VAEM: a Deep Generative Model for Heterogeneous Mixed Type Data

mixed type data

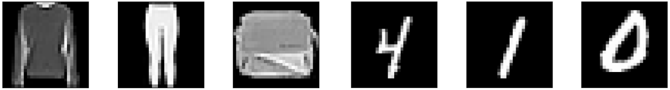
Chao Ma, Sebastian Tschiatschek, Richard E. Turner, José Miguel Hernández-Lobato, Cheng Zhang

Motivation

VAEs are typically applied in datasets where each data dimension has

- similar **statistical type** (e.g. continuous, binary, categorical, etc.),
- and similar **marginal distributions**.

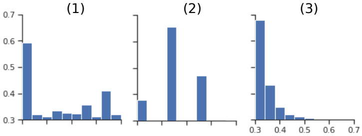
For example, image datasets or MNIST and Fashion MNIST datasets:



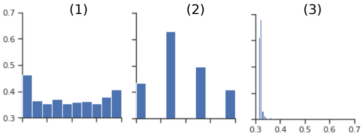
However, many real-world datasets contain variables with

- **different statistical types**
- and **different marginal properties** (bi-modal, heavy-tails, skewed, etc.).

In these cases, VAE models can result in **a poor fit to the data** since the different **likelihood factors** in the decoder will have very **different contributions**:



Observed data, ground truth



Samples from VAE

How can we reduce these problems in VAEs?

Proposed approach: VAEM

New VAE model for heterogeneous Mixed-type data (**VAEM**) trained in **two steps**.

First step:

We train D **marginal VAEs**, one for **each data dimension**. We optimize the ELBOs

$$\mathcal{L}(\boldsymbol{\theta}_d, \phi_d) = \sum_{n=1}^{N} \mathbf{E}_{q_{\phi_d}(z_{n,d}|x_{n,d})}\left[\log \frac{p_{\theta_d}(x_{n,d}|z_{n,d})p(z_{n,d})}{q_{\phi_d}(z_{n,d}|x_{n,d})}\right], \quad d = 1, \ldots, D\,.$$

Each marginal VAE fits **1D data** using a **type-specific likelihood** $p_{\theta_d}(x_{n,d}|z_{n,d})$.

The marginal encoders $q_{\phi_d}(z_{n,d}|x_{n,d})$ map each $x_{n,d}$ into a continuous latent $z_{n,d}$.

All the $z_{n,d}$ are **homogeneously distributed** as $p(z_{n,d})$, that is, as **standard Gaussian**!

Second step:

We model $z_{n,d}$ with an additional VAE called the **dependency network**. We optimize

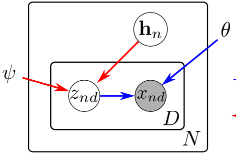
$$\mathcal{L}(\psi, \boldsymbol{\lambda}) = \sum_{n=1}^{N} \mathbf{E}_{\mathbf{z}_n \sim \prod_{d=1}^{D} q_{\phi_d}(z_{n,d}|x_{n,d})}\left\{\mathbf{E}_{q_{\boldsymbol{\lambda}}(\mathbf{h}_n|\mathbf{z}_n,\mathbf{x}_n)}\left[\log \frac{p_{\psi}(\mathbf{z}_n|\mathbf{h}_n)p(\mathbf{h}_n)}{q_{\boldsymbol{\lambda}}(\mathbf{h}_n|\mathbf{z}_n,\mathbf{x}_n)}\right]\right\}.$$

Final VAEM model obtained by combining dependency network and marginal VAEs.

The two-stage training can be shown to optimize an **ELBO on the joint model**.

Final model and dealing with missing data

After the two-stage training process, the VAEM generative model is given by



$$p_{\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_D,\boldsymbol{\psi}}(\mathbf{x}) = \int \underbrace{\left[\prod_{d=1}^{D} p_{\boldsymbol{\theta}_d}(x_d|z_d)\right]}_{\text{Marginal decoders}} \underset{\text{Dependency network decoder}}{p_{\psi}(\mathbf{z}|\mathbf{h})}\ \underset{\text{Standard Gaussian prior}}{p(\mathbf{h})}\, d\mathbf{h}\, dz_1 \cdots dz_D\,.$$

Marginal decoder

Dependency network decoder

How to train with missing data?

- The **marginal VAEs** are trained on the data available for each dimension. **No changes needed!**

- The **dependency network** is trained by optimizing the **partial ELBO**

$$\mathcal{L}'(\psi,\boldsymbol{\lambda}) = \sum_{n=1}^{N} \mathbf{E}_{\mathbf{z}_{\mathcal{O}}^{(n)} \sim \prod_{d\in\mathcal{O}} q_{\phi_d}(z_{n,d}|x_{n,d})} \left\{ \mathbf{E}_{q_{\boldsymbol{\lambda}}(\mathbf{h}_n|\mathbf{z}_{\mathcal{O}}^{(n)},\mathbf{x}_{\mathcal{O}}^{(n)})} \log\left[\frac{p_{\psi}(z_{\mathcal{O}}^{(n)}|\mathbf{h}_n)p(\mathbf{h}_n)}{q_{\boldsymbol{\lambda}}(\mathbf{h}_n|\mathbf{z}_{\mathcal{O}}^{(n)},\mathbf{x}_{\mathcal{O}}^{(n)})}\right]\right\},$$

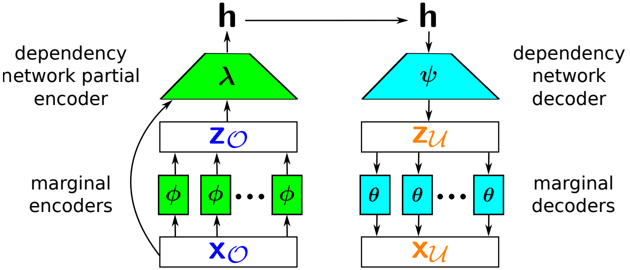
  where $q_{\boldsymbol{\lambda}}(\mathbf{h}_n|\mathbf{z}_{\mathcal{O}}^{(n)},\mathbf{x}_{\mathcal{O}}^{(n)})$ is a **PNP partial encoder** and $q_{\phi_d}(z_{n,d}|x_{n,d})$ are the **marginal encoders**.

