PREDICTIVE MODELS FOR PARAMETERS OF SEISMIC SIGNALS BASED ON ADVANCED MACHINE LEARNING

Ph.D. THESIS

by

SONIA THOMAS

DEPARTMENT OF EARTHQUAKE ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE – 247 667 (INDIA)
FEBRUARY, 2016

PREDICTIVE MODELS FOR PARAMETERS OF SEISMIC SIGNALS BASED ON ADVANCED MACHINE LEARNING

A THESIS

Submitted in partial fulfilment of the requirements for the award of the degree of

DOCTOR OF PHILOSOPHY

in

EARTHQUAKE ENGINEERING

by

SONIA THOMAS

DEPARTMENT OF EARTHQUAKE ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE – 247 667 (INDIA)
FEBRUARY, 2016

INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled "**PREDICTIVE MODELS FOR PARAMETERS OF SEISMIC SIGNALS BASED ON ADVANCED MACHINE LEARNING**" in partial fulfilment of the requirements for the award of the Degree of Doctor of Philosophy and submitted in the Department of Earthquake Engineering of the Indian Institute of Technology Roorkee, Roorkee is an authentic record of my own work carried out during a period from July,2011 to February,2016 under the supervision of Dr. G.N.Pillai, Professor, Department of Electrical Engineering and Dr.Kirat Pal, Professor, Department of Earthquake Engineering, Indian Institute of Technology Roorkee, Roorkee, India.

The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other Institute.

(**SONIA THOMAS**)

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

(G. N. PILLAI)
Supervisor

(KIRAT PAL)
Supervisor

Date: _____

ABSTRACT

In this thesis, advanced machine learning algorithms are used to develop predictive models for forecasting ground motion parameters. The machine learning algorithms used are extreme learning machines (ELM), support vector regression (SVR) and its three variations, namely ε-SVR, ν-SVR and Ls-SVR, decision trees and hybrid algorithm ANFIS (adaptive neuro fuzzy inference system). In this thesis, a novel neuro fuzzy algorithm, RANFIS (randomized ANFIS) is also proposed for forecasting ground motion parameters. This advanced learning machine integrates the explicit knowledge of the fuzzy systems with the learning capabilities of neural networks, as in the case of conventional adaptive neuro fuzzy inference system (ANFIS). In RANFIS, to accelerate the learning speed without compromising the generalization capability, the fuzzy layer parameters are not tuned.

The three time domain ground motion parameters which are predicted by the developed predictive models are peak ground acceleration (PGA), peak ground velocity (PGV) and peak ground displacement (PGD). Each ground motion parameter is related to mainly to four seismic parameters, namely earthquake magnitude, faulting mechanism, source to site distance and average soil shear wave velocity. The model is developed using real earthquake records obtained from the database released by PEER (Pacific Earthquake Engineering Research Center)

Conventionally, the ground motion parameters are estimated using strong ground motion prediction equations which are also known as attenuation equations. Ground motion prediction equations (GMPEs) are equations that related the ground motion parameter PGA, PGV, PGD to independent parameters like earthquake magnitude, source to site distance and site conditions. They are developed using the traditional regression analysis method. The development of GMPEs involves highly complex computation because of the high nonlinearity and inhomogeneous dependencies among the parameters. The regression analysis is applied for the computation after reducing the complexities by including assumptions. Incorporating the simplified assumptions into modelling leads to very large errors. Thus, there is a huge need for the modelling of ground motion parameters using newer techniques so as to reduce the existing complexities. These overheads are minimized by using advanced learning machines.

The predictive models for forecasting ground motion parameter, developed using advanced learning machines have many advantages. For modelling using machine learning, it is not required to assume linear dependencies among the variables. Thus, there are no assumptions made and no irrelevant coefficients are required. This makes the predictive models developed using advanced machine learning computationally faster. Moreover, using the advanced learning machines, efficient predictive models with higher precision and lesser error measure is obtained. In this study, all the developed prediction models based on advanced machine learning, are compared to the existing GMPEs as well as the existing benchmark models. The existing GMPE models are Ambraseys et al model [6], Campbell and Bozorgnia model [29] and Smit et al model [142]. The existing benchmark models are ANN/SA model by Alavi and Gandomi, GP/OLS model by Gandomi et al, MEP model by Alavi et al [5] and GP/SA model by Mohammadnejad et al [100]. The quantitative and the qualitative analysis of all the proposed prediction models based on advanced machine learning algorithm shows that the developed prediction models have a good prediction accuracy for the forecasting of ground motion parameter.

The significance of the proposed work in this thesis is the application of advance machine learning for faster and easy prediction of the ground motion parameters. The ground motion parameters are the most relevant criteria required for designing any earthquake resistant infrastructure. With growing urbanization, there is tremendous increase in the population density in earthquake prone areas, which in turn is increasing the demand for earthquake resistant structure.

All the developed models are tested on the real earthquake data. The database used for modelling is the database known as NGA WEST 1 compiled and systematized by Pacific Earthquake Engineering Research Center (PEER) in 2003 as a part of a project named PEER-NGA project. The database file is termed as NGA flatfile V 7.3. The predictive models are trained on 2252 earthquake records and tested on 563 earthquake records. To further validate the efficacy of the proposed models, the models are tested on another set of 140 earthquake records.

In this study, the different types of learning methods used are namely neural network learning, kernel method learning, hybrid models and decision tree learning. The hybrid models used in this study are neuro fuzzy techniques which combine the fuzzy logic and neural networks.

In this study, six different prediction models are proposed. The ground motion parameter prediction model developed based on neural network learning are ANN model, and ELM (extreme learning machines) model. The ground motion parameter prediction model developed based on kernel method learning are ε-support vector regression model, ν-support vector regression and Ls-SVR (least square support vector regression) model. The ground motion parameter prediction model developed based on hybrid models are ANFIS (adaptive neuro fuzzy inference system) model and the novel neuro fuzzy technique, RANFIS (randomized ANFIS) model. The ground motion parameter prediction model developed based decision tree learning is a regression tree model.

In this study, a further comparative study of all the developed models is done to deduce the best prediction model. Furthermore a comparative study of the learning effectiveness of each algorithm is done in terms of measure of 'overfitness'. The overfitness measure is a comparison of the training error with the testing error. This comparative analysis further highlights the advantages and drawbacks of each advanced machine learning algorithm.

In this study all the comparisons and conclusions are well validated, as the models are based on real earthquake data, rather than the synthetic data. Furthermore, it is observed that the proposed novel neuro fuzzy technique RANFIS proves to be promising prediction algorithm for forecasting ground motion parameters and hence could be applied to other prediction problem in various domains.

ACKNOWLEDGMENT

I am highly obligated to my both guides, Dr. G.N. Pillai and Dr. Kirat Pal for supervising and providing me the opportunity to work in this emerging field of research. Their guidance and support throughout the thesis inspired me to carry out this work. The discussions with them and their enthusiasm kept me motivated. Their ideas have been of great help in exploring the area which would otherwise have been impossible. I express my sincere, heartfelt gratitude to them.

I would also like to express my gratitude to Dr. M.L. Sharma**,** Professor and Head of the department of earthquake engineering for providing all facilities and the opportunity to work in this field. I am also thankful to Dr. J.P. Narayan, DRC chairman, for his suggestions.

Last but not the least I would like to thank my mother, my brother and friends for all the support they provided, without which this thesis would have come to a standstill.

Thank you all.

Date: **Sonia Thomas**

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF MATHEMATICAL NOTATIONS

a_j	Premise parameter, deciding the position of bell shape membership function
$avg(obs_i)$	Average of observed value
$avg(pred_i)$	Average of predicted value
b_j	Premise parameter deciding the position of bell membership function
c_j	Premise parameter deciding the shape of the bell membership function
dcc	Distance between two consecutive centers of member function
D	Distance
F	Faulting mechanism
$\mathbf{f_1}$	Rule 1 of the Sugeno ANFIS network
$\mathbf{f_2}$	Rule 2 of the Sugeno ANFIS network
g(x)	Activation function of the node
H	Premise layer output matrix
k	Number of data records
m	Number of membership function
M	Earthquake Magnitude
$\mu(\boldsymbol{x})$	**i**s the membership grade for input **x**
$\mu\ (\boldsymbol{y})$	is the membership grade for input **y**.
n	Dimension of input data
N	Training data pairs
obs_i	Observed value

$pred_i$	predicted value
P	upper limit of normalized value
p_i	consequent parameter,
Q	lower limit of normalized value
q_i	consequent parameter,
R	correlation coefficient
r_i	consequent parameter
S	sum of squared errors
$\acute{\mathbf{S}}$	spilting criteria
T	Target matrix
V	Velocity
$\hat{\mathbf{y}}$	mean of variables
Y	Ground motion parameter
Z	Input variable
Z_{max}	Maximum value of input variable
Z_{min}	Minimum value of input variable
Z_n	Normalized value of input variable

LIST OF SYMBOLS

μ	Membership function
β	Linear network parameter matrix
δ	Threshold value
ε	Permissible error
ν	Parameter having value in range (0,1]
$\tilde{N}$	Number of hidden nodes
A_i	Linguistic variable
B_i	Linguistic variable
b_i	Threshold value of i^{th} hidden node
C	Tradeoff value
g	Acceleration (measuring unit of Peak Ground Acceleration)
M_0	Moment magnitude (measuring unit of earthquake magnitude)
o_j	Output of the linear layer
x_i	Input value of the dataset
X_i	i dimensional input data
x	Crisp set representing linguistic variable A_i
y	Crisp set representing linguistic variable B_i
t_i	Target value of the dataset
w_i	Weight matrix
$\overline{w_i}$	Normalized weight vector

LIST OF ACRONYMS

CI	Artificial Intelligence
AI	Artificial Intelligence
ANFIS	Adaptive neuro-fuzzy inference system
ANN	Artificial neural network
ANN/SA	Artificial neural network coupled with simulated annealing
BP	Backpropogation
CFG	Context Free Grammar
ELM	Extreme learning machines
ELM-RBF	Extreme learning machines- Radial bias function
FL	Functional link
GA	Genetic Algorithm
GDR	Generalized delta rule
GEP	Genetic expression programming
GP/OLS	Genetic programming coupled with orthogonal least squares
GMPE	Ground motion prediction equation
HLA	Hybrid learning algorithm
LDA	Linear discriminant analysis
LS-SVR	Least square support vector regression
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MSE	Mean Squared Error

NGA	New generation attenuation
NGA WEST1	Database released by PEER in 2006
NGA WEST2	Database released by PEER in 2012
PCA	Principal component analysis
PEER	Pacific Earthquake Engineering Research Center
PGA	Peak ground acceleration
PGD	Peak ground displacement
PGV	Peak ground velocity
PSO	Particle swarm optimzation
RANFIS	Randomized adaptive neuro-fuzzy inference system
RBF	Radial bias function
RMSE	Root Mean Squared Error
RVM	Relevance vectore machine
SLFN	Single layer feedforward network
SVM	Support vector machines
SVR	Support vector regression

Chapter 1

INTRODUCTION

1.1 OVERVIEW

In this thesis, advanced machine learning algorithms are used to develop predictive models for forecasting ground motion parameters associated with seismic signals. The ground motion parameter predicted are peak ground acceleration (PGA), peak ground velocity (PGV) and peak ground displacement (PGD). These parameters are in time domain.

The term seismic denotes the shaking or vibration of the earth `s crust due to an earthquake or to an artificial explosion. Hence the signals generated during an seismic activity is termed as seismic signals. The seismic signal processing is a subfield of the digital signal processing (DSP). The processing of seismic signal mainly focuses on processing of seismic data by removing noise and enhancing the weaker signals to locate the seismic events on the subsurface of the earth`s crust. The information obtained from the seismic signals assists the geologist for better understanding and interpretation of the structure of the earth`s subsurface. The seismic signals are mainly used in broad fields such as microseismic data processing, reservoir characterization and seismic data compression. The processing of seismic signals in the field of microseismic data processing is for the analysis and prediction of earthquakes. The seismic signal processing relating to reservoir characterization mainly deals with the exploration of oil fields.

The signals generated during an earthquake are recorded by an instrument named 'seismograph' and the records are termed as ground motion records. During an earthquake, the rupture of the earth`s crust begins at point, termed as the focus. A seismogram is the recording of the seismic signal as a function of time, at the particular recording station. It is recorded in 3 co-ordinate space (x, y and z) with x-and y- planes parallel to earth`s surface and z-plane perpendicular to the earth`s surface. Thus, a seismogram records the displacement of ground or

earth`s crust during an earthquake. The seismograph can record the shaking of the earth`s crust, due to an earthquake or due to a heavy explosion. A seismograph can record any range of disturbance ranging from, very small waves, such as due to a heavy wind and strong ground motion due to a powerful earthquake. The recording of the acceleration of the earth`s crust or ground during an earthquake is called accelerogram. The recording of the velocity of the earth`s crust or ground during an earthquake is called velocigram. The Fig. 1.1 shows a sample seismic recording.

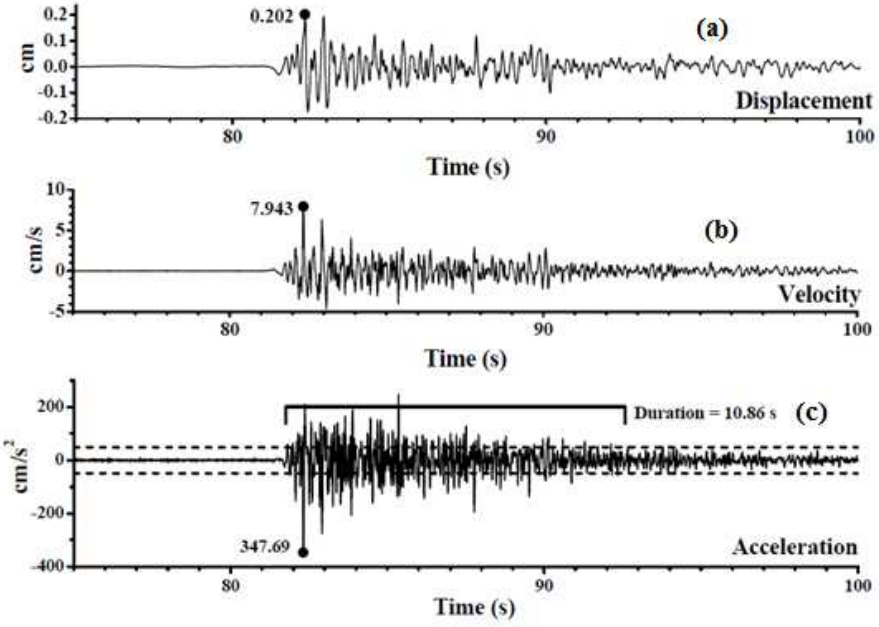

Fig. 1.1: A sample seismic recording (a) seismogram (b) velocigram (c) accelerogram

The few basic terminologies associated with an earthquake process are as follows. The pictorial representation of the terminologies is shown in Fig. 1.2 for better understanding.

i. Focus: The point at which earthquake originates within the earth. It is also called hypocenter.

ii. Epicenter: The point on the earth`s surface which is directly above the focus.

iii. Focal depth: It is the depth of earthquake focus, or in other words, it is the distance between the focus and the epicenter.

iv. Epicentral distance: It is the distance of site from the epicentre of an earthquake.

v. Hypocentral distance: It is the distance of site from the focus of an earthquake.

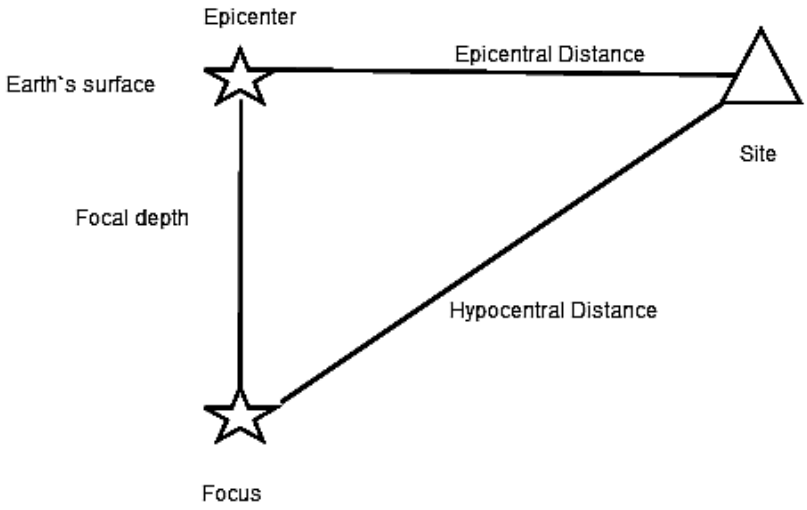

Fig. 1.2: Pictorial representation of earthquake process

vi. Earthquake magnitude: Earthquake magnitude is a number that characterizes the relative size of an earthquake.

vii. Peak ground acceleration (PGA) : it is the maximum value, (positive or negative) of the ground acceleration that appears in the accelerogram.

viii. Peak ground velocity (PGV) : It is the maximum value of the ground velocity (positive or negative) that appears in the velocigram.

ix. Peak ground displacement (PGD): It is the maximum value of the ground displacement (positive or negative) that appears in the seismogram.

The earthquake originates at focus, within the earth generating seismic waves. A seismic wave is an elastic wave generated by the impulse of an earthquake. The seismic signals travel in all directions from the focus and reach the site, where it is recorded with the help of seismographs. The path by which the wave would travel would dependant on the local geology and soil conditions. During an earthquake, damage is usually maximum at the epicentral region, where strong motion is maximum. Ground motion records from the epicentral region can be of great help in understanding the earthquake process at the source as effects of transmission paths are minimal. In the epicentral region near the source, accelerograms or strong motion records may be used to provide useful insight into the earthquake process. Analysis and interpretation of observed and simulated strong motion records holds promise for an enhanced understanding of the earthquake process

Natural calamities, which causes immense loss to the society cannot be prevented or controlled. The destruction and loss caused by the calamities are immense. Although the loss of lives during calamities can never be predicted or prevented, an attempt could be done to reduce the casualties occurring during any natural calamity. Among the various natural calamities, earthquakes are considered to be more destructive in nature. The main reason for considering earthquake as most devastating natural calamity is the human and economic losses occurring during an earthquake. The human loss is mainly due to destruction of the man made structures such as bridges, buildings etc. With the advancement in science and technology, an attempt is being made to construct infrastructures that are tolerant to seismic activity. Hence developed the concept of designing earthquake resistant structures. The most crucial parameters (Sehhati et al [131], Giacinto [46]), considered for designing earthquake resistant infrastructure are the peak ground acceleration (PGA), the peak ground velocity (PGV) and the peak ground displacement (PGD). These parameters are collectively termed as ground motion parameters.

Conventionally, the ground motion parameters are estimated using strong ground motion predictive equations. Ground motion prediction equations (GMPEs) are equations that related the ground motion parameter PGA, PGV, PGD to independent parameters like earthquake magnitude, source to site distance and site conditions. They are developed using the traditional regression analysis method. The major drawback of using conventional method is that applying

regression analysis for the development of attenuation relationships is complex because of the high nonlinearity and inhomogeneity among the parameters. Hence, there is a huge need for the modelling of ground motion parameters using newer techniques so as to reduce the existing complexities.

1.2 ADVANCED LEARNING MACHINES

Artificial Intelligence (AI) is the branch of computer science that deals with computer programs that can solve a class of problems such as pattern recognition, decision making and learning. The term 'Artificial Intelligence', was first coined by John McCarthy in 1955. The term is defined as the discipline of developing computational agents that can act intelligently. An agent is anything that acts when subjected to an environment or situation. The agent is said to have acted intelligently if agent`s actions are in accordance with the goal required by the environment, or if the agent is flexible when subjected to changes in its environment or if the agent willingly learns or adapts itself to the environment or if the agent is capable of taking appropriate decisions when subjected to choices or options.

An agent is termed as a computational agent when the actions or decisions made by the agent could be expressed in terms of computation. Hence the decision or action is furcated into a primitive operation that could be implemented on a physical device. For an instance, if the agent is a computer, the primitive operation will be carried out in hardware. There are some agents on the contrary, that cannot be computational, say rain or wind causing erosion to landscape. Hence, it is an arguable declaration if all intelligent agents are computational. The main goal of artificial intelligence is to design intelligent agents. Hence it works on the principle of understanding the fundamentals that make intelligent behavior possible. It is done by formulating a hypothesis that could make an agent act intelligently, after analyzing the agent and then proving the hypothesis by testing.

Computational intelligence (CI) is the study of nature inspired computational methods that could be applied to real world complex problems where traditional mathematical modelling fail. Computational intelligence is applied to problems such as a complex process too complex for mathematical reasoning, or the process consists of lots of uncertainties or

when the process is stochastic in nature. The CI uses the combination of 5 complementary techniques as fuzzy logic, artificial neural networks, evolutionary computing, learning theory and the probabilistic methods.

Although computational intelligence and artificial intelligence, both work on achieving long term goal of acquiring intelligence so that a machine could perform a task intellectually, there exist a key difference between computational intelligence and artificial intelligence. Computational intelligence is a subset of artificial intelligence. The artificial machine intelligence works on hard computing techniques and the computational machine intelligence works on soft computing techniques. The concept of binary logic, consisting of boolean 0 and 1, based on which modern computer works, is followed in hard computing techniques. The drawback of this concept is that not all problems could be expressed in absolute terms of 0 and 1. Hence, in this scenario, soft computing techniques come into picture. The fuzzy systems which helps in better representation of the problem, by expressing it in a range of value ranging from 0 to 1.

Machine learning (Bishop [24]) is an emerging field, that evolved from the theory of computational learning and pattern recognition, in the domain of artificial intelligence. Machine learning was defined by Arthur Samuel in 1959 as that field of study that gives machines or computers, ability to learn without being explicitly trained or programmed. Thus, it deals with the development and construction of algorithms that can learn from the data. The algorithms ,thus developed operate by building a model, from the sample data provided for training the model and hence making predictions. Machine learning has its application in a wide range of fields such as computing task where explicit designing of algorithms is not feasible, in the discipline of computational statistics, which focuses on computer based predictions etc. Machine learning and pattern recognition are facets of the same field. Based on the type of learning machine learning could be broadly classified as supervised learning, unsupervised learning and reinforcement learning. Machine learning could also be categorized as classification, regression, clustering and dimensionality reduction based on the desired output of the machine. The supervised learning algorithm has the ability to analyze the training data and to draw conclusions as either classification for discrete data or as a regression for continuous data. The generalization is the ability of any learning method to predict the exact output for every valid input.

The goal of machine learning, shifted from focusing to achieve intelligence as in the field of AI, to solving practical problems using methods from statistics and probability theory, rather than using symbolic approaches used in AI. Although machine learning employs methods used in data mining, there exists a difference. The machine learning concept focuses on learning a machine for prediction based on some properties of the training data, whereas the data mining aims to discover unknown properties of the data. The various approaches in machine learning are decision tree learning, association rule learning, artificial neural network learning, inductive logic programming, support vector machines (SVM), clustering, bayesian network, reinforcement learning, representation learning, symmetry learning, sparse dictionary learning, genetic algorithms etc. Machine learning methods are also referred as predictive modelling. Machine learning is gaining popularity and is applied in almost all fields of pattern recognition problems.

The topics relating to advanced machine learning include the application of kernel methods, graphical models, tensor approach, semi-supervised and active learning, boosting, bagging etc, to data analysis. For modelling of sophisticated and advanced machine learning models for complex real world problem, the relevant paradigms are kernel methods and graphical models.

The basic principle of kernel methods is to create a framework, so that the approaches designed for linear relations and patterns could be extended to nonlinear cases. In the kernel method approach, the data are first mapped onto a high dimensional feature space, and then various algorithms are applied to this feature space to find patterns or relations in the data. The mapping is done using functions named kernel functions. Since the mapping could be done without any constraints such like linear mapping, the relations developed in the data could be linear or nonlinear. This is the major advantage of using kernel methods. The few algorithms operating with kernels are Gaussian processes, principal components analysis (PCA), Fisher's linear discriminant analysis (LDA), spectral clustering, support vector machine (SVM), relevance vector machine (RVM), etc. The kernel functions are selected based on the problem. The commonly used kernel functions are Linear Kernel, Gaussian Kernel, Polynomial Kernel, Bessel Kernel, Bayesian Kernel, Circular Kernel, Wavelet Kernel etc. The popular kernel methods based approach is support vector machines (SVM) and relevance vector machine (RVM).

Graphical models work on developing a framework for providing solutions to complex real world problems consisting of huge datasets with a very large number of features or attributes defining the data. Graphical models are generally applied to find intelligent conclusions using local information or knowledge for such datasets. Hence a graphical model is nothing but a probabilistic model in which the structure representing the conditional dependencies between attributes or variables could be represented by a graph.

The various algorithms based on graphical models help in drawing intelligent conclusions, learning and decision making for a variety of problems such as statistics, artificial intelligence, natural language processing, computational biology, etc. The further advances in machine learning are the development of hybrid models such as ANFIS, which combines two approaches such as fuzzy logic and ANN. Furthermore, attempts are being done to improve the existing approaches. One such attempt is the development of extreme learning machines (ELM). The extreme learning machine is a generalized single layer feedforward network (SLFN) in which the hidden layer parameters are not tuned.

In this study, the problem statement is a prediction problem. The aim of the study is to develop predictive models based on advanced machine learning. The approaches used in this study are neural network base learning, kernel methods based learning and tree based learning. The advance machine learning algorithms used in this study for the development of predictive models for forecasting ground motion parameters are ELM, SVM, decision tree, and hybrid architecture ANFIS. The three time domain ground motion parameters which are predicted by the model are peak ground acceleration (PGA), peak ground velocity (PGV) and peak ground displacement (PGD). The model is developed using the database released by PEER (Pacific Earthquake Engineering Research Center) [111]. Each ground motion parameter is related to mainly to four seismic parameters, namely earthquake magnitude, faulting mechanism, source to site distance and average soil shear wave velocity.

Moreover, in this study, a novel neuro fuzzy technique, RANFIS is also proposed for forecasting ground motion parameters. The proposed RANFIS model is an improvement of the conventional ANFIS model. This advanced learning machine integrates the explicit knowledge of the fuzzy systems with the learning capabilities of neural networks, as in the case

of conventional adaptive neuro-fuzzy inference system (ANFIS). In RANFIS, to accelerate the learning speed without compromising the generalization capability, the fuzzy layer parameters are not tuned.

The experimental results obtained from all the proposed predictive models validate the improved performance of the developed predictive models for forecasting ground motion parameter, with lesser computation time compared to prior studies.

1.3 RESEARCH GAP

The ground motion parameters are estimated conventionally, using strong ground motion predictive equations, which are developed using traditional regression analysis. The application of the regression method, in the development of the ground motion prediction equations (GMPEs) equations can produce some problems due to inhomogeneities in terms of independent parameters.

Ground motion prediction equations are equations that relate the ground motion parameter PGA, PGV, PGD to independent parameters such as earthquake magnitude, source to site distance and site conditions. These parameters are highly dependant on each other. Hence the modelling of GMPEs is highly complex due to multivariable dependencies. For applying regression analysis for modelling the GMPEs, the complexities are reduced by assuming linear dependencies among the variables. Hence a number of assumptions are included in the modelling. Thus the inclusion of assumptions increases higher measure of error percentages in the predictive equation because, for a highly non-linear form of the regression, a small change in one coefficient strongly affects another coefficient's value. Thus, there is a huge need for special techniques to be employed for the modelling of ground motion parameters. The advanced learning machines can overcome these drawbacks. The predictive model developed using advanced learning machines has many advantages. For modelling using machine learning, it is not required to assume linear dependencies among the variables. Thus, there are no assumptions made and no irrelevant coefficients are included.

The modelling is for function $Y \rightarrow f(earthquake\ magnitude, distance, site\ condition)$, where Y represents the ground motion parameter. Y is PGA, PGV and PGD. Hence, in the development of predictive models using machine learning, the interdependencies of the parameters are considered, as there is no constrain that the parameters should be linearly dependent. Another advantage of using advanced machine learning is that the predictive models developed using machine learning gives better prediction accuracy in lesser computational time compared to GMPEs. In the GMPEs, since linear modelling is considered, each parameter used for modelling such as earthquake magnitude, distance and site conditions includes a very large number of coefficients. Thus the computational time for solving an GMPE equation would be very high.

As an example, a GMPE, named 'Campbell and Bozorgnia' [29] is shown in appendix A. The 'Campbell and Bozorgnia' ground motion prediction equation is given along with the table showing the values of the coefficients. The equation is modelled as a function of six variables and consists of 16 coefficients. Hence, for solving the equation, the corresponding values have to be substituted. Thus, it is clearly observed that solving a GMPE is cumbersome, time consuming and involves a lot of computational complexity. Moreover the results obtained by solving a GMPE consist of a very high percentage of error. It is observed that 'Campbell and Bozorgnia' ground motion prediction equation give an error measure of 0.93, measured as mean absolute percentage error (MAPE) [4,5,45,100] on an earthquake database of PEER [111]. Hence it is clearly observed that the GMPE models do not give accurate results.

Furthermore, from the literature survey as detailed in chapter 2 of the thesis, it is observed that there is limited application of soft computing techniques for prediction of ground motion parameters. There exist only 4 models based on soft computing techniques, as shown in Table 3.1. All these existing models are based on neural network learning. The neural network based learning such as artificial neural networks have many drawbacks such as poor generalization ability, overfitting of data and the algorithm getting stuck at the local minima. Hence it is observed that more accurate and efficient prediction models could be developed.

The advanced learning machines have proved its efficiency in many pattern recognition problems. The application of advanced machine learning for prediction of ground motion parameter should also be possible. In this study, this presupposition is analyzed and validated.

1.4 OBJECTIVE

In this thesis, advanced machine learning algorithms are used to develop predictive models for forecasting ground motion parameters. The prediction models are developed on basically 4 types of learning, namely neural network learning, kernel based methods learning, hybrid models and decision tree learning. The three ground motion parameters in time domain which are predicted are peak ground acceleration (PGA), peak ground velocity (PGV) and peak ground displacement (PGD). Each ground motion parameter is related to four seismic parameters, namely earthquake magnitude, source to site distance, average soil shear wave velocity and faulting mechanism. The model is developed using real earthquake records obtained from the database released by PEER (Pacific Earthquake Engineering Research Center) [111].

The machine learning algorithms used are extreme learning machines (ELM), support vector regression (SVR) and its three variations, namely ε-SVR, ν-SVR and LS-SVR, hybrid algorithm ANFIS (adaptive neuro fuzzy inference system) and regression tree learning. A single layer feedforward network (SLFN) is also used for modelling. The ground motion prediction model based on SLFN is developed so as to compare it with the other developed prediction models based on advanced machine learning. Thus, this comparison helps in highlighting the advantages of advanced learning machines over artificial neural networks.

In this thesis, a novel neuro-fuzzy learning machine called randomized adaptive neuro-fuzzy inference system (RANFIS) is also proposed for prediction of ground motion parameters. The RANFIS algorithm integrates the explicit knowledge of the fuzzy systems with the learning capabilities of neural networks, as in the conventional ANFIS system, but with the difference that, the fuzzy layer parameters in RANFIS are not tuned. This improvement in the architecture of ANFIS structure helps to accelerate the learning speed without compromising the generalization capability.

In this study, the advanced machine learning algorithms are used to develop the predictive model in an attempt to develop efficient ground motion prediction model compared to GMPEs. The major drawback of GMPE models is the higher measure of error in the results. The quantitative and the qualitative analysis of all the proposed prediction models based on advanced machine learning algorithm, shows that the developed prediction models have a good prediction accuracy for the forecasting of ground motion parameter. The analysis also shows that error measure in the predictive models based on advanced machine learning algorithm is very less compared to the existing predictive models as shown in Table 3.1 and GMPEs.

The study of the ground motion parameter is highly significant in the field of earthquake engineering as ground motion parameter is a vital parameter for constructing earthquake resistant structures. Hence the development of high precision predictive models for forecasting ground motion parameter prediction will be a significant contribution to the domain.

1.5 AUTHORS CONTRIBUTIONS

The following are the significant contributions drawn on the results obtained in this study:

a. In this work, six different prediction models are proposed using advanced machine learning technique for the prediction of ground motion parameters. The ground motion parameter prediction model is developed based on neural network learning (ANN, ELM), kernel method learning (ε-support vector regression, ν-support vector regression and LS-SVR (least square support vector regression)), hybrid models (ANFIS, RANFIS) and decision tree learning. The overall performance of all the developed model is better compared to the existing benchmark models in the same database.

b. All the proposed models are developed and tested on the real earthquake data [23,111]. The database used for modelling is the database compiled and systematized by Pacific Earthquake Engineering Research Center (PEER) in 2003 as a part of a project named PEER-NGA project. The database file is termed as NGA flatfile V 7.3.

c. All the developed prediction models are tested on 563 earthquake data records [111] and further tested, as an external validation, on another set of 140 earthquake data records [23]. Hence the predictive model is tested for 703 events. This validates the precision of the proposed model.

d. All the proposed prediction model is compared to 4 existing models on the same database. The benchmark models existing in the same database are ANN/SA model by Alavi and Gandomi [4], GP/OLS model by Gandomi et al [45], MEP model by Alavi et al [5] and GP/SA model by Mohammadnejad et al [100]. It is clearly observed that the all the proposed prediction model has better precision compared to all these benchmark models.

e. The efficacy of all the proposed prediction model is further validated by comparing it to existing GMPE model such as Ambraseys et al. [6], Campbell and Bozorgnia [29] and Smit et al. [142].

f. The proposed prediction model based on advance machine learning overcomes the existing uncertainties due to the regression analysis method of the GMPEs as well the computational complexities of solving complex GMPEs. The existing GMPEs uses equations developed based on regression analysis and consists of many geophysical parameters. The proposed prediction models in this study use only 4 geophysical parameters, namely earthquake magnitude, source to site distance, average soil shear wave velocity and faulting mechanism, for the forecasting of ground motion parameter.

g. The proposed prediction model based on advanced machine learning could be used as a tool for faster and accurate prediction of the ground motion parameter with lesser calculation overhead (comparatively with GMPEs), in all areas such seismic risk assessment, seismic hazard analysis, earthquake resistant structural engineering etc., where the principal ground motion parameters are used as a vital input parameter.

h. In this work, a comparative study of the all the algorithm used for developing the predictive models is done in terms of the learning effectiveness of each algorithm. The learning effectiveness is measured in terms of a measure of 'overfitness'. This comparative analysis further highlights the advantages and drawbacks of each advanced machine learning algorithm.

i. In this thesis, a novel neuro-fuzzy learning machine called randomized adaptive neuro-fuzzy inference system (RANFIS) is proposed for predicting the parameters of ground motion parameters.

1.6 ORGANIZATION OF THE THESIS

The thesis is organized into ten chapters. A brief summary of each chapter is given below.

Chapter 1 gives an overview of the thesis. It gives an introduction to the problem statement by briefly explaining ground motion parameters and machine learning. The research gap is also highlighted which substantiate the author's contribution. The organization of the thesis is also explained in this chapter.

Chapter 2 details the literature survey of the ground motion parameters and the existing advanced machine learning algorithms. The chapter is organized such as it first gives an introduction of the various advanced machine learning algorithms and its application is various domains. The next section explains the existing application of the advanced machine learning algorithms in the field of earthquake engineering and geosciences. The chapter is concluded by the section which details the existing work done in the domain of ground motion parameter prediction.

Chapter 3 explains the database used in this study and the explains the various modelling parameters. The chapter is organized such that it gives all the information relating to the modelling such as data preprocessing, input parameters, modelling function, training and testing datasets, etc. Furthermore, the chapter also mentions the various criteria based on which the performance of the developed prediction models is compared. The chapter also lists the benchmark prediction models and GMPEs existing in this same database. All the proposed

prediction models in this study are compared to these benchmark models mentioned in this chapter.

Chapter 4 details the prediction model based on neural network based learning. The chapter meticulously describes a prediction model based on extreme learning machines (ELM). The chapter also describes a very simple ANN model, a prediction model based on single layer feed forward neural network (SLFN). The SLFN model is developed so that it could be compared with the ELM based prediction model. This comparison helps in validating and substantiating the advantages of the novel algorithm, extreme learning machines, based on neural network learning.

Chapter 5 explains the prediction model developed on kernel method based learning. The algorithm used is support vector regression (SVR). This chapter details the three variations of SVR algorithm used in modelling, namely ε-Support Vector Regression, ν-Support Vector regression and LS-SVR (least square Support Vector Regression). The chapter also details the modelling parameters such as the kernel functions used and other relevant parameters. All the three prediction models developed on these 3 algorithms are compared to obtain the best prediction model based on support vector regression.

Chapter 6 describes the prediction model based on decision tree learning. A Decision tree is a predictive learning method to develop a tree like model used to predict a target, based on a set of input features. The model developed is termed as regression tree, when the predicted target values take real continuous values. This chapter details the modeling using regression tree learning.

Chapter 7 consists of the prediction model based on a hybrid model, adaptive neuro fuzzy inference system (ANFIS). The chapter details the architecture and the modelling parameters.

Chapter 8 details the prediction model based on the novel neuro fuzzy algorithm RANFIS. The chapter first details the architecture of the novel algorithm RANFIS and compares it with the conventional ANFIS model for a benchmark problem. This comparison validates the advantages of the novel algorithm. The following sections of the chapter describe the application of this novel technique for forecasting ground motion parameter.

Chapter 9 compares all the developed prediction models. In the chapters 4-8, the respective developed prediction models are compared to existing benchmark models and the GMPEs. In this chapter, all the developed predictive models in this study, for forecasting ground motion parameter are compared in terms of prediction accuracy and error measure. Hence this chapter concludes the best prediction model among all the developed prediction models in this study. Futhermore, the prediction models are compared for 'learning effectiveness' of the algorithm. This section of the chapter helps in analyzing the learning ability of the algorithms. The learning ability of the existing benchmark models is also analyzed.

Chapter 10 concludes the thesis by highlighting the significant contribution of the thesis with the scope for further research in the area.

Chapter 2

LITERATURE SURVEY

In this chapter, a comprehensive survey of the domain of the study is done. The chapter is organized as follows, the popular advanced machine learning algorithms are briefly explained in section 2.1. The section 2.2 details the study of existing machine learning algorithms implemented in the field of earthquake engineering and geosciences. The existing soft computing techniques in the domain of ground motion parameter prediction are explained in section 2.3

2.1 ADVANCED LEARNING MACHINES

It has been observed that in the recent years, there has been a tremendous increase of application of soft computing techniques and machine learning in almost all fields of engineering and sciences. The most popular soft computing technique is artificial neural networks (ANN) (Haykin [60]), because of the ease of implementation. ANN are models based on the neural or nervous systems of brains. As the brain learns from experience, this biological inspired method of computing, also learning from the experimental training data. In simple words, ANN consists of group of processing elements called neurons, which are interconnected to form a structure. The strength of the connection between neurons is symbolized by weights, which are optimized during training of the model. The commonly used algorithm used for training weights is backpropagation. The neurons consist of activation functions which decide output of the neuron. The basic structure of ANN consists of an input layer, an output layer, and one or more layers of hidden neurons. The input layer consists of the input parameters, the output layer consists of single or multi neurons depending whether its single output or multi output problem. The layer between the input and output layer is called the hidden layer and it consists of hidden neurons. The ANN structure depending on the number of hidden layers is termed as a single layer or multi layer neural networks (Hornik [62]). The artificial neural networks are the most popular technique in the domain of neural network based learning.

