



University of Venda

**RENEWABLE ENERGY FORECASTING IN
SOUTH AFRICA**

By

**Mamphaga Ratshilengo
14003914**

Master of Science Degree in Statistics

in the

Department of Statistics
School of Mathematical and Natural Sciences
University of Venda, Thohoyandou, Limpopo
South Africa

Supervisor: **Dr C. Sigauke**

Co-Supervisor: **Dr A. Bere**

12 June 2021

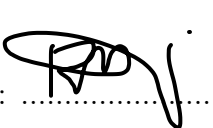
Abstract

Renewable energy forecasts are critical to renewable energy grids and backup plans, operational plans and short-term power purchases. This dissertation focused on forecasting solar irradiance at one radiometric station in South Africa using high-frequency data obtained from the Vuwani radiometric station (USAid Venda). The aim of this dissertation was to compare the predictive performance of the Genetic Algorithm (GA), recurrent neural networks (RNN) and k-nearest neighbour (KNN) models in forecasting short-term solar irradiance where KNN is used as a benchmark model. From the results it is discovered that the RNN is the best forecasting model in terms of the relative mean absolute error (rMAE). The forecasts of the machine learning algorithms combined using convex combination technique and quantile regression averaging (QRA) found that QRA is the best model. Predictive interval widths analysis with 95% level of confidence was performed and the results showed that QRA over RNN is the best model for forecasting solar irradiance when looking at the PICP and PANAW. The Diebold-Mariano test discovered that the tests fall between the -1.96 and 1.96 range, which tells us that it accepts the null hypothesis. The Murphy diagram presented and showed the 95% pointwise confidence intervals. The study will have an impact on the South African power utility decision-makers to align electricity demand and its supply in an efficient way that promotes potential economic growth and environmental sustainability.

Keywords: *Genetic algorithms, Global horizontal irradiance, K-nearest neighbour, Quantile regression averaging, Murphy diagram, Recurrent neural networks.*

Declaration

I, Mamphaga Ratshilengo [student number:14003914], declare that this mini-dissertation for a Master of Science in Statistics degree entitled, “**Renewable energy forecasting in South Africa**” is my own work and has not been submitted for any other degree at any other university or institution. The dissertation does not contain any other persons’ writing unless specifically acknowledged and cited accordingly.

Signed (Student): 

Date: ..12 June 2021.....

Dedication

This work is dedicated to my mother Talitha Thivhudziswi Mercy Ratshilengo, my brother MD Ratshilengo, my two sisters Mboniseni and Ndiitwani and my supportive girl friend Ompha Felicia Mudau.

Acknowledgment

I would like to thank God, He is the one who gave me the expertise, understanding, strength and power to carry out this dissertation. Again, I wish to thank Dr. Caston Sigauke and Dr. Alphonse Bere, my supervisors, for guidance and support through the dissertation. Again, I wish particularly to thank my mother T.T Ratshilengo and Ompha Mudau for their support, they have been supporting me in all directions without complaining and my big brother for his involvement in my academic activities, pushing me day and night. I am so grateful for having them, they are my best friends plus my family members. I am also grateful to the National e-Science Postgraduate Teaching and Training Platform (NEPTTP) for funding my studies and project manager Ms. Casey Sparkes for encouraging me to take this opportunity.

Table of contents

Abstract	i
Declaration	ii
Dedication	iv
Acknowledgement	v
Contents	v
List of Abbreviations	viii
Symbols	x
Publications	xi
1 Introduction	1
1.1 Background	3
1.2 Problem statement	6
1.3 Purpose of the study	7
1.4 Dissertation scope	7
1.5 Outline of the dissertation	8
2 Literature review	9
2.1 Introduction	9
2.2 Renewable energy components	9
2.3 Review of literature on forecasting using machine learning techniques	10
2.4 Review of literature on forecasting using non-machine learning techniques	11

2.5	An overview of renewable energy forecasting in South Africa	11
2.6	Genetic algorithm	15
2.7	Recurrent neural network	17
2.8	Conclusions from literature	19
3	Methodology	21
3.1	Introduction	21
3.2	Genetic algorithm	21
3.3	Recurrent neural networks	24
3.4	K-nearest neighbour	25
3.5	Data and features	26
3.6	Variable selection	28
3.7	Prediction intervals	28
3.8	Performance measures	29
3.8.1	Mean absolute error, Mean square error, Root mean square error, Relative MAE, Relative RMSE	31
3.8.2	Pinball loss function	31
3.8.3	Diebold-Mariano test	32
3.8.4	Murphy diagram	33
3.9	Forecast error distribution	34
3.10	Percentage improvement	34
3.11	Conclusion	35
3.12	Implementation	35
4	Data analysis and discussion	36
4.1	Introduction	36
4.2	The data	36
4.2.1	Exploratory data analysis	36
4.3	Variable selection using LASSO	43
4.4	Machine learning models	43
4.5	Combination of the forecasts	49
4.5.1	Convex combination	49
4.5.2	Quantile regression averaging	52
4.6	Models Comparative Analysis	55
4.6.1	Evaluation of prediction intervals	55
4.6.2	Residual analysis	59
4.6.3	Percentage improvement	63
4.6.4	Diebold-Mariano test	64

4.6.5	Murphy diagram	64
4.7	Conclusion	68
5	Conclusion and future work	69
5.1	Introduction	69
5.2	Findings of the dissertation	69
5.3	Future work	71
5.4	Conclusion	71
	References	72

Abbreviations

ANN	Artificial Neural Network
RNN	Recurrent Neural Network
ARIMA	Auto-Regressive Integrated Moving Average
ARMA	Auto-Regressive Moving Average
PSO	Particle Swarm Optimisation
CARDS	Coupled Auto-Regressive and Dynamical System
CV	Cross Validation
AAKR	Auto-Associative Kernel Regression
PV	Photovoltaic
ELD	Economic Load Dispatch
SUI	Solar Utility Index
GA	Genetic Algorithm
KNN	K-Nearest Neighbor
SA	South Africa
EMD	Empirical Mode Decomposition
FFNN	Feed Forward Neural Network
GHI	Global Horizontal Irradiance
LASSO	Least Absolute Shrinkage and Selection Operator
LSTM	Long Short-Term Memory
STLF	Short-Term Load Forecast
LCOE	Levelised Energy Cost
RES	Renewable Energy Sources
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
RMSE	Root Mean Square Error
rMAE	Relative Mean Absolute Error
rRMSE	Relative Root Mean Square Error
AE	Absolute Error

MD	Murphy Diagram
DM	Diebold-Mariano
MSE	Mean Square Error
NWP	Numerical Weather Prediction
PINC	Prediction Interval with Nominal Confidence
PIW	Prediction Interval Width
PI	Prediction Interval
QRA	Quantile Regression Averaging
RBF	Radial Basis Function
RBFNN	Radial Basis Function Neural Network
RF	Random Forest
GCV	Generalised Cross Validation
MLP	Multilayer Perceptron
SAURAN	Southern African Universities Radiometric Network
SVD	Singular Value Decomposition
SVM	Support Vector Machine
SVR	Support Vector Regression
W/m^2	Watts Per Square Meter.

List of Notation and Special Symbols

$F_{Y X}$	Conditional distribution function of X given X.
$F_{Y X}^{-1}$	Inverse function of the conditional distribution function of Y given X.
$Q_{Y X}$	Conditional quantile function.
$Q_{0.5}(Y X)$	Median quantile regression function.
s_j	Smooth functions.
$Q_y X, Z(\tau)$	Average loss function.
ρ_τ	Pinball loss function.
β_0	Intercept.
β_j	Coefficients.
\mathbf{B}	Model matrix of basis functions.
\mathbf{X}_t	Covariates matrix.
θ	Parameter vector.
$\ell(\theta)$	Maximised log-likelihood.
$\varepsilon_{t,\tau}$	Random error term.

Research Outputs

A list of research outputs from this dissertation is given below.

Peer Reviewed Journal Publications

1. Ratshilengo, M., Sigauke, C. and Bere, A. (2021). Short-Term Solar Power Forecasting Using Genetic Algorithms: An Application Using South African Data, Applied Sciences, 11, 4214. <https://doi.org/10.3390/app11094214>

Chapter 1

Introduction

Renewable energy sources (RES) are rising rapidly in different countries, powered by the cost reduction of wind turbines and photovoltaic (PV) panels ([Andrade and Bessa, 2017](#)). They are increasingly becoming the future's dominant energy source, but harvesting them requires an understanding of the mechanisms of their volatility and the ability to predict various environmental processes over a scale range ([Kariniotakis, 2017](#)). Such understanding of the environment is the key to renewable energy processing, in particular solar and wind power.

When the global population and industrialisation continue to rise exponentially, the fossil fuels used to generate electricity are also rapidly depleting ([Zendejboudi et al., 2018](#)). The ongoing overuse of fossil fuels in the production of electricity continues to inflict environmental problems such as global warming. RESs are environmentally friendly and inexpensive ([Mohammadi et al., 2016](#)).

Onshore wind energy offers electricity competitively particularly in comparison to fossil-fuel. Solar PV levelised energy costs (LCOE) significantly reduced by 58 percent around 2010-15. Localised solar power and offshore wind energy demonstrate a 43 percent and 35 percent reduction in the last five years, respectively ([Taylor et al., 2016](#)). Variability and uncertainty of RES pose difficulties in the maintenance of power systems that offer high-quality probabilistic forecasts.

Solar energy is among the ultimate valuable sources of renewable energy that can add to addressing current environmental and energy issues in the electrical grid ([Zhandire, 2017](#)). It is widely regarded to be the world's best-growing energy industry ([Kleissl, 2013](#)). Nonetheless, for proper and effective management of the electrical grid, the integration of solar energy into the electrical grid requires detailed forecasting ([Cristaldi et al., 2017](#)). Policymakers for power utilities face the problem of coordinating demand and electricity supply in a cost-effective way which also greatly benefit upcoming economic growth and environmental sustainability.

Solar irradiance forecasting is critical for backup programming, decision making, short-term power production, relocating certain energy sources, backup utilisation scheduling and peak load demand ([Kostylev et al., 2011](#)). It is relevant for various other activities along with PV, agriculture, medical studies applications and desalination by seawater ([Rezrazi et al., 2016](#)). The steadily increasing integration of solar systems around the world is an indication of the need for accurate knowledge and understanding of solar resources in

the design of solar electric grids. Solar irradiance studies are therefore of great importance for the optimal project and power forecasting of PV grid-connected plants.

Wind energy is the use of wind to provide mechanical energy by wind turbines to transform electrical generators and to do certain types of work, such as grinding or pumping (Pierre et al., 2018). It is also clean, renewable energy and has much less environmental consequences as opposed to fossil fuel burning. It is perhaps the source of renewable energy attracting the most publicity from researchers and professionals (Raftery et al., 2013). Nonetheless, other forms of renewable energy must face a range of operational and economic issues. These problems include modeling and forecasting of the cycle of wind power generation at different temporal and spatial rates, to eventually be used as feedback for decision-making.

When solar and wind power are becoming more popular, forecasts that are embedded in energy management systems are becoming increasingly valuable for operators of electric power systems.

Renewable energy forecasting is a rapidly developing field and continuous attempts are underway to tailor products to the needs of forecast users (Andrade and Bessa, 2017).

1.1 Background

Liu and Jordan (1960) conducted initial studies of solar energy. Their re-

search concentrated on the association of clear days on a horizontal surface among normal diffuse and global irradiance elements, with measurements from 98 sites in Canada and the United States (US). Research teams have indeed been paying much attention to solar irradiance ever since, using several methods of modeling. Solar irradiance forecasts consist mainly of physical methods and content statistical models (Yang et al., 2013). Physical methods for solar irradiance forecasts are built on numerical weather predictions (NWP).

The solar irradiance forecasting techniques are divided into three groups which are machine learning, sophisticated mathematical statistics and numerical weather forecasting (Sun et al., 2018). Researchers were not previously paying attention to machine learning techniques but currently, they are paying attention to machine learning techniques such as artificial neural networks (ANN) and support vector machines (SVM) in forecasting solar irradiance (Fan et al., 2018).

Wind power was used the same way that humans placed sails in the wind (Al-Hassan et al., 1986). In what became Iran, Afghanistan and Pakistan by the 9th century, wind driven machines used to cultivate crops and pump water, the windmill and the windpump, were founded. Wind power was generally available and not limited to or later requiring fuel sources at the banks of fast-flowing streams (Lucas, 2006).

In July 1887 Prof James Blyth of Anderson's College, Glasgow (the prede-

cessor of Strathclyde University) designed the first windmill used to generate electricity in Scotland ([Price, 2005](#)). The first world's house to have its electricity supplied by wind power installed in the garden his holiday cottage.

Different forecast perspectives have been addressed as per the massive volume of studies in solar irradiance forecasting, including long-term (duration of three or more years), short-term (commonly less than 3 months but also with a period of up to 1 year) and medium-term horizons (typically 3 months to 1 year but has a period from one to three years). Meteorological variables like temperature, humidity, precipitation, cloud cover, wind speed, geographic variables such as latitude, longitude, altitude, etc. are also included in the function variables used during earlier studies and astronomical quantities such as declination, hour, angle and zenith angle as response variable ([Khatib and Elmenreich, 2015](#)).

This study focuses on forecasting solar energy using genetic algorithm (GA), recurrent neural networks (RNN) and the k-nearest neighbours (KNN) models where the KNN is being used as a benchmark model that has not yet been implemented to South African solar energy, to the best of our knowledge. South Africa receives sunlight throughout the year so the natural advantage of South Africa is that it is included in the world's highest in renewable energy.

The global average annual 24-hour solar radiation for South Africa has been about 220 W/m^2 , 150 W/m^2 for parts of the US and about 100 W/m^2

towards Europe and the United Kingdom ([Al-Karaghoul and Kazmerski, 2013](#)). South Africa's local assets are, therefore, one of the world's best. Little work on solar irradiance and wind forecasting has been conducted in South Africa and this dissertation aims to make a contribution in this crucial field.

1.2 Problem statement

South Africa (SA) is one of the most developed countries on the African continent. SA still has a surplus of fossil fuel in the form of coal, hence many coals powered power stations exist and are newly built ([Al-Karaghoul and Kazmerski, 2013](#)). At the very same time, SA has a surplus of sunlight which is very well suited for solar water heating and generating electricity. Solar-powered heating and cooling become more appealing with the rising prices of coal-powered energy. The legislation is yet one major hurdle. Actually in SA domestic grid linked solar systems are not allowed to feedback into the grid and though the concept is available to few municipalities. So there should be a good mix of sustainable energy and conventional energy sources in SA in the future. The SA government relates the extraordinary fall in the balance of the energy reserve over recent years to rapid economic growth and the resulting rise in electricity demand ([Odhiambo, 2010](#)). [Odhiambo \(2009\)](#) observed that the correlation between electricity consumption and economic growth in SA is directly proportional. The forecasts which will be obtained in this dissertation will help decision-makers in Eskom in short-term load forecasting (STLF) planning of the operations of the renewable energy companies.

1.3 Purpose of the study

Aim

This study aims to compare the predictive performance of GA, RNN and KNN techniques on forecasting solar irradiance at one radiometric station in South Africa.

Objectives

The objectives of this study are to:

- develop GA, RNN and KNN models for short-term solar power forecasting,
- evaluate the performance of the models,
- evaluate the accuracy of the forecasts,
- evaluate the forecast combination,
- suggest policy implications.

1.4 Dissertation scope

This study is motivated by the fact that SA is experiencing difficulties in producing more electricity while there is a lot of renewable energy sources that can be used to produce more electricity such as using solar and wind energy. Hence, the purpose of this dissertation is to compare different models for forecasting renewable energy in SA and to predict the relationship between load

demand and exploratory variables such as humidity and barometric pressure.

The data which is going to be used in this dissertation is obtained from the Vuwani radiometric power station. The minute's historical load data is collected from January 2020 till October 2020. GA, RNN and the benchmark model which is the KNN will be used in predicting the relationship between load and the exploratory variables named above. The dissertation analysis will be performed using Python and R statistical packages.

1.5 Outline of the dissertation

The continuation of the dissertation is organised in the following way:

In Chapter 2, we discuss previous literature on concepts of machine learning and also the summary of studies using strategies similar to those proposed. A theoretical framework of the machine learning methods being used in this mini-dissertation is given in Chapter 3. Chapter 4 presents the analysis of the dissertation. The conclusion and recommendation of the dissertation are given in Chapter 5.

Chapter 2

Literature review

2.1 Introduction

This chapter gives a summary of renewable energy in South Africa (SA), along with summaries of certain studies that have used the proposed methodology to forecast renewable energy (solar) in various areas of the world.

2.2 Renewable energy components

Renewable energy can be categorised into three groups, Solar energy which would be the energy we get from the sun, wind energy which is the energy we get from the wind and hydroelectricity which make up 70% of all the renewable electricity and generate around 16.6% of global energy resources ([Andrade and Bessa, 2017](#)).

2.3 Review of literature on forecasting using machine learning techniques

Several alternative ways of forecasting renewable energy have been used. In addition to the machine learning methods, this part describes several of these alternative solution applicable to renewable energy forecasting.

Artificial neural networks (ANNs) are being used frequently in renewable energy forecasting. ANNs based methods actually implemented are feedforward ANN, cascade-forward backpropagation ANN, generalised regression ANN, neuro-fuzzy ANN and optimised ANN-genetic algorithm. ANNs can map relationships between various variables given there is available information to learn from ([Haykin, 2009](#)).

[Abedinia et al. \(2018\)](#) applied a hybrid neural network and metaheuristic algorithm to their study which was about solar energy forecasting. [Cadenas and Rivera \(2009\)](#) used ANN for short term wind forecasting in La Venta, Oaxaca and Mexico. In their report, ANN model used to forecast the hourly time series indicative of the site. A recurrent neural network (RNN) was proposed which was based on the control strategy for storage of the battery capacity in systems generation with renewable energy sources ([Capizzi et al., 2011](#)). [Tsai et al. \(2017\)](#) applied a neural network to the short-term load and wind power forecasting based on prediction interval.

