Contextual-window based neural network model for predicting rainfall in rural India

Paavni Rattan, Rishita Anubhai, Amogh Vasekar
Stanford University

Abstract:
We propose a neural-network based model for predicting rainfall and subsequently, the occurrence of floods for any given city in India. Our initial implementation has primary focus on a rural city in Bihar, a state which experiences erratic rainfall behavior.
We start with simpler models based on time-series analysis and linear regression, ultimately combining them using a novel contextual-window method. The initial results are promising and comparable with the state-of-the-art methods published in this domain.

Introduction:
In this project, we aim to develop novel methods that allow us to reliably predict the amount of rainfall a given Indian city would receive in any month. As an extension, we also develop a model that predicts the occurrences of floods in that area.
The motivation comes from the fact that in developing countries, a majority of the farmers rely on rainfall for cultivation of their crops, and plan their sowing or harvesting periods accordingly. Erratic rainfall behavior has a particularly adverse effect on their planning, leading to huge economic losses. In recent times, the burden of these losses due to unpredictable rainfall behavior has lead to a sharp rise in suicide among farmers. This is especially true in rural Indian areas where a majority of the population earns their livelihood through farming and related activities. We try to address this problem using machine learning techniques.

Related Work:
This field of work is popularly known as the (All India Summer Monsoon Rainfall) AISMR prediction. Since rainfall prediction can have tremendous socio-economic impact, work on AISMR prediction has been an evolving topic of research. Work on AISMR prediction began as early as the 1890s using Himalayan snowfall data, and extra factors such as wind pressure in Mauritius, Zanzibar and Seychelles. These prediction models were all evolving more along the lines of choosing predictors, rather than developing machine learning techniques.
The predictors were broadly classified under four main categories as follows:

1. Regional conditions
2. ENSO (El-Nino South Oscillation) indices
3. Cross equatorial flows
4. Global/Hemispheric conditions (*Kumar et al ,1995*)

The regional circulation conditions were observed to be most significant of these categories, while ENSO indices were also considered good indicators for this work.
After this point, while work continued to be done with respect to choosing the right indicators, machine learning techniques used for AISMR prediction also started diversifying. To avoid over fitting as a result of too many predictors, or predictors that were highly correlated, PCA became a commonly agreed technique. The two major methods used for generating the model were multi-variate linear regression and non-linear regression using neural networks. Methods such as Limited Area Model (LAM) and Mesoscale Model 5 (MM5) are categorized as Numerical Weather Prediction (NWP) methods, which also form an important category of methods.
As NWP methods became more popular, a shortcoming found with these methods was high variation in results based on initialization conditions, and their approximations of complicated physical process and interactions. As a result, ensemble methods and superensemble methods (*Krishnamurti, 2000*) became the state-of-the-art methods, which are essentially a mix of multiple NWP methods. The ensemble methods weigh the NWP methods individually based on the spatial and temporal performance of each method, and combine the results by these weights. Many recent papers have analyzed the performance of these methods (*Bhowmick – Durai, 2007*).

Data:
We primarily use data made publicly available by India Water Portal and Tyndall Centre for Climate Change Research. The data is published on a website, which we scraped due to lack of APIs, to create structured records.

The data contains information from 1901 to 2002, at monthly granularity, about all Indian cities for 11 climatic parameters listed below:

- Precipitation (i.e. rainfall)
- Cloud cover
- Wet day frequency
- Vapor pressure
- Ground frost frequency
- Potential evapotranspiration
- Reference crop evapotranspiration
- Minimum temperature
- Maximum temperature
- Average temperature
- Diurnal temperature range

Thus, in essence, we have time-series data about rainfall and related climatic parameters for Indian cities. We picked the city of Kishanganj in the state of Bihar, which has seen high deviation in rainfall, to train and test our model. Choosing one city allows us to efficiently concentrate on development and implementation of methods.
Since the data contains parameters using different metrics and scales, we normalize the data. We also employ SVD to try and remove highly correlated parameters. We found that SVD does not significantly alter the data.
The second data-set was created by annotating occurrences of floods in the state of Bihar for every month from 1926 to 2002. This was done using publicly available reports.

