Expert Systems With Applications 199 (2022) 116857

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

A new nearest neighbor-based framework for diabetes detection[☆]

Suyanto Suyanto[a,*], Selly Meliana[a], Tenia Wahyuningrum[b], Siti Khomsah[b]

[a] *School of Computing, Telkom University, Bandung, Indonesia*
[b] *Faculty of Informatics, Institut Teknologi Telkom Purwokerto, Indonesia*

ARTICLE INFO

Keywords:
Diabetes detection
k-nearest neighbor
k-means clustering
Dimensional reduction
Multi-voter multi-commission

ABSTRACT

Diabetes is one of the deadliest and costliest diseases. Today, automatic diabetes detection systems are primarily developed using deep learning (DL) approaches, which give high accuracy in classifying patients into two classes: have diabetes or not. Unfortunately, DL is a high-complexity and unexplainable black-box model. This paper proposes a new nearest neighbor-based framework to tackle those issues in classifying two diabetes datasets: binary-class Pima India Diabetes Dataset (PIDD) and multiclass Diabetes Type. A *k*-means clustering (KMC) is first carried out to remove the noises or outliers and keep the competent data in the training set. The dimension of the competent data is then reduced using an autoencoder (AE) to minimize the distances of the intra-class data but maximize that of the inter-class. A *k*-nearest neighbor (KNN) classifier and two variants: pseudo nearest neighbor rule (PNNR) and local mean-based pseudo nearest neighbor (LMPNN), are used to detect diabetes. In addition, a new variant named multi-voter multi-commission nearest neighbor (MVMCNN) is introduced. An investigation based on 5-fold cross-validation (FCV) informs that, for binary-class PIDD, the proposed combination of KMC, AE, and MVMCNN achieves the highest accuracy of 99.13%, which is slightly higher than the state-of-the-art DL-based detection model that produces 98.07%. An evaluation based on 10-FCV also indicates that, for the multiclass Diabetes Type, it obtains a higher accuracy of 95.24% than the DL-based model for predicting diabetes that gives 94.02%.

1. Introduction

Diabetes, a disease that happens when the blood glucose level is high, can be grouped into three types, namely type 1, type 2, and gestational (Federation, 2021). In the first type, the pancreas produces few (even no) insulin. The second type occurs if the pancreas produces insufficient insulin or the body is unable to utilize the insulin adequately. In gestational diabetes, high blood sugar happens during pregnancy, associated with both mother and child complications. Based on the International Diabetes Federation (Federation, 2021), 20 out of 125 million live births have hyperglycemia in pregnancy, and 84% of them are caused by gestational diabetes.

Diabetes may cause several complications, such as cardiovascular diseases (Monnier et al., 2021), blindness, stroke, amputation or kidney failure (Yang et al., 2020), early microvascular disease, and premature ovarian aging (Yi et al., 2021). These facts make diabetes a costly disease, where most costs come from the treatment and management of the complication. As described in Yi et al. (2021), the cost estimates are $24,013, $45,549, $25,431, $8907, $31,202, and $25,008 for stroke, myocardial infarction (MI), lower extremity amputation (LEA), angina, congestive heart failure (CHF), and amputation, respectively. Hence, automatic early detection of diabetes is crucial to help doctors give proper actions to handle the disease.

Recent diabetes detection systems are commonly developed using the DL-based approach. Some researchers report that DL can give a high accuracy of more than 98%, which outperforms many ML-based models, such as logistic regression (LR), decision tree (DT), support vector machine (SVM) (Islam Ayon & Milon Islam, 2019; Naz & Ahuja, 2020; Zhou et al., 2020), and artificial neural network (ANN) (Chaves & Marques, 2021). Nevertheless, DL is a high-complexity, black-box model that can be unfavorable in medical. In the training process using hundreds, thousands (or even millions) of data samples, it has to optimize millions of parameters, needing high computational resources and time (Bai et al., 2021). Besides, the black-box nature of the DL is still unresolved (Tjoa & Guan, 2020), making its decision is poorly understood by the domain experts in precision medicine. The main problem is that the softmax probabilities in DL cannot be used as an estimator of confidence (Gal, 2016). Therefore, today some researchers focus on developing new concepts and metrics to make the DL more

[☆] This research is funded by the Directorate of Research and Community Service, Telkom University, with grant number: 571/PNLT3/PPM/2020.

[*] Corresponding author.

E-mail addresses: suyanto@telkomuniversity.ac.id (S. Suyanto), sellym@telkomuniversity.ac.id (S. Meliana), tenia@ittelkom-pwt.ac.id (T. Wahyuningrum), siti@ittelkom-pwt.ac.id (S. Khomsah).

https://doi.org/10.1016/j.eswa.2022.116857
Received 30 April 2021; Received in revised form 27 November 2021; Accepted 7 March 2022
Available online 23 March 2022

understandable (Bai et al., 2021; Barredo et al., 2020). One of them is deep k-nearest neighbors (Papernot & Mcdaniel, 2018), which leverages a KNN to provide confidence and credibility in DL decisions. As we know, KNN has an explainability by conforming the distance of a test data (query) to the training data.

In this research, a new framework of diabetes detection is proposed by leveraging three schemes: clustering, dimensional reductions, and nearest neighbor-based classifiers. In the first stage, a KMC is used to remove the noisy data in both diabetes classes of the training set so that all the data are more competent to make a decision accurately. Next, an AE is applied in the second stage to reduce their dimension to provide better distributions. Finally, a KNN-based classifier and two variants: PNNR and LMPNN, are exploited to detect diabetes. In addition, a new variant named MVMCNN is introduced. A 5-FCV scheme is used to evaluate the proposed framework. A further investigation is performed by comparing the proposed model with the DL-based diabetes detection described in Naz and Ahuja (2020) and Zhou et al. (2020).

2. Related works

A few papers published in 2018 to 2020 discuss the rule-based approach in predicting diabetes, especially computer science. A paper by Mehra et al. (2020) is one of them. It follows the approach by generating IF-THEN rules from linguistic summarization for diabetes prediction. The paper checks four measurements; degree of truth, coverage, reliability, and outliers. The reliability measurement was found to be most beneficial for prediction. It stated that the approach has promising results. The research clearly stated accuracy in a number using a rule-based approach in predicting the disease found in Hayashi and Yukita (2016). They run a classification using an algorithm called Re-RX (Recursive-Rule eXtraction). The algorithm is claimed to give high accuracy. It utilizes J48graft for the sampling process to reduce its complexity. The sampling Re-RX with J48graft obtains fewer rules and produces a mean accuracy of 83.83%.

Meanwhile, the machine learning (ML) approaches are dominating the discussion of diabetes prediction. DT, NB, SVM, LR, and random forest (RF) are the five most well-known predictive ML techniques used for comparing accuracy in diabetes prediction. The accuracy varies from 76% to 78%. Based on a comparative study in Battineni et al. (2019), Jakka and Vakula Rani (2019), Kavitha and Subbaiah (2019) and Sisodia and Sisodia (2018), DT, NB, and LR take turns as the best predictor. In Kavitha and Subbaiah (2019), the authors predict diabetes at an early stage by implementing LR, NB, and DT with the accuracy of 75.3%, 76.6%, and 77.9%, respectively. In Sisodia and Sisodia (2018), DT, SVM, and NB are used in their investigation on diabetes detection, also in the early stage. Experiment on a 10-FCV shows that NB (with a prediction rate of 76.3%) outperforms the rest. In Battineni et al. (2019), LR is stated as the best predictor of diabetes by gaining accuracy of 77% and 77.6%, respectively. It outperforms NB, J48, and RF using 5, 10, 15, and 20-FCV. In Jakka and Vakula Rani (2019), using the data preprocessing to remove the duplicate and missing values from the dataset, the LR also gives higher accuracy than KNN, DT, NB, SVM, and RF.

In Zhu et al. (2019), LR is improved by enhancing the clustering process, which holds an essential role in LR. KMC is a commonly used method for clustering due to its simplicity. Nevertheless, the initial positions of the cluster centers (centroids) are sensitive and challenging to define. The authors state that utilizing principal component analysis (PCA) for KMC increases the LR accuracy with 1.98% higher accuracy than other methods.

