International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 04 April 2025, pp: 1140-1147 www.ijaem.net ISSN: 2395-5252

A Tool for Predicting Lawsuit Filings in the Brazilian Judiciary.

Jefferson Silva

Date of Submission: 20-04-2025 Date of Acceptance: 30-04-2025

ABSTRACT
The Braziliancourt system faces significant challenges in managing the fluctuating demand for litigations, which impacts resource allocation, operational efficiency, andpolicy planning. This study applies time-series forecasting techniques to predict law suit demands, aiming to provide action able insights that can enhance the efficiency of the justice system. Five forecasting algorithms, ARIMA, Prophet, NeuralProphet, DeepAR, and N-Beats, were evaluated using historical data from two state courts: the Rio Grande do Norte State Court (TJRN) andthe Minas Gerais StateCourt (TJMG). The models wereassessedbasedonthe Root Mean Square Error (RMSE), with statistical analyses, includingthe Friedman and Nemenyitests, used to compare performance. The results showed that the Prophet and Neural Prophet algorithms out performed others, with RMSE values as low as 3.44 and 4.54 for TJMG, respectively. These findings suggest that advanced time series models can effectively forecast demand, enabling the justice system to anticipate periods of high activity, optimize resource distribution, and potentially reduce costs. Future research could extend this approach by integrating additional socio economic variables and applying these models across different segments of the justice system, opening up new avenues for exploration and development in the field. This study highlights the potential of data-driven strategies to enhance operational planning and decision-making with in the judiciary.

I. INTRODUCTION

Like many other judicial systems world wide, the Brazilian court system is characterized by high demandsandproductions (1). These demands and productions vary over time, creating a complex pattern that canbe challenging to predict (2). However, accurate prediction of these patterns is crucial for efficient resource allocation, policy planning, and decision-making (3).

Time series prediction, a statistical technique that utilizes historical data topredict future values, hasbeen widely used in various fields such as finance, healthcare, and transportation (4). This paper aims to apply time series prediction techniques to the Brazilian court system’s demands and productions. Bydoingso, wehopetoprovidevaluable insights thatcould help improve the Brazilian court system’s efficiency and effectiveness.

This paper will first providean overview of the Brazilian court system, focusing on its demands and productions. We will then discuss our time-series prediction approaches, including their strengths and limitations. Followingthis, we will presentour findings, including the accuracy o four predictions and their implications for the Brazilian court system. Finally, we will conclude with a discussion of potential future research directions in this area.

Through this research, we aim to contribute to the growing literature on applying time-series prediction in [Version: 2020/08/31 v1.00] public administration. Our findings could have significant implications for the Brazilian court system and other judicial systems worldwide facing similar challenges. We hopeour research will inspire further studies in thisarea, ultimately leading to more efficient and effective judicial systems.

Contributions

The findings from this study could significantly inform resourceal location strategies with in the Brazilian justice system in severalways:

1. Predictive Insights: By accurately forecasting the demand for lawsuits, the justice system can anticipate periods of high demand and allocatere sources accordingly. This could involvead justing staffing levels, redistributing workload, or evenaltering court schedules to manage the anticipated demand;
2. Efficiency: With a clear understanding of future demands, the justice system canoptimize its operations to handle these demands efficiently. This could lead to faster processing times, reduced backlog, and overall improved efficiency in the justice system;

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 04 April 2025, pp: 1140-1147 www.ijaem.net ISSN: 2395-5252

3. Cost Savings: Accurate fore casting can also lead to cost savings. By knowing when to expect high or low demand, the justice system can make informed decisions about when to invest in additional resources and when to scale back. This could prevent unnecessary spending and lead tosignificant cost savings;
4. Policy Planning: On a broaderscale, these forecasts could inform policy planning within the justice system. Policymakerscould use this information to develop strategies that address the underlying causes offluctuations in law suit demand, potentially leading to more effective policies.

In summary, applying time-series forecasting to predict law suit demands could provide valuable insights that enhance operational efficiency, cost-effectiveness, and policy planning within the Brazilian justice system.

