

Decision Tree-Based Weather Factor Analysis for Bandar Lampung Flood Event Classification

Ridho Sholehurrohman¹, Reza Habibi²

¹*Department of Computer Science, FMIPA, University of Lampung, Bandar Lampung, Indonesia*

²*Department of Business Statistics, Faculty of Vocational Studies, ITS, Indonesia*

²*Department of Computer Science and Electrical Engineering, Kumamoto University, Japan*

*email@address: ridho.sholehurrohman@fmipa.unila.ac.id, reza.habibi@its.ac.id.

ABSTRACT

Flood events in Bandar Lampung City, Indonesia, are often triggered by extreme weather factors such as high rainfall, high humidity, and fluctuating air temperatures. This study applies the Decision Tree model as an analytical tool to examine the influence of these weather factors on flood occurrences using a dataset of 50 historical records containing information on rainfall, air temperature, humidity, and wind direction. The methodology includes data exploration, preprocessing, a single train–test split (80:20), and model evaluation using a confusion matrix. The exploratory analysis revealed that daily and weekly rainfall are the dominant factors contributing to flood events. The Decision Tree model, trained using default parameters, achieved a prediction accuracy of 90% on the test data. The visualization of the decision tree and feature importance analysis reinforce the finding that rainfall is the primary predictor of flood events. As a preliminary proof of concept, this study demonstrates that the Decision Tree model provides an interpretable approach for flood prediction. However, given the limited dataset size, further validation using larger datasets is recommended before operational implementation in early warning systems for Bandar Lampung City.

Keywords: Decision Tree, Flood Classification, Weather Analysis, Preliminary Study, Bandar Lampung.

1. INTRODUCTION

Flood events in Bandar Lampung City are often triggered by extreme weather factors such as high rainfall, high humidity, and fluctuating air temperatures. Additionally, with the geographical conditions located on the coast and the flat topography of the area, this city becomes more vulnerable to flood disasters. Given the increasingly extreme and unpredictable weather phenomena, an initial predictive system based on weather factors is needed as a proof of concept, although further validation with larger datasets is required [1], [2].

Flooding is one of the natural disasters that frequently occurs in various regions, including Bandar Lampung City. Flood events in this area are often caused by a combination of extreme weather factors and unsupportive environmental conditions. Bad weather, such as high rainfall, unstable air temperatures, and increased humidity, can worsen the situation and cause flooding that harms the community [3], [4]. Given the significant impacts caused by floods, it is very important to understand the factors

causing this disaster so that mitigation efforts and flood event predictions can be carried out effectively [5].

One approach that can be used to analyze the factors influencing flood events is the application of machine learning-based prediction models, particularly the Decision Tree. This model can map the relationship between various weather variables, such as air temperature, humidity, wind, and rainfall, with flood events, so it can be used to predict the likelihood of future flooding [6], [7]. Decision Tree was chosen because of its ability to produce clearly interpretable models, which is very useful in the context of natural disaster prediction, where fast and accurate decisions are needed. In addition, the Decision Tree also has the ability to handle large and complex data without requiring complicated parametric structures, and can handle missing and incomplete data issues, which often occur in weather data [8], [9].

Several previous studies have shown that rainfall is the main factor influencing flood occurrence. However, other factors such as air temperature, humidity, and wind can also contribute significantly to flood events. Therefore, it is important to comprehensively analyze how the combination of these factors interacts and contributes to flood events. Decision Tree can be used to identify patterns in weather data and, despite limited data availability, can provide initial predictions of flood occurrences based on these patterns [10], [11].

Decision Tree was chosen in this study because of its ability to handle complex relationships without requiring complicated parametric structures, even when working with limited datasets. This model can also handle missing data and incomplete variables, which often appear in weather data [9]. In addition, the Decision Tree model has been proven effective in producing accurate predictions in various applications, including disaster risk analysis and weather forecasting, due to its ability to handle complex data and produce easily interpretable models [12], [13].

By using the Decision Tree model, this study aims to analyze and model the relationship between weather factors and flood events in Bandar Lampung as an initial exploratory study. The results of this study are expected to provide preliminary insight into the factors causing floods. Although the dataset is limited in size, this research can serve as a foundation for future flood disaster mitigation studies using more comprehensive data. The application of this model demonstrates a potential approach that, after further validation with larger datasets, could eventually support relevant agencies in planning preventive actions.

2. MATERIAL AND METHODS

This chapter describes the research framework, data exploration and preprocessing steps, the Decision Tree algorithm used, and the model evaluation metrics applied in this study.

2.1 Research Framework

This study applies the Decision Tree algorithm to analyze weather factors on flood events in Bandar Lampung. The stages of the research implementation are shown in Figure 1.

