Fuzzy Integration to Standard Calculation of K-Nearest Neighbour Attributes

M. Adib Al Karomi [1], Ivandari [2]
[1-2] *Computer Science, STMIK Widya Pratama Pekalongan, Indonesia*

Abstract— The development of information and data in the era of the industrial revolution 4.0 is very fast. Researchers, institutions and even industry are competing to find and utilize methods in data processing that are more effective and efficient. In data mining classification, there are several best methods and are widely used by researchers. One of them is K-Nearest Neighbor (KNN). The calculation process in the KNN algorithm is carried out by comparing the testing data to all existing training data. This comparison is generally symbolized by the value of closeness or similarity between attribute records. The KNN method is proven to be good for handling large datasets and datasets with many attributes. One of the drawbacks in calculating the similarity of the KNN is that if there are attributes with a large range value, the similarity value will also be large. Conversely, if the range in an attribute is small, the similarity is also small. This condition is clearly unfair considering the types of attributes in the current data vary widely. One solution to this problem is to use standardization for all existing data attributes. Fuzzy is a model introduced by Prof. Zadeh which allows a faint value to be a value between 1 and 0. In this study the fuzzy model will be integrated in the KNN similarity calculation to obtain standardization of all data attributes. The results show that the use of the KNN algorithm in the classification of credit approval has an accuracy rate of 91.83%.

Index Terms— attribute normalization, fuzzy integration, KNN.

1. Introduction

Big Data is a research theme that is rife in this 4.0 industrial revolution. One process that cannot be separated from big data is the classification process [1]. Such a process can allow previously useless data to become more valuable [2]. Many methods or algorithms can be used in the classification process to handle big data, one of which is the K-Nearest Neighbor [3]. The K-Nearest Neighbor algorithm is included in one of the best classification algorithms and is easy to use [4] [5]. KNN is proven to be able to handle imbalance data [6]. The calculation process in using this algorithm can be said to be very easy to understand and to implement. Broadly speaking, the calculation process using this algorithm is to calculate the closeness or similarity for all training data attributes with the data testing attribute being asked. The proximity value for all existing records is recorded and then the smallest similarity value is taken from all these records. The record with the smallest similarity will be used as the classification result. The k value in the K-Nearest neighbor is the number of nearest records that will be retrieved and used as the result of the classification. If the value of k> 1 then the classification result is decided using the highest yield in the existing k proportion.

Several applications using the basic calculation of the K-Nearest neighbor can be found on the internet. In their calculations, many of the authors use the similarity function and use the k value of 1 (k = 1). The more k values used, the tendency to decrease accuracy and sensitivity performance [7]. Calculations for the classification of prospective students have been carried out and use feature selection because of the many attributes that exist so that it can complicate the calculation process [8]. This process was carried out because many data attributes were deemed unfit to be included in the classification process. One of the main problems that many researchers experience in using the k-nearest neighbor algorithm is when there are several numeric data attributes with a wide range of variants between one attribute and another. This will make the similarity value or closeness value unbalanced between one attribute and another attribute. For example, age attribute with numeric attribute type only has a maximum proximity range of 100 years. Meanwhile, income attributes of the same type can range in proximity to millions or even billions of rupiah. This will lead to inequality if the age difference of 90 years has a similarity value of 90 compared to the difference in income values of hundreds of thousands of rupiah. Actually, the difference in the income value of hundreds of thousands is insignificant when compared to the very high income range.

Fuzzy method was first discovered by prof. Lotfi A. Zadeh and many developed by researchers because of the knowledge and advantages of this method. Until the end of his life, Zadeh continued to develop his knowledge in mathematics and computers. Some developments are carried out using fuzzy logic applications [9]. Several fuzzy logic approaches have also been carried out for other research objects including intelligent systems [10]. The addition of fuzzy in KNN is proven to improve the performance of KNN classification [11]. This study uses fuzzy reasoning to normalize attributes with different ranges.

