Norm-Cosine for Convex Combinations vs XGBoost on Climate Prediction of Belo Horizonte and Shenzhen

Luiz Fernando Ramos Lemos
Inst. of Prod. Eng. & Management
Federal University of Itajubá
Itajubá, MG, Brazil
luiz.lemos@ifsuldeminas.edu.br

Gabriel Victor de Lima
Inst. of Prod. Eng. & Management
Federal University of Itajubá
Itajubá, MG, Brazil
d2025101099@unifei.edu.br

Matheus Brendon Francisco
Inst. of Prod. Eng. & Management
Federal University of Itajubá
Itajubá, MG, Brazil
matheus_brendon@unifei.edu.br

Rafael de Magalhães Dias Frinhani
Inst. of Mathematics & Computing
Federal University of Itajubá
Itajubá, MG, Brazil
frinhani@unifei.edu.br

José Arnaldo Barra Montevechi
Inst. of Prod. Eng. & Management
Federal University of Itajubá
Itajubá, MG, Brazil
montevechi@unifei.edu.br

Anderson Paulo de Paiva
Inst. of Prod. Eng. & Management
Federal University of Itajubá
Itajubá, MG, Brazil
andersonppaiva@unifei.edu.br

***Abstract*—This work proposes the NC-CC (Norm-Cosine for Convex Combination) method as an approach for multi-horizon forecasting of climate time series, comparing it with the XGBoost model. NC-CC identifies the historical time point most similar to the current one using a hybrid metric based on Euclidean distance and cosine similarity between climate variation vectors and forecasts the next k days through a convex combination of the current and analogous variations. We use daily temperature, precipitation, and wind direction data from 2019 to 2024 for the cities of Belo Horizonte (Brazil) and Shenzhen (China). We evaluate NC-CC and XBoost in 1,000 random trials, with forecast horizons of 1, 7, 15, and 30 days, considering quality metrics such as Root Mean-Square Error (RMSE), Mean Absolute Percentage Error (MAPE), RMSE/σ and accuracy within absolute thresholds. The results show that NC-CC outperforms XGBoost in terms of scalability and in tracking abrupt variations, whereas XGBoost tends to minimize quadratic error, yielding smoother forecasts.**

***Index Terms*—time series forecasting, climate prediction, XGBoost, NC-CC, convex combination, vector similarity.**

I. Introduction

Forecasting time–series variables—temperature, precipitation, economic indicators, physiological signals—underpins decision-making in climate science, economics and health. With the spread of sensors and long historical records, methods that capture seasonality, structural breaks and abrupt shifts are increasingly valuable.

Classical models such as ARIMA and deep-learning architectures like LSTM remain popular, while gradient-boosted trees—particularly XGBoost—provide high accuracy with modest training time [1]. Yet XGBoost's global regularisation can smooth local extremes [2], [3]. In contrast, history-based analogues (e.g. k-NN) reproduce local and rare patterns but are sensitive to noise [4], [5], highlighting the trade-off between robustness and fidelity.

We bridge this gap with NC-CC (*Norm–Cosine for Convex Combination*). The method retrieves the historical day most similar to the present via a hybrid Euclidean–cosine metric on change vectors and projects future values through a convex blend of current and historical variations. NC-CC is training-free, lightweight and interpretable. We benchmark it against XGBoost on daily climate series from two contrasting cities—dry, temperate Belo Horizonte (Brazil) and highly variable Shenzhen (China)—and offer four contributions:

- a history-based forecasting framework built on change-vector similarity;
- a hybrid norm–cosine metric balancing magnitude and direction;
- a zero-training implementation suitable for resource-constrained devices;
- an empirical study showing that NC-CC matches or surpasses XGBoost in capturing local extremes while scaling favourably.

The remainder of the paper is organised as follows. section II reviews related work; section III presents NC-CC, experimental setup and dataet description; section IV presents and discusses the results; and section V concludes the study and outlines future research.

II. Theoretical Foundation

Climate time series forecasting is a classical problem with critical applications in various domains such as agriculture, public health, water management, energy, and disaster mitigation [6]. Beyond meteorology, applications also extend to finance [7], industrial monitoring, and demand forecasting [8],

all of which require robust forecasting models capable of handling complex dynamics, structural breaks, and irregular patterns.

Traditional autoregressive models, including AR, ARMA, and ARIMA, have been widely used for modeling temporal dependencies [7]. However, their assumptions of linearity and stationarity often limit performance in highly nonlinear environments such as climate systems. Extensions like ARIMAX and hybrid ARIMA–machine learning models have sought to address some of these limitations [8].

Machine learning approaches, particularly ensemble methods such as XGBoost [1], [9], have gained popularity due to their ability to capture nonlinear patterns, strong regularization capabilities, and high computational efficiency. Recently, deep learning architectures — such as Long Short-Term Memory (LSTM) networks and Transformer-based models — have also demonstrated strong predictive performance in time series forecasting [10], [11]. However, these models typically require large datasets and substantial computational resources.

Parallel to these supervised approaches, analog forecasting methods have seen renewed interest due to their interpretability and data-driven nature [12]. These approaches rely on retrieving past conditions most similar to the current state and projecting future evolution accordingly. Closely related are k-Nearest Neighbors (kNN) techniques for time series forecasting, which also use distance-based retrieval of similar historical patterns. Most of these methods, however, adopt simple similarity metrics such as Euclidean distance.

III. Methodology: NC-CC Formulation and Experimental Setup

The research conducted in this paper is of an experimental and comparative nature, employing qualitative and quantitative approaches with an exploratory objective, which aims to develop a predictive model using climate data. The research integrates building and experimental methodologies aimed respectively at the development and improvement of the model, as well as its empirical validation.

