Levent YAVUZ

A Ph.D. Thesis

AGU 2023

# Heuristic Vectorized Learning Method Based PV Forecasting by Using Image Recognition-Based Sky Camera Integration Within Sensor Set

A THESIS
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF ABDULLAH GUL UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Levent YAVUZ
July 2023

# Heuristic Vectorized Learning Method Based PV Forecasting by Using Image Recognition-Based Sky Camera Integration Within Sensor Set

A THESIS
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE OF ABDULLAH GUL UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Levent YAVUZ
July 2023

**SCIENTIFIC ETHICS COMPLIANCE**

I hereby declare that all information in this document has been obtained in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name-Surname: Levent YAVUZ

Signature :

**REGULATORY COMPLIANCE**

Ph.D. thesis titled "Heuristic Vectorized Learning Method Based PV Forecasting by Using Image Recognition-Based Sky Camera Integration Within Sensor Set " has been prepared in accordance with the Thesis Writing Guidelines of the Abdullah Gul University, Graduate School of Engineering & Science.


Prepared By
Levent YAVUZ

Advisor
Assoc. Prof. Ahmet ÖNEN

Head of the Electrical and Computer Engineering Program
Assoc. Prof. Zafer AYDIN

## ACCEPTANCE AND APPROVAL

Ph.D. thesis titled "Heuristic Vectorized Learning Method Based PV Forecasting by Using Image Recognition-Based Sky Camera Integration Within Sensor Set" and prepared by Levent YAVUZ has been accepted by the jury in the Electrical and Computer Engineering Graduate Program at Abdullah Gül University, Graduate School of Engineering & Science.

… / / 2023

**JURY:**

Advisor : Assoc. Prof. Ahmet ÖNEN

Member : Assist. Prof. Bekir Hakan Aksebzeci

Member : Assist. Prof. Serkan BAHÇECİ

Member : Prof. Dr. Ferhat Daldaban

Member : Assoc. Prof. Nurettin Üstkoyuncu

**APPROVAL:**

The acceptance of this Ph.D. thesis has been approved by the decision of the Abdullah Gül University, Graduate School of Engineering & Science, Executive Board dated / / 2023 and numbered

/ /

Graduate School Dean
Prof. Dr. İrfan ALAN

ABSTRACT

## Heuristic Vectorized Learning Method Based PV Forecasting by Using Image Recognition-Based Sky Camera Integration Within Sensor Set

Levent YAVUZ
Ph.D. in Electrical and Computer Engineering
Advisor: Assoc. Prof. Ahmet ÖNEN
July 2023

In order to ensure the continuity of demand and production balance, the use of renewable energy resources (RES) by countries will increase in the near future. Solar power generation is important for the integration of renewable energy into the power grid, but it can cause problems in power systems due to the uncertain and intermittent nature of solar power. Deep learning methods provide promising results in solar energy prediction, but the performance of these models depends on the initial weights assigned to the network. In this thesis, a novel weight initialization method, the Heuristic Vectorised Learning method, which takes into account certain characteristics of solar generation data has been proposed. This method aims to achieve better accuracy in solar forecasting by combining a statistical approach with a method based on deep learning. The method was compared with other commonly used methods such as Xavier, LeCun, He and Random, and it was seen that the proposed method performed better. Overall, the proposed weight initialization method provides significant benefits for solar forecasting applications in the context of renewable energy integration into the power grid. So, to reach higher accuracy, monitoring the sky is the best option for intra-day forecasts. Therefore, a hybrid model was created for photovoltaic generation prediction by using it together with environmental sensor data. The proposed method and panel shading model achieve higher accuracy values at the Abdullah Gül University campus in Kayseri. The proposed system provides a reliable PV energy forecast for the intraday energy markets.

*Keywords: solar energy, forecasting, deep learning,* h*euristic vectorised learning method*, *initialization, artificial neural network*

# ÖZET

## Görüntü Tanıma Tabanlı Gökyüzü Kamerası Entegrasyonunu Kullanarak Sezgisel Vektörize Öğrenme Yöntemine Dayalı PV Tahmini

Levent YAVUZ
Elektrik ve Bilgisayar Mühendisliği Anabilim Dalı Doktora
Doktora Tez Yöneticisi: Doç. Dr. Ahmet ÖNEN
Temmuz-2023

Talep ve üretim dengesinin sürekliliğini sağlamak için, ülkeler yenilenebilir enerji kaynaklarını (YEK) kullanımı yakın gelecekte artış gösterecektir. Güneş enerjisi üretimi, yenilenebilir enerjinin elektrik şebekesine entegrasyonu için önemlidir, ancak güneş enerjisinin belirsiz ve kesintili doğası nedeniyle güç sistemlerinde problemlere neden olabilir. Derin öğrenme yöntemleri, güneş enerjisi tahmininde umut verici sonuçlar sağlamaktadır, ancak bu modellerin performansı ağa atanan başlangıç ağırlıklarına bağlıdır. Bu çalışmada, Sezgisel Vektörleştirilmiş Öğrenme yöntemi olarak adlandırılan yeni bir ağırlık başlatma yöntemi önerilmektedir. Bu yöntem, istatistiksel bir yaklaşımı derin öğrenmeye dayalı bir yöntemle birleştirerek güneş tahmininde daha iyi doğruluk elde etmeyi amaçlamaktadır. Yöntemin Xavier, LeCun, He ve Random gibi yaygın olarak kullanılan başka yöntemlerle karşılaştırması yapılmıştır ve önerilen yöntemin daha iyi performans gösterdiği görülmüştür. Genel olarak, önerilen ağırlık başlatma yöntemi, elektrik şebekesine yenilenebilir enerji entegrasyonu bağlamında güneş tahmini uygulamaları için önemli faydalar sağlamaktadır. Dolayısıyla, çevresel sensör verileriyle birlikte kullanılarak fotovoltaik üretim tahmini için bir hibrit model oluşturulmuştur. Önerilen yöntem ve panel gölgeleme modeli, Kayseri ilinde bulunan Abdullah Gül Üniversitesi yerleşkesinde daha yüksek doğruluk değerleri elde etmektedir. Önerilen sistem, gün içi enerji piyasaları için güvenilir bir PV enerji tahmini sağlar.

*Anahtar Kelimeler: Anahtar Kelimeler: güneş enerjisi, tahmin, derin öğrenme, başlatma, Sezgisel Vektörleştirilmiş Öğrenme*, yapay *sinir ağları*

# Acknowledgements

I would like to express my gratitude to my advisor for her time, support, patience, and valuable assistance throughout the study, especially in bureaucratic processes.

Assoc. Prof. Ahmet ÖNEN was instrumental in guiding me to complete the thesis. I am very grateful for the opportunity to work with him and for his insightful support throughout.

With the same level of gratitude, I would also like to thank the entire thesis committee members and juries, including the replaced ones, for their valuable time, comments, and understanding. In addition, I would like to thank all my colleagues who have supported me directly or indirectly during this long period that once seemed to me to never end.

Finally, I owe a deep debt of gratitude to everyone who has supported and encouraged me in my personal life. Especially to my precious mother, for their endless support throughout my life.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| AI | Artifical Intelligence |
| CVPP | Commercial Virtual Power Plant |
| DER | Distributed Energy Resources |
| DL | Deep Learning |
| DSO | Distribution System Operator |
| DT | Decision Tree |
| EMS | Energy Management System |
| ESS | Energy Storage System |
| EV | Electric Vehicle |
| HVLM | Heuristic Vectorized Learning Method |
| KNN | k-Nearest Neighbors |
| LDL | Linear Discriminant Analysis |
| LR | Linear Regression |
| MG | Microgrid |
| MILP | Mixed Integer Linear Programming |
| ML | Machine Learning |
| NB | Naïve Bayes |
| P2P | Peer-to-peer |
| PV | Photovoltaic |
| RES | Renewable Energy Sources |
| SVM | Support Vector Machine |
| TSO | Transmission System Operator |
| TVPP | Technical Virtual Power Plant |
| VM | Virtual Machine |
| VPP | Virtual Power Plant |

*To my loved ones...*

# Chapter 1

# Introduction

The continued strong development of distributed energy resources (DERs) provides a great opportunity for renewable energy investors around the world. The worldwide DERs integration grows the average rate of 20% by the end of the 20th Century [1]. Owing to priorities on carbon footprint reduction and harnessing energy from alternative sources than fossil fuel, DERs integration with existing power grid will be kept to increase more for the time to come. While these growths provide many advantages, it creates new challenges to manage the grid in an effective way. From the experiences of transmission system operator (TSO) and distribution system operator (DSO), some problems occur while integrating the DERs with an existing grid such as transmission congestion, voltage and frequency stabilities and reliability problems due to uncertain and intermittency natures of DERs.

A microgrid is a localized group of energy sources and loads that may operate at grid connected or islanded modes. The concept of microgrid is getting popular since last decade and there are many microgrids actively operating in different parts of the globe. The major investment in a microgrid is on its DERs. In many microgrids, the operators have to handle problems coming up with DERs; otherwise, green energy should be thrown away instead of being utilized. These problems create a new research area to seek solutions for integration of DERs without creating grid stability and reliability problems. One of the new solutions of eliminating of DERs negative impacts is through the transformation of microgrid to VPP. VPP coordinates all DERs as in a single agent to integrate them into the grid without compromising the grid stability and reliability, adding many other additional benefits and opportunities to consumers, prosumers and grid operators [2].

With the gain experienced from smart grid concept, has been studied on decades, VPP can be implemented easily and successfully which has already been tested in some country. Some of the smart grid technologies that may help to integrate VPP are intelligence algorithm, i.e. power generation, transmission and distribution, and demand response by using customer participation with the usage of advanced communications such as Internet protocols. Web to Energy project [3] is one of the biggest developments on smart grid that can easily adapted to VPP concepts. Communication of network of physical devices has enabled project of the Internet of things (IoT) [4]. IoT allows different devices to be sensed, communicate with each other and also controllable from remote locations. This method is applied for direct integration between intelligent devices with computer-based software offering advanced connectivity among the devices. Similar to IoT, VPP combines, communicates and behaves such as a neural network for each different DERs agents. Figure 1.1 demonstrates simply how householders share their energy within a VPP. This system aggregates all DERs and other units inside the system. Figure 1.2 shows how units are clustered and connected with centralized VPP. These clusters interact with each other in order to behave as just as one unit. The localized control center emerges as a self-organized intelligent solution. To obtain a self-intelligent system and make a decision, there are many optimization algorithms that have been developed. A detailed review study by comparing these optimization algorithms is discussed in [5]. How the payment of energy will be dispatched among generation, transmission and distribution companies is getting complicated with increased integration of DERs into the grid. However, with the usage of VPP, monitoring and coordination of DERs will be much easier that helps to put market prices out easily. Hooshmand *et al. and* Shabanzadeh *et al.* in [6, 7] show how to investigate event-based scheduling and calculation of stochastic market price in the distribution network.


**Figure 1.1 Household energy sharing scheme in a VPP**

VPP also helps to reduce the losses on the network. Conveying the load along a long path causes large losses, but with VPP technology, the consumer can be fed by the closest distributed generation (DG) that provides minimizing line losses, prevents overloading the system, and delivers electricity to consumers at a cheaper price. Recently, consumers have become active in production by using DERs named as prosumer that means consumers also produce. By being active in production makes generation forecast complicated due to uncertainty inside the DERs. Authors in [3] investigate all units (DERs, batteries, flexible loads etc.) in a comprehensive way to see the effect of uncertainties on forecasting. These uncertainties can be solved by using complex equations and computer programmes to make predictions more accurate. In a study [8], Wei *et al.* offered that the consumption statistics can be forecasted with artificial intelligence (AI) software that will be highly used in VPP in near future.

Available review papers found in the literature summaries advantages and disadvantages of VPP. Paper [5] describes the general structure and optimization of VPPs. Different implementation techniques and scheduling of VPPs have been discussed in [9, 10]. There are very limited technical review papers that have been published related to All kinds of power engineering applications (in point of VPP) and how these applications affect traditional grid systems are architecture and advancements of VPP and none of them comprehensively mentioned technical problems highlighted above which triggered the authors to this work. Discussed in this technical review paper. In particular, technical articles, which contain practice and optimization-oriented studies, have been examined and different popular theoretical approaches are compared in tabular form. In addition, well known optimization algorithms are explained from different aspects and discussed effects of AI on optimization methods.

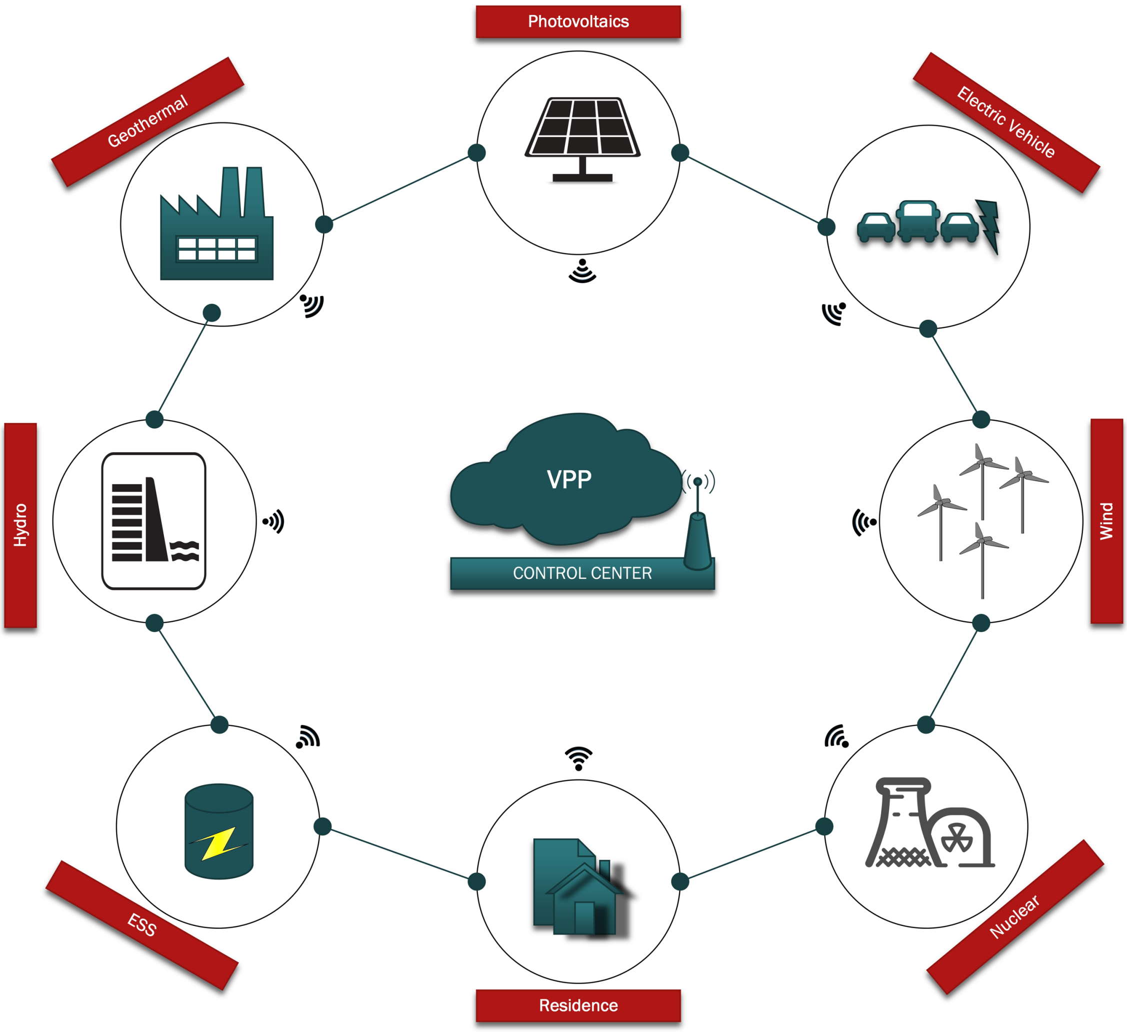



**Figure 1.2 Centralized VPP**

## 1.1 VPP Classifications

The VPP can be classified into two different categories. First one is commercial VPP (CVPP), the second one is technical VPP (TVPP) [11]. One of the main problems is that it is not clear how participants can interact among themselves and with the DSO that has all grid infrastructures. VPPs generate their own electricity and transfer this energy by using transmission line that does not belong to VPP owners, so the cost of this energy transfer needs to take into account as well. This topic requires innovative research itself to decide what kind of billing system is required to solve this chaos. Another technical aspect of the implementation of VPP is that whether the system is currently used as microgrid (islanded mode) or grid-connected mode that effects hardness level and cost of implementation and dispatching bills among participants.

### 1.1.1 Commercial VPP

The main purpose of CVPP is an economic optimization. It includes financial risks: cost, optimized revenue for exchange energy, combine economical paradigms with intelligent grid services and represents bid-offer tables. In general, studies in this area are based on risk-management methodologies. Developed models include how VPP users have agreed with each other, partnership agreements with distribution companies, also mutual agreements between different VPP groups and energy marketing issues. Thus, different mathematical methods and computer algorithms have been developed for solving such complex situations. The fundamental of most studies is relying on stochastic or deterministic approximations. However, some programmers may become unsuccessful due to the variable and flexible nature of the results by using deterministic approximate methods. It is found that recently the researchers have the tendency to focus on stochastic methods. A detailed description of these topics is given in [12], where stochastic approximation methods and solutions are also discussed for CVPP.

### 1.1.2 Technical VPP

As the name suggests, TVPP defines comprehensively about complex calculations, technical applications, storage and optimization. Mostly financial issue, monitoring and fault detection schemes are the topic of interest. This classification deals with power flow optimization, communication protocols within smart grids, technical feasibility solutions and some fuzzy algorithms about generation and consumption. In addition, security is one of the main challenges, because VPP has to keep personal information in private. One of the most troublesome problems of an intelligent system is the hostile cyber attacks and viruses. For example, consumer's personal information has to be preserved in case of any cyber attack, so TVPP deals with this situation, and because of these features it becomes significant in communication system [13].

### 1.1.3 Highlights of TVPPP an CVPP

Two different types of VPP can be characterized by the following [14]:

- Energy management systems (EMSs) offer different tariffs for consumers. Authors demonstrate calculation and implementation of these tariffs [15] (TVPP).
- Demand forecasting process deals effectively with complex commercial proposals since nowadays energy market has become such as the stock market (TVPP) [16].
- Tariffs have to be updated by manufacturers and inform to customers about new tariffs and energy rates [17] (CVPP).
- New tariffs should be updated, so tariff updater officer has to be responsible [14] (CVPP).
- Communication of VPP units occurs between each other. An implementation technique based on information and communication technology (ICT) is proposed [18] (TVPP).
- Safety and reliability parameter (TVPP).
- Generation and consumption control, optimization process and

  stability control (TVPP).

- Grid imbalance must be overcome (TVPP).
- Different weather situations are foreseen. Photovoltaic (PV) and

  wind generators depend on weather, so the producer must deal with it by using different software, complex algorithms and using satellite images (TVPP). Some forecasting applications are demonstrated by using stochastic simulation methods [14, 19, 20] (TVPP).
  • Storage control and optimization are for all units. Energy storage system (ESS) linkage application affects grid balance and change energy markets, (TVPP).

### *1.1.3* Use of Artificial Intelligence Algorithms in VPP

Increased penetrations of DERs and renewable energy sources (RESs) require the appropriate communication by using general architecture, as an example, the smart grid architecture model (SGAM) [13] can be referred that offers simplicity for any proposed smart grid architecture. The SmarterEMC2 project [22] which is used as SGAM, aims to devise new business model, increases RES usage and links ICT tools between DER and VPP.

AI can be applied to the very large area that is based on a mathematical model, the computational power of process and application. The main idea is to find maximum and/or minimum results that depend on objective functions for highly complex problems. It is specifically used as giving input values, provides what output values would be suitable for a given system. In short, it tries to find optimum results. In power system, AI algorithms are widely used in optimum power flow, loss minimization, generation cost reduction etc. As VPP involves parties such as DGs, DSO, ESS and loads, there are numerous applications of AI algorithms within VPP framework such as reducing unbalanced power flow, capacity factor, system interruption, forecasting of load and generations, sizing of RES and ESS elements and profit maximization [23]. Besides advantages of VPP technology, people have faced many problems based on AI issue. There have been too many AI techniques that can be used for different purposes with the pros and cons. Accurate calculation of energy production and consumption is one of the most important steps in order to sustain and balance demands and generations. There are many surveys that have been conducted to get real consumption statistics to improve prediction accuracy. Why energy demand prediction is very important for producers? The answer is bringing prices down and to get more sustainable and balanced power flow. Energy production and consumption example is demonstrated in [8]. Prices are varying as demand changes on installed power. The other most significant goals are to equalize demands and generations. To achieve that, AI- based algorithm is getting popular which can be expressed using Figure 1.3

Besides demand forecasting, production and price forecasting issues are also in the point of interest. Price forecasting comes into prominence at energy market. The main point is differences between the cost of production and demand.

