Bespokely handcrafted for MASCOT-NUM

April 30th 2021

Polynomial least squares and their ridges

Pranay Seshadri
Imperial College London

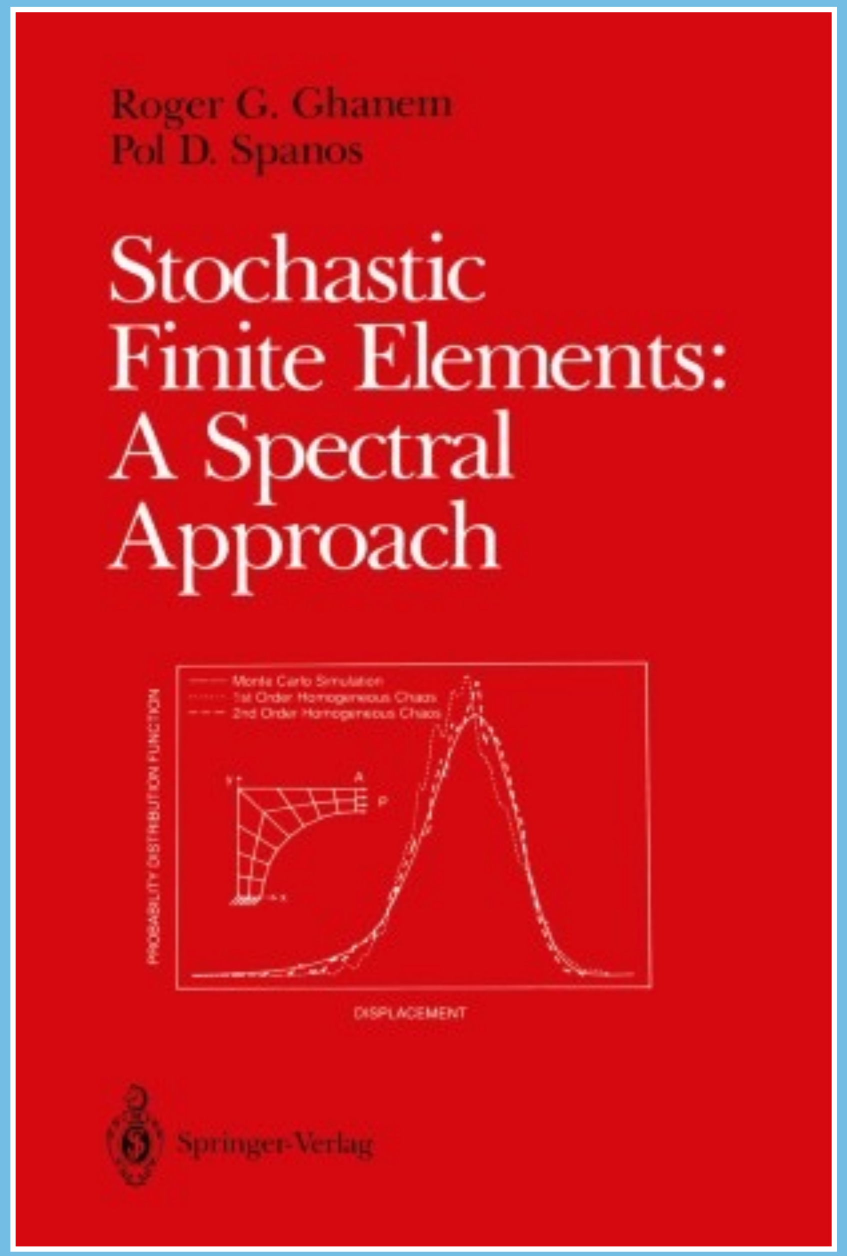
Prelude

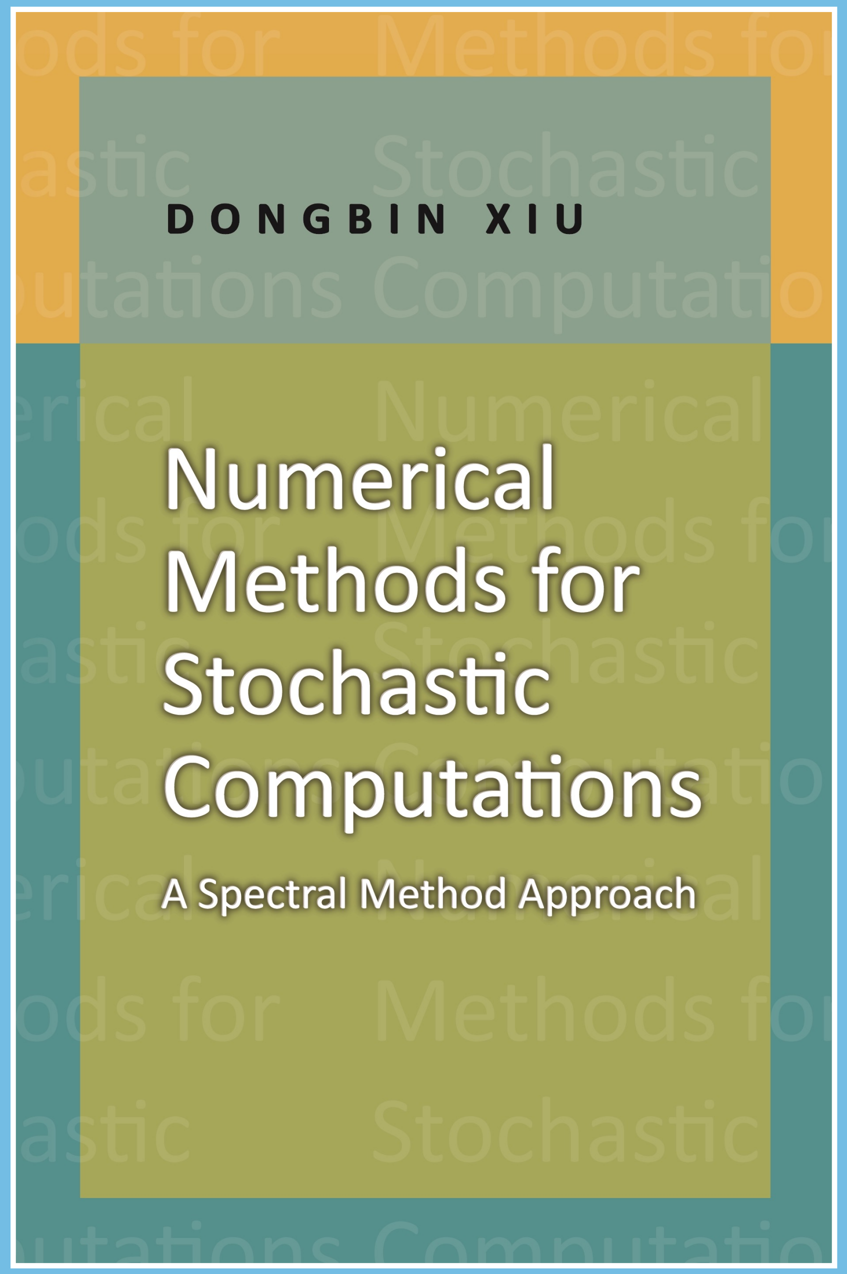
Polynomial least squares

Polynomial ridge approximations

Prelude

How this all started.

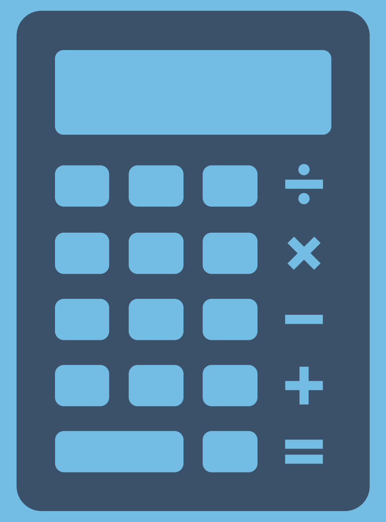




In the 2000s there was a focused interest in the idea of quantifying the uncertainty in computational models, given their increasing relevance across multiple sectors, and the increasing availability of compute.

Prelude

An uncertainty in the inputs

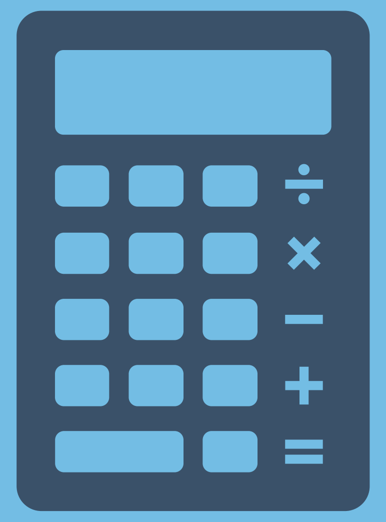
x $\longrightarrow$  $\longrightarrow$ $f(x)$?

Input parameter

Physics-based "complex model"

Prelude

An uncertainty in the inputs

x $\longrightarrow$  $\longrightarrow$ $f(x)$

?

Input parameter

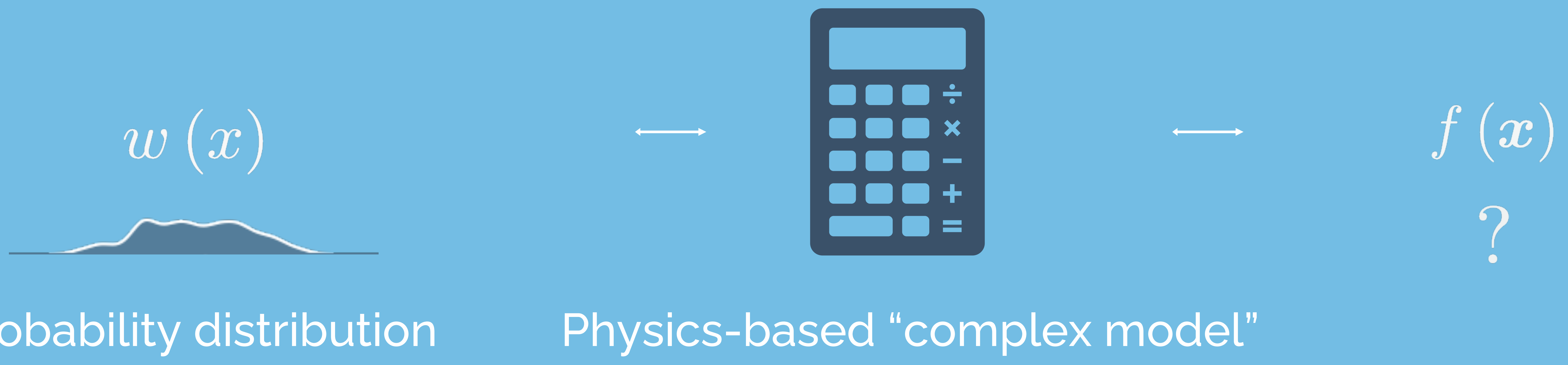
Physics-based “complex model”

BOUNDARY CONDITIONS,
EMPIRICAL VALUES,
COEFFICIENTS,
GEOMETRY PARAMETERS

PERFORMANCE,
EFFICIENCY,
MAXIMUM STRESS,
LIFT-TO-DRAG RATIOS,
PRESSURE LOSS

Prelude

An uncertainty in the inputs



MATHEMATICAL SETUP

FOR DOMAIN $D \subset \mathbb{R}^d$, LET THERE BE A WEIGHT FUNCTION $w : D \to [0, \infty)$.

MATHEMATICAL SETUP

FOR DOMAIN $D \subset \mathbb{R}^d$, LET THERE BE A WEIGHT FUNCTION $w : D \to [0, \infty)$.

CONSIDER THE SPACE

$$L_w^2 = L_w^2(D) = \left\{ u : D \to \mathbb{R} \;\middle|\; \int_D u^2(x)\, w(x)\, dx < \infty \right\}$$

MATHEMATICAL SETUP

FOR DOMAIN $D \subset \mathbb{R}^d$, LET THERE BE A WEIGHT FUNCTION $w : D \to [0, \infty)$.

CONSIDER THE SPACE

$$L^2_w = L^2_w(D) = \left\{ u : D \to \mathbb{R} \;\middle|\; \int_D u^2(x)\, w(x)\, dx < \infty \right\}$$

L^2_w IS THE HILBERT SPACE WITH INNER PRODUCT NORM

$$\langle u, v \rangle := \int_D u(x)\, v(x)\, w(x)\, dx, \qquad \|u\|^2 := \langle u, u \rangle$$

MATHEMATICAL SETUP

FOR DOMAIN $D \subset \mathbb{R}^d$, LET THERE BE A WEIGHT FUNCTION $w : D \to [0, \infty)$.

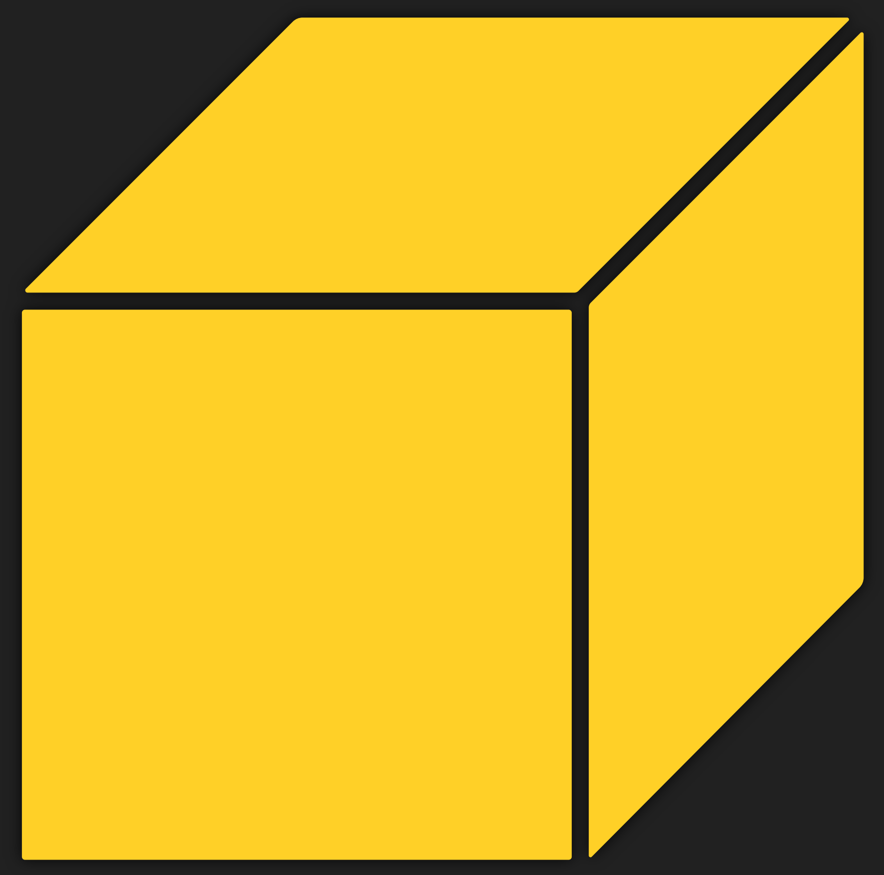
CONSIDER THE SPACE

$$L^2_w = L^2_w(D) = \left\{ u : D \to \mathbb{R} \mid \int_D u^2(x)\, w(x)\, dx < \infty \right\}$$

L^2_w IS THE HILBERT SPACE WITH INNER PRODUCT NORM

$$\langle u, v \rangle := \int_D u(x)\, v(x)\, w(x)\, dx, \quad \|u\|^2 := \langle u, u \rangle$$

AS AN EXAMPLE CONSIDER A PARAMETER SPACE $d \geq 1$ GIVEN BY $D = [-1, 1]^d$ WITH A UNIFORM WEIGHT FUNCTION $w(x) = 2^{-d}$.



MATHEMATICAL SETUP

INTERESTED IN CONSTRUCTING APPROXIMATIONS OF $f \in L^2_w$.

MATHEMATICAL SETUP

INTERESTED IN CONSTRUCTING APPROXIMATIONS OF $f \in L^2_w$.

FOR COMPUTATIONAL FEASIBILITY, APPROXIMATIONS MUST ARISE FROM A FINITE-DIMENSIONAL SUBSPACE OF L^2_w.

$$V$$

"FINITE-DIMENSIONAL SUBSPACE"

WITH DIMENSION N

LET $v_1, v_2, \ldots, v_N$ BE SUCH AN L^2_w-ORTHONORMAL BASIS FOR $V \longrightarrow \langle v_i, v_j \rangle = \delta_{i,j}$

MATHEMATICAL SETUP

BEST POSSIBLE APPROXIMATION OF $f \in L_w^2$ IS THE ORTHOGONAL PROJECTION ONTO V

$$f_N(x) := \sum_{i=1}^{N} \langle f, v_i \rangle \, v_i(x)$$

COEFFICIENTS BASIS TERMS

MATHEMATICAL SETUP

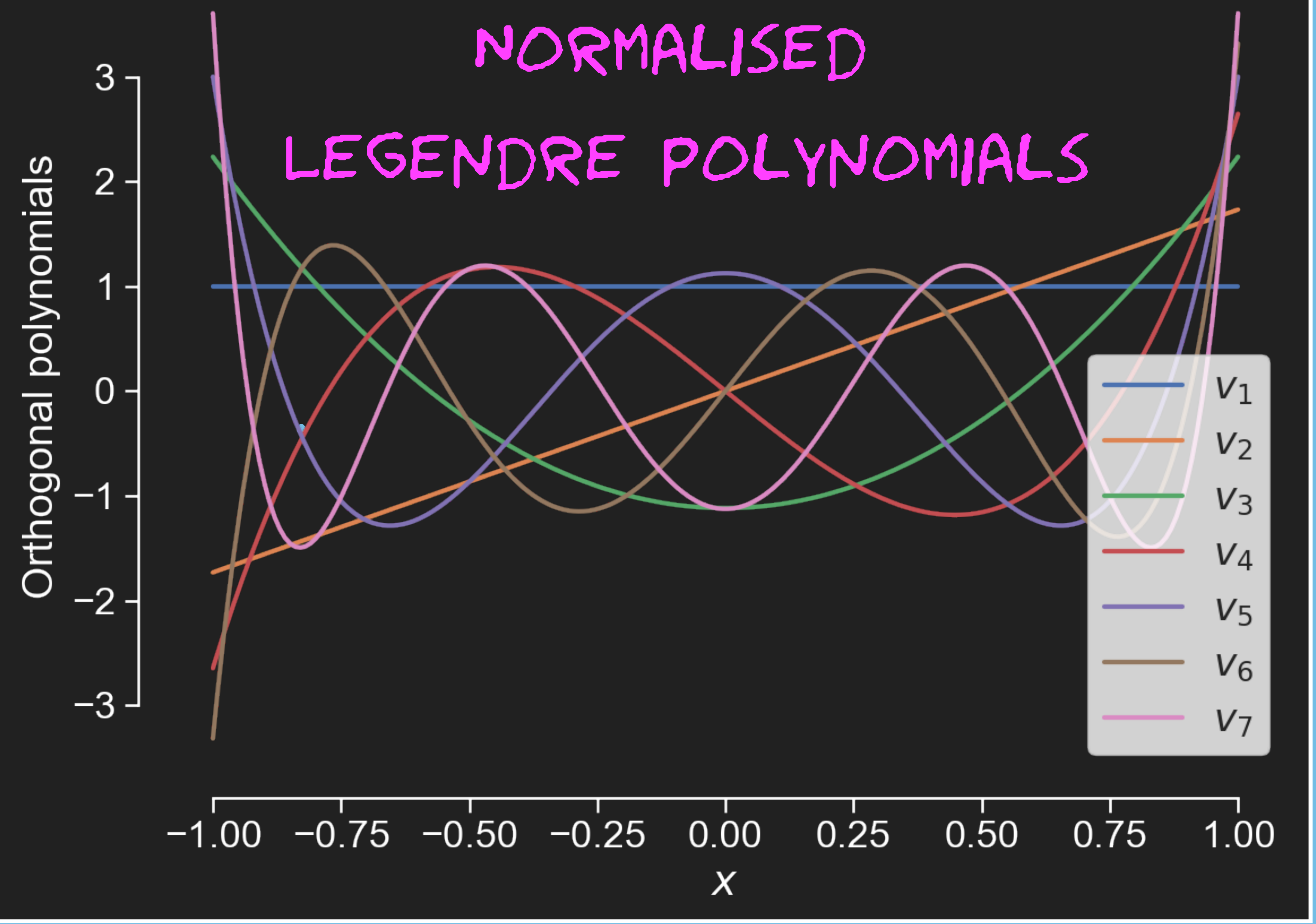
BEST POSSIBLE APPROXIMATION OF $f \in L^2_w$ IS THE ORTHOGONAL PROJECTION ONTO V

$$f_N(x) := \sum_{i=1}^{N} \underbrace{\langle f, v_i \rangle}_{\text{coefficients}} \underbrace{v_i(x)}_{\text{basis terms}}$$

COEFFICIENTS BASIS TERMS

AS AN EXAMPLE CONSIDER A PARAMETER SPACE $d = 1$ GIVEN BY $D = [-1, 1]$ WITH A UNIFORM WEIGHT FUNCTION $w(x) = 1/2$. WE CAN TAKE THE BASIS TERMS TO BE ⟶

NORMALISED LEGENDRE POLYNOMIALS



MATHEMATICAL SETUP

BEST POSSIBLE APPROXIMATION OF $f \in L_w^2$ IS THE ORTHOGONAL PROJECTION ONTO V

$$f_N(x) := \sum_{i=1}^{N} \underbrace{\langle f, v_i \rangle}_{\text{COEFFICIENTS}} \underbrace{v_i(x)}_{\text{BASIS TERMS}}$$

COEFFICIENTS OF THE APPROXIMATION NEED TO BE DETERMINED

MATHEMATICAL SETUP

BEST POSSIBLE APPROXIMATION OF $f \in L^2_w$ IS THE ORTHOGONAL PROJECTION ONTO V

$$f_N(x) := \sum_{i=1}^{N} \underbrace{\langle f, v_i \rangle}_{\text{coefficients}} \underbrace{v_i(x)}_{\text{basis terms}}$$

COEFFICIENTS BASIS TERMS

COEFFICIENTS OF THE APPROXIMATION NEED TO BE DETERMINED

$$\langle f, v_i \rangle = \int_D f(x)\, v_i(x)\, w(x)\, dx$$

$$\approx \sum_{j=1}^{M} f(x_j)\, v_i(x_j)\, \lambda_j \quad = c_i$$

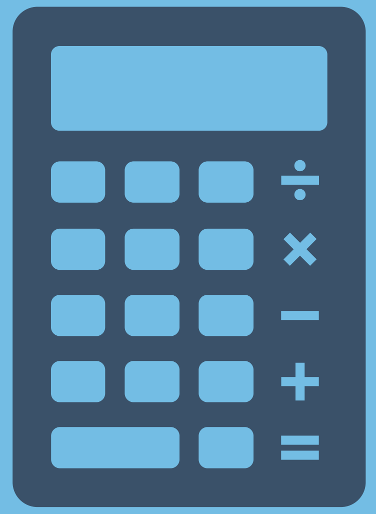
REQUIRES A QUADRATURE RULE OF THE FORM $(x_j, \lambda_j)_{j=1}^{M}$. IN 1D WE KNOW THAT GAUSS QUADRATURE POINTS ARE OPTIMAL.

Prelude

An uncertainty in the inputs

$w\left(x\right)$

Probability distribution



Physics-based “complex model”

$f\left(\boldsymbol{x}\right)$

?

