Heavy Rainfall Prediction using Gini Index in Decision Tree

AyishaSiddiqua L, Senthilkumar N C

Abstract: ***In existing systems, it happens that sometimes the data is not accurate and proper data mining techniques not being used and this increases the complexity.We as humans are bound to make mistakes while predicting weather conditions which might result in damage to both life and property. To avoid this, we use data mining algorithms for early warning of climatic conditions such as like maximum temperature, minimum temperature wind speed, rainfall, humidity, pressure, dew point, cloud, sunshine and wind direction from data to predict rainfall. But by using proper algorithms for datasets and using the right metrics, we can achieve the accurate results in prediction of rainfall. Hence, we apply the Decision tree algorithm using Gini Index in order to predict the precipitation with accuracy and it is completely based on the historical data.***

Keywords: ***Rainfall, Prediction, Decision Tree, Gini Index***

I. INTRODUCTION

To be able to predict the changes in the atmosphere for a particular location using Data mining techniques. Rainfall prediction poses right from the ancient times as a big herculean task, because it depends on various parameters to predict. Rainfall depends on humidity, temperature, pressure, wind speed, dew point, etc. The present research is focused on using the gini index as an attribute selection measure in an elegant decision tree to predict precipitation for datasets, making data preprocessing and data transformation on raw weather data set the data mining, prediction model used for rainfall prediction. Decision tree algorithm using Gini Index in order to predict the precipitation with an accuracy. The decision tree is to be constructed and classification rules are then generated. To improve accuracy random forest technique is applied to this result thereby obtaining a result with increased accuracy rate.

BACKGROUND

In the existing systems, sometimes it happens that the data is not accurate and the proper data mining techniques not being used and this increases the complexity and sometimes the data is of poor quality. But by using proper algorithms for datasets and using the right metrics, we can achieve the accurate results in perdition of rainfall.The elegant decision tree algorithm has an efficient, accurate and scalable improvement over other algorithms. Classification and

regression techniques are the important methods for decision tree formation. The main advantage of this algorithm is that it decreases the complexity in computation. This helps us in reducing the time taken and random forest is used to increase the accuracy rate.

II. LITERATURE SURVEY

This paper proposes a data-mining approach in order to predict rainfall based on reflectivity of radar and tipping bucket (TB) data. This is to be applied in vivid watershed areas of Oxford, Lowa. The radar reflectivity data has been collected with the help of NEXRAD systems whereas the TB data has been collected at various locations in Lowa. A series of five algorithms, random forest, classification and regression tree, neural network, k-nearest neighbor and support vector machines were utilized to build prediction models. Among the three proposed models, model II which was built using radar and local TB demonstrated to be most accurate in predicting rainfall in Oxford, Lowa [1]. This paper compares the implementation of rainfall prediction techniques such as Artificial Neural Network (ANN) and Fuzzy Logic (FL) in order to know which one is more accurate among them. An automatic station in Iju helped in occupying the rainfall datasets which is to be used in this research. The outcome of the research was determined based on four major criteria; Mean Absolute Error (MAE), prediction error, prediction accuracy and Root Mean Square Error (RMSE). This paper concludes that the accuracy of neural network is higher than that of fuzzy logic [2]. This paper deals with an idea that an Artificial Neural Network could be utilized in order improvise the rainfall forecast performance. An ANN model was installed at Bangkok, Thailand for about 1 to 6 hours observations which therefore led to several results.

Manuscript published on November 30, 2019.
* Correspondence Author

AyishaSiddiqua L*, M.Tech Software Engineering, Vellore Institute of Technology, Vellore 632014, India.
Senthikumar N C, Assistant Professor(SG), Vellore Institute of Technology, Vellore 632014, India.

