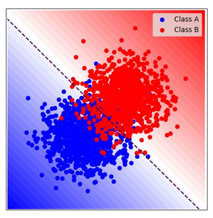
Do Deep Generative Models Know What They Don't Know?

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, Balaji Lakshminarayanan

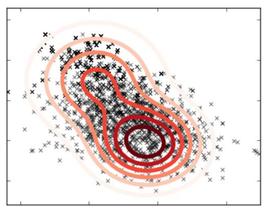
DeepMind

1. Generative Modeling



$p(\mathbf{y}|\mathbf{x})$

Discriminative Model



$p(\mathbf{x})$

Generative Model

Discriminative models are susceptible to overconfident mistakes on out-of-distribution (OOD) inputs. Generative models are widely believed to be more robust as they model not only $p(\mathbf{y}|\mathbf{x})$ but also $p(\mathbf{x}_{\text{TRAIN}})$. If an incoming sample has a low probability under $p(\mathbf{x})$, then it can be flagged as OOD and an "I don't know" message can be returned to the user [Bishop, 1994].

We challenge this assumption, showing that **deep generative models can at times assign higher density estimates to an OOD dataset than to the training data**. This phenomenon has implications not just for anomaly detection but any application for which we evaluate the density model on inputs unlike the training data: detecting covariate shift, open-set classification, active learning, semi-supervised learning, and others.

2. Motivating Observations

We trained **Glow** [Kingma & Dhariwal, 2018], a flow-based generative model, on CIFAR-10 and evaluated on the CIFAR-10 test split and on SVHN, a set of house numbers.

CIFAR-10

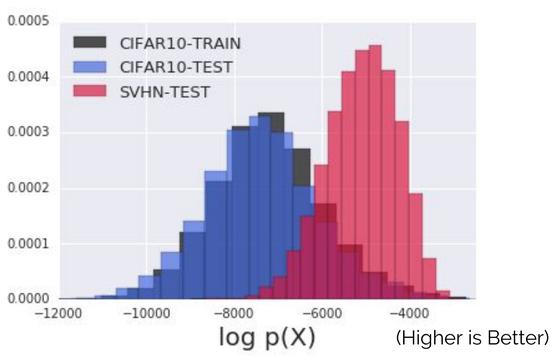
SVHN

We find that **Glow assigns a higher likelihood to SVHN than to CIFAR-10** (both train/test splits).

(Lower is Better)

Data Set	Avg. Bits Per Dimension
Glow Trained on CIFAR10	
CIFAR10-Train	3.386
CIFAR10-Test	3.464
SVHN-Test	**2.389**
Glow Trained on SVHN	
SVHN-Test	2.057

This phenomenon is persistent across SVHN images and is not due to a few outliers.

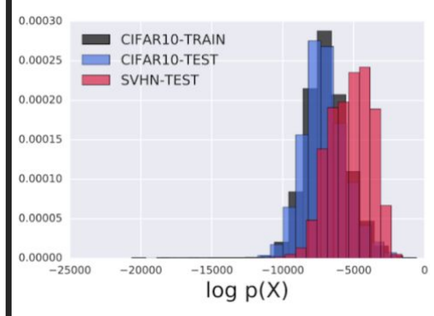


Histogram of Glow's Log-Likelihoods

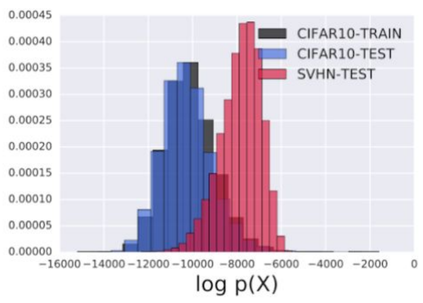
Constant (128) images are assigned a higher likelihood still: 0.6 bits-per-dimension.

3. Other Generative Models and Image Data Sets

We observe the phenomenon for **autoregressive (PixelCNN)** and **latent variable (VAE)** models as well.

