Data sampling for surrogate modeling and optimization

Tyler Chang

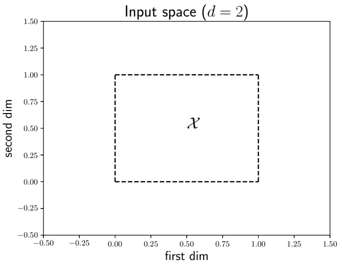
MCS Division, Argonne National Laboratory

ICIAM 2023 – Tokyo, Japan

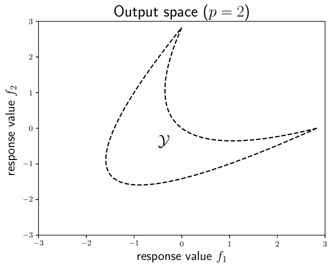
Outlines

Problem Setting and Some Background

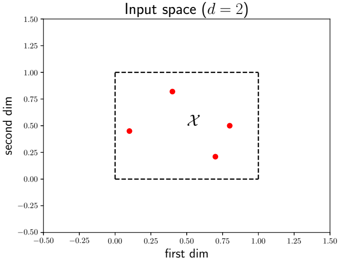
Multivariate Interpolation



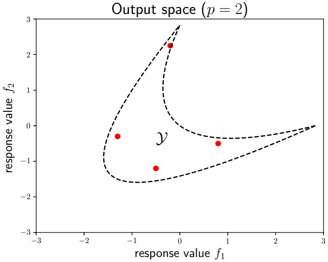
$$F : \mathcal{X} \to \mathcal{Y}$$



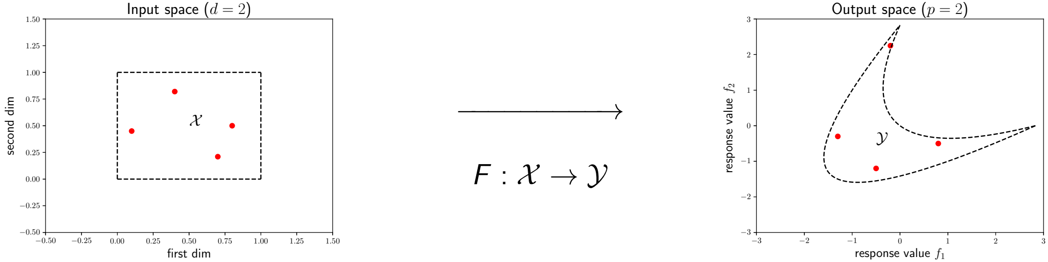
Multivariate Interpolation



$$F : \mathcal{X} \rightarrow \mathcal{Y}$$

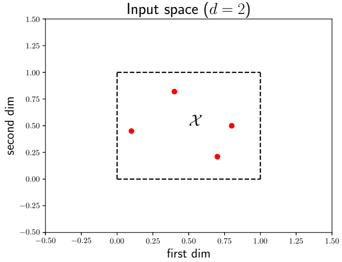


Multivariate Interpolation

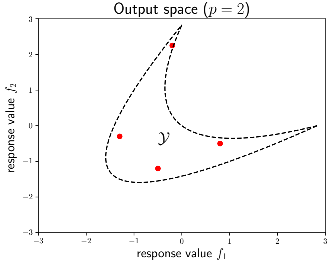


Given a set of n points $\mathcal{P}$ in $\mathcal{X}$, find $\hat{F} \approx F$ such that $\hat{F}(x) = F(x)$ for all $x \in \mathcal{P}$

Multivariate Interpolation



$$F : \mathcal{X} \to \mathcal{Y}$$

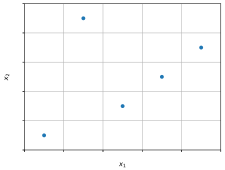


Given a set of n points $\mathcal{P}$ in $\mathcal{X}$, find $\hat{F} \approx F$ such that $\hat{F}(x) = F(x)$ for all $x \in \mathcal{P}$

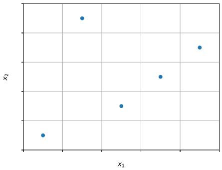
Suppose $\mathcal{X} \subset \mathbb{R}^d$ and $\mathcal{Y} \subset \mathbb{R}^p$, and F is continuous