The preliminary test portrayed a mixture of meteorological parameters such as air pressure, wet bulb temperature, cloudiness and relative humidity along with rainfall data at particular forecasting stations and in some surrounding stations as well. The predicted forecast of the 1 to 2h matched well with the observed rainfall. The efficiency of the predictions reduced between 4 to 6h. Thus, the 6h forecasting of the model was low and inaccurate. The model can still be used in urban areas to predict flood management and rainfall forecasting [3]. This paper aims to achieve a data mining method to predict rainfall by providing data intensive model instead of compute model. The required data was collected from Indian Meteorological Department (IMD). After analysing the data it was proved that only 7 of the 36 attributes were relevant when it comes to rainfall predictions. The given data intensive model resulted to be more accurate and highly-efficient. The installation of the proposed model is relatively inexpensive. Furthermore, the accuracy of the model can be increased by increasing the amount of learning data and also by designing it for scalable platforms, vertical or horizontal [4]. The main objective of this paper is to compare different rainfall predicting algorithms to find out the most accurate among them. This paper focuses on various algorithms such as, ARIMA, K-Nearest Neighbour Algorithm, SLIQ, Fuzzy Logic, Naïve Bayes, Decision Tree, ANFIS, Neural Network, etc. One of the comparison shows that both K-mean clustering and Decision tree are more suitable for this application. Although when the training set is increased it is observed that the accuracy increases initially and gradually decreases after a certain limit [5]. This paper suggests the implementation of Gini Index in Elegant Decision Tree as an attribute selection measure in order to achieve accurate prediction of precipitation. This paper also compares SLIQ Decision Tree and Elegant Decision Tree using huge amount of datasets. The overall outcome describes that Elegant Decision Tree is more accurate, efficient, and scalable and also reduces complexity in computation which saves 63% of the computational time when compared to SLIQ. It also suggests the usage of dynamic data mode against the static one since precipitation depends on dynamic attributes. Thus, developing a dynamic data mode will eventually help in analysing satellite images which will further help in predicting precipitation [6]. The main aim of this paper is to test the ability of the WRF Model to replicate rainfall in Western Uganda for a period of 20 days. It also tests six cumulus schemes using sign test method, root mean square error, mean error and extended contingency table. Schemes such as the-Grel-Fretas, the Grell 3d Ensemble, the Kain Fritsch, the New Tiedke, the Betts Miller Janji'c, the GrellDevenyi are compared in this paper. Results showed that the GrelFretas scheme had high Probability of Precipitation (POP) rainfall compared to others and thus it can be used as a basic determining scheme. But in case of heavy rainfall the Betts Miller Janji'c scheme has been recommended [7]. This paper presents a Decision Tree model which can predict rainfall along with fog, thunderstorm, cyclone, etc. This paper opens up opportunities for the study of other techniques like Fuzzy Logic, Artificial Neural Network and Genetic Algorithms. During the research, the vector depicted the accuracy of this model to be 100% but it was 87% accurate when compared with actual data. This paper also tells us that this model will get upper hand during catastrophic situations [8]. This paper aims to propose the use of Multiple Linear Regression (MLR) technique to predict rainfall at an early stage. This paper compares the outcome of MLR with clustering, Artificial Neural Network (ANN) and Regression analysis. Analysis of Udaipur rainfall datasets have been utilized in order to propose this method. The analysis was obtained using First apply correlation analysis followed by Regression Analysis. This technique was inaccurate since climate keeps changing for various reasons [9]. This paper aims to propose the Back-propagation Neural Network model to predict rainfall in India. This model relies on factors such as pressure, humidity and dew point. Two-third of the collected data was utilized for making 250 training patterns and One-third of the data was utilized for making 120 testing samples. During the training session 99.79% accuracy was obtained which reduced to 94.28% during the testing. But it is believed that this model is nearly accurate and can be used to predict rainfall [10].

III. DATASET DESCRIPTION AND SAMPLE DATA:

Datasets for rainfall prediction downloaded from climatology information services (hongkong observatory) and outliers and missing values are filtered using data cleaning process. In data preprocessing data cleaning, data integration, data transformation, data reduction takes place. The datasets are preprocessed. It is fed as inputs for training. The rainfall values are clustered using subtractive clustering and the rainfall states identified as low, medium, heavy and given as outputs for training. We need to separate the data into training and testing sets to evaluate data mining models. When we separate a data set into a training set and testing set, most of the data is used for training, and a smaller portion of the data is used for testing. Here, 80% of dataset is used for the training and the rest 20 % is used for testing.

IV. PROPOSED ALGORITHM WITH FLOWCHART

DECISION TREE ALGORITHM:

Decision tree algorithm using Gini Index in order to predict the precipitation with an accuracy and is completely based on the historical data. The decision tree is to be constructed and classification rules are then generated.

