Surrogate models and Gaussian Process regression – lecture 4/5

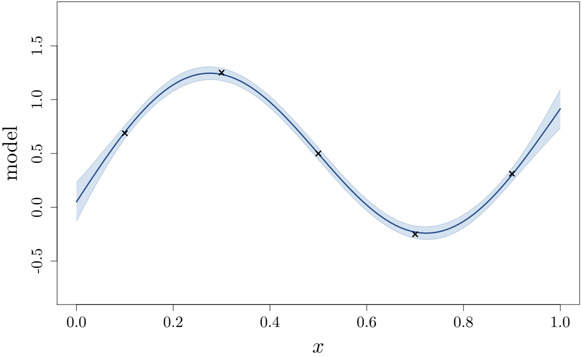
Design of experiments

Mines St-Étienne – Majeure Data Science – 2016/2017

Nicolas Durrande (durrande@emse.fr)

Introduction

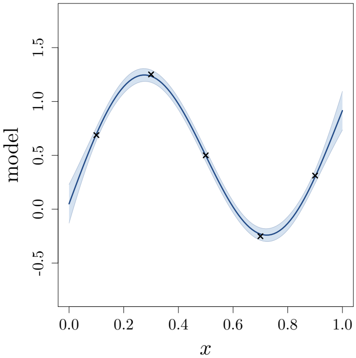
We have seen how to build a model from a given set of input/output tuples $(X,\ F)$:



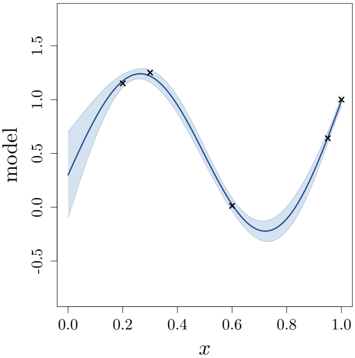
Today's question is: If the input points X can be chosen, how can we obtain the best model?

Motivating example

Same number of points but different input locations



$$RSS = 0.054$$
$$IMSE = 0.001$$



$$RSS = 0.68$$
$$IMSE = 0.004$$

Outline of the lecture

- Classical designs
- Space filling designs
- Optimal design for a given model

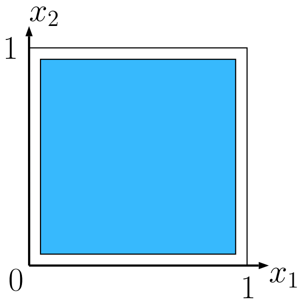
This lecture is based on a course of Victor Picheny (INRA) at Mines St-Étienne.

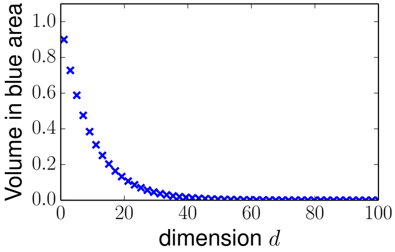
We have focus in the past lecture on 1-dimensional examples, we have to bear in mind that the input space is often of high dimension (d from 5 to 100).

Intuition is often misleading in high-dimension:

Examples 1/2

- Points in the unit cube can be far away
  $\rightarrow$ the diagonal of the unit cube is of length $\sqrt{d}$
- All the volume is near the domain boundaries
  $\rightarrow$ let us consider a hypercube of size 0.9 included in the the unit cube:

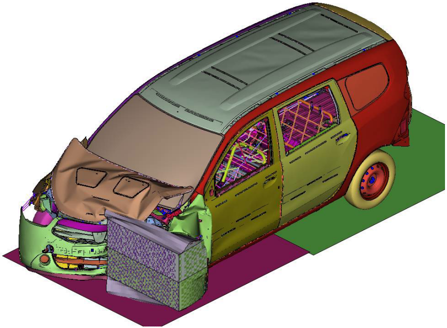




Intuition is often misleading in high-dimension:

Examples 2/2

- The number of vertices of an hypercube increases faster than we usually think

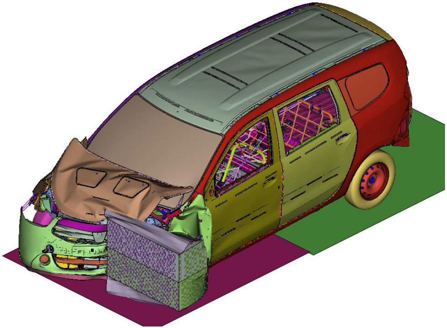


Testing all combinations of min and max input values for the 50 parameters would require...