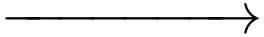
My Motivation: Multiobjective RSM

My Motivation: Multiobjective RSM

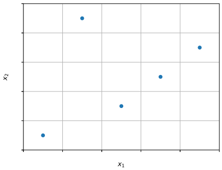


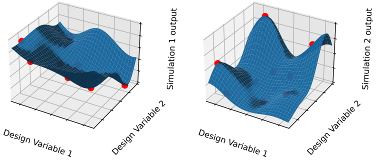
My Motivation: Multiobjective RSM



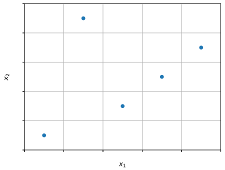


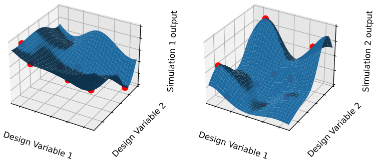
My Motivation: Multiobjective RSM

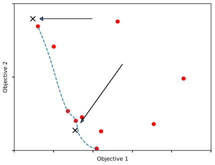




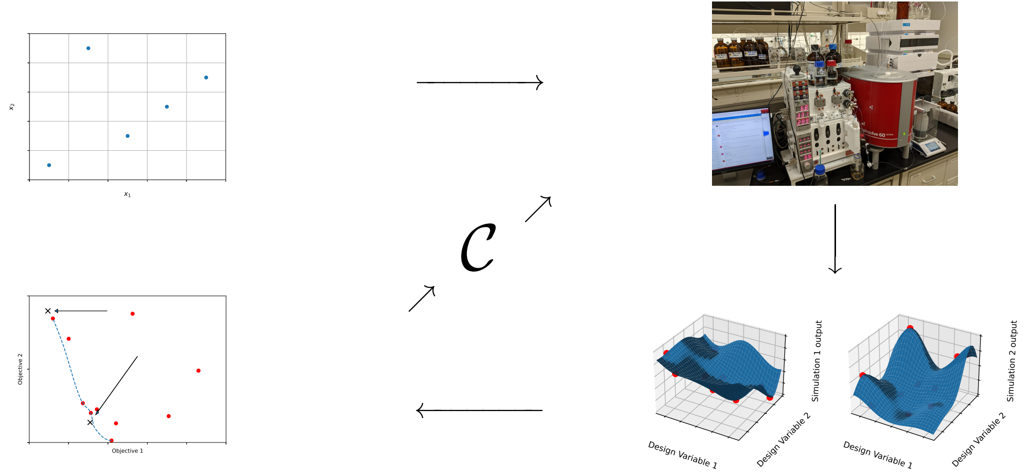
My Motivation: Multiobjective RSM



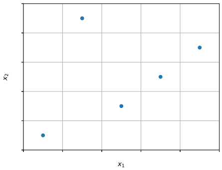




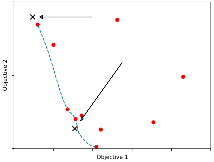
My Motivation: Multiobjective RSM

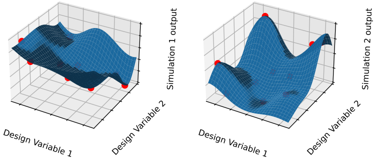


My Motivation: Multiobjective RSM



$\mathcal{C}$





https://github.com/parmoo/parmoo

Interpolation Techniques

Interpolation Techniques

- Train a neural network to 0 training error

Interpolation Techniques

- Train a neural network to 0 training error
  - For example, a fully-connected ReLU net

Interpolation Techniques

- Train a neural network to 0 training error
  - For example, a fully-connected ReLU net
- Other statistics/machine learning models

Interpolation Techniques

- Train a neural network to 0 training error
  - For example, a fully-connected ReLU net
- Other statistics/machine learning models
  - Gaussian processes
  - Support vector regressor
  - Decision tree-based methods

