Assignment 2 - Climate forecasting

TU Wien 191.021 Introduction to Computational Sustainability, 2025 Winter Semester, Group 15

Daniel Martin Pühringer
Matr.Nr.: 01556470
e01556470@student.tuwien.ac.at

Patrick Ennemoser
Matr.Nr.: 51913717
e51913717@student.tuwien.ac.at

Dragana Sunaric
Matr.Nr.: 11814569
e11814569@student.tuwien.ac.at

***Abstract*—Accurate short-term precipitation forecasting is essential for climate adaptation but is often limited by the high computational cost of traditional numerical models. In this work, we evaluate four machine learning approaches—ridge regression, a feedforward neural network, an LSTM, and WaveNet—for next-day precipitation prediction in Central Europe using the LamaH dataset. The results show that WaveNet achieves the best predictive performance, particularly after hyperparameter optimization, demonstrating that convolutional sequence models are a promising, efficient alternative for data-driven climate forecasting.**

I. Introduction

Climate change is intensifying the frequency and severity of extreme weather events, making reliable forecasting essential for effective climate adaptation [1]. Precipitation is particularly important due to its impact on hydrology, agriculture, water resources, and flood management. Accurate forecasts help protect infrastructure and support informed decision-making.

Traditional climate models rely on physics-based numerical simulations, which, although accurate, require significant computational resources and are often too slow for real-time or high-resolution local forecasting. Machine learning (ML) offers a fast, scalable alternative by learning patterns directly from historical data, enabling efficient localized precipitation predictions. [2]

This project investigates how different models effectively forecast next-day precipitation in Central Europe. We use the LamaH dataset, which includes over 35 years of daily and hourly meteorological time series from 859 catchments, along with more than 60 environmental attributes. Its large spatial and temporal variability makes it well suited for evaluating ML-based forecasting methods. [3]

The following models were evaluated in this project:

- **Ridge Regression**
- **Feedforward Neural Network (FNN)**
- **LSTM**
- **WaveNet**

The results show that among the evaluated models, WaveNet achieved the best next-day precipitation forecasts, with the optimized version reaching the lowest RMSE, followed closely by the feedforward neural network, while ridge regression performed moderately and the LSTM lagged behind. Overall, convolutional sequence models proved most effective for capturing the temporal patterns in the LamaH precipitation data.

The full source code for this evaluation is available on Github:
Assignment 2

II. Background

The following section provides the background needed to understand the experiments.

A. Background on used ML-models

The project initially used four Machine Learning approaches and optimizes the best approach. This section will explain the used ML-models as well justify their usage for this project. We used one basic model (ridge regression) and three Neural Networks (NNs)

1) Ridge Regression: Ridge regression, also known as regularized linear regression [4], is one of the simplest machine learning approaches. Unlike the other models used in this project, ridge regression is not a Neural Network. It can be understood as an advanced form of linear regression that addresses the problem of overfitting. Standard linear regression models often perform exceptionally well on training data but fail to generalize to unseen data due to having too many parameters, which makes them unstable. This problem is also called overfitting. Ridge regression mitigates this issue by introducing a regularization term that penalizes large coefficients, thereby preventing the model from becoming overly complex.

In this project, ridge regression was used as a simple baseline model. Compared to neural networks, ridge regression requires significantly fewer computational resources, resulting in lower energy consumption and reduced carbon emissions. Its inclusion highlights the trade-off between model accuracy and resource usage: achieving higher precision typically requires more complex architectures and more computationally intensive training processes, which come at increased cost.

2) Feedforward Neural Network (FNN): The FNN [5] is the simplest type of NN. In an FNN, information flows strictly in one direction from the input layer through one or more hidden layers to the output layer, without any cycles or feedback loops. This distinguishes FNNs from recurrent architectures such as RNNs or LSTMs.

