December 2015

Performance Analysis of Hybrid Algorithms For Lossless Compression of Climate Data

Bharath Chandra Mummadisetty
University of Nevada, Las Vegas, mummadis@unlv.nevada.edu

PERFORMANCE ANALYSIS OF HYBRID ALGORITHMS FOR LOSSLESS COMPRESSION OF CLIMATE DATA

by

Bharath Chandra Mummadisetty

Bachelor of Engineering

MVJ College of Engineering, Bangalore

2010

A thesis submitted in partial fulfillment of the requirements for the

Masters of Science in Electrical Engineering

Department of Electrical and Computer Engineering

Howard R.Hughes College of Engineering

The Graduate College

University of Nevada, Las Vegas

December 2015

UNLV | GRADUATE COLLEGE

Thesis Approval

The Graduate College
The University of Nevada, Las Vegas

September 9, 2015

This thesis prepared by

Bharath Chandra Mummadisetty

entitled

Performance Analysis of Hybrid Algorithms for Lossless Compression of Climate Data

is approved in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering
Department of Electrical and Computer Engineering

Shahram Latifi, Ph.D.
Examination Committee Chair

Kathryn Hausbeck Korgan, Ph.D.
Graduate College Interim Dean

Sahjendra Singh, Ph.D.
Examination Committee Member

Ebrahim Saberinia, Ph.D.
Examination Committee Member

Wolfgang Bein, Ph.D.
Graduate College Faculty Representative

ABSTRACT

PERFORMANCE ANALYSIS OF HYBRID ALGORITHMS FOR LOSSLESS COMPRESSION OF CLIMATE DATA

By

Bharath Chandra Mummadisetty

Dr.Shahram Latifi, Examination Committee Chair
Professor, Electrical and Computer Engineering Department
University of Nevada, Las Vegas

Climate data is very important and at the same time, voluminous. Every minute a new entry is recorded for different climate parameters in climate databases around the world. Given the explosive growth of data that needs to be transmitted and stored, there is a necessity to focus on developing better transmission and storage technologies. Data compression is known to be a viable and effective solution to reduce bandwidth and storage requirements of bulk data. So, the goal is to develop the best compression methods for climate data.

The methodology used is based on predictive analysis. The focus is to implement a hybrid algorithm which utilizes the functionality of Artificial Neural Networks (ANN) for prediction of climate data. ANN is a very efficient tool to generate models for

predicting climate data with great accuracy. Two types of ANN's such as Multilayer Perceptron (MLP) and Cascade Feedforward Neural Network (CFNN) are used. It is beneficial to take advantage of ANN and combine its output with lossless compression algorithms such as differential encoding and Huffman coding to generate high compression ratios.

The performance of the two techniques based on MLP and CFNN types are compared using metrics including compression ratio, Mean Square Error (MSE) and Root Mean Square Error (RMSE). The two methods are also compared with a conventional method of differential encoding followed by Huffman Coding.

The results indicate that MLP outperforms CFNN. Also compression ratios of both the proposed methods are higher than those obtained by the standard method. Compression ratios as high as 10.3, 9.8, and 9.54 are obtained for precipitation, photosynthetically active radiation, and solar radiation datasets respectively.

ACKNOWLEDGMENTS

I would like to express my sincere thanks and gratitude to, Dr. Shahram Latifi, my research advisor for all the support, and guidance he has offered me during the course of my graduate studies at University of Nevada, Las Vegas and for giving me an opportunity as a Research Assistant and trusting me. His encouragement and valuable suggestions have helped me immensely in seeking the right direction for this thesis.

I would also like to thank Dr. Sahjendra Singh, Dr.Ebrahim Saberinia, and Dr. Wolgang Bein for readily accepting my invitation to serve in my committee.

I would like to acknowledge NSF for providing me with the opportunity to work in the project titled "Solar Energy- Water - Environment Nexus". This work was supported by the National Science Foundation (NSF) grant #EPS-IIA-1301726.

My deepest gratitude to my parents Badrinath Mummadisetty and Nirmala Kumari Mummadisetty, who have been my source of inspiration, for their unconditional love, care and support they have given me at every stage of my life. I would also like to thank my brother Subhash Chandra Mummadisetty for his support and guidance in building up my career.

I would also like to thank my friends namely Nithin, Varsha, Naveen, and Vignesh for all the support and being there for me through thick and thin. Finally I would like to thank my fellow students and friends at UNLV who helped me throughout my masters.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background

We have been witnessing a revolution in the past few years in the way that communication works, and this revolution is still evolving. This includes the fast growing internet and the growth of mobile and video communication. It would be impractical to provide better clarity for communication through cell phones; it would be difficult to have images, audios, and videos on websites without data compression. With the help of data compression one can listen to music on a music player, watch movies on DVD, make long distance calls etc. Most data compression algorithms use MPEG and JPEG techniques to represent a picture, video or audio with smaller number of bits. Data compression is the field where information is encoded using fewer bits than the original representation. The more one knows about the data, the better it can be compressed. The data to be compressed can be text in a file, number, decimal data, audio signals, and images generated due to different processes. Most information now generated and used is in digital form. This digital data is represented by bytes of data. The number of bytes required to represent multimedia data can be huge. For example, it takes 40 megabytes to represent a 2 second video without the use of data compression. So one can calculate the amount of memory that is required for the storage of an entire movie. It requires more than 21 million bits to represent 1 minute of uncompressed CD-quality music (44,100

samples per second, 8 bits per sample).[1] Downloading data at this rate can be time consuming. Walmart handles more than 1 million customer transactions every hour. It is estimated that it contains more than 2.5 Petabytes of data. Google used to process more than 20,000 Terabytes of data per day. Around 100 hours of video content is posted on YouTube every minute. There are 30 billion pieces of content shared on Facebook every month. Given the explosive growth of data that needs to be transmitted and stored, we should focus on developing methods for better transmission and storage.

Significant advances that allow large volumes of information to be stored and transmitted without using compression in media such as CD-ROMs, optical fibers, asymmetric digital subscriber lines (ADSL), and cable modems have been made. However, as per Parkinson's first law, the need for mass storage and transmission increases at least twice as fast as storage and transmission capacities improve. But there are also situations in which capacity does not increase a significant amount. For example, the amount of information that can be transmitted over the air is limited and this depends on the characteristics of the atmosphere. Samuel Morse invented the Morse code which was one of the early examples of data compression in the mid-19th century. The information used by telegraph was encoded with dots and dashes. According to Morse the frequency of occurrence of some letters was more than others. He assigned shorter sequences to letters that occur more often and longer sequences to Letters that occur less frequently. Braille code uses frequency of occurrence of certain words for compression whereas Morse code uses frequency of occurrence of single characters.

In Braille coding, text is represented by 2×3 arrays of dots. Each array of six dots represents a single character in Grade 1 Braille. We obtain 26, or 64, different combinations, given six dots with two positions for each dot. 38 combinations will be left when we use 26 of these for the different letters. A few of the leftover groups are used to encode words that occur more often, such as “and” and “for.” One of the combinations is utilized as a different symbol depicting on whether the symbol that follows is a word and not a character, thus sanctioning an immense number of words to be represented by two arrays of specks. These alterations, alongside compressions of a percentage of the words, bring about a normal decrease in space of around 20%. Factual structure is being utilized to give compression in these samples; however that is by all account not the only sort of structure that exists in the information. When we talk, the physical development of our voice box directs the sorts of sounds that we can create. The working process of speech creation has a structure on the speech. In this way, as opposed to transmitting the audio signal itself, we could send data about the compliance of the voice box, which could be utilized by the receiver to synthesize the audio signal. This methodology is being utilized at present as a part of various applications, including transmission of speech over mobile radios and the manufactured voices in toys that talk.

An early form of the above mentioned methodology, called the vocoder (voice coder), was produced by Homer Dudley at Bell Laboratories in 1936. The vocoder was exhibited at the New York World's Fair in 1939, where it was a noteworthy fascination. These are only a few of the many different types of structures that can be used to obtain

compression. The structure in the information is not by any means the only thing that can be utilized to acquire compression. We can likewise make utilization of the characteristics of the user for the information. Ordinarily, for instance, when transmitting audio signal and pictures, the information is proposed to be seen by a human, and people have constrained perceptual capacities. For example, we cannot hear the very high frequency sounds that dogs can hear.When the information cannot be perceived by the user it is usually not required to keep that information [1].

Therefore, we can make utilization of the perceptual constraints of people to get compression by discarding unnecessary information. This methodology is used as a part of various compression techniques that we will visit in the following segments.

1.2 Motivation

The climate database is enormous. Every minute a new entry is recorded for different climate parameters around the world in different climate data bases. Climate data is very important for research studies and is ever growing. Given the explosive growth of data that needs to be transmitted and stored, we should focus on developing better transmission and storage technologies.

The aim is to implement a hybrid algorithm that utilizes the functionality of ANN for prediction of climate data. ANN is a very efficient technique for predicting data which can be helpful in generating models for predicting climate data with great accuracy.

It is beneficial to take advantage of this predictive analysis and combine its output with lossless compression algorithms to generate best results.

1.3 Research Goal

The goal of this thesis is to do comparative analysis of the performance of machine learning algorithms simulated on climatology datasets, and to combine the output with data compression algorithms i.e differential encoding followed by Huffman coding for better compression ratio. As climate data is very valuable for the scientific community, the precision and accuracy of the data is of utmost importance. Therefore, the thesis will be focused on lossless compression of climate data. The methods used for analysis are MLP and CFNN. The performance of the algorithms is measured by considering quality metrics i.e Compression Ratio, Saving Percentage MSE, and RMSE. All the computations and algorithms writing are conducted in MATLAB.

1.4 Related Work

A technique is presented in [2] to compress pressure, wind and precipitation data where uniform quantization is applied to the pressure data as a part of preprocessing the data. Optimal prediction techniques are used to predict the values based on the surrounding values and the differences are encoded. Wind data (wind velocity and wind direction) are transformed into polar co-ordinate form. Here entropy coding is performed after the preprocessing stage. Also, significant improvements in bandwidth can be

realized through the use of common compression techniques such as wavelet/error grid or round difference/BZIP2 compression [3]. The compression of temperature data in Kuala lampur from January 1948 until July 2010 by using Debauchies wavelet (D4) as the basis function is performed [4].Many approaches for scientific data compression have been focused primarily on combining compression with data synthesis in order to increase throughput and conserve storage. Engel son [5] compressed sequences of double precision floating point values resulting from simulations based on ordinary differential equations. In theoretical approach the numbers are treated as integers and then compressed using predictive coding, with residuals being explicitly stored in case of lossless coding or truncated for lossy coding. Ratanaworabhan proposed a lossless prediction based compression method for double- precision floating point scientific data using a DFCM (differential finite context method) value method which is based on pattern matching using a hash table holding recent encoding context. The bitwise residual was then computed using XOR operator,with compressed representation consisting of finite number of leading zeros and remaining residual bits [6].Weather forecasting is a vital application in meteorology and has been one of the most scientifically and technologically challenging problems around the world in the last decade. The use of data mining techniques in forecasting maximum temperature, rainfall, evaporation and wind speed works well [7]. This was carried out using ANN and decision tree algorithms and meteorological data collected between 2000 and 2009 from the city of Ibadan, Nigeria. A data model for the meteorological data was developed and this was used to train the

classifier algorithms. The performances of these algorithms were compared using standard performance metrics, and the algorithm which gave the best results used to generate classification rules for the mean weather variables. A predictive neural network model was also developed for the weather prediction program and the results compared with actual weather data for the predicted periods [8].The investigation was done to develop ANN for ambient air temperature prediction in Kerman city located in the south east of Iran. The mean, minimum and maximum ambient air temperature during the year 1961-2004 was used as the input parameter in feed forward network and Elman network. The values of R2, MSE and MAE variables in both networks showed that ANN approach is a desirable model in ambient air temperature prediction, while the results from Elman network are more precise than FNN network [9].

The ANN models use different geographical parameters of a location as inputs for the prediction of solar radiation as discussed in [10]. Al-Alawi and Al-Hinai [11] discussed multi-layer feed forward network, back propagation (BP) training algorithm for global radiation prediction in Seeb, Oman. The inputs used in network were location, month, mean pressure, mean temperature, mean vapor pressure, mean relative humidity, mean wind speed and mean sunshine hours. Sözen et al. [12] [13] used meteorological and geographical data as input variables in the ANN model for solar radiation estimation in Turkey. The transfer function for model is logistic sigmoid and learning algorithm is Scaled conjugate gradient, Pola-Ribiere conjugate gradient Levenberg-Marquardt.In the study undertaken by AbdAlKader and AL-Allaf, 2008 BPNN models were developed to

predict the day soil temperature for the present day by using various previous day meteorological variables in Nineveh-Iraq.The BPNN models (M4: BP, M5: Cascade BP, and M6: NARX) consisting of the combination of the input variables were constructed to obtain the best fit input structure. After ANN training, testing the model M4: BP gave a soil temperature prediction correctly by 75%. Whereas the model M5: Cascade BP gave predict correctly by 80%.While the model M6: NARX gave soil temperature prediction correctly by 95% [14].In the research work done by Tamer Khatib, Azah Mohamed, K. Sopian, M. Mahmoud , prediction of hourly solar radiation values for Kuala Lumpur was performed. This prediction was performed using the GRNN, FFNN, CFNN, and ELMNN ANNs. Prediction results show that GRNN has a higher efficacy compared to the other proposed networks. The FFNN and CFNN are still efficient at predicting solar radiation but do not predict well in poor radiation conditions such as the first and final hour of the solar day. The ELMNN was the worst at predicting the solar radiation among the proposed methods. Based on our results, GRNN is recommended for such purposes in Malaysia and other nearby regions [15].

In the research work done by Bharath Chandra Mummadsietty and Astha Puri, the problem of lossless, offline compression of climate data was addressed. They proposed a method for compression solar radiation, photo synthetically active radiation, and data logger power system voltage data using combination of differential encoding and Huffman coding [16]. Bharath Chandra Mummadsietty and Astha Puri performed lossless compression of solar radiation data using ANNs as part of preprocessing [17].

CHAPTER 2

DATA SOURCES AND SENSORS

Following are the sources from which climate data is obtained :

2.1 Data Sources

This section describes the Nevada Climate Change Portal (NCCP), different locations with sensors, and the types of climate parameters measured for each location. It also describes the different type of sensors available to measure the climate parameters and the employed hardware.

2.1.1 NCCP

- **Nevada Climate Change Portal**

  One of the key outcomes of Nevada infrastructure for climate change science RII track1, education, and outreach project is NCCP. The estimations for the air, water, plant, and soil readings originate from thirteen Nevada Climate-ecohydrological Assessment Network (NevCAN) sites. More than 1.2 billion records have been recorded in the NCCP climatology database and this number is increasing consistently. This portal helps in providing the necessary data required for long and short-term research examination [18].

The NCCP can be accessed at http://sensor.nevada.edu website. It is open for everyone and can be freely accessed. This portal provides data for interested researchers, organizations etc. The data provided is of high quality and can be used by current and future researchers for long-term studies. The data collection and data management of the climate data are very efficiently performed. The documentation is very clear and lucid in the portal. Different types of tools, information, and other resources are also available for students, researchers and educators. This portal and its underlying infrastructure were built by the cyber infrastructure component of the NSF EPSCoR project. They have helped to purchase and set up computing resources. The cyber infrastructure component of the NSF EPSCoR project also helped to manage, support the NCCP, and set up the sensor network at different locations.

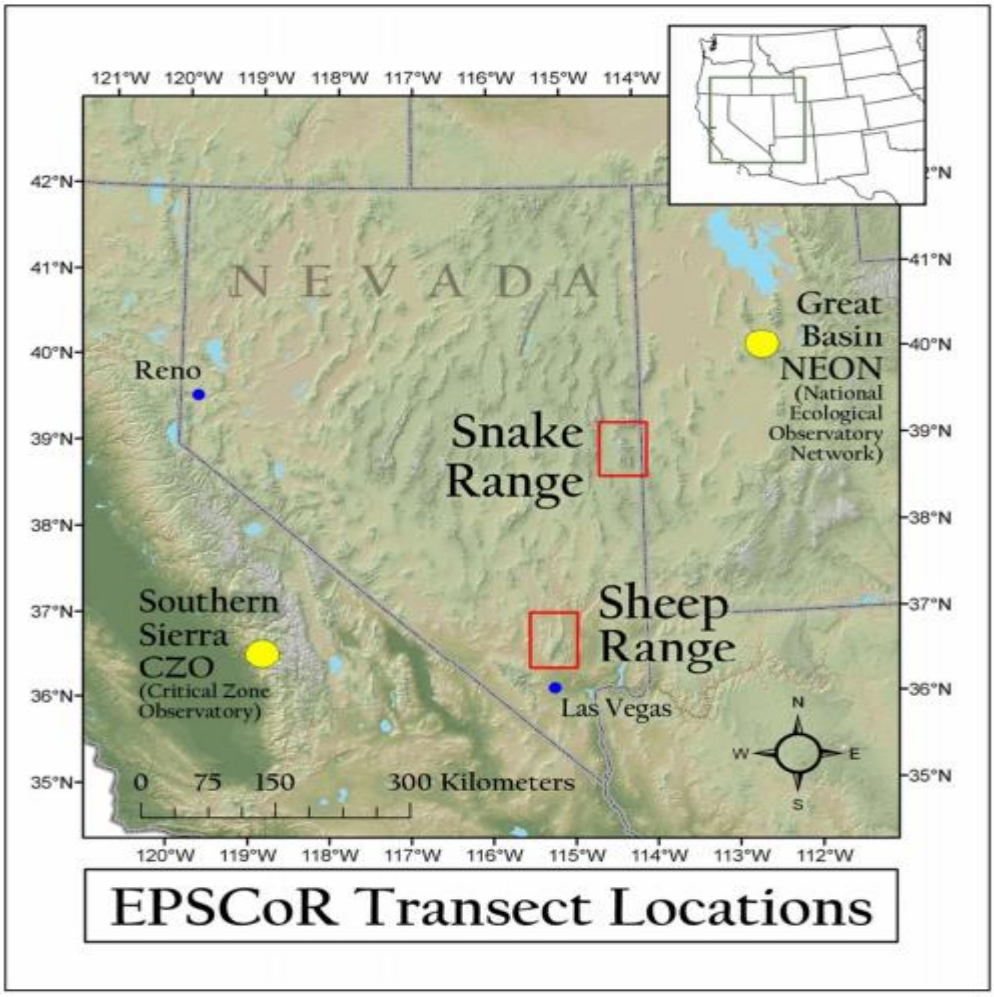

Fig 2.1 Location of EPSCoR – NevCAN stations [18].

- **NevCAN, Research Sites and Equipment**

  NevCAN is Nevada Climate-ecohydrological Assessment Network. A total of 13 sites were constructed for field observation in Eastern Nevada. These stations were manufactured in the Snake and Sheep mountain extents. Fig 2.1 shows the

location of NevCAN. The maximum elevation is 800m and the minimum elevation is 300m above the sea level for these sites. Included on the NevCAN stations in the Snake Range are atmospheric/meteorological sensing systems comprised of 9 different physical sensors which are used to measure free air temperature at 10m and 2m heights, relative humidity, air pressure, incoming and outgoing long-wave and short-wave solar radiation, wind speed/direction, and snow depth [18].

Fig 2.2 Nevada Climate Change Portal [19].

Fig 2.2 shows the screenshot of Nevada Climate Change Portal which contains the data captured in all the regions for different parameters. There are two different sensors for measuring precipitation i.e., one for rain and other for

measuring snow. Also water content and soil conditions are monitored. For water content and temperature, a vertical array of depth up to 50 cm is considered. There are nine sensors with which soil conditions are monitored. Information logging of these variables go as high as up to one minute per observation concerning the qualities of procedures at short timescales over the scene. Apart from these sites, there are also some sites which measure tree sap flow, snow water equivalent, distributed soil moisture and temperature, incremental tree growth, and Normalized Differential Vegetation Index. To visualize different processes and conditions in that area, a controllable Point-Tilz-Zoom (PTZ) camera was installed. It can take up to twenty pictures per hour and over thirteen hundred pictures are captured over the entire different sites in the network.

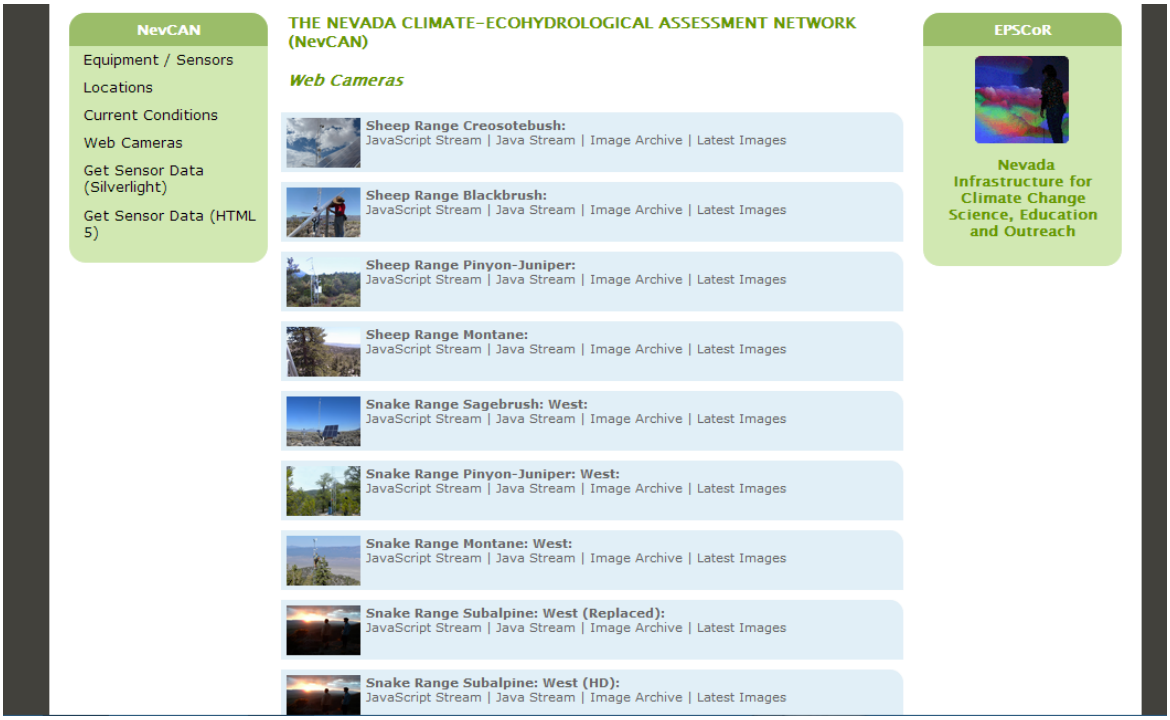

Fig 2.3: Different Locations with cameras installed [19].

