Addis Ababa University
School of Information Science
Master of Science in Information Science

Application of Data Mining for Weather Forecasting

Ephrem Tibebu

October 2015

Addis Ababa University
School of Information Science
Master of Science in Information Science

Application of Data Mining for Weather Forecasting

A Thesis Submitted to the School of Information Science of Addis Ababa University in Partial Fulfillment of the Requirements for the Degree of Master of Science in Information Science

Ephrem Tibebu

October 2015

Addis Ababa University
School of Information Science
Master of Science in Information Science

Application of Data Mining for Weather Forecasting

Ephrem Tibebu

Name and Signature of members of the Examining Board

Name	Title	Signature	Date
Tibebe Beshah (PhD)	Advisor	_____	_____
Martha Yifiru (PhD)	Examiner	_____	_____
Wondwossen Mulugeta (PhD)	Examiner	_____	_____

Declaration

This thesis is my original research, and has not been presented for a degree in any other university.

……

Ephrem Tibebu

October 2015

This thesis has been submitted for examination with our approval as university advisors

……

Tibebe Beshah (PhD)

October 2015

Dedication

I would like to dedicate this thesis to my beloved sister Tewodi who I have always been and will also be thinking about

ACKNOWLEDGEMENT

First and for most, I would like to extend my unshared thanks to the almighty God for his endless blessings.

I would also like to express my deepest sense of gratitude to my adviser Tibebe Beshah (PhD), who offered his continuous advice and encouragement throughout the course of this thesis. I thank him for the strong appreciation for the approach, treatment and the help I have got at the time of difficulties.

I would also like to thank National Meteorological Agency staff particularly Ato Tameru Kebede for providing me the necessary data for the study and for their unreserved help throughout the study time. I am also thankful to all my friends especially, Zinabu Yilma & Million Beyene for their support in conducting this research.

Finally, I must express my very deep gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my entire life. This accomplishment would not have been possible without them.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Weather forecasting has been one of the most scientifically and technologically challenging problems around the world in the last century. Making an accurate prediction is one of the major challenges meteorologist are facing all over the world. Since ancient times, weather prediction has been one of the most important domains. Scientists have tried to forecast meteorological characteristics using a number of methods, some of these methods being more accurate than others.

Accurate and timely weather forecasting is a major challenge for the National Meteorological Agency of Ethiopia. In this study, the researcher investigated the use of data mining techniques in forecasting rainfall. This was carried out using J48 decision tree, Multilayer perceptron artificial neural network, and PART rule induction algorithms and meteorological data collected between 2000 and 2014 from National Meteorological Agency of Ethiopia. A data model for the meteorological data was developed and this was used to train the classifier algorithms. The performances of these algorithms were compared using standard performance metrics, and the algorithm which gave the best results used to generate classification rules for the mean weather variables. A predictive model was also developed for the weather prediction program and the results compared with actual weather data for the predicted periods. The results show that given enough case data, Data Mining techniques can be used for weather forecasting.

To get a better awareness in choosing which model produced sound prediction and higher accuracy, 13 experiments were done with J48 algorithm and multilayer perceptron classifier, and eight experiments were done using PART rule induction, by inputting all the records with a 10 fold cross-validation mode, and inputting different percentage (%) of the record for testing the performance of the model. The next option used by the researcher to improve the performance of the model were to test if a better model could be obtained by excluding one or more of the input variables and training different models. J48 has an accuracy of 86.65%, PART has an accuracy of 84.96 and Neural Network has an accuracy of 80.03%. Then J48 algorithm has shown better prediction performance. In the future, the effective use of information and technology is important for National Meteorology to stay competitive in today's complex environment. The challenges faced when trying to make large, diverse, and often complex dataset records are considerable and employing other classification algorithms could yield better results.

ACRONYMS

NMA	National Meteorological Agency
NWP	Numerical Weather Prediction
WRF	Weather Research and Forecast
ANN	Artificial Neural Network
CRISP-DM	CRoss-Industry Standard Process for Data Mining
KDD	Knowledge Discovery Databases
SPSS	Statistical Package for the Social Sciences
CSV	Comma Separated Values
ARFF	Attribute Relation File Format
KNN	K-Nearest Neighbor
LR	Linear least-squares regression
IB3	Instance Based Learning
SMOTE	Synthetic Minority Oversampling Technique
ROC	Receiver Operating Characteristics
IT	Information Technology
CRM	Customer Relationship Management
CHAID	Chi-squared Automatic Interaction Detection,
CART	Classification and Regression Trees

CHAPTER ONE

1. INTRODUCTION

1.1. BACKGROUND OF THE STUDY

The National Meteorological Agency (NMA) of Ethiopia is mainly concerned with the forecasts and analysis of the atmospheric system affecting the two Ethiopian rainy seasons 'Belg' and 'Kiremt'. To this end, the agency uses a number of bulletins and all available information from a variety of sources abroad such as UK and US, to prepare seasonal forecasts in Ethiopia. The seasonal forecast made by the NMA is mainly about rainfall and temperature. The long range forecast unit of the NMA is responsible for preparing and issuing monthly and seasonal forecasts in Ethiopia. The methods of forecasting which are applied in preparing seasonal forecasts in the NMA are based on traditional forecasting methods like analogue, trend analysis, statistical assessment, etc. with regards to temperature and rainfall forecasting, the NMA uses a more conventional approach. That is a typical day temperature profile is obtained by taking the average of the temperature profile over the preceding thirty days. So far, the NMA does not have a data mining model of any sort that can be used to forecast the weather.

Weather forecasts provide critical information about future weather. Weather forecasts and warning are the most important service provided by the meteorological profession. Forecasts are used by government and industry to protect life and property and to improve the efficiency of operations, and by individuals to plan a wide range of activities. Weather forecasting remains a complex business, due to its chaotic and unpredictable nature. It is the process that is neither wholly science nor wholly art. Weather phenomena, usually of a complex nature, have a direct impact on the safety and/or economic stability of people. Accurate weather forecast models are important for third world countries like Ethiopia, where most of the agriculture depends on weather. It is a major concern to identify any trends for weather parameters to deviate from its periodicity, which would disrupt the economy of the country. This fear has been aggravated due to the threat by the global warming and greenhouse effect. The impact of extreme weather phenomena on society is growing more and more costly, causing infrastructure damage, injury and the loss of life. Therefore, there is need for accurate weather forecasts today more than ever

before not only as a defense against hazardous weather, but also in planning day-to-day operation of private enterprises and governments, and by individuals to enhance their quality of life.

Meteorology is the interdisciplinary scientific study of the atmosphere. It observes the changes in temperature, air pressure, and moisture & wind direction. Usually, Temperature, pressure and wind measurement and humidity are the variables that are measured by thermometer, barometer, anemometer, and hygrometer, respectively. There are many methods of collecting data observation, Doppler radar and Satellites are some of them. Weather predictions are made by collecting quantitative data about the current state of the atmosphere.

Meteorological data mining is a form of data mining concerned with finding hidden patterns inside largely available meteorological data, so that the information retrieved can be transformed into usable knowledge. Weather is one of the meteorological data that is rich by important knowledge [***1***].

Miheala [***2***], said that in order to make knowledge extraction as much as possible, different techniques could be applied. Among these techniques, data mining and, more general, knowledge discovery techniques became the most used in the recent years. The application of data mining approach in knowledge base development involves a set of techniques for searching through data sets, looking for hidden correlations and trends which are inaccessible using conventional data analysis techniques.

Data mining is a subfield of Machine Learning that enables finding interesting knowledge in very large databases. It is the most essential part of the knowledge-discovery process, which combines databases, statistics, artificial intelligence and machine learning techniques. The basic techniques for data mining includes: decision tree, artificial neural network, rule induction, Bayesian Learning, support vector machine, clustering and association rules [***2***].

1.2. STATEMENT OF THE PROBLEM

Weather forecasts and warnings are the most important services provided by the meteorological experts. Meteorology is a science where data mining techniques can be successfully applied to come up with various results in the area. Weather phenomena, usually of a complex nature, have a direct impact on the safety and/or economic stability of people. Accurate weather forecast

models are important for third world countries like Ethiopia. Accurate and timely weather forecasting is a major challenge for the National Meteorological Agency of Ethiopia. Rainfall prediction modeling involves a combination of statistical models, observation and knowledge of trends and patterns. Using these methods, reasonably accurate forecasts can be made up. However, the meteorological agency is lack of accurate and timely weather forecasting. The Agency is used to forecast weather by using observation and satellite image system and it has more than 800 stations in all over the country. The weather information is collected from all the stations with in every three hours. Then the meteorology specialist decodes all the data received from the stations in to the meteogram boards. The meteogram board consists of different attribute such as air temperature, wind speed, wind pressure, dew point temperature, depression; cloud etc. then uses Numerical Weather Prediction to forecast the weather.

Numerical Weather Prediction (NWP) is the science of predicting the weather using mathematical models of the atmosphere and oceans to predict the weather based on current weather conditions. A number of global and regional forecast models are run in different countries worldwide. In National Meteorological Agency (NMA) of Ethiopia currently there are two running Numerical Weather Prediction models, which are the fifth generation of the Mesoscale Modeling System and Weather Research and Forecast (WRF).

As the researcher understood from the discussion conducted with the meteorological experts of National Meteorology Agency of Ethiopia, the methods of forecasting which are applied in preparing forecasts in the agency are based on traditional forecasting methods like analogue, trend analysis, statistical assessment, etc. with regards to temperature and rainfall forecasting there are difficulties in processing to forecast the weather. Then the major challenge of traditional forecasting of rainfall in National Meteorological Agency is not accurate and timely weather forecasting.

As a result, almost all the decision-making processes of the agency are not supported by tools and techniques that could extract patterns from previous weather data records. Based on the preliminary discussion made with the meteorology expert, the current meteorology forecast systems do not have a learning facility.

A number of research works have been done in the area of weather forecasting by applying different techniques of data mining. Tafesse [***3***] tried to develop an accurate Artificial Neural

Network model that is capable of making one day ahead forecast of the eight 3-hourly temperature profile of Addis Ababa. He investigated the application of the feed-forward neural network with back propagation learning algorithm in forecasting temperature. He used 3 years data of temperature from January 2001 to 2003 from National Meteorological Agency. He focused on selecting one of the data mining techniques without comparing other data mining techniques related with their performance of each algorithm. He didn't consider different variables such as wind, rainfall, humidity, Air pressure etc. that influence on the daily temperature forecast.

Therefore, the aim of this study is to develop weather forecasting model by using different data mining techniques that can predict an accurate and timely rainfall forecast. Data mining consists of more than collecting and analyzing data. The tools which are used for analysis can include statistical models, mathematical algorithms and machine learning methods. These methods include algorithms that improve their performance automatically through experience, such as neural networks or decision trees.

In this regards, in an attempt to develop a weather forecasting model, this study explores and finds answers to the following research questions:-

- What are the major determinant variables that contribute weather condition?
- Which data mining models are more appropriate to predict the weather condition specifically rainfall?

1.3. OBJECTIVES OF THE STUDY

1.3.1. GENERAL OBJECTIVES

The general objective of this research work is to come up with a model for weather forecasting using data mining techniques that can predict an accurate and timely rainfall forecast for National Meteorological Agency of Ethiopia.

1.3.2. SPECIFIC OBJECTIVES

The specific objectives of this research work are stated as follows:

- To assess the potential of data mining techniques in assisting meteorological related decisions and review of literature that can support the study in the area of applying data mining technology on weather prediction.
- To collect and preprocess the dataset.
- To experiment and build a prediction model that can support a weather forecasting.
- To test and compare models performance.
- To conclude and report the research result.

1.4. METHODOLOGY

The application of data mining technology in research areas is rapidly increasing in various industries. Meteorology is among those industries which are using data mining technology extensively. Because, issues related to weather condition are increasing the amount of data stored in these sector. Thus, the application of data mining technology as a research area for these sectors is very important. Data mining technology as a tool has its own methods, procedures and techniques to be followed and used in research. These methods, procedures and techniques may be chosen as per the nature of the data and the objectives of the research.

In this research, the CRISP-DM (CRoss-Industry Standard Process for Data Mining) methodology is used. It is the most commonly used methodology for developing data mining research projects. This model describes the activities that must be done to develop a data mining research projects. The main objectives and benefits of CRISP-DM are to ensure quality of knowledge discovery research project results; reducing skills required for knowledge discovery; reducing cost and time; being general purpose i.e., widely stable across varying applications and robust i.e., insensitive to changes in the environment; tool and technique independent and tool supportable; capturing experience for reuse and supporting knowledge transfer and training [***4***]. One important factor of CRISP-DM success is the fact that CRISP-DM is industry-tool and application neutral [***5***].

The process of knowledge discovery in databases is detailed in the CRISP-DM methodology, which is the de-facto industry standard for KDD. According to this methodology the life cycle of a data mining consists of six successive phases, described below:

1. Business/Goal understanding

The starting phase of CRISP-DM focuses on understanding the business/project objectives and requirements from a business perspective, and then converting this knowledge into a data mining problem definition, and a preliminary plan designed to achieve the objectives. In this phase one works closely with meteorology experts to define the problem and determine the research goal, identify key people and learn about current solutions to the problem. In the first case the researcher focused on the best prediction of rainfall occurrence in Hawassa station, where any suitable historical data can be used for it. The researcher identified two possible alternatives with different costs, i.e. no rain and rain, which suddenly appears. Even if our main goal was to get a prediction of the best possible quality, also the interpretation of the rules used for prediction can be interesting the ability to comprehensively describe the processes leading to occurrence of rainfall. The research goals then need to be translated into the data mining goals, and include initial selection of data mining tools. The goal is to improve prediction of selected meteorological condition, leading to cost saving and increased public safety.

2. Data understanding

Data understanding phase started with the selection of data relevant for the specified problems. The researcher has investigated several available data sources as sets of physical quantities measured automatically by meteorological stations in Addis Ababa, Bahirdar and Hawassa. After making the identified data available, the researcher performed an initial data examination in order to verify the quality of the data. These operations were extended with a calculation of the basic statistical for key variables and their correlation.

This phase includes the collection of sample data, and deciding which data will be needed including its format and size. If background knowledge does exist some attributes may be ranked as more important. After that, verification of the usefulness of the data is required in respect to the data mining goal. Data needs to be verified for completeness, redundancy, missing value, etc. The researcher has selected different input samples to test and evaluate suitability of relevant data mining methods for the goals. For the purposes of rainfall

forecasting the researcher collected historical data from National Meteorological Agency of Ethiopia different station i.e. Addis Ababa, Bahirdar and Hawassa stations for 15 years (2000-2014). The quality of available meteorological data was high with low number of missing records in Hawassa station. Therefore, the researcher decided to use the data from Hawassa station. The initial dataset contains the following variables for Hawassa station: YEAR, MONTH, DAY, PERCIPT, TEMMAX, TEMMIN, RELHUM, WINDLY, SUNHRS, LATITUDE, LONGITUDE AND ELEVATION. After identifying the dataset, features related to weather condition is selected with the help of the meteorology expert.

3. Data Preparation

The third phase of CRISP-DM is data pre-processing; this is usually the most complex and also most time consuming phase of the whole data mining process; usually taking 60 to 70 percent of the overall time. The researcher applied necessary pre-processing operations to obtain ready datasets for implementation of selected mining methods. Meteorological data from all sources i.e. data extracted from messages, meteorological stations and physical model predictions were integrated into one relational database. The performed operations were very similar for both cases with little modifications based on characteristics of the data and related data mining tasks.

The researcher has identified the following data preparation tasks: The first step was data extraction from the meteorological messages broadcasted in meteorological meteogram board. This board of the messages was fixed with standard codes denoting the parts of the messages and data values. The output of this task was structured data stored in a relational database with raw extracted data. Each record in the integrated database has assigned valid from/to time interval and 3D coordinates of measured area i.e. ranges for longitude, latitude and altitude. Since each data source had different data precision and/or granularity, the goal of this operation was to interpolate measured values and compute additional data for the requested area and time with the specified data granularity.

The same approach was used for replacement of missing values. The researcher has selected a representative sample for next modeling phase. Reduction was necessary by reason of the technical restrictions inherent to some methods, but it can also lead to simplification of the

task at hand by removing irrelevant attributes and records, thus even increasing the quality of the results. The consultations with the domain experts resulted into specified valid ranges of values and detected invalid data. Out-of-range data were considered as missing values.

4. Modeling

To build a predictive model from the cleaned data, WEKA software was used. At this stage, a data mining model is developed to represent various characteristics of the underlying dataset. A number of iterations in fine-tuning the model were considered. The researcher typically iterated through process to find a best-fitted data mining model for the data by adjusting various parameters of the model. This model was used to describe the hope of discovering novel and useful patterns and trends in data; or used to predict future or unknown values. There are many data mining techniques like Artificial Neural Network, Decision Tree, Naïve Bayes, PART etc., algorithms is used to develop models and check the performance of the algorithm based on the classification accuracy, F-measure, Precision, and Recall. The researcher has tested several of these methods provided in the WEKA data mining environment. Finally the researcher selected J48 decision tree, Artificial Neural Network and PART rule induction models were applied to forecast rainfall in Hawassa station. In order to obtain optimal results, all parameters of these algorithms were tuned by testing several strategies to divide input dataset into training and test set.

This phase is the core of the data mining process, various modeling techniques are selected and applied, the application of a selected data mining method to the available data; the parameters of the method must be modified in order to obtain optimal values. The tuning is usually done by dividing the input data set into a training and validation data set; the validation part is used for assessing the quality of the model. Each method has its specific requirements for the input data; therefore, steeping back to the data preparation phases is often needed.

In order to mine hidden knowledge from the pre-processed dataset and compare the performance of classifiers, WEKA 3.6.10 is used. WEKA is chosen since it is proven to be powerful for data mining and used by many researchers for mining task and the researcher is familiar with the tool. It contains tools for data preprocessing, clustering, regression,

classification, association rules and visualization. So, In this research, Weka 3.6.10 software was used as a mining tool. In addition Microsoft Excel and SPSS software were used for data cleaning and converting the original file to CSV file format.

5. Evaluation

From a data analysis perspectives, at this phase the study have built a model that appears to have a high quality. The results of the KDD process from the point of view of the original criteria to set in the first step (goal understanding); before proceeding to the final deployment of the model it is important to thoroughly evaluate the model whether the process has been successful, or whether to take more steps in order to succeed and to be sure it is properly achieves the business objectives. The evaluation is based on quantitative indicators of the quality of models created and to compare with different data mining techniques. At the end of this stage, a decision on the use of the data mining results should be reached.

In this phase, the researcher tried to evaluate the J48 decision tree, Multilayer perceptron, and PART algorithm model performance. The analysis of the accuracy generated by the confusion matrix is used to compute the accuracy which measures the result of the classification. Another evaluating criteria used for the models performance evaluation is the detailed accuracy measure, which measures the true positive rate, the false positive rate, and the precision, recall and ROC (Receiver Operating Characteristics) of the models developed. In addition to this the major determinant variable that influences the rainfall is ranked by information gain.

6. Deployment

In this phase the researcher develop a practical deployment plan for the model. The researcher also determines the strategy or its monitoring and maintenance in order to reduce the costs of its deployment and the possibility of its incorrect application. One of the results of this phase should also be an overall evaluation of the study and the definition of recommendations for future works of similar character.

1.5. SIGNIFICANCE OF THE STUDY

The developed model can have lots of advantages for the meteorology experts and other related individuals. Some of them are as follows:

- The proposed system provides a weather forecasting model regarding which action to take in accordance to the weather condition for the society.
- To improve the decision making process for experts especially in the short and long range forecast department.
- The output of this study can also be an input for further research in this and other related areas in the context of our country.
- This study can give hands on experience for the researcher for understanding studies in the future.

1.6. SCOPE OF THE STUDY

The study focuses on developing a model with an accurate and timely weather forecasting in National Meteorological Agency of Ethiopia. To get more information, it would be good if the study could include three or more meteorological station and weather variables, but the researcher selects one station. On the other hand, to make the study more manageable and to complete the study with in the given time, the researcher delimited the variables of the weather into rainfall.

1.7. LIMITATION OF THE STUDY

As any research works, this study is not free from limitation, Here; the researcher's intention was to develop a model for an accurate and timely weather forecasting by using data mining techniques. The main limitation of this research work is lack of adequate reference material on weather forecasting by using data mining technologies especially local researches. And another limitation is due to the nature of the available dataset and the time available to complete the study, the researcher has limited the study to predict rainfall.

1.8. ORGANIZATION OF THE PAPER

The structure of this thesis is presented as follows. Chapter one provides details information about background of the study, statements of the problem, objectives of the study, methodology, scope of the study, limitation of the study, and significance of the study. Chapter two is basically dedicated for literature review. In this chapter, a detailed discussion about data mining and its task relevant for this study are included. In addition, related works in the area of weather forecasting by using data mining techniques are also included in this chapter. Chapter three is focus on data preparation and preprocessing. This chapter discussed about the selection and preparation of data process that is undertaken in the research work. This chapter is mainly used for understanding the process in preparing the data for producing quality data using the Weka filter options and the statistical software like SPSS tool. Chapter four presents experimentation and data analysis of the study. Results of the classification experiments were also discussed here. Finally, chapter five provides conclusion of the research, and also presents recommendation for future work.

CHAPTER TWO

2. LITERATURE REVIEW

2.1. OVERVIEW OF DATA MINING

Data mining have been employed successfully to build a very important application in the field of meteorology like predicting abnormal events like hurricanes, storms and river flood prediction [***6***]. These applications can maintain public safety and welfare.

Witten and Eibe [***7***] stated, the amount of data in the world, in our lives, seems to go on and on increasing and there's no end in sight. Omnipresent personal computers make it too easy to save things that previously we would have trashed. Inexpensive multi gigabyte disks make it too easy to postpone decisions about what to do with all this stuff. We simply buy another disk and keep it all. Ubiquitous electronics record our decisions, our choices in the supermarket, our financial habits, our comings and goings. We swipe our way through the world, every swipe a record in a database. The World Wide Web overwhelms us with information; meanwhile, every choice we make is recorded.

In addition, As stated by Deshpande and Thakare [***8***] noted that data can be form simple numerical figures and text documents, to more complex information such as spatial data, multimedia data, and hypertext documents. To take complete advantage of these data; data retrieval is simply not enough, it requires a tool for automatic summarization of data, extraction of the essence of information stored, and the discovery of patterns in raw data. With the enormous amount of data stored in files, databases, and other repositories, it is increasingly important, to develop powerful tool for analysis and interpretation of such data and for the extraction of interesting knowledge that could help in decision-making. The only answer to all above is 'Data Mining' [***8***].

In data mining, the data is stored electronically and the search is automated or at least augmented by computer. Even this is not particularly new. Economists, statisticians, forecasters, and communication engineers have long worked with the idea that patterns in data can be sought automatically, identified, validated, and used for prediction. What is new is the staggering increase in opportunities for finding patterns in data. The uncontrolled growth of databases in

recent years, databases on such everyday activities as customer choices, brings data mining to the lead of new business technologies. As the world grows in complexity, overwhelming us with the data it generates, data mining becomes our only hope for elucidating the patterns that underlie it.

Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner [***9***].

Data mining is defined as the process of discovering patterns in data. The process must be automatic or (more usually) semiautomatic. The patterns discovered must be meaningful in that they lead to some advantage, usually an economic advantage. The data is invariably present in substantial quantities.

According to Berry and Linoff [***10***], data mining usually makes sense when there is large amount of data. For this reason most of the algorithms developed for data mining purpose requires large volume of data so as to build and train models that are responsible for different tasks of data mining such as classification, clustering, prediction, association and the like. Moreover, the need for bulky data can be explained by a couple of reasons. Primarily in the case of small databases, it is feasible to capture appealing trends and relationships by introducing traditional tools such as spreadsheets and database query. The second reason is that most data mining tools and algorithms demand large amount of training data (data used for building a model) in order to generate unbiased models. The rationale is simple and straight forward, small training data results in unreliable generalizations based on chance patterns.

Data mining is an interesting technique that can be implemented in various areas to generate useful information from the existing large volumes of data. Data mining has thus far been successfully implemented to bring success in commercial applications. Some of the applications of data mining include discovery of interesting patterns, clustering of data based on parameters and prediction of results by using the existing data [***11***] [***12***]. There are diverse techniques and algorithms available in data mining that can be implemented for various applications. This paper proposes an efficient data mining technique for weather forecast [***13***].

2.2. DATA MINING AND KNOWLEDGE DISCOVERY PROCESS

The research is mainly concerned with data mining which is extracting useful insights from large and detailed collections of data. With the increased possibilities in modern society for companies and institutions to gather data cheaply and efficiently, this subject has become of increasing importance. This interest has inspired a rapidly maturing research field with developments both on a theoretical, as well as on a practical level with the availability of a range of commercial tools. But the researcher is mainly focused on developing a weather forecasting model by comparing different data mining techniques.

Data Mining is the process of analyzing data from different perspectives and summarizing the results as useful information. It has been defined as "the nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data" [***14***] [***15***].

Mihaela Opera [***2***] agreed that data mining is a subfield of Machine Learning that enables finding interesting knowledge (patterns, models and relationships) in very large databases. It is the most essential part of the knowledge-discovery process, which combines databases, statistics, artificial intelligence and machine learning techniques. The basic techniques for data-mining include: decision-tree induction, rule induction, instance-based learning, artificial neural networks, Bayesian learning, support vector machines, ensemble techniques, clustering, and association rules.

Knowledge Discovery Databases (KDD) is the process of extracting and refining useful knowledge from large databases. It involves three stages: inductive learning, deductive verification and human intuition. Inductive learning focuses on data and tries to generate hypotheses from them. Deductive verification evaluates the evidential support from some previously given hypotheses, while human intuition helps the discovery guiding so that it gathers the information wanted by the user, in a certain time window. Data mining could be applied to any domain where large databases are saved [***2***].

The process of data mining uses machine learning, statistics, and visualization techniques to discover and present knowledge in a form that is easily comprehensible. The word “Knowledge” in KDD refers to the discovery of patterns which are extracted from the processed data. A pattern

is an expression describing facts in a subset of the data. Thus, the difference between KDD and data mining is that:

> "KDD refers to the overall process of discovering knowledge from data while data mining refers to application of algorithms for extracting patterns from data without the additional steps of the KDD process." [***15***]

However, since Data Mining is a crucial and important part of the KDD process, most researchers use both terms interchangeably.

