Global Sensitivity Analysis for Interpretation of Black Box Functions

THE UNIVERSITY OF ADELAIDE AUSTRALIA

SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

Rastko Zivanovic

Goal: Convert a Black Box function to the functional ANOVA format (suitable for sensitivity analysis) using the smallest number of a Black Box function evaluations.

Compare: Quasi-Monte Carlo (*Quasi-regression*),
Sparse Grid (*Sparse Grid Regression*) and
Tensor Decomposition (*TT-regression*)

Example: smooth and continuous function (feed-forward neural network) obtained through a machine learning technique.

$$f(\mathbf{x}) = b_o + \sum_{i=1}^{d} w_{i\to o} x_i + \sum_{h=1}^{H} w_{h\to o}\, \phi(b_h + \sum_{i=1}^{d} w_{i\to h} x_i), \qquad \phi(z) = (1+\exp(-z))^{-1}$$

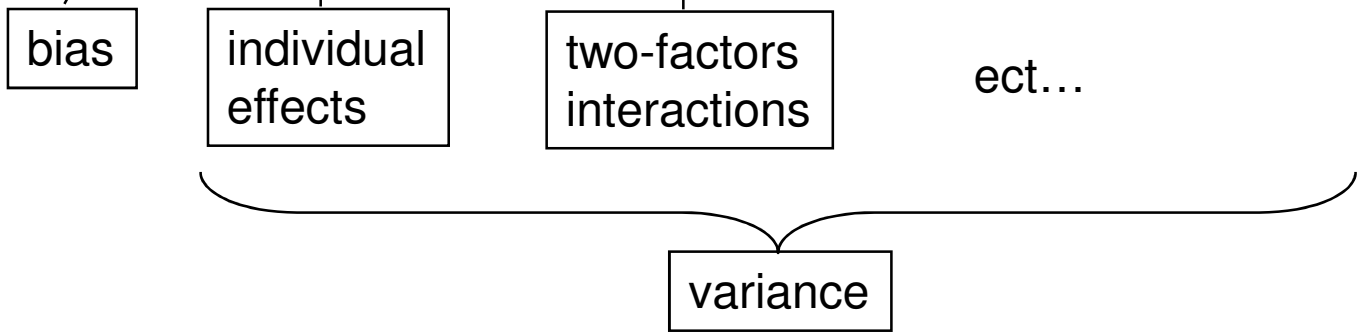
Although black box functions, like the neural network, are able to map input/output relations in data, they are not suitable for *interpretation* and *sensitivity analysis*.

Functional ANOVA

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

✓Unique decomposition if integrals of every component over any of its own variables equal to zero and components are orthogonal:

$$f(x_1,\ldots,x_d) = f_0 + \sum_{i=1}^{d} f_i(x_i) + \sum_{i=1}^{d}\sum_{j=i+1}^{d} f_{ij}(x_i,x_j) + \ldots + f_{1,2,\ldots,d}(x_1,\ldots,x_d).$$



Example *d*=3 ➔ terms: $2^3 = 8$

$$f(\mathbf{x}) = f_0 + f_1(x_1) + f_2(x_2) + f_3(x_3) + f_{1,2}(x_1,x_2) + f_{1,3}(x_1,x_3) + f_{2,3}(x_2,x_3) + f_{1,2,3}(x_1,x_2,x_3).$$

Approximation of the component functions in the functional ANOVA

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

$$f(\mathbf{x}) = f_0 + \sum_{u \subseteq \{1,2,\ldots,d\}} f_u(\mathbf{x}_u), \longrightarrow f_u(\mathbf{x}_u)$$ - a *component function* depends only on those x_i -s of factor-vector **x** which indices are in the set u

To approximate *component functions* we work via the parameterization:

$$f(\mathbf{x}) = \sum_{\mathbf{r} \in \mathrm{B}_\infty} \beta_{\mathbf{r}} \Psi_{\mathbf{r}}(\mathbf{x}) = \sum_{r_1=1}^{\infty} \sum_{r_2=1}^{\infty} \cdots \sum_{r_d=1}^{\infty} \beta_{r_1, r_2, \cdots, r_d} \Psi_{r_1, r_2, \cdots, r_d}(\mathbf{x})$$

