Formosa Journal of Science and Technology (FJST)
Vol. 4, No.1, 2025: 337-348

Classification of Autoimmune Diseases Using the K-Nearest Neighbors Algorithm

Resti Amalia[1], Ahmad Faiz Zaidan[2], Syahrul Ramadhan[3], Farhan Septian[4], Ananta Mikail Aqsha[5], Perani Rosyani[6*]
Universitas Pamulang
Corresponding Author: Perani Rosyani dosen00837@unpam.ac.id

A R T I C L E I N F O

Keywords: Autoimmune diseases, K-Nearest Neighbors, Classification, Machine Learning, Optimization

Received : 3, January
Revised : 17, January
Accepted: 31, January

©2025 Amalia, Zaindan, Ramadhan, Septian, Aqsha, Rosyani: This is an open-access article distributed under the terms of the Creative Commons Atribusi 4.0 Internasional.

A B S T R A C T

Autoimmune diseases occur when the immune system attacks the body's own tissues, causing serious complications and overlapping symptoms that challenge early detection. This study reviews the use of the K-Nearest Neighbors (K-NN) algorithm for classifying autoimmune diseases through a systematic literature review of five articles. Compared to methods like Genetic Algorithms, Support Vector Machines (SVM), and Single Layer Perceptrons (SLP), K-NN shows high accuracy when optimal parameters and neighbor counts are used. However, challenges include sensitivity to imbalanced data and high computational demands for large datasets. Combining K-NN with optimization techniques, such as Genetic Algorithms, enhances accuracy and stability. The study concludes that K-NN is effective for classifying autoimmune diseases, especially with hybrid approaches, and recommends further research with larger datasets.

ISSN-E: 2964-6804
DOI: https://doi.org/10.55927/fjst.v4i1.13443
https://journal.formosapublisher.org/index.php/fjst

INTRODUCTION

Autoimmune diseases are a group of chronic health disorders that occur due to dysfunction of the immune system. In these conditions, the immune system, which is supposed to protect the body, attacks the body's own tissues. Some common autoimmune diseases include lupus, rheumatoid arthritis and multiple sclerosis. These diseases often have symptoms that overlap with other illnesses, such as fatigue, joint pain and skin rashes, which causes great challenges in the diagnosis process.

Early diagnosis is crucial to prevent serious complications from autoimmune diseases. However, early identification is difficult through clinical examination alone due to the non-specific symptoms. Therefore, computational approaches, such as machine learning, are being applied to assist in the diagnosis and classification of autoimmune diseases. One of the widely used machine learning algorithms is K-Nearest Neighbors (K-NN).

The main objectives of this study are to assess the effectiveness of the K-NN algorithm in autoimmune disease classification using an image dataset, to analyze the advantages and disadvantages of the K-NN algorithm, especially in terms of implementation on medical data, and to identify factors that affect the success of this algorithm, and to make theoretical and practical contributions in the field of machine learning and medical technology, especially in autoimmune disease diagnosis, through synthesizing findings from various relevant literatures.

The main questions guiding this research are: To what extent is the K-NN algorithm effective in classifying autoimmune diseases compared to other algorithms as well as How can the implementation of K-NN be applied in a technology-based diagnostic system to support the clinical process in early diagnosis of autoimmune diseases?

This research focuses only on the Auto immune skin disorders dataset, which consists of scanned images of patients diagnosed with normal skin, Lupus skin disorders and Psoriasis skin disorders. While the results may provide valuable insights, they are limited to the context of this dataset only. Factors such as image resolution, quality and variation in skin presentation may affect the performance of the model. It should also be noted that this study does not address the performance comparison of the KNN algorithm with other machine learning algorithms.

This research contributes to the growing literature on machine learning applications in medical imaging. By showcasing the potential of the KNN algorithm in brain tumor image classification, this research provides a basis for further exploration and refinement.

