XIAMEN UNIVERSITY MALAYSIA

Course Code	:	AIT302
Course Name	:	Statistical Learning
Lecturer	:	Dr Shamini Raja Kumaran
Academic Session	:	2024/09
Assessment Title	:	Project
Submission Due Date	:	10/12/2024

Prepared by :

Student ID	Student Name
AIT2209952	TAN YI ZHAO
AIT2209949	SIM WEN KEN
AIT2209607	LEONG TING YI
AIT2209092	WANG YUXUAN

Date Received :

Feedback from Lecturer:

Mark:

XIAMEN UNIVERSITY MALAYSIA

Own Work Declaration

I/We hereby understand my/our work would be checked for plagiarism or other misconduct, and the softcopy would be saved for future comparison(s).

I/We hereby confirm that all the references or sources of citations have been correctly listed or presented and I/we clearly understand the serious consequence caused by any intentional or unintentional misconduct.

This work is not made on any work of other students (past or present), and it has not been submitted to any other courses or institutions before.

Signature: WENKEN TANGYA YUXUAN YAZHAO

Date: 8/12/2024

Contents

Introduction

Water scarcity is an alarming global problem in the era of an ever-growing population explosion and global warming. Aquifers are those hidden water works below the surface that we tap into not only for sustenance, but that have also become the theatre of another clash over the sustainable use of what many would see as our most precious piece of commons. Groundwater is not simply a resource; it is a crucial lifeblood of life sustaining humanity that feeds these aquifers. It gives potable water to billions, nourishes our crops, feeds industries and is the linchpin of our global civilizational arrangement. The world's population is expected to exceed 11 billion by 2100 putting further pressure on this finite resource (The Groundwater Project, n.d). So, knowing and being able to predict the dynamics of aquifers such as the one we have in Petrignano is crucial for water security now and in the future.

The Petrignano aquifer is a complex hydrological system, which is affected by several environmental factors such as rainfall patterns, temperature fluctuations, river levels, and extraction rates. Many of the details of these interactions prove difficult for traditional hydrological models to replicate. Here we intended to use statistical methods and machine learning to create a predictive model of the Petrignano aquifer and demonstrate the potential of AI to support informed decisions and sustainable water management practices. Predictive accuracy of aquifer water levels directly contributes to how we can sustainably extract from aquifers, mitigate severe droughts, regulate ecosystems connected to aquifers, and control pollution risk. This enables us to monitor that groundwater is not extracted too much, because this vital resource is essential for our next generations, and ourselves, to survive.

By combining the complexity of the Petrignano aquifer with a hybrid LSTM + ARIMA model, this project uses advanced AI functionalities. LSTM networks are very good at modeling long-term dependencies and non-linear relationships in time-series data, whereas ARIMA models are good at capturing short-term interactions and seasonality. Using a hybrid of the two above models enables us to take advantages of both resulting in better predictions of water levels. The AI-powered model will provide insights into the dynamic nature of the aquifer over the coming years, empowering relevant stakeholders to make informed decisions that balance environmentally sustainable practices and human need.

This work provides important insights for sustainable management of the Petrignano aquifer and is a general template for the application of AI and statistical learning in other aquifer systems around the world. In a water-stressed world, this is exactly the kind of innovative thinking that can help pave the way to more responsible management of water resources and ultimately, greater water security for everyone.

Work segregation

Our team consists of 4 members, the names and detailed tasks of each member are as follows:

Team Member Wang Yuxuan:

- Search for Aquifer Petrignano dataset from Kaggle.
- Clean and preprocess the data, handling any missing values
- Write the introduction section of the report, explaining the project scope, objectives, and significance.

Team Member Sim Wen Ken

- Conduct a literature review to identify potential enhancement methods.
- Provide analysis and suggestions in selecting in potential suitable enhancements methods for the dataset
- Write the literature review section, summarizing relevant studies and methods used in similar projects.
- Implement the ARIMA model using pre-processed data
- Tune the model parameters for optimal performance.
- Evaluate the performance of the ARIMA model using relevant metrics.
- Assists in documenting the implementation and results of the ARIMA model.
- Write the conclusion section of the report, provide a summary for the whole project

Team Member Leong Ting Yi:

- Collect evaluation results from both the ARIMA and LSTM + ARIMA models.

- Compare the performance of the two models using appropriate metrics.

- Create visualizations to illustrate the performance of the models.

- Summarize the findings and provide insights into the models' effectiveness in predicting water levels.

- Document the evaluation process and visualizations.

- Write the methodology section, detailing the approaches and techniques used in the project.

- Write the experimental analysis, using feedback from team members who actually implemented the models.

<u>Team member Tan Yi Zhao:</u>

- Quality Assurance of report as per guidelines, ensuring clear and concise presentation.

- Assist in experimental analysis, helping to refine and validate the models.

- Provide feedback and suggestions to enhance the content and presentation of the report.

- Research and understand the Long Short-Term Memory (LSTM) model and its combination with ARIMA.

- Implement the LSTM + ARIMA model using the pre-processed data.

- Tune the model parameters for optimal performance.

- Assists in documenting the implementation and results of the LSTM + ARIMA model.

- Evaluate the performance of the LSTM + ARIMA model using relevant metrics.

Literature Review

ARIMA Modeling: A Statistical Approach to Time Series Analysis

The Autoregressive Integrated Moving Average (ARIMA) is a popular statistical technique for examining and predicting time series data. It is especially proficient at capturing the temporal relationships and trends in a time series to forecast future values (IBM, 2024).

Components of ARIMA

The ARIMA model is characterized by three main components:

- **AR (Autoregressive) (p):** A model with this component illustrates a changing variable regressing on its lagged values. (IBM, 2024).
- **I (Integrated) (d):** This component differencing raw observations allows the time series to become stable by replacing data values with the difference between them and the preceding values (IBM, 2024).
- **MA (Moving Average) (q):** This component includes the dependency between an observation and a residual error from a moving average model on lagged observations (IBM, 2024).

Model Selection and Evaluation

The selection of appropriate values for p, d, and q is crucial for building an effective ARIMA model. This is often done through an iterative process involving:

- **Identification:** Examining the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots of the time series to determine potential values for p, d, and q (IBM, 2024).
- **Estimation:** Estimating the model parameters using statistical methods like maximum likelihood estimation (IBM, 2024).
- **Diagnostic Checking:** Evaluating the model's goodness-of-fit and checking for any violations of model assumptions (IBM, 2024).

What is an LSTM?

Fundamentally, an LSTM is a distinct variety of recurrent neural network (RNN). RNNs are created to handle data sequences where the arrangement of the components is essential. Nonetheless, conventional RNNs experience the vanishing gradient issue (Banoula, 2023). The vanishing gradient problem refers to a situation that takes place during the training of deep neural networks, in which the gradients utilized to adjust the network become very small or "vanish" as they are backpropagated from the output layers to the previous layers (GeeksforGeeks, 2023). LSTMs address this limitation with an advanced cell structure that includes:

- **Memory Cells:** These act as information reservoirs, retaining crucial details from past inputs to inform the processing of current data.

- **Gates:** Functioning as gatekeepers, these control the movement of information into and out of the memory cell. The input gate identifies which new information to retain, the forget gate decides which to eliminate, and the output gate regulates what information is transmitted to the following stage.

This intricate mechanism allows LSTMs to maintain a "memory" of past events, enabling them to discern complex patterns and relationships across extended time horizons.

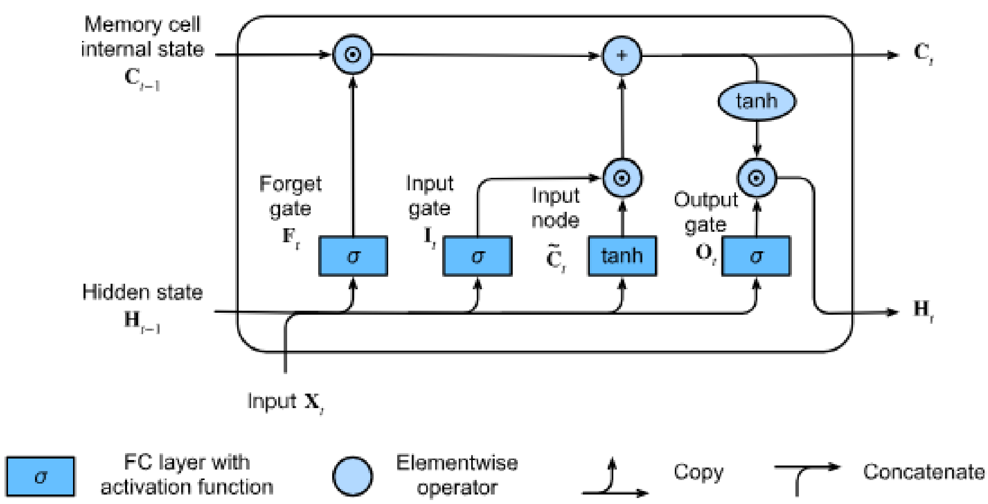

Figure 1: Structure of LSTM cell

LSTM in Time Series Analysis

LSTM is a type of artificial recurrent neural network utilized in deep learning, capable of handling complete data sequences. Owing to the model's capability to grasp long-term sequences of data, LSTM has emerged as a popular method for forecasting time series.

- **Capturing Temporal Dependencies:** LSTMs can identify intricate patterns and correlations between data points across time, such as seasonality, trends, and cyclical fluctuations (Rao,2024).

- **Handling Variable-Length Sequences:** Time series data often comes in varying lengths. LSTMs can readily accommodate these variations, making them adaptable to diverse datasets (Rao,2024).
- **Managing Missing Data:** Real-world time series data can be incomplete. LSTMs can effectively handle missing values while preserving their predictive capabilities (Rao,2024).

