University Library



A gateway to Melbourne's research publications

Minerva Access is the Institutional Repository of The University of Melbourne

Author/s:
Fahiman, Fateme

Title:
Pattern Recognition and Predictive Modelling in Smart Grids

Date:
2019

Persistent Link:
https://hdl.handle.net/11343/240577

Terms and Conditions:
Terms and Conditions: Copyright in works deposited in Minerva Access is retained by the copyright owner. The work may not be altered without permission from the copyright owner. Readers may only download, print and save electronic copies of whole works for their own personal non-commercial use. Any use that exceeds these limits requires permission from the copyright owner. Attribution is essential when quoting or paraphrasing from these works.

Pattern Recognition and Predictive Modelling in Smart Grids

Fateme Fahiman

Submitted in total fulfilment of the requirements of the degree of

Doctor of Philosophy

December 2019

School of Computing and Information Systems
THE UNIVERSITY OF MELBOURNE

Copyright © 2019 Fateme Fahiman

All rights reserved. No part of the publication may be reproduced in any form by print, photoprint, microfilm or any other means without written permission from the author except as permitted by law.

Pattern Recognition and Predictive Modelling in Smart Grids

Fateme Fahiman
Supervisors: Prof. Christopher Leckie, Dr.Sarah M.Erfani

Abstract

Smart grids are a modification of traditional electric power grid to achieve a bidirectional, automatic, intelligent and adaptive power system. In smart grids the electricity distribution and power system management is improved by leveraging advanced two-way communications and integrating computing capabilities to achieve better reliability, stability, efficiency, and security of the power system. The smart grid introduces the two-way flow of data between electricity suppliers and customers to transfer real-time information and facilitate the near real-time balance of supply-demand management. In contrast to many other industries who have the capability to store and reserve their products, the electric power industry does not have such a capability to store a massive amount of electricity using today's technologies. Therefore, due to the storage limitations of electricity, one of the crucial tasks of power system operation is to keep a balance between supply and demand at every moment. As a result, forecasting is an essential and important function in the electricity power grid.

Recent advances in the energy industry, including smart grid and smart meters, provide new capabilities to electrical utilities for forecasting electricity demand, modelling customers' usage profiles, optimizing unit commitment and preventing outages. These advances also introduce new challenges to the power grid, such as managing and analysing of large volumes of complex, high dimensional data in an efficient manner. So, utilities need to apply advanced data management and analytical models to extract actionable insights from this information. By leveraging better predictive and analytical models and the high volume of data, utility companies are able to produce a wide range of forecasts including:

1. Forecasting the amount of excess energy generation, the appropriate time to sell it, and the feasibility of transmitting it into the grid
2. Forecasting when and where contingencies are most likely to happen
3. Identifying the customers that are most likely to transfer energy back to the grid
4. Identifying the customers that are most likely to respond to demand reduction incentives and energy conservation programs
5. Considering the generation of distributed energy resources in the decision making process to manage the commitment of conventional plants
6. Considering the integration of renewable energy resources into the power grid, which are inherently intermittent, weather dependent and unpredictable, to run a clean and reliable power system.

To achieve these potential benefits, grid operators require accurate and efficient methods to mine patterns in customers and grid data, which can be integrated into their decision making frameworks. In this thesis, we develop new predictive machine learning algorithms to help address the new challenges in the smart grid era.

In the first part of this thesis, we focus on understanding customers' energy consumption behaviour (demand analytics). Previously, information about customers' energy consumption could be obtained only with coarse granularity (e.g., monthly or bi-monthly), Nowadays, using advanced metering infrastructure (or smart meters), utility companies are able to retrieve it in near real-time. By leveraging smart meter data, we propose a hierarchical demand forecasting approach. We improve the aggregated level of electricity load forecasts by first segmenting the households into several clusters, forecasting the energy consumption of each cluster, and then aggregating those forecasts. The improvements provided by this strategy depend not only on the number of clusters, but also on the size of the clusters and selecting an appropriate clustering method. We also leverage deep learning techniques to improve forecast accuracy.

Dealing with the high volume of time-series data (smart meter data) has motivated us to develop a new clustering algorithm for time-series data that is computationally efficient and accurate. In the second part of this thesis, we introduce our two new proposed clustering algorithms namely, "Fuzzy C-Shape plus (FCS+)" and "Fuzzy C-Shape plus plus (FCS++)" and we show that the two new algorithms outperform state-of-the-art shape-based clustering algorithms in terms of accuracy and efficiency.

Improving accuracy is a primary goal in any forecasting task, which is especially challenging in multi-step prediction scenarios. In the third part of this thesis we propose a robust and accurate ensemble based load forecasting framework to address some of the challenges associated with load forecasting, including unbalanced training load data, the non-stationary nature of the load data, and feature selection for predictive modelling. The performance of the proposed method is validated with real-life data from the power system in the Australian National Electricity Market, as well as through on-site implementation by the system operator.

In practice, an understanding of the uncertainty in the forecasts that an operational power grid uses is crucial in order to operate the system in a secure manner in real-time and into the future. In the fourth part of this thesis, we propose a dynamic stochastic decision support tool, based on Dynamic Bayesian Belief Networks, to quantify the level of uncertainty in order to improve situational awareness and understand the risks to power system operators. The performance of the proposed method is validated on real-life data from the Australian power system, and through actual on-site implementation by the Australian system operator, the Australian Energy Market Operator.

Declaration

This is to certify that

1. the thesis comprises only my original work towards the PhD,
2. due acknowledgement has been made in the text to all other material used,
3. the thesis is less than 100,000 words in length, exclusive of tables, maps, bibliographies and appendices.

Fateme Fahiman, December 2019

Preface

The core chapters of this thesis, Chapters 3 to 7, are based on the following papers. I declare that for each paper I am the main author who has contributed over 50%. Chapters 3 to 7 are respectively based on the articles **P1**, **P2**, **P3** and **P4**:

Published Papers

- **P1. Fateme Fahiman**, Sarah M.Erfani, Sutharshan Rajasegarar, Marimuthu Palaniswami, Christopher Leckie, Improving load forecasting based on deep learning and K-shape clustering. *in 2017 International Joint Conference on Neural Networks (IJCNN)*, 2017 May 14 (pp. 4134-4141). IEEE.

- **P2. Fateme Fahiman**, James C. Bezdek, Sarah M. Erfani, Marimuthu Palaniswami, Christopher Leckie, Fuzzy c-Shape: A New Algorithm for Clustering Finite Time Series Waveforms. *in 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2017 Jul 9 (pp. 1-8). IEEE. (The paper achieved the **Best student paper award** at IEEE International Conference on Fuzzy Systems, 2017)

- **P3. Fateme Fahiman**, Sarah M. Erfani, Christopher Leckie, Robust and Accurate Short-Term Load Forecasting: A Cluster Oriented Ensemble Learning Approach. *in 2019 International Joint Conference on Neural Networks (IJCNN), IEEE*, 2019 IEEE, (July 2019)

- **P4. Fateme Fahiman**, Steven Disano, Sarah M. Erfani, Pierluigi Mancarella, Christopher Leckie, Data Driven Dynamic Probabilistic Reserve Sizing Based on Dynamic Bayesian Belief Networks. *IEEE Transactions on Power Systems*, 2018 Dec 3;34(3):2281-91

Acknowledgements

I would like to express my deepest appreciation to those who have contributed to this thesis and supported me during this amazing journey.

I am extremely grateful to my supervisors, Professor Christopher Leckie and Dr. Sarah Erfani, for their warm support, inspiring and guiding my way through this unpredictable PhD journey. I am sincerely grateful for their knowledge and valuable advice and all their insightful discussions and suggestions during these 4 years.

I am also profoundly grateful for the team work of my co-authors and their substantial contribution to uplift the studies presented in this thesis. Thanks, Professor James C. Bezdek, Professor Pierluigi Mancarella, Dr. Sutharshan Rajasegarar and Mr. Steven Disano. I am delighted to have worked with you and I look forward to working with you again.

My sincere thanks goes to my PhD advisory Professor Marimuthu Palaniswami and Professor Harald Sandergaad for their helpful and practical suggestions on my research.

I wish to express my sincerest thanks, love and deep appreciation to my family and friends. This journey would have never happened without the unconditional love and support of my love, best friend and soulmate, Ehsan. I am deeply thankful to my beloved mother and father who always believed in me and encouraged me to pursue my education towards a PhD. I am grateful for the encouragement, love and support I received from my brother, Meisam, and my sisters, Azadeh and Sepideh.

Fateme Fahiman
Melbourne, Australia
3 December 2019

Contents

List of Symbols and Abbreviations **xix**

1 Introduction **1**
1.1 Developing Reliable and Sustainable Energy Systems . . . 1
1.2 Smart Grid . . . 3
1.2.1 Smart Grid Applications and Services Road-Map . . . 4
1.3 Role of Computing and Analytics . . . 9
1.3.1 Supply Analytics . . . 9
1.3.2 Network Analytics . . . 9
1.3.3 Demand Analytics . . . 10
1.4 Motivation for Forecasting in Power Grids . . . 10
1.5 Research Challenges Addressed in This Thesis . . . 11
1.5.1 Short-Term Load Forecasting Based on Big Data (Smart Meters) . . . 11
1.5.2 Efficient and Accurate Time-Series Clustering Method for Load Profiling . . . 14
1.5.3 Short-Term Load Forecasting Framework for Nonstationary and Unbalanced Data . . . 15
1.5.4 Managing Uncertainties in Dynamic Reserve Sizing . . . 16
1.6 Thesis Organization and Contributions . . . 17

2 Background and Literature Review **23**
2.1 Introduction . . . 23
2.2 Load Forecasting in Power Grids . . . 24
2.2.1 Horizon . . . 24
2.2.2 Process . . . 25
2.2.3 Modelling . . . 25
2.2.4 Requirements . . . 26
2.3 Load Forecasting Methodologies . . . 27
2.3.1 Feature Selection . . . 28
2.3.2 Similar Day Method . . . 28
2.3.3 Weather Station Selection . . . 28
2.3.4 Hierarchical Load Forecasting . . . 29
2.3.5 Load Profiling and Clustering Based Methods . . . 30
2.4 Load Forecasting Techniques . . . 33
2.4.1 Statistical Techniques . . . 33
2.4.2 AI Based Techniques . . . 37

2.5 Managing Uncertainties In Smart Grids . . . 40
2.5.1 Probabilistic Load Flow . . . 41
2.5.2 Contingency Analysis . . . 42
2.5.3 Unit Commitment . . . 43
2.5.4 Reliability Planning . . . 44
2.5.5 Reserve Sizing . . . 45
2.6 Conclusion . . . 46

3 Improving Load Forecasting Based on Deep Learning and K-shape Clustering 47
3.1 Introduction . . . 47
3.2 Background on Short-Term Load Forecasting . . . 49
3.2.1 Problem Statement . . . 49
3.2.2 Feature Selection . . . 52
3.3 Grouping Smart Metering Data Based on Consumption Profiles . . . 54
3.3.1 K-shape Clustering . . . 55
3.4 Deep-Learning vs Shallow Neural Network . . . 59
3.4.1 Artificial Neural Networks . . . 60
3.4.2 Deep-Learning Based on Restricted Boltzman Machine and Deep Belief Networks . . . 61
3.5 Case Study Applications . . . 63
3.5.1 Input Data . . . 63
3.5.2 Experimental Results . . . 63
3.6 Conclusion . . . 66

4 Fuzzy c-Shape: A New Algorithm for Clustering Finite Time-Series Waveforms 69
4.1 Introduction . . . 69
4.2 Related Work . . . 72
4.3 Preliminaries . . . 74
4.3.1 The Generalized Fuzzy c-Means Model . . . 75
4.3.2 Shape Based Distance . . . 78
4.3.3 Shape Based Prototypes and k-Shape Clustering . . . 80
4.4 FCS+ and FCS++ Clustering . . . 83
4.5 Cluster Validity Indices . . . 85
4.6 Numerical Experiments . . . 88
4.7 Discussion . . . 93
4.8 Conclusion . . . 96

5 Robust and Accurate Short-Term Load Forecasting Framework 97
5.1 Introduction . . . 97
5.2 Background . . . 100
5.2.1 Problem Statement . . . 100
5.2.2 Benchmark Model . . . 101
5.3 Methodology . . . 106
5.3.1 Apparent Temperature - A Novel Feature for Improving Load Forecast Accuracy . . . 106
5.3.2 Our Proposed Framework . . . 107
5.4 Case Studies . . . 113

5.4.1 Data . . . 113
5.4.2 Experiments . . . 113
5.5 Conclusion . . . 116

6 Data-Driven Dynamic Probabilistic Reserve Sizing Tool **119**
6.1 Introduction . . . 119
6.1.1 Background and Literature Review . . . 119
6.1.2 Novelty and Contributions . . . 122
6.2 Background on Bayesian Belief Networks . . . 123
6.2.1 Bayesian Belief Networks Formulation . . . 123
6.2.2 Dynamic Bayesian Belief Networks . . . 126
6.3 Methodology . . . 128
6.3.1 Problem Statement . . . 128
6.3.2 DBBN Model to Forecast Available Reserve Uncertainties . . . 130
6.3.3 Sensitivity Analysis . . . 131
6.3.4 Lack of Reserve Conditions . . . 132
6.3.5 Forecast Uncertainty Measure (FUM) . . . 133
6.4 Case Study Applications . . . 135
6.4.1 Input Data . . . 135
6.4.2 Pre-processing, Training, and Testing Model . . . 137
6.4.3 Dynamic Forecasting Uncertainty Measure . . . 139
6.4.4 Selection of Confidence Level . . . 142
6.5 Conclusion . . . 143

7 Conclusions and Future Work **145**
7.1 Conclusions . . . 145
7.2 Future Work . . . 147

List of Figures

1.1 Traditional power grid vs smart power grid [3]. 3
1.2 Registered capacity and maximum demand in South Australia 2008 to 2017 [6]. 3
1.3 Smart Grid framework presenting the potential applications, information technology, communication and energy infrastructure of the overall system. 5
1.4 A typical Smart Grid framework [228, 231] 6
1.5 Smart Grid application road-map from Automated Meter Reading (AMR) to Advanced Metering Infrastructure (AMI) to Smart Grids [82]. 7
1.6 Example of power consumption time-series at an individual household level (left) and at an aggregate utility level (right). 13
1.7 Thesis structure and connections between chapters 18

2.1 A typical short-term load forecasting process 25

3.1 Different representative methods for network-level load forecasting. . . . 51
3.2 Mean absolute percentage error achieved using different types of features 52
3.3 Mean, Mean plus standard deviation, and Mean minus standard deviation value for each cluster with applying K-means clustering method. 56
3.4 Mean, Mean plus standard deviation, and Mean minus standard deviation value for each cluster with applying K-shape clustering method. 56
3.5 Average MAPE for one month (November 2010-December 2010) vs lead times of 30 min ahead, 1 h ahead, 2 h ahead, ..., 24 h ahead for different numbers of cluster. 65
3.6 Average MAPE for one month (November 2010-December 2010) vs lead times of 30 min ahead, 1h ahead, 2h ahead, ..., 24h ahead for different numbers of cluster. 65
3.7 Average MAPE for one month (November 2010-December 2010) vs lead times of 30 min ahead, 1h ahead, 2h ahead, ..., 24h ahead for different proposed methods. 65
3.8 Normalized load profiles extracted by K-shape clustering when K=5, the optimal number of clusters. 66

4.1 Comparisons of k-Shape, FCS+ and FCS++. Average RI values of ten runs over 48 data sets. 90

4.2 Comparison for rankings of clustering accuracy methods for 3 metrics. The bars represent average rankings based on the Friedman test, and the number on the top of the bars indicates the ranking of the algorithm, from the best (1) to the worst (3) for each given measure. The ranking is determined for all datasets and finally an average is calculated as the mean of all rankings. 93
4.3 Objective functions for algorithms k-Shape, FCS+, and FCS++, are not always monotone decreasing on successive iterates. 95

5.1 Average demand for each day of week in Victoria during 2016 105
5.2 Ten top important features ranked by xGBoost for benchmark model . . . 105
5.3 (a) The cluster oriented ensemble learning framework for robust and accurate day ahead load forecasting. (b) Horizontal ensemble learning system based on XGBoost predictors. 109
5.4 The top figure shows the architecture of an LSTM block and the bottom one shows the unrolled LSTM sequential architecture 112
5.5 Monthly analysis of day-ahead forecast accuracy in terms of MAPE during the test period 2018-01-01 to 2018-12-28, in Victoria 115
5.6 Actual vs. day-ahead forecast by Framework model for period 2018-01-01 to 2018-01-14, in Victoria 115
5.7 Day-ahead forecast Absolute Error box plot during the extreme weather conditions in the test period 2018-01-01 to 2018-12-28, in Victoria 116
5.8 Day-ahead forecast Absolute Error box plot during the normal weather conditions in the test period 2018-01-01 to 2018-12-28, in Victoria 116

6.1 Graphical representation of a BBN with parent and child nodes 124
6.2 Static BBN vs Dynamic BBN 127
6.3 Sample DBBNs using seven years of historical data (2010-2017) in South Australia 131
6.4 *LOR* notices in the national electricity market in Australia, using the revised formulation of *LOR* based on *FUM* 134
6.5 Architecture of forecasting model for NSW for 6 hours ahead 138
6.6 Top graph shows the distribution in *MW* of forecasting error across all the regions in Australia including the error produced by intermittent generation forecast, conventional generators maximum availability forecast, and regional demand forecast for the 12*hrs* ahead forecasting error. Bottom graph shows the joint distribution of the top distributions extracted from the DBBN for 12*hrs* ahead forecasting error 140
6.7 Typical error distributions for a range of forecasting horizons in Victoria using almost 7 years of historical data (January 2010-July 2017) 141
6.8 Comparison of Actual FUM value with the Dynamic FUM (produced using DBBN) and the Static FUM (produced using deterministic method) in three consecutive run times in New South Wales, Australia. 142

List of Tables

1 A list of symbols and abbreviations which refers to the content of this thesis. xix

3.1 Mean Absolute Percentage Error for different clustering and forecasting methods for lead time 24hours ahead in November 2010 - December 2010 66

4.1 Grand Average of (480) CVI values (10 runs per data set x 48 data sets) . . 89
4.2 Average run time for 480 values of the Rand Index (48 data sets, 10 runs each) 91
4.3 Accuracy comparison among three clustering method using Wilcoxon test based on RI validity method. R^+ corresponds to the sum of the ranks for the method on the left and R^- for the right 91
4.4 Accuracy comparison among three clustering method using Wilcoxon test based on ARI validity method. R^+ corresponds to the sum of the ranks for the method on the left and R^- for the right 92
4.5 Accuracy comparison among three clustering method using Wilcoxon test based on NMI_M validity method. R^+ corresponds to the sum of the ranks for the method on the left and R^- for the right 92
4.6 Presumptive Objective Function Components 94

5.1 MAPE value for day ahead forecast during test period (”2018-01-01” to ”2018-12-28”) for each five states in Australia 114

6.1 CPT of the variable D given the information of variable C. 125
6.2 FUM values for the horizon of 6hrs and 24hrs ahead in Victoria for a range of scenarios. 135
6.3 The order of input variable importance for each time horizon of target variable $(Error(RXS))$ 138
6.4 Confidence levels for determination of FUM values 143
6.5 The number of LOR2 conditions declared by FUM and LCC1 criteria. Note that for all regions the FUM declared by LOR2 was one day earlier than LCC1 143

List of Symbols and Abbreviations

Table 1: A list of symbols and abbreviations which refers to the content of this thesis.

Symbol	Description
DGS	Distributed Generation Systems
RES	Renewable Energy Systems
SG	Smart Grid
AMI	Advanced Metering Infrastructure
AMR	Automatic Meter Reading
FACTS	Flexible AC Transmission Systems
HVDC	High-Voltage Direct Current
HTS	High Temperature Superconductors
DLR	Dynamic Line Rating
PEVs	Plug-in Electric Vehicles
PHEVs	Plug-in Hybrid Electric Vehicles
EIA	Energy Information Administration
DR	Demand Response
AR	Auto-Regressive
ARMA	Auto-Regressive with Moving Average
ARIMA	Auto-Regressive Integrated Moving Average
DBN	Deep Belief Nets
RBM	Restricted Boltzmann Machines
FCS+	Fuzzy C-Shape plus
FCS++	Fuzzy C-Shape plus plus
FCM	Fuzzy C-Means

Continued on next page

Table 1 – *Continued from previous page*

Symbol	Description
HCM	Hard C-Means
RES	Renewable Energy Sources
RI	Rand Index
ARI	Adjusted Rand Index
VI	Variation of Information
NMI	Normalized Mutual Information
CVIs	Cluster Validity Indices
DBBNs	Dynamic Bayesian Belief Networks
FUM	Forecasting Uncertainty Measurement
AEMO	Australian Energy Market Operator
STLF	Short-Term Load Forecasting
AI	Artificial Intelligence
GEFCom2012	Global Energy Forecasting Competition 2012
SOM	Self-Organizing Maps
DPMM	Dirichlet Process Mixture Model
PCA	Principal Component Analysis
GMM	Gaussian Mixture Model
MLR	Multiple Linear Regression
ANN	Artificial Neural Networks
SVMs	Support Vector Machines
MTLF	Medium Term Load Forecasting
NTP	Network Topology Process
BBNs	Bayesian Belief Networks
MAPE	Mean Absolute Percentage Error
SBD	Shape Based Distance
MLP	Multi-Layer Perceptron
DTW	Dynamic Time Warping
ED	Euclidean Distance

Continued on next page

Table 1 – *Continued from previous page*

Symbol	Description
LCSS	Longest Common Sub-Sequence
ERP	Edit Distance with Real Penalty
EDR	Edit Distance on Real sequence
MVM	Minimal Variance Matching
Swale	Sequence Weighted Alignment
cDTW	constrained Dynamic Time Warping
STS	Short Time-Series
GLSE	Generalized Least Squares Error
AO	Alternating Optimization
FONCs	First Order Necessary Conditions
CC	Cross Correlation
RNN	Recurrent Neural Network
LSTM	Long Short Term Memory
XGBoost	eXtreme Gradient Boosting
CART	Classification And Regression Trees
AT	Apparent Temperature
HDD	Heating Degree Days
CDD	Cooling Degree Days
BPTT	Back Propagation Through Time
BOM	Bureau of Meteorology
CPT	Conditional Probability Table
LOR	Lack of Reserve
RXS	Regional eXcess Supply
LCC	Largest Credible Contingency
NEM	National Electricity Market

Chapter 1
Introduction

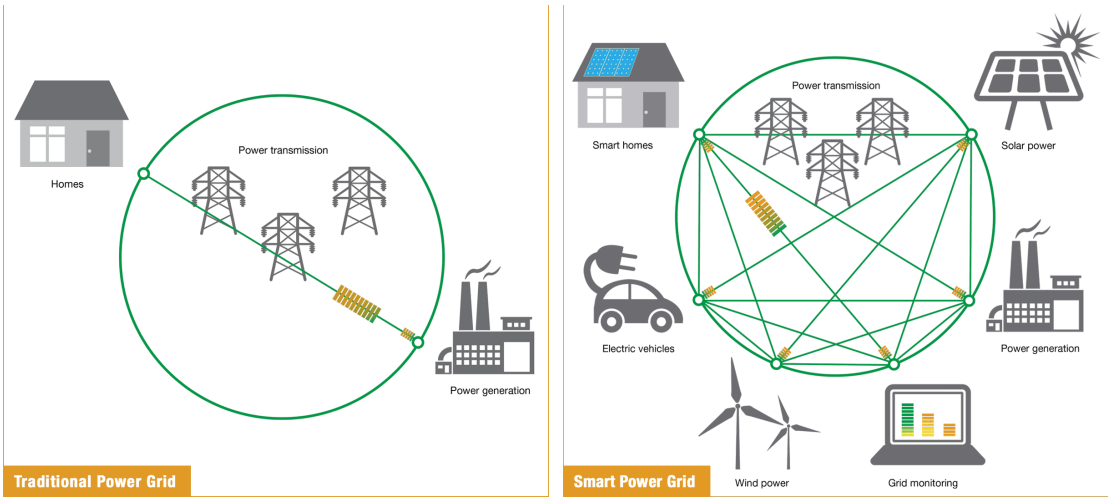
1.1 Developing Reliable and Sustainable Energy Systems

SUSTAINABLE development has been defined by the United Nations Brundtland Commission as: "development that meets the needs of the present without compromising the ability of future generations to meet their own needs" [192]. Since the industrial revolution in the 1800s, world energy demand has been increasing rapidly. This is not just due to increasing population, but also due to increasing energy usage per person. More than 80% of the world's energy demand is supplied by burning fossil fuels, which introduces two major problems: (i) Using fossil fuels as the main source of energy is unsustainable, as society is consuming these sources of energy at a much faster rate than nature is able to reproduce them, so these energy resources are non-renewable. (ii) These types of energy sources result in the emission of greenhouse gases such as CO_2 and are a major cause of climate change. Since the industrial revolution, the level of CO_2 emissions has increased dramatically. The world is now 1 °C warmer on average than it was in pre-industrial times.

These factors have motivated a range of renewable energy initiatives, alongside efforts to replace existing power infrastructure and technology with more versatile and robust forms of low-carbon alternatives. Current electricity generation infrastructure has been established with a centralized architecture, favouring a relatively small number of high capacity generations plants operating at 50 or 60 Hz and interconnected by transmission systems that use very high voltages (e.g., 400 kV). Current power systems are undergoing a wide-scale enhancement, with large-capacity distributed generation systems (DGs) being launched, which allow for the integration of thousands of generators,

displaying numerous technological capabilities, current and power levels, voltages, and typologies. This development sees more large photovoltaic and wind generators being introduced that can produce megawatts of power. Therefore, having DGs incorporated within current power grids is changing the makeup of the entire system, which introduces a range of complications. In particular, the unpredictable behaviour of renewable energy systems (RES), due to fluctuations in solar and wind conditions, complicates the task of ensuring the stability of the overall power grid.

In Figure 1.1a a typical traditional power grid consisting of scheduled generators (e.g., coal, gas or nuclear plants), transmission lines and consumers is shown. In Figure 1.1b a typical smart power grid with buses that host scheduled and renewable generators and consumers, is presented. These buses are connected through transmission lines, transformers, and phase shifters. The main task of the transmission and distribution control system is to dispatch the available energy generation capacity according to the forecast load in order to satisfy the demand-supply balance, transmission stability and security criteria. There are two types of generator sources: (a) scheduled generators such as hydrological (rivers, reservoirs, etc.), thermal (coal, gas, etc.) and nuclear generators that are committed to be available for each hour from the beginning of the day; (b) non/semi-scheduled generators that use renewable energy sources. Renewable energy generators are distributed, non-synchronous, weather dependent, and cannot be scheduled ahead of time. Hence they can disrupt the control of the demand-supply balance in the power grid. New real-time grid decision-making and control techniques are needed to address the effect of these uncertainties resulting from the integration of renewable energy sources into the power grid. To illustrate this evolution of generation capacity, Figure 1.2 shows the changes in supply capacity and maximum demand in South Australia since July 2008 to September 2017 [6]. This figure illustrates how the mixture of supply has changed over recent years, and how the level and type of reserves in the system at any time can vary according to changes in both supply and demand, which add more uncertainties to the system. While demand peaks have always exhibited a high degree of uncertainty and are likely to vary according to the weather, the power system must now also manage the increased uncertainty and variability of supply. We next introduce how the new architecture and management of smart grids aims to address these uncertainties.



(a) A typical traditional power grid, consisting of load, scheduled generators and buses

(b) A typical smart power grid, consisting of load, scheduled generators, renewable generators, and buses

Figure 1.1: Traditional power grid vs smart power grid [3].

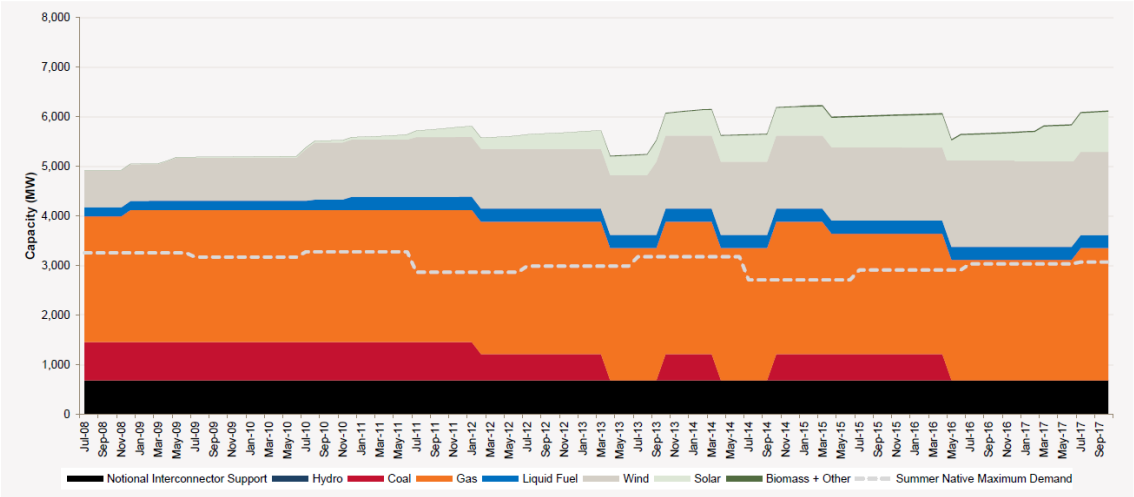


Figure 1.2: Registered capacity and maximum demand in South Australia 2008 to 2017 [6].

1.2 Smart Grid

The existing power grid is set up in a centralised manner, allowing power to travel direct from power plants to consumers in a unidirectional fashion via transmission and distribution lines. All the decision-making and organisation takes place centrally, and on a smaller scale in substations, with no intelligence located at remote terminations (loads) at all. Introducing a more distributed grid would require such decision-making to be

rolled-out locally, allowing key intelligence to flow back and forth in order to reach the same levels of stability and dependability of power supply [82].

Such an example is that of the Smart Grid (SG), which is designed with a communication infrastructure that does not assume a highly centralised set of power plants, and provide the communication and computing infrastructure to monitor and control the diverse sources [82]. This allows for several important factors, including new grid organisation solutions achieved by adapting demand-side management and ensuring effective energy storage. This is required for handling variations in supply that cannot be avoided by RES, as well as the introduction of new types of loads, such as the charging of electric vehicles, where there may be flexibility in the timing of these loads.

As highlighted by Figure 1.3, the key SG challenges are as follows: potential applications need to be characterized in order to establish the infrastructure scope and capabilities; advanced IT needs to be integrated to support monitoring, modeling and analysis of the overall grid; robust communication frameworks are needed to transport critical data and commands between multiple sources and controllers; and the overall energy infrastructure is representative of the physical infrastructure required for producing, circulating and transmitting power. The communication framework is particularly important for supporting a high-performance grid. Ensuring that all applicable grid components are connected is necessary for obtaining data on any external influences, which informs monitoring and protection tasks, as well as performance. Effective IT procedures need to support the convenient storage and sharing of relevant data about the state of supply demands, and the underlying transmission infrastructure. Figure 1.4 [228,231] shows the relationship among all the components of the different layers of the smart grid, from the energy infrastructure to the potential application. In the next section we introduce a road map of the applications and services that the smart grid infrastructure can enable.

1.2.1 Smart Grid Applications and Services Road-Map

Nations such as Australia depend on energy infrastructure that was established around a century ago, consisting of powering consumers via centralized electricity plants using a transmission network that spans hundreds or thousands of kilometres. Inevitably, there are significant energy losses included in the operation of such an infrastructure.

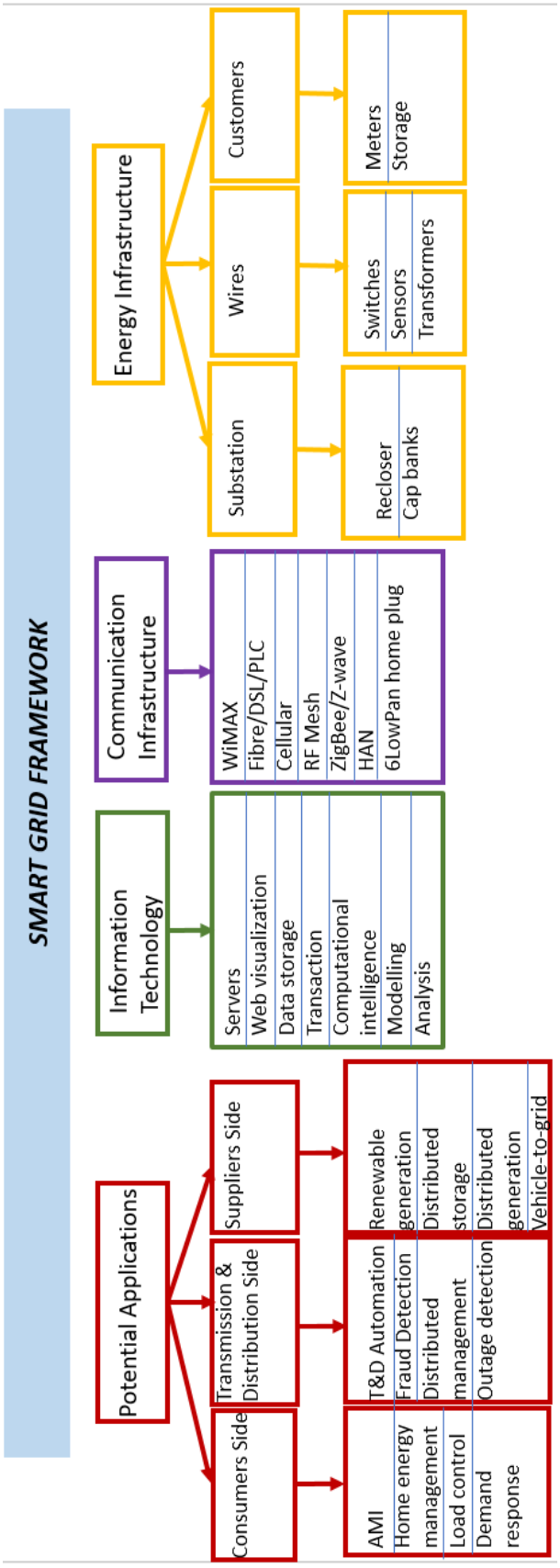


Figure 1.3: Smart Grid framework presenting the potential applications, information technology, communication and energy infrastructure of the overall system.

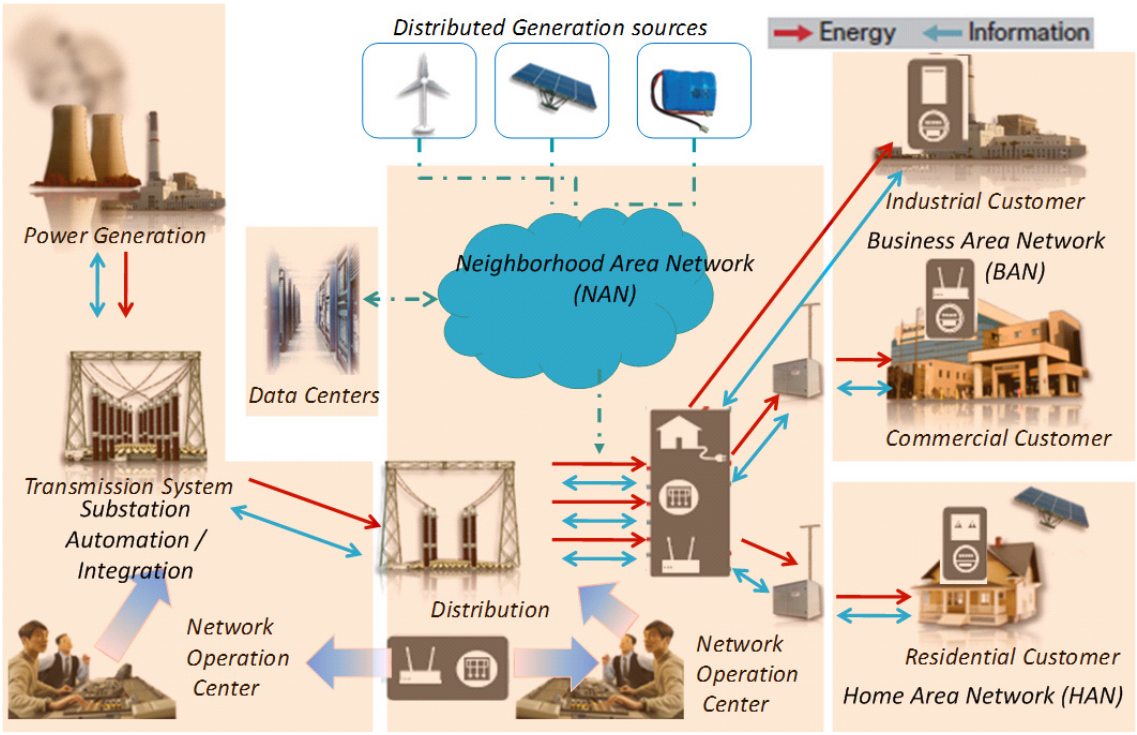


Figure 1.4: A typical Smart Grid framework [228, 231]

The power grid now in place is not designed to control solar and wind generation that is widely distributed. Furthermore, it is difficult to meet current CO_2 emission targets while relying on traditional coal and gas power generators, without integrating large scale renewable sources. Embedding SG into the existing power infrastructure, therefore, has become a key goal for utility providers and politicians, who see SG as the solution to managing demand issues in the years to come, as well as reducing carbon emissions.

Figure 1.5 [82] demonstrates how smart grid advances, supported by Advanced Metering Infrastructure (AMI), are currently being adopted, moving providers away from typical Automatic Meter Reading (AMR) networks and one-way forms of communication. Thanks to new capabilities in distribution automation, smart sensors, advanced management, and demand response, SG technology can support a range of new applications, such as outage detection, load control and emissions control. In order to achieve the potential benefits of introducing smart grids for such applications, there are a number of objectives that decision makers need to address, namely, transmission and distribution performance, renewable resources integration, and improved load forecasting.

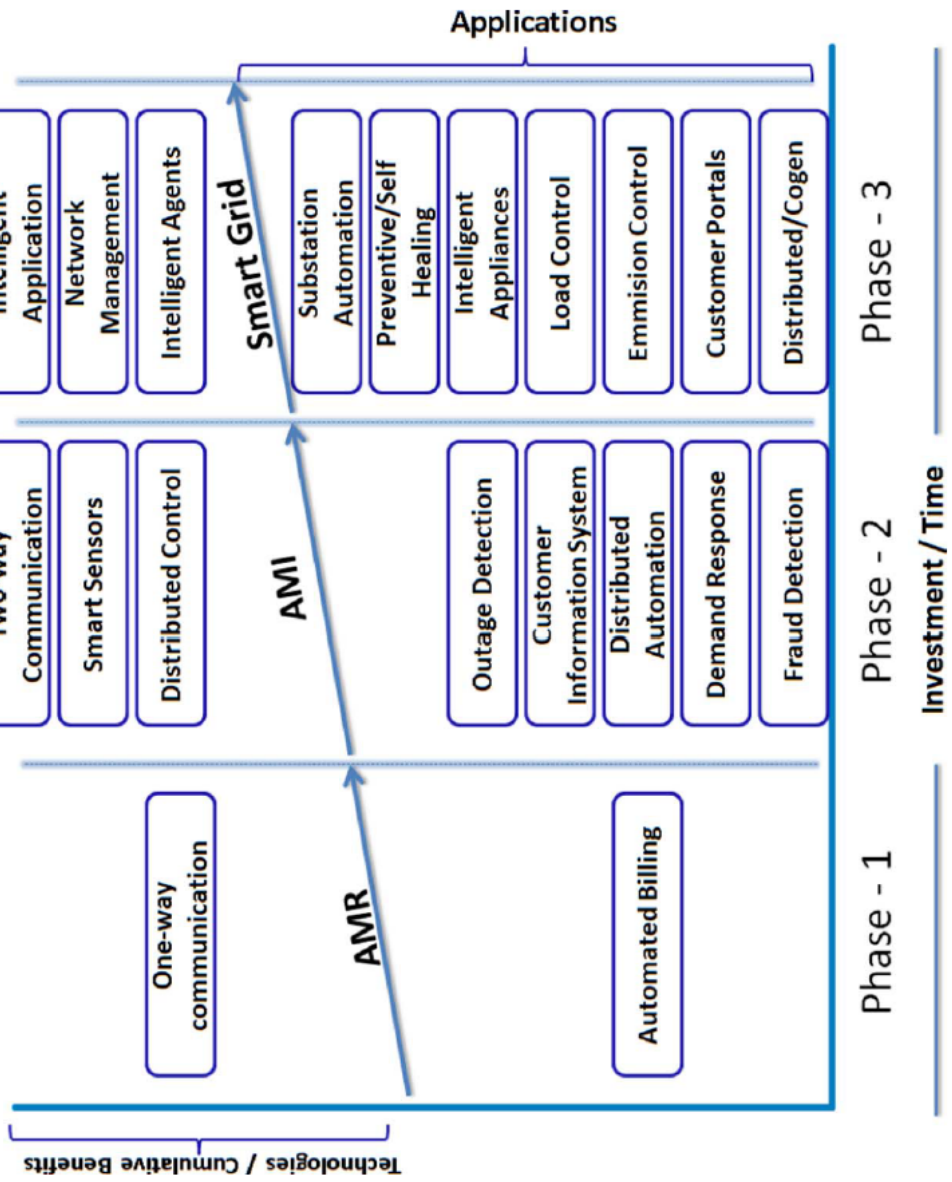


Figure 1.5: Smart Grid application road-map from Automated Meter Reading (AMR) to Advanced Metering Infrastructure (AMI) to Smart Grids [82].

Future Transmission and Distribution Systems

Around 6% to 8% of generated power is lost during transmission and distribution, which is caused by wire resistance and certain equipment the electricity flow encounters [82, 148, 152]. As SG develops, however, transmission and distribution capabilities will be upgraded, benefiting from less intrusive equipment such as FACTS (flexible AC transmission systems), which allow for power transfer with minimised losses. Also crucial for reducing transmission and distribution losses are HVDC (high-voltage direct current) advances, which can be applied in areas of high population to support better access to renewable energy [82, 148, 152]. Transmission issues can also be tackled by HTS (high-temperature superconductors), while DLR (dynamic line rating) allows for the pinpointing of equipment-related issues to avoid the consequences of overloading [82, 148, 152].

Renewable Resources

Green energy sources, like wind and solar, need to be merged with the existing power infrastructure. Alongside this, the use of Plug-in Electric Vehicles (PEVs) and Plug-in Hybrid Electric Vehicles (PHEVs) will become more feasible for many customers, reducing CO_2 from current modes of transport. Furthermore, numerous backup supplies will be integrated to make sure power is not interrupted because of unpredictable natural factors, establishing a system that can prepare for adverse weather, support more storage of power, as well as including the benefits of grid-load balancing and real-time output monitoring. As a result, yearly CO_2 reductions are likely to be around 60 to 211 million metric tonnes by 2030 [73, 82, 208].

Advanced Forecasting

Heightened demand for electricity supply differs depending on the geographical region. For example, consumption habits in the US have been estimated to rise by 1.07% annually until 2030, according to the US Energy Information Administration (EIA) [73, 82]. As the diversity of generation grows with the introduction of more variable sources, such as wind and solar, more sophisticated supply forecasting is needed. Moreover, with the introduction of methods for demand response (DR), better load forecasting is also needed

to balance demand-supply. These forecasting models need to take account of more detailed customer data from AMI, pricing information, and the effect of new loads such as PEV charging.

1.3 Role of Computing and Analytics

In order to achieve the potential of the smart grid and its possible applications, a variety of computing challenges need to be added. In this section we provide an introduction to some computing challenges with a focus on data analytics.

Information flow is one of the key benefits of smart grids, such as measurements gathered by sensors, grid operator control signals, and pricing information to inform both providers and customers. However, the ability to amass data is pointless if it cannot be put to good use. How energy providers apply the data they accrue, therefore, is critical to measuring the potential of the smart grid. In turn, achieving this demands data integration, together with machine learning techniques and statistical analysis, which can extract useful insights into the operation of the smart grid. There are three main forms of analytics in this context which focus on different sources of data, namely, supply analytics, network analytics, and demand analytics.

1.3.1 Supply Analytics

Electricity production data can be interpreted using analytics. Being able to predict the levels of wind and solar energy supply over different time scales, and preparing for the impact of adverse weather conditions is crucial for managing the supply-demand balance. Analytics, therefore, enables providers to decide whether a demand response event needs to be put in place, as well as establishing generation unit commitment and generation schedules [200].

1.3.2 Network Analytics

Key data in respect of network analytics includes voltage, current, and phase, for example, which can be gathered from operating transmission lines (the transmission lines

which are in service at the time of data gathering). These observations, inform decision-making regarding grid efficiency and security, offering the ability to predict and plan for errors as well as detecting any suspicious activity or overuse [200].

1.3.3 Demand Analytics

Understanding consumer and energy-use habits and patterns is crucial for informing providers, enabling them to assess the demands they will need to fulfil, which can include day-ahead market purchases if necessary (which are more cost effective than intra-day solutions) [200]. Additional benefits include the ability to set tariffs and to understand sales opportunities within the customer base. However, the security and privacy of customers also need to be safeguarded at all times, considering the detailed level of data being gathered.