$\mathbf{r} = (r_1, r_2, \ldots, r_d) \in \mathrm{Z}_+^d$ ➔ multi-index vector, $\mathrm{Z}_+ = \{0, 1, 2, \ldots\}$

$\Psi_{\mathbf{r}}(\mathbf{x}) = \prod_{i=1}^{d} \varphi_{r_i}(x_i)$ ➔ tensor product of orthogonal polynomials (Legendre)

B_∞ ➔ infinite set of multi-index vectors

Approximation of the component functions in the functional ANOVA

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

- Use of the *orthogonal basis functions* has an important implication: the contributions of individual terms in the model are independent and their significance can be measured by estimating the corresponding coefficients $\beta_{\mathbf{r}}$

- The coefficients that minimize the *least squares* objective function can be found by solving the following multi-dimensional integrals

$$\beta_{\mathbf{r}} = \int_{\mathrm{I}^d} f(\mathbf{x})\, \Psi_{\mathbf{r}}(\mathbf{x})\, d\mathbf{x}$$

$$\mathrm{I}^d = \left\{ \mathbf{x} \in \mathrm{R}^d : 0 \le x_i \le 1, 1 \le i \le d \right\}$$

domain of factors have been mapped on to the unit hypercube

- Related to the functional ANOVA via the expression for the component functions

$$f_u(\mathbf{x}) = \sum_{\mathbf{r} \in \mathrm{B}_u} \beta_{\mathbf{r}} \Psi_{\mathbf{r}}(\mathbf{x})$$

$$\mathrm{B}_u \subset \mathrm{B}_\infty$$

$$\mathrm{B}_u = \left\{ \mathbf{r} \;\middle|\; r_i > 0 \quad \text{and} \quad i \in u \right\}$$

Automated model selection

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

The automated model selection procedure needs to start with:

1. user selection of the factors that are expected to play a role in a model,

2. initial polynomial basis size A_0, A_1, A_∞

$$\mathrm{B}(A_0, A_1, A_\infty) = \left\{ \mathbf{r} \in \mathrm{B}_\infty : \|\mathbf{r}\|_0 \le A_0, \|\mathbf{r}\|_1 \le A_1, \|\mathbf{r}\|_\infty \le A_\infty \right\},$$

$\|\mathbf{r}\|_0 = \sum_{i=1}^{d} 1_{r_i > 0}$ ➔ number of factors used in a polynomial

$\|\mathbf{r}\|_1 = \sum_{i=1}^{d} r_i$ ➔ polynomial order

$\|\mathbf{r}\|_\infty = \max_{1 \le i \le d} r_i$ ➔ maximum degree of the monomial used for any factor

Automated model selection

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

- The contribution of the **r**-*th* term in the model is proportional to the square of its coefficient: $\beta_{\mathbf{r}}^2$

  This is the consequence of using orthogonal polynomials.

$\beta_{\mathbf{r}}^2 / \sigma^2(f)$ ➔ represents a part of the function variance apportioned to this particular term, where:

$$\sigma^2(f) = \int_{\mathrm{I}^d} f^2(\mathbf{x})\, d\mathbf{x} - \left(\int_{\mathrm{I}^d} f(\mathbf{x})\, d\mathbf{x} \right)^2$$

- A model structure is determined through the shrinkage process in which we remove all insignificant terms $\beta_{\mathbf{r}}^2 << \sigma^2(f)$ from the basis $\mathrm{B}(A_0, A_1, A_\infty)$

Automated model selection

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

- The variance of a component $f_u(\mathbf{x}_u)$ in the functional ANOVA can be written in terms of the coefficients $\beta_{\mathbf{r}}$

$$\sigma^2_{\mathrm{B}_u}(f_u) = \sum_{\mathbf{r} \in \mathrm{B}_u} \beta^2_{\mathbf{r}}$$

- Relative importance of various components can be measured using the ratio

$$S_u = 100 \times \sigma^2_{\mathrm{B}_u}(f_u) \,/\, \sigma^2(f)$$

- The following indices can be calculated for a factor:

  a single factor sensitivity,

  sensitivity to interactions with other factors and

  total sensitivity.

