北京师范大学 香港浸会大学 联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

Methodology of Generating Output Space-filling Sample for Explicit Function

NAME: Yi’an Wang (Andy)
ID: 2030005058

DATE: Dec. 25th, 2023

Content

1. Background Introduction

Space-filling Sample:

Space-filling Sample refers to the sample which can fill the sampling area/ space when the sampling point

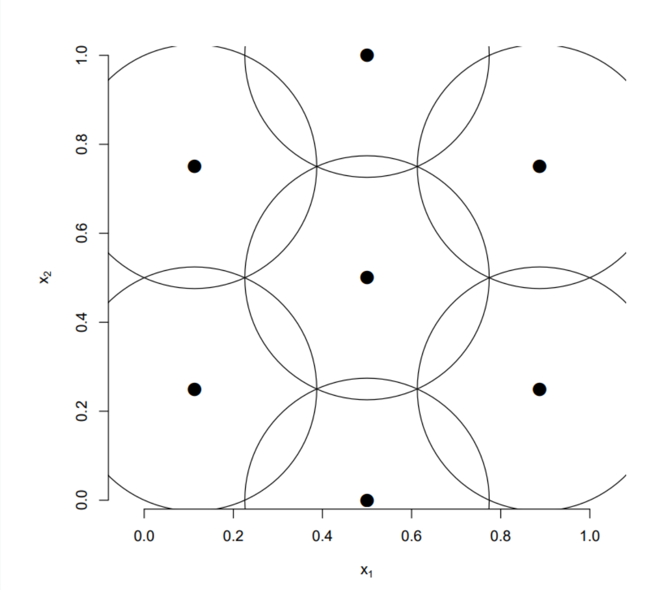
Input Space-filling Samples:

- Mckay, Beckman, and Conover (1979): Latin Hypercube Design.
- Johnson, Moore, and Ylvisaker (1990): Maximin Design, Minimax Design.
- Morries and Mitchell (1995): Maximin Latin Hypercube Design.
- …

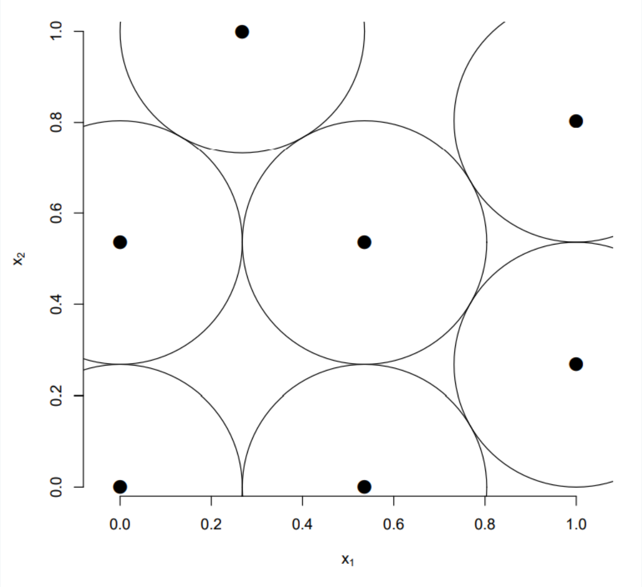
Output Space-filling Sample:

The input value which can make the corresponded output value attain a target of output space-filling.

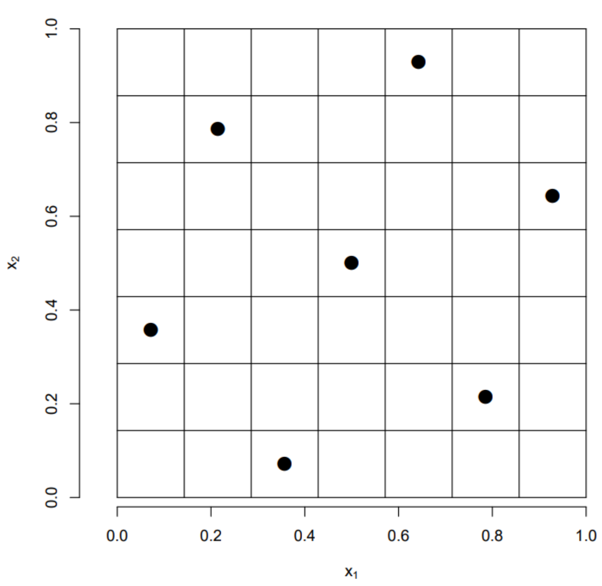
1. Background Introduction



A minimax distance design for $n = 7$ points



A maximin distance design for $n = 7$ points



A (maximin) Latin hypercube design for $n = 7$ points.

Joseph, V. Roshan. "Space-filling designs for computer experiments: A review." *Quality Engineering* 28.1 (2016): 28-35.

1. Background Introduction

- Lu, Anderson-Cook.(2021)

$$D = \{d_{ij} = \alpha d_{ij,X} + (1-\alpha) d_{ij,Y}\}, \alpha \in [0,1]$$

- Krishna, Craig, and Shi, et al. (2022) $y^* = \arg\min_{y \in \mathbf{y}} ||x^* - f(y)|| \leq \Delta$

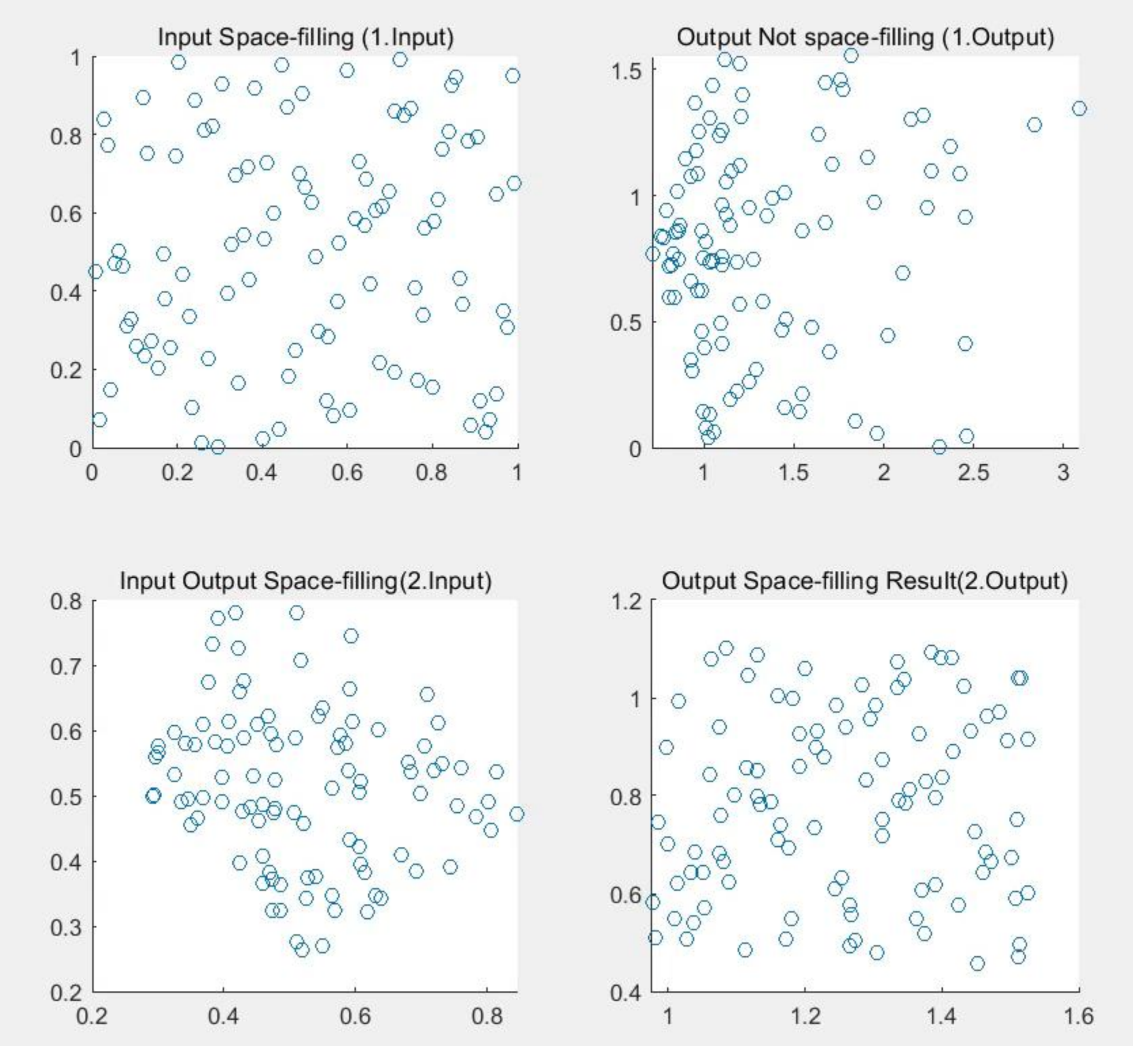
- Wang, General, Kalidindi and Joseph (2023)

$$\max_{y \in \mathbf{y}} \inf_{y_i \in f(D_n)} d_Y(y, y_i)$$

1. Background Introduction

Space-filling:
(Rough Definition)

Space-filling refers to the point which can filling the space uniformly when the number of points increasing.



Intuitive Feeling:

- ISF and OSF are not consistent.

1. Background Introduction

In this presentation, I am going to showing the methodology of generating Output Space-filling for explicit function with pure simulation perspective.

Normality Filling Filtered (NFF) Methodology

Normality: multivariate normal distribution.
Filling Filtered: Use space-filling sample as a filter to the candidate points.

2. Concept Illustration

$(-\infty, c]$

$[c, +\infty)$

Range

$[a, b]$

$(-\infty, \infty)$

2. Concept Illustration

Boundary in Range Space

To have a better illustration, we define two kinds of boundaries:

- Hard Boundary: the boundary which was a clear division between range and non-range space.
- Soft Boundary: the range space is kind of boundless in any one of the range direction. For example

$$\lim_{x \Rightarrow \infty} f(x) = \infty \qquad \lim_{x \Rightarrow c} f(x) = \infty$$

2. Concept Illustration

Bounded Output- Space-filling (BOSF)

To avoid the influence of ∞, we add a bound as our observation target in range space.

Thus, in the following discussion, we add a bound when generating Output Space-filling Sample.