THEORETICAL REVIEW

The implementation of machine learning in the medical field has been widely studied to address various challenges in disease diagnosis and classification. Among the algorithms employed, the K-Nearest Neighbors (K-NN) algorithm has garnered significant attention for its simplicity and effectiveness in classifying complex medical datasets. This section reviews

relevant studies on the application of K-NN for medical classification, particularly in autoimmune disease diagnosis.

1. K-NN Algorithm in Medical Applications

The K-NN algorithm has been demonstrated to be effective in various medical applications due to its non-parametric nature and simplicity. For example, (Annur, 2018) highlighted its potential in classification tasks where the algorithm can accurately predict disease categories based on historical data. Similarly, (Iffah'da & Anita Desiani, 2022) applied K-NN to primary biliary cirrhosis prediction, achieving high accuracy when optimal parameters were utilized.

However, challenges such as sensitivity to data imbalance and computational inefficiency in large datasets were noted in studies by (Oktaviana et al., 2024). These limitations underline the need for optimization techniques to enhance K-NN's performance in medical datasets, as also emphasized by (Sulistiyanto et al., 2023).

2. Autoimmune Disease Classification

Autoimmune diseases, characterized by the immune system attacking healthy tissues, often present overlapping symptoms, making diagnosis challenging. The use of computational approaches such as K-NN can assist in early detection. (Karim et al., 2023), explored Bayesian methods for autoimmune disease diagnosis, demonstrating the importance of leveraging machine learning to enhance diagnostic accuracy.

Additionally, (Setiawan et al., 2019) utilized Genetic Algorithms to optimize classification tasks, highlighting the potential of hybrid models. The combination of K-NN with optimization techniques like Genetic Algorithms or Particle Swarm Optimization has been suggested by multiple studies to overcome the limitations of standalone K-NN models.

3. Gap in Literature

While existing studies have shown the utility of K-NN in classifying medical data, including autoimmune diseases, there remains a need for:

- Testing K-NN performance on more diverse and balanced datasets.
- Exploring hybrid models that integrate K-NN with optimization algorithms to improve classification accuracy and computational efficiency.
- Evaluating the applicability of K-NN-based systems in real-world clinical settings, including their usability and integration with existing diagnostic tools.

METHODOLOGY

The dataset used in this study is taken from: https://www.kaggle.com/datasets/aditipathak17/autoimmune-skin-disorder-dataset in the form of a zip file that is downloaded and then extracted, the file will contain various skin image files in various categories. The dataset source was chosen because it has good image quality, detailed disease categories, and is relevant to this research.

1. Psoriasis skin (908 images)
2. Lupus skin (345 images)
3. Normal skin (1060 images)

The dataset used in this study amounted to 2313 skin images of autoimmune diseases with high resolution.

Dataset Adjustment

In the dataset, there is a problem of an unbalanced number of datasets between 3 skin types, to overcome that by adjusting the dataset to be balanced, an undersampling process is carried out to reduce the amount of data in categories with a larger number of images. Each category will be adjusted to 200 images.

Since this data imbalance has the potential to affect the performance of the classification model, undersampling will be performed. Undersampling is a method in data analysis used to handle class imbalance or unbalanced distribution between classes in a dataset. Basically, undersampling reduces the number of samples from the majority class so that the number of samples in the majority class becomes equal to the number of samples in the minority class (Muttaqin, 2023)

Adjustment Steps

a. Randomize data in categories with a larger amount of data (autoimmune disease with normal skin)
b. Randomly selecting 200 images from each category to ensure balance
c. Moving the selected images to a folder to make it more structured

Table 1. Dataset distribution after customization

Category	Total Images Before	Total Images After Undersampling
Normal Skin	1060	200
Lupus	345	200
Psoriasis	908	200

This pre-processing step is carried out to ensure all existing data is ready to be used for model training. The steps are:

1. Resizing: this is used to change the image to be more uniform for its dimensions (for example, 128x128 pixels) using the OpenCV or Pillow library.
2. Normalization: the pixel values of the image are normalized to the range [0, 1].
3. Data Augmentation: to increase the amount of data by techniques such as rotation, flipping, or brightness change to prevent overfitting.

