Photovoltaic Power Pattern Clustering Based on Conventional and Swarm Clustering Methods

by

Amr Abdullah Munshi

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Computer Engineering

Department of Electrical and Computer Engineering

University of Alberta

Abstract

Among renewable energy resources, solar energy is promising and has recently become an area of interest in research. Photovoltaic (PV) systems have the capability of converting solar energy into electrical power. The advances in PV technology, such as the reliability and the continuous reduction in capital costs, motivate the integration of PV systems into the electrical grid. The power output of PV systems is mainly influenced by the level of irradiation and ambient temperature. This leads to operational problems and instability in the power output generated from PV systems. Accordingly, the integration of these systems requires extensive study and simulations of lengthy historical data with sub-hourly time steps. However, dealing with such data is time consuming and computationally expensive. Photovoltaic power pattern (PVPP) clustering is fundamental in providing enhanced knowledge on the impacts of integrating PV systems into the electrical grid without extensive analysis and simulations. Therefore, this research aims to develop solutions that can reduce the burden of extensive studies and simulations related to the integration of PV systems into the electrical grid.

This research investigates a set of clustering methods from different clustering categories to determine the optimum number of clusters and to produce cluster representatives for PVPP data. Furthermore, the introduction of bio-inspired swarm optimization methods, such as the Ant Colony and Bat methods in clustering power patterns is presented. For the purpose of clustering and achieving efficient cluster representatives, six clustering algorithms from five different clustering categories are involved: K-means from partitional clustering, Hierarchical Ward's minimum variance (WMV) from agglomerative clustering, Fuzzy C-means (FCM) from fuzzy

clustering, self-organizing maps (SOM) from neural network based algorithms, and Ant Colony and Bat from bio-inspired swarm optimization methods. In order to evaluate the clustering methods in a comprehensive manner, the following nine internal validity indices were employed: Davies Bouldin (DBI), Dunn, Silhouette (SI), Bayesian information criterion (BIC), Xie-Beni (XB), mean square error (J), clustering dispersion indicator (CDI), mean index adequacy (MIA), and ratio of within-cluster sum-of-squares to between-cluster variation (WCBCR). The clustering results show that swarm clustering methods are comparable to conventional methods. Moreover, the Bat method was the most efficient and outperformed the other clustering methods. Therefore, five Bat algorithms with various objective functions: Bat based on Davies Bouldin Index (Bat DBI), Bat based on Dunn index (Bat Dunn), Bat based on clustering dispersion indicator (Bat CDI), Bat based on mean index adequacy (Bat MIA), and Bat based on within-cluster sum-of-squares to between-cluster variation (Bat WCBCR) are proposed to enhance the clustering results. The clustering results on two data sets show that the bio-inspired swarm clustering algorithm Bat based on WCBCR, as an objective function, produces significantly highly separated and well-compacted clusters that can be utilized in PV system simulations. In order to test the efficiency of the produced PVPP representatives in PV system simulations, a short-term PV power prediction model is presented. The results of the prediction model verify the efficiency of the PVPP clustering methodology in PV system studies.

Acknowledgement

Thank you God for helping me achieve this work.

I would like to express my gratitude to my supervisor, Dr. Yasser Mohamed, for his continuous guidance throughout the duration of my MSc studies. His valuable discussions, insightful suggestions, and constant feedback were always helpful and inspiring. Also, his continuous support and encouragement were my greatest motive to aim for the best.

I would like to express my thanks to the examiners committee for their valued time and interests in my thesis.

I would like to show my deepest gratitude and respect to my family, especially my parents, to whom I owe all the success in my life. No words can express my gratitude, but I pray that God will bless them and reward them.

Many thanks to my wife, without whom I could have never been able to achieve this work. Her patience and encouragement were always a source of strength for me.

Many thanks to my daughter and son, who accepted trading our playing time together with research time. I will make it up to them.

Table of Contents

List of Tables

List of Figures

List of Acronyms

AC: Alternating Current

ANN: Artificial Neural Networks

BA: Bat Algorithm

Bat CDI: Bat based on Clustering Dispersion Indicator

Bat DBI: Bat based on Davies Bouldin Index

Bat Dunn: Bat based on Dunn index

Bat MIA: Bat based on Mean Index Adequacy

Bat WCBCR: Bat based on Within-cluster sum-of-squares to between-cluster Variation

BIC: Bayesian Information Criterion

CDI: Clustering Dispersion Indicator

DBI: Davies Bouldin Index

DC: Direct Current

FCM: Fuzzy C-means

J: Mean Square Error Function

MAE: Mean Absolute Error

MAPE: Mean Absolute Percentage Error

MIA: Mean Index Adequacy

MSE: Mean Square Error

PCA: Principal Component Analysis

PC: Principal Component

PSO: Particle Swarm Optimization

PV: Photovoltaic

PVPP: Photovoltaic power pattern

RBF: Radial Basis Function

RMSE: Root Mean Square Error

SI: Silhouette Index

SOM: Self-Organizing Maps

SOFM: Self-Organizing Feature Maps

UPGMA: Unweighted Pair-group Method using Arithmetic Averages

WCBCR: Within-cluster sum-of-squares to between-cluster Variation

WMV: Ward's Minimum Variance

XB: Xie-Beni

Chapter 1

Introduction

1.1 Motivation

As the demand for electrical power is increasing rapidly due to the growth of the global population and industrialization, electrical systems need to increase generation. Currently, most power is generated from conventional power resources, such as fossil fuels. However, there are many financial, environmental, and availability issues associated with the increasing consumption of conventional power resources. In order to overcome many of these issues, renewable energy resources can participate in producing power. The potential of using renewable energy resources as an alternative for power generation is being widely studied. Among renewable energy resources, solar energy is promising and has recently become an area of research interest. Photovoltaic (PV) systems have the capability of converting solar energy into electrical energy. The advances in PV technology, such as the reliability and the continuous reduction in capital costs, motivate the integration of PV systems into the electrical grid.

Generally, the power output of PV systems is influenced by the level of solar irradiation and the ambient temperature [1]. This leads to operational problems and instability in the output power generated from PV systems. Accordingly, planning for the integration of these systems requires extensive study and simulations of lengthy historical data of irradiance and ambient temperature with sub-hourly time steps. Also, predicting the power output of PV systems requires the output power data with sub-hourly time steps. However, dealing with such data is time consuming and computationally expensive. For this purpose, the main focus of this thesis is to develop solutions that can reduce the burden of extensive studies and simulations related to integrating PV systems into the electrical grid.

1.2 Thesis Objectives

This research aims to develop solutions that can reduce the burden of extensive studies and simulations related to integrating PV systems into the electrical grid. Therefore, we investigate the most appropriate clustering method for establishing the PV power pattern (PVPP) grouping. We chose at least one representative algorithm from various clustering categories: K-means from partitional clustering, Hierarchical Wards' minimum variance (WMV) from agglomerative clustering, Fuzzy C-means (FCM) from fuzzy clustering, self-organizing maps (SOM) from neural network based algorithms, and Ant Colony and Bat algorithms from particle swarm optimization methods. In addition, we propose five Bat clustering algorithms with different objective functions, and test their efficiency in clustering PVPPs and presenting the optimal number of clusters.

The main objectives are summarized as follows:

- Comparing between different clustering methods.
- Introducing particle swarm optimization-based clustering algorithms in clustering PVPP.
- Introducing a proper parameter calibration process, such as the learning rate of SOM and the wavelength frequency in the Bat algorithm.
- Using an extended set of validity indices to evaluate the performance of the clustering methods.
- Reducing the dimensionality of daily PVPPs during the clustering process using principal component analysis (PCA).
- Proposing five Bat clustering algorithms based on different objective functions.
- Finding the optimum number of clusters for PVPPs and presenting a representative power pattern from each cluster that can be used in the simulations and studies of the integration of PV systems into the electrical grid.

1.3 Thesis Outline

The remainder of the thesis is organized as follows:

Chapter 2 presents an introduction to clustering and discusses the utilized clustering methods: K-means, Hierarchical WMV, FCM, SOM, Ant Colony and Bat algorithms. Also, nine validity indices, including DBI, Dunn, SI, BIC, XB, J, CDI, MIA, and WCBCR that evaluate the clustering algorithms' performance are introduced. The dimensionality reduction technique, principal component analysis (PCA), is also illustrated.

Chapter 3 introduces the layout of the methodology used to establish the PVPP grouping. The details of each step are also presented. The application of the clustering methods and validity indices are applied on a real data set and discussed. A comparison of the clustering results is presented, and the method to detect the optimum number of clusters is discussed in detail in this chapter.

Chapter 4 introduces the five proposed Bat clustering algorithms based on different objective functions. The application of these algorithms on two dimensionally reduced data sets of PVPPs is presented. Also, a detailed comparison between these algorithms' results is presented.

Chapter 5 presents a model for short-term predictions of PV power. The approach this model uses is a dedicated formulation in order to test the efficiency of the PVPP cluster representatives obtained from the previous chapters.

Chapter 6 presents the summary and conclusions of the thesis.

Chapter 2

Background

2.1 Introduction

This chapter introduces the background of the core topics in this thesis. The work related to clustering power patterns and the general preliminary notes are presented. Six clustering algorithms from five different clustering categories are discussed: K-means from partitional clustering, Hierarchical WMV from agglomerative clustering, FCM from fuzzy clustering, SOM from neural network based algorithms, and Ant Colony and Bat algorithm from particle swarm optimization methods are presented in detail. Figure 2.1 illustrates the taxonomy of the clustering algorithms. The nine validity indices utilized to evaluate the clustering results, (DBI, Dunn, SI, BIC, XB, J, CDI, MIA and WCBCR) are discussed. Finally, the feature generation method PCA is presented.

2.2 Introduction to Clustering

Clustering is an unsupervised learning procedure that has been studied in various contexts and disciplines. The aim is to combine data points into groups (clusters) based on similarity and

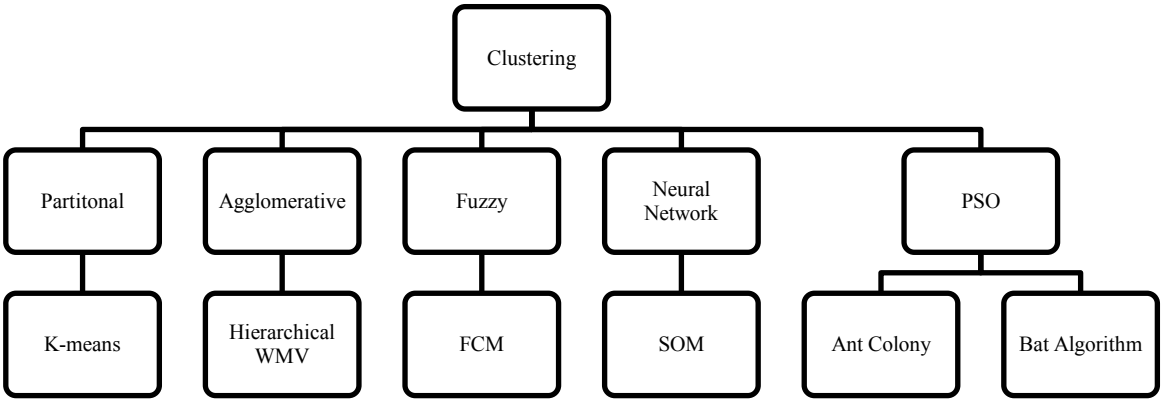

Figure 2.1: Taxonomy of the clustering algorithms that will be discussed in this chapter.

dissimilarity criteria, such that the similarity of data points within clusters is maximized and the similarity of data points from different clusters is minimized. Accordingly, the data set is represented by fewer cluster representatives that may lose fine details but achieve simplification and scalability. Each cluster contains a cluster representative (centroid), which is usually the weighted average of the data points within that cluster. There are numerous clustering methods that can be used for clustering; most of them are parametric methods that require a pre-specified number of clusters. The assignment of data points to clusters in general can be either hard (crisp) or soft (fuzzy) [2]. Crisp clustering algorithms assign each data point to exactly one cluster; whereas fuzzy clustering algorithms may assign a data point to more than one cluster with different membership degrees (usually between 0 and 1). Membership degrees close to zero imply minimal similarity between the data point and that cluster, while membership degrees close to unity imply a high degree of similarity between the data point and that cluster [3]. The sum of membership degrees for each data point must be unity. Fuzzy clustering can be a powerful tool that can deal with uncertainty and ambiguity hidden within data.

2.3 Related Work

An intensive research effort has been devoted to cluster time-series power patterns and obtaining representatives for these clusters during the last few years. Models based on clustering techniques were used to group electrical load patterns of customers in order to assist tariff formation [4]–[11], short-term forecasting [12], and demand response programs to support management decisions [13]–[15]. Also, power load clustering has been used for the classification of load profiles for ship electric consumers [16] and for estimating the power load of warships [17]. In [18], aggregate modeling of wind farms has been proposed based on the wind farm's layout and the clustering of wind speed patterns. A method to improve the management decisions of wind farms was also proposed in [19] by applying a clustering method on wind power loads.

Research interest in PVPP clustering for analyzing the power output fluctuation effects on integrating PV systems into the electrical grid [20] and for determining the optimal location and size of PV plants [21] has recently increased. In the clustering process, [20] adapted three

clustering methods: K-means, hierarchical, and a hybrid of K-means and hierarchical whereas, [21] used K-medoids and Fuzzy C-means. In [22] a Radial Basis Function (RBF) network model to predict short-term PV generation was proposed by clustering historical time-series PV power data, then constructing a prediction model at each cluster so that the prediction is based on data similarity. [23] proposed a predicting model by clustering historical data and using the weather forecast. The results of [22] and [23] showed that the prediction results depend significantly on the accuracy of the clustered data. Thus, developing clustering algorithms that produce efficient partitioning of PV power data is of practical interest. In addition, the potential of PV in becoming a major power resource world-wide [24] motivates the investigation of applying various clustering techniques to investigate the most appropriate technique for clustering PVPPs. Bio-inspired algorithms have been efficient in solving many optimization related problems. However, integrating such algorithms with data mining algorithms is still at an early stage and has not gained much attention. In [11] electrical load pattern clustering using the Ant Colony algorithm was able to detect an abnormal load pattern whereas the K-means method included this abnormal pattern in a cluster with other normal patterns. For this purpose, Ant Colony clustering and Bat clustering algorithms are included to investigate their performance on clustering PVPPs.

2.4 Preliminary Definitions

This section illustrates some general definitions and notations of the clustering algorithms and validity indices used in the context of PVPP clustering. The initial data are a set of N daily PVPP referring to a specified period of time (i.e., the fall season for the past few years). Each daily PVPP contains d time-series observations (features). The row vector $x_n = [x_{n1}, \ldots, x_{nd}]$ represents the nth PVPP for $x = 1,\ldots, N$. The PVPP data set is represented by the matrix $\boldsymbol{X} = [x_{1,}\ldots, x_N]$. The clustering process creates a partitioning of the N PVPPs into K clusters with non-overlapping PVPPs through an iterative process. Each cluster is represented by a centroid $C_k = [C_{k1},\ldots, C_{kd}]$, for $k = 1,\ldots, K$. The set of centroids is represented by the matrix $\boldsymbol{C} = [C_1,\ldots, C_K]$. The proximity measure quantifies the closeness between elements (e.g., data points and centroids). The proximity is measured using a similarity (distance) measure. In this thesis, the utilized proximity measure is the Euclidean distance given by:

$$D_{ij} = \sqrt{\sum_{l=1}^{d} \left(x_{il} - x_{jl} \right)^2} . \tag{2.1}$$

2.5 K-means Clustering Algorithm

The K-means algorithm is one of the most popular algorithms used in clustering [25]. It groups a set of *N* input data points into *K* clusters using an iterative procedure. The number of *K* clusters is a user specified parameter that depends on the desired number of clusters based on the application. The average of all data points in a cluster is a representative data point called a centroid. The main goal of K-means is to minimize the sum of the square error over all *K* clusters. Equation (2.2) indicates the objective function *J* where *K* is the required number of clusters, x_i is the *i*th data point, C_k is the centroid of the *k*th cluster, and *N* is the number of data points.

$$J = \sum_{k=1}^{K} \sum_{x_i \in c_k}^{N} || x_i - C_k ||^2 \tag{2.2}$$

The classical K-means makes adequate geometric and statistical sense for numerical data sets but does not function with data sets that contain categorical data points. The formation of clusters in the classical K-means is significantly affected by outliers. The K-means algorithm starts by randomly choosing *K* data points as centroids. The different random initialization of centroids can lead to different cluster formation, even when applied on the same data set. A poor initialization of clusters can lead to insufficient clustering results [26]. The distance between each data point and each centroid is calculated using a proximity measure, such as the Euclidean distance. Each data point is then assigned to the closest centroid. After that, the centroid of each cluster is updated based on the mean of data points in that cluster. The assignment of data points to the closest cluster and the updating of the centroids are repeated until no data points change their cluster and the centroids remain the same. Figure 2.2 presents the flowchart of K-means.
The classical K-means clustering can be summarized by the following steps [27]:

1. Initialize *K* data points randomly or with some prior knowledge $\boldsymbol{C} = [C_1, C_2, \ldots, C_K]$.

2. Calculate the distances between each data point x and centroid C and assign each data point to the nearest centroid.

$$x_i \in C_w, \quad \text{if } \| x_i - C_w \| < \| x_i - C_j \| \quad \text{for } i = 1,\ldots,N, j \neq w, \text{ and } j = 1,\ldots,K \tag{2.3}$$

3. Recalculate the centroid for each cluster,

$$C_k = \frac{1}{N_k} \sum_{x \in C_k} x \tag{2.4}$$

where N_k is the number of data points in C_k.

Repeat steps 2 and 3; terminate when there is no change for each cluster.

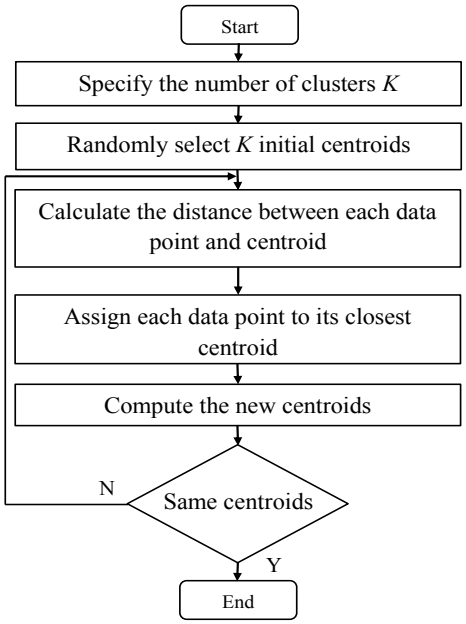

Figure 2.2: Flowchart of the general K-means algorithm.

2.6 Introduction to Hierarchical Clustering

Hierarchical algorithms can either be agglomerative (bottom-up) or divisive (top-down) [2]. An agglomerative algorithm starts by considering each data point as an individual cluster where similar clusters are merged in successive steps. Conversely, divisive hierarchical algorithms start with all data points in one cluster and then split successively until each data point is an individual cluster. The merge and split decisions are based on a similarity metric. However, different similarity metrics may yield different clustering results even when the same data set is used [28]. The resulting decomposition is represented in a convenient tree-like structure called a dendogram

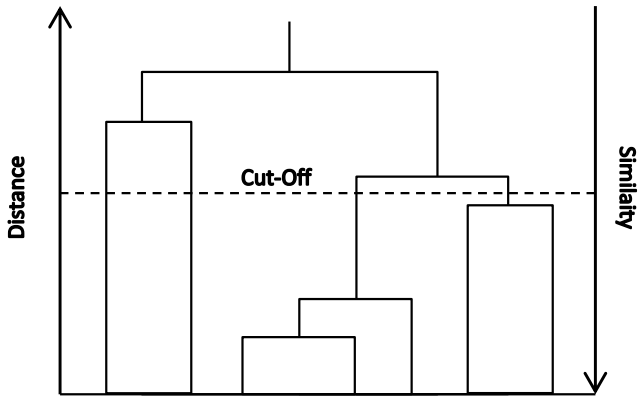

Figure 2.3: Dendogram of a hierarchical clustering result.

(Figure 2.3). In the dendogram each level of the hierarchy represents a particular grouping of data points into disjoint clusters. It is a user task to decide which level represents the desired clustering formation and how many clusters are desired in the sense that data points within each cluster are sufficiently more similar to each other than to those in other clusters [29].

2.6.1 Agglomerative Hierarchical Clustering

Agglomerative hierarchical clustering is the most commonly used approach in hierarchical clustering. It requires calculating the proximity between clusters to decide which ones to group together. There are various agglomerative approaches that have different distance definitions

between clusters (Figure 2.4). The general flowchart of the agglomerative hierarchical clustering algorithm is shown in Figure 2.5. The next subsections define some agglomerative approaches and Table 2.1 presents their formulas.

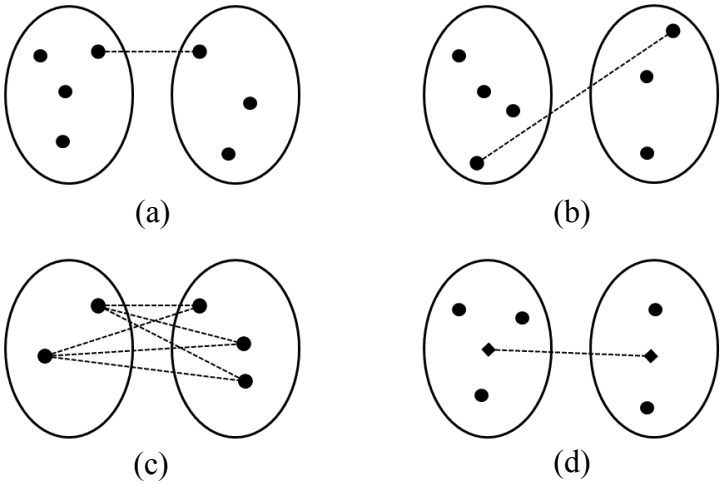

Figure 2.4: Agglomerative hierarchical clustering procedures (a) single linkage (b) complete linkage (c) average linkage (d) centroid linkage.

2.6.1.1 Single Linkage

The single linkage (nearest-neighbour) is based on the minimum distance between two data points in two different clusters. The merging of clusters is based on the most similar data points from each cluster. Single linkage tends to form clusters that may lead to heterogeneous data points clustered together [30]. This procedure is sensitive to outliers, as a new data point can extremely alter the hierarchical clustering structure [31].

2.6.1.2 Complete Linkage

Complete linkage (farthest-neighbour) is based on the maximum distance between two data points in two different clusters. The cluster similarity is based on the most dissimilar data points from each cluster. It tends to form compact sphere-like clusters [30]. This procedure finds

compact clusters with small diameters; however, some data points in a certain cluster may be much closer to other clusters than the other data points in its cluster [31].

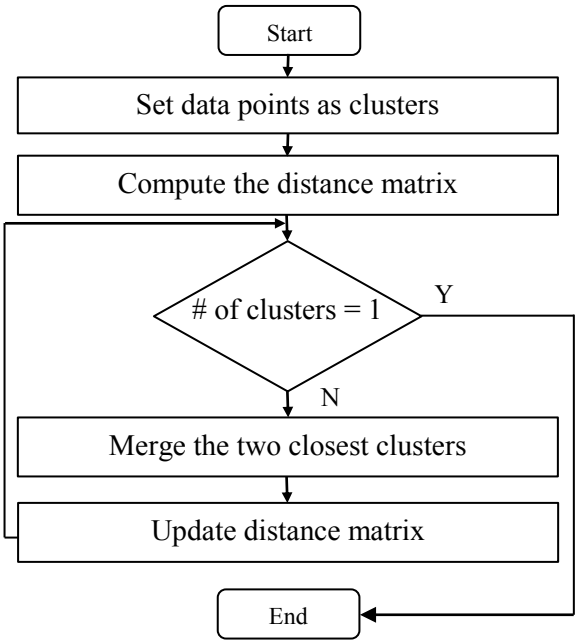

Figure 2.5: Flowchart of the general agglomerative hierarchical algorithm.

