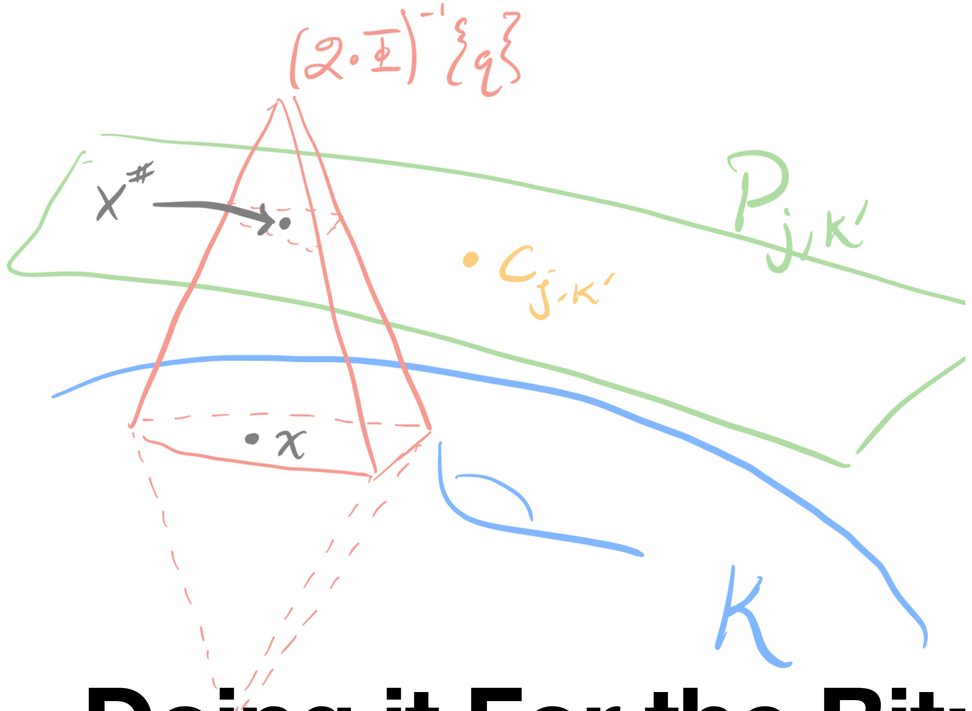
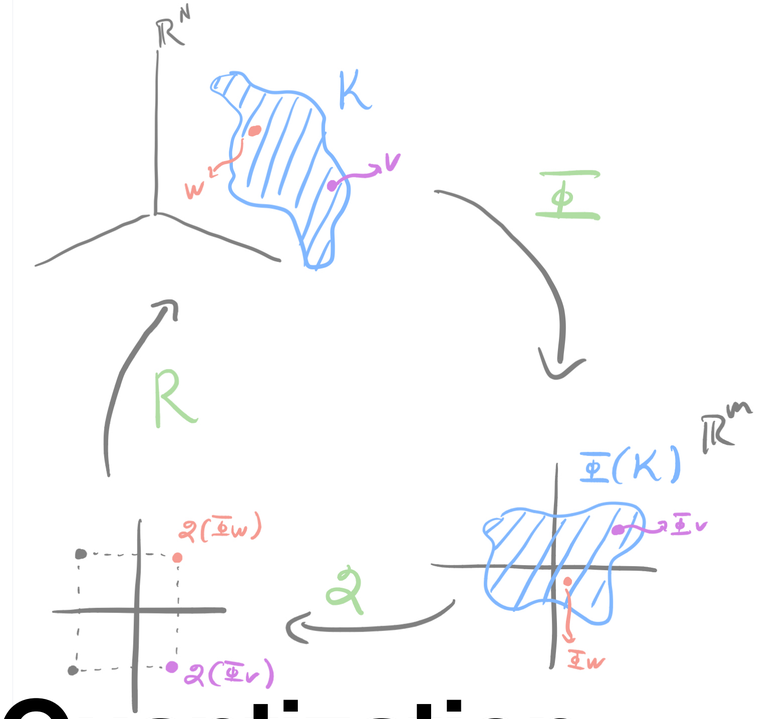
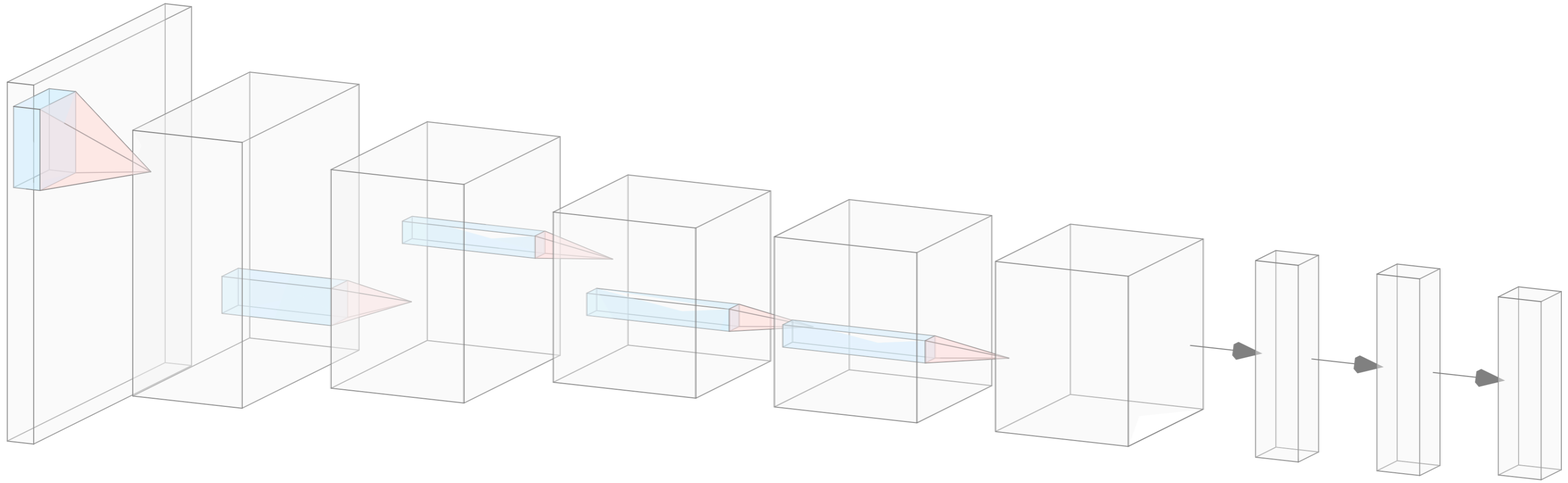




Doing it For the Bit: Applications of Quantization in Data Science & Signal Processing

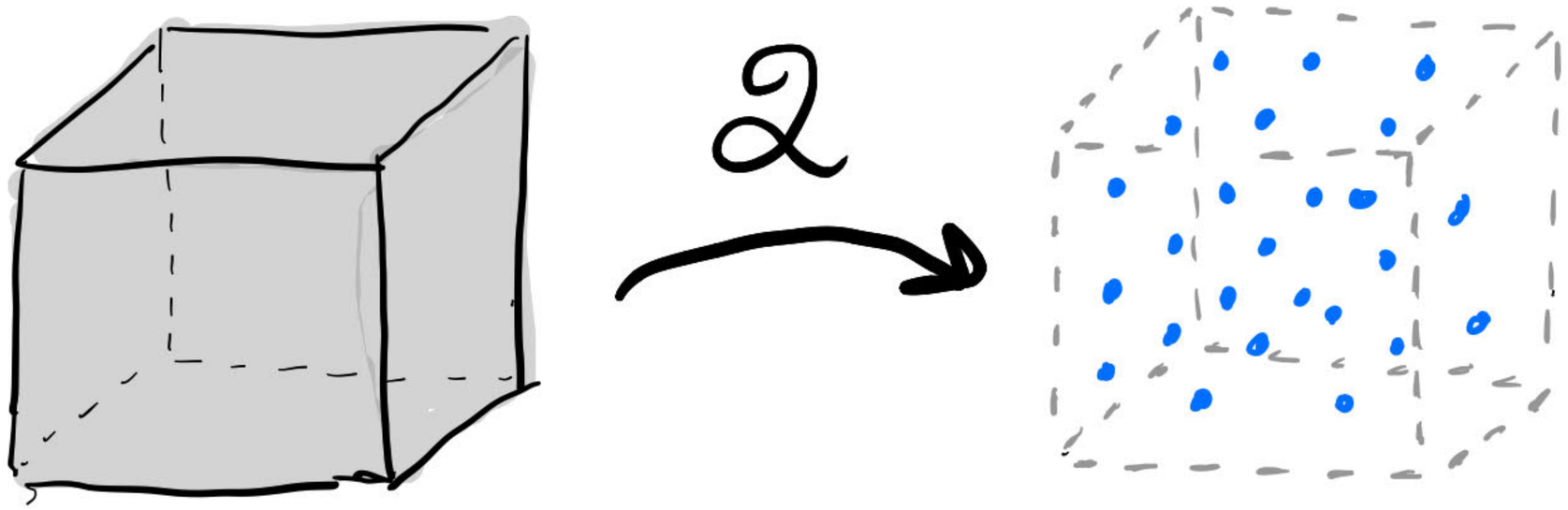
Ph.D. Defense



Eric Lybrand
February 26th, 2021
https://elybrand.github.io/

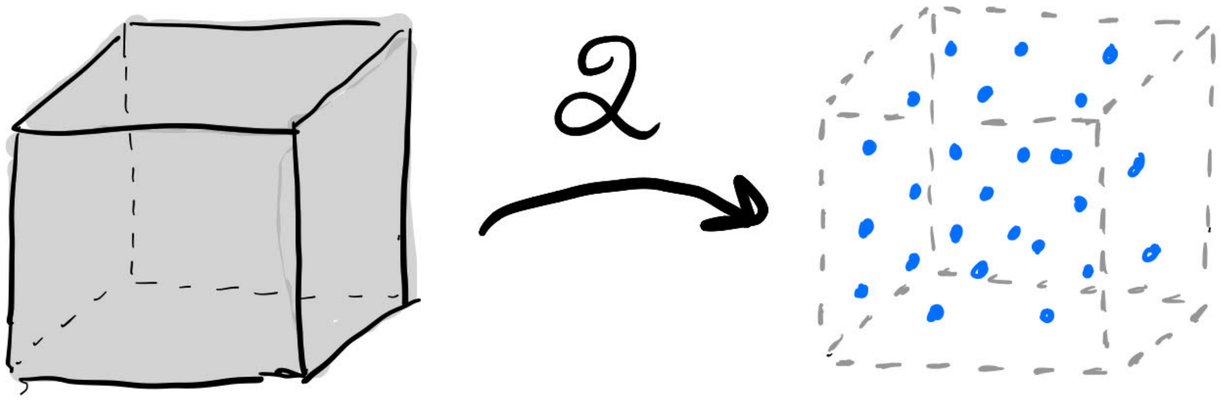
What is Quantization?

What is Quantization?

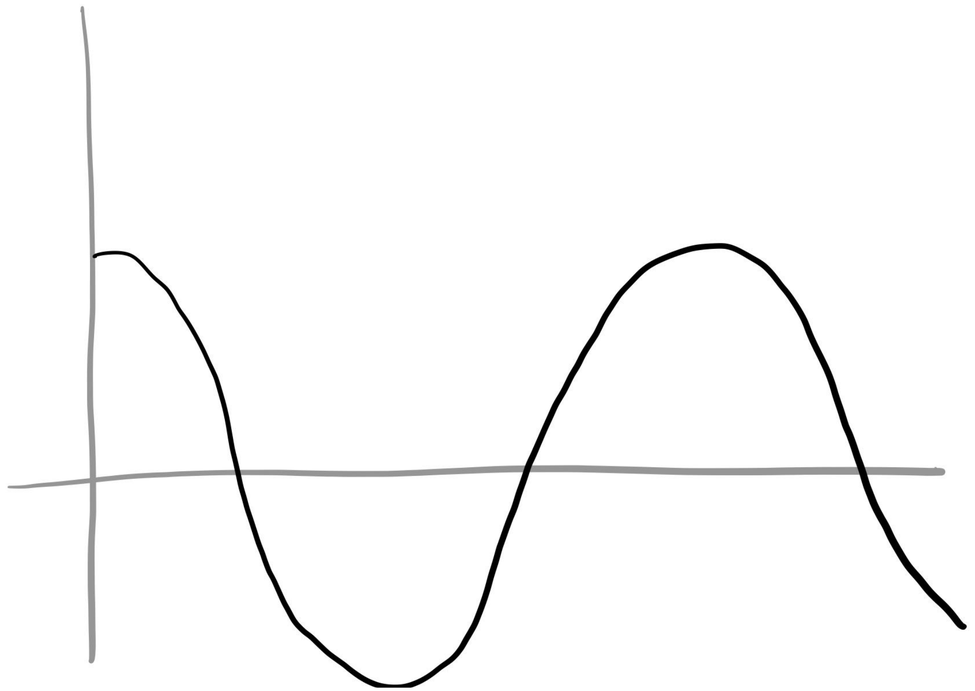


What is Quantization?

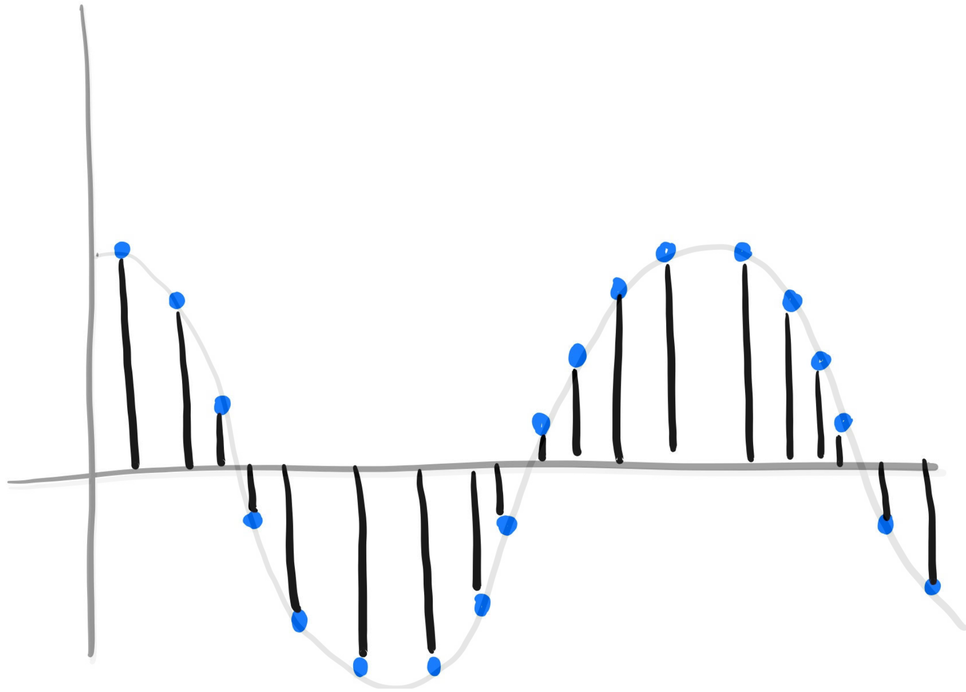
- **Goal:** Approximate the continuum with finitely many discrete values in a **codebook** $\mathcal{A}^m \subset \mathbb{R}^m$, e.g. $\mathcal{A} = \{\pm 1\}$.
  - Construct mapping $\mathcal{Q} : \mathbb{R}^m \to \mathcal{A}^m$
  - Many contexts where you **don't** want to solve $q := \arg\min_{p \in \mathcal{A}^m} \|x - p\|$.