To improve accuracy random forest technique is applied to this result thereby obtaining a result with increased accuracy rate. CART algorithm is also used for building decision tree. The dataset is divided into training and testing samples where we apply packages of party and r plot for training sample. Then, we test this system using the testing sample. From this, the misclassification rate is obtained**.**

CONSTRUCTING DECISION TREE: (Step by step)

Step 1-Take random samples and construct decision trees.

Step 2- Determine the importance of each attribute in the dataset.

Step 3- According to the weight, choose the best tree.

Step 4- According to the project perspective, the attribute that has higher importance is taken and the bar graph is plotted.

Step 5- According to the plotted graph, the value of highest frequency is taken as reference for accurate prediction.

Step 6- The dataset is divided into training and testing samples where we apply package (random forest) and

Step 7- Test this system using the testing sample.

Flow Chart with Explanation:

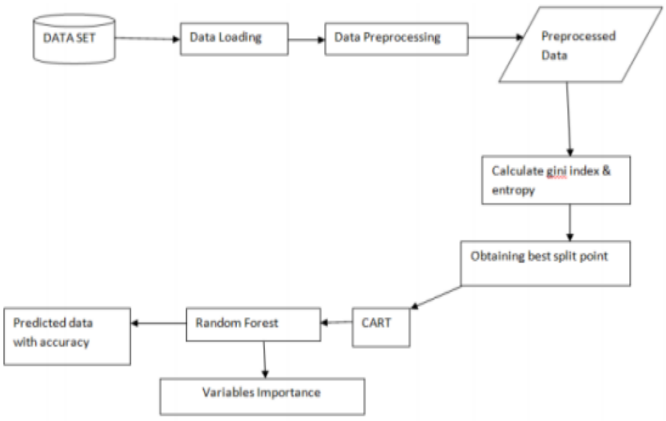
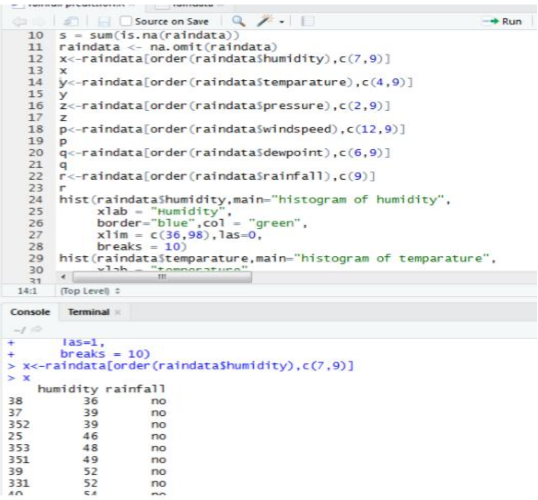


Fig.1 Flowchart for the system

EXPLANATION:

The datasets are preprocessed. It is fed as inputs for training. The rainfall values are clustered using subtractive clustering and the rainfall states identified as low, medium, heavy and given as outputs for training. We need to separate the data into training and testing sets to evaluate data mining models. When we separate a data set into a training set and testing set, most of the data is used for training, and a smaller portion of the data is used for testing. Here, 80% of dataset is used for the training and the rest 20 % is used for testing. We will apply decision tree algorithm for it. In that Decision tree algorithm, by using Gini Index entropy method we will process the rainfall prediction.
]

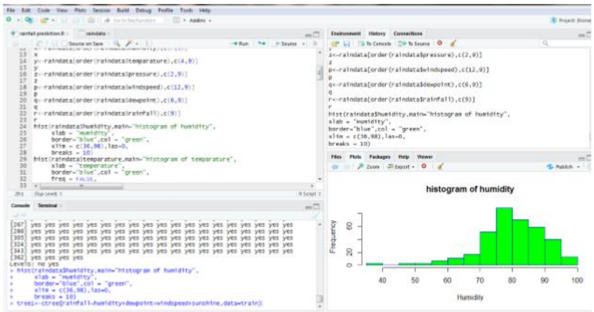
V. EXPERIMENTS RESULTS



```
s = sum(is.na(raindata))
raindata <- na.omit(raindata)
x<-raindata[order(raindata$humidity),c(7,9)]
x
y<-raindata[order(raindata$temparature),c(4,9)]
y
z<-raindata[order(raindata$pressure),c(2,9)]
z
p<-raindata[order(raindata$windspeed),c(12,9)]
p
q<-raindata[order(raindata$dewpoint),c(6,9)]
q
r<-raindata[order(raindata$rainfall),c(9)]
r
hist(raindata$humidity,main="histogram of humidity",
      xlab = "Humidity",
      border="blue",col = "green",
      xlim = c(36,98),las=0,
      breaks = 10)
hist(raindata$temparature,main="histogram of temparature",
```

