Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th February 2025. Vol. 4. No. 2; e-ISSN: 2808-4519

Optimization of Social Assistance Recipient Determination using Gradient Boosting Algorithm

Windi Herlita Vidila[1*], Rudi Kurniawan[2], Saeful Anwar[3]

[1,2,3]STMIK IKMI Cirebon
Jl. Perjuangan No. 10B Majasem, Kesambi, Cirebon, Indonesia
herlitawindi01@gmail.com[1]*

Abstract

This research aims to classify social assistance recipients to ensure the accuracy of aid distribution by utilizing the Gradient Boosting algorithm on RapidMiner. The data used is data on residents who are categorized as receiving and not receiving social assistance in Cicadas village with a total dataset consisting of 670 entries with 18 attributes that will be divided equally between eligible and ineligible recipients. This research uses KDD (Knowledge Discover in Database) analysis which includes the stages of data selection, pre-processing, transformation, modeling, and interpretation of results. This research uses a quantitative approach, focusing on the distribution of datasets in a ratio of 70:30 with a stratified sampling technique for training and testing purposes. The experimental results show that the selected method is effective in classifying recipients by obtaining an accuracy of 91.67%, this accuracy result can be relied upon to support decision-making in social assistance distribution. The findings underscore the potential of machine learning in optimizing social welfare initiatives by improving target accuracy and ensuring aid reaches the rightful recipients.

Keywords: *Gradient Boosting, Social Assistance Distribution, KDD (Knowledge Discovery in Database), Split Data, Classification Accuracy*

1. Introduction

Classification is one of the techniques in computer science that has an important role in various fields, especially in data processing for decision making. In the context of informatics, classification is used to group data into predefined categories based on certain features [1]. For example, in the social assistance sector, classification can be used to determine who is eligible to receive assistance based on various parameters such as income, family status, or social conditions. This technique enables more precise and efficient decision-making, and can reduce errors in determining social assistance recipients that often occur if done manually [2]. Therefore, classification has great potential to improve accuracy and efficiency in the distribution of social assistance, which has a direct effect on poverty alleviation and community welfare [3].

In Cicadas Village, there are still significant problems related to the determination of social assistance recipients that are often not on target. Some social assistance recipients do not meet the criteria, while some families in need do not receive assistance. This inaccuracy is largely due to subjectivity in the selection process, which is done manually by officers, as well as the limited data available, this causes the distribution of social assistance to be suboptimal and leads to injustice [4]. Therefore, a more appropriate and data-based solution is needed to improve the accuracy of determining social assistance recipients in the village. One approach that can be used is the application of machine learning algorithms, especially Gradient Boosting, to classify social assistance recipients based on available data [5].

This research aims to develop and implement a classification model using the Gradient Boosting algorithm to improve accuracy in determining social assistance recipients in Cicadas Village. By using this model, it is expected that the selection process of beneficiaries can be more objective, accurate, and based on relevant data. In addition, this research also aims to evaluate the effectiveness of using classification techniques in the context of social assistance information systems, by measuring the level of accuracy of the model in classifying individuals entitled to receive assistance. The results of this research are expected to make a significant contribution in improving a fairer and more efficient social assistance distribution system in Cicadas Village.

In this research, data analysis will be conducted using Knowledge Discovery in Database (KDD) techniques, which involve the stages of data selection, cleaning, transformation, modeling and interpretation [6]. The approach used is a quantitative experiment, where the data collected will be statistically analyzed to identify patterns that can be used to determine social assistance recipients. The tools used for analysis RapidMiner, which are programming platforms and software for data analysis and machine learning. In this case, the Gradient Boosting algorithm was chosen due to its superior ability to improve the accuracy of classification models by combining predictions from several simpler models to produce better predictions [7]. This algorithm has proven effective in various machine learning applications, including prediction and classification [8].

The results of this research are expected to have significant implications in improving the accuracy and fairness of social assistance distribution in Cicadas Village. By using the Gradient Boosting algorithm, the social assistance recipient selection system can be more objective and data-based, reducing human error in the decision-making process. In addition, the results of this study can also be a reference for village governments and related agencies in developing more efficient and transparent technology-based information systems for managing social assistance. The application of this method is expected to have a positive impact on improving community welfare, as well as opening up opportunities for the application of similar technology in other villages facing similar problems.

