Random encoding of quantized frame coefficients and compressed sensing measurements

Rayan Saab

Department of Mathematics, UC San Diego

October 2014

Joint work, in parts, with

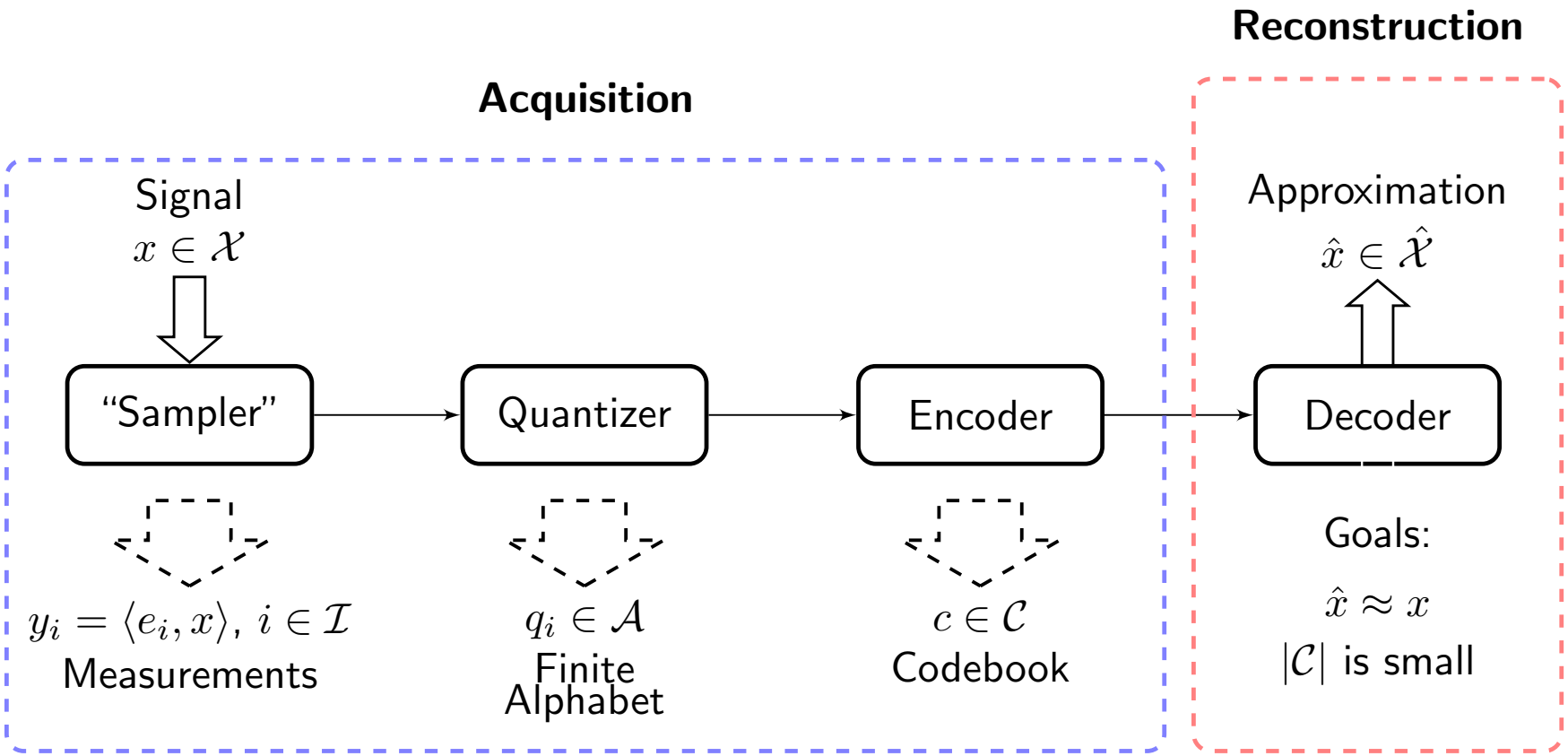
- Mark Iwen
- Rongrong Wang
- Özgür Yılmaz

Introduction

A typical signal acquisition and reconstruction paradigm:

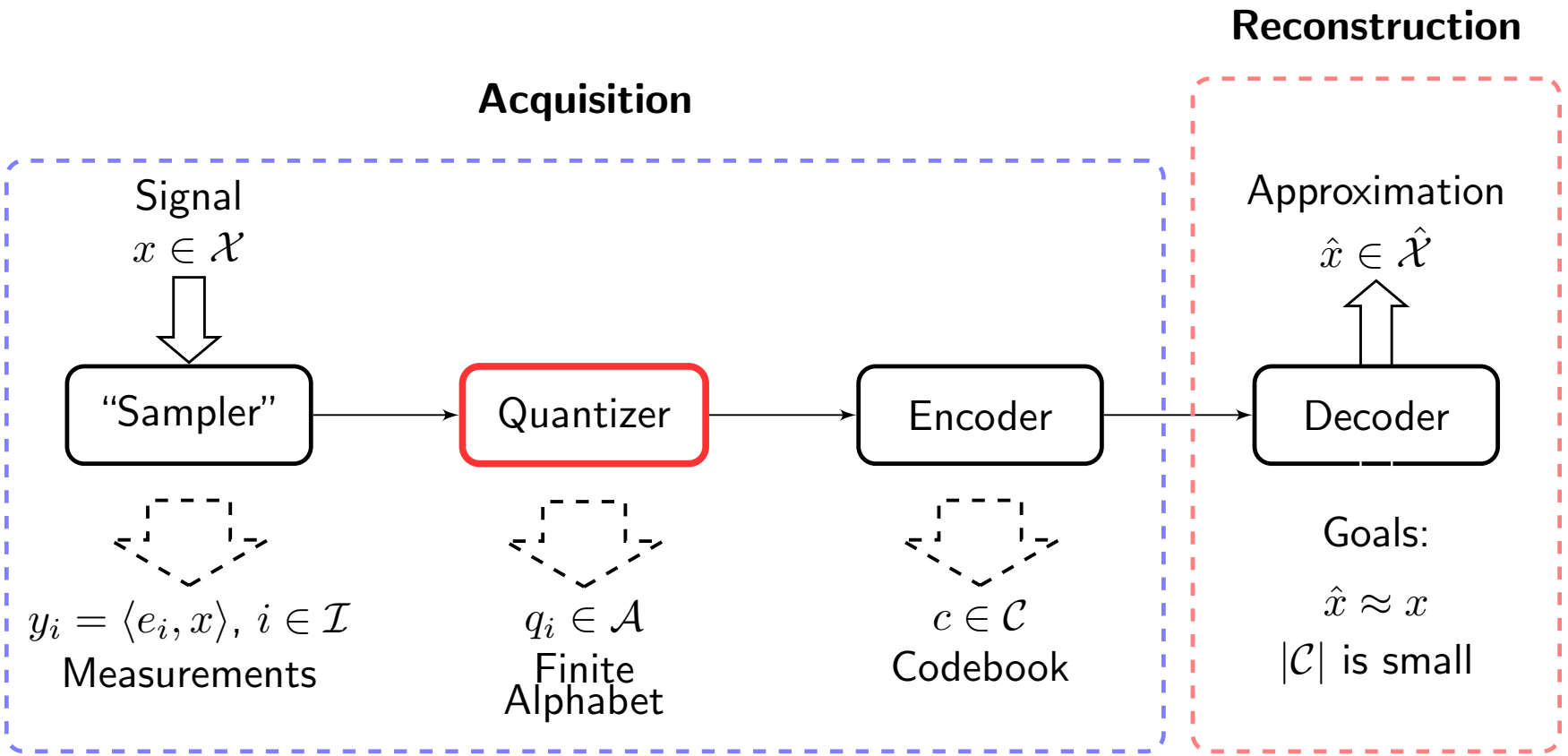
Introduction

A typical signal acquisition and reconstruction paradigm:



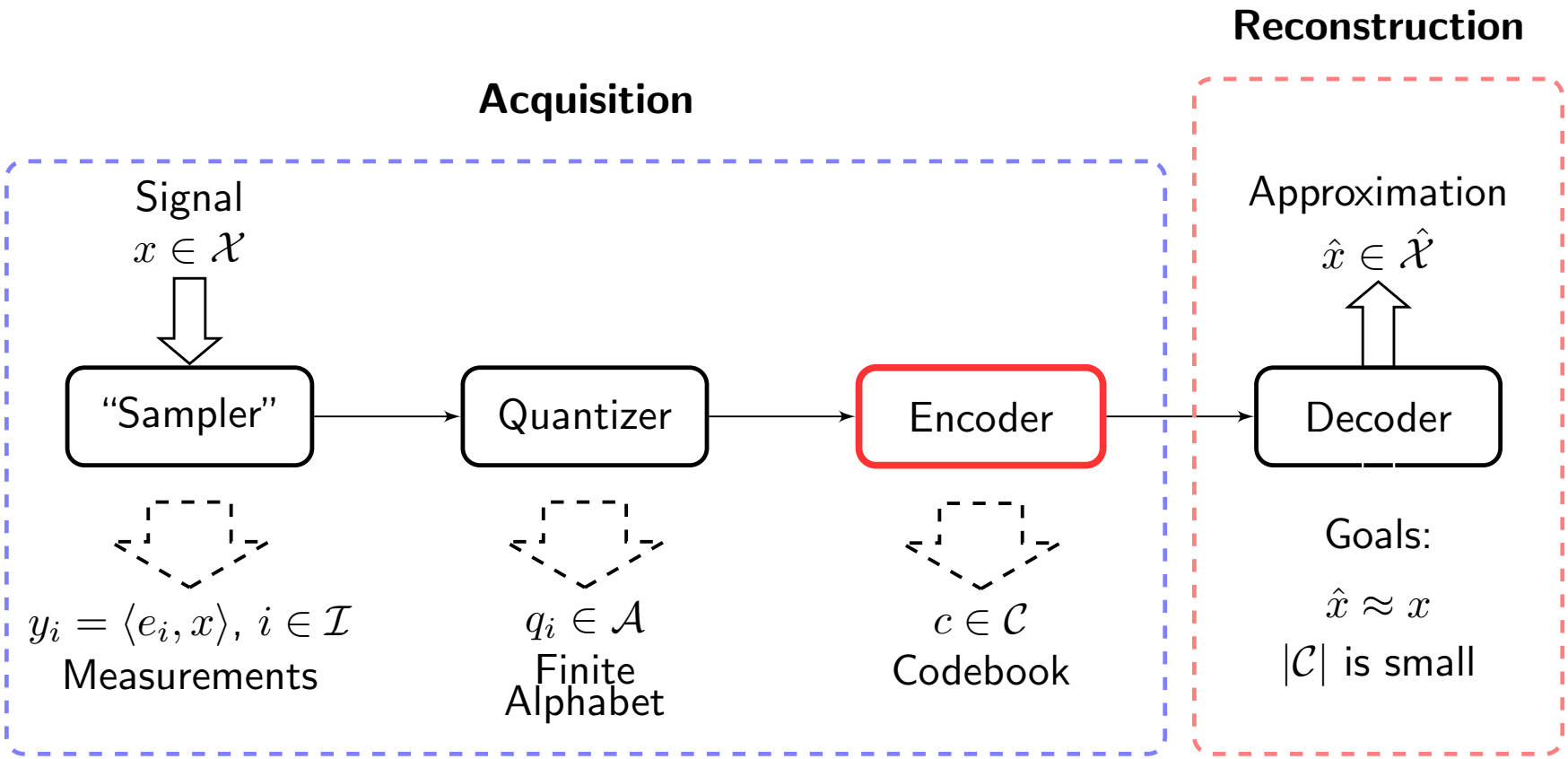
Introduction

A typical signal acquisition and reconstruction paradigm:



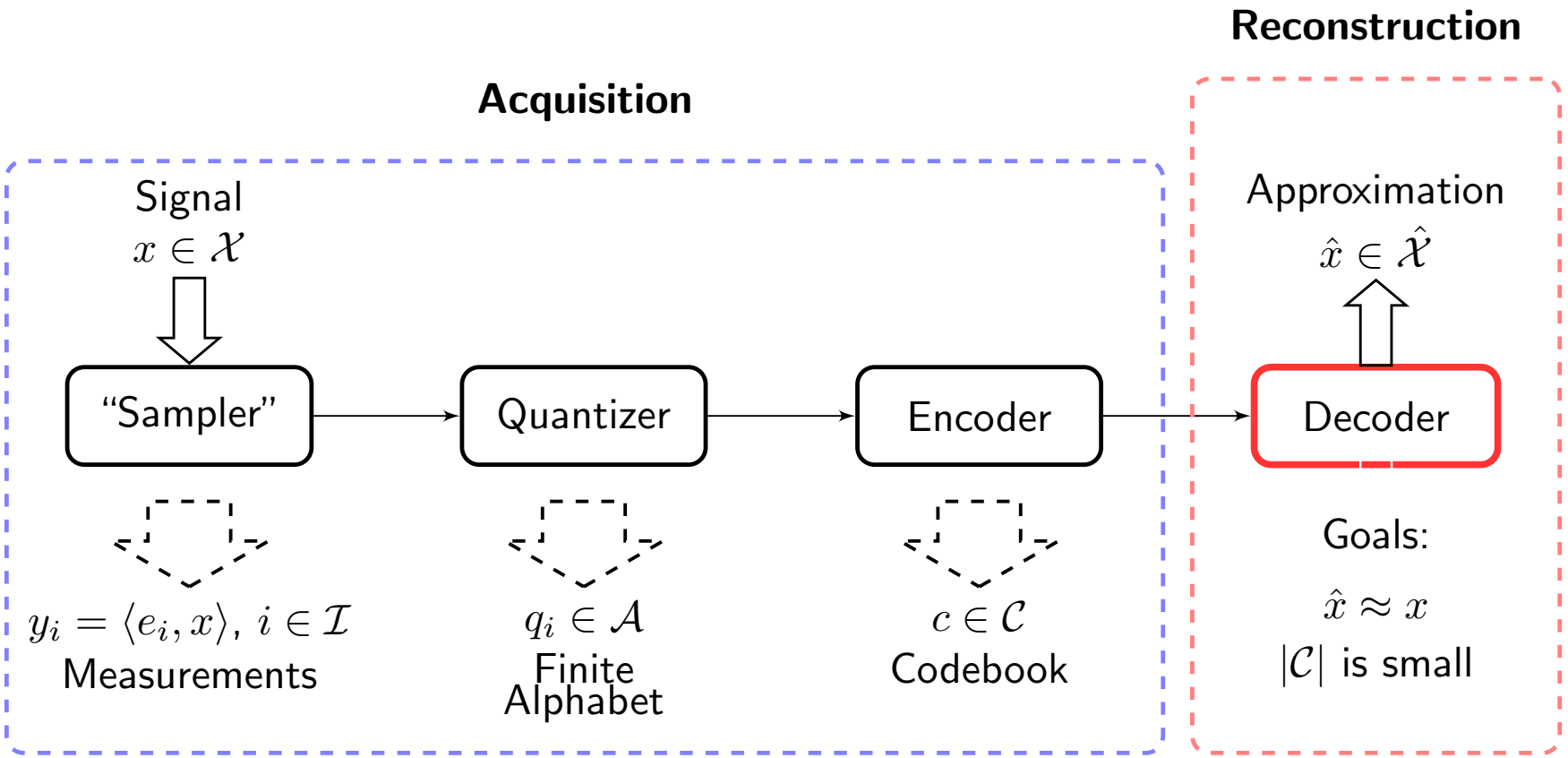
Introduction

A typical signal acquisition and reconstruction paradigm:



Introduction

A typical signal acquisition and reconstruction paradigm:



Introduction: Quantization and encoding for finite frames

Signal model: General finite dimensional signals

$$x \in \mathbb{R}^d, \text{ with } \|x\|_2 \leq 1.$$

Measurement model: Oversampling ($m > d$), i.e., take more measurements than ambient dimension.