2.4 Review of literature on forecasting using non-machine learning techniques

In addition to machine learning approaches, a variety of alternative methods were applied to renewable energy forecasting. [Peng et al. \(2013\)](#) suggested a combined autoregressive and dynamic system (CARDS) for solar irradiance prediction. The suggested approach improves the global solar irradiance forecast accuracy 1 hour ahead by 30% compared to ANNs models.

2.5 An overview of renewable energy forecasting in South Africa

A lot of research has been done in the past with interest in predicting SA's renewable energy using various methods and techniques. [Andrade and Bessa \(2017\)](#) used a system of numerical weather predictions (NWP) to conduct their research on enhancing renewable energy forecasting. Throughout their report, they suggested a feature engineering approach to obtain further details for wind and solar energy forecasts from a NWP's grid. They considered that adequate processing of direct NWP data elements can boost the forecasting capability and that the renewable energy forecaster could invest time in this knowledge discovery process.

[Tartibu and Kabengele \(2018\)](#) suggested the use of ANN as a new strategy for assessing future energy consumption levels in SA. In their study, particle swarm optimisation (PSO) has been used to train ANNs. Estimates of the annual electricity demand were determined per scenario and it was noted that the proposed ANN approach attains a comparatively better en-

ergy demand forecast within acceptable errors. [Sigauke \(2017\)](#) discussed the implementation of generalised additive models (GAMs) to the prediction of medium-term demand for electricity utilising data from SA. In his study variable selection was carried out via hierarchical interactions with the use of the least absolute shrinkage and selection operator (Lasso). [Marwala and Twala \(2014\)](#) used autoregressive moving average (ARMA), Neural networks (NNs) and Neuro-Fuzzy (NF) systems to model and forecast non-linear processes. They found that NF has a better ability than ARMA and NNs to model the system while NNs are better than ARMA.

[Mbuva et al. \(2017\)](#) discussed the one-hour and regular wind power generation forecast for the bayesian neural networks. They found that the NNs of bayesian neural networks exhibit comparable predictive performance to maximum likelihood neural networks both one hour and a day ahead forecast. [Mpfumali et al. \(2019\)](#) used Tellerie radiometric station's data for space between August 2009 to April 2010 for a probabilistic prediction of 24 hours ahead to the global horizontal irradiance (GHI). They used different techniques in forecasting including quantile regression average (QRA) and machine learning techniques and it is discovered that QRA gives higher accuracy than the machine learning techniques looking at the probabilistic error measures. [Zhandire \(2017\)](#) in their study of the classification of solar energy resources in SA using new index measured solar radiation resources in eight stations, five distinct groups using k-means clustering. He found that the solar utility index (SUI) has higher solar resources which discriminate and groups compared to other indices such as the probability of persistence

(POPD) and the fractal dimension.

King et al. (2007) proposed a GA framework using ‘negative load’ and the ‘inclusive’ strategy. They emphasized that, as wind power cost for the ‘negative load’ approach is presumed to be zero, the results does not show the actual cost. The actual cost of wind power should therefore be in the used in the ‘zero-fee’ strategy. They also discovered that the use of wind power in the economic load dispatch (ELD) computation affects cost, load volatility pictured by some other plants and reserve necessity because of the expected error.

Mellit and Shaari (2009) proposed an RNN based approach for forecasting the regular production of electricity from a PV power system (PVPS). A dataset has been used in their analysis for 4 years for RNN training and data for 1 year has been included for RNN testing. They presented the results based on the performance measure on which they used RNN, r, mean absolute percentage error (MAPE) and cumulative function distribution. In this study, three RNN models were compared where the proposed third RNN-II provided more reliable daily forecasts for generating electricity compared with the other proposed MLP and RNN-I models.

Grady et al. (2005) worked on GA approach where they used it to achieve optimal wind turbine location for higher production capacity when limiting the amount of turbines fixed and land used for every wind farm. In the study, they considered three cases which are: non-uniform wind having variable direction, uniform wind having direction and initials with six hundred (600)

individuals spread over 20 subpopulations.

[Cristaldi et al. \(2017\)](#) introduced a combination approach technique for short-term forecasting of solar irradiance and photovoltaic power (PV). The solar irradiance prediction carried out with the help of physical techniques named clear-sky techniques which forecast solar radiation in the exclusion of clouds. Short-term PV forecasting is then carried out with the help of auto-associative kernel regression (AAKR) strategy that is typically implemented for the detection of faults. The findings of the proposed model show increased efficacy in forecasting solar irradiance.

Apart from typical interplanetary radiation and hours of sunshine, [Adeala et al. \(2015\)](#) presents a report on the prediction of global solar radiation provided by the multiple linear regression. The study is undertaken in all nine South African provinces. The study revealed that the use of weather parameters for some locations increases the accuracy and efficiency of solar radiation models. [Khan and Byun \(2020\)](#) presented work on applying GA rooted optimised feature engineering and machine learning to the forecasting of energy consumption. The work was focusing on comparing various forecasting methods including XGBoost, support vector regression and k-nearest neighbor regressor algorithms. The work looked at various meteorological features which are temperature, wind speed, rain, humidity and time lags. In this work, it is discovered that combinations of prediction methods yield good results.

2.6 Genetic algorithm

John Holland adopted genetic algorithms (GAs) built on Darwin's evolutionary theory in 1960. In 1989, David E. Goldberg, his student, went on to improve it ([Sadeghi et al., 2014](#)). GA is a metaheuristic technique influenced by either a natural selection process that originally belonged to the larger number of evolutionary algorithms (EA). A metaheuristic is a genius technique that helps guide the study of other heuristics in the search for solutions to a particular problem ([Adewumi and Ali, 2010](#)). GAs are commonly used by biologically motivated operators to create high-quality optimisation solutions and search problems, such as mutation, crossover and selection ([Morales and Quezada, 1998](#)). GA's computer simulation helps populations of theoretical representations (chromosomes) of candidate solutions (individuals) to develop toward better solutions of an optimisation problem ([Adewumi and Ali, 2010](#)). Evolution starts with an initial population of all the random individuals and passes over different generations in search of optimal solutions. From each successive generation, the quality of individuals in the population is measured. Many individuals are stochastically collected, updated (mutated or reprocessed) from the existing population to produce a new population, which in the next phase of the algorithm becomes the growth population. GAs provide a generational enhancement in the fitness of chromosomes that generates certain chromosomes after several generations that are supposed to contain optimised variable settings ([Lucasius and Kateman, 1994](#)).

GAs have already been successfully implemented to different problems, ([Sivanandam and Deepa, 2008](#)), because of their simplicity, global perspective

and underlying parallel processing. It has been said that an algorithm is a successful tool in certain real-life COP situations for the startup analysis of metaheuristic implementation. The following is a brief overview of the GA image and HSAP operators.

Fitness assessment

The fitness for an individual's value into a population is formulated as a measure of the extent of fulfillment of the particular limitations that affect the allocation. The amount of fulfillment is determined as a component of paired space utilisation and weights that are assigned to contradictory constraints. Such two are used to determine the single's health value (current allocation). At the allocation levels of the hall and the block/floor (room), fitness values are taken in real numbers $[0,1]$.

For the allocation of the space, an individual's fitness within a population is defined as:

$$f = \sum_q w_q u_q \in [0, 1], \quad (2.6.1)$$

where u is the hall scale utilisation variable stated as:

$$u_q = \frac{a_{ij}}{s_i}, \quad (2.6.2)$$

where w_q is the weight of restriction q , s_i is the sum of category i students where i, j is fixed. In all individuals, throughout the population, the process is repeated when the best condition for the population is considered.

An individual's fitness value is calculated as in equation 2.6.2, with u is substituted by the usage variable at the level of the floor distribution offered as:

$$u_q = \frac{\mathbf{T}_i}{s_{ij}} \quad (2.6.3)$$

and

$$\mathbf{T}_i = \sum_{k=1}^p a_{ik}, \quad (2.6.4)$$

where a_{ik} is the amount of classification i allocations to floor k , p the integer assigning the largest(or lowest) floor for the hostel under consideration and s_{ij} is the sum of category i allocated to h all j students.

Operators of genetic algorithm

The distribution of the hall level is mutually exclusive from that of the distribution of blocks and rooms, which is why genetic algorithm (GA) operations are carried out at both levels. As both distributions align with various requirements and constraints, GA operators are implemented in a single manner. The initial population for hall distribution is randomly generated for every student group.

2.7 Recurrent neural network

Recurrent neural network (RNN) are NNs that have one or more loops of feedback. They are a class of ANNs in which a directed graphs together with a time series are generated by node-to-node connections. This allows temporary complex behavior to be seen. RNNs are extracted from feed-forward NNs

and can use their internal state (memory) to process input patterns of variable length (Prabhu and Garg, 1996). RNNs together with their variants have been implemented in various contexts whereby temporal dependence in the data is a vital implicit feature throughout the design of the model (Bianchi et al., 2017). The RNNs notable applications include sequence transduction, language modeling, speech recognition, word embedding learning, audio modeling, recognition of handwriting and image generation. In several applications, a popular version of RNN, called long-term short-term memory (LSTM), has been implemented (Hochreiter and Schmidhuber, 1997).

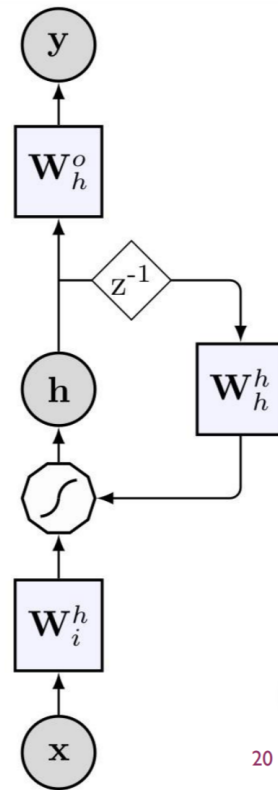
Hence RNN is also useful in time series prediction (Bianchi et al., 2016). These applications include weather forecast, load forecast and financial time series forecast. The structure presentation of the RNN is given in Figure 2.1 where it is having an input(\mathbf{x}), output(\mathbf{y}), internal state or memory of the network (\mathbf{h}), input weights (w_i^h), recurrent layer weights (w_h^h), output weights (w_h^0), time delay unit (z^{-1}) and neuron transfer function.

Using a recursive update, an RNN systematically summarises an input set in a specified-size state vector (Tokui et al., 2015). The equations of the RNN state and output difference discrete and time-independent are:

$$h[t + 1] = f(w_h^h h[t] + w_i^h x[t + 1] + b_n), \quad (2.7.1)$$

$$y[t + 1] = g(w_h^0 h[t + 1] + b_0), \quad (2.7.2)$$

where $f()$ is the transfer function of each neuron (generally the same non-linear function to all neurons). The RNN digital display is $g()$. Typically, the



20

Figure 2.1: Structure of the RNN (Bjerke, 2019).

identity function, the central processing units (neurons) give all non-linearity of the softmax function.

2.8 Conclusions from literature

There are many studies where GA, RNN and KNN (benchmark model) are used for different purposes. In this study, the use of GA, RNN and KNN on renewable energy forecasting has been discussed under the literature review. This dissertation seeks to investigate the application of GA, RNN and

the benchmark model in renewable energy forecasting in South Africa. To the author's knowledge, this is the first dissertation to implement genetic algorithms using South African data to forecast solar irradiance.

Chapter 3

Methodology

3.1 Introduction

This chapter describes the methodologies that this study will use to forecast global solar irradiance (GHI) at the Vuwani radiometric station in Limpopo province of South Africa (SA). Recurrent neural network (RNN), genetic algorithm (GA) and benchmark model which is a k-nearest neighbour (KNN) method will be discussed in this chapter. In the methodology, GHI is represented by y_t , x_{1i} represents air temperature, x_{2i} barometric pressure, x_{3i} total rainfall, x_{4i} relative humidity, x_{5i} wind direction, x_{6i} wind direction standard deviation, x_{7i} wind speed. We will also present techniques that will be used in the selection of the methods and in checking the forecast accuracy.

3.2 Genetic algorithm

The GAs follows the theory of Darwin of the survival of the fittest and links the natural selection and genetics experiment with random operators to form searching mechanisms for improved results ([Rajeev and Krishnamoorthy](#),

1998). It is provided in the previous chapter that, in the late 1960s, Holland proposed GAs and Goldberg (1989) implemented the algorithm for the first time to address engineering optimisation problems. Goldberg (1989) later illustrated the utility of GAs for structural optimisation by solving the classic ten-bar truss issue. A small group of researchers then implemented the algorithm to explore its use in structural optimisation. In forecasting the evolution of one and two-dimensional nonlinear systems over time, Darwin (1960) showed magnificent results.

The genetic algorithm that is going to be used in this study proceeds as follows:

- Initially, the time sequence $\{x(t_i), t_i = 1, \dots, N\}$ with the sequence of equations for candidate (population) for $P(\cdot)$ is given at random.
- Normally, such equations are of the form $x(t) = (A \otimes B) \otimes (C \otimes D)$, with the parameter variables A, B, C and D are the earlier state variables ($x(t - \tau), x(t - 2\tau), \dots, x(t - m\tau)$), with τ being the discrete time unit), or the actual values constant, so that the \otimes symbol is one of the known four basic arithmetic operators (+, -, \times and \div).
- Other mathematical operators may be achievable, but increasing the number of accessible operators makes the functional optimisation method difficult.
- A parameter that measures the results of equation strings in a training

set is its fitness to the information described by:

$$R^2 = 1 - \frac{\Theta}{\sigma}, \quad (3.2.1)$$

where σ is the variance and where Θ is implied by:

$$\Theta^2 = \sum_{t=m+1}^N (x(t) - p(x(t - \tau), x(t - 2\tau), \dots, x(t - m\tau)))^2. \quad (3.2.2)$$

- Values of R^2 close to one are highly accurate forecasts, whereas low positive or negative values indicate the algorithm's weak forecast capability.
- The equation strings with a higher number of R^2 would be taken to exchange the character string parts among them (reproduction and crossover) in other case discarding the few suitable individuals.
- The offspring is more difficult to produce than the parents.
- The total number of characters throughout the equation strings is upper bounded to avoid the generation of offspring with unreasonable frequency.
- A small proportion of the strong fundamental components, individual operators and variables, of the strings of the equation are progressively mutated at random.
- The process is performed several times to enhance the fitness of the developing population and the empirical method to estimate function $p(\cdot)$ is obtained at the edge of the evolutionary process.

3.3 Recurrent neural networks

Mathematically the RNNs are defined as:

$$y_j(t+1) = \varphi\left(\sum_{i=1}^{m+n} w_{ji}z_i(t)\right), \quad (3.3.1)$$

$$z_i(t) = \begin{cases} y_i(t) & (i \leq n) \\ u_{i-n} & (i > n), \end{cases} \quad (3.3.2)$$

where m stand for the proportion of inputs, n for the proportion of hidden and output neurons, φ is for the arbitrary differential component, generally a sigmoid function, y_j determines the output of the j^{th} neuron and w_{ji} the relationship between the i^{th} and the j^{th} neurons. For simplicity, the external inputs u_i and recurrent inputs y_i are represented as z_i ([Garcia-Pedrero and Gomez-Gil, 2010](#)).

RNNs defined as the networks with loops in them that enable the continuity of data ([Ugurlu et al., 2018](#)). They are used for modeling data reliant on time. The information is supplied one after the other to a network and at a single point, the network nodes save their state and then use it to alert the following step. Not the same way as multilayer perceptron (MLP), RNNs use input data temporally, making them more suitable for time series data. An RNN realises the ability through recurrent neuronal connections. A basic equation which provided an input sequence $x = (x_1, x_2, \dots, x_T)$ for the RNN hidden state h_t is:

$$h_t = \begin{cases} 0, & \text{if } (t = 0) \\ \phi(h_{t-1}, x_t) & \text{otherwise,} \end{cases} \quad (3.3.3)$$

where the Φ function is non-linear. Recurrent hidden state update is realised as follows:

$$h_t = g(wx_t + uh_{t-1}), \quad (3.3.4)$$

where g is a function of the hyperbolic tangent. Generally, this generic environment of RNN with no neurons often suffers from gradient issues that are going away.

3.4 K-nearest neighbour

The KNN algorithm is a direct, simple, supervised machine learning algorithm that is used to solve problems of both classification and regression ([Horton and Nakai, 1997](#)). The supervised machine learning algorithm (opposed to an unsupervised machine learning algorithm) is an algorithm that depends on the labeled input data when training the function that will be used to produce acceptable results when the data that is not labeled is provided. Hence, supervised machine learning algorithms are also used to address regression or classification problems.

The KNN algorithm will be performed as follows:

1. load the solar data from USAid Venda,
2. initialise K to the specified number of neighbours,
3. for every example in the data,

- calculate the distance between a query example and the current example,
 - add the distance and an example index to an ordered set,
4. order the set of distances and indices by distances (in ascending order) from the smallest to the largest,
 5. choose the first K entries in the list which has been sorted,
 6. get the labels for K entries which have been chosen,
 7. returns the K label mean if a regression occurs,
 8. returns the K label mode if it is classified.

3.5 Data and features

The GA, RNN and the benchmark model which is the k-nearest neighbour (KNN) models will be implemented using global horizontal irradiance (GHI) data from the Vuwani radiometric station in the Limpopo province of SA.

Data

The data used in this dissertation was obtained from the USAid Venda radiometric station measured at one-minute intervals accessible at <https://sauran.ac.za/>. Figure 3.1 shows the pyranometer at the USAid Venda radiometric station which is on an inside enclosure in Vuwani in the Limpopo province.



Figure 3.1: Picture showing the location of the Vuwani radiometric station (USAid Venda). Source: <https://sauran.ac.za/>

Features

Models developed in this chapter will use GHI as the response variable and the predictor variables are air temperature, barometric pressure, total rainfall, relative humidity, wind direction, wind direction standard deviation and wind speed. The data will be normalised to $[0,1]$ by using the min-max function. Min-max normalisation has become one of the common methods of optimising data. Where the minimum value of the function is converted into 0 in every feature, the maximum value into 1 in every feature and the remaining value is a decimal value between 0 and 1.