Missing data imputation

How to impute missing data with VAEM?

We approximately sample from $p_{\text{VAEM}}(\mathbf{x}_{\mathcal{U}}|\mathbf{x}_{\mathcal{O}})$ in a bottom-up and top-down way:

1. Sample $\mathbf{z}_{\mathcal{O}}$ given $\mathbf{x}_{\mathcal{O}}$ using the marginal encoders.
2. Sample $\mathbf{h}$ given $\mathbf{z}_{\mathcal{O}}$ and $\mathbf{x}_{\mathcal{O}}$ using the dependency network encoder.
3. Sample $\mathbf{z}_{\mathcal{U}}$ given $\mathbf{h}$ using the dependency network decoder.
4. Sample $\mathbf{x}_{\mathcal{U}}$ given $\mathbf{z}_{\mathcal{U}}$ using the marginal decoders.



Assessment of data generation quality

Two evaluation settings:

- Fully observed data.
- A fraction of data missing at training time (0% to 99%) and at test time (50%).

Baselines (all with same partial encoder):

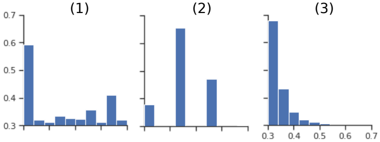
- Heterogeneous-Incomplete VAE (VAE-HI) [Nazabal et al. 2018].
- VAE and VAE with larger latent dimension.
- VAE with balanced likelihood.

Negative Log-likelihood, Fully Observed Data

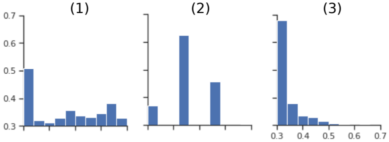
Method	VAEM (Ours)	VAE	VAE-balanced	VAE-extended	VAE-HI
Bank	**-1.15**±**.09**	2.09±.04	0.72±.01	2.06±.00	-0.72±.00
Boston	**-2.16**±**.01**	-1.69±.01	0.38±.01	-1.61±.02	2.11±.01
Avocado	**-0.16**±**.00**	0.04±.00	1.32±.01	0.04±.00	0.04±.00
Energy	-1.28±.09	**-1.47**±**.07**	0.69±.02	-1.46±.08	0.16±.00
MIMIC	**-1.01**±**.00**	0.08±.00	0.69±.00	0.08±.00	0.08±.00
Avg. Rank	**1.40**±**.36**	2.60±.61	4.40±.36	3.00±.40	3.00±.57

Negative Log-likelihood, Missing Data Setting

Method	VAEM (Ours)	VAE	VAE-balanced	VAE-extended	VAE-HI
Bank	**-1.21**±**.12**	2.09±.00	0.68±.00	2.09±.00	-0.83±.01
Boston	**-2.18**±**.03**	-1.66±.02	0.37±.00	-1.67±.01	1.58±.01
Avocado	**-0.15**±**.00**	0.04±.00	1.33±.00	0.04±.00	0.04±.00
Energy	-1.30±.05	**-1.50**±**.06**	0.67±.01	-1.50±.06	0.13±.00
MIMIC	**-1.10**±**.00**	0.08±.00	0.57±.00	0.08±.00	0.08±.00
Avg. Rank	**1.40**±**.36**	2.60±.61	4.40±.38	2.30±.44	3.00±.57



Observed data, ground truth

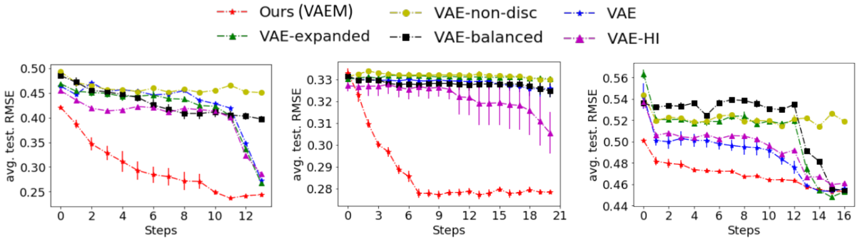


VAEM samples

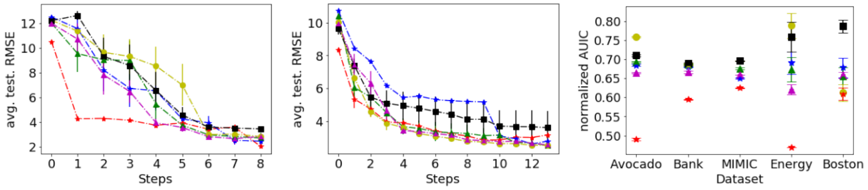
Results on sequential missing-value acquisition task

We include a **supervised predictor**.

Added an additional baseline, **VAE-no-disc**, where the supervised predictor is not used.



(a) Avocado (b) Bank (c) MIMIC



(d) Energy (e) Boston (f) AUIC Comparison