Extreme Rainfall Prediction in India

Group - 7

Debanjan Chatterjee - 20111016
P J Leo Evenss - 20111038
Shilpa Chatterjee - 20111057
Shruti Sharma - 20111061
Bopanna Tej Kiran - 20111070

Indian Institute of Technology, Kanpur

December 6, 2020

Introduction

- The variability of the Indian Summer Monsoon affects the agricultural production, industry, hydroelectric power, and causes severe strain on the national economy.
- Occurrence of extreme rainfall is an important problem in the field of meteorology as it has a enormous impact on the life of people.
- Every year people across the globe suffer from severe consequences of heavy rainfall like flood, spread of diseases, wastage of agricultural produce, loss of life and belongings etc.
- The government of India spends large amount of money to provide relief in the affected areas.

Problem Statement

The available methods for heavy rainfall prediction are able to predict only 6 h prior to the event.
We have tried to address the problem of predicting the occurrence of extreme rainfall events across regions during the summer monsoon months of June-September atleast 48-hour and 24-hour before.

The information of regional and sub-divisional rainfall is very much useful to the Researchers, Policy makers and Governmental agencies. This will ensure that least damage is caused by heavy rainfall events.

Dataset

- Daily data of weather parameters at surface and multiple levels is extracted from NCEP/NCAR reanalysis data. (Latitude - 5° to 40° north, Longitude - 65° to 100° east)
- Rainfall data is obtained from India Meteorological Department (IMD, Pune).
- Total parameters sum to 22.
- 20 years have been considered for analysis (1997-2016).

Weather Variables

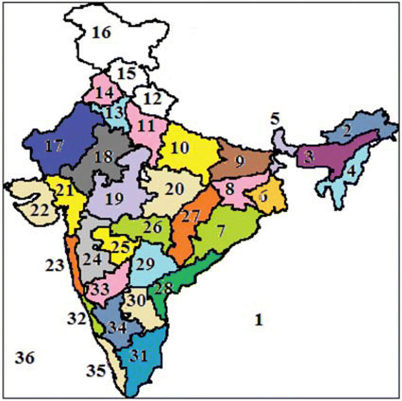
The weather variables this study considers are:

- Air temperature
- Mean sea level pressure
- Precipitable water
- Relative humidity
- Vertical wind velocity (omega)
- U-wind
- V-wind.

Apart from surface level, for multiple levels we have considered weather parameters at 850-, 600- and 400-hPa levels.

Subdivisions

There are 36 meteorological subdivisions of India.



1.	Andaman & Nicobar Islands
2.	Arunachal Pradesh
3.	Assam & Meghalaya
4.	Naga., Mani., Mizo. & Tripura
5.	Sub-Him. W. Bengal & Sikkim
6.	Gangetic West Bengal
7.	Orissa
8.	Jharkhand
9.	Bihar
10.	East Uttar Pradesh
11.	West Uttar Pradesh
12.	Uttaranchal
13.	Haryana, Chandigarh & Delhi
14.	Punjab
15.	Himachal Pradesh
16.	Jammu & Kashmir
17.	West Rajasthan
18.	East Rajasthan
19.	West Madhya Pradesh
20.	East Madhya Pradesh
21.	Gujarat
22.	Saurashtra, Kutch & Diu
23.	Konkan & Goa
24.	Madhya Maharashtra
25.	Marathwada
26.	Vidarbha
27.	Chattisgarh
28.	Coastal Andhra Pradesh
29.	Telangana
30.	Rayalaseema
31.	Tamil Nadu & Pondicherry
32.	Coastal Karnataka
33.	North Interior Karnataka
34.	South Interior Karnataka
35.	Kerala
36.	Lakshadweep

1

[1] Image Source: Meteorological Subdivision - OGD Platform India

Methodology

- The validity of any statistical analysis depends on the quality of the data used in the analysis.
- Subdivisions are distributed fairly uniformly over the country, so districts from each subdivision were selected to form the network.
- A total of 22*225 = 4950 variables available for each day since daily data is taken. For 48 h and 24 h prior prediction, the variables are increased to 4950*2 = 9900.
- Missing values are flagged with -9.96921e+36f
- Excess if $R > or = M{+}SD$, where M and SD are Mean and Standard Deviation of region's rainfall. R is the region's rainfall.