3.6 Variable selection

In the variable selection, we involve the selection of function variables that determines the target variables when limiting the number of variables from the system. The variable selection framework plays an important role in terms of avoiding overfitting, promoting analysis of the patterns and computational time reduction. We have various variable selection methods but in this mini-dissertation, we decided to use LASSO (least absolute shrinkage and selection operator) (Bien et al., 2013). We assume a regression model with the response variable Y and predictors X_1, \dots, X_p with pairwise interactions among these predictors.

3.7 Prediction intervals

A prediction interval (PI) by its nature is a useful tool for modeling uncertainty. It is composed of lower and upper boundaries which cover the unidentified target value of the future value with any probability $(1 - \alpha)\%$ called confidence level. They are more appropriate and more valuable information than point forecasts for decision-makers (Quan et al., 2014). The width of the prediction interval (PIW) is given as:

$$PIW_t = UL_t - LL_t, \quad (3.7.1)$$

where UL_t and LL_t are the upper and lower bounds respectively. In this study, probability density plots, box and whisker plots used to find the model which yields narrower PIW.

Evaluation of prediction intervals

For a prediction interval (PI) with nominal confidence (PINC) of $(1 - a)100\%$ it is formulated as the probability where $\hat{y}_{t,\tau}$ belonging to the predictive interval (LL_t, UL_t) . We provide PINC as follows:

$$\text{PINC} = P(\hat{y}_{t,\tau} \in (UL_t, LL_t) = (1 - a)100\%). \quad (3.7.2)$$

This study uses the prediction interval normalised average width (PINAW) and the prediction interval coverage probability (PICP). The PICP is described by (Quan et al., 2014):

$$\text{PICP} = \frac{1}{m} \sum_{t=1}^m I_t, \quad (3.7.3)$$

with m being the number of the forecasts and I being the binary variables given by:

$$I_t = \begin{cases} 1, & \text{if } y_t \in (UL_t, LL_t) \\ 0, & \text{if otherwise.} \end{cases} \quad (3.7.4)$$

PINAW is another measure used to assess the accuracy of forecast intervals and is provided as (Quan et al., 2014):

$$\text{PINAW} = \frac{1}{m(\max(y_t) - \min(y_t))} \sum_{t=1}^m (UL_t - LL_t). \quad (3.7.5)$$

3.8 Performance measures

The use of performance measures plays an essential role in the evaluation of forecasts. So in the performance measures, we will look at mean, variance,

skewness and kurtosis which are also the underlying descriptive measures.

Skewness

The skewness measures a distribution's degree of asymmetry around the mean. In other words, skewness tells more about the variation path of the information set: in positive skewness, it informs us that the distribution tail is more stretched on the side above the mean; in skewness, it means that the distribution tail extends to a side under the mean; skewness of the ordinary distribution is zero. The conventional definition is:

$$Skew(x_1, \dots, x_n) = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{\sigma} \right)^3, \quad (3.8.1)$$

where $\sigma = \sigma(x_1, \dots, x_n) = \sqrt{Var(x_1, \dots, x_n)}$ is the standard deviation of the distribution.

Kurtosis

Kurtosis measures the relative peakness or flatness of the distribution: positive kurtosis reveals a peak distribution (i.e. leptokurtic); negative kurtosis implies a flat distribution (i.e. platykurtic); kurtosis of a confidence interval (i.e. mesokurtic) is zero. We can define the kurtosis as follows:

$$kurt(x_1, \dots, x_n) = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{\sigma} \right)^4, \quad (3.8.2)$$

where the kurtosis of a normal distribution is 3.

3.8.1 Mean absolute error, Mean square error, Root mean square error, Relative MAE, Relative RMSE

This mini-dissertation used Mean absolute error (MAE), Mean square error (MSE), Root mean square error (RMSE), Relative MAE (rMAE) and Relative RMSE (rRMSE) to evaluate the performance of the models. We defined them as follows:

$$\text{MAE} = \frac{1}{n} \sum_{t=1}^n |y_t - \hat{y}_t|, \quad (3.8.3)$$

$$\text{rMAE} = \frac{1}{n} \sum_{t=1}^n \left[\frac{\hat{y}_t - y_t}{y_t} \right], \quad (3.8.4)$$

$$\text{MSE} = \frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2, \quad (3.8.5)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^n (y_t - \hat{y}_t)^2}{n}}, \quad (3.8.6)$$

$$\text{rRMSE} = \sqrt{\frac{1}{n} \sum_{k=1}^p \left[\frac{\hat{y}_k - y_k}{y_k} \right]^2}, \quad (3.8.7)$$

in which y_t and \hat{y}_t are the actual and predicted values, \bar{y}_t is the mean value of y_t , $t = 1, \dots, n$, k is the dummy variable time, n is the number of data elements. The smaller value error is estimated closer to the true values.

3.8.2 Pinball loss function

The pinball loss function is a metric that is used to measure the precision of the quantile forecast. Compared to the typical forecasts where the objective

is to have a forecast that is similar as possible to the observed values, the situation is biased whenever it gets to the quantile forecasts (Gergo et al., 2016). Hence the naive contrast found against the forecasts is not sufficient. The component of the pinball loss returns a value that can be defined as the accuracy of the forecast of the quantile model.

Mathematically the pinball loss is defined as:

$$\mathbf{L}_\tau = \begin{cases} (y - z)\tau & \text{if } y \leq z \\ (z - y)(1 - \tau) & \text{if } z > y, \end{cases}$$

where τ is the target quantile, y the real value and z being the quantile forecast, so \mathbf{L}_τ is the pinball loss function. The most important findings associated with the role of the pinball loss is that the lower the loss of the pinball, the more accurate the quantile forecast.

3.8.3 Diebold-Mariano test

The Diebold-Mariano test assumes the test's degree of significance is = 0.05. Since this is a two-tailed test 0.05 so that it must be divided into upper tail 0.025 and lower tail 0.025. The z-value equivalent to -0.025 is -1.96, which is a lower critical z-value. So that the upper value 1-0.025=0.975, which is the z-value of 1.96. The DM test rejects the null hypothesis without a difference if the measured DM statistic falls outside the -1.96 to 1.96 range.

3.8.4 Murphy diagram

It is built on the concept that “if something wrong can happen, it’s going wrong”. It is close to other methods of analysing such as fault trees, as they evaluate errors based on the possible causes of such errors. The murphy diagram works by comparing the mean forecasts. Hence, the mean of the forecast distribution is acquired by minimising the squared error loss function, $\mathbf{S}(x, y) = (x - y)^2$ where x be the point forecast and y be the actual observation (Werner et al., 2015). With the equation:

$$E(Y) = \underset{x}{\operatorname{argmin}} \mathbf{S}(x, y), \quad (3.8.8)$$

it is given that any scoring function meeting the constraint can be defined as:

$$\mathbf{S}(x, y) = \int_{-\infty}^{\infty} \mathbf{S}_{\theta}(x, y) d\mathbf{H}(\theta) \quad (3.8.9)$$

with \mathbf{H} be non-negative indicator and

$$\mathbf{S}_{\theta}(x, y) = \begin{cases} |y - \theta| & \text{if } \min(x, y) \leq \theta < \max(x, y) \\ 0 & \text{otherwise.} \end{cases}$$

Various \mathbf{H} assessments offer different scoring functions, but for all such scoring functions, $\mathbf{S}_{\theta}(x, y)$ is the same.

If the point forecasts for n events is given, then we can be able to get the average value of $\mathbf{S}_{\theta(x,y)}$ for each θ :

$$s(\theta) = \frac{1}{n} \sum_{i=1}^n \mathbf{S}_{\theta}(x_i, y_i), \quad (3.8.10)$$

and plot this as a θ function. This is what (Werner et al., 2015) call the “Murphy diagram”. The same approach can be applied for quantile and expertise forecasts.

3.9 Forecast error distribution

For the characterisation of forecast dispersion, the measures of central (mean and median) tendency along with the lower and upper quartiles of the residuals are used. They keep providing appropriate distribution information. In assessing distribution we also look at skewness and kurtosis which is defined in section 3.8 above. The summary statistics for the remaining best models will be obtained to highlight the model with the best renewable energy forecasting.

3.10 Percentage improvement

Percentage improvement defines the best model improvement relative to other models (Ravele, 2018). Improvement of the proportion between the best model and the other models is defined by:

$$Improvement(\%) = \left(1 - \frac{MAE_{bestmodel}}{MAE_{othermodel}}\right) \times 100\%, \quad (3.10.1)$$

where $MAE_{bestmodel}$ is the mean absolute error and MAE is defined in the previous section and $MAE_{othermodel}$ is the mean absolute error for other models.

3.11 Conclusion

This section has represented the techniques, methods, and steps to be applied when forecasting renewable energy. The techniques used to analyse the data include time-series data techniques, GA, RNN, KNN, and performance measures.

3.12 Implementation

The software packages used for data analysis in this study are Python and R. The GA and RNN algorithms are implemented in this study using the Keras deep learning package (<https://keras.io/>). KNN was build using the Scikit-learn python package.

Chapter 4

Data analysis and discussion

4.1 Introduction

This chapter presents the data analysis using the algorithms discussed in Chapter 3. The software packages that are used for data analysis in this study are Python (Faouzi and Janati, 2020) and R (Racine, 2012). Predictive performance of the algorithms (GA, RNN and KNN) are compared in this chapter for forecasting minutes solar irradiance where the K-nearest neighbour (KNN) algorithm is used as a benchmark algorithm.

4.2 The data

4.2.1 Exploratory data analysis

The frequency analysis of the day using the historic data of the period 04/01/2020 to 31/10/2020 are shown in the figures below.

From Figure 4.1 most of the long short term peak GHI occurs on the day between January 2020 and May 2020. Happening among these days because

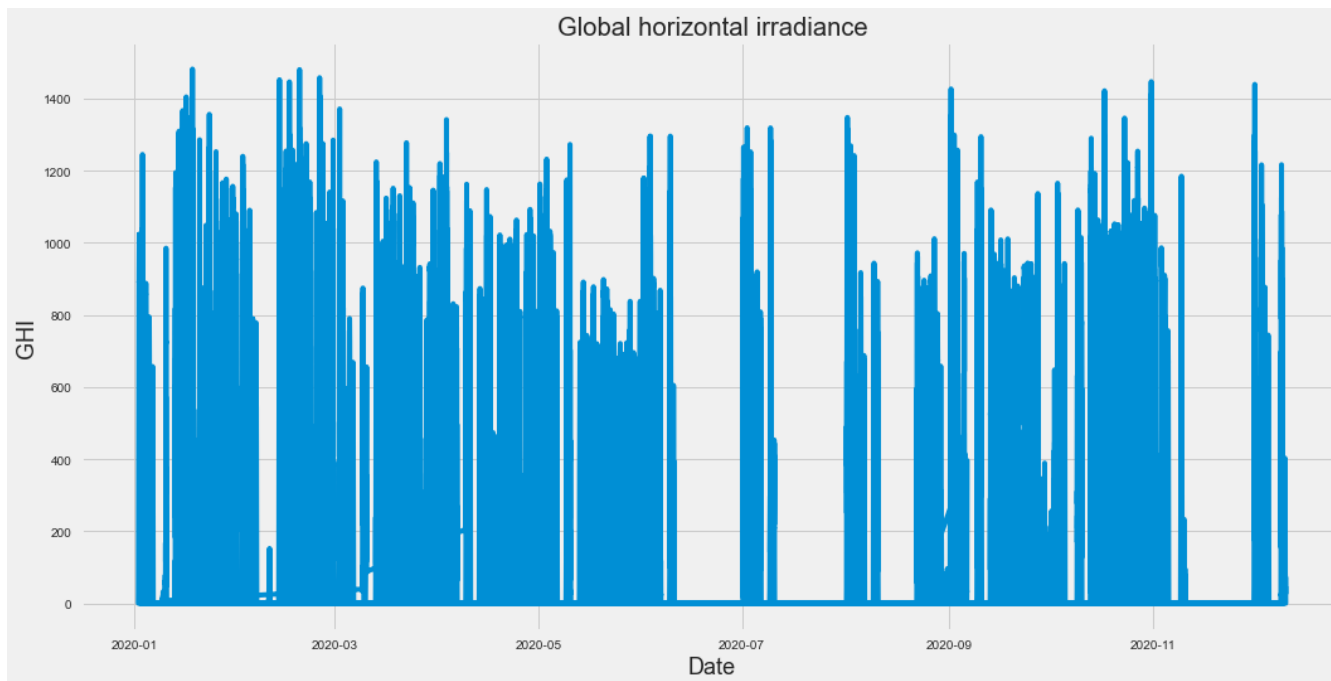


Figure 4.1: Plot of the long short term global horizontal irradiance peak for the period 04/01/2020 to 31/10/2020.

is where we experience low rainfall with a higher temperature and is where we usually have a long day with a higher temperature than the other days which are out of the higher GHI portion this usually happen in January and middle of February to the beginning of March where we start approaching a little of the winter season. We can also see from the graph that during winter June and July GHI fall due to the short time of radiations as we are in the middle of winter seasons.

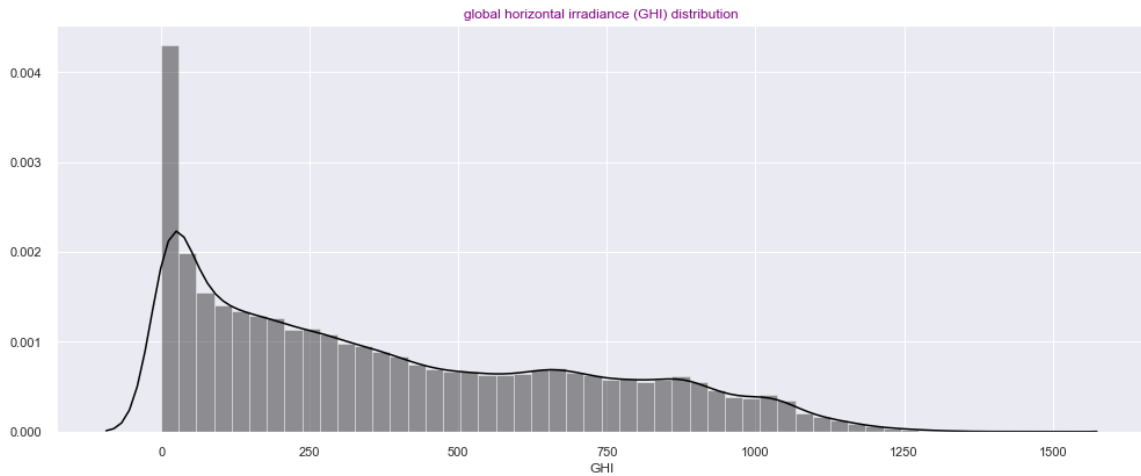


Figure 4.2: Histogram of global horizontal irradiance occurrences for the period 04/01/2020 to 31/10/2020.

A histogram with a line of best fit showing the distribution of the global horizontal irradiance is given in Figure 4.2. The histogram has a long tail at the right-hand side when you are looking at it which shows that the distribution of the global horizontal irradiance is positive and which again tells us that it is positively skewed. Time series consists of four components that consist significantly in the explanation of patterns of data. Figure 4.3 provides a pattern of the global horizontal irradiance and also shows that the distribution of the GHI is positively skewed. The plots show that the distribution of the GHI is skewed to the right.

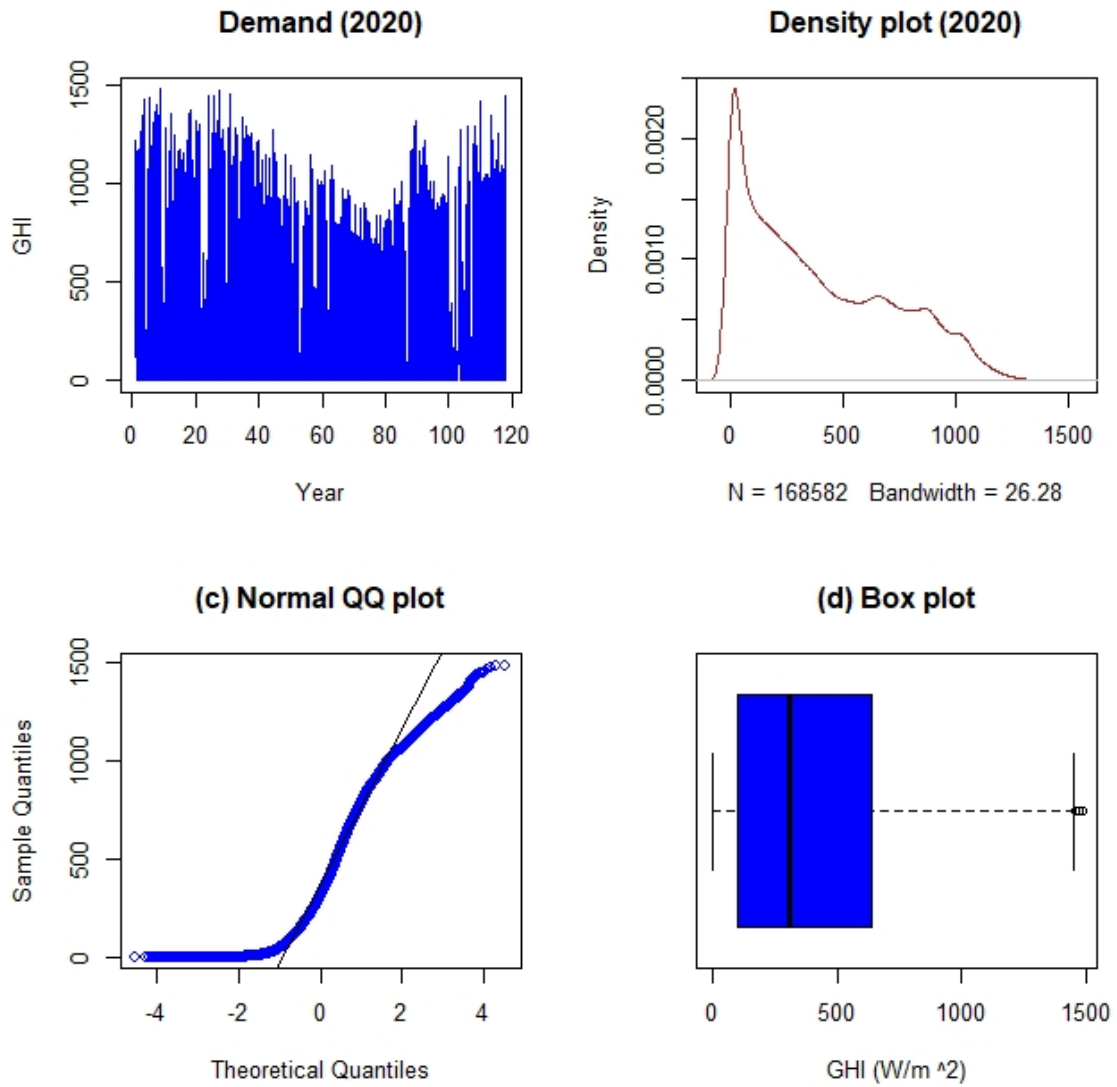


Figure 4.3: Time series plot of GHI, Density plot, Quantile-Quantile(qq) plot and Box and whisker plot.

