# An ANN based analysis of rainfall over Kerala during September -December based on satellite images of cloud formations

Rishidas S[1], Sreekumar M G[2], Sisira Ashok[3]
[1]*Associate Professor, Government Engineering College, Bartonhill*
[2, 3] *M.Tech Student, Government Engineering College, Bartonhill*
*(E-mail: rishidas19731999@gmail.com)*

*Abstract*— In this paper, we analyzed the possibilities of finding correlation between the satellite images of cloud formations in IR1 band and average daily rainfall over Kerala state. Data from images were obtained using image processing techniques and was converted into a feature vector. This feature vectors were used to train a feed forward artificial neural network which updates its weight based on the gradient descent back propagation algorithm. The results showed good correlation between the input and the target vector which consists of three classes of rainfall namely deficit, normal, and excess.

**Keywords—** *Artificial neural network, Average daily rainfall, Gradient descent algorithm, North-East monsoon, Satellite images*

## I. INTRODUCTION

Kerala lies near to the region of equator, gifted with calm and moderate climate throughout the year. The presence of fort like structures in the form of Western Ghats is the reason for this. Kerala experience four seasons namely winter, summer, two rainy seasons called South west monsoon and North east monsoon. We have North East monsoon (also known as retreating monsoon) during the months of October, November and December. In the beginning of this period, the relative differences in atmospheric pressure over the northern parts of India and the Indian Ocean along with other factors cause the wind to sweep south. This brings heavy precipitation over parts of Andhra Pradesh, Tamil Nadu, Kerala and other neighboring regions. Tamil Nadu gets around forty eight percent of its rainfall during this part of the year. There have been variations in the average rainfall during North-East monsoon in the recent past.

This study aims at finding the correlation between data from cloud images and the amount of rainfall in Kerala. For finding the possibilities of short term prediction, we analyze the relation between satellite images of a particular day with rainfall information of the next day. The remaining part of the paper can be classified as follows, section II explains about the dataset used in the work, section III describes the proposed work, section IV is results and discussion, finally section V concludes the work.

## II. DATABASE

The Satellite image data was obtained for the months of September, October, November and December. These were obtained from the Indian Meteorological Department website. A satellite senses the earth's atmosphere in specific bands of wavelength known as channels/spectral bands. The images obtained are from INSAT 3D satellite in IR1 band which has a spectral range of 10.3-11.3µm. The normal IR image obtained from the satellite is a grayscale image where the land appears bright and cloud appears dark. This is due to the fact that higher the temperature of the object, more radiance reaches the satellite, causing the object to appear bright. But for a conventional view the image is inverted such that cloud appears bright and land appears dark.

The target vector is formed from the average rainfall data available in the IMD website. We classify the daily average rainfall below 5mm as deficit, between 5 and 30 mm as normal and above 30 mm as excess. For computation purposes we represented the three classes in the form of vectors. This gives us an N x 3 vector where 'N' represents the number of samples, here the number of samples is 244. Fig. 1 shows an IR image of Kerala region from INSAT 3D satellite and its corresponding brightness temperature image.

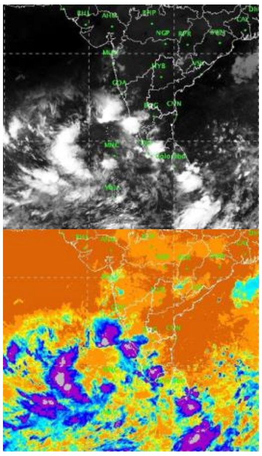

Fig. 1 IR image of Kerala along with its equivalent brightness temperature image

## III. PROPOSED WORK

The proposed method is basically a supervised learning solution where we provide the target label or target data corresponding to each of the training instances to the learning system or model. It then learns the relation between training data and the corresponding target for classifying the instances accordingly. Fig. 2 give the block diagram of supervised learning system. For this work, training data is the satellite image obtained from IMD website which is further subjected to pre-processing.

