Sparse Image Generation with Decoupled Generative Models

Yadong Lu
Department of Computer Science
University of California Irvine

Julian Collado
Department of Computer Science
University of California Irvine

Kevin Bauer
Department of Physics and Astronomy
University of California Irvine

Daniel Whiteson
Department of Physics and Astronomy
University of California Irvine

Pierre Baldi
Department of Computer Science
University of California Irvine

Abstract

Learning to generate very sparse images with deep generative models has been a challenging problem due to issues such as mode collapse and sparse gradient signals. In this work, we propose a novel model combining a neural network generator with an explicit mixture model, which induces sparsity with a learnable Dirac delta mass at zero instead of using rectified linear units at the output. Our model decouples the sparsity level and the non-zero distribution of each pixel in the data while fitting both of them simultaneously by minimizing the model's entropy. We demonstrate both theoretically and empirically that the model is able to learn a rich distribution of a sparse muon image dataset while maintaining desired physical properties such as isolation.

1 Introduction

Detailed simulations are increasingly being used in science and engineering to study complex processes. These simulations can be very slow and computationally expensive, so it is common to use approximations to make them faster. As an alternative many generative models based on Variational Auto-encoder (VAE) and Generative Adversarial Network (GAN) have been proposed for generating data with a similar distribution as the one obtained from experiments or created by simulations[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]. Some of these events can be represented as images, and in the case of particle physics they are usually very sparse images.

In particular, GAN based generative models for very sparse images synthesis in particle physics have been studied in detail by Oliveira et al. with their proposed LAGAN [12] and CALOGAN [13] models. Both of these models rely on the ReLU activation function in the final layer to induce sparsity in the output image. However, the gradient with respect to the input for the ReLU activation is zero when the input is negative, which potentially leads to a sparse gradient signal issue noted by [12]. In addition, GANs have been known to suffer from unstable training and mode collapse [14].

Our model addresses these issues by decoupling the sparsity level estimation and the non-zero pixel value distribution estimation using an explicit probabilistic model. Further, we use a neural network generator to allow for greater flexibility in the generated distribution. In a concurrent work [15], it has been shown that the neural network generator helps to enrich the output distribution. We derived

Second Workshop on Machine Learning and the Physical Sciences (NeurIPS 2019), Vancouver, Canada.

the sparse cross entropy loss based on our model and show that our objective function corresponds to a lower bound of the cross entropy between the data distribution and the generated image distribution. With the decoupling approach, we are able to model the non-zero distribution separately from the sparsity level. We empirically demonstrate its benefit in learning the correct sparsity level and it can potentially lead to faster training convergence for the model.

2 Dataset

Isolated muons were generated via the process $pp \rightarrow Z' \rightarrow \mu^+\mu^-$ with a Z' mass of 20 GeV; muons generated from this boson decay should be representative of muons from any boson decay. Non-isolated muons were generated via the process $pp \rightarrow b\bar{b}$. Both samples are generated at a center of mass energy $\sqrt{s} = 13$ TeV. Collisions and immediate decays are simulated with MADGRAPH5 [16], showered and hadronized with PYTHIA [17], and the detector response simulated with delphes [18]. The classification of these objects is senstitive to the presence of additional proton interactions, referred to as pile-up events. We overlay such interactions within the simulation with an average number of interactions per event of $\mu = 50$, as a future estimate of LHC experimental data. 100,000 signal and background events are generated.

Muons in the range $p_{\mathrm{T}} \in [10, 15]$ GeV are considered, and the signal samples are weighted such that the transverse muon momentum distributions match that of the background. Only events where a muon is identified as a track in the muon spectrometer are used.

Muon images are built from the calorimeter deposits. We preprocess the calorimeter deposit images by centering the image on the coordinates of the identified muon and only include calorimeter deposits within a $\eta - \phi$ radius of $R < 0.4$ in order to only analyze the activity near the muon and reduce topological sensitivity. We choose a 32x32 grid, which roughly corresponds with the calorimeter granularity of ATLAS and CMS.

Heat maps of the calorimeter energy deposits in $\eta - \phi$ space for both signal isolated muons and background non-isolated muons are shown in Fig. 3. The signal calorimeter deposits are uniform and can be attributed to pileup whereas the background deposits are largely radially symmetric with a dense core from the jet. The dataset is very sparse, 98.83% of the pixels in the data are zero.