1. Research Methods

This research uses the Knowledge Discovery in Database (KDD) method to process data on social assistance recipients. The Gradient Boosting algorithm is applied for classification, due to its ability to improve model accuracy by combining predictions from several simple models. With this approach, the research aims to produce a more precise classification model in determining social assistance recipients in Cicadas Village. for more details can be seen in Figure 1.

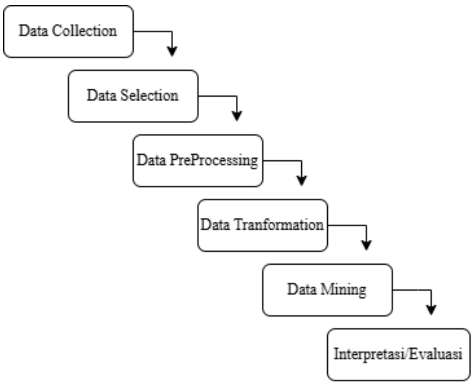

Figure 1: Research Methods

2.1. Knowledge Discovery in Databases (KDD)

KDD (Knowledge Discovery in Database) is the process of finding useful patterns or information from big data through a series of stages. It starts with data selection, which selects relevant data, then data cleaning to remove incomplete or inaccurate data. Next, the cleaned data will be transformed into a format suitable for further analysis. A key process in KDD is data mining, where statistical techniques or machine learning algorithms are used to identify patterns or relationships in the data. Once patterns are discovered, an evaluation stage is performed to assess the results, and finally, the knowledge gained will be presented in a form that can be used for decision-making. KDD is essential in processing big data to generate valuable insights, such as in the determination of more accurate social assistance recipients.

2.2. Gradient Boosting Algorithm

Gradient Boosting is a machine learning algorithm used to build robust prediction models by combining several simpler predictor models, usually decision trees. The main principle of Gradient Boosting is to iteratively improve the model, where each new model focuses on correcting the errors (residuals) of the previous model. In this process, the model first makes predictions based on existing data, then the next model tries to reduce the remaining error by learning patterns that have not been detected by the previous model. This process continues until an optimal model is reached [9]. Gradient Boosting is very effective in improving accuracy, especially in complex and large datasets, due to its ability to reduce model bias and variance by combining many weak predictor models into one stronger model [10].

The formula for the stepwise process of the gradient boosting algorithm is as follows:

$$(\ ) = \quad {}_{-1}(\ ) + \quad \cdot \quad (\ )$$

Each decision tree is built to reduce the residuals from the previous model. This process is optimized by using the gradient derivative of the loss function. For example, for each iteration the prediction model Fm(x) is updated by adding a new model hm(x), which is calculated based on the gradient of the loss function.

2.3. Classification Rules

Classification is a technique in machine learning used to classify data into predefined categories or classes based on features. In classification, models are trained using labeled datasets (data with known categories or classes) to learn patterns or relationships between features and classes. The basic rule in classification involves learning from data that has already been labeled, then using that information to predict the class of the unlabeled data. One of the commonly used methods in classification is decision tree-based algorithms, which divide data into subsets based on certain conditions [11].

Mathematically, classification can be defined as a function f:X→Y, where X is the feature space (e.g., attributes or independent variables), and Y is the class or category space (e.g., the category to be predicted). The goal is to find a function f that maps the input data X into the correct class Y. This process involves separating the data in the feature space with rules or constraints that separate one class from another. The classification model is then tested using new data to see how well it can predict the correct class.

