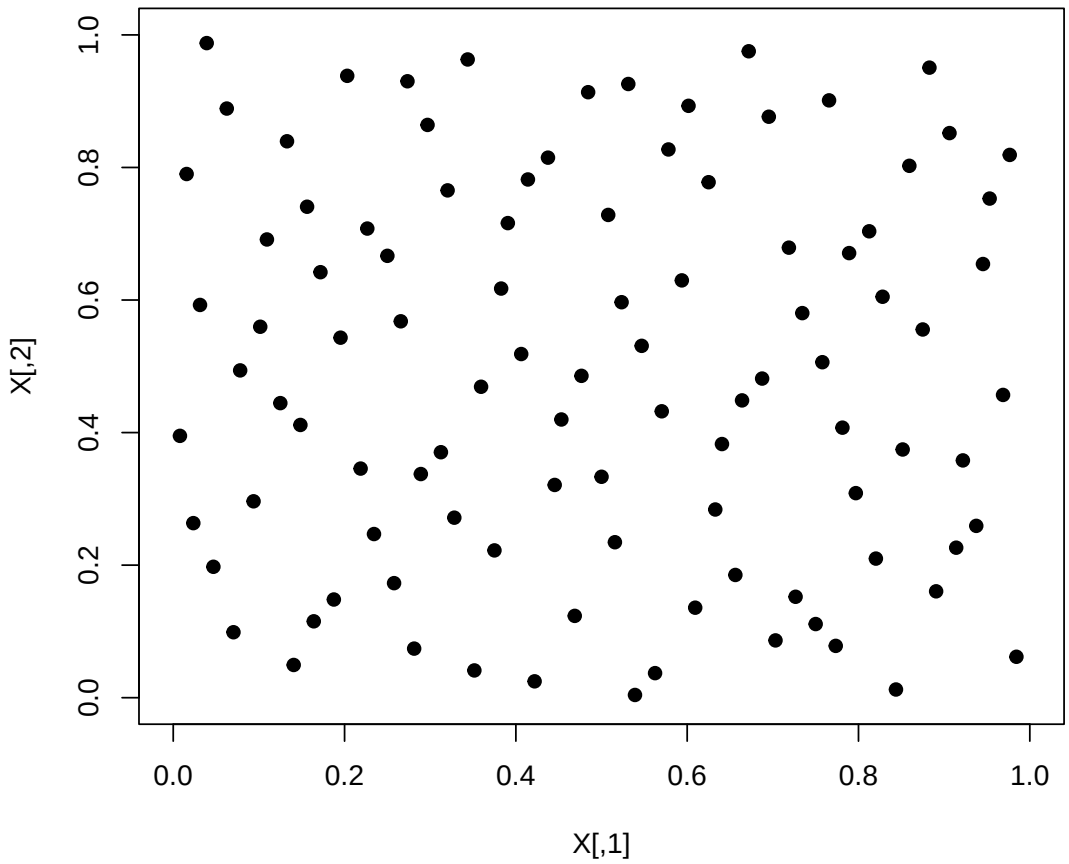


Figure: Examples of Halton sequences

Low discrepancy sequences