The Beginning

We started with a set of 7 features namely air temperature, mean sea level pressure, precipitable water, relative humidity, vertical wind velocity (omega), U-wind and V-wind at the surface level The Indian subcontinent was divided into equi-sized 225 grids and the daily data was prepared from these resulting in 225 * 7 =1575 features per day. The model trained with these features resulting in precision = 0.60 and recall = 0.16.

The features were then scaled up considering the different air pressure levels of 850-, 600- and 400-hPa and then dataset amounted to 22 feature attributes , thus 225 * 22=4950 per day and 4950 * 2=9900 for 2 consecutive days.

Feature Reduction using Auto-encoders

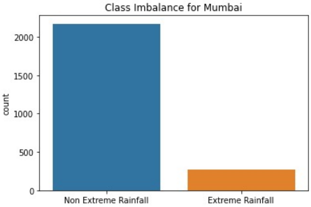
Such huge set of features if used for training a machine learning model may lead to overfitting. Therefore, we need a feature reduction technique.
Architecture of Auto-encoder -

- An input layer, (bias b = 9900)
- 2 hidden layers (initial units W = 2500, bias = 2500),
- An output layer (bias = 9900)
- Activation function - Sigmoid
- Optimizer - Adagrad
- Learning rate - 0.1
- Batch size - 100
- Cost function - MSE

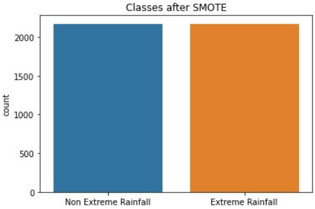
The size of encoded features comes out to be 2500 from initial 9900.

SMOTE

To effectively deal with the class imbalance problem in our biased dataset and to get the best performance, oversampling of minority class has been tackled using SMOTE.

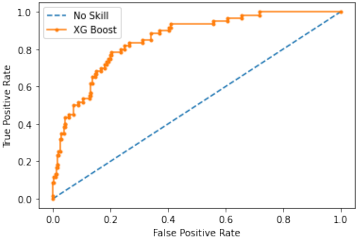


(a) Skewness in Mumbai Rainfall data

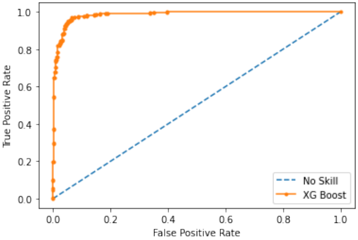


(b) After applying SMOTE

ROC Curves - Mumbai

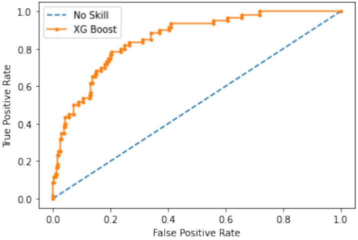


(a) Non SMOTE (0.853)

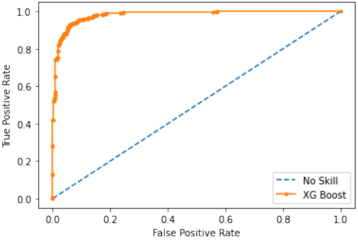


(b) After applying SMOTE (0.99)

ROC Curves - Guwahati



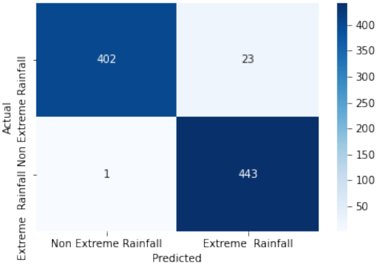
(a) Non SMOTE (0.854)



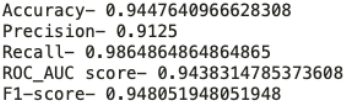
(b) After applying SMOTE (0.981)

Results - XGBoost

Auto-encoder + SMOTE + XGB (24 h, Mumbai) - High recall after dealing with all kinds of problems the data had originally, since Mumbai is one such city that receives heavy rainfall during summer monsoon months.