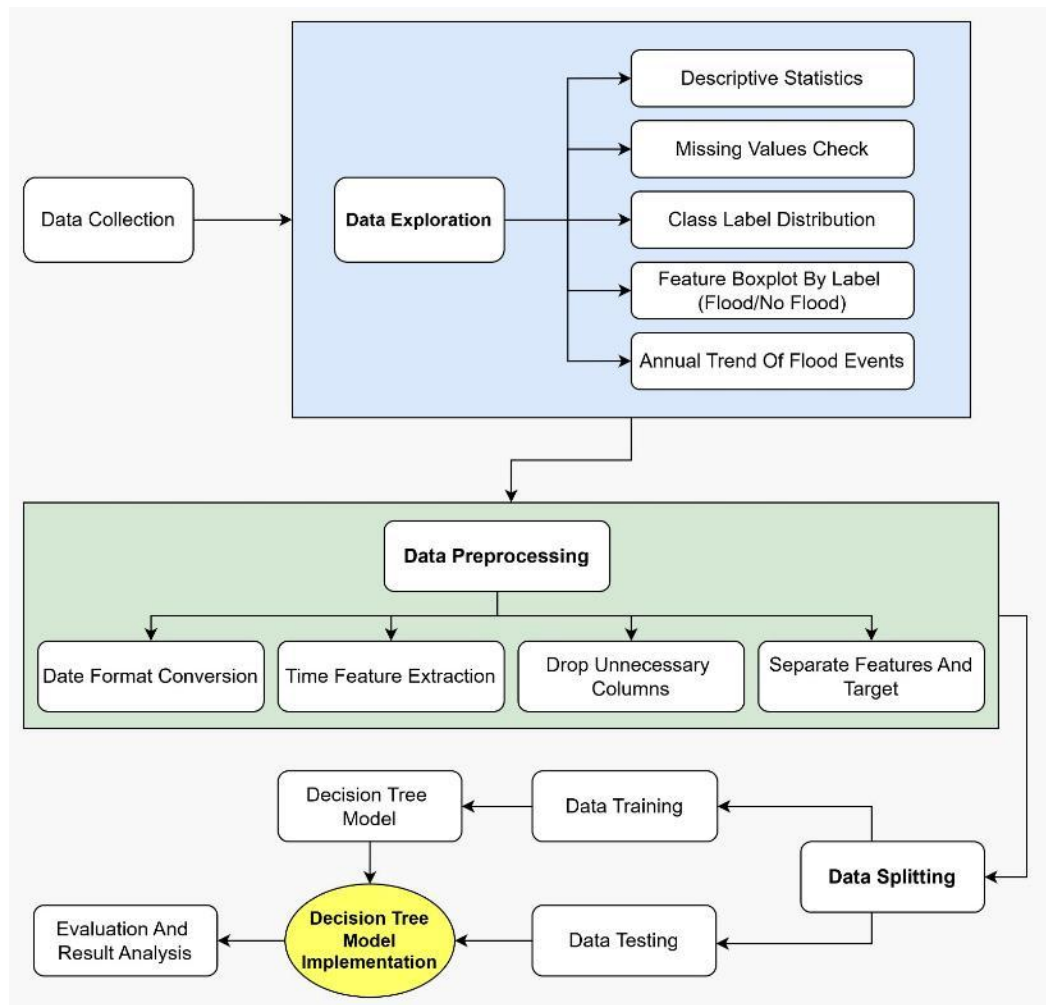


FIGURE 1. Decision Tree Research Framework

As illustrated in Figure 1, the research framework consists of several sequential stages. The first stage is data collection, where historical flood-related datasets (consisting of 50 samples) are gathered as the primary source for analysis. The collected data then undergoes data exploration and preprocessing, followed by data splitting into training and testing sets. Subsequently, the Decision Tree model is trained and tested, leading to model implementation. The framework concludes with evaluation and result analysis to assess model performance. This systematic workflow ensures proper model development, implementation, and performance validation.

2.2 Data Exploration and Preprocessing

In the Exploratory Data Analysis (EDA) stage, the main objective is to understand data characteristics through descriptive statistical analysis and visualization, to identify patterns, relationships between variables, and detect anomalies or outliers. This process includes checking feature distribution, target class balance, and correlation between variables. EDA provides important insights in selecting main features used in modeling and allows for a better understanding of data structure [14].

After that, the data preprocessing stage aims to prepare data before being used in modeling. Preprocessing steps include data format conversion, handling missing values, variable transformation, and data normalization or standardization. In addition, data is divided into two parts, namely training data and testing data, to ensure objective model evaluation and avoid overfitting [15].

2.3 Decision Tree Algorithm

Decision Tree is a machine learning algorithm used for classification and regression, which divides data based on certain features repeatedly to form an easily interpretable decision tree structure [16], [17]. In the process of making a decision tree, this algorithm uses several main functions to determine data splitting quality. Gini Impurity is used in CART (Classification and Regression Tree) to measure uncertainty at each data split. Gini impurity is calculated by the formula (1).

$$Gini(p) = 1 - \sum_{i=1}^k p_i^2 \quad (1)$$

where p_i is the proportion of classes in the node. The Entropy function, which is used in ID3, measures data uncertainty and is calculated by the formula (2).

$$Entropy(S) = \sum_{i=1}^k p_i \log_2(p_i) \quad (2)$$

In addition, Information Gain is used to select the best feature in data splitting based on entropy reduction. The calculation of Information Gain is by formula (3) as follows.

$$InformationGain = Entropy(Parent) - \sum \left(\frac{|Subset|}{|Parent|} \times Entropy(Subset) \right) \quad (3)$$

After the tree is formed, to prevent the model from overfitting, pruning is performed. Pruning can be pre-pruning, which stops tree formation early, or post-pruning, which prunes tree branches after formation to avoid excessive complexity. Although Decision Tree has advantages in interpretability and does not require complicated data preprocessing, this model is prone to overfitting on trees that are too deep. To overcome this, ensemble methods, such as Random Forest and Gradient Boosting, are used to improve accuracy by combining several decision trees, which produces a more stable and accurate model [18]-[20].

2.4 Model Evaluation

Evaluation of the Decision Tree model is carried out using a confusion matrix, which compares prediction results with original labels and produces four main components: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). From these components, metrics such as accuracy, precision, recall, and f1-score are calculated [21].

TABLE 1.
Model Evaluation Metrics

Metric	Formula
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$
Precision	$\frac{TP}{TP + FP}$
Recall	$\frac{TP}{TP + FN}$
F1-Score	$2 \times \frac{Precision \times Recall}{Precision + Recall}$

The confusion matrix is very useful for identifying model weaknesses, such as low recall in minority classes, which indicates that the model fails to detect some positive events (e.g., rare events like disasters). This happens because the model may classify many positive events as negative, which are often referred to as false negatives. Conversely, high precision indicates that the model is very careful in predicting positive classes, thus reducing the risk of false positives (errors in predicting positive events that do not actually occur), but with the possibility of under-detection in minority classes [17], [22]. This evaluation is important to adjust the focus of model development according to the application context. In the case of flood prediction, recall is often prioritized so that all potential flood events can be detected, even if there are some incorrect predictions [23].

3. RESULTS AND DISCUSSION

In this study, the data analysis process and application of the Decision Tree model followed a systematic flow consisting of several stages, allowing the identification of weather factors that significantly influence flood events.