The hypothesis is that the method we developed, called Norm-Cosine for Convex Combination (NC-CC), is capable of obtaining better quality forecasts in most cases compared to XGBoost. The specific objectives include developing the NC-CC method, comparing its performance with XGBoost using accuracy metrics for different horizons, and evaluating the generalization capabilities of the NC-CC in contrasting climates.

A. NC-CC Architecture

The NC-CC is a history-based method that incorporates both norm-based and angular similarity (cosine) into the pattern retrieval process, combining the advantages of magnitude and directional information for forecasting. Furthermore, it introduces convex combinations of historical variations, a relatively underexplored strategy in the existing time series literature, that enables more flexible multi-step forecasts while remaining computationally lightweight and unsupervised.

We divided the mathematical foundation of the NC-CC into two parts: (*i*) finding in the data history the moment most similar to the current one using norms and cosines, and then (*ii*) using the obtained data to compute the next moment through a convex combination. It is worth noting that the method can be fitted as a dynamic autoregressive model with a single lag, analogous to a k-NN model with a single neighbor. The NC-CC can reconstruct larger forecast windows using more terms or larger blocks. However, in this work, we utilize data from a single day in the present and another single day in the past to sequentially forecast future days.

As a similarity metric for selecting data from the past, we use a mixed formula that takes into account the norm and the cosine between the vectors representing the current and past moments. The similarity (p_n) is described by the Equation 1, where v represents the current day and w a past moment under evaluation:

$$p_n = 1 - \frac{||v - w||}{||v|| + ||w||} \tag{1}$$

As we are dealing with the ratio of two non-negative numbers greater than value zero, and with the ratio being at most value one due to the triangle inequality, we include these limits as shown in the Equation 2:

$$0 \leq \frac{||v - w||}{||v|| + ||w||} \leq 1, \tag{2}$$

The previous equation is obtained considering $z = -w$, in the triangular inequation expressed in Equations 3, resulting in 4, so that the maximum occurs when $-w = v$, i.e., w is opposite to v.

$$||v + z|| \leq ||v|| + ||z||, \tag{3}$$

$$||v - w|| \leq ||v|| + ||w|| \Rightarrow \frac{||v - w||}{||v|| + ||w||} \leq 1 \tag{4}$$

To maintain the same pattern used with cosine, we adopt the Equation 5, which inverts the maximum and minimum scenario.

$$p_n = 1 - \frac{||v - w||}{||v|| + ||w||} \tag{5}$$

Another similarity metric used was the cosine, which indicates how angularly close the two vectors are, with 1 being the maximum angular similarity and -1 representing collinearity but in the opposite direction. The general formula for cosine is described in Equation 6:

$$p_c = cos(v, w) = \frac{< v, w >}{||v|| \cdot ||w||} \tag{6}$$

The cosine metric is scale-insensitive, which is why we chose to embed a calculation that is sensitive to the norm. Thus, we define the Equation 7:

$$p_p = p_n \cdot p_c = \left(1 - \frac{||v - w||}{||v|| + ||w||}\right) \cdot \left(\frac{< v, w >}{||v|| \cdot ||w||}\right) \quad (7)$$

where p_p is the past moment weight, used both to select the most similar scenario (the one with the highest p_p) and in the iterative step by the convex combination; p_n is the norm based weight; and p_c is the cosine weight.

Starting from the current moment and the most similar past moment selected by weight, we perform an Iterative step, which consists of a variation obtained through the convex combination of the current variation with the past variation. Consider v the vector of the present moment and the immediately previous vector v_{-1}; we define the auxiliary variation as $\delta_a = v - v_{-1}$. This variation is used considering that we do not have data for future variation and that the variation is stationary, which is a hypothesis for the proper functioning of the model; usually, this variation will have very little influence since another hypothesis for good performance is the existence of regular behavior in the past, which leads to high weights in the composition with the past scenario.

Considering the past as w and the moment immediately after the past scenario as w_{+1}, we use the subsequent moment since we assume that the variation of the current moment will be similar to that which occurred in the past. In this case, the degree of similarity is measured by the same similarity measure that determined the selection. This variation is given by the Equation 8:

$$\delta_w = w_{+1} - w \quad (8)$$

The variation from the current moment to the next (v_{+1}) is defined by the convex combination show in Equation 9, where $p_n = (1 - p_p)$ is the weight of the moment in present. The next moment is given by Equation 10:

$$\delta_v = (1 - p_p) \cdot \delta_a + p_p \cdot \delta_w \quad (9)$$

$$v_{+1} = v + \delta_v \quad (10)$$

Subsequent moments can be obtained in iterative manner, starting from the newly obtained moment.

B. Experiment Design

1) Hardware and Software Parameters: The experiments were conducted on a Dell Inspiron 15 5510 (Intel Core i7-11390H 11th generation, 8 threads, 3.40 GHz), 15.4 GiB of RAM, running openSUSE Tumbleweed 20250306 (kernel 6.12.20-1-longterm, KDE Plasma 6.3.2), and JupyterLab (Python 3.12), with GPU acceleration provided by an NVIDIA GeForce MX450 (2048 MB, 896 CUDA cores). Numerical processing was carried out using NumPy 2 and TensorFlow 2. The NC-CC method was implemented using TensorFlow, pre-calculating norms and normalizing vectors to speed up evaluation—a phase we call preprocessing. XGBoost training was executed with a single parameter`reg:squarederror`.