2. Concept Illustration

Output Space-filling Circumstances Discussion

Output Boundary Space-filling (OBSF)

- OBSF is the space-filling with lots of points gathered in the boundary of the range.

Output Internal Space-filling (OISF)

- All the points located inside the range space.
- No significant gathering trend in the boundary of range.

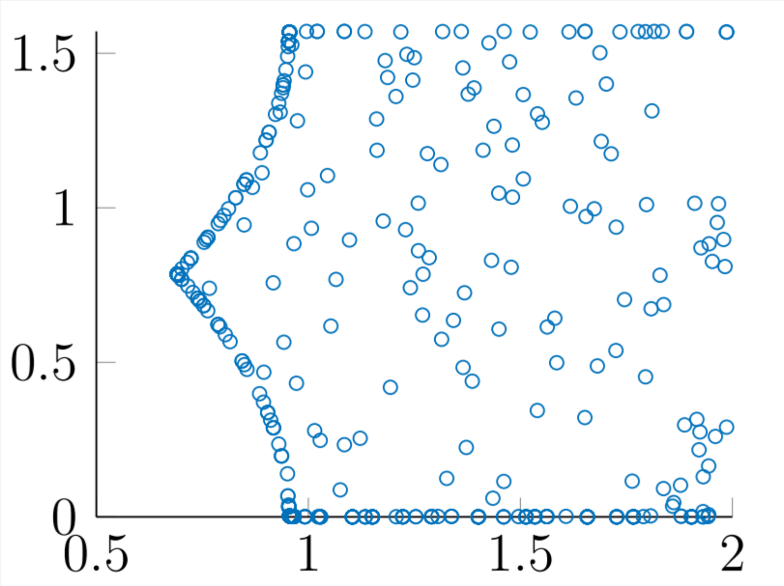
2. Concept Illustration

$$f_7 : \mathbb{R}^2 \to \mathbb{R}^2 \qquad x \in [0,1], y \in [0,1]$$

$$(X, Y) = f_7[x, y] = \left(\frac{1}{\sqrt{x^2 + y^2 + 0.1}}, \arctan \frac{y}{x} \right)$$

Fig 1.1 & Fig 1.2 (Part of)
(LHD filtered, sample size = 1000)

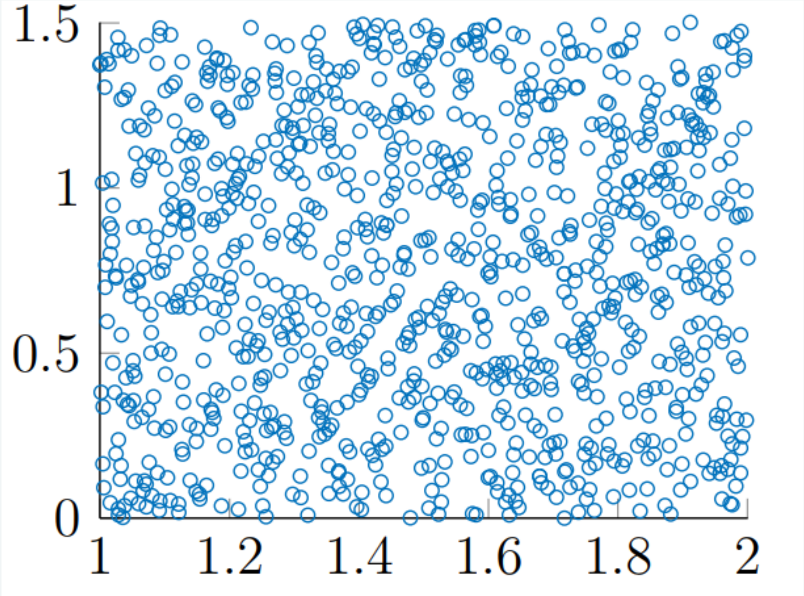
OBSF



$(X, Y) \in [-2, 2] \times [-2, 2]$

B
O
S
F

OISF



$(X, Y) \in [1, 2] \times [0, 1.5]$

2. Concept Illustration

Relationship among BOSF, OBSF and OISF

- We will set the bound in the following discussions.
- The shape you will obtained in the range space is highly determined by the bound parameter you inputted.

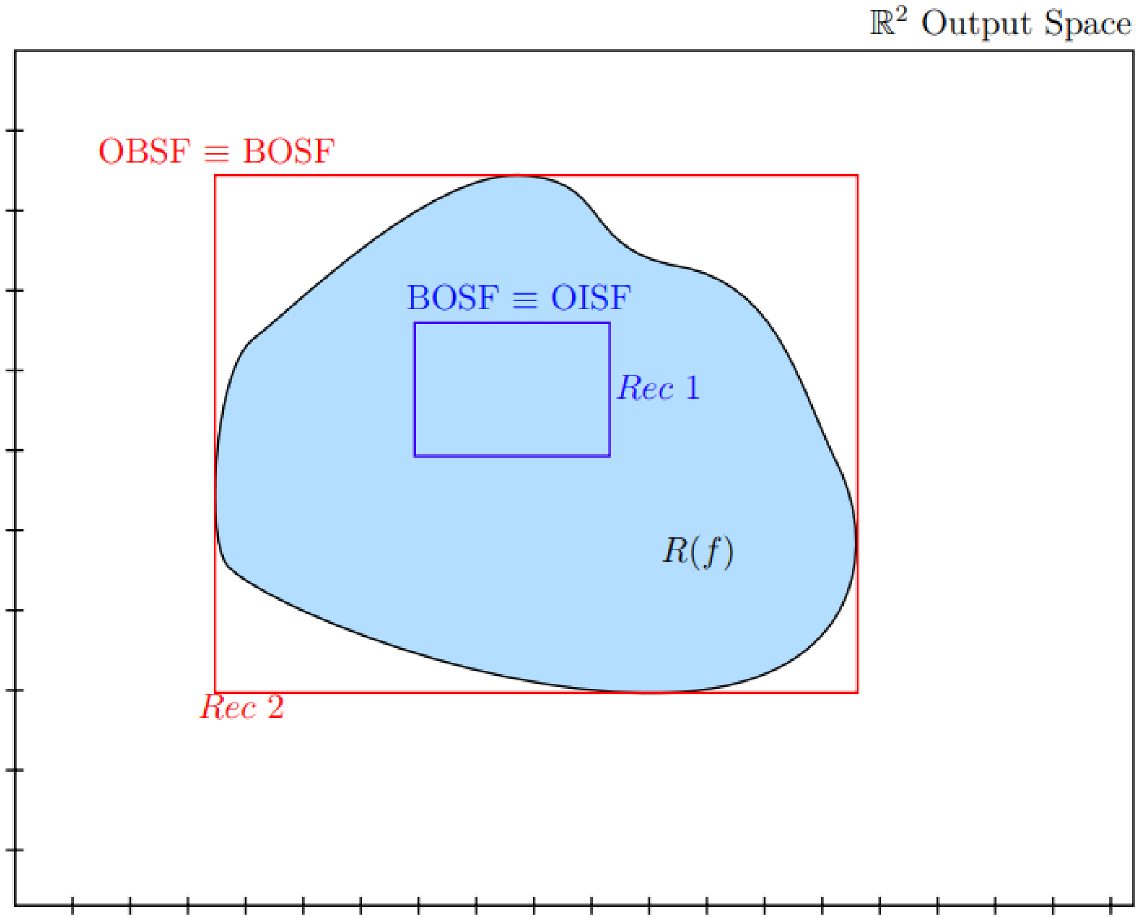
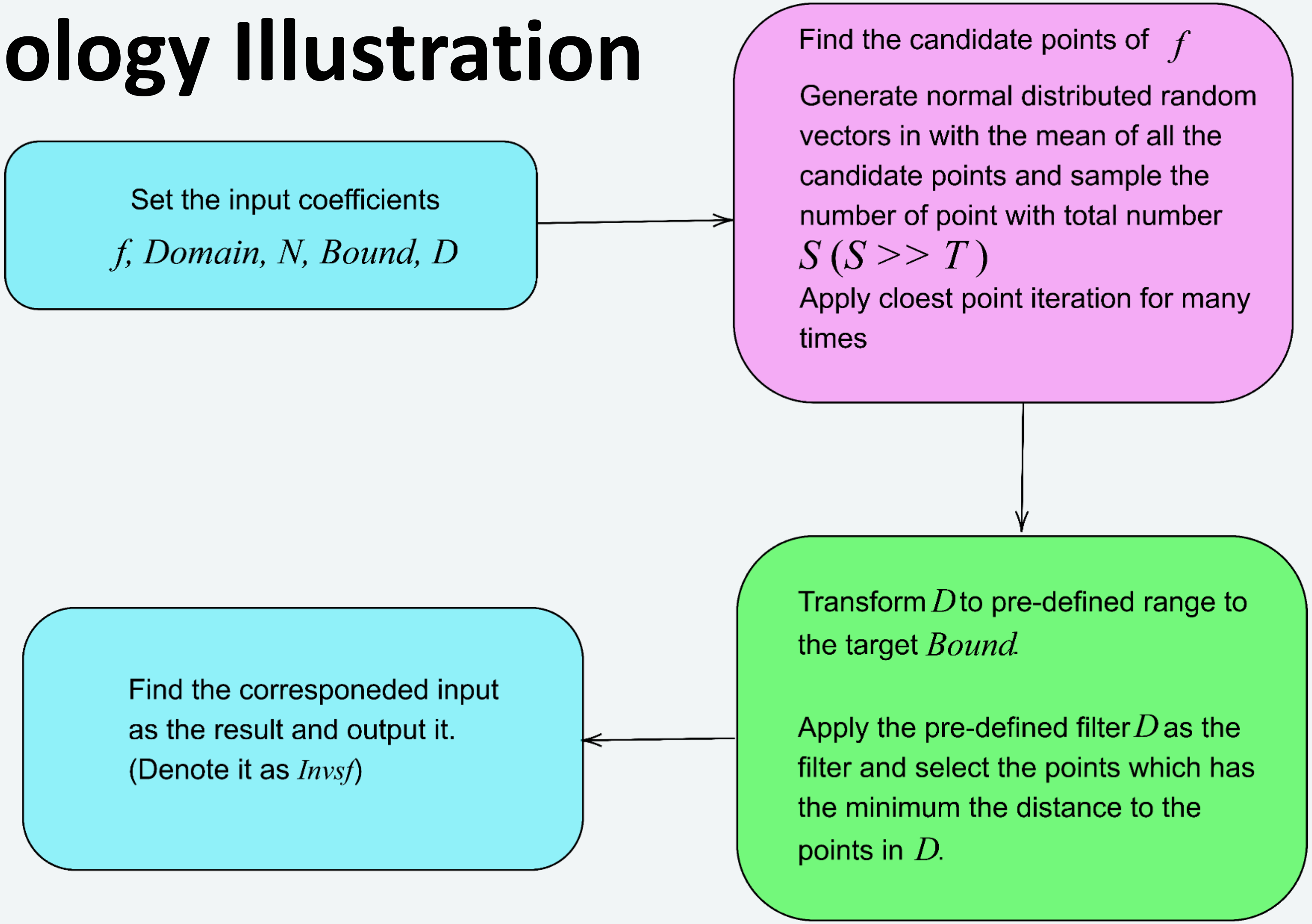


Figure 2.3: Illustration of three types output of OSF.

