Literature Review on VAEs with Correlated Latent Features

Javier Zapata

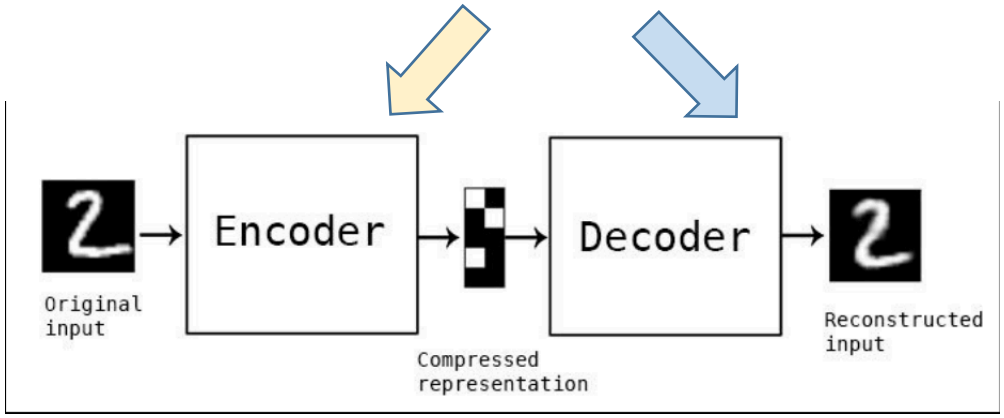
Today's plan

- VAE basics (mostly Kingma et al, 2014)
  - Autoencoders vs Variational Autoencoders
  - Variational Lower Bound
  - Reparemeterization Trick
- VAE with correlated latent features (Casale et al, 2018)

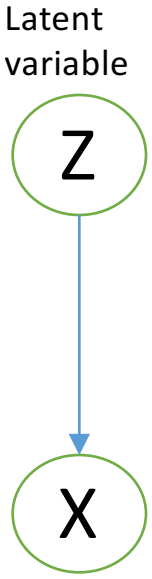
Auto-Encoder (deterministic)

- A neural network that the output is the input itself.
- Intuition:
  - A good representation should keep the information well (reconstruction error)
  - Deep + nonlinearity might help enhance the representation power

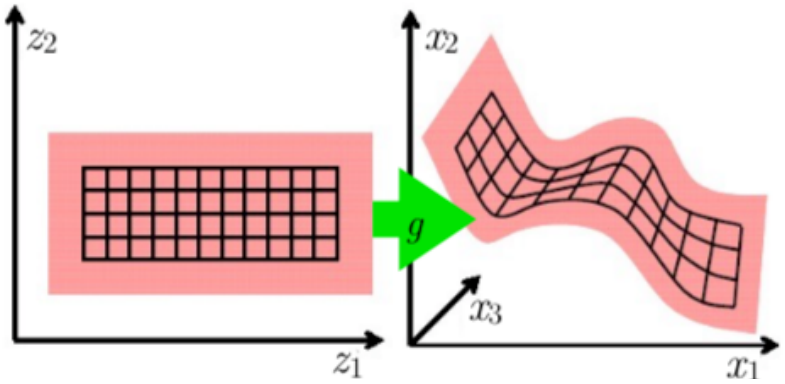
$$\min_{w_1,w_2} \|x_i - g(f(x_i; w_1); w_2\|_2$$



Variational Auto-Encoder (VAE) (probabilistic)

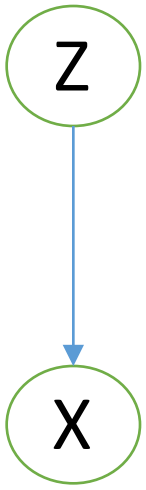


- Uses neural networks to learn a latent variable model
- A blend of variational inference & neural networks
- Seminal papers:
  - Kingma and Welling, *Auto-Encoding Variational Bayes*, *International Conference on Learning Representations (ICLR)* 2014.
  - Rezende, Mohamed and Wierstra, *Stochastic back-propagation and variational inference in deep latent Gaussian models*. ICML 2014.



Variational Auto-Encoder (VAE) (probabilistic)

Latent variable



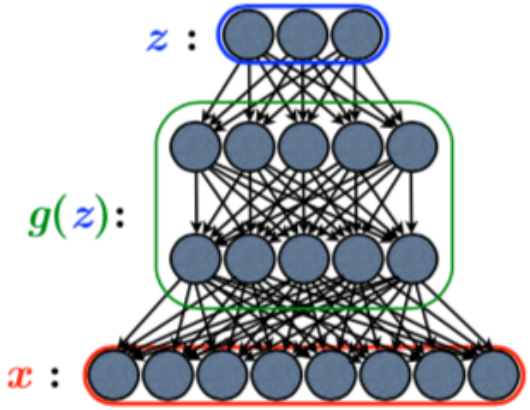
Observed Sample variable

- A blend of variational inference & neural networks
- **Latent Model**: Learn mapping from Z to X

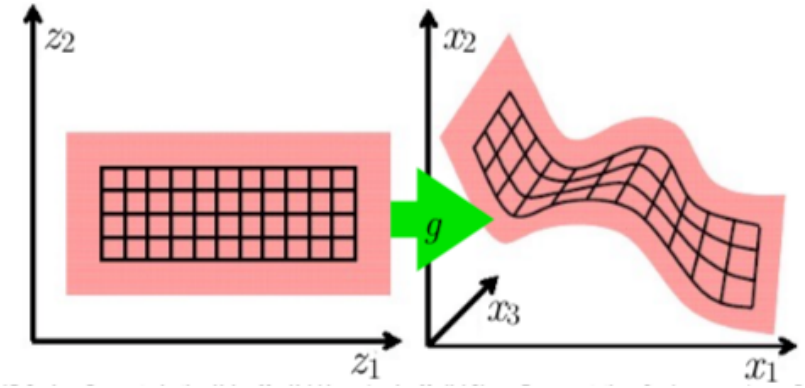
$$p(x) = \int p(x,z)dz = \int p(x|z)\,p(z)dz$$

where: $p(z)$: something simple

$p(x|z) = g(z)$ (a neural net)

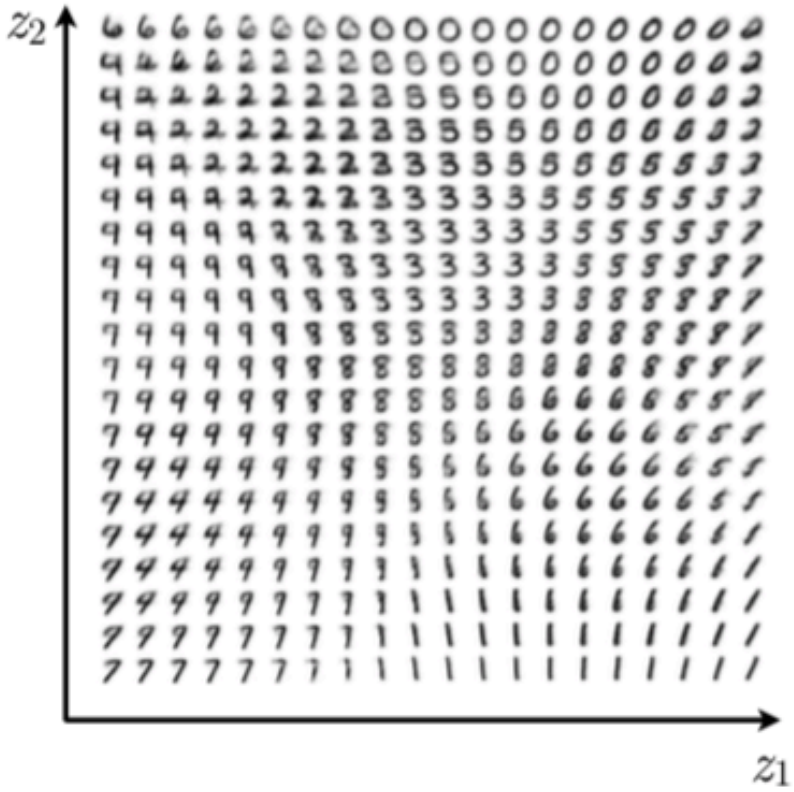


- In principle we can think of:
- **Encoder**: $p(z|x)$
- **Decoder**: $p(x|z)$
- **Objective function**: ML or MAP on $p(x_1, \ldots, x_n)$

