

A Comparative Study of LENN and CHFLANN Algorithm for Chronic Kidney Disease Data Classification

Susanta Mangar

M. tech (CSE)

Centre for Advanced Post Graduate (BPUT), Rourkela,
Odisha (769014), India

Abstract:- The essential elements of the kidney incorporate support of homeostasis through control of liquid, pH, and electrolyte equalization and blood pressure (BP). They are in charge of discharging metabolic final results and outside substances, just as the creation of the Renin chemical and the hormones 1,25-dihydroxycholecalciferol and erythropoietin. Harm to the kidney because of diabetes is the most well-known reason for Chronic Kidney Disease (CKD) in India . Type 2 to 4 chronic kidney disease can lead to consistent hyperglycemia makes the glomerulus thicken. Data mining plays a crucial role in data classification technologies. As Chronic kidney disease is the current ongoing research work in the medical science, the analysis of chronic kidney disease data has got a greater importance in the recent future. The better and faster is the data analysis technique the probability of finding the results is more accurate. Our proposed work is based on classification of diabetic data which has a greater application in current medical science research works .We have worked on two advanced neural networks, i.e. Legendre Neural Network (LeNN) and Chebyshev Functional Link based ANN (CHFLANN) and compared their performance with respect to the Accuracy and F measure for a sample chronic kidney disease dataset collected from uci machine learning repository database. By performing the simulations in MATLAB environment and analyzing the results our proposed Legendre Neural Network based architecture gives better performance as compared to the Chebyshev based approach.

CHAPTER 1

INTRODUCTION

Current Area of interest and research work on the health care segment is preventing the various diseases related to chronic kidney disease. In literature several data mining techniques have been proposed for finding out the main causes of chronic kidney disease considering several data sets. Data mining can be referred as the technique of extracting meaningful data simultaneously, analyzing and summarizing the useful information which can be used for prediction of future data or experiment. Development of data mining techniques to predict the class label of chronic kidney disease data and to calculate the misclassification error which can be further used in the patient safety research is an ongoing research area [1-3].

The primary functions of the kidney include maintenance of homeostasis through control of fluid, pH, and electrolyte balance and blood pressure (BP). They are responsible for excreting metabolic end-products and foreign substances, as well as the production of the Rennin enzyme and the hormones 1, 25-dihydroxycholecalciferol and erythropoietin (1).

Diabetes, hypertension and glomerulonephritis are the leading conditions that can lead to kidney failure (1,2). Other risk factors include autoimmune diseases, systemic infections, urinary stones or lower urinary tract obstruction, neoplasia, history of acute kidney injury, reduction in kidney mass, and low birth weight (2). Additional risk factors include ethnicity (African Americans, Asians, Pacific Islanders, and American Indians are at higher risk), hereditary factors, and prolonged consumption of over-the-counter painkillers such as aspirin, acetaminophen, and ibuprofen (1,2) low income and low education, age over 60, and the exposure to some chemical and environmental conditions (2).

Damage to the kidney due to diabetes is the most common cause of Chronic Kidney Disease (CKD) in the United States. Type 2 (and Type 1) diabetes can lead to CKD because the constant hyperglycemia causes the glomerulus to thicken. The glomerulus, which is normally responsible for filtering blood and fluids that form urine, is destroyed as the kidney starts allowing more of the albumin protein to be excreted in the urine. As the number of functioning nephrons decline, those that are left must clear more solute, until they reach a limit. At this point the concentration in body fluid increases, causing azotemia (a buildup of nitrogenous waste products such as urea in the blood and body fluids) and uremia (symptoms caused by disordered biochemical processes) (1).

1. *There are 5 stages of CKD (1):*

1. Kidney damage with normal or increased Glomerular Filtration Rate (GFR) (≥ 90)
2. Kidney damage with mild decrease in GFR (60-89)
3. Moderate decrease in GFR (30-59)
4. Severe decrease in GFR (15-29)
5. Kidney failure (GFR of >15); requires dialysis

GFR is defined as ml/min/1.73 m².

Although not an official stage, those with a GFR of ≥ 60 are at increased risk for CKD.

2. Upon admission, Mrs. Joaquin had the following signs/symptoms:

- Hypertension: B/P of 220/80
- High creatinine and BUN levels
- High Potassium and Phosphorus
- Anorexia
- Edema
- Progressive shortness of breath (SOB) with 3-pillow orthopnea
- Inability to urinate
- Malaise (feeling of unease)
- Muscle cramps
- Pruritus (itching)

1.1 MOTIVATION

Dialysis is a renal replacement procedure that replaces the filtering function of healthy kidneys by removing excessive and toxic by-products, wastes, and toxins in patients with CKD. Two types are hemodialysis (HD) and peritoneal dialysis (PD). Although both methods require a selective semi permeable membrane that allows passage of water and small- to middle-molecular weight molecules and ions, the selective membrane used in hemodialysis is a man-made dialyzer, while in peritoneal dialysis the lining of the peritoneal wall serves as the selective membrane. Waste and toxins are then removed by the fluid known as the dialysate (1).

In addition, during peritoneal dialysis, glucose is infused into the peritoneum through a catheter with the purpose of the dextrose concentration creating an osmotic gradient to remove excess fluid and toxins (3).

Nutrition Therapy	Rationale
35 kcal/kg	Adequate energy intakes are needed to maintain a proper body composition, reduce the risk for malnutrition, and prevent catabolism (1,3)
1.2 g protein/kg	A higher protein intake is recommended to maintain a proper protein balance and body composition to reduce the risk for protein-energy malnutrition (3)
2 g K	A low-potassium diet is needed to avoid hyperkalemia (3)
1 g phosphorus	Phosphorus intake should be decreased to avoid hyperphosphatemia (3)
2 g Na	A sodium restriction is recommended to avoid large interdialytic weight gains, hypertension, edema, pleural effusion and congestive heart failure (3)
1,000 ml fluid + urine output	A fluid restriction is recommended to avoid large interdialytic weight gains, hypertension, edema, pleural effusion and congestive heart failure (3)

BMI = $(170/60^2) \times 703 = 33.2$; a person with a BMI of 33 or above is considered obese. However, Mrs. Joaquin has 3+ pitting edema, and has gained 4kg in the past two weeks, which means much of her current weight is water retention.

Edema-free weight is calculated from an equation derived from NHANES II data (1). It is used to get a better estimate of the patient's actual weight minus the edema, and can be calculated as follows:

$$aBW_{ef} = BW_{ef} + [(SBW - BW_{ef}) \times 0.25] \rightarrow$$

$$165\text{lbs} + (100\text{lbs} - 165\text{lbs}) \times 0.25 = 148.75\text{lbs} (67.6\text{kg})$$

The energy requirements for CKD patients not on dialysis are 23kcal/kg to 35kcal/kg (2), while patients on hemodialysis require 35kcal/kg/day (1).

Mrs. Joaquin's energy needs once she is on hemodialysis will be, at 35kcal/kg:

$$67.6\text{kg} \times 35\text{kcal/kg} = 2,366 \text{ kcal/day}$$

Mrs. Joaquin's protein needs once she is on hemodialysis will be, at 1.2 g/kg:

$$67.6\text{kg} \times 1.2 \text{ g/kg} = 81 \text{ g protein/day}$$

Protein needs are higher for patients on dialysis due to protein losses from the treatment. Currently, Mrs. Joaquin's protein needs on hemodialysis are ≥ 1.2 kg/g of body weight. If she were on peritoneal dialysis (PD) they would be ≥ 1.2 to 1.3 g/kg (3). Her current calorie needs are at 35 kcal/kg of body weight. If she were on PD her needs would require the addition of any protein absorbed from the dialysate (3).

Vegetarian diets tend to be lower in protein, requiring that CKD patients on these diets monitor their protein status closely due to occurring losses during dialysis. Because plant proteins are a significant source of potassium and phosphorus, these minerals tend to be harder to control. Patients following a vegetarian diet may need phosphate binders to manage their phosphorus levels (3).

Currently, Mrs. Joaquin's labs reveal elevated PO₄ levels of 9.5 mg/dl Phosphate is avoided to prevent hyper phosphatemia in patients with CKD (1.3). Too much of it stimulates the parathyroid hormone, breaking down bone and leading to bone disease and osteoporosis.

Foods high in phosphorus include beverages such as ales, drinks made with chocolate or milk, canned iced teas, cocoa, beer, and dark colas; dairy products such as cheese, custard, milk, cream soups, cottage cheese, ice cream, pudding, and yogurt; animal proteins such as carp, beef liver, fish roe, oysters, crayfish, chicken liver, organ meats and sardines; legumes; whole grain products, bran cereals, nuts and seeds.

1.2 LITERATURE REVIEW

Today's computing environment credit card fraud detection is an important issue for both the common people and financial institution. Hence efficient model needs to be developed to prevent fraudsters from committing illegal activities. This study clearly reveals a comparative performance of CFLANN, MLP and Decision Tree over two different data sets for credit card fraud detection. Result shows that in both the data set MLP outperformed CFLANN and Decision Tree in fraud detection. Though FLANN with other input expansion has been successfully used in other area like prediction in which FLANN performed better than MLP but in credit card fraud detection MLP has slightly an edge over CFLANN. Future research may include comparison of MLP with few other FLANN networks with evolutionary learning technique for credit card fraud detection. Again more work needs to be done in choosing optimal network size.

Wilkinson, T. J. et.al "Characterizing skeletal muscle hemoglobin saturation during exercise using near-infrared spectroscopy in chronic kidney disease" described chronic kidney disease (CKD) patients have reduced exercise capacity. Possible contributing factors may include impaired muscle O₂ utilization through reduced mitochondria number and/or function slowing the restoration of muscle ATP concentrations via oxidative phosphorylation. Using near-infrared spectroscopy (NIRS), we explored changes in skeletal muscle hemoglobin/myoglobin O₂ saturation (SMO₂%) during exercise. They identified two discrete phases; a decline in SMO₂% during incremental exercise, followed by rapid increase upon cessation (recovery). Compared to patients with low exercise capacity [distance walked during ISWT, 269.0 (± 35.9) m], patients with a higher exercise capacity [727.1 (± 38.1) m] took 45% longer to reach their minimum SMO₂% (P = .038) and recovered (half-time recovery) 79% faster (P = .046). Compared to controls, CKD patients took significantly 56% longer to recover (i.e., restore SMO₂% to baseline, full recovery) (P = .014).

Lee, A. K. et.al "Indices of Kidney Tubular Health Improve Cardiovascular Disease Risk Prediction in Adults With Hypertension and Chronic Kidney Disease in SPRINT" stated While chronic kidney disease (CKD) is a strong risk factor for cardiovascular disease (CVD), traditional metrics of kidney function (eGFR and albuminuria) have not improved CVD risk prediction equations. We hypothesize that new measures of kidney tubular function and injury can improve CVD risk prediction. Of 1971 SPRINT participants with eGFR <60mL/min/1.73m² and without CVD at baseline, we analyzed 1858 with urine and serum biomarkers. We conducted factor analysis on 10 kidney tubule biomarkers using principal-component factor estimation and promax rotation. To examine the association between the factor scores and risk of subsequent cardiovascular events, we used adjusted Cox models. Using Harrell's C-statistic, we compared a standard CVD risk prediction model to models adding 1) factor scores of kidney tubular health, 2) eGFR and albumin-to-creatinine ratio (ACR), and 3) factor scores, eGFR and ACR. We also conducted these comparisons using ASCVD predicted risk in place of CVD risk factors among those <80 years. Found Indices of kidney tubular health based on urine and serum biomarkers may improve CVD risk prediction in adults with hypertensive CKD.

