Design Credit Project

Report on

"Flood Early Warning System using Machine Learning Techniques"

Indian Institute of Technology

Jodhpur, India.

Guided By:

Dr. Saran Aadhar

Civil Engineering

IIT Jodhpur

Submitted By:-

Modi Saurabh Mehul

Mukul Shingwani

Sawan Sanjaykumar Patel

Sanidhya S. Johri

Jeevan Singh

Abstract

This paper reports our experience with **Early Flood Forecasting** using machine learning techniques. This surface runoff analysis includes water delineation of the stage data, data validation, Linear and LSTM Modeling, forecasting, and evaluation. Stage forecasting is modeled with the linear models ARIMA (Auto Regressive Integrated Moving Average) and SARIMA (Seasonal ARIMA) and Long Short-Term Memory (LSTM) networks. When evaluated on historical data, both models achieve sufficiently high-performance metrics on the respective evaluation metrics for operational use with LSTM showing higher skills than the Linear model.

Introduction and Literature Review

Floods are a major natural threat to populations worldwide, causing thousands of fatalities and resulting in large economic damages annually. Flooding vulnerability is especially significant in low-income and middle-income countries where adequate flood mitigation measures are often lacking or of limited quality and floodplains are often heavily populated. In such regions, operational flood warning systems are key to saving lives and reducing risks and damages.

Operational flood warning systems vary in their structure, data sources, and model types, depending on their specific target region, size of basins, available data and resources, and the system development approach. Many of the systems include real-time or

forecast weather forcing data as an input into hydrological models, which in turn compute runoff and route flow through the river network, outputting streamflow at locations of interest.

Two main modeling components are important for flood warning systems, these are (i) a streamflow forecast model and (ii) an inundation model. Recent advances in the accuracy of data-driven and machine learning (ML) methods have affected a large variety of real-life applications, and encourage the use of such methodologies as core drivers for those two modeling components. ML methods have been shown to be promising for flood inundation modeling as well, providing a plausible alternative to physically-based hydraulic models, that are both highly computationally demanding and challenging to use in operational flood forecasting systems.

This report is aimed to describe the implementation of the flood forecasting models that we took in our project.

Dataset

We were provided with datasets that could be broadly classified into three categories.

1. Stage Data: A daily time series for stage levels of different stations from 1965 to 2020.

2. Discharge Data: A daily time series for discharge levels of different stations from 1965 to 2020.

3. Forcing Data: This contains daily time series for Precipitation, Tmax, Tmin, and Wind Values for different stations.

The time series for some stations had a very large number of missing or NaN values. To overcome this problem, we filtered out stations that had data available for greater than 80% of the time series. From stage and discharge data, we filtered out 14 such stations.

These 14 stations were used for watershed delineation. For each watershed, we found latitude-longitude pairs for which we had forcing data available and used it in the predictions of the respective watershed.

Methodology

- The resources included temporal forcing data of 422 stations, stage data for 1007 stations out of which 14 data rich stations are having more than 80% non-null data.
- On stage data of 14 stations, data visualization of trend, seasonality, and residual analysis is done.
- Stationarity validation on the time series is done by using ADF test and KPSS test while Residual series test is done using Ljung-Box test.
- ARIMA parameters modeling is done using ACF and PACF plots while Auto Arima is trained using AIC and BIC loss functions.
- Stage Forecasting is done using ARIMA model and SARIMA trained model.
- Water delineation is carried out on these data-rich stations and the river basin of all 14 stations is made by upstream and downstream stream analysis.

- Out of 422 forcing stations, 400 stations lied out in the obtained watershed delineation.
- Sen slope analysis on temporal annual maximum stage and forcing data.
- Sen slope analysis on frequency of stage more than 95 percentile threshold annually.
- Application and forecasting using univariate unidirectional LSTM on average daily stage time data of delineated regions.
- Application and forecasting using univariate bidirectional LSTM on average daily stage time data of delineated regions.
- Application and forecasting using multivariate bidirectional LSTM on average daily stage time data of delineated regions.

Trend Analysis

A trend is a recurring pattern and trend analysis is the practice of collecting data in an attempt to spot that pattern.

When the data varies in quick succession of time, trend analysis becomes essential as it helps us as well as a model to understand what is the expected time for gains or drops or stability.

Advantages of Trend Analysis

1. Large sample sizes: The availability of data and online tools available to handle large amounts of data allow for the sampling of data quickly and applying the results to a variety of situations.

2. Verifiable: The results of trend analysis are easily verifiable.

3. Accurate: In case of statistical data, the analysis is very close to accurate. The use of numbers makes the analysis more exacting.

4. Replicable: A trend analysis can be replicated, verified, altered, and adjusted when necessary.

Disadvantages of Trend Analysis

1. Distortions: Historical data may not be an accurate representation of a trend. A random event or pattern could distort overall findings and render incorrect results for analysis.

2. Large sample sizes: For accurately and reliably analyzing a trend, a large amount of data needs to be collected. This is both a time-consuming and costly affair.

Trend, Seasonal and Residual Analysis

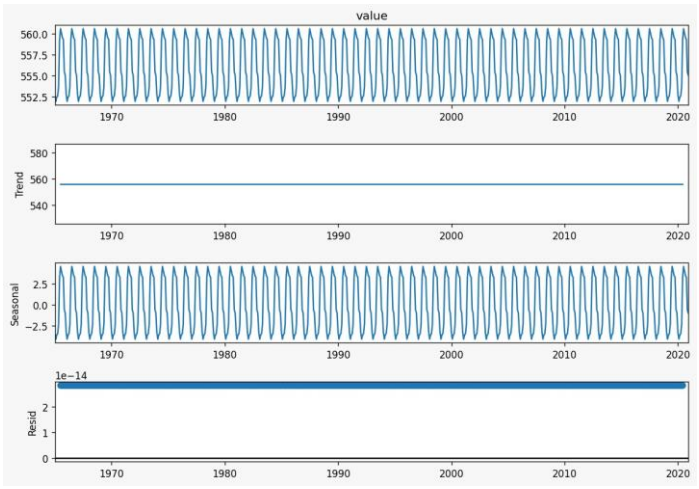


Fig: *Trend, seasonality and Residual plots for stage data of a station*

ARIMA Modeling:

The ARIMA model is well known as the Box-Jenkins method developed by Box and Jenkins. ARIMA model can be applied to a class of time-domain models commonly used for time series fitting and forecasting with a temporal correlation. ARIMA model incorporates d difference into the ARMA model and is designed for stationary time series. The ARIMA was implemented in the various fields for the forecast, e.g., monthly rainfall, streamflow, and water level. The study used an ARIMA model to forecast real-time road traffic data. Three components encompassing the general term for ARIMA (p, d, q) are used in ARIMA modeling. These three components autoregressive (AR), integrated (Differencing), and moving average (MA) are used in the respective order of p, d, and q. Differencing is a function that converts the non-stationary series into a stationary series while the AR and MA terms are determined through the temporal correlation of time series. The four phases of ARIMA modeling include model identification, parameter estimation, diagnostic checking, and prediction. ARIMA is distinct from others. It has the potential to recognize complex trends in temporary datasets and is thus widely

used for short-term predictions. The study shows that the efficiency of ARIMA in predicting either a linear or a non-linear series of intervals is satisfying. It is also a good option for predicting inter-valued time series.

