Precipitation time series prediction using wavelet-based transformers

Category: Others (Climate Forecasting/Transformer)

Hanbing Chen (hanbingc), Mofan Zhang (mofanz), Xueqi Chen (xqc1998)

1. Introduction

A long-term accurate prediction of precipitation time series is essential for water resources planners to tackle critical problems such as drought management, flood control, irrigation, structural design and eco-hydrological services ahead of time. However, precipitation is an extremely complex process and hard to predict, as it evolves in chaotic climate systems and interplays between numerous weather and climate variables. Climate change introduces even more uncertainty. Thus, precipitation time series often show significant non-linearity, and/or non- stationarity and have short-range and long-range time dependency [1]. Such characteristics are difficult to accommodate in traditional statistical methods and machine learning techniques.

Numerous attempts have been made in literature to find the most appropriate model for predicting precipitation time series. Statistical models (such as Exponential Smoothing, AutoRegressive Integrated Moving Average model (ARIMA)) have been criticized for their effectiveness for capturing the complex nonlinear and nonstationary time series[4]. In recent decades, many studies have used artificial neural networks as forecasting approaches. Among them, Recurrent Neural Network (RNN) and its variant, called Long short-term memory (LSTM) , capture intensive attention for their ability for handling non-linearity effective memory of the network[5]. However, RNN-based models often suffered from modeling both short- and long-range dependencies and time-consuming to train the model[6]. In addition, introduced in a 2017 paper[2] by Google, the **Transformer** model is a state-of-the-art neural network architecture that learns the context by tracking the relationship in sequential data. The key feature is the “self attention” mechanism, which could draw the dependencies of different positions of the sequence to generate a richer representation of a given input step. These mechanisms make it suitable for time series analysis for : a) Transformer captures the long and short dependencies in the time series as it represents each step input while considering its context, b) has faster computation compared to RNN-based model because it eliminates recurrence and enables parallelization. However, there is still some remaining challenge for Transformer to effectively represent the non-stationary climate time series with quasi-oscillatory behaviors at various scales. Furthermore, it is usually difficult to tune the parameters and interpret the results for complex architectures as Transformer.

In this study, inspired by a hybrid approach[7], we utilized a **wavelet-based transformer** model for long-term (decadal ahead) monthly precipitation time series forecasting. Decomposition techniques (such as Fourier transform and Wavelet transform) from signal processing are effective for decomposing the time series into several simpler components, which we hypothesized to be easier for the Transformer model to learn the inherent pattern. In our model, we take the advantage of wavelet transform (WT) to handle nonlinear signals at various scales ranging from slow changing, low frequency to fast changing, high frequency.

We compare the accuracy of predicting monthly precipitation time series with our WT-transformer to other popular techniques: LSTM, ARIMA, NBEATSModel, Exponential smoothing,LightGBM, vanilla transformer, based on some common metrics such as RMSE, MAPE, etc. We are interested in whether using wavelet transform as a data preprocessing technique will provide useful information for the transformer and will improve prediction accuracy. Additionally, we would like to identify the influential hyperparameter sets to provide some insights for future work using Transformer for precipitation forecasting.

2. Method

a. Data Collection & Train/Test Split

To test the performance of our proposed WT-transformer model, we obtain a 251-year monthly precipitation time series (1850 - 2100) from a publicly available climate model projection: CMIP6 climate scenario of FIO-ESM-2-0. As shown in **Fig1**, The pattern of monthly precipitation shows great irregularity, variability, and recurrence. The training/testing set split for this model was not done through any conventional proportion due to the unique nature of our raw dataset. Precipitation has seasonal cycles. Therefore, it would be unreasonable to select a random time in a year to be the starting point of our training data. Instead, the testing data is selected to be the last 20 years of monthly data in our dataset, which is approximately 10% of the total length of the original time series.

