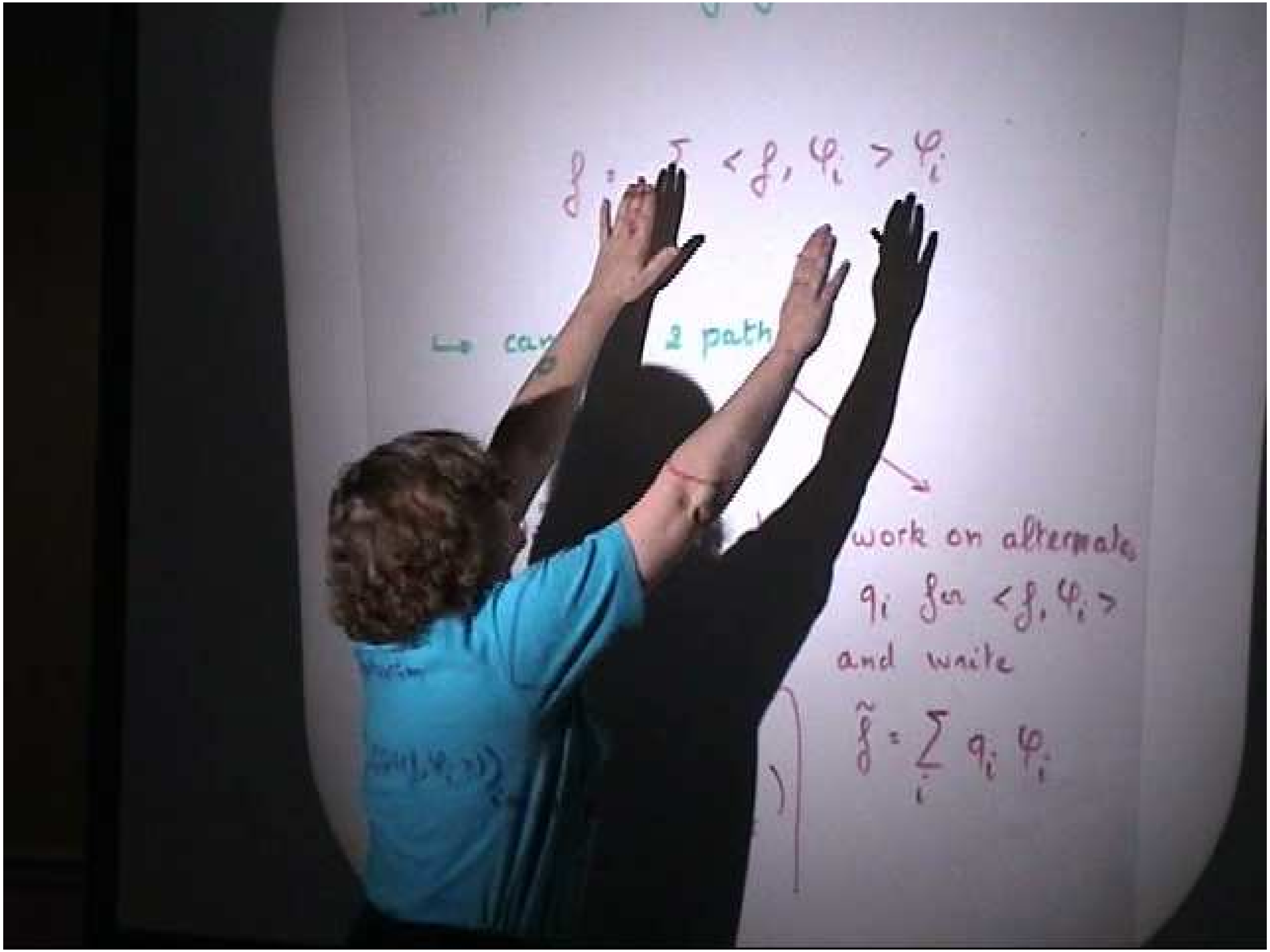
Greedy Algorithms for Manifold Recovery

Ronald DeVore

Texas A& M University

Some Personal Reflections

Ingrid Paying Homage to Her Work



Ingrid Under Arrest

Ingrid Celebrating Her Release

Ingrid Still Celebrating

What a Party Animal

Ingrid Auditioning

Ingrid Belting out Another Tune

Ingrid Getting Warmed Up

Ingrid Stealing the Show

Ingrid Grand Finale

Ingrid Biorthogonal

We got it Albert!

