Lecture #20: Variational Autoencoders

AM 207: Advanced Scientific Computing

Stochastic Methods for Data Analysis, Inference and Optimization

Fall, 2020

IACS

VE RI TAS

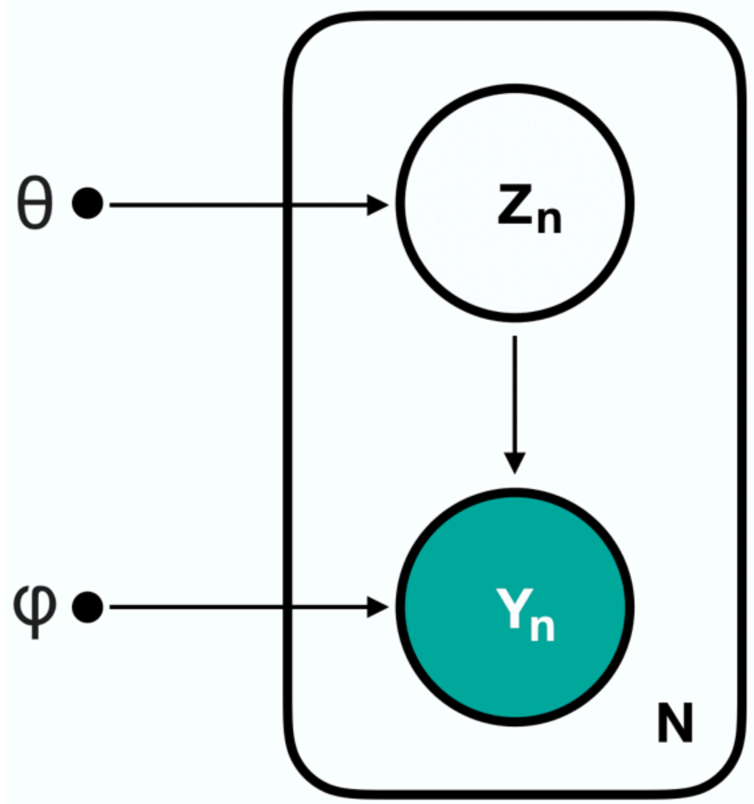
Outline

1. Applications of generative models
2. Inference for deep generative models: VAEs

Applications of generative models

Review of Latent Variable Models

Recall that models that include an observed variable Y and at least one unobserved variable Z are called ***latent variable models***. In general, our model can allow Y and Z to interact in many different ways. Earlier this semester, we studied models with one type of interaction:



$$Z_n \sim p(Z|\theta)$$
$$Y_n|Z_n \sim p(Y|Z, \phi)$$
$$n = 1, \ldots, N$$

Factor Analysis Models

In Lectures #9 and #10, we concentrated on Gaussian Mixutre Models. But in Lecture #9, we introduced ***factor analysis models***, where we posit that the observed data Y with many measurements is generated by a small set of unobserved factors Z:

$$
\begin{aligned}
Z_n &\sim \mathcal{N}(0, I),\\
Y_n | Z_n &\sim \mathcal{N}(f_{\mu,\Sigma}(Z_n), \Phi),
\end{aligned}
$$

where $f_{\mu,\Sigma}$ is a **linear function** of Z_n, in particular, $f_{\mu,\Sigma}(Z_n) = \mu + \Lambda Z_n; n = 1, \ldots, N$, $Z_n \in \mathbb{R}^{D'}$ and $Y_n \in \mathbb{R}^{D}$. We typically assume that D' is much smaller than D.

Applications

Factor analysis models are useful for biomedical data, where we typically measure a large number of characteristics of a patient (e.g. blood pressure, heart rate, etc), but these characteristics are all generated by a small list of health factors (e.g. diabetes, cancer, hypertension etc). Building a good model means we may be able to infer the list of health factors of a patient from their observed measurements.

Motivation for Generative Models

The factor analysis model:

$$
\begin{aligned}
Z_n &\sim \mathcal{N}(0, I),\\
Y_n | Z_n &\sim \mathcal{N}(f_{\mu,\Sigma}(Z_n), \Phi),
\end{aligned}
$$

where $f_{\mu,\Sigma}$ is a **linear function** of Z_n with parameters μ, Σ.

