INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

# Enhancing Weather Prediction Models through the Application of Random Forest Method and Chi-Square Feature Selection

Helena Nurramdhani Irmanda [a,*], Ermatita [b], Mohd. Khalid Awang [c], Muhammad Adrezo [a]

[a] *Faculty of Computer Science, Universitas Pembangunan Nasional Veteran Jakarta, Cilandak, Jakarta Selatan, Indonesia*
[b] *Faculty of Computer Science, Universitas Sriwijaya, Indralaya, Ogan Ilir, Indonesia*
[c] *Fakulti Informatik dan Komputeran Universitas Sultan Zainal Abidin, Besut, Terengganu, Malaysia*
*Corresponding author: *helenairmanda@upnvj.ac.id*

***Abstract*—This study discovers weather forecast methodologies, concentrating mainly on the climatic issues faced by Indramayu Regency and its considerable impact on agriculture, specifically rice production and national food security. The study emphasizes the crucial need for accurate weather forecasting, especially in the context of ongoing climate change, by highlighting the region's vulnerability to weather anomalies and their possible disruption of crop output. To solve these issues, the study investigates machine learning techniques, particularly ensemble learning methods such as Random Forest in conjunction with Chi-Square feature selection. The article thoroughly outlines the research approach, including data collection from Indonesia's Meteorology, Climatology, and Geophysics Agency (BMKG), data pre-processing, feature selection processes, and data splitting. Notably, the methodology integrates the Synthetic Minority Over-sampling Technique (SMOTE) to adjust imbalanced data and uses key weather attributes for model construction (humidity, wind speed, and direction). The resulting Random Forest model performs well, with an accuracy rate of 87.6% in forecasting different types of rainfall. However, the study indicates potential overfitting in some rainfall classes, implying the need for additional data augmentation or modeling technique refining. In conclusion, this study demonstrates the potential efficacy of ensemble learning techniques in weather prediction, focusing on the Indramayu Regency. It emphasizes the need for exact forecasts in the agricultural and fisheries industries and suggests possibilities for additional investigation, such as research into alternative prediction approaches such as deep learning.**

***Keywords*—Ensemble learning; random forest; prediction; weather.**

*Manuscript received 16 Nov. 2023; revised 19 Jan. 2024; accepted 1 Mar. 2024. Date of publication 30 Nov. 2024.*


## I. INTRODUCTION

Weather is the condition or state of the atmosphere that occurs in a specific area or region during a particular period. Weather varies in different places and can change within minutes, hours, days, or weeks. The science that studies weather is called meteorology. Meteorology is defined as the science that studies the physical and chemical characteristics of the atmosphere or forecasts weather conditions [1]. The Meteorology, Climatology, and Geophysics Agency (BMKG) issues weather predictions in Indonesia. A weather forecaster, supported by sophisticated tools, makes forecasts. Weather forecasting is essential as it affects various aspects of life, such as agriculture, aviation, livestock, etc.

The Indramayu Regency is situated along the northern coastline of Java Island and comprises ten sub-districts with 35 villages directly bordering the sea. Its location along the northern coastal area of Java results in relatively high air temperatures in this regency, ranging from 18°C to 28°C Celsius. Meanwhile, the average rainfall is approximately 61.06 mm [2]. With these characteristics, Indramayu Regency has been designated as one of the agricultural centers, particularly for national rice production, and a fishing center. However, given these circumstances, Indramayu Regency faces significant challenges in maintaining agricultural and fisheries productivity in the future, especially since Indramayu has become one of the hottest regencies in West Java in 2023. Challenges such as climate change have a profound impact on weather anomalies. Weather anomalies in the Indramayu Regency area significantly threaten rice production [3]. If this happens, it could disrupt the national food supply. Due to these weather anomalies, the agricultural sector is particularly vulnerable and experiences significant losses. The threat of weather anomalies, such as the onset of

a dry season, remains highly probable. If not addressed promptly, the risk of crop failure could recur in Indramayu.

Therefore, weather prediction is needed as an anticipatory measure to minimize the impacts that may result from climate-induced weather anomalies [4]. Many primary sectors, such as agriculture, rely on weather conditions for production. The ongoing climate change makes traditional weather prediction methods less effective. A better, more reliable weather prediction method is required to overcome these difficulties. This prediction system significantly influences a country's economy and people's lives. Data analysis and machine learning algorithms are employed to predict weather conditions. Some studies have used machine learning approaches, such as weather prediction using the artificial neural network (ANN) method [5].

This investigation involved the collection of several meteorological characteristics from the National Climate Data Centre (NCDC) India from November 2007 to October 2017. Then, one of the techniques in ANN, namely Long Short-Term Memory (LSTM), was used. LSTM was employed to predict future weather conditions. The neural network was trained using a combination of various weather parameters, including temperature, rainfall, wind speed, pressure, dew point visibility, and humidity. This research resulted in an accuracy of 84% in weather prediction. Ensemble learning is another machine learning approach that can be used for weather prediction. Among various forecasting parameter algorithms. The utilization of ensemble learning techniques has been found to yield straightforward models that exhibit high predictive accuracy across a range of problem domains [6].

One type of ensemble learning method is random forest. In the study[7], it was found that random forest provides deep insights into weather data dependencies. An accuracy of 87.90% can be achieved with this machine-learning model. This is in line with the study [8], which concluded that compared to other algorithms, the ensemble learning algorithm, namely Random Forest, achieves the best performance accuracy of 89%. To make the ensemble learning algorithm faster and more effective, a feature selection method is needed to produce accurate predictions. Feature selection is a process of dimensionality reduction. Feature selection aims to ascertain the significance of features within a dataset while eliminating unnecessary and duplicated features. Feature selection is a technique that decreases the dimensionality of data, resulting in improved operational efficiency and accelerated performance of data mining algorithms [9]. One suitable feature selection technique combined with Random Forest is Chi-Square. Previous research on breast cancer prediction using the Random Forest Classifier showed that the Chi-Square feature selection outperforms the Gail model [10], This finding is further supported by research indicating that the combination of Random Forest and Chi-Square is the best pair among all for building credit assessment models [11]. Chi-square feature selection employs statistical theory to test the independence of an attribute with its category. One of the objectives of using feature selection is to eliminate unsuitable features in classification tasks [12].