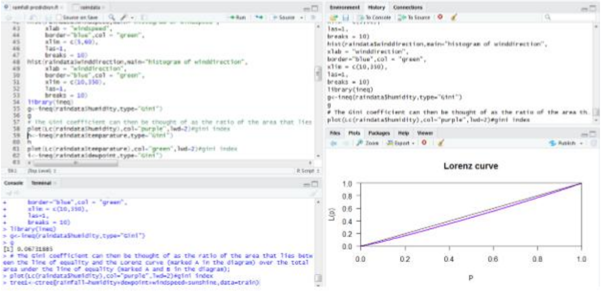
```
+       las=1,
+       breaks = 10)
> x<-raindata[order(raindata$humidity),c(7,9)]
> x
    humidity rainfall
38        36       no
37        39       no
352       39       no
25        46       no
353       48       no
351       49       no
39        52       no
331       52       no
```

Fig.2 Sorting of Data



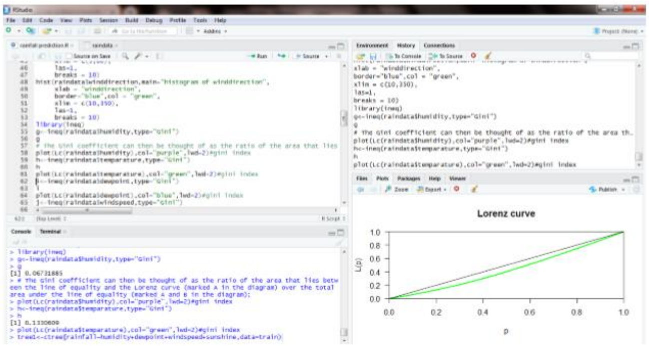
```
[267] yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes
[286] yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes
[305] yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes
[324] yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes
[343] yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes yes
[362] yes yes yes yes
Levels: no yes
> hist(raindata$humidity,main="histogram of humidity",
+       xlab = "Humidity",
+       border="blue",col = "green",
+       xlim = c(36,98),las=0,
+       breaks = 10)
> tree1<-ctree(rainfall~humidity+dewpoint+windspeed+sunshine,data=train)
```

Fig.3 Histogram for Humidity



```
+       border="blue",col = "green",
+       xlim = c(10,350),
+       las=1,
+       breaks = 10)
> library(ineq)
> g<-ineq(raindata$humidity,type="Gini")
> g
[1] 0.06731885
> # The Gini coefficient can then be thought of as the ratio of the area that lies betw
een the line of equality and the Lorenz curve (marked A in the diagram) over the total
area under the line of equality (marked A and B in the diagram);
> plot(Lc(raindata$humidity),col="purple",lwd=2)#gini index
> tree1<-ctree(rainfall~humidity+dewpoint+windspeed+sunshine,data=train)
```

Fig.4 Lorenz curve for Humidity

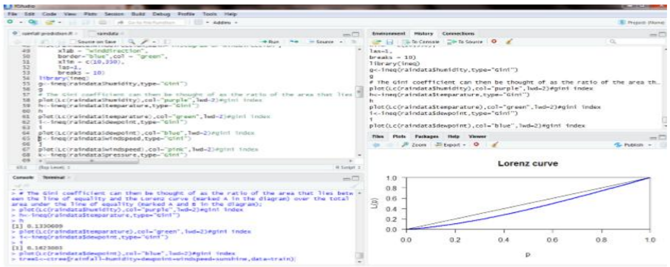


```
> library(ineq)
> g<-ineq(raindata$humidity,type="Gini")
> g
[1] 0.06731885
> # The Gini coefficient can then be thought of as the ratio of the area that lies betw
een the line of equality and the Lorenz curve (marked A in the diagram) over the total
area under the line of equality (marked A and B in the diagram);
> plot(Lc(raindata$humidity),col="purple",lwd=2)#gini index
> h<-ineq(raindata$temparature,type="Gini")
> h
[1] 0.1330609
> plot(Lc(raindata$temparature),col="green",lwd=2)#gini index
> tree1<-ctree(rainfall~humidity+dewpoint+windspeed+sunshine,data=train)
```