Interpolation Techniques

- Train a neural network to 0 training error
  - For example, a fully-connected ReLU net
- Other statistics/machine learning models
  - Gaussian processes
  - Support vector regressor
  - Decision tree-based methods
- "Classical" interpolation techniques

Interpolation Techniques

- Train a neural network to 0 training error
  - For example, a fully-connected ReLU net
- Other statistics/machine learning models
  - Gaussian processes
  - Support vector regressor
  - Decision tree-based methods
- "Classical" interpolation techniques
  - Polynomial interpolation
  - B-spline interpolation
  - RBF interpolants
  - generalized Shepard's methods
  - **Piecewise linear interpolation**

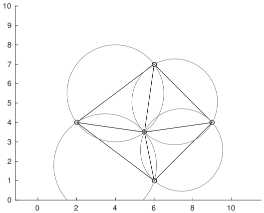
My Story with Interpolation Methods

Piecewise Linear Interpolation

- Let $\mathcal{T}(\mathcal{P})$ be a d-dimensional triangulation of $\mathcal{P}$.
- Given an interpolation point $q \in \mathcal{CH}(\mathcal{P})$, let $\mathcal{S}$ be a simplex in $\mathcal{T}(\mathcal{P})$ with vertices $s_1, \ldots, s_{d+1}$ such that $q \in \mathcal{S}$.
- Then there exist unique *convex weights* $w_1, \ldots, w_{d+1}$ such that $q = \sum_{i=1}^{d+1} w_i s_i$.

Piecewise Linear Interpolation

- Let $\mathcal{T}(\mathcal{P})$ be a d-dimensional triangulation of $\mathcal{P}$.
- Given an interpolation point $q \in \mathcal{CH}(\mathcal{P})$, let $\mathcal{S}$ be a simplex in $\mathcal{T}(\mathcal{P})$ with vertices $s_1, \ldots, s_{d+1}$ such that $q \in \mathcal{S}$.
- Then there exist unique *convex weights* $w_1, \ldots, w_{d+1}$ such that $q = \sum_{i=1}^{d+1} w_i s_i$.

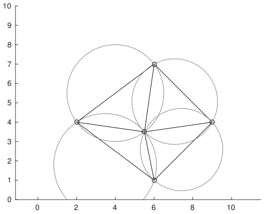


Define:

$$\hat{F}_{\mathcal{T}}(q) = F(s_1)w_1 + F(s_2)w_2 + \ \ldots \ + F(s_{d+1})w_{d+1}$$

Piecewise Linear Interpolation

- Let $\mathcal{T}(\mathcal{P})$ be a d-dimensional triangulation of $\mathcal{P}$.
- Given an interpolation point $q \in \mathcal{CH}(\mathcal{P})$, let $\mathcal{S}$ be a simplex in $\mathcal{T}(\mathcal{P})$ with vertices $s_1, \ldots, s_{d+1}$ such that $q \in \mathcal{S}$.
- Then there exist unique *convex weights* $w_1, \ldots, w_{d+1}$ such that $q = \sum_{i=1}^{d+1} w_i s_i$.



Define:

$$\hat{F}_{\mathcal{T}}(q) = F(s_1)w_1 + F(s_2)w_2 + \ \ldots \ + F(s_{d+1})w_{d+1}$$

Chang *et al. 2020. Algorithm 1012: DELAUNAYSPARSE. ACM TOMS 46(4), Article No. 38.*

Error Rates for Piecewise Linear Interpolants

For an individual component function F_i:

- Let ∇F_i be λ-Lipschitz in the 2-norm
- For $\mathcal{S} \in \mathcal{T}$ containing q,
  - let ξ be the diameter of $\mathcal{S}$ and
  - $\kappa_{\mathcal{S}}$ be the condition number of $\mathcal{S}$'s barycentric transformation matrix

Error Rates for Piecewise Linear Interpolants

For an individual component function F_i:

- Let ∇F_i be λ-Lipschitz in the 2-norm
- For $\mathcal{S} \in \mathcal{T}$ containing q,
  - let ξ be the diameter of $\mathcal{S}$ and
  - $\kappa_{\mathcal{S}}$ be the condition number of $\mathcal{S}$'s barycentric transformation matrix