Based on the background, this study will apply the ensemble learning method with Random Forest to predict the weather, especially in the Indramayu Regency. As an innovation, this research will also implement Chi-Square feature selection to choose the best features that significantly influence the weather classification model. Notably, in previous weather prediction studies, none have explored feature selection using Chi-Square. The research findings can be implemented in the Indramayu Regency, particularly in the agricultural and fisheries sectors. The implications are that the insights and research outcomes can be adopted in the planning process in Indramayu Regency, leading to the region's transformation into a smart village.

## II. Material and Method

The materials and methods section will explain the research stages and theories used in this research. The research stages are depicted in Figure 1.

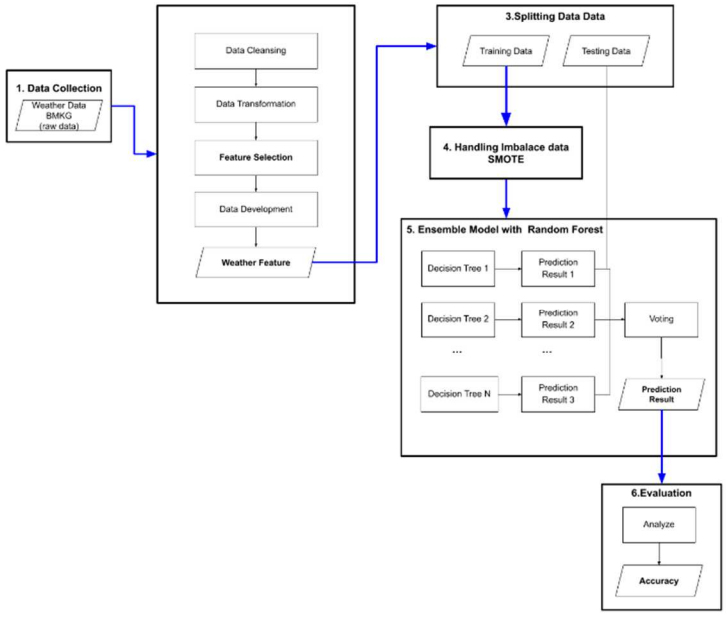


Fig. 1 Block diagram of research stages

### A. Data Collection

The dataset used as input for this system consists of weather data sourced from BMKG [13]. The attributes used include temperature, air humidity, air pressure, and wind speed. The data is planned to be collected from January 1, 2017, to February 28, 2022. There are 10 attributes of weather data, including minimum temperature, maximum temperature, average temperature, average humidity, rainfall, sunshine duration, maximum wind speed, wind direction, average wind speed, and the most frequently occurring wind direction. The initial raw data collected consists of 1825 rows of data. The target class for this data is rainfall, where the rainfall values will be categorized according to six categories: Cloudy, Light Rain, Moderate Rain, Heavy Rain, Very Heavy Rain, and Extreme Rain. Categorization will be obtained from data transformation results in the preprocessing stage, where the rules are received from the BMKG. From the data collection process, weather data were obtained, which will undergo preprocessing to be suitable for use.

### B. Preprocessing

After the data is collected, the preprocessing stage is conducted to prepare it for further processing. The preprocessing stage consists of data cleaning, data transformation, data development, and feature selection [14]. Data cleansing involves cleaning the data by addressing irrelevant and missing parts. Some steps in data cleaning include handling missing values and noise [15]. Missing values are situations where data is missing or incomplete in the database. Filling missing values can be done in several ways, such as manually filling in the missing values with the mean or other values based on the data type, removal of missing data [16]. Noise refers to useless data that tools cannot interpret. Noise may occur due to incorrect data collection, inaccurate data entries, etc. Noise in this weather data can be detected if rows with values of 8888 indicate that the data is not measurable. In this study, missing values and noise will be deleted because their presence can affect the integrity and accuracy of data analysis. The process of eliminating missing values is carried out to ensure the consistency of the dataset and the results produced from later analysis

Data transformation is the process of converting existing data into labeled data. As explained in the data collection, the numeric values of rainfall, initially measured in millimeters, will be transformed into six categories based on BMKG rules: Cloudy, Light Rain, Moderate Rain, Heavy Rain, Very Heavy Rain, and Extreme Rain. The threshold values utilized for determining the magnitude of rainfall are as follows:

a. The precipitation rate is recorded as 0 mm per day, indicating a partly cloudy weather condition, as indicated by the color gray.
b. The precipitation rate of light rain falls within the range of 0.5 to 20 millimeters per day. as green indicates.
c. Moderate rainfall is characterized by precipitation rates ranging from 20 to 50 mm per day, as indicated by yellow.
d. The precipitation rate of 50 - 100 mm per day falls into the category of heavy rain, as shown by the color orange.
e. The precipitation rate of 100 - 150 mm per day, shown in red, denotes a classification of weighty rainfall.
f. The category of extreme rainfall is defined as precipitation over 150 mm per day, as indicated by the purple color on the scale.

The next stage is feature selection, which determines which features significantly impact. These selected features are retained, and other features may be reduced. Data development aims to ensure data completeness and validity, determine data formats, and build clean and ready-to-use data that can be stored and easily retrieved when needed. Feature selection in this research uses chi-square.