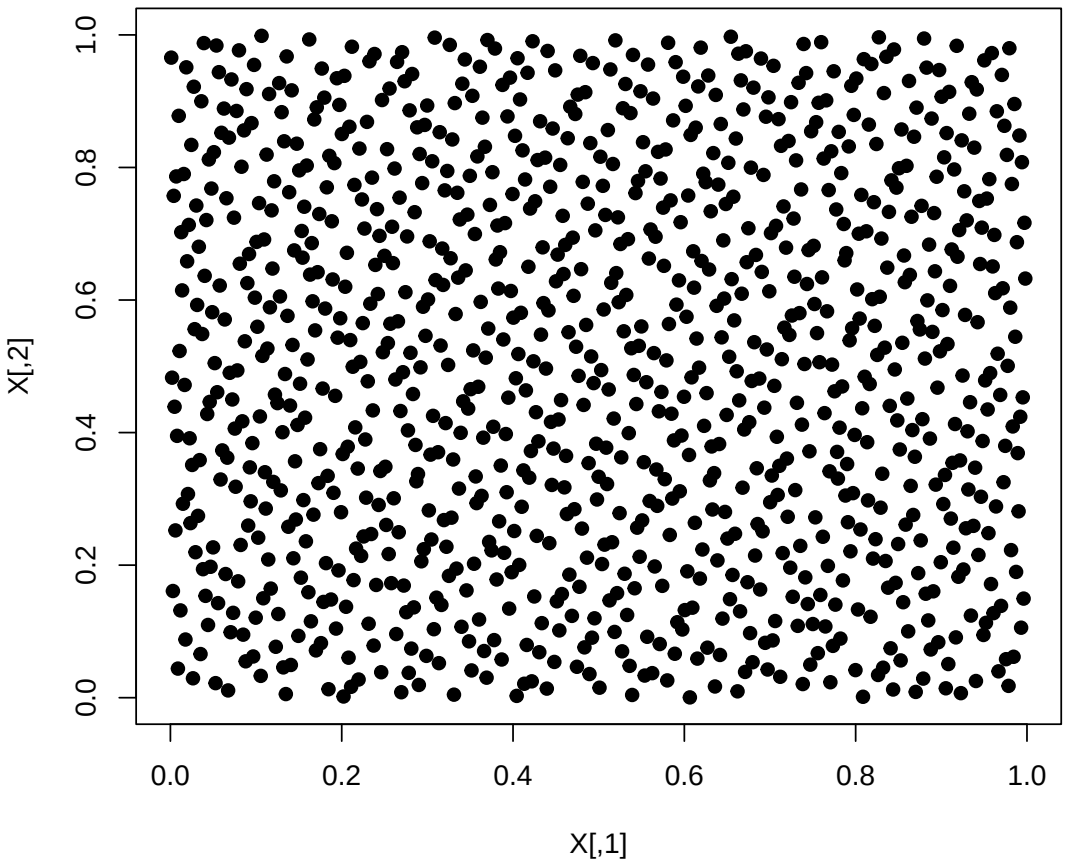
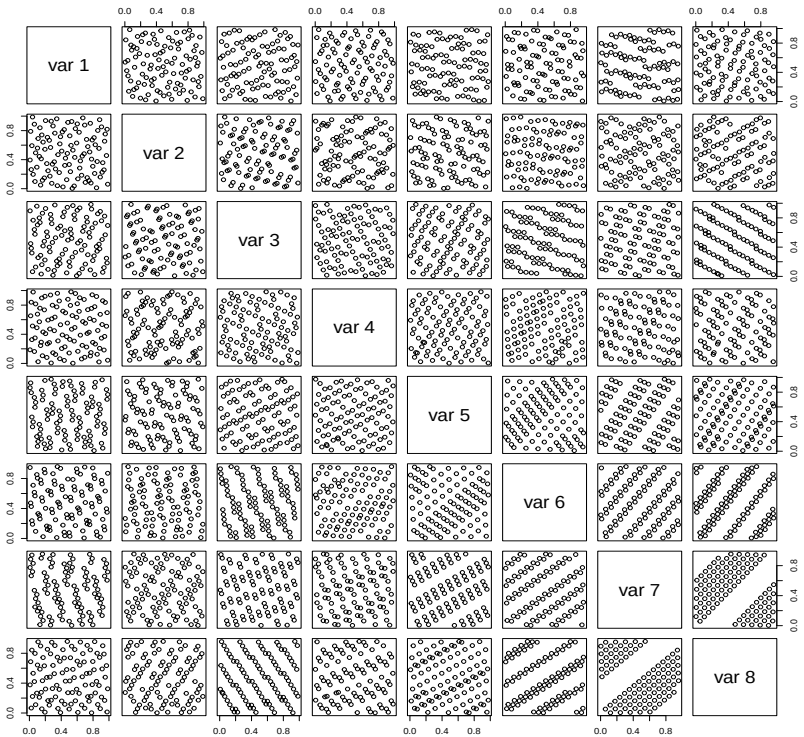