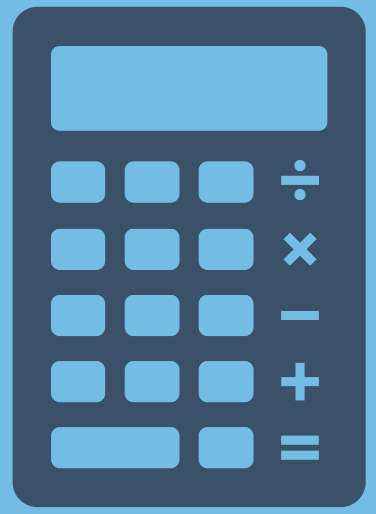
Generate points for model evaluation

Evaluate model at the points and fit a polynomial

Prelude

An uncertainty in the inputs

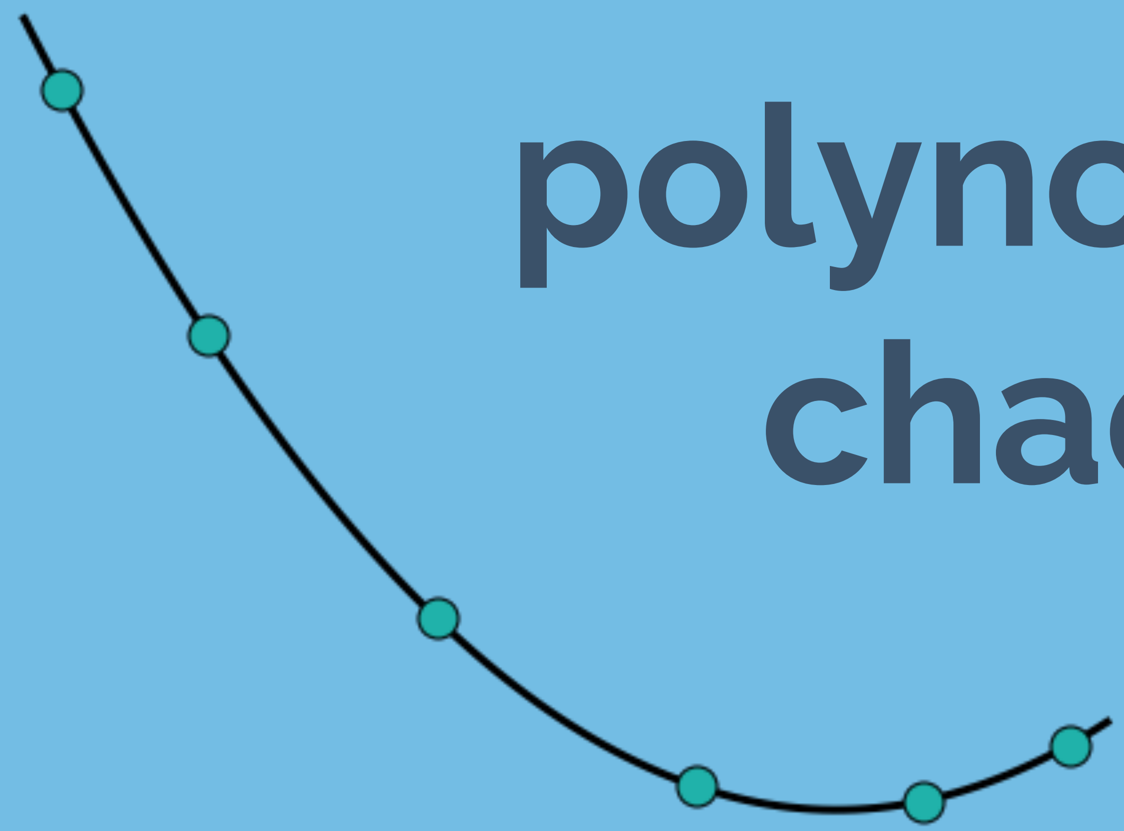
$w\left(x\right)$ $\longrightarrow$ $\longrightarrow$ $f\left(\boldsymbol{x}\right)$



?

Probability distribution

Physics-based "complex model"



polynomial chaos

Generate points for model evaluation

Evaluate model at the points and fit a polynomial

MULTIVARIATE EXTENSION

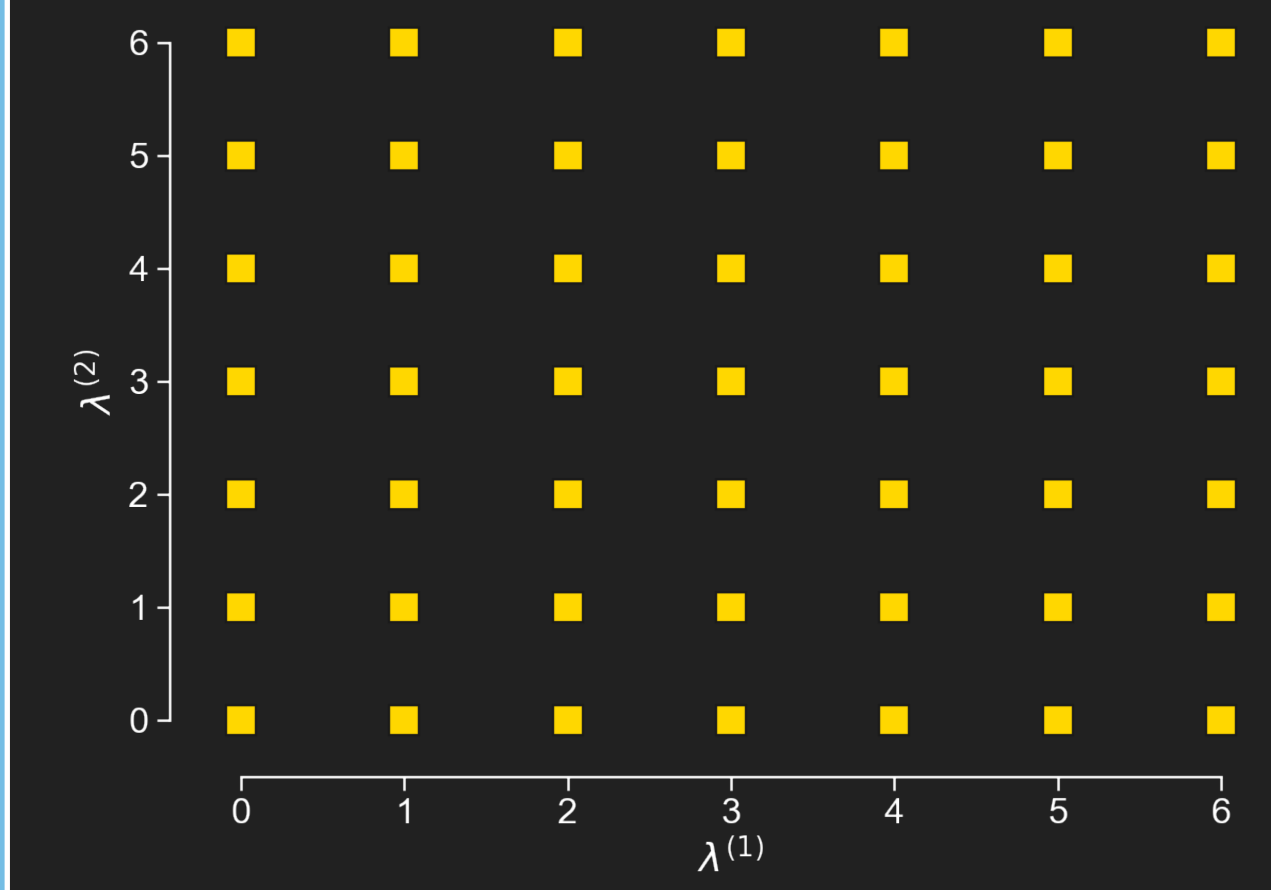
LET $\lambda \in \mathbb{N}_0^d = \{0, 1, \ldots\}^d$ DENOTE A MULTI-INDEX. WE THEN DEFINE

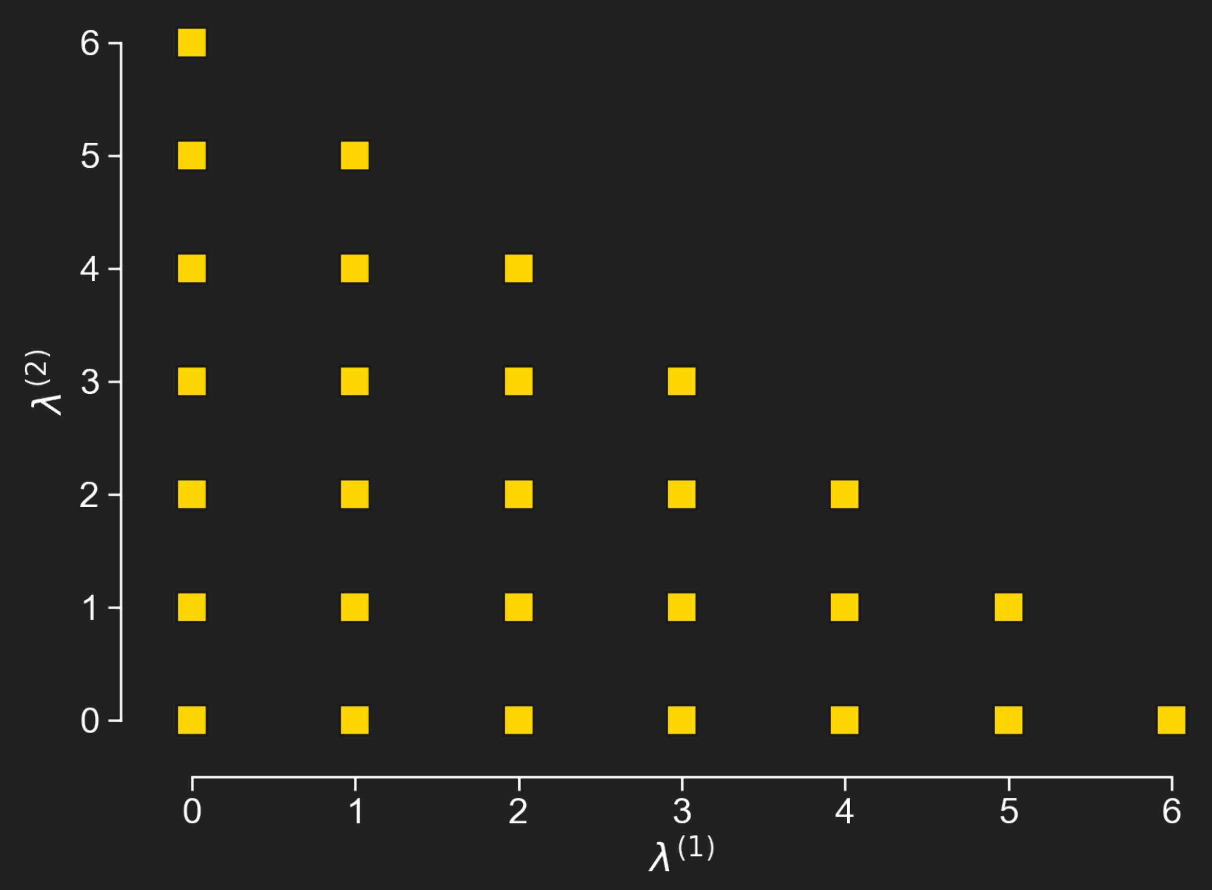
$$x = \left(x^{(1)}, x^{(2)}, \ldots, x^{(d)}\right), \quad \lambda = \left(\lambda^{(1)}, \lambda^{(2)}, \ldots, \lambda^{(d)}\right), \quad x^{\lambda} := \prod_{j=1}^{d} \left[x^{(j)}\right]^{\lambda^{(j)}}$$

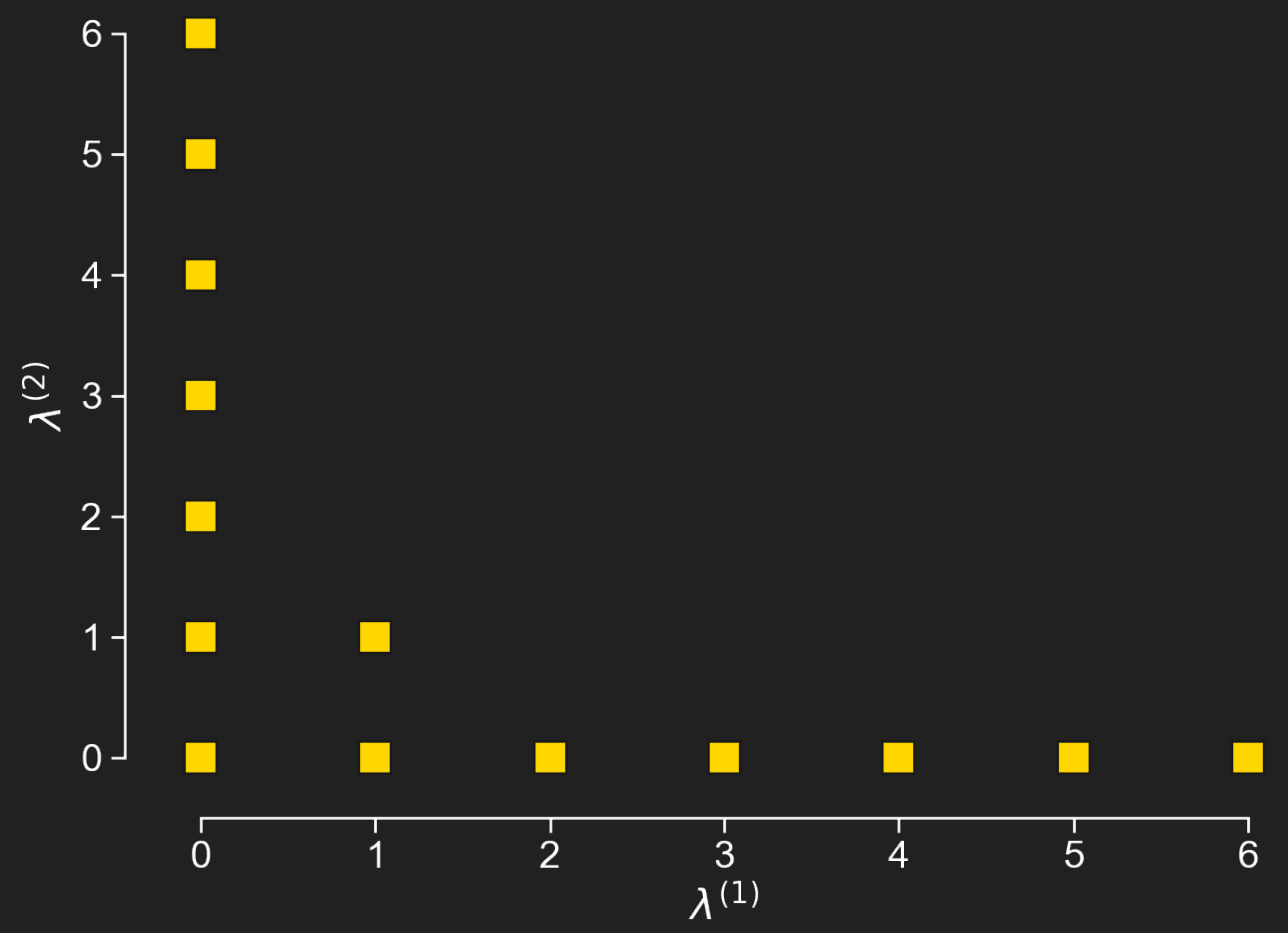
INTUITIVE WAY TO PRESENT POLYNOMIAL SPACES IN MANY DIMENSIONS IS TO

1. IDENTIFY A SET OF MULTI-INDICES;
2. DEFINE A POLYNOMIAL SPACE AS SPAN OF MONOMIALS

$$\Lambda = \{\lambda_1, \ldots, \lambda_N\} \subset \mathbb{N}_0^d \qquad \& \qquad V(\Lambda) := \operatorname{span}\left\{x^{\lambda} \mid \lambda \in \Lambda\right\}$$

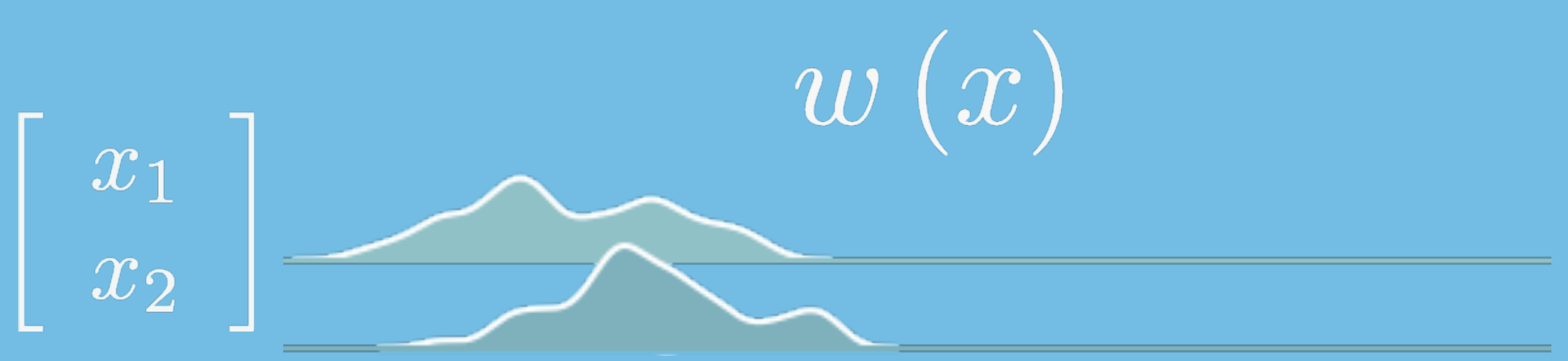




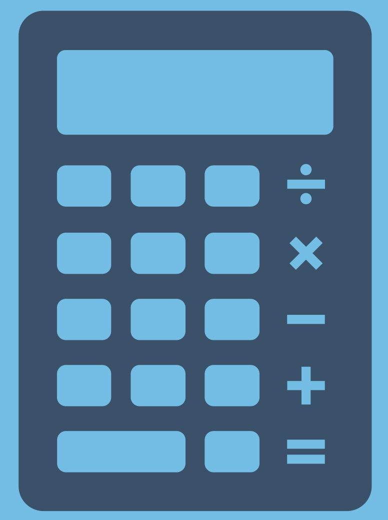


Prelude

Polynomial chaos extends across dimension (2D)



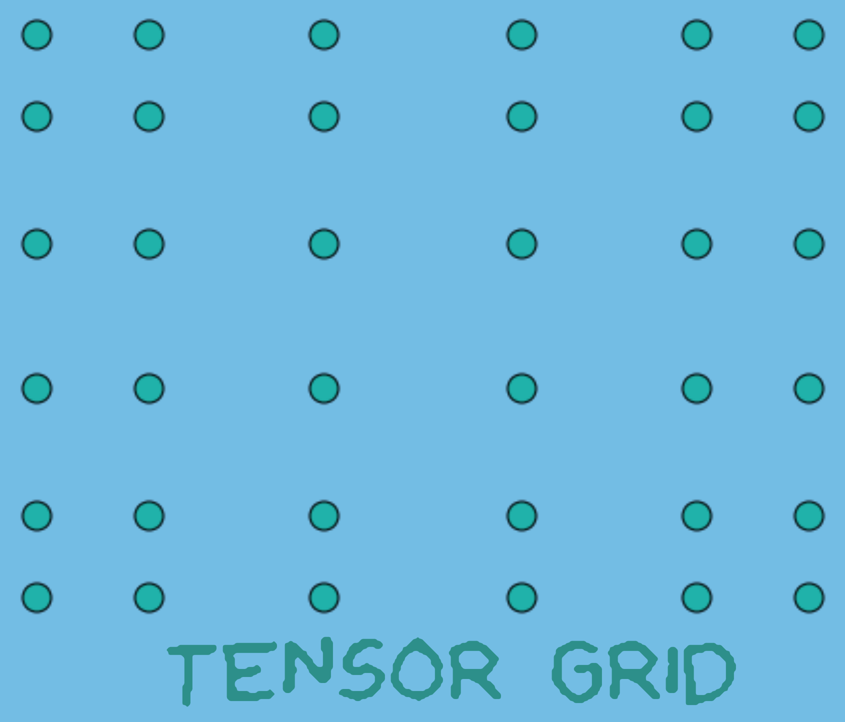
Probability distribution



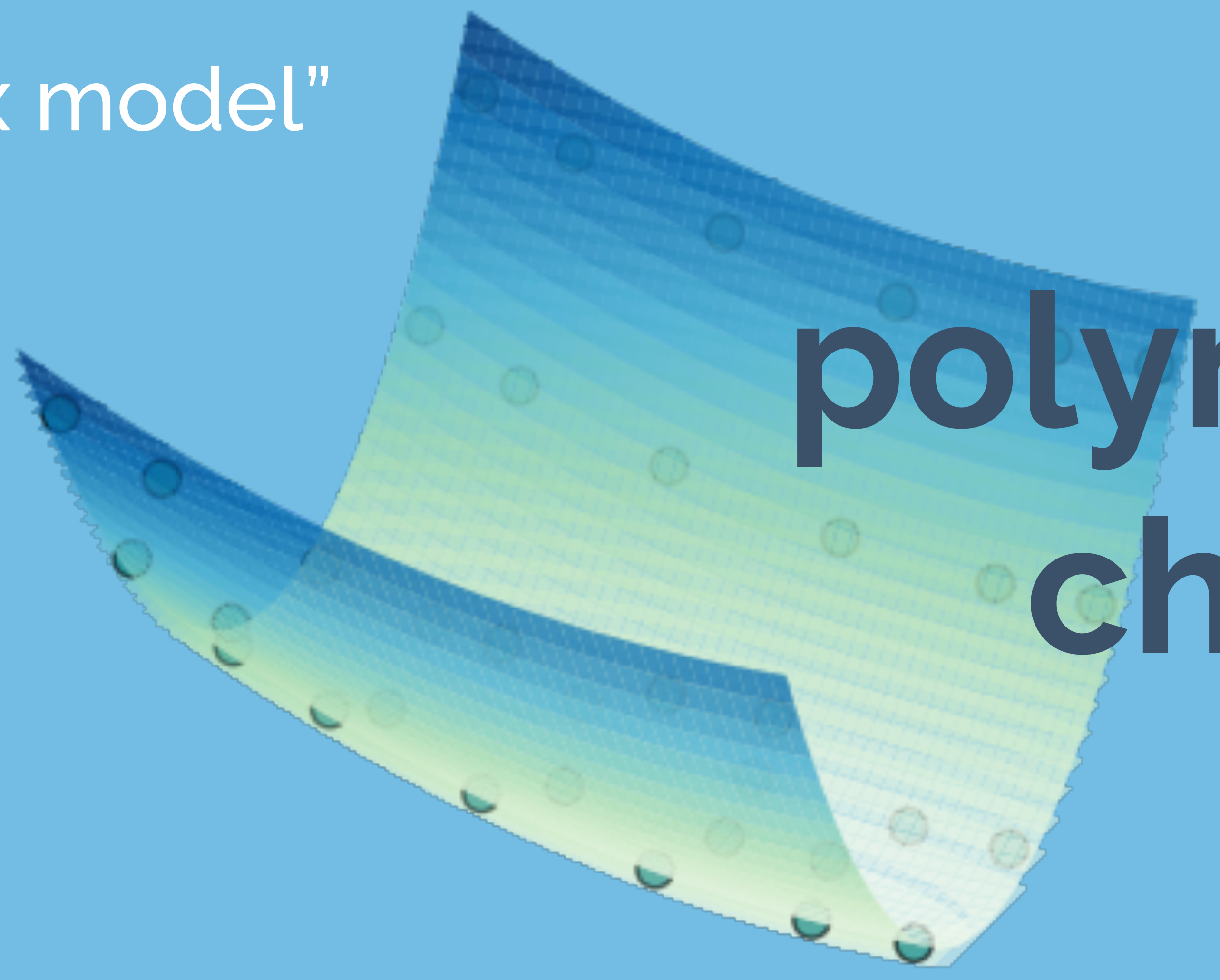
Physics-based “complex model”

$f(\boldsymbol{x})$

?



