Session: NP4.1 – Analysis of complex geoscientific time series: linear, nonlinear, and computer science perspectives

Combining variational mode decomposition and recurrent neural network to predict rainfall time series and evaluating prediction performance by universal multifractals

Hai Zhou, Daniel Schertzer, Ioulia Tchiguirinskaia

Laboratory of Hydrology Meteorology & Complexity, École des Ponts ParisTech

(hai.zhou@enpc.fr)

DL for rainfall time series prediction

Motivations:

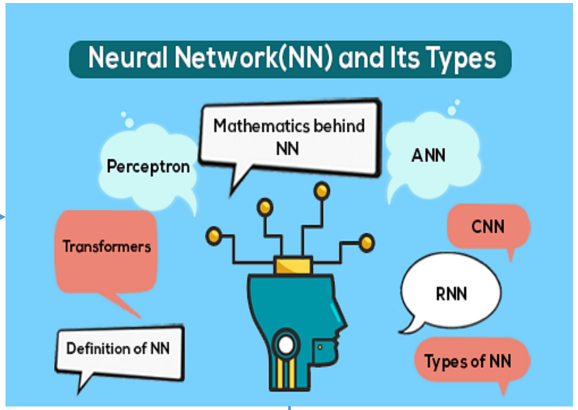
- Monitoring urban hydrological systems
- Early warning of flood and other extreme events
- Guide for water resource allocation

Challenges:

- Rainfall displays variability over a wide range of space-time scales
- Rainfall time series are extremely nonlinear

