REPUBLIC OF TURKEY
ISTANBUL GELISIM UNIVERSITY
INSTITUTE OF GRADUATE STUDIES

Department of Electrical and Electronics Engineering

IMPLEMENTING MACHINE LEARNING IN THE DETECTION OF SOLAR POWER PLANTS ANOMALIES USING A HYBRID SUPPORT VECTOR MACHINE WITH GREY WOLF OPTIMIZATION ALGORITHM

Master Thesis

Qais Ibrahim Ahmed ALSHAMMARY

Supervisor

Asst. Prof. Dr. Ercan AYKUT

Istanbul – 2023

THESIS INTRODUCTION FORM

Name and Surname	**:**	Qais Ibrahim Ahmed Alshammary
Language of the Thesis	**:**	English
Name of the Thesis	**:**	Implementing ML in the Detection of Solar Power Plants Anomalies using a Hybrid Support Vector Machine with Grey Wolf Optimization Algorithm
Institute	**:**	Istanbul Gelisim University Institute of Graduate Studies
Department	**:**	Electrical and Electronics Engineering
Thesis Type	**:**	Master
Date of the Thesis	**:**	23.05.2023
Page Number	**:**	110
Thesis Supervisors	**:**	Asst. Prof. Dr. Ercan Aykut
Index Terms	**:**	Solar Energy; Intelligent Grid System; Power Plant Anomalies; PV
Turkish Anstract	**:**	Sanayide güneş enerjisinin kullanımında önemli bir artış olmuş, bu da elektrik santrallerinden ve akıllı şebekelerden yenilenebilir enerji konusunda daha fazla farkındalığa yol açmıştır. Bu alandaki zorluklardan biri, fotovoltaik (PV) sistemlerdeki anormallikleri tespit etmektir. Bu araştırma, PV bileşenlerindeki anormallikleri belirlemek için çeşitli makine öğrenimi algoritmaları ve regresyon modelleri kullanarak bu zorluğu ele almayı amaçlamaktadır. Amaç, hangi modellerin PV sistemlerinin normal ve anormal davranışlarını en doğru şekilde ayırt edebildiğini belirlemektir. Bulgularımız, bu karmaşık problem alanında hangi makine öğrenimi

yaklaşımlarının en etkili olduğu konusunda bilinçli kararlar vermek için net bir rehberlik sağlayacaktır.

Distribution List **:**

1. To the Institute of Graduate Studies of Istanbul Gelisim University
2. To the National Thesis Center of YÖK (Higher Education Council)

Signature

Qais Ibrahim Ahmed Alshammary

REPUBLIC OF TURKEY
ISTANBUL GELISIM UNIVERSITY
INSTITUTE OF GRADUATE STUDIES

Department of Electrical and Electronics Engineering

IMPLEMENTING MACHINE LEARNING IN THE DETECTION OF SOLAR POWER PLANTS ANOMALIES USING A HYBRID SUPPORT VECTOR MACHINE WITH GREY WOLF OPTIMIZATION ALGORITHM

Master Thesis

Qais Ibrahim Ahmed ALSHAMMARY

Supervisor

Asst. Prof. Dr. Ercan AYKUT

Istanbul – 2023

DECLARATION

I hereby declare that in the preparation of this thesis, scientific and ethical rules have been followed, the works of other persons have been referenced in accordance with the scientific norms if used, there is no falsification in the used data, any part of the thesis has not been submitted to this university or any other university as another thesis.

Qais Ibrahim Ahmed Alshammary

…/…/2023

TO ISTANBUL GELISIM UNIVERSITY
THE DIRECTORATE OF GRADUATE EDUCATION INSTITUTE

The thesis study of Qais Ibrahim AHMED titled as Implementing Machine Learning in the Detection of Solar Power Plants Anomalies using a Hybrid Support Vector Machine with Grey Wolf Optimization Algorithm has been accepted as MASTER in the department of Electrical- Electronics Engineering by out jury.

Director _____
Prof. Dr. Kemalettin TERZI

Member _____
Asst. Prof. Dr. Ercan AYKUT
(Supervisor)

Member _____
Asst. Prof. Ulvi BAŞPINAR

APPROVAL

I approve that the signatures above signatures belong to the aforementioned faculty members.

... / ... / 20..

Prof. Dr. Izzet GUMUS
Director of the Institute

SUMMARY

There has been a significant increase in the use of solar energy in industry, which has led to greater awareness about renewable energy from power plants and smart grids. One challenge in this field is detecting photovoltaic (PV) system anomalies. This research aims to address this challenge by using various machine learning algorithms and regression models to identify abnormalities in PV components. The goal is to determine which models can most accurately distinguish between normal and abnormal behavior of PV systems. Our findings will provide clear guidance for making informed decisions about which machine-learning approaches are most effective in this complex problem.

Key Words: Solar Energy; Intelligent Grid System; Power Plant Anomalies; PV

ÖZET

Sanayide güneş enerjisinin kullanımında önemli bir artış olmuş, bu da elektrik santrallerinden ve akıllı şebekelerden yenilenebilir enerji konusunda daha fazla farkındalığa yol açmıştır. Bu alandaki zorluklardan biri, fotovoltaik (PV) sistemlerdeki anormallikleri tespit etmektir. Bu araştırma, PV bileşenlerindeki anormallikleri belirlemek için çeşitli makine öğrenimi algoritmaları ve regresyon modelleri kullanarak bu zorluğu ele almayı amaçlamaktadır. Amaç, hangi modellerin PV sistemlerinin normal ve anormal davranışlarını en doğru şekilde ayırt edebildiğini belirlemektir. Bulgularımız, bu karmaşık problem alanında hangi makine öğrenimi yaklaşımlarının en etkili olduğu konusunda bilinçli kararlar vermek için net bir rehberlik sağlayacaktır.

Anahtar kelimeler: Güneş enerjisi; Akıllı Şebeke Sistemi; Santral Anomalileri; PV

TABLE OF CONTENTS

ABBREVIATIONS

AFCI	:	Arc Fault Circuit İnterrupters
ANN	:	Artificial Neural Networks
ARIMA	:	Auto-Regressive İntegrated Moving Average Model
BNN	:	Bayesian Neural Network
CNN	:	Convolutional Neural Network
ConvLSTM	:	Convolutional LSTM
DWT	:	Discrete Wavelet Transform
FFT	:	Fast Fourier Transform
GAN	:	Generative Adversarial Network
GFPD	:	Ground Fault Protection Devices
GFDI	:	Ground-Fault Detection İnterrupters
GWO	:	Grey Wolf Optimization
KNN	:	K-Nearest-Neighbors
LCAD	:	Local Context-Aware Detection
LR	:	Logistic Regression
LSTM	:	Long Short-Term Memory
LTF	:	Linear Transfer Functions
MCD	:	Minimum Covariance Determinant
MLP	:	Multilayer Perceptron
MPPT	:	Maximum Power Point Tracker
MSE	:	Mean Square Error
NEC	:	National Electrical Code
OCPD	:	Overcurrent Protection Devices
PV	:	Photovoltaic
RBF	:	Radial Basis Function
RCMU	:	Residual Current Monitoring Unit
RD	:	Robust Distance
RH	:	Relative Humidity
RMSE	:	Root Mean Square Error
SSTDR	:	Spread Spectrum Time Domain Reflectometry

SVM	:	Support Vector Machine
SVR	:	Support Vector Regression
TDR	:	Time Domain Reflectometry

LIST OF TABLES

LIST OF GRAPHICS

LIST OF FIGURES

PREFACE

During this thesis's preparation and writing process, I would like to thank my esteemed Asst. Prof. Dr. Ercan Aykut.

I want to thank the thesis's jury members for their help managing this thesis and taking it forward with their valuable comments and suggestions throughout the process.

For his contribution to the thesis, Dr.Lecturer Ahmed A. A. Solyman I want to express my sincere gratitude to the Istanbul Gelişim University Electrical and Electronics Engineering Department staff, and the Institute of Science.

Finally, I would like to thank my family in particular (My Mother) and my friend (Uncle Raad Al-Shammary) for their support, and friends, Engineer Raad Al-Ezzi and Engineer Qusai Habib Al-Aboudi, for their support me during the study phase.

INTRODUCTION

In recent years, there has been a rapid growth in renewable energy, including power plants, which is expected to improve the production of clean, low-cost energy and drive economic growth (Vlaminck et al., 2022). One important issue in this field is detecting and identifying abnormal patterns in solar systems, particularly photovoltaic (PV) components (Cespedes et al., 2022; Sajun et al., 2022). Using big data and data-driven approaches, such as convolutional neural networks and deep learning systems, can effectively detect and prevent these abnormalities. These machine-learning approaches have proven to be accurate in many cases.

Solar photovoltaic (PV) systems often experience a range of abnormalities that can affect their performance (Lin et al., 2022; Meribout et al., 2023), including internal and external faults. Internal faults can result in zero power production during daylight hours, including component failure, shading, inverter shutdown, system isolation, and inverter maximum power point issues. External factors can hinder power generation, even if not caused by the PV system.

Dust, humidity, shading, and temperature are measured to be the most influential external factors in the production of PV systems. There have been numerous data science projects aimed at addressing these anomalies, including the use of artificial neural networks (ANN) for solar equipment modeling. ANN might save time and money by avoiding difficult mathematical techniques and needing fewer diagnostic procedures to establish input/output correlations.

One study (Elsheikh, Katekar, et al., 2021) used an algorithm of NN with long short-term memory (LSTM) to forecast the yield of solar stills, which can recall patterns and anticipate long-term time-series behaviors. Another study (Elsheikh, Panchal, et al., 2021) provided methods based on artificial intelligence for predicting the water output of solar distillers, incorporating a moth-flame optimizer and an LSTM model. Compared to the solitary LSTM model, the optimized LSTM model outperformed it.

Convolutional LSTM (ConvLSTM) is an effective hybrid model that combines LSTM and a convolutional neural network (CNN), was proposed by (Ibrahim et al., 2020).

It exhibited accurate prediction results with reduced latency, hidden neurons, and computational complexity. The application of deep learning techniques in various industries, such as data mining, medicine, agriculture, and wind and solar energy production, has been examined in other recent research (Aslam et al., 2021; Ibrahim et al., 2022).

CHAPTER ONE

PURPOSE OF THE THESIS

1.1. Literature Survey

Studies have looked at various methods for spotting irregularities in PV power networks. Authors (Branco et al., 2020) looked into various techniques for locating and categorizing irregularities in PV systems, including the auto-regressive integrated moving average model (ARIMA)(Deif et al., 2021), K Nearest Neighbors (KNN) classification, neural networks, and support vector machines (Hammam et al., 2022).

The authors (De Benedetti et al., 2018) developed a strategy for detecting and predicting abnormalities in PV systems and performing maintenance. The model is based on an ANN that predicts AC power generation using temperature and solar irradiance data from PV panels. Reference (Natarajan et al., 2020) presents a new method for identifying faults using thermal image processing and a support vector machine (SVM) tool that classifies characteristics as faulty or non-defective.

A model-based method for identifying irregularities in the PV plants' DC section and brief shadowing is presented in Reference (Harrou et al., 2019). The procedure comprises developing a design founded on the model of a one-diode to characterize typical PV system performance and generate fault detection residuals. The residuals are then subjected to a one-class SVM (1-SVM) to detect errors.

According to reference (Feng et al., 2020), Sundown is a method that can identify individual panel failures in solar arrays without using sensors. It works by analyzing the interactions between nearby solar panels and detecting deviations from predicted behavior.

The model can simultaneously handle several issues in multiple panels and classify abnormalities to discover possible causes, such as leaves, snow, electrical faults, and dirt. A brand-new tool called ISDIPV is presented in reference (Sanz-Bobi et al., 2012) for locating and analyzing faults in PV solar power systems. It comprises three primary parts data gathering, anomaly detection, and performance deviation diagnostics.

Reference (Zeng et al., 2022)used multilayer perceptron neural network models and linear transfer functions (LTF) to model average performance. The study suggested a data-driven method employing PV string currents as indicators to detect and categorize anomalies in PV systems. The proposed method for anomaly detection featured two steps {Global Context-Aware Anomaly Detection (GCAD) and Local Context-Aware Detection (LCAD)}, and it leveraged unsupervised machine learning techniques.

In reference (Mulongo et al., 2020), using generators as power sources for TeleInfra base stations were studied, and anomalies in fuel consumption were identified using reported data. Four classification approaches - KNN, logistic regression (LR), multilayer perceptron (MLP), and SVM - were used to identify anomalies in gasoline consumption patterns by learning the patterns using pattern recognition. The findings demonstrated that MLP was the most successful measuring and interpretation method.

Using KNN and 1-SVM for anomaly detection, a unique method of PV system monitoring is introduced in reference (Benninger et al., 2019). These self-learning algorithms greatly minimize measurement labor while enhancing the accuracy of defect monitoring. In reference (Benninger et al., 2020), a multilayer perceptron and the K-Nearest-Neighbors method are used to analyze data from a DC sensor and distinguish between different electrical current characteristics. A sensorless method for fault detection in PV plants is proposed in Reference (Firth et al., 2010), and it is based on the sharp fall in current between two MPPT sampling points.

Simulations were run to show that anomalies might be found in various settings, regardless of brightness and irradiance levels. In reference (Balzategui et al., 2021), a framework for detecting anomalies in monocrystalline solar cells is proposed. The system is divided into two parts:-

Part 1: Generative Adversarial Network (GAN) anomaly detection model is used, which can spot aberrant patterns using only perfect training examples.

Part 2: After detecting the anomalies, they are utilized to generate features that are used to train a fully convolutional network in a supervised manner.

In reference (Wang et al., 2022), a technique is proposed for real-time analysis of aerial thermography video streams. The method employs a combination of image processing and statistical machine learning techniques to detect and localize abnormalities in photovoltaic (PV) images using resilient principal component analysis (RPCA).

The method also includes post-processing techniques for image segmentation and noise reduction. Energy yield data are evaluated using various models in reference (Hempelmann et al., 2020). It is discovered that anomalous ensembles, proximity-based models, linear models, probabilistic models, and NN have the greatest detection rates.

Reference (Iyengar et al., 2018) introduces SolarClique, a data-driven approach for detecting irregularities in the electricity output of solar facilities, which does not require sensor equipment for fault/anomaly detection and relies only on the array's output and the output of adjacent arrays for operational anomaly detection. Reference (Tsai et al., 2020) suggests an anomaly detection method using a semi-supervised learning model to predict solar panel settings to avoid situations where the solar panel cannot produce electricity due to equipment degradation. This method uses a clustering model to filter normal behaviors and an Autoencoder model for neural network classification.

Using a variational autoencoder model with a decoder and encoder parameterized by recurrent NN (RNN) to capture the progressive relationship of time-series data, Reference (Pereira & Silveira, 2018) describes a comprehensive, unsupervised, and scalable method for detecting anomalies in offline and live time-series data. The study shows that the model can effectively identify abnormal configurations by utilizing probabilistic restoration measures such as anomaly scores.

Reference (Kosek & Gehrke, 2016) proposes a new approach to detecting cyber-physical intrusions in smart grids by utilizing an ensemble model incorporating non-

linear regression models and anomaly scores. This technique aims to enhance the accuracy of anomaly detection in smart grids.

In (Rossi et al., 2016), the authors describe the employment of an unsupervised collective and contextual detection algorithm to monitor the data flow of a major Czech Republic energy dispenser. The method employs clustering silhouette thresholding in combination with category clustering and conventional item-set mining techniques to detect anomalies.

Reference (Toshniwal et al., 2020) overviews many anomaly detection approaches, including graph analysis, closest neighbor, clustering, statistical, spectral, and information-theoretic methods. The input data, kind of anomalies, output data, and domain expertise determine the optimal AD method.

1.2. Problem Statement

Monitoring tools are needed to effectively operate photovoltaic (PV) systems, as they can help identify and address issues that may affect the system's performance and safety. Online monitoring can provide plant operators with important information for managing the plant and integrating it into a smart grid. Failure to detect problems with PV arrays can lead to reduced power generation and fire hazards. Early detection of anomalies on solar panels can help prevent power loss and ensure the panels' continued performance and safety. Therefore, it is important to have efficient and accurate methods for detecting anomalies in PV systems.

1.3. Aims and Objective

The following are some of the key Objectives of this thesis.

1. This study will investigate various anomaly detection models and conduct comparison tests to evaluate their precision and performance with optimized hyperparameters.

2. This study aims to define and quantify the external and internal variables that cause anomalies in PV power plants, their impacts on model accuracy, and the link between these factors and anomaly identification.

1.4. Thesis Questions

Along with the previous review of the related works, problem formulation, and the objectives of this study, we might summarize the main thesis questions by the following:

1. What is the best model for nominal prediction?.
2. What are the optimum hyperparameters for prediction models ?.
3. What is the correlation between features used in our study ?.

1.5. The Proposed Model

Based on the performance of many machine learning models, this thesis will investigate and show the best model that can accurately identify anomalies in the PV system. The effectiveness of machine learning models in identifying abnormalities will be evaluated by utilizing the correlation coefficients among plants' external and internal feature characteristics.

The data utilized were collected over 34 days at 15-minute intervals at two Indian plants for solar power (The first is in Gandikotta, Andhra, while the second is near Nasik, Maharashtra).

To assess the generation rate and identify any potential irregularities, twenty-two inverter sensors were installed at the plant and inverter levels for each plant. The sensors measured the AC and DC powers and the internal factor. Additionally, the inverter monitored the module temperatures, ambient conditions, and irradiance for meteorological measurements at the plant level (These represented external influences that may cause anomalies).

1.6. Thesis Conclusions

Successfully identified equipment failure and underperformance events with a rule-based method and linear/nonlinear modeling of the relationship between

irradiance, temperature, and DC power. This approach can be useful for real-time condition monitoring and fault detection.

1.7. Thesis Organization

The thesis is composed of five chapters.

Chapter One	:	Theoretical Background
Chapter Two	:	Purpose of the Thesis
Chapter Three	:	Data Exploration and Preparation
Chapter Four	:	Methods and Results
Chapter Five	:	Conclusions and Future Work

CHAPTER TWO

THEORETICAL BACKGROUND

This chapter reviews the research on the numerous fault types that can occur in solar PV arrays, the protection holes in traditional fixes, and the methods for fault detection, classification, localization, and protection. These current approaches' advantages and disadvantages are also investigated.

2.1. Overview of Solar PV Array Faults

The PV system can be affected by various defects and those in the PV utility grid, power conditioning unit, and array. The thesis specifically concentrates on the PV arrays that demonstrate voltage (I-V) characteristics vs. limited and non-linear current, setting them apart from conventional AC or DC power sources. As a result, PV array failures need to be carefully assessed and given further thought, according to Figure 1.

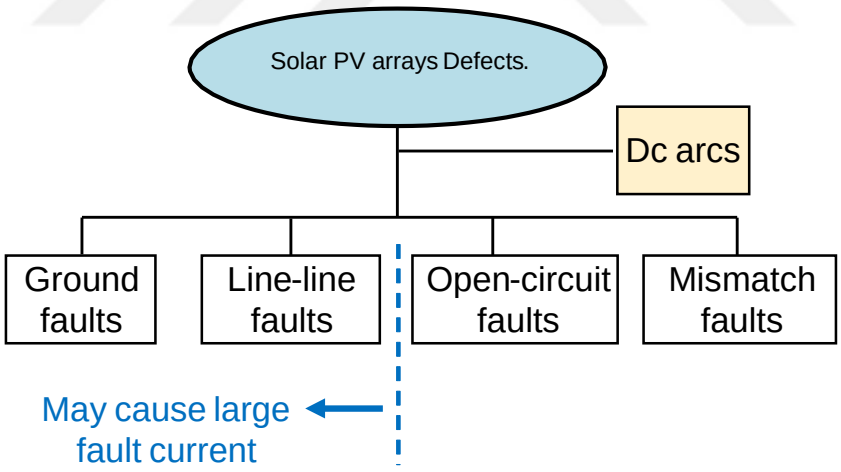


Figure 1. Frequent failures in PV arrays (DC line)

Solar PV arrays are prone to a variety of issues. This thesis assumes that the PV array is the only potential source of fault current due to the galvanic isolation typically provided by PV inverters between the PV arrays and utility grid. Ground and line-line faults pose the greatest risk for significant fault current flow among these faults (Alam

et al., 2015). Without appropriate fault detection, these faults can be the origin of dangerous DC arcs and maybe lead to fire threats in the PV array (Brooks, 2011). Furthermore, parallel or series DC arcs may develop because of one of these four fault classes (Yuventi, 2013), with series arcs resembling introduced variable resistance, making detection or extinguishing challenging (Spooner & Wilmot, 2008).

2.2. Protection Gap of Conventional Solutions

The National Electrical Code (NEC) specifies that traditional fault protection and detection relies on overcurrent protection devices (OCPD), ground-fault detection interrupters (GFDI), and Ground Fault Protection Devices (GFPD) (Association et al., 1915). The fundamental capabilities and limitations of these protections are outlined below.

2.2.1. Ground Fault Detection Interrupters

GFDI or GFPD are widely used to interrupt and detect ground faults in PV arrays of grounded PV systems (J. C. Wiles, 2012). The PV system is called grounded when the negative conductors are deliberately grounded, as depicted in Figure 2.

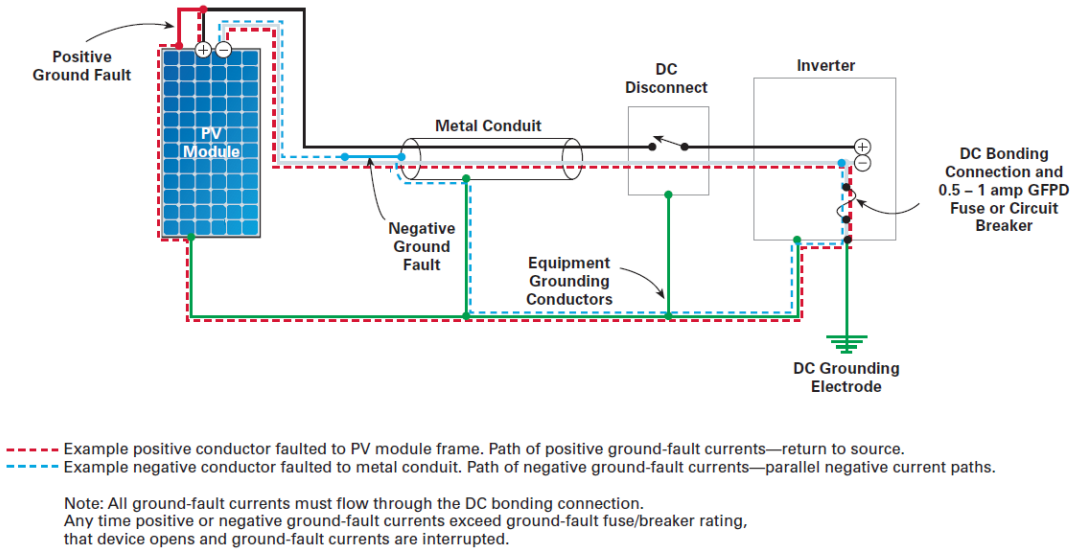


Figure 2. GFDI model in the grounded PV systems (J. Wiles, 2008)

Table 1. Maximum permissible detection range for ground current (Association et al., 1915)

DC power in KW	Maximum permissible detection range for ground current (Amperes)
0 – 25	1
25 – 50	2
50 – 100	3
100 – 250	4
< 250	5

All non-current-carrying conductive parts must be grounded, including conductor enclosures, equipment, and metallic module frames (Association et al., 1915; Zhao, 2011).

In PV inverters, a tiny fuse (within range 1A) is often incorporated to measure the ground-fault leakage current and diagnose ground faults (Inverters & Converters, 2010); the GFDI parameters must correspond to Table 1. in accordance with the standard UL 1741.

As seen in Figure 2. Regardless of whether the ground fault is positive (indicated in red) or negative (indicated in blue), the ground-fault current within the closed fault route will always return through the GFDI. If the fault current is sufficient, the GFDI (for example, a fuse) will explode. The PV inverter will then turn off, de-energizing the Solar array until an open circuit is achieved.

The described (Brooks, 2011) Bakersfield PV fire risks are believed to be generated by "blind zones" of GFDI, which have been discovered (Flicker & Johnson, 2013) in the condition of double-ground fault. Two successive ground faults occurred with the Bakersfield fire's twin ground fault. As seen in Figure 3. the original ground was situated between the ground and the negative conductor. The PV system masked the failure since the fault current stayed below the GFDI setting.