What is a Hybrid Model?

A hybrid model is a machine learning architecture that a woven several algorithms or techniques together. Pursuing to get better performance by overcoming the boundaries of standalone models. In such complicated and difficult tasks where a single model cannot explain all data, hybrid models can be particularly useful (Anifowose, 2020).

How Does a Hybrid Model Work?

Hybrid models work by integrating the strengths of different algorithms. The specific implementation varies depending on the task and the chosen models. Common approaches include:

- **Stacking:** Base models generate predictions, and a meta-model learns to combine these predictions for a final output (Anifowose, 2020).
- **Cascading:** Models are arranged sequentially, with each model's output feeding into the next (Anifowose, 2020).
- **Weighted Averaging:** Predictions from different models are combined using weighted averages, where the weights reflect model performance (Anifowose, 2020).

Are Hybrid Models Good?

Hybrid models often offer several benefits:

- **Improved Accuracy:** By combining diverse models, hybrid models can capture a wider range of data patterns, leading to better predictions (Anifowose, 2020).
- **Robustness:** Hybrid models are less prone to overfitting, as the ensemble nature helps to generalize better (Anifowose, 2020).
- **Versatility:** They can handle different data types and tasks, making them adaptable to various applications (Anifowose, 2020).

Why combine LSTM and ARIMA?

ARIMA (Autoregressive Integrated Moving Average)

- **Strengths:**
    - **Excellent for capturing linear patterns:** ARIMA models are able to specifically capture and model autocorrelations, meaning they are capable of being yet highly effective at data sets which display clear trends and seasonality (Mochurad, 2024).
    - **Simple and interpretable:** ARIMA models are simple to understand and interpret, meaning that we can derive insights about data generating process from ARIMA models (Mochurad, 2024).
    - **Well-established:** ARIMA has been used in time series analysis for multiple decades and has been demonstrated to be useful in numerous applications (Mochurad, 2024).

LSTM (Long Short-Term Memory)

- **Strengths:**
    - **Captures non-linear dependencies:** LSTMs, as a type of recurrent neural network, are capable of learning complex non-linear patterns in sequential data (Mochurad, 2024).
    - **Handles long-term dependencies:** LSTMs have this memory mechanism that enables retaining information from previous time steps, so they are appropriate for time series that exhibit long-range dependencies (Mochurad, 2024).
    - **Flexibility:** LSTMs can work with many kinds of time series data, even those with missing values or non-uniform time intervals (Mochurad, 2024).

Advantage of the combining LSTM and ARIMA

- **ARIMA captures linear patterns:** ARIMA can effectively model the linear and seasonal components of the time series (Mochurad, 2024).
- **LSTM captures non-linear patterns:** LSTM can capture the complex non-linear relationships that ARIMA may miss (Mochurad, 2024).
- **Improved accuracy:** The combination of both models can lead to more accurate predictions compared to using either model alone (Mochurad, 2024).

Discussion of Existing Works Utilizing Hybrid Models

- **Hybrid DL for Aquifer Water Quality Prediction:** Jamshidzadeh et al. to forecast the TDS total dissolved solids and EC electrical conductivity below the coastal aquifers based on a hybrid deep learning model (Jamshidzadeh et al., 2024). They proposed a hybrid model comprised of CNN, LSTM and GPRE component. The EC and TDS had respectively a NSE value of 0.96 and 0.95, with the CNNE extraction data functionality, the LSTM stored a one-time prediction while the GPRE auctioned interval predictions (Jamshidzadeh et al., 2024). In addition, the integrated model also achieved better performance than the declarative CNNE, LSTM and GPRE models used separately.
- **Fuzzy-LSTM for Seismic Control:** Zhang et al. By applying the LSTM to predict the displacement and velocity responses of structure and feed it into the fuzzy controller as input of the fuzzy control algorithm to obtain the active control forces, a new method of semi-active control strategy for structures under earthquake excitations is proposed (Zhang et al., 2023). Additionally, it utilizes the genetic algorithm to adjust the scaling factors of the displacement and velocity responses and the control force directly. Proposed Fuzzy-LSTM strategy compensates for delay control prediction ability (Zhang et al., 2023).
- **Stacking for Chronic Kidney Disease Prediction:** Khalid et al. Karakolcheva et al. (2023) applied a hybrid machine learning model (XGBoost) using a stacking classifier with a Random Forest algorithm as a meta-classifier to predict chronic kidney disease; It utilizes the UCI chronic kidney disease dataset (Khalid et al.,2023), where the features were selected using the Pearson correlation while several machine learning classification techniques were applied to evaluate their performance. The hybrid model suggested

(Khalid et al.,2023), consisting of Gaussian Naïve Bayes, Gradient Boosting and Decision Tree classifiers as base classifiers, reaches 100% accuracy against its base classifiers.

- **Optimized LSTM for Stock Price Prediction:** To predict stock prices, Gülmez (2023) proposed a deep Long Short-Term Memory (LSTM) network tuned by the Artificial Rabbits Optimization (ARO) algorithm. Optimizing the Hyperparameters of LSTM Network using ARO Algorithm for Predicting the Asset Price. The LSTM-ARO model is tested on DJIA index stocks as a dataset and compared with ANN, different LSTM types and a Genetic Algorithm optimized LSTM model (Gülmez, 2023). We evaluate the performance on four criteria: MSE, MAE, MAPE and R2. This further indicates that our proposed LSTM-ARO model outperforms all other methods in stock prices prediction for this experiment.
- **ARIMA-SGS for Groundwater Level Prediction:** Takafuji et al. (2018) compared the autoregressive integrated moving average (ARIMA), a time series model, as well as the sequential Gaussian simulation (SGS), a geostatistical model in seeing groundwater levels (Takafuji et al., 2018). Groundwater table depths were monitored at 49 wells within the site of the study: the Ecological Station of Santa Barbara (EECSB) in Brazil. The two methods were used to predict the groundwater table level and the errors were derived for measuring the performances of both methods. The output of space-distributed map directly from SGS method. However, ARIMA method requires interpolation method to be performed to get a map (Takafuji et al., 2018). In this sense, although the also perform slightly better than the SGS method and considers anomalous events, the ARIMA models are better suited for the aquifer monitoring system based on a combination of their superior precision and the fact that they can be automatically adjusted (Takafuji et al., 2018).
- **LSTM and ARIMA models for enhanced financial market predictions:** A Hybrid LSTM-ARIMA Approach for Financial Market Prediction. This method represents a hybridization of the two models by determining that LSTM is capable of capturing long-term dependencies and nonlinear relationships while ARIMA can handle simplified linear structures and seasonality (Mochruad, 2024). It is based on the RMSE criterion, their work is compared on three real datasets and the results indicate that their ensemble approach surpassed the LSTM alternative, transformer models and optimized DNN-LSTM (deep recurrent neural networks with LSTM) methods. Notably, the model attains a significant

reduction of 15% in RMSE and a minor increase in determination coefficient compared to LSTM only (Mochurad, 2024).

Methodology

This dataset focuses on aquifer characteristics derived from data collected by the Acea Group, Italy's leading water services operator. The primary goal is to develop predictive models for daily water level forecasts to optimize water resource management and ensure the sustainability of these critical water bodies.

This dataset focuses on aquifers, which are a key part of the ground level flow of water. The aquifers in particular are refreshed by river Chiascio, an important tributary to Italy's Tiber River.

Some factors that affect aquifer dynamics at present:

- **Rainfall:** Acts as a primary source of recharge, directly influencing water levels.
- **Depth to Groundwater:** Represents the vertical distance to the water table, serving as a key indicator of aquifer health.
- **Temperature:** Impacts evaporation rates, seasonal demand, and overall water availability.
- **Chiascio River Levels:** An external contributor that influences the inflow to aquifers.
- **Drainage Volume:** Represents how much water is taken out from the system, either naturally or as artificial intervention--which then has implications for overall equilibrium and sustainability of the aquifer.

Data Preprocessing

Standardizing and cleaning the data is an essential part of any data analysis or machine learning problem, including time series forecasting, but it is very important to ensure that the data set is clean, consistent, and well suited for the intended model. Preprocessing resolves typical problems like missing or inconsistent data, or even some features are less relevant, presence of noise resulting in the prediction's quality reduced dramatically. For time series dataset, this step is critical to ensure that all the data is placed into the right context: timestamps should align, interruptions in series should be avoided, the order of the data should be consistent, and season features or trends can be highlight.

We also made a heat map showing the interrelationship of each of the individual variables to identify importance of these variables. By looking at the heat map (Figure 2), we can note that “Depth_to_Groundwater_P24” and “Depth_to_Groundwater_P25” are pretty much highly correlated with one another (equal to 1), which means both of these features contain similar information about the groundwater levels, so those features are also essential for identifying aquifers (we have then simplify the two variable into one and named it as depth to groudwater). In contrast, other variables, such as “Rainfall_Bastia_Umbra”, “Temperature_Bastia_Umbra“, ”Temperature_Petrignano“, “Volume_C10_Petrignano”, and “Hydrometry_Fiume_Chiascio_Petrignano”, show weaker correlations with Depth to Groundwater, suggesting a less direct influence or predictive value for groundwater depth. By retaining only Depth to Groundwater and Date, the dataset is simplified, reducing noise and enhancing the focus on the primary variable of interest while ensuring the temporal patterns and trends inherent in time-series data are preserved. This choice improves model interpretability and computational efficiency, enabling clearer insights into groundwater dynamics over time while minimizing the impact of less relevant variables. By focusing on these two features, we emphasize the inherent time dependence within the time-series data, extracting meaningful insights or building predictive models based on past groundwater levels.