Data Driven Statistical Model
deep learning

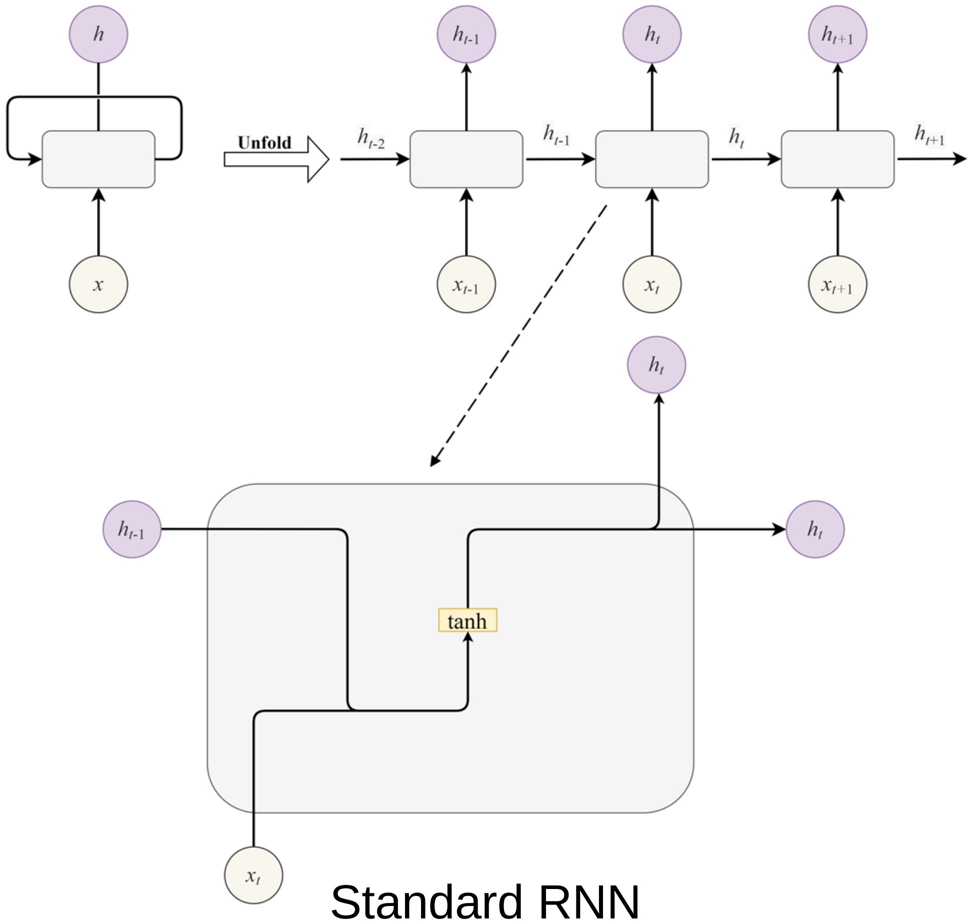
Input



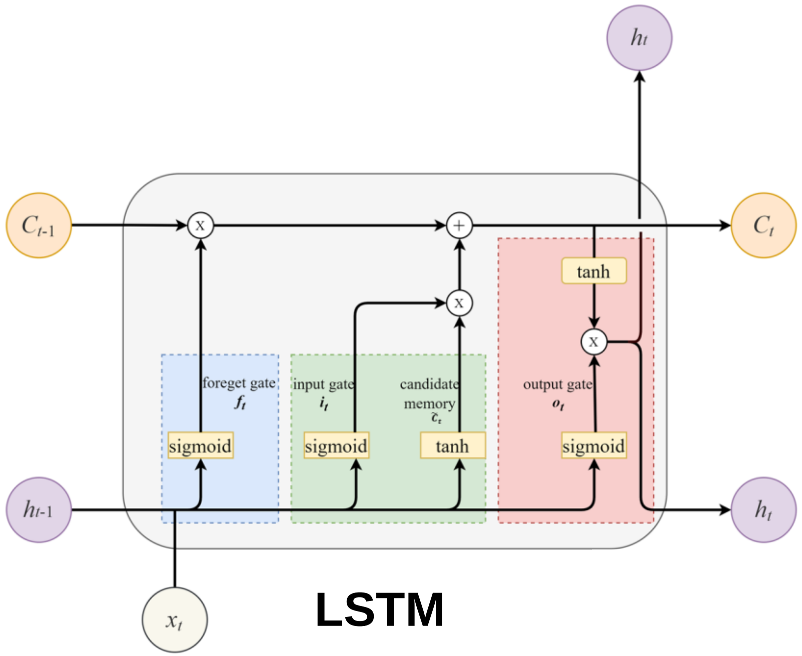
Output/prediction

source: https://www.bbc.com/news/world-europe-36451009

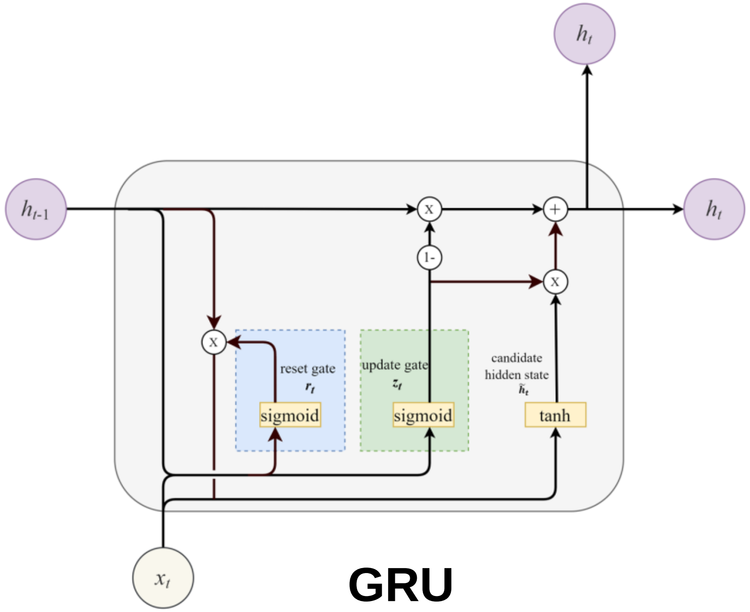
RNN(Recurrent Neural Network)



Standard RNN



LSTM



GRU

VMD (Variational Mode Decomposition)

Advantages:

- extract hidden information of complex nonlinear time series
- sound mathematical foundations
- avoid mode overlap problem

The original time series is a nonlinear signal f, and the variational problem is described as seeking K finite bandwidth mode functions $u_k(t)(k = 1,2,\dots,K)$ with central frequency, so that the sum of bandwidth estimates of each mode is minimized. The constraint condition is that the sum of each sub-sequences is equal to the input f.

$$\min_{\{u_k\},\{\omega_k\}} \left\{ \sum_{k=1}^{K} \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) \otimes u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \right\}$$

$$\text{s.t. } \textstyle\sum_{k=1}^{K} u_k = f$$

The implementation of VMD-RNN

Step1: Divide the original rainfall time series $f(t)$ $(t = 1,2,\ldots,N$, where N is the length of total data) into a training set $f_T(t)$ $(t = 1,2,\ldots,N_t$, where N_t is the training set length) and a non-training set $f_N(t)$ $(t = 1,2,\ldots,N_n$, where N_n is the non-training set length).

Step2: Decompose the training set $f_T(t)$ into K sub-sequences $u_{Ti}(t)$ $(i = 1,2,\ldots,K)$ using VMD.

Step3: Sequentially append the non-training data to the training set to generate new appended sequences $f_{TN}^j(t)$ $(j = 1,2,\ldots,N_n)$ and repeat decompose each append sequence $f_{TN}^j(t)$ into K sets of appended sub-sequences $u_{TNi}^j(t)$.