More and more sensors and equipment are added each year. As of now a total of 390 sensors are available out of which 240 are standard ones and the other are experimental. The servers for these cameras and data loggers are designated in University of Nevada, Reno. Long distance terrestrial wireless networking is used for the connectivity to these stations. Backups of the database are taken at regular basis to manage the data efficiently and to create consistency in the database. To prevent routing failures and to ensure proper connectivity in different areas parallel links are also provided in different sites [18].

Fig 2.4: View from the camera for Sheep Range Creosotebush [19].

2.2 Equipment and Sensors

The gear and instrumentation that make up every site of the elevation transects have been chosen to screen major hydrologic fluxes/streams, and in addition key adjusting environmental elements, and procedures. Each transect site is

instrumented with a typical arrangement of gear. At lower rise, non-woods locales, all instrumentation is mounted on a 10m guyed tall meteorological tower. At higher height forested destinations, the 10m tower is set to quantify wind rate and bearing,all out sun oriented radiation, and precipitation. The Fig 2.6 shows the different kind of instruments and equipment at the sites.

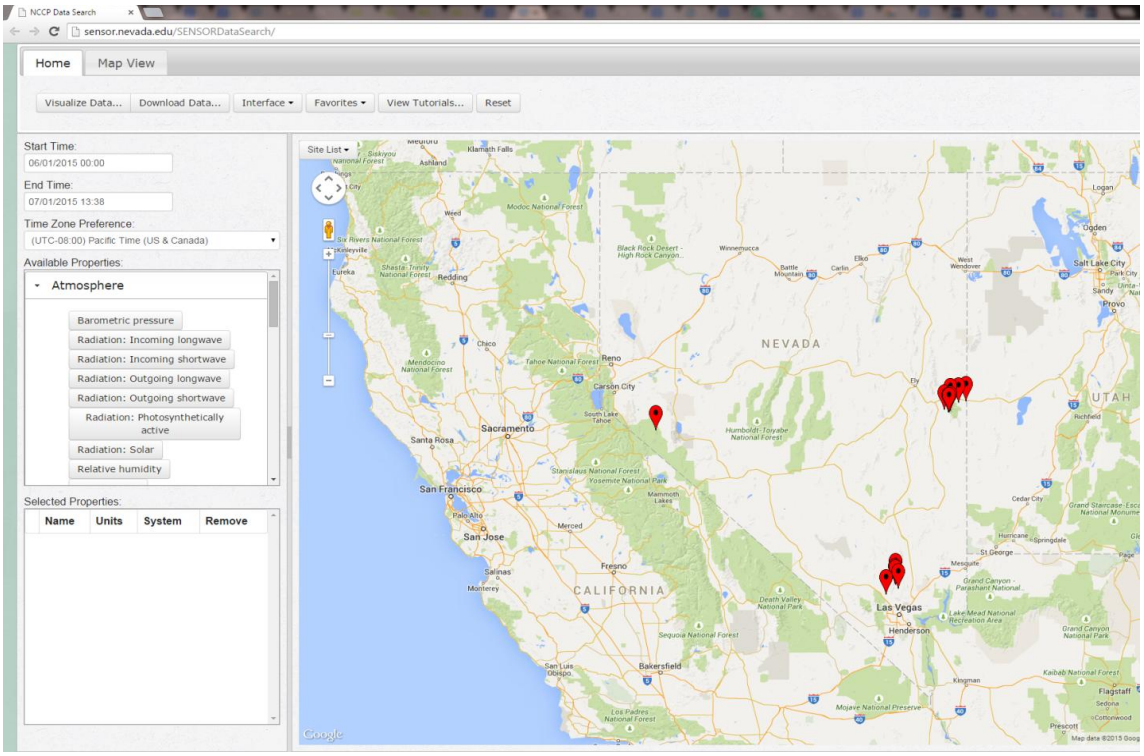

Fig 2.5: Sites and Paramters in NCCP[19]

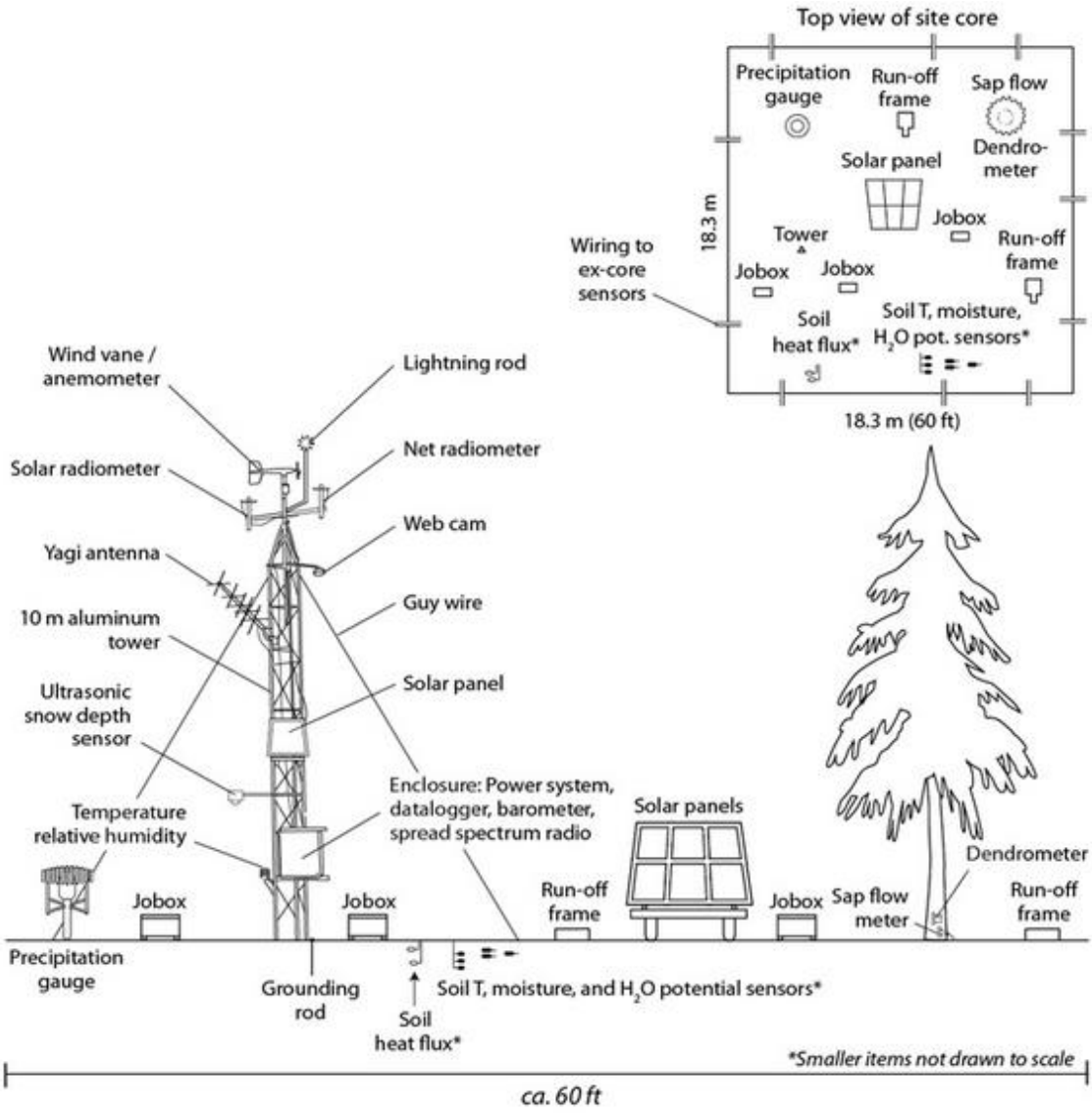

Fig 2.6: Sensors and Hardware Equipment [19]

There is one monitoring system located in the area covering up to 10000 m^2. This allows the following: (a) Complete and accurate representation of plant community with multiple repetitions is achieved at each elevation. (b) Satellite data and gridded data helps in modelling to get appropriate scale. (c) Provide experimental plots so that future studies can be replicated by providing required amount of area. Data measured at each site are collected via sensors hard-wired to Campbell Scientific Inc. multiplexers and data logger(s). All data is eventually transmitted to the data portal constructed by EPSCoR's cyber infrastructure group. The sensors are connected by wires to Campbell Scientific Inc.The data is transferred via radio link to a base station at Great Basin College or to base stations close to Sheep Range sites over a frequency of 900MHz. Finally all data is stored in Servers in UNR and can be accessed through Nevada climate change portal [19].

2.2.1 Variables Measured

Sensors at every tower area are utilized to constantly track various meteorological, abiotic and biotic variables that empower evaluation of atmosphere variability, as well as hydrological and natural reaction to atmosphere variability. Air temperature and air relative dampness are being measured that permit count of evaporative interest of the air. Different variables measured include:

- Precipitation: A gauge called One Geonor, which measures rain and snowfall at high elevations is installed in an open area. An ultrasonic snow profundity sensor

has been set by the Geonor gauge to all the more altogether survey snow profundity and also similarity of the two snow estimations. This kind of tipping container rain gauge is installed at every site.

- Plant canopy interception of snow: Snow stakes have been posted at various sites at high elevations tower locations to measure snow interception. The photographs of these stakes are taken every day by pan-tilt-zoom webcams.
- Soil infiltration and percolation: With the help of transmissometry, water flux is measured vertically and horizontally. Along with this equipment, heat sensors are also installed.
- Subsurface soil water flow: Little width piezometers will be progressed to bedrock at areas encompassing every tower, or in the quick region of the observing weirs, to appraise the potential for shallow subsurface overflow. Every piezometer will be furnished with a weight transducer that will be free of the data logger utilized at the focal point of every plot.
- Soil water content/stocks (volumetric): Loads of plant accessible water exhibit in the top layers of the dirt vadose zone are followed by water metric potential sensors and time space reflectometry tests for the main 30 cm of the dirt. These sensors are co-situated with sap stream and point dendrometer sensors.
- Snow depth: Ultrasonic snow depth sensors have been introduced at every site, to quantify snow depth. To get a website level evaluation of snow depth, and snow

depth conveyance, the graduated snow stakes have been put at plant and between bush microsites for webcam photography of snow depth every day.

- Soil temperature and thermal flux: Copper-constantan thermocouples were introduced at 2.5, 5, 10, 20 and 50 cm soil profundities to log soil temperature.
- Solar radiation measurements: Net radiation and its segments are measured with a Kipp & Zonen, which measures approaching long-wave and short-wave radiation, versus active long-wave and short-wave radiation. A photosynthetic photon flux thickness sensor measures approaching radiation utilized by plants for photosynthesis. The information from this sensor empowers clarification of worldly examples in plant transpiration. [19].

2.2.2 List of Sensors

- Geonor T-200B precipitation gauge
- Hydro Svs TB4 tipping bucket
- Judd acoustic ultrasonic snow depth sensor
- Ambient air temperature thermocouple (OMEGA Copper Const.)
- Kipp & Zonen CNR1 Net radiometer
- Quantum sensor (LiCor 190SA)
- Propeller anemometer (Wind vane, RM Young 05103)
- Barometer(Setra 278)
- Capacitative RH (CSI HMP50)

- Dual probe heat pulse(DPHP) sensor
- Soil water matric potential (Campbell Scientific Inc, 229)
- Soil heat flux(Hukseflux HFP01SC)
- Constantan thermocouple
- Time domain reflectometer (CSI CS616)
- Small diameter piezometers
- Dynamax TDP Sap velocity sensors
- Point dendrometers (Ag. Elec. Corp.)
- VB-C60 PTZ Internet Camera
- Fiber optic distributed temperature (DTS)
- NDVI, WBI ground-based sensors (Skye Instruments, Ltd
- monitoring system
- Pyranometer (LiCor 200SZ)
- Snow stake

- Geonor T-200B precipitation gauge: This Sensor is used for measurement of rain and snow generally at high elevations.
- Hydro Svs TB4 tipping bucket: Sensor measures rain and snowfall under the tree canopy for comparison with inter-plant areas.
- Judd acoustic ultrasonic snow depth sensor:

  This uses ultrasonic pulses to measure snow depth aimed at ground.

- Ambient air temperature thermocouple (OMEGA Copper Const.):

  This is used for measuring the air temperature.

- Kipp & Zonen CNR1 Net radiometer:

  This sensor measures incoming and reflected infrared radiation and also compares them.

- Quantum sensor (LiCor 190SA):

  This sensor is used for measurement of photosythetically radiation data.

- Propeller anemometer (Wind vane, RM Young 05103):

  This sensor is used for measurement of wind speed and direction.

- Barometer(Setra 278):

  This sensor is used for the measurement of barometric pressure.

- Capacitive RH (CSI HMP50):

  This sensor is used for the measurement of air temperature and relative humidity.

- Dual probe heat pulse(DPHP) sensor:

  This sensor is used for the measurement of soil thermal conductivity, diffusivity, and specific heat.

- Soil water matric potential (Campbell Scientific Inc, 229):

  Soil water matric potential (Psi)(-10 to 2500 kPa).

- Soil heat flux(Hukseflux HFP01SC):

  This sensor is used for the measurement of soil heat flux.

- Constantan thermocouple:

This sensor is used for the measurement of soil temperature.

- Time domain reflectometer (CSI CS616):

  This sensor is used for the measurement of volumetric water content, soil water storage, and water infiltration rates.

- Small diameter piezometers:

  This sensor is used for the measurement of pore-water pressure.

- Dynamax TDP Sap velocity sensors:

  Sensors measure sap flow and are installed at all dominant shrub and/or tree species.

- Point Dendrometers (Ag. Elec. Corp.):

  This sensor is used for the measurement of tree stem growth and is installed on all dominant tree species.

- Fiber optic distributed temperature (DTS):

  This sensor is used for the measurement of soil temperature gradients with depth which in turn helps to predict water content in soil.

- Acclima TDT monitoring system:

  This sensor is used for the measurement of soil moisture, salinity and temperature.

- Pyranometer (LiCor 200SZ):

  This sensor is used for the measurement of solar radiation.

- Snow stake

  This sensor is used for the measurement of snow depth using the webcam.

2.3 Conclusion

This chapter listed the data sources and characteristics of the data, different types of equipment installed at different locations and parameters used for measurement. It is clear that the amount of data generated and used by NCCP is huge. Clearly, special methods need to be developed to transmit and store such data efficiently.

CHAPTER 3

INTRODUCTION TO DATA COMPRESSION

3.1 Introduction

In this section we will discuss the basic concepts and different types of data compression in detail.

3.2 Basic Concepts

3.2.1 What is Data Compression ?

Data compression, also known as source coding, means encoding information with fewer bits than the original representation. This encoding can be possible by recognizing the pattern and the underlying structure in the data. Data can be anything from letters, symbols, words, images and videos. It can also be data generated from other processes. Most information that is generated and used is in digitized form and this is the reason data compression is necessary [1].

3.2.2 Types of Data Compression

Data compression can be classified as lossless and lossy compression. In lossless compression the original information can be retrieved back without any loss, whereas in lossy compression methods, some loss is incurred when the reconstruction take place. So the original information is not retrieved with lossy compression. Depending on the

application one needs to decide if lossy or lossless compression has to be used. For example, when we consider text compression one cannot choose to go for lossy compression as we know that in lossy compression some loss is incurred. As text involves characters and if any character is missed or there is a discrepancy then it makes a huge difference with respect to the original data. In the applications where exact representation of the original data is not required one can go for lossy compression. For example, during audio compression lossy compression is favorable as accurate value of each sample is not required. The loss of information can be tolerated in different amounts depending on the desired quality of the reconstructed signal [16].

3.3 Lossless Data Compression

In lossless compression, the original signal can be reconstructed without loss of any information. It is generally used in areas where exact reconstruction or the accuracy of the original information is utmost important. For example, one of the applications for lossless compression is text compression where the original information or signal is very important. This is because a small change in original information can create serious issues. For example, if banking related data is concerned, a small change can cause problems with millions of dollars. If the integrity of the information needs to be preserved, then one needs to go for lossless compression. For example, if a radiograph is compressed using lossy image compression technique and some features have been lost then, lossy compression could cause a serious impact as it is in concern with a human life.

So it is necessary to use lossless compression techniques in such applications. It is also not advisable to go for lossy compression with satellite related data. One can obtain many predictions with respect to vegetation and climate etc. If the reconstructed signal is not same as original information then there will be a huge change and it is not possible to obtain the same data all over again. So, it depends on the application or the area where data compression is applied [1].There are different types of lossless compression techniques which are explained as below:

3.3.1 Huffman Coding

David Huffman is the inventor of Huffman coding. It was designed as part of his class assignment in information theory course at MIT. The codes generated using this procedure is called a Huffman code. The procedure for Huffman coding is based observations with respect to optimum prefix codes. The first observation is that shorter codewords are assigned to symbols that occur more frequently and longer code words are assigned to symbols that occur less frequently. The second observation is that, if there are two symbols whose frequency of occurrence is the least then their codewords will have the same length. The first observation is logically correct. If the smaller code words are assigned to symbols which occur more frequently then the number of bits assigned per symbol will be more. So it is not optimum to assign longer code words to symbols that occur more frequently.

Now for the second observation, let us consider an example. There are two codewords which correspond to two symbols which are not of the same length and less probable. Also let us consider that the longer codeword is k bits longer than the shorter codeword. So, the shorter codeword cannot be a prefix to the longer codeword which means that even if the k bits of the longer codeword are dropped, still the two codewords will be different.

The shorter codeword cannot become a prefix to any other codeword as these codewords have least frequency of occurrence and the length of any other codeword cannot be longer than these codewords. The observation for optimal code also holds true. With this a new codeword which has a smaller average length is obtained by dropping the k bits.

We need to frame some requirements so that it does not violate the two observations as given in the above paragraph. So, the requirement is that the two least frequently occurring symbols will have only one digit that differs. For example, if there is a code m which has zeros and ones, then the two codewords can be m*0 and m*1[1].

The Huffman algorithm is simple and can be described in terms of creating a Huffman code tree. The Huffman code can be explained by creating a Huffman code tree and the procedure for this is given as below:

1. Consider each node corresponds to a symbol in the alphabet and start from that list.

2. Two nodes with least weight should be selected.

3. A parent node for these two child nodes need to be created and the weight must be equal to sum of the two child nodes.

4. The two child nodes need to be removed and this parent node is added to the list of free nodes.

5. This process is repeated from step2 till a single tree is obtained. Then a prefix code is created by travelling from root to the node of a binary tree. Zero is assigned for left branch and one is assigned to the right branch [20].

3.4 Lossy Data Compression

Lossy compression techniques involve some loss of information, and data that is compressed using lossy techniques generally cannot be recovered or reconstructed exactly. In return for accepting this distortion in the reconstruction, we can generally obtain much higher compression ratios than lossless compression. In many applications, this lack of exact reconstruction is not a problem. For example, when storing or transmitting speech, the exact value of each sample of speech is not necessary. Depending on the quality of the reconstructed speech, varying amounts of loss of information of each sample can be tolerated. If the quality of the reconstructed speech is to be similar to that heard on the telephone, a significant loss of information can be tolerated. However, if the reconstructed speech needs to be of the quality heard on a

compact disc, the amount of information loss that can be tolerated is much lower. Similarly, when viewing a reconstruction of a video sequence, the fact that the reconstruction is different from the original is generally not important as long as the differences do not result in annoying artifacts. Thus, video is generally compressed using lossy compression. Once we have developed a data compression scheme, we need to be able to measure its performance. Because there are different areas of application, different terms have been developed to describe and measure the performance [1].

3.5 Differential Encoding

In many sources of interest, the sampled source output x_n does not change a great deal from one sample to the next. This means that both the dynamic range and the variance of the sequence of differences $d_n = x_n - x_{n-1}$ are significantly smaller than that of the source output sequence. Furthermore, for correlated sources the distribution of d_n is highly peaked at zero. Given the relationship between the variance of the quantizer input and the incurred quantization error, it is also useful, to look at ways to encode the difference from one sample to the next rather than encoding the actual sample value. Techniques that transmit information by encoding differences are called differential encoding techniques [1].

3.6 Conclusion

In this chapter we have covered the need for data compression, types of data compression, and their techniques in detail.

CHAPTER 4

ARTIFICIAL NEURAL NETWORK

4.1 Introduction

Artificial neural networks (ANN) are models used to gauge or approximate functions which have a large number of inputs and outputs. They are inspired by biological neural networks of animals. ANNs generally are interconnected system of neurons, which communicate with each other. The neurons are connected to each other with the help of links which bear weights and ANNs can be tuned based on experience, making them capable to learn.