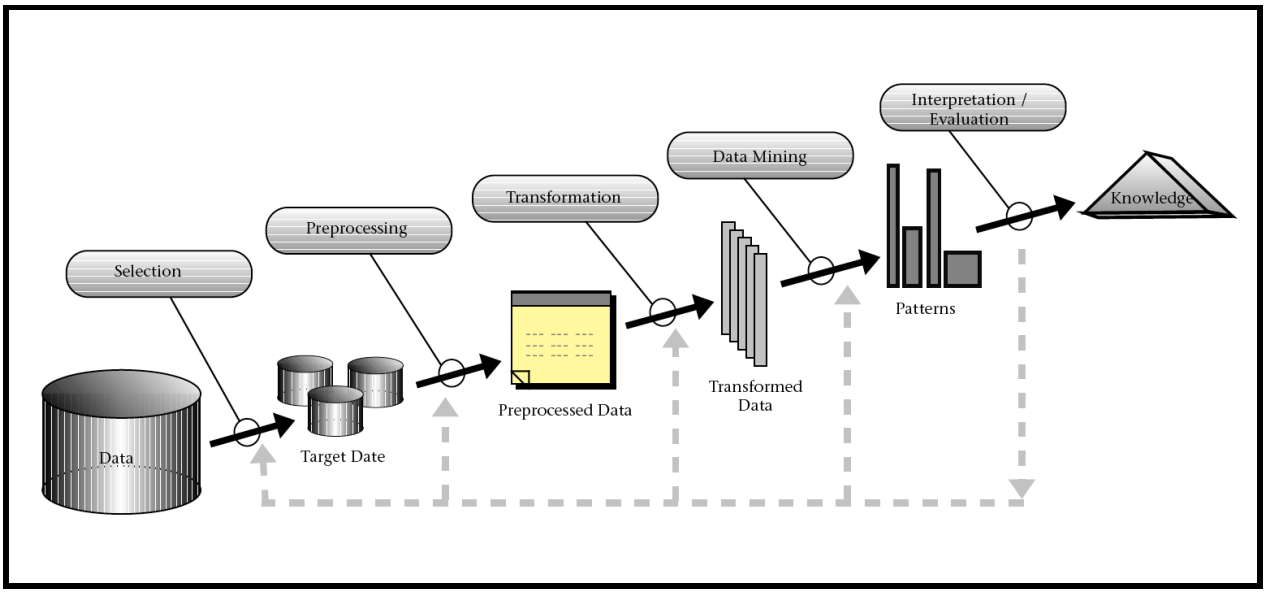


Figure 2.1. Knowledge Discoveries in Database Process

Maimon and Rokach [***16***] stated that Knowledge Discovery in Databases (KDD) is an automatic, exploratory analysis and modeling of large data repositories. KDD is the organized process of identifying valid, novel, useful, and understandable patterns from large and complex data sets. Data Mining (DM) is the core of the KDD process, involving the inferring of algorithms that explore the data, develop the model and discover previously unknown patterns. The model is used for understanding phenomena from the data, analysis and prediction.

According to [***15***], Data mining is the core stage of the knowledge discovery process that is aimed at the extraction of interesting, nontrivial, implicit, previously unknown and potentially useful information from data in large databases. Data mining projects were initially carried out in many different ways with each data analyst finding their own way of approaching the problem,

often through trial-and-error. As the data mining techniques and businesses evolved, there was a need for data analysts to better understand and standardize the knowledge discovery process, which would as a side effect demonstrate to prospective customers that data mining was sufficiently mature to be adopted as a key element of their business. This led to the development of the cross-industry standard process for data mining [***17***] [***4***], which is intended to be independent of the choice of data mining tools, industry segment, and the application/problem to be solved [***18***].

In KDD process, there are different kinds of standard methodologies or models, such as Cross-Industry Standard Process for Data Mining (CRISP-DM), Anand et al. [***19***], Fayyad et al. [***15***], Cios et al. [***20***], and Cabena et al. [***21***]. All process models consist of multiple steps executed in a sequence, which often includes loops and iterations. As Lukasz & Musilek [***22***] explained the main differences between the models lie in the number and scope of their specific steps. The following table shows the different steps of the KDD process models.

Table 2.1 Side-by-side comparison of the major existing KDDM Model

Model	Fayyad et al [15]	Cabena et al [21]	Annand & Buchner [19]	CRISP-DM	Cios et al. [20]
No. of steps	9	5	8	6	6
Refs	Fayyad et al, 1996	Cabena et al., 1998	Anand & Buchner, 1998	Chapman et al., 2000	Cios et al., 2000
Steps	1. Developing and understanding of the application domain	1. Business Objectives determination	1. Human Resource Identification	1. Business Understanding	1.Understanding the Problem Domain
			2. Problem Specification		
	2. Creating a target data set	2. Data Preparation	3. Data prospecting	2. Data Understanding	2. Understanding the data
			4. Domain knowledge elicitation		
	3. Data cleaning and preprocessing		5. Methodology Identification	3. Data preprocessing	3. Preparation of the data
	4. Data reduction and projection		6. Data Preprocessing		
	5. Choosing the data mining tasks				
	6. Choosing the data mining algorithm				
	7. Data Mining	3. Data Mining	7. Pattern Discovery	4. Modeling	4. Data Mining
	8. Interpreting mined pattern	4. Domain knowledge Elicitation	8. Knowledge Post-Processing	5. Evaluation	5. Evaluation of the Discovered Knowledge
	9. Consolidated Discovered Knowledge	5. Assimilation of Knowledge		6. Deployment	6. Using the discovered knowledge

2.2.1. CRISP-DM

The CRISP-DM (CRoss-Industry Standard Process for Data Mining) was first established in the late 1990s by four companies: Integral Solutions Ltd. (a provider of commercial data mining solutions), NCR (a database provider), DaimlerChrysler (an automobile manufacturer), and OHRA (an insurance company). The last two companies served as data and case study sources.

The development of this process model enjoys strong industrial support. It has also been supported by the ESPRIT program funded by the European Commission. The CRISP-DM Special Interest Group was created with the goal of supporting the developed process model. Currently, it includes over 300 users and tool and service providers.

The model is characterized by an easy-to-understand vocabulary and good documentation. It divides all steps into sub-steps that provide all necessary details. It also acknowledges the strong iterative nature of the process, with loops between several of the steps [***20***]. In general, it is a very successful and extensively applied model, mainly due to its grounding in practical, industrial, real-world knowledge discovery experience.

Frawley et al. [***14***], stated that data mining and knowledge discovery in databases appeared as a recognizable research discipline in the early 1990s, with the advent of a series of data mining workshops. The birth of this area was triggered by a need in the database industry to deliver solutions enhancing the traditional database management systems and technologies. At that time, these systems were able to solve the basic data management issues like how to deal with the data in transactional processing systems.

The knowledge discovery process is iterative and interactive, consisting of nine steps. The process has many "artistic" aspects in the sense that one cannot present one formula or make a complete taxonomy for the right choices for each step and application type. Thus it is required to deeply understand the process and the different needs and possibilities in each step. Taxonomy for the Data Mining methods is helping in this process.

2.3. MACHINE LEARNING

The field of machine learning seeks to answer the following questions:

"How can we build computer system that automatically improves with experience, and what are the fundamental laws that govern all the learning processes."

Mitchell [***23***] stated, machine learning is a natural outgrowth of the intersection of computer science and statistics. We might say the defining question of computer science is "How can we build machine that solves the problems, and which problems are tractable/ intractable?" The

question that largely defines statistics is "what can be inferred from data plus a set of modeling assumptions, with what reliability? The defining question for Machine Learning builds on both, but it is a distinct question.

Machine Learning focuses on the question of how to get computers to program themselves. Whereas Statistics has focused primarily on what conclusions can be inferred from data, Machine Learning incorporates additional questions about what computational architectures and algorithms can be used to most effectively capture, store, index, retrieve and merge these data, how multiple learning subtasks can be orchestrated in a larger system, and questions of computational tractability.

Carbonell et al. [***24***] defined as machine learning is the science of computer modeling of learning processes. It enables a computer to acquire knowledge from existing data or theories using certain inference strategies such as induction or deduction. Over the years, research in machine learning has been pursued with varying degrees of intensity using different approaches and placing emphases on different aspects and goals.

In machine learning, the term database typically refers to a collection of instances or examples maintained in a single file. Instances are usually fixed length feature vectors. Information is sometimes also provided about the feature names and value ranges, as in a data dictionary [***14***]. A learning algorithm takes the data set and its accompanying information as input and returns a statement representing the result of the learning as output.

2.4. DATA MINING TASKS

Data Mining deals with what kind of patterns can be mined. On the basis of kind of data to be mined there are two kinds of functions involved in Data Mining. These are Predictive modeling and Descriptive modeling. The main tasks of data mining are to specify the data mining task in form of data mining query, this query is an input to the system; data mining query is defined in terms of data mining task primitives. Using these primitives allow us to communicate in interactive manner with the data mining system. Here is the list of Data Mining Task Primitives: Set of task relevant data to be mined, kind of knowledge to be mined, background knowledge to

be used in discovery process, Interestingness measures and thresholds for pattern evaluation, Representation for visualizing the discovered patterns.

Further divide the data mining task of generating models into the following two approaches:

- Supervised or directed data mining modeling.
- Unsupervised or undirected data mining modeling.

The goal in supervised or directed data mining is the variables under investigation can split into two groups: explanatory variables and one or more dependent variables. The target of the analysis is to specify a relationship between the explanatory variables and the dependent variables as it is done in regression analysis.to apply directed data mining techniques the value of the dependent variables must be known for a sufficiently large part of the data set.

In unsupervised or undirected data mining however variable is singled out as the target. The goals of predictive and descriptive data mining are achieved by using specific data mining techniques that fall within certain primary data mining tasks. The goal is rather to establish some relationship among all the variables in the data. The user asks the computer to identify patterns in the data that may be significant. Undirected modeling is used to explain those patters and relationships one they have been found.

In addition, Han and Kamber [***25***] mentioned that data mining functionalities are used to specify the kinds of patterns to be found in data mining tasks. In general, such tasks can be classified into two categories: descriptive and predictive. Descriptive mining tasks characterize properties of the data in a target data set. Predictive mining tasks perform induction on the current data in order to make predictions. Data mining functionalities are described below:

2.4.1. PREDICTIVE MODELING

Predictive modeling is a commonly used statistical technique to predict future behavior. Predictive modeling solutions are a form of data-mining technology that works by analyzing historical and current data and generating a model to help predict future outcomes. In predictive modeling, data is collected, a statistical model is formulated, predictions are made, and the model is validated (or revised) as additional data becomes available [***26***]. Predictive models often

perform calculations during live transactions, for example, to evaluate the risk or opportunity of a given customer or transaction to guide a decision. If health insurers could accurately predict secular trends (for example, utilization), premiums would be set appropriately, profit targets would be met with more consistency, and health insurers would be more competitive in the marketplace.

As stated by Tan and Kumar [***2 7***] indicated many of the data mining applications are aimed to predict the future state of the data. Classification is a technique of mapping the target data to the predefine groups or classes. It is a supervised learning because the classes are predefined before the examination of the target data. The regression involves the learning of function that maps data item to real valued prediction variable.

Dickey [***28***] defined predictive modeling is a name given to a collection of mathematical techniques having in common the goal of finding a mathematical relationship between a target, response, or "dependent" variable and various predictor or "independent" variables with the goal in mind of measuring future values of those predictors and inserting them into the mathematical relationship to predict future values of the target variable.

In predictive modeling, data is collected for the relevant predictors, a statistical model is formulated, predictions are made and the model is validated (or revised) as additional data becomes available. The model may employ a simple linear equation or a complex neural network, mapped out by sophisticated software.

Predictive modeling is used widely in information technology (IT). In spam filtering systems, for example, predictive modeling is sometimes used to identify the probability that a given message is spam. Other applications of predictive modeling include customer relationship management (CRM), capacity planning, change management, disaster recovery, security management, engineering, meteorology and city planning.

2.3.1.1. Classification

Classification is the process of finding a model that describes and distinguishes data classes or concepts. The models are derived based on the analysis of a set of training data (i.e., data objects for which the class labels are known). The model is used to predict the class label of objects for which the class label is unknown [***29***].

Classification involves the discovery of a predictive learning function that classifies a data item into one of several predefines classes. It involves examining the features of a newly presented object and assigning to it a predefined class. Define classification has a two-step process. First a model is built describing a predetermined set of data classes or concepts and secondly, the model is used for classification.

Classification maps or classifies a data item into one of several predefined classes. It is a predictive modeling, in which we give a pre-defined grouping and try to predict the group of a new data point. A set of classification rules are generated from the classification model, based on the features of the data in the training set, which can be used to classify future data and develop a better understanding of each class in the database.

Thair Nu Phyu [***30***] also define classification is a data mining technique used to predict group membership for data instances. Several major kinds of classification method including decision tree induction, Bayesian networks, k-nearest neighbor classifier, case-based reasoning, genetic algorithm and fuzzy logic techniques.

Classification is to segregate items into several predefined classes. Given a collection of training samples, this type of task can be designed to find a model for class attributes as a function of the values of other attributes [***31***]. Data mining creates classification models by examining already classified data (cases) and inductively finding a predictive pattern. These existing cases may come from an historical database, such as people who have already undergone a particular medical treatment or moved to a new long distance service. They may come from an experiment in which a sample of the entire database is tested in the real world and the results used to create a classifier. For example, a sample of a mailing list would be sent an offer, and the results of the mailing used to develop a classification model to be applied to the entire database. Sometimes

an expert classifies a sample of the database, and this classification is then used to create the model which will be applied to the entire database [***32***].

Regression uses existing values to forecast what other values will be. In the simplest case, regression uses standard statistical techniques such as linear regression. Unfortunately, many real world problems are not simply linear projections of previous values [***32***]. Therefore, more complex techniques (e.g., logistic regression, decision trees, or neural nets) may be necessary to forecast future values.

Decision tree

The decision tree classifier uses a hierarchical or layered approach to classification. Each vertex in the tree represents a single test or decision. The outgoing edges of a vertex correspond to all possible outcomes of the test at that vertex. These outcomes partition the set of data into several subsets, which are identified by every leaf in the tree. A leaf of the tree specifies the expected value of the categorical attribute for the records described by the path from the root to that leaf. Learned trees can also be represented as sets of if then-else rules [***33***].

Mitchell [***33***], stated that decision tree methods are robust to errors, including both errors in classifying the training examples and errors in the attribute values that describe these examples. Decision tree can be used when the data contain missing attribute values.

In addition to Mitchell, Witten and Eibe [***7***] also defined decision tree. A "divide-and-conquer" approach to the problem of learning from a set of independent instances leads naturally to a style of representation called a decision tree. Nodes in a decision tree involve testing a particular attribute. Usually, the test at a node compares an attribute value with a constant. However, some trees compare two attributes with each other, or use some function of one or more attributes. Leaf nodes give a classification that applies to all instances that reach the leaf or a set of classifications, or a probability distribution over all possible classifications. To classify an unknown instance, it is routed down the tree according to the values of the attributes tested in successive nodes, and when a leaf is reached the instance is classified according to the class assigned to the leaf [***7***].

According to Han and Kamber [***25***], decision tree induction is the learning of decision trees from class-labeled training tuples. A decision tree is a flowchart-like tree structure, where each internal node (non-leaf node) denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node holds a class label. The topmost node in a tree is the root node.

Data Mining uses machine-learning methods using decision trees to classify objects based on the dependent variable. There are two main types of decision trees [***32***]. Decision trees, which are used to predict categorical variables, are called classification trees because they place instances in categories or classes. Decision trees used to predict continuous variables are called regression trees. Classification trees label records and assign them to the proper class.

Classification trees can also provide the confidence that the classification is correct. In this case, the classification tree reports the class probability, which is the confidence that a record is in a given class. Regression trees, on the other hand, estimate the value of a target variable that takes on numeric values. When a tree model is applied to data, each record flows through the tree along a path determined by a series of tests until the record reaches a leaf or terminal node of the tree. There it is given a class label based on the class of the records that reached that node in the training set or, in the case of regression trees, assigned a value based on the mean of the values that reached that leaf node in the training set.

Decision tree models are commonly used in data mining to examine the data and induce the tree and its rules that will be used to make prediction [***3 2***] . Various decision tree algorithms such as CHAID (Chi-squared Automatic Interaction Detection), C4.5/5.0, CART (Classification and Regression Trees), J48 and any with less familiar acronyms, produce trees that differ from one another in the number of splits allowed at each level of the tree, how those splits are chosen when the tree is built, and how the tree growth is limited to prevent over-fitting [***10***].

Today's data mining software tools allow the user to choose among several splitting criteria and pruning rules, and to control parameters such as minimum node size and maximum tree depth allowing one to approximate any of these algorithms.

Nearest Neighbor

Nearest Neighbor methods are very straightforward: to classify a new object, with input vector y, we simply examine the k closest training data set points to y and assign the object to the class that has the majority of points among this k. closest is defined here in terms of the p - dimensional input space. Thus we are seeking those objects in the training data that are most similar to the new object, in terms of the input variables, and then classifying the new object into the most heavily represented class among these most similar objects.

In theoretical terms, we are taking a small volume of the space of variables, centered at x, and with radius the distance to the k^{th} nearest neighbor. Then the maximum likelihood estimators of the probability that a point in this small volume belongs to each class are given by the proportion of training points in this volume that belong to each class. The k nearest neighbor method assigns a new point to the class that has the largest estimated probability. Nearest neighbor methods are essentially in the class of what we have termed "regression" methods — they directly estimate the posterior probabilities of class membership [***9***].

"Closeness" is defined in terms of a distance metric, such as Euclidean distance. The Euclidean distance between two points or tuples, say, $X_1 = (X_{11}, X_{12}, \dots, X_{1n})$ and $X_2 = (X_{21}, X_{22}, \dots, X_{2n})$, is

$$dist(\boldsymbol{X}_1, \boldsymbol{X}_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2}.$$

In other words, for each numeric attribute, we take the difference between the corresponding values of that attribute in tuple X_1 and in tuple X_2, square this difference, and accumulate it. The square root is taken of the total accumulated distance count. Typically, we normalize the values of each attribute before using the above equation [***25***].

Naïve Bayes

Bayesian classification is based on Bayes' theorem. Studies comparing classification algorithms have found a simple Bayesian classifier known as the Naïve Bayesian classifier to

be comparable in performance with decision tree and selected neural network classifiers. Bayesian classifiers have also exhibited high accuracy and speed when applied to large databases. Naive Bayesian classifiers assume that the effect of an attribute value on a given class is independent of the values of the other attributes [***25***]. This assumption is called class conditional independence. It is made to simplify the computations involved and, in this sense, is considered “naïve”.

Naïve Bayes is a term in Bayesian statistics dealing with a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions. A more descriptive term for the underlying probability model would be "independent feature model". In simple terms, a Naive Bayes classifier assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature [***34***].

Depending on the precise nature of the probability model, Naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without believing in Bayesian probability or using any Bayesian methods.

In spite of their Naive design and apparently oversimplified assumptions, Naive Bayes classifiers often work much better in many complex real-world situations than one might expect. Recently, careful analysis of the Bayesian classification problem has shown that there are some theoretical reasons for the apparently unreasonable efficacy of Naive Bayes classifiers [***7***]. Bhargavi and Jyothi [***34***], stated the advantage of the Naive Bayes classifier is that it requires a small amount of training data to estimate the parameters (means and variances of the variables) necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined and not the entire covariance matrix.

The Naïve Bayes algorithm is based on conditional probabilities. It uses Bayes' Theorem, a formula that calculates a probability by counting the frequency of values and combinations of values in the historical data. Bayes' Theorem finds the probability of an event occurring

given the probability of another event that has already occurred. If B represents the dependent event and A represents the prior event, Bayes' theorem can be stated as follows.

$$Prob\left(\frac{B}{A}\right) = \left[(prob\left(\frac{A}{B}\right) * prob(B))/prob(A)\right]$$

In probability theory Bayes theorem shows how one conditional probability (such as the probability of a hypothesis given observed evidence) depends on its inverse (in this case, the probability of that evidence given the hypothesis). In more technical terms, the theorem expresses the posterior probability (i.e. after evidence B is observed) of a hypothesis A in terms of the prior probabilities of A and B, and the probability of B given A. It implies that evidence has a stronger confirming effect if it was more unlikely before being observed.

One of the major statistical methods in data mining is Bayesian inference. The naive Bayesian classifier provides a simple and effective approach to classifier learning. It assumes that all class conditional probability densities are completely specified. Even though this assumption is often violated in real world data sets, a naïve Bayesian classifier is employed [***35***].

Artificial Neural Network

An artificial neural network (ANN) is a mathematical model or computational model based on biological neural networks, in other words, is an emulation of biological neural system. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation [***36***]. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase.

According to Hajek [***37***], a neural network is a massively parallel distributed processor that has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects: Knowledge is acquired by the network through a learning process and Interneuron connection strengths known as synaptic weights are used to store the knowledge.

Neural networks represent a brain metaphor for information processing. These models are biologically inspired rather than an exact replica of how the brain actually functions. Neural networks have been shown to be very promising systems in many forecasting applications and business classification applications due to their ability to "learn" from the data, their nonparametric nature (i.e., no rigid assumptions), and their ability to generalize. Neural computing refers to a pattern recognition methodology for machine learning. The resulting model from neural computing is often called an artificial neural network (ANN) or a neural network. Neural networks have been used in many business applications for pattern recognition, forecasting, prediction, and classification. Neural network computing is a key component of any data mining tool kit [***38***].

According to Gaur [***38***], the neural network model can be broadly divided into the following three types:

- Feed-forward networks: It regards the perception back-propagation model and the function network as representatives, and mainly used in the areas such as prediction and pattern recognition;
- Feedback network: It regards Hopfield discrete model and continuous model as representatives, and mainly used for associative memory and optimization calculation;
- Self-organization networks: it regards adaptive resonance theory (ART) model and Kohonen model as representatives, and mainly used for cluster analysis.

According to Kumar [***39***], an Artificial Neural Network is defined as a data processing system consisting of a large number of simple highly interconnected processing elements (artificial neurons) in an architecture inspired by the structure of the cerebral cortex of the brain. There are several types of architecture of ANNs. However, the two most widely used ANNs are discussed below:

a) Feed forward networks: Information flows in one direction along connecting pathways, from the input layer via the hidden layers to the final output layer. There is no feedback (loops) i.e., the output of any layer does not affect that same or preceding layer.
b) Recurrent Networks: These networks differ from feed forward network architectures in the sense that there is at least one feedback loop. Thus, in these networks, for example,

there could exist one layer with feedback connections as shown in figure below. There could also be neurons with self-feedback links, i.e. the output of a neuron is fed back into itself as input.

Neural networks are of particular interest because they offer a means of efficiently modeling large and complex problems in which there may be hundreds of predictor variables that have many interactions.(Actual biological neural networks are incomparably more complex.) Neural nets may be used in classification problems (where the output is a categorical variable) or for regressions (where the output variable is continuous). A neural network starts with an input layer, where each node corresponds to a predictor variable. These input nodes are connected to a number of nodes in a hidden layer. Each input node is connected to every node in the hidden layer. The nodes in the hidden layer may be connected to nodes in another hidden layer, or to an output layer. The output layer consists of one or more response variables [***32***].

2.4.2. DESCRIPTIVE MODELING

Descriptive modeling is to discover patterns in the data and to understand the relationships between attributes represented by the data.

The purpose of data mining is sometimes simply to describe what is going on in a complicated database in a way that increased our understanding of the people, products or processes that produced the data in the first place. They state that a good enough description of behavior will often suggest an explanation for it as well.

The descriptive model identifies the patterns or relationships in data and explores the properties of the data examined [***8***]. It describe all the data, it includes models for overall probability distribution of the data, partitioning of the p-dimensional space into groups and models describing the relationships between the variables. Clustering, association rule discovery, sequence discovery etc. are some of the examples. Clustering is similar to classification except that the groups are not predefined, but are defined by the data alone [***25***]. The association rule finds the association between different attributes. Association rule mining is a two-step process: finding all frequent item sets, generating strong association rules from the frequent item sets. Sequence discovery is a process of finding the sequence patterns in the data. This sequence can be used to understand the trend.

2.4.2.1. Clustering

Clustering is the task of segmenting a diverse group into a number of similar subgroups or clusters. Clusters of objects are formed so that objects within a cluster have high similarity in comparison to one another, but are very dissimilar to objects in other clusters. Clustering is commonly used to search for unique groupings within a data set. The distinguishing factor between clustering and classification is that in clustering there are no predefined classes and no examples. The objects are grouped together based on self-similarity.

According to Han and Kamber [***25***], Cluster analysis or simply clustering is the process of partitioning a set of data objects into subsets. Each subset is a cluster, such that objects in a cluster are similar to one another, yet dissimilar to objects in other clusters. The set of clusters resulting from a cluster analysis can be referred to as a clustering. In this context, different clustering methods may generate different clustering on the same data set. The partitioning is not performed by humans, but by the clustering algorithm. Hence, clustering is useful in that it can lead to the discovery of previously unknown groups within the data.

Cluster analysis has been widely used in many applications such as business intelligence, image pattern recognition, Web search, biology, and security. In business intelligence, clustering can be used to organize a large number of customers into groups, where customers within a group share strong similar characteristics. This facilitates the development of business strategies for enhanced customer relationship management [***25***].

Moreover, consider a consultant company with a large number of projects. To improve project management, clustering can be applied to partition projects into categories based on similarity so that project auditing and diagnosis (to improve project delivery and outcomes) can be conducted effectively [***25***].

Pavel [***40***] also defined Clustering is a division of data into groups of similar objects. Each group, called cluster, consists of objects that are similar between themselves and dissimilar to objects of other groups. Representing data by fewer clusters necessarily loses certain fine details (akin to lossy data compression), but achieves simplification. It represents many data objects by few clusters, and hence, it models data by its

clusters. In addition to this Jain and Dubes [***35***] also defined clustering is to identify a set of categories, or clusters, that describe the data.

According to Berry and Linoff [***10***], Clustering is the task of segmenting a heterogeneous population into a number of more homogeneous subgroups or clusters. What distinguishes clustering from classification is that clustering does not rely on predefined classes. In classification, each record is assigned a predefined class on the basis of a model developed through training on pre-classified examples.

The grouping step can be performed in a number of ways. "Hierarchical clustering algorithms produce a nest series of partitions based on a criterion for merging or splitting clusters based on similarity. Partitioned clustering algorithms identify the partition that optimizes a clustering criterion" [***41***]. Two general categories of clustering methods are partitioning method and hierarchical method. Traditionally clustering techniques are broadly divided into hierarchical and partitioning. Hierarchical clustering is further subdivided into agglomerative and divisive. Agglomerative clustering is start with the points as individual clusters and, at each step, merge the most similar or closest pair of clusters. This requires a definition of cluster similarity or distance. Divisive clustering is start with one, all-inclusive cluster and, at each step, split a cluster until only singleton clusters of individual points remain, at each step, which cluster to split and how to perform the split. There are two types of clustering methods partitioning and hierarchical methods.

Partitioning Methods

The simplest and most fundamental version of cluster analysis is partitioning, which organizes the objects of a set into several exclusive groups or clusters. To keep the problem specification concise, we can assume that the number of clusters is given as background knowledge. This parameter is the starting point for partitioning methods. Formally, given a data set, D, of n objects, and k, the number of clusters to form, a partitioning algorithm organizes the objects into k partitions (k <= n), where each partition represents a cluster. The clusters are formed to optimize an objective partitioning criterion, such as a dissimilarity function based on distance, so that the objects within a cluster are "similar" to

one another and "dissimilar" to objects in other clusters in terms of the data set attributes [***25***].

K-Means

The k-means algorithm defines the centroid of a cluster as the mean value of the points within the cluster. It proceeds as follows. First, it randomly selects k of the objects in D, each of which initially represents a cluster mean or center. For each of the remaining objects, an object is assigned to the cluster to which it is the most similar, based on the Euclidean distance between the object and the cluster mean. The k-means algorithm then iteratively improves the within-cluster variation.

For each cluster, it computes the new mean using the objects assigned to the cluster in the previous iteration. All the objects are then reassigned using the updated means as the new cluster centers. The iterations continue until the assignment is stable, that is, the clusters formed in the current round are the same as those formed in the previous round.