Table 4.1: The descriptive statistics of the GHI measured in W/m^2 .

Min	Max	Median	Mean	st.Div	Skewness	Kurtosis
0.0032	1481.6300	307.6562	388.0439	324.1666	0.6130	-0.7568

Table 4.2: Weather variables and the ranges of their values.

Weather variables	Ranges
Temperature ($^{\circ}C$)	0.00-40.38
Barometric pressure (mbar)	932.45-981.00
Total rainfall (mm)	0.00-2.54
Relative humidity (%)	0.00-100.00
Wind direction (WD) ($^{\circ}$)	0.00-360.00
WD standard deviation ($^{\circ}$)	0.00-78.29
Wind speed (m/s)	0.00-11.35

Descriptive statistics quantitatively compile features of the collected data. Simple outlines about the sample are given by descriptive statistics. The sample outlines are the minimum, maximum, mean, median, standard deviation, skewness and kurtosis. The summary of descriptive statistics of minutes GHI for the sampling period 04/01/2020 to 31/10/2020 is given in Table 4.1 for the Vuwani radiometric station (USAid Venda). Like in Figure 4.2 the GHI distribution shows no normal distribution because of the skewness value of 0.6130 and kurtosis value of -0.7568 that is given in the Table 4.1, the skewness shows that it is skewed to the right-hand side and it is platykurtic.

In this dissertation, non-linear trend value extracted by fitting the cubic smoothing spline function provided by the equation:

$$\pi(t) = \sum_{t=0}^n (y_t - f(t))^2 + \lambda \int \{f''(t)\}^2 dt, \quad (4.2.1)$$

where λ is the parameter of the smoothing that is predicted using the generalised cross-validation (GCV) criterion. Figure 4.4 illustrates the cubic smoothing spline and non-linear trend fitted with the projected lambda value. The extraction of the non-linear trend used was to model solar irradiance.

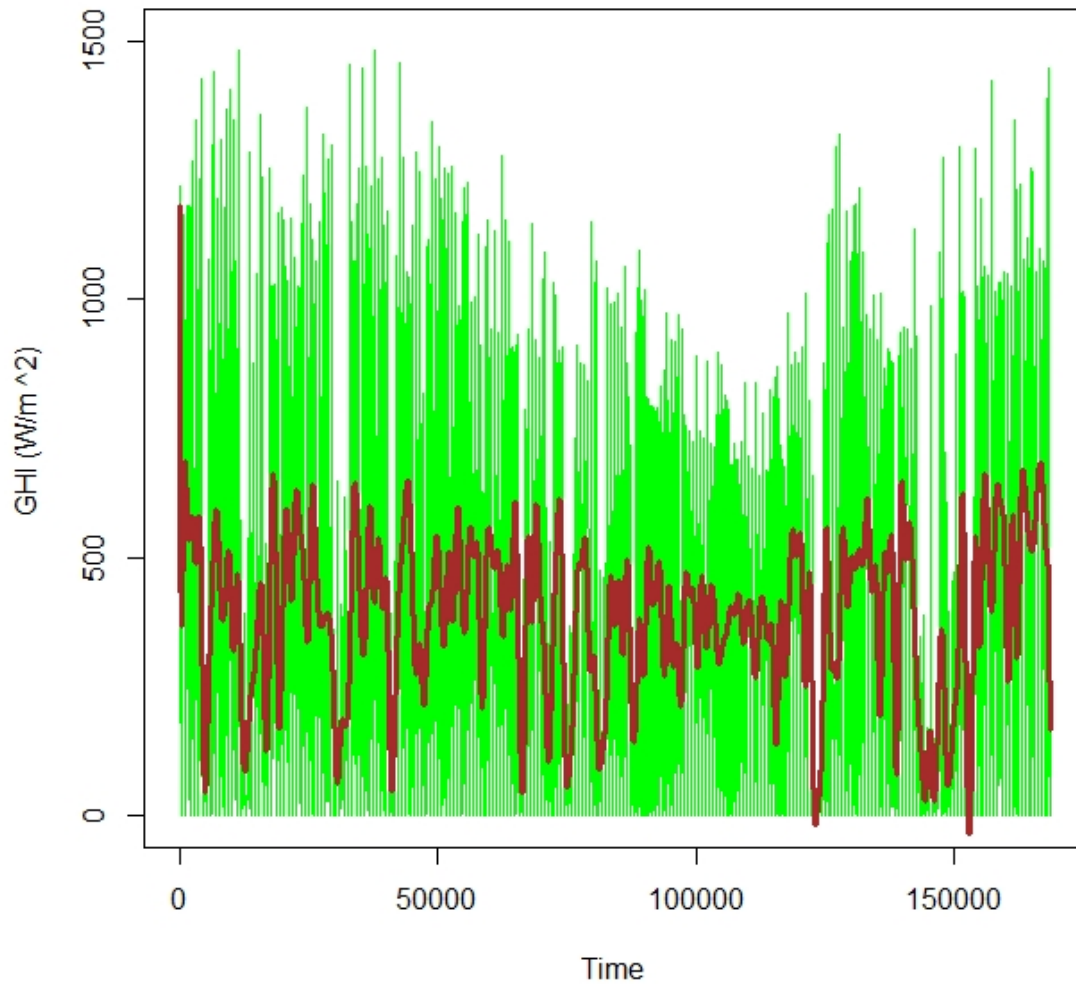


Figure 4.4: Plot of minutes global horizontal irradiance from 04/01/2020 to 31/10/2020 superimposed with a fitted cubic smoothing spline trend (non-linear).

4.3 Variable selection using LASSO

Weather variable is taken from USAid Venda. In the dissertation, variable selection is done using the LASSO to remain with a useful variable in forecasting minutes of solar irradiance. LASSO is considered as a technique of regression that is used in the selection of variables. LASSO uses the ℓ loss function penalty:

$$\hat{\beta}_{LASSO}(\lambda) = \arg \min \|\vec{y} - \mathbf{X}\hat{\beta}\|_2^2 + \lambda\|\hat{\beta}\|_1. \quad (4.3.1)$$

Table 4.3 shows parametric coefficients and the importance of the variables in forecasting minutes of solar radiation in USAid Venda assessed by LASSO. So based on the LASSO, all variables have significance in forecasting solar irradiance except total rainfall (Rain) which is having the least significance.

Table 4.3: Parametric coefficients.

Variables	Coefficients
(Intercept)	-2.158294×10^4
Temp	3.815809×10^1
RH	-1.326136×10^0
WD	1.645751×10^{-1}
Rain	0.000000
WS	1.866991×10^0
WD StdDev	7.173793×10^0
BP	2.215658×10^1

4.4 Machine learning models

In this section machine learning techniques that will be presented are genetic algorithm (GA), recurrent neural network (RNN) and the benchmark model

which is K-nearest neighbour (KNN). Like in other dissertations done previously performance measures such as relative mean absolute error (rMAE) and relative root mean square error (rRMSE) will be used to ease the process of predicting the best forecasting technique. The rRMSE and rMAE for the GA are found to be 5.96 and 5.17, for the RNN are found to be 7.54 and 4.49 and for the KNN are found to be 7.58 and 4.49. So, according to the results of rMAE on Table 4.4 the RNN is found to be the best forecasting technique of the global horizontal irradiance (GHI).

Table 4.4: Assessment of models.

	GA	RNN	KNN
RMSE	35.50	56.89	57.48
rRMSE	5.96	7.54	7.58
MAE	26.74	20.18	20.94
rMAE	5.17	4.49	4.58

Figure 4.5 shows box-plots for actual GHI and the model's forecasts which are GA, RNN and KNN for the period 04/01/2020 to 31/10/2020. The test set consists of 33716 observations. It shows how the models performed. Hence, in box-plots, it is easy to compare the models much more efficiently. GA has a much longer whisker than other models which can tell us that it varies more widely than the others in forecasting the GHI while KNN has the least whisker than the rest. Looking at the skew of the models we can realise that GA is also having a pretty skew even on either side of the median/mean. From the box of all models, we can clearly say they are positively skew as the median is at the 25% lower quantile side. We can also look at the outliers which represent the statistically different data points.

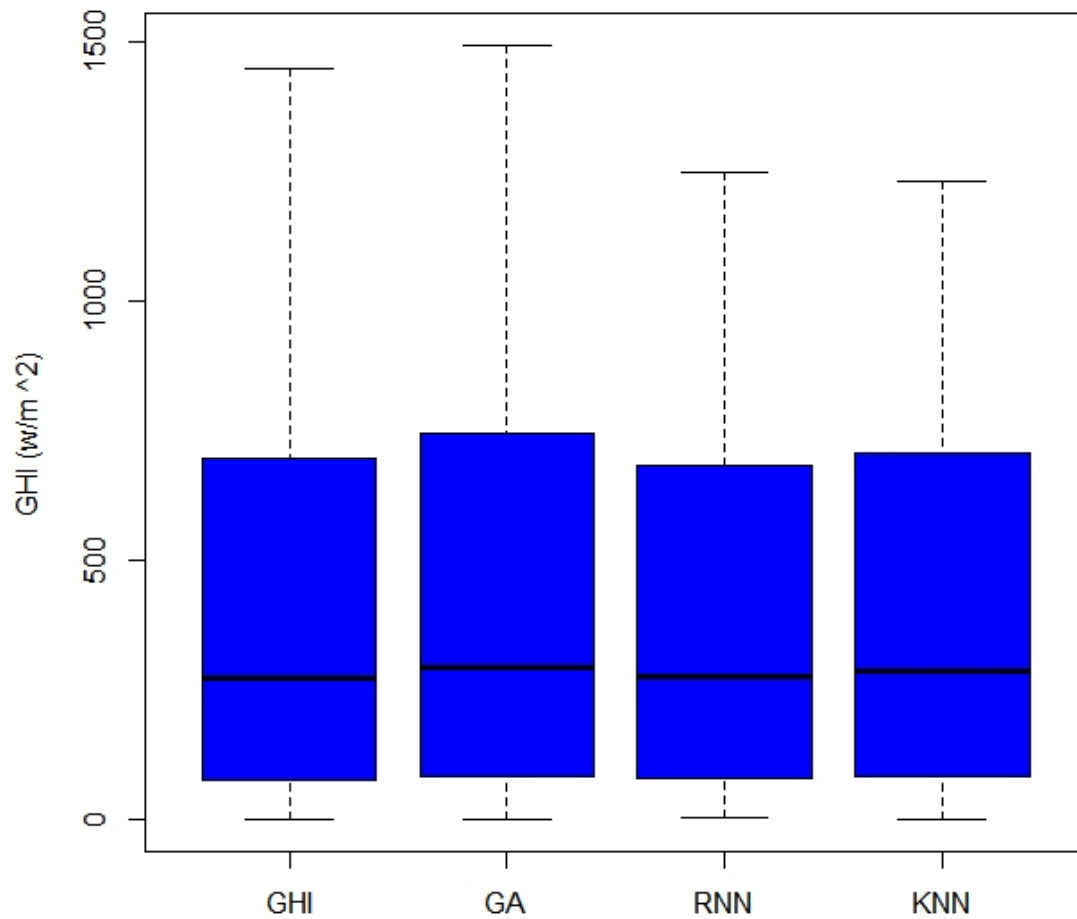


Figure 4.5: Boxplots of the actual values and the model's forecasts from left to right (a) **First one**: actual global horizontal irradiance (GHI) (b) **Second one**: genetic algorithm (GA) forecasts (c) **Third one**: recurrent neural network (RNN) forecasts and (d) **Fourth one**: k-nearest neighbour (KNN) forecasts.

The density plot of the actual GHI (solid lines) and model's forecasts (dashed lines) are shown in Figure 4.6. The forecasts from the models appear to be different from the actual observations with RNN being the one that is closer to the actual observations we can see this at actual observation 900 where the GA and KNN showing a gap between the two lines. The gap between the two lines is seen in Figure 4.6 (d) **Bottom right panel** where the green dashed line is the one showing a great gap followed by the black dashed line. This also validates the results we got from Table 4.4 that RNN is the best forecasting technique.

Figure 4.7 is the graphical plot of the actual GHI values and their forecast outcomes for the models GA, RNN and KNN, respectively. These plots show the estimations of the forecasts between each of the models and the actual GHI. Looking at the forecasts between the graphs they are having different estimations with the actual GHI. The (a) **Top left panel** of the GA forecasts overestimate the GHI with (b) **Top right panel** looks to be the one which correctly estimates the GHI and (c) **Bottom left panel** underestimates the forecasts of GHI. This tells us that the RNN model is the one that correctly forecasts the GHI with GA overestimate the forecasts of GHI and KNN underestimate the forecasts of GHI. Figure 4.7 (d) **Bottom right panel** shows the estimations of the forecasts between the RNN, GA and actual GHI.

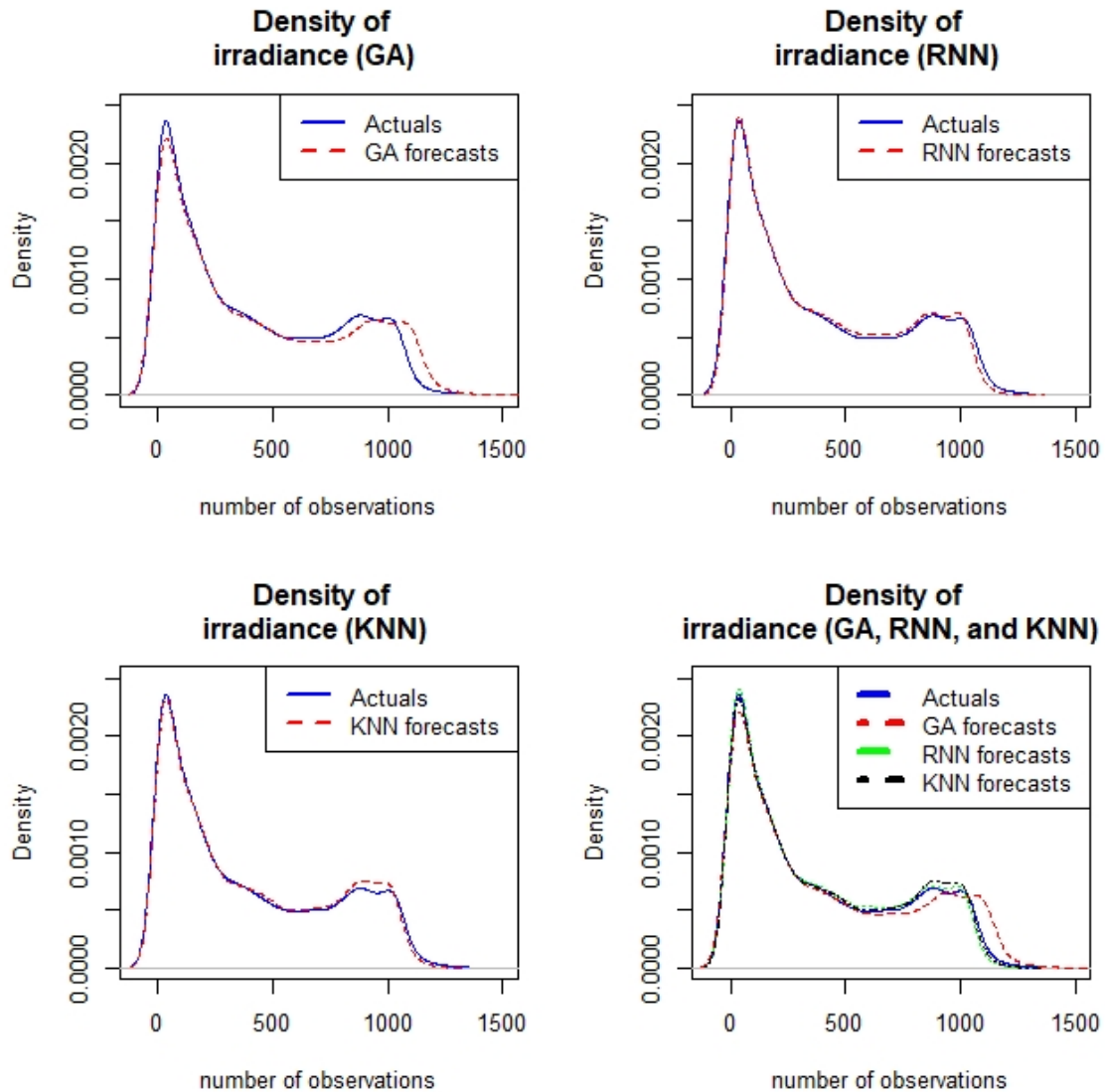


Figure 4.6: Density plots of the actual GHI (solid lines) and model's forecasts (dashed lines) where (a) **Top left panel:** Actual GHI and forecasts of GA (b) **Top right panel:** Actual GHI and forecasts of RNN (c) **Bottom left panel:** Actual GHI and forecasts of KNN and (d) **Bottom right:** Actual GHI and forecasts of RNN, GA and KNN where the actual GHI and forecasts of GA are shown by solid lines.

