

Article

Not peer-reviewed version

LSTCN Approaches for Self-Adaptive Real-Time Weather Forecasting

[Mba Tene Salomon](#)* and Kamla Vivient Corneille

Posted Date: 12 September 2025

doi: 10.20944/preprints202509.1092.v1

Keywords: Weather forecasting; Time series; Machine learning; Online learning algorithm; LSTCN; Data stream



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

LSTCN Approache for Self-Adaptive Real-Time Weather Forecasting

Mba Tene Salomon * and Kamla Vivient Corneille

University of Ngaoundéré, Cameroon

* Correspondence: mba_kaviaka@yahoo.com

Abstract

Weather observations are recorded using sensor networks in the form of data streams. This data is most often used by predictive models based on machine learning to make weather forecasts. However, weather conditions can vary (or change) gradually or abruptly due to the chaotic nature of the climate. These characteristics mean that conventional machine learning-based weather forecasting models deployed in a real-world environment are inadequate for accurately modeling the dynamics of weather conditions due to their static parameters. This highlights the need for other approaches capable of learning beyond the production phase, which will allow their parameters to be kept up to date in real time in response to changing weather conditions. The objective of our study is first to evaluate the performance of online learning approaches (LSTCN and ARIMA-Online) compared to conventional ML methods (LSTM, CNN, LSTM+CNN), and then to evaluate and compare the performance between these online learning approaches. This evaluation and comparison study will be carried out on univariate and multivariate time series with short- and long-term horizons. The metrics used are MAE, learning time, and testing time. According to the results of our study, the LSTCN model is the most effective compared to conventional machine learning methods in terms of training time, testing time, and accuracy on short- or long-term univariate or multivariate time series. Similarly, this model outperforms the ARIMA OGD (Online Gradient Descent) model in terms of accuracy and testing time. However, its training time is slightly longer than that of the ARIMA OGD model. So, LSTCN model is better suited for real-time weather forecasting

Keywords: weather forecasting; time series; machine learning; online learning algorithm; LSTCN; data stream

1. Introduction

Predicting the future is one of humanity's age-old dreams [29]. Science therefore aims to predict, but also to understand, the phenomenon observed, using an explanatory model [2,3,29]. The phenomenon chosen for our study is meteorology.

Meteorology is the interdisciplinary science of atmospheric physics, which studies weather conditions, the atmospheric environment, the phenomena produced, and the laws that govern it [4]. The process of meteorological measurement is directly linked to the establishment of a meteorological station [4–7]. In addition, a meteorological station is a set of sensors that record and provide physical measurements and meteorological parameters related to climate variations in a locality [4,5,7]. These collected meteorological data will be used for meteorological forecasting.

Meteorological forecasting is a vital issue in the field of science around the globe [8,9]. Thus, meteorological forecasting is the application of science and technology to predict atmospheric conditions for a specific location and a given period in the future [10–15]. Its purpose is to provide information to relevant individuals and organizations that can be used to reduce losses and improve societal benefits such as property protection, public health and safety, economic prosperity, and quality of life [15]. This forecasting involves data collection, data processing, and data analysis [16,22].

The data flow generated by the meteorological station constitutes time series. A time series is a sequence of observations recorded at regular intervals [17–19]. Depending on the frequency of observations, a time series can generally be hourly, daily, weekly, monthly, quarterly, or annual.

But, the decline in the performance of meteorological forecasting models can have a severe impact on both the economy and the environment. As a result, several models have been proposed over the centuries to solve the problem of predictive model performance.

First, numerical weather prediction (NWP) physical models use complex mathematical equations to obtain forecasts based on current weather conditions [22]. These models have long been used to predict events, but they do not allow for short-term predictions [23] and lack accuracy due to the chaotic nature of the atmosphere [23,24]. In addition, the complexity of these models poses significant challenges in their implementation [24]. To reduce the computing power of NWP systems, statistical and machine learning models have been proposed.