1.4 Motivation for Forecasting in Power Grids

How much electricity should be generated to meet demand for the next 24 hours over a region? What will be the temperature, humidity, wind speed, and wind direction for the next three hours? What amount of power will be generated from solar roof tops for the next day? These are the kinds of questions that an operational power grid typically needs to deal with to provide a reliable power grid. Answering these questions involves predicting observations several steps ahead from a given sequence of historical observations in the form of a time-series.

Forecasting is an essential task in many industries. For example, retailers forecast the demand of the new products to decide what price to offer to customers; tourism companies forecast the number of tourists to schedule tours; electric utilities forecast the amount of electricity consumption and generation to keep the balance between supply and demand. In contrast to many other industries who have the capability to store and reserve their products, the electric power industry does not have such capability to store a massive amount of electricity using today's technologies [102]. Therefore, due to the storage limitation of electricity, one of the crucial tasks of power system operation is to maintain a balance between supply and demand at every moment. As a result, forecasting is an

essential function in the electricity power grid.

In the smart grid era, customers can play an important role in achieving energy savings by modifying their electricity consumption pattern according to the dynamic electricity market, and selling renewable energy on an open market through demand response mechanisms [82]. The demand response mechanism depends on the convergence of accurate electric power load measurements, renewable energy, and price forecasting. Therefore, a crucial task for utility companies in this context is load forecasting. Load forecasting or electricity demand forecasting is a critical task in functions such as power supply planning, power systems operations and maintenance, transmission and distribution systems planning, financial planning, and demand side management. Due to the fundamental role of load forecasting in power system operations and planning, any error in load forecasting can result in significant cost increases for electricity suppliers and increase the chance of unexpected blackouts or brownouts. In contrast, accurate load forecasting can result in improvements to network reliability. Therefore, trying to produce more accurate load forecasting models has become a major research challenge for energy suppliers, energy marketers, financial markets, and other parties that contribute to electric power generation, distribution and transmission.

1.5 Research Challenges Addressed in This Thesis

1.5.1 Short-Term Load Forecasting Based on Big Data (Smart Meters)

Load forecasting has been widely studied since the 1970s. Traditional approaches for load forecasting are based on linear models; such as autoregressive (AR) and autoregressive with moving average (ARMA) models [143,161]. In the last two decades, nonlinear models based on machine learning methods such as neural networks [194], support vector machines [138], and k-nearest neighbours approaches [165] have drawn attention due to their superior performance in many predictive modeling competitions. Neural networks have been used widely in load forecasting due to their ability to approximate complex nonlinear relationships [92]. However, neural network methods have some potential drawbacks, such as over-fitting of the model, sensitivity to random weight initialization, and tendency to convergence to local optima [143]. To address these limitations of tra-

ditional neural networks, recently, new approaches called deep neural networks have been proposed. Deep belief networks with many nonlinear layers can represent complex features from their inputs [143], and construct a generalized model with the ability to learn these complex features from the data. In Chapter 3 we propose a method for load prediction using Hinton and Salakhutdinov's deep belief nets (DBN), which are a type of probabilistic generative neural network composed of multiple layers of restricted Boltzmann machines (RBM) [90]. In addition, in Chapter 3 we introduce two load forecasting methods, based on deep neural networks and traditional neural networks. The two proposed methods have been applied on real life residential load data, and a comparison in terms of their prediction accuracy is presented.

It is known that load forecasting for individual customers is complicated because each households' load depends on a variety of factors that may not be directly observable. In Figure 1.6 a comparison between individual household consumption patterns and utility level consumption patterns is shown over three days. It is clear that the aggregated consumption pattern is much more regular than the individual consumption patterns. At the granularity of half an hour, the energy consumed by a single house can fluctuate widely, due to the dependency on the number of residents in the home, the appliances that are used at that particular time, the lifestyle of the people living in the home, and so on. Thus, accurate load prediction for an individual household can be difficult due to the individual variations in consumption patterns.

One of the most promising applications for the large volumes of data from smart meters is to improve the accuracy of electrical load forecasting. Typical methods for improving load forecasts using the data generated from smart meters of individual customers are based on the use of clustering [116, 193]. In this approach, the knowledge about load consumption behaviour of customers is used to improve the accuracy of forecasting. Instead of developing a single forecasting model for the accumulated load consumption of all customers, clustering can be used to divide the customers into sub-populations with similar demand profiles. Then, a forecasting model can be provided for each cluster of customers according to their load profile. Thus, a more accurate model can be provided for each cluster, and then the load forecast for all consumers can be obtained by combining the forecasts of these models.

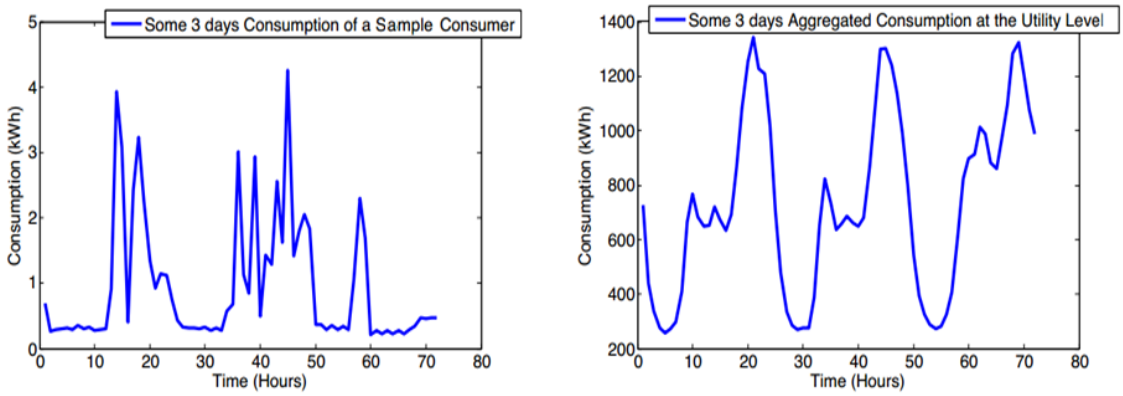


Figure 1.6: Example of power consumption time-series at an individual household level (left) and at an aggregate utility level (right).

Applying clustering as an initial step in electric load forecasting has been the focus of several studies [53, 150]. The K-means clustering method has been widely used in previous works for this purpose [178, 193]. However, little attention has been paid in previous studies to the choice of clustering method with the aim of improving the aggregate level of forecasting accuracy. In Chapter 3, we show that the accuracy of load forecasting depends not only on the load forecasting method, but also on the accuracy of the clustering method and the accuracy of extracting the load pattern of each cluster. In addition, we show that careful clustering of customers can result in smaller forecasting error. In contrast to the previous works, we apply a time-series approach to clustering the load consumption patterns of each consumer based on the shape of their daily profile. By clustering the time-series data, we can analyze its variations over time in detail, and search for common profiles that can be potentially useful for load prediction. Such profiles can be extracted according to the shape of the load profiles. We search for groups of customers that show similar behavior over time. We apply a novel clustering algorithm (k-shape clustering [170]) that aims to reduce the error in load forecasting through more accurate assignment of individuals to clusters based on their load consumption pattern.

In summary, in Chapter 3 we propose a new approach for short-term load forecasting from customer level to substation level data using an effective new combination of clustering and deep learning methods, along with a new weighed aggregation mechanism. Our evaluation using smart meter data from a publicly available real-life dataset demonstrates the improved accuracy of our approach over existing methods.

1.5.2 Efficient and Accurate Time-Series Clustering Method for Load Profiling

In Chapter 3, we show that by using an appropriate clustering method we can further improve forecasting accuracy. This motivated us to develop a new clustering algorithm for time-series data, which is quite accurate and computationally efficient. In Chapter 4 we introduce our two new proposed clustering algorithms, namely, Fuzzy C-Shape plus (FCS+) and Fuzzy C-Shape plus plus (FCS++), and we show that the two new algorithms outperform state-of-the-art shape-based clustering algorithms in terms of accuracy and efficiency. In the following we briefly overview the importance of time-series clustering, and the challenges and open questions in this area.

Increasing volumes of time-series data are becoming available from a diverse range of sources, including smart meters, smart phones and environmental sensors. The existence of large volumes of time-series data in many applications has motivated data miners to investigate specialized methods for mining time-series data. Clustering is a popular data mining method due to its powerful exploratory nature and its usefulness as a pre-processing step for other data mining techniques. Techniques for analyzing time-series data include dimensionality reduction [56], indexing [122],[125], and segmentation [123]. In addition, techniques have been developed to extract the underlying shape or trends in the data, for tasks such as pattern discovery [124, 145, 174], classification [111, 184], rule discovery and summarization. An open challenge in time-series analysis is how to cluster a set of time-series according to the similarity of their underlying shape. Most state-of-the-art approaches for shape-based clustering suffer from two main drawbacks: *(i)* they are computationally expensive, and thus not suitable for large volumes of data [145,159,174,229]; and *(ii)* these approaches are limited to specific domains [229], or their effectiveness has only been evaluated over a small number of datasets [145, 159]. We address the mentioned challenges in Chapter 4 by proposing two new clustering algorithms. We generalize the fuzzy c-means (FCM) model [28] to accommodate clustering in finite time-series data by two approaches: (i) selecting a statistical measure, cross-correlation, as a shape similarity distance measure for comparing time-series sequences; and (ii) selecting a new centroid computation technique based on a shape extraction method [170] while computing the clusters. We perform an extensive evaluation over 48 different time-series datasets and demonstrate the superiority of our technique against

existing schemes.

1.5.3 Short-Term Load Forecasting Framework for Nonstationary and Unbalanced Data

As mentioned earlier in this chapter, short-term load forecasting plays an important role in the reliability, security, market operation, and scheduling of reasonable dispatching plans for smart power grids. With increasing integration of renewable energy sources and the introduction of different demand response programs in the energy market, the load on the grid has become more volatile and less predictable than ever before [153, 155, 162, 176]. Thus, trying to produce more accurate load forecasting models has become a major research challenge for energy suppliers, energy marketers, financial markets, and other parties that contribute to electric power generation, distribution and transmission. There are many challenges associated with load forecasting. Load demand depends on different exogenous factors like temperature, humidity, wind speed, seasonal patterns related to human activities and cyclic information, like time of year/season/week/day, and type of day (holiday/working day/weekend). The selection of exogenous factors and modeling the relationship between the selected variables and load data plays a crucial role in load forecasting. In Chapter 5 we introduce a novel factor, apparent temperature, which has a major impact on the accuracy of the prediction model. Another challenge in load forecasting is the unbalanced nature of the training data sets that arise due to the frequency of abnormal situations like extreme weather conditions, which are critical events that require highly accurate forecasts, or holidays that show different load patterns. In addition, the nonstationary behaviour of the demand data adds another challenge to the accuracy of load forecasting. Hence there is a need for a load forecasting framework that can perform accurately and address all the mentioned challenges. In Chapter 5 we propose a novel framework for short-term load forecasting using an effective new combination of our proposed clustering algorithm in Chapter 4 (c-Shape clustering), LSTM networks [74] and Xgboost [48] methods. In particular, our proposed framework introduces an ensemble process together with novel features that lead to improved accuracy of the load forecasting model. The proposed framework is able to produce multiple historical models for different temporal variations, and uses ensemble learning

techniques to forecast the day ahead load with high accuracy. Moreover, it is able to handle the unbalanced nature of the training data set, which enables high accuracy in predicting the load in abnormal situations like very hot days and other severe weather conditions. The results of our proposed approach in Chapter 5 have been compared with the existing well-established Australian short-term load forecasting system over the five regions in Australia. It is shown that our predictive framework has a substantially higher accuracy over the existing methods, and it is flexible and easy to apply which makes it an effective model to supplement an operational power system.

1.5.4 Managing Uncertainties in Dynamic Reserve Sizing

With the integration of renewable energy and grid modernization, the power generation system has become more volatile and less predictable than ever before. Renewable energy sources are weather dependent, which cannot be predicted exactly; the same statement can be made for energy demand, consumer behavior, energy prices, and transmission constraints [220]. On one hand we want to produce clean, safe, and reliable energy, while on the other hand the sources of producing clean energy are inherently intermittent. Energy demand also depends on many complicated factors to model. In fact, there are many uncertainty factors associated with both demand and supply modeling, the latter depending on the former, which is the pilot of the system. So, maintaining the balance between demand and supply at all time is a challenging task in the future development and operation of electric power grids.

From a long-term perspective (one year to several years ahead), it is predicted that energy demand will increase around the world. By considering the uncertainties associated to these predictions, the margins of additional resources required to satisfy future energy demand must be evaluated. Long-term planning in power systems is essential for scheduling of new generator capacity installation, and knowledge of associated forecast uncertainties is pivotal for realistic future capacity management. In contrast, from a short-term perspective (seconds to hours ahead), to maintain frequency stability within the operational frequency acceptable range, the demand and supply must match each other as quickly as feasible. Frequency instabilities cause major security problems in power grid like grid black/brown out. Thus, a real-time stochastic decision making

mechanism is needed for risk management and to make the electrical power grids resilient against normal demand deviation as well as unplanned events like generator unit outage due to faults.

In such a system with a variety of uncertainty sources, probabilistic methods play a crucial role for characterizing, representing, and quantifying the uncertainties associated with the electrical grid operation. In particular, probabilistic methods are needed to infer the entire conditional distribution of future observations at each time step. Due to the variability and partial unpredictability of renewable energy sources (RES), generation reserve sizing is a key aspect of managing operational risks and enhancing grid reliability. Most existing approaches for reserve sizing estimate the forecasting error for RES using either simple point forecasts, or probabilistic forecasts that assume a predefined parametric model of the underlying distribution, such as a Gaussian distribution. However, these approaches are unable to accurately model the probability of critical events that occur in the tails of the distributions, which is vital for operational risk management. Further, they do not account for multiple dynamic factors that may impact operational risk. In Chapter 6 we introduce a data-driven dynamic probabilistic reserve sizing method based on the technique of Dynamic Bayesian Belief Networks [156]. The method considers the underlying distribution of forecasting errors from the availability of conventional generators (forecasting errors of their available generation capacity), demand, RES, and prevailing conditions like weather and market prices. In addition, we introduce a new dynamic metric for calculating the reliability level of the power grid in order to provide a real-time stochastic decision support tool for power system operators. The proposed method is demonstrated using seven years of real-life historical data with a granularity of 30 minutes from the power system in Australia, and has been implemented and is currently used by the Australian system operator.

1.6 Thesis Organization and Contributions

Figure 1.7 shows the structure of this thesis and the connections among the chapters. The structure of this thesis is organized as follows.

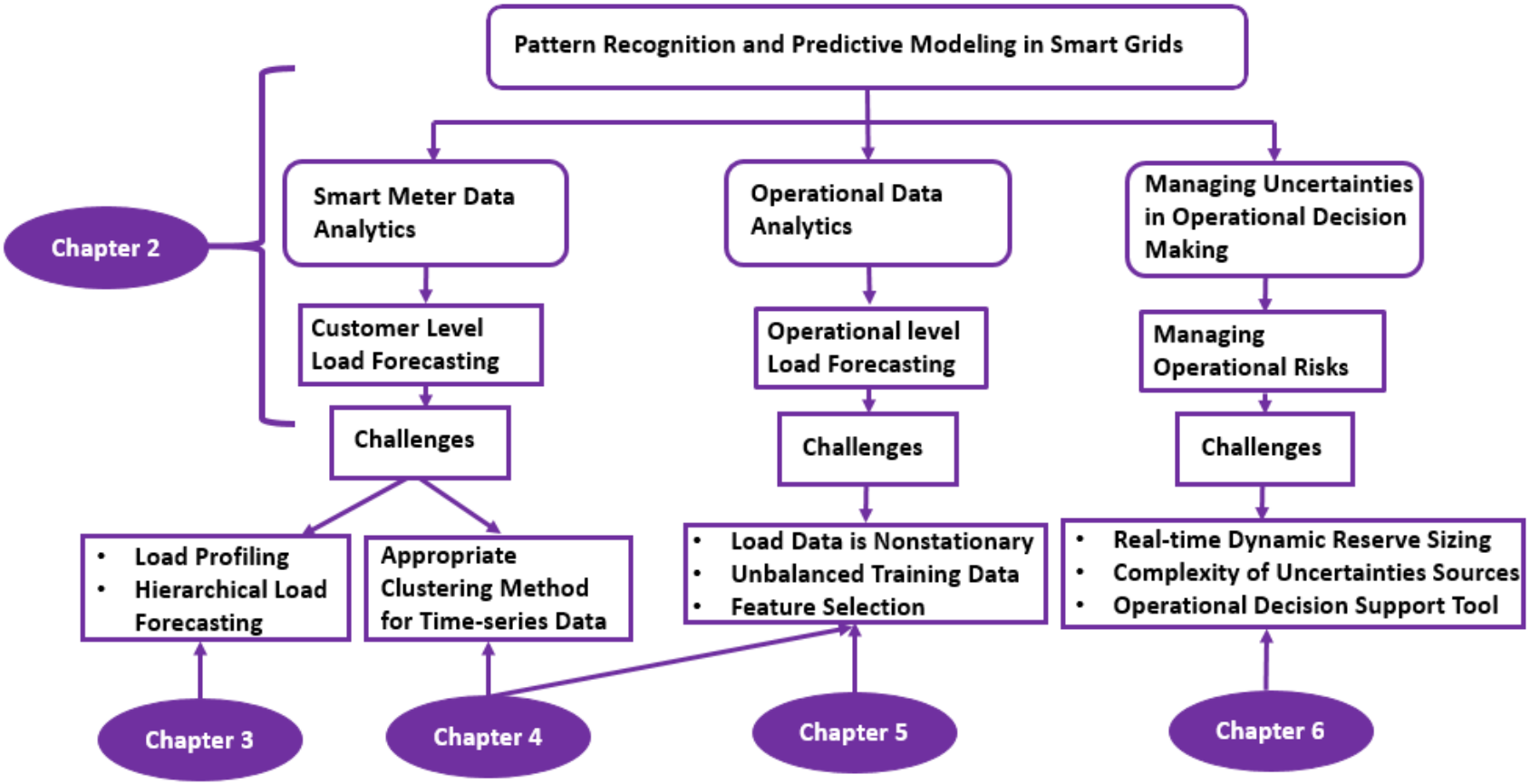


Figure 1.7: Thesis structure and connections between chapters

Chapter 2 - Background and Literature Review:

This chapter covers the preliminaries and main concepts of pattern recognition and predictive modeling in smart grids. First, a brief explanation about load forecasting is presented, followed by a literature review on the relevant categories of techniques and methodologies for load forecasting. Then the concept of managing uncertainties in smart grids is presented with a brief explanation about application side of probabilistic modeling in smart grids. Finally, we focus on pattern recognition in smart grids, and we review the literature related to time-series clustering.

Chapter 3 - Improving Load Forecasting Based on Deep Learning and K-shape Clustering:

This chapter introduces a new approach for hierarchical short-term load forecasting using an effective new combination of clustering and deep learning methods, along with a new weighed aggregation mechanism. We propose a new method for load forecasting that leads to improved accuracy of short-term load forecasting. We show that by appropriate choice of features we can significantly improve the accuracy of forecasting. Moreover, we study a novel time-series clustering method (K-shapes) [170] that can cluster consumers into the groups based on their load consumption profiles, such that the accuracy of models based on each K-shape cluster is more accurate than models based on traditional K-means clustering. In addition, we implement our proposed approaches on publicly available real-life data and demonstrate the biggest improvement in accuracy is achieved by applying both new approaches, i.e., applying the new clustering method and the new forecasting method. Moreover, in contrast to previous work [17,178,193] that simply summed the forecasting results of each cluster to obtain the final aggregate forecast, we introduce a new weighted summation method for accumulating the forecasting results of each cluster according to the size of their membership.

The outcomes of this chapter have been published in **P1** listed in the Preface.

Chapter 4 - Fuzzy c-Shape: A New Algorithm for Clustering Finite Time-Series Waveforms:

The existence of large volumes of time-series data in many applications such as smart grids has motivated us to investigate specialized methods for mining time-series data. Clustering is a popular data mining method due to its powerful exploratory nature and its usefulness as a preprocessing step for other data mining techniques. In this chapter we develop two novel clustering algorithms for time-series data that are extensions of a crisp c-shapes algorithm. The two new algorithms are heuristic derivatives of fuzzy c-means (FCM) [132]. Fuzzy c-Shapes plus (FCS+) replaces the inner product norm in the FCM model with a shape-based distance function. Fuzzy c-Shapes plus plus (FCS++) uses the shape-based distance, and also replaces the FCM cluster centers with shape-extracted prototypes. Numerical experiments on 48 real time-series data sets show that the two new algorithms outperform state-of-the-art shape-based clustering algorithms in terms of accuracy and efficiency. Four external cluster validity indices (the Rand index, Adjusted Rand Index, Variation of Information, and Normalized Mutual Information) are used to match candidate partitions generated by each of the studied algorithms. All four indices agree that for these finite waveform data sets, FCS++ gives a small improvement over FCS+, and in turn, FCS+ is better than the original crisp c-shapes method. Finally, we apply two tests of statistical significance to the three algorithms. The Wilcoxon and Friedman statistics both rank the three algorithms in exactly the same way as the four cluster validity indices.

The outcomes of this chapter have been published in the paper **P2** in the Preface, which achieved the **Best student paper award** at IEEE International Conference on Fuzzy Systems, 2017.

Chapter 5 - Robust and Accurate Short-Term Load Forecasting Framework:

In this chapter we propose a framework for grid level load forecasting to address some of the challenges for load forecasting, including unbalanced training data, non-stationary nature of the load data, and feature selection for predictive modeling. In summary the novel contributions made in this chapter are:

- We introduce a robust and accurate framework based on unsupervised ensemble

learning, which is able to produce accurate predictions even in difficult-to-predict cases that arise out of volatility, spikes, and abnormal situations.

- We introduce a way to handle unbalanced training data sets, which leads to high accuracy of forecasting in abnormal situations like extreme weather conditions.
- We address the research question of whether combining weather variables that reflect the human perception of temperature results in more accurate load forecasts than using traditional weather variables.
- We introduce novel features corresponding to evening and morning peak demand estimated by an LSTM, as key features in improving the accuracy of load forecasting.
- The performance of the proposed method is validated using real-life data from the power system in the Australian National Electricity Market, as well as through on-site implementation by the system operator. It is shown that our predictive framework has a substantially higher accuracy over existing methods in both normal and abnormal situations.
- Our proposed framework is flexible and easy to apply, which makes it an effective model to supplement any operational power system.

The outcomes of this chapter have been published in the paper **P3** in the Preface.

Chapter 6 - Data-Driven Dynamic Probabilistic Reserve Sizing Tool:

Due to the variability and partial unpredictability of renewable energy sources (RES), generation reserve sizing is a key aspect of managing operational risks and enhancing grid reliability. In this chapter, we introduce a data-driven dynamic probabilistic reserve sizing method based on the technique of Dynamic Bayesian Belief Networks (DBBNs). This technique is suitable for diverse problems of varying size and complexity, where uncertainties are inherent in the system. Because DBBNs have a simple casual graphical structure, they are one of the most effective models for representing uncertainties [33]. In addition, DBBNs are one of the best models for simplifying conditionalization, which helps in designing the decision making process [31]. Our contributions are as follows:

- We introduce a novel metric of reserve forecast uncertainty, called Forecasting Uncertainty Measurement (FUM), which can be used to calculate the reliability level of a power system and regulate the size of the reserve in a dynamic manner. Previous methods calculate the reliability level of a power system by considering predefined contingency scenarios, which have limitations in terms of accuracy and transparency, such as only considering the largest generator risk. In the new proposed measure FUM, possible contingencies and errors from historical data are considered and can be automatically updated. This new measure conveys information about the uncertainties associated with the excess supply available in each region of a power system in MW, which is easy to understand and practical for grid operators to assess the probability of a shortfall in the available capacity reserves in each region of the power grid on a continuous basis.

- For the first time, we introduce the use of DBBNs for dynamic probabilistic reserve sizing, and demonstrate their suitability in modeling system complexity.

- To the best of our knowledge, this is the first time that reserve sizing has been analyzed in a dynamic manner for a real hybrid system with high complexity arising from a wide variety of power sources, such as grid-connected and embedded RES and conventional generators, while taking into consideration a range of prevailing conditions, such as market prices, weather, and time of day.

- The performance of the proposed method is validated on real data from the Australian power system, and through actual on-site implementation by the Australian system operator, AEMO (Australian Energy Market Operator).

The outcomes of this chapter have been published in the paper **P4** in the Preface.

Chapter 7 - Conclusions and Future Work:

This chapter overviews the results and conclusions of the techniques proposed in this thesis. This chapter concludes by proposing several interesting directions for future research.

Chapter 2
Background and Literature Review

Over the past decade, by installing advanced metering infrastructure, sensors, and communication devices in the power grid, the electric power system process has become smarter than ever before. With the emergence of the Smart Grid, new challenges have arisen in the operation of the power grid, such as managing demand response, integration of renewable energy in the grid, and the coordination of distributed energy resources. If these challenges can be addressed, the Smart Grid has the potential to provide a clean and more efficient electrical power system at reduced cost.

2.1 Introduction

THIS chapter presents a literature review on some of the challenges in Smart Grids, which we presented in Section 1.5 of Chapter 1, and which we address in this thesis. In particular, we focus on the problem and approaches for load forecasting, and associated challenges of managing uncertainty in some of the decision-making functions that are based on load forecasting. Instead of presenting every related work, the references are selected carefully to focus on the major developments and representative works in the related area in each section of the chapter.

The rest of this chapter is organized as follows. Because our main focus in this thesis is on forecasting, especially load forecasting in smart grids, we start by providing an overview of the load forecasting problem in Section 2.2. Following that, in Section 2.3 we discuss methodologies that have been applied to load forecasting, including hierarchical load forecasting, load profiling approaches, and feature selection. Then in Section 2.4, we review the main techniques applied to the load forecasting problem. In Section 2.5 we provide background on the techniques for addressing the uncertainties introduced by integrating renewable energy sources to the power grid, in order to provide decision

support for operating the power grid.

2.2 Load Forecasting in Power Grids

Load forecasting is an essential aspect of a power grid management, which is widely used in all parts of the power system, such as power supply planning, power systems operations and maintenance, transmission and distribution systems planning, financial planning, and demand side management. Load forecasting can be categorized based on the electricity business needs and the factors that impact on electricity consumption. Consequently, we can provide different forecasts for different purposes, based on the time horizon over which the forecast is required.

2.2.1 Horizon

Load forecasting problems can be classified based on their forecasting horizon as follows [105]:

- Very short-term load forecasting, which corresponds to the forecasting horizon from a few minutes ahead to a few hours ahead.
- Short-term load forecasting, which includes the forecasting horizon from one day to two weeks ahead.
- Medium-term load forecasting, with a forecasting horizon ranging from two weeks to three years ahead.
- Long-term load forecasting, with a forecasting horizon ranging from three to fifty years ahead.

A more general classification is to classify load forecasting into two classes: short-term and long term load forecasting with a two weeks horizon as the cut-off point. The main reason for using a two weeks horizon as the cut-off is that weather, especially temperature, is a major factor affecting electricity usage. In practice, it is difficult to accurately predict the weather more than a week ahead. In this thesis we mainly focus on short-term load forecasting (STLF).

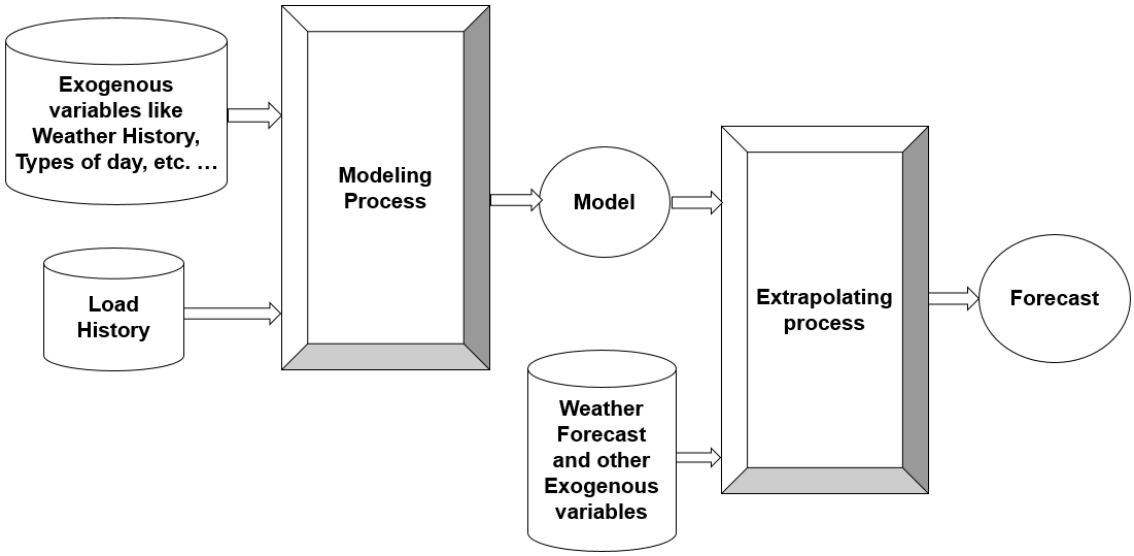


Figure 2.1: A typical short-term load forecasting process

2.2.2 Process

The typical STLF process relies on weather information and some other exogenous variables like calendar, type of day (working day/weekend), and seasonal effects. This process is shown in Figure 2.1. The exogenous variables, weather and load history are fed as inputs to the modeling process. In the modeling process, the model is trained by giving a history of observed sequences, and then the forecasting model parameters are estimated. After estimating the parameters, the weather forecast and the model are extrapolated to produce the final forecast.

2.2.3 Modelling

A wide variety of studies in STLF have been conducted to maximize forecasting accuracy. The modeling process can be divided into (1) the specific modeling technique used, and (2) the overall methodology framework in which the modeling techniques are used. The modeling techniques that have been adopted or developed for the STLF problem, which can be considered as a time-series forecasting problem, can be classified as traditional statistical methods or artificial intelligence (AI) based methods. Traditional methods introduced linear models for load forecasting, such as regression and time-series analysis [169], [105], [85]. In contrast, AI based methods such as neural networks [92], [129],

support vector machines [44], k-nearest neighbors [52], and fuzzy logistic models [181] are able to approximate the non-linear relationships between load and input variables. Methodology refers to a general framework that can be implemented with multiple techniques. For instance, a feature selection methodology can be applicable to both regression and neural network techniques. So, the main focus of methodology papers is investigating the most appropriate input variables for each problem and trying to interpret the relationship between the input variables and the electricity load consumption. For example, the effect of different weather variables like temperature and relative humidity on the load is investigated in [127], whereas the effect of wind speed and humidity is studied through the linear transformation of temperature in [128]. Generally, the electric load is mainly driven by weather and human behaviour. The effects of weather are considered by input features such as temperature, relative humidity, and wind speed. The effects of human behaviour are considered by calendar variables, e.g, working days, non-working days, holidays, etc. While both methodologies and techniques are essential for load forecasting, the literature has been focused on studying various techniques and their combinations rather than studying load forecasting methodologies.

2.2.4 Requirements

The ultimate goal of forecasting is to minimize the difference between the actual observations and their forecast values. Over forecasting refers to the condition when the forecast values are consistently higher than the actual observations. In contrast, under forecasting occurs when the forecast is lower than the actual observations. A typical measure for determining over or under forecasting is the arithmetic mean of the forecast errors during the time horizon of interest. Since load forecasting has a fundamental role in power grid, over or under load forecasting has various effects on the operation and planning of the power system. In [107] the authors discussed the impact of over and under forecasting in a power grid by application. In the following we summarize these impacts according to different planning or operation tasks.

Generation Planning

Over Forecasting: (i) Investing in more or higher capacity generators. (ii) Higher cost with better reliability. **Under Forecasting:** (i) Investing in fewer or lower capacity generators. (ii) Lower cost with worse reliability.

Transmission Planning

Over Forecasting: (i) Investing in more or larger capacity of transmission lines than required. (ii) Higher cost with better reliability. **Under Forecasting:** (i) Investing in fewer or smaller capacity of transmission lines than required. (ii) Lower cost with worse reliability. (iii) More congestion in the area with significant under estimation.

Distribution Planning

Over Forecasting: (i) Investing in more distribution substations or higher capacity than required. (ii) Constructing more distribution lines than required. (iii) Higher cost with better reliability. **Under Forecasting:** (i) Investing in fewer distribution substations or lower capacity than required. (ii) Constructing fewer distribution lines than required. (ii) Lower cost with lower reliability.

Demand Response

Over Forecasting: (i) More load to be shifted and more customers to be considered to participate in grid management than required. **Under Forecasting:** (i) Less load to be shifted and less customers to be considered to participate in grid management than required.

2.3 Load Forecasting Methodologies

Many papers in the load forecasting literature present a methodology to solve the problems that are introduced in the load forecasting area. These methodologies can be applied to different forecasting techniques. In this section we review some of the main method-

ologies, ranging from classical ones such as weather station selection and similar day method, to more recent ones such as load profiling and clustering based methods.

2.3.1 Feature Selection

For multivariate techniques that rely on explanatory variables, one of the important steps is determining the input variables. In [105] Hong introduced a feature selection method and applied it to three techniques, including MLR, ANN and fuzzy regression. The author showed that by applying the proposed feature selection methodology, the forecasting accuracy was improved for the three techniques. In [214] the authors presented a big data approach for feature selection, where a large number of lagged and moving average temperature variables were used to improve the forecast accuracy. Some other papers [64, 157] presented a step-by-step improvement in accuracy by adding salient features to the model.

2.3.2 Similar Day Method

"Similar day" method is one of the first methods that have been applied for load forecasting [102]. The idea behind this method is to train the model using historical data and find a day in the data that is similar to the day being predicted. Similarities can be measured based on weather patterns, season of the year, or day of the week. Nowadays, clustering algorithms are used for implementing the similar day method. The algorithms extract groups of similar days or similar segments of a day, and then they are merged to forecast the load profile.

2.3.3 Weather Station Selection

One of the critical tasks in electricity demand forecasting is finding the relevant weather stations for a region of interest. Selecting the relevant weather station can help to improve the load forecasting accuracy. Hong et al. [109] initiated this research problem in the load forecasting area. The authors considered two case studies including data from the NCEMC (North Carolina Electric Membership Corporation) company, the GEFCom2012 data set. A weather station selection framework based on MLR models were applied in

both case studies, however, the framework can be produced based other techniques as well, such as semi-parametric models and ANN. The framework introduced three steps for weather station selection: (1) ranking the individual weather station, (2) combining weather stations based on a greedy algorithm, and (3) ranking the combined group of stations obtained from the previous step. In [107], the authors applied the same weather station selection methodology presented by Hong et al. [109], for the long term load forecasting model in several states in the USA.

2.3.4 Hierarchical Load Forecasting

Key components of smart grid technology are smart meters, which provide granular spatial and temporal data. As a result, one of the important research topics that has been raised by the load forecasting community is how spatial and temporal hierarchies can be utilized to improve the accuracy of load forecasting.

Hong [104] proposed a hierarchical method for spatial load forecasting using a data set from a medium-sized utility in the USA. This approach used constrained multi-objective optimization technique to fit the S-curves for 3460 small regions and their aggregated levels.

Fan et al. [65] introduced a short-term multi-region load forecasting method based on a support vector regression model. The research aimed to improve load forecasting at the aggregated level by extracting the optimal combination of the region partitions under diverse weather and load conditions. The authors used the data from an electric utility in the Midwest USA, and showed that by applying multi-region load forecasting they could improve the accuracy of load forecasting at the aggregated level.

Lai et al. [134] conducted an empirical research study by applying hierarchical load forecasting methods on the ISO New England data. The authors studied several different ways of utilizing weather data into the load forecasting model including clustering and averaging weather stations, and grouping loads.

While the number of studies on spatial/geographic hierarchical forecasting are limited, there are many papers that focus on temporal hierarchical forecasting. In temporal hierarchical forecasting, 24 different models are used to provide forecasts for each hour of a day (e.g., [128]). Note that all the hierarchical methods mentioned here can be

implemented by any forecasting technique, such as regression models, semi-parametric models, ANN, etc.

Over the last decade of modernization of the power grid and with the deployment of smart meters and related communication devices, new challenges have been introduced to power system load forecasting. The IEEE Working Group on Energy Forecasting organized the Global Energy Forecasting Competition 2012 (GEFCom2012) to address some of the challenges in the modernized power grid. One of the tracks of the competition was on Hierarchical Load Forecasting, to utilize more granular spatial data for improving the accuracy of load forecasting at the grid level. In [106] the authors reviewed the methodologies applied by the top ranked group of the GEFCom2012. In summary, in the HLF track of GEFCom2012, the top ranked groups used regression and gradient boosting machines models beside applying some pre and post-processing methods such as modeling holidays, data cleaning and combining forecasts.

2.3.5 Load Profiling and Clustering Based Methods

Load profiling is the process of extracting information about the electricity consumption patterns of each consumer or a group of consumers. These kinds of information are used for demand response programs and load forecasting. In fact, having accurate knowledge of the volatility and uncertainty of the consumer's load profiles is of importance for deeper load analysis. In this section, the research studies on load analysis are reviewed from the perspectives of load profiling and clustering.

Load profiling refers to the segmentation of consumer's demand or load curves according to their electricity consumption behavior. The literature on load profiling can be divided into two categories including: indirect clustering based methods and direct clustering based method. A wide variety of clustering techniques, such as K-means, hierarchical clustering and self-organizing maps (SOM), have been applied directly to smart meter data [49, 219, 230]. In this section, first two fundamental research problems on direct clustering approaches will be reviewed. Then, the methods for indirect clustering, including load characteristics, dimensionality reduction and uncertainty-based methods, will be discussed.

There are several fundamental research problems associated with direct clustering

methods. The first research problem is how the resolution of the smart meter data affects the accuracy of clustering. In [81], the authors applied three popular clustering techniques, namely, k-means, hierarchical clustering algorithms, and the Dirichlet process mixture model (DPMM) on smart meter data with various resolutions ranging from 1 minute to 2 hours, to investigate how the temporal resolution of the data impacts on the results of clustering. Based on the results in this paper, the authors concluded that the smart meter data with a resolution of at least 30 minutes is reliable and sufficient for most of their purposes. The second research problem is about what kind of clustering techniques are suitable for use on smart meter data, which are basically time-series data. Traditional clustering methods were designed for static data, so some changes are required for applying these methods to time-series data. In [23], the authors introduced a dynamic version of k-means clustering to segment the load profiles of Spanish customers and capture the trends of clusters of customers. In [9], the authors offered a two step computationally efficient clustering technique. In the first step, K-means was implemented to create locally representative load profiles; then in the second step, clustering was further implemented on the clustering centres achieved in the first step. In this approach, the clustering method can be carried out in a distributed way and reduce the overall complexity.

In addition to direct clustering, many papers have focused on indirect clustering approaches. One of the effective techniques for addressing the high dimensionality problem of smart meters is dimensionality reduction. Dimensionality reduction techniques are applied on high dimensional data to make the data suitable for clustering in terms of computing similarities/distances on the most relevant features. In [131], the authors applied Principal Component Analysis (PCA) on yearly load profiles to reduce the dimensionality of smart meter data with hourly resolution, and then they used k-means clustering for customer segmentation. In addition, they used the components learned by PCA to identify the consumption pattern of different types of connection nodes. In [43], the authors applied PCA to identify the temporal load behaviour of each customer and spatial load profile of several customers. In addition, they proposed a modified version of the k-medoids algorithm based on the Hausdorff distance and Voronoi decomposition method in order to extract the typical load profiles and identify the outliers.

They tested their method on a real-life data set with a resolution of 15 minutes from a Southern California campus micro-grid to prove the effectiveness and efficiency of their proposed methodology. In [210], Varga et al. proposed the use of deep learning based stacked sparse auto-encoders for dimensionality reduction of load profiles and feature extraction. The authors used the compressed load profiles from the previous step and applied a locality sensitive hashing (LSH) method to classify load patterns and extract a representative profile.

Understanding the details of the local and global patterns of smart meter data are essential for identifying meaningful typical load patterns. In [10], the authors proposed three new computationally efficient feature extraction methods, including conditional filters on time-resolution-based features, calibration and normalization, and using profile errors to cluster daily electricity usage patterns. In [175], the authors applied subspace projection methods (subspace clustering and projected clustering) to extract the local and global pattern variability of customers' behaviors in order to identify the customers' contact type. In addition, the authors showed that by applying subspace projection methods the clustering process was more resilient to random noise than traditional methods. In [84], the authors introduced a method to identify the maximum energy consumption and major variability sources in residential electricity consumption behaviors. They proposed four key times periods (breakfast, daytime, evening and overnight) for analyzing the data and extracting the relevant features for clustering. A finite mixture model-based clustering method was presented and the robustness of the method was verified using a bootstrap technique.

The uncertainty and variability characteristic of smart meter data have been also examined for load clustering. In [199], the authors modeled the load variation by a log-normal distribution and then they applied a Gaussian mixture model (GMM) to cluster the load profiles and extract the dynamic behavior of customers. In [202], the authors proposed a C-vine copula based mixture model for clustering residential electricity consumption patterns. They used the C-vine copula algorithm to model the high dimension non-linear correlation among the electricity demand of various time periods. The authors showed that their proposed method has efficient performance for large data sets. In [218], the authors introduced a method for clustering the load profiles based on their dynamic

behaviors, where "dynamic behaviors" refer to relations and transitions between load profiles in adjacent periods. A time-based Markov model was used to model the dynamic characteristic of load profiles and, as a result, the large data set of load profiles was transformed into various state transition matrices. Then a clustering technique was applied to identify representative consumption patterns. The authors showed that their distributed clustering approach had lower computational complexity, which makes it suitable for big data applications.

2.4 Load Forecasting Techniques

Having described the general methodologies that have been used for load forecasting in the previous sections, we now give an overview of the load forecasting techniques that can be used in those methodologies. Load forecasting techniques can be categorized in to two groups: statistical techniques and artificial intelligence techniques.

2.4.1 Statistical Techniques

In this section, four statistical techniques are reviewed, including: Multiple Linear Regression (MLR), semi-parametric additive, auto-regressive and moving average (ARMA), and exponential smoothing models.

Multiple Linear Regression Models

Regression analysis is a statistical technique, for estimating the relationships among variables [133]. Multiple Linear Regression (MLR) models have been used widely in the literature for both STLF and LTLF. In these models the weather and calendar variables are considered as input variables and the load or some transformation of the load is considered as target variable. In MLR models, a functional form among the variables should be specified by the user and the parameters of these models are calculated using the ordinary least squares method. Some kind of MLR models such as polynomial regression models can describe the nonlinear relationship between the dependent and independent variables.

In [169], the authors presented a regression-based approach to STLF. The Pacific Gas and Electric Company's data was used as a case study for day ahead peak and hourly load forecast. The paper offered a comprehensive study on regression analysis for STLF. The authors introduced several MLR models for STLF, such as the weighted least squares technique, using binary variables for modeling holidays, using heating and cooling degree functions for modeling the temperature, and robust parameter estimation methods.

In [179], the authors produced 24 regression models, one for each hour of the day. Each model included a dynamic forecast error of the correctness of previous hours. The research was conducted as part of a competition organized by the Electric Power Research Institute (EPRI), which used the data from a utility in the northwest of the US. The accuracy of the model proposed in the paper outperformed the other competitors' models.

In [105], Hong proposed a regression based model to STLF, wich emphasized the impact of weather and calendar variables on the load. The case study used utility data from the USA, which implemented the proposed MLR models in its production environment. The author modeled the impact of several input variables on the load, such as the effect of lagged weather variables, weekend and holiday effects by using regression analysis. Three different models including linear regression, ANN, and fuzzy regression were applied to the case study, and it was concluded that the linear models showed the best accuracy among the competitors.

In [110], the authors presented a linear regression model for long term load forecasting. They first produced a STLF model using a linear model with a macroeconomic indicator as one of its input variables. Then a long term probabilistic load forecast model was generated by applying the STLF model to various scenarios. The authors concluded that the models based on hourly data have better forecasting accuracy than those based on monthly or daily data.

In [41], the authors proposed a refined parametric model for STLF. The case study was conducted as part of the Global Energy Forecasting Competition 2012. The proposed model used the temperature and calendar as input variables to the model and applied a series of model refinements, including removing outliers from the historical data, and combining models from multiple weather stations, to improve the accuracy of the STLF

model.

In [214], Wang et al. presented an extended version of the research proposed by Hong [105] about including a lagged weather variable. The authors utilized increased computing power and included a large number of lagged temperature and moving average temperature variables in the MLR models to enhance the load forecast accuracy. In addition, they showed that developing 24 models with one for each hour of a day may not result in smaller forecasting errors than one model for all 24 hours.

Semi-Parametric Additive Models

The semi-parametric additive model is a kind of regression model that is able to take into account some non-linearity and serially correlated errors. These models use a combination of the non-linear and non-parametric attributes within the framework of additive models [187]. In the concept of load forecasting, the relationship between the target variable (load) and input variables, such as temperature and calendar variables, can be estimated by these generalized additive models.

