Modeling and forecasting water inflow time series: A comparative study of classical and deep learning techniques

By

Saira Baig

Department of Statistics

Faculty of Natural Sciences

Quaid-i-Azam University, Islamabad

2023

In the Name of Allah the Most Beneficent and Merciful

Modeling and forecasting water inflow time series: A comparative study of classical and deep learning techniques

By

Saira Baig

Supervised By

Dr. Ismail Shah

Department of Statistics

Faculty of Natural Sciences

Quaid-i-Azam University, Islamabad

2023

Declaration

I **"Saira Baig"** hereby solemnly declare that this thesis entitled "Modeling and Forecasting Water Inflow Time Series: A Comparative Study of Classical and Deep Learning Techniques," submitted by me for the partial fulfillment of Master of Philosophy in Statistics, is the original work and has not been submitted concomitantly or latterly to this or any other university for any other Degree.

Dated: ———————— Signature: ————————

Dedication

I am feeling pleased and great honor to dedicate this research work to

My Beloved Parents

Whose endless affection, prayers, and wishes have been a great source of comfort during my whole education period

and to

My Supervisor

who inspired hope, ignited the imagination, and instilled a love of learning.

Acknowledgement

First and foremost I praise and acknowledge Allah Almighty, the Lord and Creator of the Universe. All respect and gratitude goes to the Holy Prophet Hazrat Muhammad (Peace Be Upon Him) who enlightens our hearts with the light of Islam and whose way of life has been always a great guidance for us. I would like to express my heartiest gratitude to my respected supervisor **Dr. Ismail Shah** for his consistent support and supervision in every step of the research. I am indebted to his guidance specifically throughout my thesis. I am very privileged and blessed by his guidance. Many thanks to him. I offer my deepest sense of gratitude, profound respect, and tribute to all other teachers, Dr. Sajid Ali, Dr. Abdul Haq, Prof. Dr. Ijaz Hussain, Dr. Manzoor Khan who instilled within me the knowledge and guided me throughout my research work. My sincere thanks go to my parents**M.Latif Baig & Mrs.Latif Baig** and siblings **M.Sajid Baig**, **Urooj Baig**, **M.Nadeem Baig**, **M.Mobeen Baig**, and **Alina Baig** for their love and support throughout my life. Without their support and encouragement, it would have been impossible for me to complete this work. I would like to say thanks to all my friends, especially **Hiba Aftab**, **Noreen Kanwal**, **Nasir Iqbal**,**Iqra Arshad**,**Hina Tariq**, **Rimsha Bibi**, and **Huma Ghafoor**, and classmates for their cooperation and help.

Saria Baig

Contents

List of Figures

List of Tables

Abstract

Anticipating how rivers will behave is crucial in overseeing water resources, especially as the climate changes rapidly. This forecasting holds considerable economic importance, as it aids in managing water for farming, preventing water scarcity, and minimizing potential flood destruction. The major rivers within the Indus River system depend on the melting of snow and glaciers. Their water levels shift significantly during different times of the year. Researchers are engaged in a study to either examine or predict how much water will enter the Indus River. The input of water, known as inflow, plays a vital role in how we manage water resources. Therefore, being able to accurately predict this inflow is essential for effectively handling water resources. For inflow forecasting and modeling, we used a comparative study of classical and deep learning techniques. This study uses 5 years of data on Indus Tarbela Inflow ranging from January 2018 to September 2022. The data is collected from the Water and Power Development Authority (WAPDA). The first four years of the data is utilized for the estimation of the models and its subsequent one year is used for one-month-ahead out-of-sample forecast purpose. In this research work, we apply the, five different forecasting techniques that have been used for forecasting one-month-ahead INdus Tarbela Inflow. These include the AutoRegressive Integrated Moving Average (ARIMA), Seasonal AutoRegressive Integrated Moving Average(SARIMA), Autoregressive neural network (ARNN), Seasonal-Naive, and Long Short-Scond Memory (LSTM). Three error measures have been used for assessing the forecasting accuracy of the above models that includes, mean absolute percentage error (MAPE), root mean square error (RMSE), and mean absolute error (MAE). The findings indicate that the LSTM approach is effective in forecasting Indu's table inflow with lower forecasting measures.

List of Abbreviations

Abbreviation	Definition
ARMA	Autoregressive and Moving Average
ARIMA	Auto-Regressive Integrated Neural Network
SARIMA	Seasonal Auto-Regressive Integrated
ANN	Artificial Neural Network
ARNN	Auto-Regressive Neural Network
S-Naive	Seasonal Naive
LSTM	Long short-term memory
WAPDA	Water And Power Development Authority
MAPE	Mean Absolute Percentage Error
RMSE	Root Mean Square Error
MAE	Mean Absolute Error
ES	Exponential Smoothing
ACF	Auto-correlation Function
PACF	Partial Auto-correlation Function
ADF	Augmented Dickey-Fuller
AIC	Akaike Information Criterion
NSE	Nash-Sutcliffe efficiency criteria
TIC	Theil inequality coefficient
R2	coefficient of determination
PP	Phillips-Perron (PP)
TIC	Theil Inequality Coefficient
m^3/s	Cubic meters per second

Chapter 1

Introduction

Hydrology is the scientific study of the distribution, movement, and properties of water in the atmosphere and on the earth's surface. It includes investigating a range of water related phenomena, such as precipitation, evaporation, snow-melt, runoff, groundwater flow, and how water behaves in rivers, lakes and oceans. Forecasting future hydrological data uses hydrological time series analysis to identify prospective laws of hydrological process change based on existing knowledge (Nguyen et al., 2020).

Accurate forecasting of hydrology is becoming more and more necessary due to significant climate change in order to manage and plan water resources, especially hydroelectric projects. For optimal resource allocation, effective operations, and risk mitigation for water-related activities, the purpose of solving engineering issues, such as the building of flood protection structures for metropolitan areas and the design and development of agricultural accurate estimates are crucial, For these planning strategies the volume of input into the dam is a significant factor (Kim et al., 2022).

Some of the most well-known scientific problems in modern hydrology are climate variability, change, and its effects on the water regime. The unavoidable force of climate change, which is transforming our globe, has ushered in a new era of floods when the risk of flooding is amplified by changing weather patterns and rising temperatures. Since flooding is the most common natural disaster, forecasting at an early stage would prevent potential catastrophes and allow for the timely set up of rescue efforts.

However, structural (dams, reservoirs, and barrages) and non-structural (disaster prevention, response systems, and floodproofing) approaches can be used to study and control

water behavior. As water from dams and embankments is used for a variety of purposes. The dam, which performs numerous tasks like guaranteeing irrigation system stability, facilitating power generation, and enabling water distribution for community advancement, is a key component of water resource management. Planning and management of water resources are improved by anticipating inflow data (Musarat et al., 2021).

If water is released to the upper river, the lower streams will fill quickly and there are more chances of floods due to overflow Pradeepakumari and Srinivasu (2019). A methodical approach could prevent overflow and lessen the likelihood of flooding or diminish the overflow. High flow can be decreased by lakes, canals, reservoirs, and ponds. If predetermined canals are on the river they decrease the level of flow with the help of flood walls, filtering the canal can decrease the crest levels and the path of flood can prevent the outflow. To change the overflow of reservoirs or the crest flow of streams, systematic approaches should be used. If water inflow exceeds the barrages' predetermined limits, a flood may result, which could cause extensive damage. If flood information is available, accurate measurements can be made using frequency distributions for individual flood locations. Inflow levels have a significant role in flood forecasting. Pakistan is plagued by more than 150 natural disasters, most of which are floods. The flood year was triggered by a record-breaking amount of rain that fell during the monsoon season in 2010. River flow forecasting plays a critical role in water resource management.

According to Wang et al. (2018), river flow forecasting is crucial for flood control that occurs in basins, and hydropower systems. Since the dawn of human civilization, the flow of numerous rivers, including the Indus, has changed. In the early 20th century, the Indus river played a crucial part in the evolution of modern technology. An essential deadline for rivers is posed by industrialization and the associated human activity. Bridge construction is a crucial endeavor for the advancement of human civilization. The inherent traits and river structure are negatively impacted by these activities. For all of water needs, Pakistan firmly relies on the Indus river system. Construction of tube wells and dams has decreased the flow of the Indus river (Khan et al., 2021). Most of the time, it is unpredictable and uncertain how river water will behave. However, using organized procedures like dams, barrages, and rivers, water performance may be monitored and controlled. Using forecasting techniques, it is possible to calculate the river flow at a specific time based on historical data, which

can help with timely decisions and prevent natural disasters. Predicting a river's water flow directly relates to developmental issues.

Predicting a river's water flow directly relates to developmental issuesMusarat et al. (2021). The Indus river's primary branches are largely dependent on glaciers and snowmelt. These sub-rivers' flow varies greatly during the entire year. Due to the rainy season, there is a greater risk of flooding throughout the summer, and the water output is lower in the winter and higher in the middle of the year. The Indus and its sub-rivers receive water from hilly areas which is why there are more chances of overflow. On the other side, water is dropping into the river by monsoon rain. The level of the mainstream of the Indus is at its lowest in the winter season. After that, the level starts rising slowly until mid-march. The level starts overflowing in the mid-summer (Cook et al., 2013). Stochastic models like the ARIMA (AutoRegressive Integrated Moving Average) and ARMA (AutoRegressive Moving Average) models are commonly utilized in time series forecasting. These models find significant application in hydrology, particularly in predicting river flows (Adnan et al., 2017). The development of forecasting models like ARIMA, SARIMA, Seasonal naive, Neural networks, and LSTM in recent years has opened up new possibilities for improving the precision and dependability of predictions in hydrological systems.

1.1 Problem Statement

The Indus river contributes 90 percent of the country's food production, as it is full of natural resources that are useful for human beings in many aspects. The main water supply for the nation's irrigation systems is the Indus river. The Indus river has played a significant role in the modern world. Due to global warming and other factors, temperatures are rising, which causes glaciers to melt quickly and alters the flow of the Indus river every day. Due to the intense monsoon rains, river flow increases in the summer, which causes floods. Floods cause extensive damage and pose a threat to both human life and Pakistan's economy. To solve these issues, many strategies are used to forecast future observations or take preventive action for the Indus river to forecast its inflow in the future. The flow must be decreased by building dams, lakes, and sub-rivers. To do it, we utilize a variety of statistical methodologies to forecast the Indus flow.

1.2 Research Objectives

- To evaluate different statistical models forecasting abilities for Indus river inflow.
- To propose an efficient model to forecast the inflow of the Indus river.
- To compare different statistical techniques for forecasting problems.
- To recommend models that have superior forecast accuracy.

1.3 Contributions

- This study compared the parametric and non-parametric models to propose an efficient model for short-term ahead Indus river water inflow.
- In addition, a comprehensive literature review based on short-term ahead forecasting is also given in the work.

1.4 Thesis Overview

The thesis summary is organized as follows:

- **Chapter 2** presents a review of the literature on forecasting models and approaches for the Indus river, including traditional regression, time series, and artificial intelligence techniques.
- **Chapter 3** provides basic information about time series data and a detailed description of different types of statistical models used to forecast Indus river inflow.
- **Chapter 4** includes the application of statistical models to the Indus river inflow.
- **Chapter 5** contains results drawn from the study and provides concluding remarks.

Chapter 2

Literature Review

Adnan et al. (2017) used two time series models to forecast the monthly streamflow of Doyian station: the Autoregressive Moving Average (ARMA) model and the Autoregressive Integrated Moving Average (ARIMA). This study employed monthly streamflow data from 1974 to 2010. The models were trained with data from the first 28 years, and forecasting was done with data from the last 7 years. Time series model accuracy in forecasting is measured by comparing the root mean square error (RMSE), mean absolute percentage error (MAPE), and Nash efficiency (NE). Because it allows time series to become stationary, the ARIMA model outperforms the ARMA time series models in forecasting and training.

To predict monthly discharge at Hit station on the Euphrates River in Iraq, (Shathir and Saleh, 2016) evaluates seven ARIMA family models. 480 observations were examined between October 1932 and September 1972 using IBM SPSS statistics 21. Statistical tests having a 95 percent significant probability, such as the T-test and F-test, were used to detect changes in mean and variance. The model with the lowest error and the best agreement between observed and anticipated discharge was (2,0,1)(0,1,1).

Pini et al. (2020) conducted a study that employed many machine learning approaches and tried to predict water input to Lake Como in Italy. For various days, one-day to three-day forecasts are given. Three statistical measures are used to evaluate these models: MAE, RMSE, and Nash-Sutcliffe Efficiency Factor. The experimental results reveal that artificial Neural Network (ANN) outperforms Support Vector Regression and Random Forest for streamflow prediction with MAE and RMSE and that ANN outperforms the other models, which may be due to ANN's ability to learn the non-linear pattern of the data.