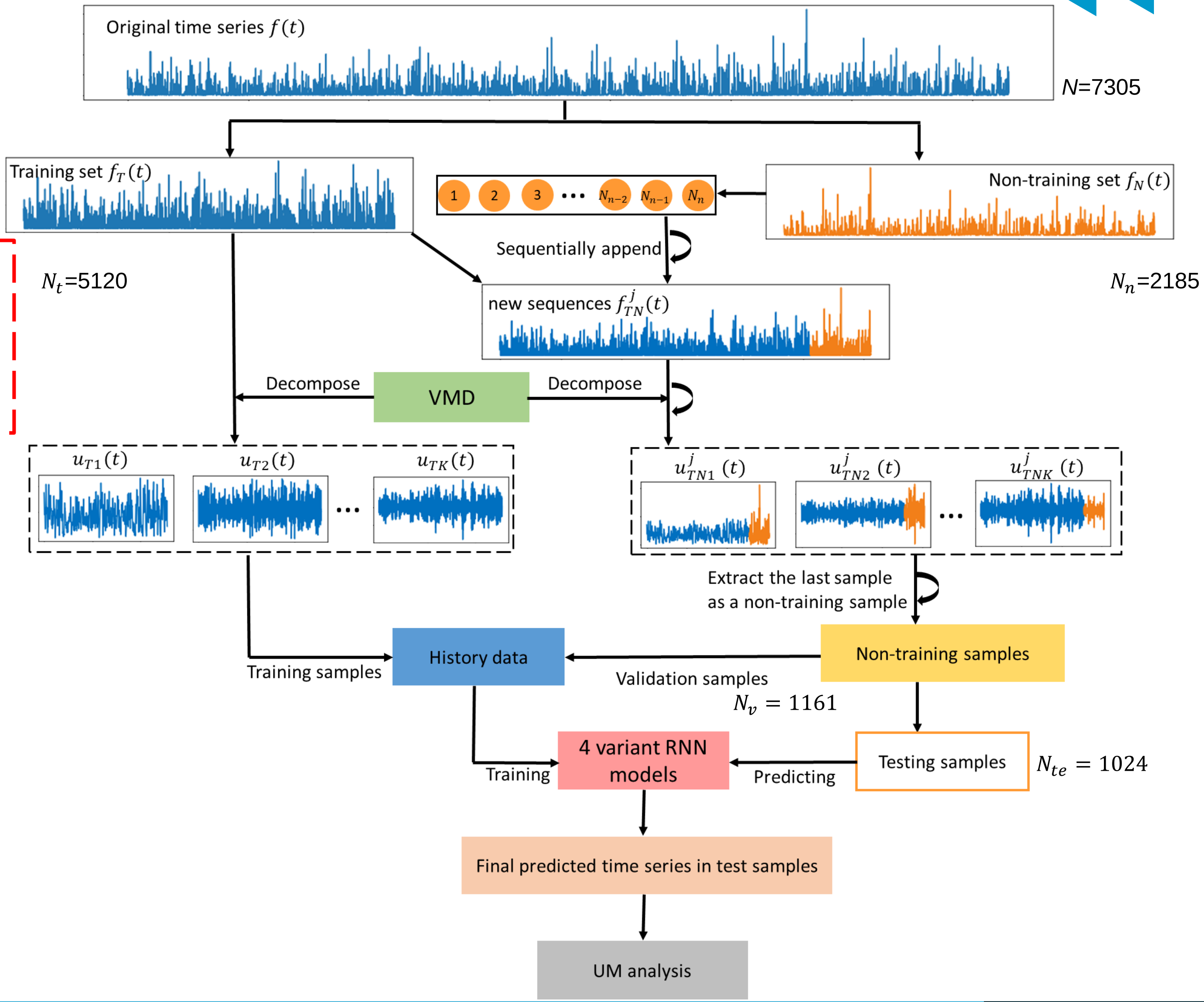
Step4: Exact the last sample of each set of appended sub-sequences as a non-training sample and divide the generated non-training samples into two subsets: validation and testing samples.

Step5: For each sub-sequences, combine data from the training set and validation samples as history data, which then were used to train four variant RNN models and tune hyperparameters to find an ideal predicting model with optimal parameters.

Step6: For each sub-sequences, input testing samples into the correspond predicting models, and obtain individual predicted results $y_i(t)$.

Step7: Aggregate the predicting results of each sub-sequences to generate the final predicted result $y(t) = \sum_{i=k}^{K} y_i(t)$.

Step8: Utilize the framework of UM to analyze the predicted and actual time series in the testing samples.



UM(Universal Multifractals)

Let's denote ε_λ is a conservative field at resolution λ, the statistical moment of order q can be defined as:

$$\langle \varepsilon_\lambda \rangle \approx \lambda^{K(q)}$$

$K(q)$ is the moment scaling function, characterizing the variability of the field at all scales.

$$K(q) = \begin{cases} \dfrac{C_1}{\alpha - 1}(q^\alpha - q) & \alpha \neq 1 \\ C_1 q \ln q & \alpha = 1 \end{cases}$$

C_1 – the mean intermittency co-dimension;

α – the multifractality index.