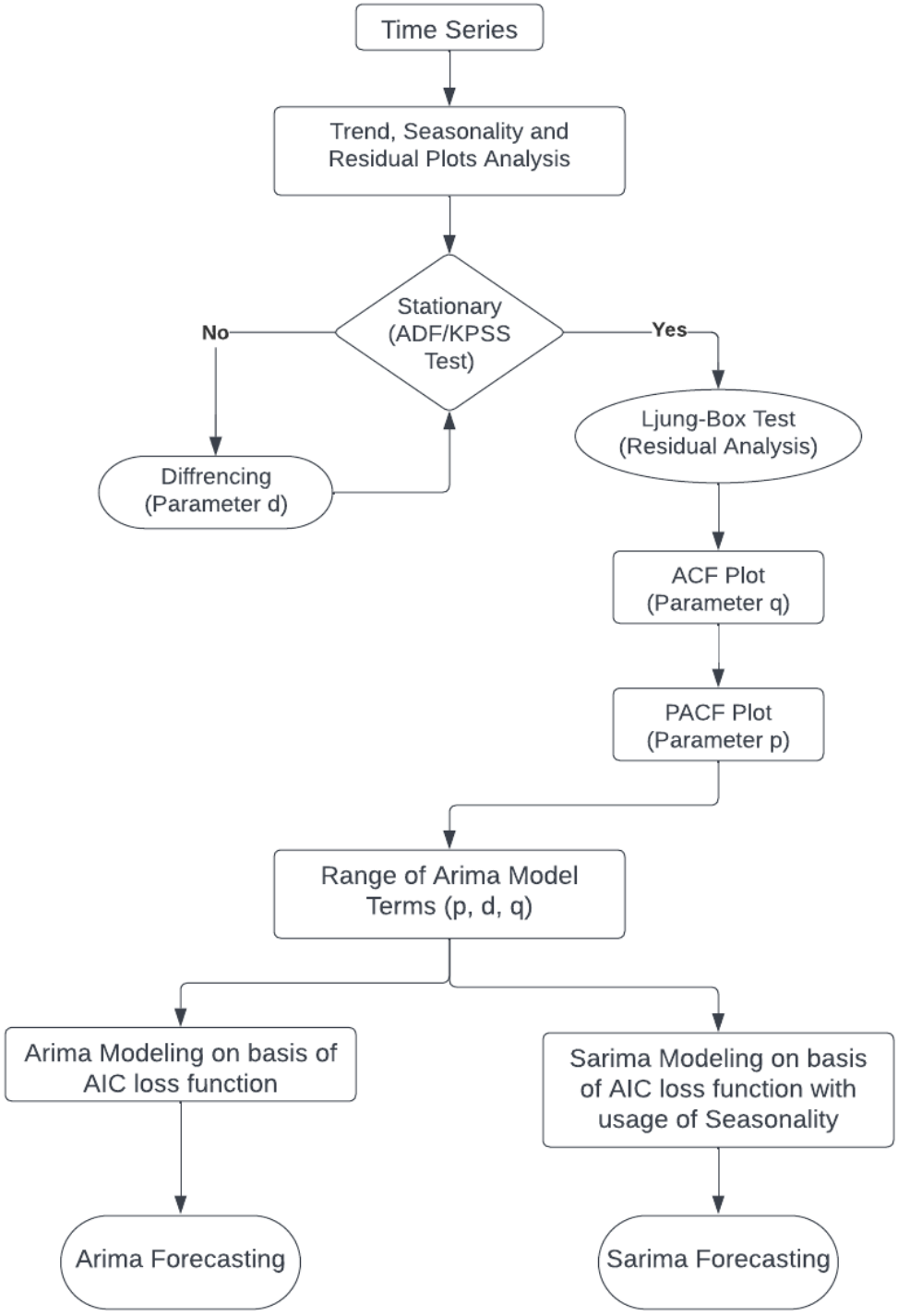


Fig: Box-Jenkins Method

Augmented Dickey-Fuller Test (ADF Test):

Augmented Dicky-Fuller test (ADF) is used to check the null hypothesis that a unit root is present in the time series. The function of this test is to check for the seasonal variation, variance and trend. The alternate hypothesis (H0) in ADF indicates that time series is trend-stationarity. Therefore, the null hypothesis (H1) in ADF indicates that the series is non-stationary. The significance level used for the P-value is 5% or 0.05. Therefore, if the series is non-stationary, differencing is needed to convert the non-stationary into stationary series. Hence, no differencing is needed if the series is stationary and the series can be modelled using the ARMA model. The ADF statistics are a negative number used in the analysis. The more negative the results, the greater will be the refusal of the null hypothesis that at some degree of confidence there is a unit root.

KPSS Test:

The Kwiatkowski-Phillips-Schmidt-Shin (KPSS) is based on linear regression and applied to statistically assess the stationarity of a time series. KPSS tests the null hypothesis that the observed time series is stationary and the alternative hypothesis is that there is a unit root in the time series. If the statistical properties of a time series change with time, it is said to have a unit root and thus it is non-stationary.

Ljung Box Test:

The Ljung-Box test is used to check the residual, that is whether it is in a random sequence of numbers. The Ljung-Box test is a test based on the null hypothesis, H0: The model does not exhibit a lack of fit, against the alternate hypothesis H1: The model exhibits a lack of fit[30]. To classify the presence of any structure in the observed sequence, the

model with a significant value of less than 5 percent or 0.05 for P was considered. Hence, the model was not accounted for. Therefore, if the model has a significant P value, it shows that the model exhibits a lack of fit. The residual of a model with a P-value of more than 0.05 is indicative of white noise and is considered to be an adequate model.

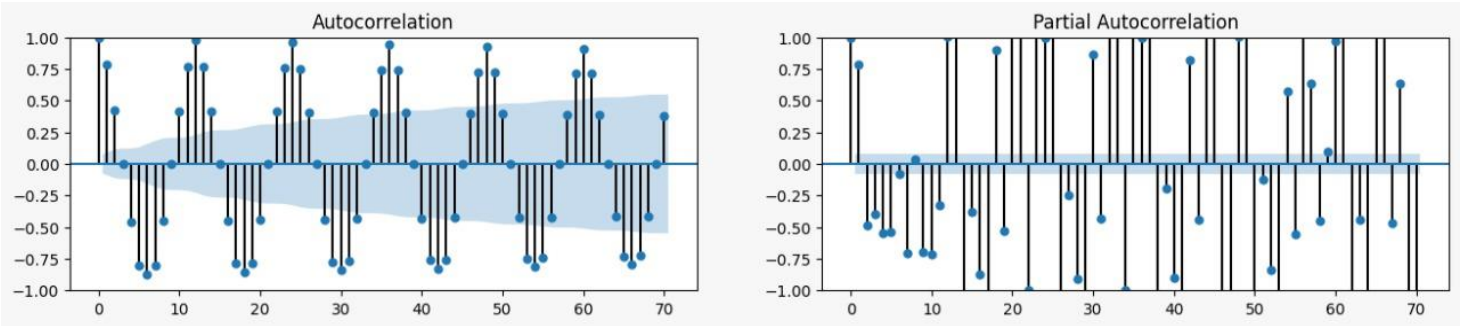
AIC and BIC:

The Akaike Information Information Criterion (AIC) and the Bayesian Information Criterion (BIC) are standards for evaluating the accuracy and goodness of statistical model fitting and efficient methods for assessing the p and q orders. The accuracy of the model can be determined by using the term Mean Absolute Percentage Error (MAPE)

$$AIC = 2k - 2\ln(L)$$
$$BIC = -2\ln(L) + k\ln(n)$$
$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{x_i - \bar{x}_i}{x_i}\right| * 100$$

shows the equation for the AIC, BIC equation, and MAPE. In the equation AIC and BIC, the L is the log-likelihood in the maximum value of the model, n is the sample size of the series and k is the number of parameters that are calculated in the model. Whereas for the equation MAPE, x_i is the actual value of the i-th, and ¯x_i is the forecast value of the i-th.