3. Methodology Illustration

Notations	Meaning
f	Input function
Domain	Input domain matrix should be written as $\text{Domain} = \begin{bmatrix} LB_1 & UB_1 \\ LB_2 & UB_2 \\ \dots & \dots \\ LB_m & UB_m \end{bmatrix}$ for m input domain.
N	The number of points we want to obtain for the result sample.
Bound	Predefined observable region space for output space should be written as $\text{Bound} = \begin{bmatrix} LBB_1 & UBB_1 \\ LBB_2 & UBB_2 \\ \dots & \dots \\ LBB_n & UBB_n \end{bmatrix}$ LBB: Lower Boundary Bound. UBB: Lower Boundary Bound.
D	The filter sample for generation. It should have the same size of N
InvSF	The output space-filling result

3. Methodology Illustration



3. Methodology Illustration

We define $f : X \longrightarrow Y \qquad X \in \mathbb{R}^m, Y \in \mathbb{R}^n$

$$\mathbb{R}^m \to \mathbb{R}^n \quad Y = [y_1, y_2, \ldots, y_n] = f(x_1, x_2, .., x_m) = f(X)$$

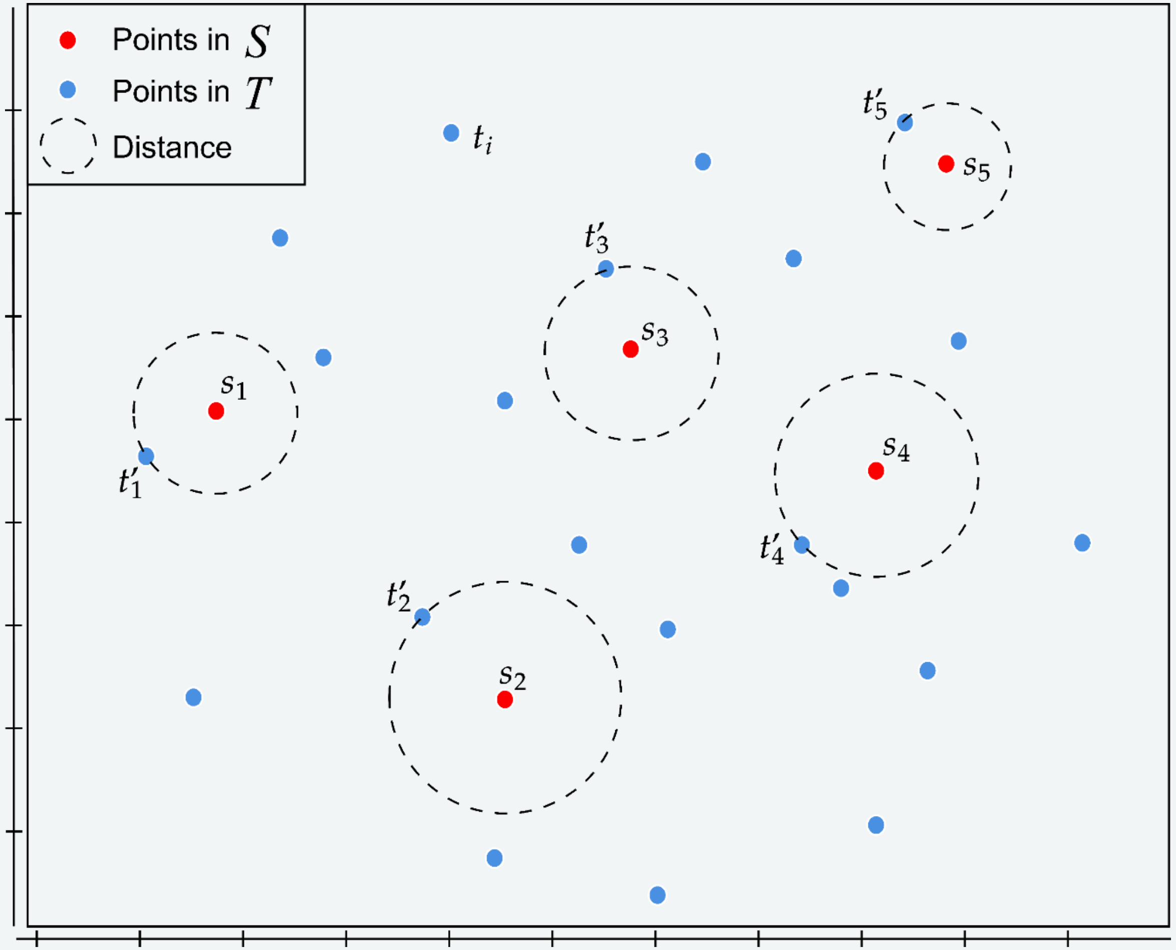
$$\begin{aligned} y_1 &= \tilde{f}_1(x_1, x_2, \ldots, x_m) \\ y_2 &= \tilde{f}_2(x_1, x_2, \ldots, x_m) \\ \cdots &= \qquad \cdots \\ y_i &= \tilde{f}_i(x_1, x_2, .., x_m) \\ \cdots &= \qquad \cdots \\ y_n &= \tilde{f}_n(x_1, x_2, \ldots, x_m) \end{aligned} \quad \Longleftrightarrow \quad Y = f(X) = [y_1, y_2, \ldots, y_n]$$

3. Methodology Illustration

Given two point sets S and T, the components of $S = \{s_1, s_2, ..., s_m\}$ and $T = \{t_1, t_2, ..., t_n\}$ have the same bound, and $|T| = n >> m = |S|$. We take the point t_i from T which has the minimum distance to a specific s_j. Rename it as t_j^* and find all of it to construct a new set $T^* = \{t_1^*, t_2^*, ..., t_m^*\}$. Then the pattern of T^* will approximate to the pattern in S, and denote it as $T^* \overset{\mathcal{D}}{\approx} S$.

t is Space-filling, s is the filtered point. And we increase the dense of t, then we will find that t_i* close to s_i

If we repeatedly find the closest point in among all the candidate points and then keep find the closest one. This process is named as **Closest Point Iteration**.



3. Methodology Illustration

Closest Point Iteration

First Iternation

filter points	s_1	s_2	s_3	s_4	...	s_N
First time selection	t_{11}^*	t_{12}^*	t_{13}^*	t_{14}^*	...	t_{1N}^*
Second time selection	t_{21}^*	t_{22}^*	t_{23}^*	t_{24}^*	...	t_{2N}^*
Third time selection	t_{31}^*	t_{32}^*	t_{33}^*	t_{34}^*	...	t_{3N}^*
...	...	...	...	...	...	...
k^{th} time selection	t_{k1}^*	t_{k2}^*	t_{k3}^*	t_{k4}^*	...	t_{kN}^*
...	...	...	...	...	...	...
n^{th} time selection	t_{n1}^*	t_{n2}^*	t_{n3}^*	t_{n4}^*	...	t_{nN}^*

$$t_j^{**} = \{t_{ij}^* | \min d(t_{ij}^*, s_j), i = 1, .., n; j = 1, ..., N\}$$

3. Methodology Illustration

Closest Point Iteration

k^{th} **Iternation**

filter points	s_1	s_2	s_3	s_4	...	s_N
First time selection	t_{11}^{k*}	t_{12}^{k*}	t_{13}^{k*}	t_{14}^{k*}	...	t_{1N}^{k*}
Second time selection	t_{21}^{k*}	t_{22}^{k*}	t_{23}^{k*}	t_{24}^{k*}	...	t_{2N}^{k*}
Third time selection	t_{31}^{k*}	t_{32}^{k*}	t_{33}^{k*}	t_{34}^{k*}	...	t_{3N}^{k*}
...	...	...	...	...	...	...
k^{th} time selection	t_{k1}^{k*}	t_{k2}^{k*}	t_{k3}^{k*}	t_{k4}^{k*}	...	t_{kN}^{k*}
...	...	...	...	...	...	...
n^{th} time selection	t_{n1}^{k*}	t_{n2}^{k*}	t_{n3}^{k*}	t_{n4}^{k*}	...	t_{nN}^{k*}

$$t_i^{(k+1)*} = \{t_{ij}^{k*} | \min d(t_{ij}^{k*}, s_j), i = 1, .., n; j = 1, ..., N\}$$

3. Methodology Illustration

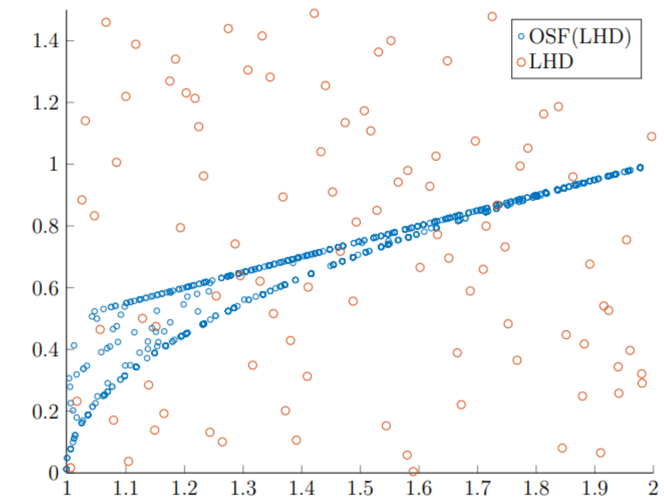
Advantage:

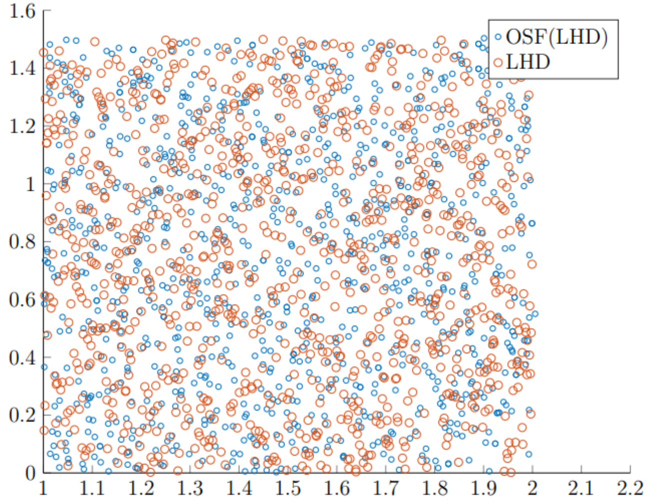
- It has a excellent performance of finding closest approximation. (Later I will illustrate)
- High adaptive property of output pattern.