Jagdish C. Patra et.al proposed a novel computationally efficient neural network, named Legendre neural network (LeNN), for nonlinear channel equalization in digital communication systems with 4-QAM signal constellation. Since LeNN is a single layer NN, it has lower computational complexity and simpler implementation compared to MLP. We have carried out extensive simulations with several channels and nonlinear models and found the superiority of LeNN-based equalizer over MLP-based equalizer in terms of MSE convergence rate, BER and computational complexity. The proposed LeNN-based equalizer has similar performance as FLANN-based equalizer, which we had proposed earlier. Besides, as the computation of Legendre functions is less complex than trigonometric functions, LeNN takes less training-time than FLANN. Because of lower computational requirement and ability to perform complex mapping between multi-dimensional input and output spaces, the LeNN has potential applications in other areas of science and engineering.

Abdullah A. Aljumah et.al states the prevalence of chronic kidney disease is increasing among Saudi Arabian patients. The present study concludes that elderly chronic kidney disease patients should be given an assessment and a treatment plan that is suited to their needs and lifestyles. Public health awareness of simple measures such as low sugar diet, exercise, and avoiding obesity should be promoted by health care providers. In this study, predictions on the effectiveness of different treatment methods for young and old age groups were elucidated. The preferential orders of treatment were found to be different for the young and old age groups. Diet control, weight reduction, exercise and smoking cessation are mutually beneficial to each other for the treatment of chronic kidney disease. The collective and collaborative modes of these treatment, along with drug (type 2) and insulin (type 1) treatment, are found effective control the effects of chronic kidney disease.

Emirhan Gülçin et.al states the condition of high blood sugar (glucose) level is called as chronic kidney disease mellitus. Cause of this disease can be either insufficient insulin production or improper response of body cell to insulin. Chronic kidney disease patients should use different drugs in order to keep their blood sugar level within normal range values. The purpose of this study is to develop a data mining model for which will predict a suitable dosage planning for chronic kidney disease patients.

Medical records of 89 different patient records were used in this study. 318 chronic kidney disease assays were extracted using these patient records. ANFIS and Rough Set methods were used for dosage planning objective. According to the results of ANFIS and Rough Set methods, ANFIS is a more successful and reliable method for chronic kidney disease drug planning objective when compared to Rough Set method.

Joseph et.al "Data mining a chronic kidney disease data warehouse" shown that a data squashing algorithm to reduce a massive data set is more powerful and accurate than using a random sample [16]. Squashing is a form of lossy compression that attempts to preserve statistical information [15]. In our study, we used all the data available in the CART analysis and did not sample. If a massive dataset were used, such as all chronic kidney diseases in a nationwide Medicare database, memory and time constraints may require limitations on numbers of observations used. The newer data squashing techniques may be a better approach than random sampling in these massive datasets. Transactional healthcare data mining, exemplified in the chronic kidney disease data warehouses discussed above, involves a number of tricky data transformations that require close collaboration between domain experts and data miners [10]. Even with ideal collaboration or overlapping expertise, we need to develop new ways to extract variables from relational databases containing time series and sequencing information. Part of the answer lies in collaborative groups that can have additional insights. Part of the answer lies in the further development of data mining tools that act directly on a relational database without transformation to explicit data arrays.

Syed et.al "Identifying high-cost patients using data mining techniques and a small set of non-trivial attributes" proposed built multiple data mining models to predict very high-cost individuals in the top 5 percentile among General populations in the MEPS dataset. The CHAID decision tree classifier performs more accurate, and returns higher G-mean and AUC values compared to other classifiers including C5 decision trees and neural networks. We also identified the following 5 attributes as the small set of attributes to proactively predict very high-cost percentile (top5percentileofcosts) instances among the general population (ranked according to their estimating power and relevance in the final model): RTHLTH: perceived health status; AGE: age of the individual in years; ANYLIM: presence of any limitation(physical, sensory, or cognitive)in individual; BOWEL: time elapsed since and CHOLCK: time elapsed since last blood cholesterol check. This small set of attributes includes non-trivial but easy- to-survey measures of self-perception of health, age, along with two preventive health indicators (history of blood cholesterol checks, and colonic preventive interventions) and presence or absence of any physical, sensory, or cognitive limitations. Consequently, the results of this study are useful for policy makers, health planners, and insurers to proactively plan the delivery of healthcare services.

Ryan M. McCabe et.al "Using Data Mining to Predict Errors in Chronic Disease Care" presents the fact that physician and patient models were used to generate simulated encounter data from which data mining tools could be developed. The ability to use these tools to predict errors of omission in real (and synthetic) patient data suggests that future developments in this kind of work have the potential to enable identification and correction of physician decision making strategies that lead to encounter-specific treatment errors.21.

Santosh Kumar Nanda et.al proposed system models for noise prediction were validated using simulation studies. The studies were carried out by using MATLAB simulation environment. For validation of the model the data was collected from Central Pollution and Control Board, NewDelhi. Table1 represents the data set. In this propose system, NO₂, CO, SO₂ and O₃ are all the dependent parameters, where temperature and humidity are the independent parameters. In this research, due to availability of less data, here a procedure was follows. At first, using the data set in table1, statistical relationship or regression models with good R² were developed. Expressions 1.2 to 1.5 are represents the regression models. After Generating regression models, the system can design by suing these models. Training and testing data sets were generated using the regression models. The system network was trained with this dataset. In this proposed system, total number of 3200 data set were generated, where 3000 data set were used for training and rest of others were used as testing data set. Supervised training Method was applied here to train the intelligent system. It also represents the performance of the statistical model with LeNN model. It was seen that the performance of the proposed system was good prediction capacity as compared to regression models.

CHAPTER 2

NEURAL NETWORK

2.1 NEURAL NETWORKS

Based on sets of notes prepared by Dr. Patrick H. Corr, Brendan Coburn and John Gilligan

Neural Networks (also known as *Connectionist models* or *Parallel Distributed Processing models*) are information processing systems which model the brain's cognitive process by imitating some of its basic structures and operations. Interest in these

networks was originally biologically motivated. They were developed with the expectation of gaining new insight into the workings of the brain. In the 1940's and 1950's there was a certain amount of success in the research and development of neural networks but eventually the attraction of these systems declined due to a number of factors. However, within the past decade interest in neural networks has been revived. Although these networks are still helpful in the research of the brain's cognitive process, today they are actually implemented in the processing of information.

In 1943 two scientists, Warren McCulloch and Walter Pitts, proposed the first artificial model of a biological neuron [McC]. This synthetic neuron is still the basis for most of today's neural networks.

Rosenblatt came up with his two layered perceptron which was subsequently shown to be defective by Papert and Minsky which lead to a huge decline in funding and interest in Neural Networks.

2.1.1 Other Developments

During this period, even though there was a lack of funding and interest in neural networks, a small number of researchers continued to investigate the potential of neural models. A number of papers were published, but none had any great impact. Many of these reports concentrated on the potential of neural networks for aiding in the explanation of biological behavior (e.g. [Mal], [Bro], [Mar], [Bie], [Coo]). Others focused on real world implementations. In 1972 Teuvo Kohonen and James A. Anderson independently proposed the same model for associative memory [Koh], [An1] and in 1976 Marr and Poggio applied a neural network to a realistic problem in computational vision, stereopsis [Mar]. Other projects included [Lit], [Gr1], [Gr2], [Ama], [An2], [McC].

2.1.2 The Discovery of Back propagation

The back propagation learning algorithm was developed independently by Rumelhart [Ru1], [Ru2], Le Cun [Cun] and Parker [Par] in 1986. It was subsequently discovered that the algorithm had also been described by Paul Werbos in his Harvard PhD thesis in 1974 [Wer]. Error back propagation networks are the most widely used neural network model as they can be applied to almost any problem that requires pattern mapping. It was the discovery of this paradigm that brought neural networks out of the research area and into real world implementation.

A. *hat and why?*

Neural Networks: a bottom-up attempt to model the functionality of the brain.

Two main areas of activity:

- Biological
 - Try to model biological neural systems
- Computational
 - **Artificial** neural networks are biologically inspired but **not necessarily biologically plausible**
 - So may use other terms: Connectionism, Parallel Distributed Processing and Adaptive Systems Theory.

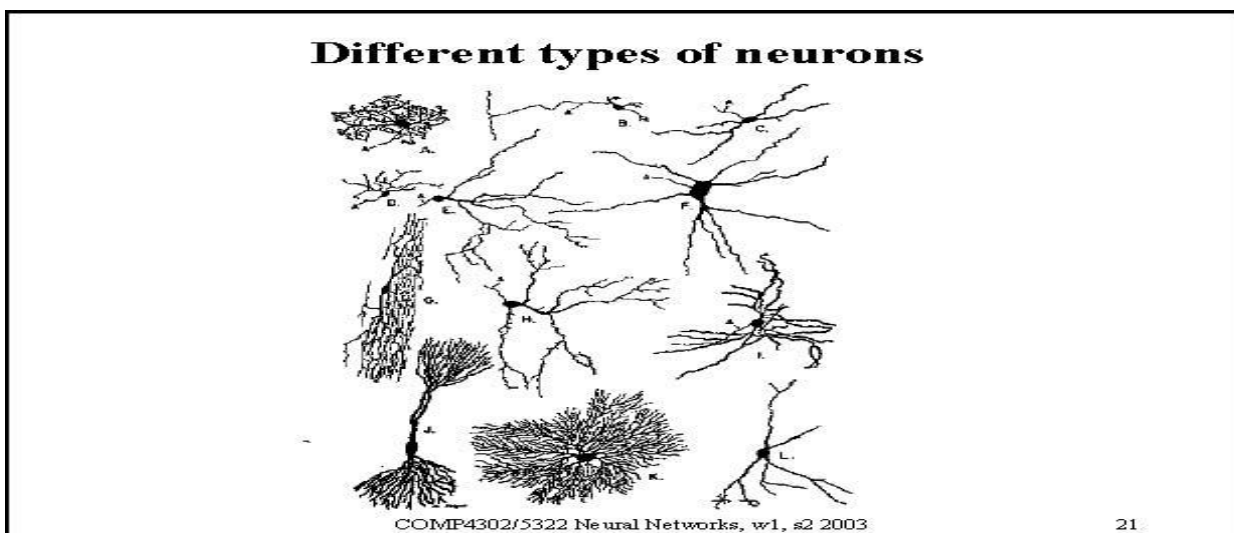


Figure 2.1 Different types of neurons

A simplified view of a neuron is shown in the diagram below.

