Supplemental materials

Diffusion model based downscaling of extreme weather in southern Europe

Joshua Miller, Kate Halladay, Rachel James, Peter Watson

---

1 More metrics

Here we show the spread-skill diagram for our model on both the extreme and validation datasets. The spread skill diagram compares the root mean squared error (RMSE) to the spread of diffusion model samples (RMSS). This allows one to compare error in the diffusion model sames versus the root mean square spread of the different diffusion model samples. The higher the spread, the more different the diffusion model samples are from one another.

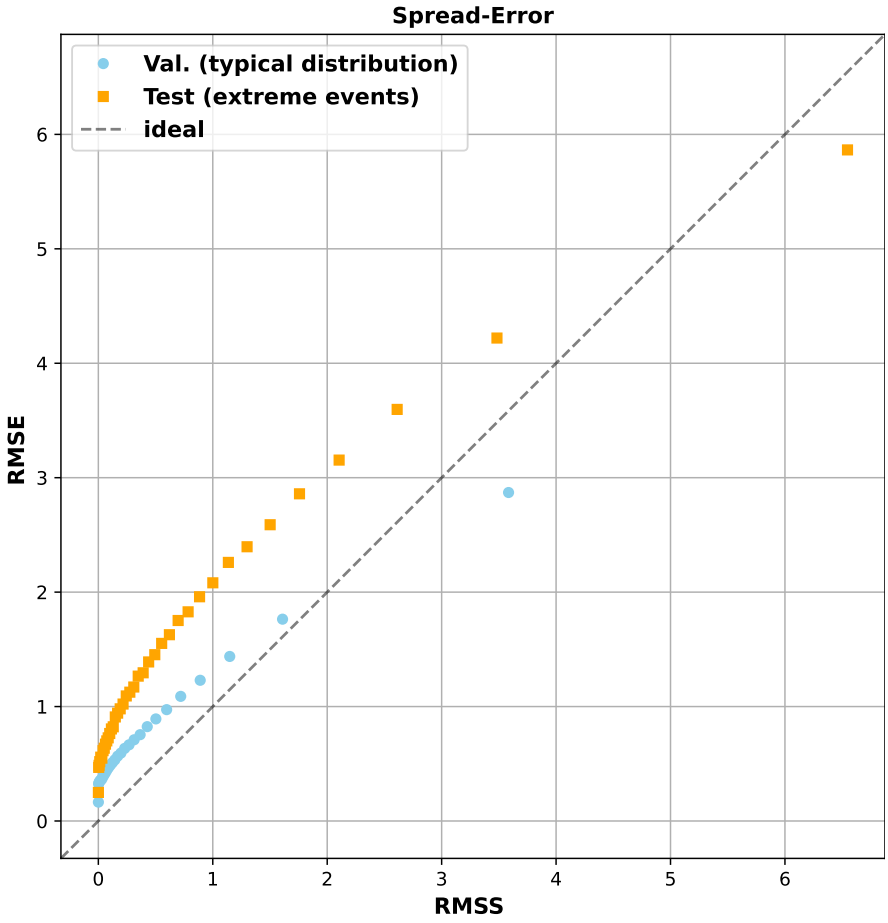

Figure 1: The spread-skill plot for the diffusion model on the validation dataset (blue) and extremes dataset (orange). Units for both axes are mm/hr.

We see that the models tend to be underspread compared to the ideal relationship (black line), and there is more error on the extreme events.

2 Notes on training the diffusion model

Our model contains 63 million trainable parameters and is trained using 4 NVIDIA A100 GPUs. One epoch–one pass through all of the training data–takes approximately 35 minutes, and generating one diffusion sample for one particular rainfall event takes approximately 0.66 seconds. To find the optimal epoch on which to investigate the model's performance, we generate samples on the validation data at each epoch. Then we compare the models' performance according to mean bias and CRPS to select the best epoch. In this case, the epoch selected was 35. Then the model corresponding to epoch 35 was used to generate samples on the extremes.

3 Discussion about how extreme events are selected

A key facet in interpreting our results is understanding how extreme events are defined and selected. In the poster, we discuss how we utilize a U-Net, a deterministic machine learning model, to simulate the IMERG v07 data, and then select extremes by coarsening the U-Net predictions with a $3^o \times 3^o$ average pooling kernel and then applying a 5mm/hr threshold.

There are two reasons why we used a U-Net to select extremes. The first is that some rainfall events are highly stochastic in nature; thus, even a perfectly skillful model at mapping the input data to the observed rainfall would fail to accurately simulate these events, because the stochasticity isn't captured in the input data. Secondly, we perform a "perfect model" experiment to illustrate the problems with selecting extremes based soley on observed intensity. We simulate all of the IMERG data using the Karras et al. (2022) diffusion model four times–four samples for each hour (the number four is arbitrary). Then, we set the first sample four each hour, "Sample 1," as the new ground truth data and apply the $3^o \times 3^o$ coarsening and 5mm/hr threshold to extract extremes. Then, we plot the histogram of the extreme events from "Sample 1," and for the *same timesteps* we plot the histograms of "Samples 2,3,4." The results are shown in *Figure 2*. Observe how there is a clear separation between "Sample 1" and "Samples 2,3,4." Because all samples come from the same diffusion model, the underestimation is *only* a result of sampling bias, not model error.

