Learning a Compressed Sensing Measurement Matrix via Gradient Unrolling

Shanshan Wu[1], Alex Dimakis[1], Sujay Sanghavi[1], Felix Yu[2], Dan Holtmann-Rice[2], Dmitry Strocheus[2], Afshin Rostamizadeh[2], Sanjiv Kumar[2]

1. University of Texas at Austin 2. Google Research, New York

[Motivation]

- High-dimensional data are often sparse (see examples in Table 1).
- Unlike image/video data, there is no notion of spatial/time locality.
- Prefer dimensionality reduction via linear operation.

Goal: Can we learn a lossless linear sketch of high-dimensional sparse data?

[Problem formulation]

- Given data points $x_1, x_2, \ldots, x_n \in \mathbb{R}^d$ that are high-dimensional sparse and have additional (but unknown) structure in their support.
- We formulate the problem as learning a measurement matrix $A \in \mathbb{R}^{m\times d}$ $(m \ll d)$.
- Given A and the measurements $y_i = Ax_i \in \mathbb{R}^m$, x_i can be estimated as

$$f(A,y) \coloneqq \arg\min_{x' \in \mathbb{R}^d} \|x'\|_1 \quad \text{s.t.} \quad Ax' = y$$

[ℓ_1-min decoder]

- Our problem becomes

$$\min_{A \in \mathbb{R}^{m\times d}} \sum_{i=1}^{n} \|x_i - f(A, Ax_i)\|_2^2$$

Problem: How to compute gradient w.r.t. A?

[Our algorithm]

Key Idea Approximate $f(A,y)$ by T-step projected subgradient update.

$$x^{(t+1)} = \Pi\left(x^{(t)} - \alpha_t \operatorname{sign}(x^{(t)})\right) = x^{(t)} - \alpha_t (I - A^\dagger A)\operatorname{sign}(x^{(t)})$$

