Controllable Generation for Climate Modeling

Moulik Choraria
University of Illinois at Urbana-Champaign
moulikc2@illinois.edu

Daniela Szwarcman
IBM Research
daniela.szw@ibm.com

Bianca Zadrozny
IBM Research
biancaz@br.ibm.com

Campbell D. Watson
IBM Research
cwatson@us.ibm.com

Lav R. Varshney
University of Illinois at Urbana-Champaign
varshney@illinois.edu

Abstract

Recent years have seen increased interest in modeling future climate trends, especially from the point of view of accurately predicting, understanding and mitigating downstream impacts. For instance, current state-of-the-art process-based agriculture models rely on high-resolution climate data during the growing season for accurate estimation of crop yields. However, high-resolution climate data for future climates is unavailable and needs to be simulated, and that too for multiple possible climate scenarios, which becomes prohibitively expensive via traditional methods. Meanwhile, deep generative models leveraging the expressivity of neural networks have shown immense promise in modeling distributions in high dimensions. Here, we cast the problem of simulation of climate scenarios in a generative modeling framework. Specifically, we leverage the GAN (Generative Adversarial Network) framework for simulating synthetic climate scenarios. We condition the model by quantifying the degree of “extremeness" of the observed sample, which allows us to sample from different parts of the distribution. We demonstrate the efficacy of the proposed method on the CHIRPS precipitation dataset.

1 Introduction

Noticeable shifts in climate patterns have been observed over the last few decades, including an increase in the frequency and intensity of extreme weather events. Such events often result in severe societal impacts, with risks to agriculture, access to water resources, energy management and transportation, among others [1]. Therefore, accurate modeling and prediction of climate patterns is of immense interest. For instance, process-based models, like crop growth or hydrological models, that are used to assess potential impacts on climate-sensitive sectors [2, 3] generally require long time series of high-resolution weather data [2, 4].

Stochastic weather generators have been widely used to provide synthetic weather series that represent plausible climate scenarios to the impact models [5, 6, 2, 3]. The first stochastic weather generators date to the 1980s [7], and since then, several techniques have been explored with different levels of complexity [4]. However, traditional models often fail to generate extremes such as droughts or flooding [2]. Some weather generators can be conditioned on external variables, e.g. [2], which produces weather sequences consistent with temperature and precipitation seasonal forecasts. That is, the external variables *control* the generation of weather fields. The idea of controlling weather

Tackling Climate Change with Machine Learning: workshop at NeurIPS 2021.

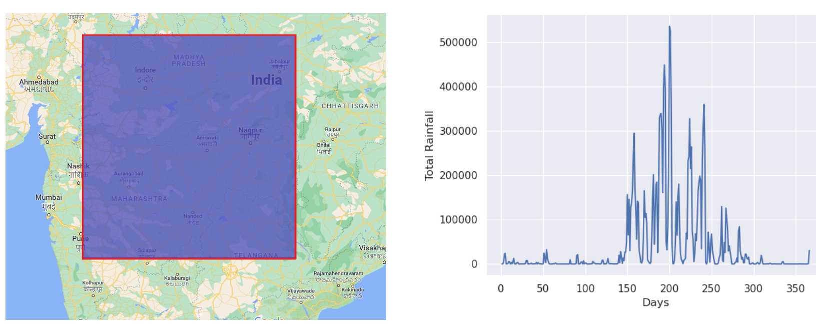
field synthesis is essential for producing what-if scenarios to study its particular effects on sensitive areas. For example, local governments may need to investigate the risks of floods in a region subject to consecutive days of plausible heavy rainfall events.

In recent years, machine learning based generative models have achieved notable success in modeling high dimensional complex distributions, which has led to their application in stochastic weather generation. For instance, Variational AutoEncoders (VAEs) [8] have been used to generate precipitation data and control synthesis by sampling from different regions of the latent space [9], while Generative Adversarial Network (GAN) [10] based approaches have been shown to generate realistic precipitation samples [11] and model weather extremes [12]. In this work, we make use of the GAN framework for stochastic weather modeling, focusing on conditional generation. Conditional GANs [13] have shown great success in several tasks involving an input controlling the data generation, such as image-to-image translation [14, 15]. More recently, researchers have explored these models for weather generation [12, 16]. In this paper, we leverage the conditional GAN framework to generate synthetic precipitation scenarios. For this, we propose a simple yet effective conditioning metric and empirically demonstrate that it allows to accurately model and sample from different quantiles of precipitation in the data distribution.