Let us consider an example of handwriting recognition. Here the input is an image where the neurons are actuated by pixels of the picture. The activation of these neurons is transmitted to other neurons after it is changed by a function. This process stops after the output neuron is activated which tells us the character that was read from the picture. ANNs can solve wide variety of problems as it can learn from data like applications in rule-based programming, speech recognition, computer vision etc.

4.2 Types of Neural Network

There are many types of ANNs. They are the models that are used to approximate functions that are unknown. They are inspired from biological neural networks. It works in a similar way like the biological process, for example the electrical signals are

transmitted through the neurons between the input and output from the brain. ANNs are their biological counterparts, which are efficient and effective at what they are intended to do, but they bear only slight resemblance to their biological counterparts. Some ANNs are adaptive systems and are used for example to model populations and environments, which constantly change.

ANNs are also used to model populations and environment which change frequently. ANNs can be built using hardware and also through software. There are different types of ANNs such as feedforward neural network, radial basis function network, elman neural network, cascade feed forward neural network(CFNN), physical neural network etc.

4.3 Multilayer Perceptron

The Multilayer Perceptron (MLP) model consists of multiple layers of nodes, with each layer connected to the next one usually in a feed forward way. It consists of either three or more than three layers: input layer, one or more hidden layers, and an output layer of nonlinearly-activating nodes. It is thus considered a deep neural network. Fig 4.1 describes a basic MLP neural network model. Each node in one layer connects with a certain weight Wij to every node in the following layer. This model maps set of input data onto a set of appropriate outputs. Each neuron in one layer has direct connections to the neurons of the subsequent layer [21]. Except for the input nodes, each node is a

neuron with a nonlinear activation function. In our work the sigmoid function is used as the activation function which is used widely in many applications.

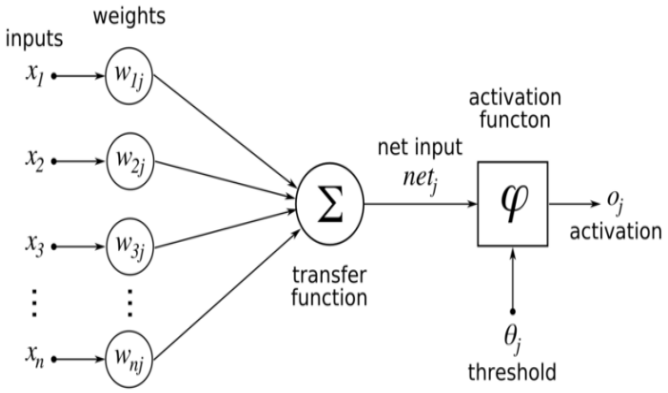

Fig 4.1: Multilayer Perceptron [21].

MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable. The weight of the connection is updated after every entry in the input data is handled. This process is called supervised learning. The amount of learning depends on how small the error is or how close the predicted value is to the expected result. Using this information, the algorithm adjusts the weights of connections so that error function's value is reduced by a small amount. This process is performed multiple times until the error reduces to small numbers. In our experiment we have applied gradient descent non-linear optimization. For this, the derivative of the error

function with respect to the network weights is calculated, and the weights are then changed such that the error decreases [21].

MLP utilizes back propagation algorithm, which is the standard algorithm for supervised learning pattern recognition process. Also, it is the subject of ongoing research in computational neuroscience and parallel distributed processing. These are widely used in research activities due to their ability to solve problems to, get approximate solutions for extremely complex problems.

The algorithm works as below:

Given a set of k-dimensional inputs represented as a column vector as given below:

$$\vec{x} = [x_1, x_2, \ldots x_k]^T \tag{4.1}$$

And a nonlinear neuron with synaptic weights from the inputs:

$$\vec{w} = [w_1, w_2, \ldots, w_k]^T \tag{4.2}$$

Then the output of the neuron is defined as follows:

$$y = \varphi(\vec{w}^T \vec{x}) = \varphi(\sum_{i=1}^{k} w_i x_i) \tag{4.3}$$

We will assume that the sigmoidal function is the simple logistic function as below:

$$\varphi(u) = \frac{1}{1+e^{-u}} \tag{4.4}$$

This function has the useful property that such that:

$$\frac{d\varphi}{du} = \varphi(u)(1 - \varphi(u)) \tag{4.5}$$

Feed forward back propagation is typically applied to multiple layers of neurons, where the inputs are called the input layer, the layer of neurons that take the inputs is called the hidden layer, and the next layer of neurons take inputs from the outputs of the hidden layer is called the output layer. There is no direct connectivity between the output layer and the input layer.

If there are inputs, hidden neurons, output neurons, and the weights from inputs to hidden neurons are (i being the input index and j being the hidden neuron index), and the weights from hidden neurons to output neurons are (i being the hidden neuron index and j being the output neuron index), then the equations for the network are as follows:

$$n_{Hj} = \sum_{i=1}^{N_I} w_{Hij} x_i, j \in \{1, 2, \ldots, N_H\} \tag{4.6}$$

$$y_{Hj} = \varphi(n_{Hj}) \tag{4.7}$$

$$n_{Oj} = \sum_{i=1}^{N_H} w_{Oij} y_{Hi}, j \in \{1, 2, \ldots, N_O\} \tag{4.8}$$

$$y_{Oj} = \varphi(n_{Oj}) \tag{4.9}$$

If the desired outputs for a given input vector are, then the update rules for the weights are as follows:

$$\delta_{Oj} = (t_j - y_{Oj}) \tag{4.10}$$

$$\Delta w_{Oij} = \eta \delta_{Oj} y_{Hi} \tag{4.11}$$

$$\delta_{Hj} = (\sum_{k=1}^{N_O} \delta_{Ok} w_{Ojk}) y_{Hj} (1 - y_{Hj}) \tag{4.12}$$

$$\Delta w_{Hij} = \eta \delta_{Hj} x_i \tag{4.13}$$

Where η is some small learning rate, δ_{oj} is an error term for output neuron j and δ_{hj} is a back propagated error term for hidden neuron j.

4.4 Cascade Feedforward Neural Network

This model of neural networks is similar to the feed forward neural networks in a way that the output from every layer is connected to the next layer. The components include input layer, hidden layers, and output layer like other neural networks [15]. In this case, each input is connected to every other hidden layer in the form of a cascade, because of which it can learn associations of high complexity. Fig 4.2 illustrates the basic CFNN model. The input layer consists of multi-dimensional vectors. Apart from the inputs, the bias is fed into each of the hidden and output neurons. In the hidden layer, each input neuron is multiplied by a weight, and the resulting weighted values are added together to produce a combined value. The weighted sum is fed into a transfer function, which then outputs a value. The outputs from the hidden layer are distributed to the output layer that receives values from all of the input neurons (including the bias) and all of the hidden layer neurons. Each value presented to an output neuron is multiplied by a

weight, and the resulting weighted values are added together again to produce a combined value [15].

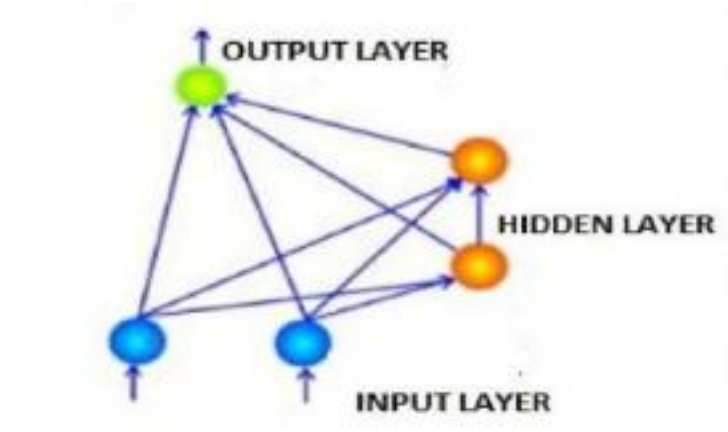

Fig 4.2: Cascade Feedforward Neural Network [15].

4.5 Conclusion

In this chapter we have dealt with the different types of ANNs like MLP and CFNN.

CHAPTER 5

DIFFERENT METHODS TO COMPRESS CLIMATE DATA

5.1 Introduction

In this section, we will describe different methods implemented to compress the climate data.

5.2 Compression of Weather Forecast Data

This method describes the way in which data compression is performed on the data obtained from weather prediction models. In this method the speed of encoding and decoding, the complexity of coding techniques, precision of the decoded data are considered along with the compatibility with Gridded Binary(GRIB) [2]. GRIB is a data format generally used in meteorology to store weather data.

Predictive coding is the best method when compared to discrete cosine transform and wavelet based coding in this context. The following steps will describe the process:-

- Quantization: The various types of forecast fields have different ranges and different distributions within these ranges. For example, the range of altitude might increase from 800 m(meters) at 1000 hPa(hectopascal) to 3000 m at 50 hPa. Furthermore, the fields are only known within the limits of the observation errors. These errors can serve as guidelines for precision requirements imposed on the compression algorithms. For altitude, the maximal observation errors are assumed to be 5 m at 1000 hPa and 25 m at 50 hPa. Based on the errors, quantization of the

data can be performed within these tolerances. For a large dataset, the requirements are less severe. Therefore, before uniform quantization is applied, it is required to perform a logarithmic transformation. After this data is encoded, exact reconstruction of the quantized values can be obtained.

- Optimal Prediction: The data can be predicted from surrounding data points as there is a lot of spatial coherence. It is enough to store the differences between the actual and estimated data points. The differences need to be small so that it can be encoded with fewer bits. The data is predicted by taking linear combination of few reconstructed data points.
- Entropy coding: The data can be further compressed by techniques like Huffman and Arithmetic coding.

The following parameters were considered:

1. Pressure

   The high surfaces have low pressure and high tolerance. They are relatively smooth as compared to surfaces which are low. Table 5.1 shows the result. To compare the GRIB format with the result given in Table 5.1, the number of bits needed to represent a single data value at a predetermined accuracy is calculated in GRIB format. This number can be directly compared to the bits per sample(bps) column in Table 5.1.

isobaric surface hPa	range m	accuracy m	file size kB	bit rate bps	comp. ratio
1000	753	5	5.3	1.5	4.7
950	765	6	4.9	1.4	5.0
850	814	6	4.6	1.3	5.4
700	980	7	4.2	1.2	5.8
500	1261	10	3.9	1.1	6.4
400	1465	12	3.8	1.1	6.4
300	1741	13	4.0	1.1	6.4
250	1932	14	3.9	1.1	6.4
200	2128	16	3.4	1.0	7.0
100	2485	22	2.1	0.6	11.7
50	2875	25	1.7	0.5	14.0

Table 5.1. Result of datasets containing different constant pressure surfaces [2].

2. Precipitation

For precipitation data, it is not sufficient to just apply uniform quantization as in the previous case as much higher resolution is required, in this case. The data is quantized using a threshold S and values less than this threshold are considered zero. The logarithmic transformation is applied to the values greater than or equal to the threshold.

$$x = \text{round}(10\log_{10}(x) / R) \tag{5.1}$$

At the decoder the number $10\log_{10}(x)=R+\delta$ with a rounding error of $|\delta| \leq 0.5$ will be received. Thus, the reconstructed value x' of x will be

$$x' = 10^{(\log(x) + \delta R/10)} = 10^{\delta R/10}\, x \tag{5.2}$$

Then the relative error is

$$|x'-x/x| \leq 10^{R/20} - 1 = 0.115R \quad (5.3)$$

For example, with a logarithmic resolution R = 0.1dB, a relative error bound of 1.15% is obtained. Here precipitation is measured in kg/m2 and the threshold was set to S = 0:001 kg/m2 as shown in Table 5.2.Here, the performance cannot be compared directly with GRIB as logarithmically scaled files are not supported.

data type	range kg/m^2	accuracy	file size kB	bps
large scale	0 – 111	11.50%	4.4	3.0
precipitation		1.15%	9.2	6.4
convective	0 – 230	11.50%	6.3	3.7
precipitation		1.15%	11.4	7.8

Table 5.2. Compression result for precipitation datasets [2].

data type	range	accuracy	file size	bps
wind angle	36^o	5.0^o	3.1 kB	2.1
		0.5^o	7.4 kB	5.1
wind speed	13 m/s	0.50 m/s	1.9 kB	1.3
		0.05 m/s	5.5 kB	3.8

Table 5.3. Compression result for a wind dataset [2].

3. Wind

Wind data is usually given as 2 dimensional vector but the precision requirements are more intuitively based on wind direction and wind speed. So the data is converted to polar coordinate form by transforming the data. Then, the data can

be quantized informally. Table 5.3 summarizes the results for one representative data.

4. Other data

   The experiment can be completed with studies on dataset for the roughness-length and diffusion coefficient, see Tables 5.4 and 5.5.[2]

data type	range	accuracy	file size	bps
diffusion coefficient	0.0003 – 0.067	$5.5 \cdot 10^{-5}$ $5.5 \cdot 10^{-6}$	7.7 kB 14.4 kB	5.3 9.9

Table 5.4. Compression result for a diffusion coefficient dataset [2].

data type	range	accuracy	file size	bps
roughness length	0.000001 – 6.8	11.5% 1.15%	3.8 kB 8.7 kB	2.6 6.0

Table 5.5. Compression result for a roughness length dataset [2].

5.3 Compression of Temperature Data by Using Daubechies Wavelets

This method describes the compression of temperature data in Kuala Lumpur from the year 1948 to 2010 using wavelets. Here hard thresholding is used for compression of the data.

Below steps are followed for compression of temperature data:

1. Consider the input signal with length N.
2. Apply Wavelet transform to the data.
3. Hard thresholding values are found. The values which lie below the threshold are set to zero.

4. Only those values which are not zero are kept. Then wavelet compression is applied to the actual data along with the threshold values.
5. Then inverse discrete wavelet transform is applied for reconstruction of the signal and this gives an approximation of the original signal.

Below are the details on hard thresholding :
Say, we are given wavelet coefficient w and threshold value λ, the hard threshold value of the coefficient can be written as:

$$\eta_{hard}(w,\lambda) = w\ I(|w|>\lambda) \tag{5.4}$$

where I is the usual indicator function. The reason for using the wavelet function is that it is similar to the shape of the original data. Figure 5.1 shows plot of the original series. Before we do the compression, we decompose the original time series (signal) by using Daubechies 4 (D4) at level 5. From Figure 5.2, we can notice that at level 5, the approximations (a5) and detail (d5) have shown clearly the shape and characteristics of the original data. In other words, all characteristics of the data have been exactly recaptured via multiresolution analysis(MRA) or wavelet decomposition. This is why wavelets are so efficient for time series analysis [4].

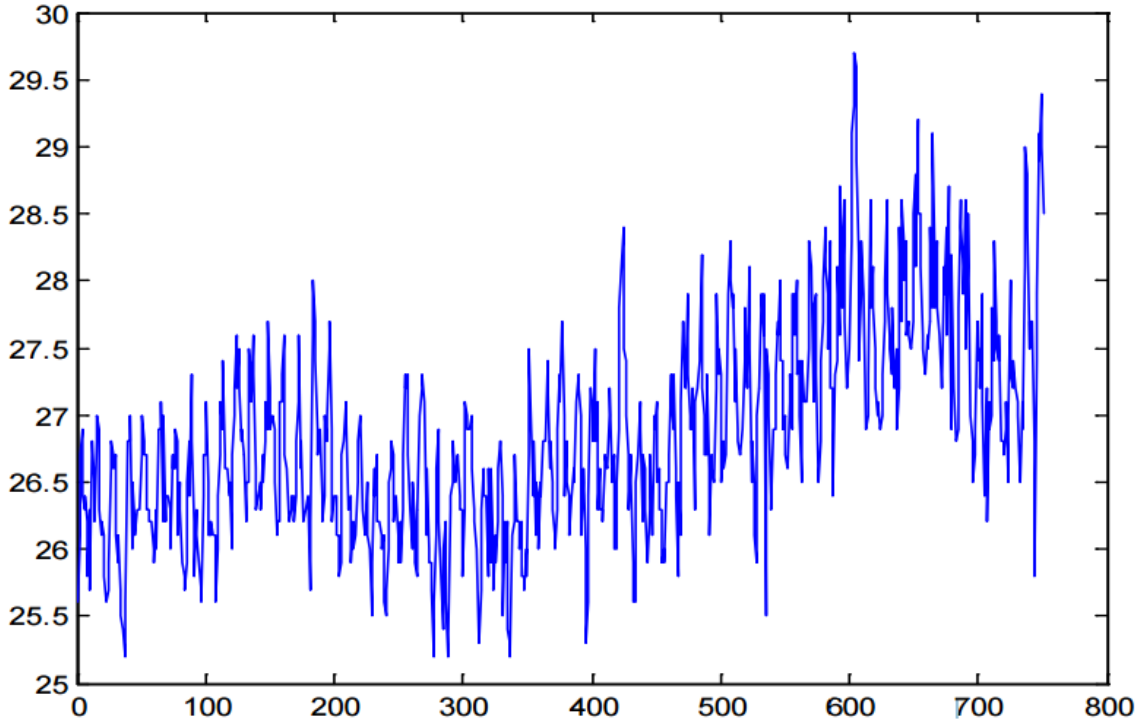

Fig 5.1. Original Series [4].

The smooth version of the original signal is obtained after the high frequencies are filtered out. The original signal can be reconstructed back at level 5 by the summation of (d1+d2+d3+d4+d5+a5).This can be seen from the figure 5.3. Here, it can be seen that the compressed signal resembles the original signal. Indeed, based on statistical results in Table 5.6, the RMSE is 1.23×10^{-3} and CR is 1:10. This indicates that the compressed signal is quite good in term of compression quality. By using D4 wavelet and applying level 5 compression with hard threshold value (0.8166), good compression is obtained. Figure 5.3 shows the wavelet decomposition of original temperature data. We can reconstruct the signal by adding details from level 1 until level 5 and approximation at level 5 [4].

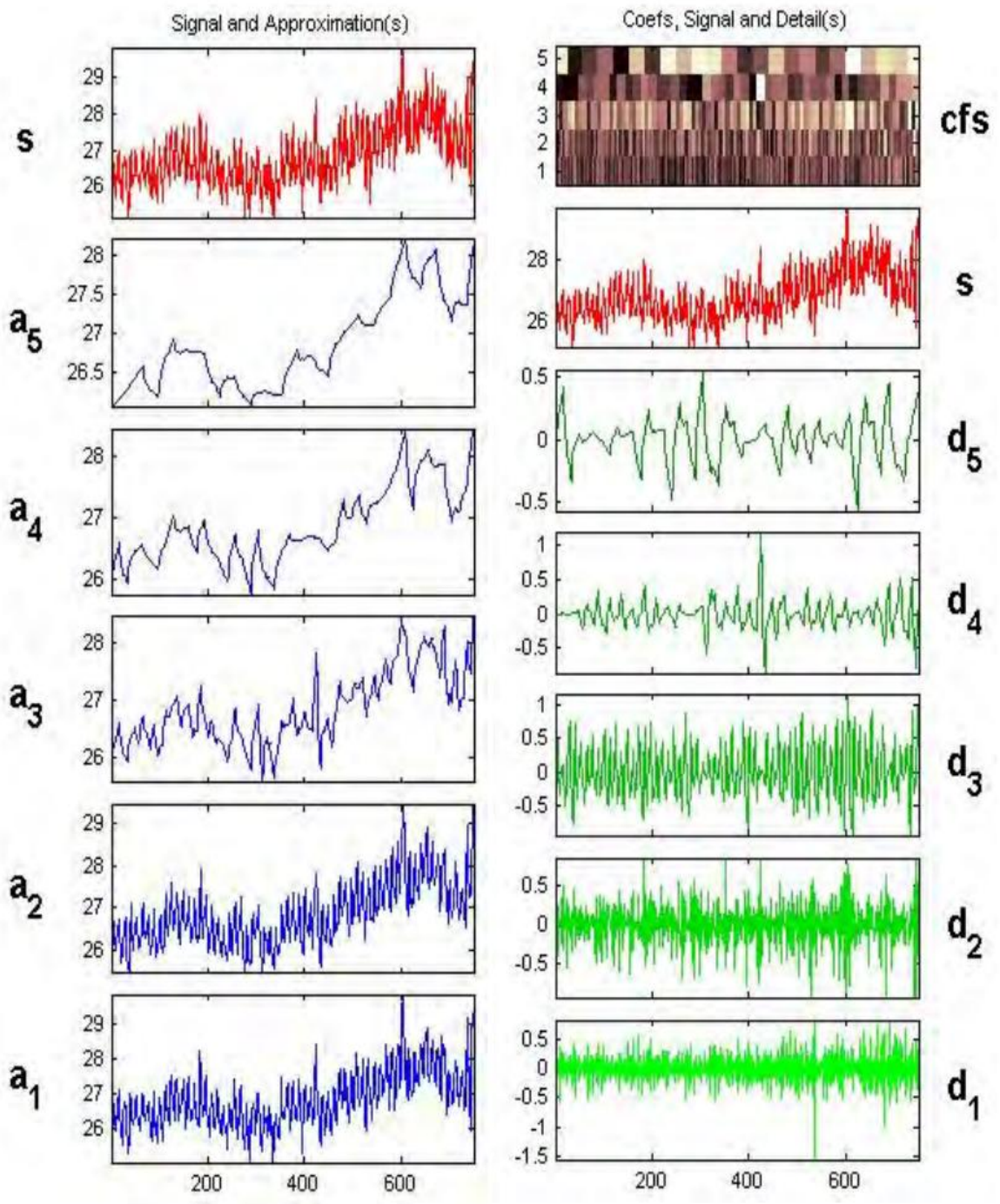

Fig 5.2. Wavelet decomposition at level 5 for original temperature data in Kuala Lumpur by using D4- left (approximation) and right (detail) [4].