3.1 Data Collection

Research data was obtained from an online KaggleHub repository in CSV format accessible via the link: <https://www.kaggle.com/datasets/ramadhannurpambudi/dataset-kejadian-banjir-dan-variabel-pendukung>. The dataset consists of 50 entries containing historical information on weather conditions and flood event status. Variables included are: air temperature, humidity, surface wind direction, monsoon wind direction, daily rainfall, total weekly rainfall, and flood event label (1 = occurred, 0 = did not occur). Although not explicitly mentioning the geographical location, this data is assumed to represent meteorological and hydrological conditions in the Bandar Lampung City area, in line with the context and focus of analysis in this study. This dataset became the basis for building the Decision Tree model to analyze the relationship between weather factors and flood events.

3.2 Data Exploration

Data exploration was conducted to understand the initial characteristics of the dataset before being applied to the Decision Tree model. This process included descriptive statistical analysis, checking target class balance, feature distribution against flood events, and correlation between weather variables. First, the descriptive statistics of the dataset used provide an initial overview of the value distribution of each weather variable analyzed in Figure 2 as follows.

Statistik Deskriptif:							
	Rata-Rata Suhu Udara (°C)	Rata-Rata Kelembapan (%)	Rata-Rata Angin Permukaan (°)	Musim - Angin Monsun (°)	Jumlah Curah Hujan (mm)	Total Curah Hujan 1 Minggu (mm)	Terjadi Banjir (1=Ya, 0=Tidak)
count	50.000000	50.000000	50.000000	50.000000	50.000000	50.000000	50.000000
mean	26.722000	84.432000	189.200000	246.000000	32.420000	54.780000	0.420000
std	1.351234	7.309665	105.401855	99.713876	39.464316	58.129795	0.498569
min	24.000000	68.000000	20.000000	130.000000	2.000000	0.000000	0.000000
25%	25.550000	78.250000	110.000000	130.000000	5.000000	12.000000	0.000000
50%	26.900000	84.500000	180.000000	330.000000	12.500000	40.000000	0.000000
75%	27.900000	90.875000	290.000000	330.000000	48.500000	79.000000	1.000000
max	28.800000	98.500000	360.000000	330.000000	166.000000	264.000000	1.000000

FIGURE 2. Descriptive Statistics of the Dataset

Based on Figure 2, the average air temperature is 26.72°C with a range between 24.0°C and 28.8°C, indicating relatively stable temperatures. Average humidity is 84.43% with moderate variation. Surface wind direction shows high variation (standard deviation 105.40°), while monsoon wind tends to be centered at 330°. Rainfall has a wide distribution, with daily rainfall averaging 32.42 mm and a maximum of 166 mm, and weekly rainfall averaging 54.78 mm with a maximum of 264 mm. This condition reflects the potential for extreme rain events that can trigger floods. The target variable "Flood Occurrence" shows 42% flood events from total observations, so the class distribution can be considered balanced. Overall, these results reinforce the suspicion that rainfall and humidity are important factors that need to be further analyzed in the flood prediction model. Next, Figure 3 show the distribution of the target class (Flood Occurrence) is visualized in a bar chart using a countplot.

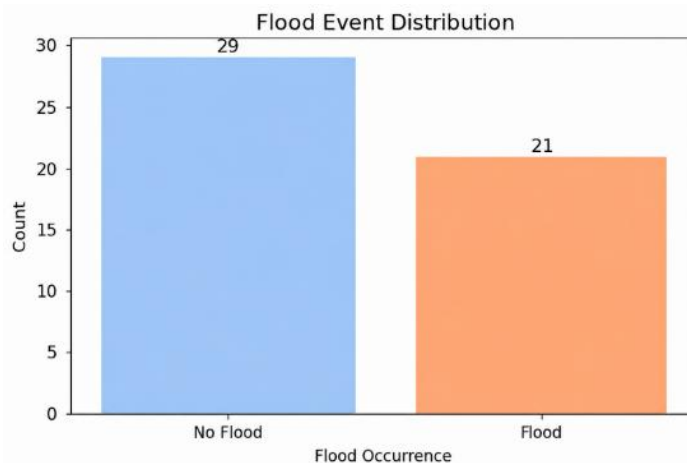


FIGURE 3. Distribution of Target Class in the Dataset

Based on Figure 3, the distribution of the target class from 2010 to 2020 shows that there are 21 flood cases and 29 non-flood cases, meaning the dataset is relatively balanced and suitable for use without data balancing techniques. Third, to see the distribution of weather variable values between flood and non-flood conditions, data visualization was performed using boxplots for several main features as follows.