FNNs are well suited for tasks that require straightforward, one-directional data processing. Their architecture typically consists of a multilayer perceptron (MLP) structure, comprising an input layer, several hidden layers, and an output layer, and they are trained using backpropagation. In this project, the FNN serves as the simplest neural network baseline, demonstrating the fundamental capabilities of neural networks beyond linear models.

3) Long Short-Term Memory (LSTM): Long Short-Term Memory networks (LSTMs) [6] are a specialized form of recurrent neural networks (RNNs). RNNs are particularly well suited for sequential data, where the order of observations provides valuable information for prediction. This property makes them appropriate for the dataset used in this project. Unlike feedforward networks, RNNs incorporate a form of memory by feeding information back into the network, allowing them to model temporal dependencies, as commonly seen in applications such as speech recognition.

However, RNNs are more difficult to train than FNNs due to their recurrent structure, which can lead to vanishing or exploding gradient problems. LSTMs address these challenges through a gated architecture that replaces repeated matrix multiplications with additive operations, enabling more stable gradient flow. As a result, LSTMs significantly improved the performance of sequence modeling tasks and played a key role in the advancement of speech recognition technologies, serving as a conceptual predecessor to more advanced architectures such as WaveNet.

4) WaveNet: WaveNet [7] was one of the first artificial intelligence models capable of generating natural-sounding speech. Although originally developed for audio generation, WaveNet has proven to be highly effective at modeling complex, non-linear dependencies across multiple timescales, making it well suited for time-series data.

WaveNet employs causal convolutions and a deep, hierarchical structure that allow it to capture temporal patterns more effectively than feedforward neural networks. Compared to LSTMs, WaveNet is particularly strong at modeling very long-range dependencies, which recurrent architectures often struggle to maintain. By relying on convolutional neural networks rather than recurrent connections, WaveNet was designed to overcome key limitations of the LSTM paradigm, offering a powerful alternative for sequence modeling tasks.

B. Comparison of approaches

Table I shows a structured comparison of the four evaluated machine learning approaches in terms of model type, temporal awareness, representational capacity, memory characteristics, training speed, and predictive performance.

C. Climate Modeling and ML-based forecasting

Climate models are computational representations of Earth's atmosphere and hydrological systems that simulate interactions between climate variables. They are essential for understanding and predicting weather and long-term climate patterns. Traditional numerical models solve complex physical equations, but this requires significant computational resources. As a result, these models are slow, energy-intensive, and often unable to provide high-resolution or real-time forecasts. [8]

Machine learning offers a complementary, data-driven alternative. ML models can learn patterns directly from historical observations and generate predictions much faster and with far lower energy consumption. Systems such as FourCastNet [2] demonstrate that ML can achieve competitive accuracy while being highly scalable and computationally efficient.

However, ML approaches in climate science also face challenges. Available datasets are limited in historical depth, often noisy, and geographically uneven. Biases arise from incomplete coverage, and models must be carefully designed to ensure robustness. Additionally, many ML models lack interpretability, making it difficult to understand the physical meaning behind their predictions.

These limitations and opportunities motivate the exploration of ML methods for climate forecasting, including the next-day precipitation prediction task studied in this project.

D. Dataset

This project uses the LamaH (Large-Sample Data for Hydrology and Environmental Sciences) dataset [3], a comprehensive hydrological and meteorological dataset covering Central Europe. LamaH contains data from 859 gauged catchments across more than 170,000 km², spanning diverse landscapes from lowland regions to high alpine areas.

The dataset includes over 35 years of daily and hourly meteorological time series, such as precipitation, temperature, and runoff measurements, along with more than 60 catchment attributes describing topography, land cover, soil, vegetation, and climate characteristics. This combination of long-term temporal records and rich spatial features makes LamaH well suited for machine learning–based forecasting. Its variability and high-resolution coverage provide a robust foundation for training and evaluating models that aim to predict next-day precipitation across Central Europe.