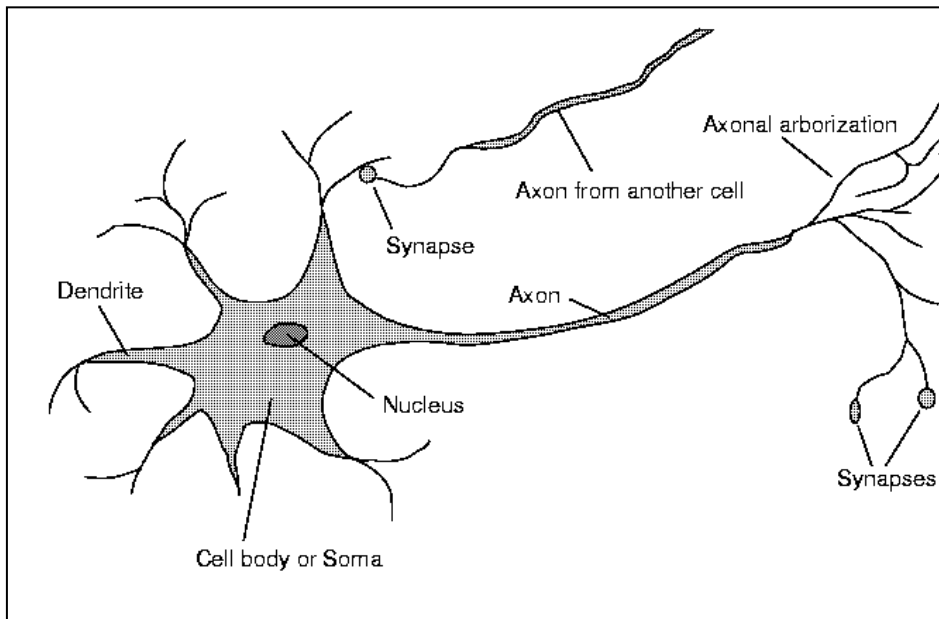


Figure 2.2 Simplified structures of neurons

Signals move from neuron to neuron via electrochemical reactions. The synapses release a chemical transmitter which enters the dendrite. This raises or lowers the electrical potential of the cell body.

The soma sums the inputs it receives and once a threshold level is reached an electrical impulse is sent down the axon (often known as firing).

These impulses eventually reach synapses and the cycle continues.

Synapses which raise the potential within a cell body are called *excitatory*. Synapses which lower the potential are called *inhibitory*.

It has been found that synapse exhibit *plasticity*. This means that long-term changes in the strengths of the connections can be formed depending on the firing patterns of other neurons. This is thought to be the basis for learning in our brains.

2.1.3 Modelling a Neuron

To model the brain we need to model a neuron. Each neuron performs a simple computation. It receives signals from its input links and it uses these values to compute the activation level (or output) for the neuron. This value is passed to other neurons via its output links.

The input value received of a neuron is calculated by summing the weighted input values from its input links. That is

$$in_i = \sum_j W_{j,i} a_j$$

An activation function takes the neuron input value and produces a value which becomes the output value of the neuron. This value is passed to other neurons in the network.

This is summarized in this diagram and the notes below.

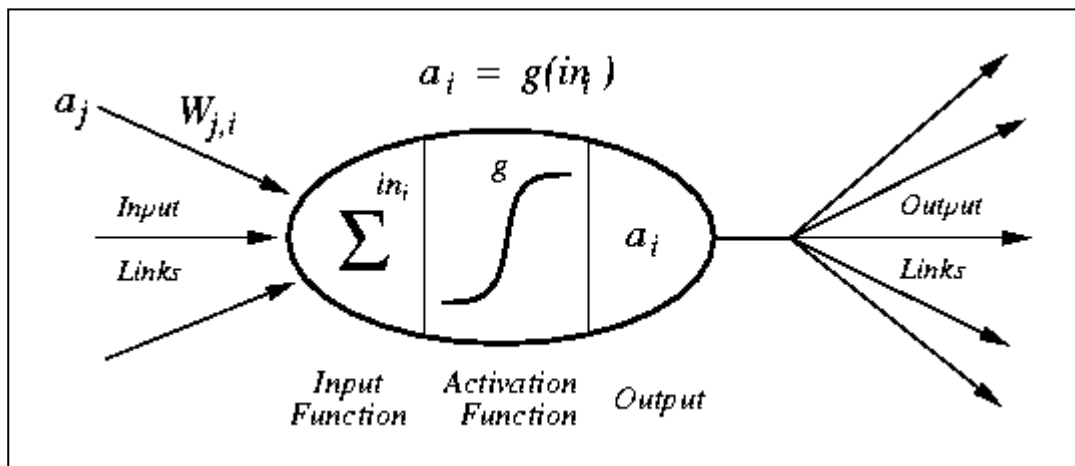


Figure 2.3 Summarized diagrams of neurons

- a_j : Activation value of unit j
- $w_{j,i}$: Weight on the link from unit j to unit i
- in_i : Weighted sum of inputs to unit i
- a_i : Activation value of unit i (also known as the output value)
- g : Activation function

A neuron is connected to other neurons via its input and output links. Each incoming neuron has an activation value and each connection has a weight associated with it.

The neuron sums the incoming weighted values and this value is input to an activation function. The output of the activation function is the output from the neuron.

Some common activation functions are shown below.

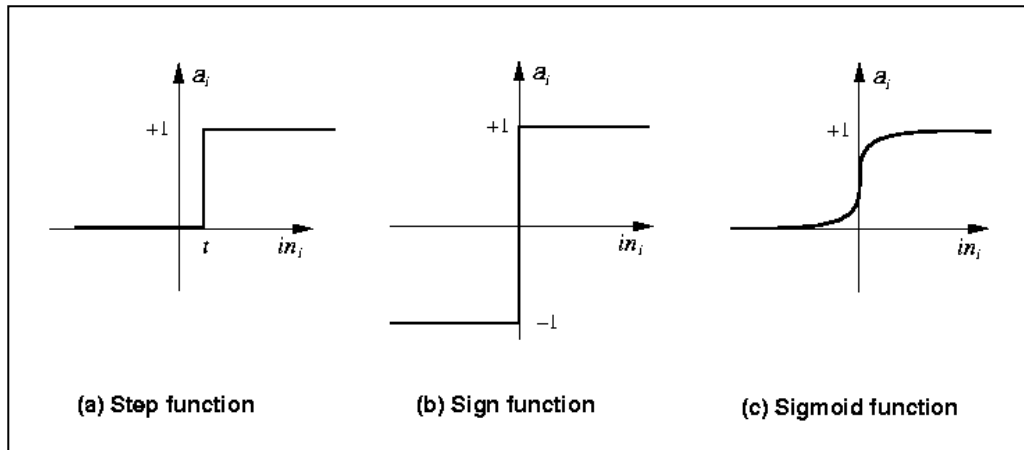


Figure 2.4 Activation functions of Neurons

These functions can be defined as follows.

$$\begin{aligned} \text{Step}_t(x) &= 1 \text{ if } x \geq t, \text{ else } 0 \\ \text{Sign}(x) &= +1 \text{ if } x \geq 0, \text{ else } -1 \\ \text{Sigmoid}(x) &= 1 / (1 + e^{-x}) \end{aligned}$$

On occasions an identity function is also used (i.e. where the input to the neuron becomes the output). This function is normally used in the input layer where the inputs to the neural network are passed into the network unchanged.

2.1.4 Interests in neural network differ according to profession.

Neurobiologists and psychologists - understanding our brain

Engineers and physicists - a tool to recognize patterns in noisy data (see Ts at right)

Business analysts and engineers - a tool for modeling data

Computer scientists and mathematicians - networks offer an alternative model of computing: machines that may be *taught* rather than programmed

Artificial Intelligence, cognitive scientists and philosophers – Sub symbolic processing (reasoning with patterns, not symbols)

Some Application Areas

A good overview of NN applications is provided on the pages set up by the DTI Neural Applications programmed and the later Smart software for decision makers programmed

- NCTT programmed
- Smart software for decision makers

2.1.5 Limitations In The Use Of Neural Networks

- Neural systems are inherently parallel but are normally simulated on sequential machines.
 - Processing time can rise quickly as the size of the problem grows - The **Scaling Problem**
 - However, a direct hardware approach would lose the flexibility offered by a software implementation.
 - In consequence, neural networks have been used to address only small problems.
- The performance of a network can be sensitive to the quality and type of preprocessing of the input data.
- Neural networks cannot explain the results they obtain; their rules of operation are completely unknown.
- Performance is measured by statistical methods giving rise to distrust on the part of potential users.
- Many of the design decisions required in developing an application are not well understood.

1) Comparison of neural techniques and symbolic artificial intelligence

Early work on neural systems was largely abandoned after serious limitations of earlier models were highlighted in 1969.

Growth of Artificial Intelligence based on the hypothesis that thought processes could be modeled using a set of symbols and applying a set of logical transformation rules.

The symbolic approach has a number of limitations:

- It is essentially sequential and difficult to parallelize
- When the quantity of data increases, the methods may suffer a combinatorial explosion.
- An item of knowledge is represented by a precise object, perhaps a byte in memory, or a production rule. This localized representation of knowledge does not lend itself to a robust system.
- The learning process seems difficult to simulate in a symbolic system.
- The connectionist approach offers the following advantages over the symbolic approach:
 - parallel and real-time operation of many different components
 - the distributed representation of knowledge
 - Learning by modifying connection weights.

MULTIPLE LAYER FEED FORWARD NETWORKS

Solving non linearly separable problems

As pointed out before, XOR is an example of a non linearly separable problem which two layer neural nets cannot solve. By adding another layer, the hidden layer, such problems can be solved.

2.2 Types of Neural Networks

There are many **types of artificial neural networks (ANN)**. Artificial neural networks are computational models inspired by biological neural networks, and are used to approximate functions that are generally unknown. Particularly, they are inspired by the behavior of neurons and the electrical signals they convey between input (such as from the eyes or nerve endings in the hand), processing, and output from the brain (such as reacting to light, touch, or heat).

2.2.1 Feed Forward neural network

The feed forward neural network was the first and arguably most simple type of artificial neural network devised. In this network the information moves in only one direction — forwards: From the input nodes data goes through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network. Feed forward networks can be constructed from different types of units, e.g. binary the simplest example being the perceptron. Continuous neurons, frequently with sigmoidal activation, are used in the context of back propagation of error.

2.2.2 Radial basis function network (RBF)

Radial basis functions are powerful techniques for interpolation in multidimensional space. A RBF is a function which has built into a distance criterion with respect to a center. Radial basis functions have been applied in the area of neural networks where they may be used as a replacement for the sigmoidal hidden layer transfer characteristic in multi-layer perceptrons. RBF networks have two layers of processing: In the first, input is mapped onto each RBF in the 'hidden' layer. The RBF chosen is usually a Gaussian. In regression problems the output layer is then a linear combination of hidden layer values representing mean predicted output. The interpretation of this output layer value is the same as a regression model in statistics. In classification problems the output layer is typically a sigmoid function of a linear combination of hidden layer values, representing a posterior probability. Performance in both cases is often improved by shrinkage techniques, known as ridge regression in classical statistics and known to correspond to a prior belief in small parameter values (and therefore smooth output functions) in a Bayesian framework. RBF networks have the advantage of not suffering from local minima in the same way as Multi-Layer Perceptrons. This is because the only parameters that are adjusted in the learning process are the linear mapping from hidden layer to output layer. Linearity ensures that the error surface is quadratic and therefore has a single easily found minimum. In regression problems this can be found in one matrix operation. In classification problems the fixed non-linearity introduced by the sigmoid output function is most efficiently dealt with using iteratively re-weighted least squares.

