Graph Neural Networks for Enhancing Ensemble Forecasts of Extreme Rainfall

Christopher Bülte, Sohir Maskey, Philipp Scholl, Jonas von Berg and **Gitta Kutyniok**
Ludwig-Maximilians-Universität München
Munich Center for Machine Learning (MCML)

1. Motivation

- Ensemble numerical weather prediction systems (NWPs) exhibit biases and difficulties in capturing extreme weather and spatial dependencies
- Post-processing methods rarely focus on precipitation and its characteristics

Need to account for complex spatial dependencies and tail behavior!

- **Benefits**: Risk mitigation through more reliable forecasts of extreme events

2. Contributions

- Explicitly model precipitation based on known characteristics:
  - Account for the frequent occurrence of dry periods with a lower end **point mass**
  - Utilize **generalized Pareto distribution** to capture the tail behavior associated with heavy rainfall
- Leveraging graph neural networks to post-process ensemble forecasts of precipitation

3. Probabilistic Precipitation Modeling

Methodology

We base our approach on a distributional regression network (DRN, [1]) that uses a neural network to output the parameters of a specified predictive distribution. The model is then trained by minimizing the Continuous Ranked Probability Score (CRPS, [2]).

Statistical model

Precipitation rates are typically modeled via the lognormal distribution [3]. For stability reasons, we apply a log transformation and model the transformed precipitation rate Y as

$$F_Y(y) := \begin{cases} 0, & y < c \\ \tilde{F}(y) := p + (1-p)\Phi_{\mu,\sigma^2}(y), & y \in [c, u] \\ \tilde{F}(u) + (1 - \tilde{F}(u))\mathrm{GPD}_{u,\sigma_u^2,\xi}(y), & y > u, \end{cases} \tag{1}$$

- c: is added for numerical stability, defining zero precipitation below a certain threshold
- p: the discrete probability mass p at the left endpoint of the distribution accounts for zero precipitation
- u: we model the upper tail $\mathbb{P}(Y \leq x + u \mid Y > u)$ for a threshold u with a generalized Pareto distribution (GPD)

We then use a GNN to **predict the parameters** $\{p, \mu, \sigma^2, \sigma_u^2, \xi, u\}$ per individual station with various meteorological variables from the NWP ensemble as input.

4. Spatial Dependence Modeling with GNNs

Graph representation of the data [4]

- Define graph G with nodes given by the stations
- Edge between nodes i, j if geodesic distance $D_{i,j} \leq d_{max}$ pre-defined threshold

Graph Neural Network

For the GNN, associate with every node a $n_{\text{ens}} \times F$-dimensional feature matrix, with number of ensemble members n_{ens} and number of features F. Define the edge weights $w_{i,j}$ between nodes i and j as the inverse normalized geodesic distances, capturing spatial relationships between stations.

- DeepSet: To obtain a **permutation-invariant** initial node embedding we use a DeepSet [5]: $h_v^{(0)} = \Psi\left(\sum_{n=1}^{n_{\text{ens}}} \rho\left(x_{v,n}\right)\right)$, where $x_{v,n} \in \mathbb{R}^F$ is the node feature vector for station v and ensemble member n and ρ, Ψ are both MLPs.
- Graph Isomorphism Network with Edge features (GINE): To integrate both **node and edge information**, we use a GINE [6] with the following updating rule of node v at layer $t > 0$:

$$h_v^{(t)} = h_v^{(t-1)} + \mathrm{MLP}^{(t)}\left((1 + \epsilon^{(t)}) \cdot h_v^{(t-1)} + \sum_{u \in \mathcal{N}(v)} \mathrm{ReLU}\left(h_u^{(t-1)} + w_{u,v} \right) \right), \tag{2}$$

where $\mathcal{N}(v) := \{u \in V \mid (v, u) \in E(G)\}$.

5. Experimental setup

Data:

We use EUPPBench, a benchmark dataset for ensemble post-processing [7].

- Forecasts and reforecasts from 122 weather stations across Europe
- 730 daily operational forecasts with 51 ensemble members and 4180 reforecasts with 11 ensemble members and a total of 31 variables each

Predict the total precipitation in [mm] accumulated over six hours.