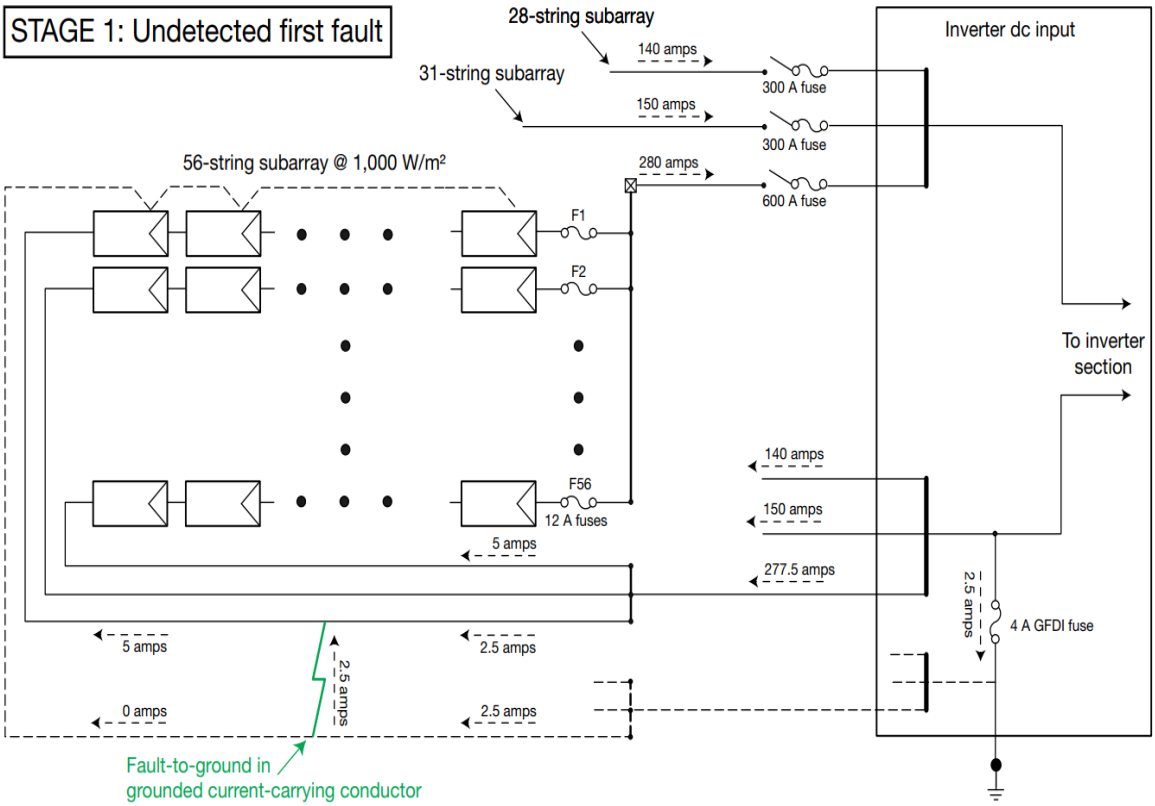


Figure 3. The primary hidden ground fault is: Relation between the equipment grounding conductor and a grounded current-carrying conductor (Brooks, 2011)

As illustrated in Figure 4. the second ground fault occurred between the conductor responsible for carrying ungrounded current (i.e., the positive conductor) and the conductor responsible for grounding the device. The inverter's GFDI fuse quickly blew due to the massive ground-fault current that resulted from this. Regrettably, the primary and secondary fault sites impeded the fault path, enabling the large fault current to run continually for a prolonged duration. Owing to the fault current's high heat output and exceeding the conductors' current rating, the fault current caused fire danger and insulation failure.

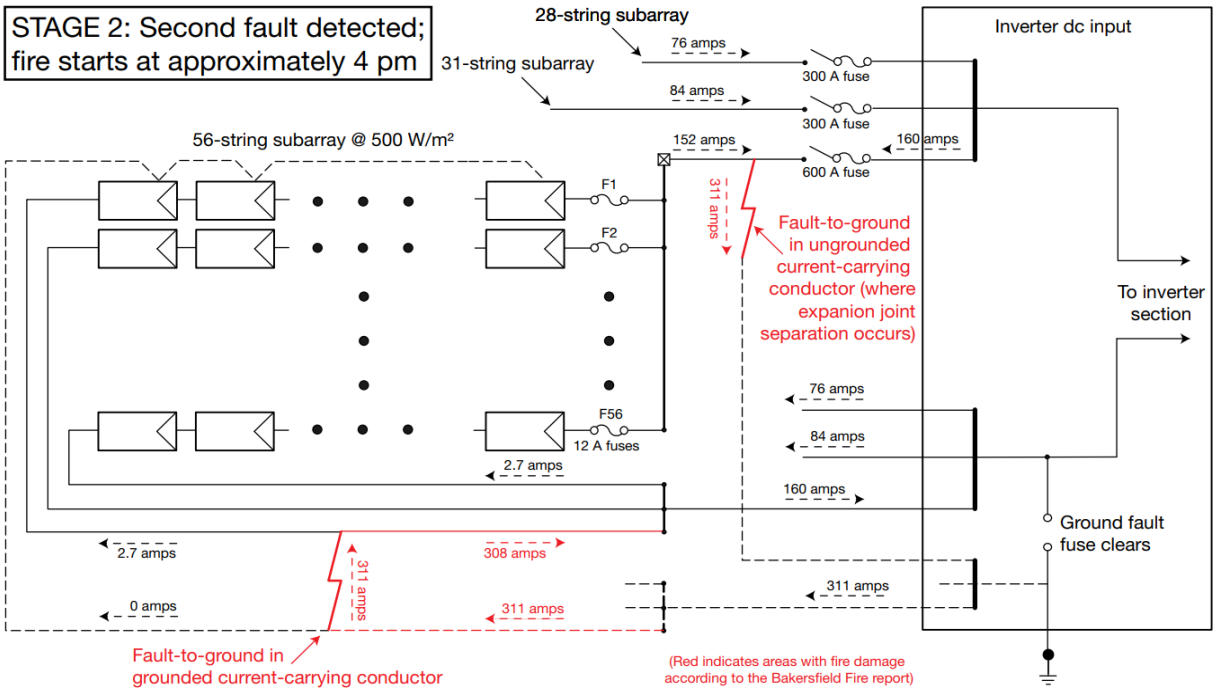


Figure 4. The secondary hidden ground fault

Relation between the equipment-grounding conductor and ungrounded current-carrying conductor (Brooks, 2011)

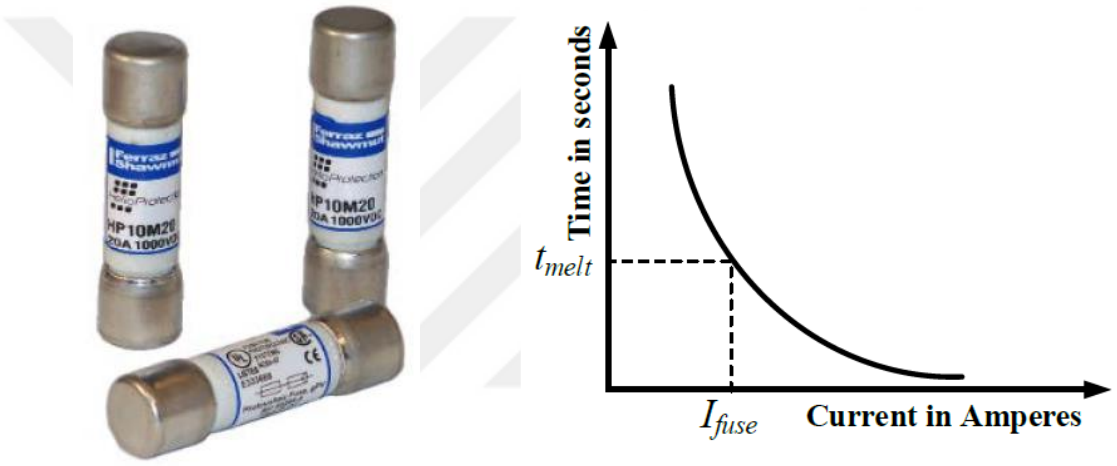
Double-ground faults, a potentially lethal fault situation, have also been implicated in other fire events, including the 2011 Mount Holly, North Carolina, solar PV fire (Association et al., 1915). The initial ground fault was not detected in this instance because the current level was much below the GFDI's rating. The subsequent second ground fault activated the GFDI. The ground faults could not be eliminated because the blocked channel permitted the continuous passage of ground-fault current. In the end, there was a fire.

2.2.2. Devices for Overcurrent Protection

The National Electrical Code (NEC) of the United States mandates the installation of overcurrent protection devices (OCPD) in series with each PV string (Association et al., 1915) to safeguard modules and cables from fault-induced overcurrent. Under standard test conditions, the OCPD's rated current must be at least 156 percent of the PV modules rated short-circuit current (ISC) (Association et al., 1915). It must also adhere to UL standard 2579 (Fuses—Part, 2010) and European IEC

standard 60269-6 (Fuses—Part, 2010). In accordance with IEC standard 60269-6 (Fuses—Part, 2010), the non-fusing current (I_{nf}) of PV fuses must be 1.13 In (at least 1.76ISC). The precise comparison between the NEC, IEC and UL criteria for selecting PV fuses is shown in (Ziar et al., 2014).

Moreover, as shown in Figure 5. the fuse has non-linear melting characteristics (Current vs. Time needed for melting fuse components). Often, a higher current than Inf implies a speedier melting process. For example, it may take several hours to melt a fuse if the current is slightly more than Inf. If the PV fault current remains below Inf, the fuse safety gap exists because it stops the fuse from blowing.



(a) The fuse of Solar PV (Guide, 2014).

(b) Relation between current and time

Figure 5. The curve of fuses shows melting time and current.

Given that solar PV arrays is the sole source of fault current, characteristics such as maximum power point tracker (MPPT), low solar irradiance, high fault resistance, and small-mismatched fault site may significantly lower the amount of backed current into the faulty string (I back) (Zhao, De Palma, et al., 2012). Hence, the fuse protection gap may exist if the fault current maintains beyond Inf.

2.2.3. Summary

OCPD and GFDI have a protection gap when specific fault situations occur within the PV array. Due to this, the flaw may not be adequately repaired, and it may continue undetected until the PV system fails.

Many ways to close the protection gap created by traditional protection and fault detection systems were suggested. These fixes might rely on the isolation of transformers in PV inverters and the grounding of PV systems. Figure 6 shows several present classifications, fault detection, protection, and localization approaches to PV arrays that have been created to boost the reliability, safety, and efficiency of solar PV systems. The remainder of this chapter explores the advantages and disadvantages of each present choice for each category.

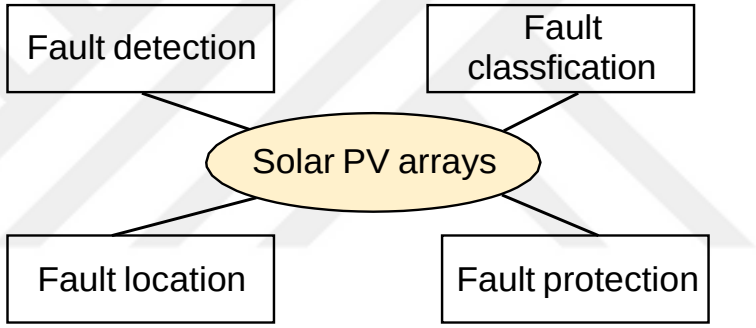


Figure 6. General ideas for enhancing PV-system dependability and protection

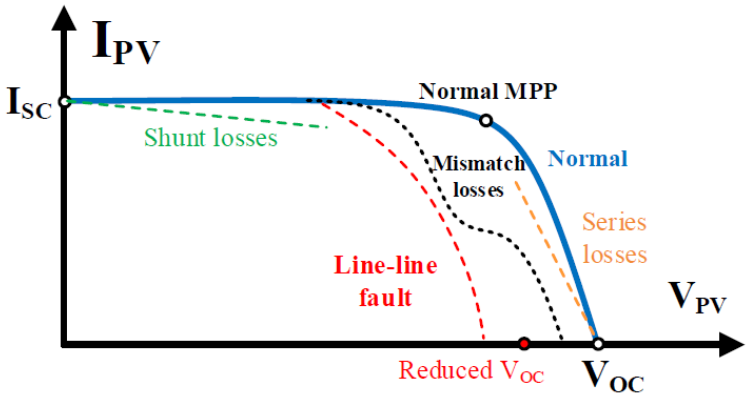
2.3. Current Solutions for Fault Detection Classification and Location

The phrase "an indication that something is going wrong in the monitored system" defines fault detection (Gertler, 1998). Fault classification may automatically determine the fault type in addition to fault detection. Fault location can approximate the cable fault area to expedite system recovery and aid in the search for the problem. This makes fault detection, classification, and placement essential for tracking and identifying unanticipated issues in solar PV systems. A brief discussion and summary of existing options (Table 2.).

Table 2. Overview of current PV fault detection, classification, and location

Fault Type	Method	Fault Detection	Fault Classification	Fault Location
Ground faults	GFDI	√	x	x
	Insulation Resistance monitor	√	x	x
	RCD	√	√	x
	TDR	√	√	√
	SSTDR	√	√	√
Line-line faults	I-V Curve Analysis	√	x	x
	Statistical Method	√	x	x
	OCPD	√	x	x
	RCD	√	x	x
	TDR	√	√	√
	I-V Curve Analysis	√	√	x
	Statistical Method	√	x	x
	Performance Comparison	√	x	x
	Capture Loss Analysis	√	√	x
	Performance Ratio	√	x	x
	Machine Learning	√	√	x
Open–circuit faults	I-V Curve Analysis	√	x	x
	Performance Comparison	√	√	x
	Performance Ratio	√	x	x
	Machine Learning	√	√	x
	Machine Learning	√	√	x
Mismatch faults	I-V Curve Analysis	√	x	x
	Performance Comparison	√	x	x

Fault Type	Method	Fault Detection	Fault Classification	Fault Location
DC-Arc faults	Ir Thermography	√	x	√
	Internal Series Resistance	√	√	x
	Machine Learning	√	√	x
	AFCI	√	x	x
	Machine Learning	√	x	x



Graphic 1. The current-voltage curve shows PV characteristics.

2.3.1. Quantitative – Model-Based Solutions

2.3.1.1. Current – Voltage curve Analysis

An I-V curve study can determine the effective points of a PV string, array, or module. As observed in Graphic 1, the I-V curve exposes crucial PV features. Hence, it is feasible to classify and identify PV issues using I-V characteristics, for example, decreased voltage, mismatch losses, shunt losses, series losses, and decreased current (Hernday, 2011).

2.3.1.2. Performance analysis of a PV system

Energy may be lost due to PV system failures, fire risks, and safety concerns. In-home PV systems in the UK, the energy lost due to failures has been examined and classified (Firth et al., 2010). It was determined that the yearly energy loss in PV systems due to faults might reach 18.9%. As a result, it is important to keep an eye on the performance of PV systems, create fault detection techniques, and research problem patterns.

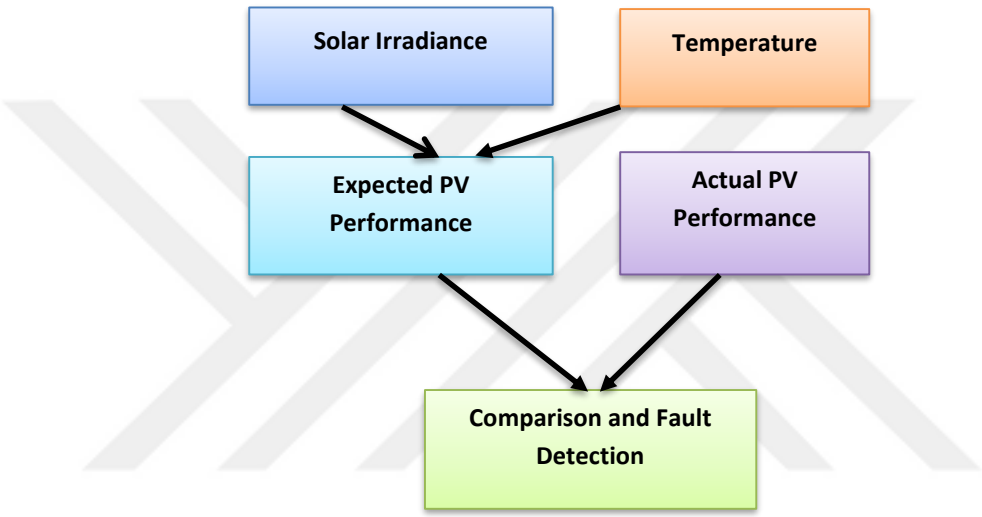


Figure 7. PV system performance analysis

Performance comparison contrasts real-time operating PV performance with simulated performance (Drews et al., 2007; Stellbogen, 1993). Performance comparison has recently been suggested as a method for finding flaws. In general, it contrasts the performance achieved with what was anticipated. Simple: a fault may be indicated by a large disparity in output performance as measured and produced. As seen in Figure 7. the usual performance evaluation includes defect detection, performance comparison, predicted and measured PV performance, meteorological data such as temperature and sun irradiation, and expected PV performance as a baseline.

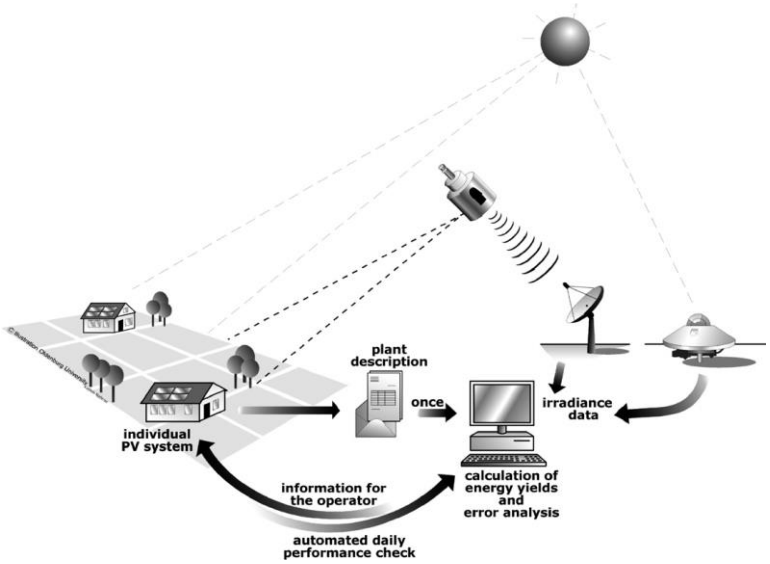


Figure 8. PV fault detection system overview utilizing performance comparison (Drews et al., 2007)

To prevent energy and related cost losses in PV systems provides, as one example, an automated PV performance comparison that tracks the difference between anticipated and actual energy output in real-time (Drews et al., 2007), as shown in Figure 8. The simulation model is fed data from the defect detection system, which includes data on ambient temperature and solar irradiance obtained from satellites (Psim). It keeps track of the AC output power (P actual) and contrasts it with Psim in the interim. Continuous total blackout, snow cover, energy loss, and changing energy loss are the four main categories of faults. The drawback is that it responds slowly because it continuously analyses PV power, equivalent to energy yield. For instance, this fault detection technique might require at least a day.

2.3.1.3. Performance Ratio (PR)

References (Blaesser & Munro, 1993; Commission & others, 1998; Haeberlin & Beutler, 1995) propose using Performance Ratio (PR) as a normalized metric to evaluate the energy yield of a PV system and assess its performance. PR is suitable for fault detection as it considers the system losses, regardless of the inclination and orientation of the panel. It is regularly expressed as Equation 1, where the final yield (Y_f) is divided by the reference yield (Y_r).

$$performance\ ratio\ = \frac{Y_f}{Y_r} \qquad (1)$$

Figure 9. This demonstrates the procedure for computing Y_f and Y_r in grid-connected PV systems, where Y_f represents the utility grid normalized AC energy output.

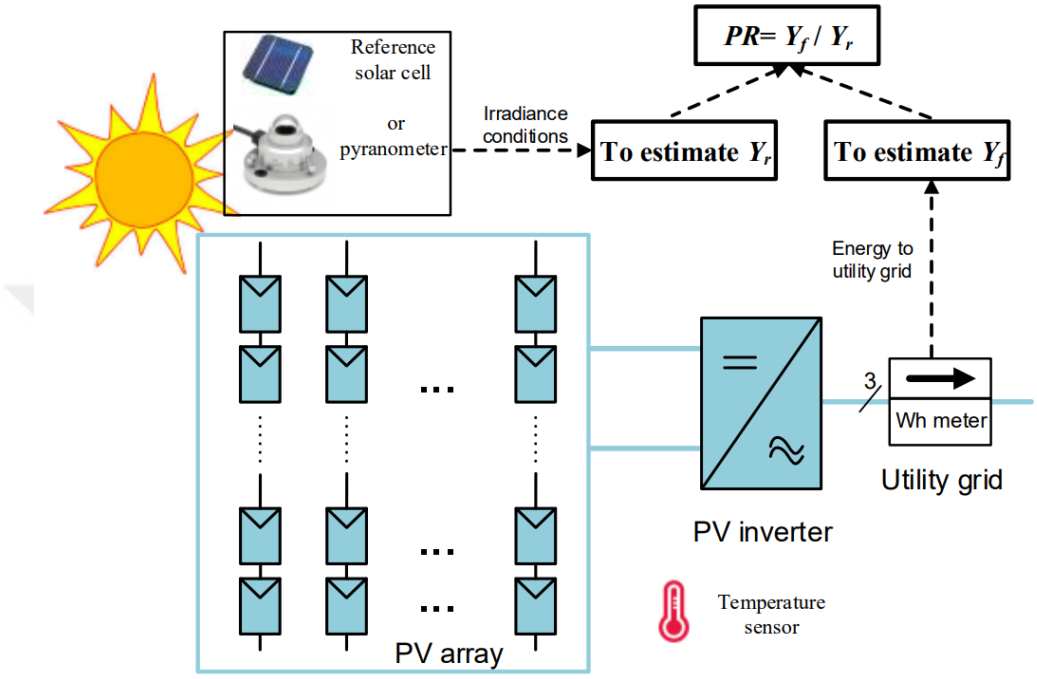


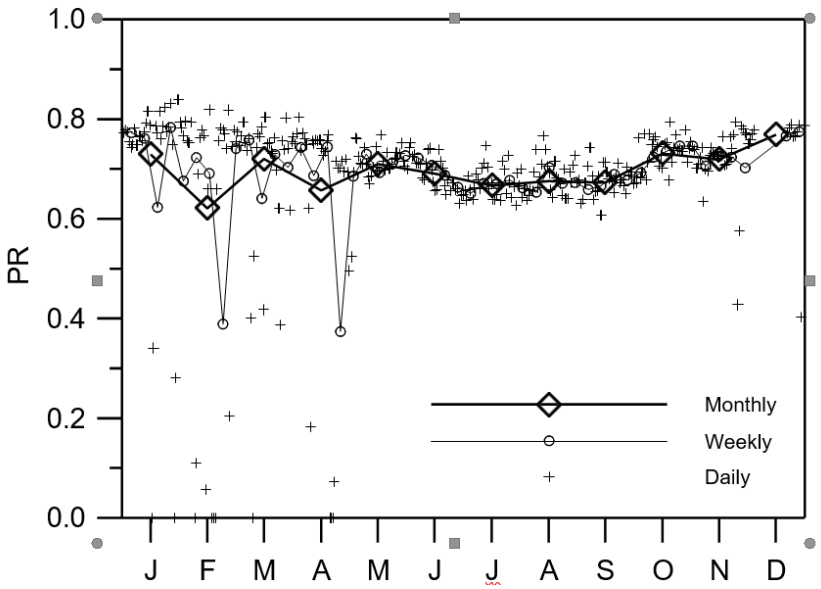
Figure 9. Calculate Y_r and Y_f for PV systems connected to the grid

It is described in Equation 2; in the given equation, E represents the AC power output, while P0 represents the nominal PV power.

$$Y_f = \frac{E}{P} \quad (kWh/kW)\ or\ (hours) \qquad (2)$$

Y_r stands for the normalized sun irradiation circumstances on the other side. It is described in Equation 3, where G_{STC} equals 1 kW/m^2 and H represents the total solar irradiance in-plane in (kWh/m^2).

$$Y_r = \frac{H}{G_{STC}} \quad (hours) \qquad (3)$$



Graphic 2. 2001 performance ratio graphics for photovoltaic systems (Marion et al., 2005)

PV system PR readings are typically reported monthly or annually (long-term). PR delivers improved short-term results, such as every day or every week, which can be employed in PV systems to detect defects. Under normal operating conditions, PR values for PV systems usually range between 0.65 and 0.8, as shown in Graphic 2. However, a much lower PR implies system defects or anomalies, such as partial shading, wrong system size, MPPT failures, inverter malfunctions, or PV array issues. The graph also demonstrates that shorter periods, such as daily, offer superior fault-detection resolution and faster response times than monthly or weekly data (Haeberlin & Beutler, 1995).

2.3.1.4. Analysis of Capture Loss

To circumvent the shortcomings of PR as a metric (Chouder & Silvestre, 2010; Commission & others, 1998) established the idea of system capture loss (L_c) to identify system-level abnormalities better. L_c refers to the operating losses of the PV array and is defined in Equation 4, where Y_A represents the daily energy output per KW of the installed PV array, and Y_r denotes the reference yield. In addition, L_c can be further subdivided into other capture loss (L_{cm}) and thermal capture loss (L_{ct}), as illustrated in Equation 5.

$$L_c = Y_r - Y_A \quad (4)$$

$$L_c = L_{ct} - L_{cm} \quad (5)$$

The measured capture loss (L_{cm}) should stay within the theoretical boundary for a normal PV array. Using Equation 6, where σ represents the simulated capture losses L_c sim standard deviation of, faults can be identified.

$$L_{c\,sim} - 2\delta < L_{c\,mes} < L_{c\,sim} + 2\delta \quad (6)$$

An approach for fault classification, which is an extension of (Chouder & Silvestre, 2010), was developed using power loss analysis. This approach relies on the variation between the observed and simulated parameters.