Hierarchical Methods

While partitioning methods meet the basic clustering requirement of organizing a set of objects into a number of exclusive groups, in some situations we may want to partition our data into groups at different levels such as in a hierarchy. A hierarchical clustering method works by grouping data objects into a hierarchy or "tree" of clusters. Representing data objects in the form of a hierarchy is useful for data summarization and visualization. Hierarchical clustering methods can encounter difficulties regarding the selection of merge or split points. Such a decision is critical, because once a group of objects is merged or split, the process at the next step will operate on the newly generated clusters. It will neither undo what was done previously, nor perform object swapping between clusters. Thus, merge or split decisions, if not well chosen, may lead to low-quality clusters. Moreover, the methods do not scale well because each decision of merge or split needs to examine and evaluate many objects or clusters.

Agglomerative Hierarchical Clustering

An agglomerative hierarchical clustering method uses a bottom-up strategy. It typically starts by letting each object form its own cluster and iteratively merges clusters into larger and larger clusters, until all the objects are in a single cluster or certain termination conditions are satisfied. The single cluster becomes the hierarchy's root [***25***]. According to Jain, Agglomerative hierarchical classification places each object in its own cluster and gradually merges these atomic clusters into larger and larger cluster until all objects are in a single cluster.

Divisive Hierarchical Clustering

A divisive hierarchical clustering method employs a top-down strategy. It starts by placing all objects in one cluster, which is the hierarchy's root. It then divides the root cluster into several smaller sub clusters, and recursively partitions those clusters into smaller ones. The partitioning process continues until each cluster at the lowest level is coherent enough—either containing only one object or the objects within a cluster are sufficiently similar to each other. Divisive hierarchical classification reverses the process by starting with all objects in one cluster and subdividing into smaller pieces.

2.5. DATA MINING, ARTIFICIAL INTELLEGENCE AND STATISTICS

Data mining takes advantage of advances in the fields of artificial intelligence (AI) and statistics. Both disciplines have been working on problems of pattern recognition and classification. Both communities have made great contributions to the understanding and application of neural nets and decision trees.

Data mining does not replace traditional statistical techniques. Rather, it is an extension of statistical methods that is in part the result of a major change in the statistics community. The development of most statistical techniques was, until recently, based on elegant theory and analytical methods that worked quite well on the modest amounts of data being analyzed. The increased power of computers and their lower cost, coupled with the need to analyze enormous data sets with millions of rows, have allowed the development of new techniques based on a brute-force exploration of possible solutions.

New techniques include relatively recent algorithms like neural nets and decision trees, and new approaches to older algorithms such as discriminant analysis. By virtue of bringing to bear the increased computer power on the huge volumes of available data, these techniques can approximate almost any functional form or interaction on their own. Traditional statistical techniques rely on the modeler to specify the functional form and interactions.

The key point is that data mining is the application of these and other AI and statistical techniques to common business problems in a fashion that makes these techniques available to the skilled knowledge worker as well as the trained statistics professional. Data mining is a tool for increasing the productivity of people trying to build predictive models [***32***].

2.6. WEATHER FORECASTING

Ancient weather forecasting methods usually relied on observed patterns of events, also termed pattern recognition. For example, it might be observed that if the sunset was particularly red, the following day often brought fair weather. This experience accumulated over the generations to produce weather lore. However, not all of these predictions prove reliable, and many of them have since been found not to stand up to rigorous statistical testing.

Robert Fitz Roy was appointed in 1854 as chief of a new department within the Board of Trade to deal with the collection of weather data at sea as a service to mariners. This was the forerunner of the modern Meteorological Office. All ship captains were tasked with collating data on the weather and computing it, with the use of tested instruments that were loaned for this purpose. A terrible storm in 1859 that caused the loss of the Royal Charter inspired FitzRoy to develop charts to allow predictions to be made, which he called "forecasting the weather", thus coining the term "weather forecast".

It was not until the 20th century that advances in the understanding of atmospheric physics led to the foundation of modern numerical weather prediction. In 1922, English scientist Lewis Fry Richardson published "Weather Prediction by Numerical Process", after finding notes and derivations he worked on as an ambulance driver in World War I. He described therein how small terms in the prognostic fluid dynamics equations governing atmospheric flow could be

neglected, and a finite differencing scheme in time and space could be devised, to allow numerical prediction solutions to be found.

The basic idea of numerical weather prediction is to sample the state of the fluid at a given time and use the equations of fluid dynamics and thermodynamics to estimate the state of the fluid at some time in the future. The main inputs from country-based weather services are surface observations from automated weather stations at ground level over land and from weather buoys at sea. The World Meteorological Organization acts to standardize the instrumentation, observing practices and timing of these observations worldwide. Stations either report hourly in METAR reports, or every six hours in SYNOP reports. Sites launch radiosondes, which rise through the depth of the troposphere and well into the stratosphere. Data from weather satellites are used in areas where traditional data sources are not available. Compared with similar data from radiosondes, the satellite data has the advantage of global coverage, however at a lower accuracy and resolution. Meteorological radar provides information on precipitation location and intensity, which can be used to estimate precipitation accumulations over time. Additionally, if pulse Doppler weather radar is used then wind speed and direction can be determined.

Models are initialized using this observed data. The irregularly spaced observations are processed by data assimilation and objective analysis methods, which perform quality control and obtain values at locations usable by the model's mathematical algorithms (usually an evenly spaced grid). The data are then used in the model as the starting point for a forecast. Commonly, the set of equations used to predict the known as the physics and dynamics of the atmosphere are called primitive equations. These equations are initialized from the analysis data and rates of change are determined. The rates of change predict the state of the atmosphere a short time into the future. The equations are then applied to this new atmospheric state to find new rates of change, and these new rates of change predict the atmosphere at a yet further time into the future. This time stepping procedure is continually repeated until the solution reaches the desired forecast time. The length of the time step is related to the distance between the points on the computational grid. [***42***]

Weather forecasting plays a significant role in meteorology. Weather forecasting remains a formidable challenge because of its data intensive and frenzied nature. Generally two methods are used to forecast weather:

a) The empirical approach and
b) The dynamical approach.

The first approach is based on the occurrence of analogues and it is often referred to as analogue forecasting. This approach is useful in predicting local scale weather if recorded cases are plentiful. The second case is based upon equations and forward simulations of the atmosphere and is often referred to as computer modeling. The dynamical approach is useful to predict large scale weather phenomena and may not predict short term weather efficiently. Most weather prediction systems use a combination of both the techniques [***43***].

2.7. RELATED WORKS

Many scholars have tried to use data mining techniques in areas related to weather forecasting. But, the researcher found one local research related to weather forecasting by using data mining techniques. Tafesse [***3***] tried to develop an accurate Artificial Neural Network model that is capable of making one day ahead forecast of the temperature profile of Addis Ababa. He investigated the application of the feed-forward neural network with back propagation learning algorithm in forecasting temperature. He investigated the application of the feed-forward neural network with back propagation learning algorithm in forecasting temperature. He used 3 years data of temperature from January 2001 to 2003 from National Meteorological Agency. Predict daily average, maximum and minimum temperature for Patras city in Greek by using six different data mining methods: Feed Forward Back Propagation (BP), k-Nearest Neighbor (KNN), linear least-squares regression (LR), Decision tree and instance based learning (IB3). They use four years period data [2002-2005] of temperature, relative humidity and rainfall. The results they obtained in this study were accurate in terms of Correlation Coefficient and Root Mean Square. [***44***]

Olayia et al. [***45***] has presented a data mining techniques in forecasting maximum temperature, rainfall, evaporation and wind speeds using Artificial Neural network & decision tree. They used

a meteor logical data collected between 2000 and 2009 from the city Ibadan. The performance of the algorithm compared using standard performance metrics and the result shows that data mining techniques can be used for weather forecasting & climate changes and ANN has a better performance.

Kalyanker & Alaspurkar [***46***] described the overall process of data mining for weather and to study on weather data using data mining techniques of Clustering K-means partitioning method. By using these techniques, they acquire a weather data and can find the hidden patterns inside the large data set from Gaza City.

Dutta & Tahbilder [***47***] has described the data mining techniques in forecasting monthly rainfall of Assam. This was carried out using traditional statistical techniques Multiple Linear Regression. They used a six year meteorological data from regional meteorological center of Assam and the performance of this model measured in adjusted R-squared. The model considers maximum Temperature, minimum Temperature, wind speed, mean sea level as predictors. They found 63% accuracy in variation of rainfall for their proposed model. The result shows that the prediction model based on multiple linear regressions indicates acceptable accuracy.

Tasha R. Inniss [***48***] used a mathematical programming and statistical techniques and methodologies to develop a seasonal clustering technique for determining clusters of time series data, and applied this techniques to weather and aviation data to determine probabilistic distributions of arrival capacity scenarios, which can be used for efficient traffic flow management. The seasonal clustering technique is modeled as a set partitioning integer programming problem and resulting clustering's are evaluated using the mean square ratio criterion. The resulting seasonal distributions, which have satisfied the mean square ratio criterion, can be used for the required inputs into stochastic ground holding models. In combination, the result would give the optimal number of flight to ground in a ground delay program to aid more efficient traffic flow management.

Kotsiantis et al. [***44***], investigated the efficiency of data mining techniques in estimating minimum, maximum and mean temperature values. Using temperature data from the city of Patras in Greece, a Regression algorithm is applied for the number of results. The performance of these algorithms has been evaluated using standard statistical indicators, such as Correlation

Coefficient, Root Mean Squared Error, etc and also proposed a hybrid data mining techniques that can be used to predict more accurately the mean daily temperature values. It was found that the regression algorithms could enable experts to predict temperature values with satisfying accuracy using as input the temperatures of the previous years. The hybrid data mining techniques produce the most accurate results.

Godfrey et al., [***13***] presented the data mining an activity that was employed in weather data prediction or forecasting. The approach employed is the enhanced Group Method of Data Handling [e-GMDH). The weather data used for the research include daily temperature, daily pressure and monthly rainfall. The results of e-GMDH were compared with those of PNN and its variant, e-PNN. The showed that end users of data mining should endeavor to follow the methodologies of data mining since suspicious data points or outliers in a vast amount of data could give unrealistic results which may affect knowledge inference. Empirical results also show that there are various advantages and disadvantages for the different techniques considered. There is little reason to expect that one can find a uniformly best learning algorithm for optimization of the performance for different weather datasets.

S.Nkrintra et al. [***49***] described the development of a statistical forecasting methods for SMR over Thailand using multiple regression and local polynomial-based nonparametric approaches. SST, sea level pressure (SLP), wind speed, EiNino Southern Oscillation Index (ENSO), IOD was chosen as predictors. The experiment indicated that the correlation between observed and forecast rainfall was 0.6.

T.Sohn et al. [***50***] has developed a prediction model for the occurrence of heavy rain in South Korea using Multiple linear and logistics regression, decision tree and artificial neuron network. They used 45 synoptic factors generated by the numerical models potential predictors.

Win Thida Zaw [***51***] has developed a prediction model for determining rainfall over Myanmar using multiple linear regressions where 15 predictors have been used. As a result of several experiments, the predicted rainfall amount is close to actual values.

Table 2.2 Related Works

Author(Year)	Objectives	Methods	Key Finding
Dutta & Tahbilder (2014)	Data mining technique in forecasting monthly Rainfall of Assam.	Multiple Linear Regression	The result shows that the prediction model based on Multiple linear regressions indicates acceptable accuracy
Olayia et al. (2012)	Forecasting maximum temperature, rainfall, evaporation and wind speeds	Artificial Neural network & decision tree	The result shows that data mining techniques can be used for weather forecasting & climate changes
Win Thida Zaw (2008)	Prediction model for determining rainfall over Myanmar	Multi variables polynomial regression (MPR)	Experiments indicate that the prediction model based on MPR has higher accuracy than using MLR (Multiple Linear Regression)
Kotsiantis et al. (2007)	Predict daily average, maximum and minimum temperature for Patras city in Greek	Feed Forward Back Propagation (BP), k-Nearest Neighbor (KNN), linear least-squares regression (LR), Decision tree and instance based learning (IB3)	The results they obtained in this study were accurate in terms of Correlation Coefficient and Root Mean Square
Godfrey et al. (2007)	Data mining activity that was employed in weather data prediction or forecasting	self-organizing data mining approach employed is the enhanced Group Method of Data Handling (e-GMDH)	Experimental results indicate that the proposed approach is useful for data mining technique for forecasting weather data

S.Nkrintra et al. (2005)	The development of a statistical forecasting method for summer monsoon rainfall over Thailand	A traditional linear regression (parametric) and an adapted nonparametric method based on local polynomials	The nonparametric method seems to show improved skill in the extreme years, especially in wet years
T.Sohn, et al. (2005)	The development of a statistical model for forecasting heavy rain in South Korea. The results of four models are compared via Heidke skill scores	Linear regression model, Logistic regression model, neural network model and decision tree model	As a result, the logistic regression model is recommended
Tafesse Yirdaw (2004)	Develop an accurate ANN model that predicts the eight 3-hourly temperature of one day ahead forecast in Addis Ababa	Artificial Neural network	Neural network with back propagation learning algorithm in forecasting temperature. The model yields good result with independent cases providing about 90% correct forecast of the observed values

CHAPTER THREE

3. DATA PREPARATION AND PREPROCESSING

3.1. OVERVIEW OF DATA PREPARATION AND PREPROCESSING

Today's real world databases are highly susceptible to noisy, missing, and inconsistent data due to their typically huge size (often several gigabytes or more) and their likely origin from multiple, heterogeneous sources. Low-quality data will lead to low-quality mining results. Data pre-processing task could be critical and a very complicated task. Sometimes, the data pre-processing takes more than half of the total time spent on the solving of the data mining problem, because incomplete, noisy, and inconsistent data are commonplace properties of large real-world databases and data warehouses [***29***] [***27***] [***52***].

There are a number of data preprocessing techniques. Data cleaning can be applied to remove noise and correct inconsistencies in the data. Data integration merges data from multiple sources into a coherent data store, such as a data warehouse. Data transformations, such as normalization, may be applied. For example, normalization may improve the accuracy and efficiency of mining algorithms involving distance measurements. Data reduction can reduce the data size by aggregating, eliminating redundant features, or clustering, for instance. These techniques are not mutually exclusive; they may work together. For example, data cleaning can involve transformations to correct wrong data, such as by transforming all entries for a date field to a common format. Data processing techniques, when applied before mining, can substantially improve the overall quality of the patterns mined and/or the time required for the actual mining.

Data preprocessing techniques can improve the quality of the data, thereby helping to improve the accuracy and efficiency of the subsequent mining process. Data preprocessing is an important step in the knowledge discovery process, because quality decisions must be based on quality data. Detecting data anomalies, rectifying them early, and reducing the data to be analyzed can lead to huge payoffs for decision making [***25***].

This chapter deals with the overviews of the data source, data cleaning and data transformation of the data employed in this study. In general the researcher has followed the steps of data mining process mentioned in the first chapter of the methodology section. It was

developed based on the CRISP-DM model by adapting it to the academic purpose. As it has been stated in chapter one the main objective of this study is to build a predictive model and develop rules for weather forecasting of rain fall in Hawassa station. The initial dataset which is from 2000 to 2014 contains the following attributes YEAR, MONTH, DAY, PERCIPT, TEMMAX, TEMMIN, RELHUM, WINDLY, SUNHRS, LATITUDE, LONGITUDE, ELEVATION.

3.2. DATA SELECTION AND PREPARATION

3.2.1. DATA COLLECTION

The data used for this work was collected from Ethiopian National Meteorology Agency. The data collected from the agency 15 year's meteorology data from 2000 G.C to 2014 G.C. for three stations Addis Ababa, Hawassa and Bahirdar. The major steps involved in data preprocessing, namely, data cleaning, data integration, data reduction, and data transformation. Data integration combines data from multiple sources into a coherent store. The data selection process is data relevant to the analysis was selected and retrieved from the dataset. The meteorological dataset had twelve attributes, their type and description is presented in Table 3.1. Hawassa

Table 3.1 Description of the attributes of weather datasets

Attributes	Types	Description
YEAR	NUMERIC	Year
MONTH	NUMERIC	Month
DAY	NUMERIC	Day
PERCEPT	NUMERIC	Rainfall in mm
TMPMAX	NUMERIC	Maximum Temprature
TMPMIN	NUMERIC	Minimum Temprature
SUNHRS	NUMERIC	Shunshine
WINDLY	NUMERIC	Wind speed
RELHUM	NUMERIC	Relative Humidity
LATITUDE	NUMERIC	Latitude
LONGITUDE	NUMERIC	Longitude
ELEVATION	NUMERIC	Elevation

3.2.2. DATA PREPARATION

Data preparation is about constructing a dataset from one or more data sources to be used for exploration and modeling. It is a solid practice to start with an initial dataset to get familiar with the data, to discover first insights into the data and have a good understanding of any possible data quality issues. Data preparation is often a time consuming process and heavily prone to errors. Analyzing data that has not been carefully screened for such problems can produce highly misleading results. Then, the success of data mining projects heavily depends on the quality of the prepared data.

In the data preparation stage, which data will be used as input for data mining is decided. It may involve sampling of data, data cleaning like checking completeness of data records, removing or correcting for noise, etc. the cleaned data can be, further processed by feature selection and extraction algorithms, and by derivation of new attributes. The result would be new data records, meeting specific input requirements for the planned to be used data mining tools.
While data mining is a key stage in the knowledge discovery process, the data preprocessing process often require considerable effort. The purpose of the preprocessing stage is to cleanse the data as much as possible and to put it into a form that is suitable for use in later stages. Starting from the data extracted from the source database maintained by the Agency, a number of transformations are performed before a working dataset was built.

The data preparation phase covers all activities to construct the final data set from the initial raw data. The first step was data extraction from the meteorological messages broadcasted in meteorological meteogram board. This board of the messages was fixed with standard codes denoting the parts of the messages and data values. The output of this task was structured data stored in a relational database with raw extracted data. Each record in the integrated database has assigned valid from/to time interval and 3D coordinates of measured area i.e. ranges for longitude, latitude and altitude. Tasks like data cleaning, record and attribute selection as well as transformation of data using discretization method were included.

3.2.2.1. ATTRIBUTE SELECTION

Many factors affect the success of data mining algorithms on a given tasks. The quality of the data is one such factors- if information is irrelevant or redundant, or the data is noisy and unreliable, then knowledge discovery during training is more difficult. Attribute selection is the process of identifying and removing as much of the irrelevant and redundant information as possible. Learning algorithms differ in the amount of emphasis they place on attribute selection [***53***].

Deciding on the data that will be used for the analysis is based on several criteria, including its relevance to the data mining goals, as well as quality and technical constraints such as limits on data volume or data types [***54***]. Therefore, in this research the attribute are selected with the help of domain expert and extensive literature review. Because taking all the variables in the data base we have, feed them to the data mining tool and find those which are the best predictors may be does not work very well. One reason is that the time it takes to build a model increases with the number of variables. Another reason is that carelessly including extraneous columns can lead to incorrect models. Thus, it is necessary to ignore those attributes that are not important for analysis with the help of domain experts in order to simplify the task of modeling. As described in table 3.1, the following attributes are selected from the meteorological database: Year, Month, Day, Percept, TmpMax, TmpMin, SunHrs, Windly, and RelHum. Because taking all the variables like Longitude, Latitude and Elevation in the data base and used by the data mining tool and find those which are the best predictors may be does not work properly.

3.2.2.2. DATA CLEANING

Without clean data, the results of a data mining analysis are in question. Thus at this stage, the data analyst must either select clean subsets of data or incorporate more ambitious techniques such as estimating missing data through modeling analyses. At this point, data analysts should make sure they outline how they addressed each quality problem reported in the earlier “Verify Data Quality” step [***54***]. Data cleaning routines work to “clean” the data by filling in missing values, smoothing noisy data, identifying or removing outliers, and resolving inconsistencies. If

users believe the data are dirty, they are unlikely to trust the results of any data mining that has been applied. Furthermore, dirty data can cause confusion for the mining procedure, resulting in unreliable output. In the data cleaning stage, a consistent format for the data model was developed which take care of missing data, finding duplicated data, and weeding out of bad data. Finally, the cleaned data were transformed into a suitable format for data mining [***25***].

The researcher used MS-Excel and SPSS Software application for cleaning the data. In this section, different data cleaning tasks were carried out.

Meteorological data by nature has complicated and lots of missing values. Therefore, from the dataset collected for this research work there were missing values in the independent variables "Wind Speed", "Relative Humidity" and "Sunshine". Having efficient methods to fill up missing values extends the applicability in terms of accuracy for many DM methods. The accuracy of the tool is increased and with a larger training set better rules and decision trees can be developed which contributes towards better classification of the data to predict the pattern of rainfall. To handle those missing values, the researcher uses SPSS software to generate or fill the missing value by calculating the attribute mean value of the all the record since the values are numerical.

3.2.2.3. DATA TRANSFORMATION

In data transformation, the data are transformed or consolidated into forms appropriate for data mining. Strategies for data transformation include the following: [***25***]

- Smoothing: This works to remove noise from the data. Techniques include binning, regression, and clustering.
- Attribute construction (or feature construction), where new attributes are constructed and added from the given set of attributes to help the mining process.
- Aggregation, where summary or aggregation operations are applied to the data. For example, the daily sales data may be aggregated so as to compute monthly and annual total amounts. This step is typically used in constructing a data cube for data analysis at multiple abstraction levels.

- Normalization, where the attribute data are scaled so as to fall within a smaller range, such as -1.0 to 1.0, or 0.0 to 1.0.
- Discretization, where the raw values of a numeric attribute (e.g., age) are replaced by interval labels (e.g., 0–10, 11–20, etc.) or conceptual labels (e.g., youth, adult, senior).
- Concept hierarchy generation for nominal data, where attributes such as street can be generalized to higher-level concepts, like city or country. Many hierarchies for nominal attributes are implicit within the database schema and can be automatically defined at the schema definition level.

The data is needed to be reduced in order to make the analysis process manageable and cost-efficient. Data reduction techniques include a data discretization technique which is used to reduce the number of values for a given continuous attribute by dividing the range of the attribute into intervals. Interval labels can then be used to replace actual data values, data cube aggregation, dimension reduction (irrelevant or redundant attributes are removed), and data compression (data is encoded to reduce the size), numerously reduction (models or samples are used instead of the actual data) [***52***].

In this research the attributes was discretized to reduce the unlike values of the attributes in order to obtain Knowledge (patterns), and to make the data set suitable for data mining tools. Data transformation is necessary for adjusting missing values and categorical variables that take on too many values, and to bring information to the surface by creating new variables to represent trends and other ratios and combinations. The table shows that the attributes discretize into different bins.

Table 3.2 attributes with the class values range

Attributes	Value Range	Meaning
Rainfall (mm)	[0.00 - 0.1)	No Rain
	[0.1 - 4.9)	Light
	[4.9 - 25.0)	Moderate
	[25.0 - 100)	Heavy
Temprature (^{o}C)	[0 - 10]	Very Cold
	[10.1 - 16]	Cold
	[16.1 - 22]	Mild
	[22.1 - 28]	Warm
	[28.1 - 50)	Hot
Relative Humidity(%)	[0 - 50]	Very Dry
	[50.1 - 70]	Dry
	[70.1 - 90]	Medium Wet
	[90 - 100)	Wet
Wind Speed (m/s)	[0 - 1]	Light Wind
	[1.1 - 3]	Moderate Wind
	[3.1 - 5]	Strong Wind
	[5.1-10)	Very Strong Wind

In addition, the following data transformation and reformatting operation has been employed in order to create new attributes from the existing ones and to reformat the original values of some attributes in the dataset selected for analysis.

- Defining the class attributes: The class attribute is rainfall. This attribute is dependent variables that can help to classify the individual into groups. This classification would help to predict the likelihood that a given class is fallen at what condition with related to the independent variables.
- Defining the Relative Humidity: The derived attribute is an independent variable that can classify into different groups. This classification can help to identify in which groups the dependent variable are affected/changes its condition. This independent variables categorized into four groups. These are Very dry, Dry, Medium Wet and Wet.
- Defining Wind Speeds: This independent variable also categorized into four different groups. These are light wind, moderate wind, Strong wind and Very strong wind.
- Defining Temperature: This is an independent variables that can classify into five groups, i.e. Very Cold, Cold, Mild, Warm and Hot.

Therefore, the researcher discretized some attributes by converting numeric attributes to nominal: specify which attributes, number of bins and output binary attributes in order to get the best-fitted model. Table 3.3 provides summary of the original with their types.

Table 3.3 Summary of the original attributes

No	Original Attributes	Types
1	PRECIP	Nominal
2	TMPMAX	Nominal
3	TMPMIN	Nominal
4	WINDLY	Nominal
5	RELHUM	Nominal
6	SUNHRS	Nominal

3.2.3. DATA PREPARATION FOR WEKA TOOLS

WEKA was developed at the University of Waikato in New Zealand, and the name stands for Waikato Environment for Knowledge Analysis. It runs on almost any platform and has been tested under Linux, Windows, and Macintosh operating systems and even on a personal digital assistant. It provides a uniform interface to many different learning algorithms, along with methods for pre- and post-processing and for evaluating the result of learning schemes on any given dataset. WEKA is a collection of machine learning algorithms for solving real-world data mining problems. It contains 41 different algorithms for classification and numeric prediction [*7*]. A number of data mining methods were implemented and experimented in the WEKA software. Some of them were based on decision trees like the J48 decision tree, some are rule-based like PART and JRip, decision tables, and some of them are based on probability and regression, like the Naïve Bayes algorithms were implemented.

The researcher first has converted the original Microsoft Excel into Comma Separated Value (CSV). Then preprocessing activities are performed and the file is saved into WEKA acceptable comma separated values (CSV) or comma delimited file format. WEKA native data format is known as the ARFF (Attribute Relation File Format). It is basically a CSV (comma separated value) format with some extra headers to specify what type each attribute is (numerical, binary, nominal). The CSV file format is converted into ARFF by

using WEKA mining software, to take advantage of easier data manipulation and also compatible interaction with WEKA software.

3.3. MODEL BUILDING

According to the CRISP-DM Methodology, the next step from data preparation is model building. Data modeling refers to a group of processes in which multiple sets of data are combined and analyzed to uncover relationships or patterns. The goal of data modeling is to use past data to inform future efforts. The major activities of model building in CRISP-DM are selecting model techniques, generate test design, build Model and assess model.

3.3.1. SELECTING MODELING TECHNIQUES

The first step in model building is selecting the actual modeling techniques that are to be used. The purpose of this research is to develop a predictive model for weather condition especially rainfall, classification algorithms has been used for building the model. The analyses were performed by WEKA 3.6.10 software. In WEKA, there are more than 41 classification algorithms have been included in the software. Classification algorithm can be classified into two Rule-based classification and Decision tree induction.

A Rule-based classifier uses a set of IF-THEN rules for classification. Rules can be extracted from a decision tree. Rules may also be generated directly from training data using sequential covering algorithms. Decision tree induction is a top-down recursive tree induction algorithm, which uses an attribute selection measure to select the attribute tested for each non leaf node in the tree [***25***]. The researcher mainly focuses on the classification algorithm J48, Multilayer Perceptron and PART rule induction.

Classification is the process of building a model of class from a set of records that contains class labels. Chapman et al. [***17***], stated that "when learning classification rules, the system had to find the rules that predict the class-label, which is the predicted attribute's value, from the predicting attributes' value". Decision tree algorithm is to find out the way the attributes vector behaves for a number of instances. Also on the bases of the training instances the classes for the newly generated instances are being found [***55***]. This algorithm generates the rules for the

prediction of the targeted variable. With the help of tree classification algorithm the critical distribution of the data is easily understandable.