Methodology:
We start by analyzing the data using two tangential methods. In the first, we only use historical rainfall information and view it as a time-series of rainfall with 1092 data points (91 years x 12 months). In the second method, we use only the climatic features mentioned above to predict rainfall. Both the methods use linear regression. Our major effort focuses on effectively combining the above two methods, since each has its own advantages. We describe this method based on neural networks later in the section.
The above methods address the problem of predicting rainfall. In each of these cases, the training data is a slice of 90 years of the whole data-set. We test on the data for remaining 10 years.
To predict floods, we use logistic regression using the climatic features (the second data-set mentioned in the previous section). We train on data from 1926- 1990, and test for data from 1990 – 2002.

1. Time-series analysis
We disregard the climatic features and consider only rainfall in this approach. Given the seasonality of months in India, we follow the intuition that the rainfall in any month is correlated to the observed rainfall in the previous 12 months, along with the rainfall in the same month over the previous 10 years. For example, this model builds on the hypothesis that rainfall for the month of August in year 2002 is dependent on the rainfall observed from August 2001 to July 2002, as well as the rainfall recorded in the month of August for the years 2001, 2000... 1992. This forms our historical time-series of data.
Thus, every training data-point now contains 22 values representing the features mentioned above, and the actual value of the rainfall for the month of interest.
We then use linear regression to predict the rainfall using above features.

2. Climatic parameters analysis
This method assumes that the rainfall in a month is dependent on the 10 other climatic parameters mentioned previously. Thus, for this analysis, we use linear regression to model the rainfall as a function of all the other climatic features.

3. Feed forward Neural Network with single hidden layer
Both the complementary methods described above capture a particular relationship in the data. Our major effort has been concentrated on trying to combine the two approaches. The intuition behind this was that a model that combines the time series relationship of the data along with the effect of climatic features should be ideal. This

approach uses a combined data-set. Thus, every row represents one month of a year in the matrix input to the neural network.

(a)Non-contextual neural network
In this approach, we provide the complete data matrix as input to the neural network. Every data point is a simple concatenation of all the climatic features. The model is trained using this matrix as an input, and by tuning the parameters for number of hidden nodes, number of iterations to convergence, learning rate and non linear activation function.

(b)Contextual window based neural network
In this novel approach, we define a context for each data point based on a rolling window. The intuition behind this approach was that rainfall in a certain period of the year depends on the rainfall in the same period of previous years, rather than all the rainfall in the past. As a result, the neural network gets inputs in the form of what we define as 'context windows' i.e. windows of relevant feature vectors of the past for a given month. Example: If the subject of our attention is 'January 1950', the window is a concatenation of feature vectors of January 1949, January 1948, January 1947 and so on. Another type of context we defined was a context of the immediate past few months where January 1950 would get the context of December 1949, November 1949 and so on. We experimented with providing different 'sizes' and 'types' of context windows. We summarize this approach as *Neural Networks with Contextual Windows*. We experimented with windows based on adjoining months, same months in the previous years and same months in the available future years, and combination of these approaches. On parameter tuning, we found that the month-wise historical contextual windows gave the best results, and the intuition behind yearly cycles of seasons and rainfall seemed to hold true.

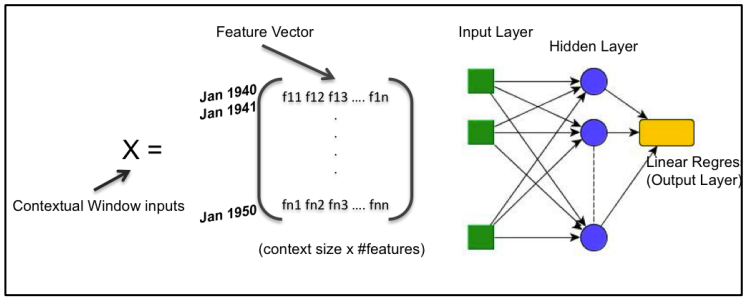

Figure 1: Contextual-window based Neural Network

Stochastic tuning of weights
We use a step by step method for tuning the weights used by neural networks, using the optimal weights obtained from one window as initialization of weights for next window. We initialize the weights randomly between (-ε,ε) in the first step where

$$\varepsilon = \sqrt{6}/\sqrt{(hidden\_size + (number\_of\_features * context\_window\_size))}$$

This is because ε = √6/√ (fan_in + fan_out) is known to be a good choice for this range of initialization in neural networks.