Several authors have proposed statistical models for meteorological forecasting. Among these authors, we have, among others, (SHIVAM, et al., 2019) [26], who used ARMA, SARIMA, and ARMAX models to predict temperature and rainfall parameters. (GARIMA and BHAWANA, 2017) [27] used the ARIMA model to predict temperature data time series. However, these methods work in univariate time series and with relatively small data sets [19,21,28]. Similarly, they do not work on nonlinear relationships [19]. Furthermore, they are more limited in predicting a long time horizon or in processing multivariate time series [19,28]. In this case, machine learning tools, in this instance deep learning, would be the ideal tool for time series forecasting according to several authors, such as [18] and [19].

To provide more accurate short- and/or long-term weather forecasts, machine learning techniques can be used to understand and analyze weather patterns. (Azencott, 2019) [29] states that, and I quote: *“At the intersection of statistics and computer science, machine learning is concerned with data modeling.”* Thus, machine learning for meteorological forecasting uses quantitative meteorological data to build models and attempts to improve model performance by learning from the dataset. According to (Sebastian and Gabriele, 2016) [30], machine learning technologies can provide intelligent models that are much simpler than physical models. It is with this in mind that (Amir et al., 2018) [23] state that ML (machine learning) models are highly accurate compared to physical and statistical models. These machine learning models, in this case offline machine learning models (traditional or conventional machine learning), can numerically formulate the non-linearity of events, based entirely on historical data without the need for knowledge of physical processes [23]. Several offline machine learning methods have been used to develop weather forecasting models, namely LSTM [10,31], CNN [32], and hybrid models (CNN + LSTM) [33]. However, once these meteorological forecasting models are designed using offline machine learning algorithms (LSTM, CNN, LSTM+CNN) and historical data, and then deployed in a real-world environment, their parameters remain static. As a result, these deployed models will encounter new environmental data dynamics that may differ from those used in the historical data. This new dynamic is due to the chaotic nature of the atmosphere, which causes gradual or sudden variations (or changes) in weather conditions. In the context of real-time meteorological forecasting, these characteristics render prediction models based on conventional ML methods (LSTM, CNN, LSTM+CNN) inadequate for accurately modeling environmental dynamics from online data streams, resulting in inaccuracy in predictive models. Similarly, existing conventional ML algorithms require model selection, which is time-consuming and not suited to the context of online learning [34]. Also, in conventional machine learning, these models are trained using the entire dataset at once. This process often requires a lot of computing time and cannot reflect changes in real time. Furthermore, these models based on offline machine learning are neither adaptive nor scalable in real time in the context of real-time meteorological forecasting, because once the models have been trained or formed, their parameters do not change.

In addition, some authors have used new approaches to improve prediction, namely algorithms capable of learning beyond the production phase, which will also enable them to be kept up to date at all times. These approaches are online machine learning methods such as ARIMA Online and LSTCN.

(CHENGHAO et al., 2016) [21] proposed online learning algorithms (ARIMA Online Newton Step and ARIMA Online Gradient Descent) for the efficient estimation of ARIMA model parameters using its recursive formulation in an online learning framework. This online learning approach processes data observations arriving sequentially and updates the models simultaneously. This online ARIMA learning is an online optimization task without noise terms. Finally, in terms of memory cost, their algorithm is independent of the sample size, making it more scalable for processing real-time time series forecasting tasks. Their algorithms have been empirically compared to two online ARMA algorithms. According to their results obtained on synthetic and real data, their proposed algorithms are effective for time series prediction. However, this approach only uses univariate time series. Furthermore, to our knowledge, this technique has not yet been used in the context of meteorological forecasting. There is another online learning approach that, in addition to using univariate time series, also processes multivariate series : the LSTCN approach.

(ALEJANDRO et al., 2022) [20] used long-term and short-term cognitive networks (LSTCN) in their article to predict wind turbine time series in online contexts. These recently introduced neural systems consist of a chain of short-term cognitive network blocks, each processing a block of time data. An LSTCN model is defined as a collection of STCN blocks, each processing a specific time slice (or range) and transferring knowledge to subsequent STCN blocks in the form of weighted matrices. An incoming data block triggers new learning on the last STCN block using the stored knowledge that the network has learned during previous iterations. Then, the prior knowledge matrices are recalculated using an aggregation operator and stored for use as prior knowledge during reasoning. According to their results, the solution of their simulations outperforms traditional neural networks and reports significantly shorter training and testing times. However, to our knowledge, this technique has not yet been used in the context of meteorological forecasting.