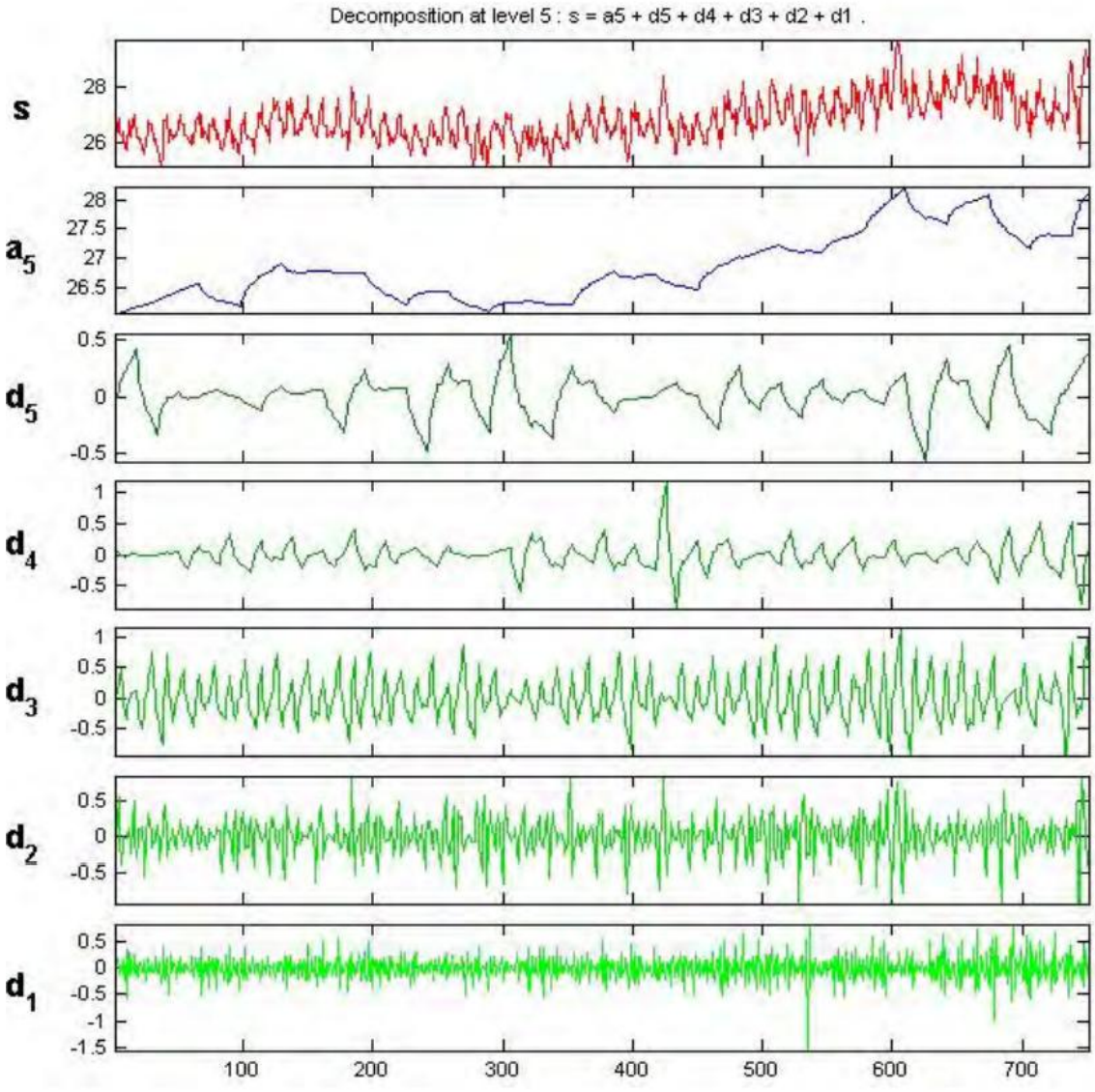

Fig 5.3: Reconstructed signal d1+d2+d3+d4+d5+a5 [4].

Wavelet	*level*	*Retained energy (%)*	*Zeroes detail (%)*	*RMSE*	*Standard Deviation (SD)*	*Median Absolute Deviation (MEAD)*	*Mean Absolute Deviation (MAD)*	*Compression Ratio (CR)*
D 4	5	99.99	84.95	1.23×10^{-3}	0.3312	0.2205	0.2667	1:10

Table 5.6 .Statistical Analysis for Compression Time Series by Using D4 [4].

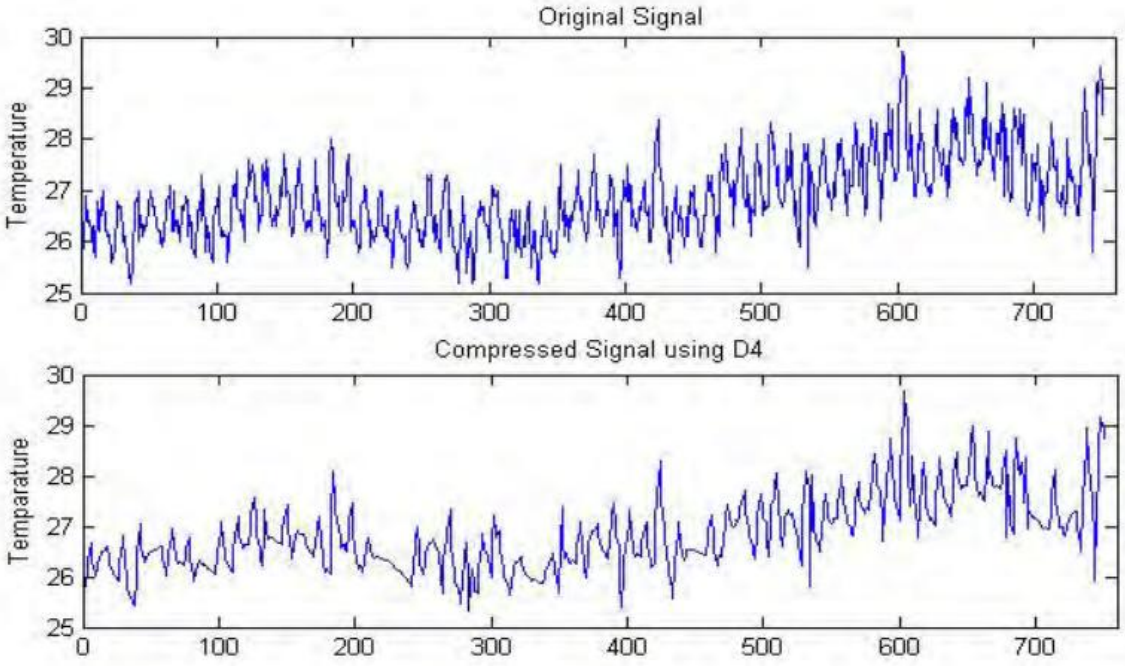

Fig 5.4.Compressed original signal (temperature data) at level 5 using D4 [4].

5.4 Data Compression Technique for Modeling of Global Solar Radiation

This method describes the use of wavelet transform and curve fitting method for compression of solar radiation data. To analyze the solar radiation data, symlet 6 and wavelets were used.

After the solar radiation data is decomposed to level 4, curve fitting method is used to derive equation to predict the horizontal solar radiation. Both polynomial fit and sinusoidal fit can be applied to the filtered solar radiation data. High correlation factors are observed for both the fits.

Fig.5.4 shows the steps to perform this method. This method combines the conventional curve fitting method and wavelet method. In general, raw data is used to find an appropriate curve fit. By using wavelet filtering technique, entire range of data can be represented within the time frame [23].

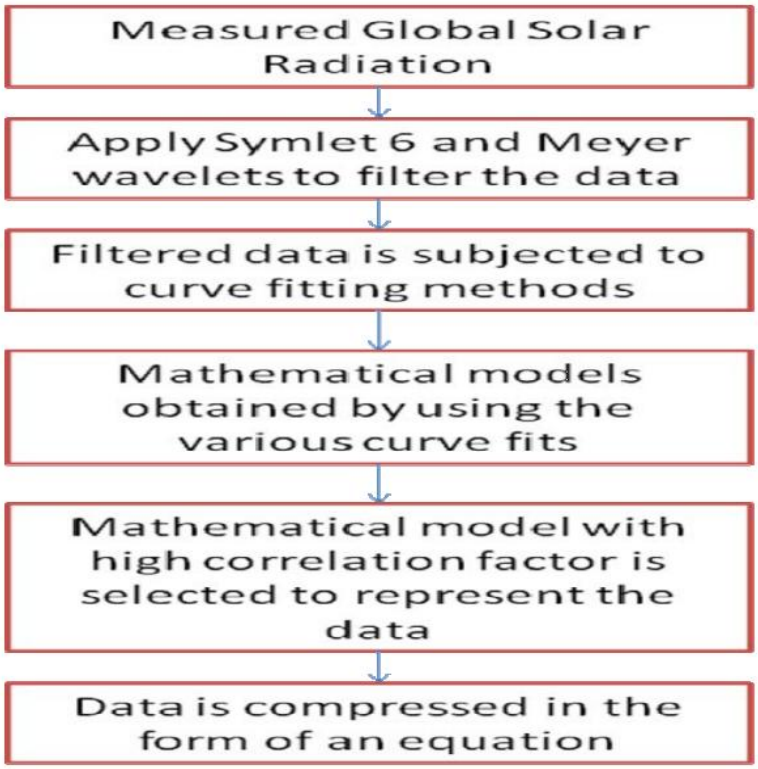

Fig 5.5.Methodology used to compress the data [23].

The solar radiation data on 13th, 16th, and 19th January 2011 is considered for the research. Fig. 5.6 shows the daily average of measured solar radiation data. The wavelet transform

is useful for the data analysis as peaks are preserved in this method. Extreme frequencies in the measured solar radiation data are filtered out from level 1 to level 4. At level 4, symmetric details that correspond to approximations at level 3 are obtained. This shows that the data fitting method can be used to model solar radiation data with high correlation factor.

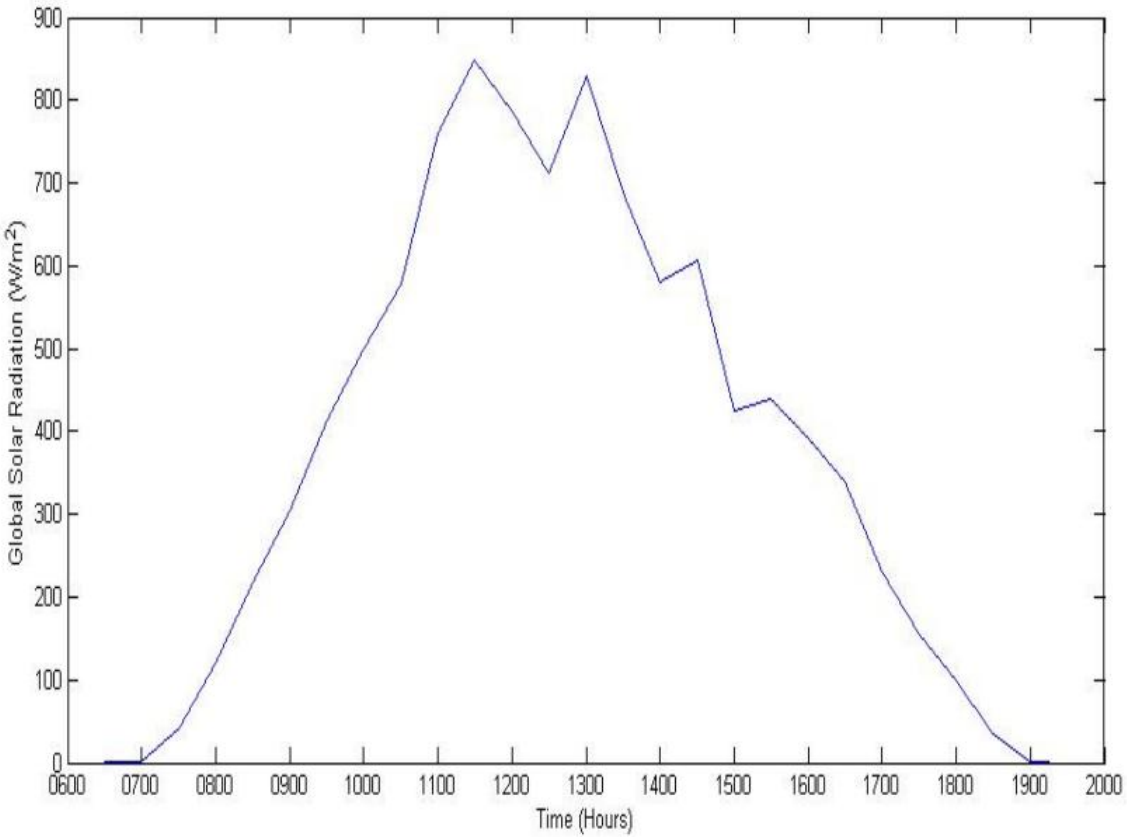

Fig 5.6. Daily average of measured solar radiation data in UTP, Malaysia [23].

From the figure 5.7, it can be clearly seen that meyer wavelet gives better filtered data that is close to the measured data. Based on the proposed method, as given in Fig. 5.4, next step is applied to obtain the appropriate mathematical model.

Modeling equation can be computed from the compressed signal with the help of symlet 6 and meyer wavelet. The polynomial curve fit gives a 99.8% correlation by using meyer wavelet approach. The polynomial coefficients are as given in Table 5.8 [23].

Measurement	**Wavelets**	
	Symlet 6	**Meyer**
Thresholding values	$\lambda_j = 113.6$	$\lambda_j = 123.8$
NOZ (%)	56.34	51.98
RE (%)	99.04	99.85
RMSE (%)	0.0913	0.0961

Table 5.7. Important parameters for the selection of suitable filtered datasets for applying curve fitting method [23].

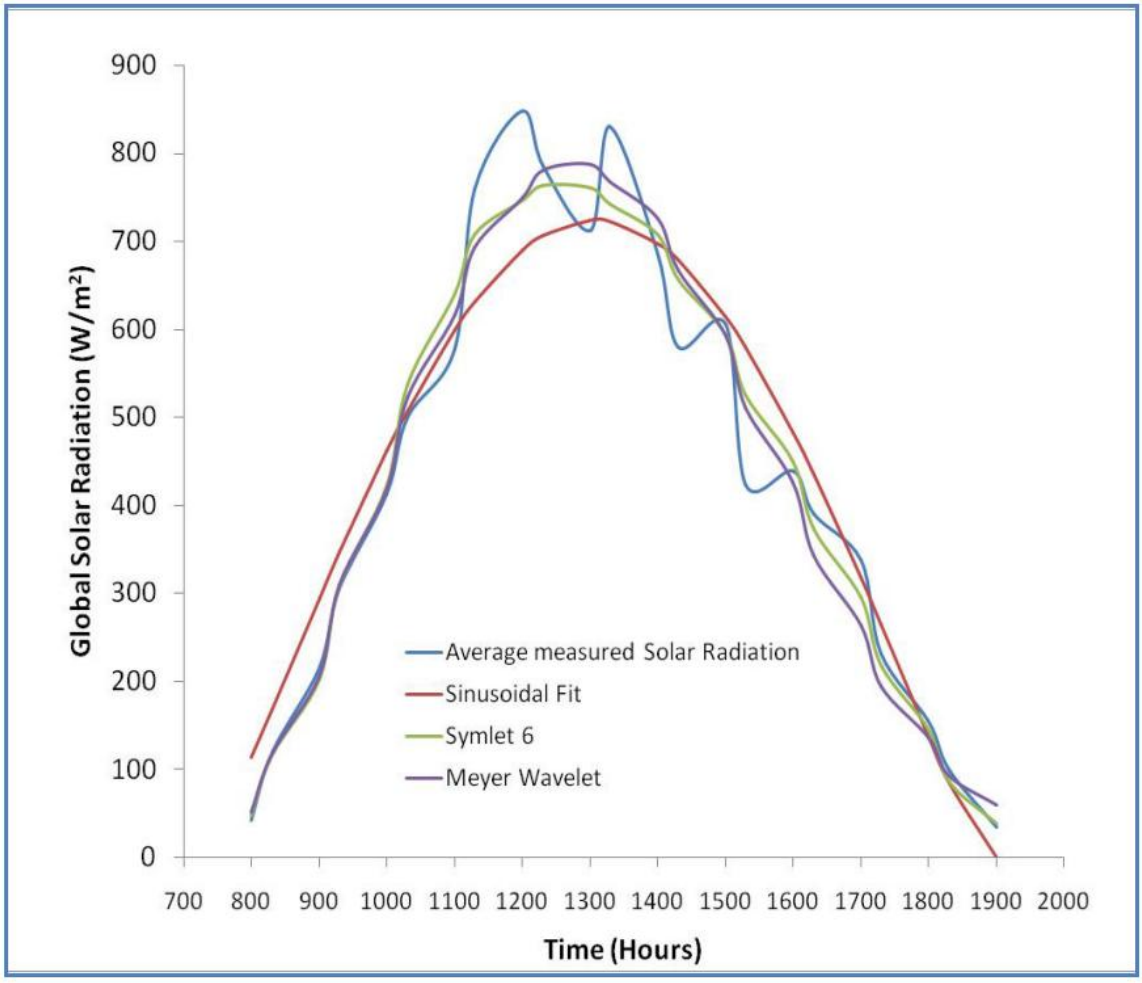

Fig 5.7.The various data compression techniques applied to the actual averaged measured global solar radiation data [23].

Coefficient	Value
a	2.43×10^{4}
b	-1.01×10^{2}
c	1.57×10^{-1}
d	-1.13×10^{-4}
e	3.81×10^{-8}
f	-4.90×10^{-12}

Table 5.8. Coefficients for polynomial fitting [23].

5.5 Conclusion

We have discussed different methodologies to compress climate data. All the above discussed methods have used lossy compression techniques.

CHAPTER 6

VISUALIZATION AND COMPRESSION OF CLIMATE DATA USING NEURAL NETWORKS

6.1 Introduction

In this chapter, we will focus on the parameters considered, site locations, properties and characteristics of the parameters, and propose an algorithm for data compression.

6.2 Parameters Considered

Parameters considered for the research are:

- Solar radiation data
- Photosynthetically active radiation data
- Precipitation data

6.3 Sites Considered

We have considered the following sites for the research:

- Sheep Range Black Brush
- Sheep Range East Sagebrush

The sheep range is located approximately 35km NNW(north north west) of Las Vegas. The sheep range black brush and sheep range east sagebrush regions are located at an altitude 1670m and 3015m respectively. For solar radiation and photosynthetically active radiation data, we have chosen the sheep range back brush region and for precipitation data, we have chosen sheep range east sagebrush region.

6.4 Data description

For all the parameters considered, we have used Minute-wise data points. The solar radiation data consists of leading zeros followed by non-zero data points and trailing zeros at the end. For this, we have considered w/m^2 (watt per meter square) as the unit of measurement.

The photosynthetically active radiation data looks similar to solar radiation data which also has leading zeros followed by non-zero data points and trailing zeros at the end. Even for this, we will consider w/m^2 as the unit of measurement.

For precipitation data, we have chosen sheep range east sagebrush region as the rainfall is scarce in sheep range black brush region. The unit for measurement for precipitation data is mm(millimeter). The precipitation data contains floating point numbers. It has one digit after the decimal point.

6.5 Data Visualization

Below are the graphs for different parameters for the year 2013 from NCCP for some of the months. All the data is minute wise.

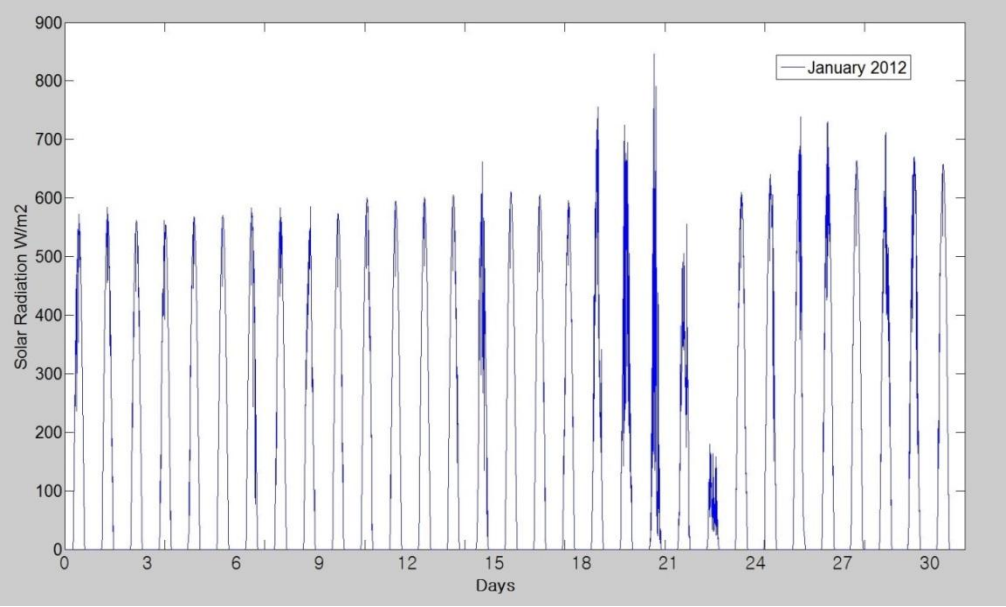

Fig 6.1: Visualization of Solar Radiation for January 2012.