RBF networks have the disadvantage of requiring good coverage of the input space by radial basis functions. RBF centers are determined with reference to the distribution of the input data, but without reference to the prediction task. As a result, representational resources may be wasted on areas of the input space that are irrelevant to the learning task. A common solution

is to associate each data point with its own centre, although this can make the linear system to be solved in the final layer rather large, and requires shrinkage techniques to avoid over fitting.

2.2.3 Kohonen self-organizing network

The self-organizing map (SOM) performs a form of unsupervised learning. A set of artificial neurons learn to map points in an input space to coordinates in an output space. The input space can have different dimensions and topology from the output space and the SOM will attempt to preserve these.

2.2.4 Spiking neural network

Spiking neural networks (SNNs) are models which explicitly take into account the timing of inputs. The network input and output are usually represented as series of spikes (delta function or more complex shapes). SNNs have an advantage of being able to process information in the time domain (signals that vary over time). They are often implemented as recurrent networks. SNNs are also a form of pulse computer.

Spiking neural networks with axonal conduction delays exhibit polychronization and hence could have a very large memory capacity. Networks of spiking neurons — and the temporal correlations of neural assemblies in such networks — have been used to model figure/ground separation and region linking in the visual system.

2.3 Multi Layer Perceptron (MLP):

A **multilayer perceptron (MLP)** is a feed forward artificial neural network model that maps sets of input data onto a set of appropriate outputs. A MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called back propagation for training the network.[1][2] MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable

MLP is a network of simple neurons called perceptrons [15]. The basic concept of a single perceptron was introduced by Rosenblatt in 1958. The perceptron computes a single output from multiple real-valued inputs by forming a linear combination according to its input weights and then possibly putting the output through some nonlinear activation function. Mathematically this can be written as

$$y = \Psi\left(\sum_{i=1}^n w_i x_i + b\right) = \Psi(w^T x + b) \quad (3)$$

Where, w denotes the vector of weights, x is the vector of inputs, b is the bias and Ψ is the activation function.

ACTIVATION FUNCTION

If a multilayer perceptron has a linear activation function in all neurons, that is, a linear function that maps the weighted inputs to the output of each neuron, then it is easily proved with linear algebra that any number of layers can be reduced to the standard two-layer input-output model (see perceptron). What makes a multilayer perceptron different is that some neurons use a nonlinear activation function which was developed to model the frequency of action potentials, or firing, of biological neurons in the brain. This function is modeled in several ways.

The two main activation functions used in current applications are both sigmoids, and are described by

$$y(v_i) = \tanh(v_i) \quad \text{and} \quad y(v_i) = (1 + e^{-v_i})^{-1},$$

In which the former function is a hyperbolic tangent which ranges from -1 to 1, and the latter, the logistic function, is similar in shape but ranges from 0 to 1. Here y_i is the output of the i th node (neuron) and v_i is the weighted sum of the input synapses. Alternative activation functions have been proposed, including the rectifier and soft plus functions. More specialized activation functions include radial basis functions which are used in another class of supervised neural network models.

Layers

The multilayer perceptron consists of three or more layers (an input and an output layer with one or more hidden layers) of nonlinearly-activating nodes and is thus considered a deep neural network. Each node in one layer connects with a certain weight w_{ij} to every node in the following layer. Some people do not include the input layer when counting the number of layers and there is disagreement about whether w_{ij} should be interpreted as the weight from i to j or the other way around.

Learning through backpropagation

Learning occurs in the perceptron by changing connection weights after each piece of data is processed, based on the amount of error in the output compared to the expected result. This is an example of supervised learning, and is carried out through back propagation, a generalization of the least mean squares algorithm in the linear perceptron.

We represent the error in output node j in the n th data point (training example) by $e_j(n) = d_j(n) - y_j(n)$, where d_j is the target value and y_j is the value produced by the perceptron. We then make corrections to the weights of the nodes based on those corrections which minimize the error in the entire output, given by

$$\mathcal{E}(n) = \frac{1}{2} \sum_j e_j^2(n)$$

Using gradient descent, we find our change in each weight to be

$$\Delta w_{ji}(n) = -\eta \frac{\partial \mathcal{E}(n)}{\partial v_j(n)} y_i(n)$$

where y_i is the output of the previous neuron and η is the learning rate, which is carefully selected to ensure that the weights converge to a response fast enough, without producing oscillations. In programming applications, this parameter typically ranges from 0.2 to 0.8

The derivative to be calculated depends on the induced local field v_j , which it varies. It is easy to prove that for an output node this derivative can be simplified to

$$-\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = e_j(n) \phi'(v_j(n))$$

where ϕ' is the derivative of the activation function described above, which itself does not vary. The analysis is more difficult for the change in weights to a hidden node, but it can be shown that the relevant derivative is

$$-\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = \phi'(v_j(n)) \sum_k -\frac{\partial \mathcal{E}(n)}{\partial v_k(n)} w_{kj}(n)$$

This depends on the change in weights of the k th nodes, which represent the output layer. So to change the hidden layer weights, we must first change the output layer weights according to the derivative of the activation function, and so this algorithm represents a back propagation of the activation function

APPLICATIONS

Multilayer perceptrons using a back propagation algorithm are the standard algorithm for any supervised learning pattern recognition process and the subject of ongoing research in computational neuroscience and parallel distributed processing. They are useful in research in terms of their ability to solve problems stochastically, which often allows one to get approximate solutions for extremely complex problems like fitness approximation.

MLPs were a popular machine learning solution in the 1980s, finding applications in diverse fields such as speech recognition, image recognition, and machine translation software,^[5] but have since the 1990s faced strong competition from the much simpler (and related^[6]) support vector machines. More recently, there has been some renewed interest in back propagation networks due to the successes of deep learning.

2.4 Functional Link Artificial Neural Network (FLANN)

It is a mathematical model or computational model that is inspired by structural and/or functional aspects of biological neural networks. It consists of an interconnected group of artificial neurons and it processes information using a model.

At the present time, owing to the improvements in technology through superior energy competence, higher labor output, continuous production methods, and operating flexibility, automation has also advanced rapidly in open and underground pits together with mineral processing plants. In parallel to this improvement, sources of noise and ambient noise at work place in the mining industry have increased significantly. In general, noise is generated from all most all the opencast mining operations from different fixed, mobile, and impulsive sources, thereby becoming an integral part of the mining environment. With increased mechanization, the problem of noise has got accentuated in opencast mines. Prolonged exposure of miners to the high levels of noise can cause noise-induced hearing loss besides several non auditory health effects [. The impact of noise in opencast mines depends upon the sound power level of the noise generators, prevailing geo mining conditions and the meteorological parameters of the mines. The noise levels need to be studied as an integrated effect of the above parameters. In mining conditions the equipment conditions and environment continuously change as the mining activity progresses. Depending on their placement, the overall mining noise emanating from the mines varies in quality and level. Thus, for environmental noise prediction models, the noise level at any receiver point needs to be the resultant sound pressure level of all the noise sources.

The need for accurately predicting the level of sound emitted in opencast mines is well established. Some of the noise forecasting models used extensively in Europe are those of the German draft standard VDI-2714 outdoor sound propagation and environmental noise model (ENM) of Australia]. These models are generally used to predict noise in petrochemical complexes, and mines. The algorithm used in these models rely for a greater part on interpolation of experimental data which is a valid and useful technique, but their applications are limited to sites which are more or less similar to those for which the experimental data were assimilated.

A number of models were developed and extensively used for the assessment of sound pressure level and their attenuation around industrial complexes. Generally, in Indian mining industry, environmental noise model developed by RTA group, Australia is mostly used to predict noise. ENM was used to predict sound pressure level in mining complexes at Moonidih Project in Jharia Coalfield, Dhanbad, India. The applied model output was represented as noise contours. The application of different noise prediction models was studied for various mines and petrochemical complexes and it was reported that VDI2714 model was the simplest and least complex model vis-à-vis other models. VDI2714 and ISO (1996) noise prediction models were used in Asyut cement plant, Asyut cement quarry and El-Gedida mine at El-Baharia oasis of Egypt to predict noise. From the study, it was concluded that the prediction models could be used to identify the safe zones with respect to the noise level in mining and industrial plants. It was also inferred that the VDI2714 model is the simplest model for prediction of noise in mining complexes and workplace. Air attenuation model was developed for noise prediction in limestone quarry and mines of Ireland. The model was used to predict attenuation in air due to absorption.

All the noise prediction models treat noise as a function of distance, sound power level, different form of attenuations such as geometrical absorptions, barrier effects, ground topography. Generally, these parameters are measured in the mines and best fitting models are applied to predict noise. Mathematical models are generally complex and cannot be implemented in real time systems. Additionally, they fail to predict the future parameters from current and past measurements. It has been seen that noise prediction is a non stationary process and soft-computing techniques like fuzzy system, adaptive neural network-based fuzzy inference system (ANFIS), neural network, and so forth, have been tested for non stationary time-series prediction nearly for two decades. There is a scope of using different soft computing techniques: fuzzy logic, artificial neural networks, radial basis functions (RBF) and so forth, for noise prediction in mines. In comparison to other soft computing techniques, functional link artificial neural network (FLANN) and Legendre Neural Network (LeNN) has less computational cost and easily implemented in hardware applications. This is the motivation on which the present research work is based.

In this paper, an attempt has been made to develop three types of functional link artificial neural network models (FLANN, PPN, and LeNN) for noise prediction of machineries used in Balaram opencast coal mine of Talcher, Orissa, India. The data assembled through surveys, measurement or knowledge to predict sound pressure level in mines is often imprecise or speculative. Since neural network-based systems are good predictive tools for imprecise and uncertainty information; therefore, the proposed approach would be the most appropriate technique for modeling the prediction of sound pressure level in opencast coal mines.

CHAPTER 3

LEGENDRE NEURAL NETWORK

3.1 Legendre polynomials

Associated Legendre polynomials are the most general solution to Legendre's Equation and Legendre polynomials are solutions that are azimuthally symmetric.

In mathematics, Legendre functions are solutions to Legendre's differential equation:

$$\frac{d}{dx} \left[(1-x^2) \frac{d}{dx} P_n(x) \right] + n(n+1)P_n(x) = 0. \quad (1)$$

They are named after Adrien-Marie Legendre. This ordinary differential equation is frequently encountered in physics and other technical fields. In particular, it occurs when solving Laplace's equation (and related partial differential equations) in spherical coordinates.