Generate points for model evaluation



polynomial chaos

Evaluate model at the points and fit a polynomial

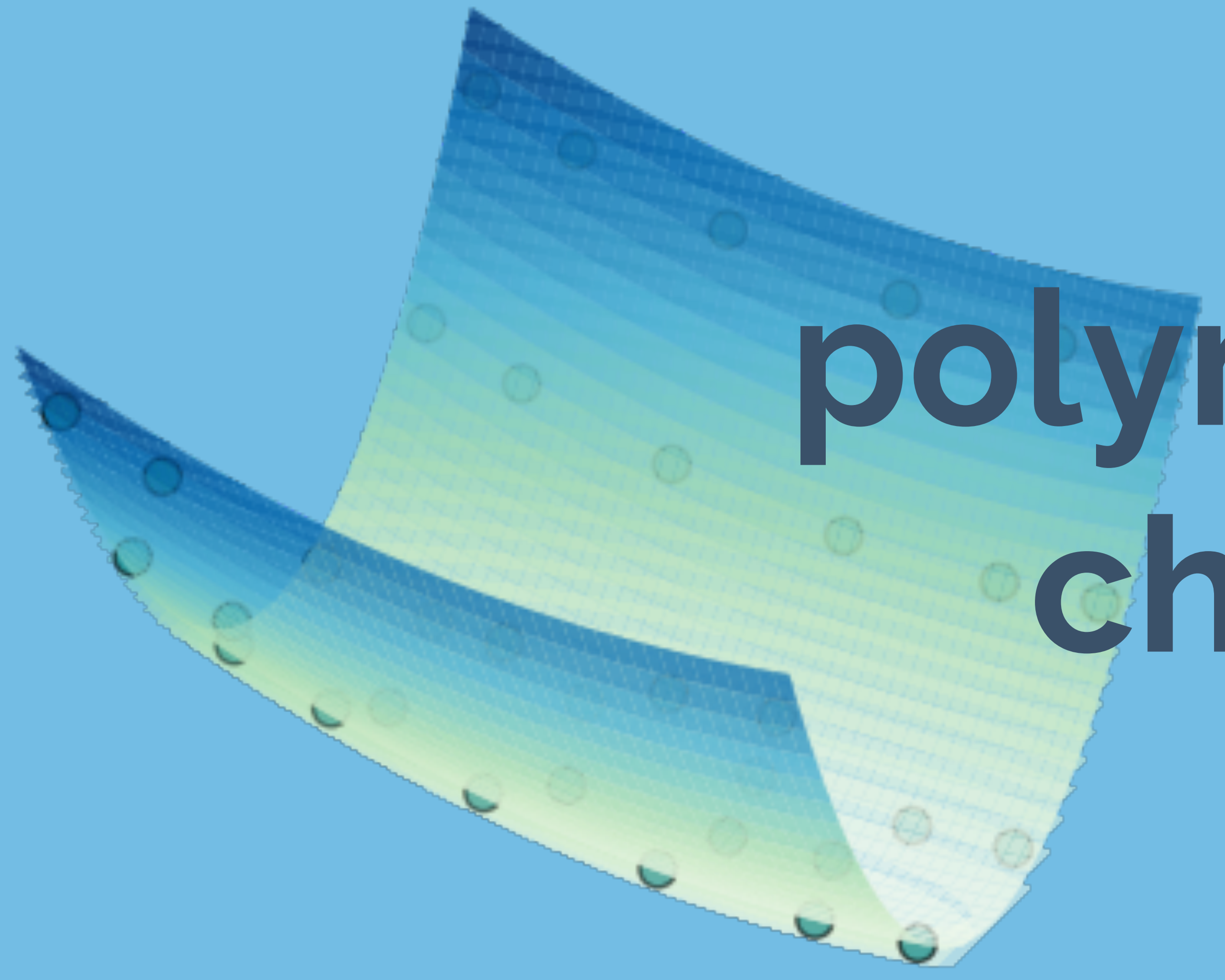
Prelude

Polynomial chaos extends across dimension (2D)

ONCE WE HAVE A POLYNOMIAL...

CAN EASILY COMPUTE:

1. MEAN, VARIANCE, SKEWNESS AND KURTOSIS.
2. PROBABILITIES OF OUTPUT.
3. SENSITIVITY INDICES (SUCH AS SOBOL').
4. GRADIENTS (USEFUL FOR OPTIMISATION).
5. CRITERION FOR DESIGN OF EXPERIMENT.



polynomial chaos

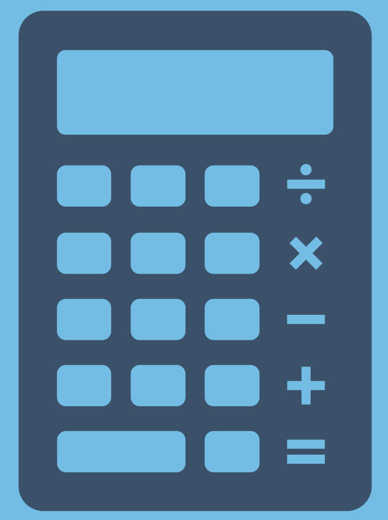
Evaluate model at the points and fit a polynomial

Prelude

Polynomial chaos extends across dimension (2D)

$w(x)$

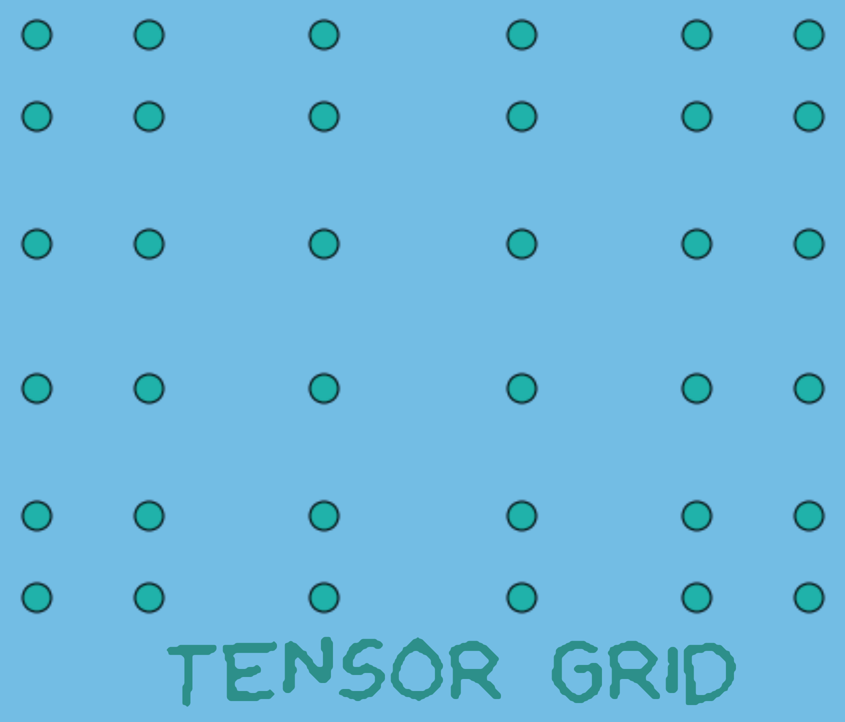
Probability distribution



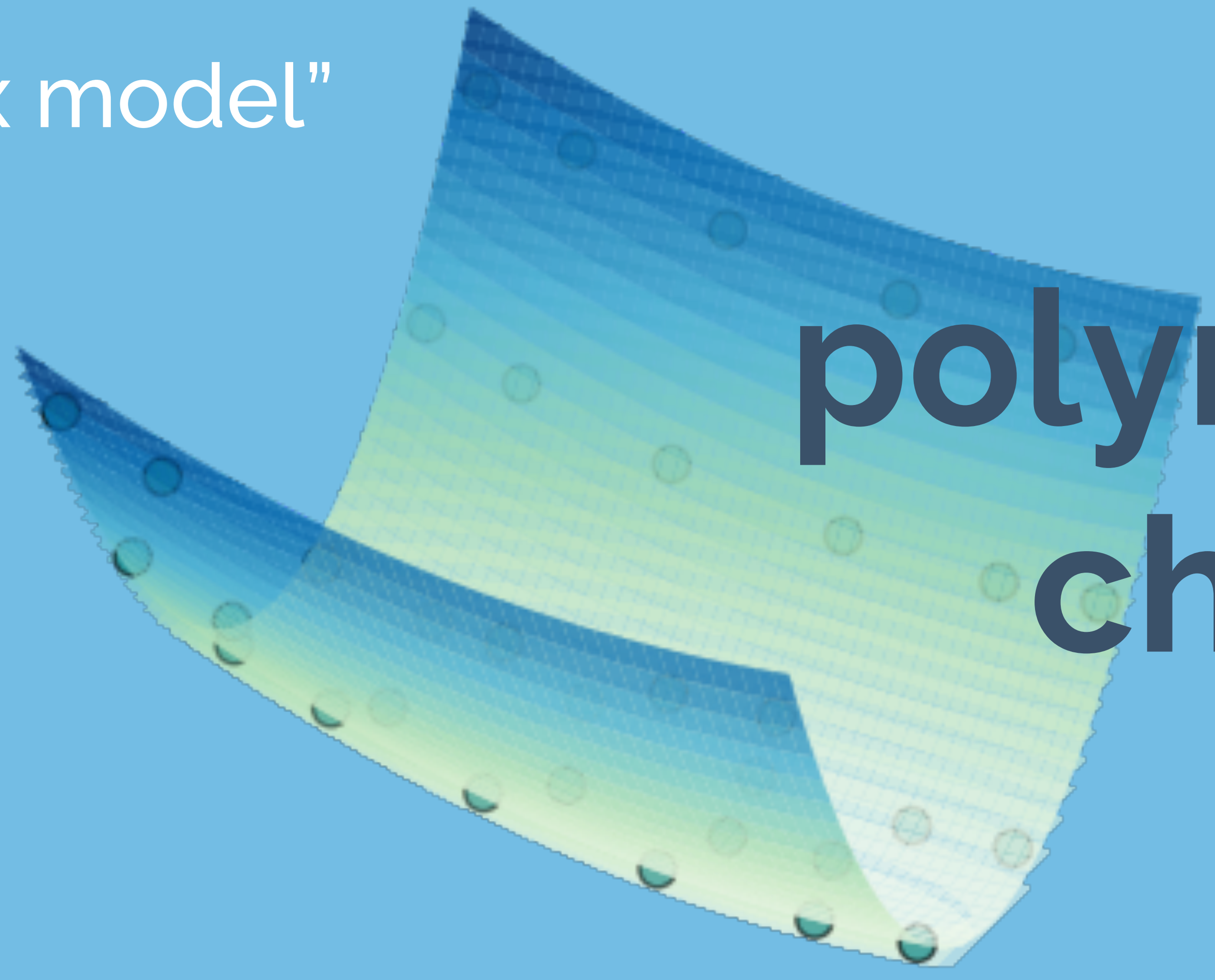
Physics-based “complex model”

$f(\boldsymbol{x})$

?



Generate points for model evaluation

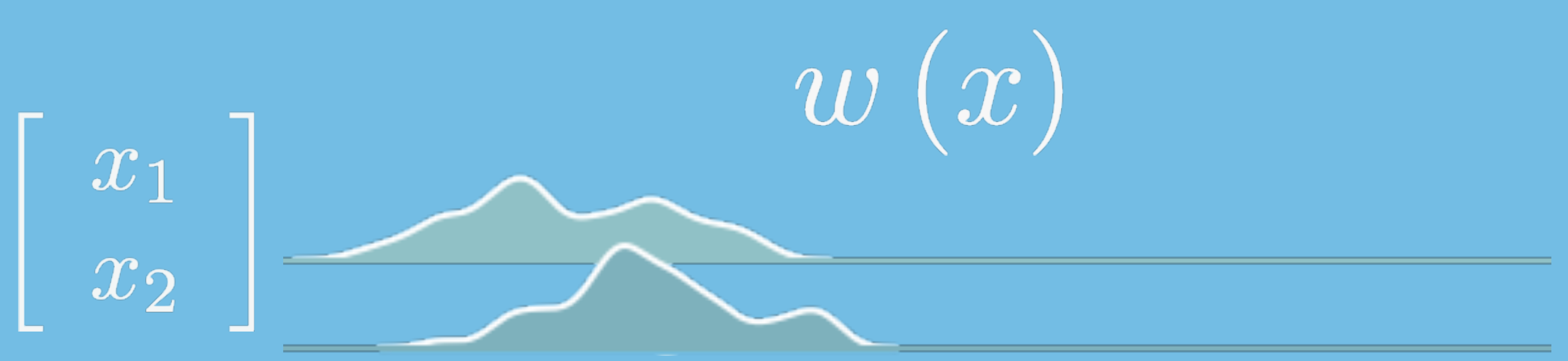


polynomial chaos

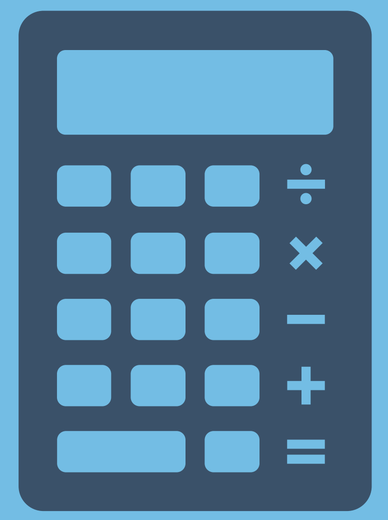
Evaluate model at the points and fit a polynomial

Prelude

Polynomial chaos extends across dimension (2D)



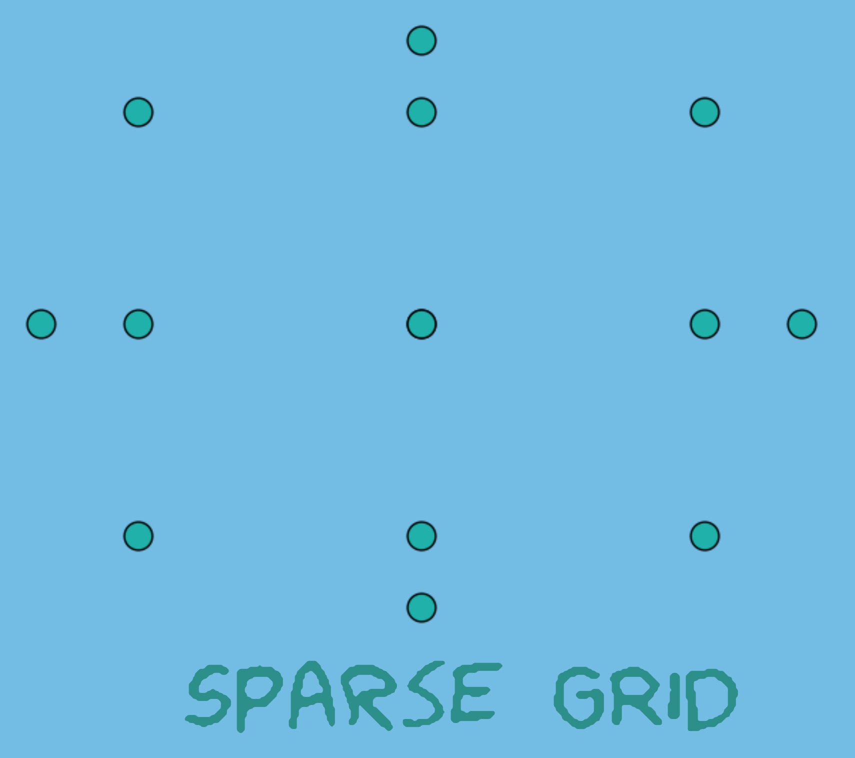
Probability distribution



Physics-based “complex model”

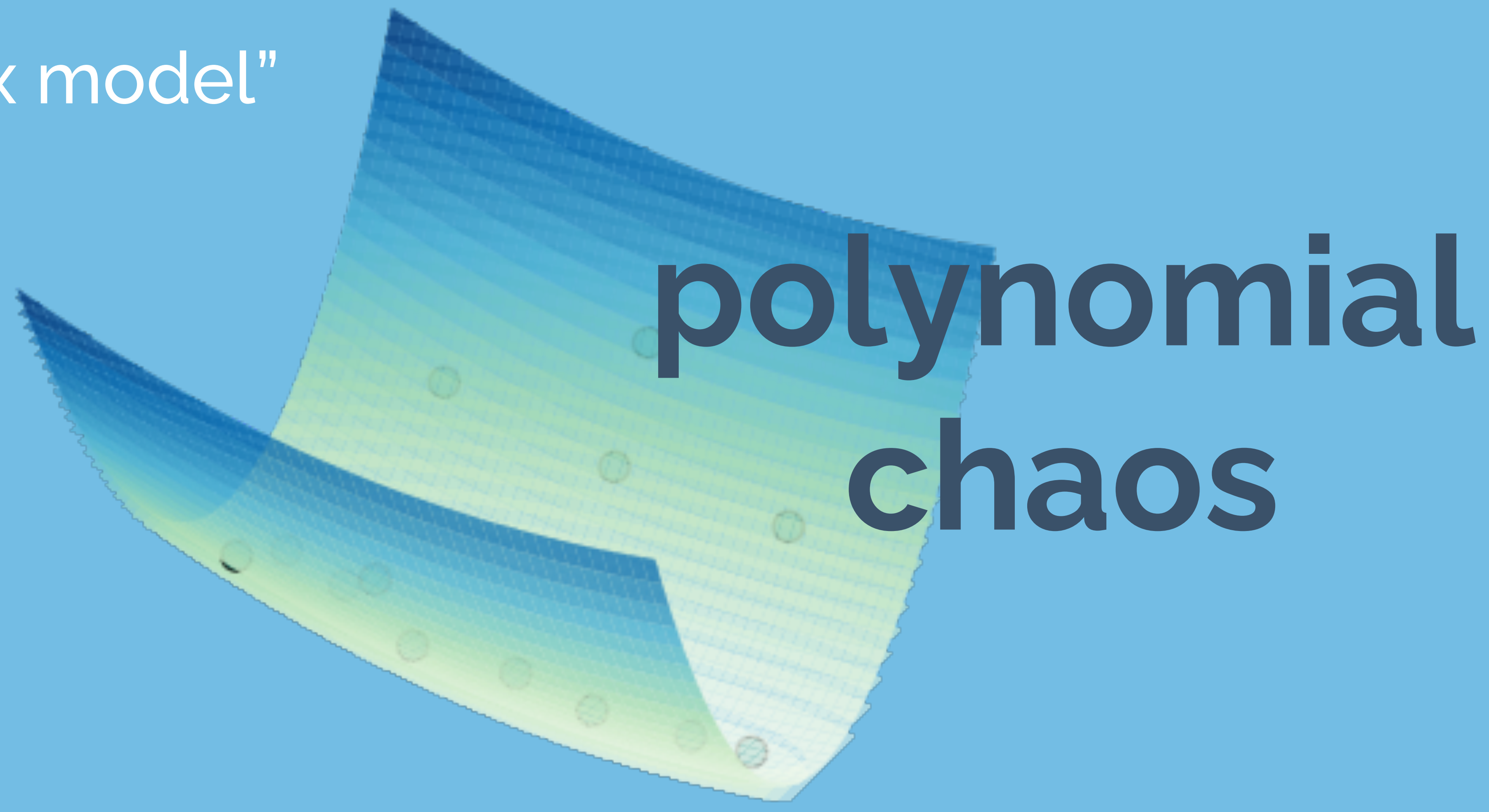
$f(x)$

?



TO REDUCE NUMBER OF MODEL EVALUATIONS

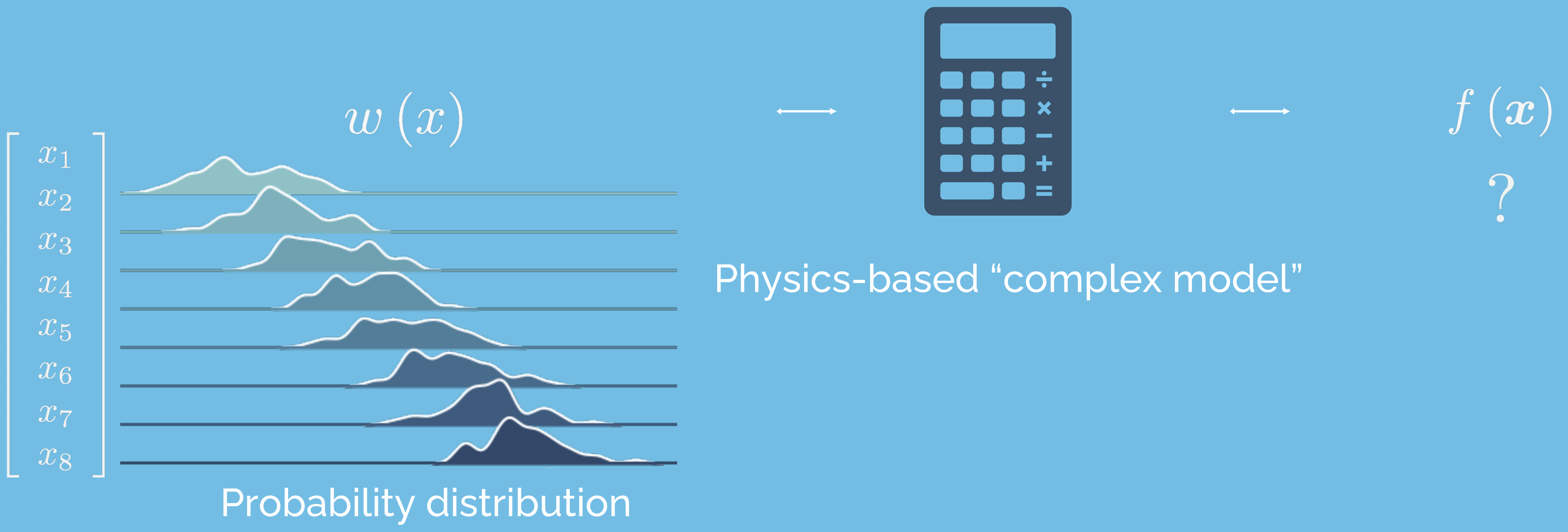
Generate points for model evaluation



Evaluate model at the points and fit a polynomial

Prelude

Polynomial chaos extends across dimension (many D)



BOUNDARY CONDITIONS,
EMPIRICAL VALUES,
COEFFICIENTS,
GEOMETRY PARAMETERS

PERFORMANCE,
EFFICIENCY,
MAXIMUM STRESS,
LIFT-TO-DRAG RATIOS,
PRESSURE LOSS

Prelude

Curse of dimensionality

Computationally prohibitive to resort to interpolation grids / quadrature rules in high dimensions.

Need a simpler approach that is not wedding to any particular multi-index set.

One approach is to frame this as a least squares problem.

