# MICROCLIMATE DATA ANALYSIS AT CULTURAL HERITAGE SITES USING THE RANDOM FOREST AND XGBOOST ALGORITHMS

IMAN AIDI ELHAM BIN HAIRUL NIZAM

UNIVERSITI TEKNOLOGI MALAYSIA

PSZ 19:16 (Pind. 1/13)

# UNIVERSITI TEKNOLOGI MALAYSIA

## DECLARATION OF THESIS / UNDERGRADUATE PROJECT REPORT AND COPYRIGHT

| | |
|---|---|
| Author's full name | : Iman Aidi Elham bin Hairul Nizam |
| Date of Birth | : 22 January 2000 |
| Title | : Microclimate Data Analysis at Cultural Heritage Sites Using the Random Forest and XGBoost Algorithms |
| Academic Session | : |

I declare that this thesis is classified as:

□ **CONFIDENTIAL** (Contains confidential information under the Official Secret Act 1972) *

□ **RESTRICTED** (Contains restricted information as specified by the organization where research was done) *

☑ **OPEN ACCESS** I agree that my thesis to be published as online open access (full text)

1. I acknowledged that Universiti Teknologi Malaysia reserves the right as follows:
2. The thesis is the property of Universiti Teknologi Malaysia
3. The Library of Universiti Teknologi Malaysia has the right to make copies for the purpose of research only.
4. The library has the right to make copies of the thesis for academic exchange.

Certified by:

| | |
|---|---|
| **SIGNATURE OF STUDENT** | **SIGNATURE OF SUPERVISOR** |
| A20EC5006 | AP. DR. MOHD SHAHIZAN OTHMAN |
| **MATRIX NUMBER** | **NAME OF SUPERVISOR** |
| Date: 7 JULY 2024 | Date: 7 JULY 2024 |

NOTES : If the thesis is CONFIDENTIAL or RESTRICTED, please attach with the letter from the organization with period and reasons for confidentiality or restriction

“I hereby declare that we have read this thesis and in my opinion this thesis is suffcient in term of scope and quality for the award of the degree of Doctor of Philosophy (Specialization)”

Signature : ____________________________________

Name of Supervisor I : MOHD SHAHIZAN OTHMAN

Date : 7 JULY 2024

**BAHAGIAN A - Pengesahan Kerjasama***

Adalah disahkan bahawa projek penyelidikan tesis ini telah dilaksanakan melalui kerjasama antara Click or tap here to enter text. dengan Click or tap here to enter text.

Disahkan oleh:

Tandatangan : Tarikh :

Nama:

Jawatan:

(Cop rasmi)

** Jika penyediaan tesis atau projek melibatkan kerjasama.*

---

**BAHAGIAN B - Untuk Kegunaan Pejabat Sekolah Pengajian Siswazah**

Tesis ini telah diperiksa dan diakui oleh:

Nama dan Alamat Pcmeriksa Luar **:**

Nama dan Alamat Pcmeriksa Dalam **:**

Nama Penyelia Lain (jika ada) **:**

Disahkan oleh Timbalan Pendaftar di SPS:

Tandatangan : Tarikh :

Nama :

# MICROCLIMATE DATA ANALYSIS AT CULTURAL HERITAGE SITES USING THE RANDOM FOREST AND XGBOOST ALGORITHMS

IMAN AIDI ELHAM BIN HAIRUL NIZAM

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Bachelor of Computer Science (Software Engineering)

School of Computing
Faculty of Engineering
Universiti Teknologi Malaysia

JULY 2024

# DECLARATION

I declare that this thesis entitled *“ Microclimate Data Analysis at Cultural Heritage Sites Using the Random Forest and XGBoost Algorithms”* is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

| | | |
|---|---|---|
| Signature | : | .................................................... |
| Name | : | Iman Aidi Elham bin Hairul Nizam |
| Date | : | 7 JULY 2024 |

# DEDICATION

This thesis is dedicated to my parents who taught me to work hard and dream big in life. Thank you to my supervisor Associate Prof Dr Mohd Shahizan Othman for guiding me throughout this thesis. Thank you too to my supportive friends who are also struggling to finish their own thesis and has helped me with this thesis either physically or morally.

# ACKNOWLEDGEMENT

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. I wish to express my sincere appreciation to my main thesis supervisor, Professor Dr. Mohd Shahizan Othman, for encouragement, guidance, critics, and friendship. Without his continued support and interest, this thesis would not have been the same as presented here.

I am also indebted to Universiti Teknologi Malaysia (UTM) for funding my Bachelor study. Librarians at UTM also deserve special thanks for their assistance in supplying the relevant literatures.

My fellow undergraduate student should also be recognised for their support. My sincere appreciation also extends to all my colleagues and others who have helped at various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space. I am grateful to all my family member.

# ABSTRACT

This study aims to develop an accurate prediction model for microclimate data in the heritage-rich cities of Johor Bahru and Melaka, Malaysia. The research focuses on key microclimate variables including temperature, rainfall, humidity, and wind speed. Historical data will be obtained from the Copernicus Climate Data Store (CDS) to train and validate the prediction models. The study will compare the performance of two machine learning algorithms, Random Forest and XGBoost, to determine the most effective method for microclimate prediction in these specific urban environments. A user-friendly dashboard will be developed using HTML to visualize both historical data and predictions, making the information accessible and interpretable. The accuracy and reliability of the prediction models will be evaluated using standard statistical measures including Mean Absolute Error, Root Mean Square Error, and R-squared score. The models will be trained using cross-validation techniques for hyperparameter tuning, and their performance will be assessed on a held-out test set representing the year 2023. This research aims to provide a valuable tool for local meteorologists, urban planners, and researchers interested in the microclimatic conditions of Johor Bahru and Melaka. The resulting predictive model and dashboard could serve as a foundation for future studies on the impact of microclimate on urban planning and heritage conservation in these historically significant Malaysian cities.

# ABSTRAK

Pemeliharaan senibina warisan memegang peranan penting dalam mengekalkan warisan budaya suatu kawasan. Kajian ini memberi tumpuan kepada pembangunan sistem pemantauan mikroiklim berasaskan pembelajaran mesin untuk membantu pihak berkuasa tempatan di Johor Bahru dan Melaka, merancang tindakan penyelenggaraan pencegahan untuk tapak warisan yang ditetapkan. Projek penyelidikan ini merangkumi mendapatkan data mikroiklim, termasuk suhu, taburan hujan, kelembapan, dan kelajuan angin, daripada Copernicus Climate Data Store (CDS). Untuk mengoptimumkan proses pemantauan dan ramalan, prestasi dua algoritma pembelajaran mesin, Random Forest dan XGBoost akan dibandingkan untuk menentukan kaedah yang paling sesuai untuk analisis mikroiklim. Projek ini juga melibatkan reka bentuk dan pembangunan papan pemuka yang memaparkan data mikroiklim masa nyata menggunakan alat visualisasi data. Keberkesanan algoritma dan papan pemuka yang dibangunkan akan diuji untuk menilai potensi mereka dalam membantu pihak berkuasa tempatan melaksanakan rancangan penyelenggaraan yang lebih berkesan untuk tapak warisan. Penyelidikan ini bertujuan untuk menyumbang kepada pemeliharaan tapak warisan budaya dengan menggunakan teknik pembelajaran mesin yang maju untuk pemantauan dan ramalan mikroiklim, yang pada akhirnya menyokong usaha-usaha konservasi yang mampan dan cekap.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# LIST OF ABBREVIATIONS

| | | |
|---|---|---|
| ML | - | Machine Learning |
| AI | - | Artificial Intelligence |
| RF | - | Random Forest |
| CH | - | Cultural Heritage |
| CSV | - | Comma-separated Values |
| LR | - | Logistic Regression |
| ANN | - | Artificial Neural Network |
| CNN | - | Convolutional Neural Network |
| KNN | - | K-Nearest Neighbour |
| XGBoost | - | Extreme Gradient Boosting |
| MAE | - | Mean Absolute Error |
| RMSE | - | Root Mean Square Error |
| R-squared | - | Coefficient of Determination |

# CHAPTER 1

# INTRODUCTION

## 1.1 Overview

Cultural heritage sites are the basis for our global and historical values. They connect us to the traditions left by our ancestors and contribute significantly to the cultural identity of human society (Lombardo et al., 2020). The preservation of cultural heritage, whether it be buildings or artifacts, is subject to various risks of damage and deterioration that result from microclimate conditions in the surrounding environment. These conditions are determined by several factors, including microclimate parameters such as temperature, humidity, airborne pollutants concentrations, air speed, and others (Fabbri & Bonora, 2021). Particularly in developing nations, these impacts pose a significant challenge to the preservation of cultural heritage (Pioppi et al., 2020). Safeguarding worldwide cultural heritage sites is of utmost importance for preserving cultural identity and human heritage, as well as promoting cultural and tourism-driven economic development (Alcaraz Tarragüel et al., 2012).

In recent years, the administration of cultural heritage sites and monuments has gained worldwide focus through the implementation of detection, monitoring, and comprehensive assessment methods. Initiatives are also underway to enhance and preserve these heritage resources by adopting suitable adaptation measures and sustainable management approaches (Guzman et al., 2020). To address these challenges, this thesis focuses on the application of advanced machine learning algorithms, namely Random Forest and XGBoost, for microclimate monitoring and prediction at cultural heritage sites. By leveraging these techniques, it aims to contribute to the preservation of cultural heritage sites under changing environmental conditions, supporting sustainable and efficient conservation efforts.

## 1.2 Problem Background

Cultural heritage sites have consistently drawn visitors who seek to spend quality time and pursue unique experiences by engaging with local cultures and communities (Ramkissoon et al., 2013). As a result, the economies of these tourist destinations largely rely on attracting visitors, encouraging repeat visits, garnering recommendations, and generating positive word-of-mouth regarding the locations (Rezapouraghdam et al., 2021). In addition, the natural environments in which tourism activities occur are also enhancing the well-being and quality of life for residents (Ramkissoon et al., 2018). Lately, Johor Bahru and Melaka have been experiencing frequent climate fluctuations that negatively impact the aesthetic appeal of the area's heritage sites, significantly affecting the industry of tourism and local economy. Generally, microclimate changes in these regions cause substantial damage to cultural heritage sites and various monuments. Consequently, striking a balance between consumption and conservation strategies presents increasing challenges for the effective management of cultural heritage sites (Buonincontri et al., 2017). Therefore, focusing on the preservation of cultural heritage and promoting sustainable tourism has become a primary objective recently to support both cultural heritage tourism and the overall well-being of communities (Megeirhi et al., 2020).

## 1.3 Research Aim

The goal of this study is to analyze vulnerable zones of cultural heritage sites and monuments in Johor Bahru and Melaka, by employing microclimate monitoring and prediction through the Random Forest and XGBoost algorithms. By assessing temperature, rainfall, humidity, and wind speed, the study aims to maintain environmental sustainability at these heritage sites. In this research, we have prepared a microclimate monitoring dashboard and evaluated the significance of factors contributing to microclimate changes. The Random Forest and XGBoost algorithms were employed to analyze the impact of these factors on the preservation of cultural heritage sites.

## 1.4 Research Objectives

The following are the objectives proposed:

(a) To identify and compare the most suitable machine learning algorithms for analyzing microclimate data in cultural heritage sites.

(b) To evaluate the performance and accuracy of the developed machine learning models in predicting microclimate conditions.

(c) To design and develop a user-friendly dashboard for displaying microclimate data trends and predictions for heritage site management.

## 1.5 Research Scopes

The scope of this research project focuses on the preservation of the Sultan Ibrahim Building in Johor Bahru and A Famosa in Melaka using machine learning-based microclimate prediction. The primary objectives are to develop machine learning algorithms and a dashboard to collect, display, and analyze microclimate parameters, thereby assisting local authorities in planning preventive maintenance for these heritage sites. Specific areas included in the scope of this research are:

(a) Obtaining microclimate data from the Copernicus Climate Data Store (CDS) for the Sultan Ibrahim Building and A Famosa. The data will include parameters such as temperature, rainfall, humidity, and wind speed.

(b) Comparing the performance of two machine learning algorithms, Random Forest and XGBoost, to determine the most suitable method for microclimate monitoring and prediction.

(c) Designing and developing a dashboard using HTML and JavaScript to display and analyze historical microclimate data trends and predictions.

(d) Evaluating the accuracy and effectiveness of the developed machine learning models and dashboard in assisting local authorities with planning more effective maintenance strategies for the heritage sites.

## 1.6 Research Contribution

A thorough literature review on microclimate impacts on cultural heritage sites reveals that many researchers have utilized various statistical and machine learning methods, including Logistic Regression (LR), Artificial Neural Network (ANN), Convolutional Neural Network (CNN), K-Nearest Neighbor (KNN), and Support Vector Machine (SVM), to create microclimate monitoring and prediction dashboards. However, the combination of Random Forest and XGBoost algorithms, along with the comprehensive analysis of temperature, humidity, and wind speed, has not yet been applied in the context of heritage site preservation. Therefore, this study offers a novel contribution to the machine learning field, particularly in modeling microclimate threats and conducting risk assessments for cultural heritage sites.

Given the current changing climate and landscape, this study is highly relevant and significantly contributes to the sustainable management of cultural heritage resources. Climate change poses a considerable threat to the integrity of heritage sites due to its impact on key environmental factors such as temperature, rainfall, humidity, and wind speed. These changes can increase the vulnerability and potential damage to cultural assets. This study provides valuable insights and technical guidance on the appropriate machine learning algorithms, and proper interpretation and evaluation of outcomes, which can inform future research and decision-making processes.

Moreover, this study has essential implications for the conservation of natural resources and heritage sites in Johor Bahru and Melaka. The findings are expected to have practical applications for professionals involved in land use planning, landscape management, archaeological preservation, and public administration. These professionals can utilize the study’s evidence-based strategies to manage cultural heritage sites and promote environmental sustainability effectively. By monitoring and predicting microclimate changes using Random Forest and XGBoost algorithms, stakeholders can better preserve and protect cultural heritage sites for future generations.

## 1.7 Report Organization

This report is organized into six chapters. Chapter 1 introduces the topic of preserving cultural heritage sites through microclimate monitoring and prediction using Random Forest and XGBoost algorithms, along with the research background, objectives, and the purpose of the study in Johor Bahru and Melaka. Chapter 2 reviews relevant literature on microclimate monitoring, including the assessment of temperature, rainfall, humidity, and wind speed, and compares various machine learning techniques for processing and analyzing data from heritage sites. Chapter 3 outlines the research methodology, detailing how the study employs Random Forest and XGBoost algorithms to measure and analyze microclimate data for the preservation of cultural heritage sites. Chapter 4 presents the experimental setup and results, explaining how the experiments were conducted to derive insights from the microclimate data. Chapter 5 focuses on the development of the dashboard, showcasing the analyzed data and visualizing trends and predictions to provide a practical tool for local authorities. Finally, Chapter 6 summarizes the study, highlighting key findings and implications for the preservation of cultural heritage sites through microclimate monitoring and prediction, and offers recommendations for future research.

# CHAPTER 2

# LITERATURE REVIEW

## 2.1 Introduction to Case Study

Climate change has emerged as a significant global challenge, impacting various sectors, including the preservation of cultural heritage sites. The increasing frequency and intensity of extreme weather events, along with gradual shifts in temperature, humidity, and wind patterns, underscore the need for adaptive solutions to safeguard these invaluable assets. One promising approach involves applying advanced algorithms, such as Random Forest and XGBoost, for microclimate monitoring and prediction at cultural heritage sites. These techniques can help preserve and protect these valuable assets by analyzing temperature, humidity, and wind speed data, crucial factors in their conservation.

This case study focuses on the implementation of Random Forest and XGBoost algorithms for microclimate monitoring and prediction at two cultural heritage sites: the Sultan Ibrahim Building in Johor Bahru and A Famosa in Melaka. By leveraging these advanced techniques and developing dashboards for data visualization and analysis, this research aims to enhance the understanding of site-specific microclimates and inform effective conservation strategies for these historic landmarks.

Through continuous assessment and refinement of these methods, researchers, conservators, and heritage site managers can collaborate to develop improved strategies for preserving cultural heritage sites like the Sultan Ibrahim Building and A Famosa under changing environmental conditions. By adopting a collaborative approach, we can ensure the protection and preservation of these invaluable assets for future generations, despite the challenges posed by climate change.

## 2.2 Importance of Preserving Cultural Heritage Sites

The preservation of cultural heritage sites holds immense significance for society, history, and identity, as these sites serve as tangible reminders of our shared past, providing valuable insights into the cultural, social, and economic development of human civilizations (Lowenthal, 1985). By protecting and maintaining these sites, we ensure the continuity of our cultural memory and allow future generations to appreciate and learn from the rich tapestry of human history (UNESCO, 1972). Moreover, cultural heritage sites contribute to a sense of belonging and pride within communities, fostering social cohesion and promoting intercultural dialogue (Smith, 2006). Furthermore, preserving these sites can offer economic benefits, as they often attract tourism and stimulate local economies (Timothy & Boyd, 2003). Given these multifaceted advantages, it is crucial to develop and implement strategies to safeguard cultural heritage sites against various threats, including the impact of microclimate factors, to ensure their longevity and continued cultural relevance.

## 2.3 Impact of Microclimate Factors on Cultural Heritage Sites

Microclimate factors, such as humidity, rainfall, temperature and wind speed play a significant role in the deterioration of cultural heritage sites. Existing studies have established the adverse effects of these factors on various materials and structures, leading to both physical and chemical degradation (Cassar, 2005; Camuffo, 2014).

Temperature fluctuations, especially in the presence of moisture, can lead to the expansion and contraction of materials like stone, brick, and mortar, resulting in cracks, delamination, and structural damage (Camuffo, 2014). Moreover, extreme temperatures can accelerate the decay of organic materials, such as wood and textiles, commonly found in cultural heritage sites (Cassar, 2005).

Humidity is another critical factor in the deterioration process. High humidity levels can cause moisture to accumulate in porous materials, leading to the growth of mold and bacteria, which can weaken and damage the structure (Lankester &

Brimblecombe, 2012). Additionally, the presence of moisture can facilitate the dissolution of soluble salts in porous materials, causing efflorescence and sub florescence, further compromising structural integrity (Cassar, 2005).

Wind speed, particularly in combination with rain, can exacerbate the erosion of building materials and increase the rate of material loss from structures (Cassar, 2005). Moreover, high wind speeds can cause physical damage to fragile elements, such as decorative features and stained-glass windows (Camuffo, 2014).

In summary, understanding the impact of microclimate factors on cultural heritage sites is crucial for developing effective preservation strategies. By identifying and mitigating the risks associated with these factors, we can better protect these invaluable resources and ensure their continued existence for future generations.

## 2.4 Traditional Methods for Cultural Heritage Sites Preservation

Traditional methods for cultural heritage site preservation often rely on reactive maintenance approaches. These approaches involve responding to issues and damage after they have already occurred, rather than anticipating and preventing them. Reactive maintenance has several limitations, making it necessary to explore initiative-taking and preventive measures for the preservation of cultural heritage sites (Staniforth, 2013).

Delayed intervention is one of the limitations of reactive maintenance, as it occurs after the damage has been detected, leading to further deterioration or irreversible loss of cultural elements (Muñoz Viñas, 2002). Additionally, reactive maintenance can be expensive, especially if the damage requires extensive interventions and specialized expertise (Stovel, 2005). Incomplete recovery can also be an issue, as advanced damage can result in the loss of original features or materials, compromising the site's authenticity and historical value (Muñoz Viñas, 2002). Moreover, interventions during reactive maintenance can be invasive or destructive, leading to further damage or exposing other areas to new risks (Matero, 1999).