Dataset Division

The dataset will be divided into three parts, namely:

1. Training data (80%): to train the model
2. Testing data (20%): for the final data of the model 3.

Distribution after the data is divided:

Table 2. Dataset Divison

Dataset	normal	lupus	psoriasis	Total
Training	160	160	160	480
Testing	40	40	40	120

The total split is made with a balanced composition so that the model can be trained and evaluated effectively.

RESEARCH RESULTS

Data preprocessing

Data preprocessing is an important process in a data mining analysis to clean, reformat, and prepare data for easier and more accurate analysis Data preprocessing is an important process in an analysis.

The overall goal of data preprocessing is to load image datasets from specific folders based on a category structure (e.g. skin diseases), resize all images to uniform dimensions (256x256), save the results to a new folder without losing the original color information of the images, and randomize the order of the data so that there is no bias when used in machine learning models.

Dataset Sharing

Dataset division is an important step in the Machine Learning process to ensure the model can learn effectively and be tested fairly. The dataset is divided into two main parts: the training set and the testing set. The dataset containing 600 images will be divided into:

- Train set: 80% of the total images (480)
- Test set: 20% of the total images (120)

The split will be done randomly using the train_test_split function from the sklearn library. Randomization with the random_state=42 parameter ensures that the split results are consistent in each trial.

Dataset Distribution for Each Category

Datasets taken from the same three categories after balancing are:

a. Normal Skin : 200 images
b. Lupus Skin : 200 images
c. Psoriasis Skin : 200 images

After dividing, the number of datasets in each category is :

A. Training Data:
   - 160 images of Normal Skin category
   - 160 images of Lupus category
   - 160 images of Psoriasis category

B. Test Data:
   - 40 images of Skin normal category
   - 40 images of Lupus category
   - 40 pictures of Psoriasis category

Preprocessing Results

After running the data preprocessing. Here are the details of the results:

1.Total Dataset

The dataset consists of 600 images :

- 200 images of normal skin category
- 200 images of lupus category
- 200 images of psoriasis category

2.Dataset division

The dataset is randomized and divided into two parts:

- Train Set: 80% of the total dataset (480 images).
- Test Set: 20% of the total dataset (120 images).

The splitting process is done using sklearn's train_test_split function. Example of data distribution:

A. Training data:
   - 160 normal skin category images
   - 160 images of Lupus category
   - 160 images of Psoriasis category

B. Test data:
   - 40 images of normal skin category
   - 40 images of lupus category
   - 40 images of psoriasis category

Classification Process

The last process is the classification process, which uses the K-Nearest Neighbors algorithm. The classification process uses 5 functions, namely:

1. Function to load images from the folder
2. Function for visualization of accuracy against k
3. Function for visualization of confusion matrix
4. Main function for classification with K-NN
5. Function to run the program

Table 3. Classification Process

Nilai K	Validation Accuracy (%)
1	0,62
2	0,57
3	0,54
4	0,54
5	0,59
6	0,55
7	0,60
8	0,53
9	0,56
10	0,57
11	0,56
12	0,55
13	0,53
14	0,53

This tuning is done by taking a range of K values from 1 to 15 which aims to find the best K value with the highest validation accuracy. In this table, the value of K = 1 gives the best validation accuracy of 0.62%.