The relationship between production and demand costs is demonstrated in [38]. Renewable energy resources such as PV and wind depend on weather conditions. Thus, we need to use advanced mathematical methods, physics, and some specialized software algorithms to forecast them accurately. Table 1.2 summarizes energy forecasting & load forecasting algorithms and their usage in VPP. All the algorithms shown in Table 1.2 are highly used as a mathematical model for optimization and forecasting in a power system. When the

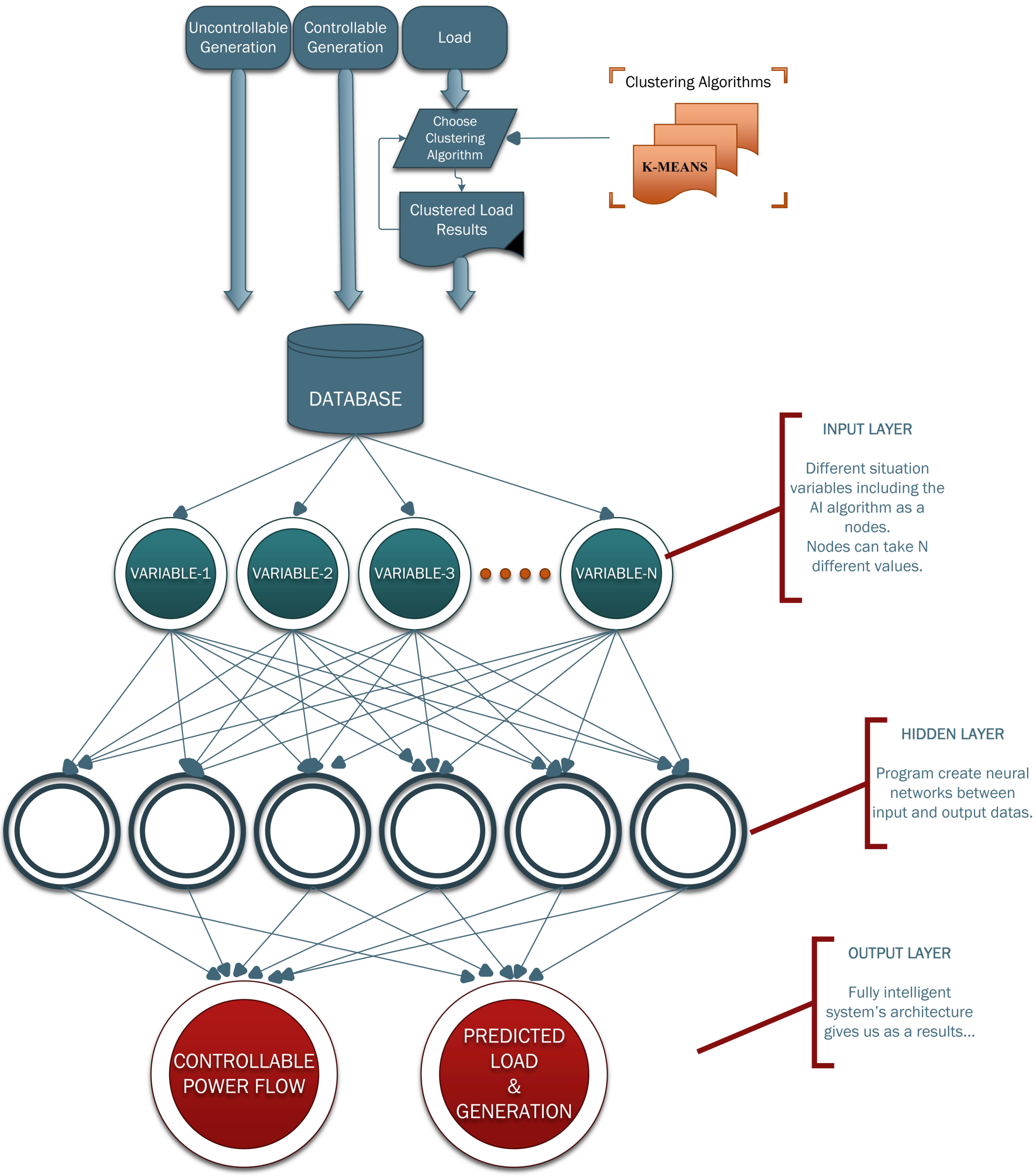



**Figure 1.3 Power flow and predicted power-load on VPP**

status of power system changes (which may include a new power supply), then optimization and forecasting algorithms are updated by the developer. In some cases, the developer will have completely irrelevant consequences and will have to go through a completely different algorithm. Instead of evaluating the problems and the situation on the background with a single solution method and approximation, it is required to simulate the system by using AI that can exhibit flexible behavior and give correct results adopting same flexibility by examining the behavior as case varies.

## 1.2 Motivation and Problem Statement

The power system is one of the biggest devices that have been developed by a human being. A number of DER units have been integrated into the grid over time, and then operation procedures changed. More complex algorithms and high-level computers are used for weather-dependent consumption prediction. Furthermore, the 2003 operation of VPP provides less complex algorithms and secures more financial profit for consumers and producers. In this chapter, the essential elements of the VPP are introduced such as VPP classifications (TVPP and CVPP), DER–TSO–DSO linkage through VPP, grid services and worldwide case studies of VPP projects have been reviewed in a comprehensive way.

In addition, features of VPP on power systems such as communication, different optimization algorithms and forecasting methods both of production and consumption, reactive power control, gateway technology and frequency control were described. So, the accurate estimation of solar power generation is crucial for transmission and distribution aggregators to manage the integration of renewable energy into the electricity grid.

Deep learning methods, especially artificial neural networks, have been shown to provide promising results in solar energy prediction. However, the performance of these models is highly dependent on the initial weights assigned to the network.

In this thesis, a novel weight initialization method, the Heuristic Vectorised Learning method, which takes into account certain characteristics of solar generation data has been proposed. The proposed method combines a statistical approach with a deep learning-based approach to achieve better accuracy in solar forecasting. The comparison between a proposed weight initialization method and several commonly used techniques including Xavier, LeCun, He, and Random was conducted in this study, utilizing a PV forecasting dataset. The results show that the proposed method outperforms other methods in terms of both accuracy and computational efficiency. Overall, the proposed weight initialization method can provide significant benefits for solar forecasting applications in the context of renewable energy integration into the power grid.

## 1.3 Thesis Outline

The remainder of the thesis is organized as follows:

The second chapter proposes machine learning algorithms applied on PV power system

scheme for a VPP usage.

The third chapter introduces initialization, LeCun, He, Xavier(Glorot), Random, methods in which most popular and well known methods investigated. After explanation of mathematical background they have been applied on PV power generation system to reach highest accuracy and compared with each other. So at the highest accuracy results has been showed on Tables.

In Chapter 4, a new approach, Heuristic Vectorized Learning Method (HVLM), has been introduced. After its mathematical background explanation, applied on PV power generation system and compared with other well-known initialization methods. Accuracy results show the proposed algorithm shows higher accuracy when compared other tehcniques and much less computational power needed on it. In addition the proposed method offers fastest running system to the users.

## 1.4 Thesis Output

This doctoral thesis presents the outputs of four important articles. The first published article, titled 'Adaptive Fault Detection Scheme Using an Optimized Self-healing Ensemble Machine Learning Algorithm,' introduces an adaptive fault detection method using an optimized self-healing ensemble machine learning algorithm. The second published article, 'Transformation of microgrid to virtual power plant–a comprehensive review,' comprehensively reviews the transformation of microgrid systems into virtual power plants.

The other two articles have been submitted to a publishing organization. The study titled 'Investigation on Cloud Thickness and Movement into PV Forecasting in the region of Kayseri, Turkey' examines the impact of cloud thickness and movement on photovoltaic forecasting in the Kayseri region of Turkey. Finally, the article 'Statistical-Based Heuristic Vectorised Learning Method for PV Forecasting' presents a statistical-based heuristic vectorized learning method for photovoltaic forecasting.

These four articles form the core outputs of the thesis, with the published ones being available to the scientific community, while the other two are currently awaiting review by the publishing organization.

1. Levent Yavuz; Ahmet Onen; S.M. Muyeen; Innocent Kamwa, "Transformation of microgrid to virtual power plant – a comprehensive review", IET Generation, Transmission & Distribution, Volume 13, Issue 11, p. 1994 – 2005, 04 June 2019.
2. L. Yavuz, A. Soran, A. Onen, X. Li and S. M. Muyeen, "Adaptive Fault Detection Scheme Using an Optimized Self-healing Ensemble Machine Learning Algorithm," in CSEE Journal of Power and Energy Systems, vol. 8, no. 4, pp. 1145-1156, July 2022.
3. Levent Yavuz, Ahmet Onen, "Investigation on Cloud Thickness and Movement into PV Forecasting in region of the Kayseri, Türkiye", is under review at IEEE Transaction on Power Systems, 2023.
4. Levent Yavuz, Ahmet Onen, "Statistical-Based Heuristic Vectorised Learning Method for PV Forecasting", is under review at IEEE Transaction on Power Systems, 2023.
5. T. S. Ustun, S. M. S. Hussain, L. Yavuz, and A. Onen, "Artificial Intelligence Based Intrusion Detection System for IEC 61850 Sampled Values under Symmetric and Asymmetric Faults," *IEEE Access*, vol. 9, pp. 56486–56495, 2021, doi: 10.1109/ACCESS.2021.3071141.
6. B. Kolukisa *et al.*, "Coronary Artery Disease Diagnosis Using Optimized Adaptive Ensemble Machine Learning Algorithm," *Int J Biosci Biochem Bioinforma*, vol. 10, no. 1, pp. 58–65, 2020, doi: 10.17706/ijbbb.2020.10.1.58-65.

7. L. Yavuz, A. Soran, A. Onen, and S. M. Muyeen, “PSO Supported Ensemble Algorithm for Bad Data Detection Against Intelligent Hacking Algorithm,” *Front Energy Res*, vol. 9, Jul. 2021, doi: 10.3389/fenrg.2021.649460.
8. L. Yavuz, A. Soran, A. Onen, and S. M. Muyeen, “Machine learning algorithms against hacking attack and detection success comparison,” in *2020 2nd International Conference on Smart Power and Internet Energy Systems, SPIES 2020*, Institute of Electrical and Electronics Engineers Inc., Sep. 2020, pp. 258–262. doi: 10.1109/SPIES48661.2020.9243033.

# Chapter 2

# Machine Learning Algorithms Application on PV Power Plants

ML applications are not the only used for power generation predictions but also it has been used large area on pv power systems. Such as, cyber security, fault detection, load predictions as well. And in this chapter we will show the fault detection application on power system and also applied Self-Healing Ensemble Machine Learning Algorithm. So, at the end we will show the benchmark of whole system results.

Legacy fault detection methods for islanding operation use various machine learning algorithms to activate the alarm system and perform controlled islanding. The main objective of this work is to propose a new cost-efficient, adaptive, and self-healing algorithm in real time that detects faults in a short period with high accuracy, even in the situations when it is difficult to detect. For this purpose, power system failures are simulated by using PSCAD-Python co-simulation. Rather than using traditional machine learning (ML) algorithms or hybrid signal processing techniques, a new framework based on optimization enabled weighted ensemble method is developed that combines essential ML algorithms differently. In the proposed method, the system will select and compound appropriate ML algorithms automatically. One of the salient features of this study is that the proposed solution works on real-time raw data without using any pre-computational techniques or pre-stored information. Therefore, the proposed technique will be able to work on different systems, topologies, or data collections. The proposed fault detection technique is validated using modified IEEE-14 bus considering networks faults which are difficult to detect.

## 2.1 Introduction

The rapid growth of energy consumption in the world brings new technologies together in power systems. In addition to traditional systems for energy generation, small, medium and large-scale on-site power generating platforms, like wind turbines or other renewable energy systems, have emerged recently. The inclusion of the massive renewable penetration causing the power system more vulnerable [1].

More importantly, increasing more renewable penetrations reduces rotating inertia. For example, the total inertia currently available in South Australia network is around 16,200 MWs. In 2017, without the Northern Power Station and Torrens Island 'A' Power Station, the total available inertia would reduce to around 10,000 MWs [2] due to the inclusions of renewable sources, which looks alarming. This may cause system instability or even a blackout, in case of catastrophic events or grid fault conditions [3-4] and faults are needed to be cleared quickly. Therefore, special precautions must be taken to avoid any greater outage and/or cascaded blackouts. Considering these, many fault detection techniques have been developed recently and reported in power system literature [5]-[6].

In the traditional power system, primarily the protective relay receives the voltage, current, and frequency information from transmission/distribution systems using instrumentation transformers. That information is typically processed by the protective relay to act in case of emergency or abnormal conditions for the desired tripping time. The relay logic algorithm takes the decision of whether to trip open or to close the circuit breaker. In the distribution system, the protective relays require a large fault current to detect the faulty condition. This is problematic for the modern distribution system which connects various Distributed Generations (DGs) including renewable sources, microgrids, and virtual power plants. Most of the DGs, nowadays, are equipped with inverter technology which typically contributes two times of per unit rated current, as rule of thumb [7]. This is troublesome because the current level may not be sufficient to trigger the relaying action and it puts the inverter equipped modern distribution system at risk.

Many fault detection mechanisms have been depicted in the power system. The real-time data acquisition device like Frequency Disturbance Recorder (FDR), optimally placed Phasor Measurement Units (PMUs), and many other equipment's are used to detect power system faults [8]. Line outage detection using phasor angle measurements are reported in [9]. Fault detection occurred in DC and AC microgrids are reported in [10]. One of the methods could be using additional equipment(s) to build a protection layer for renewable source integrated microgrid; however, this requires additional hardware cost [5]. Besides, the faulty equipment is required to be fixed or replaced in the case of equipment failures, which adds more cost and man work. It is noted that equipment failures are frequent events and blackouts are mostly triggered by these problems. Apparently, hardware-based applications seem not scalable, and they are infeasible due to financial problems. Thus, many new intelligent computer-aided prediction techniques have been developed to deal with fault scanning [11-13].

In recent years, many soft computing and data-driven approaches have been investigated to detect power system fault. The performance of these techniques can be measured by (i) the rate of the accuracy, (ii) detection time, (iii) cost-efficiency, and (iv) flexibility/adaptivity. Signal processing approaches, such as decision tree and random forest are extensively utilized to detect the fault in distribution systems connecting microgrids [14]. There are also some other signal processing approaches; Fourier Transform, Wavelet Transform, S-Transform, TT- Transform, Hilbert Huang Transform techniques have successfully been applied to detect faulty conditions [4], [15], [16]. Data-driven approaches have also received attention by power system researchers in identifying faults [17], [18]. The Machine learning technique makes computer/system learn various patterns based on the given input and hence it is found application in power system fault detection [19]. For instance, the machine learning algorithms such as the k-Nearest Neighbors (kNN) algorithm, support vector machine, etc., are adopted in identifying grid fault [20].

It is noted that the inclusion of renewable sources causes frequent voltage and frequency fluctuations [20-21] which may cause power system vulnerable making fault detection methods unreliable. Although there are no fault conditions, the protection devices or fault detection algorithms may receive erroneous signals by those random voltage and frequency fluctuations [23]. Therefore, the fault detection algorithms should also be scalable and adaptive where renewables penetration increases. It is also observed that many algorithms are designed for the static power system [24] and in the case of the modified network by adding/dropping a line or adding new buses, most of the proposed algorithms might fail, so the systems must be adaptive.

Considering the aforementioned issues, a real-time fault detection technique has been developed in this study using optimization enabled weighted ensemble-based Machine Learning Algorithm. The proposed method is simple to implement as it uses only voltage signals obtained from PMUs.

There are different solutions that use only one Machine Learning (ML) algorithm to reach the time goal [24], or complex ones that merge multi-machine learning algorithms [25-26] to obtain higher accuracy result. The proposed method, in this study, is a kind of ensembling type blending optimization algorithm where the Particle Swarm Optimization (PSO) finds the optimum weights to eliminate the forecast errors coming from each ML algorithms. Another contribution of the proposed algorithm is that it provides flexible/adaptive behaviors, so that proposed algorithm even works well with power system structure or given data set changes while each individual ML algorithms may fail in these changes. The most of study has been used overfitting technique that comes from memorization of results instead of learning that cause to reach unrealisticly 100% accuracy rate. This is due to ignoring cross validation techniques that is taken into consideration in this work to be able to represent results in real world power system environment.

Section II discusses machine learning algorithms used in fault detection and their mathematical algorithmic background is represented. The ensemble and boosting algorithms are discussed and the proposed algorithm architecture is presented.

Effectiveness and success of the proposed algorithm by comparing with individual ML algorithms and Ensemble methods are represented in Section III. Finally, in Section IV, findings of the proposed work are summarized in the results section.

## 2.2 Iterative Machine Learning Algorithms in Fault Detection

Unlike traditional computer algorithms, ML applications are not computer programming. ML creates a special algorithm for given data/situation which exactly fits the system. A faulty condition could be happening on transmission and distribution lines, for example; a bird/snake can touch the cable or a tree can fall down on the transmission/distribution line causing a short circuit. The ML technique could be an effective way to detect these faults. Some of the ML techniques which can be used in signal processing and fault detection are presented below.

### 2.2.1 k-Nearest Neighbor (kNN)

The kNN is easy to apply, a simple and effective algorithm for binary or multi-classification problems. It considers the set of observation data and checks the neighbors of the new incoming items according to the value of $k$, number of neighbors. In most of the ML applications, $k$ value is chosen as a default 3 or 5, and also Minkowski distance (1) has been chosen as a distance length metric. The process of the algorithm is simple; according to the number of neighbors and the coordinates of the new data $(x_1, y_1)$,…, $(x_n,y_n)$, k nearest neighbors are determined by measuring the distance, e.g., Euclidean distance, Minkowski distance or Manhattan distance, for all of the newcomers, and finally the system decides the clusters of the new nodes with the closest distance [28].

$$d(x,y) = \left( \sum_{i=1}^{n} |(x_i - y_i)^p| \right)^{1/p} \quad \textbf{(1)}$$

## 2.2.2 Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) can be used to reduce the size, improve computational efficiency, and reduce overfitting in non-digitized models. The LDA is very similar to Principal Component Analysis (PCA). The PCA, trying to find the orthogonal component axis of the maximum variance in a data set; The LDA is to find the feature subspace that optimizes class separability. The LDA and the PCA are linear transformation techniques that can be used to reduce the number of dimensions in a dataset [29].

The present state of the data set is used to make the data more easily separable when it is not very convenient to separate the components. To achieve this, it also takes advantage of the covariance matrix. In fact, it is not literally a classification algorithm. It can be used as a pretreatment before the classification process when there is not enough difference to distinguish the classes following the feature extraction. To distinguish between classes, LDA examines the distribution of classes and uses the difference between the average values.

## 2.2.3 Logistic Regression (LR)

Logistic regression is a statistical method used to analyze a data set with one or more independent variables that determine a result. The result is measured by a binary variable (there are only two possible results). In logistic regression, the dependent variables must only be binary. In another word, final results are only 1 (TRUE, success, etc.) or 0 (FALSE, error, etc.) as encoded data.

The purpose of logistic regression is to find the most appropriate (yet biologically plausible) model to define the relationship between a number of independent (predictive or explanatory) variables related to the two-way characteristic (dependent variable = response or outcome variable). Logistic regression produces the coefficients of a formula to estimate the probability.

## 2.2.4 Naïve Bayes (NB)

The algorithm has been widely studied since 1950, based on Bayes Theorem and it uses the idea of the simple probabilistic classifier. In statistic and computer science, the NB is represented as conditional probability [25]. Terminologically, Bayesian probability is given in (2).

$$p(C_k|x) = \frac{p(C_k)\, p(x|C_k)}{p(x)} \tag{2a}$$

$$posterior = \frac{prior\ x\ likelihood}{evidence} \tag{2b}$$

Where,
$p(C_k|x)$ is the posterior probability of class
$p(C_k)$ is the prior probability of class
$p(x|C_k)$ is the likelihood
P(x) is evidence

### 2.2.5 Decision Tree (DT)

In Machine Learning applications, the Decision Tree (DT) is one of the most preferred algorithms due to its simple background. The DT gives all possible outcomes and if you have enough data for the next future prediction, it can decide precisely. The DT uses a math-based background that relies on Shannon Information Theory and entropy calculation [30]. The biggest entropy value is start of branches and the whole tree follows it with the same scheme as shown below:

$$E(s) = \sum_{i=1}^{c} -p_i \log 2\,(p_i) \tag{3}$$

where, E(s) is entropy, and it represents the power and dominancy of the feature frequency. Therefore, the branch starts from the feature which has the biggest E(s) value; $p_i$ is the probability.

### 2.2.6 Finding Best Decision Variable for Each ML Algorithms

Machine learning algorithms take several parameters that can affect the accuracy rate. Before bagging several ML techniques as an ensemble algorithm, to understand the optimum parameters of each algorithm the proposed method checks the sub-set of parameters or decision variables given in Table 2.1 (unshaded part). The ML algorithms were run iteratively with different parameters and the best combination is obtained for a given algorithm as demonstrated in Figure 2.1 This code cycle is processed just for one time until the power system or data structure changes. The main idea of this loop is to get the best decision variables to the PSO optimization based Ensemble algorithm shown in Section IV. Briefly, the PSO-based Ensemble algorithm combines different iterative ML techniques with the best parameters and weights. Since chosen ML techniques for the bagging process will always use the same local parameters during the decision time span, optimum parameters are obtained with this looping process individually before continuing with the optimization part. Since power system structure is not always kept constant and requires some changes over time, algorithms need to be able to work with updated network structures. This ability is known as self-healing property of algorithms that provides to detect and update decision variables and weights when power system structures are changed.

For instance, when kNN is working on, the algorithm checks *k* neighbor value to find the best decision variable that provides the maximum accuracy rate for a given dataset. Although these parameters can be altered with topological changes, this proposed approach finds new values before applying the optimization algorithm. Thus, the system becomes robust and adaptive with various datasets.