Intuition is often misleading in high-dimension:

Examples 2/2

- The number of vertices of an hypercube increases faster than we usually think



Testing all combinations of min and max input values for the 50 parameters would require...

$$\mathbf{d = 50 \;\Rightarrow\; 2^d \approx 1.e15\ days}$$

(3000 times the age of the universe)

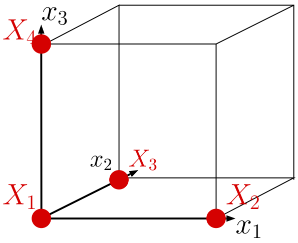
Classical designs

One at a time design

An intuitive way to check the influence of various variables is to make them change one at the time.

- All variables are fixed at a reference value (0 for example)
- One variable is changed at a time to see if there is an influence

Example



point	x_1	x_2	x_3
X_1	0	0	0
X_2	1	0	0
X_3	0	1	0
X_4	0	0	1

pros and cons of this kind of design:

- \+ require only $d+1$ observations
- \+ are easy to interpret
- − they can only see linear effects:

$$m(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3$$

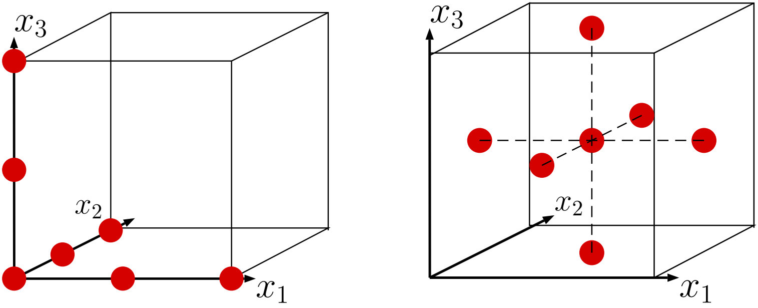
- − they do not cover the space

Exercise

How can this kind of design be adapted to estimate quadratic effect?

Solution

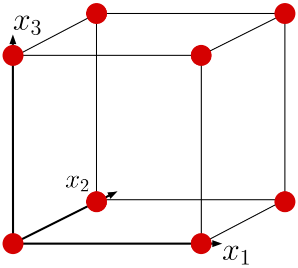
Quadratic effects can be estimated with either



we sometime talk about "star shaped" design.

Factorial designs

The principle of factorial design is to consider all combinations for $x_i \in \{0, 1\}$:



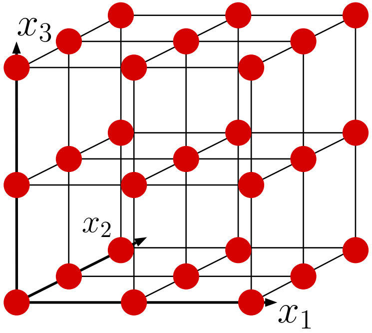
pros They allow to get all interaction terms:

$$\beta_0 + \sum_k \beta_k x_k + \sum_{j,k} \beta_{j,k} x_j x_k + \beta_{1,2,3} x_1 x_2 x_3$$

cons The number of evaluation is unrealistic when d is large

Factorial designs

It is also possible to build factorial designs with k levels:



This allows to compute quadratic effects but the number of evaluations k^d is even less realistic...

Conclusion on classical designs:

pros:

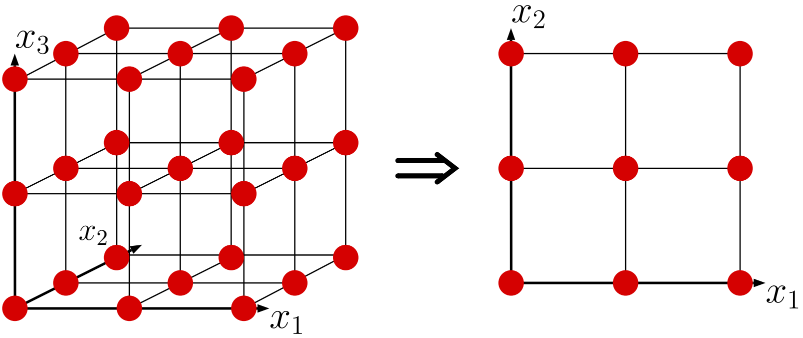
- Easy to use
- adapted to continuous or discrete variables
- Can be combined (star + factorial for example)
- Well suited (often optimal) for linear regression

cons:

- Number of evaluation is not flexible
- Number of evaluation too large in high dimension
- Points are on top of each other when projected

projection issues

Why don't we want points to be superimposed when projected?
If one of the variables has no influence, most observations become redundant...



From 27 observations, we end up with only 9...

Space filling DoE

We are now looking for designs of experiments that:

- are not model oriented
- give information about every domain of the input space
- have good projection on subspaces
- have a flexible number of points

How can we evaluate if a set of points fills the space?