For the planning and operation of water resource systems, particularly hydrologic components, future event forecasting is essential. Kisi and Kerem Cigizoglu (2007) used ANN technology to examine both long- and short-term continuous and irregular daily streamflow forecasting approaches. Three ANN techniques were used to analyze continuous and intermittent river flow data from two Turkish rivers: feed-forward back propagation (FFBP), generalized regression neural networks (GRNN), and radial basis function-based neural networks (RBF). The ANN training data was successfully prepared using the k-fold partitioning method. In terms of performance criteria, RBF beats other ANN approaches and time series models. However, FFBP had limitations, such as local minimum problems and negative flow generation.

Valipour et al. (2013) examined models such as ARMA, ARIMA, dynamic ANN, and static autoregressive artificial neural network, the research intends to anticipate the inflow of the Dez dam reservoir. The model was trained using 42 years of statistics, and it was forecasted using the last 5 years. In the buried layer, 17 neurons were used, as well as radial and sigmoid activity functions. In forecasting inflow, the dynamic ANN model with sigmoid activation component outperformed the static model. The ARIMA model outperformed the ARMA model because it converts time series to stationary data. Static and dynamic ANN with an activity sigmoid activation function, on the other hand, forecasted input from previous 60 months.

Reza et al. (2017) studied three stations in Malaysia's Bukit Merah watershed, the evaluation assessed the efficacy of both linear and non-linear methodologies for modeling time series data. Based on MAPE, RMSE, and R2, the performance evaluated. The findings demonstrated that streamflow estimation performed well by using both ARIMA and ANN approaches but ANN performs superior with short-memory data and is more adaptable to inconsistent data, and that ARIMA is appropriate for long-term time series analysis. Additionally, ANN is more adaptable and adept at recognizing data patterns than ARIMA.

In the study, Mohammadi et al. (2005) anticipated spring inflow to the Amir Kabir reservoir of Iranian Karaj river basin in the study. Techniques such as ANN, ARIMA time series, and regression analysis were used. Models were trained or calibrated on 25 years of observable data before being tested for 5 years. Three criteria were used to assess the forecast model's performance: average percentage error, average seasonal deviation, and RMS error

between observed and calculated inflows. The results revealed that the models' respective correlation coefficients for RA1, RA2, RA3, ARIMA, and ANN were 0.545, 0.844, 0.711, 0.475, and 0.891 during the verification period. The ANN model outperformed other models in terms of errors for data spanning thirty years.

Fashae et al. (2019) compared ANN and ARIMA models to estimate river Opeki discharge from 1982 to 2010. The best predictor was then used to forecast River Opeki discharge from 2010 to 2020. The correlation coefficient (r), RMSE, and projected data were used to assess the efficacy of the two models (ARIMA and ANN). While ANN's coefficient of correlation was 0.93 and its RMSE was 15.06, ARIMA's was 0.97 and its RMSE was only 0.57. The results revealed that, when compared to the ANN model, the ARIMA model appeared to outperform it. When the RMSE was looked at in the near run, ARIMA beat the ANN model. The study findings revealed that the ARIMA model outperformed the ANN model, especially when other parameters (such as meteorological data) were scarce.

Sultana and Sharma (2018) conducted a study on Swine flu, which is a respiratory illness that affects pigs' respiratory systems as well as other bodily functions. It was brought on by influenza viruses. This study used a range of time series forecasting techniques, including Box-Cox transformation, exponential smoothing, seasonal naïve, and neural networks, to estimate future Swine flu incidence in India. The applied models outcomes were compared using errors such as Mean Error, Mean Absolute Error, Root Mean Square Error, Mean Absolute Scaled Error, and Auto Correlation Function. Data from the integrated disease surveillance program were collected from 2010 to 2017. When they analyzed the final data, they discovered that the neural network forecasting model offered the best result among the others, with an accuracy of 98.4%.

Using historical data from 1961 to 2017, Katušić et al. (2022) analyzed the precision of eight data-driven approaches for forecasting future weather patterns in central Croatia. Seasonally naive, ARIMA, Error-Trend-Seasonality (ETS), Exponentially Smoothed State Space Model with Box-Cox Transformation (TBATS), DHR, NNAR, SVR, and LSTM were all included in the evaluation. In terms of accuracy in forecasting, the results showed that SVR is the best technique, followed by DHR and NNAR. While NNAR forecasts precipitation better, DHR forecasts temperature and air pressure better. Furthermore, incorporating oscillation indices as additional predictors improved the prediction accuracy of SVR, DHR,

and NNAR methods.

Pala et al. (2019) used a dataset comprised of 36-month EMF readings. In addition to the mean, naive, seasonal naive, drift, STLF, and TBATS basic models, more sophisticated ANN models like as NNETAR, MLP, and ELM were used for forecasting in the R software environment. To measure the accuracy of the models, metrics such as RMAE and MAE were used. When both MAE and RMAE performance measurements were combined, Seasonal Naive from the standard functions performed best; for neural network functions, the NNETAR function performed best. NNETAR, MLP, and ELM are examples of ANN algorithms with lower RMAE average values than the other traditional methods. This result demonstrated that ANN algorithms outperformed conventional methods when the dataset is divided for training and testing operations by 84% and 16%, respectively, as opposed to 70% and 30% Overall, the top performance values were provided by the NNETAR, Seasonal Naive, MLP, STLF, TBATS, and ELM models.

To increase forecast accuracy,Cheng et al. (2015) suggested a hybrid forecasting system that combined SVM and ANN. The approach forecasts reservoir monthly inflow data using ANN and SVM, with the processed predictive values chosen as input variables for more accurate forecasting. The monthly inflow projections of the Xinfengjiang reservoir were analyzed from 1944 to 2014 using the models: ANN, SVM, and the hybrid method. The hybrid technique outperformed ANN and SVM on five statistical variables, making it a suitable tool for reservoir dispatching and long-term operation.

Yadav and Sharma (2018), provided numerous forecasting tactics for the Bombay Stock Exchange's SENSEX (also called the BSE 30 or simply the SENSEX), BSE SENSEX using forecasting models such as ARIMA, BoxCox, Exponential Smoothing, Mean Forecasting, Naive, Seasonal Naive, and Neural Network, and then compared their mean errors to identify the most effective approach. On the Bombay Stock Exchange's (BSE) SENSEX, the analysis was conducted. When the mean error of the two models was compared to the mean error of the other models, the findings of this study showed that exponential smoothing and neural networks offer the best results.

Kabbilawsh et al. (2022) compared four univariate time-series forecasting methods for predicting rainfall time series in Kerala, India: HK-SARIMA, NSTF, YJNSTF, and SN approach. The difference in rainfall features and the usefulness of the Yeo-Johnson trans-

formation in enhancing forecast accuracy were examined using the rainfall time series of 18 stations in Kerala, India, from 1981 to 2013. To evaluate the effectiveness of each model, the following three error statistics were computed: RMSE, MAE, and NSE. In the Western lowlands and Eastern highlands, respectively, models HK-SARIMA and YJNSTF performed admirably. Eight of the twelve stations in the Central Midlands had favorable performance indicators for the HK-SARIMA model. so the conclusion was that the HK-SARIMA models were more accurate at predicting the monthly rainfall at the stations distributed across Kerala's various geographical regions.

The River flow was predicted by Tadesse and Dinka (2017) using GRETL statistical software from 1960 to 2016. Stationarity was confirmed using unit root and Mann-Kendall trend analysis. Different SARIMA models were compared based on seasonal differences in correlogram properties. For predicting river flow, the SARIMA (3,0,2)(3,1,3)12 model was chosen since it performed the best because it had low values of AI and HQ and had a pattern that was similar to actual mean monthly flows. The information offered to water resource managers and decision-makers facilitates the development and management of the Waterval River and Vaal Dam reservoir in the Olifant basin. Future research should compare SARIMA model forecast accuracy to that of computational intelligent forecasting approaches.

The ARIMA model with seasonal parameters was used by Selvi et al. (2019) to anticipate inflow series at the Palar-Porandalar dam in Tamil Nadu. The prediction and modeling process used the dam's monthly inflow data from 2003 January to 2017 December as the data source. The stationarity of the data set was confirmed using Mann-Kendall's trend test along with additional stationarity tests. Different models were found and their parameters were optimised using the Correlogram display. The residuals were then diagnostically assessed using the Autocorrelation plot and Ljung Box test. The best model was chosen based on the lowest AIC, BIC, RMSE, and Theil's U statistic values. The statistic value of the SARIMA (0, 0, 1) (1, 0, 2)12, value of 'U' was 0.8497*, which was less than one, indicating that the model was more accurate in predicting future behavior. As a result, the SARIMA (0, 0, 1) (1, 0, 2)12 model was best one to use for forecasting.

Joshi and Tyagi (2021) studied the seasonal Naive, seasonal triple exponential smoothing, and seasonal ARIMA models which were used to estimate rainfall in Bengaluru, Karnataka, India, using monthly data from January 2009 to December 2018. To compare the forecast

accuracy of these models, various measurements based on forecast errors and residual plots were used. Seasonal ARIMA surpassed the seasonal Nave and seasonal Holt-Winter's models, and ARIMA (0, 0, 2)(1, 1, 1)12 was the best-fitting SARIMA model, according to the empirical results. The best SARIMA model was also used to estimate rainfall in Bengaluru for the next three years (2019, 2020, and 2021), and it showed that rainfall was likely to decrease in most months, with the exception of May and June, over those three years. This was not good news for the government of Karnataka because Bengaluru already faces severe water shortage issues. Making rainwater collection plans, urban water management plans and other pluralistic water resource related tasks could all benefit from this study.

For one-month-ahead flow forecasting at the Kalabeili gauging station in Xinjiang, China, Abudu et al. (2010) examined the performance of time series and Jordan-Elman ANN models. When employing earlier flow conditions as predictors, the ARIMA and SARIMA models performed similarly to Jordan-Elman ANN models. It made sense to choose ARIMA modeling for improved water and environmental management since it was simple to use and accurately captures the stochastic nature of streamflow processes. The effectiveness of ANN and time series models, however, might be attributed to the use of solely prior flows as predictors. While ANN models might easily incorporate different factors, time series modeling might be difficult to do with additional predictive variables like snowfall, precipitation, and temperature data. The models in this study were only applicable to the study basin and because the forecasting models utilized in this work were limited to the study basin and these specific conditions, more research was needed to uncover predictors and improve precision.

Mohamed (2021) studied monthly flow at Malakal station, South Sudan, and was predicted and modeled using SARIMA linear stochastic models. Forecasting monthly streamflow for the White Nile River at Malakal station was critical for Sudanese and South Sudanese water resources projects, such as the Jabal al Awliya dam operation. The investigation relied on monthly flow data spanning the years 1970 to 2013. The original series analysis demonstrated an annual seasonal pattern. The results of the PP and ADF tests on the flow of water series show that it was non-stationary. Before constructing the model, the non-stationarity was eliminated by applying first-order seasonal differencing (with a twelve-month interval). Forecast accuracy was assessed using RMSE, MAE, R2, NSE, and the Theil inequality coefficient (TIC). The model had been found to have reduced MAE and RMSE values. The

R2 score of 0.869 and the NSE value of 85.3% indicated the model's performance. The value of TIC was discovered to be 0.048, indicating that it was an exact match. The actual flow values were discovered to be very close to predicted values. Finally, this demonstrated that the previously found SARIMA (1,0,1)(0,1,1)12 model was adequate.

Gelažanskas and Gamage (2015) examined data on household hot water usage was analyzed. To create a demand-side control plan depending on individual residential property projections. They investigated several forecasting methods, including seasonal decomposition, SARIMA, and ES. These models surpassed the traditional models (mean, naive, and seasonal naive), and their performance measurement values were greater. The findings demonstrated that the accuracy of forecasting was significantly influenced by seasonal decomposition. Strong daily and weekly usage trends were found in the study's analysis of aggregate data and water consumption profiles for ninety-five homes. Forecasts for the next 24 hours were generated by using approximated exponential smoothing, ARIMA, and seasonal decomposition models. "STL and ETS(A,N,N)" and "STL and ARIMA(p,d,q)" were the models that perform best.

Forecasting stream flows for miniature rivers is essential but difficult due to their lower volume. Hu et al. (2020) used LSTM, a deep learning model, to present a unique solution for time-series data. They obtained precipitation data from 11 sites and stream flow data from a station in Tunxi, China, to anticipate 6 hours in the future. They assessed the accuracy of the predictions using RMSE, MAE, and R2 measures. The LSTM model outperformed the SVR and MLP models, with an RMSE of value equal to 82.007, a 27.752 value of MAE, and an R2 of 0.970. To examine the influences on the effectiveness of the LSTM model, the researchers ran extensive tests.