ANN has gained popularity in almost all fields with diverse domains such as prediction problems (Adomaitis et al [3] , Keskin et al [78], Spandana et al [144], Thakur et al [151]), control system (Arslan et al [11], Badlani and Bhanot [14], Keskin and Goker [79], Keskin [80], Kiran and Rajput [84] , Metin et al [97,98]), brightness controller (Khan et al [81,82]) biomedical signal and image processing (ManjulaSri and Rao [92,93], ManjulaSri et al [94], Rao [120], Zuhair et al [163]), fuzzy systems (Nagaria and Saini [101], Nagaria and Singh [102]), decision systems (Ozkan and Inal [104]) and so on.

Though artificial neural networks are popular due to its ease of implementation, it suffers a major drawback which prevents it from being an efficient algorithm. Artificial neural network works on the principle of empirical risk minimization and hence the best ANN architecture is the one having minimum training error. This leads to two major issues of overfitting and local minima. ANN also has an overhead as its computational complexities are dependent on the dimension of the input space. Conventional neural networks have always been a popular machine learning technique widely used over the decades, and many researchers are constantly working for the advancement in the architecture and training speed of neural networks. One major improvement in the existing neural net architecture was the claim that the weights of the output layer are more relevant than the weights of the hidden layer by Schmidt et al [128]. This claim was well justified with feedforward network with a single hidden layer, where the weights of the hidden layer were randomly assigned and only the weights of the output layer were calculated using the pseudo inverse technique.

The same concept of random weight vector assignment was further extended into the implementation of functional link (FL) neural networks by Pao et al [108]. The random vector FL nets are similar to backpropogation (BP) or generalized delta rule (GDR) net with the difference that in FL nets the weight vectors are not learned but assumed randomly. In this architecture, the hidden layer neurons are randomly assumed, with direct links from input to output neurons and the output layer weight vector are obtained using pseudo inverse. A similar architecture of single hidden layer neural network is presented in Chen [33] uses an instant learning algorithm that rapidly decides the weights of the neural network. The paper also provides the upper bound of the number of the hidden nodes to be able to solve the output layer weight matrix exactly.

Related works on radial basis function (RBF) networks with randomly selected RBF centers and suitably selected RBF width were presented by Lowe [90] and Park et al [110]. Once the centers had been chosen, the adjustable weights of the output layer were determined by linear least-square optimization.

The recently proposed extreme learning machines (ELM) by Huang et al [65,66] are analogous to the above discussed architectures with minor variations mentioned by Wang and Wan [155]. The extreme learning machine is generalized single layer feedforward network (SLFN) in which the hidden layer parameters are not tuned. The weights of the nodes in the hidden layer in an ELM are randomly assumed, thus enabling the ELM architecture to be independent of the training data set. Hence the extreme learning machine is said to be highly scalable with lesser computational complexity. This architecture is being applied to various domains such as power systems (Huang et al [64]), forecasting (Han and Liu [59], Kaya and Uyar [72], Sun et al [149]), pattern recognition (Karpagachelvi et al [71]), biomedical signal processing (Song and Lio [147]) and so on. The extreme learning machine evolved in many stages as mentioned by Huang et al [65].

ELM uses sigmoid activation functions in the hidden layer or radial basis functions. In ELM, the bias of the output neuron is set to zero, while the bias of the output neuron in Schmidt et al [128] could be any value. Comparing ELM further with Chen [33] and Pao et al [108], it is observed that the only difference in the ELM architecture is that there is no direct link from input to output neurons. The direct link from input to output neurons was designed to deal with the linear components existing in the data. The effect on the performance of the machine by this minor variation has not been shown. This minor variation of removal of direct links in the ELM architecture could have an adverse effect on the performance if there exist linear components in the data. The ELM-RBF architecture is a slight variation of the RBF net by Lowe [90] such that RBF neurons widths are also randomized for all RBF neurons along with the RBF neuron centers. But this variation could not be seen as an improved architecture because it has been proved that RBF networks are universal approximators irrespective of having same or different RBF neuron width. In fact, it was shown that SLFN with arbitrary bounded and non constant activation function are universal approximators.

Kernel techniques are a group of novel methods for pattern analysis where support vector machines are vital elements. Kernel methods find the solution after explicitly mapping the data into the new high dimensional kernel Hilbert space. The number of coordinates is decided by the number of features of the data. Kernel trick uses kernel function for relating feature space by computing the inner product of data pairs. Support vector machine (Brereton and Lloyd [25]) is the most popular algorithm operating with kernels. Support vector machine algorithms are gaining much popularity compared to other soft computing techniques such as artificial neural networks. Though artificial neural networks are popular due to its ease of implementation, it suffers a major drawback which prevents it from being an efficient algorithm. Artificial neural network works on the principle of empirical risk minimization and hence the best ANN architecture is the one having minimum training error. This leads to two major issues of overfitting and local minima. ANN also has an overhead as its computational complexities are dependent on the dimension of the input space. Overcoming all these drawbacks, SVM provides global, unique and sparse solution to problems. It is also less prone to problem of overfitting as it works on structural risk minimization.

Therefore, this architecture is being applied to diverse domains such as credit scoring (Zhong et al [161]), prediction problems (Acir and Guzelis [1], Li-Xia et al [89], Patil et al [112], Samui and Kurup [126], Yan and Chowdhury [158], Zhou et al [162]), pattern recognition (Wang et al [156], Zhang et al [160], Samui et al 2012), control and power system (Eris et al [40], Ranaee et al [119], Xanthopoulos and Razzaghi [157]), signal processing (Rojo-Alvarez et al [124]), geotechnical engineering (Pal [105-107], Goh and Goh [49], Samui [125]), and so on. Although Support Vector Machines were initially used as a tool for pattern classification, it is gaining popularity in function estimation problem too. Chorowski et al [36] reviews a comparison between ELM and SVM as classifier. This study analyzes both learning machines as classifiers for its performance and computational time.

Decision Tree learning, a popular tool used in data mining is gaining popularity as predictive tool for supervised learning in various fields such as in medicine, used as a predictive tool for diagnosis of diseases (Lemon et al [87], Stothers et al [145]), used as a forecasting tool (Gokhale and Lyu [48]). Learning by decision trees has an overhead over other learning techniques because of the representation ability of the model which makes it intuitive and

adaptable. The interpreting of the results becomes easier in a tree representation. Moreover, since the tree representation is hierarchical in nature, the modelling is relatively easier compared to linear modelling in case of a large number of input features.

Adaptive neuro fuzzy inference system (ANFIS) is yet another soft computing technique which is gaining popularity as a computational intelligent system. ANFIS is an integrated hybrid architecture of fuzzy logic with neural networks, such that the knowledge gained by the fuzzy logic is used by the learning algorithm of the neural network. The initial fuzzy model is derived with the rules from the data, and the neural network learns and trains the rules to get the final model. The hybrid learning algorithm used in ANFIS consists of gradient descent for the fuzzy layer and least square estimate (LSE) for the linear output layer. This architecture has been implemented in various domains such as geotechnical engineering (Asadi et al [12], Azamathulla et al [13], Cabalar et al [28]), prediction problems (Bagheri et al [15], Bektas Ekici and Aksoy [16], Boyacioglu and Avci [21], Melin et al [99], Singh et al [136], Singh et al [137], Tien Bui et al [149]), image processing (Singh et al [134], Singh and Barada [135]), modelling systems (Hosoz et al [63], control systems (Khuntia and Panda [83], kurtulus and Flipo [86], Singh and Barada [133]) and so on. Although ANFIS architectures are widely used, it has a major drawback of high computational complexities, which makes the algorithm slow, with higher number of membership functions.

In artificial intelligence, there exists a domain of problems, known as search problems, that work on the principle of optimization. The popularly used algorithms for optimization problem are genetic algorithms (GA) and particle swarm optimization (PSO). Genetic algorithm belongs to the class of evolutionary algorithms, and the optimization is done by selecting the best solution from a set of candidate solutions by testing its fitness on a function termed as objective function. Particle swarm optimization is a computational method which works on optimizing the problem by repeatedly improving the candidate solution by improving the quality. Both GA and PSO are heuristic search algorithms. These algorithms are also being used in various domain such as pattern recognition (Anzar et al [8], Anzar and Sathidevi [9], Quaranta et al [116]), optimization in control system (Bhateshvar and Mathur [17], Raviprasad and Singh [138]), prediction using hybrid model (Patil et al [112]), optimization in signal processing (Rajavel and

Sathidevi [118]), improving architecture of existing learning machines such as SVM (Amari and Wu [7],Ardjani et al [10], Chou et al [38] , Sanchez [127]), and so on.

2.2 MACHINE LEARNING IN EARTHQUAKE ENGINEERING AND GEOSCIENCES

Among all the natural calamities earthquakes are the most threatening natural calamity, due its tremendous destructive property. Seismic hazard is anything associated with an earthquake like strong ground shaking, landslides, liquefaction, faulting etc. that may affect the normal activities of the people. The modelling of ground motion signal for assessment of seismic hazard such as Goda et al [47] and Sokolov and Wenzel [146] is important to evaluate the damage of an earthquake to the environment. Due to growing urbanization, there is tremendous increase in the population density in earthquake prone areas, which in turn is increasing the demand for earthquake resistant structures.

With the recent advances in the field of artificial intelligence and soft computing techniques, the traditional mathematical functions used for big data analyses and other complex analyses are replaced by them. Many researchers recently are working to merge these advanced techniques of AI with the field of earthquake engineering and geosciences.

Gandomi and Alavi [44] used a new method, multi-gene genetic programming (MGGP) method for the analysis of earthquake engineering and geotechnical systems. MGGP is a modified GP approach for selection of model structure combining traditional regression technique for parameter estimation. This study clearly highlights the drawback of traditional regression technique for the domain, such as geotechnical engineering which involves complex processes depending on multivariables. Complexity of geotechnical behavior is due to multivariable dependencies of responses of soil and rock. The traditional forms of engineering design solutions are implemented after simplifying the complexities with assumptions. Incorporating these simplified assumptions into modelling leads to very large errors. These overheads are minimized by using advance learning machines.

Bose et al [19] used ANN for earthquake early warning for finite faults. Erol and Erol [41] used machine learning for geoid modelling. The algorithms used are ANN, ANFIS and

wavelet neural networks. In this study, the models developed using machine learning are compared to the multivariable polynomial regression equations and thus the study validates the advantages of using machine learning over regression analysis.

Goyal et al [50] merged machine learning in the field of climatology. In this paper, a comparative study of learning machines such as ANN, LS-SVR, ANFIS for modelling evaporation system for the tropical climate environment. This study clearly proves that the machine learning outperforms the traditional empirical equations of this field, such as empirical Hargreaves and Samani method (HGS), and the Stephens-Stewart (SS) method.

Gullu [53] used genetic expression programming (GEP) for estimating the strength and elasticity of soil. The ground motion parameters are significant for the damage potential of structures. For designing a good structure, the site conditions and structure ability are equally relevant as ground motion parameters [53,54,57]. Chen et al [32] used support vector machines for assessing the seismic hazard for school building. Shinozuka et al [139] made the attempt of modelling the earthquake wave motion with synthetic data by using computer algorithms. This study was an attempt to merge the recently advancing computer applications to the domain of geosciences and earthquake engineering. Segou and Voulgaris [130] used the standard available software, MATLAB for processing the ground motion signals and hence estimating the ground motion parameters. Jafarian et al [68], Kermani et al [76] used genetic programming for modelling the ratio of PGV to PGA. In this study the AI technique GP has been used to develop a new parameter, the ratio of PGV to PGA. The study also validates well the significance of the new parameter for understanding the ground motion.

Chakraverty [30] used neural network for simulating the response of a two storey building subjected to earthquake. Gullu [51] used AI for prediction of shear wave velocity. Park et al [109] used ANFIS for mapping of ground subsidence hazard. Shiri et al [140] made a comparative study of soft computing for predicting fluctuation in the level ground water. The machine learning algorithms used in the study are ANN, ANFIS, SVM and GEP. Pardhan [115] used the soft computing techniques in geotechnical domain, such as predicting the landslide susceptibility. The techniques such as decision trees, SVM and neuro fuzzy model ANFIS were used and a comparative study was done. Oho and Pardhan [103] used ANFIS for mapping shallow landslides in the tropical hilly area to landslide susceptibility. Furthermore, Tien Bui et al

[150] used ANFIS for prediction of landslide susceptibility for a province in Vietnam. Seyedpoor et al [132] used soft computing for design optimal arch dam subjecting to earthquake loading. The optimization algorithm PSO was used for stochastic optimization.

Hence, from this study it is observed that the advanced machine learning has gained popularity in prediction problems relating to earthquake engineering and that it could be extended to prediction of ground motion parameters.

2.3 SOFT COMPUTING FOR PREDICTION OF GROUND MOTION PARAMETERS

Ground motion parameters are vital for designing earthquake resistant structures and seismic hazard analysis (Sehhati et al [131], Giacinto [46]). Conventionally, the ground motion parameters are estimated based on ground motion prediction equations (GMPEs) which are developed using the traditional regression analysis method. These predictive equations relate the ground motion parameter in terms of independent variables such as earthquake magnitude, source to site distance, site conditions, seismic wave propagation and earthquake source characteristics. These independent variables are described in terms of many other geophysical parameters. Geophysical parameters (Douglas [39]) which describe earthquake source characteristics are earthquake magnitude, seismic moment, fault direction, faulting mechanism based on strike, dip and rake angles, stress drop, etc. The parameters like focal depth, epicentral distance, hypocentral distance, representing the distances travelled by the seismic waves etc. constitute the path effects.

The site conditions and seismic wave propagation properties include the soil type at the recording site and propagating velocity of seismic waves as seismic waves travel with different velocity while propagating through soil and rock. The study of the faults and its offsets and slip rates by Ren et al [121,122,123] points out the uncertainties involved. Thus, the drawbacks of using regression analysis for the development of predictive equations is that, the high nonlinearity and inhomogeneity among the independent variables directly affect the coefficients of the independent variables in the developed regression equation. Moreover, in regression analysis the model is developed based on a predefined linear or nonlinear equation, with the

hypothesis of normality of residuals for testing the developed model. Hence the developed predictive equation based on regression analysis is highly uncertain due to both computational uncertainties and the uncertainties of independent variables. Thus, there is a huge need for the modelling of ground motion parameters using newer techniques so as to reduce the existing errors in the ground motion parameter estimation.

A new method called CAE (conditional average estimator) was applied to the attenuation relationship by Fajfar and Perus [42]. CAE is non parametric multi dimensional regression approach having a structure similar to neural networks. In this paper horizontal component of PGA is predicted as a function of two parameters namely earthquake magnitude and distance. The Ground motion parameters predicted using CAE method are compared to the existing Ground motion prediction equations (GMPE`s) models (Boore and Atkinson [18], Campbell and Bozorgnia [29], Abrahamson and Silva [2], Chiou and Youngs [35], Idriss [67]) by Fajfar and Perus [43] and Perus and Fajfar [113]. The major drawback of using conventional method is that applying regression analysis for the development of attenuation relationships is a lot more complex because of the high nonlinearity and inhomogeneity among the parameters. Thus, there is a huge need for the modelling of ground motion parameters using newer techniques so as to reduce the existing complexities.

Kerh and Chu [73] introduced the application of neural networks for estimating ground motion parameter, PGA. Although the prediction accuracy of the model was not very high, the study validated that soft computing could be applied in the prediction of ground motion parameter. This later lead to studies such as, Kerh et al [74] and Gunaydm and Gunaydm [58]. Kerh et al [74] applied artificial neural networks for predicting PGA using microtremor measurement. Gunaydm and Gunaydm [58] applied ANN in northwestern Turkey region for PGA prediction. Kerh and Ting [75] used back propagation neural networks for predicting the PGA in three different directions (vertical, east-west, and north-south) at stations along the high speed railway line in Taiwan using the seismic parameters and the historical earthquake data. The estimated values were compared with the microtremor measurement of the respective station.

Gullu and Ercelebi [55] made a remarkable effort to develop an attenuation relationship based on strong motion data on turkey region using artificial neural networks. Though the

correlation between the observed and predicted PGA raised many questions (Gullu and Ercelebi [56]), the attempt made indicated that neural networks could be applied in the field of seismology. An equation discovery approach was used by Markic and Stankovski [95] for modelling PGA. The equation discovery is a machine learning technique which uses context free grammar (CFG) for generating equation structures which best describes the given data. In the study, Lagramge equation discovery system is used to obtain equations for predicting PGA. The system gives a fair correlation between the observed and the predicted PGA values. The equations obtained by Lagramge equation discovery system, for estimation of PGA have lesser complexities compared to the existing GMPEs. Cabalar and Cevik [27] developed an attenuation relationship for the Turkey region using genetic programming (GP) for the prediction of the PGA. The developed model when compared to the other existing models gives better correlation. The advantage of using GP for modelling PGA is that the functions are not predefined as in the case of the traditional regression analysis.

Gandomi et al [45] developed a new GMPE model using a hybrid model of genetic programming (GP) and orthogonal least squares for prediction of PGA, PGV, and PGD. The model gives a fair correlation value with lower MSE values. The model developed is advantageous as the equations developed by this hybrid model for the prediction of ground motion parameters are comprehensible compared to the equations of GMPE models. Alavi and Gandomi [4] used a hybrid model ANN/SA (coupling of ANN with simulated annealing) to predict the principal ground motion parameters PGA, PGV and PGD. The model is better than the GMPEs for the same database as it gives good correlation value. The disadvantage of the model is the time taken to achieve acceptable MSE due to the introduction of simulated annealing. Alavi et al [4] developed a variant GMPE model for prediction of ground motion parameters using multi expression programming (MEP). This model gives comparatively reasonable prediction accuracy and validates the advantage of MEP over the traditional GMPE equations developed using regression analysis. Although this model develops the ground motion prediction equation considering the complex nature of the ground motion parameters, the model suffers the drawbacks of genetic programming (GP) based models as the functions are formed randomly and not on the physical process.

Gullu [52] made an attempt to predict Peak ground acceleration for the turkey region using a new approach called Gene Expression Programming (GEP) and conventional regression method. GEP`s are an extension to Genetic Programming. The best model is selected by ranking the models using likelihood based estimation. The model is said to have a fair validation when compared to the existing attenuation relationship for the region. Mohammadnejad et al [100] developed a novel GMPE model using the hybrid model of genetic programming (GP) and simulated annealing (SA). Although the prediction accuracy of the developed model is not very high, the developed model is advantageous as it gives a lesser complex prediction equation for prediction of principal ground motion parameters PGA, PGV and PGD.

It is observed from the study that the existing soft computing technique for the prediction of ground motion parameter is mainly neural networks and its hybrid models. Thus, there is a scope for modelling efficient predictive models.

Chapter 3

DATABASE AND DATA PREPROCESSING

3.1 INTRODUCTION

In this chapter, the database used for this study is explained. Real earthquake records are used for modelling. In order to have a common platform for comparison of the models, the database consisting of real earthquake data, available globally is considered. In this chapter the entire data preprocessing and modelling parameters are detailed. Section 3.2 details the database used for modelling. The existing models in the same database are detailed in section 3.3. Section 3.4 explains about the modelling parameters. The various steps in data preprocessing is explained in section 3.5 such as selection of input parameters, normalization of the database, splitting of the database into training and testing data sets. The chapter is concluded by section 3.6 which detailed the various criteria based on which the developed prediction models are evaluated for its efficacy and validity.

3.2 DATABASE

Pacific Earthquake Engineering Research Center (PEER) systematized and compiled a database [111] popularly known as NGA WEST 1, in 2003 as a part of a project named PEER-NGA project. The database file is termed as NGA flatfile V 7.3. The database includes a very large set of ground motion recordings recorded worldwide. A part of the database is shown in appendix B. It consists of shallow crustal earthquakes recorded in active tectonic regimes. The database is comprehensive having sets of meta-data, including around 116 geophysical parameters such as different distance measure (column 48 to column 53 of NGA flatfile V 7.3), various site characterizations, earthquake source data, etc. The database consist of 3351 earthquake records and has been used to develop few worldwide ground motion prediction equations (GMPE) models. The few GMPE models developed in this database are Abrahamson and Silva model [2], Idriss model [67], Boore and Atkinson model [18], Campbell and Bozorgnia

model [29] and Chiou and Youngs model [35]. The database used in this study is a subset of NGA flatfile V 7.3. The sample database is shown in appendix C. The data sets in this study consist of 2252 earthquake records, which is further split into training and testing data consisting of 2815 and 563 earthquake records respectively. The section 3.5 in this chapter clearly details the various steps of data preprocessing.

3.3 EXISITING MODELS IN THIS DATABASE

The Table 3.1 shows the existing prediction models based on soft computing techniques. These models are considered as benchmark models for comparison of all the proposed prediction models in this thesis. From the literature survey, it is observed that these are the only existing prediction models in this database [111] based on soft computing for prediction of ground motion parameter.

Table 3.1 Existing soft computing ground motion prediction models

Model	Approach	Authors
ANN/SA model	**Artificial neural network/ simulated annealing**	**Alavi and Gandomi 2011 [4]**
MEP model	**Multi Expression Programming**	**Alavi et al 2011 [5]**
GP/OLS model	**Genetic Programming/Orthogonal least squares**	**Gandomi et al 2011 [45]**
GP/SA	**Genetic Programming/ simulated annealing**	**(Mohammadnejad et al 2012) [100]**

Hence the efficacy of all the proposed ground motion prediction models is marked by comparing it with the above mentioned benchmark models and with the existing GMPE models on the same database.

3.4 MODELLING PARAMETERS

In this study an attempt is made to model the principal ground motion parameters (Y) as in Eq. 3.1

$$ln(Y) = f(M, F, V, ln\,(D)) \qquad (3.1)$$

where, Y is PGA (g), PGV (cm/s) and PGD (cm), M is Earthquake magnitude, F is the style of faulting, V is the average shear wave velocity and D is the distance. The modelling equation is formulated based on the information gained from the literature survey.

The Table 3.2 shows the existing modelling equations for the prediction of ground motion parameter. It is observed that in all the studies, the ground motion parameter is expressed in terms of earthquake magnitude, faulting style, the average shear wave velocity and source to site distance. In the database there are six different measures of source to site distance. In this study closest distance measure is selected in consistency with the benchmark models.

Table 3.2 Existing ground motion prediction models

Model	Approach	Database	Authors	Modelling Equation	Distance measure
ANN/SA model	**Artificial neural network/ simulated annealing**	**PEER [111]**	**Alavi and Gandomi 2011 [4]**	$\ln(PGA) = f(F, M, V, R_{cld})$	**Closet distance**
MEP model	**Multi Expression Programming**	**PEER [111]**	**Alavi et al 2011 [5]**	$ln(PGA) = f(\sin(\lambda), M, V, R_{cld})$	**Closet distance, rake angle**
GP/OLS model	**Genetic Programming/Orthogonal least squares**	**PEER [111]**	**Gandomi et al 2011 [45]**	$\ln(PGA) = f(F, M, V, R_{jb})$	**Joyner–Boore distance**
Lagramge system	**Equation Discovery approach ,CFG**	**Perus and Fajfar 2009 [42]**	**Markic and Stankovski 2013 [95]**	$\ln(PGA) = f(F, M, V, R_{jb})$	**Joyner–Boore distance**
GP/SA	**Genetic Programming/ simulated annealing**	**PEER [111]**	**Mohammadnejad et al 2012 [100]**	$\ln(PGA) = f(F, M, V, R_{cld})$	**Closet distance**

3.5 DATA PREPROCESSING

In this section, the entire data preprocessing steps are explained. The database used in this study is a subset of PEER database [111].

3.5.1 INPUT PARAMETERS

The NGA flatfile [111] consists of 116 columns, each representing a particular geophysical parameter. The sample data records are shown in appendix B with the various 116 geophysical parameter. In this study only four geophysical parameters are considered. The input parameters chosen are earthquake magnitude, faulting mechanism, shear wave velocity and source to site distance.

The detailed description of the input parameters is as follows:

a. Earthquake magnitude (M): In this study the moment magnitude (M_0) of the earthquake is represented by this variable. The moment magnitude of the earthquake best represents the energy released during the earthquake.

b. Faulting Mechanism (F): This variable is used to represent the basic three types of faulting which occur during the earthquake rupture process. The values are taken based on the 11^{th} column of the NGA flatfile database which represent the mechanism based on rake angle. Rake angle is the angle measured in anticlockwise direction on the fault plane, from strike direction to average slip direction. Fig. 3.1 shows the rake angle as pictorial representation.

   The type of faulting basically denotes the direction of the movement of the fault plane which is decided based on the values of the rake angle. Table 3.3 details the faulting mechanism given in the database based on the rake angle. Table 3.4 represents the values for F used in this study.

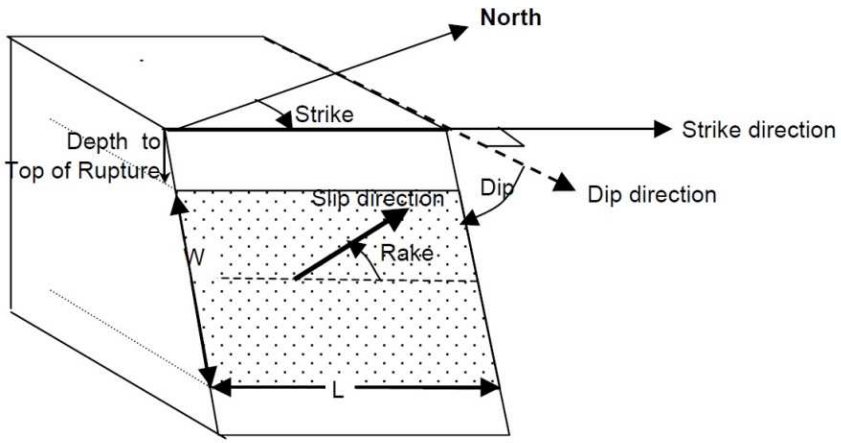

Fig. 3.1: Pictorial representation of Faulting mechanism [111]

Table 3.3: Faulting Mechanism based on Rake Angle as mentioned in PEER database

Faulting type	Mechanism Class	Rake Angle (in degrees)
Strike slip	**00**	**-180 <Rake< -150, -30 <Rake< 30, 150 <Rake< 180**
Normal	**01**	**-120<Rake< -60**
Reverse	**02**	**60 <Rake< 120**
Reverse- Oblique	**03**	**30 <Rake< 60, 120 <Rake< 150**
Normal-Oblique	**04**	**-150 <Rake< -120, -60 <Rake< -30**

Table 3.4: Description for input variable F used in this study

Values of F	Value in NGA Flatfile	Faulting style
1	02,03	Reverse
2	01,04	Normal
3	00	Strike slip

c. Velocity (V): In this study this variable represents the average shear wave velocity in the top 30 m layers at the site. This geophysical parameter is highly significant as it represents the site influence on the seismic signal.

d. Distance (D): In this study, this variable is used to represent source to site distance. Table 3.5 represents 6 different types of measured distances as given in the NGA flatfile. From the literature survey, it is observed that each of the distance measures as in Table 3 has its own significance. It is observed from Table 3.2 that Epicentral distance, Joyner-Boore distance and closest distance are relatively more significant The distance measure used in this paper is closest distance which has also been used in [4,5,100]. Fig. 3.2 gives the pictorial representation for various measures of distances.

Table 3.5: Various measures of distances in PEER-NGA database

Measured distance	Description
Epicentral distance	Distance from the recording site to epicenter
Hypocentral distance	Distance from the recording site to hypocenter.
Joyner-Boore distance	Shortest horizontal distance from the recording site to the vertical projection of the rupture
Campbell distance	Shortest distance from the recording site to the seismogenic portion of the ruptured area (Campbell, 1997).
Root mean square distance	Root-mean-squared distance
Closest distance	Closest distance from the recording site to the ruptured area

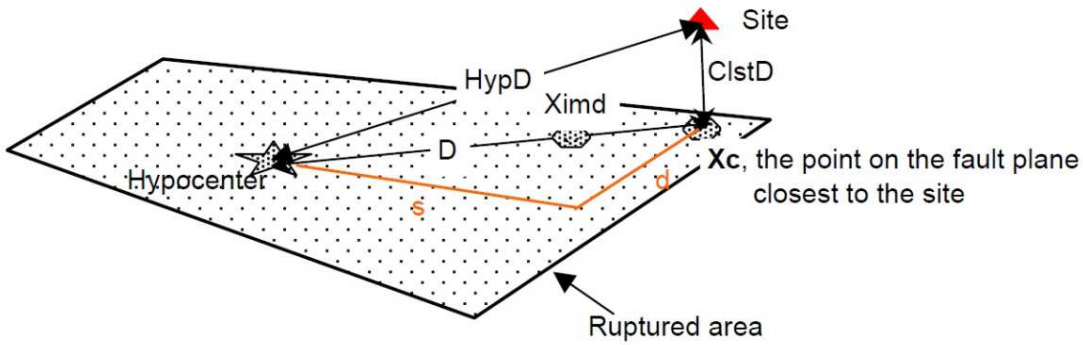

Fig. 3.2 Pictorial representation of distance measures [111]

3.5.2 DATA NORMALIZATION

The data sets used for the analysis in this study are normalized. There are several methods for the normalization of the data. The Eqs.3.2-3.4 represent the method of normalization used in this study. Let Z be the variable and Z_{max}, Z_{min} represent the maximum and minimum values of the variable respectively. Let the range within which the variables is to be normalized be [P, Q].

The range of normalized data is chosen as in the existing benchmark prediction models mentioned in Table 3.1 and is set as (0.05, 0.95). Let Z_n be the normalized value for variable and it is defined as follows

$$Z_n = cZ + d \qquad (3.2)$$

where,

$$c = (Q - P)/(Z_{max} - Z_{min}) \qquad (3.3)$$

$$d = Q - (c * Z_{max}) \qquad (3.4)$$

3.5.3 TRAINING AND TESTING DATASETS

The NGA flatfile V7.3 database consist of 3551 records, including the 116 geophysical parameters. Considering the four geophysical parameters used for modelling, the database is narrowed down. From this minimized database, the earthquake records having incomplete attribute value are deleted. Furthermore, the database is critically analyzed and the duplicate records are also removed. Thus, the corrected database has 2815 records. Hence, in this study the prediction model is developed in this database of 2815 records. The sample data used in this study is shown in appendix C. The following Table 3.6 shows the statistical parameters of the variables used in this study.

The database consisting of 2815 records is divided into training and testing data sets. These two datasets are extracted from the database such that they are in the ratio of 4:1. The segregation of entire 2815 records into these two data sets should be such that the statistical parameters of the variables used in the analysis are consistent with both sets. Thus the training data set has 2252 records and testing data set has 563 records.

Table 3.6: Statistical parameters of variables used in this study

Statistics	Maximum	Minimum	Median	Standard Deviation	Sample Variance	Range	Skewness
M	**7.9**	**5.2**	**6.3**	**0.59**	**0.35**	**2.7**	**0.81**
F	**3**	**1**	**1**	**0.85**	**0.72**	**2**	**1.19**
V(cm/s)	**2016.13**	**116.35**	**345.42**	**175.09**	**30654.89**	**1899.78**	**2.28**
D (km)	**366.03**	**0.07**	**63.79**	**50**	**2524.41**	**365.96**	**1.39**
PGA (g)	**1.66**	**0.01**	**0.04**	**0.13**	**1.53**	**1.65**	**4.19**
PGV (cm/s)	**169.96**	**0.10**	**5.01**	**14.46**	**209.02**	**169.86**	**3.59**
PGD (cm)	**232.39**	**0.01**	**1.48**	**11.21**	**125.67**	**232.38**	**7.91**

The range of normalization used in the section 3.5.2 and the segregation of the dataset into training and testing datasets is in accordance to the existing benchmark models mentioned in Table 3.1. The flowchart detailing the entire steps of data preprocessing is shown in Fig.3.3.

To further validate the developed model, other than the testing data set of 563 records, another testing data set is also considered. PEER has updated the database and termed it as NGA WEST 2. NGA WEST 2 [23] is an extension of NGA WEST 1 database, and it consists of 21,539 events, whereas the latter had 3551 events. Thus, out of 17988 events which are not included in the NGA WEST 1 database, 140 events are selected, and is named as NGA WEST 2 testing data set.

Hence all the developed prediction models are modelled on 2252 records of the training data and tested on 563 records of the testing dataset as well as on another 140 records of NGA WEST 2 dataset. Thus, the efficacy of the developed models is very well validated.

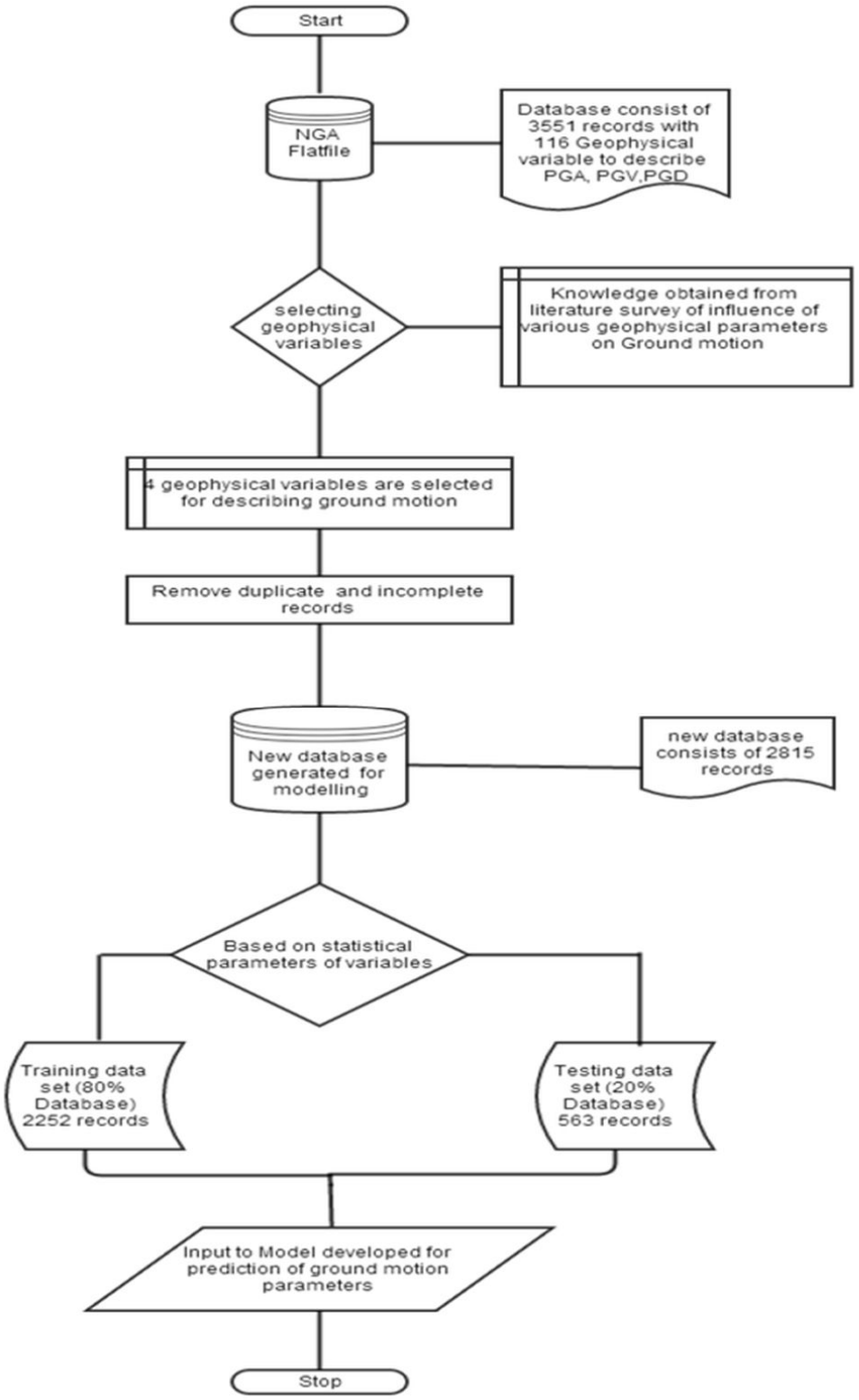

Figure 3.3: Flowchart for the Data preprocessing

3.5.4 EXPERIMENTAL ENVIRONMENT

In this thesis, all the algorithms are implemented and tested on C and MATLAB R2012 b platform with processor Intel(R) core(TM) i3-3220 and 4GB RAM. Thus the comparison of the algorithms in terms of computational time could also be justified as they are run on the common platform.

3.6 CRITERIA FOR PERFORMANCE MEASURE

There are various standard methods used for measuring the performance of the developed model. This section explains the various criteria used in this study, which acts like a yardstick for analyzing the performance measure of the developed model. The criteria chosen are inconsistent with the studies mention in Table 3.1. The criteria are valid for both training and testing data sets. The prediction accuracy of the models is measured in terms of correlation coefficient (R) and the error measure. There exists a well reasoned presumption [143] that if the correlation coefficient (R), $|R|>0.8$ and error percentage is minimum; there is a high correlation between the predicted and the observed values.

In this study, four different measures of error, namely mean absolute error (MAE), mean absolute percentage error (MAPE), mean square error (MSE) and root mean square error (RMSE) are used. These measures of error are explained as follows.

Let the total number of records be k. Let a_i denoted the observed or the real output and p_i denote the predicted value. The average value of the observed output be represented as avg (a_i) and the average value of the predicted output be represented as avg (p_i).

The equations representing the measures of errors are as follows.

i. Correlation Coefficient (R)

$$R = \sum_{i=1}^{k}(a_i - avg(a_i))(p_i - avg(p_i)))/ \sqrt{\sum_{i=1}^{k}(a_i - avg(a_i))^2 \sum_{i=1}^{k}(p_i - avg(p_i))^2}$$

ii. Mean Absolute Error (MAE)

$$MAE = \sum_{i=1}^{k} | a_i - p_i | /k$$

iii. Mean Absolute Percentage Error (MAPE)

$$MAPE = (1/k)\sum_{i=1}^{k} | (a_i - p_i)/a_i |$$

iv. Mean Squared Error (MSE)

$$MSE = \sum_{i=1}^{k} (a_i - p_i)^2)/k$$

v. Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{\sum_{i=1}^{k}(a_i - p_i)^2)/k)}$$

All the developed models are analyzed in terms of the above mentioned criteria. The scrutiny of the developed predictive models is done in terms of four different measures of error. For an efficient predictive model, the error measure should be less. The correlation coefficient (R) gives the linear relation between the observed and the predicted values. The higher value for R signifies that the relationship between the predicted and observed is linear. Hence the predicted value is comparable to the observed value. If the model has high R value, generally $|R|>0.8$, it is concluded that the prediction accuracy of the model is good.