Disadvantages:

- It does not works in the effect of range boundary.
- It is large consumption of time and computation.

OBSF





OISF

3. Methodology Illustration

Generation Methodology Illustration

maxfind	Find the mean value which located in the extreme points and max/min point .
ndim_Normal	Generate multi-dimensional normal distribution which takes the coordinate of points candidates founded in **maxfind**.
filter	Remove all the input vectors which are outside of the domain.
Min_dis_filter	Select the point which satisfy the minimum distance to every pre-defined points in D.

3. Methodology Illustration

Algorithm 1 Body Code

Input f, g, $Domain$, N, $Bound$, D.
Initialize $Invsf$.

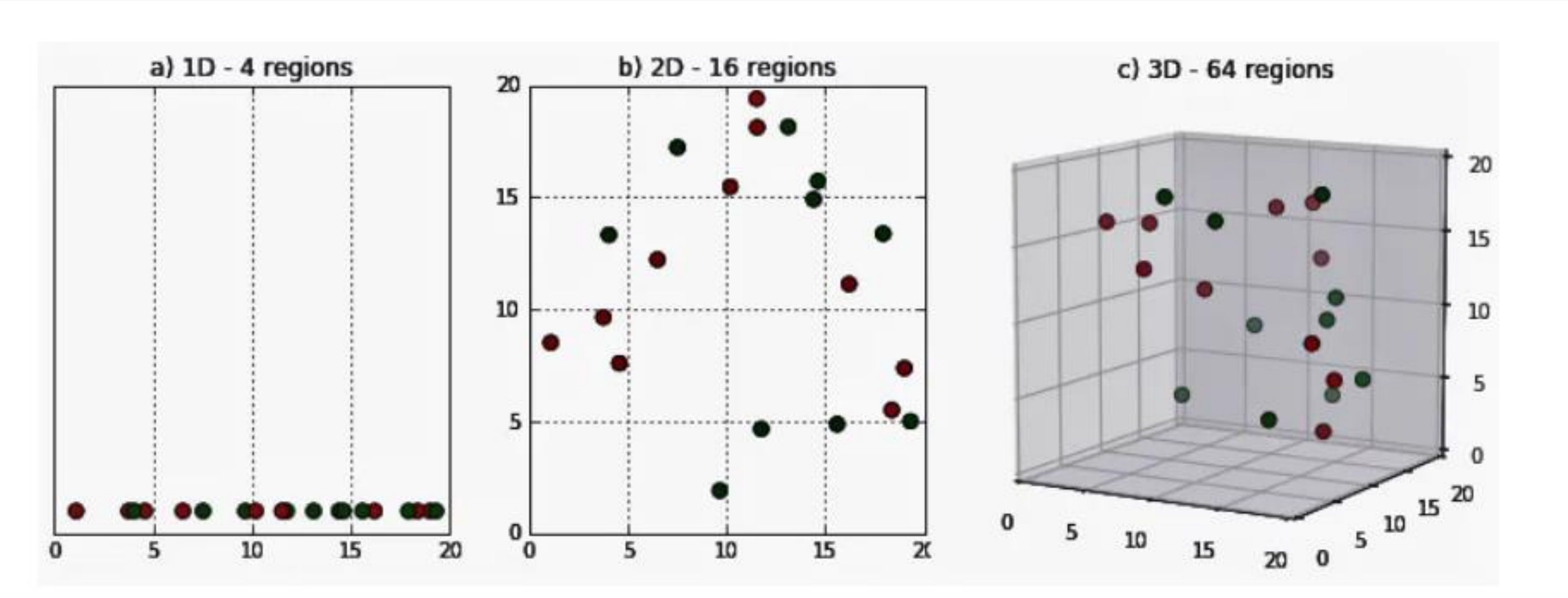
Obtain the number of rows in $Domain$, and denote it as a.
Find all the candidate mean value as MU via **maxfind** function.
Find candidates of input points via **ndim_Normal** function.
Cancel the points which are out of the domain via **filter** function.
Use **Min_dis_filter** function to select the proper points (minimum distance).

Repeat the parts insides two lines for n times ▷ Closest Point Interation.
Output $Invsf$.

3. Methodology Illustration

Curse of Dimensionality

$$C = 10^m \times N$$



https://medium.com/analytics-vidhya/the-curse-of-dimensionality-and-its-cure-f9891ab72e5c

3. Methodology Illustration

Algorithm 2 Max_find function.

Input coefficients f, g and $Domain$.
Initialize matrix M.
if f is uni-dimensional function **then**
  Find the mid value of f, which is $f(\frac{LB+UB}{2})$.
  Find all the extreme value point of f in all the marginal function $f(x)$.
else
  Find all the extreme value point of f in all the marginal function $\tilde{f}_i(x)$.
  Find all the mid-value of f in all marginal function $\tilde{f}_i(x)$, which is $f(\frac{LB_i+UB_i}{2})$ and add its input to matrix M.
end if

Record all the types of Inputs in matrix M and output it.

3. Methodology Illustration

Algorithm 3 ndim_normal function.

Input k, mu. ▷ k is the total number of point will be generated.
Initialize $Input$. ▷ mu is the mean.
Obtain the size of mu as a $a \times b$ matrix.
for j from 1 to a **do**
  Generate the Covariance matrix $Covmat$.
  ▷ we usually use$Covmat$ as a $diag\{10, ..., 10\}_{b \times b}$ matrix.
  Initialize $i = 0$.
  while $i < k$ **do** ▷ Iteratively generate every input.
    Generate the normal distributed random vector number r, which has a mean $mu(j, :)$ and variance. $Covmat$.
    $i = i + 1$.
  end while
  Collect all he values of r by matrix $Input$. And then output it.
end for

3. Methodology Illustration

Algorithm 4 filter function.

Input two matrices *Input* and *Domain*.

Initialize the *Out* as $a-$row, $b-$column matrix.

for i from 1 to a **do**

  for j from 1 to b **do**

    Cancel all the points in *Input*, which has at least one value out of the corresponded domain.

    And denote the new *Input* as *Out*

  end for

end for

Output matrix *Out*.

3. Methodology Illustration

Algorithm 5 Mis_dis_filter function.

Input variables D_1, f_D_1, $Bound$, D ▷ D_1: the sampling matrix input.
Initialize matrix Inv. ▷ f_D_1: the sampling matrix output.
Find the size of D as $n \times s$ ▷ n-row, s-column matrix.
▷ D is the filtered matrix.

for i from 1 to s **do**
    Transform the range of values D to $Bound$.
    Denote the transformed range of matrix D as D'.
end for
Find all the items in f_D_1 which has the closest distance to every point in D'.
Add all the points which are satisfying the condition of last line into matrix Inv and output it.

4. Computational Verification

Monte Carlo RPs: MCRPs
Quasi-Monte Carlo RPs: QMCRPs
Minimum Mean Square Error RPs: MMSERPs
Minimum Energy RPs: MERPs

Normal Kernel Density Function as Sample:

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - r_i) = \frac{1}{nh\sqrt{2\pi}}\sum_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{x-r_i}{h}\right)^2}$$

4. Computational Verification

Uni-output Function

$$f_1(x) = 0.7 \times N(8,10) + 0.3 \times N(1.5,1) = \frac{0.7}{\sqrt{20\pi}} e^{-\frac{1}{2}(\frac{x-8}{\sqrt{10}})^2} + \frac{0.3}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-1.5)^2}$$

$$f_2(x) = \chi^2(6) = \frac{1}{2^3\Gamma(3)} x^3 e^{-\frac{x}{2}} = \frac{x^3 e^{-\frac{x}{2}}}{16}, x \in [0, +\infty)$$

$$f_3(x) = 5e^{-5x}, x \in [0, +\infty)$$

$$f_4(x) = B(0.5, 0.5; x) = \frac{x^{0.5-1}(1-x)^{0.5-1}}{B(0.5, 0.5)} = \frac{1}{\pi\sqrt{x(1-x)}}, x \in (0,1)$$

