

c\*GMΔVÆs—q

Kolb, Yiftach

February 16, 2023



c\*GMΔVÆs—q

a Master Thesis in Bioinformatics

Advisor / Reviewer: Professor Martin Vingron  
Reviewer: Professor Tim Conrad

# Glossary

c\*GMΔVÆ (maybe Conditional) Gaussian Mixture with Dirichlet prior Variational Auto Encoder

c\*GMΔVÆs—q c\*GMΔVÆ (for) s(ingle-cell-RNA-Se)q.

# Autoencoders

A "vanilla" autoencoder is a neural networks that "learns" the identity (subject to dimensional restriction).

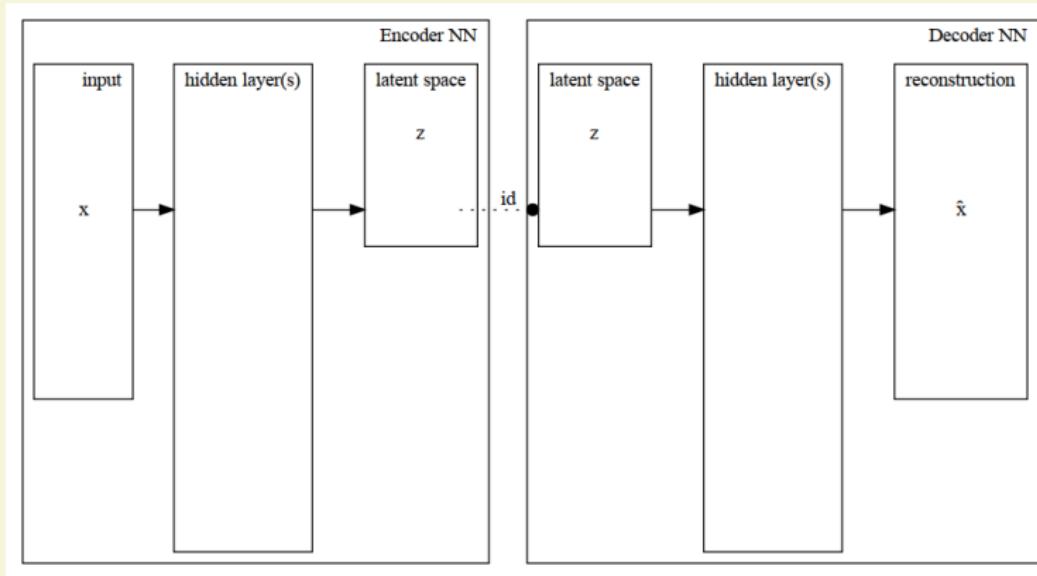


Figure: Autoencoder

# Autoencoders and PCA

(On centered data[8])

PCA

$$\tilde{\mathbf{V}} = \operatorname{argmin}_{\mathbf{W}} \{ \| \mathbf{X} - \mathbf{X} \mathbf{W} \mathbf{W}^T \|_F^2 : \mathbf{W} \in \mathbb{R}^{n \times l}, \mathbf{W}^T \mathbf{W} = \mathbf{I}_l \} \quad (1)$$

Linear AE

$$\operatorname{argmin}_{\mathbf{E}, \mathbf{D}} \{ \| \mathbf{X} - \mathbf{X} \mathbf{E} \mathbf{D} \|_F^2 : \mathbf{E}, \mathbf{D}^T \in \mathbb{R}^{n \times l}, \} \quad (2)$$

$$\tilde{\mathbf{W}} \in \operatorname{argmin}_{\mathbf{W}} \{ \| \mathbf{X} - \mathbf{X} \mathbf{W} \mathbf{W}^\dagger \|_F^2 : \mathbf{W} \in \mathbb{R}^{n \times l}, \} \quad (3)$$

$$\operatorname{span}\{\tilde{\mathbf{W}}\} = \operatorname{span}\{\tilde{\mathbf{V}}\}$$

# VAEs

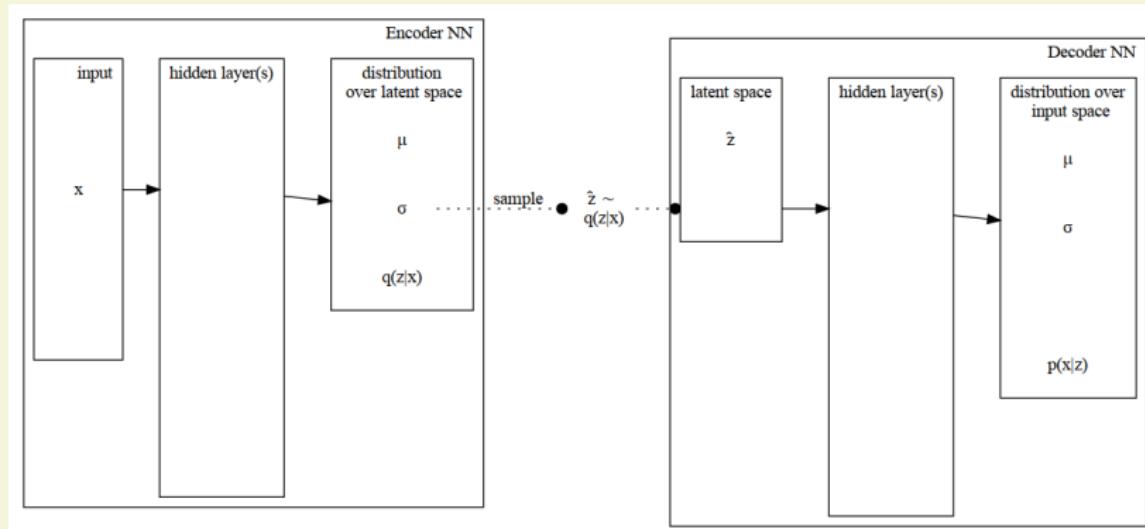


Figure: VAE

## VAE: encoding

Instead of deterministic mapping ...define distribution over the latent space ( $\mathbf{z}$ ) by mapping  $\mathbf{x}$  into the distribution parameters e.g.  $\mu(\mathbf{x}), \Sigma(\mathbf{x})$  when we use Gaussian  $q(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}|\mu, \Sigma)$ .