There are cases such that although the model gives high R value, the error measure is also high. This signifies that the model is over trained and that overfitting of the data has occurred. Hence , in this study the various measure of error is considered so that the developed predictive models are critically analyzed for overfitting.

Chapter 4

PREDICTION MODEL BASED ON NEURAL NETWORK LEARNING

In this chapter a prediction model is developed based on neural network based learning. The two architectures used are single layer feedforward neural network (SLFN) and Extreme learning machine (ELM). The chapter is organized as follows. Section 4.1 gives the introduction to neural network based learning. The prediction model based on SLFN is explained in section 4.2. The prediction model based on ELM is detailed in section 4.3. Section 4.4 compares the SLFN prediction model with the ELM prediction model. The chapter is concluded in section 4.5 which further analyzes the models developed based on neural network based learning.

4.1 INTRODUCTION

The basic of any neural network architecture (Haykin [60,61]) is perceptrons. Thus in this section, the learning of perceptron is explained. A single perceptron equation is given as, $net = \sum_{i=0}^{n} w_i * x_i + b$, where x_i is the ith input with w_i the corresponding weight. Hence single layer perceptron model, is all about training the weight vector to get the desired output.

The perceptron learning algorithm could be defined as follows:

Let x_i be the input, w_i be the weight between input and perceptron. Let net_i Be the output given by the perceptron and $target_i$ be the actual required output. The idea of the perceptron learning algorithm is to obtain the best weight vector w_i such that $target_i - net_i = 0$. The perceptron learning rule could be defined as $w_i(t+1) = w_i(t) + \delta w_i(t)$. Hence aim is to find δw_i. Let μ be the learning rate.

Case 1: when $target_i - net_i < 0$, it implies that $\sum w_i x_i$ is too large and hence w_i should be modified as $new\_weight_i = old\_weight_i - \mu$.

Therefore reframing the equation we obtain,

$$new\_w_i = old\_w_i - (\mu * x_i) \qquad (4.1)$$

Case 2: when $target_i - net_i > 0$, it implies that $\sum w_i x_i$ is too small and hence w_i should be modified as $new\_weight_i = old\_weight_i + \mu$. Therefore reframing the equation we obtain,

$$new\_w_i = old\_w_i + (\mu * x_i) \qquad (4.2)$$

Combining Eq.(4.1-4.2), it could be written as $new\_w_i = old\_w_i + \mu(target_i - net_i) * x_i$.

Thus $\delta w_i = \mu(target_i - net_i) * x_i$, is called the perceptron learning rule and μ is called the learning rate.

4.2 PREDICTION MODEL BASED ON SINGLE LAYER FEEDFORWARD NETWORK (SLFN)

A prediction model using artificial neural network is developed for forecasting PGA. The simplest ANN structure is considered for modelling. It is clearly observed that the existing benchmark models for forecasting ground motion parameter are based on ANN structure. In this study, a prediction model based on ANN structure is developed so that it could be used as a base model for comparing it with the advanced learning algorithm (ELM) based on which prediction model is proposed. Furthermore the comparative study between the two models helps in validating the advantages of the ELM.

A single layer feedforward network (SLFN) is considered in this study. It consists of one input layer having four input features, one output layer which denotes the ground motion parameter to be predicted and one hidden layer. The hidden layer consists of 20. The activation function for hidden layer is TANSIG and the weights and bias values of the network are updated according to Levenberg-Marquardt optimization using training function TRAINLM. Adaptation function is LEARNGDM. It is the gradient descent with momentum weight and bias learning function.

4.2.1 RESULTS ANALYSIS AND DISCUSSIONS

In this section, the prediction model developed using SLFN for forecasting PGA is analyzed and compared with the existing benchmark models. Fig. 4.1 shows the results obtained by the proposed model. Table 4.1 tabulates the results obtained by the proposed model in comparison with other models. It is observed that the efficacy of the model is very less compared to the existing models. Table 4.2 compares the proposed model with the existing GMPEs. It is observed that the proposed model is better than the existing GMPEs. Thus, it substantiates that neural based modelling is better than the traditional regression analysis method used by GMPEs.

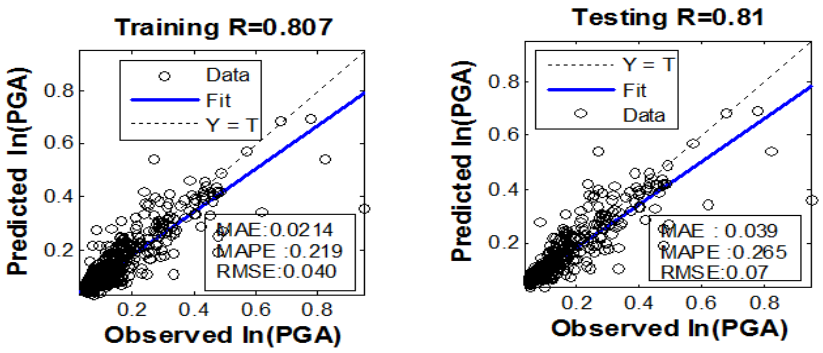

Fig. 4.1:SLFN prediction model

Table 4.1: Comparison of SLFN prediction model with existing models

Criteria	SLFN		ANN/SA [4]		GP/OLS [45]		MEP [5]		GP/SA [100]	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.8073	0.8188	0.869	0.855	0.836	0.811	0.842	0.834	0.833	0.839
MAE	0.0214	0.0397	0.30	0.46	0.478	0.488	0.363	0.697	n/a	n/a
MAPE	0.2196	0.2650	0.14	0.13	n/a	n/a	n/a	n/a	0.158	0.144
MSE	0.002	0.005	n/a	n/a	0.358	0.406	0.362	0.389	0.381	0.380
RMSE	0.040	0.070	n/a	n/a	0.836	0.637	0.602	0.624	0.617	0.616

Table 4.2: Comparison of SLFN prediction model with GMPEs models

Model	Criteria MAPE
SLFN model	0.265
Campbell and Bozorgnia [29]	0.93
Least Square regression analysis [96]	0.16
Ambraseys et al [6]	0.95
Smit et al [142]	14.58

The major drawback of neural network based modelling is that the learning is dependant on the database used for training and generalization is very poor. Table 4.3 shows the results obtained by the SLFN prediction model for NGA WEST 2 dataset. It is clearly observed that generalization is poor.

Table 4.3:SLFN prediction model for NGA WEST2 data

Criteria	SLFN
R	0.4242
MAE	0.0147
MAPE	0.2171
RMSE	0.024

From the above analysis of the results obtained it is clear that although neural network based modelling is advantageous, the use of neural network has lots of disadvantages.

Hence the new architecture based on neural network learning, extreme learning machine (ELM) was used for modelling. In this study the prediction model based on the single feedforward neural network was developed just as a case study to show the drawbacks of neural networks and to highlight the advantages and need of using ELM for modelling.

4.3 PREDICTION MODEL BASED ON EXTREME LEARNING MACHINE (ELM)

4.3.1 ELM ALGORITHM

The extreme learning machine [66] is generalized single layer feedforward network (SLFN) in which the hidden layer parameters are not tuned. The weights of the nodes in the hidden layer in an ELM are randomly assumed, thus enabling the ELM architecture to be independent of the training data set. Hence the extreme learning machine is said to be highly scalable with lesser computational complexity.

A neural network is said to have a good generalization capability if the training error as well as the norm of the weights is minimum. In ELM since there is no tuning of the hidden layer, it aims to get a minimum norm of the weights of output node for better generalization characteristic. In other words, ELM architecture is analogous to a single layer feedforward neural network in which the input weights and the hidden layer bias are fixed. The basic architecture is explained as follows:

Consider a training dataset with N samples (X_i, T_i) where $X_i = [x_{i1}, x_{i2}, ..., x_{in}]^T$ and $T_i = [t_{i1}, t_{i2}, ..., t_{im}]^T$. To solve this classification problem, consider a conventional SLFN with $\widetilde{N}$ hidden nodes and activation function g(x). The output nodes are assumed to be linear. Let W_i be the weight vector between the input nodes and the jth hidden node. Let β_i be the weight vector for the linear output nodes and b_i be the threshold of the i^{th} hidden node.

Then the output of the linear layer O_j can be obtained as

$$O_j = \sum_{i=1}^{\tilde{N}} \beta_i g_i(X_j) = \sum_{i=1}^{\tilde{N}} \beta_i g(W_i X_j + b_i) \ , j = 1 \ldots . N \qquad (4.3)$$

where,

$$W_i = [w_{i1}, w_{i2}, \ldots, w_{in}]^T \qquad (4.4)$$

$$\beta_i = [\beta_{i1}, \beta_{i1}, \ldots \beta_{im}] \qquad (4.5)$$

This network can approximate the given problem with *N* samples with zero error. Thus, there exist parameters β_i , W_i and b_i such that

$$\sum_{i=1}^{\tilde{N}} \beta_i g(W_i X_j + b_i) = T_j \qquad j = 1 \ldots . . N \qquad (4.6)$$

The above equations may be written as $H\beta = \boldsymbol{T}$ (4.7)

where,

$$\mathrm{H} = \begin{bmatrix} g(W_1 . X_1 + b_1) & \ldots & g(W_{\tilde{N}} . X_1 + b_{\tilde{N}}) \\ \vdots & \ddots & \vdots \\ g(W_1 . X_N + b_1) & \vdots & g(W_{\tilde{N}} . X_N + b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}} \qquad (4.8)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N} \times m} \qquad (4.9)$$

$$\boldsymbol{T} = \begin{bmatrix} T_1^T \\ \vdots \\ T_{\tilde{N}}^T \end{bmatrix}_{N \times m} \qquad (4.10)$$

Algorithm for basic architecture of extreme learning machine, given the number of hidden nodes and hidden node activation functions

Step 1: Randomly assign W_i , the weights between the hidden nodes and the input nodes and b_i, the threshold of hidden layer

Step 2: Calculate the hidden layer output matrix H.

Step 3: Calculate the output linear layer weights using $\beta = H^{-1}\boldsymbol{T}$ where,

H^{-1} is the Moore–Penrose generalized inverse of the matrix H. Thus the smallest norm least-squares solution of the above linear system is obtained and this solution is unique.

4.3.2 MODELLING

The following algorithm explains the procedure for development of prediction model for forecasting ground motion parameter using ELM.

ALGORITHM:

For each of the ground motion parameter (Y), with input F, V, M, D perform the following steps:

Step 1: Assign the hidden nodes with sigmoidal activation function $g(x) = {}^{1}/_{(1+e^{-\lambda x})}$

Step 2: randomly assign an initial value (i) for the number of hidden neurons

Step 3: assign random weights for hidden neurons and threshold

Step 4: the hidden layer output matrix H, is calculated as explained in section 4.3.1.

Step 5: the output layer weights are calculated using Moore-Penrose generalization.

Step 6: iterate steps 3 to 5 for about 20-40 times, to obtain the best ELM prediction model. The best prediction model is the one having the highest correlation coefficient (R) with minimum MSE.

Step 7: increment the number of hidden neurons (i) by 1

Step 8: Repeat Steps 3 to 7 until MSE becomes approximately constant.

Step 9: the best prediction model based on ELM for forecasting parameter Y has 'i' number of hidden nodes.

The following flowchart explaining the above algorithm is given in Fig. 4.2.

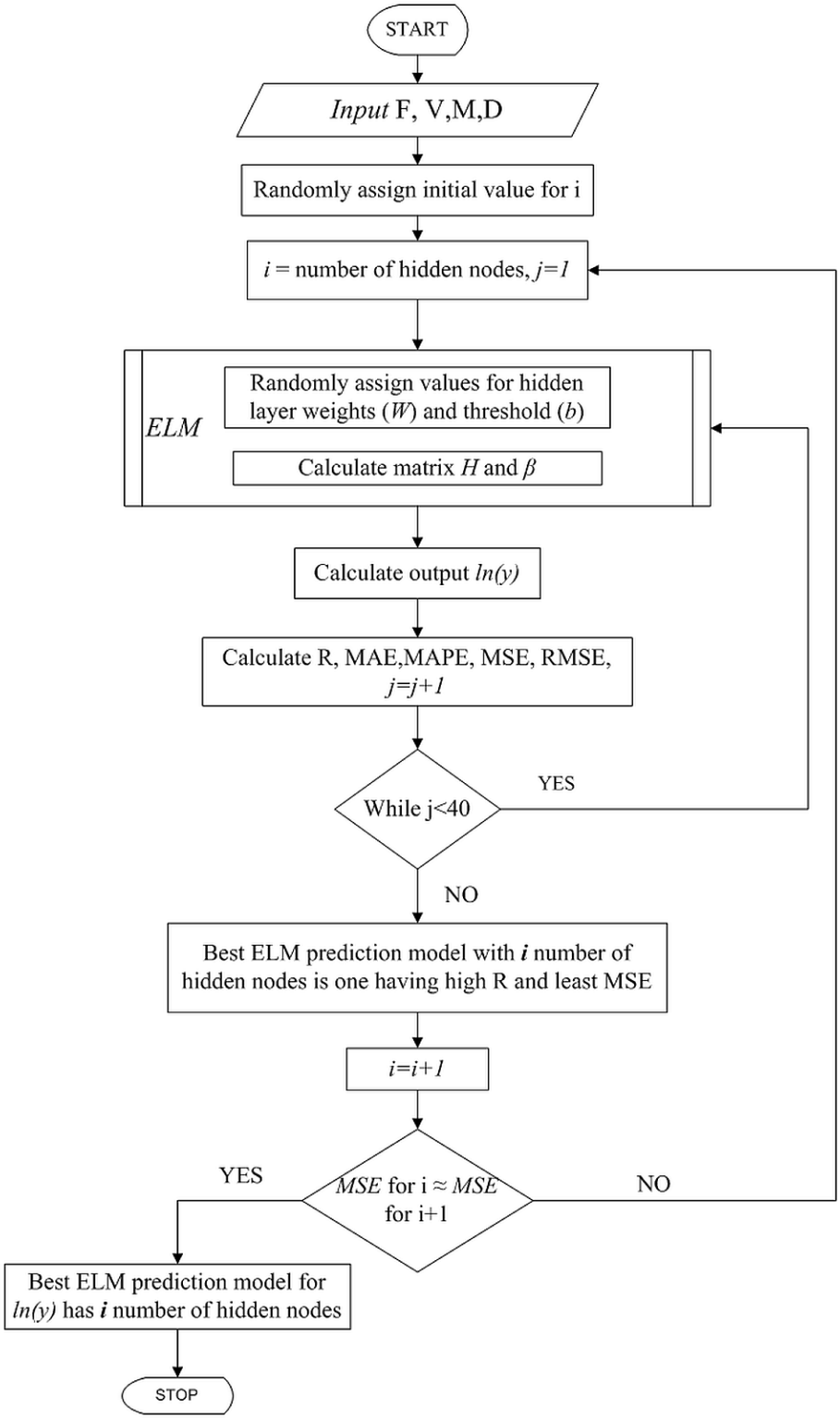

Fig. 4.2 Flowchart for ELM prediction model

4.3.3 RESULTS ANALYSIS

4.3.3.1 RESUTLS OBTAINED BY THE PROPOSED ELM MODEL

Figs. 4.3-4.5 shows the results obtained by the proposed prediction model based on ELM for forecasting PGA, PGV and PGD respectively. The results are tabulated in Table 4.7. It is observed that as the numbers of hidden nodes are increased, the error percentage decreases, hence the prediction accuracy of the model increases. It is also observed that for higher number of hidden nodes (>50), there is only a slight change in the accuracy. The best prediction model is the one having the least error percentage with fair correlation coefficient (R). Hence, from Table 4.4 is concluded that the PGA prediction model based on ELM has 80 hidden nodes. Similarly, justifying results in Table 4.5, it could be said that for the PGV prediction model based on ELM has 80 hidden nodes. Analyzing Table 4.6, it is observed that the error percentage remains constant irrespective of varying the number of hidden nodes. Thus PGD prediction model based on ELM has 50 hidden nodes.

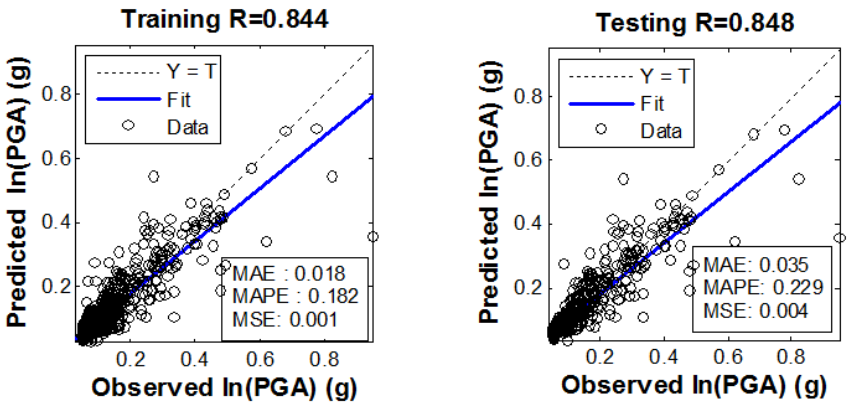

Fig. 4.3:Predicted Vs Observed value of PGA (Training Data and Testing Data)

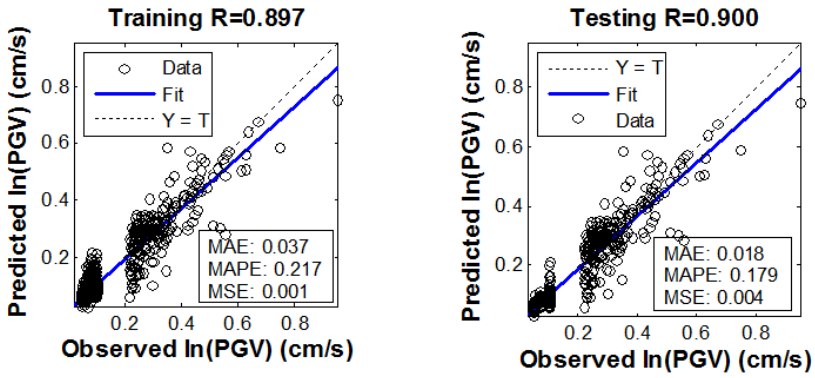

Fig. 4.4: Predicted Vs Observed value of PGV (Training Data and Testing Data)

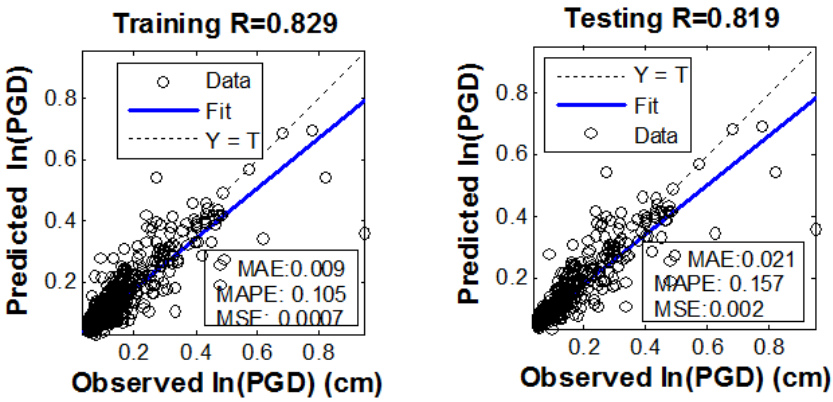

Fig. 4.5: Predicted Vs Observed value of PGD (Training Data and Testing Data)

Table 4.4: PGA prediction model with different number of hidden nodes

Criteria	Hidden nodes=20		Hidden nodes=30		Hidden nodes=40		Hidden nodes=50		Hidden nodes=60		Hidden nodes=80	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.7484	0.7815	0.7811	0.8005	0.8025	0.8183	0.8243	0.8351	0.8321	0.8387	0.8444	0.8486
MAE	0.0251	0.0434	0.0230	0.0414	0.0217	0.0389	0.0200	0.0368	0.0194	0.0366	0.0183	0.0351
MAPE	0.2671	0.2857	0.2390	0.2725	0.2239	0.2497	0.2026	0.2403	0.1961	0.2379	0.1824	0.2290
MSE	0.0021	0.0062	0.0018	0.0056	0.0017	0.0051	0.0015	0.0046	0.0014	0.0045	0.0014	0.0043

Table 4.5: PGV prediction model with different number of hidden nodes

Criteria	Hidden nodes=20		Hidden nodes=30		Hidden nodes=40		Hidden nodes=50		Hidden nodes=60		Hidden nodes=80	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.8019	0.8459	0.8417	0.8652	0.8682	0.8776	0.8770	0.8854	0.8882	0.8900	0.8974	0.9003
MAE	0.0265	0.0453	0.0253	0.0417	0.0216	0.0396	0.0212	0.0394	0.0201	0.0384	0.0369	0.0184
MAPE	0.2727	0.2647	0.2662	0.2483	0.2126	0.2226	0.2112	0.2320	0.1991	0.2223	0.2170	0.1791
MSE	0.0024	0.0063	0.0019	0.0054	0.0016	0.0048	0.0015	0.0045	0.0014	0.0043	0.0013	0.0040

Table 4.6: PGD prediction model with different number of hidden nodes

Criteria	Hidden nodes=20		Hidden nodes=30		Hidden nodes=40		Hidden nodes=50		Hidden nodes=60	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.7507	0.7688	0.7858	0.7772	0.7951	0.7838	0.8064	0.7899	0.8095	0.7893
MAE	0.0127	0.0227	0.0101	0.0205	0.0100	0.0205	0.0096	0.0204	0.0091	0.0207
MAPE	0.1650	0.1871	0.1216	0.1557	0.1209	0.1576	0.1139	0.1582	0.1050	0.1568
MSE	0.0009	0.0030	0.0008	0.0028	0.0007	0.0028	0.0007	0.0027	0.0007	0.0027

Table 4.7: Developed prediction model based on ELM

Criteria	PGA		PGV		PGD	
	Training	Testing	Training	Testing	Training	Testing
R	0.8444	0.8486	0.8974	0.9003	0.8295	0.8193
MAE	0.0183	0.0351	0.0369	0.0184	0.0091	0.0207
MAPE	0.1824	0.2290	0.2170	0.1791	0.1050	0.1568
MSE	0.0014	0.0043	0.0013	0.0040	0.0007	0.0027

4.3.3.2 COMPARISON WITH EXISTING MODELS

Tables 4.8-4.12 shows the comparison of the proposed ELM based prediction model for forecasting ground motion parameter with the existing benchmark models mentioned in Table 3.1 and the existing GMPEs on the database.

In Table 4.8 the developed prediction model based on ELM is compared with the GP / OLS model proposed by Gandomi et al [45]. It is observed that prediction model based on ELM

is better than the GP/OLS model. The accuracy is comparatively better with lesser RMSE, MAE and MAPE. In the table, where the values are not mentioned, it is denoted by n/a.

In Table 4.9, the ANN/SA model proposed by Alavi and Gandomi [4] is compared with the developed prediction model. Although, the prediction accuracy of the developed model is comparable to that of the ANN / SA model, the developed ELM based prediction model could be considered comparatively a better model due to much lower MAE and MAPE values. Moreover, it is observed that the developed prediction model is computationally faster as it gives the result with minimum MSE in 30 epochs with 108.12 seconds, whereas the ANN/SA model requires 364 epochs with 240 seconds to attain the minimum MSE value. The ANN/ SA architecture has a single hidden layer with 8 nodes and is implemented using Neural-Lab program version 3.1. The major drawback of this ANN/SA model is the high computational time due to simulated annealing.

The developed model is compared with the MEP model proposed by Alavi et al [5] in Table 4.10. The precision of the ELM based prediction model is higher than MEP model. In Table 4.11, the developed ELM model is compared to the GP/SA model proposed by Mohammadnejad et al [100]. It is observed that the GP/SA model has high error percentage, although the precision accuracy is comparable. Thus the ELM model could be considered better.

Table 4.8: Comparison of ELM prediction model with GP/OLS model [45]

	PGA				**PGV**				**PGD**			
Criteria	**ELM**		**GP/OLS**		**ELM**		**GP/OLS**		**ELM**		**GP/OLS**	
	Train	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
R	**0.845**	**0.848**	**0.836**	**0.811**	**0.897**	**0.900**	**0.822**	**0.813**	**0.829**	**0.819**	**0.836**	**0.811**
MAE	**0.018**	**0.035**	**0.478**	**0.488**	**0.037**	**0.018**	**n/a**	**0.506**	**0.009**	**0.020**	**n/a**	**n/a**
MAPE	**0.182**	**0.229**	**n/a**	**n/a**	**0.217**	**0.179**	**0.512**	**n/a**	**0.114**	**0.158**	**0.660**	**0.681**
RMSE	**0.037**	**0.065**	**0.836**	**0.637**	**0.063**	**0.036**	**0.649**	**0.637**	**0.026**	**0.052**	**0.850**	**0.901**

Table 4.9: Comparison of ELM prediction model with ANN/SA model [4]

	PGA				PGV				PGD			
Criteria	**ELM**		**ANN/SA**		**ELM**		**ANN/SA**		**ELM**		**ANN/SA**	
	Train	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
R	**0.845**	**0.848**	**0.869**	**0.855**	**0.897**	**0.900**	**0.867**	**0.874**	**0.829**	**0.819**	**0.870**	**0.869**
MAE	**0.018**	**0.035**	**0.30**	**0.46**	**0.037**	**0.018**	**0.34**	**0.45**	**0.009**	**0.020**	**0.62**	**0.62**
MAPE	**0.182**	**0.229**	**0.14**	**0.13**	**0.217**	**0.179**	**1.06**	**2.17**	**0.114**	**0.158**	**1.74**	**1.66**

Table 4.10: Comparison of ELM prediction model with MEP model [5]

	PGA				PGV				PGD			
Criteria	**ELM**		**MEP**		**ELM**		**MEP**		**ELM**		**MEP**	
	Train	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
R	**0.845**	**0.848**	**0.842**	**0.834**	**0.897**	**0.900**	**0.837**	**0.828**	**0.829**	**0.819**	**0.846**	**0.840**
MAE	**0.018**	**0.035**	**0.363**	**0.697**	**0.037**	**0.018**	**0.402**	**0.726**	**0.009**	**0.020**	**0.733**	**0.829**
RMSE	**0.037**	**0.065**	**0.602**	**0.624**	**0.063**	**0.036**	**0.634**	**0.671**	**0.026**	**0.052**	**0.856**	**0.899**

Table 4.11: Comparison of ELM prediction model with GP/SA model [100]

	PGA				PGV				PGD			
Criteria	**ELM**		**GP/SA**		**ELM**		**GP/SA**		**ELM**		**GP/SA**	
	Train	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
R	**0.845**	**0.848**	**0.833**	**0.839**	**0.897**	**0.900**	**0.833**	**0.837**	**0.829**	**0.819**	**0.847**	**0.854**
MAPE	**0.182**	**0.229**	**0.158**	**0.143**	**0.217**	**0.179**	**1.27**	**2.35**	**0.114**	**0.158**	**1.61**	**1.68**
RMSE	**0.037**	**0.065**	**0.617**	**0.616**	**0.063**	**0.036**	**0.645**	**0.648**	**0.026**	**0.052**	**0.845**	**0.846**

The developed model is further compared with three other GMPE models, namely Ambraseys et al model [6], Campbell-Bozorgnia model [29] and Smit et al model [142] in Table 4.12. From Table 4.12, it is clearly observed that the developed ELM based prediction model is better than the existing GMPE models due to lower percentage of the mean absolute error.

Table 4.12: Comparison of ELM prediction model with GMPEs

Model	**Mean absolute error percentage**		
	PGA	**PGV**	**PGD**
ELM based prediction model	**0.229**	**0.018**	**0.158**
Campbell and Bozorgnia [29]	**0.93**	**0.78**	**5.73**
Ambraseys et al [6]	**0.95**	**n/a**	**n/a**
Smit et al [142]	**14.58**	**n/a**	**n/a**

4.4 COMPARISON OF ELM WITH SLFN

In this section, the developed neural network based prediction models are further explored in terms of computational time. In this section, the ELM is compared with SLFN, which validates the claim of preferring ELM over ANN.

To the ELM based prediction model is compared with different training algorithms of single hidden layer feedforward neural network (SLFN), having the same number of hidden nodes as that of the ELM prediction model. Table 4.13 represents the computational time taken by the different training algorithms for the prediction of ground motion parameter. It is observed that the maximum computational time is taken by gradient descent backpropagation (traingd) training algorithm (105.01 seconds) followed by Levenberg-Marquardt backpropagation (trainlm) (72.71 seconds) and BFGS quasi-Newtonbackpropagation (trainbfg) (62.35 seconds) whereas ELM takes the least time (3.6 seconds).

Table 4.14 compares the prediction accuracy of the ELM model with the different training algorithms for SFLN. It is observed that the trainlm and trainbfg training algorithm of SLFN gives satisfactory prediction accuracy from among the other training algorithms, but the overall performance of trainlm and trainbfg are inferior compared to ELM in terms of prediction accuracy as well as computational time. Hence it is clearly proved that the ELM prediction model gives faster and better prediction accuracy.

Table 4.13: Comparison of computational time of different training algorithm

Learning Algorithm		Learning Time (seconds)
ELM	**Extreme learning machine**	**3.6**
trainbfg	**BFGS quasi-Newton backpropagation**	**62.35**
traincgb	**Powell -Beale conjugate gradient backpropagation**	**31.24**
traincgf	**Fletcher-Powell conjugate gradient backpropagation**	**49.14**
traincgp	**Polak-Ribiere conjugate gradient backpropagation**	**50.32**
traingd	**Gradient descent backpropagation**	**105.01**
trainlm	**Levenberg-Marquardt backpropagation**	**72.71**
trainrp	**Resilient backpropagation (Rprop)**	**12.98**
trainscg	**Scaled conjugate gradient backpropagation**	**28.17**

Table 4.14: Comparison of different training algorithm of SLFN with ELM

	PGA		PGV		PGD	
Function Name	**Training R**	**Testing R**	**Training R**	**Testing R**	**Training R**	**Testing R**
ELM	**0.8444**	**0.8486**	**0.8974**	**0.9003**	**0.8295**	**0.8193**
trainbfg	**0.838**	**0.748**	**0.814**	**0.806**	**0.779**	**0.838**
traincgb	**0.554**	**0.569**	**0.851**	**0.813**	**0.724**	**0.668**
traincgf	**0.71**	**0.726**	**0.854**	**0.799**	**0.593**	**0.621**
traincgp	**0.712**	**0.597**	**0.661**	**0.727**	**0.769**	**0.795**
traingd	**0.1752**	**0.0812**	**0.241**	**0.23**	**0.092**	**0.045**
trainlm	**0.816**	**0.834**	**0.904**	**0.826**	**0.795**	**0.867**
trainrp	**0.699**	**0.529**	**0.858**	**0.769**	**0.569**	**0.649**
trainscg	**0.735**	**0.73**	**0.724**	**0.673**	**0.741**	**0.863**

4.5 CONCLUSION

In this chapter, a prediction model is proposed for the forecasting of ground motion parameter based on neural network based learning. The algorithm used is ELM, which has many advantages over ANN, the most popular neural network based learning model. The comparison of the developed ELM prediction model with the existing models and the traditional training algorithm of neural networks validates the higher generalization ability and faster prediction speed of the ELM model. Moreover, the architecture and training of the proposed model is much simpler as the only computational overhead is in the determination of pseudo inverse matrix for calculating the weights of the output layer. The developed ELM based prediction model overcomes the drawback of ANN and the existing benchmark prediction models in this domain, such as extensive computational time. Thus, ELM provides better precision accuracy in lesser computational time.

Like all other algorithm, ELM also has few drawbacks. Though neural network based learning methods such as ANN are popular due to its ease of implementation, it suffers a major drawback which prevents it from being an efficient algorithm. Artificial neural network works on the principle of empirical risk minimization and hence the best ANN architecture is the one having minimum training error. This leads to two major issues of overfitting and local minima. ANN also has an overhead as its computational complexities are dependent on the dimension of the input space. Overcoming all these drawbacks, a new learning method, support vector machines (SVM) based on kernel method is gaining popularity. SVM provides global, unique and sparse solution to problems. It is also less prone to problem of overfitting as it works on structural risk minimization. The next chapter details a prediction model proposed based on SVM.

CHAPTER 5

PREDICTION MODEL BASED ON KERNEL METHODS LEARNING

5.1 INTRODUCTION

In machine learning, support vector machines (SVM) are the most popular algorithm based on kernel methods. SVM is gaining popularity compared to other soft computing techniques such as artificial neural networks (ANN) as it overcomes the bottleneck issues of ANN such as overfitting and local minima. SVM is generally applied for classification problems. When applied for regression problem, (Burges [26] , Smola and Scholkopf [141]) it is termed as support vector regression (SVR).

In this chapter, prediction models are developed using three variations of SVR learning algorithms, namely ε-SVR, ν-SVR and LS-SVR for forecasting peak ground acceleration (PGA). Using three kernel functions, namely linear kernel, polynomial kernel and RBF kernel for each of the three learning algorithms, 7 prediction models are developed. All the 7 models are compared in terms of prediction accuracy, error percentage and overfitness to obtain the best prediction model. The chapter is organized as follows. Section 5.2 explains the basics of the three learning algorithms ε-SVR, ν-SVR and LS-SVR. The experimental environment used for the modelling is detailed in section 5.3. Section 5.4 analyzes the results obtained and validates the efficacy of the model by comparing it with existing benchmark models. The chapter is concluded by section 5.5 which further explores the significance of the proposed prediction model.

5.2 SUPPORT VECTOR REGRESSION (SVR) Learning Algorithm

Support Vector Machine (SVM) works on principle of VC theory by Vapnik and Chervonenkis [152], which is purely based on statistical learning. The problem is solved by equating it to a quadratic programming problem with inequality constraint. The least squares

support vector machine (LS-SVM) is a variation of SVM, which uses equality constraint. When SVM is applied to a regression problem, it is termed as support vector regression (SVR).

The SVR algorithm is explained as follows:

Consider a given set of training data $\{(x_1,y_1),(x_2,y_2)\ldots.(x_n,y_n)\}$, where $x_i \in R^d$, $y_i \in R$, i =1….n. The training data x_i from the input space X is mapped onto a feature space Q, using kernel function k, such as $k(x_i, \acute{x}_i) = \langle \varphi(x_i), \varphi(\acute{x}_i) \rangle$. For simplicity, we begin with linear functions. Let f be the linear function having the form as in Eq. 5.1

$$f(x) = w^t x + b = \langle w, x \rangle + b, \ where\ w \in X, b\ \in R, \ \langle .,. \rangle \text{ denote dot product} \qquad (5.1)$$

The following subsections explain about the three variations of SVR algorithm used in this study.

5.2.1 ε- SUPPORT VECTOR REGRESSION (ε-SVR)

In ε- SV regression by Vapnik [153], the function *f(x)* is calculated such that it is flat, but at the same time has a maximum deviation of ε. Hence the permissible error band for the function is [-ε, ε]. The function *f(x)* attains flatness when the value of *w* is small and to obtain the minimum value for w is to obtain the minimum norm solution which is $\|w\|^2 = \langle w, w \rangle$. Hence reformulating the problem as a feasible convex optimization problem we obtain Eq.5.2

$$\text{minimize : } \frac{1}{2}\|w\|^2 \text{ subject to } \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon \end{cases} \qquad (5.2)$$

The above equation could be again reformulated to include infeasible constraints of the problem by introducing slack variables.

$$\text{minimize : } \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\epsilon_i + \epsilon_i^*) \quad \text{subject to } \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon + \epsilon_i \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon + \epsilon_i^* \\ \epsilon_i, \epsilon_i^* \geq 0 \end{cases} \qquad (5.3)$$

where, C is the tradeoff between permissible error ε and $\|w\|$.

The aim is to minimize the empirical risk value given as $E = \frac{1}{N}\sum_{i=1}^{N}|y_i - \acute{y}_i|_\varepsilon$ where $|y_i - \acute{y}_i|_\varepsilon$ is ε -insensitive loss function as shown in Fig. 5.1, such that $|y_i - \acute{y}_i|_\varepsilon = \begin{cases} 0, & |y_i - \acute{y}_i| \leq \varepsilon \\ |y_i - \acute{y}_i| - \varepsilon, & otherwise \end{cases}$ where y_i and $\acute{y}_i$ are target and predicted value respectively. This formulation is also known as linear ε-insensitive loss regression. (Cortes and Vapnik [37], Vapnik [154]).

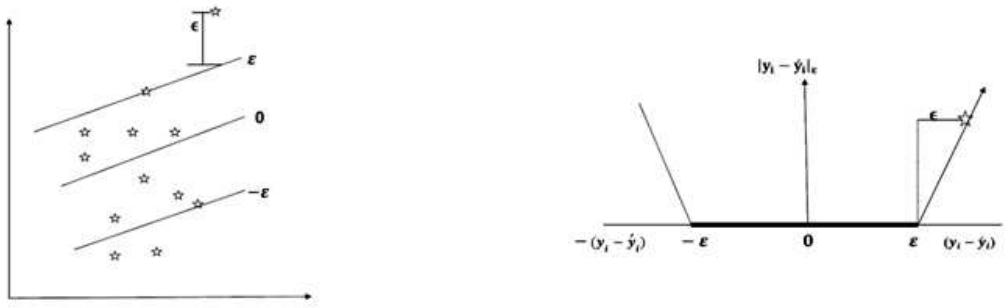

Fig. 5.1: Support Vector Regression with ε insensitive loss function

Eq.(5.3) could be solved using dual formulation involving Lagrange multipliers. Hence the objective function of Eq.(5.3) is replaced by corresponding Lagrange function. Let L be the Lagrangian function and $\alpha_i, \alpha_i^*, \beta_i, \beta_i^*$ be the corresponding Lagrange multipliers or dual variables. Hence the Eq.(5.3) is replaced as follows:

$$L = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\epsilon_i + \epsilon_i^*) - \sum_{i=1}^{n}(\alpha_i\epsilon_i + \alpha_i^*\epsilon_i{}^*) - p - q \qquad (5.4)$$

where, $p = \sum_{i=1}^{n}\beta_i\,(\varepsilon + \epsilon_i - y_i + \langle w, x_i\rangle + b)$, $q = \sum_{i=1}^{n}\beta_i^*(\varepsilon + \epsilon_i^* + y_i - \langle w, x_i\rangle - b)$,

$$\alpha_i, \alpha_i^*, \beta_i, \beta_i^* \geq 0.$$

For all feasible solution of the primal and dual variables for the convex optimization objective function, there exist a saddle point. (Mangasarian [91]). At optimality, the gap between the primal and dual objective function decreases (strong duality theorem (Theorem 6.4.3,Bazaraa et al [22]).

Taking the partial derivative of Lagrangian function L with respect to the primal variables $(w, b, \epsilon_i, \epsilon_i^*)$ we obtain Eq.(5.5).