Time-series applied to Justice systems

Uponextensive review of the existing literature, no prior research has specifically targeted the forecasting of law suitdemands within the Brazilian justice system. While there are studies that have concentrated on predicting law suit lead times (5), developing systems for jurisprudence text mining (6), orautomating manual tasks (7), none have explored the intersection of time-series analysis and the Brazilian justice system.

In aninter national context, there are indeedstudies that apply time-series analysis to justice systems. However, these predominantly aim to elucidatepast trends and validate the oretical constructs through time-series data ratherthan making future predictions (8), (9). This gap in the literature under scores the novelty and potential significance of the present study.

Proposed approach for time-series forecasting

Adopting time-series forecasting techniques for predicting law suitdemands in the Brazilian justice system presents a compelling opportunity for enhancing operational efficiency.

Accurate forecasts can facilitate proactivere source allocation, reducing costs associated with unexpected law suit surges. Further more, by anticipating the volume and nature of demands, the justice system can better manage the time required for the course of legal proceedings, potentially accelerating case resolutions. Thisnotonly improves the productivity of the legal workforce but also enhances the overall efficiencyofthe justice system. Consequently, theintegrationof time-series forecastingintotheoperational framework of the Brazilian justice system could yield significant benefits, a prospect that this paper aims to explore in detail in the ensuing sections.

Algorithms

In the field of time-series forecasting, numerous algorithms have been proposed, each exhibiting unique advantages and potential drawbacks. This document comprehensively examines five algorithms: ARIMA (10), Deep AR (11), N-Beats (12), Prophet (13), and Neural Prophet (14). Each algorithm is elucidated in detail, encompassing its foundational principles, strengths, and limitations. This exposition is designed to caterto a wide audience, ranging from experienced data scientists to novices in the field. The objective is to enhance the reader's understanding of these algorithms and aid in selecting an appropriate algorithm for specific forecasting requirements. Consequently, this documentis a thorough guide into the intricate do main of time-series forecasting algorithms.

ARIMA The Auto Regressive Integrated Moving Average (ARIMA) is a forecasting method used in time-series analysis. A time series is a sequence of numerical data points taken at successive equally spaced points in time. In investing, a time series tracks the movement of the chosen data points, such as the stock price, over a specified period of time, with data points recorded at regular intervals.

ARIMA captures theauto-correlation in the data. Auto correlation is a mathematical representation of the degree of similarity between a given time series and a lagged version of itself over successive time intervals (15). It isthesame as calculating the correlation between two different time series, exceptauto-correlation uses thesame time series twice: oncein its original form and on celagged one or more time periods. The ARIMA model is typically represented as ARIMA(p, d, q) where:

- p is the order of the Auto-Regressive part. A nautoregressive (AR) model predicts future behavior based on past behavior (16). It's used for forecasting when there is some correlation between values in a time series and the values that precede and succeed them;
- d is the degree of first differencing involved. Differencing is a statistical technique that converts a non-stationary time series into a stationaryone. A stationary time series' properties do not depend on the time the series is observed, thus making it easier to forecast;
- q is the order of the Moving Average part. A moving average (MA) is a widely used indicator in technical analysis that helps smooth out

priceaction by filtering out the “noise” from random short-termpricefluctuations.

Deep AR is a forecasting method that uses recurrent neural networks (RNNs), an artificial neural network designed to recognize patterns across time. Unlike a regular neural network, which processes each input independently, RNNs have loops that allow information to be passed from one step in the network to the next (17).

Deep AR is designed to handle multiple related time series and leverage their historical information to make predictions. For example, if you wanted to predict the demand for a particular product, you could use Deep AR to analyze pastsales data for that product and related products.

N-Beats N-Beatsis a deep-learning model for time series forecasting. Deep learning is a typeof machine learning thattrains a computer to perform human-like tasks, such as recognizing speech, identifying images, or making predictions (18). In stead of organizing data to run through predefined equations, deep learning sets upbasic parameters about the data and trains the computer tolearn independently by recognizing patterns using many processing layers.

