Autoencoding Probabilistic Circuits

TECHNISCHE UNIVERSITÄT DARMSTADT

Steven Braun[1] Sahil Sidheekh[5]
Antonio Vergari[6] Sriraam Natarajan[5]
Martin Mundt[7] Kristian Kersting[1,2,3,4]

KompAKI

hessian.AI

[1]TU Darmstadt [2]hessian.AI [3]DFKI [4]Centre for Cog. Science, TU Darmstadt
[5]Univ. of Texas [6]Univ. of Edinburgh [7] Univ. of Bremen

Research Question

*Can we marry **tractable inference** of probabilistic circuits (PCs) and the modeling capacity of neural networks (NNs) for **representation learning**?*

Tractable Repr. Learning

Autoencoding Probabilistic Circuits (APCs) are hybrid autoencoders with a **PC encoder** and a **NN decoder**, trained end-to-end. PC models the joint distribution $p_{\mathcal{C}}(\mathbf{X}, \mathbf{Z})$, NN models the decoding function $p_\theta(\mathbf{X} \mid \mathbf{Z})$.

Encode: $\mathbf{z} \sim p_{\mathcal{C}}(\mathbf{Z} \mid \mathbf{X})$ **Decode**: $\mathbf{x} \sim p_\theta(\mathbf{X} \mid \mathbf{Z})$

End-to-End Training

- Enabled by **diff. PC sampling** with SIMPLE.
- Loss combines three objectives: $\mathcal{L} = \lambda_{\text{REC}}\mathcal{L}_{\text{REC}} + \lambda_{\text{KLD}}\mathcal{L}_{\text{KLD}} + \lambda_{\text{NLL}}\mathcal{L}_{\text{NLL}}$
- **Reconstruction**: Ensures embeddings $\mathbf{z}$ capture enough information for p_θ to reconstruct input $\mathbf{x}$

$$\mathcal{L}_{\text{REC}} = -\frac{1}{B}\sum_{i=1}^{B} \log p_\theta(\mathbf{x}_i \mid \mathbf{z}_i)$$

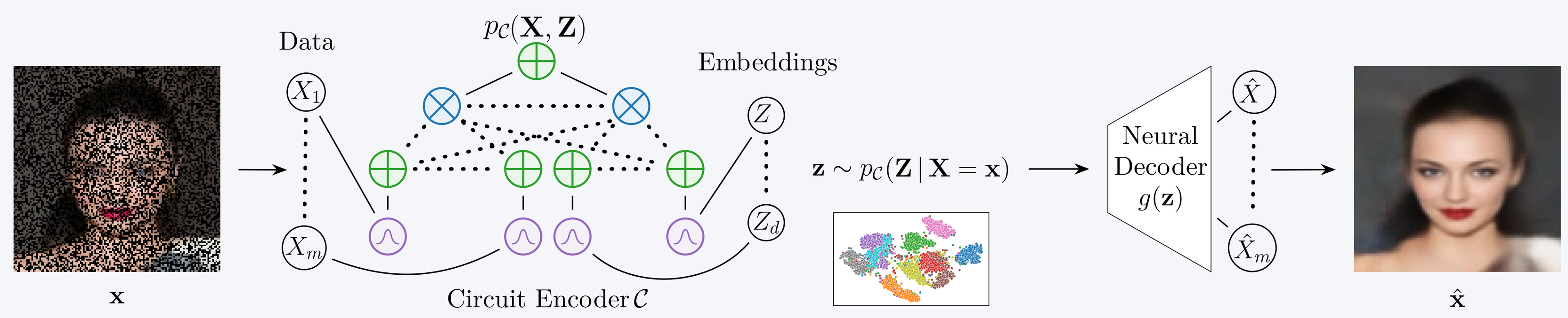
- **Embedding Regularization**: Forces the learned embedding distribution to match a simple prior q (e.g., Gaussian) using the KL divergence

$$\mathcal{L}_{\text{KLD}} = \sum_{i=1}^{B} \text{KLD}(p_{\mathcal{C}'}(\mathbf{Z} \mid \mathbf{x}_i) \parallel q(\mathbf{Z}))$$

- **Likelihood Regularization**: Regularizes the encoder to conform to the joint probability of data and embeddings