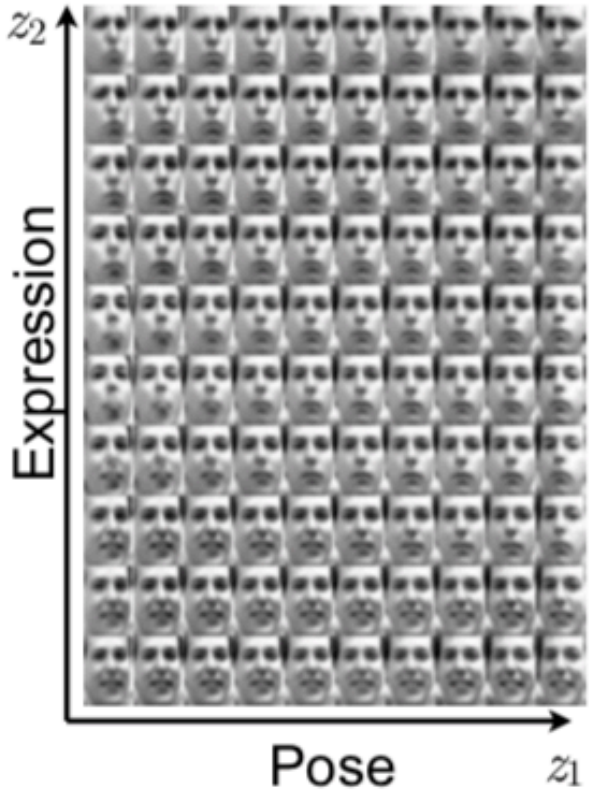


What VAE can do? Learn manifolds

Learned MNIST manifold

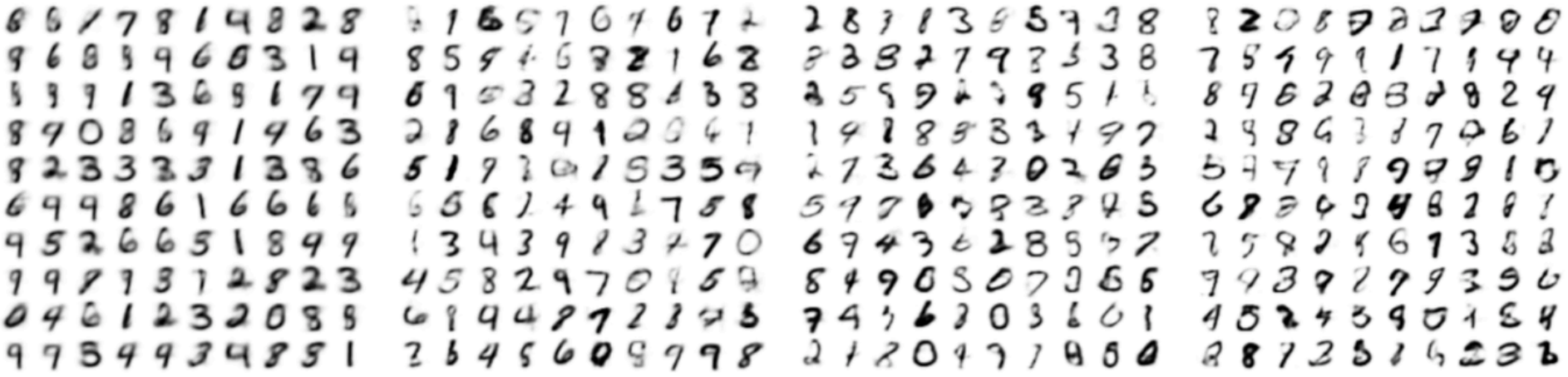


Learned face manifold



What VAE can do?

Sample from generative learned model

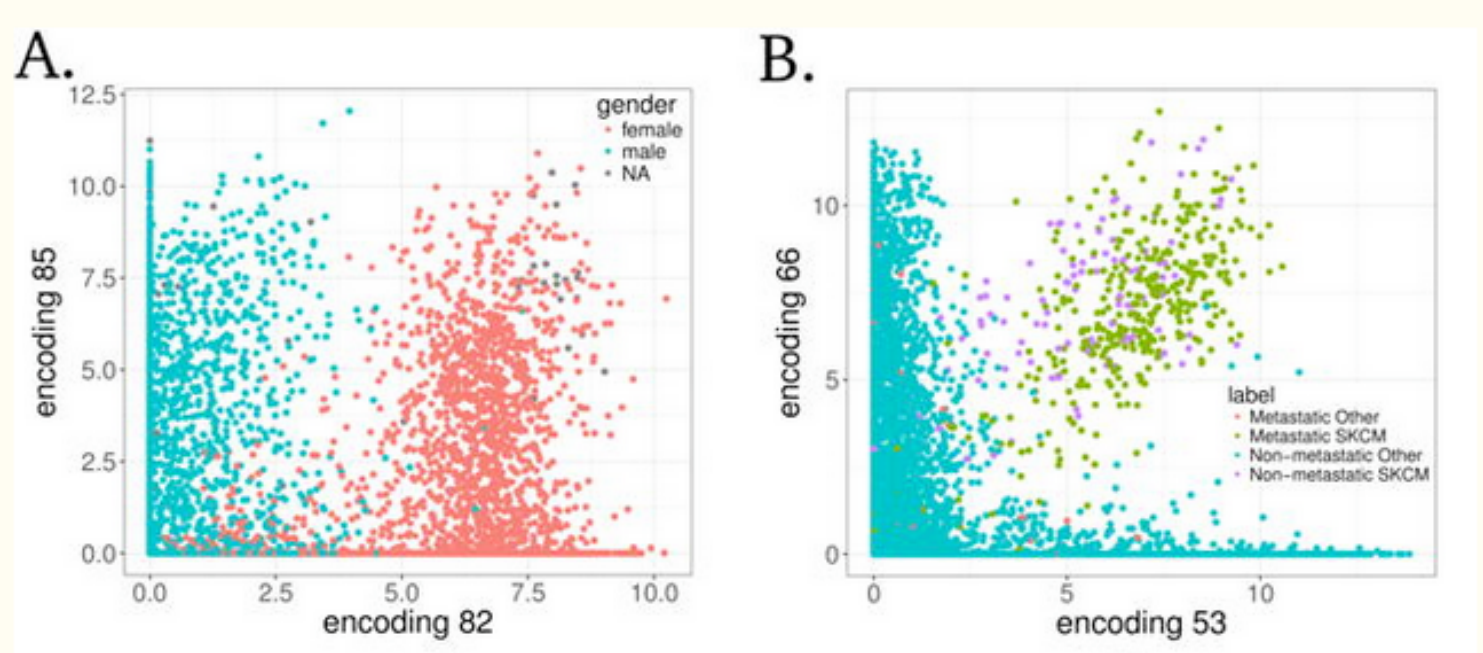


(a) 2-D latent space (b) 5-D latent space (c) 10-D latent space (d) 20-D latent space

Figure 5: Random samples from learned generative models of MNIST for different dimensionalities of latent space.