2. Results and Discussion

3.1. Results

3.1.1 Data Collection

The data used in this study is a dataset taken from the Cicadas village office in excel form with the name Bansos Cicadas Dataset Lengkap. this data set consists of 670 data with 18 attributes as shown in Table 1 below:

Table 1: Overview of Dataset

No	Nik	Nama_Penerima	…	Bansos 2024
1	3210112…	ENDRIYANTO	…	YA
2	3210114…	DINI HAQI ZULFA	…	YA
3	3210092…	TONI AHMAD	…	YA
4	3210110…	SIGIT NUGRAHA	…	YA
5	3210112…	AEP SAEPUDIN	…	YA
6	3210110…	DUDUNG ABDU	…	YA
7	3210110…	ENDING SUPARDI	…	YA
8	3210115…	RASIMAH	…	YA
9	3210111…	DADI JUNAEDI	…	YA
10	3274031…	WUWUN PREDIA	…	YA
…	…	…	…	…
670	3210112…	RUSNATA	…	TIDAK

for more details in Table 2 shows information about attribute data types

Table 2: Statistic Datasets

Attributes	Data Type
No	Integer
NIK	Real
Nama_Penerima	Nominal
Tanggal Lahir	Date-Time
Jenis Kelamin	Nominal
Desa/Kel.	Nominal
Alamat	Nominal
RW	Integer
RT	Integer
Hub.Kel	Nominal
Pekerjaan	Nominal
Pendapatan Umum	Nominal
Kepemilikan Tanah	Nominal
Pendidikan	Nominal
Status Kawin	Nominal
AK	Integer
Ket KPM	Nominal
Bansos 2024	Nominal

3.1.2 Data Selection

The process model at the Data Selection step in RapidMiner can be seen in Figure 2

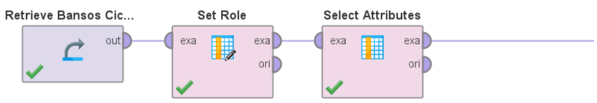

Figure 2: Data Selection Process in RapidMiner

To import external data into the analysis environment, the retrieve operator is used. This operator serves to retrieve data from various sources, such as local files (e.g. CSV, Excel), databases, or other data sources, and load them into the RapidMiner workflow for further processing.In the Retrieve operator there are parameters as in Table 3

Table 3: Retrieve Parameter

Parameter	Value
Repository Entry	Bansos Cicadas Dataset Lengkap 670

Table 4 shows the results of the retrieve operator

Table 4: Retrieve Output Result

No	Nik	Nama_Penerima	…	Bansos 2024
1	3210112…	ENDRIYANTO	…	YA
2	3210114…	DINI HAQI ZULFA	…	YA
3	3210092…	TONI AHMAD	…	YA
4	3210110…	SIGIT NUGRAHA	…	YA
5	3210112…	AEP SAEPUDIN	…	YA
6	3210110…	DUDUNG ABDU	…	YA
7	3210110…	ENDING SUPARDI	…	YA
8	3210115…	RASIMAH	…	YA
9	3210111…	DADI JUNAEDI	…	YA
10	3274031…	WUWUN PREDIA	…	YA
…	…	…	…	…
670	3210112…	RUSNATA	…	TIDAK

After the data is imported using the retrieve operator, the data will then be labeled with a target using the Set Role operator. This set role operator serves to define a specific role for each attribute, which is important in preparing data for analysis or modeling, especially in the context of machine learning. The attribute that will be the target or label is the “Bansos 2024” attribute, then the “No” attribute will be used as an Id as shown in table 5

Table 5: Set Role Parameter

No	Parameter	Role
1	Bansos 2024	Target (Label)
2	No	Id

from the use of set roles, Table 6 shows the information obtained from the set role operator.

Table 6: Set Role Output Result

No.	Name	Description
1.	Record	670
2.	Atribut Spesial	2
3.	Atribut Reguler	16
4.	Atribut:	
	Bansos 2024 (Label)	Nominal
	No (Id)	Integer
	NIK	Real
	Nama_Penerima	Nominal
	Tanggal Lahir	Date-Time
	Jenis Kelamin	Nominal
	Desa/Kel.	Nominal
	Alamat	Nominal
	RW	Integer
	RT	Integer
	Hub.Kel	Nominal
	Pekerjaan	Nominal
	Pendapatan Umum	Nominal
	Kepemilikan Tanah	Nominal
	Pendidikan	Nominal
	Status Kawin	Nominal
	AK	Integer
	Ket KPM	Nominal