2. Literature Review

2.1. Related Research

Research using the KNN algorithm has been carried out a lot [12], the previous discussion was to optimize the k parameter for the classification of student heregistration [13] [14], and to increase the accuracy value for the classification of diabetes mellitus[15]. In some studies also use other algorithms such as gain ratio. In several studies, the calculation of KNN was carried out using one method and compared with the improved method. As a result, this method has many weaknesses that can be corrected by adding other methods.

KNN is widely used for credit card fraud detection and is often compared with several other algorithms [16]. In this study, a comparison of the K-Nearest Neighbor method was carried out; Logistic Regression; Machine Learning; Naive Bayes; Support Vector Machine. Knn obtained an accuracy rate of 96.91%.

Optimization of the KNN algorithm is widely used. one of them is using fuzzy logic. The FkNN was used in a data set of 11 million and demonstrated improved accuracy and processing speed [5]. In another study, the fuzzy KNN was used and named Bonferroni Mean Fuzzy KNN (BM-FKNN). This method is proven to improve the accuracy of the KNN classification [11].

2.2. Data Mining

Data Mining, also called Knowledge Discovery in Database (KDD), is a field of science that discusses patterns of data. A process of obtaining knowledge or patterns from data sets is called data mining [17]. Data mining is an extraction process to obtain information that was previously unknown from data [1]. Data mining can analyze old cases to find patterns from data using pattern recognition techniques such as statistics and mathematics [18]. Large data sets can be meaningless if information or knowledge cannot be retrieved. Data mining analyzes the large data then creates a certain rule, pattern, or model to identify new data that is not in the stored data line [2].

Data mining has several functions. Based on the learning method, the data mining function is divided into 2 [19], namely Supervised Learning, Unsupervised Learning. Supervised learning must have sample data or it is often called training data. Meanwhile, unsupervised learning does not require training data. Classification is a data mining function that is classified as supervised learning.

2.2. Classification

Classification is classified as supervised learning because in the classification process there is a learning process with past data. The learning process uses algorithms to recognize patterns from existing or past data which can later be applied to new data that the group does not yet know. In the classification there is a label attribute or it can also be called a destination attribute. Label is an attribute that will be searched for data mining algorithm calculations. In carrying out a classification, past data is needed which will later be processed into a rule or a new knowledge. Classification problems are basically as follows[20]:

1. Classification problems depart from the available training data.
2. The training data will be processed using a classification algorithm.
3. Classification problems end with the production of knowledge which is represented in the form of diagrams, rules or knowledge.

2.2. K-Nearest Neighbor Algorithm

K-Nearest Neighbor (K-NN) [21] is an approach to finding cases by calculating the closeness between new cases and old cases, which is based on matching weights of a number of existing features [22]. K in k-NN is the number of neighbors that will be taken to make decisions.

3. Research Methods

This research was conducted in several structured stages. Figure 1 is the framework that underlies this research. While the stages carried out in this study are as follows:

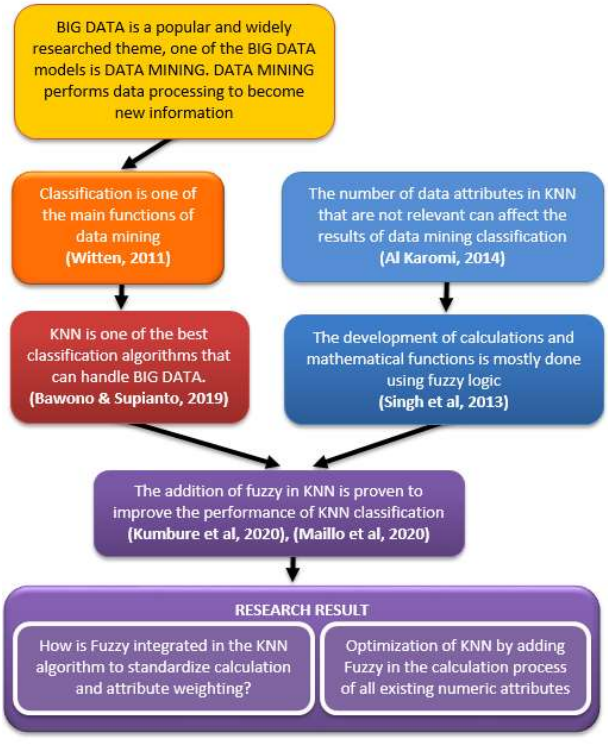

Figure 1. Research framework

3.1. Data Collection

This stage is done by taking a dataset from public data. The data used is a classification dataset with label / learning (supervised learning). The main priority of the data used is data with numeric attribute types. It is possible that the nominal data attribute is also included in the classification process.