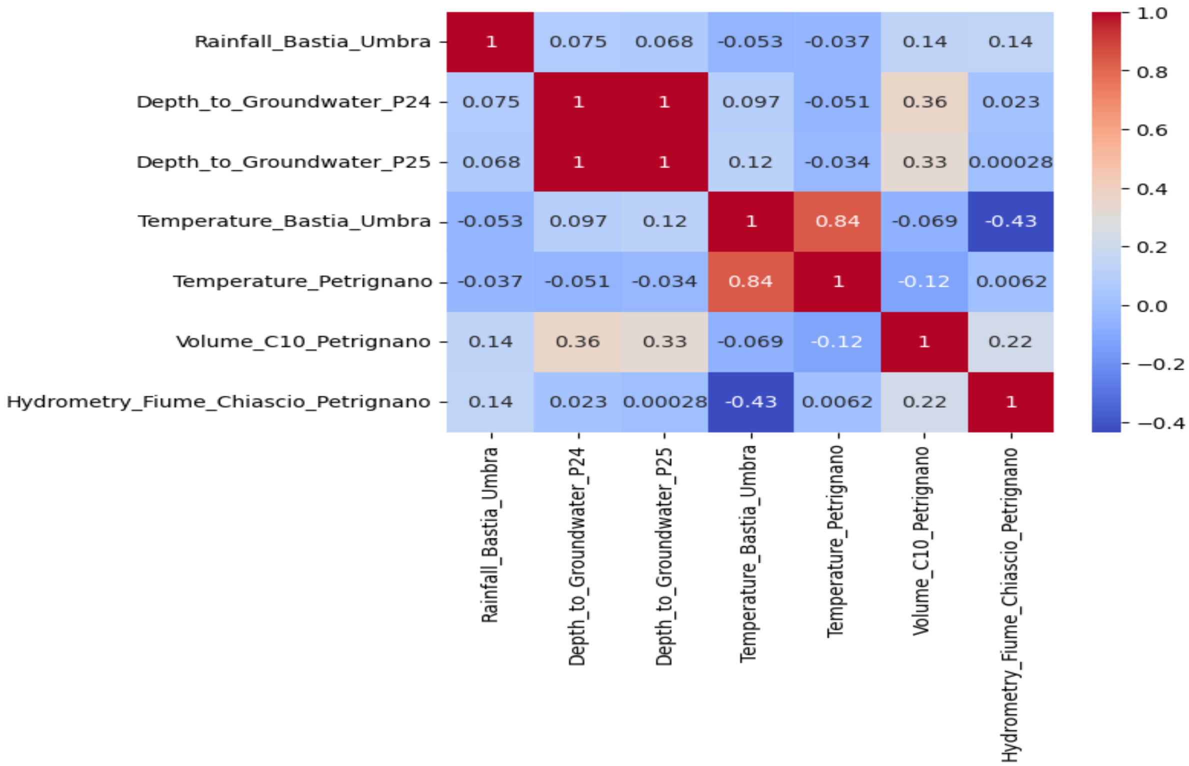

Figure 2: Heat map of the dataset features

Handle missing value

After dropping unused features, we obtained 39 missing values, and the next step is to process the missing values. Our goal is to gain insight into the most efficient ways to process and fill in these missing ('NaN') in time series data. a critical step in maintaining data integrity and ensuring accurate predictions. Unprocessed missing values can disrupt the temporal continuity necessary for reliable analyses. As an initial exploratory step, we temporarily fill in the 'NaN' values with placeholder values such as '0' or '25' to visualise the gaps. This is not just for aesthetic purposes, but to better understand the patterns and nature of the missing data, such as whether they are randomly distributed, clustered in a particular period, or consistent with other dataset characteristics.

The decision to use -25 instead of 0 is made to make the missing data points stand out more in the visualisation. While filling in the 0 could make the gap look larger, a very large gap would appear in the picture so that we could not visibly observe from the picture where the missing values were actually located. By using -25 we can create an obvious scaled down mark on the chart, making it easier to determine the exact location of the missing values without causing misinterpretation.

It is important to note that this is an interim, exploratory measure intended to aid visualisation. Our ultimate goal is to apply an estimation method that is robust and contextually appropriate. In the next section we will go through several methods to find the most suitable method.

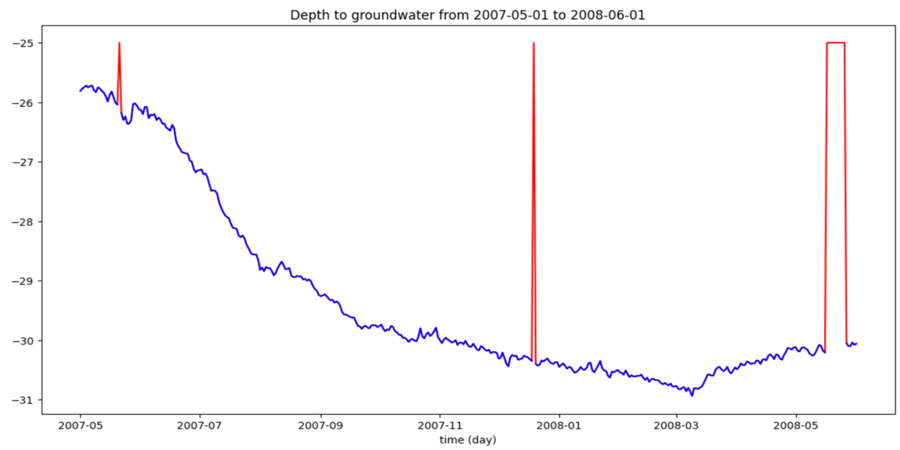

Figure 3: By filling the NaN value with -25, the graph clearly shows where the data are missing, allowing for better interpretation of trends and gaps in the dataset.

Comparison of different ways of handling missing values

What we are trying to do here is to perform some analytical drill-down to figure out the best way in which we can fill out missing values (NaN) from a time series dataset. Here we are investigating several fill strategies to determine which method retains the underlying trends, seasonal and cyclical behaviours inherent in the groundwater depth time series, while controlling for the effect of stretching the data by filling in any gaps.

The first step is to test what happens when we fill NaN with a fixed value for instance 0. Though the simplest form of this can have devastating effects of distorting the data set even when 0 is not a meaningfully accurate or realistic value, inserting false dips in the continuity of the time series. Then we try filling NaN with column mean value. This method assumes the mean is a fair-filling approach, so it does not go to extremes. But there is pitfall, it glosses over local variability and trends, flattening out dynamic segments of the data and potentially obliterating critical patterns.

We also cover forward filling, which fills in missing values with the previous value, as well as backward filling, which fills in notices with the next valid observation. Such methods are capable of preserving continuity and work particularly well on datasets where gradual changes happen. Nevertheless, they are not risk-free; a meaningful and seasonal change in the data may mean forwards and backwards filling introduces unwarranted old or future information into a given observation, not properly representing the data set.

Finally, we empirically test interpolation, which uses the trend between two neighbouring data points to estimate missing values. Arguably, this is the most advanced approach when it comes to the faithful replication of the productivity of the dataset. It provides a value that suits the local context, striking a balance between smoothness and accuracy. Yet, interpolation is very sensitive to smoothness assumptions, which might fail in datasets with abrupt corners or strange arrangements.

By visualizing these methods side by side within a specific date range, we can directly compare their impact. The red lines highlight the dataset after applying each imputation method, while the blue lines represent the original data, including gaps. This aggressive, methodical exploration is essential to ensure that the final imputation strategy not only fills gaps but also preserves the integrity and predictive potential of the time series.

