DATA-DRIVEN ALGORITHMS FOR GAUSSIAN MEASUREMENT MATRIX DESIGN IN COMPRESSIVE SENSING

Yang Sun[1] and Jonathan Scarlett[1,2]

[1]Department of Computer Science, National University of Singapore

[2]Department of Mathematics & Institute of Data Science, National University of Singapore

HIGHLIGHTS

Two novel compressive sensing measurement matrix designs focusing more energy on the "most important" parts of the signal:

- **Variance-Proportional Sampling:** place more energy at locations where the signal tends to vary more
- **Iteratively-Learned Power Allocation:** iteratively up-weigh or down-weigh the energy according to reconstruction quality

SETUP

- Recover $\mathbf{x} \in \mathbb{R}^n$ via linear measurements $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{z}$
- Given observation $\mathbf{y} \in \mathbb{R}^\ell$, matrix $\mathbf{A} \in \mathbb{R}^{\ell \times n}$, and noise $\mathbf{z} \in \mathbb{R}^\ell$
- High-dimensional regime $\ell \ll n$ with structure on $\mathbf{x}$
- Find a decoder to estimate $\hat{\mathbf{x}}$
- Performance metric $\mathrm{MSE}(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{n}\|\mathbf{x} - \hat{\mathbf{x}}\|^2$

OBJECTIVE

- Each row $\mathbf{A}_i \sim \mathcal{N}(0, v_i)$ with different variances $(v_1, \ldots, v_n)$
- $\sum_{i=1}^n v_i = nP$ for some predefined "power level" P
- Assume we have access to **training data**
- **Our aim is to choose** $(v_1, \ldots, v_n)$ **using the training data**

VARIANCE-PROPORTIONAL SAMPLING

- First computes the empirical variance $\hat{\sigma}_i^2$ of each signal entry with respect to the training data
- Then chooses $v_i \propto \hat{\sigma}_i^2$, such that $v_i = nP \cdot \frac{\hat{\sigma}_i^2}{\sum_{i=1}^n \hat{\sigma}_j^2}$

ITERATIVELY-LEARNED POWER ALLOCATION

Initialize $(v_1, \ldots, v_n) = (P, \ldots, P)$ and for iterations $t = 1$ to T:

- Generate a random $\mathbf{A}$ according to current $(v_1, \ldots, v_n)$
- Calculate MSE of $(\mathbf{x}, \hat{\mathbf{x}})$ of each signal entry averaged over B training data
- Entries with the top αn MSE values: $v_i = v_i e^{-\lambda_t}$
  Entries with the bottom αn MSE values: $v_i = v_i e^{\lambda_t}$
  Entries for the rest: v_i unchanged
- Rescale $(v_1, \ldots, v_n)$ such that $\sum_{i=1}^n v_i = nP$

Hyperparameters details:

- Update proportion $\alpha \in \left(0, \frac{1}{2}\right)$
- Update extent $\lambda_t = \lambda_0 \cdot \eta^{t-1}$ with initial λ_0 and decay η
- Number of iterations T
- Batch size B

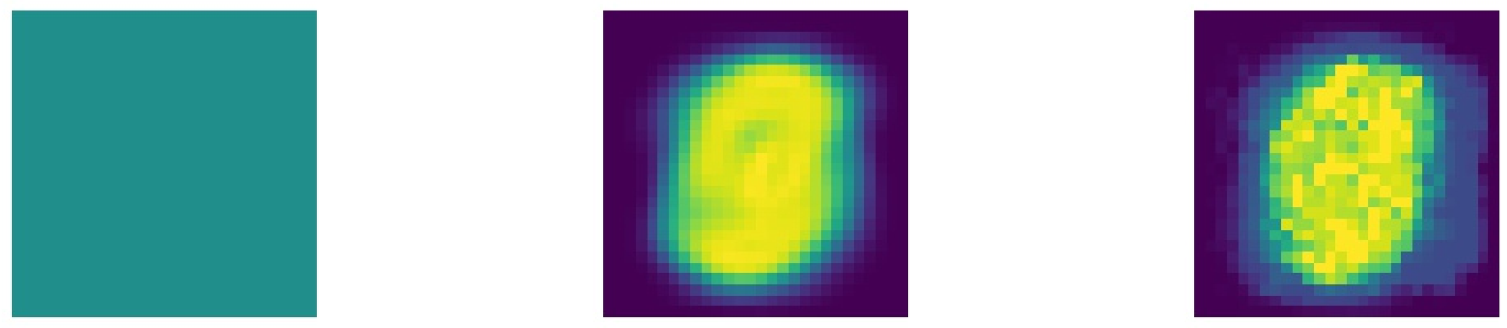
EXPERIMENT

- $P = 1$ such that $\mathbf{A} \sim \mathcal{N}(0, 1)$ as **baseline**
- $\alpha = \frac{1}{3}$; $\lambda_0 = 0.1$; $\eta = 0.95$; $T = 20$; B varies across the datasets
- Both **variance distribution map** and **recovery performance figure** are produced (shown left)