This is an example of a ***generative model***, that is, after learning the parameters μ, Σ of f, we can **generate** synthetic data by sampling $Z_n \sim \mathcal{N}(0, I)$ and then generating a synthetic observation by sampling $Y_n | Z_n \sim \mathcal{N}(f_{\mu,\Sigma}(Z_n), \Phi)$.

In many applications, like health care, where data collection is an financially and time-wise costly operation, synthetic data from a generative model can be extremely useful!

Generative models can also be used to perform imputation of missing data (i.e. fill-in missing covariates of observed data).

Motivation for Deep Generative Models

The problem with the factor analysis model is that marginal distribution of the data Y is a Gaussian distribution, since Y is the result of a Gaussian distribution $\mathcal{N}(0, I)$ transformed linearly by $f_{\mu,\Sigma}$. But in practice, the distribution of real data is complex! That is, we need **nonlinear transformations**.

A deep generative model can be defined as follows:

$$
\begin{aligned}
Z_n &\sim \mathcal{N}(0, I), \\
Y_n | Z_n &\sim \mathcal{N}(f_{\mathbf{W}}(Z_n), \Phi),
\end{aligned}
$$

where f is a **nonlinear function** of Z_n, parametrized by a **neural network** with weights $\mathbf{W}$. The deep generative model we will study today, ***variational autoencoders***, will have exactly this generative model.

Generating Data with Variational Autoencoders

For example, if we train a deep generative model (a VAE) to capture the distribution of a set of image data consisting of celebrity faces, we can use this generative model to generate realistic looking synthetic faces:

Inference for deep generative models: VAEs

Expectation Maximization: Estimating the MLE for Latent Variable Models

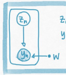
Given a deep latent variable model $p(Y, Z|\mathbf{W}) = p(Y|f_{\mathbf{W}}(Z))p(Z)$, we are interested computing the MLE of parameters $\mathbf{W}$, i.e. the parameters that maximizes the likelihood of the observed data:

$$\begin{aligned}
\mathbf{W}_{\text{MLE}} &= \underset{\mathbf{W}}{\text{argmax}}\ \ell(\mathbf{W}) \\
&= \underset{\mathbf{W}}{\text{argmax}}\ \log \prod_{n=1}^{N} \int_{\Omega_Z} p(y_n, z_n|\mathbf{W}) dz \\
&= \underset{\mathbf{W}}{\text{argmax}}\ \log \prod_{n=1}^{N} \int_{\Omega_Z} p(y_n|f_{\mathbf{W}}(z_n))p(z_n) dz
\end{aligned}$$

where Ω_Z is the domain of Z.

In the case of factor analysis models and mixture of Gaussian models, we used Expectation Maximization in order to maximize a lower bound of the observed data log-likelihood, the ELBO. How will this work for deep latent variable models?

MAXIMUM LIKELIHOOD ESTIMATION FOR DEEP GENERATIVE MODELS



$z_n \sim p(z_n)$
$y_n \sim p_W(y_n|z_n)$

FOR EXAMPLE: $z_n \sim N(0, I)$, $z_n \in \mathbb{R}^{D'}$
$y_n \sim N(f_W(z_n), \sigma^2 I)$, $y_n \in \mathbb{R}^{D}$

ALTERNATIVELY: $z_n \sim N(0, I)$,
$y_n = f_W(z_n) + \varepsilon$, $\varepsilon \sim N(0, \sigma^2 I)$

GOAL: 1. LEARN W_{MLE} SO WE CAN MODEL THE OBSERVED DATA DISTRIBUTION

$$\prod_{n=1}^{N} p_{W_{MLE}}(y_n) = \prod_{n=1}^{N} \int_{\Omega_z} p_{W_{MLE}}(y_n|z_n) p(z_n)\, dz_n$$

