**Project Progress Report**

**Progress so far:**

* Project Topic: Prediction of Streamflow Using ARIMA Time Series.
* Data used: Streamflow data.
* Python program was utilized in data pre-processing and analysis.

**Completed tasks include:**

* Data cleaning – I ensured the date variable was converted to a date format. Also, I checked for duplicated rows and missing values. Furthermore, I imputed missing values found only on the streamflow rate variable (Q (m3/s)) using linear interpolation. I trimmed the data (filtered data from January 1967 to August 2017) to enhance data quality to build a good ARIMA model for forecasting. Data from September 2017 to December 2017 contained missing values.
* Dropping Variables – I dropped extraneous variables such as month, year, day, and Unnamed that will have not any impact on building the model.
* Indexed Streamflow data using the date variable.
* Converting daily records of streamflow to a monthly average streamflow.
* Splitting Data: After ensuring the data was complete and consistent, I split the data into train and test sets. The train data which consists of streamflow from the beginning of January 1967 to the end of December 2014 was used to build the model. The test data would be used to compare the predictions of the model.
* Implementing the Dickey-Fuller Test – I checked for stationarity of the train data using the Dickey-Fuller test. The P-value was less than 0.05 hence stationarity was confirmed, this is to ensure accurate modeling and forecasting.
* Decomposing the time series of the train data by plotting graphs to have an overview of the trends and seasons in the streamflow data.
* Plotting of the pacf and acf to find the value of the autoregressive term (p) and the moving average term (q) of the ARIMA model.
* Additionally, using the auto arima in python from the pmdarima library to find best ARIMA model with the train data.
* I used the ARIMA model to predict the streamflow in the timeframe as the test data. After, I compared the results of the predicted streamflow with the test data and evaluated the model using the prediction’s accuracy and root mean squared error.

**Challenges:**

* I had computation and memory challenges when analyzing the daily streamflow from 1967 to 2017.
* I also had the challenge of reading and deriving the values of the autoregressive term (p) and moving average term (q) from the pacf and acf plots respectively to build an efficient ARIMA model.

I addressed the above challenges as follows:

* Due to computation and memory challenges, I observed and analysed the average monthly streamflow rather than the daily streamflow.
* I used the ‘auto\_arima’ function from the pmdarima library to find the best ARIMA model containing the values of p and q.

**Next steps:**

* Next is to build a model with the library Prophet using the train data and compare its predicted data against the test data.
* I will then evaluate both the ARIMA and the prophet models by using their prediction’s accuracies and root mean squared errors.
* I do not see any potential challenges coming up in the project.