2.3.2. Solutions Based on Process – History

2.3.2.1. Statistical Methods

Based on energy generation, statistical techniques are suggested to identify PV system anomalies (Vergura et al., 2009). Explicitly, inferential and descriptive statistics are used for the measured energy production of each PV plant subarray. The results of experiments demonstrate that one can be wired incorrectly out of 22 typical PV panels using the suggested method.

Multivariate outlier rules have been presented to detect PV faults using the minimum covariance determinant (MCD) (Vergura et al., 2009). Specifically, the MCD is employed to compute the full distance (RD) based on multiple current and voltage measurements at different precise times for the PV modules. The fault detection criteria become simple: if the estimated RD exceeds a certain threshold, the PV module is at fault.

2.3.2.2. Machine Learning

Machine learning is a form of artificial intelligence that extracts information from a given PV data collection. Supervised learning techniques include a subset of machine learning. As seen in Figure 10, supervised learning techniques may make predictions by studying the system. This depends on a vast quantity of labeled data. In PV installations, numerous supervised learning models have been put forth. Artificial neural networks (ANN) have been created for monitoring PV health status (Riley &

Johnson, 2012), assessing PV performance under partial shade (Nguyen et al., 2009), and identifying short-circuit faults in PV arrays (Karatepe et al., 2011). The effects of soiling on massive PV arrays have been predicted using polynomial regression models and Bayesian Neural Network (BNN) (Pavan et al., 2013). The K-nearest neighbor, support vector machine (SVM), and decision-tree model are employed for PV defect detection and classification (Zhao, Yang, et al., 2012).

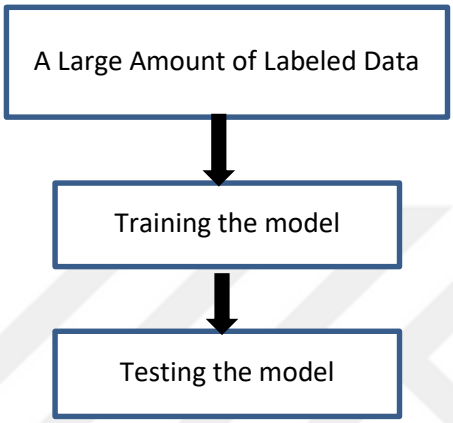


Figure 10. Supervised learning

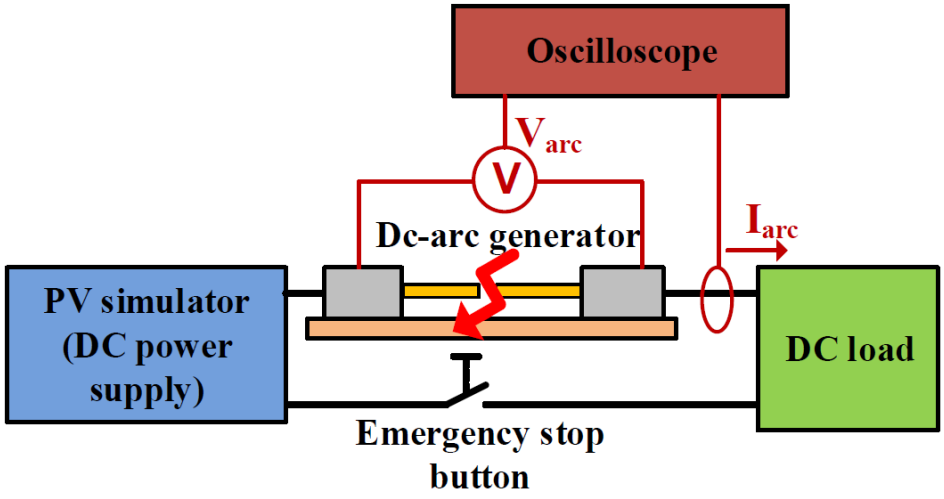
2.3.3. Solutions Based on Signal–Processing

2.3.3.1. Dc Arc – Fault Circuit Interrupter

The NEC 2014 Version (Association et al., 1915) specifies DC arc-fault circuit interrupters (AFCI) to cover the gaps in protection produced by conventional OCPD and GFDI. If GFDI and OCPD (Strobl & Meckler, 2010) do not remove the fault appropriately, a DC arc may arise in PV arrays. An electric arc is "an electrical breakdown of a gas that creates a continual plasma discharge, resulting from a current traveling through a normally nonconductive material such as air" (Yuventi, 2013). At the Northeastern University Power Electronics lab, a dc arc is generated using a dc arc generator and a photovoltaic (PV) simulator, as seen in Figure 11 (a). Figure 11 (b) displays the experimental configuration for the dc arc experiment. In the case of low-power, arc current Iarc is 3A, and arc voltage is 30V dc. To guarantee that the dc arc continues uninterruptedly, the PV simulator continues operating close to the maximum power point (MPP) in the arc fault. When the current and voltage ratings grow in actual

PV fields, dc arcs may have a significantly greater amount of power, which might swiftly spark fire dangers.

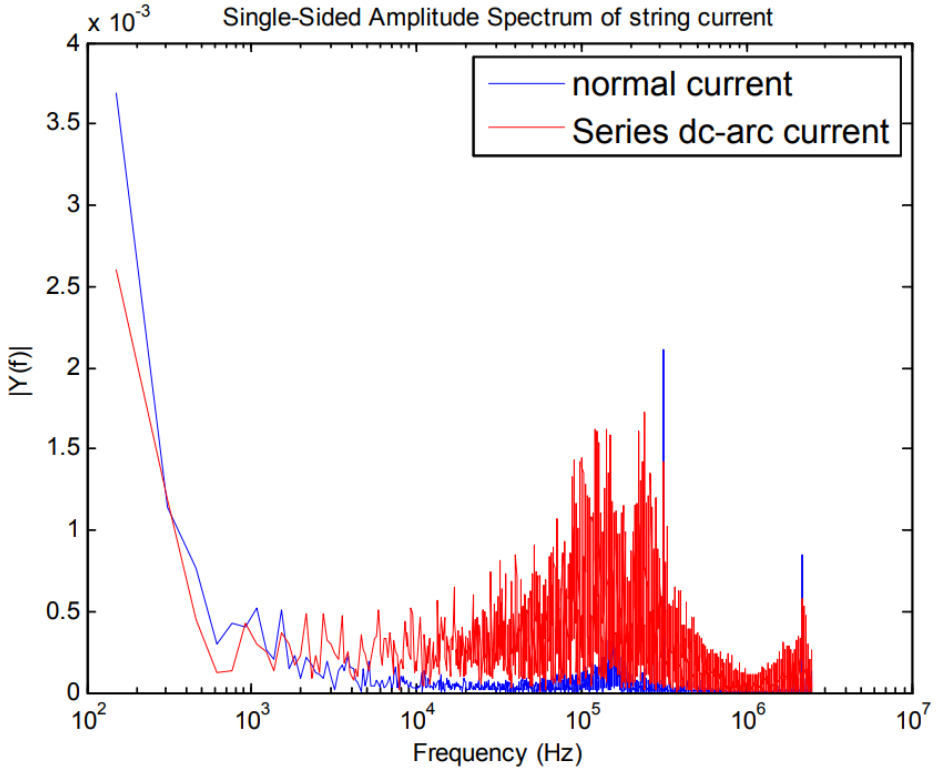
(a) A dc arc generator creates a dc arc.



(b) A diagram of an experimental dc arc setup.

Figure 11. Northeastern university dc arc testing configuration

A substantial quantity of AC noise will be produced in the dc arc, as seen in Graphic 3. which can be employed as a signature for defect detection. AFCI should identify and stop parallel and series arcing problems in dc PV sources and output circuits. The Fast Fourier Transform (FFT), The Discrete Wavelet Transform (DWT), and ANN have all been used to produce several dc-arc detection techniques.



Graphic 3. Frequency domain analysis for series vs. Normal dc-arc current.

2.3.3.2. Insulation Resistance Monitor

When a PV system is de-energized and not grounded, ground-fault protection is implemented via insulation resistance monitoring (Hernández & Vidal, 2009). The PV array's insulation resistance ranges from k to M (Ω), and when direct contact faults or insulation faults (such dangerous ground faults) happen, it dramatically drops. Hence, a ground fault can be recognized if the insulating resistance suddenly decreases. An insulation model for solar PV, parallel and series insulation resistance, PV working circumstances, and PV module leakage capacitance has been proposed (Hernández et al., 2010). Also, as PV module relative humidity (RH) or temperature rises, the experimental results demonstrate that a PV array's insulation resistance reduces.

A commercially available insulation resistance monitor is displayed in Figure 12. For ground-fault detection, continuously measures positive-ground and negative-ground insulating resistance v(Momoh & Button, 2003). Both electrified and de-energized PV systems can employ this method.

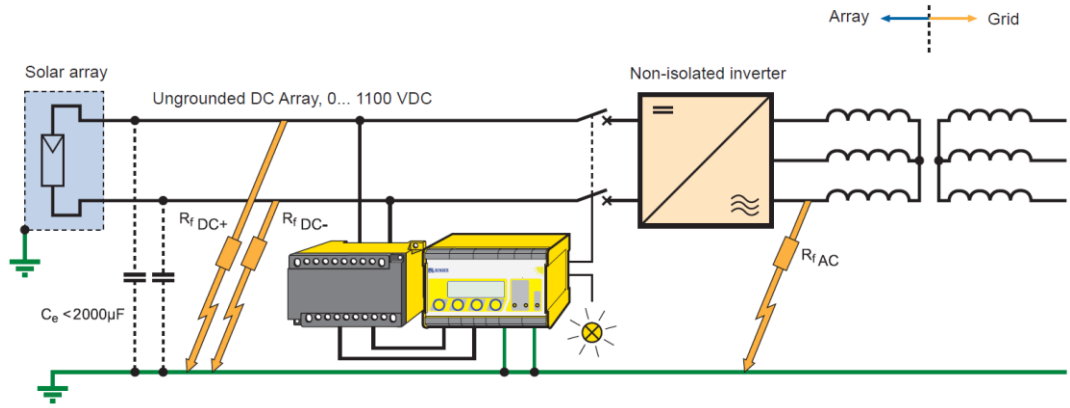


Figure 12. Insulation resistance monitoring system (Marion et al., 2005)

2.3.3.3. Residual Current Detector (RCD)

n a power system, the residual current detector (RCD) functions as a differential relay by measuring the current difference between the protected zone's input and output terminals (refer to Figure 13). If the difference in current reaches a certain level, an alarm is triggered to signal a problem. Nevertheless, the use of transformerless PV inverters may lead to a significant leakage current due to the parasitic capacitance of the PV array. This could cause unnecessary RCD tripping (Blackburn & Domin, 2015).

In power systems, the optimum protection method for generators, motors, reactors and transformers has been known as differential protection for over 50 years (Blackburn & Domin, 2015). The protected area's electrical parameters, usually the current entering and exiting, are monitored by differential relays.

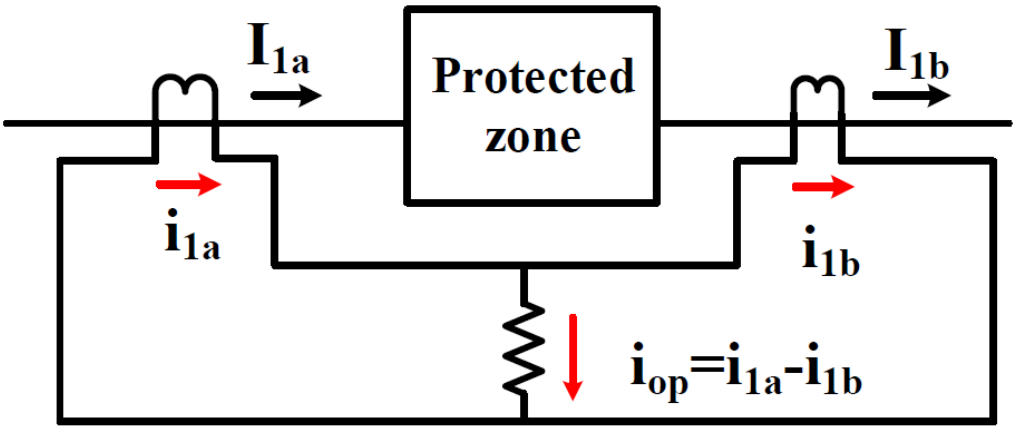


Figure 13. The development of fault-detecting differential relays (Blackburn & Domin, 2015)

The main currents entering and exiting the protected zone are I1a and I1b, as illustrated in Figure 13. Moreover, I1a and I1b are the corresponding secondary currents. The operational current of the relay, iop = i1a − i1b, separates them. A fault alert will be transmitted if the amplitude of |iop| exceeds "0" or a minimum limit.

A residual current monitoring unit (RCMU) or residual current detector (RCD) has been proposed for detecting defects in PV arrays, equivalent to the differential relay used throughout power systems, as shown in Figure 14.

RCD#1 or #2 might be a single PV string or a full PV array, and they both monitor the total currents entering and leaving the protected area. A defect will be identified once |iop| exceeds each threshold. For instance, String 1 has a ground fault at location G1 with ground fault current Ig. i1a i1b and ipos /= ineg. RCDs #1 and #2 will correctly identify the fault as well.

The RCD protection method has a limitation known as a "blind spot" when the protected area is not in contact with an external fault point. This can occur due to degradations within the protected zones and open-circuit or short-circuit problems. In these circumstances, it should be equal to ineg, and ipos should be close to zero.

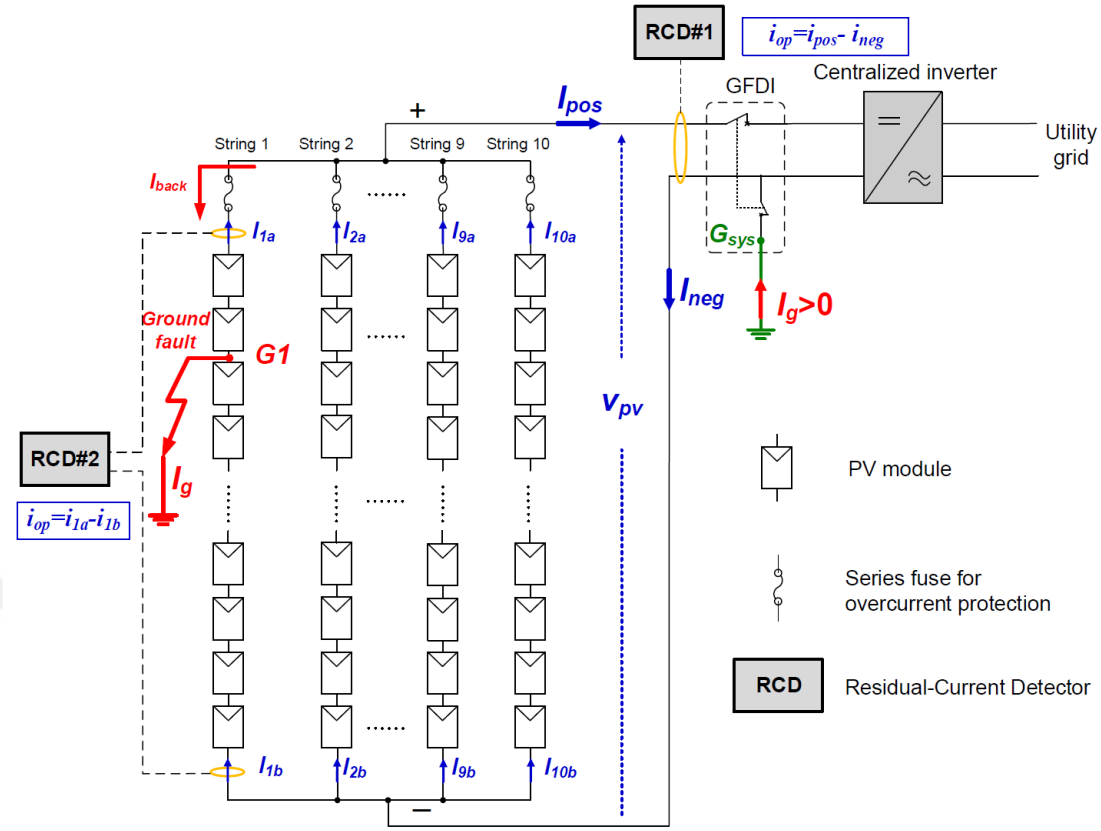


Figure 14. PV arrays employing RCD for protection.

2.3.3.4. Time Domain Reflectometry

Time Domain Reflectometry (TDR) is a measuring technique that involves injecting specific waveforms, such as an impulse or step signals, and analyzing the waveforms that reflect to determine the characteristics of an electrical wire or cable. TDR compares the reflections produced by the standard or specified impedance to those produced by the unknown line environment. TDR can therefore be utilized to diagnose the problematic cables or wires. In a large electrical system, TDR is useful as it can detect and classify faults and pinpoint the issue with precision.

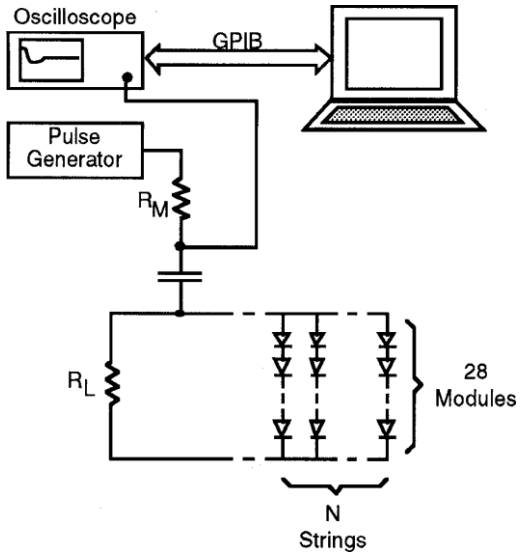
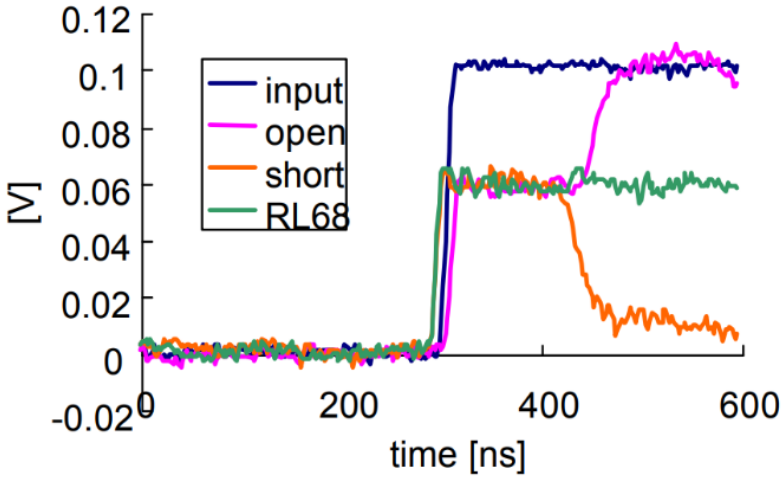


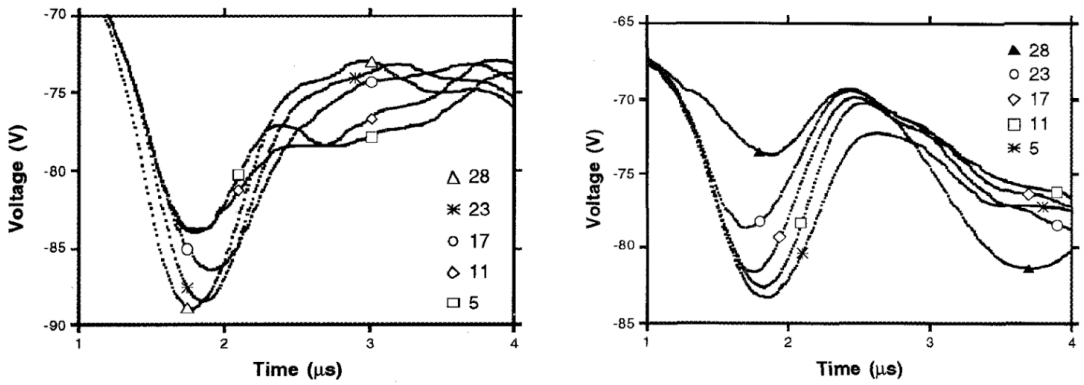
Figure 15. Diagram of the TDR layout in the PV region. (Schirone et al., 1994)



Graphic 4. Input and reflected waves with cable (Takashima et al., 2006)

The use of TDR for fault location in PV fields has been proposed in studies (Takashima et al., 2006, 2009). In Figure 15. the schematic diagram is displayed.

An example of this can be seen in Graphic 4. which displays a step input into a PV string and its corresponding reflected waveform. Waveforms change based on conditions such as resistance load, open circuit fault, and short circuit fault.



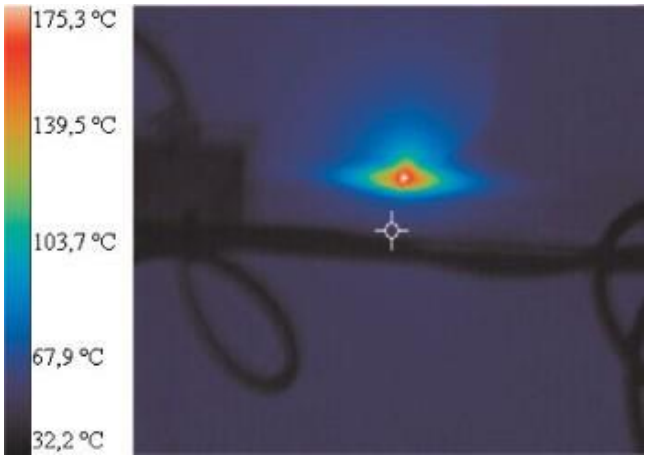
(a) TDR on open-circuit PV strings. (b) TDR on short-circuit PV strings.

Graphic 5. TDR on open/short circuit PV.

Graphic 5. İllustrates the TDR outcomes for short-circuit and open-circuit defects in PV strings using a negative input step. The different TDR responses can be beneficial in accurately identifying the fault location. Nevertheless, TDR has numerous shortcomings. The test PV system should first be down to avoid affecting the energy yield. Second, it is less effective at autonomous defect detection since it needs human input to monitor and understand the reflected waveforms.

2.3.3.5. Spread Spectrum Time Domain Reflectometry

Spread spectrum time domain reflectometry (SSTDR), a commercially available alternative to TDR, has been suggested to find aero plane wiring issues (Smith et al., 2005). SSTDR is an effective method for locating live wire faults due to its strong noise immunity and low-test signal levels. During ground-fault conditions, the autocorrelation peaks generated by SSTDR are significantly higher than the normal peaks, which is useful for detecting ground faults in PV systems. This justifies the defect detection algorithm put out in (Alam et al., 2013). SSTDR has yet to be suggested for locating or categorizing faults.



(a) IR thermal image shows a hotspot on a PV module's back (Muñoz et al., 2008)

(b) Camera for thermal imaging (Ancuta & Cepisca, 2011)

Figure 16. PV array utilizing infrared (IR) thermography for fault detection.

2.3.3.6. Infrared Thermography

Infrared (IR) thermography was utilized to detect mismatch problems in PV systems, such as PV module hot spots (Figure 16). As reported in studies, these can be the source of power loss and irreparable damage to the modules (Ancuta & Cepisca, 2011; Muñoz et al., 2008). While IR inspection is routine and periodic, it can be expensive if it necessitates more labor costs from PV maintenance staff throughout the modules' lifetime.

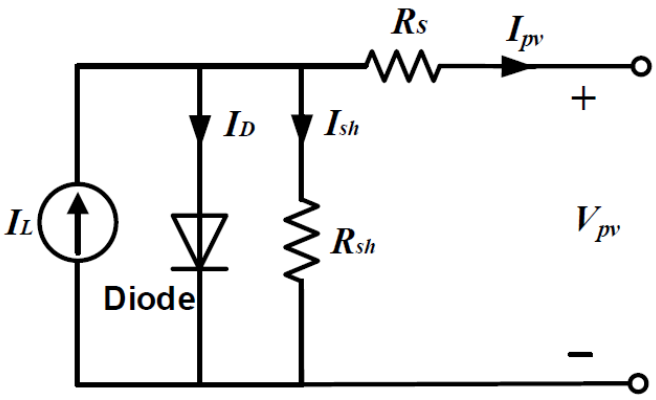


Figure 17. The one-diode model equivalent circuit

2.3.3.7. Internal Series Resistance

The frequently utilized one-diode model schematic design is represented in (Association et al., 1915) Figure 17. As reported, the circuit current equation is given by Equation 7.

$$I_{pv} = I_L - I_S\left[\exp\left(\frac{V_{pv}+I_{pv}R_S}{A\cdot k\cdot T\cdot N_S}\cdot q - 1\right)\right] - \frac{V_{pv}+I_{pv}R_S}{R_{sh}} \quad (7)$$

The PV module output current is represented by Ipv, while IL represents the light-generated current. IS represents the diode's saturation current, and Vpv is the module voltage. Rsh and Rs are equivalent shunt resistances and internal series, correspondingly. Additionally, q (1.38 10^-23 J/K) and k (1.38 10^-23 J/K) are the electron charge and Boltzmann constant, respectively. NS is the number of solar cells placed in series within the module, and T is the temperature in Kelvin of the PV module. Finally, A denotes the diode's ideal factor.

