# Sensor-Drift-Aware Time-Series Anomaly Detection for Climate Stations

Bryce Chen, Victoria Huang and Chen Wang
National Institute of Water and Atmospheric Research
301 Evans Bay Parade, Hataitai, Wellington 6021
Bryce.Chen@niwa.co.nz, Victoria.Huang@niwa.co.nz, chen.wang@niwa.co.nz

## Abstract

*Sensor data collected from climate stations has been used in various scientific applications and environmental monitoring. Maintaining the data quality is essential to guarantee the reliability and accuracy of science outputs, potentially impacting many critical decision making processes. Existing sensor anomaly detection techniques are mostly designed for general purposes, and may not be suitable for climate sensors which require complex handling of seasonality, spatial relationship and sensor interdependency. Current quality control process is deficient in climate sensor drift detection, which is a slow degradation of sensor accuracy over time. Recent development of anomaly detection in climate sensor domain is limited, it's often constrained to particular sensor types, and not focused on drift detection. In this paper, we present a new drift-aware time series anomaly detection framework which leverages the spatial-temporal correlation of the climate sensor network and significantly improves climate sensor drift detection capability. Moreover, the proposed semi-supervised learning approach helps to generalise the solution for various types of sensors and anomalies. Our experiments using real-world dataset have demonstrated promising and competitive performance in regards to sensitivity, false alarm control, and computational efficiency suitable for real-time or near-real-time applications.*

## 1. Introduction

To support the science programmes and commercial applications, a large number of observational stations across New Zealand were established and maintained since 1850 [22, 24]. These stations rely on a wide range of deployed sensors to collect climate observations including wind speed, temperature, precipitation, atmospheric pressure, sunshine duration etc. Hourly and daily sensor data are archived at CliDB, New Zealand's National Climate Database [1].

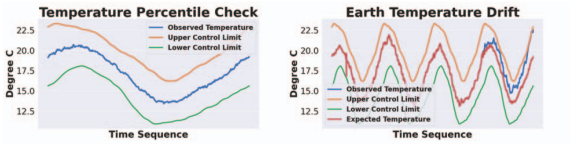
There are two major challenges in managing the sensing system. The first challenge is to automatically and effectively identify anomalies in the sensor data among normal seasonal variations. In particular, real-world data collection is prone to anomalies due to reasons like ambient environment change, data transmission issue, sensor malfunction, wrong calibration and mis-operation. Additionally, sensor drift [1], a subtle anomaly characterized by a gradual drop in sensor accuracy, also happens from time to time due to various reasons, e.g., hardware quality issues and sensor degradation, making it difficult to differentiate from actual weather/climate extremes. The other challenge comes from the scale and heterogeneity of the data. Currently, there are over 600 active stations across New Zealand and each station is equipped with multiple different sensors.

In general, anomalous data are identified during Quality Control (QC) using techniques like range and percentile checks (see Fig. 1(a)), validating against physical constraints and historical statistics. However, these anomalies are often required to be examined manually by experts, which can be time-consuming. Meanwhile, current QC and statistical methods [19] often fail to detect sensor drifts. For instance, the earth temperature data in blue as depicted in Fig. 1(b) has deviated from expected behavior lately, but still resides within statistical percentile limits generated from past seasonal data. Note that data with undetected drifts can be used in weather forecasting tools for months and even years, affecting public understanding and decision-making in critical services like fire stations, city councils, and other governmental agencies. The data will also be used to generate historical statistics used in QC, making drift detection even more difficult.

In this paper, our goal is to address the challenge of anomalous drift detection of various kinds of climate sensors. Our major contributions are: (1) proposed a gen-

[1] CliDB is an online climate data platform provided by National Institute of Water and Atmospheric Research (NIWA), New Zealand. For more information, visit https://cliflo.niwa.co.nz/. It includes climate observation data and statistical summaries from approximately 6500 climate stations.