To address these limitations, there is a need to shift towards initiative-taking and preventive measures, such as regular monitoring, preventive conservation, maintenance planning, and capacity building for local stakeholders. Regular inspections and monitoring can help identify early signs of deterioration or potential threats (Caple, 2008), while preventive conservation can reduce or eliminate risk factors contributing to the site's deterioration, such as controlling humidity and temperature (Muñoz Viñas, 2002). Maintenance planning, including preventive measures and timely interventions, can also help address potential issues (Caple, 2008). Capacity building through training and education for local stakeholders can further enhance the site's preservation efforts (Ashley-Smith, 2016).

In conclusion, the preservation of cultural heritage sites requires a shift towards initiative-taking and preventive measures, which can minimize the risk of irreversible damage, maintain the site's authenticity and historical value, and reduce the overall cost of preservation efforts. By adopting regular monitoring, preventive conservation, maintenance planning, and capacity building for local stakeholders, cultural heritage sites can be better preserved for future generations (Muñoz Viñas, 2002; Caple, 2008; Ashley-Smith, 2016).

## 2.5 Machine Learning Algorithms

This study develops machine learning-based methods for microclimate monitoring and prediction at cultural heritage sites, using the supervised learning concept. This involves training a regressor to assign labels to specific data points or regions in the dataset, enabling it to identify hidden patterns and signatures of various labelled factors and make accurate predictions. To ensure effective monitoring and prediction using a variety of data sources, it is crucial to use classifiers that can manage large-scale data and achieve high accuracy quickly. The study focuses on two regressor, XGBoost and Random Forest, which are both capable of achieving these requirements.

### 2.5.1 Random Forest

Breiman's Random Forest algorithm, introduced in 2001, is a widely used ensemble learning model that is known for its versatility in performing various tasks such as classification, regression, clustering, interaction detection, and variable selection (Rahmati et al., 2017; Belgiu and Drăguţ, 2016). This learning method leverages the aggregation of decision trees, which divide input data based on specific parameters in a tree-like structure (Ma and Cheng, 2016; Breiman, 2001) (Figure 1). Unlike other learning methods, Random Forest is designed to manage complex datasets with high dimensionality, noisy, and missing data, making it particularly useful for microclimate monitoring and prediction at cultural heritage sites.

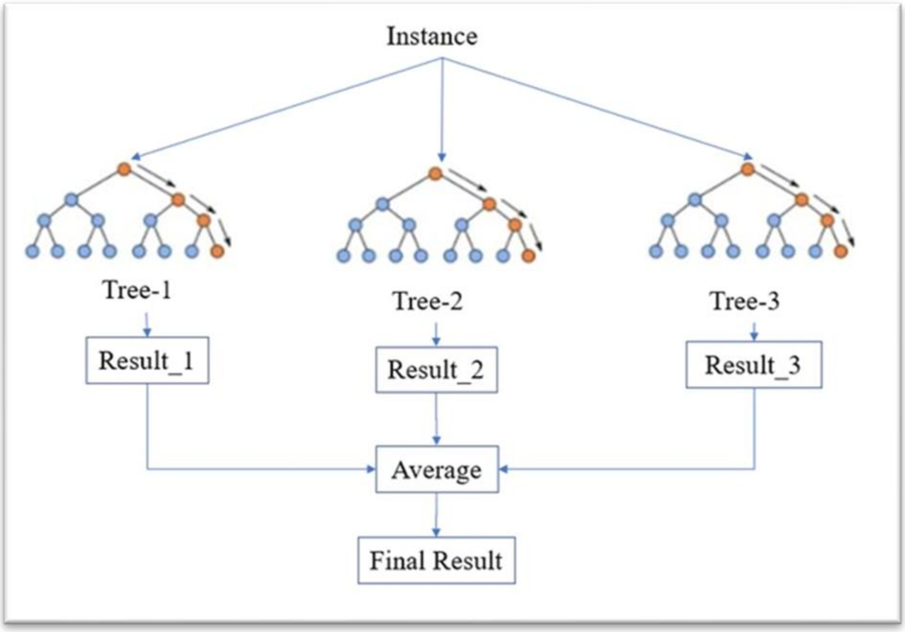


Figure 1: Random Forest Model Architecture

Each decision tree in a Random Forest model is built using a bootstrapped sample of the data, with nodes split according to the optimal subset and randomly selected predictors at each stage (Araki et al., 2018; Rahmati et al., 2017). The final classification is based on the majority vote of the decision trees, and output is generated accordingly (Micheletti et al., 2014; Rahmati et al., 2017). This approach helps prevent overfitting, where a model learns the training data too well and fails to generalize well to new data. Random Forest's robustness and high-performance capabilities have made it a popular choice in various fields, including image analysis, remote sensing, and ecology.

Furthermore, Random Forest is a highly flexible model that can manage a wide range of input variables, including categorical and continuous variables, and can deal with missing data points by imputing values. The algorithm can also determine the importance of each input variable in predicting the output, enabling researchers to identify the most influential parameters for microclimate monitoring and prediction at cultural heritage sites. Additionally, researchers have developed various extensions and modifications to improve the algorithm's efficiency, such as parallel computing, pruning techniques, and feature importance measures (Balogun et al., 2021; Tella et al., 2021).

One notable feature of Random Forest is its ability to manage interactions between variables, which is important in predicting microclimate parameters at cultural heritage sites. The algorithm can identify and model complex interactions between multiple variables, allowing researchers to better understand the relationships between environmental factors and microclimate patterns. This feature is particularly valuable in cultural heritage sites, where environmental conditions can vary significantly and interactions between environmental factors can be complex.

In conclusion, Random Forest is a powerful machine learning algorithm that has proven to be a valuable tool for microclimate monitoring and prediction at cultural heritage sites. Its robustness, flexibility, and high-performance capabilities make it an attractive choice for managing complex datasets with high dimensionality, noisy, and missing data. Moreover, the algorithm's ability to identify and model interactions between variables provides researchers with valuable insights into the complex relationships between environmental factors and microclimate patterns.

### 2.5.2 XGBoost

XGBoost is a popular machine learning algorithm that is commonly used for classification tasks. It belongs to the family of boosting algorithms, where multiple weak learners are combined to create a strong model. The algorithm works by iteratively adding decision trees to the model and adjusting their weights based on the error rate of the previous trees (Figure 2). The result is a highly accurate classifier that can manage large and complex datasets.

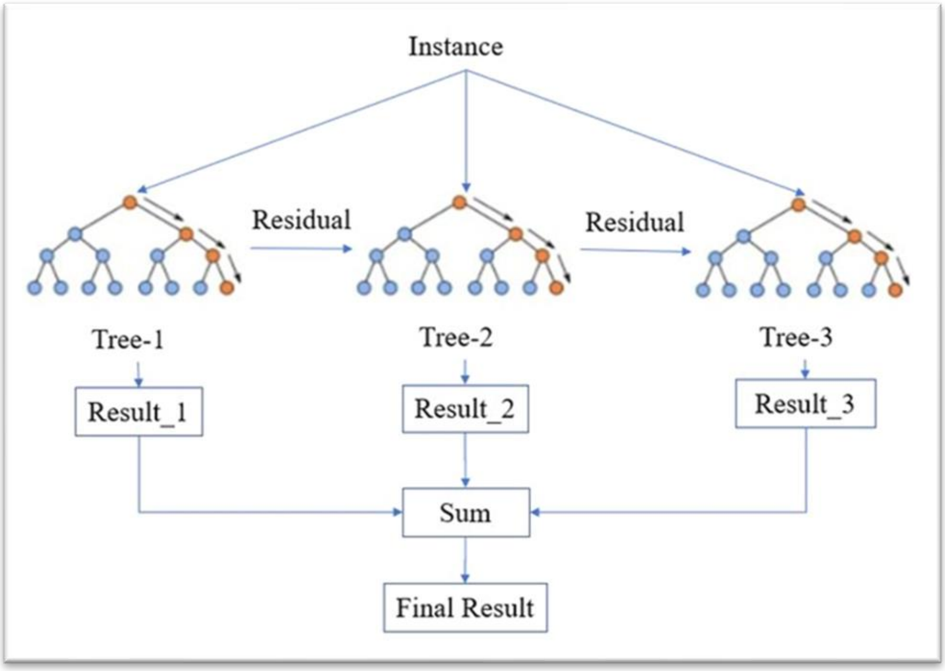


Figure 2: XGBoost Model Architecture

One of the key advantages of XGBoost is its ability to manage missing data effectively. The algorithm can use surrogate splits to compensate for missing data points, resulting in improved accuracy and robustness in the presence of missing data. XGBoost is also highly optimized for parallel computing, enabling it to process large volumes of data quickly and efficiently.

XGBoost has demonstrated high performance and accuracy when dealing with large-scale, multi-class data in various fields, including remote sensing, medical diagnosis, and natural language processing. Studies have shown that XGBoost can outperform other popular classification algorithms, such as Random Forest and Support Vector Machines (SVM), in terms of accuracy and efficiency (Bhagwat & Shankar, 2019; Zamani Joharestani et al., 2019; Rumora et al., 2020). This makes it an attractive choice for microclimate monitoring and prediction at cultural heritage sites, where large datasets and high-dimensional feature spaces are common.

XGBoost is highly scalable, which makes it an ideal choice for managing large volumes of satellite data. This enables researchers to perform microclimate monitoring and prediction in real-time, providing valuable insights into the environmental conditions at cultural heritage sites. Additionally, XGBoost is highly optimized for feature selection, allowing researchers to identify the most influential variables for microclimate monitoring and prediction.

In conclusion, XGBoost is a powerful machine learning algorithm that offers several unique advantages for microclimate monitoring and prediction at cultural heritage sites. Its ability to manage missing data, parallel computing, scalability, and feature selection capabilities make it an attractive choice for researchers and practitioners in this field. By leveraging XGBoost's powerful capabilities, researchers can gain valuable insights into the environmental conditions at cultural heritage sites, enabling them to develop more effective strategies for managing and preserving these invaluable assets for future generations.

## 2.6 Comparative Analysis of Previous Case Studies and the Uses of Machine Learning in Cultural Heritage Preservation

Several studies in the cultural heritage field apply machine learning (ML) techniques for tasks such as automatic text recognition, image annotation, and user preference recommendations. However, the use of ML in conservation science and heritage preservation studies is limited. These studies primarily focus on identifying and classifying materials or structures or using ML to monitor cultural heritage collections or sites for abnormalities. For instance, Zou et al. employed deep learning on image data to locate missing or damaged heritage components in historical buildings, while Kejser et al. used ML to classify the acidity of historic paper samples. Pei et al. utilized machine learning to predict household mite infestation based on indoor climate conditions and found that the extreme gradient boosting (XGBoost) model was the most suitable approach.

Table 1: Comparative Analysis of Previous Case Studies and the Uses of Machine Learning in Cultural Heritage Preservation

| **Case Study** | **Method Used** | **Target Site/Subject** | **Main Outcomes** |
|---|---|---|---|
| **Yu et al. (2022)** | Convolutional Neural Network Deep Learning | Dunhuang Mogao Grottoes, China | Detected wall painting deterioration; informed preventive measures |
| **(Kumar et al. (2019)** | Logistic Regression, Support Vector Machine | Damaged Heritage Sites from 2015 Nepal Earthquake | Classify heritage and not-heritage sites; damage or no damage |
| **Prieto et al., (2017)** | Multiple Linear Regression, Fuzzy Logic Models | 100 parish churches, located in Seville, Spain | Identifies relevant variables for the functional degradation of the churches. |
| **Gonthier et al., (2019)** | Support Vector Machines | Child Jesus, the crucifixion of Jesus, Saint Sebastian | Recognition of iconographic elements in artworks. |
| **Valero et al., (2019)** | Logistic Regression, Multi Class Classification | Chapel Royal in Stirling Castle, Scotland | Identifies loss of material defects and discoloration on the walls. |

## 2.7 Implementation of Random Forest and XGboost in Microclimate Monitoring and Prediction

Researchers have been exploring the performance of XGBoost and Random Forest algorithms for microclimate monitoring and prediction in numerous studies. These algorithms have proven to be effective in providing valuable insights for monitoring and managing microclimate factors in different environments.

In a study by J. Angelin Jebamalar & A. Sasi Kumar (2019), a hybrid light tree and light gradient boosting model were used for predicting PM2.5 levels. The proposed method captured PM2.5 data using a sensor with Raspberry Pi and stored it in the cloud, where the hybrid model was used for prediction. The hybrid model outperformed other algorithms, including Linear Regression, Lasso Regression, Support Vector Regression, Neural Network, Random Forest, Decision Tree, and XGBoost. Despite its advantages in handling substantial amounts of data and requiring less space, the hybrid model's limitation was its time-consuming nature.

In a study by Maryam Aljanabi (2020), the authors compared Multilayer Perceptron, XGBoost, Support Vector Regression, and Decision Tree Regressor to predict ozone levels based on temperature, humidity, wind speed, and wind direction. After pre-processing the data and performing feature selection, XGBoost emerged as the superior model for predicting ozone levels on a day-to-day basis.

Soubhik et al. (2018) compared various algorithms, including Linear Regression, Neural Network Regression, Lasso Regression, ElasticNet Regression, Decision Forest, Extra Trees, Boosted Decision Tree, XGBoost, K-Nearest Neighbor, and Ridge Regression, to predict air pollutant levels. They found that XGBoost provided better accuracy due to the arrangement of features in decreasing order of importance for predicting upcoming values. Haotian Jing & Yingchun Wang (2020) used XGBoost to predict the air quality index. By employing weak classifiers and using the shortcomings of previous weak classifiers to form a strong classifier, XGBoost reduced the error between predicted and actual values. However, it was susceptible to outliers and unwanted air pollutants, as it took the previous value into account.

Mejía et al. (2018) determined PM10 levels best with Random Forest but found that it did not accurately predict the levels of dangerous pollutants. However, Random Forest had the advantage of working with incomplete datasets. Pasupuleti et al. (2020) compared Decision Tree, Linear Regression, and Random Forest for predicting air pollutant levels using meteorological conditions and data from the Arduino platform. Random Forest provided more accurate results due to reduced overfitting and error. However, it required more memory and incurred higher costs.

In summary, XGBoost and Random Forest have been applied in various case studies for microclimate monitoring and prediction, with both algorithms demonstrating their effectiveness in predicting air pollutant levels. While they have their respective limitations, these advanced techniques offer valuable tools for researchers and practitioners seeking to understand and manage the air quality in different environments.

### 2.7.1 Comparison Between Random Forest and XGBoost Algorithms

Table 2: List of Difference between Random Forest and XGBoost Algorithms

| Criteria | XGBoost | Random Forest |
|---|---|---|
| **Model Type** | Gradient boosting decision tree ensemble (Chen & Guestrin, 2016) | Decision tree ensemble (Breiman, 2001) |
| **Learning Approach** | Gradient boosting, optimizing loss function (Friedman, 2001) | Bagging, independent decision trees combined through majority voting or averaging (Liaw & Wiener, 2002) |

| | | |
|---|---|---|
| **Managing Missing Data** | Imputation or treating missing values as separate categories (Chen & Guestrin, 2016) | Imputation or treating missing values as separate categories (Breiman, 2001) |
| **Overfitting Prevention** | Shrinkage and regularization (Chen & Guestrin, 2016) | Averaging results of multiple decision trees (Breiman, 2001) |
| **Interpretability** | Can provide feature importance information | Easier to interpret due to simpler decision tree structure (Breiman, 2001) |
| **Speed and Scalability** | Slower in training due to sequential nature (Chen & Guestrin, 2016) | Faster and more parallelizable due to independent tree construction (Breiman, 2001) |
| **Performance** | Compare using MAE, RMSE, R-squared (Caruana & Niculescu-Mizil, 2006) | Compare using MAE, RMSE, R-squared (Caruana & Niculescu-Mizil, 2006) |
| **Feature Importance** | Can rank input variables by importance (Chen & Guestrin, 2016) | Can rank input variables by importance (Breiman, 2001) |
| **Hyperparameter Tuning** | Requires tuning, may be more sensitive to hyperparameter settings (Probst, Wright, & Boulesteix, 2019) | Requires tuning, may be less sensitive to hyperparameter settings (Probst, Wright, & Boulesteix, 2019) |
| **Memory Usage** | Less memory usage due to sequential nature (Chen & Guestrin, 2016) | More memory usage due to storage of multiple decision trees (Breiman, 2001) |

## 2.8 Chapter Summary

Through this chapter, the study focuses on the preservation of cultural heritage sites through machine learning-based microclimate monitoring, with a specific focus on the application of Random Forest and XGBoost algorithms at two heritage sites in Johor Bahru and Melaka. The review begins with an overview of the impact of climate change on cultural heritage sites and the importance of their preservation. It then explores the impact of microclimate factors on cultural heritage sites, including temperature, rainfall, humidity, and wind speed, and the traditional reactive methods used for preservation. The limitations of reactive maintenance and the need for a shift towards proactive and preventive measures are discussed, such as regular monitoring and preventive conservation. Finally, the review explains the use of machine learning algorithms in microclimate monitoring and prediction, specifically Random Forest and XGBoost, and their application in this study. The review highlights the significance of using machine learning-based approaches for preserving cultural heritage sites and the potential benefits of incorporating them into preservation strategies.

# CHAPTER 3

# RESEARCH METHODOLOGY

## 3.1 Introduction

This section delves into the methodological approach employed in the study, encompassing the research design framework, procedures, and techniques utilized. The investigation follows a four-phase workflow, which will be explored in detail throughout this chapter. Each component of the framework will be thoroughly examined, providing insights into its implementation within the context of the research. Additionally, the specific techniques applied at every stage of the process will be detailed. This comprehensive overview aims to offer a clear understanding of the systematic approach underpinning the study, detailing how each step contributes to addressing the research objectives.

## 3.2 Research Framework

The research methodology framework has four main phases which are Literature Review, Data Collection and Preprocessing, Machine Learning Model Development and Dashboard Development. The framework starts with Phase 1 which is the literature review of the study, it related to the process of gathering information related to microclimate monitoring and prediction and machine learning techniques as shown in Figure 3. The framework then continues with the second phase which is the data collection and preprocessing phase of the research. The next phase is the machine learning model development phase which is the part where the algorithm is develop. The last phase is the dashboard development and analysis of result phase where end results will be reported.

