University of Maribor Press

DOI: https://doi.org/10.18690/um.fov.5.2026.5
ISBN: 978-961-299-160-9

Fault Detection in Solar Power Plants Based on Energy Production Data

Dominykas Vilčinskas,
Lukas Voveris, and
Jolita Bernatavičienė

Abstract
This research addresses the critical need for the timely identification of faults in solar power plants to minimize electricity loss. The study analyses energy production data from a Lithuanian solar power plant, comprising 143 strings distributed across 12 inverters, over a 19-month period. During data preprocessing, 16 key features were extracted from each string's time series data to represent the global structure of the data. The extraction process resulted in a transformed dataset, where each time series is represented as an object with 16 features, enabling more effective analysis. Statistical and machine learning techniques - including PCA + α-HULL, Isolation Forest (iForest), and Local Outlier Factor (LOF) - were employed to identify systems exhibiting abnormal behavior. The results demonstrate that a combination of these methods can help effectively identify outliers, with a combined anomaly score providing a comprehensive assessment of string performance. Additionally, RANSAC and DBSCAN methods were used to construct fault profiles, which enabled a more in-depth analysis of each system's performance and provided further confirmation of previously identified systems exhibiting abnormal behavior.

Keywords: solar power plants; fault detection; anomaly detection; energy production; time series features; PCA; α-HULL; Isolation Forest; Local Outlier Factor; DBSCAN

Dominykas Vilčinskas and Lukas Voveris
Vilnius University, Institute of Applied Mathematics, Naugarduko str. 24, 03225 Vilnius, Lithuania
e-mail: dominykas.vilcinskas,lukas.voveris@mif.stud.vu.lt
Jolita Bernatavičienė
Vilnius University, Institute of Data Science and Digital Technologies, Akademijos str. 4, 03225 Vilnius, Lithuania
e-mail: jolita.bernataviciene@mif.vu.lt

1 Introduction

Solar power is increasingly prevalent, largely driven by growing concerns about climate change and environmental sustainability. Its widespread acceptance stems from its reputation as a practical solution to rising energy demands without worsening environmental degradation. Governments, corporations, and individuals are turning to solar energy for its clean, renewable characteristics and its potential for long-term cost savings. However, solar power plants face various challenges: according to [1], without proper maintenance, PV power plants have a relatively high likelihood of operating unsatisfactorily, resulting in energy losses of up to 10%. Therefore, numerous studies have been conducted to identify anomalies and issues in solar power plants, as early detection and mitigation increase overall plant efficiency. Solar power plants are susceptible to various malfunctions. Vishwakarma et al. [2] categorized faults into two main groups: acute and chronic. One subset of chronic faults is shadowing. According to [3], shadows reduce both the anticipated power and the actual power output of the shaded system. Shadowing can be a persistent problem when solar irradiance is sufficiently high.

The study [4] evaluates the performance of a solar energy generation system by comparing its energy output against a reference dataset. For any solar panel system of interest, they used either global irradiance data or the energy output of a nearby system with the same capacity and specifications as a reference. Under ideal conditions, the observed solar panel system is expected to exhibit a nearly linear relationship with the reference source, except for noise arising from data input errors. To compare any system with a reference source, the Random Sample Consensus (RANSAC) method is used. Tsafarakis and van Sark conducted further studies on the same methodology [5]. The authors have developed a new algorithm that creates a reference dataset by using the data of other PV systems in the surrounding area. The authors then used RANSAC to identify outlying energy production data points. The distribution of these points was further analysed by plotting them on a scatter plot, categorized by the hour and time of day, across every day of the year. Since there were anomalous points that were not consistent in a scatter plot based on hourly and daily time (marking them as noise), ”Density-Based Spatial Clustering of Applications with Noise” (DBSCAN) [6] algorithm was used to eliminate the unwanted noise. Detected clusters are then grouped to form a profile of shadow affecting the solar panel. This profile allows for the identification of the extent and intensity of shadowing. Since solar energy production typically comes as a type of time series, other research [7] has been conducted on anomaly detection in time series, which could be useful in this context. In the paper, the authors sought to identify servers exhibiting unusual behavior by detecting anomalous patterns. The performance of each server is characterized using univariate time-series data. The article presents an idea that extracts 18 scalar features from distinct time series and then applies dimensionality reduction methods, such as Principal Component Analysis (PCA) [8], to identify patterns across these collections. This simplification enables the use of various high-dimensional outlier detection algorithms to identify anomalous time series.

This paper examines fault detection in solar power plants using energy production data. The proposed approach uses features extracted from time-series data of solar panel

strings to identify systems exhibiting abnormal behavior. The study compares the performance of three anomaly detection methods: PCA with α-HULL, Isolation Forest, and Local Outlier Factor. In addition, fault profiles are constructed for the most anomalous systems using RANSAC and DBSCAN, allowing a more detailed analysis of their behavior. The remainder of the paper is organized as follows. The second section describes the dataset and preprocessing steps. The third section explains the methodology. The fourth section presents and compares the results. Finally, the last section discusses the main conclusions and limitations of the study.

2 Data

Solar power plant data has been received from a Lithuanian company. This dataset includes detailed information on the power plant's solar energy production. There are twelve different inverters. Each inverter converts the direct current generated by a set of solar panel strings—the number of strings connected to each inverter may vary, ranging from 10 to 13. Inverters 1, 2, 3, 9, 10 have 13 strings. Inverters 4, 5, 8, 11 have 12 strings. Inverters 6, 7, 12 have 10 strings. In total, five inverters have 13 strings, four have 12, and three have 10.

Each string has exactly 34 solar panels, all of the same specifications. Energy production data for each string in the power plant are provided. Each record consists of the following features: timestamp, electric current (A), and voltage (Vdc). There are precisely 57788 records collected between March 1, 2022, and October 24, 2023, each 15 minutes apart and describing the energy generation of 143 different strings in the solar power plant.

Having electrical current (A) and voltage (Vdc) of some solar panel string at a given timestamp, the total energy generated by a string for that specific timestamp could be described as electrical power (W), which can be calculated using the given formula:

$$P = V \cdot I, \tag{1}$$

here

- P - electrical power (W),
- V - voltage (V),
- I - electrical current (A).