2.6.1.3 Average Linkage

The average linkage, UPGMA (unweighted pair-group method using arithmetic averages), is a compromise of single and complete linkages. It is based on the average distance between all the pairs of data points of two clusters. In other words, it calculates the minimum and maximum of all the pairwise distances between the data points of two clusters to average them. Consequently, the resulting clusters tend to have almost equal within-cluster variability [30].

2.6.1.4 Centroid Linkage

In this procedure, the centroid, which is an existing representative data point, of each cluster is determined first. The merging is based on the distance between two centroids.

2.6.1.5 Ward's Minimum Variance (WMV) Linkage

Another commonly used procedure in agglomerative clustering is the WMV method. This procedure differs from the other mentioned procedures, as it utilizes the variance to evaluate the distances between clusters. It merges clusters if such merging increases the overall within-cluster variance to the smallest possible degree.

The steps to perform a general agglomerative hierarchical algorithm are:

1. Assign each data point to a separate cluster.
2. Evaluate all pair-wise distances between clusters.
3. Construct a distance matrix using a distance metric.
4. Look for the pair of clusters with the shortest distance.
5. Merge the pair of clusters and remove them from the distance matrix.
6. Calculate all distances from this new cluster to all other clusters using a distance linkage and update the distance matrix.
7. Repeat from step 4 until all the clusters are grouped into one cluster.

Table 2.1: Formulas of agglomerative hierarchical clustering approaches.

Linkage	**Formula**
Single linkage	$D(X,Y) = \min_{x \in X, y \in Y} d(x,y)$
Complete linkage	$D(X,Y) = \max_{x \in X, y \in Y} d(x,y)$
Average linkage	$D(X,Y) = \frac{1}{\mid X \mid . \mid Y \mid} \sum_{x \in X} \sum_{y \in Y} d(x,y)$
Centroid linkage	$D(X,Y) = \| \overline{X} - \overline{Y} \|^2$
WMV	$D(X,Y) = \frac{\| \overline{X} - \overline{Y} \|^2}{\frac{1}{N_x} + \frac{1}{N_y}}$

2.7 Fuzzy C-means (FCM) Clustering Algorithm

The FCM algorithm is one of the oldest and most ubiquitous fuzzy clustering algorithms developed by Dunn in 1973 and improved by Bezdek in 1981. It allows each data point to belong to more than one cluster with different membership degrees. This method behaves in a similar fashion to the K-means algorithm but each data point has a membership degree with respect to each cluster. Various fuzzy clustering algorithms have been developed based on the optimization and modification of the FCM algorithm [32]. The main objective of the FCM is based on the minimization of the following objective function:

$$J_m(U,C) = \sum_{i=1}^{N}\sum_{j=1}^{C}\left(\mu_{ij}\right)^m d^2\left(x_i, C_j\right) \tag{2.5}$$

where $U = [\mu_{ij}]_{n\times k}$ is a matrix with degrees of memberships of each data point in each cluster and $\mu_{ij} \in [0, 1]$. This implies that the sum of the membership values for each data point on the K clusters must be equal to 1. The function $d(x_i, C_j)$ is the distance between x_i (the ith data point) and C_j (the centroid of the jth cluster), and $m \in [1,\infty)$ is the fuzziness parameter. The selection of these parameters is not an easy task and must be made by experience or by trial-and-error. Figure 2.6 illustrates the basic FCM flowchart.

The steps to perform FCM algorithm are [33]:

1- Select values for the number of clusters K, fuzziness parameter m, a small positive threshold value ε, and a random set of centroids $\boldsymbol{C}$.

2- If $t = 0$, calculate, or if $t > 0$, update the membership matrix U:

$$U_{ij}^{(t+1)} = \frac{1}{\sum_{k=1}^{K}\left(\frac{d(x_i, C_j)}{d(x_i, C_k)}\right)^{\left(\frac{2}{m-1}\right)}} \tag{2.6}$$

where $i = 1,\ldots, N$ and $j = 1,\ldots, K$

3- Update the fuzzy centers ($\boldsymbol{C}$) by:

$$C_j^{(t+1)} = \frac{\sum_{i=1}^{N} \mu_{ij}^m . x_i}{\sum_{i=1}^{N} \mu_{ij}^m}, \text{ for } j = 1, \ldots, K \tag{2.7}$$

4- Repeat steps 2 and 3 until the objective function J_m converges to a local minimum. This means that $\|U^{t+1} - U^t\| < \varepsilon$.

Various FCM clustering algorithms have appeared as a result of the utilization of different distance metrics and fuzziness control [27]. It should be noted that the fuzziness parameter *m* is a positive value greater than 1 and as it increases the fuzziness increases.

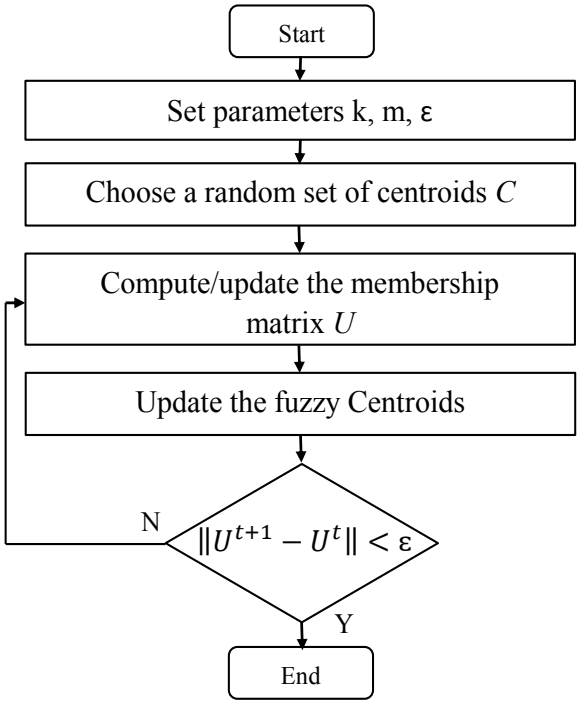

Figure 2.6: Flowchart of FCM.

2.8 Introduction to Artificial Neural Networks

Artificial neural networks (ANNs) are information processing paradigms that attempt to model biological nervous systems, such as the brain. The main element of this paradigm is the novel structure of the information processing system. An ANN is composed of a large number of highly interconnected processing elements called “neurones” that work together to solve specific problems. An ANN mimics the way people learn by example. They have the ability to learn from data in a supervised or unsupervised fashion. The ANN can be set for a specific application, such as regression, pattern recognition, or data classification through a learning process. This type of learning requires certain adjustments to the synaptic connections that exist between the neurones, as in biological systems.

2.8.1 Self-Organizing Maps (SOM)

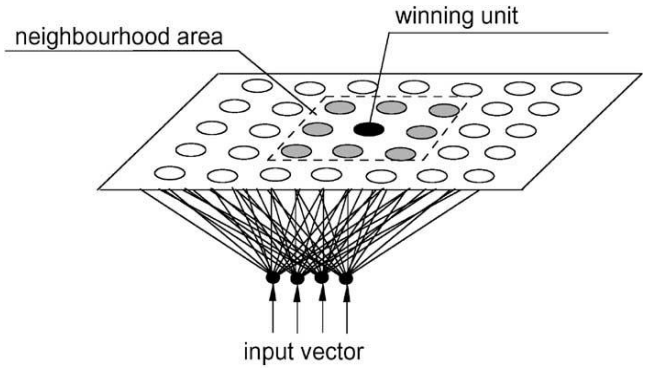

Figure 2.7: Structure of the SOM with hexagonal distance function. Retrieved from [7].

Kohonen self-organizing maps (SOM) were developed by Tuevo Kohonen in 1982 [34]. A self-organizing map (SOM) or self-organizing feature map (SOFM) is a topological unsupervised neural network that projects high-dimensional input patterns into a reduced dimensional space that can usually be visualized in a one-dimensional or two-dimensional lattice structure. Each unit in the network (lattice) is known as a neuron, and adjacent neurons are connected to each

other. The input patterns are fully connected to network neurons through adaptable weights. These weights are updated as input patterns are projected into the network, and then assigned to the best matching unit (winning neuron) based on a competition function. As more inputs (data points) are presented into the network, each neuron closest to the input vector adjusts its weight vector toward the input vectors. Figure 2.7 illustrates the main structure of the SOM. The neurons in the layer of the SOM are arranged originally in physical positions according to a certain topology function. MATLAB offers functions that can arrange the neurons in a grid, hexagonal, or random topology structure. Distances between neurons are calculated from their positions with a distance function. There are four distance functions in MATLAB: dist, boxdist, linkdist, and mandist. For the research presented in this thesis, the gridtop topology and dist distance functions have been used after a few trials and errors. A detailed discussion about SOM is given in Hagan et al. [35] and Kohonen [36]. The basic SOM procedure can be summarized in the following steps [27]:

1- Random initialization of prototype (weight) vectors $m_j^{(0)}, j = 1,\ldots, K$.

2- Project an input pattern (data point) x into the network and choose the winning neuron, J_w, based on the minimum distance to x:

$$J_w = \arg_j \ \min\left\{\| x - m_j \|\right\} \tag{2.8}$$

3- Update the prototype (weight) vectors,

$$m_j(t+1) = m_j(t) + h_{cj}(t)\left[x - m_j(t)\right], \tag{2.9}$$

where $h_{cj}(t)$ is the neighborhood kernel function centered on the winning neuron,

$$h_{cj}(t) = \eta(t) exp\left(\frac{-\| r_c - r_j \|^2}{2\sigma^2(t)}\right), \tag{2.10}$$

where r_c and r_j are the positions of the corresponding neuron on the network, $\sigma(t)$ is the monotonically decreasing kernel width, and $\eta(t)$ is the monotonically decreasing learning rate defined by:

$$\eta(t) = \eta_{\circ}\, exp\left(\frac{-t}{T_{\eta_{\circ}}}\right) > \eta_{min}\ , \tag{2.11}$$

where $\eta_{\circ}$, $\eta_{\min}$ and $T_{\eta_{\circ}}$ are the initial learning rate, the minimum learning rate, and the time parameter, respectively.

4- Repeat steps 2 and 3 until the maximum number of epochs is reached or no change of neuron position more than a positive number is observed.

It should be noted that in step 3 the prototype vectors are updated and moved closer to the input vector. The winning neuron's weights are altered proportionally to the learning rate, whereas the weights for the neighbouring neurons are updated in inverse proportion to their distance [36]. The SOM performance is significantly sensitive to the initialization of the prototype vector weights. Accordingly, the mapping could generate suboptimal partitions if the weights are not chosen properly. The topology and distance functions can be determined based on trial and error. The flowchart of SOM in its basic form is shown in Figure 2.8.

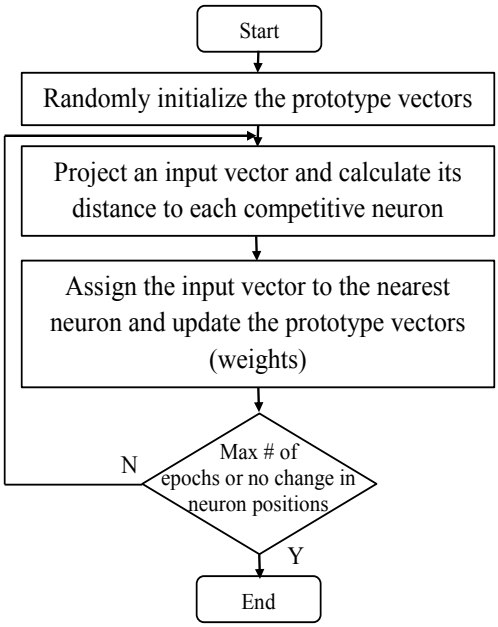

Figure 2.8: Basic flowchart of SOM.

2.9 Introduction to Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) methods are recently developed population based stochastic techniques that mimic swarm behaviour. It is considered to be a swarm intelligence technique [38]. Although each candidate (particle) member of the swarm moves in its own way, the swarm as a whole collaborates together in order to achieve a common optimization objective. Each particle stores its local best solution and updates it when a better solution is found. The behavior of particles is adjusted according to the overall best particle's solution (global best solution).

2.9.1 Ant Colony Clustering Algorithm

Ant Colony clustering is an intelligent swarm-based approach that mimics the behavior of real ants to find the shortest path between a food source and their nest. The ants communicate and exchange information about the paths by means of pheromone trails. As more ants trace a certain path and deposit their pheromone, the more attractive this path becomes and is followed by other ants. Consequently, this collaborative behaviour leads to the establishment of the shortest route path [39].

This Ant Colony clustering approach uses a dedicated formulation with respect to other applications of Ant Colony clustering [11], [40], [41]. It can be divided into three main stages: initialization, first iteration, and successive iterations.

Initalization: In the initializtion stage, the number of clusters K and the number of ants A are defined. Then the initial set of centroids $\boldsymbol{C}^{(0)}$ are randomly chosen from the data set. An initial $N \times K$ phermone matrix $\phi^{(0)}$ is constructed by computing the distances between each data point and each centroid. The resulting null distances are replaced by a relatively small value ε to avoid division by zero:

$$r_{ik}^{(0)} = \max\left\{d(x_i, C_k), \varepsilon\right\} \tag{2.12}$$

Then, auxiliary variables based on the squared inverse of distances in $\phi^{(0)}$ are calculated:

$$\varphi'^{(0)}_{ik} = 1/\left(r^{(0)}_{ik}\right)^2 \tag{2.13}$$

The components $\varphi'^{(0)}_{ik}$ of the pheromone matrix are normalized to avoid the continuous growth of pheromone components in the iterative stage. This normalizing is accomplished by dividing each auxiliary variable by the sum of auxiliary variables occurring in the same corresponding row:

$$\varphi^{(0)}_{ik} = \varphi'^{(0)}_{ik} / \sum_{k=1}^{K} \varphi'^{(0)}_{ik} \tag{2.14}$$

First iteration: For the number of ants $a = 1,\ldots, A$, each ant generates a solution path vector $S_a^{(1)}$ based on a probablistic criterion using the pheromone matrix components $\varphi^{(0)}_{ik}$. The $\boldsymbol{S}^{(1)}$ matrix is an $N \times A$ matrix that contains the solution (clusters) to which each data point is assigned for ant a. The generated solutions are determined by using the biased roulette wheel selection criterion with the probability of choice proportional to row values of the pheromone matrix $\phi^{(0)}$. The pseudo code to implement the biased roulette wheel is as follows [42]:

1: Let $i = 1$, where i denotes the row index of the normalized pheromone matrix;
2: $sum = \varphi^{(m)}_{ik}$, m is the iteration number;
3: Generate $rand \sim U(0, 1)$;
4: **while** $sum < rand$ **do;**
5: $i = i + 1$, (i.e., advance to the next index);
6: $sum = sum + \varphi^{(m)}_{ik}$;
7: **end while;**
8: Return i as the selected cluster for $S_{an}^{(m+1)}$;

The set of centroids $\boldsymbol{C}_a^{(1)}$ is now obtained for each $S_a^{(1)}$ vector by averaging the data points assigned to a specified cluster. Hence, A clustering solution vectors and centroid sets are obtained. Each clustering solution is evaluated by a fitness function based on the sum of square errors:

$$\psi_a^{(m)} = \sum_{k=1}^{K} \sum_{x_i \in c_{ak}}^{N} \| x_i - C_{ak} \|^2 , \quad \text{for } a = 1,\ldots, A , \tag{2.15}$$

where m is the iteration number.

In this fitness function, lower values indicate better clustering solutions. Thus, the set of $S_a^{(m)}$ and $\boldsymbol{C}_a^{(m)}$ leading to the lowest fitness values are considered to be the best sets, defined as $\tilde{\boldsymbol{S}}_a^{(m)}$ and $\tilde{\boldsymbol{C}}_a^{(m)}$, respectively, and these replace the initial ones. Finally, the pheromone matrix is updated to $\phi^{(1)}$ by:

$$r_{ik}^{(m)} = \max\left\{ d(x_i, \tilde{\boldsymbol{C}}_{ak}^{(m)}), \varepsilon \right\} . \tag{2.16}$$

Differently from the initialization stage, the auxiliary variables $\varphi'^{(1)}_{ik}$ are calculated by adding a pheromone reinforcement term to (2.13)

$$\varphi'^{(m)}_{ik} = \varphi'^{(m-1)}_{ik} + 1 / \left(r_{ik}^{(m)} \right)^2 . \tag{2.17}$$

The components of $\varphi'^{(m)}_{ik}$ of the pheromone matrix are then normalized with m = 1 to avoid continuous growth of pheromone components in the iterative stage:

$$\varphi_{ik}^{(m)} = \varphi'^{(m)}_{ik} / \sum_{k=1}^{K} \varphi'^{(m)}_{ik} \tag{2.18}$$

Successive iterations: To avoid losing the best solution sets, the solution and centroid set for the first ant (a=1) is set to equal $\tilde{\boldsymbol{S}}_a^{(m)}$ and $\tilde{\boldsymbol{C}}_a^{(m)}$. For the successive ants a = 2,…, A, at each iteration m, solution vectors $S_a^{(m)}$ are generated based on the roulette wheel criterion as indicated in the first iteration.

The following operations are the same as the ones mentioned in the first iteration, with the obtaining of the set of clusters $\boldsymbol{C}_a^{(m)}$ and evaluating each ant's solution then obtaining the best

solution vector $\tilde{\boldsymbol{S}}_a^{(m)}$ and set of centroids $\tilde{\boldsymbol{C}}_a^{(m)}$. At the end of each successive iteration, the pheromone matrix $\phi^{(m)}$ is updated by following (2.16) through (2.18).

Stop criterion: An effective criterion in heuristic methods is to stop when there is no noticable improvement in the fitness function after a specified number of successive iterations. For the purpose of preventing excessive computation time, a user defined maximum number of iteration is adopted here.

Final clustering results: The assignment of data points to clusters is achieved by taking the index of the highest value in each row of the pheromone matrix ϕ. Hence, the final centroids can be obtained by averaging the data points assigned to each cluster.

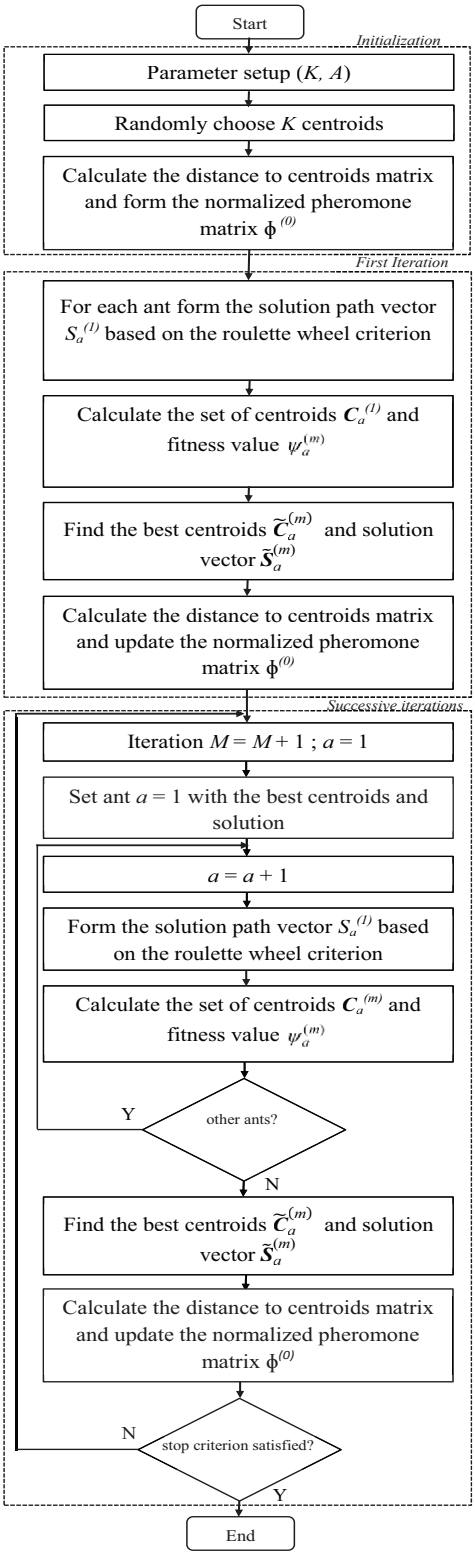

Figure 2.9: Flowchart of Ant Colony clustering algorithm.

2.9.2 Bat Clustering Algorithm

The Bat algorithm (BA) is a swarm intelligence algorithm that was first introduced in [38] for solution optimization problems. It has been found to be efficient and has expanded significantly. Recently, clustering has been addressed by applying BA concepts [43].

Bats are fascinating animals that have an advanced capability of echolocation. A famous type of bats that uses echolocation extensively is microbats. Microbats emit a loud and short pulse of sound (echolocation) and wait a fraction of time for the echo to return back to their ears. Accordingly, bats can determine how far they are from the surrounding objects. Moreover, bats have the capability of distinguishing between an obstacle and prey, which allows them to search for prey even in darkness. When searching for prey, the loudness increases and decreases when approaching towards prey. The main idea of the BA is to mimic bat behavior when tracking prey. In order to model this algorithm, the idealization proposed by [38] can be followed as:

1- All bats use echolocation to sense distance, and they can differentiate between food and prey and background barriers in some magical way;
2- Bats fly randomly with velocity v_i at position y_i with a fixed frequency f_{min}, varying wavelength λ and loudness A_0 to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission $r \in [0,1]$, depending on the proximity of their target;
3- Although the loudness A can vary in many ways, we assume that the loudness varies from a large (positive) A_0 to a minimum constant value A_{min}.

At each iteration each bat's position y_i is associated with a fitness indicator expressing its performance. The search is intensified by a local random walk (exploitation) and the selection of the best current position (solution) continues until a certain stop criteria is met.

The Bat clustering algorithm [43] can be divided into three main stages, as follows:

Initialization: In this stage, the number of clusters K and the number of bats, $b = 1,\ldots, B,$ are defined. Each bat (b) is then assigned an emission rate r, a frequency value $f_b \in [f_{min}, f_{max}]$, loudness value $A_b \in [A_0, A_{min}]$, and a random solution vector (S_b) showing to which cluster each data point is assigned. Accordingly, an $N \times B$ matrix $\boldsymbol{S}$ is formed where N is the number of data

points, and each column vector represents the *b*th bat solution vector. The initial set of centroids $\boldsymbol{C}_b^{(0)}$ are calculated from each S_b vector by averaging the data points assigned to a specified cluster. The set of centroids $\boldsymbol{C}$ is also referred to the bats' positions *y*.