Numerical integration techniques suitable for coefficient estimation

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

- The critical issue in an approximation problem based on the functional ANOVA is the numerical integration of multivariate functions over the multidimensional problem domain.

- For a *d*-dimensional function with bounded variation, the integration error of **Quasi-Monte Carlo** will decrease with the number of samples *N* as

$O(\ln^d(N)/N)$ ←→ **Monte Carlo Integration** $O(N^{-1/2})$

-The method which ***exploits smoothness*** to increase the convergence rate has been proposed by Smolyak (1963), and it is extensively studied under the name of **Sparse Grid** (Gerstner and Griebel, 1998).

Numerical integration techniques suitable for coefficient estimation

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

- *Lower number of function evaluations compared to the tensor product rule*

  ➔ achieved by combining univariate quadrature rules of different accuracy levels in the tensor product, instead of having the univariate rules of the same accuracy as in the classical use of the tensor product.

- Example: **tensor product rule (second order polynomials)**

$$\left\{1, x_1, {x_1}^2\right\} \otimes \left\{1, x_2, {x_2}^2\right\} = \left\{1, x_1, x_2, x_1 x_2, {x_1}^2, {x_2}^2, {x_1}^2 x_2, x_1 {x_2}^2, {x_1}^2 {x_2}^2\right\}$$

truncated Taylor expansion $\left\{1, x_1, x_2, x_1 x_2, {x_1}^2, {x_2}^2\right\}$

bounded order for all of its terms ➔ SGI

neglected without loosing accuracy → $\left\{{x_1}^2 x_2, x_1 {x_2}^2, {x_1}^2 {x_2}^2\right\}$

Numerical integration techniques suitable for coefficient estimation

THE UNIVERSITY OF ADELAIDE
AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

Function approximation using ***Tensor Product Series***.
FIRST BIVARIATE FUNCTIONS

Related to the Singular Value Decomposition (SVD) (Karhunen-Loeve expansion) truncated after r terms (rank):

$$f(x_1, x_2) \approx \sum_{i=1}^{r} \sigma_i \; g_{1i}(x_1) g_{2i}(x_2)$$

Faster singular values decay for smoother functions → small rank.

A rank r bivariate function approximation can be computed by sampling on n x n tensor grid and computing matrix SVD.

***n*^{2} function evaluations and *O*(*n*^{3}) operations**

Numerical integration techniques suitable for coefficient estimation

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

Near-optimal rank r approximation for a given approximation accuracy ε can be computed using the ***Splitting Operator*** (F.W. Chapman 2003).

Algorithm:

1. For a splitting point (a_1, b_1) – pivot location, construct a rank 1 approximation

$$f_1(x_1, x_2) = \frac{f(x_1, b_1) f(a_1, x_2)}{f(a_1, b_1)} = d_1 \, g_{11}(x_1) g_{21}(x_2), \qquad d_1 = 1/f(a_1, b_1)$$

   which interpolates function along two lines $x_1 = a_1, \; x_2 = b_1$

2. Calculate the residual $res_1(x_1, x_2) = f(x_1, x_2) - f_1(x_1, x_2)$

3. For a splitting point (a_2, b_2) in $res_1(x_1, x_2)$, construct a rank 2 approximation

$$f_2(x_1, x_2) = f_1(x_1, x_2) + \frac{res_1(x_1, b_2) \, res_1(a_2, x_2)}{res_1(a_2, b_2)} = d_1 \, g_{11}(x_1) g_{21}(x_2) + d_2 \, g_{12}(x_1) g_{22}(x_2),$$