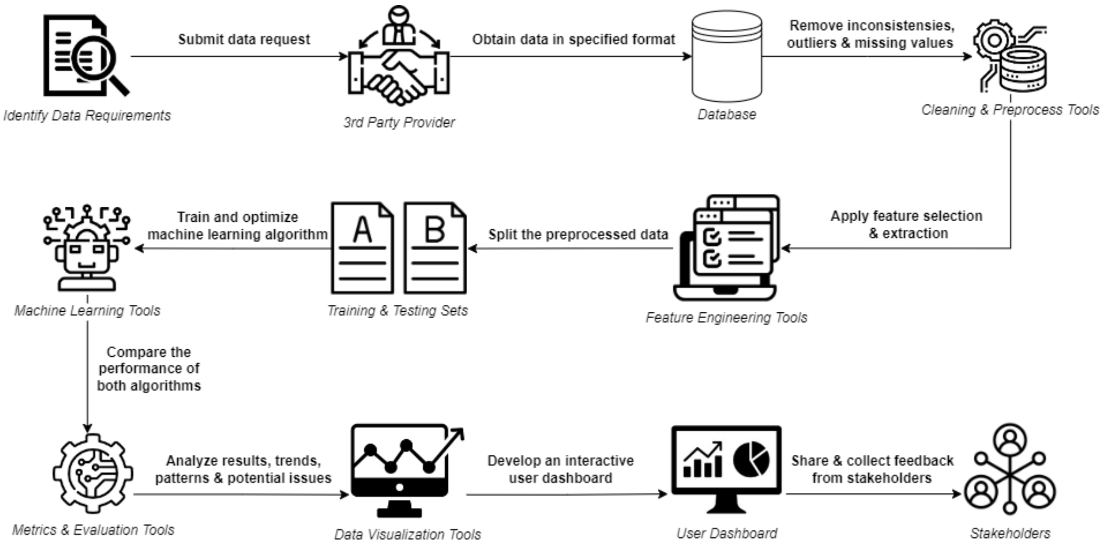

Identify Data Requirements
Submit data request
3rd Party Provider
Obtain data in specified format
Database
Remove inconsistensies, outliers & missing values
Cleaning & Preprocess Tools
Apply feature selection & extraction
Feature Engineering Tools
Split the preprocessed data
A
B
Training & Testing Sets
Train and optimize machine learning algorithm
Machine Learning Tools
Compare the performance of both algorithms
Metrics & Evaluation Tools
Analyze results, trends, patterns & potential issues
Data Visualization Tools
Develop an interactive user dashboard
User Dashboard
Share & collect feedback from stakeholders
Stakeholders

Figure 3: Research Framework

### 3.2.1 Phase 1: Literature Review

In the first phase, a comprehensive literature review is conducted to gather relevant information on preserving cultural heritage sites through microclimate monitoring and prediction using Random Forest and XGBoost algorithms. Various scholarly sources, including journals, articles, and theses, are explored to understand the current state of research in this field. The literature review delves into topics such as data collection methods, pre-processing techniques, and the application of machine learning models. By examining existing studies, this phase helps identify gaps, challenges, and potential solutions for effectively monitoring and predicting microclimate conditions at heritage sites. The insights gained from the literature review form the foundation for the subsequent phases and guide the research towards developing a robust methodology.

### 3.2.2 Phase 2: Data Collection and Preprocessing

In the second phase, the focus shifts to obtaining microclimate data from the Copernicus Climate Data Store (CDS) for a specific heritage site in Johor Bahru. This data, encompassing temperature, relative humidity, precipitation, and wind speed measurements, serves as the basis for subsequent analysis and modelling. To ensure data quality, a rigorous pre-processing stage is undertaken, which involves cleaning the raw data and addressing any missing values or outliers. Techniques such as interpolation, statistical analysis, and data imputation are employed to enhance the integrity and accuracy of the collected data. Additionally, feature engineering techniques are applied to extract meaningful features from the raw data, enabling the capturing of temporal dependencies and relationships between variables. This phase prepares the dataset for further analysis and model development in the subsequent phases.

### 3.2.3 Phase 3: Machine Learning Model Development

The third phase revolves around the development of machine learning models for microclimate monitoring and prediction. The pre-processed data is divided into training and testing sets, with the training set used to train and optimize the Random Forest and XGBoost algorithms. Various parameters and hyperparameters are fine-tuned using techniques like grid search and cross-validation to achieve optimal model performance. The trained models are then evaluated using appropriate assessment metrics, such as mean absolute error and mean squared error, to assess their predictive capabilities. This evaluation process helps determine the effectiveness and performance of the Random Forest and XGBoost algorithms in accurately predicting microclimate patterns for the designated heritage site. The models' performance and generalizability are crucial factors in ensuring the reliability and usefulness of the developed models for microclimate monitoring and prediction.

### 3.2.4 Phase 4: Dashboard Development and Analysis of Result

In the fourth phase, a user-friendly dashboard is designed and developed to visualize and present the microclimate data in a comprehensible manner. The dashboard provides real-time insights into the microclimate conditions of the heritage site, displaying key metrics such as rainfall, temperature, humidity, and wind speed. The trained machine learning models are integrated into the dashboard to provide recommendations for preventive maintenance actions based on the analysed data. Additionally, HTML dashboard is incorporated to facilitate a better understanding of trends and patterns in the microclimate data. The dashboard serves as a valuable tool for local authorities and stakeholders involved in the preservation of cultural heritage sites, enabling them to make informed decisions and take proactive measures to mitigate potential issues that may impact the site's condition.

## 3.3 Justification of Tools, Techniques and Data

The chosen tools for this research include data collection from the Copernicus Climate Data Store (CDS), data pre-processing techniques, machine learning algorithms (Random Forest and XGBoost), and dashboard development. These tools have been selected based on their suitability for addressing the research objectives and providing valuable insights for preventive maintenance strategies at the designated heritage site in Johor Bahru.

Microclimate data from the Copernicus Climate Data Store (CDS) is essential for understanding the environmental conditions at the heritage site. The data includes monthly measurements of temperature, humidity, rainfall, and wind speed from 1940 to 2023. These parameters are crucial for assessing the impact of environmental factors on the site's structural integrity and identifying potential maintenance issues. The data for 2024 is available but only up to June; hence, we decided to predict the 2023 data to allow for comparison with the actual data for that year.

The research utilizes machine learning techniques to analyze the collected microclimate data and develop predictive models for preventive maintenance. Random Forest and XGBoost algorithms were chosen due to their proven effectiveness in handling complex relationships between variables and managing both regression and classification tasks. However, regressors were specifically chosen over classifiers in this study to predict continuous microclimate variables, providing more precise and actionable predictions for future microclimate conditions. Python was selected for implementing these machine learning algorithms due to its robustness and extensive libraries that facilitate data manipulation and analysis. The following Python tools and libraries were employed: Spyder, an Integrated Development Environment (IDE) used for writing and running Python code; Pandas, for data manipulation and analysis, particularly for handling large datasets; NumPy, for numerical computations and efficient array handling; Scikit-learn (sklearn), which provides a wide range of machine learning algorithms, including Random Forest and XGBoost; and Matplotlib and Seaborn, used for data visualization, creating plots and graphs that help in understanding the data and the results of the analysis.

The development of a user-friendly dashboard integrating microclimate data and machine learning models provides a powerful tool for monitoring and maintenance planning. The dashboard is developed using HTML, JavaScript, and CSS, enabling interactive and dynamic visualizations. The graphs, plots, and heatmaps within the HTML dashboard help users understand trends and patterns in the data, facilitating proactive decision-making. This allows local authorities to respond promptly to changing microclimate conditions and potential threats to the heritage site.

## 3.4 Chapter Summary

This chapter summarized the four phases of the research study. The literature review phase involved a comprehensive review of relevant literature, providing a solid knowledge base for the subsequent phases. The data collection and pre-processing phase focused on obtaining and cleaning microclimate data, while the machine learning model development phase involved training and evaluating Random Forest and XGBoost algorithms. The final phase focused on developing a user-friendly dashboard that visualizes the microclimate data and provides maintenance recommendations.

# CHAPTER 4

# EXPERIMENTAL DESIGN AND SETUP

## 4.1 Introduction

This chapter discusses in depth the experimental setup and result of the research methodology described in the previous chapter. The proposed solution will be broken down into several steps, including the data collection, pre-processing, training and testing and machine learning models development.

## 4.2 Proposed Solution

The proposed solution encompasses several steps to address the objectives outlined in the research scope. Initially, microclimate data will be acquired from the Copernicus Climate Data Store (CDS), focusing on parameters like temperature, humidity, precipitation, and wind speed. Subsequently, the collected data will undergo preprocessing to ensure quality and reliability, including handling missing values and outliers. Relevant features will be selected for model training, emphasizing factors such as temperature variations, humidity levels, and wind patterns. Two machine learning algorithms, Random Forest and XGBoost, will be implemented and compared for their performance in microclimate monitoring and prediction. Additionally, a dashboard will be designed and developed using HTML, Javascript and CSS to display the analysis of microclimate data. Finally, the effectiveness of the developed algorithms and dashboard will be evaluated based on their ability to assist local authorities in planning preventive maintenance actions for the heritage site, considering metrics such as prediction accuracy and user feedback. Through these steps, the proposed solution aims to provide a comprehensive framework for enhancing microclimate monitoring and management for the preservation of the Sultan Ibrahim Building, Johor Bahru and A Famosa, Melaka

## 4.3 Flow of Overall Data Processing

In this section, we outline the process of downloading, extracting, and processing microclimate data from the Copernicus database and subsequently generating various analyses and visualizations.

To begin, we downloaded the dataset covering the period from 1940 to 2023 for each microclimate from the CDS. The data was acquired in NetCDF format, which is a standard format for storing multidimensional scientific data. To convert the NetCDF files into a more usable format, we used a Python script, referred to as Script 20. This script extracts the data into DAT files, for instance, Extract-194001.dat for the January 1940 Rain dataset. Each DAT file consists of three columns: latitude, longitude, and the Rain data. The script outputs all the latitude and longitude coordinates along with the corresponding rain data for each observed area.

Next, we used the DAT files generated by Script 20 as input for Script 54. Script 54 processes these files to plot the monthly data, specifically creating Heatmaps of Monthly Accumulated Rainfall for each month of the desired year. Following this, we employed Script 60 to extract data for specific latitude and longitude coordinates, such as those for Sultan Ibrahim Building and A Famosa. Script 60 identifies the nearest available latitude and longitude in the dataset to the chosen coordinates. For example, when selecting the coordinates for Sultan Ibrahim Building, the output file will include columns for the year and month, the chosen latitude and longitude, the nearest latitude and longitude from the data, and the corresponding rain data. Each month's data is saved in separate files, and data for different years is organized into distinct folders.

We then used Script 80 to merge the datasets from various files into a comprehensive dataset, facilitating further analysis. Finally, with the merged dataset from Script 80, we ran Script 100 to generate histograms depicting the monthly average rainfall trends over the selected period. This visualization allows for an easy comparison of rainfall patterns across different months and years.

The Plotting script performs a comprehensive analysis of climate data for Melaka and Johor Bahru, focusing on temperature, wind speed, relative humidity, and rainfall. It begins by loading the respective datasets from CSV files from the merged datasets in script 80 and reshaping them from a wide format to a long format to facilitate easier plotting and analysis. The data is then merged into a single dataset, allowing for a holistic view of the climate variables. Various scatter plots are created to explore the relationships between the different climate variables, and a correlation matrix is generated to visualize the strength of these relationships. The script also produces anomaly plots for each climate variable, highlighting deviations over time. Additionally, it generates trend lines to illustrate the average monthly values for each variable, providing insights into their general behavior throughout the year. Finally, the merged dataset is saved to a CSV file for further analysis.

Next, the prediction script utilizes machine learning models to predict climate data for Melaka in 2024. It starts by loading datasets for rainfall, temperature, humidity, and wind speed. The core functionality is encapsulated in the train_and_predict function, which separates the data into features and target variables, excluding the year 2024. It then trains two models, Random Forest and XGBoost, for each month using data from all other years. Predictions for 2024 are generated for each month by these models. The script prints the predicted values and visualizes them using bar plots, comparing the outputs of the Random Forest and XGBoost models. This process is repeated for each type of climate data, providing monthly predictions for rainfall in millimeters, temperature in degrees Celsius, relative humidity in percentage, and wind speed in meters per second.

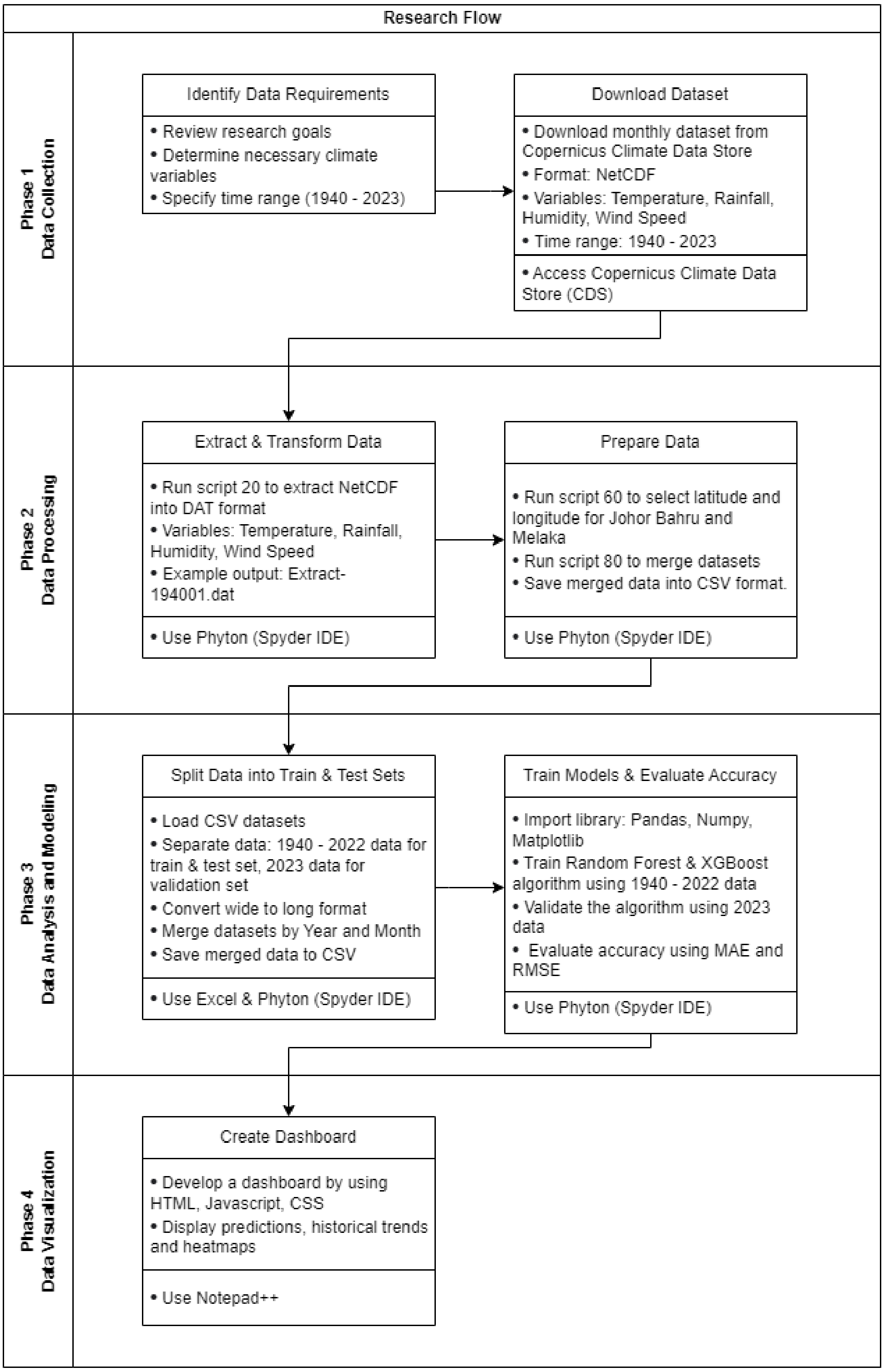

Research Flow
Phase 1
Data Collection
Identify Data Requirements
• Review research goals
• Determine necessary climate variables
• Specify time range (1940 - 2023)
Download Dataset
• Download monthly dataset from Copernicus Climate Data Store
• Format: NetCDF
• Variables: Temperature, Rainfall, Humidity, Wind Speed
• Time range: 1940 - 2023
• Access Copernicus Climate Data Store (CDS)
Phase 2
Data Processing
Extract & Transform Data
• Run script 20 to extract NetCDF into DAT format
• Variables: Temperature, Rainfall, Humidity, Wind Speed
• Example output: Extract-194001.dat
• Use Phyton (Spyder IDE)
Prepare Data
• Run script 60 to select latitude and longitude for Johor Bahru and Melaka
• Run script 80 to merge datasets
• Save merged data into CSV format.
• Use Phyton (Spyder IDE)
Phase 3
Data Analysis and Modeling
Split Data into Train & Test Sets
• Load CSV datasets
• Separate data: 1940 - 2022 data for train & test set, 2023 data for validation set
• Convert wide to long format
• Merge datasets by Year and Month
• Save merged data to CSV
• Use Excel & Phyton (Spyder IDE)
Train Models & Evaluate Accuracy
• Import library: Pandas, Numpy, Matplotlib
• Train Random Forest & XGBoost algorithm using 1940 - 2022 data
• Validate the algorithm using 2023 data
• Evaluate accuracy using MAE and RMSE
• Use Phyton (Spyder IDE)
Phase 4
Data Visualization
Create Dashboard
• Develop a dashboard by using HTML, Javascript, CSS
• Display predictions, historical trends and heatmaps
• Use Notepad++

Figure 4: Research Flow

## 4.4 Study Area

This research focuses on two significant cultural heritage sites in Malaysia. The first is A Famosa in Melaka, a 16th-century Portuguese fortress located in the historic city. It is one of the oldest surviving European architectural remains in Southeast Asia, situated at approximately 2.1936° N, 102.2501° E. The second site is the Sultan Ibrahim Building in Johor Bahru, an iconic administrative building built in the early 20th century. It stands as a prime example of colonial architecture, located at approximately 1.4616° N, 103.7622° E. Both sites are subject to the tropical climate of Malaysia, characterized by high temperatures, humidity, and significant rainfall throughout the year

