International Journal of Application on Sciences, Technology and Engineering (IJASTE)
Volume 1, Issue 1, 2023. ISSN 2987-2499

# The Data Analysis of Determining Potential Flood-Prone Areas in DKI Jakarta Using Classification Model Approach

Saiful Hadi[1, b)], Devi Fitrianah[2, a)], Vina Ayumi [1, c)], Siew Mooi Lim[3, d)]

[1]*Informatics Engineering Dept. Universitas Mercu Buana, Jakarta, Indonesia 11650*
[2]*Computer Science Dept. School of Computer Science, Bina Nusantara University, Jakarta, Indonesia 11480*
[3]*Faculty of Computing and Information Technology, Tunku Abdul Rahman University College, Kuala Lumpur, Malaysia*

[a)] *Corresponding author: devi.fitrianah@binus.ac.id*
[b)] *saifulhadi.privat@gmail.com*
[c)] *vina.ayumi@mercubuana.ac.id*
[d)] *siewmooi@tar.edu.my*

*Submitted: November-December 2022, Revised: January 2023, Accepted: February 22, 2023*

**Abstract.** The study aims to analyze the flood data in the five Jakarta areas. The focus is to determine areas that are potentially prone to flooding. Currently, there is no identification of potential flood areas in Jakarta based on actual parameters such as total rainfall, altitude, and population density. Based on this problem, a data-driven method is applied using the C4.5 algorithm. Decision trees are used to assist the classification process in determining areas that have the potential to be prone to flooding have potential flooding in the DKI Jakarta area. The 10-fold cross validation is employed along with the confusion matrix to evaluate the model between the actual and predicted results. The study shows that this algorithm can model the Jakarta potential flood-prone areas with an accuracy value of 87.20% with precision and recall values of 90.62% and 94.84%. Based on the model, the predicted flooding area can be identified utilizing the parameters.

## INTRODUCTION

DKI Jakarta is an area that has relatively low terrain. With an altitude of average ± 7 meters above sea level [1], Jakarta has a high level of flood vulnerability every year. Flooding is known as rising water levels, which causes submergence of land. This can occur due to several factors such as nature, humans and the environment [2]. One of the causes of flooding is the high intensity of rainfall. Based on data obtained from the National Disaster Management Agency or *Badan Nasional Penanggulangan Bencana* (BNPB), almost every flood event in DKI Jakarta is caused by high rainfall. Flood disasters can cause severe losses in several urban industrial fields such as business and transportation. This is caused by the rise of water to the surface so that industrial activities are hampered. In addition, flooding can also cause many losses to the community, such as physical and material losses, diseases caused by dirty water, and even deaths. Therefore, to prevent this from happening, modeling is needed to classify areas that have the potential to be flood prone in DKI Jakarta.

In this study, modeling flood-prone areas were carried out using data mining techniques. This technique can process large amounts of data in classifying data based on class categories and generating new data classifications [3]. Research on modeling flood-prone areas was previously carried out by Ahmad Khusaeri *et al.* [4] using the C4.5 algorithm for modeling flood-prone areas in Kabupaten Karawang, Jawa Barat.

In this study, the data used is based on annual reports obtained from several government agencies such as the National Disaster Management Agency or *Badan Nasional Penanggulangan Bencana* (BNPB), the Agency for Meteorology, Climatology and Geophysics or *Badan Meteorologi Klimatologi dan Geofisika* (BMKG), and the Central Statistics Agency or *Badan Pusat Statistik* (BPS) of DKI Jakarta. Then an experiment was conducted using the C4.5 decision tree algorithm to determine potential flood-prone areas in DKI Jakarta.

## Related Work Regarding to Potential Flood-areas

There are several studies related to potential areas prone to flooding. In 2017 [4] used C4.5 algorithm to modeling of flood-prone sites In this study, an accuracy value of 84.385% was produced. These results indicate that the C4.5 algorithm can predict potential areas prone to flooding. Then, a similar study was also conducted by [5]. In contrast to [4] this study analyzes the level of flood vulnerability using Weighted Product method. This study resulted in an accuracy value of 68% in the very high vulnerability class and 80.4% in the high vulnerability class. The classification results are implemented in flood hazard maps using the QGIS application.

Prediction of flood-prone areas was also carried out by developing a flood early warning system and weather information [7]. The flood prediction is determined by using data mining techniques with the C4.5 algorithm. Based on the tests, the use of C4.5 algorithm shows that the highest factor that can cause flooding is rainfall, followed by humidity and temperature.