Quantization: $\Sigma\Delta$ quantization.

Encoding: Lossy (Idea: decrease reconstruction accuracy slightly, decrease number of bits significantly.)

Decoding (reconstruction): Linear, albeit in lower dimensions, hence potentially much faster.

Introduction: Quantization and encoding for finite frames

Signal model: General finite dimensional signals

$$x \in \mathbb{R}^d, \text{ with } \|x\|_2 \leq 1.$$

Measurement model: Oversampling ($m > d$), i.e., take more measurements than ambient dimension.

Quantization: $\Sigma\Delta$ quantization.

Encoding: Lossy (Idea: decrease reconstruction accuracy slightly, decrease number of bits significantly.)

Decoding (reconstruction): Linear, albeit in lower dimensions, hence potentially much faster.

Introduction: Quantization and encoding for finite frames

Signal model: General finite dimensional signals

$$x \in \mathbb{R}^d, \text{ with } \|x\|_2 \leq 1.$$

Measurement model: Oversampling ($m > d$), i.e., take more measurements than ambient dimension.

Quantization: $\Sigma\Delta$ quantization.

Encoding: Lossy (Idea: decrease reconstruction accuracy slightly, decrease number of bits significantly.)

Decoding (reconstruction): Linear, albeit in lower dimensions, hence potentially much faster.

Introduction: Quantization and encoding for finite frames

Signal model: General finite dimensional signals

$$x \in \mathbb{R}^d, \text{ with } \|x\|_2 \leq 1.$$

Measurement model: Oversampling ($m > d$), i.e., take more measurements than ambient dimension.

Quantization: $\Sigma\Delta$ quantization.

Encoding: Lossy (Idea: decrease reconstruction accuracy slightly, decrease number of bits significantly.)

Decoding (reconstruction): Linear, albeit in lower dimensions, hence potentially much faster.

Introduction: Quantization and encoding for finite frames

Signal model: General finite dimensional signals

$$x \in \mathbb{R}^d, \text{ with } \|x\|_2 \leq 1.$$

Measurement model: Oversampling ($m > d$), i.e., take more measurements than ambient dimension.

Quantization: $\Sigma\Delta$ quantization.

Encoding: Lossy (Idea: decrease reconstruction accuracy slightly, decrease number of bits significantly.)

Decoding (reconstruction): Linear, albeit in lower dimensions, hence potentially much faster.

Introduction: Quantization and encoding for finite frames

Signal model: General finite dimensional signals

$$x \in \mathbb{R}^d, \text{ with } \|x\|_2 \leq 1.$$

Measurement model: Oversampling ($m > d$), i.e., take more measurements than ambient dimension.

Quantization: $\Sigma\Delta$ quantization.

Encoding: Lossy (Idea: decrease reconstruction accuracy slightly, decrease number of bits significantly.)

Decoding (reconstruction): Linear, albeit in lower dimensions, hence potentially much faster.

Finite frames

Recall $x \in \mathbb{R}^d$, $\|x\|_2 \leq 1$

Recall $m \gg d$ linear measurements: $y_i = \langle e_i, x \rangle, \quad i \in \{1, ..., m\}$

Finite frames

Recall $x \in \mathbb{R}^d$, $\|x\|_2 \leq 1$

Recall $m \gg d$ linear measurements: $y_i = \langle e_i, x \rangle, \quad i \in \{1, ..., m\}$

$$y = Ex$$

Finite frames

Recall $x \in \mathbb{R}^d$, $\|x\|_2 \leq 1$

Recall $m \gg d$ linear measurements: $y_i = \langle e_i, x \rangle, \quad i \in \{1, ..., m\}$

$$y = Ex$$

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_i \\ \vdots \\ \vdots \\ y_n \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} -e_1- \\ -e_2- \\ \vdots \\ -e_i- \\ \vdots \\ \vdots \\ -e_n- \end{bmatrix}}_{E} \begin{bmatrix} | \\ x \\ | \end{bmatrix}$$

Finite frames

Recall $x \in \mathbb{R}^d$, $\|x\|_2 \leq 1$

Recall $m \gg d$ linear measurements: $y_i = \langle e_i, x \rangle, \quad i \in \{1, ..., m\}$

$$y = Ex$$

Why frames? 1) $m > d$ can allow robustness to errors and erasures.

2) In the quantization context, we can improve the accuracy of the reconstruction.

Finite frames

Recall $x \in \mathbb{R}^d$, $\|x\|_2 \leq 1$

Recall $m \gg d$ linear measurements: $y_i = \langle e_i, x \rangle, \quad i \in \{1, ..., m\}$

$$y = Ex$$

Why frames? 1) $m > d$ can allow robustness to errors and erasures.

2) In the quantization context, we can improve the accuracy of the reconstruction.

The quantization problem

Vector quantization: (measuring the worst case error in ℓ_2 norm)

- To achieve an error of ϵ, must cover B_2^d with balls of radius ϵ
- So,
$$\sum \text{Volume(balls)} \geq \text{Volume } (B_2^d).$$
- So,
$$\#\text{balls} \times \epsilon^d \geq 1.$$
- But $\mathcal{R} := \text{ total number of bits } = \log_2(\#\text{balls})$, so
$$\boxed{\epsilon \geq 2^{-\mathcal{R}/d}.}$$
- On the other hand, there exist covers with
$$\#\text{balls} \leq \left(\frac{3}{\epsilon}\right)^d \implies \boxed{\epsilon \leq 3 \cdot 2^{-\mathcal{R}/d}}$$

The quantization problem

Vector quantization: (measuring the worst case error in ℓ_2 norm)

- To achieve an error of ϵ, must cover B_2^d with balls of radius ϵ
- So,
$$\sum \text{Volume(balls)} \geq \text{Volume } (B_2^d).$$
- So,
$$\#\text{balls} \times \epsilon^d \geq 1.$$
- But $\mathcal{R} := \text{ total number of bits } = \log_2(\#\text{balls})$, so
$$\boxed{\epsilon \geq 2^{-\mathcal{R}/d}.}$$
- On the other hand, there exist covers with
$$\#\text{balls} \leq \left(\frac{3}{\epsilon}\right)^d \implies \boxed{\epsilon \leq 3 \cdot 2^{-\mathcal{R}/d}}$$

The quantization problem

Vector quantization: (measuring the worst case error in ℓ_2 norm)

- To achieve an error of ϵ, must cover B_2^d with balls of radius ϵ
- So,

$$\sum \text{Volume(balls)} \geq \text{Volume } (B_2^d).$$

- So,

$$\#\text{balls} \times \epsilon^d \geq 1.$$

- But $\mathcal{R} :=$ total number of bits $= \log_2(\#\text{balls})$, so

$$\boxed{\epsilon \geq 2^{-\mathcal{R}/d}.}$$

- On the other hand, there exist covers with

$$\#\text{balls} \leq \left(\frac{3}{\epsilon}\right)^d \implies \boxed{\epsilon \leq 3 \cdot 2^{-\mathcal{R}/d}}$$

The quantization problem

Vector quantization: (measuring the worst case error in ℓ_2 norm)

- To achieve an error of ϵ, must cover B_2^d with balls of radius ϵ
- So,

$$\sum \text{Volume(balls)} \geq \text{Volume } (B_2^d).$$

- So,

$$\#\text{balls} \times \epsilon^d \geq 1.$$

- But $\mathcal{R} :=$ total number of bits $= \log_2(\#\text{balls})$, so

$$\boxed{\epsilon \geq 2^{-\mathcal{R}/d}.}$$

- On the other hand, there exist covers with

$$\#\text{balls} \leq \left(\frac{3}{\epsilon}\right)^d \implies \boxed{\epsilon \leq 3 \cdot 2^{-\mathcal{R}/d}}$$

The quantization problem

Issues with vector quantization as above:

1) Assumes direct access to x, which is not always possible (e.g., MRI, coded aperture, channel estimation).
Must work with Ex.

2) As ϵ decreases the number of balls increases as ϵ^{-d}.

To encode we replace x by the closest center of a ball.
Potentially too many comparisons (curse of dimensionality).

3) Need hardware that can tell apart centers separated by ϵ.

The quantization problem

Issues with vector quantization as above:

1) Assumes direct access to x, which is not always possible (e.g., MRI, coded aperture, channel estimation).
Must work with Ex.

2) As ϵ decreases the number of balls increases as ϵ^{-d}.

To encode we replace x by the closest center of a ball.
Potentially too many comparisons (curse of dimensionality).

3) Need hardware that can tell apart centers separated by ϵ.

The quantization problem

Issues with vector quantization as above:

1) Assumes direct access to x, which is not always possible (e.g., MRI, coded aperture, channel estimation).
Must work with Ex.

2) As ϵ decreases the number of balls increases as ϵ^{-d}.

To encode we replace x by the closest center of a ball.
Potentially too many comparisons (curse of dimensionality).

3) Need hardware that can tell apart centers separated by ϵ.

The quantization problem

Issues with vector quantization as above:

1) Assumes direct access to x, which is not always possible (e.g., MRI, coded aperture, channel estimation).
Must work with Ex.

2) As ϵ decreases the number of balls increases as ϵ^{-d}.

To encode we replace x by the closest center of a ball.
Potentially too many comparisons (curse of dimensionality).

3) Need hardware that can tell apart centers separated by ϵ.

The quantization problem

Issues with vector quantization as above:

1) Assumes direct access to x, which is not always possible (e.g., MRI, coded aperture, channel estimation).
Must work with Ex.

2) As ϵ decreases the number of balls increases as ϵ^{-d}.

To encode we replace x by the closest center of a ball.
Potentially too many comparisons (curse of dimensionality).

3) Need hardware that can tell apart centers separated by ϵ.

The quantization problem

The quantization problem

Scalar quantization of frame coefficients:

Alphabet: $\mathcal{A}$ is a fixed finite set, e.g., $\mathcal{A} = \{+1, -1\}$.

Goal: For each $i \in \{1, ..., m\}$:

$$\langle e_i, x \rangle \mapsto q_i \in \mathcal{A}.$$

Scalar quantizer associated with the alphabet $\mathcal{A}$

$$\boxed{Q(v) := \arg\min_{q \in \mathcal{A}} |q - v|.}$$

Example: $Q(\langle e_i, x \rangle) = \text{sign}(\langle e_i, x \rangle)$.

The quantization problem

Scalar quantization of frame coefficients:

Alphabet: $\mathcal{A}$ is a fixed finite set, e.g., $\mathcal{A} = \{+1, -1\}$.

Goal: For each $i \in \{1, ..., m\}$:

$$\langle e_i, x \rangle \mapsto q_i \in \mathcal{A}.$$

Scalar quantizer associated with the alphabet $\mathcal{A}$

$$Q(v) := \arg\min_{q \in \mathcal{A}} |q - v|.$$

Example: $Q(\langle e_i, x \rangle) = \text{sign}(\langle e_i, x \rangle)$.

The quantization problem

Scalar quantization of frame coefficients:

Alphabet: $\mathcal{A}$ is a fixed finite set, e.g., $\mathcal{A} = \{+1, -1\}$.

Goal: For each $i \in \{1, ..., m\}$:

$$\langle e_i, x \rangle \mapsto q_i \in \mathcal{A}.$$

Scalar quantizer associated with the alphabet $\mathcal{A}$

$$\boxed{Q(v) := \arg\min_{q \in \mathcal{A}} |q - v|.}$$

Example: $Q(\langle e_i, x \rangle) = \text{sign}(\langle e_i, x \rangle)$.

The quantization problem

Scalar quantization of frame coefficients:

Alphabet: $\mathcal{A}$ is a fixed finite set, e.g., $\mathcal{A} = \{+1, -1\}$.

Goal: For each $i \in \{1, ..., m\}$:

$$\langle e_i, x \rangle \mapsto q_i \in \mathcal{A}.$$

Scalar quantizer associated with the alphabet $\mathcal{A}$

$$\boxed{Q(v) := \arg\min_{q \in \mathcal{A}} |q - v|.}$$

Example: $Q(\langle e_i, x \rangle) = \text{sign}(\langle e_i, x \rangle)$.

The quantization problem

Issue with scalar quantization:

It is highly inefficient from a rate-distortion perspective:

Define $b := \log_2 |\mathcal{A}|$ and note that b is fixed.

Theorem (Goyal, Vetterli, Thao (1998))
The expected error (over x) satisfies

$$(E\|x - \hat{x}\|_2^2)^{1/2} \gtrsim \frac{d}{m} 2^{-b}.$$

Since the total rate is $\mathcal{R} := mb$,

$$\epsilon \gtrsim \frac{d \cdot b}{\mathcal{R}} 2^{-b}.$$

The quantization problem

Issue with scalar quantization:

It is highly inefficient from a rate-distortion perspective:

Define $b := \log_2 |\mathcal{A}|$ and note that b is fixed.