$$d_2 = 1/f(a_2, b_2)$$

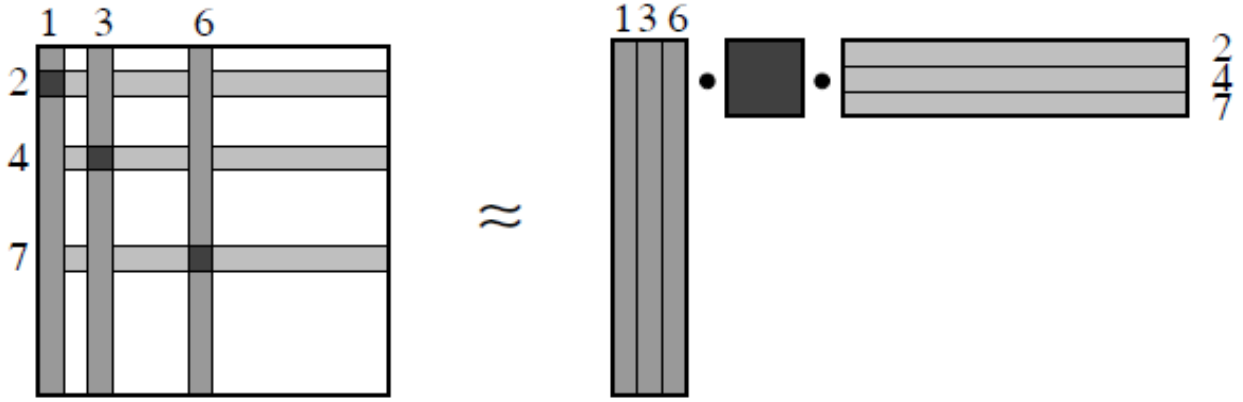
Numerical integration techniques suitable for coefficient estimation

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

The rank *r* approximation is

$$f_r(x_1, x_2) = \mathbf{C}(x_1)\,\mathbf{D}\,\mathbf{R}(x_2), \qquad \mathbf{D} = diag\,(d_1, \ldots, d_r)$$

If we sample:



e.g. *n* samples in x_1 and *n* samples in x_2, *r* times

NEEDS ADDITIONAL SAMPLING FOR COMPLETE PIVOTING (MAX VALUE)

SVD: $\mathbf{C}(x_1)\,\mathbf{D}\,\mathbf{R}(x_2) = \mathbf{Q}_C(x_1)\underbrace{\mathbf{R}_C\mathbf{D}\mathbf{R}_R^T}_{\mathbf{U}\Sigma\mathbf{V}^T}\mathbf{Q}_R(x_2)^T$

Numerical integration techniques suitable for coefficient estimation

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

Examples (exact decompositions):

$$\sin(x_1 + x_2) = \sin(x_1)\cos(x_2) + \cos(x_1)\sin(x_2) = \underbrace{\begin{bmatrix}\sin(x_1) & \cos(x_1)\end{bmatrix}}_{\mathbf{G}_1(x_1)} \underbrace{\begin{bmatrix}\cos(x_2) \\ \sin(x_2)\end{bmatrix}}_{\mathbf{G}_2(x_2)}$$

$$\sin(x_1 + \ldots + x_d) =$$

$$\underbrace{\begin{bmatrix}\sin(x_1) & \cos(x_1)\end{bmatrix}}_{\mathbf{G}_1(x_1)} \underbrace{\begin{bmatrix}\cos(x_2) & -\sin(x_2) \\ \sin(x_2) & \cos(x_2)\end{bmatrix}}_{\mathbf{G}_2(x_2)} \ldots \underbrace{\begin{bmatrix}\cos(x_{d-1}) & -\sin(x_{d-1}) \\ \sin(x_{d-1}) & \cos(x_{d-1})\end{bmatrix}}_{\mathbf{G}_{d-1}(x_{d-1})} \underbrace{\begin{bmatrix}\cos(x_d) \\ \sin(x_d)\end{bmatrix}}_{\mathbf{G}_d(x_d)}$$

Multivariate tensor decomposition:

$$f_1(x_1, \ldots, x_d) = \mathbf{G}_1(x_1)\mathbf{G}_2(x_2)\ldots\mathbf{G}_{d-1}(x_{d-1})\mathbf{G}_d(x_d)$$

Each core $\mathbf{G}_k(x_k)$ is $r_{k-1} \times r_k$ matrix; depends on the continuous coordinate x_k