3 Decoupled Generative Models

The model has two components as shown in Figure 1. The first component is a neural network which transforms the distribution of an input noise vector into distribution parameters $\boldsymbol{\eta}$ and $\boldsymbol{\pi}$ in order to enrich the mixture distribution in the second component, and account for dependency among all pixels of the explicit model. The second component is a mixture distribution of dirac delta mass π_i at zero, and a non-zero tractable probability density $p_{\eta_i}(x_i)$. $\boldsymbol{\pi}$ then corresponds to the mean parameter of the Bernoulli vector which controls the sparsity level of the images while $\boldsymbol{\eta}$ captures the non-zero pixel intensity for the ith pixel. After sampling a Bernoulli variable from $\boldsymbol{z} \sim Bern(\boldsymbol{\pi})$, together with a random variable $\boldsymbol{\beta} \sim p_{\boldsymbol{\eta}}(\boldsymbol{\beta})$. we use the element-wise product of those two as the output $\boldsymbol{x} = \boldsymbol{z} \odot \boldsymbol{\beta}$. The complete generating process is shown in Figure 1.

The distribution of the output generated by the model can be seen as a compound distribution of an explicit mixture model with its distribution parameters induced by a neural network generator. Below, we demonstrate how to optimize the distribution $p_{\boldsymbol{\phi_1}}(\boldsymbol{\beta})$ and $p_{\boldsymbol{\phi_2}}(\boldsymbol{\pi})$, where $\boldsymbol{\phi_1}$ and $\boldsymbol{\phi_2}$ are the generator parameters for $\boldsymbol{\beta}$ and $\boldsymbol{\pi}$ respectively.

Sparse Cross Entropy Loss A natural choice of the training objective is to maximize the cross entropy $\mathcal{H}(p_D(\boldsymbol{x}), \boldsymbol{x})$ and the generated distribution $p_{\boldsymbol{\theta}}(\boldsymbol{x})$, where $\boldsymbol{\theta} = \{\boldsymbol{\pi}, \boldsymbol{\eta}\}$. However, since we want to decouple the sparsity level and the non-zero distribution estimation by fitting $p(x_i|z_i = 0)$ and $p(x_i|z_i = 1)$, we show it is more straightforward to consider the joint distribution $p_{\boldsymbol{\eta},\boldsymbol{\pi}}(\boldsymbol{x}, \boldsymbol{z})$. Suppose the output $\boldsymbol{x} \in R^M$, and we assume conditional independence of $p(x_i|\pi_i, \eta_i)$ among the pixels. We parameterize the conditional density of $p(x_i|z_i = 1) = p_{\eta_i}(x_i)$, where it can be any explicit density function with parameter η_i, and $p(x_i|z_i = 0) = \delta_0(x_i)$, a dirac delta mass at zero. In the following proposition, we derive the analytical form of the sparse cross entropy.

Proposition 1 *Consider an augmentation of $\boldsymbol{x} \sim p_D(\boldsymbol{x})$ from the observed data space: $\mathcal{X} = \{(\boldsymbol{x}, \boldsymbol{z}) : \boldsymbol{x} \in \mathcal{R}^M$ and $\boldsymbol{z} \in \{0, 1\}^M\}$, where $z_i = 1$ if $x_i \neq 0$, and $z_i = 0$ if $x_i = 0$. The*

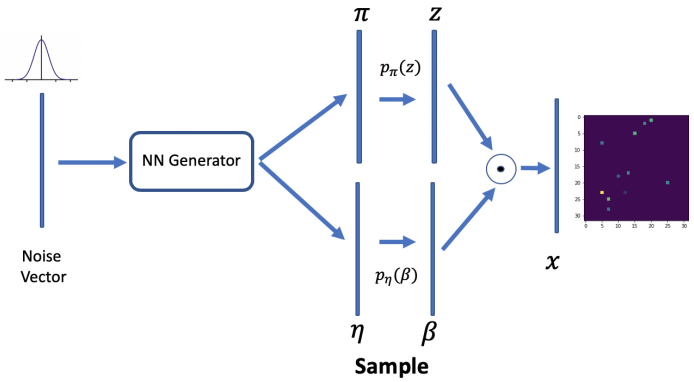


Figure 1: Architecture of the proposed sparse image generator. The noise vector is fed into a neural network generator, where the model learns the dependency of each pixel value of the image. The network then outputs two vectors: $\boldsymbol{\pi}$ and $\boldsymbol{\eta}$, which are used to sample $\boldsymbol{z}$, $\boldsymbol{\beta}$.