$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{n}(\beta_i - \beta_i^*)x_i$, $\frac{\partial L}{\partial b} = \sum_{i=1}^{n}(\beta_i^* - \beta_i)$, $\frac{\partial L}{\partial \epsilon_i} = C - \beta_i - \alpha_i$, $\frac{\partial L}{\partial \epsilon_i^*} = C - \beta_i^* - \alpha_i^*$ (5.5)

Hence, at optimality by theory, $\frac{\partial L}{\partial w} = 0$, $\frac{\partial L}{\partial b} = 0$, $\frac{\partial L}{\partial \epsilon_i} = 0$, $\frac{\partial L}{\partial \epsilon_i^*} = 0$. Thus eliminating dual variables, we obtain $\alpha_i = C - \beta_i$, $\alpha_i^* = C - \beta_i^*$ (5.6)

Substituting Eq.(5.5,5.60) in Eq. (5.4), the dual optimization problem is formulated as follows.

$$\text{Maximize} : \begin{cases} -\frac{1}{2}\sum_{i=i}^{n}\sum_{j=1}^{n}(\beta_i - \beta_i^*)(\beta_j - \beta_j^*)\langle x_i, x_j\rangle \\ -\varepsilon\sum_{i=1}^{n}(\beta_i + \beta_i^*) + \sum_{i=1}^{n} y_i\,(\beta_i - \beta_i^*) \end{cases} \qquad (5.7)$$

$$\text{subject to } \sum_{i=1}^{n}(\beta_i - \beta_i^*) \text{ and } \beta_i, \beta_i^* \in [0, C].$$

Rewriting Eq.(5.5) we obtain,

$$w = \sum_{i=1}^{n}(\beta_i - \beta_i^*)x_i \qquad (5.8)$$

Substituting in Eq.(5.8) in Eq.(5.1) we obtain the solution to the function *f(x)* as shown in Fig. 5.2.

$$f(x) = \sum_{i=1}^{n}(\beta_i - \beta_i^*)\langle x_i, x\rangle + b \qquad (5.9)$$

w is linear combination of training inputs x_i in the input space *X*. Hence this is called the support vector expansion, where the function complexity is dependent on the number of support vectors rather than the dimensionalities of the input space X. *b* is computed using KKT conditions (Kuhn and Tucker [85]).

At KKT optimality condition ($constraint.dual\ variable = 0$). Thus we obtain

$$(C - \beta_i)\epsilon_i = 0,(C - \beta_i^*)\epsilon_i^* = 0, \beta_i(\varepsilon + \epsilon_i - y_i + \langle w, x_i\rangle + b) = 0,$$
$$\beta_i(\varepsilon + \epsilon_i - y_i + \langle w, x_i\rangle + b) = 0 \quad (5.10)$$

From Eq. (10), $\forall(x_i, y_i)\ with\ \beta_i, \beta_i^* = C$, there exists points in space *X*, that lie outside the [-ε, ε] band and there can never be a condition such that the set of dual variables $[\beta_i, \beta_i^*]$ is simultaneous zero as $\beta_i\beta_i^* = 0$.

Hence the solution for 'b'(Keerthi et al [77]) is as follows.

$$b = y_i - \langle w, x_i\rangle - \varepsilon, \quad \forall\, \beta_i \in (0, C) \quad (5.11)$$
$$b = y_i - \langle w, x_i\rangle + \varepsilon, \quad \forall\, \beta_i{}^* \in (0, C) \quad (5.12)$$

Thus the function estimation for ε-SVR model for is

$$f(x) = \sum_{i=1}^{n}(\beta_i - \beta_i^*)\langle x_i, x\rangle + b \text{ , with } b = \begin{cases} y_i - \langle w, x_i\rangle - \varepsilon, & \forall\, \beta_i \in (0, C) \\ y_i - \langle w, x_i\rangle + \varepsilon, & \forall\, \beta_i{}^* \in (0, C) \end{cases} \quad (5.13)$$

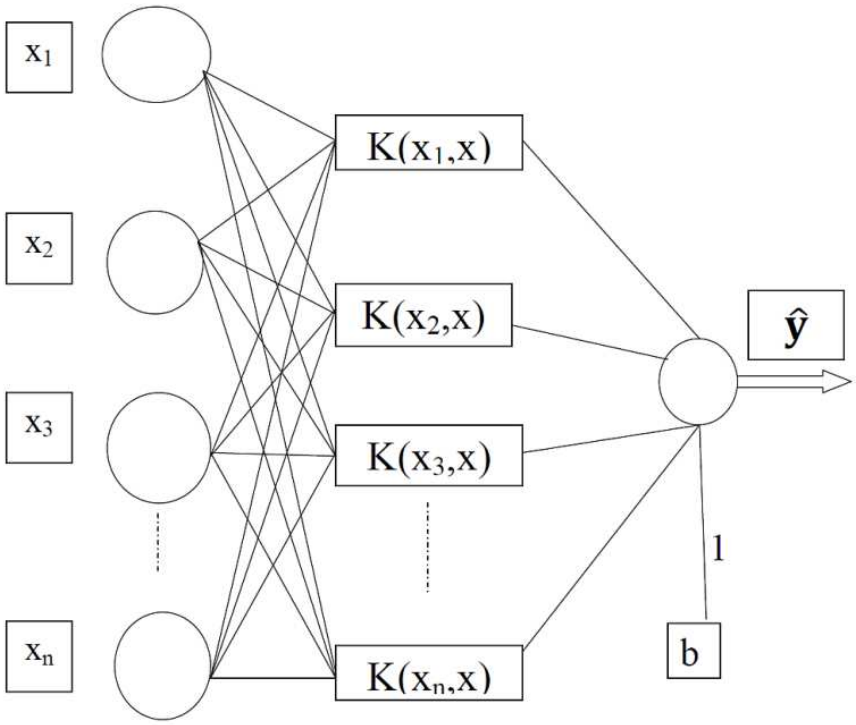

Fig. 5.2: Support vector regression architecture

5.2.2 ν- SUPPORT VECTOR REGRESSION (ν-SVR)

Another version of SVR, ν-SVR was proposed by (Scholkopf et al [129]) which uses parameter ν of range (0, 1]. It is similar to ε-SVR with ε itself considered as a parameter to have a control on the count of support vector. The formulation for ν-SVR is similar to ε-SVR, with a slight change. Thus Eq.(5.3) is reformulated for ν-SVR as

$$\text{minimize : } \frac{1}{2}\|w\|^2 + C\nu\varepsilon + C\sum_{i=1}^{n}(\epsilon_i + \epsilon_i^*) \quad \text{subject to } \begin{cases} y_i - \langle w, x_i\rangle - b \le \varepsilon + \epsilon_i \\ \langle w, x_i\rangle + b - y_i \le \varepsilon + \epsilon_i^* \\ \epsilon_i, \epsilon_i^* \ge 0, \varepsilon \ge 0 \end{cases} \quad (5.14)$$

The dual formulation of ν-SVR is similar to ε-SVR, as given in Eq.(5.7), with only change in the constraint. The new constraint for ν-SVR is $\beta_i, \beta_i^* \in [0, Cv]$.

5.2.3 LEAST SQUARE SUPPORT VECTOR REGRESSION (LS-SVR)

The function estimation problem using least squares support vector is formulated in this section. The linear model of LS-SVR is similar to Eq.(5.1) (Ye and Xiong [159]). Consider a given set of training data $\{(x_1,y_1),(x_2,y_2)\ldots.(x_n,y_n)\}$, where $x_i \in R^d$, $y_i \in R$, i =1….n. Let f be the linear function, by Eq. (5.1) we have $f(x) = w^t x + b$ .Let δ, γ denote the error vector and the column vector respectively. Keeping all the notations in the above section, the regression estimation problem is formulated as

$$\text{minimize: } \frac{1}{2}w^t H w + \frac{1}{2}\delta^t\delta \text{ subject to } K(x_i,x_i)w + \gamma b + \delta - y_i = 0 \quad (5.15)$$

The Lagrange multipliers method used in section 3.1.1 is used to solve convex optimization problem Eq.(5.15).

Let L be the Lagrangian function and β_i be the corresponding Lagrange multiplier. Let x_i and y_i be denoted as x,y, Hence the corresponding Lagrangian function for Eq.(5.15) is as follows:

$$L = \frac{1}{2}w^t H w + \frac{1}{2}\delta^t\delta - \beta^t[\,K(x,x)w + \gamma b + \delta - y] \qquad (\delta,\alpha) \in R^d \quad (5.16)$$

As per the theory explained above section, at optimality the constraints are as follows:

$$\left.\frac{\partial L}{\partial w^t} = Hw - K^t\beta = 0, \frac{\partial L}{\partial b} = \gamma^t\beta = 0\ ,\ \frac{\partial L}{\partial \delta} = \delta - \beta = 0, \frac{\partial L}{\partial \alpha^t} = Kw + \gamma b + \delta - y = 0\right\} \quad (5.17)$$

Solving we get,

$$w = K^t H^{-1}\beta\ ,\ \delta = \beta \quad (5.18)$$

Combining Eq.(5.16) – Eq.(5.18) we obtain,

$$\begin{bmatrix} 0 & \gamma^t \\ \gamma & KH^{-1}K^t + I \end{bmatrix}\begin{bmatrix} b \\ \beta \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix} \quad (5.19)$$

Let H=K, assuming that the kernel K being symmetric positive definite, Eq.(5.19) reduces as

$$\begin{bmatrix} 0 & \gamma^t \\ \gamma & K + I \end{bmatrix}\begin{bmatrix} b \\ \beta \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix} \quad (5.20)$$

Eq.(5.20) is analogous to standard LS-SVM (Sukyens et al [148]).

For solving K, the assumption is H=I. Hence the restrictions of symmetry, positive definiteness, semi definiteness, and continuity on K is removed.

Hence we obtain,

$$w = K^t\beta\ ,\ \begin{bmatrix} 0 & \gamma^t \\ \gamma & KK^t + I \end{bmatrix}\begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix} \quad (5.21)$$

In Eq.(5.21) KK^t is positive semi definite, with no restrictions on K thus making Eq.(5.22) linearally solvable.

The LS-SVR model for function estimation is $f(x) = KK^t\beta + \gamma b$ (5.22)

5.3 MODELLING

The SVR algorithm is used to develop a predictive model for forecasting ground motion parameter, peak ground acceleration (PGA). The algorithms are implemented and tested on C and MATLAB R2012 b platform on a PC with processor Intel(R) core(TM) i3-3220 and 4GB RAM. The Libsvm package (Chang [31]) is used to implement ε-SVR and ν-SVR. LS-SVR is

implemented using ls-svmlab version 1.8 (Brabanter [20]). The principle for parameter selection for SVR is given by Cherkassy and Ma [34]. In this study, the following kernel function and parameters are selected.

The kernel function $k(x_i, x_j)$ defined in libsvm package is as follow:

Linear kernel : $k(x_i, x_j) = x_i^T x_j$ (5.23)

Polynomial Kernel: $k(x_i, x_j) = (gamma * x_i^T * x_j + r)^{degree}$ (5.24)

RBF Kernel: $k(x_i, x_j) = e^{-gamma * \|x_i - x_j\|^2}$ (5.25)

The kernel function $k(x_i, x_j)$defined in ls-svmlab package is as follow:

Linear kernel: $k(x_i, x_j) = x_i^T x_j$ (5.26)

RBF Kernel: $k(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}}$ (5.27)

- ε-SVR uses parameters C [0, ∞) and ε [0, inf) to apply a penalty to the optimization for points which were not correctly predicted. There is no penalty associated with points which are predicted within distance ε from the actual value. By decreasing ε , closer fitting to the data is obtained.
- ν-SVR uses parameters C [0, ∞) and ν [0,1]. The ε penalty parameter was replaced by ν. ν represents an upper bound on the fraction of training samples which are errors poorly predicted and a lower bound on the fraction of samples which are support vectors. ε and ν are versions of the penalty parameter.
- The other two parameters used are C (cost) and gamma. The cost represents the penalty associated with errors larger than epsilon. Increasing cost value gives closer fitting to data. Parameter gamma controls the shape of the separating hyperplane. Increasing gamma usually increases number of support vectors.

5.3.1 ε-SVR PARAMETERS

The ε-SVR algorithm is implemented with linear and RBF kernels. The tolerance error for termination criteria is set at 0.1. A Grid search algorithm with cross validation is used to find the parameters of RBF kernel. Hence the value of C thus obtained is 128. The value of gamma is set as 0.25(1/number of features) and ε as 0.1.

5.3.2 ν-SVR PARAMETERS

The ν-SVR algorithm is implemented with linear, polynomial and RBF kernels. The parameters such as tolerance error for termination criteria, gamma and C are same to that of ε-SVR algorithm. The value for ν is set as 0.5. The degree is set as 3 for polynomial kernel.

5.3.3 LS-SVR PARAMETERS

The LS-SVR algorithm is implemented with linear and RBF kernels. The parameter gamma here is called a regularization parameter, which determines the tradeoff between the minimization of training error and the smoothness of the estimated function. The simulated coupling method is used to obtain the best value for gamma. For linear kernel the value obtained is 1.52 and for RBF kernel gamma is 2.321. For RBF kernel sig2 is additional parameter which represents the variance of the kernel function. The value obtained is 0.19.

5.4 RESULTS ANALYSIS AND DISCUSSION

This section evaluates the results obtained. For the better justification of the obtained results, this section, is further divided into subsections. The subsection 5.4.1 describes the result obtained by developed model. The subsection 5.4.2 compares the developed model with other existing models on the same database. The subsection 5.4.3 further evaluates the learning effectiveness of the algorithms.

5.4.1 RESULTS OBTAINED FROM ε-SVR, ν-SVR and LS-SVR PREDICTION MODELS.

Tables 5.1-5.3 shows the result obtained from ε-SVR, ν-SVR and LS-SVR PGA Prediction model for both training and testing data. Table 5.4 shows the result obtained from the developed models for NGA WEST 2 data.

Table 5.1: ε-SVR Prediction Model for PGA

	ε-SVR			
Kernel	**Linear**		**RBF**	
	Train	**Test**	**Train**	**Test**
R	**0.4836**	**0.5932**	**0.6958**	**0.7643**
MAE	**0.0625**	**0.0733**	**0.0531**	**0.0580**
MAPE	**0.8223**	**0.7117**	**0.7160**	**0.5581**
MSE	**0.0057**	**0.0102**	**0.0044**	**0.0064**

It is observed from Table 5.1-5.3 that the best prediction model for forecasting PGA is LS-SVR RBF kernel prediction model followed by ν-SVR RBF kernel prediction model and ε SVR RBF kernel prediction model. Though the accuracy of the other models is not in the acceptable range [143], it is observed that the error percentages are low. The best prediction model is LS-SVR RBF kernel prediction model as it gives good prediction accuracy with low error percentage and also gives a fair correlation for NGA WEST 2 data, which is a dataset outside the training database.

Table 5.2: v-SVR Prediction Model for PGA

	v-SVR					
Kernel	**Linear**		**Polynomial**		**RBF**	
	Train	**Test**	**Train**	**Test**	**Train**	**Test**
R	**0.5058**	**0.5515**	**0.5364**	**0.5312**	**0.7313**	**0.7629**
MAE	**0.0246**	**0.0622**	**0.0247**	**0.0622**	**0.0239**	**0.0460**
MAPE	**0.1956**	**0.2931**	**0.1934**	**0.2978**	**0.2460**	**0.2765**
MSE	**0.0040**	**0.0151**	**0.0040**	**0.0150**	**0.0023**	**0.0079**

Table 5.3: LS-SVR Prediction model for PGA

	LS-SVR			
Kernel	**Linear**		**RBF**	
	Train	**Test**	**Train**	**Test**
R	**0.5194**	**0.5946**	**0.8719**	**0.8700**
MAE	**0.0315**	**0.0578**	**0.0149**	**0.0316**
MAPE	**0.3424**	**0.3651**	**0.1367**	**0.1899**
MSE	**0.0034**	**0.0113**	**0.0011**	**0.0037**

Table 5.4: Prediction model for Testing NGA West 2 data

	ε-SVR		ν-SVR			LS-SVR	
Kernel	**Linear**	**RBF**	**Linear**	**Polynomial**	**RBF**	**Linear**	**RBF**
R	**0.5715**	**0.6585**	**0.7061**	**0.5737**	**0.7171**	**0.6803**	**0.8140**
MAE	**0.0393**	**0.0537**	**0.0073**	**0.0076**	**0.0148**	**0.0160**	**0.0060**
MAPE	**0.5867**	**0.8308**	**0.1059**	**0.1056**	**0.2180**	**0.2461**	**0.0853**
MSE	**0.0022**	**0.0036**	**0.0001**	**0.0001**	**0.0003**	**0.0004**	**0.0001**

5.4.2 COMPARISON WITH EXISTING MODELS

Table 5.5 gives the comparison of the best developed PGA prediction model with other existing models on the same database. The LS-SVR RBF kernel prediction model is compared to four models, namely ANN/SA (a hybrid model of artificial neural network coupled with simulated annealing (Alavi and Gandomi 2011 [4])), GP/OLS (a hybrid model of genetic programming coupled with orthogonal least squares (Gandomi et al. 2011 [45])), MEP (multi expression programming) Alavi et al 2011 [5])) and GP/SA (a hybrid model of genetic programming coupled with simulated annealing (Mohammadnejad et al 2012 [100])). In Table 5.5 where the corresponding values is missing is denoted by symbol n/a. It is observed that the prediction accuracy of the developed model is better compared to other existing models. Another important observation is that the error percentage of the developed model is much less compared to all the existing benchmark models.

Table 5.5: LS-SVR prediction model with different existing model (n/a denote value not mentioned)

Criteria	LS-SVR model (RBF kernel)		ANN/SA [4]		GP/OLS [45]		MEP [5]		GP/SA [100]	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.8719	0.8700	0.869	0.855	0.836	0.811	0.842	0.834	0.833	0.839
MAE	0.0149	0.0316	0.30	0.46	0.478	0.488	0.363	0.697	n/a	n/a
MAPE	0.1367	0.1899	0.14	0.13	n/a	n/a	n/a	n/a	0.158	0.144
MSE	0.0011	0.0037	n/a	n/a	0.358	0.406	0.362	0.389	0.381	0.380

Table 5.6: Comparison of developed prediction model with GMPEs

Model	Criteria MAPE
LS-SVR model (RBF kernel)	0.18
Campbell and Bozorgnia [29]	0.93
Ambraseys et al [6]	0.95
Smith et al [142]	14.58

In Table 5.6, the LS-SVR RBF kernel prediction model is compared with existing GMPEs. The comparison is done for the testing data based on the criteria MAPE. It is observed that the LS-SVR RBF kernel prediction model gives the least error percentage, compared to the existing GMPEs. The developed model has comparatively lesser computational overhead with better precision.

5.4.3 COMPARING THE LEARNING EFFECTIVENESS OF THE SVR ALGORITHM

Tables 5.1-5.6 shows the results and the comparison between the prediction models. The comparison is done based on the criteria as given in section 3.6. Table 5.7 compares all the developed models along with the existing benchmark models in terms of accuracy, testing error and overfitness (Zhong et al [161]). The testing error is indicator of overfitness. The measure of overfitness is calculated as in Eq. (5.28). Thus the overall authenticity of prediction precision of the model is collectively given by the accuracy, testing error and the Overfitness measure.

$$\text{overfitness} = \frac{\|\mathbf{error(testing)} - \mathbf{error(training)}\|}{\mathbf{error(training)}} \qquad (5.28)$$

From Table 5.7, it is observed that the best prediction model is LS-SVR RBF kernel model as it gives a high prediction accuracy with lesser overfitness measure.

It is observed that the ANN/SA model [4], though having slightly higher precision than GP/OLS [45] and GP/SA model [100] in terms of accuracy (correlation coefficient), the testing error is comparable and the overfitness measure is better for GP based hybrid models. Thus the performances of the three existing models are comparable. The MEP model [5] clearly shows that the overfitness measure is very high. Similarly, though the precision accuracy of ν-SVR RBF kernel model in terms of correlation coefficient (R) is slightly less, the testing error is comparable to the 3 existing models ANN/SA, GP/OLS and GP/SA, with much lesser overfitness measure. Thus the overall performance of ν-SVR RBF kernel prediction model could be assumed to be satisfactory. Similarly the overall performance of ε-SVR RBF kernel prediction model could also be satisfactory.

Another important inference from Table 5.7 is that it is observed that the measure of overfitness for all models of SVR is much lesser than the hybrid model of artificial neural network coupled with simulated annealing [4] and MEP model [5]. Hence it further validates the claim of support vector machines being less prone to overfitting than an artificial neural network with better generalization.

Table 5.7: Comparison of models for overall performance

Model	Kernel	Accuracy	Testing Error	Overfitness
ε-SVR	linear	0.593	0.0733	0.173
	RBF	0.764	0.058	0.092
ν-SVR	linear	0.551	0.0622	0.49
	Polynomial	0.531	0.0622	0.54
	RBF	0.763	0.046	0.12
LS-SVR	linear	0.595	0.0578	0.067
	RBF	0.870	0.0316	0.38
GP/SA		0.839	0.144	0.21
GP/OLS		0.811	0.49	0.24
MEP		0.834	0.69	0.92
ANN/SA		0.855	0.46	0.53

5.5 CONCLUSION

In this chapter, a kernel based learning method is used to develop a predictive model for forecasting PGA, a ground motion parameter. The results obtained are analyzed meticulously to validate the efficacy of the developed learning machine using SVR. The study shows that the predictive model developed using kernel methods based learning is efficient compared to existing benchmark models in this domain.

Among the 7 developed models, the LS-SVR model with RBF kernel is precise and has a better generalization with lesser measure of overfitting. Hence it is also proved that SVR has better generalization than neural networks. In the following chapter another learning method is introduced to develop a predictive model for forecasting ground motion parameter.

CHAPTER 6

PREDICTION MODEL BASED ON TREE BASED LEARNING

6.1 INTRODUCTION

In this chapter, a prediction model for forecasting ground motion parameter is developed using decision tree. Decision Tree learning, a popular tool used in data mining is gaining popularity a predictive tool for supervised learning in various fields. A Decision tree is a predictive learning method to develop a tree like model used to predict a target, based on set of input features. The model developed is termed as regression tree, when the predicted target values take real continuous values. Learning by decision trees has an overhead over other learning techniques because of the representation ability of the model which makes it intuitive and adaptable. The interpreting of the results becomes easy in tree representation. Moreover, since tree representation is hierarchical in nature, the modelling is relatively easier compared to linear modelling in case of a large number of input features. The chapter is organized as follows. Section 6.2 explains the algorithm for developing regression tree. The experimental environment used for the modelling is detailed in section 6.3. Section 6.4 analyzes the results obtained and validates the efficacy of the model by comparing it with existing benchmark models. The chapter is concluded by section 6.5 which further explores the significance of the proposed prediction model.

6.2 REGRESSION TREE LEARNING ALGORITHM

Linear regression has a major drawback, of not being able to obtain a single predictive equation for modelling systems for data, having many input features with high non linearity among features. Alternative approach would be non linear regression modelling regression tree, where the entire dataset is partitioned or divided into subsets in a structure similar to that of a tree using divide and conquer greedy algorithm. The subsets are again divided as recursive partition, until the smallest subset could be easily represented. A tree is a collection of nodes and branches.

The start node is termed as root and the terminal node is termed as leaf node. A parent node will have two child nodes, left node and right node. The size of the tree is the number of terminal nodes.

The three main steps involved in the development of a regression tree [88] are the split process, partition process and the pruning process.

i. Split Process: the choice for the splitting variable and the splitting points is done such that, for any node n and split s, there should be gain for some measuring criteria $\varphi(n, s)$.
ii. Partition Process: this process includes the binary partition of dataset to further recursive partitioning, including setting of the termination criteria, to decide when a node to be a terminal node.
iii. Pruning Process: this process includes combining the internal nodes of Tree, so as to get optimal tree length. Very large tree might lead to overfitting of data.

Splitting Criteria: Suppose the regression tree (T) is to be modelled for Y having dataset $\{(x_1,y_1),(x_2,y_2)\ldots.(x_n,y_n)\}$. Let p be the leaf node and the data points belonging to leaf node p be $\{(x_i,y_i),(x_2,y_2)\ldots.(x_c,y_c)\}$. The model for leaf node p is $\hat{y} = \frac{1}{c}\sum_{i=1}^{c} y_i$, the mean of dependent variables belonging to node p. Let S (sum of squared errors) be the splitting criteria, then $\acute{S} = \sum_{p\varepsilon leaf nodes(T)} \sum_{i\varepsilon p}(y_i - \hat{y})^2$. Let P is the number of leaf nodes for T, then $\acute{S} = \sum_{p\varepsilon leaf nodes(T)} P v_p$, where $v_p$is the within-leave variance of leaf p. The splitting for each node is done so as to minimize the S.

Hence the **algorithm** for Regression Tree could be summarized as follows:

Step 1: start with a single node with entire data points. For this node calculate $\hat{y}$ and $\acute{S}$.

Step 2: split the dataset belonging to the node into two half planes R_1, R_2 such that, for splitting variable j and for split s, $R_1(j,s) = \{X|X_j \leq s\}$ and $R_2(j,s) = \{X|X_j > s\}$. For this node calculate $\hat{y}$ and $\acute{S}$.

Step 3: repeat step 2 for all independent variables and the pair of (j,s) having minimum $\acute{S}$, is selected as best pair for split.

Step 4: Repeat step 2 and 3 for growing tree, until termination criteria is met or when all independent variables at a particular node gives same value.

Step 5 : termination criteria is set such as, $\acute{S}$ falls beyond a particular set threshold δ or when the data points belonging to particular node p is less than q points.

Step 6: to obtain the optimal tree length, pruning is done using cross-validation.

Step 6 (a): Let T_1 be $\acute{S}$ for all leaf nodes of a common parent. Let T_2 be $\acute{S}$, by considering the parent node as leaf node. If $T_2 < T_1$, then leaf nodes are pruned and the parent node becomes leaf node.

6.3 MODELLING

The prediction model is developed based on decision tree learning using the algorithm explained in section 6.2. Three different models are developed for forecasting PGA, PGV and PGD respectively. The algorithm is implemented on C and MATLAB. The best tree model will have optimal tree length. Pruning is a technique in machine learning which is used to reduce the length of decision tree by removing irrelevant branching. In this study, 10 fold cross validation technique is used to prune the tree, so as to obtain the optimal regression tree prediction model. It is observed from Figs. 6.1-6.3 that by cross validation the minimum MSE is obtained for number of leaf nodes 20, 10, 10 respectively for forecasting PGA, PGV and PGD. Hence the tree prediction model for forecasting PGA, PGV and PGD has length 20,10,10 respectively.

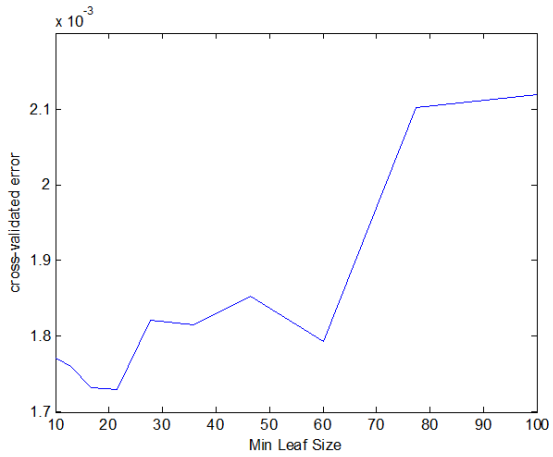

Fig. 6.1: Pruning of Regression Tree model for PGA

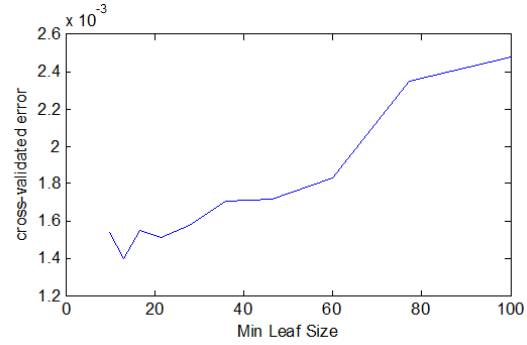

Fig. 6.2: Pruning of Regression Tree model for PGV

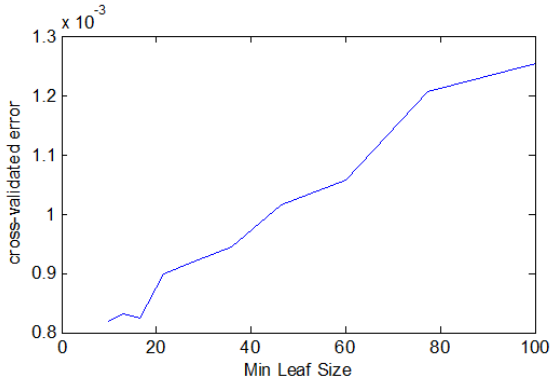

Fig.6.3: Pruning of Regression Tree model for PGD

6.4 RESULT ANALYSIS AND DISCUSSION

The results obtained from the PGA prediction model is shown in Fig. 6.4. The comparison of the developed PGA prediction model with the existing benchmark models is tabulated in Table 6.1. It is observed that the proposed prediction model has good precision with much lesser error percentage, for forecasting all the ground motion parameter. Although the precision accuracy is comparable with the existing benchmark models, it is clearly observed from

the Table 6.1 that the percentage error is much lesser for the proposed model. Thus, it validates that overfitting of data is comparatively lesser in the proposed model.

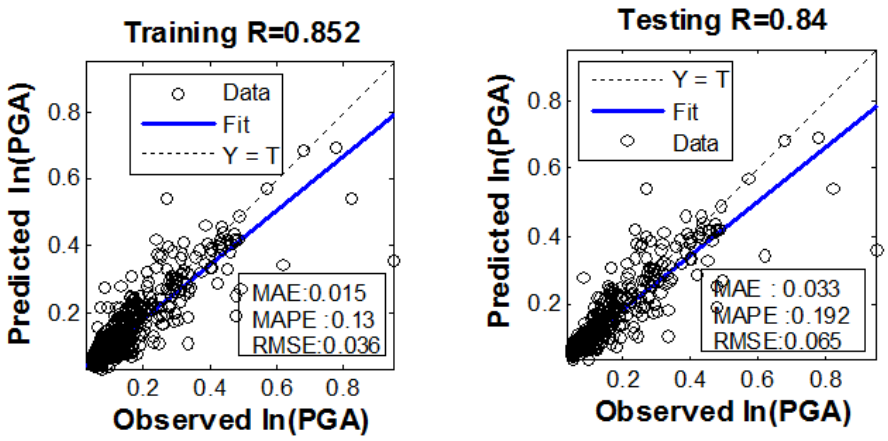

Fig. 6.4: Predicted Vs Observed value of PGA for developed prediction model.

Table 6.1: Comparison of developed prediction model for PGA with existing models

Criteria	Decision Tree		ANN/SA [4]		GP/OLS [45]		MEP [5]		GP/SA [100]	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.8528	0.840	0.869	0.855	0.836	0.811	0.842	0.834	0.833	0.839
MAE	0.0149	0.033	0.30	0.46	0.478	0.488	0.363	0.697	n/a	n/a
MAPE	0.1314	0.192	0.14	0.13	n/a	n/a	n/a	n/a	0.158	0.144
MSE	0.001	0.004	n/a	n/a	0.358	0.406	0.362	0.389	0.381	0.380
RMSE	0.036	0.065	n/a	n/a	0.836	0.637	0.602	0.624	0.617	0.616

The Fig.6.5 shows the results obtained from the PGV prediction model. The comparison of the developed PGV prediction model with the existing benchmark models is tabulated in Table 6.2. It is observed that the proposed prediction model has good precision with much lesser error percentage, for forecasting all the ground motion parameter compared with the existing benchmark models.

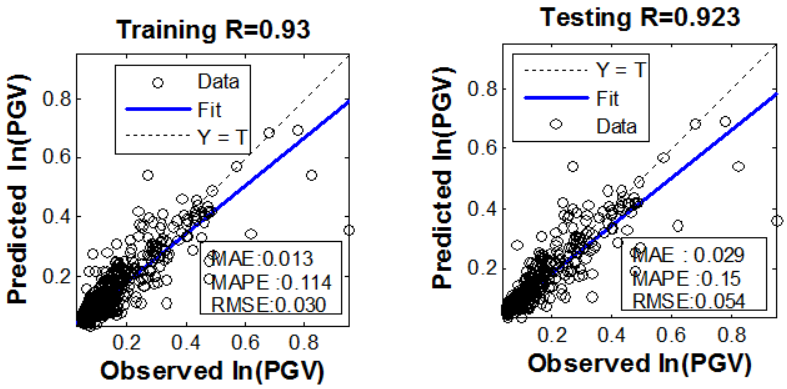

Fig. 6.5: Predicted Vs Observed value of PGV for developed prediction model.

Table 6.2: Comparison of developed prediction model for PGV with existing models

Criteria	Decision Tree		ANN/SA [4]		GP/OLS [45]		MEP [5]		GP/SA [100]	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.9280	0.9234	0.867	0.874	0.822	0.813	0.837	0.828	0.833	0.837
MAE	0.0130	0.0292	0.34	0.45	n/a	0.506	0.402	0.726	n/a	n/a
MAPE	0.1140	0.1502	1.06	2.17	0.512	n/a	n/a	n/a	1.27	2.35
MSE	0.0009	0.0029	n/a	n/a	0.421	0.405	n/a	n/a	n/a	n/a
RMSE	0.030	0.054	n/a	n/a	0.649	0.637	0.634	0.671	0.645	0.648

The results obtained from the PGD prediction model is shown in Fig. 6.6. The comparison of the developed PGA prediction model with the existing benchmark models is tabulated in Table 6.3. Although the precision accuracy is comparable with the existing benchmark models, it is clearly observed from the Table 6.3 that the percentage error is much lesser for the proposed model.

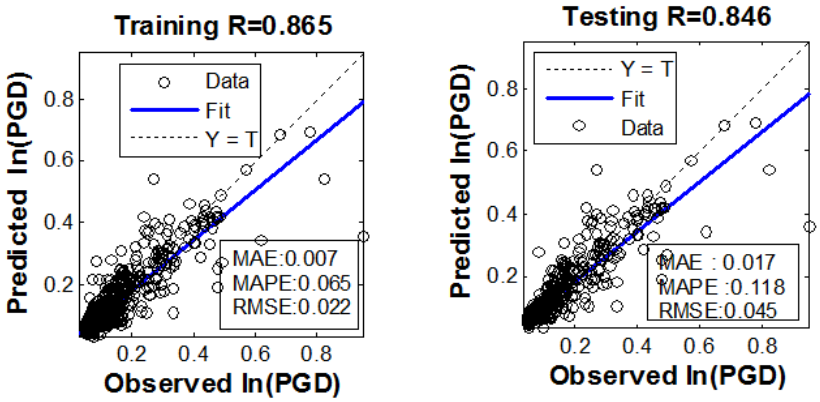

Fig.6.6: Predicted Vs Observed value of PGD for developed prediction model.

Table 6.3: Comparison of developed prediction model for PGD with existing models

Criteria	Decision Tree		ANN/SA [4]		GP/OLS [45]		MEP [5]		GP/SA [100]	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.8652	0.8460	0.870	0.869	0.836	0.811	0.846	0.840	0.847	0.854
MAE	0.0066	0.0175	0.62	0.62	n/a	n/a	0.733	0.829	n/a	n/a
MAPE	0.0655	0.1177	1.74	1.66	0.660	0.681	n/a	n/a	1.61	1.68
RMSE	0.022	0.045	n/a	n/a	0.850	0.901	0.856	0.899	0.845	0.846

The table 6.4 tabulates the results obtained by the proposed model for NGA West data. The results show that the proposed model gives a fair generalization, although results are not in the acceptable range [143].

Table 6.4: Comparison of developed prediction model for NGAWest data

Criteria	PGA	PGV	PGV
R	0.7004	0.7417	0.7492
MAE	0.0065	0.0058	0.0048
MAPE	0.0876	0.0884	0.0715
RMSE	0.01	0.022	0.022

The proposed prediction models are further compared to existing GMPEs in Table 6.5. It is clearly observed that proposed models are better than existing GMPEs which validates the drawback of linear regression.

Table 6.5: Comparison of developed prediction model with GMPEs (n/a denote value not mentioned)

Model	Mean absolute error percentage		
	PGA	PGV	PGD
Tree based prediction model	0.192	0.15	0.12
Campbell and Bozorgnia [29]	0.93	0.78	5.73
Ambraseys et al [6]	0.95	n/a	n/a
Smit et al [142]	14.58	n/a	n/a

6.5 CONCLUSION

In this study, simple regression tree learning is used to develop the predictive model for forecasting ground motion parameters. Tree based regression models are at advantage due to the faster prediction as the interpretation from tree structure is easier. It is clearly observed from the results that the proposed prediction model is efficient compared to the existing benchmark models in the domain. Now there is major difference between clustering and regression tree. In clustering, we try to cluster in such a way so as to maximize the information gained by the cluster for the independent variable X. In regression trees, the branching or dividing is done in such a manner that the leaf node gives maximum information about the dependent variable Y. Hence the branching starts at a root node in a greedy search manner, with binary splitting the root node into two daughter nodes. The daughter nodes are further split in a binary fashion. At each split there is gain in information about Y.

Although the growing of tree structure is simple, it has a major drawback such as the tree structure is purely dependant on the training data and hence a small change in the training data would result in obtained a different tree structure. Moreover a conflict arises in choosing the independent variables for node, when multiple independent variables stand equally good and the selection is by mere chance. Thus, it also affects the structure of the tree.

Hence it could be concluded that although the proposed model is efficient compared to the existing models, more efficious models could be developed. In the next chapter, an hybrid architecture of neuro fuzzy system is used to develop predictive model.

CHAPTER 7

PREDICTION MODEL BASED ON ADAPTIVE NEURO FUZZY INFERENCE SYSTEM (ANFIS)

7.1 INTRODUCTION

In this chapter, adaptive network based fuzzy interference system (ANFIS) is used to model a prediction system for forecasting peak ground acceleration (PGA). It is an integrated hybrid architecture of fuzzy logic with neural networks such that the knowledge gained by the fuzzy logic is used by the learning algorithm of neural network. The initial fuzzy model is derived with the rules from the data, and the neural network learns and trains the rules to get the final model. The hybrid learning algorithm used in ANFIS consists of gradient descent and LSE (least square estimate). The chapter is organized as follows. Section 7.2 details the architecture and the learning algorithms of the hybrid ANFIS model. Section 7.3 analyzes the results obtained and the efficacy of the model is validated by comparing it with existing benchmark models. The chapter is concluded by section 7.4 which discusses the advantages and disadvantages of the proposed ANFIS prediction model.

7.2 ADAPTIVE NEURO FUZZY INFERENCE SYSTEM

An adaptive neuro fuzzy inference system (ANFIS) is a hybrid architecture of adaptive networks and fuzzy inference system, with a hybrid learning algorithm (HLA). This section is further divided into two subsections to clearly comprehend this hybrid system. The subsection 7.2.1 explains the architecture of the hybrid ANFIS system and the subsection 7.2.2 explains the learning algorithm.