J48 is an open source java implementation of the C4.5 algorithm in WEKA data mining tools. C4.5 is a program that creates a decision tree based on a set of labeled input data. This algorithm was developed by Ross Quinlan. The decision tree generated by C4.5 can be used for classification and for this reason; C4.5 is often referred as a statistical classifier. J48 is an extension of ID3. The WEKA tool provides a number of options associated with tree pruning. In case of potential over fitting pruning can be used as a tool for précising. In other algorithms the classification is performed recursively till every single node/leaf is pure, that is the classification of the data should be as perfect as possible. This algorithm generates the rules from which particular identity of that data is generated. There is a lot of parameter to use in this algorithm for instance binary split, level of Confidence, pruning etc.

An artificial neural network (ANN), often just called a "neural network" (NN), is a mathematical model or computational model based on biological neural networks, in other words, they imitate the way the human brain learns and use rules inferred from data patterns to construct hidden layers of logic for analysis [***56***]. The feed forward neural network was the first and arguably simplest type of artificial neural network devised. In this network, the information moves in only one direction, forward, from the input nodes, through the hidden nodes (if any) and to the output nodes. A neural network is feed forward if there exists a method, which numbers all the nodes in the network such that there is no connection from a node with a large number to a node with a smaller number. All the connections are from nodes with small numbers to nodes with larger numbers. A neural network is recurrent if such a numbering method does not exist. Contrary to feed forward networks, recurrent neural networks are models with bidirectional data flow. While a feed forward network propagates data linearly from input to output, recurrent neural networks also propagate data from later processing stages to earlier stages.

PART is a separate-and-conquer rule learner proposed by [***7***]. The algorithm generates sets of rules called ‘decision lists’ which are ordered set of rules. PART builds a partial C4.5 decision tree in each iteration and converts the "best" leaf into a rule. PART algorithm combines the divide-and-conquer strategy (the top-down approach) for decision tree construction with the separate-and-conquer approach for rule learning. The separate-and-conquer strategy first builds a

rule and then removes those instances that the rule covers. These consecutive activities continue recursively for the remaining instances until none are left which generates sets of rules called 'decision lists' or ordered set of rules.

3.3.2. GENERATE TEST DESIGN

The most important thing to remember about model building is that it is an iterative process. The process of building predictive models requires a well-defined training and validation protocol in order to insure the most accurate and robust predictions. This kind of protocol is sometimes called supervised learning. The essence of supervised learning is to train your model on a portion of the data, then test and validate it on the remainder of the data. A model is built when the cycle of training and testing is completed. In this phase, the researcher uses both testing methods i.e. percentage split and 10-fold cross validation, with percentage split the dataset divided into two groups; these are one for model training and one for model testing. In WEKA by default the training set 66% of the instances in the data set is used and for the test set the remaining part. Cross-validation is especially used when the amount of data is limited. Instead of reserving a part for testing, cross-validation repeats the training and testing process several times with different random samples.

The 10-fold cross-validation is the data is divided randomly into 10 parts in which the classes are represented in approximately the same proportions as in the full dataset (stratification). Each part is held out in turn and the algorithm is trained on the nine remaining parts; then its error rate is calculated on the holdout set. Finally, the 10 error estimates are averaged to yield an overall error estimate. Further reading about this topic and an explanation for the preference for 10-fold cross-validation can be found in [*7*]. Therefore, the model is built on the training data and its quality estimated on the test set. The quality of the data measured in data mining is the error rates. This error rate measured in classification accuracy, the standard accuracy measurement in data mining is precision and recall.

CHAPTER FOUR

4. EXPERIMENTATIONS AND DATA ANALYSIS

This chapter mainly discusses the models to be built and experiments carried out together with their analysis. The experiments were run on a larger dataset in order to address the main objectives of the research study with respect to the minimum data set that consists of nine attributes. This will help in understanding the different stages that were used in various data mining algorithms.

In this study an attempt was made to design a model that enables to predict the rainfall status in Hawassa station of Ethiopian Meteorological Agency. To this end, J48 decision tree, Neural Network-multilayer perceptron and PART rule induction are the algorithms with which predictive model building experiments are conducted. These algorithms split the dataset to learn a model and test its performance on a dataset prepared for a study. In 10 fold cross validation, one option in WEKA for the purpose mentioned; the dataset is split into 10 equal parts. The algorithm is trained on nine–tenth of the dataset and then the classifier is tested on one-tenth. This way, the error of the resultant model will be the average of all the models found during each fold or iteration. Meteorology database was consulted to extract the dataset required for training and testing the models created by the classifiers. For creating predictive model a total size of 4749 records were used for training and testing. After applying SMOTE techniques the dataset size is 15389. The validations were done using 10-fold cross validation and percentage (%) split test option.

4.1. DATA PREPARATION

The data analysis and classification was carried out using WEKA Software application. The data set collected from the meteorology agency consists of 4749 records of weather data and the researcher also considered the initial data set for this study. A dataset of weather in Hawassa station was imbalanced if the classification categories are not approximately equally represented [***57***]. Performance of data mining algorithms is typically evaluated using predictive accuracy. However, this is not appropriate when the data is imbalanced and/or the cost difference of error is large. In the case of National Meteorological Agency of Ethiopia meteorology data, the class variable (Rainfall) status has a higher imbalance. Therefore, the researcher used Synthetic

Minority Oversampling Technique (SMOTE) automatic operation by filter where minority classes are over sampled by generating synthetic examples of minority class and adding them to the dataset. This way, the class distribution in the dataset changes and probability of correctly classifying minority class increases.

Many real world data mining applications involve learning from imbalanced datasets, where the particular events of interest may be very few when compared to the other classes. Learning from data sets that contain rare events usually produces biased classifiers that have a higher predictive accuracy over the majority class, but poorer predictive accuracy over the minority class of interest.

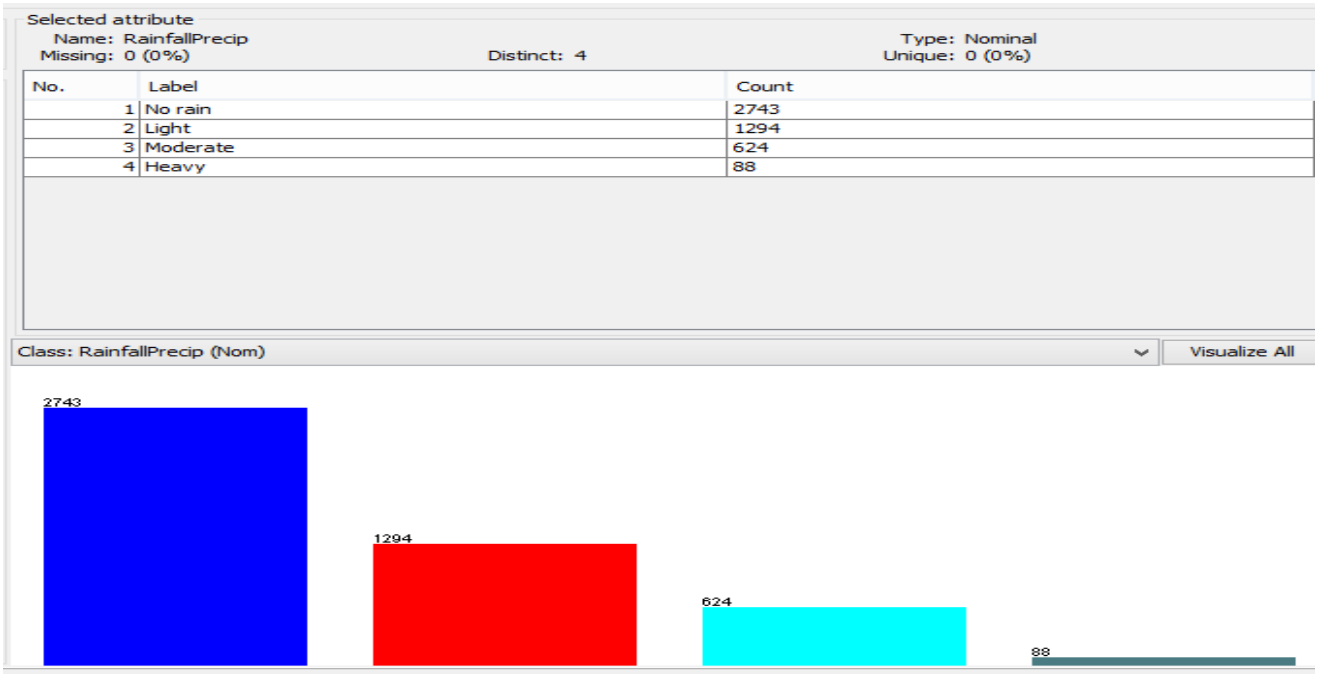


Figure 4.1 Class distribution in a data set based on 'Rainfall' as a target class before 'SMOTE'

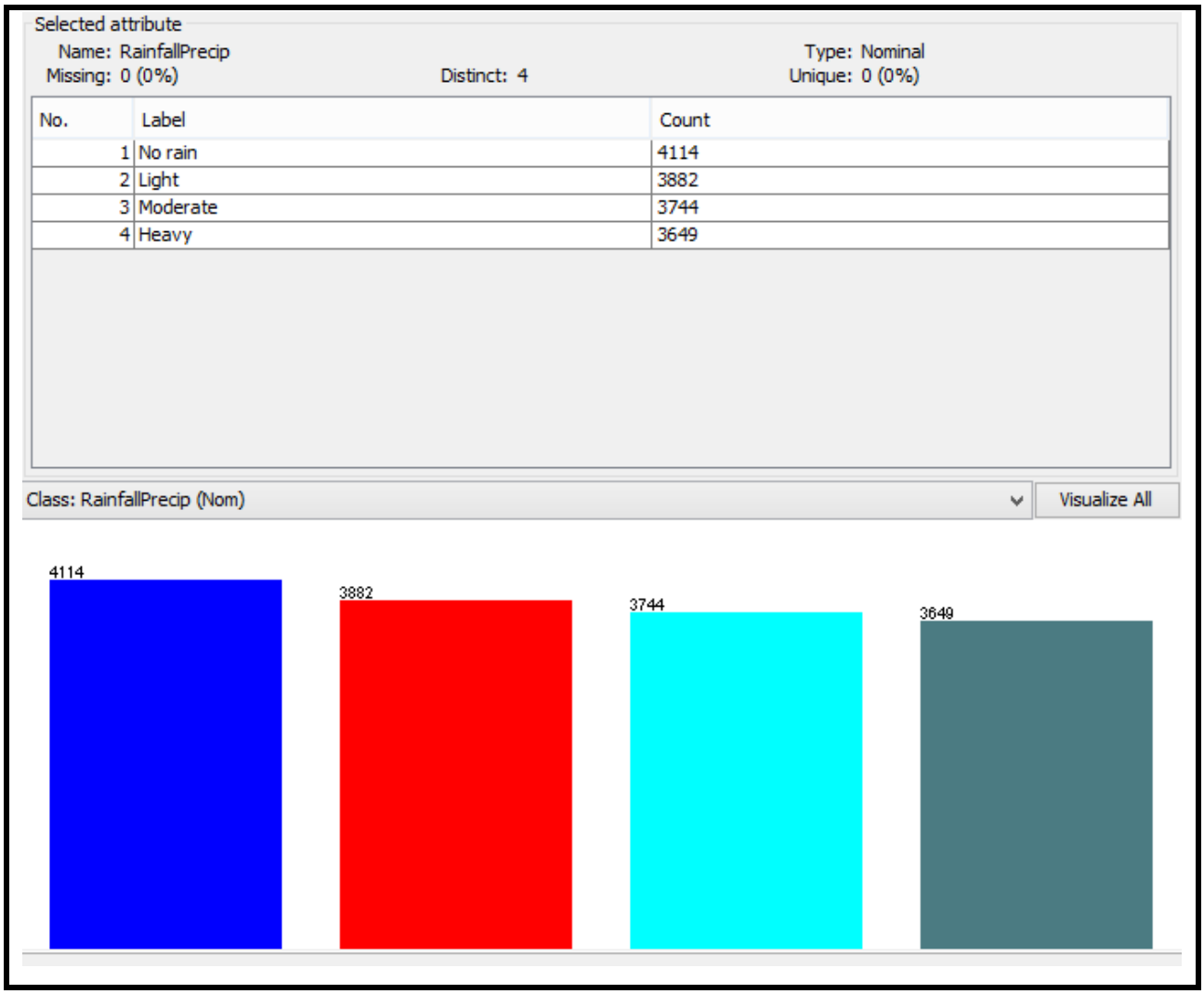


Figure 4.2 Class distribution in a data set based on 'Rainfall' as a target class after SMOTE Applied

As you can see from the above two figures i.e. Figure 4.1 & 4.2, the class attribute after SMOTE operation applied to the minority class. Originally the majority class of no Rain was 2743 and the minority class of Heavy rain was 88. Therefore, there is an imbalanced data set for implementing the classification techniques. So after SMOTE techniques the gap between the class attributes is almost the same or slightly difference.

4.2. MODEL BUILDING USING J48 DECISION TREE

A decision tree is a predictive machine-learning model that decides the target value (dependent variable) of a new sample based on various attribute values of the available data. The internal nodes of a decision tree denote the different attributes; the branches between the nodes tell us the possible values that these attributes can have in the observed samples, while the terminal nodes tell us the final value (classification) of the dependent variable.

A decision tree is a classifier in which previously unobserved records can be fed into the tree. At each node it will be sent either left or right according to some test. Ultimately, it will reach a leaf node and be given the label associated with that leaf. From the results of the decision tree classifier, it is possible to generate interesting rules. In fact, decision tree methods are often chosen for their ability to generate understandable rules in addition to their classification and prediction capabilities.

The classification C4.5 algorithms were implemented in WEKA to build a classification model in such a way that testing the model would be possible after training it. For most of the experiments carried out in this phase, the experiment was percentage (%) split test options was used, the total record was partitioned in to two, the training and test datasets.

Experiments are performed with various method parameters, mainly changing types of decision tree parameters. The J48 classifier window enables us to switch the parameters to build different decision tree scenarios.
It is also important to describe the J48 classifier parameters those allow us for intelligently adjusting them. J48 classifier Parameter Options are described which is taken from WEKA Manual.

The details of the decision tree used in WEKA are discusses in chapter two. For the decision tree to be created, algorithms are required to be executed from the training data following the test data. Once the trees are extracted and selected the best one, the rule is created based on the tree and the association between the attributes. Classification on the test data is done based on the decision tree that is created.

All the classification techniques have similar screens. The bottom right section of the screen displays the classifier output. The classifier outputs results based on the majority class, that is, the outcome of the experiment which is always the class with maximum number of cases. This is considered the study case in this research and also takes the least computation time. Classification of data and the confusion matrix is displayed in the classifier output screen below the decision tree.

Experiments are performed with various methods, mainly changing types of decision tree parameters. The J48 classifier window enables us to switch the parameters to build different decision tree experiment. It is also important to describe the J48 classifier parameters those allow us for intelligently adjusting them. Below in Table 4.1 J48 classifier Parameter Options are described which is taken from WEKA Manual.

Table 4.1 J48 decision tree parameter option of WEKA

Parameter Option	Description
binarySplits	Whether to use binary splits on nominal attributes when building the trees
confidenceFactor	The confidence factor used for pruning (smaller values incur more pruning)
debug	If set to true, classifier may output additional info to the console
minNumObj	The minimum number of instances per leaf
numFolds	Determines the amount of data used for reduced-error pruning. One fold is used for pruning, the rest for growing the tree
reducedErrorPruning	Whether reduced-error pruning is used instead of C.4.5 pruning
saveInstanceData	Whether to save the training data for visualization
seed	The seed used for randomizing the data when reduced-error pruning is used
subtreeRaising	Whether to consider the subtree raising operation when pruning
unpruned	Whether pruning is performed
useLaplace	Whether counts at leaves are smoothed based on Laplace

Based on WEKA data mining tool implemented J48 classifier with different parameter, the researcher employs J48 for applying decision tree classification model on the weather data of National Meteorological Agency of Ethiopia in Hawassa station dataset preprocessed as in the previous chapter.

There are thirteen experiments that are experimented for decision tree classification in this research. These experiments are analyzed to compare them to each other in terms of different performance matrices values, accuracies, number of leaves, and size of tree generated, ROC curves and execution time.

The experiments for decision tree classification that are experimented in this research are as listed below.

Experiment 1: J48 decision tree algorithm with Unpruned, confidence factor 0.25, default Minimum number of instance (minNumObj) for a leaf of 2 and 10-fold cross validation test mode.

Experiment 2: J48 decision tree algorithm with Unpruned, confidence factor 0.25, default Minimum number of instance (minNumObj) for a leaf of 2 and 66% split test mode.

Experiment 3: J48 decision tree algorithm with pruned, confidence factor 0.25, default Minimum number of instance (minNumObj) for a leaf of 2 and 70% split test mode.

Experiment 4: J48 decision tree algorithm with Unpruned, confidence factor 0.25, default Minimum number of instance (minNumObj) for a leaf of 2 and 80% split test mode.

Experiment 5: J48 decision tree algorithm with Unpruned, confidence factor 0.25, default Minimum number of instance (minNumObj) for a leaf of 2 and 85% split test mode.

Experiment 6: J48 decision tree algorithm with Unpruned, confidence factor 0.25, default Minimum number of instance (minNumObj) for a leaf of 2 and 90% split test mode.

Experiment 7: J48 decision tree algorithm with pruned, confidence factor 0.25, default Minimum number of instance (minNumObj) for a leaf of 2 and 85% split test mode.

Experiment 8: J48 decision tree algorithm with pruned, confidence factor 0.25, default Minimum number of instance (minNumObj) for a leaf of 2 and 10-fold cross validation test mode.

Experiment 9: J48 decision tree algorithm with Unpruned, confidence factor 0.30, default Minimum number of instance (minNumObj) for a leaf of 2 and 85% split test mode.

Experiment 10: J48 decision tree algorithm with pruned, confidence factor 0.30, default Minimum number of instance (minNumObj) for a leaf of 2 and 85% split test mode.

Experiment 11: J48 decision tree algorithm with Unpruned, confidence factor 0.35, default Minimum number of instance (minNumObj) for a leaf of 2 and 85% split test mode.

Experiment 12: J48 decision tree algorithm with Unpruned, confidence factor 0.25, default Minimum number of instance (minNumObj) for a leaf of 10 and 85% split test mode.

Experiment 13: J48 decision tree algorithm with Pruned, confidence factor 0.15, default Minimum number of instance (minNumObj) for a leaf of 2 and 85% split test mode.

Classification accuracy is not sufficient as a standard performance measure. ROC analysis and metrics such as precision, recall and F-measure have been used to understand the performance of the learning algorithm on the minority class.

These experiments were analyzed to compare them in terms of different performance matrices values, accuracies, size of trees, no. of leaves, time taken in sec. in the execution, and ROC/AUC curve. The models were also compared with regard to the pattern or KD of the predictive model. Accuracy is the percentage of predictions that are correct. The precision is the measure of accuracy provided that a specific class has been predicted. The sensitivity is the measure of the ability of a prediction model to select instances of a certain class from a data set. The specificity corresponds to the true negative rate which is commonly used in two class problems.

Table 4.2 Experiments result of J48 decision tree

S.No	Performance Measure	Experiment												
		1	2	3	4	5	6	7	8	9	10	11	12	13
1	Testing Mode	10 fold	66%	70%	80%	85%	90%	85%	10 Fold	90%	85%	85	85	85
2	Confidence factor	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.3	0.3	0.35	0.25	0.15
3	Prunning	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE
4	minNumObj	2	2	2	2	2	2	2	2	2	2	2	10	2
5	Size of Tree	7535	7535	3363	7535	7535	7535	3363	3363	7535	3814	4098	1073	2846
6	No. of Leaves	6884	6884	3013	6884	6884	6884	3013	3013	6884	3423	3689	964	2549
7	Time taken to Build	0.44	1.76	0.5	0.17	0.35	0.17	0.28	0.43	0.16	0.27	0.38	0.2	0.41
8	Recall	86.1	84.9	82.8	86.4	86.7	85.8	84.4	83.5	86.7	84.7	85.1	78.1	83.8
9	Precision	86	84.8	82.7	86.3	86.6	85.8	84.3	83.4	86.6	84.6	85.1	77.8	83.7
10	F-Measure	86	84.9	82.8	86.4	86.6	85.8	84.4	83.5	86.6	84.6	85.1	77.9	83.7
11	ROC	94.5	93.8	93.1	94.5	95	94.6	94.2	93.5	94.5	94.4	94.4	92.3	94.1
12	Correctly Classified	13243	4444	3825	2660	2000	1321	1984	12851	2000	1954	1965	1803	1933
13	Incorrectly Classified	2146	788	792	418	308	218	360	2538	308	354	343	505	375
14	Mean Absolute Error	0.0804	0.0887	0.114	0.0807	0.079	0.0818	0.105	0.1084	0.079	0.1018	0.098	0.15	0.112
15	Accuracy	86.055	84.94	82.85	86.42	86.65	85.84	84.4	83.51	86.65	84.66	85.13	78.11	83.75

As can be observed from this Table 4.2, experiments 5 and 9 have comparatively better than the other experiments with extracted accuracies and rules. This is because, the researcher used to build J48 decision tree with default confidence factor (i.e. 0.25) and 90% split test mode and also the pruned parameter of the classifier. This result portrays that due to the adjustment of some of the parameters, the size of tree reduced to 7535 and the number of leaves has become 6884.

The model has accuracy of 86.65% using 90% split and 86.05% accuracy using 10-fold cross-validation test options. Moreover, the model has a true positive rate of 86.7% and false positive of 4.4% for 90% test option and also a true positive rate of 86.1% and false positive rate of 4.7% for 10-fold cross-validation. The best J48 decision tree model of the classification generated from experiment 9 of the 90% split test mode. The model shows a better performance evaluation than other models. The 90% split test model also scored a better performance than 10-fold cross-validation. Therefore, the test options mode used to build the decision tree for experiment 9 with 90% split test mode options which is J48 pruned decision tree with default confidence factor (i.e. 0.25), are statistically significant in splitting the decision tree.

From the above Table 4.2, since experiment 5 and 9 were carried out to construct the required decision tree with 85% and 90% split test mode and had a reasonably good accuracy respectively. In addition to this, experiment 9, which is the corresponding rule extraction

experiment from the J48 decision tree constructed, was selected. In this regard, generally, the reasons of selecting experiments 9 from all the experiments carried out could be mentioned follows:

- The number of records considered is relatively large.
- The number of leaves and size of the tree in experiment 9 are manageable; and the number of rules extracted in experiment 9 is reasonable.
- The test mode, which is the 90% split test mode, used in the experiment 9 is acceptable.
- The accuracy of the resulting model is comparatively better than others.

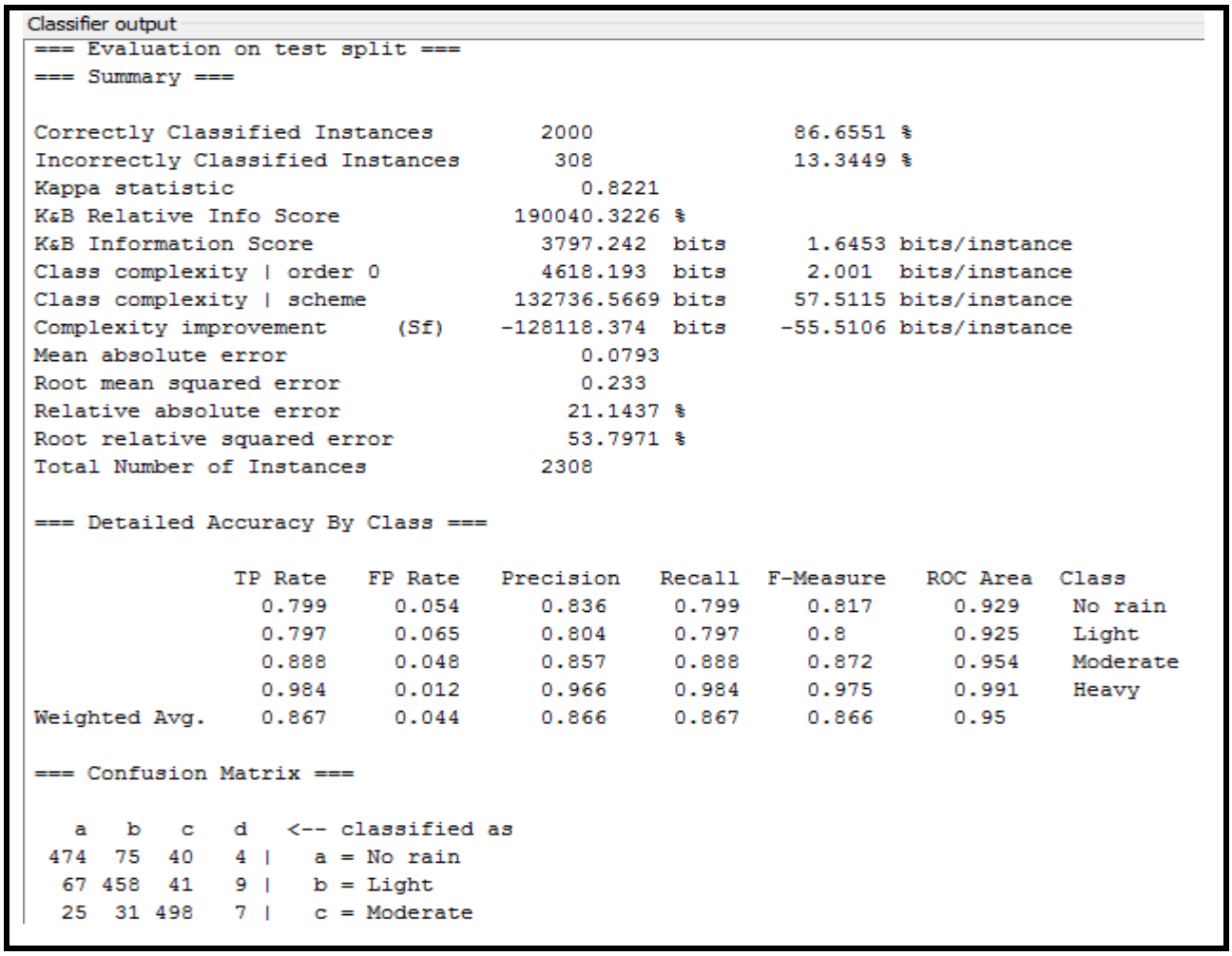
As a result the full outputs of the selected working J48 decision tree in experiment 9 and the corresponding rules extracted in experiment 9 was annexed for reference.

4.2.1. CONFUSION MATRIX FOR J48 DECISION TREE MODEL

According to [***58***], a confusion matrix contains information about actual and predicted classifications done by a classification system. Performance of such systems is commonly evaluated using the data in the matrix.

A confusion matrix provides a quick understanding of model accuracy and the types of errors the model makes when scoring records. It is the result of a test task for classification models. Moreover, the overall predictive accuracy on unseen instances it is often helpful to see a breakdown of the classifier's performance.