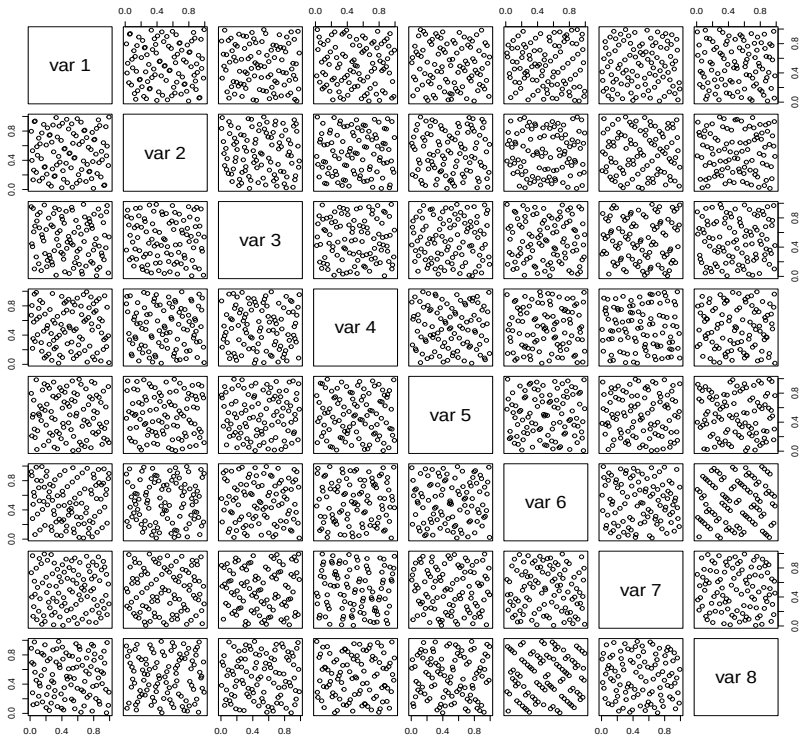
Figure: Examples of Halton sequences

Low discrepancy sequences

The main drawback of LDS for design of experiment is that they behave poorly in projection, in particular in the last coordinates, although they behave better than a uniform design in dimension d.

This effect is striking for Halton(left, below) and Faure LDS, and moderate for Sobol LDS (right, below), which explain that Sobol LDS are more often used.

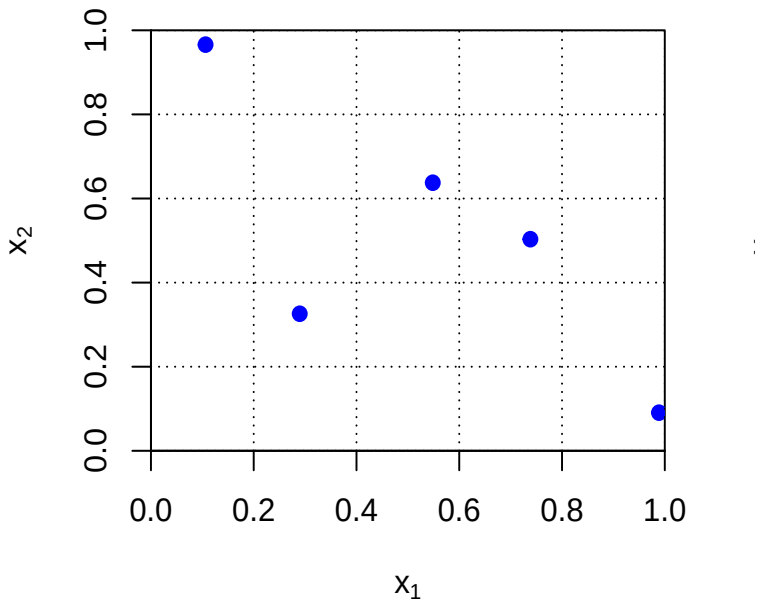




Latin hypercube designs

In 2D, a n-point *Latin hypercube design* (LHD) on $\mathbb{X} = [0,1]^2$ is a random design such that there is exactly one point per row and exactly one point per column.
It can be built with one permutation s_1 of $\{1, \ldots, n\}$

