Manifold Learning with Variational Auto-encoder for Medical Image Analysis

Eunbyung Park
Department of Computer Science
University of North Carolina at Chapel Hill
eunbyung@cs.unc.edu

Abstract

Manifold learning of medical images has been successfully used for many applications, such as segmentation, registration, and classification of clinical parameters by modeling anatomical variability. In many applications, two aspects, generative property and capturing shape variability have been considered very important[4]. In this project, we analyze brain MRI images by applying variational auto-encoder(VAE)[7, 8], which was introduced very recently and has received much attention in machine learning and computer vision community due to its promising generative results and manifold learning perspective. We evaluate the VAE on the OASIS dataset and experimental results show that it can learn low dimensional manifold that can be used for generation and classification of many clinical parameters. such as age, MMSE, and CDR.

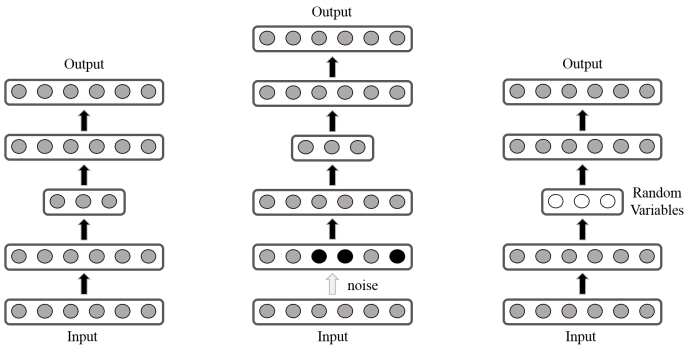
1 Introduction

In medical image modalities, such as MRI and CT images, they can be regarded as a data point in a very high-dimensional space, while the real data only lies in a much lower intrinsic dimension space. The manifold learning have been suggested for uncovering meaningful low dimensional space from the data in high dimensional space. In contrast to linear dimensionality reduction techniques, such as principle component analysis(PCA), the manifold learning can provide more powerful non-linear dimensionality reduction by preserving the local structure of the input data. Many applications, including clustering and classification, now become much more effective in the transformed low dimensional space.

There have been various approaches for manifold learning, such as locally linear embedding(LLE)[10], Laplacian eigenmaps(LEM)[2], Isomaps[12], and so on. Most of existing approaches are based on the proximity graph that requires the assumption that the manifold space is locally linear. In addition, they are very sensitive to the choice of a distance measure, which means we should explore appropriate distance measures for each image modalities and tasks[4]. In this project, we propose to apply the variational autoencoder(VAE), which is one of the deep learning methods, for learning the manifold without the assumption of linearity and specific distance measure.

In many neuroimaging applications, two important aspects have been considered in manifold learning[4]. First, it should be able to capture shape variability across the sets of images since shape is a statistically significant predictor for various clinical studies. Another important property is generative capability that can construct brain images given manifold coordinates. Unlike existing auto-encoder based methods, the proposed method VAE inherently has generative property.[1]

[1]There have been some ideas about the probabilistic interpretation of auto-encoders as a generative model.[3]



(a) Basic (b) Denoising[15] (c) Variational[7, 8]

Figure 1: Various auto-encoder models

2 Variational Auto-encoder

2.1 Various auto-encoder models

Figure 1 shows various auto-encoder models. (a) is a basic form of auto-encoder. First, it takes an input and the input goes through an encoder, which gives us low dimensional output. We can interpret this as coordinates of the manifold. Second, it takes the output of the encoder and produces reconstructed outputs with same dimension as the input. We optimize this model with reconstruction error between the input and the output, e.g. squared error or bernoulli cross entropy. The performance of this traditional auto-encoder has fallen short compared to RBM(Restricted Boltzmann Machine) approaches[5] in terms of good feature learning. There have been many approaches in order to improve performance. [15] suggested denoising auto-encoder depicted in (b). It inserted noise to the inputs before the inputs are fed into the encoder in order to learn features that are more robust to small perturbations of the input. [9] suggested contractive auto-encoder that introduce sensitivity penalization term in the objective function, measured as the Frobenius norm of Jacobian of the non-linear mapping of the inputs. It encourages the model to be less sensitive to small variations around example. Very recently, variational auto-encoder was introduced and its representation units are random variables. In other words, the model itself is probabilistic directed graphical model unlike the previous deterministic models.

