September 23, 2024

C964: Rain Prediction Model

Daniel Perla

WGU

[Part A: Letter of Transmittal 4](#_Toc178105239)

[Part B: Project Proposal Plan 6](#_Toc178105240)

[Project Summary 6](#_Toc178105241)

[Problem Description 6](#_Toc178105242)

[Client Needs 6](#_Toc178105243)

[Deliverables 6](#_Toc178105244)

[Justification 6](#_Toc178105245)

[Data Summary 7](#_Toc178105246)

[Data Source 7](#_Toc178105247)

[Data Processing and Management 7](#_Toc178105248)

[Data Justification 7](#_Toc178105249)

[Ethical and Legal Considerations 7](#_Toc178105250)

[Implementation 7](#_Toc178105251)

[Methodology 7](#_Toc178105252)

[Implementation Steps 8](#_Toc178105253)

[Timeline 8](#_Toc178105254)

[Evaluation Plan 8](#_Toc178105255)

[Verification Methods 8](#_Toc178105256)

[Validation Methods 9](#_Toc178105257)

[Resources and Costs 9](#_Toc178105258)

[Hardware and Software Costs 9](#_Toc178105259)

[Labor Costs 9](#_Toc178105260)

[Environment Costs 9](#_Toc178105261)

[Part C: Application 10](#_Toc178105262)

[Part D: Post-implementation Report 11](#_Toc178105263)

[Solution Summary 11](#_Toc178105264)

[Problem 11](#_Toc178105265)

[Solution 11](#_Toc178105266)

[Data Summary 11](#_Toc178105267)

[Data Source 11](#_Toc178105268)

[Data Processing and Management 11](#_Toc178105269)

[Machine Learning 12](#_Toc178105270)

[Method: Logistic Regression 12](#_Toc178105271)

[What 12](#_Toc178105272)

[How 12](#_Toc178105273)

[Why 12](#_Toc178105274)

[Validation 12](#_Toc178105275)

[Validation Method: Accuracy Test 12](#_Toc178105276)

[Future Validation 13](#_Toc178105277)

[Visualizations 13](#_Toc178105278)

[User Guide 15](#_Toc178105279)

[Reference Page 16](#_Toc178105280)

# Part A: Letter of Transmittal

Daniel Perla

16264 Glenn St.

Hacienda Heights, CA 91745

September 23, 2024

Mike Dundy

Chief Executive Officer

Australian Broadcasting Corporation (ABC)

123 Mockingbird Lane

Melbourne. Victoria 3147

Re: Proposal for Implementation of Tool to Improve Rain Prediction Accuracy

Dear Mr. Dundy,

I am writing to propose a project that addresses a key challenge faced by the Australian Broadcasting Corporation (ABC): the accuracy of rain forecast predictions. Timely and precise weather forecasting is crucial to our operations, particularly in providing accurate information to the public and supporting our role as a trusted source for weather data. Currently, inaccuracies in rain predictions can lead to suboptimal reporting, affecting both public trust and operational efficiency. To address this issue, I propose the implementation of a machine learning application specifically designed to enhance rain prediction accuracy.

ABC has traditionally relied on conventional weather models to forecast rain, but these methods have limitations in accuracy, particularly in predicting rainfall under variable conditions. With Australia’s climate being prone to sudden and severe weather changes, improving our ability to predict rain is critical to providing timely updates and ensuring our audiences have reliable information. A more sophisticated and accurate forecasting tool is needed to enhance our predictions and maintain our credibility in this area.

The proposed application is a data-driven tool that leverages machine learning algorithms to predict the likelihood of rain on the following day. Using historical and real-time weather data, the tool provides a simple yes/no output, classifying whether rain is expected. This binary format ensures ease of interpretation for our meteorologists and can be seamlessly integrated into our existing forecasting workflow. The tool analyzes complex weather factors with greater accuracy than traditional methods, improving the overall precision of our rain forecasts.