1. Compute the distance between points

maximin the minimum distance between two points of the design should be large:

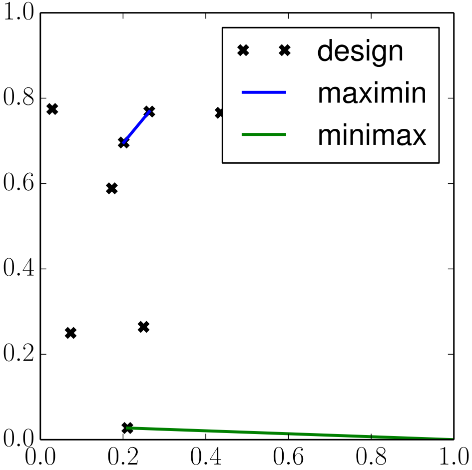
Optimisation problem is: $$\max_{X_1,\dots,X_n} [\min_{i \neq j} dist(X_i, X_j)]$$

minimax the maximum distance between any point of the space and closest design point should be small:

Optimisation problem is: $$\min_{X_1,\dots,X_n} (\max_{x \in D} [\min_{i} dist(x, X_i)])$$

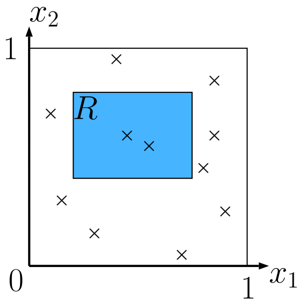
The second criterion is much more difficult to optimise

These criteria can be illustrated on a simple 2-dimensional example



2. Compare the distribution with an uniform distribution

Discrepency is a measure of non uniformity. It compares the number of points in a hyper-rectangle with the expected number of samples from a uniform distribution



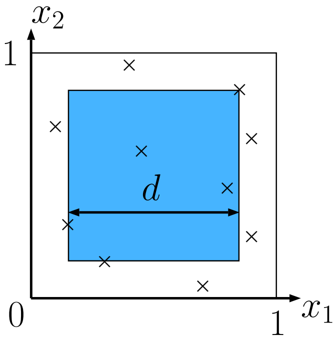
The probability for a uniform variable to be in R is 0.22 and we observe an empirical probability of $2/11$. The discrepancy (w.r.t. R) is then:

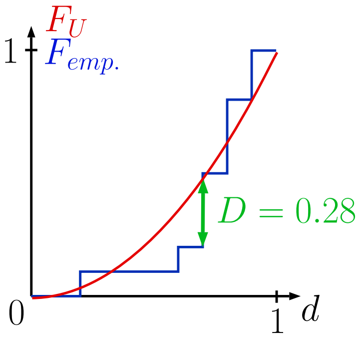
$$D_R = |0.22 - 2/11| = 0.038$$

Discrepency is defined as the sup of the distance between the empirical and analytical cdf.

Discrepency is often computed by fixing:

- one of the hyper-rectangle summit at the origin
- the hyper-rectangle centre at the domain centre





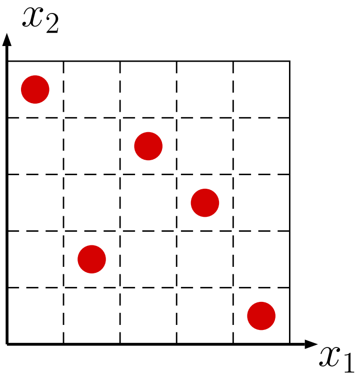
The maximum is located where the rectangle is tangent to points
$\rightarrow$ The optimisation is over a finite space

We will discuss three types of space filling designs:

- Latin hypercubes
- low discrepancy sequences
- centroidal Voronoi tesselations

Latin hypercubes

Latin hypercubes are designs where the domain is sliced in n^d blocks and where there is only one point per 'column':



These designs have good projection properties

A well known example of LHS in 2D is...

A well known example of LHS in 2D is... Sudoku

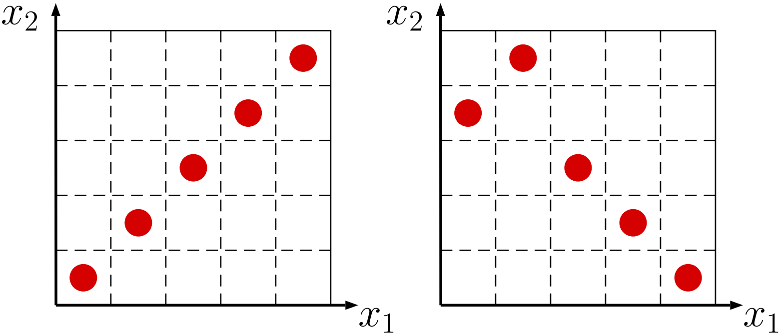
4	3	1	6	7	9	5	2	8
9	6	7	2	5	8	3	4	1
5	8	2	1	4	3	9	6	7
6	5	9	8	1	7	2	3	4
3	2	8	5	6	4	1	7	9
7	1	4	9	3	2	8	5	6
8	7	3	4	2	1	6	9	5
1	4	5	3	9	6	7	8	2
2	9	6	7	8	5	4	1	3

