Appendix

A.1 Network Architecture

Feature extraction network: Since we introduced the random hidden state $\mathbf{h}_t$ for the recurrent neural network, we use neural networks $\varphi_\tau^{\mathbf{x}}$ and $\varphi_\tau^{\mathbf{z}}$ for feature extraction from $\mathbf{x}_t$ and $\mathbf{z}_t$, respectively.

- $\varphi_\tau^{\mathbf{x}}(\mathbf{x}_t) = \mathbf{W}_1\mathbf{x}_t + b_1$
- $\varphi_\tau^{\mathbf{z}}(\mathbf{z}_t) = \mathbf{W}_3\text{relu}(\mathbf{W}_2\mathbf{z} + b_2) + b_3$

After feature extraction from $\mathbf{x}_t$ and $\mathbf{z}_t$, then, we stack $\mathbf{x}_t$ and $\mathbf{z}_t$ with $\mathbf{h}_{t-1}$ together for the inference and generative model respectively.

Artificial data model structure:

- Encoder:
  - $\mu(\mathbf{x}_t + \mathbf{h}_{t-1}) = \mathbf{W}_1(\mathbf{x}_t + \mathbf{h}_{t-1}) + b_1$.
  - $\sigma(\mathbf{x}_t + \mathbf{h}_{t-1}) = \exp(\mathbf{W}_2(\mathbf{x}_t + \mathbf{h}_{t-1}) + b_2)$.
- Deconder:
  - $\mu(\mathbf{z}_t + \mathbf{h}_{t-1}) = \mathbf{W}_{3t}(\mathbf{z}_t + \mathbf{h}_{t-1}) + b_3$.
  - $\sigma(\mathbf{z}_t + \mathbf{h}_{t-1}) = \exp(b_4)$.

Motion capture data model structure:

- Encoder:
  - $\mu(\mathbf{x}_t + \mathbf{h}_{t-1}) = \mathbf{W}_2\text{relu}(\mathbf{W}_1(\mathbf{x}_t + \mathbf{h}_{t-1})) + b_1$.
  - $\sigma(\mathbf{x}_t + \mathbf{h}_{t-1}) = \exp(\mathbf{W}_3\text{relu}(\mathbf{W}_1(\mathbf{x}_t + \mathbf{h}_{t-1})) + b_2)$.
- Deconder:
  - $\mu(\mathbf{z}_t + \mathbf{h}_{t-1}) = \mathbf{W}_{3t}\tanh(\mathbf{z}_t + \mathbf{h}_{t-1}) + b_3$.
  - $\sigma(\mathbf{z}_t + \mathbf{h}_{t-1}) = \exp(b_4)$.

Metabolomic data model structure:

- Encoder:
  - $\mu(\mathbf{x}_t + \mathbf{h}_{t-1}) = \mathbf{W}_2\text{relu}(\mathbf{W}_1(\mathbf{x}_t + \mathbf{h}_{t-1})) + b_1$.
  - $\sigma(\mathbf{x}_t + \mathbf{h}_{t-1}) = \exp(\mathbf{W}_3\text{relu}(\mathbf{W}_1(\mathbf{x}_t + \mathbf{h}_{t-1})) + b_2)$.
- Deconder:
  - $\mu(\mathbf{z}_t + \mathbf{h}_{t-1}) = \mathbf{W}_{3t}\tanh(\mathbf{z}_t + \mathbf{h}_{t-1}) + b_3$.
  - $\sigma(\mathbf{z}_t + \mathbf{h}_{t-1}) = \exp(b_4)$.