Then

$$|F_i(q) - \hat{F}_{\mathcal{T}}(q)| \leq \frac{\lambda}{2}(1 + \sqrt{d}\kappa_{\mathcal{S}})\xi^2$$

Error Rates for Piecewise Linear Interpolants

For an individual component function F_i:

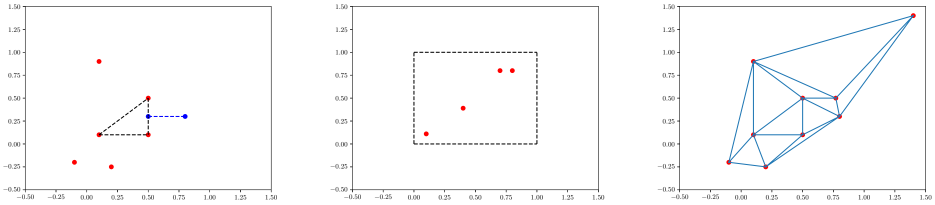
- Let ∇F_i be λ-Lipschitz in the 2-norm
- For $\mathcal{S} \in \mathcal{T}$ containing q,
  - let ξ be the diameter of $\mathcal{S}$ and
  - $\kappa_{\mathcal{S}}$ be the condition number of $\mathcal{S}$'s barycentric transformation matrix

Then

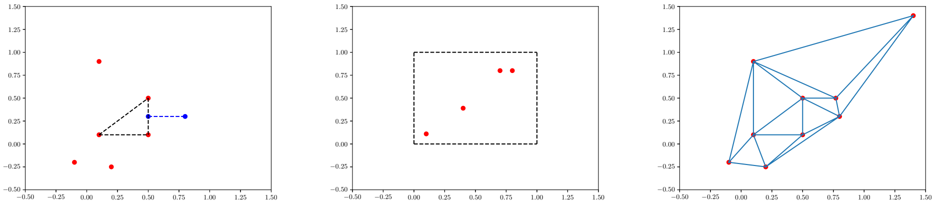
$$|F_i(q) - \hat{F}_{\mathcal{T}}(q)| \leq \frac{\lambda}{2}(1 + \sqrt{d}\kappa_{\mathcal{S}})\xi^2$$

Lux, Watson, ***Chang****, et al. 2021. Interpolation of sparse high-dimensional data. Numerical Algorithms 88(1), 281–313.*

Issues Solving Real-World Problems

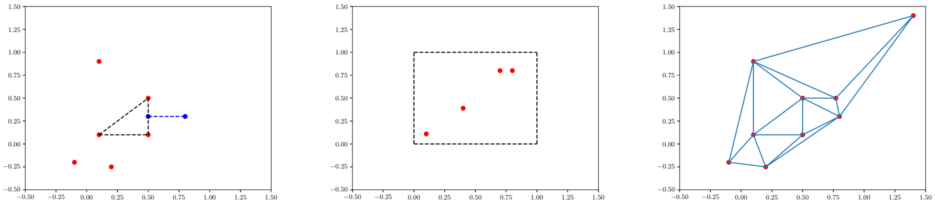


Issues Solving Real-World Problems



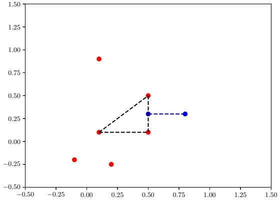
- Too many extrapolation points (information lost in projection)

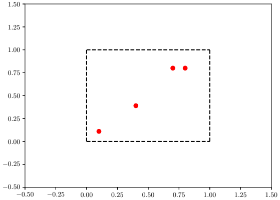
Issues Solving Real-World Problems

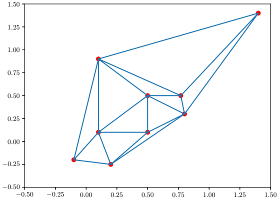


- Too many extrapolation points (information lost in projection)
- “Flat” data set – cannot (accurately) triangulate

Issues Solving Real-World Problems







- Too many extrapolation points (information lost in projection)
- “Flat” data set – cannot (accurately) triangulate
- Several massive simplices (high error-rate, poor conditioning)

What does the theory say?

What does the theory say?