7.2.1 ADAPTIVE NEURO FUZZY INFERENCE SYSTEM ARCHITECTURE

The basic architecture of adaptive network based fuzzy inference system (ANFIS) is shown in Fig.7.1. It is a hybrid architecture of Takagi and Sugeno`s Fuzzy inference system [Fig. 7.2] with adaptive network. Let Eqs. (7.1-7.2) represent the two rules of Takagi and Sugeno`s

type representing fuzzy inference system, such that x, y are the crisp set inputs and the A_i, B_i are the corresponding linguistic variables.

$$RULE1: If\ x\ is\ A_1\ and\ y\ is\ B_1 \quad THEN\ f_1 = p_1x + q_1y + r_1 \tag{7.1}$$

$$RULE2: If\ x\ is\ A_2\ and\ y\ is\ B_2 \quad THEN\ f_2 = p_2x + q_2y + r_2 \tag{7.2}$$

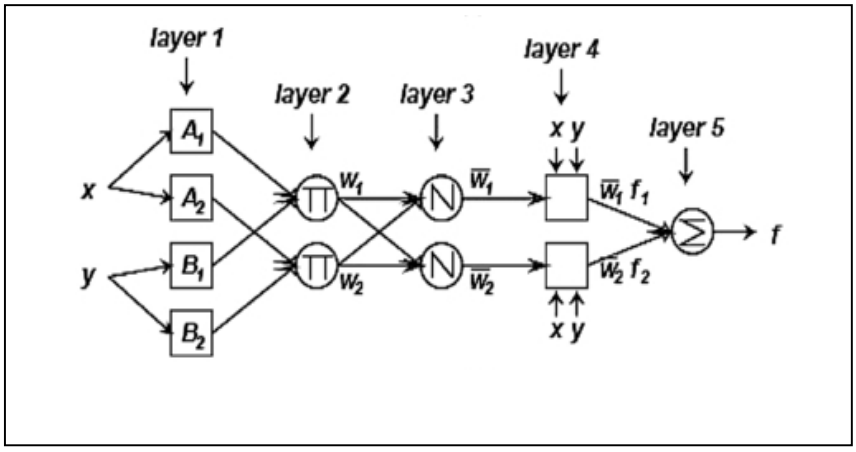

Fig. 7.1: ANFIS architecture

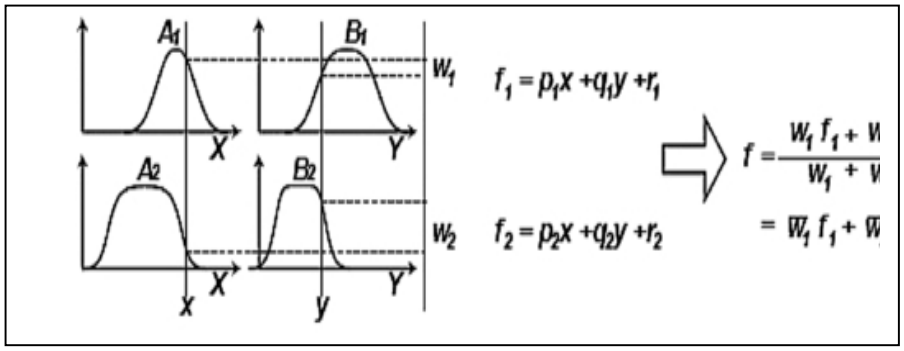

Fig. 7.2: First order Sugeno Fuzzy inference model

The Figure 7.1 shows the ANFIS architecture [70] and the figure 7.2 shows the inference mechanism and the defuzzification (f) of Takagi and Sugeno`s type fuzzy inference systems. An adaptive network is a multi layer feedforward network in which each node performs the designated node function. It basically consists of two types of nodes, namely adaptive node (represented by square) and fixed node (represented by a circle). An adaptive node has

parameters and the set of parameters for an adaptive network consist of all parameters of all adaptive nodes in the network. These parameters are updated based on gradient based learning algorithm during the training. Fixed nodes do not have any parameters. The entire ANFIS architecture consists of 5 layers and each layer is explained as follows:

i. Layer 1: All nodes in this layer are adaptive nodes, with the node function as given in Eq.(7.3).

$$O_i^1 = \mu_{A_i}(x) \qquad (7.3)$$

where x is the crisp input, A_i is the corresponding linguistic variable and i is the number of nodes in the layer. O_i^1 represents fuzzy membership value for A_i. Generally Bell shaped membership functions are used to provide smoothness in membership grade and flexibility in the core of membership functions. The mathematical representation of bell shape membership function is given in Eq. (7.4).

$$\mu_A(x) = \frac{1}{1+\left|\frac{x-c_i}{a_i}\right|^{2b_i}} \qquad (7.4)$$

where a_i, b_i, and c_i are position and shape deciding parameters respectively and are also called premise parameters.

ii. Layer 2: All nodes in this layer are fixed nodes. Hence all nodes perform the fixed operation of multiplying all the respective node inputs. The output of each node w_i in this layer represents the firing strength of the fuzzy rule [Eq. 7.5].

$$w_i = \mu_{A_i}(x)\mu_{B_i}(y), \quad i = 1,2 \qquad (7.5)$$

iii. Layer 3: All nodes in this layer are fixed nodes. Hence all nodes perform the fixed operation as given in Eq. 7.6. The output of this layer $\overline{w_i}$ is called normalized firing strength.

$$\overline{w_i} = \frac{w_i}{w_1 + w_2}, i = 1,2 \qquad (7.6)$$

iv. Layer 4: All nodes in this layer are adaptive nodes, with the node function as given in Eq.(7.7).

$$O_i^4 = \overline{w}_i f_i = \overline{w}_i (p_i x + q_i y + r_i) \quad (7.7)$$

where O_i^4 is the output of the layer, $\overline{w}_i$ is the layer 3 output, and p_i, q_i, r_i are called the consequent parameters.

v. Layer 5: This layer consists of single fixed node whose output *out* is calculated as in Eq. (7.8). *out* is the overall output of the architecture,which is the summation of all incoming signals.

$$out = f = \sum_i \overline{w}_i f_i = \overline{w}_1 (p_1 x + q_1 y + r_1) + \overline{w}_2 (p_2 x + q_2 y + r_2) \quad (7.8)$$

7.2.2 ADAPTIVE NEURO FUZZY INFERENCE SYSTEM LEARNING ALGORITHM

In this section the algorithm used for training the premise and consequent parameters is discussed. A hybrid learning algorithm, comprising of gradient descent and LSE (least square estimate) is used in ANFIS. The hybrid learning algorithm overcomes the drawbacks of traditional learning algorithm gradient method such as slow learning rate and local minima. The hybrid algorithm is a two pass algorithm in which the consequent parameters are determined using LSE in the first pass and in latter pass the premise parameters are determined using gradient descent. Table 7.1 shows the the two passes of the algorithm and the parameters that are trained during each pass.

Table 7.1: Description for 2 pass Hybrid Algorithm

Pass	Premise Parameters	Consequent Parameters	Value considered
Forward Pass	**Fixed**	**LSE**	**Output of Nodes**
Backward Pass	**Gradient Descent**	**Fixed**	**Error measure**

1. PASS 1 FORWARD PASS:

During this pass, the initial value for premise is assumed. Now keeping premise parameters as fixed, the consequent parameters are calculated using least square estimate. It is done so by allowing the input data and the node functional signals pass forward all nodes till the layer 4. Thus the output f [Fig. 7.1] could be equated as a linear combination of consequent parameters. Hence, by rearranging the terms in Eq. (7.8), we get Eq.(7.9).

$$f = (\bar{w}_1 x)p_1 + (\bar{w}_1 y)q_1 + (\bar{w}_1)r_1 + (\bar{w}_2 x)p_2 + (\bar{w}_2 y)q_2 + (\bar{w}_2)r_2 \quad (7.9)$$

Eq. (7.9) could further be written as $f = YZ$ (7.10) where, Z is an unknown vector whose value is to be calculated. Thus, if Y is invertible matrix Eq. (7.10) could be written as $Z = Y^{-1}f$ (7.11). Another alternative solution for Eq. (10) is by considering the Pseudo inverse of Z. Hence we obtain the solution as

$$Z = (Y^T Y)^{-1} Y^T f \quad (7.12)$$

2. PASS 2 BACKWARD PASS:

During this pass, the error measure is propagated backwards to update the premise parameters, keeping the consequent parameters constant. The updating of the premise parameters is based on the gradient descent method. Let training data has n values. Hence the overall error measure E is defined as $E_n = \sum_{k=1}^{n} E_k$ (7.13). Let ω be the parameter, β be the learning rate for the parameter. The Eq. (7.14) represents the overall error measure for the parameter. The parameter is updated as shown in Eq. (7.15). Thus the new update parameter value would be as shown in Eq. (7.16).

$$\frac{\partial E}{\partial \omega} = \sum_{k=1}^{n} \frac{\partial E_k}{\partial \omega} \quad (7.14)$$

$$\Delta\omega = -\beta \frac{\partial E}{\partial \omega} \quad (7.15)$$

$$\omega_{new} = \omega + \Delta\omega \quad (7.16)$$

7.3 RESULT ANALYSES AND DISCUSSION

In this section, the overall performance of the developed model is analyzed. The subsection 7.3.1 details the results obtained by the developed ANFIS prediction model. The comparison of the developed model with the existing benchmark models is discussed in the subsection 7.3.2.

7.3.1 ANFIS PREDICTION MODEL

Table 7.2 shows the results obtained for the ANFIS prediction model for membership function value 2. It is observed that the optimum results is obtained at epoch 50 because beyond this epoch the error percentage becomes a constant. The results obtained for ANFIS prediction model for membership function value 3 is given in Table 7.3. The best model is obtained for epoch 15 because with further increase in the number of epochs, there is no considerable change in the MAE, MAPE, MSE values as well as the R value. The value of membership function is not further incremented because further increase of the value of the membership function increases the computational complexities of the algorithm leading to a drastic increase of the learning time of the algorithm, hence making it slower.

Thus the ANFIS prediction model for forecasting the peak ground acceleration has membership function value 3 and the best MSE is obtained at epoch 15 in 60.25 seconds. The initial and final membership function of the four input variables of the developed prediction model is shown in Figs. 7.3 and 7.4 respectively. The Fig. 7.5 shows the correlation coefficient (R) and the error measure of the developed prediction model for PGA, separately for training and testing data sets. The Fig. 7.6 shows the results for NGA WEST 2 dataset.

Table 7.2: Comparison of results of ANFIS model with membership function =2

Criteria	Epoch =10		Epoch =20		Epoch =50		Epoch =100	
Run Time	1.6 s		3.01 s		7.30 s		14.49 s	
	Train	Test	Train	Test	Train	Test	Train	Test
R	0.841	0.844	0.855	0.854	0.8604	0.859	0.864	0.860
MAE	0.019	0.035	0.017	0.034	0.0164	0.0328	0.016	0.0327
MAPE	0.183	0.226	0.164	0.214	0.1573	0.2064	0.1522	0.2029
MSE	0.0014	0.0043	0.0013	0.0040	0.0012	0.0039	0.0012	0.0038
RMSE	0.037	0.066	0.036	0.063	0.035	0.062	0.035	0.061

Table 7.3: Comparison of results of ANFIS model with membership function =3

Criteria	Epoch =10		Epoch =15		Epoch =16		Epoch =20	
Run Time	44.8 s		60.25 s		65.9 s		83.35 s	
	Train	Test	Train	Test	Train	Test	Train	Test
R	0.883	0.878	0.887	0.884	0.889	0.885	0.892	0.889
MAE	0.0146	0.029	0.0144	0.0292	0.0144	0.029	0.0143	0.0287
MAPE	0.139	0.185	0.1372	0.1861	0.1367	0.187	0.136	0.1829
MSE	0.0010	0.0034	0.0009	0.0032	0.0009	0.0032	0.0009	0.0031
RMSE	0.032	0.058	0.032	0.057	0.030	0.056	0.039	0.056

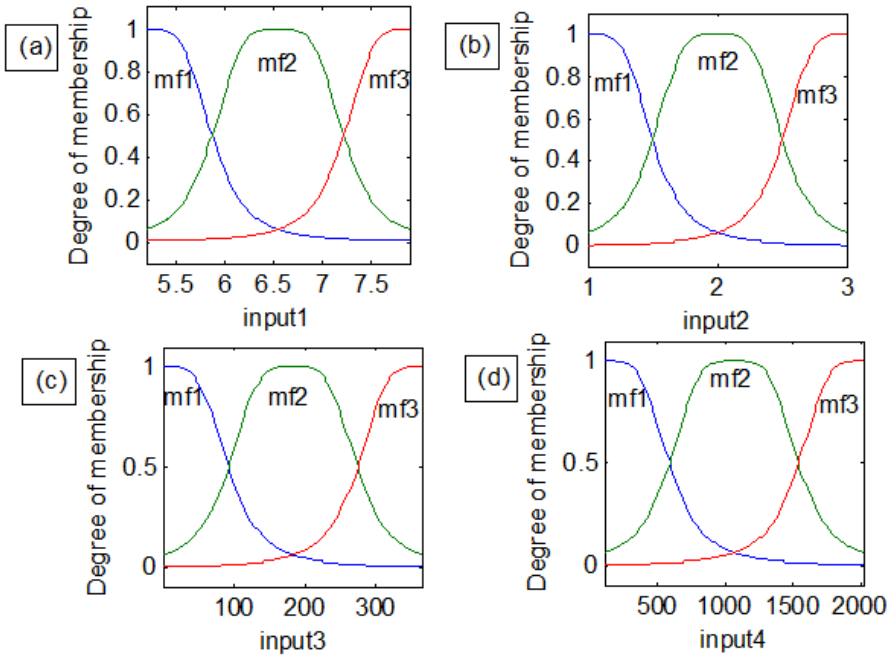

Fig. 7.3: Initial Membership function for ANFIS prediction model: (a) input1M (b) input2 F (c) input 3 D (d) input 4 V

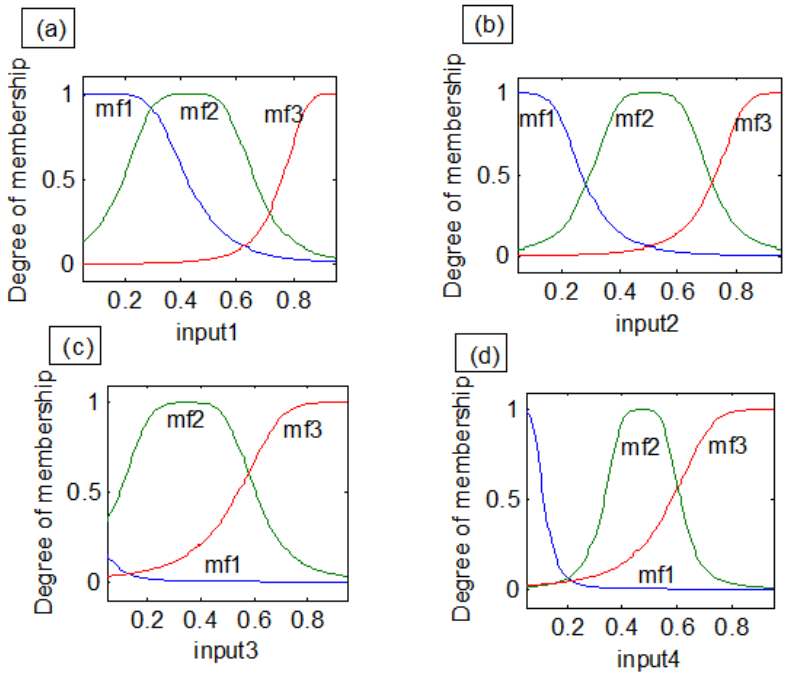

Fig. 7.4: Final Membership function for ANFIS prediction model: (a) input1M (b) input2 F (c) input 3 D (d) input 4 V

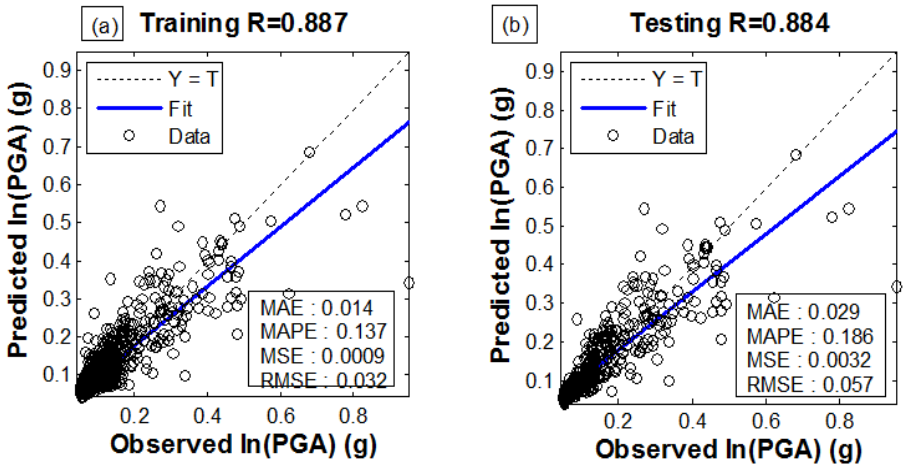

Figure 7.5: Predicted Vs Observed value of PGA (a) Training Data (b) Testing Data

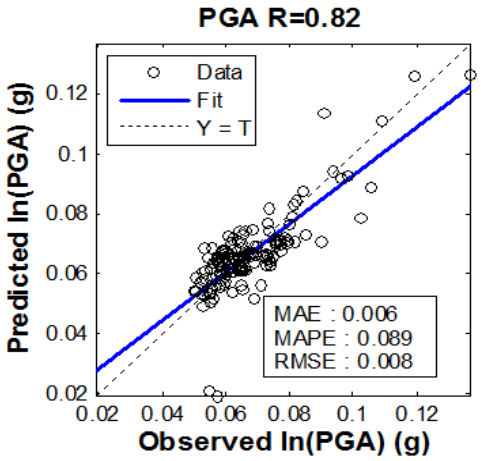

Figure 7.6: Predicted Vs Observed value of PGA

7.3.2 COMPARISON WITH EXISTING MODELS

Table 7.4 shows the comparison of the developed ANFIS prediction model with 4 other existing models in the same database, namely GP/OLS model [45], ANN/SA model [4], MEP model [5] and GP/SA model [100]. In the Table 7.4, the symbol n/a is used to denote the missing values. It is clearly observed that developed prediction model has better precision compared to the existing models. It is also observed that the developed prediction model is faster. The ANN/SA model [4] requires 364 epochs with 240 seconds to attain the minimum MSE value, whereas the developed prediction model attains best MSE in 60.25 seconds.

Table 7.4: Comparison of PGA with different existing models

Criteria	ANFIS Model mem_fn=3 Epoch=15		GP/OLS model [45]		ANN/SA model [4]		MEP model [5]		GP/SA [100]	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.88	0.88	0.836	0.811	0.869	0.855	0.842	0.834	0.833	0.839
MAE	0.01	0.03	0.48	0.48	0.30	0.46	0.363	0.697	n/a	n/a
MAPE	0.14	0.18	n/a	n/a	0.14	0.13	n/a	n/a	0.158	0.144
RMSE	0.032	0.057	0.836	0.637	n/a	n/a	0.602	0.624	0.617	0.616

To further validate the efficacy of the developed prediction model, it is compared to few existing GMPE models. There are numerous attenuation relationships for the prediction of principal ground motion parameter PGA and comparing with all of them is not feasible. Table 9 shows the comparison of the developed model with fewer GMPEs developed in the same database. The developed model is also compared with three other GMPE models, namely Campbell-Bozorgnia model [29], Ambraseys et al model [6] and Smit et al model [142]. The criterion for comparison is the error percentage in the prediction of the principal ground motion parameter PGA for the testing dataset (563 records). The result in Table 7.5 clearly approves the efficiency of the ANFIS prediction model.

Table 7.5: Comparison of PGA model with GMPEs

Model	Criteria MAPE
ANFIS prediction model	**0.14**
Campbell and Bozorgnia (2007)	**0.93**
Ambraseys et al (1996)	**0.95**
Smit et al (2000)	**14.58**

7.4 CONCLUSION

The results obtained show that proposed ANFIS prediction model is efficient compared to all existing hybrid models in the same database as well as the existing GMPE models. Although the precision of the developed model is good, it still could be improved. ANFIS architectures has a major drawback of high computational complexities with higher number of membership functions, which makes the algorithm slow. Hence a new novel neuro fuzzy technique, RANFIS (randomized ANFIS) is proposed and is explained in the next chapter.

CHAPTER 8

PREDICTION MODEL BASED ON NOVEL NEURO FUZZY LEARNING MACHINE, RANDOMIZED ANFIS (RANFIS)

8.1 INTRODUCTION

In this chapter, a new learning algorithm called RANFIS is used to develop a prediction model for forecasting the parameters of ground motion. RANFIS is an improved conventional ANFIS, overcoming the computational overhead of the hybrid learning algorithm (HLA) algorithm used for the learning in ANFIS. RANFIS is a hybrid algorithm combining the advantages of random assumption of the weight vector for the hidden layer in feedforward nets and the adaptive network based fuzzy inference system (ANFIS). The proposed RANFIS algorithm is simpler and faster than the conventional ANFIS [69,114,115].

The chapter is organized as follows. Section 8.2 details the architecture and the learning algorithms of the RANFIS model. The modelling parameters are detailed in section 8.3. Section 8.4 analyzes the results obtained and the efficacy of the model is validated by comparing it with existing benchmark models. The chapter is concluded by section 8.5 which discusses the advantages and disadvantages of the proposed ANFIS prediction model.

8.2 RANDOMIZED ANFIS

In randomized single layer feedforward neural network (SLFN), hidden layer weights and biases are randomly chosen. The linear output layer weights are determined analytically, reducing the computational cost. To solve this classification problem, consider a conventional SLFN with hidden nodes and activation function g(x). Consider a training dataset with N samples (x_i, t_i) where $\boldsymbol{x_i} = [\boldsymbol{x_{i1}}, \boldsymbol{x_{i2}}, ..., \boldsymbol{x_{in}}]^{\boldsymbol{T}}$ and $t_i = [t_{i1}, t_{i2}, ..., t_{im}]^T$.

The output of the linear layer o_j can be obtained as

$\sum_{i=1}^{\tilde{N}} \beta_i g_i(x_j) = \sum_{i=1}^{\tilde{N}} \beta_i g(w_i x_j + b_i) = o_j \quad for \quad j = 1,....N$ where $w_i = [w_{i1}, w_{i2}, ..., w_{in}]^T$ is the weight vector between the input nodes and the j[th] hidden node, $\beta_i = [\beta_{i1}, \beta_{i1}, .. \beta_{im}]$ is the weight vector for the linear output nodes and b_i is the threshold of the i^{th} hidden node. This network can approximate the given problem with *N* samples with zero error. That means there exist parameters β_i, w_i and b_i such that $\sum_{i=1}^{\tilde{N}} \beta_i g(w_i x_j + b_i) = t_j$ for $j = 1,.....N$, where *t* is the target.The above N equations may be written as

$$H\beta = T \tag{8.1}$$

where

$$H = \begin{bmatrix} g(w_1.x_1 + b_1) & \dots & g(w_{\tilde{N}}.x_1 + b_{\tilde{N}}) \\ \vdots & \ddots & \vdots \\ g(w_1.x_N + b_1) & \vdots & g(w_{\tilde{N}}.x_N + b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}}$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N} \times m} \text{ and } T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m}$$

Given a training set, the number of hidden nodes and hidden node activation functions, the algorithm for a single layer feedforward network with random hidden weights is given as follows:

Step 1: The weights between hidden nodes and input nodes w_i and the threshold of the hidden nodes b_i are randomly assigned.

Step 2: Calculate the hidden layer output matrix H.

Step 3: Calculate the output linear layer weights using

$$\beta = H^{-1}T \tag{8.2}$$

where H^{-1} is the Moore–Penrose generalized inverse of matrix H. This gives the smallest norm least-squares solution of the above linear system and this solution is unique.

8.2.1 RANDOMIZED ANFIS (RANFIS) ARCHITECTURE

The structure of randomized ANFIS is same as that of conventional ANFIS [70] and is shown in Fig 1. The architecture uses Sugeno type fuzzy inference rules. It is assumed that two rules are used for the knowledge representation. A two rule Sugeno ANFIS network has rules of the form:

$$If\ x\ is\ A_1\ and\ y\ is\ B_1 \quad THEN\ f_1 = p_1 x + q_1 y + r_1$$

$$If\ x\ is\ A_2\ and\ y\ is\ B_2 \quad THEN\ f_2 = p_2 x + q_2 y + r_2$$

where, x and y are crisp inputs, A_i, B_i are linguistic variables.

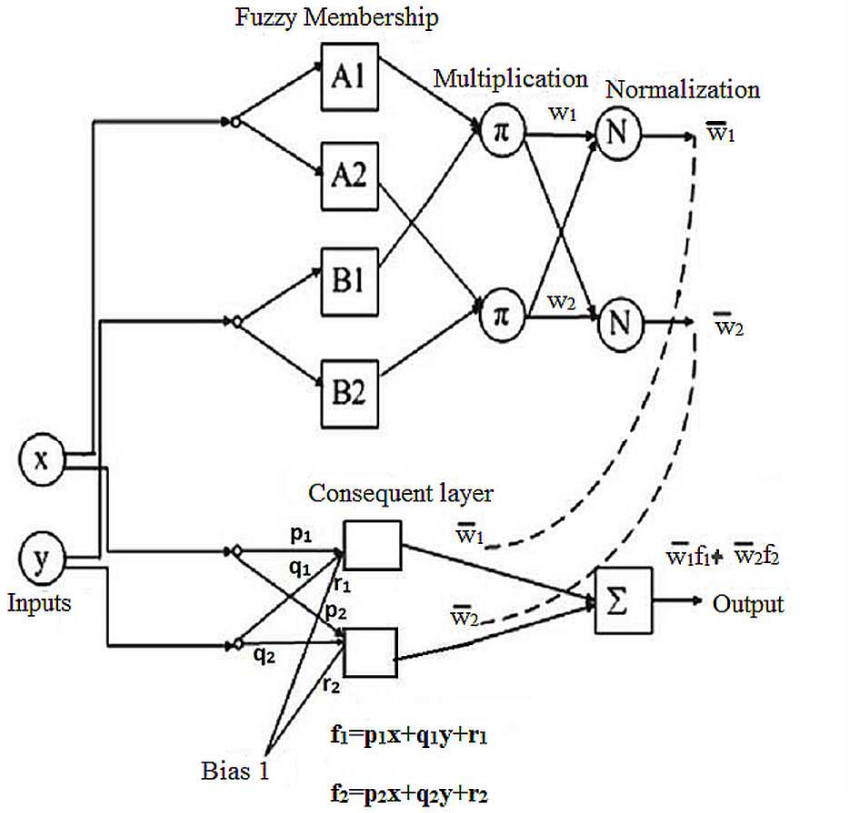

Fig 8.1: Architecture of RANFIS [114]

The top part of Fig. 8.1 has three layers and represents the premise part of the fuzzy rules and the bottom part has two layers and represents the consequent part. In the top part, the nodes in the first layer represent the fuzzy membership functions. The output of each node is

$\mu_{A_i}(x) for\ i = 1,2$ and $\mu_{B_{i-2}}(y) for\ i = 3,4$ where $\mu(x)$ is the membership grade for input x and $\mu(y)$ is the membership grade for input y. Fixed nodes are used in the second layer. The product t-norm is used to '*and*' the membership grades of premise parameters, as it gives a smoothing effect. Firing strengths w_i of the fuzzy rules are calculated as

$$w_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y), \quad i = 1,2 \tag{8.3}$$

Normalized firing strengths $\overline{w}_i$ are calculated in the third layer which also consists of fixed nodes.

$$\overline{w}_i = \frac{w_i}{w_1 + w_2} \tag{8.4}$$

The first layer in the bottom part of Fig. 8.1 represents a linear neural network with p_i, q_i and r_i as weight parameters. These weight parameters are adaptive and are learned using least square estimation method. Assuming normalized firing strengths of fuzzy rules and weight parameters are known, the output of this is given as

$$\overline{w}_i f_i = \overline{w}_i (p_i x + q_i y + r_i) \tag{8.5}$$

The second layer in the bottom part computes the overall output as:

$$t = \sum_i \overline{w}_i f_i = \overline{w}_1 (p_1 x + q_1 y + r_1) + \overline{w}_2 (p_2 x + q_2 y + r_2) \tag{8.6}$$

Bell shaped membership functions are used in the premise part, to provide smoothness in membership grade and flexibility in the core of membership functions. The Eq. 8.7 gives the mathematical representation of bell shape membership function.

$$\mu_A(x) = \frac{1}{1 + \left| \frac{x - c_j}{a_j} \right|^{2b_j}} \tag{8.7}$$

where, the premise parameters a_j, b_j, and c_j are position and shape deciding parameters.

If the sample size of the two dimensional (x and y) data in Fig. 1 is N, the targets $t_1, t_2 \ldots t_N$ can be found out using (8.6). The N linear equations can be expressed in matrix form as

$$\begin{bmatrix} t_1 \\ t_2 \\ . \\ . \\ . \\ t_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} \overline{w}_1 x & \overline{w}_1 y & \overline{w}_1 & \overline{w}_2 x & \overline{w}_2 y & \overline{w}_2 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \end{bmatrix}_{N\times 6} \begin{bmatrix} p_1 \\ q_1 \\ r_1 \\ p_2 \\ q_2 \\ r_2 \end{bmatrix}_{6\times 1}$$

The above matrix equation completely describes the two-rule RANFIS structure shown in Fig. 8.1. The premise parameters a_j, b_j and c_j and the consequent parameters p_i , q_i and r_i are to be evaluated using the RANFIS algorithm which is explained in the next subsection.

In the general case, if there are N training data pairs, the N linear equations can be represented in matrix form as

$$T_{N\times 1} = H_{N\times m^n (n+1)} \beta_{m^n (n+1)\times 1} \qquad (8.8)$$

where, m is the number of membership functions, n is the dimension of input data and m^n is the maximum number of rules used.

8.2.2 RANFIS ALGORITHM

In the conventional ANFIS, the premise parameters are determined using gradient descent methods like backpropagation algorithm. Consequent parameters are learned using linear network's training methods like least mean squares estimation [69]. In this type of hybrid learning algorithm (HLA), in the forward pass, the input patterns are applied, assuming fixed premise parameters, the optimal consequent parameters are calculated by an iterative least mean square procedure. In the second pass called as backward pass the patterns are propagated again, and in this epoch, back propagation is used to change the premise parameters to reduce the training error, while the consequent parameters remain fixed. This procedure is continued until the training error is minimized.

In RANFIS, the strategy of random assumption of weights is applied to tune the premise parameters of the fuzzy rules. These parameters (a_j, b_j, and c_j) are randomly selected with some

constraints on the ranges of these parameters. The randomness is controlled in RANFIS due to the explicit knowledge embedded in the linguistic variables in the premise part of the fuzzy rules. Once the premise parameters for all the inputs are selected, the H matrix in (8.8) can be easily determined. Then the unknown linear network parameters can be determined as

$$\beta = H^{-1}T \tag{8.9}$$

Consider the training data, with *n* dimensional inputs and one dimensional target as $[X_1 \quad X_2 \quad . \quad . \quad X_n ; T]$

For input membership functions, the range of each input function is defined as

$$range_i = \max\{X_i\} - \min\{X_i\} \text{ for } i = 1,2,\ldots n \tag{8.10}$$

If there are *m* uniformly distributed membership functions, then the default parameters (a_j^*, b_j^*, c_j^*) of the *j*th membership function are given by

$$a_j^* = \frac{range_j}{2m-2} \tag{8.11}$$

The default value of b_j^* is 2 and c_j^* is the center of uniformly distributed membership function.

With these details, the **RANFIS algorithm** can be summarized in the following steps:

***Step1*:** Randomly assign the premise parameters (a_j, b_j, c_j) within the following ranges.

$\frac{a_j^*}{2} \le a_j \le \frac{3a_j^*}{2}$, $\left(c_j^* - \frac{d_{cc}}{2}\right) < c_j < \left(c_j^* + \frac{d_{cc}}{2}\right)$ where d_{cc} is the distance between two consecutive centers and b_j is selected from the range 1.9 to 2.1.

Step2: Calculate the premise layer output matrix H in (8.8).

Step3: Calculate the linear network parameter matrix β using (8.9).

Step 4: Training runs are repeated for 50-70 times to select the best model.

8.2.3 COMPARISON OF RANFIS WITH ANFIS

The following example would substantiate the advantage of RANFIS over the conventional ANFIS [70].

Let Z be 2 input (x,y) sinc function given in (8.12). Let the range of x and y be [-10, 10]. RANFIS and conventional ANFIS algorithm are used for modeling the function. For training and testing the algorithm, 121 and 100 equally spaced pairs are chosen respectively from this range. Four membership functions and hence 16 rules are used for adapting the parameters of both algorithms. It is implemented and tested on C and MATLAB R2012 b platform with processor Intel(R) core(TM) i3-3220 and 4GB RAM.

$$Z = sinc\,(x, y) = \frac{\sin(x)}{x} * \frac{\sin(y)}{y} \qquad (8.12)$$

Fig. 8.2 and Fig. 8.3 show the final membership functions of x and y obtained after training the conventional ANFIS and RANFIS respectively.

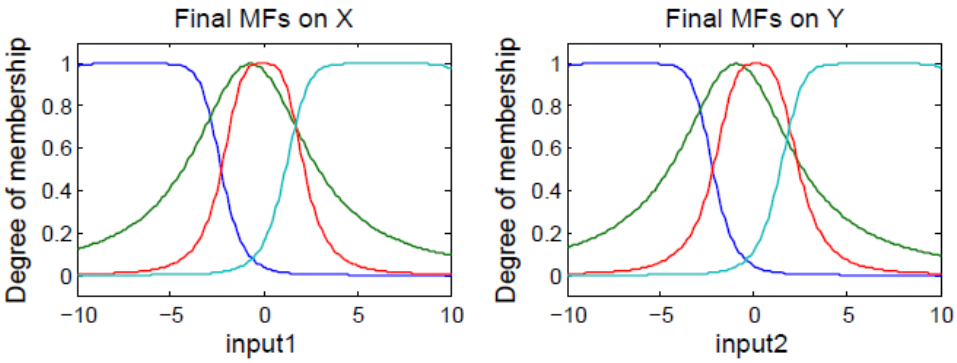

Fig 8.2: Final membership functions after training for conventional ANFIS

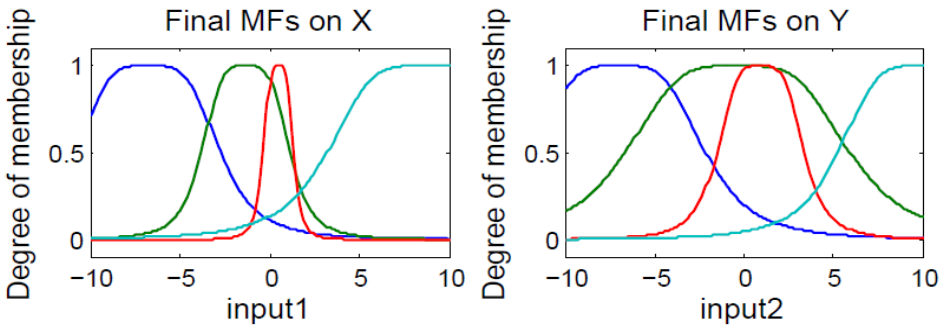

Fig. 8. 3: Final membership functions after training for RANFIS

Table 8.1: Comparison of conventional ANFIS (HLA) and RANFIS algorithm

Membership function		Conventional ANFIS (HLA)	RANFIS algorithm
4	Training Time (sec)	23.607	3.072
	RMSE Training	3.5e-4	0.0023
	RMSE Testing	2.1263	0.4825
5	Training Time (sec)	97.064	5.112
	RMSE Training	7.9e-5	1.78e-4
	RMSE Testing	4.5413	1.4510

From Table 8.1, it is clearly observed that the proposed RANFIS algorithm is faster with better testing accuracy compared to the hybrid learning algorithm of conventional ANFIS. It is also observed that with the increase in the number of membership functions, there is a drastic increase in the training time for conventional ANFIS whereas for RANFIS, the training time is comparatively much lesser. Hence the proposed RANFIS algorithm is simple compared to the computation involving HLA in conventional ANFIS and is faster and efficient. The qualitative knowledge embedded in the premise part of the fuzzy rules reduces the randomness of the first layer parameters.

8.3 MODELLING

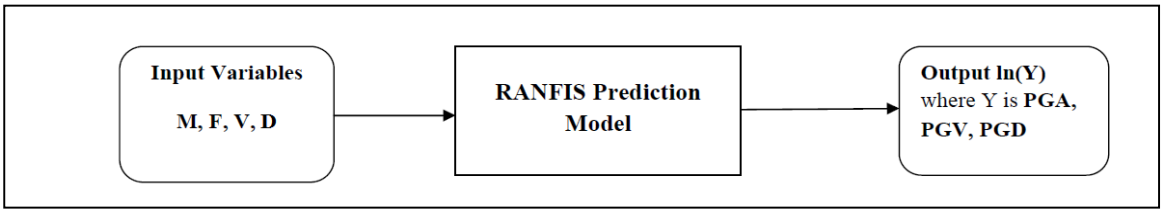

Fig 8.4: Model for prediction of principal Ground motion parameters.

The three ground motion parameters that are modelled are peak ground acceleration (PGA), peak ground velocity (PGV) and peak ground displacement (PGD). The developed ground motion parameter prediction model as shown in Fig.8.4, is expressed in terms of earthquake magnitude, faulting mechanism, site conditions and source to site distance. The number of membership functions is taken as 3. The input parameters to the model are earthquake magnitude, faulting mechanism, average shear wave velocity and closest distance. The output is principal ground motion parameter (Y), Y being PGA, PGV and PGD respectively. Hence the model is executed three times independently for PGA, PGV and PGD respectively. The model is analyzed based on two sets of data, namely training data set and testing data set.

The modelling parameters of the developed RANFIS prediction model are given in the Tables 8.2 and 8.3. The Table 8.2 lists the values of the premise parameters (a, b, c) for each of the four input variables of PGA, PGV and PGD prediction models. The consequent parameter matrix is of the order 81x5 and hence in the Table 8.3 only the maximum and minimum value is listed.

Table 8.2: The modelling parameters (premise) of RANFIS Prediction Model.