## 2.3 Ensemble Algorithms

Each ML algorithms has got different tuning parameters, for example in kNN algorithm k value must be calculated to reach best recall result before the combination. In proposed method, before the combination of intelligent algorithms, the parameters tuned for each algorithm. To achieve that best parameter calculation, a code cycle has

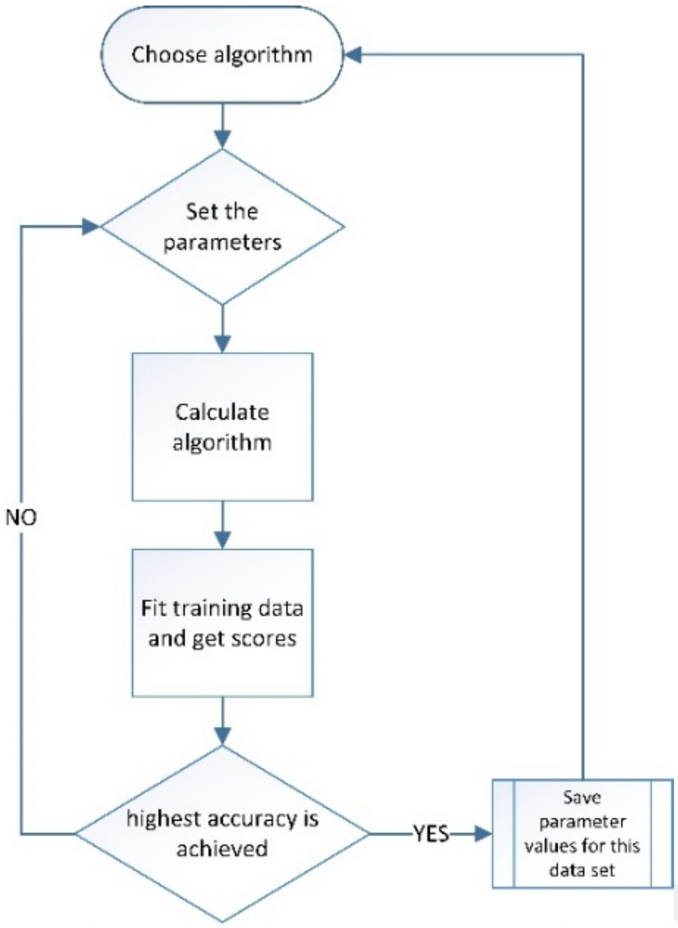



**Figure 2.1 Best decision value calculation for ML algorithms**

been developed first, for each algorithm. But this calculation has been done once. After that, combination process starts.

### 2.3.1 Baggin Method

Majority/Hard voting is a simple case of voting methods with a voting algorithm which is given in (4).

$$\forall C \in Classifier$$

$$\varphi = mode\{C_1(x), C_2(x), C_3(x), \ldots\ldots, C_N(x)\} \quad \textbf{(4a)}$$

$$\begin{pmatrix} C_1(x) \\ C_2(x) \\ \vdots \\ C_N(x) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} \quad \textbf{(4b)}$$

$$\varphi = mode\{0,..,1\} = 0\ or\ 1 \quad (4c)$$

Where $\varphi$ is the final decision of total results and it uses Python's mode command. This is a piece of code cycle; $C_1, C_2, \ldots, C_N$ are classifiers decision results, either 0 or 1.
In this scheme, each ML algorithms come up with the decision on the given test case, separately. The final decision will be given with the agreement of the majority.

***Soft voting*** gives the average probability of the decisions rather than counting the votes on positive or negative decisions coming from the ML algorithms. For example, when three algorithms give the decisions with (0.60, 0.60, 0.15), then hard voting will decide negative since there are 2 positives and one negative. However, the soft voting will decide it as negative due to the average of the probability which is 0.45. In this example, algorithms have the same significance, however, the weights can be different if the contribution of the algorithms is not the same.

### 2.3.2 Running schemes

AdaBoost (AB) iteratively repeat the weak learner algorithm with given instances. In each iteration, misclassified data items are re-weighted according to the information gained from the previous step. With this feedback mechanism, the AB runs a classifier, changes the weights, runs another classifier, repeat until most of the items classified proper. Thus, there is no parallel calculation, each step must follow the previous ones like a chain.
The Bagging and boosting processes are represented in Figure 2.2 Not only the bagging process precedes boosting type but also it is reinforced with PSO that results in faster processing time when compared with any other ML algorithm or boosting method.

**Table 2.1 Parameters to calculate the decision variables for selected ML and Ensemble algorithms**

| ML and Ensemble Algorithms | Decision Variables | Interval |
|---|---|---|
| kNN | number of neighbors | [3 to 21, incremented by 1] |
| LDA | tolerance | 0.0001 |
| LR | - | - |
| DT | - | - |
| NB | - | - |
| **AB** | Estimator | [1 to 100, incremented by 5] SAMME,SAMME.R |
| **GB** | number of trees | [11 to 71, incremented by 1] |
| | seed | [1 to 11, incremented by 1] |

Gradient Boosting (GB) also combines multiple ML algorithms based on weights. It collects weak learners and make a new stronger learner and works as a team. A data set is applied to this algorithm in order to get the best result out of it. The GB has got 2 types of algorithms; one of them is SAMME.R and the other one is SAMME. Each algorithm's decision variable values are listed in Table 2.1 (shaded part)

## 2.4 Proposed PSO Based Weighted Ensemble Approach

The workflow for the proposed algorithm is given in Figure 2.3 which combines five different ML algorithms explained in Sect. 2.3, by using a novel weighted ensemble approach blended with Particle Swarm Optimization (PSO) technique to detect the faults on the power system accurately. The proposed approach is explained with the following steps:

Collect the data from the sensors and all other parts of the systems, e.g. SCADA, to train the system periodically.

For each ML algorithms, the best decision variables of the techniques are calculated with Brute-Force approach as given in Sect. 2.4.

The weights of ML algorithms for the bagging process, which is based on soft-voting technique, are calculated by Particle Swarm Optimization module. Thus, PSO will give the best set of weights for the soft-voting approach. In that part, bagging is applied with the PSO calculated optimized weights until the next training time window.

1- The ensembling process uses a minimum 2 and maximum 5 ML algorithms according to PSO results. In that part also cross validation was applied to dataset to evaluate the predictive performance of the model results in training dataset step. In doing that one of the well-known method, K-Fold Cross Validation, is applied as cv=5. That means, 80% of data was chosen for training part and 20% of data was for testing purposes.
2- Self-healing algorithm has been developed which is adaptive against the structural changes and the algorithm collects the data and control the power system in case of any faults. Based on each algorithm weights obtained in step-4, power system control action is triggered, in this case, in PSCAD simulator.
3- Re-train the whole system when the system hits the structural or behavioral changes and re-calculate the optimum values by following the same steps to obtain best weights for each algorithm which is self-improved algorithm ability.

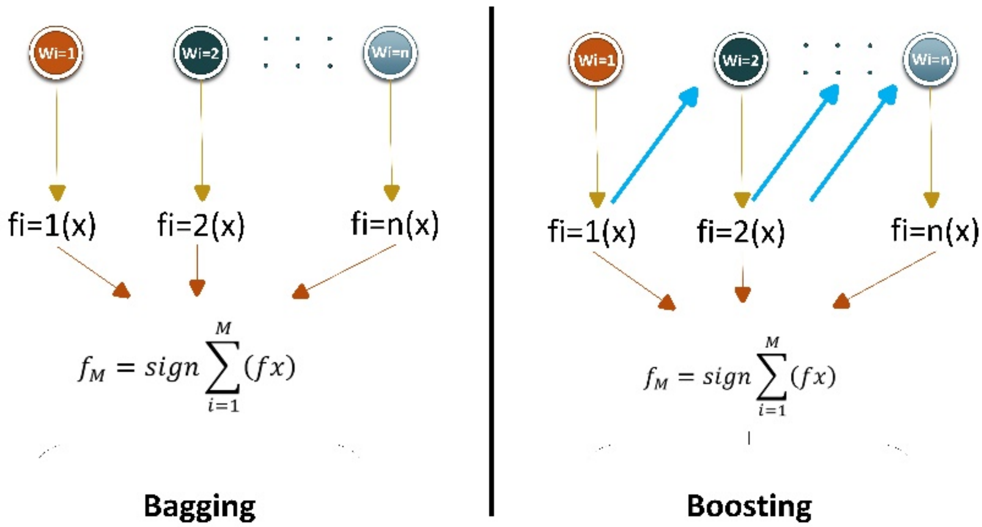



**Figure 2.2 Bagging and boosting methods**

4- If there is no structural change e.g., add/drop a new line, adding new renewable sources, step-6 would be remained.

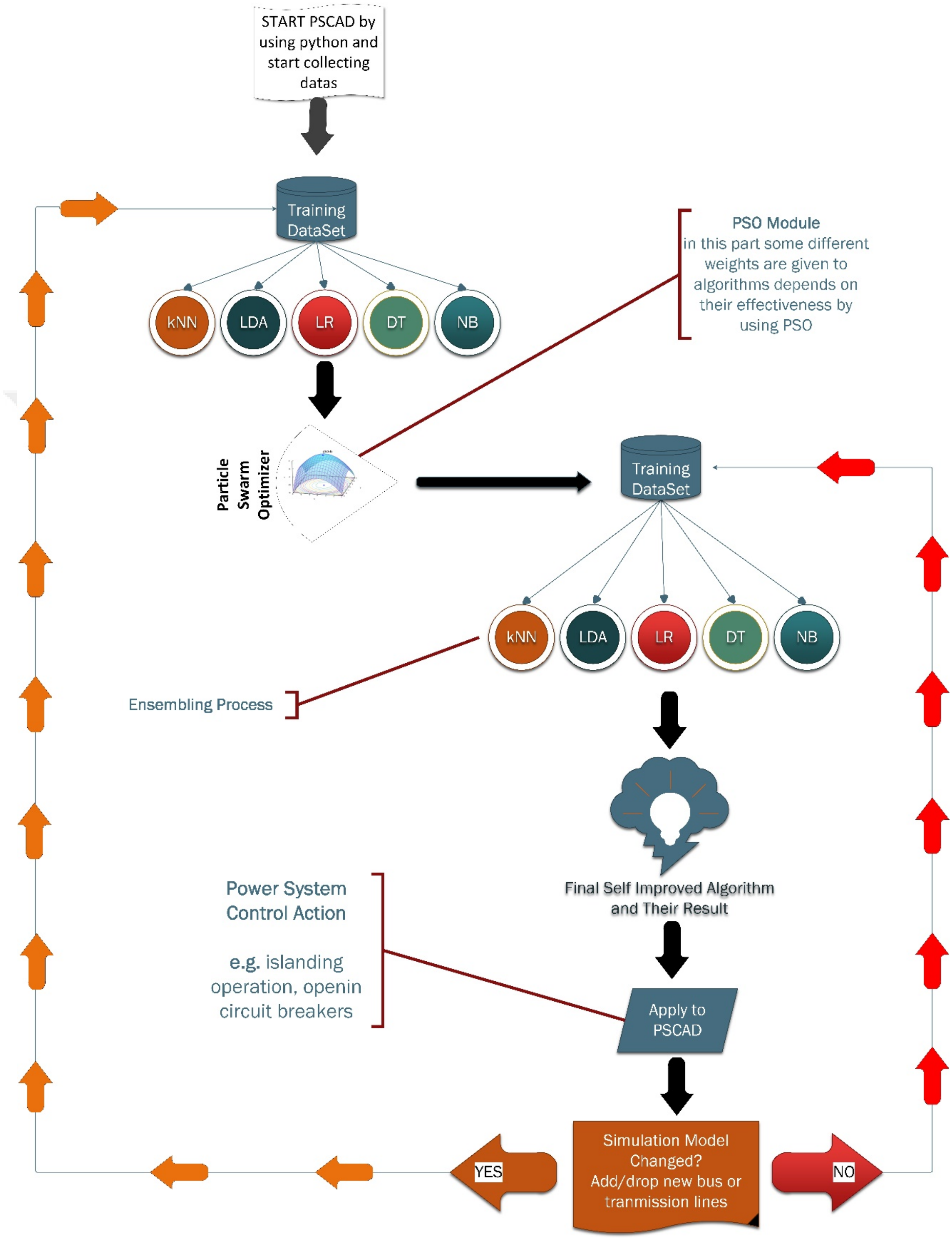



**Figure 2.3 Workflow for the PSO based weighted ensemble algorithm.**

Since each algorithm has pros/cons for different characteristics of datasets, all of them are combined together that couples the powerful sides of each. As a result, the proposed method becomes flexible and adaptive in case of any structural changes, which is very normal and frequent behavior of the power system. This method does not require to apply any signal processing techniques or any other pre-processing method like feature selection techniques, but at the same time, it can obtain very high accuracy even with raw data. Most of the methods use feature selection techniques to analyze the training data to obtain high accuracy results for the given data set. However, the same selected features, or parameters that used in the specific algorithm, can fail on another dataset, or scenario, because of its different characteristics. Thus, the proposed methods should be dynamically adapted to the collected data, which is rare and hard to optimize and apply to the real environment. It is noted that, with the proposed method, real-time processing could be possible since it has an ability to work with unprocessed newly collected data, as one of the powerful sides with the self-healing adaptive background.

In the ensemble method, the key point is that the system will give *0* weight to the algorithm if it is not needed in the selected set. Each weight represents the power rate of the algorithm for the given data set during the soft-voting process. The PSO finds the best bag and provides the weights for each classifier in the combination. The generalized approach is shown below:

$$\mathrm{M} = \{\mathrm{mla}_1, \mathrm{mla}_2, \dots \mathrm{mla}_\mathrm{N}\}, \qquad \text{ML Algorithms} \tag{5.1}$$

$$\boldsymbol{K} = \{\boldsymbol{i_0}, \boldsymbol{i_1}, \dots, \boldsymbol{i_k}\}, \qquad \boldsymbol{Class\ Labels} \tag{5b}$$

$$\boldsymbol{W} = \{\boldsymbol{w_1}, \boldsymbol{w_2}, \dots, \boldsymbol{w_N}\}, \qquad \boldsymbol{Weights} \tag{5c}$$

$$\boldsymbol{\theta}(\mathbf{M}, \mathbf{W}, \mathbf{K}) = \underset{\mathbf{i}}{\mathbf{argmax}} \sum_{\mathbf{j=1}}^{\mathbf{N}} \mathbf{w_j} * \mathbf{p}(\mathbf{mla_j} | \mathbf{i}) \tag{5d}$$

$$\boldsymbol{PSO}(\boldsymbol{mla_1}, .., \boldsymbol{mla_N}) = \{\boldsymbol{w_1}, \dots, \boldsymbol{w_N}\} \tag{5e}$$

Where, M is the set of weak machine learning algorithms represented by mla, K represents the class labels -which is ‘fault’ or ‘not fault’ in the scope of this paper-, W shows the weights of the algorithms. $\theta(M, W, K)$ is the function of the bagging process, and PSO calculates the optimal set of weights for $\theta(M, W, K)$ that obtains the highest accuracy.

The main purpose behind this idea is explained in Figure 2.4 with a mock-up example. With the proposed method, each algorithm will close the gaps of other collaborative algorithms and try to get a consensus all together if there is a fault or not. However, this consensus should also be as fast as possible because of the time limitation while working in real-time environment. By using the best-scored classifiers combination and their calculated weights, the proposed model will be able to run the ensemble algorithm in real-time for islanding detection. Please note that the weight calculation process is computed just once unless there is no modification on the power system topology. In case of any structural changes in the power system model, the proposed algorithm will detect that difference and train itself again and again until reaches the saturation.

## 2.4 Result and Discussion

Islanding or fault situation must be detected as fast and accurate as possible, and for this purpose conventionally signal processing techniques have aided to obtain great predictions with machine learning algorithms. In addition to these techniques, many feature selection methods are applied in most of the technique in the literature to deal with deficiencies and system defects. However, it focuses on the given dataset and provides excellent results for the specific state. When the technique provides high accuracy rates with feature selection, the same technique could fail on a different or dynamic data structure. When a ML algorithm improved and specialized for power system structure it cannot be used on different structures or it can fail in case of any difference in the power system structure. Thus, most of the common solutions are not applicable in the real environment. On the other hand, many improved ML techniques show very high accuracy results which means these improved models memorize the data instead of learning. Thus, it can be easily recognized when the algorithm results in 100% accuracy when there is overfitting problem.

To overcome these situations, a weighted and self-healing ensemble technique has been proposed in this study by choosing 5 most robust machine learning algorithms. The IEEE 14 bus power system model is used to show the effectiveness of the proposed weighted ensemble-based machine learning approach for fault detection. In addition to the well-known IEEE 14 bus model, the models shown in Figure 2.5, to compare the performance under entirely different characteristics of power systems in pointing out its adaptive and self-healing schemes.

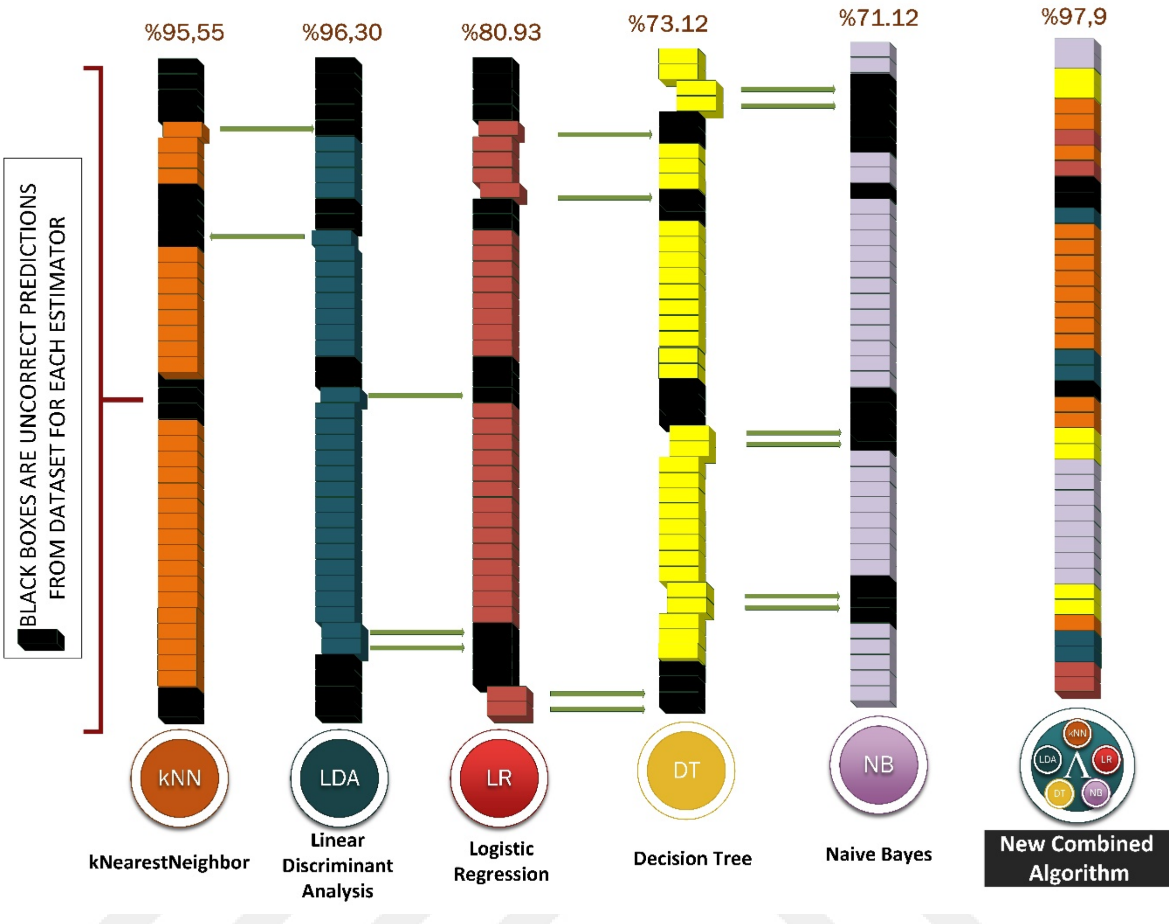



**Figure 2.4 PSO effects for each ML algorithm**

In this scenario, the IEEE 14 bus model system has been modified connecting renewable sources at bus 3, 6, 8 by providing intermittency and uncertainties of voltage and frequencies to test the proposed method's adaptivity. In both cases, PSCAD/EMTDC software is coupled with Python to solve the problems and bring solutions in a co-simulated platform to mimic real-time scenario. PSCAD is being used to simulate the power system and Python for implementing ML algorithms. Time step of the co-simulation is kept as 50µs. By running the simulation 5 sec, 32 millions of data are generated for analysis. About 80% of the gathered data is being used for training the algorithm and then some part of remaining data e.g., 20% data is used for testing of each algorithm. Cross-validation was chosen as five which means algorithms test the next 20% part of the dataset and this process continues five times until all data was used for the test. Than the results are obtained by taking average of five different accuracy result. For example kNN accuracy results are %89, %93.25, %97, %91, %92.5 for each cross-validation cycle. Average of five of the accuracy value is 92.55% as stated in Table 2.2 Overfitting problem was handled in this way and get more realistic results.

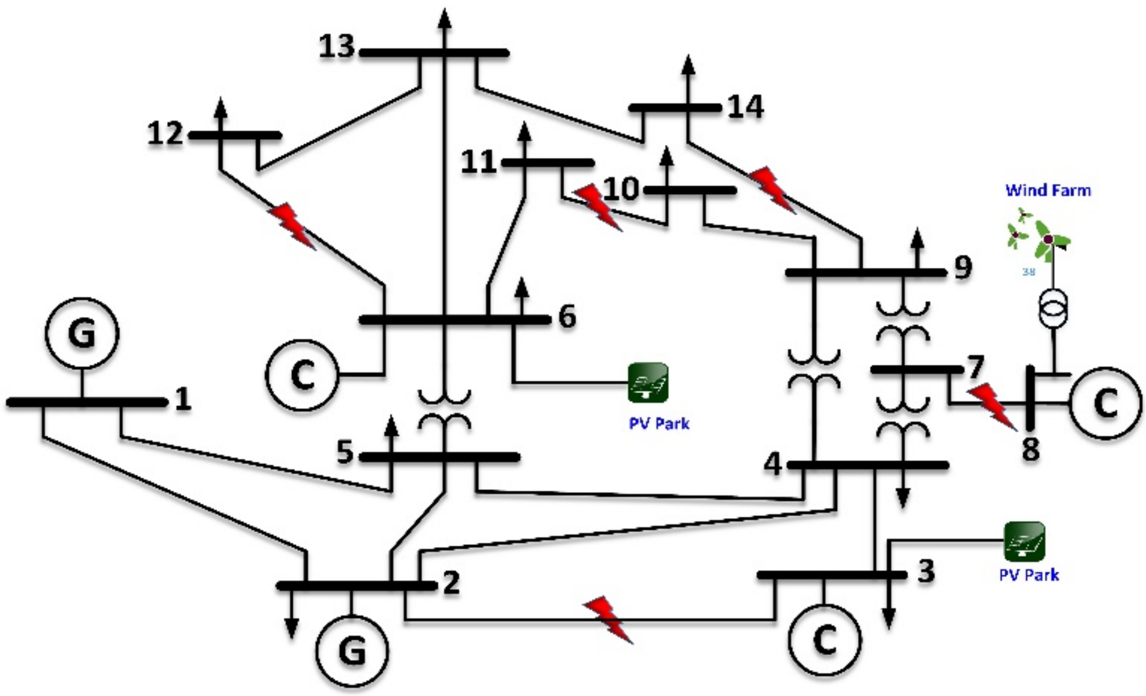



**Figure 2.5 Modified IEEE 14 Bus study system including renewable sources**

After these steps, each ML algorithms (kNN, LDA, LR, NB, DT), boosting algorithms (AB, GB), and finally proposed PSO based ensemble method are investigated and compared in terms of effectiveness, adaptivity, and process-time under two different case scenarios.

