TTIC 31230, Fundamentals of Deep Learning

David McAllester, Winter 2020

Interpretable Latent Variables

Latent Variables

$$P_\Phi(y) = \sum_z P_\Phi(z)P_\Phi(y|z) = E_{z\sim P_\Phi(z)}\; P_\Phi(y|z)$$

Or

$$P_\Phi(y|x) = \sum_z P_\Phi(z|x)P_\Phi(y|z,x) = E_{z\sim P_\Phi(z|x)}\; P_\Phi(y|z,x)$$

Here z is a latent variable.

Interpretable Latent Variables

$$P_\Phi(y) = \sum_z P_\Phi(z)P_\Phi(y|z) = E_{z \sim P_\Phi(z)} \; P_\Phi(y|z)$$

Here we often think of z as the causal source of y.

For example z might be a physical scene causing image y.

Or z might be the intended utterance causing speech signal y.

In these situations a latent variable model should more accurately represent the distribution on y.

Interpretable Latent Variables

$$P_\Phi(y) = \sum_z P_\Phi(z) P_\Phi(y|z) = E_{z \sim P_\Phi(z)}\, P_\Phi(y|z)$$

$P_\Phi(z)$ is called the prior.

Given an observation of y (the evidence) $P_\Phi(z|y)$ is called the posterior.

Variational Bayesian inference involves approximating the posterior.

Colorization with Latent Segmentation

x $\qquad\qquad$ $\hat{y}$ $\qquad\qquad$ y

Larsson et al., 2016

Colorization is a natural self-supervised learning problem — we delete the color and then try to recover it from the grey-level image.

Can colorization be used to learn segmentation?

Segmentation is latent — not determined by the color label.

Colorization with Latent Segmentation

x $\hat{y}$ y

Larsson et al., 2016

x is a grey level image.

y is a color image drawn from $\mathrm{Pop}(y|x)$.

$\hat{y}$ is an arbitrary color image.

$P_\Phi(\hat{y}|x)$ is the probability that model Φ assigns to the color image $\hat{y}$ given grey level image x.

Colorization with Latent Segmentation

x $\hat{y}$ y

$$P_\Phi(\hat{y}|x) = \sum_z P_\Phi(z|x)P_\Phi(\hat{y}|z,x).$$

$$\begin{aligned} &\text{input } x \\ P_\Phi(z|x) &= \ldots \quad \text{semantic segmentation} \\ P_\Phi(\hat{y}|z,x) &= \ldots \quad \text{segment colorization} \end{aligned}$$

Assumptions

We assume models $P_\Phi(z)$ and $P_\Phi(y|z)$ are both samplable and computable.

In other words, we can sample from these distributions and for any given z and y we can compute $P_\Phi(z)$ and $P_\Phi(y|z)$.

These are nontrivial assumptions.

A loopy graphical model is neither (efficiently) samplable nor computable.

Cases Where the Assumptions Hold

In CTC we have that z is the sequence with blanks and y is the result of removing the blanks from z.

In a hidden markov model z is the sequence of hidden states and y is the sequence of emissions.

An autoregressive model, such as an autoregressive language model, is both samplable and computable.