Theorem (Goyal, Vetterli, Thao (1998))
The expected error (over x) satisfies

$$(E\|x - \hat{x}\|_2^2)^{1/2} \gtrsim \frac{d}{m} 2^{-b}.$$

Since the total rate is $\mathcal{R} := mb$,

$$\epsilon \gtrsim \frac{d \cdot b}{\mathcal{R}} 2^{-b}.$$

The quantization problem

Issue with scalar quantization:

It is highly inefficient from a rate-distortion perspective:

Define $b := \log_2 |\mathcal{A}|$ and note that b is fixed.

Theorem (Goyal, Vetterli, Thao (1998))
The expected error (over x) satisfies

$$(E\|x - \hat{x}\|_2^2)^{1/2} \gtrsim \frac{d}{m} 2^{-b}.$$

Since the total rate is $\mathcal{R} := mb$,

$$\epsilon \gtrsim \frac{d \cdot b}{\mathcal{R}} 2^{-b}.$$

The quantization problem

Issue with scalar quantization:

It is highly inefficient from a rate-distortion perspective:

Define $b := \log_2 |\mathcal{A}|$ and note that b is fixed.

Theorem (Goyal, Vetterli, Thao (1998))
The expected error (over x) satisfies

$$(E\|x - \hat{x}\|_2^2)^{1/2} \gtrsim \frac{d}{m} 2^{-b}.$$

Since the total rate is $\mathcal{R} := mb$,

$$\epsilon \gtrsim \frac{d \cdot b}{\mathcal{R}} 2^{-b}.$$

The quantization problem

Issue with scalar quantization:

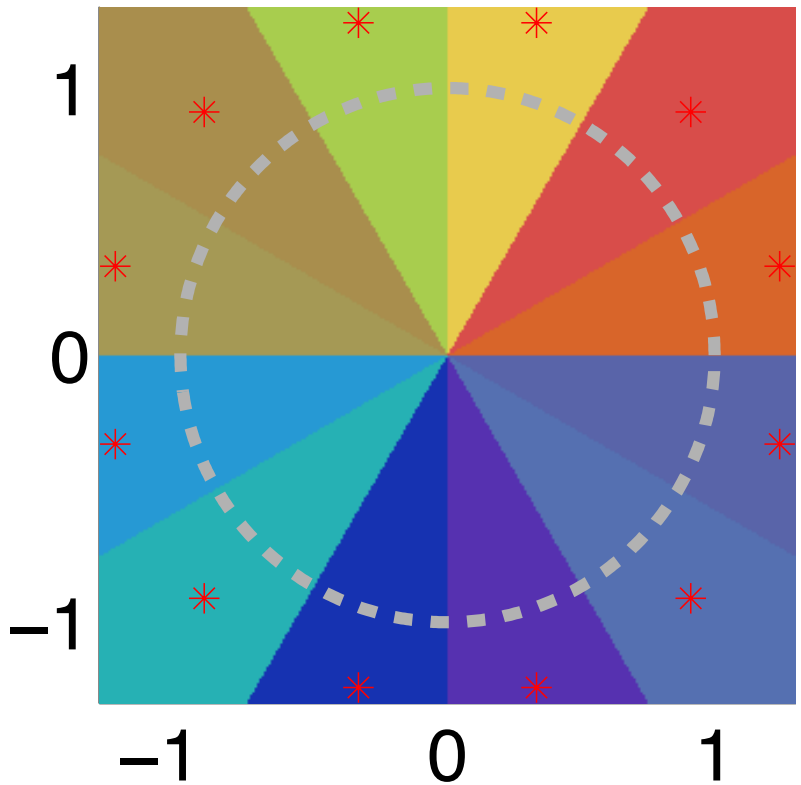


Figure : Quantization cells and reconstruction points: Scalar quantization and canonical dual

$\Sigma\Delta$ quantization

$\Sigma\Delta$ quantization

Scalar quantizer associated with the alphabet $\mathcal{A}$

$$Q(v) := \arg\min_{q\in\mathcal{A}} |q - v|.$$

1st order $\Sigma\Delta$ scheme with alphabet $\mathcal{A}$ (greedy rule):

$$\begin{cases} q_i & = & Q(u_{i-1} + y_i) \\ (\Delta u)_i & := & u_i - u_{i-1} = y_i - q_i \end{cases}$$

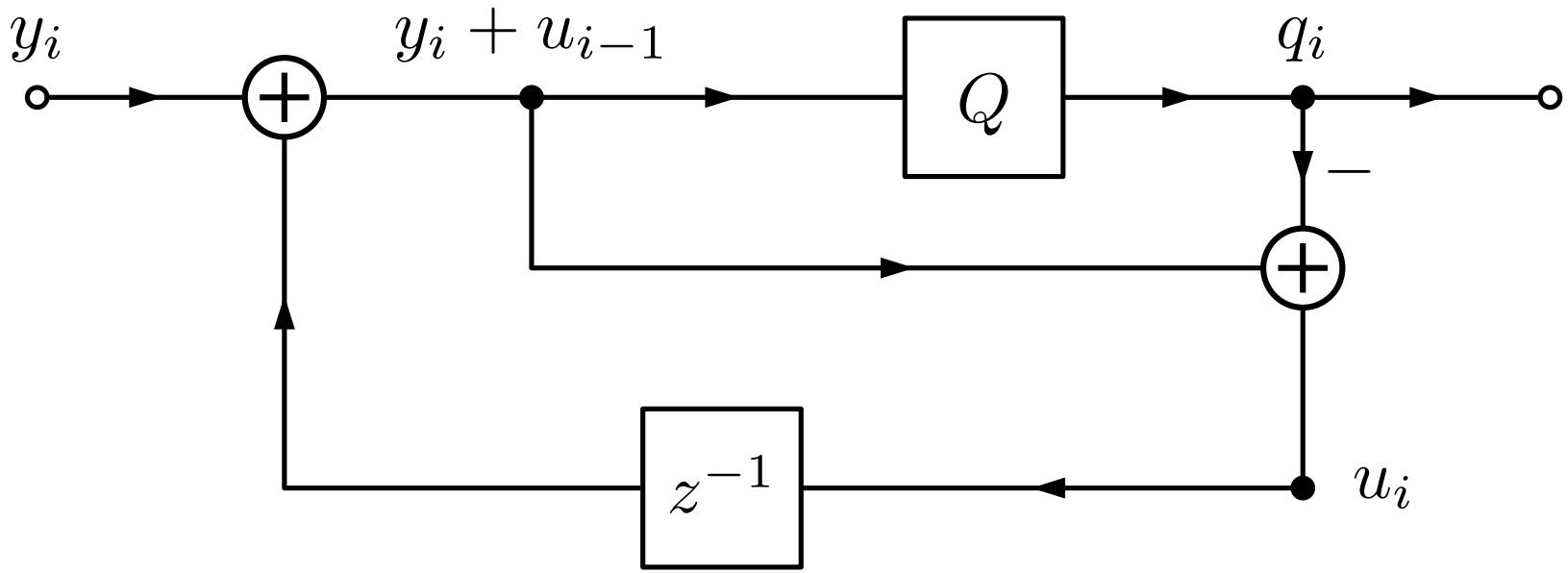
ΣΔ quantization

Scalar quantizer associated with the alphabet $\mathcal{A}$

$$Q(v) := \arg\min_{q\in\mathcal{A}} |q - v|.$$

1st order ΣΔ scheme with alphabet $\mathcal{A}$ (greedy rule):

$$\begin{cases} q_i & = & Q(u_{i-1} + y_i) \\ (\Delta u)_i & := & u_i - u_{i-1} = y_i - q_i \end{cases}$$



ΣΔ quantization

Scalar quantizer associated with the alphabet $\mathcal{A}$

$$Q(v) := \arg\min_{q\in\mathcal{A}} |q - v|.$$

1st order ΣΔ scheme with alphabet $\mathcal{A}$ (greedy rule):

$$\begin{cases} q_i & = & Q(u_{i-1} + y_i) \\ (\Delta u)_i & := & u_i - u_{i-1} = y_i - q_i \end{cases}$$

rth order ΣΔ scheme with alphabet $\mathcal{A}$ (arbitrary rule ρ):

$$\begin{cases} q_i & = & Q\left(\rho(u_{i-r}, ..., u_{i-1}, y_i)\right) \\ (\Delta^r u)_i & := & y_i - q_i \end{cases}$$

Stability:

$$\|y\|_\infty \leq C_1 \implies \|u\|_\infty \leq C_2.$$

ΣΔ quantization

Scalar quantizer associated with the alphabet $\mathcal{A}$

$$Q(v) := \arg\min_{q\in\mathcal{A}} |q - v|.$$

1st order ΣΔ scheme with alphabet $\mathcal{A}$ (greedy rule):

$$\begin{cases} q_i & = & Q(u_{i-1} + y_i) \\ (\Delta u)_i & := & u_i - u_{i-1} = y_i - q_i \end{cases}$$

rth order ΣΔ scheme with alphabet $\mathcal{A}$ (arbitrary rule ρ):

$$\begin{cases} q_i & = & Q\left(\rho(u_{i-r}, ..., u_{i-1}, y_i)\right) \\ (\Delta^r u)_i & := & y_i - q_i \end{cases}$$

Stability:

$$\|y\|_\infty \leq C_1 \implies \|u\|_\infty \leq C_2.$$

Why $\Sigma\Delta$ for finite frames?

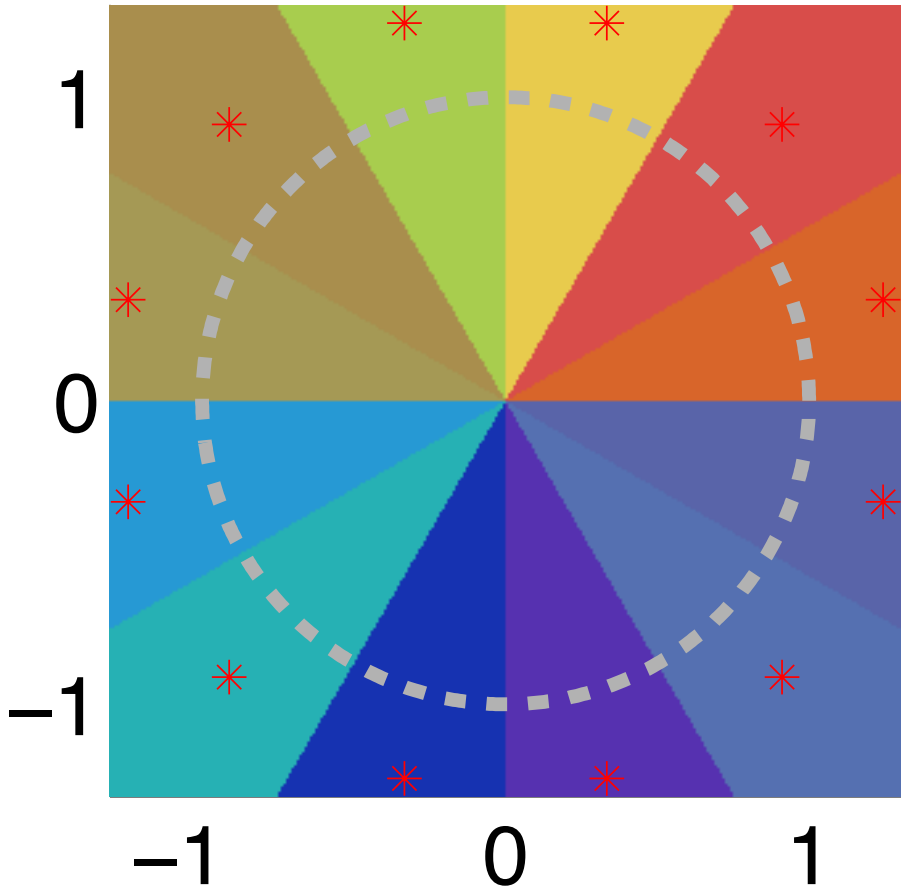


Figure : Quantization cells and reconstruction points: Scalar quantization and canonical dual

Why $\Sigma\Delta$ for finite frames?

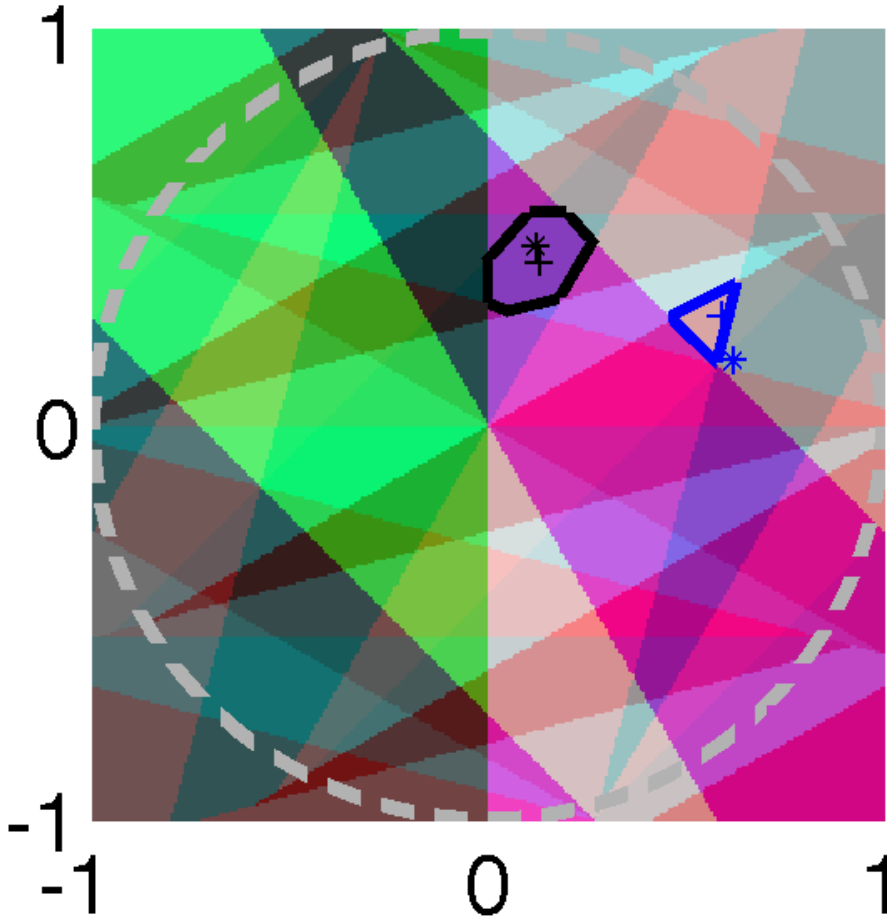


Figure : Quantization cells and reconstruction points: 1st order $\Sigma\Delta$ quantization and two duals

Error estimates: Previous work (smooth frames)

Goyal, Vetterli, Thao (1998)
Scalar quantization (Lower bound – holds for any frame):

$$(E\|x-\hat{x}\|_2^2)^{1/2} \gtrsim \frac{d}{m}.$$

Benedetto et al. (2006), Bodmann et al. (2006), Blum et al. (2010)
Smooth frames, rth order $\Sigma\Delta$:

$$\|x-\hat{x}\|_2 \leq C(r,E)\left(\frac{d}{m}\right)^r$$

Krahmer, S., Ward (2012)
harmonic frames, Sobolev self-dual frames:

$$\exists r(m) \text{ such that } \|x-\hat{x}\|_2 \leq C_1\ e^{-c_2\sqrt{\frac{m}{d}}}$$

Random frames

Obervation: For smooth frames, the error decay polynomially (and by optimizing the order r, root-exponentially) in the number of measurements.