	PGA				PGV			PGD		
Input	**Mem.Fun**	**a**	**b**	**c**	**a**	**b**	**c**	**a**	**b**	**c**
1	**1**	**0.3358**	**1.9333**	**0.0616**	**0.3025**	**2.0353**	**-0.0182**	**0.2889**	**1.9413**	**0.0724**
	2	**0.3965**	**2.0299**	**0.3371**	**0.1710**	**1.9622**	**0.7169**	**0.1538**	**2.0617**	**0.6738**
	3	**0.1483**	**2.0491**	**1.1707**	**0.1264**	**1.9471**	**1.1347**	**0.2915**	**1.9095**	**0.7428**
2	**1**	**0.3472**	**2.0293**	**0.2557**	**0.2221**	**2.0733**	**0.2069**	**0.0149**	**1.9185**	**0.2210**
	2	**0.3736**	**2.0181**	**0.4969**	**0.4162**	**1.9017**	**0.5032**	**0.3027**	**2.0956**	**0.5739**
	3	**0.3595**	**1.9334**	**0.8502**	**0.3578**	**2.0162**	**0.7432**	**0.2764**	**2.0090**	**0.7515**
3	**1**	**0.0020**	**1.9906**	**-0.0037**	**0.0185**	**1.9880**	**0.0084**	**0.0083**	**2.0181**	**-0.0454**
	2	**0.0575**	**1.9421**	**0.7205**	**0.3377**	**1.9028**	**0.4125**	**0.1951**	**1.9187**	**0.3217**
	3	**0.3711**	**2.0866**	**0.9116**	**0.0645**	**2.0594**	**1.0502**	**0.3544**	**2.0584**	**1.1501**
4	**1**	**0.1266**	**1.9821**	**-0.0450**	**0.2089**	**1.9788**	**0.1247**	**0.0459**	**1.9048**	**0.2661**
	2	**0.2179**	**1.9943**	**0.6440**	**0.0540**	**1.9780**	**0.6198**	**0.1318**	**2.0554**	**0.4705**
	3	**0.2815**	**1.9116**	**0.8781**	**0.3124**	**1.9489**	**0.8227**	**0.3238**	**2.0785**	**0.9371**

Table 8.3: The modelling parameters (consequent) RANFIS Prediction Model.

	Consequent parameters (81x5)		
	PGA	**PGV**	**PGD**
minimum	**-1.55**	**-4.55**	**-6.25**
maximum	**1.61**	**4.07**	**5.61**

8.4 RESULT ANALYSES AND DISCUSSION

In this section, the overall performance of the developed model is analyzed. The subsection 8.4.1 details the results obtained by the developed RANFIS prediction model. The subsection 8.4.2 compares the developed model with the existing benchmark models.

8.4.1 RESULTS OBTAINED BY THE DEVELOPED RANFIS MODEL.

The best result is obtained for 100 epochs with number of membership functions equal to 3, and the developed prediction model takes 183 seconds to attain the best MSE value. The performance of the developed RANFIS prediction model is gauged in terms of prediction accuracy and the error percentage. The performance of the developed model is described in the Figs.8.5-8.7 for both training and testing datasets. The Fig. 8.5 shows the results obtained for the prediction of peak ground acceleration. The Figs. 8.6 and 8.7 represent the results obtained for the prediction of peak ground velocity and peak ground displacement respectively. It is observed from the Figs 8.5-8.7 that the developed prediction model, for the prediction of PGA, PGV and PGD respectively, is efficacious as the models gives high prediction accuracy with lesser error percentage.

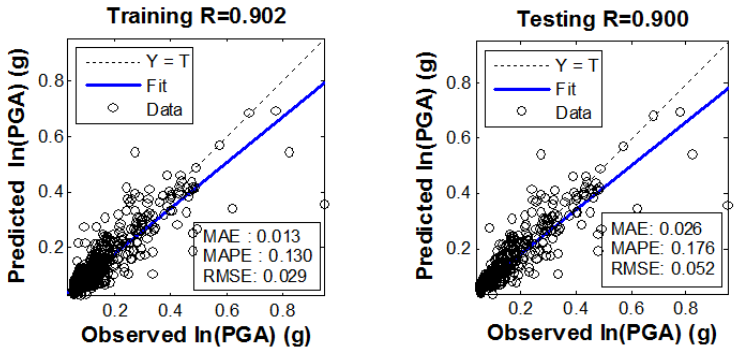

Fig 8.5: Predicted Vs Observed value of PGA (Training Data and Testing Data)

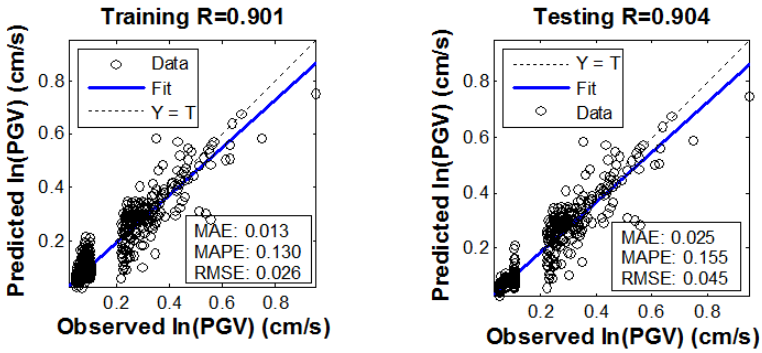

Fig 8.6: Predicted Vs Observed value of PGV (Training Data and Testing Data)

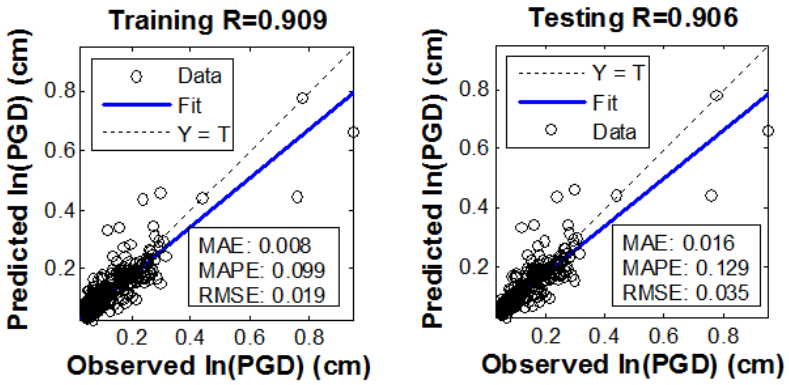

Fig 8.7: Predicted Vs Observed value of PGD (Training Data and Testing Data)

To further validate the potential of the developed RANFIS model, it is tested further on NGA WEST 2 dataset. Figs 8.8-8.10 shows the results obtained. The model gives good accuracy which supports the claim that the developed prediction model has good generalization capability.

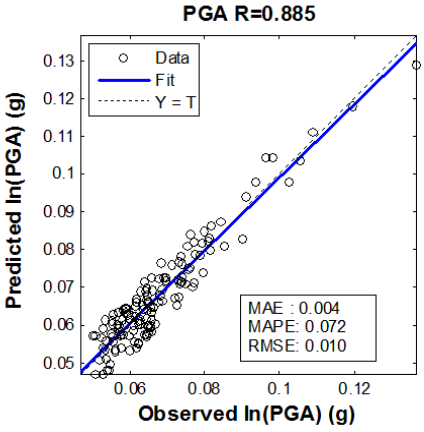

Fig 8.8: Predicted Vs Observed value of PGA

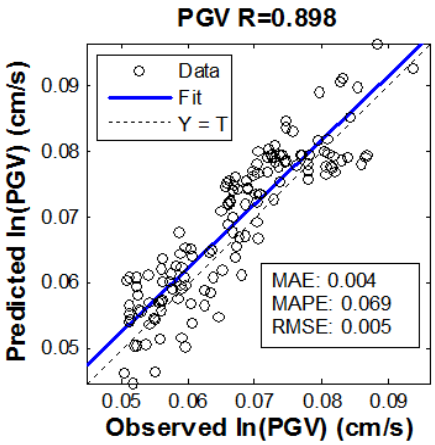

Fig. 8.9: Predicted Vs Observed value of PGV

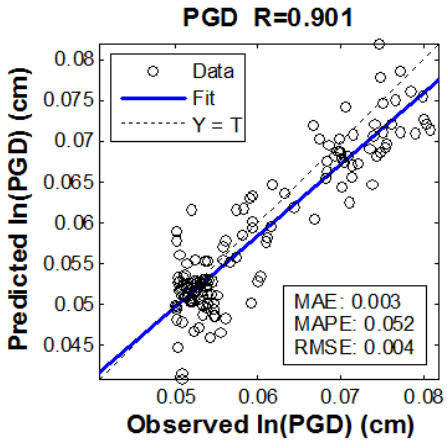

Fig 8.10: Predicted Vs Observed value of PGD

8.4.2 COMPARISON WITH EXISTING MODELS

The developed RANFIS prediction model is compared with 4 other existing models in the same database, namely GP/OLS model [45],ANN/SA model [4], MEP model [5] and GP/SA model [100] developed in the same database. The experimental results are tabulated in Tables 8.4-8.6. In the tables, where the corresponding values are missing, it is denoted by the symbol

n/a. The Table 8.4 represents the comparison of the developed RANFIS model with the existing models for the prediction of peak ground acceleration. The Tables 8.5 and 8.6 shows the comparison between the developed model and the existing models, for the prediction of PGV and PGD respectively. It is observed from the Tables 8.4-8.6, that the RANFIS model gives comparatively higher prediction accuracy with extremely small error percentage. Furthermore, it is observed that the RANFIS model is faster. The RANFIS model takes 183 seconds with 100 epochs to attain the best MSE value, whereas the ANN/SA prediction model [4] takes 364 epochs with 240 seconds to attain the minimum MSE value.

Table 8.4: Comparison of RANFIS prediction model for PGA with existing models

Criteria	RANFIS		ANN/SA [4]		GP/OLS [45]		MEP [5]		GP/SA [100]	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.902	0.900	0.869	0.855	0.836	0.811	0.842	0.834	0.833	0.839
MAE	0.013	0.026	0.30	0.46	0.478	0.488	0.363	0.697	n/a	n/a
MAPE	0.130	0.176	0.14	0.13	n/a	n/a	n/a	n/a	0.158	0.144
MSE	0.001	0.003	n/a	n/a	0.358	0.406	0.362	0.389	0.381	0.380
RMSE	0.029	0.052	n/a	n/a	0.836	0.637	0.602	0.624	0.617	0.616

Table 8.5: Comparison of RANFIS prediction model for PGV with existing models

Criteria	RANFIS		ANN/SA [4]		GP/OLS [45]		MEP [5]		GP/SA [100]	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.901	0.904	0.867	0.874	0.822	0.813	0.837	0.828	0.833	0.837
MAE	0.013	0.025	0.34	0.45	n/a	0.506	0.402	0.726	n/a	n/a
MAPE	0.130	0.155	1.06	2.17	0.512	n/a	n/a	n/a	1.27	2.35
MSE	0.001	0.002	n/a	n/a	0.421	0.405	n/a	n/a	n/a	n/a
RMSE	0.026	0.045	n/a	n/a	0.649	0.637	0.634	0.671	0.645	0.648

Table 8.6: Comparison of RANFIS prediction model for PGD with existing models

Criteria	RANFIS		ANN/SA [4]		GP/OLS [45]		MEP [5]		GP/SA [100]	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
R	0.909	0.906	0.870	0.869	0.836	0.811	0.846	0.840	0.847	0.854
MAE	0.008	0.016	0.62	0.62	n/a	n/a	0.733	0.829	n/a	n/a
MAPE	0.099	0.129	1.74	1.66	0.660	0.681	n/a	n/a	1.61	1.68
RMSE	0.019	0.035	n/a	n/a	0.850	0.901	0.856	0.899	0.845	0.846

It is clearly observed from the Tables 8.4-8.6, that the RANFIS model is efficient compared to other existing models. The developed RANFIS model is further compared with few existing GMPEs to prove the efficacy of the RANFIS model. There are numerous attenuation relationships for the prediction of principal ground motion parameter and comparing with all of them is not feasible. Hence the developed RANFIS model is compared with the few GMPEs developed in the same database. The criterion for comparison is the error percentage in the prediction of each of the principal ground motion parameters (PGA, PGV, and PGD) for the testing dataset (563 records). It is observed from the Table 8.7, that the developed RANFIS model gives better precision as the error percentage is extremely less. Another advantage of the developed RANFIS model is that it is much simpler as compared to GMPEs as the RANFIS prediction model includes only 4 geophysical parameters as input, whereas GMPEs include many geophysical parameters.

Table 8.7: Comparison of model with GMPEs (n/a denote value not mentioned)

Model	**Criteria MAPE**		
	PGA	**PGV**	**PGD**
RANFIS	**0.175**	**0.158**	**0.108**
Campbell and Bozorgnia [29]	**0.93**	**0.78**	**5.73**
Least Square regression analysis [96]	**0.16**	**2.47**	**1.70**
Ambraseys et al [6]	**0.95**	**n/a**	**n/a**
Smit et al [142]	**14.58**	**n/a**	**n/a**

8.5 CONCLUSION

In this study, a new and faster hybrid machine learning algorithm, RANFIS is used to predict the three principal ground motion parameters PGA, PGV and PGD. The hybrid network used in modelling also avoids the randomness of the results inherent in randomized SLFN networks by incorporating explicit knowledge representation using fuzzy membership functions. The results obtained show that the developed prediction model is efficient as it gives better precision results in lesser computational time compared to the existing GMPEs models and soft computing models. Moreover, the simulation studies show that the RANFIS algorithm is faster than the conventional adaptive neural fuzzy inference system learning algorithm.

In the next chapter, all the developed prediction models are compared for concluding the best prediction model for forecasting ground motion parameter, proposed in this study.

CHAPTER 9
COMPARISON OF ALL PROPOSED PREDICTION MODELS

9.1 INTRODUCTION

In this chapter all the developed predictive models for forecasting the ground motion parameters are compared and the learning effectiveness of each algorithm is further explored. In chapters 4-8 of this study, the respective developed prediction models are compared to existing benchmark models and the GMPEs. In this chapter, all the predictive models proposed in this study, for forecasting ground motion parameters are compared in terms of prediction accuracy and error measure. Hence this chapter concludes the best prediction model among all the developed prediction models in this study. Futhermore, the prediction models are compared for 'learning effectiveness' of the algorithm which analyzes the learning ability of the algorithms. In this chapter for the comparative analyzes of the developed ground motion parameter prediction models, the predictive model for forecasting peak ground acceleration is considered. The chapter is organized as follows. Section 9.2 details the comparison of all the 6 developed predictive models with respect to the training data. The predictive models are further compared in terms of testing data in section 9.3. The learning effectiveness of all the predictive models is analyzed in section 9.4. The discussion on the comparison of predictive models is concluded in section 9.5.

9.2 COMPARISON OF TRAINING DATA

In this section, the all the developed peak ground acceleration predictive model is compared for training data consisting of 2252 earthquake records. Table 9.1 tabulates the performance during training of the developed models. The models are tabulated in the order of their performance. It is observed that the novel neuro fuzzy technique (RANFIS) is the best predictive model as it gives high prediction accuracy. It is also observed that the prediction accuracy for prediction model based on hybrid architecture ANFIS is comparable with prediction model based on kernel method learning (LS-SVR model). Similarly, the ELM based prediction model performs better than the decision tree and SLFN based prediction model.

Fig 9.1 shows the correlation coefficient (R) of all the developed prediction models with various measures of training error. Fig. 9.2 shows the comparison of the correlation coefficient (R) with training error measured in terms of mean absolute percentage error (MAPE), for all the developed prediction models.

Table 9.1: Comparison of all developed models for training data (PGA)

Criteria	RANFIS	ANFIS	LS-SVR (RBF kernel)	ELM	Decision tree	SLFN	ν –SVR (RBF kernel)	ε –SVR (RBF kernel)
R	**0.902**	**0.88**	**0.872**	**0.845**	**0.852**	**0.807**	**0.731**	**0.696**
MAE	**0.013**	**0.010**	**0.014**	**0.018**	**0.015**	**0.021**	**0.024**	**0.053**
MAPE	**0.130**	**0.140**	**0.137**	**0.182**	**0.131**	**0.219**	**0.246**	**0.716**
RMSE	**0.029**	**0.032**	**0.033**	**0.037**	**0.036**	**0.040**	**0.048**	**0.066**

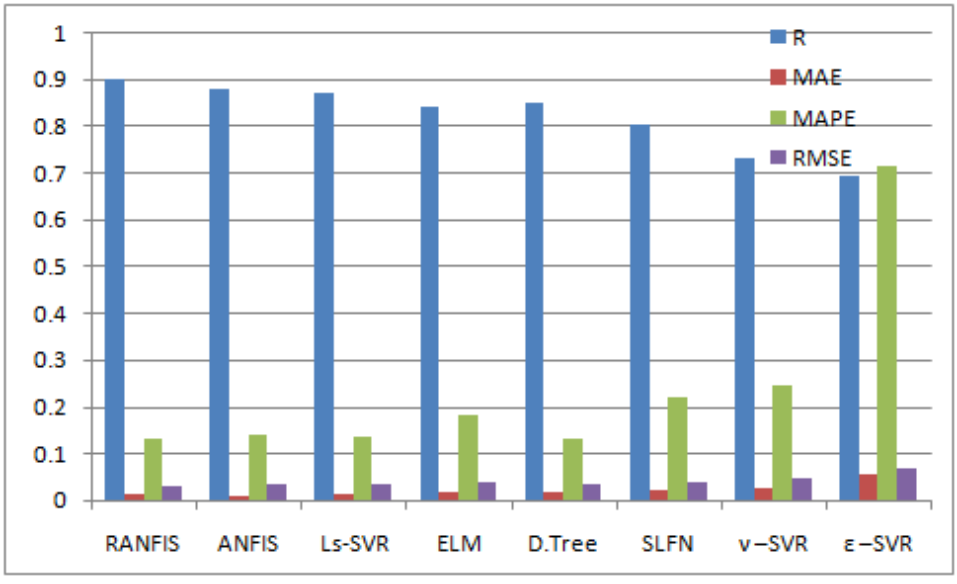

Fig. 9.1 Comparison of all developed predictive models for train data

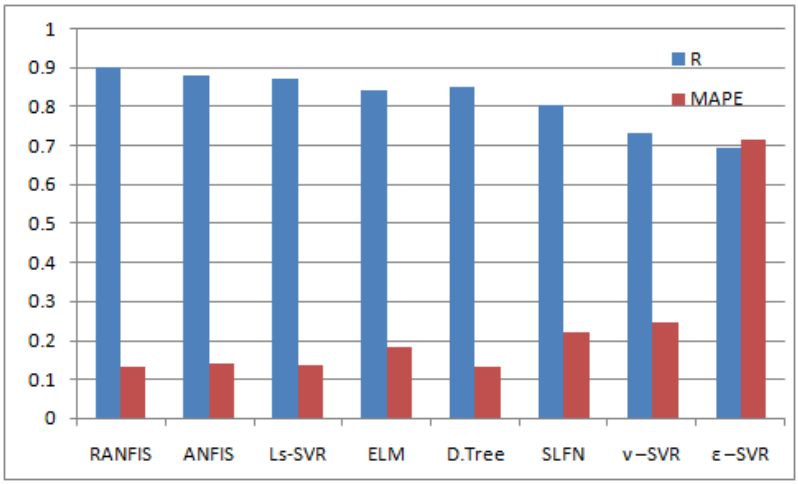

Fig. 9.2: Comparison of R with MAPE for all developed predictive models for train data

9.3 COMPARISON OF TESTING DATA

In this section, the developed peak ground acceleration predictive models are compared for testing data consisting of 563 earthquake records as well as for the NGA WEST 2 dataset consisting of 140 earthquake records. For any developed predictive model, the efficacy of the predictive model depends on the performance of the predictive model on the testing data sets. In this work, the models are tested for 703 earthquake records.

The Table 9.2 represents the comparison of the developed models for the test data of 563 records. The models are tabulated in the order of the performance of the models. Fig. 9.3 represents the correlation coefficient (R) along with four measures of testing error, for all the developed predictive models for the test data of 563 records. It is observed that the best prediction algorithm is the novel neuro fuzzy technique, RANFIS. It outperforms the existing ANFIS architecture. It is also observed that the kernel based model (LS-SVR model) outperforms the neural based model (ELM model). The performance of the LS-SVR based prediction model is comparable to ANFIS based prediction model. The Fig 9.4 shows the

comparison of correlation coefficient (R) with testing error MAPE for all developed predictive models for the test data of 563 records.

Table 9.2: Comparison of all developed model for testing data (PGA)

Criteria	RANFIS	ANFIS	LS-SVR (RBF kernel)	ELM	Decision tree	SLFN	ν –SVR (RBF kernel)	ε –SVR (RBF kernel)
R	**0.900**	**0.88**	**0.870**	**0.849**	**0.840**	**0.819**	**0.763**	**0.764**
MAE	**0.026**	**0.030**	**0.032**	**0.035**	**0.033**	**0.039**	**0.046**	**0.058**
MAPE	**0.176**	**0.180**	**0.189**	**0.229**	**0.192**	**0.266**	**0.277**	**0.558**
RMSE	**0.052**	**0.057**	**0.061**	**0.066**	**0.065**	**0.070**	**0.088**	**0.080**

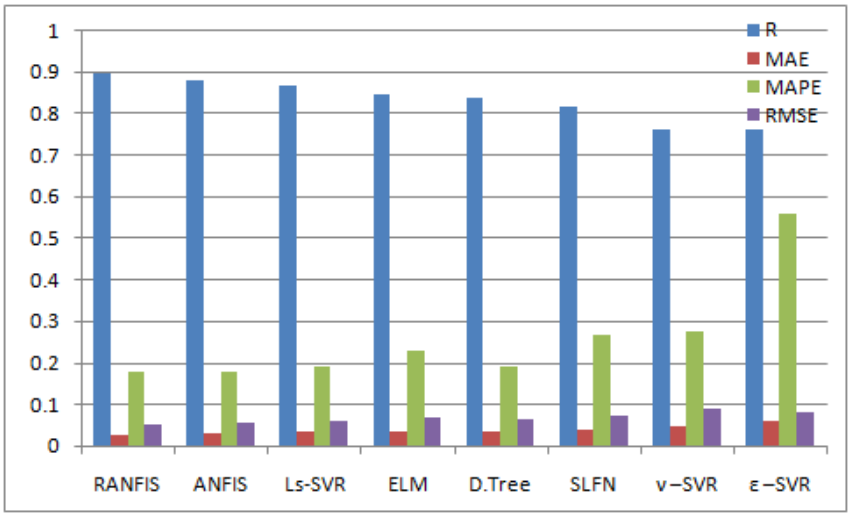

Fig 9.3 Comparison of all developed predictive models for test data

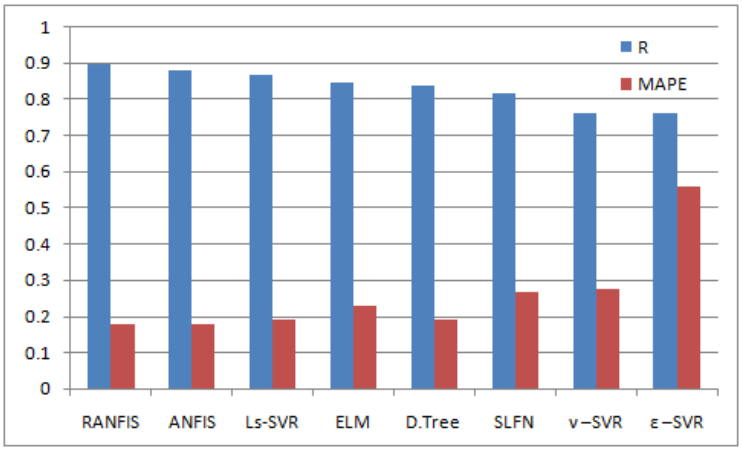

Fig. 9.4: Comparison of R with MAPE for all developed predictive models for test data

The developed prediction models are further tested on NGA WEST 2 dataset consisting of 140 earthquake records. The Table 9.3 tabulates the performance of the developed predictive models on NGA WEST 2 dataset. The further evaluation of performance of the developed prediction models on this dataset validates the efficacy of the model. The results show that the developed models are independent of the database on which the model is trained and works well for any set of data, which matches the statistical parameters of the training data set. All the developed prediction model is trained for earthquake magnitude ranging from 5.2 to 7.9. The NGA WEST 2 dataset consists of 140 earthquake records, outside the training database but within this range of magnitude.

Table 9.3: Comparison of all developed Prediction models for NGAWest 2 data (PGA)

Criteria	RANFIS	ANFIS	LS-SVR (RBF kernel)	ν –SVR (RBF kernel)	Decision Tree model	ε –SVR (RBF kernel)	ELM	SLFN model
R	**0.885**	**0.820**	**0.8140**	**0.7171**	**0.7004**	**0.6585**	**0.542**	**0.4242**
MAE	**0.004**	**0.006**	**0.0060**	**0.0148**	**0.0065**	**0.0537**	**0.046**	**0.0147**
MAPE	**0.072**	**0.089**	**0.0853**	**0.2180**	**0.0876**	**0.8308**	**0.291**	**0.2171**
RMSE	**0.010**	**0.008**	**0.010**	**0.017**	**0.010**	**0.060**	**0.031**	**0.024**

It is clearly observed that the novel neuro fuzzy algorithm RANFIS outperforms all the other developed models. Another interesting observation from Table 9.3 is that it clearly shows that the generalization of kernel method (SVR) based model is much better than the neural based learning (ELM, ANN). Furthermore, it is also observed that the generalization is better for ELM compared to SLFN, which substantiates the advantage of ELM over ANN. Fig. 9.5 represents the correlation coefficient (R) along with four measures of testing error, for all the developed models for the NGA WEST 2 dataset. Fig. 9.6 shows the comparison of R, the correlation coefficient for the NGA WEST 2 dataset with testing error MAPE. It is observed that the RANFIS algorithm has the least testing error with highest accuracy.

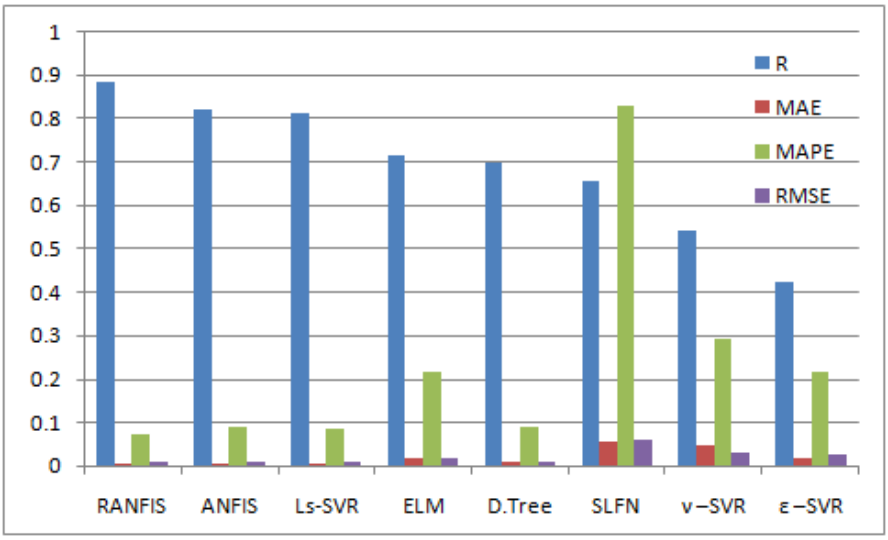

Fig. 9.5: Comparison of all developed predictive models for NGA WEST 2 dataset

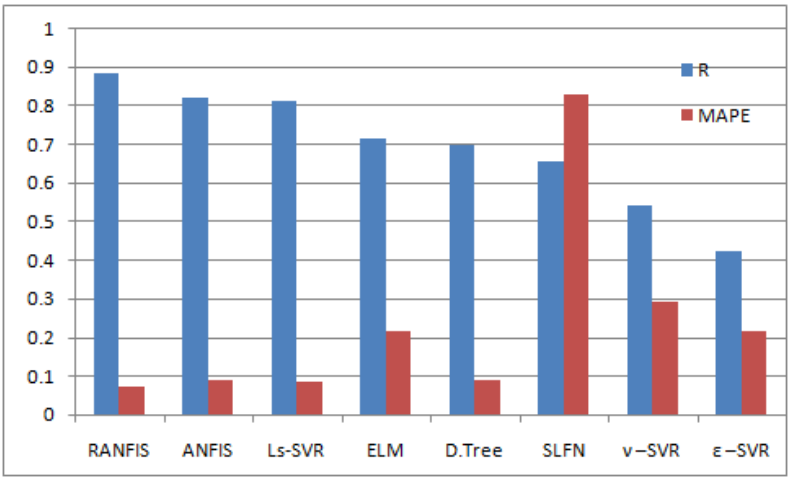

Fig. 9.6: Comparison of R with MAPE for all developed predictive models for NGA WEST 2 dataset

9.4 OVERFITNESS MEASURE OF LEARNING ALGORITHMS

In this section, a comparative study is done based on the learning ability of the prediction algorithms used in the study, along with the existing benchmark models, as mentioned in Table 3.1. The efficacy of the learning capability of the algorithms is based on the measure of 'overfitness'. All the models considered in this section are the developed predictive models, as well existing benchmark models, for forecasting PGA.

The efficacy of any model is analyzed on the testing error. The overfitness measure [161] is a comparison between training and testing error. The measure of overfitness is calculated as given in Eq. 5.28, mentioned in section 5.4.3 in chapter 5. The measure of error considered is mean absolute error percentage (MAPE), detailed in section 3.6 of chapter 3. The testing data considered in evaluating the overfitness measure consists of 563 earthquake records. The comparative results obtained on testing data from all the developed predictive models are tabulated in Table 9.2.

Higher measure of overfitness shows that the algorithm has a drawback of overfitting the data. Hence it is used as a criteria to compare models having comparable prediction accuracy.

Table 9.4 tabulates the measure of overfitness for each algorithm. The model having a comparatively lesser measure of overfitness is said to have better generalization.

From the results shown in Tables 9.1-9.3, it is concluded that the RANFIS prediction model is the best developed predictive algorithm. The learning algorithm proposed in the RANFIS model, a novel neuro fuzzy technique, has high prediction accuracy with lesser error measure and overfitness measure. It is observed that RANFIS model gives the least testing error measure among all the developed models. The measure of overfitness is acceptable as the RANFIS model gives good precision (as shown in Table 9.3), for the NGA WEST 2 dataset, consisting of 140 earthquake records, which is a set of earthquake data outside the database used for modelling. Thus, the novel neuro fuzzy technique is efficacious as it gives remarkable precision accuracy, with much lesser error percentage and without overfitting the data during modelling.

Table 9.4: Comparison of all Prediction models for overfitness

Model	Accuracy	Testing Error	Overfitness
RANFIS	**0.90**	**0.176**	**0.34**
ANFIS	**0.88**	**0.180**	**0.29**
ELM	**0.85**	**0.229**	**0.26**
ε-SVR (RBF kernel)	**0.76**	**0.558**	**0.22**
ν-SVR (RBF kernel)	**0.76**	**0.277**	**0.13**
LS-SVR (RBF kernel)	**0.87**	**0.189**	**0.38**
Decision Tree	**0.84**	**0.192**	**0.47**
SLFN	**0.82**	**0.266**	**0.28**
GP/SA	**0.839**	**0.144**	**0.21**
GP/OLS	**0.811**	**0.49**	**0.24**
MEP	**0.834**	**0.69**	**0.92**
ANN/SA	**0.855**	**0.46**	**0.53**

Further comparing the developed predictive models, it is observed from Tables 9.1-9.3, that ANFIS predictive model and LS-SVR prediction model have commensurate performance. The prediction accuracy and the error measure are comparable. It is observed from Table 9.4, that the overfitness measure for the 2 pass hybrid learning algorithm (HLA) of ANFIS model is comparatively lesser than the kernel method based learning algorithm, LS-SVR. Hence, comparatively it could be said that ANFIS based prediction model for forecasting ground motion parameter is better than LS-SVR based prediction model. Although ANFIS model has advantages, it has few bottleneck issues. In this study, the ANFIS model with membership function 3 was considered. ANFIS architecture has the drawback of higher computational time and complexities with higher membership function.

It is further observed from Tables 9.1-9.3, that the performance of decision tree based prediction model is comparable with the ELM based prediction model. From the Table 9.4 it is clearly observed that decision tree based model has a higher measure of overfitness. This substantiates the major drawback of decision tree modelling. The key issue of decision tree modelling is that the tree structure obtained after modelling is highly dependant on the dataset. Any variation in the dataset results in a different tree structure. Thus, there is overfitting of the data. The results in Table 9.4 proves this issue of decision trees. Hence the ELM based prediction model is comparatively better than a decision tree based model.

The further vital observation from the Table 9.4 is that the overfitness for SVR based models is less compared to SLFN model, as well the ANN/SA model. Thus, it validates the claim by the kernel based learning method (SVR) that it overcomes the major drawback of overfitting of data, as in the case of neural network based methods. Hence it is validated that SVR has a better generalization than ANN. Moreover, it is concluded from the Table 9.4, that ELM has a better generalization than ANN, as overfitness measure is less for ELM compared to SLFN, as well as ANN/SA models.

9.5 CONCLUSION

In this chapter, all the developed predictive models in this study, for forecasting ground motion parameters are compared. It is clearly observed from the result analysis in chapters 4-8, that all the developed prediction models for forecasting ground motion parameter outperforms the existing benchmark models as well as the existing conventional GMPEs. All the developed predictive models based on advanced machine learning have better accuracy and lesser measure of the error percentage compared to the existing models. In this chapter, all the proposed predictive models are compared in terms of prediction accuracy and the different measure of error percentages. Furthermore, the prediction algorithms are analyzed for their learning capability. The learning ability of the algorithms is expressed in terms of a measure termed 'overfitness', which defines overfitting, the bottleneck issue of any leaning machine. An efficient learning machine should be one which does not overfit the data.

From the results tabulated in Tables 9.1-9.4, it is concluded that from among all the proposed predictive models in this study, the best prediction model for forecasting ground motion parameter is RANFIS model, the improved conventional ANFIS. It is followed by the ANFIS model, the hybrid neuro fuzzy architecture, followed by the LS-SVR model, followed by the ELM based prediction model. Hence it is concluded that the hybrid neuro fuzzy architecture outperforms the kernel based learning method, SVR.

The major drawback of the hybrid learning algorithm (HLA) of the ANFIS model is the computational time due to the complexities which arises at the higher membership function. It is observed from Table 8.1 that the RANFIS algorithm overcomes this drawback of ANFIS.

It is also observed from Table 9.4 that the SVR algorithm has a lesser measure of overfitness. The ε-SVR model and ν- SVR models have the least measure of overfitness measure. Among the kernel method based learning, the LS-SVR model gives the best result. The results show that the LS-SVR model has a slightly higher measure of overfitness. It is because for obtaining the optimal value of gamma for LS-SVR algorithm, simulated coupling method is used. The simulated annealing (SA) is a probabilistic optimization algorithm and has its own drawbacks.

The results also show that kernel based learning has lesser measure of overfitness compared to ELM. Hence it is concluded that support vector regression has definitely a better generalization compared to neural networks.

The ELM based model outperforms the decision tree based modelling as well as the neural based models such as the SLFN and the benchmark model ANN/SA. This further supports the claim that ELM has a better generalization than ANN.

It is further observed that the highest value for overfitness measure is for tree based modelling. Thus, the major drawback of the tree based modelling which is data dependency is clearly highlighted. The existing benchmark models such as GP/OLS, GP/SA and MEP models could be considered as an attempt to develop efficient tree like structures. The genetic programming (GP) is an evolutionary algorithm similar to genetic algorithms (GA). The multi expression programming (MEP) model is again a variant of GP model. The final output of the GP model is a tree like structure. It is observed from Table 9.4 that GP/OLS and GP/SA models have a lesser measure of overfitness. The MEP model is highly dependant on the data as, in MEP a linear representation is used. The measure of overfitness for the MEP model as in Table 9.4 clearly validates this drawback.

It is also observed that developed models have better precision compared to genetic programming as well as its existing hybrid architectures (GP/SA). The next chapter concludes the study by highlighting the significance of the work done in this thesis.

Chapter 10

CONCLUSION AND FUTURE SCOPE

10.1 CONCLUSION

In this study, predictive models based on advanced machine learning are developed for forecasting ground motion parameters. The advanced machine learning algorithms used are extreme learning machines (ELM) which is based on neural network learning, three variations of support vector regression (SVR) which is based on kernel methods learning, decision tree, and two hybrid architecture ANFIS and RANFIS (randomized ANFIS). ANFIS is an integrated hybrid architecture of fuzzy logic with neural networks such that the knowledge gained by the fuzzy logic is used by the learning algorithm of neural network. RANFIS is the new learning algorithm proposed for predictive models in this study. The RANFIS algorithm integrates the explicit knowledge of the fuzzy systems with the learning capabilities of neural networks, as in the conventional ANFIS system, but with the difference that, the fuzzy layer parameters in RANFIS are not tuned. This improvement in the architecture of ANFIS structure helps to accelerate the learning speed without compromising the generalization capability.

All the machine learning algorithms used in this study are novel in this domain. All the developed predictive models are meticulously analyzed and the results are well validated. It is clearly observed that the developed predictive models in this study are advantageous when compared to the existing benchmark models in this domain as shown in Table 3.1, as well as with the traditional ground motion parameter equations (GMPEs). The existing benchmark models are ANN/SA model by Alavi and Gandomi [4], GP/OLS model by Gandomi et al [45], MEP model by Alavi et al [5] and GP/SA model by Mohammadnejad et al [100] and the existing GMPE models are Ambraseys et al. [6], Smit et al. [142], and Campbell and Bozorgnia [29]. The chapter 9 further details the comparative study among all the developed ground motion parameter prediction models. In this comparative study of the various proposed algorithms, the learning effectiveness of each algorithm is analyzed, which further highlights the advantages and drawbacks of each advanced machine learning algorithm.

Furthermore, the proposed novel neuro fuzzy technique, 'RANFIS' proves to be promising prediction algorithm for forecasting ground motion parameters.

This study manifests that advanced learning machines could be effectively used as a predictive tool in this domain. In this study, the database used is real earthquake data recorded worldwide. The predictive models developed in this study are not data dependant. In this study, the models are developed for earthquake magnitude ranging from 5.2 to 7.9, in the scale of moment magnitude. The models could be trained for any range of earthquake magnitude, as per the user requirement.

All the developed predictive models were trained and tested on earthquake records obtained from PEER database [111]. The developed models were further tested on another dataset NGA WEST 2 [23] consisting of 140 earthquake records. Hence it is validated that the developed predictive models are not constrained on the data.

The study also brings out the drawbacks of linear regression analysis, which is used for the generation of GMPEs. It is observed from the results obtained in this study, that the developed predictive models for forecasting ground motion parameter, not only gives faster and more accurate ground motion parameter prediction but also the drawbacks of linear regression modelling is overcomed. The key issues of the ground motion prediction equations (GMPEs) developed using regression analysis is that the results have a higher measure of error as well the overhead of solving the equation as it consist of a large number of coefficients. All the proposed prediction models in this study use only 4 geophysical parameters for the forecasting of ground motion parameter. The results obtained show that the developed prediction models are efficient as it gives better precision results in lesser computational time compared to the existing GMPEs models and soft computing models. Moreover, in this study all the developed models are analyzed on 4 different measures of error. The detailed analysis of the advanced machine learning algorithms is done in chapter 9.

The vital application of the proposed model is that it could be used as tool for faster prediction of the ground motion parameter with lesser calculation overhead, in all areas such seismic risk assessment, seismic hazard analysis, earthquake resistant structural engineering, etc. where the principal ground motion parameters are used as a vital input.