Gorban et al. 2017. Stochastic separation theorems. Neural Networks 94, 255-259.

What does the theory say?

Gorban et al. 2017. Stochastic separation theorems. Neural Networks 94, 255-259.

My take:
Let $\mathcal{P}$ be randomly sampled by a distribution μ, then

What does the theory say?

Gorban et al. 2017. Stochastic separation theorems. Neural Networks 94, 255-259.

My take:
Let $\mathcal{P}$ be randomly sampled by a distribution μ, then

$$\mathbb{E}\left[\text{vol}(\mathcal{CH}(\mathcal{P}))\right] \to 0 \text{ as } d \text{ increase}$$

What does the theory say?

Gorban et al. 2017. Stochastic separation theorems. Neural Networks 94, 255-259.

My take:
Let $\mathcal{P}$ be randomly sampled by a distribution μ, then

$$\mathbb{E}\left[\mathrm{vol}(\mathcal{CH}(\mathcal{P}))\right] \to 0 \text{ as } d \text{ increase} \quad \text{(exponentially)}$$

But does it happen in practice?

But does it happen in practice?

Yousefzadeh 2020. Deep learning generalization and the convex hull of training sets. Deep Learning through Info. Geom. Workshop, NeurIPS.

But does it happen in practice?

Yousefzadeh 2020. Deep learning generalization and the convex hull of training sets. Deep Learning through Info. Geom. Workshop, NeurIPS.

My take:
Yes, it does... and it's bad.

The Geometry of Bad Data

Error Rates for Piecewise Linear Interpolants

For an individual component function F_i:

- Let ∇F_i be λ-Lipschitz in the 2-norm
- For $\mathcal{S} \in \mathcal{T}$ containing q,
  - let ξ be the diameter of $\mathcal{S}$ and
  - $\kappa_{\mathcal{S}}$ be the condition number of $\mathcal{S}$'s barycentric transformation matrix

Then

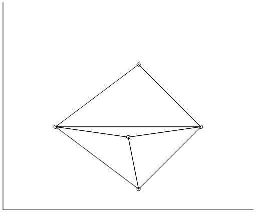
$$|F_i(q) - \hat{F}_{\mathcal{T}}(q)| \leq \frac{\lambda}{2}(1 + \sqrt{d}\kappa_{\mathcal{S}})\xi^2$$

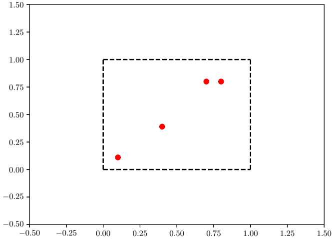
Lux, Watson, ***Chang****, et al. 2021. Interpolation of sparse high-dimensional data. Numerical Algorithms 88(1), 281–313.*

What Could Go Wrong: Poor data conditioning

What Could Go Wrong: Poor data conditioning

- Simplices are long and narrow
- $\kappa_{\mathcal{S}}$ term blows up
- $\hat{F}_{\mathcal{T}}$ is hard to compute and low accuracy
- Ex: data lies close to a lower-dimensional manifold

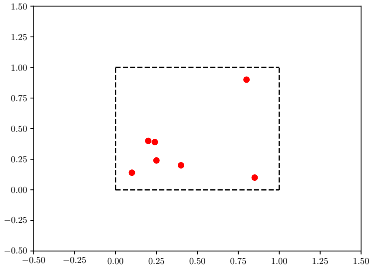




What Could Go Wrong: Data imbalance

What Could Go Wrong: Data imbalance

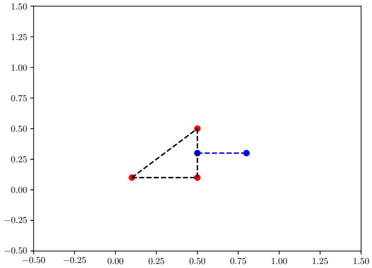
- Data accumulates in subregions of $\mathcal{X}$
- ξ stays large in less-dense regions of $\mathcal{X}$
- Can also result in poor conditioning, but they are not the same
- Called *high discepancy*