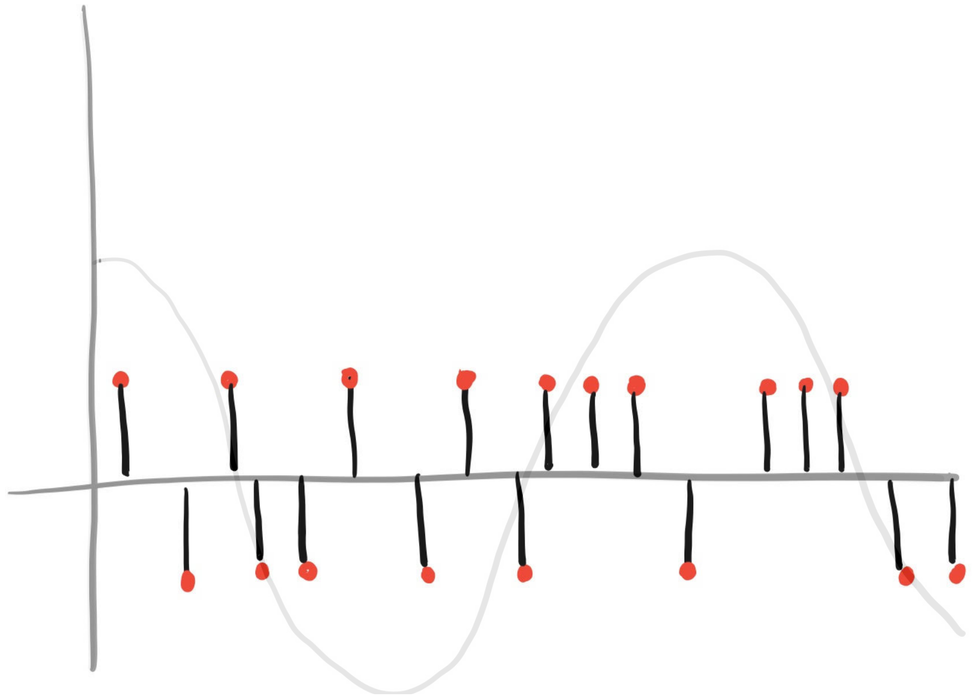
Example: A/D Conversion



Example: A/D Conversion

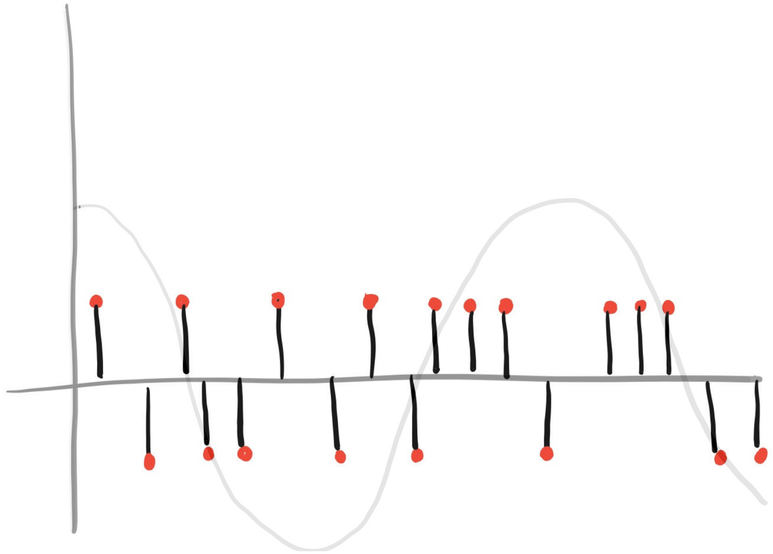


Example: A/D Conversion



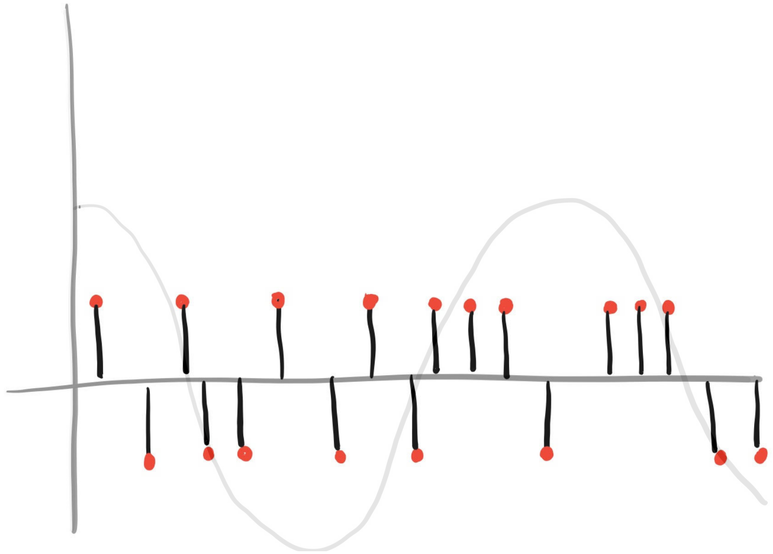
Example: A/D Conversion

- **Goal:** Approximately recover $x \in \mathbb{R}^N$ from quantized measurements $q := \mathcal{Q}(\Phi x) \in \mathscr{A}^m$, $\Phi \in \mathbb{R}^{m \times N}$ **fixed**.
  - **Observation:** if $m \neq N$, then need a decoder $D : \mathscr{A}^m \to \mathbb{R}^N$
  - **Observation**: if $m > N$, and if we had access to Φx, then we could apply left inverse $\Psi \in \mathbb{R}^{N \times m}$ and get $\Psi \Phi x = x$.



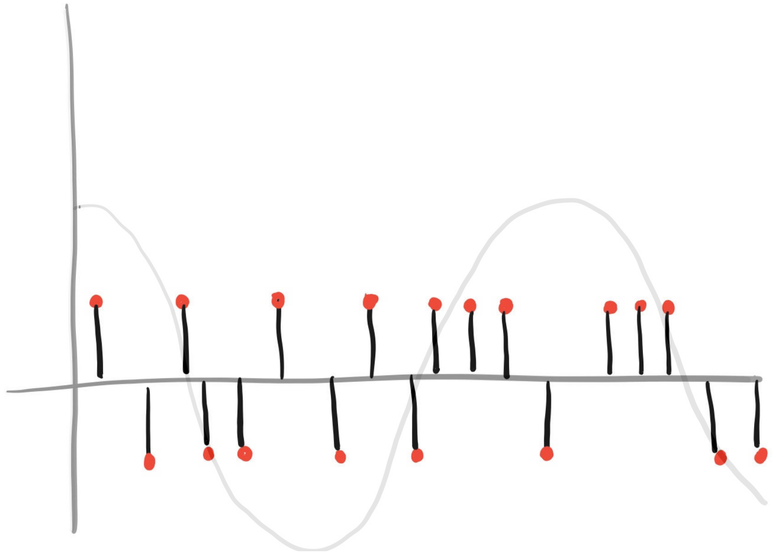
Example: A/D Conversion

- **Goal:** Approximately recover $x \in \mathbb{R}^N$ from quantized measurements $q := \mathcal{Q}(\Phi x) \in \mathcal{A}^m$, $\Phi \in \mathbb{R}^{m \times N}$ **fixed, and** $m \geq N$**.**
  - Choose Ψ with $\Psi\Phi = I$, and construct q so $\mathscr{D}(x, q) = \|\Psi(\Phi x - q)\|_2 = \|x - \Psi q\| \approx 0.$
    - Intuition: larger m gives more “null space” to push error into.



Example: A/D Conversion

- Given Ψ with $\Psi\Phi = I$, construct q so $\mathscr{D}(x, q) = \|\Psi(\Phi x - q)\|_2 = \|x - \Psi q\| \approx 0.$
  - **Choice 1 (MSQ):** $q = \mathrm{round}_{\mathscr{A}}(\Phi x)$
    - $\mathscr{D}(x, q) \gtrsim m^{-1}$ (Goyal, Vetterli, Thao, 1998)

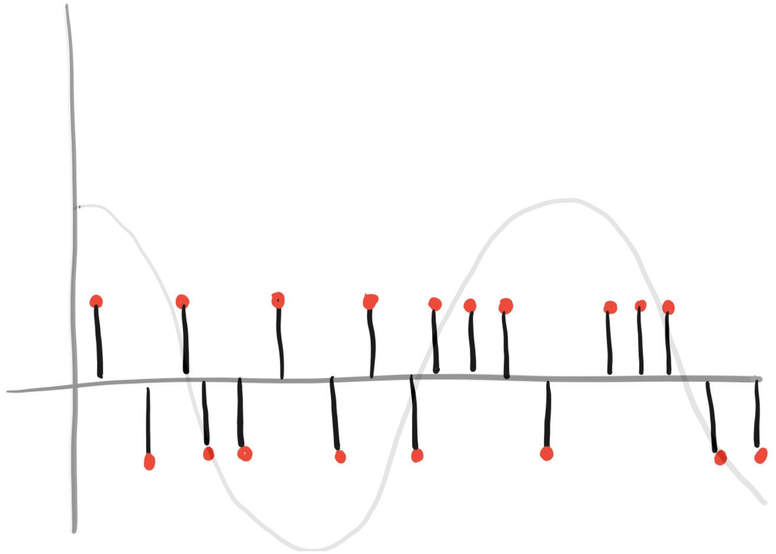


Example: A/D Conversion

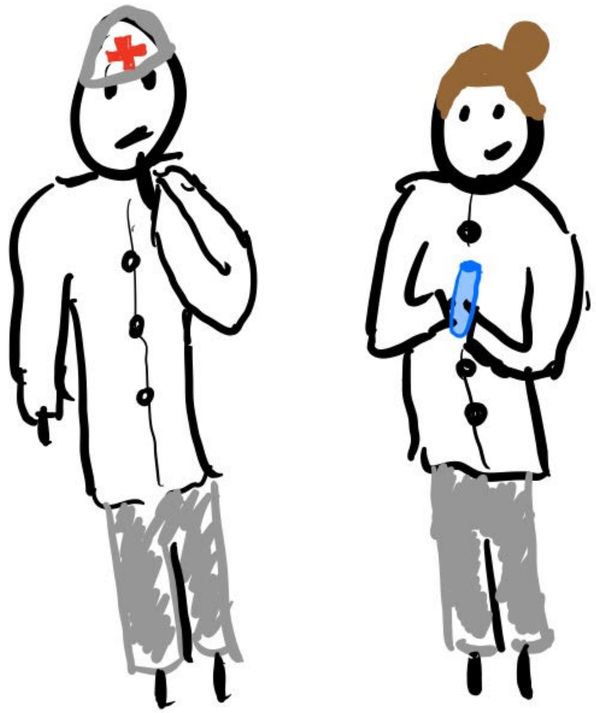
- Given Ψ with $\Psi\Phi = I$, construct q so
  $\mathscr{D}(x, q) = \|\Psi(\Phi x - q)\|_2 = \|x - \Psi q\| \approx 0.$
  - **Choice 1 (MSQ):** $q = \text{round}_{\mathscr{A}}(\Phi x)$
    - $\mathscr{D}(x, q) \gtrsim m^{-1}$ (Goyal, Vetterli, Thao, 1998)
  - **Choice 2** ($\Sigma\Delta$):

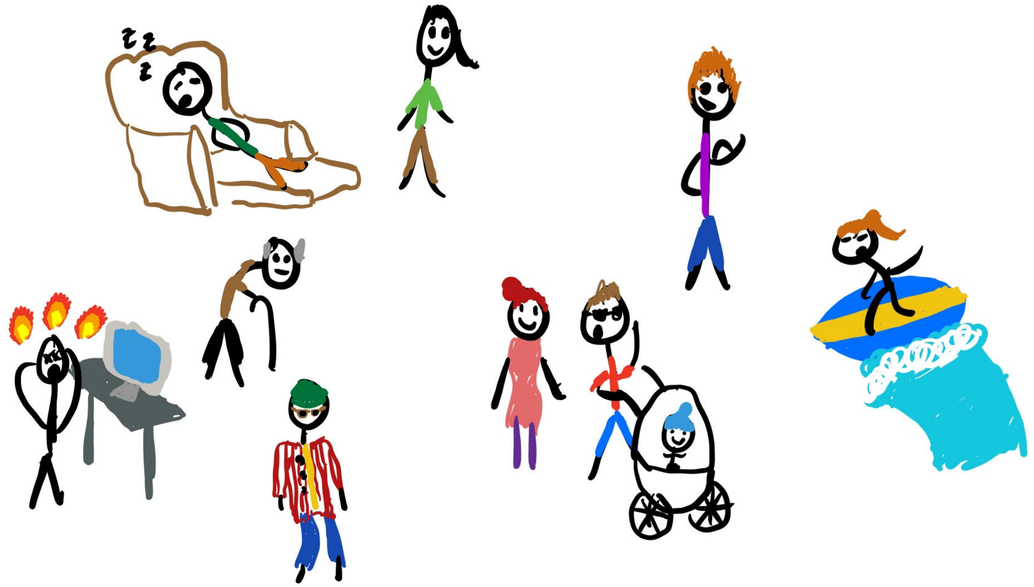
    $$\mathscr{Q}(y_i) := \arg\min_{\mathscr{A}} |y_i + u_{i-1}|,$$
    $$u_i := u_{i-1} + y_i - q_i$$

    - For order r schemes, $\mathscr{D}(x, q) \lesssim m^{-r}$
      - Daubechies, Devore 2003; or Güntürk, Lammers, Powell, Saab, Yilmaz 2010

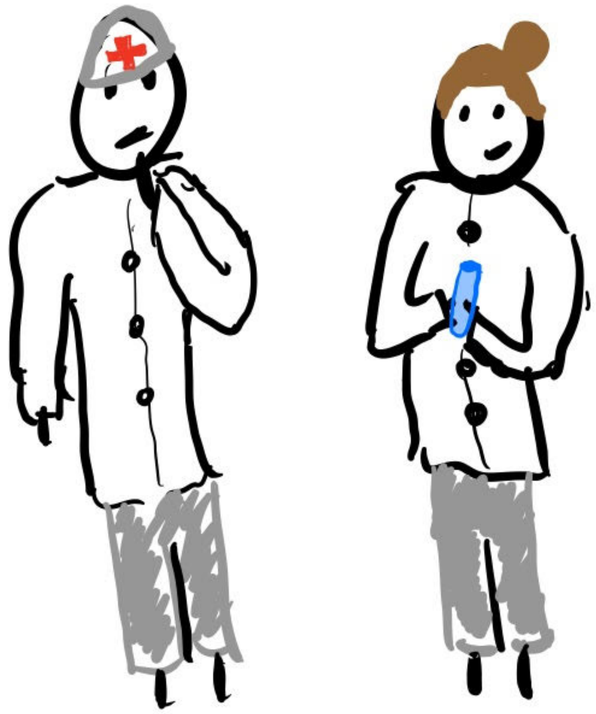


Example: Discrepancy Theory



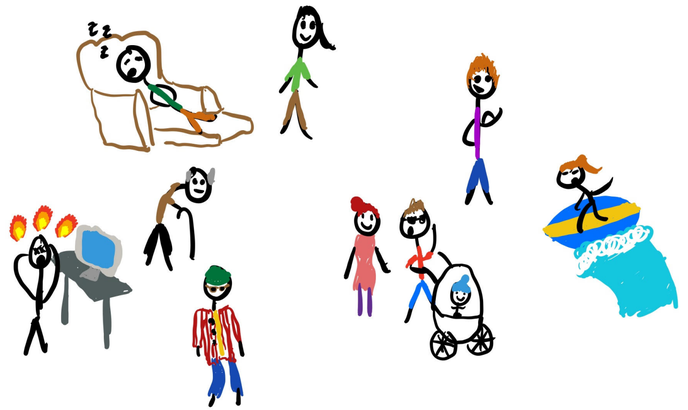


Example: Discrepancy Theory

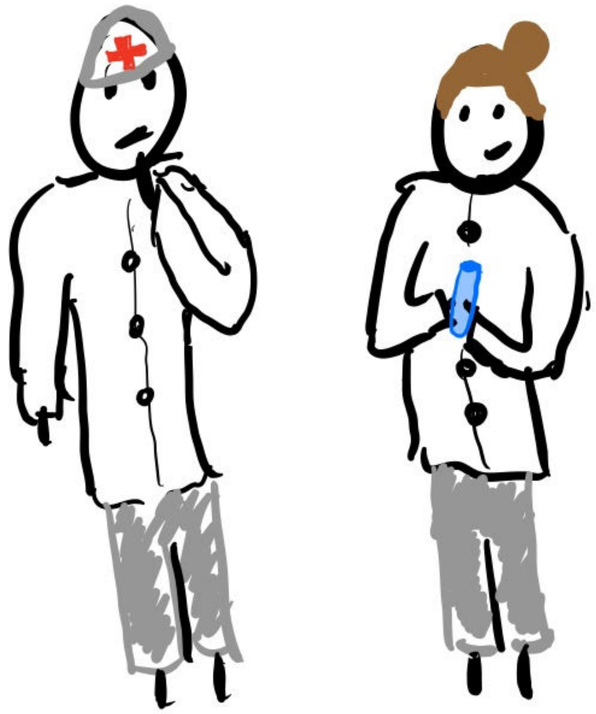


Subject Covariates

$$X = \begin{bmatrix} - & x_1^T & - \\ - & \cdots & - \\ - & x_m^T & - \end{bmatrix}$$



Example: Discrepancy Theory



Control Covariates

$$\hat{X} = \begin{bmatrix} - & \hat{x}_1^T & - \\ - & \cdots & - \\ - & \hat{x}_{m_1}^T & - \end{bmatrix}$$

Treatment Covariates

$$\widetilde{X} = \begin{bmatrix} - & \widetilde{x}_1^T & - \\ - & \cdots & - \\ - & \widetilde{x}_{m_2}^T & - \end{bmatrix}$$

Example: Discrepancy Theory

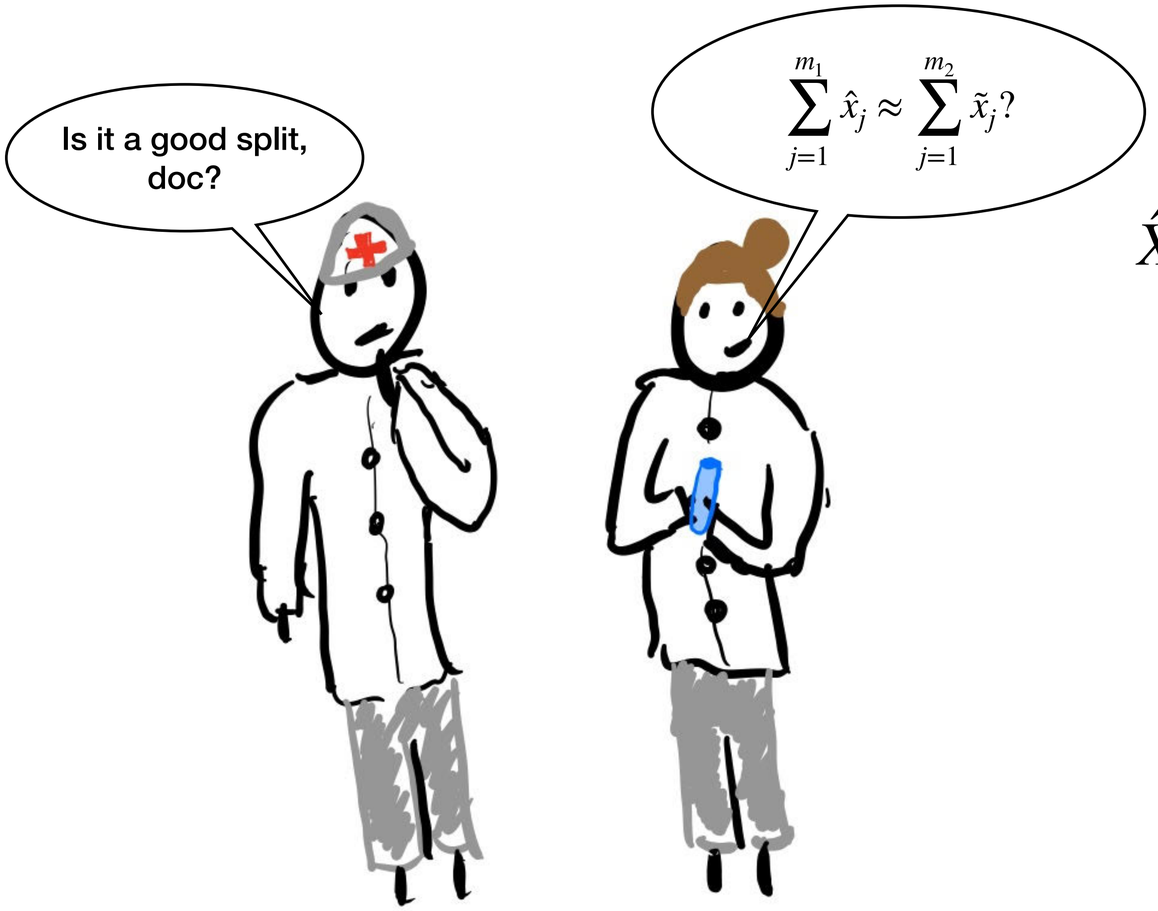
Control Covariates

$$\hat{X} = \begin{bmatrix} - & \hat{x}_1^T & - \\ - & \cdots & - \\ - & \hat{x}_{m_1}^T & - \end{bmatrix}$$