ACF and PACF Plots



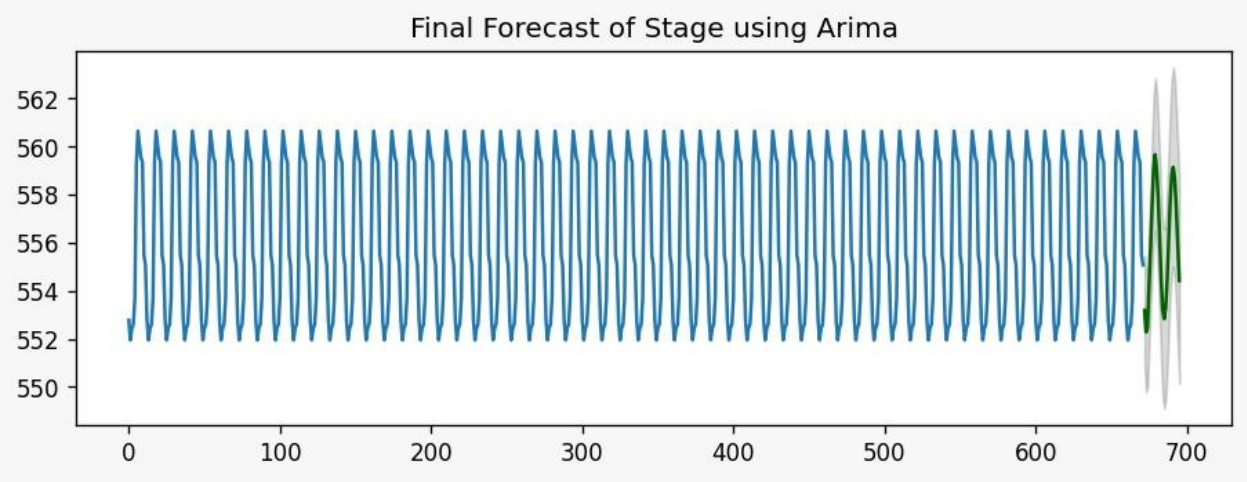


Fig: *Final Forecasting using ARIMA*

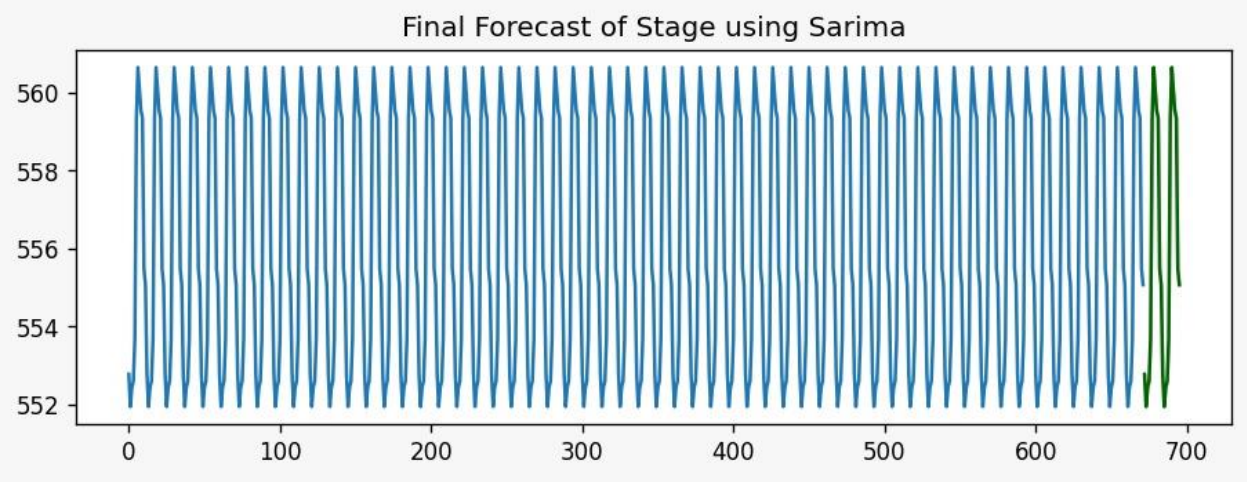


Fig: *Final Forecasting using SARIMA*

Watershed Delineation

In our dataset, there are 14 stations for which the data availability is greater than 80 percent. So we divided all other stations into 14 parts using watershed delineation considering 14 stations as the outlet points.

A watershed also called a basin, drainage, or catchment area is the land area that contributes runoff to an outlet point. The selection of the outlet

point determines a watershed boundary, and the process of defining the boundary of a watershed is known as watershed delineation.

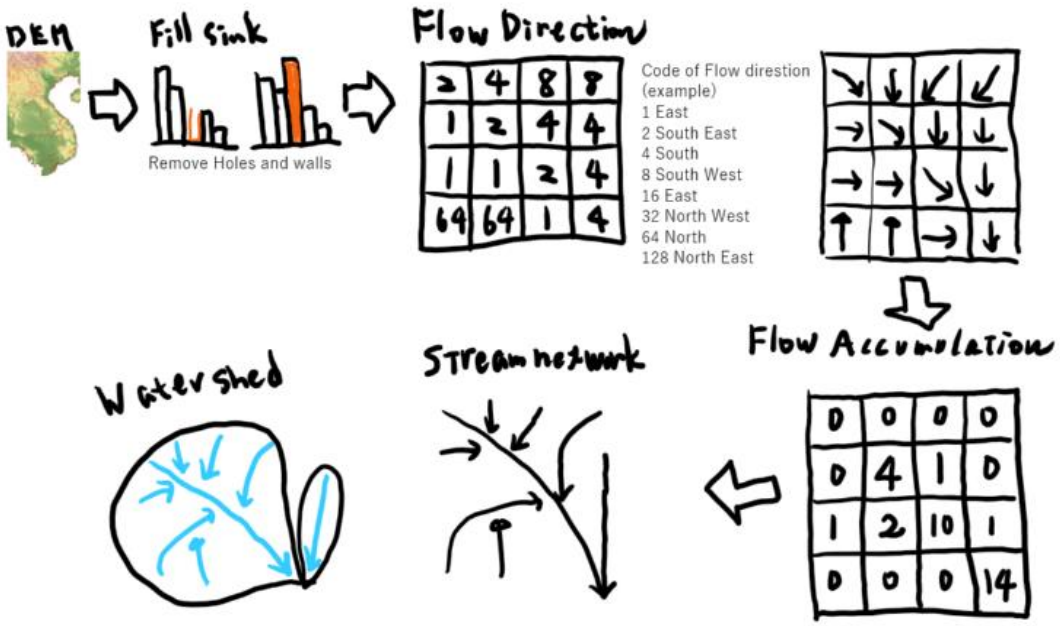


Fig: *Steps for performing Water Delineation*

Watershed can be generated from DEM. The fill sink is the preprocessing of DEM to remove holes and walls(Noise). Filled DEM is used to generate Flow direction which identifies water flows. Flow accumulation shows how much water comes into the grid. A large number shows big streams. Flow direction and Flow accumulation are used to generate a stream network and a watershed.