N-Beats uses a backcast and forecast mechanism. Backcasting is a way of testing a predictive model by using historical data (19). Forecasting, however, involves predicting future data points based on past and present data.

Prophetis a forecasting procedure that is based on an additive model. An additive model suggests that the components are added to make the prediction (20). This is a useful model when the components of a time series are roughly linear. Prophetisrobusttomissing data and shifts in the trend andtypicallyhandles outliers well.

The underlying model for theProphetis:
$y(t) = g(t)+s(t)+h(t)+\epsilon_t$

Where:

- g(t) represents the trend function, which models non periodic changes
- s(t) represents periodic changes (e.g., weekly, yearly)
- h(t) represents the effects of holidays
- εt represents the errorterm

Neural Prophet Neural Prophetis a neural network-based time series model builton top of PyTorch, a popular deeplearning library in Python (21). It extendsthe popular Prophet model to include more complex seasonality patterns and nonlinear trends.

NeuralProphet uses a similar additive model as Prophetbut replaces the linear orlogistic trend with a fully connected neural network. This allows it to model more complex trends. A fully connected neural network is a neural network where all the neurons in onelayer are connected to the neurons in the next layer. In a fully connected layer, eachneuron is connected to every neuron in the previous layer, andeach connection has its ownweight. Thisis a totally general purpose connection pattern and makes no as sumptions about the features in the input data, thus not using any prior knowledge about features.

The dataset: DATAJUD

DATAJUD, established by Resolution CNJ n. 331/2020, serves as Brazil’s primary data source for the Judiciary Statistics System (SIESPJ). It is a centralized repository responsible for storing data and metadata related to all physicalor electronic legal proceedings, whether public or confidential, from the courts specified in clauses II to VII ofArticle 92 ofthe Federal Constitution. The data stored in DATAJUD isused for studies and diagnostics of the Judiciary to contribute to the construction and monitoring of public policies, optimize work routines with the unification of systems, promote data integration among publicentities, and provide greater transparency to the Judiciary. This comprehensive databaseis crucial in analyzing and forecasting lawsuit demands in the Brazilian justice system.

II. METHODOLOGY

The experimental methodology adopted in this study is delineated as presented in Figure1.

Figure 1. The sequence of steps followed by the study. It starts collecting and storing the data, organizes it in a time-series way, and evaluates all five algorithms using the statistical test.

The metadata about lawsuits was gathered from the DATAJUD data base using the provided API*. We picked three different Justice Courts: the State Justice Courtof Rio Grande do Norte (TJRN), and the State Justice Courtof Minas Gerais (TJMG). The choice was based on the size of the Justice Court, with the TJMG a big courtand the TJRN a smallcourt

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 04 April 2025, pp: 1140-1147 www.ijaem.net ISSN: 2395-5252

– the former has twelve times more lawsuits than the latter.

The data, from January 1, 2018, to May 1, 2024, was subsequently aggregated daily. This aggregation facilitated the formation of a time series, which was then subjected to the Augmented Dickey-Fuller (ADF) test. The ADF test, a common statistical procedure, was employed to as certain the stationary of the time series (22). Stationary, a fundamental assumption in many time series models, implies that the statistical properties of the series do notchange over time.

The Root Mean Square Error (RMSE) was selected as the primary metric for evaluation due to its sensitivity to outliers (23). This metricprovides a robust measure of the prediction error, considering both the variance and bias of the predictions.

Subsequently, each algorithm delineated in Section was evaluated. The implementation was using Python 3.11. The evaluation was based on 30 independent executions to ensure the robustness of the results. Each algorithm was tasked with forecasting the subsequent 180 days based on the historical data. The outcomes were evaluated using the Friedman and Nemenyitests (24).

To ensure there producibility of the results, a fixed seed of 19 wasa dopted across all algorithms. The complete parametrization for each algorithm is presented in Appendix ?? for reference.