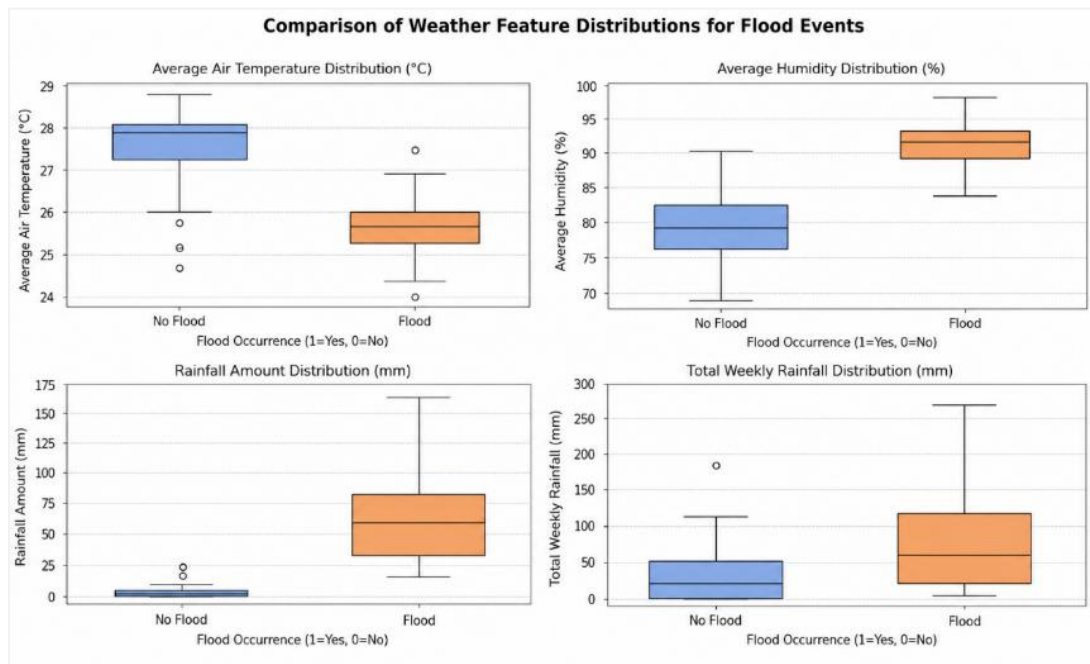


FIGURE 4. Boxplot of Weather Feature Distribution against Flood Events

The visualization results in Figure 4 show that the distribution of weekly and daily rainfall is significantly higher in the "flood" class compared to the "non-flood" class. This is indicated by the median value and interquartile range (IQR) shifting upward in the flood class. Some outliers are also seen in weekly rainfall, indicating extreme events that may directly contribute to flooding. For humidity, the median and distribution also appear slightly higher in flood events, although the difference is not too striking. Meanwhile for air temperature, the difference between the two classes is not significant, and the distribution is relatively similar, indicating that air temperature may not be a dominant factor in triggering floods.

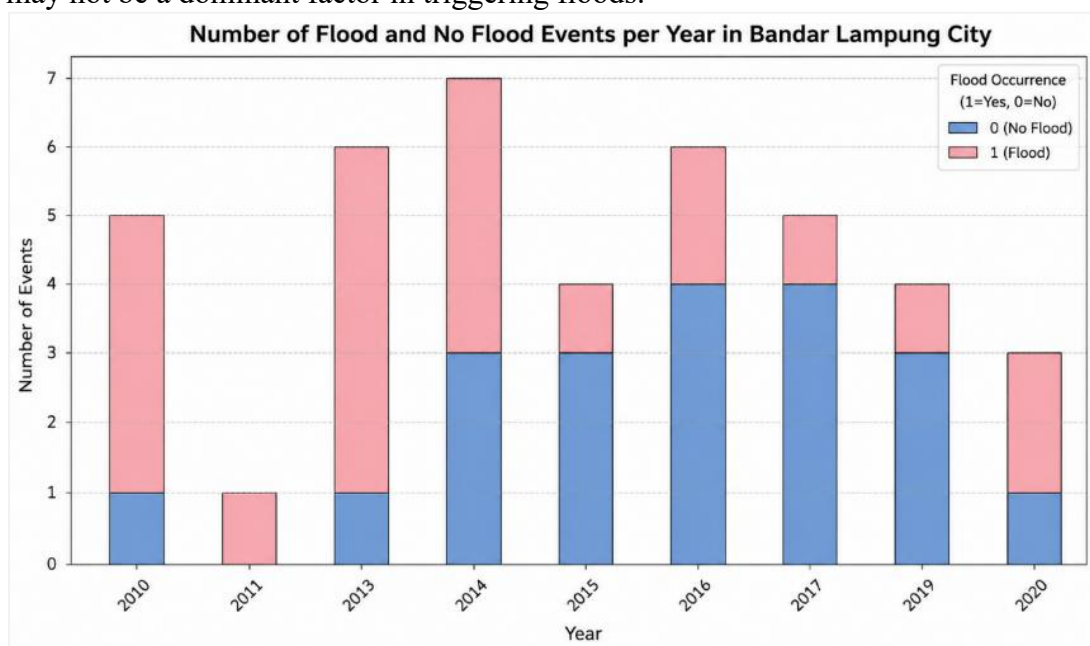


FIGURE 5. Visualization of Flood Event Distribution per Year from 2010-2020

Correlation analysis between features shows that Total Weekly Rainfall and Daily Rainfall have the strongest relationship with flood events, while wind direction shows low correlation. To support temporal analysis, the Flood Date column was converted to datetime format and turned into a Year feature.

3.3 Data Preprocessing

The preprocessing stage was carried out to ensure that the dataset used in modeling is clean, consistent, and ready to be analyzed using the Decision Tree algorithm. The steps taken in data preprocessing in this study are as follows:

- ✓ **Date Format Conversion**

The data in the Flood Date column was originally a string with month names in Indonesian. To facilitate temporal processing and analysis, month names were converted to English, then converted to datetime format using the `pd.to_datetime()` function from the pandas library.

- ✓ **Time Feature Extraction**

After successful date conversion, the year component was extracted from the Flood Date column. This year information is stored in a new column named Year, which is useful for flood event trend analysis by year, and can potentially be used as an additional feature in modeling.

- ✓ **Removal of Unnecessary Columns**

The Flood Date column that has been converted and extracted is no longer used in the modeling process because it is non-numerical and does not directly contribute to predictions. Therefore, this column was removed from the dataset to avoid interference in model training.

- ✓ **Separation of Features and Target**

The data was then separated into two main parts: Features (X): all numerical columns representing weather and time variables (except labels). Target (y): Flood Occurrence column (1=Yes, 0=No), which indicates whether a flood occurred or not. This separation aims to separate input and output so that they can be used in training and evaluating the Decision Tree classification model.

After the data was processed and separated into features (X) and target (y), the next step was to split the data into two parts using the `train_test_split` function with a proportion of 80% for training and 20% for testing. The `random_state=42` parameter was used to keep the split results consistent when repeated. This split aims to train the model on some data and test its performance on data that has never been seen before.

3.4 Decision Tree Model Implementation

The Decision Tree model was applied to the training data that had been previously processed. The algorithm used was `DecisionTreeClassifier` from the scikit-learn library with initial parameters `criterion='gini'` and `max_depth=None`. The model was trained using training data (X_train, y_train), then tested on testing data (X_test) to evaluate its performance.

```
model = DecisionTreeClassifier(criterion='gini', max_depth=None, random_state=42)
model.fit(X_train, y_train)
```

DecisionTreeClassifier

DecisionTreeClassifier(random_state=42)

FIGURE 6. Decision Tree Model Training

After the Decision Tree model was trained using the training data, the next step was to test the model's performance on the testing data that had been previously separated. This process aims to determine the extent to which the model can generalize patterns from the training data to new data that has never been seen before. Evaluation was carried out by comparing the model's prediction results with the actual labels on the test data.