cross-entropy between data distribution $p_D(\boldsymbol{x}, \boldsymbol{z})$ *and the generated distribution* $p_{\boldsymbol{\eta},\boldsymbol{\pi}}(\boldsymbol{x}, \boldsymbol{z})$ *is*

$$\begin{aligned} -\mathcal{H}(p_D(\boldsymbol{x}, \boldsymbol{z}), p_{\boldsymbol{\eta},\boldsymbol{\pi}}(\boldsymbol{x}, \boldsymbol{z})) &= E_{\boldsymbol{x},\boldsymbol{z}\sim p_D(\boldsymbol{x},\boldsymbol{z})} \log p_{\boldsymbol{\eta},\boldsymbol{\pi}}(\boldsymbol{x}, \boldsymbol{z}) \\ &= E_{\boldsymbol{x},\boldsymbol{z}\sim p_D(\boldsymbol{x},\boldsymbol{z})} \sum_{i=1}^{M} [log\pi^{z_i} + \log(1-\pi)^{(1-z_i)} + \log p_{\eta_i}(x_i)] \end{aligned} \tag{1}$$

Since the conditional independence is assumed, we have $p_{\boldsymbol{\eta},\boldsymbol{\pi}}(\boldsymbol{x}, \boldsymbol{z}) = \sum_{i=1}^{M} p_{\eta_i,\pi_i}(x_i, z_i)$. The above proposition can be proven by expressing the joint distribution as $p(x_i, z_i) = p(x_i|z_i)p(z_i)$. Since z_i is discrete, the marginal $p(x_i, z_i)$ can be decomposed into $p(x_i, z_i) = p(x_i|z_i = 1)p(z_i = 1) + p(x_i|z_i = 0)p(z_i = 0)$.

The above cross entropy objective turns out to have nice interpretation as well. When the true pixel value of the i-th pixel is zero, it will only penalize the probability of $\pi_i = p(z_i = 1)$ to be large, while having no penalty on β_i. And when $x_i > 0$, it will penalize π_i to be small as well as β_i different from x_i. In comparison, if a simple mean squared error (MSE) loss is used, which is typically the case in plain variational autoencoders [19], it is not clear how the model is able learn the mixture distribution of output x_i. While the sparse cross entropy loss (1) allows us to model the distribution when $x_i \neq 0$ separately from the point mass placed at $x_i = 0$ for every i. It is worth noting that the sparse cross entropy loss can easily be applied to the likelihood model of VAE, which itself is of independent interest.

The expectation in (1) is approximated using Monte Carlo integration by taking samples of $\boldsymbol{x}$, $\boldsymbol{z}$ and computing the average. Combining the distribution of $\boldsymbol{\eta}, \boldsymbol{\pi}$ which is induced by the neural network generator, the complete training objective is:

$$\text{Minimize} \quad -E_{p_D(\boldsymbol{x},\boldsymbol{z})} E_{p_{\phi_1,\phi_2}(\boldsymbol{\pi},\boldsymbol{\eta})} \log p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\pi}, \boldsymbol{\eta}) \quad \text{w.r.t} \quad \phi_1, \phi_2 \tag{2}$$

Again, Monte Carlo integration is used to approximate $E_{p_{\phi_1,\phi_2}(\boldsymbol{\pi},\boldsymbol{\eta})} \log p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\pi}, \boldsymbol{\eta})$.

Proposition 2 *Minimizing the objective (2) is equivalent to minimizing a lower bound of the cross entropy* $-E_{\boldsymbol{x},\boldsymbol{z}\sim p_D(\boldsymbol{x},\boldsymbol{z})} \log p_{\phi_1,\phi_2}(\boldsymbol{x}, \boldsymbol{z})$

Noting that $E_{p_{\phi_1,\phi_2}(\boldsymbol{\pi},\boldsymbol{\beta})} \log p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\pi}, \boldsymbol{\beta}) \leq \log p_{\phi_1,\phi_2}(\boldsymbol{x}, \boldsymbol{z})$, proposition 2 is an application of Jensen's inequality.