Thismethodologyaimedat a comprehensive and unbiased evaluation of the algorithms, there by providing robust insights intotheir performance in predicting the demands and productions of the Braziliancourt system.

III. EXPERIMENTS AND RESULTS

We will split the results by the dataset. Therefore, we start with TJRN.

TJRN

The Friedman test, a non-parametric statistical method, is employed todiscern differences across multiple treatment test attempts. This method, developed by Milton Friedman, ranks eachrow (or block) together and then considers the values of ranks by columns. Below, we have the obtained results after evaluation:

• statistic: 46.5867; • p-value: 1.8592e-09.

The statistic value of 46.5867 represents the test statistic calculated by the Friedman test. This value quantifiesthe deviation of the observed rank sums from the rank sums expected under thenull hypothesis, which posits no difference between the algorithms. A larger statistic value indicates a greater deviation from the null hypothesis.

The p-value of 1.8592e-09 is a function of the observed sample resultsused for testing a statistical hypothesis. Before thetest, a threshold value, known as the significance level of the test (denoted as α), ischosen. Given the extremely small p-value in this case (1.8592e-09), the null hypothesis is rejected, concluding that significant differences exist between the algorithm's performance. Table 1 presents the analysis of algorithms under the Nemeyi test.

The Table1 presents the results of the Nemenyistatistical analysis comparing the performance offive forecasting

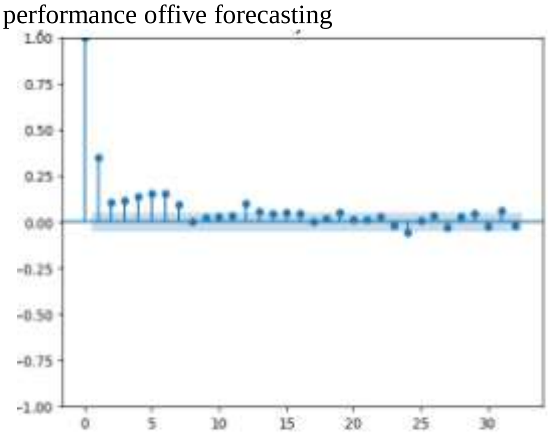

Figure 2. The partial auto-correlation method demonstrates a strong correlation with the previouslag.

algorithms: ARIMA, Prophet, neural Prophet, Deep AR, and N-Beats. The analysis, conducted using the TJRN database which is specifically designed for forecasting research, provides numerical values in the matrix that represent the statistical comparison between the algorithms.

The analysis clearly shows that the Prophet and neural Prophet algorithms stand out, surpassing the other three algorithms. Compared to ARIMA, DeepAR, and N Beats, Prophet and neural Prophet consistently demonstrated superior or comparable performance. Even though the ARIMA method did not match the effectiveness of Prophet or neural Prophet, it showed similar efficacy to both Deep A R and N-Beats.

These results have practical implications, suggesting that Prophet and neural Prophet are the two most effective algorithms for the TJRN database. This quantitative comparison offers valuable insights intotherelative performance of different forecasting methods, whichcanbe crucial for selecting appropriate forecasting models in realworld data science applications.

TJMG

Figure 2 presents the Partial Auto Correlation Function (PACF) for the TJMG dataset.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 04 April 2025, pp: 1140-1147 www.ijaem.net ISSN: 2395-5252

Figure 2 presents the Partial Autocorrelation Function (PACF) plot for the TJMG dataset. The PACF plote valuates the correlation of a time series with its lags, underscoring its importance in our analysis. In the given PACF plot, the horizontal axis represents the number of lags, extending from 0 to approximately 30. The vertical axis measures the partial auto correlation values ranging from -1.0 to 1.0.

Table 1. Comparison among the algorithms for TJRN database: The Prophetandneural Prophet algorithms surpassed Deep AR and N-Beats. The ARIMA method reached a similar performance to Deep AR and N-Beats. Therefore, the two best algorithms were Prophet and neural Prophet.