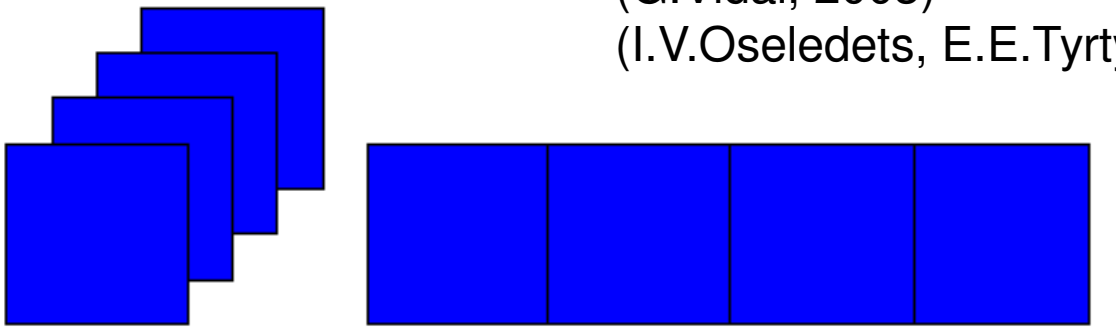
Terms: *Matrix Product State* (S.Östlund, S.Rommer, 1995)
Tensor Train (I.V.Oseledets, E.E.Tyrtyshnikov, 2009)

Numerical integration techniques suitable for coefficient estimation

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

Multivariate tensor decomposition via splitting of the unfolding matrices:

(G.Vidal, 2003)
(I.V.Oseledets, E.E.Tyrtyshnikov, 2010)



By introducing tensor product grid we get multidimensional tensor (array)

$$\mathbf{A}(i_1,\ldots,i_d) = f(x_1(i_1),\ldots,x_d(i_d))$$

Use splitting of the unfolding matrix to separate i_1 :

$$\mathbf{A}(i_1,\ldots,i_d) = \sum_{\alpha_1=1}^{r_1} \mathbf{G}_1(i_1,\alpha_1)\,\mathbf{V}(\alpha_1 i_2, i_3,\ldots,i_d),$$

separate i_2 : $\mathbf{V}(\alpha_1 i_2, i_3,\ldots,i_d) = \sum_{\alpha_2=1}^{r_2} \mathbf{G}_2(\alpha_1,i_2,\alpha_2)\,\mathbf{W}(\alpha_2 i_3, i_4,\ldots,i_d)$, etc.

Numerical integration techniques suitable for coefficient estimation

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

Bivariate function integration: replace the calculation of one integral in two dimensions by 2*r* integrals each in one dimension, where *r* is the rank of the Tensor Product Series.

$$\beta_{\mathbf{r}} = \int_0^1 \int_0^1 f(x_1, x_2)\varphi_{r_1}(x_1)\varphi_{r_2}(x_2)dx_1dx_2$$

$$\approx \sum_{i=1}^{r}\left(\int_0^1 g_{1i}(x_1)\varphi_{r_1}(x_1)dx_1\right)\left(\int_0^1 g_{2i}(x_2)\varphi_{r_2}(x_2)dx_2\right)$$

We need a univariate quadrature rule defined with n nodes and weights (x_j, w_j)

$$\int_0^1 g_{1i}(x_1)\varphi_{r_1}(x_1)dx_1 \approx \sum_{j=1}^{n} w_j g_{1i}(x_{1j})\varphi_{r_1}(x_{1j})$$

Numerical integration techniques suitable for coefficient estimation

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

Multivariate function integration via tensor decomposition

$$\beta_{\mathbf{r}} = \int_{\mathbf{I}^d} f(\mathbf{x})\, \Psi_{\mathbf{r}}(\mathbf{x})\, d\mathbf{x}, \qquad \Psi_{\mathbf{r}}(\mathbf{x}) = \prod_{i=1}^{d} \varphi_{ri}(x_i)$$

$$\beta_{\mathbf{r}} = \int_{\mathbf{I}^d} \prod_{i=1}^{d} \mathbf{G}_i(x_i)\varphi_{ri}(x_i)\, d\mathbf{x} \quad = \prod_{i=1}^{d} \int_0^1 \mathbf{G}_i(x_i)\varphi_{ri}(x_i) dx_i$$

$$\approx \prod_{i=1}^{d} \sum_{j=1}^{n} w_j \mathbf{G}_i(x_{ij})\varphi_{ri}(x_{ij})$$

we use the same n in all modes (dimensions)

