Comparison of performance of k-nearest neighbor algorithm using smote and k-nearest neighbor algorithm without smote in diagnosis of diabetes disease in balanced data

A G Pertiwi, N Bachtiar, R Kusumaningrum, I Waspada, A Wibowo

1Department of Informatics, Faculty of Science and Mathematics, Diponegoro University
Jl. Prof Soedarto, SH, Ngesrep, Semarang 50275, Indonesia

Abstract. According to the Indonesian Health Profile in 2017, diabetes is one of the causes of death for almost 70% in the world. The high mortality rate induces the need for making the effort to reduce the number of people with diabetes by conducting studies that lead to making a diagnosis so that can detect a person with diabetes accurately. This study tries to compare the performance of the K-Nearest Neighbors algorithm using Synthetic Minority Over-sampling Technique and the K-Nearest Neighbors algorithm without Synthetic Minority Over-sampling Technique in diagnosing diabetes on imbalanced datasets. The parameters tested are the k value of the K-Nearest Neighbors and Synthetic Minority Over-sampling Technique. The testing is carried out using the K-Fold Cross Validation strategy. The data used in this study were 3876 data from Pertamina Central Hospital. Based on the results of tests conducted, it shows that the value of accuracy produced in diagnosing diabetes by using Synthetic Minority Over-sampling Technique is better than the accuracy produced without using Synthetic Minority Over-sampling Technique with the highest accuracy increase of 8.25%. The highest average accuracy is obtained when the value of k = 3 in the K-Nearest Neighbors, k = 5 in the Synthetic Minority Over-sampling Technique, and fold = 10, which reaches 78.06%.

1. Introduction
Cancer Based on the Indonesian Health Profile, in 2013 diabetes was the disease with the sixth highest number of sufferers in Indonesia with a proportion of 4.8% under hypertension, arthritis, stroke, dental and mouth problems, and chronic obstructive pulmonary disease. In addition, diabetes is also one of the causes of death of almost 70% worldwide [1]. Therefore, efforts should be made to reduce the number of people with diabetes by conducting studies that lead to making a diagnosis so that it can be detected if someone has diabetes, so prevention and treatment can be done for people with diabetes with a comprehensive approach. One way is to use machine learning as a learning technique.

K-Nearest Neighbors algorithm is one algorithm that is commonly used for classification, but it can also be used for estimation and prediction [2]. The K-Nearest Neighbor algorithm classifies new data that is not yet known by its class by selecting k data that is located closest to the new data. The class with the most frequency from the nearest k data will be chosen as the predicted class for new data. In general, the value of k uses an odd number so that there is no equal distance in the classification process. The distance or proximity of neighbors is calculated using Euclidean distance [3]. The K-Nearest Neighbor algorithm has several advantages, which are simple and easy to implement, and effective when
the training data is large, so in this study the K-Nearest Neighbor algorithm is chosen for classification. However, the K-Nearest Neighbor algorithm is not equipped with the ability to work on unbalanced datasets [4].

SMOTE (Synthetic Minority Over-sampling Technique) is one of the over-sampling techniques that is often used to deal with imbalanced datasets by making synthetic data in minor data classes so that the data becomes balanced [5] and is expected impact on better classification performance. In some study cases, there has been an increase in performance (in this case accuracy) from around 65% for initial data distribution to around 80% because SMOTE has done so that the initially unbalanced data becomes really balanced [6]. The SMOTE algorithm can also be used to help K-Nearest Neighbor algorithms that cannot handle unbalanced dataset cases. Thus, in this study we trying to compare the performance of the K-Nearest Neighbors algorithm using Synthetic Minority Over-sampling Technique and the K-Nearest Neighbors algorithm without Synthetic Minority Over-sampling Technique in diagnosing diabetes on imbalanced datasets.

2. Methods

2.1. Synthetic minority over-sampling technique

One approach to handling unbalanced data cases is the Synthetic Minority Over-sampling Technique (SMOTE), which was first introduced by Nithes V. Chawla[5]. Synthetic Minority Over-sampling Technique is one of the derivatives of over-sampling. If the over-sampling method is principled to reproduce data at random, it is different from SMOTE. This approach works by adding data by making replication from minority data. Replication is known as synthetic data (synthetic data). The SMOTE method works by finding k-nearest neighbor (ie k nearest data as much as k) for each data in the minority class, after which synthetic data is made as much as the desired percentage of duplication between the minor and k-nearest neighbor chosen at random [4]. Illustration of data distribution after applying SMOTE can be seen in Figure 1.