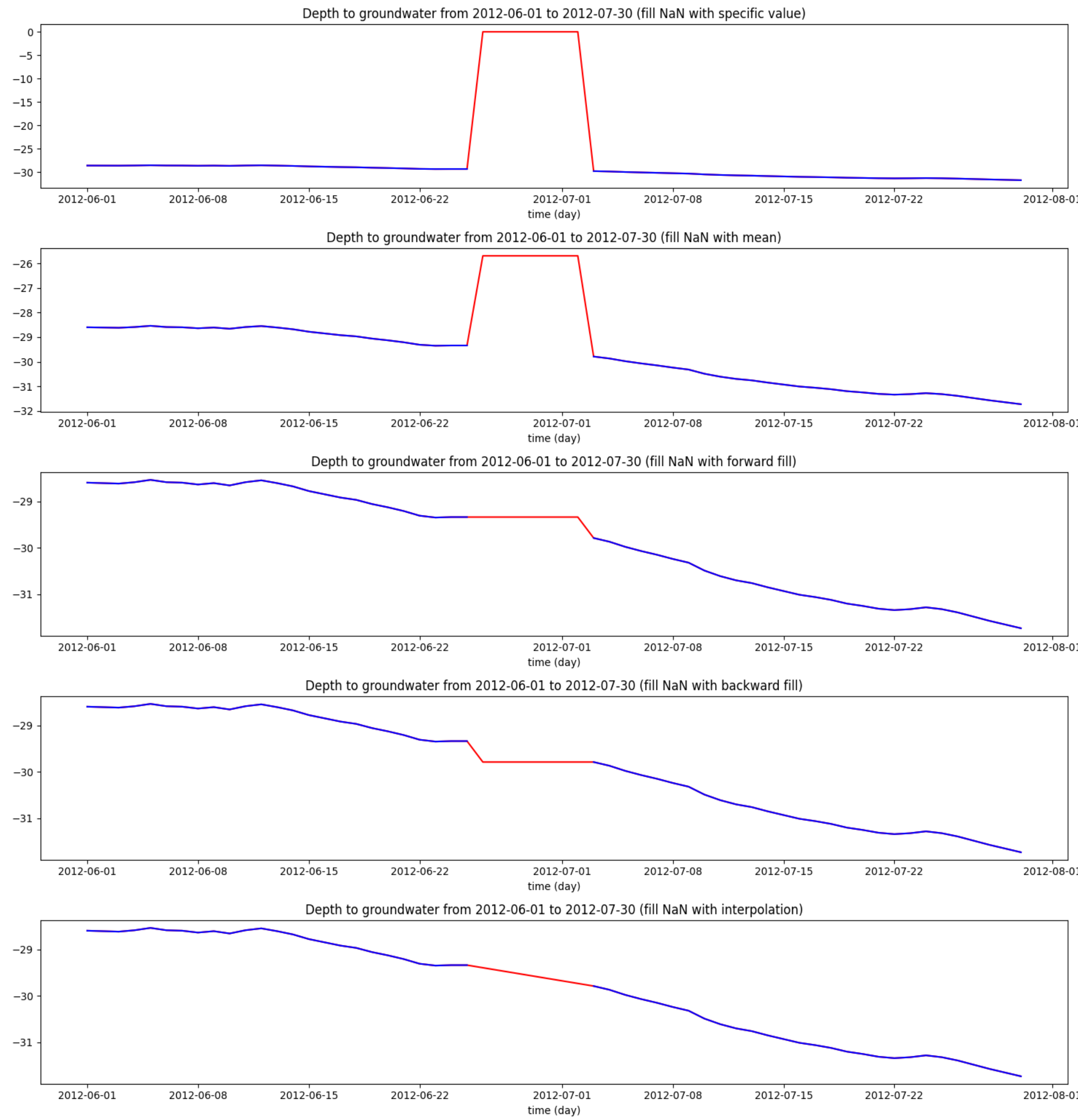

Figure 4: Comparison of handling missing values with different method

The above results (Figure 4) show the effects of various methods applied to missing values ("NaN") in a time series dataset, each with different characteristics. Filling with a fixed value ("0") produces unrealistic convex curvature that disrupts the continuity of the dataset, while mean filling introduces flat segments that oversimplify the data and eliminate local trends. Forward and backward filling maintain continuity by propagating the last or next known value, respectively, but can lead to unnatural plateaus that distort the data during periods of significant change. In contrast, interpolation provides smooth and seamless transitions across gaps, effectively preserving the natural flow and dynamics of the data, making it the most robust method. These results confirm theoretical expectations and highlight that interpolation is particularly well suited for datasets with gradual trends, both ensuring continuity and minimizing distortion. By comparing different methods for handling missing values, we found that interpolation seems to be the best choice because it can smoothly fill NaN values with existing values. After this series of operations, our data no longer has missing values.

Resampling

We performed down sampling on the data after addressing the missing values. Down sampling is one of the most common methods utilized in time series analysis that lowers the frequency of the time series in order to simplify it and make it smooth. Concretely, we transformed the data from daily level to weekly level, which shrunk the data points by a considerable amount and made it easier for long time series to analyse. Reducing the unpredictability of the underlying and only using larger time frames to fit and predict patterns improved our model's accuracy and at the same time removing noise from daily fluctuations to focus more on general trends. We down sampled to weekly data to bring attention to long-term trends as this helped to understand better the driving behaviour in the data itself.

Figure 5 below demonstrates that the lines in the previous image were not smooth and had a lot of jagged edges and spikes. Once we resample the data, we can clearly see that the lines in the chart have become more smooth and more natural.

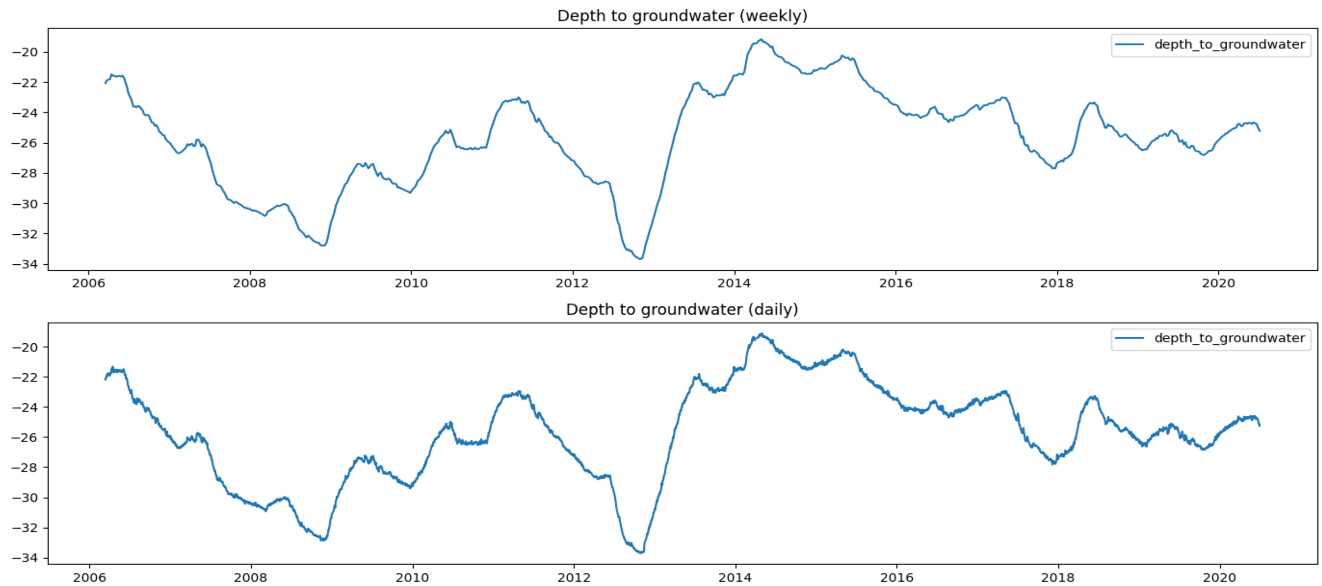

Figure 5: Resampling data from daily to weekly

ARIMA Model

After preprocessing the data, we can proceed to create our very first model. The first one is the ARIMA (AutoRegressive Integrated Moving Average) model to explore and predict the time series data. Out of the general statistical modelling method, ARIMA is a strong statistical modelling method developed directly for time series data which is a great method for our water table level prediction task. ARIMA is mostly used to find underlying patterns like trends, seasonality or autocorrelation in the data and use them for prediction.

We begin with the first test, which is based the Augmented Dickey-Fuller (ADF) test, because the ARIMA model assumes that the data needs to be stationary. A time series is said to be stationary if the mean, variance, and autocorrelation structure of the series do not change over time, thus rendering the series more predictable and simpler to model. In case ADF test confirms a time series is non-stationary, we can apply approximate transformations like differencing. This makes sure that it does not break the assumptions of ARIMA model.

Next, we proceed to the ACF and PACF plots, which help us to better understand the temporal structure of the data. ACF plot is used to examine the entire correlation of the series associated by its lagged values which identifies the patterns which are affected by moving average (MA). On the other hand, the PACF plot is aimed at displaying partial (direct) correlations between the series and a particular lag, revealing the autoregressive (AR) aspect. These tools enable us to decide p (AR order) and q (MA order) for ARIMA using data driven approach.

We split the dataset into training and test set to guarantee the model has not been overfitted to the data. We took 75% out for training and 25% for testing. ARIMA model is trained on the training data, but the test data is kept intact to evaluate the model predictive power on unseen data. This is an important step to know if the model generalises well so that it can be deemed as reliable for real world predictions.

In the final step, based on the determined parameters, the ARIMA model is fitted and simulated to identify the relationship between the past and future values of the groundwater depth. By keeping the processes structured and articulate we make sure that our ARIMA model not only works fine but also make practical sense and satisfies the result required for optimising the water resources.

However, the results show the ARIMA does not manage to catch the basic hidden patterns in this data. This can be due to the shortcomings of the ARIMA, which we are going to elaborate on in the experimental analysis part, and hence we have proposed to include LSTM with the ARIMA model.

LSTM + ARIMA

To improve forecasting, we developed a hybrid forecasting model, which combines Long Short-Term Memory LSTM + Autoregressive Integrated Moving Average ARIMA model, to obtain the complementary benefits of both approaches for robust and accurate time series forecasting. The hybrid overcomes each model's individual drawbacks by dividing the forecasting task into two parts, first capturing complex nonlinear relations and then resolving residual linear trends.

The hybrid model breaks the forecasting task into two stages, allowing each component to specialize in its core competence. Our model is an LSTM model, specifically designed to learn the complex, non-linear and temporal dependencies in the data such as those seasonal variations, outlier trends or chaotic behaviour. Conversely, the ARIMA model is capable of modelling linearities and autocorrelation in the residuals left over after the LSTM predictions, so that no underlying structure is ignored.

The purpose of implementing this hybrid LSTM + ARIMA model is to achieve superior time series forecasting performance by leveraging the complementary strengths of two distinct modeling approaches: machine learning and statistical methods. Real-world time series data often exhibit a mix of complex nonlinear relationships and simpler linear patterns or autocorrelations. Traditional models or machine learning techniques alone may struggle to capture the full complexity of such data.