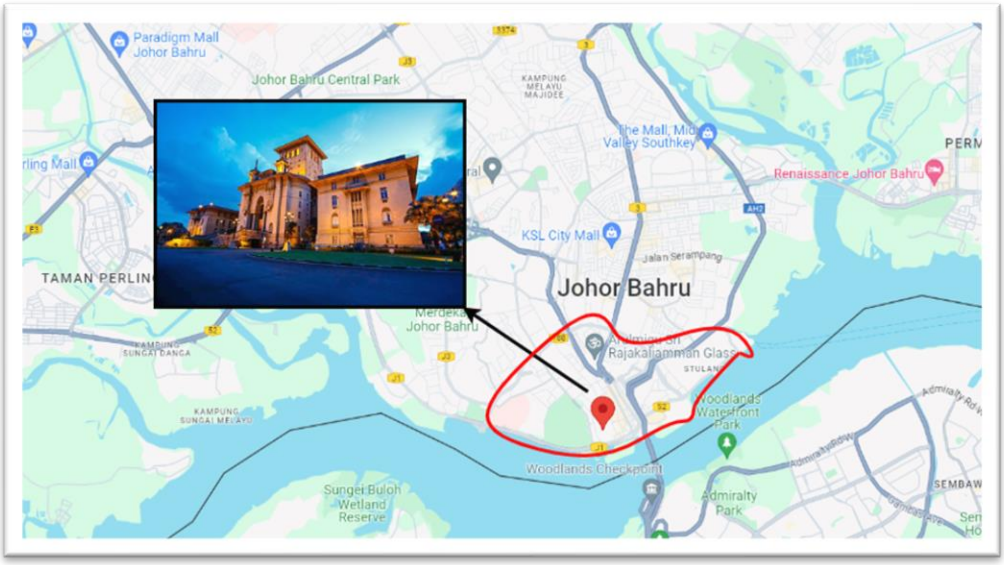


Figure 5: Sultan Ibrahim Building, Johor Bahru

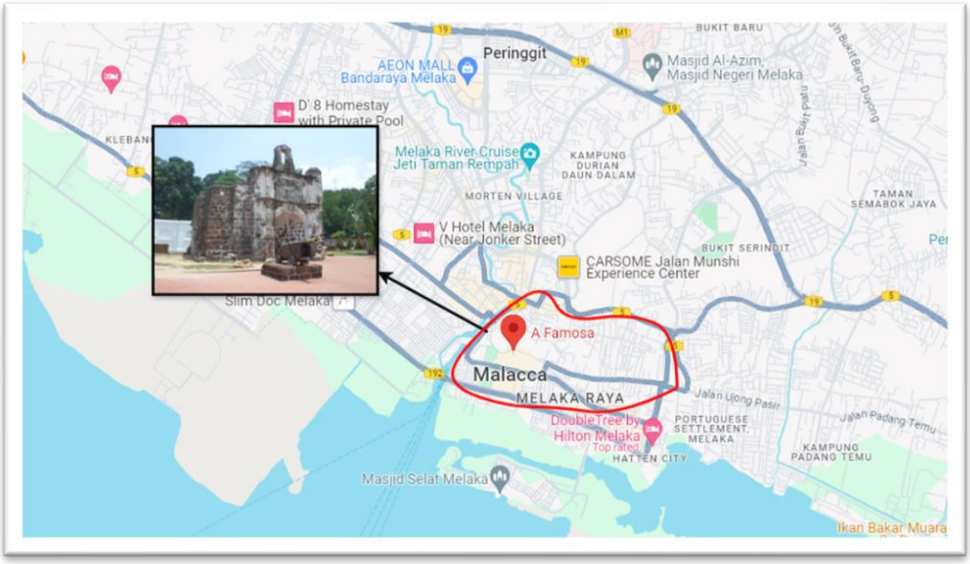

BUKIT BARU
Peringgit
AEON MALL Bandaraya Melaka
Masjid Al-Azim, Masjid Negeri Melaka
D' 8 Homestay with Private Pool
KLEBANG
Melaka River Cruise Jeti Taman Rempah
KAMPUNG DURIAN DAUN DALAM
MORTEN VILLAGE
TAMAN SEMABOK JAYA
V Hotel Melaka (Near Jonker Street)
CARSOME Jalan Munshi Experience Center
BUKIT SERINDIT
Slim Doc Melaka
A Famosa
Malacca
MELAKA RAYA
Jalan Ujong Pasir
Jalan Padang Temu
DoubleTree by Hilton Melaka
PORTUGUESE SETTLEMENT, MELAKA
HATTEN CITY
KAMPUNG PADANG TEMU
Masjid Selat Melaka
Ikan Bakar Muara

Figure 6: A Famosa, Melaka

## 4.5 Data Collection and Extraction

In this research study, the microclimate data was obtained from the ECMWF Reanalysis v5 (ERA5) dataset, provided by the Copernicus Climate Data Store (CDS) at the European Centre for Medium-Range Weather Forecasts (ECMWF). ERA5 is the fifth generation of ECMWF's atmospheric reanalysis, offering a comprehensive global climate analysis spanning from January 1940 to the present day.

The data was available in the NetCDF format, a widely used self-describing, machine-independent data format for array-oriented scientific data. To extract and process the data, we utilized Spyder, a powerful Integrated Development Environment (IDE) for Python, which facilitated efficient data handling and analysis.

For this research, we collected microclimate data spanning a substantial time range, from 1940 to 2023. This extended historical period allowed us to gather a significant amount of data, enabling more reliable and robust predictions related to the microclimate conditions at the designated heritage site.

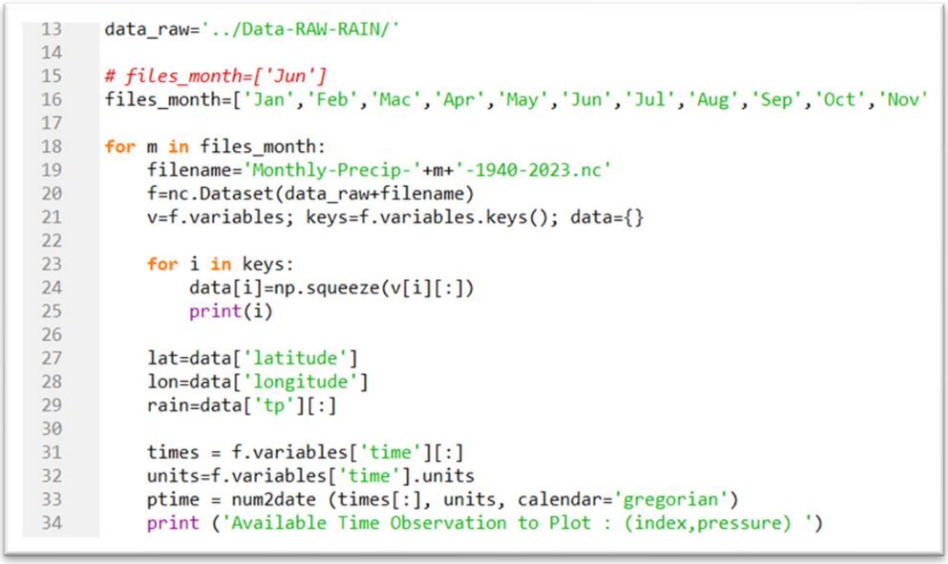
Historical data plays a crucial role in understanding and predicting microclimate patterns in the context of heritage sites. By analyzing long-term data trends, we can gain valuable insights into the site's microclimate dynamics, including temperature, humidity, wind patterns, and precipitation levels. These insights are essential for developing effective strategies to preserve and protect the heritage site from potential environmental impacts.

### 4.5.1 Data Collection

The data was collected from the Climate Data Store (CDS) Copernicus website. The chosen dataset for this research is ERA5 monthly averaged data on single levels from 1940 to present. Single level is chosen because the CDS mentioned that dataset on single level is better for forecasting rather than dataset on pressure level. Hence, for each rainfall, humidity, temperature and wind speed, we downloaded the monthly data from 1940 to 2023 at the specified time (00:00 UTC). and covers a defined geographical area. The data is requested in NetCDF format, which is a common format for storing multidimensional scientific data. The geographical area of interest is specified by the latitude and longitude coordinates [10, 90, -10, 130], representing the region from 10°S to 90°N latitude and from 130°E to 10°W longitude. The retrieved data is then stored in a NetCDF file (e.g., Monthly-Precip-Jan-1940-2024.nc) for further analysis and visualization.

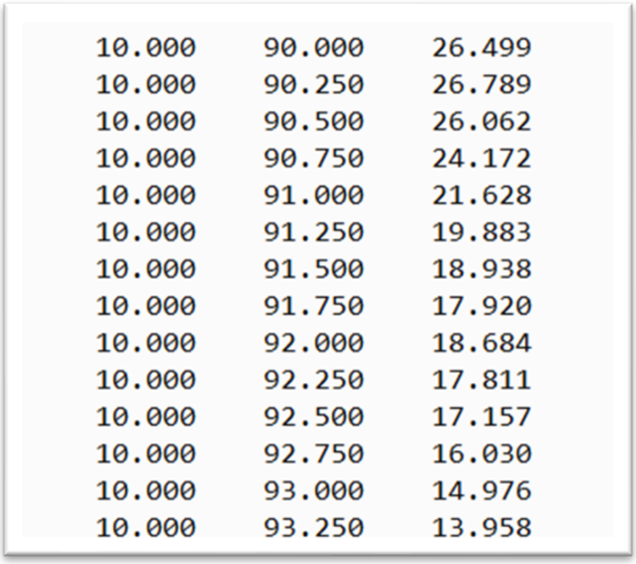
The process followed by processing and extracting the the raw data files as shown in Figure 7. The script loops through a list of month names (e.g., 'Jan', 'Feb', 'Mar', etc.) and opens each corresponding NetCDF file containing temperature data for that month. It reads the latitude, longitude, and temperature variables from the file, as well as the time dimension and its associated units. For each file, the script converts the time values from the NetCDF file to Python datetime objects using the num2date function from the NetCDF4 library. It then prints the available time observations, allowing the user to select a specific time index for further processing.

After selecting a time index, the script extracts the corresponding temperature data and converts it from Kelvin to Celsius. It then creates a grid of latitude and longitude values using np.meshgrid and combines the temperature data with the grid coordinates into a single 2D array. The script then creates a folder structure based on the year and saves the extracted temperature data, along with the corresponding latitude and longitude coordinates, into a text file named Extract-YYYYMM.dat. The file is saved in a subdirectory named after the year, within a directory called '20-Extract' located in the current working directory. Overall, this script is designed to extract and process temperature data from a set of NetCDF files, convert the data to a more accessible format which is DAT files format. This process can be useful for further analysis, visualization, or integration with other data sources.



```python
data_raw='../Data-RAW-RAIN/'

# files_month=['Jun']
files_month=['Jan','Feb','Mac','Apr','May','Jun','Jul','Aug','Sep','Oct','Nov'

for m in files_month:
    filename='Monthly-Precip-'+m+'-1940-2023.nc'
    f=nc.Dataset(data_raw+filename)
    v=f.variables; keys=f.variables.keys(); data={}

    for i in keys:
        data[i]=np.squeeze(v[i][:])
        print(i)

    lat=data['latitude']
    lon=data['longitude']
    rain=data['tp'][:]

    times = f.variables['time'][:]
    units=f.variables['time'].units
    ptime = num2date (times[:], units, calendar='gregorian')
    print ('Available Time Observation to Plot : (index,pressure) ')
```

Figure 7: Data Extraction and Processing

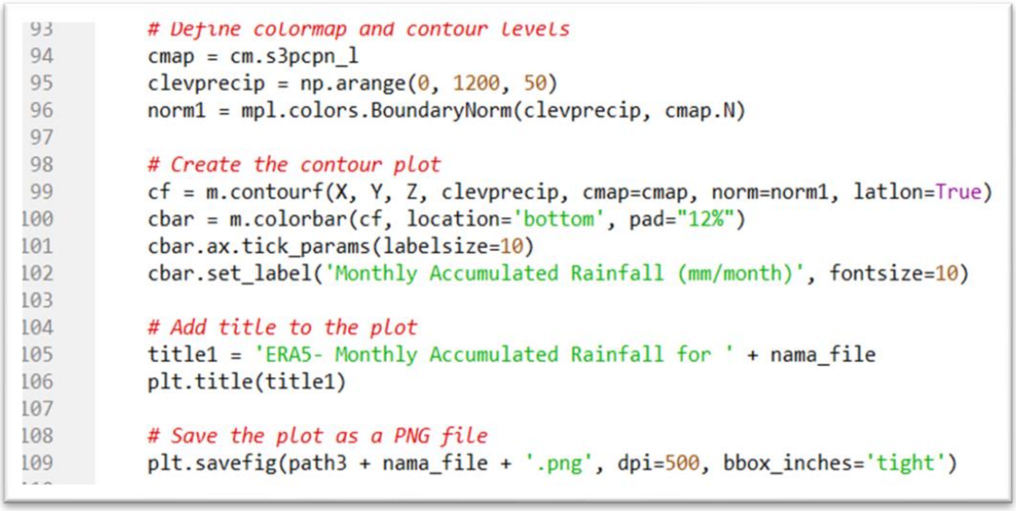


```
10.000     90.000     26.499
10.000     90.250     26.789
10.000     90.500     26.062
10.000     90.750     24.172
10.000     91.000     21.628
10.000     91.250     19.883
10.000     91.500     18.938
10.000     91.750     17.920
10.000     92.000     18.684
10.000     92.250     17.811
10.000     92.500     17.157
10.000     92.750     16.030
10.000     93.000     14.976
10.000     93.250     13.958
```

Figure 8: Processed Dataset in DAT file format

,

The script in Figure 9 is designed to visualize monthly average temperature data for a specific region, with Malaysia as an example. It starts by importing necessary libraries for data processing and plotting. After setting up the directory structure, it reads temperature data files, processes them to extract latitude, longitude, and temperature values, and creates a contour plot of the temperature data on a map. The map is centred on the region of interest which in this case, Malaysia and includes country and state boundaries, coastlines, parallels, and meridians.



```
# Define colormap and contour levels
cmap = cm.s3pcpn_l
clevprecip = np.arange(0, 1200, 50)
norm1 = mpl.colors.BoundaryNorm(clevprecip, cmap.N)

# Create the contour plot
cf = m.contourf(X, Y, Z, clevprecip, cmap=cmap, norm=norm1, latlon=True)
cbar = m.colorbar(cf, location='bottom', pad="12%")
cbar.ax.tick_params(labelsize=10)
cbar.set_label('Monthly Accumulated Rainfall (mm/month)', fontsize=10)

# Add title to the plot
title1 = 'ERA5- Monthly Accumulated Rainfall for ' + nama_file
plt.title(title1)

# Save the plot as a PNG file
plt.savefig(path3 + nama_file + '.png', dpi=500, bbox_inches='tight')
```

Figure 9: Plotting Monthly Average Temperature Data

Additionally, it overlays administrative boundaries of Malaysian states obtained from a shapefile. The script utilizes a colormap to represent temperature variations and adds a colorbar for reference. Finally, it saves the generated plot images, each corresponding to a specific data file, for further analysis or visualization. Overall, this script facilitates the visualization of monthly temperature patterns for a chosen region, aiding in understanding climate variations over time.

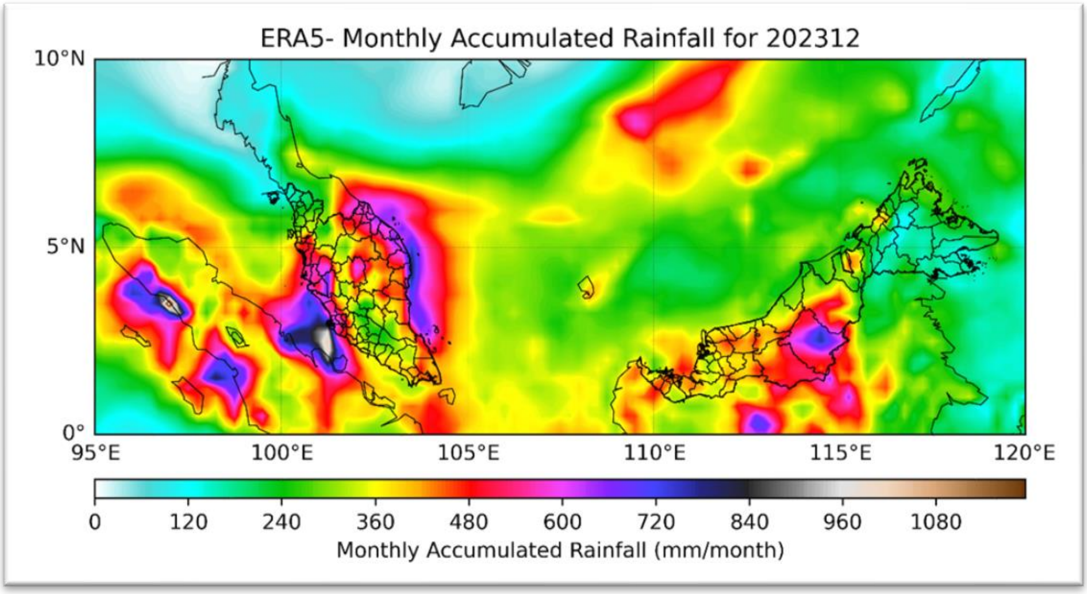
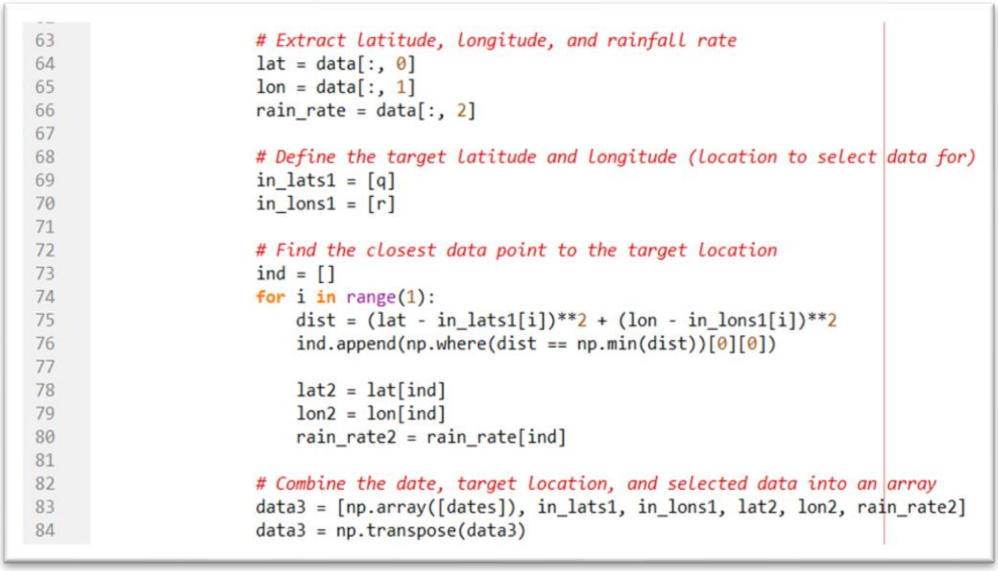

ERA5- Monthly Accumulated Rainfall for 202312
10°N
5°N
0°
95°E
100°E
105°E
110°E
115°E
120°E
0
120
240
360
480
600
720
840
960
1080
Monthly Accumulated Rainfall (mm/month)

Figure 10: Monthly Accumulated Rainfall Sample Output

The script described in Figure 11 is designed to extract location-specific meteorological data from a collection of data files. It allows users to specify one or more locations of interest, along with optional criteria such as specific years or a range of years. After importing necessary libraries, the script prompts users to input the location(s) they are interested in, along with any desired years. It then creates directories for storing both the original data files and the extracted location-specific data.

For each combination of year and location, the script searches for matching data files and reads the data, typically containing latitude, longitude, and meteorological parameters like temperature or rainfall rates. It calculates the distance between each data point and the specified location to find the closest data point, from which it extracts relevant values. The extracted location-specific data is then saved to new files, named to indicate the location, year, and time period. This process streamlines the extraction of meteorological data tailored to specific locations, eliminating the need for manual data searching and filtering. This functionality is valuable for researchers, meteorologists, or anyone interested in analysing weather patterns or environmental conditions at specific locations.