What VAE can do? Interpret gene expressions

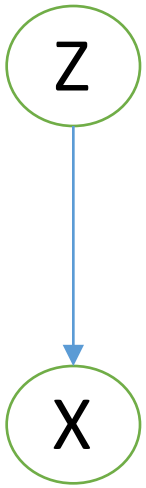
- Extracting a biologically relevant latent space: (Way et al, 2018)
  - Encoding 5,000 gene expression vectors into a vector of 100 dimensions
  - Distinguishing patients gender and metastatic activation



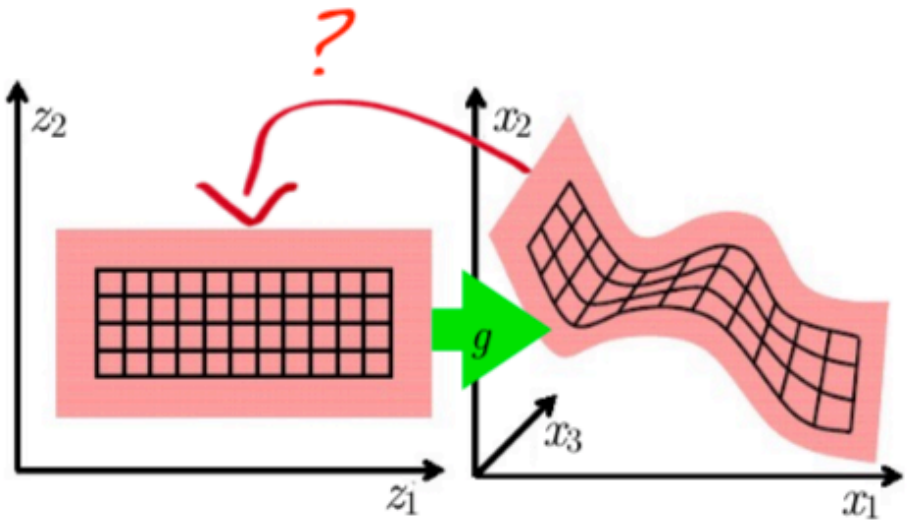
The inference / learning challenge

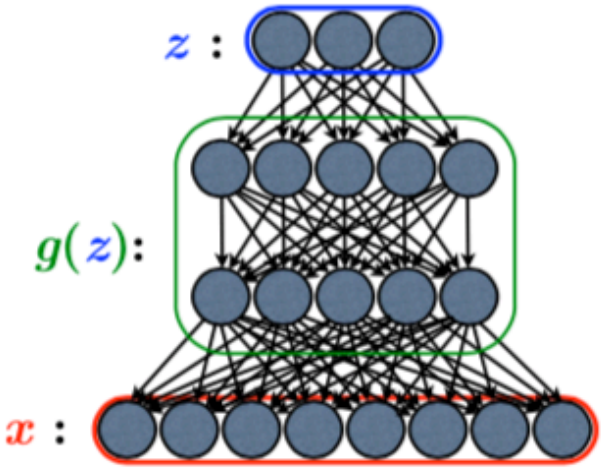
- **Where does z come from?** — The classic directed model dilemma.
- Computing the posterior $p(\boldsymbol{z} \mid \boldsymbol{x})$ is intractable.
- We need it to train the directed model.

Latent variable



Observed Sample variable





Example:

Decoder:

$p_\theta(z): Z \sim N(0, I)$

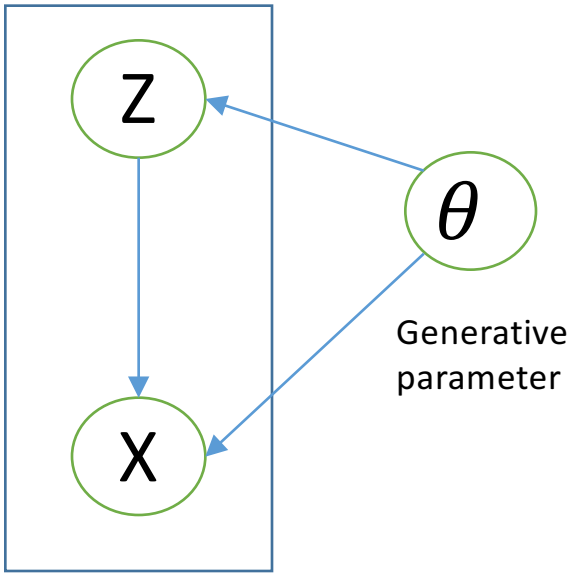
$p_\theta(x|z): X|Z \sim No\left(\mu_{(z,\theta)}, \sigma^2_{(z,\theta)} I\right)$

where $\mu = W_1 h + b_1$

$\log \sigma^2 = W_2 h + b_2$ (componentwise)

$h = \tanh(W_3 z + b_3)$ (componentwise)

so $\theta := \{W_1, W_2, W_3, b_1, b_2, b_3\}$



Encoder:

$$p_\theta(z|x) = \frac{p_\theta(x|z)p_\theta(z)}{p_\theta(x)} = \frac{p_\theta(x|z)p_\theta(z)}{\int p_\theta(x|z)dz}$$

Usually true posterior is intractable ☹

- The VAE approach: introduce an simplified posterior $\boldsymbol{q_\phi(z|x)}$ that learns to approximate the true posterior $p_\theta(z|x)$
- For this to happen we use a **variational lower bound (aka ELBO):**

$$\log\big(p(\boldsymbol{x})\big) = \mathcal{L}(\theta, \phi, x) + KL(q||p)$$

where : $KL(q||p) = \mathbb{E}_{q(Z|x)}\left[\log\left(\frac{q(Z|x)}{p(Z|x)}\right)\right] \geq 0$

$$\mathcal{L}(\theta, \phi, x) = \mathbb{E}_{q(Z|x)}\big[\log\big(p(x, Z)\big) - \log\big(q(Z|x)\big)\big]$$

$$= \mathbb{E}_{q(Z|x)}\big[\log\big(p(x|Z)\big) + \log(p(Z)) - \log\big(q(Z|x)\big)\big]$$

$$= \mathbb{E}_{\boldsymbol{q(Z|x)}}\big[\boldsymbol{log\big(p(x|Z)\big)}\big] - \boldsymbol{KL(q(Z|x)||p(Z))}$$

reconstruction term **regularization term**

- We include a neural network into $q_\phi(z|x)$ and now we solve:

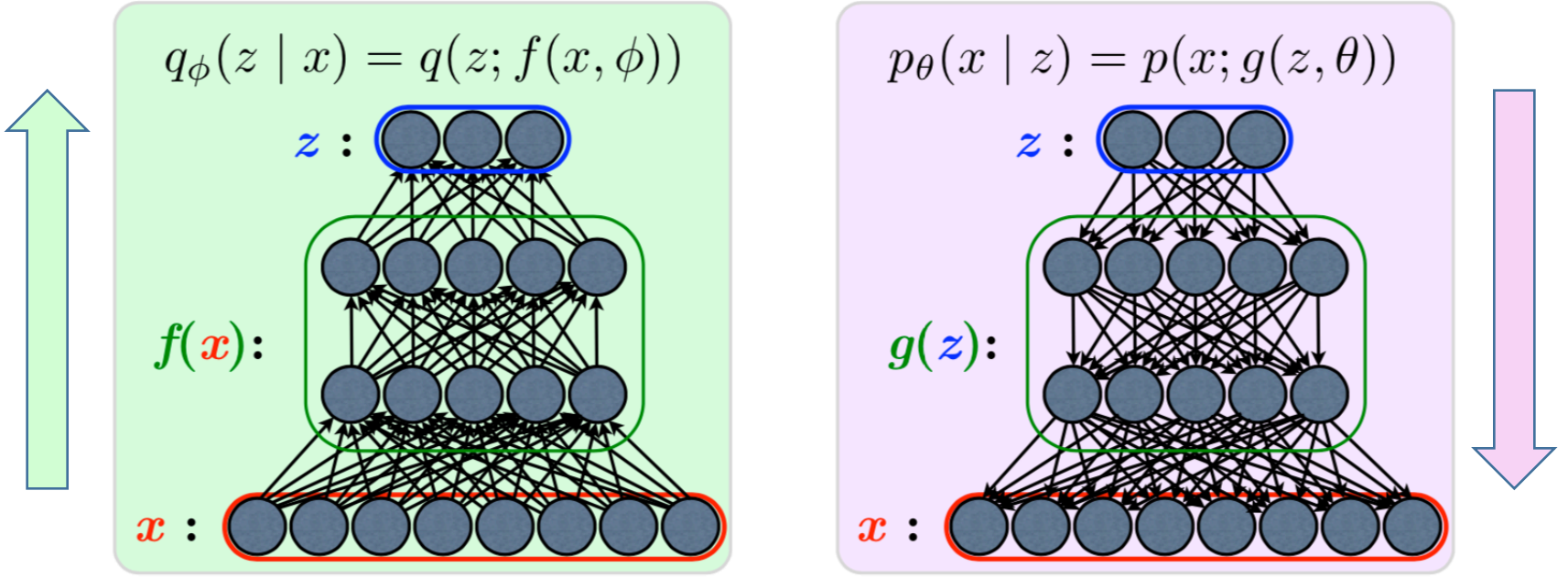
$$\max_{\theta,\phi} \mathcal{L}(\theta,\phi,x) = \max_{\theta,\phi} \mathbb{E}_{q_\phi(Z|x)} \left[\log\left(p_\theta(x,Z)\right) - \log\left(q_\phi(Z|x)\right)\right]$$

Encoder

Z $given$ X, ϕ

Decoder

X $given$ Z, θ



Example:

Decoder:

$p_\theta(z): Z \sim N(0, I)$

$p_\theta(x|z): X|Z = z \sim No\left(\mu_{(z,\theta)}, \sigma^2_{(z,\theta)} I\right)$

where $\mu = W_1 h + b_1$

$\log \sigma^2 = W_2 h + b_2$ (componentwise)

$h = \tanh(W_3 z + b_3)$ (componentwise)

so $\theta \coloneqq \{W_1, W_2, W_3, b_1, b_2, b_3\}$

Encoder:

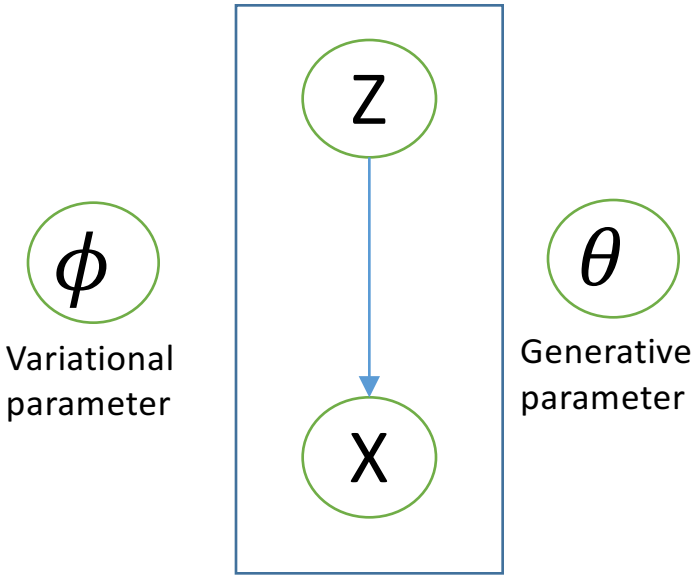
$q_\phi(z|x): Z|X = x \sim No\left(\tilde{\mu}_{(x,\phi)}, \tilde{\sigma}^2_{(x,\phi)} I\right)$

where $\tilde{\mu} = \tilde{W}_1 h + \tilde{b}_1$

$\log \tilde{\sigma}^2 = \tilde{W}_2 h + \tilde{b}_2$ (componentwise)

$h = \tanh(\tilde{W}_3 x + \tilde{b}_3)$ (componentwise)

so $\phi \coloneqq \{\tilde{W}_1, \tilde{W}_2, \tilde{W}_3, \tilde{b}_1, \tilde{b}_2, \tilde{b}_3\}$



Reparameterization Trick

- We want to maximize the ELBO, wrt θ and ϕ using stochastic gradient descent

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[-\log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) + \log p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})\right]$$

- Usual (naïve) Monte Carlo gradient exhibits high variance

$$\nabla_{\boldsymbol{\phi}} \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z})}\left[f(\mathbf{z})\right] = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z})}\left[f(\mathbf{z}) \nabla_{q_{\boldsymbol{\phi}}(\mathbf{z})} \log q_{\boldsymbol{\phi}}(\mathbf{z})\right] \simeq \frac{1}{L} \sum_{l=1}^{L} f(\mathbf{z}) \nabla_{q_{\boldsymbol{\phi}}(\mathbf{z}^{(l)})} \log q_{\boldsymbol{\phi}}(\mathbf{z}^{(l)})$$

- We use **Stochastic Gradient Variational Bayes (SGVB)** (Kingma 2014)

$$q_\phi(z|x): Z|X = x \sim g_\phi(\epsilon, x) \text{ with } \epsilon \sim known\ distribution$$

- Example:

$$q_\phi(z|x): Z|X = x \sim No\left(\tilde{\mu}_{(x,\phi)}, \tilde{\sigma}^2_{(x,\phi)} I\right)$$

Sample using: $Z = \tilde{\mu} + \tilde{\sigma}^2 \epsilon$ with $\epsilon \sim N(0, I)$