Question: Can we extend these results to hold for non-smooth (particularly, random) frames?

Random frames

Obervation: For smooth frames, the error decay polynomially (and by optimizing the order r, root-exponentially) in the number of measurements.

Question: Can we extend these results to hold for non-smooth (particularly, random) frames?

Error estimates (sub-Gaussian random frames)

Definition: If two random variables $\eta \sim \mathcal{D}_1$ and $\xi \sim \mathcal{D}_2$ satisfy $P(|\eta| > t) \leq KP(|\xi| > t)$ for some constant K and all $t \geq 0$, then we say that η is K-dominated by ξ (or, alternatively, by $\mathcal{D}_2$).

Definition: A random variable is sub-Gaussian with parameter $c > 0$ if it is e-dominated by $\mathcal{N}(0, c^2)$.

Remark: This is equivalent to the standard definitions of sub-Gaussian random variables via their moments or, for zero-mean r.v.'s, their m.g.f.'s

Definition: A matrix/frame E is sub-Gaussian with parameter c, mean μ and variance σ^2 if its entries are independent, sub-Gaussian random variables with mean μ, variance σ^2, and parameter c.

Error estimates (sub-Gaussian random frames)

Definition: If two random variables $\eta \sim \mathcal{D}_1$ and $\xi \sim \mathcal{D}_2$ satisfy $P(|\eta| > t) \leq KP(|\xi| > t)$ for some constant K and all $t \geq 0$, then we say that η is K-dominated by ξ (or, alternatively, by $\mathcal{D}_2$).

Definition: A random variable is sub-Gaussian with parameter $c > 0$ if it is e-dominated by $\mathcal{N}(0, c^2)$.

Remark: This is equivalent to the standard definitions of sub-Gaussian random variables via their moments or, for zero-mean r.v.'s, their m.g.f.'s

Definition: A matrix/frame E is sub-Gaussian with parameter c, mean μ and variance σ^2 if its entries are independent, sub-Gaussian random variables with mean μ, variance σ^2, and parameter c.

Error estimates (sub-Gaussian random frames)

Definition: If two random variables $\eta \sim \mathcal{D}_1$ and $\xi \sim \mathcal{D}_2$ satisfy $P(|\eta| > t) \leq KP(|\xi| > t)$ for some constant K and all $t \geq 0$, then we say that η is K-dominated by ξ (or, alternatively, by $\mathcal{D}_2$).

Definition: A random variable is sub-Gaussian with parameter $c > 0$ if it is e-dominated by $\mathcal{N}(0, c^2)$.

Remark: This is equivalent to the standard definitions of sub-Gaussian random variables via their moments or, for zero-mean r.v.'s, their m.g.f.'s

Definition: A matrix/frame E is sub-Gaussian with parameter c, mean μ and variance σ^2 if its entries are independent, sub-Gaussian random variables with mean μ, variance σ^2, and parameter c.

Error estimates (sub-Gaussian random frames)

Definition: If two random variables $\eta \sim \mathcal{D}_1$ and $\xi \sim \mathcal{D}_2$ satisfy $P(|\eta| > t) \leq KP(|\xi| > t)$ for some constant K and all $t \geq 0$, then we say that η is K-dominated by ξ (or, alternatively, by $\mathcal{D}_2$).

Definition: A random variable is sub-Gaussian with parameter $c > 0$ if it is e-dominated by $\mathcal{N}(0, c^2)$.

Remark: This is equivalent to the standard definitions of sub-Gaussian random variables via their moments or, for zero-mean r.v.'s, their m.g.f.'s

Definition: A matrix/frame E is sub-Gaussian with parameter c, mean μ and variance σ^2 if its entries are independent, sub-Gaussian random variables with mean μ, variance σ^2, and parameter c.

Error estimates (sub-Gaussian random frames)

Güntürk, Lammers, Powell, S., Yılmaz (2013)
Krahmer, S., Yılmaz (2013)

With high probability on the draw of E:

$$\|x - \hat{x}\|_2 \leq C(r) \left(\frac{d}{m}\right)^{\alpha(r-1/2)},$$

(where $\alpha < 1$ controls the probability)

Krahmer, S., Yılmaz (2013)
$\forall m \geq m_0, \exists r(m)$ such that

$$\|x - \hat{x}\|_2 \leq C_1 \ \exp\left(-c_2 \left(\frac{m}{d}\right)^{\alpha/2}\right).$$

Further gains: Encoding the bit-stream

Obervation: Best error-rates we've seen decay root-exponentially in the number of measurements, hence in the number of bits:

$$\epsilon \lesssim e^{-c\sqrt{\mathcal{R}/d}}$$

Question: Can we encode (compress) the $\Sigma\Delta$ bit-stream to obtain a distortion that decreases exponentially fast with the bit-rate?

Recall: Exponential decay is the best we can hope for.

Denote by D the $m \times m$ difference matrix

$$D := \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \cdots & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & -1 & 1 \end{bmatrix}$$

$$Ex + D^r u = q$$

Further gains: Encoding the bit-stream

Obervation: Best error-rates we've seen decay root-exponentially in the number of measurements, hence in the number of bits:

$$\epsilon \lesssim e^{-c\sqrt{\mathcal{R}/d}}$$

Question: Can we encode (compress) the $\Sigma\Delta$ bit-stream to obtain a distortion that decreases exponentially fast with the bit-rate?

Recall: Exponential decay is the best we can hope for.

Denote by D the $m \times m$ difference matrix

$$D := \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \cdots & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & -1 & 1 \end{bmatrix}$$

$$Ex + D^r u = q$$

Further gains: Encoding the bit-stream

Obervation: Best error-rates we've seen decay root-exponentially in the number of measurements, hence in the number of bits:

$$\epsilon \lesssim e^{-c\sqrt{\mathcal{R}/d}}$$

Question: Can we encode (compress) the $\Sigma\Delta$ bit-stream to obtain a distortion that decreases exponentially fast with the bit-rate?

Recall: Exponential decay is the best we can hope for.

Denote by D the $m \times m$ difference matrix

$$D := \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \cdots & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & -1 & 1 \end{bmatrix}$$

$$Ex + D^r u = q$$

Further gains: Encoding the bit-stream

Obervation: Best error-rates we've seen decay root-exponentially in the number of measurements, hence in the number of bits:

$$\epsilon \lesssim e^{-c\sqrt{\mathcal{R}/d}}$$

Question: Can we encode (compress) the $\Sigma\Delta$ bit-stream to obtain a distortion that decreases exponentially fast with the bit-rate?

Recall: Exponential decay is the best we can hope for.

Denote by D the $m \times m$ difference matrix

$$D := \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \cdots & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & -1 & 1 \end{bmatrix}$$

$$\boxed{Ex + D^r u = q}$$

Idea:

Since

$$Ex + D^r u = q \qquad \Leftrightarrow \qquad D^{-r}(Ex - q) = u$$

and since

$$\|u\|_2 \leq \sqrt{m}\frac{\delta}{2}$$

it is reasonable to approximate x from q by

$$\hat{x} := \arg\min_{z \in \mathbb{R}^d} \|D^{-r}(Ez - q)\|_2.$$

Implicitly: all the previously mentioned results use this approach.

Can approximate $\hat{x}$ with

$$x^\sharp := \arg\min_{z \in \mathbb{R}^d} \|BD^{-r}(Ez - q)\|_2,$$

where B is a JL matrix.

Idea:

Since

$$Ex + D^r u = q \qquad \Leftrightarrow \qquad D^{-r}(Ex - q) = u$$

and since

$$\|u\|_2 \leq \sqrt{m}\frac{\delta}{2}$$

it is reasonable to approximate x from q by

$$\hat{x} := \arg\min_{z \in \mathbb{R}^d} \|D^{-r}(Ez - q)\|_2.$$

Implicitly: all the previously mentioned results use this approach.

Can approximate $\hat{x}$ with

$$x^\sharp := \arg\min_{z \in \mathbb{R}^d} \|BD^{-r}(Ez - q)\|_2,$$

where B is a JL matrix.

Idea:

Since

$$Ex + D^r u = q \qquad \Leftrightarrow \qquad D^{-r}(Ex - q) = u$$

and since

$$\|u\|_2 \leq \sqrt{m}\frac{\delta}{2}$$

it is reasonable to approximate x from q by

$$\hat{x} := \arg\min_{z \in \mathbb{R}^d} \|D^{-r}(Ez - q)\|_2.$$

Implicitly: all the previously mentioned results use this approach.

Can approximate $\hat{x}$ with

$$x^\sharp := \arg\min_{z \in \mathbb{R}^d} \|BD^{-r}(Ez - q)\|_2,$$

where B is a JL matrix.

Idea:

Since

$$Ex + D^r u = q \qquad \Leftrightarrow \qquad D^{-r}(Ex - q) = u$$

and since

$$\|u\|_2 \leq \sqrt{m}\frac{\delta}{2}$$

it is reasonable to approximate x from q by

$$\hat{x} := \arg\min_{z \in \mathbb{R}^d} \|D^{-r}(Ez - q)\|_2.$$

Implicitly: all the previously mentioned results use this approach.

Can approximate $\hat{x}$ with

$$x^\sharp := \arg\min_{z \in \mathbb{R}^d} \|BD^{-r}(Ez - q)\|_2,$$

where B is a JL matrix.

Idea:

Since

$$Ex + D^r u = q \qquad \Leftrightarrow \qquad D^{-r}(Ex - q) = u$$

and since

$$\|u\|_2 \leq \sqrt{m}\frac{\delta}{2}$$

it is reasonable to approximate x from q by

$$\hat{x} := \arg\min_{z \in \mathbb{R}^d} \|D^{-r}(Ez - q)\|_2.$$

Implicitly: all the previously mentioned results use this approach.

Can approximate $\hat{x}$ with

$$\boxed{x^\sharp := \arg\min_{z \in \mathbb{R}^d} \|BD^{-r}(Ez - q)\|_2,}$$

where B is a JL matrix.

Near optimal encoding/decoding

The encoding algorithm:

Step 1: Digitally integrate the $\Sigma\Delta$ bit-stream $\longrightarrow D^{-r}q$

Step 2: Reduce the dimension of the integrated bit-stream via a discrete Johnson-Lindenstrauss embedding $\longrightarrow BD^{-r}q$.

Example: B is a $\ell \times m$ Bernoulli matrix with ± 1 entries and $\ell \approx d$.

Step 3: Assign binary labels to entries of $BD^{-r}q$.

Near optimal encoding/decoding

The encoding algorithm:

Step 1: Digitally integrate the $\Sigma\Delta$ bit-stream $\longrightarrow D^{-r}q$

Step 2: Reduce the dimension of the integrated bit-stream via a discrete Johnson-Lindenstrauss embedding $\longrightarrow BD^{-r}q$.

Example: B is a $\ell \times m$ Bernoulli matrix with ± 1 entries and $\ell \approx d$.

Step 3: Assign binary labels to entries of $BD^{-r}q$.

Near optimal encoding/decoding

The encoding algorithm:

Step 1: Digitally integrate the $\Sigma\Delta$ bit-stream $\longrightarrow D^{-r}q$

Step 2: Reduce the dimension of the integrated bit-stream via a discrete Johnson-Lindenstrauss embedding $\longrightarrow BD^{-r}q$.

Example: B is a $\ell \times m$ Bernoulli matrix with ± 1 entries and $\ell \approx d$.

Step 3: Assign binary labels to entries of $BD^{-r}q$.

Near optimal encoding/decoding

The encoding algorithm:

Step 1: Digitally integrate the $\Sigma\Delta$ bit-stream $\longrightarrow D^{-r}q$

Step 2: Reduce the dimension of the integrated bit-stream via a discrete Johnson-Lindenstrauss embedding $\longrightarrow BD^{-r}q$.

Example: B is a $\ell \times m$ Bernoulli matrix with ± 1 entries and $\ell \approx d$.

Step 3: Assign binary labels to entries of $BD^{-r}q$.

The decoding algorithm:

Step 1: Recover $BD^{-r}q$ from is binary labels.

Step 2: Apply an appropriate linear operator $G := (BD^{-r}E)^{\dagger}$ to $BD^{-r}q$

$$\hat{x} = GBD^{-r}q.$$

Near optimal encoding/decoding

The encoding algorithm:

Step 1: Digitally integrate the $\Sigma\Delta$ bit-stream $\longrightarrow D^{-r}q$

Step 2: Reduce the dimension of the integrated bit-stream via a discrete Johnson-Lindenstrauss embedding $\longrightarrow BD^{-r}q$.

Example: B is a $\ell \times m$ Bernoulli matrix with ± 1 entries and $\ell \approx d$.

Step 3: Assign binary labels to entries of $BD^{-r}q$.

The decoding algorithm:

Step 1: Recover $BD^{-r}q$ from is binary labels.