If we focus on one digit (say 4), we obtain a LHD:

●	3	1	6	7	9	5	2	8
9	6	7	2	5	8	3	●	1
5	8	2	1	●	3	9	6	7
6	5	9	8	1	7	2	3	●
3	2	8	5	6	●	1	7	9
7	1	●	9	3	2	8	5	6
8	7	3	●	2	1	6	9	5
1	●	5	3	9	6	7	8	2
2	9	6	7	8	5	●	1	3

Sudoku have more properties that LHD: the generalisation is called **orthogonal array**.

Latin hypercubes do not necessarily cover the space very well...



They have to be combined with a criterion such as maximin.

Exercise

- Generate a 5 points LH in dimension 3.
- How would you program a function $LHD(n, d)$?

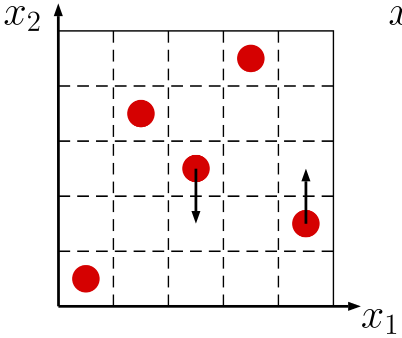
Exercise

How would you optimize LHD?

Exercise

How would you optimize LHD?

the coordinates of two points can be exchanged:



LHD optimization with simulated annealing:

Morris and Mitchell Algorithm

1. Generate LHD
2. find "bad" points according to maximin
3. choose randomly a column of this critical point and exchange it with an randomly selected other point
4. if the criteria is improved, the modification is accepted
5. otherwise, it is accepted with a probability of

$$\exp\left(\frac{maximin_{new} - maximin_{old}}{T}\right)$$

Low discrepancy sequences

Low discrepancy sequences are deterministic sequences that converge toward the uniform distribution.

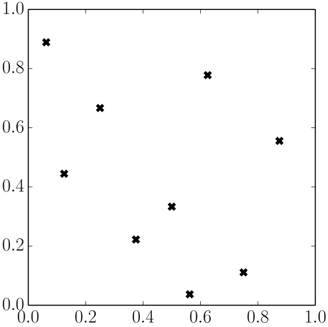
- They cover the space quickly and evenly
- They are easy to build
- It is easy to add new points

Many low discrepancy sequences can be found in the literature: Halton, Hammerley, Sobol', Faure, van der Corput, ...

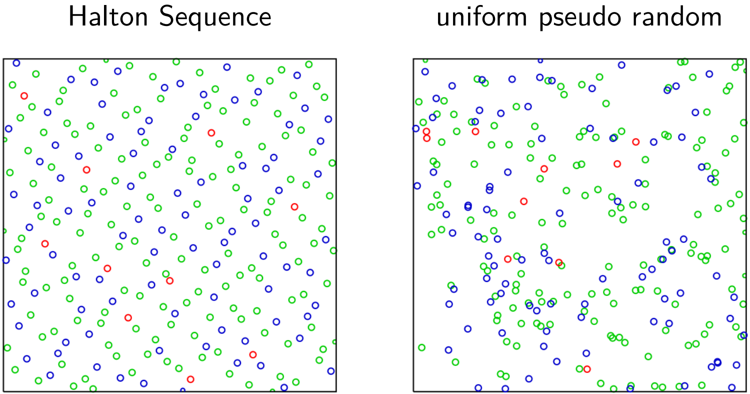
Example (Halton sequence)

Let a and b be two integers with no common dividers (say 2 and 3). The x_1 and x_2 coordinates of the Halton sequence are:

$$x_1 = 1/2,\ 1/4,\ 3/4,\ 1/8,\ 5/8,\ 3/8,\ 7/8,\ 1/16,\ 9/16, \dots$$
$$x_2 = 1/3,\ 2/3,\ 1/9,\ 4/9,\ 7/9,\ 2/9,\ 5/9,\ 8/9,\ 1/27, \dots$$



Example (Halton sequence)