2.2 Variational auto-encoder

VAE is a deep directed graphical model with latent variable $\mathbf{z}$(outputs of the encoder). It is known to be intractable to compute posterior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$. In the VAE framework, $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ is introduced which learns to approximate the true posterior by optimizing the variational lower bound. It uses encoder network to map input image into continuous latent variables($q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$) and uses decoder network to map latent variables to reconstructed image($p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$). The variational lower bound for individual datapoint $\mathbf{x}_i$ can be written as following,

$$L(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}_i) = -D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}_i)||p_{\boldsymbol{\theta}}(\mathbf{z})) + \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}_i)}\big[\log p_{\boldsymbol{\theta}}(\mathbf{x}_i|\mathbf{z})\big] \tag{1}$$

The first RHS term is the KL divergence of the approximate from the true posterior. And the second RHS term is expected reconstruction error w.r.t the approximate posterior $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}_i)$. In order to optimize the lower bound, we might want to differentiate $L(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}_i)$ and do gradient descent with standard back propagation algorithm. In our model, we assumed that both $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}_i)$ and $p_{\boldsymbol{\theta}}(\mathbf{z})$ are gaussian. So, we can integrate KL divergence term analytically.

$$-D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}_i)||p_{\boldsymbol{\theta}}(\mathbf{z})) = \frac{1}{2}\sum_{j=1}^{J}(1 + \log(\boldsymbol{\sigma}_j^2) - \boldsymbol{\mu}_j^2 - \boldsymbol{\sigma}_j^2) \tag{2}$$

, where J is the number of dimension of $\mathbf{z}$. Mean $\boldsymbol{\mu}$ and standard deviation $\boldsymbol{\sigma}$ are simply outputs of encoder function of $\mathbf{x}$ and the variational parameter $\boldsymbol{\phi}$. However, for the expected reconstruction term, the gradient of $\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}_i)}\big[\log p_{\boldsymbol{\theta}}(\mathbf{x}_i|\mathbf{z}))\big]$ is not straightforward. [7, 8] suggested practical estimator of its derivatives w.r.t. the parameters with the reparameterization trick. They reparameterize the random variable $\mathbf{z}$ using a differentiable transformation with auxiliary noise random variable $\boldsymbol{\epsilon}$. And now we can form Monte Carlo estimates of expectation of transformed function $f(\mathbf{z})$. The resulting estimator for a datapoint $\mathbf{x}_i$ is

$$L(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}_i) \approx \frac{1}{2}\sum_{j=1}^{J}(1 + \log(\boldsymbol{\sigma}_j^2) - \boldsymbol{\mu}_j^2 - \boldsymbol{\sigma}_j^2) + \frac{1}{L}\sum_{l=1}^{L}\log p_{\boldsymbol{\theta}}(\mathbf{x}_i|\mathbf{z}_{i,l}))$$
$$\text{where} \quad \mathbf{z}_{i,j} = \boldsymbol{\mu}_i + \boldsymbol{\sigma}_i \odot \boldsymbol{\epsilon}_l \quad \text{and} \quad \boldsymbol{\epsilon}_i \sim \mathcal{N}(0, \mathbf{I})$$

Again, we can compute $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ with deterministic encoder network. For $\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$, we can use bernoulli cross-entropy loss function with deterministic decoder network. You can find more detail version in [7].

In practice, we put data examples into the encoder network and get the parameters of distribution of the latent variables, e.g. mean and standard deviation for gaussian distribution. And, we sample from the distribution of the latent variables given the parameters. Once we get the samples, we put them into the decoder network and compute loss function, e.g. bernoulli cross-entropy. Because we have differentiable lower bound estimates, we can do back propagation to compute gradient w.r.t the parameters of the encoder and decoder network.