Step 2: Apply an appropriate linear operator $G := (BD^{-r}E)^{\dagger}$ to $BD^{-r}q$

$$\boxed{\hat{x} = GBD^{-r}q.}$$

Near optimal encoding/decoding: Numerical experiments

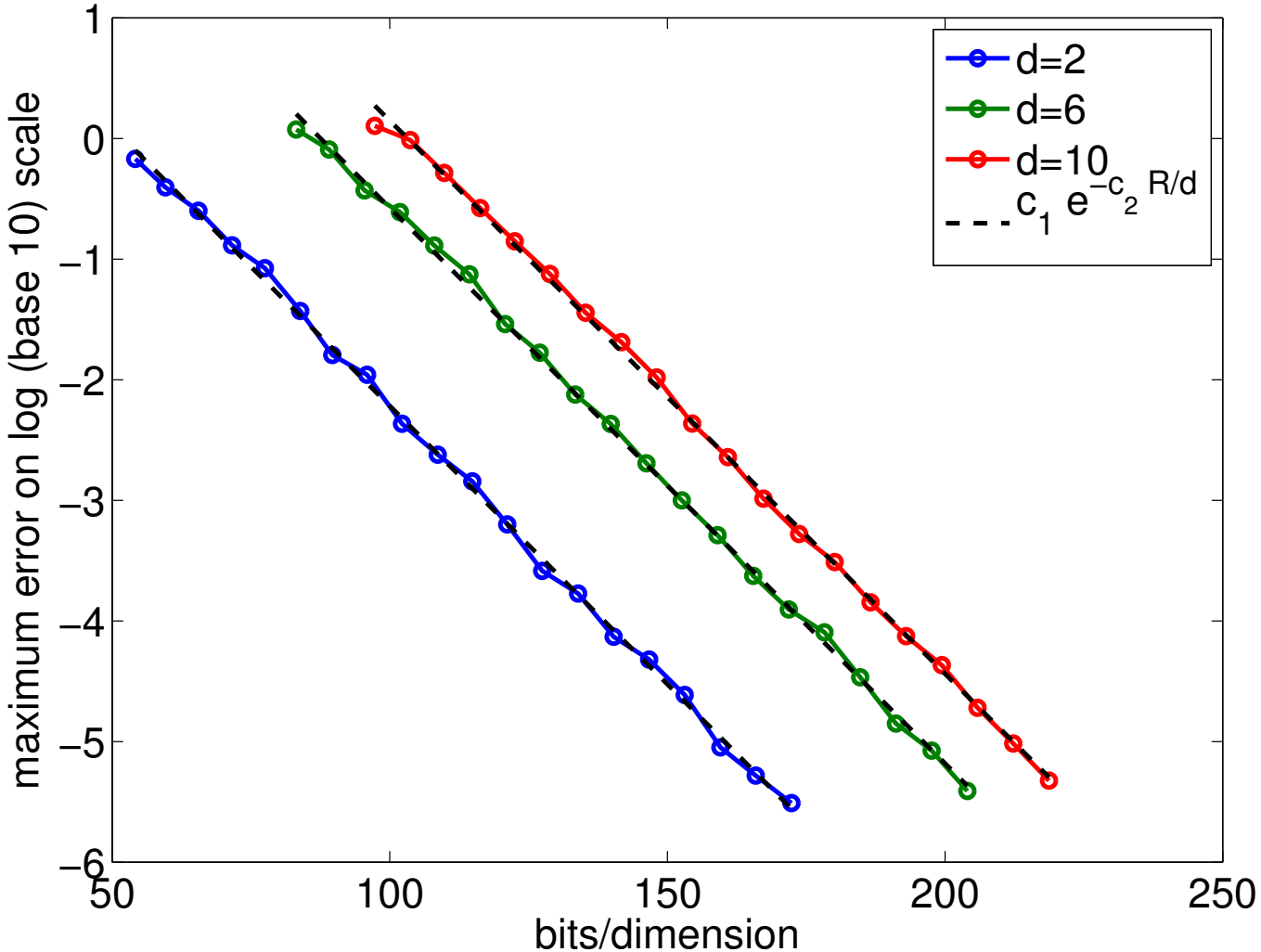


Figure : The maximum error versus the number of bits, 1st order $\Sigma\Delta$. The maximum is over 5000 instances of x from the unit ball of $\mathbb{R}^d$, for $d = 2, 6$ and 10. (the y-axis is on log-scale)

Near optimal encoding/decoding: Numerical experiments

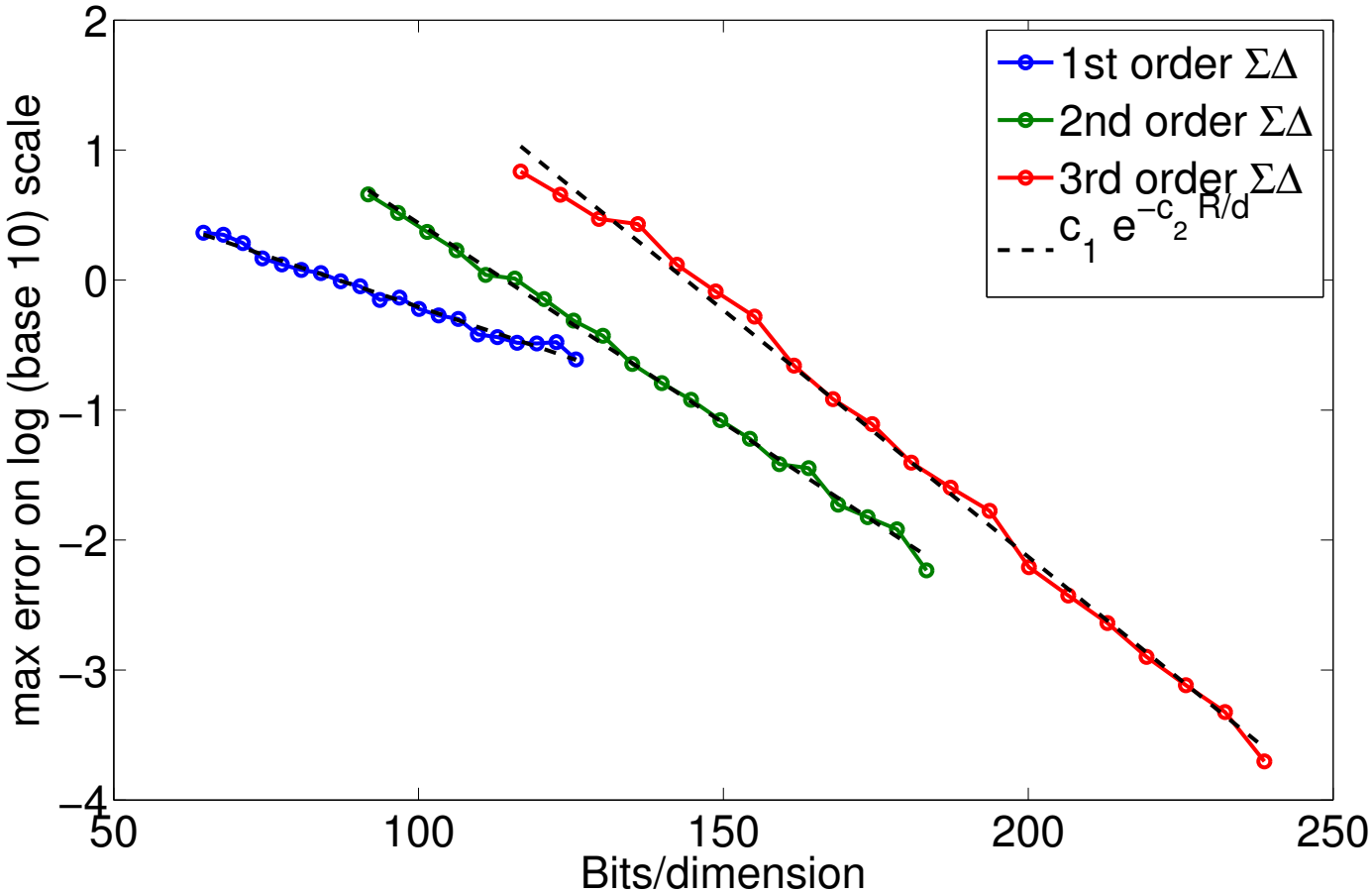


Figure : The maximum error versus the number of bits, 1st, 2nd and 3rd order $\Sigma\Delta$. The maximum is over 1000 instances of x from the unit ball of $\mathbb{R}^d$, for $d = 20$.

Near optimal encoding/decoding: Theoretical guarantees

Definition: E is an (r, C, α) frame if

1. $\|Ex\|_\infty \leq 1 \quad \forall x \in B_2^d$
2. $\sigma_d(D^{-r}E) \geq Cm^\alpha$ where D is the difference matrix.

Near optimal encoding/decoding: Theoretical guarantees

Definition: E is an (r, C, α) frame if

1. $\|Ex\|_\infty \leq 1 \quad \forall x \in B_2^d$
2. $\sigma_d(D^{-r}E) \geq Cm^\alpha$ where D is the difference matrix.

Iwen, S.(2013)
Let $E : \mathbb{R}^d \mapsto \mathbb{R}^m$ be an (r, C, α) frame, with $\alpha > 1$.

There exists a universal (linear) encoding scheme, using at most $\mathcal{R}$ bits, and an associated (linear) reconstruction scheme such that for all $x \in B_2^d$:

$$\|x - \hat{x}\|_2 \lesssim 2^{-c\frac{\mathcal{R}}{d}}$$

Near optimal encoding/decoding: Theoretical guarantees

Definition: E is an (r, C, α) frame if

1. $\|Ex\|_\infty \leq 1 \quad \forall x \in B_2^d$
2. $\sigma_d(D^{-r}E) \geq Cm^\alpha$ where D is the difference matrix.

Iwen, S.(2013)
Let $E : \mathbb{R}^d \mapsto \mathbb{R}^m$ be an (r, C, α) frame, with $\alpha > 1$.

There exists a universal (linear) encoding scheme, using at most $\mathcal{R}$ bits, and an associated (linear) reconstruction scheme such that for all $x \in B_2^d$:

$$\|x - \hat{x}\|_2 \lesssim 2^{-c\frac{\mathcal{R}}{d}}$$

Note: All the frames from the previous theorems are (r, C, α) frames.

Note: Not trivial to show that a frame is (r, C, α).

Proof

First, control the error via (almost) the same error bounds obtained in the absence of encoding.

Second, control the bit-rate by ensuring a "small discrete range"

Proof

First, control the error via (almost) the same error bounds obtained in the absence of encoding.

Second, control the bit-rate by ensuring a "small discrete range"

Proof

Recall: We encode via $BD^{-r}q$.

And decode via $x^\sharp := \arg\min \|BD^{-r}q - BD^{-r}Ex\|_2$

Equivalently: $x^\sharp = GBD^{-r}q$ with $G := (BD^{-r}E)^\dagger$

So the error satisfies: $\|x - \hat{x}\|_2 \leq \|(BD^{-r}E)^\dagger\|_{2\to 2}\|Bu\|_2$

Again, we must bound the smallest singular value of a random matrix:
$\sigma_{min}((BD^{-r}E))$

Proof

Recall: We encode via $BD^{-r}q$.

And decode via $x^\sharp := \arg\min \|BD^{-r}q - BD^{-r}Ex\|_2$

Equivalently: $x^\sharp = GBD^{-r}q$ with $G := (BD^{-r}E)^\dagger$

So the error satisfies: $\|x - \hat{x}\|_2 \leq \|(BD^{-r}E)^\dagger\|_{2\to 2}\|Bu\|_2$

Again, we must bound the smallest singular value of a random matrix:
$\sigma_{min}((BD^{-r}E))$

Proof

We need

Johnson, Lindenstrauss (1984)
Let $\mathcal{S}$ be a set of N points in $\mathbb{R}^m$ and let $\ell \gtrsim \log(N)/\varepsilon^2$. Projecting S onto a random ℓ-dimensional subspace preserves pairwise distances, up to $1 \pm \varepsilon$ (with positive probability).

Achlioptas (2001)
$\ell \geq \frac{4\ln N + 2\ln(1/p)}{\epsilon^2/2 - \epsilon^3/3} \implies B$ is a JL-embedding with probability $> 1 - p$.

Apply JL lemma to a ρ-net of the unit sphere of the range of $D^{-r}E$ with $\rho \approx \varepsilon$. There exists such a net with fewer than $(3/\rho)^d$ points.

So we can show: $\ell \geq \frac{48d\ln(12/\varepsilon) + 24\ln(1/p)}{\varepsilon^2} \implies B$ is an ε-isometry on the range of $D^{-r}E$ with probability $> 1 - p$.

Proof

We need

Johnson, Lindenstrauss (1984)
Let $\mathcal{S}$ be a set of N points in $\mathbb{R}^m$ and let $\ell \gtrsim \log(N)/\varepsilon^2$. Projecting S onto a random ℓ-dimensional subspace preserves pairwise distances, up to $1 \pm \varepsilon$ (with positive probability).

Achlioptas (2001)
$\ell \geq \frac{4\ln N + 2\ln(1/p)}{\epsilon^2/2 - \epsilon^3/3} \implies B$ is a JL-embedding with probability $> 1 - p$.

Apply JL lemma to a ρ-net of the unit sphere of the range of $D^{-r}E$ with $\rho \approx \varepsilon$. There exists such a net with fewer than $(3/\rho)^d$ points.

So we can show: $\ell \geq \frac{48d\ln(12/\varepsilon) + 24\ln(1/p)}{\varepsilon^2} \implies B$ is an ε-isometry on the range of $D^{-r}E$ with probability $> 1 - p$.