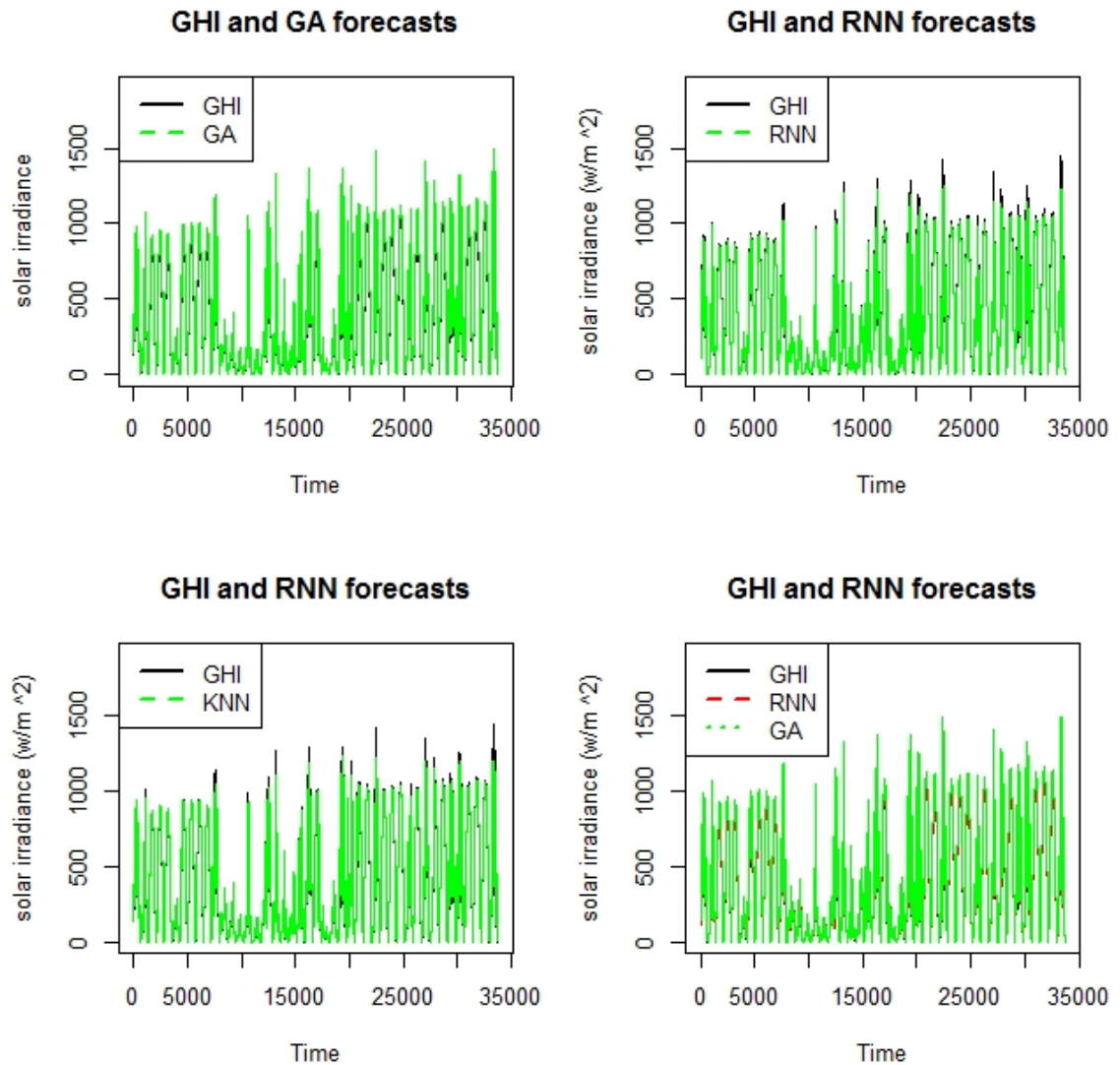


Figure 4.7: Graphical plot of the actual GHI (solid line) and model's forecasts (dashed line) (a) **Top left panel:** Actual GHI and GA forecasts (b) **Top right panel:** Actual GHI and RNN forecasts (c) **Bottom left panel:** Actual GHI and KNN forecasts and (d) **Bottom right panel:** Actual GHI and RNN and GA forecasts.

4.5 Combination of the forecasts

This section provides outcomes of the forecast blend of forecasts for machine learning techniques. Convex combination and QRA are the forecast combination techniques used in this section.

4.5.1 Convex combination

Combining forecasts lead to improved accuracy. The R package which is known as ‘opera’ is used to compile forecasts from various regression based time series fitted in this dissertation (Timmermann, 2006). In this case, models developed are known as experts. To merge the expert forecasts based on their previous outcomes, the package proposes several flexible and robust methods. In this dissertation, the opera package calculates the weights when combining forecasts. The concept of convex combination involves measuring the series of instant losses incurred by the expert prediction using the loss function.

The loss function may be centered on pinball loss, square and absolute. Table 4.5 summarises the weight allocation for the forecast combination models with Square (Mix 1), Pinball (Mix 2) and Absolute (Mix 3) and the accuracy measures RMSE and MAE. Looking at the accuracy measure we can see that Mix 1 and Mix 3 have equal least RMSE (43.0439) and MAE (14.00380) values compared to Mix 2.

Figure 4.8 shows the average loss suffered by the models when the y-axis is the pinball loss. From Figure 4.8 the convex combination model is the

best forecasting model followed by the Uniform, RNN, KNN and GA as well where RNN, KNN and GA are the forecasts.

Table 4.5: Evaluation of the forecasts models combination.

Models (experts)			
	GA	KNN	RNN
Square (Mix 1)	0.699	0.000	0.301
Pinball (Mix 2)	0.241	$4.35e - 21$	0.759
Absolute (Mix 3)	0.241	$3.7e - 21$	0.759
Accuracy measures			
	Mix 1	Mix 2	Mix 3
RMSE	43.04390	28.49575	43.04939
MAE	14.00380	19.57792	14.00380

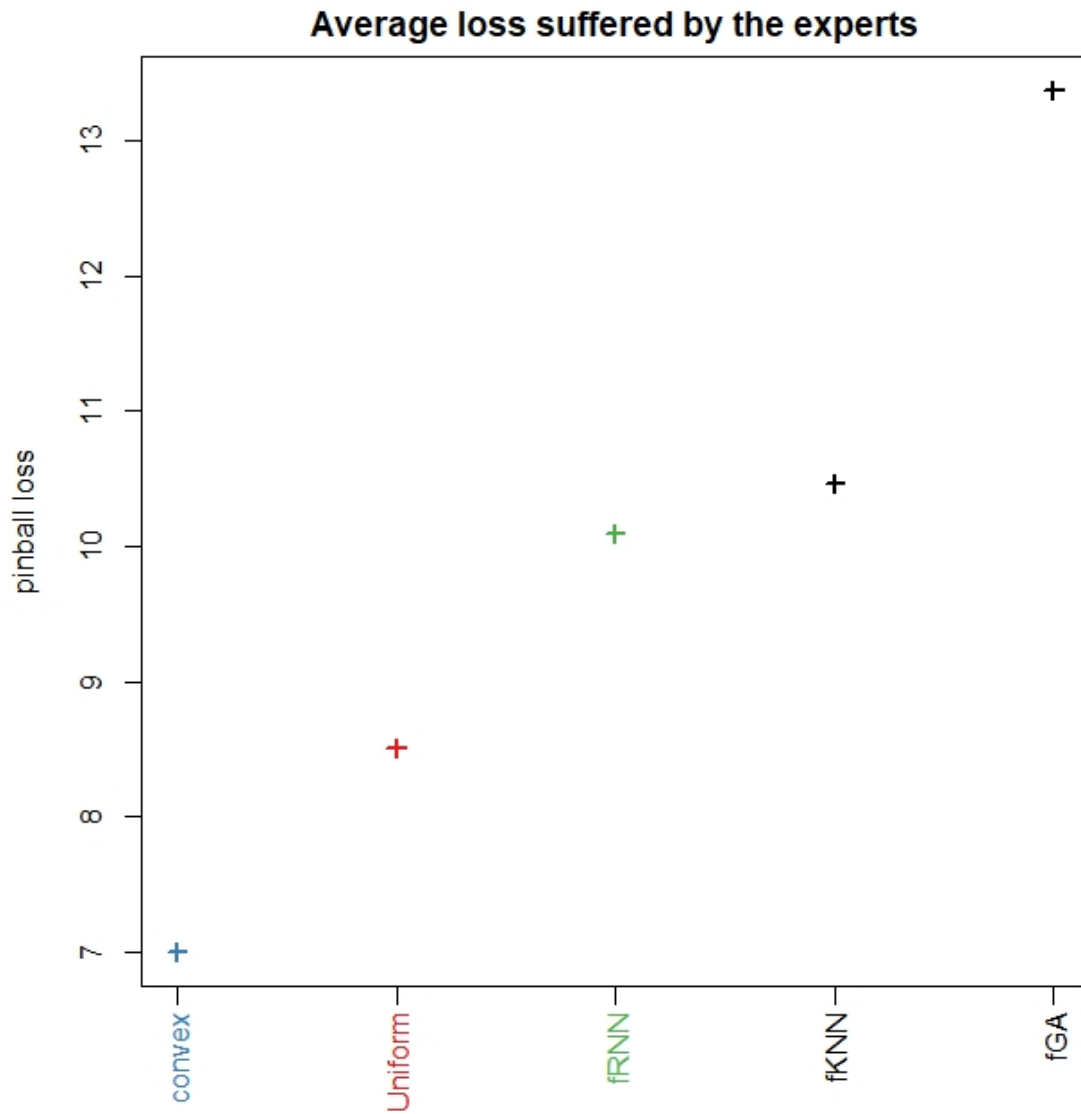


Figure 4.8: This plot shows the Average loss suffered by the models.

4.5.2 Quantile regression averaging

QRA is a forecast combination techniques to the computation of PI. It includes the application of quantile regression to a limited number of individual forecast models or expert point forecasts. From a functional point of view, QRA is a promising alternative as it enables the current progress of point prediction to be useful. In this case, GA, RNN and KNN models are joined based on QRA. QRA is given by:

$$y_{t,\tau}(QRA) = \beta_0 + \beta_1 fM_1 + \beta_2 fM_2 + \beta_3 fM_3 + \varepsilon_t, \quad (4.5.1)$$

where fM_1 , fM_2 and fM_3 are the forecasts from GA, RNN and KNN.

Table 4.6 offers a description of the accuracy measures for all the machine learning methods together with the QRA method. Looking at the MAE, QRA is the best forecasting model than GA, RNN and KNN. Again, MAE shows an improvement after averaging the forecast. Going into the results of absolute loss suffered by the models referring to the pinball losses it also shows that the QRA is the best one as it shows the smallest pinball loss average value.

Combining both the forecasts from the machine learning techniques improved the GHI predictions. Figure 4.9 gives the plots of the QRA forecasts. By looking at the graphs we can discover that they are close to each other in both actual observations and the forecasts. Figure 4.10 displays forecast plots for the last day of the GHI in SA. These forecasts are considered to be relatively close to the actual observations, particularly on the QRA forecasts.

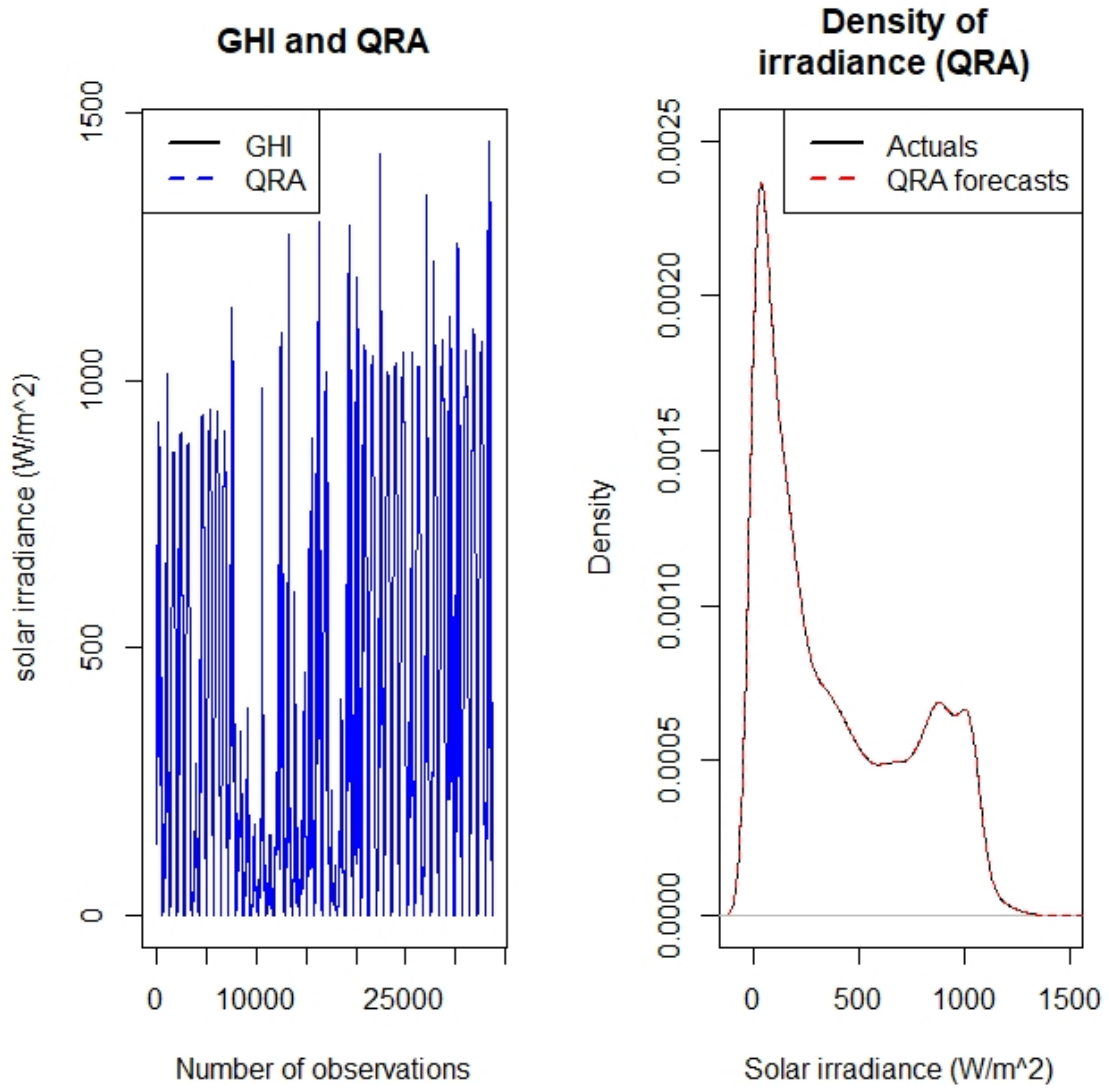


Figure 4.9: Graphical plot of the actual GHI (solid line), QRA forecasts (dashed line) and the density plot.

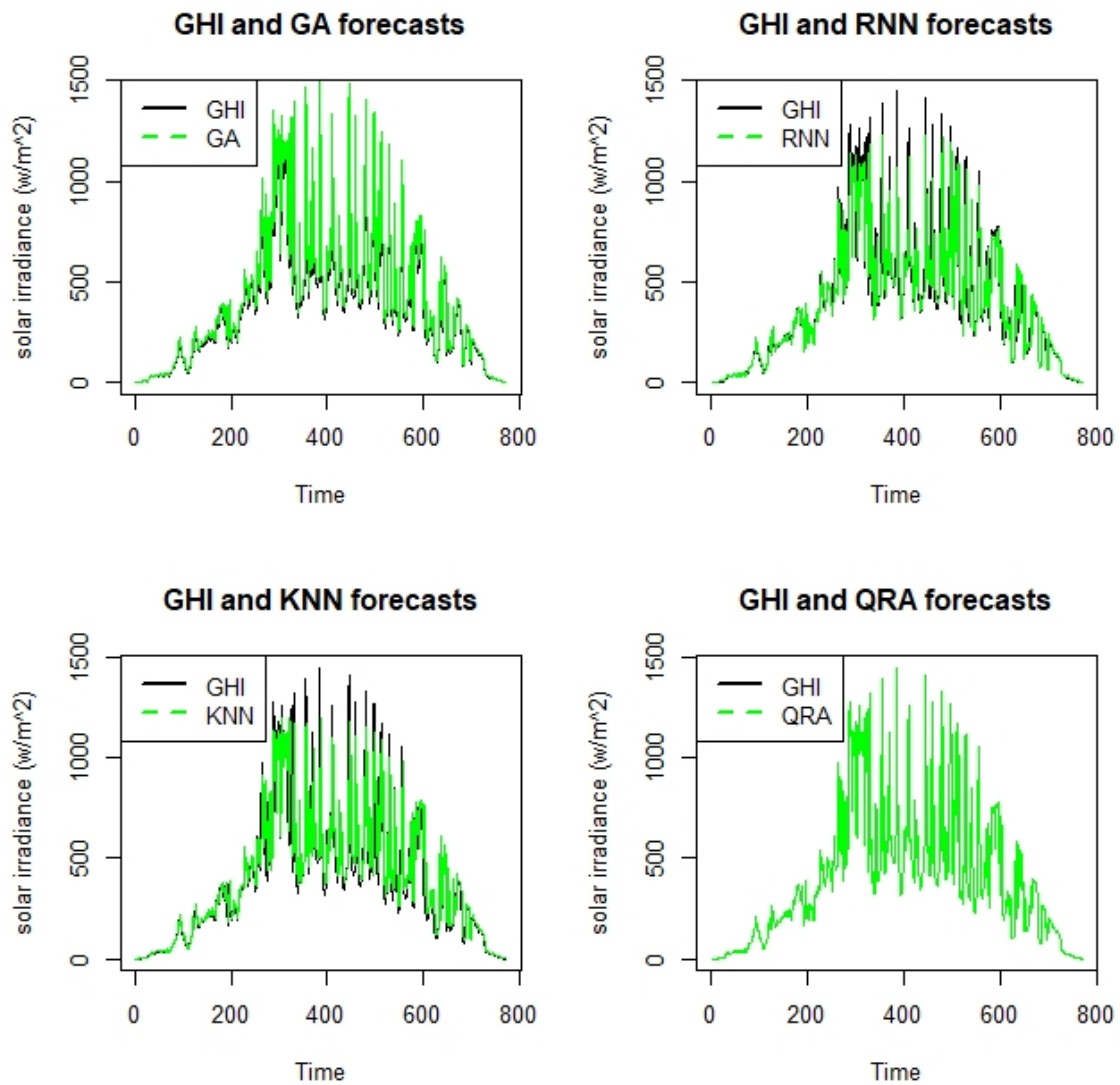


Figure 4.10: Graphical plot of the actual GHI (solid line) and the model's forecasts for the last day of the GHI.