Figure 1 presents a comparative analysis of daily energy generation for each string across selected months: December 2022, January, February, June, July, and August 2023. These months were chosen to capture the contrasting patterns in energy production between the colder, low-solar-irradiance months and the peak solar activity during summer. For each month, the days illustrated correspond to the peak daily energy generation observed across the entire power plant. The data reveal a seasonal pattern, with energy production significantly lower in winter than in summer. Moreover, during the summer period, discrepancies between individual strings become more evident, as underperforming strings, those producing less electricity than the majority, stand out

more. In contrast, during the winter months, such deviations are minimal or nearly absent. This comparative analysis emphasizes both the seasonal impact on overall energy output and the variability in relative performance among the strings under different solar irradiance conditions.

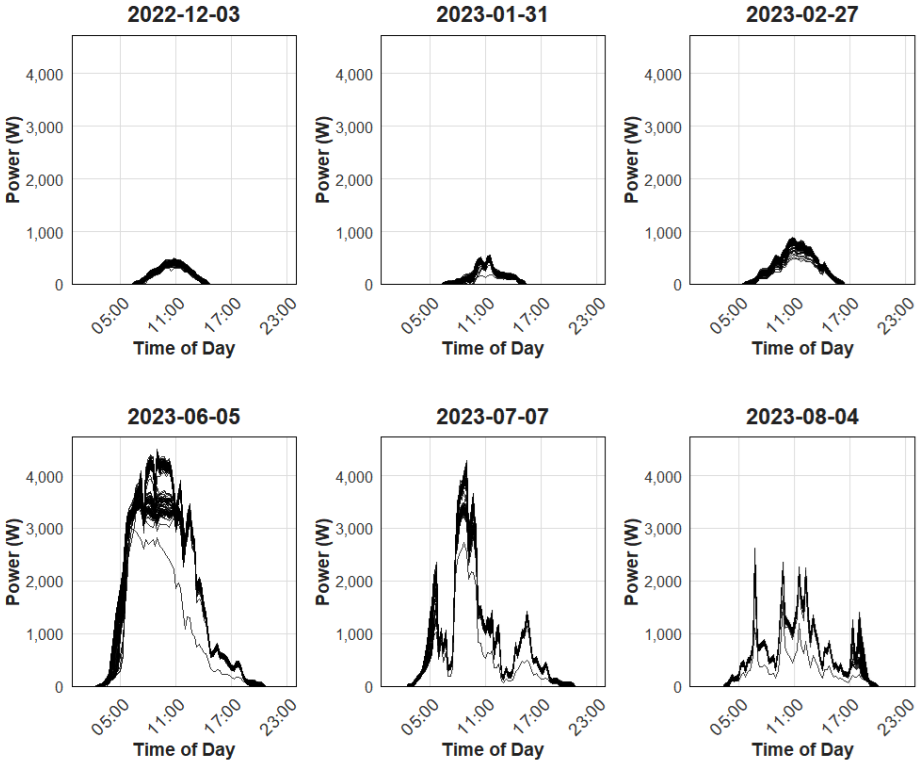

Fig. 1: Example of energy generation for each string in the solar power plant during different months.

Table 1 presents descriptive statistics for the total energy generated by solar panel strings. In fact, the 1st quartile Q_1 shows us that only 25 % of strings generated less than 43000 kWh. The similarity between the mean and median values suggests that there are no outliers based solely on total energy generated, as the mean is typically highly influenced by their presence.

Further, the total energy generated by the power plant during the same months was calculated, as shown in Figure 1. The doughnut plot presented in Figure 2 shows the distribution of the total generated energy across all four seasons. The largest share of

Table 1: Statistics of Total Power (kW)

Min.	Q_1	Median	Mean	Q_3	Max.
27826	43123	44148	44328	45876	48058

energy was generated in summer, accounting for 44.4% of the total annual production. Spring contributed 36.4%, autumn 14.9%, while winter represented the smallest share at 4.3%. These results show a clear seasonal pattern in the power plant's energy generation, with most electricity produced during periods of higher solar irradiance and longer daylight duration.

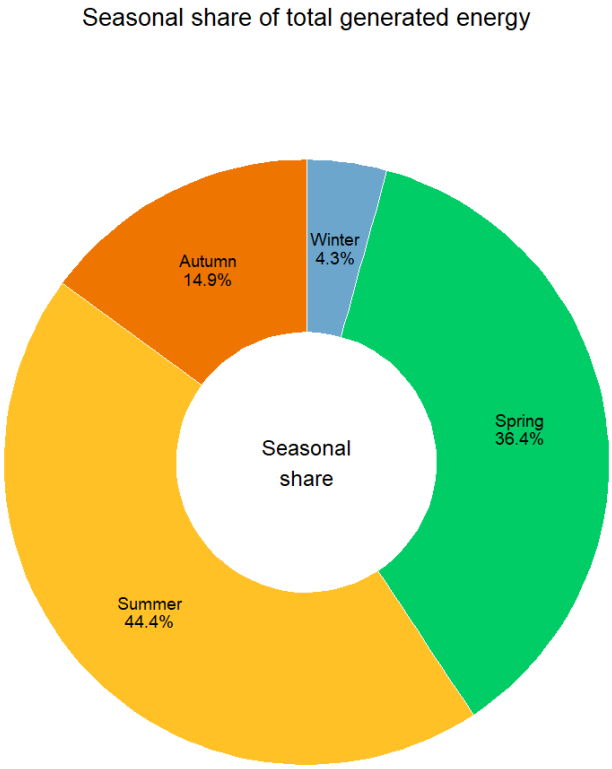

Fig. 2: Seasonal share of total generated energy of the entire power plant.

A brief data-quality analysis was performed before the anomaly-detection stage. The dataset contained 748 timestamps for which all measurements were missing across all 143 strings. Because neither electric current nor voltage was measured, 213,928 missing values were observed. The missing observations were grouped into three consecutive time intervals: 2022-05-24 22:30:00 to 2022-06-01 13:00:00, 2022-09-15 14:00:00 to 2022-09-15 15:00:00, and 2023-06-29 12:15:00 to 2023-06-29 15:00:00. To preserve continuity of the time series, the missing values were imputed by averaging the corresponding observations from the same time one day before and one day after the missing interval. This approach was chosen because the data are strongly periodic and neighboring days preserve the main daily production pattern.