- Create initial points $(i, s_1(i))$, $i = 1, \ldots, n$.
- Make them random by removing $(U_{1,i}, U_{2,i})$, where $U_{1,1}, U_{2,1}, \ldots, U_{1,n}, U_{2,n}$ are i.i.d. uniform on $[0,1]$.
- Divide by n to map the points to $[0,1]$.



Latin hypercube designs

By construction, a LHD is such that there are no replications in projection onto x_1 and x_2. We can make it space-filling by adding a constraint, typically maximin.

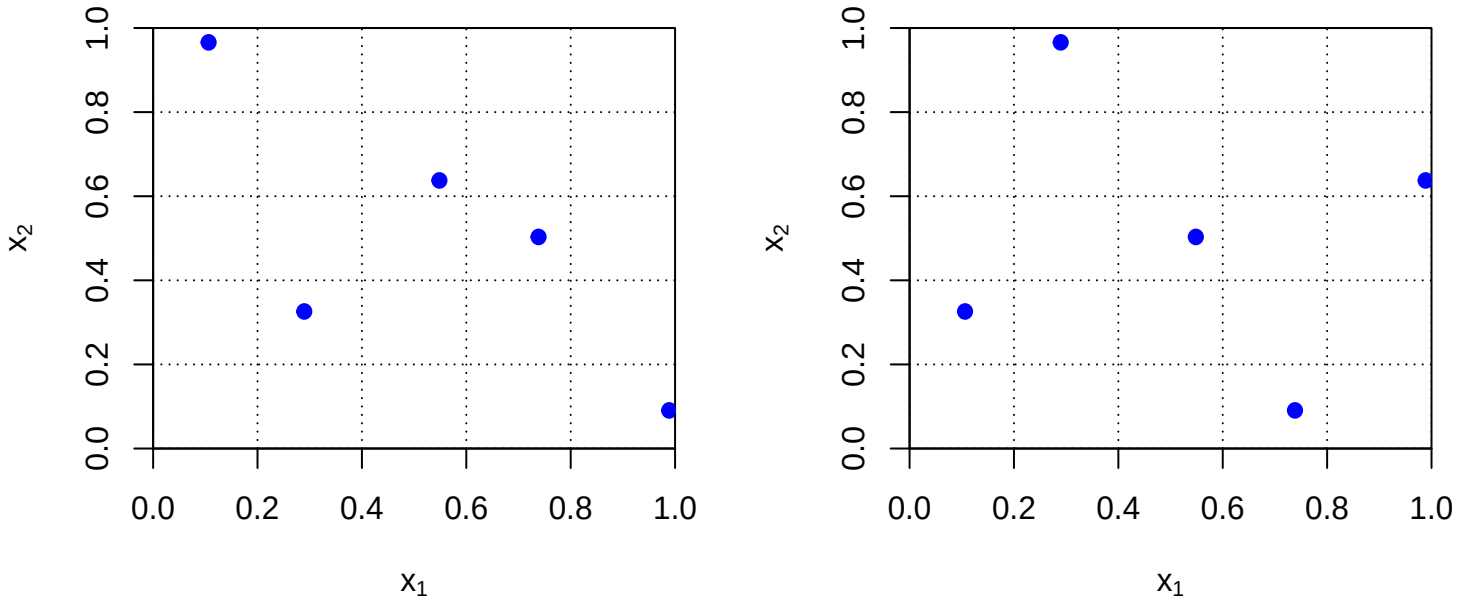


Figure: Examples of 5-point LHDs. Left: a random LHD, associated to the permutation $(5, 2, 4, 3, 1)$. Right: an approximate maximin LHD.