2) Experiments: For each location, the data from 2019 to 2023 was defined as the training set, and the data from 2024 as the validation set. A deterministic 30-day forecast was performed starting from an arbitrary day 123 in 2024. Was conducted 1,000 random trials with horizons $k = \{1, 7, 15, 30\}$ days, whose results are presented in Section IV. The experiment conditions were analogous for both methods in Belo Horizonte and Shenzhen. Illustratively, we highlight a 30-day forecast starting on day 170 (a persistently dry period in BH) and on day 225 in SZ (a highly variable period) to compare model behaviors as can be seen in Figures 1 and 3.

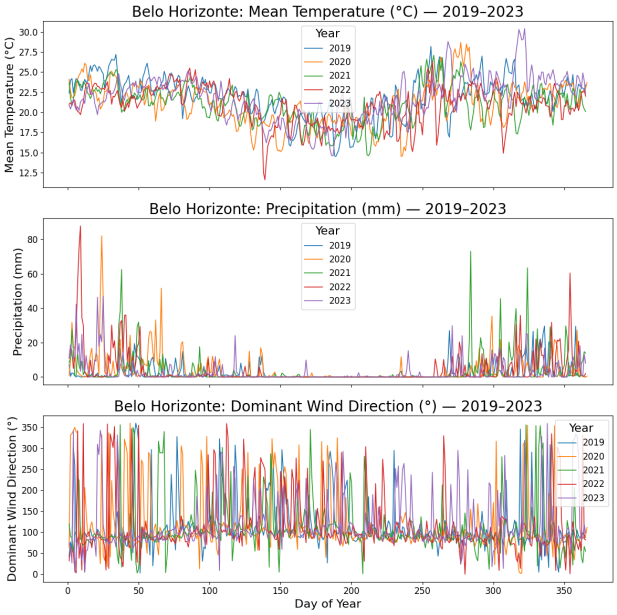


Fig. 1. Time series of mean temperature (°C), precipitation (mm), and dominant wind direction (°) for the Belo Horizonte dataset from 2019 to 2023.

3) Data Acquisition and Preprocessing: Daily historical climate data for Belo Horizonte (Brazil) and Shenzhen (China) were retrieved from the Open-Meteo archive API. For both cities, the period January 1, 2019 to December 31, 2023 was obtained, and for Belo Horizonte an additional test period from January 1, 2024 to December 31, 2024. Each API call requested daily values of `temperature_2m_mean`, `rain_sum`, and `winddirection_10m_dominant`, returned in the local time zones (`America/Sao_Paulo` and `Asia/Shanghai`). The JSON responses were parsed into `pandas` DataFrames, with `time` converted to a datetime index and columns renamed to `meantemp`, `rain`, and `wdir_dominant`, using only the `requests`, `pandas`, and `NumPy` libraries.

To ensure uninterrupted daily records, each DataFrame was reindexed against a complete date range for its period, and an

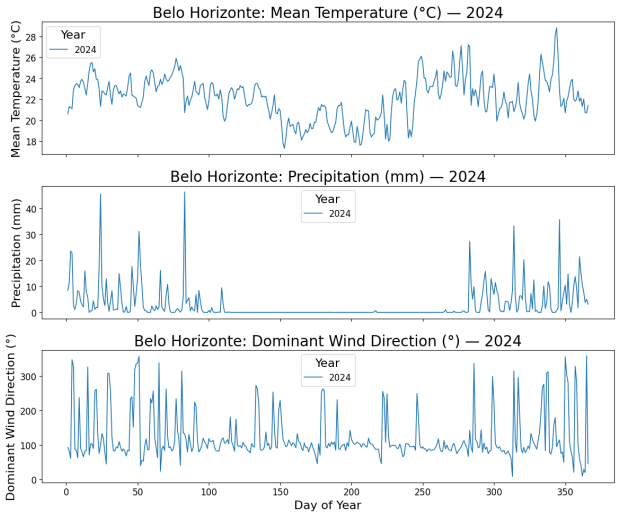


Fig. 2. Time series of mean temperature (°C), precipitation (mm), and dominant wind direction (°) for the Belo Horizonte dataset in 2024.

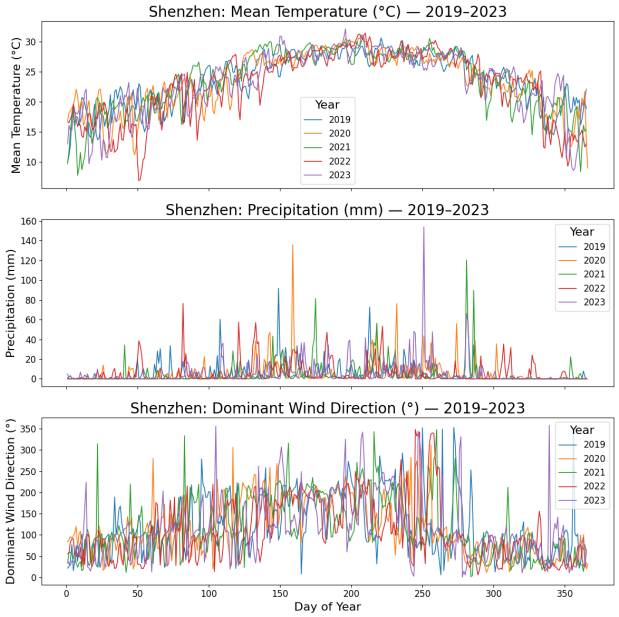


Fig. 3. Time series of mean temperature (°C), precipitation (mm), and dominant wind direction (°) for the Shenzhen dataset from 2019 to 2023.

assertion confirmed zero missing entries.