$$f_5(x) = \Gamma(3, 2; x) = \frac{8}{\Gamma(3)} x^2 e^{-2x}$$

$$f_6(x) = \frac{e^{-\frac{(x-2)}{3}}}{3(1+e^{-\frac{(x-2)}{3}})^2}$$

4. Computational Verification

Uni-output Function

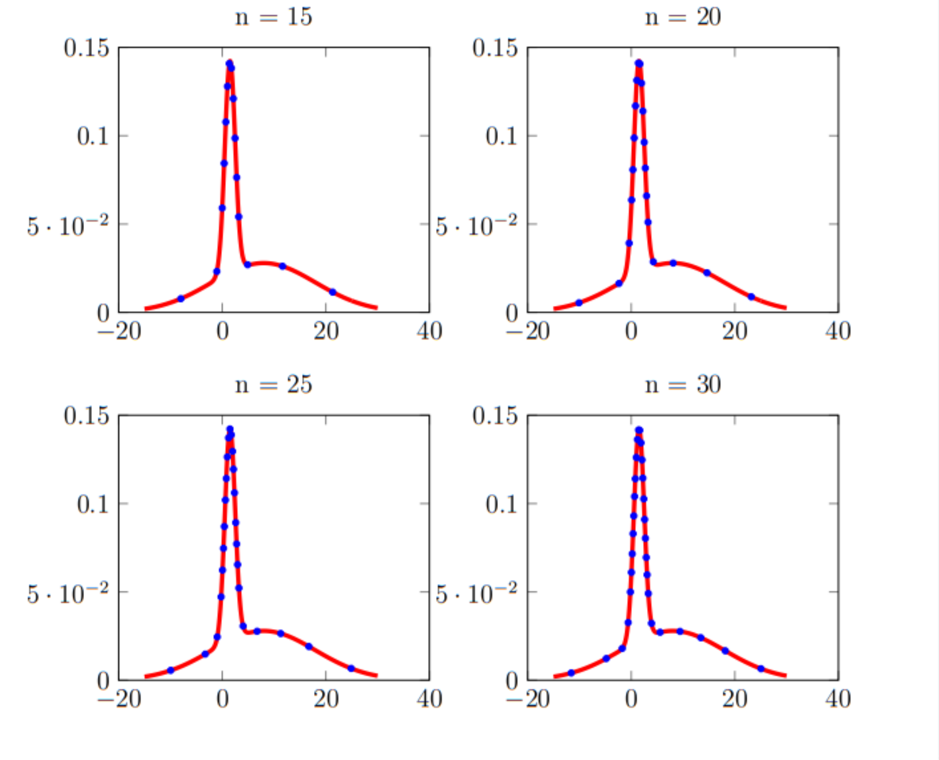
Generated Results for f_4

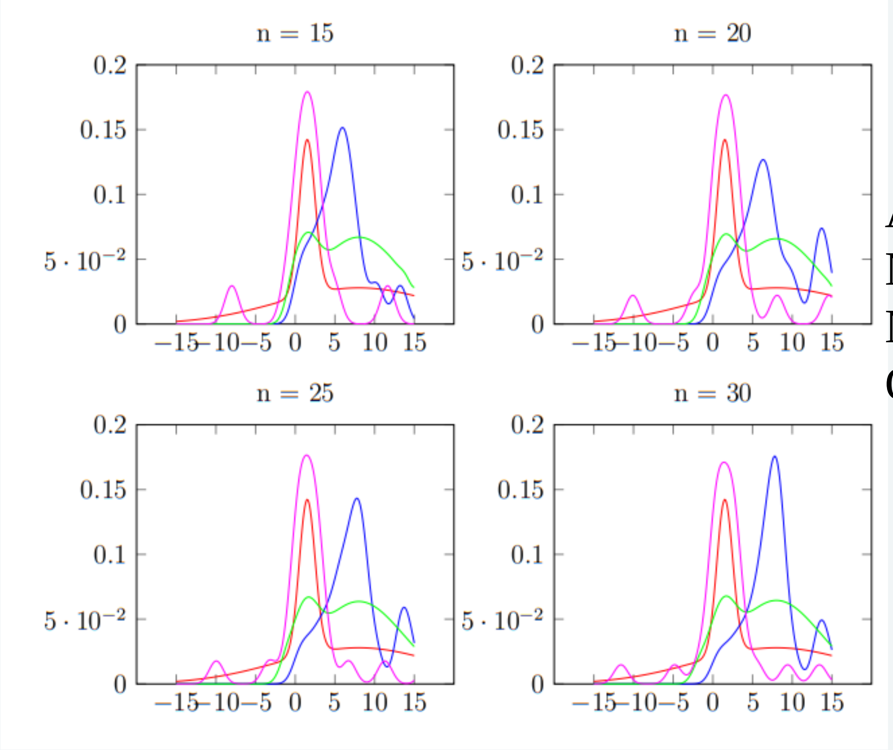
Input Sample for f_4																				
$n = 1$	0.443113978																			
$n = 2 \sim 12$	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
$n = 13$	0.011219565	0.015685377	0.026456324	0.076542513	0.151587185	0.318662692	0.493439896	0.68066061	0.833550461	0.92392586	0.971658778	0.984382085	0.989672495							
$n = 14$	0.011361914	0.021563652	0.036434118	0.067398355	0.143683456	0.257397257	0.392808197	0.590175631	0.735715252	0.866477973	0.948377291	0.973392356	0.985132452	0.989088605						
$n = 15$	0.010366291	0.014804369	0.02126863	0.051897083	0.099587676	0.187633044	0.331421473	0.51794091	0.684060299	0.797823204	0.88463242	0.940097947	0.972072037	0.985679482	0.989191182					
$n = 16$	0.011517282	0.016059324	0.02651643	0.047838484	0.080964413	0.1382628	0.257908827	0.425751574	0.594484573	0.730340039	0.851100077	0.909893927	0.957152631	0.978308961	0.98618045	0.989863069				
$n = 17$	0.010030374	0.013649364	0.019519724	0.033781559	0.062352793	0.115299839	0.23041299	0.361180497	0.528693988	0.665554984	0.78785453	0.892337608	0.941724677	0.965742599	0.977194531	0.987015231	0.989548752			
$n = 18$	0.010267329	0.013114268	0.022170526	0.032683179	0.049025206	0.094578285	0.166793353	0.268723919	0.41868575	0.577216888	0.726921686	0.83545174	0.915004385	0.953373648	0.973321272	0.982521125	0.986804853	0.98991328		
$n = 19$	0.010117066	0.012360502	0.016056356	0.022811434	0.039921745	0.070354206	0.124932555	0.212965004	0.3492925	0.528804213	0.675101341	0.796919801	0.868477204	0.920631524	0.955834618	0.973693647	0.983578885	0.987648692	0.990255485	
$n = 20$	0.009675273	0.012317045	0.015763183	0.021654202	0.038207185	0.069429476	0.112549592	0.184100744	0.308035711	0.455088389	0.604837518	0.745070761	0.841966235	0.916951439	0.955159536	0.972754987	0.979918218	0.985277988	0.988454325	0.990202243
$n = 21 \sim 30$	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

4. Computational Verification

Uni-output Function

$$f_1(x) = 0.7 \times N(8,10) + 0.3 \times N(1.5,1) = \frac{0.7}{\sqrt{20\pi}} e^{-\frac{1}{2}\left(\frac{x-8}{\sqrt{10}}\right)^2} + \frac{0.3}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-1.5)^2}$$



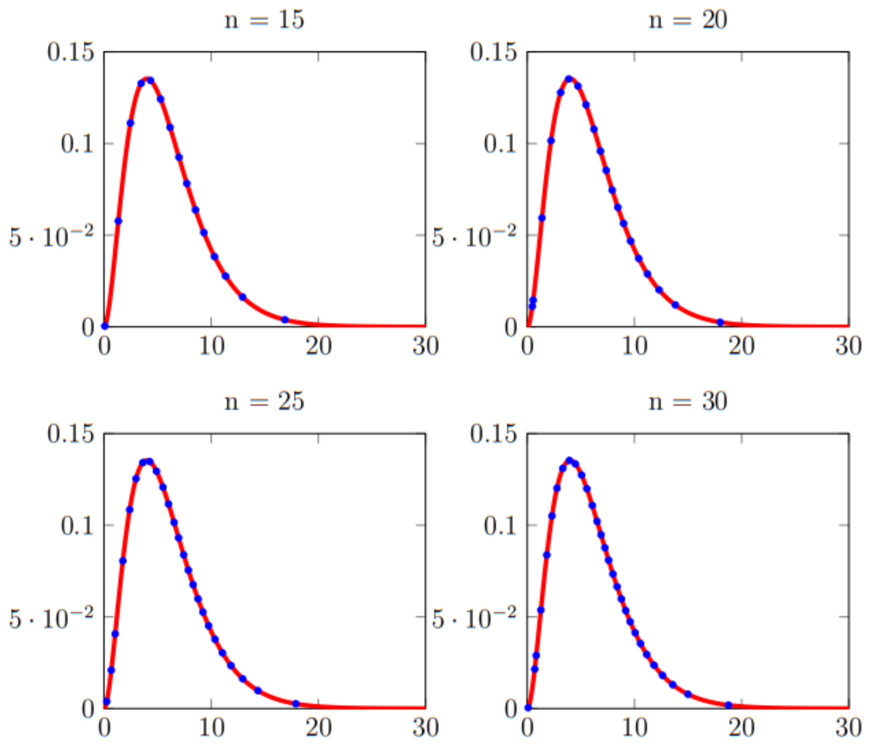


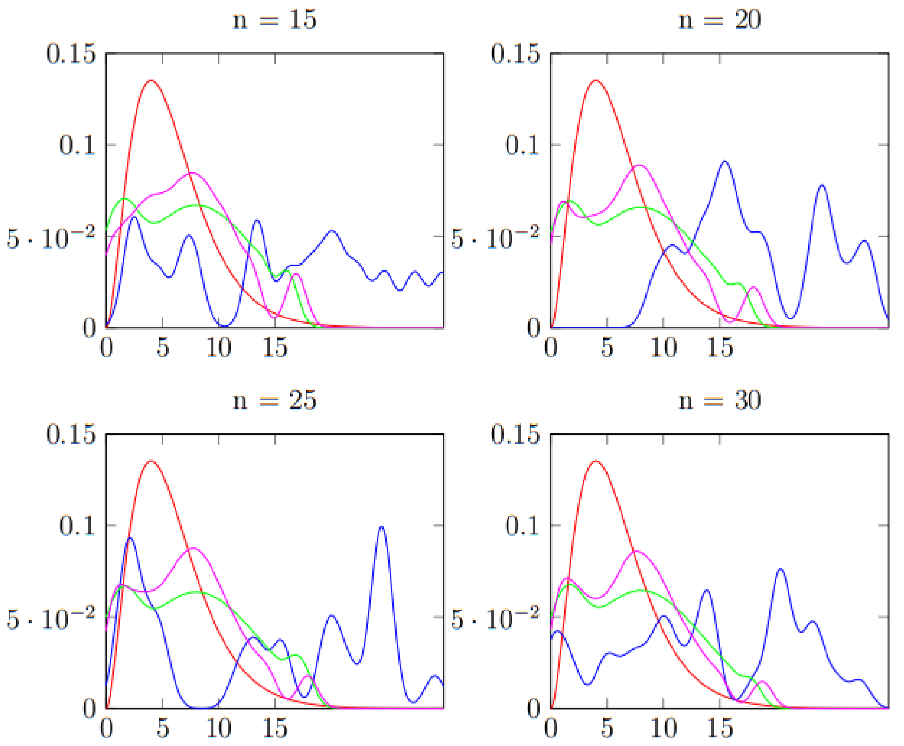
Actual Function: Red
MERPs KDE: Blue
MMSERPs KDE: Green
OSF Sample KDE: Magenta