Latin hypercube designs

In d-dimensions, a n-point *Latin hypercube design* (LHD) on $\mathbb{X} = [0,1]^d$ is defined by $d-1$ permutations $s_1, \ldots, s_{d-1}$ of $1, \ldots, n$.

Question: what is the number of LHDs? Consequence for finding a maximin LHD?

Latin hypercube designs

LHD is an example of stratified sampling: for each LHD, the proportion of points that belong to a row (or a column) is forced to be equal to the theoretical one.

The speed of convergence is higher to compute integrals:

If $g : \mathbb{X} \to \mathbb{R}$ is monotonic with respect to all its arguments, if $x_1, \ldots, x_n$ are the points of a LHD, and if $u_1, \ldots, u_n$ are the points of a random uniform design (Monte Carlo sampling), then approximating $\int_{\mathbb{X}} g(x)dx$ by the sample mean is more precise with a LHD (Mckay et al., 2000, Section 2):

$$\mathbb{V}\mathrm{ar}\left(\frac{1}{n}\sum_{i=1}^{n} g(x_i)\right) \leq \mathbb{V}\mathrm{ar}\left(\frac{1}{n}\sum_{i=1}^{n} g(u_i)\right).$$

The monotonicity assumption can be relaxed asymptotically: for large n the result is valid, and the variance reduction depends on the extent to which g is additive (Stein, 1987).

Detecting alignements: the radial scanning statistic

The radial scanning statistic (RSS) automatically detects the clusters in projections onto 1-dimensional axis.

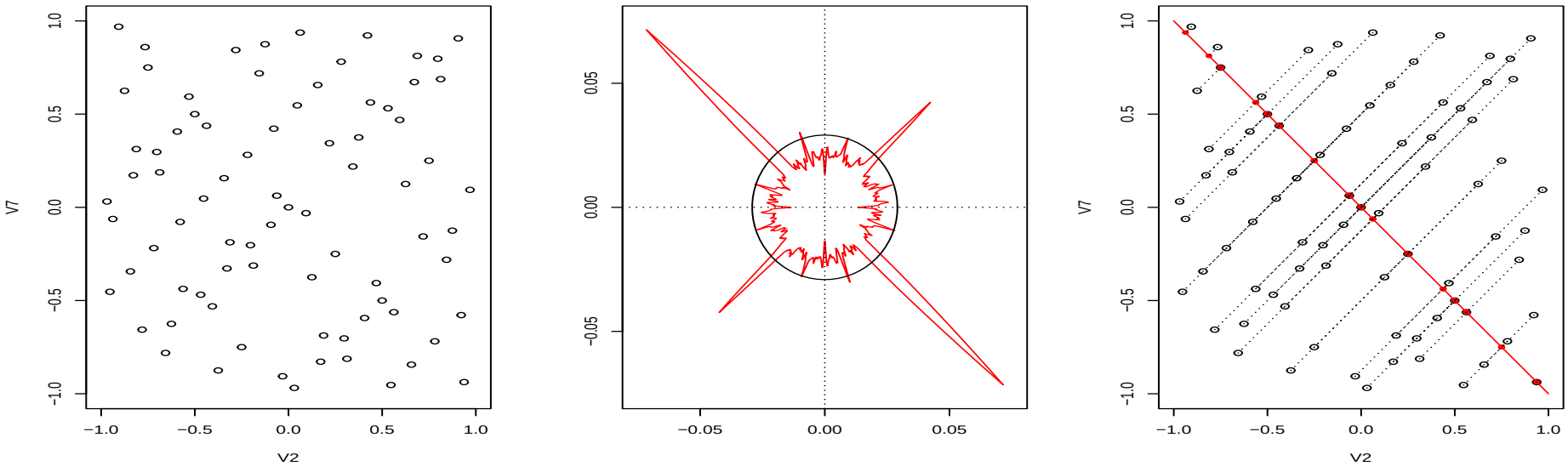


Figure: Defects of a 8D Sobol sequence, detected by the RSS. Left: projection onto the worst 2D space (x_2, x_7). Middle: RSS curve. Right: Projected points for the worst angle.