## Classification Technique

Classification is a data mining technique used to predict an instance of data into a class [8], [9]. Classification is included in supervised learning because the model is based on existing data sets. The Classification is divided into two stages. The first stage is the training stage where the data is analyzed using an algorithm. The second stage is the classification process. Where the classified data is entered into the appropriate class and produces a model [10], [11].

A decision tree is a straightforward classification algorithm that form like a tree structure. The internal node decision tree shows the test of each attribute. The branch formed represents the test result and the leaf node shows the class [12]. Decision trees can be used to retrieve information to help with decision-making systems [13]. To optimize the performance of the classifier, each internal node is divided into two or more parts. So that each path from the root node to the leaf node forms the decision to determine the new class. One of the famous decision tree algorithms at the moment is C4.5 [14].

The C4.5 algorithm is known as a simple decision tree which is the development of the ID3 decision tree algorithm by Quinlan [15], [16] . This algorithm can be used as a method of classification and prediction by forming a decision tree. There are several ways to form a decision tree in the C4.5 algorithm, calculating the entropy value, gain and split from each training data attribute, and then generating a gain ratio [17].

To determine the root node, the attribute is selected based on the largest gain ratio value [18]. The entropy can use the following formula [19]:

$$Entropy(S) = \sum_{i=1}^{n} -pi * \log_2 pi \tag{1}$$

Where:
n : Number of Partition S
pi : Proportion $S_i$ to S

Then, we can use the following formula to calculate the gain ration value:

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy(S_i) \tag{2}$$

Where:
S : Set Case
A : Attribute
n : Number of Attribute Partition A
$|S_i|$ : Number of cases on the i partition
|S| : Number of Cases in S

## Confusion Matrix

Confusion Matrix is a method used in evaluating classification models to predict true or false objects [20]. Evaluation is done to find the accuracy value of the classification model. Confusion Matrix contains columns representing prediction data classes and rows representing original or other data classes [21]. Explanation of the Confusion Matrix table can be seen in Table 1.

**TABLE 1**. Confusion Matrix

| Actual Class | Predicted Class | |
|---|---|---|
| | Yes | No |
| Yes | A (True Positive) | B (False Negative) |
| No | C (False Positive) | D (True Negative) |

True Positive (TP) is the number of positive records in a dataset that are classified as positive. True Negative (TN) is the number of negative records in the data set that are classified as negative. While, False Positive (FP) is the number of negative records in the data set that are classified as positive. False Negative (FN) is the number of positive records in the data set that are classified as negative [22].

# METHODOLOGY

There are several stages including data collection and integration, data pre-processing, algorithm implementation, modeling, and finally evaluation and model validation. Overall, the methodology in this study is shown in Fig. 1 below.

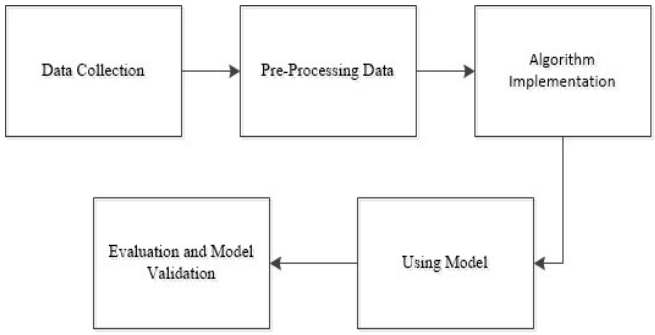


**FIGURE 1.** Research Methodology

## Data Collection and Integration

The data used in this study came from several data sources. First, the flood disaster data was obtained from BNPB. This data contains information on flood events (including impacts). The next data is obtained from BPS DKI Jakarta. This data is an annual report data on population density and altitude. Then the latest data is obtained from BMKG. This data includes the intensity of rainfall per month [23]. These data are data in the DKI Jakarta area within 2 years (2013 to 2014) with a total of 1008 data.

Next is the data integration process by combining some of the data into one data table. The data contains several parameters such as year, month, city, sub-district, population density, altitude, rainfall and flood data as a label. The examples from the dataset used in this study are shown in Table 2.

**TABLE 2.** Data Collection Sample

| Attribute | Description | Data Type | Value |
|---|---|---|---|
| 1 | Year | Int | 2013 |
| 2 | Month | Date | January |

| | | | |
|---|---|---|---|
| 3 | City | Varchar | Jakarta Barat |
| 4 | Sub-District | Varchar | Cengkareng |
| 5 | Population Density | Int | 18550 |
| 6 | Altitude | Real | 7 |
| 7 | Rainfall | Real | 549.4 |
| 8 | Flood | Label | Yes<br>No |

## Pre-processing Data

Pre-processing data is a process that aims to convert raw data into data format so that it can be used to apply data mining techniques and improve data quality [24]. The data obtained in this study still has shortcoming. Therefore, pre-processing is needed to get the best results. There are two types of pre-processing carried out in this study: data cleaning and data transformation.