IV. Results and Discussion

The comparison between the proposed NC-CC method and XGBoost was conducted using classical regression metrics for time series and predictive models, including RMSE (Root Mean Square Error), MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error), and the RMSE/σ ratio. We also present comparative execution-time benchmarks - Table II and Table I based on the observed runtimes of the datasets across different forecasting horizons, demonstrating the scalability of the NC-CC method relative to XGBoost.

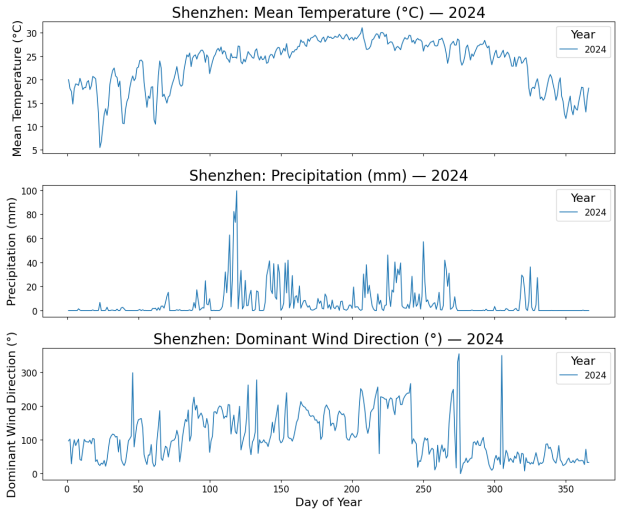


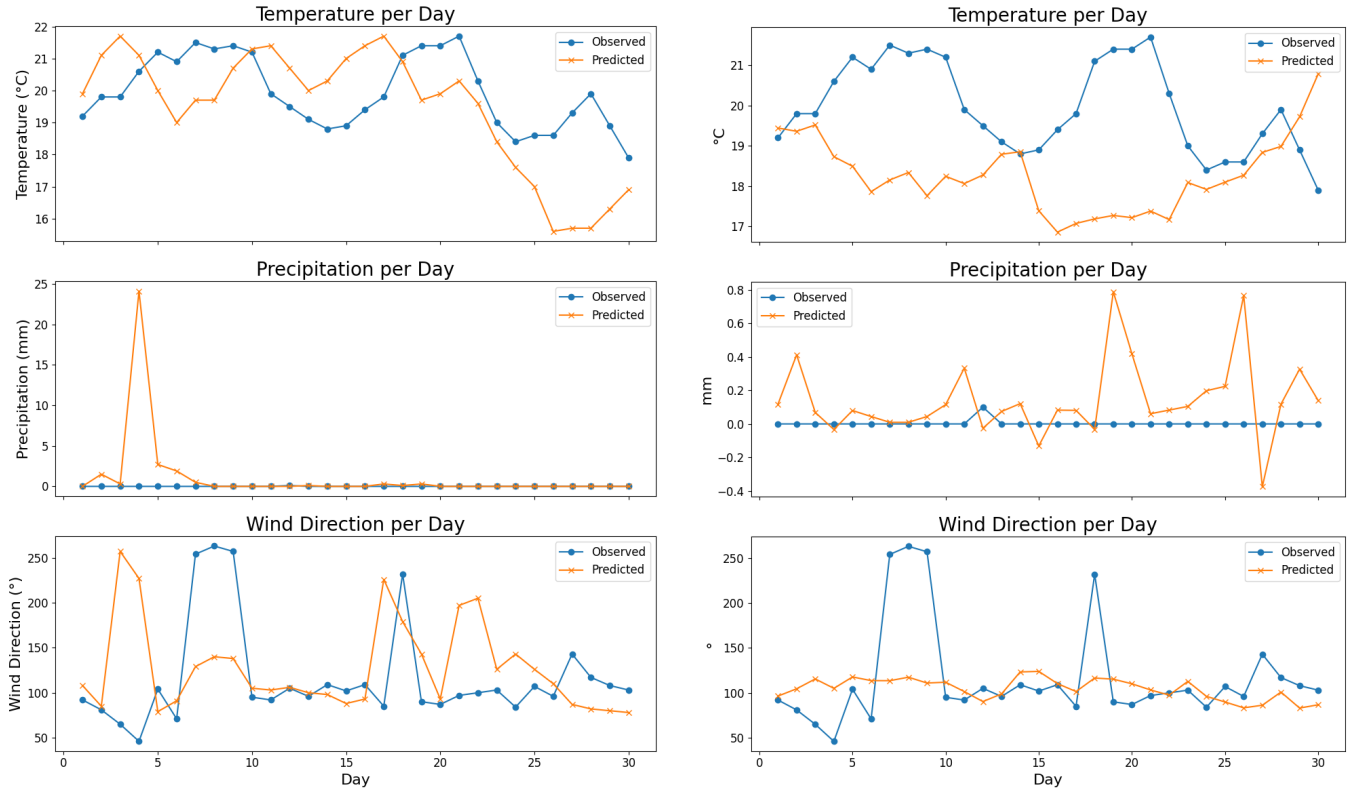
Fig. 4. Time series of mean temperature (°C), precipitation (mm), and dominant wind direction (°) for the Shenzhen dataset in 2024.

Figure 5 and Figure 6 display the graphics of Temperature, Precipitation, and Wind Direction, actual and forecasted values generated by the NC-CC and XGBoost methods, based on 30 days of climate data from Belo Horizonte - day 170 - and Shenzhen - day 225 - cities. The tables show accuracy rates, error metrics, and performance across multiple prediction horizons for temperature and precipitation. Results are analyzed for each city, highlighting the behavior under distinct climatic conditions.