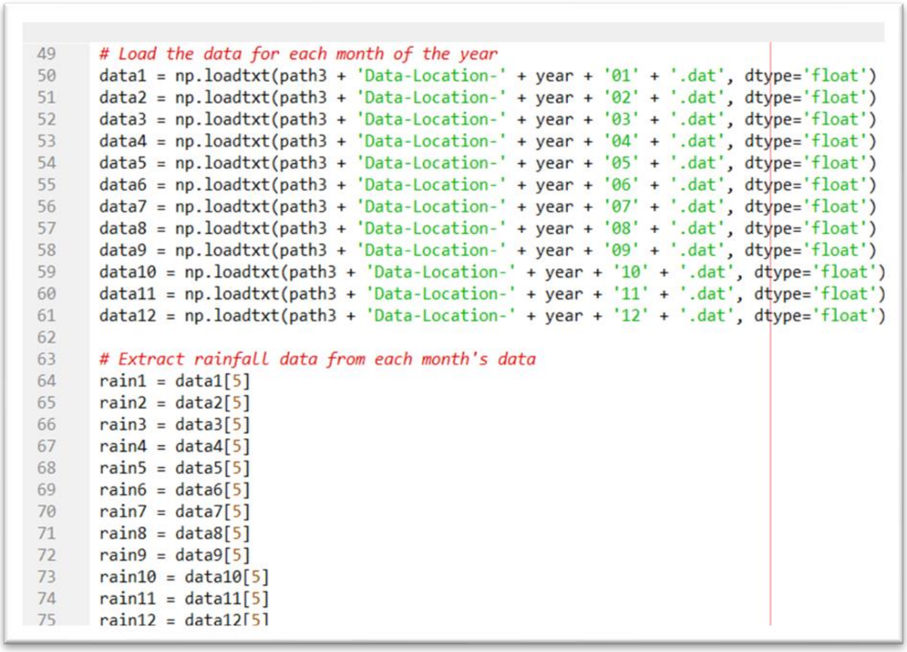


```python
# Extract latitude, longitude, and rainfall rate
lat = data[:, 0]
lon = data[:, 1]
rain_rate = data[:, 2]

# Define the target latitude and longitude (location to select data for)
in_lats1 = [q]
in_lons1 = [r]

# Find the closest data point to the target location
ind = []
for i in range(1):
    dist = (lat - in_lats1[i])**2 + (lon - in_lons1[i])**2
    ind.append(np.where(dist == np.min(dist))[0][0])

    lat2 = lat[ind]
    lon2 = lon[ind]
    rain_rate2 = rain_rate[ind]

# Combine the date, target location, and selected data into an array
data3 = [np.array([dates]), in_lats1, in_lons1, lat2, lon2, rain_rate2]
data3 = np.transpose(data3)
```

Figure 11: Extracting Location-Specific Data from Data Files

The script described in Figure 7 is designed to merge and consolidate location-specific microclimate data from multiple files into a single file for each location of interest. This consolidation process helps in organizing and simplifying the data for easier analysis or visualization. After importing necessary libraries and defining the locations and years of interest, the script creates directories for storing both the extracted location-specific data files and the merged data files.

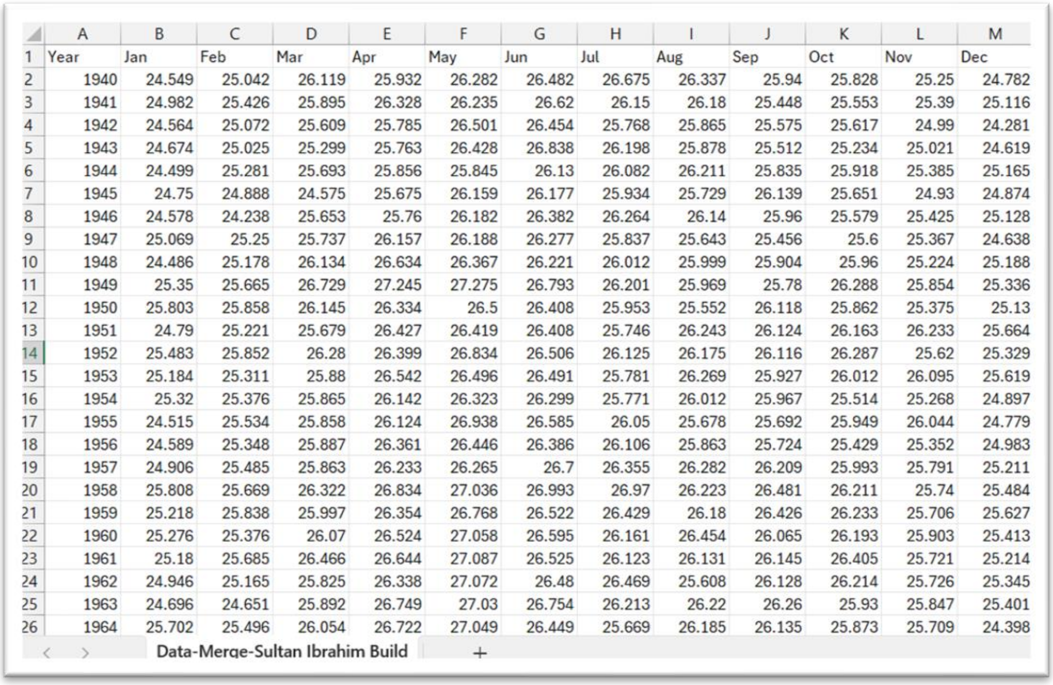
For each combination of year and location, the script reads the monthly data files containing information such as date, latitude, longitude, and meteorological parameters like temperature or rainfall rates. It then extracts the relevant temperature or rainfall rate values for each month and organizes them into a list, along with the corresponding year. These lists are then appended to a larger list, accumulating the data for all years and locations. Once all specified years and locations have been processed, the accumulated data list is saved to a new file in the '80-Merge-Data' directory. This file contains the merged and consolidated data for the specific location, with each row representing a year and the corresponding temperature or rainfall rate values for each month.



```
# Load the data for each month of the year
data1 = np.loadtxt(path3 + 'Data-Location-' + year + '01' + '.dat', dtype='float')
data2 = np.loadtxt(path3 + 'Data-Location-' + year + '02' + '.dat', dtype='float')
data3 = np.loadtxt(path3 + 'Data-Location-' + year + '03' + '.dat', dtype='float')
data4 = np.loadtxt(path3 + 'Data-Location-' + year + '04' + '.dat', dtype='float')
data5 = np.loadtxt(path3 + 'Data-Location-' + year + '05' + '.dat', dtype='float')
data6 = np.loadtxt(path3 + 'Data-Location-' + year + '06' + '.dat', dtype='float')
data7 = np.loadtxt(path3 + 'Data-Location-' + year + '07' + '.dat', dtype='float')
data8 = np.loadtxt(path3 + 'Data-Location-' + year + '08' + '.dat', dtype='float')
data9 = np.loadtxt(path3 + 'Data-Location-' + year + '09' + '.dat', dtype='float')
data10 = np.loadtxt(path3 + 'Data-Location-' + year + '10' + '.dat', dtype='float')
data11 = np.loadtxt(path3 + 'Data-Location-' + year + '11' + '.dat', dtype='float')
data12 = np.loadtxt(path3 + 'Data-Location-' + year + '12' + '.dat', dtype='float')

# Extract rainfall data from each month's data
rain1 = data1[5]
rain2 = data2[5]
rain3 = data3[5]
rain4 = data4[5]
rain5 = data5[5]
rain6 = data6[5]
rain7 = data7[5]
rain8 = data8[5]
rain9 = data9[5]
rain10 = data10[5]
rain11 = data11[5]
rain12 = data12[5]
```

Figure 12: Merging and Consolidating Location-Specific Microclimate Data

By running this script, users can efficiently merge and consolidate location-specific meteorological data from multiple files into a single file for each location of interest. This consolidated data can then be further analysed to identify trends, patterns, or anomalies in the meteorological data over time, or used for creating visualizations or reports.

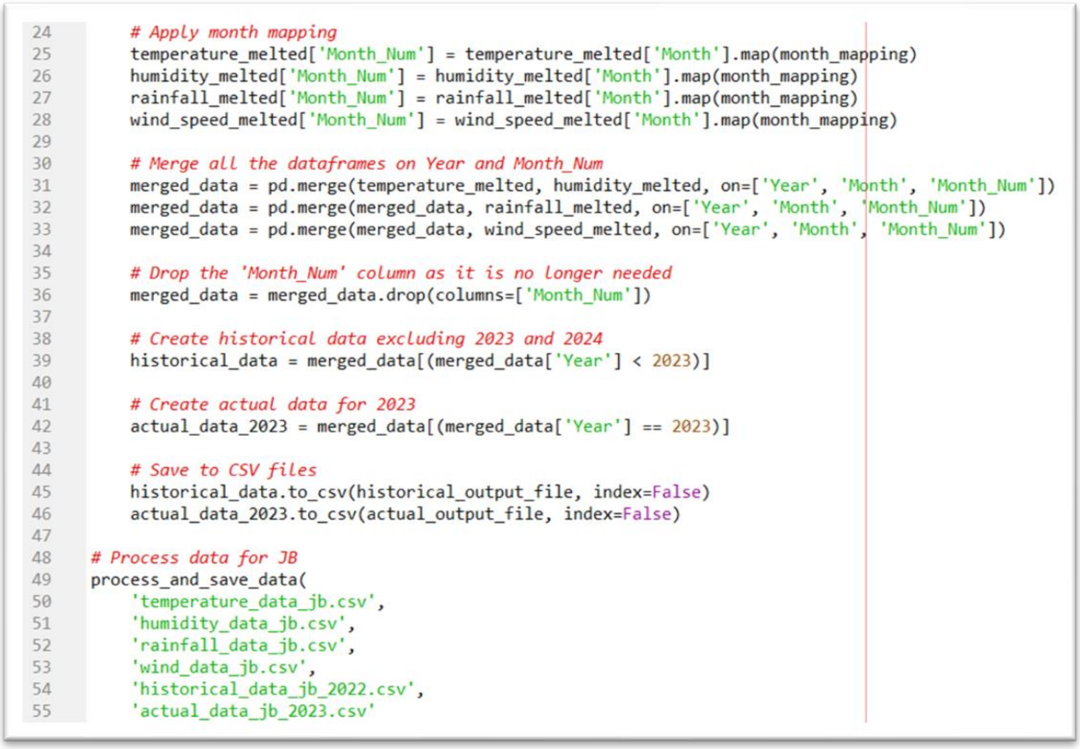


| | A | B | C | D | E | F | G | H | I | J | K | L | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Year | Jan | Feb | Mar | Apr | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec |
| 2 | 1940 | 24.549 | 25.042 | 26.119 | 25.932 | 26.282 | 26.482 | 26.675 | 26.337 | 25.94 | 25.828 | 25.25 | 24.782 |
| 3 | 1941 | 24.982 | 25.426 | 25.895 | 26.328 | 26.235 | 26.62 | 26.15 | 26.18 | 25.448 | 25.553 | 25.39 | 25.116 |
| 4 | 1942 | 24.564 | 25.072 | 25.609 | 25.785 | 26.501 | 26.454 | 25.768 | 25.865 | 25.575 | 25.617 | 24.99 | 24.281 |
| 5 | 1943 | 24.674 | 25.025 | 25.299 | 25.763 | 26.428 | 26.838 | 26.198 | 25.878 | 25.512 | 25.234 | 25.021 | 24.619 |
| 6 | 1944 | 24.499 | 25.281 | 25.693 | 25.856 | 25.845 | 26.13 | 26.082 | 26.211 | 25.835 | 25.918 | 25.385 | 25.165 |
| 7 | 1945 | 24.75 | 24.888 | 24.575 | 25.675 | 26.159 | 26.177 | 25.934 | 25.729 | 26.139 | 25.651 | 24.93 | 24.874 |
| 8 | 1946 | 24.578 | 24.238 | 25.653 | 25.76 | 26.182 | 26.382 | 26.264 | 26.14 | 25.96 | 25.579 | 25.425 | 25.128 |
| 9 | 1947 | 25.069 | 25.25 | 25.737 | 26.157 | 26.188 | 26.277 | 25.837 | 25.643 | 25.456 | 25.6 | 25.367 | 24.638 |
| 10 | 1948 | 24.486 | 25.178 | 26.134 | 26.634 | 26.367 | 26.221 | 26.012 | 25.999 | 25.904 | 25.96 | 25.224 | 25.188 |
| 11 | 1949 | 25.35 | 25.665 | 26.729 | 27.245 | 27.275 | 26.793 | 26.201 | 25.969 | 25.78 | 26.288 | 25.854 | 25.336 |
| 12 | 1950 | 25.803 | 25.858 | 26.145 | 26.334 | 26.5 | 26.408 | 25.953 | 25.552 | 26.118 | 25.862 | 25.375 | 25.13 |
| 13 | 1951 | 24.79 | 25.221 | 25.679 | 26.427 | 26.419 | 26.408 | 25.746 | 26.243 | 26.124 | 26.163 | 26.233 | 25.664 |
| 14 | 1952 | 25.483 | 25.852 | 26.28 | 26.399 | 26.834 | 26.506 | 26.125 | 26.175 | 26.116 | 26.287 | 25.62 | 25.329 |
| 15 | 1953 | 25.184 | 25.311 | 25.88 | 26.542 | 26.496 | 26.491 | 25.781 | 26.269 | 25.927 | 26.012 | 26.095 | 25.619 |
| 16 | 1954 | 25.32 | 25.376 | 25.865 | 26.142 | 26.323 | 26.299 | 25.771 | 26.012 | 25.967 | 25.514 | 25.268 | 24.897 |
| 17 | 1955 | 24.515 | 25.534 | 25.858 | 26.124 | 26.938 | 26.585 | 26.05 | 25.678 | 25.692 | 25.949 | 26.044 | 24.779 |
| 18 | 1956 | 24.589 | 25.348 | 25.887 | 26.361 | 26.446 | 26.386 | 26.106 | 25.863 | 25.724 | 25.429 | 25.352 | 24.983 |
| 19 | 1957 | 24.906 | 25.485 | 25.863 | 26.233 | 26.265 | 26.7 | 26.355 | 26.282 | 26.209 | 25.993 | 25.791 | 25.211 |
| 20 | 1958 | 25.808 | 25.669 | 26.322 | 26.834 | 27.036 | 26.993 | 26.97 | 26.223 | 26.481 | 26.211 | 25.74 | 25.484 |
| 21 | 1959 | 25.218 | 25.838 | 25.997 | 26.354 | 26.768 | 26.522 | 26.429 | 26.18 | 26.426 | 26.233 | 25.706 | 25.627 |
| 22 | 1960 | 25.276 | 25.376 | 26.07 | 26.524 | 27.058 | 26.595 | 26.161 | 26.454 | 26.065 | 26.193 | 25.903 | 25.413 |
| 23 | 1961 | 25.18 | 25.685 | 26.466 | 26.644 | 27.087 | 26.525 | 26.123 | 26.131 | 26.145 | 26.405 | 25.721 | 25.214 |
| 24 | 1962 | 24.946 | 25.165 | 25.825 | 26.338 | 27.072 | 26.48 | 26.469 | 25.608 | 26.128 | 26.214 | 25.726 | 25.345 |
| 25 | 1963 | 24.696 | 24.651 | 25.892 | 26.749 | 27.03 | 26.754 | 26.213 | 26.22 | 26.26 | 25.93 | 25.847 | 25.401 |
| 26 | 1964 | 25.702 | 25.496 | 26.054 | 26.722 | 27.049 | 26.449 | 25.669 | 26.185 | 26.135 | 25.873 | 25.709 | 24.398 |

Figure 13: Merged Temperature Data for Sultan Ibrahim Building Area

### 4.5.2 Splitting of Data into Training and Testing Sets

In this research study, the provided code for data processing and saving facilitates the preparation of datasets for machine learning model training and testing. The function process_and_save_data is responsible for loading and preprocessing multiple types of microclimate data (temperature, humidity, rainfall, and wind speed) and merging them into a single dataset. Initially, the function reads in the CSV files containing temperature, humidity, rainfall, and wind speed data for a given location. Each dataset is then melted from a wide format to a long format, where each row represents a single observation (Year, Month, Value). This step ensures that the data is in a consistent format for merging.



```
    # Apply month mapping
    temperature_melted['Month_Num'] = temperature_melted['Month'].map(month_mapping)
    humidity_melted['Month_Num'] = humidity_melted['Month'].map(month_mapping)
    rainfall_melted['Month_Num'] = rainfall_melted['Month'].map(month_mapping)
    wind_speed_melted['Month_Num'] = wind_speed_melted['Month'].map(month_mapping)

    # Merge all the dataframes on Year and Month_Num
    merged_data = pd.merge(temperature_melted, humidity_melted, on=['Year', 'Month', 'Month_Num'])
    merged_data = pd.merge(merged_data, rainfall_melted, on=['Year', 'Month', 'Month_Num'])
    merged_data = pd.merge(merged_data, wind_speed_melted, on=['Year', 'Month', 'Month_Num'])

    # Drop the 'Month_Num' column as it is no longer needed
    merged_data = merged_data.drop(columns=['Month_Num'])

    # Create historical data excluding 2023 and 2024
    historical_data = merged_data[(merged_data['Year'] < 2023)]

    # Create actual data for 2023
    actual_data_2023 = merged_data[(merged_data['Year'] == 2023)]

    # Save to CSV files
    historical_data.to_csv(historical_output_file, index=False)
    actual_data_2023.to_csv(actual_output_file, index=False)

# Process data for JB
process_and_save_data(
    'temperature_data_jb.csv',
    'humidity_data_jb.csv',
    'rainfall_data_jb.csv',
    'wind_data_jb.csv',
    'historical_data_jb_2022.csv',
    'actual_data_jb_2023.csv'
```

Figure 14: Split the pre-processed data into training and testing sets

A dictionary mapping month names to numeric values is applied to convert month names into numerical representations, which aids in the merging process. The reshaped datasets are merged into a single dataframe on the columns 'Year' and 'Month_Num', consolidating all microclimate variables into one dataset. The merged dataset is then filtered to exclude data from 2023 and beyond, creating the historical data subset which is saved as a CSV file for training purposes. Additionally, data specific to the year 2023 is extracted and saved separately, representing the actual data for testing and validation.

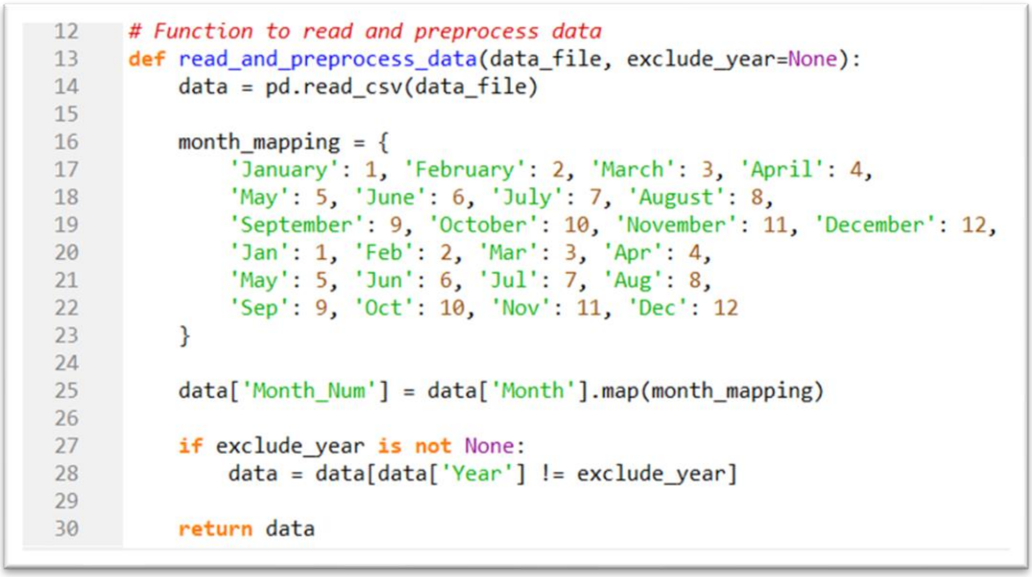
In this context, the split into training and testing sets is achieved by separating historical data (up to 2022) and recent data (2023). This approach ensures that the model is trained on past data and tested on the most recent data, simulating a real-world scenario where future predictions are made based on historical patterns. Splitting data into training and testing sets is essential for model validation, preventing overfitting, and ensuring reliable performance metrics. It allows for the evaluation of the model's performance on unseen data, providing an estimate of how well the model generalizes to new data. By training the model on one subset and testing on another, we can detect overfitting, where the model performs well on training data but poorly on test data. This systematic approach to data preprocessing, merging, and splitting lays the groundwork for robust machine learning model development and evaluation in the context of microclimate data analysis.