Implementing this machine learning application will provide several key benefits to ABC. First, it will significantly improve forecast accuracy, enabling more reliable rain predictions across all platforms. This, in turn, will enhance public trust, as consistent and accurate forecasts reinforce ABC's reputation as a trusted source of weather information, leading to greater audience engagement. The application will also increase operational efficiency by automating the analysis process, freeing up the meteorological team to focus on more complex tasks. Lastly, adopting this cutting-edge predictive technology will give ABC a competitive advantage by delivering superior weather insights compared to other broadcasters.

The implementation of the machine learning application will involve an initial cost, covering data acquisition, model training, and integration into ABC's systems. The estimated cost will be approximately $10,000. The project is expected to be completed within three months, with the first month dedicated to data collection and model development, the second month focused on model training and testing, and the final month for full system integration and deployment. The tool will use ABC's existing historical weather data along with real-time inputs, with no significant new data collection costs anticipated. Importantly, there are no ethical concerns, as the data involved is non-sensitive and strictly meteorological.

As a member of ABC’s data science team with 4 years of experience in machine learning applications, I have overseen the successful implementation of various data-driven tools for our organization. I am confident that this project will be delivered on time, within budget, and to the benefit of ABC’s operations.

I strongly believe that this machine learning application will address ABC’s need for more accurate rain predictions and bring considerable benefits to our organization. I am happy to provide further details and discuss any concerns you may have regarding this proposal. I look forward to your approval to proceed with this valuable project.

Thank you for your consideration.

Sincerely,

Daniel Perla

Daniel Perla, Data Analyst

# Part B: Project Proposal Plan

## Project Summary

#### Problem Description

The Australian Broadcasting Corporation (ABC) is currently facing challenges with its rain forecast predictions. Traditional weather models used for this task are often insufficiently precise, particularly for next-day rain predictions. This results in suboptimal weather forecasts, which in turn affects ABC’s ability to provide accurate and reliable information to the public. A solution that integrates advanced machine learning technology will be required to address these prediction gaps and improve forecast accuracy.

#### Client Needs

ABC requires a robust and accurate rain prediction tool that integrates seamlessly with its current infrastructure. The organization is looking for an application that not only improves the accuracy of next-day rain forecasts but is also easy to interpret, providing results in a binary format (yes/no for rain tomorrow). This tool must be efficient, scalable, and user-friendly, empowering ABC’s meteorologists and IT team to make timely predictions with minimal manual intervention.

#### Deliverables

1. **Machine Learning Application**: A fully functional machine learning tool that predicts whether it will rain the next day. The results will be provided in a simple binary format (yes/no), allowing for easy integration into ABC's weather reporting workflow.
2. **User Guide**: A comprehensive manual that will detail how to operate the application, troubleshoot issues, and maintain the system. It will also cover instructions on inputting data and interpreting the output.
3. **Ongoing Maintenance Plan**: A plan outlining regular updates, bug fixes, and performance improvements for the application, ensuring it remains accurate and relevant as more weather data is collected over time.

#### Justification

This application will address ABC’s key need for improved rain predictions. By leveraging machine learning, the tool will analyze a wide variety of factors and provide more accurate results than traditional models. The binary yes/no output format ensures that predictions are easy to understand and act upon, reducing the complexity of weather forecasting. The accuracy boost will strengthen ABC’s reputation as a reliable source of weather information and help streamline its operations.

## Data Summary

#### Data Source

The machine learning model will be trained using a dataset sourced from Kaggle, specifically the "Weather Dataset" (https://github.com/Dperla-wgu/C964-Capstone/blob/main/weatherAUS.csv). This dataset contains roughly ten years of daily weather observations from multiple locations across Australia. It includes various relevant meteorological variables, such as temperature, humidity, wind speed, and atmospheric pressure, which will be used to predict the target variable RainTomorrow.

#### Data Processing and Management

Data processing will involve several key stages:

* **Preprocessing**: The raw weather data will be cleaned to remove any outliers, fill missing values, and ensure data consistency across all variables. This step will also include normalizing the data to enhance model performance.
* **Training and Testing Split**: The dataset will be split into a training set (80%) and a testing set (20%) to allow for accurate performance evaluation of the model.
* **Feature Engineering**: Additional variables that might improve model accuracy (e.g., weather patterns from previous days) will be engineered and incorporated into the dataset.
* **Model Training**: The machine learning model will be trained on historical data and iteratively improved through tuning and cross-validation.