## 2.4.1 Case 1: Fault Analysis and Comparison

The most frequently occurred fault in power systems is single-line-to-ground (1LG) fault and it is one of the most difficult ones to detect, compared to severe double-line-to-ground (2LG), and three-line-to-ground (3LG) faults. In this study, the 1LG fault has been selected and applied randomly for 0.15 s in lines (7-8, 10-11, 9-14, 2-3, 6-12) at 5 different time instants (1.6, 2.3, 2.8, 3.8, 4.4), respectively. Different fault locations are chosen to observe the effect of the proposed algorithm, as voltage deeps vary randomly which is shown in Figure 2.6 Also each busses frequency and phase values have been used for prediction process. Figure 2.6 shows just an example of them.

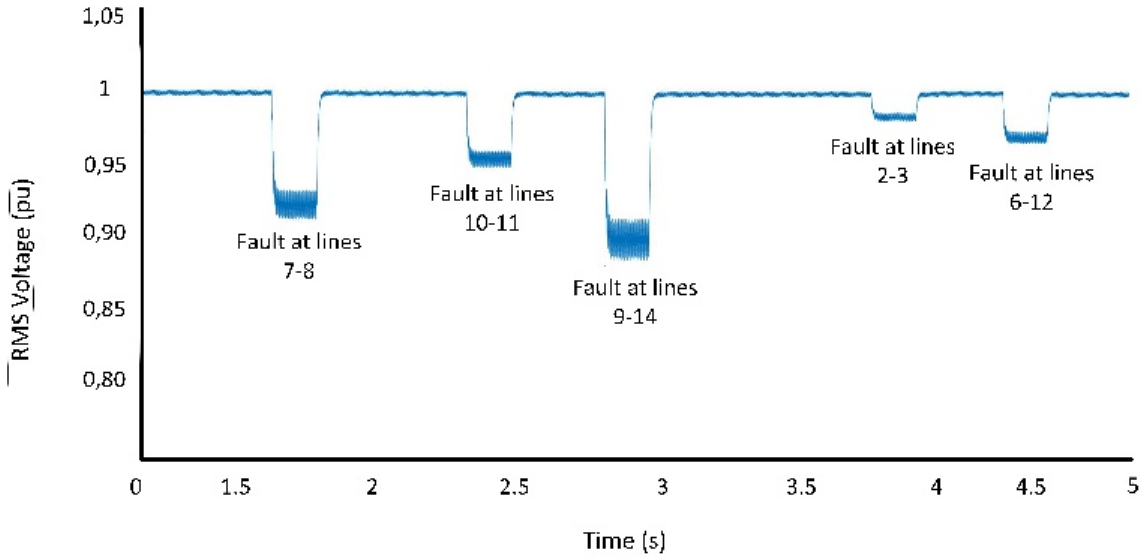



**Figure 2.6 Voltage response at bus 9 (1LG fault)**

Table 2.2 shows how PSO-based weighted ensemble method, which is a combination of multiple (at least 2, up to 5) ML algorithms, can work on-fly effectively compared to well-known boosting algorithms for on IEEE 14 bus standard model.

Table 2.2 also shows a comparison of the proposed PSO-based ensemble method with other boosting algorithms (GB and AB), in terms of time and accuracy viewpoints using IEEE 14 bus model. Accuracy results show the proposed algorithm is around 5% more accurate, and too much faster then individual and boosting algorithms.

In process time-wise, PSO-based Ensemble algorithm works faster than other approaches because of its bagging scheme and pre-computed background in which as an evolutionary algorithm PSO particles communicate with each other and they use their own previous best solution. Thus, it can reach final results very fast way. Each ML algorithms are strengthened by PSO, so PSO trains the algorithms to find the best weights, however, this process merely happens one time unless the power system structures are kept constants.

**Table 2.2 Decision parameters and accuracies for each classifier on Classical IEEE 14 Bus model**

| Algorithm | Parameters | Algorithms Individual Process Times (s) | Accuracy Results % |
|---|---|---|---|
| **K Nearest Neighbor** | Neighbor=9 | 1.7 | **92.55** |
| **Linear Discriminant Analysis** | Tolerance=0.0001 Solver=svd | 1.49 | **92.35** |
| **Logistic Regression** | - | 2.3 | **80.93** |
| **Naïve Bayes** | - | 0.6 | **73.12** |
| **Decision Tree** | criterion=gini splitter= best | 2.0 | **71.12** |
| **AdaBoost** | - | 0.57 | **93.13** |
| **Gradient Boosting** | - | 0.29 | **92.85** |
| **PSO-Ensemble** | PSO Weights of Combination of 5 ML Algorithm: kNN 0.51; LDA 0.49; LR 0.24; DT 0.14; NB 0.82 | **0.021** | **97.93** |

On the other hand, the same collected dataset used for comparison with boosting algorithms and the proposed method gives better results both in terms of both process time and accuracy. Bagging application is faster than any other boosting algorithms (AB and GB) due to the parallel processing structure. In the proposed method, different powerful ML algorithms close their gaps as working together. That is the reason why the proposed method gives better results.

### 2.4.2 Case 2: Power System Structure Adaptivity Test

As mentioned earlier, the structural change is reflected in the simulation by adding 3 renewable sources at bus 3, 6, 8 and this system is named as modified IEEE 14 Bus model to test the proposed method in adaptivity and self-healing. The 1LG fault is considered in this case too and captured data-set of the voltage signal from Bus 9 is considered for the analysis.

Each individual well-known algorithm and boosting algorithms are tested and compared with the proposed method using modified IEEE 14 bus model. In this case, accuracy results can be seen in Table 2.3 which shows proposed method adaptivity is very high when compared with other methods. Table 2.3 also shows that the proposed method provides almost 5% better accuracy performance compared to boosting algorithms and individual ML algorithms accuracy results in modified IEEE 14 bus model.

**Table 2.3 Decision parameters and accuracies for each classifier on Modified IEEE 14 Bus model (with PV)**

| Algorithm | Parameters | Algorithms Individual Process Times (s) | Accuracy Results % |
|---|---|---|---|
| **K Nearest Neighbor** | Neighbor=9 | 1.94 | **92.61** |
| **Linear Discriminant Analysis** | Tolerance=0.0001<br>Solver=svd | 1.63 | **87.38** |
| **Logistic Regression** | - | 2.45 | **76.3** |
| **Naïve Bayes** | - | 0.83 | **84.5** |
| **Decision Tree** | criterion=gini<br>splitter=best | 2.13 | **82.38** |
| **AdaBoost** | - | 0.57 | **81.23** |
| **Gradient Boosting** | - | 0.29 | **91.35** |
| **PSO-Ensemble** | PSO Weights of Combination of 5 ML Algorithm<br>kNN 0.69<br>LDA 0.40<br>LR 0.39<br>DT 0.23<br>NB 0.85 | **0.036** | **96.68** |

Different algorithms result clearly show (in Table 2.3) that some ML algorithms can provide high accuracy(kNN and GB accuracies are more than 90% as in Table 2.3) on the other hand some ML algorithms cannot reach high accuracy levels on different structure of power systems. That is the main reason to develop the bagging-based ensemble algorithms to strength deficiencies and weaknesses in ML algorithm in different scenarios and/or structures.

It is noted that Gradient boosting works more accurate than Adaboost on modified IEEE model. The main reason for that is by adding renewable sources the gathered data had more noise because of voltage/frequency fluctuations, and therefore, the Adaboost algorithm can be easily defeated by noise when compared with Gradient Boosting. With respect to accuracy and process time, the proposed PSO based Ensemble method shows very high results over boosting algorithms. It is shown that the proposed method has an adaptive characteristic, and it can work way better than any platform without changing pre-computational techniques so that it provides a specific solution for a given data set. Since the PSO weights are calculated dynamically, whenever needed, the PSO-based Ensemble method can be easily adapted in different schemes and power topologies, so that it can train and predict data at the same time.

The performance of the PSO-based Ensemble method is significantly not affected in uncertain cases in voltage and frequencies such as adding renewable sources. That means the addition of renewable sources affects the individual machine learning algorithms' performance, however, the proposed method progress well even in this situation.

## 2.4 Test Results

The PSO-based Ensemble method is proposed in this study to detect faults in the power system. The proposed algorithm is tested on IEEE 14 bus system and modified model by adding newly commissioned renewable sources as a means to present the case of structural change of power system. In the proposed method, there are no signal processing or feature selection techniques needed, and just raw dataset has been used for the prediction. Thus, the proposed method is not just specialized for the dataset, it is also adaptive and flexible for any kind of structure-based dataset. It is found that the proposed method provides much accurate results than individual machine learning algorithms using both IEEE models (original and modified). The proposed method's accuracy rates are acquired as 97.93% for IEEE classical model and 96.68% for the modified model. While IEEE 1547 standards allow 2 seconds delay to detect the possible faults, proposed method got better results up to 0.021s for IEEE 14 bus and 0.036s for the modified IEEE 14 bus model. This means, the computational time is almost 1% of IEEE standards which is small enough for power system study. Thus, the proposed method will provide higher and faster results than the most popular machine learning algorithms and also provides adaptivity for any structural changes.

# Chapter 3

# Initialization Methods Applied on Deep Learning Algorithms on PV Power Plants

ML applications are not computer programming, like traditional computer algorithms. ML creates a special algorithm for a given data/situation which exactly fits the system.

In the power system, researchers are faced with different problems that are so close to needed ML applications. Power transmission and distribution problems depend on too many variable states. So, there are some solutions applied on power system. In this chapter we will check the algorithms and their methods refer to combining weak learning algorithms and transform them into a strong learner with additional processes. Weak learners can work sequentially, and each predictor tries to fix the previous results via the boosting method. The other approach is to combine the results of weak learners with the bagging method. As a result, the proposed method becomes flexible and adaptive in case of any structural changes, which is very normal and frequent behavior of the power system. This method does not require to apply any signal processing techniques or any other pre-processing method like feature selection techniques, but at the same time, it can obtain very high accuracy even with raw data. Most of the methods use feature selection techniques to analyze the training data to obtain high accuracy results for the given data set.

## 3.1 Introduction

With the increase in population and technological developments since the last century, the world has more demand to deliver. However, as a result of excessive and irregular use of fossil fuels, it causes a series of environmental problems such as climate change, global warming, air pollution and acid rain. Capturing solar energy through photovoltaics panels to generate electricity has been in demand in the field of renewable energy since recent years, and the most important reason for this increasing demand is the rapid growth perspective and high investment levels after decreasing fossil resource reserves [31]–[34].

The energy generation has cons with respect to power generation consumed as it produced. So, it requires planning in advance for producers and consumers as well. That should be optimized to keep stable operation in power systems. One of the important factor to help these stable operation is to have an accurate generation forecasting, that is not an easy task. There are many environmental factors that affect the energy production of solar panels (wind intensity, pressure, humidity, etc.) which they are depended on whether situations and directly unpredictable. So, it is very important for transmission and distribution aggregators to obtain energy production information and accurate forecasting models. So, there are many different types of systems, especially tens of thousands machine learning implementations, have been developed to forecast more accurate energy production for intra-day and day-ahead energy market.

Artificial Intelligence (AI) and Deep Learning (DL), which have been popular recently, appears as applications in many different places in daily life. DL is a field of study that covers Artificial Neural Networks (ANN) with one or more hidden layers. In summary, the computer uses at least one ANN from the input data at hand and obtains new output data with many different algorithms. AI is the ability of a machine to exhibit human-like skills such as reasoning, learning, planning and creativity. The important point here is that non-intelligent entities can solve problems, and the automation method is uncertain. DL consists of three layers, the input layer, the hidden layers, and the output layer. While the input data is transferred to the system in the input layer, it performs different type of mathematical calculations on the inputs in the hidden layer. One of the difficulties in creating ANN is to decide the number of hidden layers as well as the number of neurons for each layer. In the output layer, shows the output data obtained with the help of the previous hidden layer. The connections in between are called artificial neural networks. The connection of ANN with each cell are called weights that take coefficients value between 0 and 1[35]. In order to obtain accurate model and satisfying results the weights of input-hidden-output layers' weights must be calculated accurately.

Most of the problems encountered in ANN are gradient disappearance/burst problems encountered during the training phase. Some of the basis weight assignment techniques offer a solution to this problem, but do not provide sufficient effectiveness in general [36], [37]. Studies have been made to improve performance by analyzing weight initiation techniques using methods such as gradient descent. The algorithms are mostly created with traditional weight initialization techniques. However, models created with traditional weight initialization techniques do not have high performance in learning capacity [38]. The combination of traditional weight initiation techniques and their features play a major role in the accuracy of the model. Comparison of different weight initialization techniques used in ANN provides clearer information about the performance of the model. Since random weight initiation techniques affect what the model will do in the training phase, different techniques should be used in the training phase [39], [40].

In the literature, there are four most used weight assignment methods are Xavier, LeCun, He, Random. Random Initialization for neural networks aims to break symmetry by randomly initialized weights. These methods initialize the weight values to be close to zero and as a result, the symmetrical distribution is broken. In this way, each neuron does not repeat the same calculation anymore, and in addition, all bias values can safely start with the same value. These randomly initialized weights are optimized using different approaches with gradient descent techniques. The new weight coefficients obtained after the optimization process reduce the margin of error in the estimation algorithm and provide a better result [41]. However, the most significant disadvantage of this method is that if the weight value is too large or small, the margin of error increases [42], [43]. For He method Recent deep Convolutional Neural Network (CNNs) are often initialized by random weights assigned from Gaussian distributions [41] . Difficulties were observed in converging very deep models created with the constant standard deviation [44]. In the following, Kaiming He et al. derived a theoretically more capable initialization method by taking ReLU/PReLU into account. This technique is focused on composing the weight assignment more effective as a result of a variance- based formulation process with some assumptions formed. At the end of this initialization technique, the gradient loss that will occur in forward and backward propagation operations is prevented. Xavier Initialization the method also called Glorot Initialization, was first introduced when Xavier Glorot and Yoshua Bengio published their landmark paper "Understanding the Challenge of Training Deep Feedforward Neural Networks"[45]. The assignment system using method functions that approach symmetrically and symmetrically around a certain value that was used before has changed with this method. The purpose of Xavier Initialization is to keep variance and gradient popping or vanishing to a minimum. It provides this process by initializing the weights so that the variance of the activations is the same in each layer [46]. The initializing values of the weights may have a considerable effect during the training. Weights have to be chosen randomly. However, the activation functions are principally activated in their linear regions. LeCun Yann et al. argue that learning weights are very small or very large then the activation function performs slow learning with the saturation of results in small gradients. Reaching this state requires the normalization of the learning set, the selection of the activation function, and the coordination of the weight initialization method. It is stipulated that the distribution of the outputs of each node has a standard distribution of approximately 1 [47]. Normalizing the training set in this input layer should be achieved. Therefore, As Boulila Wadii indicate that LeCun initialization solves the increasing variance with the number of inputs and sets constant variance [48], [49]. The biggest disadvantage of the Lecun Initialization method is that it is created only for activation functions that can be differentiated at $z = 0$, and it is produced for use with the tanh activation function. As mentioned above, there are multiple weight initialization methods in deep learning algorithms.

The proposed method, the Heuristic Vectorized Method, aims to be used in the PV forecasting on data sets with a few and many variables. Heuristic Vectorized Learning method offers flexibility in increasing the performance of the algorithm in artificial neural networks. The contribution of the Heuristic Vectorized Learning method is as follows. It has a faster and more effective way of solving problems than basic neural network algorithms by examining the relationships between minimum information and data. This system basically performs weight initialization by examining rows and columns where other variables remain constant while one variable changes, calculating the effect on the desired output. This new method will then be compared with the aforementioned Xavier, He, LeCun and Random initialization methods and its benefits in PV estimation will be suggested.

## 3.2 Methodology

The process of generating power using solar photovoltaic (PV) includes converting energy from the sun into electricity using solar cells. As with all the energy conversions, the efficiency is not perfect and requires applications to track and optimize it. The efficiency of the conversion is dependent on many environmental factors. In general, the nine most important factors will be considered are humidity, weather temperature, cell temperature, external cell temperature, wind speed, wind direction, and atmospheric pressure. Although, these factors have been used many applications in literature, cloud motions and thickness values are neglected to be obtaining correct forecasting.

In PV based power systems, there is a massive impact arose from the cost, due to lack of accurate forecast of PV generation. Besides to the cost, the frequency balance in the system is also corrupted and the power system management becomes an issue [50]–[54]. The intermittent generation and uncertain behaviors of renewable energy sources negatively affect power system operations such as power quality, stability, and reliability [55]. To come up with a solution for these problems, there are several approaches proposed to forecast the power generation. These approaches contain methods which can be categorized into four types as persistence, statistical, machine-learning, and hybrid method based on the use of historical data of PV power output and related environmental variables (sensor values) [50]–[56].

The simplest stochastic learning technique is the persistence forecast which is based on current or recent PV power plant or radiometer output and extrapolated to account for changing sun angles. Persistence forecast accuracy highly relies on forecast duration as cloudiness varies between states [57]. The statistical approach is based on previously recorded measured time data series and mostly seen in short horizons. They are simpler than physical models due to less input data requirement and less computation efforts for the power forecasting [58].

In recent years, various machine learning approaches including deep learning algorithms such as ANN (Artificial Neural Network), CNN (Convolutional Neural Network), RNN (Recurrent Neural Network), and LSTM (Long Short Term Memory) are also becoming quite famous in PV Power forecasting [59]–[61]. However, the algorithms developed for PV power forecasting provide accuracy rates generally between 90%-95% around the world because of the number of environmental and conditional changes that affect the power generation process such as temperature, humidity, air pressure, panel shadowing, and wind etc. Moreover, machine learning includes the complexity of computation along with the desirable accuracy rates.

Majority of the parameters that changes in a day are not far from predictability, but because of the stochastic structure of cloud motions, power generation process evolves to unforecastable scheme. As the clouds move between panel and the sun, the energy generation ratio drops to considerably an insufficient level. A study shows that multiple deep learning models with the integration of both static sky image units and dynamic sky image flow are explicitly explored for short-term prediction of cloud motion. Even if the developed model successfully detects and segments the clouds, it is not capable of forecasting ahead of one hour. In addition, the examined systems do not have the data abundance of the newly developed system, that is, environmental data such as temperature and humidity were not used except for the camera image. For this reason, the newly developed model has longer forward forecasting and better accuracy than previously designed models [62]. For the short-term PV Generation forecasting, it is obvious that without investigation of cloud motion, this study reveals that following the clouds in very short 15-minute forward forecasts increased the accuracy of the system by 2%. Moreover, when this result is compared with the newly designed model, the newly designed model has increased its accuracy by approximately 4% with cloud motion and thickness data. However, this increase is for a forward- hour forecast, not for a very short term of 15 minutes. [14].

Deep learning-based cloud detection and segmentation studies, in which sky images were collected with the fisheye camera used in previous years for solar energy forecasting, resulted in a commendable accuracy with 15-minute and less forward forecasting [62]–[65]

There are forward-looking PV forecasting studies in different methods than the proposed model. Deep learning algorithms Back Propagation Neural Network (BPNN) and different cloud segmentation algorithms k-means clustering have been used [65]–[67]. The main differences from the proposes system are that the cloud thickness is not included in the cloud movement forecasting, as the forecast time increases, the accuracy values decrease rapidly. The proposed deep learning model gives results with very high accuracy in the forecasting ahead of one hour by integrating the cloud thickness into forecasting.

Considering the aforementioned issues aspects, this study was developed by incorporating cloud thickness as well as cloud motion into the estimation of energy production in the deep learning model. In the forecasting process, algorithms were developed for cloud thickness, cloud movement, and sun position, and a system was created to feed the deep learning algorithm. And data collection hardware application for learning the deep learning algorithm was also carried out. The main contributions are summarized as follows:

- Cloud thickness-motion forecast integrated in PV forecasting by using multiple convolutional neural networks mainly named as the adaptive simple linear iterative clustering (A-SLIC) algorithm has been proposed.
- Deep learning-based Cloud Detection has been used. Cloud motion forecasting handled by using A Gated Recurrent Unit (GRU) that is one of the types of Recurrent Neural Network.
- Creating an integrated cloud motion and thickness deep learning model by combining cloud motion and thickness estimation results with environmental factors.
- Not only the collected data were utilized in simulation environment, but also the hardware implementation was carried out in a real university environment.
- ANN consists of more than one layer and there are weights between both layers. These weights are used to calculate the data in the next layer, which is a linear transformation of the values in the previous layers, is used to calculate these weights.
- The heuristic vectorization method is defined as a problem-solving technique that examines and uses minimal information, past results, or relationships between data to produce an applicable and practical solution to a specific problem in a short time.