Fig.5 Lorenz curve for Temperature

Retrieval Number: D8503118419/2019©BEIESP
DOI:10.35940/ijrte.D8503.118419
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

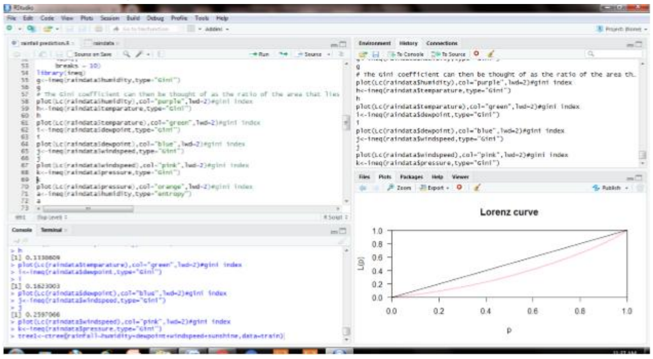


```
     xlab = "winddirection",
     border="blue",col = "green",
     xlim = c(10,350),
     las=1,
     breaks = 10)
library(ineq)
g<-ineq(raindata$humidity,type="Gini")
g
# The Gini coefficient can then be thought of as the ratio of the area that lies
plot(Lc(raindata$humidity),col="purple",lwd=2)#gini index
h<-ineq(raindata$temparature,type="Gini")
h
plot(Lc(raindata$temparature),col="green",lwd=2)#gini index
i<-ineq(raindata$dewpoint,type="Gini")
i
plot(Lc(raindata$dewpoint),col="blue",lwd=2)#gini index
j<-ineq(raindata$windspeed,type="Gini")
j
plot(Lc(raindata$windspeed),col="pink",lwd=2)#gini index
k<-ineq(raindata$pressure,type="Gini")
```

```
las=1,
breaks = 10)
library(ineq)
g<-ineq(raindata$humidity,type="Gini")
g
# The Gini coefficient can then be thought of as the ratio of the area th…
plot(Lc(raindata$humidity),col="purple",lwd=2)#gini index
h<-ineq(raindata$temparature,type="Gini")
h
plot(Lc(raindata$temparature),col="green",lwd=2)#gini index
i<-ineq(raindata$dewpoint,type="Gini")
i
plot(Lc(raindata$dewpoint),col="blue",lwd=2)#gini index
```

```
> # The Gini coefficient can then be thought of as the ratio of the area that lies betw
een the line of equality and the Lorenz curve (marked A in the diagram) over the total
area under the line of equality (marked A and B in the diagram);
> plot(Lc(raindata$humidity),col="purple",lwd=2)#gini index
> h<-ineq(raindata$temparature,type="Gini")
> h
[1] 0.1330609
> plot(Lc(raindata$temparature),col="green",lwd=2)#gini index
> i<-ineq(raindata$dewpoint,type="Gini")
> i
[1] 0.1623003
> plot(Lc(raindata$dewpoint),col="blue",lwd=2)#gini index
> tree1<-ctree(rainfall~humidity+dewpoint+windspeed+sunshine,data=train)
```

