Uncertainty-Aware Prediction of Climate Extremes Using Fine-Tuned Time-Series Foundation Models

Imran Nasim
IBM UK; University of Surrey
imran.nasim@ibm.com
i.nasim@surrey.ac.uk

João Lucas de Sousa Almeida
IBM Research Brazil
joao.lucas.sousa.almeida@ibm.com

Abstract

AI-driven weather forecasting models, particularly foundation models, have achieved significant advancements in both speed and accuracy. However, accurately forecasting rare, high-impact extreme events, such as storms and heatwaves, remains a critical challenge. These models often underestimate event intensity and frequency, limiting their reliability in operational and risk-sensitive contexts. In this study, we investigate uncertainty-aware extreme event forecasting using the recently introduced time-series foundation model, Tiny Time Mixers (TTM). We develop and compare two uncertainty quantification approaches, hyperparameter ensembling and Monte Carlo (MC) dropout, and evaluate their ability to improve classification of extreme events. Our results show that incorporating predictive uncertainty significantly enhances performance compared to zero-shot TTM, and that the choice of uncertainty method and threshold critically affects model behavior. We find that the hyperparameter ensemble yields more stable and accurate predictions, particularly for rare storm events, highlighting the value of lightweight ensemble models for uncertainty-calibrated forecasting.

1 Introduction

AI-driven, data-centric weather forecasting models have garnered substantial attention [23, 12], particularly with the emergence of foundation models for climate science [3, 25, 4]. These models have shown remarkable performance, offering improvements in both predictive accuracy and inference speed over usual numerical methods. In addition, they offer greater flexibility than classical reduced-order techniques, which typically rely on hand-engineered dimensionality reduction and may struggle to capture the nonlinearity of extreme events [27, 18, 20, 19, 21]. This has enabled rapid ensemble simulations and probabilistic forecasts, making them invaluable for large-scale and real-time prediction tasks [11, 5]. Despite these advances, a limitation remains: the ability of AI-based climate emulators to accurately capture rare but high-impact extreme events, such as storms, droughts, and heatwaves [24]. These events are inherently rare, presenting data imbalance challenges [10, 26], and are often underestimated by neural forecasting models [22]. This underestimation may stem from biases introduced by data scarcity or spectral smoothing effects [6]. An often overlooked factor is the single-deterministic nature of inference in most foundation models. Despite stochasticity during training, such models produce fixed outputs for a given input, hyperparameter setting, and initialization [28, 13, 29]. To address the challenge, we propose an ensemble-based uncertainty quantification framework using the recently introduced Tiny Time Mixer (TTM) ([8]), comparing hyperparameter ensembling and Monte Carlo (MC) dropout to evaluate predictive uncertainty of extreme events [15]. Our results show that modeling uncertainty improves detection and that tuning its level is crucial for optimal performance. Hyperparameter ensembling outperforms MC Dropout, offering more stable predictions, especially for rare events like storms, providing a stronger basis for uncertainty-aware decision-making.

Tackling Climate Change with Machine Learning workshop at NeurIPS 2025

2 Methods

Neural Architecture: TTM builds upon the TSMixer architecture [7], which was originally developed for multivariate time-series forecasting and has recently been applied to chemical kinetics [14], as well as to other areas of scientific machine learning [16, 17]. It features a lightweight MLP-based design, making it significantly smaller and more efficient than transformer-based models. The version used in this study has approximately 1 million parameters, including both backbone and decoder. Preprocessing involves normalizing the time series and applying a patching procedure that splits the data into smaller chunks, preserving local structure while reducing computational cost [8]. The encoder processes these patches using MLP blocks and gated attention, combining linear layers, nonlinearities, and dropout to generate an intermediate embedding. The decoder reconstructs this into the original space, and a final linear "head" maps it to the target forecast horizon. This streamlined setup offers strong predictive power with low computational overhead. A schematic of the TTM architecture is shown in Figure 1.

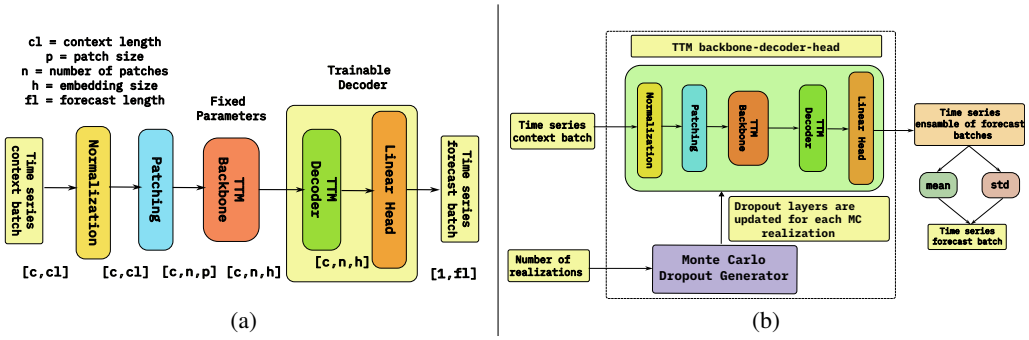

Figure 1: a) Overview of the TinyTimeMixer architecture. b) A scheme demonstrating how the Monte Carlo dropout is orchestrated.

Dataset and processing: We obtained weather data from 567 stations across Brazil via the Brazilian National Institute of Meteorology (INMET) web interface [2], covering the period from 2000 to December 2023. Due to varying installation dates and data gaps, not all stations span the full period. After quality checks, we selected data from 2019–2023, which offered the best national coverage with minimal missing values. Stations with more than 1% missing data were excluded, resulting in 18 stations. This threshold ensured that missing periods were short enough to minimize distortion from imputation. Remaining gaps were filled using nearest-neighbor interpolation. For training, validation, and testing, we applied a temporal split of 90%, 5%, and 5%, respectively. This setup provided sufficient training data while allowing robust evaluation. We also benchmarked our fine-tuned models against the base TTM using its zero-shot predictions, referred to throughout as the "zero-shot model".
Extreme event classification: Extreme events were identified using a sliding window method applied to temperature and pressure data, with thresholds derived from seasonal percentiles. A short heatwave was defined as a 6-hour period where all temperature values exceeded the 95th percentile for the respective month, while a storm was defined as a 6-hour period where all pressure values fell below the 5th percentile. This approach captures consecutive extreme conditions, which are more impactful than isolated anomalies, and accounts for seasonal variability across diverse locations. For further details, see Appendix D.
Model Ensemble: In order to produce the model ensemble, we fine-tune multiple TTM versions, each one for a different choice of hyperparameters, using the limits defined in Table 3. In the evaluation stage, the test dataset is passed through the models ensemble generating a set of possible solutions. In the end, we evaluated the mean and standard deviation curves for this set and used them to create a confidence interval, as seen in Figure 2. The confidence interval in turn is used to evaluate the indices of interest, occurrence of short heatwave and storms, according the following criterion: $T_{ens} = T_{mean} + \alpha_T \, T_{std}$ and $P_{ens} = P_{mean} + \alpha_P \, P_{std}$. This approach is based on the assumption that increasing temperatures can compensate for model-induced losses, improving short heatwave detection. Similarly, reducing pressure values may help mitigate false negatives in storm event predictions. The hyperparameters α_T and α_P were tested using a hypersearch loop, however our initial hyperparameter search did not reveal a significant improvement, leading us to adopt a

limit of $|\alpha_T| = |\alpha_P| = 2$ as a reasonable choice for our experiments. For further details on the experimental setup and computational cost, see Appendix C.

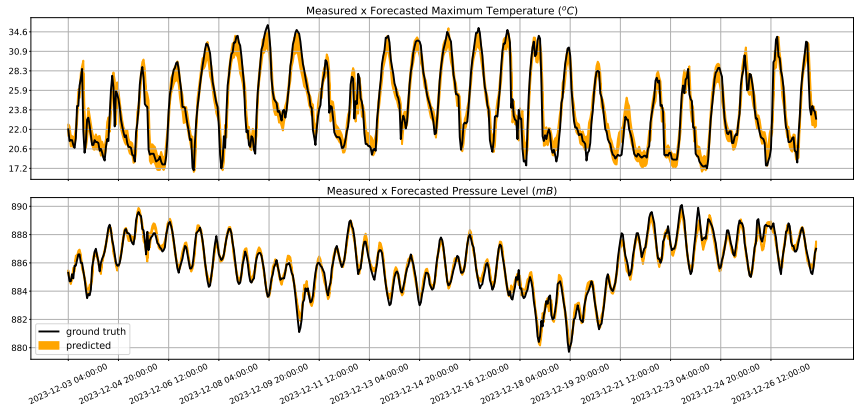

Figure 2: Maximum temperature and pressure level forecasting for Brasilia (Brazilian capital). The confidence interval was constructed using 2σ for each variable using the hyperparameter ensemble method. The curves were created by composing the forecast length of our model over 4 days.

MC Dropout: To compare the uncertainty quantification in the time series forecasts of our ensemble approach, we consider a *Monte Carlo Dropout* (MC Dropout) approach as a baseline. For details on MC Dropout and implementation, see Apdx C.1. We apply this technique to the Tiny Time Mixer (TTM) architecture for our best performing fine-tuned model to capture epistemic uncertainty in our multivariate time series predictions. A schematic demonstrating how MC dropout is applied in the TTM architecture is presented in Figure 1.

3 Results

We compare the predictive behavior of the hyperparameter ensemble method and the MC Dropout method, for comparison with the zero-shot performance see Apdx A. As shown in Tables 1 and 2, MC Dropout exhibits greater variability in predicted extreme event counts across uncertainty thresholds. This is further quantified using the average *coefficient of variation* metric defined in Appendix E, with per-location comparisons reported in Table 8. While MC Dropout performs slightly better at detecting short heatwaves under conservative uncertainty ($\sigma_T = 1$), it consistently overestimates event counts under higher uncertainty settings ($\sigma_T = 2, \sigma_P = -1$), particularly for storm events. In contrast, the hyperparameter ensemble method yields more stable predictions across thresholds and aligns more closely with observed event frequencies. This suggests that the ensemble approach is more suitable for operational settings requiring tighter control over false positives and uncertainty calibration. To further investigate this, we compare confusion matrices for each method at their respective optimal uncertainty thresholds, selected based on aggregated event count accuracy. Specifically, we use $\alpha_T = 2$ and $\alpha_P = -1$ for the ensemble method, and $\alpha_T = 1$ and $\alpha_P = 1$ for MC Dropout. The normalized confusion matrices, shown in Figure 3, reveal that both methods perform comparably on short heatwaves—hyperparameter ensemble slightly favors recall (0.63 vs. 0.60), while MC Dropout has slightly higher precision (FP rate 0.06 vs. 0.07). However, for storms, the hyperparameter ensemble achieves a substantially higher true positive rate (0.94 vs. 0.63) with minimal false positives, whereas MC Dropout under-detects many true events. These findings reinforce that the hyperparameter ensemble method offers better accuracy and stability, particularly in high-uncertainty or rare-event regimes, making it the more appropriate choice when reliable uncertainty quantification is critical.

4 Conclusion

We have fine-tuned the TTM using weather station data to forecast short-term events and evaluate extreme event prediction, such as heatwaves and storms. We propose extending base fine-tuning through hyperparameter ensemble and MC-dropout methods. Both approaches significantly outperform zero-

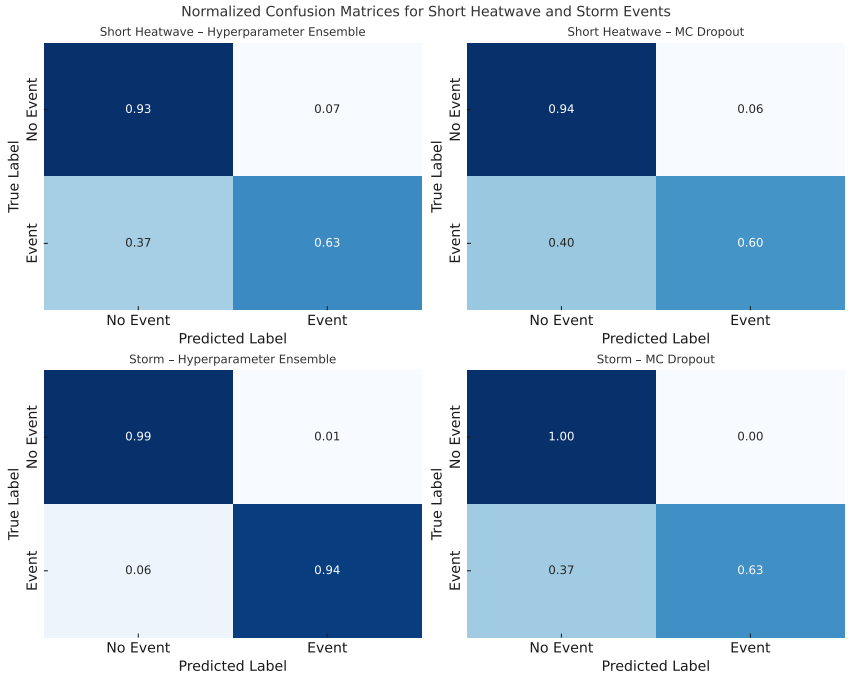

Figure 3: Normalized confusion matrices for short heatwave and storm event classification using the best-performing uncertainty thresholds for each method. Each matrix shows the true vs. predicted labels for the hyperparameter ensemble (left panels) and MC Dropout (right panels) methods, evaluated at the thresholds that most closely matched the observed event counts ($\alpha_T = 2, \alpha_P = -1$ for the ensemble method; $\alpha_T = 1, \alpha_P = 1$ for MC Dropout).

shot models in extreme events; however, hyperparameter ensemble offers improved accuracy and stability compared to MC-dropout, particularly in high-uncertainty or rare-event scenarios, making it preferable for reliable uncertainty quantification. While the model showcases reliable forecasts respecting periodicity and trends, it still struggles to identify many extreme events based on the metrics considered in this work. A significant finding reveals that small inaccuracies in temperature and pressure predictions significantly impact the classification of extreme events, emphasizing their sensitivity. Analysis suggests that generic fine-tuning approaches may not be effective for managing class imbalance inherent to extreme event prediction. Developing a specific formalism focusing on rare event detection using techniques such as class-balanced loss functions and ensemble methods could enhance the model's ability to capture these events. Despite significant differences in weather patterns and event distributions, the model shows adaptability to varied conditions, indicating that the TTM has the potential to perform well in diverse settings. This is further emphasized by the diversity of extreme events across locations in our dataset which demonstrates the generalizability of fine-tuning. Finally, while the architecture provides a strong foundation, further exploration of decoder and head architectures, as well as strategies to mitigate amplitude discrepancies, could enhance the model's performance. These findings suggest that small, efficient models like TTM have considerable potential to forecast extreme events and serve as a promising avenue for future research.

References

[1] Granite time series (ttm-r2). `https://huggingface.co/ibm-granite/granite-timeseries-ttm-r2`. Accessed: 2025-01-02.

[2] Instituto nacional de meteorologia, inmet. `https://portal.inmet.gov.br/`. Acessed: 2025-01-03.

[3] Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian. Accurate medium-range global weather forecasting with 3d neural networks. *Nature*, 619(7970):533–538, 2023.

[4] Cristian Bodnar, Wessel P Bruinsma, Ana Lucic, Megan Stanley, Johannes Brandstetter, Patrick Garvan, Maik Riechert, Jonathan Weyn, Haiyu Dong, Anna Vaughan, et al. Aurora: A foundation model of the atmosphere. *arXiv preprint arXiv:2405.13063*, 2024.

[5] Salva Rühling Cachay, Brian Henn, Oliver Watt-Meyer, Christopher S Bretherton, and Rose Yu. Probabilistic emulation of a global climate model with spherical dyffusion. *arXiv preprint arXiv:2406.14798*, 2024.

[6] Ashesh Chattopadhyay and Pedram Hassanzadeh. Long-term instabilities of deep learning-based digital twins of the climate system: The cause and a solution. *arXiv preprint arXiv:2304.07029*, 2023.

[7] Si-An Chen, Chun-Liang Li, Nate Yoder, Sercan O. Arik, and Tomas Pfister. TSMixer: An All-MLP architecture for time series forecasting, 2023.

[8] Vijay Ekambaram, Arindam Jati, Nam H Nguyen, Pankaj Dayama, Chandra Reddy, Wesley M Gifford, and Jayant Kalagnanam. TTMs: Fast multi-level tiny time mixers for improved zero-shot and few-shot forecasting of multivariate time series. *arXiv preprint arXiv:2401.03955*, 2024.

[9] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[10] Bartosz Krawczyk. Learning from imbalanced data: open challenges and future directions. *Progress in artificial intelligence*, 5(4):221–232, 2016.

[11] Thorsten Kurth, Shashank Subramanian, Peter Harrington, Jaideep Pathak, Morteza Mardani, David Hall, Andrea Miele, Karthik Kashinath, and Anima Anandkumar. Fourcastnet: Accelerating global high-resolution weather forecasting using adaptive fourier neural operators. In *Proceedings of the platform for advanced scientific computing conference*, pages 1–11, 2023.

[12] Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, et al. Learning skillful medium-range global weather forecasting. *Science*, 382(6677):1416–1421, 2023.

[13] Lizhi Liao, Heng Li, Weiyi Shang, and Lei Ma. An empirical study of the impact of hyperparameter tuning and model optimization on the performance properties of deep neural networks. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 31(3):1–40, 2022.

[14] Imran Nasim and Joaõ Lucas de Sousa Almeida. Towards foundation models for the industrial forecasting of chemical kinetics. *arXiv preprint arXiv:2408.10720*, 2024.

[15] Imran Nasim and JOÃO LUCAS DE SOUSA ALMEIDA. Fine-tuning for extreme event prediction: Are ensemble methods all you need? In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2025.

[16] Imran Nasim and João Lucas de Sousa Almeida. Fine-tuned mlp-mixer foundation models as data-driven numerical surrogates? In *NeurIPS 2024 Workshop on Data-driven and Differentiable Simulations, Surrogates, and Solvers*, 2024.

[17] Imran Nasim and Joao Lucas de Sousa Almeida. Fine-tuned mlp-mixers as data-driven numerical surrogates? In *Annual Conference on Neural Information Processing Systems*, 2024.

[18] Imran Nasim and Joao Lucas de Sousa Almeida. Using neural implicit flow to represent latent dynamics of canonical systems. In *International Conference on Scientific Computing and Machine Learning*, 2024.

[19] Imran Nasim and Michael E Henderson. Dynamically meaningful latent representations of dynamical systems. *Mathematics*, 12(3):476, 2024.

[20] Imran Nasim and Melanie Weber. Learning reduced order dynamics via geometric representations. In *International Conference on Scientific Computing and Machine Learning*, 2024.

[21] Imran Nasim and Melanie Weber. Automated manifold learning for reduced order modeling. *arXiv preprint arXiv:2506.01741*, 2025.

[22] Olivier C Pasche, Jonathan Wider, Zhongwei Zhang, Jakob Zscheischler, and Sebastian Engelke. Validating deep-learning weather forecast models on recent high-impact extreme events. *Artificial Intelligence for the Earth Systems*, 2024.

[23] Jaideep Pathak, Shashank Subramanian, Peter Harrington, Sanjeev Raja, Ashesh Chattopadhyay, Morteza Mardani, Thorsten Kurth, David Hall, Zongyi Li, Kamyar Azizzadenesheli, et al. Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators. *arXiv preprint arXiv:2202.11214*, 2022.

[24] Y Qiang Sun, Pedram Hassanzadeh, Mohsen Zand, Ashesh Chattopadhyay, Jonathan Weare, and Dorian S Abbot. Can ai weather models predict out-of-distribution gray swan tropical cyclones? *arXiv preprint arXiv:2410.14932*, 2024.

[25] Daniela Szwarcman, Sujit Roy, Paolo Fraccaro, Þorsteinn Elí Gíslason, Benedikt Blumenstiel, Rinki Ghosal, Pedro Henrique de Oliveira, Joao Lucas de Sousa Almeida, Rocco Sedona, Yanghui Kang, et al. Prithvi-eo-2.0: A versatile multi-temporal foundation model for earth observation applications. *arXiv preprint arXiv:2412.02732*, 2024.

[26] John E Walsh, Thomas J Ballinger, Eugénie S Euskirchen, Edward Hanna, Johanna Mård, James E Overland, Helge Tangen, and Timo Vihma. Extreme weather and climate events in northern areas: A review. *Earth-Science Reviews*, 209:103324, 2020.

[27] Rui Wang, Karthik Kashinath, Mustafa Mustafa, Adrian Albert, and Rose Yu. Towards physics-informed deep learning for turbulent flow prediction. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1457–1466, 2020.

[28] Florian Wenzel, Jasper Snoek, Dustin Tran, and Rodolphe Jenatton. Hyperparameter ensembles for robustness and uncertainty quantification. *Advances in Neural Information Processing Systems*, 33:6514–6527, 2020.

[29] Donglin Zhuang, Xingyao Zhang, Shuaiwen Song, and Sara Hooker. Randomness in neural network training: Characterizing the impact of tooling. *Proceedings of Machine Learning and Systems*, 4:316–336, 2022.

A Zero-shot Performance

Fine-tuned Hyper-parameter Ensemble vs Zero-shot Performance: To evaluate the performance of the TTM in predicting extreme weather events, we first compare the results of two different approaches: a zero-shot model and our ensemble-based fine-tuned model approach. Table 9 presents the zero-shot predictions, while Table 1 shows the fine-tuned ensemble predictions incorporating uncertainty estimates. In the zero-shot scenario, where the model is applied without prior fine-tuning, the predicted extreme event counts show considerable discrepancies when compared to observed values. The aggregated results indicate an underestimation of short heatwave (SH) occurrences and storm events. While some locations, such as Valença and Montes Claros, exhibit reasonable alignment with observed counts, others, like Areia and Imperatriz, show significant underprediction, highlighting the limitations of the zero-shot approach in capturing extreme events. In contrast, the fine-tuned ensemble approach significantly improves predictive performance by incorporating uncertainty. As shown in Table 1, the predicted counts with a confidence interval of 2σ align more closely with the observed data. This improvement demonstrates that the ensemble-based fine-tuning effectively mitigates underestimation, particularly in locations such as Maria da Fé, Brasilia, and São Mateus, where zero-shot predictions were notably inaccurate. Moreover, the fine-tuned model provides a better characterization of uncertainty, as observed in the range of values predicted across different confidence levels ($\pm 1\sigma$ and $\pm 2\sigma$). This allows for a more robust assessment of extreme events, improving the model's reliability in critical scenarios. However, some locations, such as Pradópolis, still exhibit discrepancies, indicating potential areas for further model improvement. Importantly, our findings suggest that tuning the level of uncertainty is crucial for achieving optimal extreme event classification. The observed improvements across multiple locations highlight the

importance of selecting appropriate uncertainty scaling factors, which directly influence the balance between under- and over-prediction. Our results emphasize that fine-tuning uncertainty parameters can enhance the model's sensitivity to extreme events, providing more reliable and actionable insights for decision-making. In addition to the event counts, we evaluated both the zero-shot and ensemble methods using precision, recall, and F1 score metrics which are presented in Appendix F. Overall, our findings demonstrate that fine-tuning with an ensemble approach not only improves the accuracy of extreme event prediction but also provides a more reliable framework for uncertainty quantification of such events.

B Extreme Event Counts Across Locations

In the tables below we provide the event counts across locations for all of the methods considered in this study.

Table 1: Extreme event counts across locations (real vs predicted with uncertainty) for the Hyperparameter Ensemble method.

Location	**Real SH**	**Real Storm**	**Pred. SH** ($\alpha_T = 1$)	**Pred. SH** ($\alpha_T = 2$)	**Pred. Storm** ($\alpha_P = 1$)	**Pred. Storm** ($\alpha_P = -1$)
Salvador	61	66	49	67	49	68
Maria da Fé	133	38	129	165	58	133
Carmo	165	99	188	233	122	167
Valença	196	136	54	80	65	97
Timóteo	86	80	119	159	46	79
Duque de Caxias	131	77	152	194	24	30
Pradópolis	148	52	0	0	12	22
Passa Quatro	128	52	135	191	29	39
Montes Claros	198	60	46	79	6	32
São Mateus	123	90	57	85	73	124
Areia	0	22	129	150	78	124
Imperatriz	70	44	117	148	37	52
Florianopolis	82	90	137	161	19	30
Conceicao das alagoas	126	67	88	124	61	123
Brasilia	157	35	56	80	51	67
Campos dos goytacazes	136	68	96	147	55	89
Itaberaba	77	75	126	145	48	74
Gama ponte alta	187	30	182	245	44	79
Aggregated	2204	1181	1860	2453	877	1429

Table 2: Extreme Event Counts Across Locations (Real vs Predicted with Uncertainty) for the MC dropout approach.

Location	**Real SH**	**Real Storm**	**Pred. SH** ($\alpha_T = 1$)	**Pred. SH** ($\alpha_T = 2$)	**Pred. Storm** ($\alpha_P = 1$)	**Pred. Storm** ($\alpha_P = -1$)
Salvador	61	66	55	139	47	96
Maria da Fé	133	38	157	202	28	64
Carmo	165	99	141	197	50	159
Valença	196	136	190	258	105	206
Timóteo	86	80	69	109	50	111
Duque de Caxias	131	77	59	108	16	44
Pradópolis	148	52	107	169	26	56
Passa Quatro	128	52	166	242	52	71
Montes Claros	198	60	73	132	28	67
São Mateus	123	90	75	131	97	118
Areia	0	22	132	178	97	121
Imperatriz	70	44	113	179	36	87
Florianopolis	82	90	128	181	40	61
Conceicao das alagoas	126	67	77	132	103	93
Brasilia	157	35	59	95	44	55
Campos dos goytacazes	136	68	76	129	64	71
Itaberaba	77	75	102	148	53	62
Gama ponte alta	187	30	207	291	64	77
Aggregated	2204	1181	2065	2956	795	1938

C Model training and hyperparameter search details

We performed experiments using the backbones versions available from HuggingFace [1] within a hypersearch pipeline in which just a few parameters of interest were left to be optimized. For this work the hyperparameters we have chosen were fewshot fraction and context length. As the backbones for TinyTimeMixer are defined with different choices for context and forecast lengths

it means that different backbones can be downloaded for each hypersearch realization. We also experimented with freezing the backbone during the finetuning stage, but we have seen that this option has almost no discernable effect. The list of hyperparameter definition and values for the hyperparamater optimization framework is given in Table 3. We executed a total number of 100 trials based on the limits shown in Table 3 where the best model was selected based on the minimization of the mean squared error. The best choices for the hyperparameters are shown in the last column. All the experiments were performed using a single NVIDIA v100 GPU and took a total time of approximately $6h$ to be finished. Using the same hardware, the model took $283\,s$ to perform the inference for all the test subdatasets. Inference using Monte Carlo (MC) dropout, on the other hand, has taken approximately $635\,s$ to produce results for 50 realizations.

Table 3: Hyperparameter definition and values used in our hyperparameter optimization framework.

Hyperparameter	**Description**	**Values**	**Best choice**
fewshot fraction	Fraction of data used for few-shot fine-tuning	5% - 15%	9%
head dropout rate	Dropout rate applied to the model's head	0.4	-
dropout rate	General dropout rate applied throughout the model	0.4	-
learning rate	Learning rate for the AdamW optimizer	0.001	-
Learning rate scheduler	Method used to uptate the learning rate	OneCycleLR	-
batch size	Batch size for training and evaluation	64	-
number of epochs	Maximum number of epochs for training	50	-
freeze backbone	Whether to freeze the backbone during fine-tuning	`True`	-
context length	Length of input sequence	512, 1024	1024
forecast length	Length of forecasted output sequence	96	-
number of trials	Number of trials for the hypersearch	100	-

C.1 MC Dropout Implementation for TTM

MC Dropout approximates Bayesian inference using standard dropout during both training and inference [9]. Rather than disabling dropout at test time, MC Dropout performs multiple stochastic forward passes through the model with dropout active, treating each pass as a sample from a variational posterior. Given an input $\mathbf{x}$, we compute T stochastic outputs $\{\hat{y}^{(t)}\}_{t=1}^{T}$ using different dropout masks, and estimate the predictive mean and variance as:

$$\mathbb{E}[y] \approx \frac{1}{T}\sum_{t=1}^{T}\hat{y}^{(t)}, \quad \text{Var}(y) \approx \frac{1}{T}\sum_{t=1}^{T}\left(\hat{y}^{(t)} - \mathbb{E}[y]\right)^2.$$

To implement MC Dropout with the TTM architecture, we make lightweight but effective modifications to the standard Hugging Face `Trainer` class. Our goal is to enable stochasticity during inference in order to approximate Bayesian model uncertainty.

Dropout Activation During Evaluation: Normally, dropout layers are deactivated during evaluation. To perform MC Dropout, we modify the training loop to keep the model in training mode (`train()`), ensuring dropout remains active during inference. This allows each forward pass to use a different dropout mask.

Injecting Stochasticity: We explicitly reinitialize the random number generator before each forward pass using a new randomly generated seed. This guarantees that different dropout masks are sampled across passes, while keeping all other components of the inference pipeline deterministic.

Ensuring Stability: To avoid introducing unintended variability from non-dropout components, such as batch normalization, we retain these layers in evaluation mode throughout inference. This ensures that the only source of randomness arises from dropout, allowing for a clean estimation of epistemic uncertainty.

To evaluate this approach, we perform T stochastic forward passes over the same evaluation dataset, recording the predicted outputs or losses for each pass. The predictive mean and variance are then computed across these T samples. This provides an estimate of both the expected model output and the associated uncertainty for each prediction.

D Extreme Event Classification Details

We identify extreme events using a sliding window approach based on seasonal thresholds. For each calendar month, we compute the 95th percentile of temperature and the 5th percentile of pressure values. Short heatwaves are defined as periods of six consecutive hours where all temperature values exceed the monthly 95th percentile, and storms are defined similarly for pressure falling below the 5th percentile.

A sliding window of size 6 is applied across the time series, labeling a segment as an extreme event if all values within the window satisfy the respective condition. To account for overlapping detections due to the sliding window, reported event counts are divided by the window length (6 hours) to estimate the number of unique events.

This percentile-based method adapts to seasonal variability and generalizes across locations with differing climate regimes. It emphasizes the detection of temporally clustered extreme conditions, which are of greater practical importance than isolated anomalies. Real and predicted extreme event counts are reported in Tables 1 and 9.

E Coefficient of Variation-Based Uncertainty Metric

To quantitatively compare the uncertainty estimates produced by the Hyperparameter Ensemble and MC Dropout methods, we use the *coefficient of variation* (CV) as a relative uncertainty metric. For each timestamp and location, we compute the temperature CV as $\mathrm{CV}_T = \sigma_T/\mu_T$ and the pressure CV as $\mathrm{CV}_P = \sigma_P/\mu_P$, where σ and μ denote the ensemble standard deviation and mean, respectively. We report the average CV per location by taking the mean of all timestamp-level CVs:

$$\overline{\mathrm{CV}}_T = \frac{1}{N}\sum_{i=1}^{N}\frac{\sigma_T^{(i)}}{\mu_T^{(i)}}, \quad \overline{\mathrm{CV}}_P = \frac{1}{N}\sum_{i=1}^{N}\frac{\sigma_P^{(i)}}{\mu_P^{(i)}}.$$

An overall uncertainty score per location is then defined as

$$\overline{\mathrm{CV}} = \frac{\overline{\mathrm{CV}}_T + \overline{\mathrm{CV}}_P}{2}.$$

For readability, temperature CVs are scaled by a factor of 10^2 and pressure CVs by 10^4. The resulting average CV values are reported in Table 8.

F Model Classification Performance

F.1 Fine-tuned performance

Here we present the extreme event classification metric values for the ensemble methods described in the main text.

F.2 Zero-shot performance

For completeness, we also include the results of the zero-shot predictions for extreme events which is given below:

Table 4: Performance metrics comparison across locations for the Hyperparameter ensemble method with $+1\sigma$ and $+1\sigma$ uncertainty for temperature and pressure respectively.

Location	SH Precision	SH Recall	SH F1	Storm Precision	Storm Recall	Storm F1
Salvador	0.347	0.279	0.309	0.971	1.000	0.985
Carmo	0.961	0.752	0.844	0.737	0.990	0.845
Valença	0.957	0.918	0.938	0.802	0.985	0.884
Timóteo	0.889	0.558	0.686	0.794	0.962	0.870
Pradópolis	0.824	0.662	0.734	0.658	1.000	0.794
Gama	0.967	0.786	0.867	0.933	0.933	0.933
Areia	0.000	0.000	0.000	1.000	1.000	1.000
Maria da Fé	0.830	0.842	0.836	0.949	0.974	0.961
Imperatriz	0.978	0.643	0.776	0.812	0.591	0.684
Florianopolis	0.930	0.646	0.763	0.702	0.967	0.813
Duque de Caxias	0.853	0.840	0.846	0.589	0.948	0.726
Passa Quatro	0.863	0.789	0.824	0.827	0.827	0.827
Brasilia	0.942	0.822	0.878	0.933	0.800	0.862
Campos dos Goytacazes	0.920	0.596	0.723	0.545	0.985	0.702
Itaberaba	0.786	0.571	0.662	1.000	0.893	0.944
São Mateus	0.781	0.610	0.685	0.910	0.900	0.905
Conceição das Alagoas	0.841	0.841	0.841	0.811	0.896	0.851
Montes Claros	0.879	0.808	0.842	0.747	0.983	0.849
Aggregated	0.877	0.740	0.799	0.798	0.936	0.852

Table 5: Performance metrics comparison across locations for the Hyperparameter ensemble method with $+2\sigma$ and -1σ uncertainty for temperature and pressure respectively.

Location	SH Precision	SH Recall	SH F1	Storm Precision	Storm Recall	Storm F1
Salvador	0.343	0.377	0.359	1.000	0.742	0.852
Carmo	0.897	0.897	0.897	0.862	0.505	0.637
Valença	0.820	0.974	0.890	0.934	0.838	0.884
Timóteo	0.762	0.709	0.735	0.892	0.725	0.800
Pradópolis	0.767	0.824	0.795	0.848	0.750	0.796
Gama	0.861	0.893	0.877	1.000	0.800	0.889
Areia	0.000	0.000	0.000	1.000	0.545	0.706
Maria da Fé	0.660	0.947	0.778	1.000	0.763	0.866
Imperatriz	0.734	0.829	0.779	1.000	0.136	0.240
Florianopolis	0.847	0.878	0.862	0.973	0.789	0.871
Duque de Caxias	0.760	0.870	0.811	0.782	0.792	0.787
Passa Quatro	0.723	0.836	0.775	1.000	0.712	0.831
Brasilia	0.901	0.924	0.912	0.947	0.514	0.667
Campos dos Goytacazes	0.887	0.809	0.846	0.967	0.868	0.915
Itaberaba	0.700	0.727	0.713	1.000	0.680	0.810
São Mateus	0.646	0.772	0.704	1.000	0.611	0.759
Conceição das Alagoas	0.834	0.960	0.893	1.000	0.716	0.835
Montes Claros	0.776	0.960	0.858	1.000	0.733	0.846
Aggregated	0.782	0.865	0.819	0.947	0.699	0.791

Table 6: Performance metrics comparison across locations for the MC dropout method with $+1\sigma$ and $+1\sigma$ uncertainty for temperature and pressure respectively.

Location	SH Precision	SH Recall	SH F1	Storm Precision	Storm Recall	Storm F1
Salvador	0.309	0.279	0.293	1.000	0.712	0.832
Carmo	0.922	0.788	0.850	0.880	0.444	0.591
Valença	0.921	0.893	0.907	0.933	0.721	0.813
Timóteo	0.884	0.709	0.787	0.860	0.537	0.662
Pradópolis	0.765	0.615	0.682	0.943	0.635	0.759
Gama	0.934	0.821	0.874	0.922	0.711	0.802
Areia	0.000	0.000	0.000	1.000	0.947	0.973
Maria da Fé	0.800	0.820	0.810	0.951	0.921	0.936
Imperatriz	0.968	0.614	0.751	0.818	0.593	0.686
Florianopolis	0.888	0.656	0.756	0.769	0.613	0.683
Duque de Caxias	0.880	0.821	0.849	0.652	0.852	0.738
Passa Quatro	0.871	0.809	0.839	0.851	0.760	0.803
Brasilia	0.955	0.833	0.890	0.944	0.800	0.867
Campos dos Goytacazes	0.883	0.627	0.734	0.632	0.781	0.699
Itaberaba	0.774	0.558	0.647	0.975	0.862	0.915
São Mateus	0.782	0.642	0.705	0.897	0.779	0.834
Conceição das Alagoas	0.793	0.793	0.793	0.821	0.744	0.780
Montes Claros	0.874	0.803	0.837	0.745	0.983	0.847
Aggregated	0.865	0.751	0.804	0.826	0.776	0.800

Table 7: Performance metrics comparison across locations for the MC dropout method with +2σ and -1σ uncertainty for temperature and pressure respectively.

Location	**SH Precision**	**SH Recall**	**SH F1**	**Storm Precision**	**Storm Recall**	**Storm F1**
Salvador	0.353	0.803	0.490	0.688	1.000	0.815
Carmo	0.792	0.945	0.862	0.623	1.000	0.767
Valença	0.729	0.959	0.828	0.650	0.985	0.784
Timóteo	0.670	0.849	0.749	0.712	0.988	0.827
Pradópolis	0.680	0.791	0.731	0.486	0.981	0.650
Gama	0.870	0.950	0.909	0.815	0.941	0.874
Areia	0.000	0.000	0.000	0.919	0.973	0.945
Maria da Fé	0.783	0.872	0.825	0.884	0.974	0.927
Imperatriz	0.916	0.771	0.837	0.750	0.818	0.783
Florianopolis	0.833	0.707	0.765	0.682	0.720	0.700
Duque de Caxias	0.789	0.809	0.799	0.585	0.852	0.694
Passa Quatro	0.812	0.844	0.828	0.674	0.865	0.757
Brasilia	0.918	0.846	0.880	0.902	0.800	0.848
Campos dos Goytacazes	0.765	0.713	0.738	0.519	0.698	0.595
Itaberaba	0.682	0.636	0.658	0.853	0.828	0.840
São Mateus	0.668	0.715	0.690	0.741	0.800	0.769
Conceição das Alagoas	0.740	0.746	0.743	0.775	0.697	0.734
Montes Claros	0.854	0.803	0.828	0.740	0.983	0.844
Aggregated	0.774	0.768	0.764	0.698	0.811	0.741

Table 8: Comparison of Average Coefficient of Variation (CV) Between Hyperparameter Ensemble and MC Dropout Methods Across Locations. Temperature CVs are scaled by 10^{-2} and Pressure CVs by 10^{-4}.

Location	$\overline{\text{CV}}_T$ (Hyper)	$\overline{\text{CV}}_T$ (MC)	$\overline{\text{CV}}_P$ (Hyper)	$\overline{\text{CV}}_P$ (MC)	$\overline{\text{CV}}$ (Hyper)	$\overline{\text{CV}}$ (MC)
Salvador	0.646	0.965	1.742	2.682	0.332	0.496
Maria da Fé	1.764	2.565	2.057	3.159	0.893	1.298
Carmo	1.333	1.985	3.021	4.638	0.681	1.016
Valença	1.491	2.242	3.104	4.714	0.761	1.145
Timóteo	0.812	1.207	2.538	3.926	0.419	0.623
Duque de Caxias	1.499	2.244	3.639	5.468	0.768	1.150
Pradópolis	1.562	2.250	2.197	3.425	0.792	1.142
Passa Quatro	1.683	2.470	2.202	3.394	0.852	1.252
Montes Claros	1.375	2.006	2.105	3.259	0.698	1.019
São Mateus	0.893	1.338	2.582	3.958	0.459	0.689
Areia	1.010	1.476	1.215	1.890	0.511	0.748
Imperatriz	1.050	1.463	1.505	2.278	0.533	0.743
Florianopolis	0.970	1.467	4.217	6.317	0.506	0.765
Conceição das Alagoas	1.436	2.070	1.983	3.077	0.728	1.050
Brasilia	1.246	1.792	1.570	2.462	0.631	0.908
Campos dos Goytacazes	1.104	1.650	3.366	5.116	0.569	0.851
Itaberaba	1.216	1.768	2.109	3.194	0.618	0.900
Gama ponte alta	1.369	1.957	1.586	2.472	0.692	0.991
Aggregated	1.248	1.829	2.374	3.635	0.636	0.932

G Comparison between Ensemble Forecasting and MC Dropout for others stations

The Figures above show the comparison in the uncertainty forecast approaches for each weather station where the hyperparameter ensemble method forecasts are the top panels and the MC dropout forecasts are the bottom panels. It is observable that the MC Dropout has a trend to produce larger confidence intervals compared to the hyperparameter ensemble approach.

Table 9: Zero-Shot Model Event Counts

Location	Real SH Count	Real Storm Count	Pred. SH Count	Pred. Storm Count
Salvador	61	66	12	58
Carmo	165	99	91	71
Valença	196	136	142	108
Timóteo	86	80	46	68
Pradópolis	148	52	84	49
Gama	187	30	142	25
Areia	0	22	0	13
Maria da Fé	133	38	120	36
Imperatriz	70	44	38	6
Florianopolis	82	90	38	82
Duque de Caxias	131	77	63	64
Passa Quatro	128	52	80	36
Brasilia	157	35	121	20
Campos dos Goytacazes	136	68	80	60
Itaberaba	77	75	6	47
São Mateus	123	90	67	78
Conceição das Alagoas	126	67	108	32
Montes Claros	198	60	168	53
Aggregated	2204	1181	1406	906

Table 10: Zero-Shot Model Performance Metrics

Location	SH Precision	SH Recall	SH F1	Storm Precision	Storm Recall	Storm F1
Salvador	1.000	0.197	0.329	0.862	0.758	0.806
Carmo	0.967	0.533	0.687	0.972	0.697	0.812
Valença	0.944	0.684	0.793	0.935	0.743	0.828
Timóteo	0.870	0.465	0.606	0.941	0.800	0.865
Pradópolis	0.893	0.507	0.647	0.857	0.808	0.832
Gama	0.887	0.674	0.766	1.000	0.833	0.909
Areia	0.000	0.000	0.000	1.000	0.591	0.743
Maria da Fé	0.767	0.692	0.727	0.972	0.921	0.946
Imperatriz	0.684	0.371	0.481	1.000	0.136	0.240
Florianopolis	0.921	0.427	0.583	0.939	0.856	0.895
Duque de Caxias	0.984	0.473	0.639	0.859	0.714	0.780
Passa Quatro	0.988	0.617	0.760	0.917	0.635	0.750
Brasilia	0.950	0.732	0.827	0.950	0.543	0.691
Campos dos Goytacazes	0.825	0.485	0.611	0.850	0.750	0.797
Itaberaba	1.000	0.078	0.145	0.979	0.613	0.754
São Mateus	0.821	0.447	0.579	0.974	0.844	0.905
Conceição das Alagoas	0.833	0.714	0.769	1.000	0.478	0.646
Montes Claros	0.863	0.732	0.792	0.925	0.817	0.867
Aggregated	0.896	0.565	0.675	0.937	0.714	0.796

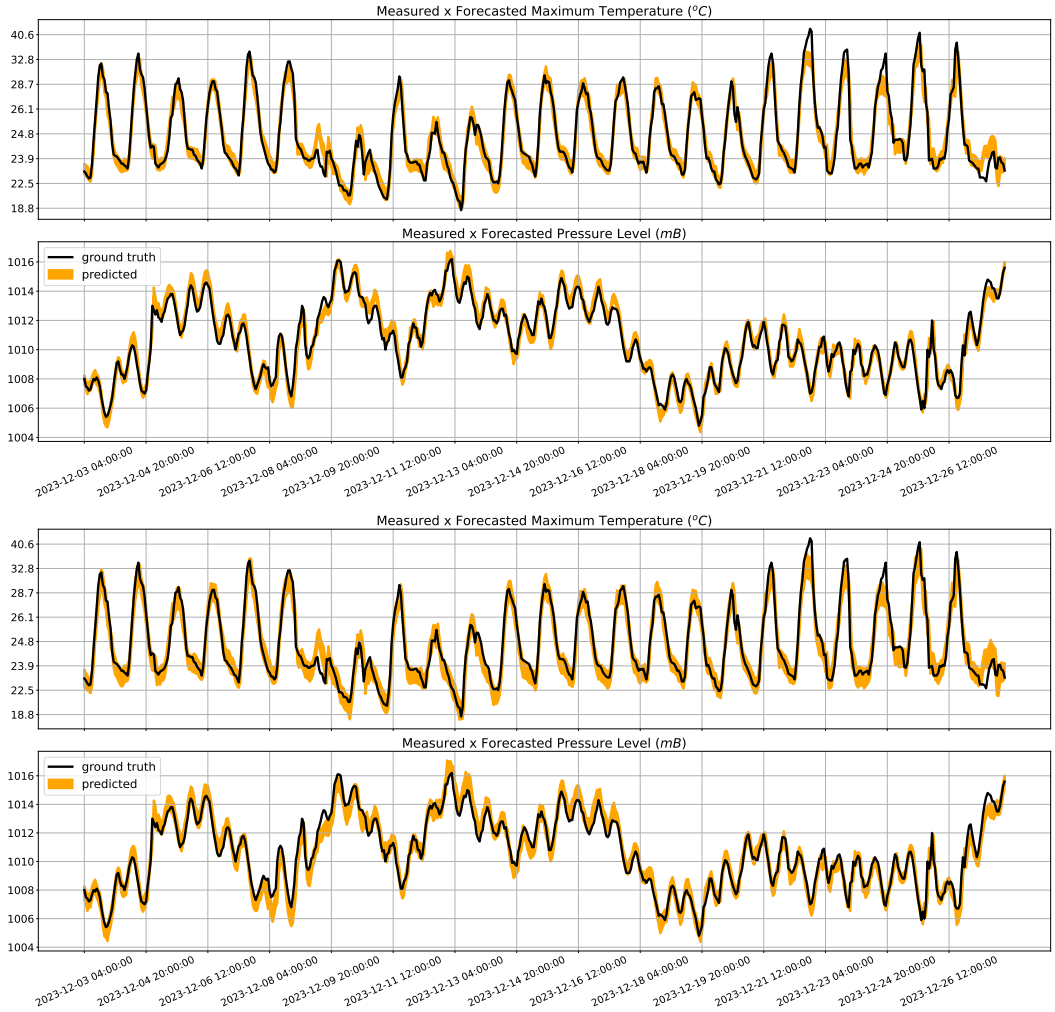

Figure 4: Uncertainty forecast comparison for the hyperparameter ensemble methods (top panel) and the MC dropout method (bottom panel). Station located in Campos dos Goytacazes (Rio de Janeiro state).

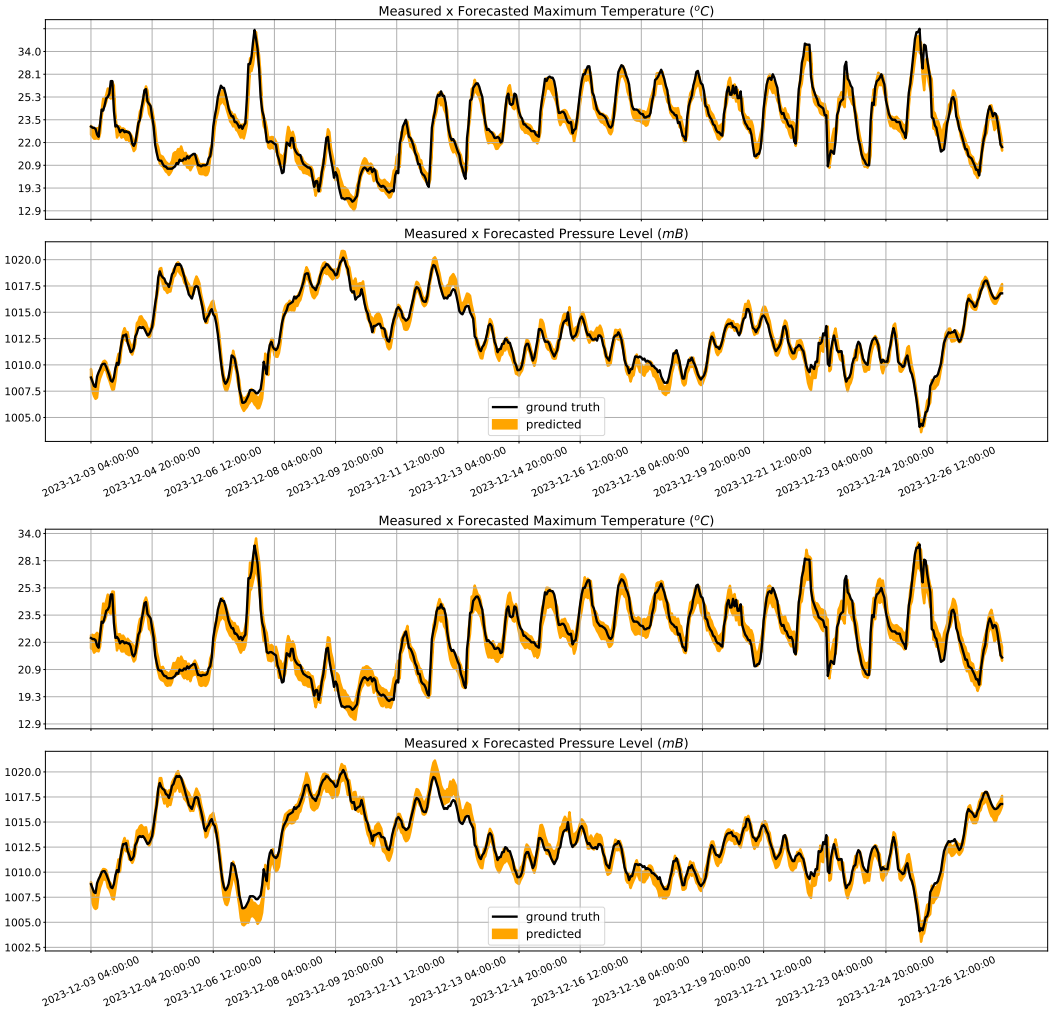

Figure 5: Uncertainty forecast comparison for the hyperparameter ensemble methods (top panel) and the MC dropout method (bottom panel). Station located in Florianópolis (Santa Catarina state capital).

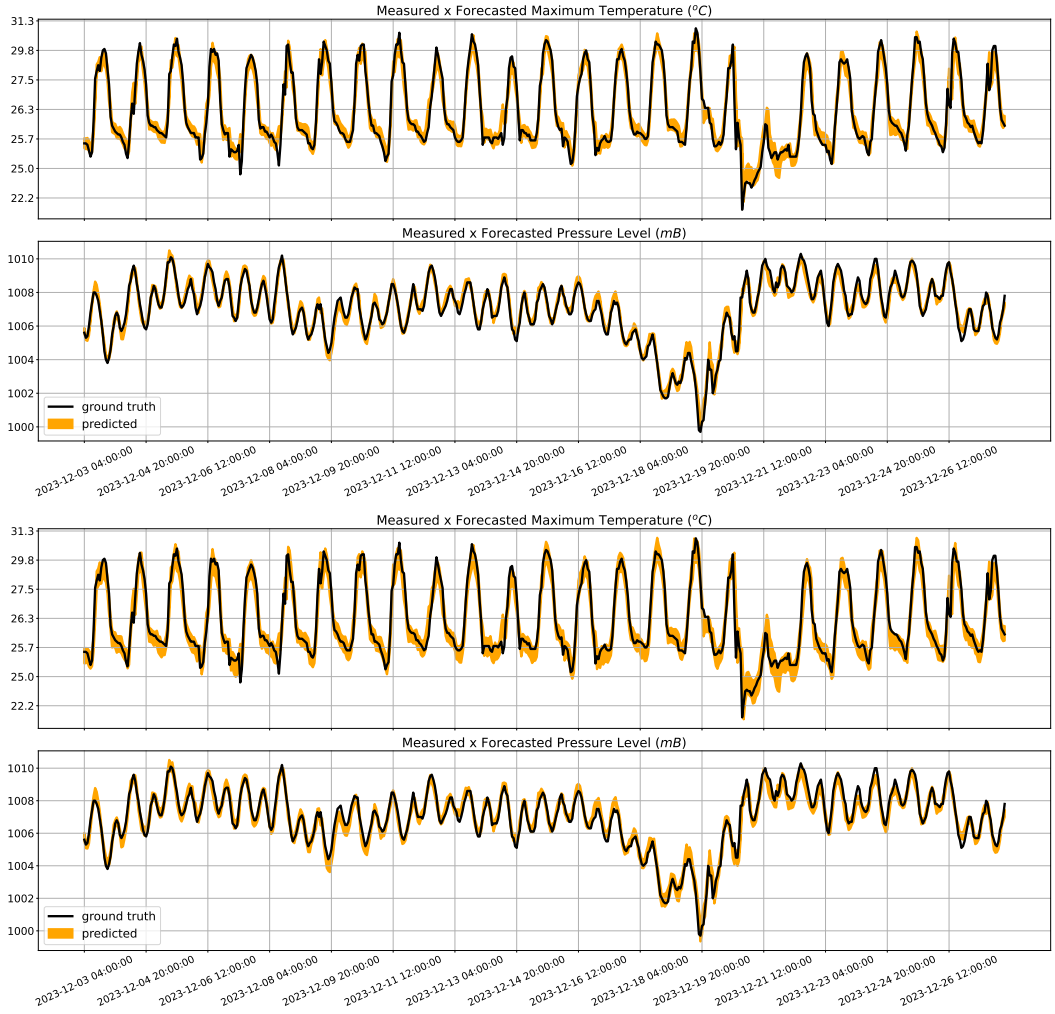

Figure 6: Uncertainty forecast comparison for the hyperparameter ensemble methods (top panel) and the MC dropout method (bottom panel). Station located in Salvador (Bahia state capital).