The forecasting methods used for the environment, particularly for weather, are based on physical, statistical, or machine learning models [20]. According to [20], although conventional machine learning models often achieve the highest performance compared to other models, their deployment in real-world applications shows inappropriate results in terms of accuracy. This drop in performance is due to the static parameters of these real-time forecasting models because once these conventional ML models are generated, their parameters no longer change, making it impossible to self-adapt or update to the non-linearity of the environment, i.e., to gradual and/or sudden changes in the climate. These variations or changes in meteorology or climate lead to changes in the internal structure of meteorological data, i.e., new distributions of meteorological data. Conventional machine learning is also known as offline machine learning.

One of the main challenges of these models is to improve the accuracy of weather forecasting models. Real-time updates could ensure the accuracy of predictive models in the context of real-time meteorological forecasting. As indicated in [21], enabling forecasting systems to adapt to climate variability and change is a vital necessity.

To overcome these limitations, online learning approaches have been proposed, such as the LSTCN (Long Short-Term Cognitive Network) method [20] and ARIMA-OGD [21]. To our knowledge, these approaches have not been applied in the context of real-time meteorological forecasting. The objective of our study is first to evaluate the performance of these online learning approaches against conventional ML methods (LSTM, CNN, LSTM+CNN), and then to evaluate and compare the performance between these online learning approaches (LSTCN and ARIMA-Online). This evaluation and comparison study will be carried out on univariate and multivariate time series and with short- and long-term horizons.

The remainder of this paper is organized as follows. Section 2 presents the materials and methods. Section 3 illustrates the results and analysis. Section 4 discusses the findings. Finally, Section 5 concludes the paper.

2. Materials and Methods

A time series is a sequence of meteorological observations recorded at regular time intervals [17–19]. Depending on the frequency of meteorological observations, a time series can generally be hourly, daily, weekly, monthly, quarterly, and annual [19]. Let $x \in \mathbb{R}$ be a meteorological variable observed on a discrete time scale over a period $t \in \{1, 2, \dots, T\}$ where $T \in \mathbb{N}$ is the number of meteorological observations. Therefore, a univariate time series is defined as a sequence of meteorological observations $\{x^{(t)}\}_{t=1}^T = \{x^{(1)}, x^{(2)}, \dots, x^{(T)}\}$. Similarly, we can define a multivariate series as a sequence $\{X^{(t)}\}_{t=1}^T = \{X^{(1)}, X^{(2)}, \dots, X^{(T)}\}$ of vectors of M weather variables, such as $X^t = \{x_1^{(t)}, x_2^{(t)}, \dots, x_M^{(t)}\}$. A model F is used to predict the next time steps in advance, denoted as L , such that $L < T$.

We will use two sets of models for our meteorological forecasting simulations of our meteorological data. These are the models based on conventional Machine Learning methods (LSTM, CNN, CNN+LSTM) and the models based on online learning methods (ARIMA-OGD and LSTCN). First of all, we will conduct a performance comparison between these two sets of methods, and then conduct a performance comparison between the online learning models.

These two sets of machine learning models consist of two categories of learning models for meteorological forecasting, mainly SISO (Single-Input Single-Output) and MIMO (Multi-Input Multi-Output) [24]. The SISO is used in univariate time series, while the MIMO is used in multivariate time series. Regarding the SISO model, only one variable is introduced into the learning model to generate only one output. And the MIMO model, on the other hand, allows for the introduction of N variables into the learning model to generate N outputs.

Our simulations will be conducted on univariate and multivariate time series over short-term and long-term horizons. Following the same logic as [24], in our study, the short-term horizon is assigned to 1 hour and the long-term horizon is assigned from 2 hours onward.

Our historical meteorological dataset consists of meteorological data from the city of Ngaoundéré-Cameroon, spanning from 01 – 01 – 2011 to 31 – 12 – 2020, covering a period of 10 years. The frequency of observations is hourly in our dataset. Our dataset contains the following five meteorological parameters: temperature, precipitation, atmospheric pressure, air humidity, dew point. The total number of observations is 87,671 meteorological observations (or records).