An additional issue concerned negative current values. Across all strings, 54,690 cases of negative electric current were observed. Since negative current values do not represent meaningful production in this context and are likely due to inverter behavior during non-generating periods or measurement noise, these values were replaced with zero before power calculations. This preprocessing step ensured that the derived power values reflected only non-negative production levels and reduced the risk of introducing artificial anomalies in subsequent analyses.

To better assess whether simple aggregate statistics were sufficient to identify faulty strings, additional exploratory analysis was conducted. Pairwise Pearson correlations between strings were generally high, with an average correlation of 0.987, a median of 0.991, and a maximum of 0.999. At the same time, the minimum observed correlation was 0.780, indicating that a small number of strings deviated from the common behavior of the plant. This result suggests that most strings behave in a highly similar manner over time. In contrast, a small number exhibit persistent differences that may be linked to faults or long-term performance degradation.

A similar conclusion follows from the distribution of total generated power. Although the first quartile was 43,123 kW and the median was 44,148 kW, only three strings generated less than 40,000 kW over the full study period. This shows that total production alone is not sufficient for identifying all problematic systems, because many faults may affect only specific hours or seasons rather than the overall cumulative output. For this reason, the later methodology relies on a broader set of extracted features that capture temporal structure, seasonality, and changes in behavior across the whole time series.

3 Methodology

A summary of the workflow is presented in Figure 3. The work is mainly comprised of two parts. The first section addresses outlier detection, which aims to identify abnormally behaving solar panel strings. The second one focuses on constructing fault profiles for the most anomalous strings, providing a more detailed analysis of each system's performance and insights into previous findings.

In order to apply the chosen unsupervised anomaly detection methods to the solar panel string data, it is necessary to extract features that effectively capture the global structure and underlying patterns of the time series (see Table 2).

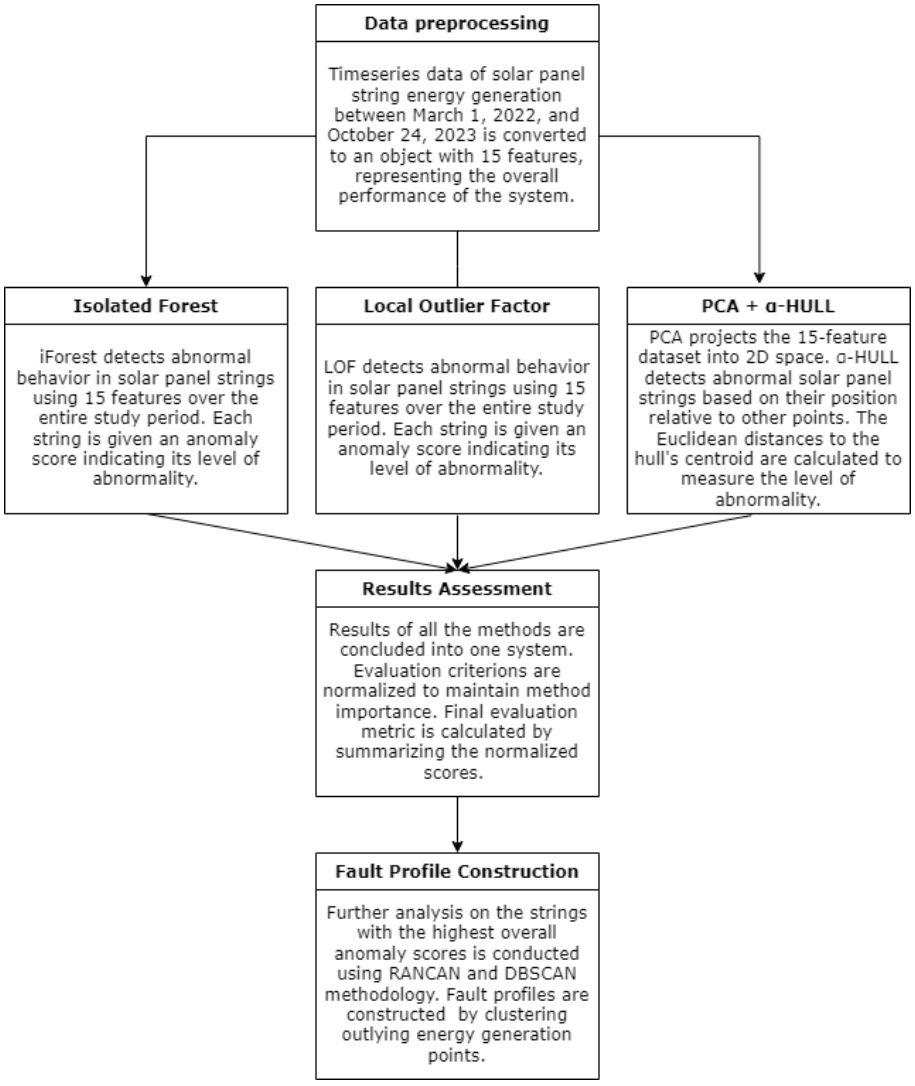

Fig. 3: Summary workflow chart.

To compute features like ACF1, trend strength, linearity, curvature, seasonal strength, entropy, peaks and troughs the R package *tsfeatures* [12] was used. These features are calculated using the STL (Seasonal and Trend decomposition using Loess) [13] method. The decomposition contains a trend, up to M seasonal components and a remainder component:

$$x_t = f_t + s_{1,t} + ...s_{M,t} + e_t \tag{2}$$

where

Table 2: Features extracted from solar panel string time series for unsupervised anomaly detection.

Variable	Description
Mean	Mean value of the time series.
Variance	Variance of the time series.
ACF1	First-order autocorrelation.
$\text{Trend}_{\text{Strength}}$	Indicates how strong the trend is.
Linearity	Degree to which the data follows a straight line.
Curvature	Degree to which the data follows a curved pattern.
$\text{Seasonality}_{\text{Strength}}$	Indicates how strong the seasonal pattern is.
Entropy	Measures the "forecastability" of the time series.
Peak	Strength of peaks; computed from size and location of peaks in the seasonal component.
Trough	Strength of troughs; computed from size and location of troughs in the seasonal component.
Crossings	Number of times the time series crosses the mean line.