Subject of Talk

- I will introduce a Greedy Algorithm for approximating a compact manifold $\mathcal{K}$ in any Banach space X by selecting well chosen snapshots of the manifold

Subject of Talk

- I will introduce a Greedy Algorithm for approximating a compact manifold $\mathcal{K}$ in any Banach space X by selecting well chosen snapshots of the manifold
  - This algorithm originated in generating Reduced Bases for model reduction in Parametric and Stochastic PDEs

Subject of Talk

- I will introduce a Greedy Algorithm for approximating a compact manifold $\mathcal{K}$ in any Banach space X by selecting well chosen snapshots of the manifold
  - This algorithm originated in generating Reduced Bases for model reduction in Parametric and Stochastic PDEs
  - It applies to manifolds depending on a large (even an infinite) number of parameters

Subject of Talk

- I will introduce a Greedy Algorithm for approximating a compact manifold $\mathcal{K}$ in any Banach space X by selecting well chosen snapshots of the manifold
  - This algorithm originated in generating Reduced Bases for model reduction in Parametric and Stochastic PDEs
  - It applies to manifolds depending on a large (even an infinite) number of parameters
  - Typically greedy algorithms do not perform well

Subject of Talk

- I will introduce a Greedy Algorithm for approximating a compact manifold $\mathcal{K}$ in any Banach space X by selecting well chosen snapshots of the manifold
  - This algorithm originated in generating Reduced Bases for model reduction in Parametric and Stochastic PDEs
  - It applies to manifolds depending on a large (even an infinite) number of parameters
  - Typically greedy algorithms do not perform well
  - We shall try to understand if this is the case for this algorithm

Reduced Basis

- Example for parametric elliptic problems

$$-\mathrm{div}(a\nabla_x u_a) = f(x), \quad x \in D, a \in \mathcal{A}$$
$$u_a(x,y) = 0, \quad x \in \partial D, a \in \mathcal{A}$$

Reduced Basis

- Example for parametric elliptic problems

$$-\mathrm{div}(a\nabla_x u_a) = f(x), \quad x \in D, a \in \mathcal{A}$$
$$u_a(x,y) = 0, \quad x \in \partial D, a \in \mathcal{A}$$

- We want to capture the solution manifold
$\mathcal{U}_{\mathcal{A}} := \{u_a : a \in \mathcal{A}\} \subset H_0^1(D)$

Reduced Basis

- Example for parametric elliptic problems

$$-\mathrm{div}(a\nabla_x u_a) = f(x), \quad x \in D, a \in \mathcal{A}$$
$$u_a(x,y) = 0, \quad x \in \partial D, a \in \mathcal{A}$$

- We want to capture the solution manifold $\mathcal{U}_{\mathcal{A}} := \{u_a : a \in \mathcal{A}\} \subset H_0^1(D)$
  - A typical case is the affine model

$$a(x,y) = a_0(x) + \sum_{i=1}^{\infty} y_i \psi_i(x), \quad |y_i| \leq 1, \quad i = 1, 2, \ldots$$

Reduced Basis

- Example for parametric elliptic problems

$$-\mathrm{div}(a\nabla_x u_a) = f(x), \quad x \in D, a \in \mathcal{A}$$
$$u_a(x,y) = 0, \quad x \in \partial D, a \in \mathcal{A}$$

- We want to capture the solution manifold
  $\mathcal{U}_\mathcal{A} := \{u_a : a \in \mathcal{A}\} \subset H_0^1(D)$
  - A typical case is the affine model

    $$a(x,y) = a_0(x) + \sum_{i=1}^\infty y_i \psi_i(x), \quad |y_i| \le 1, \quad i = 1,2,\ldots$$

  - Possibly infinite number of parameters
    $y = (y_1, y_2, \ldots)$

Reduced Basis

- Example for parametric elliptic problems

$$-\mathrm{div}(a\nabla_x u_a) = f(x), \quad x \in D, a \in \mathcal{A}$$
$$u_a(x,y) = 0, \quad x \in \partial D, a \in \mathcal{A}$$

- We want to capture the solution manifold $\mathcal{U}_{\mathcal{A}} := \{u_a : a \in \mathcal{A}\} \subset H_0^1(D)$
  - A typical case is the affine model

$$a(x,y) = a_0(x) + \sum_{i=1}^{\infty} y_i \psi_i(x), \quad |y_i| \le 1, \quad i = 1, 2, \ldots$$

  - Possibly infinite number of parameters $y = (y_1, y_2, \ldots)$
  - Smoothness determined by how fast $\|\psi_j\|_{L_\infty} \to 0$