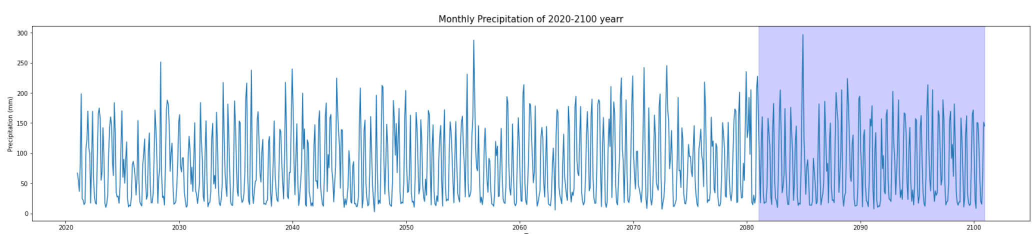


Fig. 1. Monthly precipitation of the last 80 years (2020-2100 year). With test set highlight

b. Wavelet-based Transformer model

Here, we used an advanced signal processing approach as a data-preprocessing technique before Transformer: Discrete wavelet transform (DWT) to decompose the original complex signal into several simpler component series. These component series include several detailed representations (high frequency) , low frequency, and a smooth component (trend) of the original time series. Then each level of the component series is fed into an independent Transformer to make predictions. To be more specific, we used an iterative approach for multi-step ahead prediction. At last, we implemented the reverse DWT to obtain the full time series predictions from predicted components.

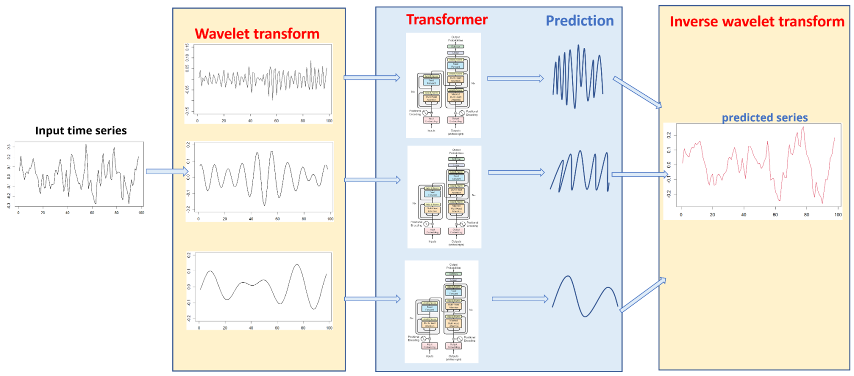


Fig. 2. Method schematic for WT-transformer

c. Simpler NN baselines

We started by building other Neural Networks that have simpler model structures to understand common challenges in time-series . One observation of the results from these models is that they tend to capture high frequency patterns well but fail to predict the irregularities in the time series, such as sudden high spikes of precipitation. More complicated NNs such as LSTM and N-BEATS Model were able to reproduce some of the noises in the original time series. These models also serve as references for evaluating prediction performance of the W-transformer. Results see in the following section.

d. Hyperparameter Tuning

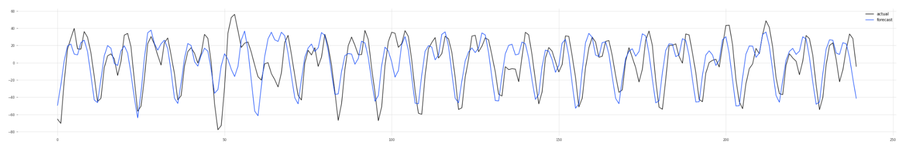
After setting up a baseline transformer model, we performed hyperparameter tuning through a grid search approach, where we tried a set of hyperparameter values while fixing others for cross-validation. After careful examination and understanding of the adjustable hyperparameters, we identified several significant hyperparameters to tune first, based on the characteristics of our model.

Input_chunk_length is one of the most influential hyperparameters in our model. It specifies the size of the look-back window during each training epoch. After several experiments, we found the model was able to produce good predictions for high-frequency level wavelet decompositions but extremely performed poorly on low frequency wavelets (See Appendix, figure A.8), using a constant input chunk size. We speculated that this could be caused by insufficient look-back window size. Low frequency components require larger look-back windows as the time dependency is longer. The input chunk length that can capture high frequency patterns may only capture a linear portion of the low frequency wavelets since they are much more stretched. This speculation is confirmed by the drastic improvement in low frequency predictions after we implemented an increased input chunk length for low frequency components. This is tuned further by training one wavelet component at a time and finding an optimal chunk length. The output_chunk_length was kept at half of the input_chunk_length.