Exploitation: Each solution vector S_b is evaluated by a fitness function based on the sum of square errors:

$$\psi_b^{(t)} = \sum_{k=1}^{K} \sum_{x_i \in c_{bk}}^{N} \| x_i - C_{bk} \|^2 , \text{ for } b = 1, \ldots, B \tag{2.19}$$

In the previous fitness function, lower values indicate better clustering results. The lowest fitness value is defined as $\tilde{\boldsymbol{\psi}}$, and the corresponding S_b and y_b are considered to be the best sets, defined as $\tilde{\boldsymbol{S}}$ and $\tilde{\boldsymbol{y}}$, respectively. New *B* solution vectors are generated by adjusting the frequency, updating the velocity, and updating the positions of the bats:

$$f_b = f_{min} + \left(f_{min} - f_{max} \right) \beta , \tag{2.20}$$

$$v_b^{(t)} = v_b^{(t-1)} + \left[y_b^{(t)} - \tilde{\boldsymbol{y}} \right] f_b , \tag{2.21}$$

$$y_b^{(t)} = y_b^{(t-1)} + v_b^{(t)} , \tag{2.22}$$

where β denotes a random value within the interval [0, 1].

A new random number $\beta 2$ between [0, 1] is generated, and if it is greater than the pulse rate *r*, a new local search solution is generated around $y_b^{(t)}$,

$$y_b^{(t)} = \tilde{\boldsymbol{y}} + \varepsilon . \mathcal{A} , \tag{2.23}$$

where ε is a small value that attempts to direct and strengthen the random walk, and $\mathcal{A}$ is a randomly generated normal distribution vector of the same size as *y*. Now the distance between each data point and its position is computed, and each data point is assigned to the lowest distance solution (i.e., each data point is assigned to the nearest centroid). Then the

corresponding fitness is computed $\psi_b^{(t)}$. Finally, a random number $\beta 3$ between [0, 1] is generated, and if it is less than the loudness A and the computed fitness $\psi_b^{(t)}$ is less than $\psi_b^{(t\text{-}1)}$, it accepts the new position for that bat, increases r, and decreases A.

Updating clustering results: In this stage if one of the generated positions improves the best fitness function, then $\tilde{\boldsymbol{S}}$ and $\tilde{\boldsymbol{y}}$ are updated.

Stop criterion: This process continues until a user-predefined maximum number of iterations M is reached. Then the final set of centroids $\boldsymbol{C}$ can be obtained from $\tilde{\boldsymbol{y}}$.

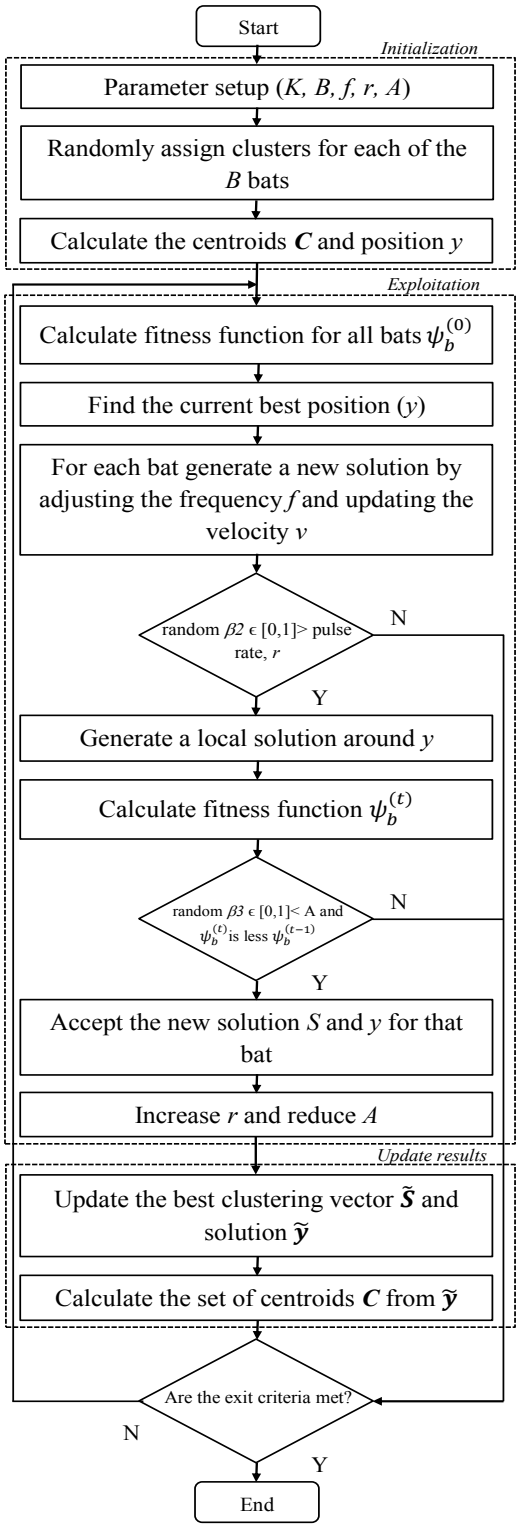

Figure 2.10: Flowchart of Bat clustering algorithm.

2.10 Validity Indices

Various clustering algorithms have been developed. They behave differently depending on the features of the data set and the assumptions in defining clusters. In addition, for the same data set, the clustering algorithm can present different results when different parameters are tweaked [44]. Therefore, the produced clustering results require evaluation to find the partitioning that best fits the underlying data set. Clustering validity indices are usually utilized to measure the quality of the clustering results. There are three approaches of validation indices: external indices, internal indices, and relative indices [44]. External indices measure the agreement between two partitions. The first partition is the a priori, a pre-specified clustering structure of the data set, such as labels, and the second partition results from the clustering procedure. Internal indices are used to measure the goodness of the clustering structure without any external information. The relative validity indices use external or internal indices to compare different clustering algorithms with each another. Internal validity indices are appropriate for evaluating the partitioning of clustering algorithms, as it is an unsupervised task where pre-specified knowledge about the classification of the data set is unavailable. There are various internal validity indices for crisp and fuzzy clustering algorithms that have different aspects in measuring the optimization of clustering.

The approach of a validity index in assessing clustering consists of running a clustering algorithm several times for different numbers of partitions (clusters) and selecting the clustering result that optimizes the validity index. The evaluation of the optimal clustering partition is based on two parameters: compactness and separateness. The compactness is defined by the closeness of the members (data points) of each cluster. The separateness is defined by the distance of separation of the clusters, which should be as separated from each other as possible. It should be noted that various distance matrices are utilized to measure the compactness and separateness. Thus, the main goal of a cluster validity index is to identify a compact and separate partition of clusters that presents the optimal clustering quality [45]. The following are compactness and separation formulas:

$$Compactness = \frac{1}{N}\sum_{k=1}^{K}\frac{1}{N_k}\sum_{x_i \in C_k} \| x_i - C_k \|^2 \quad (2.24)$$

,

$$Separation = \frac{1}{K}\sum_{1\le q<l}^{K} \| C_l - C_q \|^2 \quad (2.25)$$

.

In the next subsections nine validity indices based on properly defined metrics and indicators are introduced: the DBI, Dunn, SI, BIC, XB, J, CDI, MIA and WCBCR.

2.10.1 Davies-Bouldin Index

The DBI [46] identifies clusters with high compactness and low separateness. It is a function of the ratio of the sum of the within cluster scatter to the between cluster separations defined as:

$$DBI = \frac{1}{K}\sum_{i=1}^{K}\max_{j, i\neq j}\frac{\hat{d}(\Omega_i)+\hat{d}(\Omega_j)}{d\left(C_i, C_j\right)} \quad (2.26)$$

where $\hat{d}(\Omega_i)$ and $\hat{d}(\Omega_j)$ are the average distances between all data points in cluster *i* and *j* to their respective cluster centroids and $d(C_i, C_j)$ is the distance between the centroids of clusters *i* and *j*, respectively. A smaller Davies-Bouldin value indicates compact clusters and large distances between cluster centroids.

2.10.2 Dunn Index

The Dunn's index identifies a clustering scheme as a ratio between the minimal inter-cluster distances to the maximal intra-cluster distance. This can be achieved by the following formula [47]:

$$Dunn = \min_{1\le i\le C}\left\{\min_{\substack{1\le j\le C \\ j\neq i}}\left\{\frac{d\left(C_i, C_j\right)}{\max_{1\le k\le C}\left(d\left(C_k\right)\right)}\right\}\right\} \quad (2.27)$$

where $d(C_k)$ is the intra-cluster distance of cluster k. It can be observed that large values indicate the existence of compact and well-separated clusters. Thus, the number of clusters chosen for a particular algorithm is the one that generates the largest value as an optimum number of clusters.

2.10.3 Silhouette Index (SI)

The SI [48] calculates the silhouette width for each data point, average silhouette width for each cluster, and the average silhouette width for the entire data set.

For a given cluster C_k, this approach assigns a quality measure to each data point in C_k, known as the silhouette width. The silhouette width is a confidence indicator on the membership of the ith data point in cluster C_k and is defined by the following formula:

$$s(i) = \frac{b(i) - a(i)}{\max\left(a(i), b(i)\right)} \quad , \tag{2.28}$$

where $a(i)$ is the average distance between the ith data point and all data points in the same cluster of C_k, and $b(i)$ is the minimum average distance between the ith data point and all data points not included in the same cluster. The $s(i)$ value will vary between $-1 \leq s(i) \leq 1$. A value close to 1 indicates that the data point i is classified to the right cluster, whereas a value close to -1 indicates the misclassification of that data point. A value close to 0 indicates that a data point contained within one cluster is at an equal distance away from another cluster and could be contained within either cluster. The average silhouette width that represents the heterogeneity of a given cluster C_k is calculated by:

$$S_j = \frac{1}{n}\sum_{i=1}^{n} s(i) \tag{2.29}$$

where n is the number of data points in $s(i)$. The overall global silhouette width denoted by GS is defined by:

$$GS = \frac{1}{K}\sum_{i=1}^{K} S_j \quad . \tag{2.30}$$

In order to choose the optimal number of clusters using the SI index, the clustering that presents the maximal *GS* is chosen.

2.10.4 Bayesian Information Criterion (BIC)

The BIC [49] is a criterion for model selection to fit into a given data set. The formula for BIC is based on [50], as follows:

$$BIC = L(\theta) - \frac{1}{2} K \log N, \tag{2.31}$$

where $L(\theta)$ is the log-likelihood function of data point x. Assuming a Gaussian distribution of the data, the maximal likelihood estimate for the variance of the kth cluster is given by:

$$\sum_i = \frac{1}{N_k - K} \sum_{j=1}^{N_k} \| x_j - C_k \|^2 \tag{2.32}$$

The final BIC formula can be written as:

$$BIC = \left(\sum_{k=1}^{K} N_k \log \frac{N_k}{N} - \frac{N_k d}{2} \log(2\pi) - \frac{N_k}{2} \log \sum_k \frac{N_k - K}{2} \right) - \frac{K \log N}{2} \tag{2.33}$$

where d is the dimension of the patterns. The maximal BIC value indicates strong evidence for the correct number of clusters.

2.10.5 Xie-Beni (XB) index

The XB validation index [51] involves the U matrix (from FCM) and data set to evaluate the clustering of fuzzy algorithms. However, it can be applied to validate the clustering of crisp algorithms. The XB index is defined as the ratio of the total variation to the minimum separation of the clusters. It can be calculated by [52]:

$$XB = \frac{1}{N}\left(\frac{\sum_{j=1}^{K}\sum_{i=1}^{N} U_{ij} \parallel C_j - x_i \parallel}{\min\limits_{i \neq j} \parallel C_i - C_j \parallel} \right). \tag{2.34}$$

Small values of XB define compact and well-separated clusters.

2.10.6 Mean Square Error or Error Function (J)

The J function [4] expresses the distance of each data point from its cluster centroid with the same weight values:

$$J = \frac{1}{N}\sum_{i=1}^{N} d\left(x_i, C_j\right), \quad \text{for } j = 1, \ldots, K \tag{2.35}$$

The J function is a decreasing function regarding the number of clusters. When it reaches a knee point, the optimum number of clusters can be found.

2.10.7 Clustering Dispersion Indicator (CDI)

The CDI [4] is the ratio of the mean intra-set distance between data points in the same cluster ($\hat{d}(\Omega_k)$) and the intra-set distance between the cluster centroids ($\hat{d}(C)$):

$$CDI = \frac{1}{\hat{d}\left(C\right)}\sqrt{\sum_{i=1}^{K} \hat{d}\left(\Omega_k\right)}. \tag{2.36}$$

Lower CDI values indicate better clustering results. However, an increasing number of clusters decreases the CDI value. A knee point can define the optimum number of clusters.

2.10.8 Mean Index Adequacy (MIA)

The MIA [4] is the average of distances between each data point assigned to the same cluster (Ω_k) and its centroid:

$$MIA = \sqrt{\frac{1}{K}\sum_{i=1}^{K} d\left(\Omega_k, C_k\right)} \quad (2.37)$$

The choice of the number of clusters is similar to that in CDI.

2.10.9 Ratio of within-cluster sum-of-squares to between-cluster variation (WCBCR)

The WCBCR [53] depends on the sum of squared distances between each data point and its centroid as well as the distances between centroids:

$$WCBCR = \frac{\sum_{k=1}^{K}\sum_{x_i \in C_k} d\left(x_i, C_k\right)}{\sum_{1 \le q < p}^{K} d\left(C_p, C_q\right)}. \quad (2.38)$$

The choice of the number of clusters is similar to CDI and MIA.

2.11 Principle Component Analysis (PCA)

Principle Component Analysis (PCA), invented by Karl Pearson in 1901 [54], is a technique usually used to reduce the dimensionality of large data sets while retaining as much variation as in the original data. The PCA is based on an orthogonal linear transformation that transforms the data into a new coordinate or feature space. The components of the new space are uncorrelated, in a sense that the first coordinate (first principle component) lies in the direction of the greatest variance of the original data set. The second principle component lies in the direction of the second greatest variation in the original data set and so on. In order to reduce the dimensionality of the data, the higher principle components that retain the least variance of the original data set are neglected [54], [55].

The PCA is one of the most popular techniques for feature generation. It has been used extensively in feature generation in order to reduce the dimensionality of features in data sets. The steps of applying PCA can be summarized as follows [56]:

1- Consider an $N \times d$ data set, where N is the number of data points, and d is the number of features (pattern) for each data point.
2- Calculate the mean value of each feature (column).
3- Subtract the mean value from each feature. The result is an $N \times d$ matrix $\boldsymbol{Z}$ with zero mean features.
4- Calculate the covariance matrix $\boldsymbol{Cov}$ using equation (2.39).

$$Cov_{N\times N} = \frac{1}{d}(Z^{T}_{N\times d} \times Z_{N\times d}) \tag{2.39}$$

5- Calculate the eigenvectors and eigenvalues $\boldsymbol{\lambda}$.
6- Sort the columns of the eigenvector matrix in descending order according to the eigenvalues. The result is an $N \times N$ matrix $\boldsymbol{Y}$ with principle components (PC) in the highest order of retained variation.
7- Select the number of PCs $\boldsymbol{L}$ from matrix $\boldsymbol{Y}$. Hence, an $\boldsymbol{N} \times \boldsymbol{L}$ matrix $\boldsymbol{V}$ is formed.

The PCA in this thesis was done using the MATLAB function “princomp”. The number of PCs was chosen based on retaining at least 95% of the variation from any given data set.

Chapter 3

PVPP Clustering Methodology and Application[1,2]

3.1 Introduction

This chapter presents the methodology to cluster PVPPs using clustering methods and validity indices discussed in the previous chapter. The main idea of the presented method is to use a great amount of historical data in an efficient and intelligent manner, while preserving the temporal information. The steps for preparing the data (including data acquisition, cleaning, and conversion) are highlighted in this chapter. Six clustering algorithms from various clustering categories are tested to investigate the appropriate method for establishing the PVPP grouping process. The criteria used to choose the best clustering algorithm and the optimum number of clusters is discussed. The application of the methodology is applied on a real data set and the results are discussed.

3.2 General Methodology

The clustering of PVPPs is achieved by applying a pattern recognition methodology on historical time series data. This historical data consists of irradiance and ambient temperature at a certain site for the past couple of years with an appropriate time resolution. The time resolution should be able to capture the short-term fluctuations in the irradiance and ambient temperature. The data is to be converted to daily PVPPs. The next step is to group together the PVPPs that have similar features and choose a representative for each group. The representative PVPPs can be used instead of the original data set. The general layout of the method is presented in Figure 3.1 and the basic steps are discussed in the following sub-sections.

---

[1]Part of Chapter 3 of this thesis has been published as: A. A. A. Munshi, and Y. A.-R. I. Mohamed, "Photovoltaic power pattern grouping based on bat bio-inspired clustering," *Proc. 40th PVSC*, pp.1461-1466, 8-13 June 2014.

[2]Chapter 3 of this thesis has been submitted as: A. A. Munshi, and Y. A.-R. I. Mohamed, "Photovoltaic power pattern clustering based on conventional and swarm clustering methods", submitted to *IEEE Systems Journal*, Sep. 2014.

3.2.1 Data Pre-processing

Input: Historical irradiance and ambient temperature for a certain location with proper time steps.

Output: Noise-suppressed daily time series of irradiance and ambient temperature.

Description: The irradiance and ambient temperature data for the past few years are divided into segments where each segment represents a day. The daily time series patterns of irradiance and ambient temperature are examined for normality.

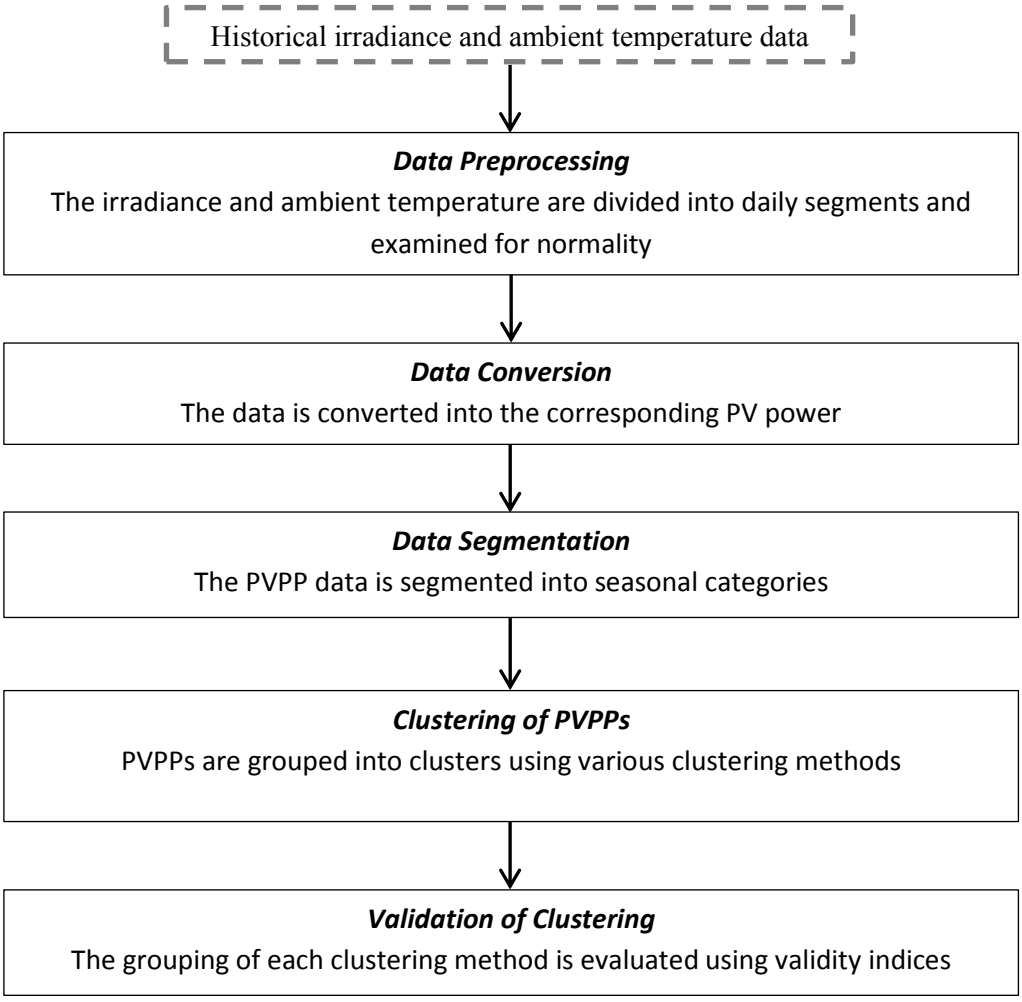

Figure 3.1: Layout of the methodology.

3.2.2 Data Conversion

Input: Noise-suppressed daily time series irradiance and ambient temperature.

Output: Time series of the corresponding AC power of the PV system (PVPP).

Description: The AC power output time series of the PV system can be estimated from the irradiance and ambient temperature time series data by using an appropriate model.

3.2.3 Data Segmentation

Input: Time series of PVPPs.

Output: Categorical segments of daily PVPPs.

Description: Each year can be divided into categorical segments (e.g., seasonal categories). The similar categories for each year are segmented together.

3.2.4 Clustering of PVPPs

Input: Categorical segments of PVPPs.

Output: Different groupings of PVPPs from each clustering method.

Description: Each category of data is clustered by each clustering method. The results are different groupings of data, clustered according to the perspective of the applied clustering method.

3.2.5 Validation of Clustering

Input: Grouping results of each clustering method.

Output: Validity index values.

Description: The results of the clustering methods are evaluated by properly defined metrics and indicators (validity indices). An evaluation value with respect to the utilized validity index is an indicator of how well the clustering method grouped the data.

3.3 Data Preprocessing

The irradiance and ambient temperature time series data are obtained from a weather station for a certain site with an appropriate time resolution. This step can be divided into two sub-steps: 1) the irradiance and ambient temperature time series data are divided into segments where each segment represents a day. The daily patterns of irradiance and ambient temperature are row vectors and each column is an observation of irradiance/temperature at a certain time step. The common periods when there is no irradiance are removed to reduce the dimensionality of the data. 2) The daily patterns of irradiance and ambient temperature are examined for normality in order to modify or delete values that are observed to be incorrect (noise suppression).

3.3.1 Irradiance Time-resolution and Periods Required

The time resolution of the irradiance data should be able to capture the fluctuations of the irradiance as it affects the power output of PV systems. In addition, time resolution plays an important role in the accuracy of the results.

In order to study the performance and impacts of PV systems, the time steps of irradiance data should be high enough to capture the sub-hourly fluctuation of irradiance [57]. Moreover, irradiance data with high sub-hourly time steps will have higher auto-correlation coefficients values than irradiance data with a one-hour time resolution [58]. Figure 3.2 compares between fluctuations in irradiance of a day for one hour, 30 minutes, and ten minutes. It can be observed that the one-hour time step is not able to capture fluctuations in irradiance during the day. The 30-minute time step can capture fluctuations, but much of the temporal information is lost. On the other hand, the ten-minute time step can capture fluctuation with more temporal information and provide more accurate results. The choice of time step is also determined by the availability of the data.

Periods when no irradiance is available are removed, as no PV power can be generated during those periods. Figure 3.3 plots the daily irradiance values for one year of the on-hand data; it can be observed that periods from 8:00 PM to 4:00 AM have no irradiance consistently throughout the year and can be removed in order to reduce the dimensionality of the data. The removal of

periods when no irradiance exists can vary from one data set to another depending on the location of the obtained data.