Table 4.6: Machine learning and QRA model's comparative analysis.

Accuracy measures	GA	RNN	KNN	QRA
RMSE	35.50	56.89	57.48	< 0.0001
MAE	26.74	20.18	20.94	< 0.0001
Avg pinball loss	13.37	10.09	10.47	< 0.0001

4.6 Models Comparative Analysis

In this section, the estimation of the fitted models based on the empirical prediction intervals (PIs) with the forecast error distributions from each of the models was discussed.

4.6.1 Evaluation of prediction intervals

Table 4.7 provides descriptive statistics of the PIWs for models GA, RNN, KNN, Convex and QRA with such a confidence level of 95% for the PINC value. The skewness of all the models is near zero that tells us the distribution of the models is normal. With the values of the Kurtosis less than 3, it tells us that the distribution of the models is all platykurtic. The smallest standard deviation is of the KNN model that tells us that it has smaller PIWs compared to the other models. Figure 4.11 shows box plots of the prediction interval widths for the fitted models GA, RNN, KNN, Convex and QRA. KNN has a narrower PI compared to the GA, RNN, Convex and QRA.

A comparison of the best models using PICP, PINAW and PINAD is shown in Table 4.8. The PICP taken to be 95 percent level of confidence where it is valid for the entire models. QRA is having a higher PICP than all models with a PICP of 99.68%, which is valid because is greater than 95%. This

Table 4.7: PIWs for models comparison.

	Mean	Median	min	max	st.Dev.	Skewness	Kurtosis
GA	388.129	271.512	-0.724	1390.929	347.462	0.595	-1.044
RNN	337.577	241.119	-0.196	1095.501	298.812	0.568	-1.088
KNN	336.131	241.459	-0.527	1048.681	295.169	0.541	-1.138
Convex	357.612	230.176	-0.395	1132.315	317.467	0.584	-1.121
QRA	390.768	272.624	0.006	1446.843	349.910	0.601	-1.033

shows that the predictive intervals developed by QRA are more informative with valid PICP ($\geq 95\%$). RNN has a lower PINAW than all model models. This shows that a narrower PANAW is offered by RNN, so RNN is our best model when compared to the entire models. RNN has a lower PINAD than the other models, which also confirms that RNN is the best model. A model with the narrowest PINAW and smaller PINAD is recognised to be the best fitting model (Sun et al., 2017). So, RNN is the best fitting model.

Table 4.8: Comparative analysis of the best models with the confidence interval (CI) (PICP, PINAW and PINAD) at 95 percent.

Models	PICP	PINAW	PINAD
GA	98.00%	11.81%	0.07%
RNN	98.60%	11.47%	0.05%
KNN	98.12%	15.09%	0.08%
Convex	96.27%	12.34%	0.08%
QRA	99.68%	18.12%	0.06%

Figure 4.12 provides the density plots of widths of the PIs for the forecasting models GA, RNN, KNN, Convex and QRA. The density plots have a similar shape with different dimensions where GA and QRA have the widest PI.

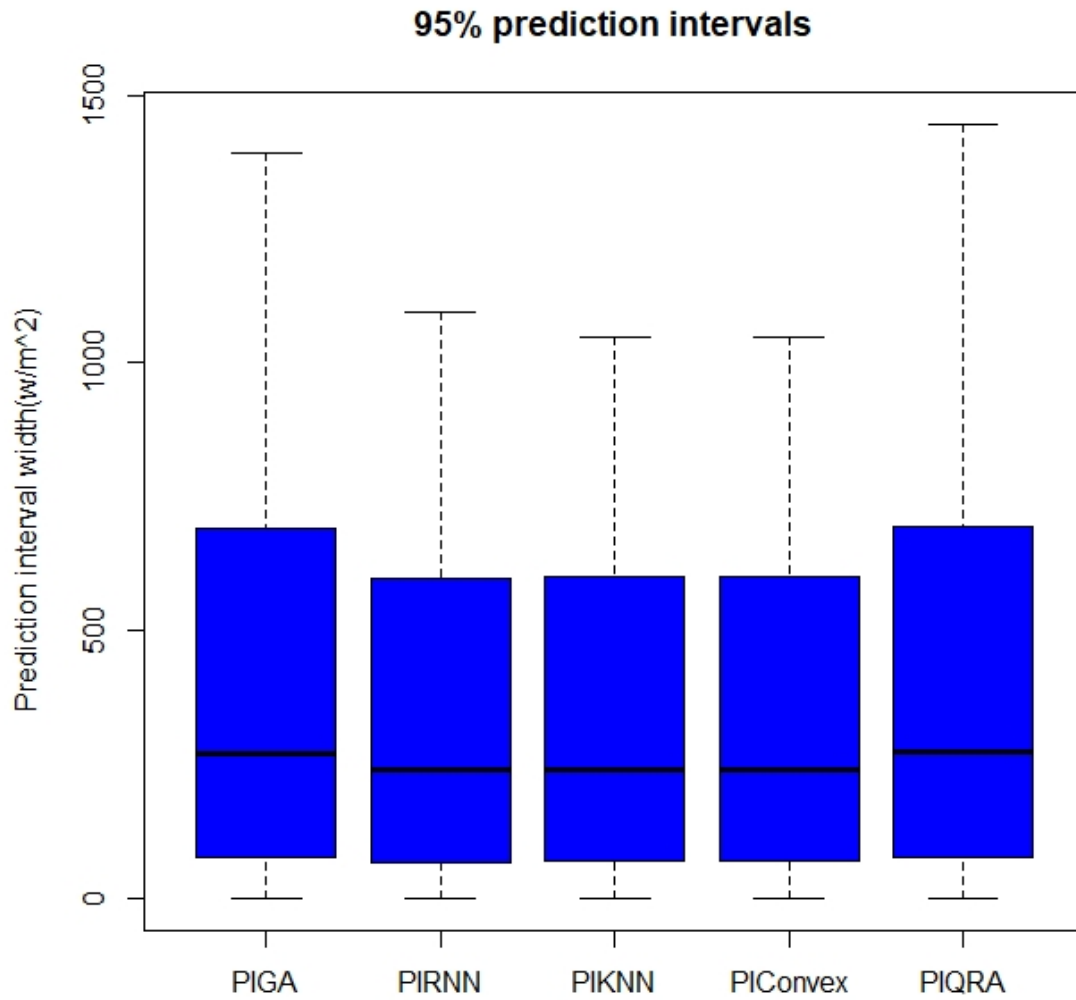


Figure 4.11: Box plots of the prediction interval widths (PIWs) for model GA, RNN, KNN, Convex and QRA.

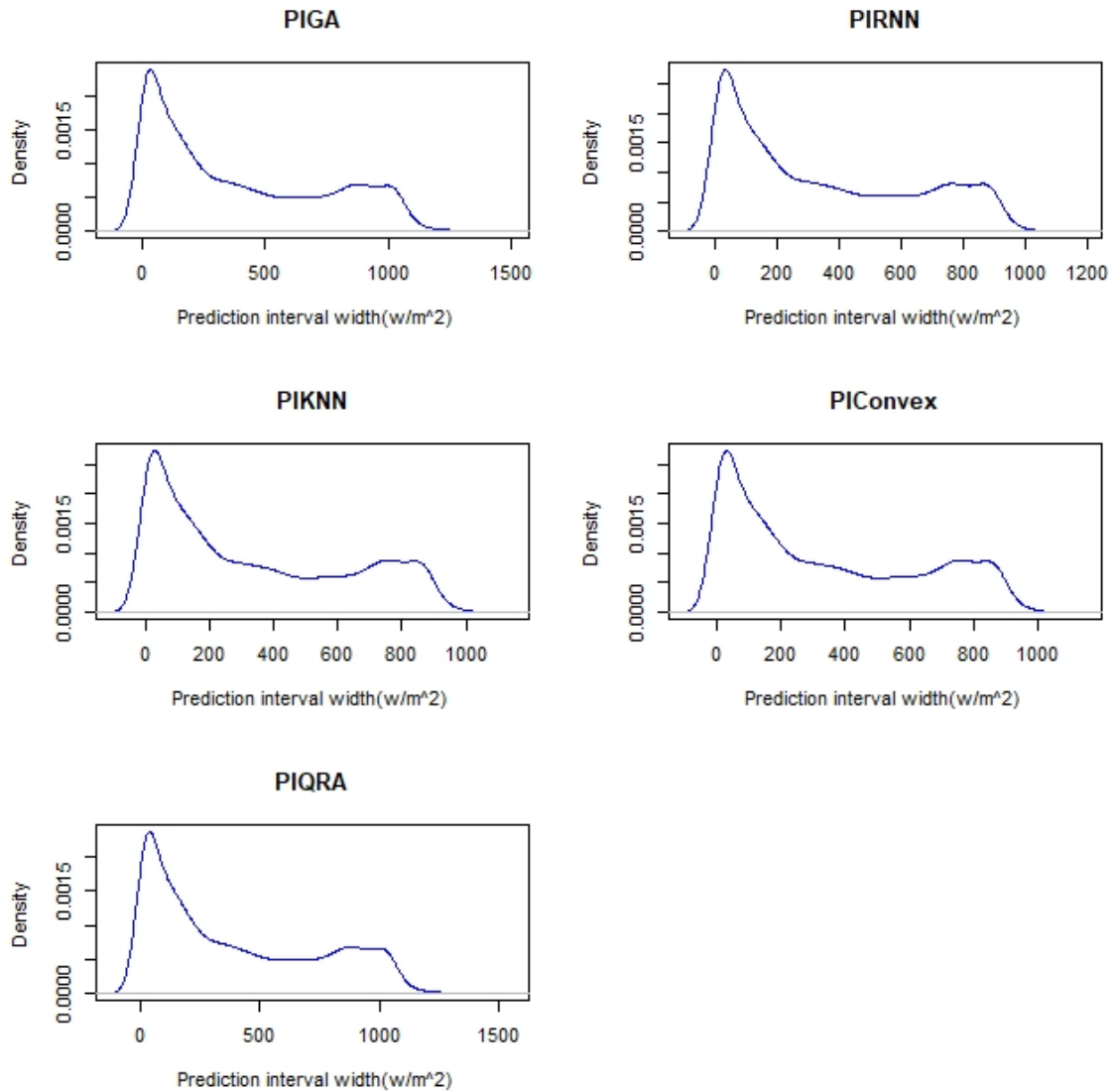


Figure 4.12: Density plots of the prediction interval widths for modes GA, RNN, KNN, Convex and QRA.

4.6.2 Residual analysis

Table 4.9 provides descriptive statistics of the residuals for models GA, RNN, KNN, Convex and QRA with such a confidence level of 95% for the PINC value. From the table, we can see that GA is the one with the smallest standard deviation compared to other models that tell us that it has a smaller error distribution and is the best model compared to others. For the GA, RNN, KNN and Convex models, the error distributions are normal because their skewness is close to zero, where the QRA is skewed to the right that tells us that it is positively skewed. The values for kurtosis are greater than 3 for four models and less than 3 for the GA model, showing that the distributions for the four models are leptokurtic and for the GA model is platykurtic.

Table 4.9: Residual comparison of the models.

	mean	median	min	max	StDev	Skewness	Kurtosis
GA	-26.741	-19.250	-83.121	-0.103	23.355	-0.541	-1.138
RNN	3.779	-0.240	-688.628	764.295	56.762	0.479	41.139
KNN	-5.435	-4.325	-747.010	796.949	57.220	0.077	43.475
Convex	-3.576	-3.017	-539.927	568.753	42.899	0.145	43.148
QRA	2.639	1.531	-44.413	57.499	4.329	1.043	25.391

Figure 4.13 is the box plots of the forecast errors for the entire fitted models ResGA, ResRNN, ResKNN, ResConvex and ResQRA where ResGA, ResRNN, ResKNN, ResConvex and ResQRA are the residuals from GA, RNN, KNN, Convex and QRA. ResQRA has a narrower error distribution compared to the other models, this tells us that ResQRA is considered to be the best model compared to all the models presented. Figure 4.14 gives the density plots of the forecast errors for both the ResGA, ResRNN, ResKNN,

ResConvex and ResQRA forecasting models. The plots are similar except for the forecast error of ResGA.

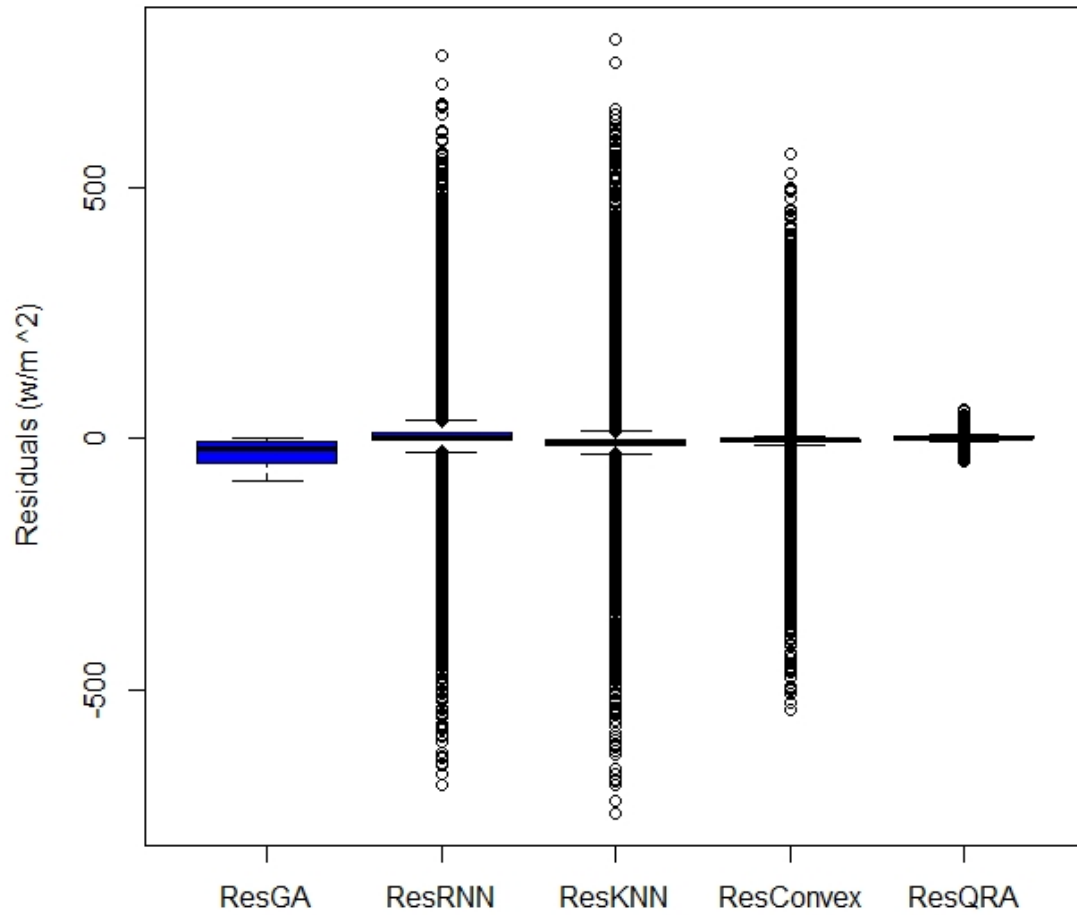


Figure 4.13: Box plots of the residuals from ResGA, ResRNN, ResKNN, ResConvex and ResQRA.

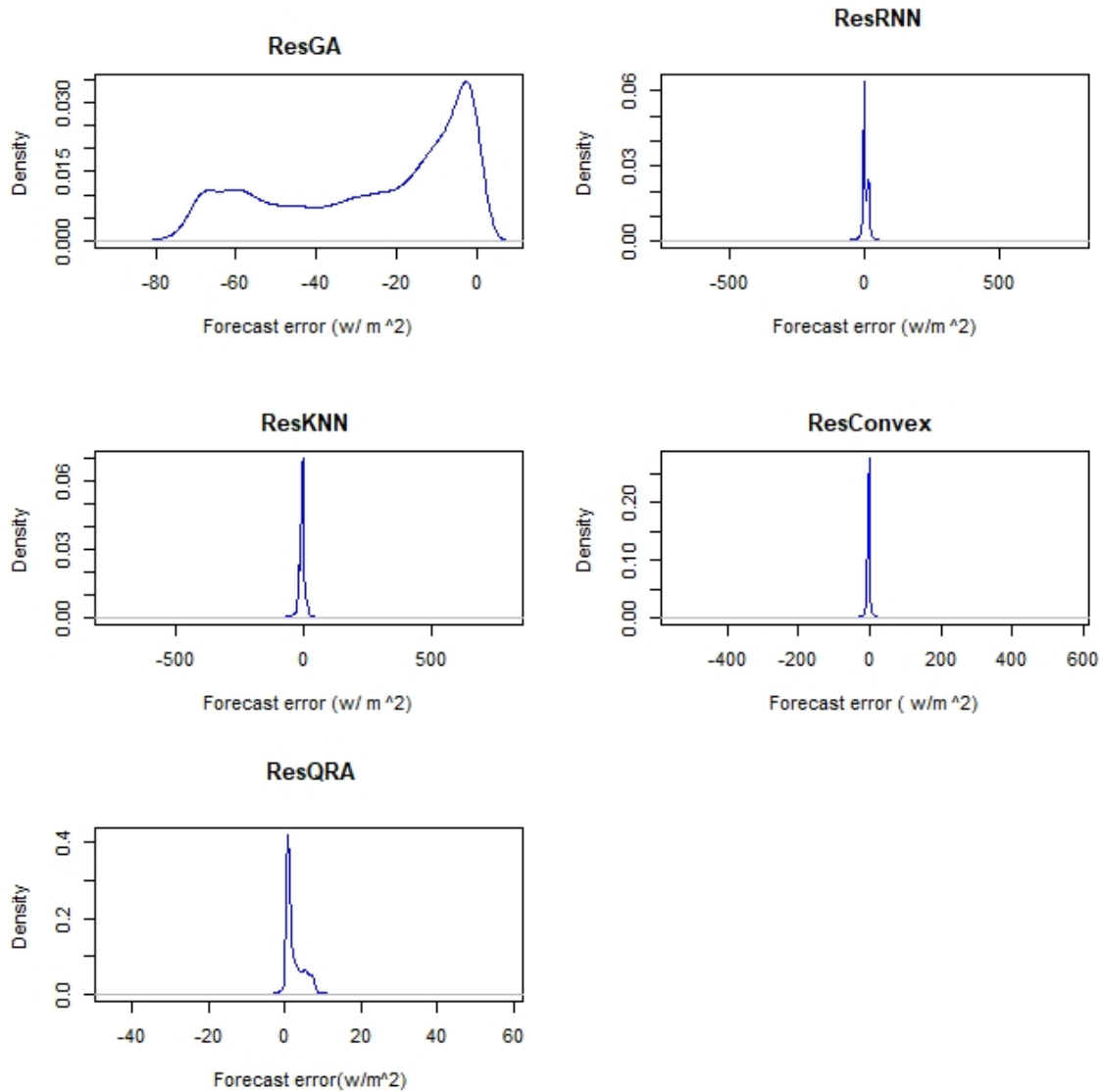


Figure 4.14: The error distribution of the forecast techniques for ResGA, ResRNN, ResKNN, ResConvex and ResQRA.