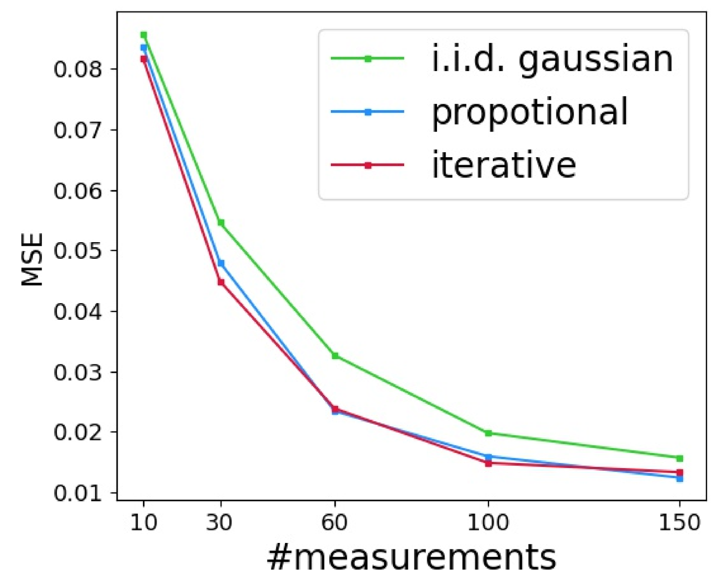
LINK

- For further references for the paper and code, please go to `https://github.com/ethangela/Data_Driven_Measurement_Matrix_Design`

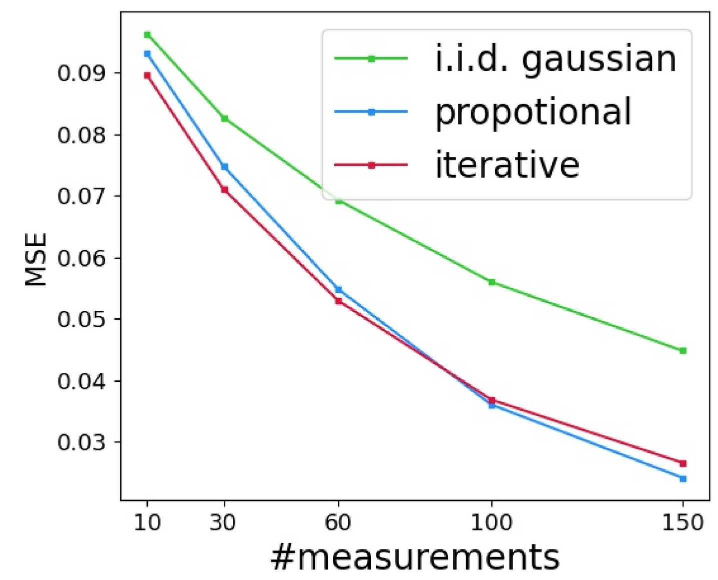
MNIST RESULT



i.i.d gaussian | proportional | iterative $\mathcal{N}(0,8)/\ell = 30$

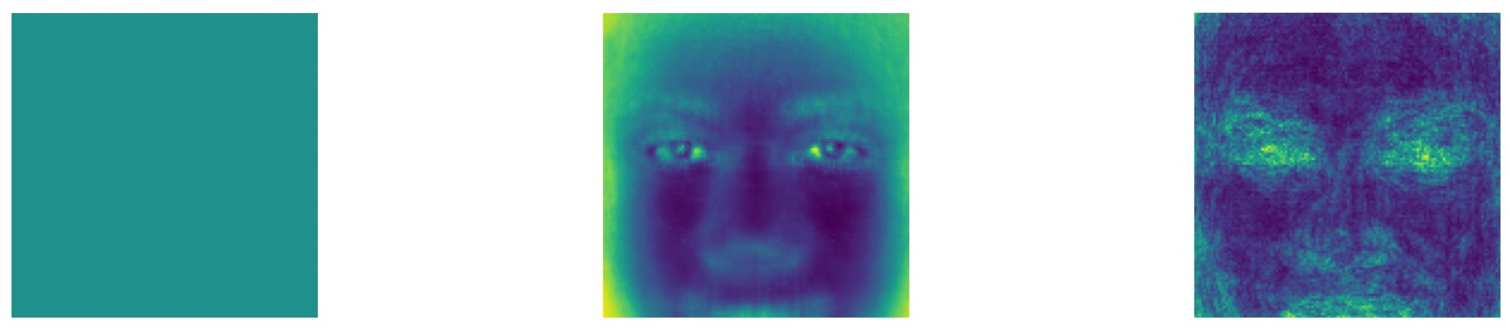


$z \sim \mathcal{N}(0, 3)$

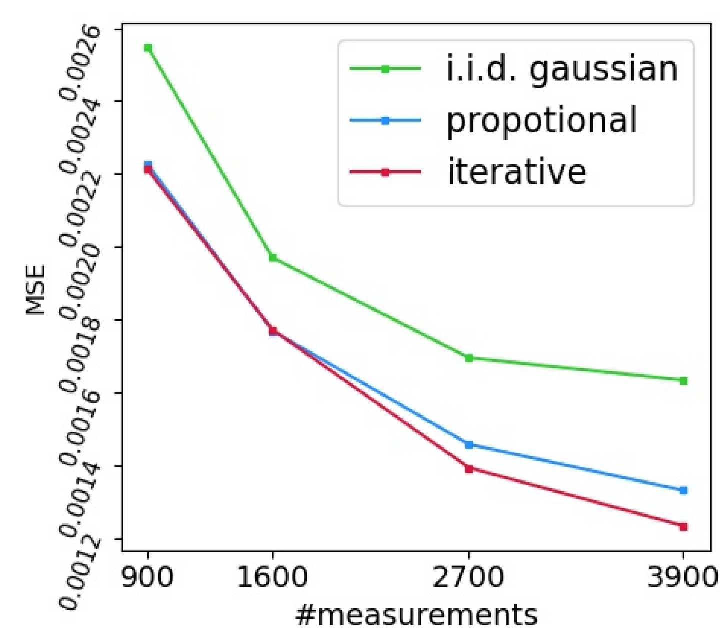


$z \sim \mathcal{N}(0, 15)$

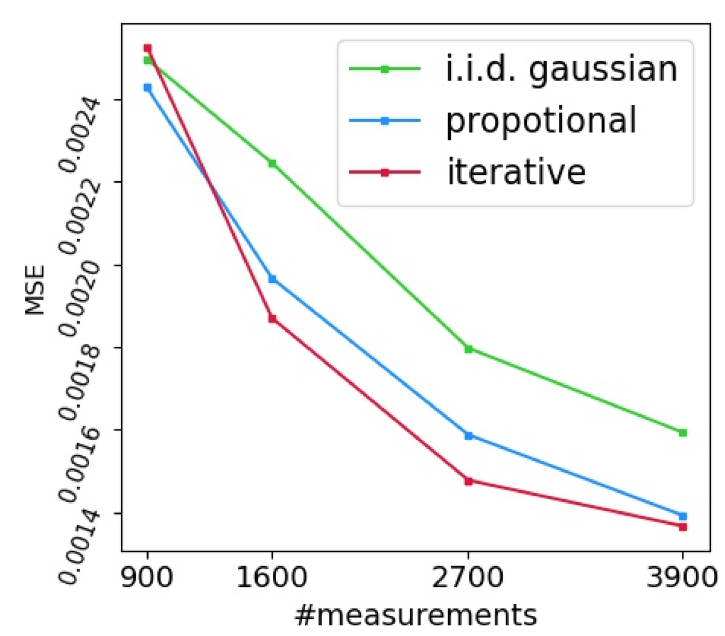
CELEBA-HQ RESULT



i.i.d gaussian | proportional | iterative $\mathcal{N}(0,16)/\ell = 900$

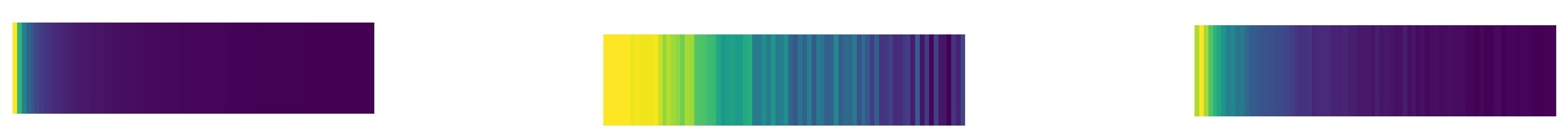


$z \sim \mathcal{N}(0, 8)$



$z \sim \mathcal{N}(0, 16)$

SYNTHETIC DATA RESULT



proportional | iterative | ℓ_1-AE