Evaluation metrics:

$$\mathrm{CRPS}(F, y) := \int_{-\infty}^{\infty} (F(x) - \mathbb{1}_{y \leq x})^2 \, dx,$$

$$\mathrm{Brier}(F, y) := (F(0.01) - \mathbb{1}_{y \leq 0.01\mathrm{mm/6h}})^2,$$

$$\mathrm{QS}_{0.99}(F, y) := (\mathbb{1}\{y \leq F^{-1}(0.99)\} - 0.99) \cdot (F^{-1}(0.99) - y).$$

6. Results

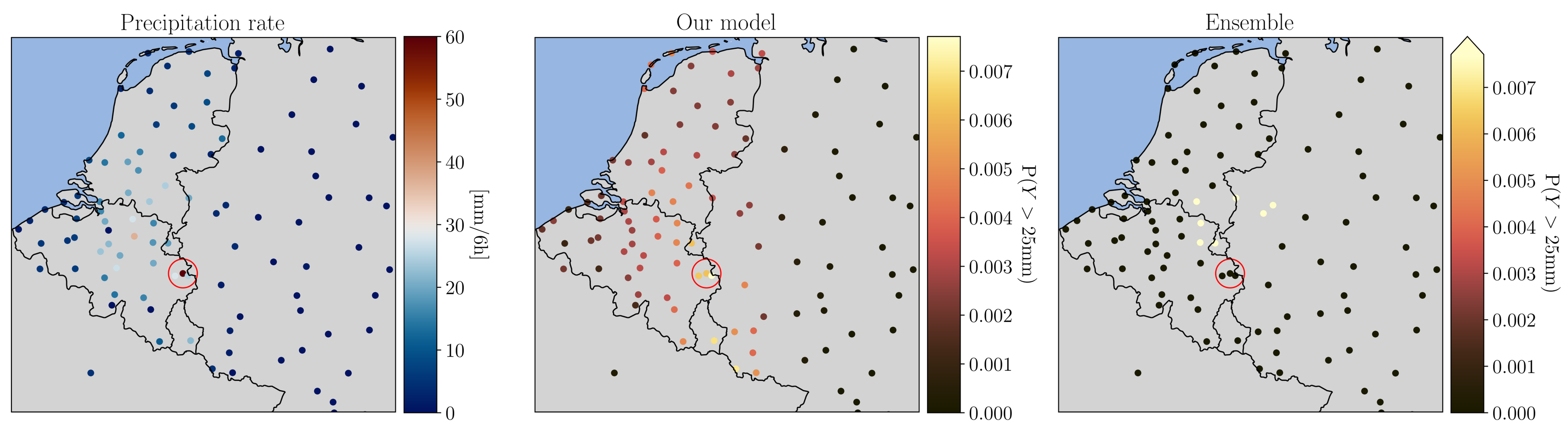

Figure 1: Threshold exceedance probability $\mathbb{P}[Y > 25\text{mm}]$ for our model and the ensemble prediction on an extreme precipitation event (April 29, 2018).