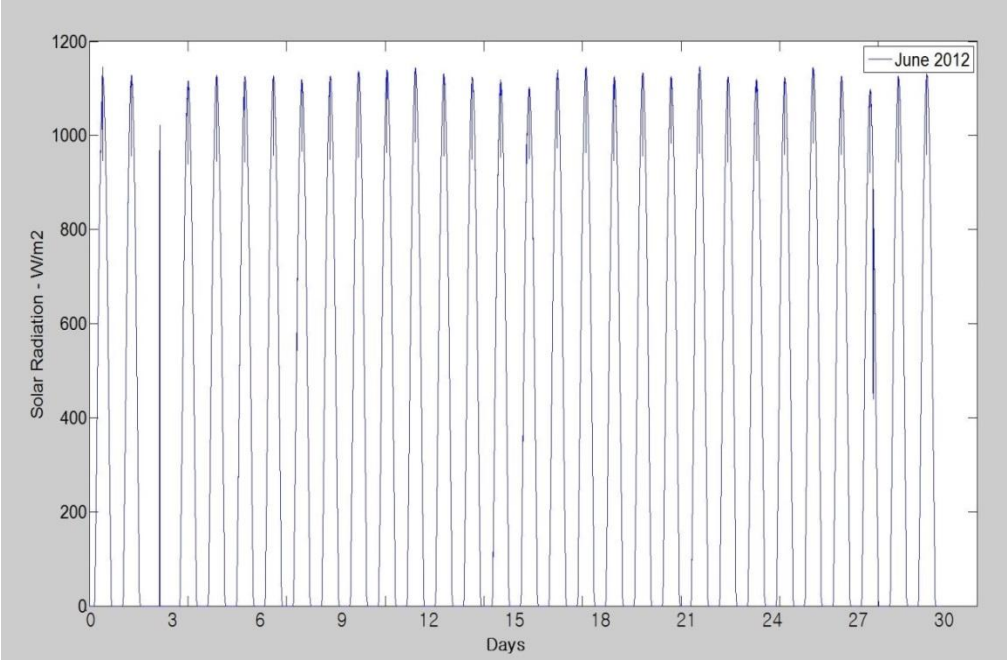

Fig 6.2: Visualization of Solar Radiation for June 2012.

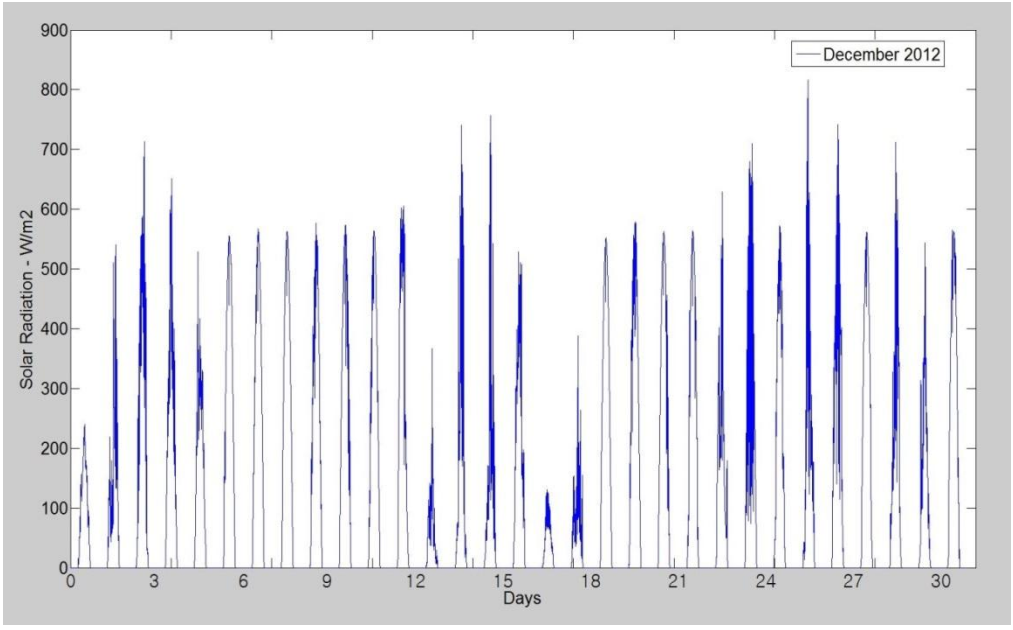

Fig 6.3: Visualization of Solar Radiation for December 2012.

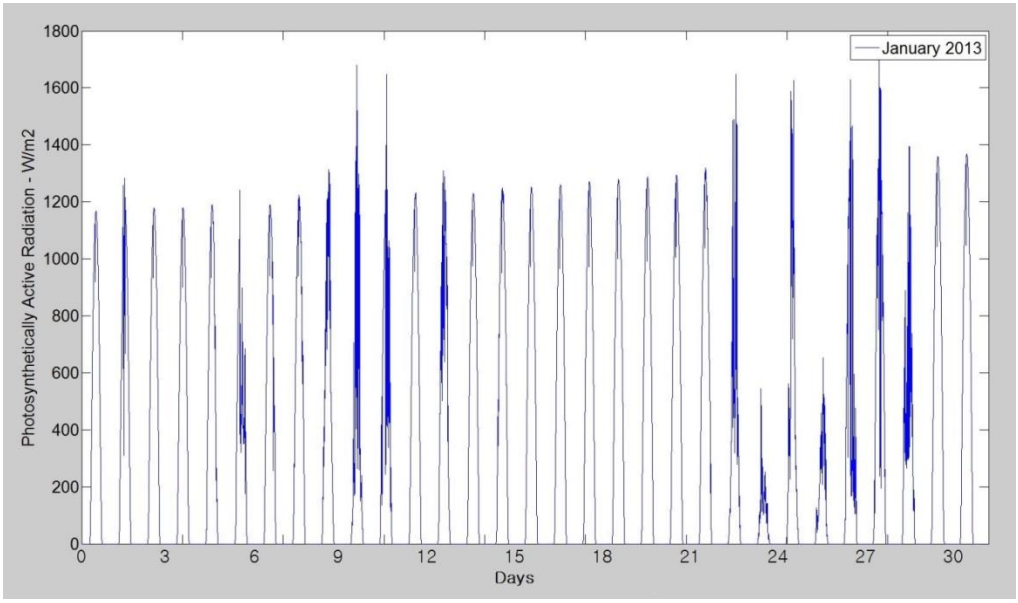

Fig 6.4: Visualization of Photosynthetically Active Radiation Data for January 2013.

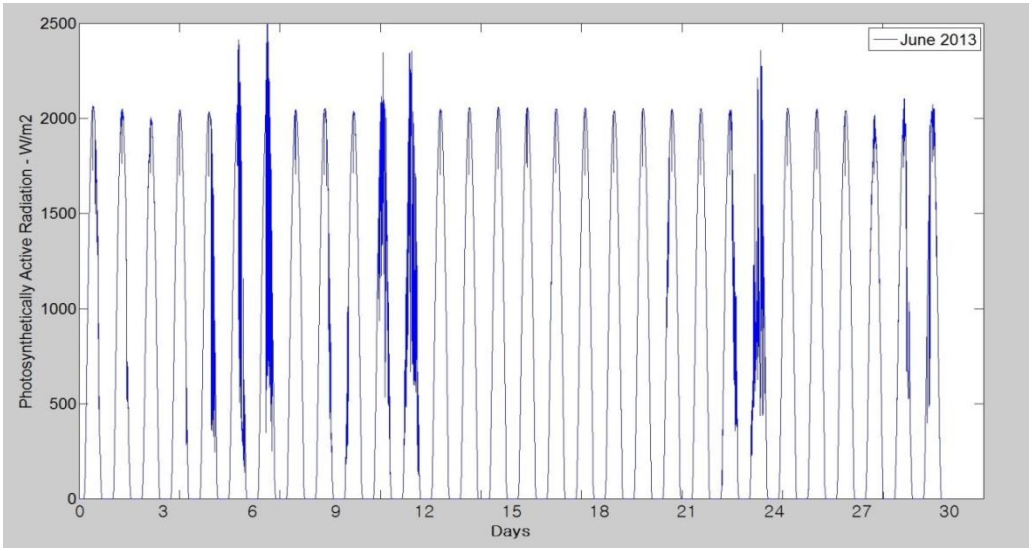

Fig 6.5: Visualization of Photosynthetically Active Radiation Data for January 2013.

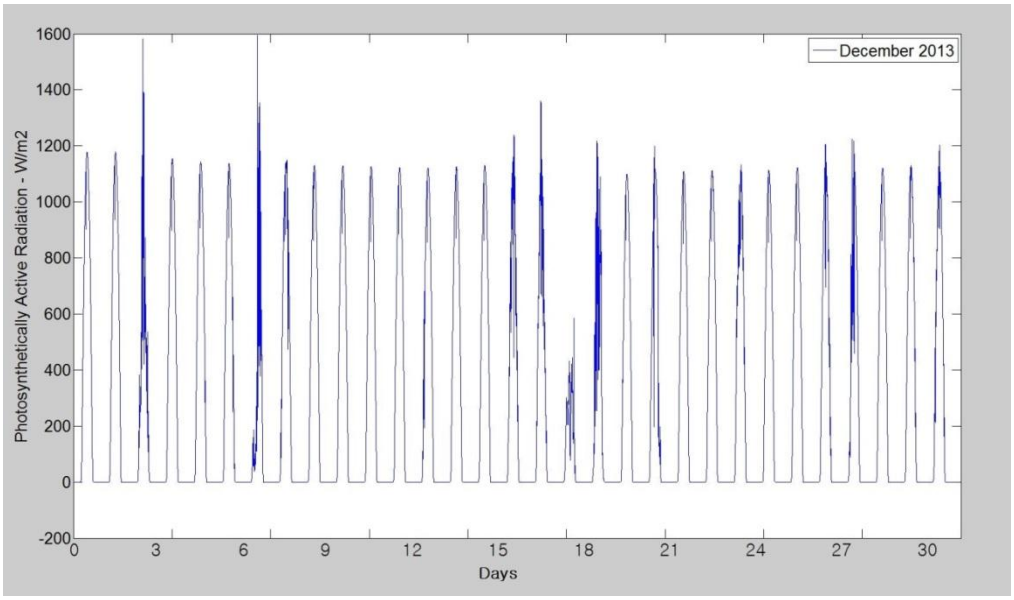

Fig 6.6: Visualization of Photosynthetically Active Radiation Data for December 2013.

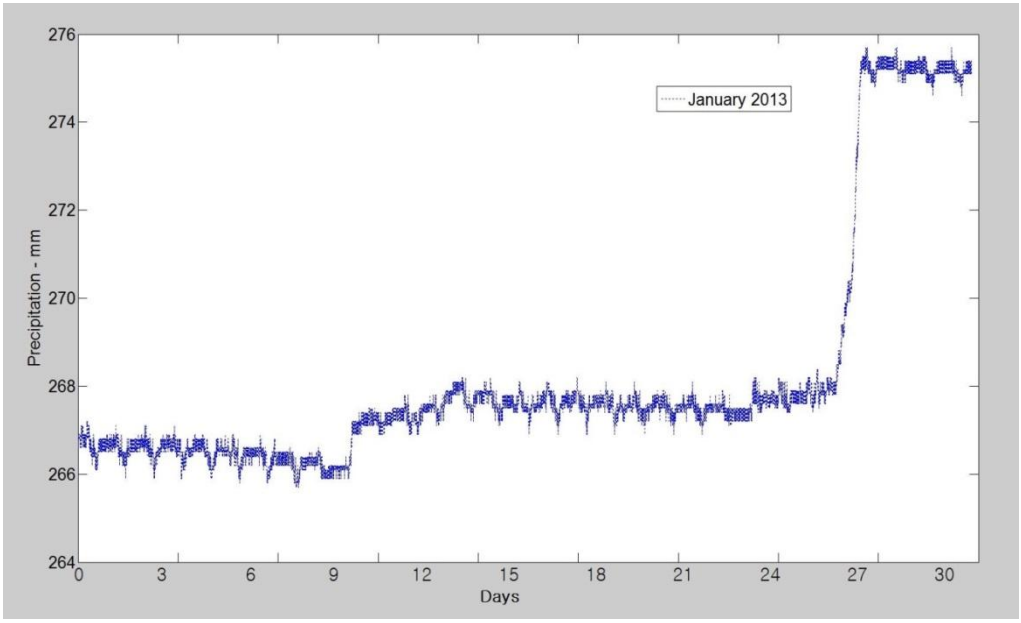

Fig 6.7: Visualization of Precipitation Data for January 2013.

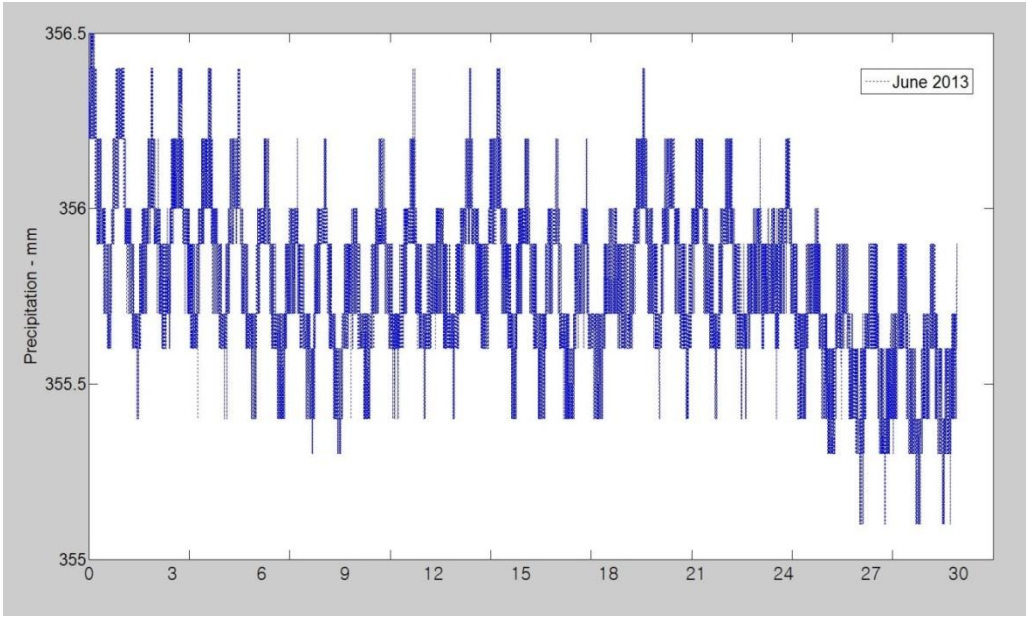

Fig 6.8: Visualization of Precipitation Data for June 2013.

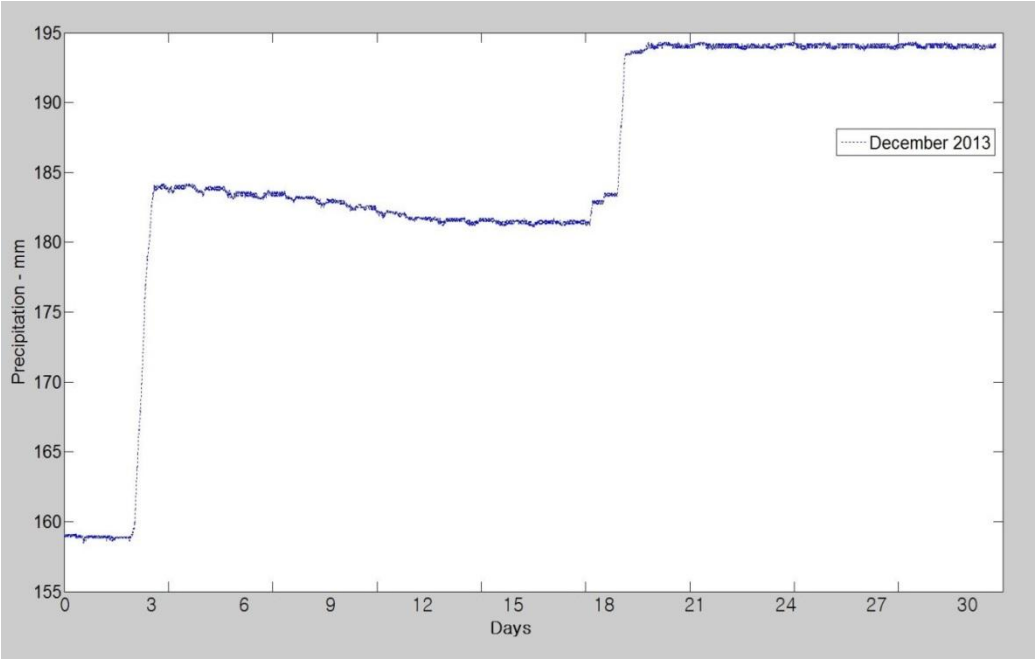

Fig 6.9: Visualization of Precipitation Data for December 2013.

The solar radiation data is downloaded for the year 2012 and the photosynthetically active radiation, and precipitation data is downloaded for the year 2013 from NCCP. The above plots give an idea of how the actual data looks like. As mentioned earlier, the solar radiation and photosynthetically active radiation data have leading zeros , non-zero data points, and trailing zeros at the end. This can be seen in Fig 6.1 to Fig 6.6. Fig 6.7 to Fig 6.9 shows visualization of precipitation data for January, June and December 2013. The precipitation data does not show much variations.

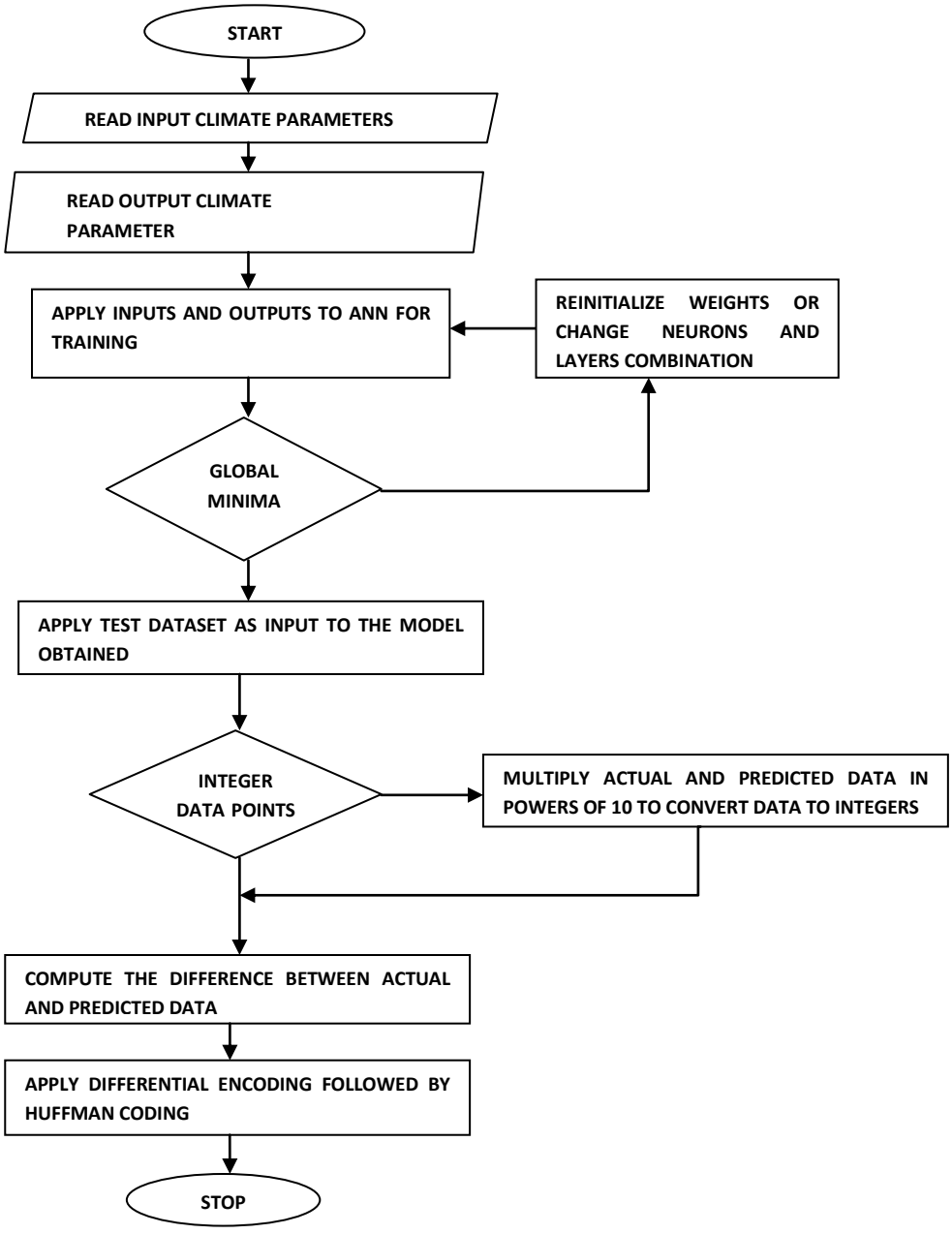

Fig 6.10: Flow chart of the proposed algorithm.

6.6 Proposed algorithm for solar radiation data

We have used a hybrid method to combine ANNs with data compression algorithms to better compress the data. The steps followed are as below:

1. The inputs (day, month, incoming longwave radiation data, incoming shortwave radiation data, outgoing longwave radiation data, outgoing shortwave radiation data, photosynthetically active radiation data, and temperature) are presented to the ANNs. After training the data, prediction of solar radiation for next year is obtained.
2. The solar radiation data consists of leading zeros, non-zero data points and trailing zeros; this enables us to store the positions of starting and ending non-zero data points.
3. The indices of non-zero data points are used to concatenate actual as well as predicted data.
4. For pre-processing, the difference between the actual and the predicted data is computed.
5. The result of this pre-processing technique is differentially encoded, and then Huffman coding is applied in the last stage.