According to these contributions, in this part we will check different types of applied weight initializations on DL layers, 4 of the most used weight assignment methods are Xavier, LeCun, He, Random. In this section, the weight initialization methods and finally

Heuristic Vectorized Learning Method (HVLM) will be mentioned by next chapter going down, with its mathematical fundamentals and results.

### 3.2.1 Random Initialization

Random initialization weights are assigned close to 0 and tests are started after these assignments. The importance of random start weight assignment is explained by a new approach to deep ANN pruning called the Lottery Ticket Hypothesis (LTH) [20]. It suggests that there are random subnets ("winning lottery tickets") that can achieve similar or better performance than the entire model with the same number of training steps according to the LTH approach. After this step, the remaining parameters are reset to their random initial state, and the subnets can then be retrained to a similar optimum compared to the original larger network. As can be understood from here, this subnet is not related to any organization that emerged during the training. This subnet is a subset of random initial weights. The LTH [21]–[23]approach suggests that the weights are highly correlated. In addition, that suggests that it may not require training to achieve high productivity. Results from the article Effect of Initial Configuration of Weights on Training and Function of Artificial Neural Networks have been successfully trained together with the Stochastic Gradient Descent (SGD) [24] method of ANNs. As a result of this training, it was determined that the initial configuration of the weights approached its close neighborhood. It has been confirmed that random initialization has great positive effects on training and model performance with these studies.

An ANN function for the supervised classification of data in deep learning is given below (Eq.1).

$$\boldsymbol{f(x, \emptyset_f) = y} \tag{1}$$

In this function.
x: input image parameter
y: the assigned class label parameter θ: model parameter

In the above equation, f is called the model architecture, and several approaches exist to create this model architecture function [25], [26]. One of these approaches is random initialization. In this method, we obtain the model weight parameters $\theta = Wn_l$, $n_l+1,1$, which is a collection of randomly initialized weight matrices. All matrices represent the connections between two successive layers. These randomly initialized weights are optimized using different approaches with gradient descent techniques. The new weight coefficients obtained after the optimization process reduce the margin of error in the estimation algorithm and provide a better result [11].

### 3.2.2 Xavier Initialization

As mentioned above, the method, also called Xavier Initiation, Glorot Initiation, was first introduced as an idea by Xavier Glorot and Yoshua Bengio [15], and with this method, landmark research in weight initiation began. Xavier initialization is an attempt to improve the initialization of neural-generated and accordingly network-heavy inputs to avoid some of the common problems that exist in machine learning. One of the most important features of Xavier initialization is that the variance of the outputs of a network layer must equal the variance of the inputs. With this feature, high accuracy is achieved with the weights initiated in machine learning. As a result of recent studies, the researchers stated that Xavier initialization facilitates better results of accuracy by choosing the neuron activation functions in a reasonable range in machine learning technologies, by weight-balanced initialization. In addition to these, it is suitable for use in different areas by working with more than one activation function.

Activation functions with which it operates:

- Linear Activation Function
- Tanh Activation Function
- Logistic Activation Function - SoftMax Activation Function

Xavier Initialization aims to initialize weights such that the variance of activations is the same at each layer. With this method, constant variance values help prevent the gradient from bursting or disappearing.

To help derive the initialization values, the following simplifying assumptions should be examined in turn.

- Weights and inputs must be zero-centered.
- Weights and inputs are independent and must be initialized in the same way.
- It is necessary to initialize the biases to zero.
- It is necessary to use the tanh () activation function, which is approximately linear with small inputs.

which is approximately linear with small inputs.

The full derivation gives the following initialization rule for all weights.

$$\boldsymbol{Var}\left(\boldsymbol{a}_i^{[\ell-1]}\right) = \boldsymbol{Var}\left(\boldsymbol{a}_i^{[\ell]}\right) \tag{2}$$

$$= \boldsymbol{Var}\left(\boldsymbol{z}_i^{[\ell]}\right) \tag{3}$$

The first equation linearity of tanh is around zero which means that tanh(z)≈z. After that step,

$$= \mathbf{Var}\left(\sum_{j=1}^{n^{[\ell-1]}} \boldsymbol{w}_{ij}^{[\ell]} \boldsymbol{a}_j^{[\ell-1]}\right) \tag{4}$$

$$= \sum_{j=1}^{n^{[\ell-1]}} Var\left(w_{ij}^{[\ell]} a_j^{[\ell-1]}\right) \tag{5}$$

In the second step of equation, variance of independent sum equation is calculated as a Var(X+Y) =Var(X)+Var(Y).

$$= \sum_{j=1}^{n^{[\ell-1]}} E\left[w_{ij}^{[\ell]}\right]^2 \mathrm{Var}\ \left(a_j^{[\ell-1]}\right) + E\left[a_j^{[\ell-1]}\right]^2 \mathrm{Var}\ \left(w_{ij}^{[\ell]}\right) + \mathrm{Var}\left(w_{ij}^{[\ell]}\right) Var\left(a_j^{[\ell-1]}\right) \tag{6}$$

In the third step of equation, variance of independent product equation is calculated as.

$$Var(XY) = E[X|^2\, Var(Y) + E \mid Y]^2\, Var(X) + Var(X)\, Var(Y) \tag{7}$$

At the final step of equation shown in below shows final result of equation

$$= n^{[\ell-1]}\, Var\left(w_{ij}^{[\ell]}\right) Var\left(a_j^{[\ell-1]}\right) \ \Rightarrow Var(W) = \frac{1}{n^{[\ell-1]}} \tag{8}$$

### 3.2.3 He Initialization

Difficulties in weight convergence are observed in Deep Neural Network models when the weights are initialized using a normal distribution with a fixed standard deviation. This is because the variance of the weights is not considered, resulting in activation values that are too high or too low, resulting in gradient problems that explode or disappear during backpropagation. This problem becomes even greater as the depth of the Neural Network increases. Kaiming He et al. developed the He initialization technique for the ReLU activation function, which also considers the activation function [14].

First things first, we need to know that X and Y are independent random variables. The X term refers to the inputs, and the Y term refers to the output obtained as a result of the prediction. W term represents the weights of the layers.

$$Var(X+Y) = Var(X) + Var(Y) \quad \textbf{(9)}$$

$$Var(XY) = Var(X)Var(Y) + (E[X])^2 Var(Y) + Var(X)(E[Y])^2 \quad \textbf{(10)}$$

Assume that $y^k = W^k * x^k + b^k$ and $x^{k+1} = f(y^k)$; k is layer number and f is activation function. In these equations y, x and b are column vectors and W is a matrix. These equations valid for Feedforward NNs as well as CNNs. Assumptions valid for each k layer which is made to achieve the He initialization.

$$* \quad W^k \ and \ y^k \ have \ zero \ mean \ and \ symmetrical \ around \ zero$$

$$* \quad First \ part, b^k \ is \ initialized \ to \ zero \ vector, initially \ do \ not \ require \ any \ bias \ value$$

$$* \quad W^k, x^k \ and \ y^k \ independent \ of \ each \ other$$

$W^k$ coefficients have been replaced with W coefficients in order to analyze the formulas more easily. Each element for $y_i$ forward equations valid for all y.

$$y_i = W_{i1} * x_1 + W_{i2} * x_2 + \cdots + W_{in} * x_n + b_i \quad \textbf{(11)}$$

$$Var(y_i) = Var(W_{i1} * x_1 + W_{i2} * x_2 + \cdots + W_{in} * x_n) \ = n * Var\left(W_{ij} * x_j\right) \quad \textbf{(12)}$$

$$= n * (Var\left(W_{ij}\right) * Var\left(x_j\right) + \left(E\left[W_{ij}\right]\right)^2 * Var\left(x_j\right) + Var\left(W_{ij} * \left(E\left[x_j\right]\right)^2\right) \quad \textbf{(13)}$$

$$= n * \left(Var(W_{ij}) * Var(x_j) + (0)^2 * Var(x_j) + Var(W_{ij}) * \left(E[x_j]\right)^2\right) \tag{14}$$

$$= n * Var(W_{ij}) * \left(Var(x_j) + \left(E[x_j]\right)^2\right) \tag{15}$$

$$= n * Var(W_{ij}) * E[x_j^2] \tag{16}$$

Important thing to know about He initialization, $E[x_j^2] \neq Var(x_j)\ unless\ E[X_j] = 0$. The cause of that He initialization customized for ReLU activation function which does not have zero mean. Since the relationship between x and y is not yet used, we can simplify the $E[x_j^2]$ term to replace the $E[x_j^2]$ term we found in the previous formula.

$$E[x^2] = \int_{-\infty}^{\infty} x^2 * P(x)dx \tag{17}$$

$$= \int_{-\infty}^{\infty} max(0, y)^2 * P(y)dy \tag{18}$$

$$= \int_{0}^{\infty} y^2 * P(y)dy \tag{19}$$

$$= 0.5 * \int_{-\infty}^{\infty} y^2 * P(y)dy \tag{20}$$

$$= 0.5 * Var(y) \tag{21}$$

By combining previously derived expression of $Var(y_i)$, to arrive at the general He initialization formula, it is possible to time decay of the index number.

$$Var(y_i^l) = 0.5 * n^l * W_{ij}^l * Var(y_j^{l-1}) \tag{22}$$

$$Var(y^l) = 0.5 * n^l * W^l * Var(y^{l-1}) \tag{23}$$

$$Var(y^L) = Var(y^1) * \left(\prod_{l=2}^{L} \frac{n^l}{2} * Var(W^l)\right) \tag{24}$$

To prevent exploding or vanishing gradients problem, variance at the input must be equal to variance at the output. This means that each term inside the product must be equal to 1.

$$\frac{n^l}{2} * Var(W^l) = 1\ ,\ \forall l \tag{25}$$

$$W \sim N\left(0, \left(\frac{2}{n^l}\right)\right) \tag{26}$$

As a summary of the He initialization, operations concluded between which values the weight assignments would be made.

### 3.2.4 LeCun Initialization

The feature of LeCun initialization aims to prevent gradients from disappearing or exploding during backpropagation. It prevents the disappearance or explosion of the gradients by solving the increasing variance with the input numbers and finding the constant variance as a result of this solution, with these fixed variances found [18]. This method was first proposed by LeCun et al [17], and LeCun initialization aims to prevent the vanishing or explosion of the gradients during the backpropagation which is mentioned before. In LeCun initiation, the errors obtained in the output layer form a function of the activations in the output layer. Therefore, the loss or explosion of gradients in the output layer will cause more errors in the operations to be performed on the other layer.

As a result, activations explode or disappear during forward propagation. For this reason, LeCun aimed to prevent activations from exploding and disappearing. One of the biggest features of the LeCun method and also the biggest disadvantage is that LeCun initialization initially only works with the tanh activation function, but today with the development of the sigmoid activation function, it can also be used with sigmoid. However, its use is still limited compared to other initialization methods.

$$\mathrm{Var}\left(\mathbb{Z}_1^{[1]}\right) = \sum_j \mathbb{E}\left(w_j\right)^2 \mathrm{Var}\left(x_j\right) + \mathbb{E}\left(x_j\right)^2 \mathrm{Var}\left(w_j\right) + \mathrm{Var}\left(x_j\right) Var\left(w_j\right) \quad \textbf{(27)}$$

$$\forall\, Var\left(\mathbb{Z}_1^{[1]}\right) = \sum_j Var\left(x_j\right) Var\left(w_j\right) \quad \textbf{(28)}$$

$$\forall\, Var\left(\mathbb{Z}_1^{[1]}\right) = n_{in}\, Var(w)\, Var(x) \quad \textbf{(29)}$$

- Aim: Activations to have the same variance as the features is shown in below

$$\begin{gathered} Var\left(\mathbb{Z}_1^{[1]}\right) = Var\,(x) \\ \Rightarrow n_{in}\, Var\,(w) = 1 \end{gathered} \quad \textbf{(30)}$$

So, $\mathrm{Var}\,(aX) = a^2 \mathrm{Var}\,(x) \Rightarrow$ need to scale our weights by $\sqrt{\frac{1}{n_{\mathrm{in}}}}$

Thus, effectively in LeCun Initialization we initialize weights from $N\left(0, \frac{1}{n_{\mathrm{in}}}\right)$ (32).

## 3.4 Results of Initialization Methods

In this section, the accuracy rates obtained by performing tests on the same PV dataset using the different starting techniques mentioned above are stated. The schema of the created algorithm is 9 (input layer), 10 (hidden layer) and 1 (output layer) as seen in Figure 3.1 below. Investigations were made using different weight initiation methods between layers. In Figure 3.1, it is seen what the different weight initiation techniques are made between the layers and how these combinations are created. In addition, the weights obtained by using the Heuristic Vectorized Learning method, on which the algorithm is built on this step, are assigned between the input layer and the hidden layer. As can be seen in Figure 3.1 for the other layer, the results were examined using different weight assignment methods. Supervised learning and LSTM are used in this review code and there are 50 epochs in LSTM. In order to examine the accuracy rates of the algorithms, the algorithm consisting of combinations of different methods was repeated 100 times and the average accuracy rate was calculated.

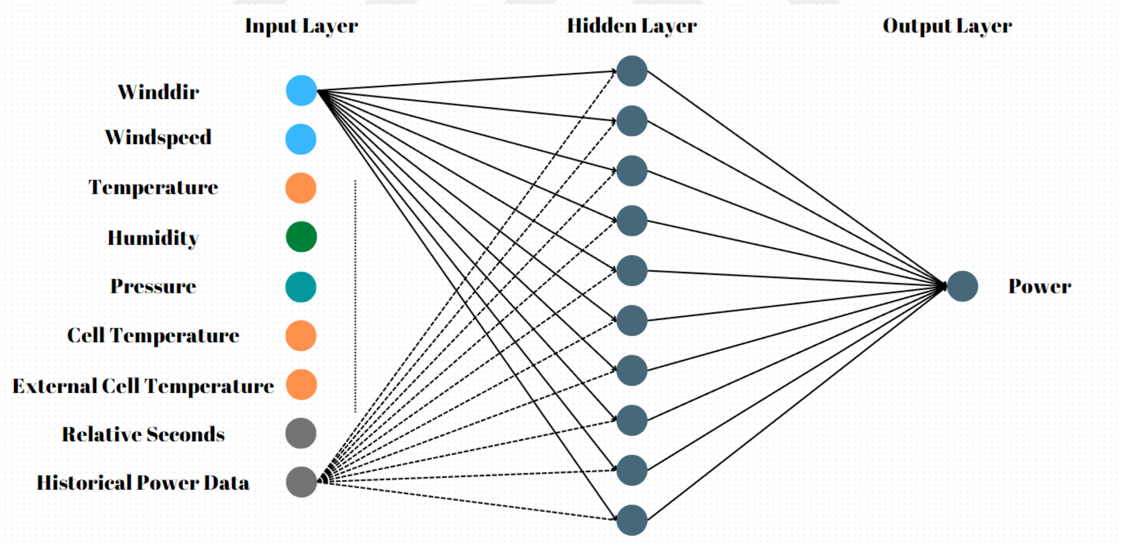



**Figure 3.1 Neural Network Scheme**

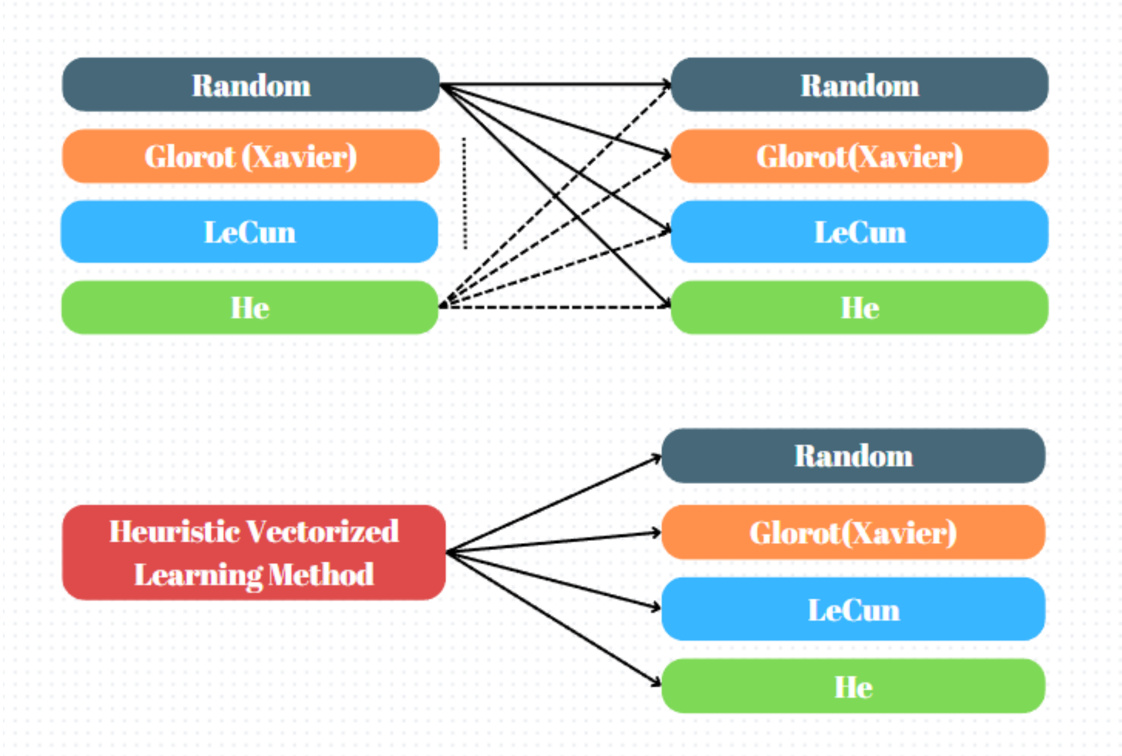



**Figure 3.2 Weight Assignment Scheme on Generated Algorithm**

### 3.4.1 LeCun Initialization Method Results

In the first stage of the combination test, the LeCun method was used to calculate the weights between the input and the hidden layer. In the LeCun method, the SeLu activation function was used as the activation function. Afterwards, the linear activation function was used as the activation function while the Glorot (Xavier) initialization method was used to calculate the weights between the hidden layer and the output layer. In the second test, while the LeCun method remained constant, the ReLU activation function was used as the activation function while the He initialization method was used to calculate the weights between the hidden layer and the output layer. In the third test, while the LeCun method remained constant, the ReLU activation function was used as the activation function in the Random initialization method to calculate the weights between the hidden layer and the output layer. The Figure 3.3 represents the results of the tests performed.

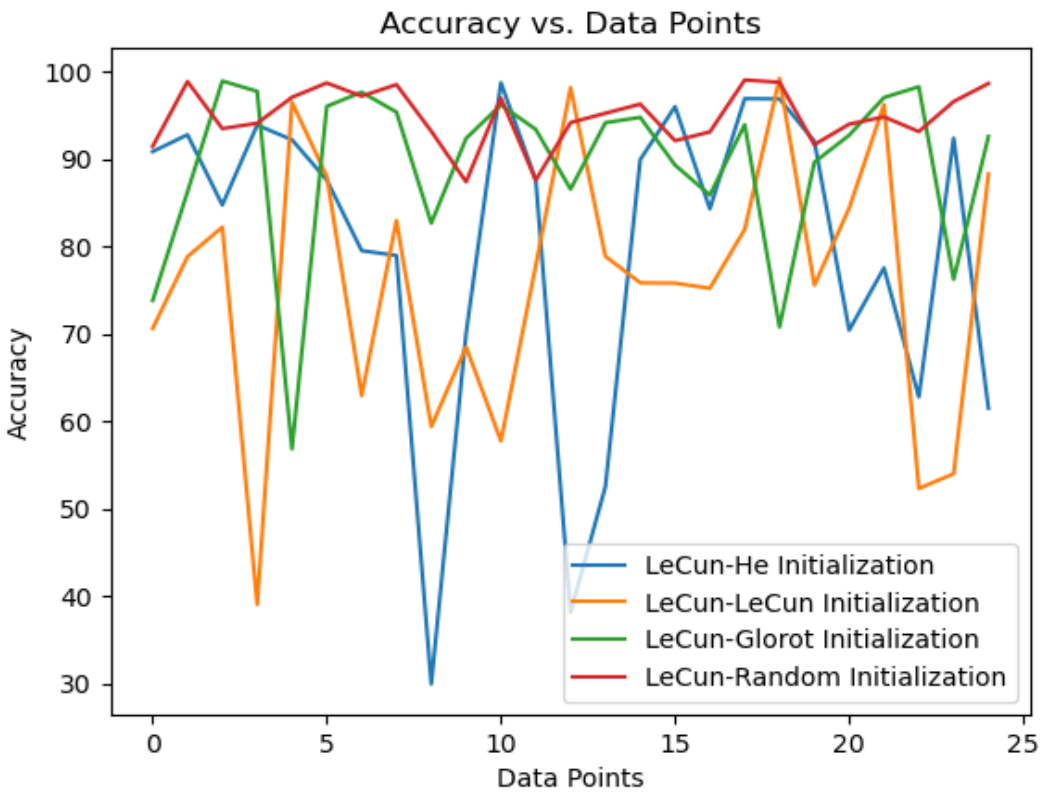



**Figure 3.3 LeCun- General Initialization Techniques Test Results**

### 3.4.2 Glorot (Xavier) Initialization Method Results

At the first stage of the second combination test, the Glorot initialization method was used to calculate the weights between the input and the hidden layer. In the Glorot initialization method, the linear activation function was used as the activation function. Afterwards, the LeCun initialization method was used to calculate the weights between the hidden layer and the output layer with use SeLu activation function. In the second test, the Glorot initialization method remained constant, the He initialization method was used to calculate the weights between the hidden layer and the output layer with use ReLU activation function. In the third test, while the Glorot initialization method remained constant, the ReLU activation function was used as the activation function in the Random initialization method to calculate the weights between the hidden layer and the output layer. The Figure 3.4 represents the results of the tests performed.