Reduced Basis

- Example for parametric elliptic problems

$$-\mathrm{div}(a\nabla_x u_a) = f(x), \quad x \in D, a \in \mathcal{A}$$
$$u_a(x,y) = 0, \quad x \in \partial D, a \in \mathcal{A}$$

- We want to capture the solution manifold $\mathcal{U}_{\mathcal{A}} := \{u_a : a \in \mathcal{A}\} \subset H_0^1(D)$
    - A typical case is the affine model

$$a(x,y) = a_0(x) + \sum_{i=1}^{\infty} y_i \psi_i(x), \quad |y_i| \leq 1, \quad i = 1, 2, \ldots$$

    - Possibly infinite number of parameters $y = (y_1, y_2, \ldots)$
    - Smoothness determined by how fast $\|\psi_j\|_{L_\infty} \to 0$
- Choose snapshots $u_{a_1}, \ldots, u_{a_n}$ so that $X_n := \mathrm{span}\{u_{a_1}, \ldots, u_{a_n}\}$ is a good Galerkin space

Reduced Basis

- Example for parametric elliptic problems

$$-\mathrm{div}(a\nabla_x u_a) = f(x), \quad x \in D, a \in \mathcal{A}$$
$$u_a(x,y) = 0, \quad x \in \partial D, a \in \mathcal{A}$$

- We want to capture the solution manifold $\mathcal{U}_\mathcal{A} := \{u_a : a \in \mathcal{A}\} \subset H_0^1(D)$
  - A typical case is the affine model

$$a(x,y) = a_0(x) + \sum_{i=1}^\infty y_i\psi_i(x), \quad |y_i| \le 1, \quad i = 1,2,\dots$$

  - Possibly infinite number of parameters $y = (y_1, y_2, \dots)$
  - Smoothness determined by how fast $\|\psi_j\|_{L_\infty} \to 0$
- Choose snapshots $u_{a_1}, \dots, u_{a_n}$ so that $X_n := \mathrm{span}\{u_{a_1}, \dots, u_{a_n}\}$ is a good Galerkin space
  - Given a query $a \in \mathcal{A}$, fast on line solution by the Galerkin projection $P_n u_a$ onto X_n

The Approximation Problem

- All action will take place in a Banach space X

The Approximation Problem

- All action will take place in a Banach space X
- We are given a compact set of functions $\mathcal{K}$, viewed to be a manifold, which we wish to approximate by a linear space (linear manifold)

The Approximation Problem

- All action will take place in a Banach space X
- We are given a compact set of functions $\mathcal{K}$, viewed to be a manifold, which we wish to approximate by a linear space (linear manifold)
- We look for a finite number of functions $f_0, \ldots, f_{N-1} \in \mathcal{K}$ such that for $V_N := \mathrm{span}\{f_0, \ldots, f_{N-1}\}$, we have

$$\mathrm{dist}(f, V_N)_X \leq \epsilon, \quad \forall f \in \mathcal{K}$$

The Approximation Problem

- All action will take place in a Banach space X
- We are given a compact set of functions $\mathcal{K}$, viewed to be a manifold, which we wish to approximate by a linear space (linear manifold)
- We look for a finite number of functions $f_0, \ldots, f_{N-1} \in \mathcal{K}$ such that for $V_N := \mathrm{span}\{f_0, \ldots, f_{N-1}\}$, we have

$$\mathrm{dist}(f, V_N)_X \leq \epsilon, \quad \forall f \in \mathcal{K}$$

- Of course for the given tolerance ϵ we want N to be as small as possible

The (Pure) Greedy Algorithm

- $f_0 := \operatorname{argmax}\{\|f\|_X : f \in \mathcal{K}\}$

The (Pure) Greedy Algorithm

- $f_0 := \mathrm{argmax}\{\|f\|_X : f \in \mathcal{K}\}$
- If $f_0, \dots, f_{n-1}$ have been chosen, define

The (Pure) Greedy Algorithm

- $f_0 := \operatorname{argmax}\{\|f\|_X : f \in \mathcal{K}\}$
- If $f_0, \dots, f_{n-1}$ have been chosen, define
  - $V_n := \operatorname{span}\{f_0, \dots, f_{n-1}\}$