Comparison table	Ridge Regression	FNN	LSTM	WaveNet
Type	Statistical (i.e. Linear)	Simple Neural Network	Recurrent Neural Network	Convolutional Neural Network
Awareness of time	No awareness	No awareness	Yes (Sequential time)	Yes (Sequential time)
Type of relationships	Linear	Non-Linear	Complex Non-Linear	Complex Non-Linear
Memory	Fixed (Input window)	Fixed (Input window)	Dynamic (Long term)	Dynamic (Very long term)
Training Speed	Very fast	Fast	Slow	Fast (due to Parallelization)
MSE RESULTS (TEMP)	Ridge RMSE: 0.1305339655250115	FNN: DNN RMSE: 0.0958	LSTM RMSE: 0.8976 (2 epchs); 0.8228(10 epochs); 0.7880 (10 epochs, better architecture	WaveNet RMSE: 0.09304463230544957 (20 epochs)

TABLE I: Comparison of used ML approach based on various dimensions.

An overview of the meteorological variables used in this work, including their daily aggregation methods and units, is provided in Table II.

E. Models

III. Data Analysis

To analyse the LamaH meteorological data, we randomly selected 100 location files from the dataset and merged them into a single dataframe for exploration and preprocessing. The key steps of the data analysis are summarized below.

A. Null Value Analysis

The null-value scan showed that for the 100 location files we chose, none of the variables contained missing values, meaning the selected LamaH meteorological time series are fully complete for the sampled catchments.

B. Correlation Analysis

To assess the linear relationships between all predictor variables and the target variable (prec), we computed the feature–target correlations across the merged dataset. The analysis highlights which meteorological and environmental variables exhibit the strongest influence on precipitation.

The most positively correlated features were:

- *10m_wind_u*: Wind speed (towards east, 10m above surface)
- *2m_dp_temp_max/mean/min*: max/mean/min dew-point temperature (2m above surface)
- *lai_high_veg*: One-sided leaf area (per square meter of ground surface) for tall vegetation

These variables are intuitively linked to precipitation, as moisture availability (dew point), vegetation-driven evapotranspiration (LAI), and wind dynamics can contribute to atmospheric conditions favorable for rainfall.

The strongest negative correlations were observed for:

- *surf_net_therm_rad_mean/max*: surface net thermal radiation (mean and max)
- *surf_net_solar_rad_mean/max*: surface net solar radiation (mean and max)
- *surf_press*: surface pressure

Lower radiation values and decreasing surface pressure are typically associated with cloud formation and incoming storm systems, explaining their inverse relationship with precipitation.

Figure 1 shows a heatmap of all feature–target correlations, providing a visual summary of these relationships.

C. Naive Baseline Forecasts

To evaluate the performance of more complex forecasting models, we first establish two naive baselines: a Moving Average (MA) forecast and a lag-1 (t–1) forecast.

a) Moving Average (MA): A moving average forecast predicts the next value by taking the average of the previous n observations, thereby smoothing short-term fluctuations. [9] For our baseline we chose *n=3* We observed that increasing the window size leads to a higher Mean Squared Error (MSE), as the forecast becomes less responsive to recent changes in the data.

b) t–1 Forecast: A t–1 forecast simply uses the previous time step's value as the prediction. This approach often performs surprisingly well because many natural processes (e.g., temperature, precipitation, river flow, electricity load) exhibit strong autocorrelation—meaning the current state is highly similar to the recent past. As a result, a t–1 baseline can outperform more complex models on short prediction horizons.

c) Forecast results: To evaluate the predictive performance of the naive baseline models, we computed the Mean Squared Error (MSE) for both forecasting approaches. The t–1 (lag-1) forecast achieved an MSE of 8.06, indicating that simply using the previous value provides a reasonably accurate short-term prediction given the strong autocorrelation present in the data. In comparison, the Moving Average (MA)

