Deep generative models for biologists

Clayton W. Seitz

January 13, 2022

Outline

Deep Generative Models

Probabilistic Graphical Models

References

Discriminative and generative models

Say we have a set of variables $\mathbf{x} = (x_1, x_2, ..., x_n)$ which might have some statistical dependence

In supervised discriminative learning, we may use observations of $\mathbf{x}$ to try and learn distributions such as $p(x_2|x_1)$ (i.e., inference)

The variable $\mathbf{x}$ might be an amino acid sequence, DNA sequence, microscopy image, etc.

In supervised generative learning, we try to explicity learn the joint distribution $p(\mathbf{x}) = p(x_1|x_2, ..., x_n)p(x_2|x_3, ..., x_n), ..., p(x_n)$, which is generally more difficult.

The basic sampling problem

Suppose we are given a joint distribution

$$p(\mathbf{x}) = \frac{1}{Z}\tilde{p}(\mathbf{x})$$

where $p(\mathbf{x})$ is easy to compute but Z is (too) hard to compute.

This very important situation arises in several contexts:

1. In Bayesian models where $p(x_1, x_2) := p(x_1|x_2)p(x_2)$ is easy to compute but $Z = \int p(x_1|x_2)p(x_2)dx_2$ can be very difficult or impossible to compute.

2. In models from statistical physics, e.g. the Ising model, we only know $p(\mathbf{x}) = e^{-H(\mathbf{x})}$ where $H(\mathbf{x})$ is the Hamiltonian - the Ising model is an example of a Markov network or an undirected graphical model.

Approximating the joint distribution

Suppose we are given a joint distribution

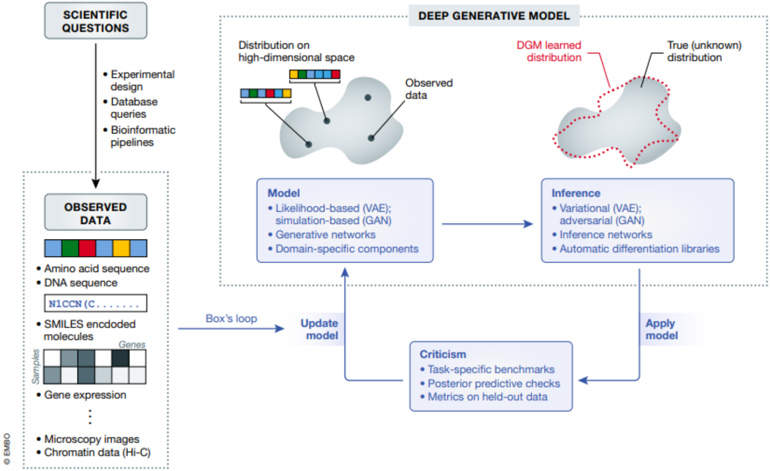
$$p(\mathbf{x}) = \frac{1}{Z}\tilde{p}(\mathbf{x})$$

Variational methods are generally useful for Bayesian inference like $p(x_1|x_2)$ but can also be used to evaluate $p(\mathbf{x})$ by autoencoding $\mathbf{x}$ (called a variational autoencoder)

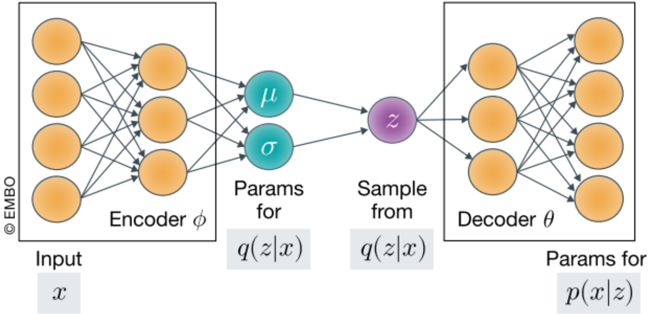
Generative adversarial networks (GANs) model $p(\mathbf{x})$ directly

In special scenarios, we may know $\tilde{p}(\mathbf{x})$ and we can use Monte-Carlo Markov Chain (MCMC) methods

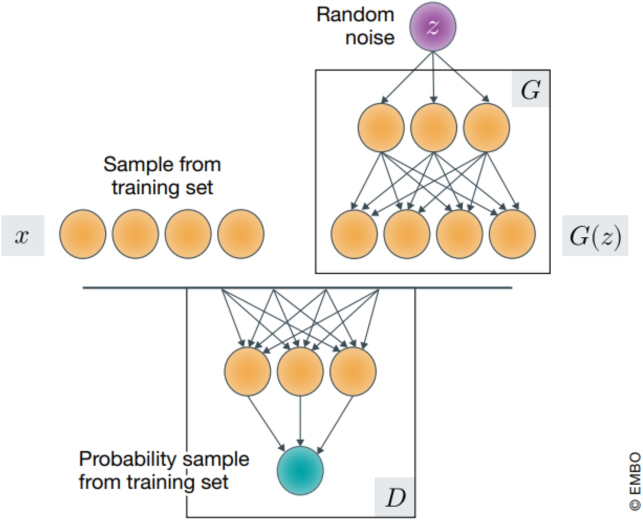
Applying generative models to biological data



Generative models: variational autoencoder



Generative models: adversarial networks



© EMBO

Cool biological applications of VAEs and GANs

Sequencing, Imaging, Other stuff

Monte-Carlo Markov Chain (MCMC)

- MCMC algorithms were originally developed in the 1940's by physicists at Los Alamos
- They were interested in modeling the probabilistic behavior of collections of atomic particles
- Simulation was difficult – the normalization constant Z was not known
- The term "Monte-Carlo" was coined at Los Alamos.
- Ulam and Metropolis overcame this problem by constructing a Markov chain for which the desired distribution was the stationary distribution
- Introduced to statistics and generalized with the Metropolis-Hastings algorithm (1970) and the Gibbs sampler of Geman and Geman (1984).

Monte-Carlo Markov Chain (MCMC)

MCMC is used when we know the functional form of $p(\mathbf{x})$ up to the normalization constant e.g., Ising model

MCMC methods do not model $p(\mathbf{x})$ directly but allow us to draw samples $\mathbf{x} \sim p(\mathbf{x})$

Gibbs sampling

Probabilistic graphical models

Using Gibbs sampling with graphical models

References I