Xu et al. (2020) evaluated the effectiveness of LSTM networks in the Hun and Upper Yangtze river basins, concentrating on daily and 10-day average flow projections. The study demonstrated that because of precipitation and flow magnitudes, the non-linear transformation was required to improve efficiency; however, utilizing a completely linked layer with an activation function could lower learning efficiency. To balance learning efficiency and stability, batch size and LSTM cell number should be carefully adjusted. The LSTM network beats many hydrological models in terms of Nash-Sutcliffe Efficiency, coefficient of determination, and relative error, demonstrating its effectiveness in learning difficult hydrological modeling

processes.

Fang et al. (2021) used a sequence-to-sequence prediction model to present a multi-zone indoor temperature forecast model LSTM-based method to save energy in buildings with accurate indoor temperature forecasting while occupant comfort was not sacrificed. For multi-step forecasting, a sequence-to-sequence (seq2seq) model based on LSTM was developed. Several criteria were employed to evaluate the model's ability to forecast out-of-sample, and a custom score was provided to take into account unique features of indoor temperature with minor daily variations. The model performed better in short-term forecasting tests as compared to Prophet and seasonal Nave models. A cross-series methodology for learning was utilized for multi-zone indoor temperature estimation, and intervals for predictions using the Monte-Carlo dropout technique were used to quantify parameter uncertainty.

Costa Silva et al. (2021) suggested an ensemble method for predicting water flow at the Jirau Hydroelectric Power Plant (HPP) in Brazil using recurrent neural networks. The model's predictive power was evaluated in terms of RMSE and MAE, and its performance was contrasted with that of individual LSTM models and the statistical model employed by the Jirau HPP. The effectiveness of the examined models was compared in five different situations. In four out of five instances, the ensemble LSTM model beat the statistical model and demonstrated greater accuracy than other individual LSTM models. This method could be utilized to work with the Jirau HPP for effective energy production and control and was promising for water flow predictions based on river tributaries.

In this study Atashi et al. (2022) examined time series forecasting's accuracy as well as focusing on water levels for flood warning systems. It examined SARIMA, RF, and LSTM as three flood forecasting methodologies for the Red River of the Northern and discovered that the LSTM approach outperformed SARIMA and RF methods in terms of results and accuracy of prediction performance. The findings demonstrated that SARIMA was efficient for modeling nonlinear data whereas LSTM was more accurate when modelling linear data. LSTM surpassed RF and SARIMA in all prediction times, with RMSE values that were 77.22% and 78.70% lower, respectively, according to experimental results. In terms of LSTM model, the RMSE difference between the RF and SARIMA estimated had been greatly reduced at the Drayton and Grand Forks stations.

In order to predict discharge at the East Branch of the Delaware River up to seven days

in advance, Mehedi et al. (2022) developed an LSTM neural network regression model. A comparative analysis using predictions from CNN and MLP were also offered to evaluate the LSTM's performance. The overall distribution of the projected values of discharge from LSTM, CNN, and MLP were substantially equal to the observed data, indicating that all three methods functioned satisfactorily. However, when it came to predicting discharge based on historical data, the LSTM strategy performed better than the CNN and MLP-based systems. The LSTM algorithm's performance indicator, RMSE, was 151.52 ft3/s, compared to CNN and MLP's 235.74 ft3/s and 489.07 ft3/s, respectively, and a MAPE value of 0.92%, 2.17%, and 2.95%. The model's performance could be enhanced by repeating iterations or increasing the number of epochs. However, after 100 epochs, LSTM surpassed all other algorithms with the least amount of error.

Parasyris et al. (2022), produced Meteorological predictions by employing techniques like SARIMA, as well as AI approaches like LSTM neural networks and even hybrid amalgamations of both methodologies. These approaches used a range of meteorological variables, temperature, relative humidity, etc, unlike the singular focus of SARIMA. These factors were divided into those that are seasonal and those that have stochastic behavior, such as air direction and velocity. To assess the predictive forecast capability, established methods like climatological forecasts and persistence models were employed as benchmarks. Among the methods considered, the hybrid approach excelled particularly in predicting temperature and wind speed, with SARIMA following suit. For humidity forecasts, LSTM outperformed the rest, a trend that holds even after refinement.

Many applications require local temperature forecasts for up to 24 hours. The SARIMA model, or, to put it another way, the naïve prediction, is commonly used to generate these projections. Kreuzer et al. (2020) investigated whether deep neural networks could outperform the outcomes of the aforementioned techniques. In addition to univariate LSTM networks, They presented an alternative technique based on a 2D-convolutional LSTM network. To benchmark their technique, built a case study using data from five different weather stations in Germany. After performing admirably for the first few hours, the multivariate LSTM network and convolutional, for longer perspective forecasting, the LSTM network outperformed the SARIMA model, seasonal nave model, and univariate LSTM networks. Both multivariate methods function more effectively when the temperature fluctuated during the

day. Their proposed technique, which was based on a convolutional LSTM system, fared the best overall on all of the test data sets analyzed.

According to Belvederesi et al. (2022), river flow forecasting models aid in the understanding, prediction, and management of surface-water resource problems like flooding and declining water quality. However, because of seasonal and annual variations, predicting could be difficult in cold climates. Researchers used regionalization, geographical calibration, interpolation, and regression techniques to enhance forecasting effectiveness. Process-based models were more accurate, but the data collection and calibration parameters were costly and time-consuming. Efforts to overcome the lack of data availability, user-friendly interfaces, and standardization of calibration and validation dataset choices were highlighted in Canadian studies.

Chapter 3

Methodology

3.1 Case Study Description

The Indus River is one of Pakistan's longest rivers. The river connects the boundaries of four countries including Pakistan, China, India, and Afghanistan. Indus River has Five main tributaries, Jhelum, Chenab, Sutlaj, Kabul, and Ravi. The first tributary is the Kabul River which is located at Mithan-Kot where Jhelum and Indus river meets. Jhelum, Chenab, Ravi, and Sutlaj meet at the head of Panjnad, a single river after the head Punjanad travels alone and drops into the river Indus. Two-thirds of water for basic human necessities or for irrigation comes from the Indus River and its sub-rivers. Many sub river drop into the Indus River, Kabul is one of the leading rivers that drops in the Indus with its sub-river including Swat and Panjkora. There are so many barrages located on the Indus river and one of the largest barrages is Sukkur Barrage. These barrages make the largest hydropower project that produces 1450MW of electricity. Overflow at the river Indus affects the environment and agricultural activities. Three main water reservoirs are located at the Indus River like Tarbela, Mangla, and Chashma.We are primarily concerned with the Indus Tarbela Dam. Tarbela Dam, located on Indus River in Khyber Pakhtunkhwa (KP), is one of the world's largest earth-fill dams, some 130 kilometres north of Islamabad. The Dam was built in 1970 and finished in 1974 by Pakistan's WAPDA, with irrigation for downstream Indus plains, flood management, and low-cost hydroelectric power generation as its main goals. Pakistan's resources are greatly impacted by Tarbela Dam, which provides 52% of irrigation releases and 30% of the country's energy demands. At full capacity, the dam generates a reservoir

that is over 100 kilometres long and 260 km square. However, decades of flooding and erosion have reduced its initial storage of 11.9 billion cubic metres to 6.8 billion cubic metres. The main dam structure, which measures 2743 metres in length and 148 metres in height, is built of soil and rock fill. The left bank of the river is connected to the island by adjacent concrete auxiliary dams Ishfaque et al. (2022). Dam has two spillway structures at its auxiliary dams. The main spillway has a capacity of 18,406 m/s while the auxiliary spillway has a capacity of 24,070 m/s. Notably, instead of being used to produce hydroelectric power, more than 70% of the water released via the dam flows over these spillways. The World Bank and the Asian Infrastructure Investment Bank are backing fifth Tarbela Dam extension project. The goal of this project was to raise the hydroelectric capacity of dam from 4888 MW to 6298 MW.

3.2 Data Description:

The flow rate of Pakistan's Indus River is the study's primary focus. The inflow rate of the Indus Tarbela is one of the variables being studied. The inflow rate of data from 2018 to 2022 has been taken from Water and Power Development Authority(WAPDA) for the purpose of this investigation.

Every day, real-time monitoring data is collected to ensure the smooth running of Pakistan's Tarbela dam. The Indus River's inflow is measured in Cusecs*1000, the dam's outflow is measured in Cusecs*1000, and the barrage level is measured in feet daily using real-time sensor data. The Cusecs stands for "cubic feet per second," and 1 Cusec = 28.32 Litres.

There are 169,650 km^2 in the Indus basin upstream of Tarbela Dam. Between the Great Karakoram and Himalayan ranges, where more than 90% of the land is located, meltwaters from these mountains play a vital role in annual flow that flows into Tarbela. In the remaining area of basin, which is situated immediately upstream of the dam, monsoon rains frequently occur during the months of July, August, and September. The discharge of monsoon rains, which generates severe floods of short duration, delays snowmelt discharge. Tarbela receives an average of 81,000 Mcm per year (TAMS 1998). The yearly runoff variability in the Indus is fairly minimal because a substantial portion of its runoff originates from snowfall. Peak snowmelt volumes can range between 5,660 and 11,300 m3/s, with additional rainfall

frequently adding up to a maximum of 5,660 m3/s. Due to this Pakistan has faced floods and other disasters events so many times, which caused many losses to livelihood and economy. It is vital to work with hydrological data with different statistical and machine-learning techniques to predict uncertain events that can occur in the future.

3.3 Box Jenkins Methodology

Box Jenkins Methodology is a repetitive procedure that makes an ARIMA model for seasonal and trend factor, measure accurate weighting parameters and test the model or repeat the process accurately. Box-Jenkins method was designated for making simulation and forecasting tools that depend on The capacity of the method to deal with complicated situations, its flexibility in rectifying dependent time series, its beneficial statistical and mathematical processes, in case of any risk its programmability is effective, and the most important one is that its implementation is very simple or we can say its easy to use.

For any data set, Box-Jenkins provides an appropriate forecasting model. Its methodology also provides a structured approach for creating, interpreting and forecasting time series models. This methodology takes recent observation as a starting value and then estimates forecasting error for future prediction.

Only in stationary time series Box-Jenkins method can be applied. A series that does not have any seasonality or trend pattern is known as a stationary series. Most of the time the data we use is nonstationary, first we apply different transformation methods to make them stationary (Lu and AbouRizk (2009)). The box-Jenkins method recommends short and long(seasonal) to attain stationarity in mean, and to attain stationarity in variance logarithmic or power transformations are applied. Nelson and Plosser (1982) suggest that some of the series give good results with differencing, and some provide better results with linear detrending. Most of the time, the series we are using is nonstationary, so it is difficult to apply any nonstationary series, Box-Jenkins ARIMA modeling provides a widely used approach to take the differences of a non-stationary series to make them stationary. Then AR, MA, or ARMA models can easily fit into the series. If there is seasonality in the series the Box-Jenkins suggest seasonal models with long-term (seasonal) differencing, if we require stationarity in the mean.

3.4 Autoregressive Integrated Moving Average Model

ARIMA (Auto-Regressive Integrated Moving Average) is a statistical model employed for forecasting time series. It is a class of models that integrates the ideas of auto-regression (AR) and moving average (MA) models with the idea of integration (I) to handle nonstationary time series data. The ARIMA model is utilized for identifying patterns in the time series data and then using these patterns to make predictions about future values. Following are the assumptions of the ARIMA model (Lee and Ko, 2011).

- **Stationarity:** Time series data are presumed to be stationary by ARIMA, which means that the mean, variance, and covariance remain constant across time and are unaffected by the series' location. Data that is not stationary must be changed to become stationary.

- **Auto-correlation:** ARIMA makes the assumption that there is some auto-correlation in the time series data, which means that the residuals (the difference between the forecasted value and the actual value.) are associated over time. The ARIMA model uses auto-correlation, which can be either positive or negative, to improve predictions.

3.4.1 Parameters of ARIMA

Three parameters characterize ARIMA models: (p,d,q).

- p is the order of the autoregression component.

- d is the order of differentiation

- q is the order of the moving average component.

3.4.2 Autoregressive Model

Time series data has always been associated with its past values.The autoregressive process as their name suggests, is regression on themselves. AR model specifies that current values are set by their previous values. If the present values depend instantly on previous values, it is known as AR model. In other words, a model in which independent variables are lags of

dependent variables or dependent variables are regressed by their own values (Shrestha and Bhatta, 2018)

Specifically, a pth order auto-regressive processes y_t satifies the equation:

$$Y_t = \varphi_1 Y_{t-1} + \varphi_2 Y_{t-2} + \ldots + \varphi_p Y_{t-p} + \epsilon_t$$

The series' present value yt is the linear sum of the p's latest previous values of itself plus an "innovation" term $epsilon_t$ that integrates anything new in the series at time t that cannot be explained by the prior values. For each t, we suppose that $epsilon_t$ is independent of $yt-1$, $yt-2$, $yt-3$, and so on. Yule (1926) pioneered work on auto-regressive processes.