- **Trace Moment (TM):**

i. Calculate the empirical statistical moment $\langle \varepsilon_\lambda^q \rangle$

ii. Plot the logarithm of $\langle \varepsilon_\lambda^q \rangle$ versus the logarithm of λ

iii. Perform linear regression to obtain $K(q)$

iv. $K'(1) = C_1$ and $K''(1) = \alpha C_1$

- **Double Trace Moment (DTM):**

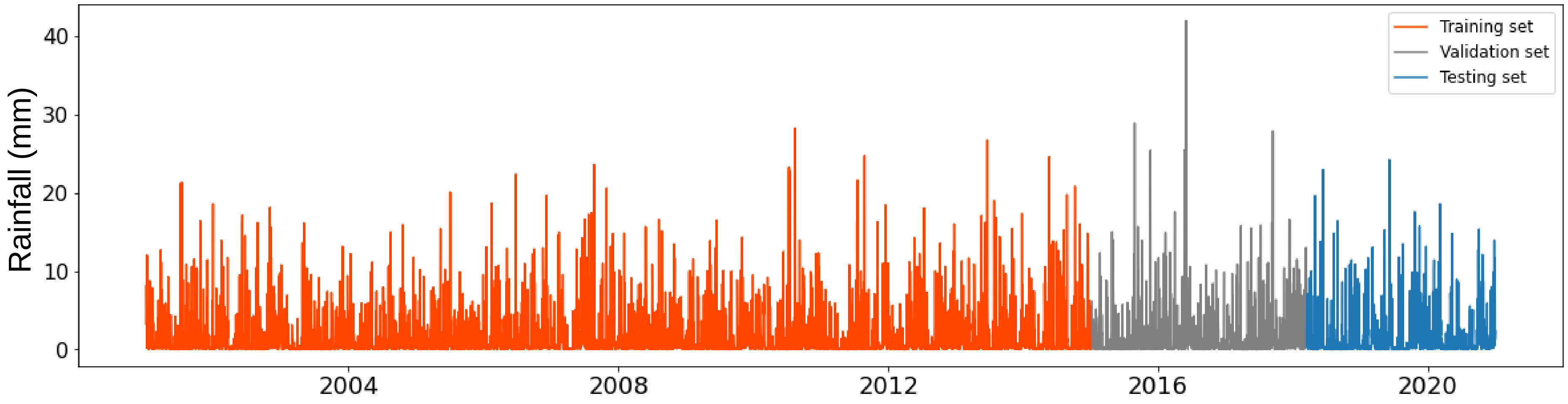
i. $\varepsilon_\lambda^{(\eta)}$ is renormalized by upscaling the η-power of the field at maximum resolution

ii. $\left\langle \varepsilon_\lambda^{(\eta)q} \right\rangle \approx \lambda^{K(q,\eta)}$, $K(q,\eta) = \eta^\alpha K(q)$

iii. C_1 and α are obtained by the slope and intercept of the linear portion of the log-log plot of $K(q,\eta)$ vs η

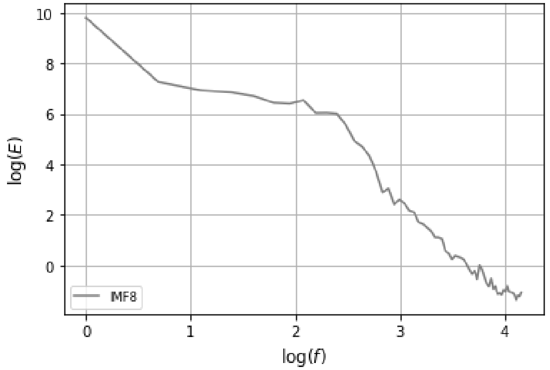
Daily time series

Dataset: Daily rainfall in Paris over the last 20 years (7305 data)