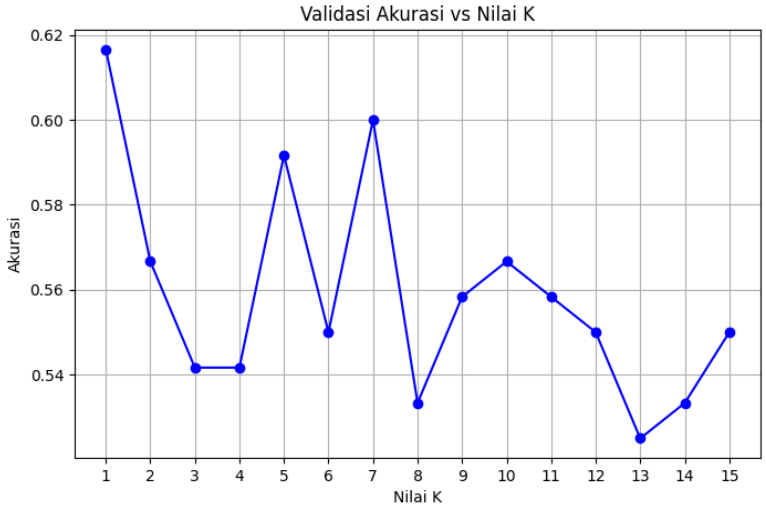

Figure 1. Validation Accuracy vs K Values

- The value of K = 1 gives the highest accuracy of about 0.62 (62%).
- As K increases to K=3 the accuracy decreases drastically to 0.54 (54%).
- At K=7, there is an increase in accuracy to 0.61 (61%), showing better performance than other K values in the range.
- After K=7, the accuracy tends to fluctuate but generally decreases, reaching the lowest value of about 0.53 (53%) at K=13.
- The accuracy at K=15 improved slightly compared to K=13 but was still lower than the maximum accuracy.

Testing Model

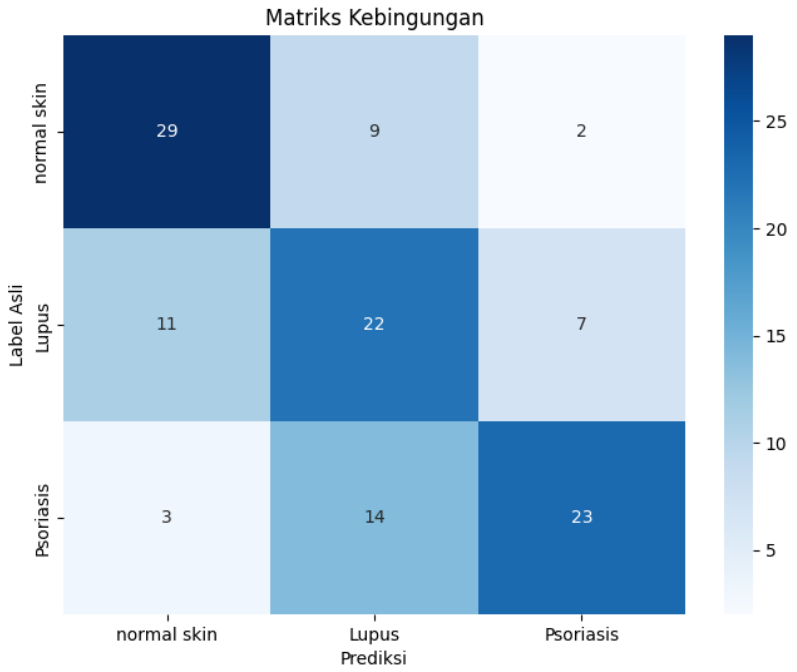

Figure 2. Testing Model

Confusion Matrix

The confusion matrix provides a detailed description of the model's performance in classifying the three classes, namely "normal skin", "lupus", and "psoriasis". The results show that the model successfully classified 29 samples of the "normal skin" class, 22 samples of the "Lupus" class, and 23 samples of the "Psoriasis" class correctly. However, there are some misclassifications that need to be considered.

From this analysis, it can be concluded that the model still needs to be improved, especially in reducing errors in the "lupus & psoriasis" class. One of the steps that can be taken is to add more training data to the class or re-evaluate the features used to improve the model's ability to distinguish patterns between classes. Thus, the overall accuracy of the model is expected to increase significantly.