The Chi-Square feature evaluation explains the importance of each original feature. Users can keep the majority of them while discarding the less important ones [17]. The Chi-Squared test statistic is employed in Chi-Squared feature selection to assess the significance of features concerning the target class. The computation of the Chi-Square statistic involves the utilization of Equation (1), wherein the observed value represents the count of actual class observations, and the predicted value denotes the anticipated count of class observations under the assumption that there is no association between the feature and the class. The Chi-Square test necessitates discretizing numerical features before computation, resulting in the summation being performed across all feature values. [12].

$$X^2 = \sum(observed - expected)^2/(expected) \quad (1)$$

A strong Chi-Square test result suggests that the feature and the target class are closely related, and therefore, we should use that feature in the dataset. Out of the ten attributes of weather data, including minimum temperature, maximum temperature, average temperature, average humidity, rainfall, duration of sunshine, maximum wind speed, wind direction, average wind speed, and the most frequently occurring wind direction, the selection of the most influential attributes will be conducted. Numerous essential steps in the feature selection process utilize the chi-square test. First, a contingency table reflects the relationship between the evaluated categorical variables. The chi-square value is then calculated, which measures how much the observed frequency distribution differs from what would be predicted if the variables were unrelated. The degrees of freedom are then determined based on the number of categories in the variable. Next, by comparing the chi-square value to the crucial value at a given significance level, it is possible to determine whether there is a meaningful link between variables. Variables with a considerable impact can be chosen as relevant features for further research. As a result, this technique aids in identifying the most informative characteristics within the context of categorical variable connections.

### C. Data Splitting

After preprocessing, the obtained dataset is divided into training and testing data. The training and testing data are selected randomly using resampling. The combination of the training and testing data used in this study is 70:30, respectively [18]. Regarding imbalanced data issues, one of the strategies that can be applied in sampling techniques includes both oversampling and undersampling [19]. SMOTE

(Synthetic Minority Oversampling Technique) is one of the sampling techniques that can improve the classifier's accuracy for the minority class [20]. This is because SMOTE can present samples from the minority class to be learned during the learning process, allowing the learning model to create a broader coverage of the classification model area.

### *D. Modeling with Random Forest*

Ensemble learning is a broad name for systems that mix various strategies to create judgments, most commonly in guided machine learning tasks such as classification. It is an algorithm that, given a labeled dataset, generates a model that generalizes the data [21]. Predictions for additional unlabeled data may be produced using the developed model. Ensemble learning allows for the integration of diverse machine learning algorithms, such as decision trees, neural networks, and linear regression models, among others [22]. The fundamental tenet of ensemble learning is that the amalgamation of several models allows for the potential compensation of errors made by individual learners. Consequently, the ensemble's overall predictive performance surpasses that of a solitary learner. [23].

The fundamental concept of the ensemble classification model may be delineated into two distinct stages: (1) the acquisition of classification outcomes using various weak classifiers and (2) the amalgamation of these diverse outcomes into a consistent function, ultimately culminating in the result through a voting mechanism. Bagging, AdaBoost, random forest, random subspace, and gradient boosting are often employed in ensemble classification methods in several domains [24].

The concept of random forest emerged in the mid-1990s and gained significant recognition through a publication in the early 2000s, which has since become highly influential, accumulating around 30,000 citations by mid-2017 [25]. The popularity of random forests is increasing due to their simplicity and anticipated efficacy. In addition, the random forest algorithm is commonly recognized as a relatively straightforward strategy to adjust compared to alternative methods. The system utilizes a substantial quantity of unpruned autonomous decision trees [26].

Figure 2 and Figure 3 explain the formation of individual trees in a random forest. The essence of random forest is to build multiple decision tree classifiers and then make decisions based on the voting of the decision trees created. The steps involved in creating a Random Forest include [27]:

a. Selecting a subset of N data from the training set.
b. Building a Decision Tree from the selected N data.
c. Choosing the number of N-trees (a collection of trees) to be created. Next, repeat Steps one and two (building decision trees) multiple times, such as 200 times, 300 times, etc.
d. Each N-tree predicts the group of new datasets. Then, the new dataset is assigned to the group or label with the highest probability among all the N-tree combinations.

**Algorithm** : The Random Forest algorithm
**Input**: $IDT$(a decision tree inducer),$T$(the number of iterations), $S$(training set),
$\mu$(the subsample size), $N$(number of attributes used in each node)
**Output**: $M_t$:$\forall t = 1, ..., T$
**for** *each* $t$ *in* $1, ..., T$ **do**
$S_t \leftarrow$ Sample $\mu$ instances from $S$ with replacement.
Build classifier $M_t$ using $IDT(N)$ on $S_t$
$t++$
**end**

Fig. 1 Random Forest algorithm

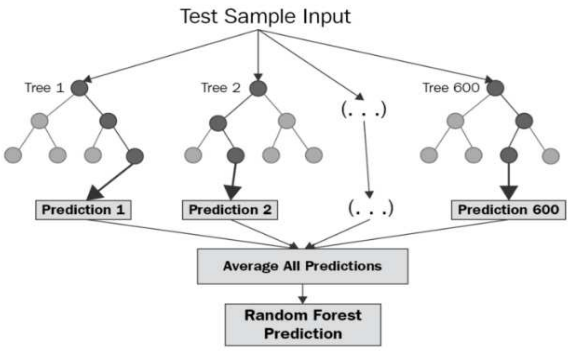


Fig. 2 Random Forest classifier illustration

Moreover, the Random Forest algorithm can handle both regression and classification tasks. It can also discern and select the most significant characteristics from the provided training dataset. [28]. Weather data modeling with random forest follows the following steps [29] :

a. Taking n bootstrap samples with replacement from the training data.
b. Determining the number of predictor variables that will be randomly selected while determining the split when constructing the classification tree.
c. Constructing the classification tree where the selection of the best node is based on randomly chosen predictor variables.
d. Perform classification prediction for the training data.
e. Repeat steps b to d for N replications.
f. Perform majority voting for the classification prediction results from N replications.
g. Form classification trees.
h. Calculate the classification accuracy of the training data.
i. Calculate the classification accuracy of the testing data.
j. Repeat steps b to h while trying different combinations of the number of trees (K), namely 100, 300, and 500.
k. Select the combination of the number of trees with the highest accuracy.