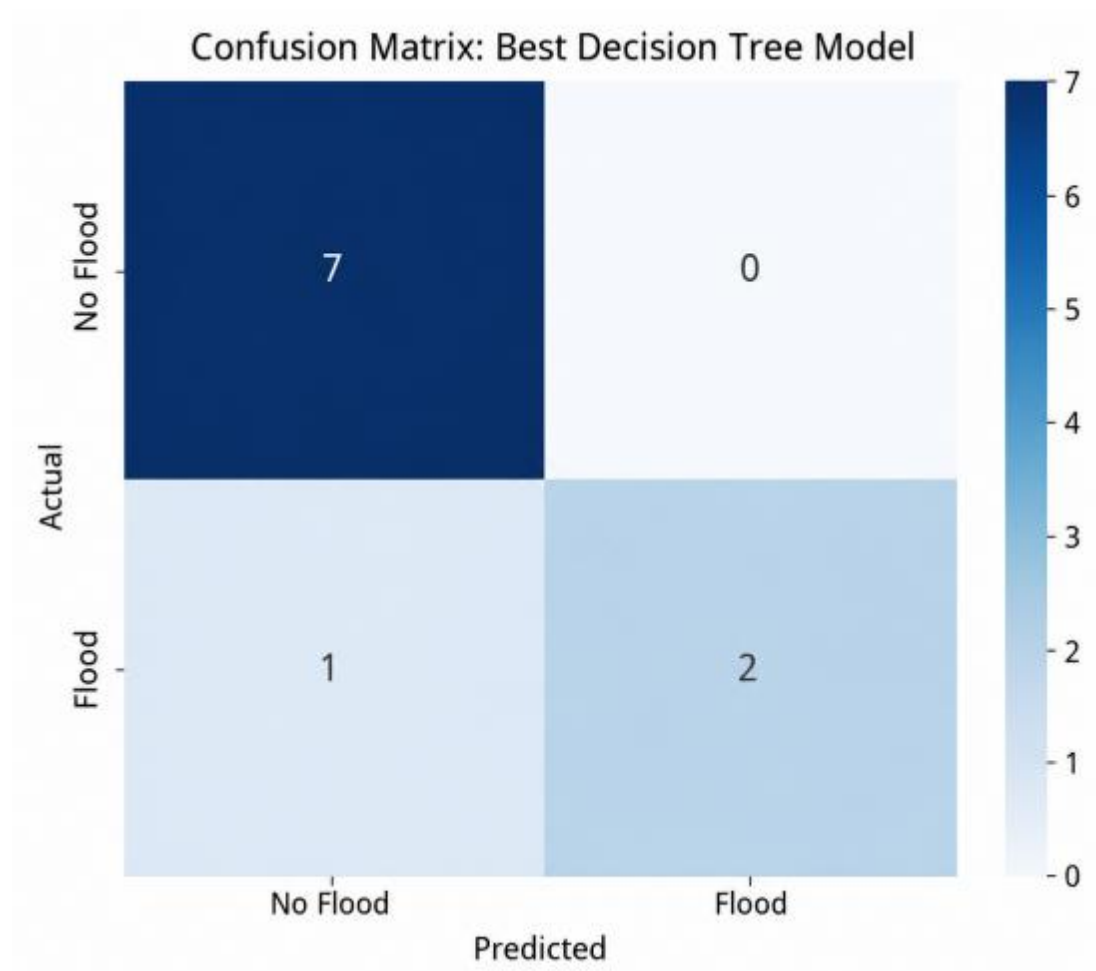


FIGURE 7. Decision Tree Model Evaluation of Confusion Matrix

Based on the evaluation results figure 7 shown in the confusion matrix, the Decision Tree model was able to classify the test data quite well. Of the total ten test data, nine data were successfully predicted correctly, while one data experienced misclassification. This resulted in an accuracy value of 90%, which indicates that the

model has a high level of accuracy in distinguishing between flood and non-flood events. This accuracy reflects that the model can be used as a tool to predict potential floods based on available weather factors. Further evaluation of each class is shown in Figure 8, which shows precision, recall, and f1-score for flood and non-flood classes.