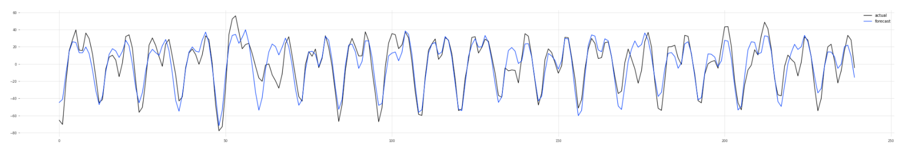
Taking advice from the teaching team as well as several research papers, we also tuned the number of encoder/decoder layers of the transformer model. The encoder consists of a multi-head attention layer, which is what made the transformer suitable for this application. Multi-head attention layers allow the model to manage a mix of short-term and long-term dependency information from the time-series. After several experiments, we found 2 layers of both encoder and decoder to be most suitable for our application.

Batch size determines the length of time sequences analyzed during each training pass. The default batch size 32 produced the best outcome compared to base-2 numbers used. For example the level 3 wavelet prediction results are shown in Fig 3 for batch_size=16,32 and 64:

batch size = 16



batch size = 32



batch size = 64

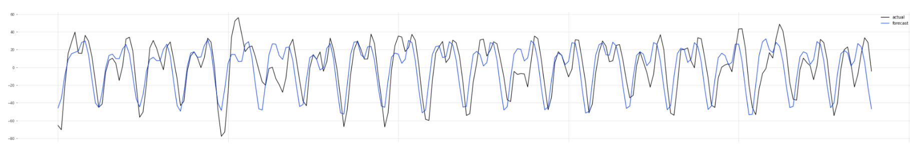


Fig. 3. Prediction vs test data for level 3 wavelet decomposition

The number of epochs are selected by observing the change of validation loss during the training processes while tuning the other parameters. For the lower frequency levels, which are the most difficult to train due to the presence of regions with low gradients, the loss began to plateau around 200 epochs. This is used as the final epoch number for model training.

e. Data Scaling

In addition to hyperparameter tuning, we experimented with different ways of scaling input data, which greatly affected training efficiency. Scaling the training set independently of the testing set was crucial for our model training to be effective because the predicted time series need to be inversely transformed to compare with the testing set. Since only the training data would be the input for the W-transformer, using a scaler that is affected by a part of data that is not considered in the model training(testing data), would make the comparison biased. After wavelet decomposition, each wavelet is fitted and transformed by a separate standard scaler, which is stored. The stored scalers are later used to inversely transform predicted data to convert it back to unscaled precipitation for comparison with testing data.

3. Results:

Table 1 shows the prediction evaluation metrics of the different models., with bolded numbers representing the best performances. The W-Transformer outperformed all other models in MAPE, MAE, SMAPE, and MASE, and only has a slightly larger RMSE compared to the vanilla transformer model. This confirms our hypothesis that the extra information provided by different levels of wavelets can significantly improve the transformer's ability to handle nonstationarities and irregularities of the time series.

Metrics	LSTM	ARIMA	N-BEATS	Exponential Smoothing	LightGBM	Vanilla Transformer	W-Transformer
RMSE	32.9428	34.7632	37.7139	37.5077	32.0246	**29.871**	29.8995
MAPE	0.3858	0.8886	0.4569	1.0239	0.4136	0.4958	**0.0454**
MAE	23.5448	28.0699	26.7883	31.5506	22.3352	22.3765	**22.2004**
SMAPE	199.7946	191.9346	209.5326	194.2304	202.1887	204.8519	**0.3669**
MASE	1.4045	1.4673	1.4249	1.5586	1.4449	1.5402	**0.4989**

Table. 1. Evaluation metrics of model predictions

Although there is still room for improvement in terms of accurately predicting the magnitudes of precipitation, the final W-Transformer model was able to timely capture the pattern, as shown in Fig. 4.