Hence, lossless compression of solar radiation data is achieved.

6.7 Proposed algorithm for photosynthetically active radiation data

Below are the steps followed to compress photosynthetically active radiation data:

1. The inputs (day, month, incoming longwave radiation data, incoming shortwave radiation data, outgoing longwave radiation data, outgoing shortwave radiation data, humidity, and solar radiation data) are presented to the ANNs. After training the data, prediction of photosynthetically active radiation for next year is obtained.
2. The photosynthetically active radiation data consists of leading zeros, non-zero data points and then trailing zeros; this enables us to store the positions of starting and ending non-zero data points.
3. The indices of non-zero data points are used to concatenate actual as well as predicted data.
4. For pre-processing, the difference between actual data and predicted data is obtained.
5. The result of this pre-processing technique is differentially encoded, and then Huffman coding is applied in the last stage.

Hence, lossless compression of photosynthetically active radiation data is achieved.

6.8 Proposed algorithm for precipitation data

The steps followed are as below:

1. The inputs (day, month, incoming longwave radiation data, incoming shortwave radiation data, outgoing longwave radiation data, outgoing shortwave radiation

data, humidity, and solar radiation data) are presented to the ANNs. After training the data, prediction of precipitation for next year is obtained.

2. Since precipitation data contains floating point numbers both the actual and predicted data is multiplied by 10 to make them integers.
3. For pre-processing , the difference between actual and predicted data is computed.
4. The result of this pre-processing technique is differentially encoded, and then Huffman coding is applied in the last stage.

With this, lossless compression of precipitation data is achieved.

6.9 Conclusion

In this chapter, we have dealt with the different parameters considered, different sites, and properties of the parameters. We have noticed that solar radiation data and photosynthetically active radiation data has leading zeros, non-zero data points and trailing zeros and the range for precipitation data is not large. We have also explained the proposed algorithm used for solar radiation, photosynthetically active radiation and precipitation data.

CHAPTER 7

PERFORMANCE ANALYSIS OF PROPOSED ALGORITHM USING MLP AND CFNN

7.1 Introduction

In this chapter, we will describe how to implement the proposed compression algorithm for solar radiation, photosynthetically active radiation and precipitation data using MLP and CFNN. We will compare the performance metrics of both methods against each other and check which one gives best results. Also, the two methods are compared with a conventional method of differential encoding followed by Huffman Coding.

7.2 Implementation of proposed algorithm using MLP and CFNN

The solar radiation, incoming radiation, outgoing radiation, temperature, humidity, and photosynthetically active radiation data is downloaded from NCCP for the year 2012. All the inputs and outputs are minute wise data. There are total of 1440 data points per day for each parameter. As the training data is for one year, the inputs are stacked together as 2 dimensional matrix of 9 * 525600 having a total of 4730400 data points. Here solar radiation is measured in w/m^2 and temperature in deg(degrees).The incoming shortwave radiation, incoming long wave radiation, outgoing shortwave radiation, outgoing long wave radiation and photosynthetically active radiation data are measured in w/m^2. There are 9 inputs for this method namely, time, month, temperature, humidity,

incoming short wave radiation, outgoing shortwave radiation, incoming long wave radiation, outgoing long wave radiation, and photosynthetically active radiation data. The input data is for the year 2012 and the output is solar radiation data of 1 dimensional matrix having 44640 data points. The inputs and the output are presented to the ANN. In this case, we have used MLP for training the given dataset. After multiple iterations, mean square error 25 is obtained. The same process is repeated with CFNN and a performance of 27.5 is obtained using 3 layers and 27 neurons in each layer. In the next step, we download the input data for year 2013 from NCCP. This data is presented to the model trained above and the predicted values for solar radiation data for the year 2013 are obtained. Then, the actual solar radiation data for the year 2013 is downloaded from the data portal. Now, the difference between actual and predicted data is computed. When the MSE is small, then the difference between actual and predicted data will be less which is good for the compression. On this data, differential encoding is applied which helps in increasing the probability of occurrence of a symbol. Then, Huffman coding is applied which will encode the data with fewer bits thus giving lossless compression of solar radiation data.

As part of decompression, the first step is to decode the Huffman code. Now the first number obtained from decoding process is kept as the first data point for the next stage. We add this number to the second number from the decoded data which will be the second data point for the next stage and this process will continue till the last data point of the decoded data. Next, each data point from this stage is added to the predicted data.

Thus, the total process of lossless compression and decompression is completed for solar radiation data.

The photosynthetically active radiation, incoming radiation, outgoing radiation, humidity, and solar radiation data is downloaded from NCCP for the year 2013. All the inputs and outputs are minute wise data. There are total of 1440 data points per day per parameter. As the training data is for one year, the inputs are stacked together as 2 dimensional matrix of 8 * 525600 having a total of 4204800 data points. Here precipitation is measured in deg-m, temperature in deg and , incoming, outgoing radiation, and solar radiation data in w/m^2. There are 8 inputs for this method namely time, month, humidity, incoming short wave radiation, outgoing shortwave radiation, incoming long wave radiation, outgoing long wave radiation, and solar radiation data. The input data considered is for the year 2013 and the output is photosynthetically active radiation data of 1 dimensional matrix having 44640 data points. The inputs and the output are presented to the ANN. In this case, we have used MLP for training the given dataset. After multiple iterations, mean square error of 126 is obtained. The same process is repeated with CFNN and a performance of 136 is obtained using 3 layers and 24 neurons in each later. In the next step, we download the input data for year 2014 from NCCP. This data is presented to the model trained above and we arrive at the predicted values for photosynthetically active radiation data for the year 2014. Then, the photosynthetically active radiation data for the year 2014 is downloaded from the data portal. Now, the difference between actual precipitation and the predicted data is

computed. On the data obtained, differential encoding is applied which will help in increasing the probability of occurrence of a symbol. Upon this data Huffman coding is applied which will encode the data with fewer bits thus giving lossless data compression of photosynthetically active radiation data.

The procedure for decompression of photosynthetically active radiation data is same as solar radiation data.

The precipitation, temperature, humidity, incoming radiation, outgoing radiation, and solar radiation data are downloaded from NCCP for the year 2013. All the inputs and outputs are minute wise time series data. The training data fed to the ANN is for the year 2013.The inputs are stacked together as 2 dimensional matrix of 8 * 525600 having a total of 4204800 data points. Here, precipitation is measured in mm and temperature in deg. The incoming radiation, outgoing radiation, and solar radiation are measured in w/m^2. There are 8 inputs for this method namely time, month, temperature, humidity, incoming short wave radiation, outgoing shortwave radiation, incoming long wave radiation, outgoing long wave radiation, and solar radiation data. The input data considered is for the year 2013 and the output is a 1 dimensional matrix having 44640 data points. The inputs and the output are presented to the ANN. In this case, we have used MLP for training the given dataset. After multiple iterations, MSE of 10.1 is obtained. To obtain this performance, we have used ANN with 2 layers and 20 neurons present in each layer. In the next step, we download the input data for year 2014 from

NCCP. This data is presented to the model trained above and then predicted values for precipitation data for the year 2014 are obtained. Then, the precipitation data for the year 2014 is downloaded from NCCP. Since the precipitation data contains floating point numbers, all the data points are multiplied by 10 to make both inputs and output as integers. In this way, it will be beneficial for pre-processing the data. Now, the difference between actual precipitation and predicted data is computed. When MSE is small, then the difference will be small which is great for the compression. On the data obtained from the previous stage, differential encoding is applied which will help in increasing the probability of occurrence of a symbol. Upon this data, Huffman coding is applied which will encode the data with fewer bits thus giving lossless data compression of precipitation data.

The procedure for decompression of precipitation data is similar to solar radiation and photosynthetically active radiation data. The only difference is that, the data points obtained in the final stage have to be divided by 10 to get back the actual data. In this way, lossless compression of precipitation data is achieved.

7.3 Results

The different performance metrics used for the research are MSE, RMSE, compression ratio, and saving percentage. Table 7.1 and 7.2 shows MSE and RMSE for MLP and CFNN for all the parameters considered. The results show that performance of

MLP is better than CFNN. This means that, MLP gives better prediction of climate data when compared to CFNN.

MSE

Parameter	MLP	CFNN
Solar Radiation	25	27.5
Photosynthetically Active Radiation	126	136
Precipitation	10.1	52

Table 7.1: Mean Square Error for all the parameters.

RMSE

Parameter	MLP	CFNN
Solar Radiation	5	5.25
Photosynthetically Active Radiation	11.22	11.66
Precipitation	3.17	7.21

Table 7.2: Root Mean Square Error for all the parameters.

Table 7.3 to 7.8 shows the compression ratios and saving percentage of all the parameters using MLP and CFNN. The results show that MLP outperforms CFNN and gives better compression ratios.

Month	Uncompressed(bits)	Compressed(bits)	Compression ratio
Jan	758880	85608	8.86
Feb	758880	79467	9.54
Mar	758880	91529	8.29
Apr	734400	87308	8.41
May	758880	90276	8.4
June	758880	90090	8.42
July	803520	95594	8.4
August	803520	99824	8.04
Sep	777600	103394	7.52
Oct	803520	89579	8.96
Nov	777600	97382	7.98

Table 7.3: Compression Ratio for Solar Radiation Data for year 2013 using MLP.

Month	Uncompressed(bits)	Compressed(bits)	Compression ratio
Jan	758880	150530	5.04
Feb	758880	160760	4.72
Mar	758880	148641	5.10
Apr	734400	131046	5.60
May	758880	134080	5.65
Jun	758880	142100	5.34
Jul	803520	173450	4.63
Aug	803520	140181	5.73
Sep	777600	164532	4.72
Oct	803520	132236	6.07
Nov	777600	123748	6.28
Dec	803520	127610	6.29

Table 7.4: Compression Ratio for Solar Radiation Data for year 2013 using CFNN.

Month	Uncompressed(bits)	Compressed(bits)	Compression ratio
Jan	446400	48521	9.2
Feb	432000	49090	8.8
Mar	446400	57011	7.83
Apr	432000	63436	6.81
May	446400	67534	6.61
June	432000	68680	6.29
July	446400	79289	5.63
Aug	446400	68888	6.48
Sep	432000	54545	7.92
Oct	446400	46940	9.51
Nov	432000	48484	8.91
Dec	446400	45551	9.8

Table 7.5: Compression Ratio for Photosynthetically Data for year 2014 using MLP.

Month	Uncompressed(bits)	Compressed(bits)	Compression ratio
Jan	446400	49544	9.01
Feb	432000	50467	8.56
Mar	446400	58200	7.67
Apr	432000	66461	6.5
May	446400	70521	6.33
June	432000	68899	6.27
July	446400	82209	5.43
Aug	446400	71653	6.23
Sep	432000	57600	7.5
Oct	446400	51606	8.65
Nov	432000	51798	8.34
Dec	446400	46211	9.66

Table 7.6: Compression Ratio for Photosynthetically Data for year 2014 using CFNN.

Month	Uncompressed(bits)	Compressed(bits)	Compression ratio
Jan	491040	65791	7.46
Feb	561600	79888	7.02
Mar	580320	86982	6.67
Apr	561600	99644	5.63
May	580320	111166	5.22
June	561600	104840	5.35
July	580320	146563	3.95
Aug	580320	123429	4.70
Sep	561600	138351	4.05
Oct	580320	125848	4.61
Nov	561600	61660	9.10
Dec	580320	76009	7.63

Table 7.7: Compression Ratio for Precipitation Data for year 2014 using CFNN.

Month	Uncompressed(bits)	Compressed(bits)	Compression ratio
Jan	491040	58152	8.44
Feb	561600	78321	7.17
Mar	580320	80843	7.18
Apr	561600	96198	5.83
May	580320	113440	5.11
June	561600	114428	4.90
July	580320	150219	3.86
Aug	580320	122289	4.74
Sep	561600	132544	4.23
Oct	580320	121019	4.79
Nov	561600	54495	10.30
Dec	446400	72524	8.00

Table 7.8: Compression Ratio for Precipitation Data for year 2014 using MLP.

We have also performed differential encoding on the solar radiation, photosynthetically active radiation, and precipitation datasets followed by Huffman coding. Then, we have compared the results of the proposed compression algorithm with the standard method of differential encoding followed by Huffman coding. The results are given in tables 7.9,7.10, and 7.11. The results show that the proposed compression algorithms performed better than the standard method.

Month	Uncompressed(bits)	Compressed(bits for Huffman coding)	Compression ratio using MLP	Compression ratio using CFNN	Compression ratio using Huffman Coding
Jan	758880	169771	8.86	5.04	4.47
Feb	758880	204000	9.54	4.72	3.72
Mar	758880	237150	8.29	5.10	3.20
Apr	734400	171588	8.41	5.60	4.28
May	758880	166786	8.40	5.65	4.55
Jun	758880	216205	8.42	5.34	3.51
Jul	803520	270545	8.40	4.63	2.97
Aug	803520	207092	8.04	5.73	3.88
Sep	777600	243761	7.52	4.72	3.19
Oct	803520	203939	8.96	6.07	3.94
Nov	777600	188737	7.98	6.28	4.12
Dec	803520	142721	7.81	6.29	5.63

Table 7.9: Comparison of CR for Solar radiation data between MLP,CFNN and Huffman Coding.

Month	Uncompressed(bits)	Compressed(bits for Huffman coding)	Compression ratio using MLP	Compression ratio using CFNN	Compression ratio using Differential encoding and Huffman Coding
Jan	446400	89280	9.2	9.01	5.00
Feb	432000	119008	8.8	8.56	3.63
Mar	446400	135683	7.83	7.67	3.29
Apr	432000	141176	6.81	6.5	3.06
May	446400	142619	6.61	6.33	3.13
Jun	432000	144966	6.29	6.27	2.98
Jul	446400	192413	5.63	5.43	2.32
Aug	446400	125042	6.48	6.23	3.57
Sep	432000	111627	7.92	7.5	3.87
Oct	446400	102857	9.51	8.65	4.34
Nov	432000	90376	8.91	8.34	4.78
Dec	446400	89280	9.8	9.66	5.65

Table 7.10: Comparison of CR for photosynthetically active radiation data between MLP, CFNN and Huffman Coding.

Month	Uncompressed(bits)	Compressed(bits for Huffman coding)	Compression ratio using MLP	Compression ratio using CFNN	Compression ratio using Differential encoding and Huffman Coding
Jan	491040	99000	8.44	7.46	4.96
Feb	561600	94386	7.17	7.029	5.95
Mar	580320	124800	7.18	6.67	4.65
Apr	561600	133396	5.83	5.63	4.21
May	580320	122689	5.11	5.22	4.73
Jun	561600	157752	4.9	5.35	3.56
Jul	580320	205060	3.86	3.95	2.83
Aug	580320	231203	4.74	4.7	2.51
Sep	561600	191020	4.23	4.05	2.94
Oct	580320	166280	4.8	4.61	3.49
Nov	561600	178285	10.3	9.1	3.15
Dec	580320	99000	8	7.63	3.23

Table 7.11: Comparison of CR for precipitation data between MLP,CFNN and Huffman Coding.

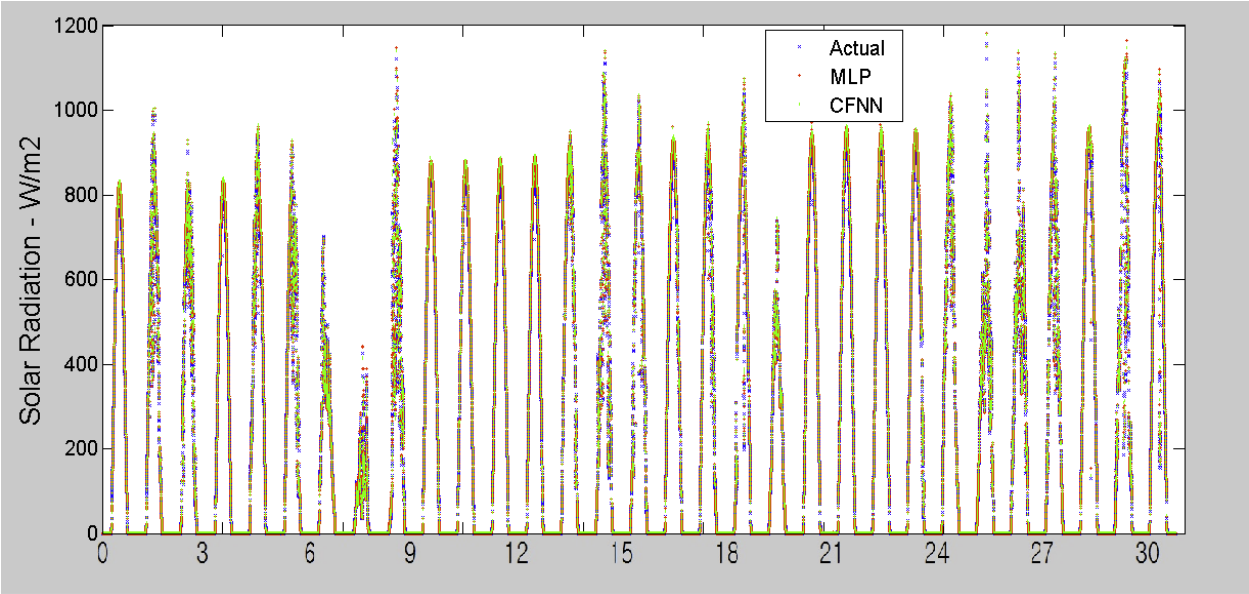

Fig 7.1: Actual and Predicted Values for Solar Radiation Data for January 2014.

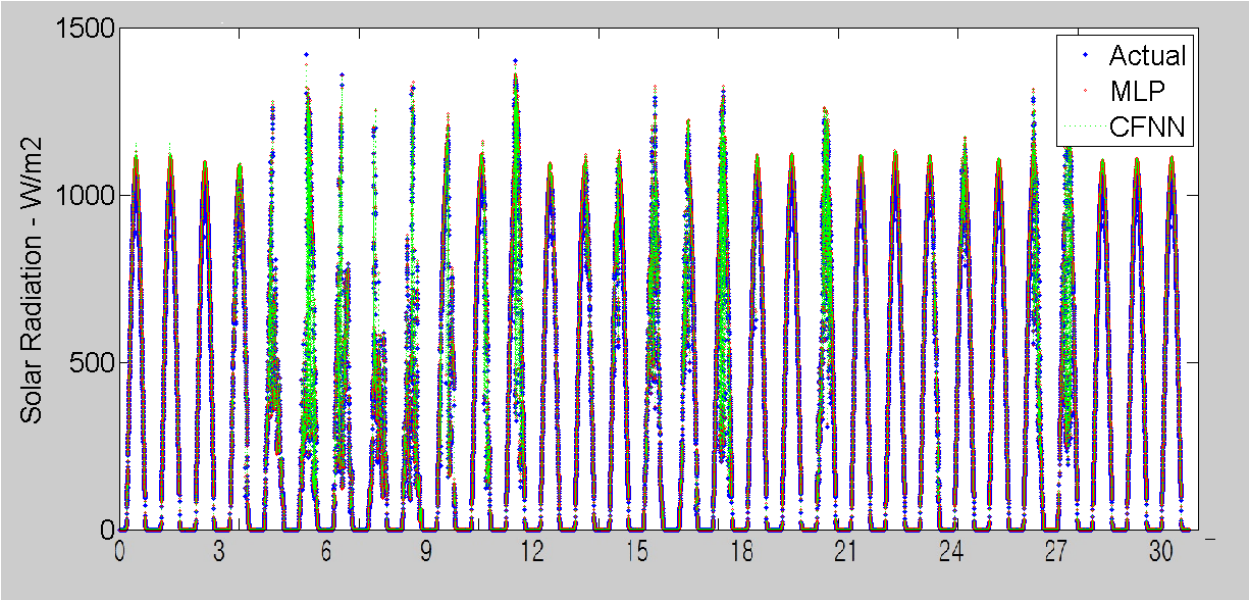

Fig 7.2: Actual and Predicted Values for Solar Radiation Data for June 2014.

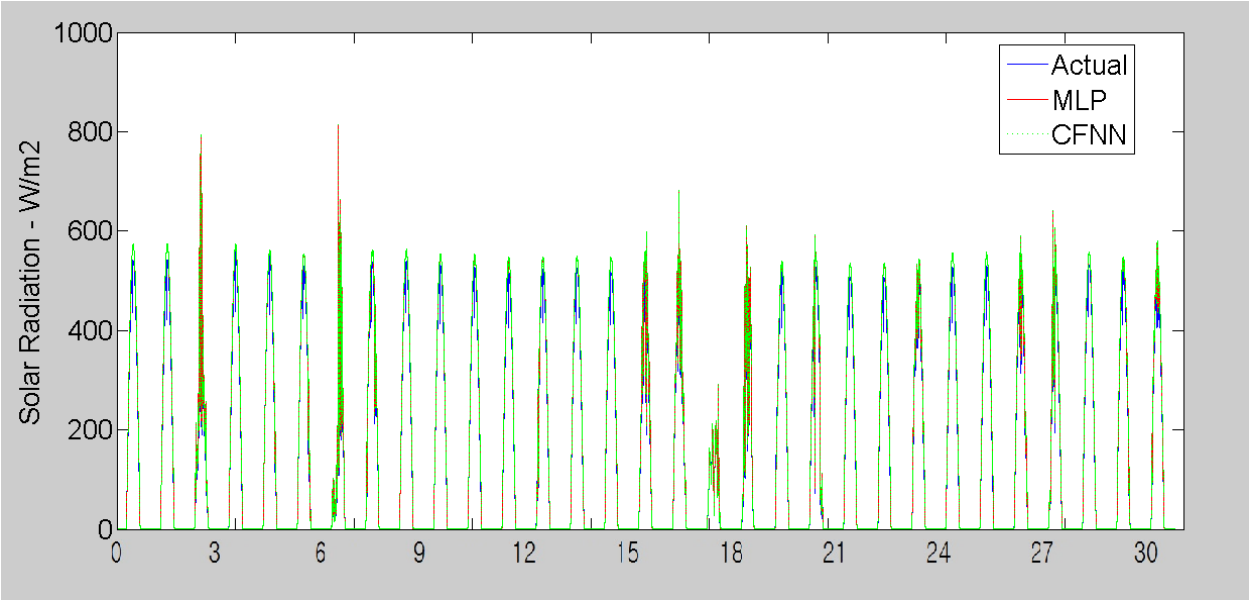

Fig 7.3: Actual and Predicted Values for Solar Radiation Data for December 2014.