For the simulation of SISO models, we will use only the Temperature parameter. Regarding the simulation of MIMO models, we will use five meteorological parameters.

Subsequently, our dataset will be divided into 80% for training data and 20% for test data. The metrics used will be MAE, training time, and testing time.

We will conduct an evaluation and performance study between the models to determine the most performing model in the case of univariate series on one hand and in the case of multivariate series on the other hand. This study will be conducted based on short-term and long-term horizons.

The programming language used to code all these methods is Python, version 2.9.1. Our simulations will be performed on the Jupiter Notebook IDE. Our PC on which our simulations will be performed has the following characteristics: Our PC on which our simulations will be performed has the following characteristics: Brand HP, operating system (Windows 10 Professional 64-bit), memory (4GB), hard drive (500GB), processor type (AMD), and processor speed (1.80GHz).

3. Results and Analysis

The results and our analyzes will be done respectively on univariate time series (for SISO models) and multivariate time series (for MIMO models). Let's start with the results and our analyzes on univariate time series.

It should be noted that our results on univariate time series are based solely on the temperature parameter. Table 1 presents the metrics of our different models in the case of univariate series for a prediction horizon of 1 hour.

Table 1. Metrics on univariate series with a prediction horizon of 1 hour (short-term horizon).

<i>SISO</i> : 1 hour					
Methods	Error(MAE)		Time		Parameter
	Test	Training	Test		
LSTM	$8972 * 10^{-6}$	1960s	10.7s		Temperature
CNN	$9860 * 10^{-6}$	560s	6.5s		
LSTM+CNN	$6909 * 10^{-6}$	3196s	16.2s		
LSTCN	$5020 * 10^{-6}$	20.28s	0.053s		
ARIMA OGD	$7828 * 10^{-6}$	5.85s	1.60s		

Table 1 shows that the LSTCN model performs best in terms of mean absolute error ($5020 * 10^{-6}$). It also has the best execution time for test data (0.053 seconds). In addition, it is considered the second-best model in terms of training time (20.28 seconds). In contrast, the ARIMA OGD model performs the worst in terms of mean absolute error ($7828 * 10^{-6}$). However, it has the fastest training time (5.85 seconds).

Table 2 and Table 3 present the metrics of our different models for long-term horizons, namely: a 2-hour horizon for Table 2 and a 12-hour horizon for Table 3.

Table 2. Metrics on univariate series with a 2-hour prediction horizon (long-term horizon).

<i>SISO</i> : 2 hour					
Methods	Error(MAE)		Time		Parameter
	Test	Training	Test		
LSTM	$19478 * 10^{-6}$	1534s	10.7s		Temperature
CNN	$20325 * 10^{-6}$	688s	5.8s		
LSTM+CNN	$12294 * 10^{-6}$	3878s	66s		
LSTCN	$11499 * 10^{-6}$	8.3s	0.06s		
ARIMA OGD	$78172 * 10^{-6}$	5.15s	1.15s		

Table 3. Metrics on univariate series with a 12-hour prediction horizon (long-term horizon).

<i>SISO</i> : 12 hour					
Methods	Error(MAE)		Time		Parameter
	Test	Training	Test		
LSTM	$46739 * 10^{-6}$	778s	10.9s		Temperature
CNN	$71043 * 10^{-6}$	253s	13.9s		
LSTM+CNN	$36029 * 10^{-6}$	2061s	47.6s		
LSTCN	$33464 * 10^{-6}$	111s	0.33s		
ARIMA OGD	1.19696	5.12s	1.10s		

Regarding Tables 2 and 3, for the long-term horizon, the LSTCN model performs best in terms of mean absolute error, with $11499 * 10^{-6}$ for the 2-hour prediction horizon and $33464 * 10^{-6}$ for the 12-hour prediction horizon. It also has the shortest testing time for both prediction horizons, namely 2 hours (0.06 seconds) and 12 hours (0.33 seconds). In addition, this model is considered to be the second best model with a reduced training time of 8.3 seconds. Furthermore, the ARIMA OGD model is the least performant model in terms of mean absolute error in the long-term horizons. However, this

model is considered to be the best model in terms of training time for the 2-hour (5.15 seconds) and 12-hour (5.12 seconds) prediction horizons. After analyzing our various SISO model results, we will now analyze the various MIMO model results.