979-8-3503-5409-6/24/$31.00 ©2024 Crown
DOI 10.1109/CAI59869.2024.00228




(a) Percentile QC check (b) Temperature sensor drift

Figure 1: Sample QC Check and Sensor Drift.

eralised multivariate sensor anomaly detection framework for climate stations via semi-supervised learning approach, with predicted data potentially usable for data imputation; (2) developed a sensor drift detection methodology using predicted sensor data taking into account of uncertainties and achieved minimal false alarms.

## 2. Related Works

Anomaly detection has been investigated in a variety of application domains [2, 5, 10, 12, 21, 30]. For example, Robust Random Cut Forecast (RRCF) [10] introduced by Amazon performs anomaly detection of seasonal behaviors on taxi cab data; SR-CNN [21] from Microsoft transforms time series data into spectral residual for traffic data anomaly detection; DAGMM [30] from NEC lab adopts deep clustering based auto-encoder for anomaly detection on cybersecurity and disease datasets; LOF-based (Local Outlier Factor) algorithms which detects outliers while adapting to the distribution of the data streams such as TADILOF [12] and iLDCBOF [5]; SOM [2] for clustering high-dimensional time series data into similar groups in low-dimensional grid and identifying outliers as anomalies.

However, these methods are not focused on climate domain where a main challenge for anomaly detection is to detect sensor drift which is subtle and almost always coupled with both climate and seasonal changes.

Recently, the development of anomaly detection in climate domain has attracted much interest. For example, an SVR model was developed for various climate sensor data [14]; MAP imputation method together with its own version of anomaly detection algorithms was implemented on weather data [28]; Convolutional Neural Networks (CNNs) was used to spot anomalies on a weekly QC plot charts [23]; Singular Spectrum Analysis (SSA) was applied on air radiation data with k-means based method for outlier detection [16]; Dynamic graph embedding method wad developed for daily climate data and combined with local outlier factor for outlier detection [9]; DBSCAN clustering was applied for multivariate weather anomaly detection [27]; Nayak et al. modelled univariate time-series autoregressively and applied one-class support vector machines for air temperature anomaly detection [17].

Nevertheless, none of these papers considered drift detection. There are recent works on drift-aware anomalies detection [7, 8]. Fenza et al. used LSTM [7] to model sensor behavior in smart grid and monitored the statistical properties over time to detect sensor drift. However, the model did not consider uncertainty in prediction which will lead to a lot of false alarms. The model result is good for anomaly detection, but it's not sufficiently fitted enough for data imputation. Gao et al. [8] proposed to use self-attention model to learn the time series pattern and detect the anomalies by reconstruction error, the uncertainties are implicitly modelled by Gaussian Mixture model. However, the performance is only slightly better than LSTM Encoder-Decoder, and noise handling is not the focus of the paper. Very limited studies [20, 26] have been conducted on drift detection in climate sensor domain, and they only considered one particular sensor and it is difficult to generalize in spatial-temporal perspective. Moreover, many of existing works did not consider spatial correlation of climate stations which can be exploited to improve the detection accuracy.

To address these issues, we propose a drift-aware anomaly detection framework for climate sensor data, making use of spatial and temporal correlation of climate stations and independent sensors. Moreover, a dynamic threshold mechanism is proposed combining both statistical and probabilistic limits to reduce the false alarm rate, maintaining the sensitivity and improve the model robustness.

From system model level, we firstly model the time series data for climate sensor given available inputs with finite time horizon, we then generate the prediction error with respect to observation for each modelled sensor, and finally we perform drift detection by setting up control chart with dynamic thresholds.

## 3. Problem formulation

Given a univariate time series $\mathbf{Y} = \{y_t\}_{t \in T}$, each element $y_t$ is a real-valued observation at time $t$. $y_t$ is classified as an outlier if

$$\{(y_t - \hat{y}_t > \tau_{tu}) \mid (y_t - \hat{y}_t < \tau_{tl})\} \tag{1}$$

where $\hat{y}_t$ is the predicted value at time $t$, $\tau_{tu}$ and $\tau_{tl}$ are thresholds that can be time-variant.