2 Methods and Experiments

Data For our experiments, we use the Climate Hazards Group InfraRed Precipitation with Station data (CHIRPS) [17] rainfall data set. It consists of daily precipitation fields with a spatial resolution of 0.05 degrees, and spans from 1981–2021. For our experiments, we use a bounding box of 6.4 $\times$ 6.4 degrees, which equates to a resolution of 128 $\times$ 128 pixels, where each pixel represents a geographical region spanning a spatial resolution of 0.05 $\times$ 0.05 degrees. The chosen spatial region is bound by latitudes (18.0, 24.4) and longitudes (74, 80.4), a region in central India shown in Fig. 1a.

Researchers are often more interested in capturing seasonal trends, for instance precipitation during the growing season. As such, we limit our focus to precipitation samples between June–October, which roughly corresponds to the monsoon (rainy) season in central India. Fig. 1b indeed verifies that most precipitation in the region occurs during the selected months (roughly between days numbered in the range 150–290). Additionally, we note that samples within this sub-interval successfully capture most of the variation in the total amount of rainfall, allowing us to sub-sample from the large amount samples without losing fidelity. Finally, for consistent evaluation of our method, we divide our data into train and test sets, corresponding to years 1981–2011 and 2012–2021, respectively.



(a) Region of interest in central India. (b) Rainfall distribution for a given year.

Figure 1: This figure depicts the chosen region for our experiments, along with the typical annual rainfall distribution over that region.

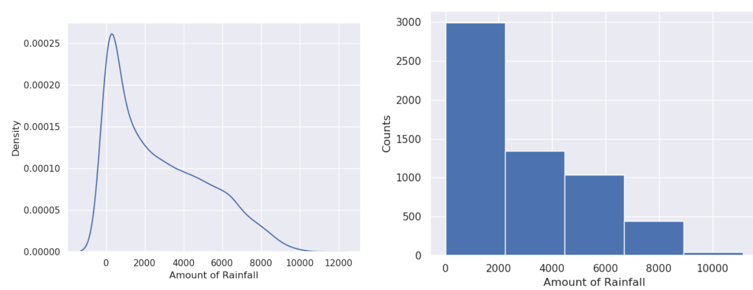
Preprocessing For training deep models on this dataset, we require the samples to be suitably normalized. However, note that unlike standard images which have a fixed support (0-255), precipitation values can be arbitrarily high and the large variations between individual pixel values often increase the difficulty of the learning task, when using standard min-max or unit-Gaussian normalization. To alleviate this issue, we first use a log transform on the individual pixel values, specifically $f(x) = \log(1 + x)$ which suppresses the large outliers that can skew the data, followed by standard min-max normalization such that pixel values lie in [0, 1] for all images. Finally, we note that samples with zero rainfall equates to nearly 10% of the total training samples. Since, these samples are not particularly interesting from a generative point of view (indeed, we can manually create zero rainfall images), we discard the majority of them from the training set.

Method We consider the GAN framework. The standard GAN consists of two models: the generator (G) and the discriminator (D). The generator, parameterized by θ_g, learns a distribution p_g over the data samples via a prior on its input noise variables $p_z(z)$ in the latent space, thus representing a map to data space as $G(z;\theta_g)$. The discriminator, parameterized as $D(x;\theta_d)$, takes a sample in the data space and outputs a single scalar $D(x)$, that represents the probability that observed x came from the true data distribution, as opposed to p_g. D is trained to maximize the probability of correctly distinguish training examples from samples coming via p_g. Simultaneously, G is trained to minimize the probability of detection via discriminator, resulting in the following minimax objective:

$$\min_G \max_D \mathbb{E}_{x\sim p_{data}} \log D(x) \mathbb{E}_{z\sim p_z(z)} \log(1 - D(G(z))).$$

For the generator and discriminator networks, we use standard Resnet[18]-based implementations.