1. Data Cleaning

   Data cleaning is a process to clean data is done to eliminate items that are not needed [25]. At this stage, data cleaning is done to eliminate the attributes of the year and month, because these attributes are not used in this study. Then, the attributes used are city, sub-district, population density, area height, rainfall, and flooding as labels.
2. Data Transformation

   At this stage, some data is transformed. First, the transformation is done to change the flood data from the information data to the category data that will be labeled. Label determination is done by categorizing areas that have been flooded and those that have not. Then for areas that have ever been flooded, they are included in "YES" category, while for areas with no historical flooding, they are included in "NO" category. The second transformation on population density and rainfall attributes is carried out by converting from integer data to category data. The conversion of values on attributes of population density and rainfall can be seen in Table 3 and Table 4.

**TABLE 3.** Data Transformation of Population Density

| Value | Transformation |
|---|---|
| 64933 — < 31214 | Low Density |
| ≥31214 — < 55935 | Medium Density |
| ≥55935 | High Density |

**TABLE 4.** Data Transformation of Rainfall

| Value | Transformation |
|---|---|
| 0 ─ 100 mm | Low |
| 101 ─ 300 mm | Medium |
| 301 ─ 400 mm | High |
| ≥ 401 mm | Very High |

Source: *Badan Meteorologi Klimatologi dan Geofisika* (BMKG)

## Algorithm Implementation

The implementation of the algorithm is done by calculating the entropy value and information gain of each attribute. The Calculation of entropy values can use formulas (1).

$$Entropy(Total) = \left( -\frac{155}{1008} * \log_2 \left( \frac{155}{1008} \right) \right) + \left( -\frac{853}{1008} * \log_2 \left( \frac{853}{1008} \right) \right) = 0.619194477 \quad (3)$$

Entropy (Total) is obtained by counting the number of 1008 cases with 155 "YES" values and 853 "NO" values. The calculation produces a value of 0.619194477. These results are then used to find the information gain value of each attribute. Information gain calculations can use formulas (2).

$$Gain(Total, curahhujan) = 0.619194477 - \left( \begin{array}{l} \left( \frac{293}{1008} * 0 \right) + \left( \frac{449}{1008} * 0.353924388 \right) \\ + \left( \frac{81}{1008} * 0.785889583 \right) \\ + \left( \frac{185}{1008} * 0.984580124 \right) \end{array} \right) = 0.217690082 \quad (4)$$

The entropy and information gain calculation results can be seen in Table 5.

**TABLE 5.** The Result of Entropy and Information Gain

| Node | Attribute | Value | Total | Yes | No | Entropy | Information Gain |
|---|---|---|---|---|---|---|---|
| 1 | Total | | 1008 | 155 | 853 | 0.619194477 | |
| | Rainfall | | | | | | 0.217690082 |
| | | Low | 293 | 0 | 293 | 0 | |
| | | Medium | 449 | 30 | 419 | 0.353924388 | |
| | | High | 81 | 19 | 62 | 0.785889583 | |
| | | Very High | 185 | 106 | 79 | 0.984580124 | |
| | City | | | | | | 0.044636294 |
| | | Jakarta Pusat | 192 | 18 | 174 | 0.448864489 | |
| **Node** | **Attribute** | **Value** | **Total** | **Yes** | **No** | **Entropy** | **Information Gain** |
| | | Jakarta Utara | 144 | 9 | 135 | 0.337290067 | |
| | | Jakarta Barat | 192 | 14 | 178 | 0.376715003 | |
| | | Jakarta Selatan | 240 | 42 | 198 | 0.669015835 | |
| | | Jakarta Timur | 240 | 72 | 168 | 0.881290899 | |
| | Altitude | | | | | | 0.024737257 |
| | | 2 | 144 | 9 | 135 | 0.337290067 | |
| | | 4 | 192 | 18 | 174 | 0.448864489 | |
| | | 7 | 216 | 35 | 181 | 0.639160599 | |
| | | 8 | 72 | 11 | 61 | 0.61674826 | |
| | | 10 | 144 | 40 | 104 | 0.852405179 | |

| | | | | | | |
|---|---|---|---|---|---|---|
| Population Density | 25.1 | 24 | 2 | 22 | 0.41381685 | 0.000513797 |
| | 26.2 | 216 | 40 | 176 | 0.691289869 | |
| | Low Density | 900 | 141 | 759 | 0.626275189 | |
| | Medium Density | 96 | 12 | 84 | 0.543564443 | |
| | High Density | 12 | 2 | 10 | 0.650022422 | |

Based on the Table V calculation, the highest gain value is obtained in the Rainfall attribute with a value of 0.217690082. From the calculation results, the largest information gain value is then used as the root node in the decision tree. Repeat the calculation until all attributes have classes.