3. 2. Data Attribute Analysis

In this process, analysis of all data attributes will be used in the classification process. This analysis process includes sorting out the attributes by numeric type for later fuzzification calculations. Meanwhile, the similarity calculation is done manually for attributes with nominal type

3. 3. Experiment Stage

In the experimental process carried out in several stages, including:

3. 3.1. Fuzzification of data attributes

The fuzzification process is carried out through several stages. The first stage is calculating the existing numeric attributes. This calculation includes recording all numeric attributes and then creating ranges for all attributes. Furthermore, the similarity value between testing data records and training data is divided by the range value for each attribute. This process will make the similarity value only have a range between 0 to 1. Similarity 0 will appear if the training data is the same as the testing data. Meanwhile, the similarity value of 1 will appear if the distance between the training data and the testing data that appears is the farthest distance.

3. 3.2. K-Nearest Neighbor calculations

The next process is a calculation using the K-Nearest neighbor algorithm. This process is actually calculating the average for all the similarity of the existing attributes. This means that if there are 8 attributes in the classification, the similarity value of all these attributes is added and then the distribution of the number of attributes is 8.

3. 3.3. Validation

The validation process is carried out using cross validation. This process is proven to be good and is widely used by researchers in validating classification results. This stage allows the dataset to be divided into 2 parts with one small part as testing data and the other large part as training data. This process is repeated as needed and generally until the entire record is assigned a one-time portion as testing data.

3. 3.3. Testing

The testing process is carried out using a configuration matrix. This stage actually only clarifies the classification calculated by the actual label. The final result of this test is an accuracy obtained from a number of records whose classification corresponds to the actual conditions, divided by the number of existing testing data. Table 1 is a representation of the configuration matrix for 2 kinds of labels or only 2 possibilities.

Table 1. Representation of confusion matrix [23]

Classification		*Predicted class*	
		Class: Yes	*Class: No*
Observed class	*Class: Yes*	***A*** *(True Positive)*	***B*** *(False Negative)*
	Class: No	***C*** *(False Positive)*	***D*** *(True Negative)*

4. Results and Discussion

4. 1. Data Collection

The dataset used has 766 records with 8 data attributes used. All of these attributes include: the name of the customer, gender, age, loan amount, term, monthly installments, nominative balance, principal arrears, interest arrears, and credit status. This data is data on credit card customers from a finance company in Indonesia. Table 2 is the metadata from the dataset prior to analysis and adjustment of records.

4. 2. Data Attribute Analysis

Of the total 766 data records, there were 19 records with age entry errors. These errors include customers aged 0, 1, 2, and 3 years. And there are some age errors up to thousands of years. Obviously this is logically impossible because toddlers may not be able to advance credit. Likewise, customers with thousands of years of age cannot possibly exist and it is clear that data input errors have occurred. Therefore, these 19 records will be eliminated and will not be used in the next process. Table 3 is the metadata of the dataset that will be used with 747 records.