Conditioning Note that within the standard GAN parametrization, the generator samples from the prior $p_z(z)$, which is generally standard Gaussian, and maps it to the data space. However, due to the uni-modal prior, the generated samples concentrate around the mode of the training data, which in our case corresponds to low precipitation samples (since on average, most days have little/no rainfall). Due to this concentration, obtaining diverse realistic samples corresponding to higher values of rainfall becomes harder. To bypass this issue, we use conditioning in the GAN framework to guide the generation. To allow sampling from different parts of the distribution, we first obtain the distribution of the total rainfall per daily precipitation field, across the entire dataset. We then discretize the distribution into a histogram with n bins, as indicated in Fig. 4. Note that bin membership in the histogram serves as a pseudo-measure for the "extremeness" of the sample. In this example, the first bin corresponds to low values of total rainfall, whereas bin 5 corresponds to extreme samples. Hence, we can use bin membership $M \in \{1, \ldots, n\}$ as a conditioning variable to guide our generation. In practice, this is implemented as in standard conditional GANs with an embedding layer, which is appended to the noise prior to indicate the value of the membership.



(a) Distribution of total rainfall over the dataset.

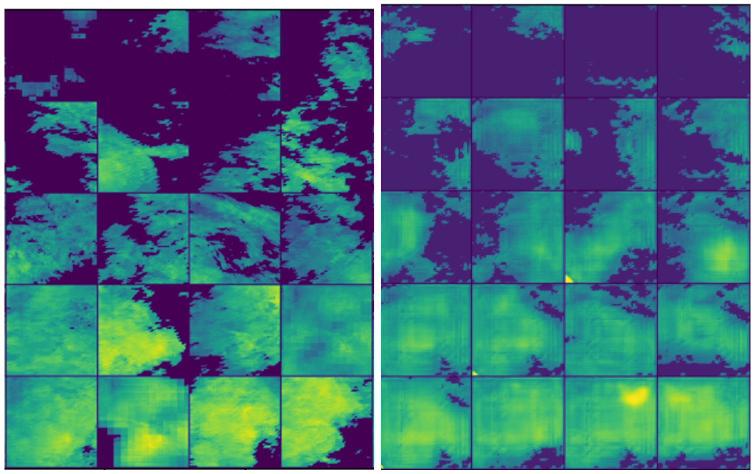
(b) Distribution on the left discretized into 5 histogram bins.

Figure 2: This figure depicts our method of conditional generation based on total rainfall, where each quantile in the discretized rainfall distribution represents a value the conditioning variable can take.

Sampling Note that conditioning by itself does not resolve the uni-modal concentration problem, since the model would largely see samples from the first bin during training. To resolve this, we use weighted sampling during training, defining a sampling weight w_i for each sample in a given bin i as $w_i \propto \frac{1}{N(i)+k.T}$, where $N(i)$ is the number of samples in bin i, T is the total number of training samples, and k is a parameter that controls the degree to which sampling resembles uniform. We set k to a small value to ensure that we roughly observe equal samples from each bin, thus oversampling from the higher rainfall quantiles.

3 Results

GAN evaluation is a notoriously hard problem [19], since the objective function does not link directly to the data distribution. For image tasks, Frechet Inception Distance (FID) [20] is a popular method of evaluation but it depends on features learnt on natural images, which has a different distribution from our dataset. Therefore, we rely qualitatively on visual inspection (Fig. 3) and quantitatively, on comparing the density of the rainfall distributions over the generated samples and the test set (Fig. 4). For the latter, we sample equal number of samples(~ 600) from each quantile in the histogram and evaluate the corresponding Kernel Density Estimates (KDE) [21] against samples from the test set.



(a) Training Samples (b) Generated Samples

Figure 3: Row i corresponds to samples from the i^{th} quantile and while some high level details are missed, we observe that total rainfall for generated the sample increases with increasing quantile.

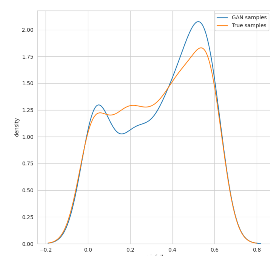


Figure 4: Total rainfall density comparison shows generated samples match the test distribution.

With regards to visual inspection, we observe the efficacy of our scheme in generating diverse samples from different rainfall quantiles. In the KDE comparison, we note that the densities from the two domains indeed match closely. However, there remains scope for improvement. First, we note the presence of grid-like artefacts, which is known to be associated with transposed convolutional operators. Secondly, the network has a tendency to smooth-en out the spatial extremes, pointing towards the need for a finer grained conditioning, for better control over individual patches in the image.