Fig.6 Lorenz curve for Dewpoint

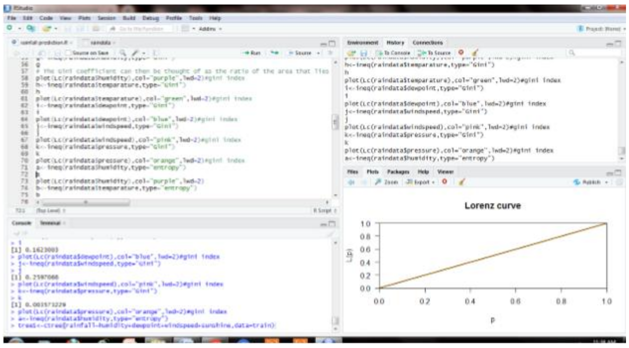


```
     breaks = 10)
library(ineq)
g<-ineq(raindata$humidity,type="Gini")
g
# The Gini coefficient can then be thought of as the ratio of the area that lies
plot(Lc(raindata$humidity),col="purple",lwd=2)#gini index
h<-ineq(raindata$temparature,type="Gini")
h
plot(Lc(raindata$temparature),col="green",lwd=2)#gini index
i<-ineq(raindata$dewpoint,type="Gini")
i
plot(Lc(raindata$dewpoint),col="blue",lwd=2)#gini index
j<-ineq(raindata$windspeed,type="Gini")
j
plot(Lc(raindata$windspeed),col="pink",lwd=2)#gini index
k<-ineq(raindata$pressure,type="Gini")
k
plot(Lc(raindata$pressure),col="orange",lwd=2)#gini index
a<-ineq(raindata$humidity,type="entropy")
a
```

```
g
# The Gini coefficient can then be thought of as the ratio of the area th…
plot(Lc(raindata$humidity),col="purple",lwd=2)#gini index
h<-ineq(raindata$temparature,type="Gini")
h
plot(Lc(raindata$temparature),col="green",lwd=2)#gini index
i<-ineq(raindata$dewpoint,type="Gini")
i
plot(Lc(raindata$dewpoint),col="blue",lwd=2)#gini index
j<-ineq(raindata$windspeed,type="Gini")
j
plot(Lc(raindata$windspeed),col="pink",lwd=2)#gini index
k<-ineq(raindata$pressure,type="Gini")
```

```
> h
[1] 0.1330609
> plot(Lc(raindata$temparature),col="green",lwd=2)#gini index
> i<-ineq(raindata$dewpoint,type="Gini")
> i
[1] 0.1623003
> plot(Lc(raindata$dewpoint),col="blue",lwd=2)#gini index
> j<-ineq(raindata$windspeed,type="Gini")
> j
[1] 0.2597066
> plot(Lc(raindata$windspeed),col="pink",lwd=2)#gini index
> k<-ineq(raindata$pressure,type="Gini")
> tree1<-ctree(rainfall~humidity+dewpoint+windspeed+sunshine,data=train)
```

Fig.7 Lorenz curve for Windspeed

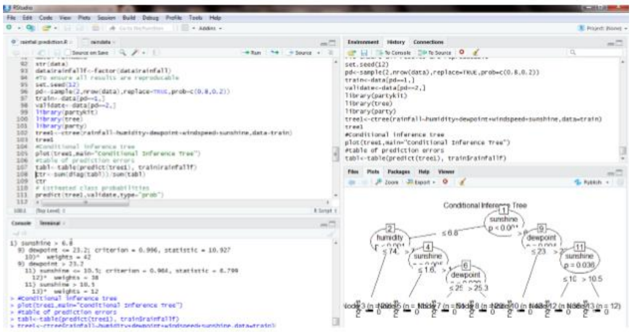


```
g
# The Gini coefficient can then be thought of as the ratio of the area that lies
plot(Lc(raindata$humidity),col="purple",lwd=2)#gini index
h<-ineq(raindata$temparature,type="Gini")
h
plot(Lc(raindata$temparature),col="green",lwd=2)#gini index
i<-ineq(raindata$dewpoint,type="Gini")
i
plot(Lc(raindata$dewpoint),col="blue",lwd=2)#gini index
j<-ineq(raindata$windspeed,type="Gini")
j
plot(Lc(raindata$windspeed),col="pink",lwd=2)#gini index
k<-ineq(raindata$pressure,type="Gini")
k
plot(Lc(raindata$pressure),col="orange",lwd=2)#gini index
a<-ineq(raindata$humidity,type="entropy")
a
plot(Lc(raindata$humidity),col="purple",lwd=2)
b<-ineq(raindata$temparature,type="entropy")
b
```

```
h<-ineq(raindata$temparature,type="Gini")
h
plot(Lc(raindata$temparature),col="green",lwd=2)#gini index
i<-ineq(raindata$dewpoint,type="Gini")
i
plot(Lc(raindata$dewpoint),col="blue",lwd=2)#gini index
j<-ineq(raindata$windspeed,type="Gini")
j
plot(Lc(raindata$windspeed),col="pink",lwd=2)#gini index
k<-ineq(raindata$pressure,type="Gini")
k
plot(Lc(raindata$pressure),col="orange",lwd=2)#gini index
a<-ineq(raindata$humidity,type="entropy")
```

```
> i
[1] 0.1623003
> plot(Lc(raindata$dewpoint),col="blue",lwd=2)#gini index
> j<-ineq(raindata$windspeed,type="Gini")
> j
[1] 0.2597066
> plot(Lc(raindata$windspeed),col="pink",lwd=2)#gini index
> k<-ineq(raindata$pressure,type="Gini")
> k
[1] 0.003571229
> plot(Lc(raindata$pressure),col="orange",lwd=2)#gini index
> a<-ineq(raindata$humidity,type="entropy")
> tree1<-ctree(rainfall~humidity+dewpoint+windspeed+sunshine,data=train)
```