The (Pure) Greedy Algorithm

- $f_0 := \operatorname{argmax}\{\|f\|_X : f \in \mathcal{K}\}$
- If $f_0, \ldots, f_{n-1}$ have been chosen, define
  - $V_n := \operatorname{span}\{f_0, \ldots, f_{n-1}\}$
  - $f_n := \operatorname*{Argmax}_{f \in \mathcal{K}} \operatorname{dist}(f, V_n)_X$

The (Pure) Greedy Algorithm

- $f_0 := \operatorname{argmax}\{\|f\|_X : f \in \mathcal{K}\}$
- If $f_0, \ldots, f_{n-1}$ have been chosen, define
  - $V_n := \operatorname{span}\{f_0, \ldots, f_{n-1}\}$
  - $f_n := \operatorname*{Argmax}_{f \in \mathcal{K}} \operatorname{dist}(f, V_n)_X$
- Thus at each step, the function f_n is chosen in a greedy manner

The (Pure) Greedy Algorithm

- $f_0 := \operatorname{argmax}\{\|f\|_X : f \in \mathcal{K}\}$
- If $f_0, \dots, f_{n-1}$ have been chosen, define
  - $V_n := \operatorname{span}\{f_0, \dots, f_{n-1}\}$
  - $f_n := \operatorname*{Argmax}_{f \in \mathcal{K}} \operatorname{dist}(f, V_n)_X$
- Thus at each step, the function f_n is chosen in a greedy manner
- Our experience tells us that such greedy strategies are usually not very good

The (Pure) Greedy Algorithm

- $f_0 := \operatorname{argmax}\{\|f\|_X : f \in \mathcal{K}\}$
- If $f_0, \dots, f_{n-1}$ have been chosen, define
  - $V_n := \operatorname{span}\{f_0, \dots, f_{n-1}\}$
  - $f_n := \operatorname*{Argmax}_{f \in \mathcal{K}} \operatorname{dist}(f, V_n)_X$
- Thus at each step, the function f_n is chosen in a greedy manner
- Our experience tells us that such greedy strategies are usually not very good
- But before going into an analysis of this algorithm let us point out that as it stands, the algorithm is too idealized for applications and therefore we consider the following weak version of this algorithm

Weak Greedy Algorithms

- Let $0 < \gamma \leq 1$

Weak Greedy Algorithms

- Let $0 < \gamma \leq 1$
- The choice $\gamma = 1$ gives the ideal greedy algorithm just described

Weak Greedy Algorithms

- Let $0 < \gamma \leq 1$
- The choice $\gamma = 1$ gives the ideal greedy algorithm just described
- The weak greedy algorithm does the following

Weak Greedy Algorithms

- Let $0 < \gamma \le 1$
- The choice $\gamma = 1$ gives the ideal greedy algorithm just described
- The weak greedy algorithm does the following
  - Choose f_0 such that $\|f_0\|_X \ge \gamma \max\{\|f\|_X : f \in \mathcal{K}\}$

Weak Greedy Algorithms

- Let $0 < \gamma \leq 1$
- The choice $\gamma = 1$ gives the ideal greedy algorithm just described
- The weak greedy algorithm does the following
  - Choose f_0 such that $\|f_0\|_X \geq \gamma \max\{\|f\|_X : f \in \mathcal{K}\}$
  - If $f_0, \ldots, f_{n-1}$ have been chosen, define $V_n := \operatorname{span}\{f_0, \ldots, f_{n-1}\}$

Weak Greedy Algorithms

- Let $0 < \gamma \leq 1$
- The choice $\gamma = 1$ gives the ideal greedy algorithm just described
- The weak greedy algorithm does the following
  - Choose f_0 such that $\|f_0\|_X \geq \gamma \max\{\|f\|_X : f \in \mathcal{K}\}$
  - If $f_0, \dots, f_{n-1}$ have been chosen, define $V_n := \operatorname{span}\{f_0, \dots, f_{n-1}\}$
  - Choose f_n so that

$$\operatorname{dist}(f, V_n)_X \geq \gamma \max_{f \in \mathcal{K}} \operatorname{dist}(f, V_n)_X = \sigma_n$$

Weak Greedy Algorithms

- Let $0 < \gamma \leq 1$
- The choice $\gamma = 1$ gives the ideal greedy algorithm just described
- The weak greedy algorithm does the following
  - Choose f_0 such that $\|f_0\|_X \geq \gamma \max\{\|f\|_X : f \in \mathcal{K}\}$
  - If $f_0, \dots, f_{n-1}$ have been chosen, define $V_n := \mathrm{span}\{f_0, \dots, f_{n-1}\}$
  - Choose f_n so that

$$\mathrm{dist}(f, V_n)_X \geq \gamma \max_{f \in \mathcal{K}} \mathrm{dist}(f, V_n)_X = \sigma_n$$