Therefore, by using a U-Net to select extreme events, we can find events which have a strong predictable component which is encoded in the input data; the model will have the correct information it needs to create an extreme event. Secondly, we avoid creating a sampling bias by using a threshold applied directly to the IMERG data.

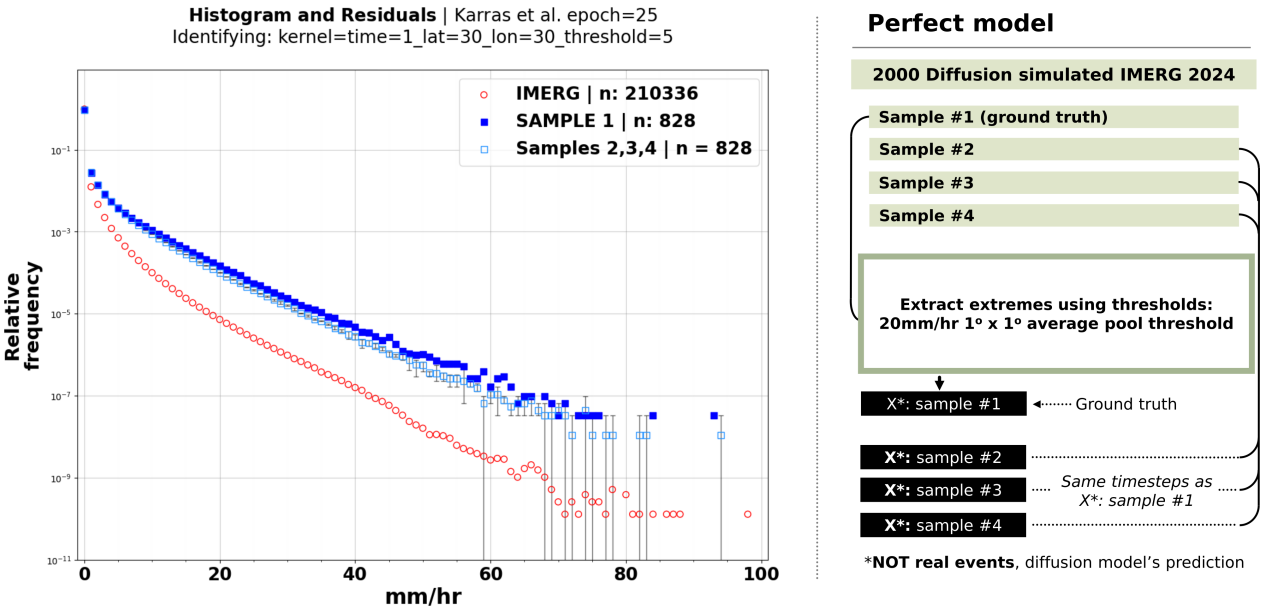

Figure 2: Perfect model experiment. The IMERG data is simulated four times using a diffusion model, and extreme events from sample 1 are selected using a $3^o \times 3^o$ average pooling kernel followed by a 5mm/hr threshold. We show the the histograms for "Sample 1" (dark blue) and "Samples 2,3,4" for the *same timesteps* (light blue).

4 Further experiments

4.1 Using an older diffusion model

In the poster we mention that we use a diffusion model which is based on Karras et al. (2022). However, this model is a refined version of a score-matching diffusion model first developed by Song et al. (2021). Despite being older, this model is quite successful in downscaling precipitation (see Addison et al. (2024)). A key difference between the two models lies in the noise-to-data stochastic ordinary differential equation (ODE) process; the Song et al (2019) model uses a first-order ODE solver to step through the noise removal steps, while the Karras et al model uses a second-order solver which they argue is more accurate and more efficient.

A rationale for using the Karras et al. (2022) model is that it outperforms the song et al model on some image generation tasks, but a key consideration is that in early tests on our dataset it was nearly 8 times faster at generating a rainfall sample. However, we are also working to replicate all experiments with the Song et al model trained on the same data.

4.2 Scaling anomalies

We also plan to test a pre-trained diffusion model on a synthetic dataset where the input variables: specific humidity, temperature, mean sea level pressure, and vorticity, are artificially scaled so as to produce an even more intense precipitation event. The amount of scaling will be controlled via the formula $new\_variable = climatology + k \times anomaly$ where k is the scaling factor.

References

Addison, H., Kendon, E., Ravuri, S., Aitchison, L., and Watson, P. A. (2024). Machine learning emulation of precipitation from km-scale regional climate simulations using a diffusion model. *arXiv preprint arXiv:2407.14158*.

Karras, T., Aittala, M., Aila, T., and Laine, S. (2022). Elucidating the design space of diffusion-based generative models. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2021). Score-based generative modelling through stochastic differential equations. In *ICLR*.