The following steps are involved in watershed delineation using QGIS:

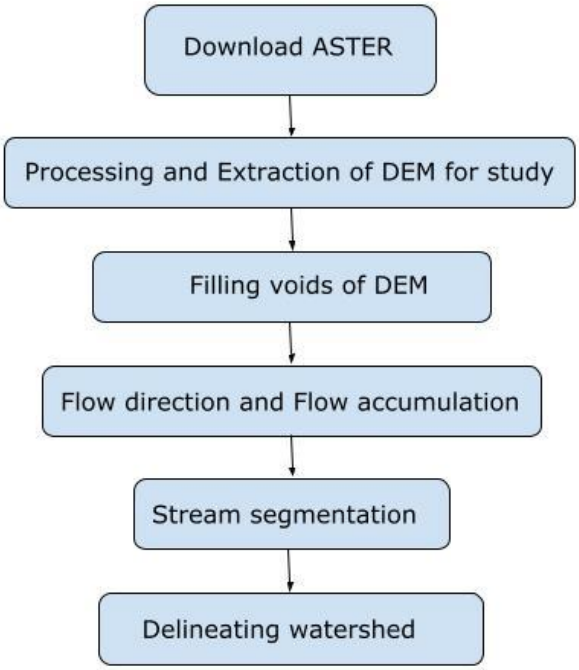


Fig: *Steps for Water Delineation*

The final result obtained after delineating the watershed at 14 outlet points is as shown in figure below.

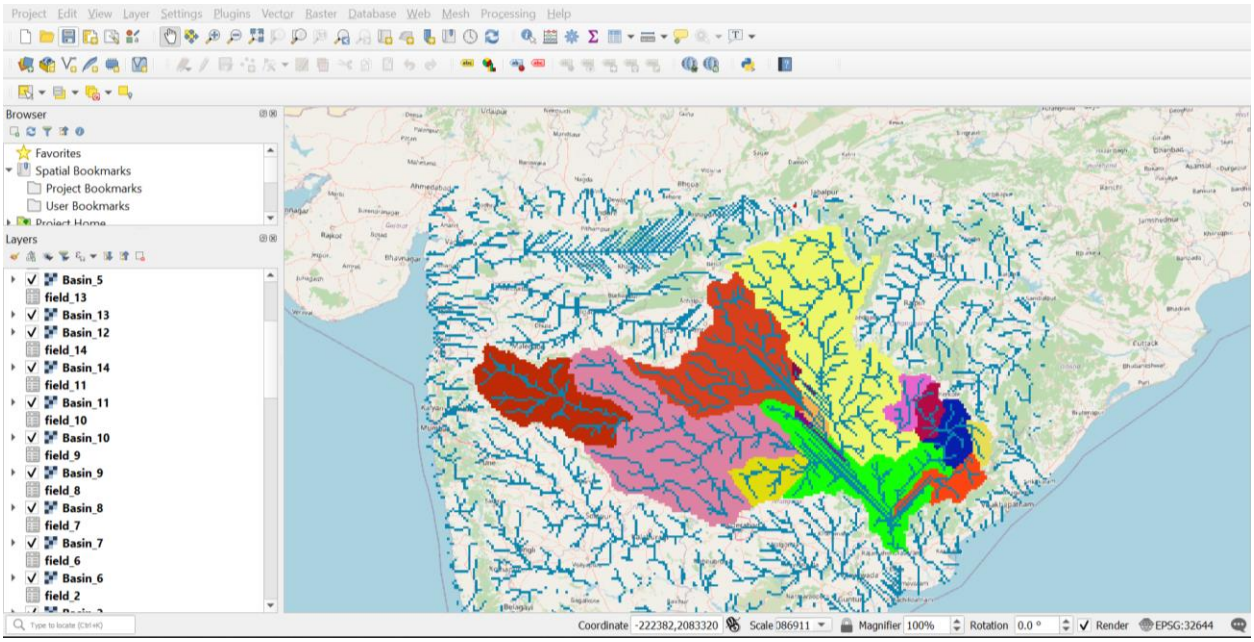


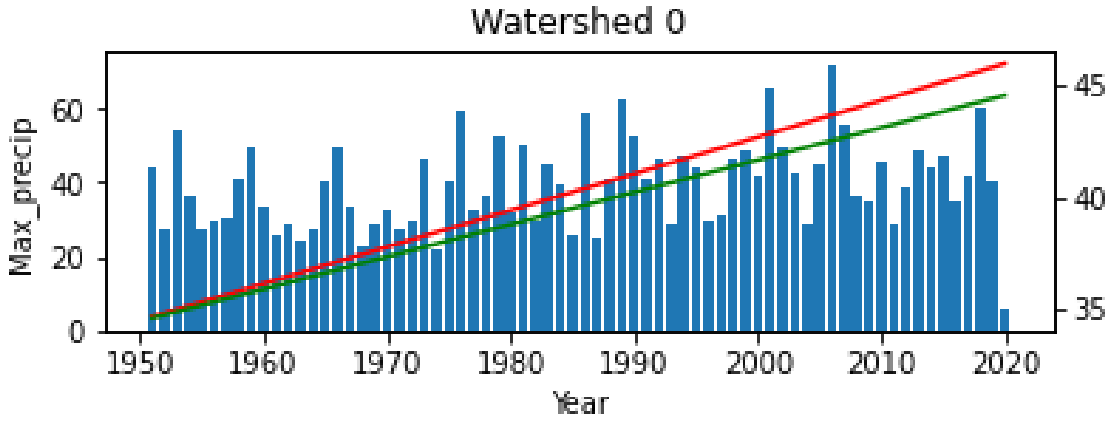
Fig: *Delineated Watershed Basin*

SEN'S SLOPE

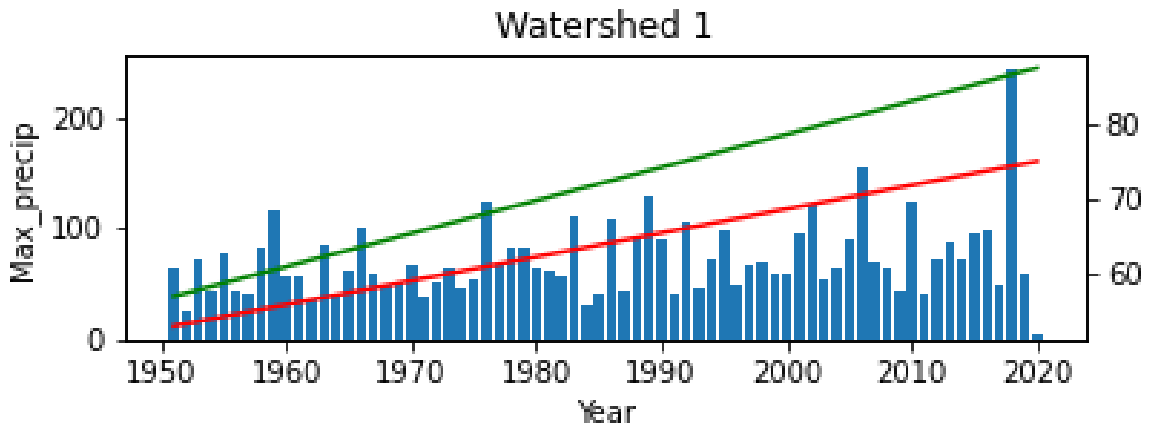
- Sens' slope estimator can be used to discover trends in univariate time series. It is fairly resistant to outliers.
- The method was first outlined by Theil and later expanded upon by Sen (1968), and is sometimes called the Theil-Sen estimator.
- The estimator is non-parametric, which means that it doesn't draw from any particular probability distribution. It is an alternative to the parametric least squares regression line.
- Where least-squares uses a weighted mean to estimate the slope, Sen's uses a median.
- Also, these were used to plot spatial plots.