4 Experiments

We conducted our experiment on the proposed muon dataset. The likelihood model $p_{\boldsymbol{\eta}}(\boldsymbol{x})$ used is an isotropic Gaussian with mean parameter $\boldsymbol{\eta}$ and standard deviation is set to the empirical standard

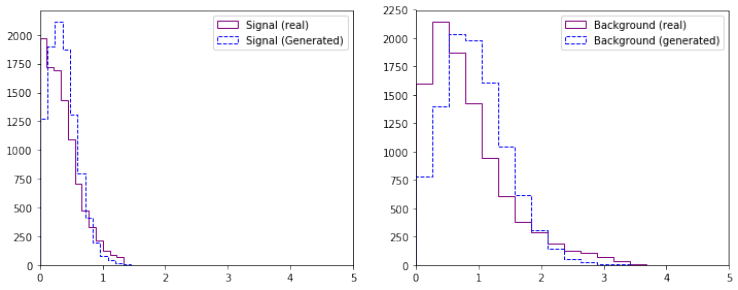
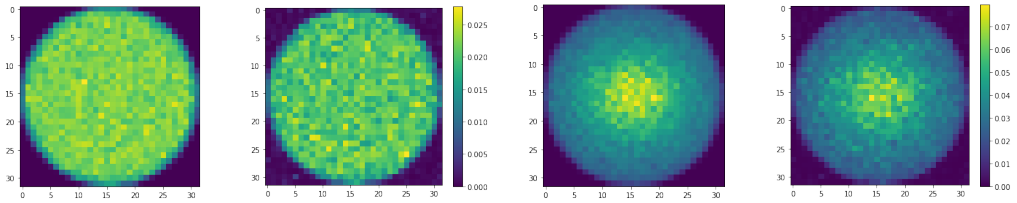


Figure 2: Isolation value distribution for signal images and background images



(a) Average of signal images

(b) Average of background images

Figure 3: Average of images across the dataset. Real images are on the left of each subfigure and the images generated by our model on the right

deviation of the image. We tried a 4-layer deconvolutional generator as well as a 4-layer fully connected generator and both of them yield similar results. And we trained two models, one on for the signal image and one for the background image. Both of the models converge quickly within 30 epochs. We validate the quality of the images in our model by qualitative and quantitative methods. Qualitatively we consider the average of all the images in the real and generated dataset across pixels. The results can be seen in figure 3. Here we can see the real and generated images are very similar. In the case of signal images the uniform distribution is reproduced while in the background we see a concentration in the center for both the real and generated image.
We consider physically motivated functions to quantitatively evaluate the performance of our model similarly to Oliveira et al. [12]. In this case we consider isolation with a radius of 0.2 at the center of the image. We calculate the isolation for an image I with muon $P_{\mathrm{T}\mu}$ at a radius 0.2 as:

$$Iso(I, P_{\mathrm{T}\mu}) = \frac{\sum_{cells}^{\Delta R<0.2} I(\phi, \eta)}{P_{\mathrm{T}\mu}} \quad \text{where} \quad \Delta R = \sqrt{\Delta\phi^2 + \Delta\eta^2} \tag{3}$$

Since all the images have a $P_{\mathrm{T}\mu}$ value in the range $P_{\mathrm{T}\mu} \in [10, 15]$ GeV, to calculate the isolation we use the known value of $P_{\mathrm{T}\mu}$ for the real images and we approximate the value for the generated images as the mean for the real images $P_{\mathrm{T}\mu} = 11.85$. We can see in figure 2 that the distributions for real and generated images are very similar but generated images have a slightly higher mean isolation value in both cases and fewer generated images have isolation values close to zero. This could be an effect of using the approximation of using the mean $P_{\mathrm{T}\mu}$ and could be improved by including $P_{\mathrm{T}\mu}$ as an input parameter for the model.

We also compare the Earth Moving Distance of the isolation value distribution between the real images and the generated images, which can be seen as the minimum amount of "work" required to transform one distribution into another. The distance for the signal and background images are 0.06 and 0.17 respectively. As a baseline, we calculated the isolation value of the images generated by a model which independently samples from the empirical distribution of each pixel, the distance is 0.35 and 0.89 for signal and background, which indicates that our model is able to capture the dependence structure among the pixels.

5 Conclusion

We propose a decoupled hierarchical generative model by compounding an explicit mixture model with an implicit distribution produced by a neural network generator. We demonstrate that both sparsity level and the non-zero pixel distribution can be effectively learned by minimizing a lower bound of the cross-entropy between the data distribution and the generated distribution. We believe our method opens up an interesting way of principled probabilistic modeling of the sparse distribution and it is easier to interpret the sparsity level comparing to GAN approaches.