What Could Go Wrong: Extrapolation

What Could Go Wrong: Extrapolation

- $\hat{F}_{\mathcal{T}}$ is only defined in the $\mathcal{CH}(\mathcal{P})$
- Will have to project q into $\mathcal{CH}(\mathcal{P})$ and interpolate projection
- Low accuracy when residual is large
- When $\mathcal{CH}(\mathcal{P})$ is a small subset of $\mathcal{X}$, overall accuracy can be poor



Same Issues for RBFs/GPs

$$\hat{F}_{RBF} = \omega^\top \begin{bmatrix} e^{-\|x_1 - x\|^2/\sigma} \\ e^{-\|x_2 - x\|^2/\sigma} \\ \vdots \\ e^{-\|x_n - x\|^2/\sigma} \end{bmatrix}$$

Same Issues for RBFs/GPs

$$\hat{F}_{RBF} = \omega^\top \begin{bmatrix} e^{-\|x_1 - x\|^2/\sigma} \\ e^{-\|x_2 - x\|^2/\sigma} \\ \vdots \\ e^{-\|x_n - x\|^2/\sigma} \end{bmatrix}$$

where $A\omega = y$ and

$$A = \begin{bmatrix} 1 & e^{-\|x_1 - x_2\|^2/\sigma} & \dots & e^{-\|x_1 - x_n\|^2/\sigma} \\ e^{-\|x_2 - x_1\|^2/\sigma} & 1 & \dots & e^{-\|x_2 - x_n\|^2/\sigma} \\ \vdots & \vdots & & \vdots \\ e^{-\|x_n - x_1\|^2/\sigma} & e^{-\|x_n - x_2\|^2/\sigma} & \dots & 1 \end{bmatrix} \qquad y = \begin{bmatrix} F(x_1) \\ F(x_2) \\ \vdots \\ F(x_n) \end{bmatrix}$$

RBF/GP Conditioning and Accuracy

$$A = \begin{bmatrix} 1 & e^{-\|x_1 - x_2\|^2/\sigma} & \dots & e^{-\|x_1 - x_n\|^2/\sigma} \\ e^{-\|x_2 - x_1\|^2/\sigma} & 1 & \dots & e^{-\|x_2 - x_n\|^2/\sigma} \\ \vdots & \vdots & & \vdots \\ e^{-\|x_n - x_1\|^2/\sigma} & e^{-\|x_n - x_2\|^2/\sigma} & \dots & 1 \end{bmatrix}$$

RBF/GP Conditioning and Accuracy

$$A = \begin{bmatrix} 1 & e^{-\|x_1-x_2\|^2/\sigma} & \dots & e^{-\|x_1-x_n\|^2/\sigma} \\ e^{-\|x_2-x_1\|^2/\sigma} & 1 & \dots & e^{-\|x_2-x_n\|^2/\sigma} \\ \vdots & \vdots & & \vdots \\ e^{-\|x_n-x_1\|^2/\sigma} & e^{-\|x_n-x_2\|^2/\sigma} & \dots & 1 \end{bmatrix}$$

- As we seek higher accuracy, data points get closer together

RBF/GP Conditioning and Accuracy

$$A = \begin{bmatrix} 1 & e^{-\|x_1-x_2\|^2/\sigma} & \dots & e^{-\|x_1-x_n\|^2/\sigma} \\ e^{-\|x_2-x_1\|^2/\sigma} & 1 & \dots & e^{-\|x_2-x_n\|^2/\sigma} \\ \vdots & \vdots & & \vdots \\ e^{-\|x_n-x_1\|^2/\sigma} & e^{-\|x_n-x_2\|^2/\sigma} & \dots & 1 \end{bmatrix}$$

- As we seek higher accuracy, data points get closer together
- As $\|x_i - x_j\| \to 0$, $A \to$ singularity

RBF/GP Conditioning and Accuracy

$$A = \begin{bmatrix} 1 & e^{-\|x_1 - x_2\|^2/\sigma} & \dots & e^{-\|x_1 - x_n\|^2/\sigma} \\ e^{-\|x_2 - x_1\|^2/\sigma} & 1 & \dots & e^{-\|x_2 - x_n\|^2/\sigma} \\ \vdots & \vdots & & \vdots \\ e^{-\|x_n - x_1\|^2/\sigma} & e^{-\|x_n - x_2\|^2/\sigma} & \dots & 1 \end{bmatrix}$$