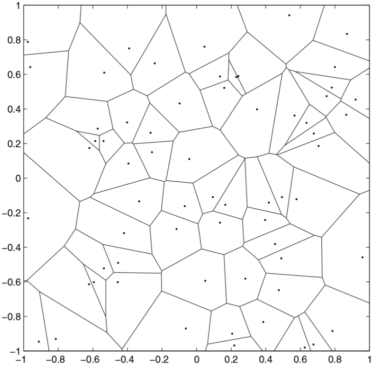
source: wikipedia

Issues with low discrepancy sequences:

- there can be alignments when projected
- there can be holes in subspaces
- points may be aligned (Example: 16 first points in basis (17,18))

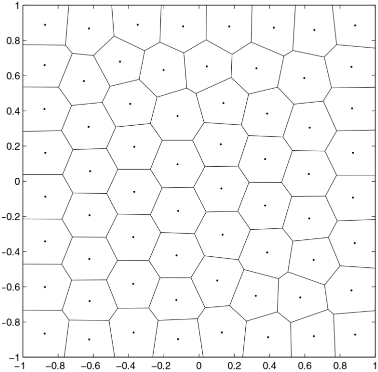
Centroidal Voronoi Tesselations

Given a set of generative points X, the **Voronoi Tesselations** (or Voronoi cells) associated to the point X_i is the region of the space such that X_i is the closest point from the set:



Source: Q. Du et Al., *Centroidal Voronoi Tessellations: Applications and Algorithms*, SIAM Review, 41-4, 1999.

Centroidal Voronoi Tesselations (CVT) is a special case of Voronoi Tesselations where the generative points correspond to the centre of mass of the cells



Source: Q. Du et Al., *Centroidal Voronoi Tessellations: Applications and Algorithms*, SIAM Review, 41-4, 1999.

Properties of CVT:

- Each point of the space is close to one generative points
- The generative points cover the space

$\Rightarrow$ The generative points of CVT can be used as design of experiment.

Generating CVT

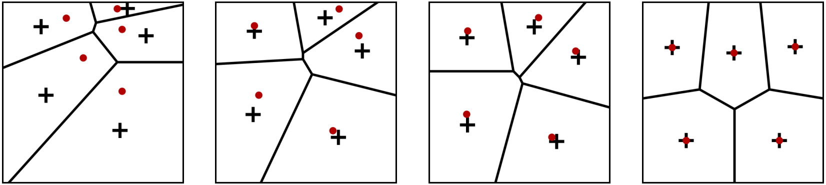
1. Lloyd's Algorithm

1 Initialize X as a set of n points

2 While $i < nb\_iter$

3 Compute the Voronoi diagram of X

4 $\mathrm{X} =$ centre of mass of each cell



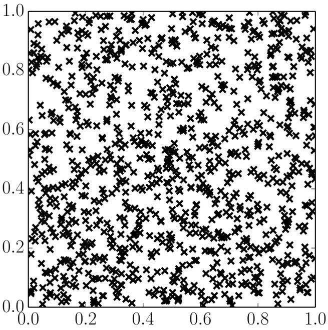
iteration 1 iteration 2 iteration 3 iteration 15

source: “Lloyd's algorithm” wikipedia page

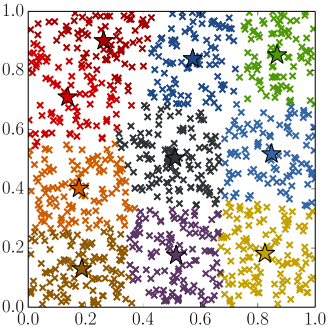
Generating CVT

2. k-means

This algorithm is very similar to Lloyd but it uses a large set of points covering the input space instead of the full continuous domain:



$\Rightarrow$



Generating CVT

3. McQueen algorithm

This algorithm is much faster than the previous ones and gives a good approximation

1 Initialize X as a set of n points

2 Initialize k as a vector of 1 with length n

3 While $i < nb\_iter$

4 generate one random point z in the input space

5 find the X_i closest to z

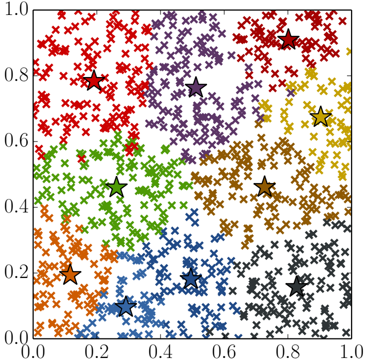
6 update $X_i = \frac{k_i x + z}{k_i + 1}$