Π: projection onto $\{x: Ax = y\}$

α_t: step size at t-th iteration

$A^\dagger = A^T(AA^T)^{-1}$: Pseudoinverse

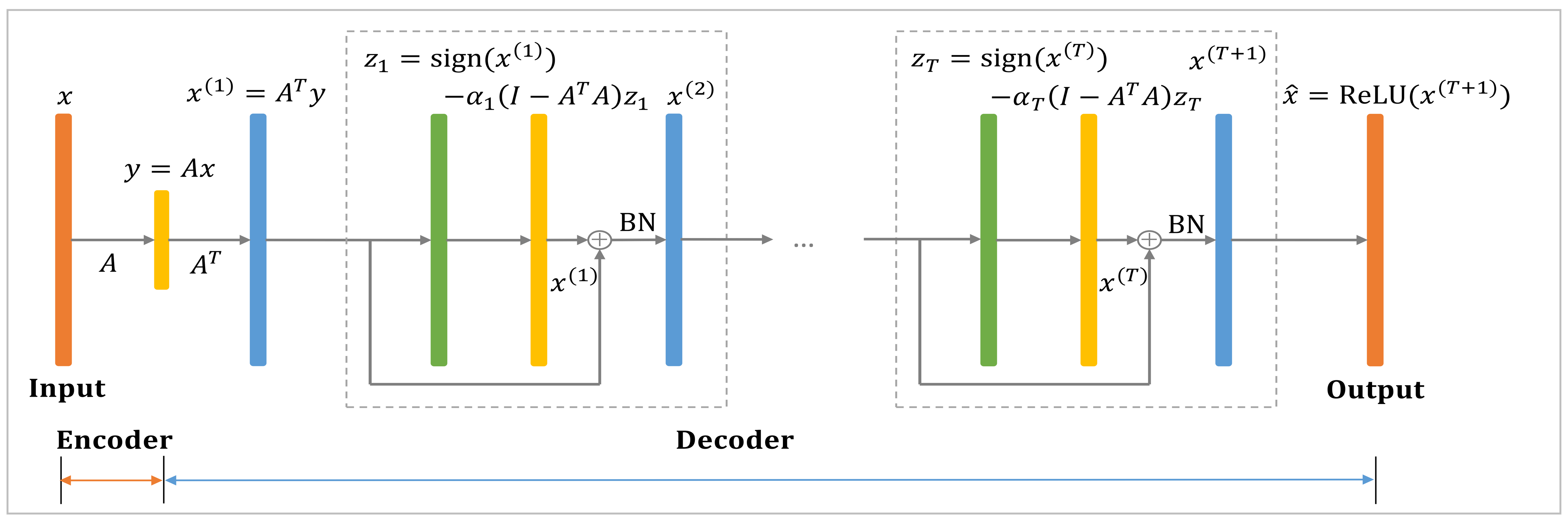

Figure 1. Network structure of our proposed autoencoder ℓ_1-AE.

Training ℓ_1-AE is trained to minimize the reconstruction error: $\min_{A,\alpha_t} \frac{1}{n}\sum_i \|x_i - \hat{x}_i\|_2^2$

[Experiments]

Code: https://github.com/wushanshan/L1AE

Table 1. Sparse datasets used in our experiments.

Dataset	Dimension	Avg NNZs	Train/valid/Test Size	Description
Synthetic 1	1000	10	6k/2k/2k	1-block sparse with block size 10
Synthetic 2	1000	10	6k/2k/2k	2-block sparse with block size 5
Synthetic 3	1000	10	6k/2k/2k	Power-law structured sparsity
Amazon	15626	9	19k/6k/6k	One-hot encoded categorical features
Wiki10-31K	30398	19	14k/3k/3k	Extreme multi-label data
RCV1	47236	76	13k/4k/4k	Bag-of-words data with TF-IDF features

We compared 10 algorithms over 2 metrics (evaluated on the test set):

1. **Fraction of recovered points:** $\|x - \hat{x}\|_2 \le 10^{-10}$
2. **Test RMSE:** $\sqrt{\sum \|x_i - \hat{x}_i\|_2^2 / n}$

