Deep Learning and Generative Networks for the simulation of high-energy physics events

Student: Francesco Vaselli   Advisor: Prof. Andrea Rizzi

Abstract, full thesis to be discussed on 14/09/2022

Introduction and problem framing

In recent years, *machine learning* techniques have been massively adopted by scientific collaboration around the world. In particular, a paradigm known as *deep learning*, which leverages multiple layers of *artificial neurons* (theorized by [16]) trained through the use of a *loss function* and *backpropagation*, has achieved a wide range of applications. Even a simple overview of the subject would be far beyond the scope of this section; we thus limit ourselves to the class of *generative models*.

In the physical sciences, the need for trustworthy and robust event generation is usually tackled by *Monte Carlo methods*, with state of the art libraries (such as [6]) capable of achieving remarkable results at the cost of computational complexity and computing times. The generated data structure is usually tabular or sparse, but may vary greatly between different experiments and collaborations. On the other hand, research in the field of computer vision has fueled development of remarkable deep learning models, focused mainly on image generation. *Generative Adversarial Networks* (GANs) [7], *Variational Autoencorders* (VAEs) [10] and *Normalizing Flows* [14] are some of the most successful frameworks developed to this date. However, such tools remain geared towards the necessities of industry; much work remains to be done to enable the use of this technologies in real, hard sciences applications. The main aim of this work is thus to develop a fast and reliable event generation framework based on deep learning. The key idea is to directly generate a high level analysis format, such as CMS NANOAOD [15], training on fully simulated events. As a benchmark to evaluate the performance of such a simulation, the search data for the decay of Higgs to muon pairs in the VBF channel has been chosen. This kind of analysis requires only a limited number of muons and jets features to be simulated while still depending upon proper handling of correlations, so it is a good benchmark for a first prototype of this deep learning based approach. The goal of this kind of simulation, that we call *flashsim*, is to generate the full detector response (simulation and reconstruction) in a negligible time compared to a full simulation, hence enabling the generation of future large datasets at low computing cost, e.g. to study systematic uncertainties in LHC Run3 or to generate HL-LHC samples.

Thesis work and personal contribution

Both GANs and VAEs have already been extensively investigated by the collaboration at CERN (see [3] and [12]); despite this, there is still a limited literature regarding behavior in low dimensionality as in our case, e.g. [9]. We initially focused on GANs, through the use of state of the art libraries such as Tensorflow [1] and Pytorch [13]. Despite some convincing results in published works, this approaches remain plagued by problems such as *mode collapse*, where the generator over-optimizes for a particular discriminator, and the discriminator never manages to learn its way out of the trap. As a result the generators rotate through a small set of output types, degrading the statistical significance of generated samples. Another common occurrence is failure to converge, due to the peculiar min-max nature of the training. We investigated possible remedies and architectures, such as *Wasserstein GAN*, which implements a loss metric derived from the *earth mover distance* between the real and generated distributions (see [2]), or *Unrolled GAN*s, which use a generator loss function that incorporates not only the current discriminator's classifications, but also the outputs of future discriminator versions (see [11]). We also implemented a custom *Bitted GAN*, trained on binarized data aiming to directly predict the bin in the histogram output distributions. We obtained no meaningful results, and simple tests performed for VAEs yielded similar outcomes.

We thus turned to the approach of Normalizing Flows, a family of methods for constructing flexible learnable probability distributions, often with neural networks, which allow us to surpass the limitations of simple parametric forms to represent complex high-dimensional distributions. In this case, a simple multivariate source of noise, for example a standard i.i.d. normal distribution, $X \sim \mathcal{N}(\mathbf{0}, I_{D\times D})$, is passed through a vector-valued invertible bijection, $g : \mathbb{R}^D \to \mathbb{R}^D$, to produce the more complex transformed variable $Y = g(X)$. We can compose such bijective transformations to produce even more complex distributions. It can be seen that, if we have L transforms $g_{(0)}, g_{(1)}, \ldots, g_{(L-1)}$, then the log-density of the transformed variable $Y = (g_{(0)} \circ g_{(1)} \circ \cdots \circ g_{(L-1)})(X)$ is:

$$\log(p_Y(y)) = \log\left(p_X\left(\left(g^{-1}_{(L-1)} \circ \cdots \circ g^{-1}_{(0)}\right)(y)\right)\right) + \sum_{l=0}^{L-1} \log\left(\left|\frac{dg^{-1}_{(l)}(y_{(l)})}{dy'}\right|\right)$$