Additionally, domain-specific metrics were considered, such as the percentage of predictions within $\pm 1^{\circ}$C for temperature, $\leq$ 1mm for precipitation an $\pm 30^{\circ}$ for main wind direction. In all tables, the best values are highlighted in bold and blue. In Accuracy Rates, the best values are maximum. In the error metrics RMSE, MAE, MAPE and RMSE/sigma, the best value is the minimum, as it quantifies the error between the predicted and observed values. RMSE penalizes larger errors more heavily because it is based on the square of the differences, while MAE calculates the average of absolute differences, making it more robust to outliers. MAPE expresses the average error in percentage terms, which facilitates interpretation relative to the actual value. The RMSE/σ ratio compares the model's error to the natural variability of the data; the smaller this ratio, the greater the model's ability to explain the observed variation. Therefore, low values in these metrics indicate more accurate predictions and higher model quality.

A. Temperature Curve Adherence

As illustrated in Figures 5(a) and 5(b), and Figures 6(a) and 6(b), both NC-CC and XGBoost demonstrate comparable ability to follow actual temperature fluctuations in both cities. However, NC-CC more closely tracked the variations, while XGBoost produced smoother forecasts, which is consistent with its bias toward minimising squared error; however, NC-CC achieved the lower RMSE in Belo Horizonte while XG-



(a) NC-CC 30-day forecast starting day 170

(b) XGBoost 30-day forecast starting day 170

Fig. 5. Comparison of thirty-day forecasts for Belo Horizonte starting on day 170: (a) NC-CC method; (b) XGBoost. Each panel shows temperature (°C), precipitation (mm), and wind direction (°) per day.

TABLE I
ERROR METRICS AND TOLERANCE RATES FOR 30-DAY FORECASTS (HORIZON = 30) FROM DAY 170 ON THE BELO HORIZONTE 2024 DATASET AND FROM DAY 225 ON THE SHENZHEN 2024 DATASET

Metric	**Belo Horizonte (day 170)**		**Shenzhen (day 225)**	
	NC-CC	**XGBoost**	**NC-CC**	**XGBoost**
Temperature				
RMSE (°C)	**1.78**	1.96	0.82	**0.78**
MAE (°C)	**1.52**	1.96	**0.64**	0.78
MAPE (%)	**7.71**	9.53	**2.29**	2.78
%±1°C	33.33	**40.00**	**83.33**	73.33
Precipitation				
RMSE (mm)	4.45	**0.184**	18.40	**11.60**
MAE (mm)	1.06	**0.184**	12.40	**11.60**
SMAPE (%)	**73.33**	200.00	105.00	**94.60**
%≤ 1 mm	86.67	**100.00**	**96.67**	**96.67**
Wind Direction				
RMSE (°)	74.00	**35.40**	90.50	**80.10**
MAE (°)	52.10	**35.40**	**72.30**	80.10
MAPE (%)	28.94	**19.66**	**40.16**	44.52
%±30°	56.67	**73.33**	**33.33**	20.00

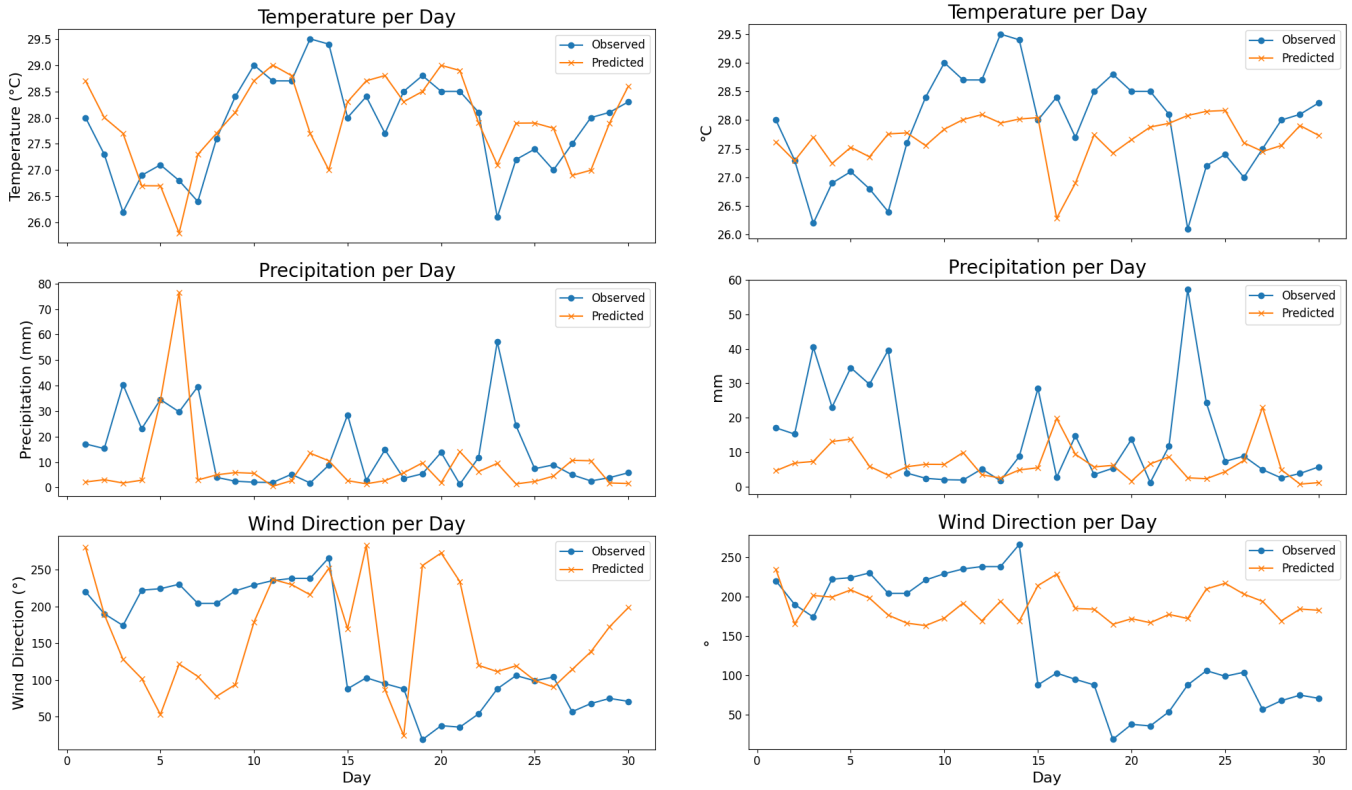
Boost edged ahead in Shenzhen. In Belo Horizonte, around the day 20 of the forecast window, the real series show a sharp drop on temperature; NC-CC follow this inflections closely as well the pattern of the curves looks more as the real pattern, while XGBoost looks more smooth, as seen in Figure 5(b).