	Arima	Prophet	neuralProphet	DeepAR	-Beats
Arima	1.00000	0.00963	0.00118	0.32918	.90000
Prophet	0.00963	1.00000	0.90000	0.00100	.01263
neuralProphet	0.00118	0.90000	1.00000	0.00100	.00162
DeepAR	0.32918	0.00100	0.00100	1.00000	.28611
N-Beats	0.90000	0.01263	0.00162	0.28611	.00000

The plotreveals a significant spikeat lag 1, which exceeds the blue-shaded are a representing the confidence intervals for statistical significance. This indicates a robust partial auto correlation at lag 1, suggesting that the series' current value significantly correlates with its immediate previous value after removing the effects of intermediate lags. Beyondlag 1, all other lags upto around 30 fall with in the significance bounds, suggesting no other significant partial auto correlations at higherl ags for this dataset.

This implies that the influence of previous observations decreases rapidly with increasing lag, a characteristic of a nautoregressive process of order 1, or AR(1). Table2 presents the mean of the RMSE metric with its standard deviation obtained through 30 executions.

Table 2. RMSE meanand its standard deviation for eachevaluatedalgorithm.

	Neural	Prophet	Deep AR		N-Beats
RMSE	3.958±4.44e −16	3.441±8.881e −16	5.438±1.805	4.537±0.08	5.032±0.40

Table 2 presents the RMSE values obtained after 30 executions and its standard deviation. ARIMA and Prophet exhibit impressive consistency, with RMSE valuesof 3.958 and 3.441, respectively, accompanied by negligible standard deviations (4.44e-16 and 8.88e-16). Neural Prophet, while maintaining a reasonable RMSE of 4.538, displays more significant variability (standard deviation: 1.48e-15). Deep AR achieves a balance, achievingan RMSE of 4.537 with a minor standard deviation of 0.08. N-Beats, although less accurate (RMSE: 5.032), provide a broader prediction range (standard deviation: 0.40). In thesequence, we analyzed these results using the Friedman test.

The Friedman test resulted in two pieces of information:

- Statistic: 105.94666
- p-value: 5.3230e-22

These values confirmed that the algorithms performeddifferently, so we need to perform the Nemeyitest to rank them.

From the Table 3 we can observe that:

- The neural Prophet method shows a significant difference in performance comparedtothe ARIMA, Deep AR, and N-Beats methods, as indicated by the p values of 0.0010;
- The Prophet method also shows a significant difference in performance compared to the

ARIMA and Deep AR methods, with p-values of 0.0840 and 0.0010, respectively;

- The ARIMA and Deep AR methods show a significant difference, with a p-value of 0.0346;
- The N-Beats method shows a significant difference in performance compared to the ARIMA and neural Prophet methods, with p-values of 0.0010.

In conclusion, the Nemenyitest results suggest that the neural Prophet and Prophet methods perform significantly differently than the other methods. These results, however, should be interpreted with caution, as the significance of the difference does not necessarilyimply the superiority of one method over another in allscenarios. It's essential to be aware of the potential limitations. Other factors, such as the characteristics of the data, the computational resources available, and the specific requirements of the forecasting task, should also be considered when choosing a forecasting method. However, with only the RMSE metric as thefocus, the best algorithm was Prophet.

IV. FINAL REMARKS AND FUTURE WORKS

This study has demonstrated the effectiveness of applying time-series forecasting techniques to predict law suit demands within the Brazilian justice system. We obtained quantitative insights into their performance by evaluating five algorithms—ARIMA, Prophet, Neural Prophet, Deep AR, and N Beats—across data sets from two courts (TJRN and TJMG).

For the TJRN dataset, the Friedman test indicated significant differences among the algorithms (statistic: 46.59, p-value: 1.86e-09). The subsequent Nemenyitest showed that Prophet and Neural Prophet out performed Deep AR and N-Beats, withcomparable performance between ARIMA, DeepAR, and N-Beats. This suggests that combining classical and deep learning methods can provide robust forecasts, particularly highlighting Prophet and Neural Prophet as the most effective algorithms for this dataset.