## 4.6 Data Analysis and Modelling

This section outlines data analysis and modelling steps in this research. The process involved data preprocessing, feature engineering, model training, evaluation, and visualization. These steps ensured the development of accurate and robust machine learning models for climate variable prediction.

### 4.6.1 Data Preprocessing

The first step in the experimental setup was to preprocess the data. The read_and_preprocess_data function was utilized to load and preprocess both historical and actual data. This function mapped month names to numerical values, excluded specific years if necessary, and handled missing values. The code snippet in Figure 15 below illustrates this process.

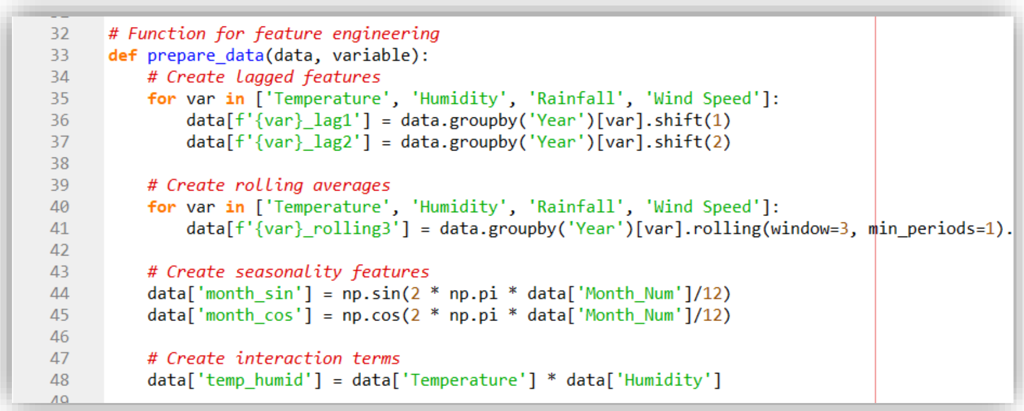


```python
# Function to read and preprocess data
def read_and_preprocess_data(data_file, exclude_year=None):
    data = pd.read_csv(data_file)

    month_mapping = {
        'January': 1, 'February': 2, 'March': 3, 'April': 4,
        'May': 5, 'June': 6, 'July': 7, 'August': 8,
        'September': 9, 'October': 10, 'November': 11, 'December': 12,
        'Jan': 1, 'Feb': 2, 'Mar': 3, 'Apr': 4,
        'May': 5, 'Jun': 6, 'Jul': 7, 'Aug': 8,
        'Sep': 9, 'Oct': 10, 'Nov': 11, 'Dec': 12
    }

    data['Month_Num'] = data['Month'].map(month_mapping)

    if exclude_year is not None:
        data = data[data['Year'] != exclude_year]

    return data
```

Figure 15: Read and Preprocess Data

### 4.6.2 Feature Engineering

Feature engineering involved creating lagged features, rolling averages, seasonality features, and interaction terms. This was implemented in the prepare_data function. Lagged features captured temporal dependencies, rolling averages smoothed out short-term fluctuations, seasonality features accounted for cyclical patterns, and interaction terms provided additional predictive power. The implementation is as in the Figure 16 below.

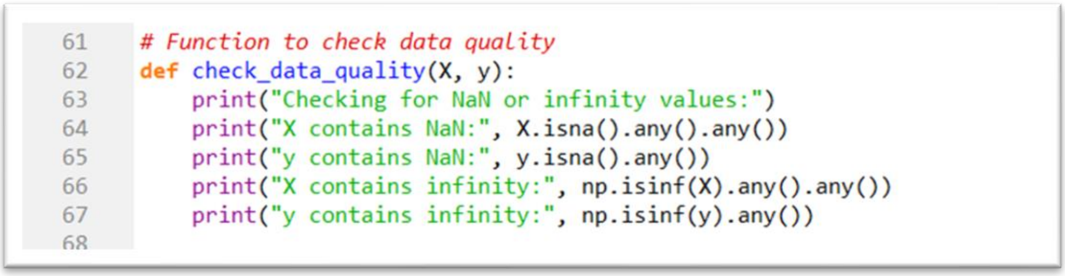


```python
# Function for feature engineering
def prepare_data(data, variable):
    # Create lagged features
    for var in ['Temperature', 'Humidity', 'Rainfall', 'Wind Speed']:
        data[f'{var}_lag1'] = data.groupby('Year')[var].shift(1)
        data[f'{var}_lag2'] = data.groupby('Year')[var].shift(2)

    # Create rolling averages
    for var in ['Temperature', 'Humidity', 'Rainfall', 'Wind Speed']:
        data[f'{var}_rolling3'] = data.groupby('Year')[var].rolling(window=3, min_periods=1).

    # Create seasonality features
    data['month_sin'] = np.sin(2 * np.pi * data['Month_Num']/12)
    data['month_cos'] = np.cos(2 * np.pi * data['Month_Num']/12)

    # Create interaction terms
    data['temp_humid'] = data['Temperature'] * data['Humidity']
```

Figure 16: Feature Engineering

### 4.6.3 Data Quality Check

Ensuring data quality was paramount before training the models. The check_data_quality function checked for the presence of NaN or infinity values in the features and target variable. This process is essential to ensure the integrity of the dataset.

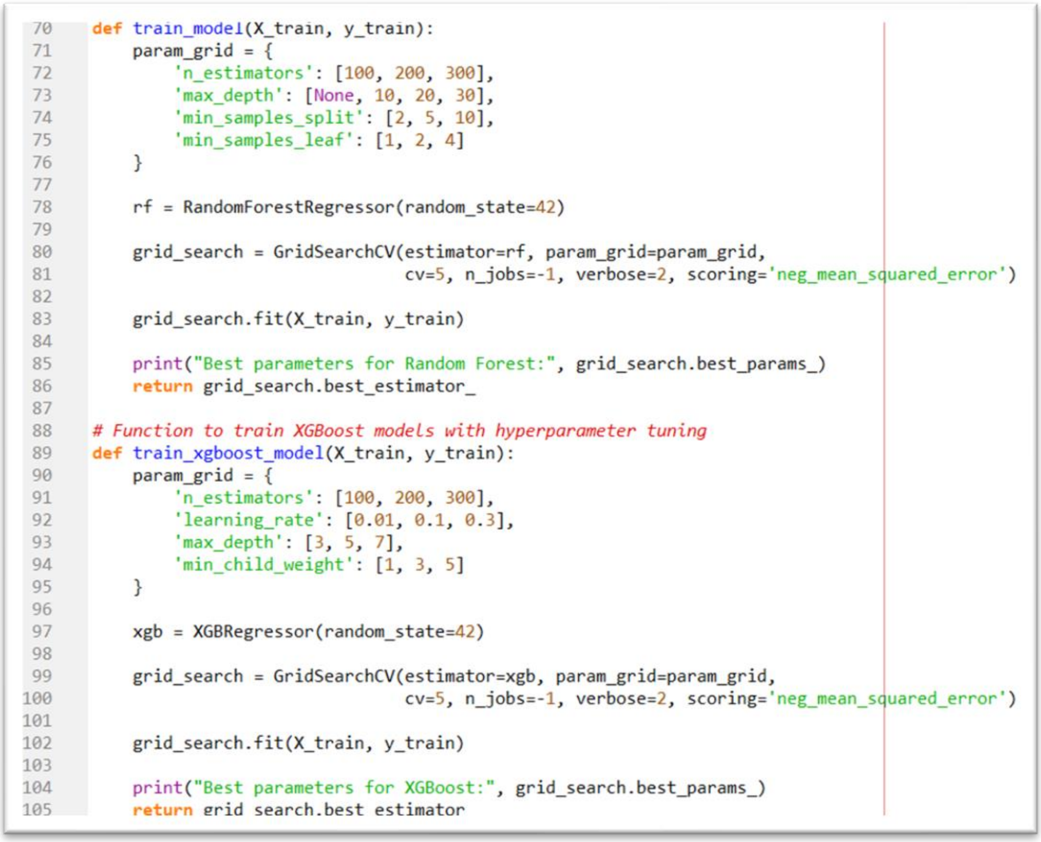


```python
# Function to check data quality
def check_data_quality(X, y):
    print("Checking for NaN or infinity values:")
    print("X contains NaN:", X.isna().any().any())
    print("y contains NaN:", y.isna().any())
    print("X contains infinity:", np.isinf(X).any().any())
    print("y contains infinity:", np.isinf(y).any())
```

Figure 17: Data Quality Check

### 4.6.4 Training of Machine Learning Models

For training the models, the data were first prepared by creating lagged features for each climate variable and splitting the data into training and testing sets. The training set included historical data, while the testing set consisted of data from 2023. The Random Forest model was trained with hyperparameter tuning using GridSearchCV, tuning parameters such as the number of trees (n_estimators), maximum depth of the trees (max_depth), minimum samples required to split a node (min_samples_split), and minimum samples required at each leaf node (min_samples_leaf). Similarly, the XGBoost model was trained with hyperparameter tuning, where the parameters tuned were the number of trees (n_estimators), learning rate, maximum tree depth (max_depth), and minimum sum of instance weight needed in a child (min_child_weight). The hyperparameter tuning process involved a grid search with cross-validation to ensure the best combination of parameters for optimal performance.



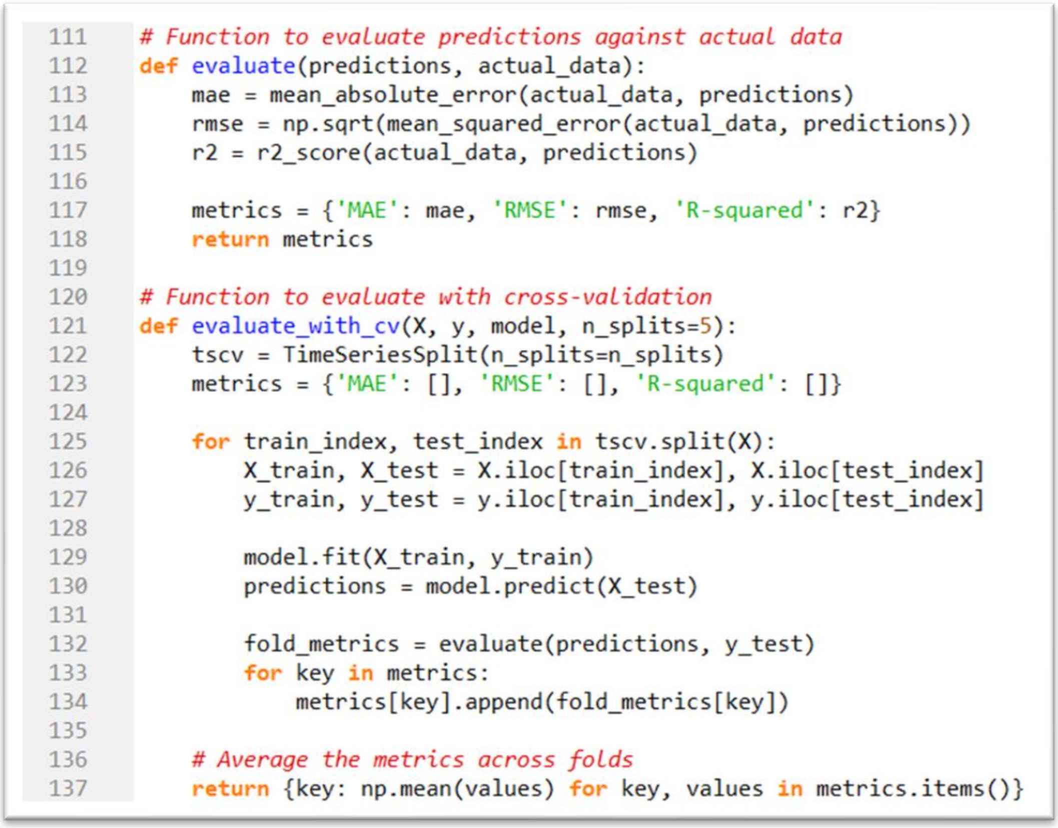
```
def train_model(X_train, y_train):
    param_grid = {
        'n_estimators': [100, 200, 300],
        'max_depth': [None, 10, 20, 30],
        'min_samples_split': [2, 5, 10],
        'min_samples_leaf': [1, 2, 4]
    }

    rf = RandomForestRegressor(random_state=42)

    grid_search = GridSearchCV(estimator=rf, param_grid=param_grid,
                               cv=5, n_jobs=-1, verbose=2, scoring='neg_mean_squared_error')

    grid_search.fit(X_train, y_train)

    print("Best parameters for Random Forest:", grid_search.best_params_)
    return grid_search.best_estimator_

# Function to train XGBoost models with hyperparameter tuning
def train_xgboost_model(X_train, y_train):
    param_grid = {
        'n_estimators': [100, 200, 300],
        'learning_rate': [0.01, 0.1, 0.3],
        'max_depth': [3, 5, 7],
        'min_child_weight': [1, 3, 5]
    }

    xgb = XGBRegressor(random_state=42)

    grid_search = GridSearchCV(estimator=xgb, param_grid=param_grid,
                               cv=5, n_jobs=-1, verbose=2, scoring='neg_mean_squared_error')

    grid_search.fit(X_train, y_train)

    print("Best parameters for XGBoost:", grid_search.best_params_)
    return grid_search.best_estimator_
```

Figure 18: Train Random Forest and XGBoost

### 4.6.5 Model Evaluation

Evaluating the accuracy of the trained models is crucial to understanding their predictive performance. Multiple metrics and cross-validation techniques were used for this purpose. The metrics included Mean Absolute Error (MAE), which measures the average magnitude of errors in predictions without considering their direction; Root Mean Squared Error (RMSE), which measures the square root of the average of squared differences between predicted and actual values, giving higher weight to larger errors; and R-squared ($R^2$), which indicates the proportion of the variance in the dependent variable predictable from the independent variables.

TimeSeriesSplit was employed for cross-validation to maintain the temporal order of the data. This method ensures that the model is evaluated on future data points that were not seen during training, providing a realistic estimate of its performance in real-world scenarios. The cross-validation process involved splitting the data into multiple training and testing sets and computing the evaluation metrics for each fold. The average of these metrics was then used to assess the overall performance of the model.

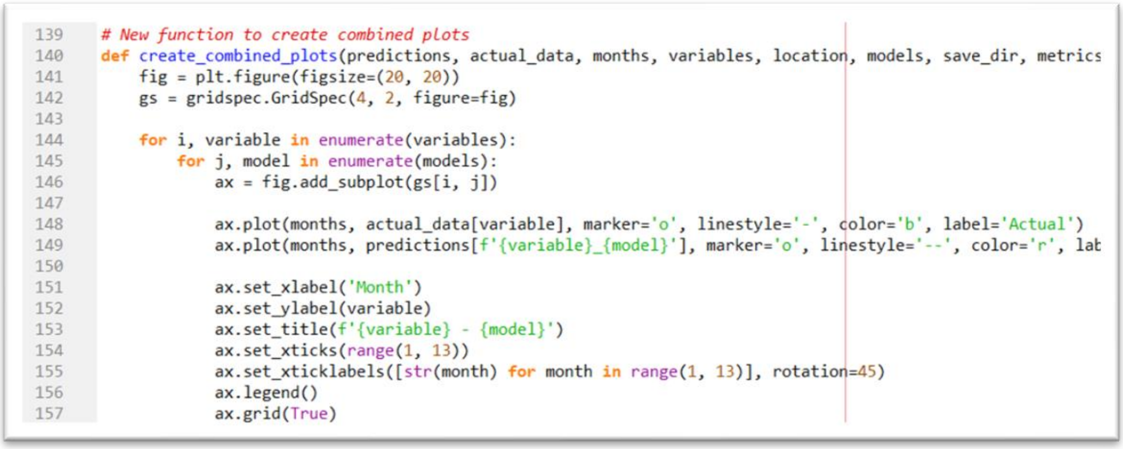


```
# Function to evaluate predictions against actual data
def evaluate(predictions, actual_data):
    mae = mean_absolute_error(actual_data, predictions)
    rmse = np.sqrt(mean_squared_error(actual_data, predictions))
    r2 = r2_score(actual_data, predictions)

    metrics = {'MAE': mae, 'RMSE': rmse, 'R-squared': r2}
    return metrics

# Function to evaluate with cross-validation
def evaluate_with_cv(X, y, model, n_splits=5):
    tscv = TimeSeriesSplit(n_splits=n_splits)
    metrics = {'MAE': [], 'RMSE': [], 'R-squared': []}

    for train_index, test_index in tscv.split(X):
        X_train, X_test = X.iloc[train_index], X.iloc[test_index]
        y_train, y_test = y.iloc[train_index], y.iloc[test_index]

        model.fit(X_train, y_train)
        predictions = model.predict(X_test)

        fold_metrics = evaluate(predictions, y_test)
        for key in metrics:
            metrics[key].append(fold_metrics[key])

    # Average the metrics across folds
    return {key: np.mean(values) for key, values in metrics.items()}
```

Figure 19: Evaluate the Accuracy of Random Forest and XGBoost

### 4.6.6 Prediction and Visualization

After training and evaluation, the models were used to make predictions. The predict function generated predictions, and the create_combined_plots function visualized the results, comparing actual and predicted values.

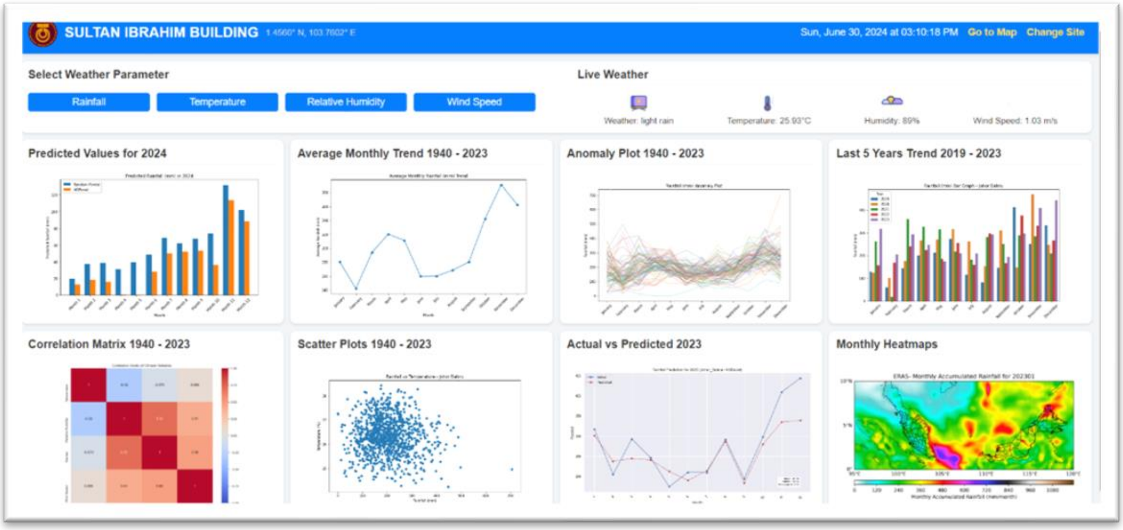


```python
# New function to create combined plots
def create_combined_plots(predictions, actual_data, months, variables, location, models, save_dir, metrics
    fig = plt.figure(figsize=(20, 20))
    gs = gridspec.GridSpec(4, 2, figure=fig)

    for i, variable in enumerate(variables):
        for j, model in enumerate(models):
            ax = fig.add_subplot(gs[i, j])

            ax.plot(months, actual_data[variable], marker='o', linestyle='-', color='b', label='Actual')
            ax.plot(months, predictions[f'{variable}_{model}'], marker='o', linestyle='--', color='r', lab

            ax.set_xlabel('Month')
            ax.set_ylabel(variable)
            ax.set_title(f'{variable} - {model}')
            ax.set_xticks(range(1, 13))
            ax.set_xticklabels([str(month) for month in range(1, 13)], rotation=45)
            ax.legend()
            ax.grid(True)
```

Figure 20: Create plots and accuracy results

## 4.7 Dashboard Development

The dashboard was developed using HTML, JavaScript, and CSS to provide an intuitive visualization of the microclimate data and predictions for the cultural heritage sites. It features multiple interactive components, including selectable weather parameters, live weather display, and various charts showing predicted values, historical trends, and correlations between climate variables. The dashboard presents data through bar charts, line graphs, scatter plots, and heatmaps, offering a comprehensive view of both historical patterns and future predictions. Key elements include a actual vs predicted value for 2023, prediction for 2024, average monthly trends from 1940-2023, anomaly plots, and a correlation matrix to illustrate relationships between different weather parameters. The interface allows users to easily switch between different heritage sites and weather variables, making it a versatile tool for researchers and site managers to analyze microclimate conditions and trends relevant to cultural heritage preservation.