![Figure 1. SMOTE illustration](image_url)

2.2. K-nearest neighbor

K-Nearest Neighbor algorithm is an algorithm that classifies objects that have the closest distance to the object. The K-Nearest Neighbor algorithm classifies new data that is not yet known by its class by selecting k data that is located closest to the new data. The most class frequency from the nearest data number of k will be chosen as the predicted class for new data. In general, the value of k uses an odd number so that there is no equal distance in the classification process. The distance or proximity of neighbors is calculated using Euclidean distance [3].

2.3. K-fold cross validation

K-Fold Cross Validation divides the data by dividing the dataset structurally into k subsets (subsets) with each part having the same amount of data. Data that has been shared is then repeated or iterated k times. One partition becomes the test data, while the remaining partition (k - 1) becomes the training data as well as the next iteration [7]. The k value that is often used in study is usually a value of 5 and a value of 10 [8].

3. Methods
This study is done by doing some steps which are collecting data, preprocessing data, do and not doing SMOTE, dividing data into training and testing data, testing with K-Nearest Neighbor, and evaluation. The overall methods can be seen from Figure 2.

3.1. Collecting data
In the first stage, which is the data collection stage, the dataset is obtained by searching for the diabetes dataset. From the diabetes dataset obtained, it will be counted how many patients have diabetes and do not have diabetes. Then the dataset will go through a preprocessing process at a later stage.

3.2. Preprocessing
In the second step, which is preprocessing data, there are three main processes: data cleaning, data selection, and data transformation. Cleaning data is done to remove inconsistent data and noise data in the dataset. Data selection is done to choose attributes that are relevant as the base of analysis to do mining process. Data transformation is done so that the learning process done in this study can process the inputs correctly. One of the process inside it is data normalization which is used to transform data into values with new range so that no data is too high or too low.

3.3. Synthetic minority over-sampling technique
In third step, there are two choices: do Synthetic Minority Over sampling Technique (SMOTE), or not doing it. If there’s no SMOTE, then dataset which are used are base dataset with unbalanced data count between positive and negative. SMOTE method aims to balance the data count which is unbalanced so
that it will get higher accuracy value. This study compares accuracy value between doing SMOTE and not doing SMOTE.

3.4. Testing
Testing step consist of doing analysis about processing training and testing diabetes dataset using K-Nearest Neighbor. In this study, we did two types of testing. The first one is using dataset which has been over-sampled using SMOTE, and the second one is using dataset which hasn’t been over-sampled using SMOTE.

3.5. Evaluation
In the last step, we did evaluation using confusion matrix as shown in Table 1 to get accuracy value, sensitivity value, and specificity value. Evaluation process consist of two types: evaluation on dataset which has been over-sampled using SMOTE, and dataset which hasn’t been over-sampled using SMOTE.

| Actual Class | Predicted Class |
|--------------|----------------|
| (+)          | True Positive (TP) |
| (-)          | False Positive (FP) |
| (-)          | False Negative (FN) |
| (+)          | True Negative (TN) |

### 4. Results and discussion

4.1. Study data
This study uses a dataset obtained from Pertamina Central Hospital with a total of 3876 data. The data consists of 1435 data for classes that are positively affected by diabetes and 2441 data for classes that are negative for diabetes, and have already passed the preprocessing stage. This data is then over-sampled using SMOTE, which is the addition of minority class data so that the data that were initially unbalanced become balanced. After that, the data is distributed on the dataset to be used as training data and test data.

4.2. Scenario
Here are the scenarios used in this study:

1. Scenario 1 : this scenario is done to find accuracy, sensitivity, specificity, from K-Nearest Neighbor in diagnosing diabetes in unbalanced data. Parameter that were tested are k value in KNN which consist of 3, 5, 7, 9, and fold value 5 and 10.
2. Scenario 2 : this scenario is done to find accuracy, sensitivity, specificity from KNN algorithm in diagnosing diabetes in unbalanced data, which have been done SMOTE so that the data become balanced. Parameters tested were k value in KNN which consist of 3, 5, 7, 9, k value in SMOTE which consist of 3 and 5, and also fold value 5 and 10.
3. Scenario 3 : in this scenario, comparison is done between scenario 1 and scenario 2. The best parameter will be used in scenario 4.
4. Scenario 4 : In this scenario, we did a diagnosis of diabetes using new testing data.