The module's series resistance (Rs) can detect PV module degradation internally. Either measurement made near open-circuit voltage (Kunz & Wagner, 2004) or I-V curve analysis (Sera et al., 2008) can be used to estimate the real Rs. Degraded PV modules typically show higher Rs values than the standard, a helpful attribute for problem detection.

2.4. Solutions based on Existing Fault Protection

To stop the defect and avoid damaging the PV components, fault protection devices can be activated by a fault signal once it has been identified. System protection aims to isolate any issues within a power system swiftly, minimizing the impact on the rest of the system and preserving its integrity, as reported in reference (Blackburn & Domin, 2015). "fault protection" and "fault clearing" can be used interchangeably to refer to this process.

In PV applications, isolating any fault areas within the solar PV arrays is recommended to minimize their impact on the rest of the system. Table 3. provides an overview of the fault protection techniques currently used for PV arrays. Blocking diodes, GFDIs (ground-fault detecting interrupters), and OCPDs are passive approaches (Zhao, De Palma, et al., 2012). On the other hand, active solutions depend on semiconductor switches circuit breakers, or contactors, to isolate and de-energize the damaged PV components after using more sophisticated sensor circuitry to detect the problem (Luebke et al., 2016; Pozsgay, 2016; Schripsema, 2014).

Table 3. An overview of current PV fault protection measures

Protection type	Method	Cost	Complexity	Limitation
Passive	GFDI	Low	Low	Blind spots
	OCPD	Low	Low	Blind spots
	Blocking diode	Low	Low	Low reliability
Active	AFCI	Medium	Medium	High-frequency interference from power condition units
	Contactor + protection relay	Medium/high	Medium	Bulky and costly for high voltage application
	Semiconductor switch + protection relay	Low / medium	Medium	PV – module level only

Passive protection devices have clear limits. We have talked about the "blind spots" of OCPD and GFDI before. Blocking diodes can interfere with OCPD's proper

operation and are not a replacement for them (Zhao, De Palma, et al., 2012). Active fault prevention devices are advantageous to passive ones in increasing fault protection. However, they rely heavily on fault detection techniques. Hence, fault detection approaches that can send a precise and timely trigger signal to active protection systems remain crucial.

An evaluation of the literature on current fault detection, localization, protection, and classification strategies for solar photovoltaic (PV) arrays is provided in this chapter (DC side). Also, the advantages and disadvantages of current techniques are investigated and contrasted. Methods for defect identification, classification, and location can generally be categorized into three groups: methods based on quantitative models, methods based on process histories, and methods based on signal processing. GFDI and OCPD are frequently employed in PV installations for fault protection and are mandated by the NEC. However, its flaws and restrictions have been found, which could result in the previously mentioned fire risks.

Several approaches have been suggested utilizing different methodologies to deal with this issue. However, drawbacks can reduce their usefulness in practical PV applications. Therefore, better fault detection techniques are urgently needed to protect PV systems from the risks associated with faults.

2.5. Support Vector Machine (SVR)

The supervised learning algorithm, Support Vector Machines (SVMs), can be applied to classification or regression applications. They operate by doing regression with the least error or by locating the hyperplane in a high-dimensional space that maximum separates various classes. When the number of features is substantially higher than the number of samples and the data is noise-free or has minimal noise, SVMs are extremely helpful. They work best when the classes are well-separated and in high-dimensional spaces. SVMs can be employed with various kernel functions, which enables them to adapt to complex non-linear relationships in the data. This is one advantage of SVMs. The "kernel technique," which enables them to operate in a higher-dimensional training dataset without explicitly computing the coordinates of the data in that space, may also be used to manage high-dimensional data.

Vapnik and his colleagues presented the support vector regression (SVR) method (Deif, Solyman, et al., 2021), and the development route was established by expanding the SVM classification algorithm (Hammam et al., 2022). As a supervised machine learning technology, SVR permits the prediction of continuous real-valued variables and is a machine learning regression technique for regression analysis (Deif, Hammam, Solyman, et al., 2021).

2.6. Support Vector Machine regression

In the support vector machine method, the model is first trained using a sample data set and then used to make predictions. Let the sample data be (x_i, y_i), where $i = 1, 2, \dots, n$. Here, x_i represents the input factor that influences the value of the cyber safety situation, and y_i represents the predicted situation value. The regression function can be expressed as (Deif & Hammam, 2020):

$$y_i = \omega \cdot \boldsymbol{x_i} + b \tag{8}$$

The formula includes the weighting matrix ω, the input vector x_i, the biasing direction b and the predicted regression value y_i. It also utilizes an ε -insensitive loss function.

$$|y - y_i|_\varepsilon = \begin{cases} 0, & |y - y_i| \le \varepsilon \\ |y - y_i| - \varepsilon, & other \end{cases} \tag{9}$$

If it is acceptable for the fitting error to pass ε, Equation (8) can be modified by introducing a relaxation factor, resulting in the following constraint:

$$\begin{aligned} min \frac{1}{2} = \| \boldsymbol{\omega} \|^2 + C \sum_{i=1}^{n} (\xi_i + \xi_i^*) \\ s.t.\ \ y_i - \boldsymbol{\omega} \cdot \boldsymbol{x}_i - b \le \varepsilon + \xi_i^* \\ \boldsymbol{\omega} \cdot \boldsymbol{x}_i + b - y_i \le \varepsilon + \xi_i \\ \xi_i, \xi_i^* \ge 0, i = 1,2 \dots \dots, n \end{aligned} \tag{10}$$

Where C is the penalty factor, and ξ_i and ξ_i^* are relaxation variables. By taking the derivative of the optimization problem specified in Equation 10, we can obtain the dual problem, as shown below:

$$\begin{gathered} min\, W(\alpha, \alpha^*) = -\varepsilon \sum_{i=1}^{l} (\alpha_i^* + \alpha_i) + \sum_{i=1}^{l} y_i(\alpha_i^* - \alpha_i) \\ -\frac{1}{2}\sum_{i,j=1}^{l} (\alpha_i^* - \alpha_i)\left(\alpha_j^* - \alpha_j\right)K\left(x_i, x_j\right) \\ s.t.\ \sum_{i=1}^{l} \alpha_i^* = \sum_{i=1}^{l} \alpha_i \\ 0 \le \alpha_i^* \le C, 0 \le \alpha_i \le C, i = 1,2 \ldots l \end{gathered} \tag{11}$$

where $\mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j})$ is the kernel function, and α_i, α_i^* are Lagrange multipliers. The regression function that results is:

$$y_i = \sum_{i=1}^{n} (\alpha_i^* - \alpha_i)K\left(x_i, x_j\right) + b \tag{12}$$

2.7. Support Vector Machine classification (SVM_C)

A brief overview of SVM classification will be provided in this section. SVMs are primarily used as binary classifiers, which means they can distinguish between two classes. Based on statistical learning theory, the primary objective of SVM classification is to identify a hyperplane that efficiently divides the positive (+1) and negative (-1) classes.

Consider the process of segregating $(\boldsymbol{x}_i, y_i)$ training data., where $i = 1,2, \ldots, m$, into two classes. The feature vector $\boldsymbol{x}_i$ belongs to the space $\mathrm{R}^n$and has n dimensions; during the class label $y_i$belongs to the set $\in \{+1, -1\}$. The generalized linear SVM aims to identify the best hyperplane for separating classes., $f(\boldsymbol{x}) = \langle \boldsymbol{w} \times \boldsymbol{x} > +b$, by solving the following optimization equation:

$$\underset{w,b}{Minimize} \frac{1}{2} \parallel w \parallel^2 \quad subject\ to\, y_i(w \cdot x_i + b) - 1 \ge 0 \tag{13}$$

SVMs aim to find the hyperplane in a high–dimensional space that maximally separates the positive and negative examples by solving an optimization problem with constraints. The orientation of the hyperplane is determined by the weight vector $\boldsymbol{w}$, and the distance of the hyperplane from the origin is determined by the bias term b. The optimum values for b and $\boldsymbol{w}$ can be calculated by minimizing the magnitude of $\boldsymbol{w}$ subject to the constraint that the distance between the hyperplane and the nearest examples is greater than or equal to some value (the margin). This constraint is

enforced using the Lagrange multipliers $\lambda_i (i = 1,2,\cdots,m)$, where m is the number of examples. The generated hyperplane has the fewest possible training mistakes.

$$f(x) = sign(\sum_{i=1}^{m} y_i \lambda_i < x_i \times x > + b^*) \quad (14)$$

sign(.) denotes a function and those $\boldsymbol{x}_i$ vectors for which $\lambda_i > 0$ are known as support vectors.

When it is not possible to define the hyperplane using linear equations, the data $\boldsymbol{x}$ can be

Linear, polynomial, radial basis function (RBF), and sigmoid kernel functions are the most often utilized. One is the RBF, commonly employed in SVM classification due to its remarkable nonlinear performance. The RBF equation is the following:

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = exp\left(-\frac{\|x_i - x_j\|^2}{\sigma^2}\right) \quad (15)$$

By incorporating the kernel function, the nonlinear SVM classifier might take the following forms:

$$f(x) = sign(\sum_{i=1}^{m} y_i \lambda_i k(\boldsymbol{x}, \boldsymbol{x}_i) + b) \quad (16)$$

2.8. Grey Wolf Optimizer (GWO)

The Grey Wolf Optimization (GWO) algorithm, a new meta-heuristic group intelligence optimization technique, was presented by Mirjalili et al. in 2014 (Deif et al., 2022; Deif, Hammam, Ahmed et al., 2021; Hammam et al., 2022). The meta-heuristic optimization technique known as Grey Wolf Optimization (GWO) was influenced by hunting strategy and the natural leadership structure of grey wolves(Zamfirache et al., 2022). The algorithm resembles the wolf pack's leadership structure, in which the alpha wolves take the lead, the beta wolves assist, and the delta wolves search for food. Each wolf in the pack is assigned a position and is rewarded based on their performance. GWO is based on these principles and uses a population of candidate solutions, called wolves, to search for the optimal solution. The wolves are assigned different positions in a pack, and their reward is based on their position

in the hierarchy. The GWO algorithm solves optimization problems with many variables and complex constraints. It is a fast, reliable, and efficient optimization algorithm that can be applied to various optimization problems.

This method optimizes the search by simulating the predatory behavior of gray wolves, which includes tracking, encirclement, chase, and assault. It has a rigid social structure where the top three wolves are regarded as the best, and the bottom three are referred to as ω. These wolves are positioned around the best wolves (α, β, and δ). Further definitions related to the GWO algorithm are provided below.

1. The distance between a gray wolf and its prey is given by:

$$\vec{D} = \left|\vec{C} \cdot \vec{X}_q(t) - \vec{X}(t)\right| \quad (17)$$

The location vector of the quarry is represented by $\vec{X}_q$, while $\vec{X}$ represents the location vector of the gray wolf. The current number of iterations is denoted by t, and $\vec{C}$ is defined as 2 times $\vec{r}_1$. Here, r_1 is a random number that ranges between 0 and 1.

At iteration t, $\vec{C}$ is set to $2 \cdot \vec{r}_1$ and $\vec{X}_q$ characterizes the prey location vector while $\vec{X}$ is the gray wolf location vector, where $r_1$is a random number between 0 - 1.

2. Location Update of the Gray Wolf:

$$\vec{X}(t+1) = \vec{X}_q(t) - \vec{A} \cdot \vec{D} \quad (18)$$

The equation $\vec{A} = 2\vec{a} \cdot \vec{r}_2 - \vec{a}, \vec{a}$ includes the convergence gene, denoted by"$\vec{a}$," which varies between 2 and 0, and the variable."$\vec{r}_2$," which is a random number between 0 and 1.

3. Prey position positioning:

$$\vec{D}_\alpha = \left|\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}\right|, \vec{D}_\beta = \left|\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}\right|, \vec{D}_\delta = \left|\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}\right| \quad (19)$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot \left(\vec{D}_\alpha\right), \vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot \left(\vec{D}_\beta\right), \vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot \left(\vec{D}_\delta\right) \quad (20)$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad (21)$$

The locations of α, β, and δ are represented by $\vec{X}_{\alpha}$, $\vec{X}_{\beta}$, and $\vec{X}_{\delta}$, respectively. $\vec{C}_1, \vec{C}_2$, and $\vec{C}_3$are random vectors, and $\vec{X}$ is the current solution location.

The principal phases of the Grey Wolf optimizing algorithm are as follows:

Phase 1 : Set the values for the grey wolf population's a, A, C, $\vec{X}_{\alpha}, \vec{X}_{\beta}, \vec{X}_{\delta}$, and $\vec{X}_{\delta}$ parameters to initialize the process.

Phase 2 : Compute the fitness values of every individual.

Phase 3 : Compare the fitness values of intelligent individuals with the fitness values of $\vec{X}_{\alpha}, \vec{X}_{\beta}$ and $\vec{X}_{\delta}$ to identify the current optimal solution and the suboptimal solution.

Phase 4 : Compute the values of a, A, and C.

Phase 5 : Update the current position of each intelligent individual according to equation (27).

Phase 6 : If the termination condition (maximum number of ıterations) is met, it will go back to Phase 2, otherwise, it will terminate.

CHAPTER THREE

DATA EXPLORATION AND PREPARATION

This chapter discusses the steps involved in data gathering, exploration, pre-processing, and finally, the correlation analysis between the features. Pre-processing is the initial and most crucial step in working with machine learning algorithms. We first discuss merging the datasets, cleaning them up, visualizing them, and then investigating the correlation. The chapter is divided into sections.

3.1. Data acquisition

The used data were gathered over 34 days with intervals of 15 minutes at two solar power plants in India (plant one is close to Gandikotta, Andhra, and plant two is close to Nasik, Maharashtra). To measure the generation rate, each plant has 22 inverter sensors connected to the inverter and the plant levels (an internal factor that can cause anomalies). The inverter measures the meteorological measurements at the plant level (they represent the external factors that can cause anomalies). The information is available, licensed, and accessible (Kannal, 2020). Table 4. displays the dataset's characteristics.

Table 4. The Study's Variables and Their Description

Variable type	Variable name	Variable Abbreviation (unit)	Variable Description
Internal factor	DC power	Power _DC (kW)	Amount of DC power generated by the inverter
	AC power	Power _AC (kW)	Amount of AC power generated by the inverter
	Total yield	Total _Power _DC (kw)	The summation of all the DC power that is generated from the inverter in time.
External factors	Solar irradiance	IRR ($kW/m2$)	The amount of electromagnetic radiation received from the sun per unit area
	Ambient temperature	Amb_Temp (C°)	The ambient temperature at the solar power plant
	Solar panel temperature	Module_Temp (C°)	The temperature that measured by attaching the sensor to the solar panel. This is the temperature indication for that module.

As we can see in Table 5, there are 22 distinct inverters with between 3104 and 3155 readings. Prediction models may have trouble due to this difference. Hence it should be considered. The largest discrepancy of 51 entries translates to a difference of roughly 13 hours because one item corresponds to a 15-minute measurement.

Table 5. Measurements Number of Each Inverter

Inverter ID	Number of Measurements
bvBOhCH3iADSZry	3155
1BY6WEcLGh8j5v7	3154
VHMLBKoKgIrUVDU	3133
7JYdWkrLSPkdwr4	3133
ZnxXDlPa8U1GXgE	3130
ih0vzX44oOqAx2f	3130
wCURE6d3bPkepu2	3126
z9Y9gH1T5YWrNuG	3126
uHbuxQJl8lW7ozc	3125
pkci93gMrogZuBj	3125
iCRJl6heRkivqQ3	3125
sjndEbLyjtCKgGv	3124
zVJPv84UY57bAof	3124
McdE0feGgRqW7Ca	3124
rGa61gmuvPhdLxV	3124
ZoEaEvLYb1n2sOq	3123
adLQvlD726eNBSB	3119
1IF53ai7Xc0U56Y	3119
zBIq5rxdHJRwDNY	3119
3PZuoBAID5Wc2HD	3118

WRmjgnKYAwPKWDb	3118
YxYtjZvoooNbGkE	3104

3.2. Data Preprocessing

Data were cleaned to eliminate anomalies, unnecessary information, and incomplete data. Data transformation was done to improve insights and produce positive results. CSV (Comma Separated Value) files are used to store the data. MinMaScaler is used to normalize the data before the features are examined. We combined the two datasets to create one dataset, which was then used to analyze the distribution and correlations after first cleaning and to prepare the data to eliminate null values and 110 unnecessary segments.

3.3. Solar power plant analysis

In order to determine whether there is any statistical difference in the power generation between the two plants, a study of two solar power plants in separate parts of India was conducted. This statistical study will determine which solar power plant is more effective at converting energy. Forecasting the weather and estimating power for the foreseeable future can be done using survey data from two solar power facilities.

3.3.1. Descriptive analysis of solar power plant – 1

Using Minitab software, statistical analysis is carried out. Minitab is free software that can be used for statistical analysis. The classical and Bayesian forms of Kovari's data analysis are straightforward.

According to Table 6, the average DC power produced by Solar Panel – 1 over all days and hours is 3147.43 W, but the average AC power produced by the Inverter is 307.803 W.

However, this is not the actual average power produced during the day Equation 22, as given below, is used to calculate the real average power of a day.

$$\text{The actual average power of a day} = \frac{\text{Total power}}{\text{no.of days}} \quad (22)$$

The following values are obtained by replacing the parameter values in Equation 22 and solving the following:

$$\text{Actual average DC power} = \frac{2.165\times10^{8}}{34} = 6.35 \times 10^{6}\ \text{W}$$

$$\text{Actual average AC power} = \frac{2.117\times10^{7}}{34} = 6.23 \times 10^{5}\ \text{W}.$$

This demonstrates that the average DC power is higher than the average AC power, indicating that the inverter's power conversion has the maximum loss and is the least effective.

Table 6. Statical analysis of solar power plant production (Plant – 1)

	DC – Power	AC – Power
Valid Value	68778	68778
Missing Value	0	0
Mean	3147.426	307.803
Median	429	41.494
St. Deviation	4036.457	394.396
Minimum Value	0	0
Maximum Value	14471.13	1410.95

Table 7. analyses the weather conditions for solar power plant – 1.

The average ambient temperature these days has been 20 C° and 35 C°, respectively 18 C° and 65 C° are the minimum and maximum module temperatures.

$$\text{Average irradiation of each day} = \frac{\text{Total irradiation}}{\text{no.of days}} = \frac{726.5}{34} = 21.36\text{kWh/m}^2$$

Table 7. Statical analysis of solar power plant production (Plant – 2)

	Ambient – Temp.	Module – Temp.	Irradiatıon
Valid Value	3182	3182	3182
Missing Value	0	0	0
Mean	25.532	31.091	0.228
Median	24.614	24.618	0.025
St. Deviation	3.355	12.261	0.301
Minimum Value	20.399	18.14	0
Maximum Value	35.252	65.546	1.222

The relationship between ambient temperature, module temperature, and irradiation is shown in Table 8. They are positively connected, as indicated by lower positive diagonal elements. The module temperature and irradiation are strongly connected, as indicated by the correlation coefficient of 0.962. A correlation value of 0.8 demonstrates the strong correlation between module temperature and ambient temperature, demonstrating the dependence of module temperature on ambient temperature. The p-value is less than 0.001, indicating that there is a chance that the probability of one out of 1,000 is incorrect.

Table 8. Correlation between irradiation, module temperature, and the ambient temperature

	Ambient – Temp.	Module – Temp.	Irradiatıo
Ambient_Temp	1		
Module_Temp	0.854	1	
Irradiatıon	0.723	0.962	1

3.3.2. Descriptive Analysis of Solar Power Plant – 2

Table 9. reveals that the average DC power production from Solar Panel – 2. overall days and hours is 246.702 W, while the average AC power output from the Inverter is 241.278 W. Despite what I said, this is not the day's average power equation can be used to determine the real daily average power (1).

$$\text{Actual average DC power} = \frac{1.67\times10^{7}}{34} = 4.9 \times 10^{5}\ \text{W}$$

$$\text{Actual average AC power} = \frac{1.63\times10^{7}}{34} = 4.79 \times 10^{5}\ \text{W}$$

This demonstrates that the average DC and AC power are equal, indicating that the inverter operates with 99 % efficiency and little loss during power conversion.

Table 9. Analysis of generation of solar power plant – 2

	DC power	AC power
Valid value	67698	67698
Missing value	0	0
Mean	246.702	241.278
Median	0	0
St. Deviation	370.57	362.112
Minimum value	0	0
Maximum value	1420.933	1385.42

Table 10. Analysis of weather conditions of solar power plant – 2

	Ambient – Temp.	Module – Temp.	Irradiation
Valid value	3259	3259	3259
Missing value	0	0	0
Mean	28.069	32.772	0.233
Median	26.981	27.535	0.019
Std. Deviation	4.062	11.344	0.313
Minimum value	20.942	20.265	0
Maximum value	39.182	66.636	1.099

The average ambient temperature throughout these days has been 21 C° and 39 C°, respectively. According to Table 10. the minimum and maximum module temperatures are 20 C° and 67 C°, respectively.

$$\text{Average irradiation of each day} = \frac{\text{Total irradiation}}{\text{no.of days}} = \frac{758.5}{34} = 22.3\text{kWh/m}^2$$

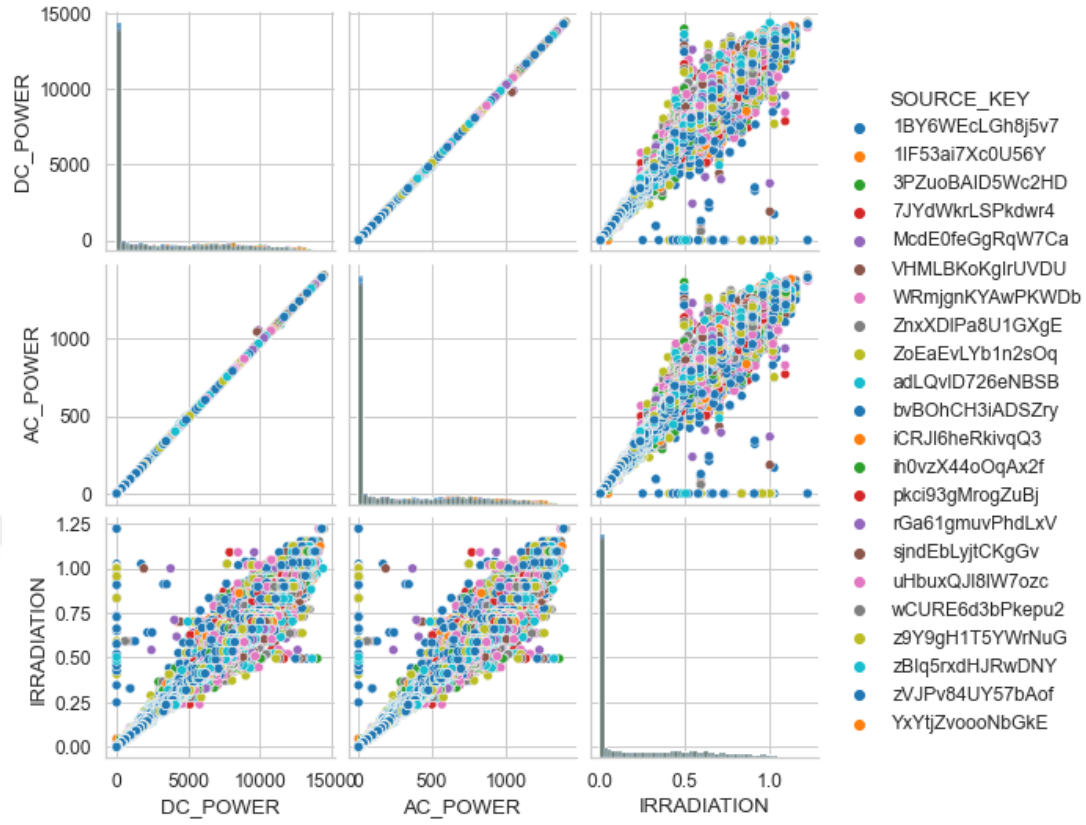
The solar power plant – 2's weather analysis has the same solar power plant-1's Pearson correlation. Irradiation and module temperature have strong relationships with each other and with ambient temperature.

3.3.3. Outliers Analysis

The following insight can be seen in Graphic 6:

- Outliers in Power-Irradiation indicate failure of the panel lines. If there is enough sunlight, but no power is generated, this points to faulty photovoltaic cells.
- Outliers in DC-AC conversion indicate failure at the inverter. If DC power is delivered, but less AC power is generated than expected, the inverter may malfunction.