VAE Pros

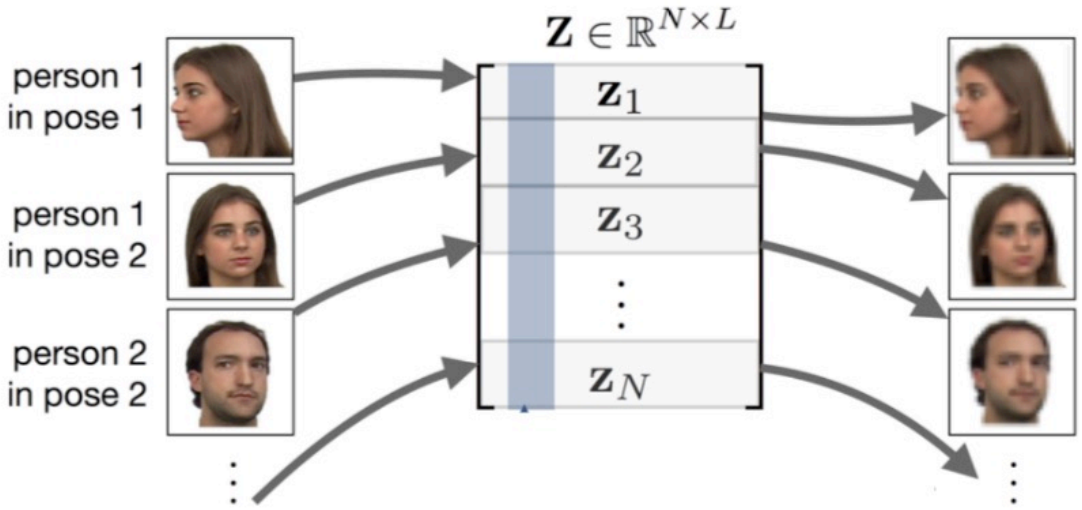
- Easy to train
- Blurry result due to minimizing the MSE based reconstruction error
- Nice probabilistic formulation, easy to introduce prior

VAE Cons

- In most VAE formulations latent variables Z are iid

- Ex: Multiple sequences of images from different cars, or medical image sequences from multiple patients.

- VAE prior should capture multiple levels of correlations at once, including time, object identities, etc.



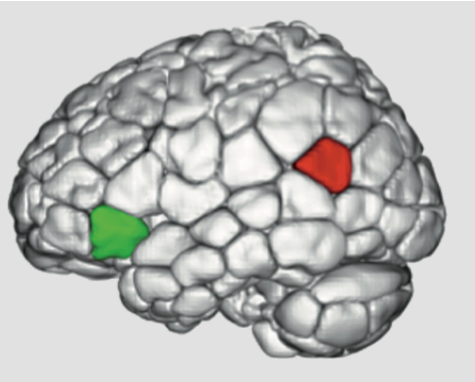
How to model the following:

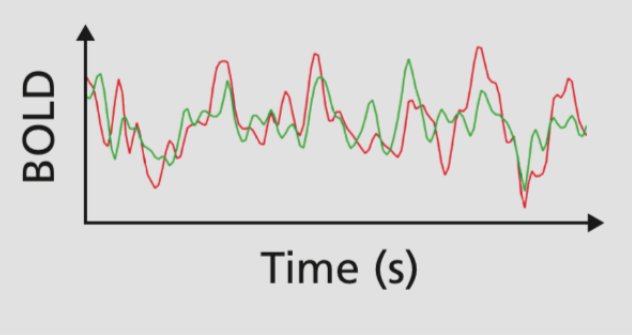
"Object (person) p_n in view (pose) q_n"???

We need some correlation in $z_1, \ldots, z_n$

More examples

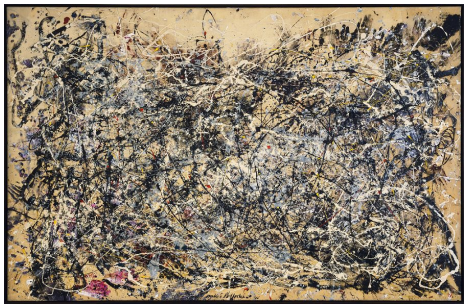
Neuroimaging: *fMRI signals (**sample**) of voxels (**objects**) under different motor tasks (**views**)*

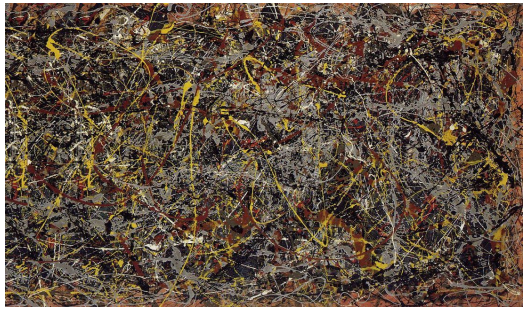




- Right hand
- Left hand
- Right foot
- Left foot
- Tongue

Art Images: *J. Pollock paintings (**sample**), Number (**objects**) Series 1,5, 8 (**views**)*

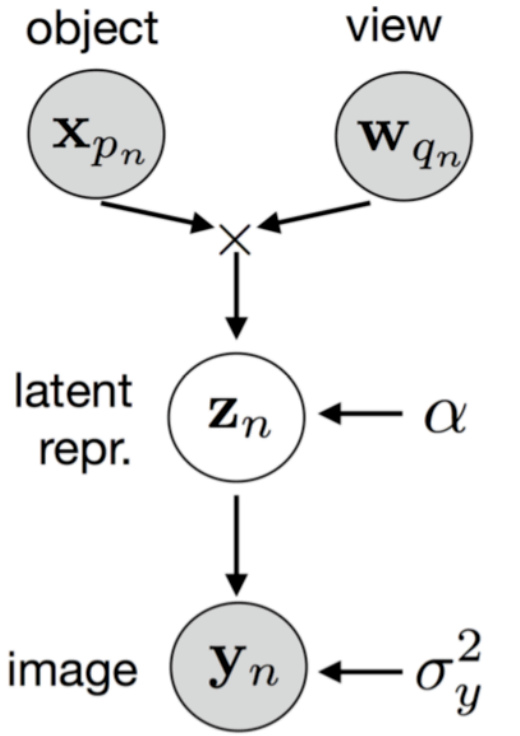




Casale et al (2018)
Gaussian Process Prior Variational Autoencoders

- $y_1, \ldots y_n$ K-dimensional **samples** (not necessarily iid)
- $z_1, \ldots z_n$ L-dimensional **latent variables**
- $x_1, \ldots x_P$ M-dimensional **object feature vectors**
- $w_1, \ldots w_Q$ R-dimensional **view feature vectors**

$N =$ # images
$L =$ # latent dims
$P =$ # objects (people)
$Q =$ # views (poses)



$n = 1, \ldots, N$

Decoder

$$p(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{W}, \boldsymbol{\phi}, \sigma_y^2, \boldsymbol{\theta}, \alpha) = \int p(\boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{\phi}, \sigma_y^2) p(\boldsymbol{Z} \mid \boldsymbol{X}, \boldsymbol{W}, \boldsymbol{\theta}, \alpha)\, d\boldsymbol{Z}$$

$$\boldsymbol{z}_n = f(\boldsymbol{x}_{p_n}, \boldsymbol{w}_{q_n}) + \boldsymbol{\eta}_n, \text{ where } \boldsymbol{\eta}_n \sim \mathcal{N}(\boldsymbol{0}, \alpha \boldsymbol{I}_L);$$

where $f(\cdot)$ generates z_n with a GP prior:

$$p(\boldsymbol{Z} \mid \boldsymbol{X}, \boldsymbol{W}, \boldsymbol{\theta}, \alpha) = \prod_{l=1}^{L} \mathcal{N}(\boldsymbol{z}^l \mid \boldsymbol{0}, \boldsymbol{K}_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{W}) + \alpha \boldsymbol{I}_N)$$

with covariance between samples n, m

$$\boldsymbol{K}_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{W})_{nm} = \mathcal{K}_{\boldsymbol{\theta}}^{(\text{view})}(\boldsymbol{w}_{q_n}, \boldsymbol{w}_{q_m}) \mathcal{K}_{\boldsymbol{\theta}}^{(\text{object})}(\boldsymbol{x}_{p_n}, \boldsymbol{x}_{p_m})$$