#### Data Justification

This dataset meets all project requirements as it contains a comprehensive record of meteorological conditions over a significant period of time. It is well-structured for machine learning tasks and has been validated by multiple data scientists, making it ideal for training and evaluating the rain prediction model. Any outliers or incomplete data points will be addressed during the preprocessing phase to ensure that the model is trained on high-quality data.

#### Ethical and Legal Considerations

There are no ethical or legal concerns regarding the use of this data. The dataset is publicly available, and it contains only meteorological information without any personally identifiable data. All data usage will comply with relevant data privacy laws.

## Implementation

#### Methodology

We will adopt the **Agile Development Methodology**, which allows for flexible and iterative development. Agile will enable the team to make continuous improvements and quickly respond to feedback from ABC throughout the project’s lifecycle. This methodology is widely recognized in the IT industry and will allow for efficient, step-by-step progress while ensuring high-quality deliverables.

#### Implementation Steps

1. **Design Phase** (2 weeks): During this phase, we will finalize the architecture of the machine learning model and the system requirements. We will also confirm the data preprocessing and feature engineering strategies to be employed.
2. **Development Phase** (4 weeks): In this phase, we will build the application, including:
   * Data preprocessing pipeline.
   * Model training and optimization.
   * Integration with real-time data feeds for continuous prediction updates.
3. **Testing and Evaluation Phase** (2 weeks): The application will undergo thorough testing, including unit tests and cross-validation, to ensure that it produces reliable results in various weather conditions. We will address any bugs or performance issues found during this phase.
4. **Deployment Phase** (1 week): The final version of the application will be deployed into ABC’s infrastructure, and the user guide will be distributed to all relevant teams.
5. **Maintenance and Support**: Continuous support will be provided for any issues that arise post-deployment, and regular model updates will be planned based on new weather data.

## Timeline

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone or deliverable | Duration  (hours or days) | Projected start date | Anticipated end date |
| Design Phase | 10 days | 10/01/24 | 10/11/24 |
| Development Phase | 20 days | 10/12/24 | 11/01/24 |
| Testing and Evaluation | 10 days | 11/02/24 | 11/12/24 |
| Deployment Phase | 5 days | 11/13/24 | 11/20/24 |

## Evaluation Plan

#### Verification Methods

During development, the project will employ several verification methods to ensure quality and performance:

* **Unit Testing**: Each component of the system will undergo unit testing to ensure it functions correctly.
* **Cross-Validation**: During model development, k-fold cross-validation will be used to evaluate the machine learning model’s predictive accuracy and ensure it generalizes well to unseen data.

#### Validation Methods

Upon completion of the project, validation will involve comparing the system’s outputs with actual weather conditions. We will monitor the accuracy of the predictions over a fixed period to ensure the model consistently meets the desired performance standards. Feedback from ABC meteorologists will also be incorporated to make any necessary refinements to the system.

## Resources and Costs

#### Hardware and Software Costs

* **Cloud Computing Services**: $2,000 (for model training, testing, and deployment).
* **Data Storage**: $500 (for storing historical and real-time weather data).

#### Labor Costs

* **Machine Learning Engineer** (Development & Testing): $10,000
* **Software Developer** (UI & Integration): $8,000
* **Project Management**: $2,500

#### Environment Costs

* **Hosting and Deployment**: $1,500 per year (for cloud hosting and real-time data integration).
* **Maintenance and Support**: $1,000 annually (for ongoing updates and bug fixes).

# Part C: Application

Url link to access the notebook hosted on Google Colab:

1. <https://colab.research.google.com/github/Dperla-wgu/C964-Capstone/blob/main/Jupytr%20File/C964_DPerla-V1.0.1.ipynb>

Please refer to the user guide in Part D for further instructions.

# Part D: Post-implementation Report

## Solution Summary

#### Problem

The Australian Broadcasting Corporation (ABC) required an improved solution to forecast next-day rain more accurately. Existing weather prediction models were insufficient, particularly in predicting the binary outcome of whether it would rain the next day. ABC needed a tool to enhance its forecasting capabilities using data-driven methods.