### *E. Evaluation*

The prediction results are analyzed to obtain accuracy values that determine whether the prediction model created is suitable for use [30]. Accuracy is a value obtained by dividing the total number of correctly predicted test data, which includes true positive (TP) and true negative (TN), by the total number of test data [31].

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP} \; x \; 100\% \qquad (2)$$

According to formula 2, TP (True Positive) refers to instances belonging to the positive class that are accurately classified as positive by the constructed classification model. A True Negative (TN) refers to cases where the negative class is accurately classified as negative. A false positive (FP) is a situation in which the negative class is erroneously classified as positive by the constructed classification model. A false negative (FN) is a situation in which a positive class is erroneously classified as negative.

In addition to assessing the prediction model's accuracy, precision and recall are computed using the following formulas.

$$Precision = \frac{TP}{TP+FP}\ x\ 100\% \quad (3)$$

$$Recall = \frac{TP}{TP+FN}\ x\ 100\% \quad (4)$$

$$Precision = \frac{TN}{TN+FP}\ x\ 100\% \quad (5)$$

The precision metric is determined by the ratio of true positive predictions to the overall number of positive predictions. The calculation of the ratio between the number of true positive predictions and the total number of actual positive instances is commonly performed using recall or sensitivity. Meanwhile, specificity can be defined as the proportion of true negative predictions to the total number of negative data points.

## III. Results and Discussion

### A. Data Collection

The data used in this study is weather data taken from the online data of BMKG (https://dataonline.bmkg.go.id/home) from the monitoring station in Kertajati, Majalengka, West Java [13]. This was done because there is no transmitter station in Indramayu. Kertajati was chosen because it is close to Indramayu. The data was collected from the years 2017 to 2022. The weather data on the BMKG website can only be downloaded once a month in XLS format, which Python can easily read. Therefore, in the data collection stage, the weather data was downloaded each month from 2017 to 2022 and combined into one XLS file. Table 1 shows an example of the weather data from the BMKG online data site.

TABLE I
EXAMPLE WEATHER DATA FROM BMKG

| Date | Tn | Tx | Tavg | … | ff_avg | ddd_car |
|---|---|---|---|---|---|---|
| 01-01-2019 | 24 | 32 | 27.3 | … | 1 | C |
| 02-01-2019 | 23.5 | 33.6 | 26.5 | … | 1 | C |
| 03-01-2019 | 23.6 | 32.8 | 26.7 | … | 1 | C |
| 04-01-2019 | 22.2 | 32.8 | 27.1 | … | 1 | C |
| 05-01-2019 | 23 | 33.2 | 27.8 | … | 1 | C |
| 06-01-2019 | 23.8 | 34 | 28.4 | … | 1 | C |
| 07-01-2019 | 24.6 | 34.3 | 28.2 | … | 1 | C |
| 08-01-2019 | 24.3 | 33 | 27.2 | … | 1 | C |
| 09-01-2019 | 24 | 33.4 | 26.8 | … | 1 | C |
| 10-01-2019 | 23.6 | 33.8 | 28.2 | … | 1 | C |

Based on Table 1, there are 10 attributes of weather data with the following descriptions:

a. The variable Tn represents the minimum temperature in degrees Celsius (°C).,
b. The variable Tx represents the maximum temperature in Celsius (°C).
c. The variable Tavg represents the average temperature in degrees Celsius (°C).
d. The variable ff_avg represents the average wind speed in metres per second (m/s).
e. The variable ddd_car represents the most frequently occurring wind direction in.
f. The RH_avg represents the average humidity in percentage.

The symbol "…" contains the following values, among others:

a. RR refers to the measurement of rainfall in millimetres (mm).
a. The variable ss represents the duration of sunshine, measured in hours.
b. The variable ff_x represents the maximum wind speed in metres per second.
c. The variable ddd_x represents the wind direction at its highest speed, measured in degrees.

Weather data were obtained from the data collection process, which will undergo preprocessing to be suitable for use.

### B. Preprocessing

*1) Data Cleansing:* The meteorological data acquired from the online BMKG website continues to exhibit instances of missing numbers. In order to address missing values in the dataset, it is necessary to eliminate rows containing unmeasured values. For instance, in the case of rainfall (RR), rows with values of 8888 or blank entries are excluded, as these indicate either the absence of measurement or non-availability of rainfall data for the corresponding day. From this data cleansing process, a total of 1698 rows of clean weather data are obtained.

*2) Data transformation:* In this stage, the rainfall data is converted into five categories: Clear/Partly Cloudy, Light Rain, Moderate Rain, Heavy Rain, Very Heavy Rain, and Extreme Rain. These six categories are grouped according to the BMKG's guidelines on the official BMKG website at https://www.bmkg.go.id/cuaca/probabilistik-curah-hujan.bmkg [32]. In Table 2, the results of rainfall categorization are presented.

TABLE II
EXAMPLE OF TRANSFORMED WEATHER DATA

| Date | Tn | Tx | Tavg | RH_avg | … | RR | Rainfall |
|---|---|---|---|---|---|---|---|
| 01-12-2017 | 24 | 30 | 27.1 | 85 | … | 4.8 | Light Rain |
| 03-12-2017 | 23 | 34 | 28 | 80 | … | 0 | Cloudy |
| 04-12-2017 | 25 | 35 | 27.2 | 83 | … | 0 | Cloudy |
| 05-12-2017 | 24 | 33 | 28 | 82 | … | 49.5 | Moderate rain |
| 06-12-2017 | 25 | 35 | 28 | 81 | … | 0 | Cloudy |
| 08-12-2017 | 24 | 33 | 27.4 | 87 | … | 87.1 | Heavy Rain |
| 09-12-2017 | 24 | 34 | 27.2 | 86 | … | 15.1 | Light Rain |
| 10-12-2017 | 24 | 33 | 27.1 | 87 | … | 4.2 | Light Rain |

Based on Table 2, examples of minimum temperature (Tn), maximum temperature (Tx), average temperature (Tavg), average humidity (RH_avg), Rainfall (RR) are presented. The "…" symbol contains sample values for the duration of sunshine (ss), maximum wind speed (ff_x), and wind direction (ddd_x).