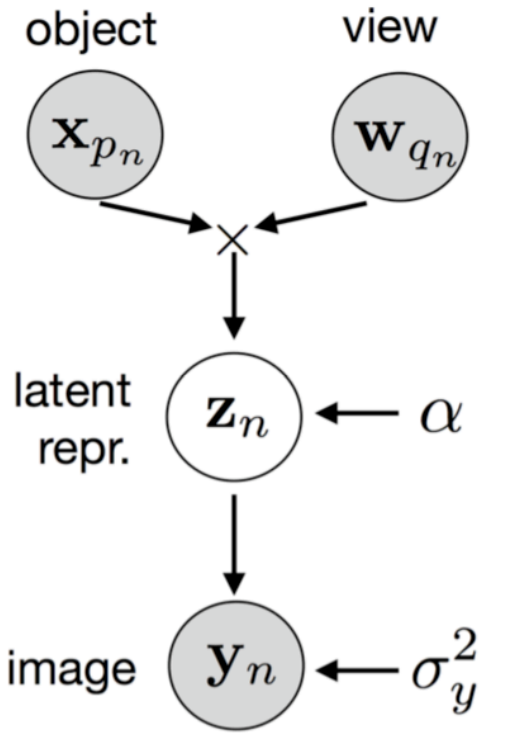
$$\boldsymbol{y}_n = g(\boldsymbol{z}_n) + \boldsymbol{\epsilon}_n, \text{ where } \boldsymbol{\epsilon}_n \sim \mathcal{N}(\boldsymbol{0}, \sigma_y^2 \boldsymbol{I}_K)$$

where g(·) projects z_n to a high-dimensional space.

For images: a Convolutional Neural Net

For cntns variable:

activation function + linear transformation

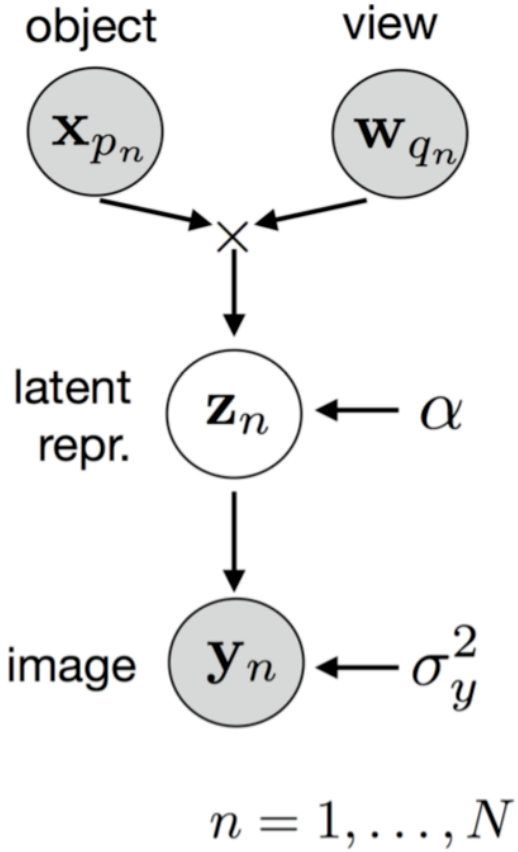


$$n = 1, \ldots, N$$

Encoder

$$q_{\psi}(\boldsymbol{Z} \mid \boldsymbol{Y}) = \prod_{n} \mathcal{N}\left(\boldsymbol{z}_n \mid \boldsymbol{\mu}^{z}_{\psi}(\boldsymbol{y}_n), \mathrm{diag}(\boldsymbol{\sigma}^{z2}_{\psi}(\boldsymbol{y}_n))\right)$$

where $\boldsymbol{\mu}^{\boldsymbol{z}}_{\boldsymbol{\psi}}$ and $\boldsymbol{\sigma}^{\boldsymbol{z}}_{\boldsymbol{\psi}}{}^{\boldsymbol{2}}$ are the variational parameters and are a neural network function of the observed data.



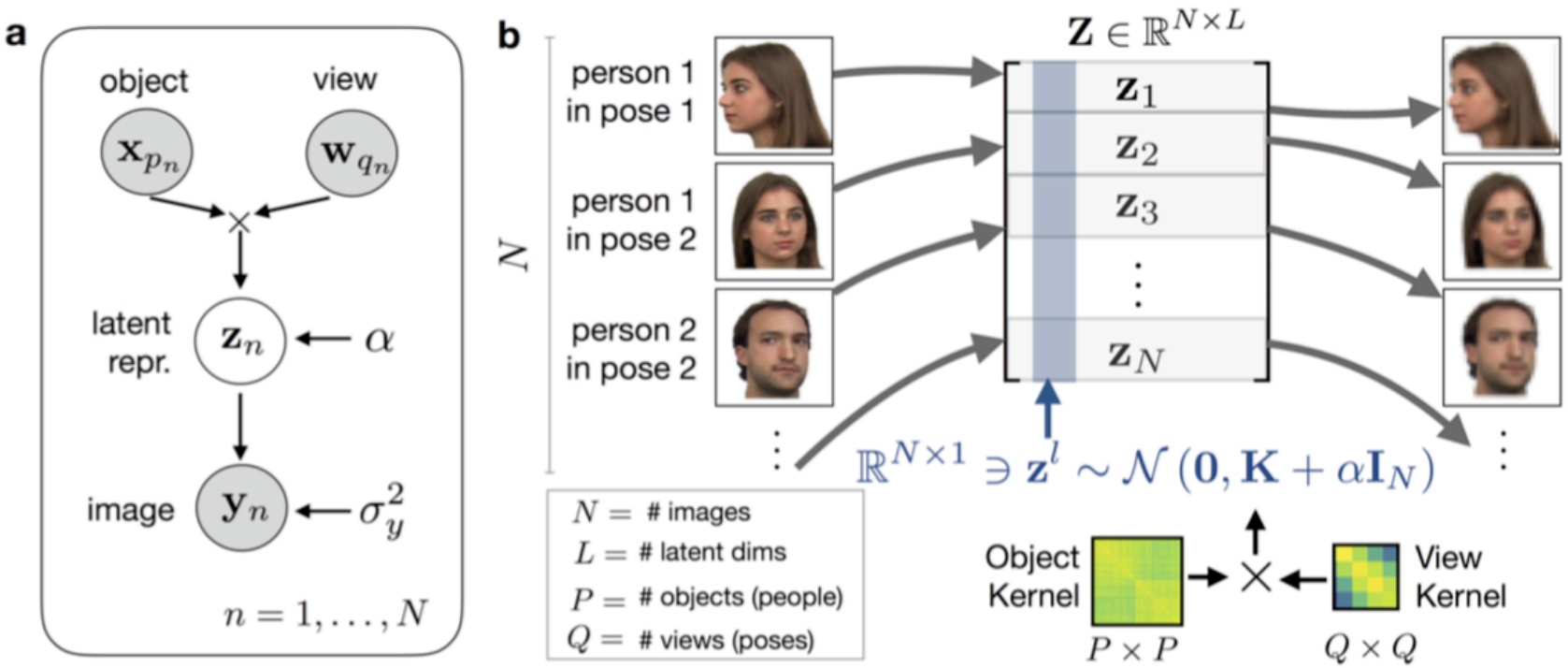


Figure 1: (**a**) Generative model underlying the proposed GPVAE. (**b**) Representation of the inference procedure in GPVAE. Each sample (here an image) is encoded in a low-dimensional space and then decoded to the original space. Covariances between samples modeled through a GP prior on each column of the latent representation matrix $\boldsymbol{Z}$.