- Notice that the sequence $f_0, f_1, \dots$ is not unique

Weak Greedy Algorithms

- Let $0 < \gamma \leq 1$
- The choice $\gamma = 1$ gives the ideal greedy algorithm just described
- The weak greedy algorithm does the following
  - Choose f_0 such that $\|f_0\|_X \geq \gamma \max\{\|f\|_X : f \in \mathcal{K}\}$
  - If $f_0, \dots, f_{n-1}$ have been chosen, define $V_n := \operatorname{span}\{f_0, \dots, f_{n-1}\}$
  - Choose f_n so that

$$\operatorname{dist}(f, V_n)_X \geq \gamma \max_{f \in \mathcal{K}} \operatorname{dist}(f, V_n)_X = \sigma_n$$

- Notice that the sequence $f_0, f_1, \dots$ is not unique
- The discussion and analysis below applies to any realization of this algorithm

Performance of Greedy Algorithms

- How should we measure performance of (weak) Greedy Algorithm: compare it with best n dimensional space

Performance of Greedy Algorithms

- How should we measure performance of (weak) Greedy Algorithm: compare it with best n dimensional space
- Kolmogorov width: $d_n(\mathcal{K})_X := \inf_{\dim(V)=n} \sup_{f \in \mathcal{K}} \operatorname{dist}(f, V)_X$

Performance of Greedy Algorithms

- How should we measure performance of (weak) Greedy Algorithm: compare it with best n dimensional space
- Kolmogorov width: $d_n(\mathcal{K})_X := \inf_{\dim(V)=n} \sup_{f\in\mathcal{K}} \operatorname{dist}(f,V)_X$
- This is a benchmark for performance

Performance of Greedy Algorithms

- How should we measure performance of (weak) Greedy Algorithm: compare it with best n dimensional space
- Kolmogorov width: $d_n(\mathcal{K})_X := \inf_{\dim(V)=n} \sup_{f\in\mathcal{K}} \operatorname{dist}(f,V)_X$
- This is a benchmark for performance
- Let $V_n := \operatorname{span}\{f_0,\dots,f_{n-1}\}$ be the subspace generated by the weak greedy selection

Performance of Greedy Algorithms

- How should we measure performance of (weak) Greedy Algorithm: compare it with best n dimensional space
- Kolmogorov width: $d_n(\mathcal{K})_X := \inf_{\dim(V)=n} \sup_{f\in\mathcal{K}} \operatorname{dist}(f,V)_X$
- This is a benchmark for performance
- Let $V_n := \operatorname{span}\{f_0,\dots,f_{n-1}\}$ be the subspace generated by the weak greedy selection
- $\sigma_n(\mathcal{K})_X := \sup_{f\in\mathcal{K}} \operatorname{dist}(f,V_n)_X$ algorithm performance

Performance of Greedy Algorithms

- How should we measure performance of (weak) Greedy Algorithm: compare it with best n dimensional space
- Kolmogorov width: $d_n(\mathcal{K})_X := \inf_{\dim(V)=n} \sup_{f \in \mathcal{K}} \operatorname{dist}(f, V)_X$
- This is a benchmark for performance
- Let $V_n := \operatorname{span}\{f_0, \ldots, f_{n-1}\}$ be the subspace generated by the weak greedy selection
- $\sigma_n(\mathcal{K})_X := \sup_{f \in \mathcal{K}} \operatorname{dist}(f, V_n)_X$ algorithm performance
- Buffa, Maday, Patera, Prud'home, Turinici (BMPPT) for Hilbert space $\mathcal{H}$

Performance of Greedy Algorithms

- How should we measure performance of (weak) Greedy Algorithm: compare it with best n dimensional space
- Kolmogorov width: $d_n(\mathcal{K})_X := \inf_{\dim(V)=n} \sup_{f \in \mathcal{K}} \operatorname{dist}(f, V)_X$
- This is a benchmark for performance
- Let $V_n := \operatorname{span}\{f_0, \dots, f_{n-1}\}$ be the subspace generated by the weak greedy selection
- $\sigma_n(\mathcal{K})_X := \sup_{f \in \mathcal{K}} \operatorname{dist}(f, V_n)_X$ algorithm performance
- Buffa, Maday, Patera, Prud'home, Turinici (BMPPT) for Hilbert space $\mathcal{H}$
- BMPPT prove that for the ideal greedy $\sigma_n(\mathcal{K})_{\mathcal{H}} \le C n 2^n d_n(\mathcal{K})_{\mathcal{H}}$