4.6.3 Percentage improvement

Table 4.10 indicates a percentage improvement of the output parameter by using the mean absolute error for the best model found in Table 4.6. The best model is QRA from Table 4.6, so the percentage improvements for other models are measured against the best QRA model and presented in Table 4.10. The highest percentage improvement is for QRA over GA that is 99.65% and the smallest is for the QRA over RNN that is 99.65%. This tells us that RNN is the second best model. The best model for forecasting solar irradiance is QRA followed by RNN.

Table 4.10: Percentage improvement.

Models	$MAE_{bestmodel}$	$MAE_{othermodel}$	Improvement (%)
QRA&GA	0.0924	26.74	99.65%
QRA&RNN	0.0924	20.18	99.54%
QRA&RNN	0.0924	20.94	99.56%

4.6.4 Diebold-Mariano test

The Diebold-Mariano test assumes the test's degree of significance is 0.05. This is a two-tailed test of 0.05 so that it must be divided into upper tail 0.025 and lower tail 0.025. The z-value equivalent to -0.025 is -1.96, which is a lower critical z-value. The upper value $1-0.025=0.975$, which is the z-value of 1.96. Diebold-Mariano rejects the null hypothesis if the measured DM statistic falls outside the -1.96 to 1.96 range. The null hypothesis states that the two forecasts have the same accuracy and the alternative hypothesis states that the two forecasts have different level of accuracy. The DM tests accept the null hypothesis because RNN and KNN have the same accuracy and the statistics fall inside the -1.96 and 1.96 range. So that the RNN is the best forecasting model.

Table 4.11: Diebold-Mariano test.

	Statistic	P-value
GA	-37.795	< 0.0001
RNN	-47.789	< 0.0001
KNN	-47.312	< 0.0001

4.6.5 Murphy diagram

Murphy diagrams (MDs) are based on the notion that if “if there is anything that can go wrong it might go wrong”. They are comparable to several other methods of analysis including such as fault trees because they analyse errors based on the possible causes of those errors. It is not essential to identify a particular scoring function before forecast evaluation when using MDs to compare forecast results (Ziegel et al., 2017). It is an advantage in the exis-

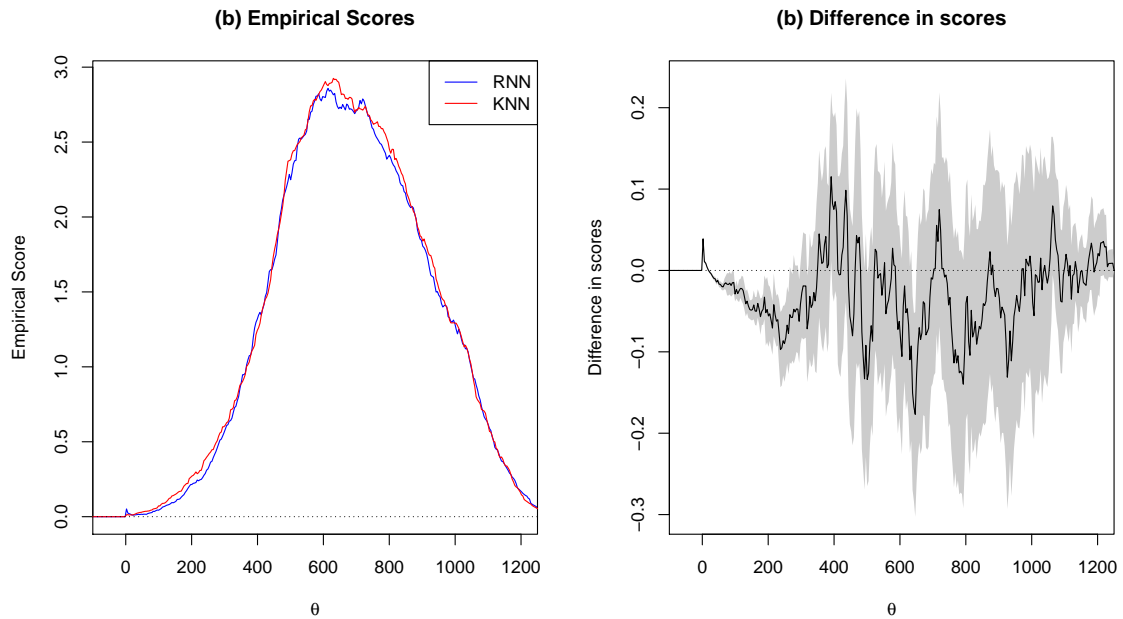


Figure 4.15: Plots of the RNN and KNN forecasts showing the relationship between the peaks of the model forecasts.

tence of potentially misspecified forecasts and non-nested datasets while the choice of a specific consistent scoring method produces a preference ordering on all potential forecast sequences that are typically difficult to explain. MDs on the other hand might lead to inconclusive cases wherein the other is governed by neither of the two forecasting models.

Figure 4.15, 4.16 and 4.17 are the Murphy diagrams where for the difference between the models, the shaded area displays 95% pointwise confidence intervals.

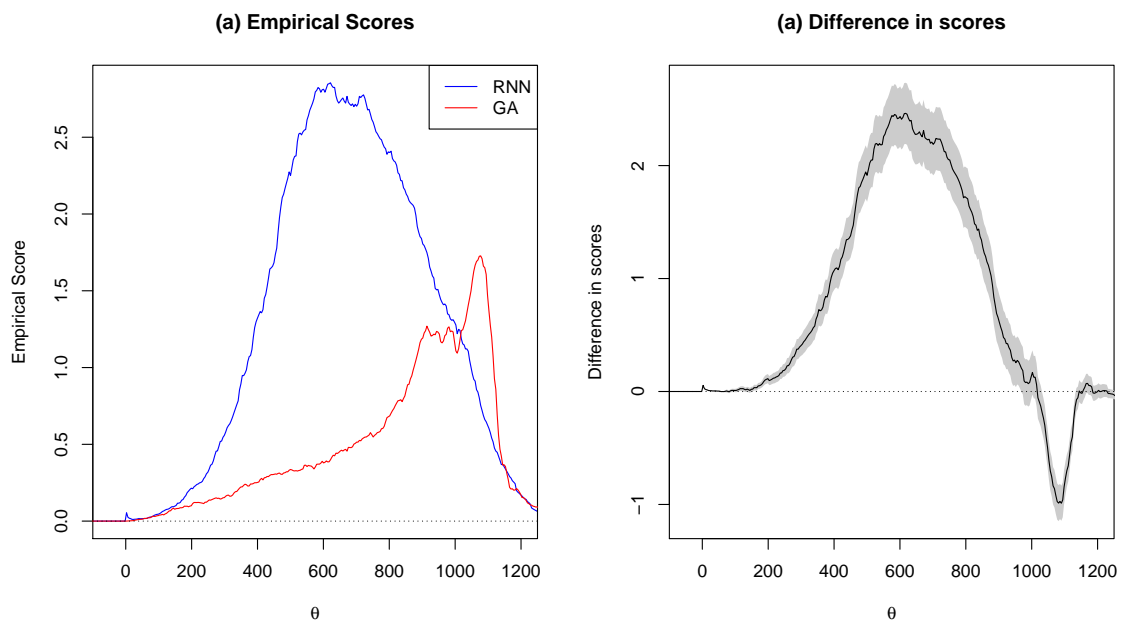


Figure 4.16: Plots of the RNN and GA forecasts showing the relationship between the peaks of the model forecasts.

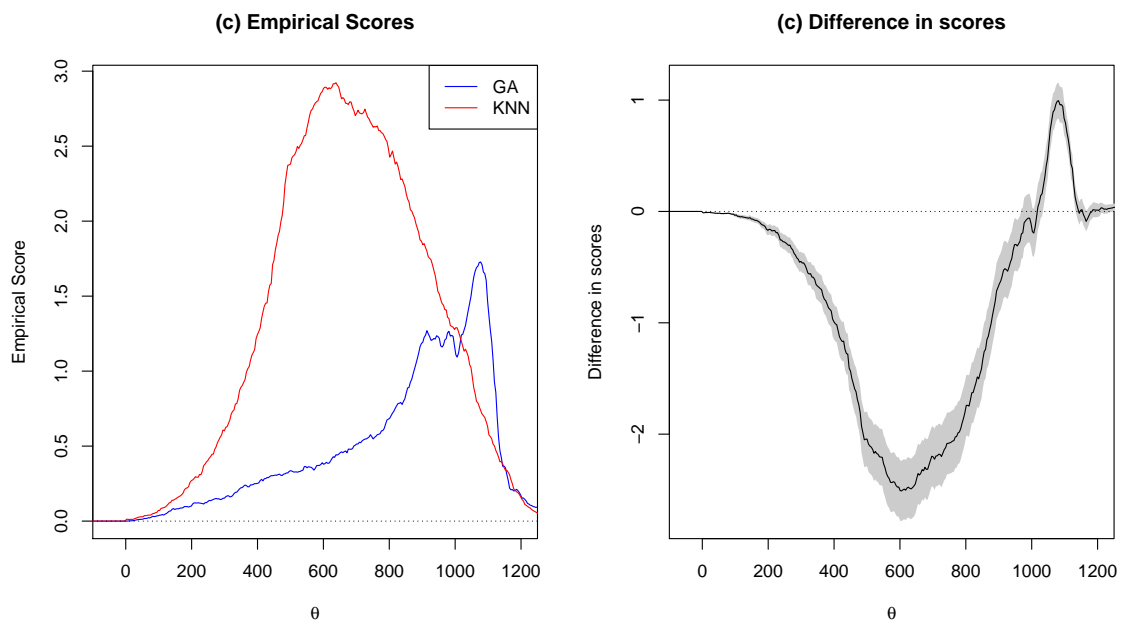


Figure 4.17: Plots of the GA and KNN forecasts showing the relationship between the peaks of the model forecasts.

4.7 Conclusion

This chapter presented the data analysis of solar irradiance using the algorithms discussed in Chapter 3. The forecasting performance of the GA, RNN and KNN algorithms was compared for forecasting minutes solar irradiance where KNN algorithm was used as a benchmark algorithm. The RNN model is found to be the best model for forecasting minutes of solar irradiance.

Chapter 5

Conclusion and future work

5.1 Introduction

This chapter presents the outcomes of the dissertation obtained in Chapter 4 and some ideas related to the dissertation, including fields for future research.

5.2 Findings of the dissertation

Renewable energy forecasts are critical to renewable energy grids and backup plans, operational plans and short-term power purchases. This dissertation focused on forecasting solar irradiance (global horizontal irradiance (GHI)) at one radiometric station in South Africa using high-frequency data (measured at one minute intervals) obtained from Vuwani radiometric station (USAid Venda). The data is from January 2020 to October 2020.

Forecasting minutes solar irradiance using recurrent neural network (RNN), genetic algorithm (GA) and k-nearest neighbour (KNN) where KNN was used as a benchmark model together with variable selection using least abso-

lute shrinkage and selection operator (LASSO) discussed in Chapter 4. The LASSO showed that total rainfall is not important in forecasting solar irradiance and it is excluded in the forecasting stage.

The results we got in Chapter 4 Section 4.4 showed that RNN is the best forecasting model in terms of the relative root mean square error (rRMSE) and relative mean square error (rMAE). In Section 4.5 forecasts for the machine learning algorithms combined with the help of the convex combination technique and QRA. Looking at the MAE and average pinball losses, QRA was found to be the best forecasting combination technique.

The boxplots in Figure 4.5, depicted that the models are positively skewed as the median was at the 25% lower quantile side. From the density plots in Figure 4.6, it is depicted that the RNN is the best forecasting technique based on the gap between the actual values and predicted values

Section 4.6 presented the predictive interval widths analysis with 95% level of confidence. It is found that the RNN has a lower PICP than all the models and a narrower PANAW than all the models. The results of the predictive interval analysis found the RNN to be the best fitting model. Subsection 4.6.3 presented the percentage improvement rates of the models. The QRA over RNN was found to be the one with the highest percentage improvement rate of 99.65%. This told us that RNN is the second-best model and QRA is the best model for forecasting solar irradiance. The Diebold-Mariano test presented in Section 4.6.4 and Murphy diagram in Section 4.6.5.

The Diebold-Mariano tests accept the null hypothesis that the RNN is the best forecasting model because the DM statistic falls between the -1.96 to 1.96 range. The Murphy diagrams displayed the 95% pointwise confidence interval.

5.3 Future work

Future researchers should look at forecasting renewable energy focusing on data from different radiometric stations in Africa, South Africa and across the world. In models, future researchers should consider looking at other machine learning techniques. Researchers in South Africa and Africa need to work on what we have and improve them so that it will help to grow the economy of our country. This study use long-short term minutes solar irradiance so that other researchers should look at using seconds, hourly, daily, weekly, monthly and yearly long and long-short term data from different radiometric stations.

5.4 Conclusion

This chapter presented a summary of the results and future research directions. This study together with its findings will have an impact on the South African power utility decision makes to align electricity demand and its supply in an efficient way that promotes potential economic growth and environmental sustainability.

References

- O. Abedinia, N. Amjady, and N. Ghadimi. Solar energy forecasting based on hybrid neural network and improved metaheuristic algorithm. *Computational Intelligence*, 34(1):241–260, 2018.
- A.A. Adeala, Z. Huan, and C.C. Enweremadu. Evaluation of global solar radiation using multiple weather parameters as predictors for South Africa provinces. *Thermal Science*, 19(suppl. 2):495–509, 2015.
- A.O. Adewumi and M.M. Ali. A multi-level genetic algorithm for a multi-stage space allocation problem. *Mathematical and Computer Modelling*, 51(1-2):109–126, 2010.
- A.Y. Al-Hassan, D.R. Hill, et al. Islamic technology; an illustrated history. 1986.
- A. Al-Karaghoul and L.L. Kazmerski. Energy consumption and water production cost of conventional and renewable-energy-powered desalination processes. *Renewable and Sustainable Energy Reviews*, 24:343–356, 2013.
- J.R. Andrade and R.J. Bessa. Improving renewable energy forecasting with a grid of numerical weather predictions. *IEEE Transactions on Sustainable Energy*, 8(4):1571–1580, 2017.

-
- F.M. Bianchi, L. Livi, and C. Alippi. Investigating echo-state networks dynamics by means of recurrence analysis. *IEEE transactions on neural networks and learning systems*, 29(2):427–439, 2016.
- F.M. Bianchi, E. Maiorino, M.C. Kampffmeyer, A. Rizzi, and R. Jenssen. *Recurrent neural networks for short-term load forecasting: an overview and comparative analysis*. Springer, 2017.
- J. Bien, J. Taylor, and R. Tibshirani. A lasso for hierarchical interactions. *Annals of Statistics*, 41(3):1111, 2013.
- M. Bjerke. Leak Detection in Water Distribution Networks using Gated Recurrent Neural Networks. Master’s thesis, NTNU, 2019.
- E. Cadenas and W. Rivera. Short term wind speed forecasting in La Venta, Oaxaca, México, using artificial neural networks. *Renewable Energy*, 34(1):274–278, 2009.
- G. Capizzi, F. Bonanno, and C. Napoli. Recurrent neural network-based control strategy for battery energy storage in generation systems with intermittent renewable energy sources. In *2011 International Conference on Clean Electrical Power (ICCEP)*, pages 336–340. IEEE, 2011.
- L. Cristaldi, G. Leone, and R. Ottoboni. A hybrid approach for solar radiation and photovoltaic power short-term forecast. In *Instrumentation and Measurement Technology Conference (I2MTC), 2017 IEEE International*, pages 1–6, 2017.
- J. Fan, X. Wang, L. Wu, H.I. Zhou, F. Zhang, X. Yu, X. Lu, and Y. Xiang. Comparison of Support Vector Machine and Extreme Gradient Boosting

- for predicting daily global solar radiation using temperature and precipitation in humid subtropical climates: A case study in China. *Energy Conversion and Management*, 164:102–111, 2018.
- J. Faouzi and H. Janati. pyts: A python package for time series classification. *Journal of Machine Learning Research*, 21(46):1–6, 2020.
- A. Garcia-Pedrero and P. Gomez-Gil. Time series forecasting using recurrent neural networks and wavelet reconstructed signals. In *2010 20th International Conference on Electronics Communications and Computers (CONI-ELECOMP)*, pages 169–173. IEEE, 2010.
- B. Gergo, G. Nagy, P. Gyozo, and S. Gabor. Forecasting framework for open access time series in energy. In *2016 IEEE International Energy Conference (ENERGYCON)*, pages 1–6. IEEE, 2016.
- S.A. Grady, M.Y. Hussaini, and M.M. Abdullah. Placement of wind turbines using genetic algorithms. *Renewable energy*, 30(2):259–270, 2005.
- S.S. Haykin. *Neural networks and learning machines*, volume 3. Pearson Upper Saddle River, NJ, USA:, 2009.
- S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
- P. Horton and K. Nakai. Better Prediction of Protein textscCellular Localisation Sites with the it k-Nearest Neighbors Classifier. In *Ismb*, volume 5, pages 147–152, 1997.
- G. Kariniotakis. *Renewable energy forecasting: from models to applications*. Woodhead Publishing, 2017.