Proof

Consequently:

$$\sigma_d(BD^{-r}E) \geq (1-\varepsilon)\sigma_d(D^{-r}E)$$

So we have the error bound

$$\|x-\hat{x}\|_2 = \|(BD^{-r}E)^{\dagger}Bu\|_2 \leq \frac{\|Bu\|_2}{\sigma_d(BD^{-r}E)} \leq \frac{m^{1/2}\|u\|_2}{(1-\varepsilon)\sigma_d(D^{-r}E)}.$$

Since E is an (r, C, α) frame:

- $\sigma_d(D^{-r}E) > Cm^{\alpha}, \quad \alpha > 1.$
- $\|Ex\|_\infty \leq 1 \implies \|u\|_2 \leq C_{\Sigma\Delta}(r)\sqrt{m}.$

Putting it all together:

$$\|x - \hat{x}\|_2 \leq c\frac{m^{1-\alpha}}{1-\varepsilon}.$$

Proof

Consequently:

$$\sigma_d(BD^{-r}E) \geq (1-\varepsilon)\sigma_d(D^{-r}E)$$

So we have the error bound

$$\|x-\hat{x}\|_2 = \|(BD^{-r}E)^{\dagger}Bu\|_2 \leq \frac{\|Bu\|_2}{\sigma_d(BD^{-r}E)} \leq \frac{m^{1/2}\|u\|_2}{(1-\varepsilon)\sigma_d(D^{-r}E)}.$$

Since E is an (r, C, α) frame:

- $\sigma_d(D^{-r}E) > Cm^{\alpha}, \quad \alpha > 1.$
- $\|Ex\|_{\infty} \leq 1 \implies \|u\|_2 \leq C_{\Sigma\Delta}(r)\sqrt{m}.$

Putting it all together:

$$\|x-\hat{x}\|_2 \leq c\frac{m^{1-\alpha}}{1-\varepsilon}.$$

Proof

Consequently:

$$\sigma_d(BD^{-r}E) \geq (1-\varepsilon)\sigma_d(D^{-r}E)$$

So we have the error bound

$$\|x-\hat{x}\|_2 = \|(BD^{-r}E)^{\dagger}Bu\|_2 \leq \frac{\|Bu\|_2}{\sigma_d(BD^{-r}E)} \leq \frac{m^{1/2}\|u\|_2}{(1-\varepsilon)\sigma_d(D^{-r}E)}.$$

Since E is an (r, C, α) frame:

- $\sigma_d(D^{-r}E) > Cm^{\alpha}, \quad \alpha > 1.$
- $\|Ex\|_{\infty} \leq 1 \implies \|u\|_2 \leq C_{\Sigma\Delta}(r)\sqrt{m}.$

Putting it all together:

$$\boxed{\|x-\hat{x}\|_2 \leq c\frac{m^{1-\alpha}}{1-\varepsilon}.}$$

Proof

We controlled the error.

$$\|x - \hat{x}\|_2 \leq c \cdot \frac{m^{1-\alpha}}{(1-\varepsilon)}.$$

Where is the encoding?

Observe that

$$\begin{aligned} q_k \in \mathcal{A} := \{\pm 1\} &\implies (D^{-1}q)_k \in \{-k, ..., k\} \\ &\implies (D^{-r}q)_k \in \{-k^r, ..., k^r\} \\ &\implies (BD^{-r}q)_k \in \{-m^{r+1}, ..., m^{r+1}\} \end{aligned}$$

Each entry of $BD^{-r}q$ can be encoded with $(r+1)\log_2(m)$ bits.

Finally, $BD^{-r}q$ can be encoded with $\boxed{\approx d(r+1)\log_2(m)}$ bits. So we are done.

Proof

We controlled the error.

$$\|x - \hat{x}\|_2 \leq c \cdot \frac{m^{1-\alpha}}{(1-\varepsilon)}.$$

Where is the encoding?

Observe that

$$\begin{aligned} q_k \in \mathcal{A} := \{\pm 1\} &\implies (D^{-1}q)_k \in \{-k, ..., k\} \\ &\implies (D^{-r}q)_k \in \{-k^r, ..., k^r\} \\ &\implies (BD^{-r}q)_k \in \{-m^{r+1}, ..., m^{r+1}\} \end{aligned}$$

Each entry of $BD^{-r}q$ can be encoded with $(r+1)\log_2(m)$ bits.

Finally, $BD^{-r}q$ can be encoded with $\approx d(r+1)\log_2(m)$ bits. So we are done.

Proof

We controlled the error.

$$\|x - \hat{x}\|_2 \leq c \cdot \frac{m^{1-\alpha}}{(1-\varepsilon)}.$$

Where is the encoding?

Observe that

$$\begin{aligned} q_k \in \mathcal{A} := \{\pm 1\} &\implies (D^{-1}q)_k \in \{-k, ..., k\} \\ &\implies (D^{-r}q)_k \in \{-k^r, ..., k^r\} \\ &\implies (BD^{-r}q)_k \in \{-m^{r+1}, ..., m^{r+1}\} \end{aligned}$$

Each entry of $BD^{-r}q$ can be encoded with $(r+1)\log_2(m)$ bits.

Finally, $BD^{-r}q$ can be encoded with $\approx d(r+1)\log_2(m)$ bits. So we are done.

Proof

We controlled the error.

$$\|x - \hat{x}\|_2 \leq c \cdot \frac{m^{1-\alpha}}{(1-\varepsilon)}.$$

Where is the encoding?

Observe that

$$\begin{aligned} q_k \in \mathcal{A} := \{\pm 1\} &\implies (D^{-1}q)_k \in \{-k, ..., k\} \\ &\implies (D^{-r}q)_k \in \{-k^r, ..., k^r\} \\ &\implies (BD^{-r}q)_k \in \{-m^{r+1}, ..., m^{r+1}\} \end{aligned}$$

Each entry of $BD^{-r}q$ can be encoded with $(r+1)\log_2(m)$ bits.

Finally, $BD^{-r}q$ can be encoded with $\approx d(r+1)\log_2(m)$ bits. So we are done.

Proof

We controlled the error.

$$\|x - \hat{x}\|_2 \leq c \cdot \frac{m^{1-\alpha}}{(1-\varepsilon)}.$$

Where is the encoding?

Observe that

$$\begin{aligned} q_k \in \mathcal{A} := \{\pm 1\} &\implies (D^{-1}q)_k \in \{-k, ..., k\} \\ &\implies (D^{-r}q)_k \in \{-k^r, ..., k^r\} \\ &\implies (BD^{-r}q)_k \in \{-m^{r+1}, ..., m^{r+1}\} \end{aligned}$$

Each entry of $BD^{-r}q$ can be encoded with $(r+1)\log_2(m)$ bits.

Finally, $BD^{-r}q$ can be encoded with $\approx d(r+1)\log_2(m)$ bits. So we are done.

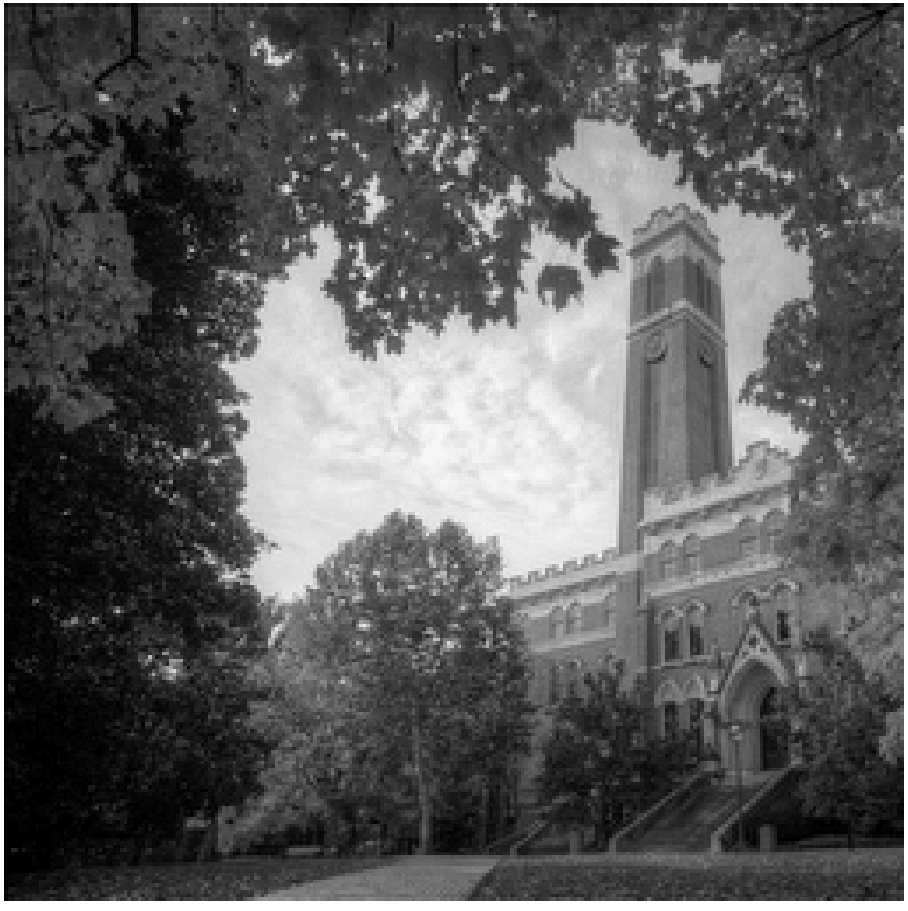
Stylized example

A stylized image acquisition example:

- Collect inner products of the image (viewed as a vector) with random vectors of ± 1 entries.
- Quantize them using 1-bit rth order $\Sigma\Delta$ quantization.
- Randomly encode them as described (here $m = 400d, \ell = 1.5d$).
- Decode as described.

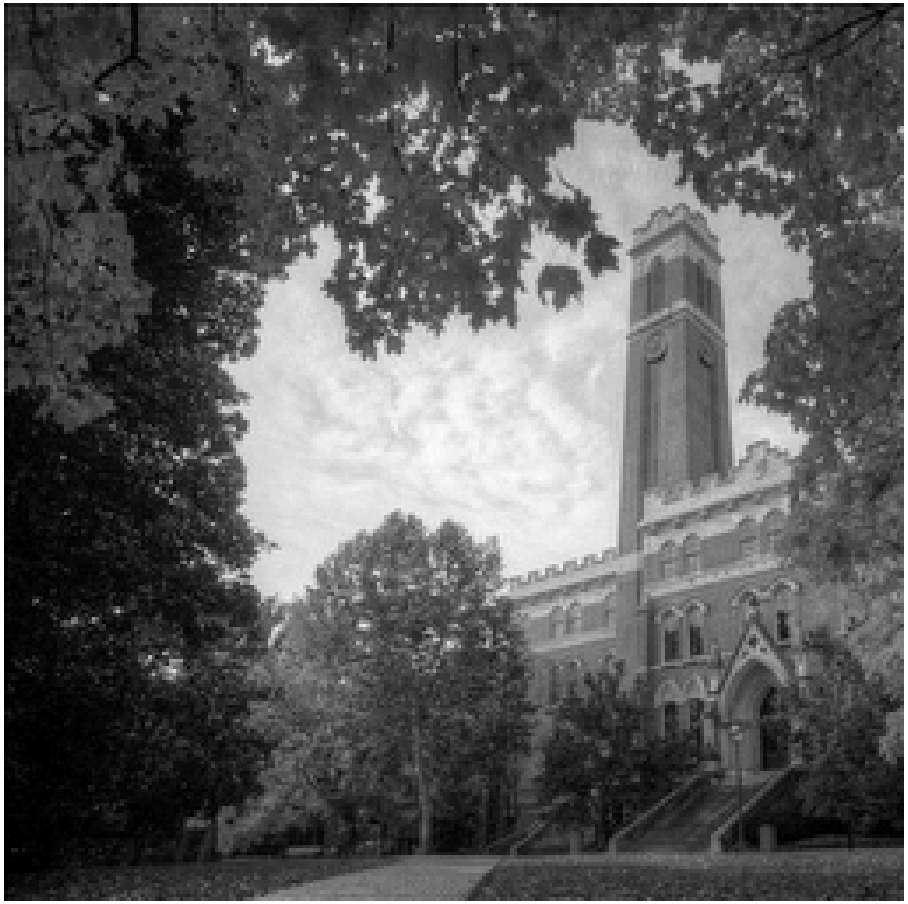
Stylized example

Original



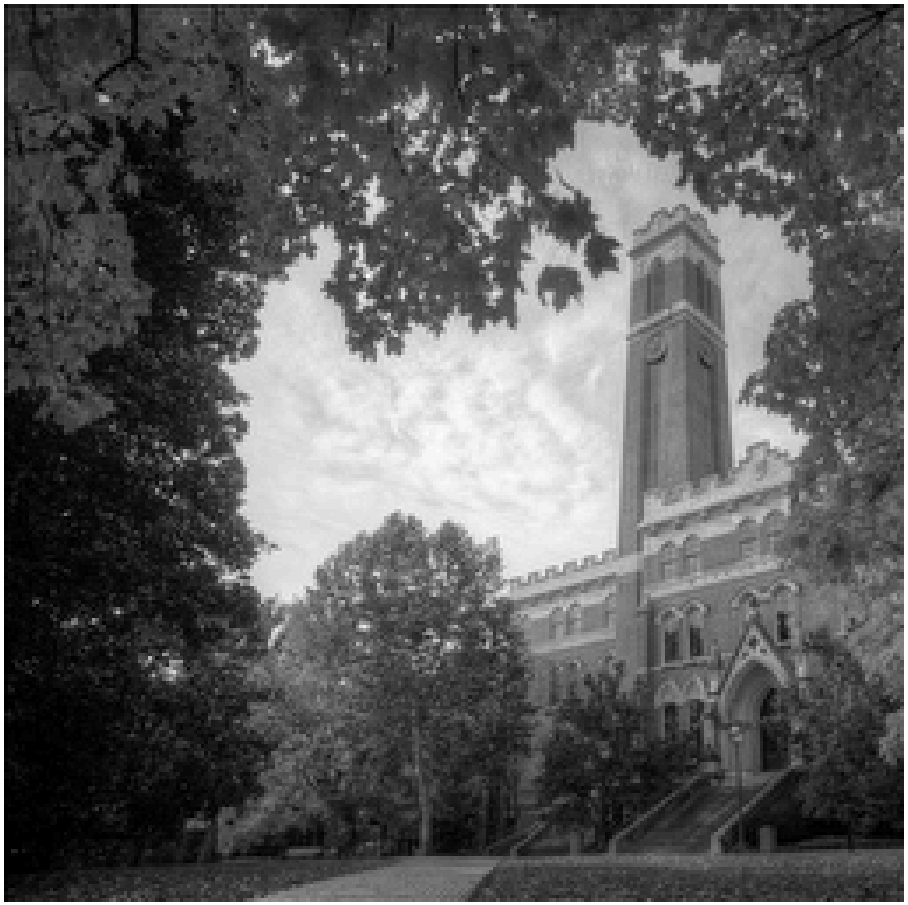
Stylized example

2rd order sig–del,0.17bpm and 1.1% rel. error



Stylized example

3rd order sig–del,0.23bpm and 0.04% rel. error



Properties of the proposed encoding:

Universality: The encoder is independent of the measurement frame. Works with high probability on any E.