The next step after the set role operator is used is the select attributes operator. The Select Attributes operator in RapidMiner serves to select specific attributes (columns) in a dataset to be used in analysis or modeling such as selecting, deleting, or modifying existing attributes based on certain criteria, such as attribute name, attribute type, or attribute role. By using this operator, users can reduce the number of attributes in the dataset, which helps in improving data processing efficiency, reducing model complexity, and avoiding overfitting. The parameters used in the select attributes operator can be seen in Table 7

Table 7: Select Attributes Parameters

No	Parameter	Value
1	Type	Include Attributes
2	Attribute Fillter Type	A Subset
3	Select Subset	Select Attributes…
		Bansos 2024
		No
		AK
		Ket KPM
		Tanggal Lahir
		Jenis Kelamin
		Alamat
		RW
		RT
		Hub.Kel

	Pekerjaan
	Pendapatan Umum
	Kepemilikan Tanah
	Pendidikan
	Status Kawin

Based on table 6 which shows the select attribute operator obtained information that the attributes used are only 15 attributes from a total of 18 attributes, this is because 3 attributes, namely the “NIK” attribute, the “Nama_penerima” attribute, the “Desa/Kel” attribute is redundant or has the same value as attribute no.

3.1.3 Data Preprocessing

The model of the preprocessing process in RapidMiner can be seen in Figure 3.

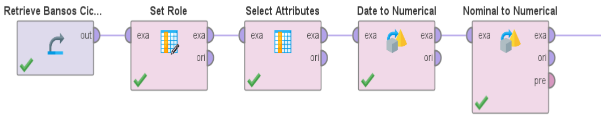

Figure 3: Preprocessing process model in RapidMiner

After the data is collected and selected, the next step is data preprocessing. data preprocessing is a series of steps or techniques used to prepare data before analyzing or applying machine learning algorithms. The steps include data cleaning to handle missing values and outliers, data transformation such as normalization and encoding, and dimensionality reduction by feature selection or extraction. In addition, data can also be merged or integrated if it comes from different sources. This preprocessing is important to ensure the data used in the analytics or machine learning model is clean, relevant, and of high quality, which in turn improves the accuracy and efficiency of the model.

At this stage, the data does not go through pre-processing because the data obtained is complete (no missing values) and clean as shown in table 8

Table 8: No Missing Data

No	Atribut	Tipe Data	Missing Value
1	No	Integer	0
2	NIK	Real	0
3	Nama_Penerima	Nominal	0
4	Tanggal Lahir	Date-Time	0
5	Jenis Kelamin	Nominal	0
6	Desa/Kel.	Nominal	0
7	Alamat	Nominal	0
8	RW	Integer	0
9	RT	Integer	0
10	Hub.Kel	Nominal	0
11	Pekerjaan	Nominal	0
12	Pendapatan Umum	Nominal	0
13	Kepemilikan Tanah	Nominal	0
14	Pendidikan	Nominal	0
15	Status Kawin	Nominal	0
16	AK	Integer	0
17	Ket KPM	Nominal	0
18	Bansos 2024	Nominal	0

3.1.4 Data Transformation

After ensuring that the dataset is complete, the data will then go through the transformation process. Data transformation is the process of changing or modifying data into a form that is more suitable or useful for analysis or application of machine learning models. The process model at the Transformation step in RapidMiner can be seen in Figure 4 below:

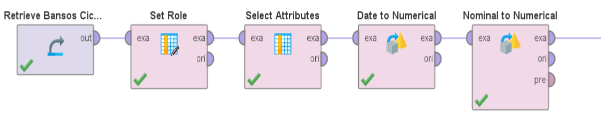

Figure 4: Transformation Process Model in RapidMiner

In this research, the transformation process is carried out using the date to numeric operator. this serves to convert date type data into numeric type data. The parameters used in the Date To Numerical operator can be seen in Table 9

Table 9: Date To Nominal Parameter

Parameter	Value
Attributes Name	TANGGAL LAHIR

Time Unit	Month
Month Relative	Year

After the data with date type is converted into numeric type, the results are shown in Table 10

