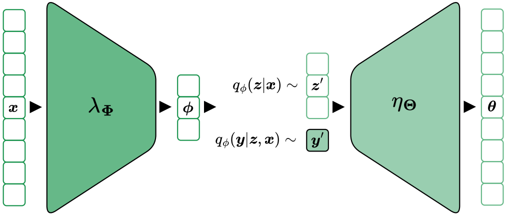
Learning Disentangled Representations with Semi-Supervised Deep Generative Models

David B. Hoffmann

December 4, 2025



Outline

1 Introduction and Formulation

2 Method

3 Experiments

4 Conclusion

5 Appendix

Motivation

Fully specified and supervised graphical models which are interpretable

Unsupervised variational auto-encoders which are uninterpretable

Motivation

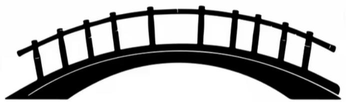
Fully specified and supervised graphical models which are interpretable

→ **GAP** ←

Unsupervised variational auto-encoders which are uninterpretable

Motivation

Fully specified and supervised graphical models which are interpretable



Semi-Supervised VAE [5]

Unsupervised variational auto-encoders which are uninterpretable

Introduction

What do Narayanaswamy et al. [5] introduce in their paper "Learning Disentangled Representations with Semi-Supervised Deep Generative Models"?

Introduction

What do Narayanaswamy et al. [5] introduce in their paper "Learning Disentangled Representations with Semi-Supervised Deep Generative Models"?

- Disentanglement refers to learning independent factors that explain the data as previously introduced by β-VAE [2] or β-TCVAE [1].

Introduction

What do Narayanaswamy et al. [5] introduce in their paper "Learning Disentangled Representations with Semi-Supervised Deep Generative Models"?

- Disentanglement refers to learning independent factors that explain the data as previously introduced by β-VAE [2] or β-TCVAE [1].
- Semi-supervised learning takes place where only part of the data is labelled. In "Semi-supervised Learning with Deep Generative Models" Kingma et al. [4] first use this concept to improve generative and classification capabilities of variational auto encoders (VAE).

Introduction

What do Narayanaswamy et al. [5] introduce in their paper "Learning Disentangled Representations with Semi-Supervised Deep Generative Models"?

- Disentanglement refers to learning independent factors that explain the data as previously introduced by β-VAE [2] or β-TCVAE [1].
- Semi-supervised learning takes place where only part of the data is labelled. In "Semi-supervised Learning with Deep Generative Models" Kingma et al. [4] first use this concept to improve generative and classification capabilities of variational auto encoders (VAE).
- Narayanaswamy et al. combine both concepts to bridge the gap between fully specified graphical models and fully unsupervised VAE representations.