The Legendre differential equation may be solved using the standard power series method. The equation has regular singular points at $x = \pm 1$ so, in general, a series solution about the origin will only converge for $|x| < 1$. When n is an integer, the solution $P_n(x)$ that is regular at $x = 1$ is also regular at $x = -1$, and the series for this solution terminates (i.e. it is a polynomial). These solutions for $n = 0, 1, 2, \dots, n$ (with the normalization $P_n(1) = 1$) form a polynomial sequence of orthogonal polynomials called the Legendre polynomials. Each Legendre polynomial $P_n(x)$ is an n th-degree polynomial. It may be expressed using Rodrigues' formula:

$$P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} [(x^2 - 1)^n].$$

That these polynomials satisfy the Legendre differential equation (1) follows by differentiating $n + 1$ times both sides of the identity

$$(x^2 - 1) \frac{d}{dx} (x^2 - 1)^n = 2nx(x^2 - 1)^n$$

and employing the general Leibniz rule for repeated differentiation.[1] The P_n can also be defined as the coefficients in a Taylor series expansion:[2]

$$\frac{1}{\sqrt{1 - 2xt + t^2}} = \sum_{n=0}^{\infty} P_n(x)t^n.$$

In physics, this ordinary generating function is the basis for multiple expansions.

$$\begin{aligned} P_n(x) &= \frac{1}{2^n} \sum_{k=0}^n \binom{n}{k}^2 (x - 1)^{n-k} (x + 1)^k \\ &= \sum_{k=0}^n \binom{n}{k} \binom{-n-1}{k} \left(\frac{1-x}{2}\right)^k \\ &= 2^n \cdot \sum_{k=0}^n x^k \binom{n}{k} \binom{\frac{n+k-1}{2}}{n}, \end{aligned}$$

The first few Legendre polynomials are:

n	$P_n(x)$
0	1
1	x
2	$\frac{1}{2}(3x^2 - 1)$
3	$\frac{1}{2}(5x^3 - 3x)$
4	$\frac{1}{8}(35x^4 - 30x^2 + 3)$
5	$\frac{1}{8}(63x^5 - 70x^3 + 15x)$
6	$\frac{1}{16}(231x^6 - 315x^4 + 105x^2 - 5)$
7	$\frac{1}{16}(429x^7 - 693x^5 + 315x^3 - 35x)$
8	$\frac{1}{128}(6435x^8 - 12012x^6 + 6930x^4 - 1260x^2 + 35)$
9	$\frac{1}{128}(12155x^9 - 25740x^7 + 18018x^5 - 4620x^3 + 315x)$
10	$\frac{1}{256}(46189x^{10} - 109395x^8 + 90090x^6 - 30030x^4 + 3465x^2 - 63)$

The graphs of these polynomials (up to $n = 5$) are shown below:

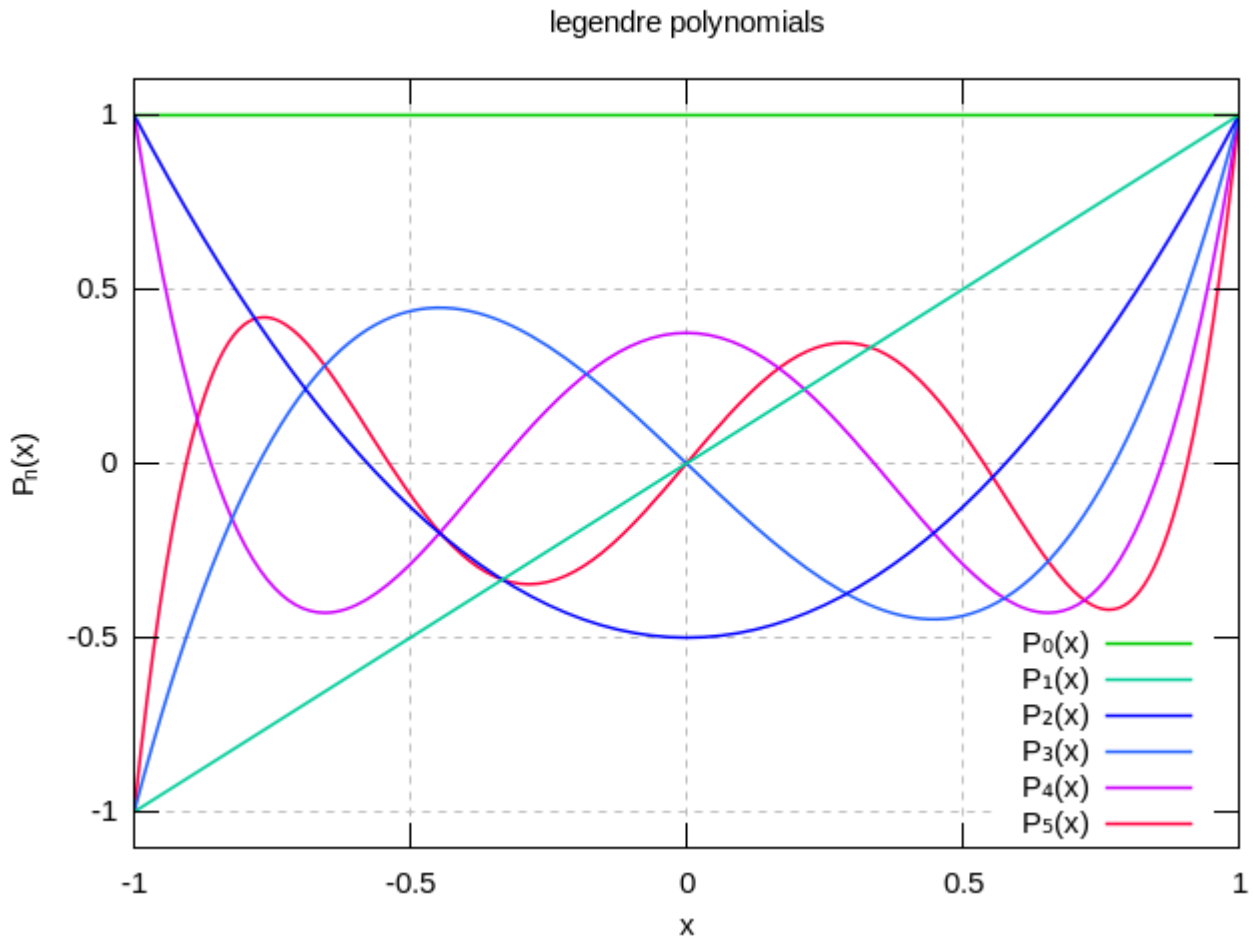


Figure 3.1 Legendre Polynomials

3.2 ORTHOGONALITY

An important property of the Legendre polynomials is that they are orthogonal with respect to

the L^2 inner product on the interval $-1 \leq x \leq 1$:

$$\int_{-1}^1 P_m(x)P_n(x) dx = \frac{2}{2n + 1} \delta_{mn}$$

(Where δ_{mn} denotes the Kronecker delta, equal to 1 if $m = n$ and to 0 otherwise). In fact, an alternative derivation of the Legendre polynomials is by carrying out the Gram–Schmidt process on the polynomials $\{1, x, x^2, \dots\}$ with respect to this inner product. The reason for this orthogonality property is that the Legendre differential equation can be viewed as a Sturm–Lowville problem, where the Legendre polynomials are Eigen of a Hermitian differential operator:

$$\frac{d}{dx} \left[(1 - x^2) \frac{d}{dx} P(x) \right] = -\lambda P(x),$$

Where the Eigen value λ corresponds to $n(n + 1)$.

3.3 APPLICATION OF LEGENDRE POLYNOMIALS

The Legendre polynomials were first introduced in 1782 by Adrien-Marie Legendre ^[3] as the coefficients in the expansion of the Newtonian potential

$$\frac{1}{|\mathbf{x} - \mathbf{x}'|} = \frac{1}{\sqrt{r^2 + r'^2 - 2rr' \cos \gamma}} = \sum_{\ell=0}^{\infty} \frac{r'^{\ell}}{r^{\ell+1}} P_{\ell}(\cos \gamma)$$

Where r and r' are the lengths of the vectors \mathbf{X} and \mathbf{X}' respectively and γ is the angle between those two vectors. The series converges when $r > r'$. The expression gives the gravitational potential associated to a point mass or the Coulomb potential associated to a point charge. The expansion using Legendre polynomials might be useful, for instance, when integrating this expression over a continuous mass or charge distribution.

Legendre polynomials occur in the solution of Laplace's equation of the static potential, $\nabla^2\Phi(\mathbf{x}) = 0$, in a charge-free region of space, using the method of separation of variables, where the boundary conditions have axial symmetry (no dependence on an azimuthal angle). Where $\hat{\mathbf{z}}$ is the axis of symmetry and θ is the angle between the position of the observer and the $\hat{\mathbf{z}}$ axis (the zenith angle), the solution for the potential will be

$$\Phi(r, \theta) = \sum_{\ell=0}^{\infty} [A_{\ell}r^{\ell} + B_{\ell}r^{-(\ell+1)}] P_{\ell}(\cos\theta).$$

A_{ℓ} and B_{ℓ} are to be determined according to the boundary condition of each problem.^[4]

They also appear when solving Schrödinger equation in three dimensions for a central force.

CHAPTER 4

CHEBYSHEV FUNCTIONAL LINK ARTIFICIAL NEURAL NETWORK

4.1 CHEBYSHEV POLYNOMIALS

In mathematics the **Chebyshev polynomials**, named after Pafnuty Chebyshev, are a sequence of orthogonal polynomials which are related to de Moivre's formula and which can be defined recursively. One usually distinguishes between **Chebyshev polynomials of the first kind** which are denoted T_n and **Chebyshev polynomials of the second kind** which are denoted U_n . The letter T is used because of the alternative transliterations of the name *Chebyshev* as *Tchebycheff*, *Tchebyshev* (French) or *Tschebyschow* (German).

The Chebyshev polynomials T_n or U_n are polynomials of degree n and the sequence of Chebyshev polynomials of either kind composes a polynomial sequence.

Chebyshev polynomials are polynomials with the largest possible leading coefficient, but subject to the condition that their absolute value on the interval $[-1,1]$ is bounded by 1. They are also the extremal polynomials for many other properties.^[2]

Chebyshev polynomials are important in approximation theory because the roots of the Chebyshev polynomials of the first kind, which are also called Chebyshev nodes, are used as nodes in polynomial interpolation. The resulting interpolation polynomial minimizes the problem of Runge's phenomenon and provides an approximation that is close to the polynomial of best approximation to a continuous function under the maximum norm. This approximation leads directly to the method of Clenshaw–Curtis quadrature.

In the study of differential equations they arise as the solution to the Chebyshev differential equations

$$(1 - x^2) y'' - x y' + n^2 y = 0$$

and

$$(1 - x^2) y'' - 3x y' + n(n + 2) y = 0$$

for the polynomials of the first and second kind, respectively. These equations are special cases of the Sturm–Lowville differential equation.

The **Chebyshev polynomials of the first kind** are defined by the recurrence relation

$$T_0(x) = 1$$

$$T_1(x) = x$$

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x).$$

The ordinary generating function for T_n is

$$\sum_{n=0}^{\infty} T_n(x)t^n = \frac{1 - tx}{1 - 2tx + t^2};$$

the exponential generating function is

$$\sum_{n=0}^{\infty} T_n(x) \frac{t^n}{n!} = \frac{1}{2} \left(e^{(x-\sqrt{x^2-1})t} + e^{(x+\sqrt{x^2-1})t} \right) = e^{tx} \cosh(t\sqrt{x^2-1}).$$

The generating function relevant for 2-dimensional potential theory and multiple expansion is

$$\sum_{n=1}^{\infty} T_n(x) \frac{t^n}{n} = \ln \frac{1}{\sqrt{1-2tx+t^2}}.$$

The **Chebyshev polynomials of the second kind** are defined by the recurrence relation

$$U_0(x) = 1$$

$$U_1(x) = 2x$$

$$U_{n+1}(x) = 2xU_n(x) - U_{n-1}(x).$$

The ordinary generating function for U_n is

$$\sum_{n=0}^{\infty} U_n(x)t^n = \frac{1}{1-2tx+t^2};$$

the exponential generating function is

$$\sum_{n=0}^{\infty} U_n(x) \frac{t^n}{n!} = e^{tx} \left(\cosh(t\sqrt{x^2-1}) + \frac{x}{\sqrt{x^2-1}} \sinh(t\sqrt{x^2-1}) \right).$$

The Chebyshev polynomials of the first kind can be defined as the unique polynomials satisfying

$$T_n(x) = \cos(n \arccos x) = \cosh(n \operatorname{arcosh} x)$$

or, in other words, as the unique polynomials satisfying

$$T_n(\cos(\vartheta)) = \cos(n\vartheta)$$

for $n = 0, 1, 2, 3, \dots$ which is a variant (equivalent transpose) of Schröder's equation, viz. $T_n(x)$ is functionally conjugate to nx , codified in the nesting property below. Further compare to the spread polynomials, in the section below.