As for future work, we are looking forward to comparing our model with LAGAN in the jet dataset used in their paper as well as testing the performance of LAGAN in our muon dataset for further comparison. In addition, more flexible distributions other than an isotropic Gaussian can be incorporated to model the non-zero distribution as long as it has tractable likelihood function.

Acknowledgments

We wish to acknowledge a hardware grant from NVIDIA. This material is based upon work supported by the National Science Foundation under grant number 1633631.

References

[1] Mustafa Mustafa, Deborah Bard, Wahid Bhimji, Zarija Lukić, Rami Al-Rfou, and Jan M. Kratochvil. Cosmogan: creating high-fidelity weak lensing convergence maps using generative adversarial networks. *Computational Astrophysics and Cosmology*, 6(1):1, 2019.

[2] P. Musella and F Pandolfi. Fast and accurate simulation of particle detectors using generative adversarial networks. *Computing and Software for Big Science*, 2018.

[3] Kai Zhou, Gergely Endrődi, Long-Gang Pang, and Horst Stöcker. Regressive and generative neural networks for scalar field theory. *Phys. Rev. D*, 100:011501, 2019.

[4] G. r. Khattak, S. Vallecorsa, and F. Carminati. Three dimensional energy parametrized generative adversarial networks for electromagnetic shower simulation. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 3913–3917, 2018.

[5] Saúl Alonso Monsalve and Leigh Whitehead. Image-based model parameter optimisation using model-assisted generative adversarial networks. 2018.

[6] K. Deja, T. Trzciński, and Ł. Graczykowski. Generative models for fast cluster simulations in the tpc for the alice experiment. In *Information Technology, Systems Research, and Computational Physics*, pages 267–280, Cham, 2020. Springer International Publishing.

[7] Maurizio Pierini Amir Farbin Benjamin Hooberman Wei Wei Matt Zhang Vitória Barin Pacela Sofia Vallecorsafac Maria Spiropulu Federico Carminati, Gulrukh Khattak and Jean-Roch Vlimant. Calorimetry with deep learning: Particle classification, energy regression, and simulation for high-energy physics. *Deep Learning for Physical Sciences, Workshop at the 31st Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

[8] Sambuddha Ghosal Balaji Pokuri Soumik Sarkar Baskar Ganapathysubramanian Chinmay Hegde Viraj Shah, Ameya Joshi. Encoding invariances in deep generative models. 2019.

[9] Aishik Ghosh Tobias Golling Gilles Louppe David Rousseau Dalila Salamani Kyle Cranmer, Stefan Gadatsch and Graeme Stewart on behalf of the ATLAS Collaboration. Deep generative models for fast shower simulation in atlas. *Bayesian Deep Learning, Workshop at the 32nd Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[10] Kaustuv Datta Dominick Olivito Bobak Hashemi, Nick Amin and Maurizio Pierini. Lhc analysis-specific datasets with generative adversarial networks. *ArXiv*, abs/1901.05282, 2019.

[11] Wieske de Swart Melissa van Beekveld Luc Hendriks Caspar van Leeuwen Damian Podareanu Roberto Ruiz de Austri Sydney Otten, Sascha Caron and Rob Verheyen. Event generation and statistical sampling for physics with deep generative models and a density information buffer. *ArXiv*, https://arxiv.org/abs/1901.00875, 2019.

[12] Michela Paganini, Luke de Oliveira, and Benjamin Nachman. Calogan: Simulating 3d high energy particle showers in multi-layer electromagnetic calorimeters with generative adversarial networks. *Phys. Rev. D*, 97, 2018.

[13] Luke de Oliveira, Michela Paganini, and Benjamin Nachman. Learning particle physics by example: Location-aware generative adversarial networks for physics synthesis. *Comput Softw Big Sci*, 2017.

[14] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015.

[15] Mingyuan Zhou Mingzhang Yin. Semi-implicit generative model. May, 2019.

[16] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H. S. Shao, T. Stelzer, P. Torrielli, and M. Zaro. The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations. *JHEP*, 07:079, 2014.

[17] Torbjorn Sjostrand, Stephen Mrenna, and Peter Z. Skands. PYTHIA 6.4 Physics and Manual. *JHEP*, 0605:026, 2006.

[18] J. de Favereau et al. DELPHES 3, A modular framework for fast simulation of a generic collider experiment. *JHEP*, 1402:057, 2014.

[19] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.