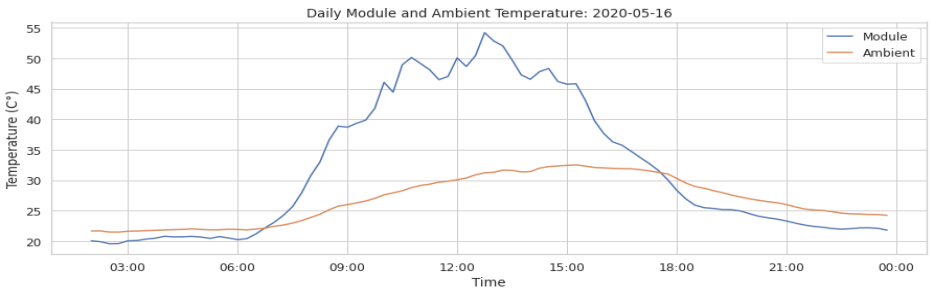
It can use these outliers for equipment fault and maintenance detection.



Graphic 6. Scatter Matrix of Features

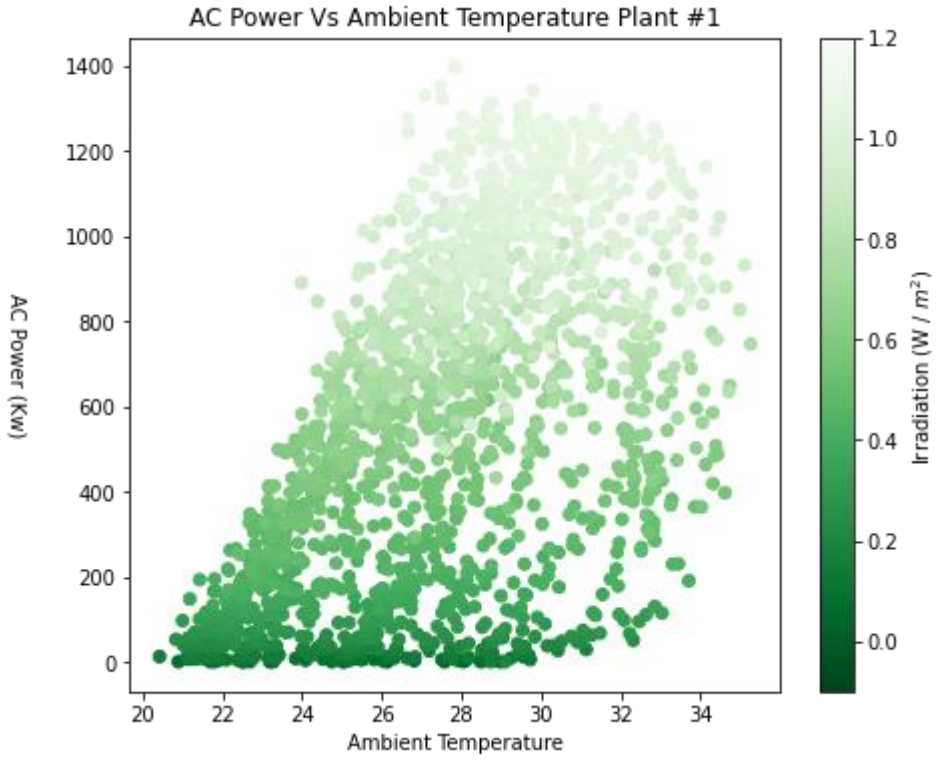
In addition, If we look closely at TOTAL_YIELD vs. SENSOR_NUM from the previous graphic, we see two groups of inverters. One group starts with a higher total yield than the other one. This is most likely because this group was installed earlier than the others.

3.3.4. Temperature analysis



Graphic 7. Daily module ambient temperature

As shown in Graphic 7. the ambient temperatures range from 20 to 35 C°, and modules reach temperatures from 18 to 65 C°. Modules reach significantly higher temperatures than their ambient air during the daytime. The ambient temperature is lagging behind the daily module cooldown. This means the modules cool down quicker than their environment.



Graphic 8. Relation between AC power, ambient temperature, and irradiation

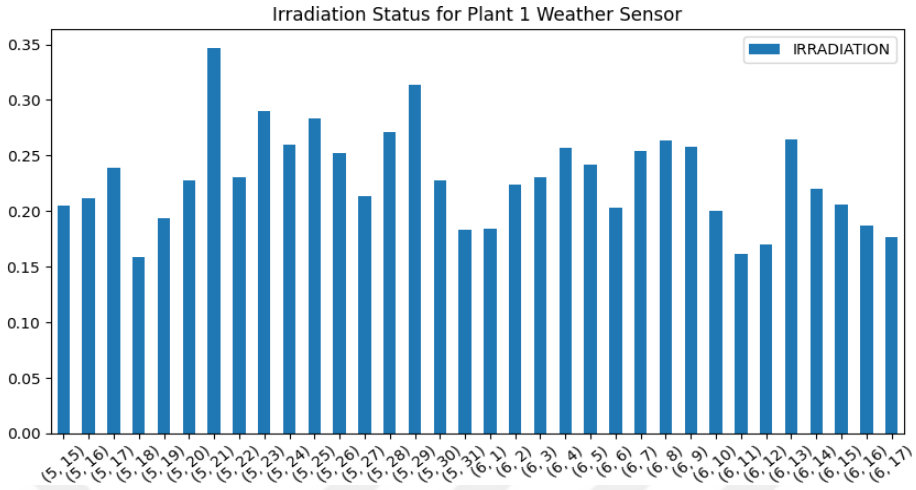
In Graphic 8. A clear positive relation between Irradiance and AC_power can be observed as predicted by the physics model. However, the relation is unclear because it also depends on irradiation.

3.3.5. Irradiation analysis

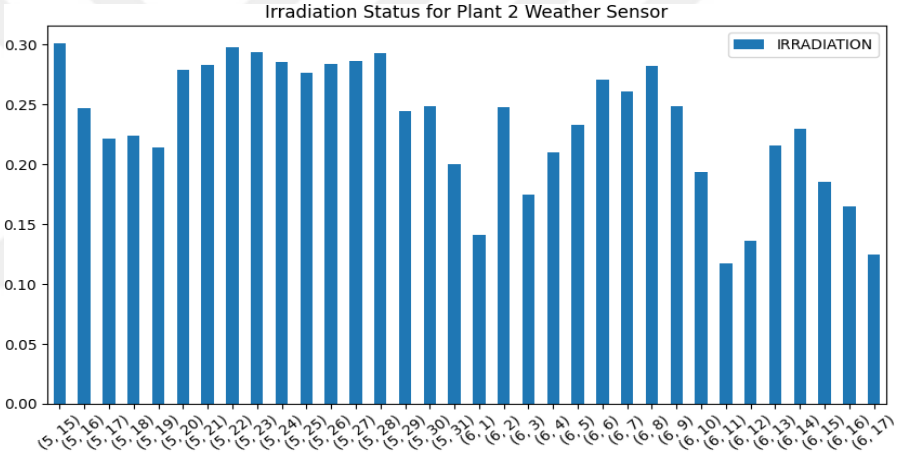
The irradiation status of Plants – 1 and 2 is depicted in Graphics 9 and 10, respectively. According to both graphical representations, Plant – 1 received a maximum of 0.34 W/m^2, and Plant – 2 received a minimum of 0.12 W/m2.

Even though there is not much difference in the daily radiation for the two plants, Plant 1 received more radiation for a month than the other two. With this information

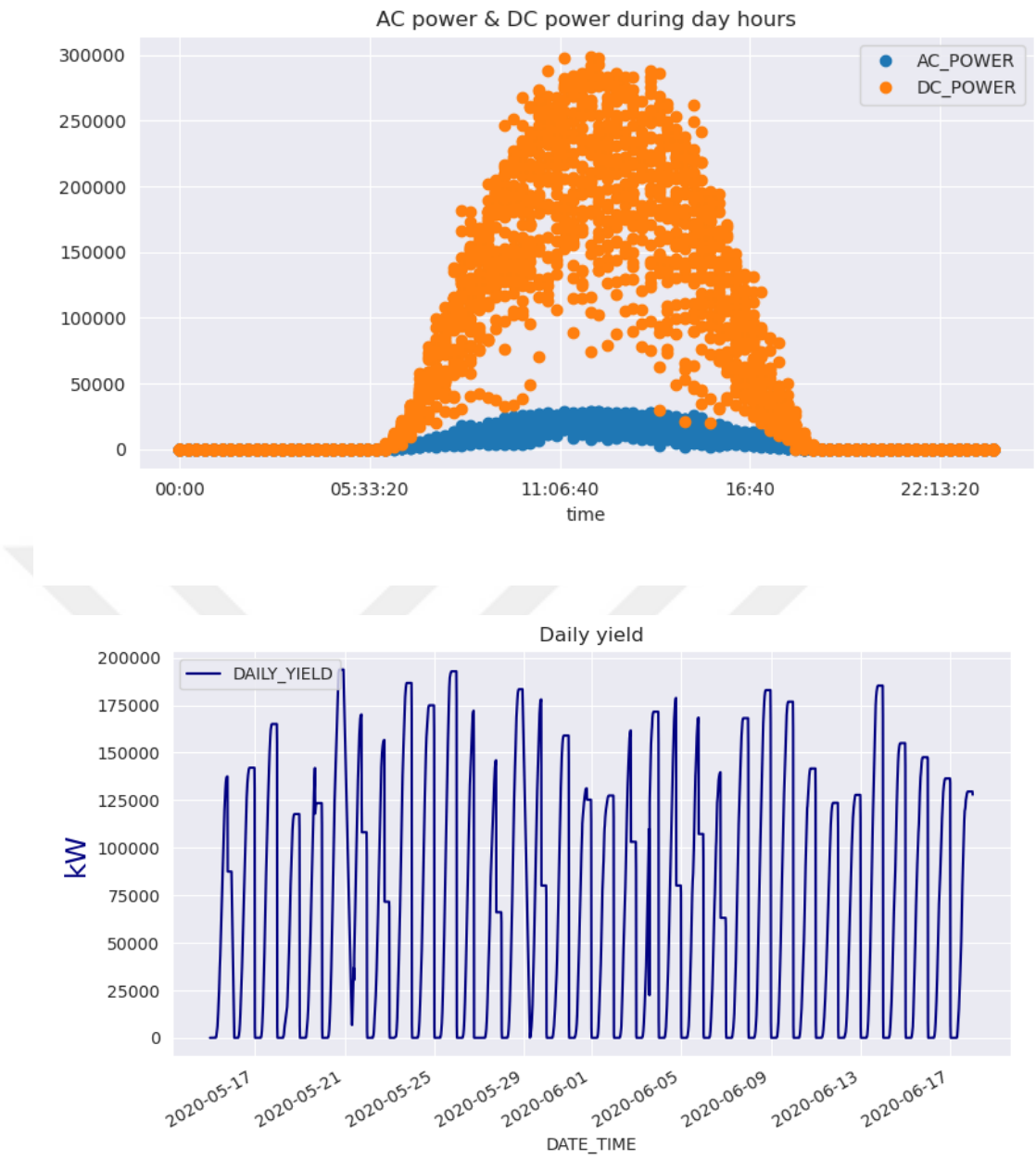
about the irradiation status of both plants, we recommend increasing Plant – 2's irradiation.



Graphic 9. Status of Plant – 1's Weather Sensor's Radiation



Graphic 10. Status of Plant – 2's Weather Sensor's Radiation



Graphic 11. Daily Yield and AC – DC Power during Day hours

The daily yield is shown in Graphic 11. According to the figure's forest, the maximum daily yield was 195000 kW, and the lowest was 120000 kW. Graphic 11.

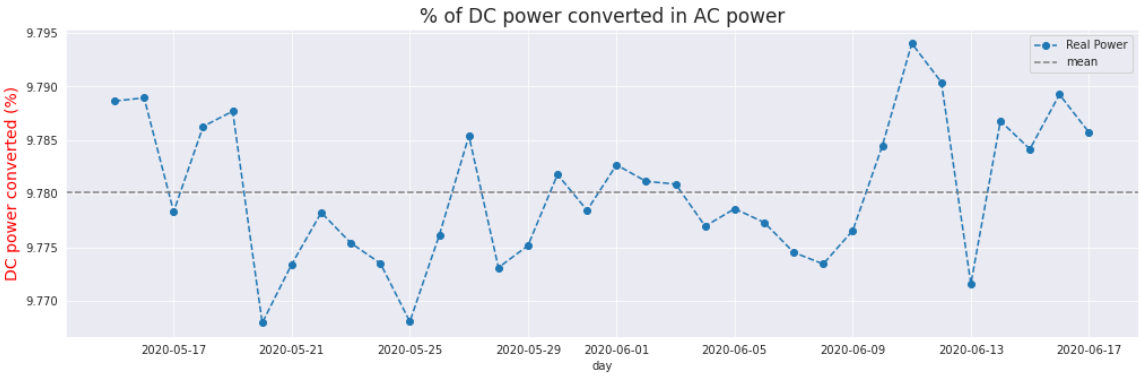
İn above graph details (AC and DC power during the day). If we closely examine the AC and DC power hours, we can conclude that the hours from 06:33:20 am to 04:40 pm are when we receive most of the DC power. Around 11:06:40 am, we received a maximum of 30000 kW of DC electricity. Since it is a solar power plant,

most power is generated during daylight. However, AC power is significantly less than DC power.

We could not survive without direct current (DC) power because many modern electrical and electronic devices rely on it for smooth operation and maybe even voltage. Neither type of power is "better" than the other; both are required. In actuality, alternating current (AC) dominates the electrical industry; all power outlets function flawlessly with AC-powered equipment, even if the current must be quickly changed to direct current (DC). This appears because DC cannot carry power from power plants to buildings over the same long distances as AC. Generating AC is also considerably simpler than DC due to how generators turn.

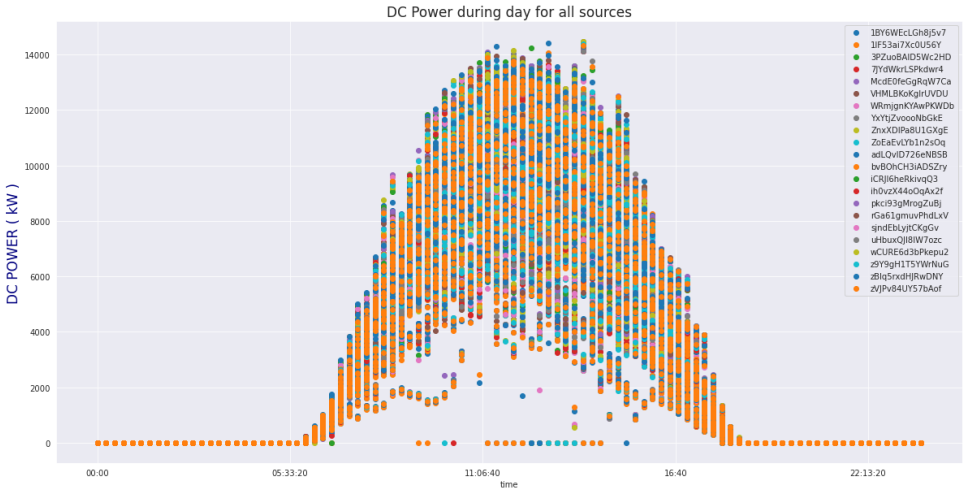
3.4. Identification of Faulty and Deficient Equipment

We need to locate real DC power to identify inadequate or flawed machinery. Graphic 12. displays the percentage of DC power transformed into AC power. Actual DC power and Mean DC power are shown in Graphic 13. The Actual power ranges from 9.765 to 9.794 %, and the mean power is 9.780 %.



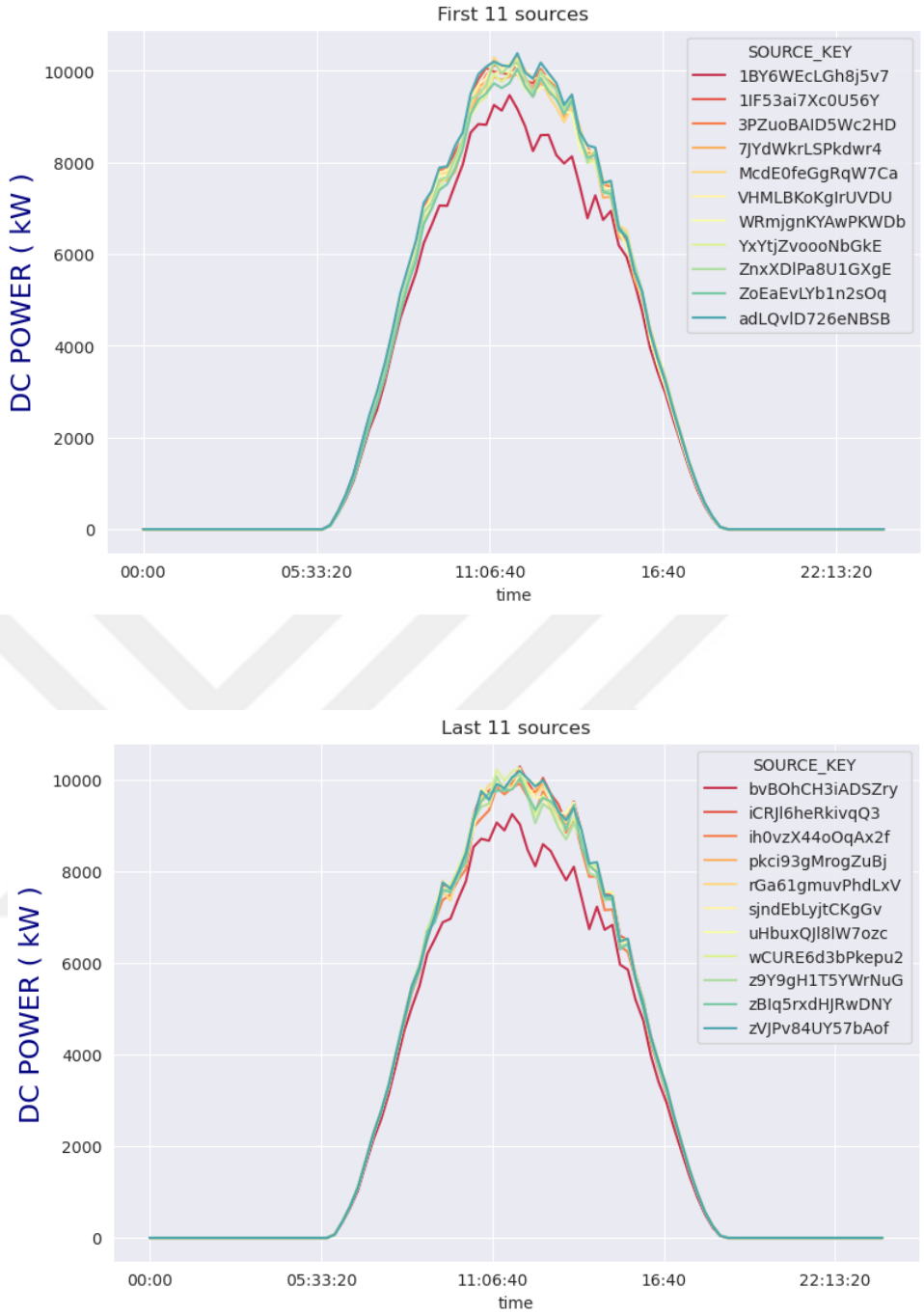
Graphic 12. Percentage of DC Power converted into AC Power

However, it still needs to provide us with precise information that would allow us to research good equipment further. Since we cannot see what is wrong with our power plant, let us look at how various inverters operate during the day. This brought us to the following graphical display, which provides details regarding DC power throughout the day from all sources.



Graphic 13. DC Power throughout a day from all the Sources

In both plants, there are numerous sources of various sorts. There are roughly 68,778 sources in one plant. Therefore, in the following section, I again focused on this query. I looked into the plant's initial and last sources to determine which was ineffective. If we successfully identify the inefficient source, we may determine the plant's overall efficiency and what other elements are causing the underperforming equipment.



Graphic 14. DC Power for First 10 and Last 10 Sources

Compared to other inverters, Graphic 14's first and last sources, 1BY6WEcLGh8j5v7 & bvBOhCH3iADSZry, show deficiencies; it is possible that these inverters need to be repaired or replaced. However, before we go into the specifics of inefficient inverters, let us look at some general plant problems. To do this, let us examine DC power generation over all 34 days.

CHAPTER FOUR

METHODS AND RESULTS

This chapter discusses the specific methods used for our anomaly detection model. Three approaches are used in this research for anomaly detection in solar power plants. The first approach uses the physical model, while the hybrid GWO_SVM regression model is used in the second approach. The third approach investigates the hybrid GWO_SVM classification model

4.1. Materials

Data from solar power plants in India (Near Gandikota, Andhra Pradesh) was gathered over 34 days with 15-minute intervals. Twenty – two inverter sensors were connected to both inverters and plants to monitor the generation rate (the internal factor that may lead to anomalies). The inverters monitored meteorological conditions at the plant level (external factor that can produce anomalies).

This data is publicly available, licensed, and accessible in accordance with (Kannal, 2020). The Variables used in this study are described in table 11 below.

Table 11. Description of Variables Used in this Study

Variable type	Variable name	Variable Abbreviation (unit)	Variable Description
Internal factor	DC power	Power _DC (KW)	The quantity of DC power produced by the inverter
	AC power	Power _AC (KW)	Amount of AC power produced by the inverter
	Total yield	Total _Power _DC (KW)	The total DC power output from the inverter over a period of time.

External factors	Solar irradiance	IRR (KW /$m2$)	The intensity of the electromagnetic radiation emitted by the sun per unit area
	Ambient temperature	Amb_Temp (C°)	The temperature around the solar power plant
	Solar panel temperature	Module_Temp (C°)	The temperature indication for the solar module is measured by attaching a sensor to the panel.

4.2. Methodology

This research investigates three different approaches for detecting anomalies in solar power facilities. The first methodology is based on a physical model, the second on a hybrid GWO SVM regression model, and the third on a hybrid GWO SVM classification model. The overall process is represented in Figure 18. and involves the following steps: data preparation, feature selection, prediction phase, and performance evaluation. These are explained in more detail in the subsequent sections.

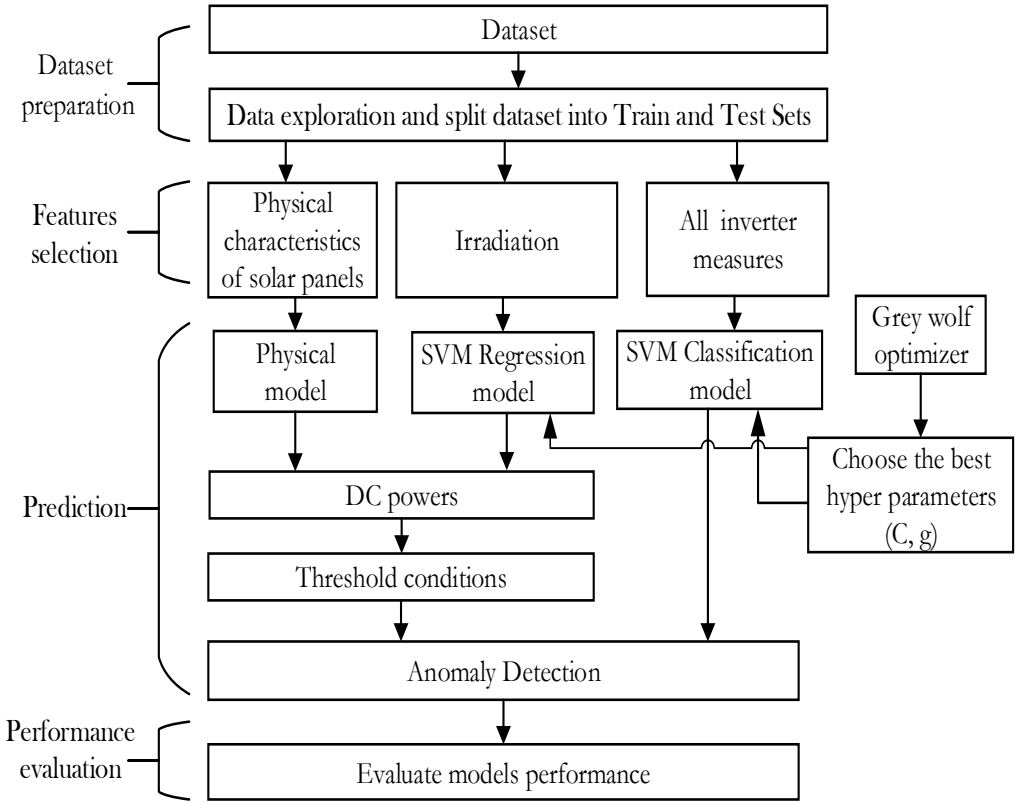


Figure 18. Overall methodology steps

4.2.1. Data preparation

The main dataset was preprocessed by combining the power generation and weather data into a single dataset, removing any measures with missing values, and looking at the scale of the variables. Then, 80 % of the dataset was randomly picked to train the prediction model, and the other 20 % was used to test the model's accuracy.

4.2.2. Features selection

The primary goal of feature selection is to improve the performance of a predictive model by utilizing only pertinent data and diminishing the computational cost of modeling by decreasing the input variable to be modeled. Spearman's rank correlation (ρ) was utilized to quantify the correlation rank between variables using the following equation:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2-1)} \quad (23)$$

Where d_i^2 is the difference between the two ranks of each observation, and n is the number of observations.