7 $k_i = k_i + 1$

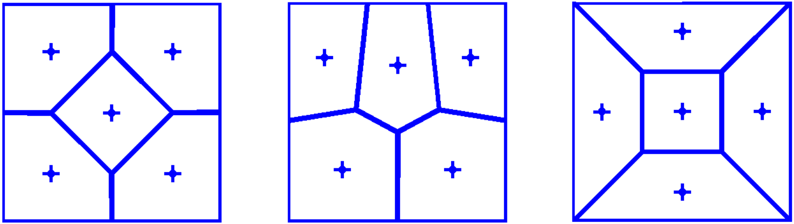
Generating CVT

3. McQueen algorithm

We obtain the following design:



CVT are not unique:



source: wikipedia page “Centroidal Voronoi Tesselations”

Optimal design for regression

Design for regression models

As detailed in lecture 1, the expression of the mean and variance of a linear regression model are:

$$m(x) = B(x)(B(X)^t B(X))^{-1} B(X)^t F$$
$$v(x) = \sigma^2 B(x)(B(X)^t B(X))^{-1} B(x)^t$$

where B is a set of basis functions, X is the DoE, F is the vector of observations and σ^2 is the variance of the observation noise.

What would be the designs such that:

- $\hat{\beta}$ is a good estimate of β?
- the prediction variance is minimal?

Exercise

We consider a linear regression model over $(0, 1)$ with one basis function $b(x) = x$ and one observation X_1.

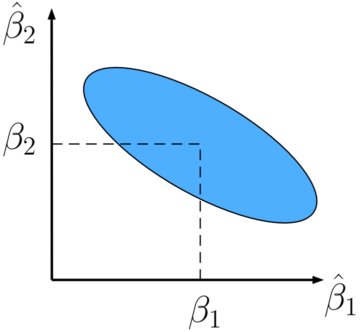
1. Give the expression of m and v.
2. What is the value of x that minimises the maximum of v?
3. Give the expression of the variance of $\hat{\beta}$.
4. What is the value of x that minimises it?

What happen if we have two basis functions $b_0(x) = 1$, $b_1(x) = x$ and two observations?

From the previous example, we can see that

- Minimising beta variance $\Leftrightarrow$ Minimising prediction variance
- X only has an influence on the term $(B(X)^t B(X))^{-1}$, which is the covariance matrix of $\hat{\beta}$ (up to a σ^2 factor)

We thus want to minimise this uncertainty.



Various criteria for the variability of the estimate:

D-optimality

The volume of the confidence ellipsoid is minimized

$$\min_X \det(B(X)^t B(X))^{-1} = \max \det B(X)^t B(X)$$

A-optimality

The sum of the coefficients variance is minimized

$$\min_X \mathrm{tr}(B(X)^t B(X))^{-1}$$

E-optimality

The maximum eigenvalue of $(B(X)^t B(X))^{-1}$ is minimized

$$\min_X \min_i \lambda_i \qquad \text{(where } \lambda_i \text{ is eigenvalue of } B(X)^t B(X))$$

Various criteria for the prediction variance:

G-optimality

maximum of the prediction variance is minimized

$$\min_X \max_x \sigma^2 B(x)(B(X)^t B(X))^{-1} B(x)^t$$

IMSE-optimality (or I-optimality)

the integrated variance is minimized

$$\min_X \int \sigma^2 B(x)(B(X)^t B(X))^{-1} B(x)^t \, \mathrm{d}x$$

In practice, the optimization of these criterion is difficult:

- Large number of variables ($n \times d$)
- multimodal function (lots of symmetries)

Some algorithms (such as Fedorov) are based on one at a time points replacement:

1. Find the worst point in the Design
2. Find a critic region (large variance)
3. Replace the "bad" point by a point in the critic region

Equivalence theorem (Kiefer and Wolfowitz)

The three conditions are equivalent

- A design is D-optimal
- A design is G-optimal
- The maximum prediction variance is p

Knowing a lower bound allows to define the efficiency of a DoE:

$$G_{eff} = 100 \times \sqrt{\frac{\max_x \sigma^2 B(x)(B(X)^t B(X))^{-1} B(x)^t}{p}}$$

Optimal design for Gaussian process regression

As previously, we can discuss two kinds of optimality:

- In the parameter estimations
- In the prediction variance

We will distinguish two cases: when the covariance parameters are known or not.

known covariance parameters

A GP Z with covariance k can be decomposed as a sum of two independent GPs:

$$Z(x) = \underbrace{k(x,X)k(X,X)^{-1}Z(X)}_{Z_X(x)} + \underbrace{Z(x) - k(x,X)k(X,X)^{-1}Z(X)}_{Z_{X^\perp}(x)}$$