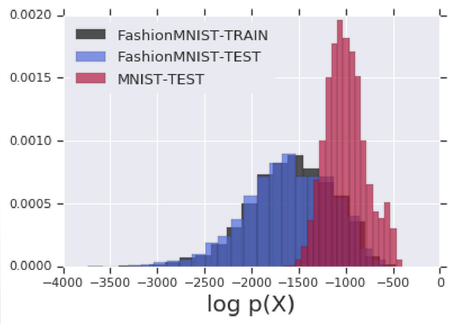


PixelCNN

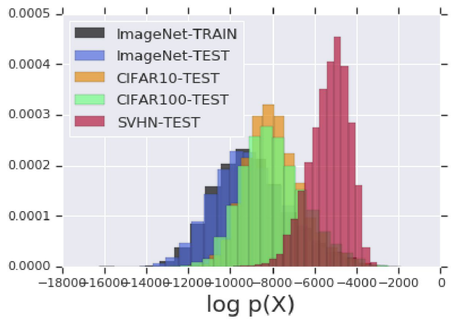


Variational Autoencoder

We observe the phenomenon for **FashionMNIST** & **ImageNet**.



Train Glow on FashionMNIST, Test on MNIST



Train Glow on ImageNet, Test on CIFAR / SVHN

4. Digging Deeper Into The Phenomenon For Flow-Based Models

The log-likelihood of flow-based models can be decomposed into **prior** and **volume** terms:

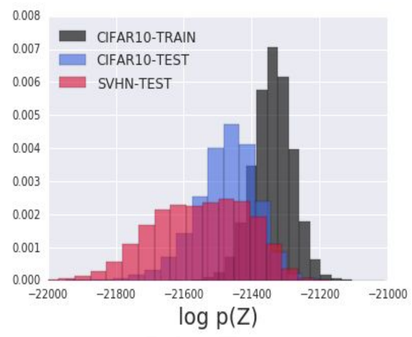
$$\log p_x(\boldsymbol{x};\boldsymbol{\theta}) = \underbrace{\log p_z(\boldsymbol{f}_\phi(\boldsymbol{x});\boldsymbol{\psi})}_{\text{prior term}} + \underbrace{\log\left|\frac{\partial \boldsymbol{f}_\phi}{\partial \boldsymbol{x}}\right|}_{\text{volume term}}$$

$\boldsymbol{f}_\phi$ denotes invertible mapping, $\boldsymbol{\psi}$ parametrizes prior p_z

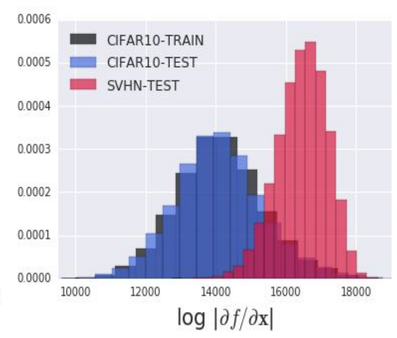
We noticed that the volume term was higher for out-of-distribution inputs (*see right*). **Maximizing the log-det.-Jacobian contradicts regularization literature** (e.g. contractive AEs), increasing sensitivity in $\boldsymbol{x}$, so we initially suspected this was the issue.

To control for the volume term, we formulated **constant-volume (CV) Glow**, which uses only additive transformations in its coupling layers. This model is non-volume-preserving, but the volume term is constant across all $\boldsymbol{x}$.

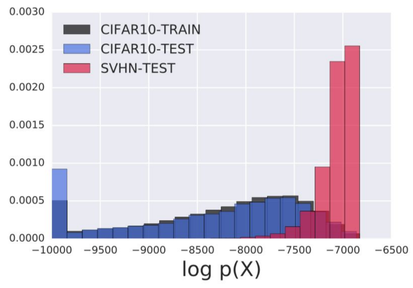
CV-Glow's log-likelihoods when trained on CIFAR-10 and tested on SVHN is shown to the right. We see that SVHN's higher likelihood persists. Bits-per-dimension (BDP) over the course of training is also shown to the right. There is no cross-over during optimization..



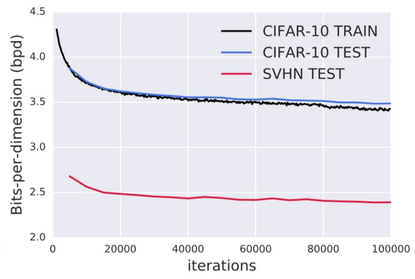
Prior Term
(CIFAR-10 vs SVHN)



Volume Term
(CIFAR-10 vs SVHN)



Constant-Volume Glow: CIFAR-10 vs SVHN Log-Likelihoods



Constant-Volume Glow: BPD During Training

5. Second-Order Analysis and Gray-Attack

Can we say something general about when we expect some other distribution q to have higher likelihood than the training distribution p*?