Table 10: Date to Numerical Output Result

No	Tanggal Lahir (Date)	Tanggal Lahir (Numerik)
1	26/05/1993	5
2	07/05/1992	5
3	28/03/1988	3
4	01/04/1986	4
5	27/06/1987	6
6	07/07/1978	7
7	04/07/1960	7
8	16/06/1949	6
9	14/02/1987	2
10	18/10/1989	10
11	15/05/1978	5
12	30/07/1983	7
13	16/07/1955	7
14	03/04/1979	4
15	31/12/1989	12
16	04/12/1991	12
17	10/05/1980	5
18	08/09/1987	9
19	04/04/1976	4
20	06/08/1979	8
…	…	…
669	28/06/1977	6
670	25/05/1970	5

The second stage in the data transformation process is using the nominal to numeric operator. this operator functions to convert nominal type data into numeric type data. The parameters of the Numerical to Binominal operator used, as in Table 11 below:

Table 11: Nominal to Numerical Parameter

Parameter	Value
Attributes Filter Type	All
Coding Type	Unique Integers

Table 12 below is the result of data transformation after processing using the Nominal to Numerical operator:

Table 12: Nominal to Numerical Output Result

Atribut	Nominal	Numerik
Jenis Kelamin	Laki-Laki	0
	Perempuan	1
Alamat	Dusun Gugunungan	0
	Dusun Timbangan	1
	Dusun Cihujan	2
	Dusun Cangkuang	3
Hub. Kel	Kepala Keluarga	0
	Istri	1
	Anak	2
Pekerjaan	Buruh Harian Lepas	0
	Mengurus Rumah Tangga	1
	Wiraswasta	2
	Karyawan Swasta	3
	Pedagang	4
	Sopir	5
	Pelajar/Mahasiswa	6
	Petani/Pekebun	7
	Pensiunan	8
	Perangkat Desa	9
	Belum/Tidak Bekerja	10
	Tukang Kayu	11
	Pegawai Negeri Sipil	12
	Industri	13
	Karyawan Bumn	14
Pendapatan Umum	Rp.1.000.000 – Rp.2.000.000	0
	Rp.500.000 – Rp.1.000.000	1
	Rp.2.000.000 – Rp.2.500.000	2
	Rp.1.500.000 – Rp.2.500.000	3
	Rp.1.500.000 – Rp.2.000.000	4
	Rp.0	5
	Rp.2.500.000 – Rp.4.000.000	6
	Rp.2.500.000 – Rp.3.500.000	7

	Rp.4.000.000 – Rp.7.000.000	8
Kepemilikan Tanah	Ya	0
	Tidak	1

3.1.5 Data Mining

After the data is transformed so that the data is ready in a form that is more suitable or useful for analysis or model application, the next step is data mining. in this data mining process, split data operators, gradient boosted trees operators, apply model operators and performance (classification) operators are used. The parameters used in the FP-Growth operator.

The following is a process model of the data mining stage of the rapid miner which is shown in Figure 5

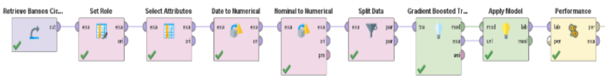

Figure 5 : Data Mining Process Model in RapidMiner

The first process in the data mining stage is to split the data using the split data operator. Data splitting is the process of dividing a dataset into two or more subsets, usually for the purpose of training and testing a model. This split is important to evaluate the performance of the model with data that was not used during training, thus avoiding overfitting. The parameters used in the split data process are shown in Table 13

Table 13: Split Data Parameter

No	Parameters	Isi
1	Rasio	70:30
2	Sampling type	Stratified Sampling

In Table 13, we get information that the split data ratio used is 70:30, this means that the data used will be divided 70% for training data and 30% for testing data.

After the data is divided using the split data operator, the next stage uses the gradient boosted tress operator. Gradient Boosted Trees operator is used to build Gradient Boosting models, which is an ensemble-based machine learning technique. Gradient Boosting combines multiple decision trees to improve the accuracy of the model by building the trees sequentially, where each new tree corrects the mistakes made by the previous tree. The parameters used in the gradient boosted tress operator are as shown in table 14.