Prelude

Polynomial least squares

Polynomial ridge approximations

POLYNOMIAL LEAST SQUARES

DEFINE AN $M \times N$ MATRIX $\mathbf{A}$ AND A VECTOR $\mathbf{f} \in \mathbb{R}^M$ WITH ENTRIES

$$(A)_{m,n} = \frac{1}{\sqrt{M}} v_n(x_m) \qquad \& \qquad (f)_m = \frac{1}{\sqrt{M}} f(x_m)$$

POLYNOMIAL LEAST SQUARES

DEFINE AN $M \times N$ MATRIX $\mathbf{A}$ AND A VECTOR $\mathbf{f} \in \mathbb{R}^M$ WITH ENTRIES

$$(A)_{m,n} = \frac{1}{\sqrt{M}} v_n(x_m) \qquad \& \qquad (f)_m = \frac{1}{\sqrt{M}} f(x_m)$$

WE ARE INTERESTED IN SOLVING

$$\mathbf{c} := \underset{\mathbf{d} \in V}{argmin} \|\mathbf{Ad} - \mathbf{f}\|_2^2$$

FOCUS IS ON THE OVERDETERMINED CASE WHERE $M \geq N$, WHERE WE ASSUME THAT $\mathbf{A}$ HAS FULL COLUMN RANK.

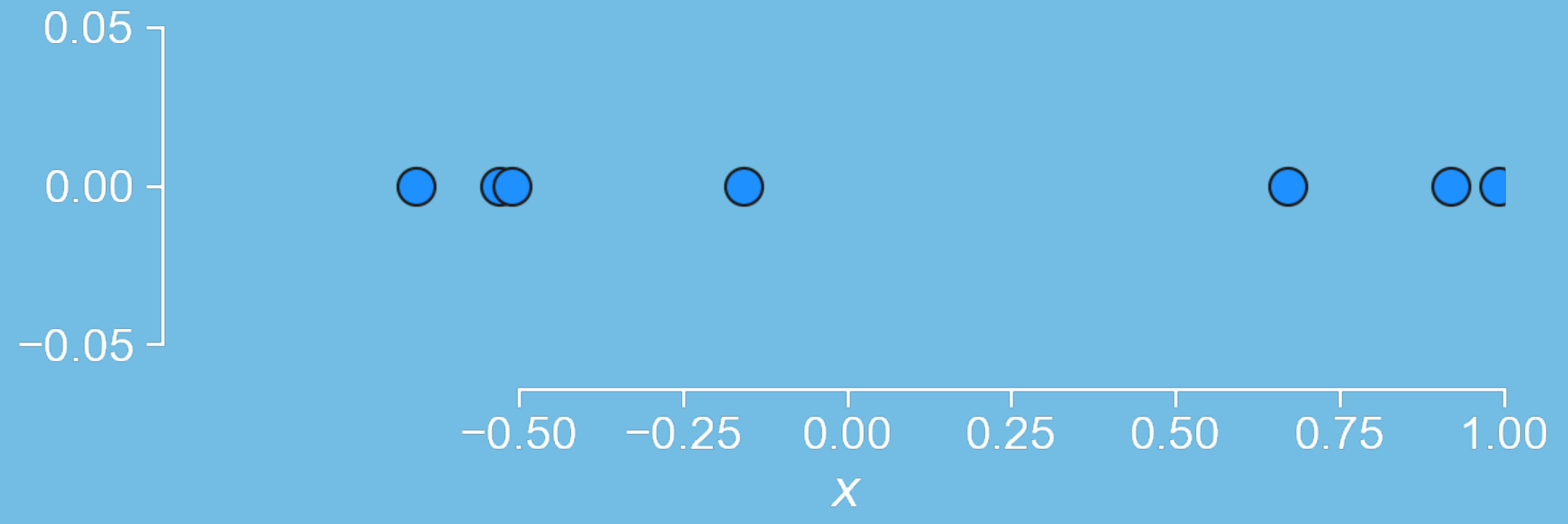
IT WILL BE USEFUL TO DEFINE $\mathbf{G} = \mathbf{A}^T \mathbf{A}$.

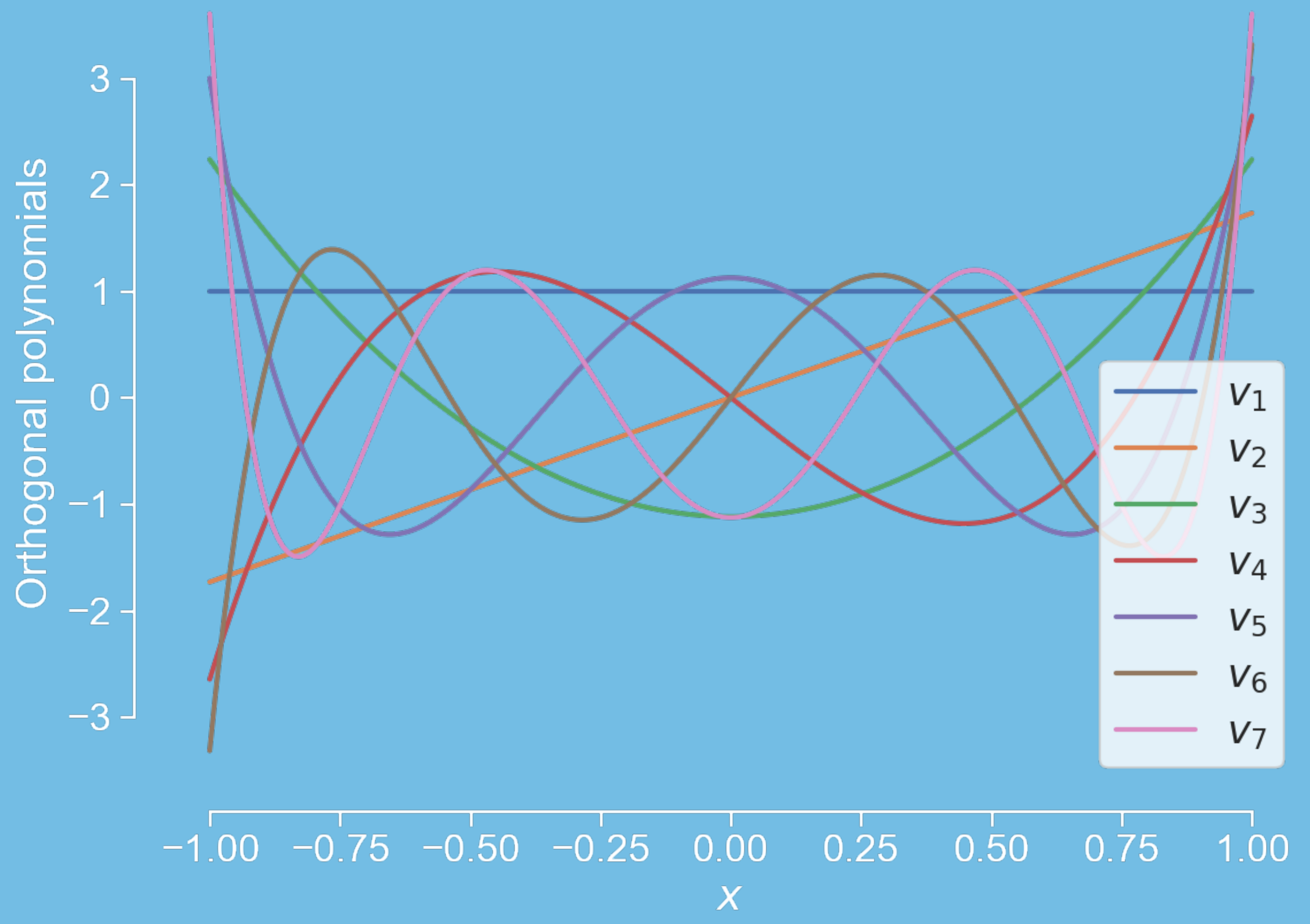
Polynomial least squares

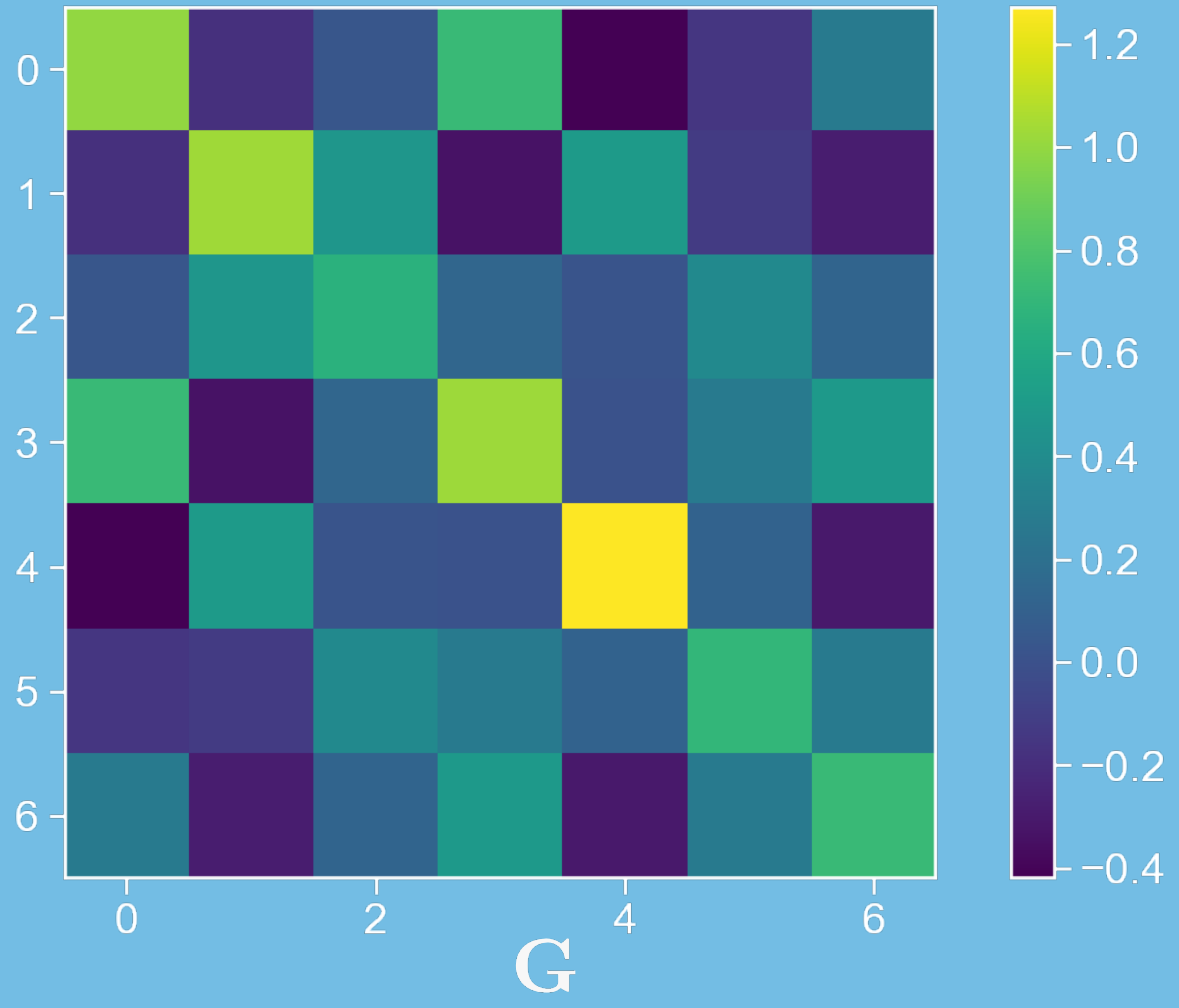
Structured vs randomised points

For randomised points and $M = N = 7$

$$(A)_{m,n} = \frac{1}{\sqrt{M}} v_n\left(x_m\right) \quad \& \quad \mathbf{G} = \mathbf{A}^T\mathbf{A}$$



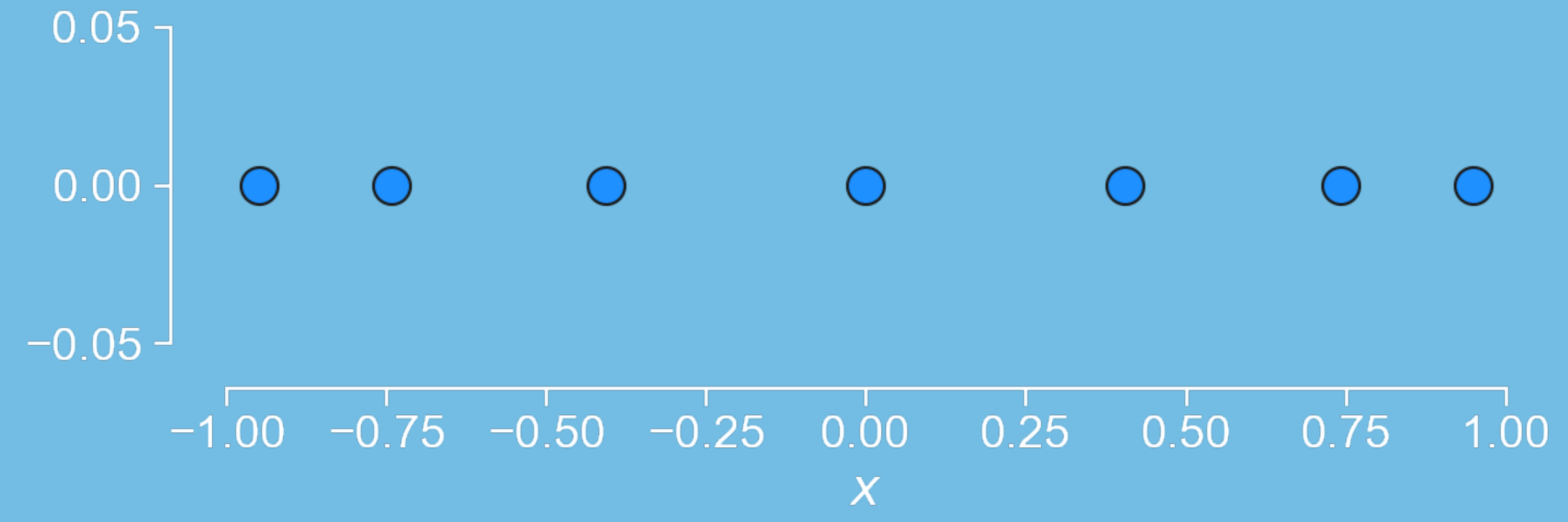


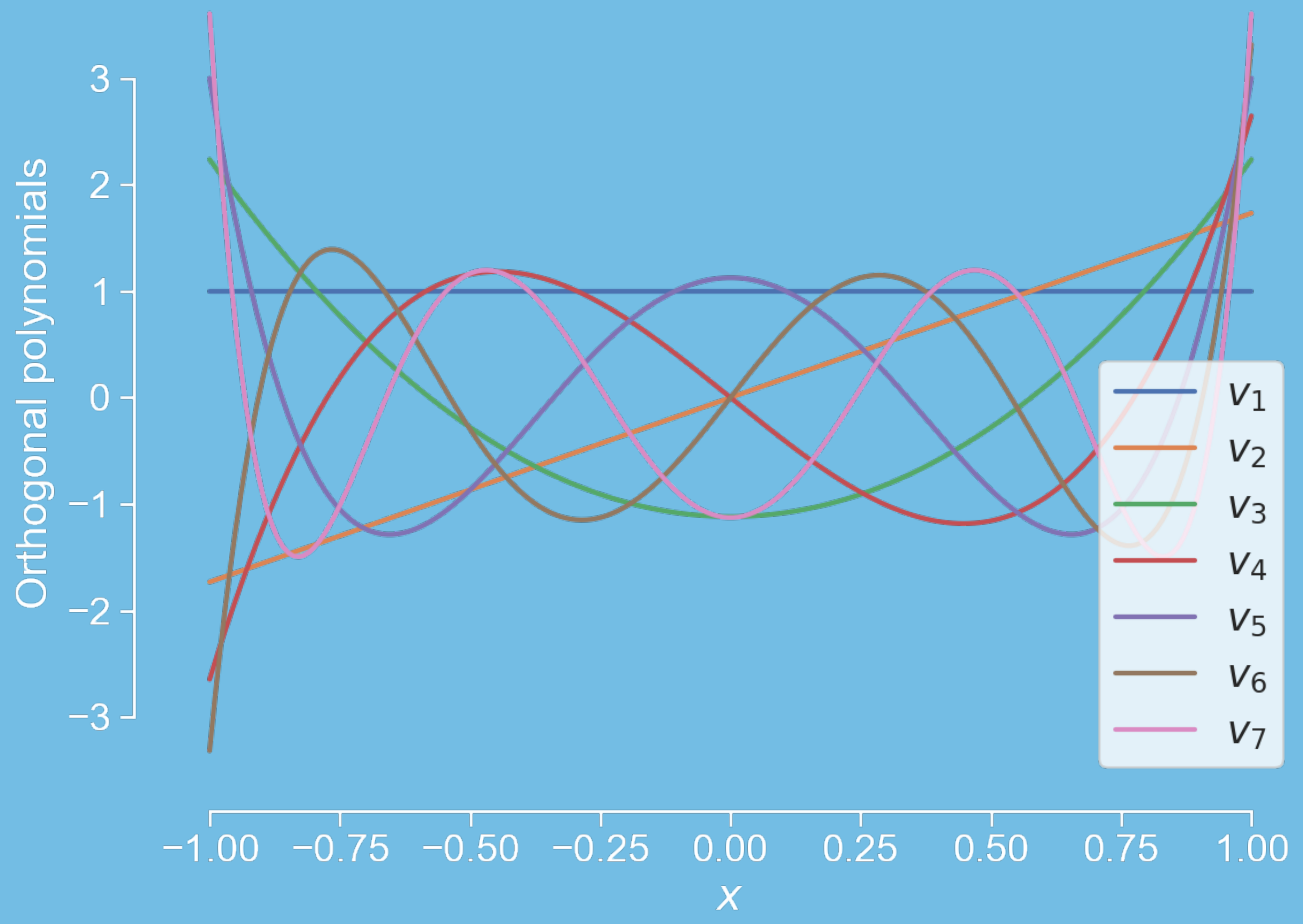


Polynomial least squares

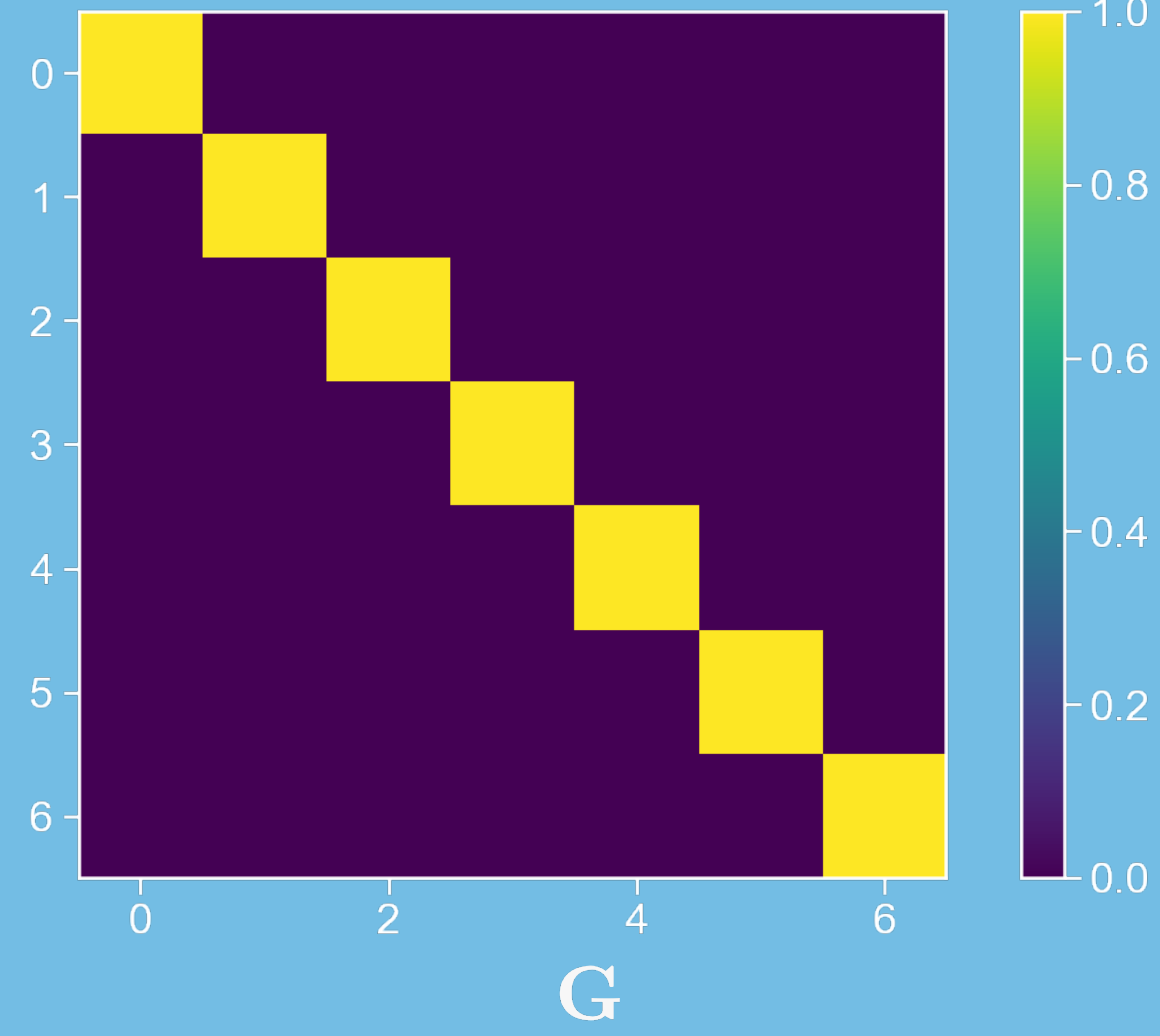
Structured vs randomised points

For Gauss-Legendre points and $M = N = 7$





$$(A)_{m,n} = \frac{1}{\sqrt{M}} v_n \left(x_m\right) \quad \& \quad \mathbf{G} = \mathbf{A}^T \mathbf{A}$$



Polynomial least squares

Structured vs randomised points

Can show that as $\mathbf{G} \to \mathbf{I}$ we will reach the best approximation. This has the consequence of yielding a well conditioned $\mathbf{A}$.

The difficulty for multivariate polynomial spaces is to identify $M \sim N$ that can satisfy this.

The key insight over the last few years has been the use of biased sampling strategies to remove the instabilities that arise when approximating.