4.2.3. Prediction phase

4.2.3.1. Physical model

According to Hooda et al. (2018), the non-linear equation can be used to model the Power DC output of a photovoltaic cell.

$$P(t) = a * E(t)(1 - b * (T(t) + (E(t)/800) * (c - 20) - 25) - d * \ln(E(t))) \quad (24)$$

Where $P(t)$ is Power DC, $E(t)$ is irradiance, Temperature $T(t)$ and coefficients *a*, *b*, *c*, *d* (Rahman et al., 2021).

4.2.3.2. GWO-SVM classification / Regression Model

GWO and SVM were combined to create GWO-SVM C and GWO-SVM R, which were used to predict Power_DC. While support vector regression models usually perform well in modeling linear and nonlinear relationships, SVR accuracy depends on the proper selection of parameters C (penalty term) and g (kernel width). These two factors are known to have a vast range of variations and significantly influence the accuracy of SVR. As there is no set procedure for choosing these values, finding the right parameters can be computationally demanding and can be seen as an optimization problem. To tackle this problem, we have used the hybrid optimization technique GWO. The GWO method optimizes the SVM regression prediction procedure by doing the following:

Step 1 : Determine the dependent and independent variables based on the model's presumptions, then provide data for the SVM training and test sets.

Step 2 : While determining the input parameters for the GWO algorithm (the values of a, A, and C), initializing the scopes of parameters c and g;

Step 3 : Each intelligent individual location carrier must have the letters c and g to initialize the gray wolf population;

Step 4 : Determine each gray wolf's fitness value by learning the training set's data using the SVM's initial c and g values.

Step 5 : The fitness value divides the gray wolf group into four unique levels: a, b, d, and x.

Step 6 : Update each gray wolf's location in accordance with the algorithm (12).

Step 7 : The GWO algorithm then modifies each intelligent person's position in accordance with the fitness value, keeping the location with the highest fitness value;

Step 8 : Model training is complete, and the ideal value of C and g is generated when the iteration times reach Max iteration (the maximum number of iterations);

Step 9 : The SVM regression forecasting model is built using the best c and g, and its performance is assessed and predicted using the test data set that was previously partitioned. The algorithm design step's flowchart is shown in Figure 19. and is based on choosing the best SVM parameters, c, and g, using the grey wolf approach, as described in the section above.

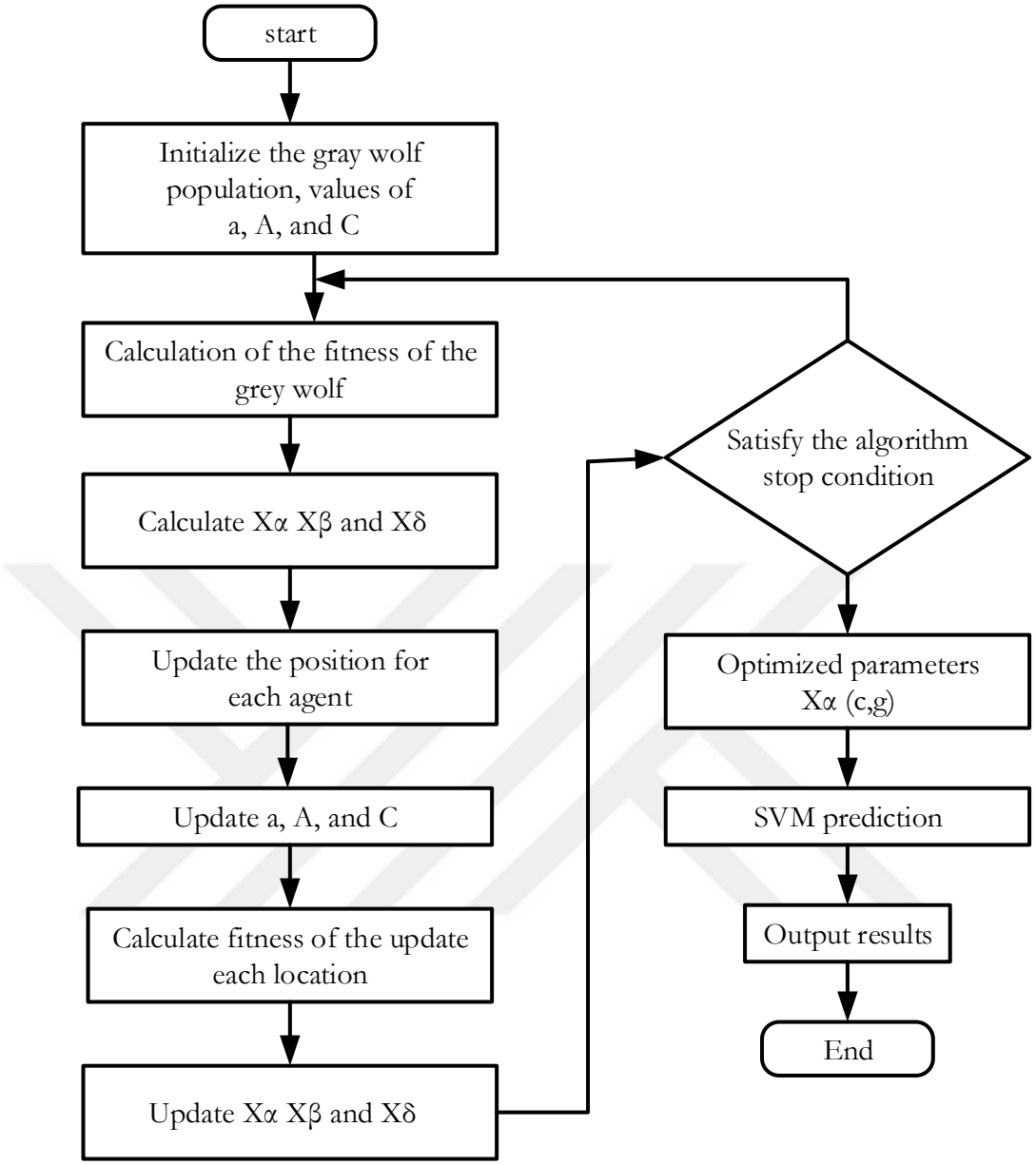


Figure 19. The proposed architecture of GWO optimizer with SVR

4.2.4. Anomaly prediction decision for models

If the value of *P(t)* derived from the GWO-SVM R and Physical models is over the threshold, it indicates that the inverter is functioning normally; however, if the value is below the threshold, it means that the inverter has failed. The GWO-SVM C model further distinguishes the two categories, classifying solar panels into three categories (Normal, Defective, and Marginal) based on the P(t) value.

$$P(t) = \begin{cases} Normal, \ P(t) \geq \theta \\ Fault \ , \ P(t) < \theta \end{cases} \quad (25)$$

In our experiment, we tested a range of threshold levels. The optimal limit was 215 kW.

4.2.5. performance evaluation

The effectiveness of the Physical and GWO-SVM R models developed in this article is evaluated using Root Mean Square Error (MSE) metrics.

$$RMSE \quad = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - x_i)^2} \quad (26)$$

In the formula, y_i is the actual situation value, x_i is the forecasted value, and n is the predicted sample size.

While the formulas (Eq. 27, 28, and 29) are used to calculate the classification accuracy of the GWO-SVM_C model.

$$Sensitivity = \frac{TP}{TP+FN} \times 100\% \quad (27)$$

$$Specifity = \frac{TN}{FP+TN} \times 100\% \quad (28)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (29)$$

The digits TP (True Positive) and TN (True Negative) represent the number of correctly identified Normal/Fault states of the inverter, while FN (False Negative) and FP (False Positive) signify the number of incorrectly identified Normal/Fault states of the inverter.

4.2.6. Experimental and Results

This project aims to create a prediction model that can classify the internal and external elements that lead to PV power plant faults. To do this, we conducted three experiments.

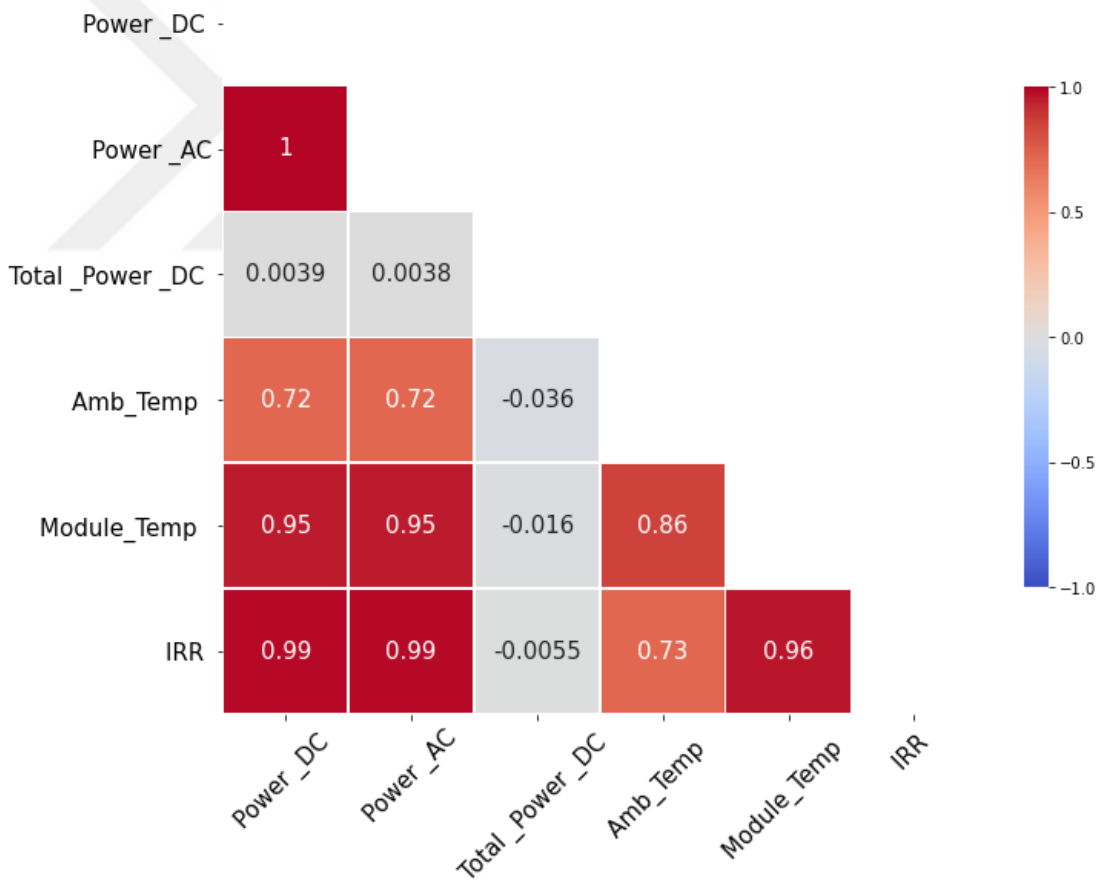
First, we studied the data characteristics' correlation to find the most important factors for training our predictive models.

Second, we tested various optimization models to establish the best hyperparameters for our prediction models.

Third, we employed three methodologies for detecting anomalies in solar power plants.

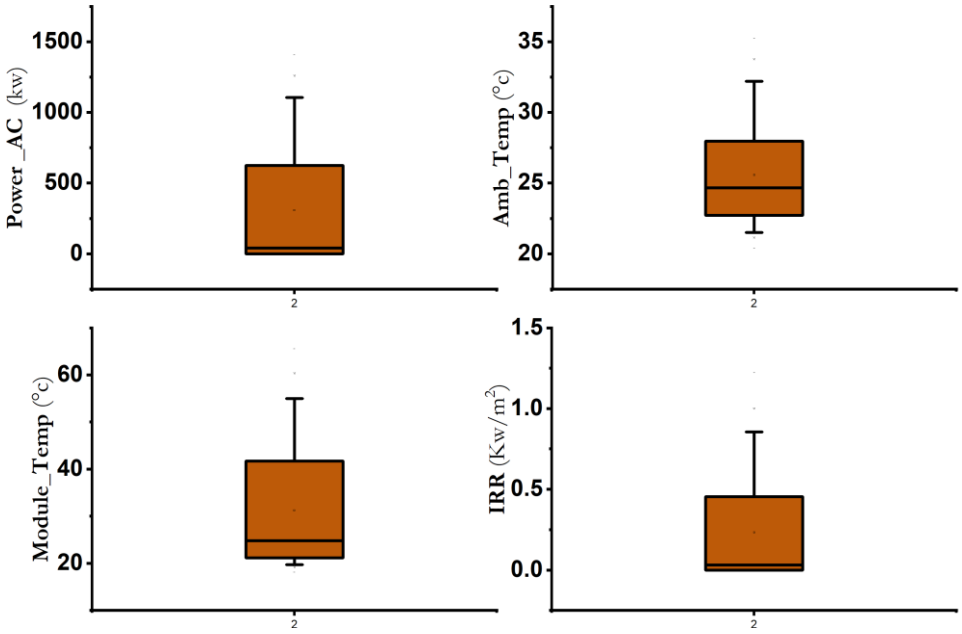
Finally, we used the most accurate prediction model to determine which days and inverters had the highest failure rate.

Graphic 15 shows the correlation coefficients between the input factors and the solar panel output Power_Dc. The correlation coefficient is a normalized covariance measure, with values ranging from (-1 to 1). A strong negative correlation of -1 implies that an increase in one variable will lead to a decrease in the other. In contrast, a strong positive correlation of 1 implies that an increase in one variable will increase the other. The diagonal values, which represent a variable's correlation with itself (autocorrelation), are equal to 1 since a variable is perfectly correlated.



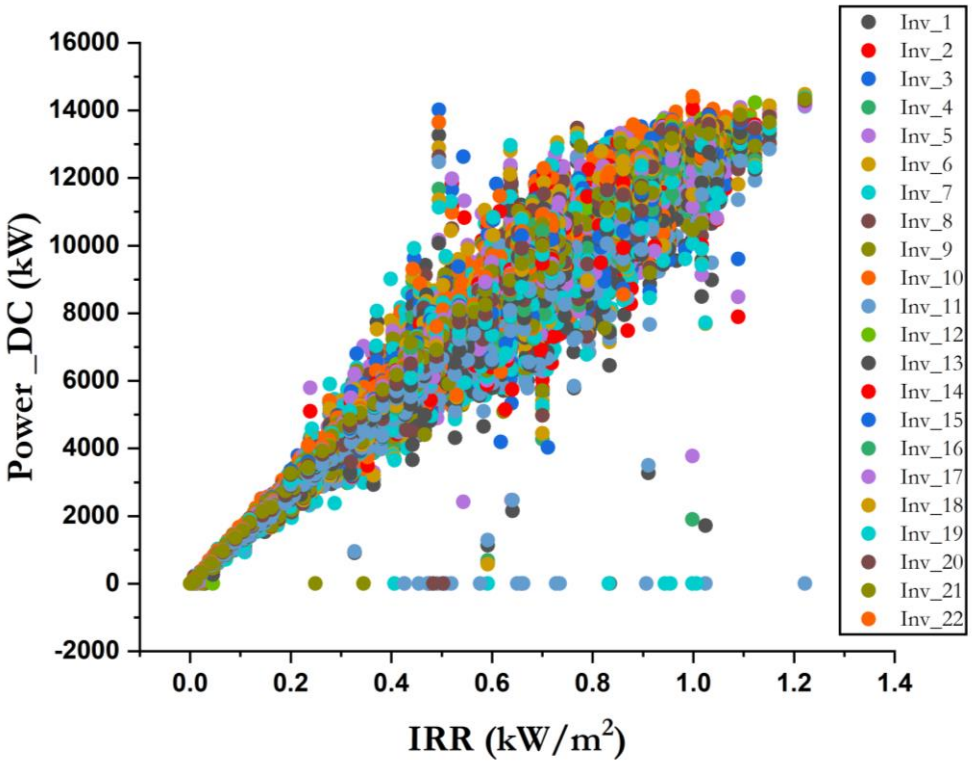
Graphic 15. Correlation matrix for dataset variables

A clear correlation can be seen between the output of Power DC and the inputs of Power AC, Module Temperature, and irradiation. In contrast, there is less of a connection between Daily Yield, Total Yield, Ambient Temperature, and Power DC. The inputs of our experiment are Power AC, Module Temperature, and irradiation, while Daily Yield, Total Yield, and Ambient Temperature are treated as outputs.



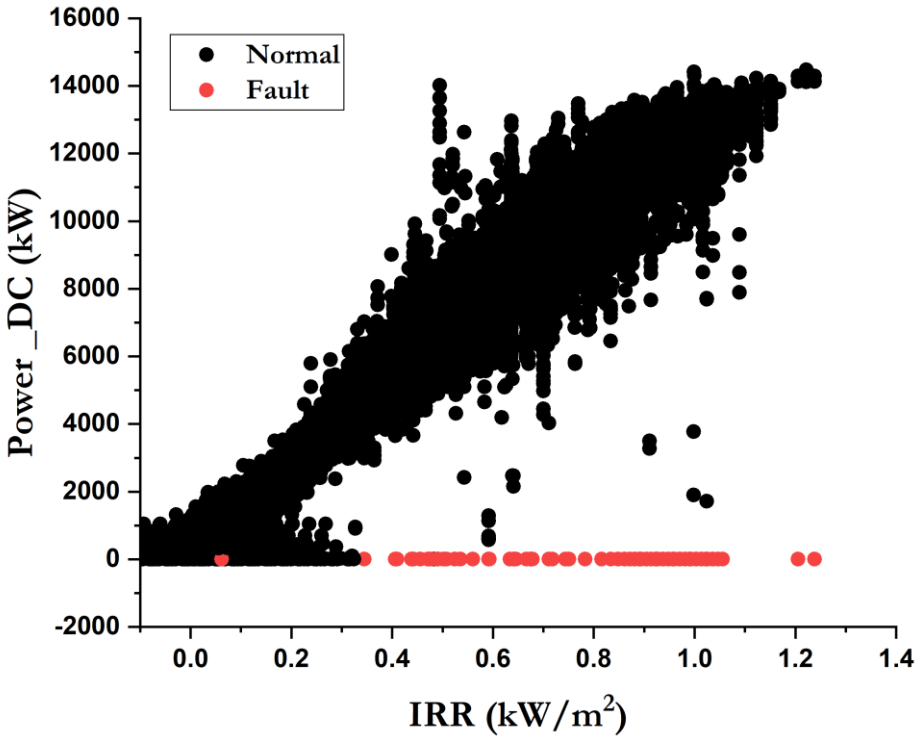
Graphic 16. Box plots for selected inputs variables

As seen in Graphic 16, the variables of our dataset have varied scales due to an error in the prediction phase. To ensure the accuracy of our models, the state of the inverter (Normal / Fault) must be indicated in the dataset. Graphic 17s scatter plot of DC power and irradiance reveals outliers, demonstrating that some inverters did not receive DC power despite enough sunlight to generate electricity. This highlights the efficiency of our solar panel lines in converting sunlight to DC power.



Graphic 17. Distribution of power_DC and IRR for each inverter

A scatter plot between DC power and irradiance (see Graphic 17), with the addition of inverter states, has been created to identify any anomalies in the solar energy-to-electricity conversion which would indicate faulty photovoltaic panel lines. Equipment failure can be assumed if no power is detected at the inverter during normal daylight operation (see Graphic 18).



Graphic 18. Distribution of power_dc and IRR for each inverter after determine inverter status

Before deploying SVM classification and regression models to forecast the amount of DC power generated by the inverter for solar power plants, the GWO method is utilized to optimize the SVM model hyperparameters in order to improve the prediction accuracy. To demonstrate the advantages of the GWO algorithm, two benchmark optimizer models have been created for comparison {Grid search (GS) and Random search (RS)}. The philosophy behind these optimizers and the rationale for their selection are discussed in reference (Strobl & Meckler, 2010)

In the GWO-SVM classification model, all dataset attributes are utilized to predict Power DC. On the other hand, the physical model only employs Irradiation and Module Temp. Table 12. displays the initial parameters for the GWO algorithm, and the SVR model was created utilizing the Jupiter Programming Language, which supports Python. Irradiation was applied in the GWO-SVM regression model to Power DC..

Table 12. Initial parameters of the GWO

Optimizer name	Parameter	Value
GWO	A	Min = 0 and max = 2
	Number of agents	100
	Iterations number	50
GS and RS	C = Linear	Min = 0.001and max = 10,000
	G = Linear, RBF, sigmoid	Min = 0.001and max = 10,000

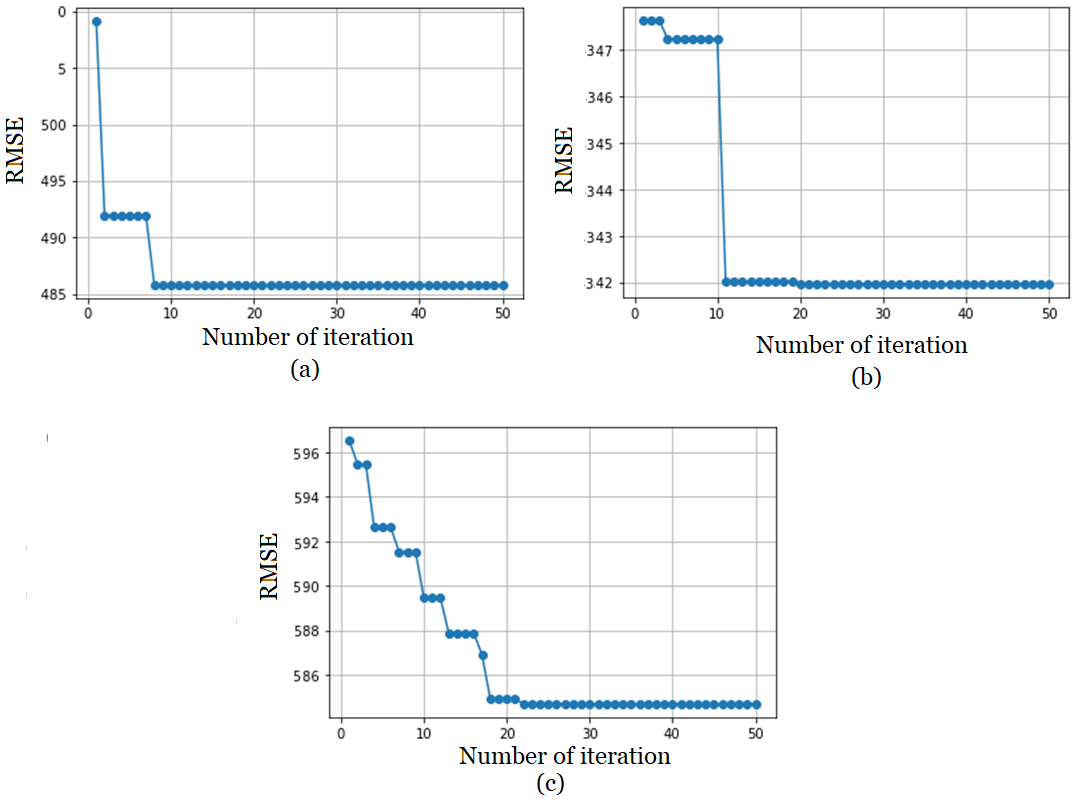
Graphic 19. depicts the prediction accuracy results of the proposed models SVM R and GWO-R with the selected benchmark models. All models were trained using the dataset from June 6 to June 21, 2020, and the RMSE values for all models are displayed in Table 13.