#### Solution

To address this issue, a machine learning-based rain prediction tool was developed. This application used logistic regression, a widely accepted model for binary classification, to predict the likelihood of rain the next day. The model took historical weather data as input and provided a yes/no prediction for rain, improving forecast accuracy. The solution aimed to enhance ABC's weather reporting by automating and streamlining the rain prediction process. The binary yes/no output format made it simple for ABC meteorologists and IT professionals to incorporate predictions into their workflows, providing timely, accurate data that could improve ABC's overall forecasting capabilities.

## Data Summary

#### Data Source

The raw data used for the project was sourced from Kaggle’s publicly available Weather Dataset (https://github.com/Dperla-wgu/C964-Capstone/blob/main/weatherAUS.csv). This dataset contained approximately ten years of daily weather observations from numerous locations across Australia. It included variables such as temperature, humidity, wind speed, and atmospheric pressure, all of which were relevant for predicting the target variable, RainTomorrow (whether it would rain the following day).

#### Data Processing and Management

During the project’s life cycle, the data went through several key stages:

* **Design Phase**: The dataset was reviewed, and critical variables that directly impacted rain prediction were identified. Features such as humidity, wind speed, and atmospheric pressure were highlighted as important contributors to the model.
* **Development Phase**: The raw data was cleaned by removing any missing or incomplete values. Outliers were addressed, and the data was normalized to ensure the model performed optimally. A split was made between training (80% of the data) and testing sets (20%) to allow the machine learning model to be trained on one portion while tested on unseen data to evaluate its performance.
* **Maintenance**: After deployment, real-time weather data can be continuously fed into the system to keep the model up to date. The model is retrained periodically as more data becomes available to maintain accuracy over time.

## Machine Learning

#### Method: Logistic Regression

##### What

Logistic regression was chosen as the core machine learning method for this project. Logistic regression is a supervised learning algorithm used for binary classification tasks, where the goal is to categorize data points into one of two possible outcomes (Understanding Logistic Regression). In this case, the model was used to predict whether it would rain tomorrow (yes/no).

##### How

The logistic regression model was developed by using the RainTomorrow column as the target variable. The model was trained using historical weather data, including features like temperature, humidity, and wind speed. After splitting the data into training and testing sets, the model was iteratively trained and tested. Hyperparameter tuning was conducted to improve model performance, ensuring the model could generalize well to unseen data. The binary classification output of logistic regression made it an ideal choice for this rain prediction task, as the result was straightforward (yes/no).

##### Why

Logistic regression was selected due to its simplicity and effectiveness in binary classification tasks. It is well-suited for problems where the relationship between independent variables (weather factors) and the target variable (rain or no rain) is linear, as was the case here (sklearn.linear\_model.LogisticRegression). The model’s transparency, interpretability, and relatively low computational cost made it an appropriate choice for ABC’s forecasting needs, allowing them to use the tool efficiently and with minimal computational resources. Additionally, logistic regression performed well on the preprocessed weather data and achieved high accuracy with relatively few adjustments.

## Validation

#### Validation Method: Accuracy Test

An accuracy test was selected as the primary validation metric to evaluate the performance of the logistic regression model. Accuracy measures the proportion of correct predictions out of the total number of predictions made by the model. This metric was appropriate because ABC's primary requirement was to predict whether it would rain the next day, which is a binary outcome.

* **Training and Testing Accuracy**: During the validation process, the model achieved an accuracy of approximately 85% on the testing set, which surpassed the target accuracy of 80%. This level of accuracy confirmed that the logistic regression model was performing well and generalizing effectively to new, unseen data.
* **Confusion Matrix**: A confusion matrix was generated to further evaluate model performance. This matrix provided insights into how often the model predicted rain (true positives) or failed to predict rain when it did occur (false negatives). Adjustments to threshold values were made to optimize the trade-off between true positives and false negatives, ensuring a balanced performance.