4. Computational Verification

Uni-output Function

$$f_2(x) = \chi^2(6) = \frac{1}{2^3\Gamma(3)} x^3 e^{-\frac{x}{2}} = \frac{x^3 e^{-\frac{x}{2}}}{16}, x \in [0, +\infty)$$



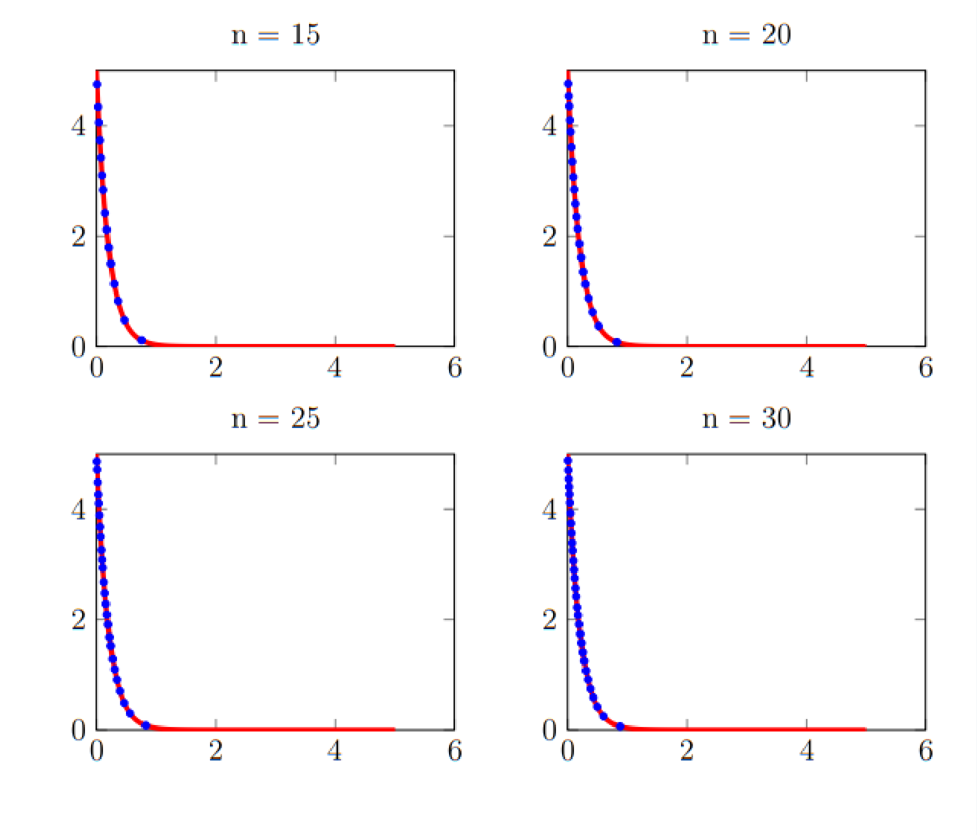


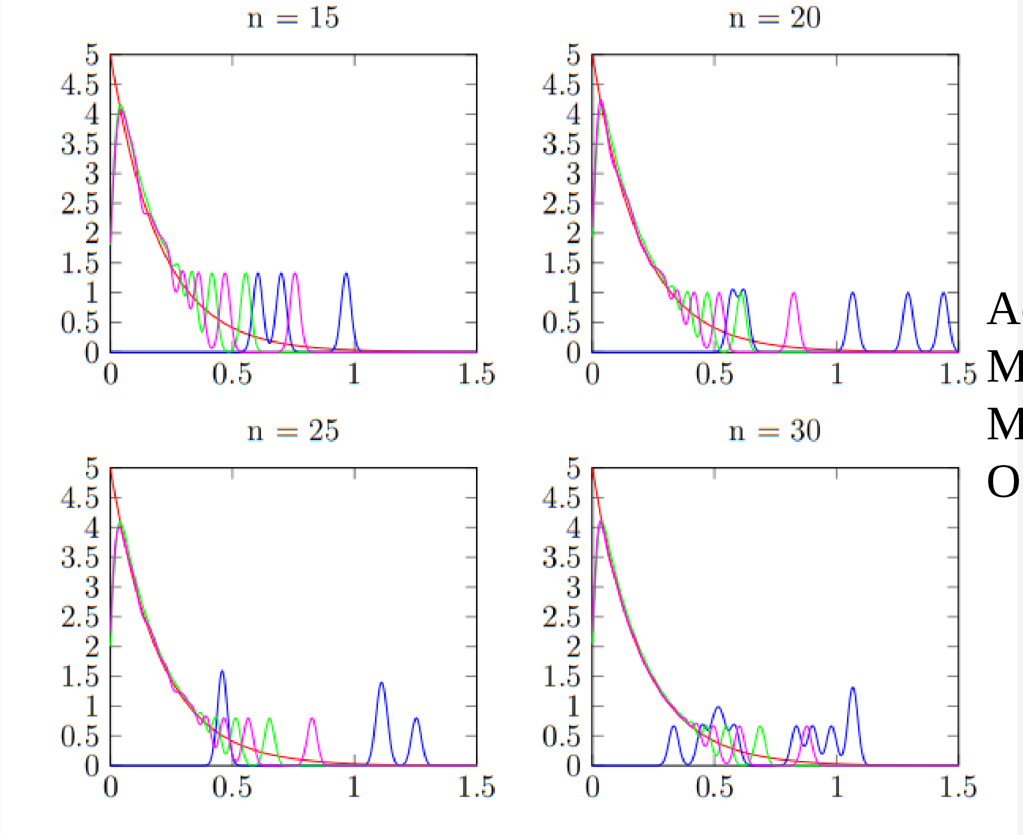
Actual Function: Red
MCRPs KDE: Blue
MMSERPs KDE: Green
OSF Sample KDE: Magenta

4. Computational Verification

Uni-output Function

$$f_3(x) = 5e^{-5x}, x \in [0, +\infty)$$





Actual Function: Red
MCRPs KDE: Blue
MMSERPs KDE: Green
OSF Sample KDE: Magenta

4. Computational Verification

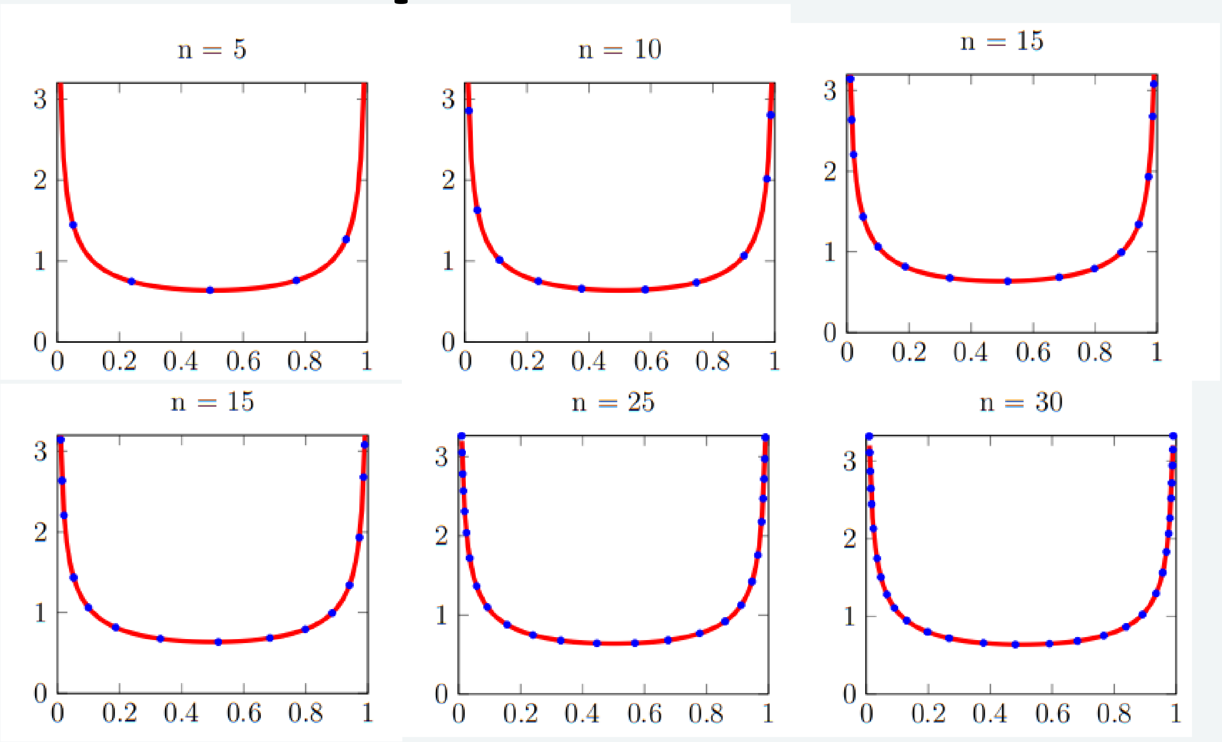
Uni-output Function

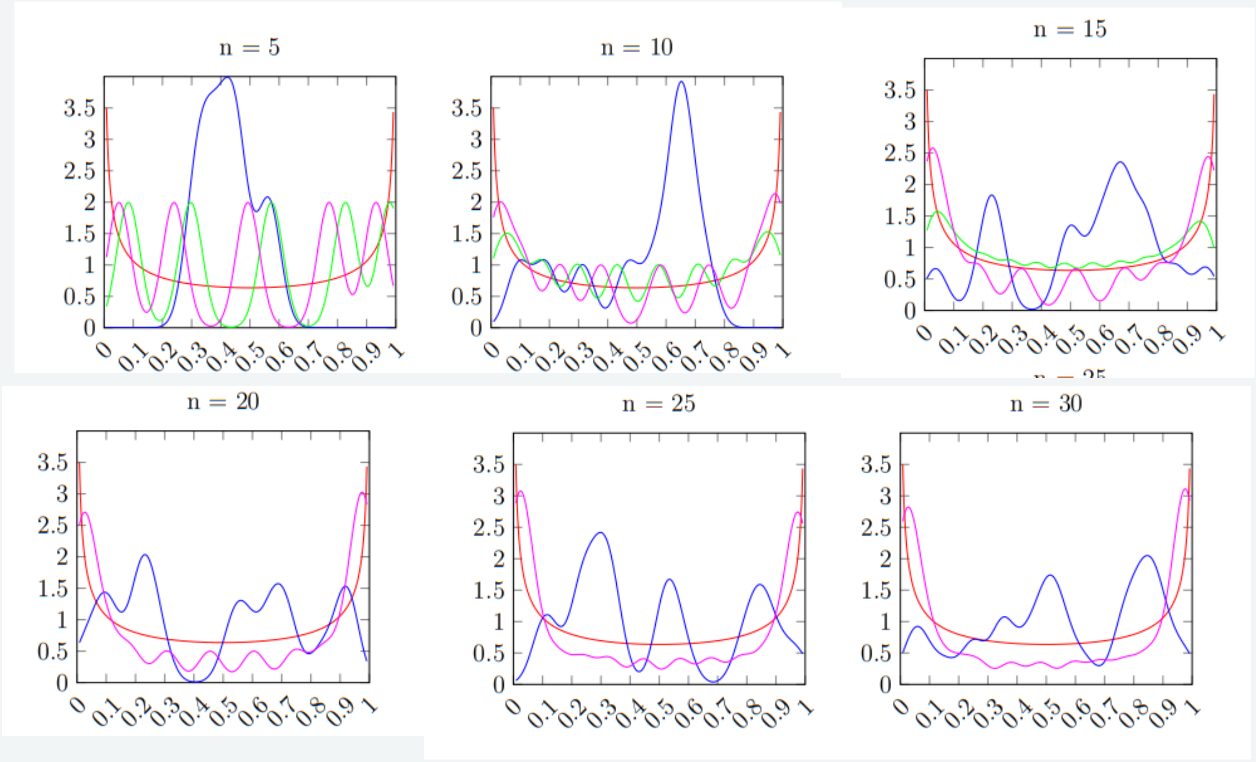
$$f_4(x) = B(0.5, 0.5; x) = \frac{x^{0.5-1}(1-x)^{0.5-1}}{B(0.5, 0.5)} = \frac{1}{\pi\sqrt{x(1-x)}}, x \in (0, 1)$$

$$\lim_{x \to 0^+} f_4(x) = \lim_{x \to 1^-} f_4(x) = +\infty$$

4. Computational Verification

Uni-output Function





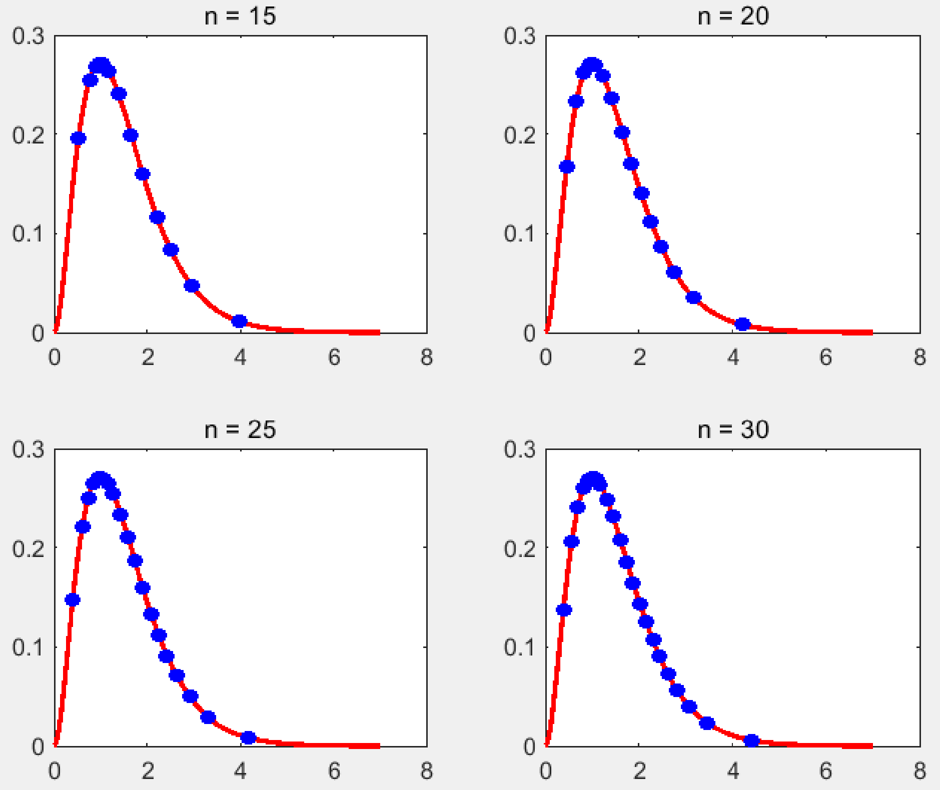
$$f_4(x) = B(0.5, 0.5; x) = \frac{x^{0.5-1}(1-x)^{0.5-1}}{B(0.5, 0.5)} = \frac{1}{\pi\sqrt{x(1-x)}}, x \in (0,1)$$

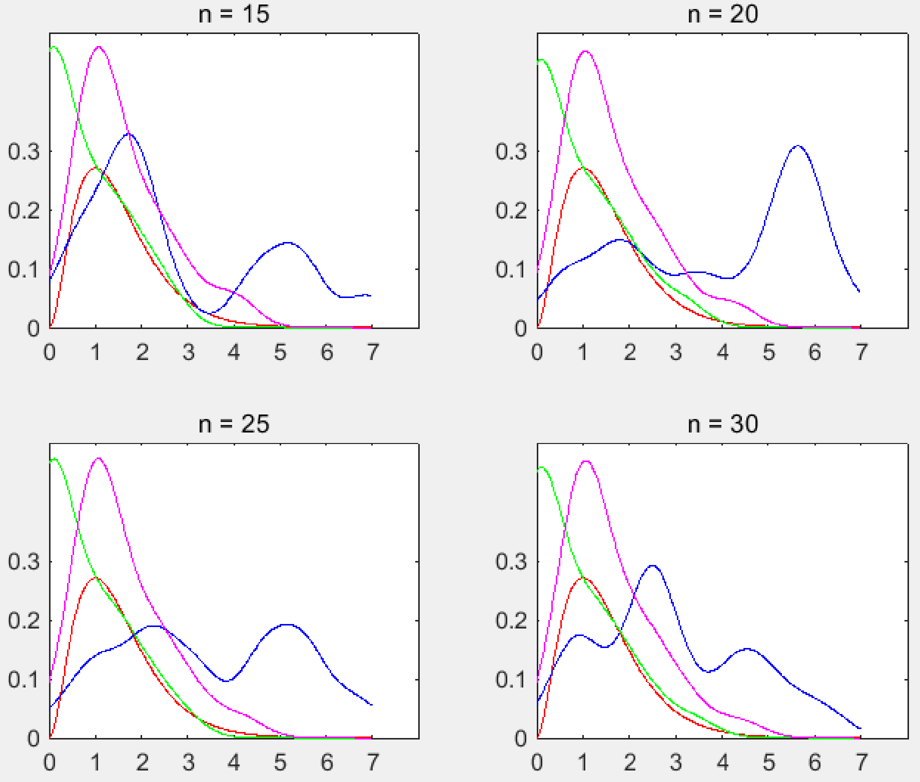
Actual Function: Red
MCRPs KDE: Blue
MMSERPs KDE: Green
OSF Sample KDE: Magenta