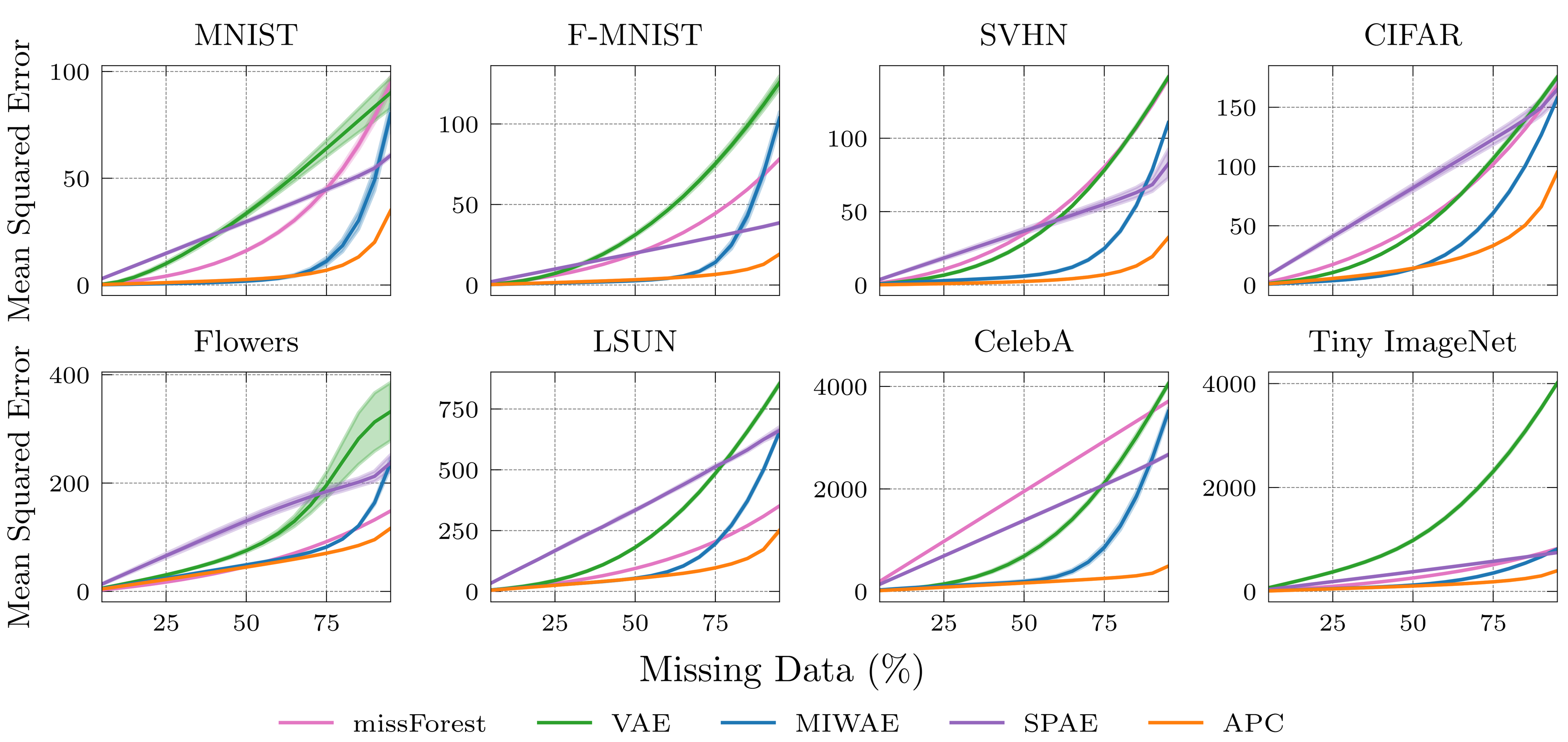
$$\mathcal{L}_{\text{NLL}} = -\frac{1}{B}\sum_{i=1}^{B} \log p_{\mathcal{C}}(\mathbf{x}_i, \mathbf{z}_i)$$

Bridging Tractable Enc. and Neural Dec.



APCs combine tractable probabilistic encoder and flexible high capacity neural decoder