Generally, $\hat{y}_t$ is predicted using a model $f(\cdot)$. The input of $f$ can be a univariate or multivariate time series. Note that climate observation data obtained by different sensors or the data obtained by the same type of sensor but at different locations can be correlated (evidence in Sec. 4.2). We consider $\hat{y}_t$ is derived from multiple co-evolving time series and the past observations of its own:

$$\hat{y}_t = f(\mathbf{X}_{m,t}, \mathbf{Y}_l) \tag{2}$$

where $\mathbf{X}_{m,t} = \{\mathbf{x}_{m,t}\}_{m \in M, t \in T}$ is a multivariate time series, and $\mathbf{Y}_l = \{\mathbf{y}_l\}_{l \in T}$ is an univariate time

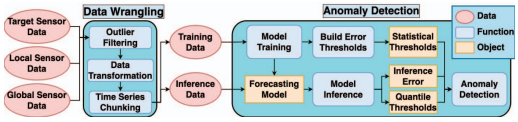



Figure 2: Proposed Anomaly Detection Framework.

series of past observations. Each series $\mathbf{x}_{m,t} = (x_{m,t-L+1}, x_{m,t-L+2}, ..., x_{m,t}) \in \mathbb{R}^L$ is a vector with a fixed-sized chunk length looking back $L$ time steps at time $t$. $\mathbf{y}_l = (y_{t-L+1}, y_{t-L+2}, ..., y_{t-1}) \in \mathbb{R}^{L-1}$ is a vector with a length $L-1$ at time $t$.

# 4. Sensor-Drift-Aware Anomaly Detection Framework

In this paper, we proposed a Sensor-Drift-Aware Anomaly Detection Framework targeting at data collected from climate stations as shown in Fig. 2. In particular, the framework consists of two main components: 1) *Data Wrangling* and 2) *Anomaly Detection*. Data Wrangling module cleans up the data, and passes it on to Anomaly Detection component for semi-supervised learning with model $f$ training and dynamic thresholds derivation. Then Eq. (1) is used to determine if an observation $y_t$ is anomaly.

## 4.1. Data Wrangling

*Outlier filtering*: Obvious outliers are removed using physical limits (e.g., the maximum temperature of a day cannot exceed 100°C). Additional outliers are removed using DBSCAN clustering [6] due to its superb capability in identifying outliers from climate data. Meanwhile, any anomalous features with more than 2% anomalies are also removed at this stage to ensure minimal corruption to the model training process.

*Data transformation*: For wind direction sensor data, data transformation is performed where data is converted from degree to radian to ensure data continuity:

$$cos(\theta_t * 2 * \pi/360) \tag{3}$$

where $\theta$ is the wind direction observation in degree at time $t$. This is important for stochastic gradient descent during optimization process, as 0 degree and 360 degrees in wind directions are the same despite the value difference.

*Data pre-processing*: Time stamp re-indexing is performed and missing sensor values are filled up using data from previous time step. Time-based covariates are extracted as additional features such as month, day, and hour. Uncorrelated sensors and sensor data derived from the same sensor reading as the target sensor are dropped. MinMax scaler is applied to the final training set to address sensor heterogeneity issue. All sensors are inversely transformed and scaled back to original units for anomaly detection and metrics calculation.

*Time series chunking*: Time series chunking is performed to break up the continuous time series into array of fixed length to facilitate model training and inference.

## 4.2. Anomaly Detection

*Prediction model:* For the model $f$ (in Eq. (2)), we make use of deep neural forecasting algorithms to model the sensor pattern at a certain location in a supervised manner, which serves as a kind of time series golden signature for each location and sensor. We can then use this established signature to test against any future data for unseen anomalies in an unsupervised manner as shown in Eq. (1).

To model the observation for one sensor $y_t$, multivariate time series consisting of local and global features are used for the model $f$ inputs. In particular, *local features* are observations from other sensors at the same climate station as they are physically correlated to each other (as evidenced by Fig. 3(a)). For example, grass minimum temperature is correlated to air and earth temperatures, but inversely correlated to soil moisture etc. Meanwhile, *global features* are the observation for the same type of sensor at nearby climate stations. Due to the presence of spatial resemblance (as illustrated in Fig. 3(b)), they can be used as additional predictors as proven in Sec. 5.2. The final predicted $\hat{y}_t$ is described below:

$$\hat{y}_t = \begin{cases} q_{0.5}(\hat{y}_t), & \text{if stochastic} \\ \hat{y}_t, & \text{otherwise} \end{cases} \tag{4}$$

where $q_{0.5}(\hat{y}_t)$ is the 50% quantile probabilistic prediction generated by a stochastic prediction model $f$.

*Dynamic thresholds:* We use both statistical and probabilistic forecasting quantile thresholds to reduce the noise. With the trained model, we are able to derive the model error and used it to compute statistical limits, i.e., upper and lower thresholds using mean $\mu_0$ and standard deviation $\sigma_0$ of the error. In particular, we use first 3 months of data to establish 4 sigma control limits (i.e., $k = 4$), which represents most variation across 4 seasons in New Zealand. On top of that, we also make use of the stochastic forecasting results returned by the forecasting model, specifically 1% and 99% quantile probabilistic predictions (i.e., $q_{0.01}(\hat{y}_t)$ and $q_{0.99}(\hat{y}_t)$) as an additional set of control limits.

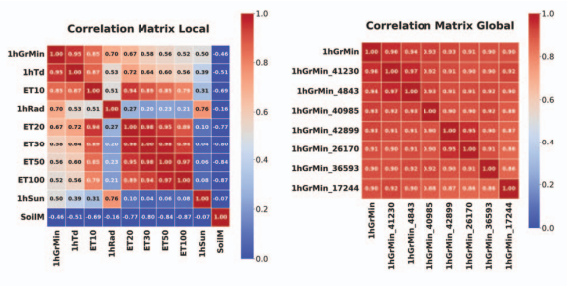
The overall threshold mechanism is represented by $\tau_t$, the set of dynamic threshold at time $t$:

$$\tau_t = (\tau_{tu}, \tau_{tl}), \quad \text{where} \tag{5}$$

$$\tau_{tu} = \begin{cases} max((\mu_0 + k\sigma_0), (q_{0.99}(\hat{y}_t) - \hat{y}_t)), & \text{if stochastic} \\ \mu_0 + k\sigma_0, & \text{otherwise} \end{cases}$$

$$\tau_{tl} = \begin{cases} min((\mu_0 - k\sigma_0), (q_{0.01}(\hat{y}_t) - \hat{y}_t)), & \text{if stochastic} \\ \mu_0 - k\sigma_0, & \text{otherwise} \end{cases}$$

We apply dynamic thresholds and implement drift-aware anomaly detection by setting up the control chart with respect to Eq. (1).




(a) Local feature correlation (b) Global feature correlation

Figure 3: Correlation Heatmaps for Local and Global Features Using Grass Minimum Temperature Observation from Station 17603 as An Example.

## 5. Experiments and Results

### 5.1. Experiment setting

**Dataset.** We collected the raw data from 8 climate stations spreading across New Zealand: Motu (Station ID: 1905), Lauder (5535), Kaitaia (17067), Lincoln (17603), Kainrl (18183), Arthur (25821), Ohakune (31621), and Akitio (38057) from 2018/12/31 till 2022/10/27. We split the raw data into training and testing datasets with a ratio 2:1. In particular, the first two years of raw data are used for training while at least a full year is used for testing. In our process, technicians historically removed entire periods of data when sensors detected anomalies, guided by some specific start and end times. Hence we adopted a simple rule for labeling raw data: if it's absent in CliDB, it's categorized as "Anomaly"; otherwise, it's presumed to be "Normal".

**Baseline Algorithms and Parameter Settings**. 3 popular algorithms were selected for comparison:

- Robust Random Cut Forecast (RRCF) [10]: we selected forest parameters of 100 trees and 256 tree size, and used best performing 0.99 quantile for anomaly detection using collusive dispalcement scores.
- SR-CNN [21]: we used hourly granularity, period of 24 and 95% sensitivity level.
- MTAD-GTN [29] : we set the window size to be 48.

We used the implementation from Bartos et al. [3] for RRCF and Azure Cognitive Service [25] API for both SR-CNN and MTAD-GTN implementations.

In terms of prediction models used in our framework, several algorithms have been considered, including NBEATS [18], NHiTS [4], and TFT [15] which all support probabilistic forecasts and meet our needs. For NBEATS and NHiTS, we used its default architecture, with 3 blocks, 3 stacks, 128 fully connected layer width, and setting the learning rate to be 0.001. For TFT, we used hidden size of 16, 1 LSTM layer, 4 attention heads, setting full attention to be False and learning rate to be 0.001. 20% dropout rate and 30 epoches were both applied for training NBEATS, NHiTS and TFT models. We also include a tree-based ensemble algorithm LightGBM [13] due to its flexibility in handling both past and future covariates although it's deterministic. For LightGBM, we chose "dart" as the boosting type, with 31 num leaves, 9 max depth, 30 estimators, 25% dropout rate, 0.2 for both alpha and lambda regularization. The above four algorithms were implemented based on DARTS [11]. For time series chunking, we set input chunk size to be 32 and output chunk size to be 1.

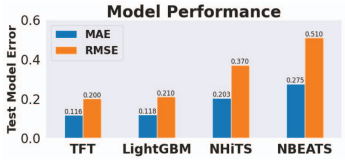



Figure 4: Model Performance on 1hRad for LightGBM, TFT, NBEATS and NHiTS.

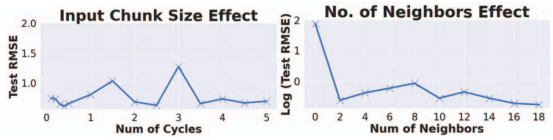
**Metrics**. For performance evaluation, we used Root Mean Square Error (RMSE) and $R^2$ as main metrics, supplemented with popular anomaly classification metrics including accuracy, False Positive (FP) rate, recall, etc.

### 5.2. Experiment results

**Prediction Model Selection.** We compared the four models mentioned above using the radiation data from Kaitaia station, as radiation is a representative example containing both seasonal and sporadic variations. If the model performs well on radiation, it could apply very well to other sensors. As shown in Fig. 4, TFT (RMSE: 0.200) and LightGBM (RMSE: 0.210) clearly outperformed NBEATS (RMSE: 0.510) and NHiTS (RMSE: 0.370) by more than 50%. The reason is that both LightGBM and TFT support future covariate which boost the forecasting performance for future time stamp on top of algorithm efficiency. For the subsequent experiments, TFT is used due to its superior performance and support for stochastic forecasting.

**Impact of Input Chunk Size.** We compared the model performance with different input chunk sizes using the hourly max temperature data from station 38057. As shown in Fig. 5(a), we did not observe very strong effect as our forecasting length is very short (i.e., output chunk size is 1) despite of the RMSE fluctuation. In our experiments, we conservatively used 1.3 times of a daily seasonality cycle of 24 hours, which is 32. We expect similar effects for other sensors as temporal relationships tend to be stronger with recent data for short horizon forecasting.

**Impact of Neighboring Stations.** Similarly, we investigated the impact of global features on the model performance. Specifically, we compared the model RMSE given different numbers of neighbouring stations as inputs. As shown in Fig. 5(b), the performance improves with more




(a) Input chunk size (b) Number of neighboring stations

Figure 5: Input Chunk Size Effect and the No. of Neighboring Stations Effect on Model Performance.

| Station | Metrics | RRCF | SR-CNN | MTAD-GTN | Ours |
|---|---|---|---|---|---|
| **Akitio** | Accuracy | 98.99% | 87.37% | 91.66% | **99.66%** |
| | Precision | 1.21% | 7.61% | 3.65% | **14.61%** |
| | Recall | **63.33%** | 65.91% | 37.68% | 52.38% |
| | FP Rate | 0.99% | 12.61% | 7.85% | **0.26%** |
| | F1-Score | 0.024 | 0.136 | 0.066 | **0.229** |
| **Kainrl** | Accuracy | **96.21%** | 84.85% | 66.70% | 96.04% |
| | Precision | 10.02% | 10.32% | 35.89% | **40.51%** |
| | Recall | 13.78% | **18.83%** | 7.77% | 12.86% |
| | FP Rate | 0.90% | 8.01% | 4.35% | **0.20%** |
| | F1-Score | 0.116 | 0.133 | 0.128 | **0.195** |

Table 1: Baseline Algorithm Benchmarking.

spatial features from neighboring sites. We set the number of neighbouring stations to be 18, which is expected to be different for different sensors and stations, which can be automated and fine-tuned during actual implementation.

**Comparison against Baselines.** Table 1 shows the comparison between our framework and baseline algorithms using data from two stations with the most anomalies. It shows that our method performs best with minimal false alarm rate for both stations while remaining good sensitivity, although it's not best overall in terms of recall rate due to blanket setting of the dynamic threshold mechanism in Sec. 4.2. In reality, both sigma control limits and stochastic quantile thresholds can be fined tuned to achieve better results across different sensors and stations.

The detailed testing results of all 8 stations using our method are summarized in Table 2. Averaging over all sensors, we achieved $R^2 = 0.869$, FP rate = 0.19% and Recall = 20.79% with overall accuracy of 99.22%. The result also shows that our framework generalised well across different sensors, except for wind direction, precipitation and sunlight sensors indicated by "1hGWd", "1hWd", "1hRain" and "1hSun". Wind direction is difficult to predict without co-modelling with wind speed, our experiment used a simple cosine transformation which resulted in loss of information; precipitation is a rare event, additional data balancing or resampling is needed to achieve a better rain event model; while sunshine duration is related to cloud movements, which are lacking in our dataset.

**Drift Detection Performance.** Fig. 6 shows the drift detection mechanism with the top figure being the control chart of prediction errors using statistical thresholds (solid green) and quantile thresholds (dotted pink), and bottom figure being the time trend overlay of predicted and actual

| Sensor | Unit | RMSE | $R^2$ | FP Rate | Recall | Precision | Accuracy |
|---|---|---|---|---|---|---|---|
| 1hGrMin | °C | 1.971 | 0.928 | 0.24% | 36.67% | 2.90% | 99.31% |
| 1hGust | m/s | 1.996 | 0.823 | 0.12% | 0.00% | 0.00% | 99.85% |
| 1hGWd | Degree | 92.573 | 0.566 | 0.14% | 12.02% | 7.73% | 99.77% |
| 1hMax | °C | 1.395 | 0.943 | 0.13% | 3.96% | 7.44% | 99.46% |
| 1hMin | °C | 1.606 | 0.936 | 0.06% | 34.21% | 24.60% | 99.77% |
| 1hRad | $W/m^2$ | 45.475 | 0.947 | 0.45% | 12.00% | 2.00% | 99.53% |
| 1hRain | mm | 0.424 | 0.665 | 0.19% | 0.00% | 0.00% | 99.79% |
| 1hSun | minutes | 11.497 | 0.739 | 0.69% | 0.00% | 0.00% | 99.54% |
| 1hTd | °C | 1.762 | 0.930 | 0.13% | 6.63% | 11.33% | 99.73% |
| 1hWd | Degree | 85.030 | 0.656 | 0.24% | 0.00% | 0.00% | 99.81% |
| 1hWs | m/s | 1.009 | 0.916 | 0.17% | 0.00% | 0.00% | 99.81% |
| 1minPress | hPa | 1.262 | 0.991 | 0.21% | 2.36% | 25.00% | 99.70% |
| 1minRh | % | 12.675 | 0.883 | 0.20% | 22.87% | 31.28% | 92.70% |
| 1minTw | °C | 0.946 | 0.984 | 0.22% | 14.24% | 94.71% | 76.09% |
| ET10 | °C | 2.353 | 0.878 | 0.16% | 16.67% | 33.33% | 99.84% |
| ET100 | °C | 0.580 | 0.948 | 0.00% | 58.33% | 100.00% | 100.00% |
| ET20 | °C | 1.055 | 0.934 | 0.08% | 75.00% | 36.04% | 99.92% |
| ET30 | °C | 1.002 | 0.996 | 0.00% | 0.00% | | 99.99% |
| ET50 | °C | 2.603 | 0.852 | 0.00% | 25.00% | 50.00% | 99.99% |
| **Average** | **NA** | **NA** | **0.869** | **0.19%** | **20.79%** | **14.24%** | **99.22%** |

Table 2: Overall Testing Results by Sensors.

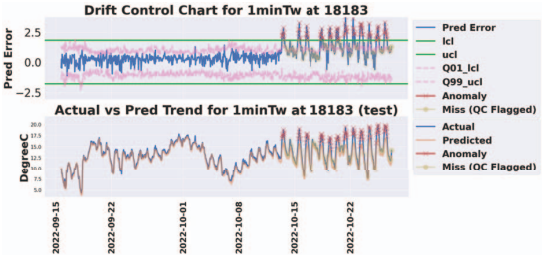



Figure 6: Sensor Drift Detection for 1minTw - Top: Drift Control Chart; Bottom: Actual vs Predicted.

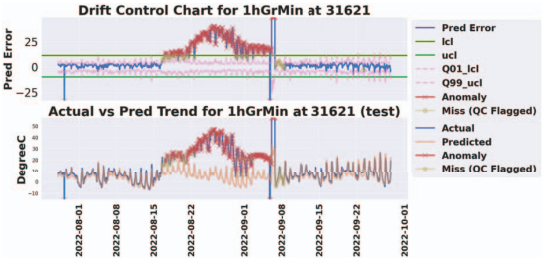



Figure 7: Sensor Drift Detection for 1hGrMin - Top: Drift Control Chart; Bottom: Actual vs Predicted.

observations. The red-cross points represent the captured anomalies which violate the dynamic thresholds. The yellow dots represent "miss". In this case, the control chart captured the drift of model error and it matched well with the sensor drift at the bottom.

Another example is shown in Fig. 7 with the same legend. The prediction error drift in mid August was captured almost immediately when it happened and matched well with sensor drift. The drift disappeared when the sensor completed maintenance in early September.

## 6. Conclusion and Future Works

In this paper, we developed an anomaly detection framework for time series climate sensors. Experiments have shown that our framework works well and able to detect sensor drift with well-controlled false alarms. Our approach has limitations in missing data handling of multivariate inputs and flexibility of hyper-parameter setting in the experiment. Both can be improved during actual deployment. For future directions, we will focus on prediction algorithm improvement and model robustness improvements which include missing input data imputation strategy and model drift detection mechanism. Specialized modelling are required to be developed for sensors like wind direction and precipitation. To further reduce the noises at periods where the sensors pose high but normal variations, we can improve the anomaly detection mechanism with better error threshold control such as adopting seasonal limits, alarm rules etc.

## References


[1] Satnam Alag et al. A methodology for intelligent sensor measurement, validation, fusion, and fault detection for equipment monitoring and diagnostics. *AI EDAM*, 15(4):307–320, 2001. 1

[2] Guilherme A Barreto and Leonardo Aguayo. Time series clustering for anomaly detection using competitive neural networks. In *International Workshop on Self-Organizing Maps*, pages 28–36. Springer, 2009. 2

[3] Bartos et al. RRCF: Implementation of the robust random cut forest algorithm for anomaly detection on streams. *Journal of Open Source Software*, 4(35):1336, 2019. 4

[4] Cristian Challu et al. Nhits: Neural hierarchical interpolation for time series forecasting. In *Proceedings of AAAI*, volume 37, pages 6989–6997, 2023. 4

[5] Ali Degirmenci and Omer Karal. Efficient density and cluster based incremental outlier detection in data streams. *Information Sciences*, 607:901–920, 2022. 2

[6] Martin Ester, Hans-Peter Kriegel, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996. 3

[7] Giuseppe Fenza et al. Drift-aware methodology for anomaly detection in smart grid. *IEEE Access*, 7:9645–9657, 2019. 2

[8] Yang Gao et al. Drift-aware anomaly detection for non-stationary time series. In *2023 IEEE International Conference on Big Data*, pages 1095–1100. IEEE, 2023. 2

[9] Jason J. Jung Gen Li. Entropy-based dynamic graph embedding for anomaly detection on multiple climate time series. *Scientific Reports*, 11(13819), 2021. 2

[10] Sudipto Guha et al. Robust random cut forest based anomaly detection on streams. In *ICML*, pages 2712–2721. PMLR, 2016. 2, 4

[11] Julien Herzen, Francesco Lässig, et al. Darts: User-friendly modern machine learning for time series. *The Journal of Machine Learning Research*, 23(1):5442–5447, 2022. 4

[12] Jen-Wei Huang et al. Tadilof: Time aware density-based incremental local outlier detection in data streams. *Sensors*, 20(20):5829, 2020. 2

[13] Guolin Ke, Qi Meng, et al. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017. 4

[14] Min-Ki Lee, Seung-Hyun Moon, et al. Detecting anomalies in meteorological data using support vector regression. *Advances in Meteorology*, 2018, 2018. 2

[15] Bryan Lim et al. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4):1748–1764, 2021. 4

[16] Yuping Lu, Jitendra Kumar, , et al. Detecting outliers in streaming time series data from arm distributed sensors. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 779–786. IEEE, 2018. 2

[17] Debanjana Nayak and Harry Perros. Automated real-time anomaly detection of temperature sensors through machine-learning. *International Journal of Sensor Networks*, 34(3):137–152, 2020. 2

[18] Boris N Oreshkin et al. N-beats: Neural basis expansion analysis for interpretable time series forecasting. *arXiv preprint arXiv:1905.10437*, 2019. 4

[19] Animesh Patcha et al. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer networks*, 51(12):3448–3470, 2007. 1

[20] Mauricio Pereira et al. Detection and quantification of temperature sensor drift using probabilistic neural networks. *Expert Systems with Applications*, 213:118884, 2023. 2

[21] Hansheng Ren, Bixiong Xu, et al. Time-series anomaly detection service at microsoft. In *Proceedings of the 25th ACM SIGKDD*, pages 3009–3017, 2019. 2, 4

[22] Blake M Seers and Nick T Shears. New zealand's climate data in r—an introduction to clifro. *The University of Auckland, New Zealand.*, 2015. 1

[23] R Srinivasan et al. Machine learning-based climate time series anomaly detection using convolutional neural networks. *Weather and Climate*, 40(1):16–31, 2020. 2

[24] Andrew Tait et al. An assessment of the accuracy of interpolated daily rainfall for new zealand. *Journal of Hydrology (New Zealand)*, pages 25–44, 2012. 1

[25] Ankita Verma et al. A detailed study of azure platform & its cognitive services. In *2019 COMITCon*, pages 129–134. IEEE, 2019. 4

[26] Georg von Arx et al. Detecting and correcting sensor drifts in long-term weather data. *Environmental monitoring and assessment*, 185:4483–4489, 2013. 2

[27] S Wibisono et al. Multivariate weather anomaly detection using dbscan clustering algorithm. In *Journal of Physics: Conference Series*, volume 1869, page 012077, 2021. 2

[28] Tadesse Zemicheal et al. Anomaly detection in the presence of missing values for weather data quality control. In *Proceedings of the 2nd ACM SIGCAS*, pages 65–73, 2019. 2

[29] Hang Zhao, Yujing Wang, et al. Multivariate time-series anomaly detection via graph attention network. In *2020 IEEE ICDM*, pages 841–850. IEEE, 2020. 4

[30] Bo Zong et al. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *ICLR*, 2018. 2