Polynomial least squares

Structured vs randomised points

We introduce a bias sampling strategy that moves us from this

$$\left(A\right)_{m,n} = \frac{1}{\sqrt{M}} v_n\left(x_m\right) \quad \& \quad \left(f\right)_m = \frac{1}{\sqrt{M}} f\left(x_m\right)$$

to

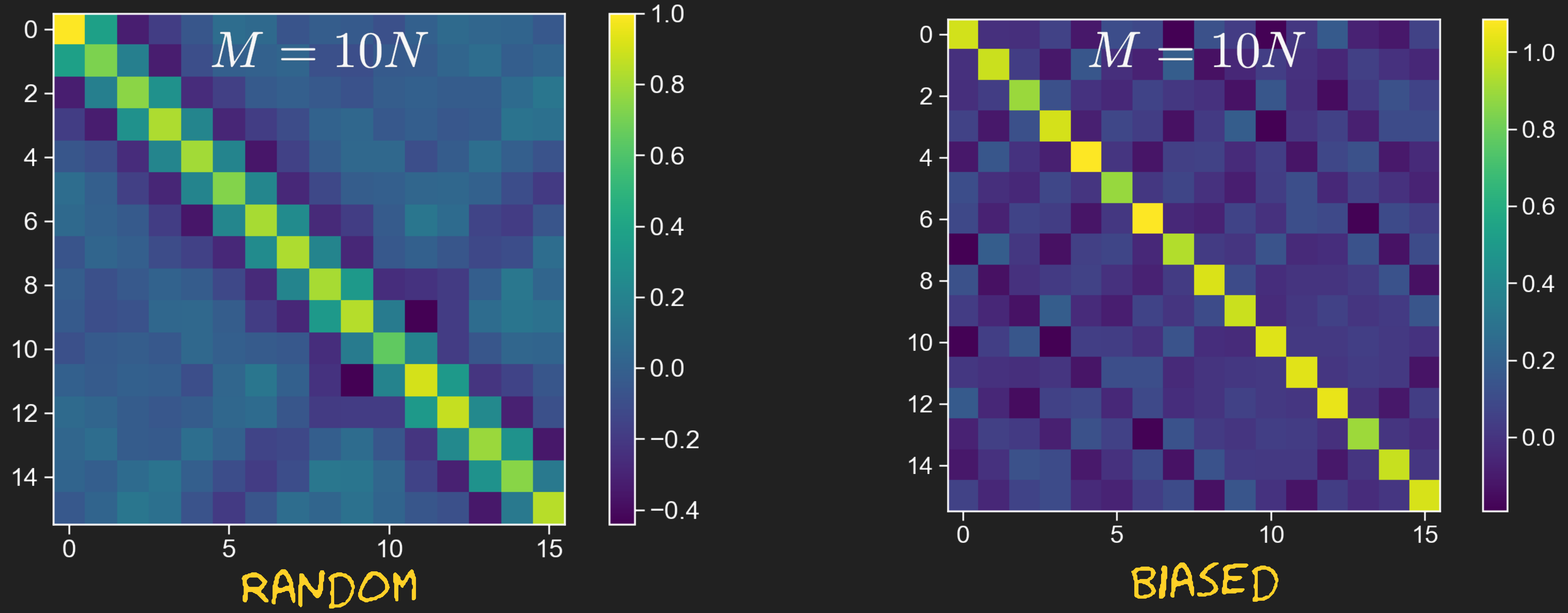
$$\left(A\right)_{m,n} = \frac{1}{\sqrt{Mq^2\left(x_m\right)}} v_n\left(x_m\right) \quad \& \quad \left(f\right)_m = \frac{1}{\sqrt{Mq^2\left(x_m\right)}} f\left(x_m\right)$$

BIASED SAMPLING

COHEN ET AL. (2013) INFORM US THAT IF WE CAN IDENTIFY THE $q(x)$ THAT CAN MINIMISE

$$\sup_{x \in D} \sum_{n=1}^{N} \left(\frac{v_n(x)}{q(x)} \right)^2$$

THEN WE HAVE A BOUND ON $M \geq N$ AND WE WILL ENSURE THAT $\mathrm{G} \rightarrow \mathrm{I}$ WITH HIGH PROBABILITY. BELOW EXAMPLE ON MULTIVARIATE POLYNOMIAL.



Prelude

Polynomial least squares

Polynomial ridge approximations

Polynomial ridge least squares

Data-driven dimension reduction

The curse of dimensionality

APPROXIMATE COMPLEX MODEL WITH A POLYNOMIAL.

$$f\left(\boldsymbol{x}\right) \approx p\left(\boldsymbol{x}\right)$$

APPROXIMATE COMPLEX MODEL WITH A POLYNOMIAL.

$$\sout{f(\boldsymbol{x}) \approx p(\boldsymbol{x})}$$

APPROXIMATE COMPLEX MODEL WITH A POLYNOMIAL RIDGE FUNCTION.

$$f(\boldsymbol{x}) \approx p\left(\mathbf{M}^T \boldsymbol{x}\right)$$

APPROXIMATE COMPLEX MODEL WITH A POLYNOMIAL.

$$\sout{f(\boldsymbol{x}) \approx p(\boldsymbol{x})}$$

APPROXIMATE COMPLEX MODEL WITH A POLYNOMIAL RIDGE FUNCTION.

$$f(\boldsymbol{x}) \approx p\left(\mathbf{M}^T \boldsymbol{x}\right)$$

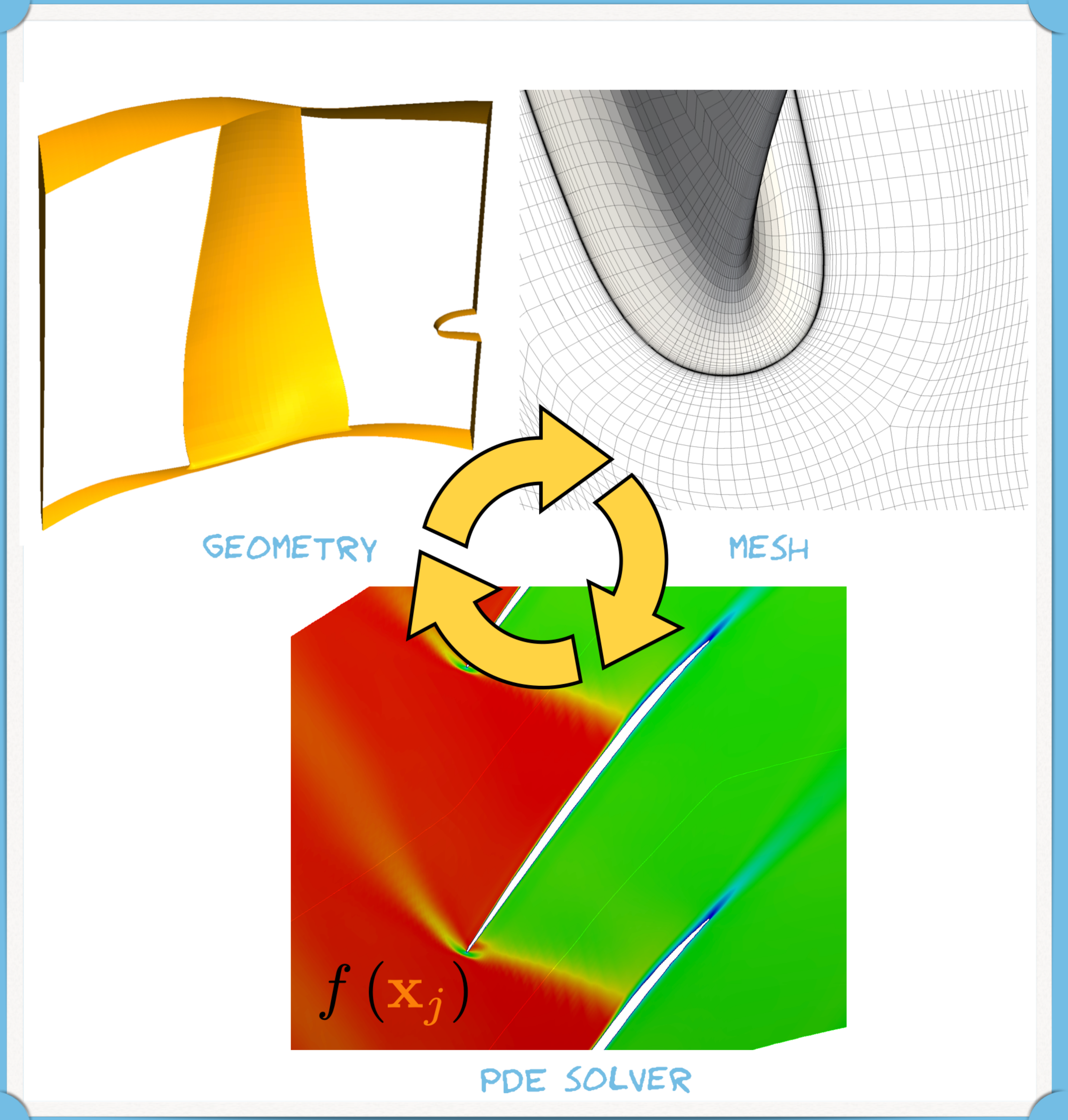
$$\mathbf{M} \in \mathbb{R}^{d \times k} \qquad \boldsymbol{x} \in \mathbb{R}^d$$

$$k << d$$

THIS YIELDS A POLYNOMIAL DEFINED OVER A SUBSPACE IN K DIMENSIONS. CHEAPER TO APPROXIMATE!

Polynomial ridge least squares

Data-driven dimension reduction



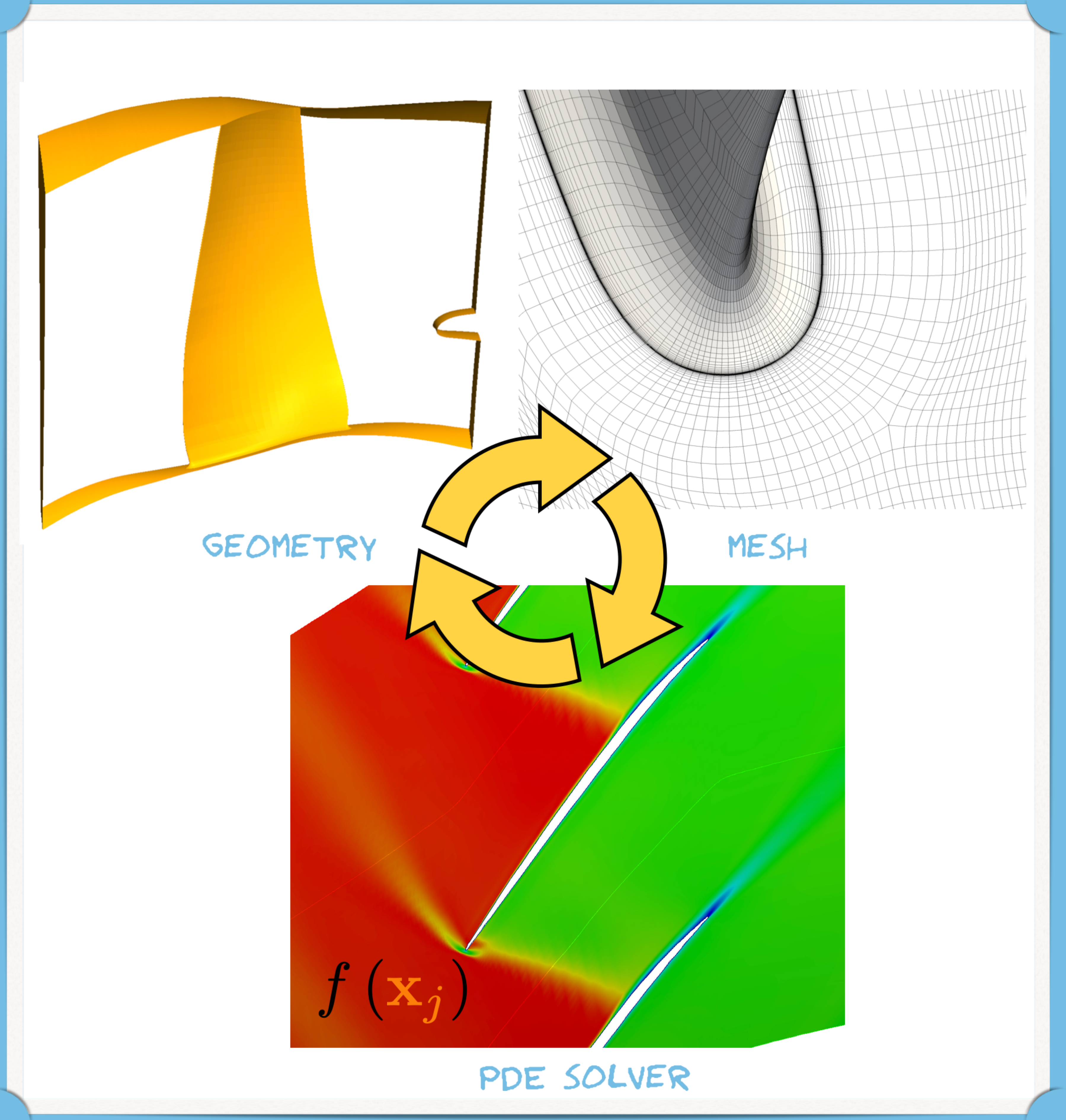
Iterate through the geometry-mesh-solver loop to generate a random design of experiment database.

DESIGNS	x_1	x_2	x_3	...	x_{25}	EFFICIENCY
1	-0.32	0.52	0.81		-0.19	93.1
2	0.55	0.37	-0.49		-0.33	92.8
⋮		⋮			⋮	⋮
350	0.74	0.61	0.31		0.16	91.4

$x \in \mathbb{R}^{25}$

Polynomial ridge least squares

Data-driven dimension reduction



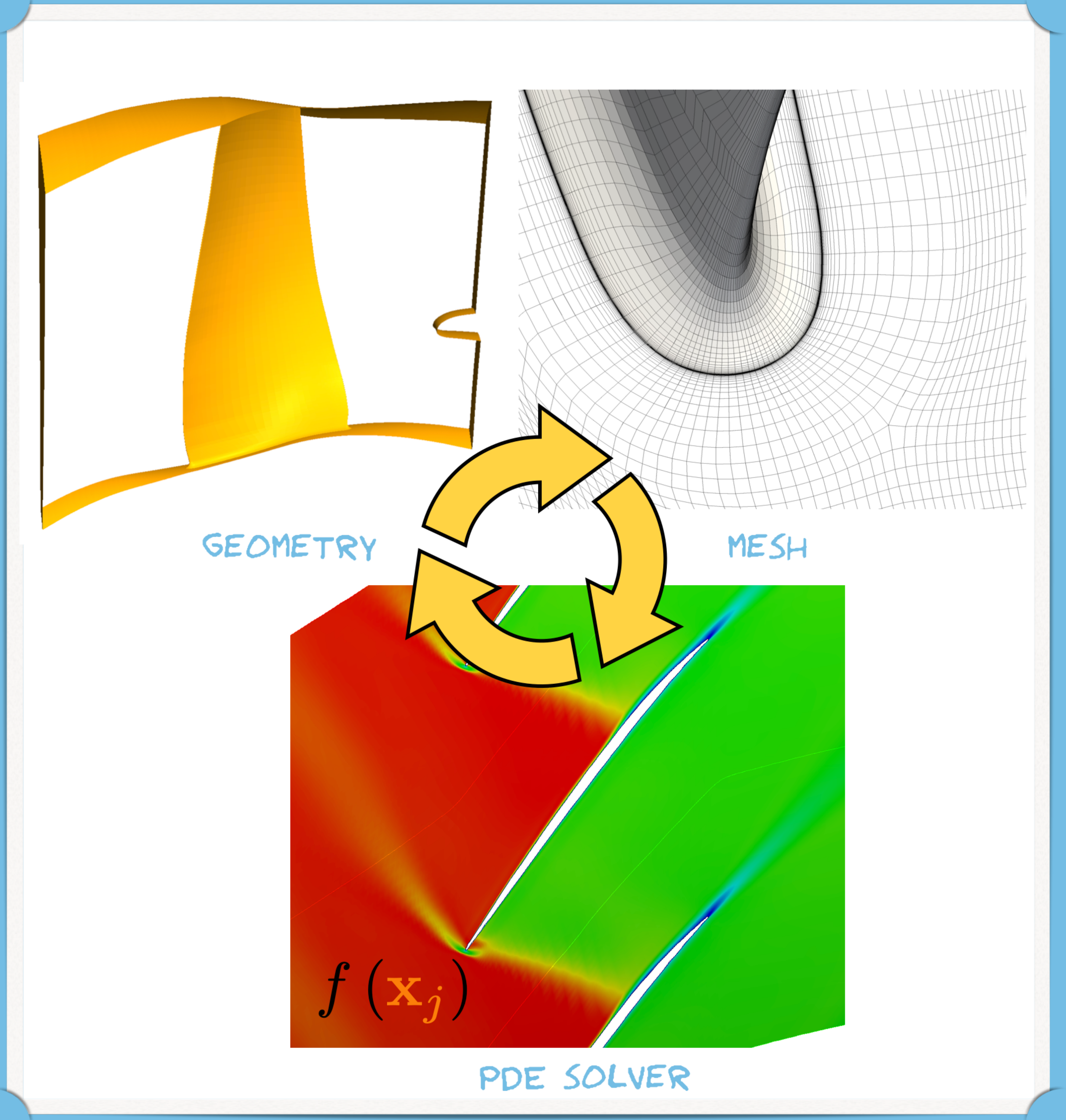
Iterate through the geometry-mesh-solver loop to generate a random design of experiment database.

DESIGNS	x_1	x_2	x_3	…	x_{25}	EFFICIENCY
1	-0.32	0.52	0.81		-0.19	93.1
2	0.55	0.37	-0.49		-0.33	92.8
⋮		⋮			⋮	⋮
350	0.74	0.61	0.31		0.16	91.4

```
sample_points=X
```

Polynomial ridge least squares

Data-driven dimension reduction



Iterate through the geometry-mesh-solver loop to generate a random design of experiment database.

DESIGNS	x_1	x_2	x_3	…	x_{25}	EFFICIENCY
1	-0.32	0.52	0.81		-0.19	93.1
2	0.55	0.37	-0.49		-0.33	92.8
⋮		⋮			⋮	⋮
350	0.74	0.61	0.31		0.16	91.4

`sample_points=X`   `sample_outputs=y`

$x \in \mathbb{R}^{25}$

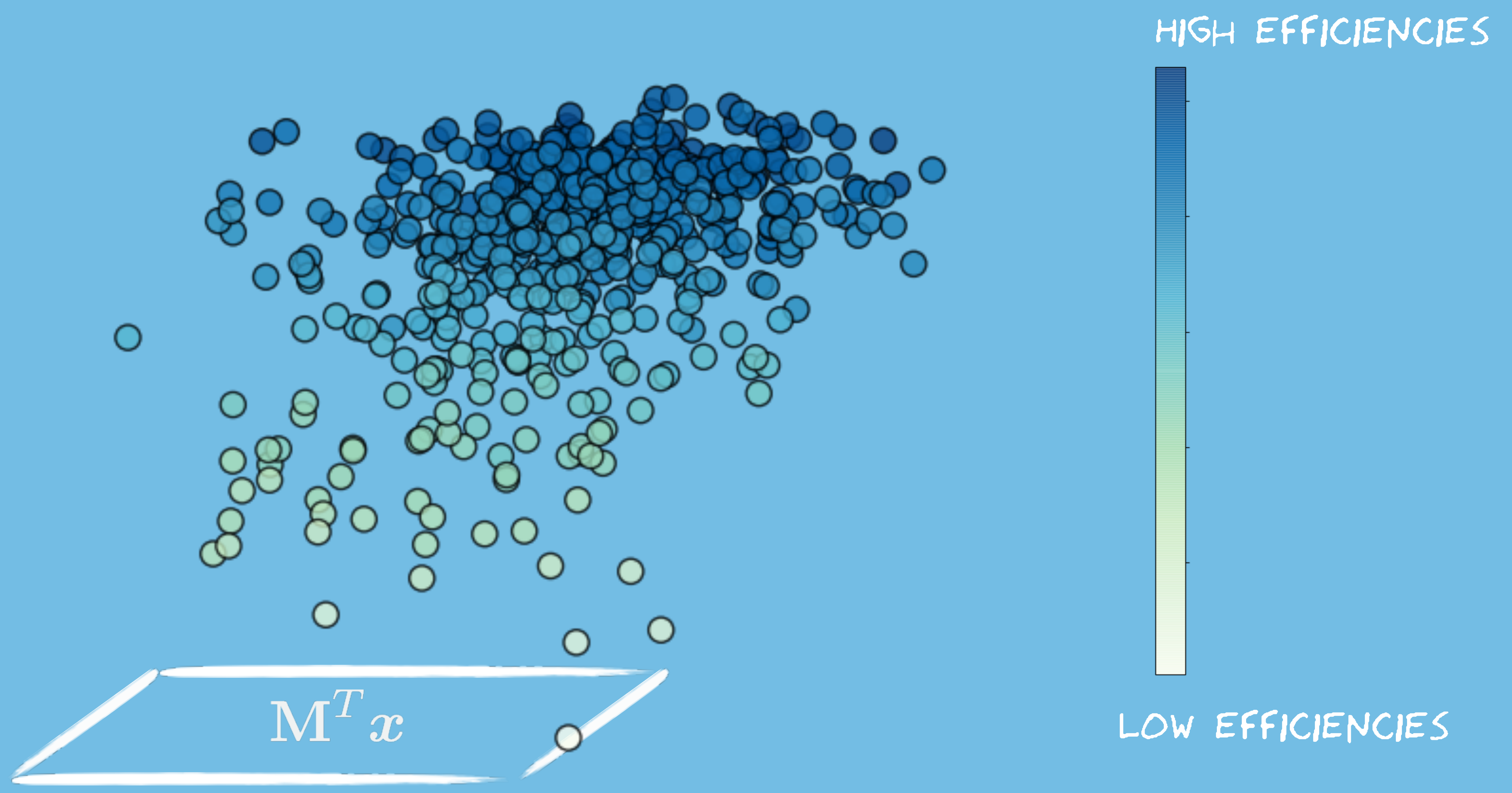
Polynomial ridge least squares

Data-driven dimension reduction

IF WE PROJECT THIS INPUT DATA RANDOMLY...

$$\mathbf{M}^T \boldsymbol{x}$$

$$\mathbf{M} \in \mathbb{R}^{25 \times 2} \qquad \boldsymbol{x} \in \mathbb{R}^{25}$$



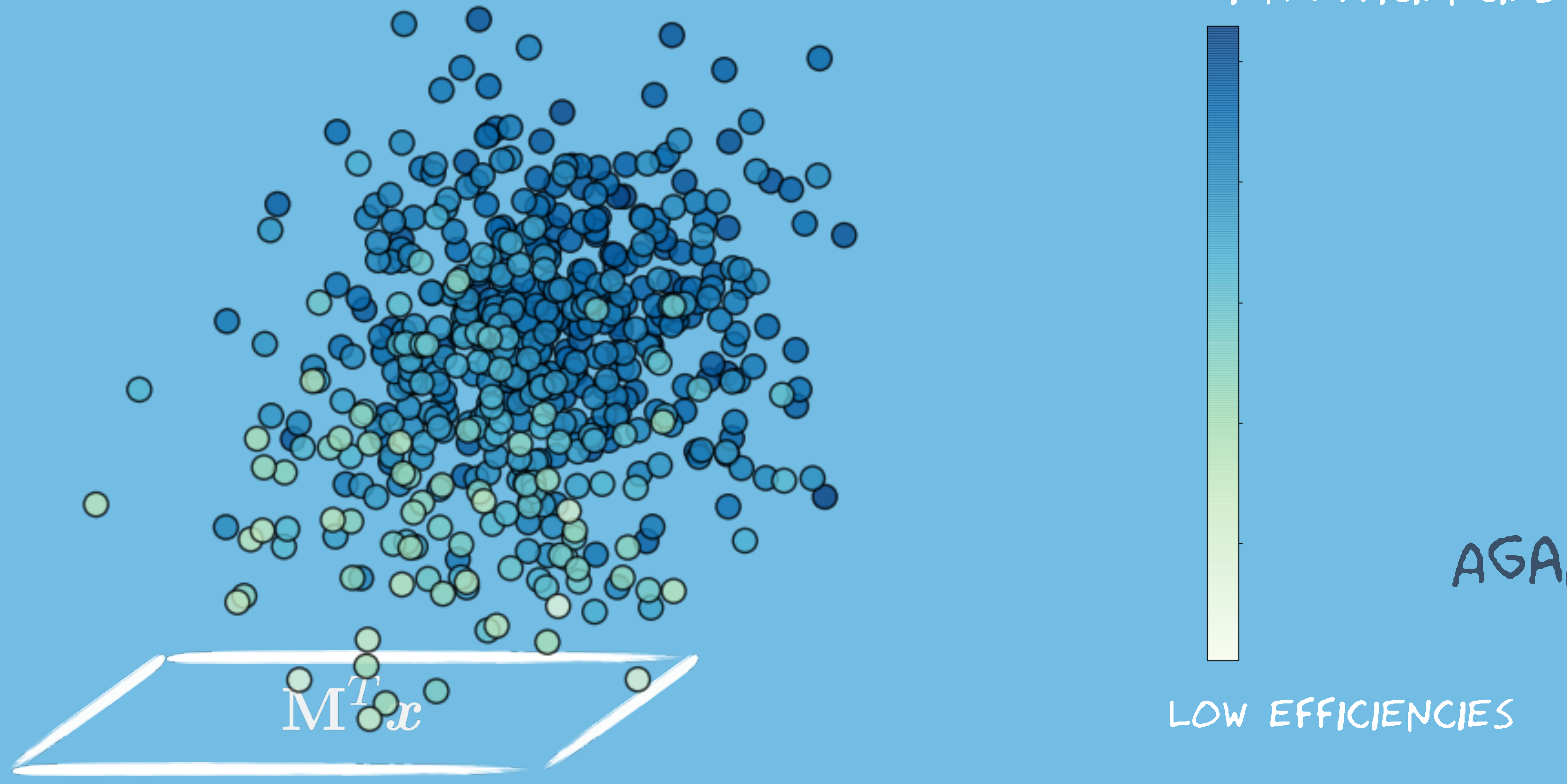
Polynomial ridge least squares

Data-driven dimension reduction

IF WE PROJECT THIS INPUT DATA RANDOMLY...