- f_t is the smoothed trend component,
- $s_{i,t}$ is the i-th seasonal component,
- e_t is a remainder component.

The trend and seasonality strengths are calculated by the following formulas:

$$Trend_{\text{Strength}} = 1 - \frac{Var(e_t)}{Var(f_t + e_t)} \tag{3}$$

$$Seasonality_{\text{Strength}} = 1 - \frac{Var(e_t)}{Var(s_{i,t} + e_t)} \tag{4}$$

Additional features included are hourly energy generation. For each object, 4 features are computed representing the total energy generated per grouped hours — from 08:00 to 11:00, 12:00-15:00, 16:00-19:00 and 20:00:23:00 between March 1, 2022, and October 24, 2023. Hours 24:00-07:00 are not included because none of the strings generated any energy during this period. In fact, this may help identify shadowing, as if some system is shaded during some specific hours, its power output will be significantly lower than others during these time-frames. Therefore, with the addition of hourly energy generation, each time series will contain a total of 15 features. Given the diverse scales of features, min-max normalization will be used.

3.1 Outlier detection

After data preprocessing, a dataset with 143 objects and 16 features has been obtained, where each row represents a solar panel string. The computed features provide insights

into the performance of each individual string throughout the entire examined time frame. Thus, now various strategies of high dimensional data anomaly detection can be applied to identify abnormally behaving systems.

3.1.1 PCA and α-HULL

Anomaly detection in high dimensional data can be effectively conducted by integrating PCA [8] with the α-hull method [11]. To begin, PCA is applied to the dataset to reduce its dimensionality, allowing the identification of the number of principal components that adequately capture the essential variance, tailored to specific use cases. Subsequently the α-hull method can be performed on the reduced dataset. The α-hull method is a generalization of convex hull method. Convex hull is the smallest convex set that contains a set of points. It is similar to stretching a rubber band around the outermost points, the area the rubber band encloses is the convex hull. The α-hull method extends this concept by introducing a parameter α, which represents the radius of a generalized disk. The α-hull method uses these disks to determine the boundary of the point set. The size of α crucially affects the resulting shape:

- When α is very small, the α-hull can potentially enclose smaller clusters of points or even individual points, showing a more detailed boundary.
- When α is very large, the α-hull approaches the convex hull, as the generalized disks become large enough to enclose all points, and the shape simplifies to that of a convex hull.

Although this method does not directly yield any anomaly score, a naive scoring system based on the Euclidean distance from the centroid C of the cluster enclosed by the hull was implemented. The centroid coordinates of cluster A were calculated using the points that are in that cluster:

$$C = (C_x, C_y) \tag{5}$$

where

$$C_x = \mathbb{E}(\sum_{o \in A} o_x) \tag{6}$$

and

$$C_y = \mathbb{E}(\sum_{o \in A} o_y) \tag{7}$$

3.1.2 Isolation forest

Isolation Forest [9] is a tree-based approach for outlier detection. It works by isolating anomalous instances from normal ones through the construction of binary tree data structures across the features of the dataset. The iForest algorithm operates through a series of steps:

1. **Random Feature Selection**. The algorithm begins by randomly selecting a feature from the dataset.
2. **Random Split Value Selection**. Once a feature is chosen, Isolation Forest randomly selects a split value between the minimum and maximum values of that feature.
3. **Binary Tree Construction**. Using the randomly selected feature and split value, iForest constructs a binary tree data structure.

These steps are iterated for the remaining features until binary trees are constructed for all features. Outliers are identified based on their average path length in each tree. Anomaly score equation:

$$s(z, n) = 2^{-\frac{\mathbb{E}(h(z))}{c(n)}}, \tag{8}$$

where:

- z is data point,
- n is total number of instances in the dataset,
- $h(z)$ is path length of a point z, measured by the number of edges z traverses from the root node until a leaf node is reached,
- $\mathbb{E}(h(z))$ is average path length of point z over all binary trees,
- $c(n) = 2H(n-1) - \frac{2(n-1)}{n}$ is path length normalization constant; $H(i)$ is the i-th harmonic number.

3.1.3 Local outlier factor

Unsupervised anomaly detection for high-dimensional data can also be done using the LOF method [10]. LOF is based on the local density deviation of a given data point with respect to its neighbors. The main advantage of this method compared to other outlier detection methods is that it can identify local outliers. The method starts by finding the k nearest neighbors $N_k(z)$ of data point z. If a tie between some points occurs, more than k points may be used. Then, for each point, the average local reachability density (LRD) is computed:

$$LRD_k(z) = \frac{\sum_{o \in N_k(z)} dist(z, o)}{|N_k(z)|}, \tag{9}$$

where

- $dist(z, o)$ – Distance measure between points z and o. In our case, the Euclidean metric is used:

$$d(z, o) = \sqrt{(z_1 - o_1)^2 + (z_2 - o_2)^2 + \cdots + (z_n - o_n)^2}. \tag{10}$$

- $|N_k(z)|$ – Number of elements in set $N_k(z)$.

The final step is to calculate the Local Outlier Factor (LOF), which is the average local reachability density of the neighbors divided by the object's own local reachability density:

$$LOF_k(z) = \frac{\sum_{o \in N_k(z)} LRD_k(o)}{|N_k(z)| \cdot LRD_k(z)}. \tag{11}$$

3.2 Fault profile construction

Before proceeding to construct the fault profile, the results provided by different models must first be evaluated. To achieve this, a system that calculates a combined anomaly score based on the outputs of various methods was introduced. Specifically, the anomaly scores from iForest, LOF, and the Euclidean distance from α-HULL were used. To ensure consistent influence among all methods, min-max normalization was applied to these scores. Then the normalized scores were aggregated to get a final evaluation metric. The five systems exhibiting the highest anomaly scores will be thoroughly investigated by using the RANSAC + DBSCAN methodology:

1. **Outlier Detection**: The first step identifies outliers in power output values. It is assumed that nearby solar systems with identical specifications exhibit a linear relationship under normal conditions. For each solar panel string, an optimal energy reference is built using its neighboring strings from the same inverter. The benchmark is the maximum energy output of these neighbors at each timestamp. Linear regression and RANSAC then estimate this linear relationship, flagging deviations as outliers.
2. **Pattern Visualization**: Outliers are plotted with hour on the x-axis and date on the y-axis, excluding nighttime. Clusters of outliers at specific times or dates reveal patterns of abnormal performance.
3. **Outlier Clustering**: DBSCAN groups dense clusters of outliers while filtering out noise. This produces a clear and consistent fault profile for each string.