Table 14: Gradient Boosted Trees Parameter

No	Parameters	Isi
1	Number Of Trees	50
2	Maximal Depth	5
3	Min Rows	10.0
4	Min Split Improvement	1.0E-5
5	Number Of Bin	20
6	Learning Rate	0.01
7	Sample Rate	1.0
8	Distribution	AUTO

The result of using the gradient boosted trees operator is a decision tree visualization model as can be seen in Figure 6.

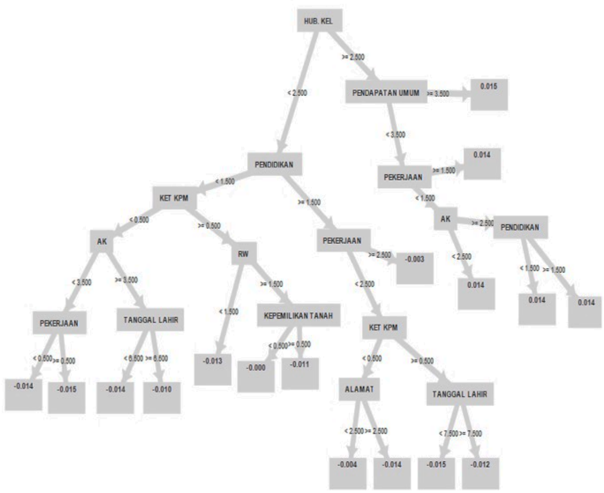

Figure 6 : Gradient Boosted Trees Output Result

The next stage uses the apply model operator. The Apply Model operator is used to apply the trained model to new data to make predictions. Once the model has been built using the gradient boosting algorithm this operator allows the model to be tested or applied to new datasets to generate predictions. In the apply model operator, no parameters are used because in the apply model operator there are no parameters that must be used.

Table 15 below shows the results of using the apply model operator

Table 15: Apply Model Output Result

No	Bansos 2024	Prediction (Bansos 2024)	Confidence YA	Conviedence TIDAK
1	YA	YA	0.694	0.305
2	YA	YA	0.691	0.308
3	YA	YA	0.694	0.305
4	YA	YA	0.672	0.327
5	YA	YA	0.698	0.301
6	YA	YA	0.697	0.302
7	YA	YA	0.642	0.357
8	YA	YA	0.568	0.431
9	YA	YA	0.568	0.431
10	YA	YA	0.617	0.382
…	…	…	…	…
670	TIDAK	TIDAK	0.477	0.523

The next stage is to use the performance operator (classification). The Performance (Classification) operator is used to evaluate the performance of the classification model after it has been applied to the test data. This operator compares the predictions generated by the model with the original values (class labels) in the test data and generates various evaluation metrics to assess how good the model is. The parameters used in the performance operator can be seen in Table 16 below:

Table 16: Performance (Classification) Parameter

Parameter	Isi
Performance	Accuracy Weighted mean recall Weighted mean precision

The results of this performance operator are in the form of a confusion matrix as shown in Table 17, Table 18, Table 19, Table 20, and Table 21 below:

Table 17: Confusion Matrix 1

	True YA	True TIDAK	Class Precision
Pred. YA	219	24	90.12%
Pred. TIDAK	15	210	93.33%
Class Recall	93.59%	89.74%	

Table 18: Confusion Matrix 2

	True YA	True TIDAK	Class Precision
Pred. YA	219	24	90.12%
Pred. TIDAK	15	210	93.33%
Class Recall	93.59%	89.74%	

Table 19: Confusion Matrix 3

	True YA	True TIDAK	Class Precision
Pred. YA	229	34	87.07%
Pred. TIDAK	5	200	97.56%
Class Recall	97.86%	85.47%	

Table 20: Confusion Matrix 4

	True YA	True TIDAK	Class Precision
Pred. YA	229	34	87.07%
Pred. TIDAK	5	200	97.56%
Class Recall	97.86%	85.47%	