## Using The Model

After all data is collected and processed, the model is tested using RapidMiner. The process of testing the model is shown in Fig. 2, and Fig. 3 below.

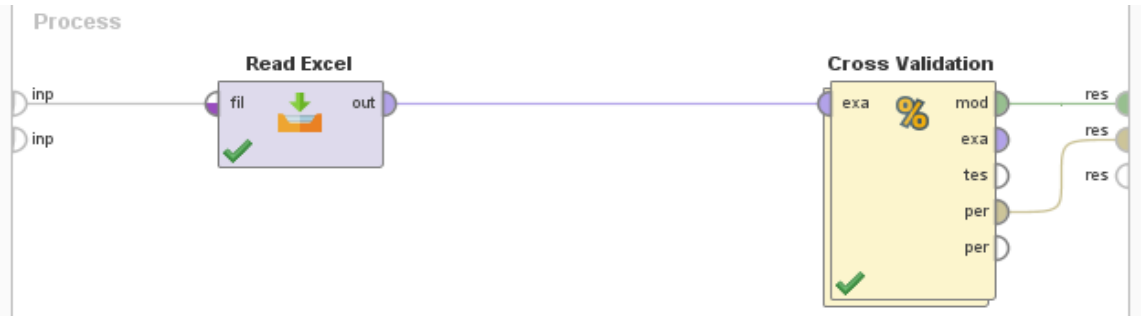


**FIGURE 2.** Operator

The testing process shown in Fig. 2, there are two main stages, Read Excel and Cross Validation. Data used for testing is imported using the Read Excel Operator. After that, the data type is determined on each attribute. The next process is the Cross Validation Operator. The data is processed into two subprocesses in this operator: training suprocess and testing subprocess. The process carried out in Cross Validation Operator is then shown in Fig. 3. Explanation from each operator can be seen as follows:

- Operator Read Excel: The read excel operator is used to import data that will be used for testing from Microsoft Excel into RapidMiner.
- Operator Cross Validation: This operator has two subprocesses: training subprocesses and testing subprocesses. Models are trained using training subprocesses. Then, the trained model is applied to the Testing subprocess. Model performance is measured during the testing phase. Testing is done using the k-fold cross validation method, by dividing the training data as many ask sections, then k-1 part is used as data for system training and the rest is used as test data. In this study, the training data is divided into 10 parts, so we have 10 subset of data to evaluate the performance of the model or algorithm. For each of the 10 subset of data, Cross Validation will use 9 fold for training and 1 fold for testing.

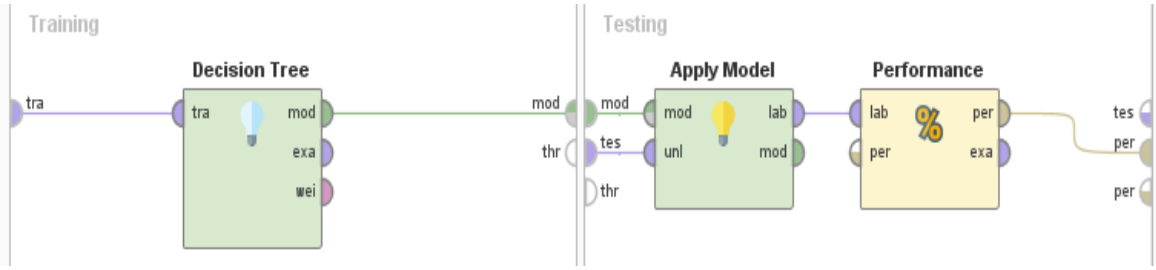


**FIGURE 3.** Operator in Cross Validation

- Operator Decision Tree: This operator is used to produce a decision tree model. In this operator, the results of enteropy and gain calculations will determine criterion values, maximal depth, confidence, minimum gain, minimum leaf size, minimum size for split and number of prepruning alternatives.
- Operator Apply Model: This operator forms model based on training data. The aim is to get predictions on data that is not visible.
- Operator Performance: This operator functions for the evaluation process that can be used on all types of learning task. This operator automatically determines the type of learning tasks and calculates the performance of the criteria.