## VAE: decoding

sample from the latent space  $\mathbf{z} \sim \mathcal{N}(\cdot|\mu, \Sigma)$

map  $\mathbf{z}$  to a distribution on the input space  $p(\mathbf{x}|\mathbf{z})$

## VAE: loss function

The *evidence lower bound (ELBO)* with respect to  $p, q$  is:

$$-\mathcal{L}(q, p, \mathbf{x}) \triangleq \int \log \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z})} dq(\mathbf{z}) \quad (4)$$

$$-\mathcal{L}(q, p) \triangleq -\mathcal{L}(q, p, \mathbf{X}) = \frac{1}{N} \sum_1^N (-\mathcal{L}(q, p, \mathbf{x}_i)) \quad (5)$$

$$\approx \mathbf{E}_{\mathbf{x}}[-\mathcal{L}(q, p, \mathbf{x})] \quad (6)$$

(we want to minimize the minus ELBO function)

# VAE: log evidence

It can be shown that maximizing the ELBO is equivalent to maximinizing the "log evidence"  $\log p(\mathbf{X})$

$$\begin{aligned}\frac{1}{N} \log p(\mathbf{X}) &= \frac{1}{N} \log \int p(\mathbf{X}, \mathbf{Z}) d\mathbf{Z} && \text{taking marginal} \\ &= \frac{1}{N} \log \int \frac{p(\mathbf{X}, \mathbf{Z})}{q(\mathbf{Z})} q(\mathbf{Z}) d\mathbf{Z} && \text{multiplying by 1 inside} \\ &= \frac{1}{N} \log \int \frac{p(\mathbf{X}, \mathbf{Z})}{q(\mathbf{Z})} dq(\mathbf{Z}) && \text{definition of } dq(\mathbf{Z}) \\ &\geq \frac{1}{N} \int \log \frac{p(\mathbf{X}, \mathbf{Z})}{q(\mathbf{Z})} dq(\mathbf{Z}) && \text{Jensen inequality} \tag{7} \\ &= \frac{1}{N} \int \sum_1^N \log \frac{p(\mathbf{x}_i, \mathbf{z}_i)}{q(\mathbf{z}_i)} dq(\mathbf{z}_i) && \text{using the iid property} \\ &= \frac{1}{N} \sum_1^N -\mathcal{L}(q, p, \mathbf{x}_i) && \text{definition of } \mathcal{L}(q, p, \mathbf{x}_i) \\ &= -\mathcal{L}(q, p, \mathbf{X}) \triangleq -\mathcal{L}(q, p) && \text{again definition of } \mathcal{L}(p, q) \quad \square\end{aligned}$$

# Monte Carlo integration

As you can see we the loss function requires us to compute an integral, which usually cannot be done analytically.

Instead the integral is approximated by Monte Carlo integration.

sample  $\mathbf{z}_i \sim q(\mathbf{z}|\mathbf{x})$  then :

$$\begin{aligned}\mathcal{L}(p, q, \mathbf{x}) &= \int -\log \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x})} dq(\mathbf{z}|\mathbf{x}) \\ &= \int -\log p(\mathbf{x}|\mathbf{z}) dq(\mathbf{z}|\mathbf{x}) + \int \log \frac{q(\mathbf{z}|\mathbf{x})}{p(\mathbf{z})} dq(\mathbf{z}|\mathbf{x}) \quad (8) \\ &\approx \frac{1}{k} \sum_{i=1}^k \left[ -\log p(\mathbf{x}|\mathbf{z}_i) + \log \frac{q(\mathbf{z}_i|\mathbf{x})}{p(\mathbf{z}_i)} \right]\end{aligned}$$

## VAE: compounding the latent distribution

More complicated distributions such as mixture distribution can be modelled by "unpacking" the latent  $\mathbf{z}$  and the observed  $\mathbf{x}$

1. Define the set of observed random vectors  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k$ , and the set of latent random vectors and stochastic parameters  $\mathbf{z}_1, \dots, \mathbf{z}_l$ .
2. Specify how to factor the generative model  
 $p(\mathbf{x}_1, \dots, \mathbf{x}_k | \mathbf{z}_1, \dots, \mathbf{z}_l)$
3. Specify how to factor the inference model  
 $q(\mathbf{z}_1 \dots \mathbf{z}_l | \mathbf{x}_1, \dots, \mathbf{x}_k)$
4. Choose appropriate priors  $p(\mathbf{z}_i)$  and
5. Choose appropriate distribution families for the  $\mathbf{x}_i$  and  $\mathbf{z}_i$ , and choose priors  $p(\mathbf{z}_i)$ .

## VAE: Graphical representation

Every distribution can be represented by a DAG. Nodes represent random variables (and also priors), and directed arrows represent conditional dependency.