The next step in data transformation is to perform encoding for rainfall. Label encoding converts each value in the column into sequential numbers. One of the Python libraries used for encoding is the label encoder. Rainfall will be encoded as 0, 1, 2, 3, 4, or 5.

```
def LABEL_ENCODING(c1):
    from sklearn import preprocessing
    label_encoder = preprocessing.LabelEncoder()
    data[c1]= label_encoder.fit_transform(data[c1])
    data[c1].unique()
LABEL_ENCODING("curah_hujan")
data
```

| | Tanggal | temp_min | temp_max | temp_rata | kelembaban_rata | kec_angin_rata | lama_penyinaran | arah_angin | kec_angin_max | curah_hujan |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 2020-01-01 | 24.4 | 30.8 | 26.6 | 88.0 | 1.0 | 4.7 | 290.0 | 4.0 | 2 |
| 1 | 2020-02-01 | 24.2 | 32.0 | 27.5 | 85.0 | 1.0 | 4.7 | 300.0 | 5.0 | 3 |
| 2 | 2020-03-01 | 24.6 | 32.2 | 27.8 | 84.0 | 2.0 | 3.7 | 300.0 | 4.0 | 0 |
| 3 | 2020-04-01 | 25.4 | 30.6 | 27.8 | 84.0 | 2.0 | 2.3 | 300.0 | 5.0 | 0 |
| 4 | 2020-05-01 | 25.0 | 31.0 | 26.8 | 88.0 | 1.0 | 0.6 | 290.0 | 5.0 | 0 |

Fig. 3 Label encoding from weather data

*3) Features Selection:* The next step is feature selection to reduce irrelevant features so that only the essential features that can determine the weather will be included in the modeling process later.

```
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import chi2

chi2_selector = SelectKBest(chi2, k=5)
chi2_selector.fit(x, y)

cols = chi2_selector.get_support(indices=True)
x = x.iloc[:,cols]
x
```

| | kelembaban_rata | kec_angin_rata | lama_penyinaran | arah_angin | kec_angin_max |
|---|---|---|---|---|---|
| 0 | 88.0 | 1.0 | 4.7 | 290.0 | 4.0 |
| 1 | 85.0 | 1.0 | 4.7 | 300.0 | 5.0 |
| 2 | 84.0 | 2.0 | 3.7 | 300.0 | 4.0 |
| 3 | 84.0 | 2.0 | 2.3 | 300.0 | 5.0 |
| 4 | 88.0 | 1.0 | 0.6 | 290.0 | 5.0 |
| ... | ... | ... | ... | ... | ... |
| 1703 | 86.0 | 1.0 | 5.0 | 340.0 | 4.0 |
| 1704 | 88.0 | 1.0 | 2.9 | 300.0 | 2.0 |
| 1705 | 88.0 | 1.0 | 1.3 | 310.0 | 3.0 |
| 1706 | 83.0 | 1.0 | 1.1 | 300.0 | 7.0 |
| 1707 | 86.0 | 1.0 | 0.9 | 300.0 | 4.0 |

1708 rows × 5 columns

Fig. 4 Feature Selection with Chi-Square

Based on Fig. 5, the features used in the modeling are average humidity, average wind speed, duration of sunshine, wind direction, and maximum wind speed. The selection of this feature is based on the results of the chi-square test, the process of which is described in the formula (1). The selected feature is also suitable for use as a weather forecast indicator. High humidity can lead to more humid conditions and the potential for rain, while low humidity is usually associated with drier weather. Average wind speed can show how the weather system moves and how wind patterns may change occasionally. The sunshine duration determines how much solar energy reaches the Earth's surface, affecting the temperature and general weather conditions. Wind direction is important because it helps predict how weather patterns will evolve. Wind direction can bring warm or cold air, humidity, or extreme weather conditions like storms. Maximum wind speed is critical in extreme climates such as hurricanes, storms, or strong winds that can cause damage.

*4) Data development:* In the data development stage, the completeness and validity of the data are ensured, the data format is determined, and the data that has been cleaned and prepared is built. In data development, normalization ensures that each attribute in the weather dataset has the same range of values, neither too large nor too small. Standard normalization techniques in the scikit-learn library include StandardScaler, MinMaxScaler, and RobustScaler. In this study, StandardScaler is used. StandardScaler removes the mean (centered at 0) and scales to unit variance (standard deviation = 1), assuming that the data is normally distributed (Gaussian) for all features.

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
data = sc.fit_transform(data)
print(data)
```

```
[[ 0.44890359 -1.48835942 -0.91044073 ...  1.24260507  0.36665557
   0.33281167]
 [ 0.2675906  -0.80500971 -0.11938856 ...  1.35243298  1.10953542
   0.89786369]
 [ 0.63021659 -0.69111809  0.14429549 ...  1.35243298  0.36665557
  -0.79729237]
 ...
 [ 0.99284258 -2.513384   -1.26201947 ...  1.46226089 -0.37622428
   0.89786369]
 [-0.82028739 -1.3744678  -0.73465136 ...  1.35243298  2.59529511
  -0.79729237]
 [ 0.0862776  -0.80500971 -0.82254604 ...  1.35243298  0.36665557
  -0.79729237]]
```

Fig. 5 Data Normalization Results

## C. Data Splitting

The data splitting procedure partitions the data into two distinct subsets: the training data and the testing data. The training dataset comprises 70% of the overall dataset, and the testing dataset comprises 30% of the overall dataset, which amounts to 1698 rows of data.

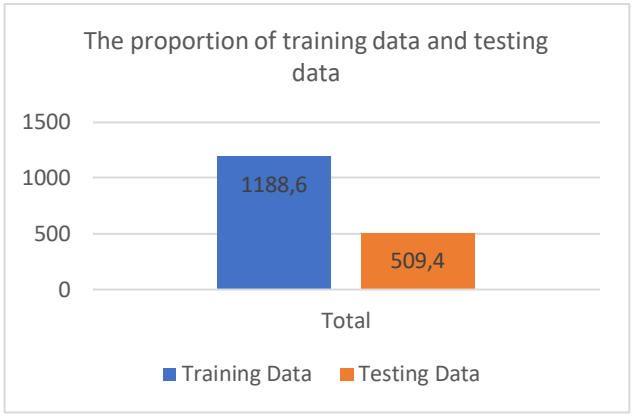


Fig. 6 Proportion of Train Data and Test Data

Below is an example Python code and the print result of the training data:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(data, y, test_size = 0.3, random_state = 0)
print(X_train)
print(X_test)
print(y_train)
print(y_test)
```

```
[[ 0.44476414 -0.15263498 -0.30130477  1.35243298 -1.11910413]
 [-1.34019393  1.20518035  0.2944518  -1.17360896  0.36665557]
 [-1.34019393  1.20518035  1.1355199  -0.07532986  0.36665557]
 ...
```