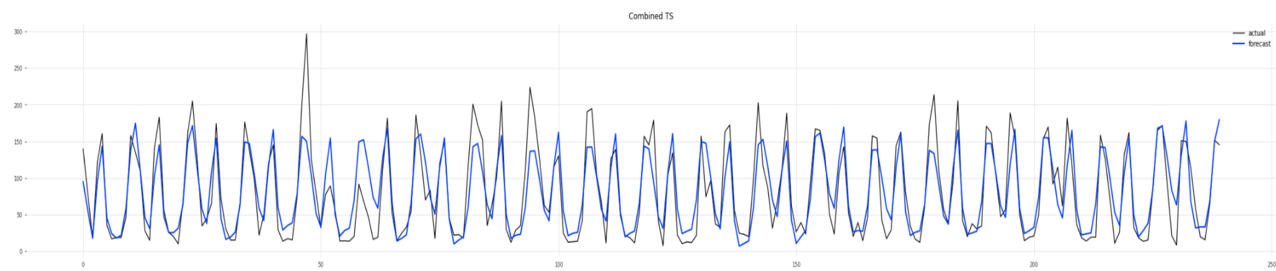


Fig. 4. Predicted precipitation of finalized W-Transformer vs actual precipitation

4. Conclusion and Discussion

In this study, we utilized the advanced wavelet-transformer model to examine the long-term prediction of precipitation time series. Precipitation is a complex atmospheric process and displays non-linearity, non-stationarity, and shows both short- and long- time dependency. Compared with vanilla transformer and other popular time series forecasting techniques, wt-transform achieves the best result for the prediction task, demonstrating using wavelet transform as a data preprocessing approach could reveal more useful hidden patterns before feeding into the transformer model. We also find that by decomposing the complex full time series into several components with different frequency bandwidth, it makes it more intuitive to tune one of the most influential hyperparameters: input-chunk_length and output_chunk_length. As these two hyperparameters are directly related to the time-dependency of time series, high frequency components require smaller input_chunk and vice versa. In complex architectures such as transformers, this could provide clear direction for hyperparameter tuning, making the process less exhaustive. Future work involves testing our model performance on more precipitation time series from a variety of locations, which could show different characteristics. Our model should be able to achieve satisfying generalizability compared to others since the decomposition procedure reveals frequency information of time series and allows customized training. We would also explore the optimal number of wavelet decomposition levels.

Contribution

We contribute equally in developing the models, coding, fine tuning the hyperparameters, and writing up. Xueqi Chen set up the AWS environment, Mofan Zhang and Hanbing Chen collected and preprocessed the data. The cat Gulu accompanied and supervised us.

Acknowledgements

We would like to thank the entire CS230 teaching team for offering us support and expert guidance throughout the quarter. It's been a really enjoyable journey, in which we learned a lot of meaningful insights about Deep Learning topics and industry trends. We look forward to deploying the deep learning techniques we learned in this quarter for our own research area and future professional endeavors.

Reference

[1] Slater, Louise J., et al. "Nonstationary weather and water extremes: a review of methods for their detection, attribution, and management." *Hydrology and Earth System Sciences* 25.7 (2021): 3897-3935

[2] Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems* 30 (2017).

[3] M. Aminghafari and J.-M. Poggi, "Forecasting time series using wavelets," International Journal of Wavelets, Multiresolution and Information Processing, vol. 5, no. 05, pp. 709–724, 2007.

[4] Rathod, S. A. N. T. O. S. H. A., et al. "Modeling and forecasting of oilseed production of India through artificial intelligence techniques." *Indian J. Agric. Sci* 88.1 (2018): 22-27.

[5] Barrera-Animas, Ari Yair, et al. "Rainfall prediction: A comparative analysis of modern machine learning algorithms for time-series forecasting." *Machine Learning with Applications* 7 (2022): 100204.

[6] Lim, Bryan, and Stefan Zohren. "Time-series forecasting with deep learning: a survey." *Philosophical Transactions of the Royal Society A* 379.2194 (2021): 20200209.

[7] Sasal L, Chakraborty T, Hadid A. W-Transformers: A Wavelet-based Transformer Framework for Univariate Time Series Forecasting[J]. arXiv preprint arXiv:2209.03945, 2022.