# VAE: base case

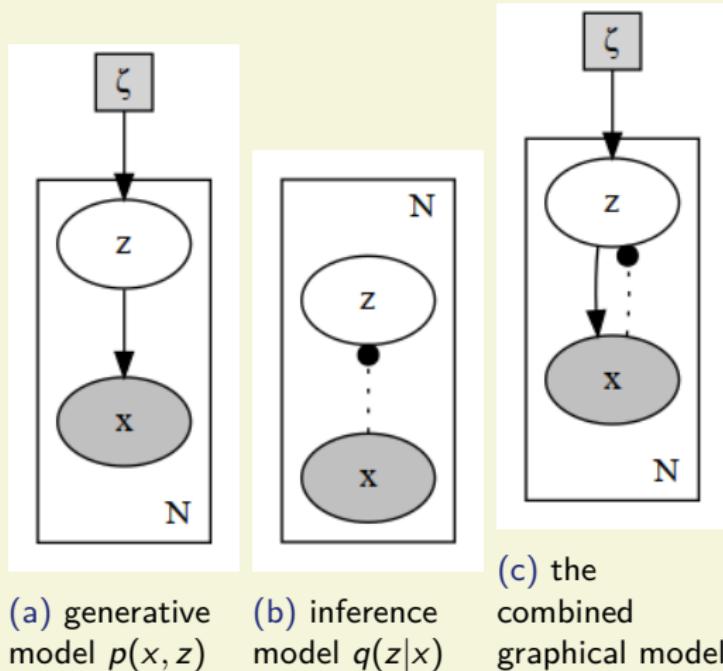
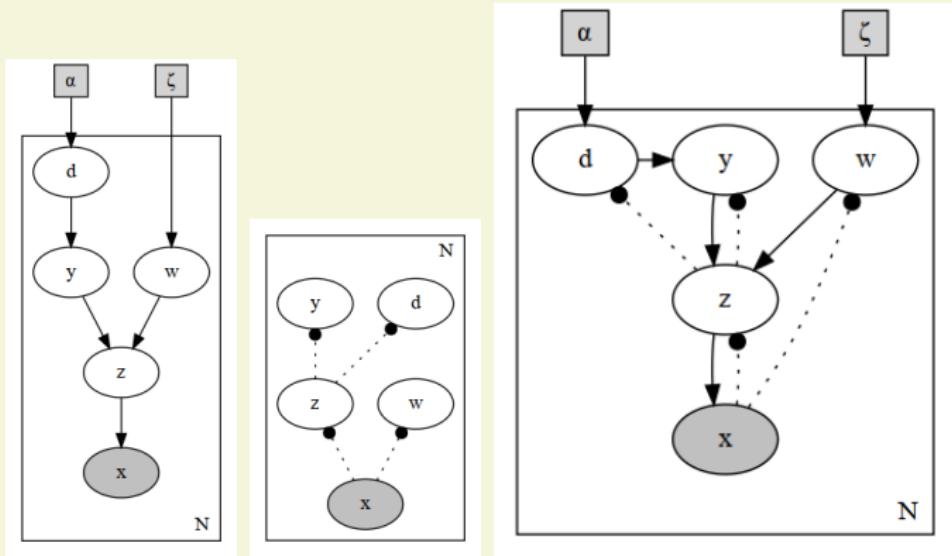


Figure: VAE graphical model

# VAE: pathological case



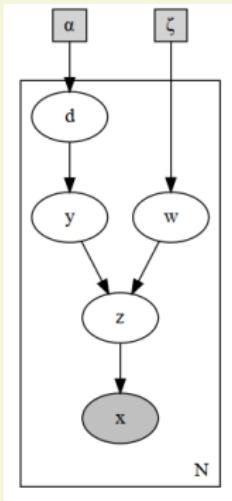
(a) generative  
model  $p(x, z)$

(b) inference  
model  $q(z|x)$

(c) the combined graphical  
model

Figure: c\*GMΔVAE graphical model

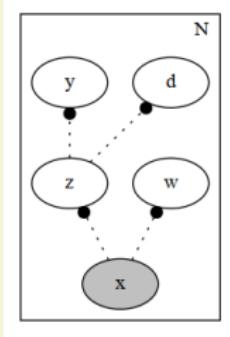
# c\*GMΔVÆ generative model



$$\begin{aligned} p(x, y, z, w, d) &= p(x|z)p(z|w, y)p(y|d)p(d)p(w) \\ p(w) &= \mathcal{N}(w|\mathbf{0}, \mathbf{1}) \\ p(d) &= \text{Dir}(d|\alpha) \\ p(y|d) &= \text{Cat}(y|d) \\ p(z|w, y) &= \mathcal{N}(z|\mu(w)_y, \sigma(w)_y) \\ p(x|z) &= \mathcal{N}(x|\mu(z), \sigma(z)) \end{aligned}$$

(9)

# c\*GMΔVÆ inference model



$$\begin{aligned} q(y, z, w, d|x) &= q(z|x)q(w|x)q(y|z)q(d|z) \\ q(z|x) &= \mathcal{N}(z|\mu_z(x), \sigma_z(x)) \\ q(w|x) &= \mathcal{N}(w|\mu_w(x), \sigma_w(x)) \quad (10) \\ q(y|z) &= \text{Cat}(y|f(z)) \\ q(d|z) &= \text{Dir}(d|g(z)) \end{aligned}$$

## c\*GMΔVÆ loss function

The loss function remains the -ELBO and we can break it into different terms:

$$\mathcal{L}(p, q, \mathbf{x}) = \int -\log \frac{p(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{w}, \mathbf{d})}{q(\mathbf{z}, \mathbf{y}, \mathbf{w}, \mathbf{d}|\mathbf{x})} dq(\mathbf{z}, \mathbf{y}, \mathbf{w}, \mathbf{d}|\mathbf{x}) \quad (11)$$

$$= \int -\log \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z}|\mathbf{w}, \mathbf{y})p(\mathbf{y}|\mathbf{d})p(\mathbf{w})p(\mathbf{d})}{q(\mathbf{z}|\mathbf{x})q(\mathbf{w}|\mathbf{x})q(\mathbf{y}|\mathbf{z})q(\mathbf{d}|\mathbf{z})} dq \quad (12)$$

$$= \int -\log p(\mathbf{x}|\mathbf{z})dq \quad (13)$$

$$+ \int \log \frac{q(\mathbf{z}|\mathbf{x})}{p(\mathbf{z}|\mathbf{w}, \mathbf{y})} dq \quad (14)$$