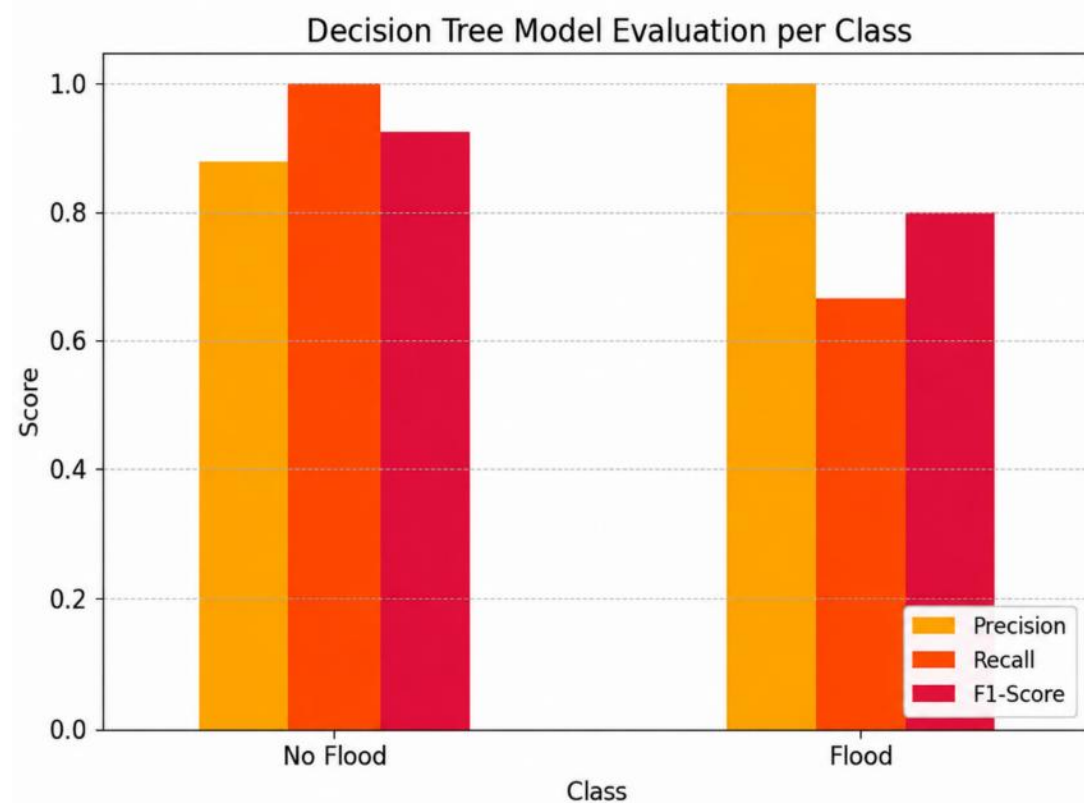


FIGURE 8. Model Evaluation Based on Precision, Recall, and F1-Score

Figure 8 shows the performance evaluation of the Decision Tree model based on precision, recall, and f1-score for each class, namely Flood and Non-Flood. In the Non-Flood class, the model showed excellent performance with a precision of 0.88, recall of 1.00, and f1-score of 0.93. This indicates that all non-flood events were correctly recognized by the model, with a very low error rate. Meanwhile, for the Flood class, precision reached a perfect 1.00, which means all flood predictions did indeed occur. However, the recall for this class was only 0.67, indicating that the model still misses some actual flood cases. The f1-score for the flood class was 0.80, reflecting a fairly good balance between accuracy and sensitivity. Overall, the model tends to be careful in predicting floods, thus reducing false positives, but still has room for improvement in terms of detecting all actual flood events.

To further understand how the model performs classification of flood events based on available weather features, a visualization of the resulting decision tree structure was carried out. This visualization provides an overview of the data separation logic used by the model, and shows what features are the main criteria in decision making.

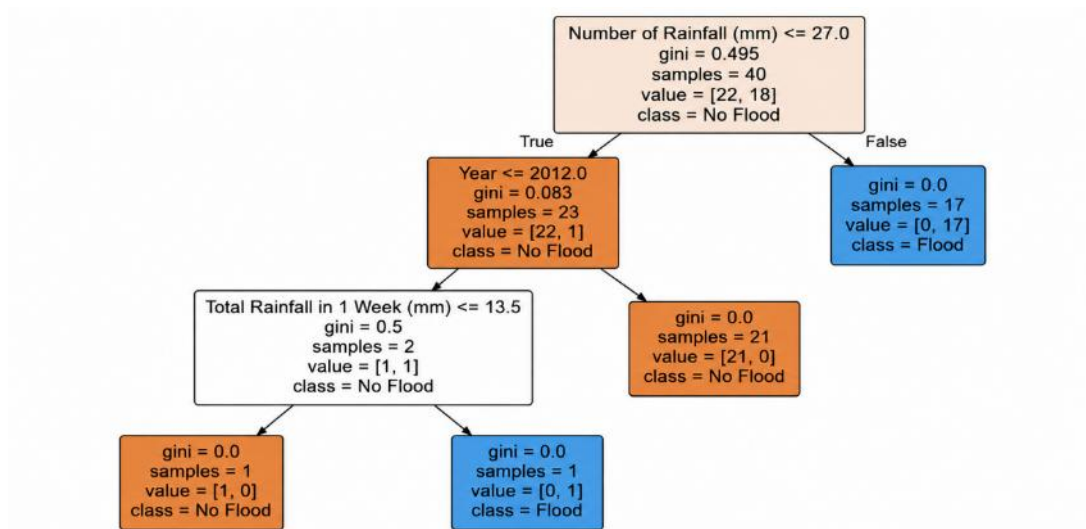


FIGURE 9. Decision Tree Visualization

The decision tree visualization by Figure 9 shows the logical structure used by the model to separate data between flood and non-flood events based on weather features. The first branch or root of the tree starts from the Daily Rainfall feature with a threshold of ≤ 27 mm, which is the main criterion in determining the initial classification. This shows that daily rainfall is the most dominant factor in influencing flood predictions.

If the daily rainfall exceeds 27 mm, the model immediately classifies the event as a flood, with a gini value of 0.0, which means all data in that node is homogeneous (truly flood). Conversely, if daily rainfall is ≤ 27 mm, the model continues to the next branch based on $\text{Year} \leq 2012$, which shows that historical data has an influence on the model's decisions, although the significance of the year feature is not too high.

On the next path, when $\text{year} \leq 2012$, the model evaluates Total Weekly Rainfall with a threshold of 13.5 mm. This node has an impurity level (gini = 0.5), indicating ambiguity or imbalance between flood and non-flood labels. This branch then divides into two pure nodes (gini = 0.0) each representing one flood data and one non-flood data. It can be seen in the decision tree visualization that the model is able to separate data well using the main features of daily rainfall, weekly rainfall, and on a smaller scale, the year of occurrence. The depth of the tree that is not too deep also indicates that the model does not overfit, and can still be well interpreted for decision-making purposes.

Next, Feature Importance analysis is presented with the aim of identifying which weather features most influence flood event predictions, so that model results are easier to interpret and can be used as a basis for more appropriate decision making.

