Autoencoders as approximations of markov chains

Scott Gigante
Advanced Topics in Data Mining and Machine Learning

January 31, 2018

1 Reading

Bengio, Yoshua, Yao, Li, Alain, Guillaume, and Vincent, Pascal. "Generalized denoising auto-encoders as generative models." *arXiv preprint arXiv: 1305.6663* (2013). URL `http://arxiv.org/abs/1305.6663`.

Dziugaite, Gintare Karolina, Daniel M. Roy, and Zoubin Ghahramani. "Training generative neural networks via maximum mean discrepancy optimization." *arXiv preprint arXiv:1505.03906* (2015). URL `http://arxiv.org/abs/1505.03906`.

2 Abstract

Recent work has shown denoising autoencoders to be capable of learning the manifold or latent space of a dataset. We will explore one model, proposed by Bengio et al., for representing an autoencoder as a generative network which learns a Markov process that walks towards peaks in density of the training data. We would like to further generalize this model to create a network which learns any kind of dynamic process. In order to do so, we use the maximum mean discrepancy (MMD) penalty (Gretton et al., 2012), a loss function which penalizes the network based on the difference between the distributions of output and target samples. This penalty is used in Dziugaite et al.'s MMDnet, a network which is capable of generating any target distribution from a random input distribution. Finally, we introduce the *transition encoder*, or *transcoder*, which applies the principle of MMDnet at each point in the input space in order to learn a dynamic process.

In "Generalized Denoising Auto-Encoders as Generative Models", Bengio et al. define a generic corruption process which is applied to the autoencoder's input data (for example, Gaussian noise), and the reconstruction process learned by the network which aims to reconstruct the original data from the corrupted data. The authors go on to define a stochastic process which consists of sampling many values from the corruption process, passing each of these through the network, and transitioning to the average of the network outputs. They prove that the limit of this process gives exactly the distribution of the training data and can be used as a pseudo-Gibbs sampling algorithm.

In order for a network to be capable of generating a chain of samples from any dynamic process, the network must be able to generate stochastic outputs from any input point without relying simply on the density of the input data. In "Training generative neural networks via maximum mean discrepancy optimization", Dziugaite et al. introduce the MMDnet, which uses the maximal mean discrepancy loss to construct a distribution. MMDnet is trained to generate random samples whose distribution matches the distribution of the training data, where the only input to the network is a random vector drawn from a uniform distribution.

We extend the two ideas above to introduce the transcoder, which learns the conditional distribution for a dynamic process at each point in the latent space of a dataset generated from the process of interest. In essence, the transcoder applies the MMDnet principle at each point in input space, giving it the ability to generate stochastic samples from a process which closely mimics the dynamic process that generated the training data. We demonstrate that the transcoder is able to learn both deterministic and stochastic processes, including harmonic and chaotic processes as well as Markov Chain Monte Carlo sampling.

3 Spotlight Question

What are some possible applications of neural networks capable of producing stochastic and/or dynamic output?