$$+ \int \log \frac{q(\mathbf{w}|\mathbf{x})}{p(\mathbf{w})} dq \quad (15)$$

$$+ \int \log \frac{q(\mathbf{y}|\mathbf{z})}{p(\mathbf{y}|\mathbf{d})} dq \quad (16)$$

$$+ \int \log \frac{q(\mathbf{d}|\mathbf{z})}{p(\mathbf{d})} dq \quad (17)$$

## c\*GMΔVÆ : supervised case

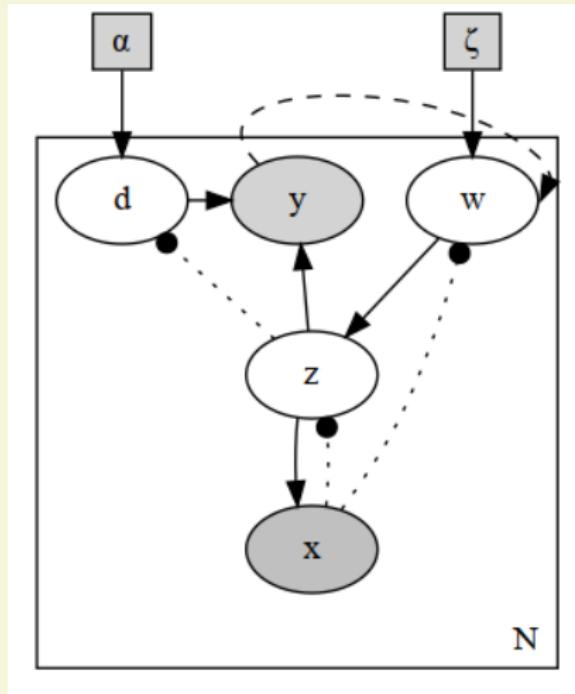
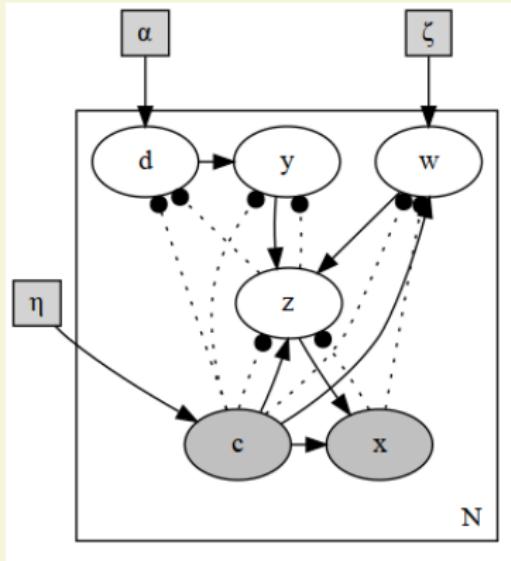
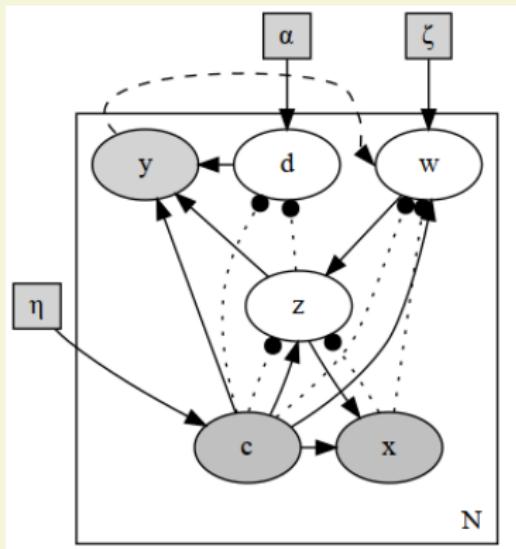


Figure: c\*GMΔVÆ : the supervised case, where  $y$  is an observed variable.

# the conditional flavor of $c^*GM\Delta VAE$

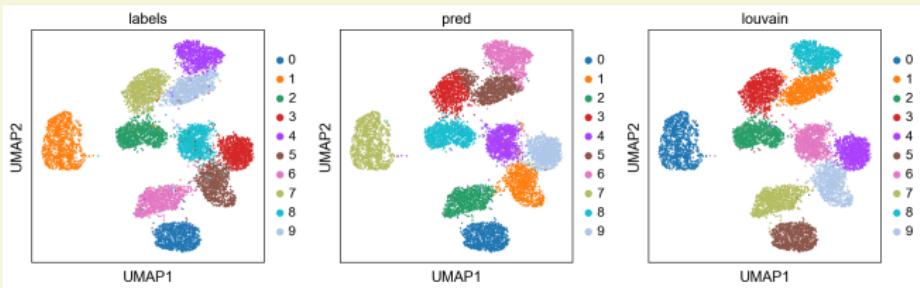


(a)  $c^*GM\Delta VAE$  (cond.),  
unsupervised case. Sorry for the  
arrow clutter



(b)  $c^*GM\Delta VAE$  (cond.), supervised  
case

# Tests on MNIST



(a) UMAP of the latent space

0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	7	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	7	9	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	9	4	1	2	3
0	5	6	7	8	7	4	1	2	3