Stage 1: LSTM Model for Nonlinear Dynamics

The first step involves transforming the time-series data so it can be used as inputs for LSTM by creating sequences of a fixed-length defined by look_back. This parameter defines the number of previous time steps, which is used to predict the next value and gives the model important context. In our design, the look_back is equal to 4, which equal to four weeks, or one month of data. As we believe the monthly variation is crucial in the model training. It then segments the time-series data into this supervised learning format, to make it possible to train the model.

```
# Build the LSTM model
model_lstm = Sequential()
model_lstm.add(LSTM(50, return_sequences=True, input_shape=(look_back, 1), kernel_regularizer=tf.keras.regularizers.l2(0.001)))
model_lstm.add(Dropout(0.2))  # Dropout layer
model_lstm.add(LSTM(50, kernel_regularizer=tf.keras.regularizers.l2(0.001)))  # Second LSTM layer with L2 regularization
model_lstm.add(Dropout(0.2))  # Dropout layer
model_lstm.add(Dense(1))  # Output layer
model_lstm.compile(loss='mean_squared_error', optimizer='adam')
```

Figure 6: LSTM model definition

The LSTM model is comprised of two layers of 50 LSTM neurons and is regularized with L2 to avoid overfitting. Each LSTM layer is followed by dropout layers which add on to the generalization by enabling random inactivation of a small percentage of neurons during training. The last layer is simply a dense layer with one output predicting the next value in the sequence. We train the model using the Adam optimizer, and the loss function is mean squared error (MSE). To save computation, early stopping is used to terminate training when the loss plateaus, thus also preventing overfitting.

Once trained, the model's standalone performance is evaluated by the generated predictions on the test set. Where the residuals — the differences between the actual training values and the LSTM's predictions on train set (train_residuals) is being collected. These residuals are vital in our approach as it reveals any patterns or structures in the data that the LSTM could not capture, often corresponding to linear relationships or remaining autocorrelations. Thus, in our hybrid model approach, these residuals will be addressed using ARIMA model, a key step of our hybrid model to get excellent performance.

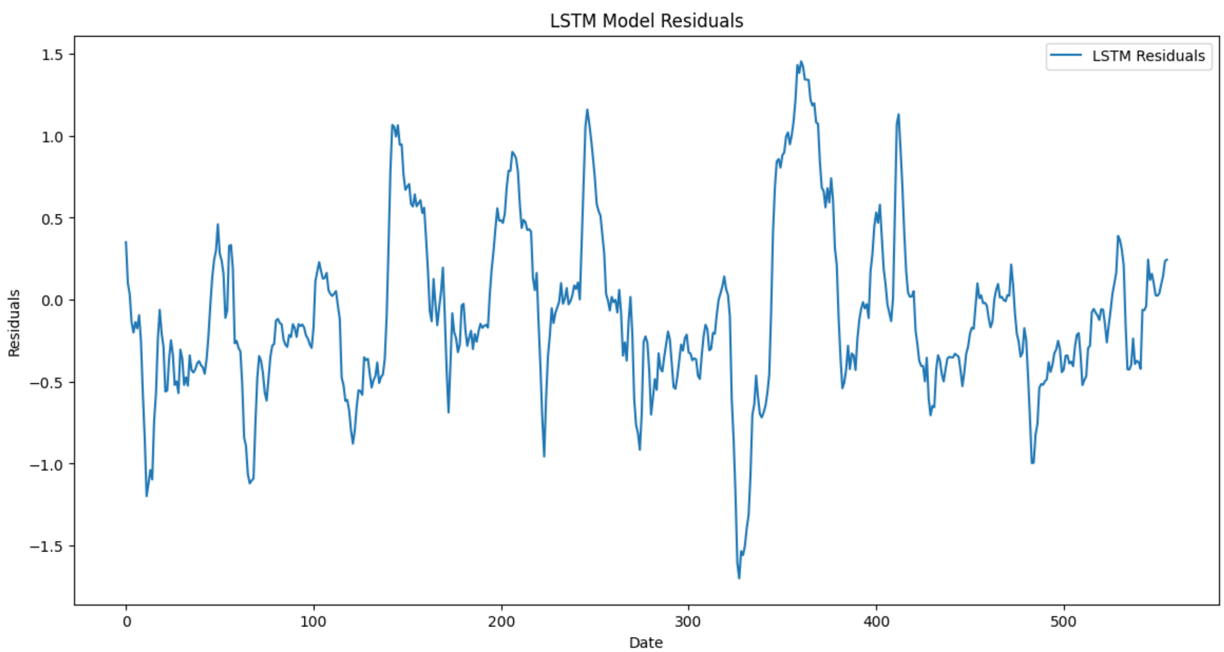

Figure 7: LSTM Model Residual

Stage 2: ARIMA Model for Residual Linear Patterns

To address these residuals, the ARIMA model is employed. ARIMA specializes in modeling stationary linear time series with strong autocorrelations. Before applying ARIMA, the stationarity of the residuals is tested using the Augmented Dickey-Fuller (ADF) test. If the residuals are stationary, ARIMA can directly model them; otherwise, differencing or transformations might be required. Significant autoregressive (p) and moving average (q) orders are determined using Partial Autocorrelation Function (PACF) and Autocorrelation Function (ACF) plots, respectively.

Stage 3: Hybrid Model and Forecast Combination

To guarantee alignment between the LSTM and ARIMA forecasts, we addressed the potential discrepancy in prediction lengths by extending the LSTM predictions to match the ARIMA output. This step is crucial because the (look_back) value of 4, representing a month's worth of weekly data, inherently limits the LSTM model's initial forecast horizon to the size of the test dataset (in other words the nature of LSTM looks back 4 weeks of data to make a prediction/forecast limits the length of prediction it makes). To overcome this limitation, additional LSTM predictions are generated iteratively, using the final sequence of test data as the starting point. For each additional step required, the LSTM model predicted the next value, which is subsequently scaled back to its original range. In the dataset, these predictions are incorporated one after the other to match the temporal consistency requirement, allowing the model to learn changing patterns over the long forecast horizon. After aligning the length of the LSTM predictions with ARIMA forecast. We can now then integrate them into hybrid structure that would combine the non-linear characteristics captured by the LSTM along with the linear residual patterns modelled by ARIMA, providing a complete definition of the forecast in a satisfactory manner.

By combining the strengths of ARIMA and LSTM, the hybrid model gives predictions from the two models together. More specifically, the ARIMA model's predictions for remaining errors (residual_forecast_arima) are combined with the LSTM prediction (test_prediction_lstm). Hence it captures both LSTM's ability to model complex nonlinear relationships and ARIMA's ability to reveal linear and residual patterns in the time series.

As a result, this LSTM + ARIMA hybrid model is shown to be an advanced and suitable approach to time series forecasting, where each model complements each other. The LSTM is good at capturing non-linear dependency and temporal patterns, and the ARIMA can account for remaining linearity and autocorrelation in residuals. Together, they offer a comprehensive solution that can deliver remarkably precise predictions that solve depth to ground water over time (detailed analysis will be discuss in the experimental analysis part).

Experimental Analysis

ARIMA Model Interpretation

Stationarity Analysis

```
ADF Statistic: -3.2589584364925495
p-value: 0.016815718504471742
Critical Values: {'1%': -3.4391937559530965, '5%': -2.8654430713273373, '10%': -
2.568848417404698}
```

Figure 8: ADF Test Results (ARIMA)

First, time series models require stationary data for forecasting. Therefore, an augmented Dickey-Fuller (ADF) test is applied in the code. The ADF test has a null hypothesis (H_0), which states that the data is non-stationary, and an alternative hypothesis (H_1), which states that the data is stationary. For the ADF test, two conditions must be satisfied to reject H_0: the ADF statistic should be lower than the critical values, and the p-value should be less than 0.05.

The results showed that the ADF Statistic (-3.26) is less than the critical value at the 5% level (-2.87) but not at the 1% level (-3.44). Furthermore, the ADF Statistic is also lower than the critical value at the 10% level (-2.57), satisfying this condition at the 10% and 5% levels. The p-value (0.01682) is less than 0.05, fulfilling the second condition. Since the ADF Statistic is more negative than the critical value at the 5% level and the p-value is below 0.05, we reject the null hypothesis at the 5% significance level. Additionally, the data satisfies the stationarity condition at the 10% level, further strengthening the conclusion.

Overall, this analysis confirms that the data is stationary at both the 5% and 10% significance levels, as the p-value is less than 0.05 and the ADF Statistic is more negative than the corresponding critical values.

Autoregressive Integrated Moving Average (ARIMA) model analysis

After the data is stationary, we start to build the ARIMA model which means we find the p and q value by ACF and PACF values.

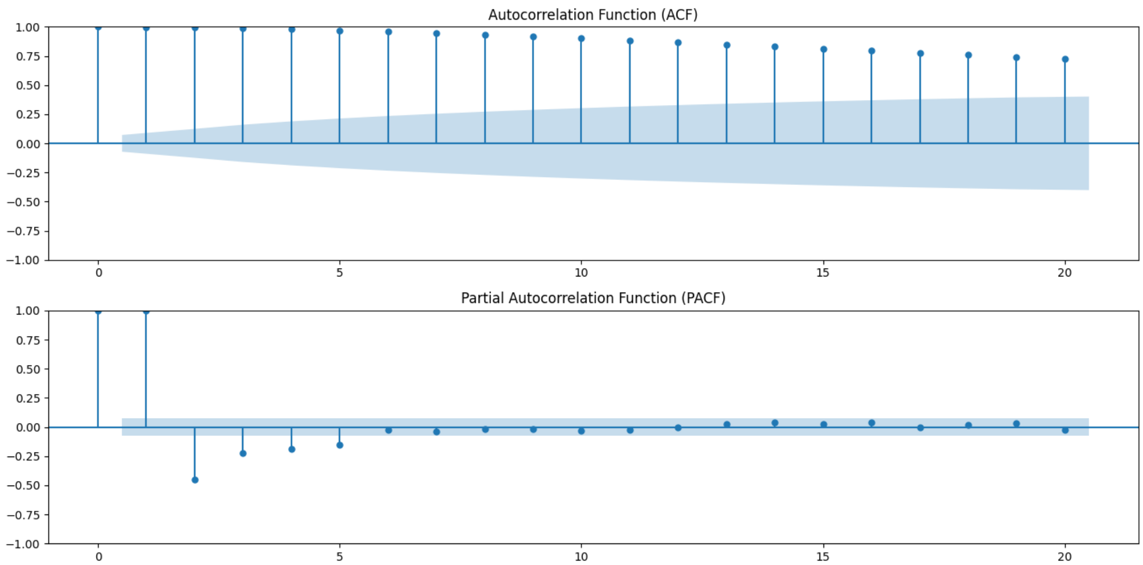

Figure 9: ACF and PACF plot

In order to get the suitable p and q value, we calculate the 95% confidence interval threshold to find the p and q values. Then the result shows that the significant p and q value is 20. Therefore, this number can be directly fitted to the ARIMA model. Although higher p and q values help capture more complex patterns in the data, it also increases the complexity of the model, which may lead to overfitting.