3 Experiments

3.1 OASIS Dataset

The OASIS brain database consists of T1 weighted MRI of 416 subjects aged between 18 and 96. It contains several clinical parameters, such as age, mini mental state examination (MMSE), clinical dementia rating (CDR) and so on. Image resolution is 176x208x176. We trained the network with 2d image using only one axial slice(middle) of volumetric brain images.

3.2 Training and Implementation

We used torch library for implementation[1]. For the encoder and decoder architecture, we chose the convolutional encoder and decoder inspired by [11]. All layers are convolutional, upsampling, and downsampling layers. We used *rmsprop*[13] as a gradient descent method. We set learning rate as -0.0005 and batchsize 16, and go through around 30000 iterations.

3.3 Results and Discussion

First, We project input data into 2D manifold space using the proposed method. We visualize learned 2D manifold space in Figure 2. Prior of the latent space is gaussian. So, we transformed unit square coordinate space according to inverse CDF of the gaussian to produce values of the latent variables. Here, we have equally spaced 11x11 grid. Note that all images in Figure 2 are generated images. As you noted, it can capture the shape variability very well. From the images in left-bottom side to the images in right-top side, the size of *X* shape in the middle of the brain(ventricle) are gradually changed.

In 2D latent space, we couldn't get very clear image. We have gotten blurry images except for ventricle part, which is the most significant visual attribute. Furthermore, we might also lose another important visual information. Therefore, we also trained the network with more high dimensional latent space(120). It turns out that the images with high dimensional latent space are much more clear. We couldn't tell the difference between real images and generated images. We show some of real images on latent space in figure 3. With the network with high dimensional latent space, we used t-sne[14] method for visualization. Here, we project original real images into 120 latent dimensional space, and find 2 dimensional position with t-sne method for visualization.

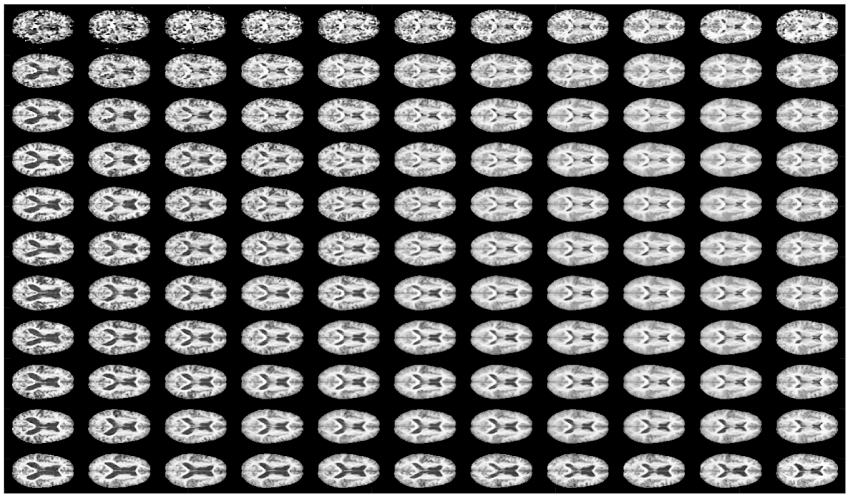


Figure 2: Visualization of 2d manifold space(Note that all images are synthesized images)