Classification Result

Table 4. Classification Result

No	Class	Precision	Recall	F1-Score	Support
1	Normal skin	0.67	0.72	0.70	40
2	Lupus skin	0.49	0.55	0.52	40
3	Psoriasis skin	0.72	0.57	0.64	40
Accuracy				0.62	120
Macro avg		0.63	0.62	0.62	120
Weighted avg		0.63	0.62	0.62	120

1. Normal skin: in this class the model is good enough to identify skin with autoimmune diseases, with a recall of 72%, but the precision only touches 67%, which means there are many false positives.
2. Lupus: in this class the model is not good enough in recognizing this disease, with a recall value of only 55%, causing a lot of data with this disease to not be detected properly.
3. Psoriasis: in this class the model is quite good at identifying autoimmune skin with psoriasis, by having a precision rate of 72%, and recall of 57%.

Overall this model has a fairly good performance for all categories, it is characterized by the F1-Score value that touches 70% in normal skin, but this research still requires improvement, especially for identifying diseases such as lupus and psoriasis.

The Macro Avg calculation gives equal weight to each class as follows:

1. Precision Macro Avg $= \frac{0.67+0.49+0.72}{3} = 0.63$
2. Recall Macro Avg $= \frac{0.72+0.55+0.57}{3} = 0.62$
3. F1-Score Macro Avg $= \frac{0.70+0.52+0.64}{3} = 0.62$

Weight Avg calculation gives weight based on the amount of data in each class as follows:

1. Precision Weight Avg $= \frac{(0.67x40)+(0.49x40)+(0.72x40)}{120} = 0.63$
2. Recall Weight Avg $= \frac{(0.72x40)+(0.55x40)+(0.57x40)}{120} = 0.62$
3. F1-Score Weight Avg $= \frac{(0.70x40)+(0.52x40)+(0.64x40)}{120} = 0.62$

The calculation results of Macro Average and Weighted Average show the same values, namely Precision, Recall, and F1-Score of 0.63, 0.62, and 0.62, respectively. This happens because the amount of data in each class tends to be the same (40 data each). Overall, the model has moderate performance in classifying data. Precision, Recall, and F1-Score of around 62-63% indicate the model can be further improved to produce more accurate predictions.

DISCUSSION

The findings of this research demonstrate that the K-Nearest Neighbors (K-NN) algorithm holds considerable promise in the classification of autoimmune diseases, as evidenced by its performance in distinguishing between normal skin, lupus, and psoriasis categories. However, the moderate accuracy metrics (precision, recall, and F1-score hovering around 62-63%) reveal that there is significant room for improvement.

The results underline the sensitivity of the K-NN algorithm to dataset characteristics. The imbalance in the dataset—addressed through undersampling—may have contributed to the loss of potentially valuable data, thereby limiting the model's ability to generalize effectively. While undersampling was necessary to balance the dataset, alternative strategies such as Synthetic Minority Over-sampling Technique (SMOTE) could be explored in future studies to preserve minority class information.

The confusion matrix analysis revealed specific challenges in classifying lupus and psoriasis, with noticeable feature overlaps leading to misclassification. This suggests the need for more nuanced feature extraction. Advanced image processing techniques, such as texture analysis or pre-trained convolutional neural networks (CNNs), could significantly improve differentiation between these categories.

Parameter tuning also influences K-NN's performance. This study observed a peak accuracy of 62% at K=1, highlighting the importance of selecting an optimal K value. Additionally, integrating distance metrics other than Euclidean, such as Manhattan or Minkowski, could enhance adaptability, as suggested by Sulistiyanto et al. (2023).

Lastly, hybrid models combining K-NN with optimization algorithms, like Genetic Algorithms or Particle Swarm Optimization, could address its limitations. These approaches have shown potential in improving computational efficiency and classification accuracy in medical applications.