SARIMAX Results

Dep. Variable:	depth_to_groundwater	No. Observations:	560
Model:	ARIMA(20, 0, 20)	Log Likelihood	381.701
Date:	Wed, 04 Dec 2024	AIC	-679.402
Time:	19:57:42	BIC	-497.629
Sample:	03-19-2006	HQIC	-608.424
	- 12-04-2016		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
const	-25.7762	1.187	-21.723	0.000	-28.102	-23.451
ar.L1	0.1437	2.902	0.050	0.961	-5.544	5.832
ar.L2	-0.2934	2.839	-0.103	0.918	-5.859	5.272
ar.L3	0.1010	2.336	0.043	0.966	-4.478	4.680
ar.L4	-0.1373	2.094	-0.066	0.948	-4.242	3.967
ar.L5	0.2655	1.673	0.159	0.874	-3.013	3.544
ar.L6	0.0269	1.957	0.014	0.989	-3.808	3.862
ar.L7	0.3109	1.655	0.188	0.851	-2.934	3.556
ar.L8	0.1215	1.828	0.066	0.947	-3.461	3.704
ar.L9	0.3562	1.631	0.218	0.827	-2.841	3.554
ar.L10	0.1896	0.874	0.217	0.828	-1.524	1.903
ar.L11	0.2580	1.100	0.235	0.815	-1.897	2.413
ar.L12	-0.0363	1.152	-0.032	0.975	-2.294	2.221
ar.L13	0.0764	1.324	0.058	0.954	-2.519	2.672
ar.L14	-0.1902	1.169	-0.163	0.871	-2.481	2.100
ar.L15	0.0373	1.084	0.034	0.973	-2.087	2.161
ar.L16	-0.2603	0.908	-0.287	0.774	-2.040	1.520
ar.L17	0.1704	0.889	0.192	0.848	-1.571	1.912
ar.L18	-0.1755	0.895	-0.196	0.845	-1.929	1.578
ar.L19	0.0393	0.902	0.044	0.965	-1.728	1.807
ar.L20	-0.1317	0.841	-0.157	0.875	-1.779	1.516
ma.L1	1.3260	2.902	0.457	0.648	-4.362	7.014
ma.L2	1.8708	3.912	0.478	0.633	-5.798	9.539
ma.L3	2.2215	3.802	0.584	0.559	-5.230	9.674
ma.L4	2.6946	4.969	0.542	0.588	-7.045	12.434
ma.L5	2.8360	6.233	0.455	0.649	-9.380	15.052
ma.L6	3.0021	6.501	0.462	0.644	-9.739	15.743
ma.L7	2.8652	6.956	0.412	0.680	-10.769	16.499
ma.L8	2.7184	6.931	0.392	0.695	-10.866	16.303
ma.L9	2.3302	6.106	0.382	0.703	-9.638	14.298
ma.L10	1.9598	5.697	0.344	0.731	-9.207	13.126
ma.L11	1.4129	5.059	0.279	0.780	-8.504	11.329
ma.L12	1.1093	3.984	0.278	0.781	-6.700	8.919
ma.L13	0.7199	3.062	0.235	0.814	-5.281	6.721
ma.L14	0.5680	2.298	0.247	0.805	-3.936	5.072
ma.L15	0.3311	1.439	0.230	0.818	-2.489	3.151
ma.L16	0.4143	1.182	0.351	0.726	-1.902	2.730
ma.L17	0.1342	1.011	0.133	0.894	-1.847	2.116
ma.L18	0.1680	0.692	0.243	0.808	-1.188	1.524
ma.L19	-0.0029	0.596	-0.005	0.996	-1.171	1.166
ma.L20	0.0592	0.164	0.360	0.719	-0.263	0.382
sigma2	0.0146	0.001	18.534	0.000	0.013	0.016

Ljung-Box (L1) (Q):	0.01	Jarque-Bera (JB):	38.90
Prob(Q):	0.94	Prob(JB):	0.00
Heteroskedasticity (H):	0.78	Skew:	0.18
Prob(H) (two-sided):	0.09	Kurtosis:	4.24

Figure 10: ARIMA model fit summary

This ARIMA (20, 0, 20) model despite seems to be promising, but the result shows that it exhibits signs of overfitting that may compromise its efficacy and interpretability.

20 orders of AR, 20 orders of MA and a constant are included in the model to account for complex dependencies in the time series data. Though the inclusion of many terms supports detailed representation of possible patterns, analysis on the model's coefficients leads to a key problem, a great number of these terms are statistically insignificant.

In particular, though the constant term itself shows a p-value under the typical 0.05 level, demonstrating its relevance in the model, the vast majority of AR and MA terms, which is all orders, do not satisfy this condition. The most appropriate fit of AR and MA components might well be interpretable as these parameters provide no useful information about in making predictions.

Including truly irrelevant terms can also lead to the model overfitting, meaning that the model tries to learn too much from the data, including noise or random fluctuations, making it less generalizable. As a result, a model with overfit may show very high accuracy on the data it learned from but fail to perform well on new data it never saw before. Which is the situation we encountered.

```
Mean Absolute Error (ARIMA): 0.8597619186454893
Mean Squared Error (ARIMA): 1.2381632497930664
Root Mean Squared Error (ARIMA): 1.1127278417443622
```

Figure 11: Evaluation Metrics for ARIMA

The root mean square error (RMSE) value (1.112727 for the ARIMA model) indicates the magnitude of the model's forecast error, with lower RMSE values indicating better performance. These results clearly show that the root mean square error is high, which proves that the model's forecast results are not ideal. This data also shows the limitations of the ARIMA model in modeling nonlinear relationships and long-term dependencies, and that it is difficult to fully capture the complexity of time series data.

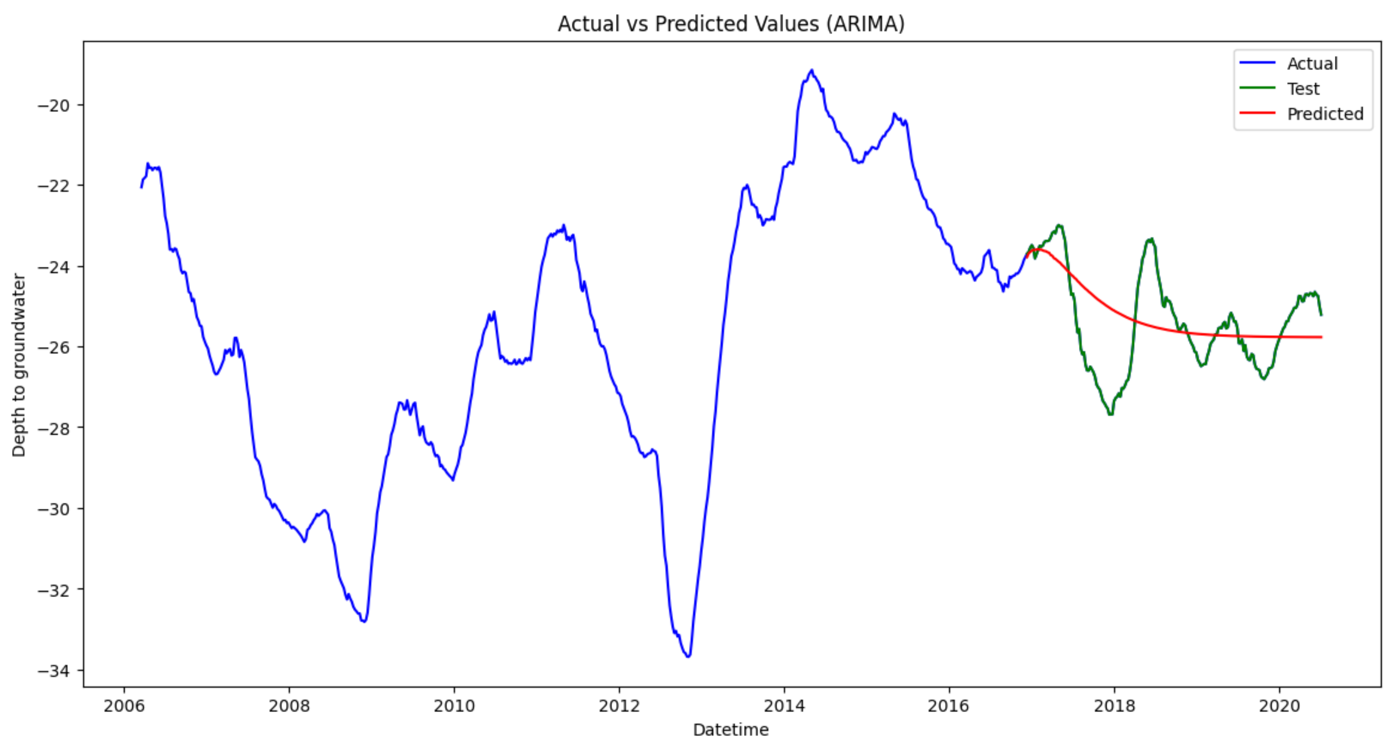

Figure 12: Actual vs Predicted Values of ARIMA model

Figure 12 reveals a key limitation of the ARIMA model, its inability to effectively capture the complex dynamics of the groundwater depth data. The predicted values (red line) exhibit a noticeable smoothing effect, failing to align with the sharp fluctuations and non-linear patterns observed in the actual values (green line) during the testing period. The spread is more defined by the Root Mean Squared Error (RMSE) which is calculated at 1.112727. Such a high RMSE suggests a good gap between predicted values with actual which does not give a lot of confidence in the ARIMA model.