10.2 FUTURE SCOPE

Although it is well validated that the proposed predictive models in this study are efficacious, there are some aspects of this study that could be considered as the future scope of the work.

A further extension of this study is to identify newer and relevant geophysical parameters, which could replace the currently used four input parameters. The four geophysical parameters considered in this study are believed to be the most prominent attribute relating to an earthquake process and ground motion parameter. From the study of seismology, it is observed that, the signal generated during an earthquake process, is dependant on many features. From the literature survey, (Douglas [39]), it is perceived that other important features are the energy released during an earthquake, the depth of an earthquake (focal depth), the soil properties, the path travelled by the signal and so on.

Furthermore, the distance measure considered for modelling is the closest distance, which is defined as the closest distance of the ruptured area to the recording site. A further study could be done so as to deduce the influence of different measures of distances on the ground motion parameter.

In this study, the novel neuro fuzzy technique 'RANFIS', was applied to forecast ground motion parameters. The proposed prediction algorithm could be applied to various complex real world problems such as in biomedical signal processing, or image processing or a stock market prediction, or prediction of other natural calamities such as landslide prediction etc.

The proposed RANFIS model is an improved conventional ANFIS. Although this proposed algorithm is efficient, it could be improved in terms of computational time. In RANFIS, the maximum number of rules used is m^n, for n dimensional input and m number of membership functions. The computational time of the algorithm could be improved if the number of rules is reduced. Hence the optimization for the reduction of the number of rules, could be a further extension of the work.

Furthermore, the efficacy of RANFIS could be tested on data, having large number of input attributes with huge number of instances. In this study, the dimensionality of the database is 2815 numbers of earthquake records with 4 input features. The study of the behavior of the RANFIS prediction algorithm for datasets having more than 10 input features with large number of records (number of records >5000), would be interesting. To improve the computational time and the efficacy of the model for such huge datasets would be a real challenge.

PUBLICATIONS

1. Sonia Thomas, G.N Pillai, Kirat Pal, Pushpak Jagtap, “Prediction of Ground Motion Parameters using Randomized ANFIS (RANFIS)”, *Applied soft computing, ISSN: 1568-4946.*
   Paper Online. doi:10.1016/j.asoc.2015.12.013

2. Sonia Thomas, G.N Pillai, Kirat Pal," Prediction of Peak Ground Acceleration using εSVR, νSVR and LsSVR Algorithm", *Geomatics, Natural Hazards and Risk, ISSN: 1947-5705*. [SCI Journal (Accepted)].

3. Sonia Thomas, G.N Pillai, Kirat Pal,“ Adaptive Neural Fuzzy Inference System for prediction of ground motion parameter ”, *REVISTA KASMERA, (ISSN: 0075-5222)* [SCI Journal (Accepted)].

4. Sonia Thomas, G.N Pillai, Kirat Pal,“ Prediction Model for Ground Motion Parameters Using Extreme Learning Machines ”, *Neural Computing and Applications, ISSN: 0941-0643*. [SCI journal, (review completed awaiting editor`s decision)].

5. Sonia Thomas, “Significance of the Parametric Analysis of Seismic Signatures: A review on the various seismic signal parameters”, *DSIJ, Science, Engineering and management, Volume 1, Issue 1 bearing the ISSN no: 2347-1603,2013* [Published]

6. Sonia Thomas, G.N Pillai, Kirat Pal, “Prediction of Peak Ground Acceleration (PGA) using Artificial Neural Networks”, *Proc. of Int. Conf. on Advances in Computer Science, AETACS, ACEEE Conference Proceedings Series 02, Elsevier Publication. ISBN 978-93-5107-193-8.* Paper Online: http://searchdl.org/public/book_series/elsevierst/6/116.pdf

BIBLIOGRAPHY

1. N. Acir and C. Güzeliş, "Automatic spike detection in EEG by a two-stage procedure based on support vector machines," *Comput. Biol. Med*., vol. 34, no. 7, pp. 561–575, 2004.
2. N.A Abrahamson and W.J Silva, 2008. "Summary of the Abrahamson & Silva NGA ground-motion relations,"*Earthq. Spectra*., vol. 24 (1), pp. 45–66, 2008.
3. R. Adomaitis, R.M. Farber, J.L. Hudson, I.G. Kevrekidis, M. Kube, and A.S.Lapedes, "Application of neural nets to system identification and bifurcation analysis of real world experimental data,"*Neural networks: Biological computers or Electronic brains*, Springer –Verlag, Paris France, pp. 87-97, 1990.
4. A. H. Alavi and A. H. Gandomi, "Prediction of principal ground-motion parameters using a hybrid method coupling artificial neural networks and simulated annealing," *Comput. Struct.*, vol. 89, no. 23–24, pp. 2176–2194, Dec. 2011.
5. A. H. Alavi, A. H. Gandomi, M. Modaresnezhad, and M. Mousavi, "New Ground-Motion Prediction Equations Using Multi Expression Programing," *J. Earthq. Eng.*, vol. 15, no. 4, pp. 511–536, 2011.
6. N.N. Ambraseys, K. A. Simpson and J.J. Bommer, "Prediction of horizontal response spectra in Europe,"*Earthq. Eng. Struct. Dyn* . vol. 25, pp. 371–400, 1996.
7. S. Amari and S. Wu, "Improving support vector machine classifiers by modifying kernel functions," *Neural Netw.*, vol. 12, no. 6, pp. 783–789, 1999.
8. S.M. Anzar and P.S. Sathidevi, "On combining multi-normalization and ancillary measures for the optimal score level fusion of fingerprint and voice biometrics,"*EURASIP, Journal on Advances in Signal Processing*.
   DOI:10.1186/1687-6180-2014, 2014.
9. S. M. Anzar and P.S. Sathidevi, "Fusion of Biometric Modalities Using Multi-Normalization and Genetic Algorithm,"***International Review on Computers and Software (I.RE.CO.S.)***, vol.7, no. 6, ISSN 1828-6003, 2012.
10. F. Ardjani, K. Sadouni, and M. Benyettou, "Optimization of SVM multiclass by particle swarm (PSO-SVM)," *2010 2nd Int. Work. Database Technol. Appl. DBTA2010 - Proc.*, no. December, pp. 32–38, 2010.

11. E. Arslan, B. Metin, C. Cakır, and O. Cicekoglu, “A Novel Grounded Lossless Inductance Simulator with CCI,” *International Twelfth Turkish Symposium on Artificial Intelligence and Neural Networks TAIN 2003*, 2-4 July, Çanakkale-Turkey, 2003.
12. M. Asadi, M. Hossein Bagheripour, and M. Eftekhari, “Development of optimal fuzzy models for predicting the strength of intact rocks,” *Comput. Geosci.*, vol. 54, pp. 107–112, 2013.
13. H. M. Azamathulla, A. A. Ghania, and S. Y. Fei, “ANFIS-based approach for predicting sediment transport in clean sewer,” *Appl. Soft Comput.*, vol. 12, no. 3, pp. 1227–1230, 2012.
14. A. Badlani, S. Bhanot, “Smart Home System Design based on Artificial Neural Networks,”*Proc. of the World Congress on Engineering and Computer Science*, vol 1, WCECS 2011, October 19-21, San Francisco, USA, 2011.
15. A. Bagheri, H. Mohammadi Peyhani, and M. Akbari, “Financial forecasting using ANFIS networks with Quantum-behaved Particle Swarm Optimization,” *Expert Syst. Appl.*, vol. 41, no. 14, pp. 6235–6250, 2014.
16. B. Bektas Ekici and U. T. Aksoy, “Prediction of building energy needs in early stage of design by using ANFIS,” *Expert Syst. Appl.*, vol. 38, no. 5, pp. 5352–5358, 2011.
17. Y.K. Bhateshvar , H. D. Mathur, H. Siguerdidjane and S. Bhanot , “Frequency Stabilization for Multi-area Thermal–Hydro Power System Using Genetic Algorithm-optimized Fuzzy Logic Controller in Deregulated Environment,”*Electric Power Components and Systems*, vol. 43, no. 2, pp. 146-156, 2015.
18. D.M. Boore and G.M. Atkinson, “Boore-Atkinson NGA ground motion relations for the geometric mean horizontal component of peak and spectral ground motion parameters,” PEER Report 2007/01, Pacific Engineering Research Center, University of California, Berkeley, CA, USA.
19. M. Böse, F. Wenzel and M. Erdik, “A neural network base approach to earthquake early warning for finite faults,” *BSSA*, vol. 98 (1), pp. 366-382, 2008.
20. K.D. Brabanter , P. Karsmakers, F. Ojeda, C. Alzate, J.D. Brabanter, K. Pelckmans, B.D. Moor , J. Vandewalle, J. A.K. Suykens, “LS-SVMlab Toolbox User’s Guide version 1.8,” *ESAT-SISTA Technical Report 10-146, August 2011*. http://www.esat.kuleuven.be/sista/lssvmlab/(seen on july 2015)

21. M. A. Boyacioglu and D. Avci, “An Adaptive Network-Based Fuzzy Inference System (ANFIS) for the prediction of stock market return: The case of the Istanbul Stock Exchange,” *Expert Syst. Appl.*, vol. 37, no. 12, pp. 7908–7912, 2010.
22. M. S Bazaraa, H.D Sherali and C.M Shetty, “Nonlinear Programming:Theory and Algorithms,” *2nd Edition Wiley*, 1993.
23. Y. Bozorgnia, “Pacific Earthquake Engineering Research Center PEER,”*University of California, Berkeley*, 2012. http://peer.berkeley.edu/ngawest2/databases.html. (seen on July 2015)
24. C. M. Bishop, “Pattern Recognition and Machine Learning,”*Springer-Verlag*, New York, 2006.
25. R. G. Brereton and G. R. Lloyd, “Support vector machines for classification and regression,” *Analyst*, vol. 135, no. 2, pp. 230–267, 2010.
26. C. Burges, “A Tutorial on Support Vector Machines for Pattern Recognition,” *Data Min. Knowl. Discov.*, vol. 2, no. 2, pp. 121–167, 1998.
27. A. F. Cabalar and A. Cevik, “Genetic programming-based attenuation relationship: An application of recent earthquakes in turkey,” *Comput. Geosci.*, vol. 35, no. 9, pp. 1884–1896, Sep. 2009.
28. A. F. Cabalar, A. Cevik, and C. Gokceoglu, “Some applications of Adaptive Neuro-Fuzzy Inference System (ANFIS) in geotechnical engineering,” *Comput. Geotech.*, vol. 40, pp. 14–33, 2012.
29. K. W. Campbell and Y. Bozorgnia, “Campbell-Bozorgnia NGA Ground Motion Relations for the Geometric Mean Horizontal Component of Peak and Spectral Ground Motion Parameters,” no. May, pp. 1–265, 2007.PEER Report 2007/02, Pacific Engineering Research Center, University of California, Berkeley, CA, USA.
30. S. Chakraverty, P. Gupta, and S. Sharma, “Neural network-based simulation for response identification of two-storey shear building subject to earthquake motion,” *Neural Comput. Appl.*, vol. 19, no. 3, pp. 367–375, 2010.
31. C. Chang and C. Lin, “LIBSVM : A Library for Support Vector Machines,” *ACM Trans. Intell. Syst. Technol.*, vol. 2, pp. 1–39, 2011.

32. C. S. Chen, M. Y. Cheng, and Y. W. Wu, “Seismic assessment of school buildings in Taiwan using the evolutionary support vector machine inference system,” *Expert Syst. Appl.*, vol. 39, no. 4, pp. 4102–4110, 2012.
33. C. L. P. Chen, “A rapid supervised learning neural network for function interpolation and approximation”, *IEEE Transactions on Neural Networks*, vol. 7, no. 5, pp. 12201230, 1996.
34. V. Cherkassky and Y. Ma, “Practical selection of SVM parameters and noise estimation for SVM regression,” *Neural Networks*, vol. 17, no. 1, pp. 113–126, 2004.
35. B.S.J.Chiou and R.R.Youngs, “An NGA model for the average horizontal component of peak ground motion and response spectra,”*Earthq. Spectra*, vol.24(1),pp.173–215, 2008.
36. J. Chorowski, J. Wang, and J. M. Zurada, “Review and performance comparison of SVM- and ELM-based classifiers,” *Neurocomputing*, vol. 128, pp. 507–516, 2014.
37. C. Cortes and V. Vapnik, “Support vector networks,”*Mach. Learn. J.*,vol. 20,pp. 273–297,1995.
38. J.S. S. Chou, M.Y. Y. Cheng, Y.W.W. Wu and A.D. D. Pham, “Optimizing parameters of support vector machine using fast messy genetic algorithm for dispute classification,” *Expert Syst. Appl.*, vol. 41, no. 8, pp. 3955–3964, 2014.
39. J. Douglas, “Earthquake ground motion estimation using strong-motion records: a review of equations for the estimation of peak ground acceleration and response spectral ordinates,” *Earth-Science Rev.*, vol. 61, no. 1–2, pp. 43–104, Apr. 2003.
40. H. Eristi, A. Ucar and Y. Demi, “Wavelet-based feature extraction and selection for classification of power system disturbances using support vector machines,” *Electric Power Syst. Res.*,vol.80(7),pp.743–752, 2010.
41. B. Erol and S. Erol, “Learning-based computing techniques in geoid modeling for precise height transformation,” *Comput. Geosci.*, vol. 52, pp. 95–107, 2013.
42. P. Fajfar and I. Perus, “A non-parametric approach to attenuation relations,”*J. of Earthq. Eng.*, vol. 1(2), pp. 319-340, 1997.
43. P. Fajfar, “How Reliable are the Ground Motion Prediction Equations?,”*SMiRT 20*, pp. 1–9, 2009.

44. A.H. Gandomi and A. H. Alavi. "A new multi-gene genetic programming approach to non-linear system modeling. Part II: geotechnical and earthquake engineering problems." *Neural Computing and Applications* 21.1: 189-201, 2012.
45. A. H. Gandomi, A. H. Alavi, M. Mousavi, and S. M. Tabatabaei, “A hybrid computational approach to derive new ground-motion prediction equations,” *Eng. Appl. Artif. Intell.*, vol. 24, no. 4, pp. 717–732, Jun. 2011.
46. G. Giacinto, R. Paolucci, and F. Roli, “Application of neural networks and statistical pattern recognition algorithms to earthquake risk evaluation,” *Pattern Recognit. Lett.*, vol. 18, no. 11, pp. 1353–1362, 1997.
47. K. Goda, F.Wenzel and R.D. Risi, “Empirical Assessment of Non-Linear Seismic Demand of Mainshock–Aftershock Ground-Motion Sequences for Japanese Earthquakes,” *Frontiers in Built Environment*, vol.1, 2015.
DOI: 10.3389/fbuil.2015.00006.
48. S. S. Gokhale and M. R. Lyu, “Regression Tree Modeling for the Prediction of Software Quality,” *Proc. Third ISSAT Int. Conf. Reliab. Qual. Des. Anaheim CA*, no. 2, pp. 31–36, 1997.
49. A.T.C Goh and S.H. Goh, “Support vector machines: their use in geotechnical engineering as illustrated using seismic liquefaction data”, *Comput. Geotech.*,vol. 34(5),pp.410–421,2007.
50. M. K. Goyal, B. Bharti, J. Quilty, J. Adamowski, and A. Pandey, “Modeling of daily pan evaporation in sub tropical climates using ANN, LS-SVR, Fuzzy Logic, and ANFIS,” *Expert Syst. Appl.*, vol. 41, no. 11, pp. 5267–5276, 2014.
51. H. Güllü, “On the prediction of shear wave velocity at local site of strong ground motion stations: an application using artificial intelligence,” *Bull. Earthq. Eng.*, vol. 11, no. 4, pp. 969–997, 2013.
52. H. Güllü, “Prediction of peak ground acceleration by genetic expression programming and regression: A comparison using likelihood-based measure,” *Eng. Geol.*, vol. 141–142, pp. 92–113, Jul. 2012.
53. H. Güllü, “Function finding via genetic expression programming for strength and elastic properties of clay treated with bottom ash,” *Eng. Appl. Artif. Intell.*, vol. 35, pp. 143–157, 2014.

54. H. Güllü, “Factorial experimental approach for effective dosage rate of stabilizer: application for fine-grained soil treated with bottom ash,” *Soils Found.*, vol. 54, no. 3, pp. 462–477, 2014.
55. H. Güllü and E. Erçelebi, “A neural network approach for attenuation relationships: An application using strong ground motion data from Turkey,” *Eng. Geol.*, vol. 93, no. 3–4, pp. 65–81, Aug. 2007.
56. H. Güllü and E. Erçelebi, “Reply to discussion by H. Sönmez and C. Gökceoglu on ‘A neural network approach for attenuation relationships: An application using strong ground motion data from Turkey’ by H. Güllü and E. Erçelebi, Eng. Geol. 93 (2007) 65–81,” *Eng. Geol.*, vol. 97, no. 1–2, pp. 94–96, Mar. 2008.
57. H. Güllü and M. Pala, “On the resonance effect by dynamic soil–structure interaction: a revelation study,” *Nat. Hazards*, vol. 72, no. 2, pp. 827–847, 2014.
58. K. Günaydın and A. Günaydın, “Peak Ground Acceleration Prediction by Artificial Neural Networks for Northwestern Turkey,” *Math. Probl. Eng.*, vol. 2008, pp. 1–20, 2008.
59. M. Han and C. Liu, “Endpoint prediction model for basic oxygen furnace steel-making based on membrane algorithm evolving extreme learning machine,” *Appl. Soft Comput. J.*, vol. 19, pp. 430–437, 2013.
60. S. Haykin, “Neural Networks a Comprehensive Foundation,”*Englewood Cliffs,NJ*: *Prentice-Hall*, 1999.
61. S. Haykin, “Neural Networks and learning machines”, Pearson Education Inc.,New Jersey, 2009.
62. K. Hornik, “Approximation capabilities of multilayer feedforward networks,” *Neural Networks*, vol. 4, no. 2, pp. 251–257, 1991.
63. M. Hosoz, H. M. Ertunc, M. Karabektas, and G. Ergen, “ANFIS modelling of the performance and emissions of a diesel engine using diesel fuel and biodiesel blends,” *Appl. Therm. Eng.*, vol. 60, no. 1–2, pp. 24–32, 2013.
64. A.H.Huang, Z.Y.Dong and Y.Wang, “Power utility non technical loss analysis with extreme learning machine method,” *IEEE Trans.Power Syst*.,vol.23(3),pp.946–955,2008.
65. G.B. Huang and D. Wang, “Advances in Extreme Learning Machines (ELM2011),” *Neurocomputing*, vol. 102, pp. 1–2, 2013.

66. G.B.Huang, Q.Y.Zhu, and C.K.Siew, “Extreme learning machine: theory and applications,”*Neurocomputing* , vol.70, pp.489–501, 2006.
67. I.M.Idriss, “An NGA empirical model for estimating the horizontal spectral values generated by shallow crustal earthquakes,”*Earthq. Spectra*, vol.24(1) ,pp. 217–242, 2008.
68. Y. Jafarian, E. Kermani, and M. H. Baziar, “Empirical predictive model for the vmax/amax ratio of strong ground motions using genetic programming,” *Comput. Geosci.*, vol. 36, no. 12, pp. 1523–1531, Dec. 2010.
69. P. Jagtap, G.N.Pillai, "Comparison of extreme-ANFIS and ANFIS networks for regression problems," in *Advance Computing Conference (IACC), 2014 IEEE International* ,pp.1190-1194, 21-22 Feb. 2014.
70. J.S. R. Jang, “ANFIS : Adaptive-Network-Based Fuzzy Inference System,” *IEEE Trans. Syst. Man. Cybern.*, vol. 23, no. 3, pp. 665–685, 1993.
71. S. Karpagachelvi, M. Arthanari, and M. Sivakumar, “Classification of electrocardiogram signals with support vector machines and extreme learning machine,” *Neural Comput. Appl.*, vol. 21pp. 1331–1339, 2012.
72. Y. Kaya and M. Uyar, “A hybrid decision support system based on rough set and extreme learning machine for diagnosis of hepatitis disease,” *Appl. Soft Comput.*, vol. 13, no. 8, pp. 3429–3438, 2013.
73. T. Kerh and D. Chu, “Neural networks approach and microtremor measurements in estimating peak ground acceleration due to strong motion,” *Adv. Eng. Softw.*, vol. 33, no. 11–12, pp. 733–742, Nov. 2002.
74. T. Kerh, T. Ku, and D. Gunaratnam, “Evaluations of the Strong Ground Motion Parameter by Neural Computing and Microtremor Measurement,” pp. 2–8, 2008, DOI:10.1109/AICCSA.2010.5587031.
75. T. Kerh and S. B. Ting, “Neural network estimation of ground peak acceleration at stations along Taiwan high-speed rail system,” *Eng. Appl. Artif. Intell.*, vol. 18, no. 7, pp. 857–866, Oct. 2005.
76. E. Kermani, Y. Jafarian, and M. H. Baziar, “New Predictive Models for the v max/a max Ratio of Strong Ground Motions using Genetic Programming,”*Int. Jour. Civil. Eng.* vol. 7, no. 4, pp. 236–247, 2009.

77. S.S Keerthi, S.K. Shevade, C. Bhattacharyya and K.R.K Murty, “Improvements to platt’s SMO algorithm for SVM classifier design,” *Neural Comput.J.*,vol. 13,pp. 637–649,2001.
78. A. U. Keskin, D. G. Yavuz, O. Dag, O. Deyneli, H. Aydın, O. Tarçın, E. Morkoyun, and S. Akalin, "Artificial Neural Network-Based Diagnosis And Classification Of Thyroid Disorders," *30th Annual Meeting of The European Thyroid Association Congress Istanbul*, 19-22 September 2004.
79. A. U. Keskin, and I. Göker, "A short note on the application of Cholesky factorization in MATLAB,"*Istanbul University, Journal of Electrical & Electronics Engineering* (IU-JEEE), Vol. 6, (1), pp. 113-115, 2006.
80. A. U. Keskin, Comments on the "Sensor calibration and compensation using artificial neural network,"*ISA Transactions - (Instrumentation, Systems, and Automation, Elsevier)*, vol. 48, Issue 2, pp. 143-144, 2009.
81. M. F. Khan, E. Khan and Z. A. Abbasi, “Multi Segment Histogram Equalization for Brightness Preserving Contrast Enhancement,”*Book chapter published by Springer Verlag in the Advances in Intelligent and Soft Computing*, vol. 166, *Advances in Computer Science, Engineering & Applications*, pp. 193-202, 2012. ISBN: 978-3-642-02899-1.
82. M. F. Khan, X. Ren and E. Khan, “Semi Dynamic Fuzzy Histogram Equalization,” *Optik-International Journal of Light and Electron Optics*, vol. 126, no. 21, pp. 2848-2853, 2015.
83. S. R. Khuntia and S. Panda, “Simulation study for automatic generation control of a multi-area power system by ANFIS approach,” *Appl. Soft Comput. J.*, vol. 12, no. 1, pp. 333–341, 2012.
84. T. R. Kiran and S. P. S. Rajput, “An effectiveness model for an indirect evaporative cooling (IEC) system: Comparison of artificial neural networks (ANN), adaptive neuro-fuzzy inference system (ANFIS) and fuzzy inference system (FIS) approach,” *Appl. Soft Comput.*, vol. 11, no. 4, pp. 3525–3533, 2011.
85. H.W.Kuhn and A.W. Tucker, “Nonlinear programming,”*In: Proc. 2nd Berkeley Symposium on Mathematical Statistics and Probabilistics*, Berkeley. University of California Press, pp. 481–492,1951.

86. B. Kurtulus and N. Flipo, “Hydraulic head interpolation using anfis-model selection and sensitivity analysis,” *Comput. Geosci.*, vol. 38, no. 1, pp. 43–51, 2012.
87. S. C. Lemon, J. Roy, M. A. Clark, P. D. Friedmann, and W. Rakowski, “Classification and regression tree analysis in public health: methodological review and comparison with logistic regression.,” *Ann. Behav. Med.*, vol. 26, no. 3, pp. 172–181, 2003.
88. R. J. Lewis “An Introduction to Classification and Regression Tree (CART) Analysis,” *2000 Annu. Meet. Soc. Acad. Emerg. Med.*, 310, p. 14p, 2000.
89. L. Li-Xia, Z. Yi-Qi, and X. Y. Liu, “Tax forecasting theory and model based on SVM optimized by PSO,” *Expert Syst. Appl.*, vol. 38, no. 1, pp. 116–120, 2011.
90. D. Lowe, “Adaptive radial basis function nonlinearities, and the problem of generalization”, *In Proc. 1st IEE.Int. Conf. Artificial Neural Networks*, pp. 171-175,1989.
91. O.L. Mangasarian,“ Multi-surface method of pattern separation,”*IEEE Trans.on Inform. Theo. IT*, vol.14, pp. 801–807, 1968.
92. R.Manjulasri and K.M.M. Rao, "Review of image processing and tele ophthalmology research projects world over in Technology,” *Spectrum Journal*, pp.42-48, 2010.
93. R.ManjulaSri and K.M.M. Rao, "Novel Image Processing Techniques to Detect Lesions using Lab View,"*IEEE Conference Publications on Annual IEEE India Conference* INDICON, pp. 1-4, 2011.
94. R.ManjulaSri, Ch.Madhubabu and K.M.M. Rao, “Automatic detection of Age Related Macular Degeneration from Retinal Images,”*Technia-International Journal of Computing Science and Communication Technologies*, vol.5, no. 2, 2013.
95. S. Markic and V. Stankovski, “Engineering Applications of Artificial Intelligence An equation-discovery approach to earthquake-ground-motion prediction,” vol. 26, pp. 1339–1347, 2013.
96. A. Maravall A, and V.Gomez, Eviews Software, Version 5, Quantitative Micro Software. Irvine C.A.: LLC; 2004.
97. B. Metin, G. Goksu, and C. Hulagu, “Information Systems for Electronic Education Infrastructure,”*Proc. of 13th IEEE International Symposium on Computational Intelligence and Informatics* (CINTI 2012), pp.109-113, Budapest, Hungary, 2012.
98. B. Metin and T. Yildirim, “Internal Control Design for Information Systems with Control Self-Assessment Method,”*IEEE 11th International Symposium on Applied Machine*

Intelligence and Informatics (SAMI 2013), pp. 227-231, 2013, Herl'any, Slovakya. DOI: 0.1109/SAMI.2013.6480981.

99. P. Melin, J. Soto, O. Castillo, and J. Soria, "A new approach for time series prediction using ensembles of ANFIS models," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 3494–3506, 2012.
100. A.K. Mohammadnejad, S.M. Mousavi, M.Torabi, M. Mousavi and A. H Alavi, "Robust attenuation relations for peak time-domain parameters of strong ground motions," *Environ Earth Sci*,vol.67,pp.3–70, 2012. DOI: 10.1007/s12665-011-1479-9.
101. R. K. Nagaria and J. P. Saini, "Fuzzy Logic Concepts and Techniques for Engineering with an Application to PID Controller,"*Proc. Sixth Annual Conference of Vijnana Parishad of India*, B.I.E.T., Jhansi (U.P.), pp.52, Dec. 26-28, 1996.
102. R.K. Nagaria and A.K.Singh, " Tuning and Control of Fuzzy Systems using Parameter Estimation Technique," *Proc. National Conference on Impact of Electronics on Automation*: The Indian Scenario, University of Roorkee, Roorkee, Jan.20-21, 1997.
103. H.J. Oh and B. Pradhan, "Application of a neuro-fuzzy model to landslide-susceptibility mapping for shallow landslides in a tropical hilly area," *Comput. Geosci.*, vol. 37, no. 9, pp. 1264–1276, 2011.
104. G. Özkan and M. İnal, "Comparison of neural network application for fuzzy and ANFIS approaches for multi-criteria decision making problems," *Appl. Soft Comput.*, vol. 24, pp. 232–238, 2014.
105. M. Pal, "Support vector machines-based modelling of seismic liquefaction potential," *Int J Numer Anal Methods Geomech*, vol.30, no.10,pp.983–996,2006.
106. M. Pal and G. M. Foody, "Feature Selection for Classification of Hyper spectral Data by SVM", *IEEE trans. geosci. and remote sens.*, vol. 48, no. 5, 2010.
107. M. Pal and P. M. Mather, "Support vector machines for classification in remote sensing," *Int. J. of Rem. Sens.*, vol. 26, no. 5, pp. 1007-1011, 2005. DOI**:**10.1080/01431160512331314083.
108. Y. H. Pao, G. H. Park, and D. J. Sobajic, "Learning and generalization characteristics of the random vector Functional-link net," *Neurocomputing*, vol. 6, pp. 163–180, 1994.

109. I. Park, J. Choi, M. Jin Lee, and S. Lee, “Application of an adaptive neuro-fuzzy inference system to ground subsidence hazard mapping,” *Comput. Geosci.*, vol. 48, pp. 228–238, 2012.

110. J. Park and I. W. Sandberg, “Universal Approximation Using Radial-Basis-Function Networks,” *Neural Comput.*, vol. 3, no. 2, pp. 246–257, 1991.

111. M.B. Power, N.A. Chiou, N.Abrahamson, and C.Roblee, “The next generation of ground motion attenuation models, NGA project: an overview,*” In: Proceeding of the 8th National Conference on Earthquake Engineering of the 8th National Conference of Earthquake Engineering*, San Francisco, (2006). paper no. 2022

112. S. G. Patil, S. Mandal, and A. V Hegde, “Genetic algorithm based support vector machine regression in predicting wave transmission of horizontally interlaced multi-layer moored floating pipe breakwater,” *Adv. Eng. Softw.*, vol. 45, no. 1, pp. 203–212, 2012.

113. I. Perus and P. Fajfar, “ Ground-motion prediction by a non-parametric approach”, *Earthquake Engng. Struct. Dyn.*, 39: 1395–1416. DOI:10.1002/eqe.1007. 2010.

114. G.N. Pillai, “Extreme ANFIS: A New Learning Machine for Faster Learning”, *Proceedings on IEEE international conference on knowledge collaboration in engineering*, Coimbatore India, January 2014.

115. G.N. Pillai, P. Jagtap and M. Nisha "Extreme learning ANFIS for control applications." In *Computational Intelligence in Control and Automation (CICA), 2014 IEEE Symposium on*, pp. 1-8. IEEE, 2014.

116. B. Pradhan, “A comparative study on the predictive ability of the decision tree, support vector machine and neuro-fuzzy models in landslide susceptibility mapping using GIS,” *Comput. Geosci.*, vol. 51, pp. 350–365, 2013.

117. G. Quaranta, G. C. Marano, R. Greco, and G. Monti, “Parametric identification of seismic isolators using differential evolution and particle swarm optimization,” *Appl. Soft Comput.J*, vol. 22, pp. 458–464, 2014.

118. R. Rajavel and P. S. Sathidevi, "A new GA optimized reliability ratio based integration weight estimation scheme for decision fusion audio-visual speech recognition," *Journal of Signal and Imaging Systems Engineering*, vol. 4, no. 2, pp. 123-131, 2011. DOI:10.1504/IJSISE.2011.041605.

119. V. Ranaee, A. Ebrahimzadeh, and R. Ghaderi, “Application of the PSO-SVM model for recognition of control chart patterns,” *ISA Trans.*, vol. 49, no. 4, pp. 577–86, 2010.
120. K.M.M. Rao," Image Processing For Medical Applications," *Proc. of the 14th World Conference on Non- Destructive Testing*, Delhi, vol. 2, pp. 1219-1224, 8-13 Dec. 1996.
121. Z. Ren, Z. Zhang, T. Chen and W. Wang, “Theoretical and quantitative analyses of the fault slip rate uncertainties from single event and erosion of the accumulated offset,”*Island Arc*, vol.22, pp.185–196, 2013. DOI: 10.1111/iar.12015.
122. Z. Ren , Z. Zhang ,T. Chen, S.Yan, J. Yin, P. Zhang , W. Zheng, H. Zhang and C. Li, “Clustering of offsets on the Haiyuan fault and their relationship to paleo earthquakes,” *Geological Society of America Bulletin* . 06/2015. DOI: 10.1130/B31155.1
123. Z.Ren , Z.Zhang, J. Yin, F. Dai and H. Zhang, “Morphogenic uncertainties of the 2008 Wenchuan Earthquake: Generating or reducing?,”*Journal of Earth Science*, vol. 25, no. 4, pp. 668-675, 2014.
124. J. L. Rojo-Álvarez, G. Camps-Valls, M. Martínez-Ramón, E. Soria-Olivas, A. Navia-Vázquez, and A. R. Figueiras-Vidal, “Support vector machines framework for linear signal processing,” *Signal Processing*, vol. 85, no. 12, pp. 2316–2326, 2005.
125. P. Samui, “Support vector machine applied to settlement of shallow foundations on cohesionless soils,” *Comput. Geotech.*, vol. 35(3), pp. 419–427, 2008.
126. P. Samui and Kurup P, “Multivariate adaptive regression spline (MARS) and least squares support vector machine (LSSVM) for OCR prediction,”*Soft. Comput*., vol.16, pp.1347–1351, 2012.
127. V. D. Sánchez A, “Advanced support vector machines and kernel methods,” *Neurocomputing*, vol. 55, no. 1–2, pp. 5–20, 2003.
128. W. F. Schmidt, M. A. Kraaijveld, and R. P. W. Duin, “Feed Forward Neural Networks with Random Weights,” *IEEE, ICPR*, pp. 1–4, 1992.
129. B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, “New Support Vector Algorithms,” *Neural Comput.*, vol. 12, no. 5, pp. 1207–1245, 2000.
130. M. Segou and N. Voulgaris, “Proschema: A Matlab application for processing strong motion records and estimating earthquake engineering parameters,” *Comput. Geosci.*, vol. 36, no. 7, pp. 977–986, 2010.

131. R. Sehhati, A. Rodriguez-Marek, M. ElGawady, and W. F. Cofer, “Effects of near-fault ground motions and equivalent pulses on multi-story structures,” *Eng. Struct.*, vol. 33, no. 3, pp. 767–779, 2011.

132. S. M. Seyedpoor, J. Salajegheh, E. Salajegheh, and S. Gholizadeh, “Optimal design of arch dams subjected to earthquake loading by a combination of simultaneous perturbation stochastic approximation and particle swarm algorithms,” *Appl. Soft Comput. J.*, vol. 11, no. 1, pp. 39–48, 2011.

133. H.Singh and S. Barada, “Generating Optimal,adaptive fuzzy neural models of dynamical systems with applications to control,” *IEEE Trans. on systems, man and cybernaetics,* vol. 28, No. 3, pp.371-391, 1998.

134. H.Singh, L. Arefeh and S. Putututunda, “A Neuro Fuzzy logic approach to material processing,” *IEEE Trans. Syst. Man. Cybern., part C: Applications and reviews,* vol. 29, No. 3, pp.362-370, 1999.

135. H.Singh and S. Barada “Neural networks for image compression in target detection,”*Optical Engineering, Society of Photo-Optical Instrumentation Engineers*, vol. 37(7), pp. 2029-2042, 1998.

136. H.Singh ,T. Meitzler, E. Sohn , A. Elgarhi and D. Nam, “Predicting search times in visually cluttered scenes using Fuzzy Logic Approach,”*Optical Engineering*, vol. 40, no. 9, pp. 1844-1851, 2001.

137. R.Singh, A. Kainthola and T.N Singh, “Estimation of elastic constant of rocks using an ANFIS approach,”*Appl. Soft Comput.J*, vol.12,pp. 40–45,2012.

138. V. Raviprasad and R. K. Singh, “Optimal sizing of PV array for critical load with parallel redundant architecture using GA,” *1st International Conference on Automation, Control, Energy and Systems* (ACES), pp. 1-7, 2014.

139. M. Shinozuka, G. Deodatis, R. Zhang, and A. S. Papageorgiou, “Modeling, synthetics and engineering applications of strong earthquake wave motion,” *Soil Dyn. Earthq. Eng.*, vol. 18, no. 3, pp. 209–228, 1999.

140. J. Shiri, O. Kisi, H. Yoon, K.-K. Lee, and A. H. Nazemi, “Predicting groundwater level fluctuations with meteorological effect implications-{A} comparative study among soft computing techniques,” *Comput. Geosci.*, vol. 56, pp. 32–44, 2013.

141. J. Smola and B. Schölkopf, “A tutorial on support vector regression,” *Stat. Comput.*, vol. 14, pp. 199–222, 2004.

142. P.Smit, V. Arzoumanian, Z.Javakhishvili, S. Arefiev, D.Mayer-Rosa, S.Balassanian, and T.Chelidze, “The digital accelerograph network in the Caucasus,”*In: Proceedings of the 2nd International Conference on Earthquake Hazard and Seismic Risk Reduction-Advances in Natural and Technological Hazards Research*, Kluwer Academic Publishers, Yerevan, Armenia(2000).

143. G.N. Smith, Probability and statistics in civil engineering. London: Collins,1986.

144. P.Spandana, K.M.M. Rao, and S.N. P. Rao, “Automated Approach for Qualitative Assesment of Breast Density and Lesion Feature Extraction for Early Detection of Breast Cancer,” *IJCSCT Technia Journal*, vol. 6,no. 1, 2013.ISSN 0974-3375.

145. L. Stothers, R. Guevara, and A. Macnab, “Classification of male lower urinary tract symptoms using mathematical modelling and a regression tree algorithm of noninvasive near-infrared spectroscopy parameters,” *Eur. Urol.*, vol. 57, no. 2, pp. 327–32, 2010.

146. V. Sokolov and F.Wenzel, “On the modeling of ground-motion field for assessment of multiple-location hazard, damage, and loss: Example of estimation of electric network performance during scenario earthquake,”*Nat. Haz.*, vol.74(3), 2014. DOI: 10.1007/s11069-014-1262-9.

147. Y. Song and P. Lio, “A new approach for epileptic seizure detection: sample entropy based feature extraction and extreme learning machine,” *J. Biomed. Sci. and Engg.,vol.* 3,pp.556-567, 2010.

148. J. A.K. Suykens, J. De Brabanter , L. Lukas, J. Vandewalle,“ Weighted least squares support vector machines: robustness and sparse approximation”, *Neurocomputing*.Vo1.48 (14): p85-105. 2002.

149. Z.L. Sun, T.M.Choi, K.F.Au,Y.Yu, “Sales forecasting using extreme learning machine with applications in fashion retailing,” *Decis. Supp. Syst.*, vol. 46(1), pp. 411–419, 2008.

150. D.Tien Bui, B. Pradhan, O. Lofman, I. Revhaug, and O. B. Dick, “Landslide susceptibility mapping at Hoa Binh province (Vietnam) using an adaptive neuro-fuzzy inference system and GIS,” *Comput. Geosci.*, vol. 45, pp. 199–211, 2012.

151. A.Thakur , S. Bhanot , S.N.Mishra, “Early Diagnosis of Ischemia Stroke using Neural Network,” *Proc. of the International Conference on Man-Machine Systems* (ICoMMS), 11–13October, Batu Ferringhi, Penang, Malaysia, 2009.

152. V. Vapnik, A.Chervonenkis. *Theory of Pattern Recognition* [in Russian]. Nauka, Moscow. 1974. (German Translation: Wapnik W. & Tscherwonenkis A., *Theorie der Zeichenerkennung*, Akademie-Verlag, Berlin, 1979).