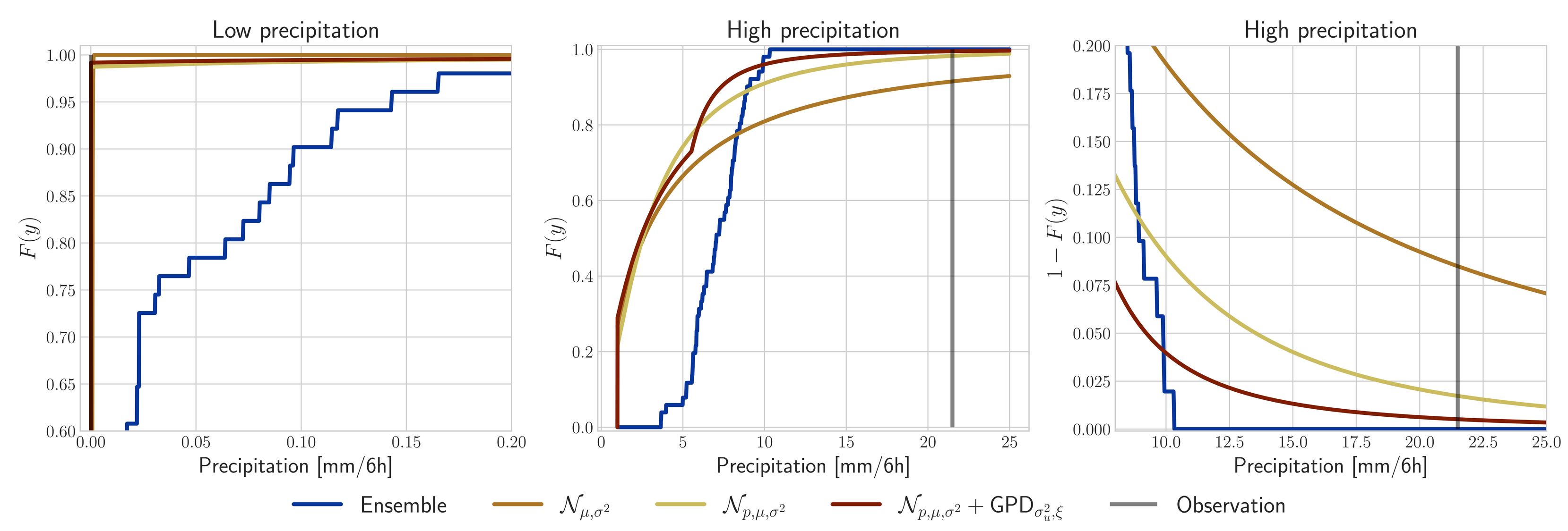

Figure 2: Sample prediction of our proposed approach, explicitly modeling low precipitation scenarios and tail behavior over a certain threshold.

Model	24h			72			120h		
Metric	CRPS	Brier	$QS_{0.99}$	CRPS	Brier	$QS_{0.99}$	CRPS	Brier	$QS_{0.99}$
ENS	0.662	0.180	0.108	0.699	0.180	0.106	0.797	0.200	0.117
$\mathcal{N}_{\mu,\sigma^2}$	0.515	0.316	0.299	0.640	0.384	0.381	0.782	0.337	0.558
$\mathcal{N}_{p,\mu,\sigma^2}$	**0.467**	**0.092**	**0.077**	**0.569**	**0.114**	**0.092**	0.682	0.139	0.117
$\mathcal{N}\text{-GPD}_{u,\sigma_u^2}$	**0.467**	**0.092**	0.082	0.597	0.119	0.099	**0.678**	**0.137**	**0.113**

- Accounting for zero precipitation greatly **improves performance**, also compared to plain ensemble
- Additional GPD modeling shows benefits mainly for **long forecasting times**

7. Summary / Outlook

- Modeling precipitation rate and corresponding extremes directly improves probabilistic predictions
- GPD approach shows improved performance for long prediction times, could potentially be improved further
- Future work: Improve GPD modeling, use the method directly on forecasting tasks

References

[1] S. Rasp, S. Lerch. Neural networks for postprocessing ensemble weather forecasts (2018). Monthly Weather Review, 146(11):3885 – 3900.
[2] T. Gneiting, M. Katzfuss. Probabilistic forecasting (2014). Annu. Rev. Stat. Appl., 1(1):125–151.
[3] B. Kedem, L. Chiu, G. North. Estimation of mean rain rate - application to satellite observations (1990). J. Geophys. Res, 95.
[4] M. Feik, S. Lerch, J. Stühmer. Graph neural networks and spatial information learning for post-processing ensemble weather forecasts (2022), ICML.
[5] M Zaheer, et al. Deep sets (2017). NeurIPS.
[6] K. Xu, et al. How powerful are graph neural networks? (2019). ICLR.
[7] J. Demaeyer, et al. The EUPPBEnch postprocessing benchmark dataset v1.0 (2023). Earth System Science Data, 15(6):2635–2653.