### A. Pre-processing

The first step is to crop the image for the region of interest. We wanted to find the correlation of cloud spread across the region given by the coordinates 0º-20º N and 70ºE-90ºE. The cropped portion is then resized into 256 x 256 pixel image. The image thus obtained was smoothened using a median filter which replaces every pixel by the median of its neighboring values. This reduces the effect of noise in the image considerably. The filtered image was then segmented and converted to a black and white image. The global threshold for segmentation was found by using Otsu's method. A two dimensional Fast Fourier Transformation is applied to the resultant image to obtain FFT coefficients which was then used for feature extraction. We also transformed the image using Discrete Cosine Transform and compared its performance with the FFT.

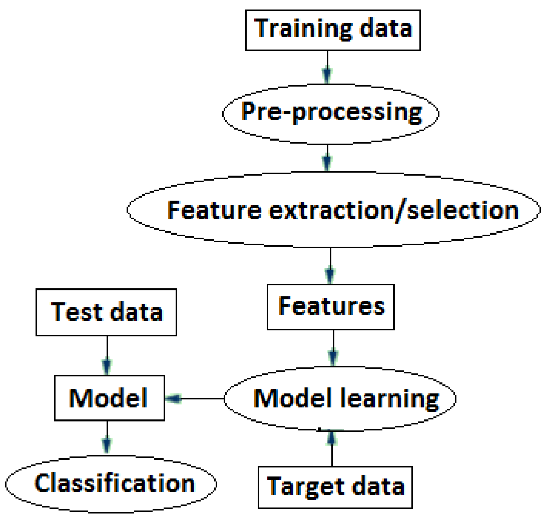


Fig. 2 Block diagram of the supervised learning system proposed in work

### B. Feature Extraction

Five statistical features from the image namely Mean, Variance, Skewness, Kurtosis and Entropy were extracted from the images. The resulting feature vector is an N x 5 matrix where N denotes the number of samples used. In our work, we used a total of 244 training instances corresponding to 122 days with 2 samples per day for daytime and nighttime images.

#### 1) Mean

Mean is normally defined as the middle value of a sequence of numbers. Basically it is defined by the ratio of sum of values to the number of values. Mean is given by the formula,

$$\mu = \frac{1}{N}\sum_{i=0}^{n} X_i \tag{1}$$

Here mean is denoted by 'μ', 'n' denotes the number of elements and $X_i$ denotes the $i^{th}$ element of the series X. For an image, 2-dimensional mean is evaluated by finding mean of individual column mean values.

#### 2) Variance

The variance of any random value can be calculated as expectation of squared deviation of the value from the mean value. Mathematically it can be represented as,

$$v = E(X-\mu)^2 \tag{2}$$

Here ' $v$ ' represents the variance, E represents the expectation operation, X and μ represents data value and mean respectively.

#### 3) Skewness

Skewness is defined as the variation in symmetry on the distribution of values. It is defined as the third standard moment for a random variable. We can find the skewness by the following equation,

$$\gamma = E\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] \tag{3}$$

Here γ gives the skewness of the distribution, σ represents the standard deviation, X and μ are the data value and mean respectively.

#### 4) Kurtosis

Shape of the distribution can also be given by kurtosis. It is defined as the fourth moment.

$$kurt(X) = E\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] \tag{4}$$

The above expression gives kurtosis where X, μ and σ are same as the previous expression.

*5) Entropy*

Entropy is used to characterize image texture. It is also termed as average information. It also gives the measure of statistical randomness. We can define entropy as,

$$E = \sum p \log_2 \frac{1}{p} \tag{5}$$

Here p is the probability of difference in neighboring pixels.