Image Generators

z $\qquad\qquad$ $y_\Phi(z)$

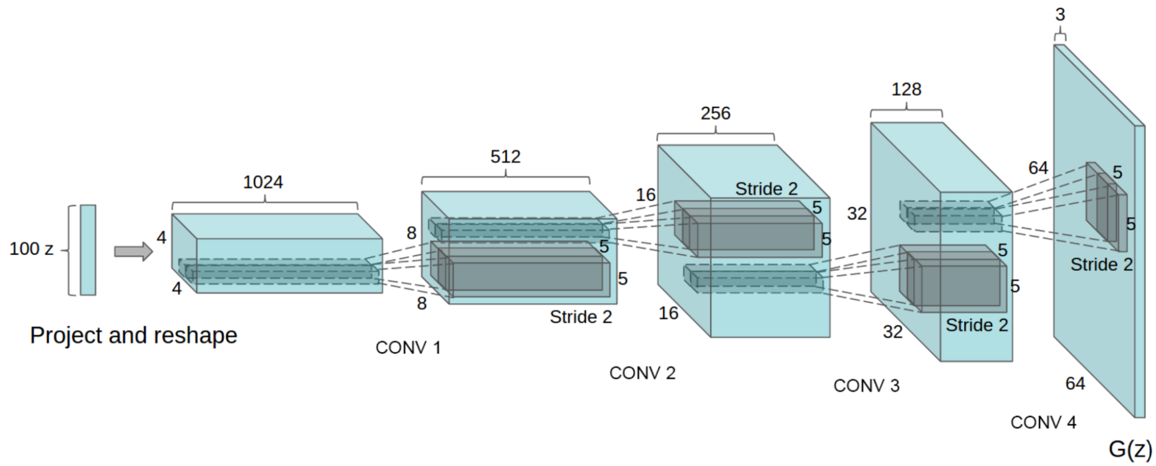

Figure 1: DCGAN generator used for LSUN scene modeling. A 100 dimensional uniform distribution Z is projected to a small spatial extent convolutional representation with many feature maps.

We can generate an image y form noise z where $p_\Phi(z)$ and $p_\Phi(y|z)$ are both samplable and computable.

Typically $p_\Phi(z)$ is $\mathcal{N}(0, I)$ reshaped as $z[X, Y, J]$

Image Generators

z $y_\Phi(z)$

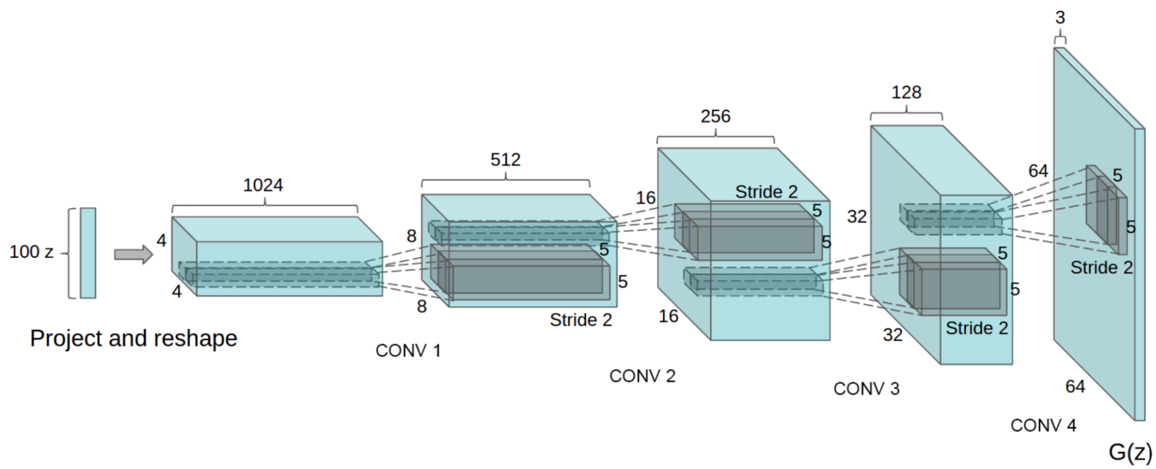

Figure 1: DCGAN generator used for LSUN scene modeling. A 100 dimensional uniform distribution Z is projected to a small spatial extent convolutional representation with many feature maps.

Our assumptions hold for image generators such as GANs, but z is typically viewed as “noise” and is not interpretable.

Modeling y

We would like to use the fundamental equation

$$\Phi^* = \underset{\Phi}{\mathrm{argmin}}\;\; E_{y \sim \mathrm{Pop}}\;\; -\ln P_\Phi(y)$$

But even when $P_\Phi(z)$ and $P_\Phi(y|z)$ are samplable and computable we cannot typically compute $P_\Phi(y)$.

Specifically, for $P_\Phi(y)$ defined by a generator we cannot compute $P_\Phi(y)$ for a test image y.

END