PLOTS

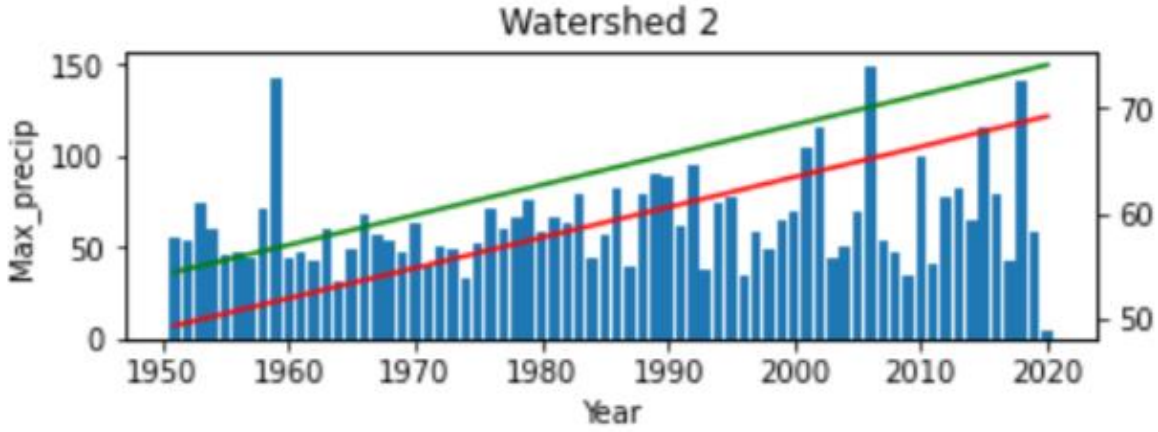
Annual maximum Stage data and the corresponding plot for Sen's Slope of the same.



Station Name: Nowrangpur, **Sen's Slope**: `0.14406343448007086`

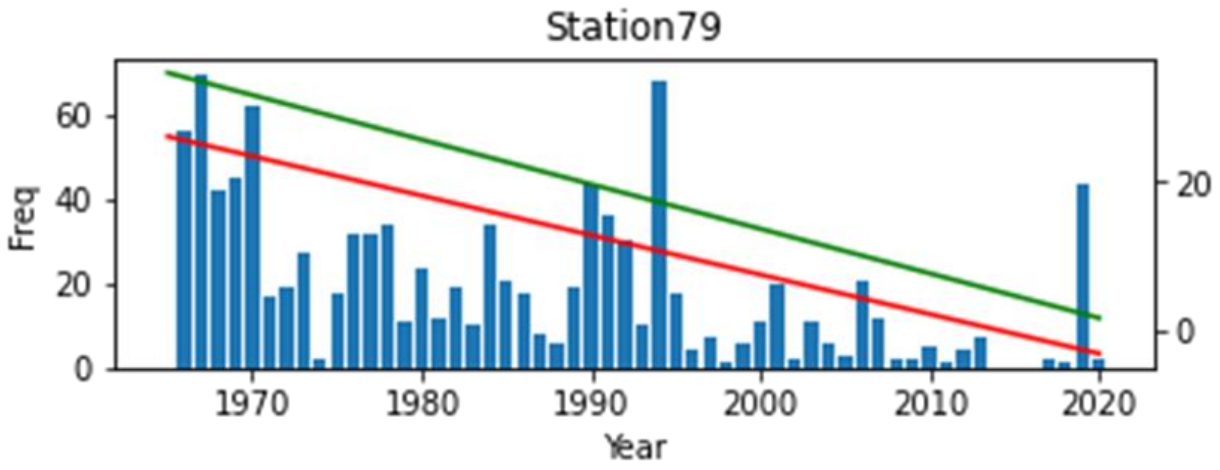


Station Name: Jagdalpur, **Sen's Slope**=0.4423926199224333

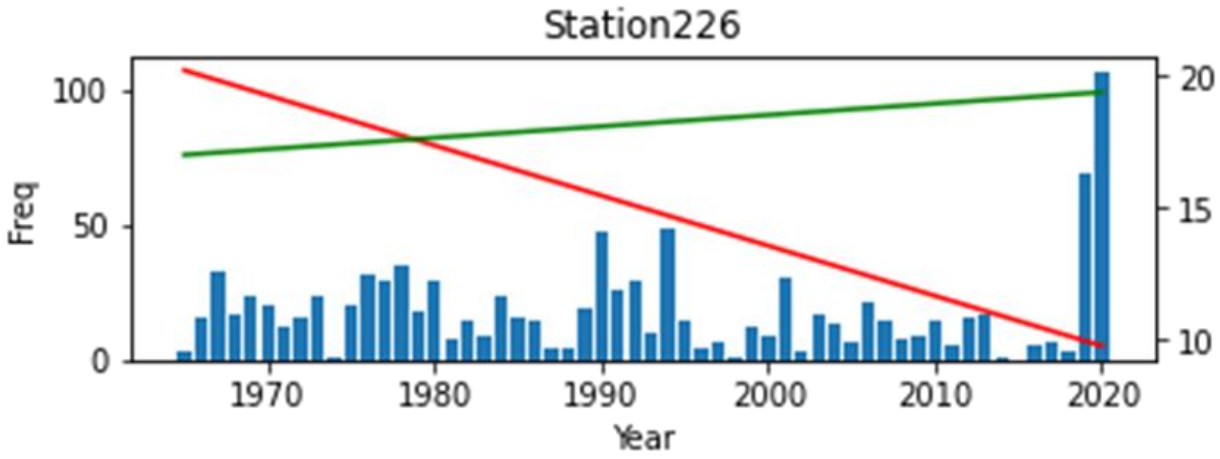


Station Name: Pathagudem, **Sen's Slope**: 0.28465108106027465

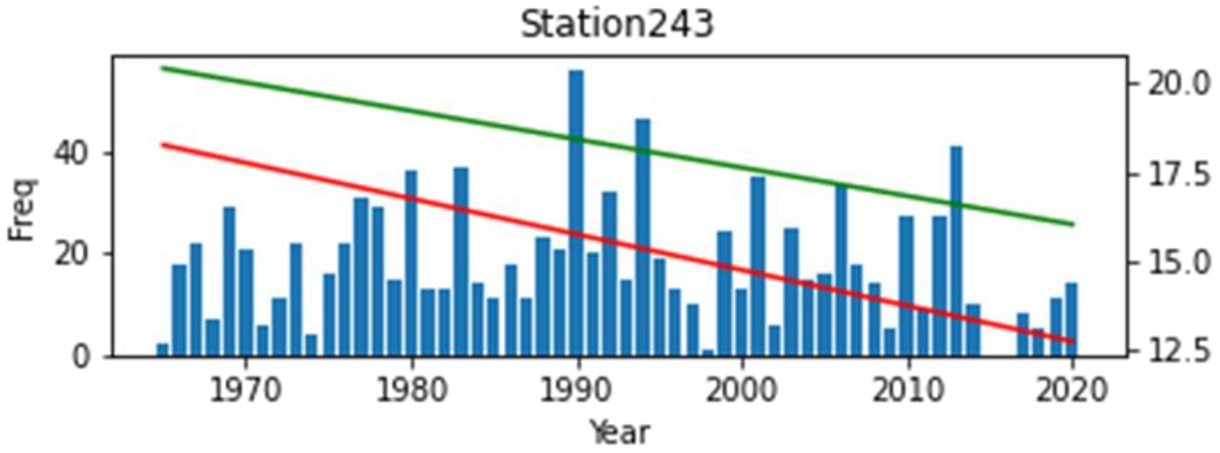
Annual Frequency of days when **stage** was greater than 95th percentile value and the corresponding plot for Sen's Slope of the same.