4 Experimental results

The implementation results of α-HULL with varying α values are depicted in Figure 4. For each α value, an α-convex hull was constructed, with the generalized disk radius set equal to α. As shown in the graphs, selecting a lower α value results in more outlying points. When α was set to 1 and 1.25, two strings from the 8th inverter, one string from 11th inverter and two strings from 12th inverter were found to be outliers. When α value was reduced to 0.75 then 3 additional outliers were detected, one of each from the 8th, 11th and 12th inverters. When α value was further lowered by 0.25, then 11 new outliers were caught. Among those 11 strings 8 were from previously not detected inverters (1st, 2nd, 4th, 5th and 9th). The maximum number of anomalous strings detected is 19. Notably, five outliers - 8.11, 8.12, 11.01, 12.02, and 12.03 — are consistently present across all variations. Here, the notation $x.y$ indicates inverter x and string y.

For every outlier identified Euclidean distance (10) between that point and the computed centroid C was calculated. Results when using $\alpha = 0.5$ are shown in Table 3. In

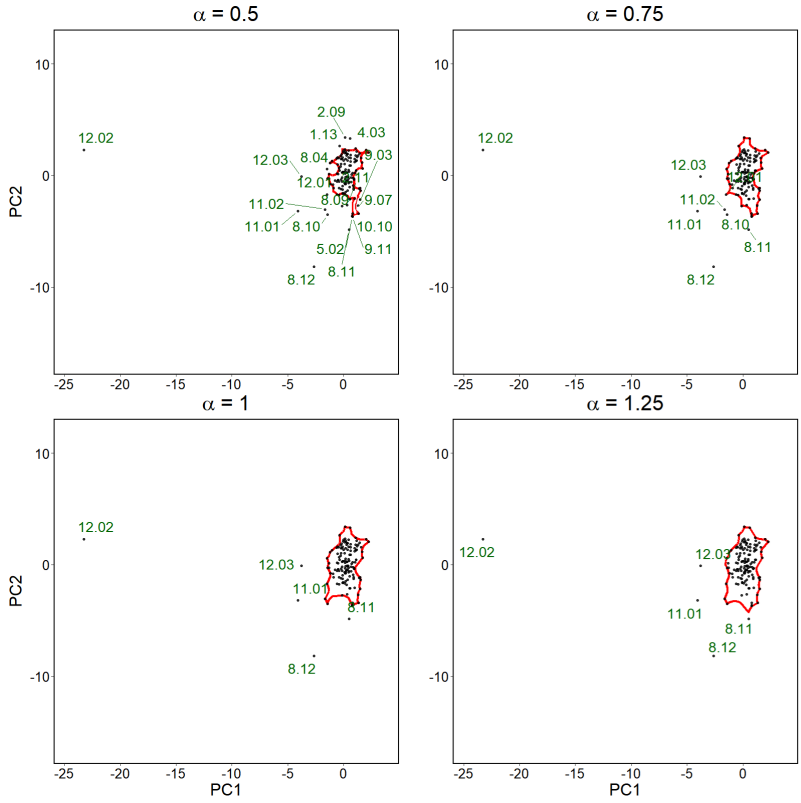

Fig. 4: α-HULL results.

scenarios with different α values, the distances differ due to discrepancies in both the outliers and the computed centroids. Hence, primarily, focus was put on the outcome when $\alpha = 0.5$, as it offers a broader selection of points.

In the case of the iForest method, a lot more information about the solar panel string is used. The algorithm uses all 16 features to detect abnormally behaving strings. The decision was made to check the results of outliers using two different *contamination* parameter values: 0.05 and 0.10. This implies that encountering a total of either 5 % or 10 % outliers within the entirety of the dataset was expected.

For the LOF method, the *contamination* parameter must also be specified. Values of 0.05 and 0.10 were chosen, expecting 5 % or 10 % outliers respectively. The neighborhood size for density computation is fixed at 10, determined after experimenting with different values. Results are shown in Table 5.

For the comparison iForest with a *contamination* parameter set to 0.10, the LOF method with a *contamination* value of 0.10, and the PCA + α-HULL method with an α value set to 0.5 were used.

Table 6 presents the shared outliers identified among results obtained from different methods. Notably, four outlier strings (12.02, 8.11, 12.01, and 8.10) are consistently

Table 3: α-Hull outlier distances, when α = 0.5.

String	Distance
12.02	23.63597
8.12	8.871502
11.01	5.533009
8.11	5.072597
8.10	4.097642
12.03	4.052169
9.11	3.939188
5.02	3.908802
10.10	3.778132
11.02	3.774506
9.07	3.662785
2.09	3.183685
4.03	3.13508
9.03	3.108478
8.09	2.979747
2.11	2.883931
12.01	2.606329
1.13	2.509189
8.04	1.778152

Table 4: iForest results.

Contamination = 0.05		Contamination = 0.1	
String	Anomaly Score	String	Anomaly Score
12.01	0.248189	12.01	0.267126
8.11	0.138375	8.11	0.157312
12.02	0.064189	12.02	0.083125
10.13	0.058485	10.13	0.077421
8.10	0.024483	8.10	0.043420
12.04	0.014717	12.04	0.033654
8.09	0.005382	8.09	0.024318
8.08	0.000806	8.08	0.019742
		6.09	0.011683
		1.02	0.005514
		12.08	0.003381
		7.10	0.002234
		1.08	0.002157
		11.12	0.000340
		8.07	0.000027

flagged across all three methods. Furthermore, there were seven mutual outliers between iForest and LOF, five between iForest and α-HULL, and six between LOF and α-HULL method pairs.