With regard to the tables below, Table 4 presents the evaluation metrics for our different models for a short-term prediction horizon (1 hour), while Table 5 and Table 6 illustrate the evaluation metrics for our different models for long-term prediction horizons, namely a 2-hour prediction horizon for Table 5 and a 12-hour prediction horizon for Table 6. Furthermore, in the tables below, the ARIMA OGD model is excluded because this model does not take multivariate series into account.

Table 4. Metrics on multivariate time series (MIMO) with a prediction horizon of 1 hour (short-term horizon).

Methods	<i>MIMO</i> : 1 hour		
	Error(MAE)	Time	
	Test	Training	Test
LSTM	$7042 * 10^{-6}$	2038s	18s
CNN	$7707 * 10^{-6}$	856s	15.1s
LSTM+CNN	$6503 * 10^{-6}$	3143s	12.7s
LSTCN	$4420 * 10^{-6}$	12.8s	0.022s

Table 5. Metrics on multivariate time series (MIMO) with a prediction horizon of 2 hours (long-term horizon).

Methods	<i>MIMO</i> : 2 hour		
	Error(MAE)	Time	
	Test	Training	Test
LSTM	$12292 * 10^{-6}$	3434s	12.8s
CNN	$13739 * 10^{-6}$	481s	7.26s
LSTM+CNN	$12297 * 10^{-6}$	1987s	13.7s
LSTCN	$1060 * 10^{-6}$	17.7s	0.038s

Table 6. Metrics on multivariate time series (MIMO) with a prediction horizon of 12 hours (long-term horizon).

Methods	<i>MIMO</i> : 12 hour		
	Error(MAE)	Time	
	Test	Training	Test
LSTM	$32370 * 10^{-6}$	1521s	11.5s
CNN	$36289 * 10^{-6}$	945s	10.2s
LSTM+CNN	$31233 * 10^{-6}$	1696s	13s
LSTCN	$30083 * 10^{-6}$	98.7s	0.23s

Regarding Table 4 presenting the metrics for the short-term horizon, Table 5 and Table 6 for the long-term horizon for MIMO models, the LSTCN model is the most performing model in terms of MAE and in terms of training and testing time.

4. Discussion

Our simulations show that online learning models have a faster training and testing time than conventional Machine Learning models on univariate and multivariate time series.

Although previous studies have not conducted a comparative analysis between the LSTCN and ARIMA OGD models, and performed simulations on meteorological data, we conducted our study

in this direction. According to our results, the forecast data generated by the LSTCN model is more accurate than the ARIMA OGD model on univariate time series for both short-term and long-term forecasts. Moreover, the training time of the LSTCN model is slightly longer than that of the ARIMA OGD model. The ARIMA OGD model is only restricted to univariate time series, which limits the simulation on multivariate time series. And yet, the LSTCN extends the simulation to multivariate time series.

The LSTCN model is also the most performant model compared to conventional Machine Learning models, whether in terms of MAE, training time, or testing on univariate and multivariate time series for short-term and long-term forecasts.

Regarding long-term predictions for univariate and multivariate series, it turns out that the accuracy of forecast data can degrade as the prediction horizon increases.

Based on the results of our simulations, we were able to observe that LSTCN model are better suited for real-time weather forecasting.

5. Conclusions

In summary, weather forecasting is the application of science and technology to predict weather conditions for a specific location and the given period in the future. Thus, several forecasting methods used for the environment, particularly for weather, are based on physical, statistical, or machine learning models. although conventional machine learning models often achieve the highest performance compared to other models, their deployment in a dynamic environment shows a certain inadequacy, namely the chaotic nature of the atmosphere in the context of real-time weather forecasting. Updating conventional models with new meteorological information is often very costly. This update could ensure the accuracy of the predictive model. To address these limitations, online learning approaches have been proposed, such as the LSTCN method and ARIMA-OGD. These approaches were compared against conventional Machine Learning methods, namely LSTM, CNN, and CNN+LSTM. According to the results, the LSTCN model is the most effective compared to conventional machine learning methods in terms of training time, testing time, and accuracy on short- or long-term univariate or multivariate time series. Similarly, this model outperforms the ARIMA OGD (Online Gradient Descent) model in terms of accuracy and testing time. However, its training time is slightly longer than that of the ARIMA OGD model. LSTCN model is better suited for real-time weather forecasting.