Inherits robustness to circuit imperfections: preserves the robustness of $\Sigma\Delta$ quantization to hardware imperfections.

Dimensionality reduction: Requires solving a least squares problem in $\ell \approx d$ dimensions, rather than $m \gg d$ dimensions.

Compressed sensing: A quick introduction

Acquisition paradigm: applies to the ubiquitous class of k-sparse or almost sparse signals in $\mathbb{R}^d$, $k \ll d$.

Sampling scheme: prescribes collecting m linear measurements via inner products with random vectors, $d \gg m \gtrsim k \log d/k$.

$$y = Ax + e.$$

Reconstruction scheme: Recover the sparse signal, say x via non-linear algorithms, e.g.

$$\tilde{x} = \arg\min_z \|z\|_1 \text{ subject to } \|Az - y\|_2 \leq \|e\|_2.$$

Recovery guarantees: With high probability, when A is random, drawn from an appropriate distribution (Candes, Romberg, Tao 2005) and (Donoho 2005)

$$m \gtrsim k \log d/k \implies \|\tilde{x} - x\|_2 \leq C(\|e\|_2 + \frac{\|x - x_k\|_1}{\sqrt{k}}).$$

Compressed sensing: A quick introduction

Acquisition paradigm: applies to the ubiquitous class of k-sparse or almost sparse signals in $\mathbb{R}^d$, $k \ll d$.

Sampling scheme: prescribes collecting m linear measurements via inner products with random vectors, $d \gg m \gtrsim k \log d/k$.

$$y = Ax + e.$$

Reconstruction scheme: Recover the sparse signal, say x via non-linear algorithms, e.g.

$$\tilde{x} = \arg\min_z \|z\|_1 \text{ subject to } \|Az - y\|_2 \leq \|e\|_2.$$

Recovery guarantees: With high probability, when A is random, drawn from an appropriate distribution (Candes, Romberg, Tao 2005) and (Donoho 2005)

$$m \gtrsim k \log d/k \implies \|\tilde{x} - x\|_2 \leq C(\|e\|_2 + \frac{\|x - x_k\|_1}{\sqrt{k}}).$$

Compressed sensing: A quick introduction

Acquisition paradigm: applies to the ubiquitous class of k-sparse or almost sparse signals in $\mathbb{R}^d$, $k \ll d$.

Sampling scheme: prescribes collecting m linear measurements via inner products with random vectors, $d \gg m \gtrsim k \log d/k$.

$$y = Ax + e.$$

Reconstruction scheme: Recover the sparse signal, say x via non-linear algorithms, e.g.

$$\tilde{x} = \arg\min_z \|z\|_1 \text{ subject to } \|Az - y\|_2 \leq \|e\|_2.$$

Recovery guarantees: With high probability, when A is random, drawn from an appropriate distribution (Candes, Romberg, Tao 2005) and (Donoho 2005)

$$m \gtrsim k \log d/k \implies \|\tilde{x} - x\|_2 \leq C(\|e\|_2 + \frac{\|x - x_k\|_1}{\sqrt{k}}).$$

Compressed sensing: A quick introduction

Acquisition paradigm: applies to the ubiquitous class of k-sparse or almost sparse signals in $\mathbb{R}^d$, $k \ll d$.

Sampling scheme: prescribes collecting m linear measurements via inner products with random vectors, $d \gg m \gtrsim k \log d/k$.

$$y = Ax + e.$$

Reconstruction scheme: Recover the sparse signal, say x via non-linear algorithms, e.g.

$$\tilde{x} = \arg\min_z \|z\|_1 \text{ subject to } \|Az - y\|_2 \leq \|e\|_2.$$

Recovery guarantees: With high probability, when A is random, drawn from an appropriate distribution (Candes, Romberg, Tao 2005) and (Donoho 2005)

$$m \gtrsim k \log d/k \implies \|\tilde{x} - x\|_2 \leq C(\|e\|_2 + \frac{\|x - x_k\|_1}{\sqrt{k}}).$$

Compressed sensing and quantization

Quantization: Theory not fully developed.

Notable exceptions:

- Güntürk, Lammers, Powell, S., Yılmaz (2013)
- Baraniuk, Foucart, Needell, Plan, Wootters (2014)
- Güntürk, Chou (2014)
- S., Wang, Yılmaz (2014)

Standard approach: Quantize with a scalar quantizer

$$y_i \longrightarrow q_i \in \mathcal{A} := \{\pm\delta/2, \pm 3\delta/2, \pm 5\delta/2, ...\}.$$

Robust recovery theorems (CRT, D) guarantee that with ℓ_1 minimization

$$\|\tilde{x} - x\|_2 \leq C\delta.$$

Issue: Error does not decay as we take more measurements!!!

Compressed sensing and quantization

Quantization: Theory not fully developed.

Notable exceptions:

- Güntürk, Lammers, Powell, S., Yılmaz (2013)
- Baraniuk, Foucart, Needell, Plan, Wootters (2014)
- Güntürk, Chou (2014)
- S., Wang, Yılmaz (2014)

Standard approach: Quantize with a scalar quantizer

$$y_i \longrightarrow q_i \in \mathcal{A} := \{\pm\delta/2, \pm 3\delta/2, \pm 5\delta/2, ...\}.$$

Robust recovery theorems (CRT, D) guarantee that with ℓ_1 minimization

$$\|\tilde{x} - x\|_2 \leq C\delta.$$

Issue: Error does not decay as we take more measurements!!!

Compressed sensing and quantization

Quantization: Theory not fully developed.

Notable exceptions:

- Güntürk, Lammers, Powell, S., Yılmaz (2013)
- Baraniuk, Foucart, Needell, Plan, Wootters (2014)
- Güntürk, Chou (2014)
- S., Wang, Yılmaz (2014)

Standard approach: Quantize with a scalar quantizer

$$y_i \longrightarrow q_i \in \mathcal{A} := \{\pm\delta/2, \pm 3\delta/2, \pm 5\delta/2, ...\}.$$

Robust recovery theorems (CRT, D) guarantee that with ℓ_1 minimization

$$\|\tilde{x} - x\|_2 \leq C\delta.$$

Issue: Error does not decay as we take more measurements!!!

Compressed sensing and quantization

Quantization: Theory not fully developed.

Notable exceptions:

- Güntürk, Lammers, Powell, S., Yılmaz (2013)
- Baraniuk, Foucart, Needell, Plan, Wootters (2014)
- Güntürk, Chou (2014)
- S., Wang, Yılmaz (2014)

Standard approach: Quantize with a scalar quantizer

$$y_i \longrightarrow q_i \in \mathcal{A} := \{\pm\delta/2, \pm 3\delta/2, \pm 5\delta/2, ...\}.$$

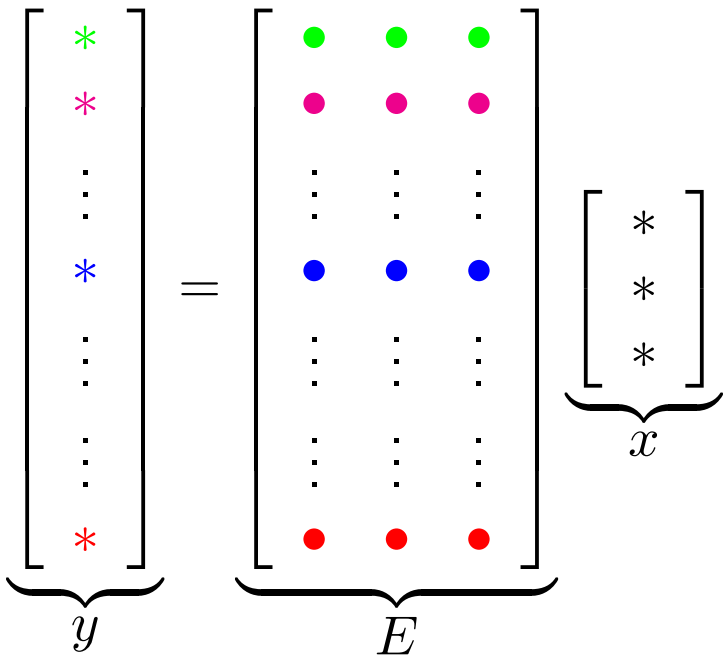
Robust recovery theorems (CRT, D) guarantee that with ℓ_1 minimization

$$\|\tilde{x} - x\|_2 \leq C\delta.$$

Issue: Error does not decay as we take more measurements!!!

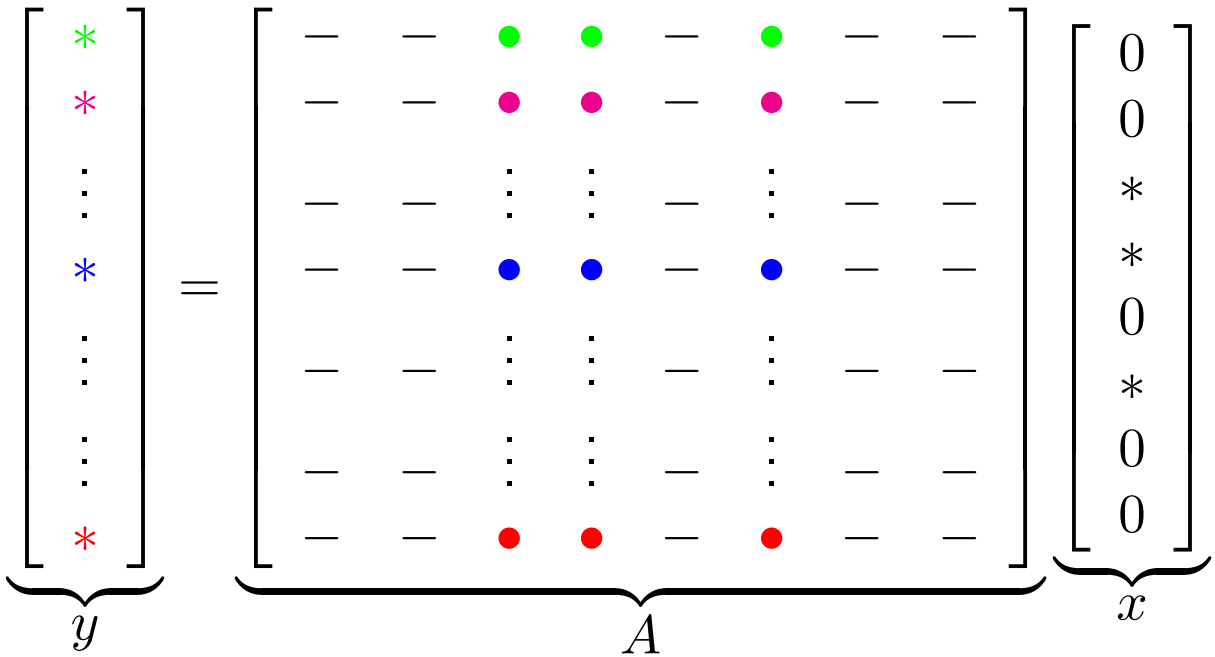
Compressed sensing: Why should $\Sigma\Delta$ work?

Recall



Compressed sensing: Why should $\Sigma\Delta$ work?

Recall



If we know the support, this is a frame quantization problem!!

Compressed sensing: Two-stage reconstruction

Step 1: Use robust recovery, e.g., ℓ_1-minimization to recover support T.

$$\min \|z\|_1 \text{ subject to } \|Az - q\|_2 \leq \sqrt{m}\delta$$

Step 2: Once the support is known, use the Sobolev dual F of the support sub-matrix $E := A_T$ to recover x from q.

Compressed sensing: Two-stage reconstruction

Step 1: Use robust recovery, e.g., ℓ_1-minimization to recover support T.

$$\min \|z\|_1 \text{ subject to } \|Az - q\|_2 \leq \sqrt{m}\delta$$

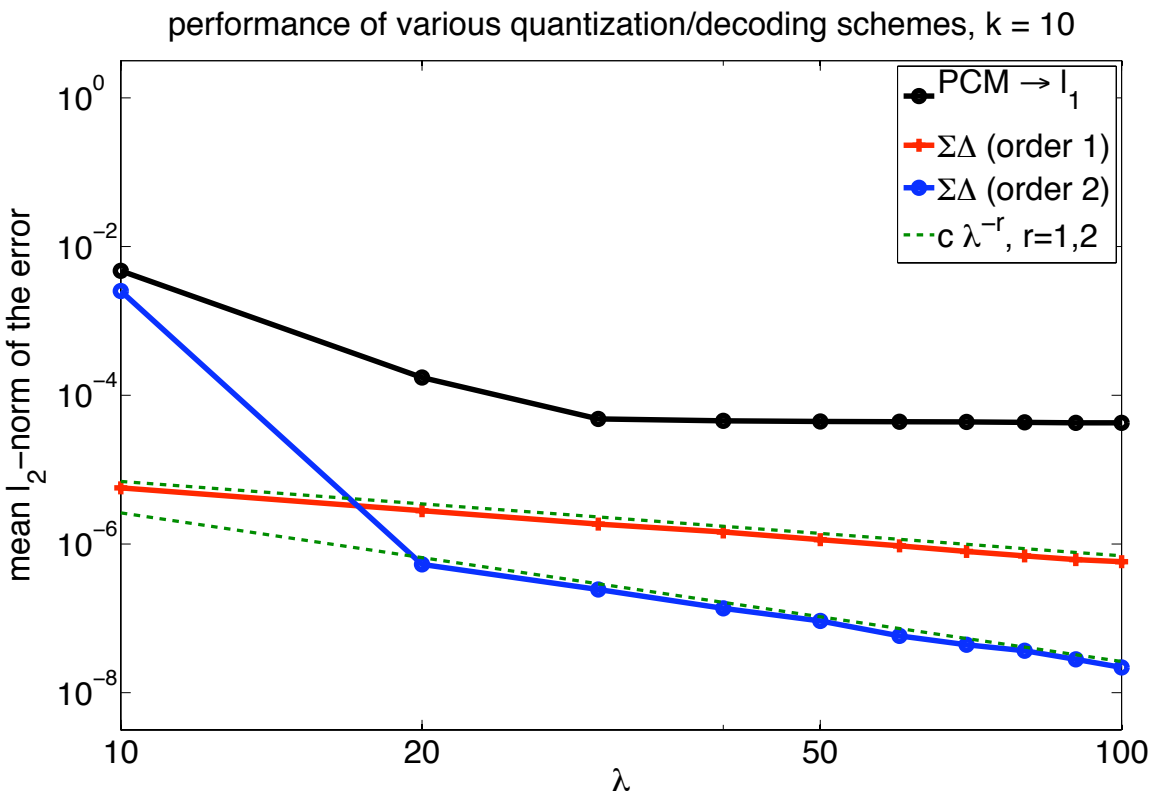
Step 2: Once the support is known, use the Sobolev dual F of the support sub-matrix $E := A_T$ to recover x from q.