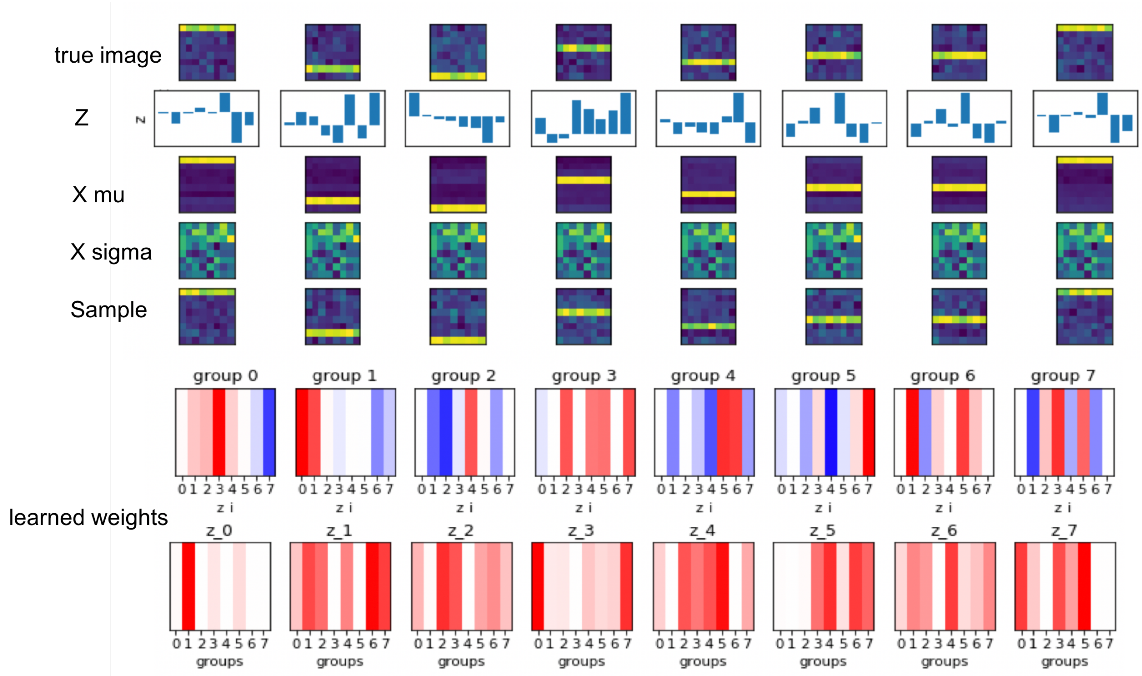

Figure 8: **Additional results of ITM-VAE on artificial data**. $k = 8$, $\lambda = 5$, iteration $= 10000$, batch-size $= 64$, $t = 8$. Here, we didn't assign the t label to each image since we want to check the model sparsity induced by λ. **true image**: the training image, **Z**: the sampled z values from encoder, **X mu**: the decoder mean, **X sigma**: the decoder sigma, **sample**: the reconstructed image from decoder mean and sigma, **learned weights**: the learned weights from the model.

A.2 Experimental Details

We ran Adam for the inference and generative net parameters optimization with learning rate 1e-3. Proximal gradient descent was run on $\mathbf{W}_t$ with learning rate 1e-4. **Artificial data**: We chose $\lambda = 5$. For the first part of experiment, we want to select λ so we randomly selected 64 images at each iteration and replicate each image 20 times as one batch, we ran for 10,000 iterations. The data structure is $20 \times 64 \times 64$. For the second part of experiment, we assign the row position of bar as the time label, so in total we have $t = 8$ different types of images, the data structure for each batch is $8 \times 64 \times 64$. **Motion capture data**: We chose $\lambda = 5$, and we used $T = 32$ frames and replicate each frame 32 times to stack as one batch ($32 \times 32 \times 59$) to train our model, optimization was run for 100 epochs. **Metabolomic data**: We chose $\lambda = 10$, we randomly selected $n = 2$ as one batch, the data structure for each batch is $12 \times 2 \times 980$, we ran 10,000 epochs.