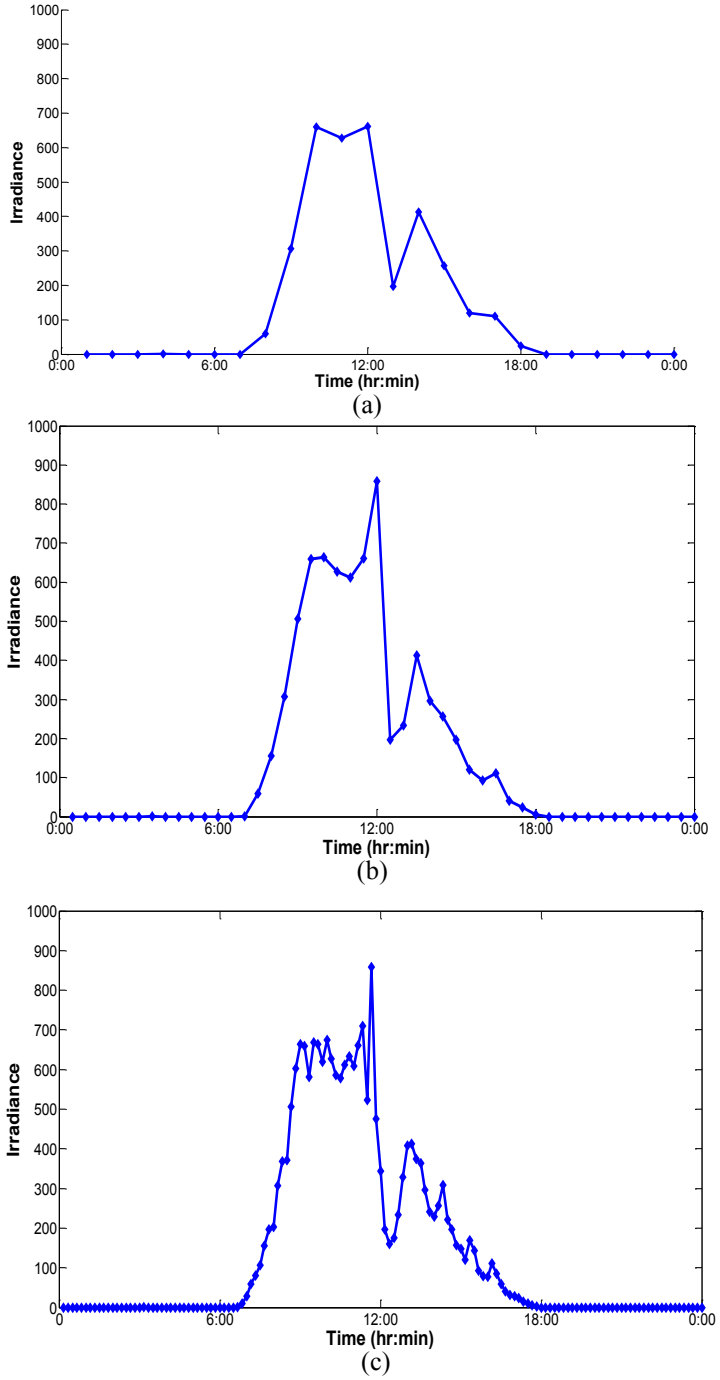

Figure 3.2: Captured fluctuations in irradiance for various time-steps: a) one-hour b) 30-minutes c) ten-minutes.

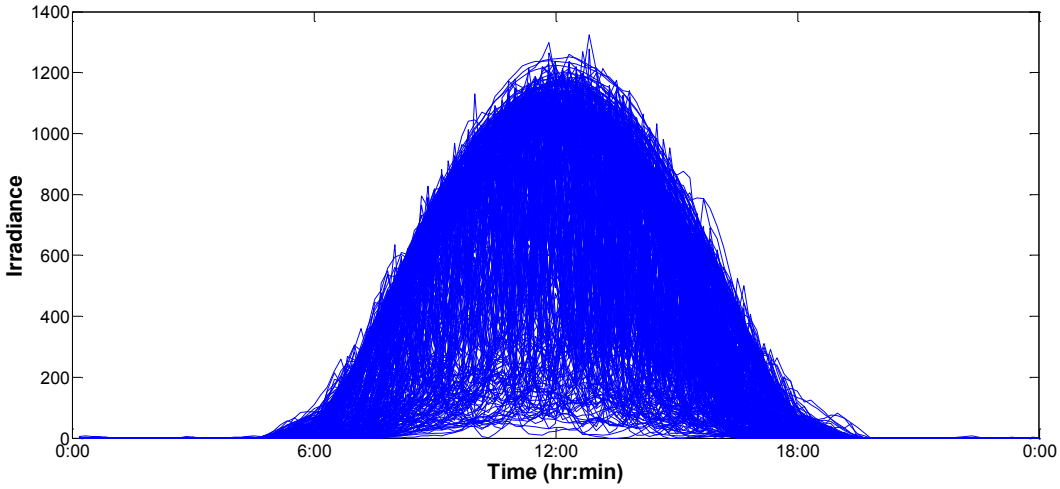

Figure 3.3: Daily irradiance patterns for one year.

3.3.2 Noise Suppression

The obtained data from weather stations are likely to include outliers due to errors in measurements. The days where irradiance values greater than 1360W/m^2 [59] and night-time periods where irradiance is observed are removed. Also, days where ambient temperature values are unavailable or abnormal are removed.

3.4 Data Conversion

The irradiance and corresponding ambient temperature data are used in a PV model to calculate the maximum DC power output of the PV system. In order to calculate the AC power generated from the PV system, it should be noted that the DC power generated from a PV array is affected by several factors, such as the power loss due to dust, the power loss due to module parameter mismatch, and the power loss due to the DC current ripple caused by the converter [60].

The AC power output time series of the PV system can be estimated from the irradiance and ambient temperature time series data by the following two steps [60]:

1. The calculation of the DC power (P_{dc}) output generated from the PV system using a suitable PV model and using the PV module data sheet is as follows:

$$T_c = T_{amb} + \left(\frac{NOCT - 20}{0.8} \right). S \quad , \tag{3.1}$$

$$P_{dc} = P_{max} \left(S / 1000 \right) \left[1 - 0.005 \left(T_c - 25 \right) \right] \quad , \tag{3.2}$$

where T_c is the cell temperature, T_{amb} is the ambient temperature, S is the irradiance, $NOCT$ is the operating cell temperature at Standard Test Conditions (STC: S=1000W/m^2, T_{amb}=25ºC), and P_{max} is the maximum rated power.

2. The calculation of the AC power (P_{ac}) output generated from the PV system using the manufacturer's efficiency curve is as follows:

$$P_{ac} = P_{dc} \times \eta_{mismatch} \times \eta_{dirt} \times \eta_{inverter} , \tag{3.3}$$

where $\eta_{mismatch}$, $\eta_{dirt,}$ and $\eta_{inverter}$ are array to mismatched modules, dirt loss, and inverter efficiency, respectively.

The AC power output of the PV system for each day at each observation constructs the daily PVPP data.

3.5 Data Segmentation

After the conversion step, the daily PVPPs of the PV system are obtained. The data is first segmented into years. In this case, 365 PVPPs are available for each year. Each year can be divided into seasonal segments (i.e., fall, winter, spring, and summer). The similar seasonal category segments are then grouped together in the same data set. This leads to groupings of data that have close profiles and can be clustered more efficiently.

3.6 Data Clustering

The main objective of the data-clustering step is to group together PVPP with close patterns in the same cluster and presenting a cluster representative. Each clustering algorithm produces

different clusters and thus, different cluster representatives. Therefore, it is necessary to apply several clustering algorithms in order to choose the most appropriate algorithm for PVPP data clustering.

For each segment of PVPP time series data, a combination of conventional clustering algorithms and bio-inspired swarm optimization clustering algorithms are applied. The conventional clustering algorithms are chosen because of their extensive utilization in literature. In addition, each one represents a clustering category, except for Ant Colony and Bat, which both fall in the same category. K-means is a representative of partitional clustering. Hierarchical clustering is a representative of agglomerative clustering. The FCM is a representative of fuzzy clustering methods. The SOM is a representative of neural network based algorithms. Ant Colony and Bat are representatives of bio-inspired optimization methods. Those clustering algorithms are used to assign the PVPP into clusters, so that PVPPs in the same cluster are more similar to each other than those in other clusters. From each cluster, a representative PVPP (centroid) can be obtained. Thus, the set of centroids can be used to represent the whole data set.

3.7 Validation of Clustering

The validation of clustering results obtained by clustering methods is a fundamental part of the clustering process. Although clustering validation is a difficult task and lacks theoretical background, examining the compactness and separation of clusters can provide an indication of how well the data are partitioned [61]. The clustering results are evaluated in order to determine the optimum number of clusters and the most efficient clustering algorithm for PVPPs. Various validity indices based on different metrics and indicators are utilized to investigate which ones are able to present adequate information about the optimum number of clusters. The success of choosing the number of clusters is expressed by detecting the index's best value at the knee point.

From the previous step, each clustering algorithm produces different clusters and different centroids. Therefore, it is essential to evaluate clustering results in order to choose the most suitable algorithm for clustering the PVPPs. The comparison among the clustering algorithms' results is held by the utilization of clustering validity indices based on properly defined metrics

and indicators. In order to evaluate the clustering results of each algorithm in a comprehensive manner, nine internal validity indices are employed: DBI, Dunn, SI, BIC, XB, J, CDI, MIA and WCBCR.

3.8 Simulation Results

The methodology was applied on data concerning three consecutive past years (2010-2012) with ten-minute time-steps of irradiation and ambient temperature from the Solar Radiation Research Laboratory [62]. The location of the obtained data has a latitude of 39.74ºN and a longitude of 105.18ºW. The irradiance data with this high time resolution (ten minutes) can lead to better accuracy due to the autocorrelation coefficients that will have higher positive values as compared to those obtained for data with lower time resolutions [58]. Thus, the 10-minute time resolution will result in 144 observations per day. The data were then examined for normality to remove abnormal and error recorded observations in irradiance or ambient temperature values. For example, two days of the year of 2011 were removed as the irradiance values were "-99999" for the whole day. In order to reduce the dimensionality of the data set, the periods when the irradiance is not available are removed. Thus, each day (data point) is limited to the period between 4:00 AM and 8:00 PM, which corresponds to 96 ten-minute time steps per day. The resulting data from the data pre-processing step for the three years were 1089 ten-minute daily irradiance and ambient temperature data points. The data was then converted to an AC power time series with respect to the SUNPOWER E20/435 solar panel data sheet [63]. The resulting data became 1089 row vectors of PVPPs. The data was segmented into four seasonal categories (i.e., fall, winter, spring, and summer) to obtain data sets with close profiles. The resulting data from the data segmentation step for the three years were 272, 270, 274, and 273 days for fall, winter, spring, and summer, respectively. For each of the clustering methods, 50 executions are carried out for two up to 20 clusters. The best result for each validity index among the 50 executions of each clustering method was registered.

3.8.1 Application of K-means

The K-means method was applied with 200 replicates. At each replicate a new set of initial centroids were chosen. The solution with the lowest intra-cluster distances was recorded.

3.8.2 Application of Ward's Hierarchical Clustering

Ward's hierarchical clustering was applied and cut-off at different levels to partition the data from two to 20 clusters. It should be mentioned that there are no other parameters for calibration, such as the maximum number of iterations. The dendogram for the fall data set is shown in Figure 3.4.

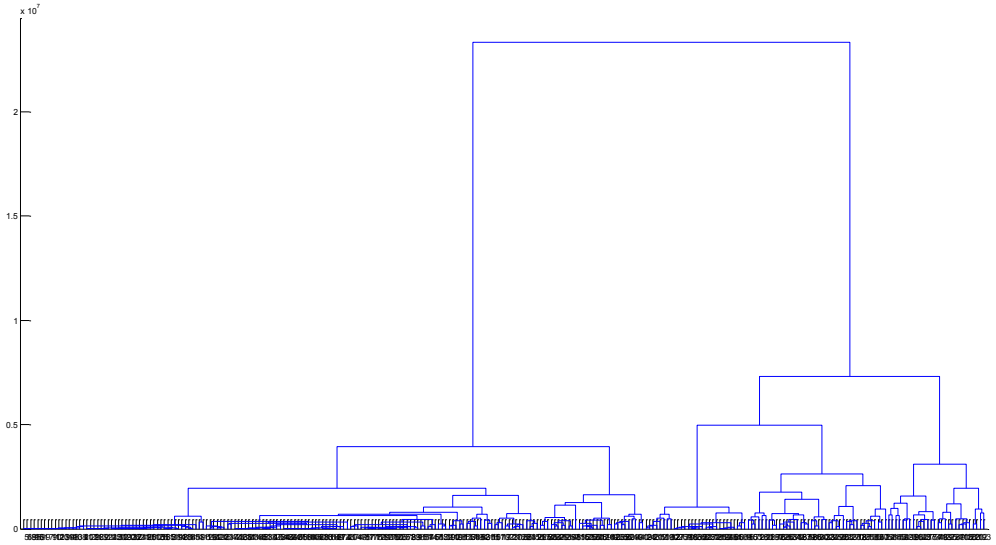

Figure 3.4: Dendogram for the fall data set using Ward's hierarchical clustering.

3.8.3 Application of FCM

The FCM algorithm was applied with different fuzziness values $m = \{2, 4, 6, 9\}$. The maximum number of epochs is 500 for the four scenarios and the upper limit of weight change between sequential iterations $\varepsilon = 10^{-4}$. The results of all adequacy values improved as the fuzziness

parameter increases. For lower fuzziness values m = {2, 4, 6}, dead clusters were produced, as shown in Figure 3.5. Hence, the simulations were applied using FCM with m = 9.

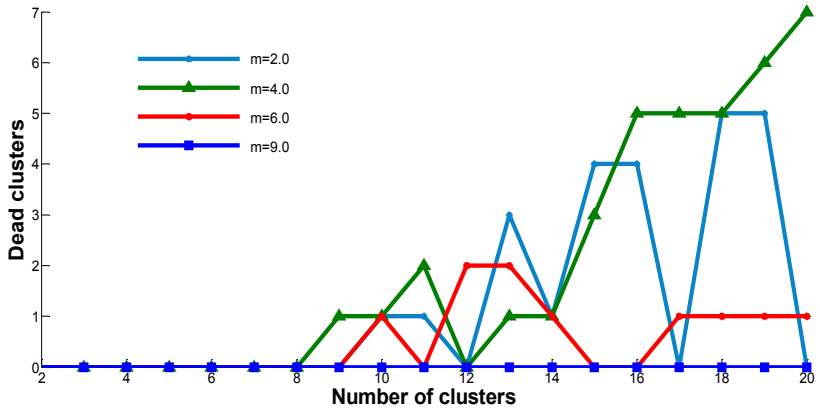

Figure 3.5: Dead clusters for the FCM method on fall data with m = {2, 4, 6, 9} for two to 20 clusters.

3.8.4 Application of SOM

The SOM has many parameters that can affect clustering results. The initial value, η_0, the minimum value, $\eta_{\min}$, the learning time rate, $T_{\eta 0,}$ and the number of epochs are calibrated in order to improve the neural network's behaviour. The SOM process was repeated for different values of learning rates, $T_{\eta 0,}$ and number of epochs with η_0 = 0.9 and $\eta_{\min}$ = 0.02. The sensitivity of the ratio of WCBCR to the $T_{\eta 0}$ and epochs parameters is presented in Figure 3.6.

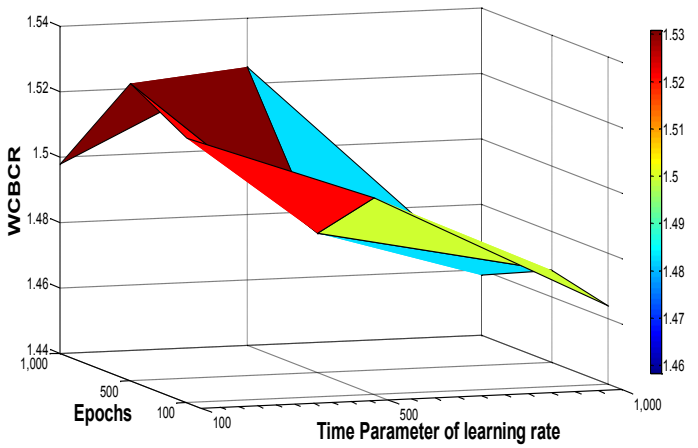

Figure 3.6: WCBCR values with respect to $T_{\eta 0}$ = {100,500,1000} and epochs = {100,500,1000} for mono-dimensional SOM with nine clusters.

3.8.5 Application of Ant Colony

The results of the parametric analysis were determined on the number of ants in the initialization and successive steps. The algorithm has been performed with 50 repetitions with A = {20, 50, and 100} and the solutions giving the best validity values were recorded. An overall best value when the number of ants increases has not been demonstrated (Figure 3.7). However, increasing the number of ants increases the number of fitness evaluations, directly proportional to the number of ants and iterations.

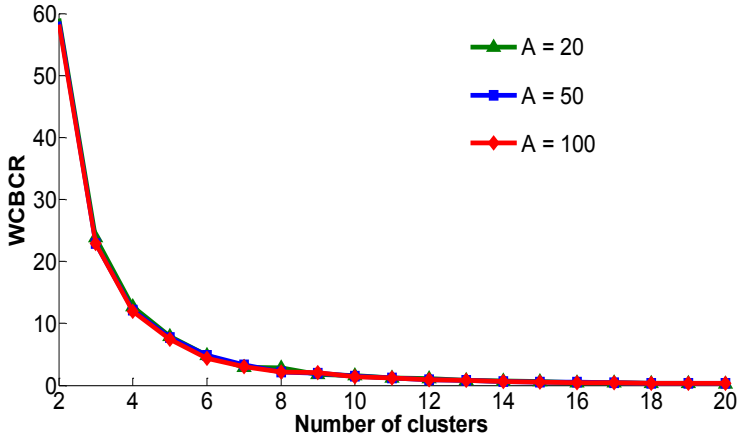

Figure 3.7: WCBCR values with respect to A = {20, 50, 100} for Ant Colony for two to 20 clusters.

3.8.6 Application of Bat

The Bat algorithm was performed with B = 50, A = 0.5, r = 0.5, and f_{min} = 0. The f_{max} and M parameters were calibrated to investigate their effect on adequacy measures. The overall best validity values were produced when f_{max} = 0.9 and M = 50 (Figure 3.8).

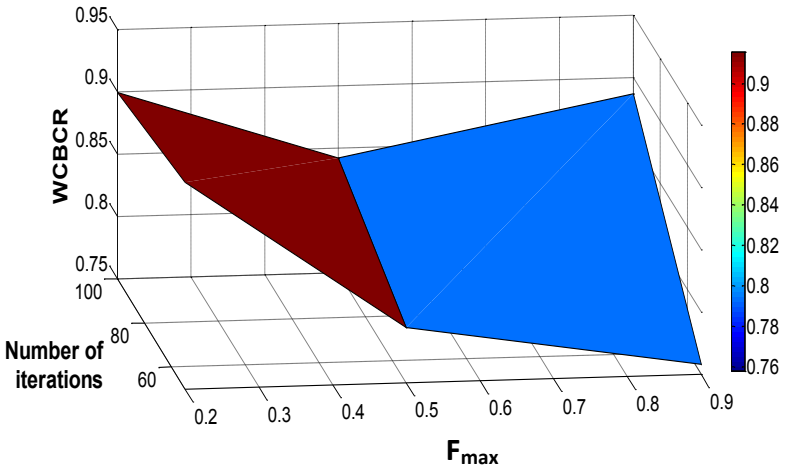

Figure 3.8: WCBCR values with respect to f_{max} = {0.2, 05, 0.9} and M = {50, 100} for Bat at ten clusters.

3.9 Comparison of Clustering Algorithms and Validity Indices

The best results for each of the aforementioned clustering algorithms on the fall data are presented in Figure 3.9. The K-means had the smallest values for the mean square error J and competitive values for WCBCR and CDI. The Hierarchical WMV had the best behaviour for XB. While FCM presented the overall worst results, it was not included in the DBI and XB plots as it produced abnormal values compared to all other clustering methods. The SOM and Ant Colony methods had an overall average performance on the validity indices but have not shown overall best results on any of validity indices. The Bat method presented the best behaviour for DBI, SI, WCBCR, CDI, and MIA. Moreover, the Bat algorithm demonstrated competitive results on all other validity indices.

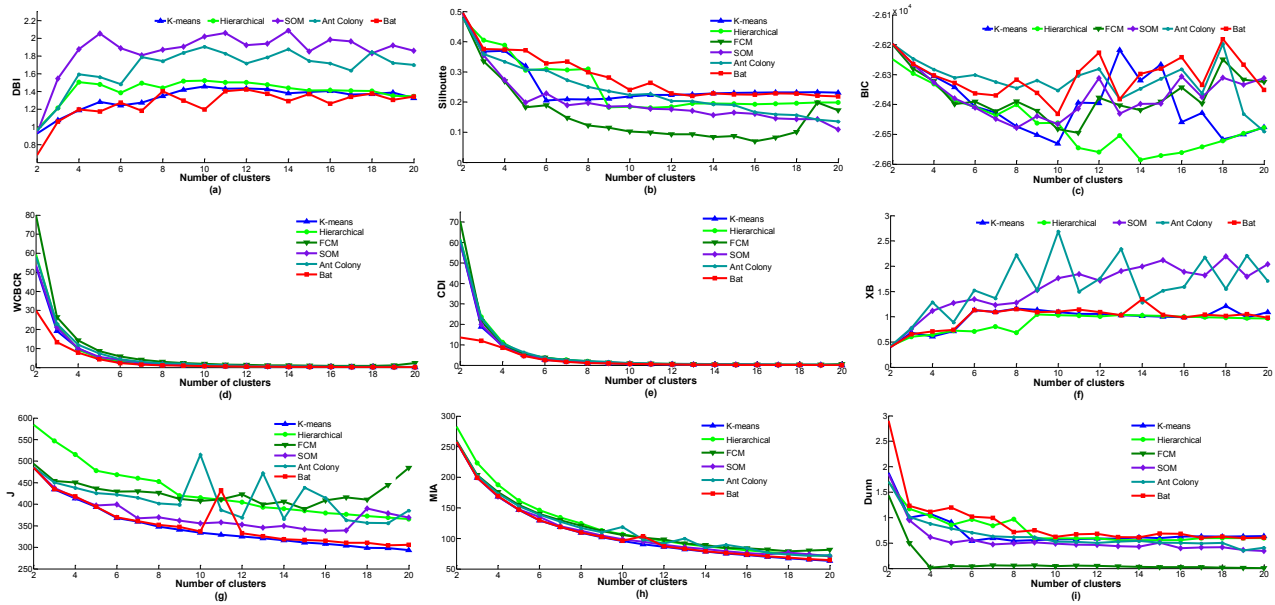

Figure 3.9: The best results of each clustering method for the fall season of the three-year data set of PVPPs for two to 20 clusters: (a) DBI (b) SI (c) BIC (d) WCBCR (e) CDI (f) XB (g) J (h) MIA (i) Dunn.