Compressed sensing: Two-stage reconstruction

Step 1: Use robust recovery, e.g., ℓ_1-minimization to recover support T.

$$\min \|z\|_1 \text{ subject to } \|Az - q\|_2 \leq \sqrt{m}\delta$$

Step 2: Once the support is known, use the Sobolev dual F of the support sub-matrix $E := A_T$ to recover x from q.



Compressed sensing: Two-stage reconstruction

(Güntürk, Lammers, Powell, S., Yılmaz 2013)
(Krahmer, S., Yılmaz 2013)

Let A be an $m \times d$ matrix with i.i.d. sub-Gaussian entries with

$$m \gtrsim k(\log d)^{1/(1-\alpha)}, \quad \alpha \in (0,1).$$

With high probability on the draw of A, we have:

For every k-sparse x, such that $\min_{j \in \mathrm{supp}(x)} |x_j| \geq C\delta$,

$$\|x - \hat{x}_{\Sigma\Delta}\|_2 \lesssim_r \left(\frac{m}{k}\right)^{-\alpha(r-\frac{1}{2})} \delta.$$

Compressed sensing: One-stage reconstruction

Decode by solving :

$$x^*_{\Sigma\Delta} := \arg\min \|z\|_1 \text{ subject to} \|D^{-r}(Az - q)\|_2 \leq C(r)\delta\sqrt{m}$$

(S., Wang, Yılmaz 2014)

Let A be an $m \times d$ matrix with i.i.d. sub-Gaussian entries with

$$m \gtrsim k(\log d)^{1/(1-\alpha)}, \quad \alpha \in (0,1).$$

With high probability on the draw of A, $\forall$ k-sparse x

$$\|x - x^*_{\Sigma\Delta}\|_2 \lesssim_r \left(\frac{m}{k}\right)^{-\alpha(r-\frac{1}{2})}\delta.$$

Remarks: The one-stage algorithm is stable and robust/ no minimum size condition on x

Compressed sensing: One-stage reconstruction

Decode by solving :

$$x^*_{\Sigma\Delta} := \arg\min \|z\|_1 \text{ subject to} \|D^{-r}(Az - q)\|_2 \leq C(r)\delta\sqrt{m}$$

(S., Wang, Yılmaz 2014)

Let A be an $m \times d$ matrix with i.i.d. sub-Gaussian entries with

$$m \gtrsim k(\log d)^{1/(1-\alpha)}, \quad \alpha \in (0,1).$$

With high probability on the draw of A, $\forall$ k-sparse x

$$\|x - x^*_{\Sigma\Delta}\|_2 \lesssim_r \left(\frac{m}{k}\right)^{-\alpha(r-\frac{1}{2})}\delta.$$

Remarks: The one-stage algorithm is stable and robust/ no minimum size condition on x

Compressed sensing: Encoding + one-stage reconstruction

Encode as in the frame case : $q \mapsto BD^{-r}q$. Requires R bits.

Here: B is an $L \times m$ Bernoulli matrix, $L \approx k \log(d/k)$

Decode by solving :

$$x^*_{\Sigma\Delta} := \arg\min \|z\|_1 \text{ subject to} \|BD^{-r}(Az - q)\|_2 \leq C(r)\delta\sqrt{mL}$$

(S., Wang, Yılmaz 2014)

Let A be an $m \times d$ sub-Gaussian matrix with $m \gtrsim k(\log d)^{1/(1-\alpha)}$.

Let B be as above.

With high probability on the draw of A and B, $\forall$ k-sparse x

$$\|x - x^*_{\Sigma\Delta}\|_2 \lesssim_r \exp\left(-c(r)\frac{R}{k \log d/k}\right)\delta.$$

Remarks: Results are valid for 1-bit quantization.

Compressed sensing: Encoding + one-stage reconstruction

Encode as in the frame case : $q \mapsto BD^{-r}q$. Requires R bits.

Here: B is an $L \times m$ Bernoulli matrix, $L \approx k \log(d/k)$

Decode by solving :

$$x^*_{\Sigma\Delta} := \arg\min \|z\|_1 \text{ subject to} \|BD^{-r}(Az - q)\|_2 \leq C(r)\delta\sqrt{mL}$$

(S., Wang, Yılmaz 2014)

Let A be an $m \times d$ sub-Gaussian matrix with $m \gtrsim k(\log d)^{1/(1-\alpha)}$.

Let B be as above.

With high probability on the draw of A and B, $\forall$ k-sparse x

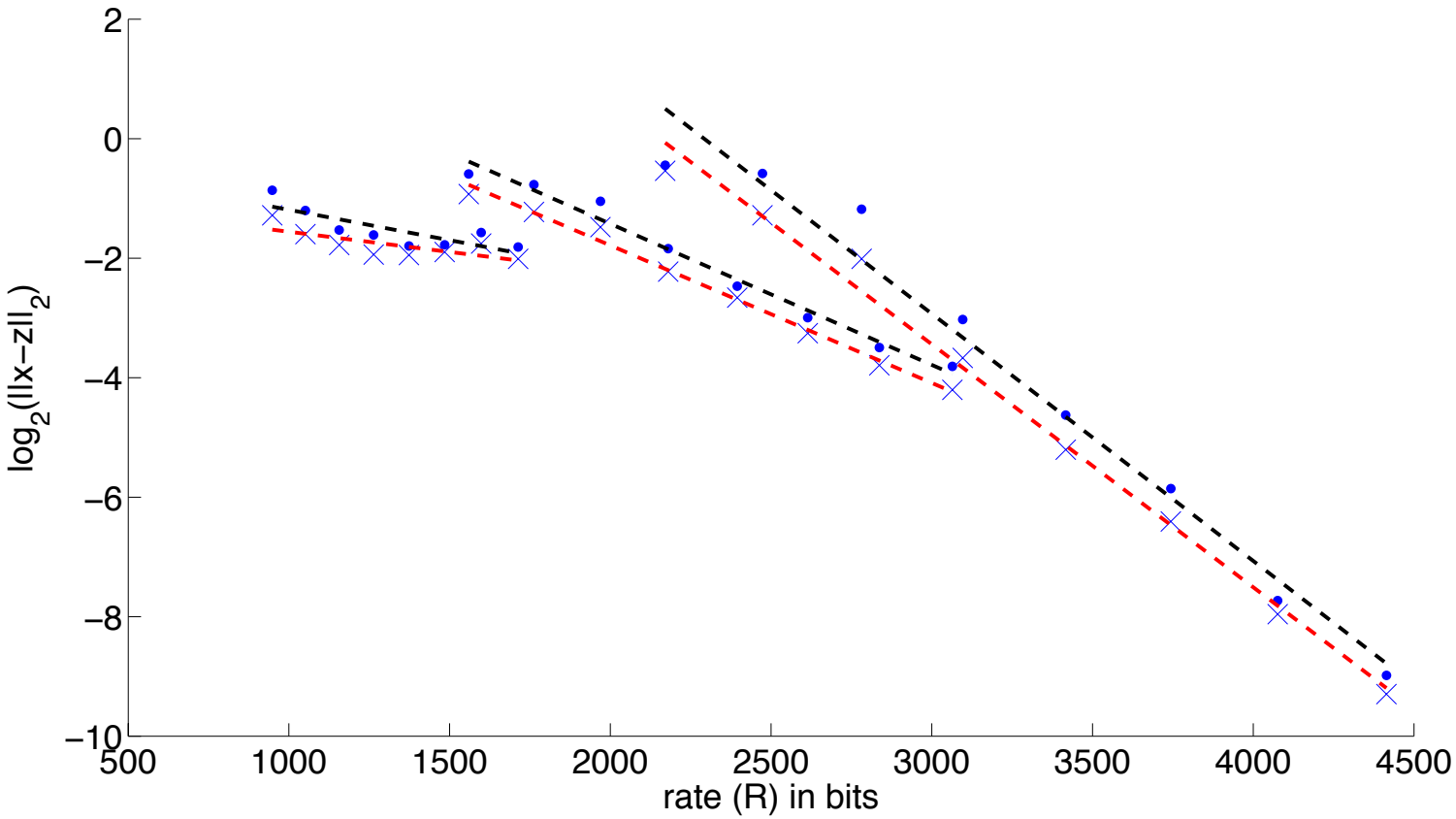
$$\|x - x^*_{\Sigma\Delta}\|_2 \lesssim_r \exp\left(-c(r)\frac{R}{k \log d/k}\right)\delta.$$

Remarks: Results are valid for 1-bit quantization.

Numerical experiments

Here: $d = 1024$, $k = 10$, $\Phi \in \mathbb{R}^{m \times d}$ Bernoulli, $m \in \{100 \cdot 2^p : \ p = 0, \ldots, 7\}$.

Distortion vs. bit budget when we use the new scheme with $\Sigma\Delta$ schemes with $\delta = 0.5$

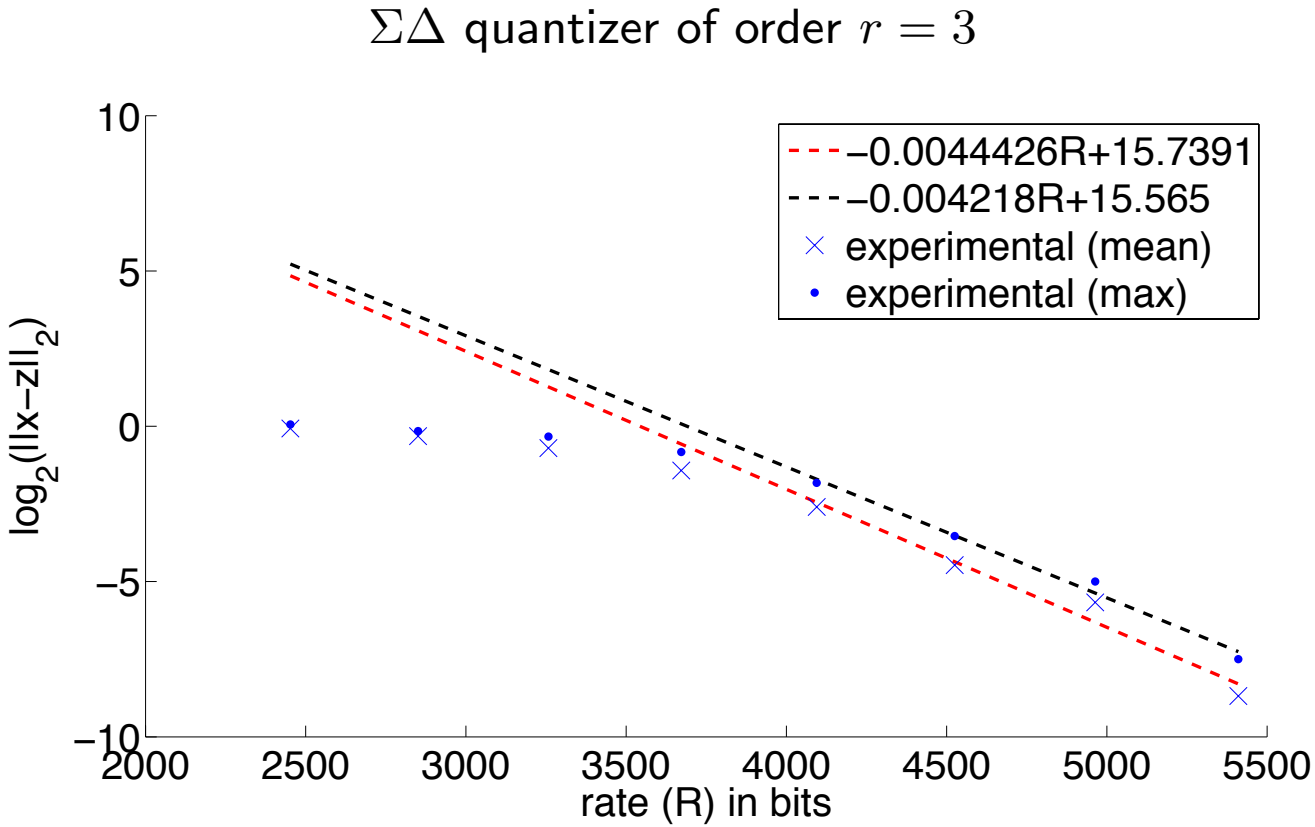


Left: MSQ
Middle: $\Sigma\Delta$ with $r = 1$
Right: $\Sigma\Delta$ with $r = 2$

One-bit compressed sensing

Here: $d = 4000$, $k = 10$, $\Phi \in \mathbb{R}^{m \times d}$, $m \in \{100 \cdot 2^p : \ p = 0, \ldots, 7\}$.

Distortion vs. bit budget when we use the new scheme with one-bit $\Sigma\Delta$ schemes with $\delta = 6$!



Compressed sensing: Concluding remarks

Scalar quantization schemes have physical limitations which in turn limit the best accuracy one can obtain.

Coarse quantization schemes, such as $\Sigma\Delta$ quantization, provide a remedy.

With a fixed quantization alphabet, they provide better error decay than scalar quantization (polynomial/root-exponential vs linear).

With an additional random compression stage, they achieve near-optimal (exponential) accuracy with respect to bit budget while using a fixed, coarse (up to 1-bit) quantization alphabet.