Hyndman et al. in [115] presented two techniques for forecasting the long term peak demand in South Australia, including a linear technique for forecasting the annual median demand, and a semi-parametric technique for forecasting the demand with a resolution of every 30 minutes. They removed the major industrial loads and then used a logarithmic transformation of the load. In addition, they used calendar, temperature, demographic and economic factor variables as the explanatory inputs to a semi-parametric model. The model was divided into two parts: (1) A regression model based on using the economic variables, seasonal demographic variables, and degree days. (2) A non-parametric model based on the other remaining variables with a resolution of every 30 minutes. Finally, the models were fed by the simulated temperatures to produce the density forecasts.

In [64], the authors used a non-parametric additive model to forecast the load in the Australian national electricity market. They provided the STLF model based on temperature effects, calendar effects, and the lagged load variables to capture the time serial correlation with the demand series.

Goude et al. in [80] applied generalized additive models to provide the short- and

middle-term demand forecast for more than 2200 substations of the French distribution network. They used some explanatory variables such as temperature and calendar variables in a generalized additive model to estimate the load for the French grid. The results showed their methodology produced good accuracy on their case study.

In [157], the authors applied semi-parametric additive models to the case study of the Global Energy Forecasting Competition 2012. They presented a temporal multi-scale model that incorporated three components: (1) A long term trend approximated by a non-parametric smoothing model. (2) The relation between electricity demand and temperature at each time step was fitted by a generalized additive model. (3) The local behaviours were modeled by a random forest model.

Exponential Smoothing Models

Exponential smoothing assigns weights to past observations that decrease exponentially over time [113, 114]. The model does not require explanatory variables, so it needs less data for the model than other techniques such as MLR and ANN. In [205], Taylor et al. conducted a comparative analysis among different models for day ahead forecasts to identify the best model. The analysis was conducted among five techniques, including periodic AR modeling, auto-regressive integrated moving average (ARIMA) modeling, an extension of Holt-Winters exponential smoothing for double seasonality, an alternative exponential smoothing formulation, and a principal component analysis (PCA) based method. The data set used in this study was based on intra-day electricity demands from 10 European countries over a period of 30 weeks. However, the drawback of this paper is that it just focused on univariate models, which did not rely on weather variables. The authors excluded multivariate models from the comparative study by citing a paper published in 1982, which declared that multivariate models were impractical for online short-term forecasting systems, in spite of the fact that regression analysis and ANN had been being used for STLF in the field for many years.

The same inference can be made from the research by Taylor [204], which analysed several techniques, including exponential smoothing models, ARIMA models, and a similar day method for very short-term forecasts with a forecast horizon of $10 - 30$ minutes ahead. In both papers some kinds of exponential smoothing models, such as double and

triple seasonal exponential smoothing, produced better accuracy than the other selected models, which did not take into account weather variables. The exponential smoothing models are popular in academic papers, however, they are rarely used in the real-life STLF problem [105]. Since weather has strong impact on the electricity demand, any changes in weather conditions can lead to a change in the load profile. Therefore, the techniques that do not rely on meteorological data are often at a disadvantage.

Auto-Regressive Moving Average Models

Auto-Regressive Moving Average (ARMA) models present a parsimonious explanation of a stationary stochastic process in terms of two polynomials: (1) An auto-regression and (2) A moving average [34,36,114]. Due to the non-stationary characteristic of the hourly electricity demand series, a variation of the ARMA model, namely the Auto-Regressive Integrated Moving Average (ARIMA) model, is usually used for load forecasting purposes.

In [222], the authors presented a comprehensive study by applying various statistical techniques for load forecasting, such as regression models, exponential smoothing, auto-regressive, ARMA, and ARIMA models. Two case study were investigated using data from California ISO to represent the modeling concepts.

2.4.2 AI Based Techniques

In this section, four Artificial Intelligence (AI) techniques, namely Artificial Neural Networks (ANN), fuzzy regression models, support vector machines (SVMs), and gradient boosting machines are reviewed. Then, a high level comparison among introduced techniques is concluded.

Artificial Neural Networks

ANN models have been used extensively for load forecasting since the 1990s. By applying an ANN model, the forecaster is able to produce a forecast without needing to explicitly model the underlying physical system [86]. While in MLR models, the forecaster is required to define the functional form among the explanatory and target vari-

ables to build the model, in ANN models the forecaster does not need to do so. In such models, the forecaster trains the model based on the historical data to construct the relationship between the input variables and the target variable (electricity demand). Then the parameters from the trained model are extracted to produce the prediction. A wide variety types of ANN models have been applied for load forecasting purposes, such as feed forward neural networks, recurrent neural networks, and radial basis function networks. Many accurate results have been reported by researchers through applying ANN models with a combination of good feature selection. Hippert et al. [92] reviewed the STLF literature that used NNs techniques. The target of the literature review was to elucidate the doubts regarding the application of NNs methods on STLF. The authors reviewed around 40 representative journal papers published in the 1990s and emphasized two facts that may have led to this doubt: (i) NNs models tend to suffer from over-fitting, possibly because of either over parameterization or over training. (ii) Although all of the 40 reviewed papers used real data to test their proposed methods, most of them were not implemented systematically: some of them did not include comparison results with standard benchmarks, and some others did not meet the standard statistical requirements in reporting the error analysis. In addition, Hippert et al. provided guidances on how to design a STLF system in four steps: data pre-processing, NN design, implementation, and validation. Although the explanation was in the context of using NNs, most of the guidances are also applicable to other techniques. The ANNSTLF (Artificial Neural Network Short-Term Load Forecaster) methodology presented in [127], is the best known application of ANN models for STLF, which was implemented in a project sponsored by EPRI. The load forecasting framework has two ANN predictors, including: a base load predictor, and a change in load predictor. Finally, an adaptive combination of these two predictions was used to produce the final forecast. The ANNSTLF model and some of its improved versions have been deployed in a production environment of a large number of utilities across the USA and Canada.

Fuzzy Regression Models

Fuzzy regression models are designed to overcome some of the limitations of linear regression models, such as insufficient numbers of observations, an unclear relation be-

tween the target variable and the explanatory variables, and hard-to-validate error distributions.

In [195], the authors applied a fuzzy linear regression model to forecast the load during holidays. They used the previous load information as an input variable to the model, without considering the weather data, and the model showed good accuracy. The authors showed that by applying a hybrid model including fuzzy linear regression and general exponential smoothing, they achieved better forecasting accuracy.

In [108], Hong et al. proposed the application of a fuzzy interaction regression method to STLF purposes. Three other models were presented in the paper, including two fuzzy regression models without interaction effects and one MLR model. The results showed that the proposed approach outperformed the three other models. In addition, the paper discussed several guidances for implementing fuzzy regression based forecasting models.

Support Vector Machines

Support Vector Machine (SVM) models are supervised learning models, which have been used for classification and regression analysis. Since SVM models tend to be resistant to the over-fitting problem, they can provide good accuracy for time-series forecasting problems.

In [44], the authors presented the winning entry methodology for a competition sponsored by the EUNITE network. They used an SVM to forecast the daily peak load of the next 31 days. The model did not include any weather variable, and the authors concluded that weather variables might not improve the accuracy of Medium Term Load Forecasting (MTLF). For the first time this paper presented the application of SVM to the MTLF pattern, which led to the popularity of implementing this model in STLF problems as well.

Gradient Boosting Models

Gradient boosting is a supervised machine learning model for regression analysis, which is based on an ensemble learning method. It produces robust and accurate forecasts

based on an ensemble of weak prediction models. Gradient boosting provides flexibility in terms of choosing an arbitrary differentiable loss function for its optimization problem.

In [203], the authors proposed a load forecasting methodology based on the gradient boosting method for the Global Energy Forecast Competition 2012 (GEFCom2012). The model used various input variables, including time of year, day of week, time of day, public holidays, current and past temperature, and lagged load variables.

In [140], the authors offered a hybrid machine learning load forecasting model using gradient boosting machines and Gaussian processes for the GEFCom2012.

2.5 Managing Uncertainties In Smart Grids

In general, operating a smart power grid is influenced by numerous uncertain parameters caused by technological, environmental, economic, social and political factors. These uncertain parameters should be identified and modeled in an appropriate mathematical form, quantified in the system analysis and transferred to the tangled stake-holders in order to make decisions. A typical transmission power grid consists of buses that host generators and consumers. These buses are connected through transmission lines, transformers, and phase shifters. The main task of transmission and distribution control system is to dispatch the available energy generation capacity according to the forecast load in order to satisfy the demand-supply balance and transmission stability. There are two types of generator sources: (i) scheduled generators such as hydrological (rivers, reservoirs, etc.), thermal (coal, gas, etc.) and nuclear generators that are committed to be available for each hour at the beginning of the day; (ii) non/semi scheduled generators that comprise renewable energy sources.

Renewable energy generators are distributed, non-synchronous, weather dependent, and cannot be scheduled ahead of time, hence they disrupt the control process of demand-supply in the power grid. New real-time grid control decision techniques are needed to address the effects of these uncertainties introduced by the integration renewable energy sources in the system. Renewable generators can be modeled by stochastic techniques, which affect all the major steps in real-time operations. The first step is gathering available real-time data from all parts of the grid, which is called the Network Topology Pro-

cess (NTP). These gathered data contain complete information on the current state of the grid. Each of the following sections present a key aspect of decision making to manage the operational reserve that influences the future state of the power network.

2.5.1 Probabilistic Load Flow

One of the important parts of power system analysis is load flow analysis, which is also known as power flow analysis. The power flow analysis involves numerical analysis of the steady state of a power system in order to extract the phase angle and magnitude of the voltage at each bus, as well as the real and reactive load flowing in each line. In reality, the future state of a system cannot be predicted with 100% accuracy. There are a variety uncertainties associated with a power system, such as load forecasting errors, changes in network configuration, and generation outages. Therefore, in order to have a better understanding of the real power system, it is essential to consider these uncertainties for power flow analysis. To address this research problem, probabilistic power flow analysis has being investigated since the 1970s.

By utilizing demand and renewables prediction tools and their distributional information, samples of possible scenarios of demand and renewable generation for a horizon of the next 6 to 24 hours can be created. Power flow under each predicted scenario is calculated and the results are combined to infer power flow distributions and the associated risk of congestion in the transmission lines. It can be presented through predictions of the appearance of peak congestion, where the peak is a function of stochastic power flows. Congestion can occur in local transmission, which causes curtailment of the generation capacity, or it can occur in the entire region due to excessive generation of renewables.

In [32], the author introduced an evaluation methodology for power flow analysis by considering the uncertainty of the bus data. First, for each bus, several scenarios including different load levels were considered with the associated probabilities. Then, the equivalent set of transmission line flow values were obtained. In [11], the authors proposed probabilistic power flow analysis through defining all the nodal loads and generation as random variables. The mean and standard deviation values of each load flow were calculated and the general balance of load flow in the system was obtained as a density function. The results were compared with those of deterministic methods and it

was shown that the vast variety of information gained through applying the probabilistic method is essential for quantitative assessment of the reliability and security of the power system.

Monte Carlo simulation has been also considered as one of the ways for evaluating the probabilistic power flow. This method is computationally expensive because it involves running various cases of deterministic power flow, however, it produces quite accurate results since it directly uses the exact power flow equation. These simulation results are usually considered as a benchmark for comparison with other probabilistic power flow methods [12].

In [47], the authors reviewed the literature of probabilistic load flows. The authors covered the basic methods including the two mentioned above. In addition, they presented techniques for improving the accuracy and efficiency of these basic methods. The application of the probabilistic load flow, such as voltage control, systems planning, and integration of distributed generation were discussed as well.

2.5.2 Contingency Analysis

Contingency analysis is a key factor of any grid operation system. It uses a power flow analysis model and analyzes scenarios under different possible contingency conditions of the network. A typical practice in grid management is to consider ”N-1” contingency scenarios, where in each scenario one critical generator or branch is assumed to fail and be taken off-line. The power flow suitability and grid stability is analyzed in the power flow step, after that the reliability of the grid operation system can be tested through examining the dispatch and flow decisions against unexpected changes in the network topology.

With integration of renewables, the contingency scenarios must be analyzed through using stochastic models. In fact, the stochastic models should be applied to carefully combine the contingency scenarios (unexpected changes in the network topology) with the renewable scenarios (unusual renewable generation outcomes) in order to create a risk-weighted analysis of the grid’s stability.

2.5.3 Unit Commitment

Unit commitment controls the balance between supply and demand by determining the number and kind of generators requires for producing the electricity power. In fact, unit commitment is a kind of optimization problem that minimizes the costs according to various limitations on the unit and the system. Popular unit commitment techniques include, Lagrangian relaxation, heuristic searches, mixed integer programming, and dynamic programming.

In [233], the authors proposed a method for investigating the impact of load uncertainty on the unit commitment risk. This method measures the probability of not having sufficient committed capacity to balance for unit failures and unpredictable load variation. The risk was estimated using a Markov model for unit failures modeling and a Gauss-Markov model for load uncertainty modeling. This paper introduced the quantitative presentation of the impact of load uncertainty on the unit commitment risk for the first time.

In [59], the authors presented a method for analyzing the effect of STLF uncertainty on the short-term unit commitment. They used a Bayesian method for producing load forecasts for one- to five-days-ahead. The risk was evaluated based on the forecast variance and calculated by determining the expected cost of the uncertainty owing to load forecast variation. They used a case study, with utility-derived power system data and temperature forecast data from the US National Weather Service.

In [209], the authors investigated the influence of the short-term load forecast uncertainty and generator availablities on the production cost and marginal cost. Several load models were considered for the study including a Gaussian model that assumed a normal distribution for the load, and Gauss-Markov regression model that assumed the load was driven by temperature and followed a Gauss-Markov process. Two years of hourly load and temperature data were used as a case study. The impact of uncertainties from load and generator availabilities on the variance of production costs was estimated by using Monte Carlo simulation. It was concluded that by considering the correlation among hourly loads and using the temperature forecast as input to the load model, the uncertainties from the load forecast was decreased, which resulted in better estimation of the expected production costs.

In [94], Hobbs et al. discussed the value of improving forecast accuracy in terms of unit commitment costs reduction. While most papers simulated the forecasting errors by stochastic models or by making some pre-assumption about their probability distribution, the authors used the actual forecasting errors for this study. It was concluded that for a 10 GW utility company, a 1% improvement in the forecast accuracy could result in saving up to 1.6 million dollar annually.

In [225], the authors presented a stochastic model for long-term security-constrained unit commitment. Random disturbances such as contingencies of generators and transmission lines were simulated using Mont Carlo simulation and the load forecast uncertainties were modeled as a uniform random variable.

In [217], the authors introduced a full-scenario unit commitment formulation for addressing the security problem in power systems due to uncertain bus loads and volatile wind power. The full-scenario analysis was conducted by using a scenario generation and reduction method based on interval mixed integer linear programming theory. By applying the proposed method, the authors showed that the worst-case impact of a volatile node injection on the unit commitment was obtained, which resulted in a secure unit commitment for the power system.

2.5.4 Reliability Planning

One of the essential tasks for planning and operating the power system is reliability. The reliability of the power grid usually is measured using the loss of load probability (LOLP). This measure refers to the probability that the amount of generation will be not enough to supply the electricity demand.

In [97], Hoffer et al. proposed a method for calculating the production costs of a set of generating units based on considering the uncertainty of the peak load forecast and extreme load values. In this model, the authors assumed that the peak load has an exponential distribution while in the traditional LOLP method the peak load was assumed to be a fixed load level. Later in [98], Hoffer et al. proposed an improved version of their method based on relaxing the assumptions of the peak load distribution by considering more advanced distributions, such as beta, triangular and gamma distributions.

In [88], the author presented a probabilistic based method for analysing the intercon-

nection assistance between different locations in a power system. The aim of this study was to evaluate the amount of power that can be transferred from one location in a grid to another by considering the transmissions limits and reliability level of the power system. They did not consider the uncertainty of load forecast in the case study, which is one of the most important factors impacting the level of power transfer. However, it was mentioned as a future direction for research.

In [29], Billinton et al. analysed the impact of load forecast accuracy on the reliability assessment of a bulk power system. Various key factors were examined in the study such as changes in the system topology, system composition, load curtailment policies, and bus load correlation levels. The load forecast uncertainties were modeled by a normal distribution, and three uncertainty scenarios were studied with 0, 5 and 10 percentiles of the standard deviation of the forecast peak load.

2.5.5 Reserve Sizing

With the integration of intermittent renewable energy into the electric power system, more uncertainties have been added to the grid. This matter degrades the reliability of the power system operation. Due to the uncertainty and unpredictability of integrated renewable energy sources in the system, having an optimal generation reserve is becoming a major challenge for managing operational risks and enhancing the grid reliability. To address this challenge, optimal reserve scheduling methods are required. Under contingency conditions, such as some level of disruption in supply, the operational reserve is the backup of generation capacity to meet demand in a short time frame. Under normal conditions, the operational reserve is considered as the size of the largest generator plus a part of the peak demand.

In the past, a wide variety of studies examined the impact of integrating renewable energy sources on restoring the supply-demand balance in power systems, ranging from deterministic to probabilistic, as well as static to dynamic approaches [37, 93, 99, 144].

In [136], Lehtila et al. presented a static model to forecast demand and the required supply capacity of electricity using a Bayesian network.

In [185], Rocha et al. proposed a decision support model to determine the future power demand of a region. The model forecasts the demand based on a linear regression

model and analyzes the influence of climatic and economic factors on the power demand by using Bayesian Belief Networks (BBNs).

In [25], Brevani et al. proposed a control system based on BBNs for controlling electricity generation in a power system. The system was designed to deal with the power fluctuations caused by the integration of wind farms in the grid.

2.6 Conclusion

In this chapter, we reviewed literature on some of the challenges in Smart Grids, which we presented in Section 1.5 of Chapter 1, and which we address in this thesis. In particular, we reviewed studies for load forecasting, and associated challenges of managing uncertainty in some of the decision-making functions that are based on load forecasting. Although there have been a number of studies on the above mentioned topics, however, recent advances in the energy industry, including smart grid and smart meters, introduce new challenges to the power grid, such as managing and analysing of large volume of complex, high dimensional data in an efficient manner. Therefore, we devote the following chapters of this thesis to developing several advanced analytical models and decision support tools to provide actionable insights for tasks such as short-term load forecasting in smart grids. Our work addresses the challenges related to time series data forecasting in smart grids. By leveraging big data from smart meters, we introduce the hierarchical short-term load forecasting method. Then, we provide a novel shape based time series clustering algorithm for customer demand profile analysis. Furthermore, we introduce a robust and accurate ensemble load forecasting framework to address the unbalanced and non-stationary nature of the smart meter data. Moreover, we propose a dynamic stochastic decision support tool based on Dynamic Bayesian Belief Networks to model the uncertainties related to the power system due to the unpredictable nature of renewable energy sources and demand behaviour of customers.

Chapter 3

Improving Load Forecasting Based on Deep Learning and K-shape Clustering

One of the most crucial tasks for utility companies is load forecasting in order to plan future demand for generation capacity and infrastructure. Improving load forecasting accuracy over a short period is a challenging open problem due to the variety of factors that influence the load, and the volume of data that needs to be considered. This chapter introduces a new approach for short-term load forecasting using an effective new combination of clustering and deep learning methods, along with a new weighted aggregation mechanism. Our evaluation using smart meter data from a publicly available real-life dataset demonstrates the improved accuracy of our approach over existing methods.

3.1 Introduction

THE smart grid introduces the two-way flow of data between electricity suppliers and customers. In this new system, customers can play an important role in achieving energy savings by modifying their electricity consumption pattern according to the dynamic electricity market [168], and selling renewable energy on an open market [40] through demand response mechanisms. The demand response mechanism depends on the convergence of accurate electric power load measurements, renewable energy, and price forecasting. Therefore, a crucial task for utility companies in this contest is load forecasting. With accurate prediction of future demand, electric utilities can make important decisions such as generating and purchasing electric load, distributing and developing their infrastructure, and optimizing load switching in order to manage their power grid.

Any error in load forecasting can result in significant cost increases for electricity

suppliers and increase the chance of unexpected blackouts or brownouts. In contrast, accurate load forecasting can result in improvements to network reliability. Therefore, trying to produce more accurate load forecasting models has become a major research challenge for energy suppliers, energy marketers, financial markets, and other parties that contribute to electric power generation, distribution and transmission.

In this chapter, we propose a method for load prediction using Hinton and Salakhutdinov's deep belief nets (DBN), which are a type of probabilistic generative neural network composed of multiple layers of restricted Boltzmann machines (RBM) [90]. In this study, two forecasting methods, based on deep neural networks and traditional neural networks, are proposed and evaluated in terms of their prediction accuracy on real life residential load data.

Key components of smart grid technology are smart meters, which provide fine-grained energy consumption information at sampling intervals of 15, 30 or 60 minutes. One of the most promising applications for such large volumes of data from smart meters is to improve the accuracy of electrical load forecasting. Typical methods for improving load forecasts using the data generated from smart meters of individual customers is based on the use of clustering [116, 193]. In this approach, the knowledge about load consumption behaviour of customers is used to improve the accuracy of forecasting. Instead of developing a single forecasting model for the accumulated load consumption of all customers, clustering can be used to divide the customers into sub-populations with similar demand profiles. Then, a forecasting model can be provided for each cluster of customers according to their load profile. Thus, a more accurate model can be provided for each cluster, and then the load forecast for all consumers can be obtained by combining the forecasts of these models. Applying clustering as an initial step in electric load forecasting has been the focus of several studies [53,150]. The K-means clustering method has been widely used in previous works for this purpose [178, 193]. However, little attention has been paid in previous studies to the choice of clustering method with the aim of improving the aggregate level of forecasting accuracy. In this section, we show that the accuracy of load forecasting depends not only on the load forecasting method, but also on the accuracy of the clustering method and the accuracy of extracting the load pattern of each cluster. In addition, we show that careful clustering of customers can result

in smaller forecasting error. We apply a novel clustering algorithm (k-shape clustering [170]) that aims to reduce the error in load forecasting through more accurate assignment of individuals to clusters based on their load consumption profile.

The **key contributions** of this chapter are the following. We propose a new method for load forecasting which leads to improved accuracy of short-term load forecasting. We show that by appropriate choice of features we can significantly improve the accuracy of forecasting. Moreover, we study a novel clustering method (K-shapes) that can cluster consumers into the groups based on their load consumption profiles, such that the accuracy of models based on each K-shape cluster is more accurate than models based on traditional K-means clustering. In addition, we implement our proposed approaches on publicly available real-life data and demonstrate the biggest improvement in accuracy is achieved by applying both new approaches, i.e., applying the new clustering method and the new forecasting method. Moreover, in contrast to previous work [17, 178, 193] that simply summed the forecasting results of each cluster to obtain the final aggregate forecast, we introduce a new weighting summation method for accumulating the forecasting results of each cluster according to the size of their membership.

The remainder of this chapter is organized as follows. Section 3.2 provides background information on short-term load forecasting and problem statements and our new approach for this problem. In Section 3.3 we discuss the new clustering method (K-shape) that we use to identify sub-populations of customers with similar behaviour. In Section 3.4 we explain the use of shallow vs deep neural network methods for learning load forecasting models. Then, in Section 3.5 we show our experimental results based on real-life smart meter data, and give a comparison of the results of our proposed method against existing methods. Finally, in Section 3.6 we conclude our study and discuss some possible directions for future work.

3.2 Background on Short-Term Load Forecasting

3.2.1 Problem Statement

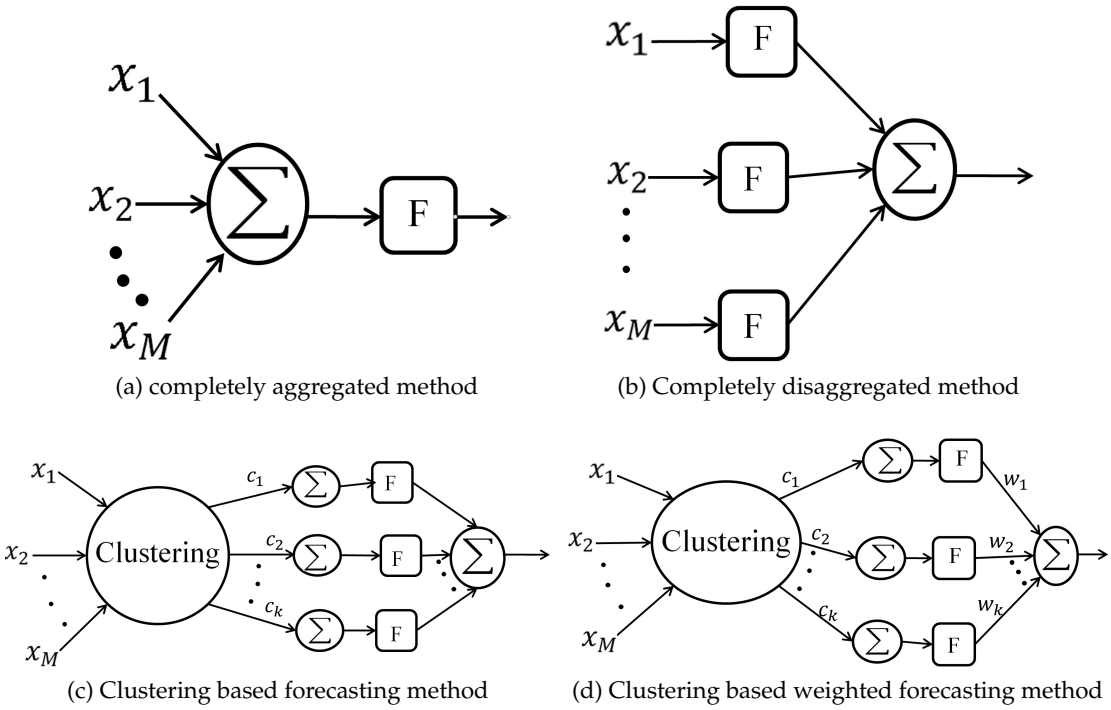
The smart meter records consist of a set of M consumers x_1, x_2, ..., x_M. The consumption history of consumer i can be defined as $x_i = \{x_i^1, x_i^2, ..., x_i^T\}$, $\forall i \in \{1, 2, ..., M\}$, where T is

the total number of historical time periods and M is the total number of consumers. Our aim is to predict the electrical load consumption for the aggregated load of the network. The total load consumption of the network can be represented as:

$$X^t_{aggr} = \sum_{i=1}^{M} x^t_i, \qquad \forall t \in \{1, 2, ..., T\}. \tag{3.1}$$

Three approaches for load prediction at the network level have been reported in the literature. The first approach is a completely aggregated method [17], which accumulates the load consumption of all consumers that belong to the network into a vector $X_{aggr} = (X^1_{aggr}, X^2_{aggr}, ..., X^T_{aggr})$ and takes this as the input feature vector for forecasting. The second method is a completely disaggregated method [17], which applies load forecasting for individual consumers, and then adds the individual predictions to obtain the prediction at the aggregated level. The third method is a clustering based forecast, where the consumers are divided into specific clusters and then the sum of the load consumption over each cluster is taken, and finally a forecast for each group is generated and these predictions for each cluster are accumulated. These three models are illustrated in Figures 3.1a, 3.1b, 3.1c. We propose a different approach in this chapter as a clustering based weighted forecasting model, which is illustrated in Figure 3.1d.

The insight of using clustering to group the individual customers is as follows. It is known that load forecasting for individual customers is complicated because each household's load depends on a variety of factors. At the granularity of half an hour, the energy consumed by a single house can fluctuate widely, due to the dependency on the number of residents in the home, the appliances that are used at the particular time, the lifestyle of the people staying in the home, and so on. Thus, accurate load prediction for an individual household can be difficult due to the individual variations in consumption patterns. To improve the overall accuracy of forecasting, an alternative approach is to group households, and analyse the aggregate level of consumption so that individual variations in consumption will tend to cancel each other out. This grouping can be achieved by the application of an appropriate clustering method, such that households with a similar load time-series are grouped together, where the total noise in each cluster would tend to cancel out.



(a) completely aggregated method (b) Completely disaggregated method

(c) Clustering based forecasting method (d) Clustering based weighted forecasting method

Figure 3.1: Different representative methods for network-level load forecasting.

We consider that by applying a clustering algorithm, k clusters $C=\{c_1, c_2, ...,c_k\}$ are obtained, so that each cluster of households in C can then be used to train a neural network. Therefore k prediction models $F_C=\{F_{c_1}, F_{c_2}, ...,F_{c_k}\}$ are generated from the k groups of consumers. To calculate the final prediction for period of interest, some studies [178, 193], simply take the sum over the predictions from the clusters as follows:

$$F_{aggr} = \sum_{i=1}^{k} F_{c_i}, \tag{3.2}$$

This summation can be used to calculate the accuracy of the forecast based on the mean absolute percentage error (MAPE):

$$MAPE = \frac{100}{T} \sum_{t=1}^{T} \left| \frac{F^t_{aggr} - X^t_{aggr}}{X^t_{aggr}} \right| . \tag{3.3}$$

This approach works well if the clusters have similar sizes. When the clusters have widely varying sizes, then better results can generally be achieved by using a weighted summation of the forecasts from each cluster. We can define these weights according to the number of members in each cluster. Assume that N_{c_i} is the number of consumers that

belong to the cluster c_i, therefore the weight of the cluster c_i can be defined as $W_{c_i} = \frac{N_{c_i}}{M}$, where M is the total number of consumers. We can then modify equations (2) and (3) as follows:

$$F_{aggr,w} = \sum_{i=1}^{k} W_{c_i}.F_{c_i}, \tag{3.4}$$

$$X_{aggr,w} = \sum_{i=1}^{k} W_{c_i}.Xaggr, c_i, \tag{3.5}$$

where X_{aggr,c_i} is the total load consumption of the consumers that belong to cluster i. Now, we can define the mean absolute percentage error as:

$$MAPE = \frac{100}{T} \sum_{t=1}^{T} \mid \frac{F^t_{aggr,w} - X^t_{aggr,w}}{X^t_{aggr,w}} \mid . \tag{3.6}$$

3.2.2 Feature Selection

The load forecasting accuracy is influenced by a variety of factors. While neural networks can fit highly non-linear models, the selection of input variables, or features, is also very important and has a major impact on the accuracy of load prediction. Figure 3.2 demonstrates the influence of human added features on the overall model performance. We considered four sets of features, namely, load variables, temperature variables, time cyclic variables, and day-type variables. A separate neural network is trained for each set of features, and the $MAPE$ of each network is shown in Figure 3.2.

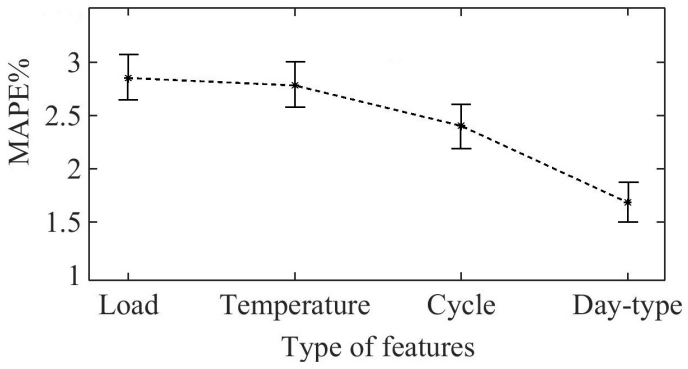


Figure 3.2: Mean absolute percentage error achieved using different types of features

Load Features

Electrical load consumption usually exhibits a daily periodicity pattern. In addition, load consumption profiles in adjacent weeks exhibit strong positive correlations. According to these characteristic we define six variables: load at time t of the previous day, load at time t-30 minutes of the previous day, load at the same day and same time of the previous week, average load of the previous day, minimum load of the previous day, and maximum load of the previous day.

Temperature Features

The weather can have a major impact on the electrical load consumption pattern. We have defined seven features as temperature variables: the temperature at time t and the same time of previous day, maximum and minimum temperature of the previous day, past 3 hours, 6 hours, and 24 hours average temperatures. Note that because the weather data set is available for an hour granularity, linear interpolations should be used for producing temperature data every 30 minutes.

Time Cyclic Features

These features reflect the cyclic characteristic of the time of day and the type of day. Cyclic variables are extracted in order to capture the cyclic nature of load time-series [161]. By spectral analysis, the half day, day, week and year periods have been identified as the dominant frequencies of the load data [34]. For each frequency, a pair of variables is considered to represent the corresponding cycles:

$$\begin{aligned} c_1(t) &= \sin(\frac{2\pi t}{T}) \\ c_2(t) &= \cos(\frac{2\pi t}{T}), \end{aligned} \tag{3.7}$$

where t is the time indicator of the half hourly granularity sampling, which extends from 1 to 17520 for one year. The variable T represents the cycle period that is equal 17520, 336, 48, and 24 for a year, a week, a day, and a half-day cycle, respectively.

Type of Day Features

Discriminating between working days and non-working days is also an important feature that should be considered. The load analysis has shown that load consumption on non-working days is lower than on working days. The non-working days (weekends and national holidays) and working days are represented by 0 and 1, respectively, as the type of day variables.

The summary of the features that have been used for training the neural networks are given as follows:

- Load features = $\{L(t-1dat), L(t-1day-30Minutes)$,$L(t-1week)$, $L_{max}(t-1day)$, $L_{min}(t-1day)$, $L_{mean}(t-1day)\}$.
- Temperature features=$\{T(t)$, $T(t-1day)$, $T_{max}(t-1day)$, $T_{min}(t-1day)$, $T_{mean}(t-3hours)$, $T_{mean}(t-6hours)$, $T_{mean}(t-24hours)\}$.
- Cycle Features = $\{\cos(\frac{2\pi t}{336}), \sin(\frac{2\pi t}{336}), \cos(\frac{2\pi t}{48}), \sin(\frac{2\pi t}{48}), \cos(\frac{2\pi t}{24}), \sin(\frac{2\pi t}{24}), \cos(\frac{2\pi t}{17520}), \sin(\frac{2\pi t}{17520})\}$
- Type of day features = $\{weekdays, weekendsandholidays\}$.

where T refers to temperature and L refers to load.

3.3 Grouping Smart Metering Data Based on Consumption Profiles

Recently, there have been a number of studies that segment customers based on their electrical consumption as a prelude to forecasting demand [39,84]. This segmentation can be performed using clustering methods on the smart meter data from customers. As we shall see, this helps to achieve a deeper understanding of the load profile characteristics as well as greater prediction accuracy.

In contrast to these previous works, we apply a time-series approach to clustering the load consumption patterns of each consumer based on the shape of their weekly profile. By clustering the time-series data, we can analyze its variations over time in detail, and search for common profiles that can be potentially useful for load prediction. Such profiles can be extracted according to the shape of the load profiles. We search for groups

of customers that show similar behavior over time. We use a novel clustering approach that aims to reduce the error in the aggregate forecast through suitable assignment of individuals to clusters based on the shape of their load profiles. In contrast to traditional clustering methods such as K-means [117, 170], which tend to cluster customers with similar average load levels, we require an approach that clusters customers based on the similarity of their load profile over the course of the day. These groups of customers with similar consumption patterns can be used to achieve a more accurate aggregate load forecast. We next describe the clustering method we have used, which is called K-shape clustering.

3.3.1 K-shape Clustering

Numerous clustering algorithms have been used for the load prediction problem. Determining the best algorithm depends greatly on the nature, purpose and mining objectives of the dataset. In this chapter, we use K-shape, a novel algorithm for time-series clustering that can conserve the shapes of time-series sequences [170]. This new time-series clustering algorithm creates consistent and well separated groups of time-series. K-shape considers the shape of the time-series during clustering, in contrast to traditional methods such as K-means, which treat the observations in a time-series as independent attributes. This novel clustering method uses a normalized, domain-independent form of cross correlation as its distance measure. Using this method, the K-shape algorithm derives a shape-based distance measure for comparing the time-series efficiently and effectively. Then, based on the properties of the shape-based distance measure, K-shape computes cluster centroids, which are used in each iteration to capture shared characteristics of the underlying data and update the assignment of time-series to clusters.

The extraction of a representative centroid for each cluster is a challenging task that critically depends on the choice of distance measure. By extracting the centroid, the clustering algorithm can effectively summarize a set of time-series in terms of only one sequence, and extract the most representative shape from the underlying data. Then, these extracted shapes or centroids are used for clustering the time-series. The robustness of K-shape clustering has been experimentally evaluated against partitional methods like K-means, K-medoid, hierarchical and spectral clustering methods, with combinations of

the most competitive distance measures [170]. It has been shown [170] that K-shape outperforms all of these approaches in terms of accuracy on time-series data. In this chapter we compare our results using K-shape and K-means methods, and it will be shown in the experimental results section that K-shape significantly outperforms K-means with respect to clustering accuracy, and as a result forecasting accuracy.

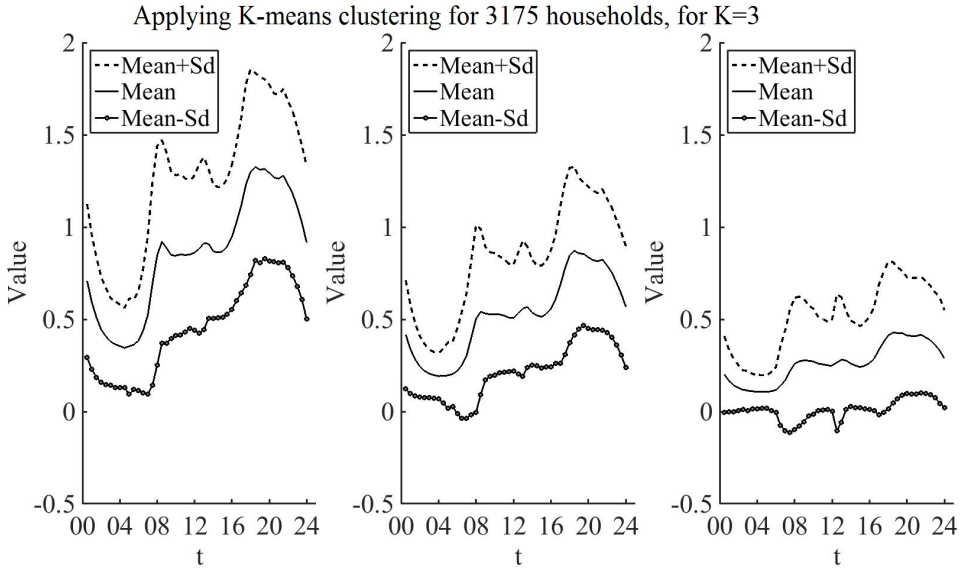


Figure 3.3: Mean, Mean plus standard deviation, and Mean minus standard deviation value for each cluster with applying K-means clustering method.

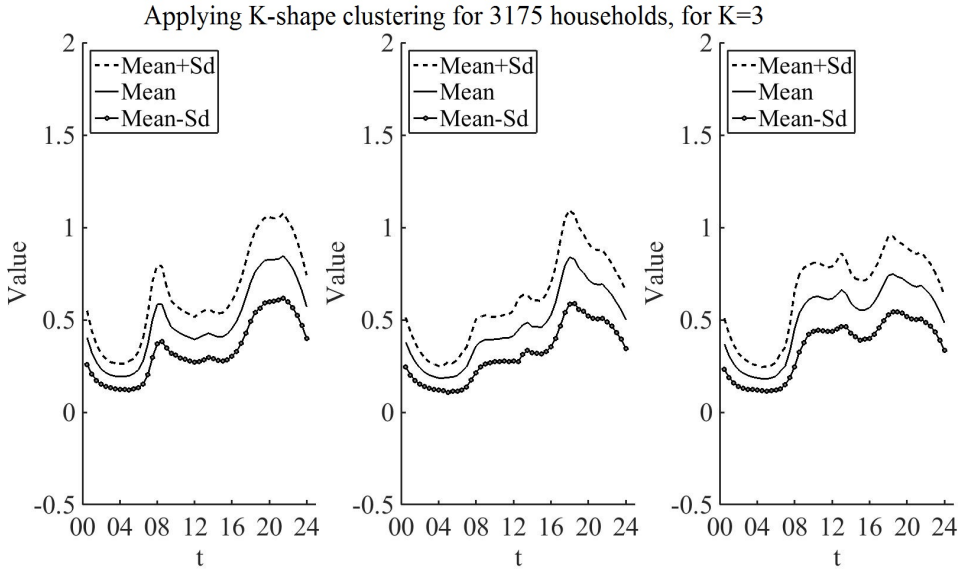


Figure 3.4: Mean, Mean plus standard deviation, and Mean minus standard deviation value for each cluster with applying K-shape clustering method.

To illustrate how K-shape clustering outperforms K-means clustering, we applied these two methods for clustering the 3175 household's load consumption our 24 hours. The clustering has been done with $K = 3$, for each method. Then the mean and standard deviation of each group were calculated in order to demonstrate the accuracy of each clustering method. Figures 3.3 and 3.4 show that our clustering approach, K-shape, outperforms K-means in terms of grouping consumers who have similar load profiles. Note that even though the input data and normalisation method was the same for both algorithms, the clustering results are noticeably different. In particular, K-means tends to group households by the average amplitude of their daily usage, while K-shape tends to group households with similar daily patterns of usage, irrespective of their average amplitude.

In summary, the K-shape method groups sequences that exhibit similar patterns into the same cluster according to their shape similarity, regardless of differences in amplitude and phase. K-shape clustering is a domain-independent, accurate, and scalable algorithm that uses a distance measure which is invariant to scaling and shifting. This clustering method can thus preserve the shapes of time-series sequences. The K-shape clustering algorithm consists of three main components: (1) a shape-based distance measure, (2) time-series shape extraction, and (3) shape-based time-series clustering. We now give an overview of each of these key components.

Shape-Based Distance

The first component is the Shape Based-Distance, which is based on a cross-correlation measure. This algorithm derives a scale- and shift-invariant time-series distance measure in a computationally efficient way. Consider two sequences $\overrightarrow{x} = (x_1, x_2, ..., x_m)$ and $\overrightarrow{y} = (y_1, y_2, ..., y_m)$. The similarity of these two sequences can be determined as follows:

$$CC_w(\overrightarrow{x}, \overrightarrow{y}) = R_{w-m}(\overrightarrow{x}, \overrightarrow{y}), \tag{3.8}$$

where $CC_w(\overrightarrow{x}, \overrightarrow{y})$ is the cross-correlation sequence with length $2m - 1$, and $R_{w-m}(\overrightarrow{x}, \overrightarrow{y})$

is computed as:

$$R_k(\overrightarrow{x}, \overrightarrow{y}) = \begin{cases} \sum_{l=1}^{m-k} x_{l+k}.y_l & : k \geq 0 \\ R_{-k}(\overrightarrow{y}, \overrightarrow{x}) & : k < 0 \end{cases} \tag{3.9}$$

For scaling invariance, this algorithm uses the z-normalization method, where each sequence $\overrightarrow{x}$ is transformed into $\overrightarrow{x}' = \frac{\overrightarrow{x}-\mu}{\sigma}$, where μ is the mean and σ is the standard deviation. In order to address the shift invariance problem this algorithm uses coefficient normalization, which generates values between $[-1, 1]$. By coefficient normalization, the cross-correlation sequence is divided by the geometric mean of the autocorrelation of the individual sequences. So, we can define the normalized cross-correlation coefficient NCC_c as follows:

$$NCC_c(\overrightarrow{x}, \overrightarrow{y}) = \frac{CC_w(\overrightarrow{x}, \overrightarrow{y})}{\sqrt{R_0(\overrightarrow{x}, \overrightarrow{x}).R_0(\overrightarrow{y}, \overrightarrow{y})}}. \tag{3.10}$$

The goal is to find the position w where $NCC_c(\overrightarrow{x}, \overrightarrow{y})$ is maximized. Thus the Shape Based Distance (SBD) can be calculated as:

$$SBD(\overrightarrow{x}, \overrightarrow{y}) = 1 - \max_w(NCC_c(\overrightarrow{x}, \overrightarrow{y})), \tag{3.11}$$

which yields values between 0 to 2, with 0 indicating perfect similarity for time-series sequences.

Time-Series Shape Extraction

The second component of K-shape clustering is defining a centroid that summarizes a set of time-series. Extracting the shape of a set of time-series, which is able to capture the shared characteristics of the underlying data, is a difficult task that depends on the SBD distance measure. To achieve accurate shape extraction, this algorithm casts centroid computation as an optimization problem, where the objective function is to find the maximizer μ_k^* of the squared similarities to all other time-series sequences. So, we can formulate the optimization problem as follows:

$$\overrightarrow{\mu}_k^* = \arg\max_{\overrightarrow{\mu}_k} \sum_{\overrightarrow{x_i} \in P_k} (NCC_c(\overrightarrow{x}, \overrightarrow{\mu_k}))^2, \tag{3.12}$$

where P_k is the k^{th} partition and $\overrightarrow{\mu}_k$ is the initial centroid for the k^{th} partition. This equation uses the previously computed centroid as a reference and aligns all time-series according to this reference sequence in each iteration of the clustering algorithm. To align the sequences with the reference sequence, this algorithm uses the SBD measure, which identifies an ideal shift for every $\overrightarrow{x_i} \in P_k$.