RF is also commonly used in predicting diabetes. Based on the previously stated papers, its accuracy is not as high as DT or LR. Nevertheless, with some modifications, the algorithm can perform better than the conventional one with an accuracy above 78%. RF suffers from processing time since the algorithm builds many DTs and uses many data. Parallel computing is applied on RF to reduce its computational time (Azizah et al., 2019). In 2020, Raghavendra and Santosh applied RF with the feature selection method by leveraging entropy evaluation-based (Raghavendra & Santosh Kumar, 2020) forward selection and backward elimination. They claimed RF achieves an accuracy of 84.1% by applying those methods. Outliers and missing values are also two problems leading to lower accuracy. In Maniruzzaman et al. (2018), a simple model is proposed to change those outliers and missing values with the computed median to improve accuracy. It states that the replacement makes RF produce a higher accuracy, up to 92.26%.

Many researchers also prefer SVM as one of the well-known methods to diagnose diabetes. Many papers combine SVM with other methods to achieve better prediction accuracy. In Lukmanto et al. (2019), SVM is utilized for training a dataset generating the fuzzy rules. It uses feature selection to retrieve essential features in a dataset; afterward, SVM trains the dataset to generate rules; finally, a fuzzy-based model is used to classify. The authors state that applying those methods retrieves an accuracy of 89.02%. Another combination using SVM is also found in Howsalya Devi et al. (2020) that utilizes a farthest first (FF) algorithm to cluster the dataset to be several subsets. The output of FF subsequently becomes the input of the SVM classifier. A sequential minimal optimization (SMO) is used to help the SVM in the training process solve quadratic programming problems, which commonly appear during the process. The proposed integration obtains a high accuracy of 99.4% in classifying which patient suffered diabetes and which one is not.

Aside from those five most frequently discussed methods above, some researchers also include KNN compared to their proposed methods. KNN is categorized as a lazy prediction method to group datasets by their similarity. In Alehegn et al. (2019), the researchers propose an ensemble approach to analyze and predict people with diabetes. The proposed approach combines KNN, RF, NB, and J48 to reach 93.62% of accuracy. Most papers discussing KNN in diabetes prediction compare the method with other well-known ML methods. Some recent articles (Jakka & Vakula Rani, 2019; Kaur & Kumari, 2019; Rajni & Amandeep, 2019; Tripathi & Kumar, 2020) compare KNN with other well-known methods: DT, NB, LR, SVM, and RF. However, based on those papers, KNN never comes first in the accuracy comparison.

DL is the latest advanced model compared to the rule-based and conventional ML ones. Many researchers are also proposing this method to diagnose diabetes mellitus. DL method is used in Naz and Ahuja (2020) to detect diabetes in the early phase, to help doctors give appropriate action or advice to stop the disease from further progress. The paper compares DL with ANN, NB, and DT. It states that DL outperforms all three models by a much higher accuracy, up to 98.07%, for PIDD.

Other research using DL to predict diabetes is found in Islam Ayon and Milon Islam (2019). The research uses a deep neural network (DNN) based on 5-FCV and 10-FCV. The 5-FCV obtains a better result than the 10-FCV by 1.24%. DL with attributes trained in 5-FCV gets an accuracy of 98.35%, while the 10-FCV reaches 97.11%.

A recent paper by Huaping Zhou in Zhou et al. (2020) proposes a DL-based model for predicting diabetes (DLPD) for not only diabetes prediction but also diabetes type prediction. DLPD combines the use of hidden layers of DNN and dropout regularization. The latter is utilized to intercept over-fitting. To achieve high accuracy, DLPD tunes several parameters and exploits a loss function of binary cross-entropy. It reaches high accuracy of 99.41% for detecting diabetes in a patient (yes or no) and 94.02% for classifying the diabetes types (1 or 2) using a testing set of 15% out of the PIDD. This result of DLPD is better than the two previous models in Islam Ayon and Milon Islam (2019) and Naz and Ahuja (2020).

Unfortunately, DL needs a high-computational resource and a long learning process. It should optimize millions of parameters for training thousands of data samples (Bai et al., 2021) and even more if a data augmentation is necessarily needed. Besides, DL is a black-box model (that is not explainable) since the softmax probabilities in DL is not a reliable estimator of confidence (Gal, 2016). Hence, today some experts focus on developing new concepts and metrics to make the DL more understandable (Bai et al., 2021) by incorporating white-box conventional machine learning models, such as DT, fuzzy rule-based learning, and KNN (Barredo et al., 2020). For instance, a deep k-nearest neighbors (Papernot & Mcdaniel, 2018) is a combined DL and KNN. KNN can conform the distance (dissimilarity) of the given test data (query) to the training data. Thus, incorporating KNN into DL provides both confidence and credibility metrics in the DL decision, making DL more explainable.

Therefore, in this research, a new lower-computational framework using nearest neighbor-based classifiers is proposed to handle those issues: expensive computation, time-consuming, and unexplainable. Besides, a new variant of the nearest neighbor classifier named MVMCNN is proposed by exploiting an advanced distance-formula to enhance the classification decision. The framework is then examined and compared with a DL-based model described in Naz and Ahuja (2020) based on a 5-FCV scheme using the same publicly accessible dataset of binary-class PIDD from the UCI Repository, which can be downloaded in Kaggle (2021). It is also evaluated and compared with the DLPD described (Zhou et al., 2020) based on a 10-FCV scheme using the same publicly accessible dataset of multiclass Diabetes Type from the Data World Repository (World, 2021).

3. Proposed framework

Fig. 1 illustrates a new low computational framework proposed in this paper. A data imputation firstly processes the dataset of N data objects to tackle the missing values. The dataset is then divided into training and testing sets based on 5 or 10-FCV. Next, the N data objects are clustered using a KMC with $k = 2$ to N to produce several clusters with the maximum densities based on the silhouette coefficients. Merge the small clusters, which have less than a particular number of members, to the closest bigger cluster if possible. Otherwise, remove them. Next, the data dimension is reduced using an AE. Next, a min–max normalization (MMN) is applied to normalize the lower-dimensional data. Finally, the normalized data is classified using nearest neighbor-based classifiers: KNN, PNNR, LMPNN, and MVMCNN.

Moreover, merging a small cluster into the closest bigger cluster (or removing it) is implemented using a simple procedure, as illustrated in Fig. 2. First, search the small clusters containing α data samples or less. Second, for each sample in a small cluster, check if the β nearest neighbors of the sample are all in the same class or not, as illustrated in Fig. 2(a). If so, keep the sample as a competent voter. Otherwise, remove it from the small cluster since it is an incompetent voter (that is contaminated by samples in another class), as illustrated in Fig. 2(b). Third, check if the number of the remaining samples in the small cluster is bigger than (or equal to) γ or not. If so, merge them to the closest big cluster. Otherwise, remove the small cluster since it has too few competent voters to make a decision, as illustrated in Fig. 2(c). Based on preliminary experiments on PIDD and Diabetes Type datasets, the three constants α, β, and γ are set to 15, 5, and 10, respectively.

The basic concept of the proposed framework is illustrated in Fig. 3. Let the blue square be the given testing data (or query point) that should be classified using the nearest neighbor classifier $k = 3$. In the original dataset, the query is wrongly classified by KNN as the red triangle class, as depicted in Fig. 3(a). In the noise-removed dataset, it can be classified by KNN as the green circle (see Fig. 3(c)). In the dimensional-reduced dataset, it is more easily classified since both classes are far apart (see Fig. 3(e)). Meanwhile, MVMCNN can give correct classifications for all three cases. In Fig. 3(b), the query is correctly classified as the green circle class in the original dataset since three too-small commissions (clusters) containing only one data sample (outlier) are not taken into account in the classification decision. As illustrated in Figs. 3(d) and 3(f), it can also be classified as the true green circle class in the noise-removed and dimensional-reduced datasets.

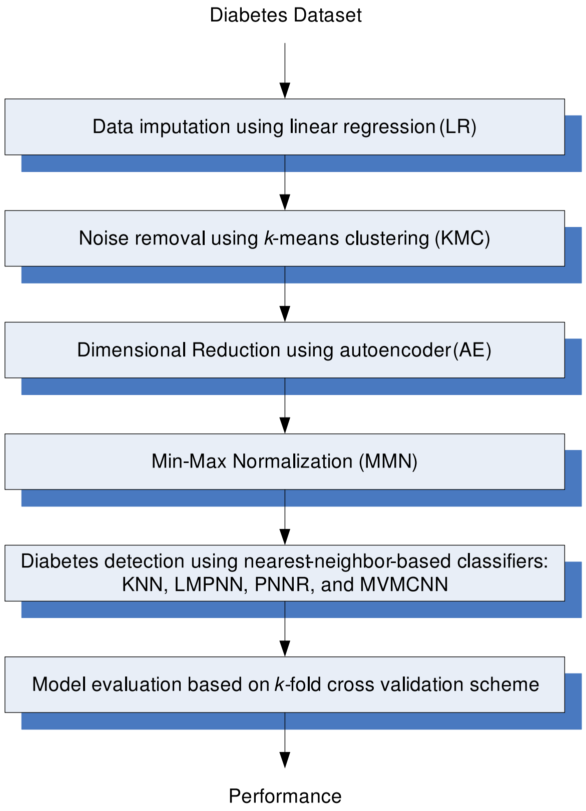

Diabetes Dataset
Data imputation using linear regression (LR)
Noise removal using k-means clustering (KMC)
Dimensional Reduction using autoencoder (AE)
Min-Max Normalization (MMN)
Diabetes detection using nearest-neighbor-based classifiers: KNN, LMPNN, PNNR, and MVMCNN
Model evaluation based on k-fold cross validation scheme
Performance

Fig. 1. Proposed framework.