4.3. Results and analysis
1. Scenario 1 : Based on the results of scenario 1 can be seen that the value of k = 9 shows the highest accuracy value, both at the value of fold = 5 and the value of fold = 10. At the value of fold = 5 the highest accuracy is 73.09% which illustrates the ability of the algorithm to be able to do correctly classified, a sensitivity value of 59.51% which illustrates the ability of the algorithm to correctly detect patients who are positive with diabetes, and a specificity value of 81.07% which describes the ability of the algorithm to precisely detect patients who are negative with diabetes. At fold value =
10, the highest accuracy is 72.55%, sensitivity value is 58.28%, and specificity value is 80.95%. While the value of k in KNN = 3 produces the lowest accuracy value, both at the value of fold = 5 and the value of fold = 10. The results of scenario 1 can be seen in Table 2:

| kfold | k KNN value | Average Accuracy | Average Sensitivity | Average Specificity |
|-------|-------------|------------------|---------------------|---------------------|
| 5     | 3           | 70,25            | 54,91               | 79,27               |
|       | 5           | 71,85            | 57,42               | 80,33               |
|       | 7           | 72,57            | 58,05               | 81,11               |
|       | 9           | 73,09            | 59,51               | 81,07               |
| 10    | 3           | 69,81            | 56,03               | 77,92               |
|       | 5           | 71,80            | 57,57               | 80,17               |
|       | 7           | 72,21            | 57,37               | 80,95               |
|       | 9           | 72,55            | 58,28               | 80,95               |

2. Scenario 2: Based on the results of scenario 2, it can be seen that at the fold value = 5 the highest accuracy is obtained when the value of k in SMOTE = 5 and the value of k in KNN = 5, which is 77.57%, with a sensitivity value of 82.92% and a specificity value of 72.22%. Whereas at the fold value = 10 the highest accuracy is obtained when the value of k in SMOTE = 5 and the value of k in KNN = 3, which is 78.06%, with a sensitivity value of 84.07% and a specificity value of 72.06%. So the results of scenario 2 shows that the value of k in SMOTE has an influence on the results of accuracy in diagnosing diabetes with the highest accuracy is achieved when the value of k in SMOTE = 5. The k value the SMOTE then will be used in scenario 3 to be compared with scenario 1. The results of scenario 2 can be seen in Table 3.

| Kfold Value | k SMOTE value | k KNN value | Average Accuracy | Average Sensitivity | Average Specificity |
|-------------|---------------|-------------|------------------|---------------------|---------------------|
| 5           | 3             | 76,98       | 83,57            | 70,38               |
|             | 5             | 76,67       | 82,83            | 70,50               |
|             | 7             | 76,26       | 82,34            | 70,18               |
|             | 9             | 76,05       | 81,19            | 70,91               |
|             | 3             | 77,45       | 83,41            | 71,49               |
| 5           | 5             | 77,57       | 82,92            | 72,22               |
|             | 7             | 77,00       | 82,22            | 71,77               |
|             | 9             | 77,22       | 82,06            | 72,39               |
|             | 3             | 78,02       | 84,88            | 71,16               |
|             | 5             | 77,41       | 83,65            | 71,16               |
|             | 7             | 77,04       | 83,29            | 70,79               |
|             | 9             | 76,53       | 81,44            | 71,61               |
| 10          | 3             | 78,06       | 84,07            | 72,06               |
|             | 5             | 77,78       | 83,65            | 71,89               |
|             | 7             | 77,94       | 83,41            | 72,47               |
|             | 9             | 77,47       | 82,47            | 72,47               |
3. Scenario 3: Based on the results of scenario 3 it can be seen that the highest accuracy of scenario 1 is at the value of \(k = 9\) and the value of fold = 5, which is 73.09\%, while the highest accuracy of scenario 2 is at the value of \(k = 3\) and the value of fold = 10, which is 78.06\%. The highest sensitivity of scenario 1 is at the value of \(k = 9\) and the value of fold = 5, which is 59.51\%, while the highest sensitivity of scenario 2 is at the value of \(k = 3\) and the value of fold = 10, which is equal to 84.07\%. The highest specificity of scenario 1 is at the value of \(k = 7\) and the value of fold = 5, that is equal to 81.11\%, while the highest specificity of scenario 2 is at the value of \(k = 7, k = 9\), and the value of fold = 10, amounting to 72.47\%. So that the highest accuracy obtained from scenario 3 is on the parameter value \(k = 3\) on KNN, \(k = 5\) on SMOTE, and the value fold = 10. The parameter is then used for scenario 4, which is testing a number of new data. The comparison graph of the accuracy of scenario 3 can be seen in Figure 3, the comparison graph for scenario 3 sensitivity can be seen in Figure 4, and the comparison graph for the specificity of scenario 3 can be seen in Figure 5.

![Figure 3. Accuracy comparison graph on scenario 3](image3)

![Figure 4. Sensitivity comparison graph on scenario 3](image4)
The results of scenario 3 also show that there is an increase in the average accuracy of each change in the value of k. The highest accuracy improvement is in the parameter value k = 3 in KNN and fold value = 10 with an increase in accuracy of 8.25%. Details on increasing the accuracy of each k value can be seen in Table 4.