Shape-based Time-Series Clustering

The third and final component is K-shape clustering, which relies on the SBD distance measure and the shape extraction method to efficiently produce clusters of time-series. The K-shape clustering algorithm takes the time-series set X and the number of clusters K as inputs. Then, this algorithm assigns the time-series in the set X to the K clusters randomly. In the next step, it computes each cluster's centroid using the Shape-Extraction algorithm. Then, once the centroids are computed, it refines the memberships of the clusters by using the SBD distance measure. The K-shape clustering algorithm repeats these steps until the algorithm converges, which occurs when there is no change in cluster memberships or the maximum number of iterations is reached (usually a small number, such as 100). The outputs of this algorithm are the assignment of the time-series to clusters, and the centroids for each cluster which represent the shared characteristics from the underlying data. In addition, each centroid depicts the most representative shape of each group, which is the load profile for each cluster.

3.4 Deep-Learning vs Shallow Neural Network

Neural networks have been widely used for electric load forecasting. this approach attempts to learn a complex nonlinear relationship between the inputs and the outputs. Recently, deep learning neural network theory has developed rapidly and achieved improvements over traditional shallow neural networks. Three aspects, namely, better parameter initialization techniques, having a large number of hidden layers and better learning algorithms, have influences the success of deep learning methods [207].

Shallow architectures with large hidden layers can suffer from the over-fitting problem [164, 167]. In contrast, deep architectures with a large number of hidden layers and

many levels of non-linearity can represent complex features from their input, and learn these features with models that have better generalization. Consequently, a shallow architecture can result in a model that does not generalise well, unless trained with very large data sets, thus increasing the computational resources required. Recent research has shown that when the problem has highly nonlinear properties, deep networks can be more efficient to train than shallow networks [20, 190, 207]. In general, deep architectures can include multiple layers of representation and abstraction to help model and generalise from a complex dataset. This enables deep architectures to provide a compact representation for a much wider range of functions than is possible using shallow architectures. A thorough description of shallow and deep architectures is presented in [21], and the interested reader is referred to that paper for further information.

In this study we use both the feed forward Multilayer Perceptron (MLP) and the Restricted Boltzman Machine (RBM) models for load prediction, and compare their results in terms of prediction accuracy. In the following, we first briefly introduce neural networks, and then introduce the deep learning method based on the Restricted Boltzman Machine (RBM).

3.4.1 Artificial Neural Networks

We executed the tests using a feed forward multilayer perceptron architecture with one input layer, two or three hidden layers and one output layer. The input consists of a vector of various features extracted from the load, temperature, and calendar variables. Each hidden layer receives a vector of inputs from the previous layer and converts it to its output vector using a linear transformation followed by a nonlinear activation function. Given a set of training data with corresponding target values $D = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), ..., (x^{(T)}, y^{(T)})\}$, for each pair $(x^{(i)}, y^{(i)})$, $x^{(i)}$ is the input to the network and $y^{(i)}$is its label. The output state for each neuron, with N_c input connections, can be calculated as:

$$y = f_a \left(\sum_{j=0}^{N_c} W_i X_i \right), \tag{3.13}$$

where W_i and X_i are the weight and input vectors respectively, and f_a is the activation function, which is the sigmoid function in this study, $f_a(z) = sigmoid(z) = \frac{1}{1+e^{-z}}$.

Let vector $o^{(i)}$ be the corresponding output of the neural network and vector $x^{(i)}$ be the input feature vector. In training the neural network, the output value $o^{(i)}$ should be close to the actual label $y^{(i)}$. The network adjusts its weights to obtain a good estimation of $y^{(i)}$ by minimizing the square error function $E = \frac{1}{2}\sum_{t=1}^{T}(o^{(t)} - y^{(t)})^2$. In the optimization process, the resilient backpropagation algorithm [83] is used, which is based on the traditional backpropagation algorithm. In contrast to traditional backpropagation, the resilient method can find better network parameters and accelerate the optimization process. In order to modify the weights of networks for finding a local minimum of the error function, resilient backpropagation allocates a separate learning rate for each weight, which can be changed during the training process.

3.4.2 Deep-Learning Based on Restricted Boltzman Machine and Deep Belief Networks

Neural networks with multiple hidden layers can be very sensitive to initialization, which can lead to much better or much worse results after training [19]. To reduce this sensitivity, Hinton et al. introduced a greedy layer-wise pre-training method [21, 91] by a combination of Restricted Boltzman Machines (RBM) to address the drawbacks of traditional neural networks.

A Restricted Boltzmann Machine (RBM) can be defined as a generative stochastic neural network model that learns a probability distribution over the input vectors. The standard type of RBM consists of binary-valued hidden units, visible units, a matrix of weights W_{j_k} corresponding to the link between hidden unit h_j and visible unit v_k, and bias weights a_k and b_j for the visible and hidden units, respectively. In addition, the RBM is a special type of energy based model that can be defined as:

$$\begin{aligned} E(v,h) &= -h^T W_v - a^T v - b^T h \\ &= -\sum_k a_k v_k - \sum_j b_j h_j - \sum_{jk} W_{jk} v_k h_j, \end{aligned} \tag{3.14}$$

where the joint distribution over the visible and hidden units is defined by $p(v,h) = \frac{e^{-E(v,h)}}{Z}$, where Z is a normalization factor.

The visible and hidden units are conditionally independent given one another. Accord-

ingly, the conditional probabilities $p(h \mid v)$ and $p(v \mid h)$ can be written as:

$$p(h \mid v) = \prod_j p(h_j \mid v)$$
$$p(v \mid h) = \prod_k p(v_k \mid h). \tag{3.15}$$

If the values of v and h are limited to the set $\{0, 1\}$, the conditional probabilistic model can be defined as:

$$p(h_j = 1 \mid v) = sigmoid(b_j + \sum_k W_{jk} h_k)$$
$$p(v_k = 1 \mid h) = sigmoid(c_k + \sum_j W_{jk} h_j). \tag{3.16}$$

DBNs are multi-layer generative models that learn one layer of features at a time from unlabelled data. The extracted features are then treated as the input for training the next layer. This efficient, greedy learning can be followed by fine-tuning the weights to improve the generative or discriminative performance of the whole network. There are a range of advantageous properties that have been identified for DBNs [21]: they can learn higher-level features that yield good classification accuracy; they are parametric models, whose training time scales linearly with the number of records; and they can use unlabelled data to learn from complex and high-dimensional datasets.

DBNs are multilayer neural networks, where each layer is typically made of RBMs and can be carefully pre-trained in an unsupervised, layer-by-layer method. Each larger of the hierarchical network is trained in an unsupervised manner, where the output of one layer becomes the input of the next layer in a bottom-up manner. After the unsupervised pre-training, the DBN uses supervised learning to precisely adjust the network. DBN defines a joint probability distribution between the input x and the hidden variables at layer i, h^i, as follows:

$$p(x, h^1, \ldots, h^l) = (\prod_j^{l-1} p(h^{i-1} \mid h^i)) p(h^{l-1}, h^l), \tag{3.17}$$

3.5 Case Study Applications

3.5.1 Input Data

The real-world smart meter records from the Commission for Energy Regulation [5] are used in this study. This dataset was recorded in Ireland from July 14, 2009 to December 31, 2010 with more than 6000 Irish consumers at a resolution of 30 min. The dataset is divided in to three groups of consumers: residential, small-to-medium enterprises, and others. In this study only the residential group of consumers are studied, which include 3176 homes and 17 months of 30 minutes interval data from August 2009 to December 2010. The first 14 months were used as the training set. One month was used for validation, and the model was tested through the last two months of data. Through data pre-processing, consumers with a large number of successive missing records were eliminated. For consumers with single missing records, linear interpolation between the records in the vicinity of the missing ones was applied to complete the dataset. Moreover, temperature data for the mentioned time periods was collected from the website: `https://www.wunderground.com`. Using these data sets, and applying the novel clustering (K-shape) and deep neural network methods, we aim to obtain the best prediction accuracy for forecasting time horizons ranging from 30 min to one day ahead.

3.5.2 Experimental Results

In this chapter, we investigate the best clustering method for grouping the load consumption data that comes from smart meters at the household level, in order to improve the accuracy of load forecasting at the network level. In addition, we study the application of deep learning to improve the performance of load forecasting. Moreover, we introduce a new method for aggregating the forecast load for each cluster to derive a network level load forecast that is more accurate than previous methods [178, 180, 193]. To evaluate the effectiveness of our method, a real-life smart meter dataset was used for testing our approach. The Neural Network and Deep Neural Network based forecasting models with K-shape and K-means clustering methods were applied to the dataset.

For K-shape and K-means clustering, the average consumption for each day of the

week during the training period of interest were obtained. In addition, for preprocessing the dataset, the standard score normalization method was applied to normalize the load into the range $[-1, 1]$. Then, the data points were arranged in a 3176-by-336 data matrix, where the dimension of the matrix refers to the number of customers and the number of features (48 segments per day for seven days of a typical week). To determine the optimal number of clusters, the MAPE at each granularity of the day for each day of the testing period was calculated with the number of clusters varying from 2 to 8. Then, the average MAPE was calculated during the whole testing period for varying numbers of clusters. Figures 3.5 and 3.6 show that the best accuracy for our dataset is obtained when $K = 5$ clusters are used in both clustering methods (K-shape and K-means). Using this value of $K = 5$, to compare the accuracy of K-means with NN, K-shape with NN, K-means with DNN and K-shape with DNN. In Figure 3.7, it is shown that K-shape with the DNN based forecasting model has the best accuracy among the four proposed methods.

For both Neural Network and Deep Neural Network methods, the features that were described in the section on feature selection have been used. Table 3.1 shows the comparative results between the proposed methods. We show the forecasting results for 24 hours ahead for each of the four methods, and it can be seen that the method using K-shape clustering with DNN has the highest forecasting accuracy for the aggregate load. Based on the forecasting results, we have demonstrated that by using our approach, load prediction accuracy can be improved by effectively clustering the customers based on their consumption pattern, and applying the forecasting model that can better generalize over the dataset. Figure 3.8 depicts the load profiles for each of the five clusters during 14 months from August 2009 to September 2010, in terms of the shape extraction in the K-shape clustering method. Note that the load profiles in each cluster are quite distinctive. In the case of the two load profiles on the right of Figure 3.8, these profiles highlight a slight change in daily usage patterns on Wednesdays and Thursdays. An interesting direction for future work is how to analyse the possible causes of such local variations in load profiles, based on contextual information about households that might be available in practice

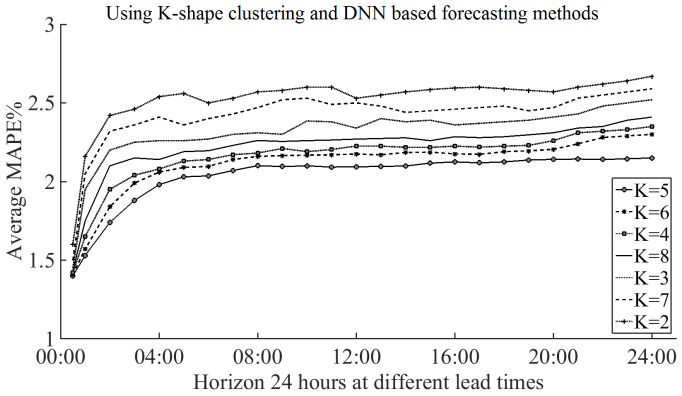


Figure 3.5: Average MAPE for one month (November 2010-December 2010) vs lead times of 30 min ahead, 1 h ahead, 2 h ahead, ..., 24 h ahead for different numbers of cluster.

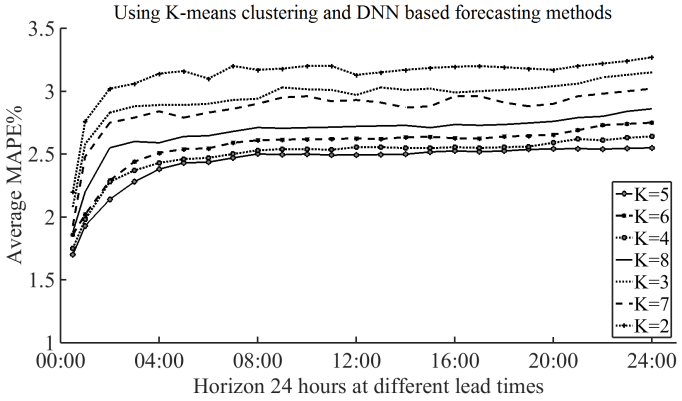


Figure 3.6: Average MAPE for one month (November 2010-December 2010) vs lead times of 30 min ahead, 1h ahead, 2h ahead, ..., 24h ahead for different numbers of cluster.

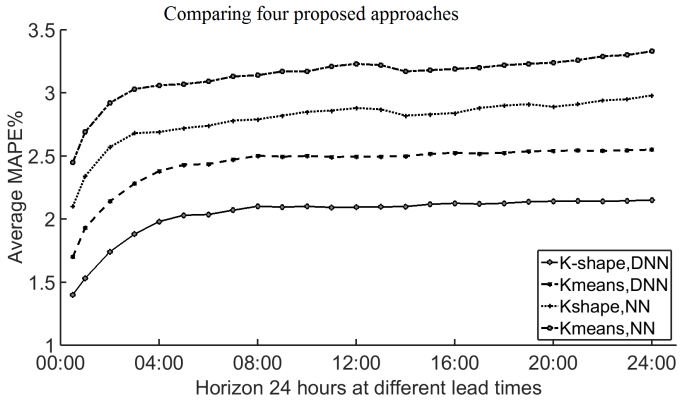


Figure 3.7: Average MAPE for one month (November 2010-December 2010) vs lead times of 30 min ahead, 1h ahead, 2h ahead, ..., 24h ahead for different proposed methods.

Table 3.1: Mean Absolute Percentage Error for different clustering and forecasting methods for lead time 24hours ahead in November 2010 - December 2010

Methods	Average MAPE%
K-shape clustering+DNN	2.15%
K-means clustering+DNN	2.55%
K-shape clustering+NN	2.98%
K-means clustering+NN	3.33%

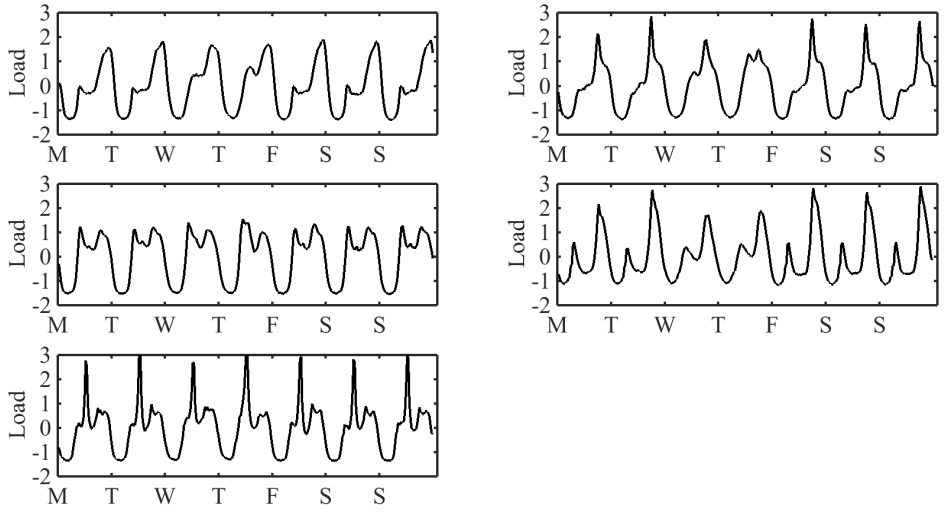


Figure 3.8: Normalized load profiles extracted by K-shape clustering when K=5, the optimal number of clusters.

3.6 Conclusion

We have presented a new approach to the problem of short-term load forecasting by using deep neural networks, which provide better generalization than traditional neural network models. In addition, we have shown that by using an appropriate clustering method we could further improve forecasting accuracy. Based on an evaluation using real-life smart meter data, we have shown that the best accuracy can be achieved when K-shape clustering is used in combination with deep neural network methods for forecasting. In addition, we have introduced a new weighting summation method for accumulating the forecasting results of each cluster according to the size of their membership.

In this chapter we showed that forecast errors can be reduced by effectively grouping different customers based on their energy consumption. We showed by applying k-shape clustering the accuracy of model is increased compared with traditional k-means cluster-

ing. In the next chapter we focus on developing a better clustering algorithm in terms of accuracy and efficiency for time-series data.

Chapter 4

Fuzzy c-Shape: A New Algorithm for Clustering Finite Time-Series Waveforms

Dealing with the high volume of time series data (smart meters data) motivated us to develop a new clustering algorithm for time-series data which is computationally efficient and accurate. Clustering is a popular data mining method due to its powerful exploratory nature and its usefulness as a preprocessing step for other data mining techniques. This chapter develops two novel clustering algorithms for time-series data that are extensions of a crisp c-shapes algorithm. The two new algorithms are heuristic derivatives of fuzzy c-means (FCM). Fuzzy c-Shapes plus (FCS+) replaces the inner product norm in the FCM model with a shape-based distance function. Fuzzy c-Shapes double plus (FCS++) uses the shape-based distance, and also replaces the FCM cluster centers with shape-extracted prototypes. Numerical experiments on 48 real time-series data sets show that the two new algorithms outperform state-of-the-art shape-based clustering algorithms in terms of accuracy and efficiency. Four external cluster validity indices (the Rand index, Adjusted Rand Index, Variation of Information, and Normalized Mutual Information) are used to match candidate partitions generated by each of the studied algorithms. All four indices agree that for these finite waveform data sets, FCS++ gives a small improvement over FCS+, and in turn, FCS+ is better than the original crisp c-shapes method. Finally, we apply two tests of statistical significance to the three algorithms. The Wilcoxon and Friedman statistics both rank the three algorithms in exactly the same way as the four cluster validity indices.

4.1 Introduction

INCREASING, large volumes of time-series data are becoming available from a diverse range of sources, including smart phones, environmental sensors and biological devices. Techniques for analyzing time-series data include dimensionality reduction

[56], indexing [122],[125], and segmentation [123]. In addition, techniques have been developed to extract the underlying shape or trends in the data, for tasks such as pattern discovery [124, 145, 174], classification [111, 184], rule discovery and summarization. An open challenge in time-series analysis is how to cluster a set of time-series according to the similarity of their underlying shape. The basic objective of this article is to generalize the fuzzy c-means (FCM) model [28] to accommodate clustering in finite time-series data.

In general, clustering is an analytic tool that can identify interesting patterns and correlation in the underlying data [87]. For example in ECG signals the goal may be to cluster the ECG signals so that normal heart signals can be distinguished from abnormal signals based on the shape and phase of the time-series signals. A key problem in finite time-series clustering is the choice of an appropriate distance measure, which can be used to analyze the similarities/dissimilarities of finite time-series. Such a distance measure should be able to address questions such as: *(i)* how to describe the underlying shape or trend in the time-series? and *(ii)* how to handle distortion in the amplitude and phase of the time-series, such as noise or jitter?

Among different distance measurement methods that have been used for time-series clustering, *Euclidean distance* (ED) and *Dynamic Time Warping* (DTW) are the most popular methods. In most of clustering techniques for time-series data, Euclidean distance is used for measuring the similarities/dissimilarities of time-series. However, this distance is most useful when we have time-series in the form of equal length records. Euclidean distance in time-series clustering only compares the points of the time-series in a fixed order, so it cannot be used for clustering time-series with time shifts. The DTW distance can be used for evaluating similarities/dissimilarities of time-series data based on their shape information. However, this method is computationally expensive, since the cost of computing the similarity/dissimilarity between each pair of time-series is quadratic in the length of the records.

Most state-of-the-art approaches for shape-based clustering suffer from two main drawbacks: *(i)* they are computationally expensive and thus, not suitable for large volumes of data [145, 159, 174, 229]; and *(ii)* these approaches are limited to specific domains [229], or their effectiveness has only been evaluated over a small number of datasets [145, 159].

k-Shape [170] is a novel method that has been proposed to address the problem of finding a suitable distance measure and clustering method for finite time-series data, where time-series sequences belong to the same cluster, exhibiting similar patterns, regardless of differences in amplitude and phase. Paparrizos and Gravano [170] select a statistical measure, cross-correlation, as a shape similarity distance measure for comparing time-series sequences. For clustering time-series data, they modify the classical crisp k-means clustering algorithm (also called *hard c-means* (HCM)) with a new centroid computation technique based on a shape extraction method while computing the clusters. These two novel modifications seem highly effective in terms of clustering accuracy and efficiency. To demonstrate the robustness of k-Shape clustering [170], the authors perform an extensive evaluation over 48 different time-series datasets and demonstrate the superiority of their technique against fifteen existing schemes.

The **key contributions** of this chapter are the following. We introduce two novel clustering algorithms. The first substitutes the *Shape Based Distance* (SBD) measure introduced in [170] for the model norm in the standard fuzzy c-means [26] algorithm, and the second uses the SBD measure and shape extraction method in [170] to update prototypes in the fuzzy c-means clustering algorithm [26]. These two modifications result in 4.5% and 58.5% improvement respectively in accuracy and efficiency in comparison to k-Shape clustering. We make a direct comparison with the results reported in the k-Shape paper, by evaluating our methods on the same UCR time-series classification archive [126], which consists of 48 labeled time-series datasets from various real-world domains.

The remainder of this chapter is organized as follows: Section 4.2 reviews related work on time-series clustering, Section 4.3 presents the relevant theoretical background on fuzzy c-means and k-Shape clustering that are based of our two new algorithms. Section 4.4 introduces our two novel clustering methods FCS++ and FCS+. Section 4.5 discusses the four external cluster validity indices used for measuring the accuracy of clustering methods. Section 4.6 evaluates our proposed methods with extensive numerical experiments, and shows that our algorithms outperform the crisp k-Shape clustering algorithm in terms of accuracy and efficiency. Sections 4.7 and Section 4.8 contain a short discussion of our conclusions and some future research directions.

4.2 Related Work

Among different techniques for analyzing time-series data, clustering is a popular data mining method due to its usefulness as a preprocessing step for other data mining techniques. There are comprehensive literature surveys on time-series clustering such as [7, 57, 183]. Generally time-series clustering algorithms fall in to two categories: Statistical-based and shape-based methods. Statistical-based methods, use characteristic measures [216], varying coefficient models [120], and subsequences of time-series as approaches for extracting features to cluster time-series data. However, there are two general drawbacks of using these methods: first, applying these methods usually leads to trial and error problems because of their requirement for adjusting multiple parameters; second, they are not domain-independent [170]. On the other hand, shape-based methods do not have these limitations and they have been used in a variety of studies. Therefore, in this chapter we focus on shape-based clustering. In the following we briefly review shape-based distance functions for time-series data as well as relevant clustering techniques that have been reported in the literature.

An efficient shape-based distance function that measures the similarities/dissimilarities between series data, should not be sensitive to the time of appearance of similar sequences [7]. Indeed, a distance function that measures the shape similarity should be invariant to phase and amplitude. Therefore, elastic methods [8], [15] like DTW [50] are more suitable for shape-based time-series clustering.

Next, we review relevant shape-based distance functions that have been introduced in previous studies. In [63] the authors introduced an indexing method using Euclidean distance in a lock-step measurement (one-to-one) manner. Sensitivity to scaling is the drawback of this method. The DTW distance has better accuracy but lower efficiency than this method. In [78] the authors proposed a cross-correlation based distance function that is very efficient for noise reduction. In addition, this distance function can summarize the temporal structure of the time-series data. In [18, 212] a non-metric similarity measure is proposed based on the *Longest Common Sub-Sequence* (LCSS), which has very good noise robustness. In this method, the similarity between trajectories is considered by giving more weight to similar segments of the sequences.

In [45] the *Edit Distance with Real Penalty* (ERP) is introduced as a metric that can

handle local shifting in time-series data. The ERP approach is a robust method against noise and is shift- and scale-invariant. In [135] the *Minimal Variance Matching* (MVM) method is proposed which is able to skip outliers automatically. MVM uses the DTW method for calculating the similarity between two time-series, but MVM, unlike DTW, can skip some target series. In addition, MVM, like the LCSS method, can allow the query sequences to match to only sub-sequence of the target sequence.

In [46] the authors introduced the *Edit Distance on Real sequence* (EDR) method, which is robust against noise, shifts, and scaling of the time-series data. This method is more robust than the ED, DTW, and ERP methods, and it is outperforms the LCSS method. EDR measures the similarity between two trajectories based on the edit distance on strings. By assigning penalties to the unmatched segments, EDR improves its accuracy, and by quantizing the distance between 0 and 1 effectively removes the noise. In [154] the *Sequence Weighted Alignment* (Swale) model is proposed to score similarities between time-series data based on rewards for matching portions, and penalties for mismatching portion of sequences.

In [57, 215] the authors conducted extensive experiments on 38 data sets from a wide variety of application domains to compare distance functions. They concluded that the *constrained Dynamic Time Warping* (cDTW) method [189] outperforms all other distance measurement methods in term of accuracy.

The *Shape Based Distance* (SBD) [170] is a novel measurement approach that handles distortions in amplitude and phase accurately, by efficiently using cross-correlation through normalization of a shape-based distance measure. According to the authors of [170], SBD achieves similar accuracy to cDTW but it does not require any tuning, and it is an order of magnitude faster. In this chapter we use SBD as the distance function in our algorithms.

Many studies have focused on partitional models of clustering using a shape-based distance function that is shift- and scale-invariant. Among partitional clustering methods, k-medoids [121] clustering has been used widely because of its ease of incorporating a shape-based distance measure [57, 215]. However, k-medoids is a computationally expensive and non-scalable method. Alternatively, other studies [145, 159, 174, 229] have focused on k-means (aka hard c-means (HCM)) [142] with various choices of centroid

computation and/or distance measure. The best performing k-means approaches have used DTW while proposing a new centroid extraction method [145, 159, 170, 174, 229].

Recently, a novel partitional clustering algorithm, k-Shape, that uses the SBD distance, was proposed in [170]. The k-Shape algorithm preserves the shapes of time-series when comparing them, and computes centroids in a scale- and phase- invariant way. The authors of [170] showed that k-Shape partitioning of 48 finite, labeled time-series data sets was superior to 15 other methods based on spectral, partitional, and hierarchical clustering approaches. k-Shape appears to be a scalable and accurate approach for time-series clustering that can be applied to a wide variety of domains.

There are also fuzzy approaches to clustering in time-series based on algorithms such as FCM. Studies based on fuzzy clustering of time-series data include [78, 151, 196]. In [151] a *short time-series* (STS) distance is proposed to measure similarities between time-series that are short in length, based on their shape. By changing the Euclidean distance to STS distance in the standard fuzzy c-means algorithm, they modified the membership matrix and centroid computation algorithm, thus producing a fuzzy time-series clustering algorithm for short time-series data. In [78] the cross-correlation method is considered as the distance measure in the fuzzy c-means clustering algorithm and arithmetic means are used for prototype computation to cluster functional MRI data. The authors in [158] used the DTW method as distance measure and applied k-means and c-medoids algorithms as clustering methods. Comparing the results of these two methods, led them to conclude that k-means cannot produce satisfactory results whereas c-medoids obtained good accuracy. In Chapter 4, we propose two novel shape-based clustering methods of time-series data based on heuristic derivatives of FCM.

4.3 Preliminaries

Many clustering algorithms exist for numerical feature vector data such as $X = \{x_1, x_2, \ldots, x_n\} \subset \Re^p$. There are three basic problems associated with clustering in X : (i) pre-clustering assessment of tendency (what value of c, the number of clusters, to use?); (ii) partitioning the data (finding the c clusters); and (iii) post-clustering validation (are the clusters useful, realistic, etc.?). Good presentations on each of the three basic problems include the

general texts [28, 61, 118, 206]. This study concerns itself with three algorithms for clustering in data where each $x_k \in X$ is a vector of length p that represents a finite portion of a waveform or time-series. Data of this type are prevalent in many fields: for example, stock market analysis [72], satellite imagery [101], computational finance [186] and neuroscience [66, 177, 188].

In this section we review the relevant theoretical background for our new clustering algorithms. In Section 4.3.1, we review fuzzy c-means clustering, which is the basis of our work. In Section 4.3.2, we introduce the shape-based distance (SBD) [170]. Section 4.3.3 introduces a new method for assigning a center to clusters based on the shape of the time-series and briefly explains the k-Shape clustering method [170].

4.3.1 The Generalized Fuzzy c-Means Model

Crisp and fuzzy/probabilistic c-partitions of n objects are, respectively, the sets of matrices defined as

$$M_{fcn} = \{U \in \Re^{cn} : for\, 1 \le i \le c; 1 \le k \le n; 0 \le u_{ik} \le 1; \sum_{i=1}^{c} u_{ik} = 1 \ \forall k; \sum_{k=1}^{n} u_{ik} > 0 \ \forall i\} \tag{4.1a}$$

$$M_{hcn} = \{U \in M_{fcn} : u_{ik} \in \{0,1\} \forall i,k\}. \tag{4.1b}$$

The *Generalized Least Squares Error* (GLSE) clustering model is the constrained optimization problem

$$min_{(U,P)}\{J_m(U,P;X) = \sum_{k=1}^{n}\sum_{i=1}^{c}(u_{ik})^m[\Delta(x_k,p_i)]^2\}. \tag{4.2}$$

In equation (4.2), m is a weighting parameter, $m \ge 1$, $U \in M_{fcn}$, $P = \{p_1, \ldots, p_c\}$ is a set of cluster prototypes, and $[\Delta(x_k, p_i)]^2$ is a measure of the squared error incurred by representing input x_k by prototype p_i. Optimal partitions U^* of X are taken from pairs (U^*, V^*) that are local minimizers of J_m.

There are dozens of instances of equation (4.2) for specific choices of P and Δ. For example, the prototypes P may be points, lines, planes, linear varieties [28], hyperquadrics [132], or even regression functions [89]. The dissimilarity measure Δ may be an inner

product norm [28], a Minkowski norm [30], or a shape-based distance [170] such as the one used in this chapter.

For many choices of (P, Δ), there is a theory to guide approximate solutions of the model in equation (4.2) by an *alternating optimization* (AO) algorithm based on Picard iteration through necessary conditions for its local extrema. The most familiar case is the original fuzzy c-means (FCM) model with accompanying FCM/AO algorithm, first reported in [26] and described at length in [28]. This case is covered by:

Theorem (1) (FCM) [26]: Let $X = \{x_1, \ldots, x_n\} \subset \Re^p$ contain at least c distinct points. Let $P = V = \{v_1, \ldots, v_c\} \subset \Re^p$, and A be a positive-definite norm inducing $p \times p$ weight matrix, $\Delta(x_k, p_i)^2 = \|x_k - v_i\|_A^2 = (x_k - v_i)^T A (x_k - v_i) > 0 \quad \forall i, k$. If $m > 1$, then $(U, V) \in M_{fcn} \times \Re^p$ may minimize $J_m(U, V; X)$ only if

$$u_{ik} = \left(\sum_{j=1}^{c} \left(\frac{\|x_k - v_i\|_A}{\|x_k - v_j\|_A} \right)^{\frac{2}{m-1}} \right)^{-1} ; 1 \le i \le c; 1 \le k \le n, \tag{4.3a}$$

$$v_i = \sum_{k=1}^{n} u_{ik}^m x_k / \sum_{k=1}^{n} u_{ik}^m; \; 1 \le i \le c. \tag{4.3b}$$

It is also well known that when $m = 1$ the partition matrix U is necessarily crisp, $U \in M_{hcn}$, and under the remaining hypotheses in Theorem (1), that equation (4.2) reduces to the classical hard c-means (HCM, or classical k-means) model which was first discussed by Lloyd in [141]. In this case, there are several ways to show that conditions (4.3) reduce to the following well known necessary conditions for minimizing J_1.

Theorem (2) (HCM) [26] : Let $X = \{x_1, \ldots, x_n\} \subset \Re^p$ contain at least c distinct points, $P = V = \{v_1, \ldots, v_c\} \subset \Re^{cp}$, and $m = 1$. If $\Delta(x_k, p_i)^2 = \|x_k - v_i\|_A^2 > 0 \quad \forall i, k$, then $(U, V) \in M_{fcn} \times \Re^{cp}$ may minimize $J_1(U, V; X)$ only if $U \in M_{hcn}$ is a *hard c-partition* of X, and

$$u_{ik} = \begin{Bmatrix} 1 & \|x_k - v_i\|_A < \|x_k - v_j\|_A \forall j \ne i \\ 0 & \text{otherwise} \end{Bmatrix} \tag{4.4a}$$

$$v_i = \sum_{k=1}^{n} u_{ik} x_k / \sum_{k=1}^{n} u_{ik}; \; 1 \le i \le c \tag{4.4b}$$

The content of Theorem (2) is often given by denoting the partition $U \in M_{hcn}$ by its equivalent set-theoretic form,

$X = \bigcup_{i=1}^{c} X_i; \oslash = X_i \bigcap_{i \neq j} X_j; \sum_{i=1}^{c} \mid X_i \mid = \sum_{i=1}^{c} n_i = n$. Using this representation, equations (4.4) take the alternate form

$$u_{ik} = \begin{Bmatrix} 1 & x_k \in X_i \\ 0 & \text{otherwise} \end{Bmatrix}; 1 \leq i \leq c,\ 1 \leq k \leq n \tag{4.5a}$$

$$v_i = \bar{v}_i = \sum_{x_k \in X_i} \left(\frac{x_k}{n_i} \right); 1 \leq i \leq c. \tag{4.5b}$$

The advantage of using equations (4.4) instead of (4.5) is that the role of the partition matrix U is clearly seen in the more general fuzzy case. Note that (4.5a) labels each point in X with the nearest prototype rule; and (4.5b) shows that the point prototypes are none other than the geometric centroids (sample means) of the c clusters in X.

There are various ways to estimate solutions for the GLSE problem (4.2) for the choice $[\Delta(x_k, p_i)]^2 = \|x_k - v_i\|_A^2$. The most popular method is Picard iteration through necessary conditions (4.3) or (4.4). For the fuzzy case, these are *first order necessary conditions* (FONCs, the gradient must vanish at extreme points). For the crisp case, looping through conditions (4.4) or (4.5) is sometimes called Lloyd iteration: (4.4b), (4.5b) is a FONC, and (4.4a), (4.5a) is necessary, but not first order. This method is summarized as Algorithm 1.

Algorithm 1 AO c-means for HCM and FCM

Input: $X = \{x_1, \ldots, x_n\} \subset \Re^p$
Output: (U, V)
1: Pick $1 < c < n : m \geq 1\ T_M = iterationLimit$
2: Model Norm: $\| x - v \|_A = \sqrt{(x-v)^T A (x-v)}$
3: $E_t = \mid J_m(U_t, V_t) - J_m(U_{t-1}, V_{t-1}) \mid$
4: $0 < \varepsilon = terminationCriteria$
5: Guess $V_0 = \{v_{10}, \ldots, v_{c0}\} \subset \Re^{cp}$
6: Calculate U_0 with V_0 and (3a) or (4a)
7: $t = 1$: $E_1 = bigNumber$
8: **while** $t < T_m$ *and* $E_t > \varepsilon$ **do**
9: Calculate U_t with V_{t-1} and (3a) or (4a)
10: Calculate V_t with (3b) or (4b)
11: $t = t + 1$
12: $(U, V) \leftarrow (U_t, V_t)$

Reference [28] contains details about most of the issues that arise concerning the use of these two clustering models and Algorithm 1. Our interest is confined to three algo-

rithms for clustering finite time-series in the special case when each $x_k \in X$ is a vector of length p that represents a finite portion of a waveform or time-series. All three algorithms are ad hoc methods that depend on equations (4.4) or (4.5) and use Algorithm 1 with appropriate modifications to find approximate solutions.

4.3.2 Shape Based Distance

The quality of clustering in feature vector data almost always depends on finding a good way to measure similarity or distance between the items represented by the data. Waveform data presents some special challenges, when represented by fixed length vectors of finite time-series. Most of the existing literature on clustering in time-series data modifies a classic clustering algorithm such as k-means by *(i)* substituting a suitable distance measure on the raw data; or *(ii)* by extracting features from the data that convert it into vectors that can be inserted directly into a classic algorithm. This chapter follows the approach taken in [170], which concentrates attention on SBD as the distance of choice for waveform clustering.

The underlying desire in waveform clustering is to use a distance measure that capture shapes differences that are invariant to phase and amplitude changes in the data. Constrained dynamic time warping is often recommended for this job [189]. Forty six (46) such measures for time-series data are discussed in [77]! This gives some idea of how long and hard the path towards a good waveform distance measure has been. The shape-based distance (SBD) measure introduced in [170] for time-series data forms the basis for our two new algorithms.

The new algorithms begin by modifying the GLSE model at (4.2) by introducing the SBD, which is tailored to the time-series case to define $\Delta(x_k, p_i)$. Specifically, this distance is introduced for a pair of z-normalized vectors x and y of length p in Algorithm 2:

Notes about Algorithm 2 The z-normalization mentioned in the input line is the standard $(0,1)$ statistical normalization, i.e., the n input vectors are linearly transformed by subtracting their feature means and dividing by their standard deviations for each of the p features in the input data. Lines 2, 5, and 8 correspond to the following equations from [170].

(A) $CC(X,Y) = \mathcal{F}^{-1}\{\mathcal{F}(x)^*\mathcal{F}(y)\}$, where $\mathcal{F}$ and $\mathcal{F}^{-1}$ are forward (inverse) discrete

Algorithm 2 [170]: $[dist, y'] = SBD(x, y)$

Input: Two z-normalized sequences $x, y \in \Re^p$
Output: Dissimilarity dist = SBD(x,y) $\in \Re^+$;Aligned sequence y' of y towards x
1: length = $2^{nextpower2(2*length(x)-1)}$
2: $CC = IFFT\{FFT(x, length)^* FFT(y, length)\}$%(A)
3: $NCC_c = CC / \| x \| \bullet \| y \|$
4: [value,index] = max(NCC_c)
5: dist= 1 - value % (B)
6: shift = index - length(x)
7: **if** $shift \geq 0$ **then**
8: y'= [zeros(1,shift), y(1:end-shift)] %(C)
9: **else**
10: y'= [y(1-shift:end), zeros(1,-shift)] %(C)

Fourier transforms, $*$ is convolution.

(B) $dist = SBD(x, y) = 1 - max_w\{CC_w(x, y)/\|x\| \bullet \|y\|\}$, where $w \in \{1, 2, \ldots, 2p - 1\}$; $CC_w(x, y) = R_{w-p}(x, y)$; and

$$R_k(x, y) = \left\{ \begin{array}{ll} \sum_{j=1}^{p-k} x_{j+k} y_j; & k \geq 0 \\ R_{-k}(x, y); & k < 0 \end{array} \right\}.$$

$$(C)\ x_{(s)} = \left\{ \begin{array}{ll} (\overbrace{0, \ldots, 0}^{|s|}, x_1, \ldots, x_{p-s}); & s \geq 0 \\ (x_{1-s}, \ldots, x_p, \underbrace{0, \ldots, 0}_{|s|}); & s < 0 \end{array} \right\}.$$

Equation (C) shows how a sliding window passes across a vector, while looking for the optimal alignment between x and y. The inputs to Algorithm 2 are vectors in $\Re^p$, so the distance $\|x_k - v_i\|_A$ in equations (4.4a) and (4.5a) can be directly replaced by $SBD(x, y)$, which is called a distance in [170]. Since $SBD(x, y)$ is produced by Algorithm 2, it is not immediately obvious whether the function $\Delta = SBD$ is really a metric on $\Re^p$. Moreover, it is not clear whether the distance emerging from Algorithm 2 should be regarded as a squared distance or not, although the authors of [170] state that it replaces $[\Delta(x, v)]^2$ in the GLSE problem. The authors of [170] make this substitution in (4.4a) and in line 3 of Algorithm 1 with $m = 1$, and call the resultant algorithm the $k - AVG + ED$ algorithm, where AVG means (4.4b) or (4.5b) is used to compute the prototype updates, and ED stands for Euclidean distance.

Paparrizos and Gravano argue in [170] that most of the research on waveform clus-

tering has used distances like DTW to the exclusion of *cross correlation* (CC) between time-series because this time honored statistical method suffers from normalization and registration issues. They assert that to be useful, the data and the CC applied to it must be appropriately normalized. The z-normalization of the inputs to Algorithm 2 gives it scale invariance, and takes care of inherent distortion in the data. They tackle the normalization of CC with equations (A)-(C) that follow Algorithm 2. Shift invariance is addressed by computing $CC_w(x,y) = R_{w-p}(x,y)$, which maximizes CC at the best match between x and y by testing each position offered by the sliding window in equation (C). They discuss three ways to normalize CC, but only use CC_w, shown and used here, which produces values in the closed interval $[-1,1]$. Consequently, $SBD(x,y) \in [0,2]$, and it takes the value 0 when x and y are perfectly similar (corresponding to zero distance, i.e., $x = y$). Efficiency of finding CC_w is addressed by using the discrete forward and inverse Fourier transforms and padding the input data so its length is always a power of 2. The overall complexity of Algorithm 2 is given as $O(plog(p))$.

4.3.3 Shape Based Prototypes and k-Shape Clustering

The second major alteration of k-means introduced in [170] is to compute prototypes that attempt to capture shape information, based on the fact that input vectors represent finite time-series waveforms. To begin, we revisit equation (4.3a) for v_i. This prototype arises for FCM by zeroing the gradient of the function $J_m(U^*, V; X)$ in the reduced optimization problem

$$min_{V \in \Re^{cp}} \left\{ J_m(U^*, V; X) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik}^*)^m \|x_k - v_i\|_A^2 \right\}. \tag{4.6}$$

U^* is fixed in (4.6), and minimization of $J_m(U^*, V; X)$ is unconstrained, so solving $\nabla_{v_i} J_m(U^*, V; X) = 0$ for v_i leads directly to (4.3a) in the fuzzy case, and (4.4a) in the crisp case. When the distance in equation (4.6) is not an inner product norm, $J_m(U^*, V; X)$ is generally not differentiable with respect to v_i. In this more general case, the reduced problem for the crisp case ($m = 1$ in (4.6)) of the GLSE model becomes:

$$min_{v \in \Re^p} \left\{ J_1(v; X) = \sum_{x_k \in X_i} \Delta(x_k - v)^2 \right\}. \tag{4.7}$$

The method of solving (4.7), sometimes referred to as the Steiner sequence problem [173], depends on the nature of the distance function $\Delta(x_k - v)$. When alignment of the observations $\{x_k\}$ with prototype v is required, this becomes the multiple sequence alignment problem, known to be NP-complete [213]. Bypassing some intermediate steps given in [170], (4.7) eventually becomes an instance of maximization of the famous Rayleigh Quotient [206], viz,

$$v_i = max_{v \in \Re^p} \left\{ \frac{v^T Q^T S Q V}{v^T v} \right\} = max_{v \in \Re^p} \left\{ \frac{v^T M v}{v^T v} \right\} \tag{4.8}$$

where $M = Q^T S Q$, $Q = I - (1/p)[\mathbf{1}]$, and $[\mathbf{1}]$ is the $p \times p$ matrix of $1's$. The solution of (4.8) is the eigenvector corresponding to the largest eigenvalue of the real symmetric matrix M, which is computed in the last line of Algorithm 3, which records this procedure as shown in [170].

Algorithm 3 [170]: $V_{i,t+1}$= **Shape Extraction**$(X, v_{i,t})$

Input: $X = \{x_1, \ldots, x_n\} \subset \Re^p$ as a $n \times p$ matrix X whose columns are z-normalized time-series vectors. $v_{i,t} \in \Re^p$ is the reference sequence against which time-series of X are aligned

Output: $v_{i,t} \in \Re^p$ % new i^{th} shape prototype

1: $X' \leftarrow [\ \ ]$
2: **for** $i = 1$ **to** n **do**
3: [dist,x'] $\leftarrow$ SBD$(v_{i,t}, X(i))$ %Algorithm 2
4: $X' \leftarrow [X'; x']$
5: **end for**
6: $S \leftarrow X'^T \bullet X'$ %S, Eq (8)
7: $Q \leftarrow I - \frac{1}{p}[\mathbf{1}]$ %Q, Eq (8)
8: $M \leftarrow Q^T \bullet S \bullet Q$ %M, Eq (8)
9: $v_{i,t} \leftarrow Eig(M, 1)$ %Extract 1^{st}ev

Notes about Algorithm 3. Lines 6, 7 and 8 compute the quantities S, Q and M needed in equation (4.8) to realize line 9, which produces updated prototype $v_{i,t+1}$ for cluster i given the points (X_i) currently in it and the current prototype $v_{i,t}$. Since there are c prototypes, the k-Shapes (of course $k = c$ here) algorithm in [170] returns to Algorithm 3, c times during the prototype refinement step of the k-Shape clustering iteration.

Now everything is in place for Papparrizos and Gravano to define their k-Shape method, repeated here from [170].

Algorithm 4 [170]: $[u, V]$= **k-Shape**(X, c)

Input: $X = \{x_1, \ldots, x_n\} \subset \Re^p$ as a $n \times p$ matrix X whose columns are z-normalized time-series vectors: c is the number of clusters produced

Output: **u** is a $n \times 1$ label vector that partitions X into c crisp clusters. V is a $c \times p$ matrix containing c shape extracted centroids of length p.