3.4.3 Integrated Process

A first-order integrated process is expressed in the following manner:

$$Y_t = Y_{t-1} + \epsilon_t \tag{3.1}$$

Difference order 1 indicates that the difference between two consecutive values of Y is constant, where ϵ_t is the white noise process.

3.4.4 Moving Average Model

The present value of the moving average is a linear mixture of present disturbance and previous disturbance. The MA indicator shows how many earlier periods have been included in the present value. or we can say that a moving average is a past error multiplied by a coefficient. MA is expressed as follows:

$$Y_t = \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} ... \theta_q \epsilon_{t-q}$$

It is referred to as a sequence of moving averages of order q. The expression of moving average arises from the fact that Y_t is gained by using the weights $1, -\theta_1$, - θ_2 ... -θ_q to the variables ϵ_t, ϵ_{t-1}, ϵ_{t-2}, ... , ϵ_{t-q} and then moving the weights and applying them to the ϵ_t, ϵ_{t-1}, ϵ_{t-2}, ... , ϵ_{t-q} to obtain y_{t+1} and so on. Slutsky (1927) was the first to investigate moving average models.

3.4.5 Autoregressive Integrated Moving Average (ARIMA) and Stationarity

The ARIMA model is the integration of three components, that are; AR(p), I(d) , and MA(q). The three parameters are found out accordingly. A series that is not stationary is distinguished in order to make it stationary. If not, the ARMA Model is applied. According to a general rule, standardized series with positive auto-correlation at lag 1 suggest AR terms to find out whether an AR or MA model is needed. At a certain level of differencing, MA terms perform the best in all other scenarios. The AR(P) model coefficients determine how quickly the series returns to its mean(Noureen et al., 2019). The series will return to its mean quickly if the total of the coefficients is close to zero. The series steadily returns to its mean if the sum is close to 1. A series exhibits moving-average behavior if some random shocks occur and are felt by two or more consecutive periods. If the MA(1) coefficient is negative, it indicates that some of the shock from the previous period is still being experienced. The ARIMA model changes to ARMA when differencing is not necessary.

When differencing is not needed, the ARIMA model becomes ARMA. Mathematially, ARMA model can be described as,

$$y_t = \beta_0 + \sum_{r=1}^{n} \varphi_r y_{t-r} + \sum_{i=1}^{q} \phi_i \epsilon_{t-i} + \epsilon_t$$

where β_0 denotes the intercept term, the parameters of AR and MA terms are $\varphi_r (r = 1, 2, 3, ..., p)$ and $\varphi_i (i = 1, 2, ..., q)$, respectively, and $\epsilon_i \sim N(0, {\sigma_\epsilon}^2)$.

3.4.6 Steps for fitting an ARIMA model

- The order of differencing is determined as the initial stage in fitting an ARIMA model.
- find out the numbers of AR and MA terms.
- The third and final step is to fit the model by making sure that the residuals are "white noise" and the highest order coefficients are significant. Also, to make sure that the forecast looks reasonable. If these things are not satisfied, return to the above steps, i.e., step 1 or 2.

3.4.7 Tools for Identification of ARIMA Model

ACF and PACF are the methods most frequently employed to identify ARIMA models. Below is an explanation of them.

Auto-correlation Function (ACF): It is the series' correlation with itself at different lags.

Partial Auto-correlation Function (PACF): It is the quantity of auto-correlation at lag k with the relationships of intervening observations removed. If the PACF unexpectedly breaks off after a few delays and the ACF slowly fades away, an AR(P) model should be chosen. The amount of spikes in the PACF graphic shows the AR(P) model's order. While suggesting the MA(q) model if the PACF dies out more gradually and the ACF abruptly stops off after a few lags. The amount of spikes in the ACF plot in this case indicates the MA term's order.

3.5 Sarima Model

In order to model a time series that takes into account both seasonal and non-seasonal elements, Box-Jenkins (1976) developed SARIMA model. The seasonal, autoregressive, intergraded, and moving average components are combined in the SARIMA.

Based on the ARIMA model, the single-integrated autoregressive moving average (SARIMA) model has the greatest potential for forecasting non-seasonal data sets for short-term study. However, certain data from time series sets show apparent seasonal swings caused by yearly fluctuations, monthly, quarterly, and other seasonal variations, as well as by some other inherent properties. Seasonal temporal data are transformed into ARIMA models using formal variance, seasonal variance, and autoregressive automatic averaging. (Kaur and Ahuja, 2019)

In short form the SARIMA model expressed as

SARIMA = (p, d, q) x (P, D, Q) s, where p,d,q are the non-seasonal components, P, D, Q are the seasonal components

A non-stationary seasonal time series is transformed to a stationary seasonal time series using finite-order seasonal and non-seasonal differences. The seasonal difference between the time series yt and the seasonal period s is referred to $\delta_s y_t$ follows:

$$\delta_s y_t = y_t - y_{t-s} \tag{3.2}$$

3.5.1 Develpoment of Sarima model

The following three steps in the development of SARIMA models are model identification, parameter estimation, and diagnostic testing. In order to estimate the parameters d, D, p, q, P, and Q, The analysis of sample autocorrelation function (ACF) and partial autocorrelation function (PACF) yields a preliminary SARIMA model. The model with the minimal AIC and Hannan-Quinn Criterion (HQ) score is the most effective. The periodic and non-seasonal AR and MA indicators are then estimated in the second stage. The final diagnostic verification stage determines whether or not the suggested model is adequate. If the model is found to be appropriate, it can be used to forecast future values; otherwise, the procedure is carried out until a suitable model is found. (Mohamed, 2021)

3.6 Artificial Neural Networks(ANNs)

Artificial neural networks(ANNs) were established as an alternate methodology for time series forecasting. Early work on Artificial Neural networks was done by Rosenblatt on the perceptron. Many people credit McClelland et al. (1986) and Rumelhart et al. (1986) for setting up the current revival in ANN technology. The highly linked structure of brain cells serves as the foundation for the ANN approach. This method is quicker than its conventional counterparts, robust in loud conditions, adaptable to a variety of challenges, and highly responsive to novel environments(Mohammadi et al., 2005).

While its popularity has risen up in recent years. To create a model that could simulate human brain intelligence in a computer was the fundamental goal of ANNs. ANNs, like human brains, strive to identify patterns and sequences in input data, learn patterns through experience, and at last create generalized output relying on the shared prior information. While ANNs were developed mostly for biological reasons, they were recently expanded for forecasting and categorizing applications in a variety of industries.

Many academics and scientists have found significant success using ANN, as one of the most popular artificial intelligence techniques, in a variety of domains, such as time-series

simulation and prediction in water resources. ANN has been shown to be an effective and trustworthy method for modeling nonlinear interactions between inputs and intended outputs in hydrologic time-series forecasting through numerous studies and experiments(Cheng et al., 2015).

The distinctive qualities of ANNs that have made them so well-liked for forecasting purposes are as follows: Data may already contain ANNs that are self-adaptive. The appropriate model is constructed based on the data presentations and descriptions in accordance with its characteristics and features, and there is no obligation to establish a model structure or any assumptions to be made about the distribution of the data. In many real-world situations where there are no technical guidelines for an appropriate data-producing technique, this approach is very useful. Second, ANNs, which are essentially nonlinear, are much more effective than conventional linear approaches like ARIMA for representing complex data structures. In several situations, ANNs were studied and found to predict significantly better than other linear models. Lastly, as postulated by Hornik along with Stinchcombe. ANNs have global approximation ability. It has proven that any function that is continuous may need to be estimated to the required precision. ANNs employ data processing that is parallel to calculate a wide variety of functions with significant precision. The problem may also be handled if the sources are incorrect or inadequate

3.6.1 The ANN Architecture

Multi-layer perceptions (MLPs) and one hidden layer feeding neural network (FNN) are the two ANNs that are most frequently used for problem prediction. A three-layer model is developed by using the input layer, hidden, and output layers. the input layer, which is where the network receives the data; the hidden layers in which data is processed secretly; and the output layer where outputs for specific inputs are generated. These Layers are connected by channels called acyclic connexons. One or more middle layers might be present. The nodes are also referred to as factories that produce components at different levels.(Benardos and Vosniakos, 2007) The forward architecture of ANN models can be described as follows:

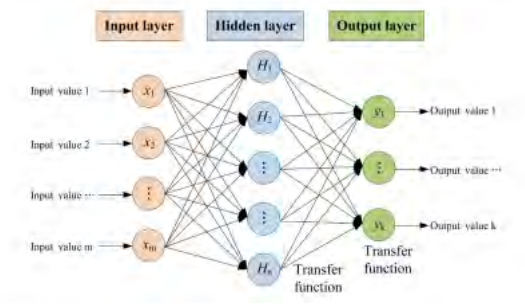

Figure 3.1: Structure of ANN

The following mathematical form is used to calculate the model's output:

$$z_t = f(\sum_{i=1}^{n} w_t * z_{t-i} + b) \tag{3.3}$$

Here $y_{t-i}(i = 1, 2, ..., n)$ and y_t are the n inputs and output respectively. The number n represents the number of inputs. wt are the connection weights that connect the two layers and b is the bias term. f is a function that is defined by the weights and network architecture.

The nonlinearity of the model can be captured using a variety of activation functions, which enables the model to learn more complicated non-linear correlations between the input and output variables. The nodes (neurons) in the network employ activation functions to calculate the node's output from its inputs. Most commonly the logistic sigmoid function is used.

$$f(y) = \frac{1}{1 + e^{-y}} \tag{3.4}$$

However Linear, hyperbolic, tangent, Gaussian, etc. may also be employed as the activation function(Kamruzzaman et al., 2006)

choosing the best activation function based on the complexity of the issue, network architecture, and type of data being used. It's critical to test many activation functions and select the one that delivers the greatest outcomes for the specific issue at hand. The relationship between weights can be estimated by using nonlinear least square algorithms based on error function minimization.

$F(\varphi) = \sum_t e_t^2 = \sum_t (y_t - \hat{y})^2$ The space of all connecting weights is shown here. The optimization techniques that can be utilized to minimize the error function are learning rules. The most prevalent learning principle in the literature is backpropagation or the generalized delta rule(Zhang, 2003).

3.6.2 Auto-Regressive Neural Network(ARNN)

A deep learning architecture called an autoregressive neural network (ARNN) uses historical data to forecast future data points. There are three layers in the neural network autoregression (NNAR) model: input, hidden, and output. It is a non-linear and parametric estimating model. This model is defined in the R forecast library, and after importing the forecast library, the NNAR model is made available.

ARNNs can be used for a variety of applications, such as voice recognition, language translation, and time series forecasting. A common technique for time series forecasting is ARNN, which aims to anticipate future values based on historical data. An ARNN is a kind of recursive neural network since it receives the prediction from the previous time step as its input at each subsequent time step. ARNNs are a useful time series forecasting technique because they can account for non-linearity in the data, especially for data with complicated patterns.ARNNs can be created using RNN or FNN, and they can be trained using supervised learning in which the network is given historical data and target values. The hidden layer has p-lagged inputs and k nodes, as shown by the notation NNAR(p,k). A neural network with an NNAR(11,6) design, for instance, uses the previous eleven observations as inputs to forecast the output, and the hidden layer contains six neurons. A NNAR(p,0) model is equivalent to an ARIMA(p,0,0) model, except it lacks the constraints on the parameters that guarantee stationarity.

Neural networks can tackle the time series forecasting problem based on these architectures. Many manual processes used in conventional modeling approaches can be removed, including stability verification, autocorrelation function testing, partial autocorrelation function checking, differentiation order selection, and so on.

3.7 LSTM-based Deep Learning Model

LSTM stands for Long short-term memory. Hochreiter and Schmidhuber invented the LSTM in 1997, and their default behavior is to recall long-term information To address gradient vanishing problem in over an extended time period series.

The use of parallel processing by GPU and cutting-edge optimization techniques has

sped and enhanced the implementation of deep learning-based models like LSTM compared to ANNs, which have a low number of processing units and layers due to the algorithm's computational limits (Li 2021).

It has been demonstrated that LSTM is capable of learning long-term dependency structures that are found in time series. (Evermann et al., 2017; Fischer & Krauss, 2018). The framework of the model is a type of recurrent network of neurons (RNN). In practice, gradients in LSTM, which has been presented as a typical RNN, disappear and explode. Traditional RNNs cannot learn long-lasting dependencies found in data sets. This approach is appropriate for numerous time-series water-related variables, such as river flow, underground table, and rainfall. It has been effectively used for financial market time series prediction, voice recognition, solar irradiance, electricity price prediction, rainfall-runoff modeling, and water flow modeling (Hu et al., 2018).