- As we seek higher accuracy, data points get closer together
- As $\|x_i - x_j\| \to 0$, $A \to$ singularity
- Decrease the shape parameter σ to keep A nonsingular

RBF/GP Conditioning and Accuracy

$$A = \begin{bmatrix} 1 & e^{-\|x_1 - x_2\|^2/\sigma} & \dots & e^{-\|x_1 - x_n\|^2/\sigma} \\ e^{-\|x_2 - x_1\|^2/\sigma} & 1 & \dots & e^{-\|x_2 - x_n\|^2/\sigma} \\ \vdots & \vdots & & \vdots \\ e^{-\|x_n - x_1\|^2/\sigma} & e^{-\|x_n - x_2\|^2/\sigma} & \dots & 1 \end{bmatrix}$$

- As we seek higher accuracy, data points get closer together
- As $\|x_i - x_j\| \to 0$, $A \to$ singularity
- Decrease the shape parameter σ to keep A nonsingular
- Descreasing σ restricts the support of $\hat{F}_{RBF}$

RBF/GP Conditioning and Accuracy

$$A = \begin{bmatrix} 1 & e^{-\|x_1 - x_2\|^2/\sigma} & \dots & e^{-\|x_1 - x_n\|^2/\sigma} \\ e^{-\|x_2 - x_1\|^2/\sigma} & 1 & \dots & e^{-\|x_2 - x_n\|^2/\sigma} \\ \vdots & \vdots & & \vdots \\ e^{-\|x_n - x_1\|^2/\sigma} & e^{-\|x_n - x_2\|^2/\sigma} & \dots & 1 \end{bmatrix}$$

- As we seek higher accuracy, data points get closer together
- As $\|x_i - x_j\| \to 0$, $A \to$ singularity
- Decrease the shape parameter σ to keep A nonsingular
- Descreasing σ restricts the support of $\hat{F}_{RBF}$
  - Inability to extrapolate outside $\mathcal{CH}(\mathcal{P})$
  - When data is imabalanced, accuracy will *decrease* in low density regions

RBF/GP Conditioning and Accuracy

$$A = \begin{bmatrix} 1 & e^{-\|x_1-x_2\|^2/\sigma} & \dots & e^{-\|x_1-x_n\|^2/\sigma} \\ e^{-\|x_2-x_1\|^2/\sigma} & 1 & \dots & e^{-\|x_2-x_n\|^2/\sigma} \\ \vdots & \vdots & & \vdots \\ e^{-\|x_n-x_1\|^2/\sigma} & e^{-\|x_n-x_2\|^2/\sigma} & \dots & 1 \end{bmatrix}$$

- As we seek higher accuracy, data points get closer together
- As $\|x_i - x_j\| \to 0$, $A \to$ singularity
- Decrease the shape parameter σ to keep A nonsingular
- Descreasing σ restricts the support of $\hat{F}_{RBF}$
  - Inability to extrapolate outside $\mathcal{CH}(\mathcal{P})$
  - When data is imabalanced, accuracy will *decrease* in low density regions
- "Uncertainty principle" – for real-world datasets, cannot have accuracy and solvability

More general with Fully Linear Models

More general with Fully Linear Models

A fully linear model is "accurate enough" to use as a gradient oracle in a δ-ball around x_0

More general with Fully Linear Models

A fully linear model is "accurate enough" to use as a gradient oracle in a δ-ball around x_0

$$|\hat{F}(x) - F(x)| \leq C_1\delta^2$$
$$\|\nabla\hat{F}(x) - \nabla F(x)\| \leq C_2\delta$$

for all $\|x - x_0\| < \delta$.

More general with Fully Linear Models

A fully linear model is "accurate enough" to use as a gradient oracle in a δ-ball around x_0

$$
\begin{aligned}
|\hat{F}(x) - F(x)| &\leq C_1\delta^2 \\
\|\nabla\hat{F}(x) - \nabla F(x)\| &\leq C_2\delta
\end{aligned}
$$

for all $\|x - x_0\| < \delta$.