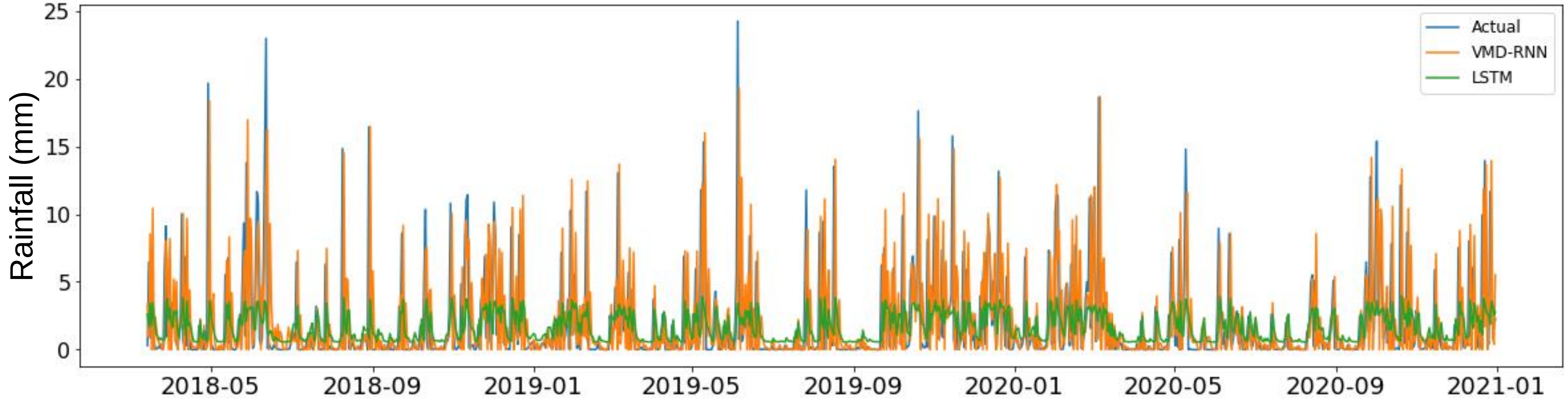


Plot PSD of the last sub-sequence to determine the number of K

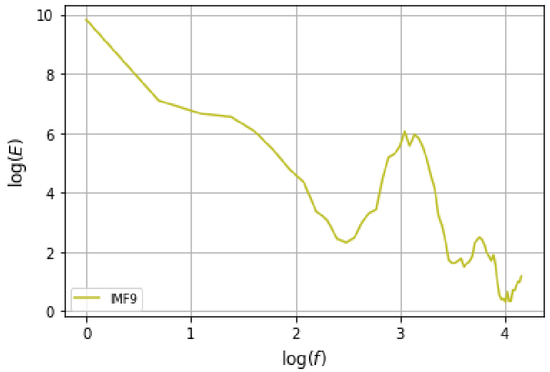
K=8



Actual and predicted daily time series by VMD-RNN and LSTM in the testing set (1024 data)