Station Name: Nowrangpur, **Sen's Slope**: `-0.5315824468085106`

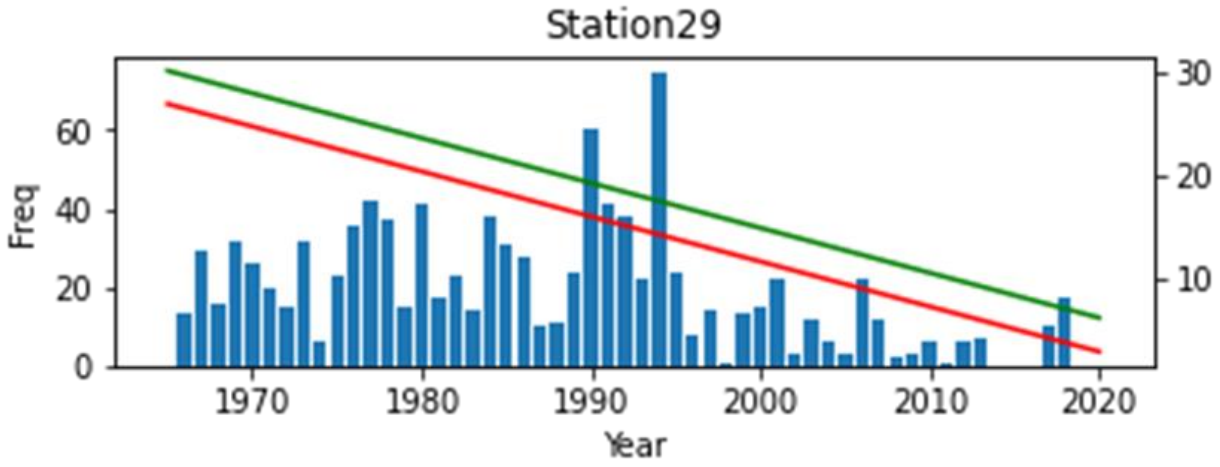


Station Name: Jagdalpur, **Sen's Slope**= `-0.19047619047619047`

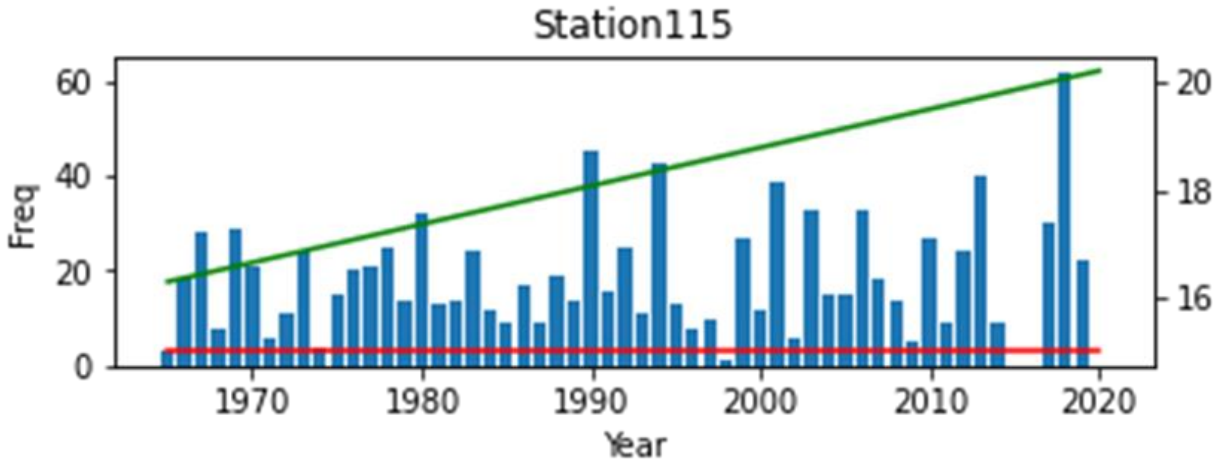


Station Name: Pathagudem, **Sen's Slope**: `-0.1`

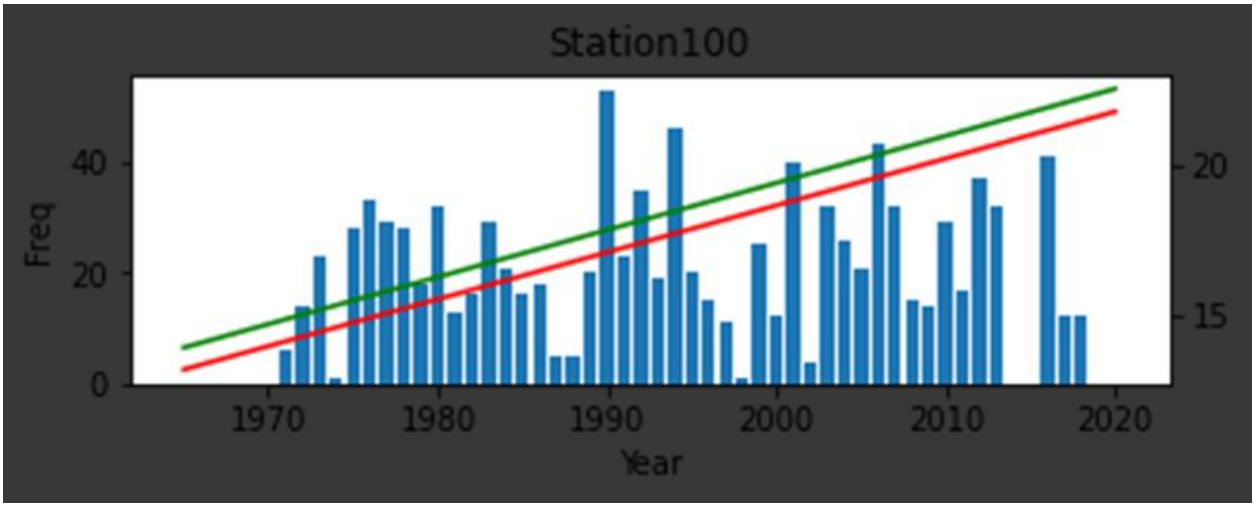
Annual Frequency of days when **discharge values** was greater than 95th percentile value and the corresponding plot for Sen's Slope of the same.