By comparing the measures of all validity indices with each other, an optimum number of clusters cannot be explicitly determined. Therefore, only indices that present adequate measures for all clustering algorithms will be used. For WCBCR and CDI indices, the performance improved as the number of clusters increased for the majority of clustering algorithms. In addition, all clustering algorithms had relatively similar measures with respect to those indices. It can be observed that the utilization of WCBCR is slightly better than CDI as it combines the distances of input data from the representative clusters and distance between clusters, which covers the J and CDI characteristics [10]. Therefore, the WCBCR was used to detect the appropriate number of clusters, while the CDI could be used to verify the number of clusters, as it is practical to use more than one validity index in evaluating a clustering method and choosing the optimum number of clusters. In Figure 3.9(d) the WCBCR index for two to 20 clusters for the fall is presented. The number of optimum clusters corresponds to the knee of the respective curve [10], [16], [17]. However, the values for clusters two to four are large. If these values are removed a knee point can be better observed. From Figure 3.10 there are a few possible knee

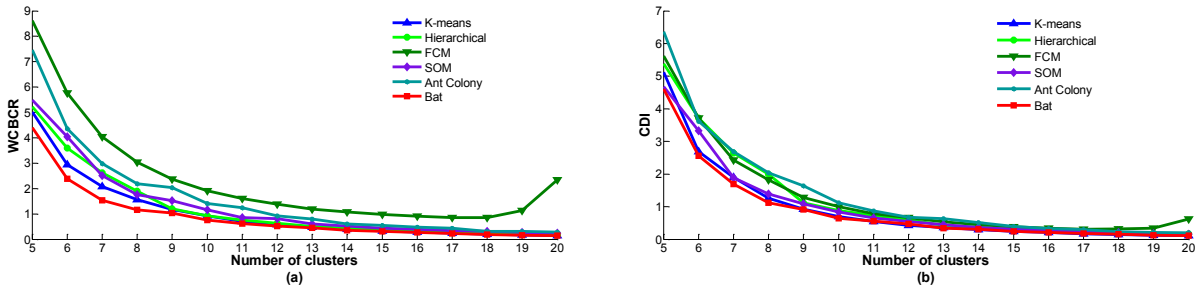

Figure 3.10: The best results of each clustering method for the fall data set of PVPPs for 5 to 20 clusters: (a) WCBCR. (b) CDI.

points between five and 12 that can be selected. In addition, the improvement of the index values is not significant after that. However, the knee point cannot be explicitly detected. For that, the angle-based method [49], [64] (Appendix A) on the WCBCR values of the Bat method was used to detect the knee points. Figure 3.11(a) presents the successive difference between points of the WCBCR index values of the Bat method. The method was able to detect four knee points at eight, 12, 14, and 18, with eight clusters having the largest knee angle. In order to validate this number, the CDI index was used. By applying the same method on the CDI index (Figure 3.11(b)), it can be observed that eight clusters can be an optimum number of clusters to represent the fall PVPP data. The comparison of validity indices values of the clustering algorithms for eight clusters is shown in Table 3.1. It can be observed that the Bat clustering algorithm presented the best performance on WCBCR and CDI indices; moreover, it presented the overall best performance results.

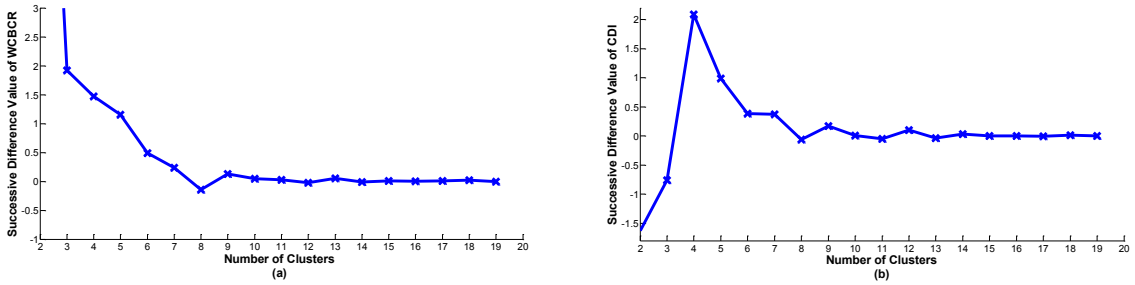

Figure 3.11: The successive difference for the angle-based method on the Bat clustering results for the fall data set of PVPPs for two to 20 clusters: (a) WCBCR (b) CDI.

Table 3.1: Validity indices of clustering algorithms for eight clusters of fall.

Validity index	DBI	Dunn	SI	BIC $\times 10^4$	WCBCR	CDI	XB	J	MIA
K-means	*1.349*	0.543	0.208	-2.647	1.565	1.264	1.160	*347.86*	*109.15*
Hierarchical WMV	1.439	*0.970*	*0.309*	-2.640	1.899	1.992	*0.686*	452.55	124.49
FCM	7.030	0.056	0.122	-2.638	3.033	1.818	78.145	426.35	120.84
SOM	1.869	0.491	0.197	-2.647	1.754	1.392	1.276	369.72	112.53
Ant Colony	1.742	0.617	0.250	-2.634	2.194	2.038	2.217	401.69	117.29
Bat	1.402	0.719	0.299	*-2.631*	*1.165*	*1.119*	1.150	352.20	109.83

Table 3.2: Comparison of clustering algorithms w.r.t compactness, separation, and CPU for eight clusters on fall data.

Clustering method	Compactness Intra-cluster dist.	Separation Inter-cluster dist.	CPU time(second)		
			Best	Worst	Average
K-means	413.823	923.078	8.1616	9.4534	8.7224
Hierarchical WMV	452.546	897.477	0.0876	*0.2035*	*0.1000*
FCM	398.745	800.907	*0.0684*	0.3835	0.1648
SOM	428.323	899.042	21.0917	22.8013	21.3620
Ant Colony	450.102	855.981	9.5542	10.0377	9.7180
Bat	*379.831*	*1.0120×10^3*	11.9034	12.5303	12.1348

The compactness and separation of the best clustering results for each clustering method showed that the best results for compactness and separation were obtained from the Bat method (Table 3.2). It should be noted that in Table 3.2 the best clustering results for each clustering method were chosen based on the WCBCR value.

The results for the clustering algorithms on winter, spring, and summer data are presented in Appendix B.

The representative PVPPs with their confidence limits for the eight clusters of fall produced by the Bat algorithm are presented in Figure 3.12. The intermediate area between the confidence limits has a probability of occurrence of 70% assuming normal distribution.

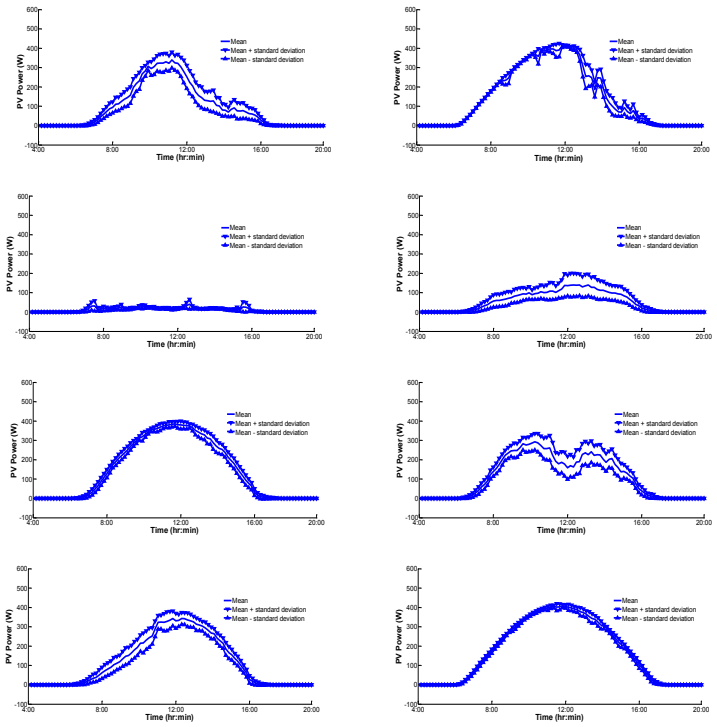

Figure 3.12: PVPPs with respective confidence intervals using Bat, assuming eight clusters.

3.10 Summary and Conclusions

In this chapter, the methodology to cluster PVPPs was presented in detail. Six clustering algorithms from different clustering categories have been tested in order to investigate the appropriate method for establishing the PVPP grouping process. The clustering algorithms that have been applied were as follows: K-means, Hierarchical WMV, FCM, SOM, Ant Colony and Bat. The clustering results of each method have been evaluated by nine validity indices (DBI, Dunn, SI, BIC, XB, J, CDI, MIA, and WCBCR) in order to obtain the optimum number of clusters that best fits the PVPP data and investigate the most efficient clustering method and validity index.

In addition to the conventional clustering methods, this chapter introduced bio-inspired swarm clustering methods to cluster PVPPs. The comparison of the clustering algorithms of different

characteristics and categories showed that bio-inspired swarm clustering methods were comparable to those conventional methods. Additionally, the clustering results of the Bat algorithm were the most efficient and outperformed the other clustering methods in many validity indices measures. However, it corresponds to increased complexity, as the number of parameters should be a priori calibrated.

The compactness and separation of the best clustering results for each clustering method showed that the best results for compactness and separation were obtained from the Bat method (Table 3.2). It should be noted that the best clustering results for each clustering method were chosen based on the WCBCR value. This supports the claim that the WCBCR validity index is an efficient indictor in evaluating PVPP clustering results. As it is practical to use more than one validity index, the angle-based method on the CDI index values also verified the values obtained by the WCBCR index. It can be observed from Table 3.2 that for the best two clustering methods (Bat and K-means), K-means outperformed Bat with respect to CPU time. Those observations were applicable to all other seasons as well.

Chapter 4

Comparisons Among Bat Algorithms with Various Objective Functions on Clustering PVPPs[3]

4.1 Introduction

From the results of the previous chapter, it was observed that the Bat clustering algorithm with the minimum mean square error, J, as an objective function (Bat J) outperformed the other clustering algorithms. Therefore, this chapter proposes five different versions of Bat algorithms as an attempt to enhance the clustering results. Each algorithm has a different objective function. The five proposed Bat algorithms are: Bat based on the Davies Bouldin Index (Bat DBI), Bat based on the Dunn index (Bat Dunn), Bat based on the clustering dispersion indicator (Bat CDI), Bat based on mean index adequacy (Bat MIA) and Bat based on within-cluster sum-of-squares to between-cluster variation (Bat WCBCR). The K-means clustering algorithm is also included in the comparison because of its extensive utilization in power pattern clustering. The dimensionality of the data is reduced by application of the PCA method in order to reduce clustering CPU time. This chapter presents the results of a detailed investigation of the performance of Bat clustering algorithms based on various objective functions to establish the grouping process of PVPPs.

4.2 General Methodology

The clustering of PVPPs is achieved by applying a similar methodology to that discussed in Chapter 3.2, except that the PVPP data is subjected to a dimension reduction technique known as PCA before applying the clustering algorithms. The layout of the methodology to investigate the performance of Bat clustering algorithms based on various objective functions and K-means to

[3]Chapter 4 of this thesis has been submitted as: A. A. Munshi, and Y. A.-R. I. Mohamed, "Comparisons among bat algorithms with various objective functions on grouping pv power patterns", submitted *to IEEE Transactions on Industrial Electronics*, Sep. 2014.

establish the grouping process of the dimension reduced PVPP data is shown in Figure 4.1. The pre-processing, data conversion, and data segmentation steps have been discussed previously in detail in Chapter 3.2. The data dimension reduction, data clustering, and evaluation and analysis of clustering results steps are discussed in the following sections.

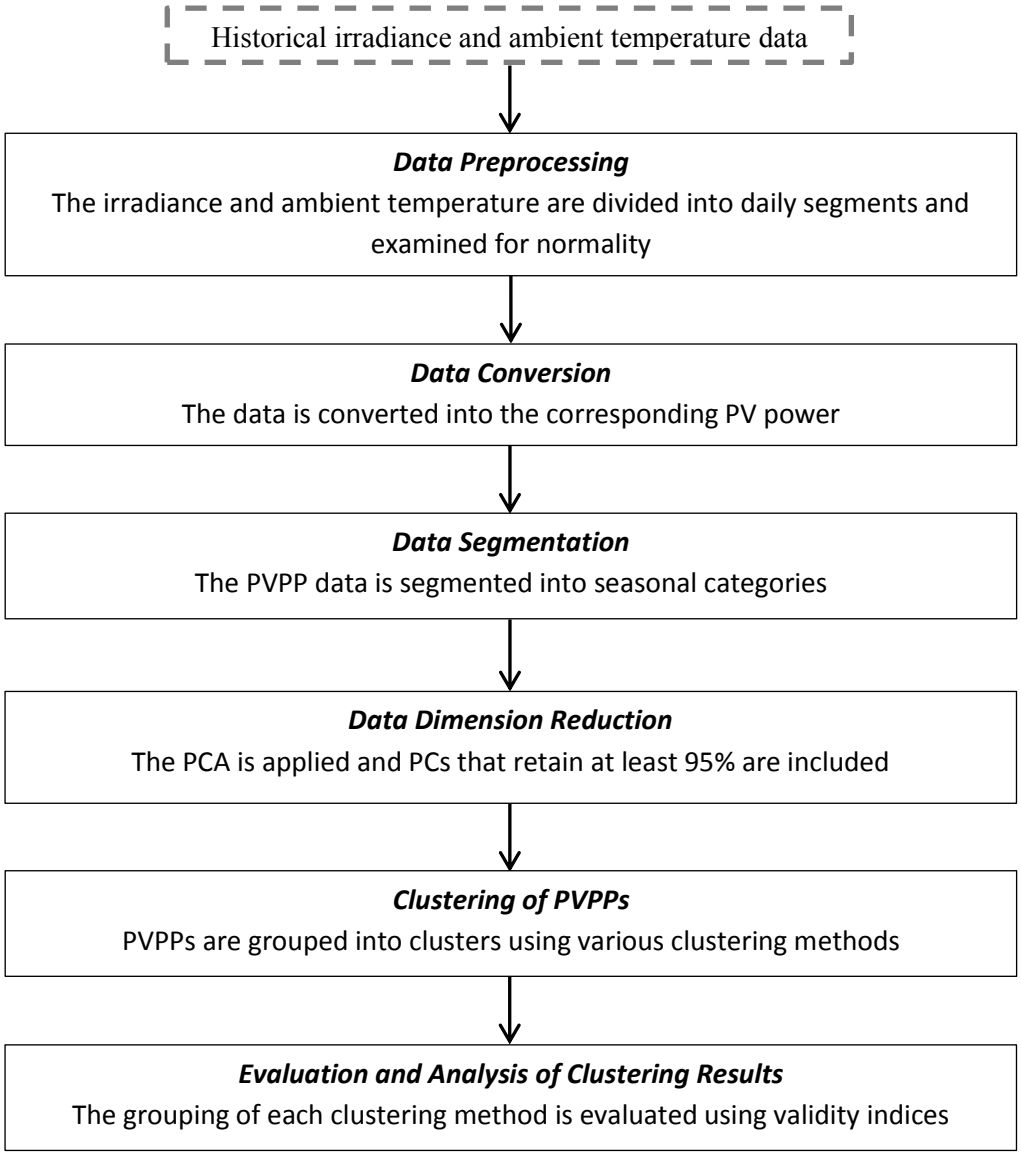

Figure 4.1: Layout of the methodology.

4.3 Data Dimension Reduction

In the pre-processing step, periods when the irradiance was not available were removed. That reduced the dimension of the data significantly. In order to reduce the remaining features, a feature generation technique (PCA) is employed to reduce the dimensionality of the PVPP data. The number of the features or principal components (PC) after applying the PCA should retain around 95% of the variation to achieve more accurate results. This step investigates the ability of the PCA to generate a set of features built in a reduced dimensional space but able to satisfactorily preserve the characteristics of the original PVPP data by using a significantly reduced number of features. For this purpose, the original N PVPPs are subject to a feature transformation process that maps them into a new dimensionally reduced data set.

4.4 Data Clustering

For each segment of PVPP time series data, seven clustering algorithms (K-means and six versions of Bat with different objective functions: Bat J, Bat DBI, Bat Dunn, Bat CDI, Bat MIA and Bat WCBCR) are used to assign PVPPs into clusters, so that patterns in the same cluster are more similar to each other than those in other clusters. From each cluster, a representative PVPP (centroid) can be obtained. Thus, the set of centroids can be used to represent the whole data.

The various objective functions for the Bat clustering algorithms are mainly validity indices integrated into the Bat algorithm. The definitions of the six validity indices (DBI, Dunn, J, CDI, MIA and WCBCR) were presented previously in Chapter 2.10.

4.5 Evaluation and Analysis of Clustering Results

From the previous step, each clustering algorithm produces different clusters and different centroids. Therefore, it is essential to evaluate the clustering results in order to choose the most suitable algorithm for clustering the PVPPs. The evaluation is based on the validity index values, the measurement of intra-cluster distance (compactness), and inter-cluster distance (separation) of the produced clusters. The compactness is represented by the average overall value of the

average distances of data points between their mean (centroid). The compactness can be defined by the following formula:

$$Compactness = \frac{1}{N}\sum_{k=1}^{K}\frac{1}{N_k}\sum_{x_i \in C_k} \| x_i - C_k \|^2 \quad . \tag{4.1}$$

The separation is measured by the overall average sum of distances between the centroids of a pair of clusters and is defined as:

$$Separation = \frac{1}{K}\sum_{1\le q<l}^{K} \| C_l - C_q \|^2 \quad . \tag{4.2}$$

4.6 Application of Bat Clustering Algorithms on PVPP Data

This methodology was applied on two data sets concerning the past three years with ten-minute time steps of irradiation and ambient temperature.

The first data set is from the Solar Radiation Research Laboratory [62] with the location latitude of 39.74ºN and longitude of 105.18ºW for three consecutive years (2010-2012). The data was then converted to an AC power time series with respect to the SUNPOWER E20/435 solar panel data sheet [63]. In order to reduce the dimensionality of the data set, the periods when the irradiance is not available were removed. Thus, each PVPP is limited to the period between 4:00 AM and 8:00 PM, which corresponds to 96 ten-minute time steps per day. Then the original PVPP data was segmented into four seasonal categories (i.e., fall, winter, spring, and summer). The use of PCA for reducing the dimensionality of the four seasons' data while retaining at least 95% of the total variance resulted in 13, 13, 18, and 24 PCs for fall, winter, spring, and summer, respectively. The Pareto diagram (Figure 4.2) shows the amount of the total variance using the first ten PCs for the summer PVPP data. For each season, the reduced PVPP data are normalized with respect to the maximum power contained in the PVPPs, as such, all PVPPs values fall between the [0,1] range. For each of the aforementioned clustering algorithms, 20 executions were conducted for two up to 20 clusters. The parameters for all Bat algorithms are defined in

Appendix C. In order to evaluate the performance of the clustering results, the original PVPP data was used, and the best result for each validity index among the 20 executions of each clustering algorithm were registered.

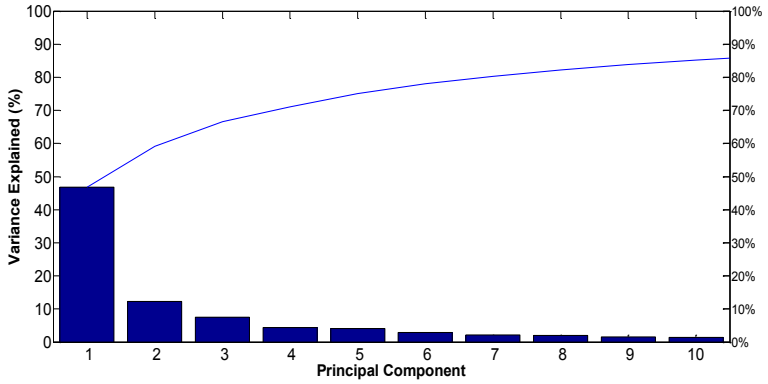

Figure 4.2: Pareto diagram of the PCA on the summer PVPPs (first data set).

The validity indices (J, DBI, Dunn, CDI, MIA, and WCBCR) were used to evaluate the performance of the clustering algorithms. The measurement of compactness and separation of the produced clusters was considered, as they form a key indicator that reflects the output quality of the clustering algorithm. It should be noted that the main objective of clustering is to achieve high values of separation and low values of compactness. The results illustrated in Figure 4.3 show that the information provided by the clustering validity indices was inconsistent. Some clustering algorithms show adequate results on certain validity indices but average results on other validity indices. Thus, an optimum number of clusters cannot be explicitly determined. For this purpose, the compactness and separation of the partitioning of each clustering algorithm should be examined, in order to investigate the best combination of clustering algorithm and validity index that presents the most compact and separate partitioning of PVPPs. It can be observed from Figure 4.3 that the knee points [10], [16], [17], were in the range of eight to 12 for the summer PVPP data. For that, the compactness and separation from eight to 12 clusters for each clustering algorithm were considered and computed.

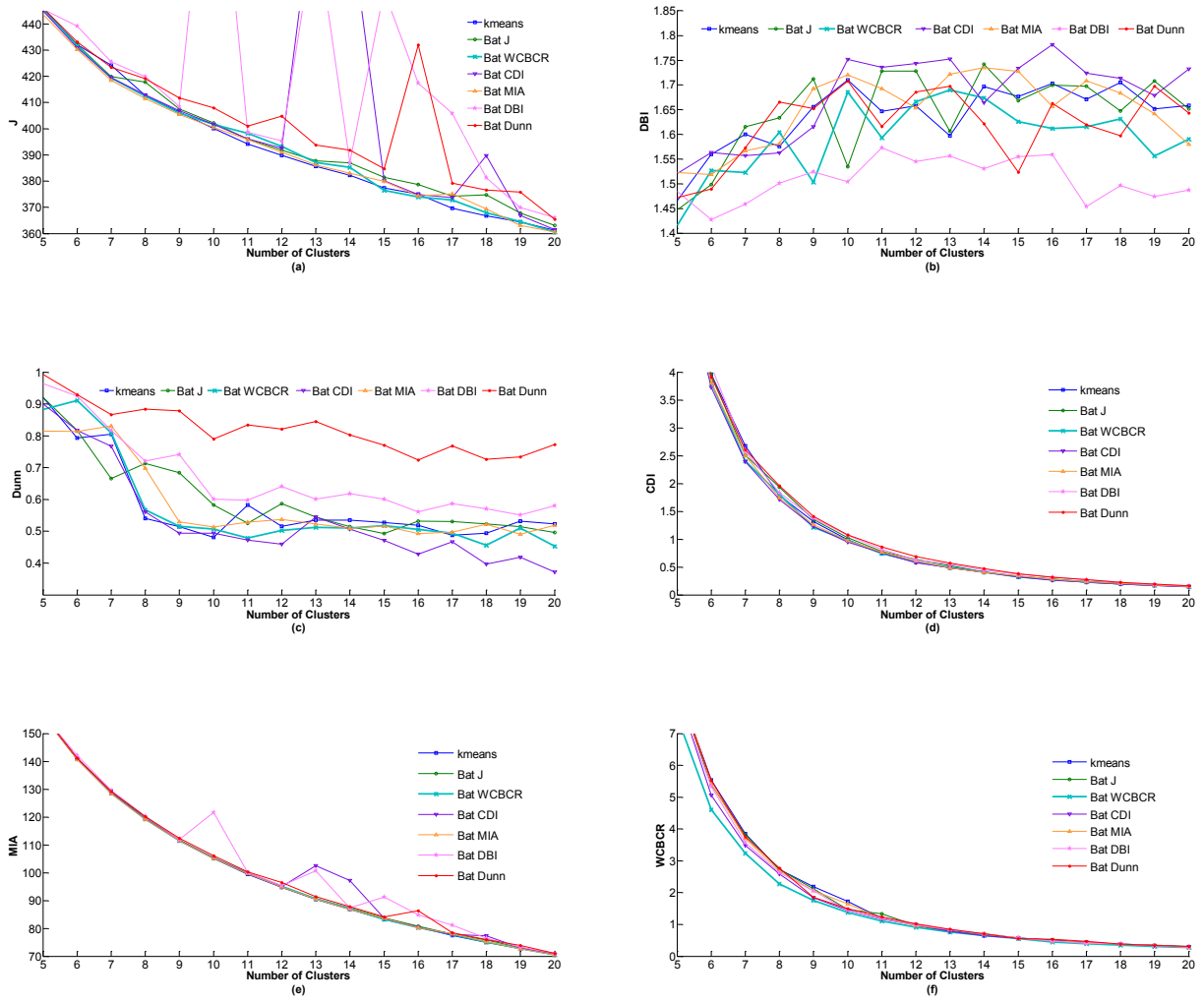

Figure 4.3: The best results of each clustering algorithm for the summer PVPP of the first data set for five to 20 clusters: (a) J (b) DBI (c) Dunn (d) CDI (e) MIA (f) WCBCR.