### *C. Training and model development*

Artificial neural networks are systems that process information and have its resemblance to human brain. Comparing to traditional systems they are good at tasks like pattern matching, classification, function optimization etc. They contain number of interconnected elements for processing called nodes. Figure shows a simple architecture of a neuron. Input to the node or the element for processing is the sum of weighted vectors. A function called activation function either excites or inhibits this summation output. This function can be log-sigmoid, tan-sigmoid, linear etc. The output is either final output or the input to another neuron. The output equation of node p is given by,

$$y_p(t+1) = a\left(\sum_{q=1}^{m} w_{pq} x_q(t) - \theta_p\right) \tag{6}$$

Where the $y_p(t+1)$ represents the output, , $w_{pq}$ represents the weight between neurons q and p, $\theta_p$ represents the bias value, $x_q(t)$ represents the input or output from $q^{th}$ neuron as the case may be. A neural network may consist of several of such neurons in layers. Based on the connections of neurons, there are several structures in which we are interested in the feedforward neural network. In this type, the connections are made with the succeeding neurons without a backward path.

In our work, a multilayer feedforward neural network, with learning function gradient descent backpropogation algorithm was selected as the model. Fig. 3 shows the neural network model used in the work with linear activation function.

The learning function adapts the weights by reducing the performance function which has been selected as the mean square error. We are interested in finding the weights which gives minimum mse. We have a hidden layer with 10 neurons and an output layer with 3 neurons. Comparisons of the performance with tansig, logsig and linear activation function have also been done. The results have been tabulated in the results and discussion section. Seventy percent of the data was used for training and 15 percent each for validation and testing. The choice of picking data values were done on a random basis.

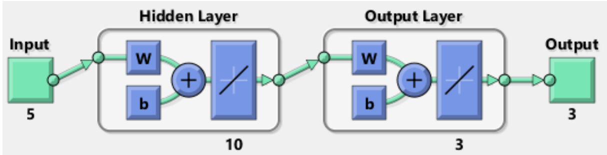


Fig. 3 Neural network model used in the work with linear activation

## IV. RESULTS AND DISCUSSION

The network was trained by giving the feature matrix and target vector as inputs and setting activation function as tan sigmoid first. After continuous trainings, a maximum accuracy of 68.9 percent was achieved. Fig. 4 shows the confusion matrix for the maximum accuracy.

The diagonal cells in green color represent correct classification and their percentage out of total samples. Each row represents output classes in which the green cells give correct classification and red cells give misclassification of samples from the other two classes. The last cell corresponding to each output class shows the percentage of correct classification and misclassification of samples in that particular output class. Similarly each column represents the target class. The last cell corresponding to each target class shows the percentage of correct classification and misclassification of samples in that particular target class. Finally the last cell in the matrix shows the overall training accuracy and error percentage.

From the figure it can be seen that all the training instances of excess class have been wrongly classified. Also, none of the training instances have been classified as excess even as a wrong classification which is understood from the all zero values in third row. Changing the activation function to log sigmoid for nodes in both the hidden layer and output layer gives an accuracy of 66.4 percent. Fig. 5 gives the confusion matrix when activation function was changed to log sigmoid.

Target Class

| Output Class | | | | |
|---|---|---|---|---|
| | 151<br>61.9% | 59<br>24.2% | 3<br>1.2% | 70.9%<br>29.1% |
| | 11<br>4.5% | 17<br>7.0% | 3<br>1.2% | 54.8%<br>45.2% |
| | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | NaN%<br>NaN% |
| | 93.2%<br>6.8% | 22.4%<br>77.6% | 0.0%<br>100% | 68.9%<br>31.1% |

Fig. 4 Confusion matrix for tan sigmoid activation function

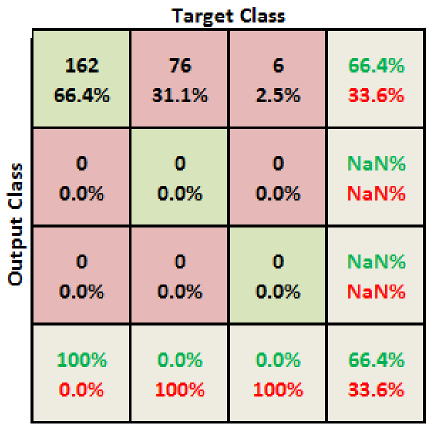


Fig. 5 Confusion matrix for log sigmoid activation function

Here also it could be seen that all the instances has been classified into the first category which corresponds to the deficit class and neither excess nor normal instances have been correctly classified. We also trained the data using linear activation function in both the 2 layers, where the function has a positive slope and passes through the origin. The corresponding confusion matrix can be given in Fig. 6. One can see from the figure that more number of instances have been classified correctly in the normal class and one instance has been classified into excess category even though it's a misclassification. The issue of no correct classification in the excess category still remains.