$iter \leftarrow 0$
$u' \leftarrow [\quad]$
while $u! = u'$ and $iter < 100$ **do**
  $u' \leftarrow u$
  % *Refinement step*
    for $j \leftarrow 1$ to c **do**
      $X' \leftarrow [\quad]$
      for $i \leftarrow 1$ to n **do**
        if u(i)= j **then**
          $X' \leftarrow [X', X(i)]$
        end if
      end for
    $v(j) \leftarrow$ Shape Extraction $(X'; v(j))$ %Algorithm3
    end for
  % *Assignment step*
    for $i \leftarrow 1$ to n **do**
      $mindist \leftarrow \infty$
      for $j \leftarrow 1$ to c **do**
        $[dist, x'] \leftarrow$ SBD $(v(j), X(i))$ %Algorithm2
        if $dist < mindist$ **then**
          $mindist \leftarrow dist$
          $u(i) \leftarrow j$
      end if
      end for
    end for
  $iter \leftarrow iter + 1$

4.4 FCS+ and FCS++ Clustering

The SBD algorithm, in conjunction with Theorem (1), offers a way to immediately modify the basic FCM algorithm for waveform inputs. We can replace the inner product norm in Line 2 of Algorithm 1 with the SBD computed by Algorithm 2. This results in our first heuristic generalization, which we will call *fuzzy c-Shape plus* (FCS+). We have added the "+" to the acronym FCS so that you won't confuse this new shape-based distance algorithm with an older one bearing the acronym FCS (*fuzzy c-shells*, [54]), in which the prototypes are shell boundaries and the model norm is an inner product A-norm.

Algorithm 5 FCS+

1: **Replace**: line 3 in Algorithm FCM with: 3 Model Norm: SBD$(x, v) \leftarrow \| x - v \|_A^2$
2: **Do**: Algorithm 1
3: $[optional]$ Harden U, line 13, Algorithm 1 with Eq(11)
4: $[optional]$ Output: $H_{mm}(U) \in M_{hcn}$

Note about Algorithm 5. The optional lines of Algorithm 5 harden the terminal fuzzy partition. This option is NOT necessary, since there are soft versions of all four CVIs based on the contingency matrix discussed in the next section [13]. However, we will use this option for FCS+ in our numerical experiments so that the comparison of FCS+ to k-Shape (which has only crisp partitioning) is equitable. Algorithm 4 represents the crisp partition of X that it produces as the $n \times 1$ label vector $\mathbf{u}$ (not bold in Algorithm 4). This is an efficient way to carry the information in the crisp case. To see how a fuzzy generalization would make sense, let us represent the partition information possessed by the crisp vector $\mathbf{u}$ as a matrix $U \in M_{hcn}$. For example, suppose the output of Algorithm 4 is $\mathbf{u} = [12131]$, so there are $c = 3$ crisp labels for $n = 5$ objects. The matrix representation for this $\mathbf{u}$ is

$$U = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.9}$$

During the refinement step, Algorithm 4 will update the three cluster centers by calling Algorithm 3 three times. Each prototype update uses only the data vectors in its cluster, illustrated graphically as follows:

$$U = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad \begin{array}{ccccc} \Longrightarrow & v_{1,t} & \rightarrow & v_{1,t+1} \\ \Longrightarrow & v_{2,t} & \rightarrow & v_{2,t+1} \\ \Longrightarrow & v_{3,t} & \rightarrow & v_{3,t+1} \end{array} \tag{4.10}$$

Equation (4.10) shows how *crisp* membership in each cluster is used to control which vectors in the data set are accessed during the update procedure. At each j, lines 6-9 of Algorithm 4 picks out only the vectors in current cluster X_j (corresponding to the 1's in the j^{th} row of U) and writes them into $X^{'}$, so when Algorithm 3 is called in line 12 of Algorithm 4, only the points in X_j are sent to it via array $X^{'}$ to update the shape based prototype for the j^{th} cluster.

The representation of the partition produced by Algorithm 4 at (4.10) shows how to generalize the k-Shape algorithm to the fuzzy case. Algorithm FCS+ produces a fuzzy $U \in M_{fcn}$, which we can harden using the maximum membership function $\mathbf{H}_{mm} : M_{fcn} \rightarrow M_{hcn}$, which operates on the columns of U. $\mathbf{H}_{mm}(U) = [\mathbf{h}(\mathbf{U}^{(1)}) \cdots \mathbf{h}(\mathbf{U}^{(n)})]$ is a hardening of U defined on the n columns $\{\mathbf{U}^{(k)}\}$ of U as follows:

$$\begin{aligned} \mathbf{h}(\mathbf{U}^{(k)}) &= (0, 0, \ldots, \underbrace{1}_{i^{th}}, \ldots)^T \\ &\Longleftrightarrow \begin{Bmatrix} 1 & u_{ik} > u_{jk}; j \neq i \\ 0 & \text{otherwise} \end{Bmatrix}. \end{aligned} \tag{4.11}$$

In words: $\mathbf{h}$ replaces the largest value in each column of U with a 1, and place 0's in the other $c-1$ slots in each column of U. When ties occur, assign the membership 1 arbitrarily to any winner, and treat the other maximums as non-winners. For example if we apply H_{mm} to the matrix

$$U' = \begin{bmatrix} 0.9 & 0.1 & 0.6 & 0.3 & 0.65 \\ 0.1 & 0.8 & 0.3 & 0.22 & 0 \\ 0 & 0.1 & 0.1 & 0.75 & 0.35 \end{bmatrix} \tag{4.12}$$

the result is that $\mathbf{H}_{mm}(U^{'}) = U$, the matrix at (4.10).

Once this conversion is made, it is a simple matter to convert $\mathbf{H}_{mm}(U^{'})$ back to the list form row by row, $\mathbf{u}\leftarrow\mathbf{H}_{mm}(U^{'})$, required as one of the inputs to Algorithm 4. With this conversion in hand, we can define the *fuzzy c-Shape* double plus (FCS++) algorithm, where the first + stands for the use of the SBD distance function (Algorithm 2), and the second + represents the use of SE prototypes via Algorithms 3 and 4.

Algorithm 6 FCS++

Input: $X = \{x_1, \ldots, x_n\} \subset \Re^p$
Output: (U, V)
Pick $1 < c < n : m \geq 1 : T_M$= iteration limit
Model Norm: $\Delta(x, v) = \text{SBD}(x, v)$
Error Norm: $E_t = \parallel V_t - V_{t-1} \parallel$
$0 < \varepsilon =$ termination criterion
Guess $V_0 = \{v_{10}, \ldots, v_{c0}\} \subset \Re^{cp}$
$t = 1 : E_1 =$ big number
while $T < T_M$ and $E_t > \varepsilon$ **do**
  Calculate U_t with V_{t-1} and (3a)
  $\mathbf{u}\leftarrow \mathbf{H}(u_t)$ %Harden U_t with Eq(11)
  %SE Refinement step
  for $j \leftarrow 1$ to c **do**
    $X' \leftarrow [\ \ ]$
    for $i \leftarrow 1$ to n **do**
      if $u(i) = j$ **then**
        $X' \leftarrow [X', X(i)]$
      end if
      $v(j) \leftarrow$ Shape Extraction $(X'; v(j))$ %Algorithm3
    end for
  end for
    $V_t = [V(1)V(2) \cdots v(c)]$
    $t = t + 1$
$(U, V) \leftarrow (u_t, V_t)$ % $U \in M_{hcn}$ is crisp

4.5 Cluster Validity Indices

The quality of our experimental outputs can be judged in a number of ways. To make direct comparisons between the k-Shape outputs in [170] to those found by FCS+ and FCS++, we will follow [170] by using external *cluster validity indices* (CVIs), which are functions that identify a "best" member amongst a set of *candidate partitions* $CP = \{U \in M_{fcn} : c_m \leq c \leq c_M\}$ of any set of $O = \{o_1, \ldots, o_n\}$ of n objects.

There are two basic types of CVIs. *Internal* CVIs use only the information available from the algorithmic outputs to assess the quality of each $U \in CP$. *External* CVIs use the information available to internal indices, but also use "outside" information about the data, which almost always means that the data are labeled by a crisp ground truth partition. So, external CVIs basically compare partitions in CP obtained by a clustering algorithm to the crisp partition of ground truth labels. This is rightly regarded as "fake clustering," since the subsets in the labeled data are, presumably, already clustered (but please note that these are labeled subsets, which may or may not be regarded as clusters by a specific model and algorithm). But this is a good way to compare different clustering algorithms (which is our aim here), and so many studies of this kind exist in the literature.

Chapter 16 in [117] is an excellent source of general information about CVIs. The seminal paper by Milligan and Cooper [149] was the first comprehensive study of the selection of an internal CVI. The use of external CVIs to choose a "good" internal CVI as discussed in [13, 137]. Many other internal CVIs that are popular for validation of crisp partitions are compared in [14], which offers a wide choice of potential CVIs that might be adapted to the present application. Here we briefly summarize the four external indices we use to evaluate our algorithms.

Let $U \in M_{hrn}$, $Q \in M_{hcn}$ and $N = UQ^T$ (in general, $r \neq c$). When a reference (ground truth) partition is available, it will be Q in the transformation $N = UQ^T$. The matrix N forms a contingency table between the two partitions, so is sometimes called a *contingency matrix*. Given Q, we match Q to candidate $U \in CP$ using $CVI(U \mid Q)$. The Rand index [182], one of the first (and still most popular) crisp external CVIs, is based on (4.4) paired comparison values derived from the elements of N :

$$a = \frac{1}{2}\sum_{i=1}^{r}\sum_{j=1}^{c} n_{ij}(n_{ij}-1); \tag{4.13a}$$

$$b = \frac{1}{2}\left(\sum_{j=1}^{c} n_{\bullet j}^2 - \sum_{i=1}^{r}\sum_{j=1}^{c} n_{ij}^2\right); \tag{4.13b}$$

$$c = \frac{1}{2}\left(\sum_{i=1}^{r} n_{i\bullet}^2 - \sum_{i=1}^{r}\sum_{j=1}^{c} n_{ij}^2\right); \tag{4.13c}$$

$$d = \frac{1}{2}\left(n^2 + \sum_{i=1}^{r}\sum_{j=1}^{c} n_{ij}^2 - \left(\sum_{i=1}^{r} n_{i\bullet}^2 + \sum_{j=1}^{c} n_{\bullet j}^2\right)\right). \tag{4.13d}$$

In equation (4.13) $n_{i\bullet}$, $n_{\bullet j}$ are, respectively, the i^{th} row and j^{th} column sums of the contingency matrix N.

CVIs are notoriously fickle, so we use four CVIs here based on different combinations of the values in the contingency matrices to see if they will all rank the three algorithms the same way. The *Rand Index* (RI) is computed with these four values as

$$RI(U \mid Q) = \frac{(a+d)}{(a+d)+(b+c)} \tag{4.14}$$

The numerator $(a+d)$ is the number of *agreements* between pairs in U and Q; $(b+c)$ is the number of *disagreements*. The RI is valued in $[0,1]$, taking its maximum if and only if $U = Q$. So, the heuristic for this index is that the maximum value of the RI over the partitions in CP points to the best match among the candidates. We call such an index a $max-optimal \quad CVI$, indicated as $(\uparrow)$. Several adjustments of (4.14) have been proposed in the literature that attempt to rectify the tendency of the RI to increase monotonically with c. Of these, the *Adjusted Rand Index* (ARI) of Hubert and Arabie [112] is the most popular:

$$ARI(U \mid Q) = \frac{\left(a - \frac{(a+c)(a+b)}{a+b+c+d}\right)}{\left(\frac{(a+c)+(a+b)}{2} - \frac{(a+c)(a+b)}{a+b+c+d}\right)}. \tag{4.15}$$

The ARI is also max-optimal $(\uparrow)$: it maximizes at 1, but its minimum may be negative if the index is less than its expected value of zero. The third external CVI we will use is the *variation of information* (VI) introduced in [146]:

$$\begin{aligned} VI(U \mid Q) = &- \sum_{i=1}^{r}\sum_{j=1}^{c} \frac{n_{ij}}{n} log \frac{n_{ij}}{n} \\ &- \left[\sum_{i=1}^{r}\sum_{j=1}^{c} \frac{n_{ij}}{n} log \frac{n_{ij}/n}{(n_{i\bullet}/n)(n_{\bullet j}/n)}\right]. \end{aligned} \tag{4.16}$$

The VI index is a metric valued in $[0, log(n)]$ which takes the value 0 when $U = Q$, so the heuristic for VI is to accept the partition achieving the minimum value over CP: the VI is min-optimal $(\downarrow)$.

The last external CVI we use is a form of mutual information, which is normalized by a

maximum calculation, so this index bears the notation $NMI_M(U \mid Q)$:

$$NMI_M(U \mid Q) = \frac{\sum_{i=1}^{r}\sum_{j=1}^{c}\frac{n_{ij}}{n}log\frac{n_{ij}/n}{(n_{i\bullet}/n)(n_{\bullet j}/n)}}{max\{H_S(U), H_S(Q)\}}, \tag{4.17}$$

where, for example, $n_i = \sum_{k=1}^{n} u_{ik}$ is the number of points in the i^{th} cluster in U, and $H_S(U) = -\sum_{k=1}^{r}\sum_{i=1}^{c}\frac{n_i}{n}log\frac{n_i}{n}$ is Shannon's entropy of U [117], and likewise for $H_S(Q)$. This index has been a good performer in several cluster validity studies [211], and provides a nice contrast to the other three indices. The range of $NMI_M(U \mid Q)$ is $[0, 1]$, and it is a max-optimal $(\uparrow)$ CVI, so larger values point to better matches between U and Q.

The four CVIs we use can all be generalized to compare soft U's to crisp Q by the method discussed in [13]. However, we will convert FCS+ outputs to the crisp partition $H_{mm}(U)$ per Equation (4.11) before computing these indices. And we MUST convert U to $H_{mm}(U)$ to implement FCS++. Consequently, all three clustering algorithms will be evaluated using the same type of information.

4.6 Numerical Experiments

We use the same 48 synthetic and real data sets that were used in [170]. The 48 sets are all z-normalized, crisply labeled, split into training and test sets, and are collected in the UCR time-series database [126]. Our experiments are based on merged versions of the training and test sets. The length of the time-series (number of points sampled from the waveforms) in each set is fixed and equal, ranging from $p = 24$ to $p = 1882$. The number of sample waveforms in the data sets ranges from $n = 56$ to $n = 9236$. Since all 48 of these data sets are class-labeled, we do not use the external CVIs to select a best candidate from a set of c-partitions at different values of c. Here we simply run FCS+ and FCS++ using the specified number of target labels for each of the 48 data sets, acknowledging that the number of labeled classes does not automatically identify the number of clusters that any model and algorithm might think are present in the data. We ran the experiments for each clustering method ten times for each data set. We implemented the clustering methods in MATLAB R2014b (64bits) using the platform: Intel(R) with core(TM)i7 processor and

clock speed at 3.60 GHz and 16 GB RAM.

Table 4.1 shows the grand averages over 480 trials for each of the four crisp external CVIs. Each index was evaluated 10 times for different runs of FCS+ and FCS++ on each of the 48 labeled data sets. The ordering for the three algorithms is the same for all four indices. Table I shows that FCS++ is slightly better than FCS+, and in turn, FCS+ outperforms k-Shape by a small margin for all four indices. This demonstrates that the overall quality of both of the new fuzzy methods for clustering waveform data is superior to the k-Shape algorithm in [170]. Figure 4.1 compares each pair of methods graphically.

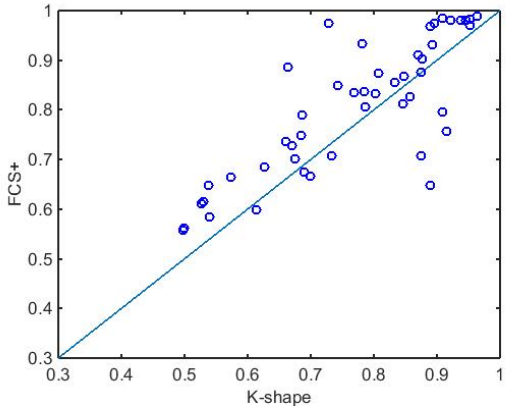
Table 4.1: Grand Average of (480) CVI values
(10 runs per data set x 48 data sets)

CVI	Type	Range	FCS++	FCS+	k-Shape
$\overline{\overline{RI}}$	$(\uparrow)$	$[0,1]$	**0.822**	0.807	0.772
$\overline{\overline{ARI}}$	$(\uparrow)$	$[-a,1]$	**0.461**	0.403	0.321
$\overline{\overline{NMI_M}}$	$(\uparrow)$	$[0,1]$	**0.641**	0.534	0.413
$\overline{\overline{VI}}$	$(\downarrow)$	$[0,logn]$	**1.010**	1.463	2.455

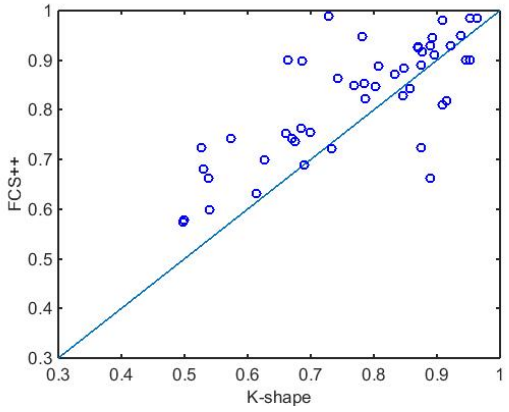
The line through the origin at 45 degrees in each view separates $[0,1] \times [0,1]$ into two half-spaces. Each of the views 4.1a, 4.1b and 4.1c plots the average values of the Rand Index over 10 runs for each of the 48 data sets. The dots represent the 48 data sets, and *the coordinates of each dot are the average values of the Rand index for the algorithm pair in each view*.

For example, the coordinates of points in Figure 4.1a are: horizontal coordinate x = average RI value achieved by 10 runs of k-Shape; vertical coordinate y = average RI value achieved by 10 runs of FCS+. There are 10 points below the line y = x in Figure 4.1a, 1 point on the line, and 37 points above the line. This means that FCS+ achieved a better average result with the RI than k-Shape on 37 of the 48 data sets, they were tied on one data set, and k-Shape had a better average RI than FCS+ on 10 data sets. Figure 4.1b shows that FCS++ is slightly better, with 38 data sets above the line. And Figure 4.1c shows that FCS++ achieves a better result than FCS+ on 38 of the 48 data sets.

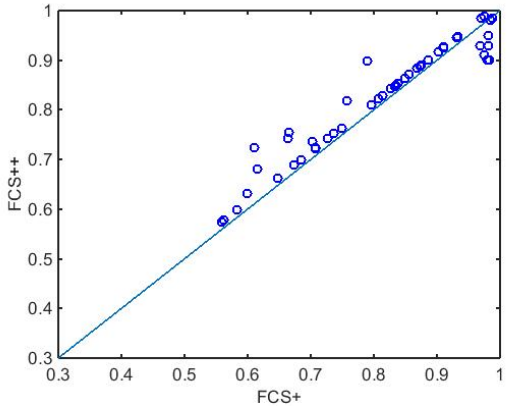
Time complexity analysis: Assume that n and p are the number and the length of the finite time-series respectively and c is the number of clusters. All three algorithms use



(a) FCS+ vs. k-Shape. Circles above the diagonal indicate datasets for which FCS+ has a better average Rand Index than k-Shape.



(b) FCS++ vs. k-Shape. Circles above the diagonal indicate datasets for which FCS++ has a better average Rand Index than k-Shape.



(c) FCS++ vs. FCS+. Circles above the diagonal indicate datasets for which FCS++ has better average Rand Index than FCS+.

Figure 4.1: Comparisons of k-Shape, FCS+ and FCS++. Average RI values of ten runs over 48 data sets.

Table 4.2: Average run time for 480 values of the Rand Index (48 data sets, 10 runs each)

Algorithm	Ave. CPU Time (in seconds)
FCS+	9.51
k-Shape	22.95
FCS++	25.04

Table 4.3: Accuracy comparison among three clustering method using Wilcoxon test based on RI validity method. R^+ corresponds to the sum of the ranks for the method on the left and R^- for the right

	RI Accuracy		
Method	R^+	R^-	ρ-value
k-Shape vs FCS++	215	**913**	$1.304e^{-04}$
k-Shape vs FCS+	238	**813**	$3.309e^{-04}$
FCS++ vs FCS+	**931**	245	$2.608e^{-04}$

the SBD function to measure dissimilarity between data points and centroids. The SBD function requires $O(plog(p))$ time to calculate this measurement. The implementation of FCM as shown in Algorithm 1 is $O(npc^2)$. k-Shape uses the k-means algorithm as its underlying clustering algorithm, where the time complexity of k-means is $O(npc)$. Now, the time complexity for the FCS+ algorithm can be calculated as $O(ncplog(p))$ time. In FCS++ and k-Shape clustering algorithms, there is a refinement step, which for every cluster calculates matrix M with $O(p^2)$ time complexity, and then computes an eigenvalue decomposition on M with $O(p^3)$ time complexity. Therefore, the complexity of the refinement step is $O(max\{np^2, cp^3\})$. As a result, the per iteration time complexity of FCS++ and k-Shape are $O(max\{nc^2plog(p), np^2, cp^3\})$ and $O(max\{ncplog(p), np^2, cp^3\})$ time respectively. Thus, all three algorithms are linear in the number of time-series, and the major portion of the computational cost rests with p, the length of the time-series.

Table 4.2 shows the average CPU time taken by each of the three algorithms to evaluate the Rand index for 10 runs of FCM on each of the 48 data sets. FCS+ takes roughly 9.5 s per run, whereas the other two required about three times that, FCS++ topping out at about 25 s per run. This is easy to understand: FCS+ does not use SE Algorithm 3, so it is considerably less expensive computationally than the other two algorithms. Overall, these algorithms are reasonably fast for the 48 data sets used in our experiments.

Statistical analysis: We used two statistical tests that assess the statistical significance

Table 4.4: Accuracy comparison among three clustering method using Wilcoxon test based on ARI validity method. R^+ corresponds to the sum of the ranks for the method on the left and R^- for the right

	ARI Accuracy		
Method	R^+	R^-	ρ-value
k-Shape vs FCS++	66.5	**1061.5**	$1.403e^{-07}$
k-Shape vs FCS+	154	**897**	$6.848e^{-06}$
FCS++ vs FCS+	**1036.5**	139.5	$4.199e^{-06}$

Table 4.5: Accuracy comparison among three clustering method using Wilcoxon test based on NMI_M validity method. R^+ corresponds to the sum of the ranks for the method on the left and R^- for the right

	NMI_M Accuracy		
Method	R^+	R^-	ρ-value
k-Shape vs FCS++	75	**1053**	$1.426e^{-07}$
k-Shape vs FCS+	81	**970**	$1.991e^{-07}$
FCS++ vs FCS+	**1009**	167	$4.199e^{-05}$

of the performance of the various methods. First, we perform pairwise comparisons between different methods using the Wilcoxon signed-rank test [223]. The test returns a ρ-value associated with each comparison, representing the lowest level of significance of a hypothesis that results in a rejection. This value can be used to determine whether two algorithms have significantly different performance and to what extent. For all the comparisons in this study the significance level α is set to 0.05. Tables 4.3, 4.4, and 4.5 summarize these comparisons on all the accuracy results from the three algorithms over 48 datasets. In the three mentioned tables, the ρ-value is less than the significance level ($\alpha = 0.05$), which means that there is a significant difference between the accuracy of algorithms. To illustrate, consider the *RI Accuracy*, where the first row compares k-Shape with FCS++. The sum of ranks for k-Shape is less than the sum of ranks for FCS++. Therefore, the accuracy of FCS++ is better than k-Shape clustering. Another statistical test that enables us to compare the three clustering algorithms is Friedman's test [69]. The Friedman test is a non-parametric statistical test for differences between groups. Figure 4.2 shows the results of applying Friedman's test to the three clustering algorithms with respect to the three max optimal CVIs used in our study. Friedman's test supports

the conclusions drawn from Wilcoxon's test, viz., that FCS++ is superior to FCS+, which is in turn superior to k-Shape.

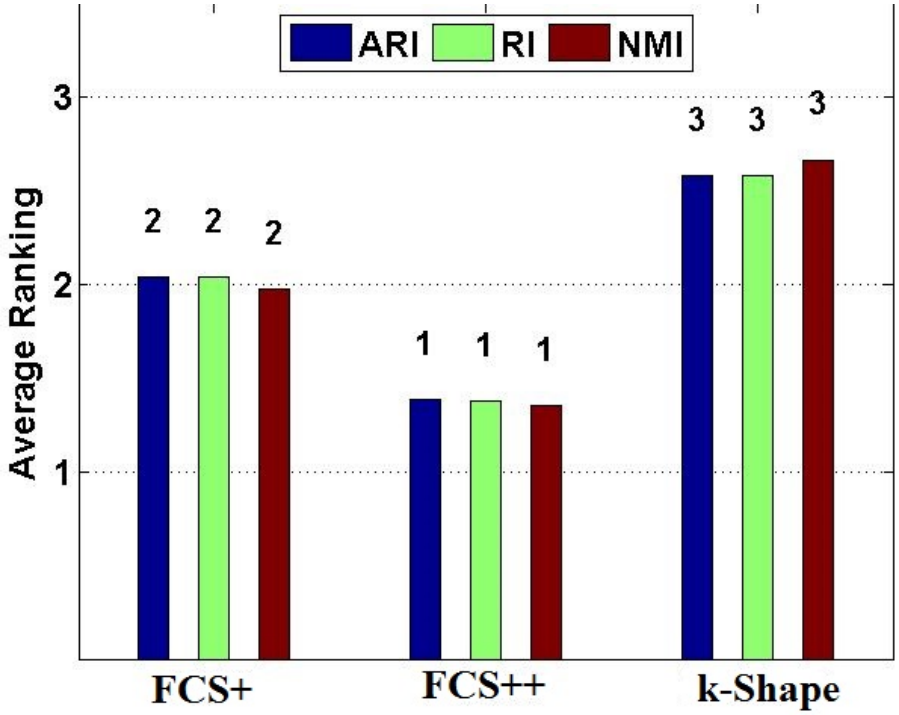


Figure 4.2: Comparison for rankings of clustering accuracy methods for 3 metrics. The bars represent average rankings based on the Friedman test, and the number on the top of the bars indicates the ranking of the algorithm, from the best (1) to the worst (3) for each given measure. The ranking is determined for all datasets and finally an average is calculated as the mean of all rankings.

4.7 Discussion

The authors of [170] assert that k-Shape effectively minimizes the classical HCM objective function $J_1(U, P; X)$ when P is computed with SE Algorithm 3 and the model norm is replaced by $SBD(x, y)$. However, *(i)* no theory is given to support their statement about minimization; *(ii)* Algorithm 2 was not shown to represent an actual metric distance; and *(iii)* it is not clear whether the distances from Algorithm 2 should properly be interpreted as squared.

The convergence properties of FCM and HCM are well known [27], and the basic theory for both depends on iterate sequences having the descent property, i.e., that for

successive iterates, $J_m(U_{t+1}, V_{t+1}) \leq J_m(U_t, V_t)$. The objective functions that might be optimized by these three heuristic algorithms are all related in principle to the LSE function at Equation (4.2), i.e.,

$$J_m(U, P; X) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik})^m [\Delta(x_k, p_i)]^2.$$

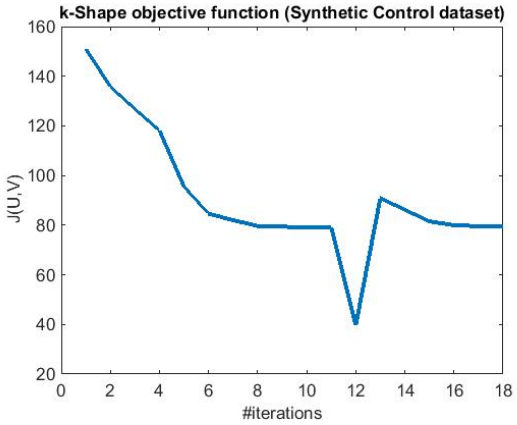
Table 4.6 lists the components of $J_m(U, P; X)$ for each of the three methods tested in this chapter, along with the corresponding information for HCM and FCM. In all cases P is a set of c vectors $V \subset \Re^{cp}$. For FCM, HCM and FCS+ they are unconstrained centroids in p space; for k-Shape and FCS++ the V's are found by shape extraction (Algorithm 3).

Do any of the heuristic methods discussed here generate iterative sequences that endow the objective function obtained by replacing $V \subset \Re^{cp}$ and $\Delta(x_k, p_i)^2$ in Equation (4.2) with the parameters shown in Table 4.6 with the descent property? The answer is no. The last column ($\downarrow$?) of Table 4.6 summarize what is known about the descent property for each of the five methods.

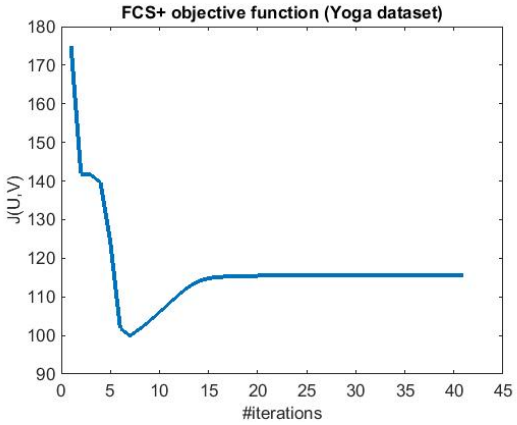
Figure 4.3 plots presumptive objective functions based on the parameters in Table 4.2 for one run of k-Shape, FCS+, and FCS++ on three different data sets. Each view in this figure shows that the objective function for each algorithm is not always monotone decreasing on successive iterates. Consequently, the descent property (and in turn, any convergence theory that requires it) is ruled out by counterexample for all three of these heuristic schemes.

Table 4.6: Presumptive Objective Function Components

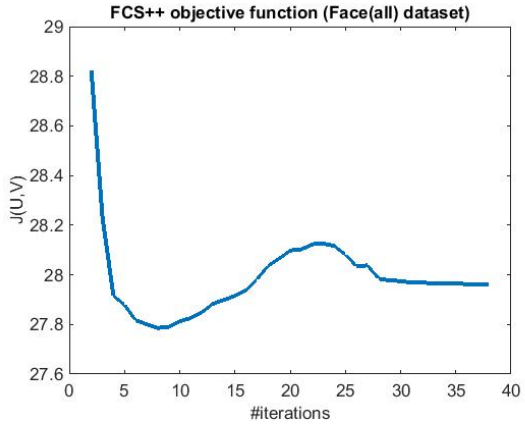
Method	U	$[\Delta(x_k, p_i)]^2$	P	$\downarrow$?
$FCM(m > 1)$	M_{fcn}	$\|x - v\|_A^2$	V	✓
$FCM(m = 1)$ $(c - means)$	M_{hcn}	$\|x - v\|_A^2$	V	✓
$k - Shape$	M_{hcn}	$SBD(x_k, p_i)$	SE V	×
$FCS+$	$H_{mm}(U)$	$SBD(x_k, p_i)$	V	×
$FCS++$	$H_{mm}(U)$	$SBD(x_k, p_i)$	SE V	×



(a) $J(U, V; X)$ for one run of k-Shape: X=Synthetic Control Data



(b) $J(U, V; X)$ for one run of FCS+: X=Yoga Data



(c) $J(U, V; X)$ for one run of FCS++: X=Face (All) Data

Figure 4.3: Objective functions for algorithms k-Shape, FCS+, and FCS++, are not always monotone decreasing on successive iterates.

4.8 Conclusion

Two new fuzzy variants of k-Shape were defined. FCS+ arises by substituting $SBD(x_k, p_i)$ for $\| x - v \|_A^2$ in FCM. FCS++ is FCS+ with the additional substitution of SE V for the prototypes in FCM. For FSC++, partitioning is adapted to the k-Shape algorithm by hardening the fuzzy partitions produced by FCM at each assignment step.

We performed the same experiments with 48 labeled waveform data sets that were done using k-Shape in [170], and compared the two fuzzy methods to k-Shape using four crisp external cluster validity indices and two statistical tests of significance. All four indices indicate that FCS++ performs somewhat better than FCS+, and in turn, FCS+ is slightly superior to k-Shape.

While our numerical results are encouraging, they are by no means definitive. More experiments are needed with other waveform data, including unlabeled data. Another avenue of pursuit is the theory: is SBD a metric? Is there any convergence theory for k-Shape, FCS+ or FCS++? Figure 4.3 suggests that the short answer is "probably not." Moreover, any satisfactory convergence theory has to overcome the fact that the SBD and SE schemes are computer programs, not functions. Finally, we have not capitalized fully on the fuzzy information available at each assignment step of the two new methods, instead opting to harden U so that $H_{mm}(U)$ can be used directly by Algorithm 3. Surely there is a better use of this information? This is the enterprise we will turn to next.

In this chapter we focused on improving the accuracy and efficiency of clustering algorithms for time-series data. In the next chapter we use the proposed algorithm in practice to improve the accuracy of load profiling. By leveraging the new proposed clustering algorithm, we introduce a framework for short-term load forecasting at the grid level, which is able to handle the non-stationary and unbalanced nature of the load data set.

Chapter 5

Robust and Accurate Short-Term Load Forecasting Framework

One of the most critical tasks for operating a power system is load forecasting in order to keep balance between demand and supply and for planning infrastructure. Errors in load forecasting can result in significant cost increases for electricity suppliers and increase the chance of unexpected black-outs or brownouts. Improving the accuracy of short-term load forecasting is a challenging open problem. This chapter proposes a novel framework for short-term load forecasting using the proposed clustering method in the previous chapter, LSTM networks and Xgboost methods. In particular, our proposed approach introduces an ensemble process together with novel features that lead to improved accuracy of the load forecasting model. The performance of the proposed framework is validated with publicly available real-life data from Australian Energy Market Operator, as well as through on-site deployment, which has led to substantially higher accuracy over existing methods.

Due to the integration of renewable energy sources and grid modernization, the power system has become more volatile and less predictable than ever before, as a result applying unsupervised methods for data mining in this area is essential. The proposed framework is able to produce multiple historical models for different temporal variations, and uses ensemble learning techniques to forecast the day ahead load with high accuracy. In addition it is able to handle unbalanced training data sets, which enables high accuracy in predicting the load in the abnormal situations like very hot days and severe weather conditions.

The results of our system have been compared with the existing well-established Australian short-term load forecasting system over the five regions in Australia. It is shown that our predictive framework has a substantially higher accuracy over the existing methods and it is flexible and easy to apply, which makes it an effective model to supplement any operational power system.

5.1 Introduction

SHORT term load forecasting (STLF), which ranges from 30 minutes to one week ahead, is a challenging task of crucial importance for power system operation due to the variety of factors that influence the load and the volume of data that needs to be considered. STLF plays an important role in the reliability, security, market operation, and scheduling of reasonable dispatching plans for smart power grids. With increasing integration of renewable energy sources and the introduction commencement of different demand response programs in the energy market, the load has become more volatile and less predictable than ever before [153, 155, 162, 176]. Thus, trying to produce more accurate load forecasting models has become a major research challenge for energy suppliers, energy marketers, financial markets, and other parties that contribute to electric power generation, distribution and transmission.

Load forecasting has been widely studied since the 1970s and a variety of methods have been proposed. These methods can be categorized into traditional statistical methods and Artificial Intelligence (AI) based methods. Traditional methods introduced linear models for load forecasting, such as regression and time-series analysis [34, 143, 161]. These methods are not suitable for non-linear problems like STLF, thus they often have poor prediction accuracy. AI based methods such as neural networks [92, 129, 160], support vector machines [166], k-nearest neighbors [52], and fuzzy models [201] are able to approximate nonlinear relationships and have been extensively applied to the STLF problem.

Although a wide range of studies have been conducted into load forecasting, producing an accurate STLF is an open research problem due to the variety of factors that influence the load, the volume of data that needs to be considered, and the non-stationary characteristic of the load data.

Recently, Recurrent Neural Network (RNN) models have been recognized as one of the most effective methods in time-series forecasting due to their ability to learn functions of arbitrary complexity and capture the time varying dynamics of the underlying system [191, 224]. In this chapter we use Long Short Term Memory (LSTM) [96], which is a special type of RNN architecture. LSTM can learn the order dependence between items in a sequence and it has the potential to learn the context required to make predictions in time-series forecasting problems, rather than having this context pre-specified and fixed

[74,76]. We use this capability of LSTM to forecast the morning and afternoon peak load for the day-ahead time horizon. We then use these two values as features inputs to a day-ahead forecasting model that generates load forecast for 24 hours ahead at intervals of 30 minutes granularity. We show that these two features have a major influence in improving the day-ahead forecast accuracy. Moreover, in order to produce a robust and accurate forecasting model we have used an unsupervised data mining approach through using c-Shape clustering which we introduced in Chapter 4 and ensemble learning. The proposed framework is able to produce multiple historical models for different temporal variations, and achieves a high accuracy of forecast by applying ensemble learning techniques. In addition, it is able to handle the challenge of unbalanced training data sets, which arise due to the frequency of abnormal situations like extreme weather conditions, which are critical events that require highly accurate forecasts.

There are many challenges associated with load forecasting. Load demand depends on different exogenous factors like temperature, humidity, wind speed, seasonal patterns related to human activities and cyclic information like time of year/season/week/day, and type of day (holiday/working day/ weekend). The selection of exogenous factors and modeling the relationship between the selected variables and load data plays a crucial role in load forecasting. In this work we introduce a novel factor, apparent temperature, which has a major impact on the accuracy of the prediction model.

In summary the novel contributions made in this chapter are:

- We introduce a robust and accurate framework based on unsupervised ensemble learning, which is able to produce accurate predictions even in difficult-to-predict cases that arise out of volatility, spikes, and abnormal situations.
- We introduce a way to handle unbalanced training data sets, which leads to high accuracy of forecasting in abnormal situations like extreme weather conditions.
- We address the research question of whether combining weather variables that reflect the human perception of temperature results in more accurate load forecasts than using traditional weather variables.
- We introduce novel features corresponding to, evening and morning peak demand estimated by an LSTM, as key features in improving the accuracy of load forecast-

ing.

- The performance of the proposed method is validated with real data from the power system in the Australian National Electricity Market as well as through on-site implementation by the system operator. It is shown that our predictive framework has a substantially higher accuracy over existing methods in both normal and abnormal situations.

- Our proposed framework is flexible and easy to apply, which makes it an effective model to supplement any operational power system.

The rest of the chapter is organized as follows: Section 5.2 presents the load forecasting problem in a formal manner and the machine learning benchmark model. In Section 5.3 we explain our approach for day-ahead demand forecasting by introducing a novel clustering based ensemble learning framework and two new input variables for improving forecast accuracy. In Section 5.4 we show our experimental results based on real-life data. Finally, in Section 5.5 we conclude and discuss some possible directions for future work.

5.2 Background

In this section we first formally define the problem, then we introduce a benchmark machine learning model for load forecasting.

5.2.1 Problem Statement

Producing an accurate electricity load forecasting result is achievable by effectively combining appropriate predictive features. We first describe how to systematically model the point load forecast in each time step, then we explain how to choose appropriate input variables for the model.

We are given a historical load data set $L_T = \{L_1, \ldots, L_T\}$, comprising T observations. The goal is to predict the H future observations $\{L_{T+1}, \ldots, L_{T+H}\}$. We can predict the electric load at time $t + h$, where $h \in \{1, 2, \ldots, H\}$, by producing different model at each

forecast horizon h as follows:

$$L_{t+h} = f_h(X_t) + \epsilon_{t+h}, \tag{5.1}$$

where ϵ_{t+h} denotes the model error, $f_h(.)$ is the model which is based on the conditional expectation $E[L_{t+h} \mid X_t]$, and X_t is defined as follows:

$$X_t = (l_t, g_{t+h}), \tag{5.2}$$

where l_t is a vector of lagged demand data occurring prior to time $t+1$, and g_{t+h} is a vector of exogenous variables at time $t+h$ which will be discussed in Section 5.2.2 in detail.

The performance of the model can be evaluated in terms of the Mean Absolute Percentage Error (MAPE):

$$MAPE = \frac{100}{T} \sum_{t=1}^{T} \mid \frac{\hat{L}_t - L_t}{L_t} \mid, \tag{5.3}$$

where $\hat{L}_t$ is the forecast value and L_t is the actual observation at time step t.

5.2.2 Benchmark Model

In this chapter we use the eXtreme Gradient Boosting (XGBoost) [48] algorithm as a benchmark machine learning model. XGBoost has been used widely in many data mining and machine learning challenges and gives state-of-the-art results on a wide range of problems [48]. Moreover, there are a range of typical input variables that have been used in the literature for the electricity load forecasting problem during the last three decades, which we use with this benchmark model.

eXtreme Gradient Boosting: A Scalable Tree Boosting System

XGBoost [48] is an improved implementation of gradient boosted decision trees [67] designed for speed and performance. XGBoost is able to create boosted trees in a well organized way, operates in parallel, and avoid overfitting by using regularization technique. It is called gradient boosting because it uses a gradient descent algorithm to optimize the

value of the objective function. Gradient boosting is an approach where new models are created that predict the residuals or errors of prior models, and then combined to make the final prediction. An important attribute of XGBoost is its scalability in all scenarios, where parallel and distributed computing can make this algorithm run more than ten times faster than existing popular solutions. In addition, It is able to handle sparse data and instance weights in approximate tree learning.

XGBoost is a supervised learning algorithm where the training data x_i (with multiple features) is used to predict the target variable y_i. The prediction model is defined as follows:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F, \tag{5.4}$$

where K is the number of trees, f is a regression tree that maps the attributes to the scores, and F is the set of all possible classification and regression trees (CART). Then, the objective function can be written as:

$$obj(\theta) = \sum_{i}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k), \tag{5.5}$$

where $l(y_i, \hat{y}_i)$ is the training loss function which measures how well the model fits the training data, and $\Omega(f_k)$ is a regularization function that measures the complexity of the model. By incorporating these two components in the objective function, XGBoost is able to optimize the training loss function to encourage simple models that tend to have smaller variance and hence more stable predictions.

Since the functions f_k are trees, XGBoost uses an additive training algorithm to find the structure and the leaf scores of the trees. It starts from constant prediction and adds a new function each time. Then, the objective function is defined as follows:

$$obj^{(t)} = \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t), \tag{5.6}$$

where $g_i = \partial_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$, $h_i = \partial^2_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$, and $\hat{y}^{(t-1)}$ is the forecast value at time step $t-1$.

The model complexity is defined based on the number of leaves of a tree and the L_2 norm of scores on the leaves. Consider regression tree $f_t(x) = w_{q(x)}, w \in R^T, q : R^d \rightarrow$

$\{1, 2, \ldots, T\}$, where W is the vector of scores on leaves, q is a function assigning each data point to the corresponding leaf, and T is the number of leaves. In XGBoost the complexity is defined as:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2. \tag{5.7}$$

XGBoost is a kind of gradient boosting decision tree [68] which converts weak learners into strong learners. Unlike the common use of input vectors to calculate the similarity between the target and training data, gradient boosting constructs the boosted trees to obtain the importance score for each feature and compare them to each other. Then specifies the importance of each input variable to the training model. Basically, importance score indicates how effective and valuable each input variable was in the training of model by constructing the boosted decision trees. The more a feature is used to make key decisions with decision trees, the higher its score becomes.

For a single decision tree T, based on method introduced by Breiman et al. [35], the importance score for each predictor feature X_m can be calculated as follows:

$$w_m^2(T) = \sum_{i=1}^{N} \hat{\beta}_i^{\,2}, \tag{5.8}$$

where $w_m^2(T)$ is the squared importance of feature X_m for tree T, N is the number of internal nodes belonging to the decision tree, and $\hat{\beta}_i^{\,2}$ is the squared estimated improvement in the squared error risk measure over the node i. Each feature is considered as a split point that partitions the region in to two subregions at every node i. The selected feature is the one that yields maximum value of $\hat{\beta}_i^{\,2}$ over the entire region. Then, the squared importance of the feature X_m is calculated by summation of the squared improvement over the N nodes in each tree T. The feature squared importances are then averaged across all of the the decision trees within the model to produce the importance rank:

$$IR_m = \frac{1}{J} \sum_{i=1}^{J} w_m^2(T_i), \tag{5.9}$$

where IR_m represents the importance rank belongs to feature m, and J is the total number of trees in the model.

Input Variables

Load forecasting accuracy is influenced by a variety of factors. While new machine learning methods can fit highly non-linear models, the selection of input variables (features) is also very important and has a major impact on the accuracy of load prediction. The features that are utilized in the benchmark model can be categorized as follows:

Load: Electric load consumption usually exhibits a daily periodic pattern. In addition, load consumption profiles in adjacent days and weeks exhibit strong positive correlations. According to these characteristics we define three variables: load at time t for the previous two days, and load at time t on the same day of the previous week.

Weather: Weather is a major driving factor of electrical load consumption patterns. Temperature is the most frequently used weather variable in the load forecasting literature. In addition, the influence of other weather variables such as wind speed and relative humidity on the accuracy of load forecasting has been recently reported in [226,227]. Thus, we consider the wind speed and relative humidity as well as temperature in our benchmark model. To incorporate the recency effect [214] of temperature features we define seven variables as follows: the temperature at time t today and the same time of the previous day, maximum and minimum temperatures of the previous day, and average temperature of the past 3 hours, 6 hours, and 24 hours.