SULTAN IBRAHIM BUILDING 1.4560° N, 103.7602° E

Sun, June 30, 2024 at 03:10:18 PM

Weather: light rain

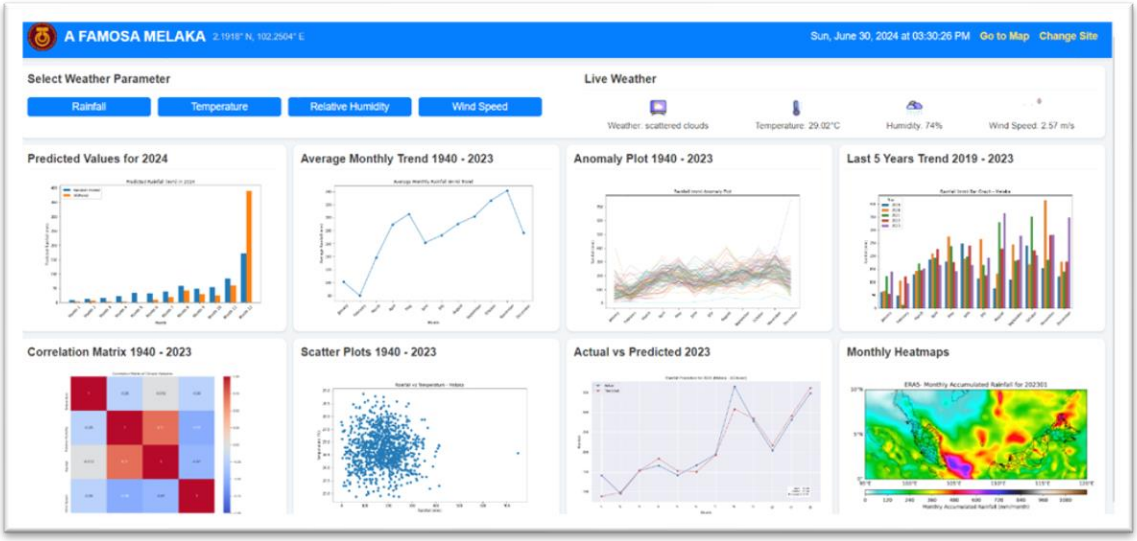
Temperature: 25.93°C

Humidity: 89%

Wind Speed: 1.03 m/s

Go to Map
Change Site
Select Weather Parameter
Rainfall
Temperature
Relative Humidity
Wind Speed
Live Weather
Predicted Values for 2024
Average Monthly Trend 1940 - 2023
Anomaly Plot 1940 - 2023
Last 5 Years Trend 2019 - 2023
Correlation Matrix 1940 - 2023
Scatter Plots 1940 - 2023
Actual vs Predicted 2023
Monthly Heatmaps
ERA5- Monthly Accumulated Rainfall for 202301

Figure 21: Johor Bahru Dashboard



A FAMOSA MELAKA 2.1918° N, 102.2504° E

Sun, June 30, 2024 at 03:30:26 PM

Weather: scattered clouds

Temperature: 29.02°C

Humidity: 74%

Wind Speed: 2.57 m/s

Go to Map
Change Site
Select Weather Parameter
Rainfall
Temperature
Relative Humidity
Wind Speed
Live Weather
Predicted Values for 2024
Average Monthly Trend 1940 - 2023
Anomaly Plot 1940 - 2023
Last 5 Years Trend 2019 - 2023
Correlation Matrix 1940 - 2023
Scatter Plots 1940 - 2023
Actual vs Predicted 2023
Monthly Heatmaps
ERA5- Monthly Accumulated Rainfall for 202301

Figure 22: Melaka Dashboard

# CHAPTER 5

# RESULT AND DISCUSSION

## 5.1 Analysis of Result

The microclimate predictions for Johor Bahru and Melaka in 2023 using Random Forest and XGBoost algorithms reveal interesting insights into the performance of these models. Both algorithms demonstrated similar prediction patterns, but with some notable differences in their accuracy and reliability. Figure 23 presents wind speed predictions for Johor Bahru and Melaka in 2023, comparing Random Forest and XGBoost models. Both algorithms capture general wind speed trends throughout the year, with XGBoost showing slightly better performance metrics. Melaka consistently experiences higher wind speeds than Johor Bahru. While the models generally track seasonal patterns, there are notable discrepancies between actual and predicted values, particularly during peak wind periods.

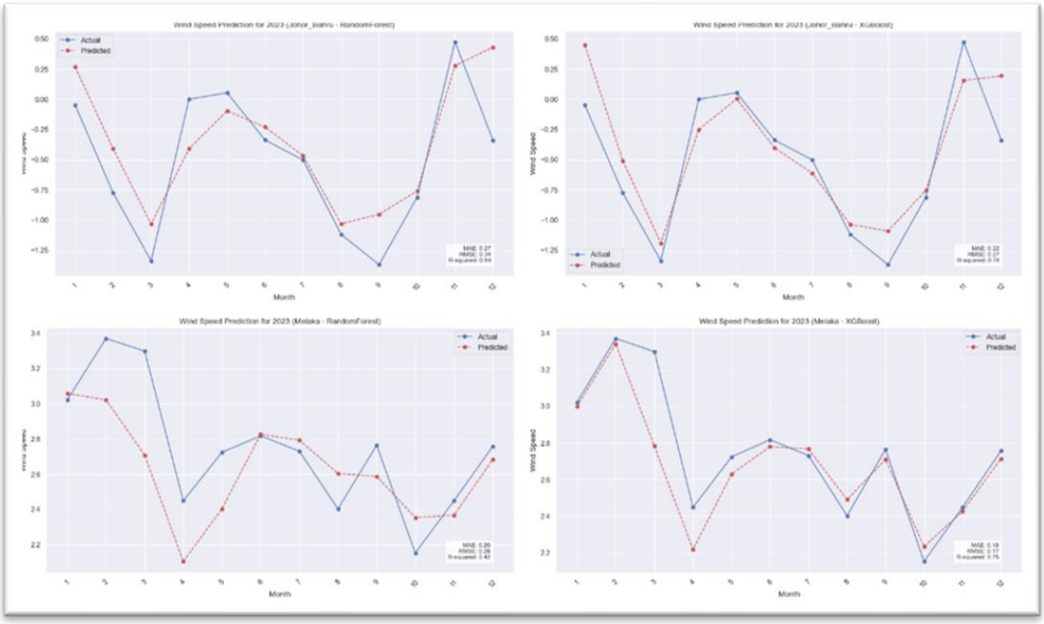


Figure 23: Wind Speed Prediction for 2023

Figure 24 illustrates temperature predictions for the same locations using the same models. Both algorithms effectively capture overall temperature patterns, with predictions appearing more accurate for Melaka than Johor Bahru. A clear temperature peak is observed around May for both areas. Interestingly, XGBoost demonstrates marginally better performance for Melaka, while Random Forest shows a slight edge in Johor Bahru predictions.

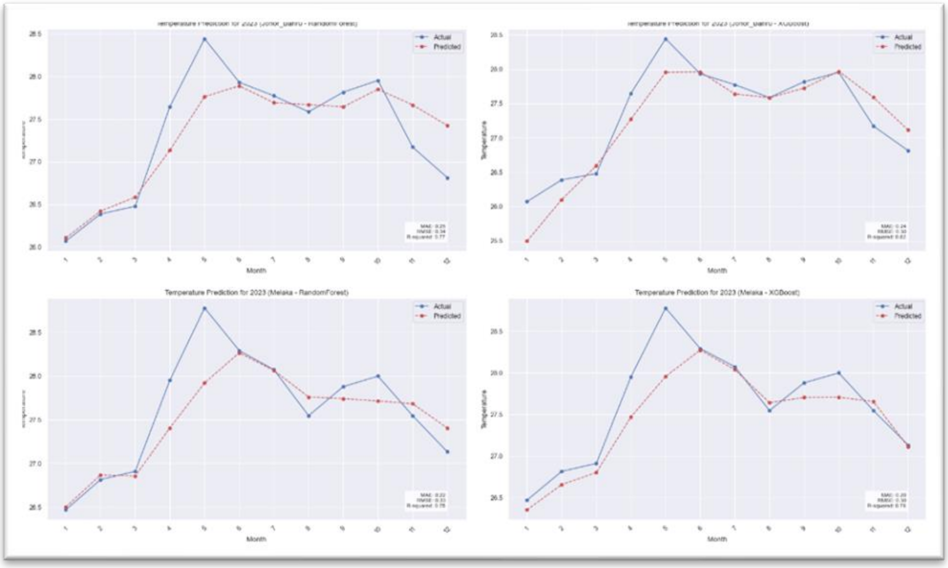


Figure 24: Temperature Prediction for 2023

Figure 25 depicts humidity predictions for 2023 in Johor Bahru and Melaka. The graphs reveal significant differences in humidity patterns between the two locations, with Johor Bahru exhibiting higher overall levels. Both models struggle to accurately predict extreme humidity fluctuations, especially in Johor Bahru. Generally, the XGBoost model outperforms Random Forest in terms of prediction accuracy for humidity across both locations.

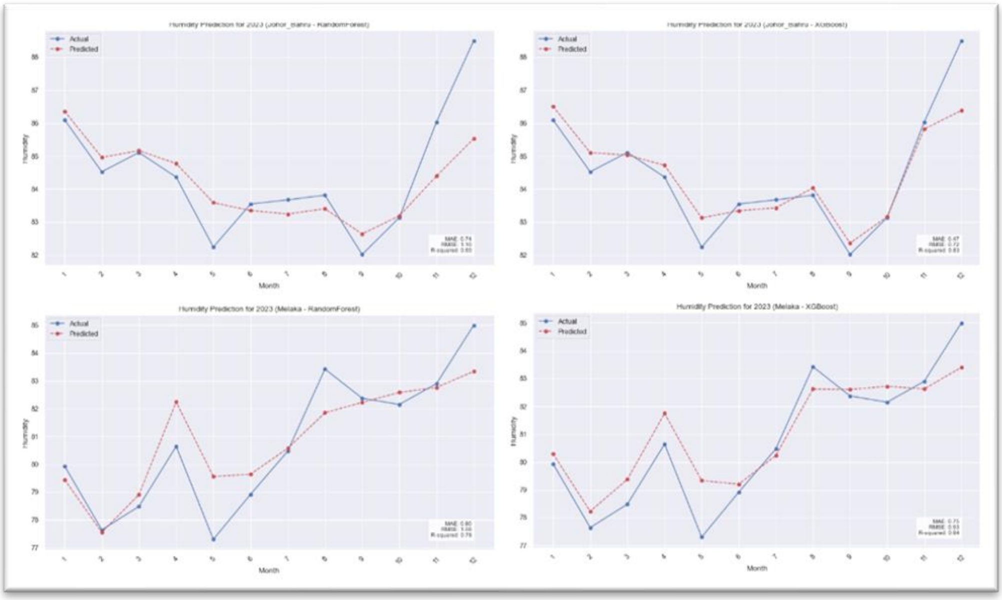


Figure 25: Humidity Prediction for 2023

Figure 26 shows rainfall predictions for both cities in 2023. The data indicates high variability in rainfall patterns throughout the year, with both models struggling to accurately predict extreme events, particularly towards year-end. Johor Bahru typically experiences higher rainfall compared to Melaka. In terms of model performance, XGBoost shows a slight advantage in predicting Melaka's rainfall, while Random Forest performs marginally better for Johor Bahru.

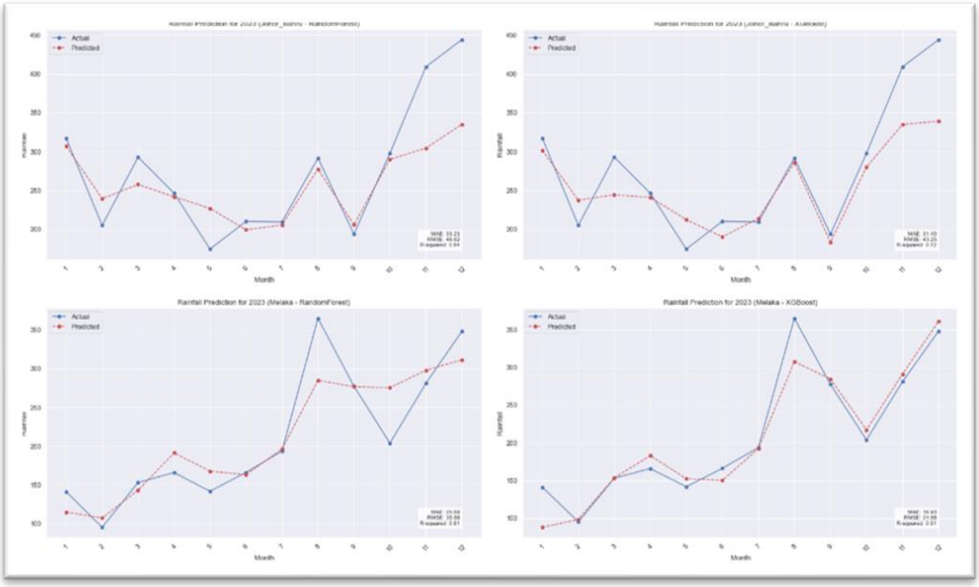


Figure 26: Rainfall Prediction for 2023

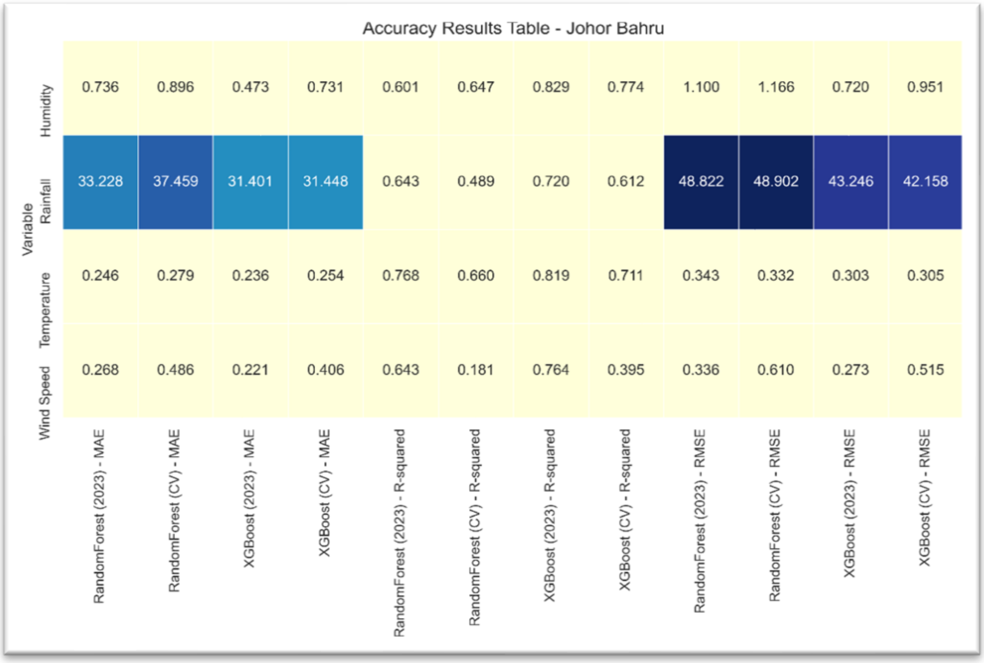
The visual representations in the graphs highlight the challenges both models face in predicting extreme values, particularly for rainfall. In the humidity graphs, both models show a general ability to follow the trend of actual humidity levels, but they struggle to capture the sharp increases or decreases, especially towards the end of the year. The rainfall graphs demonstrate a more significant disparity between predicted and actual values, with both models consistently underestimating rainfall amounts, especially during peak rainfall months.

The similar performance of Random Forest and XGBoost algorithms suggests that the limitation may lie in the features used for prediction rather than the choice of algorithm. The models' inability to accurately predict extreme events, particularly in rainfall, indicates a need for additional relevant features or a different approach to handling highly variable weather phenomena.

In conclusion, while both models show some predictive capability for humidity, they struggle significantly with rainfall prediction. The high MAE and RMSE values, coupled with negative R-squared values for rainfall, underscore the complexity of weather prediction, especially for highly variable factors like rainfall. Future work could focus on incorporating additional relevant features, exploring more advanced time series models, or considering ensemble methods that might better capture the complex patterns in weather data.

## 5.2 Evaluation of Accuracy

In Johor Bahru, the Random Forest and XGBoost models exhibited varied performance across different microclimate variables. For humidity prediction, XGBoost demonstrated superior accuracy with a MAE of 0.473 for 2023 and 0.731 for cross-validation (CV), compared to Random Forest's MAE of 0.736 and 0.896, respectively. The R-squared values also favored XGBoost, indicating a better fit for the data. In contrast, both models struggled with rainfall prediction, showing higher MAE and lower R-squared values, suggesting challenges in capturing the variability of rainfall patterns accurately. Temperature predictions were more successful for both models, with low MAE and RMSE values indicating precise forecasting abilities. Wind speed predictions showed moderate performance, with both models achieving reasonable MAE and RMSE values.