Table 4.1 illustrates the validity index value and the associated compactness and separation. In general, the results demonstrated that Bat clustering algorithms were either comparable or better that K-means with respect to the validity index, compactness, and separation values. The Bat J and Bat CDI both produced relatively compact clusters and have the highest compactness values

on ten and 12 clusters, respectively, but they produced average separated clusters. Bat DBI, Bat Dunn, and Bat MIA produced overall average results in compactness but tended to produce poor separated clusters. The Bat WCBCR presented the best overall results on the validity index, compactness, and separation values. It can be observed that when the WCBCR validity index value was the lowest, the separation was consistently high on all knee point partitions. In addition, the separation values for the Bat WCBCR were significantly higher than all other clustering algorithms and the associated compactness was lowest at nine clusters and relatively lower than the best compactness value for the other partitions (eight and ten to 12 clusters). Thus, the lower WCBCR validity index values on Bat WCBCR indicated highly separated clusters of PVPPs. Accordingly, the best combination of clustering algorithm and validity index that can present the overall best results of compactness and separation between PVPP clusters are the Bat WCBCR and WCBCR, respectively. The angle-based method (Appendix A) can be used to detect a knee point as illustrated in Chapter 3.9. Figure 4.4 presents the visualization of the first three PCs with respect to the Bat WCBCR clustering results for ten clusters of summer PVPP data. The ten cluster representatives for the summer PVPP data are presented in Figure 4.5. The CPU time of the various Bat clustering algorithms is presented in Table 4.2. It can be shown that Bat J was the fastest and Bat WCBCR was the second fastest, whereas Bat CDI had the worst CPU time among the Bat algorithms. The observations on the results of the other seasons (i.e., fall, winter, and spring) of the PVPP data were similar to those observed on the summer PVPP data.

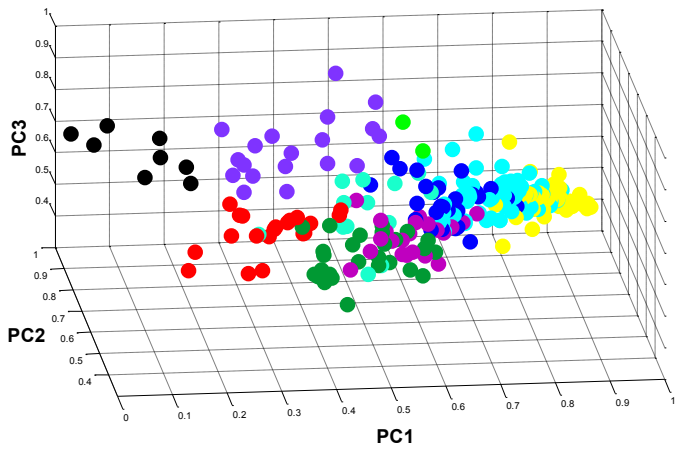

Figure 4.4: Visualization of the first three PCs w.r.t Bat WCBCR clustering results for ten clusters on summer PVPP of the first data set.

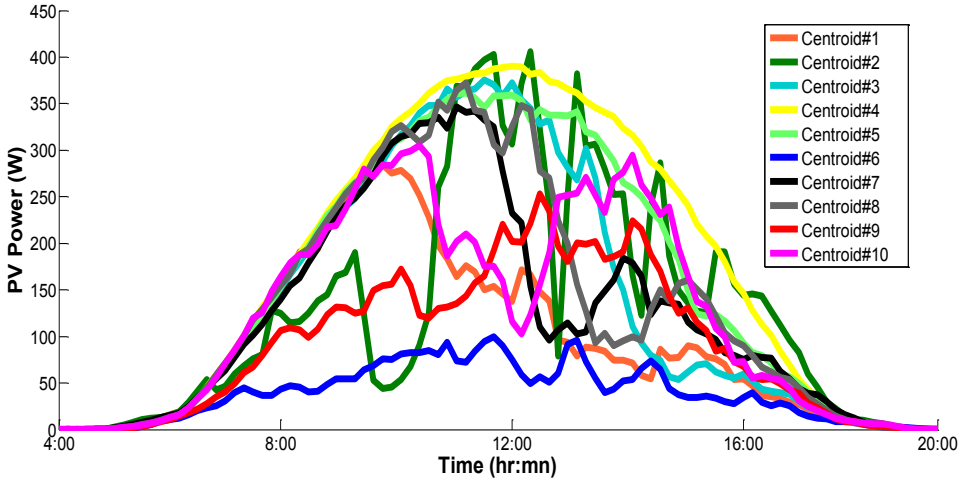

Figure 4.5: Cluster representatives for ten clusters of summer w.r.t Bat WCBCR clustering (first data set).

Table 4.1: Validity indices, compactness, and separation values for clustering algorithms on knee-points (first data set).

Validity index (VI)	8 Clusters			9 Clusters			10 Clusters			11 Clusters			12 Clusters		
	VI value	Comp.	Sep.	VI value	Comp.	Sep.	VI value	Comp.	Sep.	VI value	Comp.	Sep.	VI value	Comp.	Sep.
K-means															
J	412.649	488.207	811.368	406.364	479.526	785.517	400.101	474.184	780.750	394.231	463.134	828.970	389.852	456.994	821.239
DBI	1.576	488.207	811.368	1.656	479.526	785.517	1.710	474.184	780.750	1.646	462.963	832.028	1.658	460.202	819.774
Dunn	0.540	488.207	811.368	0.514	479.526	785.517	0.480	474.184	780.750	0.582	465.642	830.564	0.515	458.673	819.995
CDI	1.787	488.207	811.368	1.317	479.526	785.517	0.991	474.184	780.750	0.742	463.134	828.970	0.587	456.994	821.239
MIA	119.316	488.207	811.368	111.632	479.526	785.517	105.084	474.184	780.750	99.456	463.134	828.970	94.692	456.994	821.239
WCBCR	2.732	488.207	811.368	2.189	479.526	785.517	1.724	474.184	780.750	1.188	463.134	828.970	0.984	456.071	821.545
Bat J															
J	417.763	498.541	784.332	407.605	479.128	773.749	402.209	478.091	780.882	396.008	470.376	783.100	391.859	462.991	752.947
DBI	1.633	487.907	811.819	1.712	479.943	785.658	1.535	**430.450**	835.661	1.728	476.314	796.528	1.727	469.742	779.297
Dunn	0.712	487.723	784.837	0.683	484.237	771.162	0.582	470.098	837.992	0.525	476.314	796.528	0.586	465.953	839.299
CDI	1.936	498.541	784.332	1.357	479.128	773.749	1.025	478.091	780.882	0.779	470.376	783.100	0.639	462.991	752.947
MIA	120.053	498.541	784.332	111.802	479.128	773.749	105.361	478.091	780.882	99.680	470.376	783.100	94.935	462.991	752.947
WCBCR	2.729	487.907	811.819	2.120	484.291	802.574	1.461	467.147	837.782	1.338	476.314	796.528	0.971	465.953	839.299
Bat DBI															
J	419.930	490.069	752.115	408.409	480.772	803.485	536.728	469.853	796.525	398.639	475.120	798.279	395.405	467.100	770.688
DBI	1.502	485.036	828.996	1.524	482.868	795.906	1.505	466.791	807.884	1.573	466.647	793.705	1.545	465.751	826.123
Dunn	0.721	491.977	805.265	0.741	494.070	779.359	0.601	483.234	796.404	0.597	478.006	790.411	0.641	465.751	826.123
CDI	1.837	**444.791**	811.286	1.383	481.010	796.015	1.072	466.579	844.534	0.816	463.813	836.056	0.628	433.658	796.249
MIA	120.364	490.069	752.115	111.913	480.772	803.485	121.712	469.853	796.525	100.011	475.120	798.279	95.364	467.100	770.688
WCBCR	2.634	480.283	829.540	2.054	485.066	821.849	1.442	465.855	851.039	1.188	463.813	836.056	0.989	461.136	825.824
Bat Dunn															
J	419.066	500.602	791.079	411.775	486.514	758.922	407.959	486.180	754.253	401.041	478.977	768.461	404.831	475.162	769.651
DBI	1.666	499.576	800.145	1.652	476.543	852.708	1.708	476.277	815.908	1.617	438.504	821.751	1.686	468.386	816.812
Dunn	0.884	500.962	771.174	0.878	493.777	799.397	0.790	486.180	754.253	0.834	483.941	758.970	0.821	472.351	820.135
CDI	1.960	500.602	791.079	1.409	476.543	852.708	1.078	474.466	830.304	0.859	465.091	819.307	0.686	468.085	809.458
MIA	120.240	500.602	791.079	112.373	486.514	758.922	106.112	486.180	754.253	100.312	478.977	768.461	96.494	475.162	769.651
WCBCR	2.770	491.533	808.876	1.849	476.543	852.708	1.500	475.143	833.196	1.235	465.091	819.307	1.022	463.937	813.435
Bat CDI															
J	412.910	472.618	790.110	406.931	466.235	782.924	401.707	470.182	790.347	396.125	461.538	760.275	392.633	460.683	780.974
DBI	1.563	478.187	817.031	1.616	477.652	795.891	1.752	469.866	791.103	1.736	459.660	831.986	1.743	**425.032**	818.930
Dunn	0.561	490.589	819.314	0.494	472.928	781.216	0.494	463.529	842.931	0.472	474.716	787.930	0.459	471.676	791.543
CDI	1.712	472.618	790.110	1.234	466.235	782.924	0.949	470.182	790.347	0.759	461.538	760.275	0.580	460.683	780.974
MIA	119.354	472.618	790.110	111.710	466.235	782.924	105.295	470.182	790.347	99.695	461.538	760.275	95.029	460.683	780.974
WCBCR	2.591	472.694	847.528	1.843	467.110	850.383	1.440	463.529	842.931	1.173	454.458	833.701	0.982	451.509	820.471
Bat MIA															
J	411.456	481.314	814.455	405.483	477.055	795.869	400.504	465.291	782.622	395.834	471.331	780.430	391.044	458.263	777.258
DBI	1.581	485.789	815.690	1.692	477.958	799.206	1.721	464.142	808.837	1.693	469.259	777.902	1.654	437.486	821.612
Dunn	0.698	502.836	753.885	0.529	487.054	790.293	0.513	480.387	795.484	0.528	466.137	802.483	0.537	463.875	839.545
CDI	1.743	481.314	814.455	1.264	477.055	795.869	0.974	465.291	782.622	0.770	471.331	780.430	0.606	458.263	777.258
MIA	119.144	481.314	814.455	111.511	477.055	795.869	105.138	465.291	782.622	99.659	471.331	780.430	94.837	458.263	777.258
WCBCR	2.685	482.111	819.177	2.060	482.444	815.094	1.651	464.142	808.837	1.198	462.614	828.553	0.971	460.034	830.790
Bat WCBCR															
J	412.362	480.376	835.734	405.915	467.393	820.950	401.636	450.431	812.736	398.271	449.364	**858.801**	393.198	456.568	791.899
DBI	1.604	479.844	836.224	1.503	463.513	866.186	1.685	466.915	804.868	1.593	**419.425**	819.432	1.666	462.612	791.640
Dunn	0.566	481.094	842.541	0.515	463.513	866.186	0.507	455.175	**880.213**	0.478	456.360	846.390	0.502	457.337	831.361
CDI	1.804	480.376	835.734	1.214	467.393	820.950	0.978	450.431	812.736	0.741	449.364	**858.801**	0.604	456.568	791.899
MIA	119.275	480.376	835.734	111.571	467.393	820.950	105.286	450.431	812.736	99.964	449.364	**858.801**	95.097	456.568	791.899
WCBCR	2.275	472.714	**887.560**	1.755	**458.888**	**871.496**	1.374	455.175	**880.213**	1.108	434.283	856.650	0.912	431.356	**853.163**

Table 4.2: CPU time for Bat clustering algorithms of ten clusters on PVPP summer data (first data set).

Clustering Algorithm	CPU time (second)		
	Best	Worst	Average
Bat J	**26.88**	**28.41**	**27.12**
Bat DBI	39.70	46.53	41.19
Bat Dunn	45.47	46.18	45.73
Bat CDI	141.15	147.69	142.92
Bat MIA	38.9103	40.12	39.39
Bat WCBCR	32.86	33.39	33.03

The same methodology was applied on a second data set with a latitude of 21.68ºN and a longitude of 39.15ºE for three consecutive years (1999-2001). The dimensionality reduction resulted in the limitation of each PVPP to the period between 5:00 AM and 8:00 PM, which corresponds to 90 ten-minute time steps per day. While the PCA retaining at least 95% of the total variance resulted in 22, 19, 21, and 25 PCs for fall, winter, spring, and summer, respectively. The knee points were observed to be in the range of seven to 11, for the summer PVPP data (Appendix D). The results of the validity index value and the associated compactness and separation from seven to 11 clusters for each clustering algorithm are presented in Table 4.3. The observations on the results are similar to those observed on the first PVPP data set, where lower WCBCR validity index values on Bat WCBCR indicated highly separated clusters. In addition, Bat WCBCR presented the best overall results on the validity index, compactness, and separation values.

Table 4.3: Validity indices, compactness, and separation values for clustering algorithms on knee-points (second data set).

Validity index (VI)	7 Clusters			8 Clusters			9 Clusters			10 Clusters			11 Clusters		
	VI value	Comp.	Sep.	VI value	Comp.	Sep.	VI value	Comp.	Sep.	VI value	Comp.	Sep.	VI value	Comp.	Sep.
K-means															
J	149.620	210.642	516.512	147.008	220.790	509.146	145.283	224.690	519.530	142.292	216.716	501.564	141.025	215.448	513.989
DBI	1.383	210.642	516.512	1.446	220.790	509.146	1.288	218.190	527.175	1.310	223.412	538.101	1.463	212.491	503.803
Dunn	0.394	210.642	516.512	0.351	220.790	509.146	0.437	218.190	527.175	0.419	218.122	498.722	0.412	200.871	535.533
CDI	1.436	210.642	516.512	1.070	220.790	509.146	0.834	224.690	519.530	0.623	216.716	501.564	0.502	213.476	478.861
MIA	75.968	210.642	516.512	70.438	220.790	509.146	66.019	224.690	519.530	61.983	216.716	501.564	58.835	215.448	513.989
WCBCR	1.169	210.642	516.512	0.893	221.382	510.911	0.660	218.190	527.175	0.503	223.412	538.101	0.399	200.871	535.533
Bat J															
J	149.620	210.642	516.512	147.181	211.183	488.609	146.094	184.351	498.231	140.903	202.554	482.590	138.586	196.995	461.200
DBI	1.176	193.119	597.172	1.307	184.351	516.332	1.435	184.863	529.181	1.392	177.946	540.569	1.277	199.956	513.750
Dunn	0.529	192.063	567.291	0.399	208.594	485.776	0.411	218.032	497.634	0.380	227.513	493.758	0.398	215.686	463.218
CDI	1.436	210.642	516.512	1.037	211.183	488.609	0.795	184.351	498.231	0.600	202.554	482.590	0.475	196.995	461.200
MIA	75.968	210.642	516.512	70.480	211.183	488.609	66.203	184.351	498.231	61.680	202.554	482.590	58.324	196.995	461.200
WCBCR	0.992	193.119	597.172	0.783	181.895	555.265	0.645	184.863	529.181	0.481	177.946	540.569	0.403	172.006	533.477
Bat DBI															
J	155.124	208.403	558.983	148.625	213.929	524.865	144.979	190.384	481.246	142.628	188.650	478.438	142.084	194.884	507.354
DBI	1.168	200.769	566.671	1.125	**171.592**	570.514	1.185	188.325	563.818	1.270	177.734	519.936	1.145	167.404	561.934
Dunn	0.440	204.822	540.564	0.461	214.425	537.301	0.433	157.402	572.493	0.426	206.085	499.679	0.426	182.763	506.936
CDI	1.615	204.702	536.866	1.157	213.929	524.865	0.813	190.384	481.246	0.615	188.650	478.438	0.547	206.579	483.360
MIA	77.352	208.403	558.983	70.824	213.929	524.865	65.950	190.384	481.246	62.056	188.650	478.438	59.055	194.884	507.354
WCBCR	1.006	202.235	576.404	0.769	**171.592**	570.514	0.591	157.402	572.493	0.476	200.416	546.898	0.382	167.404	561.934
Bat Dunn															
J	153.323	220.389	520.677	151.686	219.287	494.343	147.971	186.068	498.963	146.050	198.131	504.922	183.004	227.096	504.238
DBI	1.324	222.465	489.668	1.275	204.732	547.603	1.251	203.306	541.255	1.232	174.927	541.199	1.285	208.799	526.927
Dunn	0.675	222.356	526.446	0.641	234.526	531.128	0.634	195.006	509.384	0.649	228.426	510.897	0.648	218.260	486.717
CDI	1.700	220.389	520.677	1.247	219.287	494.343	0.847	186.068	498.963	0.694	198.131	504.922	0.563	170.521	503.628
MIA	76.902	220.389	520.677	71.550	219.287	494.343	66.627	186.068	498.963	62.796	198.131	504.922	67.022	227.096	504.238
WCBCR	1.172	234.470	542.075	0.877	190.466	530.236	0.674	203.306	541.255	0.513	204.007	537.629	0.434	208.799	526.927
Bat CDI															
J	149.396	210.644	515.295	146.182	200.815	477.449	144.715	204.842	491.577	143.744	212.546	475.578	140.713	207.421	456.163
DBI	1.460	210.739	517.559	1.304	207.944	523.140	1.473	188.260	502.396	1.483	189.524	508.327	1.520	199.706	479.235
Dunn	0.395	212.674	511.387	0.417	217.113	518.971	0.401	213.960	473.854	0.359	188.240	496.281	0.375	207.421	456.163
CDI	1.402	210.644	515.295	0.987	200.815	477.449	0.756	204.842	491.577	0.604	212.546	475.578	0.467	207.421	456.163
MIA	75.911	210.644	515.295	70.240	200.815	477.449	65.890	204.842	491.577	62.298	212.546	475.578	58.770	207.421	456.163
WCBCR	1.165	210.665	516.753	0.851	184.384	529.040	0.704	188.260	502.396	0.535	189.524	508.327	0.474	189.458	479.925
Bat MIA															
J	149.533	191.114	482.968	147.642	208.936	489.388	145.422	215.957	493.403	143.943	210.795	492.512	140.709	208.530	452.171
DBI	1.363	209.311	516.438	1.336	183.318	528.119	1.421	215.957	493.403	1.325	168.452	500.337	1.378	168.712	505.695
Dunn	0.510	236.433	418.647	0.395	210.545	490.156	0.403	215.957	493.403	0.397	168.452	500.337	0.377	214.231	499.940
CDI	1.513	209.226	521.177	1.072	208.936	489.388	0.818	215.957	493.403	0.629	167.093	490.747	0.474	208.530	452.171
MIA	75.945	191.114	482.968	70.590	208.936	489.388	66.050	215.957	493.403	62.342	210.795	492.512	58.769	208.530	452.171
WCBCR	1.151	209.226	521.177	0.852	183.318	528.119	0.652	185.267	523.639	0.551	216.595	502.148	0.433	172.560	498.482
Bat WCBCR															
J	151.992	196.000	569.315	147.301	191.781	530.847	144.645	203.054	525.696	142.498	196.207	504.116	141.457	199.400	502.037
DBI	1.005	**163.417**	585.336	1.200	205.243	565.127	1.074	**153.185**	**598.344**	1.089	**165.558**	**583.919**	1.063	167.368	**590.870**
Dunn	0.447	197.125	566.219	0.421	198.753	531.954	0.426	210.165	527.964	0.422	214.132	536.232	0.418	**161.263**	538.838
CDI	1.607	196.000	569.315	1.099	191.781	530.847	0.845	203.054	525.696	0.654	196.207	504.116	0.518	184.839	507.111
MIA	76.567	196.000	569.315	70.508	191.781	530.847	65.874	203.054	525.696	62.028	196.207	504.116	58.925	199.400	502.037
WCBCR	0.975	195.433	**608.155**	0.753	206.288	**574.855**	0.555	**153.185**	**598.344**	0.445	172.437	575.250	0.346	202.421	585.980

4.7 Summary and Conclusions

This chapter presented a detailed investigation of the performance of Bat clustering algorithms based on various integrated objective functions to establish the clustering process of PVPPs. In addition, it compared the performance of the K-means and Bat J clustering algorithms with those new versions of Bat clustering algorithms (i.e., Bat DBI, Bat Dunn, Bat CDI, Bat MIA, Bat WCBCR) on clustering PVPP data. For the purpose of reducing the dimensionality of the PVPP data during the clustering process, the PCA technique was adopted and the number of retained PCs were those that preserved at least 95% variation.

The clustering results of each clustering algorithm have been evaluated by six validity indices: J, DBI, Dunn, CDI, MIA, and WCBCR. In addition, the separation and compactness for each clustering algorithms' partitioning at the knee points were examined in order to obtain the best combination of clustering algorithm and validity index that can provide information about the optimum number of clusters that best fits the PVPP data. The methodology was applied on two PVPP data sets; in general, the results demonstrated that Bat clustering algorithms were either comparable or outperformed K-means in the validity index, compactness, and separation values. The Bat WCBCR presented the best overall results. The best combination that can present the optimum number of clusters was the Bat WCBCR clustering algorithm and the WCBCR validity index. Together, these presented significantly highly separated and well-compacted clusters. The main purpose of this chapter was to enhance the results of the best clustering algorithm from the previous chapter. This goal has been achieve by the Bat WCBCR clustering algorithm. Moreover, the methodology included PCA that reduced the dimensionality of PVPP data, and accordingly, the clustering CPU time was reduced. Thus, the Bat WCBCR can provide well-defined PVPP clusters and cluster representatives that can be utilized in PV power application studies.

Chapter 5

Short-term Prediction of PV Power

5.1 Introduction

The integration of PV systems into the electrical grid is considered to be a challenging task due to the uncertainty of solar irradiation. A key to solve this problem is to accurately predict short-term PV power generation. The PV power prediction is essential to increasing the penetration of solar power systems in electrical grids. Accurate short-term PV power predictions can assist in the optimization of power systems and operation control. However, the accuracy of PV power prediction depends mostly on the meteorological and climatic conditions, which makes it a challenging task.

Generally, PV power prediction is based on solar irradiation. Several models have been developed in order to predict solar irradiance data. These models can be mainly classified into two categories [65]: physical models and statistical models. Physical models use mathematical equations mostly in order to describe the physics and dynamics of the atmosphere that influences solar irradiation [66]. They work well for medium and long-term solar predictions [65]. Statistical models are mainly based on analyzing time series data. They have lower complexity than physical models and have the ability to perform well for short-term predictions. These statistical models include the artificial neural network (ANN) [67], [68], autoregressive (AR) and autoregressive moving average (ARMA) [69], and support vector machine (SVM) [65] models. Such models have shown their efficiency in predicting solar irradiation.

This chapter presents a model for short-term predictions of PV power. The approach of this model uses a dedicated formulation in order to test the efficiency of the PVPP cluster representatives obtained from the previous chapters. The results are compared with a single-point (ten-min) and three-points (30-min) shifting methods.