Variable (hourly)	Daily Aggregation	Description	Unit
DOY	unchanged	Day of year	–
HOD	omitted	Hour of day	–
2m_temp	max, mean, min	Air temperature at a height of 2 m above the Earth's surface	°C
2m_dp_temp	max, mean, min	Dew point temperature at a height of 2 m above the Earth's surface	°C
10m_wind_u	mean	Horizontal wind speed towards the east at a height of 10 m above the Earth's surface	$\mathrm{m\,s^{-1}}$
10m_wind_v	mean	Horizontal wind speed towards the north at a height of 10 m above the Earth's surface	$\mathrm{m\,s^{-1}}$
fcst_alb	mean	Forecast albedo: fraction of solar shortwave radiation reflected by Earth's surface (direct and diffuse)	–
lai_high_veg	mean	One-half of the total green leaf area per unit horizontal ground surface area for high-vegetation type	–
lai_low_veg	mean	One-half of the total green leaf area per unit horizontal ground surface area for low-vegetation type	–
swe	mean	Water equivalent of snow	mm
surf_net_solar_rad	max, mean	Net shortwave radiation at the Earth's surface (incoming minus reflected), positive values indicate radiation to the surface	$\mathrm{W\,m^{-2}}$
surf_net_therm_rad	max, mean	Net thermal radiation at the Earth's surface; positive values indicate radiation emitted from the Earth	$\mathrm{W\,m^{-2}}$
surf_press	mean	Surface pressure	Pa
total_et	sum	Total evapotranspiration; positive values indicate evapotranspiration, negative values condensation	mm
prec	sum	Total amount of precipitation (liquid and frozen)	mm
volsw_123	mean	Fraction of water in topsoil layer (0–100 cm depth)	$\mathrm{m^3\,m^{-3}}$
volsw_4	mean	Fraction of water in subsoil layer (100–289 cm depth)	$\mathrm{m^3\,m^{-3}}$

TABLE II: Overview of variables included in the LAMAH dataset, their daily aggregation methods, descriptions, and units. [3]

forecast with *n=3* resulted in a slightly higher MSE of 9.61, reflecting the fact that averaging over multiple past values smooths the signal but can reduce responsiveness to rapid changes. These results show that, for this dataset, the t–1 baseline provides the better naive forecast.

To visually compare the models, the following plot displays the actual values alongside the t–1 and MA forecasts for the last 100 timestamps, illustrating how each baseline follows (or lags behind) recent variations in the data.

IV. Experiments and Results

A. Model Predictions Comparison

This section shows predictions of four different models compared to true values.

Figure 3 shows the predicted versus true values for all evaluated models on the test set, while Figure 4 reports the corresponding root mean squared error (RMSE).

The Ridge regression model yields smooth predictions centered around the mean of the target variable and achieves an RMSE of 0.1305. The feedforward neural network improves prediction accuracy, capturing larger variations in the signal and reducing the RMSE to 0.0958. The WaveNet model further improves alignment with the true sequence, achieving an RMSE of 0.0930. The optimized WaveNet configuration attains the lowest error overall, with an RMSE of 0.0893.

The LSTM model performs noticeably worse than the other neural architectures, with an RMSE of 0.1647. This is reflected in the prediction plot, where the model shows weaker alignment with the true signal compared to the feedforward and convolutional sequence models.

Overall, the results indicate that the optimized WaveNet model provides the best predictive performance among the evaluated approaches.

B. Finetuning WaveNet

Since figure 4 shows WaveNet as the best performing model, we optimized the hyperparameters of WaveNet to study the best achievable results. As stated in the lecture, we included