# Testing on synthetic conditional-categorical "blobs" dataset

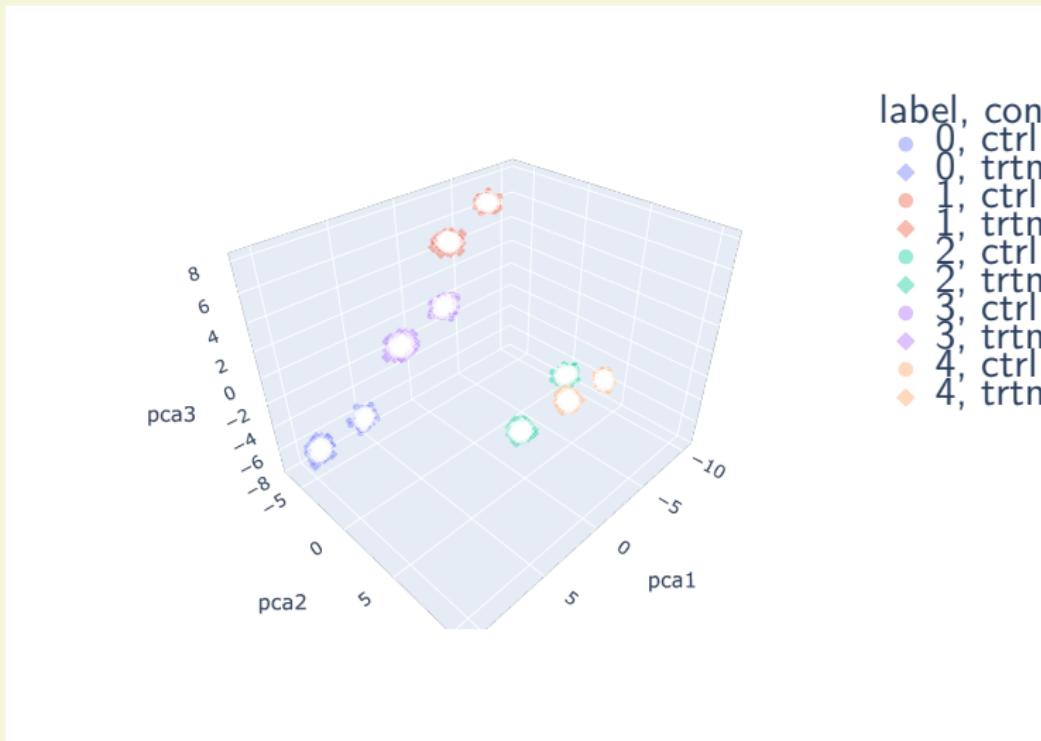


Figure: The toy dataset, showing a 3d plot of its 3 major PCA components.

## w embedding

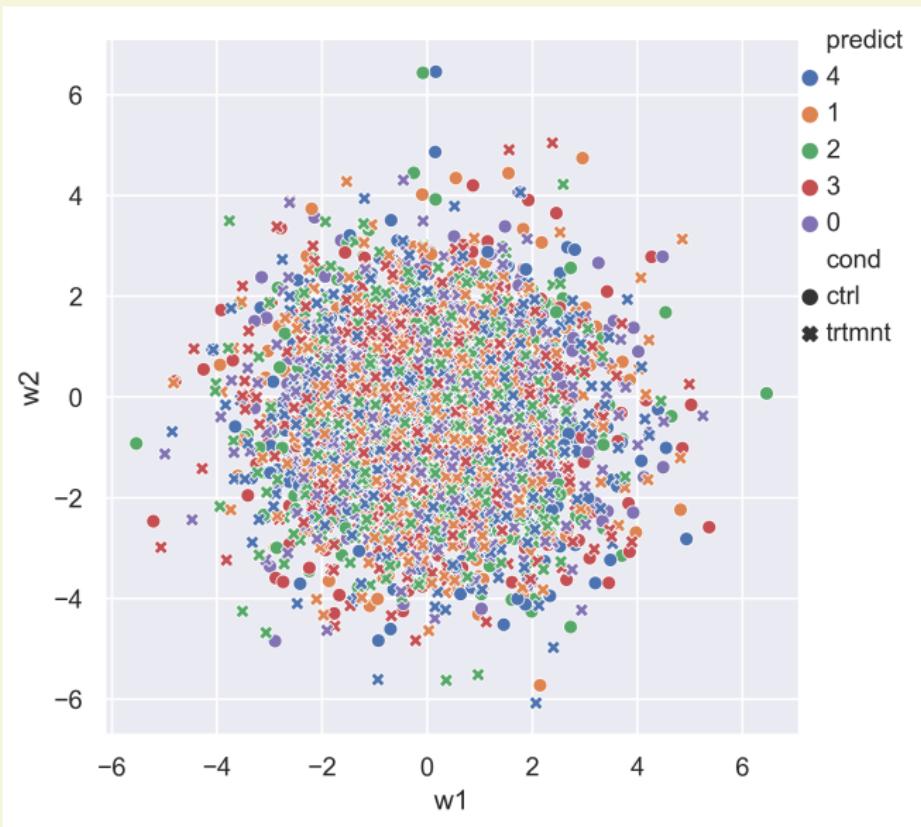


Figure: Random sampling from the encoded distribution of the  $w$  space.