Detecting alignements: the radial scanning statistic

The RSS tests the assumption $H_0 : x_1, \ldots, x_n \sim \mathcal{U}([-1,1]^d)$.

- Under H_0, the law of the orthogonal projection $\langle x_i, a \rangle$ onto $span(a)$ is known (from Lagrange, 18th century!). Its cdf F_a is written (if $\|a\| = 1, \prod_j a_j \neq 0$)

$$F_a(y) = \frac{1}{\prod_{j=1}^d 2a_j} \sum_{\mathrm{s} \in \{-1,1\}^d} \epsilon(\mathrm{s}) \frac{\max(y + < \mathrm{s}, \mathrm{a} >, 0)^d}{d!} \tag{1}$$

  where $\epsilon(\mathrm{s}) = \prod_{j=1}^d s_j$.
- Under H_0, the points $z_i = F_a(\langle x_i, a \rangle)$ $(i = 1, \ldots, n)$ are i.i.d. $\mathcal{U}[0,1]$.
- Apply a uniformity test suitable to detect clusters, e.g. the Greenwood statistics, based on 'spacings', $G_n = \sum_{i=0}^n (z_{(i+1)} - z_{(i)})^2$, with $0 = z_{(0)} \leq z_{(1)} \leq \cdots \leq z_{(n)} \leq z_{(n+1)} = 1$.

In practice, we apply RSS by first projecting onto 2D (or 3D) pairs of subspaces.

Complements

If the x_i's are independent with cdf F_i, create a space-filling DoE in the space of probabilities since $F_i(x_i)$ are i.i.d. $\mathcal{U}[0,1]$, and transport it back with quantiles.

Among other families of DoEs:

- orthogonal arrays (extensions of LHD for projection onto 2D or higher marginal spaces)
- lattices (other kinds of LDS)
- maximum entropy designs,
- point processes (such as Strauss or determinental processes)
- ...

For examples, visit the R packages: DiceDesign (Dupuy et al., 2015) and randtoolbox (Christophe and Petr, 2023).

Space-filling designs with RKHS!

