Optimization Techniques

Laboratory 9

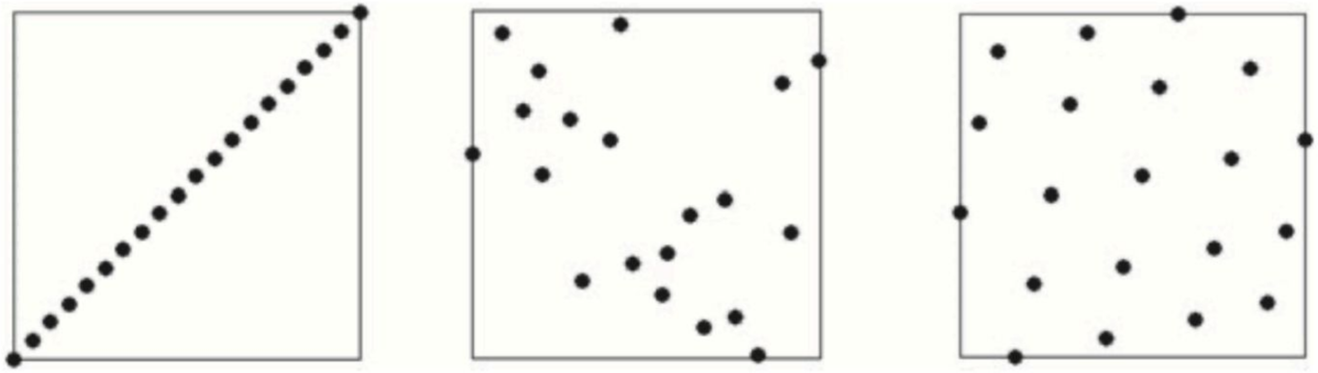
Design of Experiments

cc.rambaldimigliore@unitn.it
elia.cunegatti@unitn.it
mvincze@fbk.eu

UNIVERSITY OF TRENTO - Italy
Information Engineering
and Computer Science Department

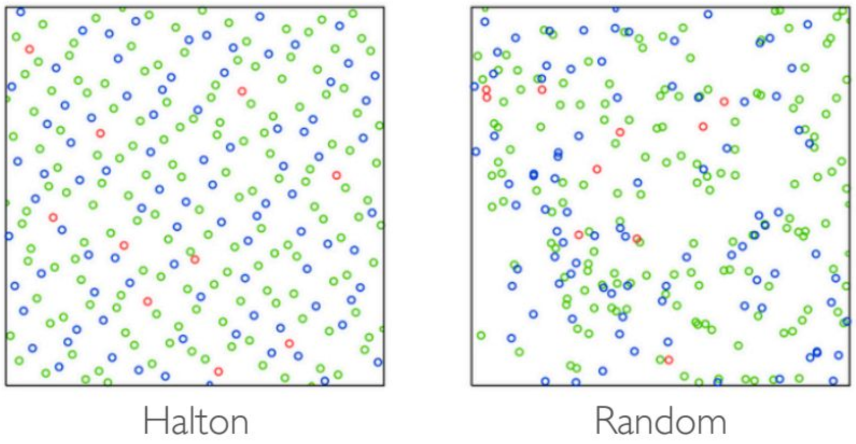
DOEs

- Space-filling sampling methods
- Initialization step to many optimization methods
- Can be used to get insight about the search space



Halton

- Generate samples in a normalized space
- Deterministic sample construction based on coprimes
- Quasi-random low-discrepancy
- Performs well in low dimensions

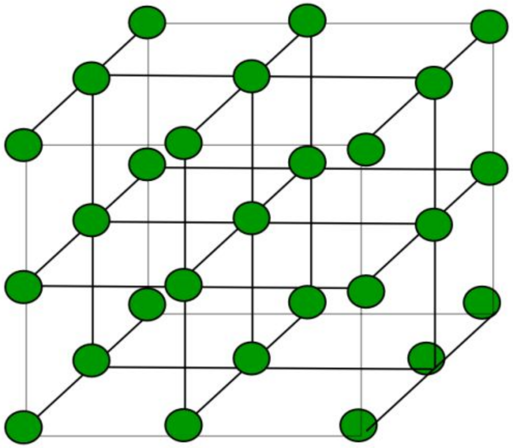



```python
def Halton(b):
    """Generator function for Halton sequence."""
    n, d = 0, 1
    while True:
        x = d - n
        if x == 1:
            n = 1
            d *= b
        else:
            y = d // b
            while x <= y:
                y //= b
            n = (b + 1) * y - x
        yield n / d
```

Full Factorial

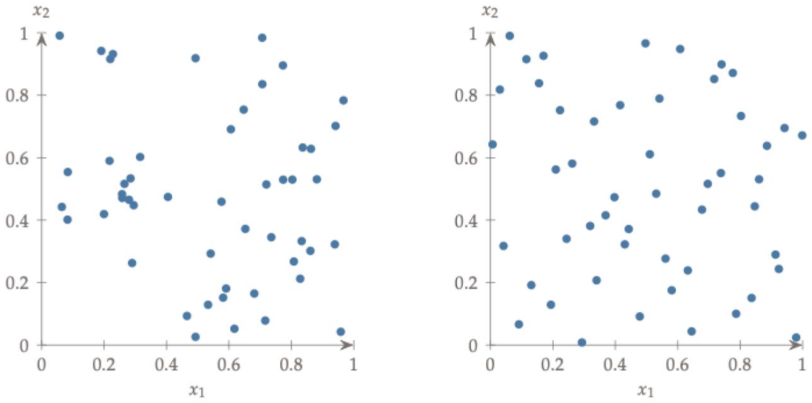
- N variables + M divisions / variable == NxM combinations

- Only applicable for low number of variables



Latin Hypercube

- Each sample is the only one in each axis-aligned hyperplane containing it
- Ensures that each sampling space dimension is roughly evenly sampled



Inputs:

n_s: Number of samples

n_d: Number of dimensions

$P = \{P_1, \ldots, P_{n_d}\}$: (optionally) A set of cumulative distribution functions

Outputs:

$X = \{x_1, \ldots, x_{n_s}\}$: Set of sample points

---

for $j = 1$ **to** n_d **do**

  for $i = 1$ **to** n_s **do**

    $V_{ij} = \frac{i}{n_s} - \frac{R_{ij}}{n_s}$ where $R_{ij} \in \mathbb{U}[0,1]$ Randomly choose a value in each equally spaced cell from uniform distribution

  end for

  $X_{*j} = P_j^{-1}(V_{*j})$ where P_j is a CDF Evaluate inverse CDF

  Randomly permute the entries of this column X_{*j} Alternatively, permute the indices $1 \ldots n_s$ in the prior `for` loop

end for