Formulation

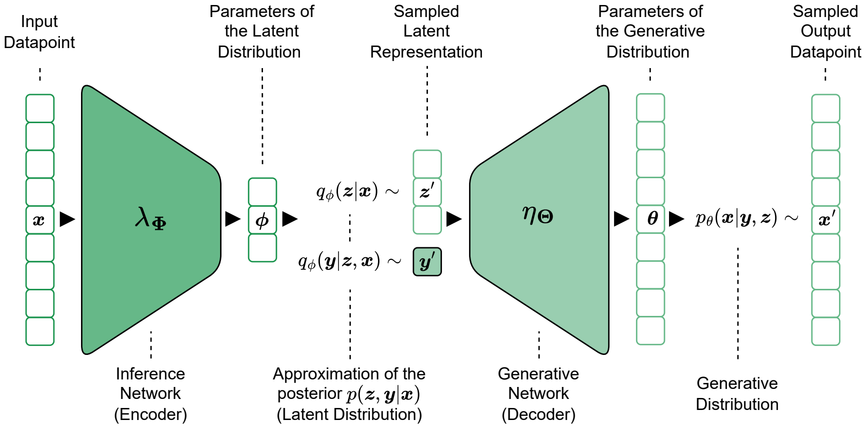


Figure: Formulation of the semi-supervised disentanglement framework

Outline

1. Introduction and Formulation
2. Method
3. Experiments
4. Conclusion
5. Appendix

Overview

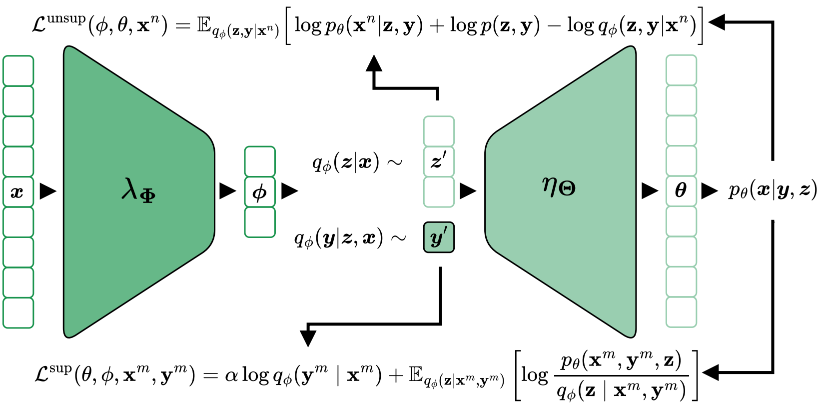


Figure: Semi-supervised disentanglement framework

Semi-Supervised Objective Formulation

The total objective combines unsupervised (x) and supervised (x, y) data, weighted by γ:

$$\mathcal{L}(\theta, \phi, \mathcal{D}, \mathcal{D}^{\text{sup}}) = \sum_{x^n \in \mathcal{D}}^{N} \mathcal{L}^{\text{unsup}}(\theta, \phi; x^n) + \gamma \sum_{(x^m, y^m) \in \mathcal{D}^{\text{sup}}}^{M} \mathcal{L}^{\text{sup}}(\theta, \phi; x^m, y^m)$$

Semi-Supervised Objective Formulation

The total objective combines unsupervised (x) and supervised (x, y) data, weighted by γ:

$$\mathcal{L}(\theta, \phi, \mathcal{D}, \mathcal{D}^{\text{sup}}) = \sum_{x^n \in \mathcal{D}}^{N} \mathcal{L}^{\text{unsup}}(\theta, \phi; x^n) + \gamma \sum_{(x^m, y^m) \in \mathcal{D}^{\text{sup}}}^{M} \mathcal{L}^{\text{sup}}(\theta, \phi; x^m, y^m)$$

The Supervised Term $\mathcal{L}^{\text{sup}}$: Defined to jointly maximize the generative likelihood and discriminative power:

$$\mathcal{L}^{\text{sup}}(\theta, \phi; x, y) = \underbrace{\alpha \log q_\phi(y|x)}_{\text{Discriminative}} + \underbrace{\mathbb{E}_{q_\phi(z|x,y)}\left[\log \frac{p_\theta(x, y, z)}{q_\phi(z|x, y)}\right]}_{\text{Generative (ELBO on joint } x, y\text{)}}$$

Semi-Supervised Objective Formulation

In the supervised term, we cannot evaluate $q_\phi(z|x, y)$ directly:

$$\mathcal{L}^{\text{sup}}(\theta, \phi; x, y) = \alpha \log q_\phi(y|x) + \mathbb{E}_{q_\phi(z|x,y)}\left[\log \frac{p_\theta(x, y, z)}{q_\phi(z|x, y)}\right]$$

Semi-Supervised Objective Formulation

In the supervised term, we cannot evaluate $q_\phi(z|x,y)$ directly:

$$\mathcal{L}^{\text{sup}}(\theta,\phi;x,y) = \alpha \log q_\phi(y|x) + \mathbb{E}_{q_\phi(z|x,y)}\left[\log \frac{p_\theta(x,y,z)}{q_\phi(z|x,y)}\right]$$