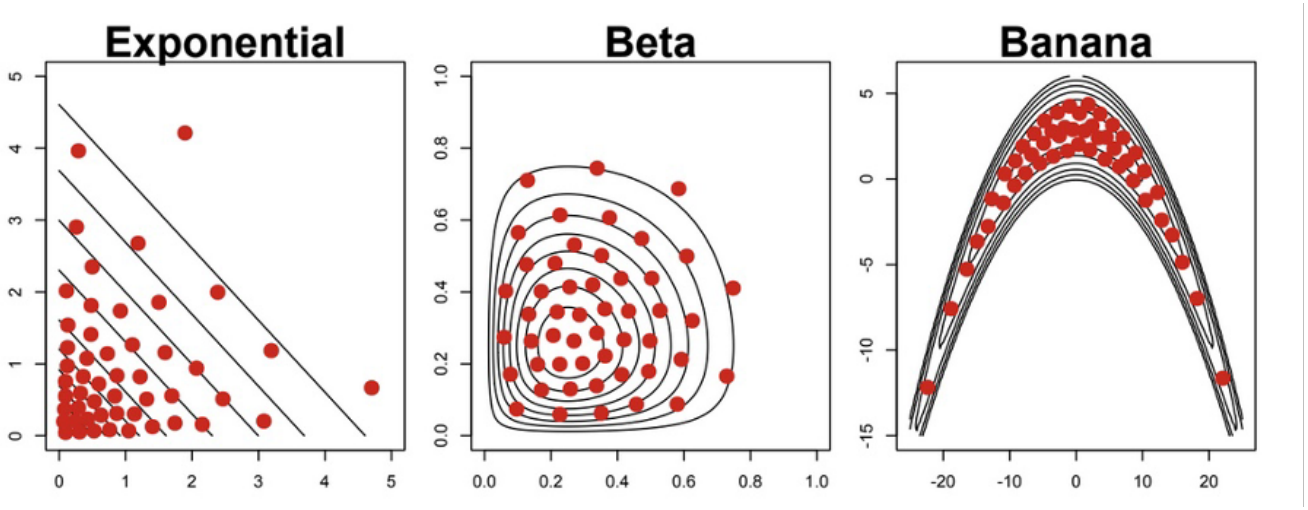


Figure: Examples of designs built by minimizing the MMD (Mak and Joseph, 2018)

Space-filling designs with RKHS!

As for LDS, the design X can be viewed as a discrete measure $P_X = \sum_{i=1}^{n} \frac{1}{n} \delta_{x_i}$, and we aim at quantifying the distance to the uniform measure P.

This can be done with kernel embedding (kernel trick), by first mapping the probability measures in a RKHS $\mathcal{H}$:

$$\begin{aligned} P &\mapsto \mu_P := \int_{\mathbb{X}} k(x,.) dP(x) \\ P_X &\mapsto \mu_{P_X} := \int_{\mathbb{X}} k(x,.) dP_X(x) = \frac{1}{n} \sum_{i=1}^{n} k(x_i,.) \end{aligned}$$

The distance between P, P_X is then measured by the distance of μ_P, μ_{P_X} in $\mathcal{H}$, which is called Maximum Mean Discrepancy

$$MMD(P, P_X) := \|\mu_P - \mu_{P_X}\|_{\mathcal{H}}$$

This approach works also for non-uniform measures and weighted discrete measures.

Space-filling designs with RKHS!

A control of the integration error is given by the MMD (as for discrepancy):

Let $\mathcal{H}$ be a RKHS with a kernel k such that $\int_{\mathbb{X}} \sqrt{k(x,x)} dP(x) < +\infty$.
Then, μ_P is well defined and for all $f \in \mathcal{H}$,

$$\left| \frac{1}{n} \sum_{i=1}^{n} f(x_i) - \int_{\mathbb{X}} f(x) dP(x) \right| \leq \|f\|_{\mathcal{H}} MMD(P, P_X)$$

Furthermore, we have

$$MMD(P, P_X)^2 = \frac{1}{n^2} \sum_{i,j=1}^{n} k(x_i, x_j) - \frac{2}{n} \sum_{i=1}^{n} \int_{\mathbb{X}} k(x_i, x) dP(x) + \int_{\mathbb{X}^2} k(x, y) dP(x) dP(y)$$

We thus aim at finding a DoE such that $MMD(P, P_X)$ is minimum. At least a numerical approach is possible.

References I

Christophe, D. and Petr, S. (2023). *randtoolbox: Generating and Testing Random Numbers*. R package version 2.0.4.

Dupuy, D., Helbert, C., and Franco, J. (2015). DiceDesign and DiceEval: Two R packages for design and analysis of computer experiments. *Journal of Statistical Software*, 65(11):1–38.

Mak, S. and Joseph, V. R. (2018). Support points. *The Annals of Statistics*, 46(6A):2562–2592.

Mckay, M. D., Beckman, R. J., and Conover, W. J. (2000). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1):55–61.

Pronzato, L. (2017). Minimax and maximin space-filling designs: some properties and methods for construction. *Journal de la société française de statistique*, 158(1):7–36.

Schaback, R. (1995). Error estimates and condition numbers for radial basis function interpolation. *Advances in Computational Mathematics*, 3(3):251–264.

Stein, M. (1987). Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29(2):143–151.