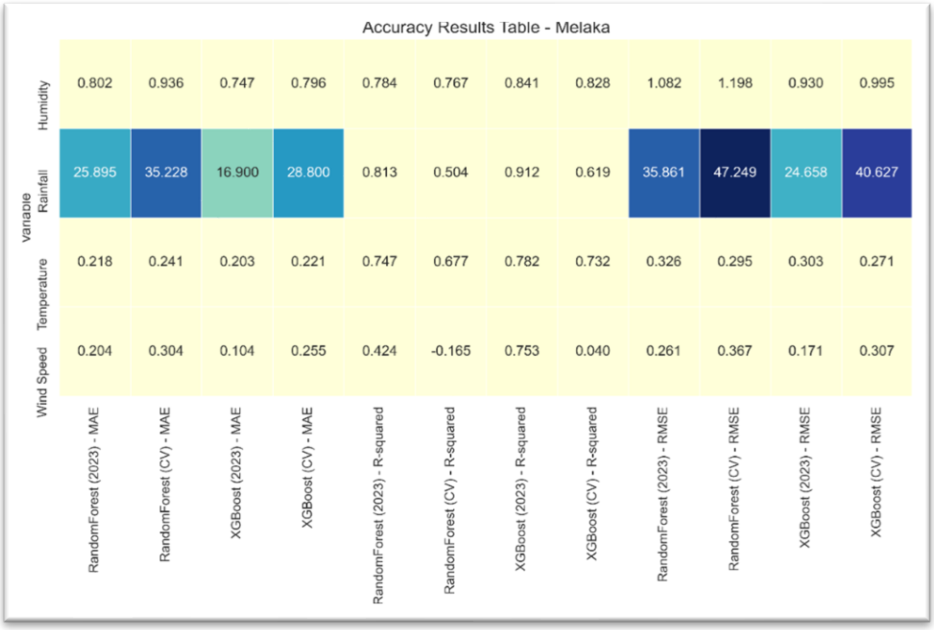


Accuracy Results Table - Johor Bahru

| Variable | RandomForest (2023) - MAE | RandomForest (CV) - MAE | XGBoost (2023) - MAE | XGBoost (CV) - MAE | RandomForest (2023) - R-squared | RandomForest (CV) - R-squared | XGBoost (2023) - R-squared | XGBoost (CV) - R-squared | RandomForest (2023) - RMSE | RandomForest (CV) - RMSE | XGBoost (2023) - RMSE | XGBoost (CV) - RMSE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Humidity | 0.736 | 0.896 | 0.473 | 0.731 | 0.601 | 0.647 | 0.829 | 0.774 | 1.100 | 1.166 | 0.720 | 0.951 |
| Rainfall | 33.228 | 37.459 | 31.401 | 31.448 | 0.643 | 0.489 | 0.720 | 0.612 | 48.822 | 48.902 | 43.246 | 42.158 |
| Temperature | 0.246 | 0.279 | 0.236 | 0.254 | 0.768 | 0.660 | 0.819 | 0.711 | 0.343 | 0.332 | 0.303 | 0.305 |
| Wind Speed | 0.268 | 0.486 | 0.221 | 0.406 | 0.643 | 0.181 | 0.764 | 0.395 | 0.336 | 0.610 | 0.273 | 0.515 |

Figure 27: Accuracy Results Table - Johor Bahru

In Melaka, XGBoost consistently outperformed Random Forest across most metrics. XGBoost showed lower MAE values for humidity, rainfall, temperature, and wind speed predictions compared to Random Forest. Particularly noteworthy were the significantly lower MAE and higher R-squared values for rainfall prediction with XGBoost, indicating its superior accuracy in capturing the complex dynamics of rainfall patterns in Melaka. However, there were indications of potential issues in model fit for wind speed predictions, as suggested by slightly negative R-squared values for XGBoost in cross-validation. Overall, both models demonstrated the potential of machine learning in accurately predicting microclimate variables, highlighting their relevance for urban planning and meteorological applications in Melaka.For humidity predictions, both Random Forest and XGBoost models showed comparable performance. The Random Forest model achieved a MAE of 1.30, RMSE of 1.51, and an R-squared value of 0.24.



Accuracy Results Table - Melaka

| Variable | RandomForest (2023) - MAE | RandomForest (CV) - MAE | XGBoost (2023) - MAE | XGBoost (CV) - MAE | RandomForest (2023) - R-squared | RandomForest (CV) - R-squared | XGBoost (2023) - R-squared | XGBoost (CV) - R-squared | RandomForest (2023) - RMSE | RandomForest (CV) - RMSE | XGBoost (2023) - RMSE | XGBoost (CV) - RMSE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Humidity | 0.802 | 0.936 | 0.747 | 0.796 | 0.784 | 0.767 | 0.841 | 0.828 | 1.082 | 1.198 | 0.930 | 0.995 |
| Rainfall | 25.895 | 35.228 | 16.900 | 28.800 | 0.813 | 0.504 | 0.912 | 0.619 | 35.861 | 47.249 | 24.658 | 40.627 |
| Temperature | 0.218 | 0.241 | 0.203 | 0.221 | 0.747 | 0.677 | 0.782 | 0.732 | 0.326 | 0.295 | 0.303 | 0.271 |
| Wind Speed | 0.204 | 0.304 | 0.104 | 0.255 | 0.424 | -0.165 | 0.753 | 0.040 | 0.261 | 0.367 | 0.171 | 0.307 |

Figure 28: Accuracy Results Table - Melaka

## 5.3 Chapter Summary

In this chapter, the results and discussion of the proposed solution for preserving cultural heritage sites through microclimate monitoring and prediction are summarized. The results of the model and the evaluation of accuracy using Random Forest and XGBoost are outlined. The parameter details are provided, along with the evaluation metrics used to assess the performance of the models. The next chapter will present the overall conclusions, highlighting the insights gained and the recommendations for future works.

# CHAPTER 6

# CONCLUSION

## 6.1 Research Outcomes

This chapter presents the outcomes of the research conducted on preserving cultural heritage sites through the application of the Random Forest and XGBoost algorithms for microclimate monitoring and prediction. The research aimed to develop an effective and efficient approach to monitor and predict the microclimate conditions at cultural heritage sites, thereby aiding in the preservation of these sites for future generations.

One of the key research outcomes is the development of a microclimate monitoring system utilizing the Random Forest and XGBoost algorithms. The system collects real-time data from various sensors deployed at cultural heritage sites, including temperature, humidity, light intensity, and air quality. The collected data is then processed and analyzed using the Random Forest and XGBoost algorithms to identify patterns and trends in microclimate conditions.

Another significant outcome of this research is the achievement of accurate microclimate prediction at cultural heritage sites. By training the Random Forest and XGBoost algorithms on historical microclimate data, the developed system can forecast future microclimate conditions with a high degree of accuracy. This prediction capability enables heritage site managers and conservationists to proactively plan and implement appropriate preservation strategies based on anticipated changes in the microclimate.

## 6.2 Contributions to Knowledge

The research conducted in this thesis has made several contributions to the field of cultural heritage preservation and microclimate monitoring. These contributions include the application of machine learning algorithms, specifically the Random Forest and XGBoost algorithms, in the context of microclimate monitoring and prediction at cultural heritage sites. By demonstrating the effectiveness of these algorithms in capturing complex relationships between various environmental factors and microclimate conditions, this research provides valuable insights into the potential of machine learning techniques for heritage site preservation.

The research presents a comprehensive framework for monitoring and predicting microclimate conditions at cultural heritage sites. This framework integrates data collection, preprocessing, analysis, and prediction using the Random Forest and XGBoost algorithms. The developed framework can serve as a guide for future researchers and practitioners in the field of cultural heritage preservation, providing a structured approach to leveraging machine learning for microclimate management.

By accurately monitoring and predicting microclimate conditions, this research contributes to the development of improved preservation strategies for cultural heritage sites. The insights gained from the analysis of microclimate data can inform decision-making processes related to site maintenance, climate control, and artifact preservation, ultimately enhancing the long-term sustainability of these important cultural assets.

## 6.3 Research Limitations

Despite the positive outcomes, this research has several limitations. One of the primary limitations is the use of monthly data instead of daily or hourly data. Utilizing more granular data could potentially improve the accuracy and reliability of the predictions. However, the available data for 2024 is incomplete, as data from January to December is still not available, which is why this research focuses on predicting 2023 data and comparing it with the actual data for that year.

Another limitation is related to hardware constraints. Due to limited computational resources, the study was conducted at only two locations. Expanding the study to include more locations would likely provide a more comprehensive understanding of microclimate variations but would also require significantly more computational power and time to process the data.

## 6.4 Future Works

While this research has achieved significant milestones in the preservation of cultural heritage sites through microclimate monitoring and prediction, there are several avenues for future research and development. Some potential areas of focus include the integration of additional data sources, such as weather forecasts, aerial imagery, and historical records, to further enhance the accuracy and reliability of microclimate prediction models. Incorporating these diverse data sets can provide a more comprehensive understanding of the factors influencing microclimate conditions and enable more robust decision-making processes.

Continued monitoring and analysis of microclimate conditions over extended periods can yield valuable insights into the long-term trends and impacts on cultural heritage sites. Future research should consider conducting longitudinal studies to capture the dynamic nature of microclimate conditions and evaluate the effectiveness of preservation strategies over time.

Encouraging collaboration and knowledge sharing among researchers, practitioners, and stakeholders in the field of cultural heritage preservation is crucial for advancing the application of microclimate monitoring and prediction techniques. Future research should focus on establishing platforms for collaboration, fostering interdisciplinary partnerships, and promoting the dissemination of research findings to maximize the impact on heritage site conservation efforts.

By addressing these future research directions, the field of microclimate monitoring and prediction for cultural heritage preservation can continue to evolve and contribute to the sustainable management of these invaluable cultural assets.

# REFERENCES

Lombardo, L., Tanyas, H., &amp; Nicu, I. C. (2020). Spatial modeling of multi-hazard threat to Cultural Heritage Sites. Engineering Geology, 277, 105776. https://doi.org/10.1016/j.enggeo.2020.105776

Fabbri, K., Bonora, A. (2021). Two new indices for preventive conservation of the cultural heritage: Predicted risk of damage and Heritage Microclimate Risk. Journal of Cultural Heritage, 47, 208–217. https://doi.org/10.1016/j.culher.2020.09.006

Pioppi, B., Pigliautile, I., Piselli, C., &amp; Pisello, A. L. (2020). Cultural Heritage Microclimate Change: Human-centric approach to experimentally investigate intra-urban overheating and numerically assess foreseen future scenarios impact. Science of The Total Environment, 703, 134448. https://doi.org/10.1016/j.scitotenv.2019.134448

Alcaraz Tarragüel, A., Krol, B., &amp; van Westen, C. (2012). Analysing the possible impact of landslides and avalanches on cultural heritage in Upper Svaneti, Georgia. Journal of Cultural Heritage, 13(4), 453–461. https://doi.org/10.1016/j.culher.2012.01.012

Sevetlidis, V., &amp; Pavlidis, G. (2019). Effective raman spectra identification with tree-based methods. Journal of Cultural Heritage, 37, 121–128. https://doi.org/10.1016/j.culher.2018.10.016

Kobayashi, K., Hwang, S.-W., Okochi, T., Lee, W.-H., &amp; Sugiyama, J. (2019). Non-destructive method for wood identification using conventional X-ray computed tomography data. Journal of Cultural Heritage, 38, 88–93. https://doi.org/10.1016/j.culher.2019.02.001

Zou, Z., Zhao, X., Zhao, P., Qi, F., &amp; Wang, N. (2019). CNN-based statistics and location estimation of missing components in routine inspection of Historic Buildings. Journal of Cultural Heritage, 38, 221–230. https://doi.org/10.1016/j.culher.2019.02.002

CC Publications Online. ICOM. (n.d.). https://www.icom-cc-publications-online.org/4417/Teaching-machines-to-think-like-conservators--Machine-learning-as-a-tool-for-predicting-the-stability-of-paper-based-archive-and-library-collections

Kejser, U. B., Ryhl-Svendsen, M., Boesgaard, C., &amp; Hansen, B. V. (n.d.). Teaching machines to think like conservators – Machine learning as a tool for predicting the stability of paper-based archive and library collections. Transcending Boundaries: Integrated Approaches to Conservation. ICOM-CC 19th Triennial Conference Preprints, Beijing, 17–21 May 2021. https://www.icom-cc-publications-online.org/4417/Teaching-machines-to-think-like-conservators--Machine-learning-as-a-tool-for-predicting-the-stability-of-paper-based-archive-and-library-collections

Pei, J., Gong, J., &amp; Wang, Z. (2020). Risk prediction of household mite infestation based on machine learning. Building and Environment, 183, 107154. https://doi.org/10.1016/j.buildenv.2020.107154

Lowenthal, D. (2015). The Past is a Foreign Country. Cambridge: Cambridge University Press.

UNESCO. (1972). Convention Concerning the Protection of the World Cultural and Natural Heritage. Paris: UNESCO.

Smith, L. (2006). Uses of Heritage. London: Routledge.

Timothy, D. J., & Boyd, S. W. (2003). Heritage Tourism. Harlow: Prentice Hall.

Cassar, M. (2005). Climate Change and the Historic Environment. London: English Heritage.

Camuffo, D. (2014). Microclimate for Cultural Heritage: Conservation, Restoration, and Maintenance of Indoor and Outdoor Monuments. Amsterdam: Elsevier.

Lankester, P., & Brimblecombe, P. (2012). The impact of future climate change on historic interiors. Science of The Total Environment, 417-418, 248-254.

Yu, T., Lin, C., Zhang, S., Wang, C., Ding, X., An, H., Liu, X., Qu, T., Wan, L., You, S., Wu, J., &amp; Zhang, J. (2022). Artificial Intelligence for Dunhuang Cultural Heritage Protection: The project and the dataset. International Journal of Computer Vision, 130(11), 2646–2673. https://doi.org/10.1007/s11263-022-01665-x

Kumar, P., Ofli, F., Imran, M., &amp; Castillo, C. (2020). Detection of disaster-affected cultural heritage sites from social media images using Deep Learning Techniques. Journal on Computing and Cultural Heritage, 13(3), 1–31. https://doi.org/10.1145/3383314

Staniforth, S. (2013). Historical Perspectives on Preventive Conservation. Getty Conservation Institute.

Stovel, H., Stanley-Price, N., &amp; Killick, R. G. (2005). Conservation of living religious heritage: Papers from the ICCROM 2003 forum on living religious history: Conserving the sacred. International Centre for the Study of the Preservation and Restoration of Cultural Property.

Muñoz Viñas (2002) Contemporary theory of conservation, Studies in Conservation, 47:sup1, 25-34, DOI: 10.1179/sic.2002.47.Supplement-1.25

Ashley-Smith, J. (2016).Risk assessment for object conservation. Routledge.

Caple, C. (2012). Preventive conservation in museums. Routledge.

Matero, F. (1999). Lessons from the Great House: Condition and treatment history as prologue to site conservation and management at Casa Grande Ruins National Monument. Conservation and Management of Archaeological Sites, 3(4), 203–224. https://doi.org/10.1179/135050399793138482

Prieto, A. J., Silva, A., de Brito, J., Macías-Bernal, J. M., &amp; Alejandre, F. J. (2017). Multiple linear regression and fuzzy logic models applied to the functional service life prediction of Cultural Heritage. Journal of Cultural Heritage, 27, 20–35. https://doi.org/10.1016/j.culher.2017.03.004

Gonthier, N., Gousseau, Y., Ladjal, S., &amp; Bonfait, O. (2019). Weakly supervised object detection in artworks. Lecture Notes in Computer Science, 692–709. https://doi.org/10.1007/978-3-030-11012-3_53

Valero, E., Forster, A., Bosché, F., Hyslop, E., Wilson, L., &amp; Turmel, A. (2019). Automated defect detection and classification in ashlar masonry walls using machine learning. Automation in Construction, 106, 102846. https://doi.org/10.1016/j.autcon.2019.102846

APPENDIX A: Gantt Chart for FYP 1

| PHASE/WEEK | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PLANNING PHASE | | | | | | | | | | | | | | | | | | | | | | | |
| Research Supervisor and Topic Research Selection | 🟩 | 🟩 | | | | | | | | | | | | | | | | | | | | | |
| Research Proposal Documentation and Submission | | | 🟩 | | | | | | | | | | | | | | | | | | | | |
| Research Proposal Interview | | | 🟩 | | | | | | | | | | | | | | | | | | | | |
| Research Proposal Correction | | | 🟩 | 🟩 | | | | | | | | | | | | | | | | | | | |
| ANALYSIS PHASE | | | | | | | | | | | | | | | | | | | | | | | |
| Chapter 1 Introduction Documentation | | | | 🟩 | 🟩 | | | | | | | | | | | | | | | | | | |
| Collection of research source, material and information searching | | | | 🟩 | 🟩 | 🟩 | | | | | | | | | | | | | | | | | |
| Chapter 2 Literature Review Documentation | | | | | | 🟩 | 🟩 | 🟩 | | | | | | | | | | | | | | | |
| Analysis of existing research | | | | | | | | 🟩 | 🟩 | 🟩 | | | | | | | | | | | | | |
| Chapter 3 Research Methodology Documentation | | | | | | | | | 🟩 | 🟩 | 🟩 | | | | | | | | | | | | |
| RESULT AND FINDINGS PHASE | | | | | | | | | | | | | | | | | | | | | | | |
| Explore solution approach | | | | | | | | | | | 🟩 | 🟩 | | | | | | | | | | | |
| Information gathering on existing application analysis | | | | | | | | | | | | 🟩 | 🟩 | | | | | | | | | | |
| Chapter 4 Research and Design Implementation Documentation | | | | | | | | | | | | | | 🟩 | 🟩 | 🟩 | | | | | | | |
| Chapter 5 Conclusion Documentation | | | | | | | | | | | | | | | | | 🟩 | | | | | | |
| Report Compilation | | | | | | | | | | | | | | | | | | 🟩 | 🟩 | 🟩 | | | |
| FYP 1 | | | | | | | | | | | | | | | | | | | | | | | |
| Presentation of FYP1 | | | | | | | | | | | | | | | | | | | | | 🟩 | | |
| Report Correction | | | | | | | | | | | | | | | | | | | | | 🟩 | 🟩 | 🟩 |

APPENDIX B: Gantt Chart for FYP 2

| PHASE/WEEK | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FYP 2 | | | | | | | | | | | | | | | | | | | |
| FYP 2 Briefing | 🟩 | | | | | | | | | | | | | | | | | | |
| ANALYSIS PHASE | | | | | | | | | | | | | | | | | | | |
| Chapter 1 Introduction Review | 🟩 | | | | | | | | | | | | | | | | | | |
| Collection of research source, material and information searching | | 🟩 | | | | | | | | | | | | | | | | | |
| Chapter 2 Literature Review | | | 🟩 | | | | | | | | | | | | | | | | |
| Analysis of existing research | | | | 🟩 | | | | | | | | | | | | | | | |
| Chapter 3 Research Methodology Review | | | | | 🟩 | 🟩 | | | | | | | | | | | | | |
| RESULT AND FINDINGS PHASE | | | | | | | | | | | | | | | | | | | |
| Explore solution approach | | | | | | 🟩 | 🟩 | | | | | | | | | | | | |
| Information gathering on existing application analysis | | | | | | | 🟩 | 🟩 | | | | | | | | | | | |
| Chapter 4 Machine Learning Model Development | | | | | | | | 🟩 | 🟩 | | | | | | | | | | |
| Experiment | | | | | | | | | 🟩 | 🟩 | 🟩 | 🟩 | | | | | | | |
| Chapter 5 Dashboard Development | | | | | | | | | | | 🟩 | 🟩 | 🟩 | | | | | | |
| Chapter 6 Conclusion Documentation and Review | | | | | | | | | | | | | 🟩 | 🟩 | | | | | |
| Thesis Compilation | | | | | | | | | | | | | | | | | | | |
| FYP 2 | | | | | | | | | | | | | | | 🟩 | | | | |
| Presentation of FYP1 | | | | | | | | | | | | | | | 🟩 | 🟩 | 🟩 | | |
| Thesis Correction | | | | | | | | | | | | | | | | | | 🟩 | 🟩 |