Appendix

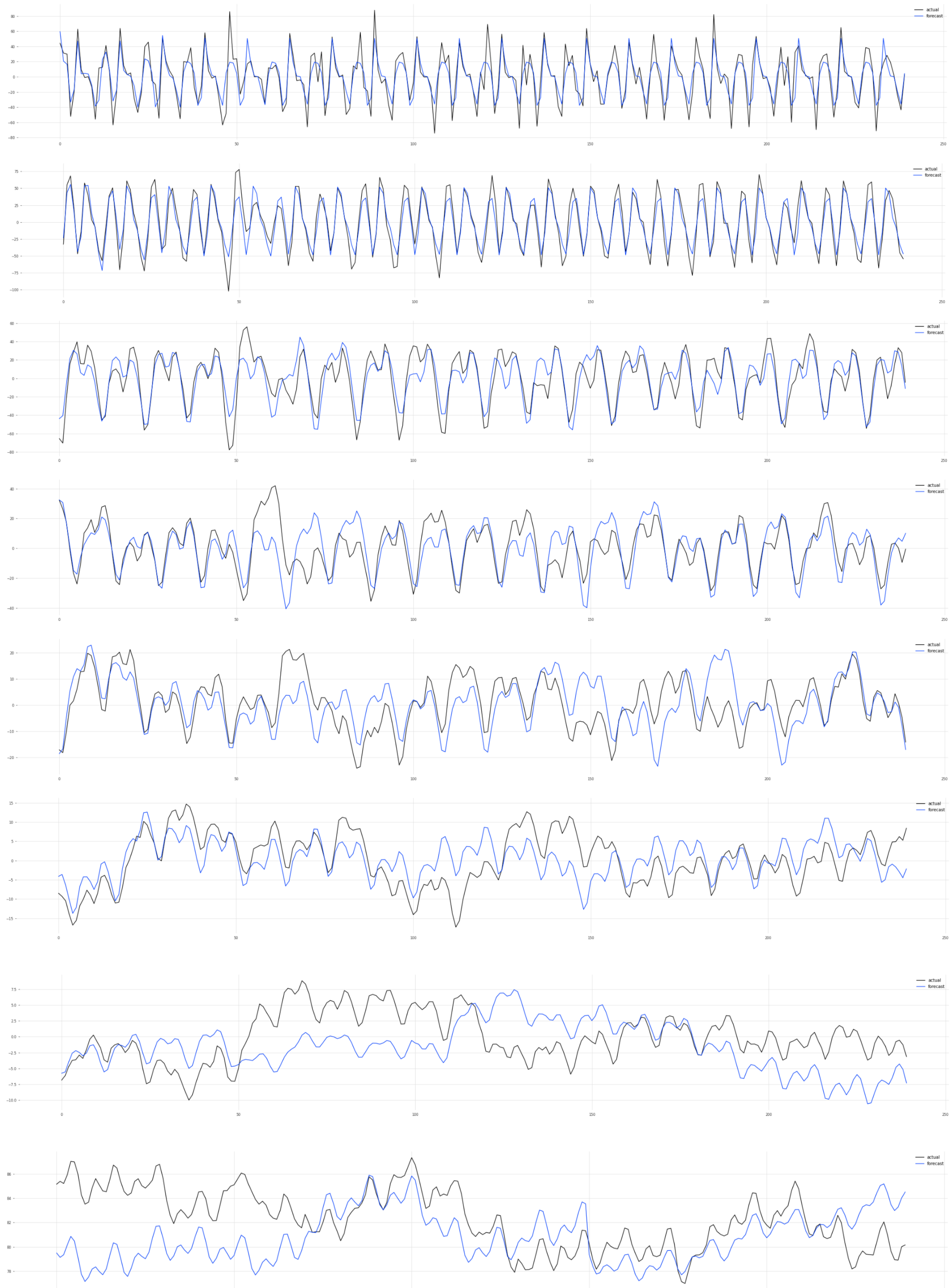


Fig. A1. W-Transformer predicted wavelets vs actual wavelets, for 8-level decompositions