We use that $q_\phi(z|x,y)$ factorizes to $\frac{q_\phi(y,z|x)}{q_\phi(y|x)}$ and get:

$$\mathcal{L}^{\text{sup}}(\theta,\phi;x,y) = (1+\alpha) \log q_\phi(y|x) + \mathbb{E}_{q_\phi(z|x,y)}\left[\log \frac{p_\theta(x,y,z)}{q_\phi(y,z|x)}\right]$$

Semi-Supervised Objective Formulation

In the supervised term, we cannot evaluate $q_\phi(z|x, y)$ directly:

$$\mathcal{L}^{\text{sup}}(\theta, \phi; x, y) = \alpha \log q_\phi(y|x) + \mathbb{E}_{q_\phi(z|x,y)}\left[\log \frac{p_\theta(x, y, z)}{q_\phi(z|x, y)}\right]$$

We use that $q_\phi(z|x, y)$ factorizes to $\frac{q_\phi(y,z|x)}{q_\phi(y|x)}$ and get:

$$\mathcal{L}^{\text{sup}}(\theta, \phi; x, y) = (1 + \alpha) \log q_\phi(y|x) + \mathbb{E}_{q_\phi(z|x,y)}\left[\log \frac{p_\theta(x, y, z)}{q_\phi(y, z|x)}\right]$$

Now we approximate the expectation and $\log q_\phi(y|x)$ with importance sampling and get:

$$\widehat{\mathcal{L}}^{\text{sup}} = \sum_{s=1}^{S} \frac{w_s}{\sum_j w_j} \log \frac{p_\theta(x, y, z_s)}{q_\phi(y, z_s \mid x)} + (1 + \alpha) \log w_s$$

Outline

1. Introduction and Formulation
2. Method
3. Experiments
4. Conclusion
5. Appendix

MNIST & SVHN: Classification

- Claim: Show that the generalization maintains comparable classification performance to the semi-supervised VAE setup in Kingma et al. [4].
- Setup: Partially specified digit label y and unsupervised style parameters $\boldsymbol{z}$ as shown below.

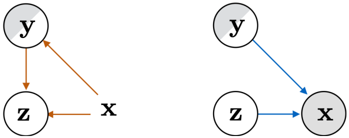


Figure: MNIST Graphical Model. (Left) The Recognition Model component. (Right) The Generator Model component.

	M	Ours	M2 [4]
MNIST $N = 50000$	100	9.71 ($\pm$ 0.91)	11.97 ($\pm$ 1.71)
	600	3.84 ($\pm$ 0.86)	4.94 ($\pm$ 0.13)
	1000	2.88 ($\pm$ 0.79)	3.60 ($\pm$ 0.56)
	3000	1.57 ($\pm$ 0.93)	3.92 ($\pm$ 0.63)

	M	Ours	M1+M2 [4]
SVHN $N = 70000$	1000	38.91 ($\pm$ 1.06)	36.02 ($\pm$ 0.10)
	3000	29.07 ($\pm$ 0.83)	—

Table: Classification error rates for different labelled-set sizes M over multiple runs, with supervision rate $\rho = \frac{\gamma M}{N+\gamma M}, \gamma = 1$. Table and Caption from Fig. 3 of the paper [5].

MNIST & SVHN: Regression

- Claim: Demonstrate that latent variables are isolated and can be used to guide generation. Only shown qualitatively.
- Setup: Same as for the classification scenario.

(a) (b) (c) (d)

Figure: (a) Visual analogies for the MNIST data, partially supervised with just 100 labels (out of 50000). They infer the style variable z and then vary the label y. (b) Exploration in style space with label y held fixed and (2D) style z varied. Visual analogies for the SVHN data when (c) partially supervised with just 1000 labels, and (d) fully supervised. Table and Caption from Fig. 2 of the paper [5].

Outline

1. Introduction and Formulation
2. Method
3. Experiments
4. Conclusion
5. Appendix

Conclusion

- General Framework: Extends semi-supervised VAEs [4] to support arbitrary dependency structures in both generative and recognition models.

Conclusion

- General Framework: Extends semi-supervised VAEs [4] to support arbitrary dependency structures in both generative and recognition models.
- Derives a generic Importance Sampling estimator that handles computationally intractable marginals, removing architectural restrictions.