Different anomaly scores were aggregated across all the methods to provide an overall assessment of the system's performance relative to others. Min-max normalization

Table 5: LOF results.

Contamination = 0.05		Contamination = 0.10	
String	Anomaly Score	String	Anomaly Score
12.01	2.955928	12.01	2.955928
12.02	2.086247	12.02	2.086247
8.11	2.049132	8.11	2.049132
10.13	1.740679	10.13	1.740679
1.02	1.52342	1.02	1.52342
12.10	1.453929	12.10	1.453929
		8.10	1.431485
		11.01	1.393056
		6.01	1.390974
		7.03	1.342076
		6.08	1.322753
		1.08	1.316031
		11.02	1.307126

Table 6: Common outliers.

Methods	Common outliers
iForest, LOF, α-HULL	12.02 ; 8.11 ; 12.01 ; 8.10
iForest, LOF	12.02 ; 8.11 ; 12.01 ; 8.10 ; 1.02 ; 1.08 ; 10.13
iForest, α-HULL	12.02 ; 8.11 ; 12.01 ; 8.10 ; 8.09
LOF, α-HULL	12.02 ; 8.11 ; 12.01 ; 8.10 ; 11.01 ; 11.02

is necessary to maintain the same level of influence between all methods. Furthermore, the total scores by summing the normalized results from all three methods were calculated. The 5 strings with the highest combined anomaly scores are shown in Table 7. In instances where a string was not identified as an outlier by a particular method, it was denoted with ”-”.

Table 7: Combined anomaly scores.

String	iForest	LOF	α-HULL	Total
12.01	1	1	0.037889	2.037889
12.02	0.311113	0.472538	1	1.783651
8.11	0.588864	0.450027	0.150722	1.189613
10.13	0.289758	0.26295	-	0.552708
8.10	0.16246	0.075424	0.106117	0.344001

The fault profile construction process can be illustrated using a specific example, focusing on the 12.01 solar panel string (see Figure 5), which has the highest anomaly

score among the systems analysed. The reference data source is constructed using other systems of the 12th inverter. In this visualization, inliers are denoted in green, while outliers are marked in red. The RANSAC method detects points that deviate from the found linear relationship. In the context of energy generation, points below indicate cases in which the examined system generated less than the reference source. The energy generation points are then visualized through a scatter plot, with the hour on the x-axis and the date on the y-axis. To ensure clarity, all energy generation points at night are removed, since neither the reference source nor the string of interest generates energy during these hours. Certain deviating energy-generation points consistently occur along the X-axis, indicating potential performance issues for the string on that particular day. However, the presence of points in a dense region along the X-axis and across the Y-axis suggests a frequent malfunctioning pattern. Hence, the denser regions formed represent a profile of consistent malfunction throughout specific hours. Some outliers are far from any dense cluster; they would be considered noise in this case. Notably, there are no outliers during the winter months. This could be explained by the fact that solar irradiance is significantly lower during winter. Later, DBSCAN was applied to cluster the points. It can be seen that six distinct clusters have been constructed among the observed points. Other energy-generation points, marked in grey, are treated as noise and are not further analysed. Based on these findings, the string exhibits recurring issues during two distinct time periods: from 10:00 to 12:00 and from 15:00 to 20:00. The consistent occurrence of these issues during specific hours suggests that the string is affected by shadowing. During periods of high solar irradiance, some objects cast shadows, obstructing the panels from sunlight and thereby reducing energy output.

It is also important to note that no outliers were observed during the winter months for the 12.01 string. A likely explanation is that solar irradiance is substantially lower in winter, and the overall plant generation is reduced during this period. As a result, differences between strings become less visible and chronic faults are harder to detect. This observation is consistent with the earlier seasonal analysis and further supports the decision to study fault profiles primarily during higher-irradiance periods.

Applying the same methodology to other solar panel strings identified as anomalies produces the results illustrated in Figure 6. Similar fault patterns emerge across distinct systems, with notable discrepancies in energy generation between 7:00-10:00 and 16:00-20:00 in most systems. The clearest fault pattern is observed in the 12.02 solar panel string, where DBSCAN identified two dense clusters, indicating significantly reduced energy production during hours 14:00-20:00. Given the consistency of these outliers during high-irradiance timeframes and their occurrence only during specific hours, this suggests a shadowing problem. This interpretation is supported by feedback from plant operators, suggesting that the detected patterns correspond to operational issues.

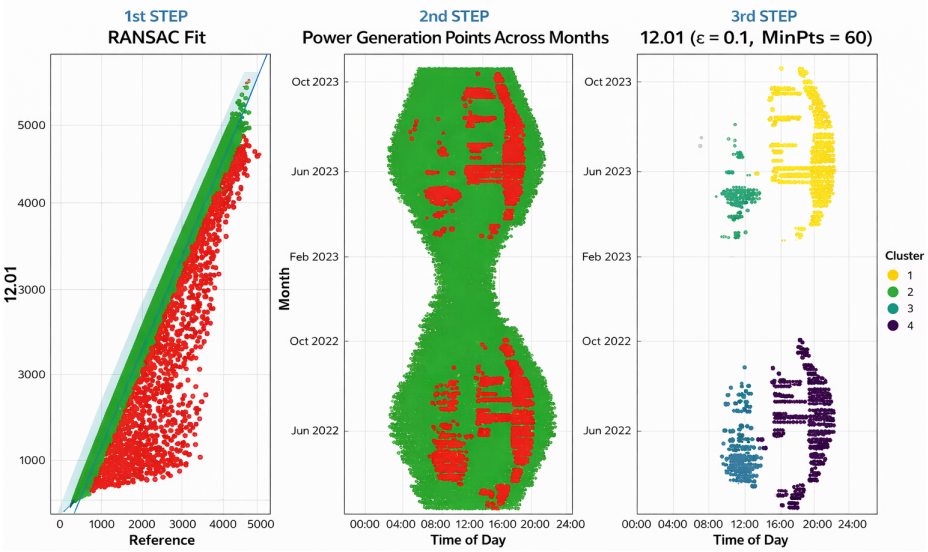

Fig. 5: Fault profile construction for 12.01 string

For comparison, the same profiling procedure was applied to a string that had not previously been identified as anomalous. String 4.03 was selected as a non-faulty example (see Figure 7). In this case, DBSCAN did not form any dense clusters from the detected outlier points. Most deviations were concentrated within a single short period and did not recur in a structured manner over time. This indicates the absence of a persistent malfunction pattern and suggests that the observed points are more likely to be noise than evidence of a chronic fault.