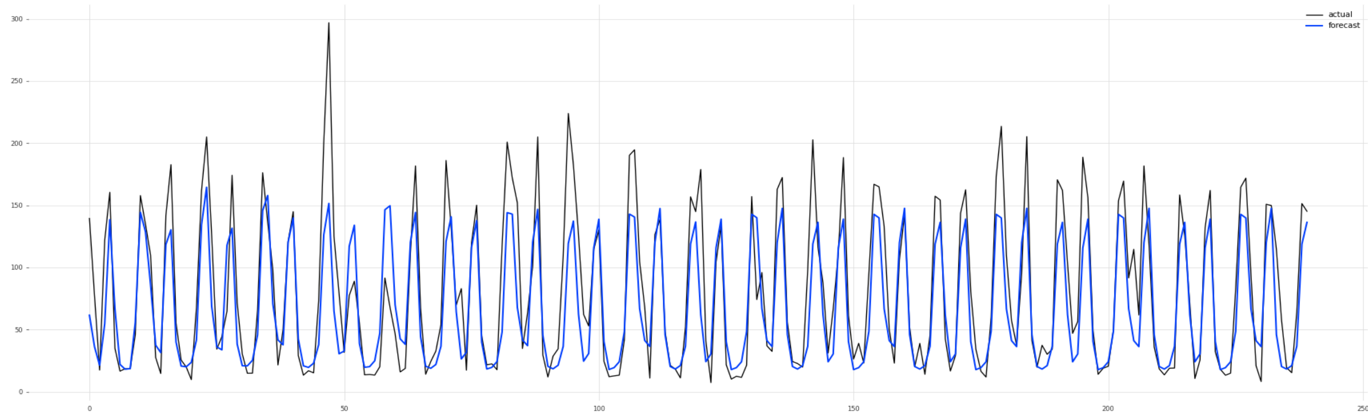


Fig. A2. LSTM predicted precipitation vs actual precipitation

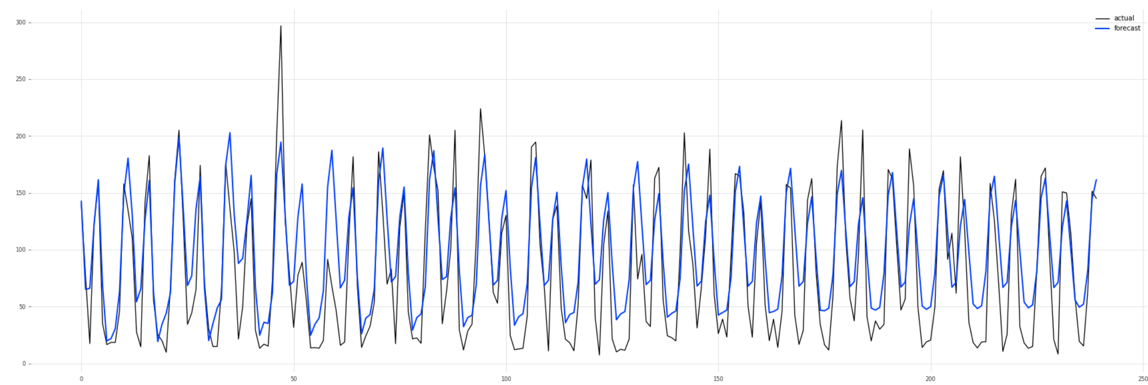


Fig. A3. ARIMA predicted precipitation vs actual precipitation

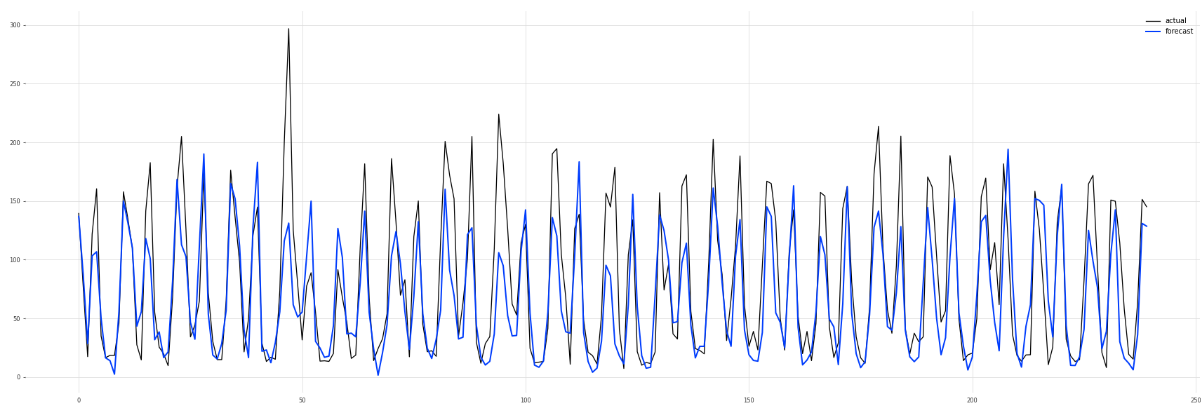