$$\mathbf{M}^T \boldsymbol{x}$$

$$\mathbf{M} \in \mathbb{R}^{25 \times 2} \qquad \boldsymbol{x} \in \mathbb{R}^{25}$$



AGAIN

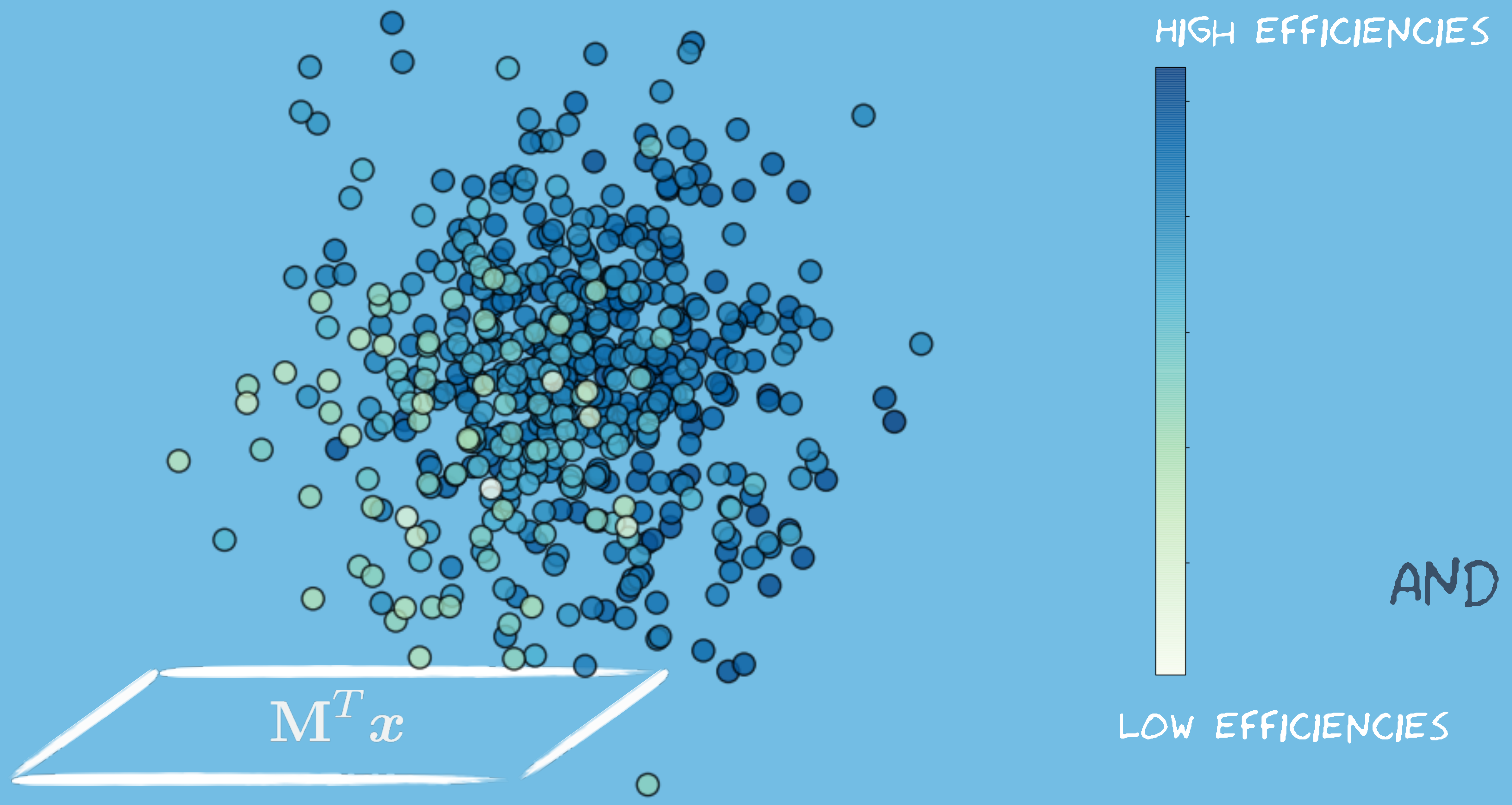
Polynomial ridge least squares

Data-driven dimension reduction

IF WE PROJECT THIS INPUT DATA RANDOMLY...

$$\mathbf{M}^T \boldsymbol{x}$$

$\mathbf{M} \in \mathbb{R}^{25 \times 2}$ $\boldsymbol{x} \in \mathbb{R}^{25}$



AND AGAIN

Polynomial ridge least squares

Data-driven dimension reduction

```
from equadratures import *

space = Subspaces(method='variable-projection', sample_points=X, sample_outputs=y)
M = sspace.get_subspace()
subspace_poly = space.get_subspace_polynomial()
subspace_poly.get_mean_and_variance()
```

WE SOLVE THE FOLLOWING PROBLEM USING HOKANSON & CONSTANTINE (2018) VIA THE METHOD OF SEPARABLE NON-LINEAR LEAST-SQUARES.

$$\underset{\mathbf{M},c}{\text{minimise}} \left\| f\left(x\right) - p_c\left(\mathbf{M}^T x\right) \right\|_2^2$$

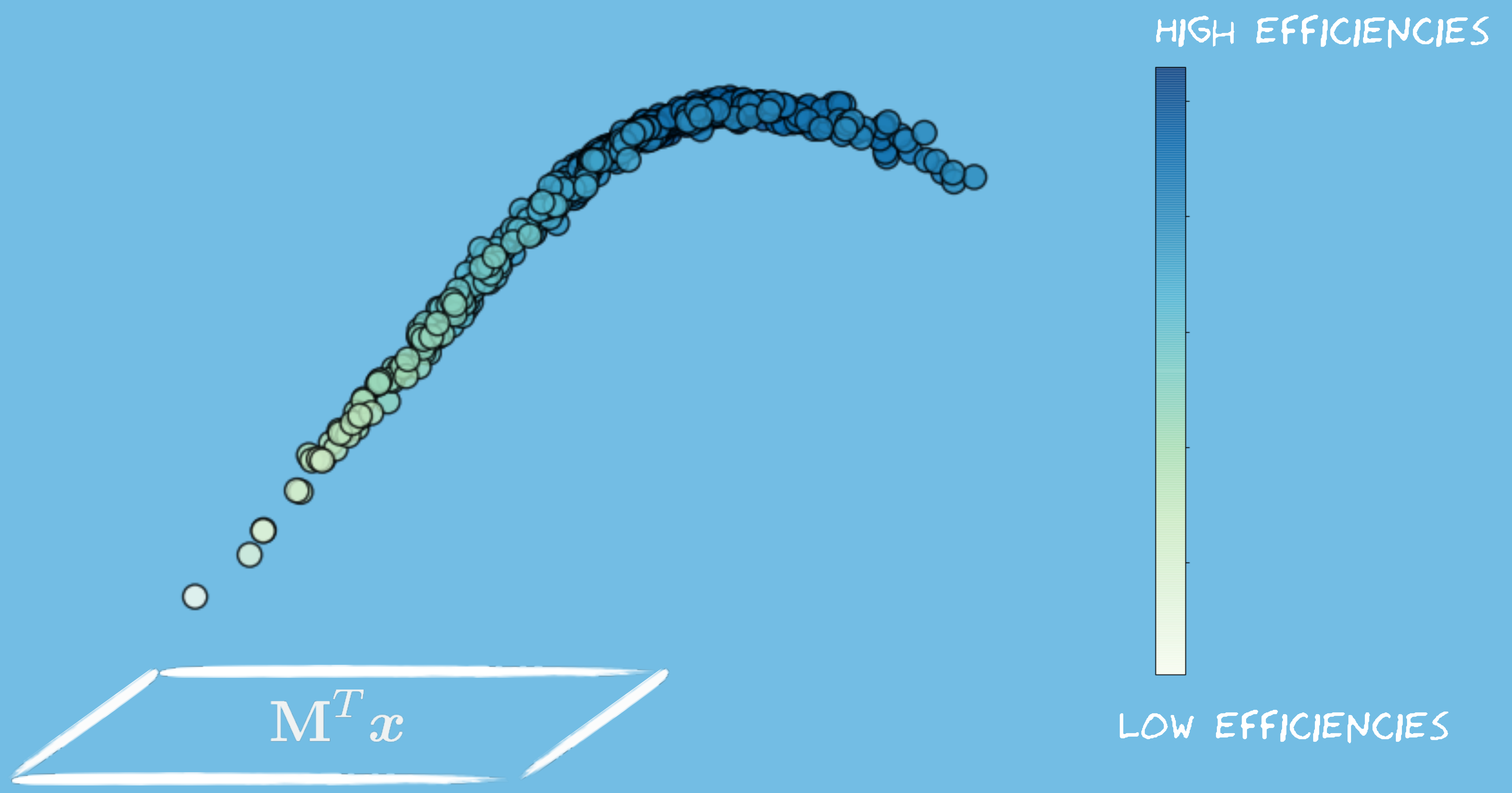
Polynomial ridge least squares

Data-driven dimension reduction

IF WE PROJECT THIS INPUT DATA USING THE COMPUTED SUBSPACE...

$$\mathbf{M}^T \boldsymbol{x}$$

$\mathbf{M} \in \mathbb{R}^{25 \times 2}$ $\boldsymbol{x} \in \mathbb{R}^{25}$



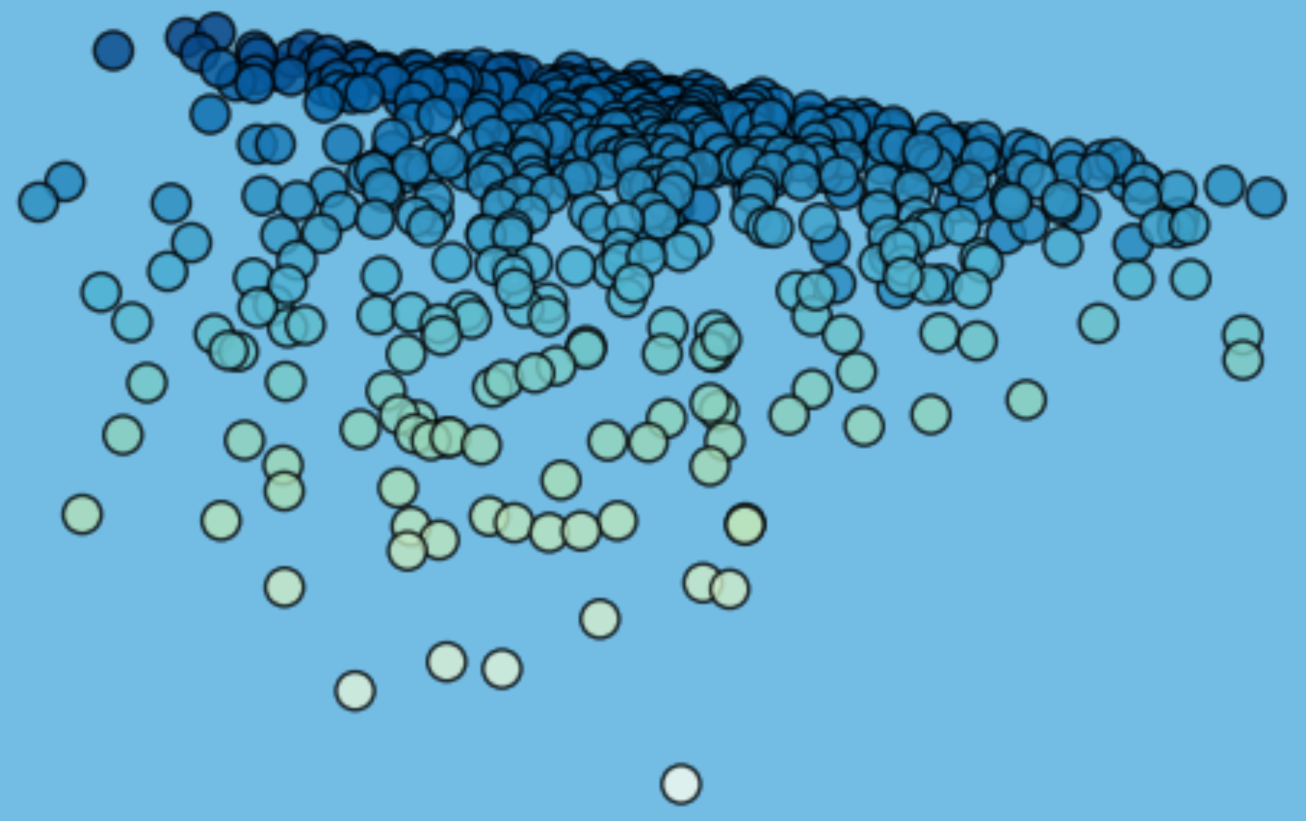
Polynomial ridge least squares

Data-driven dimension reduction

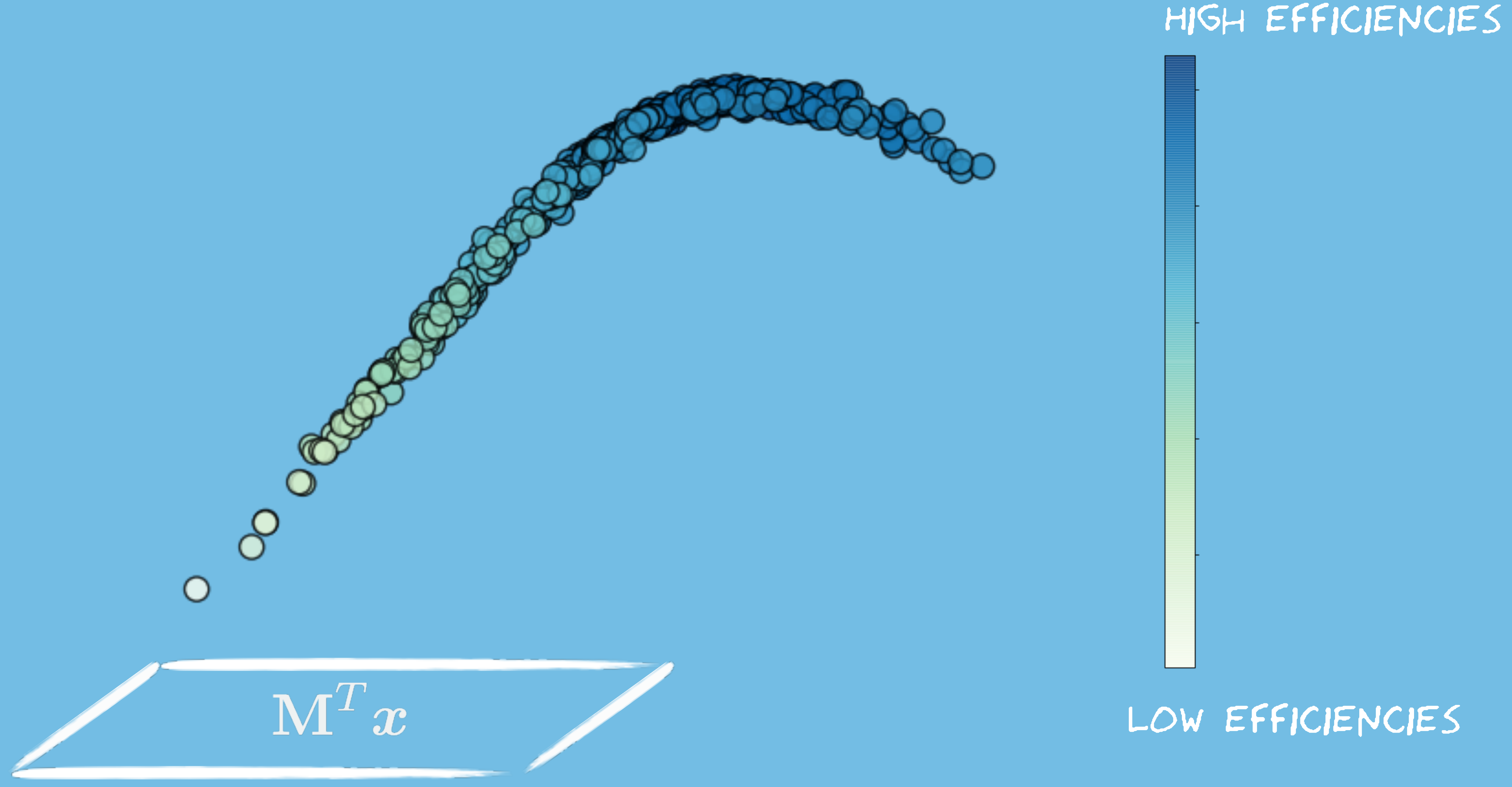
IF WE PROJECT THIS INPUT DATA USING THE COMPUTED SUBSPACE...

$$\mathbf{M}^T \boldsymbol{x}$$

$$\mathbf{M} \in \mathbb{R}^{25 \times 2} \qquad \boldsymbol{x} \in \mathbb{R}^{25}$$



SIDE VIEW



Polynomial ridge least squares

Data-driven dimension reduction

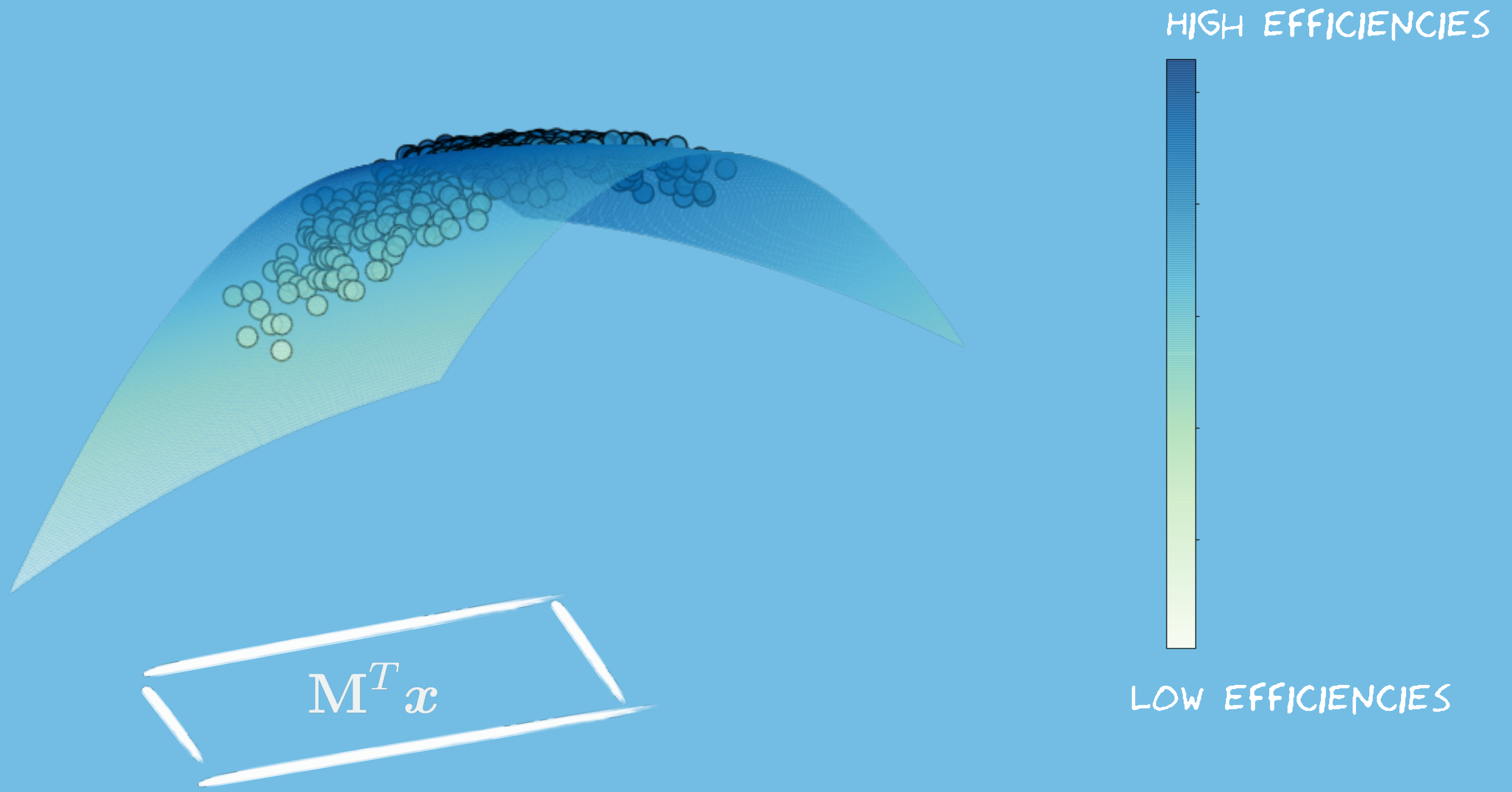
IF WE PROJECT THIS INPUT DATA USING THE COMPUTED SUBSPACE...

$$\mathbf{M}^T \boldsymbol{x}$$

$$\mathbf{M} \in \mathbb{R}^{25 \times 2} \qquad \boldsymbol{x} \in \mathbb{R}^{25}$$

CAN THEN FIT A POLYNOMIAL OVER THIS PROJECTION

$$p\left(\mathbf{M}^T \boldsymbol{x}\right)$$



Polynomial ridge least squares

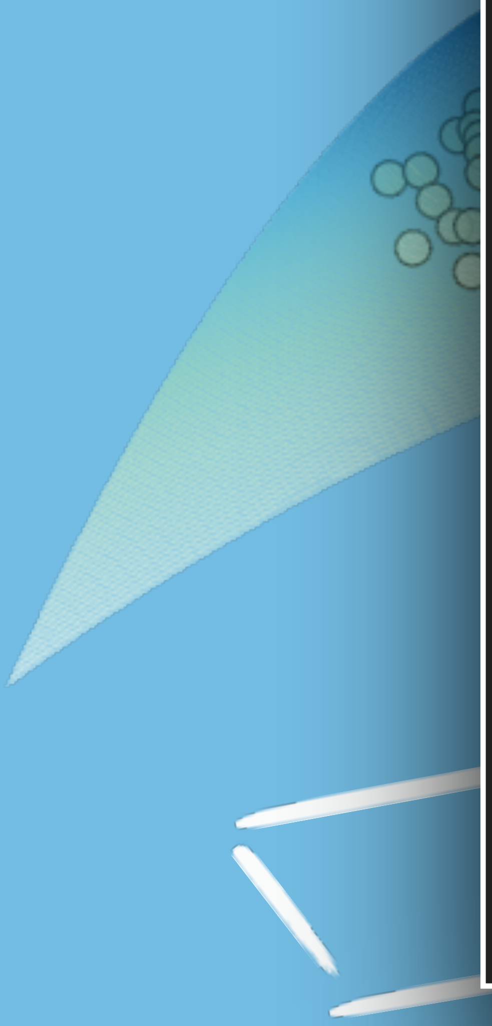
Data-driven dimension reduction

IF WE PROJECT THIS INPUT DATA US

$$\mathrm{M} \in$$

CAN THEN FIT A POLYNOMIAL OVER THIS PROJECTION

$$p\left(\mathrm{M}^{T} x\right)$$



RECALL, ONCE WE HAVE A POLYNOMIAL...