5.2 PV Power Prediction Model

At time (t), this short-term PV power prediction model predicts the future PV power generation ($t+1,t+2,\ldots,t+f$) from the past values ($t-1,t-2,\ldots,t-n$) of ambient temperature, solar irradiation, and representative PVPPs. The prediction is based on the classification of the past PVPP time steps to the representative PVPPs, then the future values are obtained from the closest PVPPs. The diagram of the model in presented in Figure 5.1 and the steps are as follows:

1- The sequence of ambient temperature and solar irradiation prior to the interval to be predicted ($t-1,t-2,\ldots,t-n$) are obtained.
2- Calculate the corresponding AC power output for ($t-1,t-2,\ldots,t-n$) using the model discussed in Chapter 3.4.
3- Calculate the distance between the obtained sequence and the corresponding time sequence of each representative PVPP.
4- Obtain the two closest PVPPs and calculate the mean distance between them. The result is three PVPPs.
5- Calculate the distance between the obtained sequence from step two and the corresponding time sequence of the three PVPPs from step four.
6- The future PV power values ($t+1,t+2,\ldots,t+f$) are obtained from the closest PVPP.

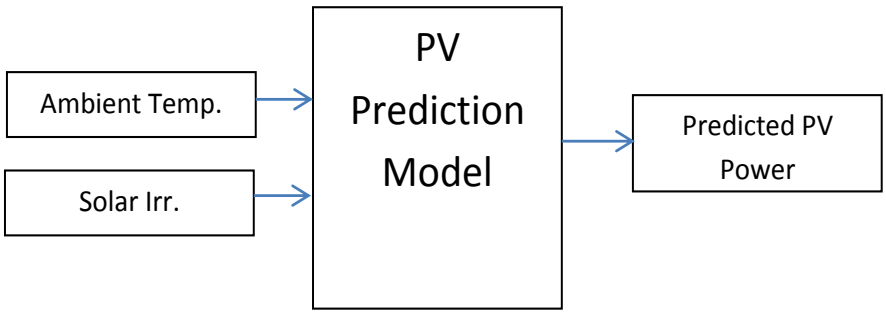

Figure 5.1: Diagram of the prediction model.

The flowchart of the model in presented in Figure 5.2.

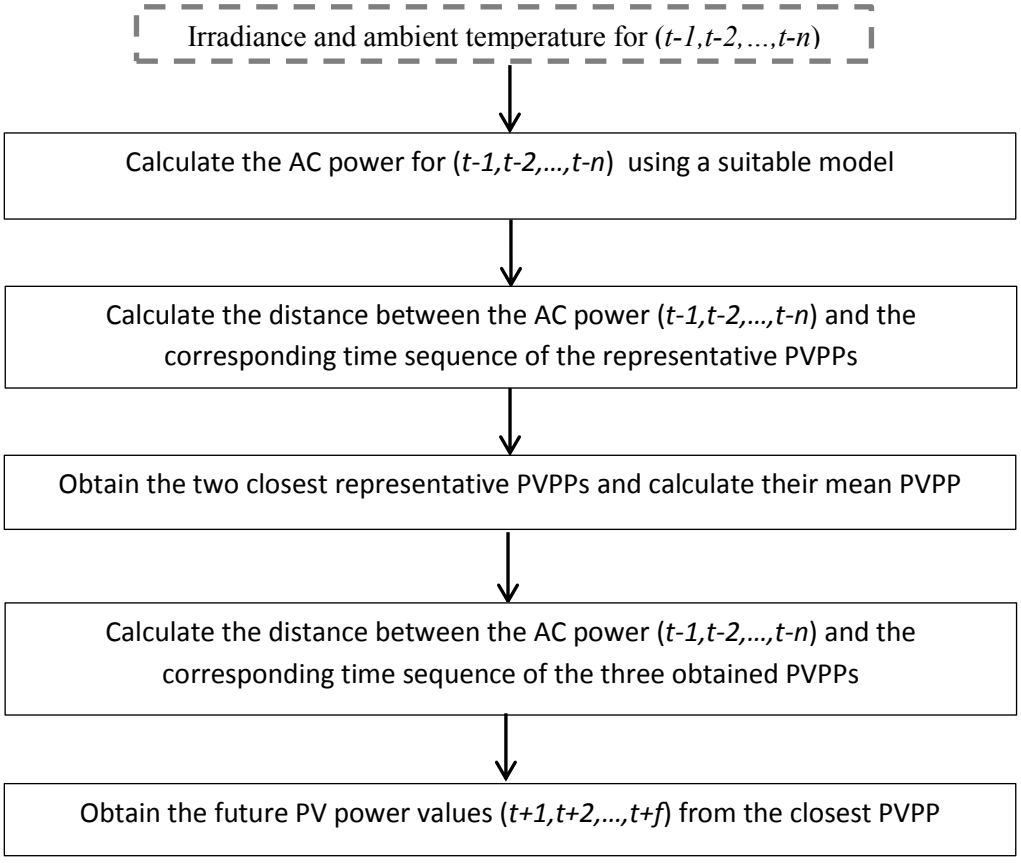

Figure 5.2: Flowchart of the model.

5.3 Application of PV Power Prediction Model

The short-term PV prediction model is applied on a real data set in order to predict ten-minute time steps of 30 minutes, 60 minutes, and 120 minutes ahead. The cluster representatives are obtained by applying the methodology of Chapter 4 using the Bat WCBCR clustering algorithm. The clustering of three consecutive years for the data resulted in 32 representative PVPPs. The ambient temperature and irradiation for a following year were converted to AC power and the

test data was obtained by choosing every tenth day of that year. Accordingly, 36 daily PVPPs were obtained. The results of predicting 30 minutes, 60 minutes, and 120 minutes ahead from a sequence of the past 30 minutes, 60 minutes, 90 minutes, and 120 minutes are illustrated in Table 5.1. It should be noted that the RMSE, MAE, and correlation coefficient (Appendix E) were calculated between the actual data and predicted data, and the RMSE and MAE arranged between 19.374 and 25.983, and 9.034 and 14.692, respectively. The correlation coefficient values were all above 97.8%. Smaller values of RMSE and MAE imply a superior prediction performance of the model. While a larger positive correlation coefficient value indicates that the data are more correlated. It can be observed that when the sequence of the prediction increased, the error increased. Also, it can be observed that increasing the past sequence does not improve the prediction. Figure 5.3 presents the comparison between the actual and predicted PV power for predicting 60 minutes from the past 30 minutes.

Table 5.1: The results between the actual and predicted data.

Past time sequence (min)	Predicted time sequence (min)	RMSE	MAE	Corr.
30	30	19.374	9.034	0.988
30	60	21.542	11.133	0.985
30	120	25.983	14.692	0.978
60	30	19.320	9.378	0.988
60	60	21.860	11.318	0.984
60	120	25.716	14.451	0.978
120	30	19.584	10.161	0.987
120	60	21.749	11.620	0.985
120	120	25.687	14.677	0.978
180	30	20.499	10.741	0.987
180	60	22.366	11.887	0.985
180	120	24.906	13.611	0.981
240	30	20.533	10.367	0.988
240	60	22.283	11.171	0.986
240	120	24.846	13.267	0.982

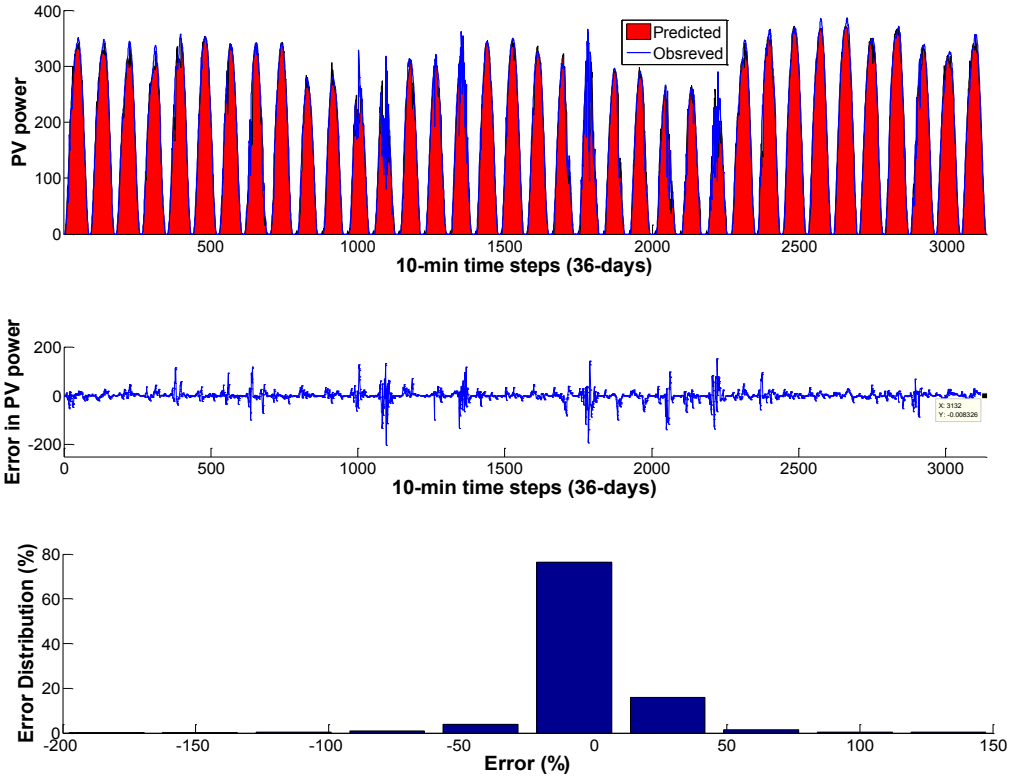

Figure 5.3: Comparison between the actual and predicted PV power
for predicting 60 minutes from the past 30 minutes.

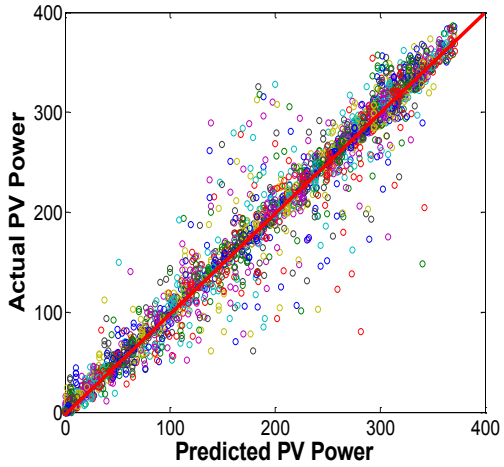

Figure 5.4: Correlation between the actual and predicted PV power.

Table 5.2: The results of the single-point (ten-min) and three-point (30-min) shifting methods.

Shifting method	Predicted time sequence (min)	RMSE	MAE	Corr.
Single-point	10	18.417	10.108	0.989
Three-point	30	33.018	24.221	0.965

Figure 5.4 shows the correlation between the actual and predicted data. It can be observed from the slope of the fitting line that the data falls close to the 45º line. Therefore, the predicted values closely match the actual data, which indicates accurate prediction. The single-point and three-points shifting methods showed significantly large RMSE and MAE.

5.4 Summary and Conclusions

This chapter presented a short-term PV power prediction model. The aim was to construct a model that can take advantage of the PVPP representatives. Also, this chapter tested the efficiency of the representative PVPPs resulting from the PVPP clustering methodology discussed in the previous chapters. The model was applied on real data and the error between the actual data and predicted data ranged between 19.374 and 25.983, and 9.034 and 14.692 for RMSE and MAE, respectively. While the correlation coefficient values were all above 97.8%. It was observed that when the sequence of the prediction increased, the error increased. Also, it was observed that increasing the past sequence does not improve the prediction accuracy for this model.

Chapter 6

6.1 Summary and Conclusions

This thesis facilitated developing a solution that can reduce the burden of extensive studies and simulations related to integrating PV systems into the electrical grid.

Chapter 2, presented an overview of clustering methods and validity indices used in the PVPP clustering process. Also, the dimensionality reduction technique (PCA) used in Chapter 4 was illustrated in detail.

In Chapter 3, the methodology to prepare the irradiance and ambient temperature data for clustering was presented in detail. Investigation of the most appropriate clustering algorithm to establish the PVPP grouping was analysed. At least one representative algorithm from various clustering categories was used (K-means from partitional clustering, Hierarchical WMV from agglomerative clustering, FCM from fuzzy clustering, SOM from neural network based algorithms, and Ant Colony and Bat algorithms from particle swarm optimization methods) to investigate the most appropriate for PVPP data clustering. The comparison of the clustering algorithms of different characteristics and categories showed that swarm clustering methods are comparable to these conventional methods. Additionally, the clustering results of the Bat algorithm were the most efficient and outperformed the other clustering methods in many validity indices measures. This motived the interest to enhance the swarm based, Bat clustering algorithm. Therefore in Chapter 4, five Bat clustering algorithms with different objective functions (Bat DBI, Bat Dunn, Bat CDI, Bat MIA, and Bat WCBCR) were proposed based on the results of Chapter 3. The methodology of Chapter 4 included PCA that reduced the dimensionality of PVPP data, and accordingly, the clustering CPU time was reduced. The methodology was applied on two PVPP data sets; in general, the results showed that Bat clustering algorithms were either comparable or outperformed K-means in the validity index, compactness, and separation values. The Bat WCBCR presented the best overall results. The best combination that presented the optimum number of clusters was the Bat WCBCR clustering algorithm and the WCBCR validity index. Together, these presented significantly highly

separated and well-compacted clusters. The main objective of the proposed Bat clustering algorithms to enhance the clustering results and obtain more efficient clustering formations of PVPP data was achieved by the Bat WCBCR clustering algorithm. Thus, the Bat WCBCR can provide well-defined PVPP clusters and cluster representatives that can be utilized in PV power output application studies.

Chapter 5 presented a short-term PV power prediction model. The model was constructed to take advantage of the PVPP representatives. Also, this chapter tested the efficiency of the representative PVPPs resulting from the PVPP clustering methodology discussed in the previous chapters. The results of the prediction model using the PVPP representatives validate the efficiency of our PVPP clustering methodology in PV system studies.

6.2 Future Work

Based on the research presented in this thesis, some of the studies that can be carried out in the future are summarized in the following:

- The development of swarm clustering methods in order to improve the accuracy of the cluster representatives in PV power studies.
- Examination of the use of other features to improve the accuracy of PVPP cluster representatives.
- Use of swarm clustering methods to investigate their efficiency in clustering wind power patterns.
- Use of the cluster representatives in other prediction models, such as Radial Basis Function (RBF) models for short-term PVPP prediction.
- Dividing the daily PVPP into two categories and applying the clustering methods.

Bibliography

[1] G. Farivar, B. Asaei, N. Haghdadi, and H. Iman-Eini, "A novel temperature estimation method for solar cells," *Proc. PEDSTC*, pp. 336-341, 16-17 Feb. 2011.

[2] A. K. Jain and R. C. Dubes*, Algorithms for Clustering Data*, ser. Prentice-Hall Advanced References series. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[3] G. J. McLachlan and S-K. Ng, "Top 10 algorithms in data mining," *Knowledge and Information Systems*, vol.14, no.1, pp. 93-113. 2008.

[4] G. Chicco, R. Napoli, P. Postolache, M. Scutariu, and C. Toader, "Customer characterization options for improving the tariff offer," *IEEE Trans. Power Syst.*, vol. 18, no. 1, pp. 381–387, Feb. 2003.

[5] V. Figueiredo, F. Rodrigues, Z. Vale, and J. B. Gouveia, "An electric energy consumer characterization framework based on data mining techniques," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 596–602, May 2005.

[6] D. Gerbec, S. Gasperic, I. Smon, and F. Gubina, "Determining the load profiles of consumers based on fuzzy logic and probability neural networks," *Proc. Inst. Elect. Eng.*, *Gener. Transm., Distrib.*, vol. 151, no. 3, pp. 395–400, 2004.

[7] G. Chicco, R. Napoli, F. Piglione, P. Postolache, M. Scutariu, and C. Toader, "Load pattern-based classification of electricity customers," *IEEE Trans. Power Syst.*, vol. 19, no. 2, pp. 1232–1239, May 2004.

[8] G. Chicco, R. Napoli, and F. Piglione, "Comparisons among clustering techniques for electricity customer classification," *IEEE Trans. Power Syst.,* vol. 21, no. 2, pp. 933–940, May 2006.

[9] S. V. Verdu, M. O. Garcia, C. Senabre, A. G. Marin, and F. J. G. Franco, "Classification, filtering, and identification of electrical customer load patterns through the use of Self-Organizing Maps," *IEEE Trans. Power Syst.*, vol. 21, no. 4, pp. 1672–1682, Nov. 2006.

[10] J. Tsekouras, N. D. Hatziargyriou, and E. N. Dialynas, "Two-stage pattern recognition of load curves for classification of electricity customers," *IEEE Trans. Power Syst.,* vol. 22, no. 3, pp. 1120–1128, August. 2007.

[11] G. Chicco, O. M. Ionel, and R. Porumb, "Electrical load pattern grouping based on centroid model with ant colony clustering," *IEEE Power Syst.*, vol. 28, no. 2, pp. 1706-1715, May 2013.

[12] G. Chicco, R. Napoli, and F. Piglione, "Load pattern clustering for short-term load forecasting of anomalous days," *Proc. IEEE PowerTech Conf.*, Porto, 10-13 Sept 2001, vol. 2.

[13] A. Gabaldon, A. Guillamon, M. C. Ruiz, S. Valero, C. Alvarez, M. Ortiz, and C. Senabre, "Development of a methodology for clustering electricity-price series to improve customer response initiatives," *IET Gener., Transm., Distrib.*, vol. 4, no. 6, pp. 706–715, 2010.

[14] G. J. Tsekouras, C. A. Anastasopoulos, F. D. Kanellos, V. T. Kontargyri, I. S. Karanasiou, A. D. Salis, and N. E. Mastorakis, "A demand side management program of vanadium redox energy storage system for an interconnected power system," *Proc. WSEAS EPESE,* Corfu Island, Greece, 2008.

[15] G. J. Tsekouras, F. D. Kanellos, V.T. Kontargyri, I. S. Karanasiou, A. D. Salis, and N. E. Mastorakis, "A new classification pattern recognition methodology for power system typical load profiles," *WSEAS Trans. Circuits and Systems*, vol. 12, no. 7, pp. 1090-1104, 2008.

[16] G. J. Tsekouras, I.K. Hatzilau, J. Prousalidis, "A new pattern recognition methodology for classification of load profiles for ships electric consumers," *Journal of Marine Engineering and Technology,* no. A14, pp. 45-58, 2009.

[17] G. Tsamopoulos, N. Giannitsas, F. D. Kanellos, and G. J. Tsekouras, "Load estimation for war-ships based on pattern recognition methods," *Journal of Computations and Modeling*, vol. 4, no. 1, pp. 207-222, 2014.

[18] M. Ali, I. S. Ilie, J. V. Milanovic, and G. Chicco, "Wind farm model aggregation using probabilistic clustering," *IEEE Trans. Power Syst.*, vol. 28, no.1, pp. 309-316, Feb. 2013.

[19] F. J. Duarte, J. M .M. Duarte, S. Ramos, A. Fred, and Z. Vale, "Daily wind power profiles determination using clustering algorithms," *Proc. IEEE Power Syst. Tech.*, pp. 1-6, Oct. 30-Nov. 2 2012.

[20] W. A. Omran, M. Kazerani, and M. M. A. Salama, "A clustering-based method for quantifying the effects of large on-grid PV systems," *IEEE Trans. Power Delivery,* vol. 25, no. 4, pp. 2617-2625, Oct. 2010.

[21] N. Haghdadi, B. Asaei, and Z. Gandomkar, "Clustering-based optimal sizing and siting of photovoltaic power plant in distribution network," *Proc. EEEIC*, pp. 266-271, 18-25 May 2012.

[22] H. Mori, and M. Takahashi, "Application of preconditioned generalized radial basis function network to prediction of photovoltaic power generation," *Proc. IEEE ISGT*, pp.1-6, 14-17 Oct. 2012.

[23] Y. Hosoda, and T. Namerikawa, "Short-term photovoltaic prediction by using H∞ filtering and clustering," *Proc. SICE*, pp.119-124, 20-23 Aug. 2012.

[24] European Photovoltaic Industry Association. (2013). Global Market Outlook: For Photovoltaics 2013-2017. [Online]. Available: http://www.epia.org/fileadmin/user_upload/Publications/GMO_2013_-_Final_PDF.pdf

[25] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recog. Lett*. vol.31, no.8, pp. 651-666. 2010

[26] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, and S. Y. Philip. "Top 10 algorithms in data mining," *Knowledge and Information Systems*, vol.14, no.1, pp. 1-37. 2008.

[27] R. Xu, and D. Wunsch, "Survey of Clustering Algorithms," *IEEE Trans. Neural Networks*, vol. 16, no. 3, pp. 645-678, 2005.

[28] E. A. Mooi and M. Sarstedt, *A Concise Guide to Market Research*, Springer-Verlag Heudelberg, 2011.

[29] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, "The Elements of Statistical Learning: Data Mining: Inference and Prediction," *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83-85, 2005.

[30] D. T. Larose, *Discovering Knowledge in Data: An Introduction to Data Mining*, John Wiley & Sons, Hoboken, NJ, USA, 2005.

[31] I. H. Witten, and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, San Francisco, 2005.

[32] R. Babuska, *Fuzzy and Neural Control DISC Course Notes*, Delft University of Technology, Netherlands, 2009.

[33] A. V. Gawand, P. Lokhande, S. Daware, and U. Kulkarni, " Image Segmentation for Nature Images using K-Mean and Fuzzy C-Mean," *International Journal of Computer Applications*, 2011.

[34] T. Kohonen and T. Honkela, *Kohonen Network*, Scholarpedia, vol. 2, no. 1, 2007. Available online: http://www.scholarpedia.org/w/index.php?title=Kohonen_network&action=cite&rev=122029

[35] M. T. Hagan, H. B. Demuth, and M. Beale, *Neural Network Design*, PWS, Boston, MA, 1996.

[36] T. Kohonen, *Self-Organizing Maps*, 3rd ed. Information Sciences. Berlin, Hiedenberg, Springer, 2001.

[37] V. S. Moertini, "Investigation of self-organizing map and its use in data clustering," *Intergal*, vol. 8, no. 1, pp. 41-52, April 2003.

[38] X. S. Yang, "A new metaheuristic bat-inspired algorithm," Nature Inspired Cooperative Strategies for Optimization (NISCO 2010), Eds. J. R. Gonzalez et al., *Studies in Computational Intelligence*, Springer Berlin, vol. 284, Springer, pp. 65-74, 2010.

[39] P. S. Shelokar, V. K. Jayaraman, and B. D. Kulkarni, "An ant colony approach for clustering," *Analytica Chimica Acta*, vol. 504, pp. 187–195, 2004.

[40] P. S. Shelokar, V. K. Jayaraman, and B. D. Kulkarni, "An ant colony approach for clustering," *Analytica Chimica Acta*, vol. 504, pp. 187–195, 2004.

[41] L. Jiang, L. Ding, Y. Peng, and C. Zhao, "An efficient clustering approach using ant colony algorithm in multidimensional search space," *Proc. FSKD*, vol.2, pp. 1085-1089, 26-28 July 2011.

[42] A. P. Engelbrecht, *Computational Intelligence: An Introduction*, 2nd Edition, John Wiley & Son, 2007, pp. 135-137.