3.7.1 The LSTM Architecture

Figure 1 illustrates the architectural layout of lstm.

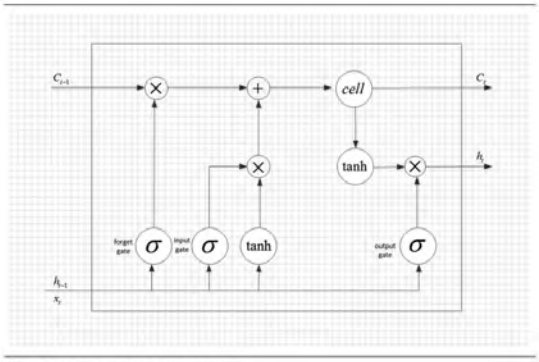

Figure 3.2: Structure of LSTM

A memory block, a forget gate, an output gate, an input gate, and one or more memory cells comprise it. The most significant component of the memory block is the memory cell, which can recall the state of the LSTM model from the prior time step.

In reality, the prior time step H_{t-1} output of the memory block and the present time series X_t function as the current input simultaneously when the current input time series X_t is a new data input into the memory block.

Forget gate determines the information in the memory cell first during the real operating procedure. Information that is unrelated to the forecast should be forgotten, while forecast-

related information should be held in reserve. The following equation represents the forget gate's activation function:

$$f_t = \sigma(S_{xf} * X_t + S_{hf} * H_{t-1} + b_f) \tag{3.5}$$

Where S_xf and S_hf are forget gate's weight parameters, b_f is its bias parameter, and $\sigma()$ is a sig mod function having a range of [0,1]. One signifies that the relevant data should be preserved in the memory cell. A value of zero, according to Baek and Kim (2018), indicates that the linked information in the memory cell must have been discarded.

The input gate chooses whether data from the current time series X_t and the memory block's output from the time step before that, H_{t-1}, should be entered into the memory cell and utilized to update the cell state in second step. First, using the [-1,1] range of the $tanh$ function, The candidate value Ct is the prospective candidate value that was used to update the cell's state at time t.

The following equation is then used to represent the input gate's activation function:

$$C_t = tanh(S_{xc}X_t + S_{hc}H_{t-1} + b_c) \tag{3.6}$$

$$i_t = \sigma(S_{xi} * X_t + S_{hi} * H_{t-1} + b_i) \tag{3.7}$$

where bc and bi are the bias parameter values for the memory cell and the input gate, respectively, and Sxc and Shc are the memory cell weight parameters. The weight parameters for the input gate are Sxi and Shi.

The next step is to modify the memory cell state in the present time t by utilising point-by-point multiplication using the candidate value Ct in the existing time t and the memory cell's prior state $Ct - 1$ as a basis.

The following equation gives the definition of the memory cell state update function.

$$C_t = f_t * C_{t-1} + i_t * C_t^{\sim} \tag{3.8}$$

When a memory cell's state is freshly finished, the output gate is used to send the result. Use the memory cell's output gate control information to output the desired outcome. The

output gate value *ot* is calculated by using the equation below.

$$o_t = \sigma(S_{xo} * X_t + S_{ho} * H_{t-1} + b_o) \tag{3.9}$$

S_{xo} and S_{ho} are the weight factors of the output gate, b_o is the bias parameters of the output gate. The state of the LSTM hidden layer is determined by the equation in the last step.

$$h_t = o_t * tanh(C_t) \tag{3.10}$$

If there are few layers and neurons, data attributes in the data set may not be effectively absorbed. If there are more layers and neurons, it might acquire more data characteristics within the data set. But it might also result in the issue of overfitting. So, by using the strategy of greedy search, we arrive at a superior result. The primary technique for LSTM model parameter adjusting is the greedy search method. The number of hidden neurons in the LSTM model and the number of hidden layers in the LSTM model serve as the primary adjustment variables during the greedy search process. A better LSTM structure can be discovered by looking for a specific range, which enhances prediction performance.

3.8 Seasonal Naive Model

A simple technique used in many fields, including statistics and machine learning, to provide a fundamental benchmark for comparison against more complex models is the usage of a naive model, also known as a baseline model. It is meant to establish a performance baseline and make it simpler to assess how well more intricate algorithms function.

The adjective "naive" denotes the fact that these models make comparatively straightforward assumptions and do not account for complex patterns or relationships in the data. Instead, they usually employ straightforward methods that require little computing effort. The basic premise of the simple naive model is that any future projection will match the most recent value seen.

To predict the time series data, two different Naive models are used. The first is called Naive-I, and it generates each forecast using the previous value of the time series. The second is called Naive-II, and it multiplies the forecast of the current observation by the rate

at which it has grown in relation to the previous observation.

SN is a better option than the naive model for highly seasonal data. A forecast for a single season is equal to the value of that season's previous forecast from the prior year.

Due to the repetitive nature of inflow, which makes our data purely periodic, the benchmark model should exhibit seasonality. As a result, we chose the seasonal naive model, that forecasts by monitoring values while simultaneously taking into account the previous season.

For predicting, this model has been chosen as the starting point.

For instance, the number for every month of July after that is anticipated to be the same as what was previously seen. This approach is effective for data with significant seasonality. The following are the Naive models for the supplied time series, let's say y_t, and the projected time series, $\hat{y_t}$ the time:

$$Naive - I : \hat{y_t} = y_t \tag{3.11}$$

$$Naive - II : \hat{y_t} = y_{t-1}[1 + \frac{yt - 1 - yt - 2}{yt - 2}] \tag{3.12}$$

$$SeasonalNaive : \hat{y_{t+h}}|t = y_{t+h-kp} \tag{3.13}$$

Where p is seasonal period and $k = [\frac{h-1}{p}] + 1$

3.9 The Modelling Framework

3.9.1 The ARIMA model Fit

Model specification In this research, The ARIMA model parameters were developed by studying the series' ACF and PACF. The ACF and PACF were exhaustively analyzed to determine the best-fitting model. The ARIMA(2, 1, 1) was to be found the best-fitted model. Thus The ARIMA (2, 1, 1) is given by the equation:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \theta_1 \epsilon_{t-1} + \epsilon_t \tag{3.14}$$

where: ϕ_1, ϕ_2 are the autoregressive cofficients and θ_1 is the moving average cofficient and ϵ_t is the residual term at time step t

Cofficients of ARIMA Parameters of the model that have been estimated from the data are represented by the coefficients. They aid in describing how the time series' previous values affect its present and future values. The calculated coefficients are as follows Autoregressive Coefficient for Lag 1 ϕ_1 = 0.9547 Autoregressive Coefficient for Lag 2 ϕ_2 = -0.1840 Moving Average Coefficient for Lag 1 θ_1 = -0.7565

Model Evaluation:

- $sigma^2$

The estimated variance of the residuals from the model (the difference between the actual and expected outcomes of the model) is given as σ^2, which is calculated to be 0.008766

- **log-likelihood**

The likelihood function's logarithm whose value is 1641.65, represents how well the model matches the data. A better match is indicated by higher values.

- **Akaike Information Criterion**

This statistic is used in the selection of models. It finds a balance between the complexity of the model and its goodness of fit. Lower AIC values indicate better models. The AIC in this situation is -3275.31.

Interpretation:

- The ARIMA model is denoted by the notation ARIMA(2, 1, 1), which stands for autoregressive, differencing, and moving average orders, respectively, of 2, 1, and 1.
- In the model equation, the lagged values and the lagged error term are combined to represent the differenced time series (y_t').
- The coefficients, standard errors (s.e.) show how accurately the parameter estimates are made.
- The error term ϵ_t's variance in the model is represented by the estimated variance σ^2.
- Among the indicators of how well the model resembles the data are the log-likelihood and AIC, with lower AIC values suggesting a better model fit.

3.9.2 The SARIMA Model Fit

Model specification

In this research, The SARIMA model parameters were developed by studying the series' ACF and PACF.To find best-fitting model, the Auto-correlation and partial auto-correlation functions were examined. The SARIMA(2,1,1)X(1,1,2)365.25 was found to be the best-fitted model. Here (2,1,1) is the non-seasonal order of the SARIMA and (1,1,2) is the seasonal order of the SARIMA Thus,

The SARIMA(2,1,1)X(1,1,2)365.25 is given by the equation:

$$(1-\varphi_1 l^{365.25}-\varphi_2(l^{365.25})^2)(1-l)(1-\vartheta_1 l^{365.25})(1-\varphi_3 l^{365.25})(1-l)(1-\vartheta_2 l^{365.25}-\vartheta_3(l^{365.25})^2) \tag{3.15}$$

Where: l is the lag operator (i.e.,$l^{365.25}y_t = y_{t-365.25}$) ϕ_1, ϕ_2 and ϕ_3 are the autoregressive and seasonal autoregressive coefficients ϑ_1, ϑ_2 and ϑ_3 are the moving average and seasonal moving average parameters, respectively ϵ_t is the residual term at time step t.

Cofficients of SARIMA

The coefficients are the model parameters that have been estimated from the data. They aid in describing how the time series' previous values affect its present and future values. The calculated coefficients are as follows:

Autoregressive Coefficient for Lag 1 ϕ_1 = -0.6803

Autoregressive Coefficient for Lag 2 ϕ_2 = 0.1035

Moving Average Coefficient for Lag 1 θ_1 = 0.5965

Seasonal Autoregressive Coefficient for Lag 1ϕ_3 =-0.4132

Seasonal Moving Average Coefficient for Lag 1θ_2 =-0.3025

Seasonal moving average coefficient for lag 2θ_3 =-0.6975

Model Evaluation:

- $sigma^2$

The estimated variance of the residuals from the model is given as σ^2, which is calculated to be 0.008772.

- log-likelihood,

The log-likelihood, which measures how well the model matches the data, is 1636.56. A better fit is indicated by a larger log probability.

- Akaike Information Criterion

AIC (Akaike Information Criterion) = -3259.13: The AIC is a measure that balances model fit and complexity. Lower AIC values indicate better models.

Interpretation: In order to determine the root causes of variation and patterns in time series data, the SARIMA model combines seasonal versions of the autoregressive (AR), moving average (MA), and differencing (I) components. The coefficients show degree and direction of each component's influence on the time series' present and future values.

3.9.3 The NNAR Model Fit

Three simple steps have been followed to build NNAR model:

- Defining parameters
- Training stage
- Testing stage

Defining parameters The auto-regressive neural network requires two parameters (p,k) as inputs to build the time series forecasting model. ARNN(p,k) model means that in hidden layer, there are p past values and k nodes in this model. In our model, $p = 2$, and $k = 2$ are selected which means that the model uses the most 2 recent values as inputs for out-of-sample forecasting and takes 2 nodes in the hidden layer.

Training Stage

In the training stage, we have used data from 4^{th} January 2018 to 31^{st} December 2021. The data set has to be defined and formatted suitably as a time series. We trained our model using the library forecast in R for one-step-ahead forecasting.

Testing stage At the testing stage, we predicted one step ahead from the period I^{st} January 2022 to 30^{th} September 2022 by setting the time horizon as h=1.

3.9.4 Seasoanl Naive fit

The Seasonal Naive (SNAIVE) forecasting method is a simple approach for making forecasts based on the premise that future values will be the same as prior seasonal cycle values. This approach is effective for time series data with strong seasonal patterns. This is how it works:

Let's break down the framework into three key steps: data preparation, forecasting, and confidence intervals.

- **Data Preparation:**

we have a time series dataset with a total of 1731 values. We've chosen the first 1457 values from 4^{th} January 2018 to 31^{st} December 2021 for training our model. This training period helps the model learn the seasonal pattern present in the data. We trained our model using the library forecast in R.

- **Forecasting:**

we're interested in predicting the next 273 values from I^{st} January 2022 to 30^{th} September 2022 in the time series.

- **SNAIVE Approach:**

SNAIVE Approach: The SNAIVE method assumes that future values will follow the same pattern as the corresponding values in the previous seasonal cycles.

- **Point Forecast:**

Point Forecast: For each of the forecasted points, the SNAIVE method generates a point forecast based on the historical pattern.

- **Confidence Intervals:**

They provide a measure of uncertainty around the point forecasts. Model provides two sets of confidence intervals: one with an 80% confidence level and another with a 95% confidence level. The wider the interval, the more uncertainty there is in the forecast.

- Seasonal NAIVE is a simple method that doesn't consider factors like trends or other potential influences on the data. It's best suited for time series with strong and regular seasonal patterns.

- The time series data is divided into individual seasonal cycles. For example, if the data is monthly, each year's worth of data would make up a seasonal cycle.
- It's a simple method that can be effective when dealing with data that exhibits regular and consistent seasonal behavior.