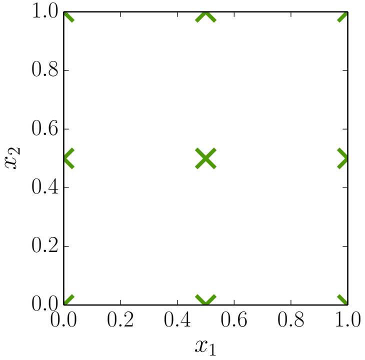
$$k(x,y) = \underbrace{k(x,X)k(X,X)^{-1}k(X,y)}_{k_X(x,y)} + \underbrace{k(x,y) - k_X(x,y)}_{k_{X^\perp}(x,y)}$$

In order to capture most of the variability of Z, we can:

- Maximize the variability of $Z_C(X)$ and apply previous D/A/E-optimality criterion to $k(X,X)$ instead of $B(X)^t B(X)$.
- Minimize the prediction error: I/G-optimality to $k_{X^\perp}(x,x)$.

known covariance parameters

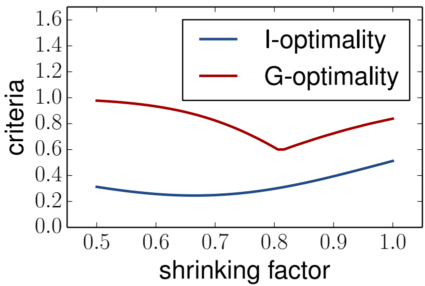
If we maximise the determinant of $k(X, X)$ for a 9 points DoE on $(0, 1)^2$, we find the following design:

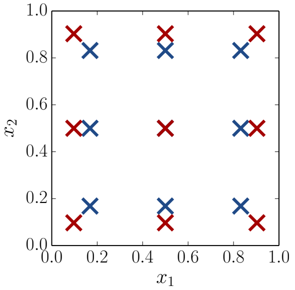


however, this design is not I-optimal nor G-optimal...

known covariance parameters

We can compute numerically the optimal shrinking factor:





$\Rightarrow$ They all give a different optimal DoE.

unknown covariance parameters

What about **unknown** covariance parameter?
The model parameters (ie the kernel's) can be estimated using maximum likelihood.

Can we find a design that gives a good parameter estimation of the variance and lengthscale?

- There is no strong theoretical results
- Good estimation of the variance requires the points to be far away
- Good estimation of the lengthscale requires the points to be close by

If the covariance structure itself is unknown, it is interesting to have in the design a large variety of inter-distances.

Small recap on optimal design for GPR

- All criteria are difficult to compute
- The optimization problem is tricky
- We don't have strong theoretical results as in regression

Good practice:

- space filling designs such as LHS
- Optimization inside a class of DoE

Remark: IMSE is more correlated to minimax than maximin

Adaptive Designs

The principle of **adaptive design** is to add the points in the design one after each other.

$\rightarrow$ the $n \times d$-dimensional optimisation is transformed into n optimisations in d dimensions.

This is still expensive

One new model has to be built for each candidate point.
Furthermore, for each candidate model:

- I-optimality requires to compute a high dimensional integral
- G-optimality requires to optimize the variance

An approximation that is not computationaly expensive is to add the new point where the model variance is maximum:

Algorithm

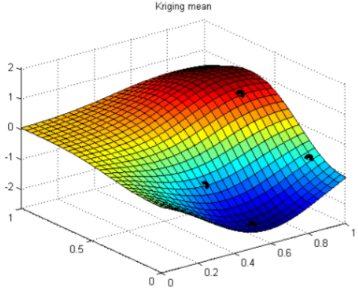
1 Build an initial DoE X with k points

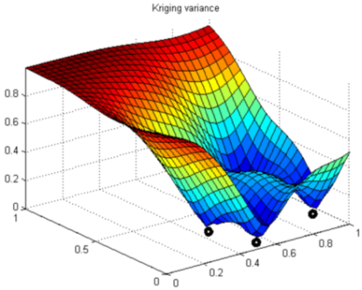
2 While $i < n - k$

3 find $x^* = \mathrm{argmax}(c(x, x))$

4 add x^* to the design and recompute $c(x, x)$

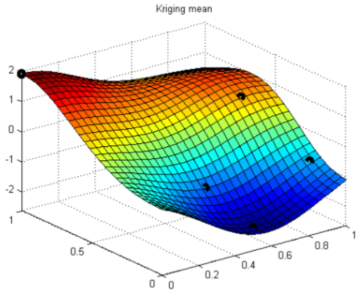
- IMSE = 0.5985
- maxMSE = 0.9991

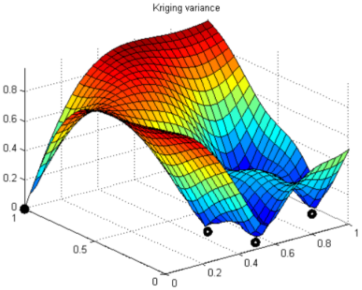




source: Lecture from V. Picheny at Mines St-Etienne

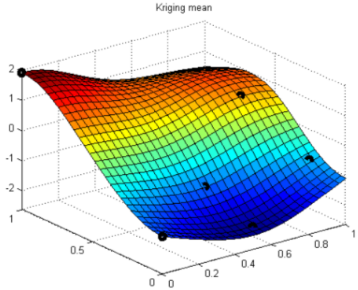
- IMSE = 0.5462
- maxMSE = 0.9665

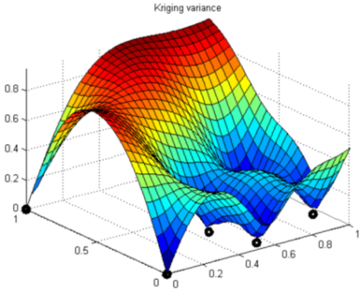




source: Lecture from V. Picheny at Mines St-Etienne

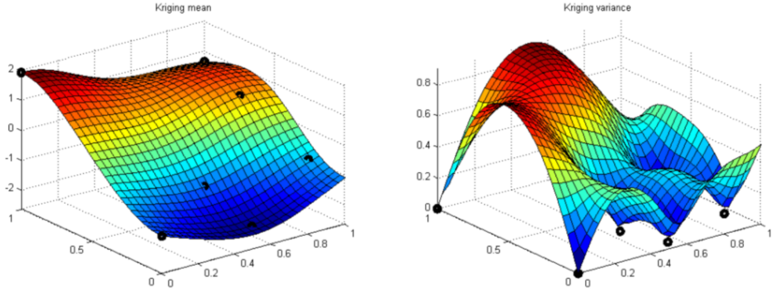
- IMSE = 0.5011
- maxMSE = 0.9466





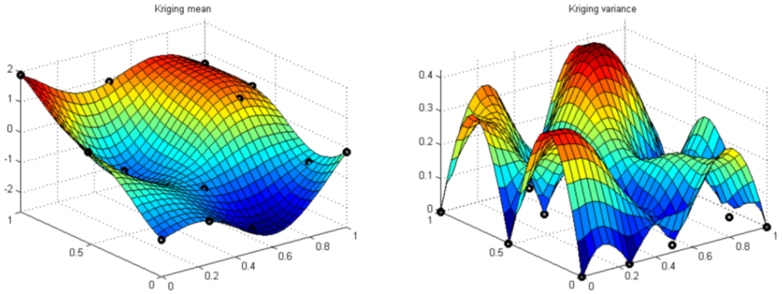
source: Lecture from V. Picheny at Mines St-Etienne

- IMSE = 0.4619
- maxMSE = 0.9035



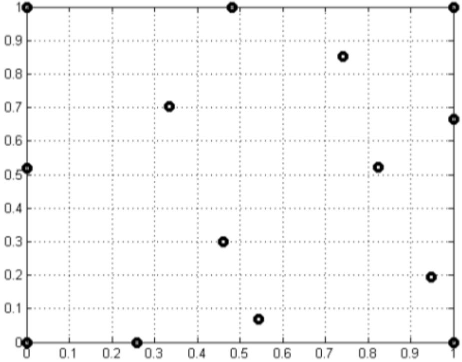
source: Lecture from V. Picheny at Mines St-Etienne

- IMSE = 0.2009
- maxMSE = 0.4226



source: Lecture from V. Picheny at Mines St-Etienne

We end-up with the following design:



It has:

- $+$ Good space filling properties
- $-$ Too much points on the boundaries

Conclusion

Design of Experiments Principles

- Control of data generation
- Thinking the experiments choices

Objectives

- Measure the influence of the variables
- Get accurate models
- Get a good inference for the models

Measure variables influence

The influence of a variable can be estimated by its LR coefficient

additive models: The number of points depends on the complexity of the univariate effects (linear quadratic, ...)

- star shaped designs
- one at a time designs

Models with interaction

- Factorial designs

Designs without model assumption

Designs without model assumption $\rightarrow$ Space filling designs

Various designs have been introduced

- Latin Hypercubes
- Low discrepancy sequences
- Centroidal voronoi tesselations

Various criteria

- Quality of projection
- Discrepancy
- maximin
- minimax

Designs with model assumption

When the form of the model is known, we can define various optimality criteria

Best model estimation

- D-optimality
- A-optimality
- E-optimality

lowest prediction error

- G-optimality
- I-optimality

For linear regression we have some interesting results...
For GPR, it's much more tricky!

Designs optimization

In general, optimizing DoE is difficult:

- large number of variables: $n \times d$
- computationally expensive criteria

Alternatives are:

- Optimizing in a given class of DoE (LHD)
- Adaptive designs
- E-optimality