Practical works on R: Design of computer experiments

B. Iooss & F. Gamboa
Barranquilla, October 2015

Required R packages: DiceDesign, randtoolbox

1. Maximin design

a. Build a maximin design with 9 points and 2 variables, each one following an uniform law U[0,1]. The idea is to generate a large number of random designs and to keep the best in terms of the mindist criterion (minimal distance between two points of the design). Use the appropriate function in the R package "DiceDesign". Visualize this design by comparing it to a Monte Carlo (purely random) design.

b. Build a full factorial design with 9 points and 2 variables using the appropriate function in the "DiceDesign" package. Visualize this design and compare it to the two previous ones. Compute their mindist criterion. Interpret the results in terms of numerical exploration of a model using these designs.

2. Low-discrepancy design

a. Build a Sobol sequence with 9 points and 2 variables using the appropriate function in the "randtoolbox" package. Visualize this design and compute its mindist criterion. Interpret.

b. Compute the L2 centered discrepancy criterion of this design and the 3 designs of §1 by using the appropriate function in the R package "DiceDesign". Interpret.

c. Build an Halton sequence with 200 points and 8 variables (each one U[0,1]) using the appropriate function in the "randtoolbox" package. Visualize all the 2D scatterplots of this design using the pairs() function. What kind of anomaly can you detect? Interpret the results in terms of numerical exploration of a model using these designs.

d. Same question than c) using a Sobol sequence.

e. In order to detect point alignments in the 2D design projections, one can scan all the directions in all the 2D projections using the radar plot technique (which realizes a statistical test about uniformity of projections). Apply this technique on the previous Sobol design using the rss2d() function of the package "DiceDesign". Interpret.

3. Latin Hypercube Sample (LHS)

a. Using the lecture slides, program a R function which allows to build a LHS with uniform margins. Apply the function to build a standard LHS with 20 points and 2 variables U[0,1].

b. Build a LHS maximin with 20 points and 2 variables U[0,1] using two different techniques:
   i. Simulate a large number of LHS and keep the best with respect to the mindist criterion.
   ii. Use a simulated annealing algorithm to optimize the design (maximizing the mindist criterion) in the LHS class. Use the maximinSA_LHS() function of the "DiceDesign" package which necessitates some precise tuning of simulated annealing parameters.

   Visualize and compare (mindist criterion) the i) and ii) designs with the standard LHS and a Monte Carlo design.