## $z$ embedding

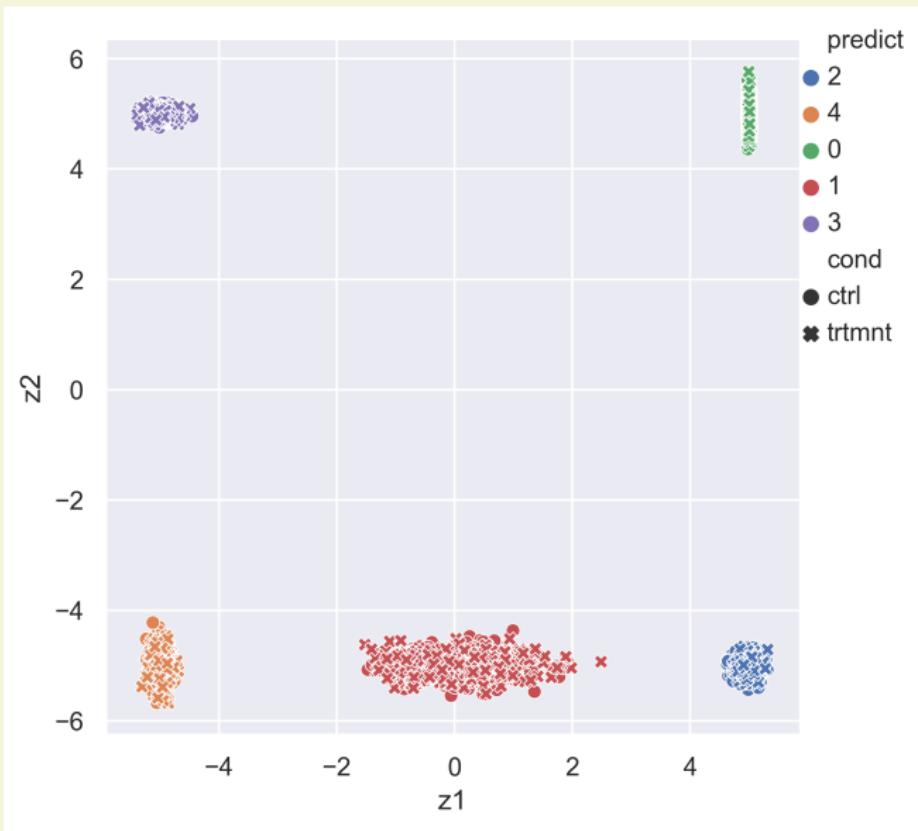


Figure: Sampling from encoded  $z$  space, fixed standard normal prior case.

# Transfer learning

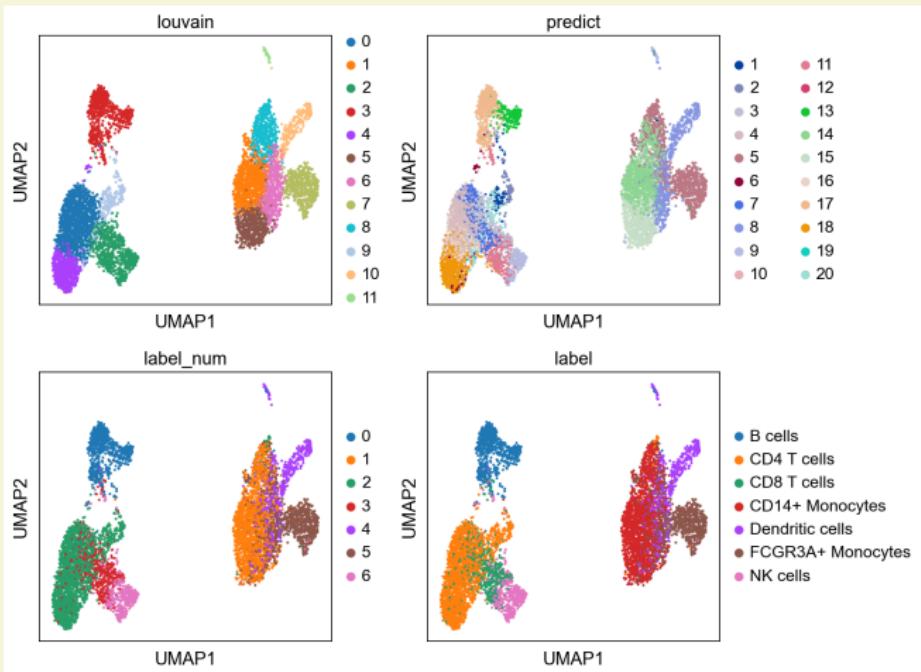


Figure: UMAP of PCA space of the Kang train data.

# Transfer learning

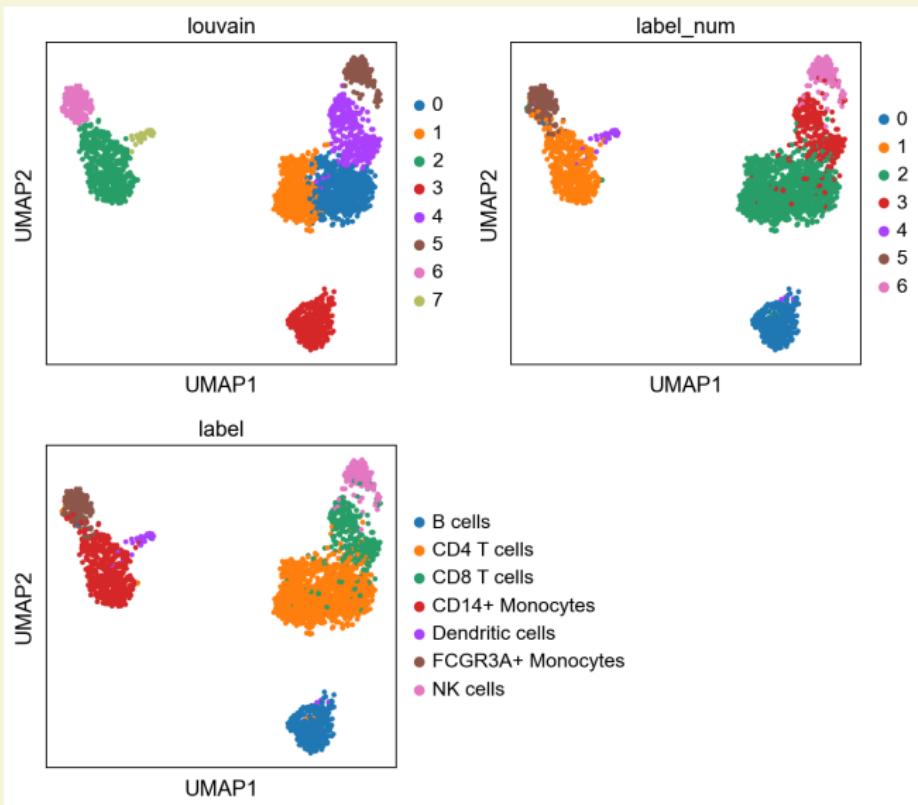


Figure: UMAP of the Zheng dataset.