Fig. 7 Example of training data

Before the modeling process, it must first be ensured whether the data is balanced for each class. Figure 9 provides an overview of the weather dataset.

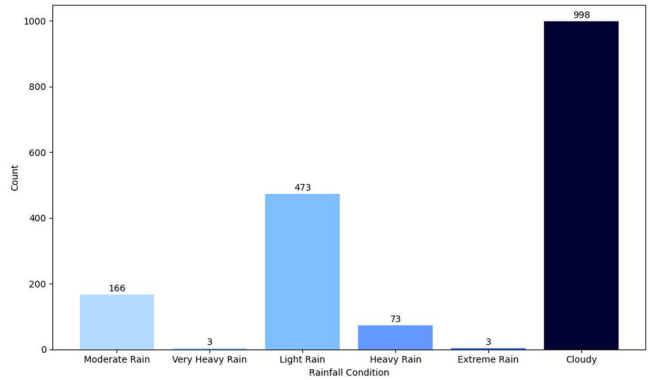


Fig. 8 Overview of Weather Datasets

Based on Figure 9, it can be concluded that the collected weather data needs to be balanced between different classes. Imbalanced data can lead to inaccurate classification results, so techniques are required to balance the dataset. One technique used to address this issue is the SMOTE technique. The Synthetic Minority Over-Sampling Technique (SMOTE) is widely used to address class imbalance. The proposed methodology involves the synthesis of additional samples from the underrepresented class to achieve a balanced dataset. This is accomplished by generating new instances from the minority class using convex combinations with neighboring examples.

```
from imblearn.over_sampling import SMOTE
from imblearn.over_sampling import RandomOverSampler
print("Before SMOTE, counts of label '0': {} \n".format(sum(y_train==0)))
print("Before SMOTE, counts of label '1': {} \n".format(sum(y_train==1)))
print("Before SMOTE, counts of label '2': {} \n".format(sum(y_train==2)))
print("Before SMOTE, counts of label '3': {} \n".format(sum(y_train==3)))
print("Before SMOTE, counts of label '4': {} \n".format(sum(y_train==4)))
print("Before SMOTE, counts of label '5': {} \n".format(sum(y_train==5)))

sm = SMOTE(random_state=42,k_neighbors=1)
X_train_res, y_train_res = sm.fit_resample(X_train, y_train)


print('After SMOTE, the shape of train_X: {}'.format(X_train_res.shape))
print('After SMOTE, the shape of train_y: {} \n'.format(y_train_res.shape))

print("After SMOTE, counts of label '0': {}".format(sum(y_train_res==0)))
print("After SMOTE, counts of label '1': {}".format(sum(y_train_res==1)))
print("After SMOTE, counts of label '2': {}".format(sum(y_train_res==2)))
print("After SMOTE, counts of label '3': {}".format(sum(y_train_res==3)))
print("After SMOTE, counts of label '4': {}".format(sum(y_train_res==4)))
print("After SMOTE, counts of label '5': {}".format(sum(y_train_res==5)))
```

Fig. 9 Process of Handling Imbalance Data

Based on the process shown in Figure 10, the results of the SMOTE sampling technique are presented in Table 3.

TABLE III
SAMPLING RESULT WITH SMOTE

| Class | Rainfall | Original Data | After SMOTE |
|---|---|---|---|
| 0 | Cloudy | 700 | 700 |
| 1 | Light rain | 2 | 700 |
| 2 | Moderate rain | 52 | 700 |
| 3 | Heavy Rain | 320 | 700 |
| 4 | Very Heavy Rain | 3 | 700 |
| 5 | Extreme Rain | 118 | 700 |

*D. Modeling with Random Forest*

The model is constructed using the ensemble learning method, namely Random Forest, to obtain weather prediction results. In this study, the target class used is rainfall, which consists of 6 classes: cloudy, light rain, moderate rain, heavy rain, very heavy rain, and extreme rain. The modeling with Random Forest is performed using the sklearn library in Python, as shown in Figure 11.

Random Forest Classifier

```
[118] from sklearn.ensemble import RandomForestClassifier
      from sklearn.ensemble import RandomForestClassifier
      from sklearn.metrics import confusion_matrix
      from sklearn.metrics import classification_report
      from sklearn.metrics import accuracy_score
      forest= RandomForestClassifier(n_estimators = 100, random_state = 42,max_depth=100)
      forest.fit(X_train_res,y_train_res)
      y_pred = forest.predict(X_test)
      cm = confusion_matrix(y_test,y_pred)
```

Fig. 10 Random Forest Classifier

The modeling is performed by setting the n_estimators, which indicates the number of trees in the forest, to 100. Then, the random state is set to 42, and the maximum depth of the trees is set to 100. The model obtained from the training process is then implemented to predict the data in the testing set and evaluated using a confusion matrix. The prediction results are then visualized in a heatmap graph. A heatmap is a rectangular plot of data represented as a color-encoded matrix. It requires a 2D dataset, which can be displayed as an array. This is a great way to visualize data, showing relationships between variables. Then, the classification report is printed for evaluation, which comes from the sklearn.metrics library.

```
[834] from sklearn.metrics import classification_report
      print(classification_report(y_test,y_pred))
```

Fig. 11 Classification Report Python Code

Based on Figure 12, the overall accuracy of the testing data is 87.6% with an average recall of 0.88, precision of 0.88, and F1-score of 0.88. Meanwhile, the detailed values of precision, recall, and F1-score for each class are shown in Table 4.