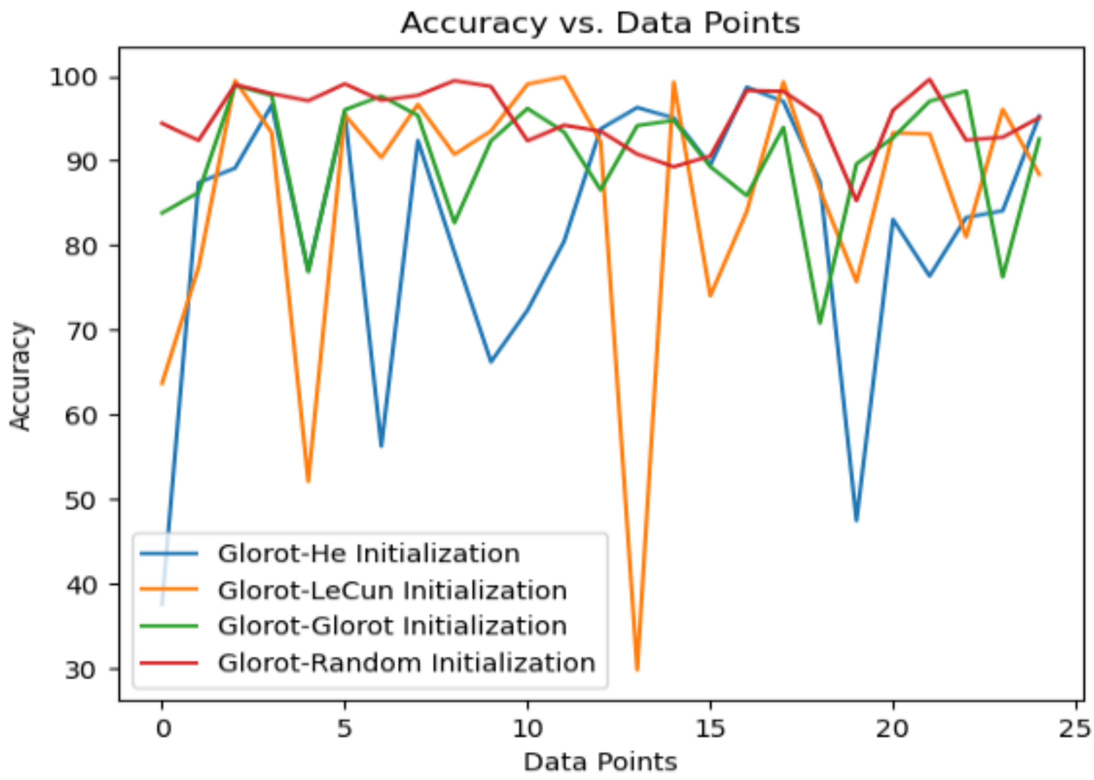



**Figure 3.4 LeCun-General Initialization Techniques Test Results**

### 3.4.3 He Initialization Method Results

In the first stage of the third combination test, He weight initialization method, which is one of the weight assignments, was used to assign weight between the input layer and the hidden layer. In order to see the effect of the He initialization method between the input layer and the hidden layer, the initialization methods in the second layer were changed by keeping the He method constant in the He initialization tests. He initialization method is an initialization technique customized to the ReLU activation function. Therefore, its use with different activation functions is not recommended in terms of correct operability of the model.

In the first test, LeCun was chosen as the weight assignment method between the hidden layer and the output layer. The LeCun weight assignment method works with the SeLU activation function, which is customized for it. In another test, the Glorot method was chosen, and linear activation function was used to calculate the weights between the hidden layer and the output layer. In the final test, the Random weight assignment method ReLU activation function was used. As a result of the tests performed, the accuracy percentages between the forecast and actual values are shown in the charts below.

Likewise, when the use of the He initialization method is kept constant between the input layer and the hidden layer, the combination results obtained with other methods are represented in Figure 3.5

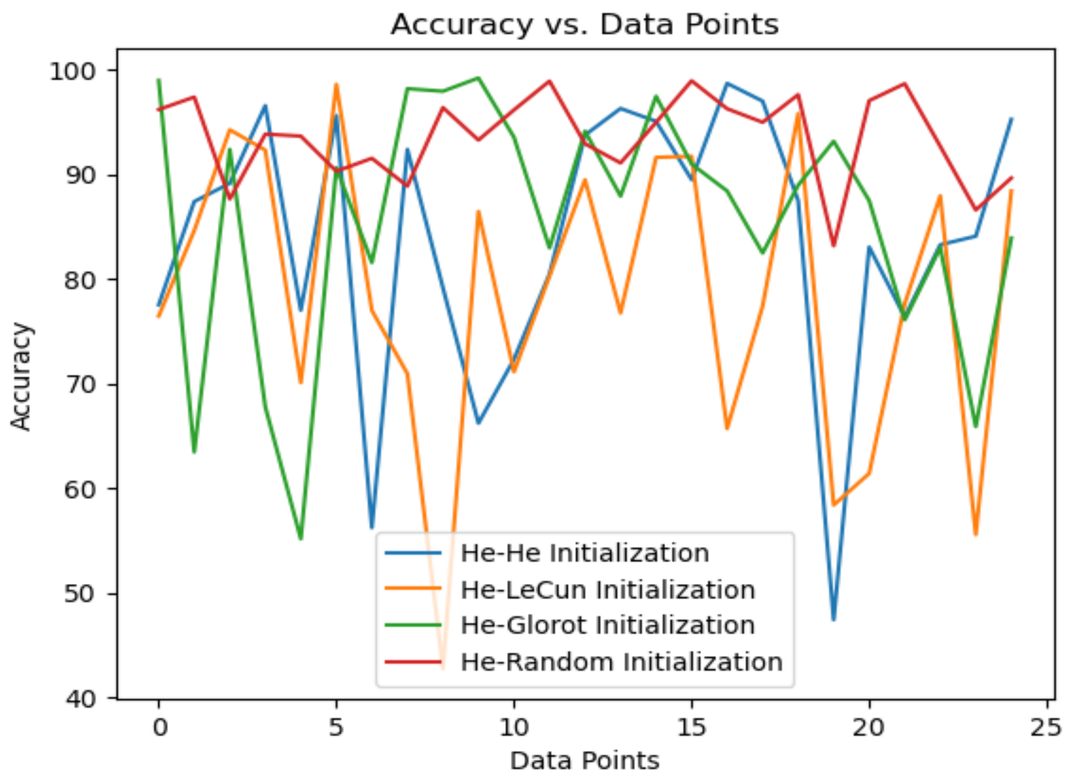



**Figure 3.5 He- General Initialization Techniques Test Results**

### 3.4.3 Random Initialization Method Results

In the first step of the final combination test, the Random weight initialization method was used to calculate the weights between the input layer and the hidden layer. It has always been applied between the input layer and the hidden layer throughout the tests. In the first test, weight assignment was made by applying the He initialization method between the hidden layer and the output layer. ReLU activation function is selected in Random and He method. In the next test, the LeCun initialization method is applied between the hidden layer and the output layer. The SeLU function was chosen as the activation function for the LeCun technique. In the last test, weight calculations were made by applying the Glorot initialization method between the hidden layer and the output layer. The linear function was preferred as the activation function of the Glorot technique. The Figure 3.6 represents the accuracy values of the tests performed as a result of different combinations.

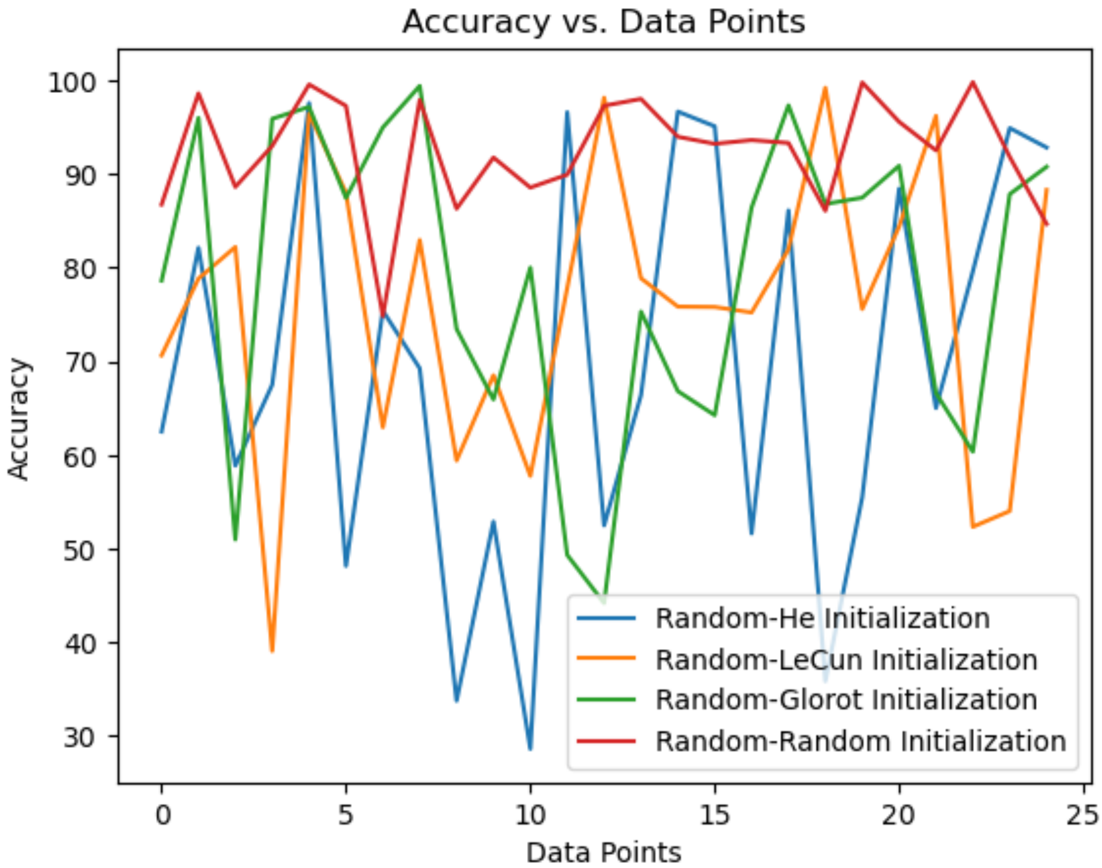



**Figure 3.6 Random- General Initialization Techniques Test Results**

# Chapter 4

# Heuristic Vectorized Learning Method (HVLM)

Renewable energy penetration levels have increased, especially since the highest penetration is reached by solar photovoltaic (PV) integration. Due to the intermittency nature of PV production, forecasts have become more important. A new artificial intelligence (AI) based weight initialization method, Heuristic Vectorized Learning Method, has been applied on PV forecasting system is proposed with integrating the cloud thickness, movement of the clouds, and position of the sun. For higher accuracy, monitoring the sky is the best option for intra-day forecasts. Also, the panel shadowing model was developed by calculating the cloud thickness, the movement of the clouds, and the position of the sun by using a deep learning approach. While cloud motion forecasting is handled by using a gated recurrent unit, cloud motion-thickness forecast is integrated by using multiple convolutional neural networks. PV generation forecast is obtained by a hybrid model for environmental sensor data and panel shadowing model in the deep learning-based artificial neural network. It was observed that with the proposed method, along with the inclusion of cloud motion-thickness in the energy production forecasting in the deep learning model, much better accuracy values were obtained in Abdullah Gul University campus, located in Kayseri, Türkiye. The proposed system provides accurate one hour-ahead PV forecasting for reliable for the energy markets.

## 4.1 Introduction

The process of generating power using solar photovoltaic (PV) includes converting energy from the sun into electricity using solar cells. As with all the energy conversions, the efficiency is not perfect and requires applications to track and optimize it. The efficiency of the conversion is dependent on many environmental factors. In general, the nine most important factors will be considered are humidity, weather temperature, cell temperature, external cell temperature, wind speed, wind direction, and atmospheric pressure. Although, these factors have been used many applications in literature, cloud motions and thickness values are neglected to be obtaining correct forecasting.

In PV based power systems, there is a massive impact arose from the cost, due to lack of accurate forecast of PV generation. Besides to the cost, the frequency balance in the system is also corrupted and the power system management becomes an issue [50]–[54]. The intermittent generation and uncertain behaviors of renewable energy sources negatively affect power system operations such as power quality, stability, and reliability [68] To come up with a solution for these problems, there are several approaches proposed to forecast the power generation. These approaches contain methods which can be categorized into four types as persistence, statistical, machine-learning, and hybrid method based on the use of historical data of PV power output and related environmental variables (sensor values) [50], [56].

The simplest stochastic learning technique is the persistence forecast which is based on current or recent PV power plant or radiometer output and extrapolated to account for changing sun angles. Persistence forecast accuracy highly relies on forecast duration as cloudiness varies between states [57]. The statistical approach is based on previously recorded measured time data series and mostly seen in short horizons. They are simpler than physical models due to less input data requirement and less computation efforts for the power forecasting [58].

In recent years, various machine learning approaches including deep learning algorithms such as ANN (Artificial Neural Network), CNN (Convolutional Neural Network), RNN (Recurrent Neural Network), and LSTM (Long Short Term Memory) are also becoming quite famous in PV Power forecasting [59]–[61]. However, the algorithms developed for PV power forecasting provide accuracy rates generally between 90%-95% around the world because of the number of environmental and conditional changes that affect the power generation process such as temperature, humidity, air pressure, panel shadowing, and wind etc. Moreover, machine learning includes the complexity of computation along with the desirable accuracy rates.

Majority of the parameters that changes in a day are not far from predictability, but because of the stochastic structure of cloud motions, power generation process evolves to unforecastable scheme. As the clouds move between panel and the sun, the energy generation ratio drops to considerably an insufficient level. A study shows that multiple deep learning models with the integration of both static sky image units and dynamic sky image flow are explicitly explored for short-term prediction of cloud motion. Even if the developed model successfully detects and segments the clouds, it is not capable of forecasting ahead of one hour. In addition, the examined systems do not have the data abundance of the newly developed system, that is, environmental data such as temperature and humidity were not used except for the camera image. For this reason, the newly developed model has longer forward forecasting and better accuracy than previously designed models [62]. For the short-term PV Generation forecasting, it is obvious that without investigation of cloud motion, this study reveals that following the clouds in very short 15-minute forward forecasts increased the accuracy of the system by 2%. Moreover, when this result is compared with the newly designed model, the newly designed model has increased its accuracy by approximately 4% with cloud motion and thickness data. However, this increase is for a forward-hour forecast, not for a very short term of 15 minutes. [63].

Deep learning-based cloud detection and segmentation studies, in which sky images were collected with the fisheye camera used in previous years for solar energy forecasting, resulted in a commendable accuracy with 15-minute and less forward forecasting [62], [64], [69].

There are forward-looking PV forecasting studies in different methods than the proposed model. Deep learning algorithms Back Propagation Neural Network (BPNN) and different cloud segmentation algorithms k-means clustering have been used[65]–[67]. The main differences from the proposes system are that the cloud thickness is not included in the cloud movement forecasting, as the forecast time increases, the accuracy values decrease rapidly. The proposed deep learning model gives results with very high accuracy in the forecasting ahead of one hour by integrating the cloud thickness into forecasting.

Considering the aforementioned issues aspects, this study was developed by incorporating cloud thickness as well as cloud motion into the estimation of energy production in the deep learning model. In the forecasting process, algorithms were developed for cloud thickness, cloud movement, and sun position, and a system was created to feed the deep learning algorithm. And data collection hardware application for learning the deep learning algorithm was also carried out. The main contributions are summarized as follows:

- Cloud thickness-motion forecast integrated in PV forecasting by using multiple convolutional neural networks mainly named as the adaptive simple linear iterative clustering (A-SLIC) algorithm has been proposed.
- Deep learning-based Cloud Detection has been used. Cloud motion forecasting handled by using A Gated Recurrent Unit (GRU) that is one of the types of Recurrent Neural Network.
- Creating an integrated cloud motion and thickness deep learning model by combining cloud motion and thickness estimation results with environmental factors.
- Not only the collected data were utilized in simulation environment, but also the hardware implementation was carried out in a real university environment.

## 4.2 Sensor Based PV Forecasting

In the deep learning algorithm, environmental variables are used for future PV energy production forecasting. Cloud images are collected with a 170-degree fish-eye camera. In addition, there are sensors that collect humidity, weather temperature, cell temperature, external cell temperature, wind speed, wind direction, and atmospheric pressure data. The sensor and camera are installed on the roof of a building with PV for real application in Abdullah Gul University campus, as can be seen in Figure 4.1

**Figure 4.1 Sensors and Fisheye Camera Installation**

A Raspberry Pi 3 is embedded in the hardware system for recording and remote communication with the environmental variables. By programming, Raspberry starts collecting data early in the morning and finishes the data collection process at the end of the day. Humidity, weather temperature, cell temperature, external cell temperature, wind speed, wind direction, atmospheric pressure data and sky images to be used in the cloud motion thickness model are collected. Recorded data can be retrieved at any time by establishing a remote connection with raspberry.

PV production data is obtained from the PV panels installed in university. It consists of 5 minutes average data of PV production data. After the environmental variables and PV production data were collected, the dataset was started to be created. One hour of PV production data consists of 12 pieces of data which 5-minute average for an hour of power generation. On the other hand, the average of the data of environmental variables was fixed in a way that 12 features of data per hour to match features and power generation.

The dataset includes 8 days of environmental variables and 8 days of PV production. Thus, it contains 80 hours of historical data in total. A separate visual dataset was created for 8-day sky images to be used in the cloud motion thickness model. The dataset containing environmental data and PV production will be used in the Sensor Based PV Forecasting model.

# 4.3 Cloud Motion and Thickness Forecast

In this part, it is possible to propose multiple convolutional neural networks created for high-resolution remote-sensing imagery. The main method used is to apply the adaptive simple linear iterative clustering (A-SLIC) algorithm to the segmentation of the raw input images. Convolutional neural networks' (MCNNs) architecture separates multi-scale features from each super pixel. Considering the outputs, the cloud is classified into different forms of cloud. A model contains four convolutional layers and two fully connected layers. The input is clustered into sub-region through a simple linear iterative cluster (SLIC) method. This model is able to identify whether the cloud is thin or thick [70], [71]. Deep learning is applied to the raw image fed by the camera input to forecast the possible position of the clouds for the near future (e.g. 3-min later). Then compared with the actual position of the clouds with the forecasting. Figure 4.2 shows the result of cloud motion ahead 3 minutes and it proves that the clouds had move through to the sun. Figure 4.3 illustrates the result of cloud thickness forecast. As it can be seen, thickness level is represented on the image with different colours.

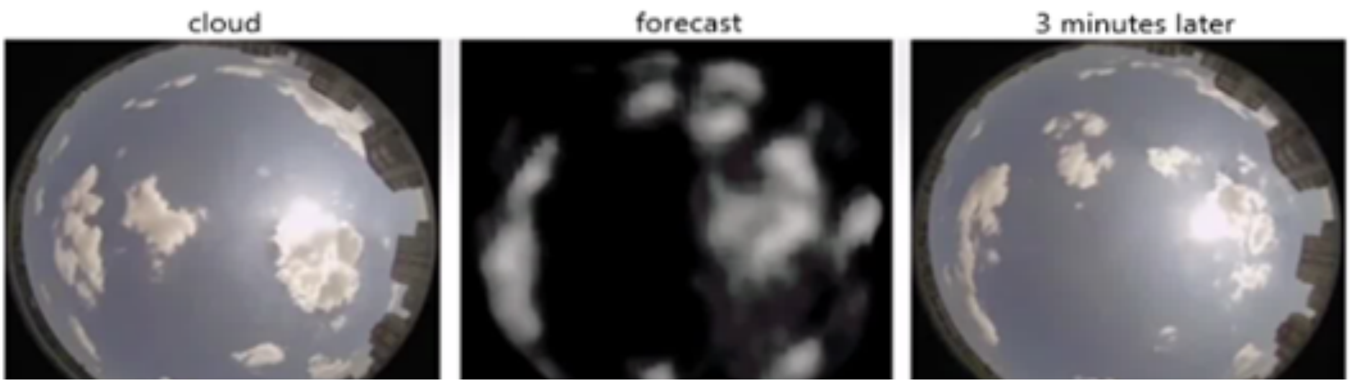



**Figure 4.2 Cloud Motion Forecasting Result**

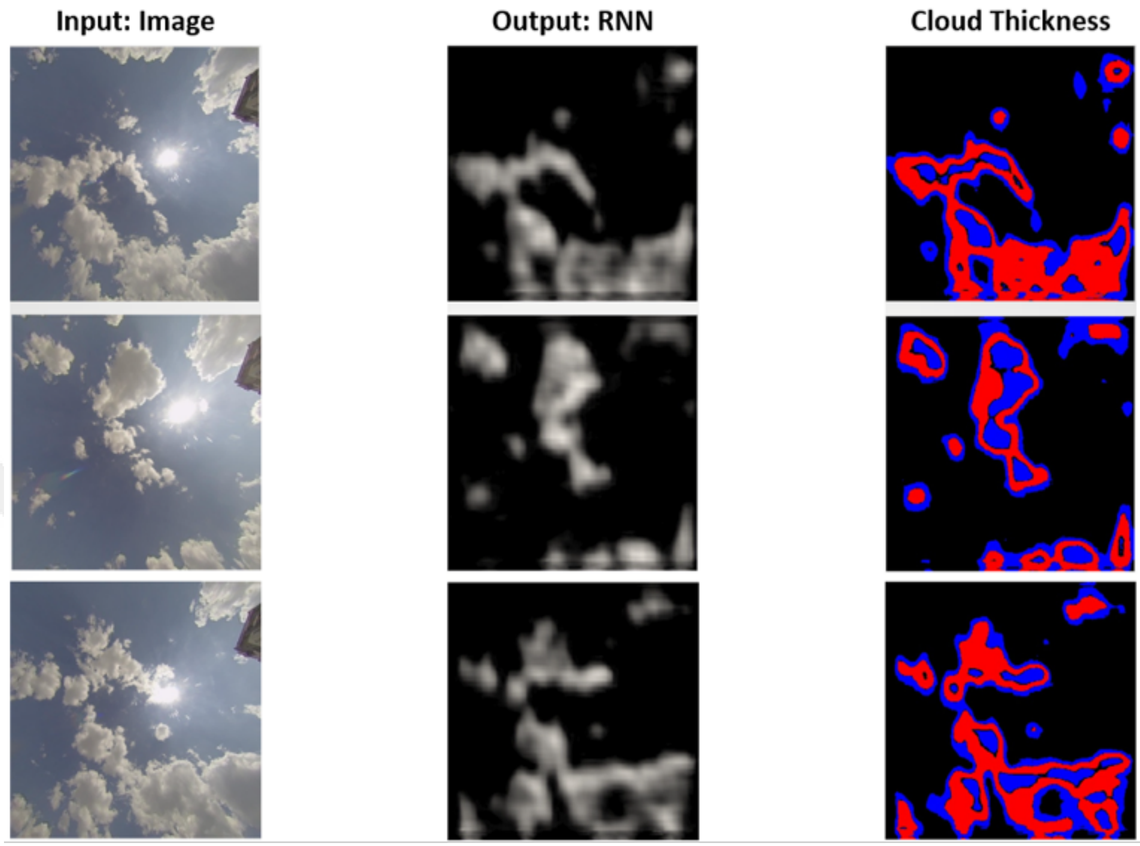



**Figure 4.3 Cloud Thickness Forecasting Result**

The flow chart in Figure 4.4 shows a visual representation of the whole system consisting of the camera and environmental sensor data, cloud mapping, determining sun position, and combining these data in a deep learning method which use artificial neural network algorithm.