Number of samples: $\mathbf{O}(ndr^2)$ where r is upper bound on the ranks of cores

Results of comparative study
QMC / SG / TT

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

Problem: predicting computer performance using a neural network model
[Venables, W. and Ripley, B. , *Modern Applied Statistics with S-PLUS*.]

$$f(\mathbf{x}) = b_o + \sum_{i=1}^{6} w_{i\to o} x_i + \sum_{h=1}^{3} w_{h\to o}\, \phi(b_h + \sum_{i=1}^{6} w_{i\to h} x_i)$$

209 computers data are used to fit the neural network model with *nnet (S-PLUS)*

$f(\mathbf{x})$	**perf**	Performance relative to IBM 370/158-3
x_1	**syct**	Cycle time in nanoseconds
x_2	**mmin**	Minimum main memory in kilobytes
x_3	**mmax**	Maximum main memory in kilobytes
x_4	**cash**	Cache size in kilobytes
x_5	**chmin**	Minimum number of channels
x_6	**chmax**	Maximum number of channels

Sensitivity analysis:

syct	mmin	mmax	cash	chmin	chmax	Importance (%)
0	0	0	0	0	1	1.25
0	0	0	0	1	0	3.65
0	0	0	1	0	0	13.01
0	0	1	0	0	0	8.615
0	0	1	1	0	0	5.401
0	1	0	0	0	0	1.147
0	1	1	0	0	0	2.575
0	1	1	1	0	0	1.047
1	0	0	0	0	0	52.13
1	0	0	0	1	0	1.069
1	0	0	1	0	0	5.47
Additive Total (individual effects)						79.819
Total (individual effects and interactions > 1%)						95.384

Results of comparative study
QMC / SG / TT

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

Sensitivity analysis by converting the neural network model into the dimension-wise expansion model (ANOVA).

Tensor product basis is composed of 1145 basis functions (multidimensional Legendre polynomials) pre-selected using limits:

$A_0 = 3$: the highest order of interaction between predictors,

$A_1 = 8$: the highest polynomial order and

$A_\infty = 4$: the highest order of the monomial used for any predictor.

Coefficients $\beta_{\mathbf{r}}$ in the model $f(\mathbf{x}) = \sum_{\mathbf{r}} \beta_{\mathbf{r}} \Psi_{\mathbf{r}}(\mathbf{x})$

are calculated by solving the integrals $\beta_{\mathbf{r}} = \int_{\mathbf{I}^d} f(\mathbf{x})\, \Psi_{\mathbf{r}}(\mathbf{x})\, d\mathbf{x}$

using: QMC (*Quasi-regression*), SG (*Sparse Grid Regression*) and *TT-regression*

Results of comparative study QMC / SG / TT

THE UNIVERSITY OF ADELAIDE AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