153. V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.

154. V. Vapnik. *Statistical Learning Theory*. John Wiley and Sons, Inc., New York, 1998.

155. L. P. Wang and Chunru Wan, “Comments on the extreme learning machine,” IEEE Trans. Neural Networks, vol.19, no.8, pp. 14941495, 2008.

156. F. Wang, Z. Zhen, Z. Mi, H. Sun, S. Su, and G. Yang, “Solar irradiance feature extraction and support vector machines based weather status pattern recognition model for short-term photovoltaic power forecasting,” *Energy Build.*, vol. 86, pp. 427–438, 2015.

157. P. Xanthopoulos and T. Razzaghi, “A weighted support vector machine method for control chart pattern recognition,” *Comput. Ind. Eng.*, vol. 70, pp. 134–149, 2014.

158. X. Yan and N. a. Chowdhury, “Mid-term electricity market clearing price forecasting: A multiple SVM approach,” *Int. J. Electr. Power Energy Syst.*, vol. 58, pp. 206–214, 2014.

159. J. Ye and T. Xiong, “SVM versus Least Squares SVM,” *J. Mach. Learn. Res.Proc. Track*, vol. 2, pp. 644–651, 2007.

160. X. Zhang, G. Wu, Z. Dong, and C. Crawford, “Embedded feature-selection support vector machine for driving pattern recognition,” *J. Franklin Inst.*, vol. 352, no. 2, pp. 669–685, 2015.

161. H. Zhong, C. Miao, Z. Shen, and Y. Feng, “Comparing the learning effectiveness of BP, ELM, I-ELM, and SVM for corporate credit ratings,” *Neurocomputing*, vol. 128, pp. 285–295, 2014.

162. J. Zhou, X. Li, and X. Shi, “Long-term prediction model of rockburst in underground openings using heuristic algorithms and support vector machines,” *Saf. Sci.*, vol. 50, no. 4, pp. 629–644, 2012.

163. M. Zuhair, S. Thomas, A.K.Keshri, R.K. Sinha, K.Pal and D.Biswas, “Automatic Identification of an Epileptic Spike Pattern in an EEG Signal using ANN”, *Proc. of the Third International Conference on Soft Computing for Problem Solving,*Advances in Intelligent Systems and Computing 258,Springer India 2014. DOI: 10.1007/978-81-322-1771-8_79.

APPENDIX A

The CB-NGA median ground motion model is given by the general equation

$$\widehat{\ln Y} = f_{mag} + f_{dis} + f_{flt} + f_{hng} + f_{site} + f_{sed} \tag{3.1}$$

where the magnitude term is given by

$$f_{mag} = \begin{cases} c_0 + c_1\mathbf{M}; & \mathbf{M} \le 5.5 \\ c_0 + c_1\mathbf{M} + c_2(\mathbf{M} - 5.5); & 5.5 < \mathbf{M} \le 6.5 \\ c_0 + c_1\mathbf{M} + c_2(\mathbf{M} - 5.5) + c_3(\mathbf{M} - 6.5); & \mathbf{M} > 6.5 \end{cases}, \tag{3.2}$$

the distance term is given by

$$f_{dis} = (c_4 + c_5\mathbf{M}) \ln\left(\sqrt{R_{RUP}^2 + c_6^2}\right), \tag{3.3}$$

the style-of-faulting term is given by

$$f_{flt} = c_7 F_{RV} f_{flt,Z} + c_8 F_{NM}, \tag{3.4}$$

$$f_{flt,Z} = \begin{cases} Z_{TOR}; & Z_{TOR} < 1 \\ 1; & Z_{TOR} \ge 1 \end{cases}, \tag{3.5}$$

the hanging-wall term is given by

$$f_{hng} = c_9 f_{hng,R} f_{hng,M} f_{hng,Z} f_{hng,\delta}, \tag{3.6}$$

$$f_{hng,R} = \begin{cases} 1; & R_{JB} = 0 \\ \left[\max\left(R_{RUP}, \sqrt{R_{JB}^2 + 1}\right) - R_{JB}\right] \Big/ \max\left(R_{RUP}, \sqrt{R_{JB}^2 + 1}\right); & R_{JB} > 0, Z_{TOR} < 1 \\ (R_{RUP} - R_{JB}) / R_{RUP}; & R_{JB} > 0, Z_{TOR} \ge 1 \end{cases}, \tag{3.7}$$

$$f_{hng,M} = \begin{cases} 0; & \mathbf{M} \le 6.0 \\ 2(\mathbf{M} - 6.0); & 6.0 < \mathbf{M} < 6.5 \\ 1; & \mathbf{M} \ge 6.5 \end{cases}, \tag{3.8}$$

$$f_{hng,Z} = \begin{cases} 0; & Z_{TOR} \ge 20 \\ (20 - Z_{TOR})/20; & 0 \le Z_{TOR} < 20 \end{cases}, \tag{3.9}$$

$$f_{hng,\delta} = \begin{cases} 1; & \delta \le 70 \\ (90 - \delta)/20; & \delta > 70 \end{cases}, \tag{3.10}$$

the shallow site response term is given by

$$f_{site} = \begin{cases} c_{10}\ln\left(\frac{V_{S30}}{k_1}\right) + k_2\left\{\ln\left[A_{1100} + c\left(\frac{V_{S30}}{k_1}\right)^n\right] - \ln\left[A_{1100} + c\right]\right\}; & V_{S30} < k_1 \\ (c_{10} + k_2 n)\ln\left(\frac{V_{S30}}{k_1}\right); & k_1 \le V_{S30} < 1100 \\ (c_{10} + k_2 n)\ln\left(\frac{1100}{k_1}\right); & V_{S30} \ge 1100 \end{cases}, \tag{3.11}$$

and the deep site response term is given by

$$f_{sed} = \begin{cases} c_{11}(Z_{2.5} - 1); & Z_{2.5} < 1 \\ 0; & 1 \le Z_{2.5} \le 3 \\ c_{12}k_3 e^{-0.75}\left[1 - e^{-0.25(Z_{2.5}-3)}\right]; & Z_{2.5} > 3 \end{cases}. \tag{3.12}$$

Table 3.1 Coefficients for CB-NGA median ground motion model.

T	c_0	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	c_{11}	c_{12}	k_1	k_2	k_3
0.010	-1.715	0.500	-0.530	-0.262	-2.118	0.170	5.60	0.280	-0.120	0.490	1.058	0.040	0.610	865	-1.186	1.839
0.020	-1.680	0.500	-0.530	-0.262	-2.123	0.170	5.60	0.280	-0.120	0.490	1.102	0.040	0.610	865	-1.219	1.840
0.030	-1.552	0.500	-0.530	-0.262	-2.145	0.170	5.60	0.280	-0.120	0.490	1.174	0.040	0.610	908	-1.273	1.841
0.050	-1.209	0.500	-0.530	-0.267	-2.199	0.170	5.74	0.280	-0.120	0.490	1.272	0.040	0.610	1054	-1.346	1.843
0.075	-0.657	0.500	-0.530	-0.302	-2.277	0.170	7.09	0.280	-0.120	0.490	1.438	0.040	0.610	1086	-1.471	1.845
0.10	-0.314	0.500	-0.530	-0.324	-2.318	0.170	8.05	0.280	-0.099	0.490	1.604	0.040	0.610	1032	-1.624	1.847
0.15	-0.133	0.500	-0.530	-0.339	-2.309	0.170	8.79	0.280	-0.048	0.490	1.928	0.040	0.610	878	-1.931	1.852
0.20	-0.486	0.500	-0.446	-0.398	-2.220	0.170	7.60	0.280	-0.012	0.490	2.194	0.040	0.610	748	-2.188	1.856
0.25	-0.890	0.500	-0.362	-0.458	-2.146	0.170	6.58	0.280	0.000	0.490	2.351	0.040	0.700	654	-2.381	1.861
0.30	-1.171	0.500	-0.294	-0.511	-2.095	0.170	6.04	0.280	0.000	0.490	2.460	0.040	0.750	587	-2.518	1.865
0.40	-1.466	0.500	-0.186	-0.592	-2.066	0.170	5.30	0.280	0.000	0.490	2.587	0.040	0.850	503	-2.657	1.874
0.50	-2.569	0.656	-0.304	-0.536	-2.041	0.170	4.73	0.280	0.000	0.490	2.544	0.040	0.883	457	-2.669	1.883
0.75	-4.844	0.972	-0.578	-0.406	-2.000	0.170	4.00	0.280	0.000	0.490	2.133	0.077	1.000	410	-2.401	1.906
1.0	-6.406	1.196	-0.772	-0.314	-2.000	0.170	4.00	0.255	0.000	0.490	1.571	0.150	1.000	400	-1.955	1.929
1.5	-8.692	1.513	-1.046	-0.185	-2.000	0.170	4.00	0.161	0.000	0.490	0.406	0.253	1.000	400	-1.025	1.974
2.0	-9.701	1.600	-0.978	-0.236	-2.000	0.170	4.00	0.094	0.000	0.371	-0.456	0.300	1.000	400	-0.299	2.019
3.0	-10.556	1.600	-0.638	-0.491	-2.000	0.170	4.00	0.000	0.000	0.154	-0.820	0.300	1.000	400	0.000	2.110
4.0	-11.212	1.600	-0.316	-0.770	-2.000	0.170	4.00	0.000	0.000	0.000	-0.820	0.300	1.000	400	0.000	2.200
5.0	-11.684	1.600	-0.070	-0.986	-2.000	0.170	4.00	0.000	0.000	0.000	-0.820	0.300	1.000	400	0.000	2.291
7.5	-12.505	1.600	-0.070	-0.656	-2.000	0.170	4.00	0.000	0.000	0.000	-0.820	0.300	1.000	400	0.000	2.517
10.0	-13.087	1.600	-0.070	-0.422	-2.000	0.170	4.00	0.000	0.000	0.000	-0.820	0.300	1.000	400	0.000	2.744
PGA	-1.715	0.500	-0.530	-0.262	-2.118	0.170	5.60	0.280	-0.120	0.490	1.058	0.040	0.610	865	-1.186	1.839
PGV	0.954	0.696	-0.309	-0.019	-2.016	0.170	4.00	0.245	0.000	0.358	1.694	0.092	1.000	400	-1.955	1.929
PGD	-5.270	1.600	-0.070	0.000	-2.000	0.170	4.00	0.000	0.000	0.000	-0.820	0.300	1.000	400	0.000	2.744

Note: $c = 1.88$ and $n = 1.18$ for all periods; PGA and PSA have units of g; PGV and PGD have units of cm/s and cm, respectively.

APPENDIX B

1	Earthquake Name	Station Name	Station Sequence Number	Station ID No.	Earthquake Magnitude	Magnitude Type	Magnitude Uncertainty: Kagan Model	Magnitude Uncertainty: Statistical	Magnitude Sample Size	Magnitude Uncertainty: Study Class	Mo (dyne.cm)	Strike (deg)	Dip (deg)	Rake Angle (deg)	Mechanism Based on Rake Angle	P-plunge (deg)
2	Helena, Montana-01	Carroll College	197	2022	6.00		0.3000			0.3	1.1220E+25	268	75	160	0	2.9
3	Helena, Montana-02	Helena Fed Bldg	198	2229	6.00	U	0.3000			0.3	1.1220E+25					
4	Humbolt Bay	Ferndale City Hall	133	1023	5.80		0.3000			0.3	5.6234E+24					
5	Imperial Valley-01	El Centro Array #9	75	117	5.00	U	0.3000			0.3	3.5481E+23					
6	Northwest Calif-01	Ferndale City Hall	133	1023	5.50		0.3000			0.3	1.9953E+24					
7	Imperial Valley-02	El Centro Array #9	75	117	6.95		0.3000	0.072	8	0.2	2.9864E+26	323	80	180	0	7.0
8	Northwest Calif-02	Ferndale City Hall	133	1023	6.60	U	0.3000			0.3	8.9125E+25					
9	Northern Calif-01	Ferndale City Hall	133	1023	6.40	U	0.3000			0.3	4.4668E+25					
10	Borrego	El Centro Array #9	75	117	6.50	U	0.3000			0.3	6.3096E+25	40	90	0	0	0.0
11	Imperial Valley-03	El Centro Array #9	75	117	5.60	U	0.3000			0.3	2.8184E+24	325	90	0	0	0.0
12	Northwest Calif-03	Ferndale City Hall	133	1023	5.80		0.3000			0.3	5.6234E+24					
13	Kern County	LA - Hollywood Stor FF	326	24303	7.36		0.3000	0.145	9	0.2	1.2303E+27	51	75	61	2	24.5
14	Kern County	Pasadena - CIT Athena	499	80053	7.36		0.3000	0.145	9	0.2	1.2303E+27	51	75	61	2	24.5
15	Kern County	Santa Barbara Courtho	92	283	7.36		0.3000	0.145	9	0.2	1.2303E+27	51	75	61	2	24.5
16	Kern County	Taft Lincoln School	148	1095	7.36		0.3000	0.145	9	0.2	1.2303E+27	51	75	61	2	24.5

1	Earthquake Name	Station Name	P-trend (deg)	T-plunge (deg)	T-trend (deg)	Hypocenter Latitude (deg)	Hypocenter Longitude (deg)	Hypocenter Depth (km)	Coseismic Surface Rupture: 1=Yes; 0=No; 99=Unknown	Coseismic Surface Rupture (Including Inferred)	Basis for Inference of Surface Rupture	Finite Rupture Model: 1=Yes; 0=No	Dept to Top Of Fault Rupture Model	Fault Rupture Length (km)	Fault Rupture Width (km)
2	Helena, Montana-01	Carroll College	316.3	24.6	225.0	46.6100	-111.9600	6	0	0	sfdoc	0			
3	Helena, Montana-02	Helena Fed Bldg				46.6200	-111.9700		0	0	sfdoc	0			
4	Humbolt Bay	Ferndale City Hall				40.4000	-125.1000		99	0	M<6	0			
5	Imperial Valley-01	El Centro Array #9				32.9000	-115.2170	16	99	0	M<6	0			
6	Northwest Calif-01	Ferndale City Hall				40.3000	-124.8000		99	0	M<6	0			
7	Imperial Valley-02	El Centro Array #9	187.5	7.1	278.4	32.7601	-115.4162	8.8	1	1		1	0	63.0	13.0
8	Northwest Calif-02	Ferndale City Hall				40.7000	-125.4000		99	99		0			
9	Northern Calif-01	Ferndale City Hall				40.4000	-124.8000		99	99		0			
10	Borrego	El Centro Array #9	175.0	0.0	85.0	33.0490	-116.0880		99	0	locdis	0			
11	Imperial Valley-03	El Centro Array #9	100.0	0.0	10.0	33.0320	-115.6540		1	1		0			
12	Northwest Calif-03	Ferndale City Hall				40.2830	-124.8000		99	0	M<6	0			
13	Kern County	LA - Hollywood Stor F	163.1	51.3	287.7	34.9767	-119.0333	16	1	1		1	0	64.7	17.0
14	Kern County	Pasadena - CIT Athen	163.1	51.3	287.7	34.9767	-119.0333	16	1	1		1	0	64.7	17.0
15	Kern County	Santa Barbara Courth	163.1	51.3	287.7	34.9767	-119.0333	16	1	1		1	0	64.7	17.0
16	Kern County	Taft Lincoln School	163.1	51.3	287.7	34.9767	-119.0333	16	1	1		1	0	64.7	17.0

1	Earthquake Name	Station Name	Fault Rupture Area (km^2)	Avg Fault Disp (cm)	Rise Time (s)	Avg Slip Velocity (cm/s)	Static Stress Drop (bars)	Preferred Rupture Velocity (km/s)	Average Vr/Vs	Percent of Moment Release in the Top 5 Km of Crust	Existence of Shallow Asperity: 0=No; 1=Yes	Depth to Top of Shallowest Asperity (km)	Earthquake in Extensional Regime: 1=Yes; 0=No	Fault Name	Slip Rate (mm/Yr)
2	Helena, Montana-01	Carroll College											0		
3	Helena, Montana-02	Helena Fed Bldg											0		
4	Humbolt Bay	Ferndale City Hall											0		
5	Imperial Valley-01	El Centro Array #9											1		
6	Northwest Calif-01	Ferndale City Hall											0		
7	Imperial Valley-02	El Centro Array #9	818.8	101.8			31.0						1	Imperial fault	20.00
8	Northwest Calif-02	Ferndale City Hall											0		
9	Northern Calif-01	Ferndale City Hall											0		
10	Borrego	El Centro Array #9											0		
11	Imperial Valley-03	El Centro Array #9											1		
12	Northwest Calif-03	Ferndale City Hall											0		
13	Kern County	LA - Hollywood Stor F	1098.6	312.8			82.3	2.2	0.7				0	White Wolf fault	2.00
14	Kern County	Pasadena - CIT Athen	1098.6	312.8			82.3	2.2	0.7				0	White Wolf fault	2.00
15	Kern County	Santa Barbara Courth	1098.6	312.8			82.3	2.2	0.7				0	White Wolf fault	2.00
16	Kern County	Taft Lincoln School	1098.6	312.8			82.3	2.2	0.7				0	White Wolf fault	2.00

1	Earthquake Name	Station Name	EpiD (km)	HypD (km)	Joyner-Boore Dist. (km)	Campbell R Dist. (km)	RmsD (km)	ClstD (km)	FW/HW Indicator	Source to Site Azimuth (deg)	X	Theta.D (deg)	SSGA (Strike Slip)	Y	Phi.D (deg)	SSGA (Dip Slip)	s (km)	d (km)	c.tilde. prime	m5	D (km)	Rfn.Hyp
2	Helena, Montana-01	Carroll College	6.31	8.71																		0.33
3	Helena, Montana-02	Helena Fed Bldg	6.31																			
4	Humbolt Bay	Ferndale City Hall	73.49																			
5	Imperial Valley-01	El Centro Array #9	33.20	36.86																		
6	Northwest Calif-01	Ferndale City Hall	54.88																			
7	Imperial Valley-02	El Centro Array #9	12.99	15.69	6.09	7.51	15.80	6.09	na	-90	0.17	36.1	0.14	0.69	31.0		10.49	8.94	1.80	1.00	13.78	0.54
8	Northwest Calif-02	Ferndale City Hall	97.00																			
9	Northern Calif-01	Ferndale City Hall	49.49																			
10	Borrego	El Centro Array #9	57.79																			0.19
11	Imperial Valley-03	El Centro Array #9	28.24																			0.86
12	Northwest Calif-03	Ferndale City Hall	55.96																			
13	Kern County	LA - Hollywood Stor F	118.26	119.29	114.62	117.75	124.34	117.75	na	93	0.11	83.2		0.34	82.6		6.84	5.89	0.93	1.00	9.02	0.78
14	Kern County	Pasadena - CIT Athen	125.81	126.77	122.65	125.59	128.72	125.59	na	90	0.08	87.6		0.34	82.1		5.24	5.89	0.91	1.00	7.88	0.86
15	Kern County	Santa Barbara Courth	88.39	89.76	81.30	82.19	106.65	82.19	na	171	0.11	6.4		0.57	48.2		6.84	9.93	1.61	1.00	12.05	0.37
16	Kern County	Taft Lincoln School	43.49	46.21	38.42	38.89	56.44	38.89	na	-109	0.11	63.2		0.66	55.7		6.84	11.45	1.43	1.00	13.34	0.16

1	Earthquake Name	Station Name	Rfp.Hyp	Rfn.Clst	Rfp.Clst	Rfn.Imd	Rfp.Imd	GMX's C1	GMX's C2	GMX's C3	Campbell's GEOCODE	Bray and Rodriguez-Marek SGS	Depth	Preferred NEHRP Based on Vs30	Preferred Vs30 (m/s)	Alternate Vs30 for CWB stations (m/sec)	Measured /Inferred Class	Sigma of Vs30 (in natural log Units)
2	Helena, Montana-01	Carroll College	0.15					E	Z	A	C	B		C	659.6		2	0.4
3	Helena, Montana-02	Helena Fed Bldg						C	Z	A	D	B		C	659.6		2	0.4
4	Humbolt Bay	Ferndale City Hall						B	Q	D	A	D	S	D	219.3		0	0.1
5	Imperial Valley-01	El Centro Array #9						E	Q	D	A	D	D	D	213.4		0	0.1
6	Northwest Calif-01	Ferndale City Hall						B	Q	D	A	D	S	D	219.3		0	0.1
7	Imperial Valley-02	El Centro Array #9	0.21	0.00	0.98	0.25	0.40	E	Q	D	A	D	D	D	213.4		0	0.1
8	Northwest Calif-02	Ferndale City Hall						B	Q	D	A	D	S	D	219.3		0	0.1
9	Northern Calif-01	Ferndale City Hall						B	Q	D	A	D	S	D	219.3		0	0.1
10	Borrego	El Centro Array #9	0.89					E	Q	D	A	D	D	D	213.4		0	0.1
11	Imperial Valley-03	El Centro Array #9	0.21					E	Q	D	A	D	D	D	213.4		0	0.1
12	Northwest Calif-03	Ferndale City Hall						B	Q	D	A	D	S	D	219.3		0	0.1
13	Kern County	LA - Hollywood Stor F	0.52	0.85	0.52	0.82	0.53	I	H	D	A	D	S	D	316.5		0	0.1
14	Kern County	Pasadena - CIT Athen	0.46	0.87	0.48	0.87	0.47	C	Q	D	B	D	D	C	415.1		0	0.2
15	Kern County	Santa Barbara Courth	0.02	0.45	0.04	0.45	0.04	C	Q	D	B	D	S	C	515.0		0	0.2
16	Kern County	Taft Lincoln School	0.65	0.43	0.70	0.29	0.68	F	Q	D	A	D	S	C	385.4		0	0.2

1	Earthquake Name	Station Name	NEHRP Classification from CGS's Site Condition Map	Geological Unit	Geology	Owner	Station Latitude	Station Longitude	STORIES	INSTLOC	Depth to Basement Rock	Site Visited	NGA Type	Age
2	Helena, Montana-01	Carroll College				USGS	46.5800	-112.0300						
3	Helena, Montana-02	Helena Fed Bldg				USGS	46.5900	-112.0400	4	BASEMENT	0.00			
4	Humbolt Bay	Ferndale City Hall	D	Qal, thin		USGS	40.5760	-124.2630	2	GROUND LEVEL				Pleistocene
5	Imperial Valley-01	El Centro Array #9	D	Qal, deep, Imperial V.		USGS	32.7940	-115.5490	2	BASEMENT	5.00			Holocene
6	Northwest Calif-01	Ferndale City Hall	D	Qal, thin		USGS	40.5760	-124.2630	2	GROUND LEVEL				Pleistocene
7	Imperial Valley-02	El Centro Array #9	D	Qal, deep, Imperial V.		USGS	32.7940	-115.5490	2	BASEMENT	5.00			Holocene
8	Northwest Calif-02	Ferndale City Hall	D	Qal, thin		USGS	40.5760	-124.2630	2	GROUND LEVEL				Pleistocene
9	Northern Calif-01	Ferndale City Hall	D	Qal, thin		USGS	40.5760	-124.2630	2	GROUND LEVEL				Pleistocene
10	Borrego	El Centro Array #9	D	Qal, deep, Imperial V.		USGS	32.7940	-115.5490	2	BASEMENT	5.00			Holocene
11	Imperial Valley-03	El Centro Array #9	D	Qal, deep, Imperial V.		USGS	32.7940	-115.5490	2	BASEMENT	5.00			Holocene
12	Northwest Calif-03	Ferndale City Hall	D	Qal, thin		USGS	40.5760	-124.2630	2	GROUND LEVEL				Pleistocene
13	Kern County	LA - Hollywood Stor F	D	Qal, thin, west LA	Holocene fine	CDMG	34.0900	-118.3390	0	GROUND LEVEL	2.50			Holocene
14	Kern County	Pasadena - CIT Athen	CD	Qoa		CIT	34.1390	-118.1210	2	BASEMENT	0.23			Holocene
15	Kern County	Santa Barbara Courth	CD	Qoa		USGS	34.4230	-119.7000	2	BASEMENT	8.50			Holocene
16	Kern County	Taft Lincoln School	D	Qal, thin		USGS	35.1500	-119.4600	1	BASEMENT	6.40			

1	Earthquake Name	Station Name	Grain Size	Depositional History	Z1 (m)	Z1.5 (m)	Z2.5 (m)	Depth to Franciscan Rock (km)	Basin	h (m)	hnorm (m)	Rsbe (m)	Rcebe (m)	Rebe (m)	Rsbe1 (m)	File Name (Horizontal 1)
2	Helena, Montana-01	Carroll College														HELENA/A-HMC180.at2
3	Helena, Montana-02	Helena Fed Bldg														HELENA/B-FEB000.at2
4	Humbolt Bay	Ferndale City Hall		Alluvial Valley	547	1327	2606									HUMBOLT/FRN225.at2
5	Imperial Valley-01	El Centro Array #9	Aggregate	Lacustrine												IMPVALL/B-ELC000.at2
6	Northwest Calif-01	Ferndale City Hall		Alluvial Valley	547	1327	2606									NWCALIF/A-FRN045.at2
7	Imperial Valley-02	El Centro Array #9	Aggregate	Lacustrine												IMPVALL/I-ELC180.at2
8	Northwest Calif-02	Ferndale City Hall		Alluvial Valley	547	1327	2606									NWCALIF/C-FRN045.at2
9	Northern Calif-01	Ferndale City Hall		Alluvial Valley	547	1327	2606									NCALIF/F-FRN225.at2
10	Borrego	El Centro Array #9	Aggregate	Lacustrine												BORREGO/B-ELC000.at2
11	Imperial Valley-03	El Centro Array #9	Aggregate	Lacustrine												IMPVALL/C-ELC000.at2
12	Northwest Calif-03	Ferndale City Hall		Alluvial Valley	547	1327	2606									NWCALIF/B-FRN224.at2
13	Kern County	LA - Hollywood Stor F	Coarse	Alluvial Valley	320	600	2169									KERN/PEL090.at2
14	Kern County	Pasadena - CIT Athen	Coarse		320	309	1676									KERN/PAS180.at2
15	Kern County	Santa Barbara Courth	Aggregate													KERN/SBA042.at2
16	Kern County	Taft Lincoln School														KERN/TAF021.at2

1	Earthquake Name	Station Name	File Name (Horizontal 2)	File Name (Vertical)	Type of Recording	Instrument Model	PEA Processing Flag	Type of Filter	npass	nroll	HP-H1 (Hz)	HP-H2 (Hz)	LP-H1 (Hz)	LP-H2 (Hz
2	Helena, Montana-01	Carroll College	HELENA/A-HMC270.at2	HELENA\A-HMCDWN	A		PEA	C	1	2.5	0.200	0.200	15.000	15.000
3	Helena, Montana-02	Helena Fed Bldg	HELENA/B-FEB090.at2	HELENA\B-FEB-UP	A		PEA	C	1	2.5	1.000	0.500	20.000	20.000
4	Humbolt Bay	Ferndale City Hall	HUMBOLT/FRN315.at2	HUMBOLT\FRN-UP	A		PEA	C	1	2.5	0.500	0.300	10.000	10.000
5	Imperial Valley-01	El Centro Array #9	IMPVALL/B-ELC090.at2	IMPVALL\B-ELC-UP	A		PEA	C	1	2.5	1.000	0.600	12.000	12.000
6	Northwest Calif-01	Ferndale City Hall	NWCALIF/A-FRN135.at2	NWCALIF\A-FRNDWN	A		PEA	C	1	2.5	0.500	0.200	11.000	11.000
7	Imperial Valley-02	El Centro Array #9	IMPVALL/I-ELC270.at2	IMPVALL\I-ELC-UP	A		PEA	C	1	2.5	0.200	0.200	15.000	15.000
8	Northwest Calif-02	Ferndale City Hall	NWCALIF/C-FRN135.at2	NWCALIF\C-FRN-UP	A		PEA	C	1	2.5	0.500	0.500	13.000	10.000
9	Northern Calif-01	Ferndale City Hall	NCALIF/F-FRN315.at2	NCALIF\F-FRN-UP	A		PEA	C	1	2.5	0.200	0.500	13.000	13.000
10	Borrego	El Centro Array #9	BORREGO/B-ELC090.at2	BORREGO\B-ELC-UP	A		PEA	C	1	2.5	0.100	0.100	15.000	15.000
11	Imperial Valley-03	El Centro Array #9	IMPVALL/C-ELC090.at2	IMPVALL\C-ELC-UP	A		PEA	C	1	2.5	0.400	0.150	13.000	12.000
12	Northwest Calif-03	Ferndale City Hall	NWCALIF/B-FRN314.at2	NWCALIF\B-FRN-UP	A		PEA	C	1	2.5	0.500	0.500	12.000	12.000
13	Kern County	LA - Hollywood Stor F	KERN/PEL180.at2	KERN\PEL-UP	A		PEA	C	1	2.5	0.200	0.200	15.000	13.000
14	Kern County	Pasadena - CIT Athen	KERN/PAS270.at2	KERN\PAS-UP	A		PEA	C	1	2.5	0.500	0.200	12.500	12.100
15	Kern County	Santa Barbara Courth	KERN/SBA132.at2	KERN\SBA-UP	A		PEA	C	1	2.5	0.500	0.500	14.300	13.200
16	Kern County	Taft Lincoln School	KERN/TAF111.at2	KERN\TAF-UP	A		PEA	C	1	2.5	0.050	0.050	13.900	13.200

APPENDIX C

1	Earthquake Magnitude	Mechanism Based on Rake Angle	Hypocenter Depth (km)	Preferred Vs30 (m/s)	EpiD (km)	HypD (km)	Joyner-Boore Dist. (km)	Campbell R Dist. (km)	RmsD (km)	ClstD (km)	D (km)	PGA (g)	PGV (cm/sec)	PGD (cm)
2	6.00	0	6	659.6	6.31	8.71						0.1674	10.36	1.70
3	6.00			659.6	6.31							0.0424	0.56	0.03
4	5.80			219.3	73.49							0.0423	3.16	0.33
5	5.00		16	213.4	33.20	36.86						0.0158	0.61	0.05
6	5.50			219.3	54.88							0.1083	6.08	0.68
7	6.95	0	8.8	213.4	12.99	15.69	6.09	7.51	15.80	6.09	13.78	0.2584	31.74	18.01
8	6.60			219.3	97.00							0.0491	3.40	0.70
9	6.40			219.3	49.49							0.1147	5.28	1.39
10	6.50	0		213.4	57.79							0.0544	3.93	1.41
11	5.60	0		213.4	28.24							0.0294	2.91	0.76
12	5.80			219.3	55.96							0.1066	5.30	0.62
13	7.36	2	16	316.5	118.26	119.29	114.62	117.75	124.34	117.75	9.02	0.0537	7.59	3.57
14	7.36	2	16	415.1	125.81	126.77	122.65	125.59	128.72	125.59	7.88	0.0497	7.28	1.89

1	Earthquake Magnitude	Mechanism Based on Rake Angle	Hypocenter Depth (km)	Preferred Vs30 (m/s)	EpiD (km)	HypD (km)	Joyner-Boore Dist. (km)	Campbell R Dist. (km)	RmsD (km)	ClstD (km)	D (km)	PGA (g)	PGV (cm/sec)	PGD (cm)
159	6.53	0	9.96	274.5	2.47	10.26	0.00	3.41	7.09	0.34	10.26	0.3438	30.39	7.62
160	6.53	0	9.96	274.5	2.62	10.30	0.00	3.40	7.82	0.65	10.28	0.2903	33.85	9.47
161	6.53	0	9.96	223.0	6.20	11.73	0.47	4.01	9.98	2.68	11.42	0.6861	53.86	12.69
162	6.53	0	9.96	208.7	43.15	44.29	8.54	10.57	23.49	10.42	39.05	0.1933	39.86	16.02
163	6.53	0	9.96	231.2	17.65	20.27	10.45	11.56	17.41	10.45	16.28	0.2329	18.52	9.20
164	6.53	0	9.96	205.8	57.14	58.00	23.17	24.60	40.49	24.60	38.74	0.1033	13.28	9.37
165	6.53	0	9.96	659.6	24.82	26.74	15.19	16.06	30.69	15.19	15.60	0.1760	14.04	5.84
166	6.53	0	9.96	274.5	18.88	21.35	7.29	8.35	22.77	7.29	15.60	0.2703	29.27	10.64
167	6.53	0	9.96	345.4	83.94	84.53	49.10	50.10	68.84	50.10	38.47	0.1220	12.64	2.33
168	6.53	0	9.96	274.5	22.43	24.55	13.52	15.34	28.95	15.30	14.16	0.1597	11.51	2.71
169	6.53	0	9.96	274.5	12.92	16.31	0.99	3.54	12.99	1.11	15.54			
170	6.53	0	9.96	274.5	33.73	35.17	22.03	22.54	40.51	22.03	15.60	0.2849	29.75	16.67
171	6.53	0	9.96	192.1	29.07	30.72	7.31	8.61	15.58	7.31	29.41	0.2165	50.98	28.04

1	Earthquake Magnitude	Mechanism Based on Rake Angle	Hypocenter Depth (km)	Preferred Vs30 (m/s)	EpiD (km)	HypD (km)	Joyner-Boore Dist. (km)	Campbell R Dist. (km)	RmsD (km)	ClstD (km)	D (km)	PGA (g)	PGV (cm/sec)	PGD (cm)
2237	5.90	2	8	614.1	56.98	57.53	45.17	48.07	54.95	48.07	15.56	0.0189	1.89	0.38
2238	5.90	2	8	272.6	47.86	48.53	39.38	42.67	46.46	42.67	15.56	0.0196	1.65	0.30
2239	5.90	2	8	473.9	52.79	53.39	44.29	47.24	51.21	47.24	15.56	0.0227	1.87	0.47
2240	5.90	2	8	473.9	48.22	48.88	41.08	44.25	47.06	44.25	11.31	0.0435	1.61	0.20
2241	5.90	2	8	395.6	55.07	55.65	47.85	50.59	53.64	50.59	11.75	0.0305	2.44	0.37
2242	5.90	2	8	379.2	53.95	54.54	45.95	48.80	52.97	48.80	13.60	0.0299	2.18	0.22
2243	5.90	2	8	473.9	49.17	49.82	41.74	44.86	47.88	44.86	12.53	0.0205	1.08	0.13

1	Earthquake Magnitude	Mechanism Based on Rake Angle	Hypocenter Depth (km)	Preferred Vs30 (m/s)	EpiD (km)	HypD (km)	Joyner-Boore Dist. (km)	Campbell R Dist. (km)	RmsD (km)	ClstD (km)	D (km)	PGA (g)	PGV (cm/sec)	PGD (cm)
2300	5.90	2	8	553.4	66.72	67.20	55.70	56.07	65.80	56.07	11.18	0.0461	2.55	0.30
2301	5.90	2	8	473.9	94.65	94.99	85.61	87.00	95.93	87.00	9.77	0.0048	0.34	0.05
2302	5.90	2	8	215.0	148.42	148.64	140.03	140.67	149.91	140.67	8.16	0.0032	0.57	0.17
2303	5.90	2	8	373.3	125.27	125.52	116.08	117.24	126.41	117.24	13.44	0.0129	1.43	0.23
2304	5.90	2	8	553.4	90.28	90.63	80.57	82.23	91.21	82.23	13.60	0.0076	0.41	0.05
2305	5.90	2	8	473.9	79.69	80.09	70.53	72.17	80.97	72.17	9.45	0.0118	0.56	0.05
2306	5.90	2	8	473.9	122.57	122.83	112.88	114.07	123.42	114.07	13.60	0.0067	0.33	0.03
2307	5.90	2	8	531.9	141.83	142.06	132.00	133.02	142.56	133.02	13.60	0.0055	0.27	0.02
2308	5.90	2	8	260.8	136.73	136.96	128.43	129.00	138.26	129.00	8.00	0.0080	0.94	0.16
2309	5.90	2	8	212.4	135.11	135.35	124.18	124.35	134.72	124.35	11.18	0.0144	2.60	0.57
2310	5.90	2	8	215.0	139.34	139.57	128.36	128.52	138.83	128.52	11.18	0.0082	1.16	0.28
2311	5.90	2	8	215.0	136.09	136.32	125.09	125.25	135.55	125.25	11.18	0.0094	1.25	0.31
2312	5.90	2	8	215.0	136.81	137.04	125.79	125.96	136.23	125.96	11.18	0.0097	1.70	0.38
2313	5.90	2	8	226.4	133.77	134.00	122.79	122.96	133.28	122.96	11.18	0.0101	1.29	0.34
2314	5.90	2	8	215.0	134.82	135.06	123.80	123.97	134.23	123.97	11.18	0.0114	1.46	0.38

1	Earthquake Magnitude	Mechanism Based on Rake Angle	Hypocenter Depth (km)	Preferred Vs30 (m/s)	EpiD (km)	HypD (km)	Joyner-Boore Dist. (km)	Campbell R Dist. (km)	RmsD (km)	ClstD (km)	D (km)	PGA (g)	PGV (cm/sec)	PGD (cm)
3443	6.30	2	16	473.9	128.31	129.31	114.03	115.46	129.14	115.46	14.02	0.0130	1.67	0.56
3444	6.30	2	16	473.9	126.61	127.62	112.56	113.63	127.37	113.63	14.88	0.0097	2.04	0.78
3445	6.30	2	16	272.6	134.32	135.27	119.59	121.25	135.27	121.25	16.13	0.0119	1.18	0.42
3446	6.30	2	16	473.9	81.09	82.65	64.17	64.95	81.30	64.95	18.45	0.0318	2.70	0.74
3447	6.30	2	16	489.2	83.09	84.62	65.94	66.70	83.23	66.70	18.45	0.0277	3.39	0.84
3448	6.30	2	16	272.6	75.32	77.00	58.78	59.63	75.65	59.63	18.45	0.0370	4.47	0.94
3449	6.30	2	16	272.6	71.85	73.61	54.50	55.41	72.00	55.41	18.45	0.0300	4.52	1.32
3450	6.30	2	16	272.6	77.58	79.22	60.21	61.03	77.68	61.03	18.45	0.0542	4.89	1.07
3451	6.30	2	16	540.7	72.89	74.63	56.06	56.94	73.18	56.94	18.45	0.0417	5.01	1.48
3452	6.30	2	16	362.0	74.37	76.07	56.93	57.80	74.45	57.80	18.45	0.0560	5.28	1.25
3453	6.30	2	16	272.6	78.85	80.46	62.33	63.12	79.17	63.12	18.45	0.0524	4.01	0.90
3454	6.30	2	16	553.4	64.62	66.58	47.42	48.47	64.89	48.47	18.45	0.0266	3.06	0.72
3455	6.30	2	16	465.6	68.43	70.27	52.15	53.10	68.86	53.10	18.45	0.0214	2.02	0.32
3456	6.30	2	16	551.2	54.57	56.87	38.50	39.78	54.80	39.78	18.45	0.0529	4.59	1.39
3457	6.30	2	16	487.3	47.32	49.95	30.44	32.04	47.73	32.04	18.45	0.0396	4.44	1.28