ELBO and Estimation

Variational lower bound:

$$\log p(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{W}, \boldsymbol{\phi}, \sigma_y^2, \boldsymbol{\theta}) \geq \mathbb{E}_{\boldsymbol{Z} \sim q_{\boldsymbol{\psi}}} \left[\sum_n \log \mathcal{N}(\boldsymbol{y}_n \mid g_{\boldsymbol{\phi}}(\boldsymbol{z}_n), \sigma_y^2 \boldsymbol{I}_K) + \log p(\boldsymbol{Z} \mid \boldsymbol{X}, \boldsymbol{W}, \boldsymbol{\theta}, \alpha) \right] + + \frac{1}{2} \sum_{nl} \log(\boldsymbol{\sigma}^{z\,2}_{\boldsymbol{\psi}}(\boldsymbol{y}_n)_l) + \text{const.}$$

Stochastic Backpropagation:

$$\mathcal{L}\left(\boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\theta}, \alpha, \sigma_y^2\right) = \frac{1}{K} \underbrace{\sum_n^N \left(\boldsymbol{y}_n - g_{\boldsymbol{\phi}}(\boldsymbol{z}_{\boldsymbol{\psi}\,n})\right)^2}_{\text{reconstruction term}} - \frac{\lambda}{L} \left[\underbrace{\log p\left(\boldsymbol{Z}_{\boldsymbol{\psi}} \mid \boldsymbol{X}, \boldsymbol{W}, \boldsymbol{\theta}, \alpha\right)}_{\text{latent-space GP model}} + \underbrace{\frac{1}{2} \sum_{nl} \log(\boldsymbol{\sigma}^{z\,2}_{\boldsymbol{\psi}}(\boldsymbol{y}_n)_l)}_{\text{regularization}} \right]$$

Where λ is a trade-off parameter between the data reconstruction term and the goodness of the latent space GP model

Reparameterization Trick

Remember that our posterior approximation is

$$q_{\boldsymbol{\psi}}(\boldsymbol{Z} \mid \boldsymbol{Y}) = \prod_n \mathcal{N}\left(\boldsymbol{z}_n \mid \boldsymbol{\mu}^z_{\boldsymbol{\psi}}(\boldsymbol{y}_n), \mathrm{diag}(\boldsymbol{\sigma}^{z^2}_{\boldsymbol{\psi}}(\boldsymbol{y}_n))\right)$$

and latent representations $\boldsymbol{Z}_{\boldsymbol{\psi}} = \left[\boldsymbol{z}_{\boldsymbol{\psi}1}, \ldots, \boldsymbol{z}_{\boldsymbol{\psi}N}\right] \in \mathbb{R}^{N \times L}$ are sampled as

$$\boldsymbol{z}_{\boldsymbol{\psi}_n} = \mu^z_{\boldsymbol{\psi}}(\boldsymbol{y}_n) + \boldsymbol{\epsilon}_n \odot \sigma^z{}_{\boldsymbol{\psi}}(\boldsymbol{y}_n),\ \boldsymbol{\epsilon}_n \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_{L \times L}),\ n = 1, \ldots, N$$

where $\odot$ represents the Hadamard product.

Predictive posterior sampling

$$p(\boldsymbol{y}_\star \mid \boldsymbol{x}_\star, \boldsymbol{w}_\star, \boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{W}) \approx \int \underbrace{p(\boldsymbol{y}_\star \mid \boldsymbol{z}_\star)}_{\text{decode GP prediction}} \underbrace{p(\boldsymbol{z}_\star \mid \boldsymbol{x}_\star, \boldsymbol{w}_\star, \boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{W})}_{\text{latent-space GP predictive posterior}} \underbrace{q(\boldsymbol{Z} \mid \boldsymbol{Y})}_{\text{encode training data}} d\boldsymbol{z}_\star d\boldsymbol{Z}$$

this term replaces the true posterior $p_\theta(z|y)$

Compared Methods

1. **GPVAE-joint**: In each iteration
   a) Fix kernel K_θ, and train all the other parameters
   b) Fix all other parameters and train K_θ for 100 epochs
   c) Train everything simultaneously for 1 epoch
2. **GPVAE-disjoint:** In each iteration iterate steps a) and b) of GPVAE-joint
3. **Conditional VAE (CVAE)**
4. **Linear Interpolation in VAE latents space (LIVAE)**

Example: Rotated MNIST (views known)

- **Data**: 400 handwritten versions of digit three with R= 16 rotation angles
- **Training Set**: 4,050 images spanning 15 rotations
- **Test Set**: 270 images with one rotation (out of sample)
- **Latent Space dimension**: L=16 **Object feature vector**: M=8

$$\boldsymbol{K}_{\boldsymbol{\theta}}(\boldsymbol{X},\boldsymbol{w})_{nm} = \beta \underbrace{\exp\left(-\frac{2\sin^2|w_{q_n} - w_{q_m}|}{\nu^2}\right)}_{\text{rotation kernel}} \cdot \underbrace{\boldsymbol{x}_{p_n}^T \boldsymbol{x}_{p_m}}_{\text{digit draw kernel}} \qquad \begin{matrix} where \\ \theta = \{\beta, \nu \geq 0\} \end{matrix}$$

Example: Rotated MNIST (views known)