In Shenzhen - day 225 - where climate variability is markedly higher, the contrasting behaviors of the two methods became more pronounced. The NC-CC method showed a stronger adherence to local variations in temperature, more closely mimicking the real fluctuations. This is particularly evident in the temperature curve, where NC-CC achieved better results in MAE and MAPE, as well as a higher proportion of predictions within a ±1°C tolerance (83.33% vs. 73.33%).

On the other hand, XGBoost generated smoother predictions, showing a tendency to regularize abrupt changes. While this smoothing effect contributed to slightly better RMSE performance (0.78°C vs. 0.82°C), it came at the cost of reduced sensitivity to local deviations — a pattern also visible in the precipitation and wind components. These observations are consistent with the theoretical design of each model: NC-CC, being history-based and data-driven, tends to preserve localized patterns; whereas XGBoost, through its ensemble regularization, emphasizes global error minimization, resulting in more stable but less reactive forecasts.

B. Precipitation: Rare Peaks and Overestimation

In Belo Horizonte, precipitation forecasting highlighted a key distinction between the models: while NC-CC captured the general absence of rainfall by correctly classifying 86.67% of the days as rainy or not, it exhibited a notable overestimation



(a) NC-CC 30-day forecast for Shenzhen (start day 225)

(b) XGBoost 30-day forecast for Shenzhen (start day 225)

Fig. 6. Comparison of 30-day climate forecasts for Shenzhen beginning on day 225, using (a) the NC-CC method and (b) XGBoost.

early in the forecast, predicting a sharp peak that did not occur (Table I). After this initial error, NC-CC consistently predicted zero precipitation, aligning well with the actual dry pattern. In contrast, XGBoost avoided such peaks and delivered perfect binary classification accuracy (100.00%), with mild oscillations around zero that proved more effective in dry conditions.

In Shenzhen, where rainfall is more frequent and intense, both models correctly identified rainfall presence in 96.67% of the cases under the 1mm tolerance criterion (Table I). However, NC-CC again generated a high peak, reflecting its tendency to overreact to isolated patterns. This behavior contributed to its higher RMSE (18.40,mm vs. 11.60,mm) and SMAPE (105.00% vs. 94.60%) when compared to XGBoost, which, although imperfect in tracking peak intensities, maintained more consistent and moderate estimates. The results suggest that XGBoost handles rainfall magnitude better in both dry and wet conditions, while NC-CC may be more suitable when the goal is to detect rain occurrence, even if not its volume.

C. Wind Direction: Responsiveness to Oscillations

Wind direction, being a circular variable, demands special consideration to avoid distortion near angular boundaries. In this study, all metrics were computed using minimal arc distances, but neither NC-CC nor XGBoost was designed to model circularity explicitly. In Belo Horizonte, XGBoost outperformed NC-CC in all error metrics — RMSE (35.40° vs. 74.00°), MAE (35.40° vs. 52.10°), and MAPE (19.66% vs. 28.94%) — and achieved a higher rate of predictions within $\pm 30°$ (73.33% vs. 56.67%) (Table I). Its outputs exhibited smoothed directional curves, favoring general trends but often missing abrupt shifts.

In Shenzhen, where directional variability was more intense, XGBoost again achieved the lowest RMSE (80.10° vs. 90.50°), but NC-CC was more accurate in terms of MAE (72.30° vs. 80.10°), MAPE (40.16% vs. 44.52%), and notably achieved a higher hit rate within $\pm 30°$ (33.33% vs. 20.00%) (Table I). These results suggest that XGBoost minimized large outliers, but NC-CC better captured frequent, smaller variations, offering greater adherence to the path of directional changes. Still, both methods would likely benefit from modeling approaches that explicitly account for angular continuity, such as representing directions via sine and cosine components.

D. Integration Between Metrics and Curve Behavior

When integrating quantitative metrics with the forecasted curve behaviors, we observe important trade-offs between the two methods. In Belo Horizonte, both models present normalized RMSEs above 1 for the 30-day temperature forecast (Table II), indicating errors greater than the natural variability: 1.40 for NC-CC and 1.46 for XGBoost. Although this suggests that neither model accurately tracks long-term variability in

TABLE II

NC-CC AND XGBOOST — TEMPERATURE FORECASTING METRICS FOR DIFFERENT HORIZONS IN **BELO HORIZONTE** AND **SHENZHEN**. TRAINING AND EVALUATION TIMES IN SECONDS.