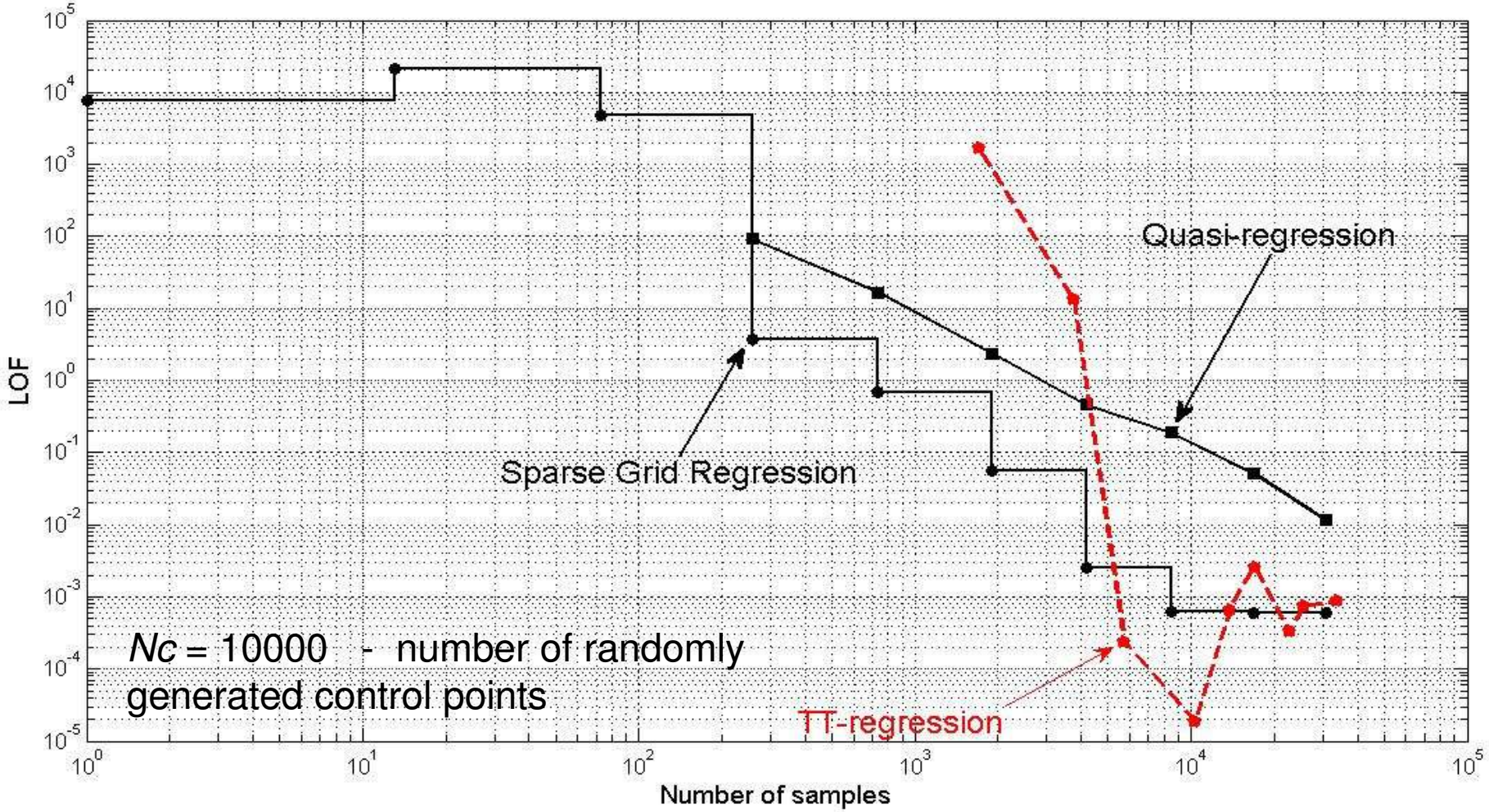
Ranks of TT cores for $\varepsilon = 0.0001$ in low-rank approximation of the unfolding matrices.

The same number of nodes n in the univariate quadratures for all modes.

n	*eval*	r_1	r_2	r_3	r_4	r_5	r_6	r_7
3	1710	1	5	10	12	8	3	1
4	3760	1	6	11	12	9	4	1
5	5750	1	7	11	12	9	4	1
6	10368	1	7	11	12	9	5	1
7	13769	1	7	11	12	9	5	1
8	16960	1	7	11	12	9	5	1
9	22680	1	7	11	11	9	5	1
10	25600	1	7	11	11	9	5	1
11	33638	1	7	11	11	9	5	1

Results of comparative study
QMC / SG / TT

THE UNIVERSITY OF ADELAIDE
AUSTRALIA
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