3.9.5 LSTM model fit

LSTM is a particular form of recurrent neural network (RNN) that is well-suited for tasks involving time series, natural languages, and more.

- **Model architecture:**

The model architecture is defined using the Keras library, the Python deep learning library which is commonly used for building and training neural networks. This specific architecture is a simple sequential model consisting of two main layers:

- **LSTM layer**
- **Dense layer**

The **LSTM layer** is the core component of this architecture. It's responsible for capturing sequential patterns and dependencies in the input data. The LSTM layer with 128 units processes input sequences and generates a hidden state that encodes information from previous time steps. One of the parameters that must be determined in the model is the number of hidden layer nodes. Experiments show that the model with 128 nodes performs well.

Dense Layer a dense layer that is entirely connected. It takes the LSTM layer's output as input and produces a single output value. The layer has 129 parameters, indicating that it has a single neuron connected to the LSTM layer and a bias term.

- **Model Compilation:**

The average squared error loss and the Adam optimizer are used to construct the model. The optimizer used determines how the loss function is minimized and consequently how the model proceeds to the ultimate result. Standard options include momentum, Adagrad, RMSProp, Adam, and so forth. The Adam optimizer is picked by experimentation.

- **Model Training:**

In model training, we use data from 4^{th} January 2018 to 31^{st} December 2021. The data set has to be defined and formatted suitably as a time series. The model is trained for lstm by using data for training with 60 epochs and a batch count of 32. During training, the model seeks to reduce the mean squared error loss. Batch size influences the amount of data handled at one time. Before processing the entire dataset, the model receives several updates via batches, which alters the process dynamics. The batch size is adjusted at 32 in this experiment since the small number of batches significantly reduces training speed while a big batch size leads to overfitting.

The periods during which the model traverses the full dataset are referred to as epochs. When the epochs are around 60, the loss of the test set is minimal. The epochs in the present study have been set at 60.

- **FORECASTING OF LSTM:**

For forecasting the input data is prepared by selecting the data points that come after the training period. An array is created to match the input shape that is expected by the model. When the forecast is generated, the predicted values are converted back to the original scale. We forecast the data points from 1^{st} January 2022 to September 30^{th} 2022

3.9.6 Performance Criteria of Forecasting Accuracy

The models' performance requirements will be evaluated using three types of error metrics. MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error), and RMSE (Root Mean Square Error) are three often used measures to quantify forecast accuracy in the context of forecasting and analyzing model performance.

- **Mean absolute error (MAE)**

The average absolute difference between forecasted and observed values is calculated using the MAE metric. It tells you how far your forecasts are from the actual values on average. The formula is as follows:

$$MAE = \frac{1}{m}\sum_{j=1}^{m}|z_j - \hat{z}_j| \tag{3.16}$$

- **Mean absolute percentage error**

MAPE is a metric that calculates the average percentage difference between forecasted and observed values. It's especially beneficial for understanding the relative magnitude of errors in percentage terms. The formula is as follows:

$$MAPE = 100 \cdot \frac{1}{m} \sum_{j=1}^{m} \frac{|z_j - \hat{z}_j|}{z_j} \tag{3.17}$$

- **Root Mean Square Error**

it quantifies the typical magnitude of the errors in a prediction or estimation by taking into account both the size and direction of the errors.

$$RMSE = \sqrt{\frac{1}{m} \sum_{j=1}^{m} (z_j - \hat{z}_j)^2} \tag{3.18}$$

Where:

$z_j$is the observed value at time step j

$\hat{z}_j$ is the estimated value at time step j

m is the number of time steps

Chapter 4

Results

4.1 Descriptive Statistics of Tarbela inflow

Table 4.1: Descriptive Statistics

Statistic	Value
Mean	77.29
Median	35.60
Variance	6092.715
Standard Deviation	78.05585
Minimum	8.60
Maximum	353.80
1st Quantile	20.50
3rd Quantile	115.70

Table 4.1 shows that the smallest value of Tarbela Inflow of our data set is 8.60 and the largest value is 353.80 which indicates a large degree of variation. 78.05585 is the standard deviation and the average value is 77.29 indicating that there is a prominent difference between average value and standard deviation.

4.2 Out-Of-Sample Tarbela inflow Forecasting

We have used five models i.e. ARIMA, SARIMA, ARNN, SNAIVE, and LSTM for one step ahead forecasting of the River Indus Tarbela inflow of Pakistan. It has been seen that the Tarbela inflow time series shows distinct properties. The time series exhibits periodicity and a significant long-term trend each month. Monthly data values are used for forecasting medium-term inflow, which often takes account of long-term trends along with annual and

seasonal periodicity. For instance, in Figure 4.1 it has been noticed that there is a growing trend in the Inflow of Indus Tarbela ranging from January 2018-September 2022.

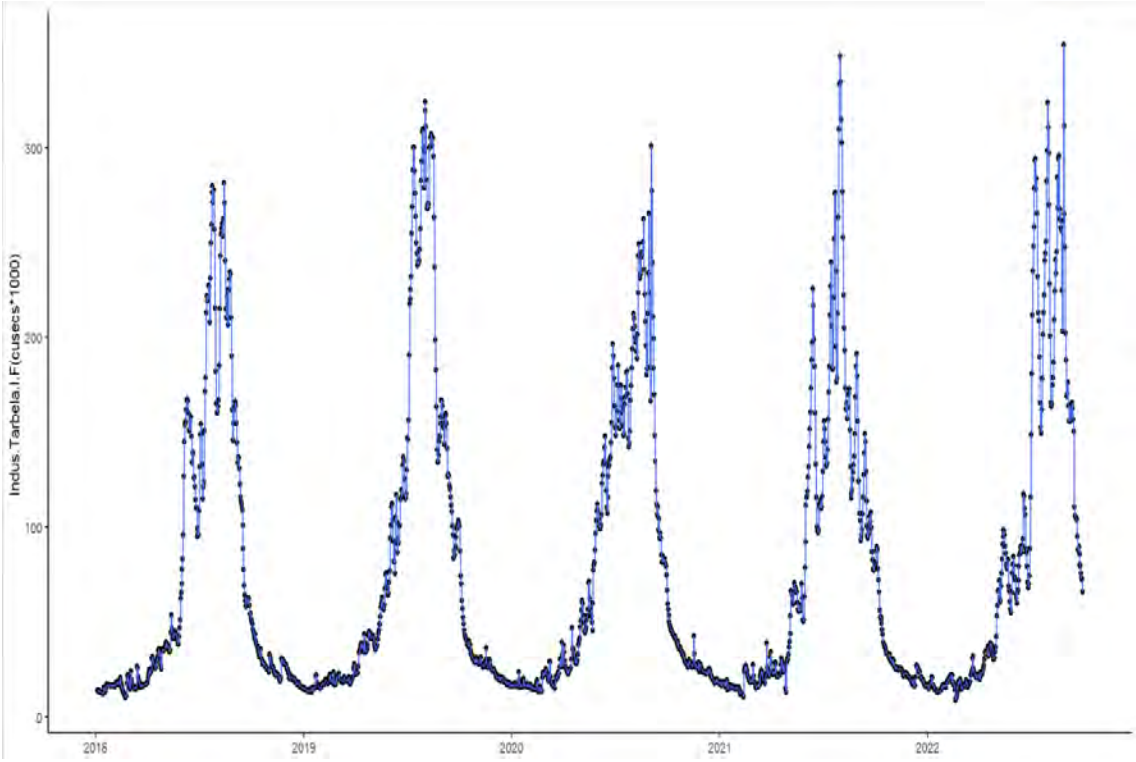

Figure 4.1: Indus Tarbela Inflow

4.3 Indus Tarbela Inflow Forecasting Results

The results of the forecasting accuracy are as follows:

Month		S-NAIVE	ARIMA	SARIMA	ARNN	LSTM
JAN	MAPE	5.270	5.650	5.621	6.215	2.010
	MAE	0.858	0.918	0.918	0.977	0.302
FEB	MAPE	10.972	11.570	11.48	10.513	1.6085
	MAE	1.6714	1.775	1.7622	1.5972	0.24344
MAR	MAPE	6.915	6.435	6.615	6.565	1.109
	MAE	1.438	1.323	1.352	1.373	0.202
APR	MAPE	5.802	5.744	5.697	5.896	1.134
	MAE	1.752	1.727	1.718	1.780	0.378
MAY	MAPE	7.731	6.944	6.897	6.527	2.969
	MAE	5.361	4.790	4.782	4.448	2.199
JUN	MAPE	7.063	6.377	6.595	6.413	2.894
	MAE	6.166	5.623	5.819	5.731	2.368
JUL	MAPE	7.720	6.964	6.588	4.926	3.153
	MAE	16.777	15.377	14.450	11.217	8.337
AUG	MAPE	9.534	8.882	9.262	7.723	3.337
	MAE	23.545	22.233	23.101	19.435	8.972
SEPT	MAPE	5.349	5.263	5.130	5.670	1.681
	MAE	6.613	6.500	6.355	7.137	1.603

The above table indicates that the monsoon season, typically occurring in July and August, is characterized by significant precipitation, often resulting in severe rain and dangerous flooding. This increased rainfall can lead to higher river flows, resulting in higher MSE and MAE values.

Figure 4.2 shows the forecasted values from January 2022 to September 2022 (273 data points) through ARIMA modeling with parameters p=2, d=1, q=1. Table 4.2 shows forecasting errors for the ARIMA model.

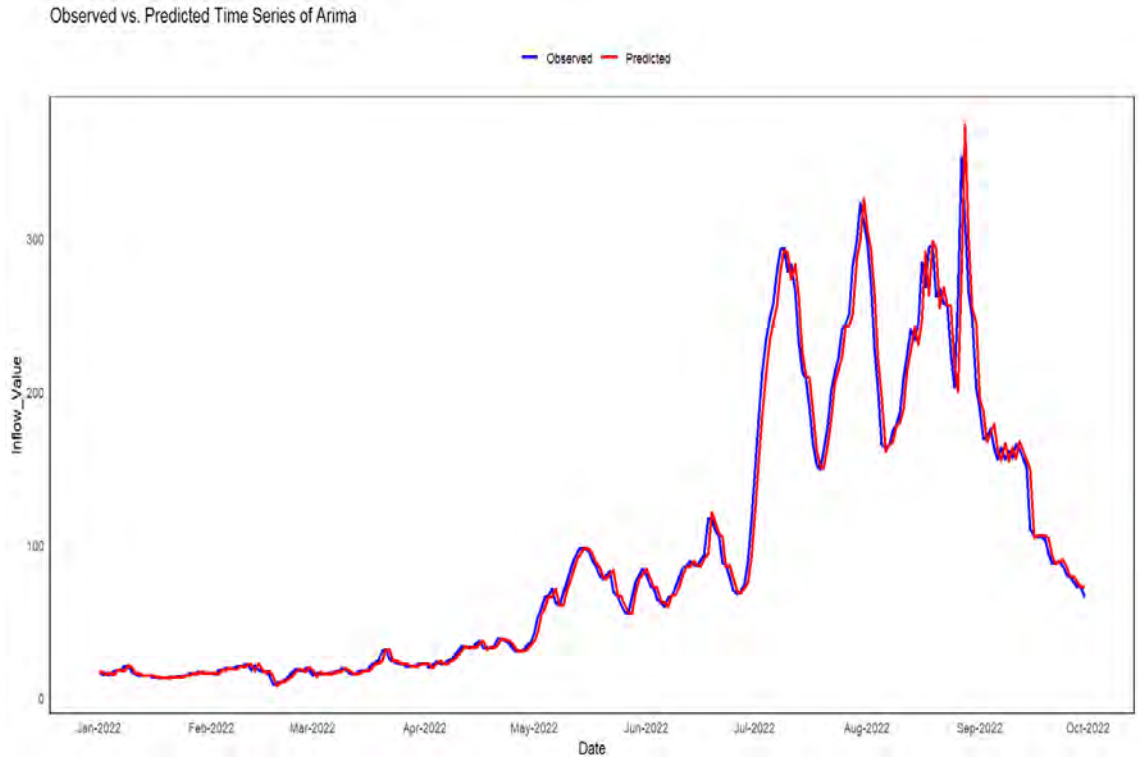

Figure 4.2: Actual and forecasted values of Tarbela inflow for ARIMA model from January 2022 to September 2022

Table 4.2: Accuracy measures of ARIMA

ARIMA	Value
MAE	6.773573
MAPE	7.057577
RMSE	12.51049

Figure 4.3 shows the forecasted values from January 2022 to September 2022 (273 data points) through SARIMA modeling with parameters p=2, d=1, q=1, P=1, D=1, Q=2, s=365.25. Table 4.3 shows forecasting errors for SARIMA model.