5 Conclusions

Through the application of various anomaly detection methods — PCA + α-HULL, Isolation Forest (iForest), and Local Outlier Factor (LOF) — multiple faulty solar panel strings in the solar power plant were successfully identified. There were four systems detected as anomalies in all three methods: 12.02, 8.11, 12.01, and 8.10.

The combination of these methods ensured a comprehensive assessment, highlighting strings with significant deviations from normal operation. The combined anomaly score, derived from the normalized results of the three methods, provided a metric for identifying the most anomalous strings. This score helped prioritize strings for further analysis.

Using RANSAC and DBSCAN methodology, detailed fault profiles were constructed for the identified faulty strings with the highest anomaly scores. These profiles revealed recurring patterns of reduced energy generation, occurring consistently during specific

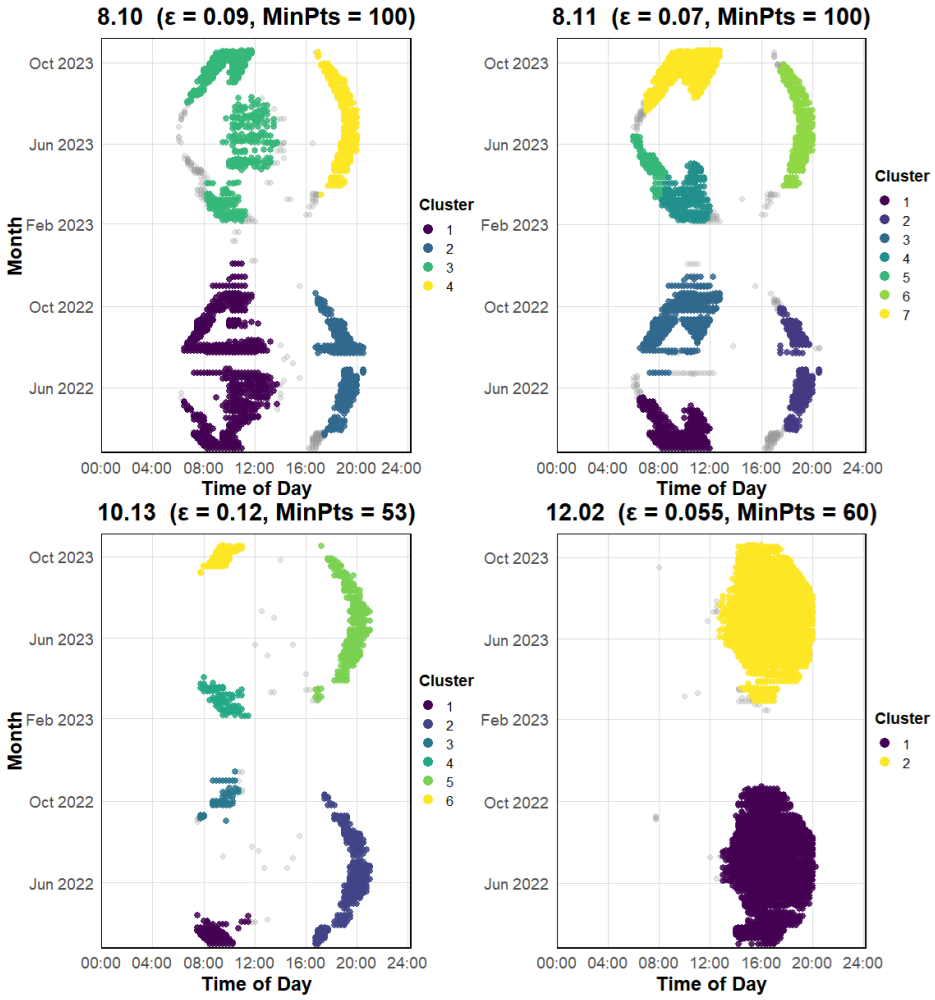

Fig. 6: Fault profiles of other strings detected as outliers

hours, indicating potential issues such as shadowing. The clustering of outliers provided a clear visualization of the malfunction periods, further validating the results from the anomaly detection methods.

This methodology also enables detailed performance analysis over any timeframe, providing clear insights into the behavior of each string. The visualizations make it easy to compare strings, and support data-driven decisions for maintenance or optimization, making it an effective tool for monitoring and managing solar power plants.

Despite the promising results, several limitations should be acknowledged.

The study relies on unsupervised methods without ground truth labels, making it difficult to evaluate detection accuracy directly. Therefore, anomalies are identified based

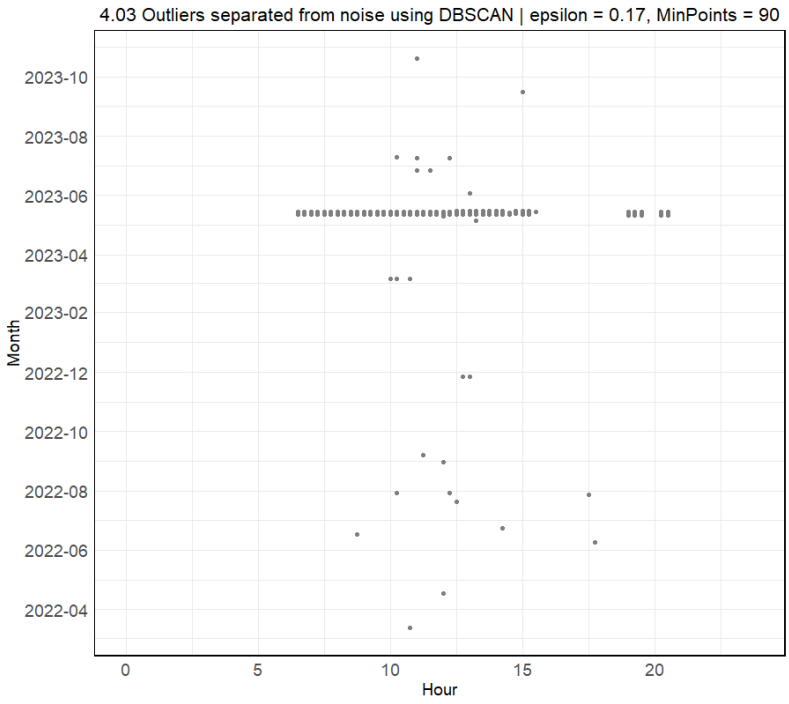

Fig. 7: 4.03 string DBSCAN results

on their consistent detection across multiple methods rather than verified fault labels. Despite this limitation, the findings are consistent with expert knowledge from plant operations, which confirmed shadowing in most of the detected strings.