Table 2. Initial dataset metadata

Role	Nama atribut	Tipe atribut	Statistics	Range	Missings
id	name	polynominal	mode = x1 (1), least = x1 (1)	x1 (1), x10 (1), x100 (1), x101 (1), x102 (1), x103 (1), x104 (1), x105 (1), x106 (1), x107 (1), x108 (1), x109 (1), x11 (1), x110 (1), x111 (1), x112 (1), x113 (1), x114 (1), x115 (1), x116 (1), x117 (1), x118 (1), x119 (1), x12 (1), x120 (1), x121 (1), ... and 716 more ... , x766 (1), x77 (1), x78 (1), x79 (1), x8 (1), x80 (1), x81 (1), x82 (1), x83 (1), x84 (1), x85 (1), x86 (1), x87 (1), x88 (1), x89 (1), x9 (1), x90 (1), x91 (1), x92 (1), x93 (1), x94 (1), x95 (1), x96 (1), x97 (1), x98 (1), x99 (1)	0.0
label	credit status	binominal	mode = Positive (556), least = Negative (210)	Positive (556), Negative (210)	0.0
regular	gender	binominal	mode = Female (462), least = Male (304)	Female (462), Male (304)	0.0
regular	age	numeric	avg = 29.161 +/- 263.166	[-7162.000 ; 1043.000]	1.0
regular	loan amount	numeric	avg = 2712482.631 +/- 9995602.067	[83333.330 ; 228655000.000]	0.0
regular	period of time	numeric	avg = 18.961 +/- 32.076	[1.000 ; 679.000]	0.0
regular	number of installments per month	numeric	avg = 233391.702 +/- 548968.221	[0.000 ; 10350000.000]	0.0
regular	nominative balance	numeric	avg = 2007385.712 +/- 8711282.360	[-4000000.000 ; 209404092.000]	0.0
regular	principal arrears	numeric	avg = 790085.298 +/- 4139216.644	[0.000 ; 91612122.240]	0.0
regular	Interest arrears	numeric	avg = 87717.084 +/- 568231.776	[0.000 ; 11000000.000]	0.0

Table 3. Metadata edit dataset

Role	Nama atribut	Tipe atribut	Statistics	Range	Missings
id	name	polynominal	mode = x1 (1), least = x1 (1)	x1 (1), x10 (1), x100 (1), x101 (1), x102 (1), x103 (1), x104 (1), x105 (1), x106 (1), x107 (1), x108 (1), x109 (1), x11 (1), x110 (1), x111 (1), x112 (1), x113 (1), x114 (1), x115 (1), x116 (1), x117 (1), x118 (1), x119 (1), x12 (1), x120 (1), x121 (1), ... and 697 more ... , x766 (1), x77 (1), x78 (1), x79 (1), x8 (1), x80 (1), x81 (1), x82 (1), x83 (1), x84 (1), x85 (1), x86 (1), x87 (1), x88 (1), x89 (1), x9 (1), x90 (1), x91 (1), x92 (1), x93 (1), x94 (1), x95 (1), x96 (1), x97 (1), x98 (1), x99 (1)	0.0
label	credit status	binominal	mode = Positive (539), least = Negative (208)	Positive (539), Negative (208)	0.0
regular	gender	binominal	mode = Female (450), least = Male (297)	Female (450), Male (297)	0.0
regular	Age (Fuzzy)	numeric	avg = 0.514 +/- 0.108	[0.000 ; 1.000]	0.0
regular	loan amount (Fuzzy)	numeric	avg = 0.976 +/- 0.058	[0.000 ; 0.999]	0.0
regular	period of time (Fuzzy)	numeric	avg = 0.972 +/- 0.048	[0.000 ; 0.999]	0.0
regular	number of installments per month (Fuzzy)	numeric	avg = 0.978 +/- 0.047	[0.000 ; 1.000]	0.0
regular	nominative balance (Fuzzy)	numeric	avg = 0.978 +/- 0.056	[0.000 ; 1.050]	0.0
regular	principal arrears (Fuzzy)	numeric	avg = 0.988 +/- 0.045	[0.000 ; 1.000]	0.0

regular	name (Fuzzy)	numeric	avg = 0.993 +/- 0.046	[0.000 ; 1.000]	0.0

4.3. Research result

4.3.1. Fuzzification Dataset

This process allows attributes with numeric types to have an equivalent range to one another, namely between 0 and 1. The tool used in this process is a spreadsheet application. In the process, the formula used is as follows:

$$Value_{Fuzzy} = \frac{maximum\ value\ (atribute) - record\ value}{maximum\ value\ (attribute)} \quad (1)$$