- P.W. Khan and Y. Byun. Genetic Algorithm Based Optimised Feature Engineering and Hybrid Machine Learning for Effective Energy Consumption Prediction. *IEEE Access*, 2020.
- T. Khatib and W. Elmenreich. A model for hourly solar radiation data generation from daily solar radiation data using a generalised regression artificial neural network. *International Journal of Photoenergy*, 2015, 2015.
- D.J. King, C.S. Ozveren, D.A. Bradley, et al. Economic load dispatch optimisation of renewable energy in power system using genetic algorithm. In *2007 IEEE Lausanne Power Tech*, pages 2174–2179. IEEE, 2007.
- J. Kleissl. *Solar energy forecasting and resource assessment*. Academic Press, 2013.
- V. Kostylev, A. Pavlovski, et al. Solar power forecasting performance—towards industry standards. In *1st international workshop on the integration of solar power into power systems, Aarhus, Denmark*, 2011.
- B.Y.H. Liu and R.C. Jordan. The interrelationship and characteristic distribution of direct, diffuse and total solar radiation. *Solar energy*, 4(3):1–19, 1960.
- A. Lucas. *Wind, water, work: ancient and medieval milling technology*, volume 8. Brill, 2006.
- C.B. Lucasius and G. Kateman. Understanding and using genetic algorithms part 2. Representation, configuration and hybridisation. *Chemometrics and Intelligent Laboratory Systems*, 25(2):99–145, 1994.

-
- L. Marwala and B. Twala. Forecasting electricity consumption in South Africa: ARMA, neural networks and neuro-fuzzy systems. In *2014 International Joint Conference on Neural Networks (IJCNN)*, pages 3049–3055. IEEE, 2014.
- R. Mbuva, M. Jonsson, N. Ehn, and P. Herman. Bayesian neural networks for one-hour ahead wind power forecasting. In *2017 IEEE 6th International Conference on Renewable Energy Research and Applications (ICRERA)*, pages 591–596. IEEE, 2017.
- A. Mellit and S. Shaari. Recurrent neural network-based forecasting of the daily electricity generation of a Photovoltaic power system. *Ecological Vehicle and Renewable Energy (EVER), Monaco, March*, pages 26–29, 2009.
- K. Mohammadi, S. Shamsirband, A.S. Danesh, M.S. Abdullah, and M. Zamani. Temperature-based estimation of global solar radiation using soft computing methodologies. *Theoretical and applied climatology*, 125(1-2): 101–112, 2016.
- A.K. Morales and C.V. Quezada. A universal eclectic genetic algorithm for constrained optimization. In *Proceedings of the 6th European congress on intelligent techniques and soft computing*, volume 1, pages 518–522, 1998.
- P. Mpfumali, C. Sigauke, A. Bere, and S. Mulaudzi. Probabilistic solar power forecasting using partially linear additive quantile regression models: an application to South African data. *Energies*, 12(18):1–28, 2019.
- N.M. Odhiambo. Electricity consumption and economic growth in South

- Africa: A trivariate causality test. *Energy Economics*, 31(5):635–640, 2009.
- N.M. Odhiambo. Energy consumption, prices and economic growth in three SSA countries: A comparative study. *Energy Policy*, 38(5):2463–2469, 2010.
- Z. Peng, S. Yoo, D. Yu, and D. Huang. Solar irradiance forecast system based on geostationary satellite. In *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 708–713. IEEE, 2013.
- P.J. Pierre, B.D. Wolaver, B.J. Labay, T.J. LaDuc, C.M. Duran, W.A. Ryberg, T.J. Hibbitts, and J.R. Andrews. Comparison of recent oil and gas, wind energy, and other anthropogenic landscape alteration factors in texas through 2014. *Environmental Management*, 61(5):805–818, 2018.
- S.M. Prabhu and D.P. Garg. Artificial neural network based robot control: An overview. *Journal of Intelligent and Robotic Systems*, 15(4):333–365, 1996.
- T.J. Price. James Blyth—Britain’s first modern wind power pioneer. *Wind engineering*, 29(3):191–200, 2005.
- H. Quan, D. Srinivasan, and A. Khosravi. Uncertainty handling using neural network-based prediction intervals for electrical load forecasting. *Energy*, 73:916–925, 2014.
- J.S. Racine. Rstudio: a platform-independent ide for r and sweave, 2012.

-
- A. Raftery, P. Pinson, T. Gneiting, and P. McSharry. Wind energy: Forecasting challenges for its operational management. *Statistical Science*, 28(4):564–585, 2013.
- S. Rajeev and C.S. Krishnamoorthy. Genetic algorithm–based methodology for design optimisation of reinforced concrete frames. *Computer-Aided Civil and Infrastructure Engineering*, 13(1):63–74, 1998.
- T. Ravele. Medium term load forecasting in South Africa using Generalised Additive models with tensor product interactions, MSC dissertation, University of Venda, South Africa, 2018.
- A. Rezrazi, S. Hanini, and M. Laidi. An optimisation methodology of artificial neural network models for predicting solar radiation: a case study. *Theoretical and applied climatology*, 123(3-4):769–783, 2016.
- J. Sadeghi, S. Sadeghi, and S.T.A. Niaki. Optimising a hybrid vendor-managed inventory and transportation problem with fuzzy demand: an improved particle swarm optimization algorithm. *Information Sciences*, 272:126–144, 2014.
- C. Sigauke. Forecasting medium-term electricity demand in a South African electric power supply system. *Journal of Energy in Southern Africa*, 28(4):54–67, 2017.
- S.N. Sivanandam and S.N. Deepa. Genetic algorithms. In *Introduction to genetic algorithms*, pages 15–37. Springer, 2008.
- S. Sun, S. Wang, G. Zhang, and J. Zheng. A decomposition-clustering-ensemble learning approach for solar radiation forecasting. *Solar Energy*, 163:189–199, 2018.

- X. Sun, Z. Wang, and J. Hu. Prediction interval construction for byproduct gas flow forecasting using optimised twin extreme learning machine. *Mathematical Problems in Engineering*, 2017, 2017.
- L.K. Tartibu and K.T. Kabengele. Forecasting net energy consumption of South Africa using artificial neural network. In *2018 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, pages 1–7. IEEE, 2018.
- M. Taylor, P. Ralon, and A. Ilas. The power to change: solar and wind cost reduction potential to 2025. *International Renewable Energy Agency (IRENA)*, 2016.
- A. Timmermann. Forecast combinations. *Handbook of economic forecasting*, 1:135–196, 2006.
- S. Tokui, K. Oono, S. Hido, and J. Clayton. Chainer: a next-generation open source framework for deep learning. In *Proceedings of workshop on machine learning systems (LearningSys) in the twenty-ninth annual conference on neural information processing systems (NIPS)*, volume 5, pages 1–6, 2015.
- S.B. Tsai, Y. Xue, J. Zhang, Q. Chen, Y. Liu, J. Zhou, and W. Dong. Models for forecasting growth trends in renewable energy. *Renewable and Sustainable Energy Reviews*, 77:1169–1178, 2017.
- U. Ugurlu, I. Oksuz, and O. Tas. Electricity price forecasting using recurrent neural networks. *Energies*, 11(5):1255, 2018.
- E. Werner, T. Gneiting, A. Jordan, and F. Krüger. Of quantiles and expectiles: Consistent scoring functions, Choquet representations, and forecast rankings. *arXiv preprint arXiv:1503.08195*, 2015.

- X. Yang, F. Jiang, and H. Liu. Short-term solar radiation prediction based on SVM with similar data, in the proceeding of 2013 IEEE RPG (2nd IET Renewable Power Generation Conference), 2013.
- A. Zendejboudi, M.A. Baseer, and R. Saidur. Application of support vector machine models for forecasting solar and wind energy resources: A review. *Journal of Cleaner Production*, 199:272–285, 2018.
- E. Zhandire. Solar resource classification in South Africa using a new index. *Journal of Energy in Southern Africa*, 28(2):61–70, 2017.
- J.F. Ziegel, F. Krüger, A. Jordan, and F. Fasciati. Murphy Diagrams: Forecast Evaluation of Expected Shortfall. *arXiv preprint arXiv:1705.04537*, 2017.

Appendix: Some selected code

```
#-----  
#packages-----  
#packages used for the following selected code:  
# forecast , ggplot2, opera, data.table, qgam , mgcv , tseries , e1071  
#mgcv, class, gmodel, keras, plyr, dplyr, glmnet.  
#-----  
library(data.table)  
library(tseries)  
library(quantreg)  
library(readxl)  
library(readr)  
library(opera)  
library(forecast)  
library(glmnet)  
library(hierNet)  
library(dplyr)  
library(plyr)  
library(mgcv)  
#Load data-----  
attach(analyticdata)  
dim(analyticdata)  
df <- analyticdata[c('GHI', 'Temp', 'RH', 'Rain', 'WS', 'WD', 'WD_StdDev', 'BP')]  
dim(df)  
head(df)  
tail(df)
```

```
#-----  
# Time Series Plot, Density Plot, QQ Plot  
# and Box Plot for solar irradiance  
win.graph()  
par(mfrow=c(2,2)) # mfrom = (2,2)  
# GHI-Global horizontal irradiance  
# ts=time series  
y=ts(df$GHI, frequency = 1440) # 60*24=1440  
plot(y,main="Demand (2020)",xlab = "Year",ylab="GHI",col="blue")  
DPED=density(df$GHI)  
plot(DPED,main="Density plot (2020)",col="brown")  
  
qqnorm(y , col = " blue " , main = " (c) Normal QQ plot " ) ## qq plot  
qqline(y) # qq plot line  
  
boxplot(y , main = " (d) Box plot " , varwidth = TRUE ,  
xlab = " GHI (W/m ^2) " , col = " blue " , horizontal = TRUE ) ## Box plot  
#-----  
#-----  
summary(y) # Summary statistics for GHI  
sd(y)  
skewness(y) # Skewness  
kurtosis(y) # Kurtosis  
#conduct Jarque-Bera test  
jarque.bera.test(y)  
#-----
```

```
# Extraction and fitting of the non-linear trend values
K<-ts(df$GHI) # ts=time series
win.graph()
plot(K, xlab="Time", ylim =c(0,1500),type ="l",
ylab ="GHI (W/m ^2) ", col = "green")
length(df$GHI)
R= smooth.spline(time(K), K)
R
lines(smooth.spline(time(K),K,spar = -0.2091344) , lwd =3 , col = " brown ")
dpdfits=fitted(( smooth.spline(time(K),K,spar = -0.2091344) ))
write.table(dpdfits,"~/ GHIfittedspline . txt " , sep ="\t ")
#-----
dt = sort(sample(nrow(df), nrow(df)*.8))
train<-df[dt,]
test<-df[-dt,]

length(test)
length(dt)
length(train)
#-----
#Variable selection Via LASSO for GHI
y<-train$GHI
x<-data.matrix(train[2:8])
print(x)
Lasso<-glmnet(y=y,x=x,family="gaussian")
plot(Lasso)
```

```
coef(LAzzo)
cv.lasso<-cv.glmnet(y=y,x=x,family="gaussian")
cv.lasso
plot(cv.lasso)
coef(cv.lasso)
predict.1<-predict(cv.lasso,newx=x)
predict.1
m.lasso<-mean((y-predict.1)^2)
m.lasso
#-----
win.graph()
par(mfrow=c(2,2))

y <- ts(GHI)
plot(y, ylim =c(0.0,1900) , main="GHI and GA forecasts",col="black",
ylab="solar irradiance",xlab="Time")
lines(fGA, col="green")
legend("topleft",col=c("black","green"),lty=1:2,lwd=2,
legend=c("GHI","GA"))

plot(y, ylim =c(0.0,1900) , main="GHI and RNN forecasts",col="black",
ylab="solar irradiance (w/m ^2)",xlab="Time")
lines(fRNN, col="green")
legend("topleft",col=c("black","green"),lty=1:2,lwd=2,
legend=c("GHI","RNN"))
```



```
plot(y, ylim =c(0.0,1900) , main="GHI and RNN forecasts",col="black",
ylab="solar irradiance (w/m ^2)",xlab="Time")
lines(fKNN, col="green")
legend("topleft",col=c("black","green"),lty=1:2,lwd=2,
legend=c("GHI","KNN"))

#-----

### GA , RNN , KNN density plots
## GHI , GA, RNN
win.graph()
par(mfrow=c(2,2))
x = density(GHI)
plot (x , ylim =c (0.0 ,0.0025) , xlim =c ( -100 ,1500) , col =" blue " ,
main =" Density of
irradiance (GA) " ,
xlab ="number of observations")

#library (forecast) #-----
head(forecasts)
win.graph()
accuracy(fGA,GHI)

#-----

Y <- GHI
X <- cbind (fGA,fRNN,fKNN)
matplot(cbind(Y,X),type="l" , col =1:4)
```

```
GA<- fGA
RNN<- fRNN
KNN<- fKNN
X <- cbind ( fGA, fRNN, fKNN)
# How strong is a specialist? Just look at the oracles below
#-----
win.graph()
G=oracle.convex
  G<- oracle(Y = Y, experts = X, loss.type = 'pinball',
  Model = "convex")
plot(G)
G

mix1 = 0.241* GA + 4.35e-21 * fKNN + 0.759 * fRNN # pinball
mix2 = 0.699 * fGA + 0.00 * fKNN +0.301 * fRNN # square
mix3 = 0.241 * fGA + 3.7e-21 * fKNN + 0.759 * fRNN # absolute
# measures of the accuracy
accuracy ( mix1 , GHI )

#-----

# #####
## QRA
# #####
attach(forecasts) # Attach the forecasts
win.graph()
```

```
#y<-ts(GHI)
plot(y,xlab="Observation number" , ylab="Minutes irradiance")

qr22.GHI=rq(GHI~fGA + fRNN + fKNN, data = forecasts, tau =0.5)

summary.rq(qr22.GHI,se="boot")
lines(qr22.GHI$fit,col="red")
fQRA=fitted(qr22.GHI)
accuracy(fQRA,GHI )
accuracy(mix1,GHI )
fconvex=mix1

#-----
# PIN BALL LOSSES
#-----

install.packages("devtools")
library(devtools)
install_github("camroach87/gefcom2017")

library(gefcom2017)
## tau: integer 1, 2, ... 99. Quantile to calculate pinball loss score for.
## y: numeric. Observed value.
## q: numeric. Predicted value for quantile tau.
#' Calculates the pinball loss score for a given quantile.
pinballloss <- function(tau, y, q) {
  pldf <- data.frame(tau = tau,
y=y,
```

```
q=q)
pldf <- pldf %>%
mutate(L = ifelse(y>=q,
tau/100 * (y-q),
(1-tau/100) * (q-y)))
return(pldf)
}
tau= 50
y= GHI
q=fGA # fKNN, fRNN, fGA

U = pinballloss(tau, y, q)
U
#write.table(U,"~/pinballfGA.txt",sep="\t")

qloss =U$L
a=ts(qloss)
plot(a)
mean(qloss)
#-----
### PIW for the entire models
#-----
## GA , RNN, KNN, Fconvex , Fqra
#-----
# ### Model PIWs comparisons at 95%
attach( PIs_UPRnew)
```

```
head (PIs_UPRnew)

PIW =c("PIGA", "PIRNN", "PIKNN", "PIConvex", "PIQRA")
win.graph()

boxplot(PIGA95,PIRNN95,PIKNN95,PIconvex95,PIQRA95,names=PIW,
horizontal=FALSE,main="95% prediction intervals" ,
ylab ="Prediction interval width(w/m^2)", col="blue")
win.graph()
par(mfrow=c(3,2))
plot (density(PIGA95), xlab =" Prediction interval width(w/m^2)" , col
=" blue ", main ="PIGA")
#-----
# Residual Error Analysis
#-----
attach(forecasts)
head(forecasts)
ResGA = forecasts$GHI-forecasts$fGA
ResRNN = forecasts$GHI-forecasts$fRNN
ResKNN = forecasts$GHI-forecasts$fKNN
ResConvex = forecasts$GHI-fconvex
ResQRA = forecasts$GHI-fQRA
#-----
library(e1071)
#-----
RESID = c(" ResGA " ," ResRNN " ," ResKNN " ," ResConvex " , " ResQRA ")
```

```
win.graph()
boxplot ( ResGA , ResRNN , ResKNN , ResConvex , ResQRA , names = RESID
, horizontal = FALSE , main ="" ,
ylab =" Residuals (w/m ^2) " , col = " blue ")

win.graph()
par ( mfrow = c (3 ,2) )
plot ( density(ResGA), xlab =" Forecast error (w/ m ^2) " , col =" blue "
, main =" ResGA ")
#-----
# Package 'murphydiagram'
#-----
# Murphy diagrams to visualize forecast comparisons
#-----
# Visual comparisons of two forecasting methods, allowing to
# study whether the ranking is robust across the class of
# elementary or extremal scoring functions.

# murphydiagram plots the extremal scores of two
# forecasting methods.

#-----
# Murphy Diagrams Rob Hyndman R codes
#-----

source("https://robjhyndman.com/Rfiles/murphy.R")
```

```
library(forecast)
library(murphydiagram)

win.graph(width=12, height=7, pointsize=12)
par(mfrow = c(1,2))

murphydiagram(GHI, fRNN, fGA, main="(a) Empirical Scores",
xlim=c(-50,1200))
legend("topright", lty=1, col=c("blue","red"),
legend=c("RNN","GA"))

murphydiagram(GHI, fRNN, fGA, type='diff', xlim=c(-50,1200),
main="(a) Difference in scores")

#-----
```