Remembering that such transformations depend on a set of learnable parameters, the previous equation naturally lends itself to being interpreted as the *loss function* for our problem.

The main challenge is in designing parametrizable multivariate bijections that have closed form expressions for both g and g^{-1}, a tractable Jacobian whose calculation scales with $O(D)$ rather than $O(D^3)$, and can express a flexible class of functions. Recent advancements have demonstrated the suitability of *spline transforms* (see [5]).

The theory of Normalizing Flows is also easily generalized to conditional distributions. We denote the variable to condition on by $C = \mathbf{c} \in \mathbb{R}^M$. A simple multivariate source of noise, for example a standard i.i.d. normal distribution, $X \sim \mathcal{N}(\mathbf{0}, I_{D\times D})$, is passed through a vector-valued bijection that also conditions on C, $g : \mathbb{R}^D \times \mathbb{R}^M \to \mathbb{R}^D$, to produce the more complex transformed variable $Y = g(X; C = \mathbf{c})$. In practice, this is usually accomplished by making the parameters for a known normalizing flow bijection g the output of a hypernet neural network that inputs $\mathbf{c}$. It is thus straightforward to condition event generation on the ground truth employed for the Monte Carlo target generation.

Following [8], we built a neural spline normalizing flow composed of 15 transformations layers and successfully learned to reproduce 15 variables for Jets from a NANOAOD sample of 1e6 events. Both the 1-d Wasserstein distance from real sample distributions and pair correlations prove the goodness of the current approach, which manages to obtain convincing samples in a fraction of the time compared to the full reconstruction.

Future work will see us working toward realistic simulation of all the target variables for the $\mathrm{H} \rightarrow \mu^{+}\mu^{-}$ events, as well as looking into possible extension into *Quantum Machine Learning* (as in [4]).

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017.

[3] Anja Butter, Tilman Plehn, and Ramon Winterhalder. How to gan lhc events. *SciPost Physics*, 7(6), Dec 2019.

[4] Su Yeon Chang, Sofia Vallecorsa, Elías F. Combarro, and Federico Carminati. Quantum generative adversarial networks in a continuous-variable architecture to simulate high energy physics detectors, 2021.

[5] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows, 2019.

[6] S. Agostinelli et al. Geant4—a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 506(3):250–303, 2003.

[7] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[8] Stephen R. Green and Jonathan Gair. Complete parameter inference for gw150914 using deep learning, 2020.

[9] Felix Jimenez, Amanda Koepke, Mary Gregg, and Michael Frey. Generative adversarial network performance in low-dimensional settings, 2021-04-20 04:04:00 2021.

[10] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2014.

[11] Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled generative adversarial networks, 2017.

[12] Sydney Otten, Sascha Caron, Wieske de Swart, Melissa van Beekveld, Luc Hendriks, Caspar van Leeuwen, Damian Podareanu, Roberto Ruiz de Austri, and Rob Verheyen. Event generation and statistical sampling for physics with deep generative models and a density information buffer, 2021.

[13] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[14] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows, 2016.

[15] Andrea Rizzi, Giovanni Petrucciani, and Marco Peruzzi. A further reduction in CMS event data for analysis: the NANOAOD format. In *European Physical Journal Web of Conferences*, volume 214 of *European Physical Journal Web of Conferences*, page 06021, July 2019.

[16] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65 6:386–408, 1958.