CAN EASILY COMPUTE:

1. MEAN, VARIANCE, SKEWNESS AND KURTOSIS.
2. PROBABILITIES OF OUTPUT.
3. SENSITIVITY INDICES (SUCH AS SOBOL').
4. GRADIENTS (USEFUL FOR OPTIMISATION).
5. CRITERION FOR DESIGN OF EXPERIMENT.

Polynomial ridge least squares

Data-driven dimension reduction

SPLITTING THE SPACE

$$x = \mathbf{I}x$$

Polynomial ridge least squares

Data-driven dimension reduction

SPLITTING THE SPACE

$$x = \mathbf{I}x$$

$$x = \left(\mathrm{M}\mathrm{M}^T + \mathrm{N}\mathrm{N}^T\right)x$$

ORTHOGONAL COMPLEMENT

Capability

Data-driven dimension reduction

SPLITTING THE SPACE

$$x = \mathbf{I}x$$

$$x = \left(\mathbf{M}\mathbf{M}^T + \mathbf{N}\mathbf{N}^T\right)x \quad \text{ORTHOGONAL COMPLEMENT}$$

$$x = \mathbf{M}\mathbf{M}^T x + \mathbf{N}\mathbf{N}^T x$$

ACTIVE SUBSPACE

INACTIVE SUBSPACE

CAN WE EXPLOIT THIS SPACE FOR DESIGN?

Polynomial ridge least squares

Applications – browse over the QR codes with your device

Fan blade design with multiple objectives.

Temperature probe design.

Fan blade design at multiple operating points.

Polynomial ridge least squares

Data-driven dimension reduction

SPLITTING THE SPACE

$$x = \mathbf{I}x$$

$$x = \left(\mathbf{M}\mathbf{M}^T + \mathbf{N}\mathbf{N}^T\right) x$$

ORTHOGONAL COMPLEMENT

$$x = \mathbf{M}\mathbf{M}^T x + \mathbf{N}\mathbf{N}^T x$$

ACTIVE SUBSPACE

INACTIVE SUBSPACE

CAN WE EXPLOIT THIS SPACE FOR SETTING MANUFACTURING TOLERANCES?

Polynomial ridge least squares

Data-driven dimension reduction

There was so much to explore, we wrote a two-part pre-print on the subject (on arXiv in a week)!

Central idea is to generate samples for both scalar- and vector-valued objectives from their inactive subspaces.

Blade Envelopes Part I: Concept and Methodology

Chun Yui Wong[†]; Pranay Seshadri[‡*], Ashley Scillitoe[*], Andrew Duncan[‡*], Geoffrey Parks[†]

[†]Department of Engineering, University of Cambridge, U.K.

[‡]Department of Mathematics, Imperial College London, U.K.

[*]Data-Centric Engineering, The Alan Turing Institute, U.K.

Blades manufactured through flank and point milling will likely exhibit geometric variability. Gauging the aerodynamic repercussions of such variability, prior to manufacturing a component, is challenging enough, let alone trying to predict what the amplified impact of any in-service degradation will be. While rules of thumb that govern the tolerance band can be devised based on expected boundary layer characteristics at known regions and levels of degradation, it remains a challenge to translate these insights into quantitative bounds for manufacturing. In this work, we tackle this challenge by leveraging ideas from dimension reduction to construct low-dimensional representations of aerodynamic performance metrics. These low-dimensional models can identify a subspace which contains designs that are invariant in performance—the inactive subspace. By sampling within this subspace, we design techniques for drafting manufacturing tolerances and for quantifying whether a scanned component should be used or scrapped. We introduce the blade envelope *as a visual and computational manufacturing guide for a blade. In this paper, the first of two parts, we discuss its underlying concept and detail its computational methodology, assuming one is interested only in the single objective of ensuring that the loss of all manufactured blades remains constant. To demonstrate the utility of our ideas we devise a series of computational experiments with the Von Karman Institute's LS89 turbine blade.*

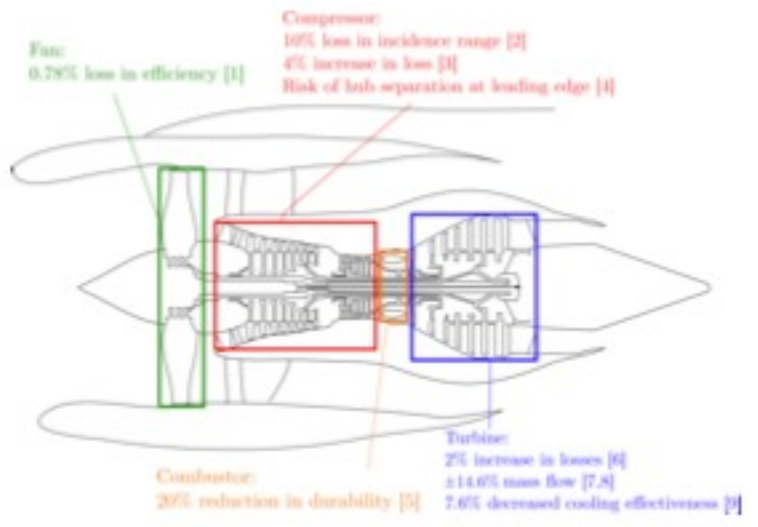


Fig. 1. Impacts associated with manufacturing variations in a jet engine; see [1–9].

1 INTRODUCTION

Manufacturing variations and in-service degradation have a sizeable impact on aerodynamic performance of a jet engine (see Figure 1). Gauging the aerodynamic repercussions of such variability prior to manufacturing a component is challenging enough, let alone trying to predict what the amplified impact of any in-service degradation might be. In a bid to reduce losses and mitigate the risks in Figure 1, designers today pursue a two-pronged approach. First, components are being designed to operate over a range of conditions (and uncertainties therein) via both robust optimization techniques [10, 11] as well as more traditional design guides such as loss buckets—i.e., loss across a range of positive and negative incidence angles [4]. In parallel, there has been a growing research effort to assess 3D manufacturing variations and in-service degradation by optically scanning (via GOM) the manufactured blades, meshing them, and running them through a flow solver [12]. Both approaches, while useful in extracting aerodynamic inference, are limiting. One of the key bottlenecks is the cost of evaluating flow quantities of interest via computational fluid dynamics (CFD), as the dimensionality of the space of manufactured geometries is too large to fully explore, even with an appropriately tailored design of experiments (DoE). To reduce the dimensionality, some authors [3, 12] use principal components analysis (PCA) to extract a few *manufacturing modes*, which correspond to modes of largest manufacturing deviation observed in the scanned blades. One drawback of this approach is that the PCA model is not performance-based, i.e. the mode of greatest geometric variability need not correspond to the mode of greatest performance scatter, a point raised by Dow and Wang [13]. Additionally, GOM scans can only be carried out on cold and manufactured components, ignoring the uncertainty on performance associated with in-service operating conditions. Finally, through neither of these paths are we offering manufacturing engineers a set of pedigree rules or guides on manufacturing for an individual component, prior to actually manufacturing the component. This motivates some of the advances in this paper. We argue that challenges associated with both manufacturing vari-

*Address all correspondence to Chun Yui Wong, cyw28@cam.ac.uk

Blade Envelopes Part II: Multiple Objectives and Inverse Design

Chun Yui Wong[†]; Pranay Seshadri[‡*], Ashley Scillitoe[*], Bryn Ubald[*]
Andrew Duncan[‡*], Geoffrey Parks[†]

[†]Department of Engineering, University of Cambridge, U.K.

[‡]Department of Mathematics, Imperial College London, U.K.

[*]Data-Centric Engineering, The Alan Turing Institute, U.K.

Blade envelopes offer a set of data-driven tolerance guidelines for manufactured components based on aerodynamic analysis. In part I of this two-part paper, a workflow for the formulation of blade envelopes is described and demonstrated. In part II, this workflow is extended to accommodate multiple objectives. This allows engineers to prescribe manufacturing guidelines that take into account multiple performance criteria.

The quality of a manufactured blade can be correlated with features derived from the distribution of primal flow quantities over the surface. We show that these distributions can be accounted for in the blade envelope using vector-valued models derived from discrete surface flow measurements. Our methods result in a set of variables that allow flexible and independent control over multiple flow characteristics and performance metrics, similar in spirit to inverse design methods. The augmentations to the blade envelope workflow presented in this paper are demonstrated on the LS89 turbine blade, focusing on the control of loss, mass flow and the isentropic Mach number distribution.

1 INTRODUCTION

In the first part of this two-part paper [1], we defined the concept of a *blade envelope*, a visual and computational guideline yielding automatic scrap-or-use decisions of manufactured turbomachinery components. Using the theory of inactive subspaces, a range of geometric designs that are invariant in loss is identified, and geometries from this invariant region can be generated with no additional computational fluid dynamics (CFD) solves. From this, the decision to scrap or keep a measured component reduces down to the computation of the Mahalanobis distance from an aerodynamic knowledge base consisting of invariant designs.

In the second part, we extend blade envelopes beyond the manufacturing stage of production, and describe how they can be used during the design stage as well. During the shape design of a highly-loaded turbine stage, the minimization of loss is often accompanied with constraints to avoid trivial solutions where the blade is unloaded. For example, in [2], the exit flow angle is constrained to be above the baseline value to ensure sufficient work extraction. In [3], the authors put an equality constraint on the mass flow rate while optimizing the loss coefficient to factor out possible reduction in entropy generation due to reduction in flow capacity. Prior work [4, 5] has leveraged active subspaces to construct 2D performance maps for compressor blade design. In the latter work, multiple objectives including the pressure ratio and flow capacity are considered by mapping contours of different objectives onto the active subspace of efficiency. Manufacturing deviations are modeled as constant excursions from the nominal design. The main drawback of this approach is the requirement to run further simulations to map out performance contours in the active subspace. In this work, we incorporate multiple aerodynamic design requirements by interpreting them as additional constraints factored into blade envelopes.

In situations where tighter control over the performance of the component is required, constraints on surface flow characteristics can be implemented. Clark [6] establishes the correlation between aerodynamic features—defined via parts of the surface isentropic Mach number distribution—and aerodynamic performance. Control over these key features can be achieved by factoring the isentropic Mach number distribution as an additional *vector-valued* objective in blade envelopes. This approach is similar in spirit to *inverse design*, where a target distribution is specified on the surface of a blade, and the blade shape is iteratively modified to give a geometry that matches the distribution. While inverse design yields an optimal geometry that fits the design criterion over the entire surface, our approach aims to find designs that satisfy the target distribution in parts of the flow that are most critical to performance. The relaxation of constraints on other locations allow a range of designs to be specified, whose expanse is explicitly quantified by the blade envelope. Moreover, we can combine the control over the surface flow profile with constraints over other scalar objectives to perform inverse design constrained on requirements on other measures of performance.

2 COMPUTATIONAL METHODOLOGIES

Blade envelopes demarcate boundaries within the space of manufactured geometries that correspond to confidence intervals of performance metrics. These envelopes are formed from statistics derived from an aerodynamic database containing geometries sampled from the inactive subspace with respect to a scalar objective. Building upon this framework, we describe two

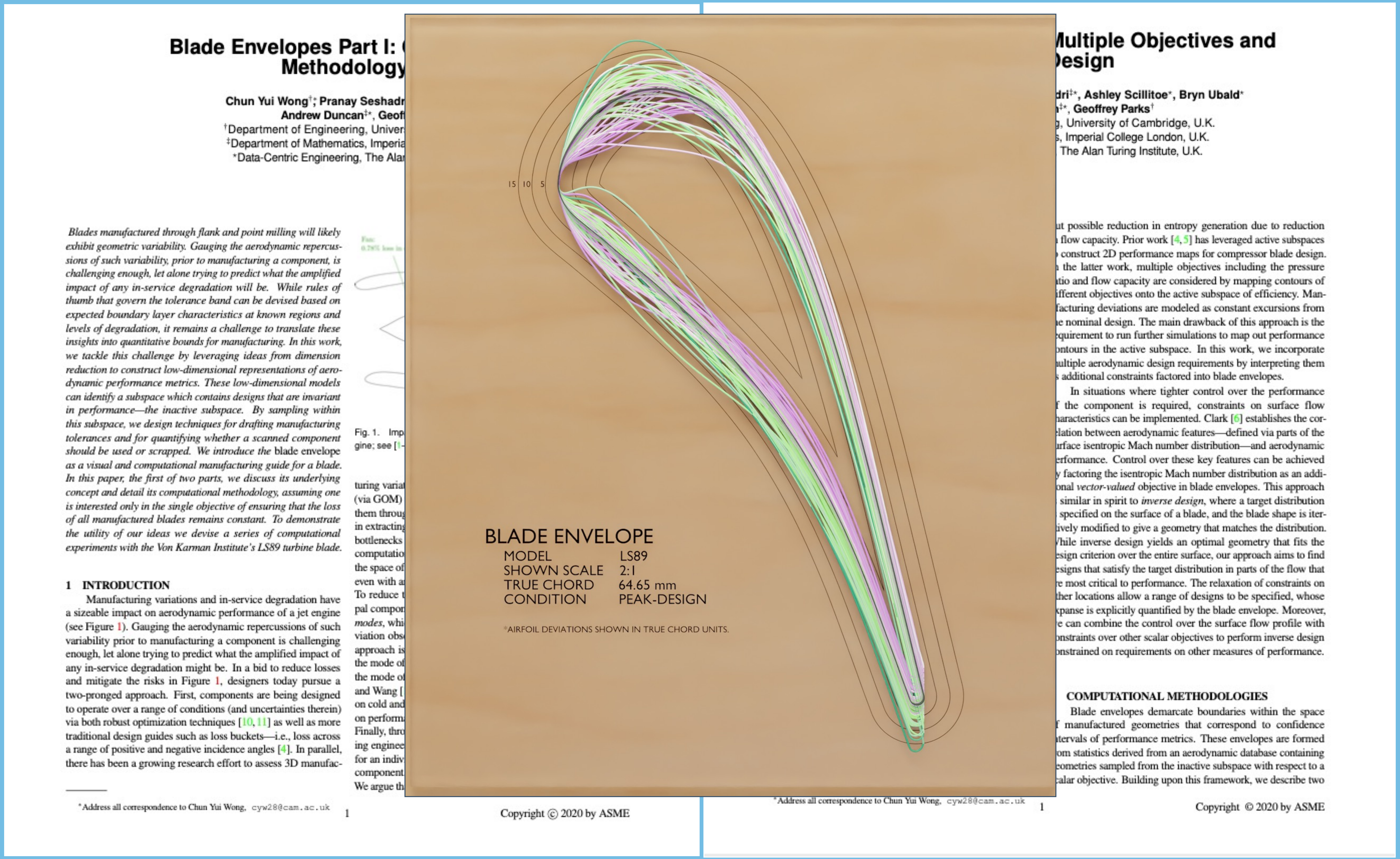
*Address all correspondence to Chun Yui Wong, cyw28@cam.ac.uk

Polynomial ridge least squares

Data-driven dimension reduction

There was so much to explore, we wrote a two-part pre-print on the subject (on arXiv in a week)!

Central idea is to generate samples for both scalar- and vector-valued objectives from their inactive subspaces.



Polynomial ridge least squares

Data-driven dimension reduction

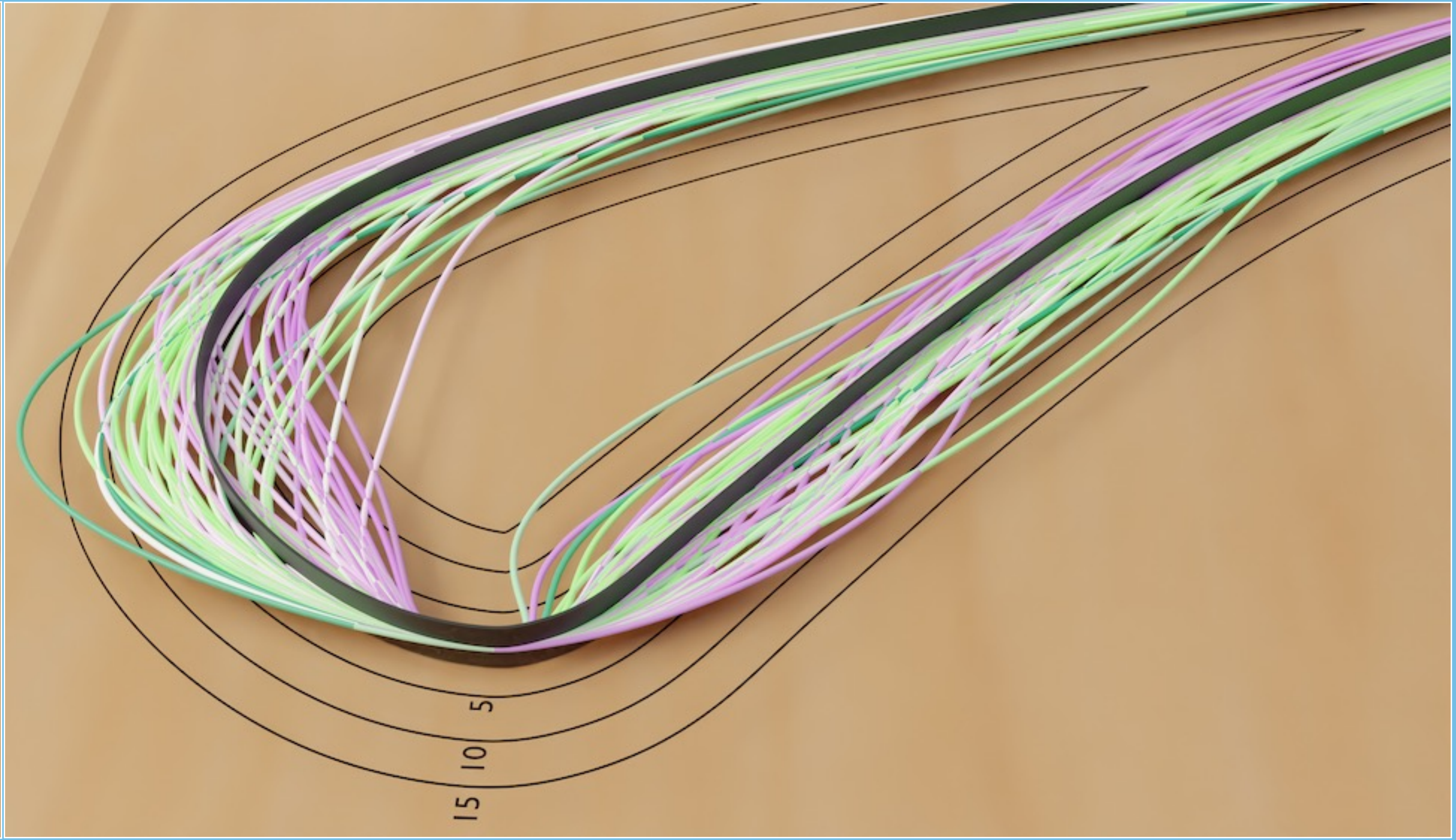
```
from equadratures import *
from scipy.linalg import null_space

space = Subspaces(method='variable-projection', sample_points=X, sample_outputs=y)
M = sspace.get_subspace() # M is a 25 by 2 matrix
N = null_space(M.T) # returns a 25 by 23 matrix
```

Polynomial ridge least squares

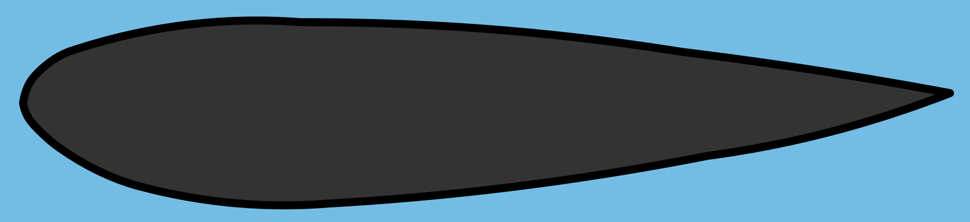
Data-driven dimension reduction

Can 3D print this and use it to make more well-informed design and manufacturing decisions.