These limitations are a result of the linear and stationary assumption that ARIMA makes. In its simplest form, ARIMA models assume that the data generating process can be presented as autoregressive (past values), integrated (differencing for stationarity), and moving average (past errors). However, the underlying processes governing groundwater depth are often characterized by non-linear relationships and complex feedback mechanisms that defy simple linear representations. As a result, ARIMA cannot learn these complex interactions well, and this is reflected in the large differences between the predictions and a big RMSE. This highlights the fact that a more complex model able to learn the non-linear nature of the dataset is required. Which brings us to the hybrid model.

Hybrid Model Interpretation (LSTM+ARIMA)

The ARIMA model is useful for understanding the time series data, however its inability to model complex non-linear patterns is also limitation. We highlighted the need of enhancement for ARIMA model which leads us to investigate the appealing combination of Long Short-Term Memory (LSTM) networks with the ARIMA model to increase its precision. The key idea behind this hybrid approach is to take advantage of the strengths of both the models, LSTM learns long term dependencies while on other hand ARIMA model is great at capturing linear relationships.

Analysis of LSTM

```
6/6 [==============================] - 1s 1ms/step
Mean Absolute Error (LSTM Only): 0.3170533144948634
Mean Squared Error (LSTM Only): 0.179884369809859
Root Mean Squared Error (LSTM Only): 0.4241277753341073
```

Figure 13: Evaluation Metrics for LSTM model alone

While ARIMA models excel at capturing short-term dependencies and linear patterns in time series data, they often struggle with non-linear relationships and long-term dependencies. This limitation is evident in the significantly higher RMSE of 1.112727 observed for the ARIMA model compared to the LSTM model alone, which achieved an RMSE of only 0.4241427, which is a big improvement from the previous results. Despite the appealing results, however, these results are considered as nearly satisfactory only, we believe at this point we haven’t reached the limit.

While LSTM has a lot of strengths, it has its shortcomings as well. Although its strengths include learning long-term dependencies and nonlinear relationships, it might miss the nuanced short-run statistical relationships designed for a time series context that ARIMA models are designed to capture. This potential drawback may prevent it from accurately capturing the intricacies of specific time series.

We solve the problem by using a hybrid approach which supports both model modes. This way we entered the LSTM learning complex relationship while still capturing short term dynamics with ARIMA. In general, this LSTM+ARIMA hybrid model can potentially outperform the individual model best on both sides, resulting in more accurate and stable forecasts.

LSTM Residual

To further enhance the model's ability to capture intricate patterns within the time series, we will utilize the residuals from the LSTM model as input for the ARIMA component. This approach allows the ARIMA to focus on modeling the remaining information not captured by the LSTM, effectively combining the strengths of both models in a complementary manner.

```
ADF Statistic: -4.77257108930703
p-value: 6.135203244217895e-05
Critical Values: {'1%': -3.442383534871275, '5%': -2.8668480382580386, '10%': -2.569597004924258}
Number of significant p values: 6
Number of significant q values: 16
```

Figure 14: ADF Test Results of Residuals

Figure 14 shows the ADF test on the residuals, it shows that the test rejects the null hypothesis, thus confirming that LSTM residuals are stationary (ADF statistic = -4.77 < −3.42 (1% significance levels), −2.87 (5% significance levels), -2.57 (10% significance levels), p-value = 6.135e-05). This result provides very strong evidence against the null hypothesis of non-stationarity and thus we can effectively model the residuals using an ARIMA model without the need for differencing.

Furthermore, 6 significant AR terms, and 16 significant MA terms are being identified based on the threshold calculation. It is easy to tell that unlike the previous 20 AR and 20 MA terms, the complexity is reduced suggesting a potential that the residual ARIMA component can captures both short-term dependencies and the impact of any remaining shocks or outliers not captured by the LSTM, having less possibility to overfitting.

SARIMAX Results

Dep. Variable:	y	No. Observations:	556
Model:	ARIMA(6, 0, 17)	Log Likelihood	382.654
Date:	Wed, 04 Dec 2024	AIC	-715.308
Time:	19:58:31	BIC	-607.288
Sample:	0	HQIC	-673.116
	- 556		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
const	-0.0837	0.078	-1.079	0.281	-0.236	0.068
ar.L1	1.2863	0.227	5.672	0.000	0.842	1.731
ar.L2	-0.1309	0.252	-0.519	0.604	-0.625	0.364
ar.L3	-0.1886	0.224	-0.843	0.399	-0.627	0.250
ar.L4	-0.4031	0.199	-2.021	0.043	-0.794	-0.012
ar.L5	1.1434	0.257	4.453	0.000	0.640	1.647
ar.L6	-0.7520	0.166	-4.525	0.000	-1.078	-0.426
ma.L1	0.1368	0.233	0.587	0.557	-0.320	0.593
ma.L2	-0.1280	0.228	-0.561	0.574	-0.575	0.319
ma.L3	-0.0744	0.202	-0.368	0.713	-0.471	0.322
ma.L4	0.0968	0.219	0.442	0.659	-0.333	0.527
ma.L5	-0.4954	0.181	-2.743	0.006	-0.849	-0.141
ma.L6	-0.2031	0.126	-1.618	0.106	-0.449	0.043
ma.L7	-0.1416	0.096	-1.473	0.141	-0.330	0.047
ma.L8	-0.1622	0.068	-2.367	0.018	-0.296	-0.028
ma.L9	0.2721	0.063	4.322	0.000	0.149	0.396
ma.L10	0.1010	0.061	1.652	0.098	-0.019	0.221
ma.L11	-0.0058	0.071	-0.081	0.935	-0.145	0.133
ma.L12	0.0621	0.069	0.903	0.366	-0.073	0.197
ma.L13	0.0244	0.066	0.369	0.712	-0.105	0.154
ma.L14	0.0768	0.058	1.332	0.183	-0.036	0.190
ma.L15	0.0067	0.058	0.115	0.909	-0.107	0.121
ma.L16	0.0667	0.063	1.061	0.289	-0.057	0.190
ma.L17	-0.0172	0.066	-0.261	0.794	-0.147	0.112
sigma2	0.0147	0.001	18.903	0.000	0.013	0.016

Ljung-Box (L1) (Q):	0.09	Jarque-Bera (JB):	41.71
Prob(Q):	0.77	Prob(JB):	0.00
Heteroskedasticity (H):	0.78	Skew:	0.23
Prob(H) (two-sided):	0.09	Kurtosis:	4.26

Figure 15: ARIMA fit summary (residuals)

A detailed examination of the ARIMA (6, 0, 17) model fitted to the LSTM residuals reveals a between the significance of autoregressive (AR) and moving average (MA) components. The presence of 6 significant AR terms indicates that the model effectively captures dependencies related to past values of the residuals. Specifically, ar.L1, ar.L4, ar.L5, and ar.L6 exhibit p-values below the 0.05 significance threshold, highlighting their crucial role in modeling the temporal dynamics inherent in the residual series.

Furthermore, the model incorporates 17 MA terms, several of which demonstrate statistical significance. Notably, ma.L5, ma.L8, and ma.L9 exhibit p-values below 0.05, indicating their substantial contribution to explaining the remaining errors not accounted for by the LSTM component. Despite many MA terms are not statistically significant, however given consideration the improvement getting from previous ARIMA model (it is important to remember that the data fit into models is different, one is the original data another is the residuals, however still we believe that it helps to show the power of ARIMA model after taking different approach). We believe that the ARIMA model effectively captures nuanced patterns and fluctuations within the residuals that may be attributed to short-term dependencies or the impact of outliers. Thus, aiding in the weakness of LSTM. Consistent with our aim, to highlight the complementary nature of the LSTM and ARIMA components within the hybrid framework, where each model contributes unique strengths to achieve a more holistic representation of the time series data.

```
Mean Absolute Error (Hybrid Model): 0.13285604052956856
Mean Squared Error (Hybrid Model): 0.0285051776100216
Root Mean Squared Error (Hybrid Model): 0.16883476422236504
```

Figure 16: Evaluation Metrics of LSTM + ARIMA

Given from the figure 18, the results are extremely excellent. The LSTM+ARIMA hybrid model, with the lowest RMSE of 0.168834, demonstrates the effectiveness of combining deep learning with traditional statistical modeling. This hybrid approach leverages ARIMA's strength in capturing short-term dependencies and seasonal patterns while utilizing LSTM's capacity for learning complex, long-term, and non-linear dynamics. The dramatic reduction in RMSE reflects the complementary nature of these models and validates the effectiveness of the transformation and integration process.

In terms of statistical analysis, this development highlights the importance of using more sophisticated modelling techniques to achieve higher accuracy predictions. The significant improvement in RMSE from the ARIMA model to the hybrid LSTM+ARIMA model also demonstrates our success in achieving our aims.