The confusion matrix for J48 decision tree presented below in table 4.3 depicts that out of the total records provided to the WEKA system 2000 (86.65%) and 308(13.34%) records were classified correctly in the class of No rain and others classes respectively. On the other hand, 75 (12.65%) records were incorrectly classified as "Light" while actually they were supposed to be in the "No rain" class, 40 (6.74%) records were classified incorrectly as "moderate" while actually they are in the "No rain" class, and 4 (0.67%) records were classified incorrectly as "heavy" while actually they are in the "No rain" class.



```
Classifier output
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances        2000               86.6551 %
Incorrectly Classified Instances       308               13.3449 %
Kappa statistic                          0.8221
K&B Relative Info Score             190040.3226 %
K&B Information Score                 3797.242  bits      1.6453 bits/instance
Class complexity | order 0            4618.193  bits      2.001  bits/instance
Class complexity | scheme           132736.5669 bits     57.5115 bits/instance
Complexity improvement     (Sf)    -128118.374  bits    -55.5106 bits/instance
Mean absolute error                      0.0793
Root mean squared error                  0.233
Relative absolute error                 21.1437 %
Root relative squared error             53.7971 %
Total Number of Instances             2308

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.799     0.054      0.836     0.799     0.817      0.929    No rain
                 0.797     0.065      0.804     0.797     0.8        0.925    Light
                 0.888     0.048      0.857     0.888     0.872      0.954    Moderate
                 0.984     0.012      0.966     0.984     0.975      0.991    Heavy
Weighted Avg.    0.867     0.044      0.866     0.867     0.866      0.95

=== Confusion Matrix ===

   a   b   c   d   <-- classified as
 474  75  40   4 |   a = No rain
  67 458  41   9 |   b = Light
  25  31 498   7 |   c = Moderate
```

Figure 4.3 Classifier output based on J48 Decision Tree

=== Confusion Matrix ===

Table 4.3 Confusion matrix of J48 Decision tree

	Predicted Class				
	a	b	c	d	<-- classified as
Actual Class	474	75	40	4	a = No Rain
	67	458	41	9	b = Light
	25	31	498	7	c = Moderate
	1	6	2	570	d = Heavy

Consecutively, to see how well the predictive model can recognize "No Rain" tuples (the positive records) and how well the predictive model which has the classifier can recognize "Rain" tuples (the negative records) which have sensitivity and specificity measures can be used. Sensitivity is also known as the true positive cases in the test data with predicted probabilities

greater than or equal to the probability threshold (correctly predicted), while specificity is the true negatives rate: Negative cases in the test data with predicted probabilities strictly less than the probability threshold (correctly predicted). Furthermore, the classifier has 79.9% sensitivity and 93.16% specificity; which discloses that the J48 decision tree classifier has an acceptable capability of recognizing the true class value.

4.2.2. ROC ANALYSIS FOR J48 DECISION TREE MODEL

According to Altman and Bland cited in the critical step before any data mining model can be used in routine clinical practice is to compare its performance with equivalent statistical methods like sensitivity and specificity. ROC (receiver operating characteristics) curves that originated from signal detection theory has added more value to these two measures by creating trade-off. AUC (Area Under Curve) is a measure of the area under the ROC curve. ROC curve is a two-dimensional graph to select possibly optimal models based on the TP rate and FP rate. It also represents trade-of between benefits (TP) and costs (FP).

In the ROC curve, the sensitivity (TP) rate is represented on the Y-axis and the 1-specificity (FP) rate on the X-axis. Each prediction result or one instance of a confusion matrix represents one point in the ROC space. Several points on a ROC graph should be noted. The lower left point (0, 0) represents that the classifier labeled all instances out of their actual class. The upper right point (1, 1) is the case where all instances are classified in their actual class. The point (0, 1) represents perfect classification and the line y = x defines the strategy of randomly guessing the class. In order to assess the overall performance of a classifier, the fraction of the total area that falls under the ROC curve is considered. AUC varies between 0 and 1. Larger AUC values indicate generally better classifier performance.

As can be seen from the detailed accuracy by class output, the ROC (Receiver Operating Characteristics) area of this model is highest (0.9293). The Area under the ROC curve in figure 4.4 is higher. Higher numbers here indicates the model is the more accurate than the others. Proper utilization of pruning methods and techniques has shown to increase classification accuracy given an induced decision tree. But the size of the tree is very large and complex to interpret. In the case of unpruned tree construction, the confidence factor has no effect for

unpruned tree experiments. As we can see in table 4.2, Experiment 5 and 9, when confidence factor increase the values of the performance evaluators does not change in the case of the unpruned tree construction. When the number of instance increases (Experiment 12), the values of model accuracy, True positive rate, ROC area curve, precision, and recall are less than the selected model (Experiment 9). But the numbers of leaves and size of tree is smaller than the other experiments.

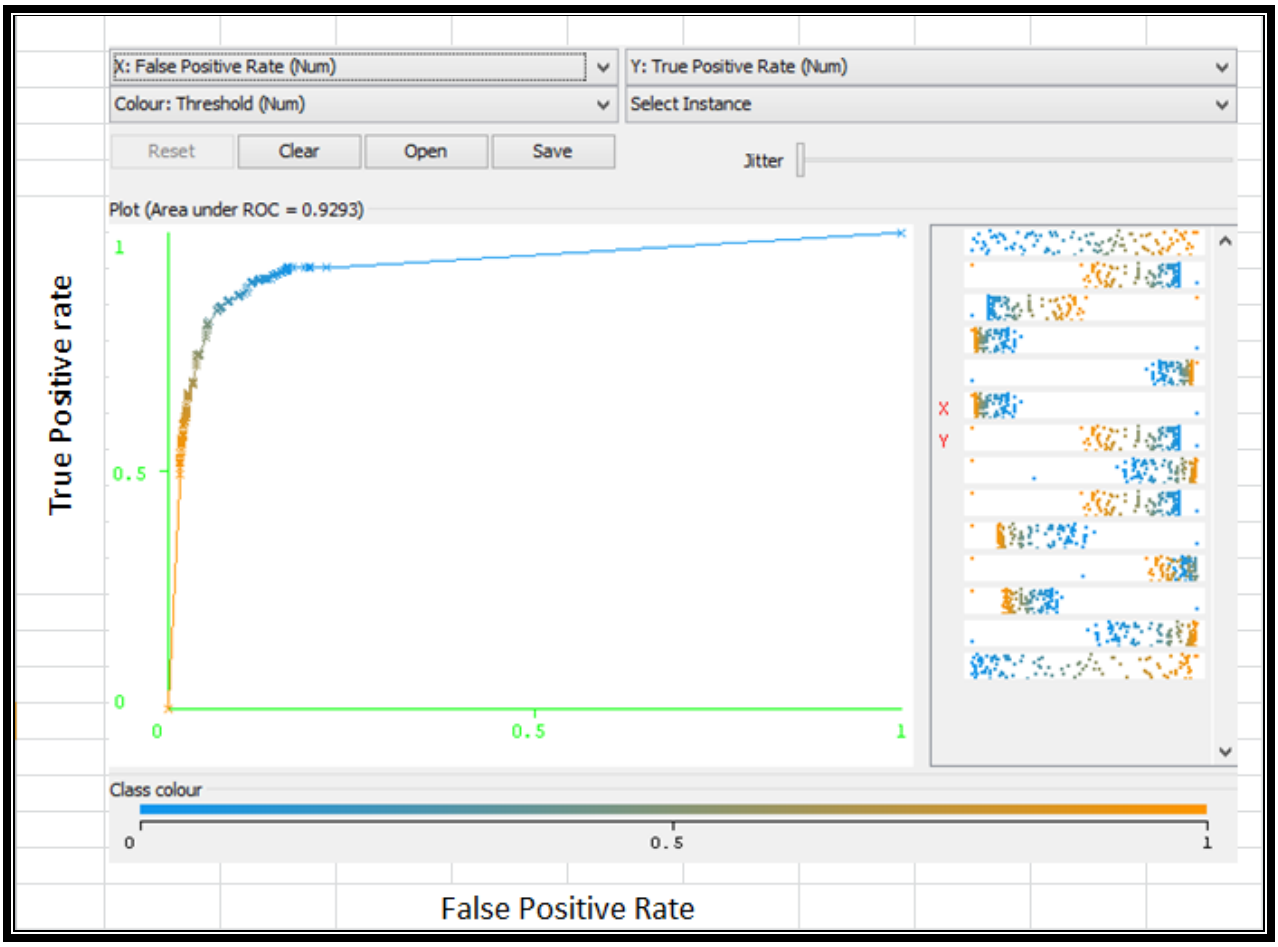


Figure 4.4 ROC Area curves of J48 Decision Tree

ROC curves are similar to lift charts in that they provide a means of comparison between individual models and determine thresholds which yield a high proportion of positive hits. In the above figure the horizontal axis of an ROC graph measures the false positive rate as a percentage. The vertical axis shows the true positive rate. The top left hand corner is the optimal location in an ROC curve, indicating high TP (true-positive) rate versus low FP (false-positive) rate.

On the other hand, increasing the minimum number of instances per leaf (minNumObj) can reduce the tree size and numbers of leaves. In this study, the researcher found that minNumObj option is useful for reducing the size of the decision tree, but there is an impact on accuracy of the model. As can be seen on table 4.2 Experiment 12 is better than the other experiments with less numbers of leave and size of tree. Therefore, this parameter is very important for producing minimized size of trees for extracting rule easily.

4.2.3. GENERATING RULES FROM J48 DECISION TREE

From the decision tree developed in the aforementioned experiments, it is possible to find out a set of rules simply by traversing the decision tree and generating a rule for each leaf and making a combination of all the tests found on the path from the root to the leaf node. This produces rules that are unambiguous in that it doesn't matter in what order they are executed. Decision tree and decision rule solutions offer a level of interpretability that is unique to symbolic models. The solutions may be directly inspected to understand the decision surfaces that exist in the data [48].

Rule 1:

If RELHUMIDITY is 'Very Dry' and sun_hrs is 'Long' then the class of the Rainfallpercept is likely to be 'No rain' (885.0/31.0). The level of assurance of the independent attribute for the status or the predicted class in the bracket can be calculated as follow:

- 885/(885+31)= 885/916= 0.966= 96.6%

Rule 2:

If RELHUMIDITY is 'Very Dry' and sun_hrs is 'Medium' and the Maximum Temperature (Max_Temp) is 'Hot' then the class of the Rainfallpercept is likely to be 'No rain' (48.0/2.0). The level of assurance of the independent attribute for the status or the predicted class in the bracket can be calculated as follow:

- 48/(48+2)= 48/50= 0.96= 96%

Rule 3:

If RELHUMIDITY is ‘Dry’ and sun_hrs is ‘Long’ and the Maximum Temperature (Max_Temp) is ‘Very Cold’ then the class of the Rainfallpercept is likely to be ‘No rain’ (543.0/37.0).

Rule 4:

If RELHUMIDITY is ‘Dry’ and sun_hrs is ‘Long’ and the Minimum Temperature (Min_Temp) is ‘Cold’ and Month is ‘January’ then the class of the Rainfallpercept is likely to be ‘No rain’ (309.0.0/93.0).

Rule 5:

If RELHUMIDITY is ‘Dry’ and sun_hrs is ‘Medium’ and the Minimum Temperature (Min_Temp) is ‘Cold’ and Month is ‘January’ and Wind Speeds is ‘light wind’ then the class of the Rainfallpercept is likely to be ‘Heavy’ (266.0.0/100.0).

Rule 6:

If RELHUMIDITY is ‘Dry’ and sun_hrs is ‘Long’ and the Minimum Temperature (Min_Temp) is ‘Cold’ and Month is ‘February’ and Wind Speeds is ‘light wind’ then the class of the Rainfallpercept is likely to be ‘No Rain’ (210.0.0/91.0).

Rule 7:

If RELHUMIDITY is ‘Dry’ and sun_hrs is ‘Medium’ and the Minimum Temperature (Min_Temp) is ‘Warm’ and Month is ‘May’ and Wind Speeds is ‘light wind’ then the class of the Rainfallpercept is likely to be ‘Heavy’ (202.0.0/9.0).

Rule 8:

If RELHUMIDITY is ‘Dry’ and the Minimum Temperature (Min_Temp) is ‘Cold’ and Month is ‘August’ and Wind Speeds is ‘light wind’ then the class of the Rainfallpercept is likely to be ‘Moderate’ (161.0.0/87.0).

Rule 9:

If RELHUMIDITY is ‘Medium wet’ and sun_hrs is ‘Long’ and the Minimum Temperature (Min_Temp) is ‘Cold’ and Month is ‘March’ and Wind Speeds is ‘light wind’ then the class of the Rainfallpercept is likely to be ‘Heavy’ (57.0.0/32.0).

Rule 10:

If RELHUMIDITY is ‘Medium wet’ and sun_hrs is ‘Medium’ and the Minimum Temperature (Min_Temp) is ‘Cold’ and Month is ‘March’ then the class of the Rainfallpercept is likely to be ‘Moderate’ (139.0.0/58.0).

Rule 11:

If RELHUMIDITY is ‘Medium wet’ and sun_hrs is ‘Long’ and the Minimum Temperature (Min_Temp) is ‘Cold’ and Month is ‘May’ then the class of the Rainfallpercept is likely to be ‘Heavy’ (534.0.0/191.0).

Rule 12:

If RELHUMIDITY is ‘Medium wet’ and sun_hrs is ‘Medium’ and the Minimum Temperature (Min_Temp) is ‘Mild’ and Month is ‘May’ and the Maximum Temperature (Max_Temp) is ‘Warm’ then the class of the Rainfallpercept is likely to be ‘Heavy’ (242.0.0/43.0).

Rule 13:

If RELHUMIDITY is ‘Medium wet’ and Month is ‘August’ and Wind_Speeds is ‘Moderate’ then the class of the Rainfallpercept is likely to be ‘Light’.(138.0/61.0).

Rule 14:

If RELHUMIDITY is ‘Medium Wet’ and MONTH is ‘March’ and minimum temperature is ‘Mild’ and the wind speeds is ‘Light Wind’ then the class of the Rainfallpercept is likely to be ‘Heavy’. (72.0/9.0)

Rule 15:

If RELHUMIDITY is 'Medium Wet' and MONTH is 'May' and the wind speeds is 'Light Wind' and minimum temperature is 'Mild' and the sunshine hours is 'Medium' and the maximum temperature 'Warm' then the class of the Rainfallpercept is likely to be 'Heavy' (242.0/43.0)

The above rules indicate how a given record could be classified based on some attribute values to construct rules and provided the class predicted by the rule. Hence, having these rules, instances were classified into the predefined classes. In fact, in classifying the rainfall records into the predefined classes, from the all attributes Year and Day were not occurred in the above generated rules which have a base for the classification tasks. The numerical value, which written at the end of the predicted class in bracket, indicates the level of assurance of the independent attribute for the status or the predicted class.

4.3. MODEL BUILDING USING ARTIFICIAL NEURAL NETWORK

WEKA's Multilayer Perceptron algorithm has several parameters which can influence its performance. In this experiment, the performance of neural network in predicting Rainfall was evaluated. Thirteen experiments were considered: building model using multilayer Perceptron with the entire attributes. The hidden Layers parameter, for instance, is used to set the number of hidden layers (if available in the network). The predefined value for hidden layer is a, which is the average of number of predictor variables and number of class values. Other important parameters include learning rate and momentum. This experiment was done on the default parameter values for all parameter, and then Multilayer Perceptron was tested by varying learning rate and momentum, training time, hidden layer and testing mode.

A Multilayer Perceptron classifier uses back propagation to classify instances. This network can be built by hand, created by an algorithm or both. The network can also be monitored and modified during training time. The nodes in this network are all sigmoid (except for when the class is numeric in which case the output nodes become unthresholded linear units).

Table 4.4 Multilayer perceptron neural network parameter

Parameter Option	Description
GUI	Brings up a gui interface. This will allow the pausing and altering of the nueral network during training.
autoBuild	Adds and connects up hidden layers in the network.
decay	This will cause the learning rate to decrease. This will divide the starting learning rate by the epoch number, to determine what the current learning rate should be.This may help to stop the network from diverging from the target output, as well as improve general performance. Note that the decaying learning rate will not be shown in the gui, only the original learning rate. If the learning rate is changed in the gui, this is treated as the starting learning rate.
hiddenLayers	This defines the hidden layers of the neural network. This is a list of positive whole numbers. 1 for each hidden layer. Comma seperated. To have no hidden layers put a single 0 here. This will only be used if autobuild is set. There are also wildcard values 'a' = (attribs + classes) / 2, 'i' = attribs, 'o' = classes , 't' = attribs + classes.
learningRate	The amount the weights are updated.
momentum	Momentum applied to the weights during updating.
nominalToBinaryFilter	This will preprocess the instances with the filter. This could help improve performance if there are nominal attributes in the data.
normalizeAttributes	This will normalize the attributes. This could help improve performance of the network. This is not reliant on the class being numeric. This will also normalize nominal attributes as well (after they have been run through the nominal to binary filter if that is in use) so that the nominal values are between -1 and 1
normalizeNumericClass	This will normalize the class if it's numeric. This could help improve performance of the network, It normalizes the class to be between -1 and 1. Note that this is only internally, the output will be scaled back to the original range.
reset	This will allow the network to reset with a lower learning rate. If the network diverges from the answer this will automatically reset the network with a lower learning rate and begin training again. This option is only available if the gui is not set. Note that if the network diverges but isn't allowed to reset it will fail the training process and return an error message.
trainingTime	The number of epochs to train through. If the validation set is non-zero then it can terminate the network early

There are thirteen experiments that are experimented for multilayer perceptron algorithm classification in this research. These experiments are analyzed to compare them to each other in terms of different performance matrices values.

The experiments for multilayer perceptron algorithm classification that are experimented in this research are as listed below.

Experiment 1: Multilayer perceptron algorithm with a hidden layer is a, learning rate 0.3, momentum 0.2, training time 10 and 70% split test mode.

Experiment 2: Multilayer perceptron algorithm with a hidden layer is a, learning rate 0.3, momentum 0.2, training time 10 and 80% split test mode.

Experiment 3: Multilayer perceptron algorithm with a hidden layer is a, learning rate 0.3, momentum 0.2, training time 10 and 10-fold cross validation test mode.

Experiment 4: Multilayer perceptron algorithm with a hidden layer is 20, learning rate 0.3, momentum 0.2, training time 10 and 10-fold cross validation test mode.

Experiment 5: Multilayer perceptron algorithm with a hidden layer is 10, learning rate 0.3, momentum 0.2, training time 10 and 10-fold cross validation test mode.

Experiment 6: Multilayer perceptron algorithm with a hidden layer is a, learning rate 0.3, momentum 0.2, and training time 50 and 85% split test mode.

Experiment 7: Multilayer perceptron algorithm with a hidden layer is 15, learning rate 0.3, momentum 0.2, and training time 50 and 10-fold cross validation test mode.

Experiment 8: Multilayer perceptron algorithm with a hidden layer is 20, learning rate 0.3, momentum 0.2, and training time 50 and 85% split test mode.

Experiment 9: Multilayer perceptron algorithm with a hidden layer is a, learning rate 0.3, momentum 0.6, and training time 50 and 90% split test mode.

Experiment 10: Multilayer perceptron algorithm with a hidden layer is a, learning rate 0.3, momentum 0.2, and training time 150 and 85% split test mode.

Experiment 11: Multilayer perceptron algorithm with a hidden layer is a, learning rate 0.3, momentum 0.2, and training time 200 and 85% split test mode.

Experiment 12: Multilayer perceptron algorithm with a default parameter for hidden layer is a, learning rate 0.3, momentum 0.2, and training time 500 and 85% split test mode.

Experiment 13: Multilayer perceptron algorithm with a default parameter for hidden layer is a, learning rate 0.3, momentum 0.2, and training time 50 and 90% split test mode.

Table 4.5 Experimental result of multilayer perceptron neural network

No	Performance Measure	Experiment												
		1	2	3	4	5	6	7	8	9	10	11	12	13
1	Testing Mode	70%	80%	10 F	10 F	10 F	85%	10 F	85%	90%	85%	85%	85%	90%
2	Hidden layer	a	a	a	20	10	a	15	20	a	a	a	a	a
3	Learning Rate	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3
4	Momentum	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.6	0.2	0.2	0.2	0.2
5	Training Time	10	10	10	10	10	50	50	50	50	150	200	500	50
6	Time taken to Build	94.14	94.99	66.64	39.39	19.03	405.9	134.1	240.35	380.39	1129.1	2657.9	3308.4	328.74
7	Recall	74.2	76.7	75.1	72.3	69.3	80	76	77.7	65.2	81.7	82.9	83.9	81.4
8	Precision	74.1	77.1	75.4	73.2	70.3	80.2	76.4	78.1	67.3	82	83.2	83.9	82.2
9	F-Measure	74.1	76.4	75.1	72.5	69.6	79.9	76	77.6	62.4	81.7	82.9	83.9	81.6
10	ROC	90.3	92	90.3	89.2	87.3	92	90.1	91	86.5	92	92.4	92.5	92.2
11	Correctly Classified	3427	2360	11559	11128	10671	1847	11700	1793	1003	1885	1914	2582	1253
12	Incorrectly Classified	1190	718	3830	4261	4718	461	3689	515	536	423	394	496	286
13	Mean Absolute Error	0.176	0.16	0.1678	0.1857	0.205	0.122	0.159	0.1388	0.1755	0.1033	0.0966	0.0881	0.118
14	Accuracy	**74.23**	**76.67**	**75.11**	**72.31**	**69.34**	**80.03**	**76.03**	**77.68**	**65**	**81.67**	**82.93**	**83.88**	**81.41**

As can be observed from the above Table 4.5, experiments 12 have comparatively better than the other experiments with extracted accuracies. This is because, the researcher used to build multilayer perceptron algorithm default learning rate, momentum & hidden layer and 85% split test mode .

Multilayer Perceptron model built using the above experiment #12 correctly classified 2582 (83.89 %) instances and 496 (16.14%) instances were incorrectly classified. The best Multilayer Perceptron model of the classification generated from experiment #12 of the 90% split test mode. The model shows a better performance evaluation than other models. The 85% split test model also scored a better performance than 10-fold cross-validation. Therefore, the test options mode

used to build the Multilayer Perceptron for experiment #12 with 85% split test mode options which is Multilayer Perceptron algorithm with default learning rate, momentum and hidden layer. As shown in table 4.5, increasing the momentum from the default value in experiment #9 momentum=0.6, the performance of the result is very low as compared to the other experiment. On the other hand, the testing mode of the multilayer perceptron algorithm has a better performance on the percentage testing mode than the 10-fold cross validation but the latter took the smallest time to build the model.

4.3.1. ROC ANALYSIS FOR MULTILAYER PERCEPTRON MODEL

As can be seen from the detailed accuracy by class output, the ROC (Receiver Operating Characteristics) area of this model is 0.9025. The Area under the ROC curve in figure 4.5 is higher. Higher numbers here indicates the model is the more accurate than the others.

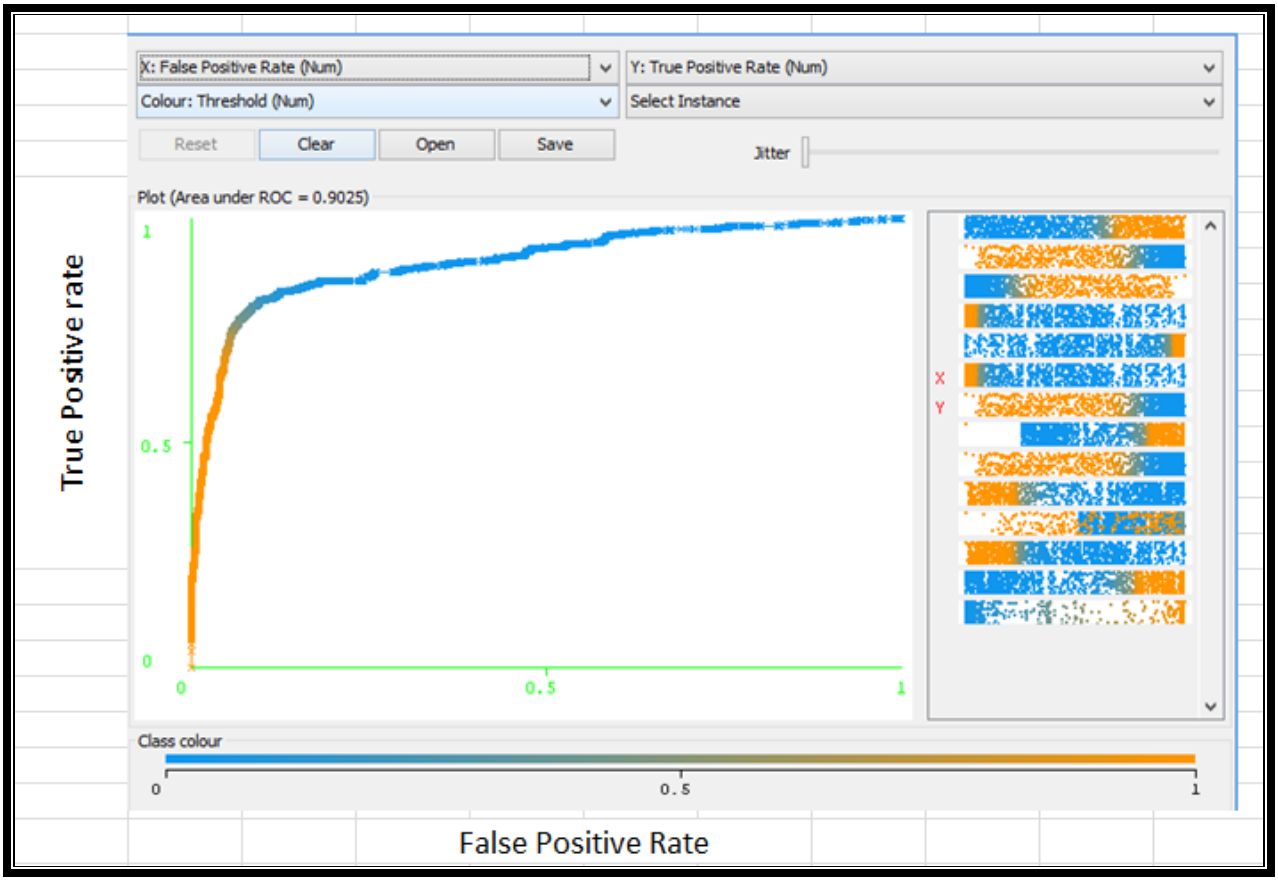


Figure 4.5 ROC Area Curve of Neural Network

4.4. MODEL BUILDING USING PART RULE INDUCTION

The researcher prefers PART rule induction algorithms over other rule induction algorithms because it has the ability and potential to produce accurate and readable rules. PART is a separate and conquer rule learner proposed by Eibe and Witten. The algorithm producing sets of rules called 'decision lists' which are ordered set of rules. A new data is compared to each rule in a list in turn, and the item is assigned the category of the first matching rule (a default is applied if no rule successfully matches.) PART builds a partial C4.5 decision tree in each iteration and makes the 'best' leaf into a rule. The algorithm is a combination of C4.5 and RIPPER rule learning.

Parameter option of PART Rule induction:

Table 4.6 PART Rule induction parameter

Parameter Option	Description
binarySplits	Whether to use binary splits on nominal attributes when building the partial trees.
confidenceFactor	The confidence factor used for pruning (smaller values incur more pruning).
debug	If set to true, classifier may output additional info to the console.
minNumObj	The minimum number of instances per rule.
numFolds	Determines the amount of data used for reduced-error pruning. One fold is used for pruning, the rest for growing the rules.
reducedErrorPruning	Whether reduced-error pruning is used instead of C.4.5 pruning.
unpruned	Whether pruning is performed.
seed	The seed used for randomizing the data when reduced-error pruning is used.