As one example, the principal arrears attribute contains between 0 and 57,000,000. the customer with the largest arrears amounted to 57,000,000. Fuzzy standardization process is done by changing all data records with calculations and standardization as follows:

Customers with 0 principal arrears will be calculated:

$$N_0 = \frac{57000000-0}{57000000} = 1 \quad (2)$$

Meanwhile, customers with a principal amount of 950000 in arrears will be calculated as follows:

$$N_{950000} = \frac{57000000-9}{57000000} = 0.983333 \quad (3)$$

The more the value of the principal arrears will make the standardized value with this fuzzy approach the value of 0. In another example, a customer with a principal arrears value of 26200000 (twenty six million two hundred thousand) will get the standardized value as follows:

$$N_{26200000} = \frac{57000000-26200000}{57000000} = 0.54 \quad (4)$$

The results of this calculation will show a value of 1 for customers with the smallest principal arrears and a value of 0 for customers with the largest / maximum principal arrears. In this calculation process, the value exchange process must be carried out by reversing the values 0 and 1. This is done to normalize the value of the principal arrears. Table 4 is the dataset metadata that has been fuzzified. Fuzzification is performed only for numeric attributes.

Table 4. Metadata dataset after fuzzification

Role	Nama atribut	Tipe atribut	Statistics	Range	Missings
id	name	polynominal	mode = x690 (1), least = x690 (1)	x1 (1), x10 (1), x100 (1), x101 (1), x102 (1), x103 (1), x104 (1), x105 (1), x106 (1), x107 (1), x108 (1), x109 (1), x11 (1), x110 (1), x111 (1), x112 (1), x113 (1), x114 (1), x115 (1), x116 (1), x117 (1), x118 (1), x119 (1), x12 (1), x120 (1), x121 (1), ... and 697 more ... , x766 (1), x77 (1), x78 (1), x79 (1), x8 (1), x80 (1), x81 (1), x82 (1), x83 (1), x84 (1), x85 (1), x86 (1), x87 (1), x88 (1), x89 (1), x9 (1), x90 (1), x91 (1), x92 (1), x93 (1), x94 (1), x95 (1), x96 (1), x97 (1), x98 (1), x99 (1)	0.0
label	credit status	binominal	mode = Positive (539), least = Negative (208)	Positive (539), Negative (208)	0.0
regular	gender	binominal	mode = Female (450), Male = L (297)	Female (450), Male (297)	0.0
regular	Age (Fuzzy)	numeric	avg = 0.486 +/- 0.108	[0.000 ; 1.000]	0.0
regular	loan amount (Fuzzy)	numeric	avg = 0.024 +/- 0.058	[0.001 ; 1.000]	0.0
regular	period of time (Fuzzy)	numeric	avg = 0.028 +/- 0.048	[0.001 ; 1.000]	0.0
regular	number of installments per month (Fuzzy)	numeric	avg = 0.022 +/- 0.047	[0.000 ; 1.000]	0.0
regular	nominative balance (Fuzzy)	numeric	avg = 0.022 +/- 0.056	[-0.050 ; 1.000]	0.0
regular	principal arrears (Fuzzy)	numeric	avg = 0.012 +/- 0.045	[0.000 ; 1.000]	0.0
regular	name (Fuzzy)	numeric	avg = 0.007 +/- 0.046	[0.000 ; 1.000]	0.0

4.3.2. K-Nearest Neighbor calculations

The process of calculating the KNN is carried out using a spreadsheet tool using the final dataset that has been previously processed. In the dataset, there is one attribute with nominal type, namely gender. Special care is taken for this attribute by creating auxiliary tables. Table 5 is a table to help the proximity of attribute values for gender.

Table 5. The value of the closeness of the gender attribute

	Male	Female
Male	0	1
Female	1	0

From table 5 it can be seen that the value of the closeness of the gender attribute is only between 1 or 0. This value is raised considering that the gender attribute type is binominal which only allows 2 variants. The value 0 appears if the contents of the attribute are the same, conversely if the contents of the attribute are different then the value is 1.