- When $\hat{F}$ interpolates, conditions for fully linearity reduce to geometric conditions
  - $d + 1$ model points in ball are affinely independent
- Only accurate within ball
  - no guarantees during extrapolation
  - no convergence in low-density regions

Summary of Bad Data

- Small convex hull
- Imbalanced
- Poorly conditioned

Summary of Bad Data

- Small convex hull
- Imbalanced
- Poorly conditioned

Real-world datasets have zero-volume in high-dimensions, which leads to all of the properties

A Proposed Solution

Design of Experiments

Design of Experiments

- Classical deterministic designs – full-factorial, central composite, box-behnkin
    - Typically don't scale well to many dimensions

Design of Experiments

- Classical deterministic designs – full-factorial, central composite, box-behnkin
  - Typically don't scale well to many dimensions
- Optimal design of experiments – A-optimal, E-optimal, D-optimal designs
  - Maximize info gain, which is roughly equivalent to max. model conditioning
  - Expensive to calculate
  - Typically require a model *a priori*

Design of Experiments

- Classical deterministic designs – full-factorial, central composite, box-behnkin
  - Typically don't scale well to many dimensions
- Optimal design of experiments – A-optimal, E-optimal, D-optimal designs
  - Maximize info gain, which is roughly equivalent to max. model conditioning
  - Expensive to calculate
  - Typically require a model *a priori*
- Geometric criteria – maximin and minimax
  - Heuristically good for maximizing convex hulls
  - Expensive to calculate in high dimensions

Design of Experiments

- Classical deterministic designs – full-factorial, central composite, box-behnkin
  - Typically don't scale well to many dimensions
- Optimal design of experiments – A-optimal, E-optimal, D-optimal designs
  - Maximize info gain, which is roughly equivalent to max. model conditioning
  - Expensive to calculate
  - Typically require a model *a priori*
- Geometric criteria – maximin and minimax
  - Heuristically good for maximizing convex hulls
  - Expensive to calculate in high dimensions
- Low discrepancy sequences – Sobol, Halton, etc.
  - Produce balanced datasets
  - Performance falls off when sample size is not chosen carefully

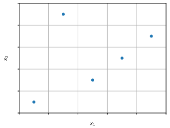
Design of Experiments

- Classical deterministic designs – full-factorial, central composite, box-behnkin
  - Typically don't scale well to many dimensions
- Optimal design of experiments – A-optimal, E-optimal, D-optimal designs
  - Maximize info gain, which is roughly equivalent to max. model conditioning
  - Expensive to calculate
  - Typically require a model *a priori*
- Geometric criteria – maximin and minimax
  - Heuristically good for maximizing convex hulls
  - Expensive to calculate in high dimensions
- Low discrepancy sequences – Sobol, Halton, etc.
  - Produce balanced datasets
  - Performance falls off when sample size is not chosen carefully
- Latin hypercubes
  - Stratifies sample – good for linear models
  - No theory for nonlinear models
  - Heuristically good in practice and cheap to calculate

Stochastic Fourier SFD

- Desired sample size
- Performance criertia

$\longrightarrow$



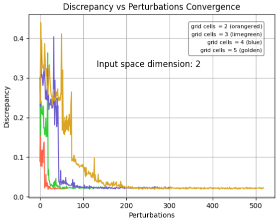
$\downarrow$

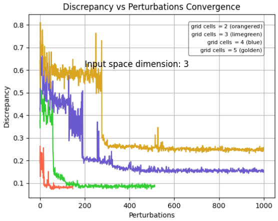
$$\alpha_1 e^{2\pi i x} + \alpha_2 e^{2\pi i (2x)} + \dots$$

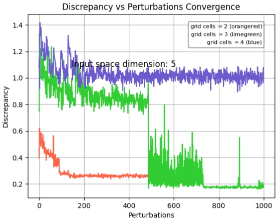
$\longleftarrow$

`COBYLA`

Preliminary Results







Resources

E-mail: tchang@anl.gov

Code: github.com/thchang/sf-sfd

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, SciDAC program under contract number DE-AC02-06CH11357.