Conclusion

- General Framework: Extends semi-supervised VAEs [4] to support arbitrary dependency structures in both generative and recognition models.
- Derives a generic Importance Sampling estimator that handles computationally intractable marginals, removing architectural restrictions.
- Structured Disentanglement: Achieves disentanglement by combining partially-specified graphical models (for interpretable factors) with flexible neural networks (for unstructured noise).

References I

[1] Ricky T. Q. Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating Sources of Disentanglement in Variational Autoencoders. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[2] I. Higgins, L. Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, M. Botvinick, S. Mohamed, and Alexander Lerchner. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. November 2016.

[3] Varun Jampani, S. M. Ali Eslami, Daniel Tarlow, Pushmeet Kohli, and John M. Winn. Consensus message passing for layered graphical models. *CoRR*, abs/1410.7452, 2014.

[4] Diederik P. Kingma, Danilo J. Rezende, Shakir Mohamed, and Max Welling. Semi-supervised Learning with Deep Generative Models. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[5] Siddharth Narayanaswamy, Brooks Paige, Jan-Willem van de Meent, Alban Desmaison, Noah Goodman, Pushmeet Kohli, Frank Wood, and Philip Torr. Learning Disentangled Representations with Semi-Supervised Deep Generative Models. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Outline

1 Introduction and Formulation

2 Method

3 Experiments

4 Conclusion

5 Appendix

Overview

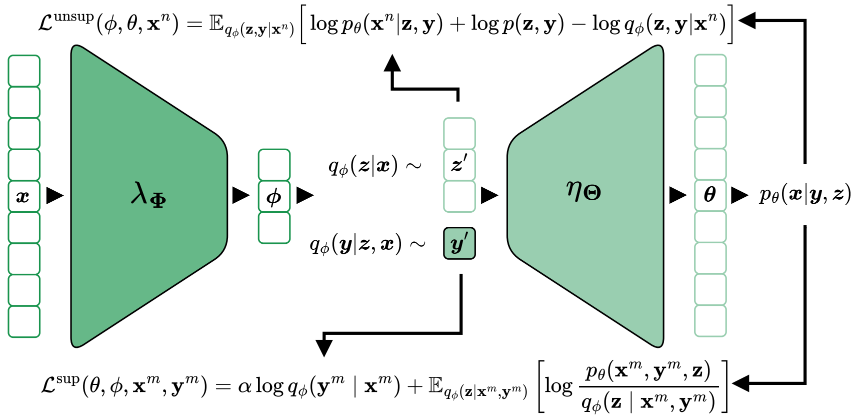


Figure: Semi-supervised disentanglement framework

Approximation with Importance Sampling

Approximate the expectation with importance sampling

$$\mathbb{E}_{q_\phi(z|x,y)}\left[\log \frac{p_\theta(x,y,z)}{q_\phi(y,z|x)}\right] \simeq \frac{1}{S}\sum_{s=1}^{S} \frac{w^s}{Z} \log \frac{p_\theta(x,y,z^s)}{q_\phi(y^m,z^s|x)}$$

Here We sample $z^s \sim q_\phi(z|x)$ from the unconditioned encoder with importance weights:

$$w^s := \frac{q_\phi(y,z^s|x)}{q_\phi(z^s|x)}, \quad Z = \frac{1}{S}\sum_{s=1}^{S} w^s$$

Using the same weights we approximate $\log q_\phi(y^m|x^m)$ with a Monte Carlo estimate of the lower bound:

$$\log q_\phi(y|x) \geq \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{q_\phi(y,z|x)}{q_\phi(z|x)}\right] \simeq \frac{1}{S}\sum_{s=1}^{S} \log w^s$$

MNIST & SVHN: Supervision Rate

- Goal: Exploration of the supervision rate which controls the balance of supervised and unsupervised objectives in the loss.
- Setup: Same graphical model as before. Here, scaling of the classification objective is held fixed at $\alpha = 50$ (MNIST) and $\alpha = 70$ (SVHN).
- Result: For sparsely labelled data ($M \ll N$), some over-representation ($\gamma > 1$) helps improve generalization with better performance on the test set. Too much over-representation leads to overfitting.