There are eight experiments that are experimented for PART rule induction algorithm classification in this research. These experiments are analyzed to compare them to each other in terms of different performance matrices values.

The experiments for PART rule induction algorithm classification that are experimented in this research are as listed below.

Experiment 1: PART rule induction algorithm with unpruned, binary split is false, confidence factor 0.25, default minimum number of instance (minNumObj) per rule of 2 and 70% split test mode.

Experiment 2: PART rule induction algorithm with unpruned, binary split is true, confidence factor 0.25, default minimum number of instance (minNumObj) per rule of 2 and 10-fold cross validation test mode.

Experiment 3: PART rule induction algorithm with pruned, binary split is false, confidence factor 0.3, default minimum number of instance (minNumObj) per rule of 2 and 70% split test mode.

Experiment 4: PART rule induction algorithm with pruned, binary split is false, confidence factor 0.3, default minimum number of instance (minNumObj) per rule of 2 and 80% split test mode.

Experiment 5: PART rule induction algorithm with pruned, binary split is false, confidence factor 0.3, default minimum number of instance (minNumObj) per rule of 2 and 85% split test mode.

Experiment 6: PART rule induction algorithm with pruned, binary split is true, confidence factor 0.25, default minimum number of instance (minNumObj) per rule of 2 and 85% split test mode.

Experiment 7: PART rule induction algorithm with pruned, binary split is true, confidence factor 0.25, default minimum number of instance (minNumObj) per rule of 2 and 90% split test mode.

Experiment 8: PART rule induction algorithm with pruned, binary split is false, confidence factor 0.25, default minimum number of instance (minNumObj) per rule of 2 and 10-fold cross validation test mode.

Table 4.7 Experimental result of PART rule induction

S.No	Performance Measure	Experiment							
		1	2	3	4	5	6	**7**	8
1	Testing Mode	70%	10 Fold	70%	80%	85%	85%	**90%**	10 Fold
2	pruning	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	**TRUE**	TRUE
3	Binary split	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	**TRUE**	FLASE
4	confidence factor	0.25	0.25	0.3	0.3	0.3	0.25	**0.25**	0.25
5	Number of rule	1000	1724	2276	2276	2276	1724	**1724**	1000
6	Time taken to Build	18.87	51.03	38.37	37.44	42.18	54.13	**49.36**	19.44
7	Recall	80.20%	86%	82.80%	84.70%	85%	85.70%	**86%**	82.50%
8	Precision	80.20%	86.10%	82.80%	84.70%	85%	85.90%	**86.20%**	82.50%
9	F-Measure	80.20%	86%	82.80%	84.70%	85%	85.80%	**86.10%**	82.40%
10	ROC	92%	92.40%	91.50%	92.50%	92.50%	92.30%	**92.20%**	93.10%
11	Correctly Classified	3705	13238	3821	2606	1961	1979	**1324**	12689
12	Incorrectly Classified	912	2151	796	472	347	329	**215**	2700
13	Mean Absolute Error	0.1229	0.0756	0.0981	0.089	0.0902	0.0781	**0.0774**	0.1139
14	Accuracy	80.25%	86.02%	82.75%	84.66%	84.96%	85.74%	**86.03%**	82.45%

As can be observed from the above Table 4.7, experiments 7 have comparatively better than the other experiments with extracted accuracies. This is because, the researcher used to build PART rule induction algorithm with pruned, binary split is true, confidence factor 0.25, default minimum number of instance (minNumObj) per rule of 2 and 90% split test mode.

PART rule induction model built using the above experiment #7 correctly classified 1324 (86.03 %) instances and 215 (13.97%) instances were incorrectly classified with recall (true positive rate) of 86%, a false positive rate of 4.7%, a precision of 86.2% and ROC curve area of 92.2%. The best PART rule induction model of the classification generated from experiment #7 of the 90% split test mode. Experiment #7 shows a better performance evaluation than other experiment. As can be seen from the above result, PART rule induction algorithm built model has an accuracy of 86.03%.

4.4.1. GENERATING RULES FROM PART RULE INDUCTION

PART rule learner with the specified scheme has resulted in 296 rules. Listing all the rules here will be quite awkward, thus, the rules which are highly predictive are selected and discussed as the finding of this study based on success ratio. The success ratio of a rule is found in parenthesis just at the end of the predictive rules. The numbers in parenthesis at the end of each rule tells the number of instances in the rule. If one or more of the rules were not pure, the numbers of

misclassified cases also are given after slash (/). The researcher has converted these numbers into percent to compare the chance of the rule to be correct with that of its chance to be incorrect. The greater the number before the parenthesis the greater the chance of the rule to predict the class indicated by that particular rule.

Rule 1:

If Max_Temp is 'Warm' AND MONTH is 'April' AND SUN_HRS is Long AND RELHUMIDITY is Dry then the class of the Rainfallpercept is likely to be Heavy (162.0/3.0)

Rule 2:

If Max_Temp is Mild AND Wind_Speeds is Light Wind AND SUN_HRS is Short AND MONTH is September then the class of the Rainfallpercept is likely to be Light (28.0/2.0)

Rule 3:

If SUN_HRS is 'Long' AND MONTH is 'June' AND RELHUMIDITY is Dry then the class of the Rainfallpercept is likely to be 'Moderate' (117.0/41.0)

Rule 4:

If RELHUMIDITY is 'Very Dry' AND SUN_HRS is 'Long' AND Min_Temp is 'Very Cold' AND MONTH is 'January' then the class of the Rainfallpercept is likely to be 'No rain' (94.0)

Rule 5:

If RELHUMIDITY is 'Very Dry' AND Max_Temp is 'Hot' AND MONTH is 'March' AND Min_Temp is 'Cold' AND SUN_HRS is 'Long' then the class of the Rainfallpercept is likely to be 'No rain' (135.0/12.0)

Rule 6:

If SUN_HRS is 'Short' AND MONTH is 'July' AND Min_Temp is 'Cold' then the class of the Rainfallpercept is likely to be 'Moderate' (17.0/2.0)

Rule 7:

If SUN_HRS is ‘Medium” AND MONTH is ‘May’ AND Max_Temp is ‘Warm’ AND RELHUMIDITY is ‘Dry’ AND Wind_Speeds is ‘Light Wind’ then the class of the Rainfallpercept is likely to be ‘Heavy’ (203.0/10.0)

Rule 8:

If RELHUMIDITY is ‘Medium Wet’ AND MONTH is ‘May’ AND Max_Temp is ‘Warm’ then the class of the Rainfallpercept is likely to be ‘Heavy’ (540.0/315.0)

Rule 9:

If RELHUMIDITY is ‘Very Dry’ AND SUN_HRS is ‘Long’ AND Max_Temp is ‘Hot’ AND MONTH is ‘February’ AND Min_Temp is ‘Cold’ then the class of the Rainfallpercept is likely to be ‘No rain’ (119.0/16.0)

Rule 10:

If MONTH is ‘May’ AND Max_Temp is ‘Hot’ AND SUN_HRS is ‘Medium’ AND Min_Temp is ‘Cold’ then the class of the Rainfallpercept is likely to be ‘No rain’ (28.0/7.0)

Rule 11:

If Max_Temp is ‘Warm’ AND RELHUMIDITY is ‘Dry’ AND MONTH is ‘June’ AND Min_Temp is ‘Cold’ then the class of the Rainfallpercept is likely to be ‘No rain’ (60.0/25.0)

4.5. PERFORMANCE EVALUATION

To develop models that can predict rainfall J48, Multilayer Perceptron and PART rule algorithms were used. Thirteen experiments were done based on all features of the dataset in J48 and multilayer perceptron algorithm and eight experiments were done for part rule induction. To compare and evaluate the techniques which were used in this study, such as Decision tree, Multilayer perceptron and PART rule induction algorithm and to select the one, which performs the best. To evaluate and compare the performance of each of the model involved in this study, the standard metrics of accuracy, precision, recall, F-measure, True-positive and False-Positive Rates were used. Time taken by each classifier to build the selected models, and number of

instances which classified correctly and incorrectly were also other parameters used to compare and evaluate classifiers' performance.

The true positive and false positive rates, precision, recall and f-measure, values of each classifier for each class label were used for evaluation and also to evaluate the performance of Multilayer Perceptron algorithm by using different parameter like learning rate, momentum, training time and testing mode tuning on the performance of Multilayer perceptron and finally, to compare the performance of the algorithms in predicting the Rainfall status.

The results of the experiment performance for J48 decision tree, Multilayer perceptron and PART rule induction are summarized in Table 4.8, including their accuracy, recall, precision and area under the ROC.

Table 4.8 Performance summary of J48 and Neural Network

Parameter	Classifier		
	J48	**Multilayer Perceptron Neural Network**	**PART**
Accuracy	86.65%	80.03%	84.96%
Recall	86.70%	80.00%	85.00%
Precision	86.60%	80.20%	85.00%
ROC	95%	92.00%	92.50%
Correctly classfied instance	2000	1847	1961
Incorectly classified instance	308	461	347
Mean Absolute Error	0.079	0.1219	0.0902

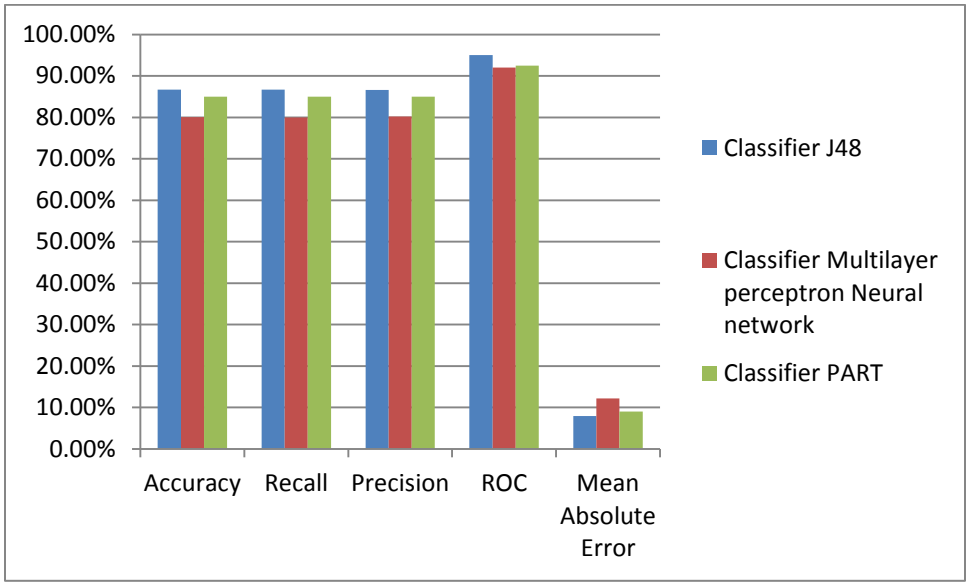


Figure 4.6 Experimental results of J48, Neural Network and PART

The main purpose of this research was defined by J48 decision tree algorithm, multilayer perceptron neural network and PART classifier model and to select the best performance. Accordingly, each experiment carried out in this study had employed J48 decision tree, multilayer perceptron neural network and PART classifier. In all experiments the same datasets were used. The output of these experiments indicated that the classification task of records using the meteorological dataset from National meteorological Agency of Ethiopia.

As shown figure 4.6 J48 decision tree has a better performance than the other classifiers in all the evaluation criteria's. The highest accuracy is found by the J48 decision tree method. Thus, it is considered also the base case. All the J48 decision tree algorithm tools tested have performed much better than the Multilayer perceptron and PART classifier method. Therefore, the overall scores of the J48 decision tree model have a better performance than that of the multilayer perceptron and PART classifier model.

In this study, the models were evaluated based on the accuracy measures discussed above (classification accuracy, recall, precision, Time taken for execution, ROC). The results were achieved using inputting 85% split test which is train a model and then supply the unseen remaining part of the record for testing the performance of the model. From the above studies,

the researcher found that the predictive model J48 decision tree achieved a classification accuracy of 86.65%, precision is 86.6%, and ROC is 95%.

4.6. INFORMATION GAIN

The information gain measure was used to select the test attribute at each node in the tree. Such a measure is referred to as an attribute selection measure or a measure of the goodness of split. The attribute with the highest information gain (or greatest entropy reduction) is chosen as the test attribute for the current node. This attribute minimizes the information needed to classify the samples in the resulting partitions and reflects the least randomness or "impurity" in these partitions. Such an information theoretic approach minimizes the expected number of tests needed to classify an object and guarantees that simple tree is found.

Since attribute selection is important in decision tree models, the researcher ranked the attributes based on information gain, Information gain measures the orders of the attributes computed using the formula. Ranking the attributes to the mining task of the decision tree was implemented by WEKA attribute ranking filter information gain.

Table 1.9 List of attribute with their information gain

No.	Ranked Attribute	Information Gain
1	1 MONTH	0.2395
2	4 RELHUMIDITY	0.1911
3	3 Min_Temp	0.1199
4	5 sun_hrs	0.113
5	2 Max_Temp	0.0862
6	6 Wind_Speeds	0.0349

As can be observed from the above Table 4.9, Month is the most determinant variable that contributes the occurrence of weather condition. And also maximum temperature, minimum temperature and relative humidity are greater impact on rainfall occurrences.

4.7. EXPERT JUDGMENT

The following discussion on the generated rules made with the meteorology experts from National Meteorological Agency. Some of the rules presents known pattern as the meteorology

experts opinion (the rules generated was agreed with meteorology experts). If relative humidity is very dry' and the sunshine hours are long then that leads to rainfall is likely to be No rain.

To add another known rule, if relative humidity is very dry and sunshine hours is medium and the maximum temperature is hot then that leads to rainfall is likely to be No rain. On the other hand, an interesting rule shows, if relative humidity is dry and the sun shine hours is medium and the minimum temperature is cold and month is January and wind speeds is light wind then that leads likely to be heavy rainfall.

Another interesting rule shows, if relative humidity is dry and the sunshine hours is medium and the minimum temperature is warm and the month is fall in May and the wind speeds is light wind then that leads to rainfall is likely to be heavy rain. On the other hand, If relative humidity is medium wet and the month falls in May and the wind speeds is light wind and the minimum temperature is mild and the sunshine hours is medium and the maximum temperature warm then that leads likely to be heavy rainfall.

Generally, it is possible to say that some of the rules obtained from the predictive model provide a pattern or knowledge and have got meaningful contribution for exploring the occurrence of the rainfall in National Meteorological Agency in Hawassa station and these findings also got acceptance by the meteorology expert. Consequently, to evaluate the significance of the above selected rules from the generated model and the attributes used to construct those rules, the relationship of the attributes with the predicted class as well as the model predicted by rules was evaluated based up on suggestions given by meteorology experts and these suggestions are if the researcher would incorporate different weather condition like altitude, latitude, mist, haize, dust, evaporation and pitche, it will enhance the performance of the model.

CHAPTER FIVE

5. CONCLUSION AND RECOMMENDATION

5.1. SUMMARY AND CONCLUSION

Meteorological data mining is a form of data mining concerned with finding hidden patterns inside largely available meteorological data, so that the information retrieved can be transformed into usable knowledge. Weather is one of the meteorological data that is rich in important knowledge. Data mining is extracting meaningful patterns and rules from large quantities of data. It is clearly useful in any field where there are large quantities of data and something worth learning. In this esteem, general use of meteorological information systems and explosive growth of weather databases require traditional manual data analysis to be coupled with methods for efficient computer-assisted analysis. Extensive amounts of weather data gathered in weather databases require specialized tools for storing and accessing data, for data analysis, and for effective use of data. Predictive modeling is the general concept of building a model that is capable of making predictions. Typically, such a model includes a machine learning algorithm that learns certain properties from a training dataset in order to make those predictions.

This study tries to investigate the potential applicability of data mining technology in developing a model to predict weather condition specifically rainfall prediction in Hawassa station of Ethiopian Meteorological Agency, so that it can support the control of unforeseen heavy rain in Ethiopia.

This study was conducted according to the CRISP-DM process model. The data was collected from Ethiopian Meteorological agency database organized from 2000 to 2014 for the research purpose. Analyzing the large volume of weather data and extracting useful information and knowledge for decision making about rainfall prediction was done. First the data was preprocessed for data cleaning, attribute and feature selection, and data transformation. This experimental research made use of three predictive modeling techniques, J48 decision tree, Multilayer perceptron Neural Network, and PART rule induction to address the problem. The experiment result shows that J48 decision tree has a higher performance rate than multilayer perceptron and PART rule induction classifiers.

J48 decision tree classifiers was achieved best performance by using pruned technique, confidence factor of 0.25, minimum numbers of instance (minNumObj) at 2, with all attributes data set, with a recall (true positive rate) of 86.67%, a false positive rate of 4.4%, a precision (positive predictive value) of 86.6%, and an accuracy of 86.65% prediction model building was selected to extract interesting rules to mention.

The other algorithm conducted in this study is Multilayer perceptron algorithms. The result show accuracy of 83.88% and correctly and incorrectly classified Instances are 2582 and 496 respectively and with recall (true positive rate) of 83.9%, false positive rate of 5.4%, a precision (positive predictive value) of 83.9 %, time taken to build a model is 3308.37 and ROC curve area 92.5.

In general, the results from this study were interesting and encouraging; it can be used as decision support for meteorology experts. The extracted rules in both algorithms are very effective for the prediction of rainfall. From these algorithms, we can observe that the attributes such as Month, Maximum temperature, Minimum temperature, sun hours and Relative humidity are the most determinant factors to predict rainfall.

5.2. RECOMMENDATION

This research work was carried out for academic purpose and is should be considered as a preliminary effort to give insight into the application of data mining technology for the specified area of the research problem. This research work can contribute a lot towards a comprehensive study in this area in the future. The results of this study have also shown that the data mining technology particularly the J48 decision tree classification technique are well applicable in the efforts of forecasting rainfall in Hawassa. Accordingly, based on the findings of this study, the researcher forwards the following recommendations that the following issues need to be addressed in future studies:

- The main objective of this study was to develop a model that can predict a rainfall in Hawassa region. The predictive model, which is developed in this research, generated

various patterns and rules. For the agency to use it effectively there is a need to design a knowledge base system, which can provide advice for the domain experts.

- All the experiments conducted using J48 decision tree, Neural Network and PART rule induction algorithms produced efficient models and interpretable rules. Hence it is important for meteorology to utilize the model developed with these data mining technique in order to use as a decision support tool in the identification of sever rainfall.
- Since this study has used a small percentage of the data which comprises only a 15 years data of weather in Hawassa station to build J48, Neural network and PART rule induction models, it is better to build more comprehensive models by using more additional data from different station.
- Although encouraging results were obtained from this study, particularly, using J48 decision tree classifier, there might be a probability to obtain more accurate and better performing results using other classification and prediction techniques which were not used by the researcher due to time constraint. Therefore, it is recommended that these classifiers should be applied and proved to this data.
- To sum up, the effective use of information and knowledge is vital for National Meteorology to stay competitive in today's complex environment. The challenges faced when trying to make large, diverse, and often complex data source are considerable. In an effort to turn information into knowledge, National Meteorological Agency should implement data mining technologies to predict an accurate and timely weather condition.

REFERENCE

[1] Sarah N.Kohail, Alaa M.El-Halees, "Implementation of data mining techniques for meteorological data analysis," *IJICT*, vol. 1, no. 3, July 2011.

[2] Mihaela O., "On the Use of Data-Mining Techniques in Knowledge-Based Systems," *Economy Informatics*, pp. 21-24, 2006.

[3] Tafesse Y., "Application of Artificial Neural Network in Weather Forecastiing," in *Addis Ababa University*, Addis Ababa, 2004.

[4] Pete Chapman, "The CRISP-DM user Guide," in *NCR System Engineering Copenhagen*, Brussels, 2009.

[5] Gonzalo M., Oscar M., and Covadonga F., "A survey of data mining and knowledge discovery process model and methodologies," *The knowledge Engineering Review*, vol. 25:2, pp. 137-166, 2010.

[6] Bartok J., Habala O., Bednar P., Gazak M., and Hluch L., "Data mining and integration for predicting significant meteorological phenomena," in *International Conference on Computational Science*, 2010, pp. 37-46.

[7] Ian H. Witten and Eibe Frank, *Data Mining Practical Machine learning Tools and Techniques*, 2nd ed. San Francisco, USA: Morgan Kaufmann Publisher, 2005.

[8] S.P.Deshpande and Dr. V.M. Thakare, "Data Mining System and Application: A review," *International Journal of Distributed and Parallel systems*, vol. 1, no. 1, pp. 32-44, September 2010.

[9] Hand D., Mannila H., and Smyth P., *Principle of Data Mining*. Massachusetts London, England: The MIT Press, 2001.

[10] Berry M., and Linoff G., *Data Mining Techniques for marketing, Sales and Customer relation management*, 2nd ed. Indianapolis, Indiana, Canada: Wiley Publishing, Inc., 2004.

[11] Cofioo A.S., Gutierrez J.M., Jakubiak B. And Melonek M., "Implementation of Data Mining Techniques for Meteorological Applications ," *Realizing Tera-computing, World Scientific*, pp. 215-240, 2003.

[12] Daniel Stojanov, Pance Panov, Andrej Kobler, Saoo Dueroski and Katerina Taokova. Learning to pridict forest fires with different data mining techniques. [Online]. http://ailab.ijs.si/dunja/SiKDD2006/Papers/Stojanova.pdf

[13] Godfrey C. Onwubolu, Petr Buryan, Sitaram Garimella, Visagaperuman Ramachandran, Viti Buadromo and Ajith Abraha, "Self-Organizing Data Mining For Weather Forecasting," in *IADIS*

European Conference Data Mining , 2007, pp. 81-88.

[14] Frawley W., Piatetsky-Shapiro G., and Matheus C., "Knowledge Discovery in Databases: An Overview," *AI Magazine*, vol. 13, no. 3, pp. 57-70, Fall 1992.

[15] Fayyad U., Piatetsky-Shapiro G., and Smyth P., "From Data Mining to Knowledge Discovery in Databases," *American Association for Artificial Intelligence*, pp. 37-54, Fall 1996.

[16] Maimon O. and Rokach L., *Data Mining and Knowledge Discovery handbook*, 2nd ed. Dordrecht Heidelberg London, England: Springer Science and Business Media, 2010.

[17] Clinton J., Kerber R., Khabaza T., and Reinartz T. Chapman P., "CRISP-DM 1.0 Step-by-step data mining guide," in *NCR Systems Engineering Copenhagen* , USA and Denmark, 2000.

[18] Furnkranz J., Gamberger D., and Lavrac N., *Foundation of Rule Learning*. Verlag Berlin Heidelberg : Springer, 2012.

[19] Alex G. Buchner, Maurice D. Mulvenna, Sarab S. Ananad, John G. Hughes. An Internet-Enabled Knowledge Discovery Process. [Online]. facweb.cs.depaul.edu/mobasher/classes/ect584/papers/buchner.pdf

[20] Cios K.J., Pedrycz W., Swiniarski R.W.,and Kurgan L., *Data Mining: A Knowledge Discovery Approach*.: Springer, 2007.

[21] Cabena P.,Hadjinian P., Stadler R.,Verhees J., and Zanasi A., *Discovering Data Mining:From Concept to Implementation*.: Prentice Hall, 1998.

[22] Lukasz A. Kurgan and Petr Musilek, "A Survey of Knowledge discovery and Data mining process models," *The knowledge Engineering Review*, vol. 21:1, pp. 1-24, 2006.

[23] Tom M.Mitchell. (2006, July) The Discipline of Machine Learning. Carnegie Mellon University.

[24] Jaime G. Carbonell, Ryszard S. Michalski and Tom M. Mitchell , "Machine Learning: A Historical and Methodological Analysis ," *AI Megazine*, vol. 4, no. 3, pp. 69-79, Fall 1983.

[25] Han J. and Kamber M., *Data Mining Concept and Techniques*, 2nd ed. San Francisco, USA: Morgan Kaufmann Publisher, 2006.

[26] [Online]. http://www.gartner.com/it-glossary/predictive-modeling.html

[27] Tan P., Steinbach M., and Kumar V., *Introduction to Data Mining*, 3rd ed. New Delhi, India: Pearson Education, 2009.

[28] David A. Dickey, "Introduction to predictive modeling with examples," in *SAS Global Forum*, North Carolina , 2012.

[29] Han J., Kamber M., and Pei J., *Data Mining Concept and techniques*, 3rd ed. Waltham, USA: Morgan Kaufmann Publisher, 2012.

[30] Thair Nu Phyu, "Survey of classification techniques in data mining," *International multi-conference of Engineers and computer scientists*, vol. 1, march 2009.

[31] Richard O. Duda, Peter E. Hart and David G. Stork. (1997, September) Pattern Classification.

[32] Two Crows Corporation. (2005) Introduction to data mining and Knowledge Discovery. [Online]. www.twocrows.com

[33] Tom M. Mitchell, *Machine Learning*.: McGraw Hill Science/Engineering/Math, March 1997.

[34] Bhargavi P. and Dr.Jyothi S., "Applying Naive Bayes Data Mining Technique for Classification of Agricultural Land Soils," *International Journal of Computer Science and Network Security*, vol. 9, no. 8, pp. 117-122, August 2009.

[35] Anil K. Jain and Richard C.Dubes, *Algorithm for Clustering data*. New Jersey, USA: Printice-Hall INC., 1988.

[36] Dr. Yashoal S. and Alok S, Chauhan, "Neural Networks in data Mining," *Journal of Theoretical and applied informaion technology*, pp. 37-42, 2005-2009.

[37] M. Hajek. (2005) Neural Network. [Online]. www.cs.ukzn.ac.za/notes/NeuralNetworks2005.pdf

[38] Priyanka Gaur, "Neural Network in Data Mining," *International Journal of Electronics and Computer science Engineering*, vol. 1, no. 3, pp. 1449-1453.

[39] Amrender Kumar. Artificial Neural Networks for Data Mining. [Online]. http://www.iasri.res.in/sscnars/data_mining/4-Artificial%20Neural%20Networks_Amrender.pdf

[40] Pavel Berkhin. (2002) Survey of Clustering data mining techniques. [Online]. www.cc.gatech.edu/~isbell/reading/papers/berkhin02survey.pdf

[41] A.K. Jain, M.N. Murty, and P.J. Flynn, "Data Clustering: A Review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264-323, September 1999.

[42] Weather Forecasting. [Online]. https://en.wikipedia.org/wiki/Weather_forecasting.html

[43] Imran Maqsood, Muhammad Riaz Khan, and Ajith Abraham, , "An ensemble of neural networks for

weather forecasting," *Springer-Verlag London Limited*, vol. 13, pp. 112-122, 2004.

[44] Kotsiantis S., Kostoulas A., Lykoudis S., Argiriou A., and Menagias K., "Using Data Mining Techniques for Estimating Minimum, Maximum and Average Daily Temperature Values," *International Journal of Mathematical, Physical and Engineering Sciences*, vol. 1, no. 1, pp. 16-20, 2008.

[45] Folorunsho Olaiya and Adesesan Barnabas Adeyemo, "Application of Data Mining Techniques in Weather prediction and Climate Change Studies," *International Journal of Information Engineering and Electronic Business*, vol. 1, pp. 51-59, February 2012.

[46] Meghali A. Kalyankar and Prof. S.J. Alaspurkar, "Data mining Techniques to Analyze the Meteorological data," *", International Journal of Advanced Research in Computer science and Software Engineering*, vol. 3, no. 2, pp. 114-118, February 2013.