Table 13. The impact of different optimization methods on power_DC prediction accuracy

Model	RMSE
RS _SVM	532.47
SVM_R	415.98
GS_SVM	400.83
GWO-SVM_R	318.04

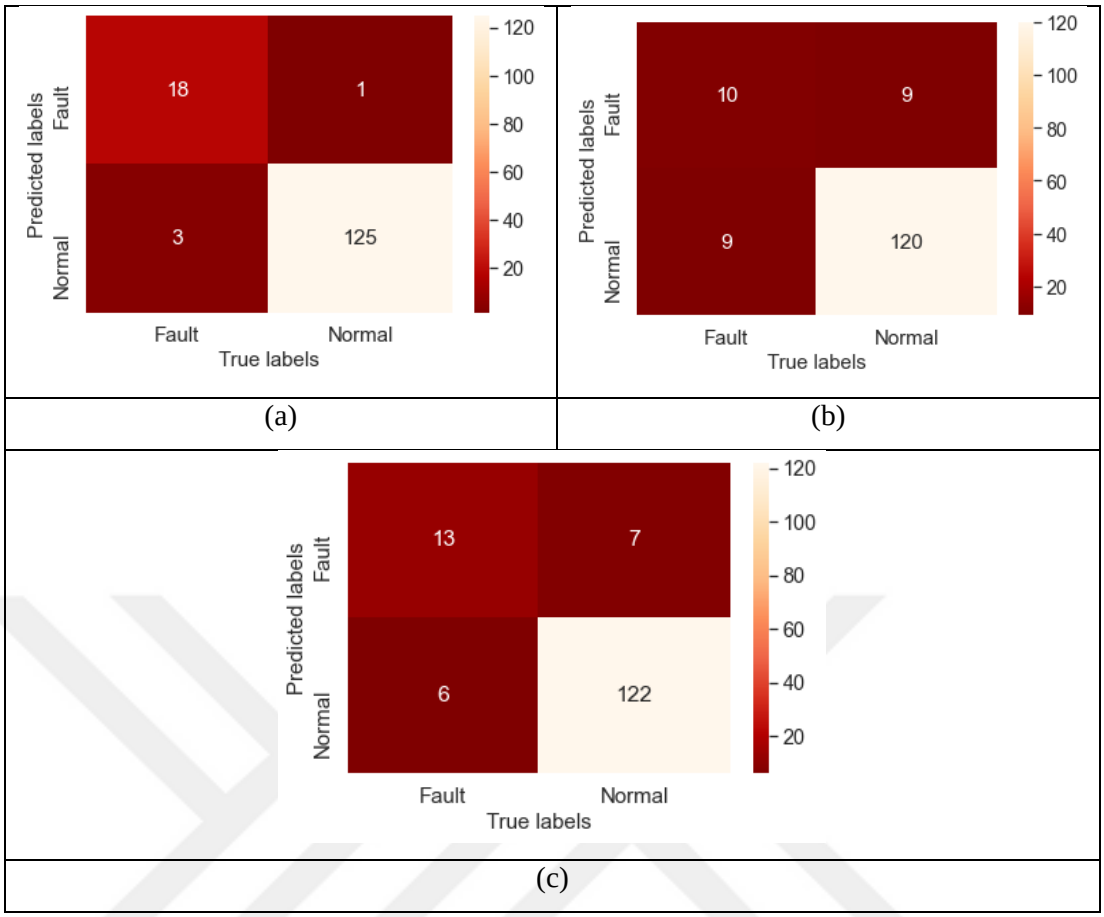
In general, the performance of the SVM R prediction model increased when GS and GWO were included. The RMSE for SVM R is reduced by 3 % when using GS and 23 % when using SVM-GW R. When SVM R is combined with SVM RS, the RMSE of Power Dc prediction rises from (415.98) to (532.47), indicating a reduction in performance. These results demonstrate that the GWO optimizer enhances the SVM's regression performance.

Graphic 19. depicts a comparison of the convergence curves of the GWO, GS, and RS optimizers. Nota bene, la capacidad fsica is la media de capacidad fsica obtenida The superior the performance, the lower the values of best fitness, worst fitness, and mean fitness. Using GWO models yields the highest performance, as determined by observation.



Graphic 19. Convergence curves for (a) GS optimizer,(b) GW optimizer and (c) RS

Three models are tested for anomaly identification in solar power plants to categorize the inverters into two classes based on their performance (Normal and Fault). The first model employs the physical model, whereas the second model employs the hybrid GWO SVM R model. The hybrid GWO SVM C model is the third model. The result of this comparison is depicted in Graphic 20. as a confusion matrix.



Graphic 20. Confusion matrix for (a) SVM-GW_C model, (b) Physical model and (c) SVM-GW_R

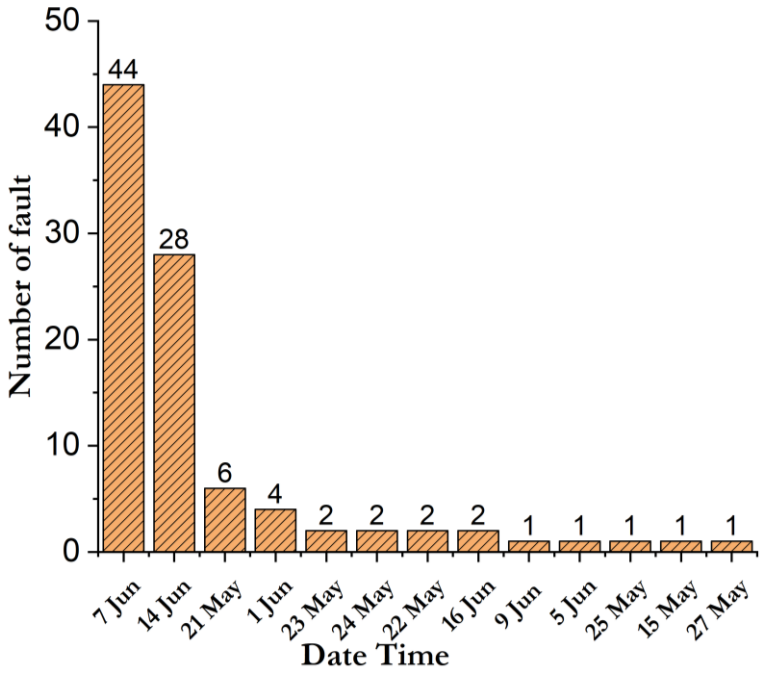
The confusion matrix reveals that the True classification of the GWO SVM C model for all classes was 143. On the other hand, the model was offered incorrectly four times, once as Fault class and twice as Normal class. Therefore, GWO SVM C attained an overall accuracy of 97.28 %, followed by the SVM-GW R model (91.22 %) and the Physical model (87.84 %) with the lowest accuracy.

In addition, sensitivity and specificity rates were determined and displayed in Table 14. based on a confusion matrix. Compared to the Physical and SVM-GW R models, the SVM-GW C has the greatest values for sensitivity and specificity.

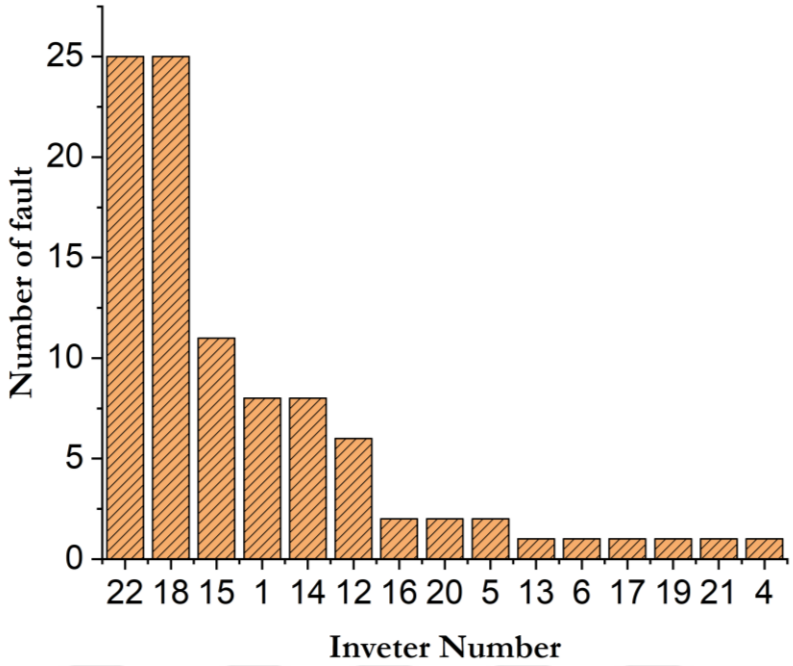
Table 14. Sensitivity and specificity rates for predictive models

Model	Sensitivity	Specificity
SVM-GW_C	85.71 %	99.21 %
SVM-GW_R	68.42 %	94.57 %
Physical model	52.63 %	93.02 %

Based on the previous findings, we can infer that the SVM-GW C model is the best model for anomaly identification in solar power plants by classifying inverter states into two categories (Normal and Fault). In addition, it is possible to discover the day and inverter with the highest failures using SVM-GW C, as illustrated in Graphics 21 and 22. On June 7th, the greatest number of errors were reported. And Inverters 18 and 22 saw the most failures.



Graphic 21. Ranking the number of the most failures days that occur in inverters based on the SVM-GW_C model anomaly detection



Graphic 22. Ranking the number of failures that had in inverters based on the SVM-GW-C model, anomaly detection

CHAPTER FIVE

CONCLUSIONS AND FUTURE WORK

5.1. Conclusion

It is essential to use data-driven methods to detect anomalies in present-day solar power plants in order to minimize downtime and optimize efficiency. This study examines the performance of three machine learning models to find the best model that can accurately detect photovoltaic (PV) system abnormalities. The correlation coefficients between the plants' internal and external feature characteristics were calculated and used to assess the models' performance in recognizing anomalies. GWO-SVM C proved adept at recognizing anomalies and accurately distinguishing the healthy signal. Further research will explore intelligent anomaly mitigation strategies. Applying the growing trend of distributed machine learning, i.e., federated learning, in large-scale intelligent solar power grids could be an interesting research topic.

5.2. Future work

The following suggestion can be considered to develop the work presented in this study.

To further this investigation, future studies may examine new active fault protection strategies, such as how to safely, quickly, and actively fix the problem. The active fault protection solution should improve system efficiency, reliability, safety, and fault immunity based on the trip signal produced by the suggested ways. Consequently, a suitable future study topic would be the combination of active fault protection approaches with the suggested methods. PV inverters assume increasing duties as PV penetration rises, including fault detection and protection, power conversions, and recording maximum power points. PV inverters can offer more safety features because they are the most sophisticated part of the PV system. Recent PV inverters, for instance, are equipped with various fault detection options, including insulation detection in ungrounded PV arrays, ground fault detection, dc-arc fault detection, and residual current detection.

Nevertheless, current PV inverter fault detection methods rely solely on instantaneous measurements and signal processing. For this reason, fault classification for PV inverters is still unavailable. Future studies may concentrate on incorporating the proposed fault classification methods into the PV inverters, making excellent use of the easily accessible historical data. Better fault detection features would be made available, which might raise the PV system's reliability and safety.

REFERENCES

Alam, M. K., Khan, F., Johnson, J., & Flicker, J. (2013). PV ground-fault detection using spread spectrum time domain reflectometry (SSTDR). *2013 IEEE Energy Conversion Congress and Exposition*, 1015–1102.

Alam, M. K., Khan, F., Johnson, J., & Flicker, J. (2015). A comprehensive review of catastrophic faults in PV arrays: types, detection, and mitigation techniques. *IEEE Journal of Photovoltaics*, *5*(3), 982–997.

Ancuta, F., & Cepisca, C. (2011). Failure analysis capabilities for PV systems. *Recent Res. Energy, Environ. Entrep. Innov*, 109–115.

Aslam, S., Herodotou, H., Mohsin, S. M., Javaid, N., Ashraf, N., & Aslam, S. (2021). A survey on deep learning methods for power load and renewable energy forecasting in smart microgrids. *Renewable and Sustainable Energy Reviews*, *144*, 110992.

Association, N. F. P., of Fire Underwriters, N. B., & Committee, N. F. P. A. N. E. C. (1915). *National electrical code* (Vol. 70). National Fire Protection Association.

Balzategui, J., Eciolaza, L., & Maestro-Watson, D. (2021). Anomaly detection and automatic labeling for solar cell quality inspection based on generative adversarial network. *Sensors*, *21*(13), 4361.

Benninger, M., Hofmann, M., & Liebschner, M. (2019). Online Monitoring System for Photovoltaic Systems Using Anomaly Detection with Machine Learning. *NEIS 2019; Conference on Sustainable Energy Supply and Energy Storage Systems*, 1–6.

Benninger, M., Hofmann, M., & Liebschner, M. (2020). Anomaly detection by comparing photovoltaic systems with machine learning methods. *NEIS 2020; Conference on Sustainable Energy Supply and Energy Storage Systems*, 1–6.

Blackburn, J. L., & Domin, T. J. (2015). *Protective relaying: principles and applications*. CRC press.

Blaesser, G., & Munro, D. (1993). Guideline for the Assessment of Photovoltaic Plants Document B. Analysis and Presentation of Monitoring Data. Report EURO 16339. *Joint Research Center, European Commission*.

Branco, P., Gonçalves, F., & Costa, A. C. (2020). Tailored algorithms for anomaly detection in photovoltaic systems. *Energies*, *13*(1), 225.

Brooks, B. (2011). The Bakersfield fire: a lesson in ground-fault protection. *SolarPro Mag*, *62*, 62–70.

Cespedes, A. J. J., Pangestu, B. H. B., Hanazawa, A., & Cho, M. (2022). Performance Evaluation of Machine Learning Methods for Anomaly Detection in CubeSat Solar Panels. *Applied Sciences*, *12*(17), 8634.

Chouder, A., & Silvestre, S. (2010). Automatic supervision and fault detection of PV systems based on power losses analysis. *Energy Conversion and Management*, *51*(10), 1929–1937.

Commission, I. E., & others. (1998). Photovoltaic system performance monitoring guidelines for measurement, data exchange, and analysis. *IEC 61724*.

De Benedetti, M., Leonardi, F., Messina, F., Santoro, C., & Vasilakos, A. (2018). Anomaly detection and predictive maintenance for photovoltaic systems. *Neurocomputing*, *310*, 59–68.

Deif, M. A., Solyman, A. A. A., & Hammam, R. E. (2021). ARIMA Model Estimation Based on Genetic Algorithm for COVID-19 Mortality Rates. *International Journal of Information Technology & Decision Making*, *20*(6), 1775–1798.

Drews, A., De Keizer, A. C., Beyer, H. G., Lorenz, E., Betcke, J., Van Sark, W., Heydenreich, W., Wiemken, E., Stettler, S., Toggweiler, P., & others. (2007). Monitoring and remote failure detection of grid-connected PV systems based on satellite observations. *Solar Energy*, *81*(4), 548–564.

Elsheikh, A. H., Katekar, V. P., Muskens, O. L., Deshmukh, S. S., Abd Elaziz, M., & Dabour, S. M. (2021). Utilization of LSTM neural network for water production forecasting of a stepped solar still with a corrugated absorber plate. *Process Safety and Environmental Protection*, *148*, 273–282.

Elsheikh, A. H., Panchal, H., Ahmadein, M., Mosleh, A. O., Sadasivuni, K. K., & Alsaleh, N. A. (2021). Productivity forecasting of solar distiller integrated with evacuated tubes and external condenser using artificial intelligence model and moth-flame optimizer. *Case Studies in Thermal Engineering*, *28*, 101671.

Feng, M., Bashir, N., Shenoy, P., Irwin, D., & Kosanovic, D. (2020). Sundown: Model-driven per-panel solar anomaly detection for residential arrays. *Proceedings of the 3rd ACM SIGCAS Conference on Computing and Sustainable Societies*, 291–295.

Firth, S. K., Lomas, K. J., & Rees, S. J. (2010). A simple model of PV system performance and its use in fault detection. *Solar Energy*, *84*(4), 624–635.

Flicker, J., & Johnson, J. (2013). *Solar America Board for Codes and Standards*.

Fuses—Part, L.-V. (2010). *6: Supplementary Requirements for Fuse-Links for the Protection of Solar Photovoltaic Energy Systems*. IEC.

Gertler, J. (1998). *Fault detection and diagnosis in engineering systems*. CRC press.

Guide, S. (2014). *Solar Power Product Solutions--Comprehensive photovoltaic protection*. Mersen.

Haeberlin, H., & Beutler, C. (1995). Normalized representation of energy and power for analysis of performance and on-line error detection in PV-systems. *Proc. 13th EU PV Conf., Nice*, 934.

Hammam, R. E., Attar, H., Amer, A., Issa, H., Vourganas, I., Solyman, A., Venu, P., Khosravi, M. R., & Deif, M. A. (2022). Prediction of Wear Rates of UHMWPE Bearing in Hip Joint Prosthesis with Support Vector Model and Grey Wolf Optimization. *Wireless Communications and Mobile Computing*, *2022*.

Harrou, F., Dairi, A., Taghezouit, B., & Sun, Y. (2019). An unsupervised monitoring procedure for detecting anomalies in photovoltaic systems using a one-class support vector machine. *Solar Energy*, *179*, 48–58.

Hempelmann, S., Feng, L., Basoglu, C., Behrens, G., Diehl, M., Friedrich, W., Brandt, S., & Pfeil, T. (2020). Evaluation of unsupervised anomaly detection approaches on photovoltaic monitoring data. *2020 47th IEEE Photovoltaic Specialists Conference (PVSC)*, 2671–2674.

Hernández, J. C., & Vidal, P. G. (2009). Guidelines for protection against electric shock in PV generators. *IEEE Transactions on Energy Conversion*, *24*(1), 274–282.

Hernández, J. C., Vidal, P. G., & Medina, A. (2010). Characterization of the insulation and leakage currents of PV generators: Relevance for human safety. *Renewable Energy*, *35*(3), 593–601.

Hernday, P. (2011). Field applications for IV curve tracers. *Solarpro, August*.

Ibrahim, M., Alsheikh, A., Al-Hindawi, Q., Al-Dahidi, S., & ElMoaqet, H. (2020). Short-time wind speed forecast using artificial learning-based algorithms. *Computational Intelligence and Neuroscience*, *2020*.

Ibrahim, M., Alsheikh, A., Awaysheh, F. M., & Alshehri, M. D. (2022). Machine learning schemes for anomaly detection in solar power plants. *Energies*, *15*(3), 1082.

Inverters, C., & Converters, C. (2010). Interconnection System Equipment for Use with Distributed Energy Resources. *Standard UL-1741*.

Iyengar, S., Lee, S., Sheldon, D., & Shenoy, P. (2018). Solarclique: Detecting anomalies in residential solar arrays. *Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies*, 1–10.

Kannal, A. (2020). Solar Power Generation Data. *Kaggle. Com. Available Online: Https://Www. Kaggle. Com/Anikannal/Solar-Powergeneration-Data (Accessed on 30 January 2022)*.

Karatepe, E., Hiyama, T., & others. (2011). Controlling of artificial neural network for fault diagnosis of photovoltaic array. *2011 16th International Conference on Intelligent System Applications to Power Systems*, 1–6.

Kosek, A. M., & Gehrke, O. (2016). Ensemble regression model-based anomaly detection for cyber-physical intrusion detection in smart grids. *2016 IEEE Electrical Power and Energy Conference (EPEC)*, 1–7.

Kunz, G., & Wagner, A. (2004). Internal series resistance determinated of only one IV-curve under illumination. *19th Photovoltaic Solar Energy Conference Paris*.

Lin, L.-S., Chen, Z.-Y., Wang, Y., & Jiang, L.-W. (2022). Improving Anomaly Detection in IoT-Based Solar Energy System Using SMOTE-PSO and SVM Model. In *Machine Learning and Artificial Intelligence* (pp. 123–131). IOS Press.

Luebke, C. J., Hastings, J. K., Pahl, B., Zuercher, J. C., & Yanniello, R. (2016). *String and system employing direct current electrical generating modules and a number of string protectors*. Google Patents.

Marion, B., Adelstein, J., Boyle, K. ea, Hayden, H., Hammond, B., Fletcher, T., Canada, B., Narang, D., Kimber, A., Mitchell, L., & others. (2005). Performance parameters for grid-connected PV systems. *Conference Record of the Thirty-First IEEE Photovoltaic Specialists Conference, 2005.*, 1601–1606.

Meribout, M., Tiwari, V. K., Herrera, J. P. P., & Baobaid, A. N. M. A. (2023). Solar Panel Inspection Techniques and Prospects. *Measurement*, 112466.

Momoh, J. A., & Button, R. (2003). Design and analysis of aerospace DC arcing faults using fast fourier transformation and artificial neural network. *2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No. 03CH37491)*, *2*, 788–793.

Mulongo, J., Atemkeng, M., Ansah-Narh, T., Rockefeller, R., Nguegnang, G. M., & Garuti, M. A. (2020). Anomaly detection in power generation plants using machine learning and neural networks. *Applied Artificial Intelligence*, *34*(1), 64–79.

Muñoz, J., Lorenzo, E., Mart\'\inez-Moreno, F., Marroyo, L., & Garc\'\ia, M. (2008). An investigation into hot-spots in two large grid-connected PV plants. *Progress in Photovoltaics: Research and Applications*, *16*(8), 693–701.

Natarajan, K., Bala, P. K., & Sampath, V. (2020). Fault detection of solar PV system using SVM and thermal image processing. *International Journal of Renewable Energy Research (IJRER)*, *10*(2), 967–977.

Nguyen, D. D., Lehman, B., & Kamarthi, S. (2009). Performance evaluation of solar photovoltaic arrays including shadow effects using neural network. *2009 IEEE Energy Conversion Congress and Exposition*, 3357–3362.

Pavan, A. M., Mellit, A., De Pieri, D., & Kalogirou, S. A. (2013). A comparison between BNN and regression polynomial methods for the evaluation of the effect of soiling in large scale photovoltaic plants. *Applied Energy*, *108*, 392–401.

Pereira, J., & Silveira, M. (2018). Unsupervised anomaly detection in energy time series data using variational recurrent autoencoders with attention. *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 1275–1282.

Pozsgay, A. (2016). *System for managing and controlling photovoltaic panels*. Google Patents.

Rahman, M. M., Khan, I., & Alameh, K. (2021). Potential measurement techniques for photovoltaic module failure diagnosis: A review. *Renewable and Sustainable Energy Reviews*, *151*, 111532.

Riley, D., & Johnson, J. (2012). Photovoltaic prognostics and heath management using learning algorithms. *2012 38th IEEE Photovoltaic Specialists Conference*, 1535–1539.

Rossi, B., Chren, S., Buhnova, B., & Pitner, T. (2016). Anomaly detection in smart grid data: An experience report. *2016 Ieee International Conference on Systems, Man, and Cybernetics (Smc)*, 2313–2318.

Sajun, A. R., Shapsough, S., Zualkernan, I., & Dhaouadi, R. (2022). Edge-based individualized anomaly detection in large-scale distributed solar farms. *ICT Express*, *8*(2), 174–178.

Sanz-Bobi, M. A., San Roque, A. M., De Marcos, A., & Bada, M. (2012). Intelligent system for a remote diagnosis of a photovoltaic solar power plant. *Journal of Physics: Conference Series*, *364*(1), 12119.

Schirone, L., Califano, F. P., Moschella, U., & Rocca, U. (1994). Fault finding in a 1 MW photovoltaic plant by reflectometry. *Proceedings of 1994 IEEE 1st World Conference on Photovoltaic Energy Conversion-WCPEC (A Joint Conference of PVSC, PVSEC and PSEC)*, *1*, 846–849.

Schripsema, J. (2014). *Reverse current fault prevention in solar panel*. Google Patents.

Sera, D., Teodorescu, R., & Rodriguez, P. (2008). Photovoltaic module diagnostics by series resistance monitoring and temperature and rated power estimation. *2008 34th Annual Conference of IEEE Industrial Electronics*, 2195–2199.

Smith, P., Furse, C., & Gunther, J. (2005). Analysis of spread spectrum time domain reflectometry for wire fault location. *IEEE Sensors Journal*, *5*(6), 1469–1478.

Spooner, E. D., & Wilmot, N. (2008). *Safety issues, arcing and fusing in PV arrays*.

Stellbogen, D. (1993). Use of PV circuit simulation for fault detection in PV array fields. *Conference Record of the Twenty Third IEEE Photovoltaic Specialists Conference-1993 (Cat. No. 93CH3283-9)*, 1302–1307.

Strobl, C., & Meckler, P. (2010). Arc faults in photovoltaic systems. *2010 Proceedings of the 56th IEEE Holm Conference on Electrical Contacts*, 1–7.

Takashima, T., Yamaguchi, J., Otani, K., Kato, K., & Ishida, M. (2006). Experimental studies of failure detection methods in PV module strings. *2006 IEEE 4th World Conference on Photovoltaic Energy Conference*, *2*, 2227–2230.

Takashima, T., Yamaguchi, J., Otani, K., Oozeki, T., Kato, K., & Ishida, M. (2009). Experimental studies of fault location in PV module strings. *Solar Energy Materials and Solar Cells*, *93*(6–7), 1079–1082.

Toshniwal, A., Mahesh, K., & Jayashree, R. (2020). Overview of anomaly detection techniques in machine learning. *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, 808–815.

Tsai, C.-W., Yang, C.-W., Hsu, F.-L., Tang, H.-M., Fan, N.-C., & Lin, C.-Y. (2020). Anomaly Detection Mechanism for Solar Generation using Semi-supervision Learning Model. *2020 Indo--Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)*, 9–13.

Vergura, S., Acciani, G., Amoruso, V., Patrono, G. E., & Vacca, F. (2009). Descriptive and Inferential Statistics for Supervising and Monitoring the Operation of PV Plants. *IEEE Transactions on Industrial Electronics*, *11*(56), 4456–4464.

Vlaminck, M., Heidbuchel, R., Philips, W., & Luong, H. (2022). Region-based CNN for anomaly detection in PV power plants using aerial imagery. *Sensors*, *22*(3), 1244.

Wang, Q., Paynabar, K., & Pacella, M. (2022). Online automatic anomaly detection for photovoltaic systems using thermography imaging and low rank matrix decomposition. *Journal of Quality Technology*, *54*(5), 503–516.

Wiles, J. (2008). Ground-fault protection is expanding. *Home Power*.

Wiles, J. C. (2012). Photovoltaic system grounding. *Southwest Technology Development Institute College of Engineering New Mexico State University October*.

Yuventi, J. (2013). DC electric arc-flash hazard-risk evaluations for photovoltaic systems. *IEEE Transactions on Power Delivery*, *29*(1), 161–167.

Zeng, C., Ye, J., Wang, Z., Zhao, N., & Wu, M. (2022). Cascade neural network-based joint sampling and reconstruction for image compressed sensing. *Signal, Image and Video Processing*, *16*(1), 47–54.

Zhao, Y. (2011). *Fault analysis in solar photovoltaic arrays*. Northeastern University.