The polynomials of the second kind satisfy:

$$U_n(\cos(\vartheta)) = \frac{\sin((n+1)\vartheta)}{\sin \vartheta},$$

which is structurally quite similar to the Dirichlet kernel $D_n(x)$:

$$D_n(x) = \frac{\sin((2n+1)(x/2))}{\sin(x/2)} = U_{2n}(\cos(x/2)).$$

That $\cos(nx)$ is an n th-degree polynomial in $\cos(x)$ can be seen by observing that $\cos(nx)$ is the real part of one side of de Moivre's formula, and the real part of the other side is a polynomial in $\cos(x)$ and $\sin(x)$, in which all powers of $\sin(x)$ are even and thus replaceable through the identity $\cos^2(x) + \sin^2(x) = 1$.

This identity is quite useful in conjunction with the recursive generating formula, inasmuch as it enables one to calculate the cosine of any integral multiple of an angle solely in terms of the cosine of the base angle.

Evaluating the first two Chebyshev polynomials,

$$T_0(x) = \cos(0x) = 1$$

And

$$T_1(\cos(x)) = \cos(x),$$

One can straightforwardly determine that

$$\cos(2\vartheta) = 2 \cos \vartheta \cos \vartheta - \cos(0\vartheta) = 2 \cos^2 \vartheta - 1$$

$$\cos(3\vartheta) = 2 \cos \vartheta \cos(2\vartheta) - \cos \vartheta = 4 \cos^3 \vartheta - 3 \cos \vartheta,$$

And so forth.

Two immediate corollaries are the *composition identity* (or **nesting property** specifying a semi group)

$$T_n(T_m(x)) = T_{nm}(x);$$

and the expression of complex exponentiation in terms of Chebyshev polynomials: given $z = a + bi$,

$$\begin{aligned} z^n &= |z|^n \left(\cos \left(n \arccos \frac{a}{|z|} \right) + i \sin \left(n \arccos \frac{a}{|z|} \right) \right) \\ &= |z|^n T_n \left(\frac{a}{|z|} \right) + ib |z|^{n-1} U_{n-1} \left(\frac{a}{|z|} \right). \end{aligned}$$

The Chebyshev polynomials can also be defined as the solutions to the Pell equation

$$T_n(x)^2 - (x^2 - 1)U_{n-1}(x)^2 = 1$$

in a ring $R[x]$.^[3] Thus, they can be generated by the standard technique for Pell equations of taking powers of a fundamental solution:

4.2 DERIVATION FROM DIFFERENTIAL EQUATION

All of the polynomial sequences arising from the differential equation above are equivalent, under scaling and/or shifting of the domain, and standardizing of the polynomials, to more restricted classes. Those restricted classes are exactly "classical orthogonal polynomials".

- Every Jacobi-like polynomial sequence can have its domain shifted and/or scaled so that its interval of orthogonality is $[-1, 1]$, and has $Q = 1 - x^2$. They can then be standardized into the **Jacobi polynomials** $P_n^{(\alpha, \beta)}$. There are several important subclasses of these: **Gegenbauer**, **Legendre**, and two types of **Chebyshev**.

- Every Leaguered-like polynomial sequence can have its domain shifted, scaled, and/or reflected so that its interval of orthogonality is $[0, \infty)$, and has $Q = x$. They can then be standardized into the **Associated Laguerre polynomials** $L_n^{(\alpha)}$. The plain **Leaguered polynomials** L_n are a subclass of these.

- Every Hermit-like polynomial sequence can have its domain shifted and/or scaled so that its interval of orthogonality is $(-\infty, \infty)$, and has $Q = 1$ and $L(0) = 0$. They can then be standardized into the **Hermit polynomials** H_n .

Because all polynomial sequences arising from a differential equation in the manner described above are trivially equivalent to the classical polynomials, the actual classical polynomials are always used.

Chebyshev polynomials

The differential equation is

$$(1 - x^2) y'' - x y' + \lambda y = 0 \quad \text{with} \quad \lambda = n^2.$$

This is **Chebyshev's equation**.

The recurrence relation is

$$T_{n+1}(x) = 2x T_n(x) - T_{n-1}(x).$$

Rodrigues' formula is

$$T_n(x) = \frac{\Gamma(1/2) \sqrt{1-x^2}}{(-2)^n \Gamma(n+1/2)} \frac{d^n}{dx^n} \left([1-x^2]^{n-1/2} \right).$$

These polynomials have the property that, in the interval of orthogonality,

$$T_n(x) = \cos(n \arccos(x)).$$

(To prove it, use the recurrence formula.)

This means that all their local minima and maxima have values of -1 and $+1$, that is, the polynomials are "level". Because of this, expansion of functions in terms of Chebyshev polynomials is sometimes used for polynomial approximations in computer math libraries.

Some authors use versions of these polynomials that have been shifted so that the interval of orthogonality is $[0, 1]$ or $[-2, 2]$.

There are also **Chebyshev polynomials of the second kind**, denoted U_n

We have:

$$U_n = \frac{1}{n+1} T'_{n+1}$$

For further details, including the expressions for the first few polynomials, see Chebyshev polynomials.

CHAPTER 5

COMPARATIVE STUDY FOR LeNN AND CHFLANN

5.1 INTRODUCTION TO DATA MINING TECHNIQUES EMPLOYED IN THE STUDY

In recent days a wide application of data mining techniques is available for chronic kidney disease data classification. The detailed procedure of developing a classification model using the Legendre Neural networks (LeNN) and Chebyshev Functional Link Artificial Neural network (CHFLANN) for chronic kidney disease data classification is discussed in the following section.

A. LEGENDRE NEURAL NETWORK

The structure of Legendre Neural Network (LeNN) is similar to FLANN (Functional link artificial neural network). In FLANN, trigonometric functions are used in the functional expansion and LeNN uses Legendre orthogonal functions [4-9]. The architecture of the LeNN consists of two components: functional expansion and learning component. The functional expansion component helps to capture the nonlinearity present in input features instead of using a number of hidden layers as done in MLP. The learning component is the single neuron present in the output layer. Due to the absence of hidden layer the complexity and the training time of the LeNN network are much less as compared to well known MLP with hidden layers.

The Legendre polynomials are denoted by $L_n(X)$, where n is the order and $-1 < x < 1$ is the argument of the polynomial.

The zero and the first order Legendre polynomial are, respectively given by

$$L_0(x) = 1 \text{ and } L_1(x) = x$$

The higher order polynomials are given by (1)

$$L_2(x) = 1/2(3x^2 - 1)$$

$$L_3(x) = 1/2(5x^3 - 3x) \quad (2)$$

$$L_4(x) = 1/8(35x^4 - 30x^2 + 3)$$

The recursive formula to generate higher order Legendre polynomials is expressed as

$$L_{n+1}(x) = 1/n+1 [(2n+1)xL_n(x) - nL_{n-1}(x)] \quad (3)$$

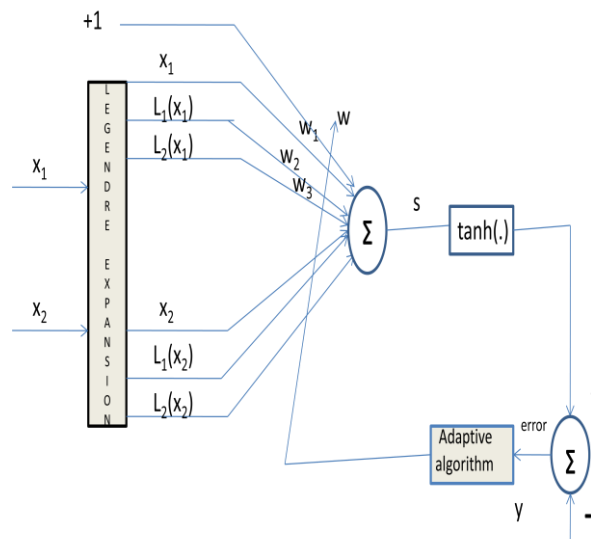


Fig 5.1 Architecture of Legendre neural network

The schematic diagram for the LeNN showing the functional Expansions and learning components is given in figure I. With a suitable order n , initially the d dimensional input pattern is expanded using the recursive equation. Then weight for each expanded input is initialized randomly and the weighted sum of the components of the enhanced input pattern is obtained using the following formula

$$\text{Weighted Sum} = w_j L_j(x) \quad (4)$$

The error obtained by comparing the output with desired output is used to update the weights of the network structure by a weight updating algorithm. Back Propagation is one of the popular training algorithms for neural network. In each iteration, the gradient of the cost function with respect the weights are determined and the weights are incremented by a fraction of the negative gradient. Let the cost function at kth instant is.

$$E_k = \frac{1}{2} [d_k - y_k]^2 = \frac{1}{2} e_k^2 \quad (5)$$

$$\frac{\partial E_k}{\partial W} = e_k \cdot \frac{\partial y_k}{\partial W}$$

The gradient of the cost function is given by

If nonlinear tanh() function is used at the output node, the update rule for the weight w_{ji} becomes

$$w_{j,k+1} = w_{j,k} + \alpha e_k (1 - y_k)^2 L_j(X) \quad (7)$$

$$w_{j,k+1} = \text{weight at } k + 1_{th} \text{ ins tan ce}$$

$$w_{j,k} = \text{weight at } k_{th} \text{ ins tan ce}$$

$$e_k = \text{error at } k_{th} \text{ ins tan ce}$$

$$y_k = \text{output obtained at } k_{th} \text{ ins tan ce}$$

$$L_j(X) = \text{value of } j_{th} \text{ legendre unit}$$

$$\alpha = \text{learning rate}$$

5.2 ALGORITHM FOR THE DESIGN

STEP 1: Normalize the input features of the chronic kidney disease data set.

STEP 2: Prepare the training and testing data set.

STEP 3: Set appropriate order for expansion and expand each input pattern.

STEP 4: Initialize a set of weights based on expansion order.

STEP 5: Find the weighted sum of the expanded input and passing it through a nonlinear function generate the output.

Step 6: Compare the output with desired output to generate the error and update the weights of the network through appropriate learning algorithm till termination condition is reached.

Step 7: After training the network fixed the weights and use it for testing.

Step 8: Find the class label of the input data set by using appropriate threshold value for the dataset.