Performance of Greedy Algorithms

- How should we measure performance of (weak) Greedy Algorithm: compare it with best n dimensional space
- Kolmogorov width: $d_n(\mathcal{K})_X := \inf_{\dim(V)=n} \sup_{f \in \mathcal{K}} \operatorname{dist}(f, V)_X$
- This is a benchmark for performance
- Let $V_n := \operatorname{span}\{f_0, \ldots, f_{n-1}\}$ be the subspace generated by the weak greedy selection
- $\sigma_n(\mathcal{K})_X := \sup_{f \in \mathcal{K}} \operatorname{dist}(f, V_n)_X$ algorithm performance
- Buffa, Maday, Patera, Prud'home, Turinici (BMPPT) for Hilbert space $\mathcal{H}$
- BMPPT prove that for the ideal greedy $\sigma_n(\mathcal{K})_{\mathcal{H}} \le C n 2^n d_n(\mathcal{K})_{\mathcal{H}}$
- Only useful if $n 2^n d_n(\mathcal{K})_{\mathcal{H}} \to 0$

Improved Convergence Results

- Binev, Cohen, Dahmen, DeVore, Petrova, Wojtaczszyk

Improved Convergence Results

- Binev, Cohen, Dahmen, DeVore, Petrova, Wojtaczszyk
- All of their results are for $X = \mathcal{H}$ a Hilbert space

Improved Convergence Results

- Binev, Cohen, Dahmen, DeVore, Petrova, Wojtaczszyk
- All of their results are for $X = \mathcal{H}$ a Hilbert space
- Their resutls apply to both greedy and weak greedy

Improved Convergence Results

- Binev, Cohen, Dahmen, DeVore, Petrova, Wojtaczszyk
- All of their results are for $X = \mathcal{H}$ a Hilbert space
- Their resutls apply to both greedy and weak greedy
- Polynomial Decay Theorem: If $d_n(\mathcal{K}) \leq Mn^{-\alpha}$, $n \geq 1$, then $\sigma_n(\mathcal{K}) \leq C_\alpha M n^{-\alpha}$, $n \geq 1$, with a fixed C_α

Improved Convergence Results

- Binev, Cohen, Dahmen, DeVore, Petrova, Wojtaczszyk
- All of their results are for $X = \mathcal{H}$ a Hilbert space
- Their resutls apply to both greedy and weak greedy
- Polynomial Decay Theorem: If $d_n(\mathcal{K}) \le Mn^{-\alpha}$, $n \ge 1$, then $\sigma_n(\mathcal{K}) \le C_\alpha M n^{-\alpha}$, $n \ge 1$, with a fixed C_α
- For polynomial decay the (weak) greedy algorithm is near optimal.

Improved Convergence Results

- Binev, Cohen, Dahmen, DeVore, Petrova, Wojtaczszyk
- All of their results are for $X = \mathcal{H}$ a Hilbert space
- Their resutls apply to both greedy and weak greedy
- Polynomial Decay Theorem: If $d_n(\mathcal{K}) \le Mn^{-\alpha}$, $n \ge 1$, then $\sigma_n(\mathcal{K}) \le C_\alpha M n^{-\alpha}$, $n \ge 1$, with a fixed C_α
- For polynomial decay the (weak) greedy algorithm is near optimal.
- Example Cohen-DeVore-Schwab: for parametric PDEs Affine model: If $(\|\psi_j\|_{L_\infty}) \in \ell_p$, $p < 1$, then $d_n(\mathcal{U}_\mathcal{A})_{H_0^1} \le C n^{1-1/p}$.

Improved Convergence Results

- Binev, Cohen, Dahmen, DeVore, Petrova, Wojtaczszyk
- All of their results are for $X = \mathcal{H}$ a Hilbert space
- Their resutls apply to both greedy and weak greedy
- Polynomial Decay Theorem: If $d_n(\mathcal{K}) \leq Mn^{-\alpha}$, $n \geq 1$, then $\sigma_n(\mathcal{K}) \leq C_\alpha Mn^{-\alpha}$, $n \geq 1$, with a fixed C_α
- For polynomial decay the (weak) greedy algorithm is near optimal.
- Example Cohen-DeVore-Schwab: for parametric PDEs Affine model: If $(\|\psi_j\|_{L_\infty}) \in \ell_p$, $p < 1$, then $d_n(\mathcal{U}_\mathcal{A})_{H_0^1} \leq Cn^{1-1/p}$.
- So Greedy Snapshots give same performance