Treatment Covariates

$$\widetilde{X} = \begin{bmatrix} - & \widetilde{x}_1^T & - \\ - & \cdots & - \\ - & \widetilde{x}_{m_2}^T & - \end{bmatrix}$$

Example: Discrepancy Theory



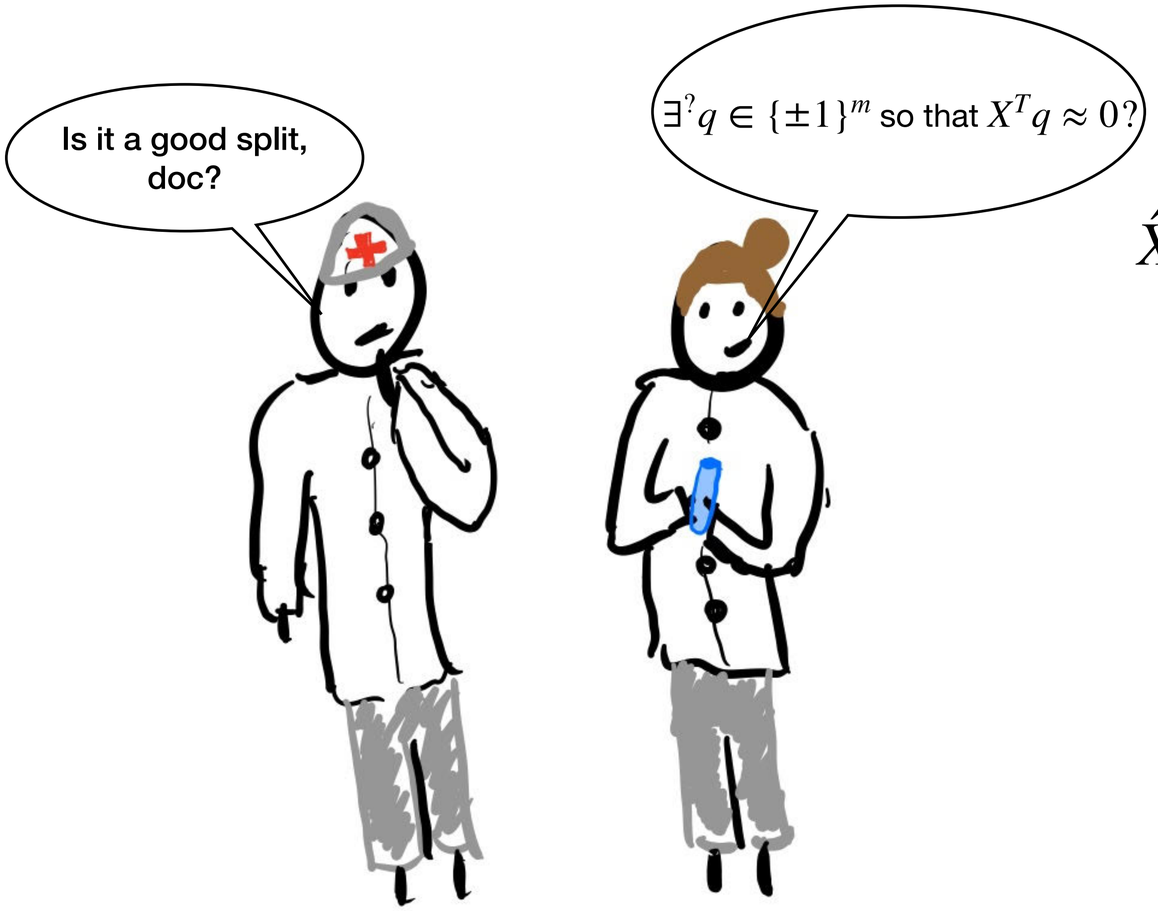
Control Covariates

$$\hat{X} = \begin{bmatrix} - & \hat{x}_1^T & - \\ - & \cdots & - \\ - & \hat{x}_{m_1}^T & - \end{bmatrix}$$

Treatment Covariates

$$\widetilde{X} = \begin{bmatrix} - & \tilde{x}_1^T & - \\ - & \cdots & - \\ - & \tilde{x}_{m_2}^T & - \end{bmatrix}$$

Example: Discrepancy Theory



Control Covariates

$$\hat{X} = \begin{bmatrix} - & \hat{x}_1^T & - \\ - & \cdots & - \\ - & \hat{x}_{m_1}^T & - \end{bmatrix}$$

Treatment Covariates

$$\widetilde{X} = \begin{bmatrix} - & \widetilde{x}_1^T & - \\ - & \cdots & - \\ - & \widetilde{x}_{m_2}^T & - \end{bmatrix}$$

Example: Discrepancy Theory

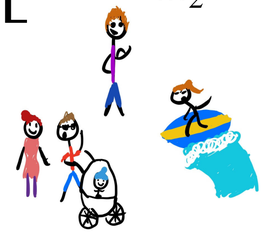
- **Goal:** Given a matrix $X \in \mathbb{R}^{m \times n}$, construct a vector $q \in \{\pm 1\}^m$ so that $\|X^T q\|_2 \approx 0$.
  - "Encoding" the vector

    $$w^* := \arg\min_{\|w\|_2 = 1} \|X^T w\|_2.$$

  - More subjects than covariates means $m > n \implies \mathscr{D}(q) = \|X^T(w^* - q)\|_2$
  - Decoder—namely, X^T—is fixed!
    - Rounding minimizes $\|w^* - q\|_2$
    - Want $w^* - q \in \ker(X^T)$

Control Covariates Treatment Covariates

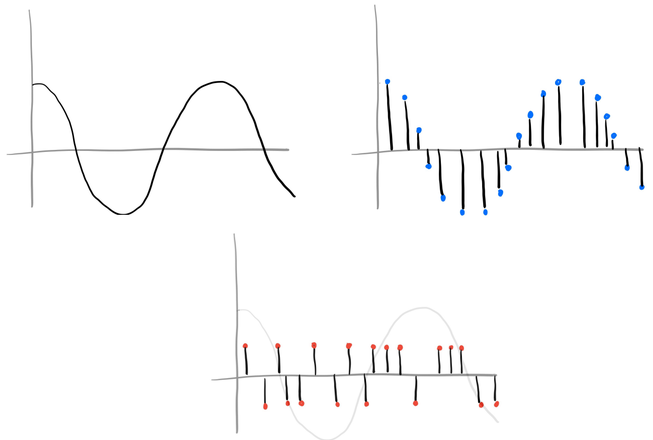
$$\hat{X} = \begin{bmatrix} - & \hat{x}_1^T & - \\ - & \cdots & - \\ - & \hat{x}_{m_1}^T & - \end{bmatrix} \qquad \widetilde{X} = \begin{bmatrix} - & \widetilde{x}_1^T & - \\ - & \cdots & - \\ - & \widetilde{x}_{m_2}^T & - \end{bmatrix}$$



The Analysis & Synthesis Problems

Approximate unknown signal from quantized measurements $q := \mathcal{Q}(\Phi x) \in \mathcal{A}^m$, $\Phi \in \mathbb{R}^{m \times n}$ **fixed**.

$$\mathscr{D}_{\Sigma\Delta}(x, q) := \|\Psi\left(\Phi x - \mathcal{Q}(\Phi x)\right)\|_2,$$



Given covariate data $X \in \mathbb{R}^{m \times n}$, construct a vector $q \in \{\pm 1\}^m$ so that $\|X^T q\|_2 \approx 0$.

$$m > n \implies \mathscr{D}(q) = \|X^T(w^* - q)\|_2$$

The Analysis & Synthesis Problems

Given an unknown signal $x \in \mathbb{R}^N$, construct
a mapping $x \in \mathbb{R}^N \to q \in \mathscr{A}^m$ to minimize

$$\|x - \Psi q\|_2.$$

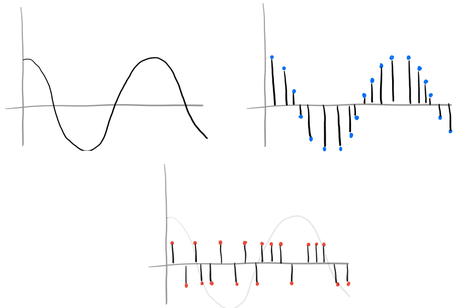
The Analysis & Synthesis Problems

Given an unknown signal $x \in \mathbb{R}^N$, construct a mapping $x \in \mathbb{R}^N \to q \in \mathscr{A}^m$ to minimize

$$\|x - \Psi q\|_2.$$

The Analysis Problem

$q = \mathcal{Q}(\Phi x), \Phi \in \mathbb{R}^{m \times N}$ is fixed. You get to choose $\mathcal{Q} : \mathbb{R}^m \to \mathscr{A}^m$ and **decoder** $\Psi \in \mathbb{R}^{N \times m}$ so long as $\Psi\Phi = I$.



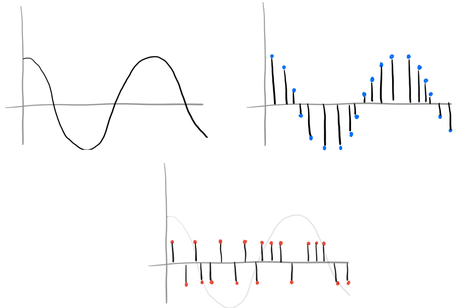
The Analysis & Synthesis Problems

Given an unknown signal $x \in \mathbb{R}^N$, construct
a mapping $x \in \mathbb{R}^N \to q \in \mathscr{A}^m$ to minimize

$$\|x - \Psi q\|_2.$$

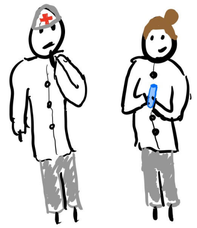
The Analysis Problem

$q = \mathcal{Q}(\Phi x), \Phi \in \mathbb{R}^{m \times N}$ is fixed. You get to choose $\mathcal{Q} : \mathbb{R}^m \to \mathscr{A}^m$ and **decoder** $\Psi \in \mathbb{R}^{N \times m}$ so long as $\Psi\Phi = I$.



The Synthesis Problem

$\Psi \in \mathbb{R}^{N \times m}$ is fixed. You get to choose $x \in \mathbb{R}^N \to q \in \mathscr{A}^m$ however you wish.



The Analysis & Synthesis Problems

Given an unknown signal $x \in \mathbb{R}^N$, construct a mapping $x \in \mathbb{R}^N \to q \in \mathscr{A}^m$ to minimize

$$\|x - \Psi q\|_2.$$

The Analysis Problem

$q = \mathcal{Q}(\Phi x), \Phi \in \mathbb{R}^{m \times N}$ is fixed. You get to choose $\mathcal{Q} : \mathbb{R}^m \to \mathscr{A}^m$ and **decoder** $\Psi \in \mathbb{R}^{N \times m}$ so long as $\Psi\Phi = I$.

- M. Iwen, E. Lybrand, A. Nelson, R. Saab. "New Algorithms and Improved Guarantees for One-Bit Compressed Sensing on Manifolds." SampTA2019. arXiv Proceedings

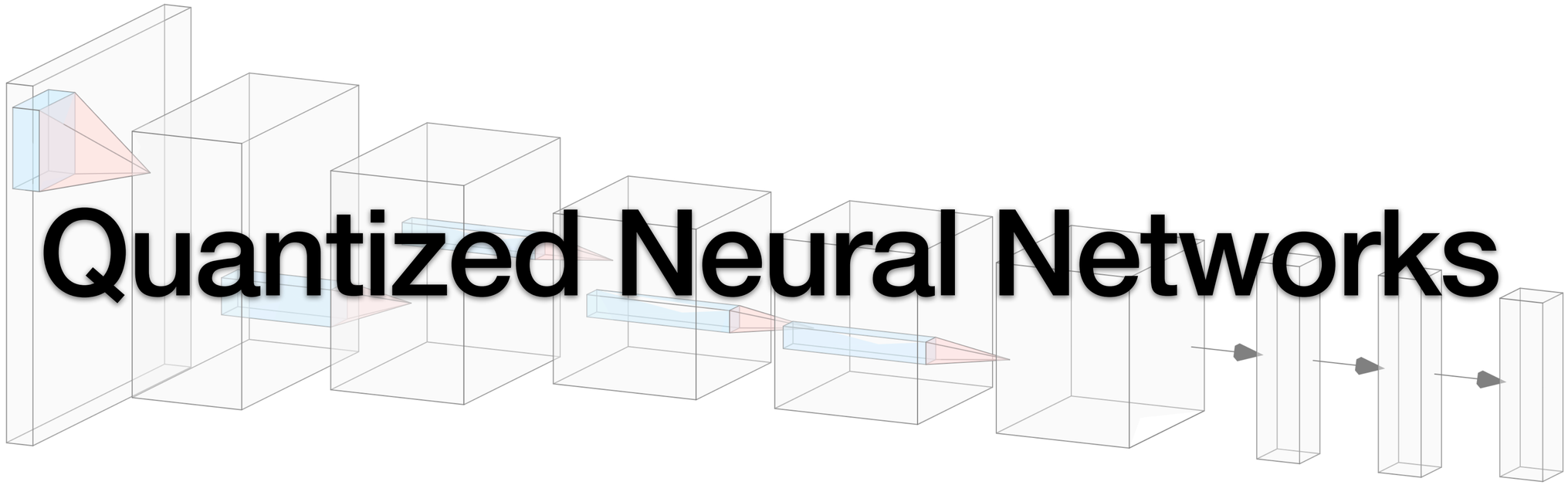
- E. Lybrand and R. Saab. "Quantization for Low-Rank Matrix Recovery." Information and Inference, 2018. arXiv Journal

The Synthesis Problem

$\Psi \in \mathbb{R}^{N \times m}$ is fixed. You get to choose $x \in \mathbb{R}^N \to q \in \mathscr{A}^m$ however you wish.

- E. Lybrand and R. Saab. "A Greedy Algorithm for Quantizing Neural Networks." In review, 2020. arXiv GitHub

Quantized Neural Networks



Joint work with Rayan Saab

Google's AlphaGo Defeats Chinese Go Master in Win for A.I.

Ke Jie, the world's top Go player, reacting during his match on Tuesday against AlphaGo, artificial intelligence software developed by a Google affiliate. China Stringer Network, via Reuters

GOOGLE

INSIDE WAYMO'S STRATEGY TO GROW THE BEST BRAINS FOR SELF-DRIVING CARS

The Google spinoff has a head start in AI, but can they maintain the lead?

By Andrew J. Hawkins | @andyjayhawk | May 9, 2018, 8:00am EDT

Illustration by Alex Castro

CADE METZ BUSINESS 09.27.2016 01:00 PM

An Infusion of AI Makes Google Translate More Powerful Than Ever

The Internet giant has unveiled an English-Chinese translation system built entirely on deep neural networks, saying it reduces error rates by 60 percent.

GLOBAL BANKING & Finance review

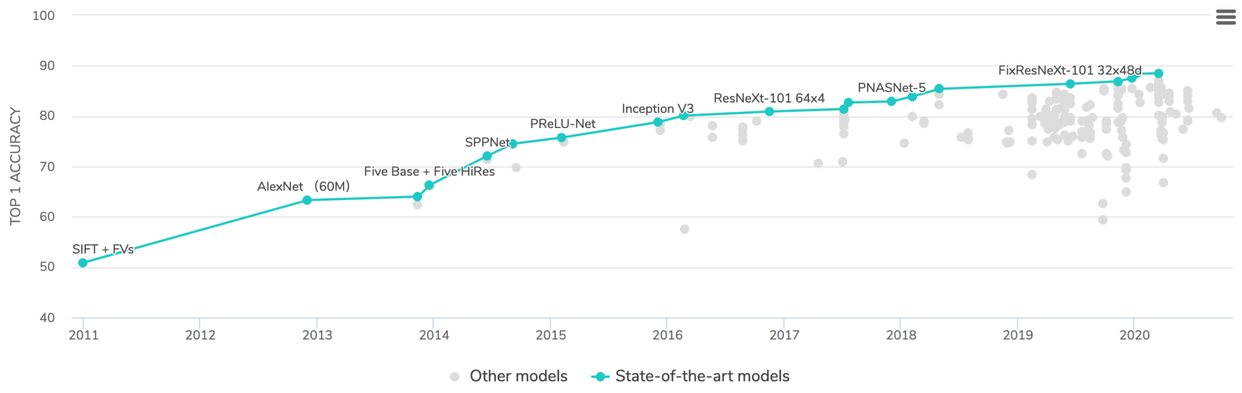
TOP STORIES INTERVIEWS BUSINESS FINANCE BANKING TECHNOLOGY INVESTING TRADING VIDEOS AWARDS MAGAZINES OUTSOURCING