References

1. Azencott, C.-A. *Introduction au Machine Learning*, 2nd ed.; DUNOD: 11 rue Paul Bert, 92240 Malakoff, France, 2019; pp. 1–240.
2. Xingjian, S.; Zhou, R.; Hao, W.; Dittmann, Y. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. *NeurIPS Proceedings* **2016**, 1–11.
3. Mohamed, S.T. Application de l'apprentissage artificiel à la prévision des crues éclair. Ecole Nationale Supérieure des Mines de Paris, Paris, 2010.
4. Marcos, A.P.-J.; Carlos, V.-F.; David, T.-P.; George, P.-M. Diseño de una estación meteorológica automática para registrar las variables solar y eólica. *Revista Arbitrada Interdisciplinaria KOINONIA* **2020**, *5*, 937–957. <https://doi.org/10.35381/r.k.v5i2.1072>
5. Aris, M.; Hanif, F.; Muhammad, I.R.; Rian, P.P.; Jony, W.W.; Irfan, A.F.A. Design of Real-time Weather Monitoring System Based on Mobile Application using Automatic Weather Station. In Proceedings of ICACOMIT, Jakarta, Indonesia, October 2017; 44–47. <https://doi.org/978-1-5386-0510-3/17/>
6. Zaid, K.H.; Hadi, J.H.; Moussa, R.A.-M.; Yaqeen S.M. Low cost smart weather station using Arduino and ZigBee. *TELKOMNIKA* **2020**, *18*, 282–288. <http://doi.org/10.12928/telkomnika.v18i1.12784>
7. Yong-Hua, C.; Si-Ren, L. Design and realization of an automatic weather station at island. Proceedings of SPIE - The International Society for Optical Engineering, China, October 2011; 8205, 1–6. <https://doi.org/10.1117/12.906350>
8. Garima, J.; Bhawna, M. A Review on Weather Forecasting Techniques. *IJARCCCE* **2016**, *5*, 177–180. <https://doi.org/10.17148/IJARCCCE.2016.51237>