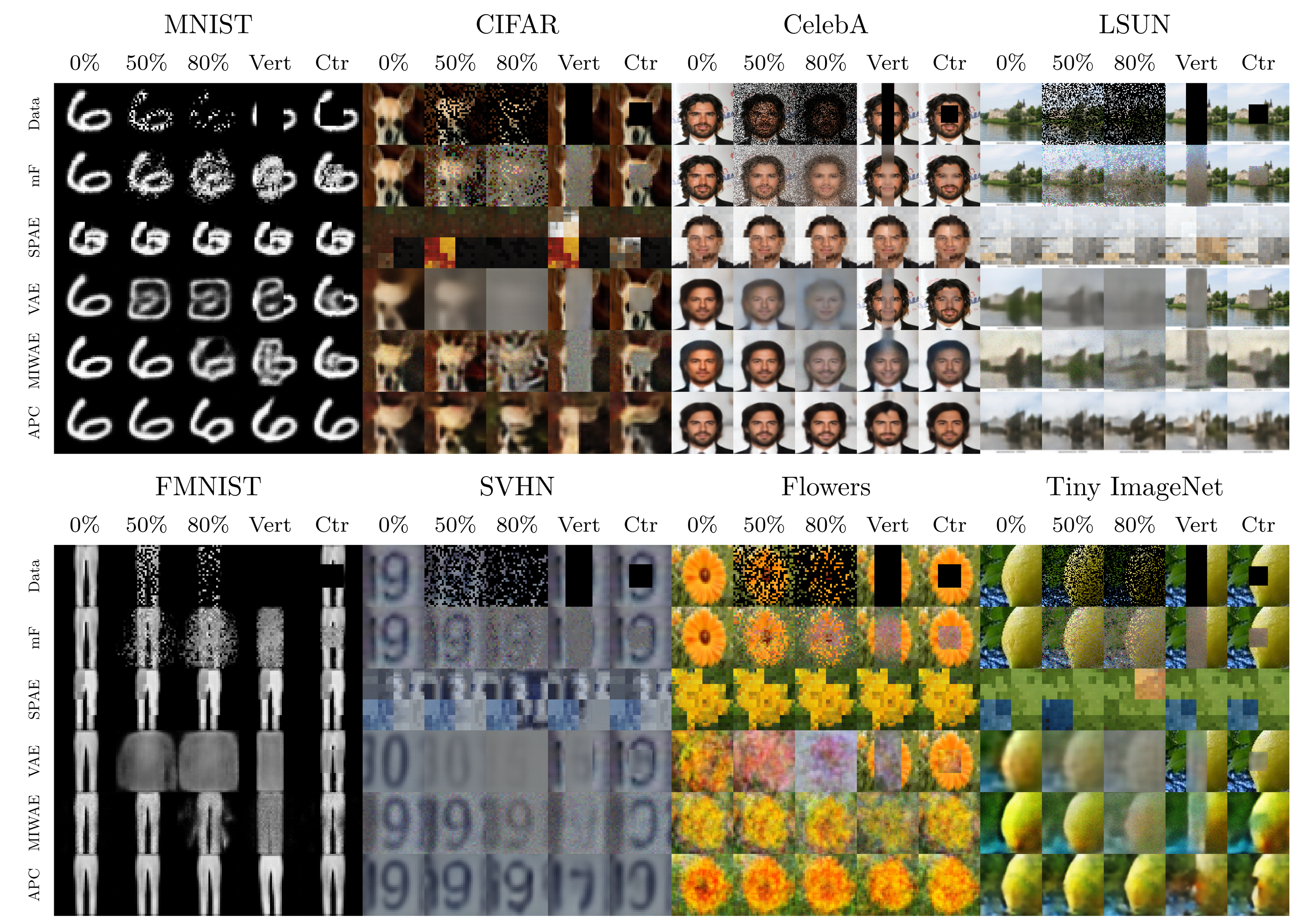
Reconstruction Performance



APCs maintain lowest reconstruction error even under high missing data rates, while neural autoencoders quickly degrade.

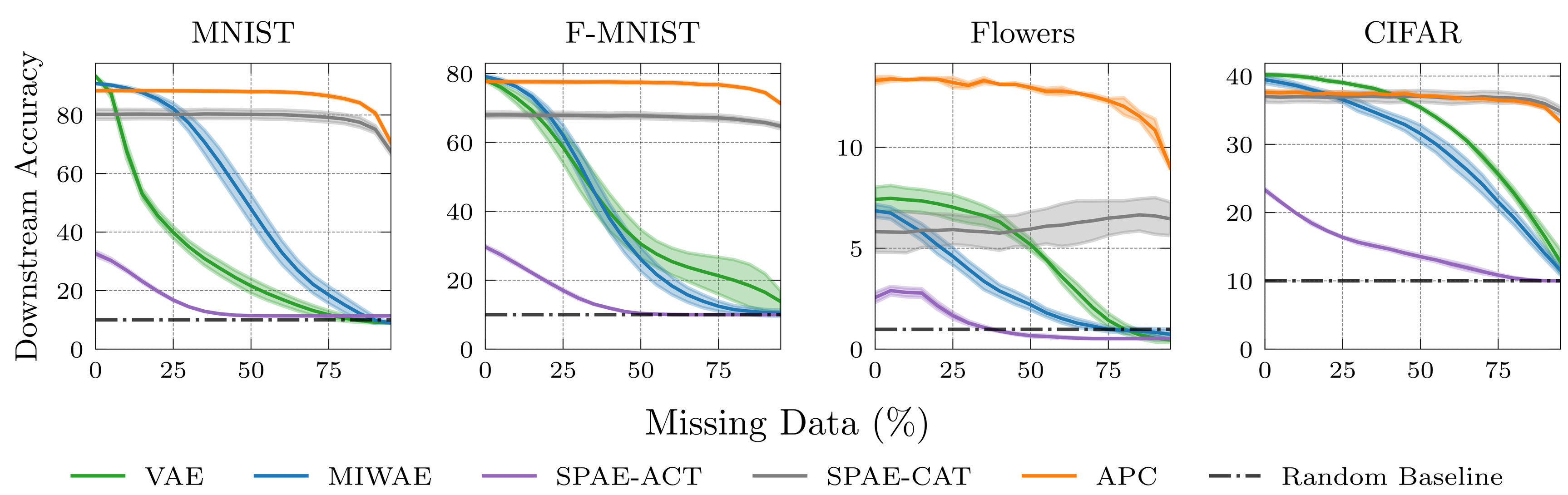
Contact: steven.braun@cs.tu-darmstadt.de