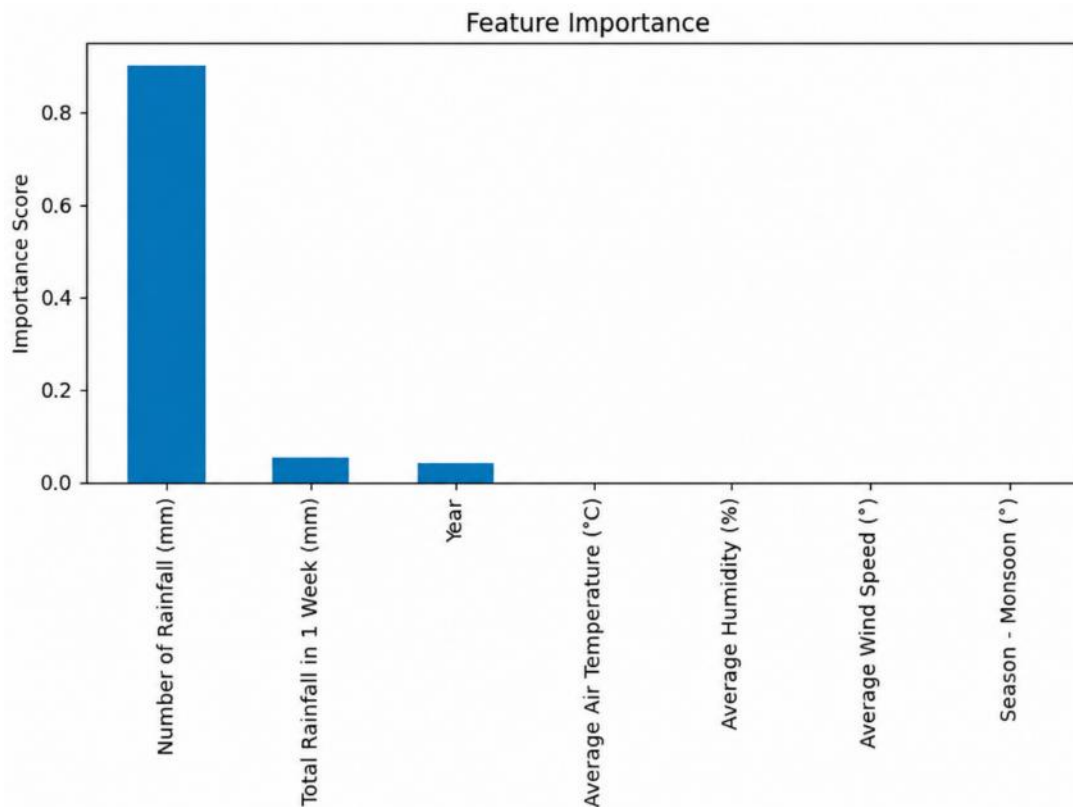


FIGURE 10. Feature Importance Analysis Visualization

Figure 10 shows the contribution level of each weather feature in the decision-making process of the Decision Tree model. From these results, it can be seen that Daily Rainfall is the most dominant feature with an importance score of almost 0.9. This confirms that the intensity of daily rainfall greatly influences flood events, and is the main indicator in model classification. Other features that also contribute, although on a much smaller scale, are Total Weekly Rainfall and Year, which each show an influence on the formation of the tree structure, but not as strong as daily rainfall. Meanwhile, other features such as air temperature, humidity, surface wind direction, and monsoon season have importance scores close to zero. This indicates that the model does not use these variables much in the data separation process, which is likely due to lack of variation or predictive contribution in the dataset used. It can be seen that Feature Importance is aligned with the previous decision tree visualization, where most of the separation is based on rainfall. This confirms that the model is highly dependent on rainfall as the main predictor in detecting potential floods.

3.5 Analysis and Discussion

Given the limited dataset size, the evaluation results of the Decision Tree model in this study showed reasonably good classification performance in distinguishing between flood and non-flood events based on weather-related factors. These findings are consistent with the data exploration results presented in Section 3.2, which showed that daily rainfall and total weekly rainfall are the two variables with significantly different distributions between flood and non-flood conditions. The boxplot

visualization and correlation analysis further support these findings, indicating that rainfall is the most dominant weather factor, while other variables such as air temperature and wind direction show relatively uniform distributions.

The Decision Tree model was trained using the default parameters of the Scikit-learn library (criterion='gini', max_depth=None, random_state=42). This configuration allows the tree to grow fully without pre-pruning, which is acceptable given the relatively small dataset size (50 samples). The model achieved an accuracy of 90% on the test data (10 samples), correctly classifying 9 out of 10 instances.

As shown in the confusion matrix (Figure 7), the model demonstrated perfect precision (1.00) for the flood class, meaning that every flood prediction was correct. However, the recall for the flood class was only 0.67, indicating that the model missed one actual flood event (false negative). In the context of flood prediction, this trade-off is important: prioritizing precision reduces false alarms (false positives) but may fail to detect some actual flood events. For early warning systems, higher recall is generally preferred to ensure that most flood events are detected, even at the cost of generating some false alarms.

The decision tree visualization (Figure 9) shows that the first split is based on the Daily Rainfall feature, with a threshold of ≤ 27 mm. If daily rainfall exceeds 27 mm, the model immediately classifies the event as a flood, with a Gini impurity of 0.0 (a pure node). Conversely, if daily rainfall is ≤ 27 mm, the model continues to evaluate other features such as Year and Total Weekly Rainfall. The relatively shallow depth of the tree indicates that the model does not overfit the training data despite using max_depth = None.

Feature importance analysis (Figure 10) reinforces that Daily Rainfall is the most dominant predictor, with an importance score close to 0.9. Total Weekly Rainfall and Year contribute only marginally, while other variables such as air temperature, humidity, and wind direction have importance scores near zero. This suggests that, for this dataset, rainfall-related features are the primary drivers of flood events, whereas other weather factors have limited predictive value.

It is important to acknowledge the limitations of this study. The dataset consists of only 50 samples, and the model was validated using a single train-test split (80:20) without cross-validation. Therefore, the reported accuracy of 90% may not be fully generalizable to other time periods or locations. The results should therefore be interpreted as a preliminary proof of concept. Future work should involve larger datasets and more robust validation techniques, such as k-fold cross-validation.