Station Name: Nowrangpur, **Sen's Slope**= `-0.43669871794871795`

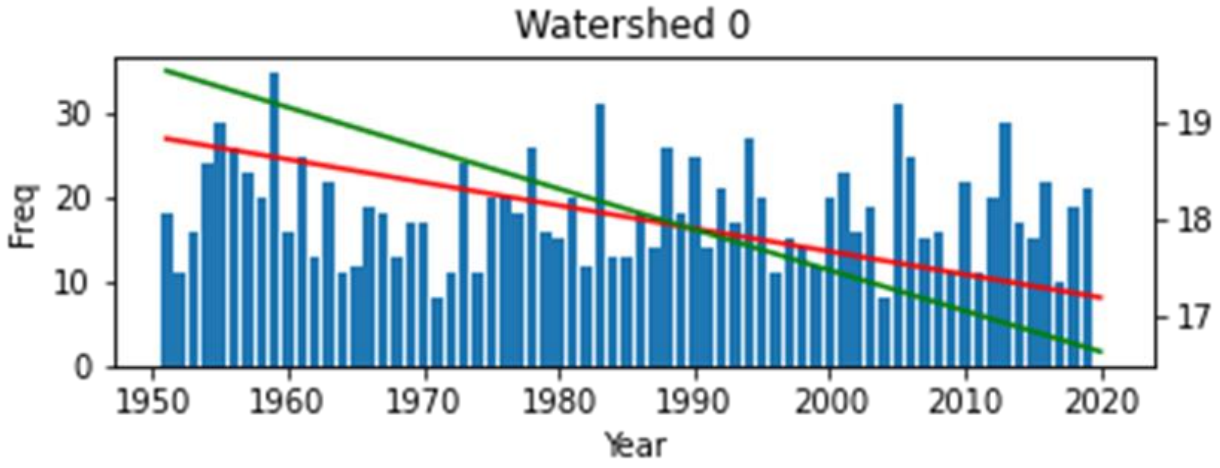


Station Name: Pathagedum, **Sen's Slope**= 0.0

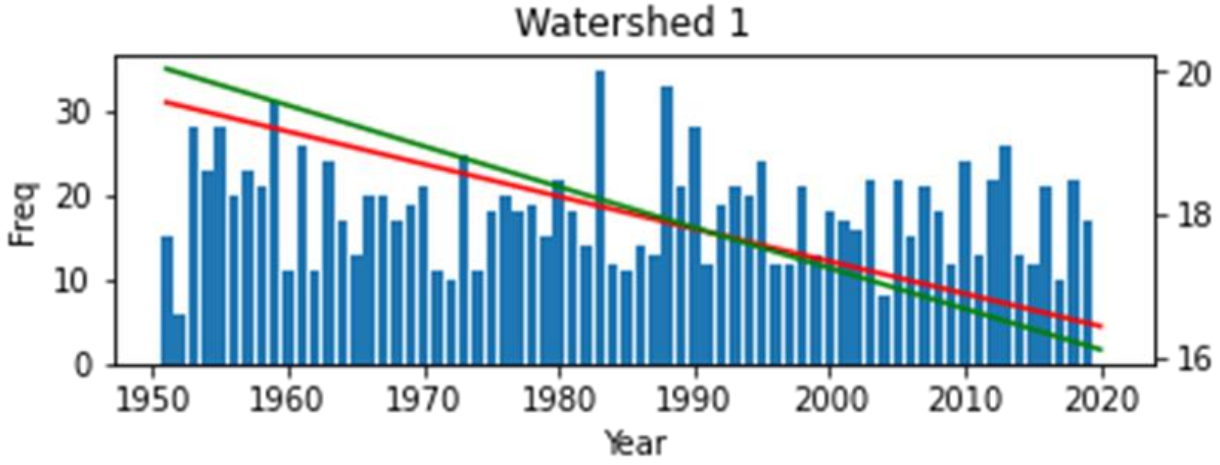


Station Name: Jagdalpur, **Sen's Slope**= 0.15707236842105263

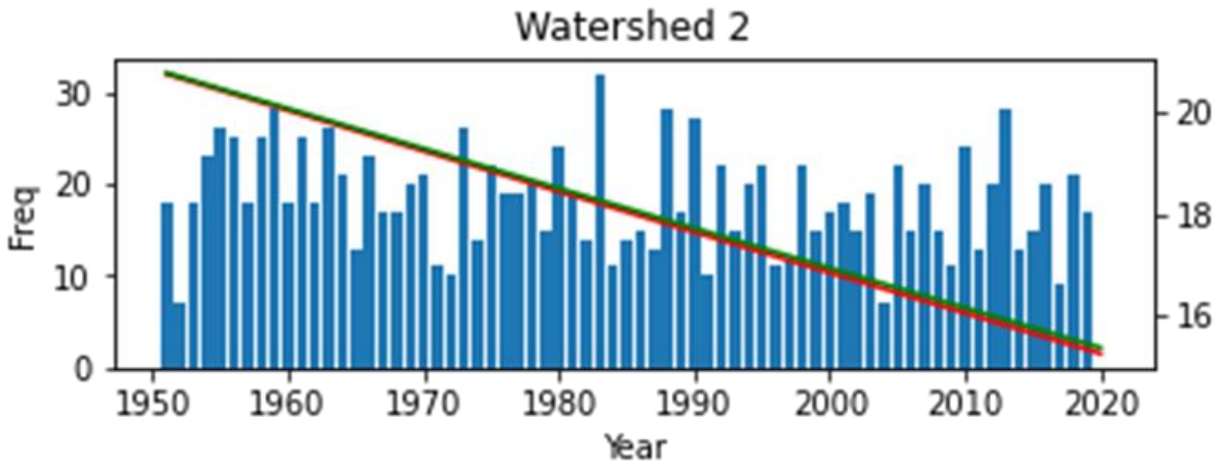
Sen slopes of average annual frequencies of precipitation values greater than 95th percentile value(for some watersheds)



Sen's Slope: `-0.023809523809523808`



Theil Value: `-0.045454545454545456`



Theil Value: `-0.08`

Long Short-Term Memory(LSTM)

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems. All recurrent neural networks have the form of a chain of repeating modules of a neural network. In standard RNNs, this repeating module will have a straightforward structure, such as a single tanh layer.

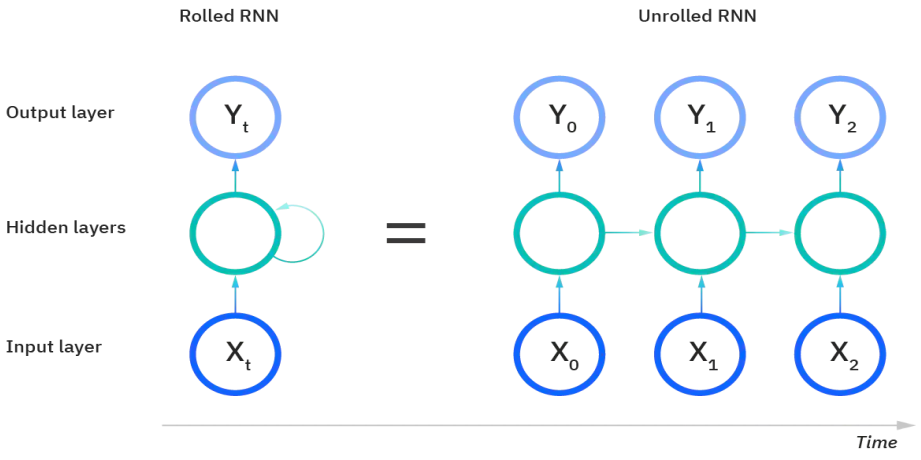


Fig: *Recurrent Neural Network (RNN)*

LSTMs also have this chain-like structure, but the repeating module has a different structure. Instead of having a single neural network layer, there are four, interacting in an extraordinary way.

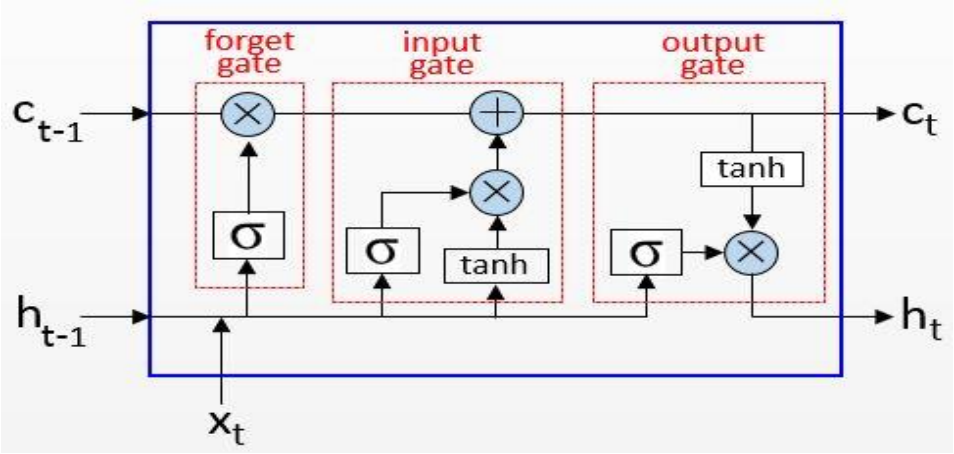


Fig: *Long Short Term Memory*

To cut short, using LSTMs we can decide which information to operate and forget using the forget-gate layer.