## Evaluation and Model Validation

Evaluation is needed to analyze and measure the accuracy obtained by using the Confusion Matrix. Calculation of Confusion Matrix values based on True Positive, False Positive, True Negative, and False Negative values. Accuracy is the percentage of records that are correctly classified in testing datasets. Model validation is done by using the k-fold cross validation technique. The dataset is divided into k-sections. A number of k-experiments are carried out. Each experiment uses k-partition data as test data and remaining partitions are used as training data. In this paper, the k-fold cross validation technique is used with a value of k = 10.

## RESULT AND DISCUSSION

Based on the entropy and information gain calculation the results shown in Table V, the rainfall attribute is chosen as the root node of the decision tree. This calculation produces the highest information gain value with a value of 0.217690082. Continue the process until all attributes have a class. The decision tree model formed is shown in Fig. 4.

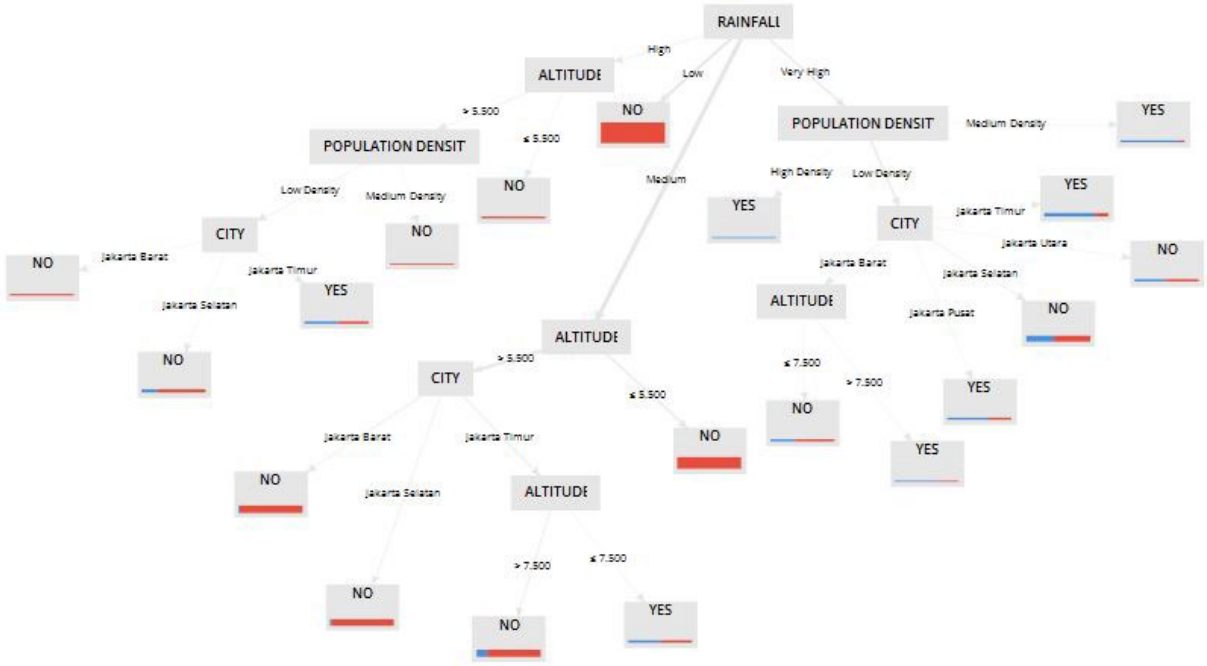


**FIGURE 4.** Decision Tree

## Tree

```
RAINFALL = High
|   ALTITUDE > 5.500
|   |   POPULATION DENSITY = Low Density
|   |   |   CITY = Jakarta Barat: NO {YES=0, NO=9}
|   |   |   CITY = Jakarta Selatan: NO {YES=8, NO=25}
|   |   |   CITY = Jakarta Timur: YES {YES=11, NO=10}
|   |   POPULATION DENSITY = Medium Density: NO {YES=0, NO=4}
|   ALTITUDE ≤ 5.500: NO {YES=0, NO=14}
RAINFALL = Low: NO {YES=0, NO=293}
RAINFALL = Medium
|   ALTITUDE > 5.500
|   |   CITY = Jakarta Barat: NO {YES=2, NO=93}
|   |   CITY = Jakarta Selatan: NO {YES=2, NO=81}
|   |   CITY = Jakarta Timur
|   |   |   ALTITUDE > 7.500: NO {YES=15, NO=76}
|   |   |   ALTITUDE ≤ 7.500: YES {YES=11, NO=11}
|   ALTITUDE ≤ 5.500: NO {YES=0, NO=158}
RAINFALL = Very High
|   POPULATION DENSITY = High Density: YES {YES=2, NO=0}
|   POPULATION DENSITY = Low Density
|   |   CITY = Jakarta Barat
|   |   |   ALTITUDE > 7.500: YES {YES=2, NO=1}
|   |   |   ALTITUDE ≤ 7.500: NO {YES=7, NO=11}
|   |   CITY = Jakarta Pusat: YES {YES=11, NO=6}
|   |   CITY = Jakarta Selatan: NO {YES=32, NO=42}
|   |   CITY = Jakarta Timur: YES {YES=31, NO=8}