Logistic Regression for predicting floods
To predict the occurrence of a flood, we have used the logistic regression method. It is essentially a binary classification problem with rainfall for a month being the input. Once we predict the amount of rainfall using the contextual-window based approach, we feed the information to a logistic regression based method for classification. As mentioned at the start of the section, the model was trained using flood occurrences from 1926.

Results:
To predict the rainfall, we use two error measures to evaluate and compare the models. The first error measure is the Mean Absolute Error (MAE) of the predicted rainfall. However, this measure can be misleading since our final goal is to predict the occurrence of floods, and not to accurately predict the rainfall. Specifically, being able to predict rainfall within a range of say, 20 to 30 mm, when the actual rainfall is in the range of 450 mm is still very accurate. In essence, a more relevant error measure should also capture the deviation of the predicted rainfall

relative to the actual rainfall. Thus, we also report Weighted Absolute Error (WAE) of our predictions, with the actual rainfall constituting the weights. This allows us to measure the variance in predictions relative to the actual rainfall observed.

In our initial analysis, the historical data explains 80% (R^2 = 0.8042) of variance in the precipitation and climatic data explains 93% (R2 = 0.9335) of the variance in precipitation. On tuning the non-contextual neural network we find the following values as optimal for parameters: number of nodes in the hidden layer = 10, number of iterations to convergence= 1000, leaning rate = 0.01 and activation function = 'tanh'. Using this as baseline we tune the contextual-window based model and find the following values as optimal for parameters: number of nodes in the hidden layer = 10, number of iterations to convergence= 1000, regularization parameter = 0.00001 and window size(if used) = 3. We also find that using only the past data is most optimal.

We find the contextual-window based approach using a 10-year look-back window to work the best using both the MAE and WAE errors. As expected, the non-contextual neural network method reports a lower MAE, but only slightly. Importantly, it reports a higher WAE as compared to the time-series approach. This suggests that non-contextual approach does not completely capture the historical relationship. Thus, it corroborates with our intuition of using contextual windows.
We use cross validation technique to test our models. Specifically, we train on data up to the year 1990 and test on data from 1991 to 2002.

	Time-series analysis	Climatic feature based regression	Non-contextual neural network	Contextual neural network
Mean Absolute Error (mm)	34.81	25.71	25.51	9.09
Weighted Absolute Error (mm	3.44×10^{-10}	5.80×10^{-10}	5.56×10^{-10}	0.77×10^{-10}