Reconstr. Missing (Completly) At Random Data



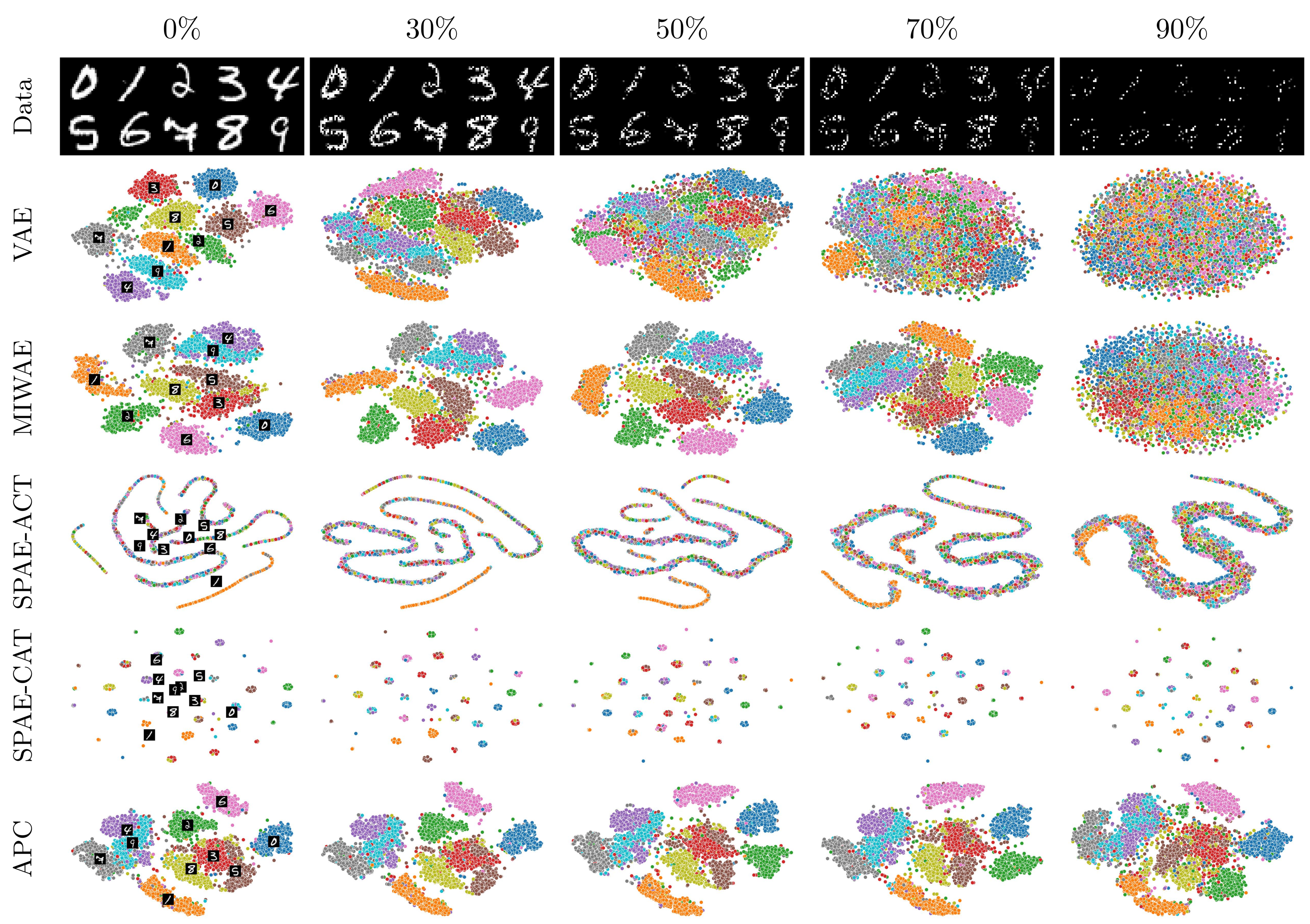
↪ **APCs can tractably marginalize over missing inputs**

Downstream Accuracy



APCs retain downstream accuracy (logistic regression) on MCAR data, while neural encoder lose separability and performance as missing data ratio increases.

Latent Structure w. MCAR Data



APCs maintain stable embeddings across missing data corruption, whereas neural encoders degrade rapidly.