Table 1
Pima Indian Diabetes Dataset (PIDD).

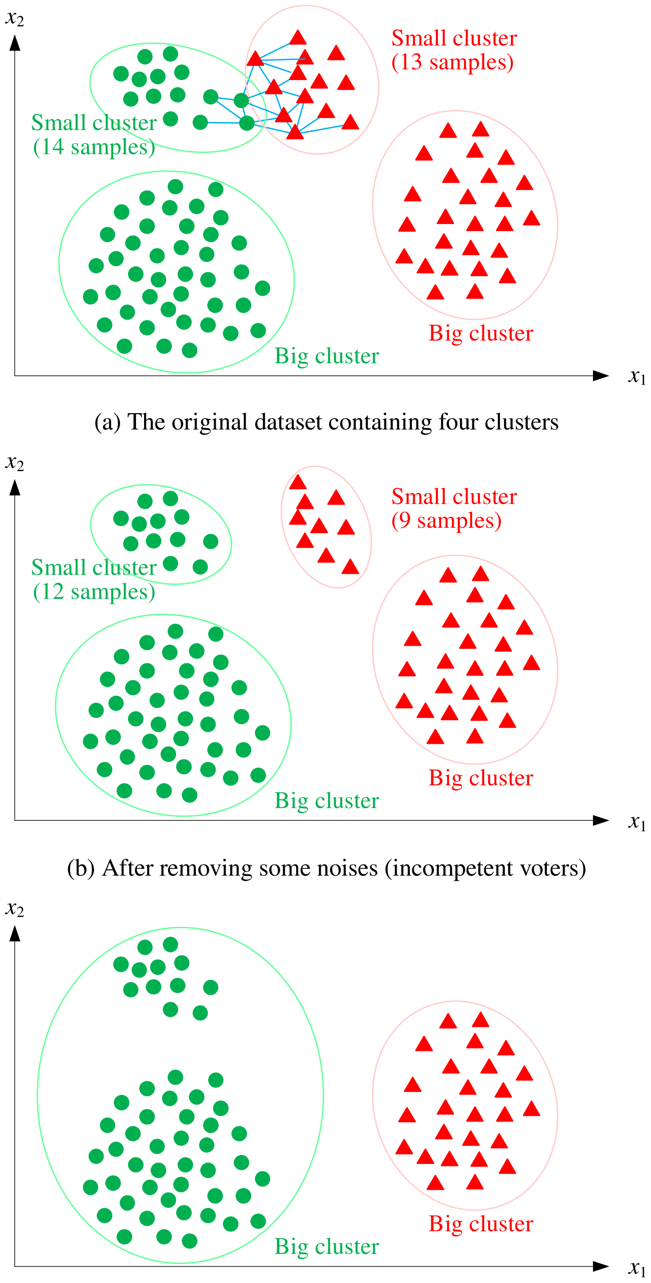
Column	Attribute	Interval
1	Pregnancies	[0, 17]
2	Glucose	[0, 199]
3	Blood pressure	[0, 122]
4	Skin thickness	[0, 99]
5	Insulin	[0, 846]
6	BMI	[0, 67.1]
7	Diabetes pedigree function	[0.0078, 59.4]
8	Age	[21, 81]
9	Outcome (Class/Label)	1/0 (Yes/No)

3.1. Diabetes datasets

The datasets used in this research are the PIDD from the UCI data repository and Diabetes Type from the Data World repository (World, 2021). PIDD is an imbalanced binary classification problem that consists of 768 instances: 500 are negative (outcome = 0) and 268 are positive (outcome = 1). Each instance has nine attributes: eight attributes to consider and one class label, as listed in Table 1. Meanwhile, the Diabetes Type dataset contains 1009 instances. Based on the seventh attribute (Type), this dataset is a multiclass classification problem with 631 Normal (62.54%), 205 Type2 (20.32%), and 173 Type1 (17.15%). Meanwhile, based on the eighth attribute (Class), it is a binary classification problem with 631 False (62.54%) and 378 True (37.46%), as illustrated in Table 2.

3.2. Data imputation

In PIDD, there are some missing values in up to ten percent of the total data objects. Hence, a linear regression-based data imputation is performed to handle the missing values. It can be easily explained as follow. First, the correlation values between a feature



(a) The original dataset containing four clusters

(b) After removing some noises (incompetent voters)

(c) After removing a small cluster

Fig. 2. Procedure to merge or remove the small clusters.

Table 2
Multiclass Diabetes Type dataset.

Column	Attribute	Interval
1	Age	[21, 81]
2	BS fast	[0, 54]
3	BS pp	[4.2, 8.8]
4	Plasma R	[7.9, 13.1]
5	Plasma F	[3.9, 9.1]
6	HbA1c	[28, 69]
7	Type	Normal/Type1/Type2
8	Class	1/0 (True/False)

(where the missing value occurs) and all the others are calculated. A linear regression model is generated using the feature with the highest correlation. Finally, the missing value is solved by predicting based on the developed linear regression model.

3.3. Noise removal

Like other real-world datasets, both PIDD and Diabetes Type datasets contain noises or outliers. In this research, a KMC is used to cluster the training set, and then a tiny cluster with less than ten data objects is removed since it is considered noise.

3.4. Dimensionality reduction

The dimension reduction is carried out using an AE. It is an unsupervised ANN, commonly used to decrease the number of features (Kannadasan et al., 2019). One of its advantages is the ability to develop a nonlinear model. It has an encoder and a decoder. The encoder maps an original data input into a lower-dimensional latent space, and the decoder can reconstruct it back into the original input.

Besides, AE can also be used as a data augmentation (over-sampling) and a feature augmentation (feature expansion). In García-ordás et al. (2021), a variational AE (VAE) is used as data augmentation and a sparse AE (SAE) as feature augmentation. The SAE and CNN classifiers trained jointly significantly improve the accuracy of CNN for the diabetes classification of PIDD. In this paper, AE is used as a dimensional reduction.

3.5. Normalization

A normalization is implemented using an MMN, which performs a linear transformation from the original data. Assume that A_{min} and A_{max} are the smallest and biggest values in the feature A, respectively. As in Pandey and Jain (2017), the MMN maps a value, x_i, of A to x_i' in the interval $[A_{min,new}, A_{max,new}]$ as follows

$$x_i' = \frac{x_i - A_{min}}{A_{max} - A_{min}}(A_{max,new} - A_{min,new}). \tag{1}$$