Time Cyclic: These features reflect the cyclic characteristic of the time of day and the type of day. Cyclic variables are extracted in order to capture the cyclic nature of load time-series [60]. By spectral analysis, the half day, day, week and year periods have been identified as the dominant frequencies of the load data [58]. For each frequency, a pair of variables is considered to represent the corresponding cycles:

$$\begin{aligned} c_1(t) &= \sin(\frac{2\pi t}{T}) \\ c_2(t) &= \cos(\frac{2\pi t}{T}), \end{aligned} \tag{5.10}$$

where t is the time indicator of the half hourly granularity sampling, which extends from 1 to 17520 for one year. The variable T represents the cycle period that is equal to 17520, 336, 48 and 24 for a year, a week, a day and a half-day cycle, respectively.

Type of Day: Load analysis has shown that load consumption on non-working days

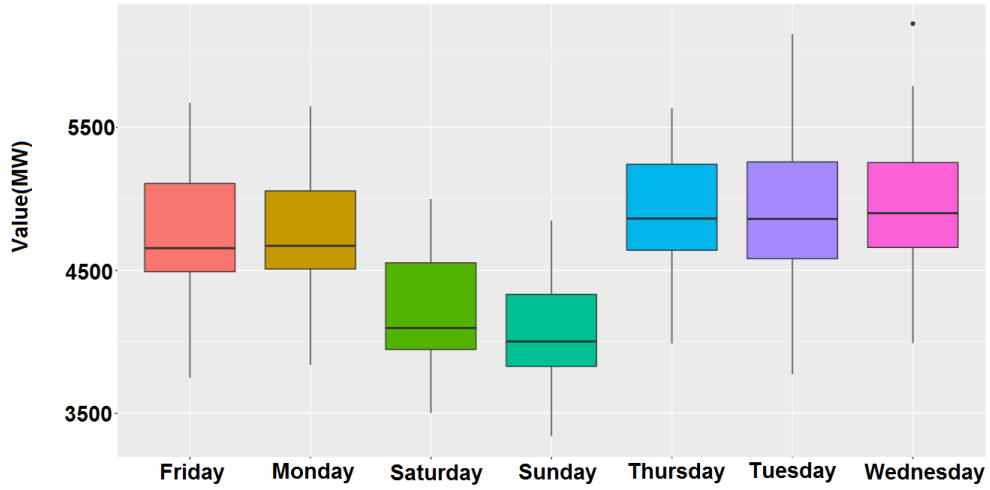


Figure 5.1: Average demand for each day of week in Victoria during 2016

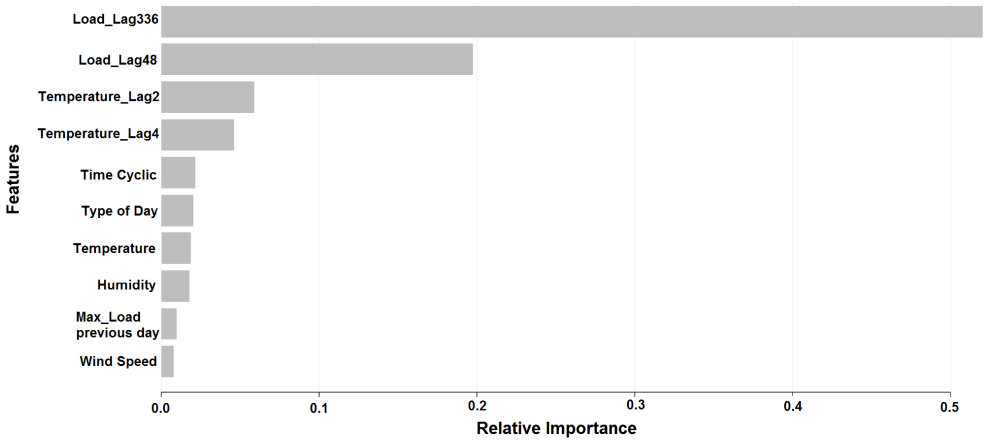


Figure 5.2: Ten top important features ranked by xGBoost for benchmark model

is lower than on working days due to the decrease in industrial loads on non-working days. In Figure 5.1 the average demand for each day of week in Victoria during 2016 is shown which confirm that the demand in weekend is lower than weekdays. We consider nine variables for the a type of day as follows: "Tuesday, Wednesday, Thursday", "Saturday", "Sunday", "Friday", "Monday", "Holiday", "Day before holiday", "Day after holiday", "Special day" which corresponds to special days which are not public holidays like the first week of January and school holidays, etc. We assign dummy variables to each of the mentioned day types through encoding them by one hot encoder. The one hot encoder maps an original element from the categorical feature vector with cardinality into a new vector with elements, where only the corresponding new element is one

while the rest of new elements are zeroes.

Given the candidate features in the previous section, we apply XGBoost algorithm to calculate the feature importance in prediction the electricity load for a day ahead. As it is shown in Figure 5.2, the most important feature is the amount of load at the same time and same day in the previous week, the second one is the the load at the same time in the previous day, and the third one is the temperature in the previous hour.

5.3 Methodology

In this section, first we introduce our proposed novel input variables to improve the accuracy of load forecasting. Then we introduce our robust and accurate short-term load forecasting framework in detail.

5.3.1 Apparent Temperature - A Novel Feature for Improving Load Forecast Accuracy

The temperature that we feel determines the amount of energy that we use for cooling or heating. The air temperature is usually used as a measure of how comfortable we feel when we want to use heating or cooling appliances. However, the air temperature is only one of the factors that have an impact on the assessment of thermal stress. Where other factors, principally humidity and wind speed, can vary widely from day to day, we need to consider the effect of all factors to assess the level of comfort realistically.

Apparent Temperature (AT) is a useful index which condenses all the factors of perceived temperature into a single value. In fact, it is the temperature equivalent perceived by humans caused by the combined effects of air temperature, relative humidity and wind speed [197]. In this chapter we adopt the AT formula based on the definition introduced by Robert et al. in [198] as follows:

$$AT = Ta + 0.33e - 0.7ws - 4, \tag{5.11}$$

where Ta is the temperature ($^{\circ}$C), ws is the wind speed (m/s), and e is water vapour

pressure (hPa) which can be calculated using the following formula:

$$e = \frac{6.105rh}{100} \exp(\frac{17.27Ta}{237.7 + Ta}), \tag{5.12}$$

where rh is the relative humidity (%).

After defining AT we can use this value to define two other weather dependent features, Heating Degree Days (HDD) and Cooling Degree Days (CDD). HDD is an indicator to quantify the amount of energy required to heat a building, while the CDD is a measure designed to indicate how much energy is needed to cool a building. Both HDD and CDD are defined relative to a based comfort temperature. We define the HDD and CDD based on AT as follows:

$$HDD = max(0, (mean(AT_t, AT_{t-1}, \ldots, AT_{t-6}) - 16.5)) \tag{5.13}$$

$$CDD = max(0, (18 - mean(AT_t, AT_{t-1}, \ldots, AT_{t-6}))) \tag{5.14}$$

5.3.2 Our Proposed Framework

In this section we first introduce our robust and accurate short-term load forecasting framework, then we explain each part of the framework in detail. Figure 5.3 shows the block diagram of the proposed framework.

We propose a cluster oriented ensemble learning framework to handle the different temporal variations and unbalanced aspects of the training load data set. The proposed framework produces multiple historical models for different temporal variations, and uses ensemble learning techniques to forecast the day-ahead load with high accuracy for both extreme weather conditions as well as normal days.

As shown in Figure 5.3a, the data set for load forecasting consists of historic load data, exogenous data, and peak load data. We partition the each of these data sets into three parts: training, validation, and test data sets. The training data is used to train the models, validation is used for tuning the parameters for each model, and the test data is used to evaluate the accuracy of the proposed framework.

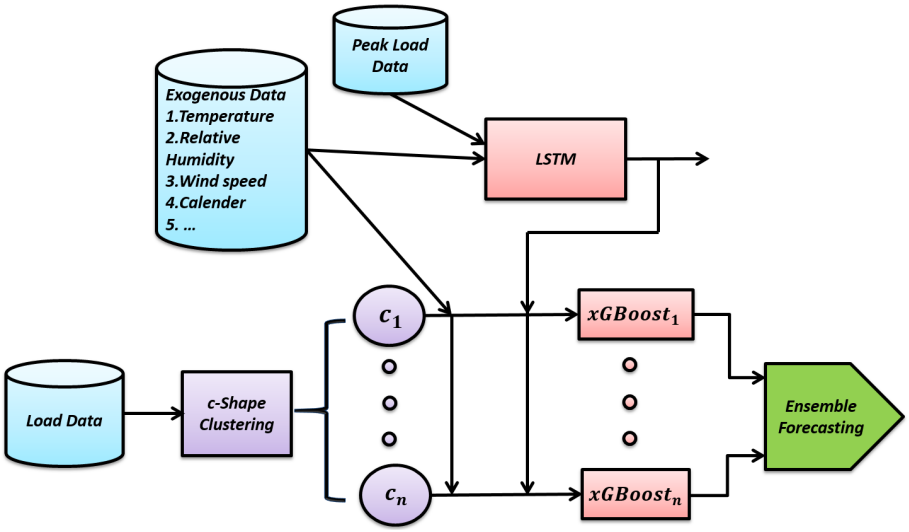
First we cluster the load data (training part) based on the daily load demand pattern by using the fuzzy c-Shape clustering algorithm which was introduced in Chapter

4. Fuzzy c-Shape clustering is a very fast and accurate clustering model that is able to group time-series data based on their shape. Then we produce the estimated morning and evening peak load for the target day by using an LSTM model. The input variables to the LSTM algorithm are the weather features, type of day, and peak load information. In the next step we produce the forecast for the target day by using the information that is related to each cluster. As a result, we produce n different forecasts for the target day. In the final stage we use the ensemble learning algorithm to make the final forecast. Next we describe each part of the framework in detail.

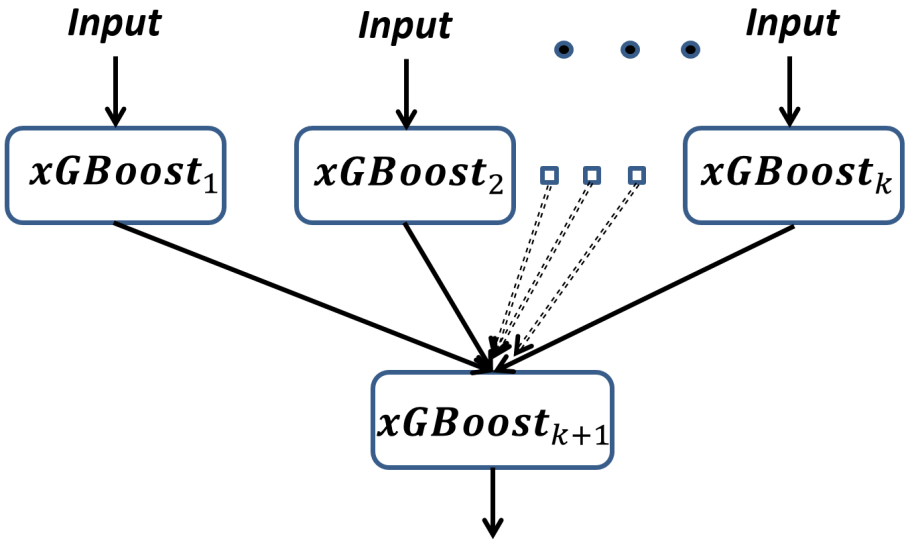
Fuzzy c-Shape Clustering

Numerous clustering algorithms have been used for the load prediction problem [53,178]. Determining the best algorithm depends heavily on the nature, purpose and mining objectives of the dataset. In this chapter, we use fuzzy c-Shape which was introduced in Chapter 4, a novel algorithm for time-series clustering that can conserve the shapes of time-series sequences. It uses a normalized, domain-independent form of cross correlation as its distance measure. Using this method, the c-Shape algorithm derives a shape-based distance measure for comparing the time-series efficiently and effectively. Then, based on the properties of the shape-based distance measure, c-Shape computes cluster centroids, which are used in each iteration to capture shared characteristics of the underlying data and update the assignment of time-series to clusters.

The extraction of a representative centroid for each cluster is a challenging task that critically depends on the choice of distance measure. By extracting the centroid, the clustering algorithm can effectively summarize a set of time-series in terms of only one sequence, and extract the most representative shape from the underlying data. Then, these extracted shapes or centroids are used for clustering the time-series. The robustness of c-Shape clustering has been experimentally evaluated against popular methods like k-Shape [170], K-means, K-medoid, hierarchical and spectral clustering methods, with combinations of the most competitive distance measures in Chapter 4. It has been that c-Shape outperforms all of these approaches in terms of accuracy on time-series data. The fuzzy c-Shape clustering algorithm consists of three main components: (1) a shape-based distance measure, (2) time-series shape extraction, and (3) shape-based time-series



(a) Framework



(b) Horizontal ensemble learning

Figure 5.3: (a) The cluster oriented ensemble learning framework for robust and accurate day ahead load forecasting. (b) Horizontal ensemble learning system based on XGBoost predictors.

clustering.

Long Short Term Memory Networks

A challenging type of predictive modeling problem is time-series prediction. Unlike regression predictive modeling, time-series also adds the complexity of a sequence dependence among the input variables. Recurrent Neural Networks (RNN) [51] are a powerful type of neural network, which is sequence based model and designed to learn the temporal correlations between previous information and the current conditions. It is trained by Backpropagation Through Time (BPTT) [221], however, RNN is faced with the vanishing gradient problem [22, 95] which limits its ability to learn long-term temporal correlations. To address this problem, Hochreiter et al. [96] introduced the Long Short Term Memory (LSTM) architecture by including a memory cell, and further improvement was presented by Gers et al. [75] with an extra forget gate. LSTM has been the most successful RNN architecture for creating large recurrent networks. Thus, it has been widely adopted to address difficult sequence problems in machine learning, and achieves state of the art outcomes.

To briefly explain the concept of LSTM, we adopt a similar description introduced by Lipton et al. in [139]. LSTM networks consist of memory blocks, instead of neurons, which are connected in layers. A block contains a memory cell for establishing temporal connections, gates for managing the state of the block, and an output. Consider the input sequence $\{x_1, x_2, \ldots, x_T\}$, where $x_t \in \Re^k$ is a k dimensional vector at time step t. LSTM defines an internal memory cell state and keeps it during the whole life cycle in order to establish the temporal connections. The memory cell state s_t captures the interaction with the intermediate output h_{t-1} and the subsequent input x_t. It uses the information of the outputs of the previous time step and the inputs of the present time step to update, maintain or erase the elements of the internal state vector. LSTM has three types of gates, (1) forget gate f_t: which conditionally decides about the information that should be thrown away from the block; (2) input gate i_t: which conditionally decides about the input values which should be used for updating the memory state; and (3) output gate o_t: which conditionally decides about the output values based on the inputs and the block's

memory. The formulation of all mentioned nodes are as follows:

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f), \tag{5.15}$$

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i), \tag{5.16}$$

$$g_t = \phi(W_{gx}x_t + W_{gh}h_{t-1} + b_g), \tag{5.17}$$

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o), \tag{5.18}$$

$$s_t = g_t \odot i_t + s_{t-1} \odot f_t, \tag{5.19}$$

$$h_t = \phi(s_t) \odot o_t, \tag{5.20}$$

where W_{gx}, W_{gh}, W_{ix}, W_{ih}, W_{fx}, W_{fh}, W_{ox}, and w_{oh} are weight matrices for the corresponding inputs of the network activation functions. $\odot$, σ, ϕ represent element wise multiplication, sigmoid activation function and tanh function, respectively. In Figure 5.4 the LSTM block and the sequential architecture of an unrolled LSTM are shown.

For training a simple one-layer LSTM recurrent network, the hyperparameter of the hidden output, s_t and h_t should be initialized. Usually these parameters are initialized to zero, i.e., $h_0 = 0$ and $s_0 = 0$. The three sigmoid functions in the LSTM block work as soft switches to decide which signals are allowed to pass the gates. These decisions for the forget f, input i, and output o gates are made based on the information of the current input x_t and the previous output h_{t-1}. The input gate determines the information that should be preserved in the internal state. The forget gate controls the information that should be forgotten from the previous state s_{t-1}. Finally, when the internal state is updated, the output gate determines the information that should be passed as the LSTM output h_t. These steps are repeated in each time step to learn the weights and biases based on minimization of the differences between the LSTM outputs and the actual training samples.

Ensemble Learning

Ensemble systems consist of an integration of two or more learning methods. The main goal is to integrate the capabilities of individual learning methods into a single com-

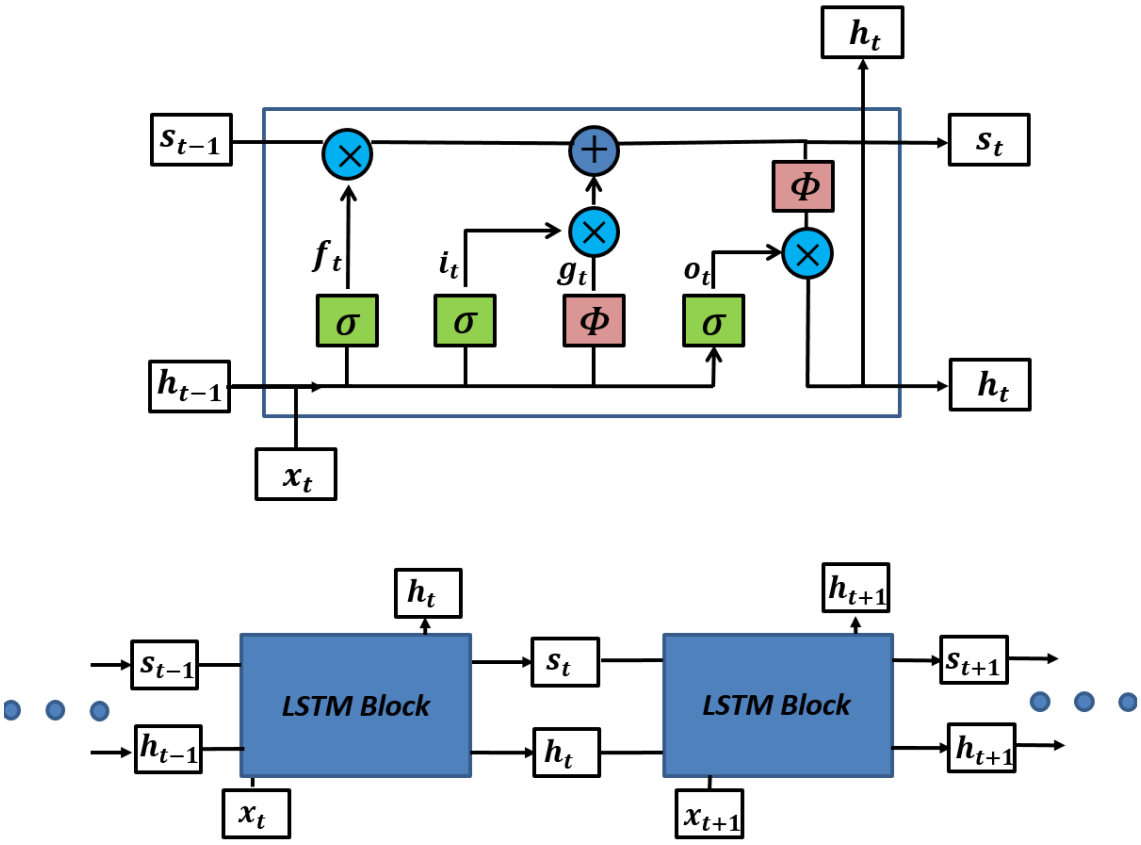


Figure 5.4: The top figure shows the architecture of an LSTM block and the bottom one shows the unrolled LSTM sequential architecture

prehensive system. There are two types of ensemble learning systems - horizontal and vertical [16]. Vertical ensemble learning systems are systematized in a vertical order such that the output of one learning method is used as an additional input to the next learning method. In contrast, horizontal ensemble learning systems are organized in a horizontal order such that the results produced by each method are combined to produce a single response to a given input. Horizontal ensemble learning systems integrate the predictive capabilities of the underlying individual forecasts. Therefore, horizontal ensembles are insensitive to errors caused by a single method and are hence more appropriate for critical applications such short-term load forecasting.

In this chapter we apply the horizontal ensemble learning approach as shown in Figure 5.3b. In the first step the c-Shape clustering is trained to group the electricity demand curves. Since c-Shape clustering is a shape based clustering algorithm, similar or related demand curves fall into the same clusters. In the next step, one predictor is trained for

each cluster. The proposed system is able to significantly improve the accuracy of prediction since it no longer relies on a single predictor, the data set is grouped into subsets that share the similar patterns, and each predictor is an expert in handling a type of energy market condition.

5.4 Case Studies

In this section we show the comparison results for the three introduced models in this chapter: (1) the benchmark model introduced in Section 5.2.2, called the "Benchmark" model; (2) the benchmark model with the addition of the new proposed features in Section 5.3.1 based on Apparent Temperature, HDD and CDD, called the "BenchmarkPlus" model; (3) our proposed framework in Section 5.3.2, called the "Framework" model. We evaluate the accuracy of the forecasts of the proposed framework, especially for difficult-to-predict days such as holidays and very hot days (which corresponds to extreme load conditions).

5.4.1 Data

The demand data is provided by the Australian Energy Market Operator. The data is aggregated smart meter data from all over five regions in Australia: Victoria, South Australia, New South Wales, Tasmania, and Queensland. The granularity of data is 30 minutes and the history of 5 years data ($2014-01-01$ to $2018-12-28$) have been used. Three years of data have been used for training, one year for validating and parameter tuning, and one year for testing the accuracy of different models presented in this chapter. The weather data is provided by the Bureau of Meteorology (BoM) [2] in Australia with a granularity of 30 minutes.

5.4.2 Experiments

Table 5.1 shows the comparison results in terms of mean absolute percentage error during the test period ($2018-01-01$ to $2018-12-28$) for the three different methods and five regions in Australia. As shown, our proposed framework improved the accuracy of day-

Table 5.1: MAPE value for day ahead forecast during test period (”2018-01-01” to ”2018-12-28”) for each five states in Australia

	Method		
State	Benchmark	BenchmarkPlus	Framework
Victoria	3.2%	2.9%	**1.9%**
Tasmania	3.5%	3.1%	**2.2%**
South Australia	4.5%	4%	**3.3%**
New South Wales	3.7%	3.4%	**2.5%**
Queensland	3%	2.8%	**2%**

ahead forecasts over each region by 1%. Based on the research by Hong et al. [103], every 1% reduction in mean percentage absolute error leads to savings of around $\$300,000$ per year for a utility with $1GW$ peak. Note that the aggregate peak load in these five region is $33GW$. So, any small improvement in the accuracy of load forecasting leads to substantial savings for utility companies.

Figure 5.5 shows the monthly comparison analysis of the day-ahead forecast accuracy among the three proposed methods during the test period in Victoria. As shown, the accuracy of the BenchmarkPlus model is slightly better than the Benchmark model, which means that by using apparent temperature instead of temperature the accuracy of the model is improved. Also, it is clear that the Framework model is more accurate than the two other models, especially during the summer periods in Victoria (January, February, March, November, December), when the difficult to predict days such as new year eve, school holidays and high temperature conditions are occur.

Figure 5.6 shows how the Framework model produces a stable prediction in the first two weeks of 2018, which contains the difficult to predict days which arising from new year and high temperature days, which impact on the consumption of the energy by customers. As shown, during this period of high demand days ($2018-01-05$, $2018-01-06$, $2018-01-11$, $2018-01-12$) the proposed framework has accurately predicted the peak.

The accuracy of the proposed models are also analyzed in the extreme weather as well as normal weather conditions. The extreme weather conditions are defined based on two standard deviations from the mean temperature as follows:

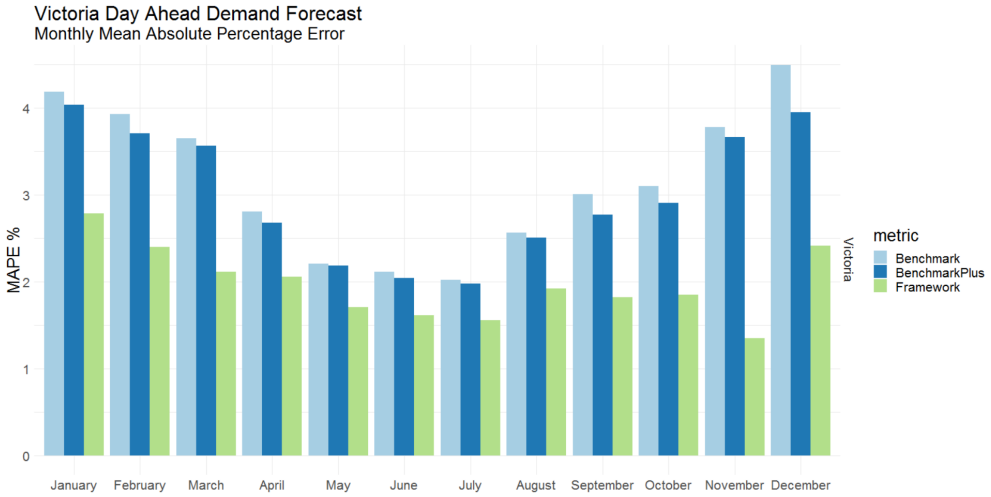


Figure 5.5: Monthly analysis of day-ahead forecast accuracy in terms of MAPE during the test period 2018-01-01 to 2018-12-28, in Victoria

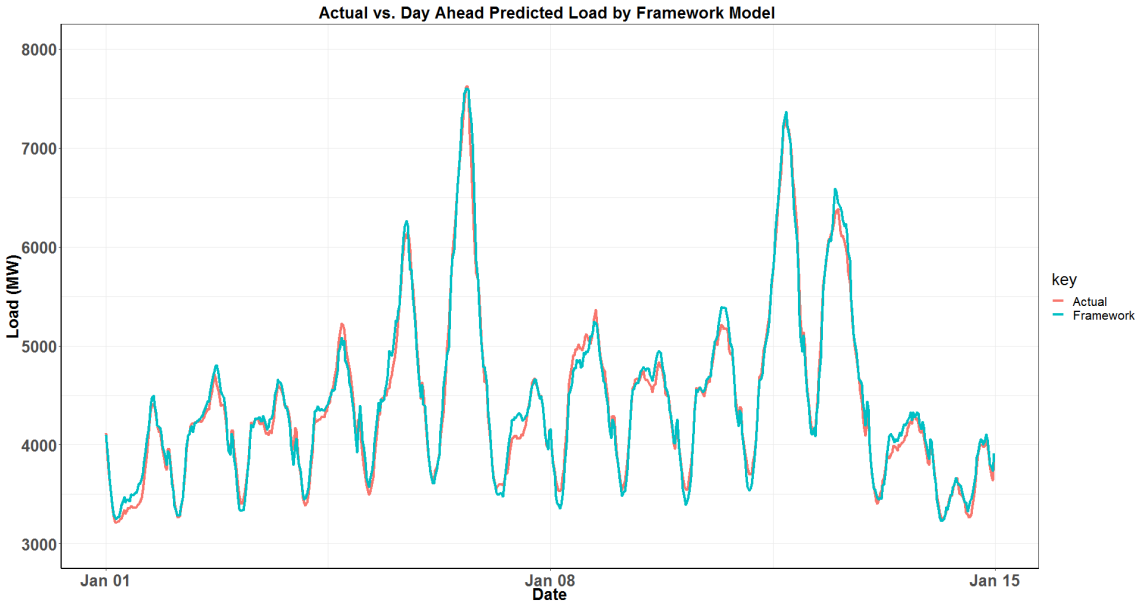


Figure 5.6: Actual vs. day-ahead forecast by Framework model for period 2018-01-01 to 2018-01-14, in Victoria

$$Weather = \left\{ \begin{array}{ll} Extreme & Temperature > \mu + 2\sigma \quad or \\ & Temperature < \mu - 2\sigma, \\ Normal & Otherwise \end{array} \right\}.$$

where μ and σ are the average and standard deviation of the temperature during the test period, respectively. Figures 5.7 and 5.8 show the box plot of extreme and normal weather conditions in the test period in Victoria for the three proposed models. In this

study the temperatures above 28°C and below 3°C are considered as extreme weather situations in Victoria, where the average of temperature during the test period is 15°C and the standard deviation is 6°C. As shown in Figures 5.7 and 5.8, the Framework model has lower absolute error than the Benchmark and Benchmarkplus models in both normal and extreme weather situations, which demonstrates the robustness of the proposed framework.

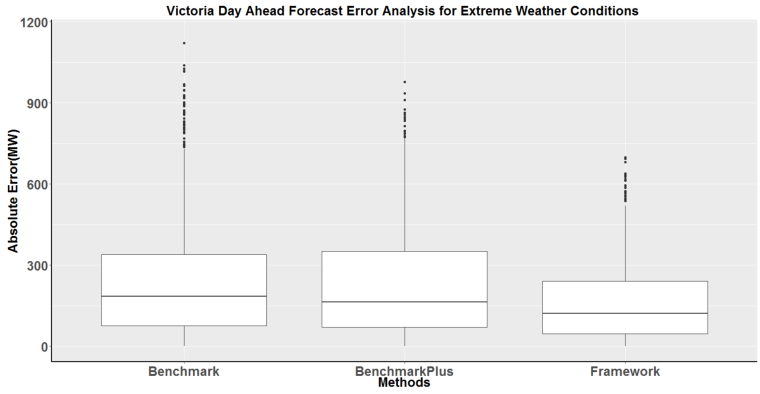


Figure 5.7: Day-ahead forecast Absolute Error box plot during the extreme weather conditions in the test period 2018-01-01 to 2018-12-28, in Victoria

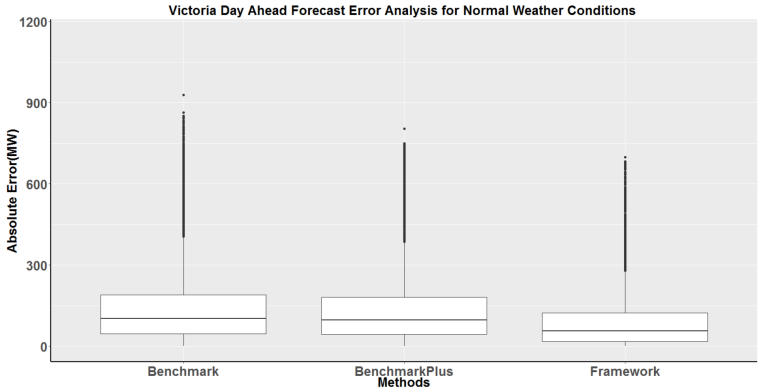


Figure 5.8: Day-ahead forecast Absolute Error box plot during the normal weather conditions in the test period 2018-01-01 to 2018-12-28, in Victoria

5.5 Conclusion

In this work we introduced a novel framework for improving day-ahead load forecasting in the electricity grid. We investigated how adding new features to the load forecast-

ing system can result in improvement of forecasting accuracy. In addition, because of the non-stationary nature of the demand data, we introduced an unsupervised ensemble learning approach, which can achieve high accuracy in situations that have not seen before by the forecasting system. We tested our proposed model for the most difficult prediction times in five regions in Australia. The results demonstrate the robustness and accuracy of our model even for these unusual and extreme load conditions.

While our experimental results are encouraging, they are by no means definitive. There are some suggestion for improving the accuracy of the framework by including some indicative for the solar rooftop generation as an input to the model. As, we know the number of solar rooftop installations are increasing, as this solar generation are behind the meter, they have been introducing new challenges for operational power system.

Ultimately, approaches to improve operational power forecasts by improving models, data and input accuracy will reach some sort of limit; physical, economic, etc. An understanding of the uncertainty in the forecasts that operational power grid uses is crucial to continue to operate the system in a secure manner in the real-time and into the future. In the next chapter we propose a dynamic stochastic decision support tool, based on Dynamic Bayesian Belief Networks, to quantify the level of uncertainty in order to improve situational awareness and understand the risks to power system operators.

Chapter 6

Data-Driven Dynamic Probabilistic Reserve Sizing Tool

In practice, an understanding of the uncertainty in the forecasts that an operational power grid uses is crucial in order to operate the system in a secure manner in real-time and into the future. Due to the variability and partial unpredictability of renewable energy sources (RES), generation reserve sizing is key for managing operational risks and enhancing grid reliability. Most existing approaches for reserve sizing estimate the forecasting error for RES using either simple point forecasts, or probabilistic forecasts that assume a predefined parametric model of the underlying distribution, such as a Gaussian distribution. However, these approaches are unable to accurately model the probability of critical events which occur in the tails of the distributions, which is vital for operational risk management. Further, they do not account for multiple dynamic factors that may impact operational risk. In this chapter, we introduce a data-driven dynamic probabilistic reserve sizing method based on the artificial intelligence technique of Dynamic Bayesian Belief Networks. The method considers the actual underlying distribution of forecasting errors from the availability of conventional generators (forecasting errors of their available generation capacity), demand, RES, and prevailing conditions like weather and market prices. In addition we introduce a new dynamic metric for calculating the reliability level of the power grid in order to provide a real-time stochastic decision support tool for power system operators. The proposed method is demonstrated using 7 years of real historical data with a granularity of 30 minutes from the power system in Australia, and has been implemented and is currently used by the Australian system operator.

6.1 Introduction

6.1.1 Background and Literature Review

WITH the growing dependence on renewable energy sources (RES) for electrical power generation, the power grid has become increasingly volatile and less predictable. Renewable energy sources are weather dependent, which can be difficult to predict accurately. Other factors such as energy demand, consumer behavior, energy prices, and transmission constraints further increase the uncertainties associated with demand and supply modeling [220].

A wide range of studies have addressed the problem of power system imbalances and relevant operational reserve requirements due to the integration of renewable energy sources [37, 93, 99, 144], ranging from deterministic to probabilistic methods, as well as static to dynamic approaches.

In deterministic methods, reserves are usually sized in accordance with a specific event, like the largest credible contingency. However, these approaches are not adequate in the presence of many sources of uncertainty, because they do not model the stochastic nature of the system, the probability of events with different severities, and the dependencies between different sources of imbalance [99].

Hence, probabilistic reserve sizing techniques have emerged in the literature as more effective methods. Probabilistic approaches are able to provide more information about the system by including many sources of uncertainties for reserve sizing, so that a required level of system reliability is met. These methods require detailed information of the sources of imbalances in the system, their probability distributions, and their correlations, and apply the reliability level of the system as a cut-off to assess the size of the reserve. By assuming that the sources of uncertainties are independent, reserves can be calculated based on the convolution of the probability density function of all independent sources of imbalances. As a result, the probability of having insufficient operating reserve capacity can be determined by a risk measure such as the reliability level or the probability of loss of generation [79, 144]. Simple statistical methods such as using the standard deviation of imbalances also belong to this family and are widely used in power system operation [100].

Probabilistic approaches can be further categorized as parametric or non-parametric. In parametric methods, the sources of uncertainties are assumed to follow a predefined family of distributions. In [163] and [130] the Gaussian distribution is used for mod-

eling uncertainties, while in [37, 55, 62, 130, 147, 163] the error in wind power forecasts is modeled by Weibull, Gamma, Levy alpha-stable, and Skew-Laplace distributions, respectively. By assuming a predefined probability distribution for the uncertainties, there is the risk that an inappropriate choice of distribution will result in a poor forecast of the tails of the distributions, which contain important information about unexpected events [220]. In contrast, non-parametric methods do not make strong assumptions about the underlying distribution of the sources of the uncertainties. By not making such assumptions, they are free to learn a much wider range of functional forms from the training data. Among non-parametric methods, non-parametric kernel based approaches are widely used to forecast wind power generation [24, 234, 235]. In [24, 234, 235], time adaptive quantile-copula methods are used to model conditional forecast errors for single or multiple wind farms.

As mentioned earlier, reserve sizing approaches can also be categorized into static and dynamic methods. Static methods typically pre-determine reserve requirements for long time periods (e.g., one year), while dynamic methods assess it for more frequent periods, depending on the current or expected status of the system. It is worth noting that probabilistic sizing can also be done in a static or dynamic way, but deterministic sizing is usually done statically. While in [25, 55, 136, 144, 163, 185] static reserve sizing methods have been investigated, in [37, 119, 147, 235] dynamic methods have been proposed. As short-term variations in the generated power by RES and the electricity demand from gate closure through to real-time result in dynamic balancing requirements, to ensure grid reliability, a dynamic reserve sizing model is more appropriate.

Usually, dynamic reserves are sized on a day-ahead basis using time adaptive probability density functions. Thus, when the uncertainty associated with all the factors considered in a power system is low, reserves can be reduced, which in turn can result in large energy and cost savings. However, previous works on dynamic reserve sizing either use parametric approaches like [37, 147, 235], which do not consider the true distributions of forecasting errors, or they use non-parametric approaches [38, 119] in combination with convolution theory to calculate the error distribution associated with the system. In practice, this approach is computationally intensive, and can be impractical in large, complex systems including large volumes of data. To address some of these chal-

lenges, one could make use of data-driven Artificial Intelligence (AI) techniques. For example, Bayesian Belief Networks (BBNs) have been used in the context of power systems to deal with forecasting electricity demand and RES; management, control, operation and optimization of the network; data analysis, and risk evaluation [25, 136, 185]. However, the power systems conditions, and thus its operational risk, change continuously, and therefore time-varying, dynamic methodologies are needed.

6.1.2 Novelty and Contributions

In this chapter, we introduce a data-driven dynamic probabilistic reserve sizing method based on Dynamic Bayesian Belief Networks (DBBNs), which is suitable for diverse problems of varying size and complexity, where uncertainties are inherent in the system. Because BBNs have a simple casual graphical structure, they are one of the most effective models for representing uncertainties [33]. In addition, BBNs are one of the best models for simplifying conditionalization, which helps in designing the decision making process [31]. Our contributions are as follows:

- We introduce a novel metric of reserve forecast uncertainty, called Forecasting Uncertainty Measurement (FUM), which can be used to calculate the reliability level of a power system and regulate the size of the reserve in a dynamic manner. Previous methods calculate the reliability level of a power system by considering predefined contingency scenarios, which have limitations in terms of accuracy and transparency, such as only considering the largest generator risk. In the new proposed measure FUM, possible contingencies and errors from historical data are considered and can be automatically updated. This new measure conveys information about the uncertainties associated with the excess supply available in each region of a power system in MW, which is easy to understand and practical for grid operators to assess the probability of a shortfall in the available capacity reserves in each region of the power grid on a continuous basis.

- For the first time, we introduce the use of DBBNs for dynamic probabilistic reserve sizing, and demonstrate their suitability in modeling system complexity.

- To the best of our knowledge, this is the first time that reserve sizing has been an-

alyzed in a dynamic manner for a real hybrid system with high complexity arising from a wide variety of power sources, such as grid-connected and embedded RES and conventional generators, while taking into consideration a range of prevailing conditions, such as market prices, weather, and time of day.

- The performance of the proposed method is validated on real data from the Australian power system, and through actual on-site implementation by the Australian system operator, AEMO (Australian Energy Market Operator).

This chapter is organized as follows: Section 6.2 presents the relevant background on BBNs and DBBNs. Section 6.3 gives the details of our proposed methodology by introducing (1) a novel dynamic stochastic decision support tool for forecasting reserve uncertainty, and (2) a novel measure of the reliability level of the grid. Section 6.4 shows our experimental results based on real data from the Australian grid. In Section 6.5 we conclude and discuss future work.

6.2 Background on Bayesian Belief Networks

This section introduces Bayesian Belief Networks and their applications, as well as their extension to Dynamic Bayesian Belief Networks, which are the basis of our proposed method.

6.2.1 Bayesian Belief Networks Formulation

Bayesian Belief Networks (BBNs) [172] are probabilistic models used in AI to reason about uncertainty [42]. They can be used to investigate and present causal relationships between essential components of a system in a simple and understandable manner. They can be readily modified and used to incorporate missing data through the application of Bayes theorem [232]. They are also useful for calculating the impact of interventions such as examining alternative policies or decisions. In addition, they can be used to investigate dependencies among these casual relationships. In particular, BBNs solve problems in which uncertainty arises due to complexity, incomplete knowledge, or stochastic phenomena.

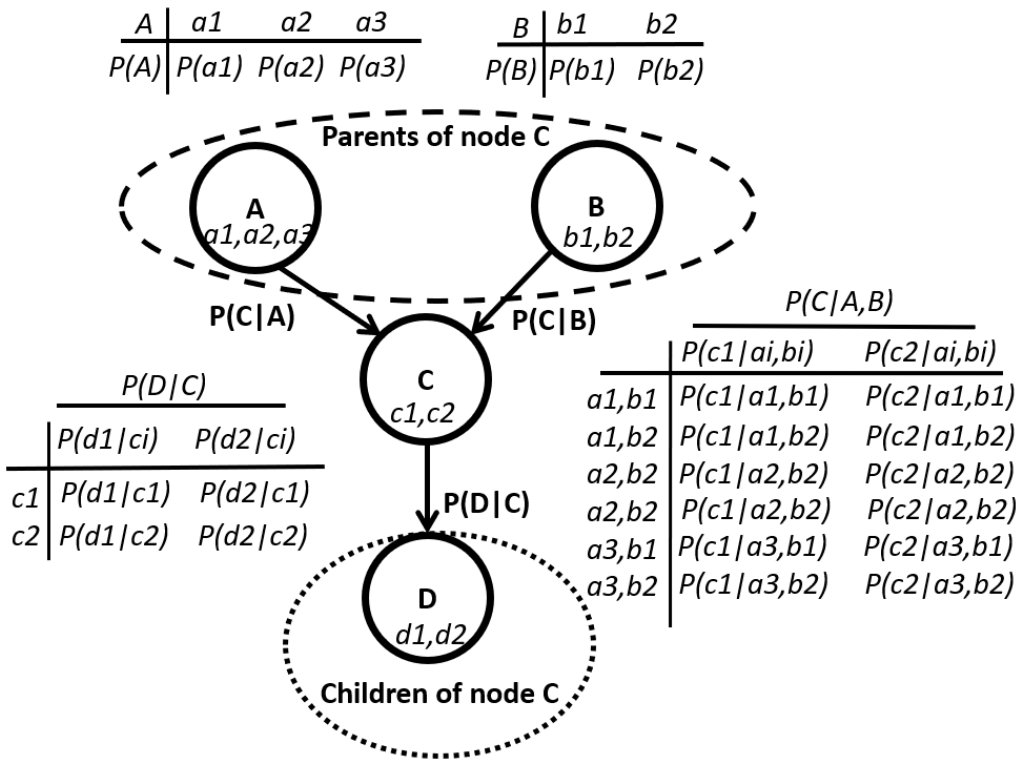


Figure 6.1: Graphical representation of a BBN with parent and child nodes

BBNs are probabilistic graphical models based on a directed acyclic graph that comprises nodes connected by edges with an associated direction, and with no cycles [42]. The nodes in the networks depict a set of random variables, $V = V_1, \ldots, V_i, \ldots, V_n$, and directed arcs connect pairs of nodes, $V_i \rightarrow V_j$, representing the dependencies between variables. In each pair of connected nodes, the parent node affects the child node according to the direction of the arc between them. The absence of an arc between two nodes denotes that those nodes are conditionally independent. For example in Figure 6.1, nodes A and B are independent of each other and node C is conditionally dependent on its parents (A and B). As tables shown in the Figure 6.1, each variable has an associated prior or conditional probability.

After assigning an appropriate topological model to the BBN, the relationships between connected nodes should be quantified. The relationship between a child node and all its parents is described by a Conditional Probability Table (CPT). The CPT is defined as follows: (i) each row represents the probability of being in a state, given a combination of values of the parent states, in Figure 6.1, for children nodes C and D, each matrix

Table 6.1: CPT of the variable D given the information of variable C.

C	c_1	c_2
D	d_1	$P(D = d_1 \mid C = c_1)$ $P(D = d_1 \mid C = c_2)$
	d_2	$P(D = d_2 \mid C = c_1)$ $P(D = d_2 \mid C = c_2)$

presents the probability of child node being in a specific state given the combination of its parents states, (ii) each row must sum to one, and (iii) a node without parents has one row corresponding to its prior distribution (in the Figure 6.1, nodes A and B). The size of the CPT for each node is the product of the numbers of states of the child node and all its parent nodes. For example consider two nodes C and D in the Figure 6.1. Node C has two states (c_1 and c_2) and node D has two states (d_1 and d_2). Based on the conditional probability $P(C|D)$, a CPT is associated with node D in order to define probability distributions over the states of D given the states of C as shown in Table 6.1.

Once a problem and its uncertainty are modeled by a BBN, it can be used to "reason" about the problem by applying new evidence, i.e., as a probabilistic inference system. Such an inference system can compute the posterior probability distribution for a set of *query* nodes, given the values for some *evidence* nodes. The types of evidence can be categorized as: (i) *definite evidence*: definite information that variable X has a specific value x; (ii) *negative evidence*: information that variable Y is not in a certain state y_1, although it may take any other values; (iii) *likelihood evidence*: uncertain sources of information. The types of reasoning using BBNs can be categorized into four categories: (i) *diagnostic* reasoning, from indications to cause; (ii) *predictive* reasoning, from new knowledge about causes to new beliefs about results; (iii) *inter-causal* reasoning, about the common causes of a predictable result; and (iv) *mixed* reasoning.