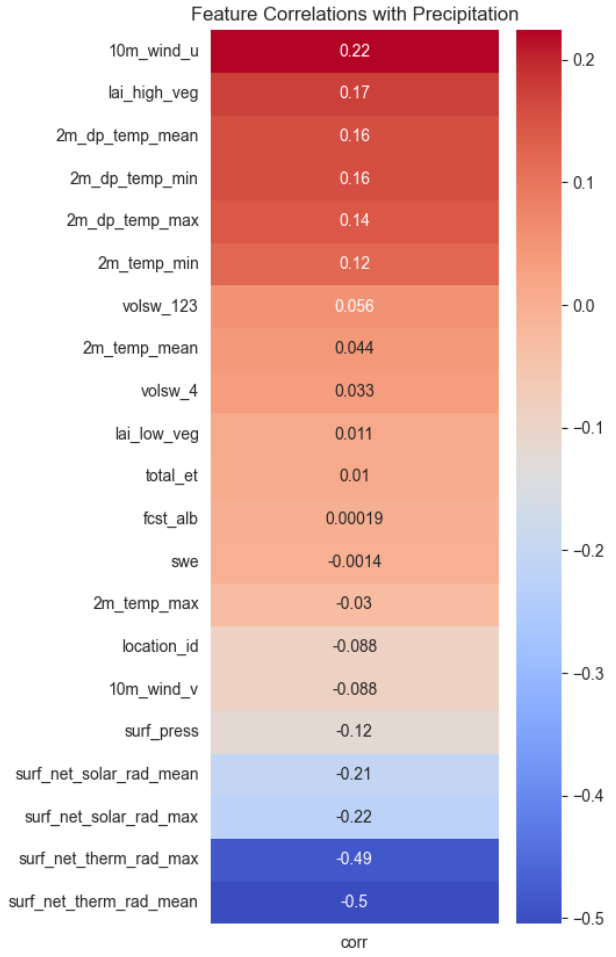


Fig. 1: Feature Importance: Correlation heatmap of all features with the precipitation target variable.

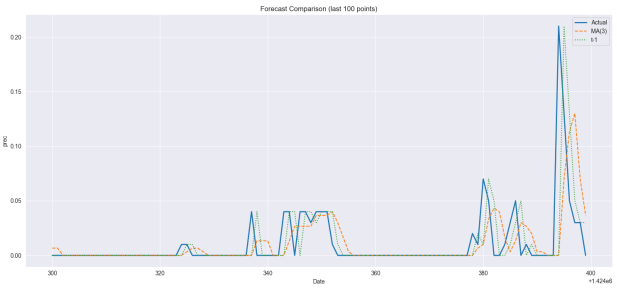
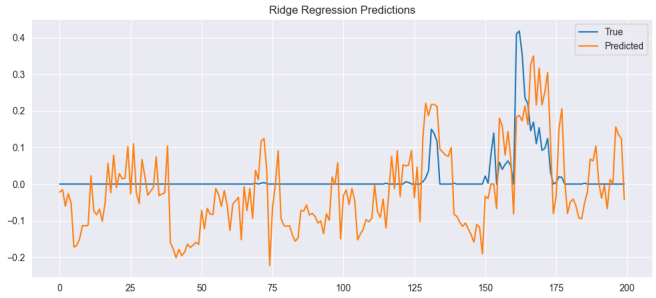


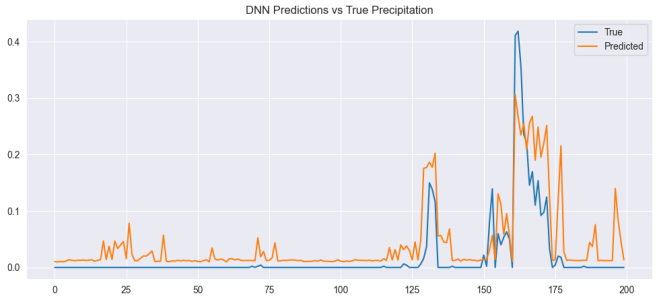
Fig. 2: Comparison of actual values with the t–1 and Moving Average (MA, n=3) forecasts for the last 100 timestamps

architectural changes as well in this optimization process. The resulting boost in performance can be seen in figure 4, where the optimized version achieves significantly better results than the original WaveNet implementation. We implemented three main changes to achieve this:

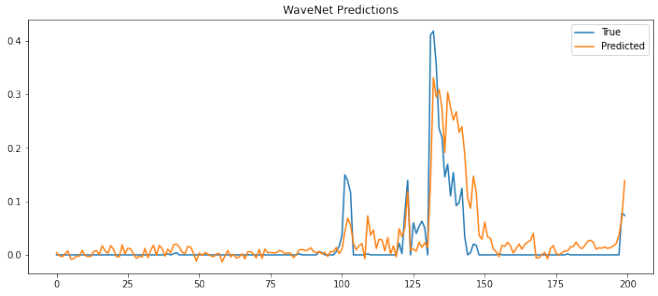
1) **Improved layer architecture**: The first improvement was done by changing one important layer of our NN. Instead of using *layers.Flatten()*, which seemed very



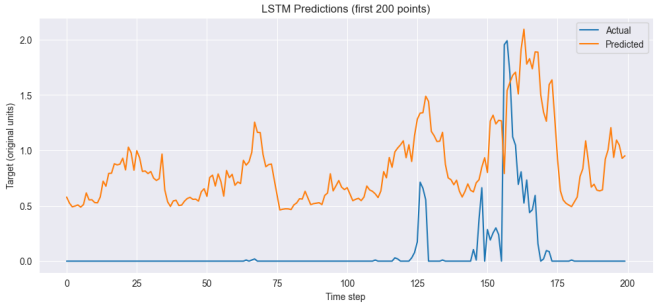
(a) Ridge Regression Predictions



(b) Feedforward Network (FFN) Predictions



(c) WaveNet Predictions



(d) LSTM Predictions

Fig. 3: Comparison of predicted vs true values for four different models. All plots show the first 200 points of the test set.

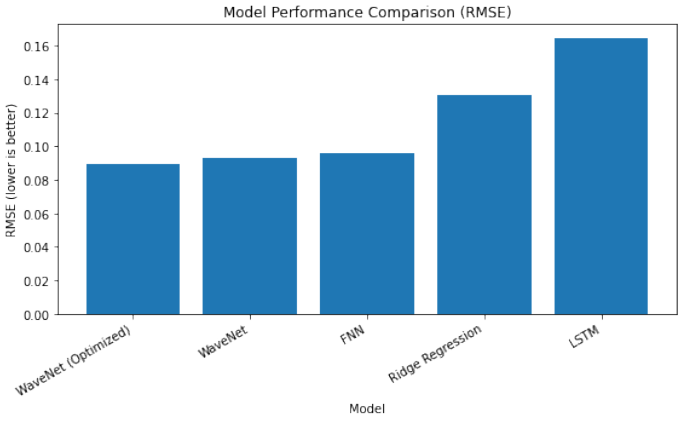


Fig. 4: Comparison of model performance using RMSE. Models are sorted from lowest to highest RMSE, where lower values indicate better predictive performance.

logical in the initial implementation, the optimized version uses *layers.GlobalAveragePooling1D()(x)* instead of *layers.Flatten()*. *Flatten()* turns [time, channels] into a larger vector which can result in making the model more sensitive to sequence length. This sequence length is important and therefore this change was a significant improvement for this time-series regression. *GlobalAveragePooling* is more flexible and reduces parameter count.

2) **Improved loss function**: The second improvement was done by changing the loss function. Initially, we used the "Mean Squared Error" loss function, which is simple and commonly used. We switched to "Huber loss" since this loss is known to be more stable with large outliers [10].
3) **Increased number of epochs with early stopping**: Our last improvement was to increase the amount of epochs. This seemed to be the most straight-forward approach for any hyperparameter tuning, however we wanted to be conscious of sustainability and the increased computational costs high amount of epochs entail. As a rule of thumb, an increase of epochs is expected to result in better performing models, however due to diminishing returns many epochs are not providing any significant improvement at a certain level and are a waste of energy. So finding a suitable amount of epochs was important to us. This is what our last optimization does: We added a callback so that the model runs 200 epochs by default, however as soon as no significant improvement is expected (because a plateau was reached), the model training terminates early and keeps the best result. In our case, this was 45 epochs.