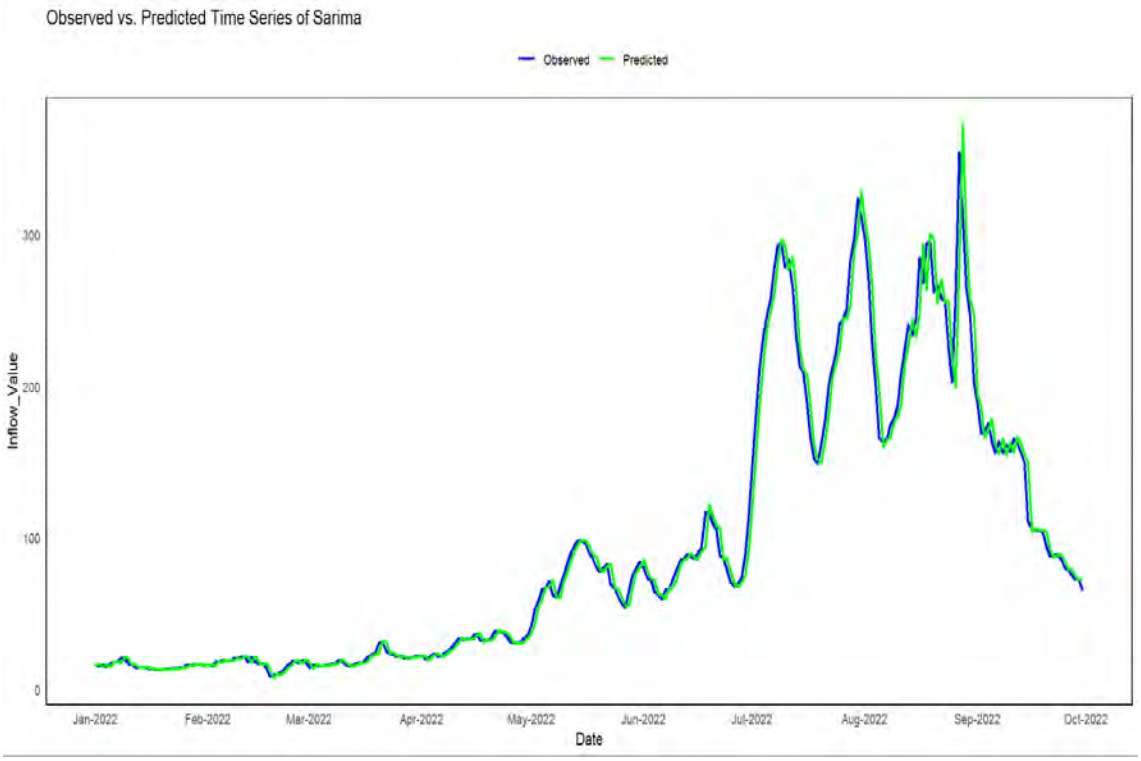

Figure 4.3: Actual and forecasted values of Tarbela indus for Sarima model from January 2022 to September 2022

Table 4.3: Accuracy measures of SARIMA

SARIMA	Value
MAE	6.772233
MAPE	7.063733
RMSE	12.50344

Figure 4.4 shows the forecasted values from January 2022 to September 2022 (273 data points) through ARNN modelling with parameters p=2 and k=2, where p is the number of input values and k is the number of nodes in hidden layer. The performance of the model was estimated using three standard accuracy measures i.e. MAPE, RMSE and MAE. The results are given in the Table 4.4.

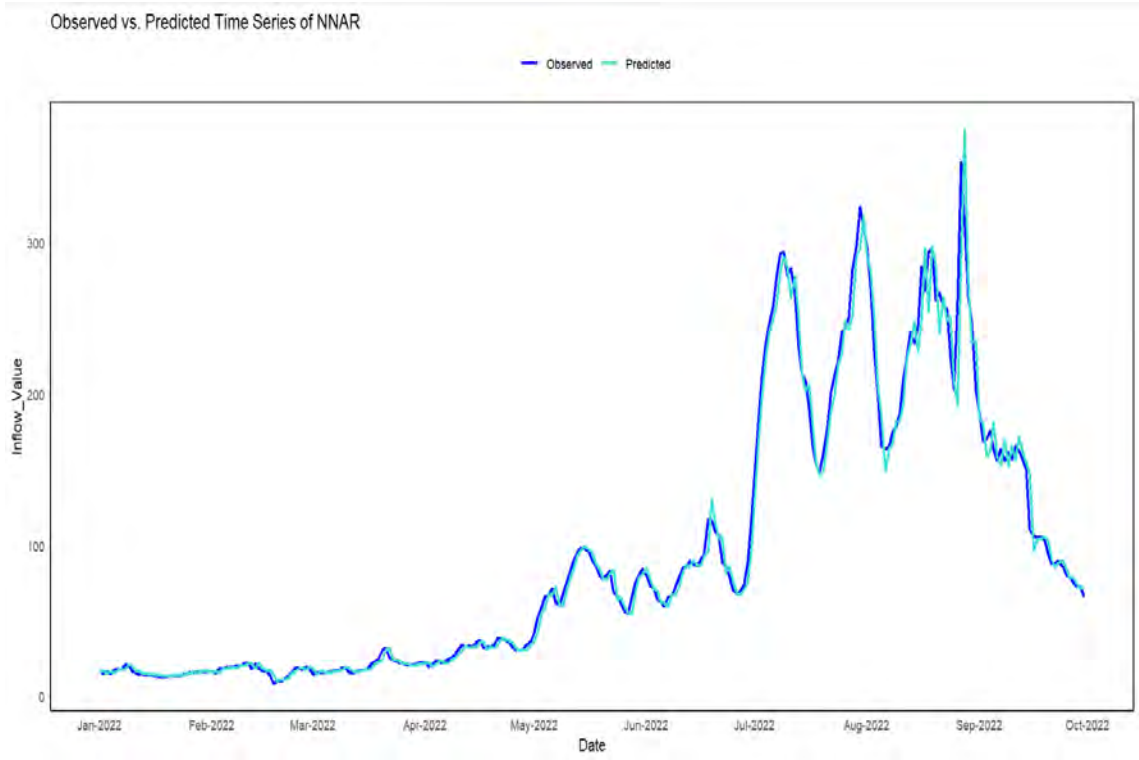

Figure 4.4: Actual and forecasted values of Tarbela Inflow for ARNN model from January 2022 to September 2022

Table 4.4: Accuracy measures of ARNN

ARNN	Value
MAE	6.045271
MAPE	6.695476
RMSE	13.23348

Figure 4.5 shows the forecasted values from January 2022 to September 2022 (273 data points) through Seasonal Naive modeling, Table 4.5 shows forecasting errors for the Seasonal NAIVE model.

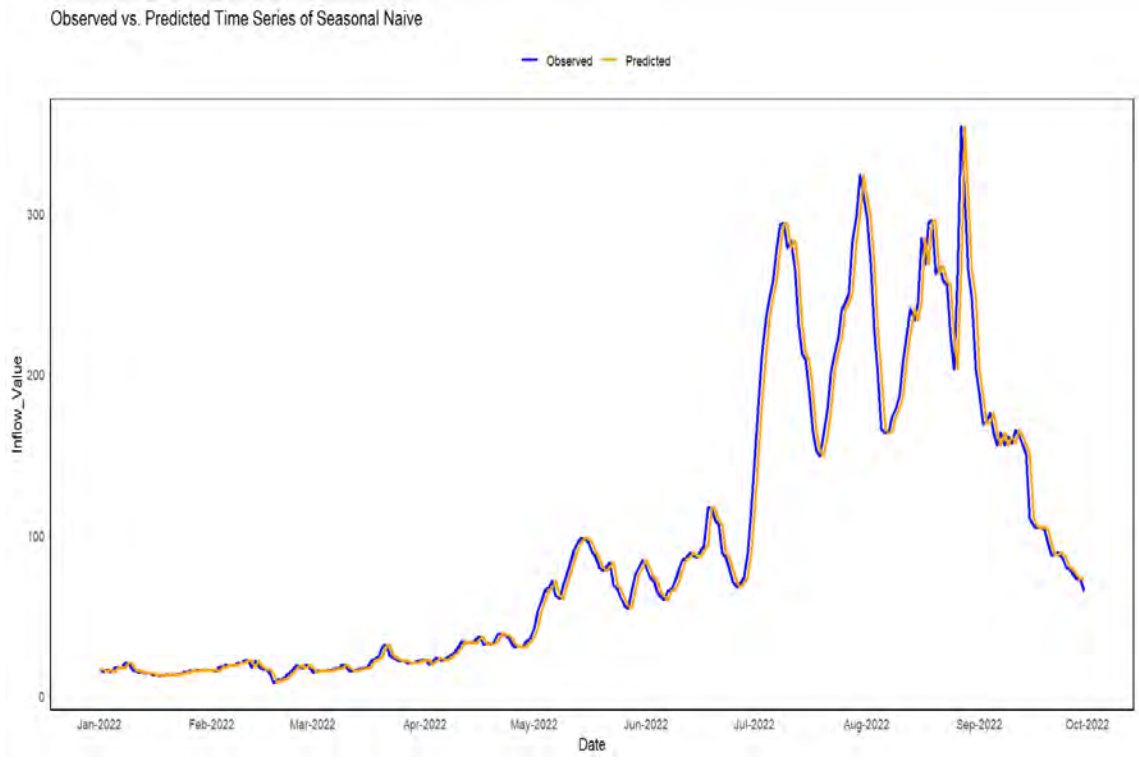

Figure 4.5: Actual and forecasted values of Tarbela inflow for seasonal naive model from January 2022 to September 2022

Table 4.5: Accuracy measures of S-NAIVE

S-NAIVE	Value
MAE	7.216484
MAPE	7.348168
RMSE	13.23348

Figure 4.6 shows the forecasted values from January 2022 to September 2022 through Long short-term memory modeling Table 4.6 shows forecasting errors for the LSTM model.

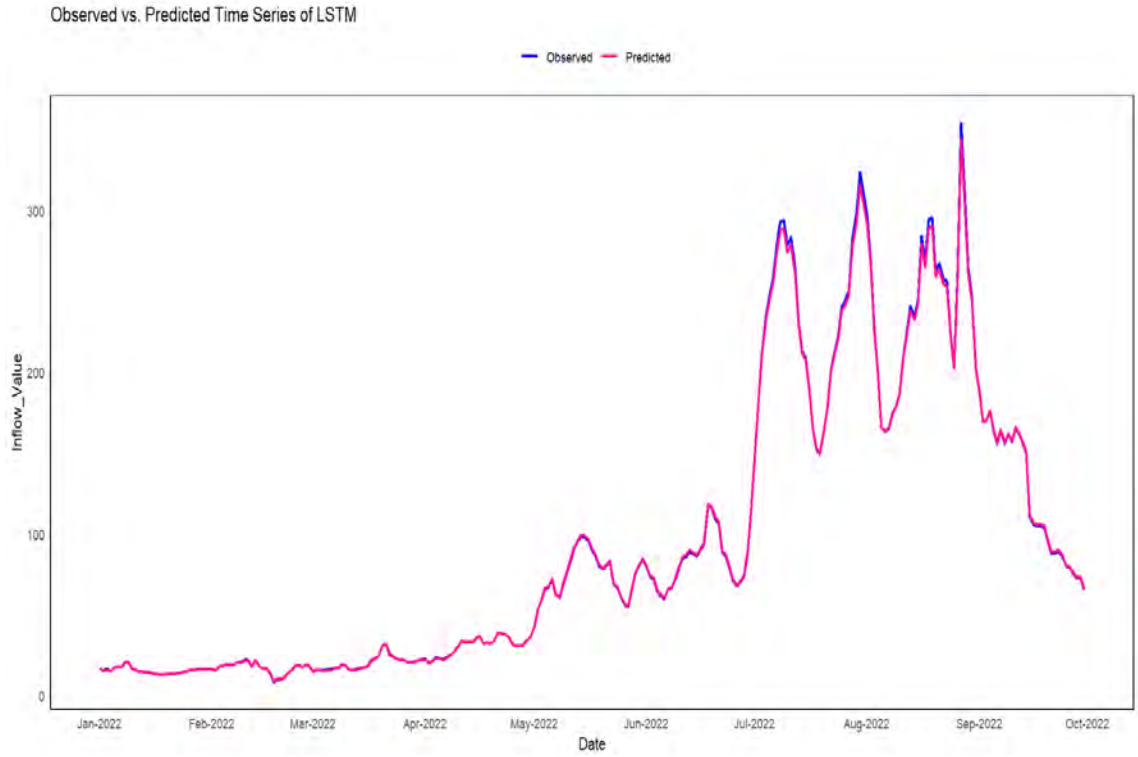

Figure 4.6: Actual and forecasted values of Tarbela inflow for LSTM model from January 2022 to September 2022

Table 4.6: Accuracy measures of LSTM

LSTM	Value
MAE	3.990651
MAPE	3.340282
RMSE	6.881253

4.4 Forecasted vs Observed Indus Tarbela Inflow

It can be observed that predicted values follow the actual values of Tarbela Inflow quite accurately. The results of the accura cy measure of all mentioned models are tabulated in Table 4.7. The table shows that LSTM performed well as compared to the other models with MAPE value 0.6239 , RMSE value 1.8098, and MAE value 0.9072.