For the TJMG dataset, the PACF analysis revealed a strong auto correlation at lag 1, indicating the importance of autoregressive models. The Friedman test confirmed the significant variance in performance (statistic: 105.95, pvalue: 5.32e-22), leading to the Nemenyitestresults, which ranked Prophet and Neural Prophetas superior to the other methods. Quantitatively, Prophet achieved an RMSE of

Table 3. Comparison among the algorithms for TJMG database: The algorithms Prophet and neural Prophet surpassed the ARIMA, DeepAR, and N-Beats. Therefore, the two best algorithms were Prophet and neural Prophet.

	Arima	Prophet	Neural Prophet	DeepAR	N-Beats
Arima	1	0.0840	0.0010	0.0346	0.0010
Prophet	0.0840	1	0.0010	0.0010	0.0010
neuralProphet	0.0010	0.0010	1	0.0126	0.90
DeepAR	0.0346	0.0010	0.0126	1	0.0681
N-Beats	0.0010	0.0010	0.90	0.0681	1

3.44 with a minimal standard deviation (8.88e-16), while Neural Prophet followed withan RMSE of 4.54, though it displayed more variability (standard deviation: 1.81). DeepARand N-Beats had higher RMSEs (4.54 and 5.03, respectively), with Deep AR showing consistency (standard deviation: 0.08) compared to N-Beats (0.40).

The sequantitative findings illustrate that, despite architectural differences, algorithms like Prophet and Neural Prophet offerrobust, low-errorpredictions, making themsuitable for operational decision-making in resource allocation and policy planning. These models' precise forecasting capabilities can lead to cost savings by better-anticipating workload surges and distributing resources accordingly, ultimately contributing to the efficiency of the justice system.

Further research should consider including external variables, such as socioeconomic factors, to enhance prediction accuracy. Additionally, applying these models to other branches of the judiciary could validate the generalizability of the findings and address broader systemic issues.

Overall, this study under scores the potential of time-series forecasting to improve strategic planning within the Brazilian justice system, driving the integration of data-driven insights for enhanced judicial efficiency.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 04 April 2025, pp: 1140-1147 www.ijaem.net ISSN: 2395-5252

REFERENCES

[1]. Da Ros, L. and M. M. Taylor. Ju´ızes eficientes, judiciario´ ineficiente no Brasil pos-1988.´BIB-Revista Brasileira de Informac¸aoBibliogr˜ afica em Ci´ enciasSociaisˆ , , No. 89, 2019, pp. 1–31.

[2]. de Sena Orsini, A. G. Jurimetria e predic¸ao: notas sobre uso dos˜ algoritmos e o Poder Judiciario.´RDUno: Revista do Programa de Pos-Graduac¸´ ao em Direito da Unochapec˜ o´, Vol. 3, No. 4, 2020, pp. 33–50.

[3]. Neves Junior, P. C. Inovac¸ao e governanc¸a no planejamento,˜ na execuc¸ao e no controle da atividade financeira do poder˜ judiciario no Brasil´ . Ph.D. thesis, Universidade de SaoPaulo,˜ 2020.

[4]. Liang, Y., H. Wen, Y. Nie, Y. Jiang, M. Jin, D. Song, S. Pan, and Q. Wen. Foundation models for time series analysis: A tutorial andsurvey. In Proceedingsofthe 30th ACM SIGKDD ConferenceonKnowledge Discovery and Data Mining. 2024, pp. 6555–6565.

[5]. Gruginskie, L. A. d. S. and G. L. R. Vaccaro. Lawsuit lead time prediction: Comparisonof data mining techniquesbasedoncategorical response variable. PloSone, Vol. 13, No. 6, 2018, p. e0198122.

[6]. Dias Canedo, E., V. Aymore Martins, V. Coelho Ribeiro,´ V. E. dos Reis, L. A. Carvalho Chaves, R. Machado Gravina,

[7]. F. Alberto Moreira Dias, F. L. Lopes de Mendonc¸a, A. L. S. Orozco, R. Balaniuk, et al. Developmentandevaluationofanintelligencea nd learning system in jurisprudencetext mining in thefieldofcompetitiondefense. Applied Sciences, Vol. 11, No. 23, 2021, p. 11365.