CONCLUSIONS AND RECOMMENDATIONS

This research aims to examine the application of the K-Nearest Neighbors (K-NN) algorithm in autoimmune disease classification through a literature review approach. Based on the analysis of previous research results, the K-NN algorithm is proven to have good accuracy potential for medical data classification, including autoimmune diseases. The success of K-NN is greatly influenced by the selection of optimal parameters, such as the number of nearest neighbors (K) and the distance metric used, such as Euclidean Distance.

However, this research also revealed some constraints on the K-NN algorithm, including sensitivity to unbalanced datasets, which can lead to majority class dominance in the classification results. In addition, this algorithm has high computational requirements when applied to large datasets, making it less efficient if not optimized.

FURTHER STUDY

Future research should focus on addressing the dataset limitations by collecting more comprehensive and balanced data from various sources. Applying advanced data augmentation techniques and exploring hybrid models combining K-NN with optimization algorithms, such as Genetic Algorithms or Particle Swarm Optimization, could improve performance. Additionally, testing the model on real-world clinical data and assessing its integration into diagnostic systems would be crucial for practical applications. Investigating the usability and effectiveness of K-NN in dynamic, multi-class environments, including rare autoimmune conditions, is also recommended.

ACKNOWLEDGMENT

This study would not have been possible without the unwavering support and contributions of many individuals and institutions. We extend our heartfelt gratitude to Universitas Pamulang for providing the necessary resources and academic environment to conduct this research. Special thanks are also due to our colleagues who offered invaluable feedback and guidance throughout the research process. Their constructive criticism and suggestions have greatly enriched the quality of this study.

We are deeply appreciative of the financial support provided by Universitas Pamulang, which made this research possible. Lastly, we would like to acknowledge the authors of the studies reviewed in this research. Their pioneering work has laid the groundwork for the present study and continues to inspire future advancements in the field of machine learning applications in medical diagnostics.

REFERENCES

Annur, H. (2018). Klasifikasi Masyarakat Miskin Menggunakan Metode Naive Bayes. *ILKOM Jurnal Ilmiah*, *10*(2), 160–165. https://doi.org/10.33096/ilkom.v10i2.303.160-165

Iffah'da, A. N., & Anita Desiani. (2022). Implementasi Algoritma K-Nearest Neighbor (K-NN) dan Single Layer Perceptron (SLP) Dalam Prediksi Penyakit Sirosis Biliari Primer. *Jurnal Ilmiah Informatika*, *7*(1), 65–74. https://doi.org/10.35316/jimi.v7i1.65-74

Karim, A., Esabella, S., Kusmanto, K., Suryadi, S., & Purba, E. (2023). Penerapan Metode Teorema Bayes Dalam Mendiagnosa Penyakit Autoimun. *Building of Informatics, Technology and Science (BITS)*, *5*(1), 254–263. https://doi.org/10.47065/bits.v5i1.3407

Muttaqin, A. (2023). *Mengatasi Data Imbalance Menggunakan Metode Undersampling NearMiss*.

Oktaviana, A., Wijaya, D. P., Pramuntadi, A., & Heksaputra, D. (2024). Prediksi Penyakit Diabetes Melitus Tipe 2 Menggunakan Algoritma K-Nearest Neighbor (K-NN). *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, *4*(3), 812–818. https://doi.org/10.57152/malcom.v4i3.1268

Setiawan, D., Putri, R. N., & Suryanita, R. (2019). Implementasi Algoritma Genetika Untuk Prediksi Penyakit Autoimun. *Rabit : Jurnal Teknologi Dan*

Sistem Informasi Univrab, *4*(1), 8–16. https://doi.org/10.36341/rabit.v4i1.595

Sulistiyanto, S., Saprudin, U., & Devani, F. T. (2023). Sistem Pakar Diagnosa Penyakit Autoimun dengan Metode Certainty Factor. *Jurnal Teknologi Informatika Dan Komputer*, *9*(2), 910–918. https://doi.org/10.37012/jtik.v9i2.1674