Fig.8 Lorenz curve for Pressure



```
str(data)
data$rainfallf<-factor(data$rainfall)
#To ensure all results are reproducable
set.seed(12)
pd<-sample(2,nrow(data),replace=TRUE,prob=c(0.8,0.2))
train<-data[pd==1,]
validate<-data[pd==2,]
library(partykit)
library(tree)
library(party)
tree1<-ctree(rainfall~humidity+dewpoint+windspeed+sunshine,data=train)
tree1
#Conditional inference tree
plot(tree1,main="Conditional Inference Tree")
#table of prediction errors
tab1<-table(predict(tree1), train$rainfallf)
ctr<-sum(diag(tab1))/sum(tab1)
ctr
# Estimated class probabilities
predict(tree1,validate,type="prob")
```

```
set.seed(12)
pd<-sample(2,nrow(data),replace=TRUE,prob=c(0.8,0.2))
train<-data[pd==1,]
validate<-data[pd==2,]
library(partykit)
library(tree)
library(party)
tree1<-ctree(rainfall~humidity+dewpoint+windspeed+sunshine,data=train)
tree1
#Conditional inference tree
plot(tree1,main="Conditional Inference Tree")
#table of prediction errors
tab1<-table(predict(tree1), train$rainfallf)
```

```
1) sunshine > 6.8
  9) dewpoint <= 23.2; criterion = 0.996, statistic = 10.927
    10)*  weights = 42
  9) dewpoint > 23.2
    11) sunshine <= 10.5; criterion = 0.964, statistic = 6.799
      12)*  weights = 38
    11) sunshine > 10.5
      13)*  weights = 12
> #Conditional inference tree
> plot(tree1,main="Conditional Inference Tree")
> #table of prediction errors
> tab1<-table(predict(tree1), train$rainfallf)
> tree1<-ctree(rainfall~humidity+dewpoint+windspeed+sunshine,data=train)
```

Fig.9 Output

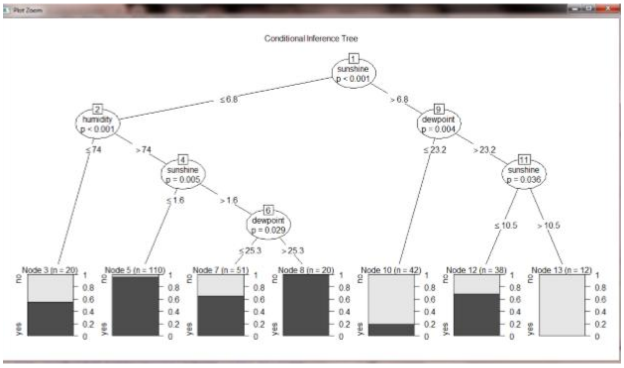


Fig.10 Decision Tree

VI. COMPARATIVE STUDY / RESULTS AND DISCUSSION

In the existing systems, sometimes it happens that the data is not accurate and the proper data mining techniques not being used and this increases the complexity and sometimes the data is of poor quality. But by using proper algorithms for datasets and using the right metrics, we can achieve the accurate results in perdition of rainfall.The elegant decision tree algorithm has an efficient, accurate and scalable improvement over other algorithms. Classification and regression techniques are the important methods for decision tree formation. The main advantage of this algorithm is that it decreases the complexity in computation. This helps us in reducing the time taken and random forest is used to increase the accuracy rate.Here the obtained outcome is a decision tree and a result with improved accuracy rate. This decision tree is used for rainfall prediction that gives the result with a misclassification rate of over 18% thereby having an accuracy rate of 84 %.