$$0 < \mathbb{E}_q[\log p(\boldsymbol{x};\boldsymbol{\theta})] - \mathbb{E}_{p^*}[\log p(\boldsymbol{x};\boldsymbol{\theta})]$$

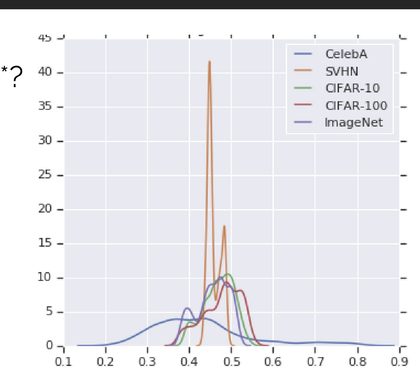
For flow-based models, we can approximate this difference with the following second-order expression:

$$\approx \frac{1}{2}\operatorname{Tr}\left\{\left[\nabla^2_{\boldsymbol{x}_0}\log p_z(f(\boldsymbol{x}_0;\boldsymbol{\phi})) + \nabla^2_{\boldsymbol{x}_0}\log\left|\frac{\partial \boldsymbol{f}_\phi}{\partial \boldsymbol{x}_0}\right|\right](\boldsymbol{\Sigma}_q - \boldsymbol{\Sigma}_{p^*})\right\}$$

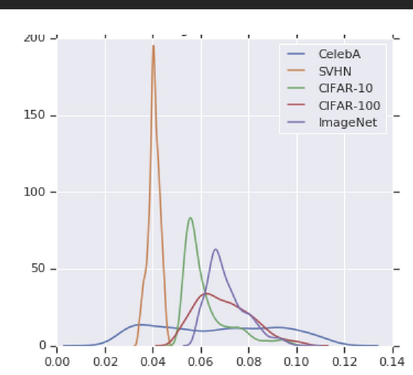
For **CV-Glow** in particular, the volume term drops out (since it does not depend on $\boldsymbol{x}$), yielding:

$$= \frac{\partial^2}{\partial z^2}\log p(\boldsymbol{z};\boldsymbol{\psi})\sum_{c=1}^{C}\left(\prod_{k=1}^{K}\sum_{j=1}^{C}u_{k,c,j}\right)^2\sum_{h,w}(\sigma^2_{q,h,w,c} - \sigma^2_{p^*,h,w,c})$$

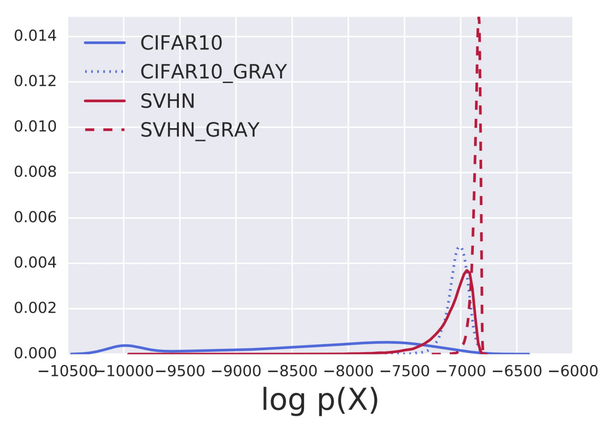
where $\boldsymbol{u}$ denotes the parameters of the 1x1 conv. kernel, σ^2 is the variance of each distribution, C is the number of channels, and K is the number of flow steps. Plugging in the empirical variances (*see right, top*) results in the quantity being **greater than zero, meaning SVHN will always have a higher likelihood under CV-Glow**. This agrees with our plot of BDP during training shown above. Furthermore, this means that **we can increase the likelihoods by lowering the input variance, which is equivalent to graying natural images**. See figure to right.



Histogram of Empirical Means



Histogram of Empirical Variances



Increasing Likelihood by Graying Images

Summary

- Density estimates from (current) deep generative models are **not always able to detect out-of-distribution inputs**.
- We should temper the enthusiasm with which we preach the benefits of generative models until their sensitivity to out-of-distribution inputs is better understood.

ArXiv Version: https://arxiv.org/abs/1810.09136

Contact: e.nalisnick@eng.cam.ac.uk, {amatsukawa, ywteh, dilang, balajiln}@google.com