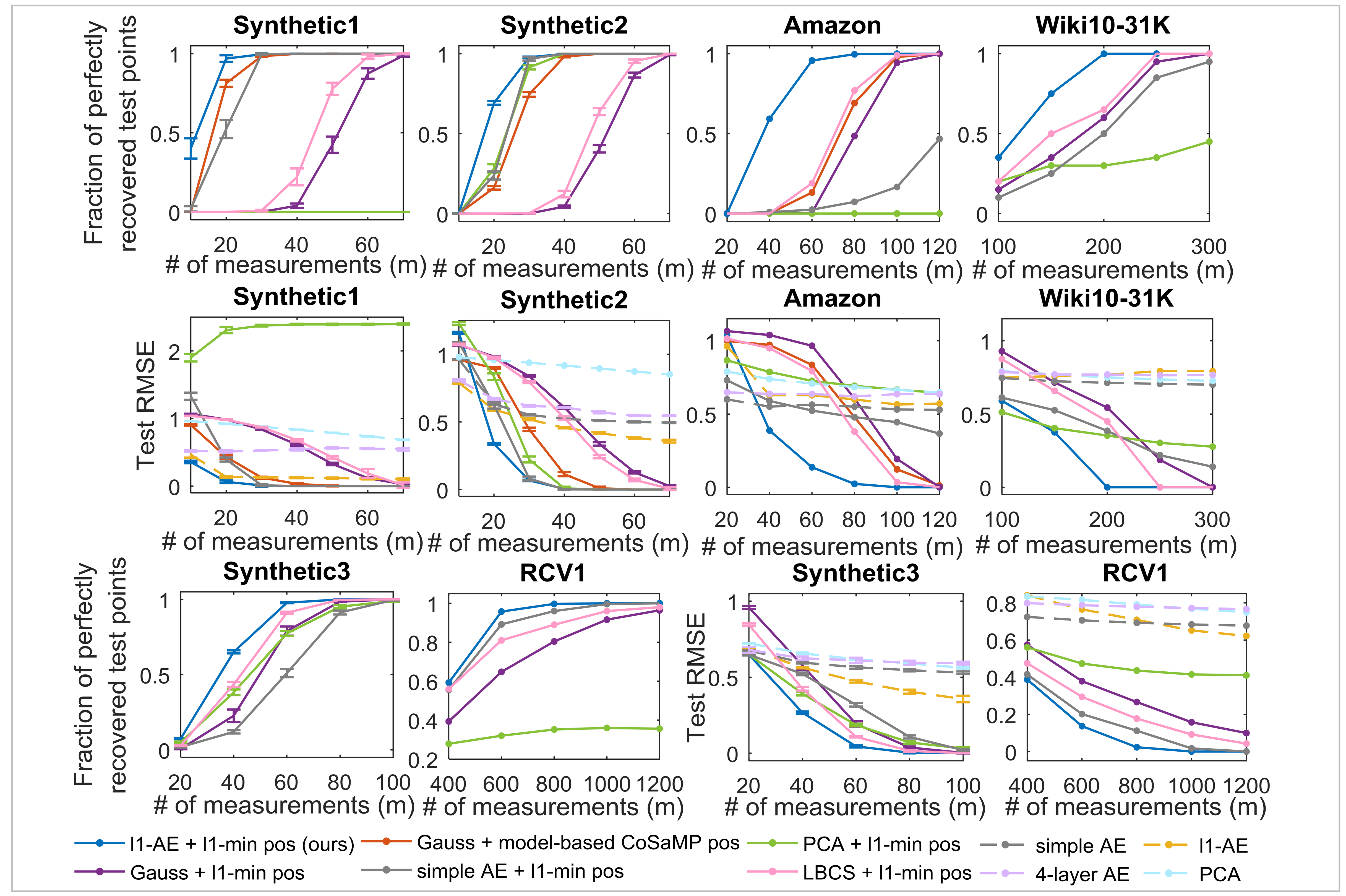

Figure 2. Our approach "ℓ_1-AE + ℓ_1-min pos" gives the best recovery performance.

Dataset	Amazon		
# measurements	40	80	120
ℓ_1-AE	0.638	0.599	0.565
ℓ_1-AE + ℓ_1-min pos (ours)	**0.387**	**0.023**	**2.8e-15**

Table 2. Test RMSE over the Amazon dataset. ℓ_1-min decoder achieves exact recovery.

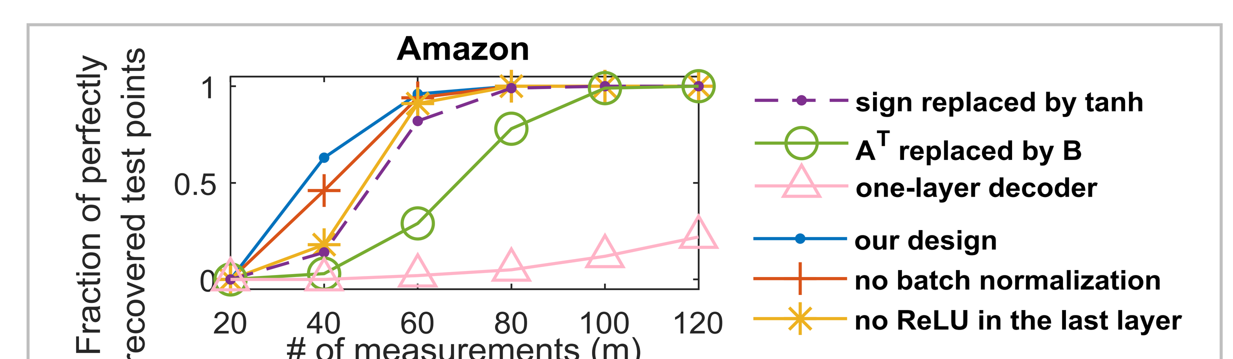

Figure 3. Our autoencoder performs the best among all variations.