Next, the testing data collection or test data is done randomly. This process is a validation process which will be discussed in the following 4.3.3 discussion. The classification results are calculated using the KNN then adjusted for the classification label to the actual label. This process is a testing process to calculate the accuracy value of an algorithm. The full results are presented in the following 4.3.4 discussion.

4.3.3. Validation Results

The validation process is carried out using the rapid miner application. This process is done using x-validation. In this process the dataset is divided into 10 parts with 1 part used as testing data, the remaining 9 parts become training data. This process is repeated up to 10 times until all data records have one right to become testing data. This kind of process is widely used by researchers to get optimal results. This process is also known as 10 folds cross validation. Figure 2 is a representation of 10 folds cross validation.

Figure 2. Representation of 10 folds cross validation

Figure 3 is a worksheet view of the rapid miner application with 3 datasets and 3 validations using the 10 folds cross validation method. While Figure 4 is a display of the K-Nearest Neighbor algorithm testing process using a configuration matrix.

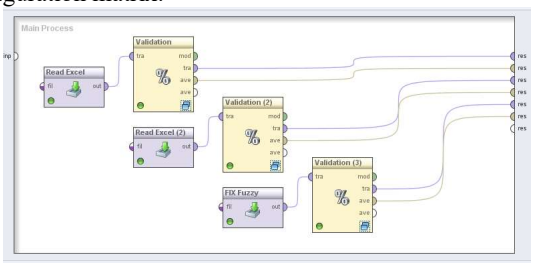

Figure 3. The rapid miner worksheet

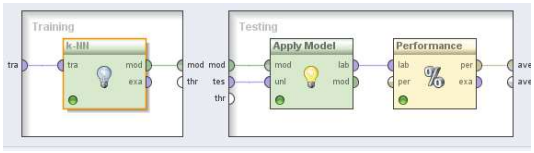

Figure 4. KNN Testing

4.3.3. Result Testing

The results showed that the accuracy level of the KNN algorithm in the classification of credit approval was 91.83%. These results were obtained using a k value of 1. Table 6 is a configuration matrix table of the research results.

Table 6. Research results

	True Positive	True Negative	class precision
Pred. Positive	**511**	31	94.28%
Pred. Negative	30	**175**	85.37%
class recall	94.45%	84.95%	

The process of calculating the accuracy value is the number of predictions that match the original label divided by the entire data record. In this study, the prediction of jams with the original label was jammed as many as 511. Current predictions with the original label were 175. with a total dataset of 747 records. This means that the level of accuracy can be calculated by (511 + 175) / 747. and the result is 91.83%.

5. Conclusion

Credit approval classification using K-Nearest Neighbor was successfully carried out with an accuracy rate of 91.83%. These results were obtained by cleaning the dataset by removing data records with invalid age fields. Fuzzy integration is carried out to standardize attributes with values that are not balanced with each other. For calculations using the rapid miner auxiliary application, fuzzy integration does not really affect the calculation results. As for manual calculations or using spreadsheets, attribute fuzzification greatly facilitates the calculation process.

6. Acknowledgment

This research is part of a research funded by DIKTI in a Beginners Lecturer Research Scheme for fiscal year 2020.

References

[1] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques 3rd Edition*, vol. 40, no. 6. Elsevier, 2011.

[2] E. Prasetyo, *Data Mining Konsep dan Aplikasi menggunakan Matlab*. Yogyakarta: Andi Offset, 2012.

[3] A. H. Bawono and A. A. Supianto, "Efisiensi Klasifikasi Big Data Menggunakan Improved Neighbour," vol. 6, no. 6, pp. 1–6, 2019, doi: 10.25126/jtiik.201962085.

[4] X. Wu, *The Top Ten Algorithms in Data Mining*. New York: Taylor & Francis Group, LLC, 2009.