If output > 0.5

Class value is 1

Else

Class Value is 0

B. CHEBYSHEV FUNCTIONAL LINK ARTIFICIAL NEURAL NETWORK

CHFLANN (Chebyshev Functional Link Neural Network) is a single layer neural network in which the original input pattern in lower dimensional is expanded to a higher dimensional space by using a set of Chebyshev orthogonal functions [10-13]. The Chebyshev polynomials are set of polynomial denoted by CH_p(X), where p refers to the order of polynomial. These polynomials are obtained as the solutions to the Chebyshev differential equation. The zero and the first order Chebyshev polynomials are respectively given by

$$Ch_0(x) = 1 \text{ and } Ch_1(x) = x. \quad (8)$$

The higher order polynomials are

$$Ch_2(x) = 2x^2 - 1$$

$$Ch_3(x) = 4x^3 - 3x \quad (9)$$

$$Ch_4(x) = 8x^4 - 8x^2 + 1$$

The recursive formula to generate higher order Chebyshev polynomial is expressed as

$$Ch_{p+1}(x) = 2xCh_p(x) - Ch_{p-1}(x)$$

After getting final expansion for CHFLANN, the weighted sum of the components of the enhanced input pattern is obtained using the below formula

$$\text{Weighted Sum} = \sum_{j=1}^m w_j CHX_j \quad (11)$$

The error obtained by comparing the output with desired output is used to update the weights of the network structure by a weight updating algorithm. In this study the popular Back propagation algorithm is used for training of the network.

Let the cost function at kth instant is.

$$E_k = \frac{1}{2} [d_k - y_k]^2 = \frac{1}{2} e_k^2 \quad (12)$$

The gradient of the cost function is given by

$$\frac{\partial E_k}{\partial W} = e_k \frac{\partial y_k}{\partial W} \quad (13)$$

The update rule for the weight w_{ij} becomes:

$$w_{j,k+1} = w_{j,k} + \alpha e_k (1 - y_k)^2 Ch_j(X) \quad (14)$$

$w_{ij,k+1}$ = weight at $k + 1$ th step

$w_{ij,k}$ = weight at k th step

e_k = error at k th step

y_k = output obtained at k th step

$Ch_j(X)$ = value of j th expanded unit

α = learning rate

5.3. INPUT DATASET

Chronic kidney disease data classification plays an important role in disease classification and analysis. It conveys information of patient. In our study the simulation is done over the PIMA Indian Chronic kidney disease dataset collected from UCI machine learning repository [14]. Description of Clinical data -- Attributes in the Database:

Clinical data

1. Diabetes Mellitus
2. Hypertension
3. Cardiovascular disease
4. Clinical data eGFR (mL/min/
5. Creatinine (mg/dL)
6. Blood urea nitrogen
7. Blood urea nitrogen
8. Age (years)
9. Class variable (0 or 1)

The data set description is given in Table I.

TABLE I DESCRIPTION OF DATASET

INDIAN CHRONIC KIDNEY DISEASE DATABASE	
No of Rows/instances	768
No of attributes plus class level	8 plus 1
No of rows taken for training	512
No. of rows taken for testing	256
No. of instances with class value 0	500
No. of instances with class value 1	268

5.4. PERFORMANCE METRICS

Classification accuracy and F- measure are the two performance metrics, used for comparing the performance of the two classification models. In Chronic kidney disease data classification problem, precision is the fraction of retrieved instances of the dataset which are relevant however recall is the relevant instances that are retrieved. From both the mathematical data we have calculated the f-measure which is the harmonic mean of the precision and recall and accuracy which is the weighted arithmetic mean of precision and recall. The accuracy and f measure for the proposed work is evaluated based on the confusion matrix values. These values are TP (true positives), FN (false negatives), FP (false positives) and TN (true negatives). The input dataset contains class level attribute as 0 or negative (neg) and 1 or positive (pos) finally we have derived confusion matrix values based on the classified result shown in table II.

TABLE II CONFUSION MATRIX

Actual Result	Classified Result	Value
1	1	TP
1	0	FN
0	0	TN
0	1	FP

The accuracy was calculated by using the following formula:

$$\begin{aligned}
 &\text{Specificity} = TP / \text{pos} \\
 &\text{Sensitivity} = TN / \text{neg} \\
 &\text{Precision} = TP / (TP + FP) \\
 &\text{Recall} = TP / FN \\
 &\text{Accuracy} = \text{sensitivity} \times \text{pos} / (\text{pos} + \text{neg}) + \text{specificity} \times \text{neg} / (\text{pos} + \text{neg}) \\
 &\text{F-measure} = (2 \times \text{precision} \times \text{recall}) / (\text{precision} + \text{recall})
 \end{aligned}
 \tag{15}$$

5.5 EXPERIMENTAL RESULTS

A. Description

To test the generalization ability of both the classifiers initially, the dataset is divided into training and testing data set. The training dataset includes 512 instances while the testing dataset includes 256 attributes respectively. The Simulation is carried out for the dataset by passing each instance of training and testing data to LeNN and CFLANN for 10 times considering the 3rd order polynomial equation and different learning rate (alpha) values and threshold values). Then the best accuracy is evaluated for a particular threshold value and learning rate, alpha. Finally the performance achieved with the best network size and threshold value of both the network is compared.

B. Experiment with LeNN

Initially with expansion order 3 and learning rate 0.08 the mean classification accuracy of the training and testing dataset observed from 10 independent runs are compared with three different threshold values. From analysis of result it is observed that, with the threshold value 0.5, the network provides better classification accuracy. The detailed observations is given in the table III. Then keeping the order and threshold value fixed, simulation is done with four different learning rate values as shown in table IV. From the analysis it is observed that the testing accuracy of the network is enhanced by reducing the value of learning rate to 0.02.

TABLE III

PERFORMANCE ANALYSIS (ACCURACY MEASURE) FOR CHRONIC KIDNEY DISEASE DATASET IN LeNN

Order	Learning rate	Threshold	Training accuracy	Testing accuracy
3	0.08	0.5	0.7617	0.8086
			0.7617	0.8047
			0.7695	0.8086
			0.7676	0.8047
	0.08	0.6	0.7559	0.7969
			0.7559	0.7930
			0.7439	0.7930
			0.7559	0.7930
	0.08	0.7	0.7423	0.7742
			0.7431	0.7816
			0.7412	0.7816
			0.7423	0.7812

TABLE IV

PERFORMANCE ANALYSIS (BEST LEARNING VALUE) FOR CHRONIC KIDNEY DISEASE DATASET IN LeNN

Order	Learning rate	Threshold Order	Training accuracy	Testing accuracy
3	0.02	0.5	0.7795	0.8164
	0.04	0.5	0.7695	0.8126
	0.06	0.5	0.7715	0.8086
	0.08	0.5	0.7695	0.8086

B. Experiment with CHFLANN

For the CHFLANN based classifier, initially simulation is done with the same expansion order 3, learning rate 0.08 and Three different threshold values. The simulation results are given in table V. Here also the better classification accuracy is obtained with threshold value 0.5. Then keeping the threshold value and expansion order fixed experiment is done with four

different learning rates as shown in table VI. From the observations, it is found that the testing accuracy of the network is enhanced by reducing the value of learning rate to 0.02.

TABLE V
 PERFORMANCE ANALYSIS (ACCURACY MEASURE) FOR CHRONIC KIDNEY DISEASE DATASET IN CHFLANN

Order	Learning rate	Threshold	Training accuracy	Testing accuracy
3	0.08	0.5	0.7520	0.8086
			0.7559	0.8086
			0.7559	0.8086
			0.7578	0.8047
	0.08	0.6	0.7500	0.8086
			0.7500	0.8047
			0.7500	0.8086
			0.7480	0.8086
	0.08	0.7	0.7500	0.8047
			0.7520	0.8008
			0.7539	0.7969
			0.7520	0.8008

TABLE VI
 PERFORMANCE ANALYSIS (BEST LEARNING VALUE) FOR CHRONIC KIDNEY DISEASE DATASET IN CHFLANN

Order	Learning rate	Threshold	Training accuracy	Testing accuracy
3	0.02	0.5	0.7617	0.8105
	0.04	0.5	0.7578	0.7969
	0.06	0.5	0.7422	0.7969
	0.08	0.5	0.7520	0.8086

C. Result Analysis

Finally the performance of the two classifications models are compared with respect to their accuracy and F-measure value. By choosing two different expansion order as shown in table VI. The analysis clearly shows that LeNN has the better performance compared to CHFLANN with respect to the accuracy and f-measure for the chronic kidney disease dataset. The convergence curve including Mean Square Error (MSE) obtained during training of both the LeNN and CHFLANN classifiers is represented in fig II and fig III respectively.

TABLE VI COMPARATIVE ANALYSIS OF LENN AND CHFLANN FOR PIMA INNDIAN CHRONIC KIDNEY DISEASE DATA CLASSIFICATION

Network	n	TRAINING		TESTING	
		ACCURACY	F-MEASURE	ACCURACY	F-MEASURE
LeNN	2	0.7834	0.9057	0.8308	0.9992
CH-FLANN		0.7776	0.9032	0.8247	0.9799
LeNN	3	0.7795	0.8972	0.8164	0.9897
CH-FLANN		0.7617	0.8858	0.8105	0.9505