We also transformed the images using Discrete Cosine Transform and the corresponding comparison of results could be tabulated as in Table 1. From the table we could see that FFT transformation yields comparatively better result and linear activation function classifies better than the other two.

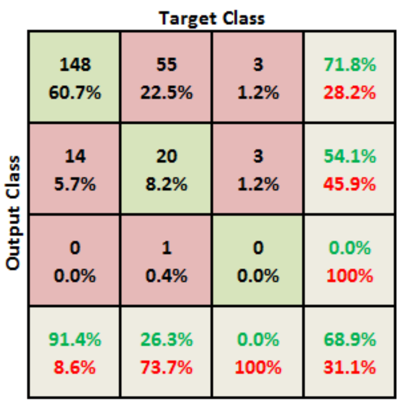


Fig. 6 Confusion matrix for linear activation function

TABLE I. ACCURACY COMPARISON

| Transformation used | Activation function | | |
|---|---|---|---|
| | *tansig* | *logsig* | *linear* |
| DCT | 68.4 | 68.4 | 68 |
| FFT | 68.9 | 66.4 | 68.9 |

## V. CONCLUSION

The correlation between satellite images in IR1 band and the average daily rainfall was analyzed through the work presented. We found that there is considerable amount of correlation between the images and the average rainfall per day in Kerala region. We formed the feature vector from the transformed images and used it to train a neural network.

From the results we could see that FFT along with linear activation function gives better performance. We also found out that there was no instance classified into the third category namely excess, also all the excess class instances were wrongly classified. This can be considered as a part of the work which needs improvisation.

## REFERENCES

[1] Anil K. Jain. "Statistical Pattern Recognition: A Review". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22(No. 1), January 2000.
[2] Anil k Jain. "Digital Image Processing". IEEE transactions on signal processing, Vol. 43(No. 2), February 1995.
[3] Simon O. Haykin. "Neural Networks and Learning Machines". Phi learning pvt. Ltd-New Delhi.
[4] Shruti Upadhyaya and RAAJ Ramsankaran. "Support Vector Machine (SVM) Based Rain area Detection from Kalpana-1 Satellite Data". ISPRS Technical Commission VIII Symposium, 09-12 December 2014, Hyderabad, India
[5] H. Vathsala, Shashidhar G, Koolagudi; 2017. Prediction model for peninsular Indian summer monsoon rainfall using data mining and statistical approaches. Computers & Geosciences 98 (2017) 55–63.
[6] Goswami, P., Srividya, P., 1996. A neural network design for long range prediction rainfall pattern. Curr. Sci. 70, 447–457.
[7] Dwivedi, Suneet, Pandey, Avinash C., 2011. Forecasting the Indian summer monsoon intraseasonal oscillations using genetic algorithm and neural network. Geophysical Research Letters 38
[8] Maheswary V Rishidas S and Sujith Kumar T P. "An Investigation into Rainfall over Kerala using Satellite Images Analyzed using Discrete Cosine Transform". IJERT, ISSN: 2278-0181, Vol. 4 Issue 04, April-2015.
[9] Singh, P., Borah, B., 2013. Indian summer monsoon rainfall prediction using artificial neural network. Stoch. Environ. Res. Risk Assess. 27, 1585–1599.
[10] Sperber, K.R., Palmer, T.N., 1996. Interannual tropical rainfall variability in general circulation model simulations associated with the atmospheric model intercomparison project. J. Clim. 9, 2727–2750.