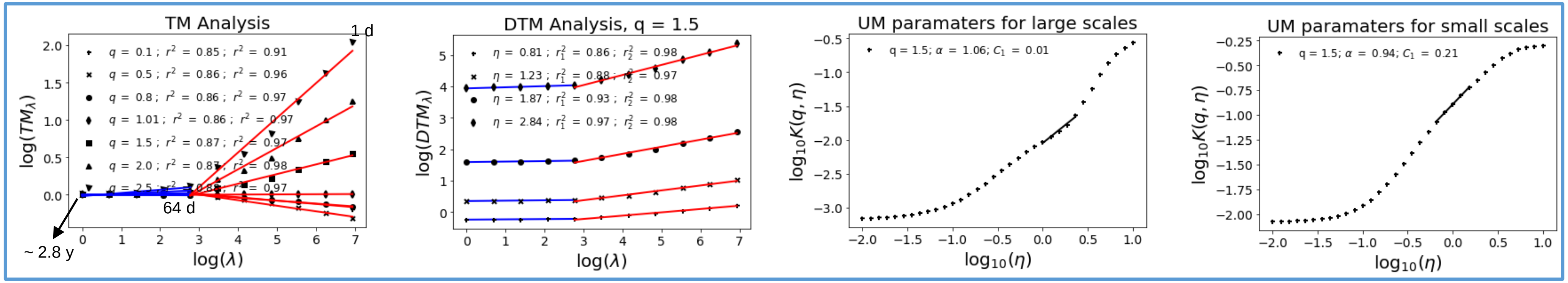


K=9

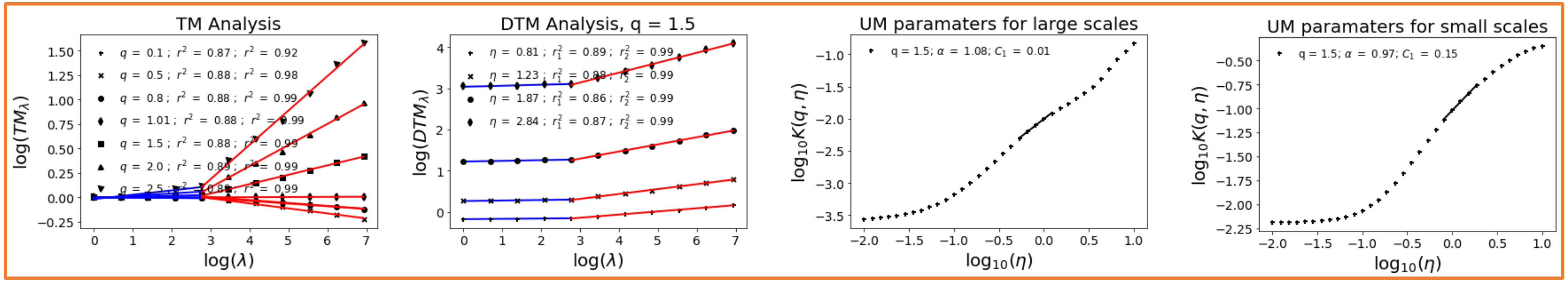


UM analysis for daily time series in the testing set

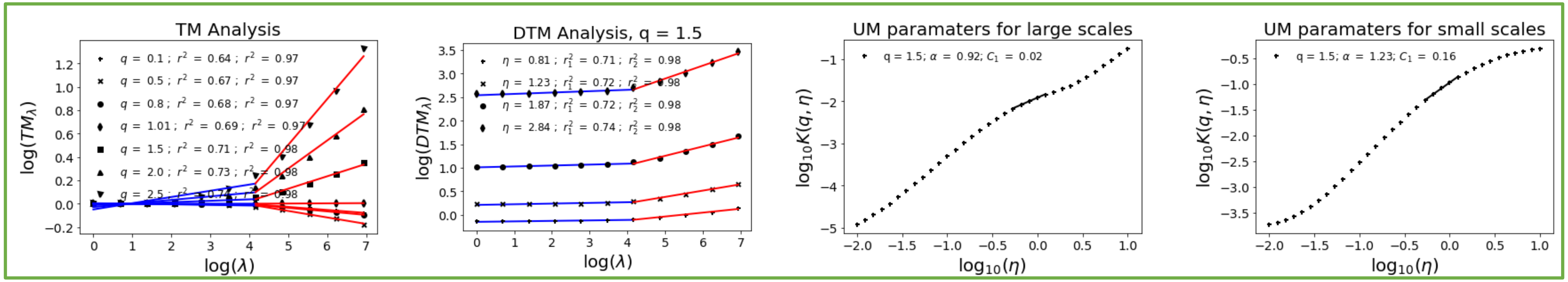
Actual time series



Predicted time series by VMD-RNN

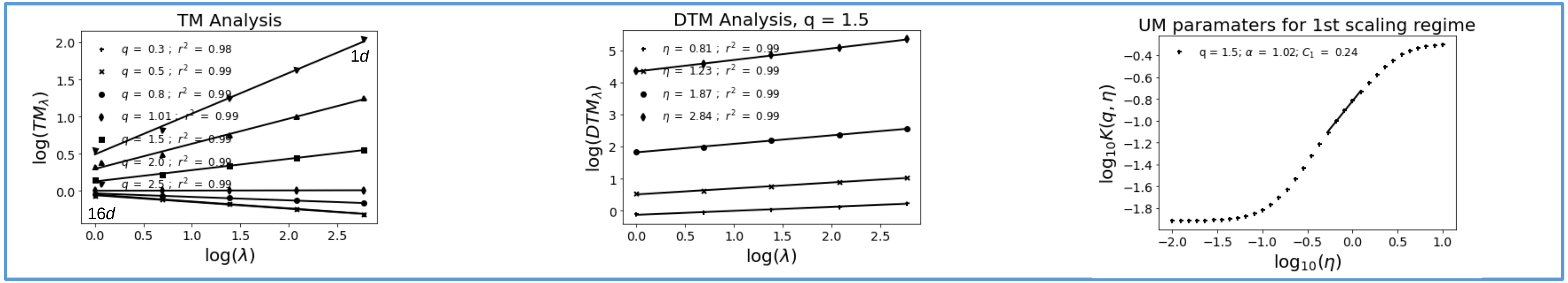


Predicted time series by LSTM

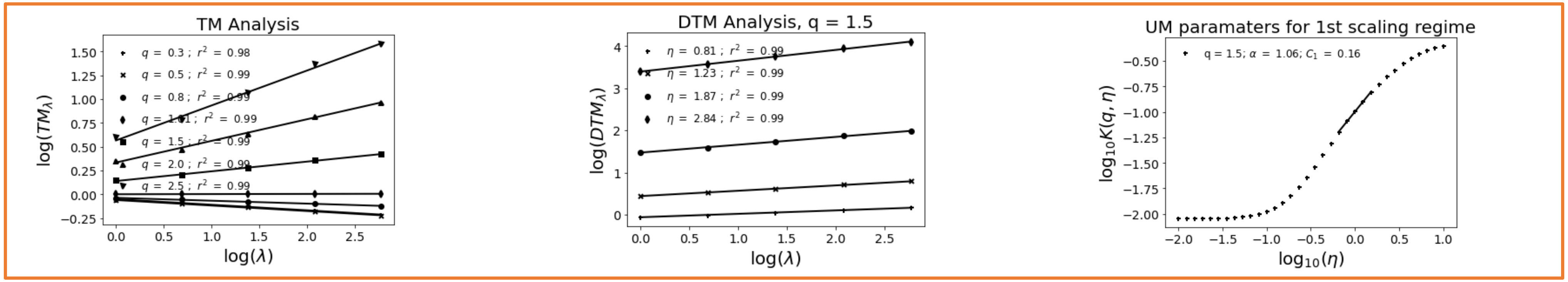


UM analysis for daily time series in the testing set (scale from 1d to 16d)