Sub exponential rates

- BCDDPW also prove comparisons for subexponential decay

Sub exponential rates

- BCDDPW also prove comparisons for subexponential decay
- Subexponential Decay Theorem: If

$$d_n(\mathcal{K}) \le Me^{-cn^\alpha}, \quad n = 1, 2, \dots,$$

then

$$\sigma_n(\mathcal{K}) \le CMe^{-c'n^\beta} \quad n = 1, 2, \dots,$$

with a fixed C, c' and $\beta := \frac{\alpha}{\alpha+1}$

Somethings bothered me

- Despite all of the previous results, there is not a good direct comparison between (σ_n) and (d_n)

Somethings bothered me

- Despite all of the previous results, there is not a good direct comparison between (σ_n) and (d_n)
- DeVore, Petrova, Wojtaczszyk

Somethings bothered me

- Despite all of the previous results, there is not a good direct comparison between (σ_n) and (d_n)
- DeVore, Petrova, Wojtaczszyk
  - Applies to any Banach space X

Somethings bothered me

- Despite all of the previous results, there is not a good direct comparison between (σ_n) and (d_n)
- DeVore, Petrova, Wojtaczszyk
  - Applies to any Banach space X
  - Better results even for Hilbert space $\mathcal{H}$

Somethings bothered me

- Despite all of the previous results, there is not a good direct comparison between (σ_n) and (d_n)
- DeVore, Petrova, Wojtaczszyk
  - Applies to any Banach space X
  - Better results even for Hilbert space $\mathcal{H}$
  - Among other comparisons:

$$\sigma_n(\mathcal{F}) \le \sqrt{2}\gamma^{-1} \min_{1\le m<n} d_m^{\frac{n-m}{n}}$$

Somethings bothered me

- Despite all of the previous results, there is not a good direct comparison between (σ_n) and (d_n)
- DeVore, Petrova, Wojtaczszyk
  - Applies to any Banach space X
  - Better results even for Hilbert space $\mathcal{H}$
  - Among other comparisons:

$$\sigma_n(\mathcal{F}) \leq \sqrt{2}\gamma^{-1} \min_{1 \leq m < n} d_m^{\frac{n-m}{n}}$$

  - In particular $\sigma_{2n}(\mathcal{F}) \leq \sqrt{2}\gamma^{-1}\sqrt{d_n(\mathcal{F})}$, $n = 1, 2 \ldots$

Somethings bothered me

- Despite all of the previous results, there is not a good direct comparison between (σ_n) and (d_n)
- DeVore, Petrova, Wojtaczszyk
  - Applies to any Banach space X
  - Better results even for Hilbert space $\mathcal{H}$
  - Among other comparisons:

    $$\sigma_n(\mathcal{F}) \leq \sqrt{2}\gamma^{-1} \min_{1 \leq m < n} d_m^{\frac{n-m}{n}}$$

  - In particular $\sigma_{2n}(\mathcal{F}) \leq \sqrt{2}\gamma^{-1}\sqrt{d_n(\mathcal{F})}$, $n = 1, 2 \ldots$
  - From their results, one can deduce slow decay theorem and better fast decay: If $d_n(\mathcal{F}) \leq C_0 e^{-c_0 n^\alpha}$, then $\sigma_n(\mathcal{F}) \leq \sqrt{2C_0}\gamma^{-1} e^{-c_1 n^\alpha}$, $c_1 = 2^{-1-2\alpha} c_0$

Somethings bothered me

- Despite all of the previous results, there is not a good direct comparison between (σ_n) and (d_n)
- DeVore, Petrova, Wojtaczszyk
  - Applies to any Banach space X
  - Better results even for Hilbert space $\mathcal{H}$
  - Among other comparisons:

    $$\sigma_n(\mathcal{F}) \leq \sqrt{2}\gamma^{-1} \min_{1\leq m<n} d_m^{\frac{n-m}{n}}$$

  - In particular $\sigma_{2n}(\mathcal{F}) \leq \sqrt{2}\gamma^{-1}\sqrt{d_n(\mathcal{F})}$, $n = 1, 2 \ldots$
  - From their results, one can deduce slow decay theorem and better fast decay: If $d_n(\mathcal{F}) \leq C_0 e^{-c_0 n^\alpha}$, then $\sigma_n(\mathcal{F}) \leq \sqrt{2C_0}\gamma^{-1} e^{-c_1 n^\alpha}$, $c_1 = 2^{-1-2\alpha} c_0$