Fig. A4. N-BEATS predicted precipitation vs actual precipitation

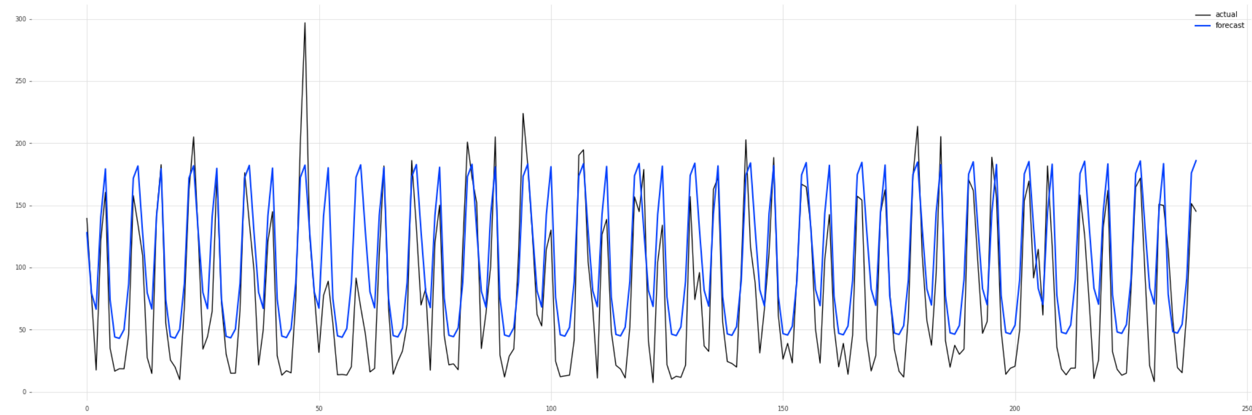


Fig. A5. Exponential Smoothing predicted precipitation vs actual precipitation

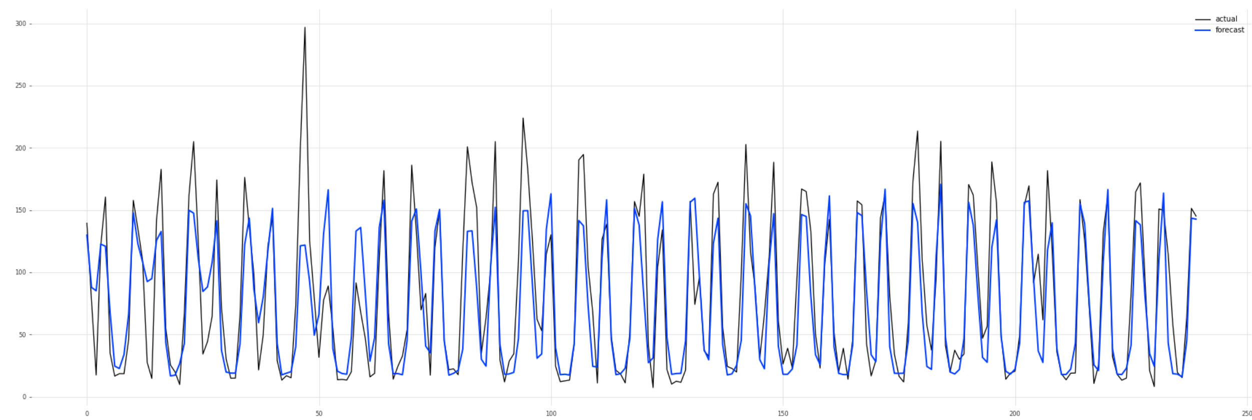


Fig. A6. LightGBM predicted precipitation vs actual precipitation