Polynomial ridge least squares

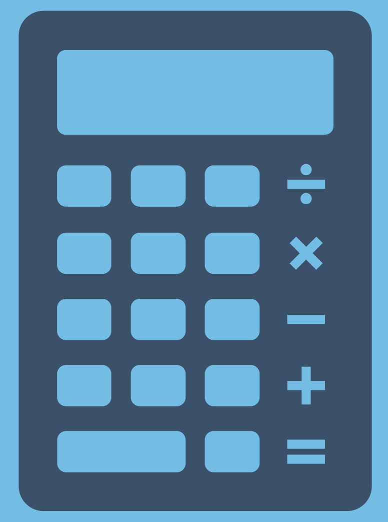
CFD flow-field estimation application



Input parameters

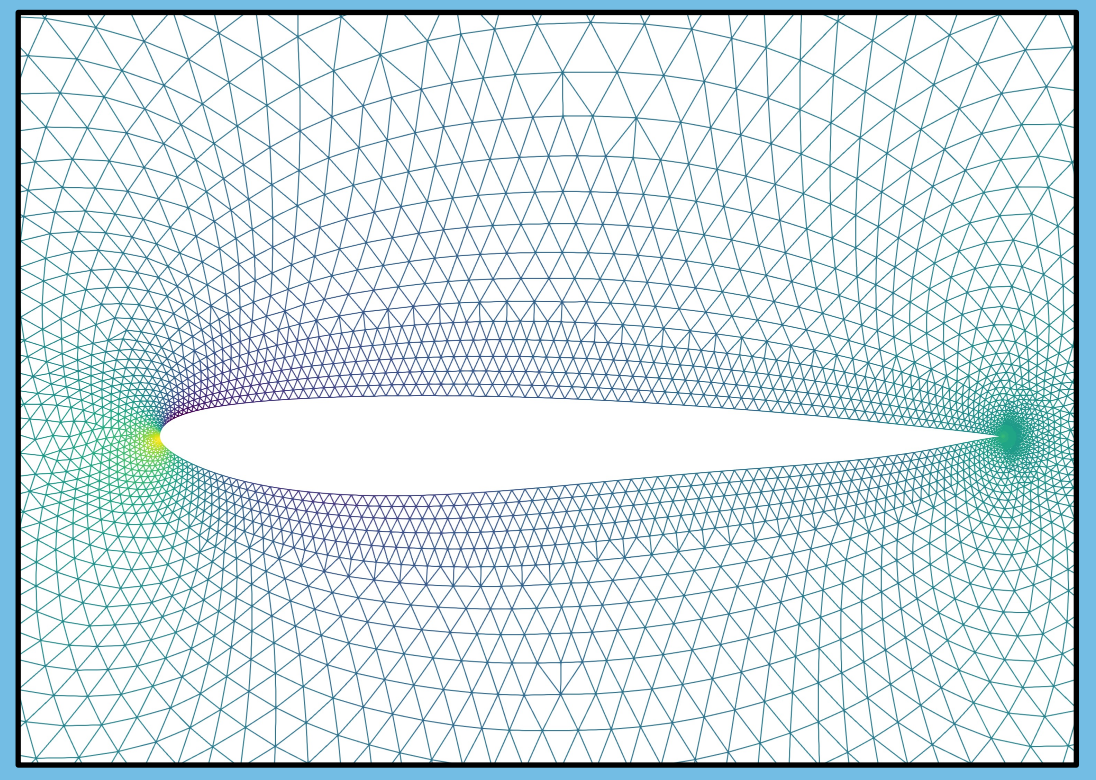
$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{50} \end{bmatrix}$$

50 HICKS-HENNE BUMP FUNCTIONS



COMPUTATIONAL FLUID DYNAMICS (REYNOLDS AVERAGED NAVIER STOKES)

$\boldsymbol{x} \in \mathbb{R}^{50}$



Output parameters

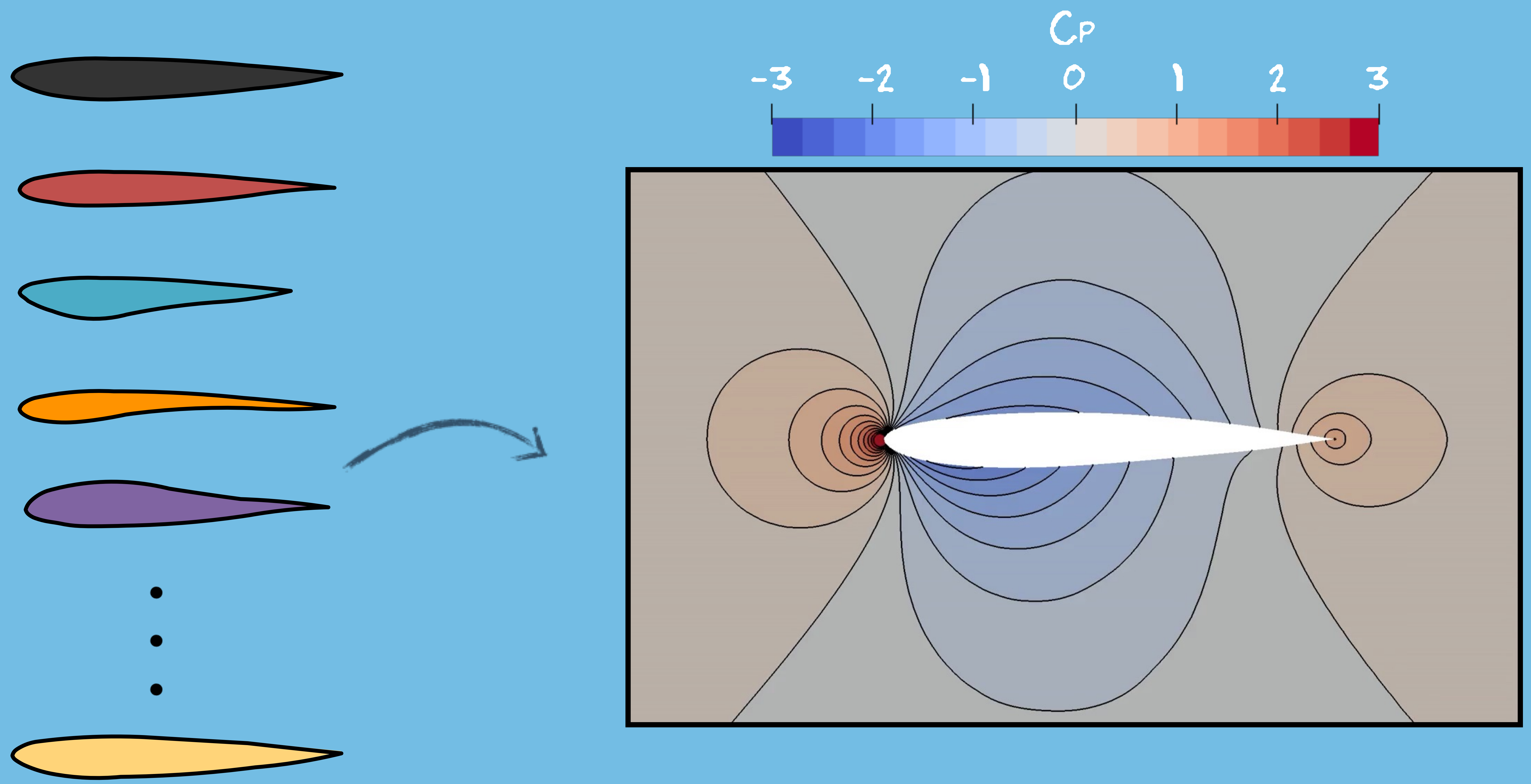
$$f\left(\boldsymbol{x}\right)$$

VELOCITY OR PRESSURE AT EACH NODE (ITS A VECTOR!)

Polynomial ridge least squares

CFD flow-field estimation application

Can we use polynomial ridge least squares to estimate the spatial field of static pressure, given a small database of CFD evaluations?



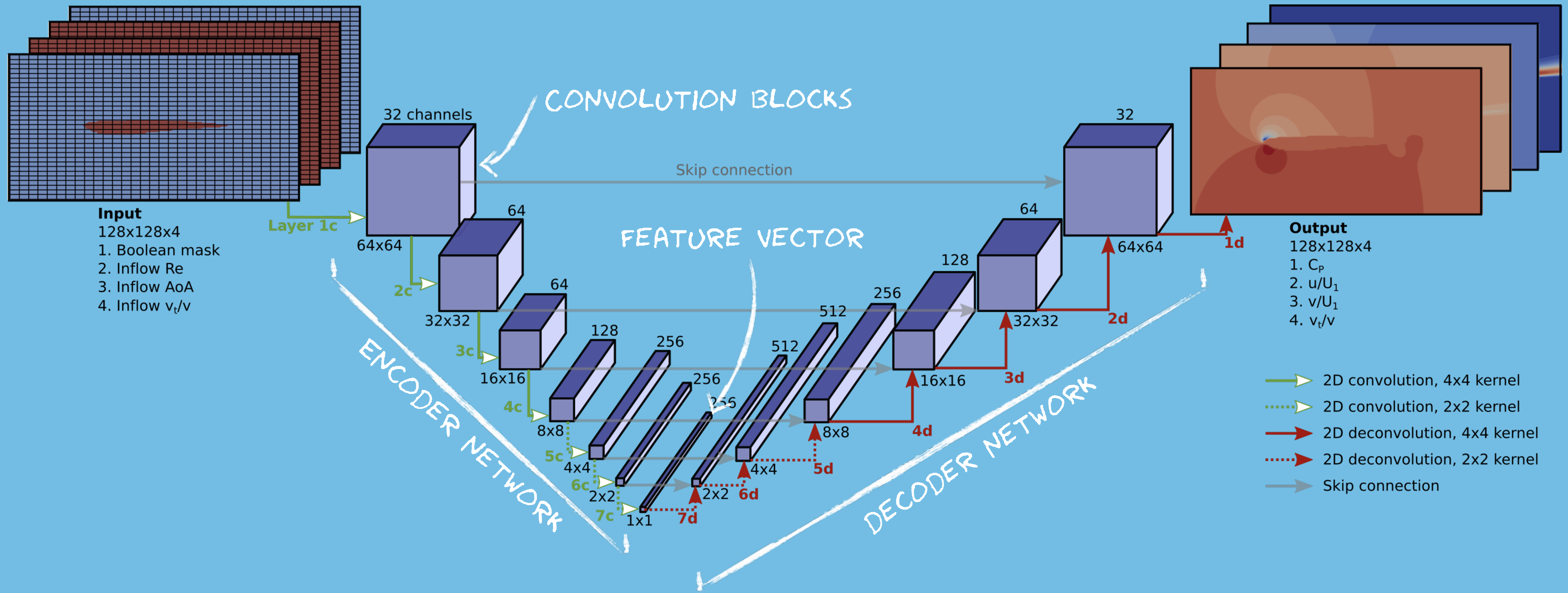
DIFFERENT DESIGNS

VELOCITY, PRESSURE, TURBULENCE FIELDS

Polynomial ridge least squares

CFD flow-field estimation application

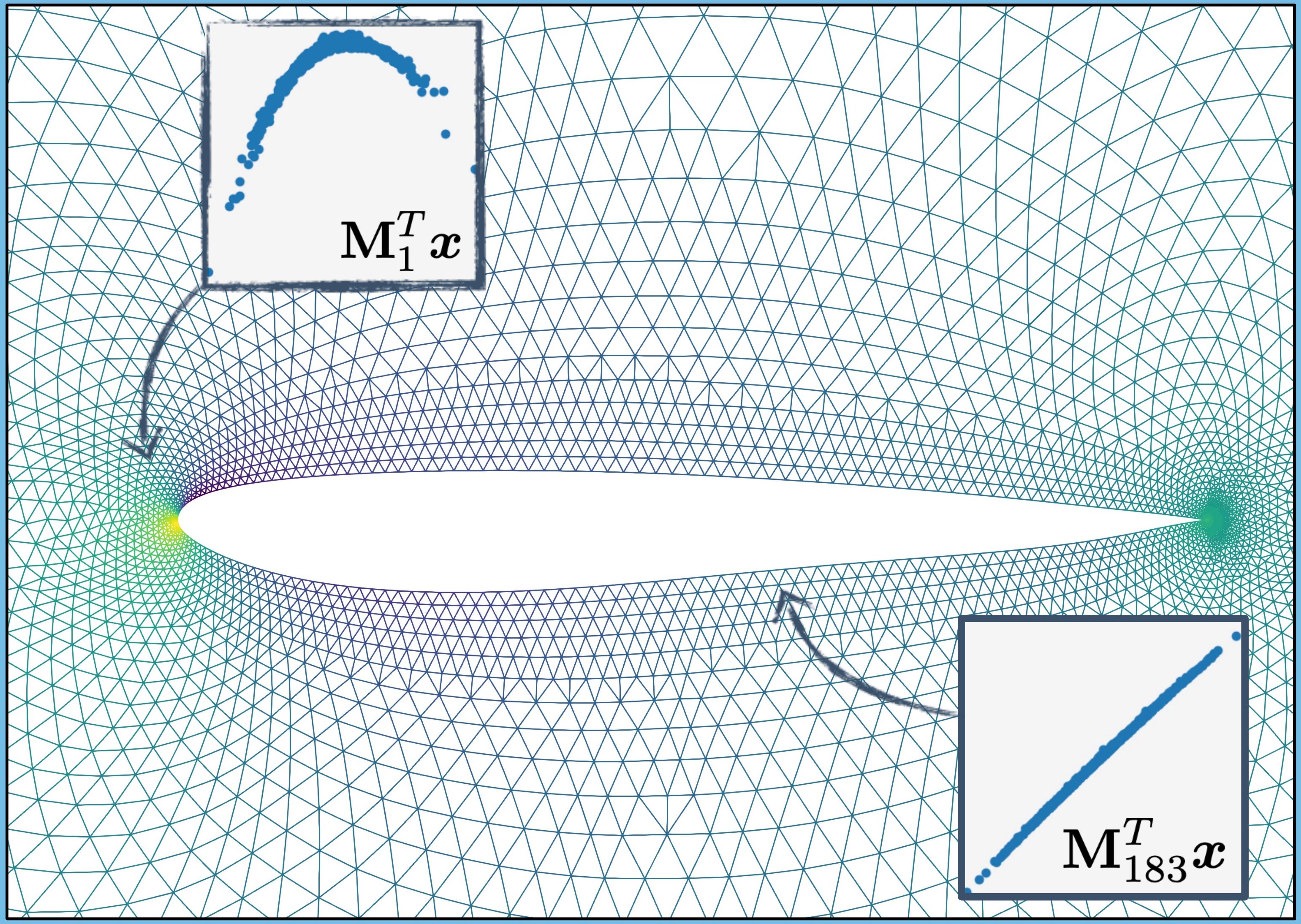
But, before we look into that, one can use a convolution neural network to approach this (from *literature* it seems that this is what all the cool kids are doing).



Polynomial ridge least squares

CFD flow-field estimation application

But there is some physical insight, if we exploit. Each node exhibits a ridge-like structure.

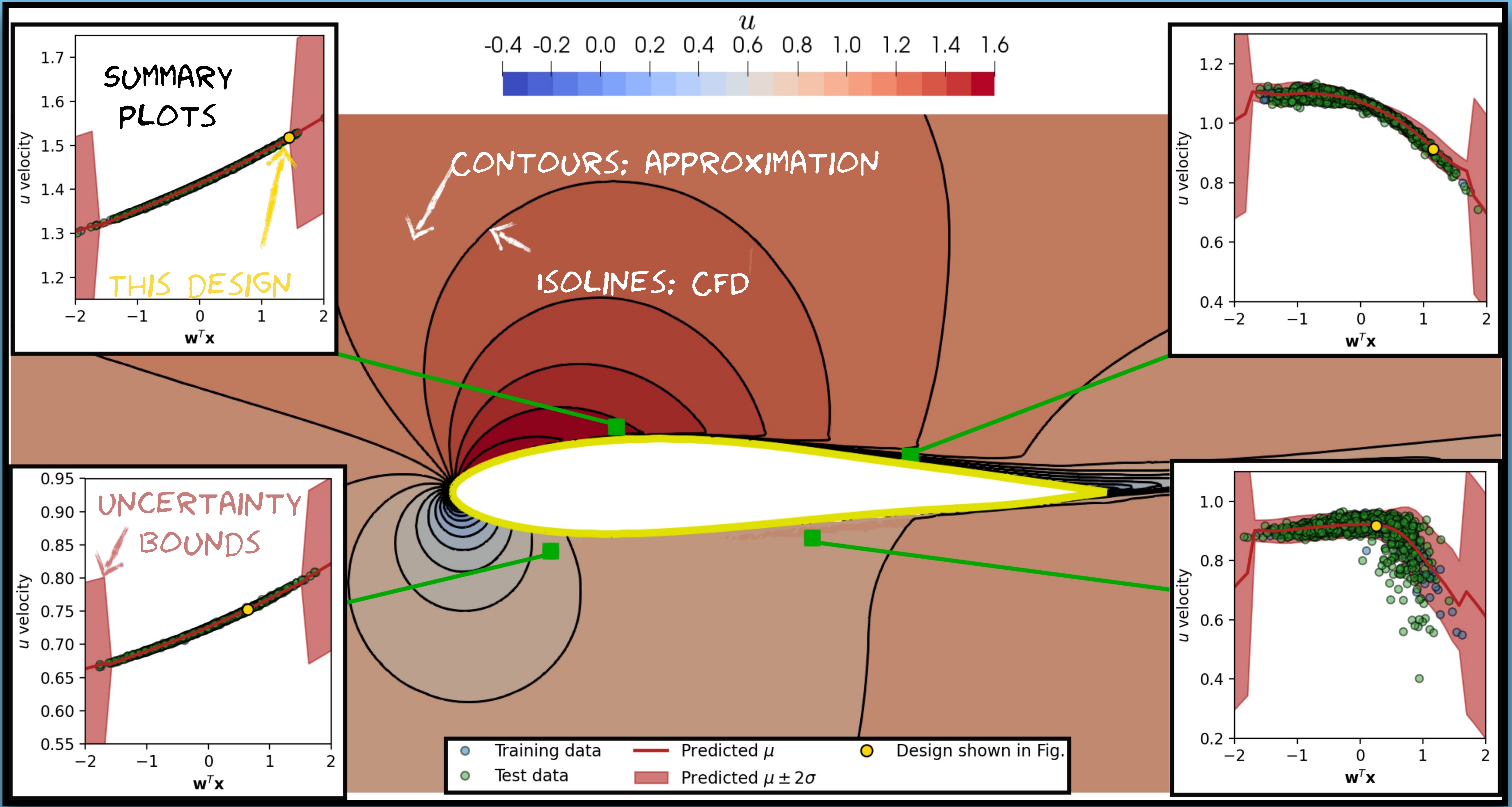


$$\mathbf{f} \approx \begin{bmatrix} g_1 \left(\mathbf{M}_1^T x \right) \\ \vdots \\ g_m \left(\mathbf{M}_m^T x \right) \end{bmatrix}$$

Polynomial ridge least squares

CFD flow-field estimation application

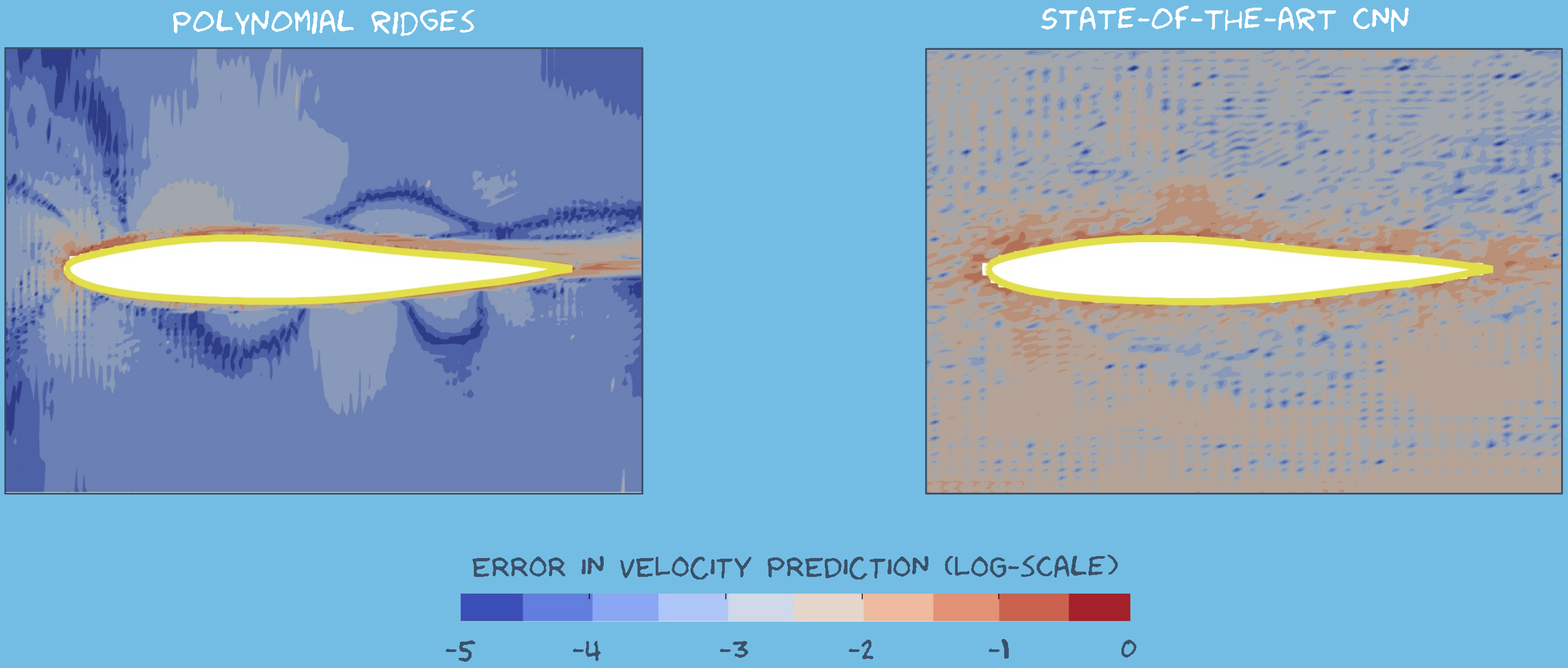
We can exploit this ridge-like structure to rapidly predict flow-fields.



Polynomial ridge least squares

CFD flow-field estimation application

Accuracy competitive with CNN!

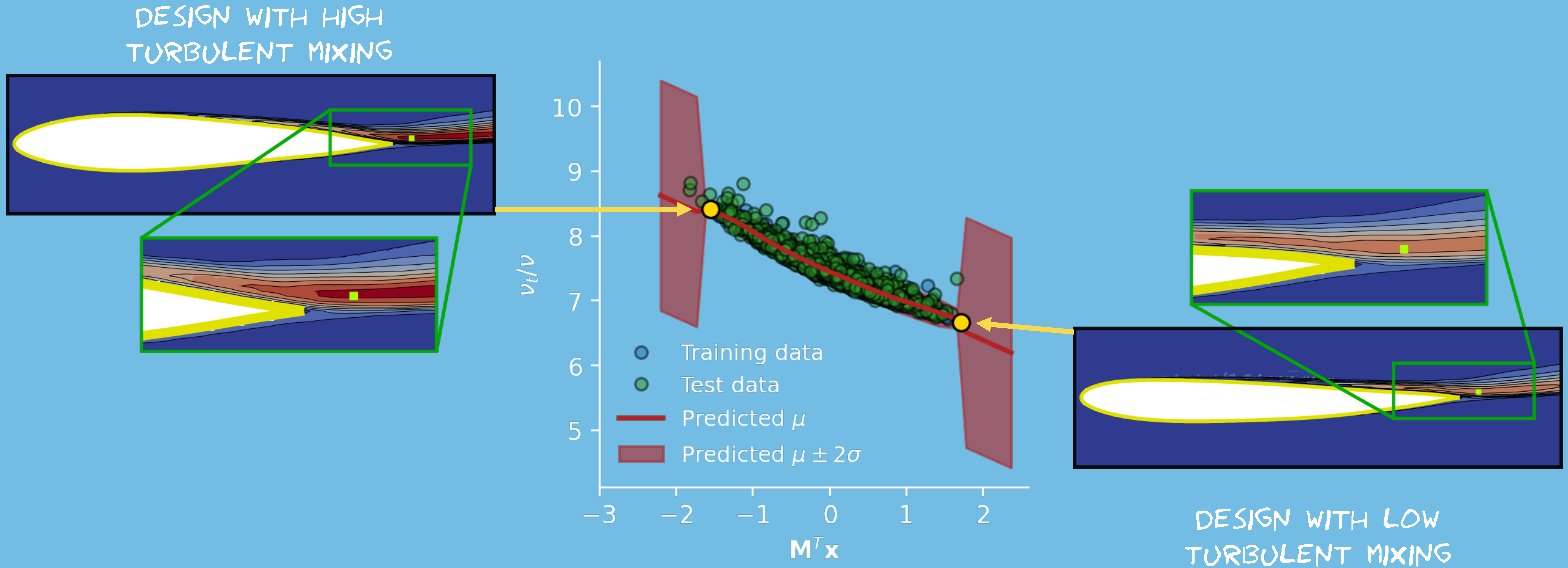


Polynomial ridge least squares

CFD flow-field estimation application

Provides more physical insight compared to CNN’s.

Example: predicting turbulent viscosity ratio:



Concluding thoughts

Polynomial approximations

Conclusions

Tremendous body of work dedicated to polynomial least squares over the past decade with a key focus on biased sampling approaches and tractable computational strategies.

In cases where physical problems admit ridge like structure, polynomial ridge approximations can be very powerful, and can abate the curse of dimensionality.

Thank You

p.seshadri <at> imperial.ac.uk
www.psesh.com

For codes, publications, and blog posts please see equadratures.org