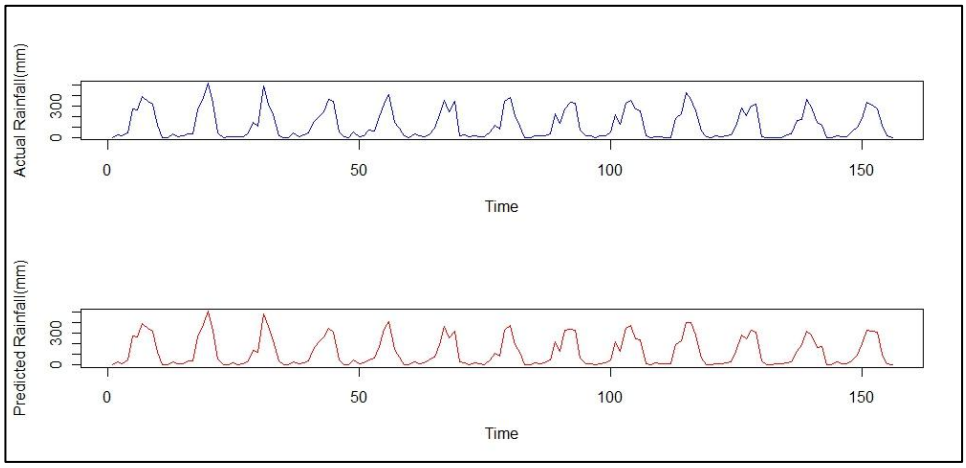

Figure 2: Actual vs. Predicted rainfall

The above figure compares the rainfall predicted by the contextual-window model against the actual observed rainfall in our testing set. It is important to note the correct prediction of peaks in some months, which allows us to proceed with the prediction of floods. The figure below shows the distribution of error (MAE) for the model. We observe a long tailed distribution indicating that most of the values predicted by us have a low mean absolute error and only a few values are very different from actual values.

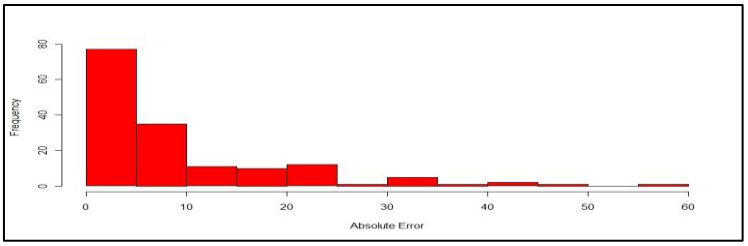

Figure 3: Distribution of error

The latest published results (*Vivekanandan, 2012*) in this domain use RMS error as the measure. Compared to their RMS error of about 20mm our model obtains a value of 14mm.

Our logistic regression model for predicting floods has a precision of 90% but a recall of only 25%. The lower recall can be explained as the occurrence of floods is a rare event, and we have limited training data.

Future Work:
We plan to evaluate our model against the data for other cities with different patterns of rainfall. We also plan to run our algorithm on data-sets used by existing algorithms so as to obtain a better comparison with these methods. We also plan to experiment further using more hidden layers, and see if it provides better results.
It is conjectured that atmospheric pressure and winds are highly correlated to rainfall, and we are trying to obtain data for the same to use in our model.
In addition, we wish to extend the work to predict droughts as well.

Conclusion:
We have successfully developed a novel method using contextual windows to predict rainfall in a rural Indian district with mean absolute error of about 9mm. Our method yields comparable results than currently published work in the domain. Subsequently, the algorithm can predict the occurrences of floods in the area. We hope our work and methods will help the concerned authorities plan better to help farmers in rural India, and also motivate usage of machine learning techniques to solve a socially relevant issue.

References:
1. *India Water Portal*. Arghyam, 27 July 2009. Web. 14 Nov. 2012. <http://www.indiawaterportal.org/metdata>.
2. Munot, A. A., and K. Krishna Kumar. "Long range prediction of Indian summer monsoon rainfall." *J. Earth Syst. Sci.* 116.1 (2007): 73-79.
3. Roy Bhoumik, S. K., and V. R. Durai. "Mulit-model ensemble forecasting of rainfall over Indian monsoon region." *Atmosfera* 21.3 (2008): 225-39.
4. Cannon, Alex J., and Ian G. McKendry. "Forecasting All-India Summer Monsoon Rainfall Using Regional Circulation Principal Components: A Comparison between Neural Network and Multiple Regression Models." *International Journal of Climatology* 19 (1999): 1561-78.
5. Guhathakurta, P., M. Rajeevan, and V. Thapliyal. "Long Range Forecasting Indian Summer Monsoon Rainfall by a Hybrid Principal Component Neural Network Model." *Meteorology and Atmospheric Physics (Springer - Verlag)* 71 (1999): 255-66.
6. Vivekanandan N., "Prediction of seasonal and annual rainfall using artificial neural network" *India Water Week 2012 – Water, Energy and Food Security*.

Big thanks to Prof. Andrew Ng and the staff of CS229 for guiding us throughout the project