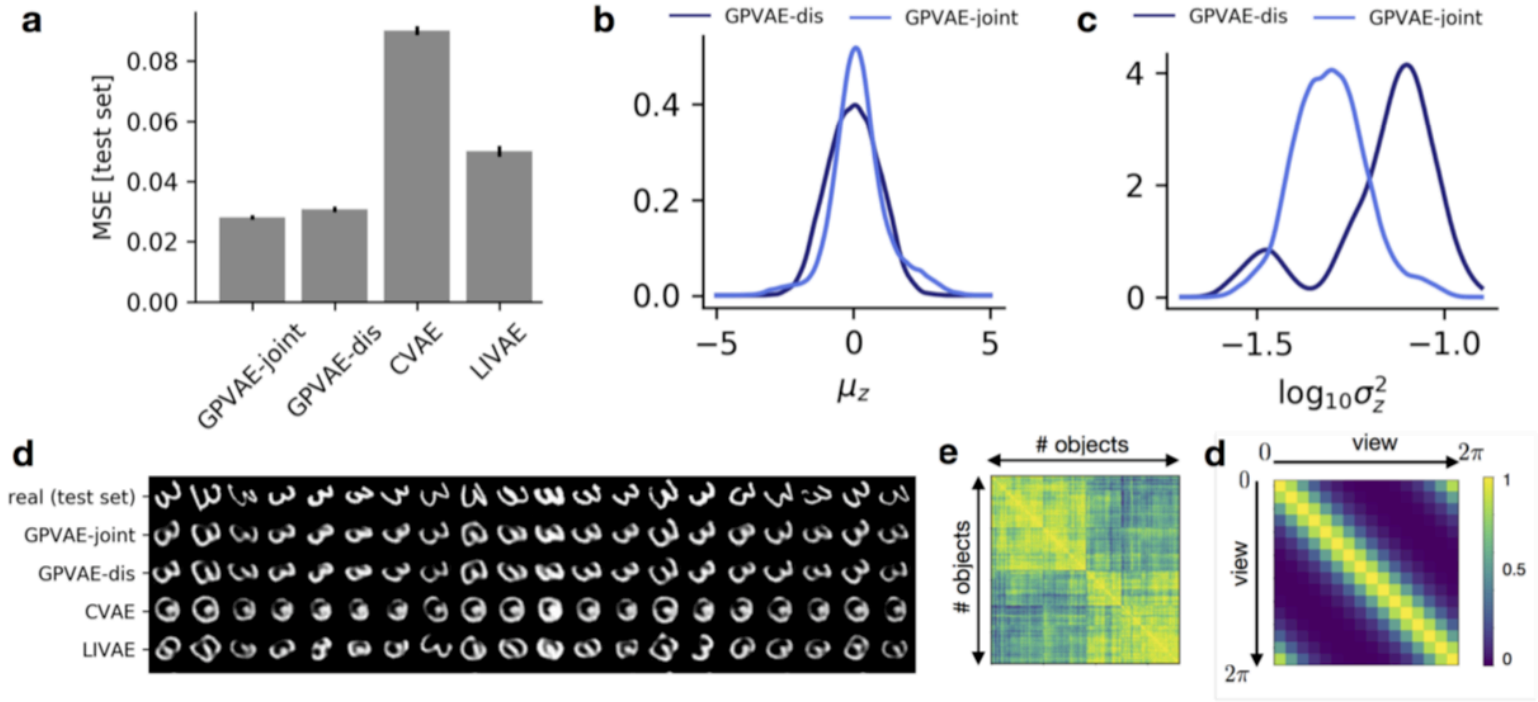
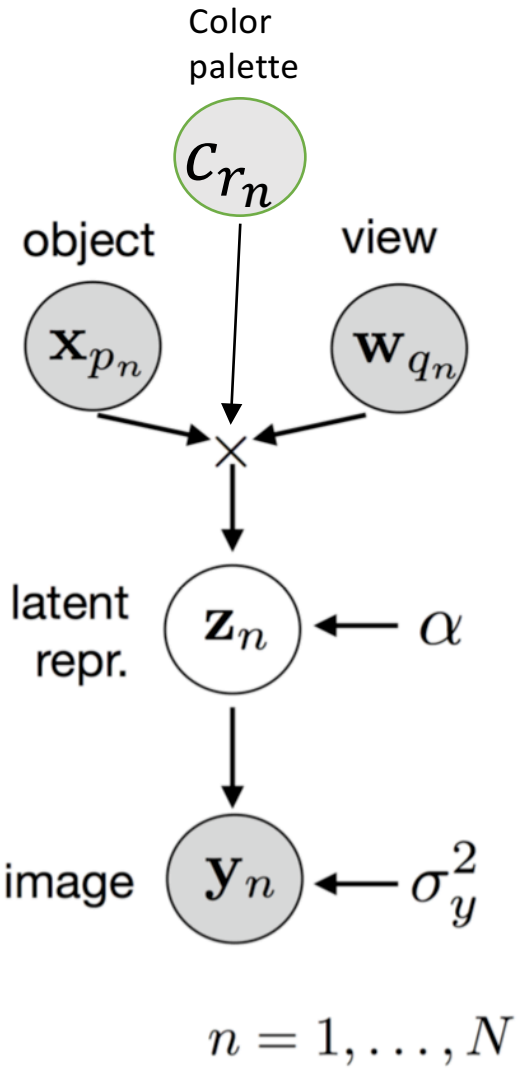


Figure 2: Results from experiments on rotated MNIST. (**a**) Mean squared error on test set. Error bars represent standard error of per-sample MSE. (**b**) Empirical density of estimated means of q_ψ, aggregated over all latent dimensions. (**c**) Empirical density of estimated log variances of q_ψ. (**d**) Out-of-sample predictions for ten random draws of digit "3" at the out-of-sample rotation state. (**e**, **f**) Object and view covariances learned through GPVAE-joint.

My Discussion: A hard example?

Andy Warhol's Mao Series #90-99

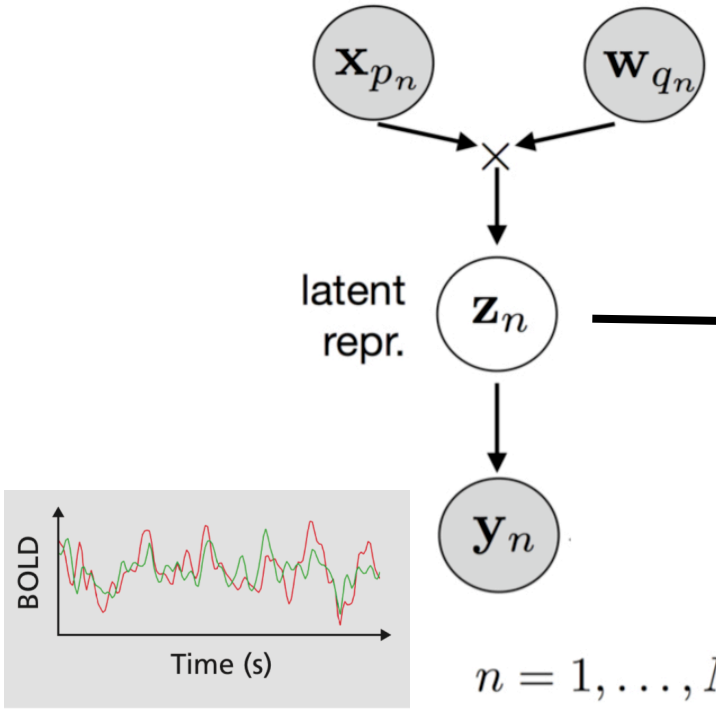
Andy Warhol's Marilyn Monroe Series #22-31



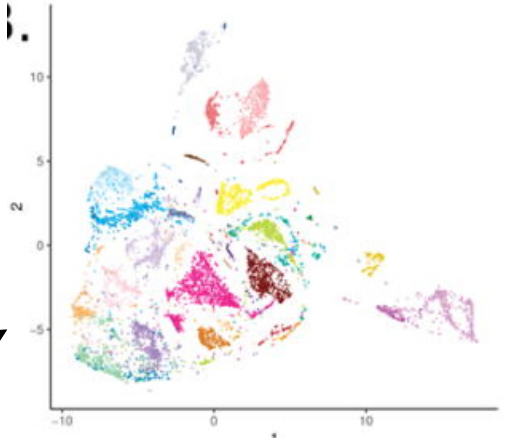
My Discussion: neuroimaging

signal (object)
(Functional PCA ??)

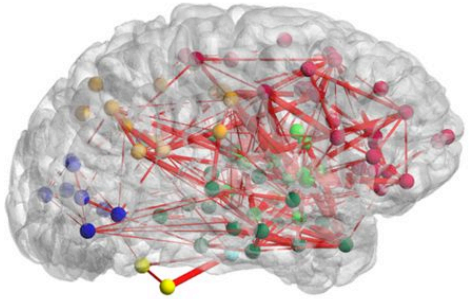
Motor Task
(view)



t-SNE Clustering
In latent space



Graphical
Lasso



Bibliography

- Courville - Variational Autoencoder and Extensions(http://videolectures.net/deeplearning2015_courville_autoencoder_extension/)
- Wang - Deep Generative Models (http://www.cs.toronto.edu/~slwang/generative_model.pdf)
- Way et. al. - Extracting a biologically relevant latent space from cancer transcriptomes with variational autoencoders (2018)
- Kingma et al - Auto-Encoding Variational Bayes (2014)
- Rezende et al - Stochastic back-propagation and variational inference in deep latent Gaussian models (2014)
- Casale et al – Gaussian Process Prior Variational Autoencoders (2018)