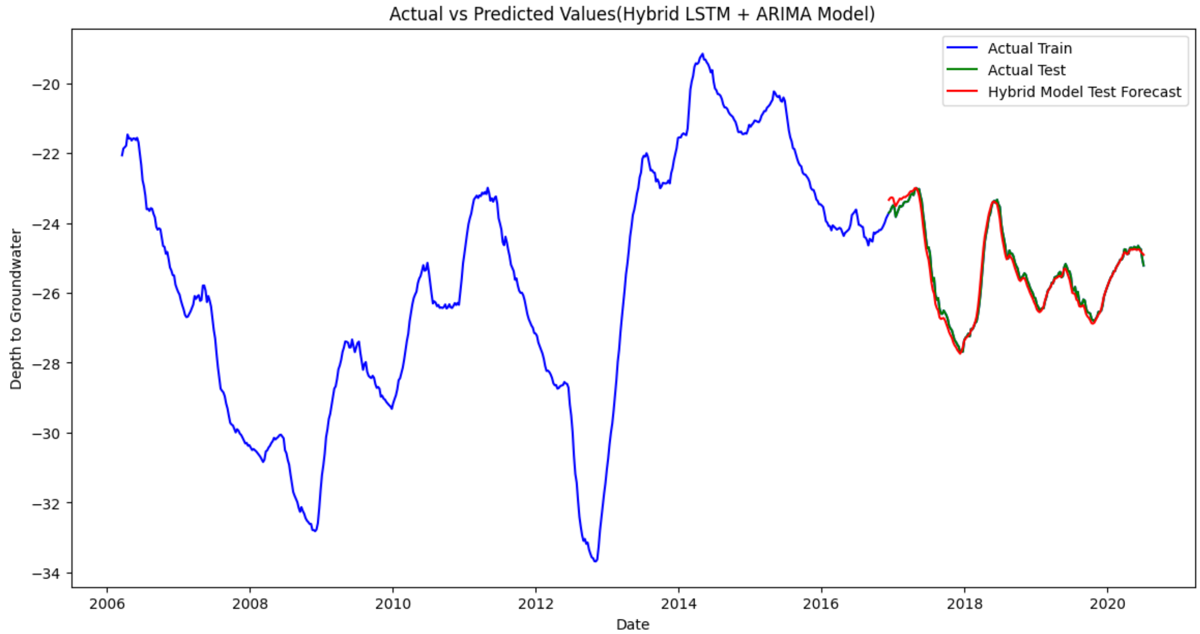

Figure 17: Actual vs Predicted Value of Hybrid Model

Experimental results of hybrid LSTM+ARIMA model in terms of accuracy (Figure16) confirm that the proposed approach improves the predictive performance such that distinctively better forecasting accuracy is achieved (when compared with stand-alone ARIMA model). Figure 17 shows the predicted values (red line) compared to the actual test data (green line), which in this case are very similar, perfectly tracking the non-linear fluctuations and complex temporal patterns typical of the depth-to-groundwater dataset. The performance boost thus observed may be attributed to the fact that this hybrid model is able to leverage the advantages of both components, LSTM is able to capture and represent non-linear dependencies and sequential features within the data while ARIMA further models the residuals to improve the prediction. The hybrid model successfully avoids the nature of ARIMA-only predictions to smooth the forecasts, returning accurate and responsive forecasts for the shifting and trending data observed in the test set. This finding reflects the strength of hybrid model which outperforms for time series problems with complex patterns.

Conclusion

The results suggest that a hybrid LSTM-ARIMA model has the potential to become a powerful tool for accurate and reliable forecasting of water levels in the Petrignano aquifer. This enabled us to successfully capture long-term dependencies, while benefiting from the short-term dynamics captured by the Gated Recurrent Neural Network, due to the combination of the two models' strengths. Using either model alone would allow the prediction of water flows; however, this joint approach demonstrates a considerable improvement in prediction accuracy, leading to better and more sustainable management of water resources. Such an approach can enable the stakeholders to take decisions on early measures to be taken regarding withdrawal of water, drought or pollution control which, in the long run, can contribute to the sustainability of the Petrignano aquifer. Additionally, this project provides a playbook for deploying statistical methods + machine learning approaches for other aquifer systems across the globe. Such innovations will be vital for the next and generations to come to secure water in a time of growing scarcity.

Future research will focus on the following main directions to improve the capabilities of the model and extend its implications: First, we may include additional data sources, such as high-resolution climate projections, regional land-use maps, and socioeconomic data, to give a better overview of the factors influencing the dynamics of the aquifer. Finally, we will examine the incorporation of more sophisticated AI approaches, including deep learning architectures (e.g., convolutional neural networks for modelling of spatial patterns, transformers for long-range dependencies) and ensemble methods, to potentially enhance prediction accuracy and capture intricate non-linear relationships. Third, we will learn how to quantify the uncertainty of the model predictions, allowing for probabilistic prediction and making strong decisions under uncertainty. This would mean using methods like Bayesian Inference, Monte Carlo Simulations, and ensemble forecasting, which will give a range of outcomes and their probabilities. Ultimately, we will create an easy-to-use interface or platform for stakeholders to access forecasts and insights from the model, enabling data-driven water management and sustainable usage of limited resources. Further, engaging in these future directions not only enhances our hybrid mode, but also benefits the potentially highly impactful global fight for sustainable water resource management by protecting locally vital groundwater stores.

References

Anifowose, F. (2020, February 7). Hybrid Machine Learning Explained in Nontechnical Terms. *JPT*. https://jpt.spe.org/hybrid-machine-learning-explained-nontechnical-terms

Banoula, M. (2023, April 12). Introduction to Long Short-Term Memory(LSTM). *Simplilearn*. https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/lstm

GeeksforGeeks. (2023, December 13). Vanishing and Exploding Gradients Problems in Deep Learning. *GeeksforGeeks*. https://www.geeksforgeeks.org/vanishing-and-exploding-gradients-problems-in-deep-learning/

Gülmez, B. (2023). Stock price prediction with optimized deep LSTM network with artificial rabbits optimization algorithm. *Expert Systems with Applications*, *227*, 120346. https://doi.org/10.1016/j.eswa.2023.120346

IBM. (2024, May 24). ARIMA models. *IBM*. https://www.ibm.com/topics/arima-model

Jamshidzadeh, Z., Latif, S. D., Ehteram, M., Khozani, Z. S., Ahmed, A. N., Sherif, M., & El-Shafie, A. (2024). An advanced hybrid deep learning model for predicting total dissolved solids and electrical conductivity (EC) in coastal aquifers. *Environmental Sciences Europe*, *36*(1), 1–19. https://doi.org/10.1186/s12302-024-00850-8

Khalid, H., Khan, A., Khan, M. Z., Mehmood, G., & Qureshi, M. S. (14 March 2023). Machine Learning Hybrid Model for the Prediction of Chronic Kidney Disease. *Computational Intelligence and Neuroscience*, *2023*(1). https://doi.org/10.1155/2023/9266889

Mochurad, L. (2024). An ensemble approach integrating LSTM and ARIMA models for enhanced financial market predictions. *Royal Society Open Science*. https://doi.org/doi:10.1098/rsos.240699

Rao,V(2024, September 18). *LSTM for Enterprise Time Series Forecasting*.LinkedIn. https://www.linkedin.com/pulse/lstm-enterprise-time-series-forecasting-vasu-rao-sdu2c/

Takafuji, E. H. de M., Rocha, M. M. da, & Manzione, R. L. (2018). Groundwater Level Prediction/Forecasting and Assessment of Uncertainty Using SGS and ARIMA Models: A Case Study in the Bauru Aquifer System (Brazil). *Natural Resources Research*, *28*(2), 487–503. https://doi.org/10.1007/s11053-018-9403-6

The Groundwater Project (n.d). Retrieved November 30, 2024, from https://gw-project.org/the-importance-of-groundwater/

Zhang, H., Wang, L., & Shi, W. (2023). Seismic control of adaptive variable stiffness intelligent structures using fuzzy control strategy combined with LSTM. *Journal of Building Engineering*, *78*, 107549. https://doi.org/10.1016/j.jobe.2023.107549

APPENDIX 1

MARKING RUBRICS

		Score and Descriptors				
No.		**Poor**	**Average**	**Excellent**	**Weight (%)**	**Mark**
Component 1: Project Development						
		0-2	**3**	**4-5**	**5**	
1	Use of Data Set	No application pre-processing data.Only use the raw data.	Apply partical complete of data pre-processing.	Apply complete data pre-processing methods on data sets.		
		0-2	**3**	**4-5**	**5**	
1	Code Quality	Design codes that are poorly structured. Not fully functional. Not documented.	Codes are sufficiently documented and mostly functional. Satisfactorily structured.	Codes are fully functional and well structured and documented.		
		0-2	**3**	**4-5**	**5**	
2	Method Functionalities	Incomplete development of the method.	Complete development of method.	Completed\ enhanced the developed methods.		
		0-2	**3-5**	**6-10**	**10**	
3	Performance Evaluation	Lack evaluation of performance and analysis.	Partially complete performance evaluation.	Provide all analysis and performance evaluation.		
				Subtotal	**25**	
Component 2: Project Report + Project Demonstration						
		0 - 6	**7 - 10**	**11-15**	**15**	
1	Project Report	Poor writing quality Poor or no formatting / presentation Lack of discussion for statistical method's results.	Satisfactory writing quality, grammar and flow. Substantial content on the statistical methods.	Good writing quality, grammar and flow Well formatted and good presentation Demonstrate excellent experimental analysis of statistical methods.		
		0-2	**3**	**4-5**	**5**	
1	Presentation	Poor quality slides Poor time management Speech that is unclear.	Satisfactory quality slides Speech that is satisfactory and understandable.	High quality slides Good time management Speech that is clear and impactful.		
		0-2	**3**	**4-5**	**5**	
2	Demonstration	Poor demonstration that is unclear of implementation methods.	Satisfactory demonstration of implementation of methods.	Excellent demonstration is fully functioning and logical.		
				Subtotal	**25**	
				Grand Total	**50**	