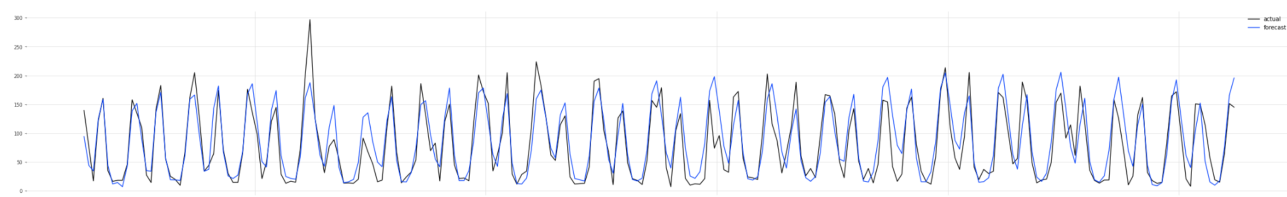


Fig. A7. Vanilla transformer predicted precipitation vs actual precipitation

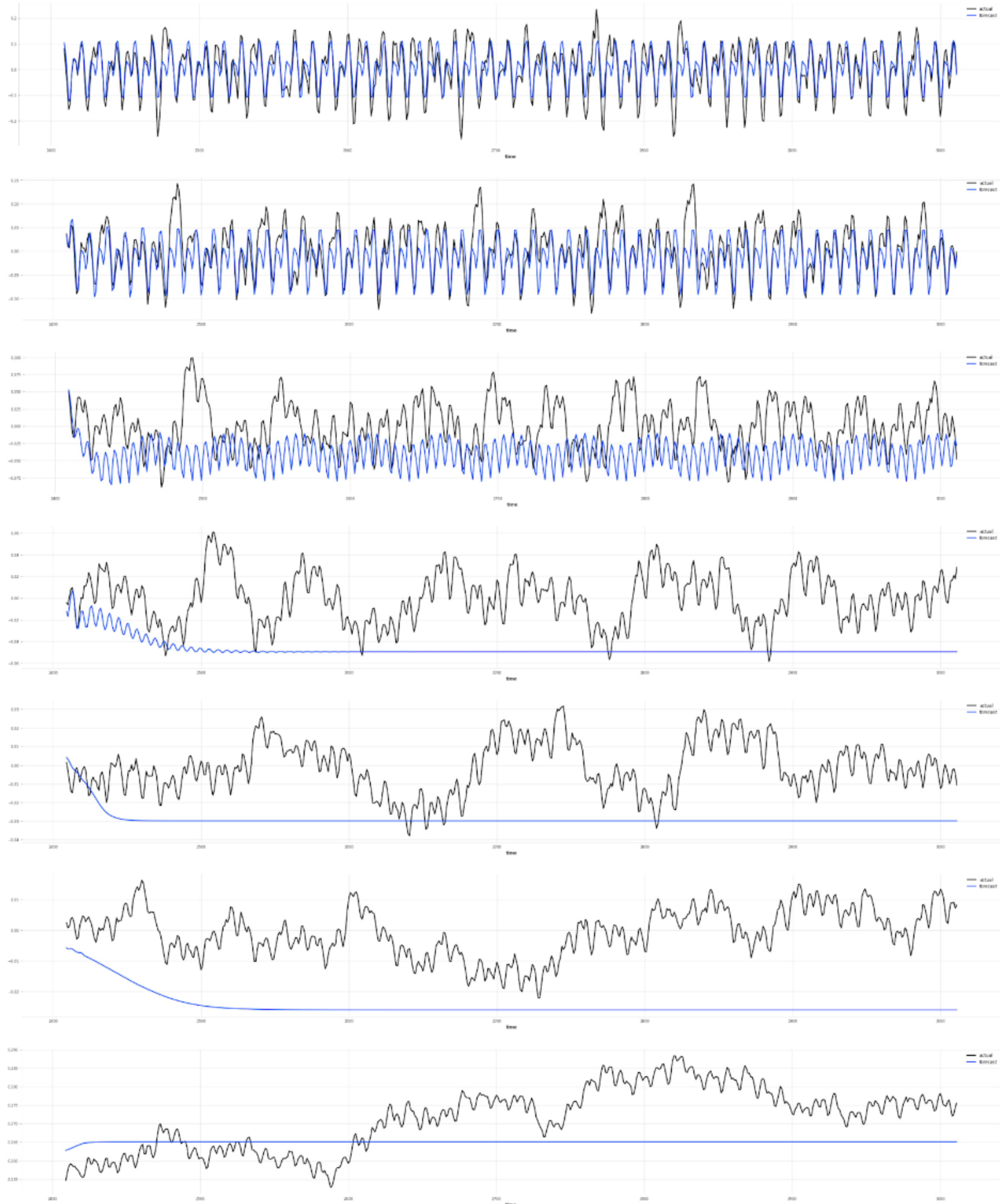


Figure A.8, Poor prediction performance of low frequency wavelets using a constant input_chunk_length