# Transfer learning

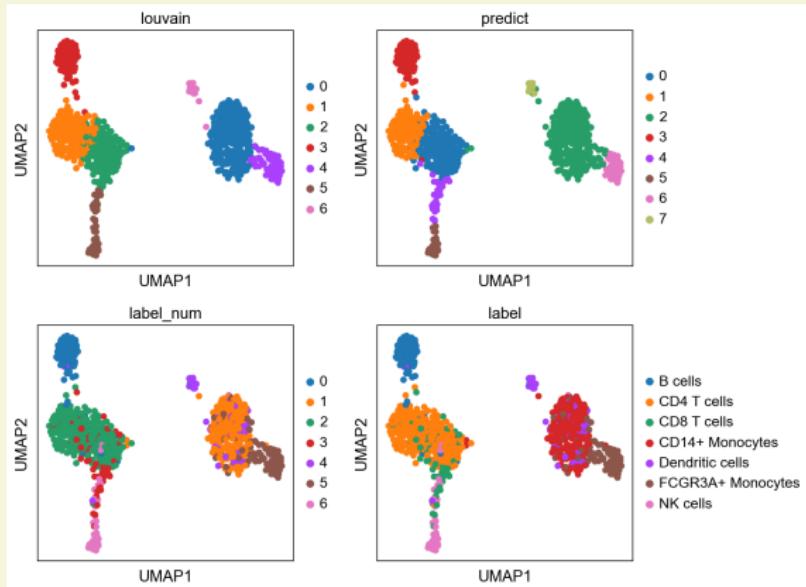


Figure: UMAP of the latent space of the Kang validation subset (the holdout).