(a) Confusion Matrix



```
Accuracy- 0.9447640966628308
Precision- 0.9125
Recall- 0.9864864864864865
ROC_AUC score- 0.9438314785373608
F1-score- 0.948051948051948
```

(b) Metrics

Results - XGBoost

Auto-encoder + XGB (48 h, Mumbai) - Low recall because of class-imbalance. Also 24 h prediction performs better than 48 h.

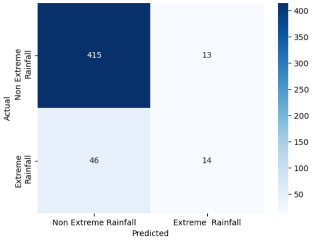


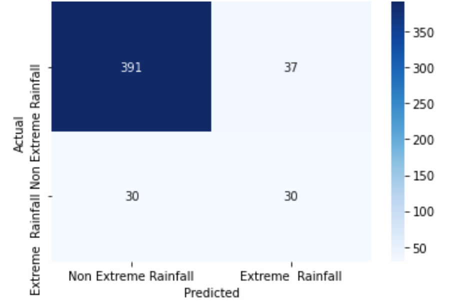
Figure: Confusion Matrix

Other Metrics

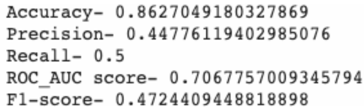
- Accuracy - 0.87
- Precision - 0.51
- Recall - 0.23, F1-score - 0.32

Results - XGBoost

XGB (24 h, Mumbai) - Low recall because no preprocessing done to balance the dataset.



(a) Confusion Matrix

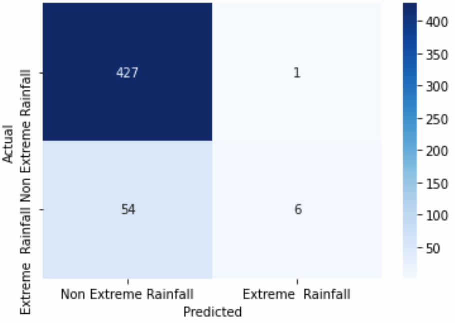


```
Accuracy- 0.8627049180327869
Precision- 0.44776119402985076
Recall- 0.5
ROC_AUC score- 0.7067757009345794
F1-score- 0.47244094488188898
```

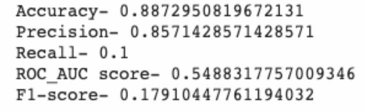
(b) Metrics

SVM

RBF kernel used.



(a) Confusion Matrix



```
Accuracy- 0.8872950819672131
Precision- 0.8571428571428571
Recall- 0.1
ROC_AUC score- 0.5488317757009346
F1-score- 0.17910447761194032
```

(b) Metrics

Discussion

In this work we have explored techniques to learn and represent weather features and use them to predict extreme rainfall events. This model works better because we include all the features and try to understand underlying patterns and dependencies unlike other approaches which rely on selective feature reduction.

Discussion

- For the country as a whole, the summer monsoon rainfall do not show any significant trend. However, there are large variations at the regional scale.
- Increasing parameters from 7 to 22 improved the quality of results to a great extent.
- 24 h prediction gave better results on average than 48 h prediction.
- SMOTE gave excellent results.
- For model without SMOTE, even though a cost-sensitive model was used, the performance of model degraded, especially for cities with low percentage of extreme rainfall days, and thus made the dataset more skewed.

External Links

Github -
https://github.com/Shruti-codes/Extreme-Rainfall_Prediction

Website -
https://shruti-codes.github.io/ExtremeRainfall/landingpage.html

Future Work

Here we have solved only a classification problem where we are only able to predict whether there will be heavy rainfall or not. In future we would also like to predict the amount of rainfall as well with the improved methods.