We started by applying the **Unidirectional LSTM** model to the daily stage data of one of the 14 watersheds (Jagdalpur station), after splitting the dataset into training and testing parts in an 80:20 ratio we got the following result:

RMSE	**MSE**	**MAE**
151.899	23073.183	45.590

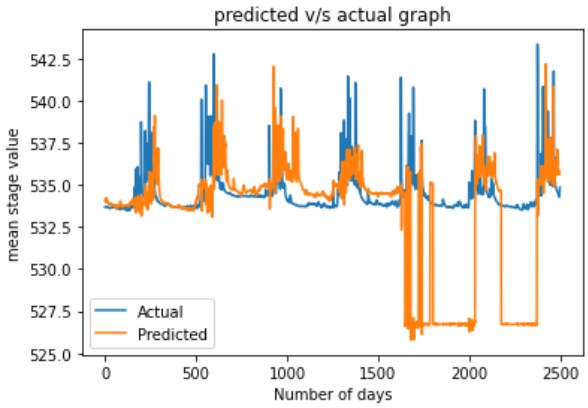


Fig: *Forecasting over 2500 days*

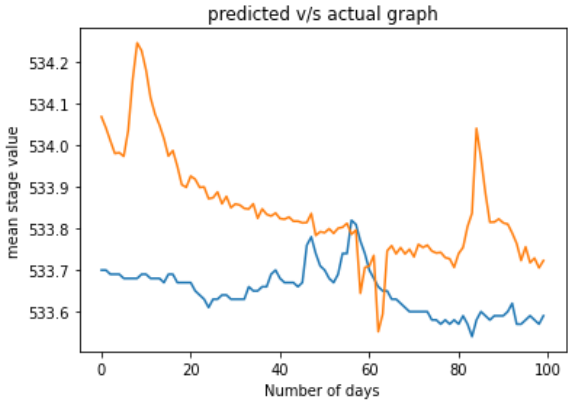


Fig: *Forecasting over 100 days*

We then decided to use the **Bi-directional LSTM** since we thought it can be beneficial to allow the LSTM model to learn the input sequence both forward and backward and concatenate both interpretations since the flood stage can rise and supposedly drop on the next day.

Station	RMSE	MSE	MAE	R2 score
Kothagudem	7.151	51.136	0.868	0.919
Nowrangpur	45.492	2069.487	3.983	0.909

Station Pathagudem plots for prediction v/s actual

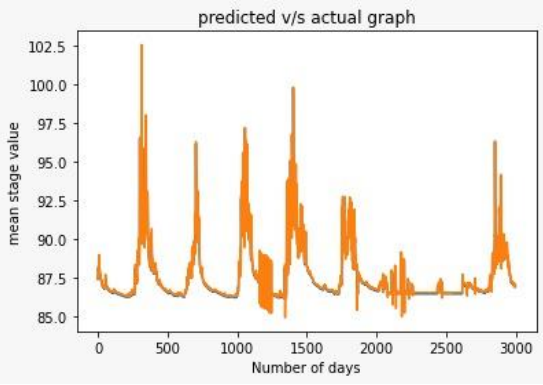


Fig: *Forecasting over 3000 days*

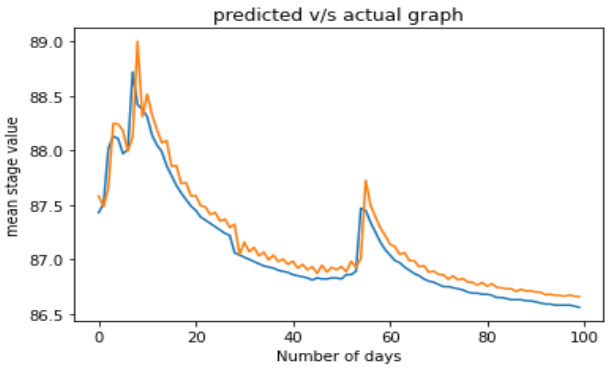


Fig: *Forecasting over 100 days*

Station Nowrangpur plots for prediction v/s actual

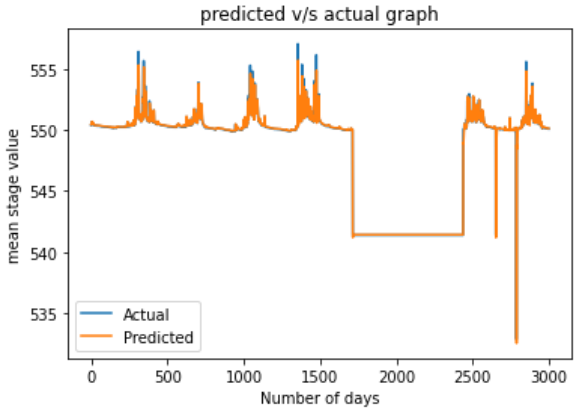


Fig: *Forecasting over 3000 days*

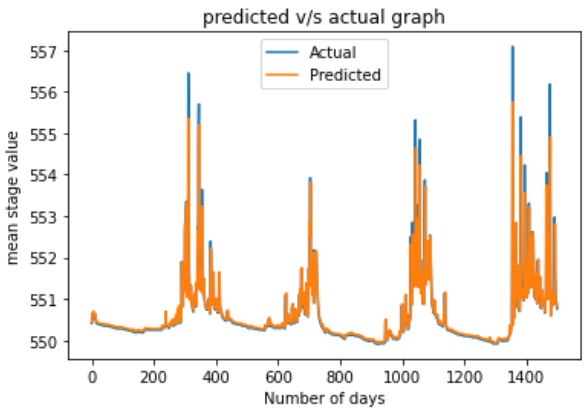


Fig: *Forecasting over 1500 days*

Conclusion

- When evaluated on historical data, both models achieve sufficiently high-performance metrics for operational use with LSTM showing higher skills than the Linear model.
- ARIMA is compared on the AIC loss matrix while LSTM is evaluated on RMSE, MAE, MSE, and R2 scores.
- ARIMA requires a series of parameters (p,q,d) which must be calculated based on data, while LSTM does not require setting such parameters. However, there are some hyperparameters we need to tune for LSTM.
- The work described in this paper advocates the benefits of applying deep learning-based algorithms and techniques to the time series-based data.

References

- **HESSD - Flood forecasting with machine learning models in an operational framework (copernicus.org)**
- **Flood Prediction using ARIMA Model in Sungai Melaka Malaysia**
- **Flood Predicting in Karkheh River Basin Using Stochastic ARIMA Model**
- **Temporal flood forecasting for trans-boundary Jhelum River**

of Greater Himalayas
- River-flood forecasting methods: the context of the Kelantan River in Malaysia
- Flood Forecasting Using Time Series Data Mining
- ANALYSIS OF DISCHARGE AND RAINFALL TIME SERIES IN THE REGION OF THE KLÁŠTORSKÉ LÚKY WETLAND IN SLOVAKIA
- Flood Forecasting via a Combination of Stochastic ARIMA Approach and Deterministic
- Flood Prediction Using ML Models
- An Overview of Flood Concepts, Challenges, and Future Directions
- ARIMA Model – Complete Guide to Time Series Forecasting in Python
- Hydrological analysis
- How to Develop LSTM Models for Time Series Forecasting