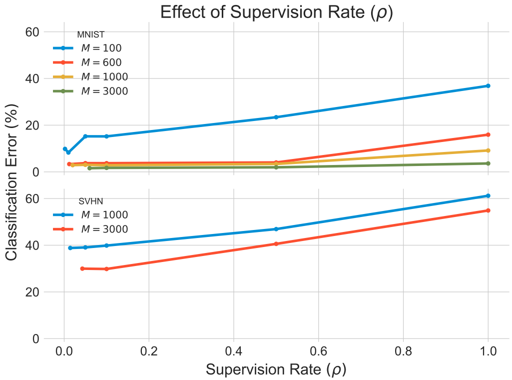


Figure: Classification error over different labelled set sizes and supervision rates for MNIST (top) and SVHN (bottom). Table and Caption from Fig. 3 of the paper [5].

Yale B Faces

- Goal: Claim that they show that their model learns the correct relationship between lighting, shading and reflectance. Instead they show that their semi-supervised model performs worse than a fully supervised counterpart.
- Setup: Partially supervised identity label *i* and lighting angle *l* with unsupervised latent variables for shading *s* and reflection *r*.

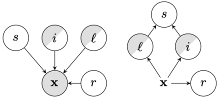
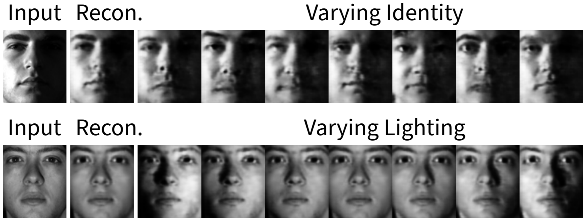


Figure: Graphical Model from Fig. 5 of [5]. (Generator and Recognition Model)



	Identity	Lighting
Ours (Full Supervision)	1.9% ($\pm$ 1.5)	3.1% ($\pm$ 3.8)
Ours (Semi-Supervised)	3.5% ($\pm$ 3.4)	17.6% ($\pm$ 1.8)
Jampani et al. [3] (plot asymptotes)	≈ 30	≈ 10

Figure: (Left:) Exploring the generative capacity of the supervised model by manipulating identity and lighting given a fixed (inferred) value of the other latent variables. (Right:) Classification and regression error rates for identity and lighting latent variables, fully-supervised, and semi-supervised. Table and Caption from Fig. 4 of the paper [5]

Multi MNIST

- Goal: Explore capacity for stochastic dimensionality.
- Setup: Use a stochastic sequence generator for each of the digits in the image in composition with the pretrained MNIST VAE from before.

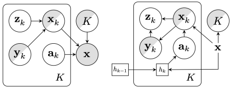
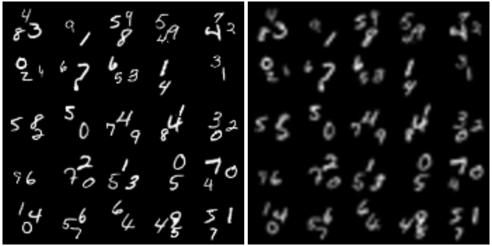
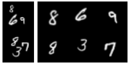


Figure: Graphical Model from Fig. 5 of [4]. (Generator and Recognition Model)

Input | Reconstruction



Decomposition



$\frac{M}{M+N}$	Count Error (%)	
	w/o MNIST	w/ MNIST
0.1	85.45 ($\pm$ 5.77)	76.33 ($\pm$ 8.91)
0.5	93.27 ($\pm$ 2.15)	80.27 ($\pm$ 5.45)
1.0	99.81 ($\pm$ 1.81)	84.79 ($\pm$ 5.11)

Figure: (Left): Example input multi-MNIST images and reconstructions. (Top-Right): Decomposition of Multi-MNIST images into constituent MNIST digits. (Bottom-Right): Count accuracy. Table and Caption from Fig. 6 of the paper [5].