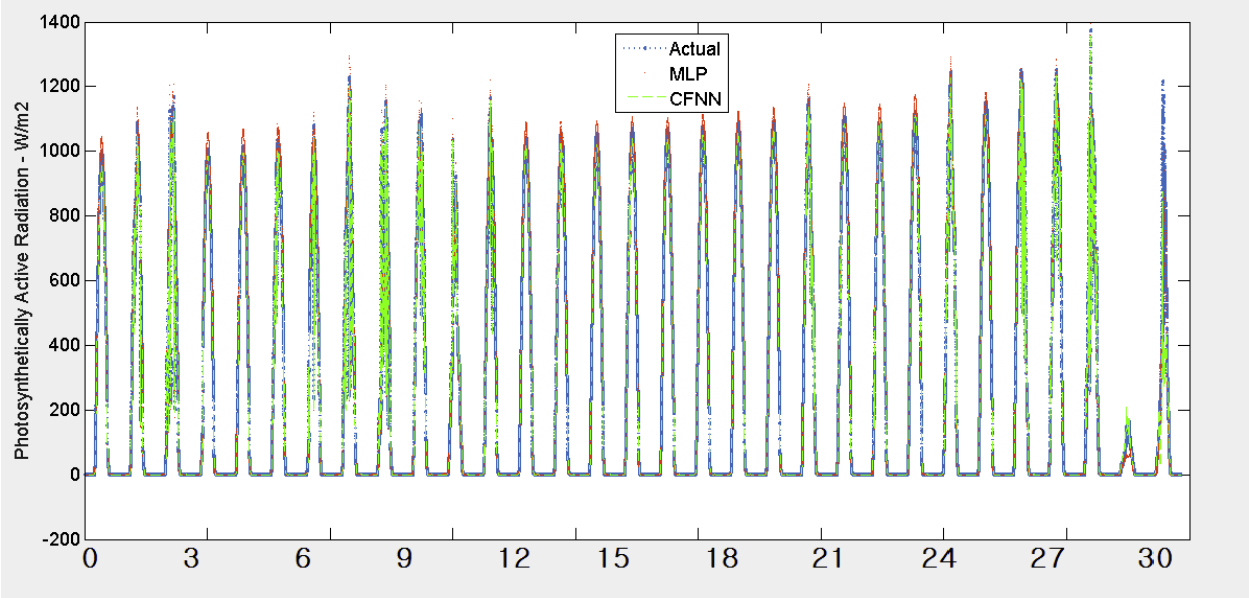

Fig 7.4: Actual and Predicted Values for Photosynthetically Active Radiation Data for January 2014.

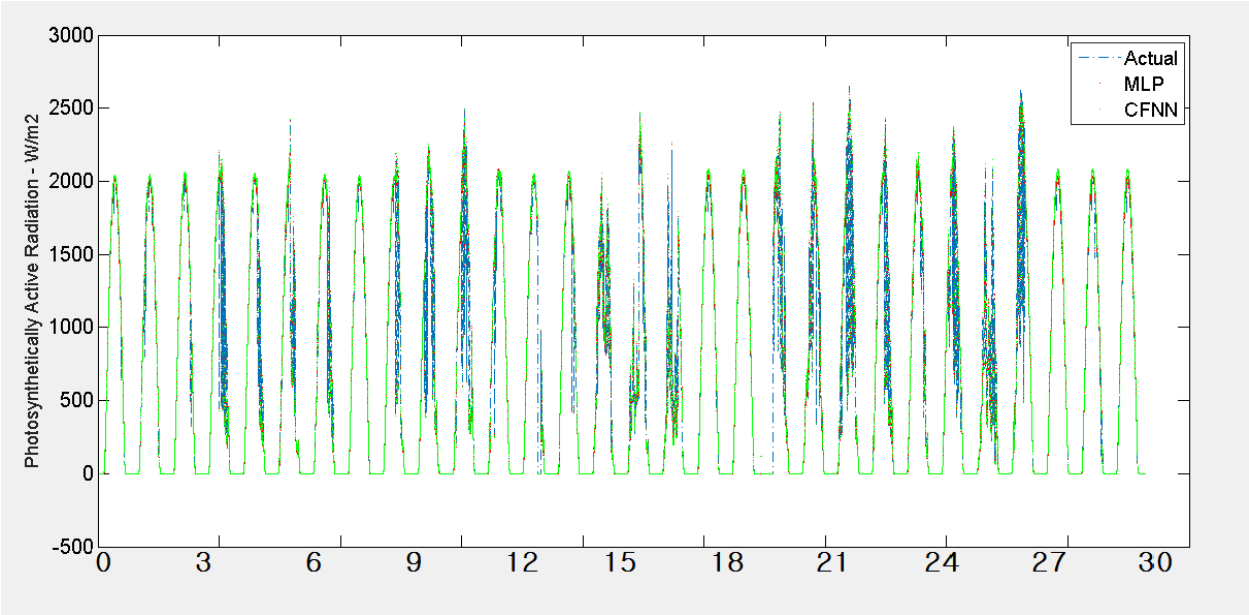

Fig 7.5: Actual and Predicted Values for Photosynthetically Active Radiation Data for June 2014.

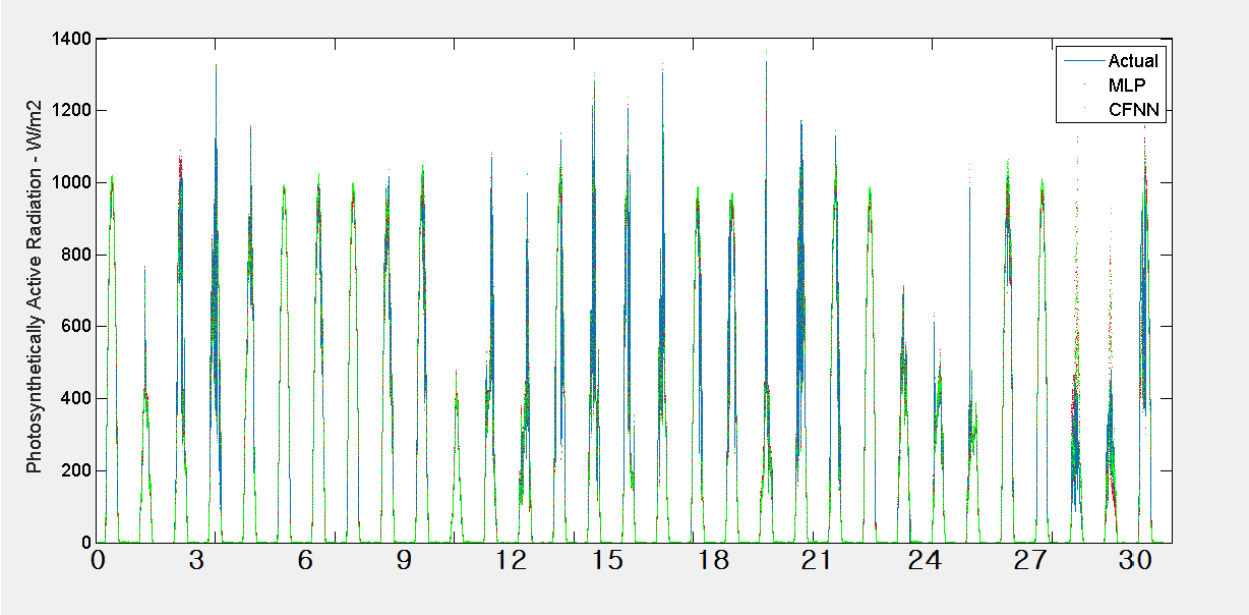

Fig 7.6: Actual and Predicted Values for Photosynthetically Active Radiation Data for December 2014.

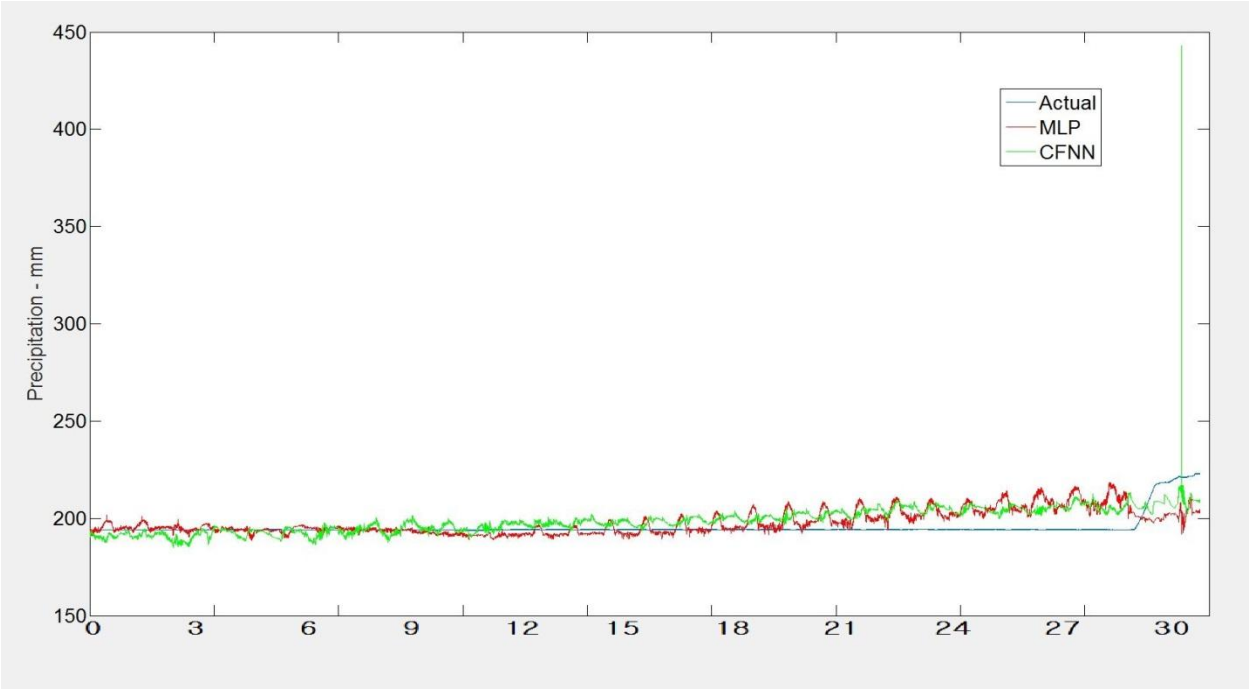

Fig 7.7: Actual and Predicted Values for Precipitation Data for January 2014.

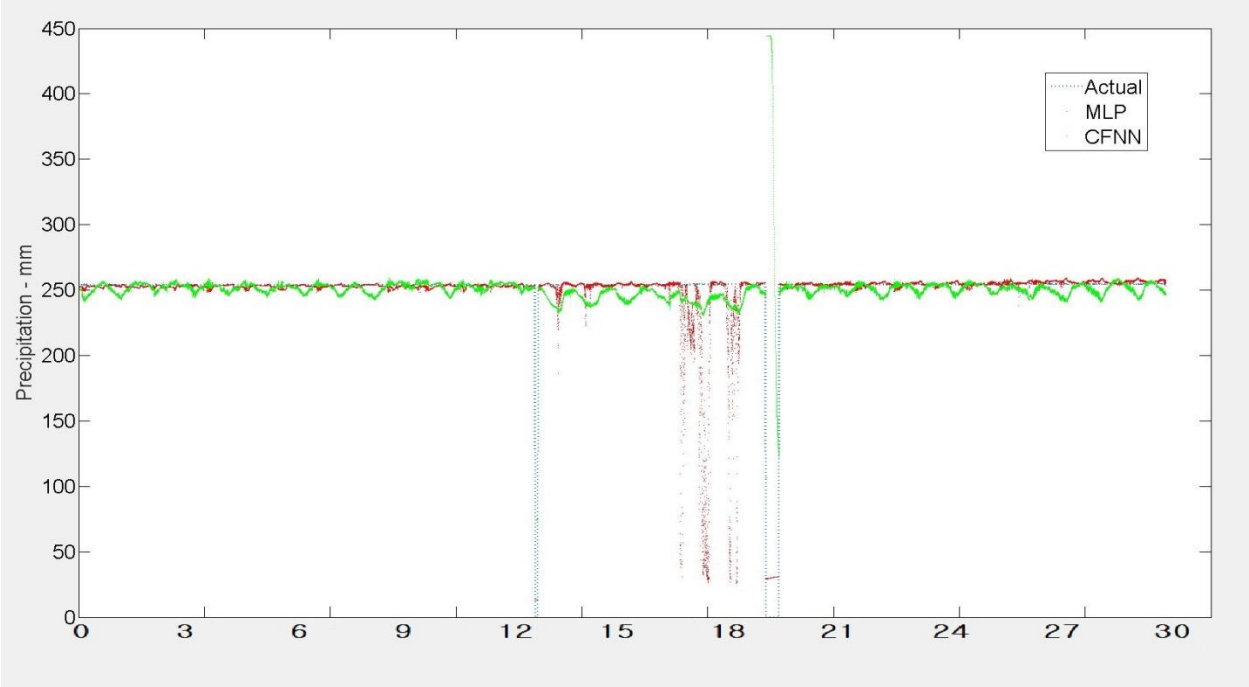

Fig 7.8: Actual and Predicted Values for Precipitation Data for June 2014.

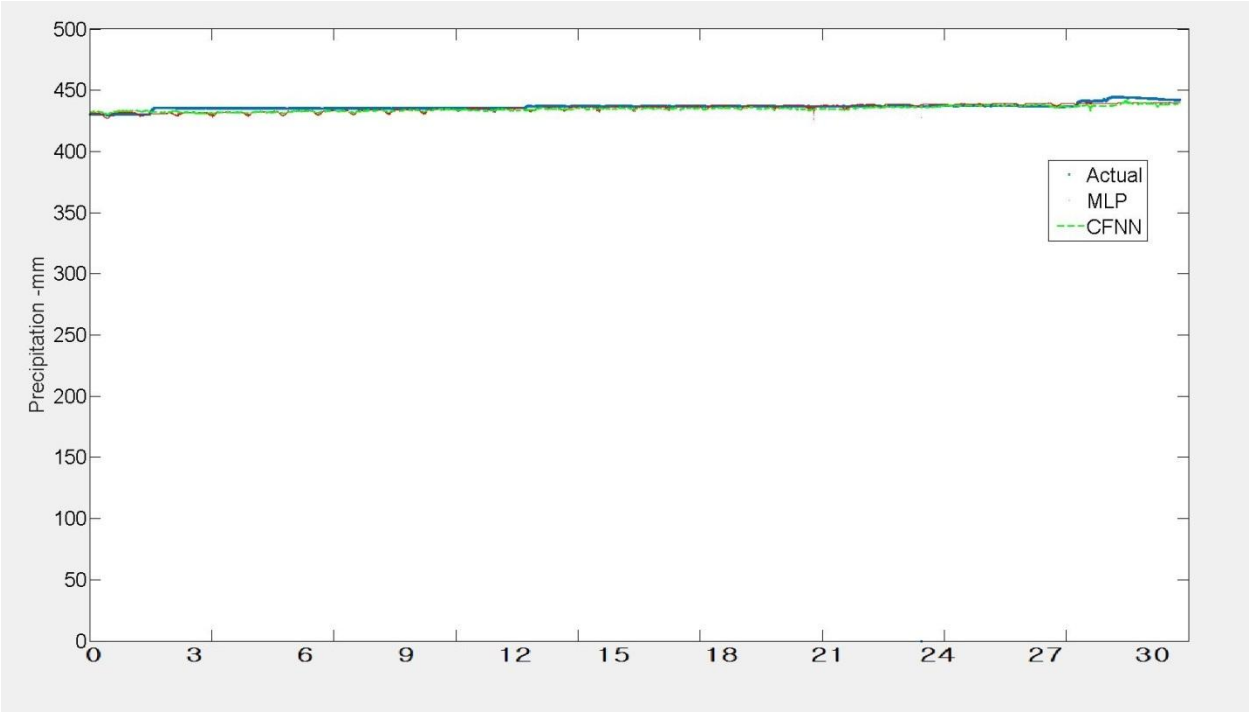

Fig 7.9: Actual and Predicted Values for Precipitation Data for December 2014.

Fig 7.1 to 7.9 shows the plots of actual and predicted (using MLP and CFNN) solar radiation, photosynthetically active radiation and precipitation data. From the plots, it can be observed that the actual and predicted data are very close to each other which is good for compression of data. This is possible because ANNs are great at prediction of climate data.

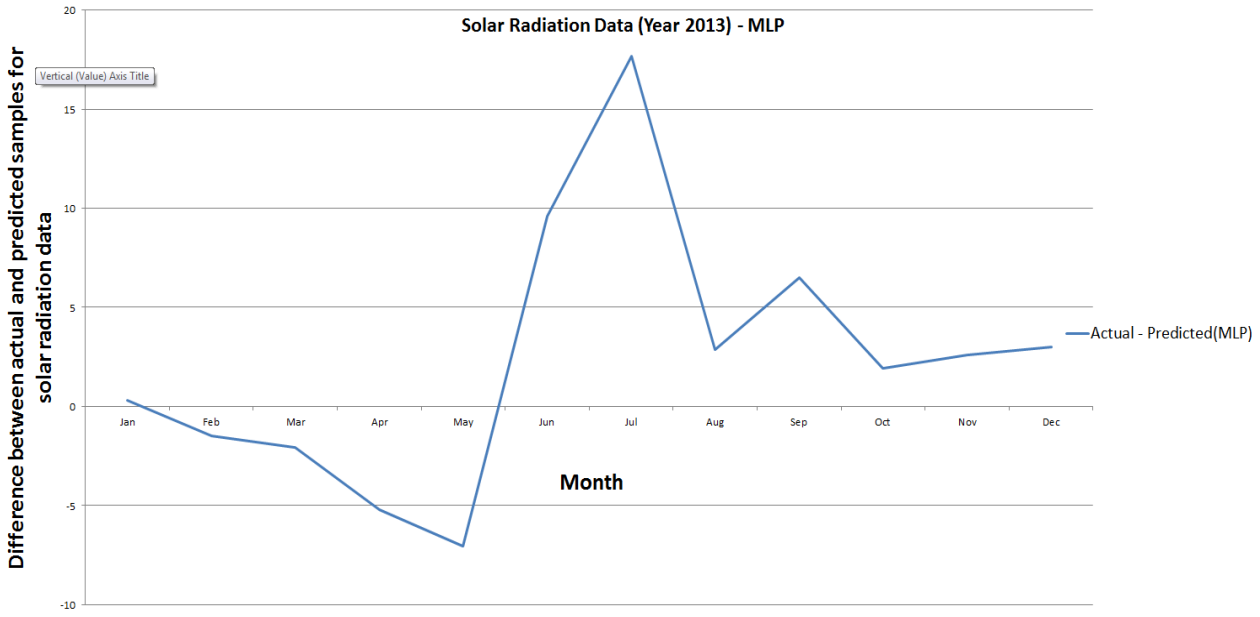

Fig 7.10: Difference between actual and predicted solar radiation data for year 2013 using MLP.

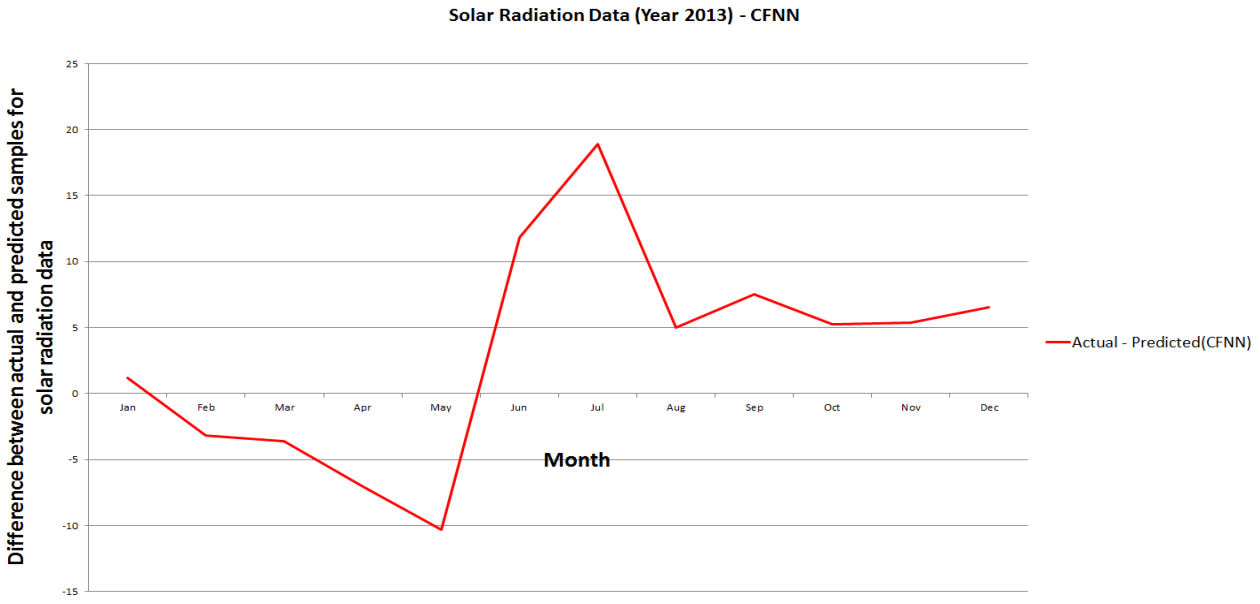

Fig 7.11: Difference between actual and predicted solar radiation data for year 2013 using CFNN.