Figure 5 shows the results of these three optimizations. The RMSE reduced from *0.093* (before hyperparameter tuning) to *0.0893* after hyperparameter tuning. This means that the accuracy of the model after the hyperparameter tuning was improved by approx. 3.9785%.

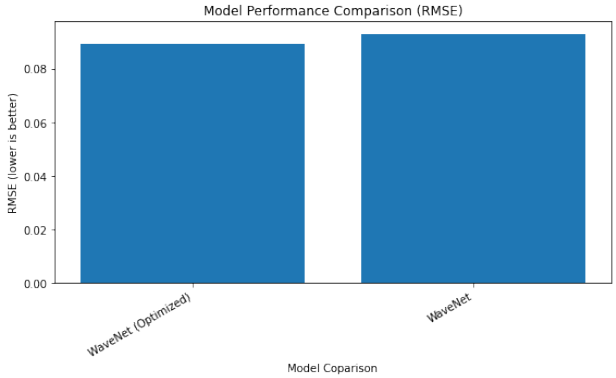


Fig. 5: Accuracy of the WaveNet model before and after hyperparameter tuning

V. Conclusions

In this assignment, we addressed the problem of climate forecasting by benchmarking Ridge Regression, FNN, LSTM, and WaveNet. The experiments revealed that all of those four machine learning approaches provide huge value in climate forecasting, offering a level of precision that far exceeds naive baselines: The transition from a naive MSE (from a t-1 Forecast) of *8.06* to a learned model RMSE of *0.0893* represents a fundamental leap (**90-fold**) in predictive accuracy, validating the use of the LamaH dataset for training complex forecasting systems.

Among the evaluated models, WaveNet emerged as the superior architecture, achieving an RMSE of 0.0930 in its base form and further improving to 0.0893 after hyperparameter optimization. This tuning of hyperparameters resulted in an improvement by approx. 4%. While the FNN performed surprisingly well, the LSTM struggled to capture long-term dependencies as effectively as the convolutional approach (WaveNet). Ultimately, this project confirms that convolutional sequence models like WaveNet are highly effective for climate forecasting.

References

[1] P. Stott, "How climate change affects extreme weather events," *Science*, vol. 352, no. 6293, pp. 1517–1518, 2016.
[2] J. Pathak, S. Subramanian, P. Harrington, S. Raja, A. Chattopadhyay, M. Mardani, T. Kurth, D. Hall, Z. Li, K. Azizzadenesheli *et al.*, "Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators," *arXiv preprint arXiv:2202.11214*, 2022.
[3] C. Klingler, K. Schulz, and M. Herrnegger, "Lamah— large-sample data for hydrology and environmental sciences for central europe," *Earth System Science Data Discussions*, vol. 2021, pp. 1–46, 2021.
[4] A. E. Hoerl and R. W. Kennard, "Ridge regression: applications to nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 69–82, 1970.
[5] I. J. Goodfellow, M. Mirza, D. Xiao, A. Courville, and Y. Bengio, "An empirical investigation of catastrophic forgetting in gradient-based neural networks," *arXiv preprint arXiv:1312.6211*, 2013.
[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[7] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[8] J. Sun, M. Xue, J. W. Wilson, I. Zawadzki, S. P. Ballard, J. Onvlee-Hooimeyer, P. Joe, D. M. Barker, P.-W. Li, B. Golding *et al.*, "Use of nwp for nowcasting convective precipitation: Recent progress and challenges," *Bulletin of the American Meteorological Society*, vol. 95, no. 3, pp. 409–426, 2014.

[9] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*. OTexts, 2018.

[10] G. for Geeks, "Huber Loss Function in Machine Learning," https://www.geeksforgeeks.org/machine-learning/huber-loss-function-in-machine-learning/, 2025, accessed: 2025-12-14.