|   |   CITY = Jakarta Utara: NO {YES=9, NO=10}
|   POPULATION DENSITY = Medium Density: YES {YES=12, NO=1}
```

**FIGURE 5.** Decision Tree Description

Based on the decision tree model formed at Fig. 4 and Fig. 5, produced rules to predict flood-prone areas. There are 19 rules formed, which can be seen as follows:

1. *If* RAINFALL = High *And* ALTITUDE >5.500, POPULATION DENSITY = Low Density, CITY = Jakarta Barat *Then FLOOD = NO.*

2. *If* RAINFALL = High *And* ALTITUDE >5.500, POPULATION DENSITY = Low Density, CITY = Jakarta Selatan *Then FLOOD = NO.*
3. *If* RAINFALL = High *And* ALTITUDE >5.500, POPULATION DENSITY = Low Density, CITY = Jakarta Timur *Then FLOOD = YES.*
4. *If* RAINFALL = High *And* ALTITUDE >5.500, POPULATION DENSITY = Medium Density *Then FLOOD = NO.*
5. *If* RAINFALL = High *And* ALTITUDE ≤5.500 *Then FLOOD = NO.*
6. *If* RAINFALL = Low *Then* FLOOD = NO.
7. *If* RAINFALL = Medium *And* ALTITUDE > 5.500, CITY = Jakarta Barat *Then FLOOD = NO.*
8. *If* RAINFALL = Medium *And* ALTITUDE > 5.500, CITY = Jakarta Selatan *Then FLOOD = NO.*
9. *If* RAINFALL = Medium *And* ALTITUDE > 5.500, CITY = Jakarta Timur, ALTITUDE > 7.500 *Then FLOOD = NO.*
10. *If* RAINFALL = Medium *And* ALTITUDE > 5.500, CITY = Jakarta Timur, ALTITUDE ≤ 7.500 *Then FLOOD = YES.*
11. *If* RAINFALL = Medium *And* ALTITUDE ≤ 5.500 *Then FLOOD = NO.*
12. *If* RAINFALL = Very High *And* POPULATION DENSITY = High Density *Then* FLOOD = YES.
13. *If* RAINFALL = Very High *And* POPULATION DENSITY = Low Density, CITY = Jakarta Barat, ALTITUDE ≤ 7.500 *Then* FLOOD = NO.
14. *If* RAINFALL = Very High *And* POPULATION DENSITY = Low Density, CITY = Jakarta Barat, ALTITUDE > 7.500 *Then* FLOOD = YES.
15. *If* RAINFALL = Very High *And* POPULATION DENSITY = Low Density, CITY = Jakarta Pusat *Then* FLOOD = YES.
16. *If* RAINFALL = Very High *And* POPULATION DENSITY = Low Density, CITY = Jakarta Selatan *Then* FLOOD = NO.
17. *If* RAINFALL = Very High *And* POPULATION DENSITY = Low Density, CITY = Jakarta Utara *Then* FLOOD = NO.
18. *If* RAINFALL = Very High *And* POPULATION DENSITY = Low Density, CITY = Jakarta Timur *Then* FLOOD = YES.
19. *If* RAINFALL = Very High *And* POPULATION DENSITY = Medium Density *Then* FLOOD = YES.