# converting treatment effect

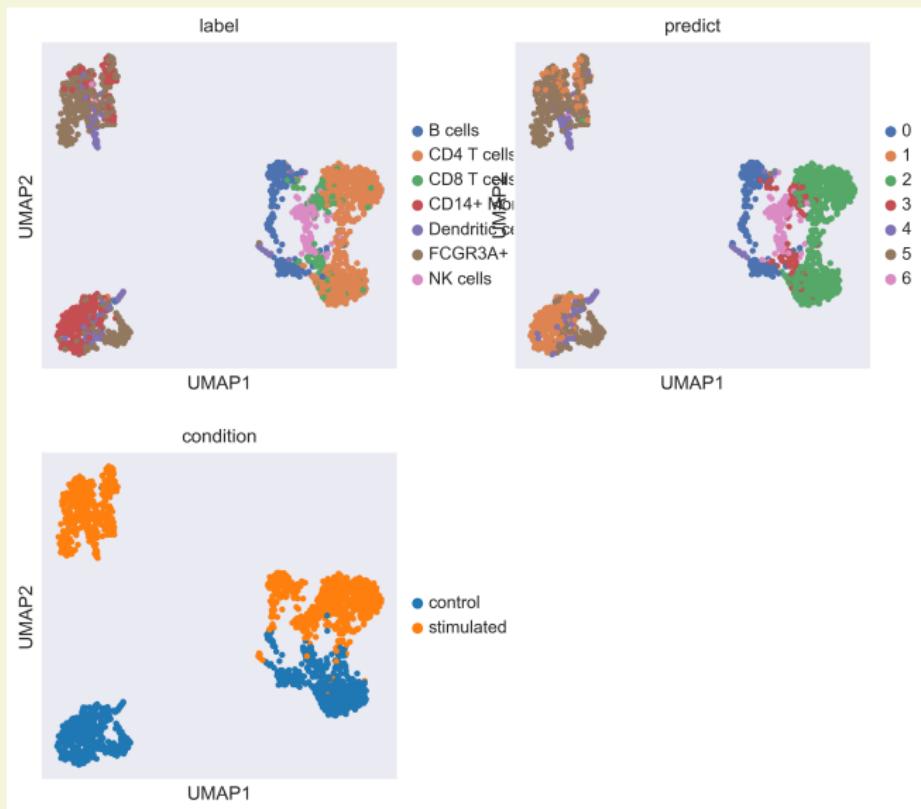


Figure: UMAP of the Kang validation dataset

# converting treatment effect

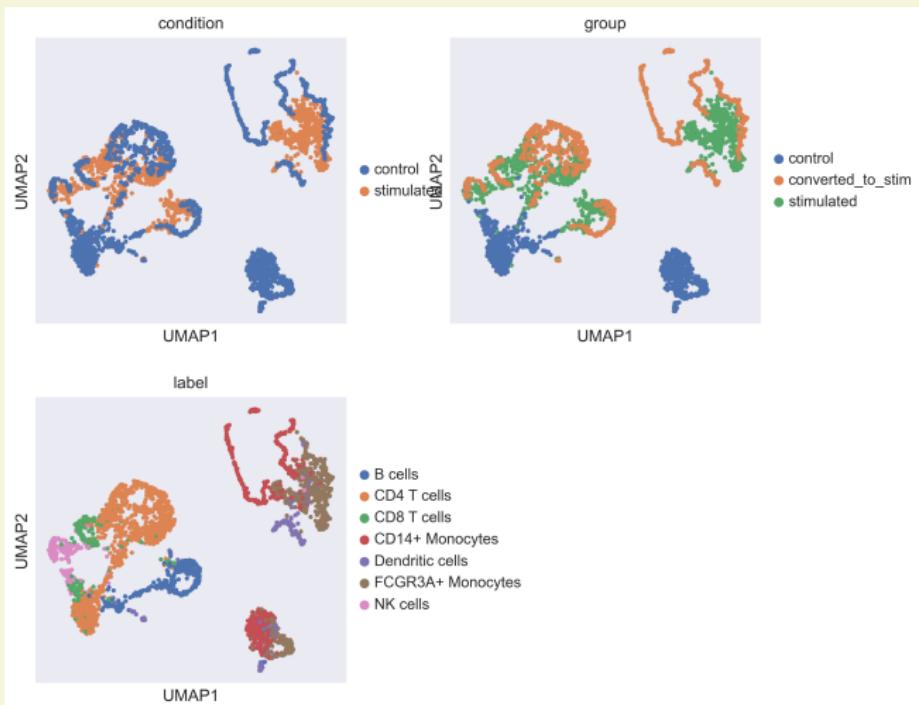
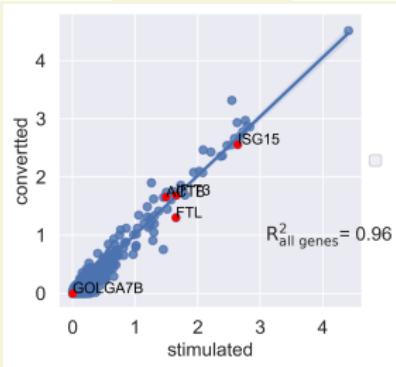
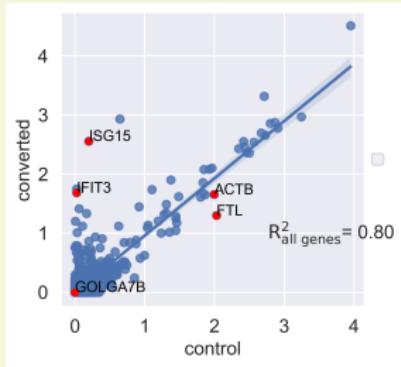
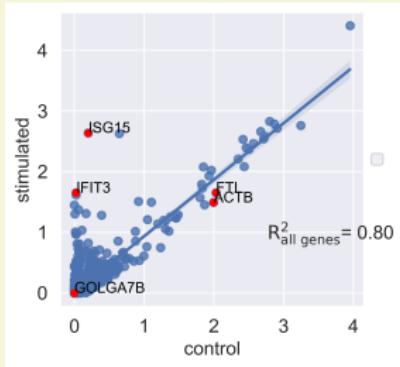


Figure: UMAP of the combined validation subset with its control group remapped into stimulated state.

# converting treatment effect



# Generating data

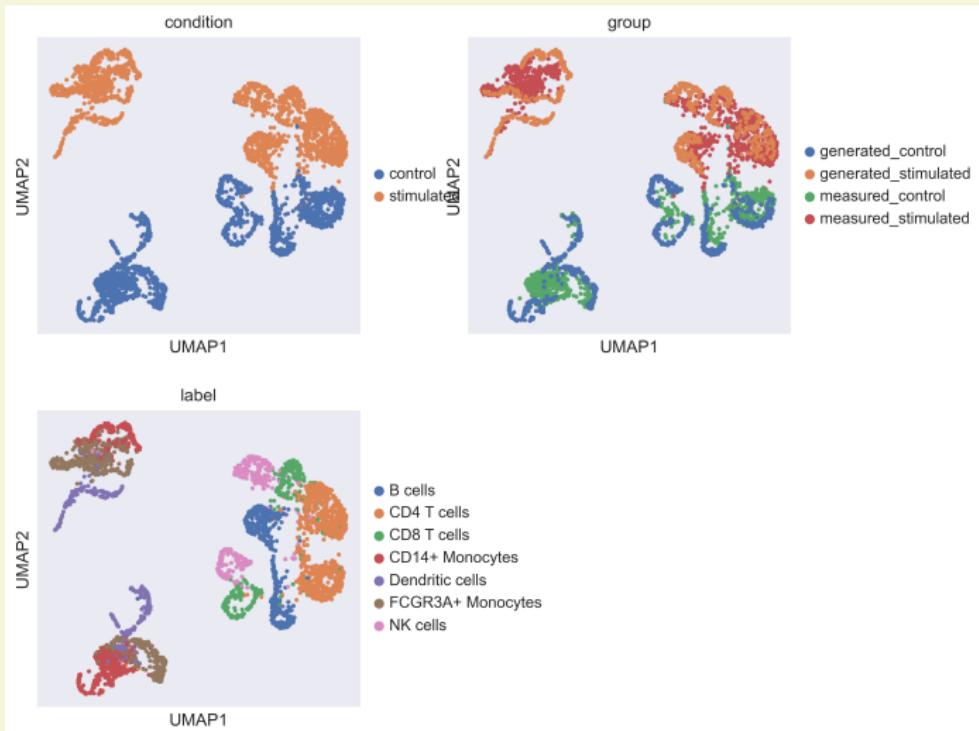


Figure: The Kang UMAP of real data combined with randomly generated data.

- [1] Christopher M Bishop. *Pattern recognition and machine learning*. Vol. 4. 4. Springer, 2006.
- [2] Nat Dilokthanakul et al. "Deep unsupervised clustering with gaussian mixture variational autoencoders". In: *arXiv preprint arXiv:1611.02648* (2016).
- [3] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).
- [4] Durk P Kingma et al. "Semi-supervised learning with deep generative models". In: *Advances in neural information processing systems* 27 (2014).
- [5] Yiftach Kolb. *The "official" c\*GMDVAE project git*. URL: <https://github.com/zelhar/mg22>.
- [6] Yiftach Kolb. *The Github project housing this thesis*. URL: <https://github.com/zelhar/mg22>.
- [7] Mohammad Lotfollahi, F Alexander Wolf, and Fabian J Theis. "scGen predicts single-cell perturbation responses". In: *Nature methods* 16.8 (2019), pp. 715–721.

- [8] Elad Plaut. “From principal subspaces to principal components with linear autoencoders”. In: *arXiv preprint arXiv:1804.10253* (2018).