The performance of the methods is also sensitive to parameter selection. Parameters such as the α value in α-HULL, contamination in Isolation Forest, and neighborhood size in LOF can influence the results. These parameters can be refined using expert knowledge or practical insights, thereby improving the reliability of anomaly detection.

In addition, the selected feature set, while informative, may not capture all aspects of abnormal behavior. External factors such as weather conditions or solar irradiance were not included and could improve detection performance.

Finally, the dataset is limited to a single solar power plant with identical component specifications, which may limit the generalizability of the results to other plants or systems with different configurations.

References

1. Perdue, M., & Gottschalg, R. (2015). Energy yields of small grid connected photovoltaic system: Effects of component reliability and maintenance. *IET Renewable Power Generation*, *9*(5), 432–

437. https://doi.org/10.1049/iet-rpg.2014.0048
2. Gupta, Y., Yadav, N. P., Singh, A., Kumar, A., & Vishwakarma, S. (2022). Faults occur in solar PV power generation system. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, *11*(3), 1368–1375.
3. Bernadette, D., Twizerimana, M., Bakundukize, A., Pierre, B. J., & Theoneste, N. (2021). Analysis of shading effects in solar PV system. *International Journal of Sustainable and Green Energy*, *10*(2), 47–62. https://doi.org/10.11648/j.ijrse.20211002.12
4. Tsafarakis, O., Sinapis, K., & Van Sark, W. G. (2018). PV system performance evaluation by clustering production data to normal and non-normal operation. *Energies*, *11*(4), Article 977. https://doi.org/10.3390/en11040977
5. Tsafarakis, O., & van Sark, W. G. (2023). A density-based time-series data analysis methodology for shadow detection in rooftop photovoltaic systems. *Progress in Photovoltaics: Research and Applications*, *31*(5), 506–523. https://doi.org/10.1002/pip.3661
6. Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (Vol. 96, No. 34, pp. 226–231). AAAI Press.
7. Hyndman, R. J., Wang, E., & Laptev, N. (2015, November). Large-scale unusual time series detection. In *2015 IEEE International Conference on Data Mining Workshop (ICDMW)* (pp. 1616–1619). IEEE. https://doi.org/10.1109/ICDMW.2015.104
8. Jaadi, Z. (2021, May 20). *A step-by-step explanation of principal component analysis (PCA)*. Built In. https://builtin.com/data-science/step-step-explanation-principal-component-analysis
9. Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008, December). Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining* (pp. 413–422). IEEE. https://doi.org/10.1109/ICDM.2008.17
10. Breunig, M. M., Kriegel, H. P., Ng, R. T., & Sander, J. (2000, May). LOF: Identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data* (pp. 93–104). Association for Computing Machinery. https://doi.org/10.1145/342009.335388
11. Pateiro-López, B., & Rodríguez-Casal, A. (2010). Generalizing the convex hull of a sample: The R package alphahull. *Journal of Statistical Software*, *34*(5), 1–28. https://doi.org/10.18637/jss.v034.i05
12. Hyndman, R., Kang, Y., Montero-Manso, P., O'Hara-Wild, M., Talagala, T., Wang, E., & Yang, Y. (2019). *tsfeatures: Time series feature extraction* [R package]. CRAN. https://CRAN.R-project.org/package=tsfeatures
13. Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). STL: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, *6*(1), 3–73.

About authors

Dominykas Vilčinskas is Data Science student at Vilnius University, currently pursuing a Master's degree after completing his Bachelor's studies in Data Science in 2025. During his studies, he has been actively involved in academic events and research activities focused on data analysis and machine learning. He has presented his work at several conferences, including a poster presentation titled "Fault Detection in Solar Power Plants Using Energy Production Data" at the 15th Conference on Data Analysis Methods for Software Systems in 2024. He also co-authored a publication on "Wage Prediction for Salaried Employees Using Machine Learning Methods," presented at the national conference Lietuvos magistrantų informatikos ir IT tyrimai in 2025. His main research interests include applied machine learning and data-driven decision making.

Lukas Voveris is a Master's student in Data Science at Vilnius University, Faculty of Mathematics and Informatics. He earned his Bachelor's degree in Data Science at Vilnius University in 2025. His research focuses on solar PV monitoring and energy analytics. He co-authored and presented the poster "Fault Detection in Solar Power Plants Using Energy Production Data" at the 15th Conference on Data Analysis Methods for Software Systems in 2024. He also authored "Implementation of Machine Learning and Statistical Techniques in Solar Energy Generation Monitoring Systems". This paper was presented at the national conference Lietuvos magistrantų informatikos ir IT tyrimai in 2025 and at the AI2SEP project conference in Varaždin, Croatia. His research interests include applied machine learning, time series forecasting, solar irradiation modeling, and decision support for energy systems.

Jolita Bernatavičienė graduated from Vilnius Pedagogical University in 2004 and received a master's degree in informatics. In 2008, she received a doctoral degree in computer science (PhD) from the Institute of Mathematics and Informatics jointly with Vilnius Gediminas Technical University. She is a senior researcher at the Cognitive Computing Group of Vilnius University's Institute of Data Science and Digital Technologies. Her research interests include databases, data mining, neural networks, image analysis, visualisation, decision support systems and internet technologies, and high-performance computing. She supervises 3 PhD students and has written more than 60 articles, 18 of which are in CA WoS database.