OUTPUT/RESULTS

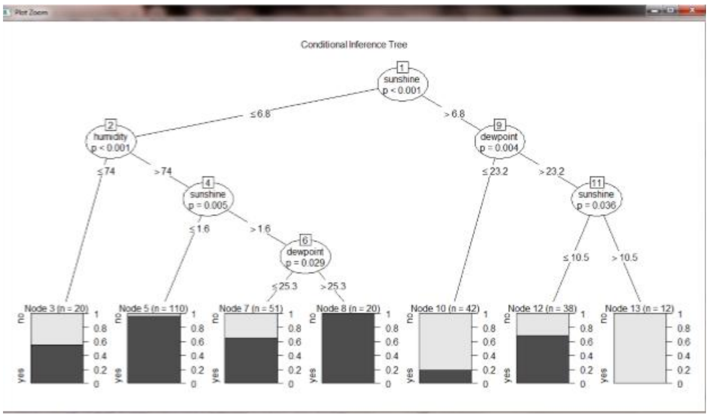


Fig.11 Final Output: Decision Tree

VII. CONCLUSION AND FUTURE WORK

The elegant decision tree algorithm has an efficient, accurate and scalable improvement over other algorithms. Classification and regression techniques are the important methods for decision tree formation. The main advantage of this algorithm is that it decreases the complexity in computation. This helps us in reducing the time taken and random forest is used to increase the accuracy rate.

REFERENCES:

1. Kusiak, A., Wei, X., Verma, A. P., & Roz, E. (2012). Modeling and prediction of rainfall using radar reflectivity data: a data-mining approach. IEEE Transactions on Geoscience and Remote Sensing, 51(4), 2337-2342.
2. Helen, Afolayan&Ojokoh, Bolanle &Falaki, A. (2016). Comparative Analysis of Rainfall Prediction Models Using Neural Network and Fuzzy Logic. International Journal of Soft Computing and engineering 2231-2307. volume 5. 4-7.
3. Hung, N. Q., Babel, M. S., Weesakul, S., & Tripathi, N. K. (2009). An artificial neural network model for rainfall forecasting in Bangkok, Thailand. Hydrology and Earth System Sciences, 13(8), 1413-1425.
4. Nikam, V. B., &Meshram, B. B. (2013, September). Modeling rainfall prediction using data mining method: A Bayesian approach. In 2013 Fifth International Conference on Computational Intelligence, Modelling and Simulation (pp. 132-136). IEEE.

5. Kumar, R. S., & Ramesh, C. (2016, August). A study on prediction of rainfall using datamining technique. In 2016 International Conference on Inventive Computation Technologies (ICICT) (Vol. 3, pp. 1-9). IEEE.
6. LV, N. P., Patro, K. R., & Naidu, M. M. (2013, July). A gini index based elegant decision tree classifier to predict precipitation. In 2013 7th Asia Modelling Symposium (pp. 46-54). IEEE.
7. Mugume, I., Waiswa, D., Mesquita, M. D. S., Reuder, J., Basalirwa, C., Bamutaze, Y., ... &Ayesiga, G. (2017). Assessing the performance of WRF model in simulating rainfall over western Uganda. Journal of Climatology and weather forecasting, 5(1), 1-9.
8. Geetha, A., &Nasira, G. M. (2014, December). Data mining for meteorological applications: Decision trees for modeling rainfall prediction. In 2014 IEEE International Conference on Computational Intelligence and Computing Research (pp. 1-4). IEEE.
9. Sethi, N., & Garg, K. (2014). Exploiting data mining technique for rainfall prediction. International Journal of Computer Science and Information Technologies, 5(3), 3982-3984.
10. Vamsidhar, E., Varma, K. V. S. R. P., Rao, P. S., &Satapati, R. (2010). Prediction of rainfall using backpropagation neural network model. International Journal on Computer Science and Engineering, 2(4), 1119-112.

AUTHORS PROFILE

AyishaSiddiqua L is currently studying M.Tech Software Engineering in Vellore Institute of Technology (VIT), Vellore. Her research interests include Data MiningTechnologies and Software Testing in Cloud platform.

Senthilkumar N C is currently working as an Assistant Professor (Selection Grade) in Vellore Institute of Technology (VIT), Vellore. He has completed his Master of Engineering in Computer Science from University of Madras; Bachelor of engineering from University of Madras. His field of specialization is Web Personalization. He is also interested in Database Technologies, Spatial Database, Data Mining and Web Mining. He has published articles in International and National Journals and Conferences.