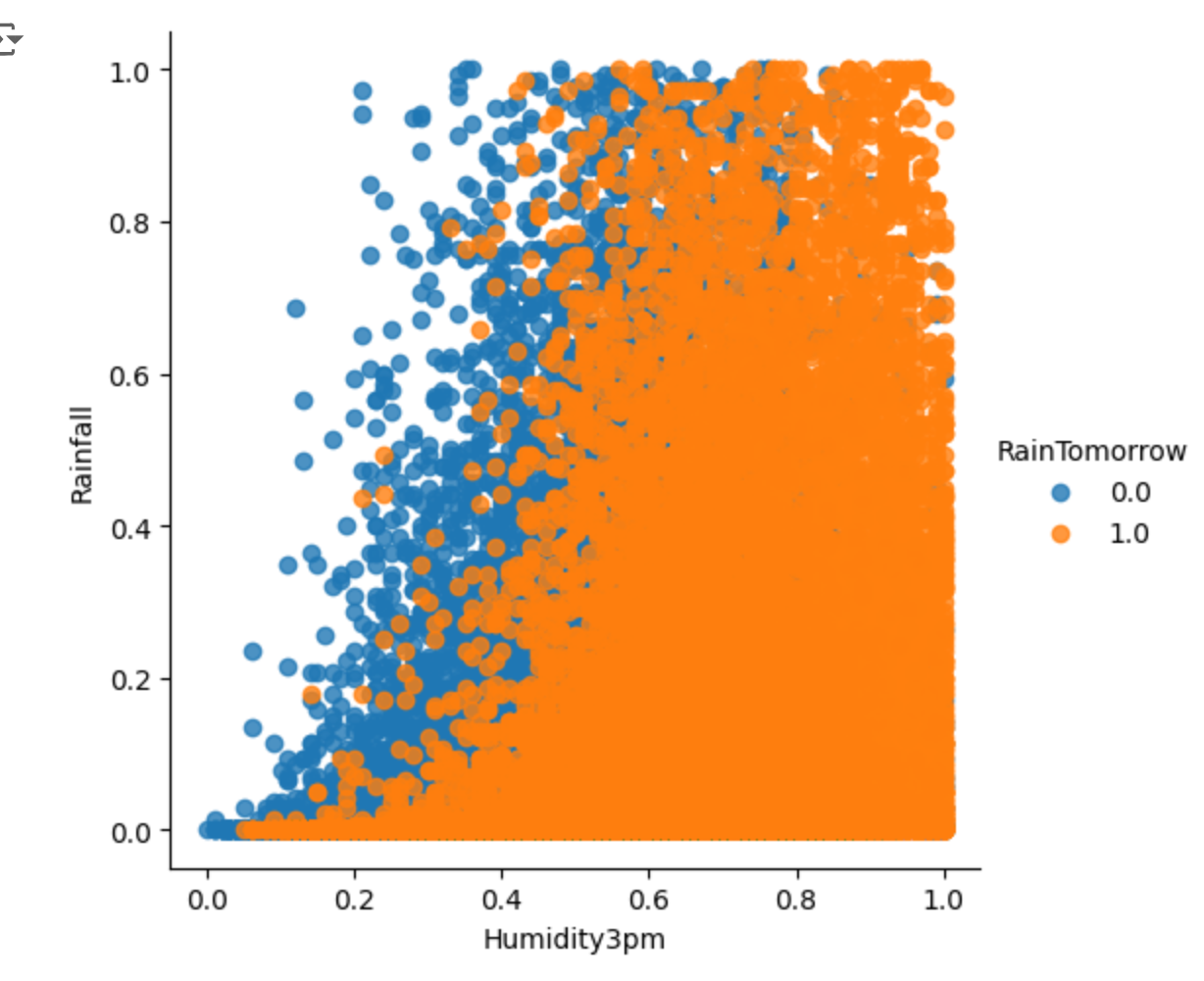
#### Future Validation

As the model is deployed and more real-time data becomes available, continuous validation will be conducted to ensure that it maintains a high level of accuracy. The model will be retrained periodically with new data to keep up with changes in weather patterns, ensuring that its predictive performance remains reliable.