TOP STORIES

Deep learning: the next frontier for money laundering detection

Lots of Complexity

Image Classification on ImageNet



View Top 1 Accuracy All models Edit

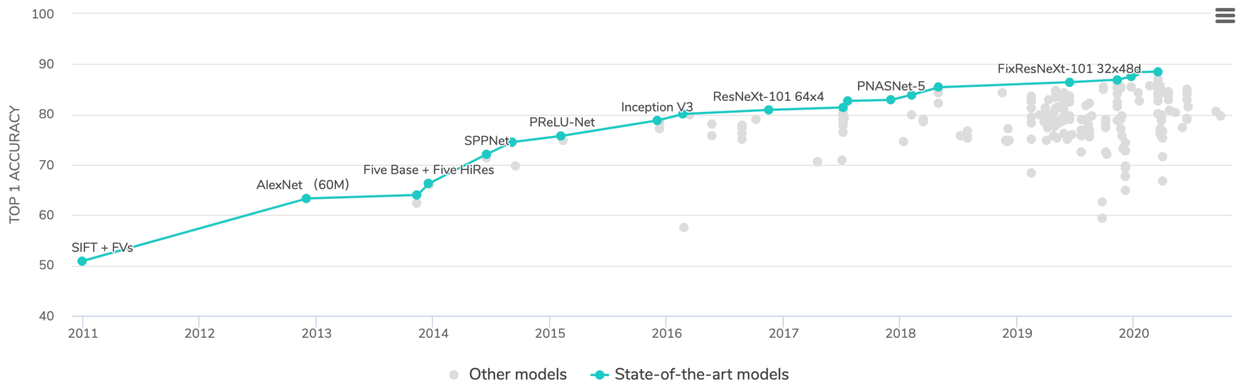
RANK	MODEL	TOP 1 ACCURACY	TOP 5 ACCURACY	NUMBER OF PARAMS	EXTRA TRAINING DATA	PAPER	CODE	RESULT	YEAR
1	**FixEfficientNet-L2**	88.5%	98.7%	480M	✓	Fixing the train-test resolution discrepancy: FixEfficientNet			2020
2	**NoisyStudent** (480M)	88.4%	98.7%	480M	✓	Self-training with Noisy Student improves ImageNet classification			2020
3	**BiT-L** (ResNet)	87.54%	98.46%	***	✓	Big Transfer (BiT): General Visual Representation Learning			2019
4	**FixEfficientNet-B7**	87.1%	98.2%	66M	✓	Fixing the train-test resolution discrepancy: FixEfficientNet			2020

***Based on ResNet152 x 4, so ~ 240M params

Lots of Complexity

7	FixResNeXt-101 32x48d	86.4%	98.0%	829M	✓	Fixing the train-test resolution discrepancy		2019
22	ResNeXt-101 32x32d	85.1%	97.5%	466M	✓	Exploring the Limits of Weakly Supervised Pretraining		2018
38	AmoebaNet-A	83.9%	96.6%	469M	✗	Regularized Evolution for Image Classifier Architecture Search		2018

Image Classification on ImageNet



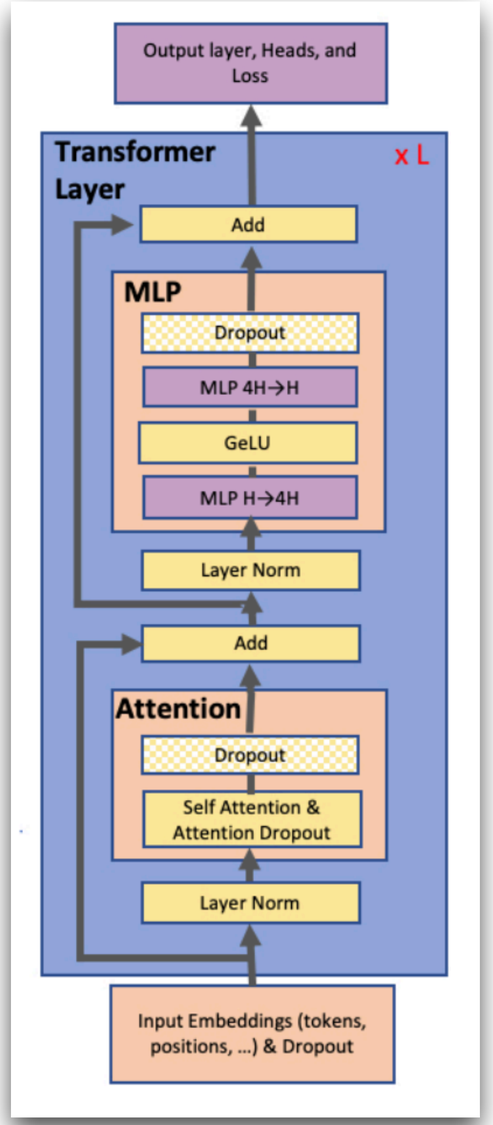
View: Top 1 Accuracy | All models | Edit

RANK	MODEL	TOP 1 ACCURACY	TOP 5 ACCURACY	NUMBER OF PARAMS	EXTRA TRAINING DATA	PAPER	CODE	RESULT	YEAR
1	FixEfficientNet-L2	88.5%	98.7%	480M	✓	Fixing the train-test resolution discrepancy: FixEfficientNet			2020
2	NoisyStudent (480M)	88.4%	98.7%	480M	✓	Self-training with Noisy Student improves ImageNet classification			2020
3	BiT-L (ResNet)	87.54%	98.46%	***	✓	Big Transfer (BiT): General Visual Representation Learning			2019
4	FixEfficientNet-B7	87.1%	98.2%	66M	✓	Fixing the train-test resolution discrepancy: FixEfficientNet			2020

***Based on ResNet152 x 4, so ~ 240M params

And it’s getting worse…

- NVIDIA MegatronLM: NLP
  - e.g. Text completion/prediction (LAMBADA), language encoding (WikiText103), question answering (RACE)
- Released Aug 13, 2019
- 8.3 billion parameters
- 33GB on disk
- Trained on 512 GPUS for 9 days
  - “Roughly 3x the yearly energy consumption of average American”



Schematic of one of the 72 sub-networks in Megatron-LM

https://heartbeat.fritz.ai/deep-learning-has-a-size-problem-ea601304cd8

Shoeybi, Mohammad, et al. “Megatron-LM: Training multi-billion parameter language models using gpu model parallelism.”

Can We Do Better?

Can We Do Better?

Can we reduce the computational & memory burden for deploying ***pre-trained*** neural networks?

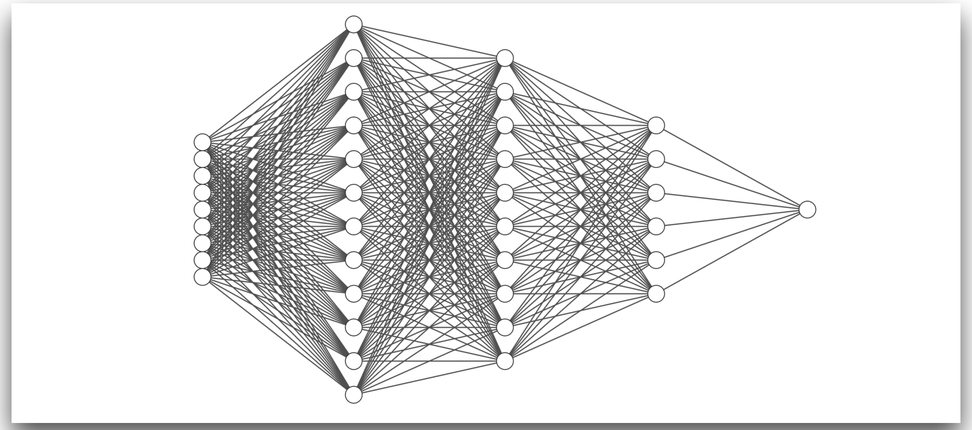
Motivating Questions

- For every neural network, does there exist a quantized neural network that approximates it well?
  - If yes,…
  - can it be constructed in a reasonable amount of time?
  - does it generalize to new data?

Set up

- Formally, define a L-layer perceptron $\Phi : \mathbb{R}^{N_0} \to \mathbb{R}^{N_L}$ by

$$\Phi(x) := \varphi \circ A^{(L)} \circ \cdots \circ \varphi \circ A^{(1)}(x),$$

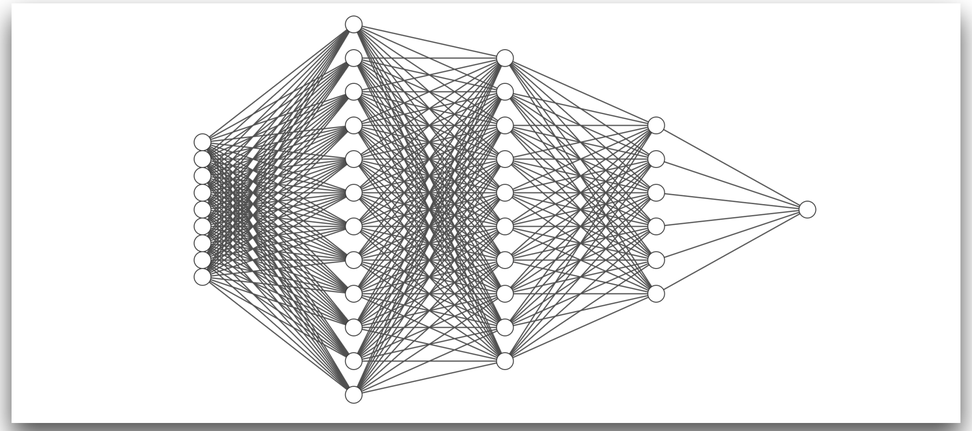


Set up

- Formally, define a L-layer perceptron $\Phi : \mathbb{R}^{N_0} \to \mathbb{R}^{N_L}$ by

$$\Phi(x) := \varphi \circ A^{(L)} \circ \cdots \circ \varphi \circ A^{(1)}(x),$$

  - $\varphi : \mathbb{R} \to \mathbb{R}$ a component-wise nonlinearity, e.g. $\varphi(x) = \max\{0,x\}$

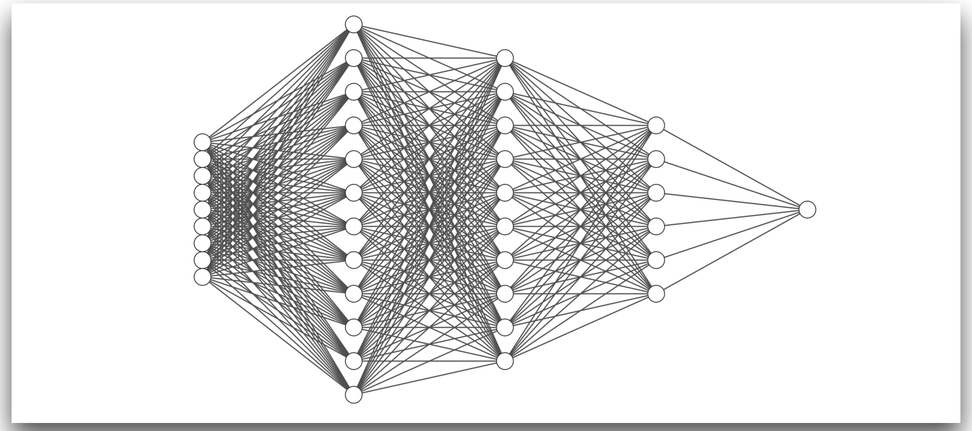


Set up

- Formally, define a L-layer perceptron $\Phi : \mathbb{R}^{N_0} \to \mathbb{R}^{N_L}$ by

$$\Phi(x) := \varphi \circ A^{(L)} \circ \cdots \circ \varphi \circ A^{(1)}(x),$$

  - $\varphi : \mathbb{R} \to \mathbb{R}$ a component-wise nonlinearity, e.g. $\varphi(x) = \max\{0,x\}$
  - $A^{(\ell)}(x) = x^T \underbrace{W^{(\ell)}}_{\text{weight matrix}} + \underbrace{b^{(\ell)T}}_{\text{bias}}, \quad W^{(\ell)} \in \mathbb{R}^{N_\ell \times N_{\ell+1}}, \quad b^{(\ell)} \in \mathbb{R}^{N_{\ell+1}}$



Set up

- $A^{(\ell)}(x) = x^T W^{(\ell)} + b^{(\ell)T}, \quad \underbrace{W^{(\ell)}}_{\text{weight matrix}} \in \mathbb{R}^{N_\ell \times N_{\ell+1}}, \quad \underbrace{b^{(\ell)}}_{\text{bias}} \in \mathbb{R}^{N_{\ell+1}}$

Set up

- $A^{(\ell)}(x) = x^T W^{(\ell)} + b^{(\ell)T}, \quad \underbrace{W^{(\ell)}}_{\text{weight matrix}} \in \mathbb{R}^{N_\ell \times N_{\ell+1}}, \quad \underbrace{b^{(\ell)}}_{\text{bias}} \in \mathbb{R}^{N_{\ell+1}}$
- Can “ignore” bias

$$x^T W + b^T = [x^T, 1] \begin{bmatrix} W \\ b^T \end{bmatrix}$$

Set up

- $A^{(\ell)}(x) = x^T W^{(\ell)} + b^{(\ell)T}, \quad \underbrace{W^{(\ell)}}_{\text{weight matrix}} \in \mathbb{R}^{N_\ell \times N_{\ell+1}}, \quad \underbrace{b^{(\ell)}}_{\text{bias}} \in \mathbb{R}^{N_{\ell+1}}$
- Can “ignore” bias

$$x^T W + b^T = [x^T, 1] \begin{bmatrix} W \\ b^T \end{bmatrix}$$

- **GOAL**: replace the $W^{(\ell)}$ with ***quantized*** matrices $Q^{(\ell)}$

Lots of Empirical Work

- Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. 2016
- Yunchao Gong, Liu Liu, Ming Yang, and Lubomir Bourdev. Compressing deep convolutional networks using vector quantization. 2014.
- Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. 2016.
- Peisong Wang and Jian Cheng. Fixed-point factorized networks. 2017.
- Hubara, Itay, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. "Quantized neural networks: Training neural networks with low precision weights and activations."

A Data Driven Approach

A Data Driven Approach

- Given:

A Data Driven Approach

- Given:

A matrix of data

$$X := \begin{bmatrix} - & x_1^T & - \\ & \vdots & \\ - & x_m^T & - \end{bmatrix} \in \mathbb{R}^{m \times N_0}$$

A Data Driven Approach

- Given:
  - Data $X \in \mathbb{R}^{m \times N_0}$,

Pretrained Neural Network

$$\Phi(x) := \varphi \circ W^{(L)} \circ \cdots \circ \varphi \circ W^{(1)}(x)$$

A Data Driven Approach

- Given:
  - Data $X \in \mathbb{R}^{m \times N_0}$,
  - Pretrained net Φ

Quantization Alphabet

$\mathscr{A}$, e.g. $\{-1,0,1\}$

A Data Driven Approach

- Given:
  - Data $X \in \mathbb{R}^{m \times N_0}$,
  - Pretrained net Φ
  - Quantization alphabet $\mathscr{A}$

Output: $Q^{(\ell)} \in \mathscr{A}^{N_\ell \times N_{\ell+1}}$ s.t.

$$\|Q^{(\ell)} \circ \varphi \circ \cdots \circ \varphi \circ Q^{(1)}(X) - W^{(\ell)} \circ \varphi \circ \cdots \circ \varphi \circ W^{(1)}(X)\|_F^2 \approx 0, \ \forall \ell$$

$$\|Q^{(\ell)} \circ \varphi \circ \cdots \circ \varphi \circ Q^{(1)}(X) - W^{(\ell)} \circ \varphi \circ \cdots \circ \varphi \circ W^{(1)}(X)\|_F^2 \approx 0, \ \forall \ell$$

$$\|Q^{(1)}(X) - W^{(1)}(X)\|_F^2$$

$$\|Q^{(1)}(X) - W^{(1)}(X)\|_F^2$$

$$= \|X(Q^{(1)} - W^{(1)})\|_F^2$$

$$\|Q^{(1)}(X) - W^{(1)}(X)\|_F^2 = \|X(Q^{(1)} - W^{(1)})\|_F^2$$

$$= \left\| X \left[q^{(1)} - w^{(1)} \quad \dots \quad q^{(N_1)} - w^{(N_1)} \right] \right\|_F^2$$

$$\|Q^{(1)}(X) - W^{(1)}(X)\|_F^2 = \|X(Q^{(1)} - W^{(1)})\|_F^2$$

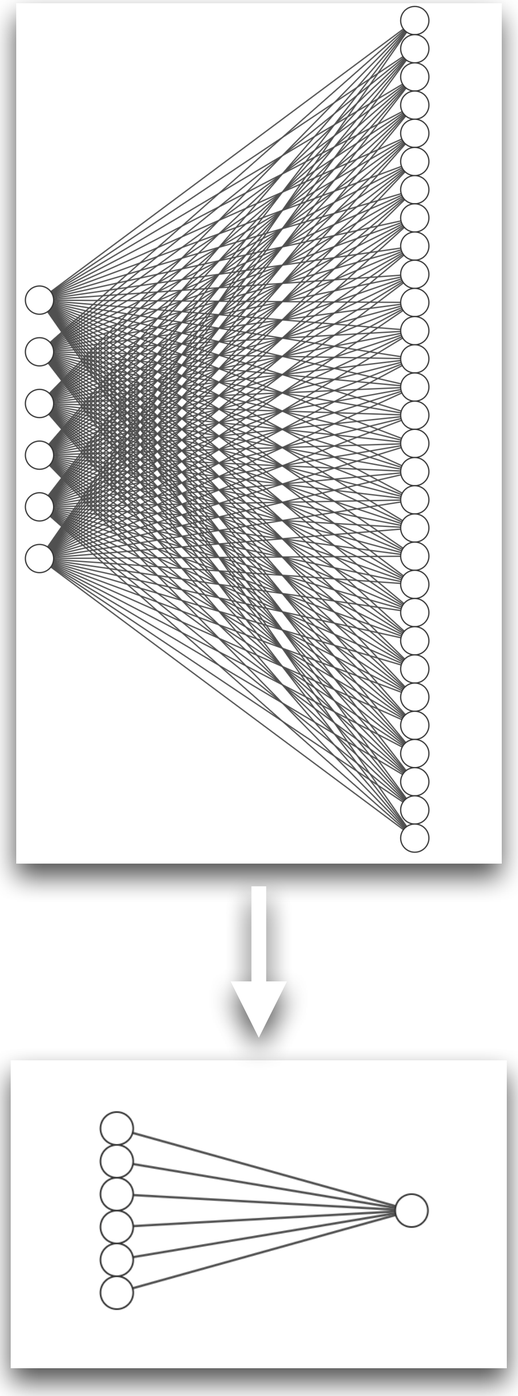
$$= \left\| X \left[q^{(1)} - w^{(1)} \quad \dots \quad q^{(N_1)} - w^{(N_1)} \right] \right\|_F^2$$

$$= \sum_{j=1}^{N_1} \|X(w^{(j)} - q^{(j)})\|_2^2$$

The First Layer

- Quantize one hidden unit at a time:

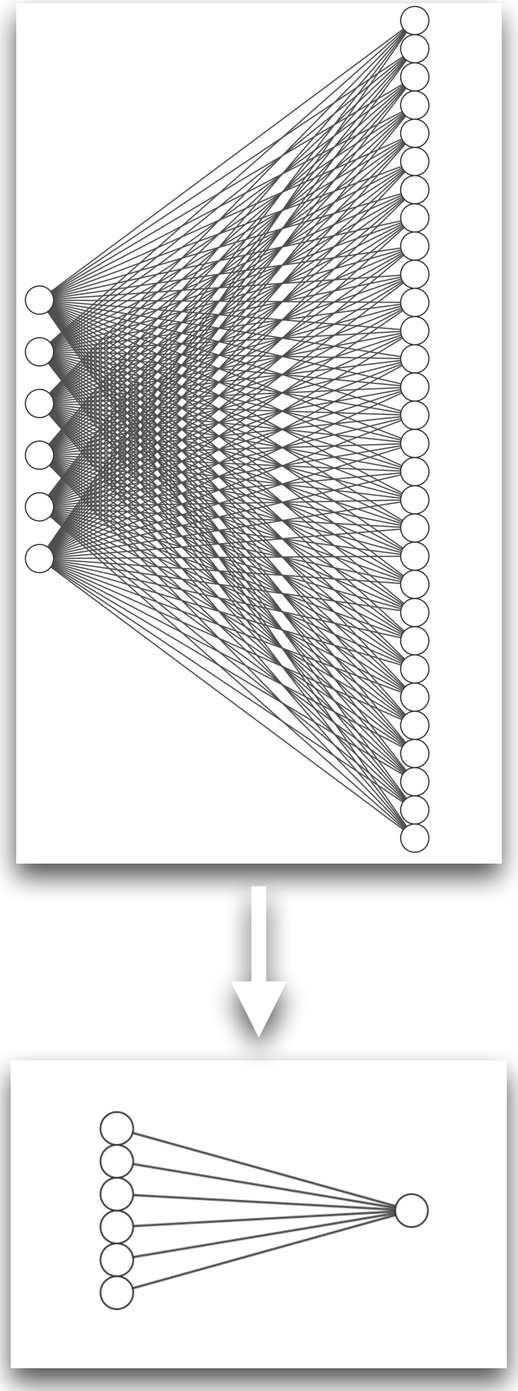
$$\|X(w^{(j)} - q^{(j)})\|_2^2$$



The First Layer

- Quantize one hidden unit at a time:

$$\|X(w - q)\|_2^2$$



The First Layer

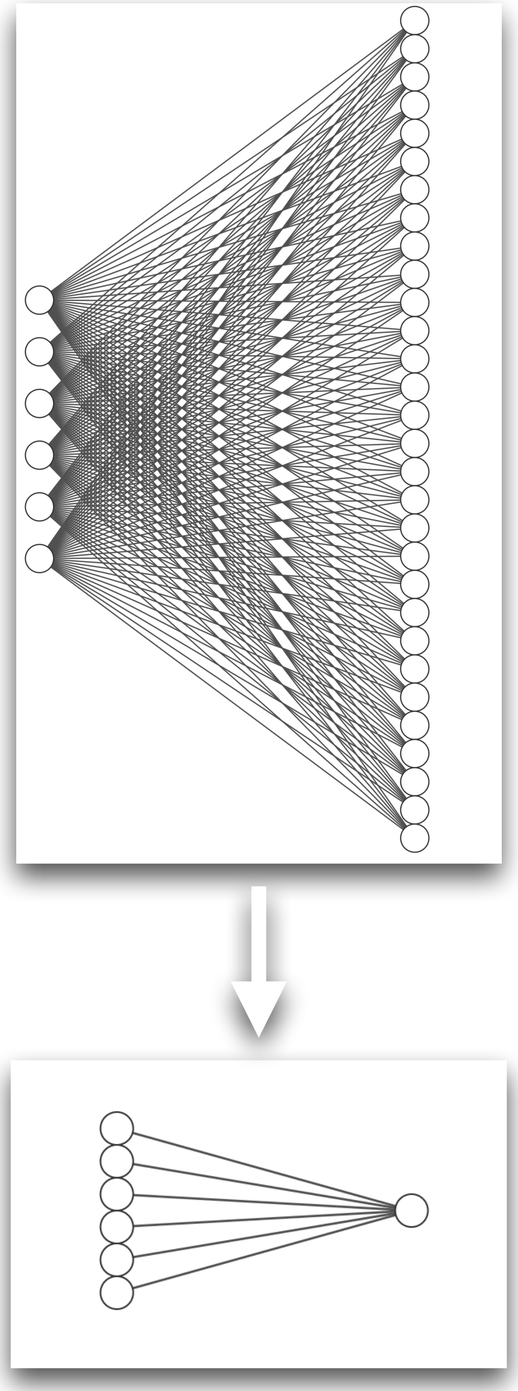
- Quantize one hidden unit at a time:

$$\|X(w-q)\|_2^2$$

  - Ideally,

$$q^\sharp := \arg\min_{p \in \mathcal{A}^{N_0}} \|Xw - Xq\|_2$$

- Is $\|Xw - Xq^\sharp\|_2$ even guaranteed to be small?



Discrepancy Theory

Discrepancy Theory

- Special case: $\mathscr{A} = \{-1,1\},\ w = 0$

Discrepancy Theory

- Special case: $\mathcal{A} = \{-1, 1\},\ w = 0$
- **Observation:** If $N_0 \leq m$ and columns of $X \in \mathbb{R}^{m \times N_0}$ orthonormal,

$$\|Xq\|_2^2 = \|q\|_2^2 = N_0 \leq m$$

Discrepancy Theory

- Special case: $\mathcal{A} = \{-1,1\},\ w = 0$
- **Observation:** If $N_0 \leq m$ and columns of $X \in \mathbb{R}^{m \times N_0}$ orthonormal,

$$\|Xq\|_2^2 = \|q\|_2^2 = N_0 \leq m$$

- In general…

Theorem (Spencer, 1985):
There exists a universal constant c > 0 so that for any $X_1, \ldots, X_N \in \mathbb{R}^m$ with $\sup_t \|X_t\|_2^2 \leq 1$, there exists $q \in \{-1,1\}^N$ such that

$$\|Xq\|_\infty \leq c \log(m)\,.$$

Discrepancy Theory

- Special case: $\mathscr{A} = \{-1,1\},\ w = 0$
- **Observation:** If $N_0 \leq m$ and columns of $X \in \mathbb{R}^{m \times N_0}$ orthonormal,

$$\|Xq\|_2^2 = \|q\|_2^2 = N_0 \leq m$$

- In general…

Komlòs Conjecture

~~Theorem (Spencer, 1985).~~

There exists a universal constant c > 0 so that for any $X_1, \ldots, X_N \in \mathbb{R}^m$ with $\sup_t \|X_t\|_2^2 \leq 1$, there exists $q \in \{-1,1\}^N$ such that

$$\|Xq\|_\infty \leq c\,\cancel{\log(m)}\,.$$

$$\implies \|Xq\|_2 \leq c\sqrt{m}\,.$$

Discrepancy Theory

- Special case: $\mathscr{A} = \{-1,1\}$, ~~$w = 0$~~ $\|w\|_\infty \leq 1$ (Lovás, Spencer, Vesztergombi 1986)
- **Observation:** If $N_0 \leq m$ and columns of $X \in \mathbb{R}^{m \times N_0}$ orthonormal,

$$\|Xq\|_2^2 = \|q\|_2^2 = N_0 \leq m$$

- In general…

Komlòs Conjecture

~~Theorem (Spencer, 1985).~~
There exists a universal constant c > 0 so that for any $X_1, \ldots, X_N \in \mathbb{R}^m$ with $\sup_t \|X_t\|_2^2 \leq 1$, there exists $q \in \{-1,1\}^N$ such that

$$\|Xw - Xq\|_\infty \leq c\,\log(m).$$

$$\implies \|Xw - Xq\|_2 \leq c\sqrt{m}\,.$$

Discrepancy Theory

- Special case: $\mathscr{A} = \{-1,1\}$, ~~$w = 0$~~ $\|w\|_\infty \leq 1$ (Lovás, Spencer, Vesztergombi 1986)
- **Observation:** If $N_0 \leq m$ and columns of $X \in \mathbb{R}^{m \times N_0}$ orthonormal,

$$\|Xq\|_2^2 = \|q\|_2^2 = N_0 \leq m$$

- In general…

Komlòs Conjecture

~~Theorem (Spencer, 1985):~~
There exists a universal constant c > 0 so that for any $X_1, \ldots, X_N \in \mathbb{R}^m$ with

$\sup_t \|X_t\|_2^2 \leq 1$, ***there exists*** $q \in \{-1,1\}^N$ such that

$$\|Xw - Xq\|_\infty \leq c\,\cancel{\log(m)}\,.$$

$$\Longrightarrow \quad \|Xw - Xq\|_2 \leq c\sqrt{m}\,.$$

Lots of follow up research since

- Giannopoulos (1997): $N = m,\ Xq \in c\log(m)K$, K origin symmetric convex set with gaussian measure $\gamma(K) \geq 1/2$
  - Non-constructive
- Banaszczyk (1998): extends Giannopoulos, arbitrary $N, m,\ Xq \in cK$
  - $\implies \|Xq\|_\infty \leq c\sqrt{\log(m)}$
  - Non-constructive
- Dadush, Garg, Lovett, Nikolov (2016): arbitrary N, m, construction of Banaszczyk's q
  - Requires solving SDP ($O(N^6)$ flops) & Cholesky decomposition ($O(N^3)$ flops) a total of $O(N^5)$ times
- Bansal, Dadush, Garg, Lovett (2018): arbitrary N, m, construction of Banaszczyk's q
  - Time complexity at least $O(N(N+m)^2)$

Motivating Questions

- For every neural network, does there exist a quantized neural network that approximates it well?
  - If yes,…
  - can it be constructed in a reasonable amount of time?
  - does it generalize to new data?

Motivating Questions

- For every **single layer** neural network, does there exist a quantized neural network that approximates it well?
  - If yes,… 
  - can it be constructed in a reasonable amount of time?
  - does it generalize to new data?

- Think about $X(w-q) = \sum_{j=1}^{N_0} w_j X_j - q_j X_j$ a
- Construct q sequentially and in a greedy fashion
  - Want $\sum_{j=1}^{t} w_j X_j \approx \sum_{j=1}^{t} q_j X_j \ , \forall t \in \{1,...,N_0\}$

- Think about $X(w-q)$ as linear combinations of columns X_t
- Construct q sequentially and in a greedy fashion
  - Want $\sum_{j=1}^{t} w_j X_j \approx \sum_{j=1}^{t} q_j X_j \ , \forall t \in \{1,...,N_0\}$

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $w \in B_\infty^{N_0}$, $\mathscr{A}$, for $t \in \{1,...,N_0\}$, set

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \arg \min_{p \in \mathscr{A}} \|u_{t-1} + w_t X_t - p X_t\|_2^2,$$

$$u_t := u_{t-1} + w_t X_t - q_t X_t$$

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $w \in B_\infty^{N_0}$, $\mathscr{A}$, for $t \in \{1,\ldots,N_0\}$, set

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \arg \min_{p \in \mathscr{A}} \|u_{t-1} + w_t X_t - p X_t\|_2^2,$$

$$u_t := u_{t-1} + w_t X_t - q_t X_t$$

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $w \in B_\infty^{N_0}$, $\mathcal{A}$, $\mathcal{Q}(x) := \arg\min_{p \in \mathcal{A}} |x - p|$, for $t \in \{1,\ldots,N_0\}$, set

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \mathcal{Q}\left(w_t + \frac{X_t^T u_{t-1}}{\|X_t\|_2^2} \right)$$

$$u_t := u_{t-1} + w_t X_t - q_t X_t$$

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $w \in B_\infty^{N_0}$, $\mathscr{A}$, $\mathscr{Q}(x) := \arg\min_{p \in \mathscr{A}} |x - p|$, for $t \in \{1,\ldots,N_0\}$, set

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \mathscr{Q}\left(w_t + \frac{X_t^T u_{t-1}}{\|X_t\|_2^2} \right) \Bigg\} \; O(m) \text{ flops}$$

$$u_t := u_{t-1} + w_t X_t - q_t X_t \Bigg\} \; O(m) \text{ flops}$$

Optimal Runtime

- Complexity of $O(N_0 m)$
- Optimal in that there are $N_0 \times m$ entries of X

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $w \in B_\infty^{N_0}$, $\mathscr{A}$, $\mathscr{Q}(x) := \arg\min_{p \in \mathscr{A}} |x - p|$, for $t \in \{1,...,N_0\}$, set

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \mathscr{Q}\left(w_t + \frac{X_t^T u_{t-1}}{\|X_t\|_2^2}\right)$$

$$u_t := u_{t-1} + w_t X_t - q_t X_t$$

the Orthonormal Walk

- $N_0 \leq m$, $\{X_t\}$ orthonormal, $\implies X_t \perp u_{t-1}\ \forall t$

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $w \in B_\infty^{N_0}$, $\mathscr{A}$, $\mathscr{Q}(x) := \arg\min_{p \in \mathscr{A}} |x - p|$, for $t \in \{1,...,N_0\}$, set

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \mathscr{Q}\left(w_t + \frac{X_t^T u_{t-1}}{\|X_t\|_2^2}\right)$$

$$u_t := u_{t-1} + w_t X_t - q_t X_t$$

the Orthonormal Walk

- $N_0 \leq m$, $\{X_t\}$ orthonormal, $\implies X_t \perp u_{t-1}\ \forall t$
- $\implies q_t = \mathscr{Q}(w_t)\ \forall t$

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $w \in B_\infty^{N_0}$, $\mathscr{A}$, $\mathscr{Q}(x) := \arg\min_{p \in \mathscr{A}} |x - p|$, for $t \in \{1,...,N_0\}$, set

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \mathscr{Q}\left(w_t + \frac{X_t^T u_{t-1}}{\|X_t\|_2^2}\right)$$

$$u_t := u_{t-1} + w_t X_t - q_t X_t$$

the Orthonormal Walk

- $N_0 \leq m$, $\{X_t\}$ orthonormal, $\implies X_t \perp u_{t-1}\ \forall t$
- $\implies q_t = \mathscr{Q}(w_t)\ \forall t$
- Pythagoras $\implies \|u_{N_0}\|_2^2 = \sum_{t=1}^{N_0} (w_t - q_t)^2 \|X_t\|_2^2 = \|w - q\|_2^2 \propto N_0$

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $w \in B_\infty^{N_0}$, $\mathscr{A}$, $\mathscr{Q}(x) := \arg\min_{p \in \mathscr{A}} |x - p|$, for $t \in \{1,...,N_0\}$, set

$$u_0 := 0 \in \mathbb{R}^m,$$

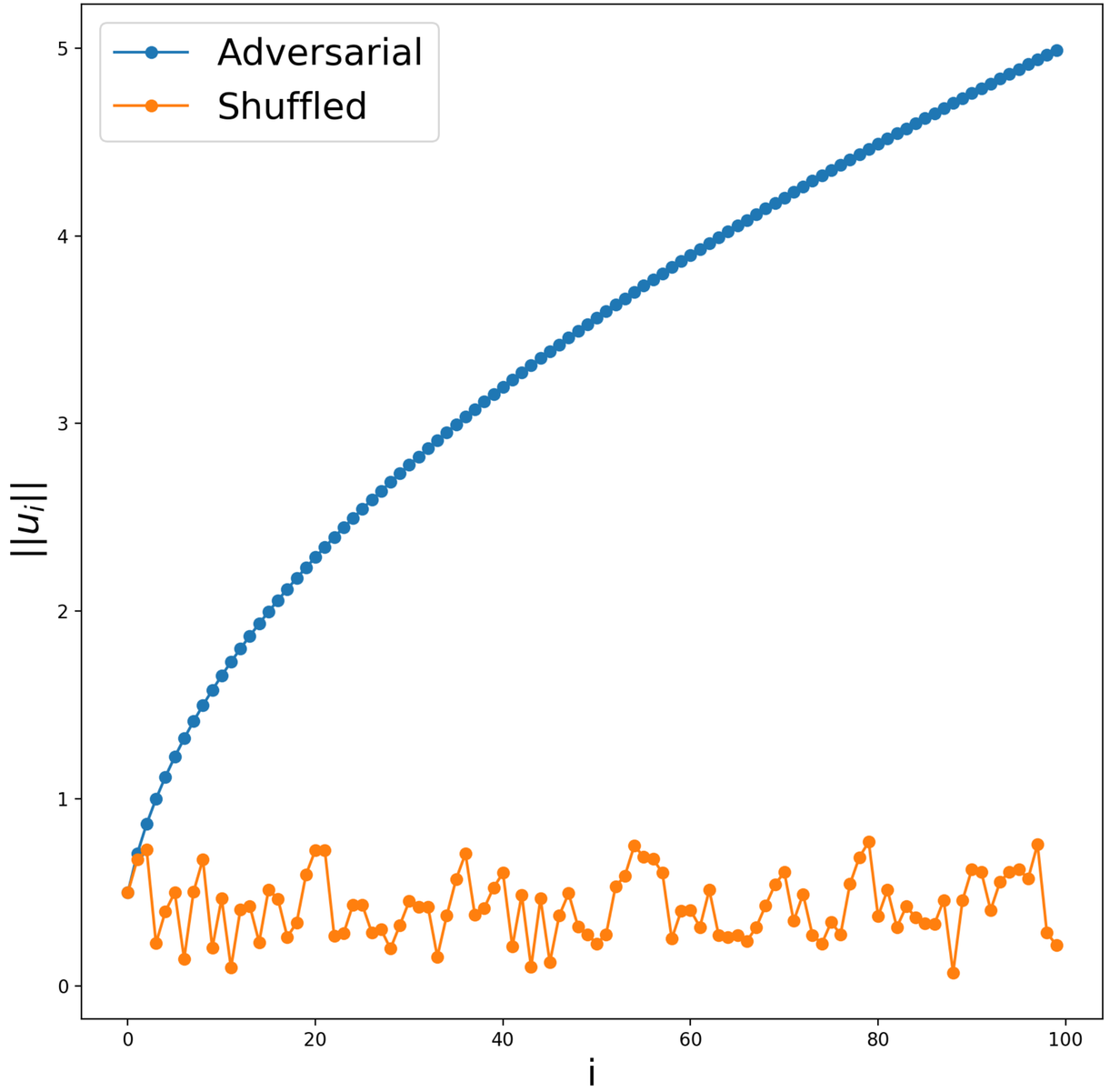
$$q_t := \mathscr{Q}\left(w_t + \frac{X_t^T u_{t-1}}{\|X_t\|_2^2}\right)$$

$$u_t := u_{t-1} + w_t X_t - q_t X_t$$

the Orthonormal Walk

- Pythagoras $\implies \|u_{N_0}\|_2^2 = \sum_{t=1}^{N_0} (w_t - q_t)^2 \|X_t\|_2^2 = \|w - q\|_2^2 \propto N_0$
- In general, choosing $X_t \perp u_{t-1}$ $\forall t$ achieves this bound for $N_0 \geq m$

Residuals of Adversarial Walk and Shuffled Walk



Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $w \in B_\infty^{N_0}$, $\mathscr{A}$, $\mathscr{Q}(x) := \arg\min_{p \in \mathscr{A}} |x - p|$, for $t \in \{1,\ldots,N_0\}$, set

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \mathscr{Q}\left(w_t + \frac{X_t^T u_{t-1}}{\|X_t\|_2^2}\right)$$

$$u_t := u_{t-1} + w_t X_t - q_t X_t$$

the One Dimensional Walk

- $X_t = X_s,\ \forall s,t,\ \|X_t\|_2 = 1\ \forall t \implies X_t \parallel u_{t-1}\ \forall t$
- $\implies q_t = \mathscr{Q}\left(w_t + \sum_{j=1}^{t-1} w_j - q_j\right)\ \forall t$ i.e. first order $\Sigma\Delta$ quantizer
- $\implies \|u_{N_0}\|_2^2 = O(1)$

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$.

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$. Suppose $X \in \mathbb{R}^{m \times N_0}$ has independent columns $X_t \sim \mathcal{N}(0, m^{-1} I_{m \times m})$,

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$. Suppose $X \in \mathbb{R}^{m \times N_0}$ has independent columns $X_t \sim \mathcal{N}(0, m^{-1} I_{m \times m})$, $w \in B_\infty^{N_0}$ satisfies $\operatorname{dist}(w_t, \{-1,0,1\}) > \varepsilon$ for all t.

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$. Suppose $X \in \mathbb{R}^{m \times N_0}$ has independent columns $X_t \sim \mathcal{N}(0, m^{-1} I_{m \times m})$, $w \in B_\infty^{N_0}$ satisfies $\operatorname{dist}(w_t, \{-1,0,1\}) > \varepsilon$ for all t. Then with high probability on the draw of X, if q is chosen according to our algorithm then

$$\|Xw - Xq\|_2 \lesssim_\varepsilon \sqrt{m} \log(N_0)$$

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$. Suppose $X \in \mathbb{R}^{m \times N_0}$ has independent columns $X_t \sim \mathscr{N}(0, m^{-1} I_{m \times m})$, $w \in B_\infty^{N_0}$ satisfies $\mathrm{dist}(w_t, \{-1,0,1\}) > \varepsilon$ for all t. Then with high probability on the draw of X, if q is chosen according to our algorithm then

$$\|Xw - Xq\|_2 \lesssim_\varepsilon \sqrt{m} \log(N_0)$$

Almost Komlòs

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$. Suppose $X \in \mathbb{R}^{m\times N_0}$ has independent columns $X_t \sim \mathscr{N}(0, m^{-1}I_{m\times m})$, $w \in B_\infty^{N_0}$ satisfies $\operatorname{dist}(w_t, \{-1,0,1\}) > \varepsilon$ for all t. Then with high probability on the draw of X, if q is chosen according to our algorithm then

$$\|Xw - Xq\|_2 \lesssim_\varepsilon \sqrt{m}\log(N_0)$$

Almost Komlòs

- Spencer (1985): $\forall X \in \mathbb{R}^{m\times N_0},\ \sup_t \|X_t\|_2 \le 1,\ \exists q \in \{-1,0,1\}$ such that

$$\|Xw - Xq\|_\infty \le c, \text{ so } \|Xw - Xq\|_2 \le c\sqrt{m}$$

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$. Suppose $X \in \mathbb{R}^{m\times N_0}$ has independent columns $X_t \sim \mathscr{N}(0, m^{-1}I_{m\times m})$, $w \in B_\infty^{N_0}$ satisfies $\mathrm{dist}(w_t, \{-1,0,1\}) > \varepsilon$ for all t. Then with high probability on the draw of X, if q is chosen according to our algorithm then

$$\|Xw - Xq\|_2 \lesssim_\varepsilon \sqrt{m}\log(N_0)$$

Almost Komlòs

- Spencer (1985): $\forall X \in \mathbb{R}^{m\times N_0},\ \sup_t \|X_t\|_2 \le 1,\ \exists q \in \{-1,0,1\}$ such that

$$\|Xw - Xq\|_\infty \le c, \text{ so } \|Xw - Xq\|_2 \le c\sqrt{m}$$

Up to log terms, constants, our bound matches

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$. Suppose $X \in \mathbb{R}^{m \times N_0}$ has independent columns $X_t \sim \mathcal{N}(0,\sigma^2 I_{m\times m})$, $w \in B_\infty^{N_0}$ satisfies $\mathrm{dist}(w_t, \{-1,0,1\}) > \varepsilon$ for all t. Then with high probability on the draw of X, if q is chosen according to our algorithm then

$$\frac{\|Xw - Xq\|_2}{\|Xw\|_2} \lesssim_\varepsilon \frac{\sqrt{m}\log(N_0)}{\|w\|_2}$$

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$. Suppose $X \in \mathbb{R}^{m \times N_0}$ has independent columns $X_t \sim \mathscr{N}(0,\sigma^2 I_{m\times m})$, $w \in B_\infty^{N_0}$ satisfies $\mathrm{dist}(w_t, \{-1,0,1\}) > \varepsilon$ for all t. Then with high probability on the draw of X, if q is chosen according to our algorithm then

$$\frac{\|Xw - Xq\|_2}{\|Xw\|_2} \lesssim_\varepsilon \frac{\sqrt{m}\log(N_0)}{\|w\|_2}$$

Overparameterization Helps

- For general w, $\|w\|_2 \propto \sqrt{N_0}$
- Up to log terms, relative training error decays like $\sqrt{\frac{m}{N_0}}$

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$. Suppose $X \in \mathbb{R}^{m\times N_0}$ has independent columns $X_t \sim \mathcal{N}(0,\sigma^2 I_{m\times m})$, $w \in B_\infty^{N_0}$ satisfies $\mathrm{dist}(w_t, \{-1,0,1\}) > \varepsilon$ for all t. Then with high probability on the draw of X, if q is chosen according to our algorithm then

$$\frac{\|Xw - Xq\|_2}{\|Xw\|_2} \lesssim_\varepsilon \frac{\sqrt{m}\log(N_0)}{\|w\|_2}$$

Corollary

If the feature vectors X_t are drawn from a d-dimensional subspace, i.e. $X_t = ZA_t$, $A_t \sim \mathcal{N}(0,\sigma^2 I_{d\times d})$, $Z \in \mathbb{R}^{N\times d}$, $Z^TZ = I$, then with high probability

$$\frac{\|Xw - Xq\|_2}{\|Xw\|_2} \lesssim_\varepsilon \frac{\sqrt{d}\log(N_0)}{\|w\|_2}$$

Motivating Questions

- For every **single layer** neural network, does there exist a quantized neural network that approximates it well?
  - If yes,… 
  - can it be constructed in a reasonable amount of time?
  - does it generalize to new data?

Motivating Questions

- For every **single layer** neural network, does there exist a quantized neural network that approximates it well **on Gaussian data**?
  - If yes,… 
  - can it be constructed in a reasonable amount of time?
  - does it generalize to new data?

Motivating Questions

- For every **single layer** neural network, does there exist a quantized neural network that approximates it well **on Gaussian data**?
  - If yes,… ✓
  - can it be constructed in a reasonable amount of time? ✓
  - does it generalize to new data?

Theorem (L., Saab, 2020)

Suppose $N_0 \gg m$. Let $X = U\Sigma V^T$ be the SVD of X, and let $z = Vg$ where $g \sim \mathcal{N}(0, \sigma_z^2 I_{m\times m})$ is drawn independently of X, w so that $\mathbb{E}[\|z\|_2^2 | V] = \mathbb{E}\|x_i\|_2^2 = \sigma^2 N_0$. Then with high probability

$$|z^T(w-q)| \lesssim \sqrt{m}\ \underbrace{\sigma m \log(N_0)}_{\text{training error}}\ .$$

Theorem (L., Saab, 2020)

Suppose $N_0 \gg m$. Let $X = U\Sigma V^T$ be the SVD of X, and let $z = Vg$ where $g \sim \mathcal{N}(0, \sigma_z^2 I_{m\times m})$ is drawn independently of X, w so that $\mathbb{E}[\|z\|_2^2 | V] = \mathbb{E}\|x_i\|_2^2 = \sigma^2 N_0$. Then with high probability

$$|z^T(w-q)| \lesssim \sqrt{m}\ \underbrace{\sigma m \log(N_0)}_{\text{training error}}\ .$$

Corollary

If the feature vectors X_t are drawn from a d-dimensional subspace, i.e. $X_t = ZA_t$, $A_t \sim \mathcal{N}(0, \sigma^2 I_{d\times d})$, $Z \in \mathbb{R}^{N\times d}$, $Z^TZ = I$, then with high probability

$$|z^T(w-q)| \lesssim \sqrt{d}\ \underbrace{\sigma d \log(N_0)}_{\text{training error}}\ .$$

Motivating Questions

- For every **single layer** neural network, does there exist a quantized neural network that approximates it well **on Gaussian data**?
  - If yes,… ✓
  - can it be constructed in a reasonable amount of time? ✓
  - does it generalize to new data?

Motivating Questions

- For every **single layer** neural network, does there exist a quantized neural network that approximates it well **on Gaussian data**?
  - If yes,… ✓
  - can it be constructed in a reasonable amount of time? ✓
  - does it generalize to new data? ✓

Numerics

$$\|Q^{(1)}(X) - W^{(1)}(X)\|_F^2$$

$$\|Q^{(2)} \circ \varphi \circ Q^{(1)}(X) - W^{(2)} \circ \varphi \circ W^{(1)}(X)\|_F^2$$

$$\| Q^{(2)} \circ \underbrace{\varphi \circ Q^{(1)}(X)}_{:=\widetilde{Y}} - W^{(2)} \circ \underbrace{\varphi \circ W^{(1)}(X)}_{:=Y} \|_F^2$$

Numerics

- We can use the same idea to quantize ***hidden*** layers
  - Quantized walk now uses ***perturbed directions*** to chase analog walk

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $\Phi^{(\ell)}$, $\widetilde{\Phi}^{(\ell)}$ analog and quantized networks up to layer ℓ, $w \in B_\infty^{N_0}$, $\mathscr{A}$, for $t \in \{1,\ldots,N_0\}$, set $Y := \Phi^{(\ell)}(X), \widetilde{Y} := \widetilde{\Phi}^{(\ell)}(X)$

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \arg\min_{p \in \mathscr{A}} \|u_{t-1} + w_t Y_t - p\widetilde{Y}_t\|_2^2,$$

$$u_t := u_{t-1} + w_t Y_t - q_t \widetilde{Y}_t$$

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $\Phi^{(\ell)}$, $\widetilde{\Phi}^{(\ell)}$ analog and quantized networks up to layer ℓ, $w \in B_\infty^{N_0}$, $\mathscr{A}$, for $t \in \{1,\ldots,N_0\}$, set $Y := \Phi^{(\ell)}(X)$, $\widetilde{Y} := \widetilde{\Phi}^{(\ell)}(X)$

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \arg\min_{p \in \mathscr{A}} \|u_{t-1} + w_t Y_t - p\widetilde{Y}_t\|_2^2,$$

$$u_t := u_{t-1} + w_t Y_t - q_t \widetilde{Y}_t$$

- For general alphabets, cross validate over two parameters:

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $\Phi^{(\ell)}$, $\widetilde{\Phi}^{(\ell)}$ analog and quantized networks up to layer ℓ, $w \in B_\infty^{N_0}$, $\mathcal{A}$, for $t \in \{1,\ldots,N_0\}$, set $Y := \Phi^{(\ell)}(X)$, $\widetilde{Y} := \widetilde{\Phi}^{(\ell)}(X)$

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \arg\min_{p \in \mathcal{A}} \|u_{t-1} + w_t Y_t - p\widetilde{Y}_t\|_2^2,$$

$$u_t := u_{t-1} + w_t Y_t - q_t \widetilde{Y}_t$$

- For general alphabets, cross validate over two parameters:
  - Size of ***equispaced*** $\mathcal{A} := \{-1 + \frac{2j}{M-1} : j \in \{0,1,\ldots,M-1\}\} \subset [-1,1]$

Algorithm

Given $X \in \mathbb{R}^{m \times N_0}$, $\Phi^{(\ell)}$, $\widetilde{\Phi}^{(\ell)}$ analog and quantized networks up to layer ℓ, $w \in B_\infty^{N_0}$, $\mathscr{A}$, for $t \in \{1,\ldots,N_0\}$, set $Y := \Phi^{(\ell)}(X)$, $\widetilde{Y} := \widetilde{\Phi}^{(\ell)}(X)$

$$u_0 := 0 \in \mathbb{R}^m,$$

$$q_t := \arg\min_{p \in \mathscr{A}} \|u_{t-1} + w_t Y_t - p\widetilde{Y}_t\|_2^2,$$

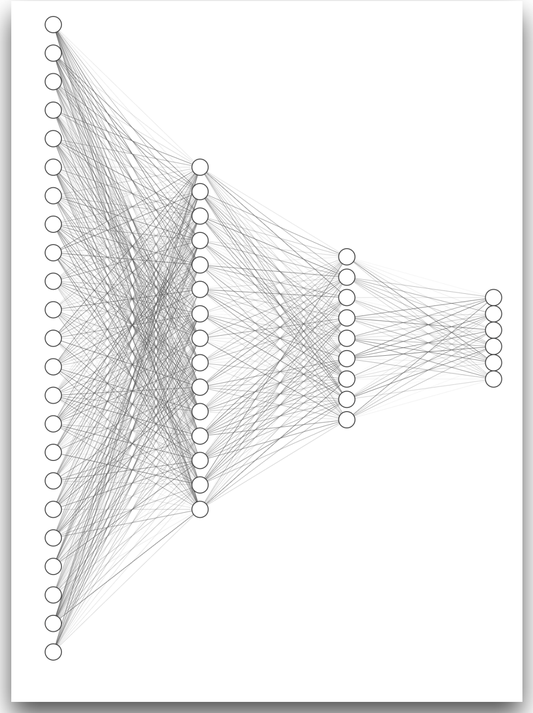
$$u_t := u_{t-1} + w_t Y_t - q_t \widetilde{Y}_t$$

- For general alphabets, cross validate over two parameters:
  - Size of equispaced $\mathscr{A} := \{-1 + \frac{2j}{M-1} : j \in \{0,1,\ldots,M-1\}\} \subset [-1,1]$
  - Radius of $\mathscr{A}$ at layer ℓ $\alpha_\ell := C_\alpha \operatorname{median}\left(|W_{i,j}^{(\ell)}| \right)$, $C_\alpha \in \{1,\ldots,10\}$
    - $\mathscr{A}_\ell := \alpha_\ell \times \mathscr{A}$

MNIST Perceptron

MNIST Perceptron

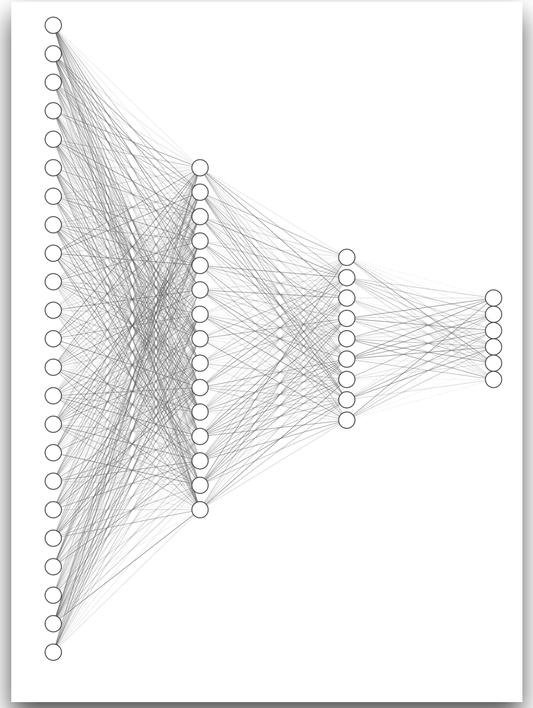
- We trained a multilayer perceptron on MNIST



$$\mathbb{R}^{784} \rightarrow \mathbb{R}^{500} \rightarrow \mathbb{R}^{300} \rightarrow \{0,1,...,9\}$$

MNIST Perceptron

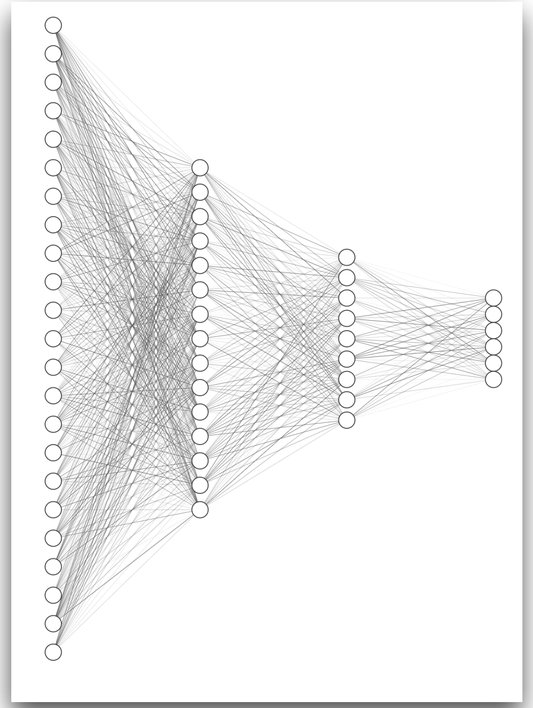
- We trained a multilayer perceptron on MNIST
- Analog training: 50,000 digits
- Quantization training: 25,000 digits
  - $\mathscr{A} = \{-1,0,1\}$



$$\mathbb{R}^{784} \to \mathbb{R}^{500} \to \mathbb{R}^{300} \to \{0,1,...,9\}$$

MNIST Perceptron

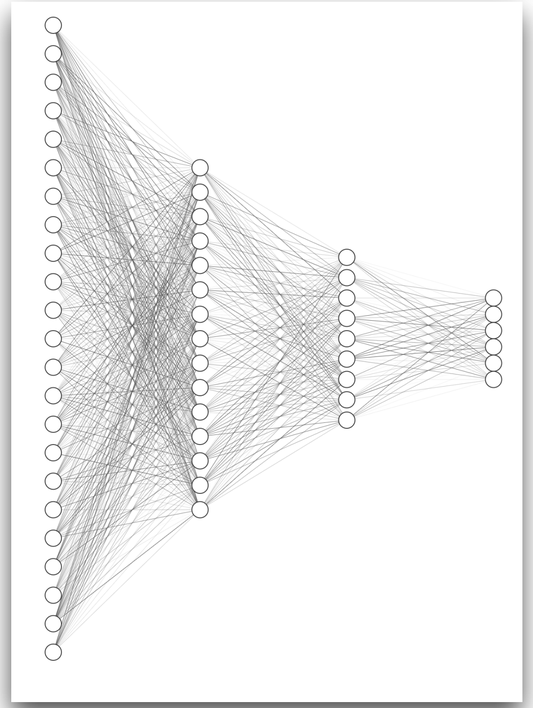
- We trained a multilayer perceptron on MNIST
- Analog training: 50,000 digits
- Quantization training: 25,000 digits
  - $\mathscr{A} = \{-1,0,1\}$
- Test data: 10,000 digits



$$\mathbb{R}^{784} \to \mathbb{R}^{500} \to \mathbb{R}^{300} \to \{0,1,...,9\}$$

MNIST Perceptron

- We trained a multilayer perceptron on MNIST
- Analog training: 50,000 digits
- Quantization training: 25,000 digits
  - $\mathscr{A} = \{-1,0,1\}$
- Test data: 10,000 digits
- Compare to simple “round weights” quantization (MSQ)



$$\mathbb{R}^{784} \rightarrow \mathbb{R}^{500} \rightarrow \mathbb{R}^{300} \rightarrow \{0,1,...,9\}$$

MNIST Perceptron

How sensitive is quantization to choice of C_α in radius $\alpha_\ell := C_\alpha \operatorname{median}\left(|W_{i,j}^{(\ell)}| \right)$?

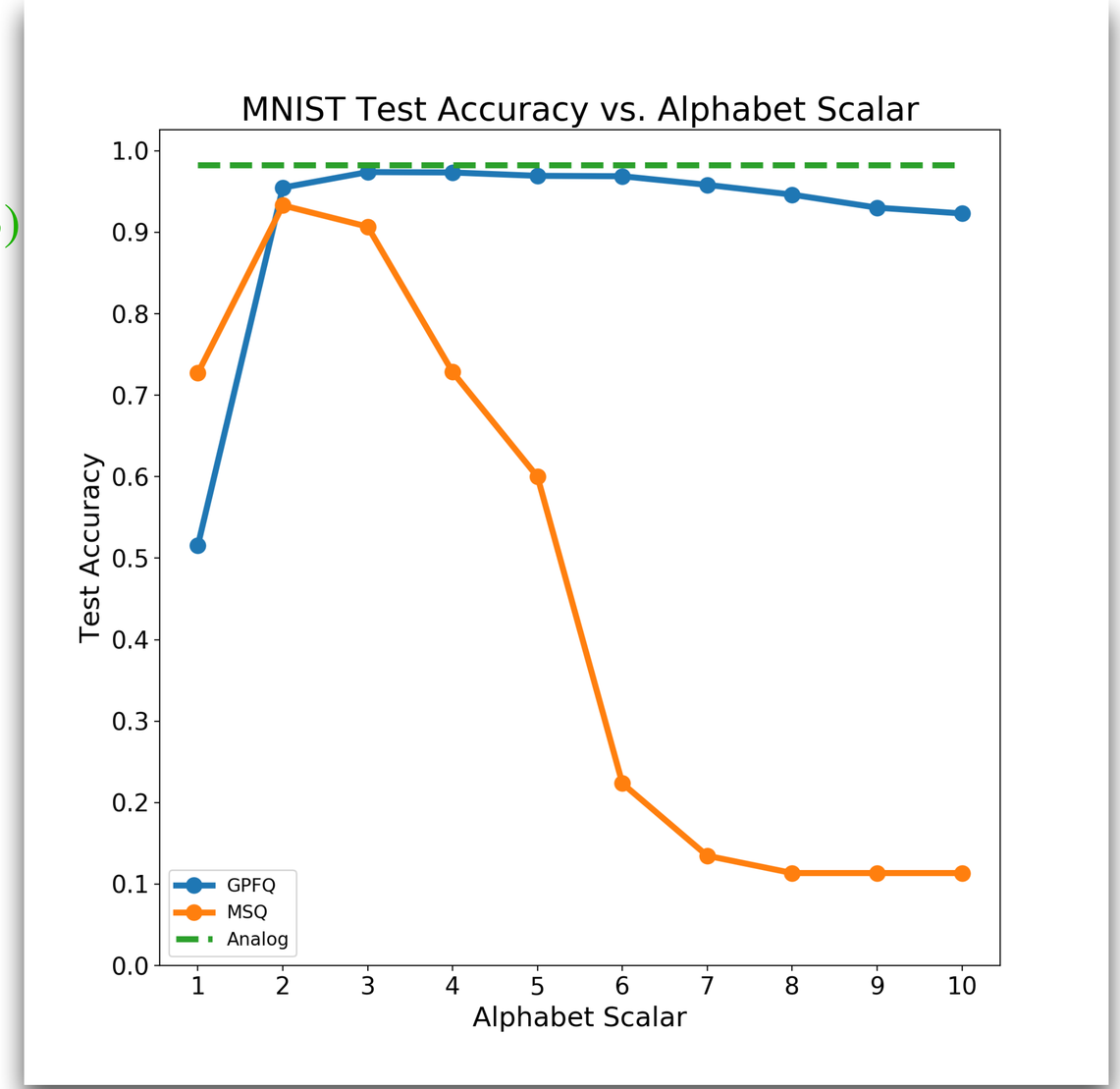
MNIST Perceptron

How sensitive is quantization to choice of C_α in radius $\alpha_\ell := C_\alpha \operatorname{median}\left(|W_{i,j}^{(\ell)}| \right)$?

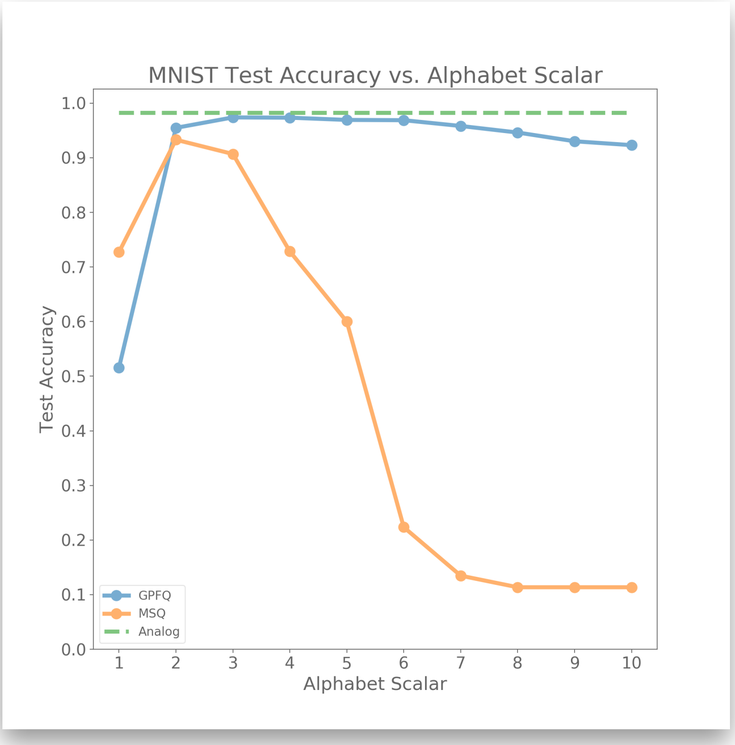
Analog: 98.24%

Best GPFQ: 97.39% $(\Delta = -0.85\%)$

Best MSQ: 93.3% $(\Delta = -4.94\%)$



MNIST Perceptron



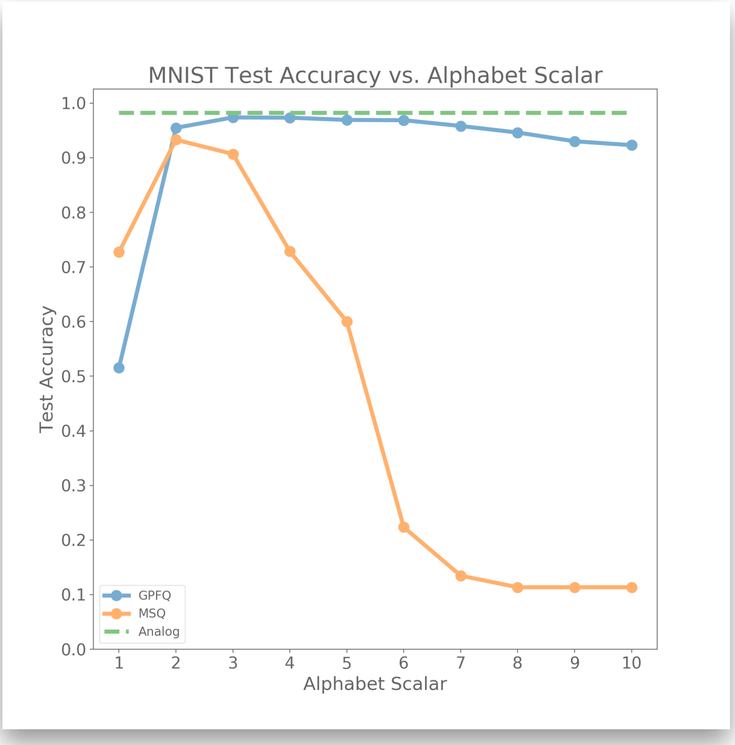
How does the test accuracy decay across layers quantized?

Analog: 98.24%

Best GPFQ: 97.39% ($\Delta = -0.85\%$)

Best MSQ: 93.3% ($\Delta = -4.94\%$)

MNIST Perceptron

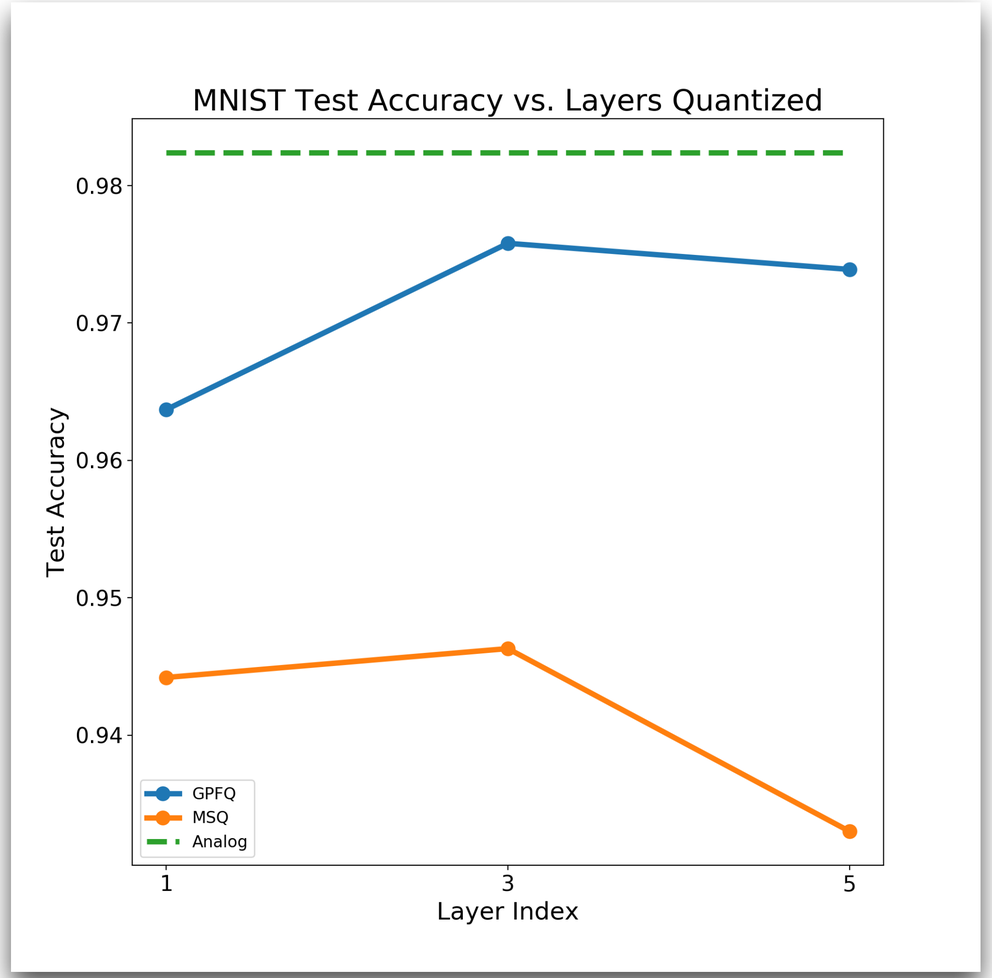


Analog: 98.24%

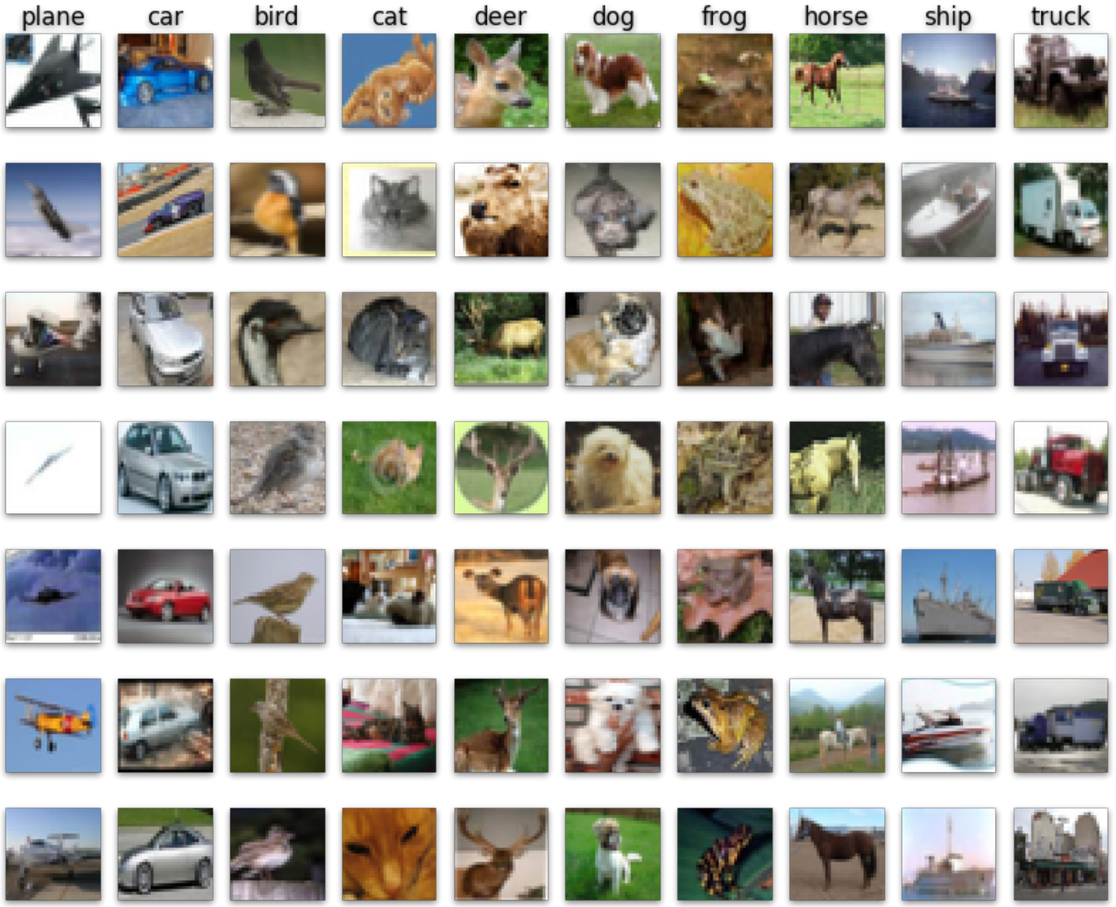
Best GPFQ: 97.39% ($\Delta = -0.85\%$)

Best MSQ: 93.3% ($\Delta = -4.94\%$)

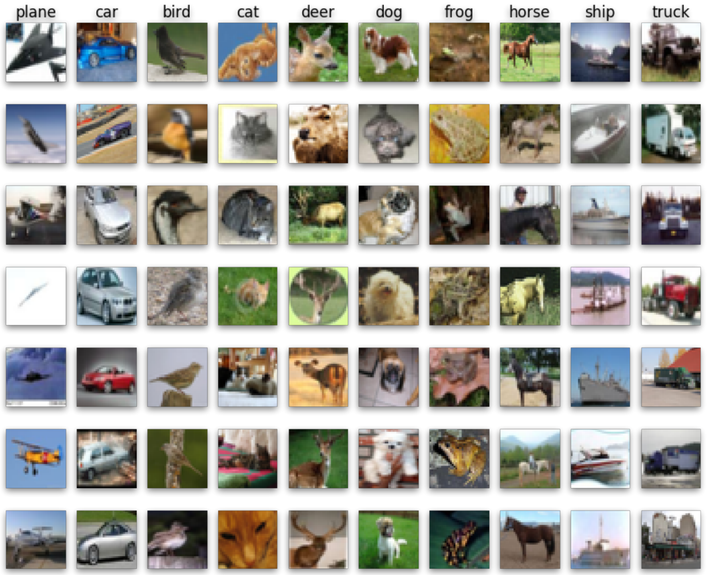
How does the test accuracy decay across layers quantized?



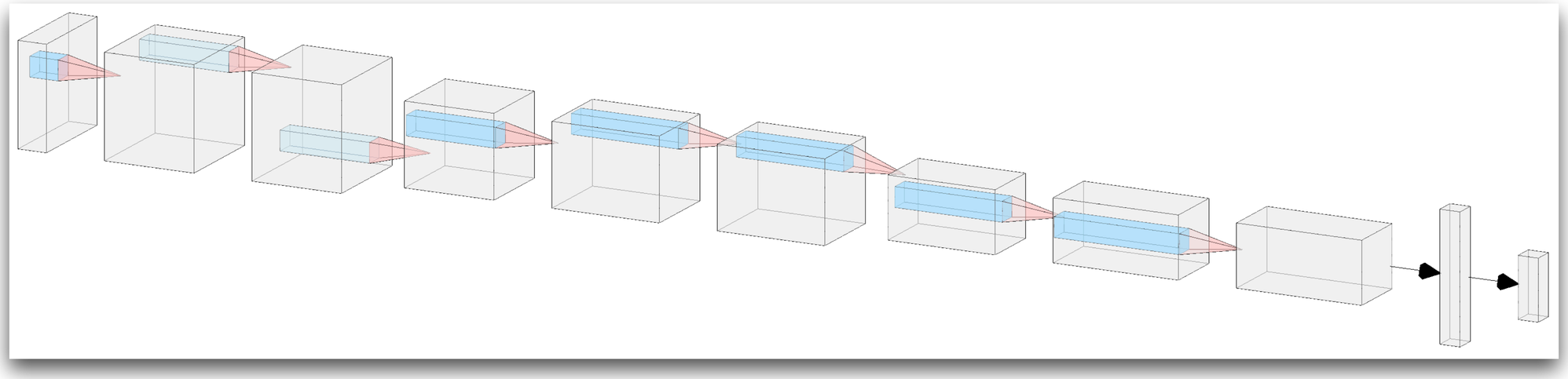
CIFAR10 ConvNet



CIFAR10 ConvNet



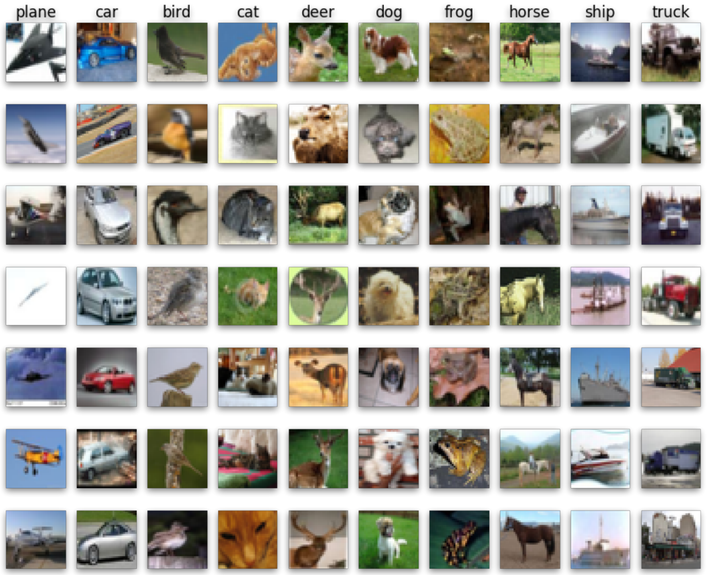
- We trained a Convolutional Neural Network
  - Linear maps $W : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^{m_1 \times m_2 \times k}$ are ***filter banks*** of k ***learned*** filters

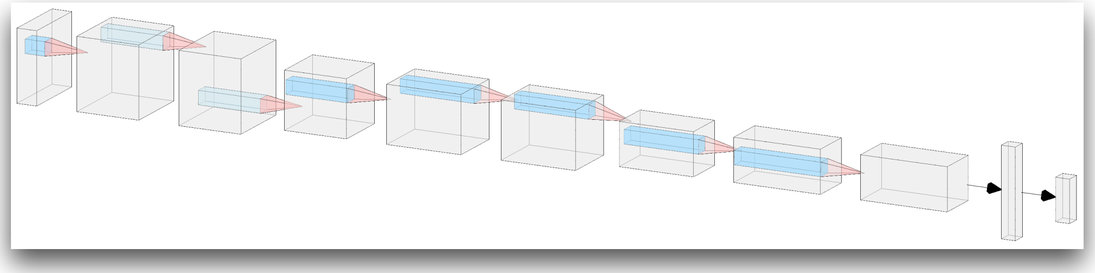


$$2 \times 32C3 \to MP2 \to 2 \times 64C3 \to MP2 \to 2 \times 128C3 \to 128FC \to 10FC$$

CIFAR10 ConvNet

- We trained a Convolutional Neural Network
  - Linear maps $W : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^{m_1 \times m_2 \times k}$ are ***filter banks*** of k ***learned*** filters
- Analog training: 50,000 images
- Quantization training: 5,000 images
  - Cross-validated size & radius of $\mathscr{A}$
- Test data: 10,000 images



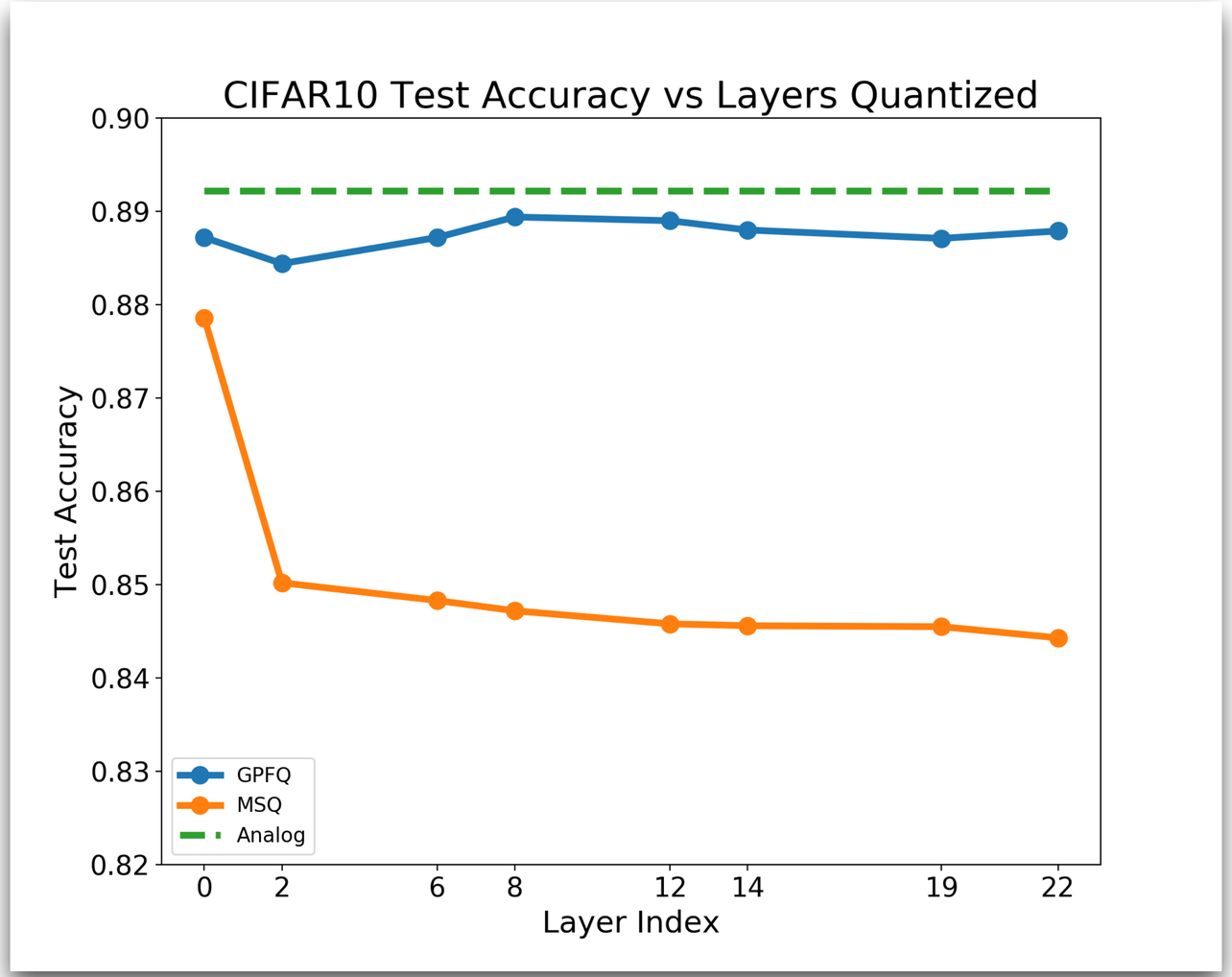


$2 \times 32C3 \to MP2 \to 2 \times 64C3 \to MP2 \to 2 \times 128C3 \to 128FC \to 10FC$

CIFAR10 ConvNet

Bits	Analog	Best GPFQ	Best MSQ
$\log_2(3)$	89.22 %	74.87 %	14.64 %
2	89.22 %	80.36 %	28.0 %
3	89.22 %	87.40 %	56.52 %
4	89.22 %	88.88 %	84.43 %

CIFAR10 ConvNet



Motivating Questions

- For every **single layer** neural network, does there exist a quantized neural network that approximates it well **on Gaussian data**?
  - If yes,… ✓
  - can it be constructed in a reasonable amount of time? ✓
  - does it generalize to new data? ✓

Motivating Questions

- For ~~every single layer~~ neural network, does there exist a quantized neural network that approximates it well ~~on Gaussian data~~?
  - If yes,… Potentially!
  - can it be constructed in a reasonable amount of time? ✓
  - does it generalize to new data? Empirically, yes!

Recap

Recap

- A Relevant Math problem
- Novel algorithm with theoretical guarantees
- Incredibly promising numerical experiments

Recap

- A Relevant Math problem
- Novel algorithm with theoretical guarantees
- Incredibly promising numerical experiments

Preprints

- E. Lybrand and R. Saab. “A Greedy Algorithm for Quantizing Neural Networks.” preprint, 2020. arXiv GitHub
- E. Lybrand, A. Ma, and R. Saab. “On the Number of Faces and Radii of Cells Induced by Gaussian Spherical Tessellations.” preprint, 2020.
- H. Huang and T. Kemp and Y. Ling and X. Luo and E. Lybrand and R. Smith and J. Wang. “Random Matrices with Independent Diagonals.” preprint, 2018.

Publications

- M. Iwen, E. Lybrand, A. Nelson, R. Saab. “New Algorithms and Improved Guarantees for One-Bit Compressed Sensing on Manifolds.” SampTA2019. arXiv Proceedings
- E. Lybrand and R. Saab. “Quantization for Low-Rank Matrix Recovery.” Information and Inference, 2018. arXiv Journal

Thanks for your attention!

Appendix

Theorem (L., Saab, 2020)

Let $\mathscr{A} = \{-1,0,1\}$. Suppose $X \in \mathbb{R}^{m\times N_0}$ has independent columns $X_t \sim \mathcal{N}(0,\sigma^2 I_{m\times m})$, $w \in B_\infty^{N_0}$ satisfies $\mathrm{dist}(w_t, \{-1,0,1\}) > \varepsilon$ for all t. Then with high probability on the draw of X, if q is chosen using the columns of X then

$$\|Xw - Xq\|_2 \lesssim_\varepsilon \sigma m \log(N_0)$$

- Bound via moment generating function
- Key ingredient: increments of the process $\|u_t\|_2^2$

$$\Delta\|u_t\|_2^2 := \|u_t\|_2^2 - \|u_{t-1}\|_2^2 = (w_t - q_t)^2\|X_t\|_2^2 + 2(w_t - q_t)\langle X_t, u_{t-1}\rangle$$

$$\Delta \|u_t\|_2^2 := \|u_t\|_2^2 - \|u_{t-1}\|_2^2 = (w_t - q_t)^2 \|X_t\|_2^2 + 2(w_t - q_t)\langle X_t, u_{t-1}\rangle$$

$$\Delta\|u_t\|_2^2 := \|u_t\|_2^2 - \|u_{t-1}\|_2^2 = (w_t - q_t)^2\|X_t\|_2^2 + 2(w_t - q_t)\langle X_t, u_{t-1}\rangle$$

$$\mathbb{P}\left(\|u_N\|_2^2 > \alpha\right) \leq e^{-\lambda\alpha}\mathbb{E}[e^{\lambda\|u_N\|_2^2}] = e^{-\lambda\alpha}\mathbb{E}\left[\mathbb{E}[e^{\lambda\|u_N\|_2^2} \,|\, \mathscr{F}_{N-1}]\right] = e^{-\lambda\alpha}\mathbb{E}\left[e^{\lambda\|u_{N-1}\|_2^2}\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2} \,|\, \mathscr{F}_{N-1}]\right]$$

$$\Delta\|u_t\|_2^2 := \|u_t\|_2^2 - \|u_{t-1}\|_2^2 = (w_t - q_t)^2\|X_t\|_2^2 + 2(w_t - q_t)\langle X_t, u_{t-1}\rangle$$

$$\mathbb{P}\left(\|u_N\|_2^2 > \alpha\right) \le e^{-\lambda\alpha}\mathbb{E}[e^{\lambda\|u_N\|_2^2}] = e^{-\lambda\alpha}\mathbb{E}\left[\mathbb{E}[e^{\lambda\|u_N\|_2^2} \mid \mathscr{F}_{N-1}]\right] = e^{-\lambda\alpha}\mathbb{E}\left[e^{\lambda\|u_{N-1}\|_2^2}\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2} \mid \mathscr{F}_{N-1}]\right]$$

Large running error

Small running error

$$\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2}1(\|u_{N-1}\|_2^2 \ge \beta) \mid \mathscr{F}_{N-1}]$$

$$\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2}1(\|u_{N-1}\|_2^2 < \beta) \mid \mathscr{F}_{N-1}]$$

$$\Delta\|u_t\|_2^2 := \|u_t\|_2^2 - \|u_{t-1}\|_2^2 = (w_t - q_t)^2\|X_t\|_2^2 + 2(w_t - q_t)\langle X_t, u_{t-1}\rangle$$

$$\mathbb{P}\left(\|u_N\|_2^2 > \alpha\right) \le e^{-\lambda\alpha}\mathbb{E}[e^{\lambda\|u_N\|_2^2}] = e^{-\lambda\alpha}\mathbb{E}\left[\mathbb{E}[e^{\lambda\|u_N\|_2^2} \mid \mathscr{F}_{N-1}]\right] = e^{-\lambda\alpha}\mathbb{E}\left[e^{\lambda\|u_{N-1}\|_2^2}\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2} \mid \mathscr{F}_{N-1}]\right]$$

Large running error

Small running error

$$\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2}1(\|u_{N-1}\|_2^2 \ge \beta) \mid \mathscr{F}_{N-1}]$$

$$\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2}1(\|u_{N-1}\|_2^2 < \beta) \mid \mathscr{F}_{N-1}]$$

$$\Delta\|u_t\|_2^2 := \|u_t\|_2^2 - \|u_{t-1}\|_2^2 = (w_t - q_t)^2\|X_t\|_2^2 + 2(w_t - q_t)\langle X_t, u_{t-1}\rangle$$

$$\mathbb{P}\left(\|u_N\|_2^2 > \alpha\right) \le e^{-\lambda\alpha}\mathbb{E}[e^{\lambda\|u_N\|_2^2}] = e^{-\lambda\alpha}\mathbb{E}\left[\mathbb{E}[e^{\lambda\|u_N\|_2^2} \mid \mathscr{F}_{N-1}]\right] = e^{-\lambda\alpha}\mathbb{E}\left[e^{\lambda\|u_{N-1}\|_2^2}\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2} \mid \mathscr{F}_{N-1}]\right]$$

Large running error

Small running error

$$\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2}1(\|u_{N-1}\|_2^2 \ge \beta) \mid \mathscr{F}_{N-1}]$$

$$\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2}1(\|u_{N-1}\|_2^2 < \beta) \mid \mathscr{F}_{N-1}]$$

$$\Delta\|u_t\|_2^2 := \|u_t\|_2^2 - \|u_{t-1}\|_2^2 = (w_t - q_t)^2\|X_t\|_2^2 + 2(w_t - q_t)\langle X_t, u_{t-1}\rangle$$

$$\mathbb{P}\left(\|u_N\|_2^2 > \alpha\right) \le e^{-\lambda\alpha}\mathbb{E}[e^{\lambda\|u_N\|_2^2}] = e^{-\lambda\alpha}\mathbb{E}\left[\mathbb{E}[e^{\lambda\|u_N\|_2^2} \mid \mathscr{F}_{N-1}]\right] = e^{-\lambda\alpha}\mathbb{E}\left[e^{\lambda\|u_{N-1}\|_2^2}\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2} \mid \mathscr{F}_{N-1}]\right]$$

Large running error

Small running error

$$\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2}1(\|u_{N-1}\|_2^2 \ge \beta) \mid \mathscr{F}_{N-1}]$$

$$\mathbb{E}[e^{\lambda\Delta\|u_N\|_2^2}1(\|u_{N-1}\|_2^2 < \beta) \mid \mathscr{F}_{N-1}]$$

Very careful analysis:

- Discontinuous q_t…handle each case separately
- Careful decomposition of multivariable integral
  - No *a priori* bound on $\|u_{t-1}\|_2^2$…m.g.f. could “explode”

Brex

Ambika Kumar • 1st
Head of Fraud @ Brex
1yr • Edited •

Today marks 1 year at Brex! It has been the most amazing experience so far, and if I could, I would hire everyone to join our team just for the breadth of experience.

...see more