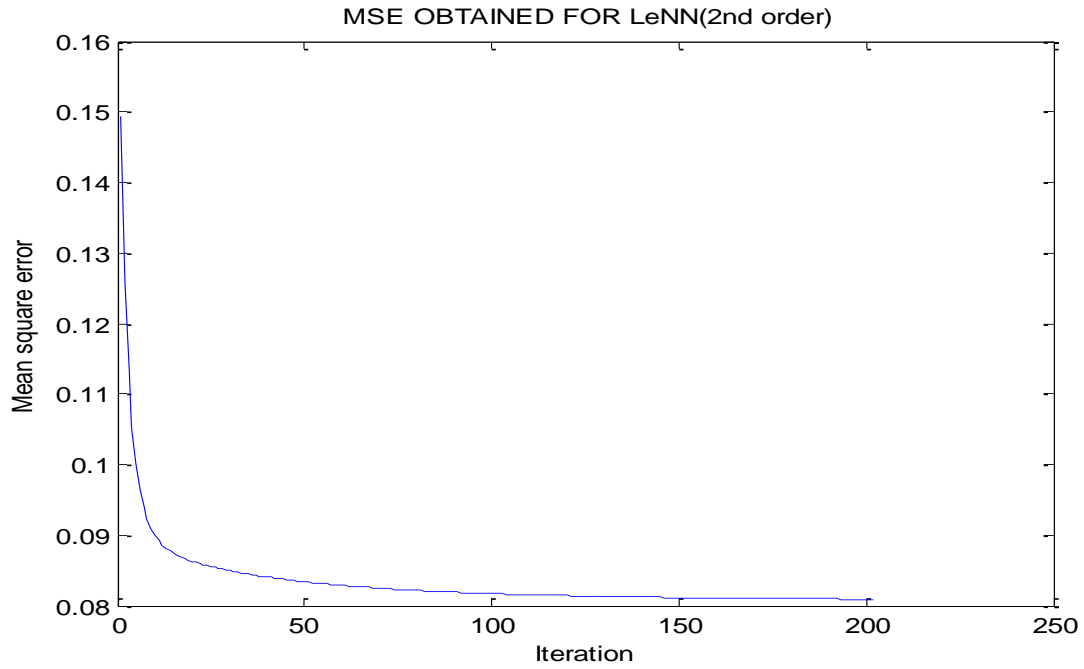


Fig 5.2 .Plot MSE vs. ITERATION for 2nd order LeNN

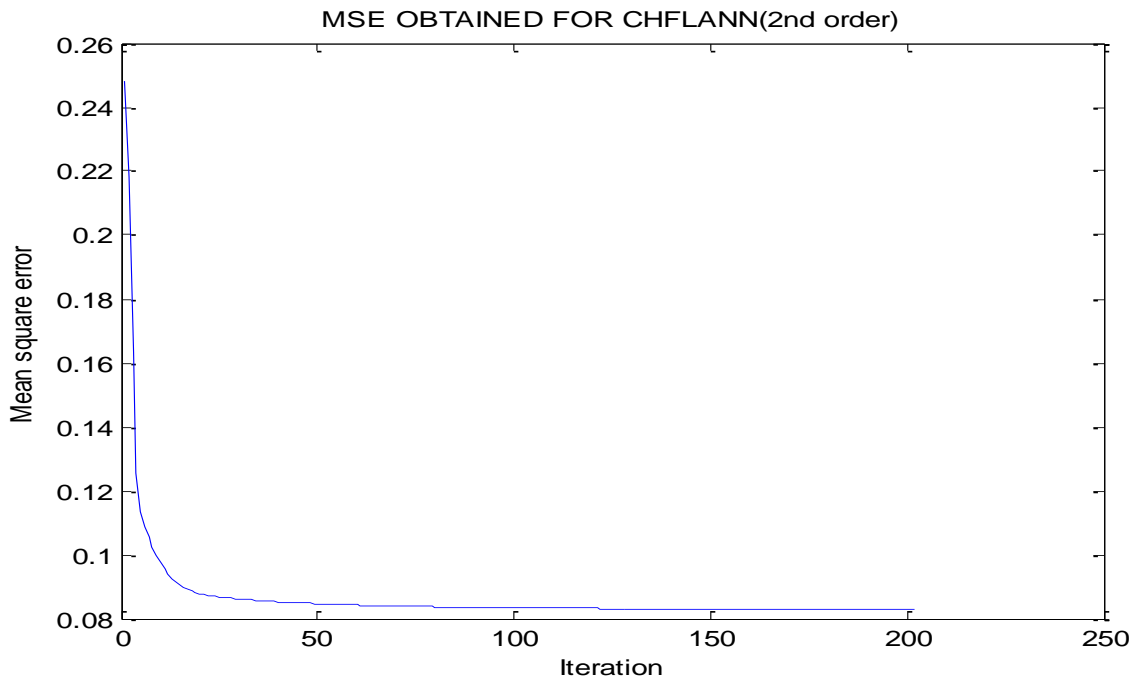


Fig 5.3 .Plot MSE vs. ITERATION For 2nd order CHFLANN

CHAPTER 6

CONCLUSION AND FUTURE WORKS

CONCLUSION:

Now a days, use of datamining techniques in health care problem is increasing rapidly. Using data mining technologies, prediction of the disease will be earlier. This thesis casts the diagonosis of chronic kidney disease disease as a classification problem and tries to develop an efficient classification model using a legendre polynomial based functional link neural network. This study clearly reveals a comparative performance of LeNN and CHFLANN over PIMA Indian Chronic kidney disease dataset for chronic kidney disease data classification.

While analysing the result for both the processes LeNN and CHFLANN, the 2nd order polynomial equation shows large variations in the result comparison as compared to the 3rd order. Hence LeNN shows better performance in accuracy calculation for 2nd order. However the f-measure gives better results for the 3rd order data classification as compared to the 2nd order in the LeNN based approach.

From the result analysis, it is clearly observed LeNN provides better performance over CHFLANN in terms of both the accuracy and f-measure value.

FUTURE WORKS:

Future study of this research will focus on different evolutionary learning algorithms for LENN, which may further increase the classification accuracy of the model. Further the performance of the network will be compared with some other known classifiers like SVM, KNN, Bayesian network etc.

REFERENCES

- [1] Wilkinson, T. J., White, A. E., Nixon, D. G., Gould, D. W., Watson, E. L., & Smith, A. C. (2019). Characterizing skeletal muscle hemoglobin saturation during exercise using near-infrared spectroscopy in chronic kidney disease. *Clinical and experimental nephrology*, 23(1), 32-42.
- [2] Lee, A. K., Katz, R., Ambrosius, W. T., Cheung, A. K., Garimella, P., Gren, L. H. & Ix, J. H. (2019). Abstract MP09: Indices of Kidney Tubular Health Improve Cardiovascular Disease Risk Prediction in Adults With Hypertension and Chronic Kidney Disease in SPRINT. *Circulation* 139 (Suppl_1), AMP09- AMP09.
- [3] Burchfield, A., & Lindahl, K. (2019, March). Direct acting anti-viral medications for hepatitis C: Clinical trials in patients with advanced chronic kidney disease .*seminars in dialysis (Vol. 32, No. 2, pp. 135-140)*.
- [4] Thompson, S., Wiebe, N., Padwal, R. S., Gyenes, G., Headley, S. A., Radhakrishnan, J., & Graham, M. (2019). The effect of exercise on blood pressure in chronic kidney disease: A systematic review and meta-analysis of randomized controlled trials .*Plus one*, 14(2), e0211032.
- [5] Waikar, S. S., Srivastava, A., Palsson, R., Shafi, T., Hsu, C. Y., Sharma, K., & Xie, D. (2019). Association of urinary oxalate excretion with the risk of chronic kidney disease progression .*JAMA internal medicine*.
- [6] Kiran, MD., Gharat, P., Vakharia, M., & Ranganathan, N. (2019). Specific Probiotics for Chronic Kidney Disease: A Review. *The Indian Practitioner*, 72(2), 29-40.
- [7] Rebholz, C. M., Surapaneni, A., Levey, A. S., Sarnak, M. J., Inker, L. A., Apple, L.J. & Grams, M. E. (2019). The Serum Metabolism Identifies Biomarkers of Dietary Acid Load in 2 Studies of Adults with Chronic Kidney Disease .*The Journal of nutrition*.
- [8] Tajti, F., Kuppe, C., Antoranz, A., Ibrahim, M. M., Kim, H., Ceccarelli, F. & Kramann, R. (2019). A functional landscape of chronic kidney disease entities from public transcriptomic data. *BioRxiv* , 265447.
- [9] Schrauben, S. J., Jepson, C., Hsu, J. Y., Wilson, F. P., Zhang, X., Lash, J. P., ... & Kao, P. (2019). Insulin resistance and chronic kidney disease progression, cardiovascular events, and death: findings from the chronic renal insufficiency cohort study .*BMC nephrology*, 20(1), 60.
- [10] J. L. Breault, C. R. Goodall, P. J. Fos , "Data mining a chronic kidney disease data warehouse", *Artificial Intelligence in Medicine* vol.26, pp.37-54, 2002.
- [11] E. G. Yildirim, A. Karahoca, T. Uar, "Dosage planning for chronic kidney disease patients using data mining methods", *Procedia Computer Science*, vol.3, pp.1374-1380, 2011.
- [12] A. A. Aljumah, M. G. Ahamad, M. K. Siddiqui, "Application of data mining: Chronic kidney disease health care in young and old patients", *Journal of King Saud University-Computer and Information Sciences*, vol.25, pp.127-136, 2013.
- [13] S. K. Nanda, D. P. Tripathy, S.S. Mahapatra, "Application of Legendre Neural Network for Air Quality Prediction", the 5th PSU-UNS International Conference on Engineering and Technology, pp. 267-272, 2011.
- [14] N. V. George and G. Panda, "A Reduced Complexity Adaptive Legendre Neural Network For Nonlinear Active Noise Control", 19th International Conference on Systems, Signals and Image Processing (IWSSIP), pp. 560-563, 2012.
- [15] N. Rodriguez, "Multiscale Legendre Neural Network for Monthly Anchovy Catches Forecasting", Third International Symposium on Intelligent Information Technology Application, pp. 598-601, 2009.
- [16] K. K. Das, J. K. Satapathy, "Legendre Neural Network for Nonlinear Active Noise Cancellation with Nonlinear Secondary Path", International Conference on Multimedia, Signal Processing and Communication Technologies, pp. 40-43, 2011.
- [17] J. C. Patra, P. K. Meher, G. Chakraborty, "Non linear channel equalization for wireless communication systems using Legendre neural networks" , *Signal Processing*, vol.89, pp.2251-2262, 2009.
- [18] F. Liu, J. Wang, "Fluctuation predictions of stock market index by Legendre neural network with random time strength function", *Neuro computing*, vol.83, pp.12-21, 2012.
- [19] M. K. Mishra, R. Dash, "A comparative study of Chebyshev Functional Link Artificial Neural Network, Multi Layer Perceptron and Decision Tree for Credit Card Fraud Detection", International Conference on Information Technology, 2014.
- [20] S. K. Mishra, G. Panda, S. Meher, "Chebyshev Functional Link Artificial Neural Network for Denoising of Image Corrupted by Salt and Pepper" , *International journal of recent trends in Engineering*, vol.1 (1), pp.413-417, 2009.
- [21] J.C. Patra, N.C. Thanh, P. K. Meher, "Computationally Efficient FLANN-Based Intelligent Stock Price Prediction System" ,*Proceedings of International joint Conference on Neural Networks*, pp.2431-2437, 2009.
- [22] B. B. Misra, S. Dehuri, "Functional Link Artificial Neural Network for Classification Task in Data Mining" , *Journal of Computer Science*, vol.3 (12), pp.948-955, 2007.
- [23] <https://archive.ics.uci.edu/ml/machine-learning-databases/statlog/>
- [24] Ngai E.W.T., Hu Yong, Wong Y.H., Chen Yijun, Sun Xin, "A Classification Framework and an Academic Review of Literature", *Decision support system* vol.50, pp.559-569, 2011.
- [25] Tej paul bhatla, Vikram prabhau, Amit dua, "Understanding Credit Card Fraud", *Cards business review*, pp.1-3, 2003.
- [26] Sahin Yusuf, Bulkan Serol, Duman Ekrem, " A Cost-Sensitive Decision Tree Approach for Fraud Detection", *Expert Systems with Applications*, vol.40, pp.5916-5923, 2013.
- [27] Duman Ekrem, Ozcelic Hamdi M., " Detecting Credit Card Fraud by Genetic Algorithm and Scatter Search", *Expert Systems with Applications*, vol.38, pp.13057-13062, 2011.
- [28] Sahin Y., Duman E., "Detecting Credit Card Fraud by Decision Tree and Support Vector Machine", *Proceeding of the International multi conference of engineers and computer scientist*, pp.315-319, 2011.
- [29] Bhattacharyya Siddharth, Jha Sanjeev, Tharakunnel Kurian, Westland Christopher J., "Data Mining for Credit Card Fraud: A Comparative Study", *Decision support system*, vol.50, pp.602-613, 2011.
- [30] Yan-Li Zhu, Jia Zhang, "Research on Data Preprocessing In Credit Card Consuming Behavior Mining", *Energy Procedia*, vol.17, pp.638- 643, 2012.