For example in the above mentioned BBN, the marginal distribution $P(D = d_1)$ can be calculated as:

$$\begin{aligned} P(D = d_1) = {} & P(D = d_1 \mid C = c_1) P(C = c_1) \\ & + P(D = d_1 \mid C = c_2) P(C = c_2). \end{aligned} \tag{6.1}$$

As a result, a BBN can calculate the probabilities associated with the state of a variable given the state of other variables, providing a powerful model for complex systems.

BBNs perform probabilistic updating by incorporating new information and evidence, and provide a combination of predictive and diagnostic reasoning by incorporating new information and evidence through Bayes' theorem. Bayes' theorem defines the conditional probability of x given y as:

$$p(x|y) = \frac{p(x)p(y|x)}{p(y)}, \tag{6.2}$$

where x and y are events, $p(x)$ and $p(y)$ are the probabilities of x and y, $p(x|y)$ is the conditional probability of x given y and $p(y|x)$ is the conditional probability of y given x. Thus, BBNs can update the probabilities of a set of events based on the observed information and the BBN structure.

6.2.2 Dynamic Bayesian Belief Networks

Most of the events that happen around us in every day life cannot be depicted based on a fixed point in time. Instead, they should be explained through a sequence of observations that lead to the inference of one final event. Time is an important dimension in the field of AI and reasoning. However, BBNs variables are time independent, i.e., a static model that does not model temporal relationships explicitly. In contrast, Dynamic Bayesian Belief Networks (DBBNs) [71, 156, 171] consider how variables change with time. DBBNs can reason about changes over time by relating variables to each other over a sequence of time steps. In this way, a user can monitor and update the system continuously over multiple time steps.

DBBNs consist of a prior network, which comprises the prior probabilities for all of the variables in the network at time step $t = 0$, and a transition network, which encodes the probabilities for each variable conditioned on other variables for all time steps, $t = 1, 2, \ldots, n$. For a DBBN in each time step, the variables and their associated probabilities are the same, except for the network in the initial time step, which has its own probability distribution. Thus, we can express a DBBN in terms of the probability distribution function on the sequence of n hidden-state variables $X = \{x_0, \ldots, x_{n-1}\}$ and the sequence of n observable variables $Y = \{y_0, \ldots, y_{n-1}\}$, where n is the time of the given

event of interest. The joint probability distribution can be formulated as:

$$P(X,Y) = \Pi_{t=1}^{n-1} P(x_t|x_{t-1}) \Pi_{t=0}^{n-1} P(y_t|x_t) P(x_0), \tag{6.3}$$

where $P(x_t|x_{t-1})$ is the state transition pdf that defines the time dependencies between states, $P(y_t|x_t)$ is the observation pdf that defines the dependencies between the observation nodes and other nodes at time step t, and $P(x_0)$ is the initial state distribution that provides the prior probability distribution at the starting point of the process. In fact, Dynamic Bayesian Belief Networks (DBBNs) are static Bayesian networks that are modeled over an arrangement of time-series. In a Dynamic Bayesian Network, each time slice is conditionally dependent on the previous one. Figure 6.2 shows a simple static BBN (left) and a DBBN (right). The arcs in the Dynamic BBN figure show the dependency of each time slice on its previous one. A DBBN can be continuous, discrete or a combination of these two based on the type of the state space of the observation and hidden variables. Similar to a BBN with a DBBN we can perform the following tasks: (i) Inference: estimating the pdf of unknown state variables given the known observations to initialize the probability distribution. (ii) Decoding: discovering the best estimates of the probability values for the sequence of hidden states that have produced the known states of observations. (iii) Training: Estimating the parameters of a DBBN system given a number of observed sequences to infer the best model for the system. (iv) Pruning: applying sensitivity analysis to find out which nodes are semantically important for inference in the DBBN structure and removing the nodes that are not important.

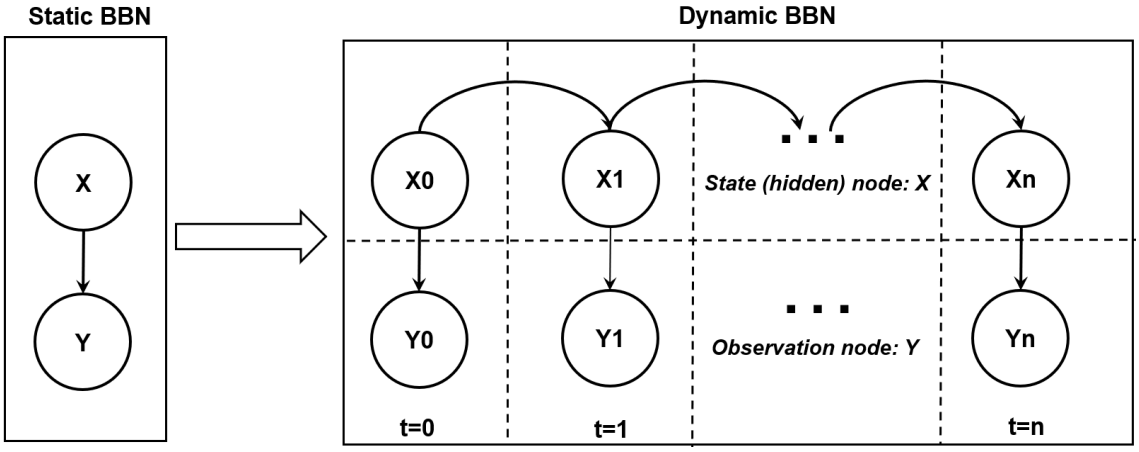


Figure 6.2: Static BBN vs Dynamic BBN

6.3 Methodology

Our aim is to apply stochastic modeling to carefully combine contingency scenarios (unexpected changes in network topology) with renewable scenarios (uncertainty of renewable generation outcomes) and demand side uncertainties to create a risk-weighted analysis of grid reliability. Furthermore, we aim to introduce a new metric that conveys information about the uncertainties, in MW, associated with the excess supply available in each region in a power system. This novel metric should be easy to understand, automatically updated with changing conditions, and practical for power grid control to continually assess the probability of a shortfall in available capacity reserves that could lead to load shedding.

6.3.1 Problem Statement

The key objective of the analysis is to forecast and quantify how much *excess*[1] power generation is available on the network above the load demand at time $t+h$, where $h = \{1, 2, \ldots, H\}$. The power generation component comprises the following three subcomponents: conventional generators, solar, and wind power generators. Note that conventional generators typically over-forecast their capacity on extremely hot days because these generators (coal and gas) may have to reduce their production to keep their cooling process within safe limits. Therefore, the mean forecast value for the "excess" supply above the demand on the network at time $t+h$ can be calculated as:

$$R\hat{X}S_{t+h} = \hat{G}_{t+h} + \hat{S}_{t+h} + \hat{W}_{t+h} - \hat{D}_{t+h}, \tag{6.4}$$

where $R\hat{X}S_{t+h}$ is the level of forecast available regional power oversupply at time $t+h$, $\hat{G}_{t+h}$ is the forecast value of the available capacity of conventional generating units at time $t+h$, $\hat{S}_{t+h}$ and $\hat{W}_{t+h}$ are the amount of predicted solar and wind power generation at time $t+h$ respectively, and $\hat{D}_{t+h}$ is the operational demand forecast value at time $t+h$.

Each of these forecasting systems has inherent errors, and these errors also depend on the weather conditions and some exogenous variables. Here, to demonstrate how each

[1] Excess power generation conventionally refers to the available generation capacity above the level of demand, which can be used as operational reserve.

forecasting system is modeled, we provide a brief explanation of the demand forecasting system. The same method is applied for solar and wind power generation forecasting systems as well.

Assume that we have a historical load demand data set $\mathbf{D}_t = \{D_1, \ldots, D_t\}$, comprising t observations. The goal is to predict the H future observations $\{D_{t+1}, \ldots, D_{t+H}\}$. Then we can define the electric load at time $t+h$, where $h \in \{1, 2, \ldots, H\}$ by producing a different model at each forecast horizon h as follows:

$$D_{t+h} = Y_h(X_t) + \epsilon_{t+h}, \tag{6.5}$$

where ϵ_{t+h} denotes the model error, $Y_h(.)$ is the model, which is based on the conditional expectation $E[D_{t+h} \mid X_t]$, and X_t is defined as follows:

$$X_t = (d_t, g_{t+h}), \tag{6.6}$$

where d_t is a vector of lagged demand data occurring prior to $t+1$, and g_{t+h} is a vector of exogenous variables at $t+h$.

Any forecasting system is associated with inherent prediction errors. As a result, forecasting the mean value of RXS does not provide meaningful information about the uncertainty of the future excess power generation in the network. Instead, we need to forecast the entire distribution of the future regional excess supply and provide a meaningful measure to assess the uncertainties associated with the excess supply available in each region of a power system. This new measure should be easy to understand and practical for power grid decision makers to assess on a continual basis the probability of a shortfall in available capacity reserves, leading to load shedding in each region.

In summary, our goal is to forecast the pdf of the random variable $Error_t(RXS)$ at time $t+h$, where:

$$Error_t(RXS) = \hat{RXS}_t - RXS_t, \tag{6.7}$$

and $\hat{RXS}_t$ and RXS_t are the mean forecast and the actual value of the regional excess

supply at time t, respectively. We can also define the variable $Error_t(RXS)$ as follows:

$$\begin{aligned} Error_t(RXS) = Error_t(G) + Error_t(S) \\ + Error_t(W) - Error_t(D), \end{aligned} \tag{6.8}$$

where $Error_t(G)$, $Error_t(S)$, $Error_t(W)$, and $Error_t(D)$ are the errors produced by each forecast at time t. As shown in Equation (6.8), the $Error_t(RXS)$ depends on the forecasting systems' error for the demand, wind, solar and conventional generators. It is worth mentioning here that all forecasting systems are also dependent on weather conditions and some exogenous variables such as price and time of day.

Next we show how we infer the error distribution of the regional excess supply by considering all the above mentioned uncertainties in a dynamic manner by using a DBBN.

6.3.2 DBBN Model to Forecast Available Reserve Uncertainties

We use a Bayesian Belief Network to understand the conditional dependency of the uncertainties in the system and provide information to decision makers in the form of an Excess Generation Reserve value. This value determines, with a certain probability, how much excess generation over demand is available in each region. In practice, in the implementation carried out by AEMO, this information is produced from every 30 minute trading interval for the next 8 days.

To achieve this aim we need to understand the current level of uncertainty in the power system when a forecast is made. DBBNs can be used to assess the conditional interdependence of forecasting uncertainty from various forecasting systems. In addition, DBBNs provide an estimate of how these uncertainties vary given the prevailing conditions, like the temperature at the time the forecast is executed. For example a 6 hour-ahead forecast that is executed at 38 degrees may have greater uncertainty than a 6 hour-ahead forecast that is executed at 18 degrees. In this study, the DBBNs are designed so that, given the prevailing weather and reserve conditions of the network, they can compute the probabilities associated with the historical forecasting errors in MW.

Figure 6.3 introduces the DBBN structure for forecasting the error distribution of re-

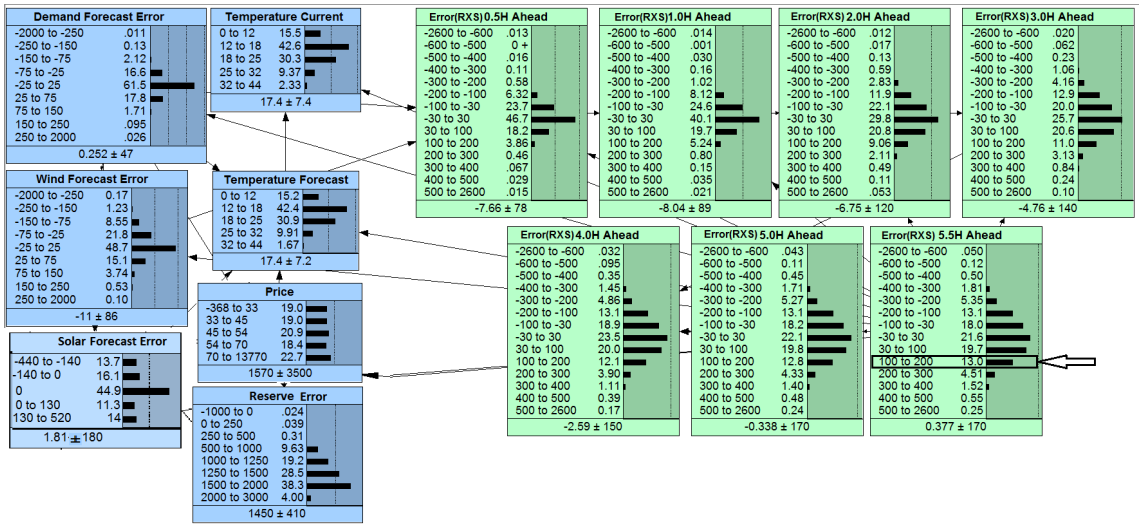


Figure 6.3: Sample DBBNs using seven years of historical data (2010-2017) in South Australia

gional excess supply. In this framework, we have included the forecasting errors associated with both conventional and intermittent generators, as well as generation prevailing conditions like temperature and market prices. Each system has a forecast and an associated forecasting error for each 30 minute trading interval. The network is trained on all of the available historical error forecasts and the associated prevailing conditions at the time of the forecast. In this network, the parent nodes are named: Demand Forecast Error, Wind Forecast Error, Solar Forecast Error, Temperature, Temperature Forecast, Reserve, and Price, which represent the prevailing conditions that existed just prior to the forecast run. The other nodes, namely $Error(RXS)nHAhead$ represent the child nodes or forecasting error for each of the next 30 minute trading intervals $n \in \{0.5, 1, \cdots, 5.5\}$. The black horizontal bars represent the probability associated with each of the corresponding bins. For example, in Figure 6.3 the highlighted horizontal bar shows that the probability of having a forecasting error between 100 and 200MW, at 5.5 hours in the future from the forecast run is 13.0%.

6.3.3 Sensitivity Analysis

Once a DBBN network has been trained it is possible to statistically assess the impact that each of the prevailing conditions (input nodes) has on the forecasting errors. Us-

ing just the link structure of the net, it can be determined which nodes are completely independent of other nodes. However, dependence is a matter of degree and once the net has been trained it can be used to assess how changes in the prevailing conditions change the probabilities or uncertainties in our forecasting error nodes (output nodes). From this analysis, in the case study applications carried out (see Section 6.4) it has been determined that, for forecast lead times below 8 hours, the most significant prevailing conditions (the conditions that cause the largest change in forecasting uncertainty) are: (a) Temperature forecast, (b) Wind forecast, (c) Last operational demand error (i.e., the error of the demand forecast from the previous 30 minute trading interval). In addition, prevailing conditions that were assessed and determined to not have a significant impact on the forecast error include: (i) Price, (ii) Last wind forecasting error (i.e., the error from the wind forecast from the previous 30 minute trading interval), and (iii) Current actual RXS.

6.3.4 Lack of Reserve Conditions

The output of the DBBN is the probability distribution of the excess generation reserve available in each region in the power system. The next step is to apply the reliability level of the power system as a cut-off to regulate the size of these reserves and provide information for decision makers. To this end, three reliability levels are defined in the Australian system based on Lack Of Reserve (LOR) conditions.

Lack of Reserve (LOR) conditions [1] indicate that there is a likelihood that the system may not have sufficient reserves to meet demand if there is a major, unexpected event. A typical practice in grid management to assess lack of reserve conditions is to consider different scenarios based on the risk of credible contingencies, such as the loss of the largest generator, or interconnection, in any region. Three levels of LOR are defined as follows: $LOR1$ is declared for a region(s) when consecutive occurrences of both the largest and the second largest relevant credible contingency events would result in load shedding occurring as a result of a shortfall of available capacity reserves; $LOR2$ is declared when the occurrence of the largest relevant credible contingency event would result in load shedding; and $LOR3$ is the highest level of alert, which indicates the balance is so tight that load shedding is imminent or has begun.

6.3.5 Forecast Uncertainty Measure (FUM)

The definition of LOR has not adequately taken into account the range of risks to reliability that now exist, given the changing nature of the power grid. In addition, it does not take into account any forecasting uncertainties when the reserve assessment is calculated.

In this section we therefore introduce a revised definition for LOR that also takes into account a measure of the reserve forecast uncertainty. As predictions are made further into the future about forecast excess supply, the uncertainty of those predictions increases due to weather-related changes that affect the levels of demand and the availability of RES output. In order to ensure that adequate reserve capacity is available, we need to define a reasonable upper limit on the forecast error. We do this by defining a measure called the Forecast Uncertainty Measure (FUM), which quantifies a confidence limit on the forecasting error of a probabilistic forecast of the excess supply.

Consider the forecast of excess supply for a given time, region and expected conditions. The DBBN model provides an estimated probability distribution for the error in the mean forecast of regional excess supply $Error(RXS)$. The system operator needs to know an upper confidence limit on the random variable $Error(RXS)$, so that they can ensure that there is an adequate power reserve. For a given confidence level α (e.g., $\alpha = 95\%$), we define FUM as follows:

$$FUM : P(Error(RXS) \leq FUM) = \alpha, \tag{6.9}$$

Thus FUM provides an upper bound on the forecast $Error(RXS)$ with confidence level α. Intuitively, the higher the required confidence α in the prediction error, the higher FUM needs to be. In this way, we can convert the output of the DBBN into a meaningful concept that decision makers can use, i.e., a measure of the $Error(RXS)$, in MW, of the forecasting error distribution at a given time in the future.

The confidence level α for determining FUM should be set at a level such that the system operator reasonably expects to achieve an appropriate balance between: (i) reducing the chance of a situation where load shedding arises as a result of reserve forecasting error, and (ii) increasing the likelihood of unnecessary declarations due to an overly conservative confidence level. The confidence level should also decrease monotonically with

increasing forecasting horizon, and be consistent across regions for the same forecasting horizon.

Based on our proposal for FUM, we now introduce a new formulation for calculating a revised LOR threshold to assess the probability of a shortfall in available reserve capacity that could lead to load shedding in each region:

$$LOR1 = max(LCC_2, FUM) \tag{6.10a}$$

$$LOR2 = max(LCC_1, FUM) \tag{6.10b}$$

where LCC_1 is the size in MW of the single Largest Credible Contingency event (LCC), and LCC_2 is the size in MW of the two largest credible contingencies, assuming they occur consecutively with sufficient time to return the power system to a secure operating state prior to the second event. In Figure 6.4 we show how the proposed LOR thresholds are able to effectively reflect the impact of extreme FUM values.

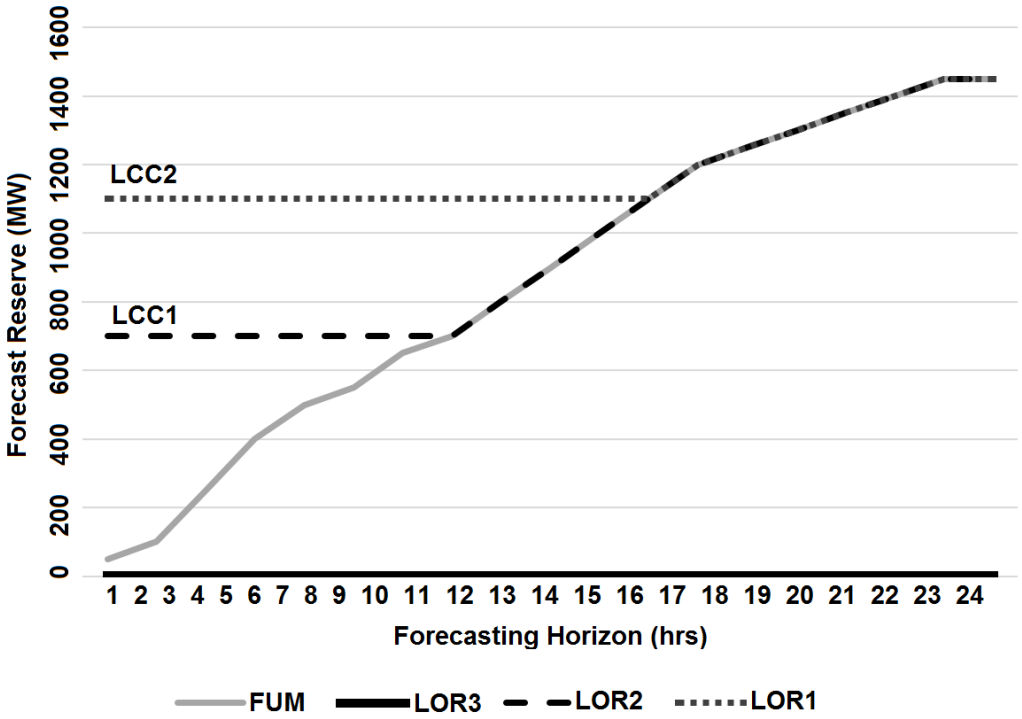


Figure 6.4: LOR notices in the national electricity market in Australia, using the revised formulation of LOR based on FUM

In summary, the prevailing conditions at the time of each forecast are provided as input to the DBBN network, so that the DBBN's inferred probability distribution of the

Table 6.2: FUM values for the horizon of 6hrs and 24hrs ahead in Victoria for a range of scenarios.

Scenarios		FUM (MW)	
Temperature($^\circ C$)	$Wind(MW)$	$6hrs$	$24hrs$
> 31	> 800	1750	1192
> 31	< 200	1750	953
< 13	> 800	845	993
< 13	< 200	667	795

$Error(RXS)$ of the forecast can be used to generate the Forecasting Uncertainty Measure (FUM) in MW for each 30 minute trading interval. Note that the FUM value is associated with a predetermined confidence level. The impact that the prevailing conditions have on each of the error distributions of each forecasting system is automatically accounted for within the trained DBBN, and only the final result is reported. As an example, Table 6.2 provides the likely FUM values for a range of different scenarios for a major region in Australia.

6.4 Case Study Applications

This study is intended to: (i) Describe how a power grid operator continually assesses the probability of capacity reserves being insufficient to avoid load shedding; (ii) Describe how the probability assessment applies in relation to different periods of time; (iii) Specify at least three probability levels at which the power grid operator will declare a corresponding lack-of-reserve condition in relation to a specified period of time, indicating an increased probability of load shedding (other than the reduction or disconnection of interruptible load).

6.4.1 Input Data

The Australian Energy Market operator (AEMO), is responsible for operating the power grid for the National Electricity Market (NEM), which serves the eastern and south-eastern regions of Australia, as well as the Western Australian South West Integrated System. The performance of our proposed method has been validated with real data

provided by AEMO, which has also implemented our method over the five states of the NEM, namely, Tasmania, Victoria, New South Wales, South Australia, and Queensland. Seven years of historical data (Actual and forecast data for demand and different types of supply) with a granularity of 30 minutes has been used for training the DBBN model for each of these regions using the NETICA software, which is a powerful, easy-to-use, and comprehensive program for working with belief networks and influence diagrams [4]. The experiments are run on an AZURE cloud using $\sim$ 20 CPUs and 128 GB RAM.

The input data into the model can be divided to ten types of data as follows:

1. Forecasts (at multiple horizons) and current observations of weather data, which includes precipitation, wind speed, temperature, cloud cover, relative humidity and solar radiation, as well as the temperature forecast differential between different forecast providers at multiple horizons.

2. Current generation level across each generation type: gas, coal, hydro, battery, solar farms, solar roof tops, and wind farms.

3. Forecasts of RES generators at multiple time horizons.

4. Demand forecast error data: Demand forecast error for the most recent complete trading interval, comparing the forecast generated one interval prior to the trading interval to the actual. This captures recent demand forecast errors and the sensitivity analysis indicates the importance of this input for the first 6 hrs of the horizon.

5. Regional reference price $(\$/MWh)$.

6. Prior trading interval forecast of Reserve Error.

7. Forecasts of reserve levels at multiple horizons.

8. Forecasts of inter-connector flow at multiple horizons.

9. Time of day (daytime / night-time forecast): For example, a 6 hour ahead forecast that was run at $22:00$ for $04:00$ (overnight forecast) would have less uncertainty associated with the error than a forecast run at $10:00$ for $16:00$ (daily peak forecast). These prevailing conditions/input data are used as predictor variables in the model. The aim is to train the model in order to forecast the target variable

($Error(RXS)$ in MW) for each of the next 144 trading intervals using the predictor variables. We used seven years history of forecast and actual data (all generation types and demand) in each region in Australia to produce the $Error(RXS)$ distribution for each of the next 144 trading intervals. Then the model is trained using the predictor variables and target variable ($Error(RXS)$).

6.4.2 Pre-processing, Training, and Testing Model

The aim is to forecast the $Error(RXS)$ in MW for different time horizons using the input data as the predictor variables. In the pre-processing step, we need to clean the data by imputing any missing data and removing any outliers. We processed a huge volume of data (approximately one billion records) to generate the error distribution. Specifically, every 30 minutes we receive a new data record and we have 144 time horizons for forecast, trained on historical values of the different input variables since 2011. This data was cleaned and pivoted into a form that can be fed into the DBBN. Overall, data preprocessing and cleaning required two months of effort, while only two minutes is needed to train the model.

The next step after data cleaning is the sensitivity analysis. A sensitivity analysis of each of the proposed predictors was conducted to determine which inputs lead to the largest shifts in the distribution of RXS for each region. The results from the sensitivity analysis for each region were used to guide the selection of a consistent set of input predictors with consideration of the maximum number of nodes that can be compiled in a DBBN (too many nodes cause the network to fail to converge). For example, solar radiation forecast inputs indicate to the model the expected solar irradiance influencing rooftop PV generation and generation from large scale solar farms. This input is prioritised given the current penetration and continued growth of rooftop PV, and the predicted ramp up of large scale solar generation. So, in the sensitivity analysis step we identify the input variables that have the most impact on the target variable ($Error(RXS)$). Table 6.3 shows the top five most important input variables that are detected by the DBBN for NSW region and for different time horizons of the target variable ($Error(RXS)$). As shown in Table 6.3, the amount of coal generation and demand error at the run time have the most impact on the short-term forecast error (6 hours ahead). In general, using the

Table 6.3: The order of input variable importance for each time horizon of target variable ($Error(RXS)$)

$Error(RXS)_{6hr}$	$Error(RXS)_{24hr}$	$Error(RXS)_{48hr}$
coal gen	solar radiation	solar radiation
demand error	relative humidity	relative humidity
solar gen	temperature	solar gen
solar radiation	wind speed	wind speed
wind speed	solar gen	temperature

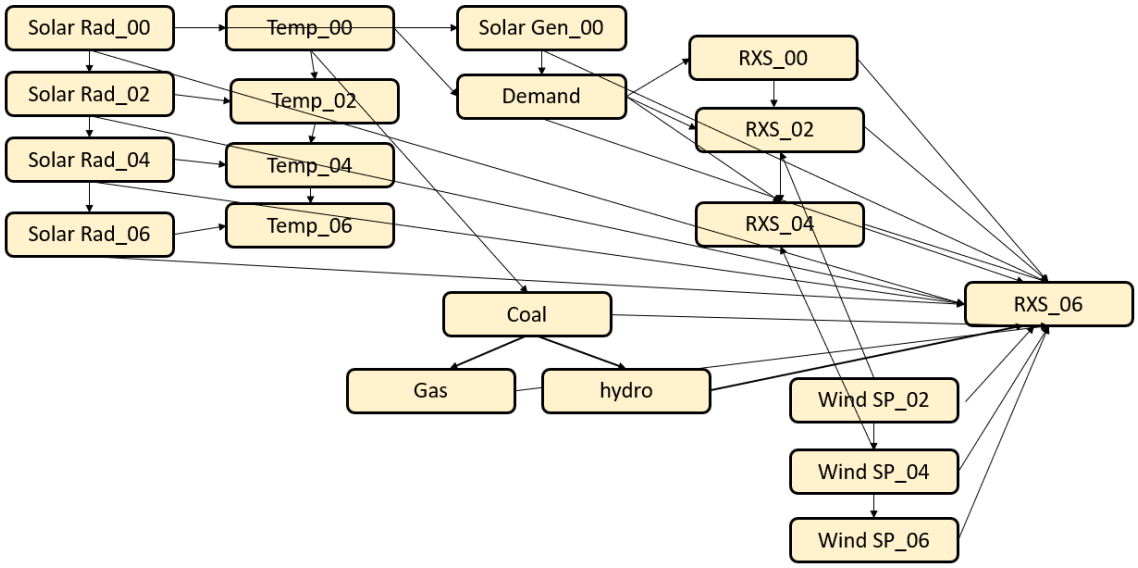


Figure 6.5: Architecture of forecasting model for NSW for 6 hours ahead

generation type in the model is very important for short-term risk assessment. For example, if you have a large availability of wind relative to gas, you should expect the larger error than if you have a large availability of gas relative to wind. It also concludes that if there is low demand then you should not expect a large error because you need less generation capacity in the network. For 24 hours and 48 hours ahead, solar radiation can have a large influence on error, which is expected because Australia is experiencing rapid growth rooftop solar, and solar radiance influences generation from rooftop PV and large scale solar farms. This input is prioritized given the current penetration and continued growth of rooftop PV, and the predicted ramp up of large scale solar generation.

The variables selected in the sensitivity analysis step are used as inputs to train the

DBBN model. In fact in the training step, by feeding a number of sequences of observations to the model, we try to estimate the parameters of the DBBN model such that they best fit to the observed data and provide the best model for the system. In Figure 6.5 the structure of the training model for 3 hours ahead forecast horizon is shown. The model was trained using the TAN (Tree Augmented Naive) Bayes method [70] to construct the appropriate link structure between nodes. Netica has two options for defining the link structure between nodes: manually and automatically based on the TAN method. In this study because of the wide variety of input variables to the model, we used the automatic option of the Netica software for defining the link structure.

After training the model for different target time horizons, the trained model was saved for use in the test step. In the test step we apply the saved trained model for the period of interest and use it to predict the target variable (Error(RXS)).

6.4.3 Dynamic Forecasting Uncertainty Measure

In this study we used 7 years historical data of forecast and actual measured demand, RES generators supply and conventional generators supply over all five regions in Australia for every 30 minute trading interval since January 2010 to July 2017. Then, by using our DBBN model we produced the probabilistic forecast value of RXS for the next 384 trading intervals up to 8 days ahead. Each 30 minute forecast is assessed against the actuals for each of the next 144 trading intervals. For example, a forecast run at 01-01-2017 01:00 would have forecasts for each 30 minutes interval from 01-01-2017 01:30 to 04-01-2017 01:00 and an $Error(RXS)$ distribution can be created for each of these forecasts.

In Figure 6.6 the top graph shows the error distribution associated with the 12 hours ahead forecast of intermittent generators, conventional generators and load demand in all five regions in the NEM. In absolute terms the RES generation has a small materiality but high frequency of error. The distribution of the error associated with the demand forecast has a lower frequency, but higher materiality in MW, and is symmetrically distributed. The errors in conventional generation availability was biased towards an over-estimate of the availability by generators for a day-ahead. These generators provide information about their dispatch for the next 24 hours based on the weather forecast and in real time they usually reduce the amount in generation that they had expected because

of the hot weather they cannot work as efficiently.

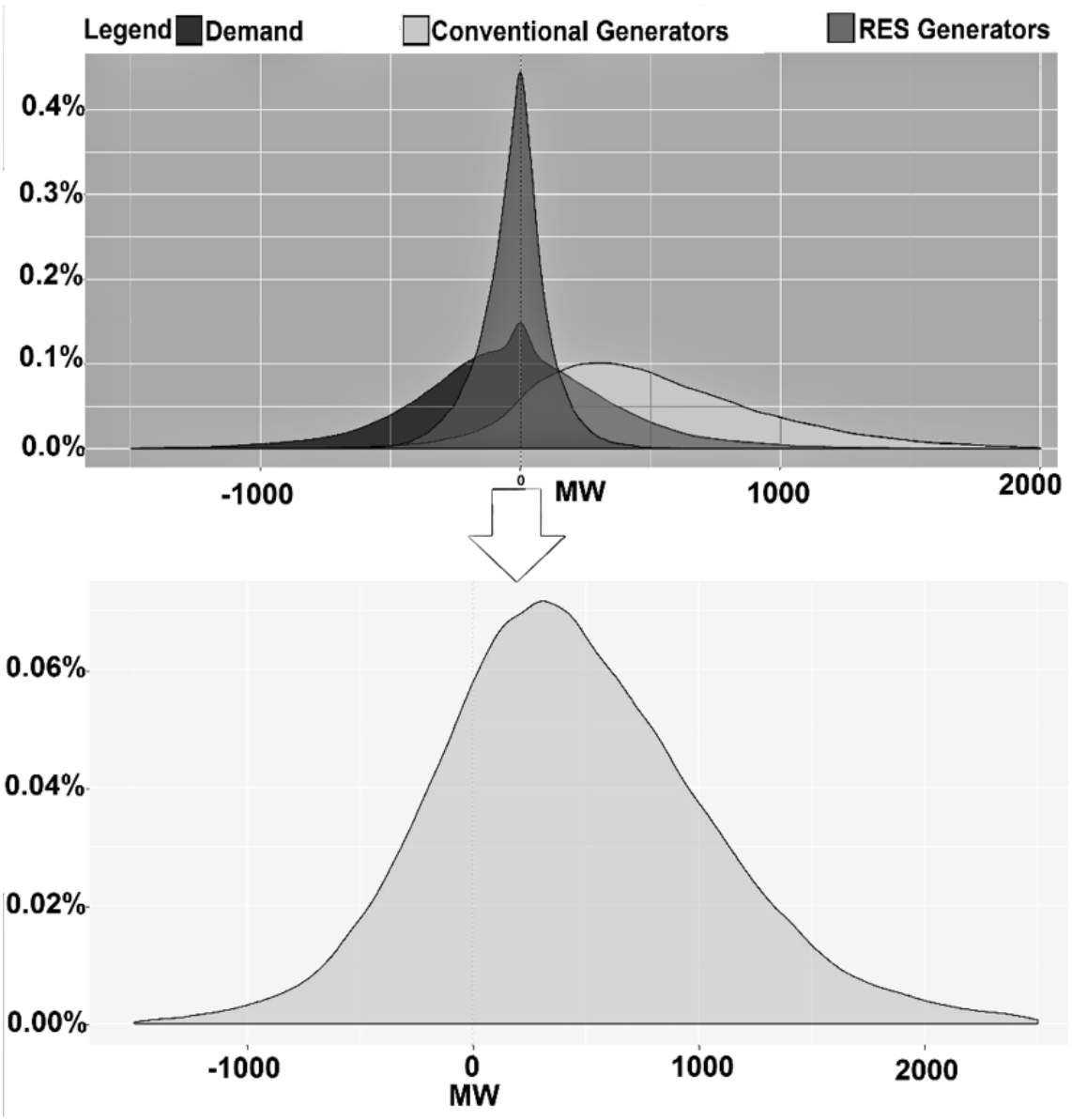


Figure 6.6: Top graph shows the distribution in MW of forecasting error across all the regions in Australia including the error produced by intermittent generation forecast, conventional generators maximum availability forecast, and regional demand forecast for the 12hrs ahead forecasting error. Bottom graph shows the joint distribution of the top distributions extracted from the DBBN for 12hrs ahead forecasting error

The bottom graph in Figure 6.6 shows the joint distribution of imbalances, $Error(RXS)$, which is inferred by a DBBN learned on the inputs from the top figure.

The graphs in Figure 6.7 show the error distributions for the demand forecasts, intermittent generation forecasts and regional aggregate availability of conventional gen-

eration over the seven years period ending July 2017 for errors in the range $-800MW$ to $+800MW$ for different time intervals of forecasting. In these graphs a positive sign means that the forecast value is greater than the actual value. Based on the inferred $Error(RXS)$

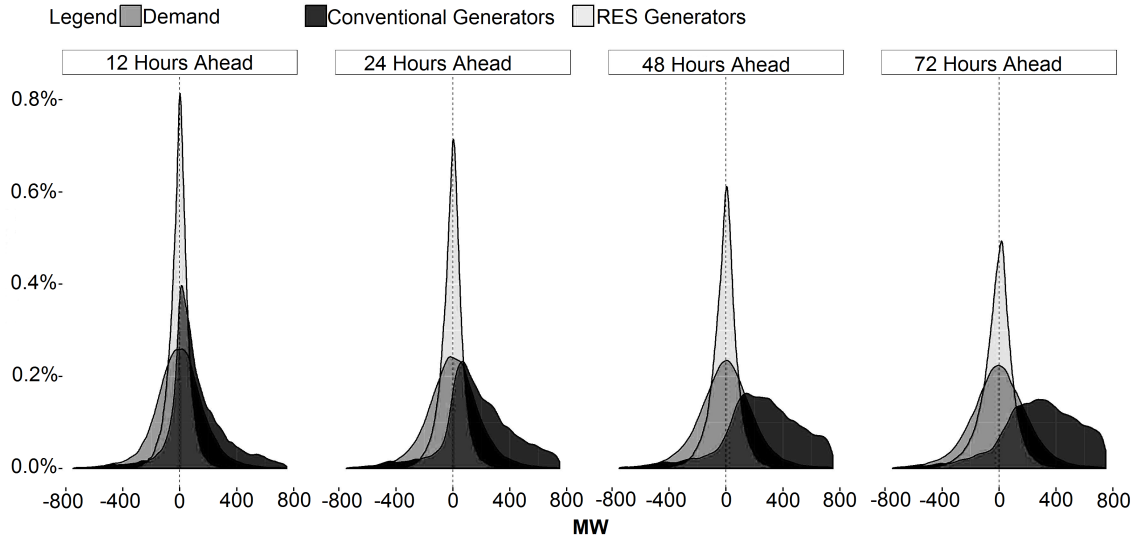


Figure 6.7: Typical error distributions for a range of forecasting horizons in Victoria using almost 7 years of historical data (January 2010-July 2017)

distribution, the FUM value (in MW) can then be calculated. In the dynamic version of FUM the error distributions are updated by using a DBBN based on the current prevailing conditions when the forecast is produced. The difference between the dynamic and static version of FUM is shown in Figure 6.8. In this figure the Dynamic FUM shows the amount of FUM produced by proposed DBBN method, the static FUM is produced by deterministic method and the dotted line shows the Actual FUM in the system at the time of the run. Each graph shows the static, dynamic, and actual value of FUM at consecutive time intervals of every 30 minutes. As shown the dynamic FUM is able to reflect the actual FUM in the system much more accurately in comparison to static FUM (deterministic method).

It is worth noting that any changes to forecasting systems that result in a change in any of the error distributions (for example, an upgrade to the forecasting system resulting in an improvement in forecasting accuracy) can be reflected in the DBBN and subsequent FUM values.

The output of the DBBN is a measure of the $Error(RXS)$ in MW for each of the next

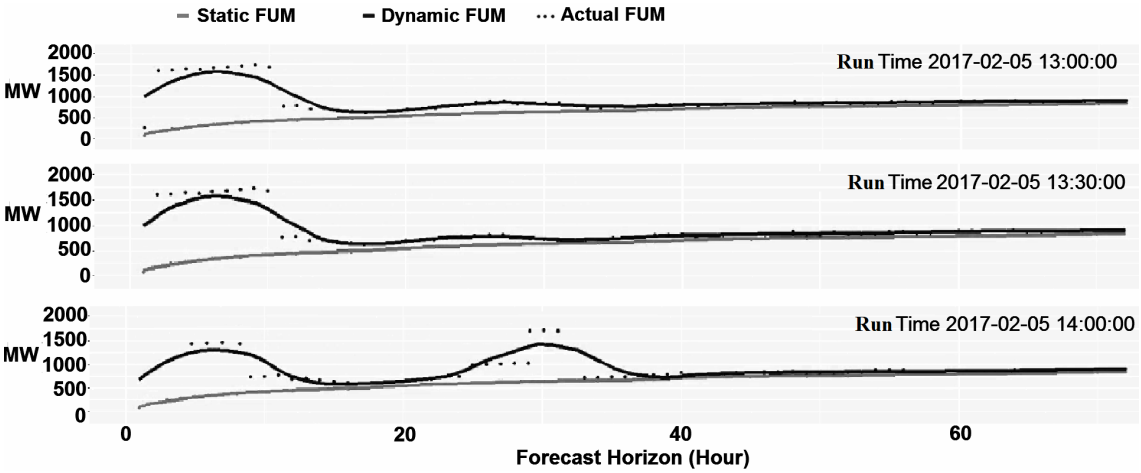


Figure 6.8: Comparison of Actual FUM value with the Dynamic FUM (produced using DBBN) and the Static FUM (produced using deterministic method) in three consecutive run times in New South Wales, Australia.

144 trading intervals. As we are dealing with a forecasting error distribution, the "At Risk" *MW* of *FUM* is associated with a confidence level. The DBBN outputs a *MW* value for each of the forecasts, associated with the fixed confidence level.

6.4.4 Selection of Confidence Level

The final step of our analysis is selecting the *MW* values associated with the required confidence level. In order to determine the associated *MW* value for the given confidence level, all the probabilities are accumulated until they exceed the specified confidence level, and the associated *MW* value in that bin becomes the *FUM* value. The appropriate confidence level used for the forecast uncertainty measure, *FUM*, used in determining *LOR* conditions needs to achieve a balance between reducing the chance of a situation where load shedding arises due to lack of action by system operators as a result of a reserve forecasting error, and increasing the likelihood of unnecessary declarations due to an overly conservative confidence level. The confidence levels are selected based on a sensitivity analysis of the number of *LOR* declarations to changes in the confidence level. The proposed confidence levels are specified in Table 6.4. In Table 6.5 the number of LOR conditions declared due to FUM (using the proposed dynamic method) and LOR conditions declared due to LCC1 (using the deterministic method) are shown during the test period of July 2016 to July 2017. In all case studies, FUM could declare the LOR2

conditions one day earlier than the deterministic method (LCC1), which gives more time to the power grid operator to maintain the balance between demand and supply.

Table 6.4: Confidence levels for determination of FUM values

Region(s)	Forecasting Horizon (Hrs)	Confidence level
All	0.5 to 15	98%
All	15.5 to 18	97%
All	18.5 to 21	96%
All	21.5 to 72	95%

Table 6.5: The number of LOR2 conditions declared by FUM and LCC1 criteria. Note that for all regions the FUM declared by LOR2 was one day earlier than LCC1

Region	LOR2s declared by FUM	LOR2s declared by LCC1
New South Wales	14	3
Queensland	4	1
South Australia	23	1
Tasmania	6	3
Victoria	16	3

6.5 Conclusion

In this chapter we have introduced a novel data-driven artificial intelligence-based methodology for dynamic probabilistic reserve sizing. The proposed methodology makes use of Dynamic Bayesian Belief Networks as a simple graphical model and can incorporate any uncertainty associated with power system operation. We also introduced a novel metric to measure uncertainties, in *MW*, to incorporate in the reliability level assessment. This new proposed method enables grid operators to assess the size of the reserve in a more flexible way, allowing decision makers to move from typical existing contingency-based deterministic approach to dynamic probabilistic assessment that also accounts for all relevant risk factors that could affect reserve levels. Hence, the risk of involuntary load shedding can be forecast more accurately, and the analysis is performed with clear, intuitive

graphical support. In the case studies, we have run extensive analysis on the prevailing conditions on the error distribution produced by key uncertainty factors in the five states of the Australian NEM using 7 years of forecast and actual data of wind, solar and conventional power generation, demand, weather, and market prices. The methodology is currently being used by AEMO for their risk assessment and reserve sizing.

Chapter 7
Conclusions and Future Work

7.1 Conclusions

In this thesis we have addressed several significant challenges that arise in smart grids in terms of managing and analysing complex high volume data for forecasting, and managing the uncertainties in the grid in the presence of renewable energy sources. We proposed several advanced analytical models and decision support tools to provide actionable insights for tasks such as short-term load forecasting in smart grids.

In Chapter 3, a novel approach for hierarchical load forecasting was presented. We leveraged the K-Shape clustering algorithm, which is a kind of shape based clustering algorithms, for clustering customers. Customers are clustered based on the shape of their daily load profile. Then we can produce short-term load forecasts at each cluster level using Deep Belief Networks (DBN). In addition, we introduced a new weighted summation method for accumulating the forecasting results of each cluster according to the size of their membership. We used the case study of real-life publicly available data to evaluate our proposed methodology. Our results showed that careful clustering of customers based on our K-Shape clustering method results in smaller forecasting errors than using traditional k-means clustering in this context.

In Chapter 4, we introduced two new time-series shape based clustering algorithms namely, "Fuzzy C-Shape plus (FCS+)" and "Fuzzy C-Shape plus plus (FCS++)", and showed that these two new algorithms outperform state-of-the-art shape-based clustering algorithms in terms of accuracy and efficiency. We generalized the fuzzy c-means (FCM) model to accommodate clustering in finite time-series data by two approaches: (i) selecting a statistical measure, cross-correlation, as a shape similarity distance mea-

sure for comparing time-series sequences, and (ii) selecting a new centroid computation technique based on a shape extraction method when computing the clusters. We performed extensive experiments over 48 different time-series datasets and demonstrated the superiority of our technique against existing schemes based on four external cluster validity indices (the Rand index, Adjusted Rand Index, Variation of Information, and Normalized Mutual Information) and two statistical tests (Wilcoxon and Friedman).