2. LEARN THE LATENT CODES z_n

SINCE WE DON'T OBSERVE z_n, WE CAN ONLY MAXIMIZE OBSERVED DATA LOG LIKELIHOOD

$$\begin{aligned} W_{MLE} &= \underset{W}{\operatorname{argmax}} \log \prod_{n=1}^{N} p_W(y_n) \\ &= \underset{W}{\operatorname{argmax}} \sum_{n=1}^{N} \log \int_{\Omega_z} p_W(y_n|z_n) p(z_n)\, dz_n \\ &= \underset{W}{\operatorname{argmax}} \sum_{n=1}^{N} \log \mathbb{E}_{p(z_n)} \left[p_W(y_n|z_n) \right] \end{aligned}$$

THE PROBLEM: $\nabla_W \sum_n \log \mathbb{E}_{p(z_n)} [p_W(y_n|z_n)] = \sum_n \frac{\nabla_W \mathbb{E}_{p(z_n)}[p_W(y_n|z_n)]}{\mathbb{E}_{p(z_n)}[p_W(y_n|z_n)]}$ } MC ESTIMATE HIGH VARIANCE

MAXIMIZING THE ELBO INSTEAD:

$$\underbrace{\sum_{n=1}^{N} \log \mathbb{E}_{p(z_n)} \left[p_W(y_n|z_n) \right]}_{\text{OBSERVED DATA LOG LIKELIHOOD}} = \sum_{n=1}^{N} \log \int \underbrace{\frac{p(z_n)}{q(z_n)}}_{\text{IMPORTANCE WEIGHT}} p_W(y_n|z_n) \underbrace{q(z_n)}_{\text{IMPORTANCE DISTRIBUTION}} dz_n \quad \text{(IMPORTANCE SAMPLING)}$$

$$= \sum_{n=1}^{N} \log \mathbb{E}_{q(z_n)} \left[\frac{p_W(y_n|z_n) p(z_n)}{q(z_n)} \right]$$

$$\geq \underbrace{\sum_{n=1}^{N} \mathbb{E}_{q(z_n)} \left[\log \frac{p_W(y_n|z_n) p(z_n)}{q(z_n)} \right]}_{\text{ELBO}(W, q)}$$

JENSEN'S INEQUALITY

$$\underbrace{\max_W \log \prod_{n=1}^{N} p_W(y_n)}_{\text{OBSERVED DATA LOG-LIKELIHOOD}} \geq \max_{W, q} \text{ELBO}(W, q)$$

INSTEAD OF MAXIMIZING THE OBSERVED DATA LOG-LIKELIHOOD, WE MAXIMIZE THE ELBO

MAXIMIZING THE ELBO:

IN EXPECTATION MAXIMIZATION:

M STEP: $\max_W \text{ELBO}(W, q^*)$ CAN BE DONE BY GRADIENT DESCENT

$$\nabla_W \sum_{n=1}^{N} \mathbb{E}_{q(z_n)} \left[\log \frac{p_W(y_n|z_n) p(z_n)}{q(z_n)} \right] = \sum_{n=1}^{N} \underbrace{\mathbb{E}_{q(z_n)}}_{\text{MC ESTIMATE}} \left[\underbrace{\nabla_W \log \frac{p_W(y_n|z_n) p(z_n)}{q(z_n)}}_{\text{AUTOGRAD}} \right]$$

E STEP: $\max_q \text{ELBO}(W^*, q)$ HAS A SOLUTION (LECTURE #9)

$$q^* = \underset{q}{\operatorname{argmax}}\, \text{ELBO}(W^*, q) = \underbrace{p_{W^*}(z_n|y_n)}_{\text{posterior}}$$

THE PROBLEM: IF $y_n = \text{nn.forward}(z_n, W^*) + \varepsilon$, $p_{W^*}(z_n|y_n)$ IS INTRACTABLE TO COMPUTE IN CLOSED FORM, UNLIKE THE CASE OF GAUSSIAN MIXTURES AND FACTOR ANALYSIS.