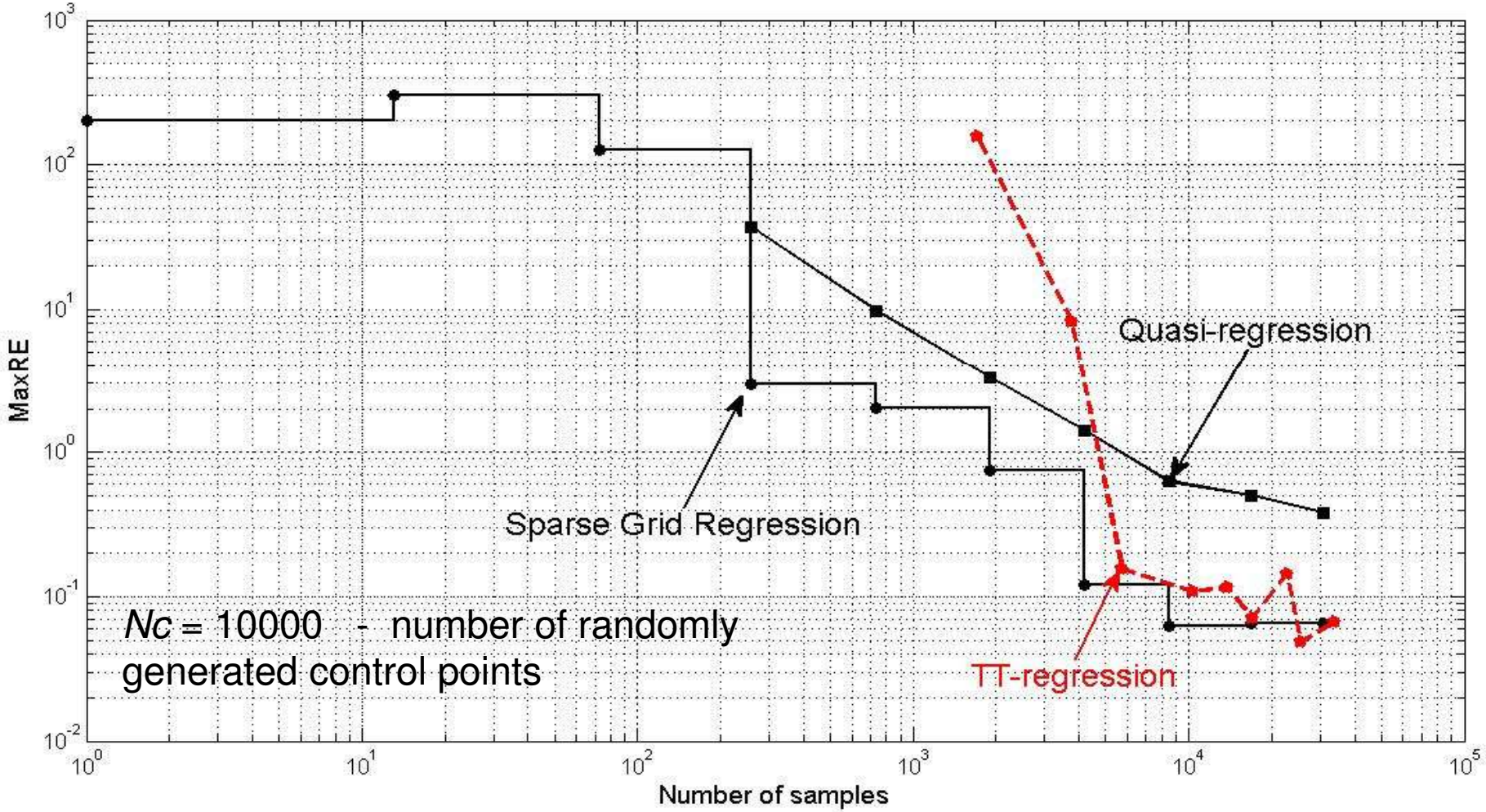


$$\mathrm{LOF} = (1/N_c)\sum_{n=1}^{N_c}(f_n - \hat{f}_n)^2 / \sigma^2(f)$$

Results of comparative study
QMC / SG / TT

THE UNIVERSITY OF ADELAIDE
AUSTRALIA

SCHOOL OF ELECTRICAL AND
ELECTRONIC ENGINEERING



$$\mathrm{MaxRe} = \max_n \left| (f_n - \hat{f}_n)/f_n \right|$$