3.6. Nearest neighbor-based classifiers

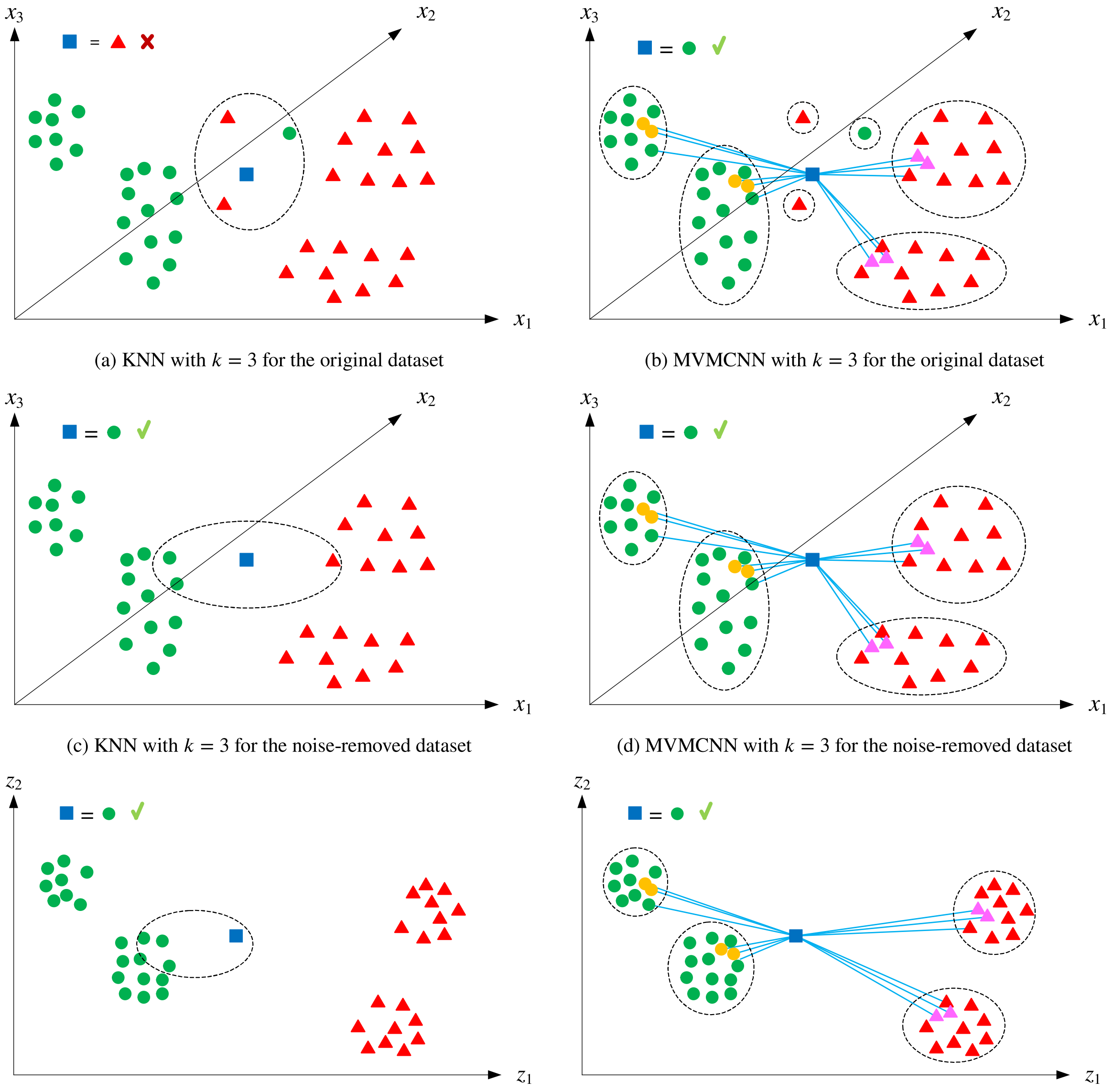
A nearest neighbor-based classifier is a simple model with lazy learning. However, it has three main advantages: explainability, a fast learning process, and local decision. Hence, many researchers have proposed various variants, such as PNNR and LMPNN.

3.6.1. KNN

As the name suggests, KNN is a model that works locally based on k closest instances (nearest neighbors) out of all the data objects in the original dataset used to train (Zhang et al., 2018). Each data object is a vector in a multi-dimensional feature space with a label. KNN predicts a label of the testing data (query) using a majority voting scheme from the k closest data objects commonly selected by a distance of Euclidean

$$d(P,Q) = \sum_{i=1}^{n}(P_i - Q_i)^2, \tag{2}$$

where P$(p_1, p_2, \dots, p_n)$ and Q$(q_1, q_2, \dots, q_n)$ are data objects and n is the number of attributes or features (dimension) (Harrison, 2018). Although KNN has some advantages, it also has one main disadvantage: the majority voting used in KNN tends to obtain a misclassification for a noisy dataset.



(a) KNN with $k = 3$ for the original dataset

(b) MVMCNN with $k = 3$ for the original dataset

(c) KNN with $k = 3$ for the noise-removed dataset

(d) MVMCNN with $k = 3$ for the noise-removed dataset

(e) KNN with $k = 3$ for the dimensional-reduced dataset

(f) MVMCNN with $k = 3$ for the dimensional-reduced dataset

Fig. 3. Basic concept of the proposed framework.

3.6.2. PNNR

PNNR is commonly better than KNN (Zeng et al., 2009). It works simply using three steps. First, it finds k closest instances (neighbors) in every class. Second, for each class, it calculates the total distance of k instances using a gradual weighting depending on their rankings. Hence, it can be said that PNNR uses a multi-voter scheme, which makes this model outperforms KNN that uses the single-voter one. Third, it decides the winning class by minimizing those total distances.

Moreover, the gradual weighting is straightforwardly defined by formulating the weight W_j^i of the jth neighbor x_j^i from the class ith as

$$W_j^i = \frac{1}{j}, \tag{3}$$

where $j = 1, 2, \ldots, k$ is the ascending ordered-ranking.

3.6.3. LMPNN

LMPNN is another variant KNN proposed by Gou et al. (2014). It is an improved PNNR, where it uses k local mean vectors of the k closest instances. It also uses a similar neighborhood weight as in the PNNR. However, it searches the pseudo nearest neighbors using different ways: k local mean vectors.

A local mean vector $\bar{x}_j^i$ of the first j first neighbors of the testing data or query x of the ith class is calculated by

$$\bar{x}_j^i = \frac{1}{j}\sum_{l=1}^{j} x_l^i, \tag{4}$$

where $1 \leq j \leq k$.

3.6.4. MVMCNN

MVMCNN is proposed here based on the LMPNN previously described in Section 3.6.3. First, it splits each class into some subclasses or clusters (commissions). After that, it predicts the label of the testing data (query) by choosing a commission with the lowest total distance of k instances, which is calculated by Eq. (4). The MVMCNN pseudocode, which is adopted from Suyanto et al. (2022), is shown in detail in Algorithm 1.

Furthermore, Fig. 4 illustrates the differences between MVMCNN with PNN and LMPNN in a binary classification problem. Suppose a testing data (shown by a blue square), should be classified as a green circle (Class 1) or a red triangle (Class 2). In this case, assume that the testing data belongs to Class 1. Next, let the means of two and three

Algorithm 1: MVMCNN

```
Result: $C_{best}$ as the output class
Split a given dataset into $N_c$ clusters based on their classes;
Name the $N_c$ clusters as $C_{i,m}$ representing the $i$th class and the
  $m$th commission;
for every commission, compute the distance of LMPNN based on
  Eq. (4);
Find the lowest distance $C_{i,m}$;
Return a class with minimum distance as the output class $C_{best}$;
```

first neighbors in Class 1, weighted by $\frac{1}{2}$ and $\frac{1}{3}$, are given by the orange circles. Meanwhile, those in Class 2 are provided by the pink triangles.

When $k = 3$, PNNR wrongly predicts the label of the testing data because the minimum total distance is reached by Class 2. Similarly, LMPNN also gives a wrong classification even though it gives a more accurate total distance: the total distance of Class 2 is only slightly smaller than Class 1. Meanwhile, MVMCNN correctly classifies it since Commission 1 in Class 1 ($C_{1,1}$) achieves the minimum total distance among all the generated commissions.