A.3 Supplementary Figures

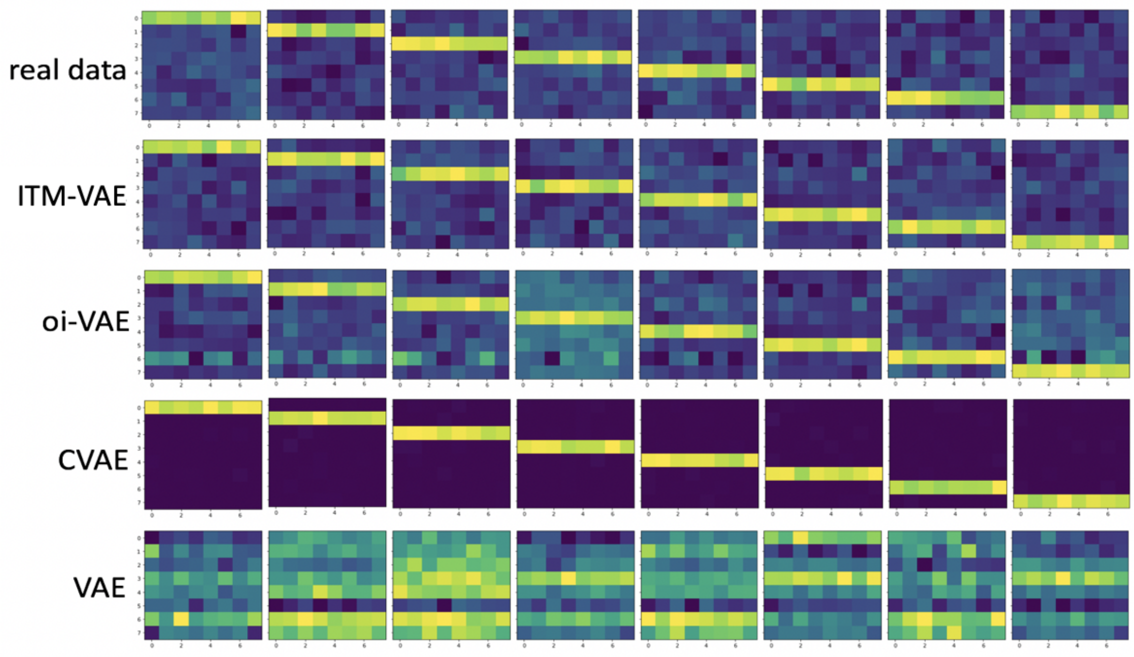

Figure 9: **Additional results of ITM-VAE on artificial data**. Reconstructed images.

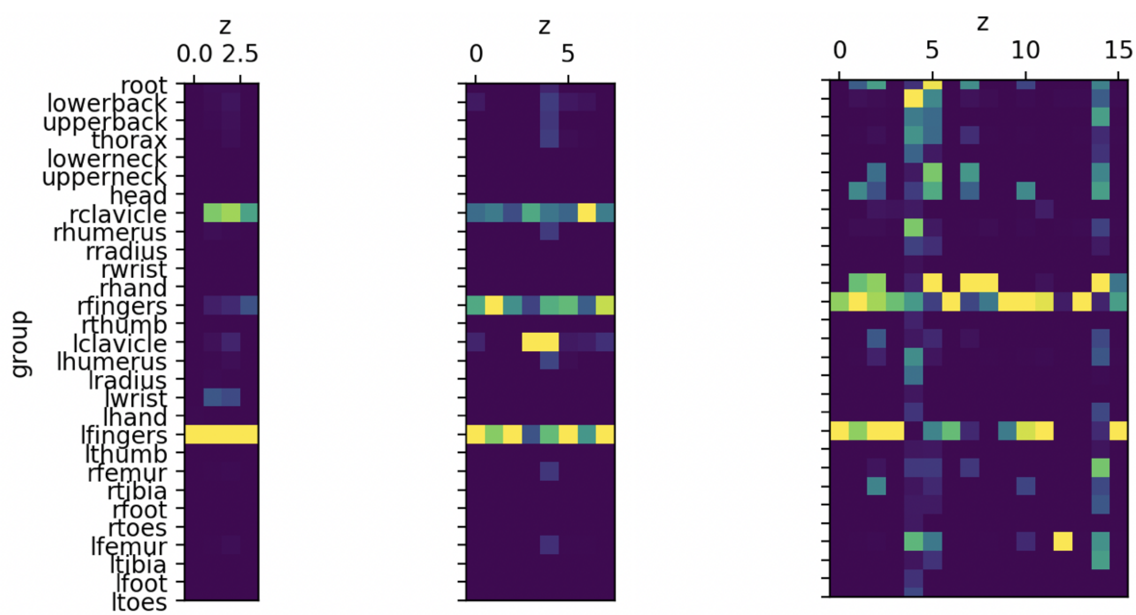

Figure 10: **Additional results from motion capture data.** The learned $\mathbf{W}_{t:,j}^{(g)}$ at time point $t = 10$ for $k = 4$ (left), $k = 8$ (middle), and $k = 16$ (right) with $\lambda = 5$.