4. Computational Verification

Uni-output Function

$$f_5(x) = \Gamma(3, 2; x) = \frac{8}{\Gamma(3)} x^2 e^{-2x}$$



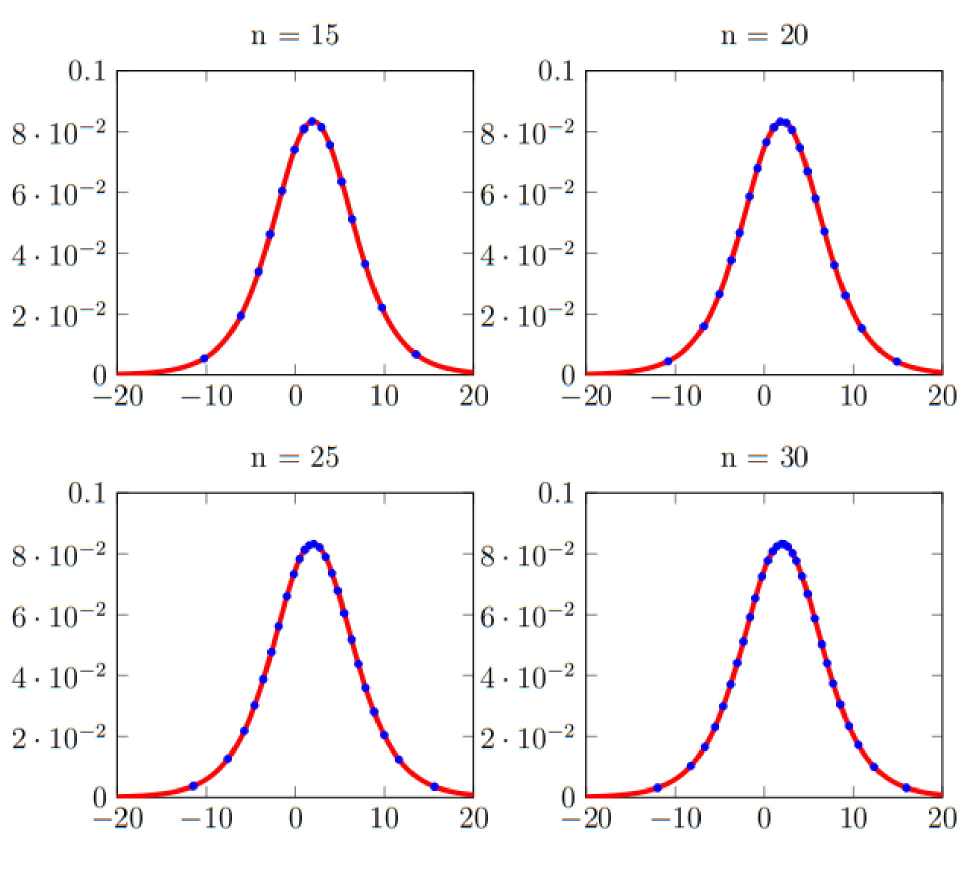


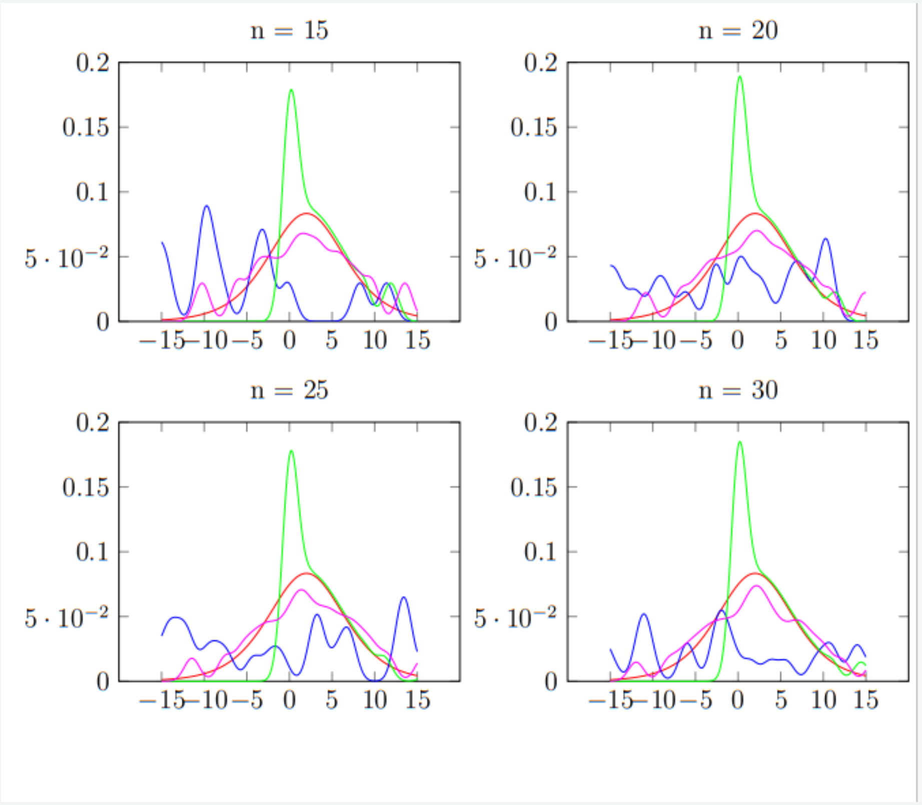
Actual Function: Red
MCRPs KDE: Blue
MMSERPs KDE: Green
OSF Sample KDE: Magenta

4. Computational Verification

Uni-output Function

$$f_6(x) = \frac{e^{-\frac{(x-2)}{3}}}{3(1+e^{-\frac{(x-2)}{3}})^2}$$





Actual Function: Red
MCRPs KDE: Blue
MMSERPs KDE: Green
OSF Sample KDE: Magenta

4. Computational Verification

Multi-output Function

Cosine Similarity

Cosine similarity is a measure of similarity between two non-zero vectors defined in an inner product space.

$$S(a,b) = \frac{||a \cdot b||}{||a|| \cdot ||b||} = \frac{\sum_{i=1}^{n} a_i \times b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\sqrt{\sum_{i=1}^{n} b_i^2}}$$

When the value of $S(a,b)$ approximate to 1, which means the point a and b are similar to each other. On the country, it means that the a and b far away.

4. Computational Verification

Multi-output Function

$$(X,Y) = f_7[x,y] = \left(\frac{1}{\sqrt{x^2+y^2+0.1}}, \arctan\frac{y}{x}\right), x \in [0,1], y \in [0,1]$$

$$(X,Y) = f_8[x,y] = \left(x^2+y^2, xy\right), x \in [0,2], y \in [0,2]$$

Function	Label	min	max	mean	median
f_7	OSF(LHD) v.s. LHD	0.9939	1.000	0.9993	0.9997
f_8	OSF(LHD) v.s. LHD	0.9937	1.000	0.9994	0.9997

Table 5.3: Cosine Similarity of f_7 and f_8

5. Conclusion & Future Work

- Develop a test method to test the approximated distribution of two discrete sample point in any dimensional space.
- Enhancing Current NFF methodology for higher speed and accuracy.
- Apply NFF to the marginal detection.
- Avoid OBSF automatically, if we only want to obtain OISF?
- Generalized OSF sample in RPs perspective.
- Combined with Machine Learning, which can be used as training dataset.

Happy Christmas

博文雅志 真知笃行

In knowledge and in deeds, unto the whole person

Reference

[1] Elsawah, A. M., Wang, Y. A., and Tank, F. (2023). Minimum energy representative points. Journal of Computational and Applied Mathematics, 438, 115526.

[2] Wang, S, Generale, A. P, Kalidindi, S. R. and Joseph, V. R. (2023). Sequential Designs for Filling Output Spaces. arXiv preprint arXiv:2305.07202.

[3] Lu, L. and Anderson-Cook, C. M. (2021). Input-response space-filling designs. Quality and Reliability Engineering International, 37(8), 3529-3551.

[4] Viana, F. A, Venter, G. and Balabanov, V. (2010). An algorithm for fast optimal Latin hypercube design of experiments. International journal for numerical methods in engineering, 82(2), 135-156.

[5] Krishna, A., Craig, S. R., Shi, C. and Joseph, V. R. (2022). Inverse design of acoustic metasurfaces using space-filling points. Applied Physics Letters, 121(7), 071701.

[6] Banach, S. (1922). Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. Fundamenta mathematicae, 3(1), 133-181.

[7] 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.

[8] Krishna, A., Tran, H., Huang, C., Ramprasad, R. and Joseph, V. R. (2023). Adaptive exploration and optimization of materials crystal structures. INFORMS Journal on Data Science.

[9] Jourdan, A. (2021). Space-filling designs with a Dirichlet distribution for mixture experiments.

Reference

[10] Hoyer, P. O. (2004). Non-negative matrix factorization with sparseness constraints. Journal of machine learning research, 5(9).

[11] DiCiccio, T. J. and Efron, B. (1996). Bootstrap confidence intervals. Statistical science, 11(3), 189-228.

[12] Yule, G. U. and Kendall, M. G. (1911). An introduction to the theory of statistics. Charles Griffin and Company. London UK, 147-148.

[13] Pearson, K. (1901). On Lines and Planes of Closest Fit to Systems of Points in Space, Philosophical Magazine, 2 (11): 559–572. http://doi.org/10.1080/ 14786440109462720.

[14] Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. Journal of educational psychology, 24(6), 417. [15] Hotelling, H. (1936). Relations between two sets of variates. Biometrika, 28 (3/4),321–377. http://doi.org/10.2307/2333955.

[16] Elsawah, A. M., Wang, Y. A., Celem, S. M. and Qin, H. (2023). A novel technique for constructing nonregular nine-level designs: Adjusted multiple tripling technique. Journal of Computational and Applied Mathematics, 424, 115016.

[17] Gnedenko, B. V., Kolmogorov, A. N., Doob, J. L. and Hsu, P. L. (1968). Limit distributions for sums of independent random variables (Vol. 233). MA: Addisonwesley.

[18] Duong, T., & Hazelton, M. L. (2005). Convergence rates for unconstrained bandwidth matrix selectors in multivariate kernel density estimation. Journal of Multivariate Analysis, 93(2), 417-433

Reference

[19] Cohen, Irwin., and Bustard, Thomas. (1966). Atomic orbitals: Limitations and variations. Journal of Chemical Education, 43(4), 187.www.doi.org/10.1021/ ed043p187
[20] Cao, Y., Liang, J., and Gao, N. (2022). A new method for identifying significant genes from gene expression data. Biom Biostat Int J, 11(4), 140-146.
[21] Joseph, V. R. (2016). Space-filling designs for computer experiments: A review. Quality Engineering, 28(1), 28-35.
[22] Cao, Y., Liang, J., Gao, N., and Sun Z. (2023). Multiple Mean Comparison for Gene Expression Data via F -Type Tests under High Dimension with A Small Sample Size. American Journal of Biomedical Science and Research, 18(3), 306- 314.
[23] Elsawah, A. M., Wang, Y. A., and Tank, F. (2024). Minimum energy representative points. Journal of Computational and Applied Mathematics, 438, 115526.
[24] Newman, D. (1939). The distribution of range in samples from a normal population, expressed in terms of an independent estimate of standard deviation. Biometrika, 31(1/2), 20-30.
[25] Gumbel, E. J. (1947). The distribution of the range. The Annals of Mathematical Statistics, 18(3), 384-412.
[26] Feller, W. (1951). The asymptotic distribution of the range of sums of independent random variables. The annals of mathematical statistics, 427-432.
[27] Wallace, D. L. (1958). Asymptotic approximations to distributions. The Annals of Mathematical Statistics, 29(3), 635-654.

Reference

[28] Snapp, R. R., and Venkatesh, S. S. (1998). Asymptotic expansions of the k nearest neighbor risk. The Annals of Statistics, 26(3), 850-878.
[29] Julier, S. J., and Uhlmann, J. K. (2004). Unscented filtering and nonlinear estimation. Proceedings of the IEEE, 92(3), 401-422.
[30] Jaffer, A. (2014). Recurrence for Pandimensional Space-Filling Functions. arXiv preprint arXiv:1402.1807.
[31] Johnson, M. E., Moore, L. M., and Ylvisaker, D. (1990). Minimax and maximin distance designs. Journal of Statistical Planning and Inference, 26, 131–148.
[32] Yang, L., Zhou, Y., and Liu, M. Q. (2021). Maximin distance designs based on densest packings. Metrika, 84(5), 615-634.
[33] Fang, K., Zhou, M., and Wang, W. (2014). Applications of the representative points in statistical simulations. Science China Mathematics, 57, 2609-2620.
[34] Jiang, J. J., He, P., and Fang, K. T. (2015). An interesting property of the arcsine distribution and its applications. Statistics & Probability Letters, 105, 88-95.
[35] Xu, L. H., Fang, K. T., and He, P. (2022). Properties and generation of representative points of the exponential distribution. Statistical Papers, 1-27.
[36] Li, Y., Fang, K. T., He, P., and Peng, H. (2022). Representative points from a mixture of two normal distributions. Mathematics, 10(21), 3952.


Massive Thanks