4. Results and discussion

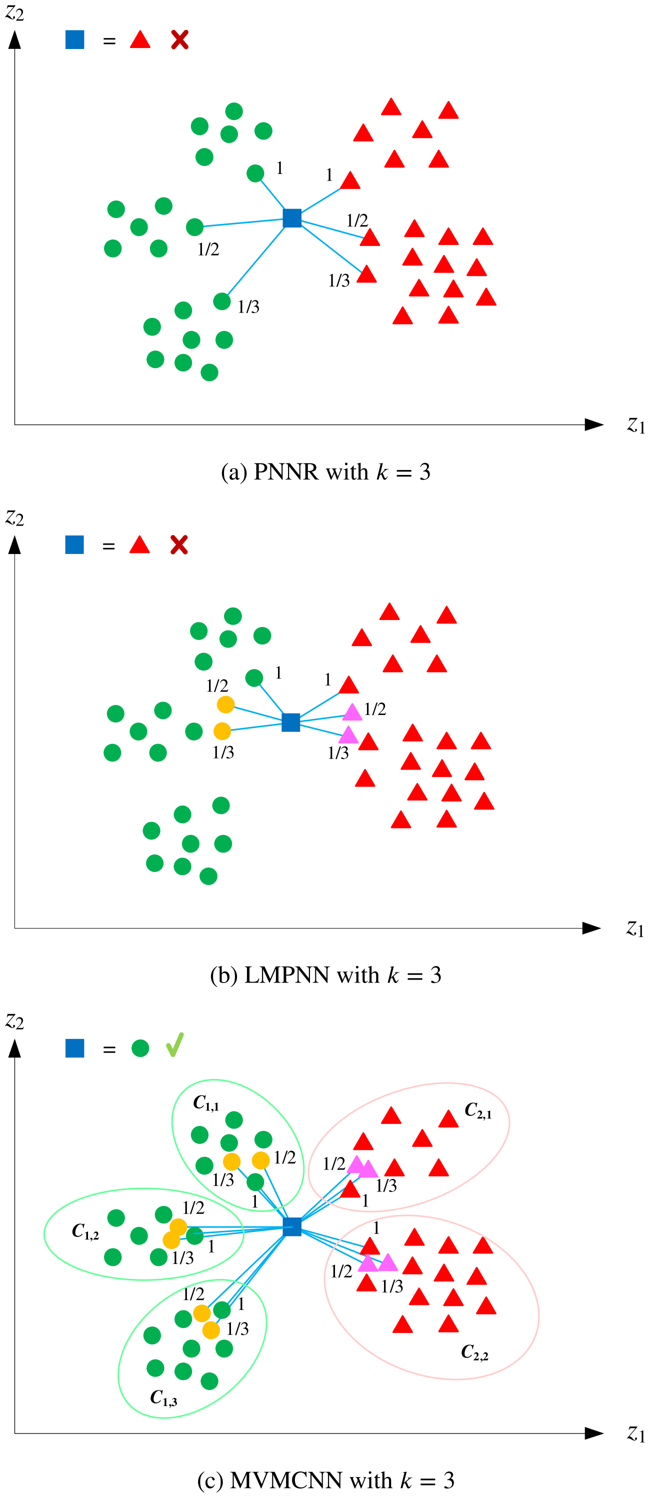
The evaluation is performed in two stages. Firstly, MVMCNN is competed with LMPNN using 10-FCV for ten real-world public accessible datasets from UCI Repository for various $k = 1$ to 15 to investigate the benefit of the multi-commission model in the MVMCNN. It is not compared to KNN and PNNR since both models are worse than LMPNN, as examined in Gou et al. (2014). Secondly, the proposed framework is examined using two diabetes datasets: PIDD (Kaggle, 2021) and Diabetes Type (World, 2021).

4.1. Evaluation of the proposed MVMCNN

In MVMCNN, the k-means clustering is performed to any class to create a few clusters (commissions). Based on preliminary observation, the parameter c (the number of clusters) is limited to 2 to 6. This scheme aims to create clusters with maximum samples and silhouette coefficients. As the datasets vary from hundreds to thousands of data samples, the neighborhood sizes k for MVMCNN and LMPNN are limited to 1 to 15. Thus, when a tiny cluster containing 14 samples or less is generated, it is merged into the closest big cluster. But, if it is not possible to be merged due to the number of generated clusters being only 2 (minimum), it can be accepted as it is. If it cannot be merged because of the significant decrement of silhouette coefficient, it is removed. Finally, MVMCNN predicts the label of the query by selecting the commission (with information of its class) with the lowest total distance, which is calculated using Eq. (4).

Table 3 illustrates the highest mean accuracies equipped with the corresponding standard deviations as well as the optimum neighborhood sizes k and the Wilcoxon's rank-sum tests (WRST) in the parentheses produced by LMPNN and MVMCNN for ten varying UCI datasets. The symbols −, +, and ≈ in the parentheses show that the current results (of 10-FCV) are significantly worse, significantly better, and not significant than the results MVMCNN in terms of WRST using a significance level of 0.05, respectively. The highest mean accuracies are provided by bold texts.

MVMCNN obtains significantly higher accuracies than LMPNN for 7 out of 10 datasets: three binary-class and four multiclass. It provides the same or slightly higher accuracies than LMPNN for two datasets: one binary-class (Wdbc) and one multiclass (OptDigits). It gives a lower accuracy only for one multiclass dataset of Cardiotocography. Statistically, the Friedman mean rank (FMR) test informs that MVMCNN is the first rank with a score of 1.10, lower than LMPNN (1.80). Therefore, the multi-commission scheme in MVMCNN benefits in decreasing the biased decision made by the single-commission in the LMPNN. The optimum k that gives the highest accuracies in MVMCNN are generally smaller (or equal) than the LMPNN because the classification is conducted in each commission (not class). This fact indicates that the classification in MVMCNN is performed by the most competent voters in each commission without any contamination from incompetent voters of other commissions. However, MVMCNN may give a worse performance because of the enforcement of clustering into two



(a) PNNR with $k = 3$

(b) LMPNN with $k = 3$

(c) MVMCNN with $k = 3$

Fig. 4. Differences between MVMCNN with PNN and LMPNN.

Table 3
The highest mean accuracies (%) and the standard deviations as well as the optimum neighborhood sizes k and the Wilcoxon's rank-sum tests (WRST) in the parentheses produced by LMPNN and MVMCNN for ten UCI datasets. The symbols −, +, and ≈ in the parentheses show that the current result is significantly worse, significantly better, and not significant than the result of MVMCNN in terms of WRST using a significance level of 0.05, respectively.

Dataset	#Samples	#Attributes	#Classes	LMPNN	MVMCNN
Transfusion	748	4	2	67.80 ± 9.12(15)(−)	**72.47 ± 4.29 (15)**
Glass	146	9	2	81.86 ± 12.08(2)(−)	**92.06 ± 12.17(1)**
Parkinsons	195	22	2	77.76 ± 12.35(14)(−)	**81.92 ± 11.19(10)**
Wdbc	569	30	2	**93.61 ± 2.94(7)**(≈)	**93.61 ± 2.94(7)**
Wine	178	13	3	75.65 ± 7.69(13)(−)	**76.01 ± 14.49(3)**
Thyroid	7200	21	3	98.35 ± 1.08(10)(−)	**98.40 ± 1.10(10)**
Letter	7648	16	10	97.06 ± 0.44(15)(−)	**97.92 ± 3.18(15)**
Cardiotocography	2126	21	10	**55.04 ± 5.17(13)(+)**	54.89 ± 4.18(15)
OptDigits	5620	64	10	99.04 ± 0.43(9)(≈)	**99.06 ± 0.40(8)**
Vowel	528	10	11	63.94 ± 8.02(15)(−)	**65.65 ± 5.41(8)**
Friedman mean rank (FMR)				1.80	1.10
Rank				2	1

to six clusters. For instance, it is happened in the Cardiotocography dataset due to the difficulty in generating at least two clusters in each class. Enforcing split each class in this dataset into at least two clusters make MVMCNN biased in the decision-making. Nevertheless, keeping one cluster for every class makes it produce the same classification as LMPNN.

4.2. Accuracy of the proposed framework

The experiments are performed using the PIDD based on a 5-FCV scheme. First, PIDD is randomly split into the training sets of 4-folds and the testing sets of 1-fold to give fair comparisons. Six different models are then trained using KNN, KMC+KNN, and combination models of KMC, AE, with KNN, PNNR, LMPNN, and MVMCNN.

In the first model, both imputation and normalization are performed to the training sets, but KMC and AE are not. Hence, this model is named KNN. It is a baseline model of diabetes detection in the proposed framework. In the second model, a KMC is incorporated into the first model. Therefore, it is called KMC+KNN. In the third model, an AE is combined with the second model to be KMC+AE+KNN. Finally, the fourth to the sixth models are KMC+AE+PNNR, KMC+AE+LMPNN, and KMC+AE+MVMCNN that use three KNN variants: PNNR, LMPNN, and the new proposed MVMCNN, respectively.

All the models are then evaluated to see the impact of KMC-based noise removal and AE-based dimensional reduction in detecting diabetes. After that, they are finally compared based on the averaged accuracies produced from five experiments based on the 5-FCV scheme using both PIDD and Diabetes Type datasets.

For the PIDD dataset, the results in Fig. 5 inform that KNN gets the lowest averaged accuracy of 81.80% since it uses a majority voting on a noisy dataset. The KMC+KNN (second model) improves the accuracy drastically to 86.70%. This result indicates that the KMC-based noise removal can remove the incompetent voters and keep the competent ones.

Next, the KMC+AE+KNN model produces an averaged accuracy of 92.20%. It shows that AE, which reduces the data dimension from 8 to 3, can enhance the data distributions in both classes of PIDD. The 3-dimensional data provides the optimum features that minimize the intra-class distance and maximize the inter-class ones.

The KMC+AE+PNNR model achieves a higher accuracy of 97.80%, which indicates that considering multi voters in each class improves the decision process. Next, the KMC+AE+LMPNN model slightly improves the classification rate to 98.98%. It shows that the decision-making in LMPNN is conceptually better than PNNR.