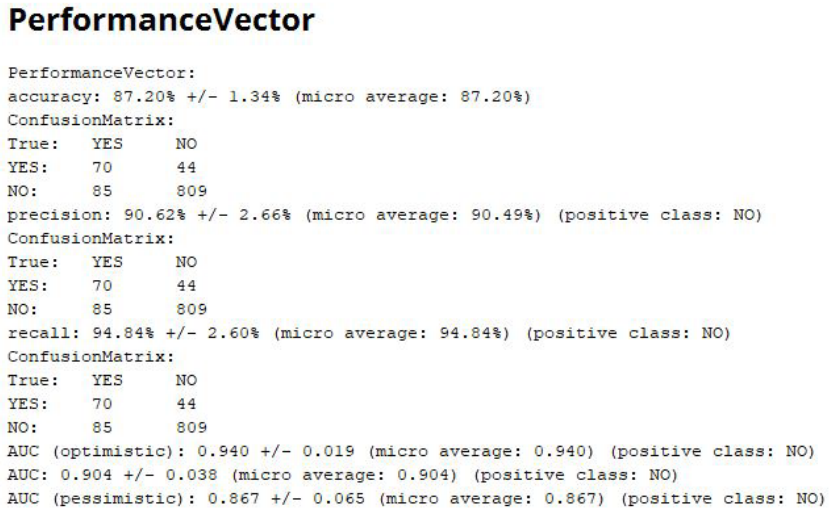


**FIGURE 6.** PerformanceVector Description

After testing using Cross Validation, the test results are evaluated manually by calculating the Confusion Matrix value. The results of the Confusion Matrix calculation can be seen in Table 6.

**TABLE 6.** Confusion Matrix

| | True YES | True NO |
|---|---|---|
| Pred. YES | 70 | 44 |
| Pred. NO | 85 | 809 |

$$Accuracy = \left( \left( \frac{70+809}{1008} \right) * 100\% \right) = 87,20\% \quad (6)$$

$$\Pr ecision = \left( \left( \frac{809}{809+85} \right) * 100\% \right) = 90,62\% \quad (7)$$

$$\mathrm{Re}\,call = \left( \left( \frac{809}{809+44} \right) * 100\% \right) = 94,84\% \quad (8)$$

The results of the Confusion Matrix calculations that are done manually produce the same accuracy, precision, and recall values as shown in Fig. 6. With an accuracy value of 87.20%, precision 90.62%, and recall of 94.8%, it shows that the C4.5 algorithm can be used in predicting and determining potential flood-prone areas in DKI Jakarta.

## CONCLUSION

The results of research conducted on 1008 total data, it shows that the C4.5 algorithm can be used to determine potential flood-prone areas. This research begins with pre-processing the original data. There are two types of pre-processing that have been done, data cleaning and data transformation. First, the data is constructed by removing a number of unnecessary attributes and performing several data transformations to convert and define labels.

The results of this study are then manually evaluated using Confusion Matrix. The results of that process produce an accuracy value of 87.20% with a value of precision 90.62% and recall 94.84%. Based on these results, it can be concluded that the C4.5 algorithm can be implemented to predict and determine potential flood-prone areas in DKI Jakarta, so that future floods can be prevented. With this research, it is expected that further studies of potential flood-prone areas with more parameters to produce better accuracy values.

## REFERENCES

1. BPS DKI Jakarta, "Jakarta Dalam Angka 2018," DKI Jakarta, Accessed on https://jakarta.bps.go.id/publication/2018/08/16/67d90391b7996f51d1c625c4/provinsi-dki-jakarta-dalam-angka-2018.html at Sep 2019.
2. H. R. Fazeli, "A Study of Volunteered Geographic Information (VGI) Assessment Methods For Flood Hazard Mapping: A Review," *J. Teknol.*, vol. 10, pp. 127–134, 2015.
3. S. S. Nikam, "A Comparative Study of Classification Techniques in Data Mining Algorithms," *Int. J. Mod. Trends Eng. Res.*, vol. 4, no. 7, pp. 58–63, 2017.
4. A. Khusaeri, S. Ilham, D. Nurhasanah, D. Delpidat, and B. N. Sari, "Algoritma c4.5 untuk pemodelan daerah rawan banjir studi kasus kabupaten karawang jawa barat," vol. 9, pp. 132–136, 2017.
5. E. Darwiyanto, "Aplikasi GIS Klasifikasi Tingkat Kerawanan Banjir Wilayah Kabupaten Bandung Menggunakan Metode Weighted Product," *Indones. J. Comput.*, vol. 2, no. 1, p. 59, 2017.
6. R. K. Abdullah and E. Utami, "Studi Komparasi Metode SVM dan Naive Bayes pada Data Bencana Banjir di Indonesia pembaca ataupun peneliti bisa melihat pola yang tersembunyi di."
7. R. Putra, . Z., E. Madona, and A. Nasution, "Desain dan Implementasi Peringatan Dini Banjir Menggunakan Data Mining dengan Wireless Sensor Network," *J. Nas. Tek. Elektro*, vol. 5, no. 2, p. 181, 2018.
8. T. N. Phyu, "Survey of classification techniques in data mining," *Proc. Int. MultiConference Eng. Comput. Sci.*, vol. 1, pp. 18–20, 2009.
9. M. Sadikin and F. Alfiandi, "Comparative study of classification method on customer candidate data to predict its potential risk," *Int. J. Electr. Comput. Eng.*, vol. 8, no. 6, pp. 4763–4771, 2018.