	Belo Horizonte									
	NC-CC (Preprocessing: 0.00194 s)					**XGBoost** (Training: 0.12373 s)				
Days	**RMSE±SD**	**MAE±SD**	**MAPE (%)**	**RMSE/σ**	**Eval. (s)**	**RMSE±SD**	**MAE±SD**	**MAPE (%)**	**RMSE/σ**	**Eval. (s)**
1	0.98 ± 0.90	0.98 ± 0.90	4.41	0.49	1.35228	1.16 ± 0.80	0.85 ± 0.80	3.77	0.58	0.40340
7	2.17 ± 1.32	1.86 ± 1.21	8.37	1.07	2.46124	2.15 ± 1.04	1.62 ± 0.94	7.17	1.06	1.62305
15	2.53 ± 1.11	2.11 ± 1.00	9.54	1.23	4.06966	2.57 ± 1.06	1.98 ± 0.98	8.73	1.26	3.35958
30	2.82 ± 1.01	2.31 ± 0.91	10.44	1.40	7.51244	2.97 ± 1.11	2.33 ± 1.02	10.22	1.46	7.18469
	Shenzhen									
	NC-CC (Preprocessing: 0.01288 s)					**XGBoost** (Training: 0.12740 s)				
Days	**RMSE±SD**	**MAE±SD**	**MAPE (%)**	**RMSE/σ**	**Eval. (s)**	**RMSE±SD**	**MAE±SD**	**MAPE (%)**	**RMSE/σ**	**Eval. (s)**
1	1.43 ± 1.40	1.43 ± 1.40	7.40	0.28	1.12590	1.54 ± 1.08	1.09 ± 1.08	5.47	0.31	0.36752
7	2.96 ± 2.10	2.54 ± 1.93	13.40	0.61	2.40988	3.25 ± 1.80	2.34 ± 1.59	12.50	0.64	2.30697
15	3.46 ± 2.08	2.88 ± 1.79	14.88	0.69	4.32344	3.80 ± 1.88	2.80 ± 1.64	14.56	0.75	5.43177
30	3.99 ± 2.22	3.30 ± 1.93	16.10	0.84	7.23022	4.19 ± 1.91	3.17 ± 1.74	15.25	0.88	8.19315

TABLE III

PERCENTAGE OF FORECASTS WITHIN TOLERANCES — ±1°C, ≤1MM AND ±30° — FOR DIFFERENT HORIZONS IN **BELO HORIZONTE** AND **SHENZHEN**.

Days	**Temp. (±1°C)**		**Prec. (≤1mm)**		**Wind (±30°)**	
	NC-CC (%)	**XGBoost (%)**	**NC-CC (%)**	**XGBoost (%)**	**NC-CC (%)**	**XGBoost (%)**
			Belo Horizonte			
1	61.10	69.70	75.20	71.80	60.20	62.80
7	38.70	42.81	60.43	57.51	54.26	62.09
15	32.75	34.69	61.19	57.20	50.75	59.91
30	29.83	29.10	60.29	56.94	52.53	61.50
			Shenzhen			
1	49.40	59.20	73.20	76.50	54.90	63.80
7	31.96	35.07	65.96	61.26	38.89	40.29
15	28.67	29.30	62.95	56.51	34.46	37.45
30	25.94	26.25	59.92	55.47	32.08	32.35

absolute terms, across the 30-day window NC-CC delivers the lowest RMSE at 1, 15 and 30 days and the best RMSE/σ on three of the four horizons, while XGBoost keeps a modest advantage in MAE on the shorter horizons. In Shenzhen, the relative performance improves for both methods, with RMSE/σ remaining below 1 for all horizons (Table II). NC-CC outperforms XGBoost at every horizon in this metric — for example, at 30 days: 0.84 for NC-CC versus 0.88 for XGBoost — indicating a better capacity to capture temperature variability over time.

Considering the percentage of predictions within predefined tolerance thresholds — $\pm 1,^{\circ}$ C for temperature, $\leq$ 1,mm for precipitation, and $\pm 30^{\circ}$ for wind — different patterns emerge (Table III). For temperature, XGBoost has the higher ±1 °C hit-rate over the first three horizons (e.g., 42.81 % vs 38.70% at 7 days), while NC-CC edges ahead at 30 days (29.83 % vs 29.10 %), as seen on (Table III). This suggests a tendency of XGBoost to produce predictions more concentrated around observed values, a pattern reflected in its higher hit-rate inside the ±1 °C band. On the other hand, for precipitation, NC-CC outperforms XGBoost in most horizons and in both cities, particularly in Belo Horizonte: at the 30-day mark, NC-CC achieves 60.29% of correct classifications (rain/no-rain) within the 1,mm threshold, compared to 56.94% for XGBoost (Table III). This aligns with the visual inspection in Figure 5(a), where NC-CC effectively captures extended dry periods despite overestimating early in the horizon. For wind direction, XGBoost demonstrates a clear advantage in most cases, with higher tolerance rates in both locations. At 30 days in Belo Horizonte, it reaches 61.50% within $\pm 30^{\circ}$, while NC-CC attains 52.53% (Table III), reflecting XGBoost's smoother behavior in angular trends.

In summary, NC-CC tends to better preserve climate structure and variability, especially in temperature and precipitation under longer horizons, while XGBoost excels in generating concentrated forecasts that improve hit rates within strict tolerances, particularly in more stable variables such as wind direction. The choice between methods thus depends on whether the forecasting goal prioritizes structural adherence and variability (favoring NC-CC) or local precision and error minimization (favoring XGBoost).

V. Conclusion

This paper presented NC-CC (*Norm–Cosine for Convex Combination*) as an unsupervised, history-based strategy for multi-step climate prediction and compared it with the well-established XGBoost. Using six years of daily records for Belo Horizonte (Brazil) and Shenzhen (China), we ran 1 000 Monte-Carlo trials over four prediction horizons ($k = 1, 7, 15, 30$).

Across the longer horizons ($k \geq 15$), NC-CC consistently matched or surpassed XGBoost in RMSE, MAE and MAPE for temperature and precipitation, while keeping RMSE $/\sigma <$ 1 on the more volatile Shenzhen series. At the one-day horizon, however, XGBoost retained a slight edge in absolute errors and in the proportion of temperature estimates within the $\pm 1^{\circ}$C band, underscoring its strength in very short-term regression. For precipitation, NC-CC proved more reliable in flagging rain events within the ≤ 1 mm tolerance, whereas XGBoost provided smaller volumetric errors—revealing a clear trade-off between detecting an event and estimating its magnitude. A similar pattern emerged in wind direction: XGBoost produced smoother angular trajectories and therefore higher hit rates inside the $\pm 30^{\circ}$ window, but NC-CC followed rapid oscillations more closely, albeit with larger outliers.

Beyond accuracy, the two models diverge sharply in operational cost. NC-CC requires only a millisecond-scale preprocessing pass and no supervised training, making it easy to embed on modest hardware. XGBoost, while still fast at inference, demands explicit training, feature engineering and hyper-parameter tuning—a heavier burden in situations where rapid deployment or on-device computation is essential.

Naturally, the proposed approach is not without limitations. Its reliance on a single nearest neighbour can exaggerate isolated rainfall peaks, and—like XGBoost—it treats wind direction as a linear variable, which can distort circular statistics. We also restricted the analysis to daily aggregates; sub-daily resolutions or other climatic zones may reveal different behaviours.

Future work will therefore focus on adaptive weights between norm and cosine components, multi-neighbour (vectorial k-NN) extensions and hybrid ensembles that blend NC-CC's local sensitivity with the global generalisation of tree-boosting.

In sum, a carefully designed non-parametric analogue method such as NC-CC can rival—and sometimes surpass—state-of-the-art supervised models, reminding us that simplicity, when aligned with domain structure, remains a powerful asset in climate time-series forecasting.

Acknowledgments

The authors gratefully acknowledge the financial support of the Production Engineering and Management Institute at the Federal University (PRPPG–UNIFEI), and the institutional support of the Federal Institute of Southern Minas Gerais (IFSULDEMINAS), including the leave granted for full-time doctoral dedication. This work was also supported by the Coordination for the Improvement of Higher Education Personnel (CAPES), the National Council for Scientific and Technological Development (CNPq), and the Research Support Foundation of the State of Minas Gerais (FAPEMIG).

References

[1] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. New York, NY, USA: Association for Computing Machinery, 2016, pp. 785–794. [Online]. Available: https://doi.org/10.1145/2939672.2939785

[2] N. Shabbir, R. Ahmadiahangar, A. Rosin, M. Jawad, J. Kilter, and J. Martins, "XgBoost based short-term electrical load forecasting considering trends & periodicity in historical data," in *2023 IEEE International Conference on Energy Technologies for Future Grids (ETFG)*. Wollongong, NSW, Australia: IEEE, 2023, pp. 1–6. [Online]. Available: https://doi.org/10.1109/ETFG55873.2023.10407926

[3] A. Sutou and J. Wang, "Influence-balanced xgboost: Improving xgboost for imbalanced data using influence functions," *IEEE Access*, vol. 12, pp. 193 473–193 486, 2024. [Online]. Available: https://doi.org/10.1109/ACCESS.2024.3520159

[4] J. Gou, J. Yi, S. Liu, W. Wang, Y. Bao, and F. Dong, "A survey on the k-Nearest Neighbor algorithm," *Computers & Electrical Engineering*, vol. 90, p. 107005, 2021. [Online]. Available: https://doi.org/10.1016/j.compeleceng.2020.107005

[5] J. Neyra, V. B. Siramshetty, and H. I. Ashqar, "The effect of different feature selection methods on models created with XGBoost," 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2411.05937

[6] D. S. Wilks, *Statistical Methods in the Atmospheric Sciences*, 3rd ed., ser. International Geophysics Series, Vol. 100. Academic Press, 2011.

[7] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*, 5th ed. John Wiley & Sons, 2015.

[8] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, 3rd ed. OTexts, 2021. [Online]. Available: https://otexts.com/fpp3/

[9] P. H. L. Junior and L. Cabral, "Exploring natural language processing for fakenews detection and classification: A comparative analysis of naive bayes, svm and xgboost," in *Proceedings of the XVI Brazilian Conference on Computational Intelligence (CBIC '23)*. Salvador, Brazil: Sociedade Brasileira de Inteligência Computacional, 2023. [Online]. Available: https://doi.org/10.21528/CBIC2023-117

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[11] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y. Wang, and X. Yan, "Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting," 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1907.00235

[12] A. Chattopadhyay, E. Nabizadeh, and P. Hassanzadeh, "Analog forecasting of extreme-causing weather patterns using deep learning," *Journal of Advances in Modeling Earth Systems*, vol. 12, no. 2, p. e2019MS001958, 2020. [Online]. Available: https://doi.org/10.1029/2019MS001958