[47] Pinky Saikia Dutta and Hitesh Tahbilder, "Prediction of rainfall using data mining techniques over Assama," *Indian Journal of Computer science and Engineering*, vol. 5, no. 2, pp. 85-90, Apr-May 2014.

[48] Tasha R. Inniss, "Seasonal Clustering techniques for time series data," *European Journal of Operational Research* , vol. 175, no. 1, pp. 376-384, 2006.

[49] Nkrintra Singhrattna, Balaji Rajagopalan, Martyn Clark and K. Krishna Kumar, "Seasonal Forecasting of Thailand summer monsoon rainfall," *International Journal of Climatology, Royal Meteorological Society*, vol. 25, pp. 649-664, 2005.

[50] Keon Tae Sohn, Jeong Hyeong Lee, Soon Hwan Lee, and Chan Su Ryu, "Statistical prediction of Heavy Rain in South Korea," *Advances in Atmospheric Science*, vol. 22, no. 5, pp. 703-710, 2005.

[51] Wint Thida Zaw and Thinn Thu Naing, "Empirical Statistical Modeling of Rainfall prediction over Myanmar," *World Academy of Science, Engineering and technology*, vol. 2, pp. 468-471, October 2008.

[52] Chakrabarti.S ,Earl C., Eibe F., Ralf H.G., Jaiwei H. , Xia J., Micheline K., Sam S.L.,Thomas P. ,Richard E. ,Dorian P., Mamdouh R.,Markus S.,Toby J. and Witten H., *Data mining know it all*. Burlington, USA: Morgan Kaufmann Publishers, 2009.

[53] Mark Hall and Geoffrey Holmes. (2002, April) Benchmarking Attribute Selection Techniques for Discrete Class Data Mining. [Online]. www.cs.waikato.ac.nz/~mhall/HallHolmesTKDE.pdf

[54] Colin Shearer, "The CRISP-DM Model:The New blue print for data mining," *Journal of Data Warehouse*, vol. 5, no. 4, pp. 13-22, Fall 2000.

[55] Thales Sehn Korting. C4.5 algorithm and Multivariate Decision Trees. [Online].

http://www.academia.edu/1983952/C4._5_algorithm_and_Multivariate_Decision_Trees

[56] Singh Y. and Chauhan S., "Neural Networks in Data Mining," *Journal of*, pp. 37-42, 2005.

[57] Nitesh V. Chawla, Aleksandar Lazarevic, Lawrence O. Hall and Kevin W. Bowyer, "SMOTEBoost: Improving Prediction of the Minority Class in Boosting," in *7th European Conference on Principles and Practice of Knowledge Discovery in Databases*, Dubrovnik, Croatia, 2003, pp. 107-119.

[58] Kohavi and provost, "The case against accuracy Estimation for comparingInduction algorithm," in *Proceedings of the fifteen International Conference on machine Learning*, Madison, 1998.

APPENDICES

APPENDIX A: J48 DECISION TREE OUTPUT

=== Run information ===

Scheme:weka.classifiers.trees.J48 -U -M 2

Relation: AwassaAug15-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-last-weka.filters.supervised.instance.SMOTE-C0-K5-P100.0-S1-weka.filters.supervised.instance.SMOTE-C0-K5-P100.0-S1-weka.filters.supervised.instance.SMOTE-C0-K5-P100.0-S1-weka.filters.supervised.instance.SMOTE-C0-K5-P100.0-S1-weka.filters.supervised.instance.SMOTE-C0-K5-P100.0-S1-weka.filters.supervised.instance.SMOTE-C0-K5-P100.0-S1-weka.filters.supervised.instance.SMOTE-C0-K5-P50.0-S1-weka.filters.supervised.instance.SMOTE-C0-K5-P50.0-S1-weka.filters.supervised.instance.SMOTE-C0-K5-P50.0-S1-weka.filters.supervised.instance.SMOTE-C0-K5-P20.0-S1-weka.filters.supervised.instance.SMOTE-C0-K5-P8.0-S1

Instances: 15389

Attributes: 9 YEAR, MONTH, DAY, Max_Temp, Min_Temp, RELHUMIDITY, sun_hrs, Wind_Speeds, RainfallPrecip

Test mode: split 85.0% train, remainder test

J48 unpruned tree

RELHUMIDITY = Very Dry

| sun_hrs = Long: No rain (885.0/31.0)

| sun_hrs = Medium

| | Max_Temp = Hot: No rain (48.0/2.0)

| | Max_Temp = Warm: Light (6.0/3.0)

| | Max_Temp = Mild: Light (2.0)

| sun_hrs = Short: Moderate (20.0/3.0)

RELHUMIDITY = Dry

| MONTH = January

| | sun_hrs = Long: No rain (412.0/112.0)

| | sun_hrs = Medium

| | | YEAR = 2000: Light (4.0/1.0)

| | | YEAR = 2001: No rain (19.0)

| | | YEAR = 2002

| | | | Max_Temp = Hot: Light (2.0)

| | | | Max_Temp = Warm: Moderate (12.0/2.0)

| | | | Max_Temp = Mild: Moderate (0.0)

| | | YEAR = 2003: No rain (13.0/5.0)

| | | YEAR = 2004: Light (2.0)

| | | YEAR = 2005: Heavy (172.0/4.0)

| | | YEAR = 2006

| | | | Min_Temp = Very Cold: No rain (1.0)

| | | | Min_Temp = Cold

| | | | | Wind_Speeds = Light Wind: Moderate (8.0/1.0)

| | | | | Wind_Speeds = Moderate Wind: No rain (2.0)

| | | | | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| | | | Min_Temp = Mild: No rain (2.0/1.0)

| | | YEAR = 2007: No rain (4.0/2.0)

| | | YEAR = 2008: Light (6.0/1.0)

| | | YEAR = 2009: No rain (7.0/1.0)

| | | YEAR = 2010: Moderate (27.0/2.0)

| | | YEAR = 2011: No rain (3.0)

| | | YEAR = 2012: Heavy (0.0)

| | sun_hrs = Short: Light (12.0/1.0)

| MONTH = February

| | YEAR = 2000

| | | sun_hrs = Long: No rain (8.0)

| | | sun_hrs = Medium: Light (2.0)

| | | sun_hrs = Short: No rain (0.0)

| | YEAR = 2001: Light (45.0/23.0)

| | YEAR = 2002

| | | Wind_Speeds = Light Wind: Light (29.0/2.0)

| | | Wind_Speeds = Moderate Wind: No rain (2.0)

| | | Wind_Speeds = Very Strong Wind: Light (0.0)

| | YEAR = 2003: No rain (26.0/11.0)

| | YEAR = 2004

| | | Min_Temp = Very Cold: No rain (11.0)

| | | Min_Temp = Cold: Moderate (33.0/10.0)

| | | Min_Temp = Mild: Moderate (0.0)

| | YEAR = 2005: No rain (13.0/2.0)

| | YEAR = 2006: No rain (28.0/2.0)

| | YEAR = 2007: No rain (40.0/18.0)

| | YEAR = 2008: No rain (19.0/2.0)

| | YEAR = 2009

| | | Wind_Speeds = Light Wind: Light (32.0/14.0)

| | | Wind_Speeds = Moderate Wind: No rain (9.0/1.0)

| | | Wind_Speeds = Very Strong Wind: No rain (0.0)

| | YEAR = 2010

| | | sun_hrs = Long: No rain (16.0/4.0)

| | | sun_hrs = Medium: Moderate (5.0/2.0)

| | | sun_hrs = Short: No rain (2.0/1.0)

| | YEAR = 2011

| | | sun_hrs = Long: No rain (18.0/3.0)

| | | sun_hrs = Medium: Light (6.0/1.0)

| | | sun_hrs = Short: No rain (0.0)

| | YEAR = 2012

| | | sun_hrs = Long: No rain (17.0/2.0)

| | | sun_hrs = Medium: Light (10.0/2.0)

| | | sun_hrs = Short: No rain (0.0)

| MONTH = March

| | YEAR = 2000

| | | sun_hrs = Long: Light (16.0/1.0)

| | | sun_hrs = Medium

| | | | Wind_Speeds = Light Wind: Moderate (9.0)

| | | | Wind_Speeds = Moderate Wind: No rain (2.0/1.0)

| | | | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| | | sun_hrs = Short: Light (0.0)

| | YEAR = 2001

| | | Max_Temp = Hot: Light (57.0/14.0)

| | | Max_Temp = Warm: Moderate (3.0/1.0)

| | | Max_Temp = Mild: Light (0.0)

| | YEAR = 2002

| | | Max_Temp = Hot

| | | | sun_hrs = Long: Heavy (96.0/38.0)

| | | | sun_hrs = Medium: Light (8.0/4.0)

| | | | sun_hrs = Short: Heavy (0.0)

| | | Max_Temp = Warm: Moderate (7.0)

| | | Max_Temp = Mild: Heavy (0.0)

| | YEAR = 2003

| | | sun_hrs = Long

| | | | Wind_Speeds = Light Wind: Moderate (33.0/16.0)

| | | | Wind_Speeds = Moderate Wind: Light (4.0)

| | | | Wind_Speeds = Very Strong Wind: Light (0.0)

| | | sun_hrs = Medium: Light (32.0/2.0)

| | | sun_hrs = Short: Light (0.0)

| | YEAR = 2004

| | | Wind_Speeds = Light Wind: No rain (21.0/7.0)

| | | Wind_Speeds = Moderate Wind: Light (7.0)

| | | Wind_Speeds = Very Strong Wind: No rain (0.0)

| | YEAR = 2005

| | | sun_hrs = Long: Light (31.0/13.0)

| | | sun_hrs = Medium

| | | | Min_Temp = Very Cold: Moderate (0.0)

| | | | Min_Temp = Cold: Moderate (13.0/4.0)

| | | | Min_Temp = Mild: No rain (2.0/1.0)

| | | sun_hrs = Short: No rain (1.0)

| | YEAR = 2006

| | | Max_Temp = Hot: Moderate (75.0/11.0)

| | | Max_Temp = Warm: No rain (6.0)

| | | Max_Temp = Mild: Moderate (0.0)

| | YEAR = 2007

| | | sun_hrs = Long: Moderate (36.0/6.0)

| | | sun_hrs = Medium: Light (11.0/2.0)

| | | sun_hrs = Short: Light (1.0)

| | YEAR = 2008: Light (8.0/3.0)

| | YEAR = 2009

| | | sun_hrs = Long: Light (43.0/21.0)

| | | sun_hrs = Medium: No rain (3.0)

| | | sun_hrs = Short: Light (0.0)

| | YEAR = 2010

| | | sun_hrs = Long: No rain (15.0/3.0)

| | | sun_hrs = Medium

| | | | Max_Temp = Hot: No rain (3.0/1.0)

| | | | Max_Temp = Warm: Moderate (32.0/5.0)

| | | | Max_Temp = Mild: Moderate (0.0)

| | | sun_hrs = Short: Moderate (0.0)

| | YEAR = 2011

| | | sun_hrs = Long: Heavy (86.0/28.0)

| | | sun_hrs = Medium

| | | | Min_Temp = Very Cold: No rain (0.0)

| | | | Min_Temp = Cold: No rain (2.0)

| | | | Min_Temp = Mild: Light (2.0)

| | | sun_hrs = Short: Light (1.0)

| | YEAR = 2012: Light (20.0/10.0)

| MONTH = April

| | sun_hrs = Long

| | | YEAR = 2000

| | | | Wind_Speeds = Light Wind: Light (10.0/5.0)

| | | | Wind_Speeds = Moderate Wind: No rain (2.0)

| | | | Wind_Speeds = Very Strong Wind: No rain (0.0)

| | | YEAR = 2001: No rain (26.0/9.0)

| | | YEAR = 2002

| | | | Max_Temp = Hot: Moderate (41.0/15.0)

| | | | Max_Temp = Warm: Heavy (158.0/1.0)

| | | | Max_Temp = Mild: Heavy (0.0)

| | | YEAR = 2003: No rain (8.0/4.0)

| | | YEAR = 2004

| | | | Wind_Speeds = Light Wind: Light (19.0/4.0)

| | | | Wind_Speeds = Moderate Wind: No rain (2.0)

| | | | Wind_Speeds = Very Strong Wind: Light (0.0)

| | | YEAR = 2005: No rain (13.0/5.0)

| | | YEAR = 2006: No rain (9.0/1.0)

| | | YEAR = 2007: Light (43.0/6.0)

| | | YEAR = 2008: No rain (11.0/5.0)

| | | YEAR = 2009: No rain (24.0/10.0)

| | | YEAR = 2010: No rain (16.0/7.0)

| | | YEAR = 2011: Heavy (263.0/29.0)

| | | YEAR = 2012: Heavy (60.0/7.0)

| | sun_hrs = Medium

| | | YEAR = 2000

| | | | Max_Temp = Hot: Moderate (55.0/15.0)

| | | | Max_Temp = Warm: Light (8.0/2.0)

| | | | Max_Temp = Mild: Moderate (0.0)

| | | YEAR = 2001: Light (9.0/1.0)

| | | YEAR = 2002

| | | | Max_Temp = Hot

| | | | | Min_Temp = Very Cold: Moderate (0.0)

| | | | | Min_Temp = Cold: Moderate (30.0/12.0)

| | | | | Min_Temp = Mild: Light (7.0)

| | | | Max_Temp = Warm: No rain (4.0)

| | | | Max_Temp = Mild: Moderate (0.0)

| | | YEAR = 2003: No rain (1.0)

| | | YEAR = 2004: No rain (5.0/1.0)

| | | YEAR = 2005: Light (2.0/1.0)

| | | YEAR = 2006: No rain (4.0/2.0)

| | | YEAR = 2007: Light (7.0)

| | | YEAR = 2008

| | | | Wind_Speeds = Light Wind: Light (19.0/9.0)

| | | | Wind_Speeds = Moderate Wind

| | | | | Min_Temp = Very Cold: No rain (0.0)

| | | | | Min_Temp = Cold: No rain (9.0/1.0)

| | | | | Min_Temp = Mild: Light (2.0/1.0)

| | | | Wind_Speeds = Very Strong Wind: No rain (0.0)

| | | YEAR = 2009: Light (22.0/5.0)

| | | YEAR = 2010: Light (5.0/2.0)

| | | YEAR = 2011

| | | | Max_Temp = Hot: Light (31.0/16.0)

| | | | Max_Temp = Warm: No rain (2.0)

| | | | Max_Temp = Mild: Light (0.0)

| | | YEAR = 2012

| | | | Min_Temp = Very Cold: Moderate (0.0)

| | | | Min_Temp = Cold

| | | | | Max_Temp = Hot: Moderate (21.0/8.0)

| | | | | Max_Temp = Warm: No rain (2.0/1.0)

| | | | | Max_Temp = Mild: Moderate (0.0)

| | | | Min_Temp = Mild: Light (5.0/1.0)

| | sun_hrs = Short

| | | YEAR = 2000: Moderate (2.0)

| | | YEAR = 2001: No rain (2.0/1.0)

| | | YEAR = 2002: No rain (1.0)

| | | YEAR = 2003: No rain (0.0)

| | | YEAR = 2004: No rain (0.0)

| | | YEAR = 2005: No rain (0.0)

| | | YEAR = 2006: No rain (1.0)

| | | YEAR = 2007: Light (1.0)

| | | YEAR = 2008: Light (2.0)

| | | YEAR = 2009: No rain (1.0)

| | | YEAR = 2010: No rain (0.0)

| | | YEAR = 2011: No rain (2.0)

| | | YEAR = 2012: Heavy (2.0)

| MONTH = May

| | sun_hrs = Long

| | | YEAR = 2000

| | | | Wind_Speeds = Light Wind: Light (32.0/7.0)

| | | | Wind_Speeds = Moderate Wind: No rain (2.0)

| | | | Wind_Speeds = Very Strong Wind: Light (0.0)

| | | YEAR = 2001: Light (14.0/5.0)

| | | YEAR = 2002: No rain (11.0/3.0)

| | | YEAR = 2003: No rain (19.0/4.0)

| | | YEAR = 2004

| | | | Max_Temp = Hot: No rain (22.0/11.0)

| | | | Max_Temp = Warm: Moderate (13.0/4.0)

| | | | Max_Temp = Mild: No rain (0.0)

| | | YEAR = 2005: No rain (1.0)

| | | YEAR = 2006: No rain (15.0/5.0)

| | | YEAR = 2007: No rain (9.0/3.0)

| | | YEAR = 2008

| | | | Max_Temp = Hot: Moderate (13.0/2.0)

| | | | Max_Temp = Warm

| | | | | Wind_Speeds = Light Wind: Light (3.0)

| | | | | Wind_Speeds = Moderate Wind: No rain (2.0)

| | | | | Wind_Speeds = Very Strong Wind: Light (0.0)

| | | | Max_Temp = Mild: Moderate (0.0)

| | | YEAR = 2009: No rain (22.0/4.0)

| | | YEAR = 2010: Moderate (1.0)

| | | YEAR = 2011: Light (5.0/2.0)

| | | YEAR = 2012: No rain (16.0/1.0)

| | sun_hrs = Medium

| | | Max_Temp = Hot: No rain (42.0/7.0)

| | | Max_Temp = Warm

| | | | YEAR = 2000: No rain (1.0)

| | | | YEAR = 2001

| | | | | Wind_Speeds = Light Wind: No rain (3.0/1.0)

| | | | | Wind_Speeds = Moderate Wind: Moderate (2.0)

| | | | | Wind_Speeds = Very Strong Wind: No rain (0.0)

| | | | YEAR = 2002: Moderate (5.0)

| | | | YEAR = 2003: No rain (2.0/1.0)

| | | | YEAR = 2004: Heavy (0.0)

| | | | YEAR = 2005: Heavy (0.0)

| | | | YEAR = 2006: Light (2.0/1.0)

| | | | YEAR = 2007: Heavy (0.0)

| | | | YEAR = 2008

| | | | | Wind_Speeds = Light Wind: Light (2.0)

| | | | | Wind_Speeds = Moderate Wind: No rain (2.0)

| | | | | Wind_Speeds = Very Strong Wind: No rain (0.0)

| | | | YEAR = 2009: Heavy (0.0)

| | | | YEAR = 2010: Heavy (192.0)

| | | | YEAR = 2011: No rain (1.0)

| | | | YEAR = 2012: No rain (1.0)

| | | Max_Temp = Mild: Heavy (0.0)

| | sun_hrs = Short: No rain (4.0/2.0)

| MONTH = June

| | Min_Temp = Very Cold: Light (11.0/2.0)

| | Min_Temp = Cold

| | | YEAR = 2000

| | | | sun_hrs = Long

| | | | | Wind_Speeds = Light Wind: Moderate (15.0/1.0)

| | | | | Wind_Speeds = Moderate Wind: No rain (11.0/4.0)

| | | | | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| | | | sun_hrs = Medium: No rain (15.0/4.0)

| | | | sun_hrs = Short: Moderate (0.0)

| | | YEAR = 2001

| | | | sun_hrs = Long: Moderate (29.0/3.0)

| | | | sun_hrs = Medium

| | | | | Wind_Speeds = Light Wind: Light (12.0/2.0)

| | | | | Wind_Speeds = Moderate Wind: Moderate (8.0/3.0)

| | | | | Wind_Speeds = Very Strong Wind: Light (0.0)

| | | | sun_hrs = Short: Moderate (0.0)

| | | YEAR = 2002

| | | | sun_hrs = Long

| | | | | Wind_Speeds = Light Wind: No rain (7.0/3.0)

| | | | | Wind_Speeds = Moderate Wind: Light (12.0)

| | | | | Wind_Speeds = Very Strong Wind: Light (0.0)

| | | | sun_hrs = Medium: Moderate (19.0/8.0)

| | | | sun_hrs = Short: Light (0.0)

| | | YEAR = 2003

| | | | Wind_Speeds = Light Wind

| | | | | sun_hrs = Long: Moderate (26.0/10.0)

| | | | | sun_hrs = Medium: Light (2.0)

| | | | | sun_hrs = Short: Moderate (0.0)

| | | | Wind_Speeds = Moderate Wind: Light (9.0)

| | | | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| | | YEAR = 2004

| | | | sun_hrs = Long

| | | | | Wind_Speeds = Light Wind: Moderate (23.0/1.0)

| | | | | Wind_Speeds = Moderate Wind: No rain (4.0/2.0)

| | | | | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| | | | sun_hrs = Medium

| | | | | Wind_Speeds = Light Wind: No rain (11.0/2.0)

| | | | | Wind_Speeds = Moderate Wind: Light (17.0/7.0)

| | | | | Wind_Speeds = Very Strong Wind: No rain (0.0)

| | | | sun_hrs = Short: Moderate (0.0)

| | | YEAR = 2005: No rain (13.0/4.0)

| | | YEAR = 2006: No rain (14.0/6.0)

| | | YEAR = 2007: Light (4.0/2.0)

| | | YEAR = 2008

| | | | sun_hrs = Long: No rain (6.0/1.0)

| | | | sun_hrs = Medium: Moderate (7.0/3.0)

| | | | sun_hrs = Short: No rain (0.0)

| | | YEAR = 2009

| | | | sun_hrs = Long: Light (24.0/8.0)

| | | | sun_hrs = Medium: No rain (4.0/1.0)

| | | | sun_hrs = Short: Light (3.0)

| | | YEAR = 2010: No rain (7.0/3.0)

| | | YEAR = 2011: Moderate (0.0)

| | | YEAR = 2012: No rain (14.0/2.0)

| | Min_Temp = Mild

| | | Wind_Speeds = Light Wind: Moderate (47.0/7.0)

| | | Wind_Speeds = Moderate Wind: Light (10.0/4.0)

| | | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| MONTH = July

| | Wind_Speeds = Light Wind

| | | YEAR = 2000: Light (8.0/2.0)

| | | YEAR = 2001

| | | | sun_hrs = Long: Moderate (13.0/1.0)

| | | | sun_hrs = Medium: Light (5.0)

| | | | sun_hrs = Short: Moderate (0.0)

| | | YEAR = 2002

| | | | sun_hrs = Long: No rain (4.0)

| | | | sun_hrs = Medium: Light (21.0/6.0)

| | | | sun_hrs = Short: No rain (1.0)

| | | YEAR = 2003: Moderate (9.0/1.0)

| | | YEAR = 2004: Moderate (43.0/10.0)

| | | YEAR = 2005

| | | | sun_hrs = Long: Moderate (4.0/1.0)

| | | | sun_hrs = Medium: No rain (5.0/2.0)

| | | | sun_hrs = Short: No rain (0.0)

| | | YEAR = 2006: Moderate (7.0/3.0)

| | | YEAR = 2007: No rain (2.0/1.0)

| | | YEAR = 2008: Light (1.0)

| | | YEAR = 2009

| | | | sun_hrs = Long: No rain (5.0/1.0)

| | | | sun_hrs = Medium: Light (5.0/2.0)

| | | | sun_hrs = Short: Light (1.0)

| | | YEAR = 2010

| | | | sun_hrs = Long: Moderate (0.0)

| | | | sun_hrs = Medium: No rain (4.0/2.0)

| | | | sun_hrs = Short: Moderate (3.0)

| | | YEAR = 2011: Light (1.0)

| | | YEAR = 2012: Moderate (0.0)

| | Wind_Speeds = Moderate Wind

| | | YEAR = 2000: Light (11.0/3.0)

| | | YEAR = 2001: Light (5.0/2.0)

| | | YEAR = 2002

| | | | sun_hrs = Long: No rain (11.0/3.0)

| | | | sun_hrs = Medium: Light (33.0/5.0)

| | | | sun_hrs = Short: Light (0.0)

| | | YEAR = 2003: Light (1.0)

| | | YEAR = 2004

| | | | sun_hrs = Long: Light (0.0)

| | | | sun_hrs = Medium: Light (19.0/6.0)

| | | | sun_hrs = Short: No rain (2.0)

| | | YEAR = 2005: No rain (5.0/2.0)

| | | YEAR = 2006: No rain (3.0/1.0)

| | | YEAR = 2007: No rain (3.0/1.0)

| | | YEAR = 2008: No rain (6.0/1.0)

| | | YEAR = 2009: No rain (13.0/1.0)

| | | YEAR = 2010: No rain (3.0)

| | | YEAR = 2011: Light (0.0)

| | | YEAR = 2012: Light (0.0)

| | Wind_Speeds = Very Strong Wind: No rain (1.0)

| MONTH = August

| | Wind_Speeds = Light Wind

| | | YEAR = 2000: No rain (11.0/5.0)

| | | YEAR = 2001

| | | | sun_hrs = Long: Light (37.0/18.0)

| | | | sun_hrs = Medium: Moderate (14.0/4.0)

| | | | sun_hrs = Short: Moderate (0.0)

| | | YEAR = 2002

| | | | sun_hrs = Long: Heavy (6.0/3.0)

| | | | sun_hrs = Medium: No rain (9.0/3.0)

| | | | sun_hrs = Short: No rain (0.0)

| | | YEAR = 2003

| | | | sun_hrs = Long: Moderate (8.0/2.0)

| | | | sun_hrs = Medium: Light (2.0)

| | | | sun_hrs = Short: Moderate (0.0)

| | | YEAR = 2004

| | | | sun_hrs = Long: No rain (5.0/1.0)

| | | | sun_hrs = Medium: Moderate (30.0/12.0)

| | | | sun_hrs = Short: Moderate (8.0/1.0)

| | | YEAR = 2005: No rain (11.0/6.0)

| | | YEAR = 2006: Moderate (1.0)

| | | YEAR = 2007: Moderate (0.0)

| | | YEAR = 2008: No rain (2.0/1.0)

| | | YEAR = 2009

| | | | sun_hrs = Long: Light (4.0/1.0)

| | | | sun_hrs = Medium

| | | | | Min_Temp = Very Cold: No rain (0.0)

| | | | | Min_Temp = Cold: No rain (4.0/2.0)

| | | | | Min_Temp = Mild: Moderate (3.0/1.0)

| | | | sun_hrs = Short: Moderate (6.0)

| | | YEAR = 2010: Heavy (3.0/1.0)

| | | YEAR = 2011: No rain (3.0)

| | | YEAR = 2012: Moderate (5.0/2.0)

| | Wind_Speeds = Moderate Wind

| | | YEAR = 2000: Light (9.0/3.0)

| | | YEAR = 2001: No rain (0.0)

| | | YEAR = 2002: Light (7.0/1.0)

| | | YEAR = 2003: No rain (1.0)

| | | YEAR = 2004: No rain (3.0)

| | | YEAR = 2005: Light (9.0/4.0)

| | | YEAR = 2006: No rain (1.0)

| | | YEAR = 2007: No rain (0.0)

| | | YEAR = 2008: No rain (2.0/1.0)

| | | YEAR = 2009

| | | | Min_Temp = Very Cold: No rain (0.0)

| | | | Min_Temp = Cold: No rain (9.0/2.0)

| | | | Min_Temp = Mild: Light (2.0)

| | | YEAR = 2010: No rain (2.0/1.0)

| | | YEAR = 2011: No rain (1.0)

| | | YEAR = 2012: No rain (0.0)

| | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| MONTH = Septmeber

| | YEAR = 2000: No rain (3.0/1.0)

| | YEAR = 2001: Light (9.0/2.0)

| | YEAR = 2002: No rain (18.0/7.0)

| | YEAR = 2003

| | | sun_hrs = Long: No rain (6.0/3.0)

| | | sun_hrs = Medium: Light (16.0/1.0)

| | | sun_hrs = Short: Light (0.0)

| | YEAR = 2004: Light (4.0/2.0)