In Chapter 5, we presented a robust and accurate short-term load forecasting framework based on an unsupervised ensemble learning approach, using our proposed clustering algorithm in Chapter 4 combined with horizontal ensemble learning. In addition, we presented a novel factor, apparent temperature, as an important input variable into the model. The proposed framework addressed some of the challenges in the load forecasting problem including unbalanced training data, the non-stationary nature of the load data, and feature selection for predictive modeling. The performance of the proposed framework was validated with real data from the power system in the Australian National Electricity Market as well as through on-site implementation by the system operator. It was shown that our predictive framework has a substantially higher accuracy over existing methods in both normal and abnormal situations.

In Chapter 6, we introduced a novel data-driven methodology for dynamic probabilistic reserve sizing. The proposed methodology makes use of Dynamic Bayesian Belief Networks as a simple graphical model and can incorporate a range of prior knowledge about uncertainties associated with power system operation. We also introduced a novel metric to measure uncertainties, in MW, that can be incorporated in the reliability level assessment. This new proposed method enables grid operators to assess the size of the reserve in a more flexible way, allowing decision makers to move from typical existing contingency-based deterministic approaches to dynamic probabilistic assessment that also accounts for relevant risk factors that could affect reserve levels. Hence, the risk of involuntary load shedding can be forecast more accurately, and the analysis can be performed with clear, intuitive graphical support. In the case studies, we run extensive analysis on the prevailing conditions on the error distribution produced by key uncertainty factors in the five states of the Australian National Energy Market using seven years of forecast and actual data of wind, solar and conventional power generation, de-

mand, weather, and market prices. The methodology is currently being used by Australian Energy Market operator for their risk assessment and reserve sizing.

7.2 Future Work

This thesis highlights several interesting directions for future research:

- Renewable energy sources are weather dependent. In utility-scale, wind and solar farms, the output of these farms can be metered individually. Therefore, we can provide forecasts for each farm based on their historical output data. As a result, the net demand can be calculated by subtracting the load forecast from the renewable generation forecasts. However, nowadays, with high penetration of rooftop solar in the grid, there is a need to forecast the amount of energy that is generated by these unmetered energy sources. Since rooftop solar is behind the meter, this generation capacity is not directly monitored and producing forecasts for them is a challenging task. Accurate forecasting of these unmetered sources is an important focus for future research.

- Our methods for clustering are carefully designed for batch operation. However, the customer metering data arrives as a continuous stream of measurements, and the customers' behaviour evolves over time. So, there is a need to produce incremental or streaming clustering algorithms that are suitable for load profiling in this on-line setting.

- In Chapter 6, future work is needed to incorporate the proposed reserve sizing method into a full unit commitment model. There are other forecasting uncertainties that contribute to the level of reserve forecasting uncertainty. These relate to: (i) Network limitations, both inter-regional and intra-regional; (ii) Supply-demand balance in neighbouring regions; (iii) Energy limitations on scheduled generating units. Analysis indicates that each of these factors can, at times, contribute to significant levels of uncertainty in the reserve forecasting process, and without accounting for them the current RXS definition may be overly conservative. For example, in circumstances where a generating unit in a region shares network capacity with

an inter-connector, then an un-forecast reduction in the availability of this generator unit could result in an un-forecast increase in inter-connector support. Under the current RXS definition this scenario is not considered and may result in an overly conservative distribution of RXS errors. A challenge for future research is to redefine the RXS to accommodate this scenario.

Bibliography

[1] Aemo market notifications. [Online]. Available: https://www.aemo.com.au/Media-Centre/AEMO-market-notifications-explained

[2] "Australian bureau of meteorology weather stations." [Online]. Available: http://www.bom.gov.au/climate/cdo/about/sites.shtml

[3] "Georgia institute of technology news center." [Online]. Available: https://www.news.gatech.edu/features/building-power-grid-future

[4] Norsys software corp. [Online]. Available: https://norsys.com

[5] "Smart metering trial data publication," November 2013. [Online]. Available: http://www.cer.ie/en/information-centre-reportsand-publications.aspx?article=5dd4bce4-ebd8-475e-b78d-da24e4ff7339

[6] "Australian energy market operator," November 2017. [Online]. Available: https://www.aemo.com.au/Media-Centre/AEMO-releases-summer-readiness-report-for-2017-18

[7] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering–a decade review," *Information Systems*, vol. 53, pp. 16–38, 2015.

[8] R. Agrawal, C. Faloutsos, and A. Swami, "Efficient similarity search in sequence databases," in *International Conference on Foundations of Data Organization and Algorithms*. Springer, 1993, pp. 69–84.

[9] O. Y. Al-Jarrah, Y. Al-Hammadi, P. D. Yoo, and S. Muhaidat, "Multi-layered clustering for power consumption profiling in smart grids," *IEEE Access*, vol. 5, pp. 18 459–18 468, 2017.

[10] R. Al-Otaibi, N. Jin, T. Wilcox, and P. Flach, "Feature construction and calibration for clustering daily load curves from smart-meter data," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 2, pp. 645–654, 2016.

[11] R. Allan, B. Borkowska, and C. Grigg, "Probabilistic analysis of power flows," in *Proceedings of the Institution of Electrical Engineers*, vol. 121, no. 12. IET, 1974, pp. 1551–1556.

[12] R. Allan, A. L. Da Silva, and R. Burchett, "Evaluation methods and accuracy in probabilistic load flow solutions," *IEEE Transactions on Power Apparatus and Systems*, no. 5, pp. 2539–2546, 1981.

[13] D. T. Anderson, J. C. Bezdek, M. Popescu, and J. M. Keller, "Comparing fuzzy, probabilistic, and possibilistic partitions," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 5, pp. 906–918, 2010.

[14] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. PéRez, and I. Perona, "An extensive comparative study of cluster validity indices," *Pattern Recognition*, vol. 46, no. 1, pp. 243–256, 2013.

[15] W. G. Aref, M. G. Elfeky, and A. K. Elmagarmid, "Incremental, online, and merge mining of partial periodic patterns in time-series databases," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 3, pp. 332–342, 2004.

[16] R. Avnimelech and N. Intrator, "Boosted mixture of experts: an ensemble learning scheme," *Neural computation*, vol. 11, no. 2, pp. 483–497, 1999.

[17] S. Bandyopadhyay, T. Ganu, H. Khadilkar, and V. Arya, "Individual and aggregate electrical load forecasting: One for all and all for one," in *Proceedings of the 2015 ACM Sixth International Conference on Future Energy Systems*. ACM, 2015, pp. 121–130.

[18] A. Banerjee and J. Ghosh, "Clickstream clustering using weighted longest common subsequences," in *Proceedings of the web mining workshop at the 1st SIAM conference on data mining*, vol. 143, 2001, p. 144.

[19] Y. Bengio, "Deep learning of representations: Looking forward," in *International Conference on Statistical Language and Speech Processing*. Springer, 2013, pp. 1–37.

[20] Y. Bengio and O. Delalleau, "Justifying and generalizing contrastive divergence," *Neural computation*, vol. 21, no. 6, pp. 1601–1621, 2009.

[21] Y. Bengio *et al.*, "Learning deep architectures for AI," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[22] Y. Bengio, P. Simard, P. Frasconi *et al.*, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

[23] I. Benítez, A. Quijano, J.-L. Díez, and I. Delgado, "Dynamic clustering segmentation applied to load profiles of energy consumption from spanish customers," *International Journal of Electrical Power & Energy Systems*, vol. 55, pp. 437–448, 2014.

[24] R. J. Bessa, V. Miranda, A. Botterud, Z. Zhou, and J. Wang, "Time-adaptive quantile-copula for wind power probabilistic forecasting," *Renewable Energy*, vol. 40, no. 1, pp. 29–39, 2012.

[25] H. Bevrani, F. Daneshfar, and T. Hiyama, "A new intelligent agent-based agc design with real-time application," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 994–1002, 2012.

[26] J. C. Bezdek, "Fuzzy mathematics in pattern classification," *Ph. D. Dissertation, Applied Mathematics, Cornell University*, 1973.

[27] ——, "A convergence theorem for the fuzzy isodata clustering algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 1, pp. 1–8, 1980.

[28] ——, *Pattern recognition with fuzzy objective function algorithms*. Springer Science & Business Media, 2013.

[29] R. Billinton and D. Huang, "Effects of load forecast uncertainty on bulk electric system reliability evaluation," *IEEE Transactions on Power Systems*, vol. 23, no. 2, pp. 418–425, 2008.

[30] L. Bobrowski and J. C. Bezdek, "C-means clustering with the l/sub l/and l/sub infinity/norms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 3, pp. 545–554, 1991.

[31] T. Boneh, G. T. Weymouth, P. Newham, R. Potts, J. Bally, A. E. Nicholson, and K. B. Korb, "Fog forecasting for melbourne airport using a bayesian decision network," *Weather and Forecasting*, vol. 30, no. 5, pp. 1218–1233, 2015.

[32] B. Borkowska, "Probabilistic load flow," *IEEE Transactions on Power Apparatus and Systems*, no. 3, pp. 752–759, 1974.

[33] M. Borunda, O. Jaramillo, A. Reyes, and P. H. Ibargüengoytia, "Bayesian networks in renewable energy systems: A bibliographical survey," *Renewable and Sustainable Energy Reviews*, vol. 62, pp. 32–45, 2016.

[34] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[35] L. Breiman, *Classification and regression trees*. Routledge, 2017.

[36] P. J. Brockwell, R. A. Davis, and M. V. Calder, *Introduction to time series and forecasting*. Springer, 2002, vol. 2.

[37] K. Bruninx and E. Delarue, "A statistical description of the error on wind power forecasts for probabilistic reserve sizing," *IEEE Transactions on Sustainable Energy*, vol. 5, no. 3, pp. 995–1002, 2014.

[38] M. Bucksteeg, L. Niesen, and C. Weber, "Impacts of dynamic probabilistic reserve sizing techniques on reserve requirements and system costs," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 4, pp. 1408–1420, 2016.

[39] H.-Â. Cao, C. Beckel, and T. Staake, "Are domestic load profiles stable over time? an attempt to identify target households for demand side management campaigns," in *IECON 2013-39th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2013, pp. 4733–4738.

[40] C. Cecati, C. Citro, and P. Siano, "Combined operations of renewable energy systems and responsive demand in a smart grid," *IEEE Transactions on Sustainable Energy*, vol. 2, no. 4, pp. 468–476, 2011.

[41] N. Charlton and C. Singleton, "A refined parametric model for short term load forecasting," *International Journal of Forecasting*, vol. 30, no. 2, pp. 364–368, 2014.

[42] E. Charniak, "Bayesian networks without tears." *AI magazine*, vol. 12, no. 4, p. 50, 1991.

[43] C. Chelmis, J. Kolte, and V. K. Prasanna, "Big data analytics for demand response: Clustering over space and time," in *2015 IEEE International Conference on Big Data (Big Data)*. IEEE, 2015, pp. 2223–2232.

[44] B.-J. Chen, M.-W. Chang *et al.*, "Load forecasting using support vector machines: A study on eunite competition 2001," *IEEE transactions on Power Systems*, vol. 19, no. 4, pp. 1821–1830, 2004.

[45] L. Chen and R. Ng, "On the marriage of lp-norms and edit distance," in *Proceedings of the Thirtieth International Conference on Very Large Data Bases-Volume 30*. VLDB Endowment, 2004, pp. 792–803.

[46] L. Chen, M. T. Özsu, and V. Oria, "Robust and fast similarity search for moving object trajectories," in *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*. ACM, 2005, pp. 491–502.

[47] P. Chen, Z. Chen, and B. Bak-Jensen, "Probabilistic load flow: A review," in *2008 Third International Conference on Electric Utility Deregulation and Restructuring and Power Technologies*. IEEE, 2008, pp. 1586–1591.

[48] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 785–794.

[49] G. Chicco, "Overview and performance assessment of the clustering methods for electrical load pattern grouping," *Energy*, vol. 42, no. 1, pp. 68–80, 2012.

[50] S. Chu, E. Keogh, D. Hart, and M. Pazzani, "Iterative deepening dynamic time warping for time series," in *Proceedings of the 2002 SIAM International Conference on Data Mining*. SIAM, 2002, pp. 195–212.

[51] J. T. Connor, R. D. Martin, and L. E. Atlas, "Recurrent neural networks and robust time series prediction," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 240–254, 1994.

[52] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[53] P. G. Da Silva, D. Ilić, and S. Karnouskos, "The impact of smart grid prosumer grouping on forecasting accuracy and its benefits for local electricity market trading," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 402–410, 2014.

[54] R. N. Dave and K. Bhaswan, "Adaptive fuzzy c-shells clustering and detection of ellipses," *IEEE Transactions on Neural Networks*, vol. 3, no. 5, pp. 643–662, 1992.

[55] K. Dietrich, J. Latorre, L. Olmos, A. Ramos, and I. Perez-Arriaga, "Stochastic unit commitment considering uncertain wind production in an isolated system," in *4th Conference on Energy Economics and Technology*, 2009, pp. 1–6.

[56] C. Ding, X. He, H. Zha, and H. D. Simon, "Adaptive dimension reduction for clustering high dimensional data," in *2002 IEEE International Conference on Data Mining, 2002. Proceedings.* IEEE, 2002, pp. 147–154.

[57] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh, "Querying and mining of time series data: experimental comparison of representations and distance measures," *Proceedings of the VLDB Endowment*, vol. 1, no. 2, pp. 1542–1552, 2008.

[58] N. Ding, Y. Bésanger, F. Wurtz, G. Antoine, and P. Deschamps, "Time series method for short-term load forecasting using smart metering in distribution systems," in *2011 IEEE Trondheim PowerTech*. IEEE, 2011, pp. 1–6.

[59] A. P. Douglas, A. M. Breipohl, F. N. Lee, and R. Adapa, "Risk due to load forecast uncertainty in short term power system planning," *IEEE Transactions on Power Systems*, vol. 13, no. 4, pp. 1493–1499, 1998.

[60] I. Drezga and S. Rahman, "Input variable selection for ann-based short-term load forecasting," *IEEE Transactions on Power Systems*, vol. 13, no. 4, pp. 1238–1244, 1998.

[61] R. O. Duda, P. E. Hart *et al.*, *Pattern classification and scene analysis*. Wiley New York, 1973, vol. 3.

[62] Y. Dvorkin, M. A. Ortega-Vazuqez, and D. S. Kirschen, "Wind generation as a reserve provider," *IET Generation, Transmission & Distribution*, vol. 9, no. 8, pp. 779–787, 2015.

[63] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos, *Fast subsequence matching in time-series databases*. ACM, 1994, vol. 23, no. 2.

[64] S. Fan and R. J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model," *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 134–141, 2012.

[65] S. Fan, K. Methaprayoon, and W.-J. Lee, "Multiregion load forecasting for system with large geographical area," *IEEE Transactions on Industry Applications*, vol. 45, no. 4, pp. 1452–1459, 2009.

[66] A. Friedman, M. D. Keselman, L. G. Gibb, and A. M. Graybiel, "A multistage mathematical approach to automated clustering of high-dimensional noisy data," *Proceedings of the National Academy of Sciences*, vol. 112, no. 14, pp. 4477–4482, 2015.

[67] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[68] ——, "Stochastic gradient boosting," *Computational statistics & data analysis*, vol. 38, no. 4, pp. 367–378, 2002.

[69] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the american statistical association*, vol. 32, no. 200, pp. 675–701, 1937.

[70] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine learning*, vol. 29, no. 2-3, pp. 131–163, 1997.

[71] N. Friedman, K. Murphy, and S. Russell, "Learning the structure of dynamic probabilistic networks," in *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1998, pp. 139–147.

[72] M. Gavrilov, D. Anguelov, P. Indyk, and R. Motwani, "Mining the stock market: Which measure is best," in *Proc. of the 6th ACM SIGKDD*, 2000, pp. 487–496.

[73] C. W. Gellings, *The smart grid: enabling energy efficiency and demand response*. The Fairmont Press, Inc., 2009.

[74] F. A. Gers and E. Schmidhuber, "LSTM recurrent networks learn simple context-free and context-sensitive languages," *IEEE Transactions on Neural Networks*, vol. 12, no. 6, pp. 1333–1340, 2001.

[75] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," 1999.

[76] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber, "Learning precise timing with LSTM recurrent networks," *Journal of machine learning research*, vol. 3, no. Aug, pp. 115–143, 2002.

[77] R. Giusti and G. E. Batista, "An empirical comparison of dissimilarity measures for time series classification," in *2013 Brazilian Conference on Intelligent Systems*. IEEE, 2013, pp. 82–88.

[78] X. Golay, S. Kollias, G. Stoll, D. Meier, A. Valavanis, and P. Boesiger, "A new correlation-based fuzzy logic clustering algorithm for fmri," *Magnetic Resonance in Medicine*, vol. 40, no. 2, pp. 249–260, 1998.

[79] H. Gooi, D. Mendes, K. Bell, and D. Kirschen, "Optimal scheduling of spinning reserve," *IEEE Transactions on Power Systems*, vol. 14, no. 4, pp. 1485–1492, 1999.

[80] Y. Goude, R. Nedellec, and N. Kong, "Local short and middle term electricity load forecasting with semi-parametric additive models," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 440–446, 2014.

[81] R. Granell, C. J. Axon, and D. C. Wallom, "Impacts of raw data temporal resolution using selected clustering methods on residential electricity load profiles," *IEEE Transactions on Power Systems*, vol. 30, no. 6, pp. 3217–3224, 2015.

[82] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "A survey on smart grid potential applications and communication requirements," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 28–42, 2012.

[83] F. Günther and S. Fritsch, "neuralnet: Training of neural networks," *The R journal*, vol. 2, no. 1, pp. 30–38, 2010.

[84] S. Haben, C. Singleton, and P. Grindrod, "Analysis and clustering of residential customers energy behavioral demand using smart meter data," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 136–144, 2016.

[85] M. T. Hagan and S. M. Behr, "The time series approach to short term load forecasting," *IEEE Transactions on Power Systems*, vol. 2, no. 3, pp. 785–791, 1987.

[86] M. T. Hagan, H. B. Demuth, M. H. Beale, and O. De Jesús, *Neural network design*. Pws Pub. Boston, 1996, vol. 20.

[87] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *Journal of intelligent information systems*, vol. 17, no. 2-3, pp. 107–145, 2001.

[88] G. Hamoud, "Probabilistic assessment of interconnection assistance between power systems," *IEEE Transactions on Power Systems*, vol. 13, no. 2, pp. 535–542, 1998.

[89] R. J. Hathaway and J. C. Bezdek, "Switching regression models and fuzzy clustering," *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 3, pp. 195–204, 1993.

[90] G. E. Hinton, "Deep belief networks," *Scholarpedia*, vol. 4, no. 5, p. 5947, 2009.

[91] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[92] H. S. Hippert, C. E. Pedreira, and R. C. Souza, "Neural networks for short-term load forecasting: A review and evaluation," *IEEE Transactions on Power Systems*, vol. 16, no. 1, pp. 44–55, 2001.

[93] L. Hirth and I. Ziegenhagen, "Balancing power and variable renewables: Three links," *Renewable and Sustainable Energy Reviews*, vol. 50, pp. 1035–1051, 2015.

[94] B. F. Hobbs, S. Jitprapaikulsarn, S. Konda, V. Chankong, K. A. Loparo, and D. J. Maratukulam, "Analysis of the value for unit commitment of improved load forecasts," *IEEE Transactions on Power Systems*, vol. 14, no. 4, pp. 1342–1348, 1999.

[95] S. Hochreiter, Y. Bengio, P. Frasconi, J. Schmidhuber *et al.*, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," 2001.

[96] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[97] J. Hoffer and P. Dörfner, "Reliability and production cost calculation with peak load forecast uncertainty," *International Journal of Electrical Power & Energy Systems*, vol. 13, no. 4, pp. 223–229, 1991.

[98] J. Hoffer and M. Prill, "On the models of peak load forecast uncertainty in probabilistic production costing algorithms," *International Journal of Electrical Power & Energy Systems*, vol. 18, no. 3, pp. 153–160, 1996.

[99] H. Holttinen, M. Milligan, E. Ela, N. Menemenlis, J. Dobschinski, B. Rawn, R. J. Bessa, D. Flynn, E. Gomez-Lazaro, and N. K. Detlefsen, "Methodologies to determine operating reserves due to increased wind power," *IEEE Transactions on Sustainable Energy*, vol. 3, no. 4, pp. 713–723, 2012.

[100] H. Holttinen, M. Milligan, B. Kirby, T. Acker, V. Neimane, and T. Molinski, "Using standard deviation as a measure of increased operational reserve requirement for wind power," *Wind Engineering*, vol. 32, no. 4, pp. 355–377, 2008.

[101] R. Honda, S. Wang, T. Kikuchi, and O. Konishi, "Mining of moving objects from time-series images and its application to satellite weather imagery," *Journal of Intelligent Information Systems*, vol. 19, no. 1, pp. 79–93, 2002.

[102] T. Hong, "Energy forecasting: Past, present, and future," *Foresight: The International Journal of Applied Forecasting*, no. 32, pp. 43–48, 2014.

[103] ——, "Crystal ball lessons in predictive analytics," *EnergyBiz Mag*, vol. 12, no. 2, pp. 35–37, 2015.

[104] T. Hong *et al.*, "Long-term spatial load forecasting using human-machine co-construct intelligence framework," 2008.

[105] ——, "Short term electric load forecasting." 2010.

[106] T. Hong, P. Pinson, and S. Fan, "Global energy forecasting competition 2012," 2014.

[107] T. Hong and M. Shahidehpour, "Load forecasting case study," National association of regulatory utility commissioners, 2015. [Online]. Available: http://naruc.org/Grants/Documents/Load20Forecasting20Case20Study20_20UNCC.pdf

[108] T. Hong and P. Wang, "Fuzzy interaction regression for short term load forecasting," *Fuzzy optimization and decision making*, vol. 13, no. 1, pp. 91–103, 2014.

[109] T. Hong, P. Wang, and L. White, "Weather station selection for electric load forecasting," *International Journal of Forecasting*, vol. 31, no. 2, pp. 286–295, 2015.

[110] T. Hong, J. Wilson, and J. Xie, "Long term probabilistic load forecasting and normalization with hourly information," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 456–462, 2014.

[111] B. Hu, Y. Chen, and E. Keogh, "Time series classification under more realistic assumptions," in *Proceedings of the 2013 SIAM International Conference on Data Mining*. SIAM, 2013, pp. 578–586.

[112] L. Hubert and P. Arabie, "Comparing partitions," *Journal of classification*, vol. 2, no. 1, pp. 193–218, 1985.

[113] R. Hyndman, A. B. Koehler, J. K. Ord, and R. D. Snyder, *Forecasting with exponential smoothing: the state space approach*. Springer Science & Business Media, 2008.

[114] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*. OTexts, 2018.

[115] R. J. Hyndman and S. Fan, "Density forecasting for long-term peak electricity demand," *IEEE Transactions on Power Systems*, vol. 25, no. 2, pp. 1142–1153, 2010.

[116] D. Ilić, P. G. da Silva, S. Karnouskos, and M. Jacobi, "Impact assessment of smart meter grouping on the accuracy of forecasting algorithms," in *Proceedings of the 28th Annual ACM Symposium on Applied Computing*. ACM, 2013, pp. 673–679.

[117] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010.

[118] A. K. Jain and R. C. Dubes, "Algorithms for clustering data," 1988.

[119] D. Jost, M. Speckmann, F. Sandau, and R. Schwinn, "A new method for day-ahead sizing of control reserve in germany under a 100% renewable energy sources scenario," *Electric Power Systems Research*, vol. 119, pp. 485–491, 2015.

[120] K. Kalpakis, D. Gada, and V. Puttagunta, "Distance measures for effective clustering of arima time-series," in *Proceedings 2001 IEEE International Conference on Data Mining*. IEEE, 2001, pp. 273–280.

[121] L. Kaufman and P. J. Rousseeuw, *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009, vol. 344.

[122] E. Keogh, "A decade of progress in indexing and mining large time series databases," in *Proceedings of the 32nd International Conference on Very Large Data Bases*. VLDB Endowment, 2006, pp. 1268–1268.

[123] E. Keogh, S. Chu, D. Hart, and M. Pazzani, "Segmenting time series: A survey and novel approach," in *Data mining in time series databases*. World Scientific, 2004, pp. 1–21.

[124] E. Keogh and J. Lin, "Clustering of time-series subsequences is meaningless: implications for previous and future research," *Knowledge and information systems*, vol. 8, no. 2, pp. 154–177, 2005.

[125] E. Keogh and C. A. Ratanamahatana, "Exact indexing of dynamic time warping," *Knowledge and information systems*, vol. 7, no. 3, pp. 358–386, 2005.

[126] E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana, "The ucr time series classification/clustering homepage (2006)," *URL www. cs. ucr. edu/~ eamonn/time_ series_data*, 2002.

[127] A. Khotanzad, R. Afkhami-Rohani, T.-L. Lu, A. Abaye, M. Davis, and D. J. Maratukulam, "Annstlf-a neural-network-based electric load forecasting system," *IEEE Transactions on Neural Networks*, vol. 8, no. 4, pp. 835–846, 1997.

[128] A. Khotanzad, R. Afkhami-Rohani, and D. Maratukulam, "Annstlf-artificial neural network short-term load forecaster-generation three," *IEEE Transactions on Power Systems*, vol. 13, no. 4, pp. 1413–1422, 1998.

[129] K.-H. Kim, J.-K. Park, K.-J. Hwang, and S.-H. Kim, "Implementation of hybrid short-term load forecasting system using artificial neural networks and fuzzy expert systems," *IEEE Transactions on Power Systems*, vol. 10, no. 3, pp. 1534–1539, 1995.

[130] S. Kippelt, T. Schlüter, and C. Rehtanz, "Flexible dimensioning of control reserve for future energy scenarios," in *2013 IEEE Grenoble Conference*. IEEE, 2013, pp. 1–6.

[131] M. Koivisto, P. Heine, I. Mellin, and M. Lehtonen, "Clustering of connection points and load modeling in distribution systems," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1255–1265, 2013.

[132] R. Krishnapuram, H. Frigui, and O. Nasraoui, "Fuzzy and possibilistic shell clustering algorithms and their application to boundary detection and surface approximation. i," *IEEE Transactions on Fuzzy Systems*, vol. 3, no. 1, pp. 29–43, 1995.

[133] M. H. Kutner, C. J. Nachtsheim, J. Neter, W. Li *et al.*, "Applied linear statistical models," 2005.

[134] S.-H. Lai and T. Hong, "When one size no longer fits all: Electric load forecasting with a geographic hierarchy," *SAS White Paper*, 2013.

[135] L. J. Latecki, V. Megalooikonomou, Q. Wang, R. Lakaemper, C. A. Ratanamahatana, and E. Keogh, "Elastic partial matching of time series," in *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer, 2005, pp. 577–584.

[136] A. Lehtilä, P. Silvennoinen, and J. Vira, "A belief network model for forecasting within the electricity sector," *Technological Forecasting and Social Change*, vol. 38, no. 2, pp. 135–150, 1990.

[137] Y. Lei, J. C. Bezdek, J. Chan, N. X. Vinh, S. Romano, and J. Bailey, "Extending information-theoretic validity indices for fuzzy clustering," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 4, pp. 1013–1018, 2017.

[138] W. Lipo, *Support vector machines: theory and applications*. Springer Science & Business Media, 2005, vol. 177.

[139] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00019*, 2015.

[140] J. R. Lloyd, "Gefcom2012 hierarchical load forecasting: Gradient boosting machines and gaussian processes," *International Journal of Forecasting*, vol. 30, no. 2, pp. 369–374, 2014.

[141] S. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[142] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14. Oakland, CA, USA, 1967, pp. 281–297.

[143] F. Martínez-Álvarez, A. Troncoso, G. Asencio-Cortés, and J. Riquelme, "A survey on data mining techniques applied to electricity-related time series forecasting," *Energies*, vol. 8, no. 11, pp. 13 162–13 193, 2015.

[144] C. Maurer, S. Krahl, and H. Weber, "Dimensioning of secondary and tertiary control reserve by probabilistic methods," *European Transactions on Electrical Power*, vol. 19, no. 4, pp. 544–552, 2009.

[145] W. Meesrikamolkul, V. Niennattrakul, and C. A. Ratanamahatana, "Shape-based clustering for time series data," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2012, pp. 530–541.

[146] M. Meilă, "Comparing clusteringsan information based distance," *Journal of multivariate analysis*, vol. 98, no. 5, pp. 873–895, 2007.

[147] N. Menemenlis, M. Huneault, and A. Robitaille, "Computation of dynamic operating balancing reserve for wind power integration for the time-horizon 1–48 hours," *IEEE Transactions on Sustainable Energy*, vol. 3, no. 4, pp. 692–702, 2012.

[148] G. Migliavacca, *Advanced technologies for future transmission grids*. Springer Science & Business Media, 2012.

[149] G. W. Milligan and M. C. Cooper, "An examination of procedures for determining the number of clusters in a data set," *Psychometrika*, vol. 50, no. 2, pp. 159–179, 1985.

[150] M. Misiti, Y. Misiti, G. Oppenheim, J.-M. Poggi *et al.*, "Optimized clusters for disaggregated electricity load forecasting," *Revstat*, vol. 8, no. 2, pp. 105–124, 2010.

[151] C. S. Möller-Levet, F. Klawonn, K.-H. Cho, and O. Wolkenhauer, "Fuzzy clustering of short time-series and unevenly distributed sampling points," in *International Symposium on Intelligent Data Analysis*. Springer, 2003, pp. 330–340.

[152] J. A. Momoh, *Smart grid: fundamentals of design and analysis*. John Wiley & Sons, 2012, vol. 63.

[153] J. M. Morales, A. J. Conejo, H. Madsen, P. Pinson, and M. Zugno, *Integrating renewables in electricity markets: operational problems*. Springer Science & Business Media, 2013, vol. 205.

[154] M. D. Morse and J. M. Patel, "An efficient and accurate method for evaluating time series similarity," in *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*. ACM, 2007, pp. 569–580.

[155] A. Motamedi, H. Zareipour, and W. D. Rosehart, "Electricity price and demand forecasting in smart grids," *IEEE Transactions on Smart Grid*, vol. 3, no. 2, pp. 664–674, 2012.

[156] K. P. Murphy and S. Russell, "Dynamic bayesian networks: representation, inference and learning," 2002.

[157] R. Nedellec, J. Cugliari, and Y. Goude, "Gefcom2012: Electric load forecasting and backcasting with semi-parametric models," *International Journal of Forecasting*, vol. 30, no. 2, pp. 375–381, 2014.

[158] V. Niennattrakul and C. A. Ratanamahatana, "On clustering multimedia time series data using k-means and dynamic time warping," in *2007 International Conference on Multimedia and Ubiquitous Engineering (MUE'07)*. IEEE, 2007, pp. 733–738.

[159] ——, "Shape averaging under time warping," in *2009 6th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, vol. 2. IEEE, 2009, pp. 626–629.

[160] D.-x. Niu, H.-f. Shi, and D. D. Wu, "Short-term load forecasting using bayesian neural networks learned by hybrid monte carlo algorithm," *Applied Soft Computing*, vol. 12, no. 6, pp. 1822–1827, 2012.

[161] J. Nowicka-Zagrajek and R. Weron, "Modeling electricity loads in california: Arma models with hyperbolic noise," *Signal Processing*, vol. 82, no. 12, pp. 1903–1915, 2002.

[162] J. Nowotarski, B. Liu, R. Weron, and T. Hong, "Improving short term load forecast accuracy via combining sister forecasts," *Energy*, vol. 98, pp. 40–49, 2016.

[163] M. A. Ortega-Vazquez and D. S. Kirschen, "Estimating the spinning reserve requirements in systems with significant wind power generation penetration," *IEEE Transactions on Power Systems*, vol. 24, no. 1, pp. 114–124, 2009.

[164] V. Pacelli, V. Bevilacqua, and M. Azzollini, "An artificial neural network model to forecast exchange rates," *Journal of Intelligent Learning Systems and Applications*, vol. 3, no. 02, p. 57, 2011.

[165] C. Padraig and S. Jane Delany, "k-nearest neighbour classifiers," *Multiple Classifier Systems*, vol. 34, no. 8, 2007.

[166] P.-F. Pai and W.-C. Hong, "Support vector machines with simulated annealing algorithms in electricity load forecasting," *Energy Conversion and Management*, vol. 46, no. 17, pp. 2669–2688, 2005.

[167] G. Panchal, A. Ganatra, Y. Kosta, and D. Panchal, "Behaviour analysis of multilayer perceptrons with multiple hidden neurons and hidden layers," *International Journal of Computer Theory and Engineering*, vol. 3, no. 2, pp. 332–337, 2011.

[168] S. Paoletti, M. Casini, A. Giannitrapani, A. Facchini, A. Garulli, and A. Vicino, "Load forecasting for active distribution networks," in *2011 2nd IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies*. IEEE, 2011, pp. 1–6.

[169] A. D. Papalexopoulos and T. C. Hesterberg, "A regression-based approach to short-term system load forecasting," *IEEE Transactions on Power Systems*, vol. 5, no. 4, pp. 1535–1547, 1990.

[170] J. Paparrizos and L. Gravano, "k-shape: Efficient and accurate clustering of time series," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*. ACM, 2015, pp. 1855–1870.

[171] V. I. Pavlovic and T. S. Huang, *Dynamic bayesian networks for information fusion with applications to human-computer interfaces*. University of Illinois at Urbana-Champaign, 1999, vol. 44.

[172] J. Pearl, "Probabilistic inference in intelligent systems," 1988.

[173] F. Petitjean and P. Gançarski, "Summarizing a set of time series by averaging: From steiner sequence to compact multiple alignment," *Theoretical Computer Science*, vol. 414, no. 1, pp. 76–91, 2012.

[174] F. Petitjean, A. Ketterlin, and P. Gançarski, "A global averaging method for dynamic time warping, with applications to clustering," *Pattern Recognition*, vol. 44, no. 3, pp. 678–693, 2011.

[175] M. Piao, H. S. Shon, J. Y. Lee, and K. H. Ryu, "Subspace projection method based clustering analysis in load profiling," *IEEE Transactions on Power Systems*, vol. 29, no. 6, pp. 2628–2635, 2014.

[176] P. Pinson *et al.*, "Wind energy: Forecasting challenges for its operational management," *Statistical Science*, vol. 28, no. 4, pp. 564–585, 2013.

[177] C. Pouzat, O. Mazor, and G. Laurent, "Using noise signature to optimize spike-sorting and to assess neuronal classification quality," *Journal of neuroscience methods*, vol. 122, no. 1, pp. 43–57, 2002.

[178] F. L. Quilumba, W.-J. Lee, H. Huang, D. Y. Wang, and R. L. Szabados, "Using smart meter data to improve the accuracy of intraday load forecasting considering customer behavior similarities," *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 911–918, 2015.

[179] R. Ramanathan, R. Engle, C. W. Granger, F. Vahid-Araghi, and C. Brace, "Short-run forecasts of electricity loads and peaks," *International Journal of Forecasting*, vol. 13, no. 2, pp. 161–174, 1997.

[180] S. Ramos, J. Soares, Z. Vale, and S. Ramos, "Short-term load forecasting based on load profiling," in *2013 IEEE Power & Energy Society General Meeting*. IEEE, 2013, pp. 1–5.

[181] D. Ranaweera, N. Hubele, and G. Karady, "Fuzzy logic for short term load forecasting," *International Journal of Electrical Power & Energy Systems*, vol. 18, no. 4, pp. 215–222, 1996.

[182] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical association*, vol. 66, no. 336, pp. 846–850, 1971.

[183] S. Rani and G. Sikka, "Recent techniques of clustering of time series data: a survey," *International Journal of Computer Applications*, vol. 52, no. 15, 2012.

[184] C. A. Ratanamahatana and E. Keogh, "Making time-series classification more accurate using learned constraints," in *Proceedings of the 2004 SIAM International Conference on Data Mining*. SIAM, 2004, pp. 11–22.

[185] C. A. Rocha, Á. L. de Santana, C. R. Francês, U. Bezerra, A. Tupiassú, V. Gato, L. Rego, and J. Costa, "Decision support in power systems based on load forecasting models and influence analysis of climatic and socio-economic factors," in *Optics East 2006*. International Society for Optics and Photonics, 2006, pp. 63 830I–63 830I.

[186] E. J. Ruiz, V. Hristidis, C. Castillo, A. Gionis, and A. Jaimes, "Correlating financial time series with micro-blogging activity," in *Proceedings of the fifth ACM International Conference on Web Search and Data Mining*. ACM, 2012, pp. 513–522.

[187] D. Ruppert, M. P. Wand, and R. J. Carroll, "Semiparametric regression during 2003–2007," *Electronic journal of statistics*, vol. 3, p. 1193, 2009.

[188] U. Rutishauser, E. M. Schuman, and A. N. Mamelak, "Online detection and sorting of extracellularly recorded action potentials in human medial temporal lobe recordings, in vivo," *Journal of neuroscience methods*, vol. 154, no. 1-2, pp. 204–224, 2006.

[189] H. Sakoe, S. Chiba, A. Waibel, and K. Lee, "Dynamic programming algorithm optimization for spoken word recognition," *Readings in speech recognition*, vol. 159, p. 224, 1990.

[190] R. Salakhutdinov and G. Hinton, "Deep boltzmann machines," in *Artificial intelligence and statistics*, 2009, pp. 448–455.

[191] A. M. Schäfer and H.-G. Zimmermann, "Recurrent neural networks are universal approximators," *International Journal of Neural Systems*, vol. 17, no. 04, pp. 253–263, 2007.

[192] S. W. Session, "World commission on environment and development," *Our Common Future; WCED, Ed.; Oxford University Press: Oxford, UK*, 1987.

[193] A. Shahzadeh, A. Khosravi, and S. Nahavandi, "Improving load forecast accuracy by clustering consumers using smart meter data," in *2015 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2015, pp. 1–7.

[194] a. o. Simon S., Haykin, *Neural networks and learning machines*. Upper Saddle River: Pearson education, 2009, vol. 3.

[195] K.-B. Song, Y.-S. Baek, D. H. Hong, and G. Jang, "Short-term load forecasting for the holidays using fuzzy linear regression method," *IEEE Transactions on Power Systems*, vol. 20, no. 1, pp. 96–101, 2005.

[196] Q. Song and B. S. Chissom, "Fuzzy time series and its models," *Fuzzy sets and systems*, vol. 54, no. 3, pp. 269–277, 1993.

[197] R. G. Steadman, "A universal scale of apparent temperature," *Journal of Climate and Applied Meteorology*, vol. 23, no. 12, pp. 1674–1687, 1984.

[198] ——, "Norms of apparent temperature in australia," *Aust. Met. Mag*, vol. 43, pp. 1–16, 1994.

[199] B. Stephen, A. J. Mutanen, S. Galloway, G. Burt, and P. Järventausta, "Enhanced load profiling for residential network customers," *IEEE Transactions on Power Delivery*, vol. 29, no. 1, pp. 88–96, 2014.

[200] C. L. Stimmel, *Big data analytics strategies for the smart grid*. Auerbach Publications, 2016.

[201] L. Suganthi, S. Iniyan, and A. A. Samuel, "Applications of fuzzy logic in renewable energy systems–a review," *Renewable and sustainable energy reviews*, vol. 48, pp. 585–607, 2015.

[202] M. Sun, I. Konstantelos, and G. Strbac, "C-vine copula mixture model for clustering of residential electrical load pattern data," *IEEE Transactions on Power Systems*, vol. 32, no. 3, pp. 2382–2393, 2017.

[203] S. B. Taieb and R. J. Hyndman, "A gradient boosting approach to the kaggle load forecasting competition," *International Journal of Forecasting*, vol. 30, no. 2, pp. 382–394, 2014.

[204] J. W. Taylor, "An evaluation of methods for very short-term load forecasting using minute-by-minute british data," *International Journal of Forecasting*, vol. 24, no. 4, pp. 645–658, 2008.

[205] J. W. Taylor, P. E. McSharry *et al.*, "Short-term load forecasting methods: An evaluation based on european data," *IEEE Transactions on Power Systems*, vol. 22, no. 4, pp. 2213–2219, 2007.

[206] S. Theodoridis and K. Koutroumbas, *Pattern Recognition & Matlab Intro*. Academic Press, Inc., 2010.

[207] T. Tieleman, "Training restricted boltzmann machines using approximations to the likelihood gradient," in *Proceedings of the 25th International Conference on Machine Learning*. ACM, 2008, pp. 1064–1071.

[208] J. Twidell and T. Weir, *Renewable energy resources*. Routledge, 2015.

[209] J. Valenzuela, M. Mazumdar, and A. Kapoor, "Influence of temperature and load forecast uncertainty on estimates of power generation production costs," *IEEE Transactions on Power Systems*, vol. 15, no. 2, pp. 668–674, 2000.

[210] E. D. Varga, S. F. Beretka, C. Noce, and G. Sapienza, "Robust real-time load profile encoding and classification framework for efficient power systems operation," *IEEE Transactions on Power Systems*, vol. 30, no. 4, pp. 1897–1904, 2015.

[211] N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance," *Journal of Machine Learning Research*, vol. 11, no. Oct, pp. 2837–2854, 2010.

[212] M. Vlachos, D. Gunopoulos, and G. Kollios, "Discovering similar multidimensional trajectories," in *icde*. IEEE, 2002, p. 0673.

[213] L. Wang and T. Jiang, "On the complexity of multiple sequence alignment," *Journal of computational biology*, vol. 1, no. 4, pp. 337–348, 1994.

[214] P. Wang, B. Liu, and T. Hong, "Electric load forecasting with recency effect: A big data approach," *International Journal of Forecasting*, vol. 32, no. 3, pp. 585–597, 2016.

[215] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, and E. Keogh, "Experimental comparison of representation methods and distance measures for time series data," *Data Mining and Knowledge Discovery*, vol. 26, no. 2, pp. 275–309, 2013.

[216] X. Wang, K. Smith, and R. Hyndman, "Characteristic-based clustering for time series data," *Data mining and knowledge Discovery*, vol. 13, no. 3, pp. 335–364, 2006.

[217] Y. Wang, Q. Xia, and C. Kang, "Unit commitment with volatile node injections by using interval optimization," *IEEE Transactions on Power Systems*, vol. 26, no. 3, pp. 1705–1713, 2011.

[218] Y. Wang, Q. Chen, C. Kang, and Q. Xia, "Clustering of electricity consumption behavior dynamics toward big data applications," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2437–2447, 2016.

[219] Y. Wang, Q. Chen, C. Kang, M. Zhang, K. Wang, and Y. Zhao, "Load profiling and its application to demand response: A review," *Tsinghua Science and Technology*, vol. 20, no. 2, pp. 117–129, 2015.

[220] K. Warren, R. Ambrosio, B. Chen, Y. Fu, S. Ghosh, D. Phan, M. Sinn, C. Tian, and C. Visweswariah, "Managing uncertainty in electricity generation and demand forecasting," *IBM Journal of Research and Development*, vol. 60, no. 1, pp. 8–1, 2016.

[221] P. J. Werbos *et al.*, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[222] R. Weron, *Modeling and forecasting electricity loads and prices: A statistical approach*. John Wiley & Sons, 2007, vol. 403.

[223] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics bulletin*, vol. 1, no. 6, pp. 80–83, 1945.

[224] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural computation*, vol. 1, no. 2, pp. 270–280, 1989.

[225] L. Wu, M. Shahidehpour, and T. Li, "Stochastic security-constrained unit commitment," *IEEE Transactions on Power Systems*, vol. 22, no. 2, pp. 800–811, 2007.

[226] J. Xie, Y. Chen, T. Hong, and T. D. Laing, "Relative humidity for load forecasting models," *IEEE Transactions on Smart Grid*, vol. 9, no. 1, pp. 191–198, 2018.

[227] J. Xie and T. Hong, "Wind speed for load forecasting models," *Sustainability*, vol. 9, no. 5, p. 795, 2017.

[228] Y. Yan, Y. Qian, H. Sharif, and D. Tipper, "A survey on smart grid communication infrastructures: Motivations, requirements and challenges," *IEEE communications surveys & tutorials*, vol. 15, no. 1, pp. 5–20, 2012.

[229] J. Yang and J. Leskovec, "Patterns of temporal variation in online media," in *Proceedings of the fourth ACM International Conference on Web Search and Data Mining*. ACM, 2011, pp. 177–186.

[230] S.-l. Yang, C. Shen *et al.*, "A review of electric load classification in smart grid environment," *Renewable and Sustainable Energy Reviews*, vol. 24, pp. 103–110, 2013.

[231] R. Yu, Y. Zhang, S. Gjessing, C. Yuen, S. Xie, and M. Guizani, "Cognitive radio based hierarchical communications infrastructure for smart grid," *IEEE network*, vol. 25, no. 5, pp. 6–14, 2011.

[232] D. Zelterman, "Bayesian artificial intelligence," 2005.

[233] D. Zhai, A. Breipohl, F. Lee, and R. Adapa, "The effect of load uncertainty on unit commitment risk," *IEEE Transactions on Power Systems*, vol. 9, no. 1, pp. 510–517, 1994.

[234] Y. Zhang, J. Wang, and X. Wang, "Review on probabilistic forecasting of wind power generation," *Renewable and Sustainable Energy Reviews*, vol. 32, pp. 255–270, 2014.

[235] Z. Zhou and A. Botterud, "Dynamic scheduling of operating reserves in co-optimized electricity markets with wind power," *IEEE Transactions on Power Systems*, vol. 29, no. 1, pp. 160–171, 2014.