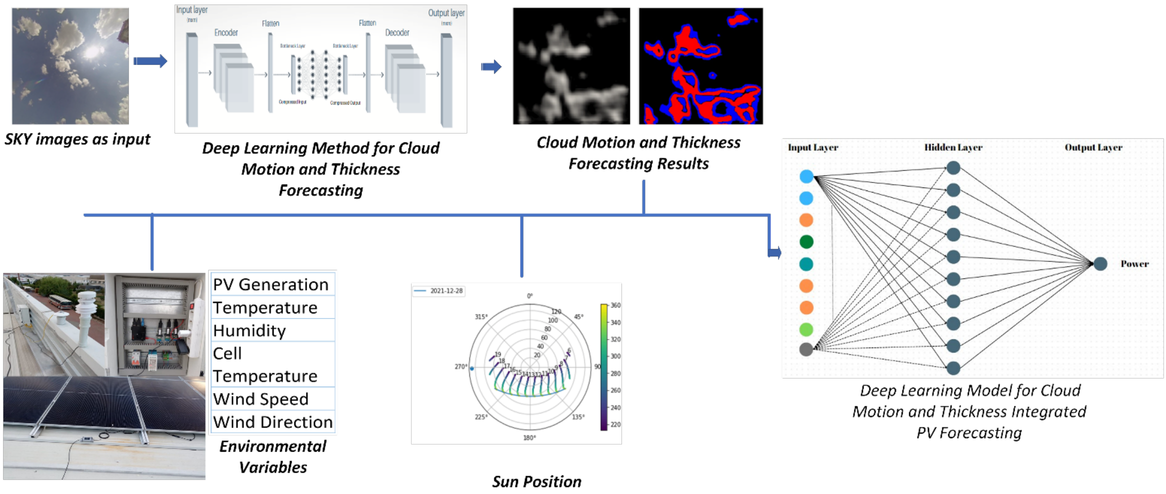


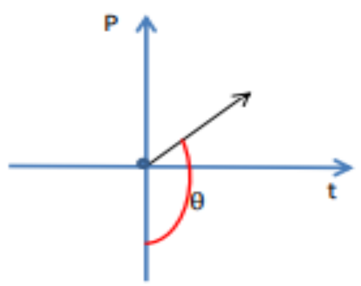

**Figure 4.4 Visual representing of the cloud motion and thickness integrated PV forecasting model**

## 4.4 Heuristic Vectorized Learning Method

The heuristic vectorization method is defined as a problem-solving technique that examines and uses minimal information, past results, or relationships between data to produce an applicable and practical solution to a specific problem in a short time. In machine learning and artificial intelligence disciplines, when there is a high level of data flow, it is one of the primary approaches to be used when it is not practical to follow a step-by-step algorithm. In this article, the purpose of the Heuristic Vectorized Learning Method (HVLM) is to calculate the effect on the desired output by determining the rows and columns where other variables remain constant while one variable changes. After the calculations, the weight initialization process was performed, accelerating the estimation algorithm, resulting in its execution without loss of time.

$$f_1 = r_1(cosQ_1 j + sinQ_1 j) \tag{1}$$

f1 is the vector of feature 1, i and j are the unit vectors, r1 is the magnitude and Q is the angle from negative axis.




$$f(Q)_\gamma = \begin{cases} r_\gamma[\cos(Q_\gamma - 90)\, i - \sin(Q_\gamma - 90)\, j], & Q_\gamma > 90 \\ r_\gamma[\cos(Q_\gamma)\, i - \sin(Q_\gamma)\, j], & else \end{cases}$$

ɤ shows the values for each iteration. Q>90 means the negative slope, and in the negative slope, the angle Q is extracted from 90 due to the trigonometric calculation.

f is the vector feature vectors, multiplication of environmental variables with the coefficients, summing all the features, the generation prediction is obtained.

In a daily generation prediction, the vectors will be constituted from past time to the current time in a day. The last data point of parameters is multiplied by their particular vectors. To obtain also how these parameters will change in a day, a general daily form of each parameter will be extracted. Then, the last data points of parameters will occupy a vector to see where exactly they are changing. Multiplying these vectors with specified coefficients of them, we will also obtain the changes.

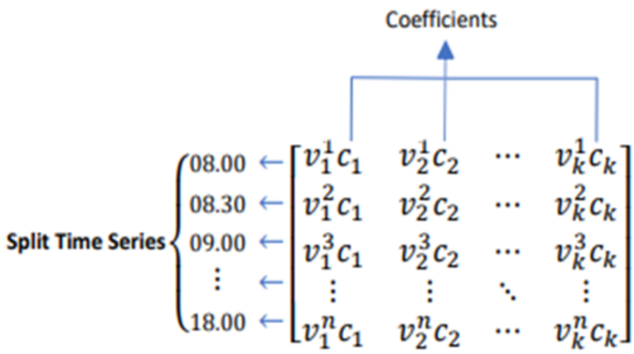



**Figure 4.5 C is the coefficient of each variable, and V is the environmental variables in split time series.**

## 4.5 HVLM Results

The model was tested by using different weight assignment techniques together with the Heuristic Vectorised Learning method. Weight assignments between the input layer and the hidden layer were assigned using the HVLM. It is tested using four other different assignment methods used in model testing between the hidden layer and the output layer. As was done in the previous tests, the average accuracy values obtained by making the model 100 repetitions are stated. With the HVLM, the convergence of the model has increased, speeding up the model for power estimation. It has greatly accelerated the model by taking the weight assignment load in the first layer that the model needs to calculate.

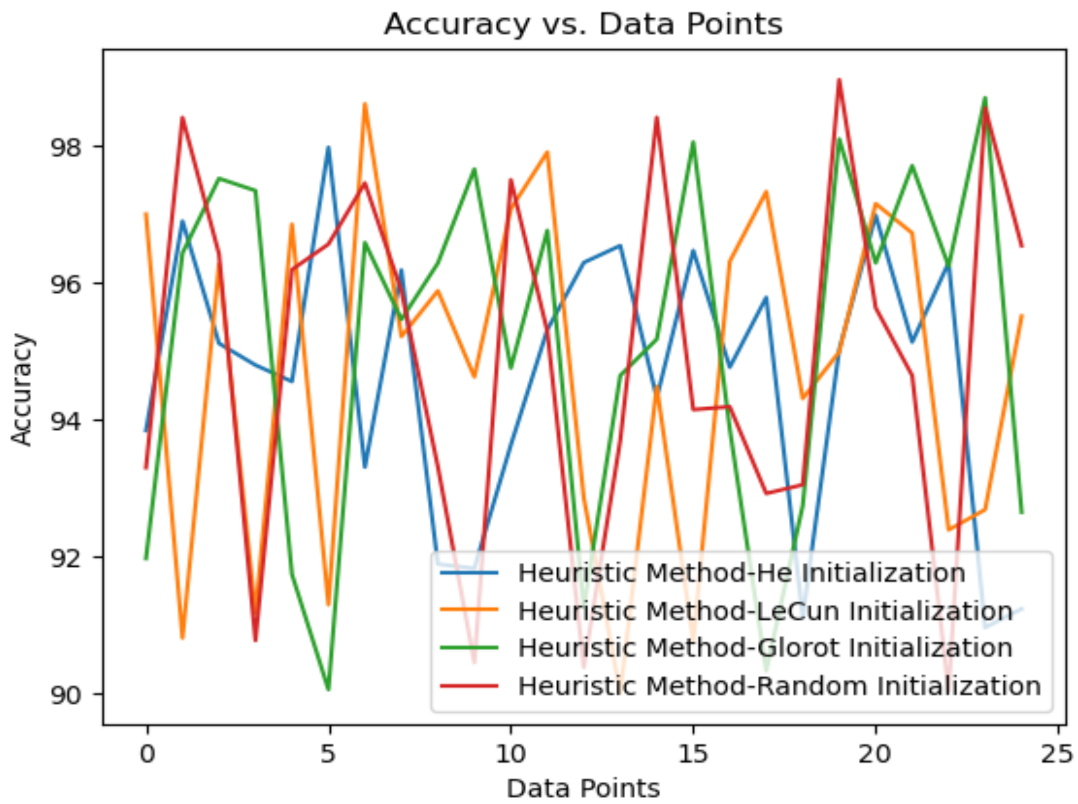



**Figure 4.6 Heuristic Vectorised Learning Method- General Initialization Techniques Test Results**

**Table 4.1 Test Results Of Weight Initialization Techniques**

| **Initialization Methods** | **LECUN** | **Glorot** | **He** | **Random** |
|---|---|---|---|---|
| **LeCun** | 85.8538% | 87.3865% | 80.1131% | 94.8394% |
| **Glorot** | 84.4629% | 86.6916% | 82.0226% | 95.6741% |
| **He** | 81.4912% | 85.0668% | 83.1321% | 94.8653% |
| **Random** | 74.6694% | 80.7122% | 68.5723% | 93.6943% |

**Table 4.2 Test Results Of Heuristic Vectorized Method And Weight Initialization Techniques**

| Methods | LECUN | Glorot | He | Random |
|---|---|---|---|---|
| **Heuristic Vectorized Learning** | 95.3602% | 94.5553% | 94.8306% | **95.7069%** |

The accuracy percentages obtained as a result of the tests performed on the model are given in Table 4.2 As a result of the tests, the highest accuracy value of the model with the PV data set was obtained in the combination of Glorot and Random weighting assignment with a ratio of 95.6741. The Glorot weight initialization method given between the input and hidden layer and the random weight method given between the hidden layer and the output layer, when their different advantages come together, increase the operating performance of the model to a high degree. In addition, when an overview of the table is taken, the highest accuracy rates are observed when the random weight assignment method is given between the hidden layer and the output layer.

The accuracy values obtained using the Heuristic Vectorized Learning method are given in Table 4.2 The test process is started by giving the weight values obtained by the Heuristic Vectorized method to the model between the input and the hidden layer. The other four weight assignment methods used in the tests are applied between the hidden layer and the output layer, and it is examined in which combination the model performs better. Compared to other assignment techniques, Heuristic Vectorized Learning method achieves results by eliminating problems such as faster results in the model, avoidance of high data usage, and gradient loss/explosion that can be encountered in assignment methods. As can be seen in the Table 4.3, the accuracy values obtained with the heuristic method have a higher accuracy rate than other weight assignment techniques and reach the result faster at the same workload.

**Table 4.3 Elapsed Time Results of Weight Initialization Techniques and Heuristic Vectorized Method**

| Initialization Methods | LECUN | Glorot | He | Random |
|---|---|---|---|---|
| **LeCun** | 3.893498 s | 4.37682 s | 4.32139 s | 4.483223 s |
| **Glorot** | 4.475323 s | 4.51739 s | 4.45921 s | 4.393425 s |
| **He** | 4.572191 s | 4.51201 s | 4.37042 s | 4.465539 s |
| **Random** | 4.326623 s | 4.52567 s | 4.33881 s | 4.309628 s |

A table has been presented above that indicates the duration of test results obtained through various weight initialization techniques, along with the Heuristic Vectorized Learning method. The results reveal that the Heuristic Vectorized Learning method outperforms conventional weight assignment techniques by providing highly accurate results in a comparatively shorter time.

Various weight assignment techniques are used in the working mechanism of the structures used in data estimation in algorithms created in neural networks. There are functions for which the weight assignment techniques used are the optimum working range or customized. For this reason, models created in neural network disciplines are not always close to perfect.

The random weight assignment method is one of the most preferred weight assignment methods in algorithms. It improves the symmetry breaking process and increases the working speed of the algorithm with its fast convergence feature. But on the other hand, it can cause the weights assigned in forward and backward propagation to enter the saturation point and the gradient to disappear in gradual descent.

LeCun weight assignment technique, on the other hand, gives the best performance with the SeLU activation function, which is a customized function. LeCun weight assignment technique has the effects of solving the increasing variance problem and eliminating gradient problems. On the other hand, taking into account the input data in forward propagation and using it in fixed-width networks affects the efficient operation of the algorithm. In addition, as mentioned above, the use of LeCun weight assignment method, which gives the best efficiency with its customized activation function, with non-differentiable activation functions will cause gradient problems and negatively affect the operation of the algorithm.

Another method used in the algorithm is the Xavier weight assignment method. The Xavier weight assignment method can eliminate the gradient dissipation/burst problems that will occur in forward and backward propagation. Except that, this weight assignment method is not used with non-differentiable activation functions like the LeCun method. In addition, neuron deaths in the layers can be seen during the data training phase of the algorithm.

The last weight initialization method used in the test is the He method. He weights initialization method solves dying neuron problems to a high degree according to Xavier method. In addition, it eliminates the possibility of problems such as gradient disappearance/explosion that may occur in forward and backward propagation. On the contrary, using He weight initialization method with differentiable activation functions between layers in neural networks reduces the working performance of the model.

The Heuristic Vectorized Learning Method is preferred to be used when processing high-order data among algorithms based on neural networks. This method also provides convenience in model building in terms of understanding and implementing. Instead of using random weight initialization techniques among the models established in neural networks, weight assignment is made by taking a small part of the data. In this way, the working speed of the model increases and the convergence time decreases. It adds flexibility to the model and increases operational efficiency. Among other weight initialization techniques, problems such as gradient disappearance/explosion and neuron death are not encountered.

Each of the tests using weight initialization techniques was performed using 50 epochs and the results were recorded. Each of the tests performed using the Heuristic Vectorized Learning method consists of 10 epochs. The Heuristic Vectorized Learning method shows higher performance in a shorter time with the same data set. The Heuristic Vectorized Learning method applied for weight initialization between the input layer and the hidden layer provides fast short-term solutions to the model, allowing it to reach the result in a short time by avoiding high-degree descending processes for optimum weight initialization. In this way, it increases the convergence performance of the model. Moreover, unlike other weight initialization techniques, it prevents gradient disappearance/burst, neuron death and increasing variance problems. Using different weight initialization techniques in the model created using neural networks, Heuristic Vectorized Learning method performs lower than the model used, it lasts faster and cannot achieve a stable value as a result of 100 tests on accuracy. The fluctuations in the accuracy values obtained when weight initialization techniques are used are high. Therefore, it is not correct to say that the model is a healthy result even if the average value is high. When looking at the Heuristic Vectorized Learning method, the observed fluctuations in the accuracy value of the model are less. The

Heuristic Vectorized Learning method shows that it is more efficient than other weight initialization techniques, with its high performance compared to other weight initialization techniques and the shorter duration of the algorithm.

# Chapter 5

# CONCLUSION AND SOCIAL IMPACT

## 5.1 Conclusion

In Chapter 1 VPP presented a comprehensive overview. PV panels highlighting and their significance of the research has been showed. Additionally, it delineated the study's research inquiries and goals, along with the approach employed to tackle these inquiries.

In chapter 2, chapter explores the wide-ranging applications of machine learning (ML) in the field of power generation, particularly in photovoltaic (PV) power systems. ML is utilized for various purposes such as power generation predictions, cyber security, fault detection, and load predictions. The focus of this chapter is on fault detection in power systems and the implementation of a self-healing ensemble machine learning algorithm. The aim is to propose a cost-efficient and adaptive algorithm that can accurately detect faults in real-time, even in challenging scenarios.

In chapter 3, ML applications differ from traditional computer algorithms in that they create specialized algorithms tailored to specific data or situations. In the field of power systems, researchers face numerous challenges that are well-suited for ML applications. Power transmission and distribution problems involve a multitude of variables and states, prompting the application of various solutions. This chapter focuses on exploring algorithms and methods that combine weak learning algorithms to create a robust learner through additional processes. Weak learners operate sequentially, with each predictor attempting to improve upon the previous results using boosting methods. Another approach involves combining the outcomes of weak learners through the bagging method. As a result, the proposed method exhibits flexibility and adaptability to structural changes, which are common occurrences in power systems. This method does not rely on signal processing or pre-processing techniques like feature selection.

In chapter 4, Tests were made on different model algorithms prepared for power generation estimation using PV data sets, and the results were compared. Although weight initialization techniques offer solutions to the problems that may be encountered in the model created based on neural networks, they do not provide as high performance and speed as the Heuristic Vectorized Learning method. The main starting points of weight assignment techniques are to provide solutions to problems that may occur in the algorithm such as gradient burst/disappearance, increasing variance problems. It brings the algorithm to the desired target by making a random weight initialization based on a mathematical formulation. In this process, random initialization techniques cannot reach a stable performance level. In the random weight initialization, it performs in the epochs, the neural networks can converge quickly from time to time and can perform at a high accuracy rate. However, in the light of repetitions, it is observed that a stable accuracy rate cannot be achieved when the results obtained with weight initialization techniques are compared, and a great decrease is observed when the accuracy value of the tests is averaged. The Heuristic Vectorized Learning method is used as a calculation technique for the weights between the input layer and the hidden layer of the model created based on neural networks, with minor data load, by utilizing the time intervals in the data in order to solve the problems quickly. Since the weight initialization made with the Heuristic Vectorized Learning method are created by examining the data, problems such as gradient burst/disappearance have been avoided, and problems resulting in the death of the input, hidden and output layers have not been encountered in the algorithms. As has been demonstrated by the tests created on the model, the Heuristic Vectorized Learning method can be customized on PV datasets and offers faster and higher accuracy than random weight initialization techniques.

## 5.2 Social Impact and Future Prospect

Creating a machine learning algorithm with high accuracy for PV power generation can have significant social impacts. Here are a few potential social implications:

1- Renewable Energy Adoption: By developing an accurate machine learning algorithm for PV power generation, it can help enhance the efficiency and reliability of renewable energy systems. This can encourage the widespread adoption of solar power, reducing reliance on fossil fuels and mitigating environmental issues such as climate change and air pollution.

2- Energy Cost Reduction: A precise machine learning algorithm can optimize the performance of PV power generation systems, leading to cost reductions in electricity production. This can make solar energy more affordable and accessible to a broader population, potentially reducing energy poverty and improving energy equity.
3- Energy Security: Increased accuracy in PV power generation predictions can enhance energy planning and management. It allows for better forecasting of power generation, enabling more efficient utilization of solar energy resources and facilitating grid integration. This can contribute to energy security by reducing dependence on external energy sources and improving overall system resilience.
4- Job Creation and Economic Growth: The development and implementation of advanced machine learning algorithms in the PV power sector can stimulate innovation, research, and development. This, in turn, can create job opportunities in the renewable energy industry and foster economic growth in related sectors.
5- Environmental Benefits: High accuracy in PV power generation prediction helps optimize system performance, enabling better utilization of solar resources. By increasing the efficiency of solar energy conversion, it reduces greenhouse gas emissions associated with conventional energy generation, thus contributing to environmental preservation and a more sustainable future.

It's important to note that the social impact of a new approach of the machine learning algorithm depends on its successful implementation and integration into real-world systems.

And also creating a new way on machine learning algorithms with high accuracy for PV power generation holds promising future prospects. Here are some potential areas of impact and future prospects:

1- Enhanced Efficiency: A new machine learning algorithm can further optimize the efficiency of PV power generation systems. By accurately predicting solar energy output, the algorithm can improve the overall performance of solar panels, tracking mechanisms, and energy storage systems. This increased efficiency will contribute to higher power generation and improved cost-effectiveness.

2- Grid Integration and Stability: Accurate machine learning algorithms can facilitate better integration of PV power into the existing energy grid. Advanced algorithms can help address grid stability issues by predicting power fluctuations, voltage variations, and reactive power requirements. This enables seamless integration of solar energy into the grid, reducing the need for conventional backup power sources and enhancing overall grid stability.

3- Advanced Forecasting and Planning: Improved machine learning algorithms can offer more accurate solar power generation forecasts on different timescales. This enables utilities and energy planners to optimize resource allocation, plan maintenance schedules, and make informed decisions about grid expansion and energy trading. Enhanced forecasting can also support the integration of other renewable energy sources and improve overall energy management.

4- Smart Energy Management Systems: Machine learning algorithms can play a crucial role in developing intelligent energy management systems. These systems can optimize the operation of PV power generation, energy storage, and consumption based on real-time data, demand patterns, and weather conditions. By dynamically adjusting energy flow and storage, such systems can maximize the utilization of solar energy and minimize waste.