[8]. Limberger, T., D. B. d. S. Giannakos, and M. M. Szinvelski. CanJudgesbeReplacedbyMachines? The Brazilian Case.

[9]. Mexicanlaw review, Vol. 14, No. 2, 2022, pp. 53–81.

[10]. Fry, C. E., T. G. McGuire, and R. G. Frank. Medicaidexpansion'sspillovertothe criminal justice system: evidencefromsixurbancounties. RSF: The Russell Sage Foundation Journalofthe Social Sciences, Vol. 6, No. 2, 2020, pp. 244– 263.

[11]. Kuettel, B. T. IncreasingLiberalization: A Time Series

[12]. AnalysisofthePublic'sMoodtowardDrugs. Justice Quarterly, Vol. 41, No. 4, 2024, pp. 475–493.

[13]. Ho, S. L. and M. Xie. The use of ARIMA models for reliabilityforecastingandanalysis. Computers& industrial engineering, Vol. 35, No. 1-2, 1998, pp. 213–216.

[14]. Salinas, D., V. Flunkert, J. Gasthaus, and T. Januschowski. DeepAR: Probabilisticforecastingwithautoregressivere current networks. Internationaljournalofforecasting, Vol. 36, No. 3, 2020, pp. 1181–1191.

[15]. Oreshkin, B. N., D. Carpov, N. Chapados, and Y. Bengio. NBEATS: Neural basisexpansionanalysis for interpretable time series forecasting. arXivpreprint arXiv:1905.10437.

[16]. Taylor, S. J. and B. Letham. Forecastingatscale. The American Statistician, Vol. 72, No. 1, 2018, pp. 37–45.

[17]. Triebe, O., H. Hewamalage, P. Pilyugina, N. Laptev, C. Bergmeir, and R. Rajagopal. Neuralprophet: Explainableforecastingatscale. arXivpreprint arXiv:2111.15397.

[18]. Bartlett, M. S. Onthetheoreticalspecificationandsamplingpr opertiesofautocorrelated time-series. SupplementtotheJournalofthe Royal Statistical Society, Vol. 8, No. 1, 1946, pp. 27–41.

[19]. Fuller, W. A. and D. P. Hasza. Properties ofpredictors for autoregressive time series. Journalofthe American StatisticalAssociation, Vol. 76, No. 373, 1981, pp. 155–161.

[20]. Medsker, L. R., L. Jain, et al. Recurrentneural networks.

[21]. Design andApplications, Vol. 5, No. 64-67, 2001, p. 2.

[22]. LeCun, Y., Y. Bengio, and G. Hinton. Deep learning. nature, Vol. 521, No. 7553, 2015, pp. 436–444.

[23]. Dreborg, K. H. Essenceofbackcasting. Futures, Vol. 28, No. 9, 1996, pp. 813–828.

[24]. Hastie, T. J. Generalizedadditive models. In Statistical models in S. Routledge, 2017, pp. 249–307.

[25]. Paszke, A., S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: Animperativestyle, high-performance deep learning library. Advances in neural informationprocessing systems, Vol. 32.

[26]. Cheung, Y.-W. and K. S. Lai. LagorderandcriticalvaluesoftheaugmentedDi ckey–Fuller test. Journalof Business &

Economic Statistics, Vol. 13, No. 3, 1995, pp. 277–280.
[27]. Hodson, T. O. Root meansquareerror (RMSE) ormeanabsoluteerror (MAE): When to use themornot. Geoscientific Model DevelopmentDiscussions, Vol. 2022, 2022, pp. 1–10.
[28]. Pereira, D. G., A. Afonso, and F. M. Medeiros. Overview ofFriedman'stestand post-hoc analysis. Communications in Statistics-SimulationandComputation, Vol. 44, No. 10, 2015, pp. 2636–2653.