TABLE IV
MODEL EVALUATION

| Class | Rainfall | Accuracy | Precision | Recall | F1 Score |
|---|---|---|---|---|---|
| 0 | Cloudy | 0.76 | 0.84 | 0.76 | 0.80 |
| 1 | Light rain | 0.99 | 0.99 | 1.00 | 0.99 |
| 2 | Moderate rain | 0.93 | 0.87 | 0.93 | 0.9 |
| 3 | Heavy Rain | 0.74 | 0.74 | 0.74 | 0.74 |
| 4 | Very Heavy Rain | 0.99 | 1.00 | 1.00 | 1.00 |
| 5 | Extreme Rain | 0.84 | 0.84 | 0.84 | 0.84 |

Based on Table 3, With a 76% accuracy rate, the model predicts cloudy weather with a 76% success rate. The model's accuracy in forecasting light rain is an impressive 99%. The model can expect moderate rain with a 93% accuracy rate. The precision, recall, and F1 Score for heavy rain were all 74%, demonstrating an excellent balance between memory and precision, even though the accuracy was only 74%. Despite this, overall accuracy might be improved. The model has high accuracy and other metrics (99-100%) for predicting heavy rain. Suppose the model has a high accuracy or metric, such as 99-100%, for a particular class (such as "Very Heavy Rain"). This can indicate that the model may be over-expressing the training data and not generalizing well to new

situations. Accuracy of 84%, precision, recall, and F1 Score of 84% for the extreme weather class.

### *E. Evaluation*

The model has high performance, especially for the Light Rain and Very Heavy Rain classes. The Cloudy and Extreme Rain classes also performed well, although with slightly lower accuracy. The Moderate rain and Heavy Rain classes show a balance between precision and recall. Based on this accuracy value, it is concluded that the model can predict weather classes (rainfall). However, there is still some overfitting for courses like light rain and very heavy rain, so it is necessary to add more training data to improve the results significantly.

## IV. CONCLUSION

The findings derived from weather forecasting modeling utilizing the ensemble learning approach, specifically random forest, with five features and 6 target classes, it can be concluded that the model achieved an accuracy of 87.6%, which is considered quite good. A suggestion for future research is to experiment with other prediction methods, such as deep learning. Deep learning models, particularly neural networks, can effectively capture complex patterns and relationships in data, improving accuracy, precision, and recall in weather forecasting. Additionally, implementing the results in a mobile-based application could make it easily accessible to the public. This could make the weather forecasting information easily accessible to the public.

## ACKNOWLEDGMENT

The authors would like to acknowledge the financial support provided by the Research and Community Service Institute (LPPM) of Universitas Pembangunan Nasional Veteran Jakarta and Indramayu Regency Government to conduct this research.

## REFERENCES

[1] K. Kbbi, "Kamus Besar Bahasa Indonesia (KBBI)," *Kementerian Pendidikan Dan Budaya*, 2016.

[2] Kabupaten Indramayu, "Kabupaten Indramayu - Website Resmi Pemerintah Daerah Provinsi Jawa Barat." Accessed: Mar. 17, 2022. [Online]. Available: https://jabarprov.go.id/index.php/pages/id/1052

[3] JabarProv, "Anomali Cuaca Ancam Produksi Padi - Website Resmi Pemerintah Daerah Provinsi Jawa Barat." Accessed: Mar. 17, 2022. [Online]. Available: https://jabarprov.go.id/index.php/news/8773/Anomali_Cuaca_Ancam_Produksi_Padi

[4] B. Bochenek and Z. Ustrnul, "Machine Learning in Weather Prediction and Climate Analyses—Applications and Perspectives," Atmosphere, vol. 13, no. 2, p. 180, Jan. 2022, doi:10.3390/atmos13020180.

[5] D. N. Fente and D. Kumar Singh, "Weather Forecasting Using Artificial Neural Network," 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), pp. 1757–1761, Apr. 2018, doi:10.1109/icicct.2018.8473167.

[6] P. Karvelis, S. Kolios, G. Georgoulas, and C. Stylios, "Ensemble learning for forecasting main meteorological parameters," 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 3711–3714, Oct. 2017, doi: 10.1109/smc.2017.8123210.

[7] N. Singh, S. Chaturvedi, and S. Akhter, "Weather Forecasting Using Machine Learning Algorithm," 2019 International Conference on Signal Processing and Communication (ICSC), Mar. 2019, doi: 0.1109/icsc45622.2019.8938211.

[8] F. Q. Kareem, A. M. Abdulazeez, and D. A. Hasan, "Predicting Weather Forecasting State Based on Data Mining Classification Algorithms," Asian Journal of Research in Computer Science, pp. 13–24, Jun. 2021, doi: 10.9734/ajrcos/2021/v9i330222.

[9] U. M. Khaire and R. Dhanalakshmi, "Stability of feature selection algorithm: A review," Journal of King Saud University - Computer and Information Sciences, vol. 34, no. 4, pp. 1060–1073, Apr. 2022, doi: 10.1016/j.jksuci.2019.06.012.

[10] B. O. Macaulay, B. S. Aribisala, S. A. Akande, B. A. Akinnuwesi, and O. A. Olabanjo, "Breast cancer risk prediction in African women using Random Forest Classifier," Cancer Treatment and Research Communications, vol. 28, p. 100396, 2021, doi:10.1016/j.ctarc.2021.100396.