**Table 4. Accuracy Improvements on Scenario 3**

| Kfold value | K-NN value | Scenario 1 | Scenario 2 with k=5 in SMOTE | Accuracy Improvements |
|-------------|------------|------------|-------------------------------|-----------------------|
|             | 3          | 70.25%     | 77.45%                        | 7.20%                 |
|             | 5          | 71.85%     | 77.57%                        | 5.72%                 |
|             | 7          | 72.57%     | 77.00%                        | 4.43%                 |
|             | 9          | 73.09%     | 77.22%                        | 4.13%                 |
|             | 3          | 69.81%     | 78.06%                        | 8.25%                 |
|             | 5          | 71.80%     | 77.78%                        | 5.98%                 |
|             | 7          | 72.21%     | 77.94%                        | 5.73%                 |
|             | 9          | 72.55%     | 77.47%                        | 4.92%                 |

4. Scenario 4: In scenario 4, the objective is to test the new data using the best parameters from the results of scenario 3. The new data used are 20 data. The results of scenario 4 experiments are in the form of a confusion matrix as shown in Table 5.

**Table 5. Confusion Matrix Scenario 4**

| Actual Class | Predicted Class |       |       |
|--------------|----------------|-------|-------|
| (+)          | (+)            | 9     | 1     |
| (-)          | (-)            | 2     | 8     |

Table 5 shows that of the 20 new data tested against the test scenario with the best parameters obtained, the value of k = 3 in KNN, k value = 5 in SMOTE, and the value of fold = 10 produces a
lot of data that are in the right conditions with total of 17 data, which is the sum of the exact conditions in the class that was diagnosed with diabetes as much as 9 data, and the exact condition in the class that was diagnosed with diabetes as much as 8 data. In addition, there are 3 data that are incorrectly classified, 1 data predicted as negative where the actual is positive and there are 2 data which classified being positive where actually are negative. Based on the explanation above, it can be calculated the value of accuracy, sensitivity, and specificity of the results of scenario 4 as follows:

\[
\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{9 + 8}{9 + 8 + 1 + 2} = 85\%
\]

\[
\text{sensitivity} = \frac{TP}{TP + FN} = \frac{9}{9 + 1} = 90\% 
\]

\[
\text{specificity} = \frac{TN}{TN + FP} = \frac{8}{8 + 2} = 80\%
\]

Then the results of scenario 4 produce an accuracy value of 85%, a sensitivity value of 90%, and a specificity value of 80% in diagnosing diabetes using 20 new data. So, 90% of the condition can predict whether new data is diagnosed with diabetes, and 80% of the condition can predict whether data is not diagnosed with diabetes.

5. Conclusion

In this paper, we compare the performance of the K-Nearest Neighbors algorithm using Synthetic Minority Over-sampling Technique and the K-Nearest Neighbors algorithm without Synthetic Minority Over-sampling Technique in diagnosing diabetes on imbalanced datasets. Based on the study, changes in the value of k on K-Nearest Neighbor and SMOTE affect the average value of accuracy, sensitivity, and specificity in diagnosing diabetes with the highest accuracy value of 78.06%. The value is obtained with the parameter value k = 3 on K-Nearest Neighbor, k value = 5 on SMOTE, and fold value = 10. The accuracy produced in diagnosing diabetes using KNN with SMOTE is better than the accuracy produced using KNN without SMOTE for unbalanced dataset cases. Testing the K-Nearest Neighbor algorithm with over-sampling using the Synthetic Minority Over-sampling Technique method of 20 new data with the best parameters resulting in an average value of 85% accuracy, an average sensitivity value of 90%, and an average value of specificity by 80%.

References

[1] Kemenkes RI 2018 Profil Kesehatan Indonesia Tahun 2017. Jakarta: Kementrian Kesehatan Republik Indonesia
[2] Larose D T 2005 Discovering Knowledge in Data: An Introduction to Data Mining. Canada: John Willey & Sons Inc.
[3] Shofia E N, Putri R R, & Arwan A 2017 Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer 1 426-435
[4] Siringoringo R 2018 Jurnal ISD 3
[5] Chawla N V, Bowyer K, Hall L and Kegelmeyer W 2002 Journal of Artificial Intelligence Research 321-357
[6] Pears R, Finlay J, & Connor A M 2014 Synthetic Minority Over-sampling TTechnique (SMOTE) for Predicting Software Build Outcomes arXiv.
[7] Kohavi R 1995 International Joint Conference on Artificial Intelligence (IJCAI) 1137-1145.
[8] Hermawan, & Yoannita 2018 JTKSI 01 22-25