The KMC+AE+MVMCNN model eventually gets the highest mean accuracy of 99.13%. It proves that the new multi-commission scheme proposed in this paper is able to boost the decision-making in LMPNN. In addition, this result is higher than the DL-based model that obtains 98.07%. All those results inform that the new framework proposed in this paper is able to enhance KNN, where each procedure plays an essential role as it is designed.

For the multiclass Diabetes Type classification, the proposed framework produces similar results. Fig. 6 illustrates that KNN gets the lowest averaged accuracy of 87.61%. This accuracy is higher than PIDD since the Diabetes Type dataset contains fewer attributes and lower noises. Furthermore, the KMC+KNN model increases the averaged accuracy to 90.98%. This result indicates that the KMC can also remove the incompetent voters in the Diabetes Type dataset.

Meanwhile, the KMC+AE+KNN model produces a mean accuracy of 92.47%. It shows that AE, which reduces the data dimension from 6 to 3, can enhance the data distributions in the three classes in the Diabetes Type dataset. The 3-dimensional data provides the optimum features that reduce intra-class distances while increasing inter-class ones.

Next, the KMC+AE+PNNR model gives a higher mean accuracy of 93.36%. This result indicates that considering multi voters in each of the three classes improves decision making. The KMC+AE+LMPNN model then slightly improves the averaged accuracy to 94.55%. It indicates that the decision-making in LMPNN is conceptually better than PNNR.

Eventually, the KMC+AE+MVMCNN model reaches the highest mean accuracy of 95.24%. It indicates that the proposed multi-commission model is able to improve decision-making in LMPNN. Impressively, this model gives higher accuracy than the DL model that gets 94.02%. All those results indicate that our framework is capable of increasing the KNN accuracy. Besides, they also inform that each procedure can work as it is designed.

Compared to other machine learning models, our framework is also better. For the dataset of PIDD with a similar evaluation scheme, our framework significantly outperforms DT, NB, ANN, SVM, RF, and LR (Cıhan & Coşkun, 2021), LR with the feature selection procedure, ensemble models with max voting and stacking (Rajendra & Latifi, 2021), and Adaptive Boosting (Kalagotla et al., 2021). This result can be achieved since our framework uses proper preprocessing procedures, especially the one that removes noises (incompetent voters) or merges the small clusters into the closest bigger ones.

Based on the evaluations on both PIDD and Diabetes Type datasets, it can be implied that the conventional lazy learner KNN can be enhanced using several procedures in the proposed framework to outperform the other machine learning models and also the modern deep learner. Nevertheless, since the framework uses a KMC sensitive to the randomly generated initial centroids, its performance may decrease for some non-spherical distributed datasets. Besides, the MVMCNN performance may decrease for several datasets that are hard to split into at least two clusters for each class.

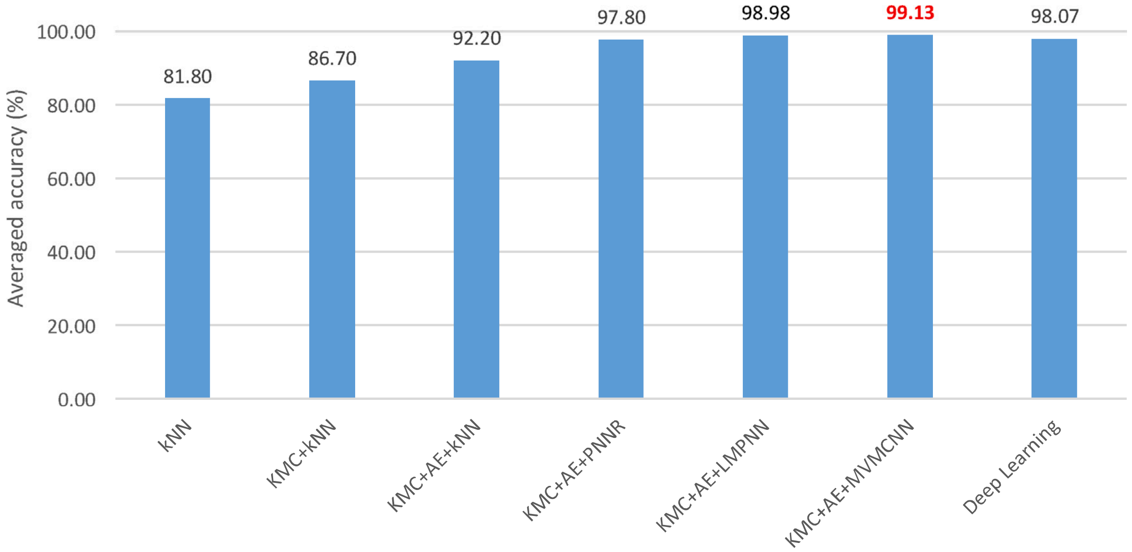


Fig. 5. Accuracies produced by the classification models for the PIDD.

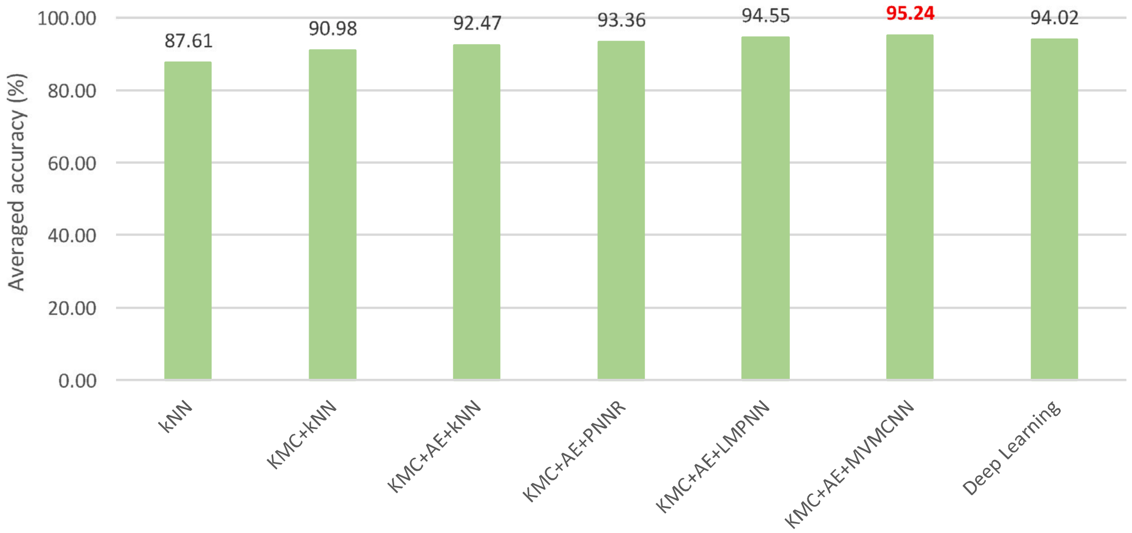


Fig. 6. Accuracies produced by the classification models for the Diabetes Type dataset.

4.3. Processing time of the proposed framework

KNN and DL have similar learning processes but are opposite during testing, where KNN should find k nearest neighbors needing high computation while DL does a forward calculation quickly. Hence, to get fairness, the evaluation of the processing time is carried out by calculating the computational time of the learning process based on 5-FCV using an Intel Core i7-8850H and 16 GB RAM. The computational time of the learning process in KNN is defined as the time needed to find the optimum k by running the KNN 75 times for $k = 1$ to 15 and for 5-FCV, where the optimum k is the one giving the highest mean accuracy. Meanwhile, the computational time of the learning process in the DL-based model is defined as the time needed to find the highest accuracy for 5-FCV using a higher-performance computer with graphical processing units (GPU).

Fig. 7 shows that KNN gives the lowest average processing time of 204.37 s since it has the lowest computational complexity of the procedure of searching k nearest neighbors. Next, the KMC+KNN model increases the average processing time to be 299.94 s. It means that incorporating KMC increases the processing time up to 95.57 s. Meanwhile, the combination models of KMC and AE with KNN, PNNR, and LMPNN increase the average processing time to be around 680 s because of the complexity of AE. Next, the KMC+AE+MVMCNN model gives a longer processing time of 921.09 s since MVMCNN needs additional processes to split each class into two clusters or more and calculate the total distance in each cluster. Finally, the DL-based model needs the highest processing time because it needs some epochs to converge to a stable high accuracy. Furthermore, the slightly longer processing times happened to the Diabetes Type dataset since it has more samples and more classes than PIDD, as illustrated in Fig. 7. These facts inform that the proposed framework of KMC+AE+MVMCNN is considered the most optimum model in terms of accuracy and processing time.