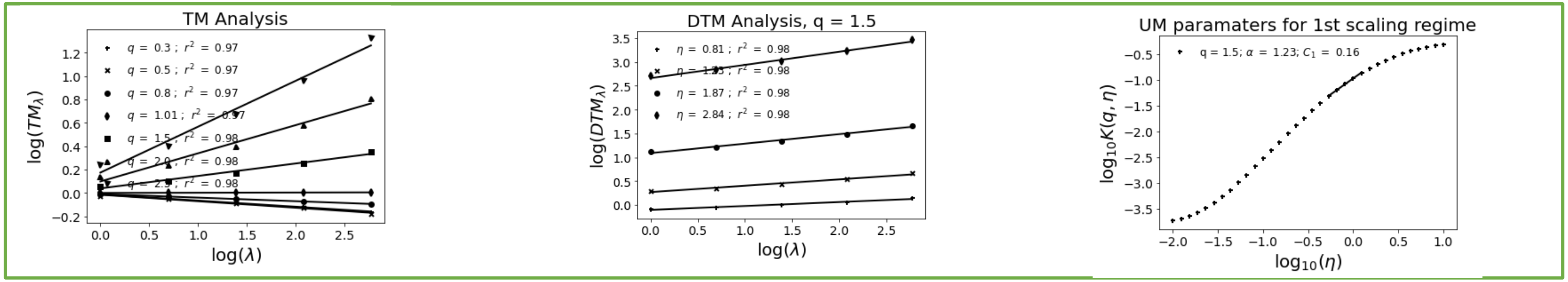
Actual time series



Predicted time series by VMD-RNN

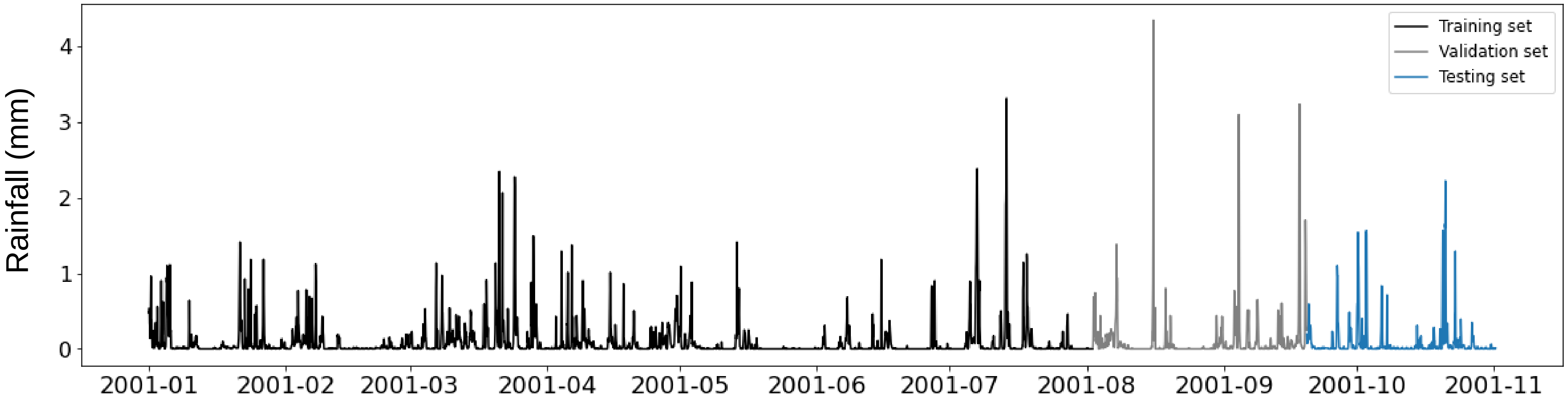


Predicted time series by LSTM without decomposition

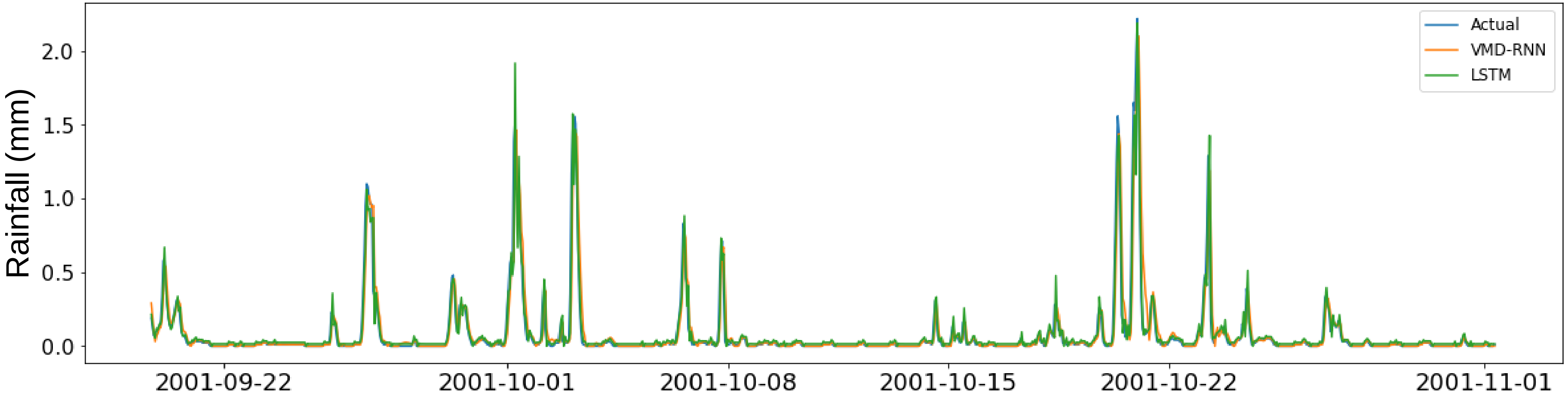


Hourly time series results

Dataset: hourly rainfall in Paris in 2001 (7305 data)