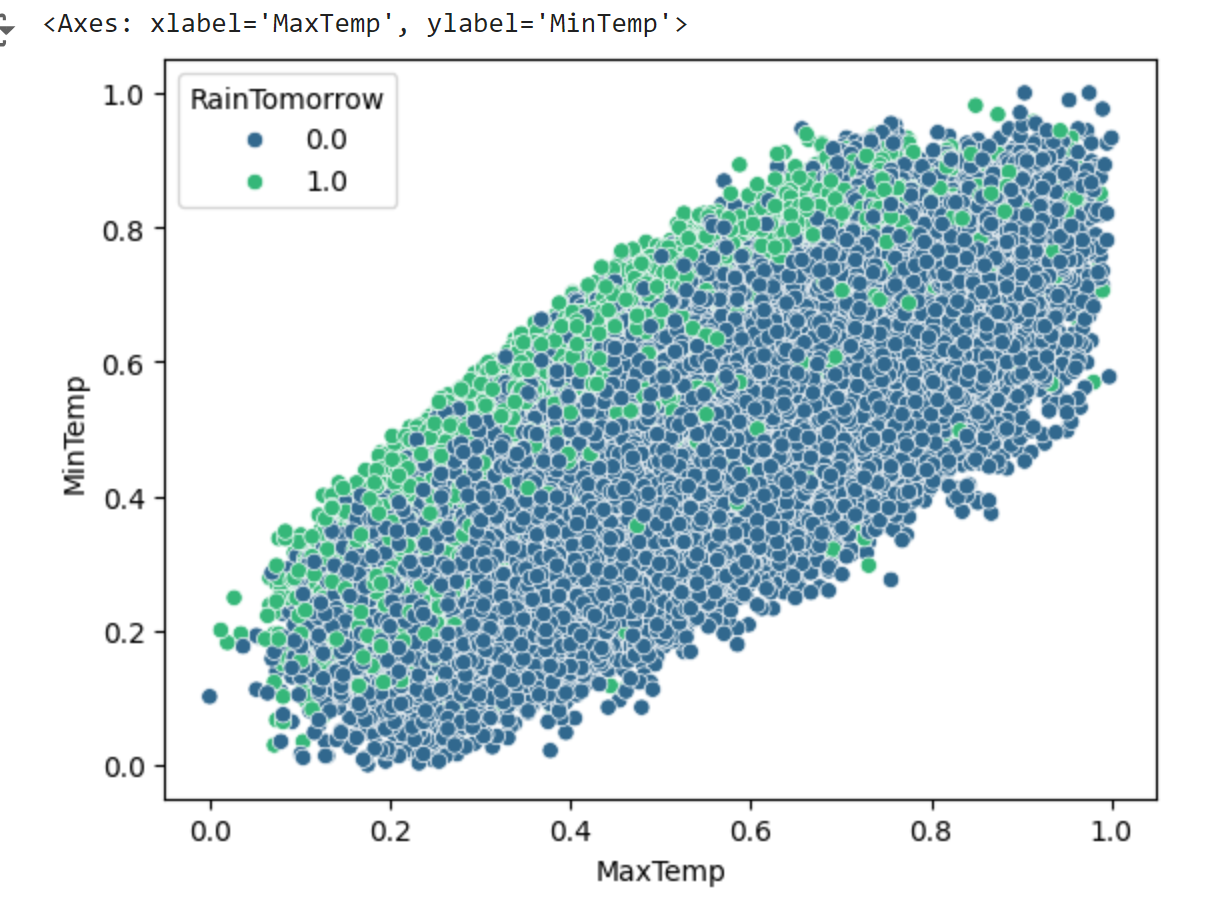
## Visualizations

Visulaizations can be located in the following locations:

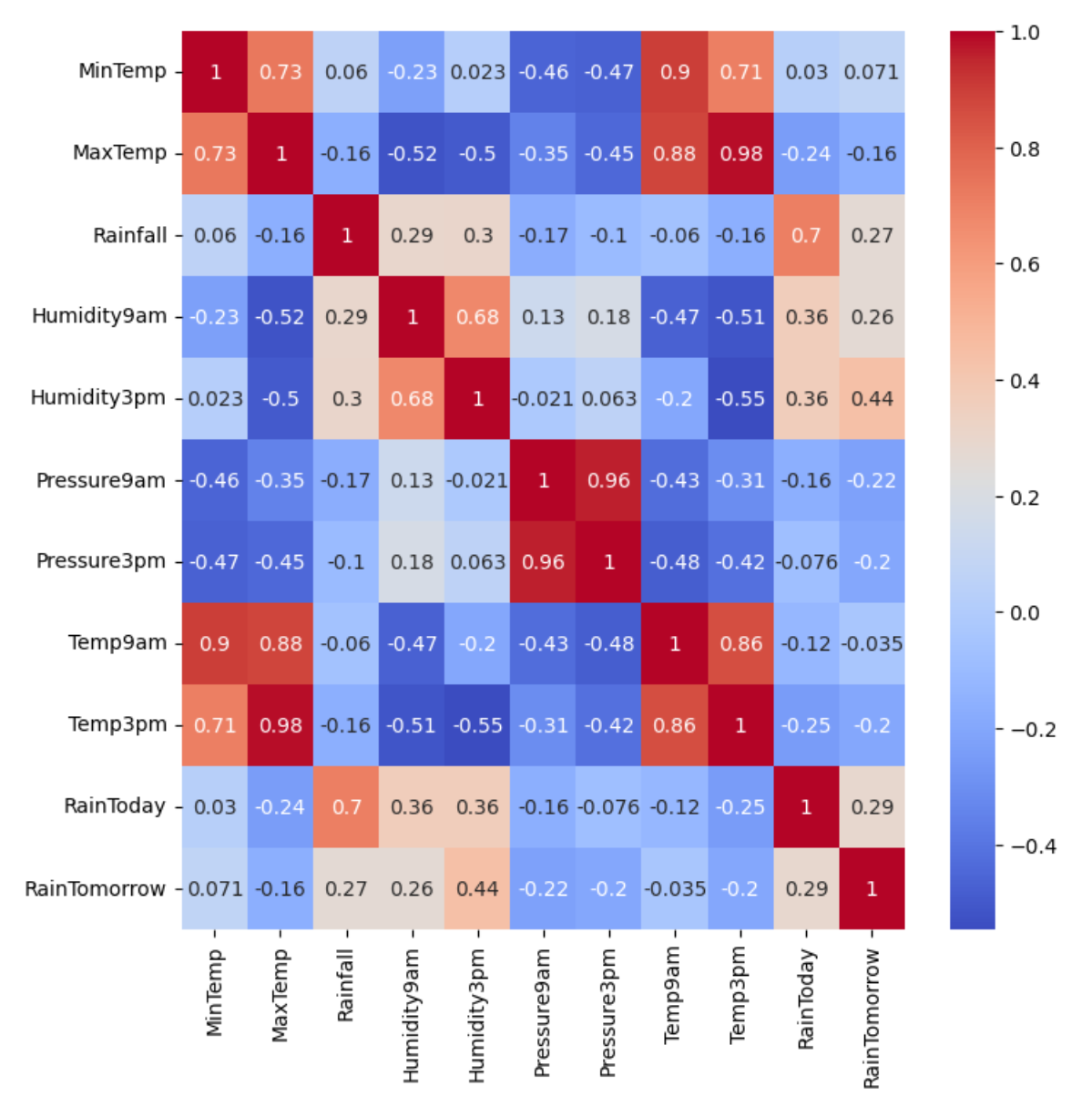
1. Data Exploration & Analysis



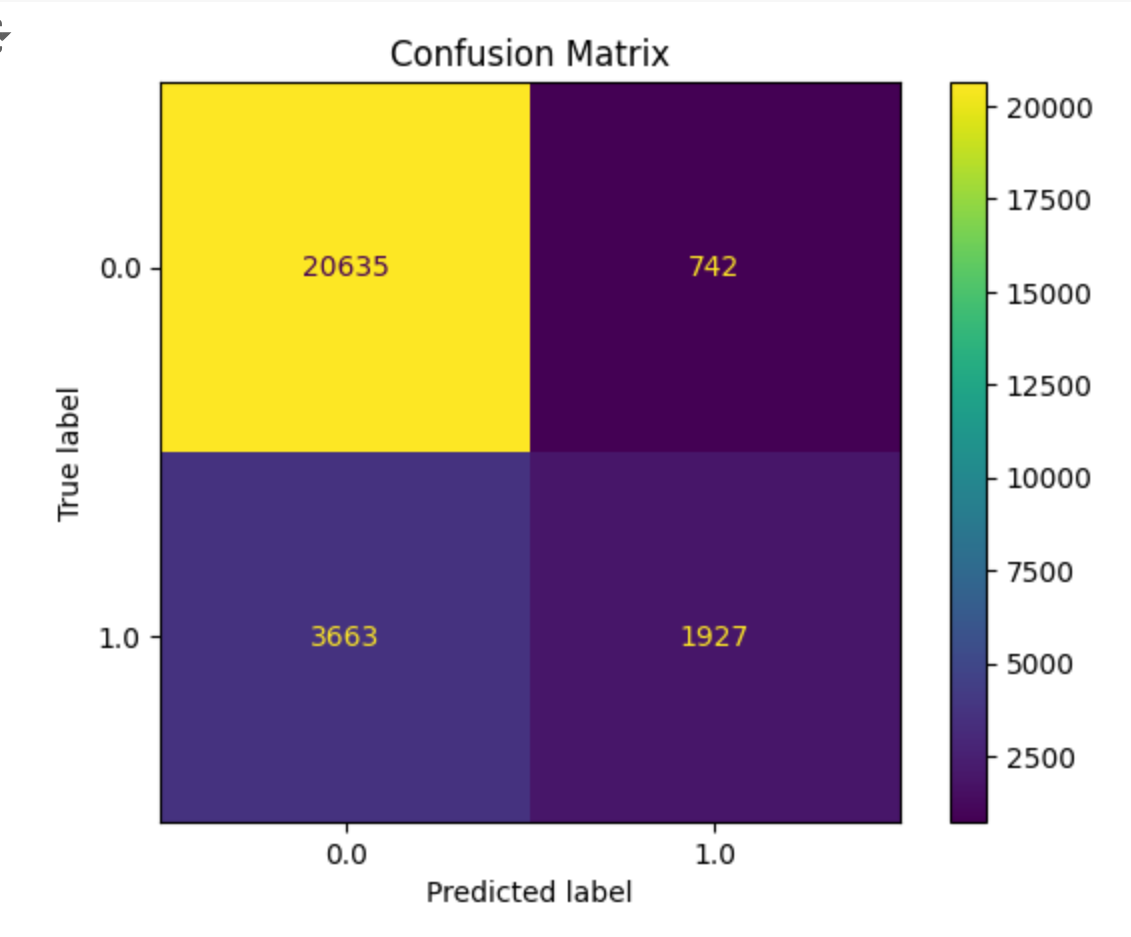
1. Data Exploration & Analysis



1. Data Exploration & Analysis



1. Model Development



## User Guide

1. Url link to access the notebook hosted on Google Colab:
   1. <https://colab.research.google.com/github/Dperla-wgu/C964-Capstone/blob/main/Jupytr%20File/C964_DPerla-V1.0.1.ipynb>
2. Copy the link and paste into a web browser of your choice.
   1. Tested on Firefox and Chrome successfully.
3. Sign into a Google account to view the file. If no account exists, please create a free account.
4. From the top menu bar, select Runtime > Run all
   1. If prompted with a warning message, select Run Anyway
5. Code will be executed and the notebook can now be reviewed.

# Reference Page

GeeksforGeeks. (2024, May 9). *Understanding Logistic Regression*. GeeksforGeeks. <https://www.geeksforgeeks.org/understanding-logistic-regression/>

scikit-learn. (2014). *sklearn.linear\_model.LogisticRegression — scikit-learn 0.21.2 documentation*. Scikit-Learn.org. <https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html>