[11] S. K. Trivedi, "A study on credit scoring modeling with different feature selection and machine learning approaches," Technology in Society, vol. 63, p. 101413, Nov. 2020, doi:10.1016/j.techsoc.2020.101413.

[12] R. Spencer, F. Thabtah, N. Abdelhamid, and M. Thompson, "Exploring feature selection and classification methods for predicting heart disease," Digital Health, vol. 6, Jan. 2020, doi:10.1177/2055207620914777.

[13] BMKG, "Data Online - Pusat Database - BMKG." Accessed: Nov. 11, 2023. [Online]. Available: https://dataonline.bmkg.go.id/home.

[14] D. Munková, M. Munk, and M. Vozár, "Data Pre-processing Evaluation for Text Mining: Transaction/Sequence Model," Procedia Computer Science, vol. 18, pp. 1198–1207, 2013, doi: 10.1016/j.procs.2013.05.286.

[15] I. F. Ilyas and X. Chu, *Data cleaning*. Morgan & Claypool, 2019.

[16] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, "A survey on missing data in machine learning," Journal of Big Data, vol. 8, no. 1, Oct. 2021, doi: 10.1186/s40537-021-00516-9.

[17] S. Bahassine, A. Madani, M. Al-Sarem, and M. Kissi, "Feature selection using an improved Chi-square for Arabic text classification," Journal of King Saud University - Computer and Information Sciences, vol. 32, no. 2, pp. 225–231, Feb. 2020, doi: 10.1016/j.jksuci.2018.05.010.

[18] Q. H. Nguyen et al., "Influence of Data Splitting on Performance of Machine Learning Models in Prediction of Shear Strength of Soil," Mathematical Problems in Engineering, vol. 2021, pp. 1–15, Feb. 2021, doi: 10.1155/2021/4832864.

[19] H. Ali, M. N. Mohd Salleh, R. Saedudin, K. Hussain, and M. F. Mushtaq, "Imbalance class problems in data mining: a review," Indonesian Journal of Electrical Engineering and Computer Science, vol. 14, no. 3, p. 1552, Jun. 2019, doi: 10.11591/ijeecs.v14.i3.pp1552-1563.

[20] G. A. Pradipta, R. Wardoyo, A. Musdholifah, I. N. H. Sanjaya, and M. Ismail, "SMOTE for Handling Imbalanced Data Problem : A Review," 2021 Sixth International Conference on Informatics and Computing (ICIC), pp. 1–8, Nov. 2021, doi: 10.1109/icic54025.2021.9632912.

[21] Z.-H. Zhou, "Ensemble Learning," Machine Learning, pp. 181–210, 2021, doi: 10.1007/978-981-15-1967-3_8.

[22] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, "A survey on ensemble learning," Frontiers of Computer Science, vol. 14, no. 2, pp. 241–258, Aug. 2019, doi: 10.1007/s11704-019-8208-z.

[23] O. Sagi and L. Rokach, "Ensemble learning: A survey," WIREs Data Mining and Knowledge Discovery, vol. 8, no. 4, Feb. 2018, doi:10.1002/widm.1249.

[24] X. Gao, M. I. Ramli, and S. M. Z. Syed Zainal Ariffin, "A Comparative Analysis of Combination of CNN-Based Models with Ensemble Learning on Imbalanced Data," JOIV : International Journal on Informatics Visualization, vol. 8, no. 1, p. 456, Mar. 2024, doi:10.62527/joiv.8.1.2194.

[25] H. Tyralis, G. Papacharalampous, and A. Langousis, "A Brief Review of Random Forests for Water Scientists and Practitioners and Their Recent History in Water Resources," Water, vol. 11, no. 5, p. 910, Apr. 2019, doi: 10.3390/w11050910.

[26] A. E. K. Gunawan and A. Wibowo, "Stock Price Movement Classification Using Ensembled Model of Long Short-Term Memory (LSTM) and Random Forest (RF)," JOIV : International Journal on Informatics Visualization, vol. 7, no. 4, Dec. 2023, doi:10.30630/joiv.7.4.1640.

[27] H. Hairani, A. Anggrawan, and D. Priyanto, "Improvement Performance of the Random Forest Method on Unbalanced Diabetes Data Classification Using Smote-Tomek Link," JOIV : International Journal on Informatics Visualization, vol. 7, no. 1, p. 258, Feb. 2023, doi:10.30630/joiv.7.1.1069.

[28] Gde Agung Brahmana Suryanegara, Adiwijaya, and Mahendra Dwifebri Purbolaksono, "Peningkatan Hasil Klasifikasi pada Algoritma Random Forest untuk Deteksi Pasien Penderita Diabetes Menggunakan Metode Normalisasi," Jurnal RESTI (Rekayasa Sistem dan Teknologi

Informasi), vol. 5, no. 1, pp. 114–122, Feb. 2021, doi: 10.29207/resti.v5i1.2880.
[29] J. L. Speiser, M. E. Miller, J. Tooze, and E. Ip, "A comparison of random forest variable selection methods for classification prediction modeling," Expert Systems with Applications, vol. 134, pp. 93–101, Nov. 2019, doi:10.1016/j.eswa.2019.05.028.
[30] J. Wang, X. Jing, Z. Yan, Y. Fu, W. Pedrycz, and L. T. Yang, "A Survey on Trust Evaluation Based on Machine Learning," ACM Computing Surveys, vol. 53, no. 5, pp. 1–36, Sep. 2020, doi: 10.1145/3408292.
[31] S. Adinugroho and Y. A. Sari, *Implementasi Data Mining Menggunakan Weka*. Universitas Brawijaya Press, 2018.
[32] BMKG, "Probabilistik Curah Hujan 20 mm (tiap 24 jam) | BMKG." Accessed: Jan. 16, 2024. [Online]. Available: https://www.bmkg.go.id/cuaca/probabilistik-curah-hujan.bmkg.