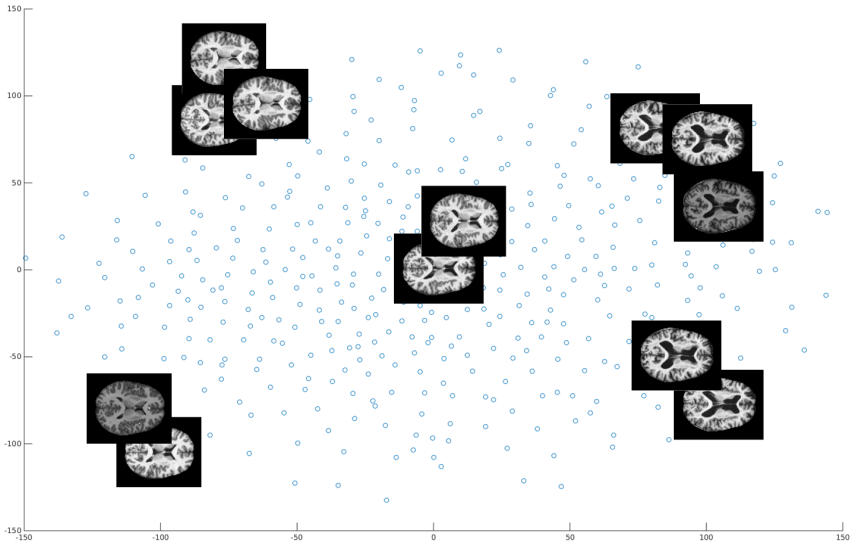


Figure 3: T-sne visualization of learned high dimensional manifold space

In figure 4, we show the relationship between visual cues and clinical parameters, age, MMSE, and CDR. We can easily figure out the relationship between important visual cue(ventricle) and three clinical parameters. The larger ventricle usually means the older person, the higher chance that people had cognitive impairment(MMSE), and more serious stage of dementia(CDR).

Note that this is fully unsupervised learning approach. We didn't introduce any clinical parameters during the training process. The network learned everything from the only visual information.

4 Conclusion and Future Work

We applied recently proposed variational auto-encoder for brain MRI images. We showed promising results on manifold learning and its generative capability. Due to the time constraint, we haven't compared to existing methods and we leave it as future work. In addition, we could have done

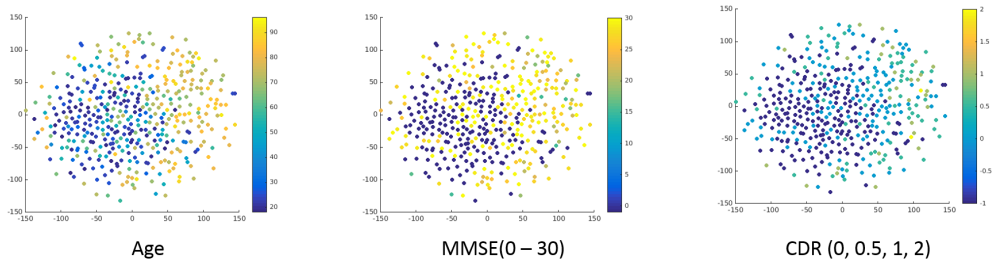


Figure 4: T-sne visualization of relationship between learned manifold and clinical parameters

classification task top of the learned manifold space. We can simple apply existing classifier or introduce clinical knowledge into the network during the training process[6].

References

[1] Torch. `http://torch.ch/`.

[2] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 2003.

[3] Yoshua Bengio, Li Yao, Guillaume Alain, and Pascal Vincent. Generalized denoising auto-encoders as generative models. In *Advances in Neural Information Processing Systems (NIPS)*. 2013.

[4] Samuel Gerber, Tolga Tasdizen, Thomas P Fletcher, and Ross Whitaker. Manifold modeling for brain population analysis. In *Proceedings of Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2009.

[5] Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 2006.

[6] Diederik P. Kingma, Danilo Jimenez Rezende, Shakir Mohamed, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems (NIPS)*. 2014.

[7] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

[8] Danilo J. Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, 2014.

[9] Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. Contracting auto-encoders: Explicit invariance during feature extraction. In *In Proceedings of the Twenty-eight International Conference on Machine Learning (ICML)*, 2011.

[10] Lawrence K. Saul and Sam T. Roweis. Think globally, fit locally: Unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research*, 2003.

[11] Pushmeet Kohli Joshua B. Tenenbaum Tejas D. Kulkarni, Will Whitney. Deep convolutional inverse graphics network. In *Advances in Neural Information Processing Systems (NIPS)*. 2015.

[12] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 2000.

[13] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5 - rmsprop, coursera: Neural networks for machine learning. 2012.

[14] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 2008.

[15] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *In Proceedings of the Twenty-eight International Conference on Machine Learning (ICML)*, 2008.