Table 21: Confusion Matrix 21

	True YA	True TIDAK	Class Precision
Pred. YA	216	21	91.14%
Pred. TIDAK	18	213	92.21%
Class Recall	92.31%	91.03%	

3.1.6 Interpretation and Evaluation

Based on table 17, table 18, table 19, table 20, and table 21, it can be concluded that the best accuracy with a value of 91.67 is obtained with different attribute attribute configurations. for more details can be seen in table 22 below:

Table 22: Hasil Akurasi Terbaik

No	Kombinasi Atribut	Jumlah Atribut	Akurasi
1	Ket KPM, AK, Kepemilikan Tanah, Pekerjaan, Hub.Kel, RW, Alamat, Jenis Kelamin, Tanggal Lahir	9	91,67%
2	Ket KPM, AK, Status Kawin, Kepemilikan Tanah, Pekerjaan, Hub.Kel, RW, Alamat, Jenis Kelamin, Tanggal Lahir	10	91,67%
3	AK, Pendidikan, Kepemilikan Tanah, Pendapatan Umum, Hub.Kel, RT, RW, Alamat, Jenis Kelamin, Tanggal Lahir	10	91,67%
4	AK, Status Kawin, Pendidikan, Kepemilikan Tanah, Pendapatan Umum, Hub.Kel, RT, RW, Alamat, Jenis Kelamin, Tanggal Lahir	11	91,67%
5	AK, Pendidikan, Kepemilikan Tanah, Pendapatan Umum, Pekerjaan, Hub.Kel, RT, RW, Alamat, Jenis Kelamin, Tanggal Lahir	11	91,67%

Based on Table 22, the experimental results using the Gradient Boosted Trees method show some important findings as follows:

1. the consistency of accuracy can be seen from all attribute combinations that produce the same accuracy value, which is 91.67%, even though the number of attributes used varies. This indicates that the performance of the model remains stable against the different attributes selected;
2. regarding the effect of the number of attributes, increasing the number of attributes from 9 to 11 did not cause an increase or decrease in accuracy. In other words, the model is quite resistant to changes in the number of attributes used;
3. the importance of attribute selection can be seen even though the accuracy remains the same, as certain attribute combinations provide deeper insights into the attributes that most influence the prediction. Attributes such as AK, Land Ownership, Family Relationship, and Address often appear in various combinations, so they can be considered to have a significant role in the prediction;
4. in terms of model efficiency, using fewer attributes can achieve the same accuracy as using more attributes. This shows that the complexity of the model can be reduced without degrading the performance, which is beneficial for the application of the model in real situations.

3.2. Discussion

First, the experimental results show that the gradient boosting algorithm has been successful in classifying the data of social assistance recipients of Cicadas village with the best cauration value of 91.67 with 5 different attribute configurations as can be seen in table 22. this is in line with research J. Wang., which states that the gradient boosting algorithm is effectively used for classification models [8].

Secondly, in the research results it can be seen that some attributes have a significant effect on the accuracy of the model. this means that choosing or using only a few attributes has resulted in high accuracy, this can be used as a reference to reduce the amount of data that is too much. The selection of several significant attributes in this experiment is reinforced by research conducted by Zulaikhah which states that the selection of influential attributes will produce more specific and relevant results [12].

Third, based on Table 22, a high accuracy Result of 91.67% is obtained when a 70:30 ratio is used on split data using stratified sampling. This is in line with research conducted by Ananda which states that accuracy results of more than 70% are considered high and proven to be efficient [13].

3. Conclusions

1. The highest accuracy value obtained from using the Gradient Boosting algorithm in the classification of social assistance recipients is 91.67%. This result shows that the Gradient Boosting algorithm is able to achieve an optimal level of accuracy in the classification process.
2. The attributes that have a significant influence on improving the accuracy of the social assistance recipient classification model include Ket KPM, AK, Kepemilikan Tanah, Pekerjaan, Hub.Kel, RW, Alamat, Jenis Kelamin, and Tanggal Lahir. These attributes make a major contribution in improving the performance of the classification model.
3. The split value that affects the accuracy performance of the model is obtained through the use of Stratified Sampling Technique with a data split ratio of 70:30.