INNOVATION #1: VARIATIONAL INFERENCE FOR POSTERIOR $p_W(z_n|y_n)$

CHOOSE A MEAN-FIELD VARIATIONAL FAMILY $q(z_n) = N(z_n; \mu_n, \Sigma_n)$, Σ_n DIAGONAL

IN THE E-STEP: $\mu_n^*, \Sigma_n^* = \underset{\mu_n, \Sigma_n}{\operatorname{argmin}} D_{KL}[q_{\mu_n,\Sigma_n}(z_n) \,\|\, p_{W^*}(z_n|y_n)]$

$$\equiv \underset{\mu_n, \Sigma_n}{\operatorname{argmax}} \underbrace{\mathbb{E}_{q_{\mu_n,\Sigma_n}(z_n)} \left[\log \frac{p_{W^*}(y_n|z_n) p(z_n)}{q_{\mu_n, \Sigma_n}(z_n)} \right]}_{\text{ELBO}(W^*, q)}$$

THE PROBLEM: IF N IS HUGE THEN WE NEED TO RUN VI A HUGE NUMBER OF TIMES! COMPUTATIONALLY INTRACTABLE!

INNOVATION #2: AMORTIZED INFERENCE

IDEA: IF WE SAW THAT $y_n = 1$ AND $\mu_n = 1.01$, $\sigma^2 = 0.5$
THEN IF $y_m \approx y_n$, WE WOULD EXPECT $p(z_n|y_n) \approx p(z_m|y_m)$,
HENCE $\sigma_m^2 \approx \sigma_n^2$, $\mu_m \approx \mu_n$.

THAT IS, WE SHOULD BE ABLE TO PREDICT μ_n, Σ_n of $p(z_n|y_n)$ USING y_n, I.E. WE LEARN A FUNCTION $g_{V^*}(y_n) = \mu_{V^*}(y_n), \Sigma_{V^*}(y_n)$ s.t.

$$V^* = \underset{V}{\operatorname{argmin}} \sum_n D_{KL} [\underbrace{N(\mu_V(y_n), \Sigma_V(y_n))}_{q_V(z_n)} \,\|\, p_{W^*}(z_n|y_n)]$$

$$\equiv \underset{V}{\operatorname{argmax}} \underbrace{\sum_n \mathbb{E}_{q_V(z_n)} \left[\frac{p_{W^*}(z_n, y_n)}{q_V(z_n)} \right]}_{\text{ELBO}(W^*, V)}$$

GRADIENT DESCENT: MAXIMIZING THE ELBO WITH RESPECT TO V

$$\nabla_V \text{ELBO}(W^*, V) = \nabla_V \sum_n \mathbb{E}_{q_V(z_n)} \left[\frac{p_{W^*}(z_n, y_n)}{q_V(z_n)} \right]$$

$$= \sum_n \nabla_V \mathbb{E}_{\varepsilon \sim N(0,I)} \left[\frac{p_{W^*}(\varepsilon \Sigma_V^{1/2}(y_n) + \mu_V(y_n), y_n)}{q_V(\varepsilon \Sigma_V^{1/2}(y_n) + \mu_V(y_n))} \right]$$

$$= \sum_n \mathbb{E}_{\varepsilon \sim N(0,I)} \left[\underbrace{\nabla_V \frac{p_{W^*}(\varepsilon \Sigma_V^{1/2}(y_n) + \mu_V(y_n), y_n)}{q_V(\varepsilon \Sigma_V^{1/2}(y_n) + \mu_V(y_n))}}_{\text{AUTOGRAD}} \right]$$

INNOVATION #3: JOINT TRAINING

RATHER THAN COORDINATE ASCENT OVER W AND V ITERATIVELY, WE PERFORM GRADIENT DESCENT OVER W & V SIMULTANEOUSLY!

$$W^*, V^* = \underset{W,V}{\operatorname{argmax}}\, \text{ELBO}(W, q_V)$$

GRADIENT:

$$\nabla_{W,V} \text{ELBO}(W, q_V) = \nabla_{W,V} \sum_n \mathbb{E}_{\varepsilon \sim N(0,I)} \left[\frac{p_W(\varepsilon \Sigma_V^{1/2}(y_n) + \mu_V(y_n), y_n)}{q_V(\varepsilon \Sigma_V^{1/2}(y_n) + \mu_V(y_n))} \right]$$