[5] J. Maillo, S. Garcia, J. Luengo, F. Herrera, and I. Triguero, "Fast and Scalable Approaches to Accelerate the Fuzzy k-Nearest Neighbors Classifier for Big Data," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 5, pp. 874–886, 2020, doi: 10.1109/TFUZZ.2019.2936356.

[6] X. Zhang *et al.*, "Balancing large margin nearest neighbours for imbalanced data," *J. Eng.*, vol. 2020, no. 13, pp. 316–321, 2020, doi: 10.1049/joe.2019.1178.

[7] ikhsan wisnuadji Gamadarenda and I. Waspada, "Implementasi Data Mining Untuk Deteksi Penyakit Ginjal Kronis (Pgk) Menggunakan K-Nearest Neighbor (Knn) Dengan Backward Elimination," vol. 7, no. 2, pp. 417–426, 2018, doi: 10.25126/jtiik.202071896.

[8] M. A. Al Karomi, M. R. Maulana, S. J. Prasetiyono, Ivandari, and Arochman, "Strengthening campus finance by analyzing attribute attributes for student registration classifications." p. 1, 2019, [Online]. Available: https://jurnal.polines.ac.id/index.php/jaict/article/view/1431.

[9] H. Singh *et al.*, "Real-Life Applications of Fuzzy Logic," vol. 2013, 2013.

[10] F. Sets, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic," vol. 90, pp. 111–127, 1997.

[11] M. Mailagaha Kumbure, P. Luukka, and M. Collan, "A new fuzzy k-nearest neighbor classifier based on the Bonferroni mean," *Pattern Recognit. Lett.*, vol. 140, pp. 172–178, 2020, doi: 10.1016/j.patrec.2020.10.005.

[12] Y. Zhang, J. Wu, J. Wang, and C. Xing, "A Transformation-Based Framework for KNN Set Similarity Search," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 3, pp. 409–423, 2020, doi: 10.1109/TKDE.2018.2886189.

[13] M. A. Al Karomi, "Optimasi Parameter K pada Algoritma KNN untuk Klasifikasi heregistrasi mahasiswa Program Studi Teknik Informatika STMIK Widya Pratama Jl . Patriot 25 Pekalongan Email : adib.comp@gmail.com," *IC-TECH*, vol. X, no. 0285, p. 5, 2015.

[14] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, "Efficient kNN classification with different numbers of nearest neighbors," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 29, no. 5, pp. 1774–1785, 2018, doi: 10.1109/TNNLS.2017.2673241.

[15] I. Indrayanti, S. Devi, and M. A. Al Karomi, "Peningkatan Akurasi Algoritma KNN dengan Seleksi Fitur Gain Ratio untuk Klasifikasi Penyakit Diabetes Mellitus," *IC-TECH*, vol. XIII, no. 2, pp. 1–6, 2017.

[16] O. Adepoju, J. Wosowei, S. Lawte, and H. Jaiman, "Comparative Evaluation of Credit Card Fraud Detection Using Machine Learning Techniques," *2019 Glob. Conf. Adv. Technol. GCAT 2019*, pp. 1–6, 2019, doi: 10.1109/GCAT47503.2019.8978372.

[17] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques 3rd Edition*. Elsevier, 2011.

[18] D. T. Larose, *Discovering Knowledge in Data: an Introduction to Data Mining*. John Wiley & Sons, 2005.

[19] B. Santosa, *Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis*, Edisi Pert. Yogyakarta: Graha Ilmu, 2007.

[20] S. Susanto and D. Suryadi, *Pengantar Data Mining: Menggali Pengetahuan dari Bongkahan Data*. Yogyakarta: Andi Offset, 2010.

[21] T. M. Cover and P. E. Hart, "Nearest Neighbor Pattern Classification," vol. I, 1967.

[22] Kusrini and L. E. Taufiq, *Algoritma Data Mining*. Yogyakarta: Andi Offset, 2009.

[23] F. Gorunescu, *Data Mining: Concepts; Models and Techniques*. Springer, 2011.