10. D. Fitrianah, N. H. Praptono, A. N. Hidayanto, and A. M. Arymurthy, "Feature Exploration for Prediction of Potential Tuna Fishing Zones," vol. 5, no. 4, pp. 270–274, 2015.
11. K. Adhatrao, A. Gaykar, A. Dhawan, R. Jha, and V. Honrao, "Predicting Students' Performance Using Id3 And C4.5 Classification Algorithms," *Int. J. Data Min. Knowl. Manag. Process*, vol. 3, no. 5, pp. 39–52, 2013.
12. R. D. H. Devi and M. I. Devi, "Outlier detection algorithm combined with decision tree classifier for early diagnosis of breast cancer," *Int. J. Adv. Eng. Technol.*, vol. 7, no. 2, pp. 93–98, 2016.
13. S. K. Yadav, "Data Mining : A Prediction for Performance Improvement of Engineering Students using Classification," *World Comput. Sci. Inf. Technol. J. (WCSIT*, vol. 2, no. 2, pp. 51–56, 2012.
14. W. Dai and W. Ji, "A mapreduce implementation of C4.5 decision tree algorithm," *Int. J. Database Theory Appl.*, vol. 7, no. 1, pp. 49–60, 2014.
15. S. Sathyadevan and R. R. Nair, "Comparative Analysis of Decision Tree Algorithms: ID3, C4.5 and Random Forest," vol. 711, pp. 549–562, 2019.
16. Z. Masetic, A. Subasi, and J. Azemovic, "Malicious Web Sites Detection using C4.5 Decision Tree," vol. 5, no. 1, 2016.
17. M. Mirqotussa'adah, B. Prasetiyo, S. Alimah, E. Sugiharti, and M. A. Muslim, "Penerapan Dizcretization dan Teknik Bagging Untuk Meningkatkan Akurasi Klasifikasi Berbasis Ensemble pada Algoritma C4.5 dalam Mendiagnosa Diabetes," *Lontar Komput. J. Ilm. Teknol. Inf.*, vol. 8, no. 2, p. 135, 2017.
18. Ihsan and R. Wajhillah, "Penerapan Algoritma C4.5 Terhadap Diagnosa Penyakit Demam Tifoid Berbasis Mobile," *J. Swabumi AMIK BSI Sukabumi*, vol. III, no. 1, pp. 50–58, 2015.
19. E. Buulolo, N. Silalahi, Fadlina, and R. Rahim, "C4.5 Algorithm To Predict the Impact of the Earthquake," *Int. J. Eng. Res. Technol.*, vol. 6, no. 2, 2017.
20. M. T. Firmansyah and Rusito, "Implementasi Metode Decision Tree Dan Algoritma C4.5 Untuk Klasifikasi Data Nasabah Bank," *Infokam*, vol. XII, pp. 1–12, 2016.
21. J. A. Suyatno, F. Nhita, and A. A. Rohmawati, "Rainfall forecasting in Bandung regency using C4.5 algorithm," *2018 6th Int. Conf. Inf. Commun. Technol. ICoICT 2018*, vol. 0, no. c, pp. 324–328, 2018.
22. D. T. Wahyuni and A. Luthfiarta, "Prediksi Hasil Pemilu Legislatif DKI Jakarta Menggunakan Naive Bayes Dengan Algoritma Genetika Sebagai Fitur Seleksi," *Jur. Tek. Inform. FIK Udinus*, 2014.
23. Badan Meteorologi Klimatologi dan Geofisika, "Data Curah Hujan DKI Jakarta," 2019. Accessed on https://reactora.net/data-curah-hujan-dki-jakarta/ at Sep 2019.
24. K. Rajesh and S. Anand, "Analysis of SEER Dataset for Breast Cancer Diagnosis using C4 . 5 Classification Algorithm," vol. 1, no. 2, pp. 72–77, 2012.
25. L. S. Malang *et al.*, "Penerapan Metode Naive Bayes dalam Pengklasifikasi Trafik Jaringan," *Smatik J.*, vol. Vol 06, no. January 2016, pp. 26–36, 2017.