5. Conclusion

A new framework to classify diabetic patients in PIDD and Diabetes Type datasets has been created. Besides, a new KNN variant named MVMCNN is also proposed. The superiority of MVMCNN on the LMPNN has been comprehensively investigated using ten benchmark datasets from UCI Repository. Investigation on binary-class PIDD and multiclass Diabetes Type datasets shows that the proposed framework works very well, where each procedure gives a unique contribution. In addition,

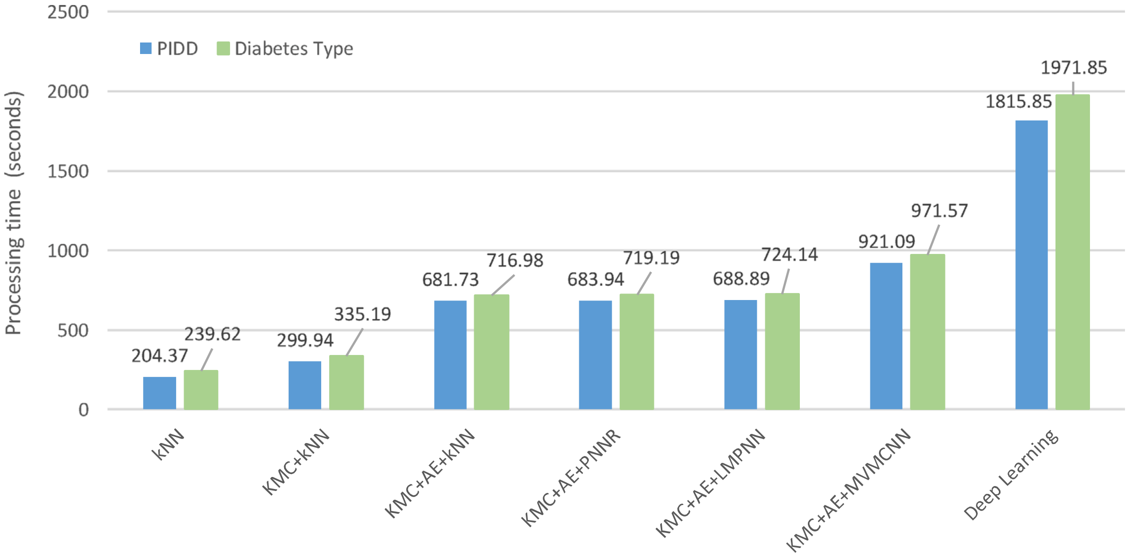


Fig. 7. Processing times given by the classification models for PIDD and Diabetes Type dataset.

the combination of MVMCNN with KMC and AE achieves the best classification rate, which is slightly better than the DL-based model. Nevertheless, the framework performance may decrease for datasets that are non-spherical and hard to split into at least two clusters for each class due to the KMC and MVMCNN, respectively. As future work, MVMCNN will be combined with advanced data imputation, noise removal, and dimensional reduction methods to overcome the limitation and solve more complex classification problems.

CRediT authorship contribution statement

Suyanto Suyanto: Conceptualization, Methodology, Supervision, Funding acquisition, Writing – original draft. **Selly Meliana:** Formal analysis, Project administration, Resources, Review and editing. **Tenia Wahyuningrum:** Data curation, Investigation, Funding acquisition, Review and editing. **Siti Khomsah:** Software, Validation, Review and editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Alehegn, M., Joshi, R. R., & Mulay, P. (2019). Diabetes analysis and prediction using random forest, KNN, Naïve Bayes, and J48: An ensemble approach. *International Journal of Scientific and Technology Research*, *8*, 1346–1354.

Azizah, N., Riza, L. S., & Wihardi, Y. (2019). Implementation of random forest algorithm with parallel computing in R. *Journal of Physics: Conference Series*, *1280*, http://dx.doi.org/10.1088/1742-6596/1280/2/022028.

Bai, X., Wang, X., Liu, X., Liu, Q., Song, J., Sebe, N., & Kim, B. (2021). Explainable deep learning for efficient and robust pattern recognition: A survey of recent developments. *Pattern Recognition*, *120*, Article 108102. http://dx.doi.org/10.1016/j.patcog.2021.108102, URL: https://www.sciencedirect.com/science/article/pii/S0031320321002892.

Barredo, A., Díaz-rodríguez, N., Del, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, *58*(October 2019), 82–115. http://dx.doi.org/10.1016/j.inffus.2019.12.012.

Battineni, G., Sagaro, G. G., Nalini, C., Amenta, F., & Tayebati, S. K. (2019). Comparative machine-learning approach: A follow-up study on type 2 diabetes predictions by cross-validation methods. *Machines*, *7*, 1–11. http://dx.doi.org/10.35940/ijitee.K2155.0981119.

Chaves, L., & Marques, G. (2021). Data mining techniques for early diagnosis of diabetes: A comparative study. *Applied Sciences*, *11*(5), http://dx.doi.org/10.3390/app11052218, URL: https://www.mdpi.com/2076-3417/11/5/2218.

Cıhan, P., & Coşkun, H. (2021). Performance comparison of machine learning models for diabetes prediction. In *2021 29th signal processing and communications applications conference (SIU)* (pp. 1–4). http://dx.doi.org/10.1109/SIU53274.2021.9477824.

International Diabetes Federation (2021). About diabetes. URL: https://idf.org/aboutdiabetes/what-is-diabetes.html.

Gal, Y. (2016). *Uncertainty in deep learning* (Ph.D. thesis), University of Cambridge.

García-ordás, M. T., Benavides, C., Benítez-andrades, J. A., Alaiz-moretón, H., & García-rodríguez, I. (2021). Diabetes detection using deep learning techniques with oversampling and feature augmentation. *Computer Methods and Programs in Biomedicine*, *202*, http://dx.doi.org/10.1016/j.cmpb.2021.105968.

Gou, J., Zhan, Y., Rao, Y., Shen, X., Wang, X., & He, W. (2014). Improved pseudo nearest neighbor classification. *Knowledge-Based Systems*, *70*, 361–375. http://dx.doi.org/10.1016/j.knosys.2014.07.020.

Harrison, O. (2018). Machine learning basics with the K-nearest neighbors algorithm. https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761.

Hayashi, Y., & Yukita, S. (2016). Rule extraction using Recursive-Rule extraction algorithm with J48graft combined with sampling selection techniques for the diagnosis of type 2 diabetes mellitus in the Pima Indian dataset. *Informatics in Medicine Unlocked*, *2*, 92–104. http://dx.doi.org/10.1016/j.imu.2016.02.001.

Howsalya Devi, R. D., Bai, A., & Nagarajan, N. (2020). A novel hybrid approach for diagnosing diabetes mellitus using farthest first and support vector machine algorithms. *Obesity Medicine*, *17*, Article 100152. http://dx.doi.org/10.1016/j.obmed.2019.100152.

Islam Ayon, S., & Milon Islam, M. (2019). Diabetes prediction: A deep learning approach. *International Journal of Information Engineering and Electronic Business*, *11*(2), 21–27. http://dx.doi.org/10.5815/ijieeb.2019.02.03.

Jakka, A., & Vakula Rani, J. (2019). Performance evaluation of machine learning models for diabetes prediction. *International Journal of Innovative Technology and Exploring Engineering*, *8*, 1976–1980. http://dx.doi.org/10.35940/ijitee.K2155.0981119.

Kaggle (2021). Pima Indians diabetes. URL: https://www.kaggle.com/kumargh/pimaindiansdiabetescsv?select=pima-indians-diabetes.csv.

Kalagotla, S. K., Gangashetty, S. V., & Giridhar, K. (2021). A novel stacking technique for prediction of diabetes. *Computers in Biology and Medicine*, *135*, Article 104554. http://dx.doi.org/10.1016/j.compbiomed.2021.104554, URL: https://www.sciencedirect.com/science/article/pii/S0010482521003486.

Kannadasan, K., Edla, D. R., & Kuppili, V. (2019). Type 2 diabetes data classification using stacked autoencoders in deep neural networks. *Clinical Epidemiology and Global Health*, *7*(4), 530–535. http://dx.doi.org/10.1016/j.cegh.2018.12.004.

Kaur, H., & Kumari, V. (2019). Predictive modelling and analytics for diabetes using a machine learning approach. *Applied Computing and Informatics*, http://dx.doi.org/10.1016/j.aci.2018.12.004.

Kavitha, M., & Subbaiah, S. (2019). Implementing classification algorithms for predicting chronic diabetes diseases. *International Journal of Engineering and Advanced Technology*, *8*, 1748–1751. http://dx.doi.org/10.35940/ijeat.F1328.0986S319.