Zhao, Y., De Palma, J.-F., Mosesian, J., Lyons, R., & Lehman, B. (2012). Line--line fault analysis and protection challenges in solar photovoltaic arrays. *IEEE Transactions on Industrial Electronics*, *60*(9), 3784–3795.

Zhao, Y., Yang, L., Lehman, B., de Palma, J.-F., Mosesian, J., & Lyons, R. (2012). Decision tree-based fault detection and classification in solar photovoltaic arrays. *2012 Twenty-Seventh Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, 93–99.

Ziar, H., Farhangi, S., & Asaei, B. (2014). Modification to wiring and protection standards of photovoltaic systems. *IEEE Journal of Photovoltaics*, *4*(6), 1603–1609.

APPENDIXES

PYTHON CODE

1. Import & Preprocessing

In [ ]:
```
import os
import pandas as pd
import numpy as np
import datetime
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
import plotly.express as px
import plotly.graph_objects as go
import matplotlib.dates as mdates
xformatter = mdates.DateFormatter('%H:%M') # for time
axis plots
import sklearn
from scipy.optimize import curve_fit
import warnings
warnings.filterwarnings('ignore')
```

In [ ]:
```
# Import all available data
df_gen1 =
pd.read_csv("C:/Users/Administrator/Desktop/Plant_1_Gener
ation_Data.csv")
df_gen2 = pd.read_csv("C:/Users/Administrator/Desktop
/Plant_2_Generation_Data.csv")
df_weather1 = pd.read_csv("C:/Users/Administrator/Desktop
/Plant_1_Weather_Sensor_Data.csv")
df_weather2 = pd.read_csv("C:/Users/Administrator/Desktop
/Plant_2_Weather_Sensor_Data.csv")
```

2. Preprocess And Merge Datasets

In [ ]:
```
# adjust datetime format
df_gen1['DATE_TIME'] =
pd.to_datetime(df_gen1['DATE_TIME'],format = '%d-%m-%Y
%H:%M')
df_weather1['DATE_TIME'] =
pd.to_datetime(df_weather1['DATE_TIME'],format = '%Y-%m-
%d %H:%M:%S')
```

```
# drop unnecessary columns and merge both dataframes
along DATE_TIME
df_plant1 = pd.merge(df_gen1.drop(columns =
['PLANT_ID']), df_weather1.drop(columns = ['PLANT_ID',
'SOURCE_KEY']), on='DATE_TIME')
```

In [ ]:

```
# add inverter number column to dataframe
sensorkeys = df_plant1.SOURCE_KEY.unique().tolist() #
unique sensor keys
sensornumbers = list(range(1,len(sensorkeys)+1)) # sensor
number
dict_sensor = dict(zip(sensorkeys, sensornumbers)) #
dictionary of sensor numbers and corresponding keys
# add column
df_plant1['SENSOR_NUM'] = 0
for i in range(df_gen1.shape[0]):
    df_plant1['SENSOR_NUM'][i] =
dict_sensor[df_gen1["SOURCE_KEY"][i]]
# add Sensor Number as string
df_plant1["SENSOR_NAME"] =
df_plant1["SENSOR_NUM"].apply(str) # add string column of
sensor name
```

In [ ]:

```
# adding separate time and date columns
df_plant1["DATE"] =
pd.to_datetime(df_plant1["DATE_TIME"]).dt.date # add new
column with date
df_plant1["TIME"] =
pd.to_datetime(df_plant1["DATE_TIME"]).dt.time # add new
column with time
# add hours and minutes for ml models
df_plant1['HOURS'] =
pd.to_datetime(df_plant1['TIME'],format='%H:%M:%S').dt.ho
ur
df_plant1['MINUTES'] =
pd.to_datetime(df_plant1['TIME'],format='%H:%M:%S').dt.mi
nute
df_plant1['MINUTES_PASS'] = df_plant1['MINUTES'] +
df_plant1['HOURS']*60
# add date as string column
df_plant1["DATE_STR"] = df_plant1["DATE"].astype(str) #
add column with date as string
```

In [ ]:

```
df_plant1.head()
```

In [ ]:

```
#import data
```

```
pd.DataFrame(df_plant1).to_csv("C:/Users/Administrator/De
sktop/66.csv")
```

There are two days with significantly lower temperature ("bad weather") on. Such events may be difficult to forecast without access to more weather data (air pressure, wind, humidity, cloud formation etc.) and advanced weather forecasting models.

3. Rule-based Fault Detection

During the data exploration we found a simple way to identify faulty equipment: If there is no power measured at the inverter during normal daytime operation, we can assume/identify equipment failure. Let's create a new column ("STATUS") that identifies faulty operation:

In [ ]:
```
# Function to check if time is during daytime operation
def time_in_range(start, end, x):
    """Return true if x is in the range [start, end]"""
    if start <= end:
        return start <= x <= end
    else:
        return start <= x or x <= end
# set normal daytime operation range
start=datetime.time(6,30,0) # sunrise
end=datetime.time(17,30,0) # sunset
# Create new column to check proper operation
# Return "Normal" if operation is normal and "Fault" if
operation is faulty
df_plant1["STATUS"] = 0
for index in df_plant1.index:
    if  time_in_range(start, end,
df_plant1["TIME"][index]) and
df_plant1["DC_POWER"][index] == 0:
        df_plant1["STATUS"][index] = "Fault"
    else:
        df_plant1["STATUS"][index] = "Normal"
```

In [ ]:
```
#import data
pd.DataFrame(df_plant1).to_csv("C:/Users/Administrator/De
sktop/3000.csv")
```

In [ ]:
```
fig = px.scatter(df_plant1, x="IRRADIATION",
y="DC_POWER", title="Fault Identification",
```

```
color="STATUS", labels={"DC_POWER":"DC Power (kW)",
"IRRADIATION":"Irradiation"})
fig.update_traces(marker=dict(size=3, opacity=0.7),
selector=dict(mode='marker'))
sns.set_style("white")
sns.set_theme()
fig.show()
```

3.1. Days With Faults

In [ ]:
```
df_plant1[df_plant1["STATUS"]==
"Fault"]["DATE"].value_counts()
```

In [ ]:
```
fig=px.bar(df_plant1[df_plant1["STATUS"]==
"Fault"]["DATE"].value_counts(), title="Fault Events:
Rule-based", labels={"value":"Faults", "index":"Date",
"SENSOR_NAME":"Inverter"})
fig.update(layout_showlegend=False)
```

3.2. Number Of Recorded Faults

In [ ]:
```
df_plant1.STATUS.value_counts()
```

In [ ]:
```
print("There are {} records of faulty operation!"
.format(df_plant1.STATUS.value_counts().Fault))
```

3.3. Inverters with faults

In [ ]:
```
df_plant1[df_plant1["STATUS"]==
"Fault"]["SENSOR_NAME"].value_counts()
```

In [ ]:
```
fig=px.bar(df_plant1[df_plant1["STATUS"]==
"Fault"]["SENSOR_NAME"].value_counts(), title="Inverter
Faults: Rule-based", labels={"value":"Faults",
"index":"Inverter", "SENSOR_NAME":"Inverter"})
fig.update(layout_showlegend=False)
```

3.4. Summary

In [ ]:
```
print("The most faults were recorded on {} and {}."
.format(df_plant1[df_plant1["STATUS"]==
"Fault"]["DATE"].value_counts().index[0],
```

```
df_plant1[df_plant1["STATUS"]==
"Fault"]["DATE"].value_counts().index[1]))
print("Inverter {} and {} had the most failures."
.format(df_plant1[df_plant1["STATUS"]==
"Fault"]["SOURCE_KEY"].value_counts().index[0],df_plant1[
df_plant1["STATUS"]==
"Fault"]["SOURCE_KEY"].value_counts().index[1]))
```

4. Fault Detection with Regression Models

```
In [ ]:
from sklearn.ensemble import RandomForestRegressor
from sklearn.linear_model import LinearRegression
from sklearn.svm import SVR
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.neural_network import MLPRegressor

In [ ]:
from sklearn.model_selection import GridSearchCV
rf_params = {
    'C': [1,10, 100],
    "kernel":['poly','rbf','sigmoid'],
    "epsilon":[0.01,0.1,1]}
clf = SVR(gamma='scale')
reg = GridSearchCV(clf, rf_params, cv=3,
scoring='neg_mean_squared_error')

In [ ]:
# Model
#reg = LinearRegression()
#reg = RandomForestRegressor(max_depth=8,
min_samples_leaf=5, max_features='auto')
#reg = MLPRegressor(hidden_layer_sizes=(32,32,16),
activation='relu', #  solver='adam', alpha=0.001,
max_iter=500, learning_rate='adaptive',#
learning_rate_init=0.01, shuffle=True)
#reg = make_pipeline(StandardScaler(), SVR(C=10.0,
epsilon=0.5))
# choose training data
train_dates = ["2020-05-16", "2020-05-17","2020-05-18"
,"2020-05-19", "2020-05-20", "2020-05-21"]
df_train =
df_plant1[df_plant1["DATE_STR"].isin(train_dates)]
#fit & predict
reg.fit(df_train[["IRRADIATION"]], df_train.DC_POWER)
prediction = reg.predict(df_plant1[["IRRADIATION"]])
# save prediction, residual, and absolute residual
df_train["Prediction"] =
reg.predict(df_train[["IRRADIATION"]])
```

```
df_train["Residual"] = df_train["Prediction"] -
df_train["DC_POWER"]
df_plant1["Prediction"] =
reg.predict(df_plant1[["IRRADIATION"]])
df_plant1["Residual"] = df_plant1["Prediction"] -
df_plant1["DC_POWER"]
df_plant1["Residual_abs"] = df_plant1["Residual"].abs()

In [ ]:
from sklearn.metrics import mean_absolute_error,
mean_squared_error
mean_squared_error(df_train["DC_POWER"],
df_train["Prediction"], squared=False)

In [ ]:
fig, axes = plt.subplots(nrows=1, ncols=1,
figsize=(12,5))
plt.scatter(df_plant1.IRRADIATION, df_plant1.DC_POWER,
label="Measured")
plt.scatter(df_plant1.IRRADIATION, df_plant1.Prediction,
color="r", label="LR Prediction")
sns.set_style("whitegrid")
plt.legend()
plt.xlabel("Irradiation (kW/m²)")
plt.ylabel("DC Power (kW)")
plt.title("linear Model Prediction")
plt.show()

In [ ]:
fig = px.scatter(df_plant1, x="DATE_TIME", y="DC_POWER",
title="Fault Identification: Linear model (Zoomed in)",
color="Residual_abs", labels={"DC_POWER":"DC Power (kW)",
"DATE_TIME":"Date Time", "Residual_abs":"Residual"},
range_x=[datetime.date(2020, 6, 1), datetime.date(2020,
6, 17)])
fig.update_traces(marker=dict(size=3, opacity=0.7),
selector=dict(mode='marker'))
fig.show()

In [ ]:
# choose data
day = "2020-06-07"
inverter1 = "2"
inverter2 = "22"
df_pred = df_plant1[(df_plant1["DATE_STR"] ==
day)].copy()
sns.set_style("whitegrid")
fig, axes = plt.subplots(nrows=1, ncols=2,
figsize=(15,5))
```

```
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter1].DC_POWER, label="Measured DC",
color="b", ax=axes[0])
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter1].Residual, label="Residual", color="g",
ax=axes[0])
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter1].Prediction, label="Predicted DC",
color="r", ax=axes[0])
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter2].DC_POWER, label="Measured DC",
color="b", ax=axes[1])
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter2].Residual, label="Residual", color="g",
ax=axes[1])
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter2].Prediction, label="Predicted DC",
color="r", ax=axes[1])
plt.gcf().axes[0].xaxis.set_major_formatter(xformatter) #
set xaxis format
plt.gcf().axes[1].xaxis.set_major_formatter(xformatter) #
set xaxis format
axes[0].set_xlabel("Time")
axes[1].set_xlabel("Time")
axes[0].set_ylabel("DC Power (kW)")
axes[1].set_ylabel("")
axes[1].set_ylim(-2500, 14000)
axes[0].set_title("Example: Normal Operation")
axes[1].set_title("Example: Fault Detected")
plt.show()

In [ ]:
fig, axes = plt.subplots(nrows=1, ncols=1,
figsize=(16,5))
inverter2= "22"
df_pred2 =
df_plant1[df_plant1["SENSOR_NAME"]==inverter2].copy()
sns.lineplot(df_pred2.DATE_TIME,df_pred2.Prediction,
label="Predicted DC", color="r")
sns.lineplot(df_pred2.DATE_TIME,df_pred2.DC_POWER,
label="Measured DC", color="b")
sns.lineplot(df_pred2.DATE_TIME,df_pred2.Residual-5000,
label="Residual (Offset)", color="g")
plt.xlabel("Date")
plt.ylabel("Power (kW)")
plt.title("Fault Detection: Inverter 22")
plt.show()
```

```
In [ ]:
# set confidence range for residual for fault
limit_fault=4000
# Create new column to check proper operation
# Return "Normal" if operation is normal and "Fault" if
operation is faulty
df_plant1["STATUS_LR"] = 0
for index in df_plant1.index:
    if  df_plant1["Residual"][index] > limit_fault:
        df_plant1["STATUS_LR"][index] = "Fault"
    else:
        df_plant1["STATUS_LR"][index] = "Normal"

In [ ]:
df_plant1[df_plant1["STATUS_LR"]==
"Fault"]["DATE"].value_counts()

In [ ]:
fig=px.bar(df_plant1[df_plant1["STATUS_LR"]==
"Fault"]["DATE"].value_counts(), title="Faults: linear
Model", labels={"value":"Faults", "index":"Date",
"SENSOR_NAME":"Inverter"}, )
fig.update(layout_showlegend=False)

In [ ]:
fig=px.bar(df_plant1[df_plant1["STATUS_LR"]==
"Fault"]["SENSOR_NAME"].value_counts(),
title="Underperformance & Faults: linear Model",
labels={"value":"Faults", "index":"Inverter",
"SENSOR_NAME":"Inverter"})
fig.update(layout_showlegend=False)

In [ ]:
df_plant1[df_plant1["STATUS_LR"]==
"Fault"]["SENSOR_NAME"].value_counts()

In [ ]:
fig = px.scatter(df_plant1, x="DATE_TIME", y="DC_POWER",
title="Underperformance & Faults: linear Model (Zoomed
in)", color="STATUS_LR", labels={"DC_POWER":"DC Power
(kW)", "DATE_TIME":"Date Time",
"STATUS_LR":"Status"},range_x=[datetime.date(2020, 6, 1),
datetime.date(2020, 6, 17)])
fig.update_traces(marker=dict(size=3, opacity=0.7),
selector=dict(mode='marker'))
fig.show()

In [ ]:
print("The most anomalies were recorded on {} and {}."
.format(df_plant1[df_plant1["STATUS_LR"]==
```

```
"Fault"]["DATE"].value_counts().index[0],
df_plant1[df_plant1["STATUS_LR"]==
"Fault"]["DATE"].value_counts().index[1]))
print("Inverter {} and {} had the most events of
failure/underperformance."
.format(df_plant1[df_plant1["STATUS_LR"]==
"Fault"]["SENSOR_NAME"].value_counts().index[0],df_plant1
[df_plant1["STATUS_LR"]==
"Fault"]["SENSOR_NAME"].value_counts().index[1]))
```

4.1. Non-linear Model

According to Hooda et al. (2018) the generated power of a photovoltaic cell can be modeled by the nonlinear equation \begin{equation} P(t) = a E(t) \left(1-b (T(t) + \frac{E(t)}{800} (c-20) - 25) - d \ln(E(t))\right) \end{equation} with irradiance $E(t)$, Temperature $T(t)$ and coefficients a,b,c,d.

In [ ]:

```
def func(X, a, b, c, d):
    '''Nonlinear function to predict DC power output from
Irradiation and Temperature.'''
    x,y = X
    x=x*1000
    y=y*1000
    return a*x*(1-b*(y+x/800*(c-20)-25)-d*np.log(x+1e-
10))
# fit function
p0 = [1.,0.,-1.e4,-1.e-1] # starting values
popt, pcov = curve_fit(func, (df_train.IRRADIATION,
df_train.MODULE_TEMPERATURE), df_train.DC_POWER, p0,
maxfev=5000)
sigma_abcd = np.sqrt(np.diagonal(pcov))
# predict & save
df_train["Prediction_NL"] = func((df_train.IRRADIATION,
df_train.MODULE_TEMPERATURE), *popt)
df_train["Residual_NL"] = df_train["Prediction_NL"] -
df_train["DC_POWER"]
df_plant1["Prediction_NL"] = func((df_plant1.IRRADIATION,
df_plant1.MODULE_TEMPERATURE), *popt)
df_plant1["Residual_NL"] = df_plant1["Prediction_NL"] -
df_plant1["DC_POWER"]
```

In [ ]:

```
from sklearn.metrics import mean_squared_error
```

In [ ]:
```
mean_squared_error(df_train["DC_POWER"],
df_train["Prediction_NL"], squared=False)
```

In [ ]:
```
fig, axes = plt.subplots(nrows=1, ncols=1,
figsize=(12,5))
plt.scatter(df_plant1.IRRADIATION, df_plant1.DC_POWER,
label="Measured")
plt.scatter(df_plant1.IRRADIATION,
df_plant1.Prediction_NL, color="r", label="NL
Prediction")
plt.legend()
plt.xlabel("Irradiation (kW/m²)")
plt.ylabel("DC Power (kW)")
plt.title("Nonlinear Model Prediction")
plt.show()
```

In [ ]:
```
# choose data
day = "2020-06-07"
inverter1 = "2"
inverter2 = "22"
df_pred = df_plant1[(df_plant1["DATE_STR"] ==
day)].copy()
fig, axes = plt.subplots(nrows=1, ncols=2,
figsize=(14,5))
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter1].DC_POWER, label="Measured DC",
color="b", ax=axes[0])
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter1].Residual_NL, label="NL Residual",
color="g", ax=axes[0])
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter1].Prediction_NL, label="NL Predicted
DC", color="r", ax=axes[0])
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter2].DC_POWER, label="Measured DC",
color="b", ax=axes[1])
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter2].Residual_NL, label="NL Residual",
color="g", ax=axes[1])
sns.lineplot(df_pred.DATE_TIME,df_pred[df_pred["SENSOR_NA
ME"] == inverter2].Prediction_NL, label="NL Predicted
DC", color="r", ax=axes[1])
plt.gcf().axes[0].xaxis.set_major_formatter(xformatter) #
set xaxis format
plt.gcf().axes[1].xaxis.set_major_formatter(xformatter) #
set xaxis format
axes[0].set_xlabel("Time")
```

```
axes[1].set_xlabel("Time")
axes[0].set_ylabel("DC Power (kW)")
axes[1].set_ylabel("")
axes[0].set_ylim(-3000, 14000)
axes[1].set_ylim(-3000, 14000)
axes[0].set_title("Example: Normal Operation")
axes[1].set_title("Example: Fault Detected")
plt.show()
```

In [ ]:

```
fig, axes = plt.subplots(nrows=1, ncols=1,
figsize=(14,5))
inverter2= "1"
df_pred2 =
df_plant1[df_plant1["SENSOR_NAME"]==inverter2].copy()
sns.lineplot(df_pred2.DATE_TIME,df_pred2.Prediction_NL,
label="NL Prediction", color="r")
sns.lineplot(df_pred2.DATE_TIME,df_pred2.DC_POWER,
label="Measured DC", color="b")
sns.lineplot(df_pred2.DATE_TIME,df_pred2.Residual_NL-
5000, label="NL Residual (Offset)", color="g")
plt.xlabel("Date")
plt.ylabel("Power (kW)")
plt.title("Fault Detection Example: Inverter
{}".format(inverter2))
plt.show()
```

In [ ]:

```
fig, axes = plt.subplots(nrows=1, ncols=1,
figsize=(14,5))
inverter2= "22"
df_pred2 =
df_plant1[df_plant1["SENSOR_NAME"]==inverter2].copy()
sns.lineplot(df_pred2.DATE_TIME,df_pred2.Prediction_NL,
label="NL Prediction", color="r")
sns.lineplot(df_pred2.DATE_TIME,df_pred2.DC_POWER,
label="Measured DC", color="b")
sns.lineplot(df_pred2.DATE_TIME,df_pred2.Residual_NL-
5000, label="NL Residual (Offset)", color="g")
plt.xlabel("Date")
plt.ylabel("Power (kW)")
plt.title("Fault Detection
Example:Inverter{}".format(inverter2))
plt.show()
```

5. Model Comparison

To compare the two models we can take a look at their respective residuals. The nonlinear model seems to perform slightly better than the linear model, especially at times of high irradiance.

In [ ]:
```
plt.figure(figsize=(8,8))
sns.scatterplot(df_train.Prediction, df_train.Residual,
color="b", label="LI Residual")
sns.scatterplot(df_train.Prediction_NL,
df_train.Residual_NL, color="r", label="NL Residual")
axes = plt.gca()
plt.ylabel("Residual")
plt.xlabel("Predicted DC Power")
plt.title("Model Comparison")
plt.show()
```

5.1. NL Fault Detection

Let's now use the irradiance and temperature data to predict the expected DC power with the nonlinear model. This allows us to identify addtional anomalies by comparing the measured DC power with the prediction.

The additional anomalies indicate equipment underperformance or need for maintenance.

In [ ]:
```
# set confidence range for residual for fault
limit_fault=4000
# Create new column to check proper operation
# Return "Normal" if operation is normal and "Fault" if
operation is faulty
df_plant1["STATUS_NL"] = 0
for index in df_plant1.index:
    if  df_plant1["Residual_NL"][index] > limit_fault:
        df_plant1["STATUS_NL"][index] = "Fault"
    else:
        df_plant1["STATUS_NL"][index] = "Normal"
```

In [ ]:
```
fig=px.bar(df_plant1[df_plant1["STATUS_NL"]==
"Fault"]["DATE"].value_counts(), title="Faults: Nonlinear
```

```
Model", labels={"value":"Faults", "index":"Date",
"SENSOR_NAME":"Inverter"}, )
fig.update(layout_showlegend=False)
```

In [ ]:

```
fig=px.bar(df_plant1[df_plant1["STATUS_NL"]==
"Fault"]["SENSOR_NAME"].value_counts(),
title="Underperformance & Faults: Nonlinear Model",
labels={"value":"Faults", "index":"Inverter",
"SENSOR_NAME":"Inverter"})
fig.update(layout_showlegend=False)
```

In [ ]:

```
df_plant1[df_plant1["STATUS_NL"]==
"Fault"]["SENSOR_NAME"].value_counts()
```

In [ ]:

```
df_plant1[df_plant1["STATUS_NL"]==
"Fault"]["DATE"].value_counts()
```

In [ ]:

```
fig = px.scatter(df_plant1, x="DATE_TIME", y="DC_POWER",
title="Underperformance & Faults: Nonlinear Model (Zoomed
in)", color="STATUS_NL", labels={"DC_POWER":"DC Power
(kW)", "DATE_TIME":"Date Time",
"STATUS_NL":"Status"},range_x=[datetime.date(2020, 6, 1),
datetime.date(2020, 6, 17)])
fig.update_traces(marker=dict(size=3, opacity=0.7),
selector=dict(mode='marker'))
fig.show()
```

In [ ]:

```
print("The most anomalies were recorded on {} and {}."
.format(df_plant1[df_plant1["STATUS_NL"]==
"Fault"]["DATE"].value_counts().index[0],
df_plant1[df_plant1["STATUS_NL"]==
"Fault"]["DATE"].value_counts().index[1]))
print("Inverter {} and {} had the most events of
failure/underperformance."
.format(df_plant1[df_plant1["STATUS_NL"]==
"Fault"]["SOURCE_KEY"].value_counts().index[0],df_plant1[
df_plant1["STATUS_NL"]==
"Fault"]["SOURCE_KEY"].value_counts().index[1]))
```

RESUME

Personal Information

Surname, name : Qais Ibrahim Ahmed Alshammary

Nationality : Iraqi

Education

Degree	Education Unit	Graduation Date
Master	Electrical – Electronic Engineering	2023
Bachelor	Electrical – Power Engineering	2003
High School	Qutayba Secondary School	1998

Work Experience

Year	Place	Title
2015 Till now	Alhamediah Power Plant 4*169 MW	Power Plant Manager
2004 - 2015	Protection Engineer in Ministry of Electrcity	Field Engineer

Foreing Language:

Arabic and English

Publications:

Implementing ML in Detection of Solar Power Plants Anomalies using a Hybrid Support Vector Machine with Grey Wolf Optimization Algorithm

Hobbies:

Reading, Scientific Research, Footbal

systems

an Open Access Journal by MDPI

CITESCORE 4.3

IMPACT FACTOR 2.895

CERTIFICATE OF ACCEPTANCE

Certificate of acceptance for the manuscript (**systems-2254283**) titled:

Development of a Hybrid Support Vector Machine with Grey Wolf Optimization Algorithm for Detection of the Solar Power Plants Anomalies.

Authored by:

Qais Ibrahim Alshammary; Hani Attar; Ayman Amer; Mohanad A. Deif; Ahmed A. A. Solyman

has been accepted in *Systems* (ISSN 2079-8954) on 30 April 2023

MDPI Academic Open Access Publishing since 1996

Basel, April 2023