| | YEAR = 2005: No rain (5.0/2.0)

| | YEAR = 2006: Moderate (7.0/2.0)

| | YEAR = 2007: Light (0.0)

| | YEAR = 2008: Light (14.0)

| | YEAR = 2009

| | | sun_hrs = Long: No rain (4.0)

| | | sun_hrs = Medium

| | | | Min_Temp = Very Cold: No rain (0.0)

| | | | Min_Temp = Cold: No rain (7.0/3.0)

| | | | Min_Temp = Mild: Moderate (2.0)

| | | sun_hrs = Short: Moderate (4.0)

| | YEAR = 2010: Heavy (53.0/2.0)

| | YEAR = 2011: Light (1.0)

| | YEAR = 2012: No rain (1.0)

| MONTH = October

| | YEAR = 2000

| | | sun_hrs = Long: No rain (3.0)

| | | sun_hrs = Medium: Light (5.0/2.0)

| | | sun_hrs = Short: Light (1.0)

| | YEAR = 2001

| | | sun_hrs = Long: No rain (19.0/5.0)

| | | sun_hrs = Medium: Light (7.0/1.0)

| | | sun_hrs = Short: No rain (0.0)

| | YEAR = 2002

| | | sun_hrs = Long

| | | | Min_Temp = Very Cold: No rain (3.0)

| | | | Min_Temp = Cold: Light (26.0/11.0)

| | | | Min_Temp = Mild: Light (0.0)

| | | sun_hrs = Medium

| | | | Min_Temp = Very Cold: Light (3.0/1.0)

| | | | Min_Temp = Cold: No rain (19.0/3.0)

| | | | Min_Temp = Mild: No rain (0.0)

| | | sun_hrs = Short: No rain (2.0)

| | YEAR = 2003

| | | Max_Temp = Hot: No rain (32.0/10.0)

| | | Max_Temp = Warm: Light (12.0/1.0)

| | | Max_Temp = Mild: No rain (0.0)

| | YEAR = 2004: No rain (19.0)

| | YEAR = 2005: No rain (31.0/5.0)

| | YEAR = 2006: No rain (9.0/2.0)

| | YEAR = 2007: No rain (24.0/2.0)

| | YEAR = 2008: No rain (22.0/5.0)

| | YEAR = 2009

| | | Min_Temp = Very Cold: No rain (3.0)

| | | Min_Temp = Cold

| | | | Max_Temp = Hot: Light (21.0/9.0)

| | | | Max_Temp = Warm

| | | | | sun_hrs = Long: No rain (6.0)

| | | | | sun_hrs = Medium: Light (3.0)

| | | | | sun_hrs = Short: No rain (0.0)

| | | | Max_Temp = Mild: Light (0.0)

| | | Min_Temp = Mild: No rain (1.0)

| | YEAR = 2010: No rain (29.0/3.0)

| | YEAR = 2011: No rain (31.0)

| | YEAR = 2012

| | | sun_hrs = Long: No rain (29.0/6.0)

| | | sun_hrs = Medium: Light (5.0/1.0)

| | | sun_hrs = Short: No rain (2.0/1.0)

| MONTH = November

| | Min_Temp = Very Cold: No rain (161.0/1.0)

| | Min_Temp = Cold

| | | YEAR = 2000: No rain (19.0/2.0)

| | | YEAR = 2001

| | | | sun_hrs = Long: No rain (8.0/2.0)

| | | | sun_hrs = Medium: Light (15.0/4.0)

| | | | sun_hrs = Short: Light (1.0)

| | | YEAR = 2002: No rain (7.0)

| | | YEAR = 2003: No rain (26.0/10.0)

| | | YEAR = 2004

| | | | Max_Temp = Hot: No rain (18.0)

| | | | Max_Temp = Warm

| | | | | sun_hrs = Long: No rain (4.0)

| | | | | sun_hrs = Medium: Moderate (41.0/5.0)

| | | | | sun_hrs = Short: Moderate (0.0)

| | | | Max_Temp = Mild: Moderate (0.0)

| | | YEAR = 2005: No rain (5.0/1.0)

| | | YEAR = 2006: No rain (30.0/3.0)

| | | YEAR = 2007: No rain (25.0/6.0)

| | | YEAR = 2008: No rain (13.0/1.0)

| | | YEAR = 2009: No rain (6.0/1.0)

| | | YEAR = 2010: No rain (37.0/18.0)

| | | YEAR = 2011

| | | | Max_Temp = Hot: Moderate (15.0/5.0)

| | | | Max_Temp = Warm: No rain (7.0/1.0)

| | | | Max_Temp = Mild: Moderate (0.0)

| | | YEAR = 2012

| | | | Max_Temp = Hot: Light (39.0/16.0)

| | | | Max_Temp = Warm: No rain (2.0)

| | | | Max_Temp = Mild: Light (0.0)

| | Min_Temp = Mild

| | | Max_Temp = Hot: Light (2.0/1.0)

| | | Max_Temp = Warm: No rain (2.0)

| | | Max_Temp = Mild: No rain (0.0)

| MONTH = December

| | Min_Temp = Very Cold: No rain (161.0/3.0)

| | Min_Temp = Cold

| | | YEAR = 2000

| | | | sun_hrs = Long: No rain (15.0)

| | | | sun_hrs = Medium: Light (3.0/1.0)

| | | | sun_hrs = Short: No rain (0.0)

| | | YEAR = 2001: No rain (21.0/5.0)

| | | YEAR = 2002

| | | | Max_Temp = Hot: No rain (21.0/1.0)

| | | | Max_Temp = Warm: Light (6.0/2.0)

| | | | Max_Temp = Mild: No rain (0.0)

| | | YEAR = 2003: No rain (17.0/3.0)

| | | YEAR = 2004: No rain (34.0/7.0)

| | | YEAR = 2005: No rain (4.0)

| | | YEAR = 2006: No rain (29.0/2.0)

| | | YEAR = 2007: No rain (10.0)

| | | YEAR = 2008: No rain (20.0/1.0)

| | | YEAR = 2009

| | | | Max_Temp = Hot: No rain (15.0/2.0)

| | | | Max_Temp = Warm: Light (9.0/3.0)

| | | | Max_Temp = Mild: No rain (0.0)

| | | YEAR = 2010

| | | | sun_hrs = Long

| | | | | Max_Temp = Hot: Moderate (30.0/7.0)

| | | | | Max_Temp = Warm: No rain (4.0)

| | | | | Max_Temp = Mild: Moderate (0.0)

| | | | sun_hrs = Medium: No rain (8.0)

| | | | sun_hrs = Short: Moderate (0.0)

| | | YEAR = 2011: No rain (10.0)

| | | YEAR = 2012: No rain (25.0/2.0)

| | Min_Temp = Mild: Light (5.0/3.0)

RELHUMIDITY = Medium Wet

| MONTH = January

| | Min_Temp = Very Cold: Light (0.0)

| | Min_Temp = Cold

| | | Max_Temp = Hot: Moderate (12.0/3.0)

| | | Max_Temp = Warm

| | | | Wind_Speeds = Light Wind: Light (45.0/16.0)

| | | | Wind_Speeds = Moderate Wind: No rain (2.0/1.0)

| | | | Wind_Speeds = Very Strong Wind: Light (0.0)

| | | Max_Temp = Mild: Light (0.0)

| | Min_Temp = Mild: Moderate (16.0/2.0)

| MONTH = February

| | Min_Temp = Very Cold: Moderate (0.0)

| | Min_Temp = Cold: Moderate (90.0/28.0)

| | Min_Temp = Mild

| | | Max_Temp = Hot: Light (2.0)

| | | Max_Temp = Warm

| | | | sun_hrs = Long: No rain (0.0)

| | | | sun_hrs = Medium: Light (6.0/3.0)

| | | | sun_hrs = Short: No rain (3.0/1.0)

| | | Max_Temp = Mild: Light (0.0)

| MONTH = March

| | Min_Temp = Very Cold: Moderate (0.0)

| | Min_Temp = Cold

| | | sun_hrs = Long

| | | | Wind_Speeds = Light Wind: Heavy (57.0/32.0)

| | | | Wind_Speeds = Moderate Wind: Moderate (2.0)

| | | | Wind_Speeds = Very Strong Wind: Heavy (0.0)

| | | sun_hrs = Medium: Moderate (139.0/58.0)

| | | sun_hrs = Short

| | | | Max_Temp = Hot: Light (3.0/1.0)

| | | | Max_Temp = Warm: Moderate (69.0/14.0)

| | | | Max_Temp = Mild: Moderate (0.0)

| | Min_Temp = Mild

| | | Wind_Speeds = Light Wind: Heavy (72.0/9.0)

| | | Wind_Speeds = Moderate Wind: Moderate (3.0/1.0)

| | | Wind_Speeds = Very Strong Wind: Heavy (0.0)

| MONTH = April

| | Min_Temp = Very Cold: Moderate (0.0)

| | Min_Temp = Cold

| | | Max_Temp = Hot

| | | | sun_hrs = Long: Light (21.0/1.0)

| | | | sun_hrs = Medium

| | | | | Wind_Speeds = Light Wind: Moderate (80.0/36.0)

| | | | | Wind_Speeds = Moderate Wind: No rain (3.0/2.0)

| | | | | Wind_Speeds = Very Strong Wind: Light (1.0)

| | | | sun_hrs = Short: Light (2.0/1.0)

| | | Max_Temp = Warm

| | | | sun_hrs = Long: Moderate (41.0/7.0)

| | | | sun_hrs = Medium

| | | | | Wind_Speeds = Light Wind: Heavy (456.0/230.0)

| | | | | Wind_Speeds = Moderate Wind: No rain (5.0/3.0)

| | | | | Wind_Speeds = Very Strong Wind: Heavy (0.0)

| | | | sun_hrs = Short: Light (51.0/28.0)

| | | Max_Temp = Mild: Heavy (0.0)

| | Min_Temp = Mild

| | | sun_hrs = Long: Moderate (17.0/1.0)

| | | sun_hrs = Medium

| | | | Max_Temp = Hot: Light (8.0/2.0)

| | | | Max_Temp = Warm

| | | | | Wind_Speeds = Light Wind: Moderate (88.0/36.0)

| | | | | Wind_Speeds = Moderate Wind: No rain (2.0/1.0)

| | | | | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| | | | Max_Temp = Mild: Moderate (0.0)

| | | sun_hrs = Short: Light (46.0/12.0)

| MONTH = May

| | Wind_Speeds = Light Wind

| | | Min_Temp = Very Cold: Heavy (0.0)

| | | Min_Temp = Cold

| | | | sun_hrs = Long: Heavy (534.0/191.0)

| | | | sun_hrs = Medium

| | | | | Max_Temp = Hot: Moderate (54.0/21.0)

| | | | | Max_Temp = Warm: Heavy (540.0/315.0)

| | | | | Max_Temp = Mild: Heavy (0.0)

| | | | sun_hrs = Short: Moderate (64.0/26.0)

| | | Min_Temp = Mild

| | | | sun_hrs = Long

| | | | | Max_Temp = Hot: No rain (2.0/1.0)

| | | | | Max_Temp = Warm: Moderate (13.0/3.0)

| | | | | Max_Temp = Mild: Moderate (0.0)

| | | | sun_hrs = Medium

| | | | | Max_Temp = Hot: Light (11.0)

| | | | | Max_Temp = Warm: Heavy (242.0/43.0)

| | | | | Max_Temp = Mild: Heavy (0.0)

| | | | sun_hrs = Short

| | | | | Max_Temp = Hot: Light (2.0)

| | | | | Max_Temp = Warm: Heavy (263.0/43.0)

| | | | | Max_Temp = Mild: Heavy (0.0)

| | Wind_Speeds = Moderate Wind

| | | Min_Temp = Very Cold: Light (0.0)

| | | Min_Temp = Cold: Light (53.0/26.0)

| | | Min_Temp = Mild

| | | | sun_hrs = Long: Moderate (11.0/1.0)

| | | | sun_hrs = Medium: Light (21.0/11.0)

| | | | sun_hrs = Short: Moderate (2.0)

| | Wind_Speeds = Very Strong Wind: Heavy (0.0)

| MONTH = June

| | Wind_Speeds = Light Wind

| | | sun_hrs = Long

| | | | Max_Temp = Hot: No rain (2.0/1.0)

| | | | Max_Temp = Warm: Moderate (68.0/36.0)

| | | | Max_Temp = Mild: Moderate (0.0)

| | | sun_hrs = Medium

| | | | Min_Temp = Very Cold: Heavy (0.0)

| | | | Min_Temp = Cold

| | | | | Max_Temp = Hot: No rain (2.0/1.0)

| | | | | Max_Temp = Warm: Heavy (289.0/170.0)

| | | | | Max_Temp = Mild: Heavy (0.0)

| | | | Min_Temp = Mild: Light (29.0/14.0)

| | | sun_hrs = Short

| | | | Min_Temp = Very Cold: Heavy (0.0)

| | | | Min_Temp = Cold: Heavy (278.0/124.0)

| | | | Min_Temp = Mild: No rain (10.0/5.0)

| | Wind_Speeds = Moderate Wind

| | | sun_hrs = Long

| | | | Min_Temp = Very Cold: Light (0.0)

| | | | Min_Temp = Cold: Light (50.0/28.0)

| | | | Min_Temp = Mild: No rain (4.0/1.0)

| | | sun_hrs = Medium

| | | | Min_Temp = Very Cold: Moderate (0.0)

| | | | Min_Temp = Cold: Moderate (208.0/123.0)

| | | | Min_Temp = Mild: No rain (18.0/10.0)

| | | sun_hrs = Short

| | | | Min_Temp = Very Cold: Light (0.0)

| | | | Min_Temp = Cold: Light (23.0/9.0)

| | | | Min_Temp = Mild: Moderate (14.0/4.0)

| | Wind_Speeds = Very Strong Wind: Heavy (0.0)

| MONTH = July

| | Wind_Speeds = Light Wind

| | | Max_Temp = Hot: Light (1.0)

| | | Max_Temp = Warm

| | | | sun_hrs = Long: No rain (20.0/10.0)

| | | | sun_hrs = Medium: Heavy (620.0/341.0)

| | | | sun_hrs = Short

| | | | | Min_Temp = Very Cold: Moderate (1.0)

| | | | | Min_Temp = Cold: Heavy (551.0/269.0)

| | | | | Min_Temp = Mild: Moderate (55.0/12.0)

| | | Max_Temp = Mild

| | | | Min_Temp = Very Cold: Moderate (0.0)

| | | | Min_Temp = Cold: Moderate (72.0/36.0)

| | | | Min_Temp = Mild: Light (2.0/1.0)

| | Wind_Speeds = Moderate Wind

| | | sun_hrs = Long: No rain (6.0/1.0)

| | | sun_hrs = Medium: Light (73.0/18.0)

| | | sun_hrs = Short

| | | | Min_Temp = Very Cold: Moderate (0.0)

| | | | Min_Temp = Cold

| | | | | Max_Temp = Hot: Light (0.0)

| | | | | Max_Temp = Warm: Light (34.0/11.0)

| | | | | Max_Temp = Mild: No rain (2.0/1.0)

| | | | Min_Temp = Mild: Moderate (21.0/2.0)

| | Wind_Speeds = Very Strong Wind: Heavy (0.0)

| MONTH = August

| | Wind_Speeds = Light Wind

| | | Min_Temp = Very Cold: Light (1.0)

| | | Min_Temp = Cold

| | | | Max_Temp = Hot: Moderate (5.0)

| | | | Max_Temp = Warm

| | | | | sun_hrs = Long: Light (76.0/41.0)

| | | | | sun_hrs = Medium: Light (459.0/273.0)

| | | | | sun_hrs = Short: Moderate (214.0/105.0)

| | | | Max_Temp = Mild

| | | | | sun_hrs = Long: Light (0.0)

| | | | | sun_hrs = Medium: Moderate (16.0/5.0)

| | | | | sun_hrs = Short: Light (55.0/21.0)

| | | Min_Temp = Mild: Moderate (69.0/22.0)

| | Wind_Speeds = Moderate Wind: Light (138.0/61.0)

| | Wind_Speeds = Very Strong Wind: No rain (1.0)

| MONTH = Septmeber

| | Max_Temp = Hot: Light (4.0/2.0)

| | Max_Temp = Warm

| | | Min_Temp = Very Cold

| | | | sun_hrs = Long: Light (3.0)

| | | | sun_hrs = Medium: No rain (2.0)

| | | | sun_hrs = Short: Light (0.0)

| | | Min_Temp = Cold

| | | | sun_hrs = Long: Light (129.0/60.0)

| | | | sun_hrs = Medium

| | | | | Wind_Speeds = Light Wind: Moderate (706.0/456.0)

| | | | | Wind_Speeds = Moderate Wind: Light (18.0/11.0)

| | | | | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| | | | sun_hrs = Short

| | | | | Wind_Speeds = Light Wind: Moderate (322.0/216.0)

| | | | | Wind_Speeds = Moderate Wind: Light (10.0/4.0)

| | | | | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| | | Min_Temp = Mild

| | | | sun_hrs = Long: Moderate (2.0)

| | | | sun_hrs = Medium

| | | | | Wind_Speeds = Light Wind: Light (34.0/10.0)

| | | | | Wind_Speeds = Moderate Wind: Moderate (18.0/6.0)

| | | | | Wind_Speeds = Very Strong Wind: Light (0.0)

| | | | sun_hrs = Short

| | | | | Wind_Speeds = Light Wind: Moderate (29.0/8.0)

| | | | | Wind_Speeds = Moderate Wind: Light (3.0)

| | | | | Wind_Speeds = Very Strong Wind: Moderate (0.0)

| | Max_Temp = Mild: Light (29.0/2.0)

| MONTH = October

| | Max_Temp = Hot

| | | sun_hrs = Long: Light (9.0/2.0)

| | | sun_hrs = Medium: No rain (22.0/10.0)

| | | sun_hrs = Short: Light (0.0)

| | Max_Temp = Warm: Light (492.0/227.0)

| | Max_Temp = Mild: Light (4.0/1.0)

| MONTH = November

| | Max_Temp = Hot: No rain (6.0/2.0)

| | Max_Temp = Warm

| | | sun_hrs = Long: Heavy (20.0/4.0)

| | | sun_hrs = Medium

| | | | Min_Temp = Very Cold: Heavy (0.0)

| | | | Min_Temp = Cold: Heavy (312.0/141.0)

| | | | Min_Temp = Mild: No rain (2.0)

| | | sun_hrs = Short: Light (5.0/2.0)

| | Max_Temp = Mild: Heavy (0.0)

| MONTH = December

| | Max_Temp = Hot: No rain (4.0/2.0)

| | Max_Temp = Warm: Light (73.0/40.0)

| | Max_Temp = Mild: Light (0.0)

RELHUMIDITY = Wet

| sun_hrs = Long: No rain (5.0/2.0)

| sun_hrs = Medium: No rain (5.0/3.0)

| sun_hrs = Short

| | Max_Temp = Hot: Heavy (1.0)

| | Max_Temp = Warm

| | | MONTH = January: Light (0.0)

| | | MONTH = February: Light (0.0)

| | | MONTH = March: Light (0.0)

| | | MONTH = April: Light (0.0)

| | | MONTH = May: Light (0.0)

| | | MONTH = June: Light (3.0/1.0)

| | | MONTH = July: Light (0.0)

| | | MONTH = August

| | | | Min_Temp = Very Cold: Moderate (0.0)

| | | | Min_Temp = Cold: Moderate (3.0/1.0)

| | | | Min_Temp = Mild: Light (2.0/1.0)

| | | MONTH = Septmeber: Light (0.0)

| | | MONTH = October: Light (0.0)

| | | MONTH = November: Light (0.0)

| | | MONTH = December: Light (0.0)

| | Max_Temp = Mild: Moderate (6.0/2.0)

Classifier output

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances        2000               86.6551 %
Incorrectly Classified Instances       308               13.3449 %
Kappa statistic                          0.8221
K&B Relative Info Score             190040.3226 %
K&B Information Score                 3797.242  bits      1.6453 bits/instance
Class complexity | order 0           4618.193  bits      2.001  bits/instance
Class complexity | scheme           132736.5669 bits     57.5115 bits/instance
Complexity improvement     (Sf)    -128118.374  bits    -55.5106 bits/instance
Mean absolute error                      0.0793
Root mean squared error                  0.233
Relative absolute error                 21.1437 %
Root relative squared error             53.7971 %
Total Number of Instances             2308

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.799     0.054      0.836     0.799     0.817      0.929    No rain
                 0.797     0.065      0.804     0.797     0.8        0.925    Light
                 0.888     0.048      0.857     0.888     0.872      0.954    Moderate
                 0.984     0.012      0.966     0.984     0.975      0.991    Heavy
Weighted Avg.    0.867     0.044      0.866     0.867     0.866      0.95

=== Confusion Matrix ===

   a   b   c   d   <-- classified as
 474  75  40   4 |   a = No rain
  67 458  41   9 |   b = Light
  25  31 498   7 |   c = Moderate
```

APPENDIX B: MULTILAYER PERCEPTRON NEURAL NETWORK OUTPUT

Classifier output

```
Correctly Classified Instances        2582               83.8856 %
Incorrectly Classified Instances       496               16.1144 %
Kappa statistic                          0.7851
K&B Relative Info Score             245806.7278 %
K&B Information Score                 4911.613  bits      1.5957 bits/instance
Class complexity | order 0            6156.2013 bits      2.0001 bits/instance
Class complexity | scheme             8398.4852 bits      2.7286 bits/instance
Complexity improvement     (Sf)      -2242.284  bits     -0.7285 bits/instance
Mean absolute error                      0.0881
Root mean squared error                  0.2679
Relative absolute error                 23.4923 %
Root relative squared error             61.8719 %
Total Number of Instances             3078

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.753     0.061      0.812     0.753     0.781      0.893    No rain
                 0.794     0.091      0.749     0.794     0.771      0.89     Light
                 0.845     0.052      0.841     0.845     0.843      0.925    Moderate
                 0.971     0.012      0.962     0.971     0.966      0.994    Heavy
Weighted Avg.    0.839     0.054      0.839     0.839     0.839      0.925

=== Confusion Matrix ===

   a   b   c   d   <-- classified as
 599 130  56  10 |   a = No rain
  90 621  61  10 |   b = Light
  45  63 636   9 |   c = Moderate
   4  15   3 726 |   d = Heavy
```

APPENDIX C: PART RULE INDUCTION OUTPUT

Classifier output

```
Correctly Classified Instances        1324               86.0299 %
Incorrectly Classified Instances       215               13.9701 %
Kappa statistic                          0.8136
K&B Relative Info Score             126659.4902 %
K&B Information Score                 2531.1356 bits      1.6447 bits/instance
Class complexity | order 0            3077.8392 bits      1.9999 bits/instance
Class complexity | scheme           177367.5528 bits    115.2486 bits/instance
Complexity improvement     (Sf)    -174289.7136 bits   -113.2487 bits/instance
Mean absolute error                      0.0774
Root mean squared error                  0.2521
Relative absolute error                 20.6535 %
Root relative squared error             58.2199 %
Total Number of Instances             1539

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.824     0.08       0.789     0.824     0.806      0.893    No rain
                 0.785     0.073      0.777     0.785     0.781      0.888    Light
                 0.859     0.028      0.905     0.859     0.881      0.921    Moderate
                 0.974     0.006      0.982     0.974     0.978      0.985    Heavy
Weighted Avg.    0.86      0.047      0.862     0.86      0.861      0.922

=== Confusion Matrix ===

   a   b   c   d   <-- classified as
 336  52  17   3 |   a = No rain
  66 296  12   3 |   b = Light
  23  28 316   1 |   c = Moderate
   1   5   4 376 |   d = Heavy
```

PART decision list

RELHUMIDITY = Very Dry AND sun_hrs = Long AND Min_Temp = Very Cold AND MONTH = January: No rain (94.0)

RELHUMIDITY = Very Dry AND sun_hrs = Long AND Min_Temp = Very Cold AND MONTH = February: No rain (84.0)

RELHUMIDITY = Very Dry AND sun_hrs = Long AND MONTH = November: No rain (117.0)

RELHUMIDITY = Very Dry AND sun_hrs = Long AND MONTH = December AND Min_Temp = Very Cold: No rain (81.0)

RELHUMIDITY = Very Dry AND Max_Temp = Hot AND Wind_Speeds = Moderate Wind: No rain (65.0)

RELHUMIDITY = Very Dry AND Max_Temp = Hot AND MONTH = January AND sun_hrs = Long: No rain (51.0)

RELHUMIDITY = Very Dry AND Max_Temp = Hot AND MONTH = March AND Min_Temp = Very Cold AND

sun_hrs = Long: No rain (60.0/2.0)

RELHUMIDITY = Very Dry AND Max_Temp = Hot AND MONTH = March AND Min_Temp = Cold AND

sun_hrs = Long: No rain (135.0/12.0)

RELHUMIDITY = Very Dry AND sun_hrs = Long AND MONTH = April: No rain (24.0)

RELHUMIDITY = Very Dry AND Max_Temp = Hot AND MONTH = March AND Min_Temp = Cold: No rain (16.0/1.0)

RELHUMIDITY = Very Dry AND sun_hrs = Long AND Max_Temp = Hot AND MONTH = February AND Min_Temp = Cold: No rain (119.0/16.0)

RELHUMIDITY = Very Dry AND sun_hrs = Long AND Max_Temp = Hot AND MONTH = December: No rain (24.0/1.0)

RELHUMIDITY = Very Dry AND sun_hrs = Long: No rain (34.0)

RELHUMIDITY = Very Dry AND Max_Temp = Hot AND MONTH = February: No rain (12.0)

RELHUMIDITY = Very Dry AND Max_Temp = Hot AND MONTH = March: No rain (7.0)

RELHUMIDITY = Very Dry AND Max_Temp = Hot AND MONTH = January: No rain (3.0/1.0)

RELHUMIDITY = Very Dry AND Max_Temp = Hot: No rain (7.0)

RELHUMIDITY = Very Dry AND sun_hrs = Short AND MONTH = August AND Max_Temp = Warm: Moderate (12.0/1.0)

RELHUMIDITY = Very Dry AND MONTH = July: Moderate (7.0)

RELHUMIDITY = Very Dry AND Max_Temp = Mild AND sun_hrs = Medium: Light (2.0)

RELHUMIDITY = Very Dry AND Max_Temp = Warm: Light (5.0/2.0)

RELHUMIDITY = Wet AND sun_hrs = Short AND Max_Temp = Mild: Moderate (6.0/2.0)

RELHUMIDITY = Medium Wet AND MONTH = July AND sun_hrs = Short AND Wind_Speeds = Light Wind AND Max_Temp = Warm: Heavy (551.0/269.0)

MONTH = December AND sun_hrs = Long AND Min_Temp = Very Cold AND Max_Temp = Warm: No rain (86.0)

MONTH = December AND sun_hrs = Long AND Max_Temp = Warm AND Wind_Speeds = Light Wind: No rain (74.0/7.0)

Max_Temp = Hot AND sun_hrs = Medium AND MONTH = January AND Wind_Speeds = Moderate Wind: Light (5.0/2.0)

MONTH = April AND sun_hrs = Medium AND Wind_Speeds = Light Wind AND Max_Temp = Hot: Moderate (183.0/100.0)

sun_hrs = Short AND MONTH = September AND RELHUMIDITY = Medium Wet: Light (10.0/4.0)

sun_hrs = Short AND MONTH = September: Moderate (6.0/1.0)