Table 4.7: Accuracy measures of the models for one month ahead out of sample forecast

	MAE	MAPE	RMSE
ARIMA	6.773573	7.057577	12.51049
SARIMA	6.772233	7.063733	12.50344
ARNN	6.045271	6.695476	13.23348
S-NAIVE	7.216484	7.348168	13.23348
LSTM	3.990651	3.340282	6.881253

Chapter 5

Conclusion And Future Work

In the realm of the administration of water resources, accurate forecasting of Tarbela Indus inflow is critical for successful planning and decision-making.

The Indus River is full of natural resources, and it is one of the largest rivers in the world. It merges into the Arabian Sea and rises from Tibet mountains 90% of the food production of Pakistan relies on the Indus River. Indus River consists of two major basins, the Upper Indus Basin and Lower Indus Basin. Rain or melting glaciers and snow will increase the flow of upper Indus River due to which lower Indus basin will be overfilled and may lead to flood. To overcome this issue there is need to make reserviors, dams, barrages, etc. These can reduce the flow rate of rivers and to protect them from floods and other natural disasters. The flow of the river should be maintained to a certain level by proper maintenance. This study is performed on the daily flow data of the river Indus from 2018 to 2022 in which one factor is taken under study which is inflow rate. Flow of the river changes daily because of rain melting of snow and many other factors. If the river overflows then a flood will occur that causes great destruction. In order to solve this problem, we conducted this research and employed several approaches.

The purpose of this study was to advance the field of hydrological forecasting and introduce an efficient model to forecast onward Indus Tarbela Inflow by applying time series models and advanced machine learning models. In this regard, data from January 2018 to September 2022 of Indus Tarbela Pakistan is taken to assess the accuracy of models. The first four years of the data set are utilized for model estimate, while the remaining 273 data points from 2022 are used for out-of-sample forecast accuracy. To predict these data

points we evaluate the performance of five distinct forecasting techniques: Autoregressive Integrated Moving Average (ARIMA), Long Short-Term Memory (LSTM), Seasonal Naive, Autoregressive Neural Network (ARNN), and Seasonal Autoregressive Integrated Moving Average (SARIMA).

The rigorous investigation of historical data covering a significant time period, together with the use of the above-mentioned models, revealed significant insights. by comparing forecast accuracy measures, Among these strategies, we can find that the LSTM model demonstrated exceptional prediction performance with low values of MAE, MAPE, and RMSE indicating its ability to capture complicated temporal correlations within inflow data. The study demonstrated the ability of advanced machine learning and time series approaches to identify patterns in the complicated hydrological dynamics of the Tarbela Indus inflow.

The work presented in this thesis can be extended by incorporating various exogenous variables such as rain, snowmelt, and temperature into the model and studying their impact on out-of-sample forecasts

Also, different models can be used on the same data sets and compare their results. Different accuracy criteria can also be used to compare the results.

Advances in Tarbela Indus inflow forecasting include the use of hybrid models, the integration of data sources such as climatic data, satellite imaging, and reservoir levels, and the quantification of forecast uncertainty. Real-time data integration and climate change adaption are critical for effective decision-making. Machine learning models, such as LSTM, can improve interpretability and forecast long-term trends, allowing for more sustainable water management plans. Comparative studies with other forecasting approaches and hydrological models provide a thorough grasp of the advantages and disadvantages.

Bibliography

Abudu, S., Cui, C.-l., King, J. P., and Abudukadeer, K. (2010). Comparison of performance of statistical models in forecasting monthly streamflow of kizil river, china. *Water Science and Engineering*, 3(3):269–281.

Adnan, R. M., Yuan, X., Kisi, O., and Curtef, V. (2017). Application of time series models for streamflow forecasting. *Civil and Environmental Research*, 9(3):56–63.

Atashi, V., Gorji, H. T., Shahabi, S. M., Kardan, R., and Lim, Y. H. (2022). Water level forecasting using deep learning time-series analysis: A case study of red river of the north. *Water*, 14(12):1971.

Belvederesi, C., Zaghloul, M. S., Achari, G., Gupta, A., and Hassan, Q. K. (2022). Modelling river flow in cold and ungauged regions: A review of the purposes, methods, and challenges. *Environmental Reviews*, 30(1):159–173.

Benardos, P. and Vosniakos, G.-C. (2007). Optimizing feedforward artificial neural network architecture. *Engineering applications of artificial intelligence*, 20(3):365–382.

Cheng, C.-T., Feng, Z.-K., Niu, W.-J., and Liao, S.-L. (2015). Heuristic methods for reservoir monthly inflow forecasting: A case study of xinfengjiang reservoir in pearl river, china. *Water*, 7(8):4477–4495.

Cook, E. R., Palmer, J. G., Ahmed, M., Woodhouse, C. A., Fenwick, P., Zafar, M. U., Wahab, M., and Khan, N. (2013). Five centuries of upper indus river flow from tree rings. *Journal of hydrology*, 486:365–375.

Costa Silva, D. F., Galvão Filho, A. R., Carvalho, R. V., de Souza L. Ribeiro, F., and Coelho, C. J. (2021). Water flow forecasting based on river tributaries using long short-term memory ensemble model. *Energies*, 14(22):7707.

Fang, Z., Crimier, N., Scanu, L., Midelet, A., Alyafi, A., and Delinchant, B. (2021). Multi-zone indoor temperature prediction with lstm-based sequence to sequence model. *Energy and Buildings*, 245:111053.

Fashae, O. A., Olusola, A. O., Ndubuisi, I., and Udomboso, C. G. (2019). Comparing ann and arima model in predicting the discharge of river opeki from 2010 to 2020. *River research and applications*, 35(2):169–177.

Gelažanskas, L. and Gamage, K. A. (2015). Forecasting hot water consumption in residential houses. *Energies*, 8(11):12702–12717.

Hu, Y., Yan, L., Hang, T., and Feng, J. (2020). Stream-flow forecasting of small rivers based on lstm. *arXiv preprint arXiv:2001.05681*.

Ishfaque, M., Dai, Q., Haq, N. u., Jadoon, K., Shahzad, S. M., and Janjuhah, H. T. (2022). Use of recurrent neural network with long short-term memory for seepage prediction at tarbela dam, kp, pakistan. *Energies*, 15(9):3123.

Joshi, H. and Tyagi, D. (2021). Forecasting and modeling monthly rainfall in bengaluru, india: An application of time series models. *Int. J. Sci. Res. in Mathematical and Statistical Sciences Vol*, 8(1).

Kabbilawsh, P., Kumar, D. S., and Chithra, N. (2022). Performance evaluation of univariate time-series techniques for forecasting monthly rainfall data. *Journal of Water and Climate Change*, 13(12):4151–4176.

Kamruzzaman, J., Begg, R., and Sarker, R. (2006). *Artificial neural networks in finance and manufacturing*. IGI Global.

Katušić, D., Pripužić, K., Maradin, M., and Pripužić, M. (2022). A comparison of data-driven methods in prediction of weather patterns in central croatia. *Earth Science Informatics*, 15(2):1249–1265.

Kaur, H. and Ahuja, S. (2019). Sarima modelling for forecasting the electricity consumption of a health care building.

Khan, U., Janjuhah, H. T., Kontakiotis, G., Rehman, A., and Zarkogiannis, S. D. (2021). Natural processes and anthropogenic activity in the indus river sedimentary environment in pakistan: A critical review. *Journal of Marine Science and Engineering*, 9(10):1109.

Kim, B.-J., Lee, Y.-T., and Kim, B.-H. (2022). A study on the optimal deep learning model for dam inflow prediction. *Water*, 14(17):2766.

Kisi, O. and Kerem Cigizoglu, H. (2007). Comparison of different ann techniques in river flow prediction. *Civil Engineering and Environmental Systems*, 24(3):211–231.

Kreuzer, D., Munz, M., and Schlüter, S. (2020). Short-term temperature forecasts using a convolutional neural network—an application to different weather stations in germany. *Machine Learning with Applications*, 2:100007.

Lee, C.-M. and Ko, C.-N. (2011). Short-term load forecasting using lifting scheme and arima models. *Expert Systems with Applications*, 38(5):5902–5911.

Mehedi, M. A. A., Khosravi, M., Yazdan, M. M. S., and Shabanian, H. (2022). Exploring temporal dynamics of river discharge using univariate long short-term memory (lstm) recurrent neural network at east branch of delaware river. *Hydrology*, 9(11):202.

Mohamed, T. M. (2021). Forecasting of monthly flow for the white nile river (south sudan). *American Journal of Water Science and Engineering*, 7(3):103–112.

Mohammadi, K., Eslami, H., and DAYANI, D. S. (2005). Comparison of regression, arima and ann models for reservoir inflow forecasting using snowmelt equivalent (a case study of karaj).

Musarat, M. A., Alaloul, W. S., Rabbani, M. B. A., Ali, M., Altaf, M., Fediuk, R., Vatin, N., Klyuev, S., Bukhari, H., Sadiq, A., et al. (2021). Kabul river flow prediction using automated arima forecasting: A machine learning approach. *Sustainability*, 13(19):10720.

Nguyen, X. H. et al. (2020). Combining statistical machine learning models with arima for water level forecasting: The case of the red river. *Advances in Water Resources*, 142:103656.

Noureen, S., Atique, S., Roy, V., and Bayne, S. (2019). Analysis and application of seasonal arima model in energy demand forecasting: A case study of small scale agricultural load. In *2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 521–524. IEEE.

Pala, Z., Ünlük, İ. H., and Yaldız, E. (2019). Forecasting of electromagnetic radiation time series: An empirical comparative approach. *The Applied Computational Electromagnetics Society Journal (ACES)*, pages 1238–1241.

Parasyris, A., Alexandrakis, G., Kozyrakis, G. V., Spanoudaki, K., and Kampanis, N. A. (2022). Predicting meteorological variables on local level with sarima, lstm and hybrid techniques. *Atmosphere*, 13(6):878.

Pini, M., Scalvini, A., Liaqat, M. U., Ranzi, R., Serina, I., and Mehmood, T. (2020). Evaluation of machine learning techniques for inflow prediction in lake como, italy. *Procedia Computer Science*, 176:918–927.

Pradeepakumari, B. and Srinivasu, K. (2019). Dam inflow prediction by using artificial neural network reservoir computing. *Int. J. Eng. Adv. Technol*, 9:662–667.

Reza, M., Harun, S., and Askari, M. (2017). Streamflow forecasting in bukit merah watershed by using arima and ann. *Portal: Jurnal Teknik Sipil*, 9(1).

Selvi, P., Mahendran, K., et al. (2019). Forecasting the monthly inflow rate of the palar-porundalar dam in tamil nadu using sarima model. *Journal of Applied and Natural Science*, 11(2):375–378.

Shathir, A. K. and Saleh, L. A. M. (2016). Best arima models for forecasting inflow of hit station. *Basrah Journal for Engineering Sciences*, 16(1):62–71.

Shrestha, M. B. and Bhatta, G. R. (2018). Selecting appropriate methodological framework for time series data analysis. *The Journal of Finance and Data Science*, 4(2):71–89.

Sultana, N. and Sharma, N. (2018). Statistical models for predicting swine f1u incidences in india. In *2018 First international conference on secure cyber computing and communication (ICSCCC)*, pages 134–138. IEEE.

Tadesse, K. B. and Dinka, M. O. (2017). Application of sarima model to forecasting monthly flows in waterval river, south africa. *Journal of water and land development*, 35(1):229.

Valipour, M., Banihabib, M. E., and Behbahani, S. M. R. (2013). Comparison of the arma, arima, and the autoregressive artificial neural network models in forecasting the monthly inflow of dez dam reservoir. *Journal of hydrology*, 476:433–441.

Wang, Z.-Y., Qiu, J., and Li, F.-F. (2018). Hybrid models combining emd/eemd and arima for long-term streamflow forecasting. *Water*, 10(7):853.

Xu, W., Jiang, Y., Zhang, X., Li, Y., Zhang, R., and Fu, G. (2020). Using long short-term memory networks for river flow prediction. *Hydrology Research*, 51(6):1358–1376.

Yadav, S. and Sharma, K. P. (2018). Statistical analysis and forecasting models for stock market. In *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, pages 117–121. IEEE.

Zhang, G. P. (2003). Time series forecasting using a hybrid arima and neural network model. *Neurocomputing*, 50:159–175.