4. CONCLUSION

Based on the results and discussion of this research, it can be concluded that the Decision Tree model was successfully applied to analyze the influence of weather factors on flood events in Bandar Lampung City using a dataset of 50 historical samples. The data exploration results showed that daily rainfall and total weekly rainfall are the weather variables that most clearly distinguish between flood and non-flood conditions. These findings are reinforced by the decision tree structure and feature importance analysis, which revealed that the model is highly dependent on rainfall-related variables as the primary predictors.

The Decision Tree model was trained using the default parameters of the Scikit-learn library (criterion = 'gini', max_depth = None) and validated using a single train-test split (80:20). The model achieved a prediction accuracy of 90% on the test data. It demonstrated perfect precision (1.00) for the flood class, indicating that every predicted flood event was correctly classified. However, the recall for the flood class was only 0.67, meaning that one actual flood event was not detected. In the context of flood prediction, this trade-off suggests that the model prioritizes reducing false alarms over detecting all flood events.

Overall, the Decision Tree model proved effective as a preliminary proof of concept for identifying the relationship between weather variables and flood events. However, due to the limited dataset size (50 samples) and the absence of cross-validation, the results should be interpreted cautiously and cannot be fully generalized to other time periods or locations. For future development, it is recommended that this research be expanded using larger and more balanced datasets. Additionally, the application of cross-validation techniques, such as k-fold cross-validation, and ensemble methods such as Random Forest and Gradient Boosting should be considered to improve model robustness and increase sensitivity to flood event detection.

REFERENCES

- [1] D. A. Sari, A. R. Nugroho, and D. Rahmawati, "Analisis kerentanan banjir di kawasan perkotaan dengan pendekatan spasial," *Jurnal Geografi*, vol. 12, no. 2, pp. 99-107, 2020.
- [2] A. Pratama, R. S. Dewi, and H. Syahputra, "Studi curah hujan ekstrem terhadap kejadian banjir di Kota Bandar Lampung," *Jurnal Meteorologi dan Geofisika*, vol. 8, no. 1, pp. 45-54, 2021.
- [3] S. Hadi, D. P. Rini, and D. Kartikasari, "Faktor penyebab banjir dan upaya mitigasi di wilayah perkotaan," *Jurnal Lingkungan dan Bencana Geologi*, vol. 11, no. 1, pp. 12-18, 2020.
- [4] F. R. Putra and E. M. Sari, "Analisis cuaca ekstrem dan dampaknya terhadap banjir di wilayah pesisir," *Jurnal Iklim dan Lingkungan*, vol. 9, no. 2, pp. 88-94, 2021.
- [5] T. Ismail, F. Abdullah, and P. Lestari, "Pengaruh faktor iklim terhadap frekuensi banjir di wilayah tropis," *Jurnal Teknik Sipil dan Lingkungan*, vol. 6, no. 3, pp. 201-209, 2020.
- [6] I. Budi, R. Hidayat, and A. Mulyana, "Pemanfaatan algoritma Decision Tree untuk prediksi banjir berbasis cuaca," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 1, pp. 34-42, 2021.
- [7] S. Wibowo and R. A. Nugroho, "Perbandingan algoritma pohon keputusan dalam prediksi banjir," in *Seminar Nasional Sistem Informasi Indonesia (SESINDO)*, 2021, pp. 112-118.
- [8] A. Navada, A. Ansari, S. Patil, and B. Sonkamble, "Overview of use of decision tree algorithms in machine learning," *International Journal of Computer Applications*, vol. 6, no. 7, pp. 1-6, 2011.
- [9] Y. Song and Y. Lu, "Decision tree methods: Applications for classification and prediction," *Shanghai Archives of Psychiatry*, vol. 27, no. 2, pp. 130-135, 2015.

- [10] F. Fahri and M. A. Ramdhani, "Analisis faktor cuaca terhadap banjir menggunakan data mining," in Prosiding Seminar Nasional Sains dan Teknologi, 2018, pp. 211-218.
- [11] H. Patel and D. Rana, "Survey on decision tree algorithm for classification," International Journal of Engineering Development and Research (IJEDR), vol. 2, no. 1, pp. 1-5, 2014.
- [12] W. Y. Loh, "Fifty years of classification and regression trees," International Statistical Review, vol. 88, no. S1, pp. S35-S63, 2020.
- [13] T. Zhang and H. Yang, "Decision tree applications in weather-based predictions," Journal of Applied Meteorology and Climatology, vol. 60, no. 4, pp. 543-555, 2021.
- [14] B. Iglewicz and D. C. Hoaglin, How to Detect and Handle Outliers. Hoboken: John Wiley & Sons, 2020.
- [15] S. B. Kotsiantis, "Data preprocessing for supervised learning," International Journal of Computer Science, vol. 1, no. 2, pp. 111-117, 2020.
- [16] T. M. Mitchell, Machine Learning. New York: McGraw-Hill, 2021.
- [17] L. Rokach, Data Mining with Decision Trees: Theory and Applications, 3rd ed. Singapore: World Scientific, 2021.
- [18] M. Kuhn and K. Johnson, Applied Predictive Modeling. New York: Springer, 2020.
- [19] Z. H. Zhou, Ensemble Methods: Foundations and Algorithms. Boca Raton: CRC Press, 2022.
- [20] J. Larson, "Understanding overfitting and pruning in decision trees," Journal of Data Science Practice, vol. 3, no. 1, pp. 14-22, 2020.
- [21] M. F. Amin, "Confusion Matrix in Binary Classification Problems: A Step-by-Step Tutorial," Journal of Engineering Research, vol. 6, no. 5, Dec. 2022, doi:10.21608/erjeng.2022.274526.
- [22] D. A. Shofiana, M. Caniadi, R. Sholehurrohman, and Aristoteles, "Decision Tree Algorithms in Water Quality Classification: A Comparative Study of Random Forest, XGBoost, and C5.0," Science and Technology Indonesia, vol. 10, no. 4, pp. 999–1011, Oct 2025. <https://doi.org/10.26554/sti.2025.10.4.999-1011>.
- [23] A. Mosavi, P. Ozturk, and K. W. Chau, "Flood Prediction Using Machine Learning Models: Literature Review," Water, vol. 10, no. 11, p. 1536, Nov. 2018, doi:10.3390/w10111536.