[43] R. Tang, S. Fong, X.-S. Yang, and S. Deb, "Integrating nature-inspired optimization algorithms to k-means," *Proc. ICDIM*, pp. 116-123, 22-24 Aug. 2012.

[44] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "Cluster validity methods: part I," *ACM SIGMOD Record*, vol. 31, no. 2, pp. 40-45, June 2002.

[45] N. Bolshakova and F. Azuaje, "Cluster validation techniques for genome expression data", *Signal Processing*, vol. 83, pp. 825-833, 2002.

[46] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no.2, pp. 224-227, April 1979.

[47] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *J. Cybren.*, vol. 3, pp. 32-57, 1973.

[48] R. Jain, and R. Koronios, "Innovation in the cluster validating techniques," *Fuzzy Optimization and Decision Making*, vol. 7, no. 3, pp. 233–241, 2008.

[49] Q. Zhao, M. Xu, and P. Franti, "Knee point detection on Bayesian information criterion," *Proc. IEEE ICTAI*, vol.2, pp. 431-438, 3-5 Nov. 2008.

[50] R. E. Kass, and L. Wasserman, "A reference Bayesian test for nested Hypotheses and its relationship to the Schwarz Criterion," *Journal of the American Statistical Association*, vol. 90, no. 431, pp. 928-934, 1995.

[51] X. Xie and G. Beni, "A Validity Measure for Fuzzy Clustering," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp.841 - 847, 1991.

[52] Z. Liang, P. Zhang, and J. Zhao, "Optimization of the number of clusters in fuzzy clustering," *Proc. ICCDA*, vol. 3, pp. 580-584, 2010.

[53] D. Hand, H. Manilla, and, P. Smyth, *Principles of Data Mining*, Cambridge, MA: MIT Press, 2001.

[54] I. T. Jolliffe, *Principal Component Analysis*, Springer-Verlag New York, Inc., Second Edition, 2002.

[55] C. Chatfield, and A. J. Collins, *Introduction to Multivariate Analysis*, Chapman and Hall, New York, 1980.

[56] W. Omran, "Performance analysis of grid-connected photovoltaic systems," Ph. D. dissertation, ECE, Waterloo, ON, 2010.

[57] G. Vijayakumar, M. Kummert, S. Klein, W. Beckman, "Analysis of short-term solar radiation data," *Solar Energy,* vol.79, pp. 495–504, 2005.

[58] C. Craggs, E. M. Conway and N. M. Pearsall, "Statistical investigation of the optimal averaging time for solar irradiance on horizontal and vertical surfaces in the UK," *Solar Energy*, vol.68, pp. 179–187, 2000.

[59] G. Kopp, and J. L. Lean, "A new, lower value of total solar irradiance: evidence and climate significance," *Geophys. Res. Lett*., vol. 38, 2011.

[60] G. M. Masters, *Renewable and Efficient Electric Power Systems*, John Wiley & Son, 2004, pp. 505-531.

[61] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. Perez, and I. Perona, "An extensive comparative study of cluster validity indices," *Pattern Recognition*, vol. 46, pp. 243-256, 2013.

[62] Solar Radiation Research Laboratory (BMS) available online at: http://www.nrel.gov/midc/srrl_bms. Retrieved January 2014.

[63] SUNPOWER, "E20/435 Solar Panel," SPR-435NE-WHT-D datasheet, 2011 [Revised Jan. 2014].

[64] J. Branke, K. Deb, H. Dierolf, and M. Oswald, "Finding knees in multi-objective optimization," *in Proc. Int. Conf. Parallel Problem Solving from Nature*, vol. 3242, pp. 722-731, 2004.

[65] J. Zeng, and W. Qiao, "Short-term solar power prediction using a support vector machine," *Renewable Energy*, vol. 52, pp. 118-127, 2013.

[66] V. Badescu, *Modeling solar radiation at the earth surface*, Springer, 2008.

[67] A. Koca, H. F. Oztop, Y. Varol, and G. O. Koca, "Estimation of solar radiation using artificial neural networks with different input parameters for mediterranean region of anatolia in turkey," *Expert Systems with Applications*, vol. 38, pp. 8756-8762, 2011.

[68] S-H. Cao, J-B. Chen, W-B. Weng, and J-C. Cao, "Study of daily solar irradiance forecast on chaos optimization neural networks," *Natural Science*, vol. 1, pp. 30-36, 2009.

[69] G. Reikard, "Predicting solar radiation at high resolutions: a comparison of time series forecasts," *Solar Energy*, vol. 83, pp. 342-349, 2009.

Appendix A

Angle-Based Knee Detection Method

Given a fixed number of clusters $K \geq 2$ and a clustering algorithm, finding the clustering that best fits the data set involves the following steps:

1- Select a proper cluster validity index.

2- Repeat a clustering algorithm successively for number of clusters, K from a pre-defined minimum to a pre-defined maximum.

3- Plot the "number of clusters vs. criterion metric" graph.

4- Calculate the difference between previous and afterward index values:

$$DiffFun(k) = F(k-1) + F(k+1) - F(k). \tag{A.1}$$

5- Select n local significant changes and calculate the angle of those points:

$$Angle(k) = a\tan(1/|F(k) - F(k-1)|) + a\tan(1/|F(k+1) - F(k)|). \tag{A.2}$$

6- Select k that has the largest angle as a knee point.

Appendix B

The Clustering Results for Summer, Spring, and Winter Seasons

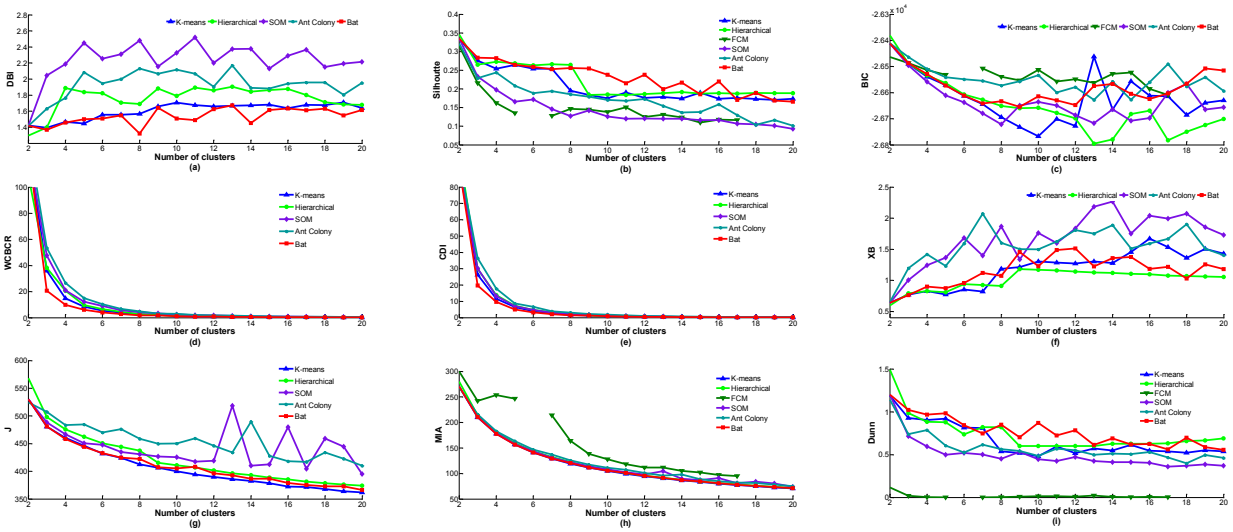

Figure B.1: The best results of each clustering method for the Summer data set of PVPPs for two to 20 clusters: (a) DBI (b) SI (c) BIC (d) WCBCR (e) CDI (f) XB (g) J (h) MIA (i) Dunn.

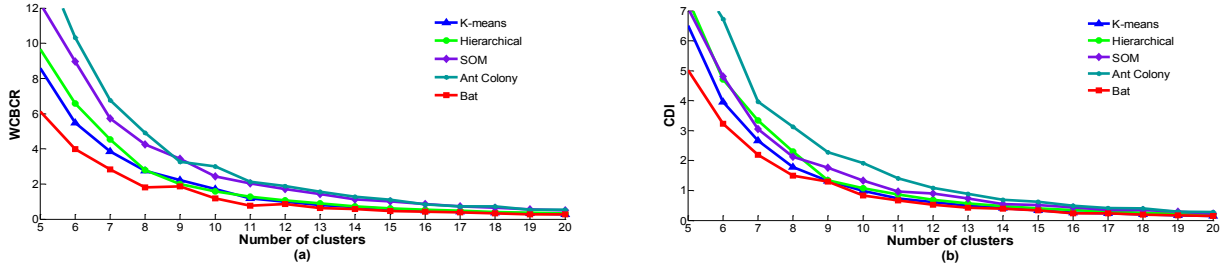

Figure B.2: The best results of each clustering method for the Summer data set of PVPPs for five to 20 clusters: (a) WCBCR (b) CDI.

Table B.1: Comparison of clustering algorithms for ten clusters (summer).

Validity index	DBI	Dunn	SI	BIC $\times 10^4$	WCBCR	CDI	XB	J	MIA
K-means	1.706	0.481	0.174	-2.676	1.711	0.991	1.304	*400.10*	*105.08*
Hierarchical WMV	1.790	0.600	0.185	-2.665	1.584	1.079	*1.169*	410.98	106.50
FCM	44.951	0.014	0.138	-2.651	37.61	7.188	159.859	593.49	127.98
SOM	2.327	0.445	0.126	*-2.663*	2.438	1.329	1.760	425.54	108.37
Ant Colony	2.116	0.484	0.170	-2.653	3.004	1.913	1.499	450.23	111.47
Bat	*1.507*	*0.870*	*0.238*	-2.661	*1.185*	*0.834*	1.228	405.27	105.76

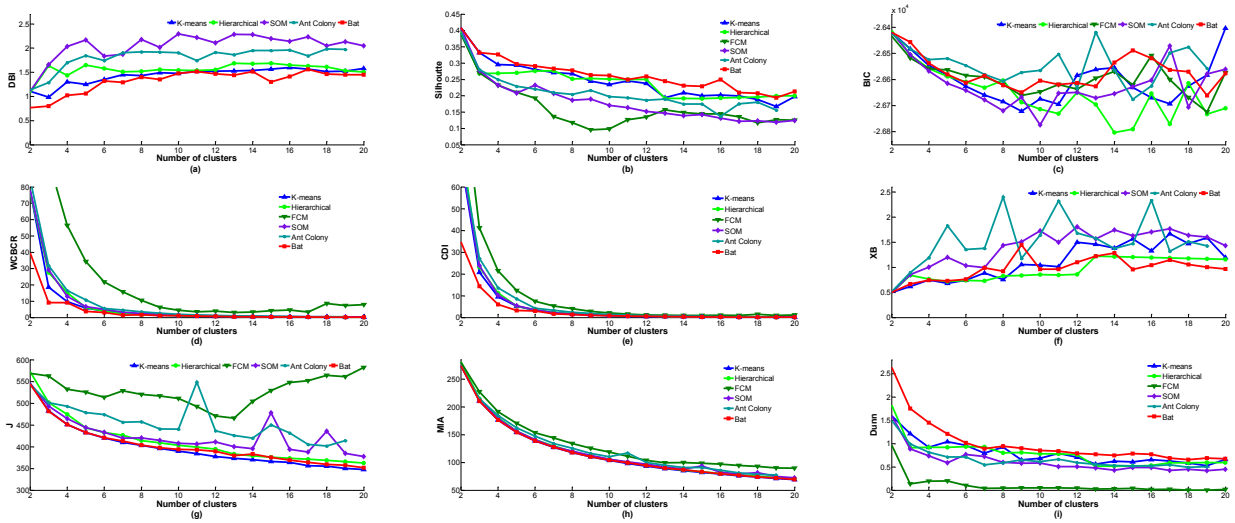

Figure B.3: The best results of each clustering method for the Spring data set of PVPPs for two to 20 clusters: (a) DBI (b) SI (c) BIC (d) WCBCR (e) CDI (f) XB (g) J (h) MIA (i) Dunn.

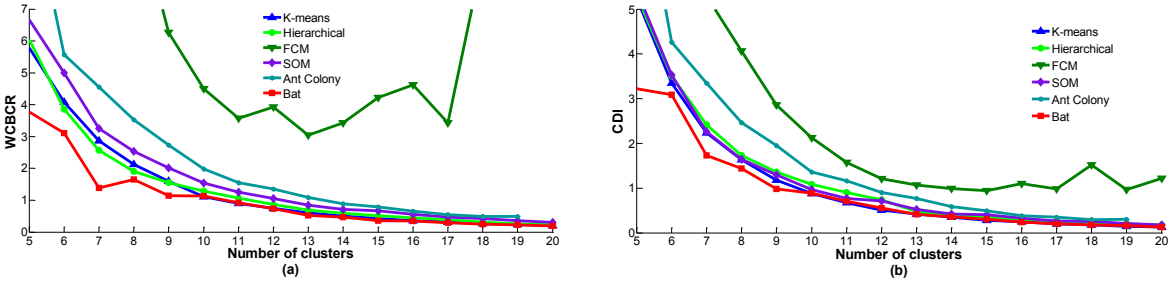

Figure B.4: The best results of each clustering method for the Spring data set of PVPPs for five to 20 clusters: (a) WCBCR (b) CDI.

Table B.2: Comparison of clustering algorithms for seven clusters (spring).

Validity index	DBI	Dunn	SI	BIC $\times 10^4$	WCBCR	CDI	XB	J	MIA
K-means	1.448	0.794	0.241	-2.6659	2.864	2.239	0.887	*410.18*	*127.17*
Hierarchical WMV	1.511	*0.931*	0.276	-2.6631	2.563	2.417	*0.732*	426.38	129.65
FCM	14.938	0.041	0.136	-2.6590	15.671	5.302	57.879	528.54	144.35
SOM	1.871	0.725	0.207	-2.6678	3.250	2.259	0.995	419.67	128.63
Ant Colony	1.904	0.544	0.209	*-2.6581*	4.550	3.344	1.379	456.32	134.13
Bat	*1.291*	0.894	*0.283*	-2.6586	*1.386*	*1.730*	0.991	413.09	127.62

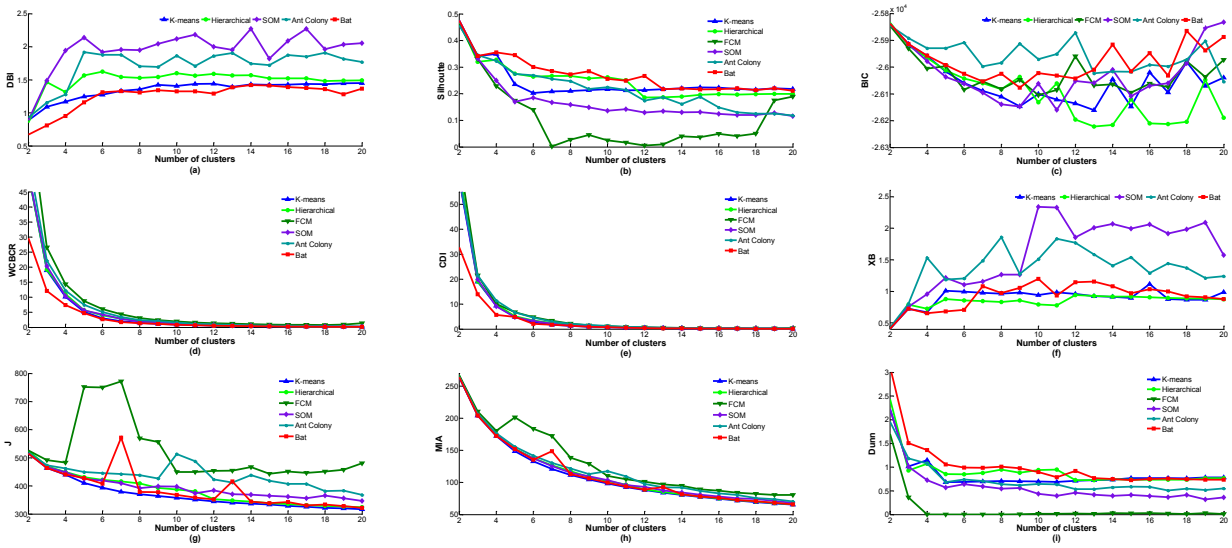

Figure B.5: The best results of each clustering method for the winter data set of PVPPs for 2 to 20 clusters: (a) DBI (b) SI (c) BIC (d) WCBCR (e) CDI (f) XB (g) J (h) MIA (i) Dunn.

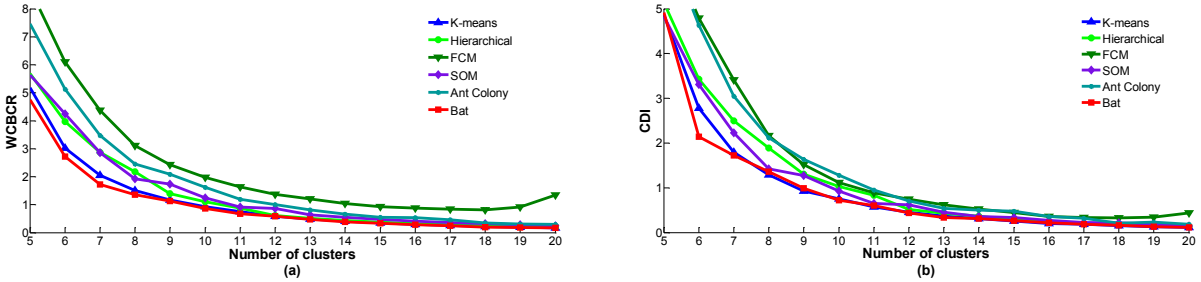

Figure B.6: The best results of each clustering method for the winter data set of PVPPs for five to 20 clusters: (a) WCBCR (b) CDI.

Table B.3: Comparison of clustering algorithms for ten clusters (winter).

Validity index	DBI	Dunn	SI	BIC $\times 10^4$	WCBCR	CDI	XB	J	MIA
K-means	1.406	0.699	0.216	-2.610	0.923	0.751	0.940	*357.99*	*98.31*
Hierarchical WMV	1.601	0.941	0.262	-2.613	1.098	1.029	*0.791*	387.19	102.24
FCM	19.974	0.014	0.024	-2.610	1.977	1.117	155.477	449.85	110.20
SOM	2.118	0.439	0.135	-2.606	1.237	0.926	2.338	397.49	103.59
Ant Colony	1.865	0.645	0.225	*-2.597*	1.621	1.282	1.510	513.28	117.72
Bat	*1.326*	*0.896*	*0.255*	-2.602	*0.875*	*0.722*	1.198	369.40	99.73

Appendix C

Parameters of Bat Clustering Algorithms

Table C.1: Parameters of Bat Clustering Algorithms.

Parameter	Value
B (bat population)	20
A_0 (loudness)	0.5
r (pulse rate)	0.5
f_{min} (minimum frequency)	0
f_{max} (maximum frequency)	0.9
M (Iterations)	50

Appendix D

The best results of each clustering algorithm for the summer PVPP of the second data set.

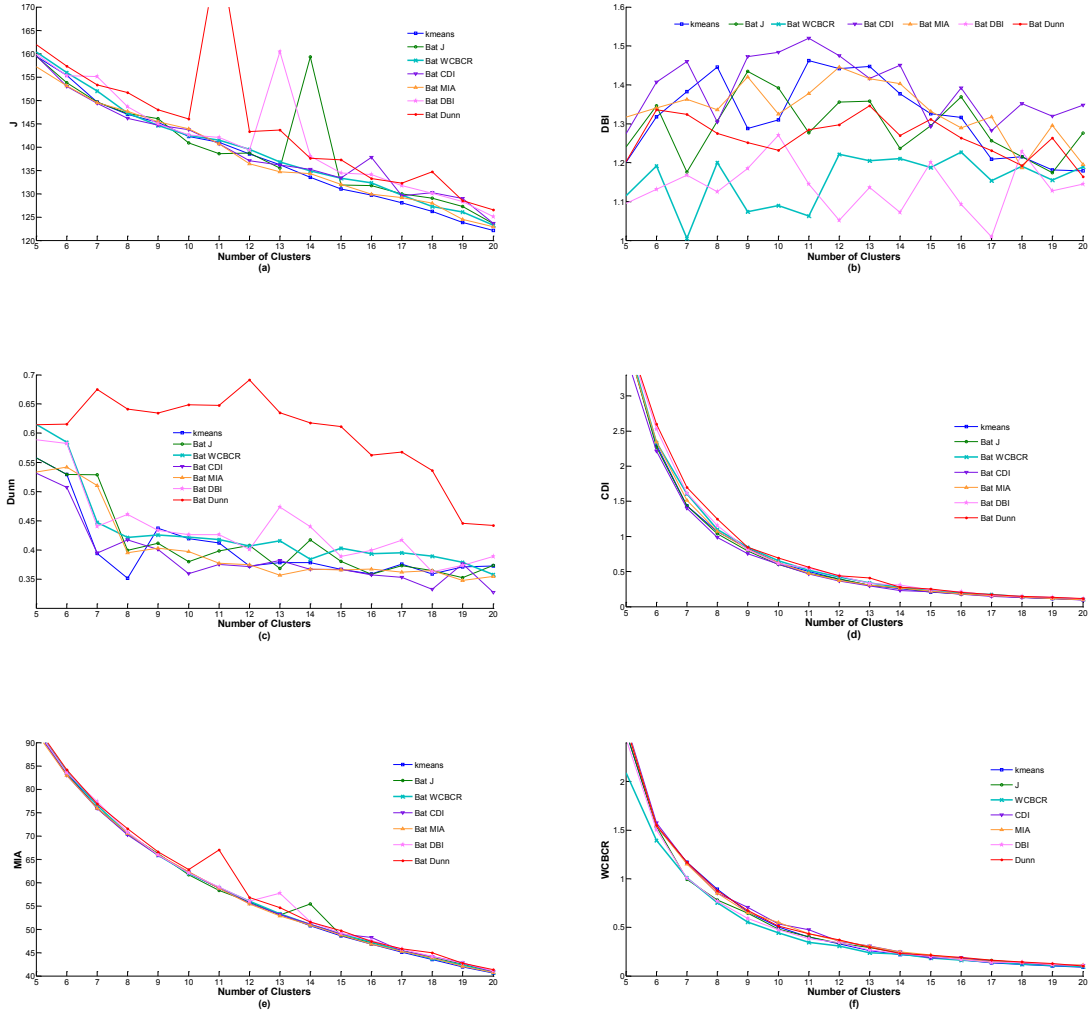

Figure D.1: The best results of each clustering algorithm for the summer PVPP of the second data set for five to 20 clusters: (a) J (b) DBI (c) Dunn (d) CDI (e) MIA (f) WCBCR.

Appendix E

MRSE, MAE, and Correlation Coefficient

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_{observed} - P_{predicted})^2} \tag{E.1}$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|P_{observed} - P_{predicted}\right| \tag{E.2}$$

$$\text{Correlation Coefficient} = \frac{\sum_{i=1}^{N}\left(P_{observed} - \mu_{observed}\right).\left(P_{predicted} - \mu_{predicted}\right)}{\sqrt{\sum_{i=1}^{N}\left(P_{observed} - \mu_{observed}\right)^2}\sqrt{\sum_{i=1}^{N}\left(P_{predicted} - \mu_{predicted}\right)^2}} \tag{E.3}$$

where $P_{observed}$ and $P_{predicted}$ are the observed and predicted daily PVPPs, respectively; $\mu_{observed}$ and $\mu_{predicted}$ are the mean values of $P_{observed}$ and $P_{predicted}$, respectively.