Banach Space Case

- DPW prove results for any Banach space X

Banach Space Case

- DPW prove results for any Banach space X
- For the weak greedy algorithm with constant γ in a Banach pace X, we have the following:

Banach Space Case

- DPW prove results for any Banach space X
- For the weak greedy algorithm with constant γ in a Banach pace X, we have the following:
  - In particular $\sigma_{2n}(\mathcal{F}) \leq \sqrt{2n}\gamma^{-1}\sqrt{d_n(\mathcal{F})}$, $\quad n = 1, 2 \ldots$

Banach Space Case

- DPW prove results for any Banach space X
- For the weak greedy algorithm with constant γ in a Banach pace X, we have the following:
  - In particular $\sigma_{2n}(\mathcal{F}) \leq \sqrt{2n}\gamma^{-1}\sqrt{d_n(\mathcal{F})}$, $\quad n = 1, 2 \ldots$
  - If $d_n(\mathcal{F}) \leq C_0 n^{-\alpha}, \;\; n = 1, 2, \ldots$, then for all $\beta > 1/2$

$$\sigma_n(\mathcal{F}) \leq C_1 n^{-\alpha+\beta}, \;\; n = 1, 2, \ldots$$

Banach Space Case

- DPW prove results for any Banach space X
- For the weak greedy algorithm with constant γ in a Banach pace X, we have the following:
  - In particular $\sigma_{2n}(\mathcal{F}) \leq \sqrt{2n}\gamma^{-1}\sqrt{d_n(\mathcal{F})}$, $\quad n = 1, 2 \ldots$
  - If $d_n(\mathcal{F}) \leq C_0 n^{-\alpha}, \ \ n = 1, 2, \ldots$, then for all $\beta > 1/2$

    $$\sigma_n(\mathcal{F}) \leq C_1 n^{-\alpha+\beta}, \ \ n = 1, 2, \ldots$$

  - If $d_n(\mathcal{F}) \leq C_0 e^{-c_0 n^\alpha}, \ \ n = 1, 2, \ldots$, then

    $$\sigma_n(\mathcal{F}) \leq \sqrt{2C_0}\gamma^{-1} e^{-c_1 n^\alpha}, \ \ n = 1, 2, \ldots,$$

Banach Space Case

- DPW prove results for any Banach space X
- For the weak greedy algorithm with constant γ in a Banach pace X, we have the following:
  - In particular $\sigma_{2n}(\mathcal{F}) \le \sqrt{2n}\gamma^{-1}\sqrt{d_n(\mathcal{F})}$, $\quad n = 1, 2 \ldots$
  - If $d_n(\mathcal{F}) \le C_0 n^{-\alpha}, \quad n = 1, 2, \ldots$, then for all $\beta > 1/2$

    $$\sigma_n(\mathcal{F}) \le C_1 n^{-\alpha+\beta}, \quad n = 1, 2, \ldots$$

  - If $d_n(\mathcal{F}) \le C_0 e^{-c_0 n^\alpha}, \quad n = 1, 2, \ldots$, then

    $$\sigma_n(\mathcal{F}) \le \sqrt{2C_0}\gamma^{-1} e^{-c_1 n^\alpha}, \quad n = 1, 2, \ldots,$$

- The loss of $\sqrt{n}$ is necessary.

Banach Space Case

- DPW prove results for any Banach space X
- For the weak greedy algorithm with constant γ in a Banach pace X, we have the following:
  - In particular $\sigma_{2n}(\mathcal{F}) \leq \sqrt{2n}\gamma^{-1}\sqrt{d_n(\mathcal{F})}, \quad n = 1, 2 \ldots$
  - If $d_n(\mathcal{F}) \leq C_0 n^{-\alpha}, \ \ n = 1, 2, \ldots$, then for all $\beta > 1/2$

    $$\sigma_n(\mathcal{F}) \leq C_1 n^{-\alpha+\beta}, \ \ n = 1, 2, \ldots$$

  - If $d_n(\mathcal{F}) \leq C_0 e^{-c_0 n^\alpha}, \ \ n = 1, 2, \ldots$, then

    $$\sigma_n(\mathcal{F}) \leq \sqrt{2C_0}\gamma^{-1} e^{-c_1 n^\alpha}, \ \ n = 1, 2, \ldots,$$

- The loss of $\sqrt{n}$ is necessary.
- Think Kadec-Snowbar