4 Discussion

In this work, we developed a novel method of conditioning weather generators that serves as an effective tool for controlling the synthesis of weather fields. Our work opens up multiple lines of future work. Firstly, we note that a finer grained discretization would enable more control of the generation process, including that of extreme scenarios. Secondly, this method can easily be extended for generation of other weather variables and to a different choice of statistics. Additionally, note that we can extend this method for generating higher resolution images: one could create corresponding lower resolution maps whose each co-ordinate conditions/defines the extremeness of a mini-grid in the higher resolution image. Finally, experimenting with this method in other generative frameworks such as state-of-the-art diffusion models is another interesting line of research.

References

[1] Colin Raymond, Radley M Horton, Jakob Zscheischler, Olivia Martius, Amir AghaKouchak, Jennifer Balch, Steven G Bowen, Suzana J Camargo, Jeremy Hess, Kai Kornhuber, et al. Understanding and managing connected extreme events. *Nature Climate Change*, 10(7):611–621, 2020.

[2] Andrew Verdin, Balaji Rajagopalan, William Kleiber, Guillermo Podestá, and Federico Bert. A conditional stochastic weather generator for seasonal to multi-decadal simulations. *Journal of Hydrology*, 556:835–846, 2018.

[3] Martin Dubrovskỳ, Josef Buchtele, and Zdeněk Žalud. High-frequency and low-frequency variability in stochastic daily weather generator and its effect on agricultural and hydrologic modelling. *Climatic Change*, 63(1):145–179, 2004.

[4] Nadav Peleg, Simone Fatichi, Athanasios Paschalis, Peter Molnar, and Paolo Burlando. An advanced stochastic weather generator for simulating 2-d high-resolution climate variables. *Journal of Advances in Modeling Earth Systems*, 9(3):1595–1627, 2017.

[5] Nadav Peleg, Peter Molnar, Paolo Burlando, and Simone Fatichi. Exploring stochastic climate uncertainty in space and time using a gridded hourly weather generator. *Journal of Hydrology*, 571:627–641, 2019.

[6] Lionel Benoit, Mathieu Vrac, and Gregoire Mariethoz. Nonstationary stochastic rain type generation: accounting for climate drivers. *Hydrology and Earth System Sciences*, 24(5):2841–2854, 2020.

[7] Samaneh Sohrabi, François P Brissette, and Richard Arsenault. Coupling large-scale climate indices with a stochastic weather generator to improve long-term streamflow forecasts in a canadian watershed. *Journal of Hydrology*, 594:125925, 2021.

[8] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013.

[9] Dario Augusto Borges Oliveira, Jorge Guevara Diaz, Bianca Zadrozny, and Campbell Watson. Controlling weather field synthesis using variational autoencoders. *arXiv preprint arXiv:2108.00048*, 2021.

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

[11] Xuelong Li, Kai Kou, and Bin Zhao. Weather gan: Multi-domain weather translation using generative adversarial networks, 2021.

[12] Siddharth Bhatia, Arjit Jain, and Bryan Hooi. Exgan: Adversarial generation of extreme samples. In *Proc. AAAI Conf. Artif. Intell.*, volume 35, pages 6750–6758, 2021.

[13] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets, 2014.

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proc. IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)*, pages 1125–1134, 2017.

[15] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 2223–2232, 2017.

[16] Cunguang Wang, Guoqiang Tang, and Pierre Gentine. Precipgan: Merging microwave and infrared data for satellite precipitation estimation using generative adversarial network. *Geophysical Research Letters*, 48(5):e2020GL092032, 2021.

[17] Chris Funk, Pete Peterson, Martin Landsfeld, Diego Pedreros, James Verdin, Shraddhanand Shukla, Gregory Husak, James Rowland, Laura Harrison, Andrew Hoell, and Joel Michaelsen. The climate hazards infrared precipitation with stations—a new environmental record for monitoring extremes. *Scientific Data*, 2(1):150066, December 2015.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[19] Ali Borji. Pros and cons of GAN evaluation measures. *CoRR*, abs/1802.03446, 2018.

[20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Günter Klambauer, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a nash equilibrium. *CoRR*, abs/1706.08500, 2017.

[21] Yen-Chi Chen. A tutorial on kernel density estimation and recent advances, 2017.