9. Marzieh, F.; Mostafa, H.K.; Seyed, M.J.; Ebrahim, M. Big Data Analytics in Weather Forecasting: A Systematic Review. *Archives of Computational Methods in Engineering*, **2021**, *21*. <https://doi.org/10.1007/s11831-021-09616-4>
10. Siddharth, S.; Mayank, K.; Ambuj, G.; Anil, K.M. Weather Forecasting using Machine Learning Techniques. *SSRN Electronic Journal* **2019**. <https://doi.org/10.2139/ssrn.3350281>
11. Ning, Y.; Lewis, W.; Preeti, D. A Weather Prediction Model with Big Data. **2018**.
12. Pablo, R.L. Application of machine learning techniques to weather forecasting. DOCTORAL THESIS, University of the BASQUE Country UPV/EHU, 2018.
13. Zhanjie, W.; Mazharul Mujib, A.B.M. The Weather Forecast Using Data Mining Research Based on Cloud Computing. *Journal of Physics Conference Series* **2017**, *910*, 012020. <https://doi.org/10.1088/1742-6596/910/1/012020>
14. Priya, S.; Radhamani, A. Weather Prediction based on Wireless Sensor Network and Internet of Things with Analysis using Hybrid SSOA with MA. *Research Square* **2021**. <https://doi.org/10.21203/rs.3.rs-911875/v1>
15. Deepak, D.; Chetan, W. A literature review on Improvement of Weather prediction by using Machine learning. *SSRN Electronic Journal* **2022**. <https://doi.org/10.2139/ssrn.4140237>
16. Shruti, D.; Vibhakar, P.; Rohit, M.; Ruchi, D. Machine learning for weather forecasting. In *Machine Learning for Sustainable Development*; 2021, pp. 161–174. <https://doi.org/10.1515/9783110702514-010>
17. Selva, P. Analyse de séries chronologiques en Python - Un guide complet avec des exemples. 2019
18. Pal, A.; PKS, P. Practical Time Series Analysis; Packt, 2017.
19. Marco, P. Time Series Forecasting in Python; Manning; 2022.
20. Alejandro, M.-H; Gonzalo, N.; Agnieszka, J.; Yamsleydi, S.; Koen, V. Online learning of windmill time series using Long Short-term Cognitive Networkd. *Expert Systems with Applications* **2022**, *205*, 1–10. <https://doi.org/10.1016/j.eswa.2022.117721>
21. Chenghao, L.; Steven, C.H.; Peilin, Z.; Jianling, S. Online ARIMA Algorithms for Time Series Prediction. In Proceedings of the AAI Conference on Artificial Intelligence, 21 February 2016; 30, 1867–1873. <https://doi.org/10.1609/aaai.v30i1.10257>
22. Ismaila, O. Machine learning-based algorithms for weather forecasting. *International Journal of Artificial Intelligence and Machine Learning* **2022**, *2*, 12–20. <https://doi.org/10.51483/IJAIML.2.2.2022.12-20>
23. Amir, M.; Pinar, O.; Chau, K.-W. Flood Prediction Using Machine Learning Models : Literature Review. *Water* **2018**, *10*, 1536. <https://doi.org/10.3390/w10111536>
M. AMIR, O. PINAR et K.-w. CHAU, «Flood Prediction Using Machine Learning Models : Literature Review,» *Water*, 27 October 2018.
24. Pradeep, H.; Ardhendu, B.; Marcello, T.; Ella, P.; Morteza, G. Francesco, P.; Yonghuai, L. Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station. *Soft Computing* **2020**, *24*. <https://doi.org/10.1007/s00500-020-04954-0>
25. Bupalov, L.; Katser, L.; Kozitsin, V.; Makarov, E. Online ARIMA. **2019** .
26. Shivam, B.; Rajat, B.; Ankit, S.C.; Akshay, K.D.; Indu, C. Fuzzy Logic based Crop Yield Prediction using Temperature and Rainfall parameters predicted through ARMA, SARIMA, and ARMAX models. In Proceedings of 2019 Twelfth International Conference on Contemporary Computing (IC3), India, August 2019. <https://doi.org/10.1109/IC3.2019.8844901>
27. Garima, J.; Bhawana, M. A Study of Time Series Models ARIMA and ETS. *SSRN Electronic Journal* **2017**, *4*, 57–63. <https://doi.org/10.2139/ssrn.2898968>
28. Gonzalo, N.; Isel, G.; Agnieszka; Yamsleydi, S. Long Short-term Cognitive Networks,» Machine Learning with Applications. *Arxiv* **2021**, *1*, 1–10. <https://doi.org/10.48550/arXiv.2106.16233>
29. Azencott, C.-A. Introduction au Machine Learning; DUNOD: Paris, 2019
30. Sebastian, S.; Gabriele, M. Predicting Weather Forecast Uncertainty with Machine Learning. *Quarterly Journal of the Royal Meteorological Society* **2016**, *144*, 1–17. <https://doi.org/10.1002/qj.3410>
31. Zi-Qi, H.; Ying-Chih, C.; Chich-Yu, W. Real-Time Weather Monitoring and Prediction Using City Buses and Machine Learning. *Sensors* **2020**, *20*, 5173. <https://doi.org/10.3390/s20185173>
32. Jonathan, A.W.; ale, R.D.; Rich, C. Improving Data-Driven Global Weather Prediction Using Deep Convolutional Neural Networks on a Cubed Sphere. *Journal of Advances in Modeling Earth systems* **2020**, *20*, 1–17. <https://doi.org/10.1029/2020MS002109>
33. Lim, S.-C.; Huh, J.-H; Hong, S.-H; Park, C.-Y.; Kim, J.-C. Solar Power Forecasting CNN-LSTM Hybride Model. *Energies* **2022**, *15*, 8233. <https://doi.org/10.3390/en15218233>

34. Weijia, S.; Lukas, F.R.; Fikret, S.; Sahin, A. Adaptive Online Learning for the Autoregressive Integrated Moving Average Models. *mathematics* **2021**, *9*, 1523. <https://doi.org/10.3390/math9131523>

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.