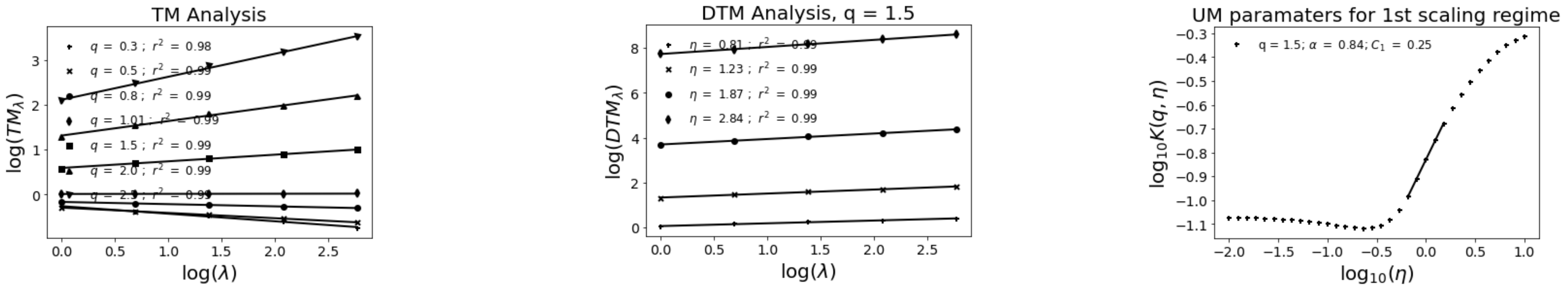


Actual and predicted hourly time series by VMD-RNN and LSTM in the testing set (1024 data)

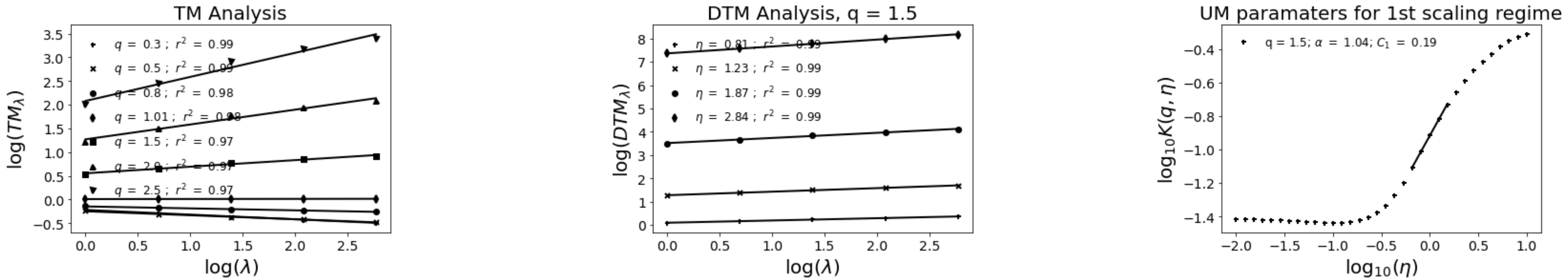


UM analysis for hourly time series in the testing set (scale from 1*h* to 16*h*)

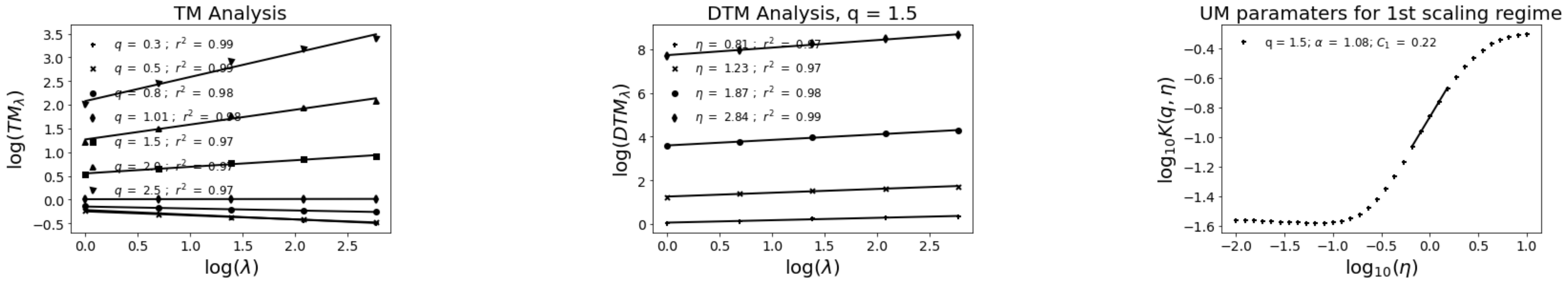
Actual time series



Predicted time series by VMD-RNN



Predicted time series by LSTM without decomposition



Thank you for your attention

Contact: hai.zhou@enpc.fr