Lukmanto, R. B., Suharjito, Nugroho, A., & Akbar, H. (2019). Early detection of diabetes mellitus using feature selection and fuzzy support vector machine. *Procedia Computer Science*, *157*, 46–54. http://dx.doi.org/10.1016/j.procs.2019.08.140.

Maniruzzaman, M., Rahman, M. J., Al-MehediHasan, M., Suri, H. S., Abedin, M. M., El-Baz, A., & Suri, J. S. (2018). Accurate diabetes risk stratification using machine learning: Role of missing value and outliers. *Journal of Medical Systems*, *42*, 1–17. http://dx.doi.org/10.1007/s10916-018-0940-7.

Mehra, P., Seth, T., & Muhuri, P. K. (2020). Generating quality IF-THEN rules for diabetes using linguistic summarization. In *IEEE international conference on fuzzy systems, Vol. 2020-July*. http://dx.doi.org/10.1109/FUZZ48607.2020.9177662.

Monnier, L., Colette, C., & Owens, D. (2021). Glucose variability and diabetes complications: Risk factor or biomarker? Can we disentangle the "Gordian Knot"? *Diabetes and Metabolism*, *47*(3), Article 101225. http://dx.doi.org/10.1016/j.diabet.2021.101225.

Naz, H., & Ahuja, S. (2020). Deep learning approach for diabetes prediction using PIMA Indian dataset. *Journal of Diabetes and Metabolic Disorders*, *19*(1), 391–403. http://dx.doi.org/10.1007/s40200-020-00520-5.

Pandey, A., & Jain, A. (2017). Comparative analysis of KNN algorithm using various normalization techniques. *International Journal of Computer Network and Information Security*, *9*(11), 36–42. http://dx.doi.org/10.5815/ijcnis.2017.11.04.

Papernot, N., & Mcdaniel, P. (2018). Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. arXiv arXiv:arXiv:1803.04765v1.

Raghavendra, S., & Santosh Kumar, J. (2020). Performance evaluation of random forest with feature selection methods in prediction of diabetes. *International Journal of Electrical and Computer Engineering*, *10*, 353–359. http://dx.doi.org/10.11591/ijece.v10i1.pp353-359.

Rajendra, P., & Latifi, S. (2021). Prediction of diabetes using logistic regression and ensemble techniques. *Computer Methods and Programs in Biomedicine Update*, *1*, Article 100032. http://dx.doi.org/10.1016/j.cmpbup.2021.100032, URL: https://www.sciencedirect.com/science/article/pii/S2666990021000318.

Rajni, & Amandeep (2019). RB-bayes algorithm for the prediction of diabetic in "PIMA Indian dataset". *International Journal of Electrical and Computer Engineering*, *9*, 4866–4872. http://dx.doi.org/10.11591/ijece.v9i6.pp4866-4872.

Sisodia, D., & Sisodia, D. S. (2018). Prediction of diabetes using classification algorithms. *Procedia Computer Science*, *132*, 1578–1585. http://dx.doi.org/10.1016/j.procs.2018.05.122.

Suyanto, S., Yunanto, P. E., Wahyuningrum, T., & Khomsah, S. (2022). A multi-voter multi-commission nearest neighbor classifier. *Journal of King Saud University - Computer and Information Sciences*, http://dx.doi.org/10.1016/j.jksuci.2022.01.018, URL: https://www.sciencedirect.com/science/article/pii/S1319157822000313.

Tjoa, E., & Guan, C. (2020). A survey on explainable artificial intelligence (XAI): Toward medical XAI. *IEEE Transactions on Neural Networks and Learning Systems*, *PP*, http://dx.doi.org/10.1109/tnnls.2020.3027314.

Tripathi, G., & Kumar, R. (2020). Early prediction of diabetes mellitus using machine learning. (pp. 1009–1014). http://dx.doi.org/10.1109/ICRITO48877.2020.9197832.

World, Data (2021). Diabetes type dataset. URL: https://data.world/abelvikas/diabetes-type-dataset/workspace/file?filename=Diabetestype.csv.

Yang, W., Cintina, I., Hoerger, T., Neuwahl, S. J., Shao, H., Laxy, M., & Zhang, P. (2020). Estimating costs of diabetes complications in people <65 years in the U.S. using panel data. *Journal of Diabetes and its Complications*, *34*(12), Article 107735. http://dx.doi.org/10.1016/j.jdiacomp.2020.107735.

Yi, Y., El Khoudary, S. R., Buchanich, J. M., Miller, R. G., Rubinstein, D., Orchard, T. J., & Costacou, T. (2021). Association of age at diabetes complication diagnosis with age at natural menopause in women with type 1 diabetes: The Pittsburgh Epidemiology of Diabetes Complications (EDC) study. *Journal of Diabetes and its Complications*, *35*(3), Article 107832. http://dx.doi.org/10.1016/j.jdiacomp.2020.107832.

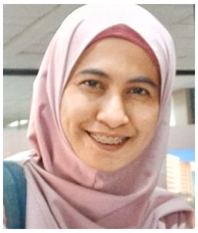
Zeng, Y., Yang, Y., & Zhao, L. (2009). Pseudo nearest neighbor rule for pattern classification. *Expert Systems with Applications*, *36*(2), 3587–3595. http://dx.doi.org/10.1016/j.eswa.2008.02.003.

Zhang, S., Li, X., Zong, M., Zhu, X., & Wang, R. (2018). Efficient kNN classification with different numbers of nearest neighbors. *IEEE Transactions on Neural Networks and Learning Systems*, *29*(5), 1774–1785. http://dx.doi.org/10.1109/TNNLS.2017.2673241.

Zhou, H., Myrzashova, R., & Zheng, R. (2020). Diabetes prediction model based on an enhanced deep neural network. *EURASIP Journal on Wireless Communications and Networking*, *2020*, http://dx.doi.org/10.1186/s13638-020-01765-7.

Zhu, C., Idemudia, C. U., & Feng, W. (2019). Improved logistic regression model for diabetes prediction by integrating PCA and K-means techniques. *Informatics in Medicine Unlocked*, *17*, Article 100179. http://dx.doi.org/10.1016/j.imu.2019.100179.

Suyanto Suyanto received the B.Sc. in Informatics Engineering from STT Telkom (now Telkom University), Bandung, Indonesia, in 1998, the M.Sc. on Complex Adaptive Systems from Chalmers University of Technology, Goteborg, Sweden, in 2006, and the Doctor on Computer Science from Universitas Gadjah Mada in 2016. Since 2000, he joined STT Telkom as a lecturer in the School of Computing. On 01 November 2021, he got a full professor in Artificial Intelligence. His research interests include artificial intelligence, machine learning, deep learning, swarm intelligence, speech processing, and computational linguistics. Scopus ID: 56843751100, Researcher ID: AAB-5223-2021, Publon ID: 4171399, Orcid: https://orcid.org/0000-0002-8897-8091.



Selly Meliana received the B.Sc. in Computer Science from Universitas Indonesia, Jakarta, Indonesia, in 2001 and the M.Sc. in Computing from Telkom University, Bandung, Indonesia, in 2019. She worked as a university lecturer in the Informatics Faculty of PASIM University from 2004 until 2019 and joined Telkom University in early 2020. Her research interests include artificial intelligence, machine learning, computational thinking, and education-related technology. Scopus ID: 57206899266, Researcher ID: AAD-6181-2021, Publon ID: 4215581, Orcid: https://orcid.org/0000-0001-6342-524X.

Tenia Wahyuningrum is a lecturer in the Faculty of Informatics at Institut Teknologi Telkom Purwokerto, Indonesia. She took her undergraduate (S.Kom) at STMIK Widya Utama Purwokerto in 2005, and Master (M.T) in the Electrical Engineering Department at Institut Teknologi Bandung in 2010. He earned a Doctoral from the Department of Computer Science, Universitas Gadjah Mada Yogyakarta, Indonesia. Her research areas of interest are Human–Computer Interaction, Algorithm and programming, and Software Engineering. Scopus ID: 57190841874, Publon ID: O-5088-2016, Orcid: https://orcid.org/0000-0002-4759-2029.

Siti Khomsah, S.Kom., M.Cs. received the B.Sc. on Informatics Engineering from Institute of Technology AKPRIND, Yogyakarta, Indonesia in 2005 and the M.Sc. on Computing from Gadjah Mada University, Yogyakarta, Indonesia, in 2015. Since 2019, she works as a lecturer in the School of Data Science, Institute Technology Telkom Purwokerto. Her research interests are Data Mining, Machine Learning, and Sentiment Analysis. Scopus ID: 57215420947, Orcid: https://orcid.org/0000-0002-9967-4341.