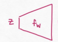
$$= \sum_n \mathbb{E}_{\varepsilon \sim N(0,I)} \left[\underbrace{\nabla_{W,V} \frac{p_W(\varepsilon \Sigma_V^{1/2}(y_n) + \mu_V(y_n), y_n)}{q_V(\varepsilon \Sigma_V^{1/2}(y_n) + \mu_V(y_n))}}_{\text{AUTOGRAD}} \right]$$

VARIATIONAL AUTOENCODER (VAE)

A VARIATIONAL AUTOENCODER IS BOTH A MODEL AND AN INFERENCE METHOD:

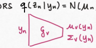
GENERATIVE MODEL:

$z_n \sim p(z_n) = N(0, I)$
$y_n = f_W(z_n) + \varepsilon$, $\varepsilon \sim N(0, \sigma^2 I)$



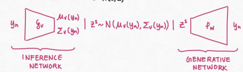
INFERENCE MODEL: FOR POSTERIORS $q(z_n|y_n) = N(\mu_n, \Sigma_n) \approx p(z_n|y_n)$

$g_V(y) = \mu_V(y_n), \Sigma_V(y_n)$



TRAINING OBJECTIVE: $\max_{W,V} \text{ELBO}(W, q_V)$

$$W^*, V^* = \underset{W,V}{\operatorname{argmax}} \sum_n \mathbb{E}_{\varepsilon \sim N(0,I)} \left[\frac{p_W(\varepsilon \Sigma_V^{1/2}(y_n) + \mu_V(y_n), y_n)}{q_V(\varepsilon \Sigma_V^{1/2}(y_n) + \mu_V(y_n))} \right]$$



DIFFERENT PRESENTATIONS OF THE ELBO:

I. JOINT PLUS ENTROPY

$$\text{ELBO}(W, q_V) = \sum_n \mathbb{E}_{q_V} \left[\log \frac{p_W(y_n, z_n)}{q_V(z_n)} \right] = \sum_n \mathbb{E}_{q_V} \log p_W(y_n, z_n) - \mathbb{E}_{q_V}[\log q_V]$$

$$= \sum_n \Big[\mathbb{E}_{q_V} \log p_W(y_n, z_n) + \underbrace{\mathbb{H}[q_V]}_{\text{ENTROPY OF } q_V} \Big]$$

II. EXPECTED LIKELIHOOD MINUS KL

$$\text{ELBO}(W, q_V) = \sum_n \mathbb{E}_{q_V} \left[\log \frac{p_W(y_n, z_n)}{q_V(z_n)} \right] = \sum_n \mathbb{E}_{q_V} \left[\log \frac{p_W(y_n|z_n) p(z_n)}{q_V(z_n)} \right]$$

$$= \sum_n \left[\mathbb{E}_{q_V} [\log p_W(y_n|z_n)] - \mathbb{E}_{q_V} \left[\log \frac{q_V(z_n)}{p(z_n)} \right] \right]$$

$$= \sum_n \left[\mathbb{E}_{q_V} [\log p_W(y_n|z_n)] - D_{KL}[q_\phi(z_n) \,\|\, p(z_n)] \right]$$

III. OBSERVED DATA LOG-LIKELIHOOD MINUS KL

$$\text{ELBO}(W, q_V) = \sum_n \mathbb{E}_{q_V} \left[\log \frac{p_W(y_n, z_n)}{q_V(z_n)} \right] = \sum_n \mathbb{E}_{q_V} \left[\log \frac{p_W(y_n, z_n)\, p_W(y_n)}{q_V(z_n)\, p_W(y_n)} \right]$$

$$= \sum_n \mathbb{E}_{q_V} \left[\log \frac{p_W(z_n|y_n)}{q_V(z_n)} \cdot p_W(y_n) \right]$$

$$= \sum_n \left[\mathbb{E}_{q_V} \log p_W(y_n) - \mathbb{E}_{q_V} \log \frac{q_V(z_n)}{p_W(z_n|y_n)} \right]$$

$$= \sum_n \left[\log p_W(y_n) - D_{KL}[q_V(z_n) \,\|\, p_W(z_n|y_n)] \right]$$

Generating Data with Variational Autoencoders

For example, if we train a deep generative model (a VAE) to capture the distribution of a set of image data consisting of celebrity faces, we can use this generative model to generate realistic looking synthetic faces: