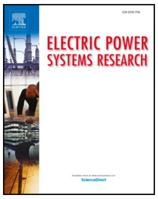
Contents lists available at ScienceDirect

Electric Power Systems Research

journal homepage: www.elsevier.com/locate/epsr



Review of preprocessing methods for univariate volatile time-series in power system applications

Kumar Gaurav Ranjan[a], B Rajanarayan Prusty[b,*], Debashisha Jena[c]

[a] National Institute of Science and Technology, Berhampur, India
[b] Vellore Institute of Technology, Vellore, India
[c] National Institute of Technology Karnataka, Surathkal, India

ARTICLE INFO

Keywords:
False outlier
Outlier detection and correction
Preprocessing
True outlier
Volatile data

ABSTRACT

Outlier detection and correction of time-series referred to as preprocessing, play a vital role in forecasting in power systems. Rigorous research on this topic has been made in the past few decades and is still ongoing. In this paper, a detailed survey of different preprocessing methods is made, and the existing preprocessing methods are categorized. Also, the preprocessing capability of each method is highlighted. The well-established methods of each category applicable to univariate data are critically analyzed and compared based on their preprocessing ability. The result analysis includes applying the well-established methods to volatile time-series frequently used in power system applications. PV generation, load power, and ambient temperature time-series (clean and raw) of different time-step collected from various places/weather zones are considered for index-based and graphical-based comparison among the well-established methods. The impact of change in the crucial parameter(s) values and time-resolution of the data on the methods' performance is also elucidated in this paper. The pros and cons of methods are discussed along with the scope for improvisation.

1. Introduction

In recent years, the integration of renewable generations with electric power systems has increased interest due to environmental and economic benefits. Forecasting of input variables constitutes the foundation for the power system steady-state forecasting [1-3]. The inputs of the above studies include load power, renewable generations, weather variables such as ambient temperature, wind speed, etc. The generation of high-quality synthetic information for such inputs from real-world historical time-series through data preprocessing technique is imperative in obtaining improved accuracy in forecasting [3-7]. The data preprocessing is followed by input selection to reduce the prediction uncertainty [8-11]. Machine learning algorithms have gained popularity in various fields, such as time-series forecasting, mainly modeling the nonlinear relationships among the variables. The time-series of these inputs being highly volatile (volatility in a time-series refers to the phenomenon where the conditional variance varies over time) and often vulnerable to outliers poses several challenges to the preprocessing process. A data point in a time-series is said to be an outlier if (a) it is considerably dissimilar to the remaining points in the dataset (not following the trend and periodic pattern) [12], (b) it stimulates suspicion, i.e., it is missing data [13], (c) it is considerably far from its neighborhood [14], or (d) it appears in a low-density region [14]. The process of detection of these data points is known as outlier detection or anomaly detection. If outlier detection is followed by outlier correction, it is called the preprocessing of data. Preprocessing has a vast area of applications, and it is strenuous to list out all of them. However, the areas where the use of data preprocessing is indispensable are highlighted below:

a) *Forecasting:* Preprocessing increases the accuracy of forecast models that use historical data to forecast load power, PV generation, wind speed, ambient temperature, etc. for power system expansion planning, operational planning and real-time operation [15].
b) *Sensor networks:* Data collected from various wireless sensors exhibit several unique characteristics. In such cases, outlier detection helps in detecting faulty sensors or unusual events such as intrusion [14].
c) *Health care analysis and medical diagnosis:* Detection of abnormal patterns in patients' medical test data helps in the proper diagnosis of the condition and allows doctors to take adequate measures [14].
d) *Fraud detection:* Outlier detection is needed for detecting the fraudulent usage of credit cards and other critical information [16].
e) *Intrusion detection:* Outlier detection helps in intrusion detection, which refers to detecting malicious activity (break-ins, penetrations,

[*] Corresponding author.
E-mail address: b.r.prusty@ieee.org (B.R. Prusty).

https://doi.org/10.1016/j.epsr.2020.106885
Received 21 March 2020; Received in revised form 4 October 2020; Accepted 8 October 2020
Available online 23 October 2020

or other forms of computer abuse) in a computer-related system from a security perspective [17].

f) *Data logs and process logs:* Outlier detection guides companies to gain concealed insights on their websites, which reduces the cost and effort later. Automated outlier detection techniques are used to detect unusual patterns in large volume logs [18].

g) *Image processing:* Outlier detection is needed to detect the changes or abnormal regions in a static image. The domains under this include digit recognition, mammographic image, satellite imagery, video surveillance, and spectroscopy [14].

Apart from the above areas, other outlier detections and correction applications include finance, text and social media, earth sciences, the internet of things (IoT), etc.

Rigorous research on the preprocessing of volatile time-series has been made in the past six decades and is still ongoing. In 2005, the first attempt was made to compare the existing preprocessing methods [19]. But, until then, a limited number of preprocessing methods were available. From 2009 to 2016, several elegant works were produced to compare the different preprocessing methods [20-27]. Though many methods were included in the review, few important methods were not thoroughly analyzed or were ignored [15,18,20,21], or a particular category of preprocessing methods [21,22,24,27] were focused. In 2019, a meticulous study was proposed that categorizes a vast number of preprocessing methods and highlights their merits and lacunae [14]. Nevertheless, few methods were excluded from the analysis. The preprocessing capability of all the methods, comparison among the methods, and applicability of different methods to different data is not elucidated. The impact of significant parameter values and the time-step of the data on the preprocessing methods' performance were not analyzed. All the above-listed lacunae are addressed in this paper.

The significant contributions of this paper are as follows:

- Various outlier detection and correction approaches are categorized, and the preprocessing capability of each method is highlighted.
- Well-established methods of each category (only applicable to univariate data) are chosen for the analysis, and their algorithms are explained in detail.
- The well-established methods are applied to various time-series, and their performances are critically analyzed and compared.
- The impact of significant parameter values on the performance of the well-established methods is elucidated.
- The effect of the time-resolution of the data on the well-established methods' performance is elaborated.

This paper aims to visualize the differences among the preprocessing methods in a lucid manner and guide a novice researcher to choose the best preprocessing way according to the type of data and its application.

The paper is organized as follows: In Section 2, existing preprocessing methods are categorized. Well-established preprocessing methods are explained in Section 3, and the basis for comparison of those methods is discussed in Section 4. Result analysis, including a detailed comparison amidst different methods on various platforms, merits, and lacunae of each method and scope for improvisation, is included in Section 5. The concluding remarks are provided in Section 6.

2. Existing preprocessing methods

In this paper, preprocessing methods explicitly applicable to time-series are considered for analysis. There are also preprocessing methods that apply to data that are not time-series [28]. The existing outlier detection and correction methods can be categorized into four major types, defined as under:

a) *Statistical-based methods:* In these types of methods, the data points are modeled using a stochastic distribution and outliers are detected based on the relationship with the distribution model. Under the assumption that the distribution fits the dataset, the outliers are those points that do not agree with or conform to the underlying model for the data.
   i) Parametric methods: These statistical-based methods assume the time-series to follow a specific parametric distribution model or exhibit some characteristics.
   ii) Non-parametric methods: These types of statistical-based methods do not assume the time-series to follow a specific parametric distribution model or exhibit any characteristics.

b) *Density-based methods:* The density-based outlier detection methods work on the core principle that an outlier can be found in a low-density region. Simultaneously, non-outliers (also known as inliers/genuine data points) are assumed to appear in dense neighborhoods. They compare the local points' densities with their local neighbors' densities. Density-based methods use sophisticated mechanisms to model the outlierness (degree of being an outlier) of data points. Usually, it involves investigating the local densities of the point being studied and its nearest neighbors. The outlierness metric of a data point is typically a ratio of its density against its nearest neighbors' averaged densities, making it relative. Density-based methods show a more robust modeling capability but, at the same time, require expensive computation.

c) *Distance-based methods:* These methods detect outliers by computation of the distances between points, where the data points far from their nearest neighbor are considered outliers. Distance-based outliers can also be defined, given the distance measure of feature space, as:
   i) Points lying within the distance d but having less than p different samples.
   ii) The top n points farthest from the k^{th} nearest neighbor.
   iii) The top n points with the greatest average distance to the k nearest neighbors.

Distance-based outlier detection approaches are moderate and scale well for large data sets having medium to high dimensionality. They are flexible, computationally efficient, and tend to have a robust foundation.

a) *Cluster-based methods:* Clustering-based techniques usually rely on the usage of clustering methods to describe data behavior. For this, the small clusters with significantly fewer data points than others are labeled as outliers.

The existing preprocessing methods are categorized and chronologically shown in Table 1. Their outlier correction capability is mentioned with it. Further, the table also contains information about any available improvisation(s) in those methods and the outlier correction capability in their improvised versions. It is worthy to note that statistical-based and distance-based methods apply to univariate time-series. In contrast, density-based and cluster-based methods are only appropriate for multivariate time-series. The advantages and disadvantages of the existing preprocessing methods are listed in Table 2.

3. Well-established preprocessing methods

Amidst the numerous preprocessing methods, each category's well-established techniques are analyzed in detail. SWP and portrait dataset-based methods are parametric statistical methods for preprocessing. They are found to be applied in several articles owing to their decent preprocessing capabilities [34-36,38]. The SWP-based method is widely used because of its ability to trace the trend and seasonality of time-series better than other algorithms. The portrait dataset-based method gives a new way of visualizing data, portraying the data's hidden

Table 1
Classification of existing preprocessing methods.

Category		Year	Original method	OCC	Improvised version	OCC
Statistical-based	Parametric	1989	ARMA-based [29,30]	No	–	–
		2005	Chebyshev theorem-based [31]	No	Integrated symbolic aggregation approximation-based [32]	Yes
		2008	ARMAsel-based [33]	Yes	–	–
		2010	Sliding window prediction (SWP)-based [34-36]	Yes	–	–
		2010	Kernel density estimation-based [37]	No	–	–
		2014	Portrait dataset-based [38]	Yes	–	–
		2014	ARIMAX-based [39]	Yes	–	–
		2016	Outlier characteristics-based [40]	Yes	–	–
		2016	Bayesian classifier-based [41]	No	–	–
		2016	Bernoulli Detector [42]	No	–	–
		2017	ARX and ANN model-based [43]	Yes	–	–
	Non-parametric	1964	Huber-skip M-estimator-based [44,45]	No	–	–
		2005	Second order difference-based [46]	Yes	Integrated symbolic aggregation approximation-based [32]	Yes
		2010	B-spline smoothing-based [47]	Yes	–	–
		2010	Skeleton point-based [48]	No	–	–
		2011	Kernel smoothing-based [49]	Yes	–	–
		2011	Robust NP-SC [50]	Yes	–	–
		2011	LS-SV [51]	No	–	–
		2014	Arithmetic progression-based [52]	No	–	–
		2018	DRM-based [53]	Yes	–	–
		2018	Time-Window-based [54]	Yes	–	–
		2019	Probabilistic deep autoencoder [55]	Yes	–	–
Density-based		2000	Local outlier factor (LOF) [3,56]	No	Connective-based outlier factor (COF) [57] LOcal Correlation Integral (LOCI)-based [58] Influenced outlierness (INFLO) [59] Distributed LOF computing (DLC) [60]	No No No No
		2004	Relative density factor (RDF) [61]	No	–	–
		2007	Kernel density functions-based [62]	No	RKOF [63] Weighted Kernel estimation-based [64] KDE-based [65] Adaptive kernel density-based [66] KELOS [67]	No No No No No
		2009	Local outlier probabilities (LoOP) [68]	No	–	–
		2009	Resolution-based outlier factor (ROF) [69]	No	Dynamic window outlier factor (DWOF) [70]	No
		2017	Relative Density-Based Outlier Score (RDOS) [71]	No	–	–
Distance-based		2011	Grubbs methods [72]	No	Modified Grubbs method [73]	No
		2014	Nearest neighbors-based [74]	No	*k*NN-based [75] Adapted *k*NN [76]	No No
		2015	LDOF [77]	No	–	–
Cluster-based		2007	D-stream [78]	No	–	–
		2008	*k*-means clustering based [79,80]	No	*k*NN-SVM [81] *k*-medoids-based [82] MST inspired *k*NN-based [83] *k*NN-FSVM [84]	No No No No
		2000	Isolation forest [85-87]	No	Extended isolation forest [88] *k*-means-based isolation forest [89] SSO-IF [90]	No No No
		2009	SDStream [91]	No	–	–
		2010	Intelligent outlier detection algorithm (IODA) [92]	No	–	–
		2012	Weighting attributes in clustering-based [93]	No	–	–

Note: OCC stands for “outlier correction capability.”.

characteristics like trend and seasonality, thus enhancing the approach's performance. B-spline smoothing-based method [47] is a non-parametric statistical method used by many researchers as a benchmark to compare their methods. This approach can preprocess any volatile time-series without making any assumption about the data's distribution or periodicity. The *k*-nearest neighbor (*k*NN)-based method [75] is one of the most widely used distance-based methods for outlier detection, because of its ease application to various kinds of data without knowing the distribution or characteristics of the data. In addition to the merits mentioned above, selecting these parameter values is less tedious than other methods.

The SWP-based, the portrait data-based, and the B-spline smoothing-based preprocessing methods and the *k*NN-based outlier detection method are elaborated respectively in Sections 3.1, 3.2, 3.3, and 3.4. The above approaches are explained by considering a univariate time-series $X = \{x_1, x_2, ..., x_n\}$of length n.

3.1. Sliding window prediction-based approach

It is a prediction-based method which uses the nearest neighborhood of data [34-36]. Considering a time-series X, the k'^{th} nearest neighborhood $n_i^{k'}$of width $2k'$ of a data point x_i at a time instant i is defined as,

$$n_i^{k'} = \{x_{i-2k'}, x_{i-2k'+1}, ..., x_{i-1}\}, \tag{1}$$

where $n_i^{k'}$is one-sided window which considers $2k'$ data points before x_i.

A both-sided window which considers k' data points before x_i and k' data points after x_i is given as

$$n_i^{k'} = \{x_{i-k'}, x_{i-k'+1}, ..., x_{i-1}, x_{i+1}, x_{i+2}, ..., x_{i+k'}\}. \tag{2}$$

In general, one-sided window is preferred over both-sided window in case of outlier detection as k' data points after x_i may have undetected and uncorrected outliers. The predicted value of x_i, i.e., x_i', based upon one-sided nearest neighborhood $n_i^{k'}$is defined as

Table 2
Advantages and disadvantages of existing preprocessing methods.

Statistical-based	Parametric	Advantages	• Mathematically admissible, and have a high performance speed once the model is built. • Efficient and it is possible to interpret the outliers in real sense. • Most of the methods are capable of giving an approach to correct outliers.
		Disadvantages	• The knowledge about the distribution of the data is needed a priori. The data may not precisely fit the assumed distribution, which will result in inappropriate modeling and degrading the performance. • Typically, not applicable to multivariate data from the view point of computational cost.
	Non-parametric	Advantages	• In addition to all the advantages of the parametric methods, they do not make any assumption about the distribution of the data. Therefore, no prior information about the distribution of the data is required for modeling and preprocessing.
		Disadvantages	• Not suitable for multivariate and large datasets as intensive computations are involved.
Density-based		Advantages	• Do not depend on the assumed distribution of the data to calculate the density estimate, which are further used to detect outliers. • Capable of performing well in multivariate data, and can detect outliers nearby dense regions. • Require very less parameter optimization for an optimal performance.
		Disadvantages	• Complicated and computationally intensive methods. • Not applicable to univariate data. • No approach to correct outliers.
Distance-based		Advantages	• Simple to understand as well as to comprehend. • Do not depend on the assumed distribution to fit the data.
		Disadvantages	• Incapable of correcting outliers. • Not applicable to multivariate data. • Not preferable for the preprocessing of large datasets.
Cluster-based		Advantages	• Independent of the distribution of the data. • Capable of detecting outliers in multivariate data.
		Disadvantages	• Not applicable to univariate data, nor capable for correcting outliers. • Cannot detect outliers which occur in cluster, as it is against their assumption. • Methods are binary in nature; hence do not give any quantitative indication of a data point's outlierness.

$$x_i' = \frac{\sum_{j=1}^{2k'} w_{i-j}x_{i-j}}{\sum_{j=1}^{2k'} w_{i-j}}, \tag{3}$$

where $w_{i-j}$is the weight of data point x_{i-j}, which is inversely proportional to the distance between points $x_i$and x_{i-j}, which is given as

$$w_{i-j} = \frac{1}{|x_i - x_{i-j}|}. \tag{4}$$

Greater the distance between the data points x_i and x_{i-j}, smaller is the weight w_{i-j}, indicating that the dependence of $x_i$on $x_{i-j}$is insignificant and vice versa. Based on the prediction in (3), predicted confidence interval (PCI) for x_i is given as,

$$\mathrm{PCI} = x_i' \pm \left(q_{\frac{\alpha}{2},2k'-1} \times s\sqrt{1 - \frac{1}{2k'}}\right), \tag{5}$$

where $q_{\frac{\alpha}{2},2k'-1}$is the $100 \times (1-\alpha)$%percentile critical value for the Student's t-distribution with $2k'-1$degrees of freedom and s is the standard deviation of the neighborhood sample $n_i^{k'}$.

If x_i does not lie within PCI, it is treated as an outlier. The corrupted data point x_i is replaced by the corrected data, which is the predicted value x_i'using (3). Finally, the window is updated for the next data point. This detection and correction process continues until the last data point of the time-series is checked. The flowchart for preprocessing using SWP is as given in Fig. 1.

3.2. Portrait dataset-based approach

The method [38] applies a new data visualization technique, termed portrait, on the data by analyzing the periodic patterns in it and reorganizes the data for ease of analysis. The portrait data are composed of data points falling within the corresponding time intervals, i.e., the j^{th} basic portrait dataset (BPD) p_j is constructed as

$$p_j = \{x_i | i = j + gT, j = 1, 2, \cdots, T, g = 0, 1, \cdots, N^{'}, N^{'} = n/T\}, \tag{6}$$

where T is the fundamental period which is calculated by using Fast Fourier Transform (FFT).

The characteristic vector of basic portrait dataset p_j is defined as

$$v_j = [\theta_j, M_j], \tag{7}$$

```
Begin: Input the given univariate time-series $X$ and select a window width $2k'$.
for $i = 2k' + 1$ until $i = n$
    do
    •   Compute the $k'^{\text{th}}$ nearest neighborhood $n_i^{k'}$ using (1).
    •   Compute the predicted value $x_i'$ of $x_i$ using (3).
    •   Compute the PCI for $x_i$ using (5).
     if $x_i$ is not within the PCI
        Replace the data point $x_i$ with its predicted value $x_i'$.
    end if
end for
```

Fig. 1. Algorithm for SWP-based preprocessing.

where θ_j and M_j represent the median and the median absolute deviation (MAD) of p_j, respectively.

The similarity between two BPDs p_j, $p_{j'}$with characteristic vectors v_j, $v_{j'}$, respectively, is defined as

$$s_{jj'}' = \begin{cases} \infty \ \text{if}\ v_j = v_{j'} \\ 1/\|v_j - v_{j'}\|_2\ \text{, otherwise} \end{cases}. \tag{8}$$

Multiple BPDs with similar characteristic vectors are merged into one virtual portrait dataset (VPD) to speed up the preprocessing. Once the landscape data is converted into portrait datasets, each portrait dataset's outlier region can be calculated according to the type of distribution the dataset follows and its length. Since performing statistical tests on data, which is affected by outliers, to find the distribution is not advisable, several potential cases for outlier detection are considered.

Case 1: For a portrait dataset following normal distribution $N(\mu, \sigma^2)$ and for any confidence coefficient $\alpha, 0 < \alpha < 1$, α-outlier region is given as

$$out(\alpha, (\mu, \sigma^2)) = \left\{x: |x - \mu| > Z_{1-\frac{\alpha}{2}}\sigma\right\}, \tag{9}$$

where μ and σ are the mean and standard deviation of the portrait dataset, respectively and $Z_{1-\alpha/2}$is the $100 \times (1 - \alpha/2)$percentile of $N(0, 1)$. Since the data contains outliers, median and MAD are used instead of mean and standard deviation.

Case 2: For a portrait dataset following gamma distribution $G(\beta,\ \phi)$ with shape parameter β and scale parameter ϕ, its α-outlier region is given as

$$out(\alpha,\ (\beta,\ \phi)) = \left\{x\colon x < F_{\frac{\alpha}{2}}^{-1}(\beta,\ \phi) \text{ or } x > F_{1-\frac{\alpha}{2}}^{-1}(\beta,\ \phi)\right\}, \tag{10}$$

where $F^{-1}$is the inverse cumulative distribution function of $G(\beta,\ \phi)$, and $F_{1-\alpha/2}^{-1}(\beta,\ \phi)$is the $100 \times (1 - \alpha/2)$percentile of $G(\beta,\ \phi)$.

Case 3: When the dataset is small in length, box plot is used to define the outlier region and it is given as

$$out(\rho,\ (Q_1,\ Q_3)) = \{x\colon x < Q_1 - \rho \times \text{IQR} \quad \text{or } x > Q_3 + \rho \times \text{IQR}\}, \tag{11}$$

where ρ is an index of significance, Q_1 and Q_3 are the 25th and 75th percentiles, respectively, and the difference $(Q_3 - Q_1)$is called inter-quartile range (IQR).

A data point lying in the outlier region of the corresponding portrait dataset is treated as an outlier. Then the outlier is replaced by an acceptable value of the corresponding dataset, i.e., the mean value for Case 1 and 3 and the value of β/ϕ for Case 2. The flowchart for portrait data-based preprocessing is as given in Fig. 2.

3.3. B-spline smoothing-based approach

The method [47] is based on the modeling of inherent patterns of a time-series to detect the presence of outliers. The data can be can modeled as

$$X = m(t) + \varepsilon, \tag{12}$$

where $m(t)$ is the underlying function and ε is the error term which is assumed to be normally and independently distributed with mean of zero and constant variance σ^2.

The foremost task is to find a suitable estimate of the function $m(t)$, i.e., $\hat{m}(t)$, using the collected data. Based on the estimate of $m(t)$, the point-wise confidence interval is built. Data points lying outside the confidence interval are marked as outliers and are replaced by the estimated value $\hat{m}(t_i)$. In order to estimate $m(t)$, B-spline smoothing uses a basis function system consisting of a set of known basis functions and it is expressed in vector form as

$$m(t) = c^{\mathrm{T}}\vec{\varphi}, \tag{13}$$

where $\vec{c} = (c_1,\ c_2, ..., c_b)$is the coefficient vector, $\vec{\varphi} = (\varphi_1,\ \varphi_2, ..., \varphi_b)$is the vector of basis functions and b denotes the number of basis functions considered. In order to estimate the coefficients c from the data, a $n \times b$ matrix is defined as

$$\Phi = \begin{bmatrix} \varphi_1(t_1) & \varphi_2(t_1) & \cdots & \varphi_b(t_1) \\ \varphi_1(t_2) & \varphi_2(t_2) & \cdots & \varphi_b(t_2) \\ \vdots & \vdots & & \vdots \\ \varphi_1(t_n) & \varphi_2(t_n) & \cdots & \varphi_b(t_n) \end{bmatrix}, \tag{14}$$

```
Begin: Input the given univariate time-series $X$ .
    • Calculate the fundamental period $T$ using FFT.
    • Construct the BPDs using (6).
    • Construct the characteristic vector $v_j$ for each BPD using (7).
    • Construct the similarity matrix $s'$ using (8).
    • Merge the similar BPDs into minimum number of VPDs.
for $i = 1$ until $i =$ total number of VPDs
    do
        • Calculate the outlier region for the $i^{\text{th}}$ VPD using (9), (10) or (11).
    for $j = 1$ until $j =$ length of $i^{\text{th}}$ VPD
        do
        if $j^{\text{th}}$ data point lies in the outlier region
            Replace the data point with an acceptable value of $i^{\text{th}}$ VPD
        end if
    end for
end for
```

Fig. 2. Algorithm for portrait data-based preprocessing.

```
Begin: Select a time-series $X$ .
    • Construct the basis function vector $\vec{\phi}$ .
    • Construct $\Phi$ using (14).
    • Calculate the coefficient vector $\hat{c}$ using (15).
    • Calculate the fitted value vector $\hat{X}$ using (16).
    • Calculate the degrees of freedom df using (18).
    • Calculate the square of predicted error $(\hat{E}_i)^2$ using (19), (20) and (21).
for $i = 1$ until $i = n$
    do
        • Select a data point $x_i$ .
        • Calculate the confidence interval for $x_i$ using (22).
    if $x_i$ is not within the confidence interval
        Replace $x_i$ with its fitted value $\hat{x}_i$ .
    end if
end for
```

Fig. 3. Algorithm for B-spline smoothing-based preprocessing.

```
Begin: Select a time-series $X$ .
    • Prepare the embedding matrix $X_{em}$ as in (23).
    • Calculate the SED of each window from all other windows using (24), except for near-in-time window(s).
    • Assign smallest SED of a window as the outlier index of that window.
    • Calculate the threshold value of outlier index.
for $e = 1$ until $e = n - L' + 1$
    do
        • Select a window $r_e$ .
        • Compare its outlier index $OI_e$ with the threshold value $OI^{\alpha}$ .
        if $OI_e > OI^{\alpha}$
            Mark all in data points in the $r_e$ as outliers.
        end if
end for
```

Fig. 4. Algorithm for *k*NN-based outlier detection.

Table 3
Significance of crucial function(s)/parameter(s) of different methods.

Method	Crucial function(s)/parameter (s)	Significance	
SWP-based	Window width ($2k'$)	It decides the number of input data points given to the model for the preprocessing of outliers. Inappropriate window width will lead to under/over prediction.	
Portrait data-based	Fundamental time period (T)	It guides the construction of primary portrait datasets. The inaccurate period will result in unstable portrait datasets.	
B-spline smoothing-based	Basis function vector ($\vec{\phi}\,(t)$)	It is the soul of the method. From curve fitting to preprocessing of outliers, depends on how well the basis function(s) is chosen. Everything else is derived from it.	
*k*NN-based	Window length (L')	It decides the neighborhood of a data point. Similarity among windows increases with the window length.	Both are used to decide the threshold value of the outlier index.

Table 4
Comprehensive description of the time information and data samples of various cases under consideration.

Case No.	Data	Place/Weather zone	Notation	Data type	Time information	Time-step	Sample size
1	PV generation	Parkesburg, USA	PV	Raw	Noon, 2012–2013	Daily	730
2	Load power	Anchorage, USA	P_{D1}	Clean	Jan, 2004	Hourly	744
3		North-Central Texas, USA	P_{D2}	Raw	Jan, 2012	Hourly	744
4			P_{D3}	Raw	10 am, 2012–2013	Daily	730
5	Ambient temperature	Berhampur, India	T_{AMB1}	Raw	Jan, 2010	Hourly	744

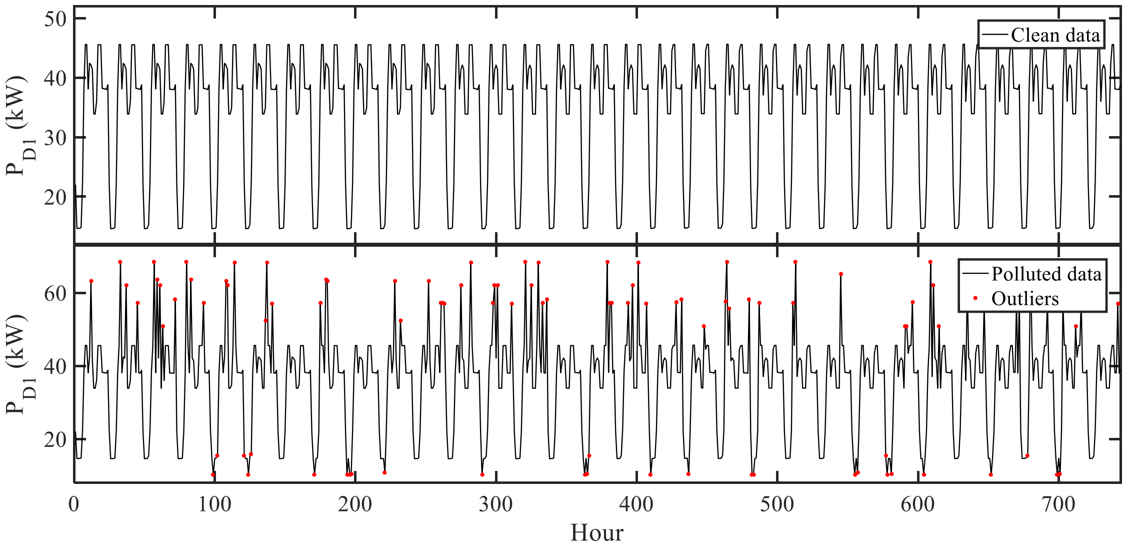


Fig. 5. Clean and polluted univariate load power data.

where $\Phi[i, j] = \phi_j(t_i)$represents the value of the j^{th} basis function at time t_i.

Theestimate of the coefficient vector c is obtained as

$$\widehat{c} = (\Phi^{\mathrm{T}}\Phi + \lambda R)^{-1}\Phi^{\mathrm{T}}X, \tag{15}$$

where λ is the smoothing parameter, $R = \int D^2\vec{\phi}\,(t)D^2\vec{\phi}\,(t)^T dt$, and D^2(.) is the second order derivative operator.

The fitted value vector $\hat{X}$is computed by

$$\widehat{X} = \Phi\hat{\vec{c}}, \tag{16}$$

or

$$\widehat{X} = SX, \tag{17}$$

where $S = \Phi(\Phi^T\Phi + \lambda R)^{-1}\Phi^T$is called hat matrix, as it converts the dependent variable vector X into its fitted value $\hat{X}$.

The number of degrees of freedom of the fit is the trace of the hat matrix

$$\mathrm{df} = \mathrm{trace}(\mathrm{S}), \tag{18}$$

where trace(.) calculates the sum of diagonal elements of a matrix.

The square of estimated predicted error is given by

$$(\widehat{E_i})^2 = \mathrm{MSE} + (E_i(\widehat{x_i}))^2, \tag{19}$$

where MSE is the mean square error calculated as

$$\mathrm{MSE} = \frac{1}{n{-}\mathrm{df}} \sum_{i=1^n} (x_i - \widehat{x_i})^2 \tag{20}$$

and $(E_i(\hat{x}_i))^2$is the sampling variance to the fit at time t_i and is given by diagonal elements of the matrixVar$[\hat{X}]$

$$\mathrm{Var}[\hat{X}] = S \times S^{\mathrm{T}} \times \mathrm{MSE}. \tag{21}$$

For a given significance level α, the $100 \times (1 - \alpha)\%$confidence interval at time t_i is computed as

$$\left[\widehat{x_i} - Z_{1-\alpha/2} \times \widehat{E_i}, \widehat{x_i} + Z_{1-\alpha/2} \times \widehat{E_i}\right], \tag{22}$$

where $Z_{1-\alpha/2}$is the $100 \times (1 - \alpha/2)$percentile of the standard normal distribution.

Any data point outside the confidence interval is treated as an outlier. The flowchart for data preprocessing using B-spline smoothing

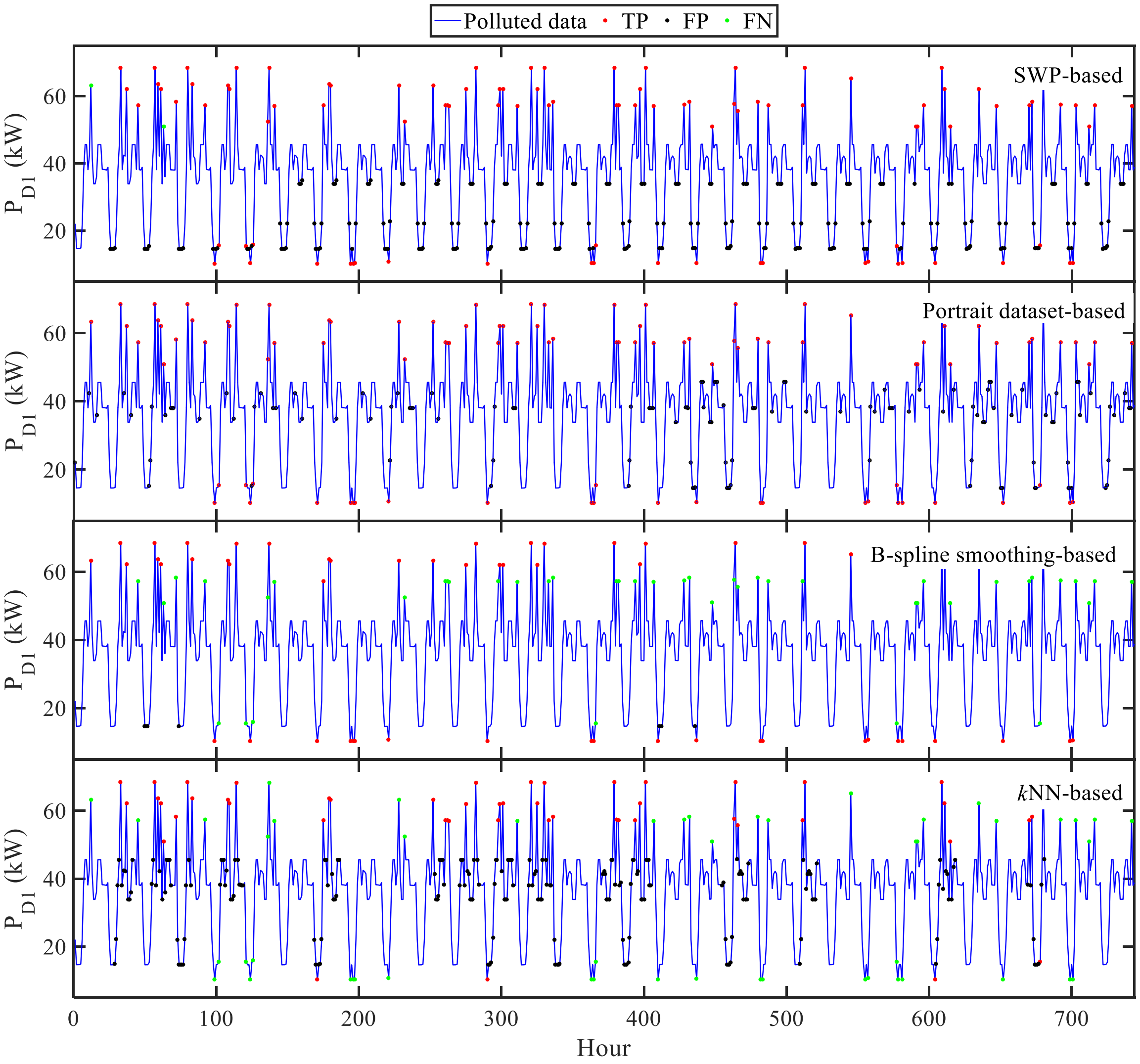


Fig. 6. Outlier detection performance when applied to univariate polluted data.

Table 5
Outlier detection performance comparison when applied to univariate polluted data.

Indices	SWP-based	Portrait dataset-based	B-spline smoothing-based	*k*NN-based
No. of TP	98	100	56	50
No. of FP	208	133	7	187
No. of FN	2	0	44	50
P	0.3203	0.4292	0.8889	0.2110
R	0.98	1	0.56	0.5
F-measure	0.4828	0.6006	0.6871	0.2967

is as shown in Fig. 3.

3.4. k-nearest neighbor-based approach

The method segregates the entire time-series into several groups or windows [75]. The time-series X can be divided into windows as

$$X_{em} = \begin{bmatrix} x_1 & x_2 & \dots & x_{L'} \\ x_2 & x_3 & \dots & x_{L'+1} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n-L'+1} & x_{n-L'+2} & \cdots & x_n \end{bmatrix}, \tag{23}$$

where X_{em} is the embedding matrix, row r_e denotes the e^{th} window and L denotes the number of data points in each window.

Each window of X_{em} is then compared with the other windows, using the square of Euclidean distance (SED) as follows:

$$d^2(r_e, r_f) = \sum_{j=1}^{L'} \left(x_{e-j+L'} - x_{f-j+L'}\right)^2. \tag{24}$$

Accordingly, an outlier index value OI_e for the e^{th} window r_e is determined, which is the k^{th} smallest SED between r_e and all other windows except for the near-in-time window(s) of r_e. The near-in-time windows of r_e are those which have at least one data point in common with r_e. Once each window attains its outlier index value, it is necessary to determine a threshold value for outlier detection based on the obtained sequence of outlier indices $\{OI_e\}_{e=1}^{n-L'+1}$. The threshold value of outlier index $OI^{\alpha}$is taken as δ^{th} highest value of$\{OI_e\}_{e=1}^{n-L'+1}$, where $\delta$is the nearest integer of $\alpha \times (n - L' + 1)$with confidence level α. Finally, the windows with outlier index higher than the threshold value are regarded as outliers. Despite the proficiencies of the methods discussed above, accuracy of each method highly varies with the value of its crucial parameter(s), hence, optimization of crucial parameter(s) is of utmost importance. The flowchart for *k*NN-based outlier detection is given in Fig. 4.

The accuracy of each of the methods explained above depends upon specific crucial parameters, whose significance is elucidated in Table 3.

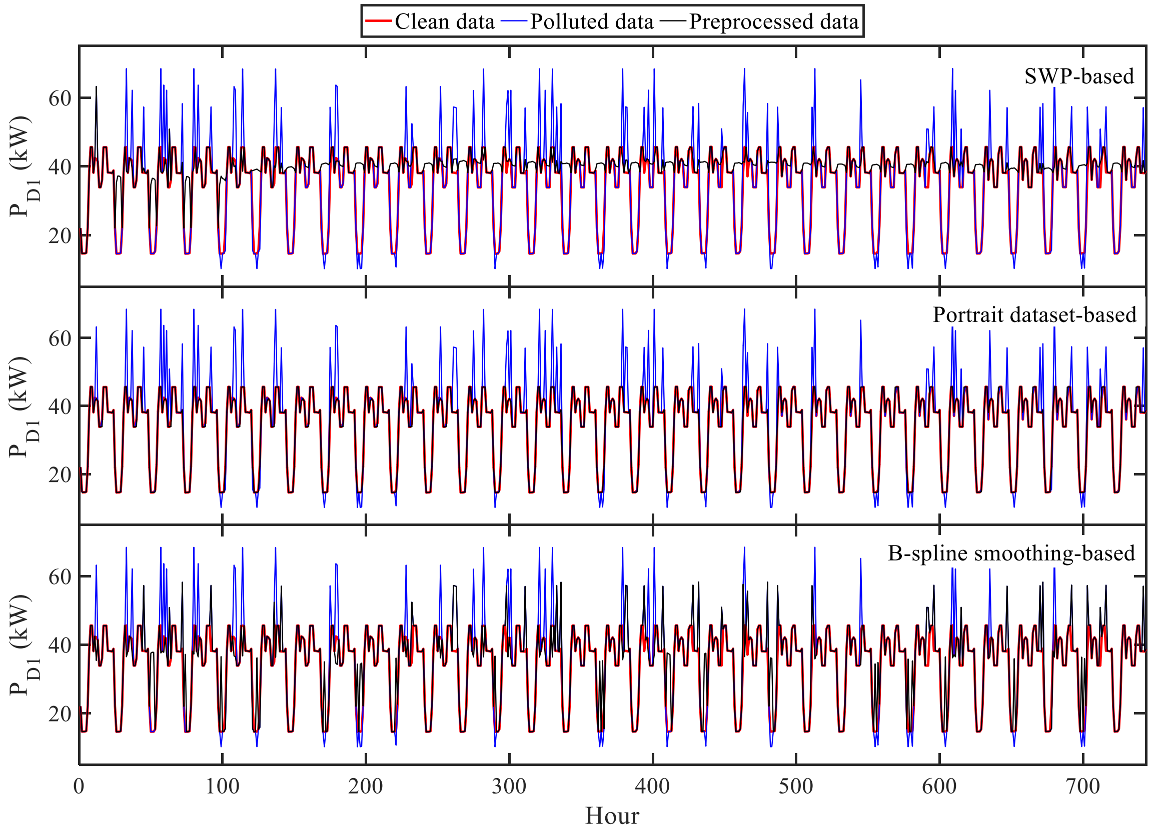


Fig. 7. Outlier correction performance comparison when applied to polluted data.

In addition to the parameters interpreted in the table, the performance of the selective well-established methods that are only applicable to univariate data is affected by the confidence coefficient. The confidence coefficient partly influences the confidence interval, which is the range in which a data point is treated normal/genuine.

4. Basis for comparison of preprocessing methods

The preprocessing methods existing in the literature differ from each other in their preprocessing capabilities. Therefore, the basis for comparing these methods should be different. In this paper, separate measures are discussed to compare the outlier detection and outlier correction capabilities of the different preprocessing methods. Two distinct data type assumptions are considered: clean data and raw data. The former assumes that the dataset is free from outliers, while the latter, mostly the case, does not make such an assumption.

4.1. Basis for comparison of methods' outlier detection capability

4.1.1. For clean data

This paper follows the general standard for comparison amidst methods based on their outlier detection capability, as frequently used by many authors [38,47]. Firstly, for this kind of data, outliers are introduced manually, and then the method is applied to the data, and its performance is analyzed. Three indicators used are as defined under:

a) True positive (TP), i.e., the data points identified correctly as outliers by the method.
b) False positive (FP), i.e., the genuine data points marked as outliers by the method.
c) False negative (FN), i.e., the outliers that are not identified by the method.

Based on these indicators, three bases for comparison are defined as follows:

1) Precision (P): It is the percentage of outliers detected correctly, out of the total number of data points marked as outliers by the method. The higher the P-value of a method better is its performance over other methods.
2) Recall (R): It is the percentage of outliers detected correctly by the method with respect to the total number of outliers present in the dataset. The method with the highest recall value is considered best.
3) F-measure (F): It the harmonic mean of precision and recall. Its significance lies in the comparison of methods when two methods have contradictory P and R values. For example, let 'A' and 'B' be the two preprocessing methods. 'A' has higher precision than 'B', whereas 'B' has higher recall than 'A.' In this case, the method with higher F-measure is considered better.

4.1.2. For raw data

In a practical scenario, prior knowledge about the outliers is not available mostly. Therefore, when applied to clean data, the measures used for comparing the methods cannot be applied to compare the methods when applied to raw data. Hence, an index-based comparison of methods (as in Section 4.1.1) is not possible. Further, these indices give only a numeric comparison among the methods and ultimately fail to highlight the reason for a method's failure or success. The knowledge of these reasons could guide a novice researcher in selecting an appropriate method. Therefore, when applied to such data, the effectiveness of a method is solely judged by analyzing its approach and data-specific performance. Also, as the information about outliers is not known a priori, the only way to verify whether a data point is a true outlier or not is to check the near-by data points and see whether it follows similar characteristics.

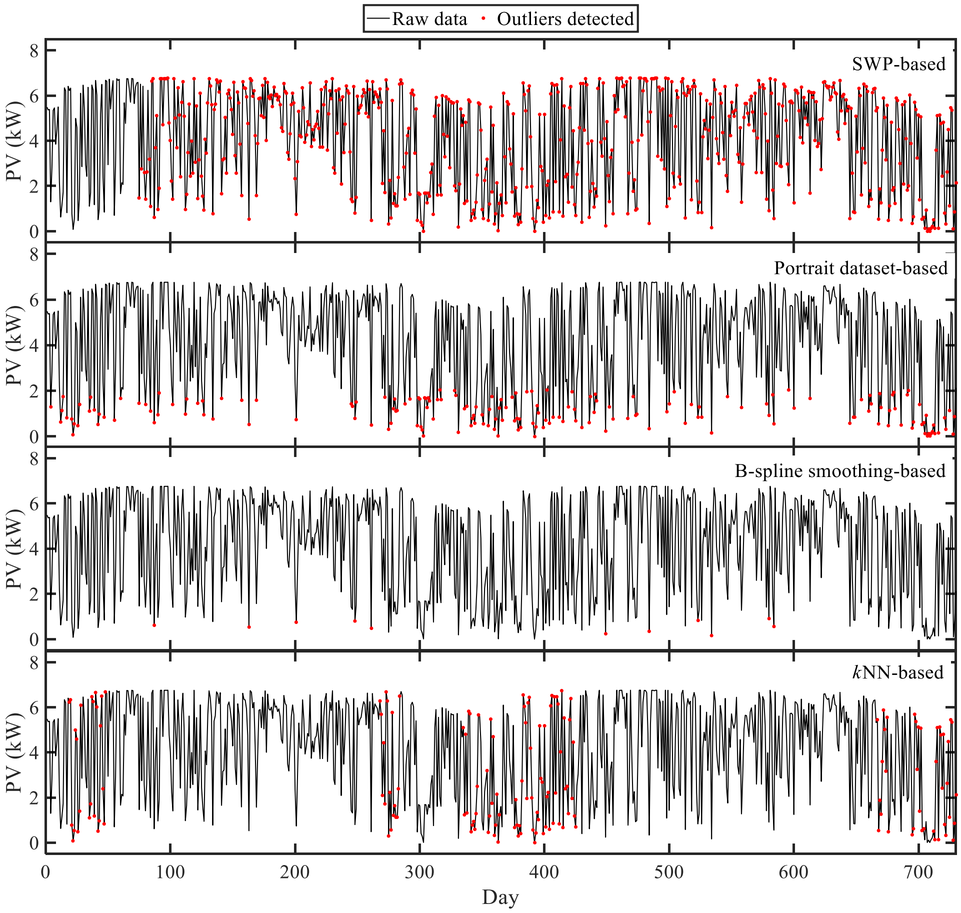


Fig. 8. Comparison of outlier detection capability when applied to PV generation time-series.

4.2. Basis for comparison of methods' outlier correction capability

Amongst the vast number of preprocessing methods, few can correct the detected outliers. Therefore, there is no such generalized standard to compare the outlier correction capability of a method. Hence, this paper uses a graphical approach to examine the outlier correction capability of different applied methods. According to the graphical approach, the data points identified as outliers, after correction, should follow the trend and seasonality of the dataset. The well-established preprocessing methods are compared in the subsequent section using these measures as the basis for comparison.

5. Result analysis

5.1. Datasets

The PV generation data is collected from a rooftop PV installation in the USA [94]. The load power data are collected from a restaurant in Anchorage, USA [95], and the North-Central weather zone of Texas, USA [96]. The ambient temperature data is collected from Berhampur, India [97]. A detailed description of time information and the sample size is elaborated in Table 4. The various cases in Table 4 are used in the result analysis.

5.2. Preprocessing performance comparison when applied to clean data

The objective of the analysis in this section is to compare the well-established methods' performance when applied to clean data. Firstly, the analysis is made for well-established methods applicable to univariate time-series (clean data, i.e., Case 2 of Table 4). The data is shown in Fig. 5. Observing the data in Fig. 5, it is evident that the data is clean (free from outliers). Therefore, for performance comparison using the indices defined in Section 4.1.1, 100 outliers are introduced in the data, and are now termed "polluted data." This is done by manipulating 100 randomly selected data points. A data point's value is increased by 50% if it is higher than the mean value; else, the data point's value is decreased by 30%. The polluted data is also shown in Fig. 5.

The well-established methods applicable to univariate time-series, as elucidated in Section 3, are applied to polluted data, and their outlier detection performance is compared in Fig. 6 and Table 5. Each method is tested with a range of significant parameter values, and the set of parameters' values is chosen at which the performance of the method is optimum. The comparison results show that the B-spline smoothing-based approach performs better than all other methods. It detects comparatively less number of true outliers but also detects only a few false outliers because of the sensitivity of the fitted curve towards the outliers present in the data. As all the data points, including outliers, are used for fitting a curve, under-fitting/over-fitting is the major drawback of this approach. The outlier detection and correction performances, dependent on the fitted curve, are affected due to under-fitting/over-fitting. Whereas, portrait dataset-based detects all the outliers introduced in the data and many false outliers. The method marks many genuine data points as outliers. It is based on the assumption that data within a portrait dataset are similar, which is generally not the case in volatile data. The assumption leads to the detection of genuine data points within a portrait dataset as outliers. The SWP-based approach gives comparable results to that of a portrait dataset-based approach but marks a higher number of genuine data points as outliers. The method fails because of its outlier correction approach. The SWP-based approach uses a type of AR model for predicting the value of data point

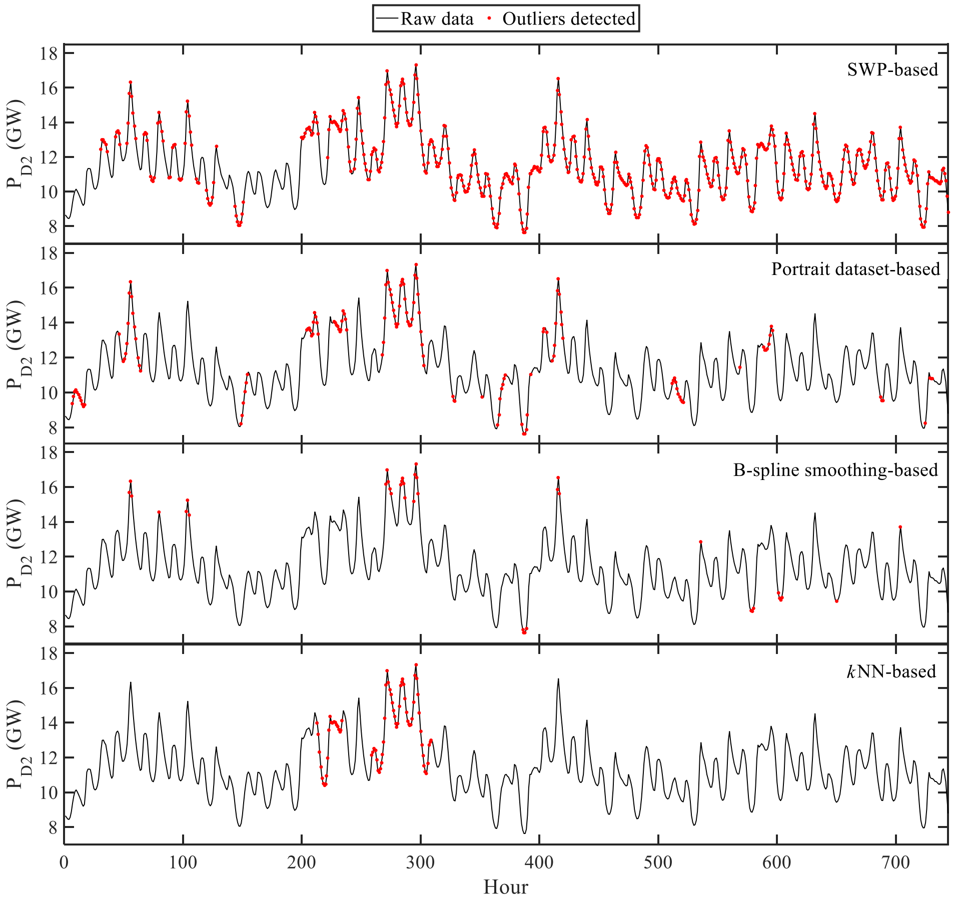


Fig. 9. Comparison of outlier detection capability when applied to load power time-series.

at every time instant, which is only applicable to stationary data. The inaccurate prediction leads to calculating an inappropriate confidence interval, which results in the detection of a genuine data point as an outlier, is replaced by the incorrect predicted value. The data point is further used for predicting the next data points. The error propagates, and after a certain time-instant, the method marks all the data points as outliers [98].

The *k*NN-based approach performs below average. The accuracy of the method decreases with an increase in volatility effect [98]. It assumes that windows containing outliers are distinct from the rest, which is not the case in volatile data. Further, this approach cannot detect outliers at particular time-instants; instead, it marks a bunch of data points (window which contains outliers) as outliers. This results in the detection of a high number of false outliers.

The outlier correction performance is compared in Fig. 7.

It is evident from Fig. 7 that the portrait dataset-based approach follows the trend of previous comparison results (shown in Fig. 6 and Table 5) and performs better than other methods. The preprocessed data deviates least from the clean data and follows the periodic pattern of the clean data, even after detecting a high number of false outliers, because the clean data is less volatile, and the data within a portrait dataset are similar. The performance of the portrait dataset-based approach is different when the method is applied to highly volatile data. The outliers, after correction using a B-spline smoothing-based approach, also follow the clean data. But, because of this approach's inability to detect all the outliers, the overall preprocessed data deviates from the clean data. Whereas the SWP-based approach completely fails to preprocess the data, and the data after preprocessing is unable to follow the periodic pattern. From the above outlier detection and correction comparison results, it is perceptible that the portrait dataset-based approach outperforms the other methods when the data is less volatile and clean.

All the analyses in this section are done considering less volatile clean data. But, the performance of the well-established methods varies when they are applied to more volatile raw data. This analysis is done in the subsequent section.

5.3. Preprocessing performance comparison when applied to raw data

As the objective of this section is to analyze the performances of the well-established methods when applied to volatile raw data, Cases 1, 3, and 5 of Table 4 are used for the analyses. Since the information about outliers is not known a priori in raw data, an index-based comparison is not possible. The methods are applied to various time-series, having different time-resolution, and their performances are compared by observing the graphical results. The outlier detection capability of well-established methods when applied to different raw data, are elucidated in Figs. 8, 9, and 10.

From the application of different well-established methods on various types of datasets in Figs. 8, 9, and 10, it is evident that the B-spline smoothing-based approach performs better than other methods. Although it detects a smaller number of true outliers, it marks a very less number of genuine data points as outliers. Whereas, portrait dataset-based and *k*NN-based approaches detect a large number of false outliers along with the true outliers. The application of SWP-based approach to any type of data fails after a certain time-instant and marks all the data points as outliers. The reason for such performance of the methods is already explained in detail in Section 5.2.

To compare the outlier correction capability of well-established methods when applied to volatile raw data, they are applied to various

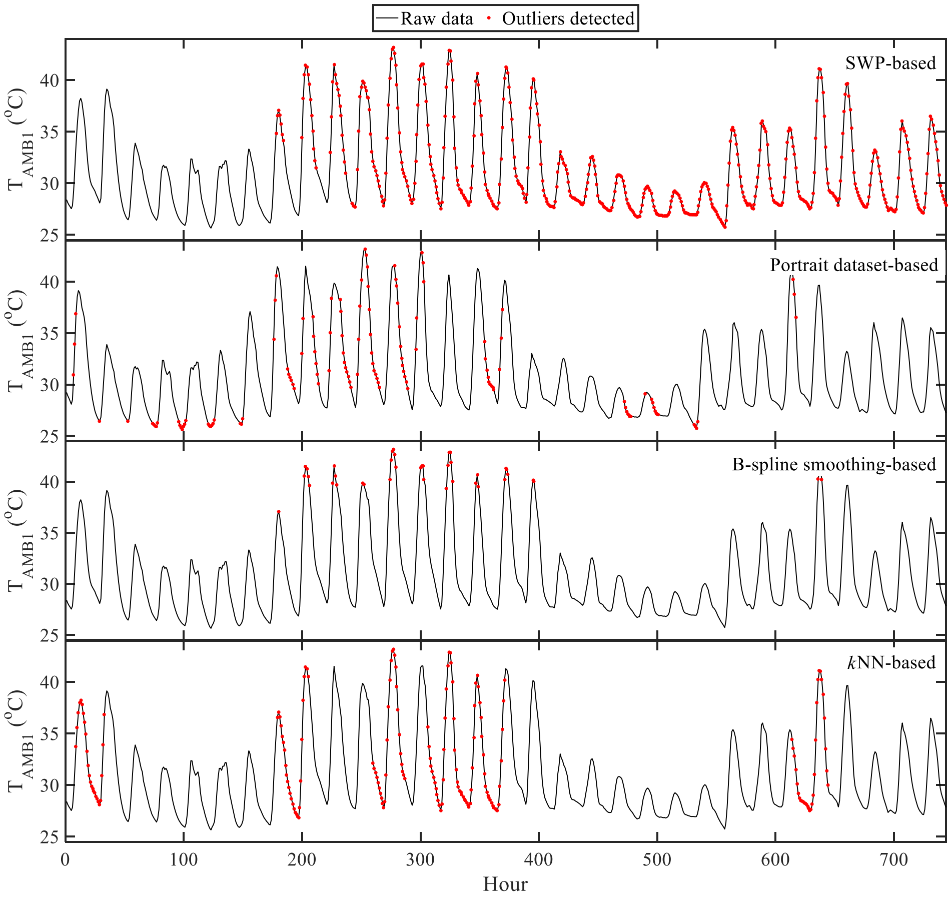


Fig. 10. Comparison of outlier detection capability when applied to ambient temperature data.

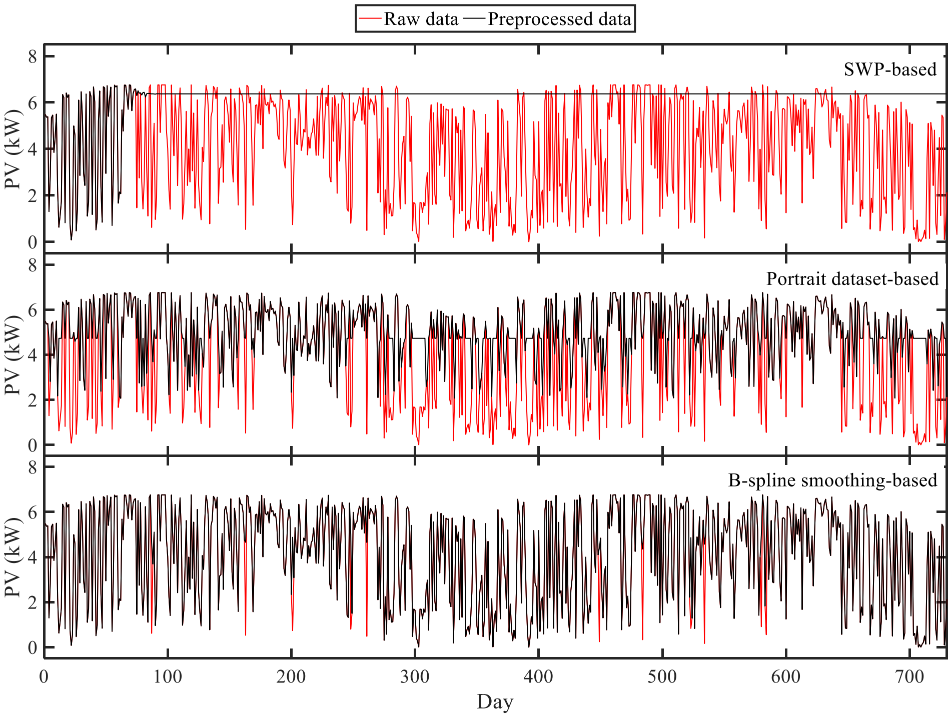


Fig. 11. Comparison of outlier correction capability when applied to PV generation data.

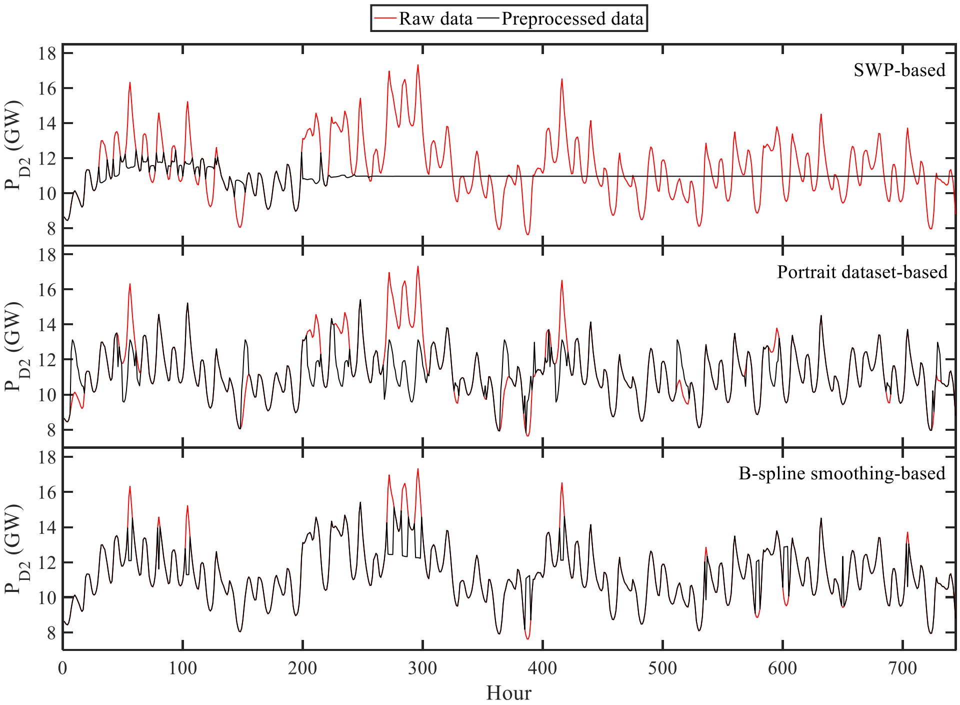


Fig. 12. Comparison of outlier correction capability when applied to data in load power data.

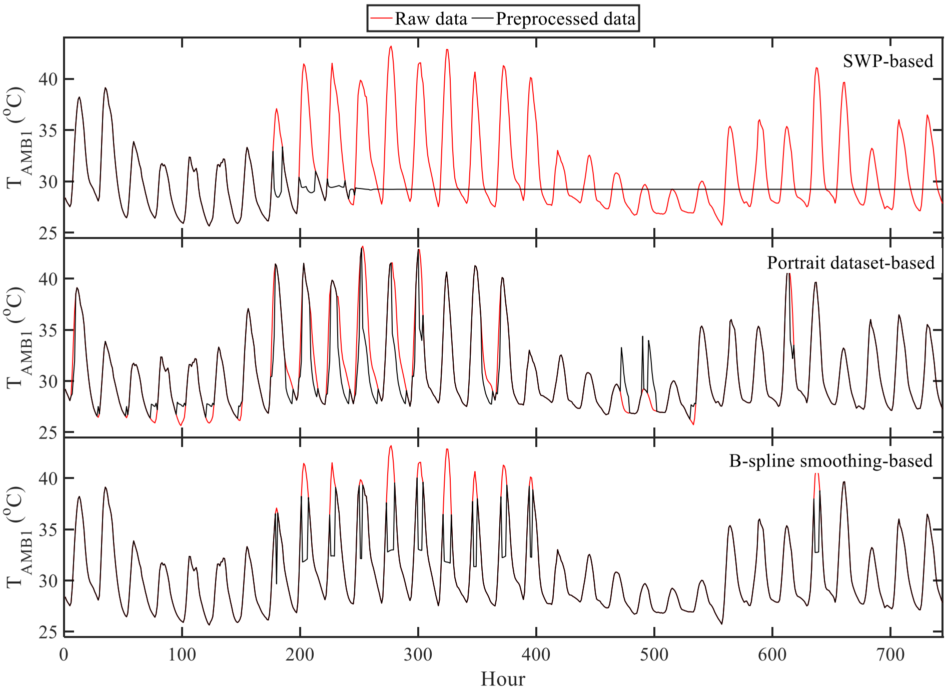


Fig. 13. Comparison of outlier correction capability when applied to ambient temperature data.

raw data, and the results are analyzed in Figs. 11, 12, and 13.

It is clear from the above analysis that B-spline smoothing-based approach outperforms other methods when applied to volatile raw data. The data after preprocessing follows the trend and seasonal pattern of the raw data. The benefit of using this approach is that the method is dependent on the fitted curve, which can be controlled by the user. Whereas, portrait dataset-based approach is unable to keep the characteristics of raw data intact after preprocessing. The main reason for the approach's failure is the assumption that the data within a portrait dataset are stable. But, in volatile data, data within a portrait dataset have trend and periodic patterns. Due to this, many false outliers are detected, and the data after correction is unable to trace the trend and periodic patterns of the raw data. The SWP-based approach fails after a certain time-instant due to the propagation of prediction error, which affects the detection and correction of outliers. Therefore, the data after preprocessing becomes constant after a certain time-instant.

From the detailed analyses in Sections 5.2 and 5.3, the following important points are worth noting. The SWP-based approach cannot preprocess any volatile data, as it fails after a certain time-instant in all kinds of data. The portrait dataset-based method performs well in case

Table 6
Effect on window width on performance of SWP-based approach.

Indices	Window width (2k′)		
	$2k' = 12$	$2k' = 16$	$2k' = 20$
No. of TP	99	99	98
No. of FP	613	234	208
No. of FN	1	1	2
P	0.1390	0.2973	0.3203
R	0.99	0.99	0.98
F-measure	0.2438	0.4573	0.4828

Table 7
Effect of VPDs on performance of portrait dataset-based approach.

Indices	6 VPDs	12 VPDs	24 VPDs
No. of TP	84	69	100
No. of FP	130	61	133
No. of FN	16	31	0
P	0.3925	0.5307	0.4292
R	0.84	0.69	1
F-measure	0.5350	0.5999	0.6006

of less volatile data, where the data within a portrait dataset are similar. Still, the accuracy decreases when applied to highly volatile data. The B-spline smoothing-based method is better than other methods in all aspects, irrespective of the data is raw or volatile. The *k*NN-based method is also not advisable for any volatile data. Because it assumes the window containing outliers, it detects many false outliers, and it does only half preprocessing as it is not capable of correcting outliers.

The crucial parameter(s), time-resolution, and volatility of the data significantly impact the outlier detection and correction performance of the well-established methods. These effects are elaborated in Section 5.4.

5.4. Impact of change of crucial parameter(s) values, and time-resolution on preprocessing

The performances of the well-established methods are discussed in Section 5.2 and 5.3. The reasons for their observed performance and the parameter effect are explained in detail in this section.

5.4.1. Sliding window prediction-based approach

The effect of crucial parameters on the SWP-based approach's performance is inferred, considering Case 2 of Table 4. The method is applied to the data by selecting different window width values, and the results are shown in Table 6. It is evident from the table that window width has a significant impact on the method's performance. The reason lies in the model that is used by the SWP-based approach. It uses a type of AR model for predicting the value of data points at each instant, and the window width is the order of the model. Therefore, an inappropriate window width establishes a wrong relationship among the data, predicting an inappropriate value of data at each time instant. Due to inaccurate prediction, a wrong confidence interval is computed, which results in the detection of genuine data points as outliers. The inappropriate predicted values replace the detected outliers, and the preprocessed data is unable to follow the trend and seasonal patterns of the original data. This phenomenon becomes more prominent in volatile data, as evident from the results of Sections 5.2 and 5.3.

5.4.2. Portrait dataset-based approach

The effect of VPDs on the method's performance is demonstrated using Case 2 of Table 4. The method is applied to data by merging similar portrait datasets in different numbers of VPDs (randomly selected), and the results are compared in Table 7. It is inferred from Table 7 that the number of VPDs has a significant effect on the performance of the method. As the method assumes the data within a portrait dataset to be stable, a similar portrait dataset should be merged for the method's optimum performance. But the method detects a large number of false outliers in any data; the data within a portrait dataset are not stable in case of volatile data. Its correction capability is better than other methods when applied to less volatile data. However, its application to volatile data, as evident from Figs. 11 and 12 indicate that the preprocessed data cannot follow the characteristics of the raw data. The method works well only with a stable (non-volatile) portrait dataset, which is generally not the case.

5.4.3. B-spline smoothing-based approach

This method works better than other methods in most of the cases. The method is solely dependent on the fitted curve. The fitted curve results from the basis functions used, which is in the user's hand. The only lacuna this method faces is the sensitivity of the fitted curve towards the outliers present in the data. Case 4 of Table 4 is considered for the analysis to demonstrate this sensitivity. The method is applied to the data, and then ten outliers are introduced manually from day 96–105. The method is applied to the data again, and the results are compared in Fig. 14. It is evident from the figure that the fitted B-spline curve is sensitive towards the outliers present in the data as the method does not detect all the outliers when applied to manipulated data, as detected before on raw data. The inclusion of all the data points (outliers and genuine data points) for fitting the curve to the data is the root cause of this issue. The shape of the curve changes to fit the curve to all the data points. Due to the change in the curve's shape, some outliers go

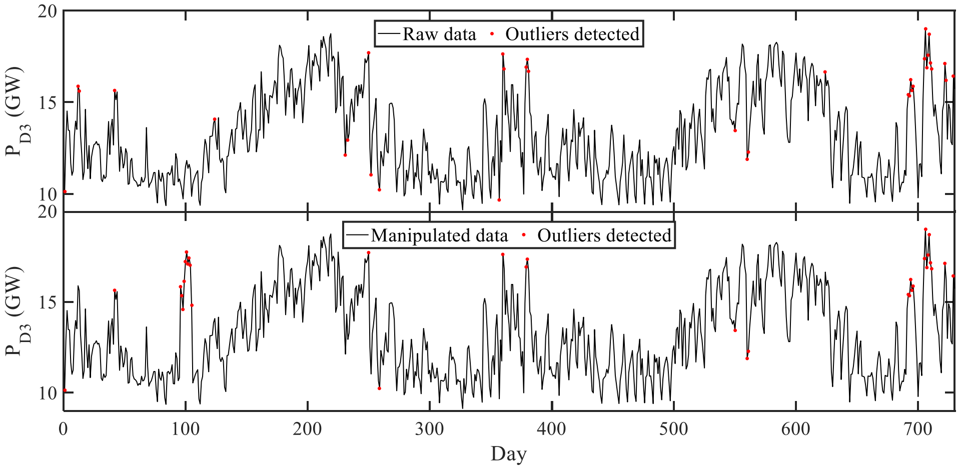


Fig. 14. Sensitivity of the fitted B-spline curve towards the outliers present in the data.

Table 8.
Effect of basis functions on the performance of B-spline smoothing-based approach.

Indices	Basis functions			
	Fourier terms		Random function	
	Sensible candidate set	Random selection	Cubic	Quadratic
No. of TP	56	16	39	30
No. of FP	7	0	41	50
No. of FN	44	84	61	70
P	0.8889	1	0.4875	0.325
R	0.56	0.16	0.39	0.30
F-measure	0.6871	0.2759	0.4333	0.3333

Table 9
Effect of crucial parameters on the performance of *k*NN-based approach.

Indices	$k = 3$			$k = 5$		
	$L' = 12$	$L' = 16$	$L' = 24$	$L' = 12$	$L' = 16$	$L' = 24$
No. of TP	35	33	31	31	34	27
No. of FP	113	118	128	106	121	113
No. of FN	65	67	69	69	66	73
P	0.2365	0.2185	0.195	0.2263	0.2194	0.1929
R	0.35	0.33	0.31	0.31	0.34	0.27
F-measure	0.2823	0.2629	0.2394	0.2616	0.2667	0.2250

Table 10
Effect of L1 and L2 norms on the performance of *k*NN-based approach when applied to polluted data.

Indices	L1 norm	L2 norm
No. of TP	39	35
No. of FP	108	111
No. of FN	61	65
P	0.2653	0.2397
R	0.39	0.35
F-measure	0.3158	0.2846

Table 11.
Review on the applicability of preprocessing methods when applied to volatile time-series.

Confidence coefficient	Indices	SWP-based	Portrait dataset-based	B-spline smoothing-based	*k*NN-based
0.05	No. of TP	98	100	23	35
	No. of FP	208	133	0	111
	No. of FN	2	0	77	65
	P	0.3203	0.4292	1	0.2397
	R	0.98	1	0.23	0.35
	F-measure	0.4828	0.6006	0.374	0.2846
0.10	No. of TP	99	100	56	50
	No. of FP	229	143	7	187
	No. of FN	1	0	44	50
	P	0.3018	0.4115	0.8889	0.2110
	R	0.99	1	0.56	0.5
	F-measure	0.4626	0.5831	0.6871	0.2967
0.20	No. of TP	99	100	92	58
	No. of FP	348	180	102	270
	No. of FN	1	0	8	42
	P	0.2215	0.3571	0.4742	0.1768
	R	0.99	1	0.92	0.58
	F-measure	0.3620	0.5263	0.6259	0.2710

undetected, as outlier detection is solely based on the fitted curve. However, the outlier correction capability is better than other methods, regardless of the data's volatility effect.

Besides the sensitivity of the fitted curve towards the outliers present in the data, the method's performance also depends on the basis functions used for fitting the curve. Case 2 of Table 4 is chosen to show this dependency, and the method is applied using different basis functions. As the data is periodic (like other data used in the result analyses), the method as proposed in [7] is applied to each week of the considered time-series to select a sensible candidate set of frequency components for capturing the seasonalities. The method is first implemented using Fourier terms corresponding to the dominant frequency components (weekly and daily). Later, it is implemented using Fourier terms corresponding to the frequency components selected randomly (weekly), and using random functions as basis functions, and the results are compared in Table 8.

It is evident from Table 8 that the performance is sensitive towards the basis functions used for fitting a curve. The selection of any random function/ frequency component does yield good results. Hence, for any particular periodic data, dominant frequency components should be revealed to fit a curve that captures the seasonalities of the data and get good preprocessing results.

5.4.4. *k-nearest neighbor-based approach*

Case 2 of Table 4 is considered for the analysis, and the method is applied to the data by setting different values of the crucial parameters. The effect of crucial parameters on the performance of the method is elucidated in Table 9. In addition to that, to illustrate the effect of the norm of the method's performance, it is implemented using L1 norm instead of the conventional L2 norm. The comparison results are shown in Table 10.

It is inferred from Table 9 that the parameter k does not have a significant role in determining the performance. In contrast, the window length (L') has a substantial impact on the performance of the method. Table 10 depicts that the method's performance is slightly elevated by using the L1 norm to calculate the distance among widows, as the L1 norm is more robust compared to the L2 norm. As the L2 norm is used in the base papers, the method is implemented using the same throughout the result analyses. Also, the method is unable to detect outliers at a particular time instant. It marks a group of data as outliers, including genuine data points, as evident from Sections 5.2 and 5.3. In addition to this, the method's incapability to correct outliers makes it the least preferred method out of the other well-established methods.

5.4.5. *Effect of confidence coefficient*

The performance of the well-established methods is affected by the confidence coefficient. Case 2 of Table 4 is considered to show this effect, and the methods are applied to the data by setting different values of confidence coefficient. The results are compared in Table 11.

It is evident from Table 11 that the confidence coefficient influences the performance of the respective well-established as well as the proposed method. In other words, the range in which a data point is treated as normal/genuine, i.e., the confidence interval, is partly dependent on the confidence coefficient. In addition to the confidence coefficient, the confidence interval depends on other method-specific factors (for example, in SWP-based approach, the confidence interval for a data point also depends on its predicted value, the standard deviation of its neighborhood, and the chosen window width), which are explained in Section 3.2. Therefore, the degree with which the confidence coefficient affects the performance of a method varies from method-to-method. An inappropriate confidence coefficient results in inaccurate confidence intervals, which further decreases the number of true outliers and increases the number of false outliers, being detected. Hence, a proper confidence coefficient must be chosen for the optimum performance of a method, which builds a point-wise confidence interval to detect outliers.

In addition to the crucial parameter(s), the well-established methods are also sensitive to the data's volatility effect, which is explained in detail in Table 12.

Table 12
Review on the applicability of well-established methods when applied to volatile time-series.

Methods	Sensitivity of method to volatility effect
SWP-based	The outlier correction approach fails when applied to volatile data. The error propagates, leading to the complete failure of preprocessing after a certain period. Hence, this method does not apply to volatile time-series.
Portrait data-based	Assumes that data within a portrait dataset are stable. Hence, this method fails when applied to volatile time-series as the variation within the portrait dataset increases.
B-spline smoothing-based	The method applies to volatile data, as long as the curve is adequately fitted using a suitable basis function. The performance of the method does not degrade with an increase in the volatility effect of the data.
*k*NN-based	Assumes that the window containing outlier(s) will be distinct from all other windows. Hence, this method fails when applied to volatile time-series as genuine windows are very different from each other in a volatile time-series.

Table 13
Merits, lacunae and scope for improvisation of well-established methods.

SWP-based	Merits	• Does not use any statistical parameter that is affected by the presence of outliers. • The non-volatile time-series does not lose its inherent trend and seasonal pattern after preprocessing.
	Lacunae	• Fails to preprocess a volatile time-series. • No generalized rules suggested for the optimization the crucial parameter.
	Scope	• Way for optimal selection of window width is needed. • Better outlier correction approach is required.
Portrait dataset-based	Merits	• Does not use any statistical parameter that is affected by the presence of outliers. • Capable of visualizing the hidden characteristics like trend and seasonality. • The non-volatile time-series does not lose its inherent trend and seasonal pattern after preprocessing.
	Lacunae	• Uses statistical parameters that are affected by the presence of outliers. • Decrease in accuracy with increase in volatility of time-series. • Unclear algorithm for merging similar portrait datasets. • Outlier correction approach fails with the increase in variation within the portrait datasets.
	Scope	• Algorithm for merging similar portrait dataset. • Better outlier correction approach is required in case of volatile time-series.
B-spline smoothing-based	Merits	• Does not use any statistical parameter that is affected by the presence of outliers. • The non-volatile as well as volatile time-series does not loses its inherent trend and seasonal pattern after preprocessing.
	Lacunae	• Selection of appropriate basis functions and the number of basis functions to be used are the major concerns. • Under-fitting and over-fitting leads to respectively the detection of false outliers, overlooking a genuine outlier. • Time taking process and high computational power is needed for its application on large data. • Since non-parametric regression model is used, which does not consider pre-defined structural form of the data, the fitted curve is sensitive to the presence of outliers.
	Scope	• Sensitivity of the fitted curve towards the outliers present in the data need to be reduced.
*k*NN-based	Merits	• Does not require prior knowledge of the type of distribution a time-series exhibits. • Capable of detecting outliers in a non-volatile time-series.
	Lacunae	• Unable to correct outliers, hence, does only half the work. • Cannot detect outliers at a particular time instant. • Outlier detection algorithm fails when applied to volatile time-series. • No generalized rules suggested optimizing the crucial parameters.
	Scope	• Outlier correction approach is needed. • Way to detect outliers at a particular time-instant need to be found.

5.5. Merits, lacunae and scopes for improvisation

The merits, lacunae, and scopes for improvisation for the well-established outlier detection and correction approaches are discussed in Table 13.

6. Conclusion

A rigorous survey of research in data preprocessing is carried out by categorizing numerous preprocessing methods. The concept of an outlier, the preprocessing applications, and the major categories of preprocessing methods are elaborated understandably. Many existing preprocessing methods are analyzed, categorized, and the preprocessing capability of each technique is highlighted. The well-established techniques of each category are elaborated with a detailed explanation of their algorithm. Two different types of data, i.e., the less volatile clean data and the highly volatile raw data, are considered for the result analyses. Different bases are defined for comparing the well-established methods when applied to clean data and raw data. The well-established methods are applied to univariate (clean as well as raw) time-series with different time-resolutions. The performances of the well-established techniques are critically analyzed based on their outlier detection and outlier correction capability. The impact of change in crucial parameter(s) values and time-resolution of data on the selective methods' performance is also elucidated. Finally, the merits, lacunae, and scope for improvisation of the well-established techniques are highlighted.

From the detailed analyses of the different well-established methods when applied to the less volatile clean data and the more volatile raw data, the significant inferences are as follows:

1) The SWP-based approach performs well with less volatile clean data as compared to the more volatile raw data. It has a lower precision of 0.3203 but a comparatively high recall of 0.98. Due to its dependence on the AR type prediction model, it fails when applied to highly volatile time-series. Its preprocessing capability is highly dependent on the window width and the confidence coefficient, as evident from the analyses. Hence, an optimal selection of window width along with a better outlier correction technique is needed.
2) The portrait dataset-based approach has a decent performance when applied to less volatile data. Despite many false positives, it is found to have perfect recall of 1 in the considered case-2. However, it cannot retrace the patterns in the data after outlier correction, when applied to the more volatile raw data. It is affected by the number of VPDs. Due to the assumption of stable data within a portrait dataset; similar portraits need to be merged for optimum performance. A

proper algorithm for this is essential. Also, improvement is required in the outlier correction technique with volatile data.

3) The B-spline smoothing approach is found to perform relatively better than the other well-established methods on clean and raw data. It has the highest F-measure of 0.6871, among all the methods compared, which indicates its better performance. It gives the least number of false positives, contrary to the portrait dataset-based and *k*NN-based approaches. This method is sensitive to the outliers present in the data and the basis function used for fitting the curve. It sometimes results in under-/over-fitting of the non-parametric regression curve, leading to false detections.
4) The *k*NN-based approach can only detect outliers and cannot correct them. Among the different well-established methods, it has the worst performance when applied to the clean data, as indicated by the lowest values of precision, recall, and F-measure, i.e., 0.2110, 0.5, and 0.2967 respectively. It detects the lowest number of true outliers with many false positives. Due to its sensitivity towards the window length and distance norm, it does not perform well and is not advisable for volatile data. It needs algorithms for outlier detection at specific time-instants and also for outlier correction.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A. Bracale, P. Caramia, G. Carpinelli, A.R. Di Fazio, P Varilone, A bayesian-based approach for a short-term steady-state forecast of a smart grid, IEEE Trans. Smart Grid 4 (4) (2013) 1760–1771.
[2] B.R. Prusty, D Jena, An over-limit risk assessment of PV integrated power system using probabilistic load flow based on multi-time instant uncertainty modeling, Renew. Energy 116 (2018) 367–383.
[3] L. Zheng, W. Hu, Y Min, Raw wind data preprocessing: a data-mining approach, IEEE Trans. Sustain. Energy 6 (1) (2015) 11–19.
[4] J. An, Z. Bie, X. Chen, B. Hua, S Liu, A generalized data preprocessing method for wind power prediction, 2013 IEEE Power & Energy Society General Meeting, 2013, pp. 1–5.
[5] J.Y. Wang, Z. Qian, H. Zareipour, D Wood, Performance assessment of photovoltaic modules based on daily energy generation estimation, Energy 165 (2018) 1160–1172.
[6] B.R. Prusty, D Jena, Uncertainty Modeling Steps for Probabilistic Steady-State Analysis, Applications of Computing, Automation and Wireless Systems in Electrical Engineering, Springer, Singapore, 2019, pp. 1169–1177.
[7] B.R. Prusty, D. Jena, A spatiotemporal probabilistic model-based temperature-augmented probabilistic load flow considering PV generations, Int. Trans. Electric. Energy Syst. 29 (5) (2019) e2819 https://doi.org/10.1002/2050-7038.2819.
[8] J.S. Park, C.G. Park, K.E Lee, Simultaneous outlier detection and variable selection via difference-based regression model and stochastic search variable selection, Commun. Stat. Appl. Methods 26 (2) (2019) 149–161.
[9] D.D. Prastyo, F.S. Nabila, M.H. Lee, N. Suhermi, S.F Fam, VAR and GSTAR-based feature selection in support vector regression for multivariate spatio-temporal forecasting, International Conference on Soft Computing in Data Science, Singapore, Springer, 2018 Aug 15, pp. 46–57.
[10] J.L. Speiser, M.E. Miller, J. Tooze, E Ip, A comparison of random forest variable selection methods for classification prediction modeling, Expert Syst. Appl. 134 (2019) 93–101.
[11] Y. Jiang, Y. Wang, J. Zhang, B. Xie, J. Liao, W Liao, Outlier detection and robust variable selection via the penalized weighted LAD-LASSO method, J. Appl. Stat. (2020) 1–13.
[12] HawkinsD.M.Identification of outliers. Monographs on Applied Probability and Statistics ;1980.
[13] E.Q McCallum, Bad Data handbook: Mapping the World of Data Problems, O'Reilly Media, Sebastopol, CA, 2012.
[14] H. Wang, M.J. Bah, M. Hammad, Progress in outlier detection techniques: a survey, IEEE Access 7 (2019) 107964–108000.
[15] X. Dong, L. Qian, L Huang, A CNN based bagging learning approach to short-term load forecasting in smart grid, IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), San Francisco, CA, 2017, pp. 1–6.
[16] S. Ghosh, L Reilly D, Credit card fraud detection with a neural-network, Proceedings of the 27th Annual Hawaii International Conference on System Science, Los Alamitos, CA, 1994, p. 3.
[17] D. Dasgupta, N. Majumdar, Outlier detection in multidimensional data using negative selection algorithm, Proceedings of the IEEE Conference on Evolutionary Computation, Hawaii, 2002, pp. 1039–1044.
[18] S. Ghanbari, A.B. Hashemi, C Amza, Stage-aware anomaly detection through tracking log points, Proceedings of the 15th International Middleware Conference, 2014, pp. 253–264.
[19] H.J Escalante, A comparison of outlier detection algorithms for machine learning, Proceedings of the International Conference on Communications in Computing, 2005, pp. 228–237.
[20] ZhangJ.Advancements of outlier detection: a survey. ICST Trans. Scalable Inform. Syst.2013;13(1):1–26.
[21] D. Dave, T Varma, A review of various statistical methods for outlier detection, Int. J. Comput. Sci. Eng. Technol. 5 (2) (2014) 137–140.
[22] BhosaleS.V. A Survey: outlier Detection in Streaming Data Using Clustering Approached. IJCSIT) Int. J. Comp. Sci. Inform. Tech.2014;5:6050–6053.
[23] V. Chandola, A. Banerjee, V Kumar, Anomaly detection: a survey, ACM Comput. Surv. (CSUR) 41 (3) (2009) 1–58.
[24] JohansenS., NielsenB.Outlier detection algorithms for least squares time series regression. Available at SSRN 2510281; 2014.
[25] M. Gupta, J. Gao, C.C. Aggarwal, J Han, Outlier detection for temporal data: a survey, IEEE Trans. Knowl. Data Eng. 26 (9) (2013) 2250–2267.
[26] Z.A. Bakar, R. Mohemad, A. Ahmad, M.M Deris, A comparative study for outlier detection techniques in data mining, 2006 IEEE conference on cybernetics and intelligent systems, 2006, pp. 1–6.
[27] M. Goldstein, S Uchida, A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data, PLoS ONE 11 (4) (2016) e0152173.
[28] P.J. Rousseeuw, I. Ruts, J.W Tukey, The bagplot: a bivariate boxplot, Am. Stat. 53 (4) (1999) 382–387.
[29] B. Abraham, A Chuang, Outlier detection and time series modeling, Technometrics 31 (2) (1989) 241–248.
[30] G.M. Ljung, On outlier detection in time series, J. R. Stat. Soc. Series B (Methodological) 55 (2) (1993) 559–567.
[31] B.G. Amidan, T.A. Ferryman, S.K Cooley, Data outlier detection using the Chebyshev theorem, 2005 IEEE Aerospace Conference, 2005, pp. 3814–3819.
[32] M Yue, An integrated anomaly detection method for load forecasting data under cyberattacks, 2017 IEEE Power & Energy Society General Meeting, 2017, pp. 1–5.
[33] P.M Broersen, ARMAsel for detection and correction of outliers in univariate stochastic data, IEEE Trans. Instrum. Meas. 57 (3) (2008) 446–453.
[34] HillD.J., MinskerB.S.Anomaly detection in streaming environmental sensor data: a data-driven modeling approach. Environ. Model. Softw.2010;25(9):1014–1022.
[35] YuY., ZhuY., LiS. Time series outlier detection based on sliding window prediction. Math. Probl. Eng.2014;1–14.
[36] L. Ma, X. Gu, B Wang, Correction of outliers in temperature time series based on sliding window prediction in meteorological sensor network, Information 8 (2) (2017) 60.
[37] V.K. Samparthi, H.K Verma, Outlier detection of data in wireless sensor networks using kernel density estimation, Int. J. Comput. Appl. 5 (7) (2010) 28–32.
[38] G. Tang, K. Wu, J. Lei, Z. Bi, J Tang, From landscape to portrait: a new approach for outlier detection in load curve data, IEEE Trans. Smart Grid 5 (4) (2014) 1764–1773.
[39] H.N. Akouemo, R.J Povinelli, Time series outlier detection and imputation, 2014 IEEE PES General Meeting| Conference & Exposition, 2014, pp. 1–5.
[40] YeX., LuZ., QiaoY., MinY., O'MalleyM. Identification and correction of outliers in wind farm time series power data. IEEE Trans. Pow. Syst.2016;31(6):4197–4205.
[41] H.N. Akouemo, R.J Povinelli, Probabilistic anomaly detection in natural gas time series data, Int. J. Forecast. 32 (3) (2016) 948–956.
[42] HarléF., ChatelainF., Gouy-PaillerC., AchardS. Bayesian model for multiple change-points detection in multivariate time series. IEEE Trans. Signal Process. 2016;64(16):4351–4362.
[43] AkouemoH.N., PovinelliR.J.Data improving in time series using ARX and ANN models. IEEE Trans. Pow. Syst.2017;32(5):3352–3359.
[44] HuberP.J.Robust estimation of a location parameter. Anna. Math. Stat. 1964;35:73101.
[45] P.J. Huber, E.M Ronchetti, Robust Statistics, Wiley, New York, 2009.
[46] J. Yang, J Stenzel, Historical load curve correction for short-term load forecasting, 2005 International Power Engineering Conference, 2005, pp. 1–40.
[47] J. Chen, W. Li, A. Lau, J. Cao, K Wang, Automated load curve data cleansing in power systems, IEEE Trans. Smart Grid 1 (2) (2010) 213–221.
[48] A. Zhang, W. Zhong, W Liu, Detection of outlier patterns in call records based on skeleton points, 2010 International Conference on Artificial Intelligence and Computational Intelligence, 2 2010, pp. 145–149.
[49] GuoZ., LiW., LauA., Inga-RojasT., WangK. Detecting X-outliers in load curve data in power systems. IEEE Trans. Pow. Syst.2011;27(2):875–884.
[50] MateosG., GiannakisG.B. Robust nonparametric regression via sparsity control with application to load curve data cleansing. IEEE Trans. Signal Process. 2011;60(4):1571–1584.
[51] L. Fang, M. Zhi-zhong, An online outlier detection method for process control time series, 2011 Chinese Control and Decision Conference (CCDC), 2011, pp. 3263–3267.
[52] K.K.L.B. Adikaram, M.A. Hussein, M. Effenberger, T Becker, Outlier detection method in linear regression based on sum of arithmetic progression, Sci. World J. (2014).
[53] L.U.O. Jian, H.O.N.G. Tao, Y.U.E Meng, Real-time anomaly detection for very short-term load forecasting, J. Mod. Pow. Syst. Clean Energy 6 (2) (2018) 235–243.
[54] C. Jeenanunta, K.D. Abeyrathna, M.S. Dilhani, S.W. Hnin, P.P Phyo, Time series

outlier detection for short-term electricity load demand forecasting, Int. Sci. J. Eng. Tech. (ISJET) 2 (1) (2018) 37–50.

[55] Y. Lin, J Wang, Probabilistic deep autoencoder for power system measurement outlier detection and reconstruction, IEEE Trans. Smart Grid 11 (2) (2020) 1796–1798.

[56] M. Breunig, H. Kriegel, R. Ng, J Sander, LOF: identifying density based local outliers, Proceedings of the 2000 ACM SIGMOD International Conference on management of data, Dallas, 29 2000, pp. 93–104.

[57] J. Tang, Z. Chen, A.W.C. Fu, D.W Cheung, Enhancing effectiveness of outlier detections for low density patterns, Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2002, pp. 535–548.

[58] S. Papadimitriou, H. Kitagawa, P.B. Gibbons, C Faloutsos, LOCI: fast outlier detection using the local correlation integral, Proceedings of the 19th International Conference on Data Engineering, Los Alamitos, USA, IEEE Computer Society Press, 2003, pp. 315–326.

[59] W. Jin, A.K.H. Tung, J. Han, W Wang, Ranking outliers using symmetric neighborhood relationship, Proceedings of the 10th Pacific-Asia conference on Advances in knowledge Discovery and Data Mining (PAKDD'06), 2006, pp. 577–593.

[60] M. Bai, X. Wang, J. Xin, G Wang, An efficient algorithm for distributed density-based outlier detection on big data, Neurocomputing 181 (2016) 19–28.

[61] D. Ren, B. Wang, W Perrizo, RDF: a density-based outlier detection method using vertical data representation, International Conference on Data Mining (ICDM'04), 2004, pp. 503–506.

[62] L.J. Latecki, A. Lazarevic, D Pokrajac, Outlier detection with kernel density functions, Proceedings of the 5th international conference on Machine Learning and Data mining in pattern Recognition, 2007, pp. 61–75.

[63] J. Gao, W. Hu, Z.M. Zhang, X. Zhang, O Wu, RKOF: robust kernel-based local outlier detection, Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2011, pp. 270–283.

[64] M. Pavlidou, G Zioutas, Kernel density outlier detector, Topics in Nonparametric Statistics, Springer, New York, NY, 2014, pp. 241–250.

[65] ZhengZ., JeongH.Y., HuangT., ShuaJ. Kde based outlier detection on distributed data streams in multimedia network. Multim. Tool Appl.2016;1–19.

[66] L. Zhang, J. Lin, R Karim, Adaptive kernel density-based anomaly detection for nonlinear systems, Knowl. Based Syst. 139 (2018) 50–63.

[67] X. Qin, L. Cao, E.A. Rundensteiner, S Madden, Scalable kernel density estimation-based local outlier detection over large data streams, EDBT (2019) 421–432.

[68] H. Kriegel, P. Kröger, E. Schubert, A Zimek, LoOP: local outlier probabilities, Proceedings of the 18th ACM conference on Information and knowledge management (CIKM'09), Hong Kong, China, 2009, pp. 1649–1652.

[69] H. Fan, O.R. Za¨ıane, A. Foss, J Wu, Resolution-based outlier factor: detecting the top-n most outlying data points in engineering data, Knowl. Inf. Syst. 19 (1) (2009) 31–51.

[70] R. Momtaz, N. Mohssen, M.A Gowayyed, DWOF: a robust density-based outlier detection approach, Iberian Conference on Pattern Recognition and Image Analysis, 2013, pp. 517–525.

[71] B. Tang, H He, A local density-based approach for outlier detection, Neurocomputing 241 (2017) 171–180.

[72] T Jin, Research on reversed modeling method for thermal power unit, School of Energy, Power and Mechanical Engineering, Beijing, North China Eleetric Power University, 2011Vol. Doctor.

[73] M. Qi, Z. Fu, F Chen, Outliers detection method of multiple measuring points of parameters in power plant units, Appl. Therm. Eng. 85 (2015) 297–303.

[74] I.M. Cecílio, J.R. Ottewill, J. Pretlove, N.F Thornhill, Nearest neighbors method for detecting transient disturbances in process and electromechanical systems, J. Process Control 24 (9) (2014) 1382–1393.

[75] L. Cai, N.F. Thornhill, S. Kuenzel, B.C Pal, Real-time detection of power system disturbances based on $ K $-nearest neighbor analysis, IEEE Access 5 (2017) 5631–5639.

[76] Y. Djenouri, A. Belhadi, J.C.W. Lin, A Cano, Adapted k-nearest neighbors for detecting anomalies on spatio–temporal traffic flow, IEEE Access 7 (2019) 10015–10027.

[77] S. Mohseni, A Fakharzade, A new local distace-based outlier detection approach for fuzzy data by vertex metric, 2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI), 2015, pp. 551–554.

[78] Y. Chen, L Tu, Density-based clustering for real-time stream data, the Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. KDD '07, New York, NY, USA, 2007, pp. 133–142.

[79] Al-ZoubiBelalM.An effective clustering-based approach for outlier detection. Eur. J. Sci. Res.2009;28(2):310–316.

[80] P. Yang, B Huang, KNN based outlier detection algorithm in large dataset, 2008 International Workshop on Education Technology and Training & 2008 International Workshop on Geoscience and Remote Sensing, 1 2008, pp. 611–613.

[81] S. Xu, C. Hu, L. Wang, G Zhang, Support vector machines based on K nearest neighbor algorithm for outlier detection in WSNs, 2012 8th International Conference on Wireless Communications, Networking and Mobile Computing, 2012, pp. 1–4.

[82] H. Rizk, S. Elgokhy, A.M Sarhan, A hybrid outlier detection algorithm based on partitioning clustering and density measures, 2015 Tenth International Conference on Computer Engineering & Systems (ICCES), 2015, pp. 175–181.

[83] X. Wang, X.L. Wang, Y. Ma, D.M Wilkes, A fast MST-inspired kNN-based outlier detection method, Inf. Syst. 48 (2015) 89–112.

[84] J. Liu, E Zio, KNN-FSVM for fault detection in high-speed trains, 2018 IEEE International Conference on Prognostics and Health Management (ICPHM), 2018, pp. 1–7.

[85] F.T. Liu, K.M. Ting, Z.H Zhou, Isolation forest, 2008 Eighth IEEE International Conference on Data Mining, 2018, pp. 413–422.

[86] LinZ., LiuX., ColluM. Wind power prediction based on high-frequency SCADA data along with isolation forest and deep learning neural networks. Int. J. Elect. Pow. Energy Syst.2020;118- 105835.

[87] ShenL., DuH., LiuS., ChenS., QiaoL., LiuS., LiJ. Real time outlier monitoring for power transformer fault diagnosis based on isolated forest. In IOP Conf. Series Mater. Sci. Eng.2020;715(1):012033.

[88] HaririS., KindM.C., BrunnerR.J. Extended isolation forest. 2018;arXiv preprint arXiv:1811.02141.

[89] P. Karczmarek, A. Kiersztyn, W. Pedrycz, E Al, K-Means-based isolation forest, Knowl. Based Syst. (2020) 105659.

[90] ChaithanyaP.S., PriyangaS., PravinrajS., SriramV.S.SSO-IF: an Outlier Detection Approach for Intrusion Detection in SCADA Systems. In Invent. Commun. Comput. Tech.2020;921–929.

[91] J. Ren, R Ma, Density-based data streams clustering over sliding windows, International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), 5 2009, pp. 248–252.

[92] WeekleyR.A., GoodrichR.K., CornmanL.B. An algorithm for classification and outlier detection of time-series data. J. Atmos. Ocean. Tech.2010;27(1):94–107.

[93] ToshniwalD.A framework for outlier detection in evolving data streams by weighting attributes in clustering. Procedia Tech.2012;6:214–222.

[94] Hourly photovoltaic generation data. [Online]. Available: < https://www.pvoutput.org >.

[95] Hourly load consumption. [Online]. Available: < https://openei.org/datasets/files/961/pub >.

[96] Hourly load data. [Online]. Available: < http://www.ercot.com/gridinfo/load/load_hist >.

[97] Hourly temperature data. [Online]. Available: < https://maps.nrel.gov/nsrdb-viewer >.

[98] K.G. Ranjan, B.R. Prusty, D Jena, Comparison of two data cleaning methods as applied to volatile time-series, 2019 International Conference on Power Electronics Applications and Technology in Present Energy Scenario (PETPES), Mangalore, India, 2019, pp. 1–6.