5- Technological Innovations: Advancements in machine learning algorithms for PV power generation can drive technological innovations. This includes the development of more efficient solar panel designs, sophisticated monitoring systems, and novel energy storage solutions. These innovations can contribute to reducing the cost of solar power systems, increasing their scalability, and expanding their applicability in various settings.

It's important to note that the future prospects of machine learning algorithms in PV power generation depend on continuous research, development, and collaboration between academia, industry, and policymakers. Ethical considerations, regulatory frameworks, and addressing challenges related to data availability and privacy will also play a crucial role in shaping the future of these algorithms.

# BIBLIOGRAPHY


[1] L. L. Grigsby, *Power system stability and control*. 2017. doi: 10.4324/b12113.

[2] "Update To Renewable Energy Integration In South Australia Joint Aemo And Electranet Report New South Wales Queensland South Australia Victoria Australian Capital Territory Tasmania Western Australia Important Notice Purpose," 2016.

[3] A. Esmaeilian and M. Kezunovic, "Prevention of power grid blackouts using intentional islanding scheme," *IEEE Trans Ind Appl*, vol. 53, no. 1, pp. 622–629, 2017, doi: 10.1109/TIA.2016.2614772.

[4] S. Raza, H. Mokhlis, H. Arof, J. A. Laghari, and L. Wang, "Application of signal processing techniques for islanding detection of distributed generation in distribution network: A review," *Energy Convers Manag*, vol. 96, pp. 613–624, May 2015, doi: 10.1016/J.ENCONMAN.2015.03.029.

[5] D. S. Pillai and N. Rajasekar, "A comprehensive review on protection challenges and fault diagnosis in PV systems," *Renewable and Sustainable Energy Reviews*, vol. 91, no. April, pp. 18–40, 2018, doi: 10.1016/j.rser.2018.03.082.

[6] D. S. Pillai and N. Rajasekar, "Metaheuristic algorithms for PV parameter identification: A comprehensive review with an application to threshold setting for fault detection in PV systems," *Renewable and Sustainable Energy Reviews*, vol. 82, no. October 2017, pp. 3503–3525, 2018, doi: 10.1016/j.rser.2017.10.107.

[7] J. Keller and B. Kroposki, "Understanding Fault Characteristics of Inverter-Based Distributed Energy Resources," 2010. doi: 10.2172/971441.

[8] H. Jiang, J. J. Zhang, W. Gao, and Z. Wu, "Fault detection, identification, and location in smart grid based on data-driven computational methods," *IEEE Trans Smart Grid*, vol. 5, no. 6, pp. 2947–2956, 2014, doi: 10.1109/TSG.2014.2330624.

[9] J. E. Tate and T. J. Overbye, "Line outage detection using phasor angle measurements," *IEEE Transactions on Power Systems*, 2008, doi: 10.1109/TPWRS.2008.2004826.

[10] L. Che, M. E. Khodayar, and M. Shahidehpour, "Adaptive protection system for microgrids: Protection practices of a functional microgrid system," *IEEE Electrification Magazine*, 2014, doi: 10.1109/MELE.2013.2297031.

[11] V. Vittal, N. Senroy, and G. T. Heydt, "Decision Tree Assisted Controlled Islanding," *IEEE Transactions on Power Systems*, vol. 21, no. 4, pp. 1790–1797, 2006, doi: 10.1109/TPWRS.2006.882470.

[12] J. A. Laghari, H. Mokhlis, M. Karimi, A. H. A. Bakar, and H. Mohamad, "Computational Intelligence based techniques for islanding detection of distributed generation in distribution network: A review," *Energy Convers Manag*, vol. 88, pp. 139–152, Dec. 2014, doi: 10.1016/J.ENCONMAN.2014.08.024.

[13] A. Khamis, Y. Xu, Z. Y. Dong, and R. Zhang, "Faster Detection of Microgrid Islanding Events Using an Adaptive Ensemble Classifier," *IEEE Trans Smart Grid*, vol. 9, no. 3, pp. 1889–1899, 2018, doi: 10.1109/TSG.2016.2601656.

[14] S. Kar, S. R. Samantaray, and M. D. Zadeh, "Data-Mining Model Based Intelligent Differential Microgrid Protection Scheme," *IEEE Syst J*, 2017, doi: 10.1109/JSYST.2014.2380432.

[15] J. P. Singh and N. Bouguila, "Proportional data clustering using K-means algorithm: A comparison of different distances," *2017 IEEE International Conference on Industrial Technology (ICIT)*, pp. 1048–1052, 2017, doi: 10.1109/ICIT.2017.7915506.

[16] H. Shayeghi, B. Sobhani, E. Shahryari, and A. Akbarimajd, "Optimal neuro-fuzzy based islanding detection method for Distributed Generation," *Neurocomputing*, vol. 177, pp. 478–488, Feb. 2016, doi: 10.1016/J.NEUCOM.2015.11.056.

[17] R. Aggarwal and Yonghua Song, "Artificial neural networks in power systems. Part 3: Examples of applications in power systems," *Power Engineering Journal*, vol. 12, no. 6, pp. 279–287, Dec. 1998, doi: 10.1049/pe:19980609.

[18] K. S. Swarup and H. S. Chandrasekharaiah, "Fault detection and diagnosis of power systems using artificial neural networks," in *Proceedings of the First International Forum on Applications of Neural Networks to Power Systems*, IEEE, pp. 102–106. doi: 10.1109/ANN.1991.213505.

[19] K. B. Debnath and M. Mourshed, "Forecasting methods in energy planning models," *Renewable and Sustainable Energy Reviews*, vol. 88, no. September 2017, pp. 297–325, 2018, doi: 10.1016/j.rser.2018.02.002.

[20] L. S. Moulin, A. P. Alves Da Silva, M. A. El-Sharkawi, and R. J. Marks, "Support vector machines for transient stability analysis of large-scale power systems," *IEEE Transactions on Power Systems*, 2004, doi: 10.1109/TPWRS.2004.826018.

[21] M. Nayeripour, M. Hoseintabar, and T. Niknam, "Frequency deviation control by coordination control of FC and double-layer capacitor in an autonomous hybrid renewable energy power generation system," *Renew Energy*, 2011, doi: 10.1016/j.renene.2010.12.012.

[22] W. Grycan, B. Brusilowicz, and M. Kupaj, "Photovoltaic farm impact on parameters of power quality and the current legislation," *Solar Energy*, vol. 165, pp. 189–198, May 2018, doi: 10.1016/J.SOLENER.2018.03.016.

[23] M. Datta, T. Senjyu, A. Yona, T. Funabashi, and C. H. Kim, "A frequency-control approach by photovoltaic generator in a PV-diesel hybrid power system," *IEEE Transactions on Energy Conversion*, 2011, doi: 10.1109/TEC.2010.2089688.

[24] P. Sharma and S. Banerjee, "Enhancing network management frameworks with SDN-like control," *... Management (IM ...*, 2013.

[25] O. N. Faqhruldin, E. F. El-Saadany, and H. H. Zeineldin, "Naive Bayesian islanding detection technique for distributed generation in modern distribution system," in *2012 IEEE Electrical Power and Energy Conference*, IEEE, Oct. 2012, pp. 69–74. doi: 10.1109/EPEC.2012.6474982.

[26] H. He, W. Zhang, and S. Zhang, "A novel ensemble method for credit scoring: Adaption of different imbalance ratios," *Expert Syst Appl*, vol. 98, pp. 105–117, 2018, doi: 10.1016/j.eswa.2018.01.012.

[27] S. R. Mohanty, P. K. Ray, N. Kishor, and B. K. Panigrahi, "Classification of disturbances in hybrid DG system using modular PNN and SVM," *International*

*Journal of Electrical Power and Energy Systems*, vol. 44, no. 1. pp. 764–777, 2013. doi: 10.1016/j.ijepes.2012.08.020.

[28] J. Nordhaug Myhre, K. Øyvind Mikalsen, S. Løkse, and R. Jenssen, “Robust clustering using a kNN mode seeking ensemble,” *Pattern Recognit*, vol. 76, pp. 491–505, Apr. 2018, doi: 10.1016/j.patcog.2017.11.023.

[29] L. D. Analysis, “Introduction to LDA LDA,” *Cancer Lett*, 2005, doi: 10.4169/000298910X485923.

[30] M. A. Hassan, A. Khalil, S. Kaseb, and M. A. Kassem, “Exploring the potential of tree-based ensemble methods in solar radiation modeling,” *Appl Energy*, vol. 203, pp. 897–916, 2017, doi: 10.1016/j.apenergy.2017.06.104.

[31] P. G. V. Sampaio and M. O. A. González, “Photovoltaic solar energy: Conceptual framework,” *Renewable and Sustainable Energy Reviews*, vol. 74. Elsevier Ltd, pp. 590–601, 2017. doi: 10.1016/j.rser.2017.02.081.

[32] L. Yavuz, A. Soran, A. Onen, and S. M. Muyeen, “PSO Supported Ensemble Algorithm for Bad Data Detection Against Intelligent Hacking Algorithm,” *Front Energy Res*, vol. 9, Jul. 2021, doi: 10.3389/fenrg.2021.649460.

[33] L. Yavuz, A. Soran, A. Onen, X. Li, and S. M. Muyeen, “Adaptive Fault Detection Scheme Using an Optimized Self-healing Ensemble Machine Learning Algorithm,” *CSEE Journal of Power and Energy Systems*, vol. 8, no. 4, pp. 1145–1156, Jul. 2022, doi: 10.17775/CSEEJPES.2020.03760.

[34] L. Yavuz, A. Önen, S. M. Muyeen, and I. Kamwa, “Transformation of microgrid to virtual power plant - A comprehensive review,” *IET Generation, Transmission and Distribution*, vol. 13, no. 11, pp. 2077–2087, Jun. 2019, doi: 10.1049/iet-gtd.2018.5649.

[35] T. S. Openai and D. P. K. Openai, “Weight Normalization: A Simple Reparameterization to Accelerate Training of Deep Neural Networks.” [Online]. Available: https://github.com/openai/weightnorm.

[36] S. K. Kumar, “On weight initialization in deep neural networks,” Apr. 2017, [Online]. Available: http://arxiv.org/abs/1704.08863

[37] S. Koturwar and S. Merchant, “Weight Initialization of Deep Neural Networks(DNNs) using Data Statistics,” Oct. 2017, [Online]. Available: http://arxiv.org/abs/1710.10570

[38] S. Masood, M. N. Doja, and P. Chandra, “Analysis of weight initialization methods for gradient descent with momentum,” in *International Conference on Soft Computing Techniques and Implementations, ICSCTI 2015*, Institute of Electrical and Electronics Engineers Inc., Jun. 2016, pp. 131–136. doi: 10.1109/ICSCTI.2015.7489618.

[39] M.-J. Kang and H.-C. Kim, “Comparison of Weight Initialization Techniques for Deep Neural Networks,” *International Journal of Advanced Culture Technology*, vol. 7, pp. 283–288, 2019, doi: 10.17703/IJACT.2019.7.4.283.

[40] C. K. Dewa and Afiahayati, “Suitable CNN Weight Initialization and Activation Function for Javanese Vowels Classification,” in *Procedia Computer Science*, Elsevier B.V., 2018, pp. 124–132. doi: 10.1016/j.procs.2018.10.512.

[41] L. Scabini, B. De Baets, and O. M. Bruno, “Improving Deep Neural Network Random Initialization Through Neuronal Rewiring,” Jul. 2022, [Online]. Available: http://arxiv.org/abs/2207.08148

[42] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science (1979)*, vol. 313, no. 5786, pp. 504–507, Jul. 2006, doi: 10.1126/science.1127647.

[43] F. Saeed, T. Al-Hadhrami, E. Mohammed, and M. Al-Sarem, Eds., *Advances on Smart and Soft Computing*, vol. 1399. in Advances in Intelligent Systems and Computing, vol. 1399. Singapore: Springer Singapore, 2022. doi: 10.1007/978-981-16-5559-3.

[44] K. He, X. Zhang, S. Ren, and J. Sun, “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification,” Feb. 2015, [Online]. Available: http://arxiv.org/abs/1502.01852

[45] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks.” [Online]. Available: http://www.iro.umontreal.

[46] T. Hastie, R. Tibshirani, and J. Friedman, “Springer Series in Statistics The Elements of Statistical Learning Data Mining, Inference, and Prediction.”

[47] “lecun-98b”.

[48] W. Boulila, M. Driss, E. Alshanqiti, M. Al-Sarem, F. Saeed, and M. Krichen, “Weight Initialization Techniques for Deep Learning Algorithms in Remote

Sensing: Recent Trends and Future Perspectives," 2022, pp. 477–484. doi: 10.1007/978-981-16-5559-3_39.

[49] V. V. Priya, P. Natesan, K. Venu, and E. Gothai, "Improving convergence speed of the neural network model using Meta heuristic algorithms for weight initialization," in *2021 International Conference on Computer Communication and Informatics, ICCCI 2021*, Institute of Electrical and Electronics Engineers Inc., Jan. 2021. doi: 10.1109/ICCCI50826.2021.9402415.

[50] U. K. Das *et al.*, "Forecasting of photovoltaic power generation and model optimization: A review," *Renewable and Sustainable Energy Reviews*, vol. 81. Elsevier Ltd, pp. 912–928, Jan. 01, 2018. doi: 10.1016/j.rser.2017.08.017.

[51] A. Ulutas, I. H. Altas, A. Onen, and T. S. Ustun, "Neuro-fuzzy-based model predictive energy management for grid connected microgrids," *Electronics (Switzerland)*, vol. 9, no. 6, Jun. 2020, doi: 10.3390/electronics9060900.

[52] I. Latin and A. Transactions, "Allocation and Sizing of Photovoltaic Systems to Reduce Power Losses and Economic Aspects using a new PSO approach; Allocation and Sizing of Photovoltaic Systems to Reduce Power Losses and Economic Aspects using a new PSO approach," 2022.

[53] S. Roy and H. N. V. Pico, "Transient Stability and Active Protection of Power Systems With Grid-Forming PV Power Plants," *IEEE Transactions on Power Systems*, vol. 38, no. 1, pp. 897–911, Jan. 2023, doi: 10.1109/TPWRS.2022.3165704.

[54] A. Eduardo Nieto, F. Ruiz, D. Patino, and O. Ramirez, "Classification of Electric Faults in Photovoltaic Systems Based on Voltage-Power Curves; Classification of Electric Faults in Photovoltaic Systems Based on Voltage-Power Curves," 2021.

[55] L. Yavuz, A. Önen, S. M. Muyeen, and I. Kamwa, "Transformation of microgrid to virtual power plant - A comprehensive review," *IET Generation, Transmission and Distribution*, vol. 13, no. 11, pp. 2077–2087, Jun. 2019, doi: 10.1049/iet-gtd.2018.5649.

[56] M. Cagatay Kocer *et al.*, "Cloud Induced PV Impact on Voltage Profiles for Real Microgrids; Cloud Induced PV Impact on Voltage Profiles for Real Microgrids," 2018.

[57] J. Kleissl, “Photovoltaic and Solar Forecasting: State of the Art,” 2013. [Online]. Available: https://www.researchgate.net/publication/259717376

[58] M. G. De Giorgi, P. M. Congedo, and M. Malvoni, “Photovoltaic power forecasting using statistical methods: Impact of weather data,” *IET Science, Measurement and Technology*, vol. 8, no. 3, pp. 90–97, 2014, doi: 10.1049/iet-smt.2013.0135.

[59] M. N. Akhter, S. Mekhilef, H. Mokhlis, and N. M. Shah, “Review on forecasting of photovoltaic power generation based on machine learning and metaheuristic techniques,” *IET Renewable Power Generation*, vol. 13, no. 7. Institution of Engineering and Technology, pp. 1009–1023, 2019. doi: 10.1049/iet-rpg.2018.5649.

[60] W. VanDeventer *et al.*, “Short-term PV power forecasting using hybrid GASVM technique,” *Renew Energy*, vol. 140, pp. 367–379, Sep. 2019, doi: 10.1016/j.renene.2019.02.087.

[61] “Australian Solar Energy Forecasting System Final report: project results and lessons learnt-solar-energy-forecasting-system-asefs-phase-1.”

[62] S. Hensel, M. B. Marinov, M. Koch, and D. Arnaudov, “Evaluation of deep learning-based neural network methods for cloud detection and segmentation,” *Energies (Basel)*, vol. 14, no. 19, Oct. 2021, doi: 10.3390/en14196156.

[63] W. Kong, Y. Jia, Z. Y. Dong, K. Meng, and S. Chai, “Hybrid approaches based on deep whole-sky-image learning to photovoltaic generation forecasting,” *Appl Energy*, vol. 280, Dec. 2020, doi: 10.1016/j.apenergy.2020.115875.

[64] A. Moncada, W. Richardson, and R. Vega-Avila, “Deep learning to forecast solar irradiance using a six-month UTSA SkyImager dataset,” *Energies (Basel)*, vol. 11, no. 8, Aug. 2018, doi: 10.3390/en11081988.

[65] V. Le Guen and N. Thome, “A Deep Physical Model for Solar Irradiance Forecasting with Fisheye Images.”

[66] Z. Zhen *et al.*, “Deep Learning Based Surface Irradiance Mapping Model for Solar PV Power Forecasting Using Sky Image,” *IEEE Trans Ind Appl*, vol. 56, no. 4, pp. 3385–3396, Jul. 2020, doi: 10.1109/TIA.2020.2984617.

[67] G. Li, S. Xie, B. Wang, J. Xin, Y. Li, and S. Du, “Photovoltaic Power Forecasting with a Hybrid Deep Learning Approach,” *IEEE Access*, vol. 8, pp. 175871–175880, 2020, doi: 10.1109/ACCESS.2020.3025860.

[68] L. Yavuz, A. Önen, S. M. Muyeen, and I. Kamwa, “Transformation of microgrid to virtual power plant - A comprehensive review,” *IET Generation, Transmission and Distribution*, vol. 13, no. 11, pp. 2077–2087, Jun. 2019, doi: 10.1049/iet-gtd.2018.5649.

[69] A. Niccolai, E. Ogliari, A. Nespoli, R. Zich, and V. Vanetti, “Very Short-Term Forecast: Different Classification Methods of the Whole Sky Camera Images for Sudden PV Power Variations Detection,” *Energies (Basel)*, vol. 15, no. 24, Dec. 2022, doi: 10.3390/en15249433.

[70] H. J. Lu and G. W. Chang, “A Hybrid Approach for Day-Ahead Forecast of PV Power Generation,” in *IFAC-PapersOnLine*, Elsevier B.V., Jan. 2018, pp. 634–638. doi: 10.1016/j.ifacol.2018.11.774.

[71] Z. Li, H. Shen, Q. Cheng, Y. Liu, S. You, and Z. He, “Deep learning based cloud detection for medium and high resolution remote sensing images of different sensors.” [Online]. Available: http://sendimage.whu.edu.cn/en/mscff/

# CURRICULUM VITAE

| | |
|---|---|
| 2012 – 2016 | B.Sc., Physics, Erciyes University, Kayseri, TURKEY |
| 2014 – 2017 | B.Sc., Electrical Electronics Engineering, Erciyes University, Kayseri, TURKEY |
| 2016 – 2018 | M.Sc., Nuclear Physics, Erciyes University, Kayseri, TURKEY |

2018 – Present Research Assistant, Eelectrical Electronics Engineering Department,
Abdullah Gul University, Kayseri, TURKEY

SELECTED PUBLICATIONS AND PRESENTATIONS

**J1)** Levent Yavuz; Ahmet Onen; S.M. Muyeen; Innocent Kamwa, "Transformation of microgrid to virtual power plant – a comprehensive review", IET Generation, Transmission & Distribution, Volume 13, Issue 11, p. 1994 – 2005, 04 June 2019.

**J2)** L. Yavuz, A. Soran, A. Onen, X. Li and S. M. Muyeen, "Adaptive Fault Detection Scheme Using an Optimized Self-healing Ensemble Machine Learning Algorithm," in CSEE Journal of Power and Energy Systems, vol. 8, no. 4, pp. 1145-1156, July 2022.

**J3)** Levent Yavuz, Ahmet Onen, "Investigation on Cloud Thickness and Movement into PV Forecasting in region of the Kayseri, Türkiye", is under review at IEEE Transaction on Power Systems, 2023.

**J4)** Levent Yavuz, Ahmet Onen, "Statistical-Based Heuristic Vectorised Learning Method for PV Forecasting", is under review at IEEE Transaction on Power Systems, 2023.

**J5)** T. S. Ustun, S. M. S. Hussain, L. Yavuz, and A. Onen, "Artificial Intelligence Based Intrusion Detection System for IEC 61850 Sampled Values under Symmetric and Asymmetric Faults," *IEEE Access*, vol. 9, pp. 56486–56495, 2021, doi: 10.1109/ACCESS.2021.3071141.

**J6)** B. Kolukisa *et al.*, "Coronary Artery Disease Diagnosis Using Optimized Adaptive Ensemble Machine Learning Algorithm," *Int J Biosci Biochem Bioinforma*, vol. 10, no. 1, pp. 58–65, 2020, doi: 10.17706/ijbbb.2020.10.1.58-65.

**J7)** L. Yavuz, A. Soran, A. Onen, and S. M. Muyeen, "PSO Supported Ensemble Algorithm for Bad Data Detection Against Intelligent Hacking Algorithm," *Front Energy Res*, vol. 9, Jul. 2021, doi: 10.3389/fenrg.2021.649460.

**J8)** L. Yavuz, A. Soran, A. Onen, and S. M. Muyeen, "Machine learning algorithms against hacking attack and detection success comparison," in *2020 2nd International Conference on Smart Power and Internet Energy Systems, SPIES 2020*, Institute of

Electrical and Electronics Engineers Inc., Sep. 2020, pp. 258–262. doi: 10.1109/SPIES48661.2020.9243033.