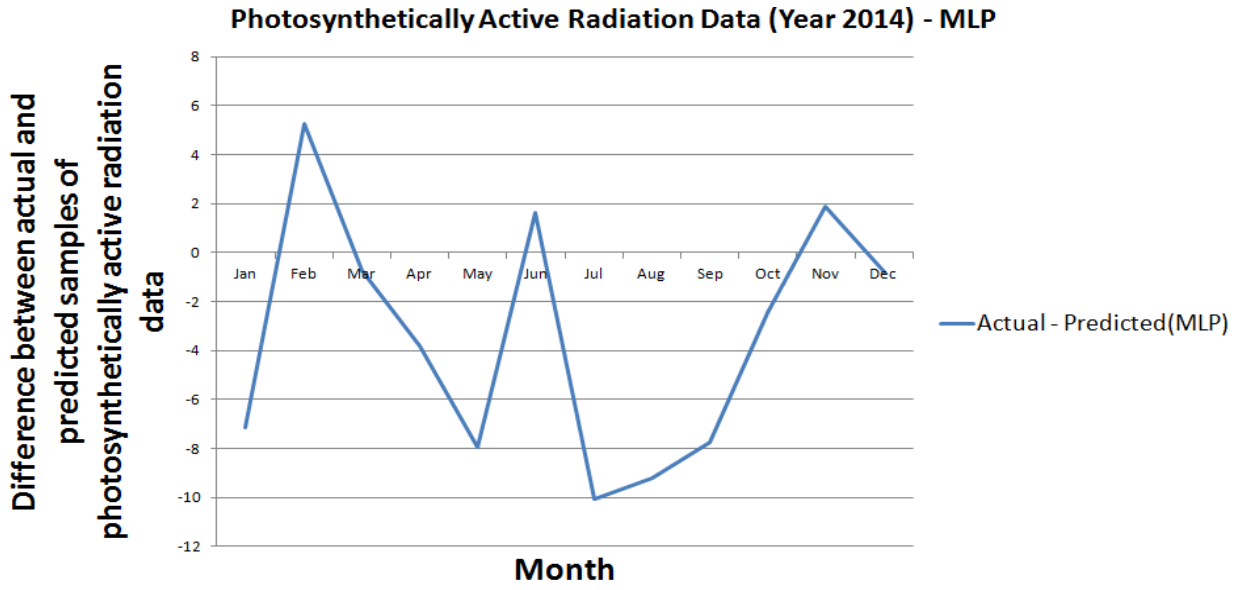

Fig 7.12: Difference between actual and predicted photosynthetically active radiation data for year 2013 using MLP.

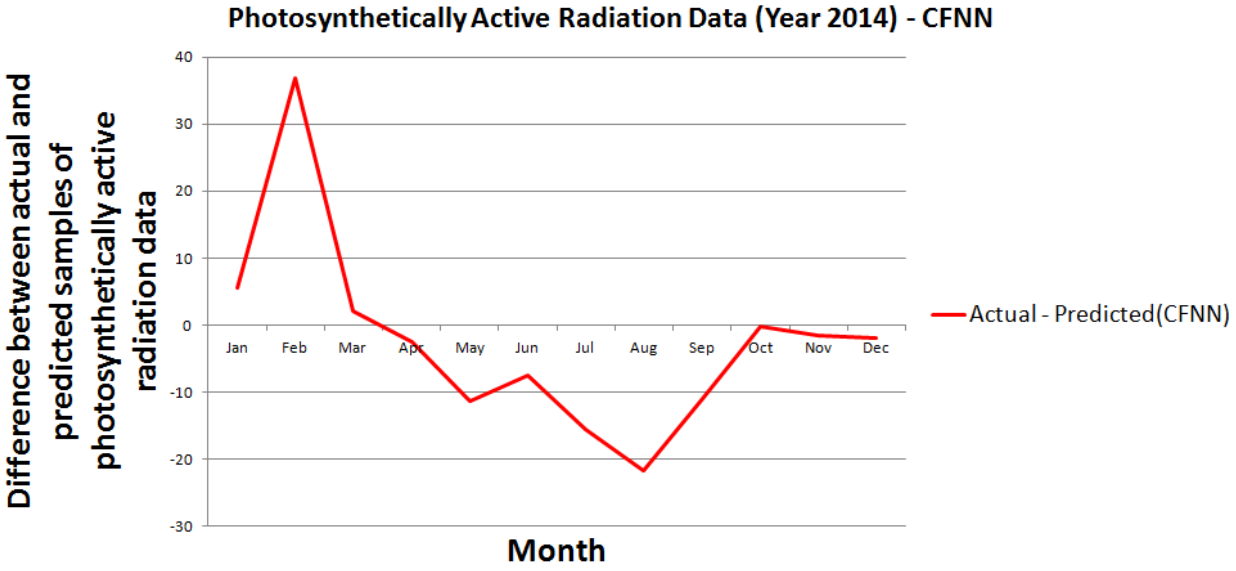

Fig 7.13: Difference between actual and predicted photosynthetically active radiation data for year 2013 using CFNN.

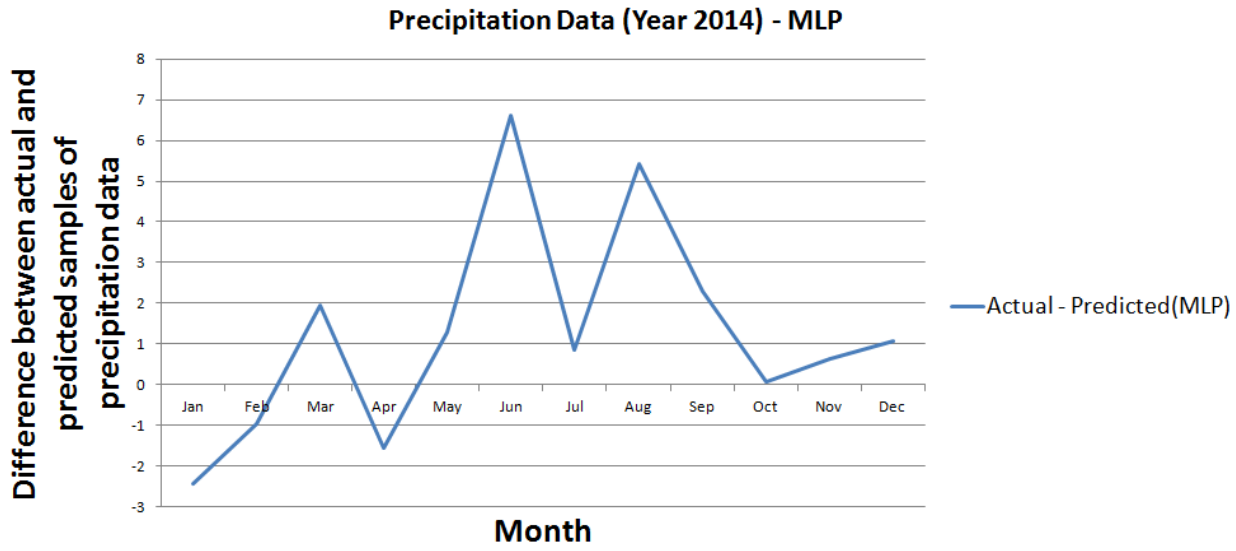

Fig 7.14: Difference between actual and predicted precipitation data for year 2013 using MLP.

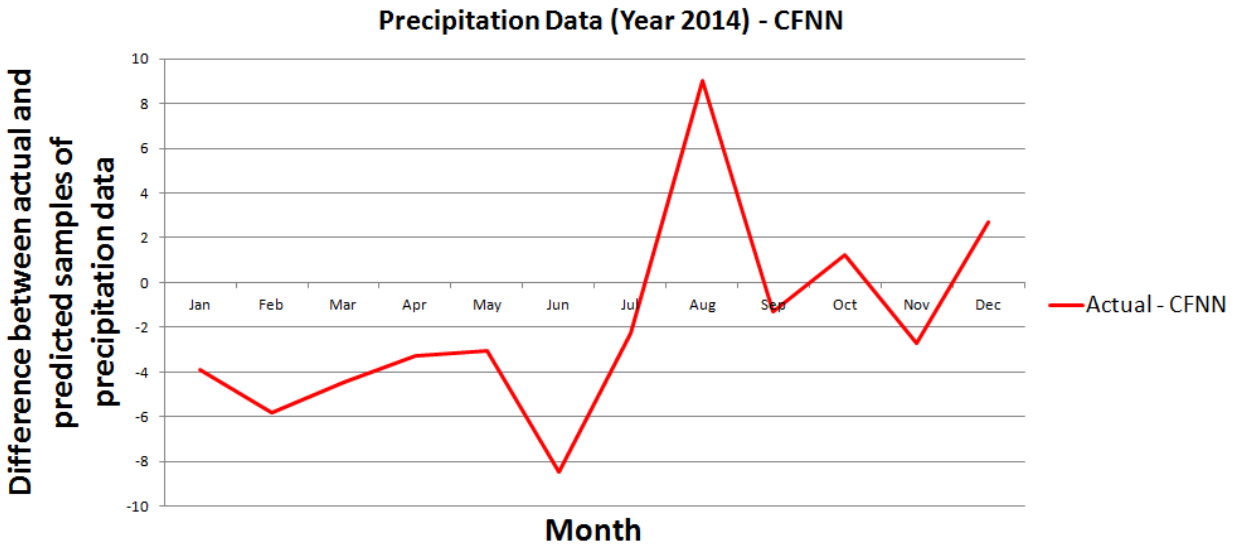

Fig 7.15: Difference between actual and predicted precipitation data for year 2013 using CFNN.

Figure 7.10 to 7.15 shows the difference between actual and predicted samples for solar radiation data, photosynthetically active radiation data and precipitation data using MLP and CFNN. The range of solar radiation data using MLP after computing the difference between actual and predicted data is -7 to 17 whereas the range of solar radiation data using CFNN is -10 to 19. The range of photosynthetically active radiation data is -10 to 5 and -21 to 38 using MLP and CFNN respectively. The range of precipitation data is -2 to 7 and -9 to 9 using MLP and CFNN respectively. After applying differential encoding , the range of the above datasets is further reduced which is suitable data compression.

The proposed method using MLP and CFNN can be applied to numeric dataset in any application. It suits best for integer and floating point data upto 2 digits after the decimal point. This method can be applied to stock market data, dataset related to flow discharge in rivers, traction performance parameters, freeway traffic data, environment, air pollution and ozone levels data, road accidents data , health care related data etc. The above method is not suitable for non-numeric data. Using this method, high compression ratios are achieved for climatology datasets due to which lesser number bits can be transmitted and stored.

7.4 Conclusion and Future work

From the results it can be concluded that, MLP is the best model for prediction of solar radiation, photosynthetically active radiation and precipitation datasets, as it proved to give best performance in all the scenarios. This in turn resulted to give better

compression ratios as compared to CFNN. We have obtained compression ratios as high as 9.8 for solar radiation , 10.3 for precipitation and 9.9 for photosynthetically active ration datasets. Also, compression ratios of the proposed methods are higher than the compression ratios obtained by the standard method of differential encoding followed by Huffman coding. Thus, this study shows that using the proposed method, lossless compression of climate data is achieved with high compression ratios which is useful for increasing storage capacity.

In future, the research can be continued by using other predication mechanisms like support vector machines and linear regression. These results can be compared with those obtained from ANN.

PUBLICATIONS

[1] Mummadisetty, B.C., Puri, A., Sharifahmadian, E. and Latifi, S. (2014) Lossless Compression of Climate Data. Proceedings of the 23rd International Conference on Systems Engineering, Vol. 330, 391-400.

[2] Mummadisetty, B. , Puri, A. , Sharifahmadian, E. and Latifi, S. (2015) A Hybrid Method for Compression of Solar Radiation Data Using Neural Networks. International Journal of Communications, Network and System Sciences,8, 217-228.

A. MATLAB Code

A.1 Compression Algorithm for Solar Radiation and Photosynthetically Data

```
i=1;
j=1;
k=1;
flag=0;
index_vec(1,1) = 1;
k1=0;
k2=0;
m=1;
while(i <= numel(dec_2013_a))
 if (flag == 0)
   if (dec_2013_a(i,1) == 0)
     ;
   else
     index_vec_zeros(j,1) = i-1;
     index_vec_data(k,1) = i;
     k=k+1;
     j=j+1;
     flag = 1;
   end
 else
    if(dec_2013_a(i,1) ~= 0)
     ;
```

```
    else
      index_vec_zeros(j,1) = i;
      index_vec_data(k,1) = i-1;
      k=k+1;
      j=j+1;
      flag=0;
    end
  end
  i=i+1;
end
data_len=numel(index_vec_data);
i=1;
in=1;
while(i <= data_len)
  k1=index_vec_data(i,1);
  k2=index_vec_data(i+1,1);
  k3=k2-k1+in;
  diff_vec_p(in:k3,1) = dec_2013_p(k1:k2,1);
  diff_vec_a(in:k3,1) = dec_2013_a(k1:k2,1);
  in=k3+1;
  i=i+2;
end
i=1;
while(i<=(numel(diff_vec_p)))
   data_vec(i,1) = diff_vec_p(i,1) - diff_vec_a(i,1);
   i = i + 1;
end
i=1;
```

```
temp_out(i,1) = data_vec(1,1);
i=i+1;
while(i<=(numel(data_vec)))
   temp_out(i,1) = data_vec(i,1) - data_vec(i-1,1);
   i = i + 1;
end
[~,indx] = unique(temp_out);
symbol = temp_out(sort(indx));
[~,indx1]=unique(symbol);
temp_freq=histc(temp_out,unique(symbol));
k=1;
freq=zeros(numel(temp_freq),1);
while(k <= numel(temp_freq))
   freq(indx1(k,1),1) = temp_freq(k,1);
   k = k + 1;
end
prob = freq/numel(temp_out);
dict = huffmandict(symbol,prob);
hcode = huffmanenco(temp_out,dict);
xlswrite('huffman_jun_2013_bp.xlsx',hcode,'B1');
i=1;
in=1;
k1=1;
k2=1;
k3=1;
l=1;
while(i <= numel(index_vec_zeros))
  k1=index_vec_zeros(i,1);
```

```
    k2=index_vec_zeros(i+1,1);
    k3=k2-k1+1;
    temp_out_zeros(l,1)=k3;
    l=l+1;
    i=i+2;
end
[~,indx3] = unique(temp_out_zeros);
symbol2 = temp_out_zeros(sort(indx3));
[~,indx4]=unique(symbol2);
temp_freq_zeros=histc(temp_out_zeros,unique(symbol2));
k=1;
freq2=zeros(numel(temp_freq_zeros),1);
while(k <= numel(temp_freq_zeros))
    freq2(indx4(k,1),1) = temp_freq_zeros(k,1);
    k = k + 1;
end
prob2 = freq2/numel(temp_out_zeros);
dict2 = huffmandict(symbol2,prob2);
hcode2 = huffmanenco(temp_out_zeros,dict2);
xlswrite('other_zeros_jun_bp.xlsx',hcode2,'B1');
```

A.2Compression Algorithm for Precipitation Data

```
[~,indx] = unique(temp_out);
symbol = temp_out(sort(indx));
[~,indx1]=unique(symbol);
```

```
temp_freq=histc(temp_out,unique(symbol));
k=1;
freq=zeros(numel(temp_freq),1);
while(k <= numel(temp_freq))
   freq(indx1(k,1),1) = temp_freq(k,1);
    k = k + 1;
end
prob = freq/numel(temp_out);
dict = huffmandict(symbol,prob);
hcode = huffmanenco(temp_out,dict);
xlswrite('huffman_en_dec.xlsx',hcode,'B1');
```

REFERENCES

[1] Sayood, K.(2006). *Introduction to Data Compression*. Third Edition, San Francisco, CA: Morgann Kaufmann Publishers.

[2] Saupe, D., Hartenstein, H., & Wergen, W.(2010). *Compression of weather forecast data*. Institutional information processing systems conference.

[3] Steffen, C. E., & Wang, N.(2003). *Weather data compression*. 19th International Conference on Interactive Information Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology, Amer. Meteor. Soc., CD-ROM, 4.9, 2003.

[4] Karim, S., Karim, B., Tahir, M., Ismail, M., Hasan, M., & Sulaiman, J.(2010). *Compression of temperature data by using daubechies wavelets*. International Conference on Mathematical Sciences.

[5] Engelson, V., Fritzson, D., & Fritzson, P.(2000, March). *Lossless Compression of High-volume Numerical*. Data compression conference, pp 574-586, 28- 30 Mar,2000.

[6] Xie, X., & Qin, Q.(2009, May). *Fast Lossless Compression of seismic Floating Point Data.* Information Technology and Applications, pp 235-238,15-17.

[7] Steinbach, M., Kumar, V., & Pang, N.T.(2006). *Introduction to data mining.* Addison Wesley.

[8] Olaiya, F.(2012, February). *Application of Data Mining Techniques in Weather Prediction and Climate Change Studies.* I.J. Information Engineering and Electronic Business, pp 51-59.

[9] Afzali, M., Afzali, A., & Zahedi, G.(2011). *Ambient Air Temperature Forecasting Using Artificial Neural Network Approach*. ICEC Conference, vol.19, pp 176-180.

[10] Yadav, A.K., & Chandel.(2014). *Solar Radiation Prediction Using Artificial Neural Network Techniques: A Re-view. Renewable and Sustainable Energy Reviews*, 33, pp772-781.

[11] Al-Alawi, S.M., & Al-Hinai.(1998). *An ANN-Based Approach for Predicting Global Radiation in Locations with No Direct Measurement Instrumentation.* Renewable Energy, 14, pp 199-204.

[12] Sözen, A., Arcaklioğlu, E., Özalp, M., & Kanit E.G.(2004). *Use of Artificial Neural Networks for Mapping of Solar Potential in Turkey*. Applied Energy, 77, pp 273-286.

[13] Sözen, A., Arcaklioğlu, E., & Özalp.(2004). *Estimation of Solar Potential in Turkey by Artificial Neural Networks Using Meteorological and Geographical Data*. Energy Conversion and Management, 45, pp 3033-3052.

[14] AbdAlKader, S.A., & AL-Allaf.(2011, May). *Backpropagation Neural Network Algorithm for Forecasting Soil Temperatures Considering Many Aspects: A Comparison*

of Different Approaches. The 5th International Conference on Information Technology, Amman, pp 11-13,13.

[15] Khatib, T., Mohamed, A., Sopian, K., and Mahmoud, M.(2012). *Assessment of Artificial Neural Networks for Hourly Solar Radiation Prediction.* International Journal of Photoenergy, Article ID: 946890.

[16] Mummadisetty, B.C., Puri, A., Sharifahmadian, E., & Latifi, S.(2014). *Lossless Compression of Climate Data*. Proceedings of the 23rd International Conference on Systems Engineering, Vol. 330, 391-400.

[17] Mummadisetty, B., Puri, A., Sharifahmadian, E., and Latifi, S.(2015). *A Hybrid Method for Compression of Solar Radiation Data Using Neural Networks*, International Journal of Communications, Network and System Sciences,8, 217-228.

[18] Dascalu, S., Frederick, C., Harris, Jr., McMahon, M., Fritzinger, E., Strachan, S., & Kelley, R.(2014). *An Overview of the Nevada Climate Change Portal*. 7th Intl. Congress International Environmental Modelling and Software Society (iEMSs) , Vol 1, pp 75-82, June 15-19, San Diego.

[19] http://sensor.nevada.edu/NCCP/Default.aspx

[20] Jani, H., & Trivedi, J.(2014). *A Survey on Different Compression Techniques Algorithm for Data Compression*. International Journal of Advanced Research in Computer Science & Technology, Vol. 2, Issue 3.

[21] Rosenblatt. F.(1962). *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Spartan Books, Washington DC.

[22] Paraskevas. T., Dimitrios, R., & Andreas, B.(2014). *Use of Artificial Neural Network for Spatial Rainfall Analysis*. Journal of Earth System Science, 123,pp 457-465.

[23] Ariffin, S., Karim, A., Singh, B., Singh, M., Razali, R., & Yahya, N.(2011). *Data Compression Technique for Modeling of Global Solar Radiation*. IEEE International Conference on Control System, Computing and Engineering, pp 348 - 352.

CURRICULUM VITAE

Graduate College
University of Nevada, Las Vegas

Bharath Chandra Mummadisetty

Home Address:
1600 E,Rochelle Ave,Apt 16
Las Vegas,Nevada 89119

Degrees:
Bachelor of Engineering, Telecommunication Engineering, 2010
MVJ College of Engineering, Bangalore, India

Master of Science, Electrical Engineering, 2015
University of Nevada, Las Vegas

Thesis Title: Performance Analysis of Hybrid Algorithms for lossless compression of climate data

Thesis Examination Committee:

Chair, Dr. Shahram Latifi
Committee Member, Dr. Erbrahim Saberinia
Committee Member, Dr. Sahjendra Singh
Graduate College Representative, Dr.Wolfgang Bein