References

[1] M. Faid, M. Jasri, and T. Rahmawati, "Perbandingan Kinerja Tool Data Mining Weka Dan Rapidminer Dalam Algoritma Klasifikasi," *Teknika*, vol. 8, no. 1, pp. 11–16, 2019, doi: 10.34148/teknika.v8i1.95.

[2] R. Pahlevi, E. S. Negara, T. Sutabri, and M. I. Herdiansyah, "Penerapan Metode Naive Bayes Untuk Menentukan Klasifikasi Kelayakan Penerimaan Bantuan Rehabilitasi Dan Pembangunan Sekolah Pada Dinas Pendidikan Dan Kebudayaan Kabupaten Banyuasin," *J. Teknol. Inform. Dan Komput.*, vol. 9, no. 2, pp. 1176–1188, 2023, doi: 10.37012/jtik.v9i2.1790.

[3] A. Aribowo, R. Kuswandhie, and Y. Primadasa, "Penerapan Dan Implementasi Algoritma CART Dalam Penentuan Kelayakan Penerima Bantuan PKH Di Desa Ngadirejo," *Cogito Smart J.*, vol. 7, no. 1, pp. 40–51, 2021, doi: 10.31154/cogito.v7i1.293.40-51.

[4] P. Oosterhoff and R. M. Yunus, "The Effects of Social Assistance Interventions on Gender, Familial and Household Relations Among Refugees and Displaced Populations: A Review of the Literature on Interventions in Syria, Iraq, Jordan and Lebanon," 2022, doi: 10.19088/basic.2022.011.

[5] M. Kivrak, "Breast Cancer Risk Prediction With Stochastic Gradient Boosting," *Clin. Cancer Investig. J.*, vol. 11, no. 2, pp. 26–31, 2022, doi: 10.51847/21qrrklo4y.

[6] D. A. Keefer and C. Blake, "The Reproducible Data Reuse (<scp>ReDaR</Scp>) Framework to Capture and Assess Multiple Data Streams," *Proc. Assoc. Inf. Sci. Technol.*, vol. 58, no. 1, pp. 230–240, 2021, doi: 10.1002/pra2.451.

[7] S. Silvianetri, I. Irman, Z. Zubaidah, P. Yeni, and R. Syafitri, "Student Preferences Watching Youtube and Its Implications for Wellbeing Counseling," 2022, doi: 10.4108/eai.11-10-2021.2319487.

[8] J. Wang, J. Xu, C. Zhao, Y. Peng, and H. Wang, "An Ensemble Feature Selection Method for High-Dimensional Data Based on Sort Aggregation," *Syst. Sci. Control Eng.*, vol. 7, no. 2, pp. 32–39, 2019, doi: 10.1080/21642583.2019.1620658.

[9] F. Septian, "Optimasi Klusterisasi pada Lama Tempo Pekerjaan Berbasis Gradient Boost Algorithm," *Indones. J. Inf. Technol.*, 2023.

[10] S. Hosen and R. Amin, "Significant of Gradient Boosting Algorithm in Data Management System," *Eng. Int.*, vol. 9, no. 2, pp. 85–100, 2021, doi: 10.18034/ei.v9i2.559.

[11] N. Endut, W. M. A. F. W. Hamzah, I. Ismail, M. K. Yusof, Y. A. Baker, and H. Yusoff, "A Systematic Literature Review on Multi-Label Classification Based on Machine Learning Algorithms," *Tem J.*, pp. 658–666, 2022, doi: 10.18421/tem112-20.

[12] S. Zulaikhah Hariyanti Rukmana, A. Aziz, and W. Harianto, "Optimasi Algoritma K-Nearest Neighbor (Knn) Dengan Normalisasi Dan Seleksi Fitur Untuk Klasifikasi Penyakit Liver," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 6, no. 2, pp. 439–445, 2022, doi: 10.36040/jati.v6i2.4722.

[13] D. Ananda, A. Pertiwi, and M. A. Muslim, "Journal of Numerical Optimization and Improvement accuracy of gradient boosting in app rating prediction on google playstore," vol. 1, no. 2, 2023.